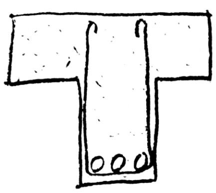
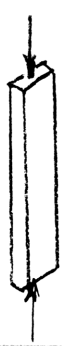
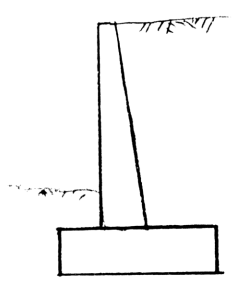
BRIDGEROM

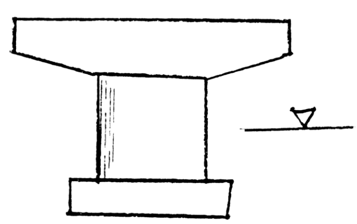
Civil and
Structural Engineering
Software
for the HP-41



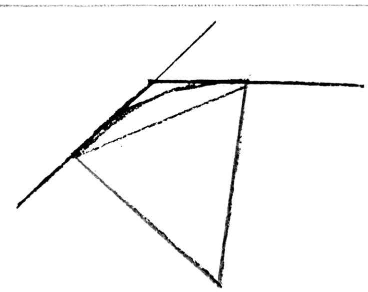
$$\frac{b \pm \sqrt{b^2-4ac}}{2a}$$

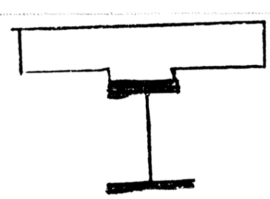






$$PV = \left(\frac{1}{i}\right)^n \times FV$$





USER FLAGS

Flag 00	clear	Full output, with intermediate results
	set	Shortened format to save time
Flag 01	clear	COGO programs use North, East
	set	COGO programs use X, Y
Flag 02	clear	BEEP and TONE warnings disabled
	set	BEEP and TONEs enabled (in subroutine S)
Flag 03	clear	Geometry programs use angles in decimal degrees (dd)
	set	Geometry programs use angles in Degrees-Minutes-Seconds (DMS)
Flag 04	clear	Structural programs follow AASHTO
	set	Structural programs use ACI/AISC/NDS

DOCUMENTATION

Documentation for most programs follows this pattern:

1) OVERVIEW — Details softkey and flag use, SIZE requirements, input explanation, and other quick reference information. For use once the user has become familiar with the program.

2) EXPLANATION — In-depth documentation of how the program works. Explains the program's features, methods, assumptions, and limitations.

3) EXAMPLES — Hand-worked examples to show how the program works and how it is used.

4) LISTING — A program listing with the stack configuration traced through every step, for a deeper look at the program.

SECTIONS OF THIS MANUAL

Part A	(yellow)	Contents; Introductory Material
Part B	(white)	Geometry Programs
Part C	(yellow)	Utility Programs and Routines
Part D	(white)	Structural Programs

NOTICE

These programs and their documentation are computational aids only. They are not intended as a substitute for professional judgement and cannot be used as such. It must be emphasized that engineering calculations involve assumptions allowing the engineer to treat real situations as mathematical problems. These programs do not "solve" real situations; they solve mathematical problems. The correlation between the assumptions made and the real situation must be judged by a competent qualified professional, and the results accorded only as much value as the initial assumptions merit.

As much as possible, these programs follow accepted practice in the profession; and every effort has been made to verify their correctness. However, the seller makes no express or implied warranty of any kind concerning these programs or their documentation, including the sample calculations included. Anyone making use of these programs or the material in this documentation does so at his or her own risk and assumes any and all liability resulting from such use.

A 1

TABLE of CONTENTS

Section A: Introduction

Section B: Geometry

Section C: Utilities

Section D: Structural Programs

LIST OF LABELS

This is a list of all the global alpha labels in the module, in the order of a CAT 2 listing, with the number of bytes necessary to copy the program to main memory (RAM). The labels with the byte counts following them are the first label of that program; i.e. line number 01. Parentheses indicate subroutines which are probably not useful as stand-alone programs.

LABEL	BYTES	DESCRIPTION
PAGE 1		
CHC	515	Horizontal curve routine for coor dinate geometry (COGO)
HC		Horizontal curve program
STOR	1186	Stores coordinate points (COGO)
INV		Finds the distance and azimuth between two points
DUP		Duplicates points (copies to other point numbers)
ANGL		Finds the angle subtended by two points about a third
LLI		Sets a point a given distance along a line defined by two other points
PART		Sets a point part way along a line defined by two other points
(XY)		Returns the difference in X and Y of two pts.
(FS)		Appends _SQ' (square feet) to alpha
(FT)		Appends ' (feet) to alpha
DUMP		Dumps (lists) coordinates of desired points
OFPT		Finds the offset of a point from a line defined by two other points
OFAZ/OFBR		Finds the offset of a point from a line defined by a point and an azimuth/bearing
IAZ/IBR		Intersect two lines, each defined by a point and an azimuth/bearing
(SP)		An internal routine to store points
IPT		Intersect two lines, each defined by two points
PRL		Set two points defining a line parallel to a line defined by two other points
LAN		Set a point by backsight angle and distance
LAZ/LBR		Set a point by azimuth/bearing and distance
(BA)		Convert bearing (angle,quadrant) to azimuth
ADJ		Adjust a traverse by compass rule
(P)		Returns the coordinates of a point

ARCS	838	Intersect two arcs (distance distance)
APT		Intersect an arc with a line defined by two points
AAZ/ABR		Intersect an arc with a line defined by a point and an azimuth/bearing
(PM)		An internal routine used to choose between two possible solutions for arc routines
EXTAN/INTAN		Finds point of external/internal tangency for two circles
PTAN		Finds point of tangency of a line through a point to a circle
C3P		Circle through three points
ATRV		Traverse along an arc by arc distance
ATVA		Traverse along an arc by angle subtended
AFRC		Traverse along an arc, a fraction of its subtended angle
FREE		List of free (unused) point numbers
AREA	209	Finds the area bounded by coordinate points
(VC*)	868	"Port of entry" to v. c. routines
VC		Elevations along a vertical curve across a skewed,crowned or superelevated bridge deck
STA		Stations along a vertical curve at a given elevation
SCOL	385	Steel column analysis (axial only)
(SFa)		Steel column allowable stress subroutine
POB		Coordinate geometry stationing along a line

PAGE 2

BWT	184	Reinforcing bar weight calculation/summation
(S)	722	Set-up routine: SIZE check, title block, etc.
(AX)		Appends the integer part of the X register to ALPHA without disturbing the stack
(N)		Prompts for reinforcing stress and concrete strength
(Fc)		Prompts for concrete strength
(Mc)		Figures concrete cracking moment and reinforcing
(U)		Calculates reinforcing by ultimate strength methods
(AS)		Displays area of steel
(IS)		Appends _SQ" (square inches) to contents of ALPHA register
(IN)		Appends " (inches) to ALPHA
(R)		Calculates minimum reinforcing
(K)		Appends "STRS=....KSI" to ALPHA
(KS)		Appends "=....KSI" to ALPHA
(M=)		Moment display
(FK)		Appends "_'K" (foot-kips) to ALPHA
(M)		Moment prompt routine
QUAD		Quadratic equation solution-prompts A, B, C
(Q)		Quadratic equation solution with C, B, A in stack
(L)		Short underline
(LL)		Long underline
(NO)		Appends "number" sign (#) to alpha
(L*)		Line of asterisks
(CL)		Long line of any user-input character or string
(BD)		Prompts for B" and D" (dimensions in inches)
(W)		WSD/LFD prompt and set-up routine
(AA)		AASHTO/ACI (AISC, NDS) display and set-up
(OK)		Displays OK or NG (no good)
NEO	683	Elastomeric (neoprene) bearing pad analysis
WBM	663	Wood beam analysis
WCOL		Wood column and beam-column analysis
(WFc)		Wood compressive stress routine
PLG	1700	Plate girder analysis
(V)		Viewing subroutine
(PV)		Print-or-view subroutine
(SFb)		Unbraced-length bending stress subroutine

PAGE 3

LFAN	1159	Ultimate strength (load factor) concrete beam analysis
Ic		Working stress concrete beam analysis
WSBM	713	Working stress concrete beam design
LFBM		Load factor concrete beam design
CORBL		WSD or LFD concrete corbel design
VST	260	Concrete shear design
(VS)		Concrete shear design subroutine
COMP	1449	Composite steel-concrete beam analysis
BM		I-beam section properties
(SBM)		I-beam section properties subroutine
(AY)		I-beam section properties subroutine
(FY)		Prompts steel yield stress
FIS	158	Decimal feet to Feet-Inches Sixteenths
FIS+		Feet-inches-sixteenths addition
FIS-		Feet-inches-sixteenths subtraction
FTd		Feet-inches-sixteenths to decimal feet
IP	185	Interpolation
ST		Storage register review
(RH)		Reinforcing ratio check

PAGE 4

FTG	714	Footing analysis under biaxial loads
SHFTG		Short footing analysis program
YZ		Stack manipulation: Y<>Z
YT		Stack manipulation: Y<>T
ZT		Stack manipulation: Z<>T
RP		Rectangular-to-polar for flag 03
PR		Polar-to-rectangular for flag 03
FI	324	PPC ROM financial analysis program
SSS	463	Triangle solution: side-side-side
ASA		Triangle solution: angle-side-angle
SAA		Triangle solution: side-angle-angle
SAS		Triangle solution: side-angle-side
SSA		Triangle solution: side-side-angle
(Dd)		Display degrees according to flag 03 convention
(Pd)		Prompt degrees by flag 03
PIER	789	AASHTO pier loads
3S2	459	Type 3S2 truck moment
3S3		Type 3S3 truck moment
T4		Type T4 truck moment
T3		Type T3 truck moment
2S1		Type 2S1 (HS20) truck moment
T2		Type T2 truck moment
CO		Cosine function for flag 03
SI		Sine function for flag 03
TA		Tangent function for flag 03
ACO		Arc-cosine for flag 03
ASI		Arc-sine for flag 03
ATA		Arc-tan for flag 03
CV	364	PPC ROM curve-fitting program
BC		PPC ROM block register-clearing routine
SV		PPC ROM "solve" program for F(x)=0
SSD	748	Stopping-sight-distance vert. curve
RW		Retaining wall loads
KJ	88	Reinf. concrete WSD constants K, J
LPT		COGO: Locate by bearing thru points
PYR		Avg. area for truncated pyramid vol
P1	62	Append "(" to Alpha
P2		Append ")" to Alpha
PP		Append " -" to Alpha
MU		Append Greek letter mu (u) to Alpha
AT		Append "@" (at) to Alpha
AM		Append "&" (ampersand) to Alpha

FOREWORD

A few notes on the use of this module and its documentation:

Starting with the basics, most programs are accessed by keying XEQ, (alpha), "Program Title," (alpha); then just answer the prompts. Exceptions to this are noted in the individual program's documentation, and are usually one of two types: some programs require input with softkeys (e.g. HC, the horizontal curve program); and some are written to be used as functions or subroutines, requiring input in the stack or specified registers. Again, see the program's documentation. Do NOT use the ENTER key unless directed to by an upward arrow (↑), which occur in prompts for multiple items of input (used mostly by the coordinate geometry routines); just key in the number and hit R/S. Although it was avoided as much as possible, some prompt sequences require values in the stack above the input value; hitting ENTER↑ changes the necessary stack organization. Consult the program listings for more information.

SOFTKEYS

People who have used HP software on the HP-41, and especially on the HP-97, will notice a difference in the way softkeys are used in this module's programs. (Softkeys are the top two rows of keys, A-J, and the shifted top row, a-e, which in USER mode access local alpha labels (A-J and a-e) in the program the user is in.) On the HP-97, input was accomplished by keying in a number, pushing a softkey, keying in the next input, pushing another softkey, and so on. In most of these programs, instead, pressing a softkey brings the program to a given point in the input sequence; it brings up a prompt for the input rather than processing the (already keyed-in) input directly. This method was chosen because of the uncertainty of blindly pushing softkeys. The tradeoff is that the softkeys are slightly faster once gotten used to. Exceptions to this idea are programs HC, the PPC ROM routines FI and CV, and the "function" routines (CO, SI, FIS, etc.) A few programs (SSD and the triangle solution routines) can be used either way.

FLAGS

The use and understanding of flags, especially flags 00 through 04 (hereinafter called "user flags") is essential to the operation of these programs. Please take the time to read the short section following on flags. If you don't know what flags are or how to use them, please consult your HP-41 Owner's Manual.

DISCLAIMER

Please read and understand the disclaimer on a preceding page. Using these programs and their documentation as a learning aid is fine; using them as a sole source of knowledge is not fine at all. And using them to replace engineering knowledge and judgement is criminal.

DOCUMENTATION

Documentation for most programs starts with an "overview" sheet detailing softkey and flag usage, SIZE requirements, and a brief explanation of input. This is intended as a review and reference sheet once the user is familiar with the program. To get familiar with the program, read the full documentation and worked examples which follow. For a really thorough knowledge of the program, consult the listing and stack trace. Documentation for programs which are intended for use as functions or subroutines may deviate from this standard.

SYNTHETIC PROGRAMMING

The user may notice a few unfamiliar symbols and commands in the program listings, prompts, and output. An understanding of these is not necessary to use the program, but for the curious, see the section "Synthetic Programming and Loose Ends."

THE PROBLEM with COMPUTERS

Finally, a plug. Studies of people using computers have noted four major problems. First, many people panic when first using new programs or machines. Second, many people, even experienced users, feel a loss of control. The computer often seems to be running things; the user can see no flow of progress in a task, cannot change his or her mind, and is shown no mercy for input mistakes. Third, there is often complacency on the part of the user over the quality of the work being done because of boredom and the feeling that the computer is doing the work while the user only tends the machine. And fourth, there is a loss of ability to do the work "the old-fashioned way", by hand, due to simple loss of practice. These problems have had an influence on the writing of this module; the programs have been written to help engineers solve problems, not to do it for them. An attempt was made to keep the user informed of the flow of the solution through the display of intermediate results as in hand calculations, and by the use of units on most prompts and output. The user has control of the programs through the simple power of the R/S key to stop an errant run, and softkey-accessible prompt sequences to correct and restart it. Hopefully the combination of units on prompts and output, the documentation, worked example calculations, and stack-traced program listings will work to eliminate most of the confusion and keep the user abreast of the method of solution. And almost any program (with the exception of AX, AY, and the neutral axis iteration portion of WSBM) can be traced by single-stepping through it.

FLAGS

Of the fifty-six flags in the HP-41 operating system, only the first thirty or so are accessible to the user. Of these, flags 11 through 17 and 21 through 29 have specific meanings or functions in certain situations or with certain peripherals, and flags 18 through 20 may have on future peripherals. Flags 00 through 10 are reserved for use in users' programs; flags 18 through 20 can usually also be used.

"USER" FLAGS

The status of flags 00 through 04 is always shown in the display; for this reason these flags are used by this module as mode-control or "User" flags. Typically the user will set these as desired and then pretty much leave them alone, just verifying their status at the beginning of each run. The programs in this module refer to flags 00 through 04 constantly but <u>they never change these flags.</u> However, other programs (either plug-in modules or the user's programs) may alter them, so they should be checked at each use. The modes controlled by these flags are:

Flag 00	clear	Full output, with intermediate results
	set	Shortened format to save time
Flag 01	clear	Coord. geometry programs use North, East
	set	Coordinate geometry programs use X, Y
Flag 02	clear	BEEP and TONEs (e.g. warnings) disabled in subroutine "S"
	set	BEEP and TONEs enabled
Flag 03	clear	Geometry programs use angles in decimal degrees (dd)
	set	Geometry programs use Degrees-Minutes-Seconds (DMS)
Flag 04	clear	Structural programs follow AASHTO
	set	Structural programs follow ACI/AISC/NDS

Flags 05 through 10, 14, 18 through 22, 25 through 27, and 29 are used by many of the programs. These flags are set and cleared as needed by the programs; <u>the user does not need to worry about their status</u> (with the exception of flags 21 and 25--see below) as the programs control it. Flag 14 is used for just one

A 11

thing by this module, and flags 06 and 07 are usually used for only one purpose. Flags 21 through 27 and 29 are used for their stated purpose as given in the HP-41 Owner's Handbook. For the convenience of users trying to trace a program, their functions are:

Flag 06	clear	WSD (Working Stress Design)
	set	LFD (Load Factor or Ultimate Strength Design methods)
Flag 07	clear	"OK" in routine OK
	set	"NG" (no good) in routine OK
Flag 14	clear	Functions L and LL print a line (------)
	set	L and LL print a line of asterisks (***)

Note that flag 14 also has the function of controlling card-reader over-write. As used by these programs it is cleared almost immediately after being set. However, if a card reader is being used there is a risk if the user single-steps through the SF 14 command but stops before the FS?C 14 command which clears it a few steps later.

Flag 21		Set in subroutine S to control the display (stops at AVIEW). If calculator is turned off and then back on again, flag 21 must be set manually to resume without XEQ'ing the program from the beginning. If there is a printer connected and on, this can be neglected.
Flag 22	clear	There has been no numeric input since cleared
	set	There has been numeric input
Flag 25	clear	Normal operation
	set	Error ignore-clear as soon as not needed
Flag 26	clear	BEEP and TONEs disabled
	set	BEEP and TONEs enabled
Flag 27	clear	Normal mode
	set	USER mode-softkeys and key assignments active
Flag 29	clear	Commas and decimal points not displayed
	set	Commas and decimal points are displayed

A12

SYNTHETIC PROGRAMMING AND OTHER LOOSE ENDS

This section will try to explain some of the mysteries of the programming in this package, and pull a few odds and ends of the module together. It probably should have been called "Loose Ends and Synthetic Programming", because the loose ends will be attempted first.

"BYPASSING" A PROMPT

First, a word used often in the documentation: "bypass." To bypass a prompt means to press the R/S (run/stop) key and thus continue without keying in a number or answering a prompt. Many prompts have default values which are used if the prompt is bypassed; consult the "Input Summary" section of each program's documentation. To check the default, backarrow the prompt. R/S may still be pressed to continue.

YES/NO PROMPTS

Most prompts in these programs ask for numerical input; some, however, require a "yes" or "no" answer to a question. This is explained in each program's Input Summary section. These programs use a simple convention: key in 1 for "yes", 0 (zero) for "no".

NUMERIC INPUT

When keying in a number in response to a prompt, simply key it in and press R/S (or the appropriate softkey for a few programs, e.g. HC and SSD). Do not use the Enter↑ key unless directed to by an upward arrow (↑) in the prompt. Although it was avoided as much as possible, some input sequences require values in the stack above the input; hitting Enter↑ alters this stack organization. The coordinate geometry programs typically require multiple input; a tradeoff was made for increased speed and convenience at the cost of increased concentration needed to avoid errors.

SOFTKEYS

Softkeys are the top two rows of keys, and the top row shifted, when USER mode is on. These are labelled A through J and a through e; When pressed (in USER mode), they try to find and execute a label of the same letter within the program that the calculator is currently set to. In order to use these keys, it is useful to understand the hierarchy the calculator follows when a softkey is pressed. First, it checks to see if a function or program has been assigned to the key. If not, the calculator looks for the corresponding local alpha label within the current program. If it finds the label it starts running the program there; if not it executes the function that the key would have executed right away if not in USER mode. If the key wasn't a softkey, the second step is skipped. To summarize:

1) Key assignment
2) Local alpha label
3) Native function

There are a couple of things to be learned from this long-winded discussion. One, it's not a good idea to assign functions or programs to the top two rows of keys because they can't function as softkeys; and two, if trying to execute a native function in USER mode, the search process is maddeningly slow. Turn off USER or assign the function to its own key (yes, contradictions).

LABELS

Having explained everything there is to know about softkey hierarchy, on to labels. Labels too have a hierarchy, easy to remember: CAT 1, CAT 2, CAT 3. When the user executes a label from the keyboard (i.e. by spelling it out), the calculator looks for it first in Catalog 1 (user-written programs), then Catalog 2 (programs in ROM, such as this module), then Catalog 3 (native HP-41 functions). This is useful because a program can be copied down into Main Memory (RAM) from ROM and tailored to the user's wishes (needless to say, at some risk of introducing bugs). Then when executed, the calculator finds and runs the altered version in RAM rather than its predecessor in ROM. It also means that a program in either RAM or ROM can have the same label as a native function, and still be found and executed.

INITIALIZATION SUBROUTINE "S"

Most of the programs use subroutine "S" to initialize the calculator. Among other things, it sets flag 21 for display control, checks the SIZE required, sets flag 26 according to the status of flag 02 to silence or enable BEEPs, and sets the trig mode to DEG. If a printer is attached, "S" also prints a header with the time and date (if a Time Module is present) and gives the opportunity to print a title, which is limited to 24 characters. The time/date line works best with the display set to FIX 4 or less. For keying the title, "S" turns Alpha mode on and then off again; simply key in the title and press R/S. The title may, of course, be bypassed.

XROM vs. XEQ

There may be some confusion about the appearance of XROM instead of XEQ in program listings. XROM means "execute from ROM" (Read Only Memory, i.e. plug-in modules). Many of the programs in this module may be used as subroutines in users' programs. They are entered into a program using the XEQ key, just as for a native HP-41 function or a user-written subroutine. The label type can be told by how it looks in a program listing:

```
23 COS          Execute native HP-41 Cosine
                function
24 XEQ "COS"    Execute user program COS from Main
                Memory
25 XROM "COS"   Execute program COS from ROM
```

Some program listings in this documentation contain both XEQ and XROM statements. This is due to the revision process; XEQ statements have been revised since the first version of this software was published. However, all these (printed) global XEQ statements were changed during the module manufacturing process and should now (hopefully) list as XROM.

LOGIC SEQUENCES

Another confusing thing which pops up frequently in these programs is the use of double and triple logic sequences. These are used for the conditionals "if A or B then..." or "if A and B then..." For example, to jump to label 14 if either flag 06 or 08 is set, the sequence is:

```
26  FC? 06     Opposite of test A  (Not A)
27  FS? 08     Test B
28  GTO 14
```

To do an "and" test requires a third conditional which must never be true. FS? 53 is often used since it always tests clear; if it is certain that the number in X is positive, for example, the conditional X<0? can be used, saving a byte of memory. For example, to jump to label 14 only if flags 06 and 08 are both set:

```
29  FS? 06     Test A
30  FC? 08     Opposite of test B (Not B)
31  FS? 53     Always false
32  GTO 14
```

"SIGN" FUNCTION

Many of these programs use the HP-41 function SIGN, a relatively little-known function. Its main usefulness lies in the fact that it stores the previous contents of X in LastX. It does this faster than most other functions, and uses only one byte.

"PRGM" ANNUNCIATOR

During long program runs with a printer attached, the message or output last displayed will remain in the display instead of being replaced by the "flying goose." However, the PRGM annunciator stays on to verify that the program is running. If this does not happen, the message is a prompt and the program is expecting input.

SYNTHETIC PROGRAMMING

The rest of this article deals with a subject that has come to be called "synthetic programming." Synthetic programming is a collection of techniques that allow the user to synthesize non-standard commands and display characters, and to access and manipulate storage registers normally not available to the user.

SHORT-FORM EXPONENTS

First, short-form exponents. A number such as 1000 may be keyed into a program as 1 E3, saving one byte of memory. However, the leading "1" is not needed. Removing it with synthetic programming methods results in the program line E3, which still means 1000 but now takes only two bytes and also executes faster. Similarly, 100 can be programmed as E2, 10 as E1, and 0.001 as E-1. Also, 1 can be represented by E0 or just E. The replacements for 1 and 10 don't save any bytes; however, they do execute faster. Zero can also be represented by a lone decimal point. Again, this (non-synthetic) representation saves no program space but is faster. To demonstrate any of these, just single-step through them.

NOP's

The conditionals DSE and ISG are often useful as simple decrement/increment functions. To achieve this, these commands are followed with a command which does nothing, or at least nothing harmful. These no-operation (NOP) commands are often short-form local labels or a STO X command. However, the only true NOP which has no effect and also takes only one byte (STO X takes two) is a synthetic command called Text-0. Text strings all have headers which tell the length of the string to follow. A Text-0 command is a header for an alpha string of length zero (it even sounds useless). It prints as " "; it displays as a superscript T (text symbol) with nothing after it.

A16

NON-STANDARD SYMBOLS

The prompts and output descriptors use a few symbols not available on a standard HP-41. These include foot and inch tics (',"), the "number" hash-mark symbol (#), and parenthesis. Some of these have been made available to the non-synthetic programmer through labels FT, IN, NO, and others; these labels simply append the symbol to whatever is in the alpha register. Using these labels actually takes one less byte than the actual synthetic append they execute, although it is a little slower. The HP-41 also has available to its display several other symbols which were not thusly made available but which can be used by anyone willing to learn synthetic programming: brackets [ ], the Greek symbol lambda (sort of), and an exclamation point. If you don't mind a messy display, there are many symbols which display as a starburst but which print as themselves: nearly the entire Greek alphabet, our lower-case alphabet, as well as many other characters.

SYNTHETIC STORAGE REGISTERS

One of the most useful things available through Synthetic Programming is the ability to access a number of the calculator's internal storage registers for use (within certain strict limitations) as regular storage registers, or to control certain aspects of the machine's behavior. These registers are located at the bottom end (i.e. they have the lowest-numbered addresses) of the calculator's memory. These first registers have been given the name "Status Registers" because their contents are recorded by the card-reader function WSTS. Starting from the bottom, the Status Registers and their functions are as follows:

T Z Y X L	Stack and Last X
M or [N or \ O or] P or /	Alpha register
Q	Printer use, internal "scratch"
⊢	Unshifted key assignments
a b	Subroutine RTN chain and current line number
c	Cold start constant, "curtain"
d	Flag register
e	Shifted key assignments

A17

The first five are the stack and LastX registers, and are accessible through normal means. Synthetic programming is not needed to use these.

Next comes the Alpha "register", actually four registers of seven bytes each. Under usual conditions the Alpha register is 24 bytes long; the last four bytes of P are used for other things by the calculator. Because of this, register P has some very confining limitations on it and finds only infrequent use; see the next paragraph. However, registers M, N, and O have no such limitations; the only caution regarding their use is that any text introduced into Alpha, by the program or the user, wipes out the data stored in these registers. They are therefore well suited to carrying intermediate constants needed only in calculations leading up to a given item of output.

Data stored in register P is altered by, among other things, number entry (but <u>not</u> RCL), any operation that causes a number to be displayed, and SST'ing (single-stepping) through a program. Thus, the few programs which use register P (AX, the Trig functions, AY, and the neutral-axis iteration portion of WSBM) cannot be single-stepped through. If you want to trace through any of these, either put a STOP after the area and R/S to get through it, or copy the program down into main memory and change command using register P to one using a numbered storage register.

The next two registers, Q and ⊢, are used too frequently by the calculator to be of much outside use. Register Q is used occasionally by some advanced synthetic programmers, but it does not appear in these programs.

Registers a and b are the line number and subroutine RTN chain registers. Register b contains the current line number and the first two and a half return addresses pending; register a contains the next three and a half return addresses. Each address, as may be guessed, is two bytes long. Every time a program comes to a RTN, these two registers shift two bytes, and the first RTN address pending becomes the current line number. Register a may thus be used if no XEQ's or RTN's are met and if the subroutine return depth does not exceed two. If an XEQ or RTN occurs, the data in registers a and b is shifted two bytes left or right, trashing the data. Even if it is shifted back by the opposite function, the data is altered. If the program has more than two RTN's pending, there is a worse problem: part of your data gets used for a return address, sending the calculator off into another dimension. The program pointer usually ends up somewhere in the key assignment registers and tries to interpret your key assignments as program steps. (If this happens, press CAT 1 to get back home, then fix the problem before it happens again.)

Register c contains two important pieces of information: the location of the "curtain", the divider between data and program storage; and a number called the "cold-start constant." When the calculator is turned on, and at certain other times, this number is checked by the processor. If it is anything other than the expected constant, the processor thinks something terrible has happened and panics. MEMORY LOST results. Register c is not used by any of these programs.

Register d contains all the system and user flags, 00-55. An HP-41 register is 7 bytes of 8 bits each; each flag is one bit. A number of things are controlled by different flags: the display setting (FIX, SCI, or ENG, and the number of places displayed), the trigonometric mode (DEG, RAD or GRAD), many print and display functions, and other things. Register d is used in these programs mostly in subroutine AX ("Append X"), which recalls d, changes display setting to FIX 0, clears flag 29 (thus getting rid of the decimal point), appends a number to Alpha using this display configuration, and then returns the display to its former status by re-storing to register d the value previously recalled.

The last Status Register is register e, which contains the shifted key assignments. Like register ⊢, e is not used by any of the programs in this package.

REFERENCES

There are several reference books available on the HP-41 in general and Synthetic Programming in particular. The two best are:

EXTEND YOUR HP-41 by W. Meir-Jedrzejowicz, and
HP-41 SYNTHETIC PROGRAMMING MADE EASY by Keith Jarett

Both are available from EduCALC Mail Store,
27953 Cabot Road
Laguna Niguel, CA 92677

GEOMETRY

PROGRAMS

BRIDGE DECK VERTICAL CURVE ELEVATIONS

xeq VC (SIZE 010) to find elevations along profile grade
(# OFSTS bypassed or answered zero)

or

(SIZE 017 + number of offsets) to find elevations across a bridge deck or street

or

(SIZE 023 + number of offsets) to find elevations across a bridge deck or street using separate crown drops for each point

SOFTKEYS (USER mode ON)

c	Input new crown information; same as C for profile grade calculations (# OFSTS = 0)
d	Input new offsets; same as C for profile grade calc's
e	Change base (P.I.) elevation (e.g. for beam seat elev.)
C	New INTERV or STA prompt; access VC from STA
E	Access STA from VC once curve parameters are input

USER FLAGS

Flag 00	clear	Full output
	set	Output of M and HI/LO point skipped; STA display skipped if using INTERV
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled (used if STA finds OFF CURVE error)
Flag 03	clear	VC prompts SKEW in decimal degrees
	set	VC prompts SKEW in D.MS (for # OFSTS not zero)

INPUT SUMMARY

Prompt	Input	Default
# OFSTS	Number of offsets from center-line, for figuring SIZE needed	Zero(P.G.)
STA PI	Station of the PVI, in feet	No default
EL PI	Elevation of the PVI, in feet	No default
VC L'	Vertical curve length, in feet	No default

Prompt	Input	Default
SL1%	Slope or grade of the back (left) tangent, in percent	No default
SL2% (only if VC L' is not 0)	Slope or grade of the forward (right) tangent, in percent	SL1%

Note: prompts in parenthesis do not appear if the # OFSTS (number of offsets) prompt was answered zero or bypassed.

(SKEW)	Skew of the line across a bridge deck (a "square" bridge has 0^o skew)	No default
(CRN1'/')	Slope of the crown from the centerline to a "break" point, in feet per foot. Bypass to enter crown drops (CRN DROP) directly for each offset.	Zero
(TO:)	Distance of the break point from centerline, in feet	12
(CRN2'/')	Slope of the crown from the break point out, in feet per foot	0.020
(OFFSET#1', etc.)	Offset of a desired line from CL in feet (e.g. a beam line)	Zero
(CRN DROP 1, etc.)	Crown drop to the desired line from profile grade; a rise is negative. Only used if CRN1 was input as zero	No default
INTERV	Station interval at which to calculate elevations, in feet	Previous INTERV
STA or POB STA	Station at which to calculate or begin calculating elevations (in feet: 28+50.00 is input 2850)	No default
EL CH (from softkey e)	Change in baseline elevation; up is positive, down is negative	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(M)	The middle ordinate, in feet
(HI/LO @...)	The station of the high or low point, if one exists
(EL)	The elevation of the high or low point

L1, L2, etc.	The elevation at offset #1, etc., on the left side of centerline
CL EL	The elevation at centerline
R1, R2, etc.	The elevation at offset #1 on the right side of centerline

PROGRAM FLAGS

Flag 05	clear	On or tangent to a vertical curve
	set	VC L' = 0 (tangent--no vertical curve)
Flag 06	clear	High or low point exists; two stations for a given elevation
	set	No high or low point; one station for a given elevation
Flag 07	clear	Left-ahead skew (positive value)
	set	Right-ahead skew (negative value)
Flag 08	clear	Profile grade only
	set	Full bridge deck with offsets
Flag 09	clear	VC
	set	STA
Flag 10	clear	Have crown slope information
	set	Have individual crown drop points
Flag 11	clear	Normal use
	set	Portions used as subroutines
Flag 18	clear	Skewed--figure both sides of crown
	set	Crowned with 0^{o} skew, so only figure one side of crown
Flag 19	clear	Station is on a vertical curve
	set	Station is on a tangent

STORAGE REGISTER USE

Register	Value
00	Sta. PI
01	SL1%
02	SL2%
03	Sta. BVC
04	r/200 $(r=g_2-g_1/L)$
05	Sta. EVC

Register	Value
06	Interval
07	El. BVC
08	Input station
09	El. PI

Note: the following registers are not used if only profile grade calculations are being done (i.e. # OFSTS was answered zero).

10	Crown drop
11	tan (skew)
12	Crown break point
13	Crown2 - Crown1
14	Crown1
15	Counter
16	Offset at centerline (zero)
17	Offset #1
18	Offset #2
19	Offset #3
20	Offset #4
21	Offset #5
22	Offset #6
23	Offset #7/Crown drop #1
24	Offset #8/Crown drop #2
	etc.

VERTICAL CURVE SOLUTIONS for ELEVATION

Programs VC and STA solve parabolic, symmetrical (equal-tangent) vertical curves. Both programs work on a single vertical curve and the tangent on either end (assumed to extend indefinitely). They both require input of the PI (point of vertical intersection) station and elevation, the back and forward grades, and the vertical curve length. Stations are expressed in feet, without the "+"; grades are in percent. See the examples for clarification.

VC calculates either the elevation of a given station on the profile grade; or, elevations at specified offsets on a crowned section along a line at a given skew to centerline. It prompts first for an interval, INTERV, on which to figure elevations (typically 50 or 100 feet). If this prompt is answered, the program prompts POB STA (beginning station) and then figures elevations at every interval, displaying the station and its elevation. If the INTERV prompt is bypassed, VC prompts repeatedly for stations (STA) at which to find the elevation. Softkey C accesses these prompts.

Typical applications of VC might be elevations of beam lines on a bridge deck, or gutter and curb lines along a street. For multiple elevations along a line at a given station, in addition to the basic vertical curve data mentioned above, VC requires the number of offsets to be calculated, the skew, crown configuration (see page 3), and the offsets desired. It then calculates the elevations at all the offsets on the left side of centerline, the centerline elevation, and then the elevations on the right side. For a crowned deck with a zero-degree skew, only one side is calculated since the crown is assumed symmetrical.

INPUT DETAILS

"Stations" are input and output in feet. For example, if the PI is at station 28+50.00, answer the STA PI prompt with 2850. Length of vertical curve (VC L') is also in feet.

Tangent grades are input in percent, as mentioned, while crown cross-slopes are input in feet per foot. Don't waste a lot of time looking for a good reason for this discrepancy; there isn't one.

Once the required data is input, VC will either prompt for a beginning station and an interval, or will prompt repeatedly for the station (STA) at which to calculate the elevation or elevations. If the interval prompt (INTERV) is answered with a non-zero number (let's say, 50) the program will ask for the beginning station (POB STA), and then will figure elevations at fifty-foot intervals from there, displaying first the station and then the elevation or elevations. It will continue doing this until it is stopped; if printing, press R/S to stop it. Then, in USER mode, press softkey C. (If the X key has something assigned to it, key GTO (alpha) C (alpha) instead.) This gets you back to the INTERV prompt.

If the INTERV prompt is bypassed (or answered zero) the programs repeatedly prompt for the station (STA) at which to find an elevation.

ELEVATIONS on TANGENT

If a point is off the vertical curve, VC will append a warning, (T), that the point is on the tangent rather than on the curve. The elevation is figured correctly; the warning is simply a reminder to verify that you're still working on the tangent and not on a preceding or following curve. This is especially important when using VC on sharply skewed bridges; the stations of the offset points are never explicitly displayed, and corners (or ends of long wingwalls) may encroach on a closely preceding or following curve.

If the area you're interested in is entirely on a tangent, input a vertical curve length of zero. The program will only prompt for one grade and the (T) warning will not be displayed, saving annoying display-scrolling time.

ADDITIONAL PARAMETERS CALCULATED

After input of the basic vertical curve parameters (PI station and elevation, grades, and length) the program calculates M (the middle ordinate), and, if there is one (if the grades are of opposite sign), the high or low point station and elevation. If this information is not wanted and you wish to speed execution, set flag 00 and it will not be displayed.

SKEW

In keeping with the standard flag usage of this package, program VC prompts for skew in either decimal degrees (dd) or degrees-minutes-seconds (DMS) depending on whether flag 03 is clear or set. A right-hand-forward skew is defined as negative; a LHF skew is defined as positive. Skews are measured from a line normal to CL Roadway; see the examples.

CROWNS

Crowns are defined by the slope from the centerline to a break point at a user-defined offset, and another slope from the break point on out. Unlike the tangent slopes, which are input in percent, the crown slopes are input in feet per foot per standard practice; see the examples for clarification. Crowns are assumed to be symmetrical about centerline. After the crown information is input, VC asks for offsets (any number can be stored) from the centerline (assumed measured normal to the centerline); these are usually input from the centerline on out, but they need not be in any order.

CROWN "DROP"

If a section is not easily described by a sloping crown, the program offers the option of directly inputting the drop from the profile. Answering the CRN1 (crown one, the slope from centerline to the break point) prompt with zero, or bypassing the prompt, directs the program to prompt for a crown drop in feet after each offset prompt. If the point is above the profile grade (e.g. a curb) the crown drop will be a negative number. When the CRN DROP feature is used, only six offsets can be stored because CRN DROP 1 is stored in the seventh offset's register. Because of this overlap, the size checker at the beginning of the program doesn't work. The minimum SIZE required is (23 + number of offsets), and the user must verify before starting that there are enough storage registers available. Since the maximum number of offsets is six with this feature, a SIZE of at least 029 will always work.

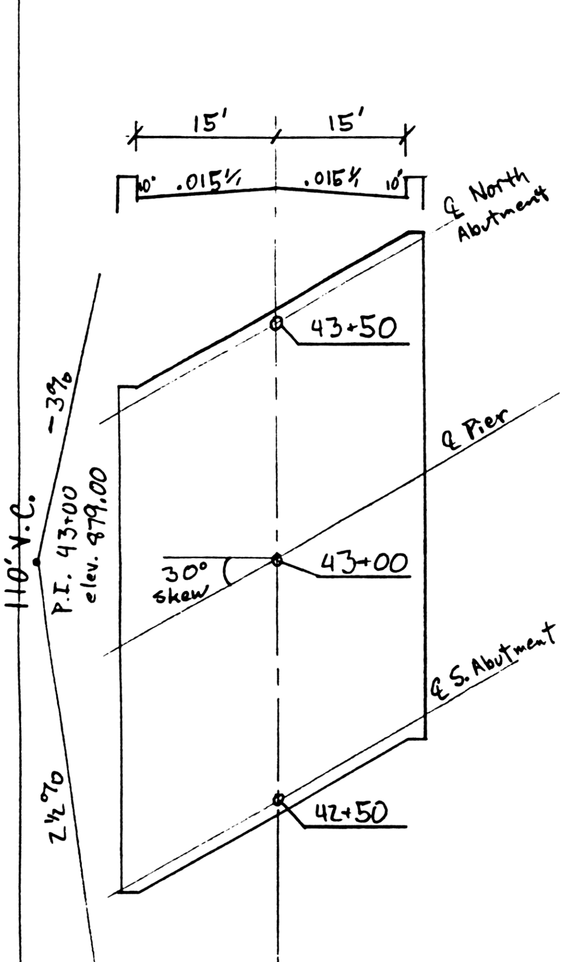
OUTPUT

If the calculator is shut off during a run, and then turned on again and restarted by pressing R/S or a softkey, flags 18 and 19 are cleared. This may affect the (T) warning and the displaying of only one side of a deck for 0^{o} skew, but the numbers are correct. See the "Program Flags" section.

ELEVATION CHANGE

If the user needs to subtract a distance from each elevation, the program can do it directly. Softkey e brings up the prompt EL CH (elevation change). It adds this number to the P.I. elevation (and BVC and EVC elevations) stored at the beginning of the run. Thus, if VC was being used to calculate bridge deck elevations and the user now wants to calculate beam seat elevations (e.g. for a superstructure height of 5.90 feet), press softkey e, answer the prompt with -5.90 (5.90, chs), and press R/S. This feature works with both VC and STA.

"VC" - Bridge Deck Elevations




```
                XEQ "VC"
# OFSTS
          2.0000    RUN
SIZE<19
                SIZE 019
                XEQ "VC"
# OFSTS
          2.0000    RUN
11:45:34 AM  12/22/88

TITLE:
30d SKEW BRIDGE     RUN
************************
STA PI
      4,300.0000    RUN
EL PI
        879.0000    RUN
VC L'
        110.0000    RUN
SL1%
          2.5000    RUN
SL2%
         -3.0000    RUN
M=-0.7563'
HI/LO @ 4,295.0000
EL=878.2500
---------
```

```
SKEW: L+,R- (dd)
        -30.0000    RUN
CRN1'/,
           .0150    RUN
TO:
         20.0000    RUN
CRN2'/,
                    RUN
OFFSET#1'
          7.5000    RUN
OFFSET#2'
         15.0000    RUN
************************
INTERV
         50.0000    RUN
POB STA
      4,250.0000    RUN
-----------------------
STA 4,250.0000:
 L2 EL=877.3085(T)
 L1 EL=877.5291
CL EL=877.7438
 R1 EL=877.7240
 R2 EL=877.6949
---------
STA 4,300.0000:
 L2 EL=878.0217
 L1 EL=878.1374
CL EL=878.2438
 R1 EL=878.1157
 R2 EL=877.9783
---------
STA 4,350.0000:
 L2 EL=877.4882
 L1 EL=877.4956
CL EL=877.4938
 R1 EL=877.2575
 R2 EL=877.0152(T)
---------
STA 4,400.0000:
 L2 EL=876.0348(T)
```

BVC Sta. $= 43 - \frac{1.1}{2} = 42+45.00$

elev. $= 879.00 - (.55 \times 2.5) = 877.625$

$$Elev_x = ax^2 + bx + c \; ;$$

$$a = \frac{g_2 - g_1}{2L} = \frac{-3 - 2.5}{(1.1) \times 2} = -2.50$$

$$b = g_1 = 2.50$$

$$c = 877.625$$

Find elevations @ ℄ and Offsets of 7.5' and 15' along ℄ South Abutment ; * ⇒ off V.C.

OFFSET	× Tan(30°)	STATION	ELEV.	CROWN	ELEV.
15' Left	-8.6603	42+41.3397*	877.5335	.225'	877.3085
7.5' Left	-4.3301	42+45.6699	877.6416	.1125'	877.5291
℄	0	42+50	877.7438	0	877.7438
7.5' Right	4.3301	42+54.3301	877.8365	.1125	877.7240
15' Right	8.6603	42+58.6603	877.9199	.225'	877.6949

Elevations on Bridge Deck with Curbs

Use same vertical curve and bridge deck; find elevations with CRN DROP.

Curb height = 10"

Sta. PI = 43+00.00
El. PI = 879.00
V.C. = 110.0'

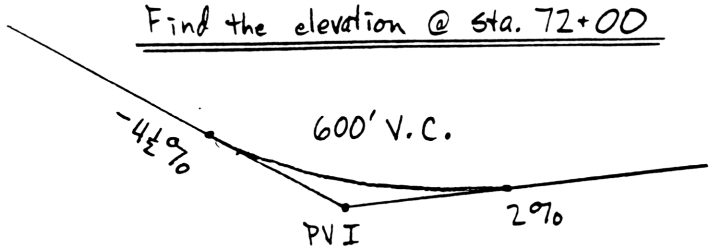
Crown drops

$7.5' \times .015' = .1125'$
$15' \times .015' = .225'$
$15' \times .015' - (10''/12) = -.6083'$

```
              XEQ "VC"
# OFSTS
          3.0000    RUN
10:29:01 AM  12/27/88

TITLE:
BRIDGE DECK         RUN
************************
STA PI
      4,300.0000    RUN
EL PI
        879.0000    RUN
VC L'
        110.0000    RUN
SL1%
          2.5000    RUN
SL2%
         -3.0000    RUN
M=-0.7563'
HI/LO @ 4,295.0000
EL=878.2500
---------
SKEW: L+,R- (dd)
        -30.0000    RUN
CRN1'/,
                    RUN
OFFSET#1'
          7.5000    RUN
CRN DROP 1
           .1125    RUN
OFFSET#2'
         15.0000    RUN
CRN DROP 2
           .2250    RUN
OFFSET#3'
         15.0000    RUN
CRN DROP 3
          -.6083    RUN
************************
INTERV
         50.0000    RUN
POB STA
      4,250.0000    RUN
-----------------------
STA 4,250.0000:
 L3 EL=878.1418(T)
 L2 EL=877.3085(T)
 L1 EL=877.5291
CL EL=877.7438
 R1 EL=877.7240
 R2 EL=877.6949
 R3 EL=878.5282
---------
STA 4,300.0000:
 L3 EL=878.8550
 L2 EL=878.0217
 L1 EL=878.1374
CL EL=878.2438
 R1 EL=878.1157
 R2 EL=877.9783
 R3 EL=878.8116
---------
STA 4,350.0000:
 L3 EL=878.3215
 L2 EL=877.4882
 L1 EL=877.4956
CL EL=877.4938
 R1 EL=877.2575
 R2 EL=877.0152(T)
 R3 EL=877.8485(T)
---------
STA 4,400.0000:
 L3 EL=876.8681(T)
 L2 EL=876.0348(T)
 L1 EL=876.0174(T)
CL EL=876.0000(T)
 R1 EL=875.7576(T)
 R2 EL=875.5152(T)
 R3 EL=876.3485(T)
---------
STA 4,450.0000:
```

Profile Grade Calculation

Find the elevation @ Sta. 72+00



Sta 73+25.00

Elev. 433.14

BVC Sta. = 70+25.00

$$\text{Elev.} = \text{El.PVI} - \frac{L}{2} \cdot G_1$$

$$= 433.14 - (3)\times(-4.5) = 446.64$$

EVC Sta. = 76+25.00

$$\text{Elev.} = \text{El.PVI} + \frac{L}{2} \times G_2 = 439.14$$

$$\text{Elev.} = ax^2 + bx + c:$$

$$a = \frac{g_2 - g_1}{2L} = \frac{2-(-4\frac{1}{2})}{2 \times 6} = .5417$$

$$b = g_1 = -4.50$$

$$c = \text{elev. BVC} = 446.64$$

Elev. @ Sta. 72+00: $x = 72 - 70.25 = 1.75$

$$\text{Elev.} = .5417\,(1.75)^2 - 4.50 \times 1.75 + 446.64$$

$$= 440.4239$$

$$M = \frac{g_2 - g_1}{2L} \times \left(\frac{L}{2}\right)^2 = \frac{(g_2 - g_1)L}{8}$$

$$= \frac{(2-(-4.5))\times 6}{8} = 4.475'$$

Low Pt. Sta.: $2ax + b = 0$

$$\frac{2(g_2 - g_1)}{2L}\,x + g_1 = 0$$

$$x = \frac{-g_1 L}{g_2 - g_1} = \frac{-(-4.5)\times 6}{2-(-4.5)} = 4.1536 \text{ Sta.}\ ;\ (70+25) + (4+15.38) = 74+40.38$$

```
                         XEQ "VC"
# OFSTS
                              RUN
11:50:10 AM  12/22/88

TITLE:
EXAMPLE                       RUN
************************
STA PI
         7,325.0000       RUN
EL PI
            433.1400      RUN
VC L'
            600.0000      RUN
SL1%
             -4.5000      RUN
SL2%
              2.0000      RUN
M=4.8750'
HI/LO @ 7,440.3846
EL=437.2938
************************
INTERV
             50.0000      RUN
POB STA
          7,150.0000      RUN
-----------------------
STA 7,150.0000:
 EL=441.8614
---------
STA 7,200.0000:
 EL=440.4239
---------
STA 7,250.0000:
 EL=439.2572
---------
STA 7,300.0000:
 EL=438.3614
---------
STA 7,350.0000:
 EL=437.7364
---------
                          SQRT
                         XEQ C
************************
INTERV
                              RUN
STA
         7,200.0000       RUN
 EL=440.4239
---------
STA
         7,300.0000       RUN
 EL=438.3614
---------
```

Bypass for only P.G. calc - need only SIZE 010

Oops- USER not on

VERTICAL CURVE STATIONS

xeq STA (SIZE 010) to find the station or stations at which an elevation occurs

SOFTKEYS (USER mode ON)

c	Same as C for STA
d	Same as C for STA
e	Change base (P.I.) elevation (e.g. for beam seat elev.)
C	Jump to program VC; new INTERV or STA prompt
E	Access STA from VC once curve parameters are input

USER FLAGS

Flag 00	clear	Full output
	set	Output of M and HI/LO point skipped
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled (used if STA finds OFF CURVE error)

INPUT SUMMARY

Prompt	Input	Default
STA PI	Station of the PVI, in feet	No default
EL PI	Elevation of the PVI, in feet	No default
VC L'	Vertical curve length, in feet	No default
SL1%	Slope or grade of the back (left) tangent, in percent	No default
SL2% (only if VC L' is not 0)	Slope or grade of the forward (right) tangent, in percent	SL1%
EL	Elevation at which to calculate the station or stations	No default
EL CH (from softkey e)	Change in baseline elevation; up is positive, down is negative	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(M)	The middle ordinate, in feet
(HI/LO @...)	The station of the high or low point, if one exists
(EL)	The elevation of the high or low point
STA	The station at which the given elevation occurs, if there is only one
STA1	The first station of two at which the given elevation occurs
STA2	The second station; will be the same as the first if at a HI/LO point
OFF CURVE	A warning that the elevation input does not exist on the curve (i.e. is higher than the high point or lower than the low point)

PROGRAM FLAGS

Flag 05	clear	On or tangent to a vertical curve
	set	VC L' = 0 (tangent--no vertical curve)
Flag 06	clear	High or low point exists; two stations for a given elevation
	set	No high or low point; one station for a given elevation
Flag 07	clear	Elevation is on curve-solution possible
	set	Elevation is OFF CURVE
Flag 08	clear	Profile grade calcs only
	set	# OFSTS not zero
Flag 09	clear	VC
	set	STA
Flag 10	clear	BVC lower than EVC (STA only)
	set	EVC lower than BVC (STA only)
Flag 11	clear	Normal operation
	set	Figuring HI/LO elevation

STORAGE REGISTER USE

Register	Value	
00	Sta. PI	
01	SL1%	
02	SL2%	
03	Sta. BVC	
04	r/200	$(r=g_2-g_1/L)$
05	Sta. EVC	
06	El. EVC	
07	El. BVC	
08	Input elevation	
09	El. PI	

B13

VERTICAL CURVE SOLUTIONS for STATION

Programs VC and STA solve parabolic, symmetrical (equal-tangent) vertical curves. Both programs work on a single vertical curve and the tangent on either end (assumed to extend indefinitely). They both require input of the PI (point of vertical intersection) station and elevation, the back and forward grades, and the vertical curve length. Stations are expressed in feet, without the "+"; grades are in percent. See the examples for clarification.

STA finds the station or stations at which a given elevation occurs. It prompts for an elevation, calculates the station or stations at which it occurs, then returns for another elevation. STA works only along profile grade. If the grades are of opposite sign, STA finds two stations for a given elevation; if not, only one station is found. If no solution is possible, OFF CURVE is displayed. The program correctly solves for points either within the curve or on either tangent. However, unlike VC, no warning (T) is given when a point is on tangent. STA also lacks the INTERV feature that VC has for automatic running.

INPUT DETAILS

As mentioned, "stations" are input and output in feet. For example, if the PI is at station 28+50.00, answer the STA PI prompt with 2850. Length of vertical curve (VC L') is also in feet. Tangent grades are input in percent.

STATIONS on TANGENT

If a point is off the vertical curve, VC will append a warning, (T), that the point is on the tangent rather than on the curve. Program STA does not have this warning feature.

If the area you're interested in is entirely on a tangent, input a vertical curve length of zero. The programs will only prompt for one grade.

ADDITIONAL PARAMETERS CALCULATED

After input of the basic vertical curve parameters (PI station and elevation, grades, and length) the program calculates M (the middle ordinate), and, if there is one (if the grades are of opposite sign), the high or low point station and elevation. If this information is not wanted and you wish to speed execution, set flag 00 and it will not be displayed.

ELEVATION CHANGE

If the user needs to subtract a distance from each elevation, the program can do it directly. Softkey e brings up the prompt EL CH (elevation change). It adds this number to the P.I. elevation (and BVC and EVC elevations) stored at the beginning of the run. Thus, if VC was being used to calculate bridge deck elevations and the user now wants to calculate beam seat elevations (e.g. for a superstructure height of 5.90 feet), press softkey e, answer the prompt with -5.90 (5.90, chs), and press R/S. This feature works with both VC and STA.

B15

"STA" - find stations for an elevation

Find stations at which elev 440.0 occurs.

$$440.00 = .5417(x)^2 - 4.50(x) + 446.64$$

$$.5417(x)^2 - 4.5(x) + 6.64 = 0$$

Root 1 = 6.3890 ; Root 2 = 1.9187

One station is (70+25) + (1+91.87)

= <u>72+16.87</u>

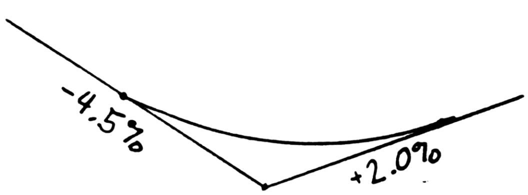
Check other station: (70+25)+(6+38.90)

= 76+63.90 > EVC sta.

∴ Second point is off curve, on tang.

$(440.00 - 433.14)/2\% = 3.430$ stations

(73+25) + (3+43.00) = <u>76+68.00</u>



PI Sta. = 73+25.00
Elev. = 433.14
600' V.C.

```
STA
                    XEQ E
---------
EL
        437.0000    RUN
OFF CURVE
---------
EL
        438.0000    RUN
STA1=7,326.2063
STA2=7,554.5629
---------
EL
        440.0000    RUN
STA1=7,216.8678
STA2=7,668.0000
---------
EL
```

Or xeq "STA"

```
                XEQ "STA"
11:55:29 AM  12/22/88

TITLE:
EXAMPLE             RUN
***********************
STA PI
       7,325.0000   RUN
EL PI
         433.1400   RUN
VC L'
         600.0000   RUN
SL1%
          -4.5000   RUN
SL2%
           2.0000   RUN
M=4.8750'
HI/LO @ 7,440.3846
EL=437.2938
---------
EL

         438.0000   RUN
STA1=7,326.2063
STA2=7,554.5629
---------
EL
         437.0000   RUN
OFF CURVE
---------
EL
         440.0000   RUN
STA1=7,216.8678
STA2=7,668.0000
---------
EL
```

From "VC", softkey E is same as "xeq STA" except uses existing vertical curve data

Program VC, STA; P1,P2 For Vertical Curves (Equal Tang.) 856 bytes Page 1 of 5

Step	X	Y	Z	T	L
01♦LBL "VC*"					
02 SF 11	Lbl#	X	Y	Z	
03 RDN	X	Y	Z	Lbl#	
04 GTO IND T					
05♦LBL "VC"					
06 SF 08					
07 CF 09					
08 CF 11					
09 CLX					
10 "# OFSTS"					
11 PROMPT					
12 X=0?	only PG?				
13 GTO 02					
14 16					
15 +	16+#off				
16 SF 25					
17 STO 15					
18 GTO 00					
19♦LBL "STA"					
20 SF 09					
21 CF 11					
22♦LBL 02					
23 CF 08					
24 9					
25♦LBL 00					
26 XROM "S"					
27 E3	1000				
28 FS? 08					
29 ST/ 15	counter				
30♦LBL 07					
31 CF 05					
32 CF 06					
33 "STA PI"					
34 PROMPT					
35 STO 00	Sta PI				
36 STO 03					
37 STO 05					
38 "EL PI"					
39 PROMPT					
40 STO 09	El PI				
41 "VC L⊦"					
42 PROMPT					
43 X=0?					
44 SF 05	tang.				
45 2					
46 /					
47 ST- 03	Sta BVC				
48 ST+ 05	Sta EVC				
49 E					
50 %					
51 STO 06	ℓ/2				ℓ/2 in Stations
52 "SL1%"					ℓ/200 in feet
53 PROMPT					

Step	X	Y	Z	T	L
54 STO 01	g_1				
55 "SL2%"					
56 FC? 05					
57 PROMPT					
58 STO 02	g_2				
59 RCL 01					
60 *	$g_1 * g_2$				
61 X>0?					
62 SF 06					
63 RCL 02	g_2				
64 LASTX	g_1				
65 -	g_2-g_1				
66 4					
67 /					
68 RCL 06	ℓ/2				
69 X≠0?					
70 /	r/2				
71 STO 04					
72 RCL 09	El. PI				
73 RCL 01	g_1				
74 RCL 06	ℓ/2				
75 *					
76 -					
77 STO 07	El. BVC				
78 FS?C 11					
79 RTN					
80 FC? 05					
81 FS? 00					
82 GTO 00					
83 RCL 02					
84 RCL 01					
85 -	g_2-g_1				
86 RCL 06	ℓ/2				
87 *					
88 4					
89 /					
90 "M="					
91 ARCL X					
92 "⊦'"					
93 AVIEW					
94 FS? 06					
95 GTO 00					
96 RCL 03	Sta BVC				
97 RCL 01	g_1				
98 RCL 04	$g_2-g_1/2L$				
99 50	50	$g_2-g_1/2L$	g_1	Sta BVC	
100 /					
101 /	-(a Sta)				
102 -	Hi/Lo Sta	Sta BVC			
103 "HI/LO @ "					
104 ARCL X					
105 AVIEW					
106 SF 11					
107 X<>Y	Sta BVC	Sta			
108 XEQ 04	El.				
109 "EL="					
110 ARCL X					
111 AVIEW					

$M = \frac{(g_2-g_1)L}{8}$

B17

	X	Y	Z	T	L

```
112♦LBL 00
113 FS? 09
114 GTO E

115♦LBL c
116 CF 18
117 FC? 08
118 GTO C
119 XROM "L"
120 CLX
121 STO 13

122♦LBL 09
123 CF 07
124 CF 10
125 "SKEW: L+,R-"
126 XROM "Pd"
127 TAN
128 STO 11          tan(skew)
129 X=0?
130 SF 18
131 X<0?
132 SF 07           RHF
133 CLX
134 "CRN1'/,"
135 PROMPT
136 X=0?
137 SF 10
138 X=0?
139 GTO d
140 STO 14          crn1
141 12
142 "TO:"
143 PROMPT
144 STO 12          Break
145 .02
146 "CRN2'/,"
147 PROMPT
148 RCL 14
149 -
150 STO 13          Δcrn

151♦LBL d
152 CLX
153 STO 16          offset@CL=0
154 RCL 15          counter
155 ABS
156 FRC
157 17
158 +               17.0__
159 STO 15

160♦LBL 06
161 "OFFSET#"
162 RCL 15          counter
163 16
164 -
165 STO 08          1,2,3,etc
```

```
166 XROM "AX"
167 "⊢'"
168 PROMPT          offset#1,2,etc.
169 STO IND 15
170 FC? 10
171 GTO 06
172 RCL 15          counter
173 6
174 +
175 RCL 08          1,2,3,etc
176 "CRN DROP "
177 XROM "AX"
178 PROMPT          CrnDrop1,2 | 1,2,3,etc. | 23,24,etc.
179 STO IND Z

180♦LBL 06
181 ISG 15          counter
182 GTO 06
183 FS?C 11
184 RTN

185♦LBL C
186 50
187 STO 06          interval
188 XROM "L*"

189♦LBL 21
190 CF 22
191 RCL 06          interv
192 "INTERV"
193 X≠0?
194 PROMPT
195 FC? 22
196 CLX
197 STO 06
198 CLA
199 X≠0?
200 "POB "
201 "⊢STA"
202 PROMPT
203 STO 08          ℄ sta.
204 RDN             interv
205 X≠0?
206 XROM "LL"

207♦LBL 32
208 FC? 08
209 GTO 10
210 RCL 11          tan(skew)
211 ABS
212 FS? 07
213 CHS
214 STO 11          tan(skew)
215  E3             1000
216 RCL 15          counter
217 ABS
218 FRC             .0__
219 *               __.
```

B18

Step	X	Y	Z	T	L
220 LASTX	FRC				
221 +					
222 CHS	--.0				
223 STO 15	counter				
224♦LBL 10					
225 "STA "					
226 ARCL 08	Sta.				
227 "├:"					
228 RCL 06	Interv				
229 X≠0?					
230 FS? 00					
231 FS? 53					
232 AVIEW					
233♦LBL 22					
234 RCL 08	Sta.				
235 FC? 08					
236 GTO 23					
237 RCL 15	counter				
238 ABS					
239 6					
240 +					
241 FS? 10					
242 RCL IND X	CrnDrop				
243 FS? 10					
244 GTO 14					
245 RCL IND 15	Offset				
246 RCL 12	Break				
247 -					
248 X<0?					
249 .	Dist₂				
250 RCL 13	Crown₂				
251 *	Drop₂				
252 RCL IND 15	Offset				
253 RCL 14	Crown₁				
254 *	Drop₁	Drop₂			
255 +	Drop				
256♦LBL 14					
257 STO 10	Drop				
258 RCL 08	Sta				
259 RCL 11	tan(skew)				
260 RCL IND 15	offset				
261 X=0?					
262 STO 10	(Drop)				
263 *	ΔSta	Sta	Drop		
264 +	NewSta	Drop			
265♦LBL 23	Sta				
266 CF 19					
267 RCL 05	StaEVC				
268 X<>Y	Sta	EVC			
269 X>Y?					
270 GTO 03					
271 RCL 03	StaBVC	Sta			
272 X<=Y?					
273 GTO 04					04 ⇒ Curve

Step	X	Y	Z	T	L
274 X<>Y	Sta				
275 RCL 01	g₁	Sta	BVC		
276 GTO 02					
277♦LBL 03	Sta				
278 RCL 02	g₂	Sta	EVC		
279♦LBL 02					
280 RCL 09	El.PI				
281 RCL 00	Sta.PI	El.PI	g%	Sta	
282 R↑	Sta	StaPI			
283 -	ΔSta	El.PI	g%	g%	
284 R↑	g	ΔSta			
285 *	ΔEl.				
286 E2	100				
287 /	ΔEl.	ElPI			
288 -	Elev.				
289 SF 19					SF19 ⇒ "on tangent"
290 GTO 11					
291♦LBL 04	StaBVC	Sta			
292 -	ΔSta				
293 E	1				
294 %	Dist, ft.				
295 ENTER↑					
296 ENTER↑					
297 RCL 04	r/2	"X"	"X"		
298 *	rX/2				
299 RCL 01	g₁	rX/2	X		
300 +					
301 *	$rX^2/2+g_1X$				
302 RCL 07	El. BVC				
303 +	Elev				
304♦LBL 11					
305 FS?C 11					
306 RTN					
307 CLA					
308 FC? 08					
309 GTO 11					
310 RCL 10	CrownDrop	Elev			
311 -	Elev.				
312 " R"					
313 RCL 15	counter				
314 X<0?					
315 " L"					
316 ABS					
317 INT	counter				
318 16					
319 -	1,2,3, etc	elev.			
320 X≠0?					
321 XROM "AX"					
322 X=0?					
323 "CL"					
324 X<>Y	elev.				

B19

Program VC, STA For ____________ ________ bytes

Line	X	Y	Z	T	L
325♦LBL 11					
326 "⊢ EL="	elev.	1,2,3,etc.			
327 ARCL X					
328 FC? 05					FC? 05 and FS? 19
329 FC? 19					
330 FS? 53					
331 "⊢(T)"					
332 XEQ "PY"					
333 FC? 08					
334 GTO 11					
335 X<>Y	1,2,3,etc				
336 X≠0?					
337 GTO 00					
338 FS? 18					
339 GTO 11					
340 SIGN	1				
341 CHS	-1				
342 ST* 11	tan(sta)				
343 ST* 15	counter				
344♦LBL 00					
345 ISG 15					
346 GTO 22					
347♦LBL 11					
348 XROM "L"					
349 RCL 06	Interv				
350 X=0?					
351 GTO 21					
352 RCL 08	Sta				
353 +	New Sta				
354 STO 08					
355 GTO 32					
356♦LBL e					
357 "EL CH"					
358 PROMPT	Δ el.				
359 ST+ 07	el. BVC				
360 ST+ 09	el. PI				
361 FC? 09					
362 GTO C					
363 ST+ 06	el. EVC				
364♦LBL E					
365 RCL 02	g_2 %				
366 E	1				
367 %	g_2				
368 RCL 05	Sta EVC				
369 RCL 00	Sta PI				
370 -	L/2	g_2			
371 *	Δ El				
372 RCL 09	el. PI				
373 +	el. EVC				
374 STO 06					

Line	X	Y	Z	T	L
375♦LBL 15					
376 XROM "L"					
377 CF 07					
378 "EL"					
379 PROMPT					
380 STO 08	Elev.				
381 "STA1="					
382 FC? 05					
383 GTO 08					
384 RCL 01	g_1				
385♦LBL 13					
386 E					
387 %	g				
388 RCL 08	Input El.				
389 RCL 09	El. PI				
390 -	Δ El				
391 X<>Y	g	Δ El			
392 /	Δ Dist				
393 RCL 00	Sta PI	Δ Sta			
394♦LBL 16					
395 +	Sta				
396 ARCL X					
397 AVIEW					
398 GTO 15					
399♦LBL 08					
400 FC? 06					
401 GTO 08					
402 CF 10					
403 RCL 06	El. EVC				
404 RCL 07	El. BVC				
405 X>Y?					
406 SF 10					
407 X>Y?					
408 X<>Y	lesser	greater			
409 RCL 08	Input El.	lesser	greater		
410 X>Y?					
411 GTO 05					
412 RCL 01					
413 FS? 10					
414 RCL 02	g	Input El			
415 GTO 13					
416♦LBL 05	Input El	lesser	greater		
417 RCL Z	greater				
418 X>Y?					
419 GTO 08					
420 RCL 02					
421 FS? 10					
422 RCL 01	g				
423 GTO 13					

B20

Step	X	Y	Z	T	L
424♦LBL 08					
425 RCL 07	El.BVC				
426 RCL 08	Input El				
427 -	ΔEl				
428 RCL 01	g1%				
429 RCL 04	r/2	g1	ΔEl		
430 SF 25					
431 XROM "Q"					
432 FC?C 25					
433 GTO 01					
434 E2	100				
435 *	Dist1				
436 X<>Y					
437 LASTX					
438 *	Dist2	Dist1			
439 FC? 06					
440 GTO 08					
441 X<=0?					
442 X<>Y	Dist				
443 RCL 03	Sta.BVC				
444 GTO 16					
445♦LBL 08					
446 X>Y?					
447 X<>Y	Min	Max			
448 X>0?					
449 GTO 05					
450 CLX					
451 RCL 08	Input El	Max			
452 RCL 07	El.BVC				
453 -	ΔEl.				
454 RCL 01	g1				
455 /					
456 E2	100				
457 *	Dist				
458♦LBL 05					
459 RCL 03	Sta BVC				
460 +	Sta				
461 ARCL X					
462 AVIEW					
463 X<>Y	Dist2				
464 RCL 05	Sta EVC				
465 LASTX	Sta BVC				
466 -	VC L				
467 X<=Y?					
468 GTO 08					
469 CLX					
470 LASTX	Sta BVC	Dist			
471 GTO 05					

FS?06 ⇒ g_1 and g_2 are same sign- only one station

Step	X	Y	Z	T	L
472♦LBL 08	L	Dist2			
473 RCL 08	Input El				
474 RCL 09	El.PI				
475 -	ΔEl				
476 RCL 02	g2				
477 /					
478 E2					
479 *	Dist				
480 RCL 00	Sta PI				
481♦LBL 05					
482 +	Sta				
483 "STA2="					
484 ARCL X					
485 AVIEW					
486 GTO 15					
487♦LBL 01					
488 SF 07					
489 "OFF"					
490 BEEP					
491 AVIEW					
492 GTO 15					
493 END					

B22

STOPPING SIGHT DISTANCE

xeq SSD (SIZE 004)

SOFTKEYS (USER mode ON)

A	Input vertical curve length, solve for SSD
B	Input design speed (mph), solve for VC L
D	Input stopping sight distance, solve for VC L

USER FLAGS

Flag 00	clear	Full output
	set	No title block printed

INPUT SUMMARY

Prompt	Input	Default
G1%	Foreslope in percent (up to the right is positive)	No default
G2%	Backslope in percent	No default
L MPH SSD (A) (B) (D)	A "menu" to the softkeys, as shown	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

SSD	The available stopping sight distance in feet
VC L	The required vertical curve length in feet

PROGRAM FLAGS

Flag 05	clear	Sag crve
	set	Crown curve

STORAGE REGISTER USE

Register	Value
00	A (G2% - G1%)
01	(400 + 3.5S)/A
02	400
03	3.5

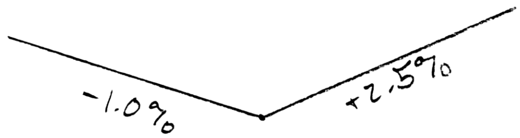
STOPPING SIGHT DISTANCE

Program SSD does stopping sight distance calculations according to the standard AASHTO road design equations and tables (with two exceptions for design speed input, given below). Given the grades, it computes the available stopping distance from the vertical curve length; or the allowable vertical curve length, from either the design speed or stopping sight distance. The program will not backsolve to find the design speed.

The grades are input in response to prompts. Then either the SSD, curve length, or design speed is input using softkeys, as indicated by a crude "menu" in the display. Once the grades are input, they need not be re-input; the program returns to the softkey prompt.

As mentioned, the program will solve a vertical curve length given a design speed. It does this by equating the design speed to a stopping sight distance with an equation derived from the tables in the AASHTO Road Design Manual. This equation gives the correct distance, and thus curve length, fro speeds at five mile-per-hour increments, except for 20 and 65 mph (due to a "bump" in the tabular values. Thus, if you have a design speed of either 20 or 65 mph, either input the tabulated SSD values (shown below) or live with a minor error. To compare tabular (AASHTO) values with the program's calculated values:

Design Speed	AASHTO S.S.D.	Program S.S.D.
20	125	100
25	150	150
30	200	200
35	250	250
40	325	325
45	400	400
50	475	475
55	550	550
60	650	650
65	725	750
70	850	850



Desire a Stopping Sight Distance of 850' (70 mph.)

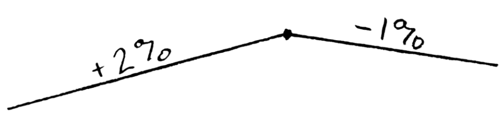
$$L = 2S - \frac{400 + 3.5S}{A}$$

$$= 2 \times 850 - \frac{400 + 3.5(850)}{3.5}$$

$$= 735.7'$$

L < S : OK

```
XEQ "SSD"
1:16:00 PM  03/15/89

TITLE:
STOPPING SIGHT DISTANCE
                    RUN
************************
G1%
        -1.0000    RUN
G2%
         2.5000    RUN
L MPH  SSD
       850.0000  XEQ D
VC L=735.7143'
---------
L MPH  SSD
        70.0000  XEQ B
VC L=735.7143'
---------
L MPH  SSD
       686.0000  XEQ A
SSD=800.2857'
---------
L MPH  SSD
```



Design a vertical curve for 55 mph.

$$L = 2S - \frac{1329}{A}$$

55 mph → S = 550' from table

$$L = 2 \cdot 550 - \frac{1329}{3} = 657.0'$$

L > S ⇒ use other equation

$$L = \frac{S^2 \cdot A}{1329} = \frac{550^2 \times 3}{1329} = 682.8'$$

```
XEQ "SSD"
1:17:13 PM  03/15/89

TITLE:
CROWN               RUN
************************
G1%
         2.0000    RUN
G2%
        -1.0000    RUN
L MPH  SSD
        55.0000  XEQ B
VC L=682.8442'
---------
L MPH  SSD
```

(B25)

Program SSD For Stopping Sight Distance ______ bytes Page 1 of 2

Step	X	Y	Z	T	L	Notes
01♦LBL "SSD"						
02 SF 27						
03 3						
04 XROM "S"						
05 "G1%"						
06 PROMPT						
07 "G2%"						
08 PROMPT	G_2	G_1				
09 CF 05						
10 X<=Y?						
11 SF 05						
12 -	$G_1 - G_2$					
13 ABS						
14 1329						
15 FC? 05						A if sag
16 SIGN						$\frac{A}{1329}$ if crown
17 /						
18 STO 00	A or $\frac{A}{1329}$					
19♦LBL 06						
20 "L MPH SSD"						
21 PROMPT						
22♦LBL B	mph					
23 1.7						
24 Y↑X	$(mph)^{1.7}$					
25 40						
26 /						
27 .2						
28 +						
29 INT						
30 25						
31 *	SSD					
32♦LBL D	SSD					
33 "VC L="						
34 ENTER↑	D	D				
35 ENTER↑	D	D	D			
36 ST+ Y	D	2D	D			
37 FS? 05						
38 GTO 07						
39 3.5	3.5	D	2D	D		
40 STO 03						
41 *	3.5D	2D	D	D		
42 4 E2	400					
43 STO 02						
44 +	3.5D+400					
45 RCL 00	A					
46 /						
47 STO 01	$\frac{3.5D+400}{A}$	2D	D	D		SAG
48 -	VC L?	D				$L = 2D - \frac{3.5D+400}{A}$
49 X<=Y?						
50 GTO 08						if $L < D$

Step	X	Y	Z	T	L	Notes
51 RDN	D					
52 X↑2	D^2					
53 RCL 01	$\frac{3.5D+400}{A}$					
54 /	VC L					$L = \frac{D^2 A}{3.5D+400}$
55 GTO 08						if $L > D$
56♦LBL 07						
57 RDN	2D	D				
58 RCL 00	A/1329					Crown
59 1/X						
60 -	$2D - \frac{1329}{A}$	D				$L = 2D - \frac{1329}{A}$
61 X<=Y?						if $L < D$
62 GTO 08						
63 X<>Y	D					
64 X↑2	D^2					
65 RCL 00	A/1329					
66 *	VC L					$L = \frac{D^2 A}{1329}$
67 GTO 08						if $L > D$
68♦LBL A	VC L					
69 ENTER↑	L	L				
70 "SSD="						
71 RCL 00	A	L	L			
72 FS? 05						
73 GTO 09						
74 *	AL	L				
75 RCL 02	400	AL	L			
76 +	AL+400	L				
77 RCL 00	A		L			
78 ST+ X	2A		L			
79 RCL 03	3.5			L		Sag
80 -	2A-3.5	AL+400	L			$D = \frac{AL+400}{2A-3.5}$
81 /	SSD?	L	L	L		
82 X>Y?						
83 GTO 08						
84 RDN	L					
85 RCL 00	A					
86 RCL 02	400	A	L	L		
87 R↑	L	400	A	L		
88 *	400L	A	L	L		
89 RCL 03	3.5	400L	A	L		
90 R↑	L	3.5	400L	A		
91 *	3.5L	400L	A	A		
92 R↑	A					
93 CHS	-A	3.5L	400L			
94 XROM "Q"	SSD					
95 GTO 08						

	X	Y	Z	T	L		X	Y	Z	T	L
Crown $SSD = \sqrt{\frac{1329 L}{A}}$ if D < L						96♦LBL 09	$\frac{A}{1329}$	L	L		
						97 /	$\frac{1329 L}{A}$				
						98 SQRT	$\sqrt{\ }$	L			
						99 X<=Y?					
						100 GTO 08					
						101 X<>Y	L				
						102 RCL 00	A/1329				
						103 1/X	1329/A				
						104 +	$L + \frac{1329}{A}$				
$SSD = \frac{L}{2} + \frac{1329}{2A}$ if D > L						105 2					
						106 /	$\frac{L}{2} + \frac{1329}{2A}$				
						107♦LBL 08					
						108 X<0?					
						109 CLX					
						110 ARCL X					
						111 XEQ "FT"					
						112 AVIEW					
						113 XROM "L"					
						114 GTO 06					
						115♦LBL "RW"					
						⋮					

HORIZONTAL CURVE SOLUTIONS

xeq HC (SIZE 002)

NOTE: The USER FLAGS, OUTPUT SUMMARY, PROGRAM FLAGS, and STORAGE REGISTER USE portions of this documentation, as well as much of the full write-up which follows, pertains to program CHC as well. CHC is also documented in the Coordinate Geometry section.

SOFTKEYS (USER mode ON)

A	Arc Length
B *	Radius
C	Chord length
d *	Degree of curve
D *	Delta angle
E	External
F (x<>y)	Exchange menus (no input)
H (mid)	Middle ordinate
J (tan)	Tangent

At least one item must be an asterisked item, and <u>this item must be input first.</u>

If the degree of curve or delta angle are input, it is up to the user to verify that the input agrees with the status of flag 03, i.e. if the angles are in degrees-minutes-seconds, flag 03 must be set. This is <u>not</u> prompted as in most programs.

USER FLAGS

Flag 00	clear	Full output
	set	Sector, segment, and fillet areas not output
Flag 03	clear	Angular input and output in decimal degrees (dd)
	set	Angles in degrees-minutes-seconds (DMS)

INPUT SUMMARY

Program HC does not use prompts for input as do most of the other programs in this package. The two prompts which are used are intended as crude menus to the softkeys, for which see above.

Note that <u>all the softkeys are active even when their menu is not displayed.</u> For example, to input radius and delta, it is not necessary to bring back the first menu. Simply key in the radius, press softkey B, key in the delta angle, and press softkey D.

OUTPUT SUMMARY	(Items in parenthesis are displayed only if flag 00 is clear)
RAD	Radius
DEG	Degree of curve
DEL	Delta, the central angle or angle subtended
ARC L	Arc length
T	Tangent length
CH	Chord length
M	Middle ordinate
E	External dimension
(SECT)	Area of the sector
(SEGM)	Area of the segment
(FIL)	Area of the fillet

PROGRAM FLAGS

Flag 06	clear set	In second menu In first menu
Flag 07	clear set	Don't have radius Have radius--solve after Delta input
Flag 09	clear set	HC CHC
Flag 10	clear set	Don't have Delta Have Delta--solve after radius input
Flag 21	set	Controls display and print behavior
Flag 27	set	Sets USER mode

STORAGE REGISTER USE

Register	Value
00	Radius
01	Delta/2 (decimal degrees stored)

HORIZONTAL CURVE SOLUTIONS

Programs HC and CHC are used for the solution of horizontal (circular) curves. HC computes all the commonly needed parameters, given any (well, almost any) two of them as input. CHC computes these same parameters from the coordinates of the PC, PT and RP (point of curvature or beginning, point of tangency or end, and radius or center point). CHC also computes the azimuths of the incoming and outgoing tangents (PC to PI and PI to PT, respectively, the PI being the point of intersection).

For input, program HC uses the softkeys A, B, C, D, d (shifted), E, F, H, and J in USER mode. USER mode is automatically turned on by the program to facilitate input. Parameters are input by keying the number in and pushing the appropriate key, as follows:

	Key	Parameter
First Menu	A	Arc Length
	B *	Radius
	C	Chord length
	d *	Degree of curve
	D *	Delta angle
	E	External
Second Menu	F (x<>y)	Exchange menus (no input)
	H (mid)	Middle ordinate
	J (tan)	Tangent

The program requires two of these parameters. At least one item must be an asterisked item, and <u>this item must be input first.</u> Radius and degree of curve are not sufficient since they are derivable one from the other; if both are input, HC will ignore the first one and await further input.

MENUS

HC uses menus to prompt input, displaying the above mnemonics (R, DL, etc.) roughly above each key. The first-row parameters are prompted first, then the second-row menu appears. To return to the first-row menu, press softkey F (x<>y). Pressing it again brings back the second-row menu. These menus are only mnemonic aids; the first-row keys are still active when the second-row menu is on, and vice versa.

B31

ANGLES

As with the other geometry programs in this package, HC and CHC format output and expect input for angles to be in decimal degrees (dd) if flag 03 is clear, and in degrees-minutes-seconds (DMS) if flag 03 is set. This is not reflected in the prompts; the user must consider the status of flag 03 when inputting the degree of curve or the delta angle. The output is labeled.

COORDINATE GEOMETRY

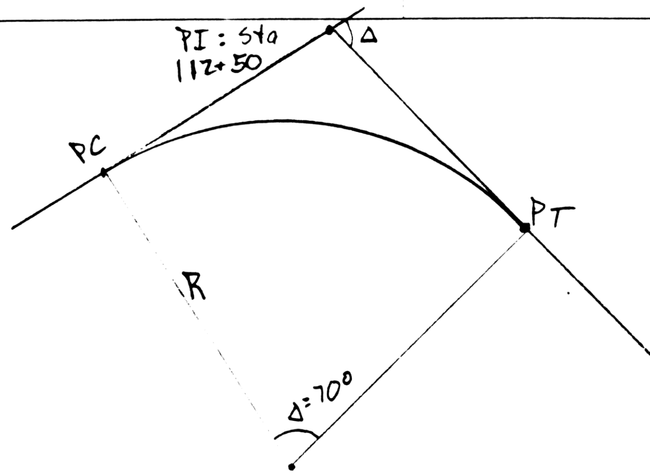
Input for CHC is much simpler (assuming your point coordinates are already stored by other coordinate geometry routines). The program prompts for the point numbers of the PC, RP, and PT. As with most other coordinate geometry routines the numbers are keyed into the stack: key in PC point number, enter, key in radius point number, enter, key in PT point number (don't enter), press R/S. CHC checks the radii PC-RP and PT-RP. The two are first rounded to the accuracy of the display setting and then tested for equality. If they aren't equal, a warning is given; the radius used is from PC to RP. (Thus, don't set the display to FIX 4 if the coordinates are only good to two decimal places.) It also calculates the north azimuths of the incoming and outgoing tangent, again in either D.MS or decimal degrees depending on flag 03. It then drops into the HC output routine, displaying all the parameters listed on the first page and the area of the sector, segment, and fillet. All the linear output is labeled with dimensions of feet and square feet. The programs will also, of course, work with inches, meters, or furlongs.

Both programs set the calculator to DEG (degree) mode upon execution (in subroutine S). If for some reason this is changed by the user during use, errors will result.

If USER mode is set when CHC is run, the softkeys are active, and once pressed (accepting whatever is in the X register as input), the menus will come up as though HC had been executed. However, when all the output has been displayed, the program returns to the CHC prompt. When using HC, all the output need not be viewed; the softkeys are active at any time to accept new input and start over (or to bring up a menu, in the case of softkey F).

FLAGS

If flag 00 is set, the areas of sector, segment, and fillet are not displayed by either HC or CHC; in CHC the distances PC-RP and PT-RP, and the tangent azimuths, are not diplayed either.



SF 03
XEQ "HC"
5:34:50 PM 01/05/89

TITLE:
CURVE EXAMPLE RUN
****
L R CH d/D E
7.3000 XEQ d
<> MID T
XEQ F
L R CH d/D E
70.0000 XEQ D

RAD=763.9437'
DEG=7.3000 DMS
DEL=70.0000 DMS
ARC L=933.3333'
T=534.9192'
CH=876.3602'
M=138.1577'
E=168.6594'

SECT=356,507.0725 SQ'
SEGM=82,300.0591 SQ'
FIL=52,141.0612 SQ'

L R CH d/D E

For a 7°30' Curve, solve for missing parameters and stations

$D = 7°30' \; ; \; \Delta = 70°$

$$R = \frac{5729.5780}{D} = 763.944'$$

$$T = R \tan(\Delta/2) = 534.919'$$

$$L = \frac{\Delta}{D} \times 100 = \frac{70}{7.5} \times 100 = 933.33'$$

$$\text{Chord} = 2 \cdot R \sin(\Delta/2) = 876.360'$$

$$M = R - R\cos(\Delta/2) = 138.158'$$

$$E = \frac{R}{\cos(\Delta/2)} - R = 168.659'$$

$$\text{Sector} = \frac{\Delta}{360} \cdot \pi R^2 = 356{,}507.07 \,\square'$$

$$\text{Segment} = \text{Sector} - \tfrac{1}{2}\,\text{Chord} \times R\cos\frac{\Delta}{2}$$

$$= 82{,}300.07 \,\square'$$

$$\text{Fillet} = \tfrac{1}{2}\,\text{Chord} \times (M+E) - \text{Segment}$$

$$= 52{,}141.03 \,\square'$$

Sta. PC = 112+50 − 534.919

= 107+15.081

Sta PT = Sta PC + L = 116+48.414

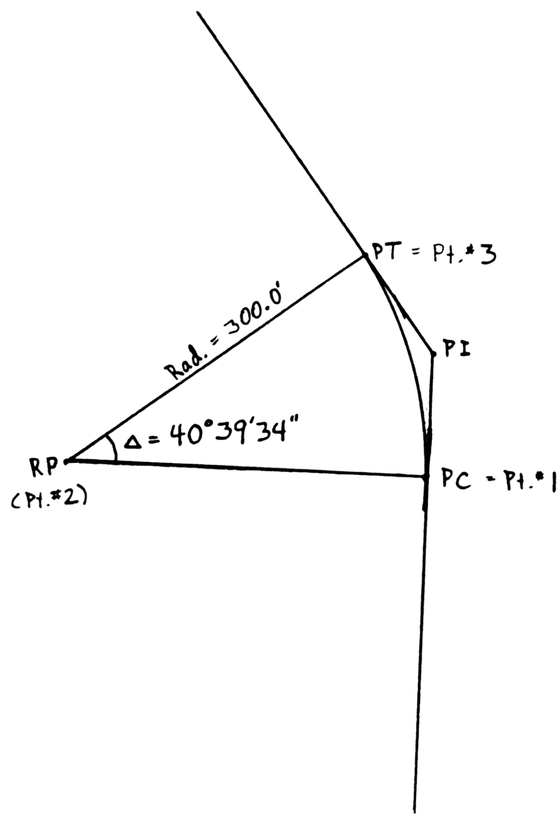
SHORT FORM - SF 00

SF 00
XEQ "HC"
L R CH d/D E
7.3000 XEQ d
<> MID T
70.0000 XEQ D

RAD=763.9437'
DEG=7.3000 DMS
DEL=70.0000 DMS
ARC L=933.3333'
T=534.9192'
CH=876.3602'
M=138.1577'
E=168.6594'

L R CH d/D E

CHC Example




```
                XEQ "STOR"
--------
PT↑N↑E
            1.0000 ENTER↑
        3,103.6749 ENTER↑
        9,827.2778    RUN
--------
PT↑N↑E
            2.0000 ENTER↑
        3,106.1477 ENTER↑
        9,527.2890    RUN
--------
PT↑N↑E
            3.0000 ENTER↑
        3,299.7338 ENTER↑
        9,756.4711    RUN
--------
PT↑N↑E
```

```
                 XEQ "CHC"
 9:53:34 AM  10/15/87

TITLE:
STERLING ST.          RUN
************************
PC↑RP↑PT
           1.0000 ENTER↑
           2.0000 ENTER↑
           3.0000    RUN
PC-RP=300.0000'
PT-RP=300.0000'
PC-PI AZ=0.2820 DMS
PI-PT AZ=-40.1114 DMS

RAD=300.0000'
DEG=19.0555 DMS
DEL=40.3934 DMS
ARC L=212.8925'
T=111.1504'
CH=208.4534'
M=18.6874'
E=19.9288'
---------
SECT=31,933.8705 SQ'
SEGM=2,613.5850 SQ'
FIL=1,411.2467 SQ'
-----------------------
PC↑RP↑PT
```

Flag 03 is set ⇒ output in D.MS

Short version with Flag 00 set gives only these parameters

```
                  SF 00
              XEQ "CHC"
PC↑RP↑PT
           1.0000 ENTER↑
           2.0000 ENTER↑
           3.0000    RUN

RAD=300.0000'
DEG=19.0555 DMS
DEL=40.3934 DMS
ARC L=212.8924'
T=111.1503'
CH=208.4533'
M=18.6873'
E=19.9287'
---------
PC↑RP↑PT
```

B34

Program HC.CHC For Horizontal Curves 515 bytes Page 1 of 3

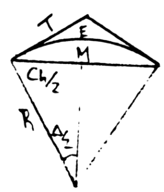
Step	X	Y	Z	T	L
01♦LBL "CHC"					
02 E					
03 XROM "S"					
04 SF 09					
05♦LBL 08					
06 FC? 09					
07 GTO 14					
08 SF 07					
09 "PC↑RP↑PT"					
10 PROMPT	PT	RP	PC		
11 STO 01	PT				
12 RDN					
13 STO 00	RP	PC		PT	
14 XEQ 09	Az_{pc}	d_{pc}			
15 X<> 01	PT				
16 X<>Y	d_{pc}	PT			
17 X<> 00	RP	PT			
18 XEQ 09	Az_{pT}	d_{pT}			
19 X<>Y	d_{pT}				
20 RCL 00	d_{pc}	d_{pT}	Az_{pT}		
21 "PC"					
22 XEQ 11					
23 X<>Y	d_{pT}	d_{pc}	Az_{pT}		
24 "PT"					
25 XEQ 11					
26 "PC≠PT"					
27 X≠Y?					
28 BEEP					
29 X≠Y?					
30 AVIEW					
31 X≠Y?					
32 XROM "L"					
33 R↑					
34 R↑	Az_{pT}				
35 RCL 01	Az_{pc}				
36 X<=Y?					
37 GTO 00					
38 180					
39 -					
40 X<>Y					
41 LASTX					
42 -					
43 X<>Y	Az'_{pc}	Az'_{pT}			
44♦LBL 00					
45 "PC-PI"					
46 XEQ 10					
47 X<>Y	Az_{pT}	Az_{pc}			
48 "PI-PT"					
49 XEQ 10					
50 -	Δ				
51 2					
52 /	Δ/2				
53 ABS					
54 STO 01	Δ/2				
55 GTO 00					

Step	X	Y	Z	T	L
56♦LBL 10					
57 FS? 00					
58 RTN					
59 "⊦ AZ"					
60 XROM "Dd"					
61 AVIEW					
62 RTN					
63♦LBL 09	RP	Pt			
64 XROM "XY"	ΔX	ΔY			
65 R-P	d	-Az			
66 X<>Y	-Az				
67 CHS	Az				
68 X>0?					
69 RTN					
70 360					
71 +					
72 RTN	Az	d			
73♦LBL "HC"					
74 CF 09					
75 SF 27	"User"				
76 E					
77 XROM "S"					
78♦LBL 14					
79 CF 07					
80 CF 10					
81♦LBL 07					
82 SF 06					
83 "L R CH d/D E"					
84 PROMPT					
85♦LBL d	degree				
86 FS? 03					
87 HR					
88 E	1				
89 %	deg/100				
90 1/X	100/deg				
91 R-D	rad.				
92♦LBL B					
93 STO 00	Rad.				FS 07 ⇒ have radius
94 SF 07					
95 FS? 10					
96 GTO 00					
97 GTO 06					
98♦LBL D	Δ				
99 FS? 03					
100 HR	Δ				
101 2					
102 /	Δ/2				
103 STO 01					FS 10 ⇒ have delta
104 SF 10					
105 FS? 07					
106 GTO 00					

B35

Program HC, CHC For Horizontal Curves ______ bytes Page 2 of 3

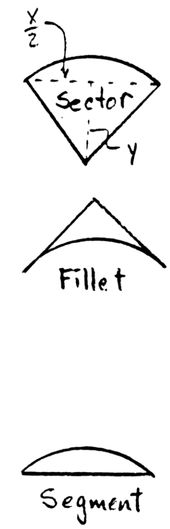
Step	X	Y	Z	T	L
107◆LBL F					
108 FC? 06					
109 GTO 07					
110◆LBL 06					
111 CF 06					
112 "<> MID T"					
113 PROMPT					
114◆LBL A	Arc L				
115 R-D	L·180/π				
116 FS? 07					
117 GTO 01					
118 RCL 01	Δ/2				
119 ST+ X	Δ				
120 GTO 03					
121◆LBL 01					
122 RCL 00	Rad				
123 ST+ X	2R				
124 /	Δ/2				
125 STO 01					
126 GTO 00					
127◆LBL C	Chord				
128 2					
129 /	Ch/2				
130 FS? 07					
131 GTO 02					
132 RCL 01	Δ/2				
133 SIN	sin Δ/2	Ch/2			
134 GTO 03					
135◆LBL 02	Ch/2				
136 RCL 00	Rad				
137 /					
138 ASIN	Δ/2				
139 STO 01					
140 GTO 00					
141◆LBL E	Ext.				
142 FS? 07					
143 GTO 05					
144 RCL 01	Δ/2				
145 2					
146 /	Δ/4				
147 TAN	tan Δ/4	E			
148 /	T				
149◆LBL J	Tang				
150 FS? 07					
151 GTO 01					
152 RCL 01	Δ/2				
153 TAN	tan Δ/2	T			
154 GTO 03					



Step	X	Y	Z	T	L
155◆LBL 01	T				
156 RCL 00	Rad				
157 /	T/R				
158 ATAN	Δ/2				
159 STO 01					
160 GTO 00					
161◆LBL H	M				
162 FS? 07					
163 GTO 04					
164 E	1				
165 RCL 01	Δ/2				
166 COS	cos Δ/2	1	M		
167 -	1-cos Δ/2	M			
168◆LBL 03					
169 /	Rad.				
170 STO 00					
171 GTO 00					
172◆LBL 04	M				
173 CHS	-M				
174◆LBL 05	E or -M				
175 RCL 00	Rad				
176 +	R+E/R-M				
177 LASTX	R				
178 X<Y?					
179 X<>Y	hypotenuse	adjacent			
180 /					
181 ACOS	Δ/2				
182 STO 01					
183◆LBL 00					
184 CF 06	"Menu"				
185 SF 21	"Print"				
186 E2	100				
187 FC? 10					
188 FS? 07					
189 X=0?					
190 GTO F					
191 RCL 00	Rad	100			
192 ADV					
193 CF 07					
194 CF 10					
195 "RAD"					
196 XEQ 12					
197 /	100/R				
198 R-D	Degree				
199 "DEG"					
200 XROM "Dd"					
201 AVIEW					

Step	X	Y	Z	T	L
202 RCL 01	Δ/2				
203 ST+ X	Δ				
204 "DEL"					
205 XROM "Dd"					
206 AVIEW					
207 RCL 00	Rad	Δ			
208 *					
209 D-R	L				
210 "ARC L"					
211 XEQ 12					
212 RCL 01	Δ/2				
213 TAN	tan Δ/2				
214 RCL 00	R				
215 *	T				
216 "T"					
217 XEQ 12					
218 RCL 01	Δ/2				
219 SIN	sin Δ/2				
220 RCL 00	Rad	sin Δ/2	T		
221 *	ch/2				
222 ST+ X	ch				
223 "CH"					
224 XEQ 12					
225 E	1				
226 RCL 01	Δ/2				
227 COS	cos Δ/2	1	ch	T	
228 -	1-cos				
229 RCL 00	Rad				
230 *	M	ch	T	T	
231 "M"					
232 XEQ 12					
233 R↑	T				
234 RCL 01	Δ/2				
235 2					
236 /	Δ/4				
237 TAN					
238 *					
239 "E"	E				
240 XEQ 12					
241 XROM "L"					
242 FS? 00					
243 GTO 08					

Step	X	Y	Z	T	L
244 RCL 00	Rad				
245 X↑2	R^2				
246 RCL 01	Δ/2				
247 *					
248 D-R	$\frac{2\pi}{360}\cdot\frac{\Delta}{2}R^2$				Area = $\frac{\Delta}{360} \times \pi R^2$
249 "SECT="					
250 XEQ 13					
251 ENTER↑					
252 ENTER↑	S	S	S		
253 RCL 01					
254 RCL 00	Rad	Δ/2	S	S	
255 P-R	X/2	Y	S	S	
256 *	Area▽				
257 -					
258 "SEGM="	Segm	S	S	S	
259 XEQ 13					
260 RCL 01	Δ/2				
261 TAN	Tan Δ/2				
262 RCL 00					
263 X↑2	R^2	Tan Δ/2	Segm	S	
264 *	◊				
265 R↑	S				
266 -					
267 "FIL="	Fillet				
268 XEQ 13					
269 XROM "LL"					
270 GTO 08					




```
271♦LBL 11
272 RND
273 FS? 00
274 RTN
275 "⊢-RP"

276♦LBL 12
277 "⊢="
278 ARCL X
279 XROM "FT"
280 AVIEW
281 RTN

282♦LBL 13
283 ARCL X
284 XROM "FS"
285 AVIEW
286 END
```

B38

COORDINATE GEOMETRY PROGRAMS

SIZE (2N+2) for "N" points

xeq STOR to initialize points and store point coordinates
- DUMP to list ("dump") points
- FREE to find free (CLR'd) points
- DUP to duplicate a block of points
- INV to find distance and azimuth between two points
- LAZ to set (locate) a point by distance and azimuth
- LBR to set a point by distance and bearing
- LAN to set a point by backsighting on another point
- LPT to set a point by distance, on the bearing through two other points
- LLI to set a point along a line determined by two other points
- ANGL to find the angle between two points about a third
- PART to set a point part way between two other points
- OFPT to find the offset of a point from a line defined by two other points
- OFAZ to find the offset from a line defined by a point and an azimuth
- OFBR to find the offset from a line defined by a point and a bearing
- IAZ to intersect two lines defined by points and azimuths
- IBR to intersect two lines defined by points and bearings
- IPT to intersect two lines each defined by two points
- PRL to set two points defining a line parallel to a line defined by two other points
- ATRV to set a point along an arc by arc length
- ATVA to set a point along an arc by angle subtended
- AFRC to set a point partway between two points along an arc
- ARCS to intersect two arcs
- APT to intersect an arc with a line defined by two points
- AAZ to intersect an arc with a line defined by a point and an azimuth
- ABR to intersect an arc with a line defined by a point and a bearing
- INTAN to find the internal tangency points of two circles
- EXTAN to find the external tangency points of two circles
- PTAN to find the points of tangency of two lines through a point, to a circle
- C3P to find a circle through three points
- ADJ to adjust a traverse by compass rule
- AREA to find the area bounded by points
- CHC to calculate horizontal (circular) curve data between two points about a third

SOFTKEYS (USER mode ON)

C Correct a point input error in AREA

None of the other COGO routines makes use of softkeys. However, the softkeys of HC are accessible during use of CHC and should be avoided.

USER FLAGS

Flag 00	clear	Point protect on
	set	Point protect off
Flag 01	clear	North-East mode
	set	X-Y mode
Flag 02	clear	Warning BEEP disabled in point-protect
	set	Warning BEEP enabled
Flag 03	clear	Angles in decimal degrees (dd)
	set	Angles in degrees-minutes-seconds (DMS)

INPUT SUMMARY

A Angle or north azimuth

A1, A2 Azimuth

B Bearing, measured from North or South; between 0° and 90° in one of the four quadrants.

Quad Quadrant of a bearing: 1 = NE
2 = SE
3 = SW
4 = NW

C Center point (radius point)

D Distance

I First point in a series

J Point of beginning: "From point J, ..."

K Backsight or reference point, or last point in a series

L, M Points defining a line

M "Proximity" point; choose the point closest to point M

N, N1, N2 Point number to be solved or found; point number under which to store the solved coordinates

R, R1, R2 Radius or radii of arcs

OUTPUT SUMMARY

PT Point number stored and displayed

N Northing (flag 01 clear)

E Easting (flag 01 clear)

X	X coordinate (flag 01 set)
Y	Y coordinate (flag 01 set)
D	Distance
AZ	North azimuth
A	Angle
OFFSET	Offset from a line

PROGRAM FLAGS

Flag 08	clear	Don't calculate offset
	set	Calculate offset
	clear	PART
	set	LLI
Flag 09	clear	Quadrant 2 or 3 (in BA)
	set	Quadrant 1 or 4
	clear	INTAN or ATVA
	set	EXTAN or ATRV
	clear	More than two points input in AREA
	set	Only one or two points input in AREA
Flag 10	clear	Softkey C has not been used in AREA
	set	Softkey C has been used in AREA
Flag 14	set	Line type control
Flag 21	set	Display control
Flag 26	clear	BEEP disabled according to flag 02
	set	BEEP enabled according to flag 02
Flag 29	clear	Display of decimal point disabled

STORAGE REGISTER USE

Register	Value
00	Working register
01	Working register
02	Easting (X) coordinate of point #1
03	Northing (Y) coordinate of point #1
04	Easting (X) coordinate of point #2
05	Northing (Y) coordinate of point #2
	Etc.

B42

COORDINATE GEOMETRY PROGRAMS

There are thirty-three user programs in the coordinate geometry section of this module (CHC, LPT, and most of those between STOR and AREA, inclusive, in a CAT 2 listing). Each of the routines has its own short description, with a sketch, and input and output information. These individual writeups follow this introduction, which gives an overview of the coordinate geometry group.

All the routines work with points, lines, and curves in two-dimensional (plane) space. The routines will work in an X-Y system if flag 01 is set, and in a North-East system if flag 01 is clear. This may be changed at any time as long as a prompt isn't pending. No matter which orientation is being used, the X (East) coordinate of a point "N" is stored in register (2N); the Y (North) coordinate is stored in register (2N+1). Thus , to store a given number of points "n" requires a SIZE of 2n+2. The available SIZE is checked when initializing points in STOR.

POINT STORAGE

All points are protected by a point storage subroutine which checks to see if a pair of registers already contain coordinates before it stores a point there. If the user designates a point number which is already used, the program BEEPs and displays USED-NEW PT? If another point number is keyed in, the program tries to store the coordinate as that point. If instead the prompt is bypassed, the existing coordinates are overwritten. The program recognizes as unused, those points which have first been initialized (cleared) in program STOR. Thus, the first thing done at the beginning of a project is usually to initialize points and then store the coordinates of one or more points in STOR. It is important to note that point-protection is <u>disabled</u> if flag 00 is set; and that the BEEP is disabled if flag 02 is clear.

ANGLES

All the COGO (coordinate geometry) routines which use angles recognize the status of flag 03. If flag 03 is clear all angles (input and output) are in decimal degrees. If flag 03 is set, all angles are in degrees-minutes-seconds. This is shown clearly in all prompts and output as (dd) or (DMS).

INPUT FORMAT

Most of these routines require multiple input. Unlike most programs in this module which prompt for input one item at a time, many of the COGO routines require two, three, or four items to be input into the stack at once. This is signified in the prompt by an upward arrow, ↑ , meaning "Enter↑." Thus for the STOR prompt PT↑ N↑ E, key in the point number, enter, key in the northing, enter, key in the easting (<u>don't</u> enter), press R/S. The program uses whatever it finds in the stack. This can produce some shortcuts; for instance, if the coordinates of point #10 are (2,2), simply key in 10, enter, 2, enter, R/S. None of the routines use numbers in the stack above the numbers requested in the <u>initial</u> prompt, so if you realize a mistake was made before R/S is pressed, nothing needs to be cleared; just key in the right numbers. However, if a program has a second prompt (usually"N", asking for the point number to store the coordinates under, or "M", the "proximity" point), important data are carried in the stack; if a mistake is caught, use RDN (roll down) to avoid changing the necessary stack configuration. This rather poor explanation will become clearer with a little use, and perhaps an examination of the program listings with the stack usage traced. If at all uncertain, just re-execute the routine.

TEMPORARY POINTS

As mentioned, each point "N" stores its coordinates in registers 2N and 2N+1. Registers 00 and 01 are the only storage registers used by the programs as working registers, the rest of the work being done in the stack and synthetic registers (about which see the section "Synthetic Programming and Other Loose Ends" at the front of the documentation). This means that if you just want the coordinates of a point displayed but not stored, call it point #0, override the point protection, and it will be temporarily stored until another program needs registers 00 and 01, at which time it will be gone.

PROXIMITY POINT

Many of the routines dealing with arcs find two points instead of just one. The routines use a simple method to determine which one to store; they prompt for a "proximity" point (always prompted as point M) and pick the point closest to it. If both points are wanted, bypass the "M" prompt and they will both be stored. It's especially important in this case to keep a running sketch to designate which point is which.

PRINTER USE

Most routines, upon finishing, return to their initial prompt to go again. The first time each routine is executed, a line of asterisks is printed (that is, if a printer is present) across the paper. If a routine is used repeatedly (for example, LAZ to traverse around an area) it's difficult when reviewing the output to tell which routine is being used. The line of asterisks makes this a little clearer; directly above the last line is the function execution. See the examples for clarification. Using a printer is strongly recommended; tracing down mistakes without one is almost impossible.

LIMITATIONS

It's important to keep in mind the scope of this COGO package. The intended uses typically involve layouts of small structural components or minor land surveying tasks. While a completely "empty" HP-41CV or CX (one with no program storage or key assignments) is capable of storing up to 158 points, anything over thirty or forty points starts to get clumsy. Large complicated layouts can be done, but only with very strong concentration and constant cross-checking, because of some of the following disadvantages:

1) There is no graphics capability. Anyone used to on-screen graphics knows how useful a simple picture can be in terms of ease of use, confidence, and error detection. For anything more than about four points, it is almost essential to have a carefully scaled drawing with point numbers, distances, and azimuths shown. It is also a very good idea to write out the intended procedure, with input, before starting. Just using the routines and getting through the input sequences requires a lot of concentration. Don't try to think your way through the problem at the same time. And, as mentioned, a printer is extremely helpful. Don't let this discourage use of the routines, though; all computer coordinate geometry work was originally done on batch machines with exactly these limitations. They simply require a little caution and concentration.

2) Although stored points are protected from alteration by other COGO routines (if flag 00 is clear), the storage registers they reside in are used quite cavalierly by other programs, wiping out coordinates. However, only a few programs in this module (PIER, PLG, and sometimes FTG) use registers above #30. It may be worthwhile to start with point number 15 or so (or 25 if PIER may be used), leaving the lower-numbered registers for use by other (non-COGO) programs. Program DUP may also be used to duplicate a block of points to a safer (higher-numbered) location for

storage, to be used as re-numbered points or copied back down later.

3) The HP-41 is limited to ten significant figures. It's not a good idea to waste too many of them in front of the decimal point; use of more than four figures before the decimal point is not recommended. Also, intersections involving two lines with not-very-different bearings is inherently inaccurate, with any COGO package.

4) While angles may be input in bearing/quadrant mode, it is clumsy; and all output is expressed as north azimuths. At best, bearings are handled awkwardly by this package. Those used to working with bearings (i.e. surveyors and others who know more about these things than the author) will probably grit their teeth and use azimuths.

5) As stated, all points are two-dimensional. Elevations are not stored.

6) There is no provision for storing descriptors with points. Those used to large-scale professional coordinate geometry programs (the author included) will miss this greatly; good descriptors are second only to interactive graphics in inspiring confidence and ease of use. Both elevations and short description capabilities (one register, six letters) were considered. However, both complicate (swell) the programming immensely; also, each point would require four registers to store instead of just two, halving the available point storage. A realistic assessment of the uses, intended scope, and inherent limitations of this package made both elevations and descriptors seem an expensive luxury.

As mentioned five disadvantages back, this listing is intended to caution, not discourage. A great deal of useful, accurate work has been done on systems with exactly the limitations stated here. People used to the old batch programs will feel right at home using these programs. They simply require a little more care than their $12,000 siblings.

COMMAND   INPUT

STOR   I↑K CLR; PT↑N↑E (flag 01 clear)
        I↑K CLR; PT↑X↑Y (flag 01 set)

Clear (initialize) all points from I to K inclusive, then store the North and East (or X and Y) coordinates of points. The I↑K CLR prompt is not repeated; the PT↑... prompt is. If enough points have already been initialized, the I↑K CLR prompt may be bypassed.

—

DUMP   I↑K

Dump (display or list) the coordinates of points I through K. Skips free (initialized) points.

—

FREE   (No input)

Finds free or unused points (points which have been initialized (CLR'd) in STOR).

—

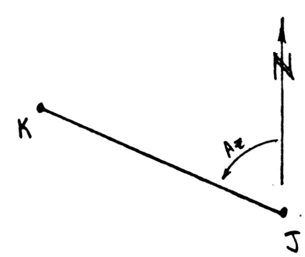
DUP   I↑K↑NEW I

Copies a block of points from I to K, to a new block starting at NEW I. Leaves the original points I through K intact. To just copy one point, key I,↑,↑, NEW I. Point protection is <u>not</u> active with DUP; it will overwrite points with with no warning. The user must also check for overlap.

—

INV   J↑K

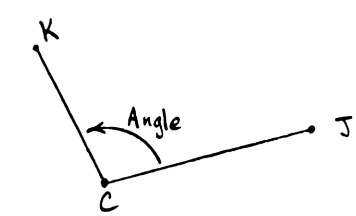
Finds the distance and azimuth between points J and K. The K point of one INV is the default J of the next. Thus to inverse from point #1 to #2, then #2 to #3, then #3 to #4: Xeq (alpha) INV (alpha); key in 1, Enter↑, 2, R/S; 3, R/S; 4, R/S. In addition, the sum of the distances is stored in register 00 for use by program ADJ.



—

ANGL   J↑C↑K

Find the angle between points J and K, measured about point C. The distances from J to C and K to C need not be the same.



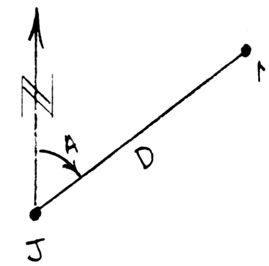
—

Note: the "↑ " symbol means "Enter↑." (dd) means "decimal degrees" (flag 03 clear); (DMS) means "Degrees-Minutes-Seconds" (flag 03 set). Clockwise angles are positive.

COMMAND INPUT

LAZ J/D/A (dd or DMS); N

Locate by azimuth. From point J, set point N at distance D and azimuth A.



—

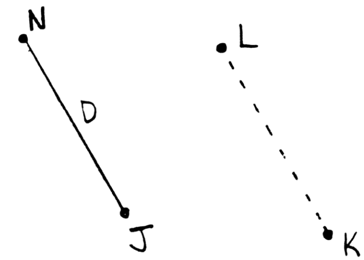
LBR J/D/B (dd or DMS); QUAD; N

Locate by bearing. From point J, set point N at a distance D and bearing B in quadrant QUAD.

—

LPT J/D/K/L; N

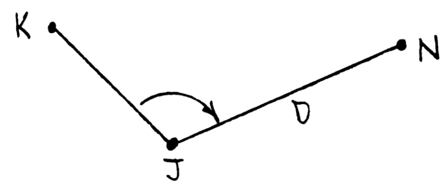
Locate by point bearing. From point J, set point N at a distance D, at the azimuth of the line through K to L.



—

LAN J/K/D/A (dd or DMS); N

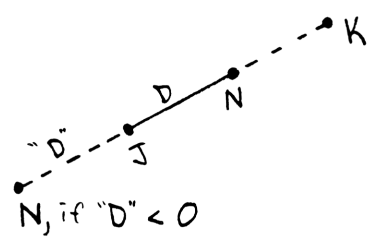
Locate by angle. From point J, backsight on point K; turn angle A (clockwise is positive) and set point N at distance D.



—

LLI J/K/D; N

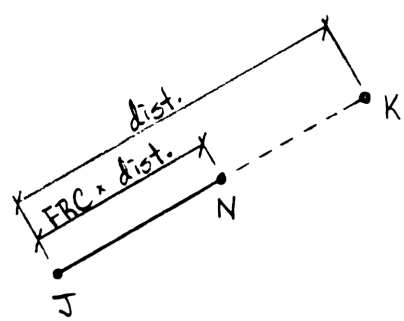
Locate on line. From J, set point N a distance D toward point K. A negative D sets point N in the opposite direction of K.



—

PART J/K/FRC; N

From point J, set point N a fraction FRC of the distance to point K. If FRC is greater then 1, N will be set beyond K. Like LLI, if FRC is negative, N will be set away from rather than toward K from point J.

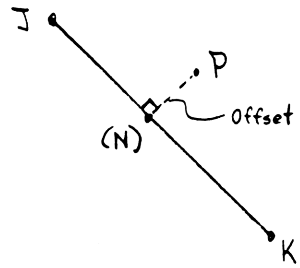


—

Note: the "/ " symbol means "Enter/." A semicolon means the items of input are prompted in separate groups. (dd) means "decimal degrees" (flag 03 clear); (DMS) means "Degrees-Minutes-Seconds" (flag 03 set). Clockwise angles are positive.

COMMAND INPUT

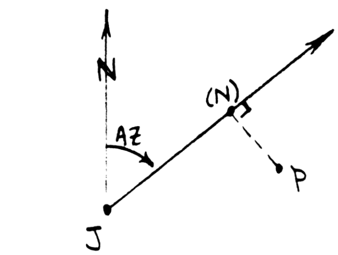
OFPT J/K/P; N



Offset by points. Find the offset of point P from the line through points J and K; set point N on line J-K. The prompt for point N may be bypassed (or answered 0) to avoid setting a point there.

—

OFAZ J/P/AZ (dd or DMS); N

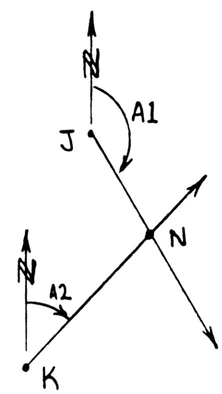


Offset by azimuth. Find the offset of point P from the line through J at azimuth AZ; set point N.

—

OFBR J/P/B (dd or DMS); QUAD; N

Offset by bearing. Find the offset of point P from the line through J at bearing B; set point N.



—

IAZ J/A1/K/A2; N

Intersect by azimuth. Set point N at the intersection of the line through point J at azimuth A1; with the line through point K at azimuth A2.

—

IBR J/B; QUAD; K/B; QUAD; N

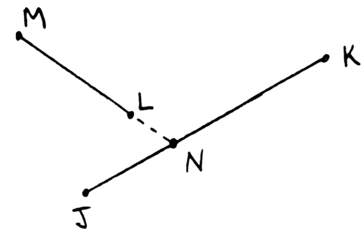
Intersect by bearing. Set point N at the intersection of the line through point J at bearing B; with the line through point K at bearing B.

—

Note: the "/ " symbol means "Enter/." A semicolon means the items of input are prompted in separate groups. (dd) means "decimal degrees" (flag 03 clear); (DMS) means "Degrees-Minutes-Seconds" (flag 03 set). Clockwise angles are positive.

COMMAND INPUT

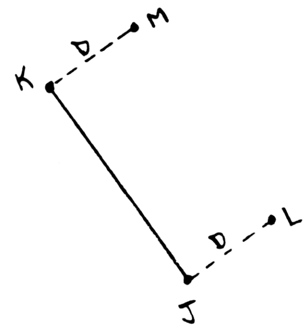
IPT J/K/L/M; N



Intersect by points. Set point N at the intersection of lines J-K and L-M. J and K need not be on opposite sides of line L-M and vice versa.

—

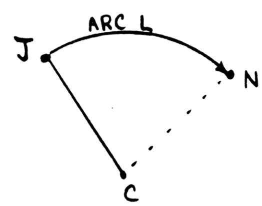
PRL J/K/D; L; M



Parallel line. Set points L and M a distance D from points J and K such that line L-M is parallel to line J-K. From J facing K, D is positive to the right.

—

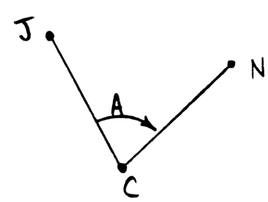
ATRV J/C/ARC L; N



Arc traverse. From point J, about point C, traverse a distance ARC L along the arc to set point N. ARC L is positive measured clockwise.

—

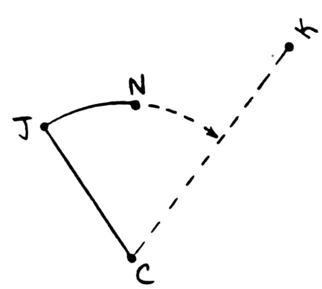
ATVA J/C/A (dd or DMS); N



Arc traverse by angle. From point J, about point C, turn an angle A to set point N on the arc. Angle A is positive measured clockwise. This function is identical to LAN except the distance (here, radius) is set by point J.

—

AFRC J/C/K/FRC; N



Arc fraction. Set point N a fraction FRC of the way along the arc from J to K about C. If J and K are not the same distance from C, the radius used is from J to C.

—

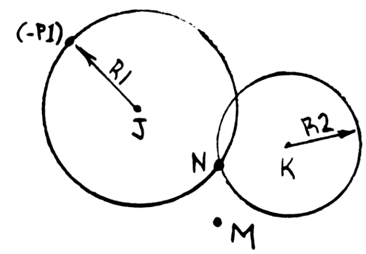
Note: the "/ " symbol means "Enter/." A semicolon means the items of input are prompted in separate groups. (dd) means "decimal degrees" (flag 03 clear); (DMS) means "Degrees-Minutes-Seconds" (flag 03 set).__Clockwise angles are positive.

B50

Note: The following arc commands have two possible solutions; the routines will pick the one closest to point M to store as point N (the solved point). If the prompt for M is bypassed, both solutions will be stored as N1 and N2.

If the prompt for a radius is answered as a negative number (e.g. -12) it is taken as point #12 and the radius to be used is the distance from that point to the center point.

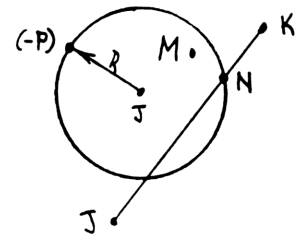
COMMAND	INPUT
ARCS	J/R1/K/R2; M; N (or N1; N2) J/-P1/K/-P2; M; N (or N1; N2)



Intersect arcs. Intersect the arc of radius R1 about point J, with the arc of radius R2 about point K, to set point N.

—

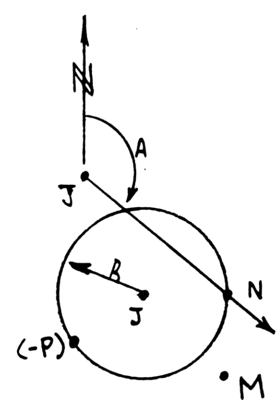
APT	J/R/K/L; M; N (or N1; N2) J/-P/K/L; M; N (or N1; N2)



Intersect arc by points. Intersect the arc of radius R about point J with the line through points K and L.

—

AAZ	J/R/K/A; M; N; (or N1; N2) J/-P/K/A; M; N; (or N1; N2)



Intersect arc by azimuth. Intersect the arc of radius R through point J with the line through point K at azimuth A.

—

ABR	J/R/K/B; QUAD; M; N J/-P/K/B; QUAD; M; N

Intersect arc by bearing. Intersect the arc of radius R through point J with the line through K at bearing B in quadrant QUAD.

—

Note: the "/ " symbol means "Enter/." A semicolon means the items of input are prompted in separate groups. (dd) means "decimal degrees" (flag 03 clear); (DMS) means "Degrees-Minutes-Seconds" (flag 03 set).

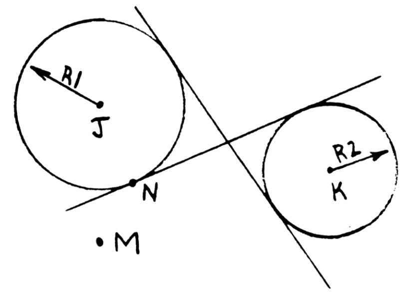
B51

Note: The following three arc commands have two possible solutions; the routines will pick the one closest to point M to store as point N (the solved point). If the prompt for M is bypassed, both solutions will be stored as N1 and N2.

If the prompt for a radius is answered as a negative number (e.g. -12) it is taken as point #12 and the radius to be used is the distance from that point to the center point.

COMMAND   INPUT

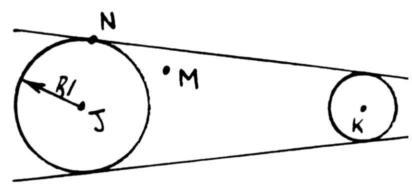
INTAN   J/R1/K/R2; M; N
J/-P1/K/-P2; M; N



Internal tangent. Finds the two points of internal tangency which are on the circle about point J.

—

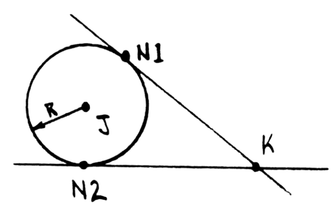
EXTAN   J/R1/K/R2; M; N
J/-P1/K/-P2; M; N



External tangency. Finds the two points of external tangency which are on the circle about point J.

—

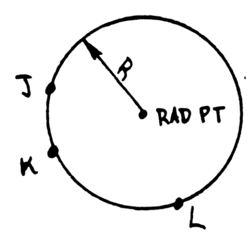
PTAN   J/R/K; M; N
J/-P/K; M; N



Tangents through a point. Finds the points of tangency to the circle about point J of the two lines through point K.

—

C3P   J/K/L; RAD PT



Circle from three points. Finds the radius and sets the center point RAD PT of the circle through any three non-colinear points.

—

Note: the "/ " symbol means "Enter/." A semicolon means the items of input are prompted in separate groups. (dd) means "decimal degrees" (flag 03 clear); (DMS) means "Degrees-Minutes-Seconds" (flag 03 set). Clockwise angles are positive.

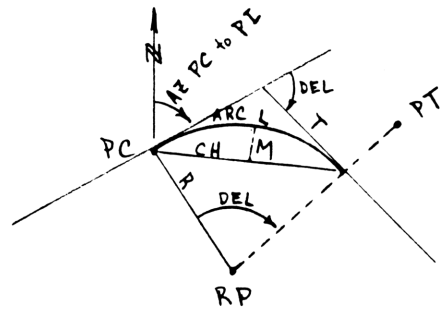
B52

COMMAND   INPUT

CHC   PC/RP/PT

Coordinate horizontal curve. Calculates the following horizontal curve data from the three points (point of curvature, radius point, and point of tangency):

Distance PC to RP
Distance PT to RP
Azimuth of incoming tangent (PC to PI)
Azimuth of outgoing tangent (PI to PT)



RAD	Radius, taken as PC to RP
DEG	Degree of curve
DEL	Delta or central angle
ARC L	Arc length
T	Tangent length (PC to PI)
CH	Chord length
M	Middle ordinate (arc to chord)
E	External (arc to PI)
SECT	Sector area
SEGM	Segment area
FIL	Fillet area

If the distances PC to RP and PT to RP aren't equal, the program displays a warning and uses the former as the radius. In this case the PT input is used only to find the angle subtended (Delta).

For more information see the documentation for program HC, of which CHC is a part.

—

Note: the "/ " symbol means "Enter/." A semicolon means the items of input are prompted in separate groups. (dd) means "decimal degrees" (flag 03 clear); (DMS) means "Degrees-Minutes-Seconds" (flag 03 set). Clockwise angles are positive.

B53

COMMAND INPUT

ADJ BEG PT; END PT; DIST'; PT#; PT#; PT#; etc.

Traverse adjustment by compass rule. Adjusts all point coordinates of a traverse proportionally to make the END PT close on the BEG PT (i.e. to make the END PT coordinates the same as the BEG PT coordinates).

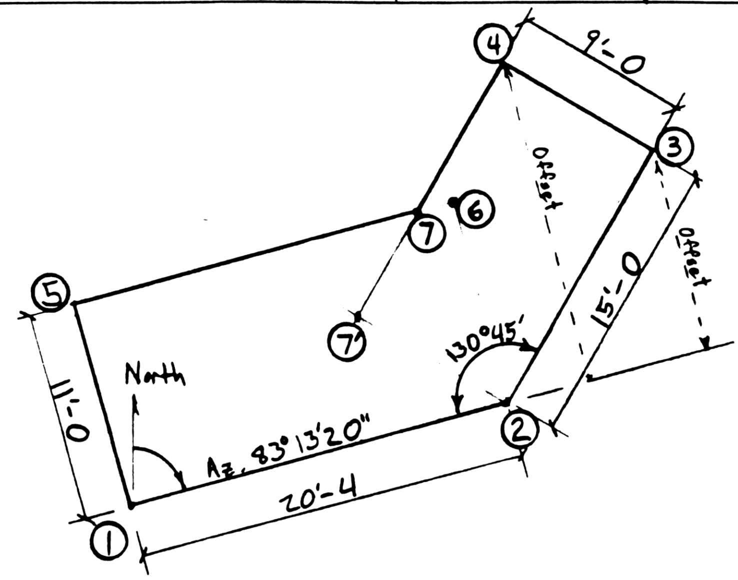
The Σ DIST' prompt asks for the total distance around the traverse, from BEG PT (beginning point, P.O.B.) to END PT. If this distance is known, simply key it in and go on. If not (the usual case), one of three routines may be run before the ADJustment to find the distance. If the points in the traverse are being set by LAZ or LBR from field notes, the distance around the traverse is accumulated (in register 00). If the points are already set, routine INV may be used to inverse around the loop; it also accumulates the distance. If using any of these three, they must run the entire traverse without stopping, since re-executing them zeroes the distance accumulated to start over. Once one of these three routines has been run from the BEG PT to the END PT, xeq ADJ, input the beginning and ending points, then bypass the Σ DIST' prompt (press R/S without keying anything in). The correct value will be inserted.

After the Σ DIST' prompt, ADJ begins prompting for points in the traverse. The first PT# prompt is the second point in the traverse, and so on; continue answering the PT# prompts until you've keyed in the END PT. Respond to the next PT# prompt with a 0 (zero), or just XEQ your next command (typically a DUMP to list the adjusted points). See the example. It is important when keying in the PT#'s not to disturb the stack, which carries needed data. Do not use the Enter↑ key; and if an incorrect number is keyed in and R/S has not yet been pressed, use RDN (rolldown) or CLX to remove it before keying in the correct number. Unfortunately, ADJ is very unforgiving of mistakes made during PT# entry, and since the points are altered as one goes around the traverse, it's also impossible to start over. Thus it is a very good idea to duplicate the original coordinates in a safe block of registers with routine DUP in order to provide a method of error recovery. DUP does not use register 00, so it does not interfere with the storage of a Σ DIST' value by LBR, LAZ, or INV as long as one of these three routines is run completely around the traverse first.

—

AREA PT1; PT2; PT3; PT4; etc.

Calculation of area bounded by any number of points (greater than two). Input points bounding the area in a clockwise direction. The same caution applies to the stack and the Enter↑ key as for routine ADJ. However, if an error is discovered after pressing R/S but before keying in the next point number, use softkey C (with USER mode on) to correct the error. See the example. To end input and calculate the area, key in 0 (zero).



a) Angles in Degrees-Minutes-Seconds
⇒ Set flag 03

Desire North/East format
⇒ Clear flag 01

b) Set point #1 : N = 20.00, E = 500.00

STOR 1↗20↗500 (R/S)

c) Set point #2 @ Dist = 20'-4, Az. 83°13'20"

LAZ 1↗20.333↗83.1320 ; 2

d) Set point #3 from #2 by backsighting on #1

LAN 2↗1↗15.0'↗130.45 ; 3

e) Set points #5 and #6 parallel to #1-#2 ;
set points #4 and #7 parallel to #3-#2

PRL 1↗2↗-11 (to the left) ; 5 ; 6
2↗3↗-9 (") ; 7 ; 4

f) Intersect lines #5-#6 and #4-#7 to set a new point #7:

IPT 5↗6↗4↗7 ; 7

g) Compute distances #5-#7 and #7-#4, and area ; and list point coordinates and remaining free points.

```
                    SF 03          Set flag
                                   03 for
                                   D.MS
          XEQ "STOR"
I↑K CLR
          1.0000 ENTER↑            Initialize
         10.0000   RUN             Pts. #1-10

11:07:25 AM  09/06/88

TITLE:
ABUTMENT FOOTING
                   RUN
*************************
PT↑N↑E
          1.0000 ENTER↑            STOR
         20.0000 ENTER↑            Pt #1
        500.0000   RUN
---------
PT↑N↑E
              XEQ "LAZ"
*************************
J↑D↑A (DMS)                        From
          1.0000 ENTER↑            Pt #1,
        20.33333 ENTER↑            go 20.33'
         83.1320   RUN             @ Az 83°13'20"
N
          2.0000   RUN             to set
                                   Pt #2
PT2: N=22.3997
 E=520.1912
---------
J↑D↑A (DMS)
              XEQ "LAN"
*************************
J↑K↑D↑A (DMS)                      From
          2.0000 ENTER↑            Pt #2,
          1.0000 ENTER↑            backsight on #1,
         15.0000 ENTER↑            go 15.0'
        130.4500   RUN             turning
N                                  130°45'
          3.0000   RUN             to set
                                   Pt #3
PT3: N=34.8393
 E=528.5731
---------
J↑K↑D↑A (DMS)
```

```
          XEQ "PRL"
************************
J↑K↑D
        1.0000 ENTER↑
        2.0000 ENTER↑
       11.0000    RUN
L
        5.0000    RUN

PT5: N=9.0769
  E=501.2982
---------
M
        6.0000    RUN

PT6: N=11.4766
  E=521.4894
---------
J↑K↑D
        1.0000 ENTER↑
        2.0000 ENTER↑
      -11.0000    RUN
L
        5.0000    RUN
USED-NEW PT?
                  RUN

PT5: N=30.9231
  E=498.7018
---------
M
        6.0000    RUN
USED-NEW PT?
                  RUN

PT6: N=33.3228
  E=518.8930
---------
J↑K↑D
        2.0000 ENTER↑
        3.0000 ENTER↑
       -9.0000    RUN
L
        7.0000    RUN

PT7: N=27.4288
  E=512.7275
---------
M
        4.0000    RUN

PT4: N=39.8685
  E=521.1093
---------
J↑K↑D
```

Oops - went the wrong way (should have been -11.0')

Do over

Bypass to overwrite

Bypass

```
          XEQ "IPT"
************************
J↑K↑L↑M
        5.0000 ENTER↑
        6.0000 ENTER↑
        4.0000 ENTER↑
        7.0000    RUN
N
        7.0000    RUN
USED-NEW PT?
                  RUN

PT7: N=33.0394
  E=516.5078
---------
J↑K↑L↑M
          XEQ "INV"
************************
J↑K
        5.0000 ENTER↑
        7.0000    RUN
D=17.9314
AZ=83.1320 DMS
---------
J↑K
        4.0000    RUN
D=8.2347
AZ=33.5820 DMS
---------
J↑K
          XEQ "OFPT"
************************
J↑K↑P
        1.0000 ENTER↑
        2.0000 ENTER↑
        4.0000    RUN
OFFSET=17.2383'
N
        0.0000    RUN

PT0: N=22.7506
  E=523.1438
---------
J↑K↑P
        1.0000 ENTER↑
        2.0000 ENTER↑
        3.0000    RUN
OFFSET=11.3635'
N
        0.0000    RUN

PT0: N=23.5553
  E=529.9142
---------
```

Overwrite the old Pt#7 with the new one.

From #5 to #7

From #7 (default) to #4

Find offset of pt #4 from line #1-#2

Set "dummy" point @ offset location on line #1-#2 to obtain coordinates

Offset of point #3 from line #1-#2

(B56)

Note that the first use (the "xeq") of each routine prints with a line of asterisks to make the output easier to follow.

```
J↑K↑P
         5.0000 ENTER↑
         6.0000 ENTER↑
         4.0000    RUN
OFFSET=6.2383'
N
                   RUN
J↑K↑P
           XROM "AREA"
************************
PT1
         1.0000    RUN
PT2
         2.0000    RUN
PT3
         3.0000    RUN
PT4
         4.0000    RUN
PT5
         5.0000    RUN
PT6
                 XEQ C
PT5
         7.0000    RUN
PT6
         6.0000    RUN
PT7
                 XEQ C
PT6
         5.0000    RUN
PT7
         0.0000    RUN
AREA=-315.0119 SQ'
---------
PT1
```

Offset of Pt#4 from line #5-#6

Bypass to avoid setting any point

Find area

Oops - the fifth point should be #7

Oops again - use softkey "C" to correct (USER mode on).

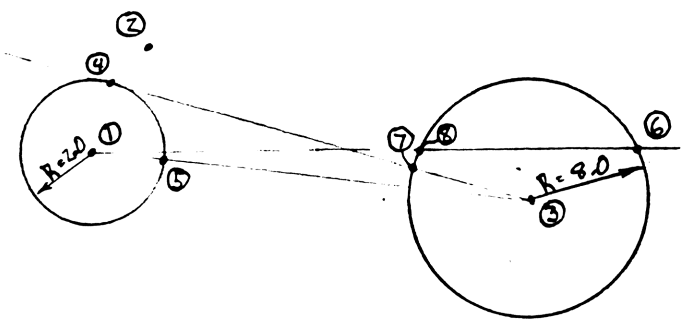
End with Pt# 0;

Points were input counter-clockwise so area displays as negative.

```
            XROM "FREE"
************************
FREE PTS:
  8
  9
  10
-----------------------
             XEQ "DUMP"
11:17:47 AM  09/06/88

TITLE:
ABUTMENT FOOTING     RUN
************************
I↑K
             1.0000 ENTER↑
            10.0000    RUN

PT1: N=20.0000
  E=500.0000
---------
PT2: N=22.3997
  E=520.1912
---------
PT3: N=34.8393
  E=528.5731
---------
PT4: N=39.8685
  E=521.1093
---------
PT5: N=30.9231
  E=498.7018
---------
PT6: N=33.3228
  E=518.8930
---------
PT7: N=33.0394
  E=516.5078
---------
I↑K
```

Have three points unused.

Points 8-10 are not listed since they are unused




```
            SF 01
```
Use (X,Y) rather than (North, East)

```
        XEQ "STOR"
I↑K CLR
     1.0000 ENTER↑
```
Initialize 9 points

```
     1.0000    CLX
```
oops

```
     9.0000    RUN

 4:57:51 PM  08/30/88

TITLE:
CURVES              RUN
************************
PT↑X↑Y
     1.0000 ENTER↑
     5.0000 ENTER↑
    15.0000    RUN
---------
PT↑X↑Y
     2.0000 ENTER↑
     6.0000 ENTER↑
    18.0000    RUN
---------
PT↑X↑Y
     3.0000 ENTER↑
    12.0000 ENTER↑
    14.0000    RUN
---------
PT↑X↑Y
        XEQ "PTAN"
************************
J↑R↑K
     1.0000 ENTER↑
     2.0000 ENTER↑
     3.0000    RUN
M
     2.0000    RUN
N
     4.0000    RUN

PT4: X=5.8313
  Y=16.8191
---------
J↑R↑K
```

Find the tangent point of the line through Pt #3, to the circle about Pt #1 of radius 2.0; of the two solutions, pick the one closest to Pt #2.

B58

```
          XEQ "APT"
************************
J↑R↑K↑L
     1.0000 ENTER↑
     2.0000 ENTER↑
     1.0000 ENTER↑
     3.0000    RUN
M
     3.0000    RUN
N
     5.0000    RUN

PT5: X=6.9799
  Y=14.7172
---------
J↑R↑K↑L
     3.0000 ENTER↑
     8.0000 ENTER↑
     3.0000 ENTER↑
     4.0000    RUN
M
     4.0000    RUN
N
    11.0000    RUN
NONEXISTENT
```

Intersect the circle about Point #3, of radius 8.0, with the line through points #3 and #4

Of the two solutions, pick the one closest to Pt #4.

Call it Pt #11

Oops - must re-execute - not enough data registers to store Pt #11

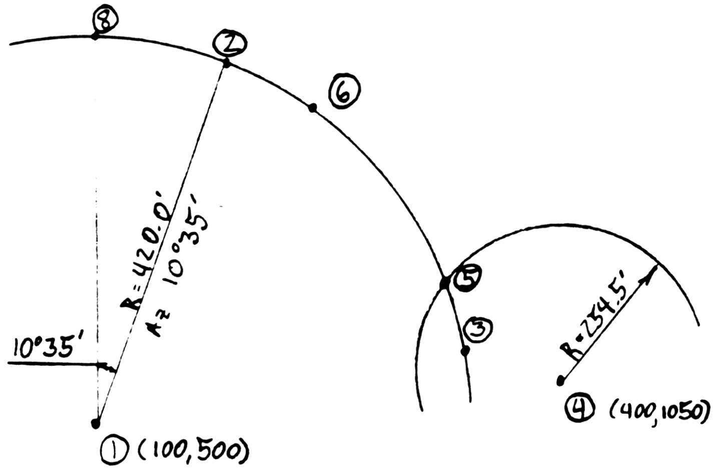
```
          XEQ "APT"
************************
J↑R↑K↑L
     3.0000 ENTER↑
     8.0000 ENTER↑
     3.0000 ENTER↑
     4.0000    RUN
M
     4.0000    RUN
N
     7.0000    RUN
```
Store as Pt #7

```
PT7: X=4.7238
  Y=17.3252
---------
J↑R↑K↑L
```

```
          XEQ "AAZ"
************************
 J↑R↑K↑A (dd)
     3.0000 ENTER↑
     8.0000 ENTER↑
     1.0000 ENTER↑
    85.0000    RUN
M
               RUN
N1
     6.0000    RUN

PT6: X=19.6673
  Y=16.2832
---------
N2
     8.0000    RUN

PT8: X=4.0527
  Y=14.9171
---------
 J↑R↑K↑A (dd)
          XEQ "FREE"
************************
FREE PTS:
  9
-----------------------
```

Pt #9 still available

Arc Example



a) Set point #1: STOR 1↗100↗500

b) Set point #2 from #1 @ R=420'
Az = 10°35':

LAZ 1↗420↗10°35'; 2

c) Set point #3 from #2, 300.0' along the arc about #1 (clockwise = +)

ATRV 2↗1↗300'; 3

d) Store coordinates of point #4 (bypass initializing sequence): N=400
E=1050

STOR (bypass CLR) 4↗400↗1050

e) Set point #5 @ intersection of arc about #4 @ R=254.5', with arc about #1 @ R=420.0' (radius of #3) pick the point closest to #2

ARCS 1↗-3↗4↗254.5; 2; 5
(3, CHS)

f) Set point #6 1/4 of the way between #2 and #3 along the arc about #1.

AFRC 2↗1↗3↗.25; 6

g) Set point #8, 10°35' counter-clockwise (-) from point #2 about #1

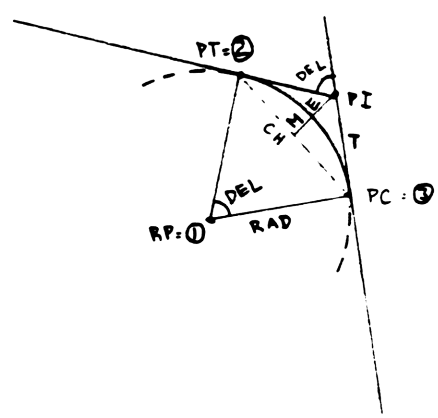
ATVA 2↗1↗-10.35; 8
(10.35, CHS)

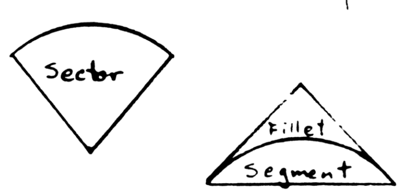
h) Find remaining free points and run out curve data from #3 to #2 about #1

```
                    CF 01        Work in (North, East)
                    SF 03        Work in D.MS

          XEQ "STOR"
I↑K CLR
          1.0000 ENTER↑          Initialize
          9.0000    RUN          points 1-9

 5:30:47 PM  08/30/88

TITLE:
ARCS
                 RUN
************************
PT↑N↑E
          1.0000 ENTER↑          Store Pt #1
        100.0000 ENTER↑
        500.0000    RUN
---------
PT↑N↑E
           XEQ "LAZ"
************************
J↑D↑A (DMS)                      Set Pt #2
          1.0000 ENTER↑
        420.0000 ENTER↑
         10.3500    RUN
N
          2.0000    RUN

PT2: N=512.8553
 E=577.1395
---------
J↑D↑A (DMS)
          XEQ "ATRV"
************************
J↑C↑ARC L
          2.0000 ENTER↑          Set Pt #3
          1.0000 ENTER↑
        300.0000    RUN
N
          3.0000    RUN

PT3: N=361.4051
 E=828.7360
---------
J↑C↑ARC L
```

```
                 XEQ "STOR"
I↑K CLR
                        RUN
PT↑N↑E
           4.0000 ENTER↑
         400.0000 ENTER↑
       1,050.0000    RUN
---------
PT↑N↑E
                 XEQ "ARCS"
************************
J↑R1↑K↑R2
           1.0000 ENTER↑
          -3.0000 ENTER↑
           4.0000 ENTER↑
         254.5000    RUN
M
           2.0000    RUN
N
           5.0000    RUN

PT5: N=398.4577
  E=795.5047
---------
J↑R1↑K↑R2
                 XEQ "AFRC"
************************
J↑C↑K↑FRC
           2.0000 ENTER↑
           1.0000 ENTER↑
           3.0000 ENTER↑
            .2500    RUN
N
           6.0000    RUN

PT6: N=492.5884
  E=649.2458
---------
J↑C↑K↑FRC
                 XEQ "ATVA"
************************
J↑C↑A (DMS)
           2.0000 ENTER↑
           1.0000 ENTER↑
         -10.3500    RUN
N
           8.0000    RUN

PT8: N=520.0000
  E=500.0000
---------
J↑C↑A (DMS)
                 XEQ "FREE"
************************
FREE PTS:
  7
  9
-----------------------
```

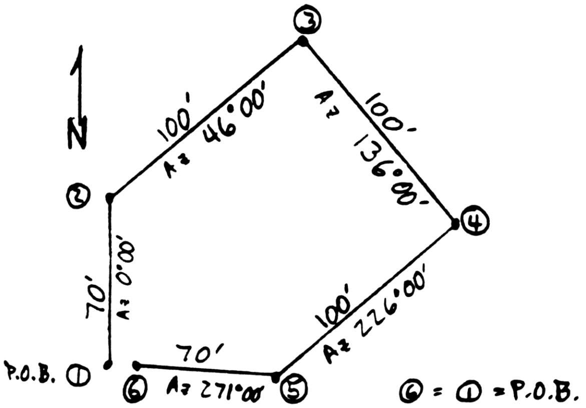
```
                   XROM "CHC"
 5:45:03 PM  08/30/88

TITLE:
420 FT RAD.           RUN
************************
PC↑RP↑PT
           3.0000 ENTER↑
           1.0000 ENTER↑
           2.0000    RUN
PC-RP=420.0000'
PT-RP=420.0000'
PC-PI AZ=-38.2928 DMS
PI-PT AZ=-79.2500 DMS

RAD=420.0000'
DEG=13.3831 DMS
DEL=40.5532 DMS
ARC L=300.0000'
T=156.7207'
CH=293.6630'
M=26.5022'
E=28.2871'
---------
SECT=63,000.0000 SQ'
SEGM=5,222.1295 SQ'
FIL=2,822.6734 SQ'
-----------------------
PC↑RP↑PT
```





B60

Traverse this circuit and adjust points by the compass rule to close.



⑥ = ① = P.O.B.

Perimeter (ΣDIST) = 440.0'

Traversing from ① (30,530) to ⑥ (29.2877, 529.4765)

Error in Northing: 30.0 - 29.2877 = .7123'

" " Easting: 530.0 - 529.4765 = .5235'

xeq STOR

I↗K CLR	1, 6
PT↗N↗E	1, 30, 530

xeq LBR

J↗D↗B (dd); QUAD; N

1, 70', 0°; 1; 2

2, 100', 46°; 1; 3

3, 100', 44°; 2; 4

4, 100', 46°; 3; 5

5, 70', 89°; 4; 6

xeq ADJ

BEG PT	1
END PT	6
ΣDIST'	(R/S) (default from LBR)
PT #	2
PT #	3
	etc.

```
                  XEQ "STOR"
I↑K CLR
              1.0000 ENTER↑
              6.0000    RUN
SIZE<14
                    SIZE 014
                  XEQ "STOR"
I↑K CLR
              1.0000 ENTER↑
              6.0000    RUN
 4:37:43 PM  08/29/88

TITLE:
TRAVERSE ADJUSTMENT  RUN
************************
PT↑N↑E
              1.0000 ENTER↑
             30.0000 ENTER↑
            530.0000    RUN
---------
PT↑N↑E
                   XEQ "LBR"
************************
J↑D↑B (dd)
              1.0000 ENTER↑
             70.0000 ENTER↑
              0.0000    RUN
QUAD
              1.0000    RUN
N
              2.0000    RUN

PT2: N=100.0000
  E=530.0000
---------
J↑D↑B (dd)
              2.0000 ENTER↑
            100.0000 ENTER↑
             46.0000    RUN
QUAD
              1.0000    RUN
N
              3.0000    RUN

PT3: N=169.4658
  E=601.9340
---------
J↑D↑B (dd)
              3.0000 ENTER↑
            100.0000 ENTER↑
             44.0000    RUN
QUAD
              2.0000    RUN
N
              4.0000    RUN

PT4: N=97.5319
  E=671.3998
---------
```

Initialize points 1-6

Oops! Not enough room - reSIZE

Try again

Initialize

Title

STOR coordinates of P.O.B. N = 30.0 E = 530.0

Begin traverse

From pt. #1, go 70.0' at a bearing of 0.00° in Quad. 1 (Northeast) to set point #2

Coord. of point #2

```
J↑D↑B (dd)
          4.0000 ENTER↑
        100.0000 ENTER↑
         46.0000    RUN
QUAD
          3.0000    RUN
N
          5.0000    RUN

PT5: N=28.0660
  E=599.4658
---------
J↑D↑B (dd)
          5.0000 ENTER↑
         70.0000 ENTER↑
         89.0000    RUN
QUAD
          4.0000    RUN
N
          6.0000    RUN

PT6: N=29.2877
  E=529.4765
---------
J↑D↑B (dd)
              XEQ "ADJ"
************************
BEG PT
          1.0000    RUN
END PT
          6.0000    RUN
Σ DIST'
                    RUN
PT#
          2.0000    RUN
PT#
          3.0000    RUN
PT#
          4.0000    RUN
PT#
          5.0000    RUN
PT#
          6.0000    RUN
PT#
          0.0000    RUN
BEG PT
```

Point #6 didn't quite close on point #1

Bypass since ΣDIST stored by LBR run

Begin with first point to be adjusted - proceed around traverse

Signify end of traverse with point #0

```
                  XEQ "DUMP"
 7:36:38 AM  08/16/88

TITLE:
ADJUSTED POINTS      RUN
************************
I↑K
             1.0000 ENTER↑
             6.0000    RUN

PT1: N=30.0000
  E=530.0000
---------
PT2: N=100.1133
  E=530.0833
---------
PT3: N=169.7410
  E=602.1362
---------
PT4: N=97.9690
  E=671.7211
---------
PT5: N=28.6650
  E=599.9061
---------
PT6: N=30.0000
  E=530.0000
---------
I↑K
                XROM "AREA"
************************
PT1
             1.0000    RUN
PT2
             2.0000    RUN
PT3
             3.0000    RUN
PT4
             4.0000    RUN
PT5
             5.0000    RUN
PT6
             0.0000    RUN
AREA=12,443.9676 SQ'
---------
PT1
```

Adjusted Points - #6 = #1

Find Area of adjusted traverse

Signify end of traverse with point #0

Traversing from ① around the loop to close @ ⑥. From the azimuths and distances measured, the coordinates of the points are calculated and the error of closure is found:

	Northing	Easting
Pt ①	30.00	530.00
Pt ⑥	29.2877	529.4765
error	0.7123'	0.5235'

Leg	Dist.
1	70'
2	100'
3	100'
4	100'
5	70'
Σ	440'

Adjust the coordinates to close the traverse (by Compass Rule).

Point	Dist.	Northing	Correction to Northing	Accum.	Adj. N	Easting	Correction East.	Accum.	Adj. Easting	Point
1		30.00	—	0	30.00	530.00	—	0	530.00	1
	70'									
2		100.00	$\frac{70'}{440'} \times .7123' = .1133'$	.1133	100.1133	530.00	$\frac{70'}{440'} \times .5235 = .0833$	.0833	530.0833	2
	100'									
3		169.4658	$\frac{100'}{440'} \times .7123' = .1619'$	.2752	169.7410	601.9340	$\frac{100'}{440'} \times .5235' = .1190$	.2023	602.1363	3
	100'									
4		97.5319	.1619'	.4371	97.9690	671.3998	.1190	.3212	671.7210	4
	100'									
5		28.0660	.1619'	.5990	28.6650	599.4658	.1190	.4402	599.9060	5
	70'									
6		29.2877	.1133'	.7123	30.00	529.4765	.0833	.5235	530.00	6

If coordinates are already stored, use INV to accumulate distance :

```
            XEQ "INV"
************************
J↑K
        1.0000 ENTER↑
        2.0000    RUN
D=70.0000
AZ=0.0000 dd
---------
J↑K
        3.0000    RUN      from #2 to #3
D=100.0000                 (default "J" is
AZ=46.0000 dd               previous "K")
---------
J↑K
        4.0000    RUN
D=100.0000
AZ=136.0000 dd
---------
J↑K
        5.0000    RUN
D=100.0000
AZ=-134.0000 dd
---------
J↑K
        6.0000    RUN      Last leg is
D=70.0000                  from #5 to #6
AZ=-89.0000 dd
---------
J↑K
            XEQ "ADJ"
************************
BEG PT
        1.0000    RUN
END PT
        6.0000    RUN
Σ DIST'
                  RUN      Bypass
PT#
        2.0000    RUN      First point to
PT#                        be corrected
        3.0000    RUN
PT#
        4.0000    RUN
PT#
        5.0000    RUN
PT#
        6.0000    RUN      Last point to be
PT#                        adjusted
        0.0000    RUN      Signify end with
BEG PT                     Pt = 0
```

```
            XEQ "DUMP"
 4:49:00 PM  08/29/88

TITLE:
ADJUSTED POINTS      RUN
************************
I↑K
        1.0000 ENTER↑
        6.0000    RUN

PT1: N=30.0000
  E=530.0000
---------
PT2: N=100.1133
  E=530.0833
---------
PT3: N=169.7410
  E=602.1362
---------
PT4: N=97.9690
  E=671.7211
---------
PT5: N=28.6650
  E=599.9061
---------
PT6: N=30.0000
  E=530.0000
---------
I↑K
            XROM "AREA"
************************
PT1
        1.0000    RUN
PT2
        2.0000    RUN
PT3
        3.0000    RUN
PT4
        6.0000    RUN      Oops
PT5
              XEQ C        Correct with
PT4                        soft key "C"
        4.0000    RUN
PT5
        5.0000    RUN
PT6
        6.0000    RUN      Having a bad
PT7                        day - end
              XEQ C        with Pt=0
PT6                        (do not use
        0.0000    RUN      Pt=6 since
AREA=12,443.9676 SQ'       it is now
---------                  identical to
PT1                        Point #1
```

Program ________ For Coordinate Geometry 1282 bytes

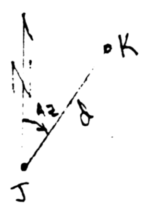
Line	X	Y	Z	T	L
01♦LBL "STOR"					
02 "I↑K CLR"					
03 CLX					
04 PROMPT	K	I			
05 X=0?					
06 GTO 10					
07 ENTER↑					
08 ST+ X	2K	K	I		
09 E	1	2K	K	I	
10 +					
11 ADV	2K+1	K	I		
12 XROM "S"	K	I			
13 E3					
14 /					
15 +	I.00K				
16 E-5					
17 +					
18 ST+ X	2I.002K02				
19 E9					
20 CHS	-1×10^9	counter			
21♦LBL 04					
22 STO IND Y					
23 ISG Y					
24 GTO 04					
25♦LBL 10					
26 "PT↑N↑E"	E	N	Pt#		
27 FS? 01					
28 "PT↑X↑Y"	Y	X	Pt#		
29 PROMPT					
30 FS? 01					
31 X<>Y	X	Y	Pt#		
32 X<> Z	Pt#	Y	X		
33 XEQ 30					
34 XROM "L"					
35 GTO 10					
36♦LBL 30					
37 SF 21					
38 FC? 02					
39 CF 26					
40 ST+ X	2P	Y	X		
41 SIGN	1				
42 RDN	Y	X			
43 CF 10					
44 RCL IND L	X'				
45 E9					
46 CHS	-1×10^9	X'	Y	X	
47 X=Y?	clear?				
48 SF 10					
49 X≠Y?					
50 BEEP					
51 R↑					
52 R↑	Y	X			
53 "USED-NEW PT#"					
54 CF 22					
55 FC? 10					
56 FS? 00					
57 FS? 53					
58 PROMPT	NewPt#	Y	X		
59 FS? 22					
60 GTO 30					
61 X<>Y	X	Y			
62 STO IND L	X				2P
63 ISG L					
64 ""	"NOP"				
65 X<>Y	Y	X			
66 STO IND L	Y				2P+1
67 LASTX	2P+1				
68 E	1	2P+1	Y	X	
69 -					
70 2					
71 /	P	Y	X	X	
72 RTN					
73♦LBL 16					
74 FC? 22					
75 RTN	Y	X			
76 X>0?					
77 XEQ 30					
78 ADV					
79♦LBL 56					
80 "PT"					
81 RCL d	flags	Pt#	Y	X	If Pt# > 0, store the coords.
82 FIX 0					
83 CF 29					
84 ARCL Y					
85 STO d					
86 RDN	P	Y	X		
87 FS? 01					
88 GTO 00					
89 "⊢: N="					
90 ARCL Y					
91 AVIEW					
92 " E="					
93 ARCL Z					
94 AVIEW					
95 XROM "L"					
96 RTN	P	Y	X		

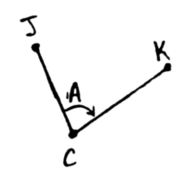
B65

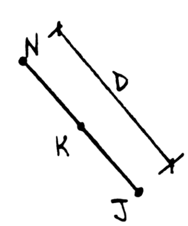
Program __________ For Coordinate Geometry 1282 bytes

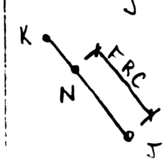
	X	Y	Z	T	L
97♦LBL 00					
98 "⊢: X="	X	-1×10^9	Pt#	Y	
99 ARCL Z					
100 AVIEW					
101 " Y="					
102 ARCL Y					
103 AVIEW					
104 XROM "L"					
105 RTN					
106♦LBL "INV"					
107 SF 14					
108 CLX					
109 STO 00					initialize R.00
110♦LBL 07					
111 XROM "L"					
112 "J↑K"					
113 PROMPT	K	J			
114 STO 01	K				
115 XROM "XY"	ΔX	ΔY	X_J	Y_J	
116 X<>Y	ΔY	ΔX			
117 R-P	d	Az			
118 ST+ 00					accum. "d"
119 "D="					
120 ARCL X					
121 AVIEW					
122 X<>Y	Az	d			
123 "AZ"					
124 XROM "Dd"					
125 AVIEW					
126 RCL 01	K	Az	d		
127 GTO 07					
128♦LBL "DUP"					
129 SF 14					
130♦LBL 03					
131 XROM "L"					
132 "I↑K↑NEW I"					
133 PROMPT	New I	K	I		
134 2	2	N	K	I	
135 ST* T					
136 ST* Z					
137 *	2N	2K	2I		
138 STO 00	2N				
139 X<>Y	2K				
140 E	1	2K	2N	2I	
141 +					
142 STO Y	2K+1	2K+1	2I	2I	
143 R↑	2I				
144 -	#reg.	2K+1	2I	2I	
145 E3	1000	#reg	2K+1	2I	
146 ST/ Z					
147 /	.00#	$\frac{2K+1}{1000}$	2I	2I	
148 ST+ 00					

	X	Y	Z	T	L
149 RDN					
150 +	2I.002K+1				
151 X<> 00	2N.00#				
152 STO 01					
153 E-3	.001				
154 +					
155 LASTX	.001	2N.00#	2I		
156 *					
157 +	2I.002N00#				
158 SF 25					Use REGMOVE if available
159 REGMOVE					
160 FS?C 25					
161 GTO 03					
162♦LBL 76					
163 RCL IND 00	X or Y				
164 STO IND 01	X or Y				
165 ISG 01					
166 ""					
167 ISG 00					
168 GTO 76					
169 GTO 03					
170♦LBL "ANGL"					
171 SF 14					
172♦LBL 11					
173 XROM "L"					
174 "J↑C↑K"	K	C	J		
175 PROMPT					
176 STO 01	K				
177 RDN					
178 STO 00	C	J			
179 XROM "XY"	ΔX_J	ΔY_J			
180 R-P	d_J	θ_J			
181 RDN	θ_J				
182 X<> 01	K				θ_J in R.01
183 RCL 00	C	K			
184 XROM "XY"	ΔX_K	ΔY_K			
185 R-P	d_K	θ_K			
186 RDN	θ_K				
187 RCL 01	θ_J				
188 -	$\Delta\theta$				
189 "A"					
190 XROM "Dd"					
191 AVIEW					
192 GTO 11					
193♦LBL "LLI"					
194 XROM "L*"					
195 SF 08					
196 GTO 09					
197♦LBL "PART"					
198 XROM "L*"					
199 CF 08					





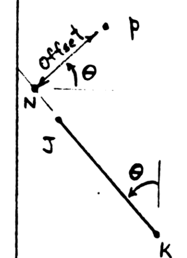


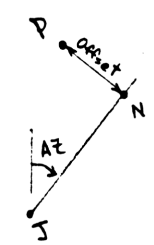


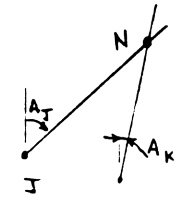
B66

Step	X	Y	Z	T	L
200♦LBL 09					
201 "J↑K↑"					
202 FC? 08					
203 "⊦FRC"	FRC	K	J		
204 FS? 08					
205 "⊦D"	D	K	J		
206 PROMPT					
207 STO 00	D or FRC				
208 RDN	K	J			
209 XROM "XY"	ΔX	ΔY	X_J	Y_J	
210 R-P	d_K	θ			
211 FC? 08					[illegible]
212 ST* 00					[illegible]
213 X<> 00	d_N	θ	X_J	Y_J	FRC in F.00
214 P-R	$ΔX_N$	$ΔY_N$			
215 X<>Y	$ΔY_N$	$ΔX_N$	X_J	Y_J	
216 "N"					
217 XROM "SP"					
218 GTO 09					
219♦LBL "XY"	K	J			
220 XROM "P"	Y_K	X_K	J		
221 X<> Z	J	X_K	Y_K		
222 XROM "P"	Y_J	X_J	X_K	Y_K	
223 RDN	X_J	X_K	Y_K	Y_J	
224 -	ΔX				
225 LASTX	X_J	ΔX	Y_K	Y_J	
226 X<> Z	Y_K	ΔX	X_J	Y_J	
227 R↑					
228 -	ΔY	ΔX	X_J		
229 X<>Y	ΔX	ΔY	X_J		
230 LASTX	Y_J				
231 RDN					
232 RTN	ΔX	ΔY	X_J	Y_J	
233♦LBL "FS"					
234 "⊦ SQ"					
235♦LBL "FT"					
236 "⊦'"					
237 RTN					
238♦LBL "DUMP"					
239 E	1				
240 XROM "S"					
241♦LBL 02					
242 "I↑K"					
243 PROMPT	K	I			
244 E3	1000				
245 /					
246 +	I.00K				
247 ADV					

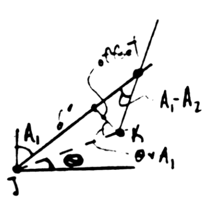
Step	X	Y	Z	T	L
248♦LBL 51					
249 ENTER↑	Pt#	Pt#			
250 XROM "P"	Y	X	Pt#		
251 RCL Z	Pt#	Y	X		
252 XEQ 56	Pt#	Y	X		
253 ISG X	Pt#+1				
254 GTO 51					
255 GTO 02					
256♦LBL "OFPT"					
257 XROM "L*"					
258♦LBL 14					
259 SF 08					
260 "J↑K↑P"					
261 PROMPT	P	K	J		
262 STO 00	P				
263 X<> Z	J	K	P		
264 STO 01	J				
265 XROM "XY"	ΔX	ΔY			
266 X<>Y	ΔY	ΔX			
267 R-P	d	θ			
268 RCL 01	J				
269 RCL 00	P	J	d	θ	
270 R↑	θ	P	J		
271 XEQ 66					
272 GTO 14					
273♦LBL "OFAZ"					
274 XROM "L*"					
275♦LBL 64					
276 "J↑P↑AZ"					
277 XROM "Pd"	Az	P	J		
278 XEQ 66					
279 GTO 64					
280♦LBL "OFBR"					
281 XROM "L*"					
282♦LBL 65					
283 "J↑P"					
284 XROM "BA"	Az	P	J		
285 XEQ 66					
286 GTO 65					
287♦LBL "IAZ"					
288 XROM "L*"					
289♦LBL 13					
290 CF 08					
291 " J↑A1↑K↑A2"					
292 XROM "Pd"	A_K	K	A_J	J	
293 X<> Z	A_J	K	A_K	J	



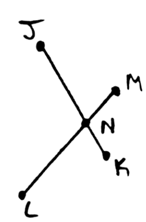




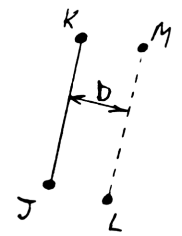
	X	Y	Z	T	L
294 FS? 03					
295 HR					
296 X<> Z	A_K	K	A_J	J	
297 XEQ 01					
298 GTO 13	N	Y_N	X_N	Reg "d"	
299♦LBL "IBR"					
300 XROM "L*"					
301♦LBL 43					
302 CF 08					
303 "J"					
304 XROM "BA"					
305 "K"					
306 XROM "BA"					
307 XEQ 01					
308 GTO 43					
309♦LBL 66	Az	P	J		
310 SF 08					
311 90	90	Az	P	J	
312 X<>Y	Az	90			
313 -	90-Az				
314 CHS	Az-90	P	J	J	
315 R↑	J				
316 LASTX	Az	J	Az-90	P	
317 R↑					
318 R↑	Az-90	P	Az	J	
319♦LBL 01	A_2	K	A_1	J	
320 STO 00	A_2				
321 RDN	K	A_1	J		
322 X<>Y	A_1	K	J		
323 STO 01	A_1				
324 RDN	K	J			
325 XROM "XY"	ΔX	ΔY	X_J	Y_J	
326 R-P	d	θ			
327 FC? 08					
328 GTO 00					
329 STO [	d				
330 RDN	θ	X_J	Y_J	d	
331 STO a	θ				
332 RCL 01	A_1				
333 +	$θ+A_1$				
334 RCL [	d				
335 P-R	offset	d'	X_J	Y_J	
336 "OFFSET="					
337 ARCL X					
338 "⊢ "					
339 AVIEW					
340 R↑					
341 R↑	X_J	Y_J			
342 RCL a	θ				
343 LASTX	d	θ	X_J	Y_J	



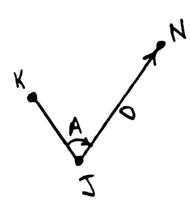
	X	Y	Z	T	L
344♦LBL 00	d	θ	X_J	Y_J	
345 X<> 00	A_2	θ	X_J	Y_J	d in R.00
346 +					
347 LASTX	A_2	$θ+A_2$	X_J	Y_J	
348 X<> 00	d	$θ+A_2$			A_2 in R.00
349 P-R	l	m			
350 X<>Y					
351 RDN	l	X_J	Y_J		l in R.00
352 X<> 00	A_2				
353 RCL 01	A_1				
354 -	A_2-A_1				
355 CHS	A_1-A_2				
356 SIN	sin				
357 ST/ 00					leaves D in R.00
358 RDN	X_J	Y_J			
359 RCL 01	A_1				
360 RCL 00	D	A_1			
361 P-R	ΔY	ΔX	X_J	Y_J	
362 GTO 00					
363♦LBL 06	Az	D	J		
364 X<>Y	D	Az	J		
365 ST+ 00	accum.				
366 P-R	ΔY	ΔX	J		
367 X<> Z	J	ΔX	ΔY		
368 XROM "P"	Y_J	X_J	ΔX	ΔY	
369♦LBL 00					
370 "N"					
371♦LBL "SP"	ΔY	ΔX	X_J	Y_J	
372 ST+ T	ΔY	ΔX	X_J	Y_N	
373 RDN	ΔX	X_J	Y_N	ΔY	
374 +	X_N	Y_N	ΔY	ΔY	
375 X<>Y	Y_N	X_N			
376 CF 22					
377 PROMPT	N	Y_N	X_N		
378 GTO 16					
379♦LBL "IPT"					
380 XROM "L*"					
381♦LBL 18					
382 CF 08					
383 "J↑K↑L↑M"					
384 PROMPT	M	L	K	J	
385 STO [	M				
386 RDN					
387 STO 01	L				
388 RDN					
389 STO 00	K	J	M	L	
390 XROM "XY"	ΔX	ΔY	X_J	Y_J	
391 X<>Y	ΔY	ΔX			
392 R-P	d	Az			
393 X<>Y	Az_{JL}				
394 X<> [	M				Az_{JL} in R.M
395 RCL 01	L	M			



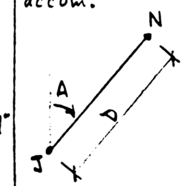
Step	X	Y	Z	T	L	Notes
396 XROM "XY"	ΔX	ΔY				
397 X<>Y	ΔY	ΔX				
398 R-P	d	Az				
399 X<> 00	K	Az_{LM}				
400 RCL [	Az_{JK}					
401 RCL 01	L					
402 RDN	Az_{JK}	K	Az_{LM}	L		
403 XEQ 01	N	Y_N	X_N	R.d		
404 GTO 18						
405♦LBL "PRL"						
406 XROM "L*"						
407♦LBL 05						
408 "J↑K↑D"						
409 PROMPT	D	K	J			
410 STO [	D					
411 RDN	K	J				
412 XROM "XY"	ΔX	ΔY	X_J	Y_J		
413 STO 00	ΔX					
414 X<>Y						
415 STO 01	ΔY	ΔX				
416 R-P	d	Az				
417 X<> [	D	Az				
418 P-R	ΔX	−ΔY				
419 X<>Y						
420 CHS	ΔY	ΔX	X_J	Y_J		
421 "L"						
422 XROM "SP"	L	Y_L	X_L	R.d		prompt must be answered; L≠0
423 X<> Z	X_L	Y_L	L			
424 RCL 00	ΔX					
425 RCL 01	ΔY	ΔX	X_L	Y_L		
426 "M"						
427 XROM "SP"	M	Y_M	X_M	R.d		
428 GTO 05						
429♦LBL "LAN"						
430 XROM "L*"						
431♦LBL 12						
432 "J↑K↑D↑A"						
433 XROM "Pd"	A	D	K	J		
434 STO [	A					
435 RDN						
436 STO 01	D					
437 ST+ 00						
438 RDN	K	J				
439 XROM "XY"	ΔX	ΔY	X_J	Y_J		
440 R-P	d	θ				
441 X<> [	A	θ				
442 -	θ−A					
443 RCL 01	D	θ−A				
444 P-R	ΔX	ΔY				
445 X<>Y	ΔY	ΔX	X_J	Y_J		
446 "N"						
447 XROM "SP"	N	Y_N	X_N	R.d		
448 GTO 12						

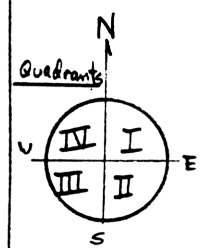


"D" is pos. to the right as looking from J to K



Step	X	Y	Z	T	L	Notes
449♦LBL "LAZ"						
450 XROM "L*"						
451 CLX						
452 STO 00						initialize Reg.00 for distance accum.
453♦LBL 08						
454 "J↑D↑A"						
455 XROM "Pd"	Az	D	J			
456 XEQ 06						
457 GTO 08	N	Y_N	X_N	Reg."d"		
458♦LBL "LBR"						
459 XROM "L*"						
460 CLX						
461 STO 00						
462♦LBL 48						
463 "J↑D"						
464 XROM "BA"	Az	D	J			
465 XEQ 06						
466 GTO 48						
467♦LBL "BA"						
468 "I↑B"						
469 XROM "Pd"	Brg	a	b	c		
470 STO 01	Brg					
471 RDN	a	b	c			
472 "QUAD"						
473 PROMPT	Q	a	b	c		
474 R↑						
475 STO [	c					
476 RDN	Q					
477 E	1	Q	a	b		
478 X≠Y?						
479 CHS ✗	±1	Q	a	b		
480 ST* 01						
481 RDN	Q	a	b			
482 CF 09						
483 1.6						
484 -	Q−1.6	a	b	b		
485 X<0?						
486 SF 09 *						
487 LASTX						
488 -	Q−3.2					
489 X>0?						
490 SF 09 **						
491 180	180	Q−3.2	a	b		
492 FS? 09						
493 CLX						
494 ST+ 01	180 or 0					
495 RDN	Q−3.2					
496 E	1					
497 +	Q−2.2	a	b	b		
498 SIGN	±1					
499 CHS #	∓1	a	b	b		





Quad I: Az=Brg
II: Az=180−Brg
III: Az=Brg−180
IV: Az=−Brg

✗ −1 ⇒ Quad II, III or IV
* FS?09 ⇒ Quad I
** FS?09 ⇒ Quad IV or I
# −1 ⇒ Quad III or IV
+1 ⇒ Quad I or II

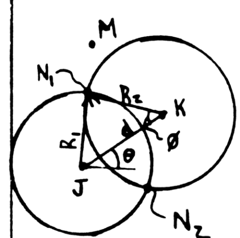
Program __________ For Coordinate Geometry 1282 bytes Page 6 of 6

Step	X	Y	Z	T	L
500 FS?C 09					
501 ABS					[illegible]
502 RCL 01					
503 *	Az	a	b	b	
504 RCL [	c				
505 RDN	Az	a	b	c	
506 RTN					
507♦LBL "ADJ"					
508 SF 14					
509♦LBL 53					
510 XROM "LL"					
511 "BEG PT"					
512 PROMPT	Beg Pt				
513 STO 01					
514 "END PT"					
515 PROMPT	End Pt	Beg Pt			
516 X<> 00	ΣDist?	Beg Pt			End Pt in R.00; default ΣDIST brought from R.00
517 "Σ DIST?"					
518 PROMPT					
519 1/X	$1/\Sigma d$	Beg Pt			
520 ENTER↑					
521 X<> 01	Beg Pt	$1/\Sigma d$			$1/\Sigma d$ in R.01
522 X<>Y	$1/\Sigma d$	Beg Pt			
523 X<> 00	End Pt	Beg Pt			$1/\Sigma d$ in R.00
524 XROM "XY"	ΔX	ΔY	X_{Beg}	X_{Beg}	
525 ST* 00	ΔX				$\Delta X/\Sigma d$ in R.00
526 X<>Y					
527 ST* 01	ΔY	ΔX			$\Delta Y/\Sigma d$ in R.01
528 R↑					
529 R↑	X_1	Y_1			
530 .	0				
531 SIGN	1	X_1	Y_1		

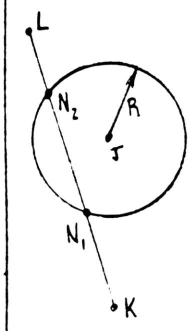
Step	X	Y	Z	T	L
532♦LBL 41					
533 RDN	X_1	Y_1			
534 "PT#"					
535 PROMPT	Pt#				
536 X=0?					
537 GTO 53					
538 LASTX	0				
539 STO]					Σ dist. in 0]
540 RDN	Pt#				
541 XROM "P"	Y_2	X_2	X_1	Y_1	
542 X<> L	2P+1				Y in Last X
543 STO [	2P+1				
544 X<> L	Y_2	X_2	X_1	Y_1	
545 ST- T					
546 X<>Y					
547 ST- Z	X_2	Y_2	ΔX	ΔY	
548 R↑					
549 R↑	ΔX	ΔY	X_2	Y_2	
550 R-P	d_1	θ_1	X_2	Y_2	
551 ST+]					Σ dist accum. so far
552 R↑					
553 R↑	X_2	Y_2			
554 RCL]	Σdist				
555 ENTER↑	Σdist	Σdist	X_2	Y_2	
556 X<> 01	$\frac{err\,Y}{\Sigma D}$				
557 ST* 01	err y/[illegible]				mult. $\frac{err\,Y}{\Sigma D}$ in R.01
558 X<> 01	$\frac{err\,Y \cdot \Sigma d}{\Sigma}$				by the distance accumulated so far, leaving ΔY in R.01
559 ST- IND [	#				
560 CLX					
561 RCL 00	$\frac{err\,X}{\Sigma D}$	Σdistance	X_2	Y_2	
562 *					
563 DSE [					# subtract ΔY from the Y coord. of the Pt.
564 ""					
565 ST- IND [	*				
566 RCL]	Σdistance	X_2	Y_2		* subtract ΔX from the X coord. of the Pt.
567 SIGN					
568 RDN	X_2	Y_2			
569 GTO 41					
570♦LBL "P"	P	a	b	c	
571 SF 21					
572 ST+ X	2P				
573 SIGN	1				2P in Last X
574 RDN	a	b	c		
575 RCL IND L	X	a	b	c	
576 ISG L					2P+1 in Last X
577 ""					
578 RCL IND L	Y	X	a	b	
579 END					

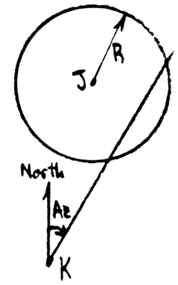
Program ________ For Coordinate Geometry 896 bytes Page 1 of 5

Step	X	Y	Z	T	L
01♦LBL "ARCS"					
02 XROM "L*"					
03♦LBL 04					
04 "J↑R1↑K↑R2"					if $R_2<0$, is taken as a pt. on the circle - radius is found in LL108
05 PROMPT					
06 XEQ 08	R_2	K	R_1?	J	
07 STO [					
08 R↑					
09 R↑	R_1	J	R_2	K	
10 XEQ 08					
11 STO \	R_1				
12 X<> T	K	J	R_2	R_1	
13 XROM "XY"	ΔX	ΔY	X_J	Y_J	
14 R↑					
15 STO 00	Y_J				
16 R↑					
17 STO 01	X_J				
18 R↑					
19 R↑	ΔX	ΔY			
20 R-P	d	θ			
21 STO]	d				
22 RCL \	R_1				
23 R-P	$(d^2-R_1^2)^{1/2}$	garbage	θ		
24 X<>Y					
25 RDN					
26 X↑2	$d^2+R_1^2$	θ			
27 RCL [	R_2				
28 ENTER↑	R_2	R_2	$d^2+R_1^2$	θ	
29 *	R_2^2				$R_2^2=R_1^2+d^2-2R_1 d\cos\phi$
30 -	$(d^2+R_1^2-R_2^2)$	θ	θ	θ	
31 2					$\cos\phi=\frac{R_1^2+d^2-R_2^2}{2R_1 d}$
32 /					
33 RCL]	d				
34 /					
35 RCL \	R_1				
36 /					
37 ACOS	φ	θ	θ	θ	
38 -	θ-φ				
39 X<>Y	θ	θ-φ			
40 LASTX	φ				
41 +	θ+φ	θ-φ			
42 X<> \	R_1				(θ+φ) in R.N
43 P-R	ΔX_1	ΔY_1			
44 X<> 01	X_J				
45 ST+ 01					$\Delta X_1+X_J=X_{N-1}$ in R.01
46 X<>Y	ΔY_1	X_J	θ		
47 X<> 00	Y_J				
48 ST+ 00					$\Delta Y_1+Y_J=Y_{N-1}$ in R.00
49 RCL \	θ+φ				
50 LASTX	R_1	θ+φ	Y_J	X_J	
51 P-R	ΔX_2	ΔY_2	Y_J	X_J	
52 ST+ T				X_{N-2}	



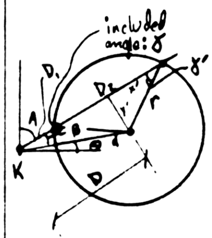
Step	X	Y	Z	T	L
53 RDN	ΔY_2	Y_J	X_{N-2}	ΔX_2	
54 +	Y_{N-2}	X_{N-2}			
55 XROM "PM"					
56 GTO 04					
57♦LBL "APT"					
58 XROM "L*"					
59♦LBL 02					
60 "J↑R↑K↑L"					
61 PROMPT	L	K	R?	J	
62 R↑					
63 R↑					
64 XEQ 08					
65 STO 01	R	J	L	K	
66 X<>Y					
67 STO [	J				
68 R↑					
69 STO 00	K				
70 R↑	L	K	J	R	
71 XROM "XY"	ΔX	ΔY			
72 X<>Y	ΔY	ΔX			
73 R-P	d	Az			
74 X<> [	J	Az			
75 RCL 01	R				
76 RCL 00	K				
77 R↑	Az	K	R	J	
78 XEQ 05					
79 GTO 02					
80♦LBL "AAZ"					
81 XROM "L*"					
82♦LBL 12					
83 "J↑R↑K↑A"					
84 XROM "Pd"	A	K	R?	J	
85 XEQ 11	A	K	R	J	
86 GTO 12					
87♦LBL "ABR"					
88 XROM "L*"					
89♦LBL 07					
90 "J↑R↑K"					
91 XROM "BA"	A	K	R?	J	
92 XEQ 11	A	K	R	J	
93 GTO 07					
94♦LBL 11	A	K	R?	J	
95 R↑					
96 R↑	R?	J	A	K	
97 XEQ 08	R	J	A	K	
98 R↑					
99 R↑	A	K	R	J	





B71

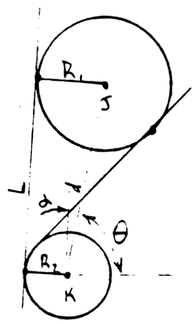
Step	X	Y	Z	T	L
100◆LBL 05	A	K	R	J	
101 STO [	A				
102 RDN	K	R	J	A	
103 X<>Y	R	K	J	A	
104 STO \	R				
105 X<> Z	J	K	R	A	
106 XROM "XY"	ΔX	ΔY	X_K	Y_K	
107 R-P	d	θ			
108 R↑					
109 STO 00	Y_K				
110 R↑					
111 STO 01	X_K				
112 R↑					
113 R↑	d	θ			
114 X<>Y	θ	d			
115 RCL [	A				
116 +	$\theta+A$	d			$\theta+A=90-\beta$
117 X<>Y	d	$90-\beta$			
118 P-R	Y'	D			
119 X↑2	$(Y')^2$				
120 RCL \	R				
121 ENTER↑	R	R	$(Y')^2$	D	
122 *	R^2	Y'^2	D	D	
123 X<>Y	Y'^2	R^2			
124 -					
125 SQRT	X'	D	D	D	
126 -	D_1				
127 RCL [	A	D_1	D	D	
128 R↑	D				
129 LASTX	X'	D	A	D_1	
130 +	D_2	A	D_1		
131 RCL [	A	D_2	A	D_1	
132 R↑	D_1	A	D_2	A	
133 P-R	ΔY_1	ΔX_1			
134 X<> 00	Y_K				
135 ST+ 00					$Y_{N\cdot 1}$ in R.00
136 X<>Y	ΔX_1	Y_K	D_2	A	
137 X<> 01	X_K	Y_K			
138 ST+ 01					$X_{N\cdot 1}$ in R.01
139 R↑	—				
140 R↑	D_2	A			
141 P-R	ΔY_2	ΔX_2	X_K	Y_K	
142 ST+ T	ΔY_2	ΔX_2	X_K	$Y_{N\cdot 2}$	
143 RDN	ΔX_2	X_K	$Y_{N\cdot 2}$	ΔY_2	
144 +	$X_{N\cdot 2}$	$Y_{N\cdot 2}$			
145 X<>Y	$Y_{N\cdot 2}$	$X_{N\cdot 2}$			
146◆LBL "PM"					
147 CF 22					
148 "M"					
149 PROMPT	M	$Y_{N\cdot 2}$	$X_{N\cdot 2}$		
150 FC? 22					
151 .	0				
152 X=0?					
153 GTO 09					



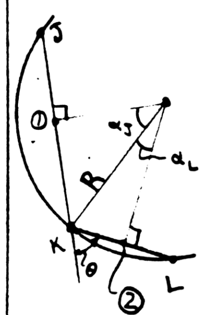
Step	X	Y	Z	T	L
154 XROM "P"	Y_M	X_M	$Y_{N\cdot 2}$	$X_{N\cdot 2}$	
155 R↑					
156 STO [	$X_{N\cdot 2}$				
157 R↑					
158 STO a	$Y_{N\cdot 2}$				
159 R↑	X_M	$Y_{N\cdot 2}$	$X_{N\cdot 2}$	Y_M	
160 ST- Z	X_M	$Y_{N\cdot 2}$	X_N-X_M	Y_M	
161 R↑	Y_M	X_M	$Y_{N\cdot 2}$	ΔX_2	
162 ST- Z	Y_M	X_M	ΔY_2	ΔX_2	
163 R↑					
164 R↑	ΔX_2	ΔY_2	Y_M	X_M	
165 R-P	d_2	garbage			
166 R↑					
167 R↑	Y_M	X_M	d_2		
168 RCL 00	$Y_{N\cdot 1}$				
169 -	ΔY_1				
170 X<>Y	X_M	ΔY_1	d_2	d_2	
171 RCL 01	$X_{N\cdot 1}$				
172 -	ΔX_1	ΔY_1			
173 R-P	d_1	garbage	d_2		
174 X<>Y					
175 X<> [	$X_{N\cdot 2}$	d_1	d_2		
176 RDN	d_1	d_2		$X_{N\cdot 2}$	
177 "N"					
178 X<=Y?					
179 GTO 01					
180 R↑	$X_{N\cdot 2}$				
181 RCL a	$Y_{N\cdot 2}$				
182 X<>Y	$X_{N\cdot 2}$	$Y_{N\cdot 2}$			
183 GTO 10					
184◆LBL 09					
185 X<> Z	$X_{N\cdot 2}$	$Y_{N\cdot 2}$	0		
186 "N1"					
187 XEQ 10					
188 "N2"					
189◆LBL 01					
190 RCL 00					
191 RCL 01	$X_{N\cdot 1}$	$Y_{N\cdot 1}$			
192◆LBL 10					
193 .	0				
194 ENTER↑	0	0	X	Y	
195 XROM "SP"					
196 RTN					
197◆LBL "EXTAN"					
198 SF 09					
199 GTO 00					
200◆LBL "INTAN"					
201 CF 09					

B72

Step	X	Y	Z	T	L
202♦LBL 00					
203 XROM "L*"					
204 "J↑R1↑K↑R2"	R_2	K	R_1	J	
205 PROMPT					
206 XEQ 01					
207 GTO 00					
208♦LBL "PTAN"					
209 XROM "L*"					
210♦LBL 13					
211 "J↑R↑K"					
212 PROMPT	K	R_J	J		
213 .	O	K	R_J	J	
214 XEQ 01					
215 GTO 13					
216♦LBL 01					
217 X<>Y	K	R_K			
218 RDN	R_K	R_J	J	K	
219 X<>Y	R_J	R_K			
220 XEQ 08					
221 STO [	R_J				
222 X<>Y	R_K				
223 XEQ 08					
224 FS? 09					
225 CHS	$\pm R_K$	R_J	J	K	
226 +					
227 STO \	$R_{J\pm K}$				
228 RDN	J	K	K	ΔR	
229 X<>Y	K	J			
230 XROM "XY"	ΔX	ΔY	X_J	Y_J	
231 R↑					
232 STO 00	Y_J				
233 R↑					
234 STO 01	X_J				
235 R↑					
236 R↑	ΔX	ΔY			
237 R-P	d	θ			
238 X↑2	d^2				
239 RCL \	ΔR				
240 ENTER↑	ΔR	ΔR	d^2	θ	
241 *	ΔR^2	d^2	θ	θ	
242 -					
243 SQRT	L	θ			
244 RCL \	ΔR	L	θ		
245 R-P	d	α	θ	θ	
246 RDN	α				
247 -	$\theta-\alpha$	θ			
248 X<>Y	θ	$\theta-\alpha$			
249 LASTX					
250 +	$\theta+\alpha$	$\theta-\alpha$			
251 RCL [	R_J				
252 P-R	ΔX_1	ΔY_1			
253 X<> 01	X_J				
254 ST+ 01					X_1 in R.01



Step	X	Y	Z	T	L
255 X<>Y	ΔY_1				
256 X<> 00	Y_J				
257 ST+ 00					Y_1 in R.00
258 LASTX	R_J	Y_J	X_J	$\theta-\alpha$	
259 R↑					
260 X<>Y	R_J	$\theta-\alpha$			
261 P-R	ΔX_2	ΔY_2	Y_J	X_J	
262 ST+ T				X_2	
263 RDN	ΔY_2	Y_J	X_2	ΔX_2	
264 +	Y_2	X_2			
265 XROM "PM"					
266 RTN					
267♦LBL "C3P"					
268 XROM "L*"					
269♦LBL 24					
270 "J↑K↑L"					
271 PROMPT	L	K	J		
272 STO 00	L				
273 RDN					
274 STO 01	K	J		L	
275 XROM "XY"	ΔX	ΔY	X_J	Y_J	
276 STO \	ΔX				
277 RDN	ΔY	X_J	Y_J		
278 2					
279 /	$\Delta Y/2$	X_J	Y_J		
280 ST+ Z	$\Delta Y/2$	X_J	$Y_{(1)}$		$Y_{(1)} = Y_J + \frac{\Delta Y}{2}$
281 X<> \	ΔX				
282 LASTX	2	ΔX	X_J	$Y_{(1)}$	
283 /					
284 +	$X_{(1)}$				
285 RCL \	$\Delta Y/2$	$X_{(1)}$	$Y_{(1)}$		
286 LASTX	$\Delta X/2$				
287 CHS	$-\Delta X/2$				
288 R-P	$d_{J-K/2}$	Az_{J-K}			
289 RDN	A_{J-K}				
290 STO \	Az				
291 RDN	$X_J+\frac{\Delta X}{2}$				
292 X<> 00	L				$X_{(1)}$ in R.00
293 STO]	L				
294 X<>Y	$Y_J+\frac{\Delta Y}{2}$				
295 X<> 01	K	L			$Y_{(1)}$ in R.01
296 XROM "XY"					
297 STO [	ΔX	ΔY	X_L	Y_L	
298 RDN	ΔY				
299 2					
300 /	$\Delta Y/2$	X_L	Y_L		
301 STO a					$\Delta Y/2$ in R.a
302 ST+ Z			$Y_{(2)}$		
303 RDN					
304 RCL [					
305 LASTX					
306 /					
307 STO [	$\Delta X/2$	X_L			
308 +	$X_{(2)}$	$Y_{(2)}$			

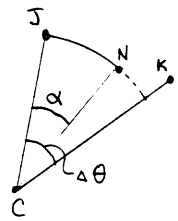


Program ________ For Coordinate Geometry 896 bytes Page 4 of 5

Step	X	Y	Z	T	L
309 RCL 00	$X_{(1)}$	$X_{(2)}$	$Y_{(2)}$		
310 -	ΔX				
311 X<>Y	$Y_{(2)}$				
312 RCL 01	$Y_{(1)}$				
313 -	ΔY				
314 X<>Y	ΔX	ΔY			
315 R-P	d_{1-2}	γ_{1-2}			
316 X<>Y	γ_{1-2}	d_{1-2}			
317 RCL a	$\Delta Y/2$				
318 RCL [	$\Delta X/2$				
319 CHS	$-\Delta X/2$	$\Delta Y/2$	γ_{1-2}	d_{1-2}	
320 R-P	d_{X-L}	γ_{XL}			
321 RDN	γ_{XL}	γ_{1-2}			
322 STO [	γ_{XL}	γ_{1-2}			
323 +					
324 X<>Y	d_{1-2}				
325 P-R					
326 RCL \					
327 STO Z					
328 RCL [					
329 -					
330 SIN					
331 /					
332 P-R					
333 RCL 01					
334 RCL 00					
335 ST+ T					
336 RDN					
337 +					
338 RCL]					
339 XROM "P"					
340 R↑					
341 ST- Z					
342 R↑					
343 ST- Z					
344 R↑					
345 R↑					
346 R-P					
347 "R="					
348 ARCL X					
349 "├'"					
350 AVIEW					
351 CLX					
352 RDN					
353 CLX					
354 RDN					
355 "RAD PT"					
356 XROM "SP"					
357 GTO 24					
358♦LBL "ATRY"					
359 XROM "L*"					
360 SF 09					
361 GTO 14					

Step	X	Y	Z	T	L
362♦LBL "ATVA"					
363 XROM "L*"					
364 CF 09					
365♦LBL 14					
366 "J↑C↑A"					
367 FS? 09					
368 "├RC L"					
369 FS? 09					
370 PROMPT	Arc L	C	J		
371 FC? 09					
372 XROM "Pd"	A	C	J		Angle or Arc L in R.00
373 STO 00					
374 RDN	C	J			
375 X<>Y	J	C			
376 XROM "XY"	ΔX	ΔY	X_c	Y_c	
377 R-P	d	θ			
378 R↑	Y_c	d	θ	X_c	Y_c in R.00
379 X<> 00	A or L				
380 FC? 09					
381 GTO 06					
382 X<>Y	d	Arc L			
383 /	Arc/Rad				
384 LASTX	d				
385 X<>Y	Arc/Rad	d			
386 R-D	Angle				
387♦LBL 06	Angle	d	θ	X_c	
388 X<>Y	d	α			
389 RDN	α	θ	X_c	d	
390 -	$\theta-\alpha$				
391 R↑	d	$\theta-\alpha$			
392 P-R	ΔX	ΔY			
393 X<>Y	ΔY	ΔX	X_c		
394 RCL 00	Y_c				
395 RDN	ΔY	ΔX	X_c	Y_c	
396 "N"					
397 XROM "SP"					
398 GTO 14					

Step	X	Y	Z	T	L	
399♦LBL "AFRC"						
400 XROM "L*"						
401♦LBL 19						
402 "J↑C↑K↑FRC"						
403 PROMPT	FRC	K	C	J		
404 STO [	FRC					
405 RDN						
406 STO 01	K					
407 RDN						
408 STO 00	C					
409 X<>Y	J	C				
410 XROM "XY"	ΔX	ΔY	X_c	Y_c		
411 R-P	d	θ				
412 X<> 00	C	θ				Rad in R.00
413 X<>Y	θ	C				
414 X<> 01	K	C				θ_J in R.01
415 XROM "XY"	ΔX	ΔY	X_c	Y_c		
416 R-P	d	θ_K				
417 RDN	θ_K					
418 RCL 01	θ_J	θ_K	X_c	Y_c		
419 -	$\Delta\theta$					
420 RCL [	Frc					
421 *	α					
422 RCL 01	θ_J					
423 +	θ_N					
424 RCL 00	Rad	θ_N				
425 P-R	ΔX	ΔY				
426 X<>Y	ΔY	ΔX	X_c	Y_c		
427 "N"						
428 XROM "SP"						
429 GTO 19						
430♦LBL "FREE"						
431 XROM "L*"						
432 SF 21						
433 "FREE PTS:"						
434 AVIEW						
435 .4						
436 STO 00						
437 E9						
438 CHS						
439 STO 01						
440♦LBL 03						
441 ISG 00						
442 RCL 00	Pt#					
443 ST+ X	reg#					



Step	X	Y	Z	T	L
444 SF 25					
445 RCL IND X	X_i				
446 FC? 25					
447 XROM "LL"					
448 FC?C 25					
449 STOP					
450 RCL 01	-E9	X_i			
451 X≠Y?					
452 GTO 03					
453 RCL 00	Pt#				
454 " "					
455 XROM "AX"					
456 AVIEW					
457 GTO 03					
458♦LBL 08					
459 X≠0?					
460 X>0?					
461 RTN	-Pt#	K	a	b	
462 R↑					
463 STO 00	b				
464 R↑					
465 STO 01	a				
466 R↑					
467 STO]	K				
468 R↑					
469 CHS	Pt#	K			
470 XROM "XY"	ΔX	ΔY	X_K	Y_K	
471 R-P	Rad.	θ			
472 RCL 00	b				
473 RCL 01	a				
474 RCL]	K				
475 R↑	Rad	K	a	b	
476 END					

Program AREA For Area by coordinates 174 bytes Page 1 of 1

Step	X	Y	Z	T	L
01♦LBL "AREA"					
02 SF 14					
03♦LBL 08					
04 XEQ "L"					
05 SF 21					
06 SF 09					
07 .					
08 STO 01	ΣArea				
09 "PT1"					
10 PROMPT	Pt#1				
11 E3					
12 /	.00P				
13 "PT2"					
14 PROMPT	Pt 2				
15 +	Pt2.00P1				
16 ST+ X	2P2.00P1				
17 E	1				
18 +	1+2P2.00P1				
19 STO 00					
20 3					
21 STO [	counter				
22 RDN	1+2P2.00	0			
23♦LBL 05					
24 SIGN	1				1+2P2.002P1 in Last X
25 RDN					
26 RCL [	counter				
27 "PT"					
28 XROM "AX"					
29 PROMPT	Pt#				
30 CF 10					
31 X<>Y	counter				
32 ISG X	counter+1				
33 ""	"NOP"				
34 STO [	new counter				
35 RDN	Pt#3				
36 ST+ X	2×Pt#3				
37 LASTX	1+2P2				
38 X<>Y	2×P3	$1+2P_2$			
39 CF 09					
40 X>0?					
41 XEQ 09	$1+2P_n$				
42 X>0?					
43 GTO 05					
44 RDN	$1+2P_{final}$				
45 RCL 00	$1+2P_2.002P_1$				
46 FRC					
47 E3	1000				
48 *	$2P_1$	$1+2P_{final}$			
49 XEQ 09	$1+2P_1$				
50 RCL 00	$1+2P_2$				
51 E	1				
52 -	$2P_2$	$1+2P_1$			
53 XEQ 09					
54 RCL 01	ΣArea2				
55 2	2				
56 /	ΣArea				
57 "AREA="					
58 ARCL X					
59 XEQ "FS"					
60 AVIEW					
61 GTO 08					
62♦LBL C					
63 FS? 09 (Still within first two points? Start over)					
64 GTO 08					
65 LASTX	$1+2P_{n-1}$				
66 X<>Y					
67 FS? 10					
68 RDN					
69 FS? 10					
70 GTO 05					
71 SF 10					
72 R↑	$\Delta X \cdot Y_{n-1}$				
73 ST- 01	(ΣArea)				
74 RDN	$1+2P_{n-1}$				
75 E	1				
76 -	$2P_{n-1}$				
77 STO [	counter				
78 R↑					
79 GTO 05					
80♦LBL 09	$2P_n$	$1+2P_{n-1}$			
81 X<>Y	$1+2P_{n-1}$				
82 FRC	$.002P_{n-2}$				
83 LASTX	$1+2P_{n-1}$				
84 X<>Y	$.002P_{n-2}$	$1+2P_{n-1}$	$2P_n$		
85 E3	1000				
86 *	$2P_{n-2}$	$1+2P_{n-1}$	$2P_n$	$2P_n$	
87 X<> Z	$2P_n$	$1+2P_{n-1}$	$2P_{n-2}$		
88 RCL IND X	X_n			$2P_{n-2}$	
89 RCL IND T	X_{n-2}	X_n	$2P_n$	$1+2P_{n-1}$	
90 -	ΔX	$2P_n$	$1+2P_{n-1}$		
91 RCL IND Z	Y_{n-1}				
92 *	$\Delta X \cdot Y$	$2P_n$	$1+2P_{n-1}$		
93 ST+ 01	(ΣA)				
94 SIGN	1				$\Delta X \cdot Y_{n-1}$ in Last X
95 RDN	$2P_n$	$1+2P_{n-1}$			
96 X<>Y	$1+2P_{n-1}$	$2P_n$			
97 E-3	.001	$1+2P_{n-1}$	$2P_n$		
98 ST* Y		$\frac{1+2P_{n-1}}{1000}$			
99 ST- Y		$.002P_{n-1}$	$2P_n$		
100 X<> L	$\Delta X \cdot Y_{n-1}$				
101 RDN	$.002P_{n-1}$	$2P_n$		$\Delta X \cdot Y_{n-1}$	
102 +					
103 E	1			•	
104 +	$1+2P_n.002P_{n-1}$		•		
105 END					

B76

STATIONING of POINTS on a LINE

xeq POB (SIZE 2n + 2 for "n" points)

SOFTKEYS (USER mode ON)

No softkeys are used by POB

USER FLAGS

Flag 00	clear	Station and offset displayed
	set	Only station displayed
Flag 03	clear	Azimuth input in decimal degrees
	set	Azimuth input in degrees-minutes-seconds

INPUT SUMMARY

Prompt	Input	Default
POB	Point of beginning. Defined as being at station 0+00.	No default
PT ON CL	A second point on the line, used to define the line. Bypass to enter an azimuth from the POB.	No default
AZ	Azimuth of the line from the POB, in decimal degrees (dd) or degrees-minutes-seconds (DMS)	No default
PT	Point for which station and offset are desired	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

STA	The station of the point input, in standard "sta + feet" notation.
(OS)	Offset of the point from the centerline, in feet. A negative offset is to the right of the line looking forward on station.

PROGRAM FLAGS

Flag 14	set	Line printing control
Flag 21	set	Printer/aview control

STORAGE REGISTER USE

Register	Value
00	POB
01	AZ
02	Point #1 Northing (Y)
03	Point #1 Easting (X)
04	Point #2 Northing (Y)
05	Point #2 Easting (X)

etc.

STATION and OFFSET of POINTS ON A LINE

Program POB finds the station and offset of coordinate points along a (straight) centerline. These points must be defined by other coordinate geometry routines in the ROM, or by user routines. A point of beginning is used, defined as station 0+00. The line is defined by either another point on the line, or the azimuth of the line. The normal procedure is that points are being created in the coordinate geometry routines which have a known relationship to a centerline. If one of these points (which must be on the centerline) has a known station, the POB is set by projecting backward from that point along the line, a distance which is that point's station in feet. Then all other points can be stationed from this POB.

If flag 00 is set, only the station is output; if flag 00 is clear, both station and offset are given. Stationing is in the standard "sta + feet" notation. The offset is in feet, a positive offset being to the left of centerline looking forward on station (i.e. the POB at your back).

SUBROUTINE USE

Parts of POB are useable as subroutines for other programs. The calling program needs to SF 11, xeq POB with either 2 or 3 in the X-register (to access label 02 or 03 in POB; see the listing). The required input (in the Y-register) is the point number for which stationing is desired (Lbl 02) or the station to be appended to Alpha (Lbl 03). The listing for POB is part of program SCOL.

B79

B60

TRIANGLE SOLUTIONS

(SIZE 002)

SOFTKEYS (USER mode ON)

A	SSS	Three sides known
B	ASA	Two angles and included side known
C	SAA	Two angles and non-included side known
D	SAS	Two sides and included angle known
E	SSA	Two sides and non-included angle known

USER FLAGS

Flag 00	clear	Full output
	set	Only area displayed
Flag 03	clear	Angles in decimal degrees (dd)
	set	Angles in Degrees-Minutes-Seconds (DMS)

INPUT SUMMARY

Prompt		Input	Default
S1/S2/S3	(SSS)	Three sides	None
A1/S2/A3	(ASA)	Two angles and included side	None
S2/A1/A2	(SAA)	Two angles and non-included side	None
S1/A3/S2	(SAS)	Two sides and included angle	None
S1/S2/A1	(SSA)	Two sides and non-included angle	None

B81

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(A1, A2, A3)	Angles, in decimal degrees or DMS
(S1, S2, S3)	Length of sides
AREA	Area of the enclosed triangle

PROGRAM FLAGS

Flag 21	set	Display control

STORAGE REGISTER USE

Register	Value
00	Working; S1
01	Working; S2
02	A3
03	S3
04	A2

B42

TRIANGLE SOLUTIONS

There are five programs concerned with solving triangles. Starting with different information, they all calculate and display the internal angles, lengths of the sides, and the area of the solved triangle. The standard triangle notation is used; e.g. side #1 is opposite angle #1.

All of the routines are accessible <u>either by softkey or by alpha label</u> (xeq ___). These softkeys operate a little differently than most softkeys in this module. Instead of bringing up a prompt for the required input, they assume the input is already in the stack; pressing the softkey starts execution of the desired routine. The triangle solution routines, and their softkeys, are:

Key	Label	Description
A	SSS	All three sides known
B	ASA	Two angles and included side known
C	SAA	Two angles and non-included side known
D	SAS	Two sides and included angle known
E	SSA	Two sides and non-included angle known

Each of the routines requires three items of input, keyed into the stack: Item one, enter↑, Item two, enter↑, Item three, softkey or R/S. If using softkeys, simply key the three items in and press the softkey (making sure USER mode is on). If prompts are desired, execute the desired label from the keyboard, key in the three input items in response to the prompt, and press R/S.

Like any other HP-41 program, in order for the softkeys to be active, the program pointer must be within the program that the local alpha (softkey) labels are in. The easiest way to do this is an alpha GTO; just GTO <u>any</u> of the six global labels listed above. Thereafter, the softkeys will be active.

Routine SSA usually has not one but two possible solutions; both are given. See the examples for clarification.

"SIZE CHECK"

None of the triangle programs use subroutine S. This means that the user must verify that the SIZE is at least 005, and that DEG mode is set.

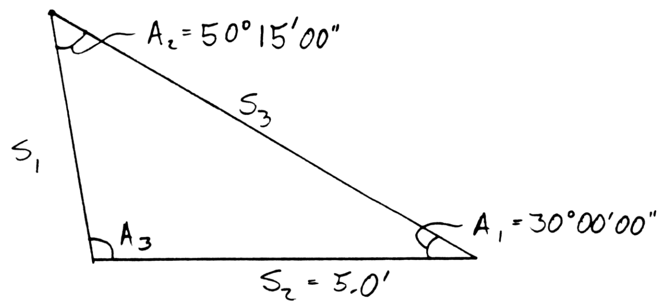
FLAGS

Like all the geometry programs in this module, the triangle routines recognize the status of flag 03. If it is clear, all angular input and output is in decimal degrees (dd); if set, degrees-minutes-seconds (DMS) is used. The output displays and input prompts are clearly labeled to reflect this. However, if softkeys are used for input, the user must verify that flag 03 matches the desired format.

Flag 00 is used by all of the triangle programs; output of the angles and sides is skipped if flag 00 is set, and only the area of the enclosed triangle is displayed.

B64

Use "XEQ":




```
            XEQ "SAA"
S2↑A1↑A2 (DMS)
      5.0000 ENTER↑
     30.0000 ENTER↑
     50.1500    RUN
A1=30.0000 DMS
A2=50.1500 DMS
A3=99.4500 DMS

S1=3.2516'
S2=5.0000'
S3=6.4094'

AREA=8.0117 SQ'
---------
S2↑A1↑A2 (DMS)
```

(Here, flag 03 was set; angles are input and output in Degrees-Minutes-Seconds, as the prompt and results indicate.)

or, **Use Local labels** (In User Mode)

(same triangle - user must check status of flag 03 to match input format)

```
            GTO "SSS"
      5.0000 ENTER↑
     30.0000 ENTER↑
     50.1500  XEQ C
A1=30.0000 DMS
A2=50.1500 DMS
A3=99.4500 DMS

S1=3.2516'
S2=5.0000'
S3=6.4094'

AREA=8.0117 SQ'
---------
```

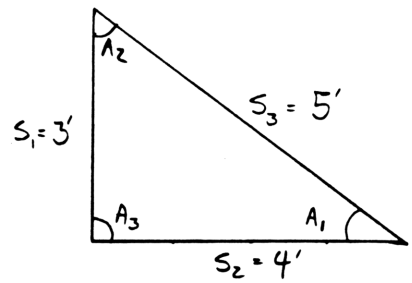
Note: To use key-accessed local labels, three things are required:

1) Calculator in User mode.

2) No key assignments on the key in question.

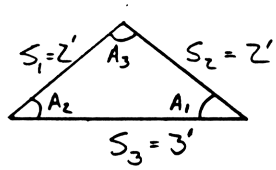
3) The calculator is "in" the **program**; that is, the program pointer is somewhere between the first line of the program and the end. Looking at the program listing, all of the routines (SSS, ASA, etc.) have a common "end" statement. Therefore the easiest way to access a local label is to go to (GTO, not XEQ) <u>any</u> of the global labels within the program (here, "SSS") and then use the local (key-accessed) labels. This only needs to be done once, at first, and then after going to another global label, or PACKing.

Use "XEQ":



(Here, flag 03 was clear; angles are in decimal degrees)

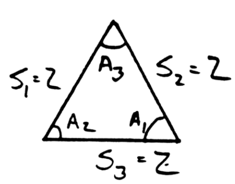
```
            XEQ "SSS"
S1↑S2↑S3
        3.0000 ENTER↑
        4.0000 ENTER↑
        5.0000    RUN
A1=36.8699 dd
A2=53.1301 dd
A3=90.0000 dd

S1=3.0000'
S2=4.0000'
S3=5.0000'

AREA=6.0000 SQ'
---------
S1↑S2↑S3
        2.0000 ENTER↑
        2.0000 ENTER↑
        3.0000    RUN
A1=41.4096 dd
A2=41.4096 dd
A3=97.1808 dd

S1=2.0000'
S2=2.0000'
S3=3.0000'

AREA=1.9843 SQ'
---------
S1↑S2↑S3
```

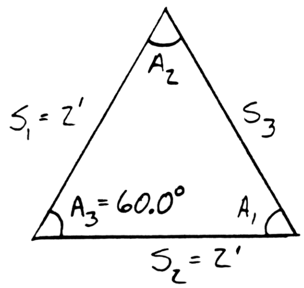


or Use local label (In User mode):

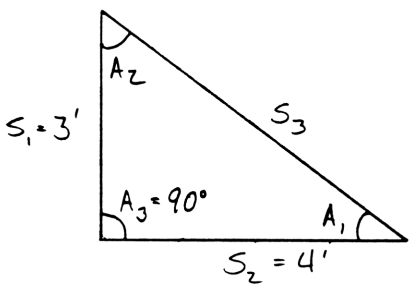
(Here, flag 03 was set; angles are in Degrees-Minutes-Seconds)

```
        2.0000 ENTER↑
        2.0000 ENTER↑
        2.0000  XEQ A
A1=60.0000 DMS
A2=60.0000 DMS
A3=60.0000 DMS

S1=2.0000'
S2=2.0000'
S3=2.0000'

AREA=1.7321 SQ'
---------
```





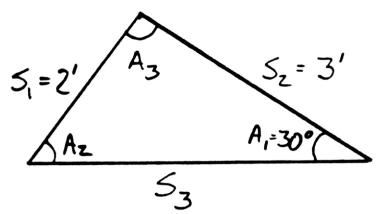
(Here, flag 03 was clear. Angles are input and output in decimal degrees.)



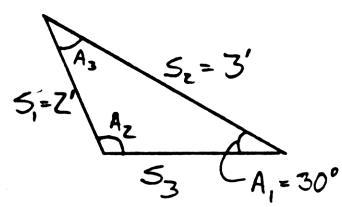

```
                    XEQ "SAS"
S1↑A3↑S2 (dd)
              2.0000 ENTER↑
             60.0000 ENTER↑
              2.0000    RUN
A1=60.0000 dd
A2=60.0000 dd
A3=60.0000 dd

S1=2.0000'
S2=2.0000'
S3=2.0000'

AREA=1.7321 SQ'
---------
S1↑A3↑S2 (dd)
              3.0000 ENTER↑
             90.0000 ENTER↑
              4.0000    RUN
A1=36.8699 dd
A2=53.1301 dd
A3=90.0000 dd

S1=3.0000'
S2=4.0000'
S3=5.0000'

AREA=6.0000 SQ'
---------
S1↑A3↑S2 (dd)
```

"Side - Side - Angle" definitions have two possible solutions. Both are given.

Solution 1:

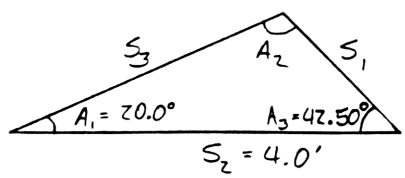


Solution 2:




```
                    XEQ "SSA"
S1↑S2↑A1 (DMS)
          2.0000 ENTER↑
          3.0000 ENTER↑
         30.0000    RUN
A1=30.0000 DMS
A2=48.3525 DMS
A3=101.2435 DMS

S1=2.0000'
S2=3.0000'
S3=3.9210'

AREA=2.9407 SQ'
---------
A1=30.0000 DMS
A2=131.2435 DMS
A3=18.3525 DMS

S1=2.0000'
S2=3.0000'
S3=1.2752'

AREA=0.9564 SQ'
---------
S1↑S2↑A1 (DMS)
```

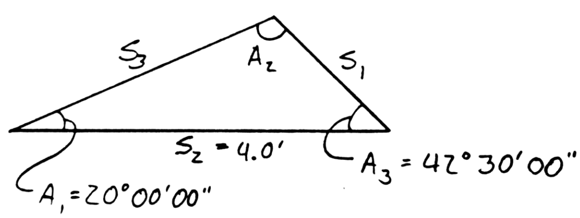
B66



(Flag 03 clear → decimal degrees)

```
                XEQ "ASA"
A1↑S2↑A3 (dd)
          20.0000 ENTER↑
           4.0000 ENTER↑
          42.5000    RUN
A1=20.0000 dd
A2=117.5000 dd
A3=42.5000 dd

S1=1.5423'
S2=4.0000'
S3=3.0466'

AREA=2.0840 SQ'
---------
A1↑S2↑A3 (dd)
```

(Flag 03 set → Degrees-Minutes-Seconds)




```
                    SF 03
                XEQ "ASA"
A1↑S2↑A3 (DMS)
          20.0000 ENTER↑
           4.0000 ENTER↑
          42.3000    RUN
A1=20.0000 DMS
A2=117.3000 DMS
A3=42.3000 DMS

S1=1.5423'
S2=4.0000'
S3=3.0465'

AREA=2.0840 SQ'
---------
A1↑S2↑A3 (DMS)
```

Program SSS, etc. For Triangle Solutions 476 bytes

Step	X	Y	Z	T	L
01♦LBL "SSS"					
02♦LBL 04					
03 "S1↑S2↑S3"					
04 PROMPT	S_3	S_2	S_1		
05♦LBL A					
06 STO 03	S_3				
07 X<>Y	S_2	S_3	S_1		
08 STO 01					
09 +	S_2+S_3	S_1			
10 X<>Y					
11 STO 00	S_1				
12 +	ΣS				
13 2					
14 /	Perim/2				
15 RCL 01	S_2				
16 RCL 00	S_1				
17 RCL 03	S_3	S_1	S_2	P/2	
18 XEQ 14	A_2				
19 STO 04					
20 RDN	P/2				
21 RCL 03	S_3				
22 RCL 00	S_1				
23 RCL 01	S_2	S_1	S_3	P/2	
24 XEQ 14					
25 STO 02	A_3				
26 RCL 04	A_2				
27 XEQ 13	A_1				
28 XEQ 01					
29 GTO 04					
30♦LBL 14					
31 *	$S_3 \cdot S_1$	S_2	P/2	P/2	
32 RDN	S_2	P/2	P/2	S_3S_1	
33 -					
34 X<>Y	P/2				
35 *	P/2(P/2−S_2)	S_3S_1	S_3S_1	S_3S_1	
36 LASTX	P/2	P/2( )	S_3S_1		
37 X<> Z	S_3S_1	P/2( )	P/2		
38 /					
39 SQRT					
40 ACOS					
41 ST+ X	A_2	P/2			
42 RTN					
43♦LBL "ASA"					
44♦LBL 05					
45 "A1↑S2↑A3"					
46 XEQ 02	A_3	S_2	A_1		
47♦LBL B					
48 FS? 03					
49 HR					
50 STO 02	A_3				
51 RCL Z	A_1				
52 FS? 03					
53 HR	A_1	A_3	S_2	A_1	

Step	X	Y	Z	T	L
54 XEQ 12					
55 GTO 05					
56♦LBL 12					
57 STO T	A_1	A_3	S_2	A_1	
58 XEQ 13	A_2	S_2	A_1	A_1	
59 STO 04	A_2				
60 SIN	sin A_2				
61 X<>Y	S_2				
62 STO 01	S_2				
63 X<>Y					
64 /	S_2/sinA_2	A_1	A_1	A_1	
65 RCL 02	A_3				
66 SIN	sin A_3				
67 *	S_3				
68 STO 03					
69 LASTX					
70 /	S_2/sinA_2				
71 X<>Y	A_1	S_2/sinA_2	A_1	A_1	
72 SIN	sin A_1				
73 *	S_1				
74 STO 00					
75 X<>Y	A_1	S_1	A_1	A_1	
76♦LBL 01	A_1				
77 RCL 04	A_2				
78 RCL 02	A_3	A_2	A_1		
79♦LBL 11					
80 SF 21					
81 FS? 00					
82 GTO 00					
83 "A1"					
84 R↑					
85 R↑	A_1	S_2	A_3	A_2	
86 XROM "Dd"					
87 AVIEW					
88 "A2"					
89 R↑	A_2	A_1	S_2	A_3	
90 XROM "Dd"					
91 AVIEW					
92 "A3"					
93 R↑	A_3	A_2	A_1	S_3	
94 XROM "Dd"					
95 AVIEW					
96 ADV					
97 "S1="					
98 ARCL 00					
99 AVIEW					
100 "S2="					
101 ARCL 01					
102 AVIEW					
103 "S3="					
104 ARCL 03					
105 AVIEW					
106 ADV					

B90

Program ________ For Triangle Solutions ______ bytes Page 2 of 3

Step	X	Y	Z	T	L
107♦LBL 00	A_3	A_2	A_1		
108 SIN					
109 RCL 00	S_1	$\sin A_3$	A_2	A_1	
110 *		A_2			
111 RCL 01	S_2		A_2		
112 *		A_2			
113 2			A_2		
114 /	$\frac{S_1S_2\sin A_3}{2}$	A_2	A_1	A_1	
115 ABS					
116 "AREA="					
117 ARCL X					
118 XEQ "FS"					
119 AVIEW					
120 XROM "L"					
121 RTN					
122♦LBL "SAA"					
123♦LBL 07					
124 "S2↑A1↑A2"					
125 XEQ 02	A_2	A_1	S_2		
126♦LBL C					
127 FS? 03					
128 HR					
129 STO 04	A_2				
130 X<>Y					
131 FS? 03					
132 HR	A_1	A_2	S_2		
133 STO 00					
134 XEQ 13	A_3				
135 STO 02					
136 RCL 00	A_1	S_2			
137 XEQ 12					
138 GTO 07					
139♦LBL "SAS"					
140♦LBL 08					
141 "S1↑A3↑S2"					
142 XEQ 02	S_2	A_3	S_1		
143♦LBL D					
144 STO 01	S_2				
145 X<>Y	A_3				
146 FS? 03					
147 HR					
148 STO 02	A_3	S_2	S_1		
149 COS	$\cos A_3$				
150 X<>Y	S_2				
151 *					
152 LASTX	S_2	$S_2\cos A_3$	S_1		
153 X↑2					
154 RDN	$S_2\cos A_3$	S_1		S_2^2	
155 X<>Y					
156 *					
157 LASTX	S_1	$S_1S_2\cos A_3$		S_2^2	
158 STO 00					

Step	X	Y	Z	T	L	
159 X↑2	S_1^2					
160 X<>Y						
161 ST+ X	$2S_1S_2\cos A_3$					$S_3^2 = S_2^2 + S_1^2$
162 -						$-2S_1S_2\cos A_3$
163 R↑	S_2^2					
164 +	S_3^2					
165 SQRT	S_3					
166 STO 03						
167 RCL 02	A_3					
168 SIN	$\sin A_3$	S_3				
169 X<>Y						
170 /	$\frac{\sin A_3}{S_3}$					
171 STO Y						
172 RCL 01	S_2	$\frac{\sin A_3}{S_3}$	$\frac{\sin A_3}{S_3}$			
173 *	$\sin A_2$					
174 ASIN	A_2					
175 STO 04						
176 X<>Y	$\frac{\sin A_3}{S_3}$	A_2				
177 RCL 00	S_1					
178 *						
179 ASIN	A_1					
180 XEQ 01						
181 GTO 08						
182♦LBL "SSA"						
183♦LBL 09						
184 "S1↑S2↑A1"						
185 XEQ 02						
186♦LBL E						
187 FS? 03						
188 HR	A_1	S_2	S_1			
189 STO T						
190 SIN	$\sin A_1$	S_2	S_1	A_1		
191 X<>Y						
192 STO 01	S_2					
193 *	$S_2\sin A_1$	S_1	A_1	A_1		
194 X<>Y						
195 STO 00	S_1					
196 /	$\frac{S_2\sin A_1}{S_1}$					$\sin A_2$
197 ASIN	A_2					$= \sin A_1 \cdot \frac{S_2}{S_1}$
198 STO 04						
199 X<>Y	A_1	A_2				
200 XEQ 13	A_3					
201 STO 02						
202 SIN	$\sin A_3$	A_1				
203 RCL 00	S_1					
204 *	$S_1\sin A_3$					
205 X<>Y	A_1					
206 SIN	$\sin A_1$					
207 /	S_3					
208 STO 03						
209 RDN	A_1					
210 XEQ 01	area	A_2	A_1	A_1		
211 CLX	0	A_2				
212 XEQ 13	$180 - A_2$					
213 STO 04	A_2'	A_1				$A_2' = 180° - A_2$

Program __________ For Triangle Solutions; Prompt/Display Angles ________ bytes Page 3 of 3

Step	X	Y	Z	T	L		X	Y	Z	T	L
214 X<>Y	A_1	A_2'	A_1	A_1							
215 XEQ 13	A_3	A_1	A_1	A_1							
216 STO 02											
217 SIN	$\sin A_3$										
218 X<>Y	A_1										
219 SIN											
220 /	$\frac{\sin A_3}{\sin A_1}$										
221 RCL 00	S_1										
222 *	S_3										
223 STO 03											
224 X<>Y	A_1										
225 XEQ 01											
226 GTO 09											
227♦LBL 13	A_1	A_3									
228 +	A_1+A_3										
229 COS	cos										
230 CHS	−cos										
231 ACOS	A_2					$\cos(180-\alpha) = -\cos\alpha$					
232 RTN											
233♦LBL "Dd"											
234 "⊢="											
235 FS? 03											
236 GTO 03											
237 ARCL X											
238 "⊢ dd"											
239 RTN											
240♦LBL 03											
241 HMS											
242 ARCL X											
243 HR											
244 "⊢ DMS"											
245 RTN											
246♦LBL "Pd"											
247 XEQ 02											
248 FS? 03											
249 HR											
250 RTN											
251♦LBL 02											
252 FS? 03											
253 "⊢ (DMS)"											
254 FC? 03											
255 "⊢ (dd)"											
256 PROMPT											
257 END											

B92

DIMENSION and TRIGONOMETRY ROUTINES

This module contains several short dimension "functions" which work with feet-inches-sixteenths addition, subtraction, and conversions to and from decimal feet. It also contains trigonometry "functions" which take or give angles in either degrees-minutes-seconds (DMS) or decimal degrees (dd) depending on whether flag 03 is set or clear; and useful utility routines.

The "functions" mimic native HP-41 functions as closely as possible. They leave the operand in Last X and are thus reversible. They use no numbered storage registers (i.e. they work with SIZE 00). Binary functions like PR and all unary functions leave the rest of the stack completely undisturbed, while the two binary functions FIS+ and FIS- effect stack rolldown. They are most useful from the keyboard when assigned to a key and executed in USER mode; they may also be executed from programs.

Dimensions

These routines work with dimensions in feet-inches-sixteenths, and are intended to be used with the display mode set to FIX 4. The form used is very similar to that used by the HP-41 function HMS. For example, 12' 2-7/16" is 12.0207. All fractional inches must be in sixteenths. Thus 6' 5-3/4" is 6.0512. Trailing zeroes need not be keyed in; 3'-10 may be keyed in as 3.1 instead of 3.1000.

FIS — Converts decimal feet to feet-inches-sixteenths. Once executed, pressing R/S once resets the display to FIX 6 (for example, to see the next digit in order to round to the nearest eighth of an inch). Pressing R/S (or executing any of the dimension routines) again resets FIX 4 format.

FTd — Converts feet-inches-sixteenths to decimal feet. Does not have the FIX 6/FIX 4 feature like FIS.

The functions FIS+ and FIS- are for adding and subtracting dimensions directly instead of converting them to decimal feet, adding, and then converting back. Their only drawback is that they are rather slow. It is helpful to realize that when adding dimensions, if there is no carryover from sixteenths to inches or inches to feet, the numbers may be added (or subtracted) directly instead of using FIS+ and FIS-.

B93

Another way to add dimensions directly is to wait until both sixteenths and inches require a carryover, then add 0.8884 to the sum. This effects a carryover automatically. If all this is confusing, just use these two functions. They work far easier if assigned to a key.

FIS+	Adds dimensions in feet-inches-sixteenths. Key in first dimension, enter, key in second dimension, xeq FIS+ (or assign to a key). Stack is rolled down just as for normal addition. Does not have the FIX 6/FIX 4 feature like FIS.
FIS-	Subtracts dimensions in feet-inches-sixteenths.

Trigonometry Functions

These functions accept and output angles in degrees-minutes-seconds if flag 03 is set, and in decimal degrees if flag 03 is clear. Thus if flag 03 is clear they are identical to their namesake HP-41 functions (except a little slower). This saves converting angles to decimal degrees to operate on them. All the functions save their operand in Last X and are thus reversible. All of them require the calculator to be set to DEG (degree) mode. Subroutine return depth is limited to two.

CO	Cosine
SI	Sine
TA	Tangent
ACO	Arc cosine
ASI	Arc sine
ATA	Arc tangent
RP	Rectangular to polar conversion
PR	Polar to rectangular conversion

SHORT UTILITY ROUTINES

These routines find use in daily manual calculations. They all use the full stack.

QUAD	Quadratic equation solution. The program prompts for A, B, and C, the constants in the quadratic equation $Ax^2 + Bx + C = 0$. It then solves (in subroutine Q) for the two possible values of x, where $x = (-B + \sqrt{B^2 - 4AC})/2A$. If $B^2 - 4AC$ is negative, the calculator flashes DATA ERROR and quits.

IP Linear interpolation or extrapolation. IP solves for Y2, where

X1→ Y1
X2← Y2
and X3→ Y3.

The program prompts for X1, Y1, X3, and Y3, then repeatedly prompts for X2 and solves for Y2. The prompts are X1↑ Y1, X3↑ Y3, and X2; the upward arrow in the first two means "enter↑ ." Thus, key in X1, enter, key in Y1, press R/S. Repeat for X3 and Y3. Then key in X2 and press R/S to get Y2 as many times as you want. The program will continue to solve for Y2 until the X2 prompt is bypassed. The routine has a default value of X1 = 0; if X1 is zero, just key in Y1 and press R/S. The user should note that the entire stack is used; <u>any intermediate calculations during input will cause an error.</u>

PYR This is an extremely simple routine that does part of the work in calculating the volume of a truncated pyramid. It uses the approximate formula

$$V = (A_1 + A_2 + \sqrt{A_1 \times A_2}) \times H/3$$

PYR doesn't actually calculate the volume. It takes as input the dimensions of the top and bottom, <u>assumed to be parallel rectangles.</u> The output, multiplied by the height H, is the volume (or a close approximation). See the examples.

B95

Convert to feet:

$10'-9\,3/16 = 10' + (9.1875''/12)$

$= 10.7656'$

```
10.0903
   XROM "FTd"
10.7656    ***
```

Convert to feet-inches-sixteenths:

$3.1294' = 3' + .1294 \times 12''$

$= 3' + 1.55''$

$= 3' + 1'' + .55 \times 16$

$= 3' + 1'' + 8.85/16^{ths}$

$= 3'-1\,9/16$

```
3.1294
   XROM "FIS"
3.0109     ***
           RUN
3.010884   ***
           RUN
3.0109     ***
```

Subtract: $9'-7\,3/16$ $-\,4'-9\,1/2$ $\Rightarrow$ $8'-18\,\frac{19}{16}$ $-4'-9\,8/16$ = $4'-9\,11/16$

```
9.0703 ENTER↑
4.0908
  XROM "FIS-"
4.0911    ***
```

Add: $9'-7\,3/16$ + $4'-9\,1/2$ + $7'-6\,5/8$ = $20'-22\,21/16$

$= 21'-11\,5/16$

```
9.0703 ENTER↑
4.0908
  XROM "FIS+"
7.0610
  XROM "FIS+"
21.1105   ***

9.0703 ENTER↑
4.0908      +
7.0610      +
20.2221   ***

 .8884      +
21.1105   ***
```

Quadratic Equation

$9x^2 + 2x - 33 = 0$

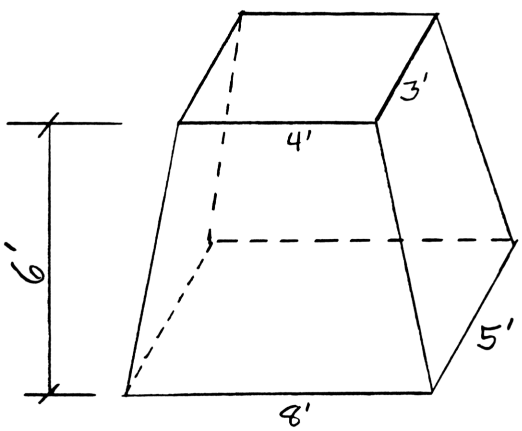
$x = \frac{-2 \pm \sqrt{2^2 - 4 \times 9 \times (-33)}}{2 \times 9} = -2.0292$

or $= 1.8070$

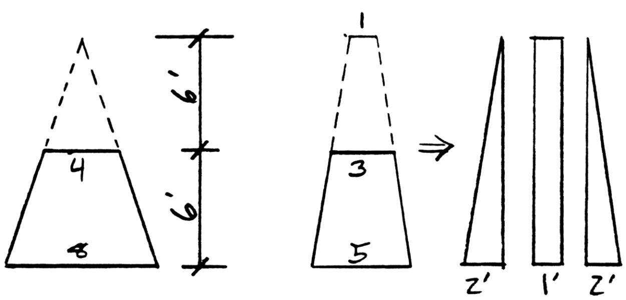
```
          XROM "QUAD"
A
          9.0000   RUN
B
          2.0000   RUN
C
        -33.0000   RUN
R1=1.8070
R2=-2.0292
A
```

Pyramid Volume

Find the volume of this pyramid. Compare different methods.



True Volume



Elevation   End

Entire

End pyramid $\frac{1}{3} \times 12' \times 2' \times 8' = 64.0$ c.f.

✓ ✓ ✓ ✓ 64.0

Center $\frac{1}{2} \times 8' \times 12' \times 1' = \underline{48.0}$

176.0 c.f.

Top Half

End pyramid $\frac{1}{3} \times 6' \times 1' \times 4' = 8.0$ c.f.

✓ ✓ ✓ ✓ = 8.0

Center $\frac{1}{2} \times 4' \times 6' \times 1' = \underline{12.0}$

24.0

Truncated vol. = 176 − 28 = **148.00 c.f.**

Program PYR approximation

$A_1 = 3 \times 4 = 12.0 \, \square'$

$A_2 = 5 \times 8 = 40.0 \, \square'$

$\sqrt{A_1 A_2} = 21.91 \, \square'$

$\text{Vol.} = \frac{1}{3} \times 6' \times (12 + 40 + 21.91)$

$= 6' \times 24.64 \, \square'$

$= \underline{\underline{147.82 \text{ c.f.}}}$

error = 0.12%

```
3.0000 ENTER↑
4.0000 ENTER↑
8.0000 ENTER↑
5.0000
   XROM "PYR"
24.6363     ***

6.0000        *
147.8178    ***
```

Simpson's Method

$V = \frac{1}{6} (A_1 + A_2 + 4A_{mid}) \times \text{Ht.}$

$A_{mid} = \left(\frac{3' + 5'}{2}\right) \times \left(\frac{4' + 8'}{2}\right) = 24.0 \, \square'$

$V = \frac{1}{6} \times 6' \times (12 + 40 + 4 \times 24)$

$= \underline{\underline{148.00 \text{ c.f.}}}$ − 0% error

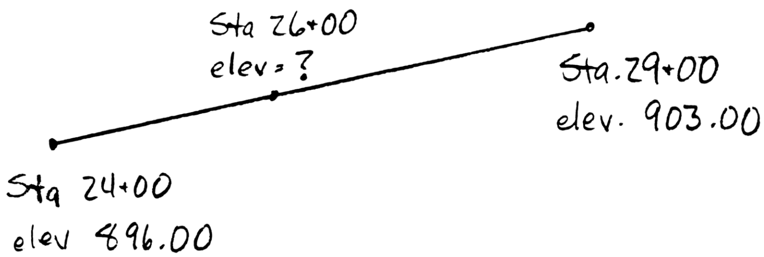
Avg. End Area

$V = \frac{1}{2}(A_1 + A_2) \times H = \frac{1}{2}(12 + 40) \times 6$

$= 156.0$ c.f. : 5.4% error

Using $\frac{1}{2}$ the interval results in $(\frac{1}{2})^2$ the error.

Interpolate/Extrapolate



$$896.0 + \frac{2600-2400}{2900-2400} \times (903.0-896.0)$$

$$= 898.80$$

```
                 XROM "IP"
X1↑Y1
      2,400.0000 ENTER↑
        896.0000    RUN
X3↑Y3
      2,900.0000 ENTER↑
        903.0000    RUN
X2
      2,600.0000    RUN
Y2=898.8000
---------
X2
      3,000.0000    RUN
Y2=904.4000
---------
X2
                    RUN
X1↑Y1
```

UTILITY

PROGRAMS

FUNCTIONS and UTILITY ROUTINES

This module contains several short routines which were written to mimic or emulate HP-41 functions; several others which are short, useful time savers for everyday work; a number of output, display, and print format functions, and several programming routines to simplify writing and using programs.

In the following write-up, one asterisk (*) signifies the use of register "a", meaning a shortened subroutine return chain. Two asterisks (**) mean part or all of the alpha register is used for "scratch-pad" storage. Three asterisks mean both are used. See the section "Synthetic Programming and Other Loose Ends" in the first part of the documentation for an explanation of these and other things that your Owner's Handbook never told you about your HP-41.

FUNCTIONS

The following routines mimic HP-41 functions as closely as possible. They leave the operand in Last X and are thus reversible. They use no numbered storage registers (i.e. they work with SIZE 00). Binary functions like PR and all unary functions leave the rest of the stack completely undisturbed, while the two binary functions FIS+ and FIS- effect stack rolldown. They are most useful from the keyboard when assigned to a key and executed in USER mode; they may also be executed from programs.

Stack Manipulation

Just like X<>Y or X<>Z, these exchange registers in the stack.

YZ	Exchange the Y and Z registers.
YT	Exchange the Y and T registers.
ZT	Exchange the Z and T registers.

Dimensions (**)

These routines work with dimensions in feet-inches-sixteenths, and are intended to be used with the display mode set to FIX 4. The form used is very similar to that used by the HP-41 function HMS. For example, 7' 11-13/16" is 7.1113; 12' 2-7/16" is 12.0207. All fractional inches must be in sixteenths. Thus 6' 5-3/4" is 6.0512.

FIS	Converts decimal feet to feet-inches-sixteenths. Once executed, pressing R/S once resets the display to FIX 6 (for example, to see the next digit in order to round to the nearest eighth of an

C1

inch). Pressing R/S (or executing any of the dimension routines) again resets FIX 4 format.

FIS+ — Adds dimensions in feet-inches-sixteenths. Key in first dimension, enter, key in second dimension, xeq FIS+ (or assign to a key). Stack is rolled down just as for normal addition. Does not have the FIX 6/FIX 4 feature like FIS.

FIS- — Subtracts dimensions in feet-inches-sixteenths.

FTd — Converts feet-inches-sixteenths to decimal feet. Does not have the FIX 6/FIX 4 feature like FIS.

Trigonometry Functions (*)

These functions accept and output angles in degrees-minutes-seconds if flag 03 is set, and in decimal degrees if flag 03 is clear. Thus if flag 03 is clear they are identical to their namesake HP-41 functions (except a little slower). They all save their operand in Last X and are thus reversible. All of them require the calculator to be set to DEG (degree) mode. Subroutine return depth is limited to two.

CO	Cosine
SI	Sine
TA	Tangent
ACO	Arc cosine
ASI	Arc sine
ATA	Arc tangent
RP	Rectangular to polar conversion
PR	Polar to rectangular conversion

SHORT UTILITY ROUTINES

These routines find use in daily manual calculations. All of them use the full stack.

QUAD — Quadratic equation solution. The program prompts for A, B, and C, the constants in the quadratic equation $Ax^2 + Bx + C = 0$. It then solves (in subroutine Q) for the two possible values of x, where $x = (-B + \sqrt{B^2 - 4AC})/2A$. If $B^2 - 4AC$ is negative, the calculator flashes DATA ERROR and quits.

(c2)

IP Linear interpolation or extrapolation. IP solves for Y2, where

X1→ Y1
X2→ Y2
and X3→ Y3.

The program prompts for X1, Y1, X3, and Y3, then repeatedly prompts for X2 and solves for Y2. The prompts are X1↑ Y1, X3↑ Y3, and X2; the upward arrow in the first two means "enter↑ ." Thus, key in X1, enter, key in Y1, press R/S. Repeat for X3 and Y3. Then key in X2 and press R/S to get Y2 as many times as you want. The program will continue to solve for Y2 until the X2 prompt is bypassed. See the example for clarification. The routine has a default value of X1 = 0; if X1 is zero, just key in Y1 and press R/S. The user should note that the entire stack is used; <u>any intermediate calculations during input will cause an error.</u>

ST Storage register review. Displays first the stack and Last X, then all the storage registers from 00 on up, until R/S is pressed or the SIZE limit is reached.

PROGRAMMING, OUTPUT, and DISPLAY

These are used almost exclusively in programs, having very little use from the keyboard. They are included in a user's program by keying (in PRGM mode) xeq (alpha) "Title", (alpha); they list as XROM "Title", which means "execute from ROM."

AX (*) Append X. Very similar to the ARCL X command. Rounds the contents of the X register and appends the integer portion of that to alpha, but leaves the original X (and the rest of the stack and Last X) undisturbed. AX is very useful, for instance, if the X register contains a counter number. Subroutine returns are limited to two deep when AX is used. AX is one of a few routines in this module that cannot be single-stepped through, because it uses register P (see "Synthetic Programming and Other Loose Ends"). If you want to single-step through it to see how it works, copy it down into RAM and replace every line using the P register with one using a numbered register instead. Since

C3

the P register is used, only 21 characters can be placed in Alpha instead of the 24 usually available.

S — Setup routine. Checks the program's SIZE requirements (the largest numbered storage register used must be in X); sets or clears flag 26 to control the beeper according to flag 02; sets flag 21 to control viewing and printer use; and sets DEG mode. If a printer is present, S prompts for a title; if a time module (or a -41CX) is also present, it prints a time and date header first. Typically, a program using S looks like this:

```
01  Lbl "YOUR TITLE"
02  12  (or whatever the
         largest storage
         register used by
         the program is)
03  XROM "S"
04  etc.
```

If your program requires no numbered storage registers, substitute

```
02  1
03  CHS  (change sign)
04  XROM "S"
```

to indicate SIZE 000 required (since if S sees zero, it thinks register 00 is needed and sets the size to 001.

V — View. Appends "=" and the contents of the X register to what the program has already (hopefully) placed in alpha, and displays it if flag 00 is clear. If flag 00 is set, V returns without doing anything. V starts at line 392 of program PLG, and is used mostly by that program. The only reasons V wasn't used in more of the module programs are that it makes it impossible to append units to the output, and that while the calculator is displaying something from V, all of the calling program's softkeys are dead (since the calling program doesn't share the same END as V). If the user sees an answer which doesn't look right, the instinctive (well, it's acquired) move to correct something by pressing a softkey gets nowhere.

C4

PV Print or View. If no printer is attached, behaves just like an AVIEW command. However, if a printer is on line, PV prints the contents of Alpha (PRA) <u>without</u> displaying it on the screen. With a long text string, this eliminates the infuriatingly slow scrolling process, speeding up output greatly. Its main disadvantage is the same as V: a program's softkeys are dead while in PV. To use a softkey, SST to return to the calling program first.

L, LL, L*, CL Line printing commands. L prints a short line of minus signs; LL prints a line all the way across the paper (24 places). L* prints a full-length line of asterisks. CL prints a line of any character (or string of characters) the program has placed in alpha; the number of times the character is repeated, and thus the length of the line, is determined by the number in the X register. This number shouldn't be larger than 24 for one character in alpha.

If flag 14 is set, both L and LL clear it and then jump to L*. Thus, a loop such as the following will print a list of asterisks the first time and a short line (----) thereafter:

```
42  SF 14

43  LBL 01
44  XEQ "L"
45    .
      .
49  GTO 01
```

L* and CL change the stack if actually executed full length (i.e. if a printer is plugged in). LL and L also alter the stack if flag 14 is set, sending them to L*. Thus, necessary data in the stack should be stored and then recalled if these are to be used.

OK Appends "OK" to alpha if flag 07 is clear. If flag 07 is set, appends "NG" (no good) and BEEPs.

Pd Prompts for degrees. Appends " (dd)" to Alpha and then prompts if flag 03 is clear. Appends " (DMS)", prompts, and then <u>converts</u> the input to

	decimal degrees if flag 03 is set.
Dd	Display degrees. If flag 03 is clear, appends an "=", the contents of the X register, and " (dd)". If flag 03 is set, appends "=", the contents of the X register converted to degrees-minutes-seconds, and " DMS". If flag 03 is set, the contents of X are converted to degrees-minutes-seconds and then back to decimal degrees; X is thus unaffected, but Last X contains the angle in degrees-minutes-seconds.
IN	Appends " (inches) to the alpha register
IS	Appends _SQ" (square inches) to alpha.
FT	Appends ' (feet) to alpha.
FS	Appends _SQ' (square feet) to alpha.
NO	Appends the number symbol (#) to alpha.
P1	Appends "(" to alpha
P2	Appends ")" to alpha
PP	Appends " ⊢" (the perpendicular sign)
MU	Appends the Greek "μ" (mu) to alpha
AT	Appends "@" (the "at" symbol) to alpha
AM	Appends "&" (ampersand) to alpha
M=	Appends M=, divides the moment (in the X register) by 12 and appends it and 'K
FK	Appends _'K (foot-kips) to alpha.
K	Appends _STRS=, the contents of the X register, and _KSI to alpha.
KS	Appends _KSI to alpha.
AS	Appends A-S=, the contents of the X register, and _SQ"
M	Prompts for M,'K
FY	Prompts for F-Y KSI
N	Prompts for F-Y or F-S, and Fc', solving for β_1, $.75P_b$ and M_{cr}, and leaves "N" in Alpha for the calling program to prompt
BD	Prompts for B" and D"
W	XEQ's "S" and asks WSD?, setting flag 06
AA	XEQ's "S" and then displays the contents of the alpha register if flag 04 is set, "AASHTO" if flag 04 is clear

ADDITIONAL PROGRAMMING ROUTINES

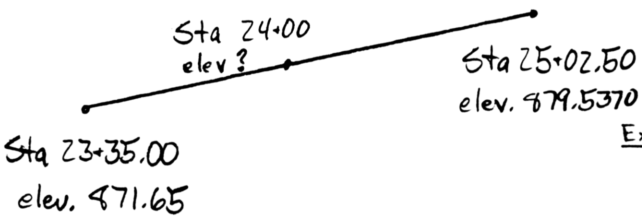
The following routines are used by programs in the ROM, and are briefly described here in case they may be of some use.

Mc	Displays 1.2 M_{cr} and calculates reinf.
SBM	"Port of entry" to program COMP
R	Displays max/min reinforcing
RH	Displays .75P_b
U	Calculates ultimate strength reinf.
VC*	"Port of entry" to programs VC and STA

C7

Interpolate:

$$\frac{2400-2335.0}{2502.5-2335.0}(879.537-871.65)+871.65 = 874.7106$$



Interpolate

Extrapolate

Start Over

```
                      XEQ "IP"
X1↑Y1
       2,335.0000 ENTER↑
         871.6500    RUN
X3↑Y3
       2,502.5000 ENTER↑
         879.5370    RUN
X2
       2,400.0000    RUN
Y2=874.7106
---------
X2
       2,500.0000    RUN
Y2=879.4193
---------
X2
       2,600.0000    RUN
Y2=884.1279
---------
X2
       3,000.0000    RUN
Y2=902.9626
---------
X2
                     RUN
X1↑Y1
```

Storage and Stack Review

(here, size 013)

```
                      XEQ "ST"
X=3.0000
Y=17,591.0646
Z=311.5196
T=1,646.9885
L=165.2249

R.0=3.5000
R.1=0.6667
R.2=0.0000
R.3=23,250.0000
R.4=-0.1111
R.5=23,550.0000
R.6=0.0000
R.7=896.0000
R.8=23,341.3100
R.9=897.0000
R.10=17,591.0646
R.11=37,380.5362
R.12=17,591.0646
```

Program S, etc. For Utility Routines 810 bytes Page 1 of 4

Step	X	Y	Z	T	L	Comments
01♦LBL "S"	MaxReg	Y	Z	T		
02 SF 21						for aview
03 SF 26						for BEEP
04 CF 14						
05 FC? 02						
06 CF 26						
07 DEG						
08 SF 25						
09 STO IND X						SIZE check
10 FC?C 25						
11 X<0?						
12 GTO 03						
13 E						
14 +						
15 "SIZE<"						
16 XROM "AX"						
17 PROMPT						
18♦LBL 03						
19 RDN	Y	Z	T			
20 FS? 00						
21 RTN						
22 FC? 55						
23 RTN						
24 SF 25						
25 CLA						
26 TIME	Time	Y	Z	T		
27 FC?C 25						
28 GTO 10						
29 ATIME						
30 "⊢ "						
31 RDN	Y	Z	T			
32 DATE	Date	Y	Z	T		
33 ADATE						
34 SF 25						
35 PRA						
36 RDN	Y	Z	T			
37 FC?C 25						
38 RTN						
39♦LBL 10	Y	Z	T			
40 ADV						
41 "TITLE:"						
42 AON						
43 PROMPT						
44 AOFF						
45 GTO 19						
46♦LBL "AX"	X	Y	Z	T	L	
47 X<> L	L				X	
48 STO ↑						LastX in R.P
49 CLX						
50 RCL d	d	Y	Z	T	X	
51 X<> a	a					"d" in R.a
52 X<> L	X	Y	Z	T	a	
53 FIX 0						
54 CF 29						
55 SF 25						SF 25 in case X-reg. contains Alpha data - for RND function
56 RND	X'	Y	Z	T		
57 X<> ↑	L					
58 X<> L	X	Y	Z	T		
59 X<> a	d	Y	Z	T		X in R.a
60 ARCL ↑						Arcl rounded X
61 STO d						
62 CLX	0	Y	Z	T		
63 STO ↑						clear Reg. P
64 X<> a	X	Y	Z	T	L	clear Reg. a
65 RTN						
66♦LBL "N"						
67 "ACI"						
68 XEQ 11						Code Display
69 FS? 06						
70 GTO 04						
71 "F-S 20,24"						
72 24						
73 XEQ 05						
74 X<>Y	24	2Fs				
75 -	24 or 16					
76 .4						
77 /	60 or 40					
78 X<> 01	Fs					Fy in R.01
79 STO 13						
80 GTO 07						
81♦LBL 05						
82 ENTER↑	default	default				
83 PROMPT	Fs					
84 X>Y?						
85 BEEP						
86 STO 01						
87 ST+ X	2Fs	default				
88 X<=Y?						
89 BEEP						
90 RTN						
91♦LBL 04						
92 "F-Y 40,60"						
93 60						
94 XEQ 05						Fy in R.01
95♦LBL 07						
96 CLA						
97♦LBL "Fc"						
98 4						
99 "⊢Fc'KSI"						
100 PROMPT	fc'					
101 9	9 ksi					
102 X<Y?						
103 BEEP						
104 RDN	fc'					
105 FS?C 10						
106 RTN						

Program S,etc. For Utility Routines 210 bytes

Step	X	Y	Z	T	L
107 ENTER↑					
108 ENTER↑					
109 SQRT	√fc'	fc'	fc'		
110 253	253	√	fc'	fc'	
111 /					
112 STO 06					
113 RDN	fc'				
114 4	4	fc'	fc'	fc'	
115 -					
116 X<0?					
117 CLX					
118 CHS					
119 20	1/05				
120 /					
121 .85					
122 +	β1				
123 .65	.65	β1?	fc'	fc'	
124 X>Y?					
125 X<>Y		β1			
126 RDN					
127 STO [	β1	fc'	fc'		
128 LASTX	.85				
129 R↑	fc'	.85	β1	fc'	
130 *					
131 RCL 01	fy				
132 /					
133 STO 00	.85fc'/fy	β1	fc'	fc'	
134 *					
135 87					
136 *					
137 RCL 01	Fy				
138 LASTX	87				
139 +	87+Fy				
140 /	ρbal				
141 .75					
142 *	.75ρbal				
143 STO 02					
144 RCL [	β1				
145 R↑	fc'	β1	.75ρbal	fc'	
146 "N"					prepared for "N" prompt in calling program
147 RTN					
148♦LBL "Mc"	bd²				
149 RCL 06	√fc'/253				
150 *					$\frac{bd^2}{6} \times \frac{1.2(7.5\sqrt{1000}\sqrt{f_c'})}{12,000} = bd^2 \times \sqrt{f_c'}/253$
151 "1.2Mc="					
152 ARCL X					
153 XROM "FK"					
154 FC? 00					
155 XEQ "PY"					
156 12					
157 *	1.2Mcr in-k				

Step	X	Y	Z	T	L
158♦LBL "U"					
159 STO a	M				
160 SIGN	1				
161 LASTX	M				
162 2	2				
163 RCL 00	fy	2	M	1	
164 /					
165 *	2M/fy	1	1	1	
166 RCL 01	.85fc'/fy				
167 /					
168 RCL 05	φbd²				
169 /					
170 -					
171 SQRT					
172 -					
173 RCL 00	fy				
174 *	As				
175 RCL a	M				
176 X<>Y	As	M			
177 RTN					
178♦LBL 02					
179 " "					
180♦LBL "AS"					
181 "⊢A-S="					
182 ARCL X					
183♦LBL "IS"					
184 "⊢ SQ"					
185♦LBL "IN"					
186 "⊢""					
187 RTN					
188♦LBL "R"	d	ρ·b			
189 *	As-1.2Mcr				
190 LASTX	d				
191 RCL 03	b				
192 *	Ac				
193 .2	200				
194 RCL 01	Fy				
195 /	200/Fy				
196 *	As-200/Fy				
197 FC? 04					
198 X<>Y	As-min				
199 FS? 00					
200 RTN					
201 XEQ 02					
202 FC? 04					
203 AVIEW					
204 "200/FY="					
205 ARCL L					
206 AVIEW					
207 XEQ 02					
208 FS? 04					
209 AVIEW					
210 ".75Pb="	As-min				

```
                      X          Y        Z        T     L
212♦LBL "K"
213 "⊦ STRS"

214♦LBL "KS"
215 "⊦="
216 ARCL X
217 "⊦ KSI"
218 RTN

219♦LBL "M="          M''·k
220 12
221 /                 M'·k
222 "⊦ M="
223 ARCL X

224♦LBL "FK"
225 "⊦ 'K"
226 RTN

227♦LBL "M"
228 12
229 CF 22
230 "⊦M,'K"
231 RTN

232♦LBL "QUAD"
233♦LBL 08
234 "A"
235 PROMPT
236 "B"
237 PROMPT
238 "C"               C          B        A               Ax²+Bx+C=0
239 PROMPT
240 X<> Z             A          B        C
241 XROM "Q"          R1         R2
242 SF 21
243 "R1"
244 XROM "Y"
245 X<>Y               R2         R1
246 "R2"
247 XROM "Y"
248 GTO 08

249♦LBL "Q"           A          B        C
250 ST/ Z
251 ST+ X             2A         B        C/A
252 /
253 CHS               -B/2A      C/A
254 ENTER↑
255 ENTER↑            B²/4A²     -B/2A    -B/2A    C/A
256 X↑2                                                   C/A = 4AC/4A²
257 R↑                C/A
258 -                 (B²-4AC)/4A²
259 SQRT              √(B²-4AC)/2A  -B/2A  -B/2A
260 ST- Z                                 R2
261 +                 R1         R2
262 RTN
```

```
                      X          Y        Z        T     L
263♦LBL "L"
264 E
265 GTO 01

266♦LBL "LL"
267 4

268♦LBL 01
269 RDN
270 FS?C 14
271 GTO 19
272 "------"
273 GTO 06

274♦LBL "NO"
275 "⊦#"
276 RTN

277♦LBL "L*"
278♦LBL 19
279 "******"
280 4

281♦LBL "CL"          counter    X        Y        Z
282 RDN

283♦LBL 06            X          Y        Z        counter
284 FC? 55
285 RTN
286 ASTO L

287♦LBL 00
288 ARCL L
289 DSE T
290 GTO 00
291 SF 25
292 PRA
293 CF 25
294 RTN               X          Y        Z        0

295♦LBL "BD"
296 RCL 03            Prev.B
297 "B="
298 PROMPT            B
299 STO 03
300 RCL 04            Prev.D
301 "D="
302 PROMPT            D
303 STO 04
304 RTN
```

Program S etc. For ________ 410 bytes Page 4 of 4

Step	X	Y	Z	T	L
305♦LBL "W"					
306 XROM "S"					
307 CF 06					
308 "WSD?"					
309 E	1				
310 PROMPT					
311 X=0?					
312 SF 06					
313 RTN					
314♦LBL "AA"					
315 ASTO Y	Max.Reg	"AISC"	z	T	
316 XROM "S"	"AISC"	z	T		
317 CLA					
318 ARCL X					
319♦LBL 11					
320 FC? 04					
321 "AASHTO"					
322 FC? 00					
323 AVIEW					
324 RTN	"AISC"	z	T		
325♦LBL "OK"					
326 FS? 07					
327 "⊢-NG"					
328 FC? 07					
329 "⊢-OK"					
330 FS? 07					
331 BEEP					
332 END					

Program FIS,FT For ____________ 157 bytes Page 1 of 1

Step	X	Y	Z	T	L
01♦LBL "FIS"					
02 XEQ 01	Ft	Y	Z	T	
03 XEQ 04	F.IS	Y			Ft
04 XEQ 02	Y	Z	T	F.IS	
05 R↑	F.IS	Y	Z	T	Ft.
06 RTN					
07 FIX 6					
08 STOP					
09 FIX 4					
10 STOP					
11♦LBL "FIS-"					
12 XEQ 05	Ft_2	Ft_1			
13 X<>Y	Ft_1	Ft_2			
14 -	ΔFt.				
15 GTO 06					
16♦LBL "FIS+"					
17 XEQ 05	Ft_2	Ft_1			
18 +	Ft				
19♦LBL 06					
20 XEQ 04	F.IS				
21 RCL \	T	F.IS			
22♦LBL 02	X	Y			
23 RCL \	T				
24 RCL]	Z	T	X	Y	
25 R↑					
26 RTN	Y	Z	T	X	
27♦LBL 04	Ft				
28 INT	Ft.				
29 LASTX					
30 FRC	.Ft				
31 12	12	Frc	Ft.	Y	
32 *	In.				
33 FRC	FRC				
34 .16	.16	FRC"	Ft.	Y	
35 ST* Y	.16	Sixteenths	Ft.	Y	
36 RDN					
37 LASTX	In.				
38 INT	In.	Sixteenths	Ft.	Y	
39 +	In.16ths				
40 E2	100				
41 /	.In16ths	Ft.	Y	Y	
42 +	F.IS				
43♦LBL 07					
44 RCL [	Ft				
45 SIGN	1				Ft in Last X
46 RDN	F.IS	Y	Y	1	
47 RTN					

Step	X	Y	Z	T	L
48♦LBL "FT"					
49 XEQ 01	F.IS	Y	Z	T	
50 XEQ 03	Ft.				
51 XEQ 07	Ft.				F.IS
52 XEQ 02	Y	Z	T	Ft.	
53 R↑	Ft.dec	Y	Z	T	F.IS
54 RTN					
55♦LBL 05	FIS_1	FIS_2			
56 XEQ 01	FIS_1	FIS_2	Z	T	
57 XEQ 03	$Ft.dec_1$	FIS_2			
58 X<>Y	$F.IS_2$	Ft.dec			
59♦LBL 03	F.IS	Y			
60 INT	Ft.				
61 LASTX					
62 FRC	.IS	Ft	Y		
63 E2	100	.IS	Ft.	Y	
64 *	In.Six				
65 FRC	.Six				
66 .16	.16	.Six	Ft.	Y	
67 ST/ Y	.16	.dec	Ft.	Y	
68 RDN					
69 LASTX	In.				
70 INT	In.	.dec	Ft.	Y	
71 +	In.dec				
72 12					
73 /	.dec	Ft.	Y	Y	
74 +	Ft.dec	Y	Y	Y	
75 RTN					
76♦LBL 01	X	Y	Z	T	
77 FIX 4					
78 STO [	X				
79 R↑					
80 STO \	T				
81 R↑					
82 STO]	Z				
83 R↑					
84 R↑	X	Y	Z	T	
85 END					

Program IP, ST, RH For Interpolation; Storage Review; Rho check 201 bytes Page 1 of 1

Step	X	Y	Z	T	L
01♦LBL "IP"					
02 SF 21					
03♦LBL 02					
04 .					
05 "X1↑Y1"					
06 PROMPT	Y_1	X_1			
07 X<>Y	X_1	Y_1			
08 "X3↑Y3"					
09 PROMPT	Y_3	X_3	X_1	Y_1	
10 R↑					
11 STO]	Y_1	Y_3	X_3	X_1	
12 -	ΔY	X_3	X_1	X_1	
13 X<> Z	X_1	X_3	ΔY	X_1	
14 -	ΔX	ΔY	X_1	X_1	
15 /	$\Delta Y/\Delta X$	Y_1	X_1	X_1	
16 RCL]	Y_1				
17 X<>Y	$\Delta Y/\Delta X$	Y_1	X_1	X_1	
18 R↑	X_1	$\Delta Y/\Delta X$	Y_1	X_1	
19♦LBL 10					
20 RDN					
21 STO a	Y/X	Y_1	X_1	X_1	
22 CF 22					
23 "X2"					
24 PROMPT	X_2				
25 FC? 22					
26 GTO 02					
27 R↑	X_1	X_2	Y/X	Y_1	
28 STO]					
29 -	ΔX	Y/X	Y_1	Y_1	
30 *	$\frac{Y}{X}\Delta X$				
31 X<>Y	Y_1				
32 +	Y_2				
33 RCL]	X_1				
34 LASTX	Y_1				$Y_2 = (Y_3-Y_1)\left(\frac{X_2-X_1}{X_3-X_1}\right)+Y_1$
35 RCL a	$\Delta Y/\Delta X$				
36 R↑	Y_2	$\Delta Y/\Delta X$	Y_1	X_1	
37 "Y2="					
38 ARCL X					
39 AVIEW					
40 XEQ "L"					
41 GTO 10					
42♦LBL "ST"					
43 SF 21					
44 "X="					
45 ARCL X					
46 AVIEW					
47 "Y="					
48 ARCL Y					
49 AVIEW					

Step	X	Y	Z	T	L
50 "Z="					
51 ARCL Z					
52 AVIEW					
53 "T="					
54 ARCL T					
55 AVIEW					
56 "L="					
57 ARCL L					
58 AVIEW					
59 ADV					
60 CLX					
61♦LBL 11	counter				
62 "R."					
63 XROM "AX"					
64 SF 25					
65 "⊦="					
66 ARCL IND X					
67 FC?C 25					
68 STOP					
69 AVIEW					
70 ISG X	counter				
71 ""	"NOP"				
72 GTO 11					
73♦LBL "RH"	ρ				
74 SF 21					
75 ENTER↑	ρ	ρ			
76 "RHO"					
77 XEQ "Y"					
78 RCL 02	$.75\rho_b$				
79 /					
80 .75					
81 *	ρ/ρ_b	ρ			
82 LASTX	.75				
83 X<>Y					
84 X<Y?					x<y? and FS? 00
85 FC? 00					
86 X=0?					
87 RTN					
88 "P/Pb="					
89 ARCL X	ρ/ρ_b	.75	ρ		
90 CF 07					
91 X>Y?					
92 SF 07					
93 XEQ "OK"					
94 XEQ "PV"					
95 END					

PPC ROM PROGRAMS

This package contains, by permission, four programs from the PPC ROM. By way of explanation, PPC is a large user's group dedicated to personal programmable calculators and computers; thus the name. Several years ago they undertook to write a group of math and general utilities programs to fill what they saw as gaps in the HP-41 function set, and to expand its usefulness. These programs were assembled into an 8K ROM module which has become something of a legend, both for its usefulness, and general excellence of programming and documentation.

The programs from the PPC ROM which have been included in this module are:

FI	Financial analysis
CV	Curve fitting (linear, exponential, log, power)
BC	Block clearing of registers
SV	Solution of a function f(x)=0 on an interval

Documentation from the original manual has been included. However, three minor changes must be noted.

First, the line numbers of the routine listings in the original documentation do not correspond to the line numbers in this module for BC and SV, due to rearrangement of the routines. Local labels have also been changed in these two routines for the same reason.

Second, BC has been modified to use the Extended Function CLRGX if it is available. This function does the same thing as the original BC but is much faster. However, the stack is affected differently by CLRGX than by BC, so carry data in the stack with caution if using this function.

Third, SV has been modified to follow this package's use of flag 00. To view successive iterations, clear flag 00 instead of setting flag 10 as the original documentation indicates.

C16

FINANCIAL ANALYSIS

Gto (not xeq) FI (SIZE 010)

SOFTKEYS/INPUT SUMMARY (USER mode ON)

Softkey	Input/Function
a	Multiplies contents of X by 12 and stores result as "n"
b	Divides contents and stores result as "%i"
c	Toggles between Continuous and Discrete compounding
d	Toggles between Beginning or End of period payment
e	Clears financial registers n, %i, PV, PMT, and FV. Sets the compounding frequency and the payment to the default value of unity. Displays a two-character mnemonic indicator of softkeys c and d (C or D, B or E)
A	Enters or solves for number of periods, n (the total number of payments or compounding periods).
B	Enters or solves for the interest rate. %i is the nominal rate for the period implied by the compounding and payment frequency values (usually the nominal annual rate).
C	Enters or solves for the present value.
D	Enters or solves for the periodic payment.
E	Enters or solves for the future value.
H	Enters the compounding frequency if different than the default, 1. CF is the number of times the interest rate is compounded during the period implied by %i. CF is ignored when continuous compounding is specified.
I	Enters the payment frequency when different than the default, 1. PF is the number of payment periods occuring during the period implied by %i. When no payments are involved, PF must be set equal to CF.
J	Displays a two-character mnemonic indicator of softkeys c and d (C or D, B or E)

USER FLAGS

Flag 00	clear	View succeeding approximations of %i
	set	View only final solution when solving for %i

OUTPUT SUMMARY

Output consists of the correct value in the X-register. No output description or units are given. See the examples in the original PPC documentation which follows.

PROGRAM FLAGS

Flag 08	clear	Discrete compounding
	set	Continuous compounding
Flag 09	clear	End of period payments
	set	Beginning of period payments

STORAGE REGISTER USE

Register	Value
00	not used
01	n
02	%i
03	PV
04	PMT
05	FV
06	function call number (indirect label)
07	%i as decimal
08	CF
09	PF

C18

a. the compounding frequency CF is not identical to the payment frequency PF, and/or,

b. interest is compounded in either discrete intervals or is continuously compounded.

When flag F08 is cleared, the discrete case is selected. When flag F08 is set, the continuous case is selected. F08 is toggled by LBL c.

Solving For n, PV, PMT, or FV.

When a solution for n, PV, PMT, or FV is required, the nominal annual interest rate i, supplied by the user, must first be converted to the effective interest rate per payment period by LBL 07. This rate, i_e, is then used by LBL 01, 03, 04, or 05 respectively to calculate the selected variable. To convert i to i_e the following expressions are used:

(19) $i_e = (1+i/CF)^{CF/PF} - 1$ (discrete case)

(20) $i_e = e^{(i/PF)} - 1$ (continuous case)

where:

- i = nominal annual interest rate
- i_e = effective interest rate per pmt. period
- CF = compounding frequency per year
- PF = payment frequency per year

Solving for interest

When a solution for interest is required, LBL 06 (for PMT≠0) or LBL 09 (for PMT=0) produces i_e as the calculated interest value. This value of i_e must then be converted to i using LBL 11. It is the value of i, not i_e which is returned as a percentage to the X-register and register R02.

To convert i_e to i, the following expressions are used:

(21) $i = CF[(1+i_e)^{PF/CF} - 1]$ (discrete case)

(22) $i = LN[(1+i_e)^{PF}]$ (continuous case)

The common label, LBL 08

Common to all calculations is LBL 08 which is used to calculate the values of A, A+1, B, and C for use in solving the selected variable. After executing the RTN instruction following LBL 08 the stack and LAST X registers contain the following data values:

Register:	Contents:
LAST X	B
T	A+1
Z	A+1
Y	A
X	C

These values are all calculated using i_e and are then used in equations (8) to (15) as selected.

Routine Listing For: **FI**

```
◆LBL "FI"
GTO IND 06
◆LBL e
◆LBL 00
 E
STO 08
STO 09
CLX
STO 01
STO 02
STO 03
STO 04
STO 05
GTO 10
◆LBL c
FC?C 08
SF 08
GTO 10
◆LBL d
FC?C 09
SF 09
◆LBL J
◆LBL 10
"D"
FS? 08
"C"
FC? 09
"⊢E"
FS? 09
"⊢B"
ASTO X
RTN
◆LBL H
STO 08
CF 22
RTN
◆LBL I
STO 09
CF 22
RTN
◆LBL a
12
*
◆LBL A
FS? 22
STO 01
FS?C 22
RTN
◆LBL 01
XEQ 07
STO Z
RCL 05
-
R↑
RCL 03
+
/
LN
RCL 07
LN1+X
/
STO 01
RTN
◆LBL b
12
/
◆LBL B
FS? 22
STO 02
FS?C 22
RTN
◆LBL 02
RCL 03
ABS
RCL 05
ABS
+
RCL 04
X=0?
GTO 09
/
ABS
1/X
LASTX
RCL 01
3
Y↑X
/
+
STO 07
◆LBL 06
XEQ 08
STO 02
RCL 03
+
STO Z
X<>Y
ST* 02
*
RCL 03
+
RCL 05
+
X<> Z
*
RCL 07
FS? 10
VIEW X
 E
+
/
RCL 01
*
RCL 02
RCL 07
/
-
/
ST- 07
RCL 07
/
 E2
*
RND
X≠0?
GTO 06
GTO 11
◆LBL 07
 E
RCL 02
%
RCL 08
RCL 09
FS? 08
X<>Y
RDN
/
STO 07
LN1+X
RCL 08
RCL 09
/
*
FS? 08
X<> 07
E↑X-1
```

Listing continued on page 161.

Listing continued from page 160.

Routine Listing For: FI

```
147 STO 07
148•LBL 08
149 E
150 RCL 07
151 FS? 09
152 ST+ Y
153 /
154 E
155 RCL 01
156 RCL 07
157 LN1+X
158 *
159 E↑X-1
160 +
161 LASTX
162 RCL 04
163 R↑
164 *
165 RTN
166•LBL 09
167 RCL 05
168 RCL 03
169 /
170 CHS
171 LN
172 RCL 01
173 /
174 E↑X-1
175 STO 07
176•LBL 11
177 CLD
178 RCL 07
179 LN1+X
180 RCL 09
181 *
182 RCL X
183 RCL 08
184 /
185 E↑X-1
186 RCL 08
187 *
188 FS? 08
189 X<>Y
190 E2
191 *
192 STO 02
193 RTN
194•LBL C
195 FS? 22
196 STO 03
197 FS?C 22
198 RTN
199•LBL 03
200 XEQ 07
201 *
202 RCL 05
203 +
204 R↑
205 /
206 CHS
207 STO 03
208 RTN
209•LBL D
210 FS? 22
211 STO 04
212 FS?C 22
213 RTN
214•LBL 04
215 XEQ 07
216 X<> L
217 *
218 CHS
219 RCL 03
220 R↑
221 *
222 RCL 05
223 +
224 X<>Y
225 /
226 STO 04
227 RTN
228•LBL E
229 FS? 22
230 STO 05
231 FS?C 22
232 RTN
233•LBL 05
234 XEQ 07
235 RCL 03
236 +
237 *
238 RCL 03
239 +
240 CHS
241 STO 05
242•LBL 12
243 END
```

Accuracy Enhancement:

The accuracy has been improved by the use of a new instruction sequence to calculate the A term:

$$(1+i)^n - 1$$

Assuming that n is stored in R01 and i (decimal) is stored in R07, the A term can be calculated in two different sequences, as follows:

Old Sequence:		New Sequence:	
RCL 07	i	RCL 07	i
1		LN1+X	LN(1+i)
+	1+i	RCL 01	n
RCL 01	n	*	n*LN(1+i)
y^x	$(1+i)^n$	E↑X-1	$(1+i)^n - 1$
1			
-	$(1+i)^n - 1$		

Mathematically, the two sequences produce identical results. However, over the range of numbers typically encountered, the LN1+X and E↑X-1 instructions prevent the severe loss of significant digits which occurs in the old sequence at the +1 and -1 steps. Reference 1 provides two examples for accuracy checking, as follows.

Examples for Accuracy Checking

These examples may be used to compare the accuracy of **FI** with other financial programs or calculators.

A.
1. Execute LBL e to clear all data registers
2. Select DISCRETE and END status (DE)
3. Key in the following variables

   n = 111.1111111
   %i = 2.222222222
   PV = 333.3333333 ($)
   PMT = 4.444444444 ($)

4. Solve for FV
   The displayed FV = -5931.822943
   The true FV = -5931.822944

B.
1. Execute LBL e to clear data registers
2. Select DISCRETE and END status (DE)
3. Key in the following variables

   n = 63
   %i = 0.000001610
   PV = 0 (no need to enter this)
   PMT = -1,000,000.00 ($)

4. Solve for FV
   The displayed FV = 63,000,031.43 ($)
   The true FV = 63,000,031.44 ($)

5. Now set FV = 0

6. Solve for PV
   The displayed PV = 62,999,967.55 ($)
   The true PV = 62,999,967.54 ($)

The above examples are taken from Reference 3 and are © copyright 1977, Hewlett-Packard Company. Reproduced with permission.

Simplified Solution Sequence

1. Solving for n:

   a. Calculate i_e, using equations (19) or (20), and LBL 07.
   b. Calculate A, A+1, B, and C using equations (5), (6), and (7), and LBL 08.
   c. Calculate n, using equation (8) and LBL 01.
   d. Store n in R01 and halt.

2. Solving for i:

   A. PMT=0

      a. Calculate i_e using equation (9) and LBL 09
      b. Calculate i using equations (21) or (22) and LBL 11.
      c. Store i and halt.

FI - FINANCIAL CALCULATIONS

This is a complete financial program that uses the top two rows of keys to either input or solve for the five standard financial values; n, %I, PV, PMT, FV.

This highly accurate program extends the capabilities of previous HP financial calculators and programs by adding two new parameters.

1. "CF" The Compounding Frequency can be specified (including continuous compounding) and may be different than the payment frequency.

2. "PF" The Payment Frequency can be specified and may be different from the compounding frequency.

This added facility simplifies the solution of some complex financial problems that are difficult to solve via the standard financial calculator or program. Canadian and European style mortgage problems can now be handled in a simple straightforward manner.

A Beginning/End of period switch is provided and a status display function allows the user to determine the current state of toggle controlled functions. The "CLEAR" financial register function incorporates default parameters that permit the user to operate the program in the same manner as typical financial calculators or programs that do not include the facility to specify different compounding and payment periods. Standard financial sign conventions are used (money paid out is negative, money received is positive).

The LN1+X and E↑X-1 functions are used in compounding routines instead of Y↑X, resulting in more precise answers than are produced by most financial programs and calculators.

BACKGROUND FOR FI

Time Value of Money:

If you borrow money you can expect to pay rent or interest for its use; conversely you expect to receive interest on money you loan or invest. When you rent property, equipment, etc., rental payments are normal; this is also true when renting or borrowing money. Therefore, money is considered to have a "time value". Money available now, has a greater value than money available at some future date, because of its rental value or the interest that it can produce during the intervening period.

Simple Interest:

If you loaned $800 to a friend with an agreement that at the end of one year he would repay you $896, the "time value" you placed on your $800 (principal) was $96 (interest) for the one year period (term) of the loan. This relationship of principal, interest and time (term) is most frequently expressed as an Annual Percentage Rate (APR). In this case the APR was 12.0% [(96/800)100]. This example illustrates the four basic factors involved in a simple interest case. The time period (one year), rate (12.0% APR), present value of the principal ($800) and the future value of the principal including interest ($896).

Compound Interest:

In many cases the interest charge is computed periodically during the term of the agreement. For example, money left in a savings account earns interest that is periodically added to the principal and in turn earns additional interest during succeeding interest periods. The accumulation of interest during the investment period represents compound interest. If the loan agreement you made with your friend had specified a "compound interest rate" of 12.0% (compounded monthly) the $800 principal would have earned $101.46 interest for the one year period. The value of the original $800.00 would be increased by 1% the first month to $808.00 which in turn would be increased by 1% to $816.08 the second month, reaching a future value of $901.46 after the twelfth iteration. The monthly compounding of the nominal annual rate (NAR) of 12% produces an effective Annual Percentage Rate (APR) of 12.683% [(101.46/800)100]. Interest may be compounded at any regular interval; annually, semiannually, monthly, weekly, daily, even continuously (a specification in some financial models).

Periodic Payments:

When money is loaned for longer periods of time it is customary for the agreement to require the borrower to make periodic payments to the lender during the term of the loan. The payments may be only large enough to repay the interest, with the principal due at the end of the loan period (an interest only loan), or large enough to fully repay both the interest and principal during the term of the loan (a fully amortized loan). Many loans fall somewhere between, with payments that do not fully cover repayment of both the principal and interest. These loans require a larger final payment (balloon) to complete their amortization. Payments may occur at the beginning or end of a payment period. If you and your friend had agreed on monthly repayment of the $800 loan at 12.0% NAR compounded monthly, twelve payments of $71.08 for a total of $852.96 would be required to amortize the loan. The $101.46 interest from the annual plan is more than the $52.96 under the monthly plan because under the monthly plan your friend would not have had the use of $800 for a full year.

Financial Transactions:

The above paragraphs introduce the basic factors that govern most financial transactions; the time period, interest rate, present value, payments, and the future value. In addition, certain conventions must be adhered to; the interest rate must be relative to the coumpounding frequency and payment periods, and the term must be expressed as the total number of payments (or compounding periods if there are no payments). Loans, leases, mortgages, annuities, savings plans, appreciation, and compound growth are among the many financial problems that can be defined in these terms. Some transactions do not involve payments, but all of the other factors play a part in "time value of money" transactions. When any one of the five (four- if no payments are involved) factors is unknown, it can be derived from formulas using the known factors. This is the function of the FI financial program.

Problem Solving Preliminaries:

Diagram or visualize the positive and negative cash flows and their timing. (See cash flow diagrams)

Clear the financial registers by pressing e , unless the problem is a continuation or minor change from the preceding problem. Note that clearing also sets the compounding and payment frequency values to 1.

Check the mnemonic status code for applicability to the current problem and change it by pressing c and/or d if necessary. The mnemonic status codes are:

CB = Continuous compounding and Beginning of period payments.
CE = Continuous compounding and End of period payments.
DB = Discrete compounding and Beginning of period payments.
DE = Discrete compounding and End of period payments.

Specify the compounding and payment frequencies by entering appropriate values and pressing H and/or I.

Generalized Cash Flow Diagrams:

Selection of the proper parameters and signs of the factors in a specific financial transaction can often be aided by constructing a cash flow diagram similar to the examples below:

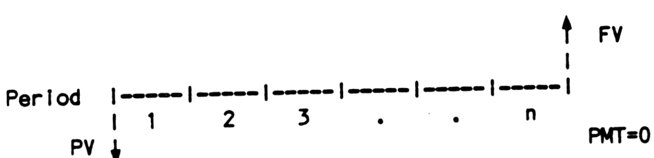
Standard Financial Conventions are:

Money RECEIVED is a POSITIVE value and is represented by an arrow above the line.

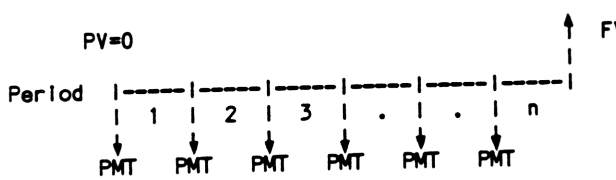
Money PAID OUT is a NEGATIVE value and is represented by an arrow below the line.

If payments are a part of the transaction the number of payments must equal the number of periods (n).

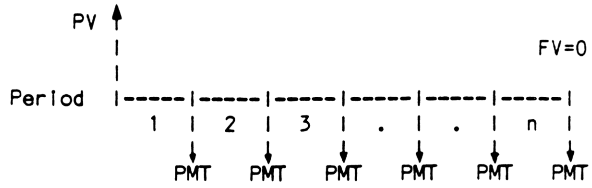
Payments may be represented as occuring at the end or beginning of the periods.



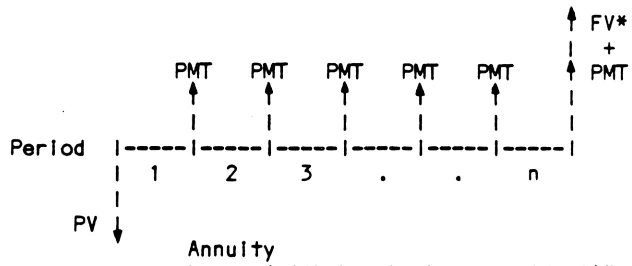
Appreciation
Depreciation
Compound Growth
Savings Account



Annuity (series of payments)
Pension Fund
Savings Plan
Sinking Fund



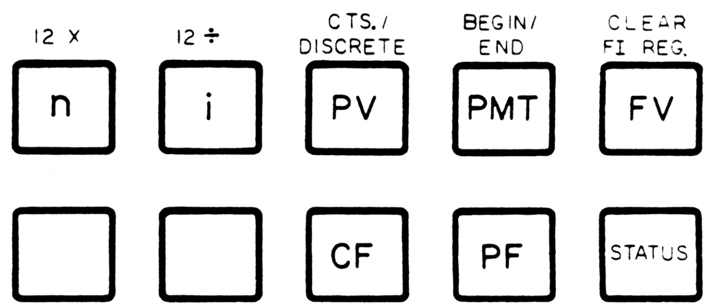
Amortization
Direct Reduction Loan
Mortgage (fully amortized)



Annuity
Lease (with buy back or residual)*
Loan or Mortgage (with balloon)*

COMPLETE INSTRUCTIONS FOR FI

By manually keying GTO "FI" the program pointer is in the ROM and the following functions become available on the top two rows of the keyboard. A minimum size required for FI is SIZE 010.



The functions provided are summarized below.

KEY	FUNCTION	(FLAG/REG)
a	Multiplies contents of X by 12 and stores result as n. n=12X	R01
b	Divides contents of X by 12 and stores result as %i. %i=X/12	R02
c	Toggles flag 08 to specify Continuous (F08 set) or Discrete (F08 clear) compounding. Status display shows C or D.	F08
d	Toggles flag 09 to specify Beginning of period payments (F09 set) or End of period payments (F09 clear). Status display shows B or E.	F09
e	Clears financial registers n, %i, PV, PMT, FV, and sets the compounding frequency (R08) and the payment frequency (R09) both to the default value of 1. CF=PF=1. Also displays a two character mnemonic indicator of the status of F08 and F09.	R01-R05 R08 R09

CM

A	Enters or Solves for number of periods. n is the total number of payments during the full term of the transaction, or if no payments are made n is the total number of compounding periods.	R01 n
B	Enters or Solves for the interest rate.** %I is the nominal rate for the period implied by the compounding and payment frequency values in R08 and R09 (usually the nominal annual rate).	R02 %I
C	Enters or Solves for the present value. Use standard financial sign conventions.	R03 PV
D	Enters or Solves for periodic payment. Use standard financial sign conventions.	R04 PMT
E	Enters or Solves for future value. Use standard financial sign conventions.	R05 FV
H	Enters the compounding frequency. CF is the number of times the interest rate is compounded during the period implied by the interest rate %I. When continuous compounding is specified the value in R08 is ignored.	R08 CF
I	Enters the payment frequency. PF is the number of payment periods occuring during the period implied by the interest rate %I. When no payments are involved PF must be set equal to CF. For continuous compounding cases where PMT = 0, set PF = 1.	R09 PF
J	Displays a two character mnemonic indicator of the status of flags F08 and F09. C = Continuous D = Discrete E = End of period B = Beginning of period	

** When solving for %I the first guess and succeeding approximations of I (decimal) may be VIEWed during each iteration by setting flag 10. F10

WARNING: Solutions using or resulting in a zero rate of interest (%I) will cause a "DATA ERROR".

MORE EXAMPLES OF F1

In the keystroke solutions shown for each example, the lower case letters a through e represent shifted key functions of keys A through E. Key in the indicated quantities and press the user defined keys as indicated in the "Do" column. Contents of the display at significant points in the solution are shown in the "See" column and are followed by identification in the "Result" column. Before running these examples, perform "MEMORY LOST" and set FIX 2 display mode. A suggestion is to go through all 15 examples, one after the other. Key GTO " F1 " and set USER mode.

Example 1: Simple Interest. Find the annual simple interest rate (%) for an $800 loan to be repayed at the end of one year with a single payment of $896.

Do:	See:	Result:
e	"DE"	Clear, Discrete/End status
1 A	1.00	n=1
800 CHS C	-800.00	PV=$800.00
896 E	896.00	FV=$896.00
B	12.00	APR=%I=12%

Example 2: Compound Interest. Find the future value of $800 after one year at a nominal rate of 12% compounded monthly. No payments are specified, so the payment frequency is set equal to the compounding frequency.

Do:	See:	Result:
e	"DE"	Clear, Discrete/End status
12 A	12.00	n=12
H I	12.00	CF=PF=12
12 B	12.00	keyboard input required to store NAR=12%=%I
800 CHS C	-800.00	PV=$800.00
E	901.46	FV=$901.46

Example 3: Periodic Payment. Find the monthly end-of-period payment required to fully amortize the loan in Example 2. A fully amortized loan has a future value of zero. Use data retained from Example 2.

Do:	See:	Result:
0 E	0.00	Set FV=$0.00
D	71.08	PMT=$71.08

Example 4: Conventional Mortgage. Find the number of monthly payments necessary to fully amortize a loan of $100,000 at a nominal rate of 13.25% compounded monthly, if end-of-period payments of $1,125.75 are made.

Do:	See:	Result:
e	"DE"	Clear, Discrete/End status
12 H I	12.00	CF=PF=12
13.25 B	13.25	NAR=13.25%=%I
100000 C	100,000.00	PV=$100,000.00
1125.75 CHS D	-1,125.75	PMT=$1,125.75
A	360.10	#pmts=n=360.10

Example 5: Final Payment. Using the same data as in the preceding example, find the amount of the final payment if n is changed to 360. The final payment will be equal to the regular payment plus any balance (FV) remaining at the end of period number 360.

Do:	See:	Result:
360 A	360.00	Set n=360 exactly
E	-108.87	FV=$108.87
RCL 04	-1,125.75	Recall PMT
+	-1,234.62	Final PMT=$1,234.62

Example 6: Balloon Payment. On long term loans, small changes in the periodic payments can generate large changes in the future value. If the monthly payment in the preceding example is rounded down to $1,125.00 how much additional (balloon) payment will be due with the final regular payment?

Do:	See:	Result:
1125 CHS D	-1,125.00	Set PMT=$1,125.00 even
E	-3,580.00	Additional balloon payment = $3,580.00

Example 7: Canadian Mortgage. Find the monthly end-of-period payment necessary to fully amortize a 25 year $85,000 loan at 11% compounded semiannually.

Do:	See:	Result:
e	"DE"	Clear, Discrete/End status
2 H	2.00	CF=2
12 I	12.00	PF=12
25 a	300.00	n=300
11 B	11.00	NAR=11%=i%
85000 C	85,000.00	PV=$85,000.00
D	-818.15	PMT=$818.15

Example 8: European Mortgage. The "effective annual rate (EAR) is used in some countries (especially in Europe) in lieu of the nominal annual rate commonly used in the United States and Canada. For a 30 year $90,000 mortgage at 14% (EAR) compute the monthly end-of-period payments. When using an EAR, the compounding frequency (CF) is set to 1.

Do:	See:	Result:
e	"DE"	Clear, Discrete/End status (CF=1 after clearing)
12 I	12.00	PF=12
30 a	360.00	n=360
14 B	14.00	EAR=14%=i%
90000 C	90,000.00	PV=$90,000.00
D	-1,007.88	PMT=$1,007.88

Example 9: Bi-Weekly Savings. Compute the future value of bi-weekly savings of $100 for 3 years at a nominal annual rate of 5.5% compounded daily. Note: Set status to "DB".

Do:	See:	Result:
e	"DE"	Clear, Discrete/End status
d	"DB"	Discrete/Begin Status
365 H	365.00	CF=365
26 I	26.00	PF=26
3 X A	78.00	n=3x26=78
5.5 B	5.50	NAR=5.5%=i%
100 CHS D	-100.00	PMT=$100.00
E	8,489.32	FV=$8,489.32

Example 10: Present Value - Annuity Due. What is the present value of $500 to be received at the beginning of each quarter over a 10 year period if money is being discounted at 10% NAR compounded monthly?

Do:	See:	Result:
e	"DB"	Clear, Discrete/Begin status
12 H	12.00	CF=12
4 I	4.00	PF=4
40 A	40.00	n=40
10 B	10.00	NAR=10%=i%
500 D	500.00	PMT=$500.00
C	-12,822.64	PV=$12,822.64

Example 11: Balloon Payment @ n+1. Compute the monthly end-of-period payment on a 3 year $20,000 loan at 15% NAR compounded monthly, with a $10,000 balloon payment due at the end of the 37th period. The balloon payment must be discounted one period to make it coincide with the last regular payment. Note: Set status to "DE".

Do:	See:	Result:
e	"DB"	Clear Financial
d	"DE"	Set Discrete/End status
12 H I	12.00	CF=PF=12
3 a	36.00	n=36
15 B	15.00	NAR=15%=i%
20000 C	20,000.00	PV=$20,000.00
XEQ 07	0.00	Calculate i as a decimal and leave in R07
10000	10,000.00	Start calculation of 10,000/(1+i) = FV
RCL 07 1 +	1.01	1+i
/ CHS E	-9876.54	FV=$9,876.54 (discounted)
D	-474.39	PMT=$474.39

The balloon payment was discounted by executing LBL 07 (XEQ 07) to develop the effective interest rate in R07. The $10,000 was then divided by (1+i) and entered in E as the discounted future value of the balloon payment.

Example 12: Effective Rate - 365/360 Basis. Compute the effective annual rate (%APR) for a nominal annual rate of 12% compounded on a 365/360 basis used by some Savings & Loan Associations.

Do:	See:	Result:
FIX 3 e	"DE"	Set up display & status
365 A	365.000	n=365
H	365.000	CF=365
360 I	360.000	PF=360
12 B	12.000	NAR=12%=i%
100 CHS C	-100.000	PV=$100.00
E	112.935	FV=$112.94
RCL 03 +	12.935	%APR=12.935%
FIX 2	12.94	Return to normal display

Example 13: Mortgage with "points". What is the true APR of a 30 year, $75,000 loan at a nominal rate of 13.25% compounded monthly, with monthly end-of-period payments of $844.33 if 3 "points" are charged? The PV must be reduced by the dollar value of the points and/or any lenders fees to establish an effective PV. Because the payments remain the same the true APR will be higher than the nominal rate.

Do:	See:	Result:
e	"DE"	Clear, Discrete/End status
12 H I	12.00	CF=PF=12
30 a	360.00	n=360
75000 ENTER↑		
3 XEQ "%" - C	72,750.00	PV=$72,750.00
844.33 CHS D	-844.33	PMT=$844.33
B	13.69	True APR=13.69%

Example 14: Equivalent Payments. Find the equivalent monthly payment required to amortize a 20 year $40,000 loan at 10.5% NAR compounded monthly, with 10 annual payments of $5,029.71 remaining. Compute PV of the remaining annual payments, then change n and PF to a

monthly basis and compute the equivalent monthly PMT.

Do:	See:	Result:
e	"DE"	Clear, Discrete/End status (PF=1 after clearing)
12 H	12.00	CF=12
10 A	10.00	n=10
10.5 B	10.50	NAR=10.5%=$I
5029.71 CHS D	-5,029.71	PMT=$5,029.71
C	29,595.88	PV=$29,595.88
12 I	12.00	PF=12, set monthly basis
10 a	120.00	n=120 (monthly)
D	-399.35	PMT=$399.35 (monthly)

Example 15: Perpetuity - Continuous Compounding
If you can purchase a single payment annuity with an initial investment of $60,000 that will be invested at 15% NAR compounded continuously, what is the maximum monthly return you can receive without reducing the $60,000 principal? If the interest rate is constant and the principal is not disturbed the payments can go on indefinitely (a perpetuity). Note that the term "n" of a perpetuity is immaterial. It can be any non-zero value. Set status to "CE".

Do:	See:	Result:
e	"DE"	Clear, Discrete/End status (CF=1 after clearing)
c	"CE"	Continuous/End status
12 A	12.00	n=12
I	12.00	PF=12
15 B	15.00	NAR=15%=$I
60000 E	60,000.00	FV=$60,000.00
CHS 1 X C	-60,000.00	Data entry flag is set so PV is stored as $60,000.00
D	754.71	PMT=$754.71

SUPPORTIVE PROGRAMS FOR FI

There are two optional routines provided below to extend the capability of the ROM routine FI. These routines are not located in the ROM, and must be loaded into RAM memory for their execution. They are named LPAS and FAST.

1. LBL LPAS

LBL LPAS "Loan Payments and Amortization Schedule" is really a full program in its own right, although it does use ROM routines FI, CJ, and CP. LPAS extends the capabilities of FI to accommodate "shifted" payment situations, when the first periodic payment does not fall at the beginning (BEGIN) or the end (END) of the first period, but at any date after the effective date. LPAS also provides an amortization schedule as an option.

2. LBL FAST - Reducing Interest Solution Time

LBL FAST is an optional routine used when solving for interest. Its purpose is to provide an initial starting guess for the interest-solving loop which is closer to the exact solution than that provided by LBL FI initial guess. The result is that interest solving execution time is usually shorter.

Don Dewey (5148) produced both supporting programs.

APPLICATION PROGRAM 1 FOR FI

LPAS - Loan Payments and Amortization Schedule

The FI program, like most financial programs and calculators, assumes that the first periodic payment occurs on either the first or last day of the payment period as specified by the beginning of period/end of period switch or toggle. Many financial agreements do not follow this convention. An agreement may call for the regular periodic payments to start earlier or later in order to provide a better match to other cash flow considerations of the borrower or lender. These agreements with "shifted" initial payment dates can be handled by conventional financial programs by computing an effective present value (PV) that compensates for the difference in interest accrued during the irregular first payment period. This computation becomes more complex when the compounding and payment frequencies (CF and PF) are unequal.

Shifting the initial payment date forces a change in the number or amount of the periodic payments or in the amount of the final or balloon payment. However, the participants to an agreement may want to specify the number and/or amount of the regular payments, and adjust the final payment to complete the amortization. Even without a shifted initial payment date or other restrictions the regular periodic payments seldom precisely complete the amortization and the final payment must be adjusted to accomplish this.

For the uninitiated or infrequent user of financial programs, the accomodation of a shifted first payment date and/or the computation of the correct final payment amount can cause problems. The following program easily handles these cases and also takes the drudgery out of computing an amortization schedule.

The LPAS program uses the FI program and the CJ and CP routines in the PPC ROM to expand the capabilities of the FI program to accomodate "shifted" initial payment dates and to compute the number and amount of periodic payments, and the final payment required to amortize a loan or to accumulate a specific future value. The information needed to prepare a loan amortization schedule may also be computed on an optional basis. The extensive capabilities of the FI program are used in their normal manner to define the parameters of a specific problem and to develop the initial solution. Two additional input parameters are provided; the effective date (ED) and the initial payment date (IP). These two dates define the length of the first payment period which need not be equal to the normal payment period implied by the payment frequency value (PF). The initial payment date (IP) also establishes the number of payments that will occur in the first year. The program computes the regular periodic payment and the final payment required to amortize a loan or to accumulate a specified future value over a specified term (n), or the number of payments and the final payment necessary to amortize a loan or to accumulate a specified future value with a specified periodic payment amount.

Conventional loans, mortgages with or without balloon payments, and Canadian or European mortgages are all acceptable to the LPAS program. Cases with payment frequencies of semi-monthly (PF=24) or less, use a 30 day month convention for determining the number of days of shift in the first payment date and the number of payments occurring in the first year. For payment frequencies greater than semi- monthly (i.e., daily, weekly, or bi-weekly) the acutal number of calendar days is used.

LPAS Program - Operation

The LPAS program computes the regular periodic payment and the final payment for both present value (PV*) and future value (FV*) cases. PV* cases involve periodic payments that reduce or amortize a present value. FV* cases involve appreciation or accumulation to a future value. The amortization schedule portion of the program supports PV* cases only. The LPAS program can be used with or without a printer (CF21).

The FI program is accessed and used to set the status (CB, CE, DB, DE), the compounding frequency (CF), the payment frequency (PF), the standard financial values (n, %I, PV, PMT, FV) and to solve for any missing financial value. Note: The FI program can be accessed by pressing "J" when the LPAS program has control. After entering the normal financial program data, the effective date of the financial agreement (ED) and the date of the initial payment (IP) are entered into the X and Y registers in the form MM.DDYYYY (Y=ED, X=IP). The IP date must not be earlier than the ED date.

The LPAS program is then executed. For easy access the LPAS program should be assigned to a key. The LPAS program was assigned to the X<>Y (F) key in the keystroke solutions in the example programs below. The program computes the regular periodic payment required to maintain the specified interest rate. The computation compensates for any fractional portion of the term and for any deviation from the normal initial payment date. When the program first stops, the computed payment (rounded to two decimal places) is in the PMT register (R04) and is displayed in the X register.

First Stop - The computed payment may be accepted, or a modified payment may be entered and substituted by pressing key "D". To continue the computation, select one of the following two options:

1. By pressing "H" the amortization period is limited to the integer portion of the term (n) and the final or balloon payment is adjusted to complete the amortization.

2. By pressing "J" the term (n) is recomputed to accomplish the amortization with the specified periodic payment with a minimum adjustment to the final or balloon payment.
The amortization choice restarts the program and the number of periodic payments and the amount of the final payment are computed. At the second stop the stack contains:

T = number of payments occuring in first year
Z = number of regular periodic payments
Y = amount of the regular periodic payment
X = amount of the combined final and balloon payments

Second Stop - An amortization schedule may be computed by pressing "E" (for PV* cases only) or control may be returned to the FI program by pressing "J".

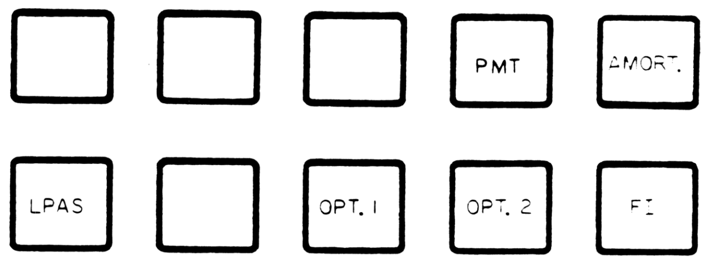
If an amortization schedule is computed and a printer is not available (FC?21) the program will stop after computing the values for each year. At each stop the stack will contain:

T = cumulative interest paid
Z = balance outstanding after last PMT for year
Y = interest paid during the year
X = year (YY)

To compute the amortization data for each succeeding year press R/S. Completion of the amortization is indicated by ** in the display. The total interest paid is in the Y register at this final stop.

After completion of the amortization, control may be returned to the FI program by pressing "J".

Keyboard Functions: (LPAS Program)



Key	Function	(Flag/Reg)
D	Enter revised periodic payment "PMT"	(R04)
E	Compute amortization data (PV* case only)	(R00-R13)
F	Enter ED and IP dates and compute periodic PMT	
H	Select Option 1 and compute final PMT	(F07)
I	Select Option 2 and compute term n and final PMT	
J	Transfer to FI program and display status	
R/S	Compute amortization data for next year	

Program requirements and limitations. CJ, CP, FI are the PPC ROM required routines. LPAS is 655 bytes SIZE=014 Flags 06-10,21,28,29

Acceptable Payment Frequencies (PF) are:

1 = annual	12 = monthly
2 = semi-annual	24 = semi-monthly
3 = tri-annual	26 = bi-weekly
4 = quarterly	52 = weekly
6 = bi-monthly	365 = daily

WARNING: Solutions using or resulting in a zero rate of interest (%I) will cause a "DATA ERROR".

The output of LPAS is printed in three sections separated by horizontal lines. The first section records the original parameters of the case. The second section records the amount and number of regular payments and the final payment necessary to satisfy the options selected. The third section displays the optional amortization schedule.

Examples:

In the keystroke solution for each example, the lower case letters a through e represent shifted functions of keys A through E. Key in the indicated quantities and press the user defined keys as indicated in the "Do" column. Contents of the display or the printed output at significant points in the solution are shown in the "See" column and are followed by identification in the "Result" column. Use FIX 2 display mode, and assign LPAS to key F (X<>Y).

Example A: Conventional Mortgage. Develop the data for an amortization schedule for a fully amortized 30-year, $100,000 mortgage at 14.75% NAR compounded monthly with end of period payments of $1,244.48 with the first payment due on November 1, 1981. Effective date of the loan is September 25, 1981. Use option 1 (H) to limit the amortization period to 360 payments.

Do:	See:	Result:
CLX STO 06	0.00	Store FI function call
XEQ " FI "	"DE"	Discrete/End status
12 H I	12.00	CF=PF=12
30 a	360.00	n=360
14.75 B	14.75	NAR=14.75%=%I
100000 CHS C	-100,000.00	PV=$100,000.00
1244.48 D	1,244.48	PMT=$1,244.48
E	-27.98	FV=$27.98
9.251981		
ENTER↑		
11.011981 F	1,247.52	PMT, shifted IP $1,247.52
H	1,248.31	Final PMT=$1.248.31
E	**	Compute amortization data, see print out below.

```
EXAMPLE A   CF=12  PF=12
◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆
PV*DE   PV   -100,000.00
  360 PMTS      1,244.48
14.750% FV        -27.98
ED  9-25-81 IP 11- 1-81*
xxxxxxxxxxxxxxxxxxxxxxxx
  359 PMTS      1,247.52
+FINAL PMT      1,248.31
xxxxxxxxxxxxxxxxxxxxxxxx
YR  INTEREST  ENDING BAL
81     2,464.   100,214.
82    14,768.   100,012.
83    14,736.    99,778.
84    14,699.    99,507.
85    14,657.    99,193.
86    14,607.    98,830.
87    14,550.    98,410.
88    14,483.    97,923.
89    14,407.    97,359.
90    14,318.    96,707.
```

```
91    14,214.    95,951.
92    14,095.    95,076.
93    13,957.    94,063.
94    13,797.    92,889.
95    13,612.    91,531.
96    13,397.    89,958.
97    13,149.    88,137.
98    12,861.    86,028.
99    12,528.    83,586.
00     2,143.    80,758.
01    11,696.    77,485.
02    11,179.    73,694.
03    10,581.    69,305.
04     9,888.    64,222.
05     9,085.    58,338.
06     8,156.    51,524.
07     7,080.    43,634.
08     5,835.    34,498.
09     4,392.    23,920.
10     2,722.    11,672.
11       804.         0.
**   348,860.
```

Note: 2 payments in 1981. The negative amortization during the first two years is due to the delayed first payment date. The asterisk following the IP date indicates a shifted initial payment date.

If a printer is not used when working Example A, after execution the stack will contain the following:

after F	after H	after E*	
T= -	T= 2.00	T = 2,464.	Σ Int.
Z= -	Z= 359.00	Z = 100,214.	E.Bal.
Y= -	Y= 1,247.52	Y= 2,464.	Yr.Int.
X= 1,247.52	X= 1,248.31	X= 81.	Year

*Press R/S to advance amortization to next year. the end of the amortization is indicated by ** in display.

Example B: Sinking Fund / Savings Plan Starting with an initial deposit of $3,000 compute the number of bi-weekly deposits of $200 and the amount of the final deposit needed to accumulate a balance of $20,000 in an account paying 8% compounded continuously, if the intial deposit (PV) is made on December 1, 1981 and the first bi-weekly deposit (PMT) is made on December 11, 1981. Set the status to CB.

Do:	See:	Result:
J	"DE"	Return to FI
c	"CE"	Set Continuous compounding
d	"CB"	Set Beginning of period payments
e	"CB"	Clear Financial Status=Continuous/Beginning CF=1 after clearing
26 I	26.00	PF=26
8 B	8.00	NAR=8%=%I
3000 CHS C	-3,000.00	PV=$3,000.00
200 CHS D	-200.00	PMT=$200.00
20000 E	20,000.00	FV=$20,000.00
A	72.43	n=72.43
12.011981		
ENTER↑		
12.111981 F	-197.29	PMT, shifted IP $197.29
200 CHS D	-200.00	Enter revised PMT $200.00
I	-91.67	Final PMT=$91.67
E	"FV* ?"	indicates attempted amortization of FV* case

```
EXAMPLE B  CF=1 PF=26
◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆
FV*CB   PV     -3,000.00
  72+ PMTS       -200.00
 8.000% FV     20,000.00
ED 12- 1-81 IP 12-11-81*
xxxxxxxxxxxxxxxxxxxxxxxx
   71 PMTS       -200.00
+FINAL PMT        -91.67
```

FV*CB = Future Value case with Continuous compounding and Beginning of period payments/deposits. The plus (+) sign following the number of payments indicates that the term includes a fractional payment period as developed from the original specifications.

Example C: Loan with Balloon Payment. Develop the amortization data for a $500,000 loan at 15% NAR with monthly compounding, to be repaid with 30 monthly end of period payments of $20,000 and a balloon payment of $3,225.30 coincident with the final payment. The loan effective date is September 14, 1981 and the first payment is scheduled for October 14, 1981.

Do:	See:	Result:
J	"CB"	Return to FI Status from previous example
c	"DB"	Set Discrete compounding
d	"DE"	Set End of period payments
e	"DE"	Clear Financial, final status= Discrete/End
12 H I	12.00	CF=PF=12
30 A	30.00	n=30
15 B	15.00	NAR=15%=%I
500000 CHS C	-500,000.00	PV=$500,000.00
20000 D	20,000.00	PMT=$20,000.00
E	3,225.30	Balloon=$3,225.30
9.141981		
ENTER↑		

10.141981 F	20,000.00	PMT=$20,000.00
H	23,225.30	Final + Balloon = $23,225.30
E	**	Compute amortization data, see print out below.

```
EXAMPLE C  CF=12 PF=12
************************
PV*DE   PV   -500,000.00
   30 PMTS    20,000.00
15.000% FV     3,225.30
ED  9-14-81 IP 10-14-81
::::::::::::::::::::::::
   29 PMTS    20,000.00
+FINAL PMT    23,225.30
::::::::::::::::::::::::
YR  INTEREST  ENDING BAL
81    18,232.   458,232.
82    56,456.   274,688.
83    26,950.    61,638.
84     1,587.         0.
**   103,225.
```

Because the initial payment occurs exactly one month after the loan effective date there is no change in the re-computed PMT.

<u>Example D:</u> Delayed First Payment This example will illustrate the effect of a different repayment plan for the loan defined in Example C. Develop the data for amortizing a $500,000 loan at 15% NAR with monthly compounding, to be repaid with 60 semi-monthly end of period payments of $10,000 and a balloon payment coincident with the final payment. The loan effective date is September 14, 1981 and the first payment is scheduled for November 1, 1981.

Do:	See:	Result:
J	"DE"	Return to F1 Status left from Example C
12 H	12.00	CF=12
24 I	24.00	PF=24
60 A	60.00	n=60
15 B	15.00	NAR=15%=$1
500000 CHS C	-500,000.00	PV=$500,000.00
10000 D	10,000.00	PMT=$10,000
E	974.25	FV=$974.25
9.141981 ENTER↑		
11.011981 F	10,268.92	PMT=$10,268.92
10000 D	10,000.00	Set PMT=$10,000.00 exactly
H	30,466.27	Final+Balloon=$30,466.27
E	**	Compute amortization data See print out below

```
EXAMPLE D  CF=12 PF=24
************************
PV*DE   PV   -500,000.00
   60 PMTS    10,000.00
15.000% FV       974.25
ED  9-14-81 IP 11- 1-81*
::::::::::::::::::::::::
   59 PMTS    10,000.00
+FINAL PMT    30,466.27
::::::::::::::::::::::::
YR  INTEREST  ENDING BAL
81    12,541.   485,968.
82    60,113.   306,081.
83    31,195.    97,277.
84     3,189.         0.
**   107,038.
```

The total interest on this repayment plan is $3,813 more than in Example C due to the delayed first payment date and the smaller payments. The borrower has the use of more money for a longer time.

LPAS Program - Equations

All equations assume the use of standard financial transaction sign conventions of money received as positive (+) and money paid out as negative (-).

Notation used:

d	= number of days in payment period
i_e	= effective interest rate per payment period
m	= integer portion of term n
n	= number of payment periods in term
s	= number of days first payment is shifted
CF	= compounding frequency per year
ED#	= effective date - day number
FV	= future value after n periods
FV_m	= future value after m periods
FV_{m-1}	= future value after m-1 periods
FV*	= future value case
INT	= interest for the year
IP#	= initial payment date - day number
NP	= number of payments in the year
PF	= payment frequency per year
PMT	= periodic payment
PMT_f	= final payment
PV	= present value
PV_e	= effective present value
PV*	= present value case

If $|FV| \le |PV|$, then PV* case
If $|FV| > |PV|$, then FV* case

The initial payment date is "shifted" when: $s \neq 0$

where: $s = IP\# - ED\#$ for beginning of period payments

$s = IP\# - ED\# + d$ for end of period payments

For financial calculations involving a "shifted" first payment date, the present value (PV) must be converted to an effective present value (PV_e) that is adjusted to compensate for the difference in interest accrued during the irregular first payment period.

$$PV_e = PV(1+i_e)^{(sPF/dCF)}$$

To precisely complete the amortization of a present value or the accrual of a future value, the final payment must be calculated separately from the regular periodic payment. The LPAS program incorporates eight variations of final payment calculations.

$PMT_f = FV_{m-1}$	PV* case, annuity due, Option 1
$= FV_m$	PV* case, annuity due, Option 2
$= FV_m + PMT$	PV* case, ordinary annuity, Option 1

$PMT_f = FV_m(1+i_e)$ PV* case, ordinary annuity Option 2

$= FV_{m-1} - FV/(1+i_e)$ FV* case, annuity due, Option 1

$= FV_m - FV/(1+i_e)$ FV* case, annuity due Option 2

$= FV_m + PMT - FV$ FV* case, ordinary annuity Option 1

$= FV_m(1+i_e) - FV$ FV* case, ordinary annuity Option 2

N.B. Values m and n are different for Options 1 and 2

The interest paid during each year of amortization is determined by the difference between the ending and beginning balances plus the sum of the payments for the year.

INT = (NP*PMT) + PV + FV

LPAS Program - Line by Line Analysis

LBL LPAS - Mainline - First Section

001 store ED and IP dates. Print separator line
007 set flag F06 (FV* case) if: |FV|>|PV|
014 calculate term (n). Print line 1 (PV)
026 Format data. Print line 2 (n)(PMT)
038 Format date. Print line 3 (%I)(FV)
044 If PF>24 set Flag 07 (calendar year basis)
050 Calculate day number of effective date (ED#)
053 Calculate day number of first PMT date (IP#)
056 Develop number of days from ED thru IP date (s)
058 Develop number of day from IP thru year end
069 Develop number of days in normal PMT period (d)
079 Adjust s for end of period payments (s)
081 Develop number of payments in first year
086 If PMT = 0, set s = 0
090 If s ≠ 0, append *. Print line 4 (ED)(IP)
095 Develop PV_e to adjust for shifted IP date (PV_e)
107 Save IP year (YY) and calculate payment (PMT)
113 --FIRST STOP--

At this stop the calculated periodic payment may be accepted or the original or a modified payment can be entered and stored by pressing key D before selecting an amortization option (H or I).

Subroutines
LBL 01 Reformat date for CJ and load print buffer
LBL 02 Calculate day number using 30/360 convention
LBL 03 Calculate day number using CJ (calendar basis)
LBL 04 Format month (MM) and day (DD) for printing
LBL 05 Display control -
06 - and column format subroutine
LBL 07 Execute specified FI routine
LBL 08 Fill buffer with specified character -
09 - and printer separator line
LBL D Store PMT in R04
LBL J Transfer control to FI and display status

- Mainline - Second Section
LBL H Option 1 - If PMT≠0, set flag F07 (set=opt. 1)
LBL I Option 2
205 Calculate new (n). If n=0 use original n (n)
213 Select (n): option 1=original option 2=new
216 Calculate FV and modify to -
LBL 10 - develop final payment (PMT_f)
LBL 11 Store final PMT_f Print separator line
263 Format data. Print line 5 (n-1)(PMT)
269 Format data. Print line 6 (PMT_f)
278 --SECOND STOP--

At this stop the amortization schedule calculation may be selected by pressing key E (for PV* cases only), or control may be returned to the FI program via key J.

Subroutine
LBL 12 Format control subroutine

LBL E - Mainline - Third Section - Amortization
284 If FV* case stop and display "FV* ?" (invalid)
288 Print separator line. Print heading line
294 Reduce payment count by number 1st yr payments
LBL 13 Develop interest for year -
14 - and calculate ending balance
LBL 15 ΣINT and format data. Print amortization line
346 Load stack for review and stop if FC?21
351 --AMORTIZATION YEAR STOP--

If flag F21 is cleared this stop will occur after the amortization calculations have been made for each year. Amortization data is available in the stack.

352 If not final year, update year & payment count
LBL 16 End routine Print total (**)(ΣINT)
384 END

Other LPAS program technical details:

Global Label: LPAS
Local Labels: D,E,H,I,J, and 01-16
Byte Count: 655 (requires one memory module)
Size Required: SIZE=014
ROM Routines called: CJ , CP , FI
Subroutine Levels: 3
Flags Used: LPAS - 06,07,21,28,29
FI - 08, 09, & 10
CJ - 10
CP - 29 & 40

Data Registers Used:

R00: multi use store	R07: i as decimal
R01: n term	R08: CF compounding freq.
R02: %I as percentage	R09: PF payment frequency
R03: PV present value	R10: multi use store
R04: PMT periodic pmt	R11: multi use store
R05: FV future value	R12: multi use store
R06: IND addr.	R13: multi use store

Status Registers: none used
Alpha Registers: all used
Σ REG: not used
Peripherals: printer recommended but not required
Stack Usage: I/O see program description
Execution Time: variable

APPLICATION PROGRAM FOR: **FI**

```
01♦LBL "LPAS"
02 STO 10
03 X<>Y
04 STO 00
05 0
06 XEQ 08
07 CF 06
08 RCL 03
09 ABS
10 RCL 05
11 ABS
12 X>Y?
13 SF 06
14 1
15 XEQ 07
16 ASTO X
17 "P"
18 FS? 06
19 "F"
20 "├V*"
21 ARCL X
22 "├  PV"
23 RCL 03
24 XEQ 05
25 ADV
26 XEQ 12
27 RCL 01
28 ENTER↑
29 INT
30 ARCL X
31 -
32 X≠0?
33 "├+"
34 "├ PMTS"
35 RCL 04
36 XEQ 05
37 ADV
38 FIX 3
39 ARCL 02
40 "├% FV"
41 RCL 05
42 XEQ 05
43 ADV
44 XEQ 12
45 CF 07
46 24
47 RCL 09
48 X>Y?
49 SF 07
50 "ED "
51 RCL 00
52 XEQ 01
53 X<> 10
54 "├ IP "
55 XEQ 01
56 ST- 10
57 STO 00
58 FIX 2
59 SF 28
60 SF 29
61 1
62 ST+ 11
63 STO 12
64 CLX
65 STO 13
66 XEQ 02
67 RCL 00
68 -
69 360
70 ENTER↑
71 6
72 FC?C 07
73 CLX
74 +
75 RCL 09
76 /
77 INT
78 STO 13
79 FC? 09
80 ST+ 10
81 +
82 LASTX
83 /
84 INT
85 STO 12
86 RCL 10
87 RCL 04
88 X≠0?
89 X<>Y
90 CHS
91 X≠0?
92 "├*"
93 FS? 21
94 PRA
95 CLA
96 RCL 08
97 RCL 13
98 *
99 /
100 RCL 09
101 *
102 RCL 07
103 LN1+X
104 *
105 E↑X
106 ST* 03
107 RCL 06
108 STO 11
109 4
110 XEQ 07
111 RND
112 STO 04
113 RTN
114♦LBL 01
115 INT
116 -100
117 STO 11
118 STO Z
119 X<>Y
120 STO 12
121 XEQ 04
122 INT
123 STO 13
124 XEQ 04
125 CHS
126 ST* 11
127 FRC
128 *
129 STO 06
130 10
131 X>Y?
132 "├0"
133 ARCL Y
134♦LBL 02
135 FS? 07
136 GTO 03
137 RCL 11
138 360
139 *
140 RCL 12
141 30
142 *
143 +
144 RCL 13
145 +
146 RTN
```

APPLICATION PROGRAM FOR: **FI**

```
147♦LBL 03
148 RCL 11
149 RCL 12
150 RCL 13
151 XROM "CJ"
152 RTN
153♦LBL 04
154 10
155 X>Y?
156 "├ "
157 ARCL Y
158 RDN
159 LASTX
160 -
161 *
162 "├-"
163 RTN
164♦LBL 05
165 FIX 2
166 9
167♦LBL 06
168 STO 06
169 X<>Y
170 SF 28
171 SF 29
172 FC? 21
173 RTN
174 ACA
175 XROM "CP"
176 CLA
177 RTN
178♦LBL 07
179 STO 06
180 XROM "FI"
181 RTN
182♦LBL 08
183 FC? 21
184 RTN
185 24
186 X<>Y
187♦LBL 09
188 ACCHR
189 DSE Y
190 GTO 09
191 PRBUF
192 RTN
193♦LBL D
194 STO 04
195 RTN
196♦LBL J
197 10
198 STO 06
199 GTO "FI"
200♦LBL H
201 RCL 04
202 X≠0?
203 SF 07
204♦LBL I
205 RCL 01
206 INT
207 STO 10
208 1
209 XEQ 07
210 INT
211 X=0?
212 RCL 10
213 FC? 07
214 STO 10
215 RCL 10
216 1
217 FS? 07
218 ST- 10
219 FS? 07
220 FC? 09
221 CLX
222 -
223 STO 01
224 RCL 07
225 1
226 +
227 STO 13
228 RCL 05
229 STO 00
230 5
231 XEQ 07
232 RCL 00
233 STO 05
234 FC? 06
235 CLX
236 STO 00
237 X<>Y
238 RCL 13
239 FS? 09
240 ST/ 00
241 X<>Y
242 FC? 09
243 GTO 10
244 RCL 00
245 -
246 GTO 11
247♦LBL 10
248 FC? 07
249 *
250 RCL 00
251 FC? 06
252 CLX
253 -
254 RCL 04
255 FC? 07
256 CLX
257 +
258♦LBL 11
259 RND
260 STO 13
261 1
262 XEQ 08
263 XEQ 12
264 ARCL 10
265 "├ PMTS"
266 RCL 04
267 XEQ 05
268 ADV
269 "+FINAL PMT"
270 RCL 13
271 XEQ 05
272 ADV
273 RCL 12
274 RCL 10
275 RCL 04
276 RCL 13
277 FIX 2
278 RTN
279♦LBL 12
280 FIX 0
281 CF 28
282 CF 29
283 RTN
284♦LBL E
285 "FV* ?"
286 FS? 06
287 PROMPT
288 1
289 XEQ 08
290 "YR  INTEREST "
291 "├ENDING BAL"
292 FS? 21
```

Listing continued on page 158.

Listing continued from 157.

BAR CODE ON PAGE 480

APPLICATION PROGRAM FOR: FI

```
293 PRA
294 CF 07
295 CLA
296 CLX
297 X<> 12
298 RCL 10
299 X<=Y?
300 SF 07
301 X<>Y
302 STO 01
303 -
304 STO 10
305+LBL 13
306 XEQ 12
307 RCL 11
308 10
309 X>Y?
310 "⊦0"
311 ARCL Y
312 RCL 03
313 RCL 04
314 RCL 01
315 *
316 +
317 STO 00
318 CLX
319 STO 05
320 FC? 07
321 GTO 14
322 RCL 13
323 ST+ 00
324 GTO 15
325+LBL 14
326 5
327 XEQ 07
328 FIX 2
329 RND
330 ST+ 00
331 STO 05
332 CHS
333 STO 03
334+LBL 15
335 FIX 0
336 RCL 00
337 RND
338 ST+ 12
339 8
340 XEQ 06
341 RCL 05
342 RND
343 8
344 XEQ 06
345 ADV
346 RCL 12
347 RCL 05
348 RCL 00
349 RCL 11
350 FC? 21
351 STOP
352 FS?C 07
353 GTO 16
354 1 E2
355 RCL 11
356 1
357 +
358 X=Y?
359 -
360 STO 11
361 RCL 10
362 STO 01
363 RCL 09
364 ST- 10
365 X<=Y?
366 STO 01
367 -
368 X<=0?
369 SF 07
370 GTO 13
371+LBL 16
372 "**"
373 RCL 12
374 8
375 XEQ 06
376 11
377 FS? 21
378 SKPCHR
379 ADV
380 "**"
381 ASTO X
382 FIX 2
383 .END.
```

APPLICATION PROGRAM 2 FOR FI

FAST - Reducing Interest Solution Time

When the solution for interest is required for PMT≠0, LBL 02 of FI produces an initial guess for the interest which is supplied to the iterative loop starting at LBL 06. In most cases the LBL 02 guess is usually "close" (in the mathematical sense) to the actual solution insuring that the interest solution is found in a reasonably short time.

Unfortunately, there will always exist a problem which will cause the LBL 02 guess to be far enough away from the actual solution to cause the execution time to be long. The optional routine presented below will provide an initial guess which tends to be "closer" to the actual solution than that provided by LBL 02, allowing a shorter execution time for most problems.

In use, the optional routine is executed in RAM memory and produces an initial guess for the interest. The guess is stored in register R07, and control of the calculator is transferred from the FAST routine to LBL 06 of the ROM program FI.

For the condition when PMT=0, the routine transfers to LBL 09 of the ROM program for an explicit solution. When solving for n, PV, PMT, or FV, the ROM is used in the usual manner. Don Dewey (5148) produced the mathematical expressions and wrote the program.

LBL FAST INSTRUCTIONS

1. Load the routine below into the calculator memory.

2. Go to LBL FI in the ROM.

3. Select desired status and enter known variables in the usual manner.

4. Either a) or b):

   a) solve for n, i, PV, PMT, or FV in the usual manner.

   b) Execute FAST to solve for interest using the optional routine. Do not use LBL B. The interest value is returned in the usual manner.

5. Repeat as needed from step 2.

APPLICATION PROGRAM FOR: FI

```
01+LBL "FAST"
02 9
03 STO 06
04 RCL 04
05 X=0?
06 GTO "FI"
07 6
08 STO 06
09 RCL 05
10 RCL 04
11 RCL 01
12 *
13 -
14 LASTX
15 RCL 05
16 +
17 RCL 03
18 +
19 RCL 01
20 RCL 03
21 *
22 X≠0?
23 /
24 ABS
25 STO 07
26 X<>Y
27 RCL 01
28 1
29 -
30 X↑2
31 RCL 04
32 *
33 RCL 05
34 -
35 RCL 03
36 +
37 3
38 *
39 /
40 ABS
41 RCL 05
42 X=0?
43 GTO "FI"
44 RCL 04
45 *
46 X>0?
47 GTO "FI"
48 RDN
49 STO 07
50 GTO "FI"
51 .END.
```

EQUATIONS USED IN FAST ROUTINE

If PMT*FV < 0 then FV case.
If PMT*FV >= 0 then PV case.

1. PV CASE:

$$i_0 = \left| \frac{n*PMT + PV + FV}{n*PV} \right|$$

Problem valid only if PV*PMT < 0.

2. FV CASE:

a) For PV≠0:

$$I_0 = \left| \frac{FV - n*PMT}{3*[(n-1)^2*PMT + PV - FV]} \right|$$

b) For PV=0:

$$I_0 = \left| \frac{FV + n*PMT}{3*[(n-1)^2*PMT + PV - FV]} \right|$$

FORMULAS USED IN FI

The basic financial equation used in this program was first reported in the Hewlett-Packard Journal of October 1977 (Ref. 3) where the description of its implementation in the HP-92 Financial Calculator was given. In this unique equation, all five financial variables (n, i, PV, PMT, FV) are accounted for, using the simple rule that money paid out is considered negative in sign, while money received is considered positive in sign.

The equation from page 23 of Ref. 3, is:

(1) $PV*(1+i)^n + PMT*[(1+i)^n - 1]/i + FV = 0$

Ordinary Annuity and Annuity Due Selection

In its present form, equation (1) is suitable for the ordinary annuity condition, when payments are made at the end of each period. To enable (1) to solve the annuity due condition when payments are made at the beginning of each period, a small modification is required. When this modification is added, equation (1) becomes:

(2) $PV*(1+i)^n + PMT*(1+iX)*[(1+i)^n - 1]/i + FV = 0$

where X=0 for ordinary annuity condition
X=1 for annuity due condition

When flag F09 is cleared, the ordinary annuity condition is selected. When flag F09 is set, the annuity due condition is selected. Flag F09 is toggled by LBL d.

With a simple algebraic rearrangement, (2) becomes:

(3) $[PV+PMT(1+iX)/i][(1+i)^n-1] + PV + FV = 0$

or

(4) $(PV + C)A + PV + FV = 0$

where

(5) $A = (1+i)^n - 1$

(6) $B = (1+iX)/i$

(7) $C = PMT*B$

The form of equation (4) simplifies the calculation procedure for all five variables, which are readily solved as follows:

(8) $n = LN[(C-FV)/(C+PV)]/LN(1+i)$

n is solved using LBL 01

(9) $i = [FV/PV]^{1/n} - 1$

For PMT=0, i is solved using LBL 09

For PMT≠0, i must be solved by iteration

(10) $PV = -[FV + (A*C)]/(A+1)$

PV is solved using LBL 03

(11) $PMT = -[FV + PV(A+1)]/(A*B)$

PMT is solved using LBL 04

(12) $FV = -[PV + A(PV + C)]$

FV is solved using LBL 05

Solution of Interest When PMT≠0

To solve for interest i when PMT≠0, an iterative technique must be employed, as equation (1) cannot be explicitly solved for i. This program uses Newton's Method, using exact expressions for the function of i and its derivative. The expressions are:

(13) $i_{k+1} = i_k - f(i_k)/f'(i_k)$

where

(14) $f(i) = A(PV+C) + PV + FV$

(15) $f'(i) = n*D*(PV+C) - (A*C)/i$

where

(16) $D = (1+i)^{n-1}$

(17) $= (A+1)/(1+i)$ as calculated by LBL 06

The iterative interest solving loop using equations (13), (14), and (15) starts at LBL 06.

Starting Guess For Interest

To solve for interest using Newton's Method, an initial starting guess must be provided. The program uses the following expression to provide the initial guess, i_0:

(18) $$i_0 = \left| \frac{PMT}{|PV| + |FV|} \right| + \left| \frac{|PV| + |FV|}{n^3*PMT} \right|$$

The closer the initial guess i_0 is to the actual solution i, the greater is the probability that the required solution will be obtained, and the shorter is the execution time.

Further Program Refinements

As well as being able to select either an ordinary annuity or annuity due situation, the program also enables solutions to be obtained when

B. PMT≠0

a. Calculate i_0 using equation (18) and LBL 02.
b. Calculate A, A+1, B, and C using equations (5), (6), (7), and LBL 08.
c. Calculate iterative solution, using equations (13), (14), and (15) and LBL 06.
d. Exit test (i) if error is too large, back to b above
(ii) if error acceptable, continue
e. Calculate i, using equations (21) or (22) and LBL 11.
f. Store i in R02 and halt.

3. Solving for PV

a. Calculate i_e using equations (19) or (20) and LBL 07.
b. Calculate A, A+1, B, and C using equations (5), (6), and (7) and LBL 08.
c. Calculate PV using equation (10) and LBL 03
d. Store PV in R03 and halt

4. Solving for PMT

a. Calculate i_e using equations (19) or (20) and LBL 07.
b. Calculate A, A+1, B, and C using equations (5), (6), and (7) and LBL 08.
c. Calculate PMT using equation (11) and LBL 04
d. Store PMT in R04 and halt.

5. Solving for FV

a. Calculate i_e using equations (19) or (20) and LBL 07.
b. Calculate A, A+1, B, and C using equations (5), (6), and (7) and LBL 08.
c. Calculate FV using equation (12) and LBL 05
d. Store FV in R05 and halt.

LINE BY LINE ANALYSIS OF FI

Lines 01-02 provide access to any FI subroutine that begins with a numeric label.

Lines 03-14 clear the financial registers R01-R05 and set CF=PF=1 in R08 and R09. Line 14 is a jump to the status display.

Lines 15-18 toggle flag F08 which controls Continuous/Discrete compounding. Line 018 is a jump to the status display.

Lines 19-21 toggle flag F09 which controls the Begin/End switch.

Lines 22-32 are the status display which shows one of the codes CE, CB, DE, DB.

Lines 33-36 store the number of compounding periods in R08 = CF = compounding frequency.

Lines 37-40 store the number of payment periods in R09 = PF = payment frequency.

Lines 41-43 multiply the X-register by 12 before entering the LBL A routine.

Lines 44-48 either store n in R01 and stop, or drop through to line 049.

Lines 49-63 solve for n via formula (8) and store n in R01.

Lines 64-66 divide the X-register by 12 before entering the LBL B routine.

Lines 67-71 either store %I in R02 and stop, or drop through to line 072.

Lines 72-127 are the major part of the program which solves for %I. Line 79 tests whether %I can be calculated directly if PMT=0. Otherwise lines 73-90 compute the initial·guess for %I via formula (18). Lines 91-126 are the recurrence loop for formulas (13)-(17). Formula (14) is complete at line 103 and formula (15) is complete at line 117. Line 127 is a branch to complete the calculation of %I.

Lines 128-165 are a special subroutine. Line 148 provides access to a second entry point within the subroutine. Lines 129-147 calculate formula (19) or (20). Lines 148-165 calculate formulas (5), (6), and (7) which are constants used by other parts of the program.

Lines 166-175 calculate formula (9) for %I when PMT=0.

Lines 176-193 finish the calculation of %I and restore the rate by calculating formula (21) or (22).

Lines 194-198 either store PV in R03 and stop, or drop through to line 199.

Lines 199-208 calculate PV via formula (10) and store PV in R03.

Lines 209-213 either store PMT in R04 and stop, or drop through to line 214.

Lines 214-227 calculate PMT via formula (11) and store PMT in R04.

Lines 228-232 either store FV in R05 and stop, or drop through to line 233.

Lines 233-241 calculate FV via formula (12) and store FV in R05.

Line 242 is provided to allow a running program to stop so the program pointer is in ROM.

NUMERIC LABELS/FUNCTIONS IN THE FI PROGRAM

Although FI is a complete self-contained program, some users may wish to use some of FI's subroutines in their own programs. The following list gives a correspondence between numeric labels and subroutines to be called as part of FI programs. To call a subroutine from one of your own programs, first store the number corresponding to the desired function in data register R06. Then use the instruction XEQ "FI" as part of your program. The execution times in seconds for the various subroutines are in parentheses in the following list.

Numeric Label Number in R06	Keyboard Label	Subroutine Function	
00	e	Clear R01-R05 and store 1 in R08 and R09	(<1 sec.)
01	A	Solve for n	(3.5 sec.)
02	B	Solve for %i	(variable)
03	C	Solve for PV	(2.5 sec.)
04	D	Solve for PMT	(3.3 sec.)
05	E	Solve for FV	(3.2 sec.)
12	None	Serves only to restore keyboard functions to top rows of keys	

The following special comments apply to FI subroutines that would be called from other programs.

First note that labels 01-05 only solve for the indicated variables, whereas the keys A-E perform the double functions of either solving or storing values. If you need to store the value of a financial variable in one of R01-R05 then your program should do that directly. Subroutines for storing are neither necessary nor provided.

The purpose of label 12 at the end of the program is to allow a running program to stop so the program pointer is in ROM and the automatic local label key assignments of the FI functions on the top row of keys will be restored to those keys. Otherwise, a running program would normally stop and leave the program pointer in RAM which would make the top row key assignments "disappear".

Numeric labels 07,08,09 & 11 are intended to be internal subroutines within FI which would seem to perform no useful purpose outside of FI. However, the truly curious PPC member may be able to jump into the middle of FI by calling these routines as "hidden functions". See the line by line analysis for the purpose of these functions. Example 11 in the FI documentation uses label 07 in this manner.

REFERENCES FOR FI

1. W.L. Crowley and F. Rode, "A Pocket-Sized Answer Machine For Business and Finance," Hewlett-Packard Journal, May 1973.

2. R.B.Neff and L. Tillman, "Three New Pocket Calculators: Smaller, Less Costly, More Powerful," Hewlett-Packard Journal, November 1975.

3. Roy E. Martin, "Printing Financial Calculator Sets New Standards For Accuracy and Capability," Hewlett-Packard Journal, October 1977.

4. Greynolds, Aronofsky, Frame, "Financial Analysis Using Calculators," McGraw-Hill, 1980.

For anyone wishing to further his/her knowledge on the subject of financial analysis, and as it applies to calculators, Reference 4 is probably the most definitive book available to date. The 470-page volume assumes the reader has no previous knowledge of financial analysis, and commences at an elementary level, taking the reader through the theory and practice. The main subjects covered are:

a. Basic Concepts in Compound Interest
b. Simple Annuities
c. General Annuities
d. Continuous Compounding and/or Payments
e. Variable Cash Flows and Internal Rate of Return
f. Balloon Annuities Using Present Values
g. Special Applications

CONTRIBUTORS HISTORY FOR FI

Graeme Dennes (1757) is responsible for programming the accuracy enhancements in FI and for writing the first substantial version of FI, after analyzing the problems associated with the various formulas involved.

The addition of the general annuity capability and an improvement to the initial guess routine were produced by Don Dewey (5148). Don also tested and further debugged the program using literally thousands of test examples. There were more changes made until the program evolved to its final form. Cliff Carrie (834) suggested changes in the use and placement of the local numeric labels. Graeme Dennes and Don Dewey are to be credited with the lion's share of the work on FI, both in programming and in the documentation.

FINAL REMARKS FOR FI

The first dedicated hand-held financial calculator was the HP-80 (Reference 1). This was later followed by the HP-22 (Reference 2) and HP-27, although they never replaced the HP-80.

As users accepted the new calculators, it wasn't long before they began to demand more in facilities, capabilities and accuracy. These needs were readily satisfied by the HP-92 printing financial calculator (Reference 3).

The HP-92 article provided the inspiration and the starting point for the FI program by setting challenges for accuracy and execution times. It also made available for the first time a single unique financial equation which accounts for all five basic financial variables.

Commencing with this single equation, the FI program was conceived as a new, original approach to providing a highly accurate and fast financial program for the HP-41C calculator. Both the facilities and mathematical approaches of FI have not been used together in any previous HP calculator program. Financial programs are in other HP-41C ROM's but the capability and accuracy justify including the FI program as part of the PPC ROM.

By applying a reformulation and simplification to the mathematics of the new HP financial equation, the execution times and accuracy have been improved over all previous programs. The execution times when solving for interest rate have been reduced by the use of a simplified, although accurate, Newton's Method for fast convergence, coupled with a routine which produces an initial starting guess "close" to the exact solution.

Previous financial programs used standard instruction sequences to calculate terms of the form

$$(1+i)^n - 1$$

as described elsewhere. Often, several different terms of this type were used in one program. The calculation of this term created most of the error in those programs. The high accuracy of FI was achieved by calculating this term using the LN1+X and E↑X-1 instructions, and calculating it only once per solution.

Thanks to the Hewlett-Packard Company for allowing the reproduction of numerical examples from Reference 3.

FURTHER ASSISTANCE ON FI

NOTES

TECHNICAL DETAILS

XROM: 10, 63	FI	SIZE: 010

Stack Usage:
- 0 T: used
- 1 Z: used
- 2 Y: used
- 3 X: used
- 4 L: used

Alpha Register Usage:
- 5 M: 2 char. status
- 6 N: not used
- 7 O: not used
- 8 P: not used

Other Status Registers:
- 9 Q: not used
- 10 ⊦: not used
- 11 a: not used
- 12 b: not used
- 13 c: not used
- 14 d: not used
- 15 e: not used

ΣREG: not used

Data Registers:
- R00: not used
- R01: n
- R02: %I
- R03: PV
- R04: PMT
- R05: FV
- R06: function call #
- R07: %I as decimal
- R08: CF
- R09: PF

N.B. R02 used as temporary store when solving for interest.

Flag Usage:
- 04: not used
- 05: not used
- 06: not used
- 07: not used
- 08: set=continuous clear=discrete
- 09: set= BEGIN clear=END
- 10: set=view iterations clear=no display
- 25: not used

Display Mode: FIX 2 recommended

Angular Mode: not used

Unused Subroutine Levels: 4

Global Labels Called:

Direct	Secondary
none	none

Local Labels In This Routine:

A, B, C, D, E, H, I, J
a, b, c, d, e
00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11, 12

Execution Time: See NUMERIC LABELS section in the FI documentation

Peripherals Required: none

Interruptible? yes

Execute Anytime? no

Program File: FI

Bytes In RAM: 324

Registers To Copy: 47

Other Comments:

C36

CURVE FITTING

Gto (not xeq) CV (SIZE 027)

SOFTKEYS/INPUT SUMMARY (USER mode ON)

Softkey	Input/Function
a	$\sum$ - (subtract from summation)
e	Initialize
A	$\sum$ + (add to summation)
B	Solve equation type J, leaving:

Register	Value
X	b
Y	a
Z	r

C	Solve for Y value given X (using the equation type and coefficients last solved)
D	Solve for X value given Y (using the equation type and coefficients last solved)
E	Solve for best equation type, leaving:

Register	Value
X	J (best curve type)
Y	b
Z	a
T	r

OUTPUT SUMMARY

J (Eq. type)	Equation	Description
1	$y = bx + a$	linear
2	$y = a\ e^{bx}$	exponential (a > 0)
3	$y = b\ Ln(x) + a$	logarithmic
4	$y = a\ x^{b}$	power (a > 0)

PROGRAM FLAGS

Flag 08	curve type
Flag 09	curve type
Flag 10	$\sum$ - function

STORAGE REGISTER USE--See original PPC documentation, following

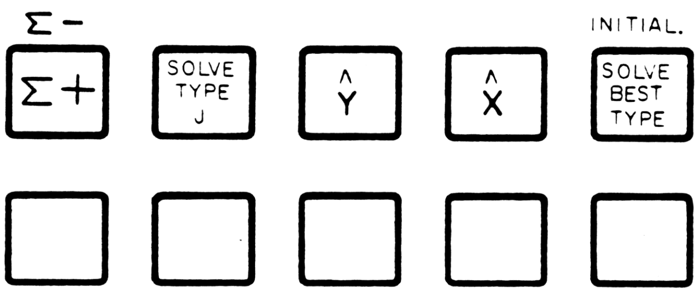
C36

CV - CURVE FIT

This program will determine a curve of best fit to a set of data points. The four standard curve types the program handles are:

1. Linear $y = b*x + a$
2. Exponential $y = a*e^{bx}$ $(a>0)$
3. Logarithmic $y = b*Ln(x) + a$
4. Power $y = a*x^b$ $(a>0)$

The program will compute the coefficients a and b in the equation of one of the above four curve types as well as compute a value r^2 called the coefficient of determination which is a measure of the goodness of fit. Once a set of data has been fit to a given curve type, a prediction may be made for the y-value given a new x-value, or a prediction may be made for the x-value given a new y-value. The functions available on the top row of keys on the keyboard are indicated in the following diagram.



These same functions are referenced in the examples and instructions by enclosing the name of the function on the key in square brackets [ ].

Example 1: Find the straight line which best fits the following data:
(1.1, 5.2), (4.5, 12.6), (8.0, 20.0),
(10.0, 23.0), (15.6, 34.0)
Then predict y when x=20 and predict x when y=25.

Plug the PPC ROM into the 41C and SIZE 027. GTO "CV" and go into USER mode. This puts the program counter in ROM and makes the curve fit functions available on the top row of keys. Pressing [INITIALIZE] will initialize the program. This clears registers R11 thru R24 so that a new set of data may be entered. In this example the 5 data points will be entered using the [Σ+] key. Key in each pair as x ENTER↑ y and push [Σ+].

Do:		See:
[INITIALIZE]		1.0000
1.1 ENTER↑ 5.2	[Σ+]	2.0000
4.5 ENTER↑ 12.6	[Σ+]	3.0000
8.0 ENTER↑ 20.0	[Σ+]	4.0000
10.0 ENTER↑ 23.0	[Σ+]	5.0000
15.6 ENTER↑ 34.0	[Σ+]	6.0000

All the data has now been entered and the parameters for the curve will be computed next. Since in this example we are interested in a straight line we key 1 (j=1) and push [SOLVE TYPE J]. When execution stops the values a, b, and r are available in the stack as:

Z:	r	and are also stored as	R08:	b
Y:	a		R09:	a
X:	b		R10:	r

For this example:

Z: r=0.999035140.
Y: a=3.499147270
X: b=1.972047542

The value r ranges between -1 and +1 and is a measure of how well the data fits the given curve type. The sign of r indicates whether the data is positively or negatively skewed. The closer r is to one of the extremes ±1 the better the fit. For this example the line has positive slope and the fit is extremely good (all sample problems seem to work well).

Having computed the values b and a (these remain stored in R08 & R09 until new data is input) we can determine new points along the line. Key in 20 and push [$\hat{y}$] for the predicted y-value. y=42.94009811 when x=20. Key in 25 and push [$\hat{x}$] for the predicted x-value. x=10.90280649 when y=25.

COMPLETE INSTRUCTIONS FOR CV

(Keyboard Operation)

1) Key GTO "CV", SIZE 027 and go into USER mode. The keyboard functions should now be now available on the top row of keys.

2) Press [INITIALIZE] to initialize the program. This step clears data registers R11 thru R24 inclusive. These registers will be used to accumulate the data for all four curve types. The display will show 1.

3) Key in the next data pair (x,y) as x ENTER↑ y and push [Σ+]. Repeat this step for all data pairs. The display will stop with a count of the number of the next data pair to be entered. This feature makes it possible to enter only the y-values when the x-values are consecutive integers which start counting from 1. In this case the display provides the x-values which need not be entered. If an improper data pair has just been input with the [Σ+] key, then immediately pressing R/S will delete the pair. Otherwise an improper or undesired data pair can be deleted by re-entering both x and y and pressing [Σ-].

4) As data pairs are entered it is possible that some x or y value is negative or zero. In these cases only one or two of the four curve types may be applied to the data. The four curve types and their respective equations are as follows:

Type J	Name	Equation
1	Linear	$y = b*x + a$
2	Exponential	$y = a*e^{bx}$ $(a>0)$
3	Logarithmic	$y = b*Ln(x) + a$
4	Power	$y = a*x^b$ $(a>0)$

If any x-values are negative or zero then only types 1 & 2 are feasible curves. If any y-values are negative or zero then only types 1 & 3 are feasible curves. If in any data pair both x and y are negative or zero then type 1 is the only feasible curve. The a coefficient must be positive for curve types 2 and 4.

5) After all data pairs have been input the next step is to select the desired curve type. This step can be accomplished in one of two ways. Under either option, the 41C should not be interrupted or else there is a possibility that the data registers will not be returned with their normal contents.

a) To fit a particular curve type, key in the number 1-4 for that type and press [SOLVE TYPE j]. The stack returns with:

Z: r	and these parameters	R07: j=curve type
Y: a	remain stored in	R08: b
X: b		R09: a
		R10: r

Step a) may be repeated at any time for any of the four curve types.

b) If all data input is positive then pressing [SOLVE BEST] will automatically choose the curve of best fit according to the curve type with largest absolute value of r. In this case the stack returns with:

T: r	and these parameters	R07: j=curve type
Z: a	remain stored in	R08: b
Y: b		R09: a
X: j=best curve type		R10: r

6) Predictions for new x or y values may be made only after step 5) has been completed. Predictions for new values are based on the settings of flags F08 and F09 which are automatically set during the fit process in step 5). The status of flags 8 and 9 for the four curve types are as follows.

		Flag 8	Flag 9
1	Linear	clear	clear
2	Exponential	set	clear
3	Logarithmic	clear	set
4	Power	set	set

In general the user need not be concerned with these flag settings, and F08 and F09 are not avalible for other use and must not be disturbed. To predict y given x, key in x and press [$\hat{y}$]. To predict x given y, key in y and press [$\hat{x}$]. In both cases the predicted value is left in the X-register.

7) New data may be added or deleted at any time via the [Σ+] or [Σ-] keys. However, step 5) must be performed after updating the data before any new predictions can be made using step 6). The parameters a and b are automatically destroyed after input of new data.

MORE EXAMPLES OF CV

Example 2: Determine whether the following data points are better suited for a logarithmic curve or a power curve. Then re-input the same x values and see how close the program predicts the y values.
(8, 2), (27, 3), (40, 3.2), (50, 3.5), (100, 4.1)

Do:	See:
[INITIALIZE]	1.0000
8 ENTER↑ 2 [Σ+]	2.0000
27 ENTER↑ 3 [Σ+]	3.0000
40 ENTER↑ 3.2 [Σ+]	4.0000
50 ENTER↑ 3.5 [Σ+]	5.0000
100 ENTER↑ 4.1 [Σ+]	6.0000

We will now try to fit a logarithmic curve type 3. Key 3 [SOLVE TYPE j]. The program returns with:

Z: 0.997148866 = r
Y: 0.267411352 = a
X: 0.822629796 = b

We next try to fit a power curve which is type 4. Key 4 [SOLVE TYPE j]. The program returns with:

Z: 0.995179948 = r
Y: 1.127479133 = a
X: 0.285458085 = b

Choosing the best r we would assume a type 3 logarithmic curve with the equation:

y = (0.822629796)*Ln(x) + 0.267411352

Since we just finished the power curve fit, the power curve parameters are still in the machine and hence we must go back and key 3 [SOLVE TYPE j] to return to the logarithmic parameters. Now we can predict the y's using the original x's. The predicted y-values are shown to four decimal places.

Do:	See:
8 [$\hat{y}$]	1.9780
27 [$\hat{y}$]	2.9787
40 [$\hat{y}$]	3.3020
50 [$\hat{y}$]	3.4856
100 [$\hat{y}$]	4.0558

Example 3: The following data fits either a linear or exponential curve. Determine which is more appropriate. (2, 12), (-1, 2), (3, 17), (5, 23) Then predict y when x = -10. After solving the above problem add the following as additional data points and resolve the same problem.
(-4, 0.713), (2.5, 10.93), (6, 47.53), (10, 254.95)

Do:	See:
[INITIALIZE]	1.0000
2 ENTER↑ 12 [Σ+]	2.0000
1 CHS ENTER↑ 2 [Σ+]	3.0000
3 ENTER↑ 17 [Σ+]	4.0000
5 ENTER↑ 23 [Σ+]	5.0000

Note that since one of the data points has a negative x the only possible curves to be fit under this program are linear or exponential. For a linear fit key 1 [SOLVE TYPE j]. The program returns:

Z: 0.997577939 = r
Y: 5.520000000 = a
X: 3.546666667 = b

For an exponential fit key 2 [SOLVE TYPE j]. The program returns:

Z: 0.958629344 = r
Y: 3.826163699 = a
X: 0.419923419 = b

Choosing the best r we find a linear fit is more appropriate. Since we just finished the exponential fit, the exponential parameters are still in the machine and hence we must go back and key 1 [SOLVE TYPE j] to return the linear parameters. Now key 10 CHS [$\hat{y}$] to predict y = -29.94666667 when x = -10.

We next add the additional data points and resolve the problem. (Do not clear the original data). The display should show 6 after entering the first new data pair below.

Do:	See:
4 CHS ENTER↑ 0.713 [Σ+]	6.0000
2.5 ENTER↑ 10.93 [Σ+]	7.0000
6 ENTER↑ 47.53 [Σ+]	8.0000
10 ENTER↑ 254.95 [Σ+]	9.0000

For a new linear fit key 1 [SOLVE TYPE j]. The data returned is:

Z: 0.765698771 = r
Y: 0.978958100 = a
X: 15.33154618 = b

For a new exponential fit key 2 [SOLVE TYPE j]. The data returned is:

Z: 0.993615263 = r
Y: 3.825595338 = a
X: 0.419945301 = b

Now choosing the best r we see that the new data reflects a change in the curve type. Since the exponential parameters should still be in the machine we can predict y when x = -10. Key 10 CHS [$\hat{y}$]. y = 0.057398396.

Example 4: Fit the best curve to the following set of data points.
(1, 2), (2, 2.828), (3, 3.464), (4, 4), (5, 4.472), (6, 4.899), (7, 5.292), (8, 5.657), (9, 6).

In this example the x-coordinates start counting from 1 and are consecutive integers. So we need only input the y-coordinates, but they must be in the proper order. The count in the display will serve as the x-coordinates.

Do:	See:
[INITIALIZE]	1.0000
2 [Σ+]	2.0000
2.828 [Σ+]	3.0000
3.464 [Σ+]	4.0000
4 [Σ+]	5.0000
4.472 [Σ+]	6.0000
4.899 [Σ+]	7.0000
5.292 [Σ+]	8.0000
5.657 [Σ+]	9.0000
6 [Σ+]	10.0000

Since all the data are positive we may use the best fit function to let the program find the best fit among all 4 curve types. Press [SOLVE BEST]. The contents of the stack when the program stops are:

T: 0.999999994 = r
Z: 1.999855865 = a
Y: 0.500043886 = b
X: 4.000000000 = best curve type

This indicates a power curve (type 4) where the equation is of the form:

$y = (2.00)*x^{0.50}$ (values rounded to 2 places)

APPLICATION PROGRAM 1 FOR CV

Curve fit solutions are often more meaningful when the points input are also plotted, superimposed on the plot of the "best fit" or selected equation type. The CVPL program will function exactly as CV functions and, after calculating the parameters a, b, r and r↑2, the program will stop with the prompt: "TO PLOT: R/S" To plot the equation calculated with the points input superimposed, simply press the R/S key. Nothing else need be done to obtain a plot. When accomplished in this way, the default situation, all numbers will be printed with 2 decimal places and the resulting plot will contain 50 plotted points. The detailed instructions include options to print other than 2 decimal places and to plot a smaller or greater number of points. The same key captions used by CV are used, plus the shifted keys b, c, and d for the optional features indicated.

Note that this program can also be used without the printer and will function essentially the same as CV but with the display labeling the points entered, showing deletions indentified as such, and labeling the parameters calculated.

The plotting program takes into account all possibilities: duplicate, identical points; almost identical points that would plot as identical; points with identical x-values but with significantly different y-values; individual single points. Any quantity of duplicate points can be handled. The points are plotted with 4 plotting characters as follows:

a. The equation of type J is plotted using a small square dot (box). One equation point is normally plotted before the first input point and after the last point. If the first input point is close to zero, it will be plotted first.

b. Individual single points are plotted with a large X.

c. Two (or more) essentially identical points are plotted with a double X, two small x's, one above the other.

d. Two (or more) points having essentially the same x-value but having different y-values are plotted with an asterisk located where the largest of the point's (based on x-value) plot should be. If desired the other points not shown for this value of x could be drawn in by hand or more points could be selected for the plot to separate very close x-values.

To simplify the program and reduce the number of registers required to store the data points, both the x- and y-value of a point are stored in one register, using a decimal point to separate them. This limits the magnitude and sign of the numbers to the following: data points must be nonzero, positive numbers and less than 1000 in magnitude. If you need to deal with larger numbers, shift all decimal points before entering them. Note: If the program is used without the printer, or by pressing "NO PLOT" with a printer, none of these restrictions apply and the "data error" message will not be encountered if you try to use negative or large numbers. See the valid use of negative entries in the CV instructions, however.

This program was developed originally as a modification to Gary Tenzer's curve fit program, "CFIT" in the PPC JOURNAL, V7N5P46, and was to be published in the JOURNAL as a stand alone program. The program was 691 steps in length (1414) bytes. With the CV routine (plus others such as the S? sorting routine) the plotting routine was completely re-written to utilize as many of the ROM routines as possible and the end result is presented below, significantly improved over my earlier version, with 439 steps and using 865 bytes (8 tracks on 4 cards). Most of Gary's displays and labeling are used in this program which partially account for the length of the program. I feel these extras are desirable, specially when using a printer.

Example 1 for CVPL: Use the same problem as Example 1 for CV . Find the linear equation for the following data: (1.1, 5.2), (4.5, 12.6), (8.0, 20.0), (10.0, 23.0), (15.6, 34.0).
Then predict y when x=20 and predict x when y=25.

Plug the PPC ROM in and using the card reader, read in all 8 sides of the program CVPL. Put the calculator in USER mode. Connect printer and put in MAN mode. Press initialize (shift E) and the display will tell you to SIZE 038 plus the number of points you plan to input. For this example SIZE 043 (=38 + 5 points). Press R/S to complete intialization of the program. See 1.00 in the display asking for the first point's values. First however, we will select 4 decimal places in the printout so key in 4 and press shift C (for the number of decimal places). See 1.00 again asking for the first point's values. Key in each point exactly as in the CV instructions by keying in X ENTER↑ Y [Σ+]. Keyboard functions assigned to keys are shown in square braces [ ] below.

Do:	See:
[Initialize]	"SIZE=38+ PTS"
XEQ "SIZE" 043	0.00 (size=38+5)
R/S	1.00 TONE 9
4 [No.Dec.Places]	1.00 TONE 9
1.1 ENTER↑ 5.2 [Σ+]	X1=1.0000, Y1=5.2000
	2.0000 TONE 9
.5 ENTER↑ 12.6 [Σ+]	X2=4.5000, Y2=12.6000
	3.0000 TONE 9
8.0 ENTER↑ 20.0 [Σ+]	X3=8.0000, Y3=20.0000
	4.0000 TONE 9
10.0 ENTER↑ 23.0 [Σ+]	X4=10.0000, Y4=23.0000
	5.0000 TONE 9
15.6 ENTER↑ 34.0 [Σ+]	X5=15.6000, Y5=34.0000
	6.0000 TONE 9

Since we want a linear curve, we key in 1 and push [SOLVE TYPE J]. When execution stops the following will be printed.

```
1: LIN
a=3.4991
b=1.9720
r=0.9990
r↑2=0.9981
```

In the calculator display see "TO PLOT: R/S"
If we now press R/S the plot will consist of 50 points. To select plot of 25 points, key in 25 and press [No.Pts.In Plot], the shifted B key. The same display will appear in the calculator (nothing is printed). Before plotting, we will first find the predicted y and x values asked for. Key in 20 and push [ŷ], the C key. Printed (and displayed) see:

"IF X = 20.0000, Y = 42.9401"

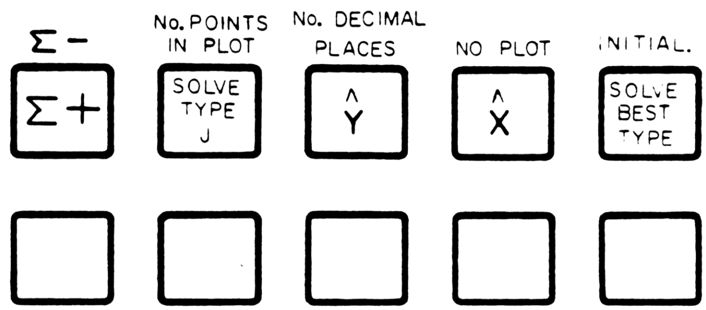
, in 25 and push [x̂], the D key and see:

"IF Y = 25.0000, X = 10.9028"

Now press R/S to plot the data. When the plotting is complete, wait for the BEEP before stopping the calculator.

The total time for this example, except for sizing the calculator was 4 min. and 25 sec. The primary consumer of time is normally the plotting, so the number of points selected greatly effects execution time. Often a short plot of 15 points is adequate.

After the BEEP has sounded the completion of the plot you can find other predicted values of x or y, select a different curve type, add points or delete points and see the effect on the new plot.



INSTRUCTIONS	INPUT DATA	KEYS	OUTPUT
1. Load cards, sides 1-8 in USER mode			0.00
2. Initialize		shift E	"SIZE=38+PTS"
3. If SIZE inadequate Otherwise go to step 4		XEQ "SIZE"	
4. Complete initialization		R/S	TONE 9 1.00
5. Optional - To print without plotting (including negative or larger numbers)		shift D	1.00
(Note: for new problem with plotting, must CF 24)			
6. Optional - Select no. of decimal places to be printed. Default is 2. Or key in n	n	shift C	TONE 9 1.00
7. Enter data point	X ENTER↑ Y	[Σ+]	Xj=--.---- Yj=--.---- TONE 9 # next point
Note: where x-values are same as displayed # of next point, input only Y and press A			
8. If point input is correct go to step 9. If incorrect, press R/S to delete the point just entered.		R/S	"**DELETE**" X=--.---- Y=--.---- TONE 9 # next point
To delete any previously entered point, re-enter exact X & Y values and press			

[Σ-] (same)

9. For each new point wait for TONE 9 and repeat step 7. (same as 7)

Note: Program will accept only positive values of X and Y in the range .01-999.99. For numbers outside of range shift decimal before entering. For a zero value use .01. "DATA ERROR" message will be displayed after an invalid entry. This note only applies with printer connected. Any values for X and Y will be accepted without a printer or after pressing "NO PLOT" with a printer. See CV instructions regarding acceptable negative numbers.

10. Calculate a,b,r,r^2:

a. For "best fit" based on largest ABS value of r: E (typical)

(Note: r & r^2 display correctly only on printer. Final caption not shown if printer not connected)

"1:LIN"
"a=--.----"
"b=--.----"
"r=--.----"
"r↑2=--.----"
"TO PLOT: R/S"

b. For selected type "J" curve Case:

1: Linear	1 B	
2: Exponential	2 B	(same)
3: Logarithmic	3 B	
4: Power	4 B	

NOTE: Step 10 must be accomplished after all data points have been entered before steps 11, 12, or 13 may be attempted.

11. Optional: select number of points to be plotted (points input plus equation points).
 a. Default value = 50 points no action required.
 b. Enter # of desired points

n shift B "TO PLOT: R/S"

12. Project y given x — $\hat{x}$ C "IF X = --.----" "Y = --.----" "TO PLOT: R/S"

13. Project x given y — $\hat{y}$ D "IF Y = --.----" "X = --.----" "TO PLOT: R/S"

14. To add additional points to same data, go to step 7.

15. Plot curve and data points R/S — Curve and points plotted

The following symbols are used:

- ▪ points on curve type "J"
- × data points, no duplicate X or Y value
- ⁑ 2 or more data points with the same X and Y values within the plotting tolerance.
- * 2 or more data points with same X-value but different Y-values. Only one of the points is plotted.

BEEP sounds after plot is complete

Note: after plotting wait for BEEP. Then you can add more points, delete points, predict new X or Y values, plot with a different number of points, calculate curve parameters with a different number of decimal places displayed or select a different curve type by going back to the above instructions.

Example 2 for CVPL: This example will demonstrate all four plotting characters described above and show how deletions and points can be added. The initial points are the following:

(70.00, 11.10), (10.40, 71.86), (22.30, 38.71), (10.50, 73.12), (40.90, 21.73), (4.20, 85.20) (100.30, 1.34), (41.30, 34.70)

Print with 4 decimal places and solve for the best fit curve. Then find the predicted value of y for X=35 and the predicted value of X for Y=100. Then plot using 30 points in the plot. Size for one additional point to be added. In the following the data in parentheses are not printed.

Do:	See:
[initialize]	("SIZE=38+PTS")
XEQ "SIZE" 047	(0.00)
R/S to complete initialization	(1.00 TONE 9)
4 [# dec. places]	(1.00 TONE 9)
70 ENTER↑ 11.1 [Σ+]	"X1=70.0000"
	"Y1=11.1000"
	(2.0000 TONE 9)
10.4 ENTER↑ 71.86 [Σ+]	"X2=10.4000"
	"Y2=71.8600"
	(3.0000 TONE 9)
22.3 ENTER↑ 38.71 [Σ+]	"X3=22.3000"
	"Y3=38.7100"
	(4.0000 TONE 9)
10.5 ENTER↑ 73.12 [Σ+]	"X4=10.5000"
	"Y4=73.1200"
	(5.0000 TONE 9)
40.9 ENTER↑ 21.63 [Σ+]	"X5=40.9000"
	"Y5=21.6300"
	(6.0000 TONE 9)

Y5 was entered in ERROR so to delete:

R/S	"DELETE"
	"X=40.9000"
	"Y=21.6300"
	(5.0000 TONE 9)

Now continue entering the correct values

40.9 ENTER↑ 21.73 [Σ+]	"X5=40.9000"
	"Y5=21.7300"
	(6.0000 TONE 9)
4.2 ENTER↑ 85.2 [Σ+]	"X6=4.2000"
	"Y6=85.2000"
	(7.0000 TONE 9)
100.3 ENTER↑ 1.34 [Σ+]	"X7=100.3000"
	"Y7=1.3400"
	(8.0000 TONE 9)
41.3 ENTER↑ 34.7 [Σ+]	"X8=41.3000"
	"Y8=34.7000"
	(9.0000 TONE 9)

Now push E for [SOLVE BEST]

"3: LOG"
"a=132.4456"
"b=-28.2822"
"r=-0.9812"
"r↑2=0.9627"
("TO PLOT: R/S")

Find the predicted values:

35 [$\hat{y}$] "IF X=35.0000"
"Y=31.8925"
100 [$\hat{x}$] ("TO PLOT: R/S")
"IF Y=100.0000"
"X=3.1494"
("TO PLOT: R/S")

Now select a 30 point plot:

30 [# points in plot] ("TO PLOT: R/S")
R/S to plot the data

After the BEEP sounds and the plotting is complete, add an additional point (71.1, 11.0), almost the same as point 1, and delete what appears to be the worst fitting point (22.30, 38.71).

71.1 ENTER↑ 11.0 [Σ+] "X9=71.1000"
"Y9=11.0000"
(10.0000 TONE 9)

22.30 ENTER↑ 38.71 [Σ-]
** DELETE **
"X=22.3000"
"Y=38.7100"
(10.0000 TONE 9)

Now again solve for the best fit.

[SOLVE BEST] "3: LOG"
"a=133.8645"
"b=-28.5171"
"r=-0.9858"
"r↑2=0.9719"
("TO PLOT: R/S")

We have slightly improved the fit to a log curve and the parameters a and b have of course changed. Now make a new plot by pressing R/S. After replotting the data, again find the predicted values of y if x=35 and x if y=100.

35 [$\hat{y}$] "IF X=35.0000"
"Y=32.4761"
100 [$\hat{x}$] "IF Y=100.0000"
"X=3.2789"

Looking at the plot, note the value of having the first input point be preceded by a point on the LOG curve. Note the double x at x=11 representing 2 almost identical points X2 and X4. The asterisk at x=41 means 2 or more points have essentially the same x-value but very different y-values. They are X5 and X8 and because X8 has a larger x-value than X5, the asterisk is plotted for Y8.

LINE BY LINE ANALYSIS OF CVPL

Lines 02-11 set up default conditions for 50 point plot and 2 decimal place printout. Lines 14-21 display next point to be input. Lines 22-30 are the delete routine using R/S. Lines 31-70 are the delete routine for later deletion of a point which first combines x and y in a single number as YYYYY.XXXXX after rounding to 2 decimal places, then searches stored points registers for the same point. When the point is found a copy of the last point stored is made in that register. Flag F05 prevents display of point number for a delete. Input of new points are added to CV statistical registers (71-118), then x and y values are checked for sign and magnitude and rounded to 2 decimal places and stored in YYYYY.XXXXX format. Lines 86-186 recall full numbers (not rounded) from CV for printing to number of decimal places selected and printout is formatted for input points, deleted x and y, and calculated parameters a, b, r, and r↑2. Lines 187-192 display plotting prompt "TO PLOT: R/S" only if printer connected, so program can be used without printer. Lines 193-200 store the barcoded input plotting symbols. Lines 201-217 exchange registers R07-11 with R33-37 using BE so data needed for CV statistical registers will be saved for later use, not lost when "PRPLOT" in printer ROM uses registers R07-11. Lines 218-236 use BX to find maximum and minimum y values of input points, then increase maximum and decrease minimum y by 25% of range to allow for equation points to be plotted outside of range of input points. Lines 237-241 make Ymin=0 if this value would have become negative after the 25% adjustment. These lines also determine the y-plotting increment used to see if 2 points have essentially same y-value. Lines 244-258 store "CRV" as the curve name for PRPLOT. The next function performed is a reverse of the left and right sides of the decimal point. Points are now stored as XXXXX.YYYYY (244) and S2 is used to sort the stored points to find maximum and minimum x and for faster plotting (246). Also calculated is the x-plotting increment using the range of x-values and number of points wanted in plot. If the x-minimum is smaller than plotting increment, lines 259-266 make the 1st point plotted the smallest x-value of the points; otherwise the x-minimum is set so one equation point will be plotted first. X-max made large enough that PRPLOT will never stop plot so one equation point can be plotted after largest x-values of input points (267-275). Stop routine initiated when one equation point beyond last point has been plotted. Lines 277-292 restore the statistical registers for CV by XEQ BE , then reverse stored points to original YYYYY.XXXXX format (284). Lines 293-296 reset the counter and "BEEP", ready for changes to data, etc. Flags 02 and 00 are used to determine if plotting is complete, lines 329-330. Routine to check stored points to see if they should be plotted at this x-value (297-323), checks +50% of plotting increment from this plotting point. If flag F03 is set (324) at least one point to be plotted here, and still checking for others. Plotting symbol to be used selected (340-360) and stored in R03 for "PRPLOT" to use for plotting. Where 2 input points have essentially the same x-value, checks to see if their y-values are also essentially the same (361-378). Flag F04 is set when 2 points have the same y-values, F01 is set when they have significantly different y-values (375-377). Plotting routines for the 4 curve types are in steps 379-399. The routine to reverse the left and right sides of the stored points (from the decimal point) is LBL 16, steps 400-419. Storage routines for optional selection of number of points in plot and number of decimal places in printout are in steps 425-435.
NOTE: The BLSPEC numbers for the plotting characters, if barcodes are not used, are:

box:	0,	0,	28,	28,	28,	0,	0
large X:	0,	34,	20,	8,	20,	34,	0
double x:	0,	0,	73,	54,	54,	73,	0
asterisk:	0,	20,	8,	62,	8,	20,	0

The ROM routine BI can also be used to create the equivalent of these BLDSPEC characters.

```
01♦LBL "CVPL"
02♦LBL e
03 4900
 STO 29
 2
06 STO 38
07 .
08 "SIZE=38+ PTS"
09 PROMPT
10 STO 06
11 XROM "CV"
12 39.999
13 STO 30
14♦LBL 12
15 RCL 18
16 1
17 +
18 CLA
19 ARCL X
20 TONE 9
21 PROMPT
22 DSE 30
23 SIN
24 SF 10
25 6
26 STO 06
27 RCL 08
28 RCL 09
29 XROM "CV"
30 GTO 08
31♦LBL a
32 SF 10
33 6
34 STO 06
35 RDN
36 XROM "CV"
 RCL 08
 RND
39 1 E3
40 /
41 STO 00
42 RCL 09
43 RND
44 1 E2
45 *
46 ST+ 00
47 RCL 30
48 1
49 -
50 STO 27
51 39.999
52 STO 30
53♦LBL 13
54 RCL IND 30
55 RCL 00
56 X=Y?
57 GTO 11
58 ISG 30
59 GTO 13
60♦LBL 11
61 RCL IND 27
62 STO IND 30
63 RCL 27
64 STO 30
65♦LBL 08
66 SF 12
67 "** DELETE **"
68 AVIEW
 SF 05
 GTO 09
 ♦LBL A
72 1
73 STO 06
74 RDN
75 XROM "CV"
76 RCL 08
77 XEQ 14
78 1 E3
79 /
80 STO IND 30
81 RCL 09
82 XEQ 14
83 1 E2
84 *
85 ST+ IND 30
86♦LBL 09
87 SF 12
88 FIX 0
89 "X"
90 FC? 05
91 ARCL 18
92 FIX IND 38
93 "⊢="
94 ARCL 08
95 AVIEW
96 PSE
97 FIX 0
98 "Y"
99 FC? 05
100 ARCL 18
101 FIX IND 38
102 "⊢="
103 ARCL 09
104 AVIEW
105 PSE
106 ADV
107 FC?C 05
108 ISG 30
109 GTO 12
110♦LBL 14
111 FIX 2
112 999.99
113 X<>Y
114 RND
115 X>0?
116 X>Y?
117 XEQ 17
118 RTN
119♦LBL C
120 SF 03
121♦LBL D
122 FIX IND 38
123 SF 12
124 STO 28
125 3
126 FC? 03
127 4
128 STO 06
129 RCL 28
130 XROM "CV"
131 "IF X="
132 FC? 03
133 "IF Y="
134 ARCL 28
135 AVIEW
136 PSE
137 "Y="
138 FC?C 03
139 "X="
140 ARCL X
141 AVIEW
142 PSE
143 ADV
144 GTO 07
145♦LBL B
146 ENTER↑
147 2
148 GTO 11
149♦LBL E
150 5
151♦LBL 11
152 FIX IND 38
153 SF 12
154 STO 06
155 RDN
156 XROM "CV"
157 "1: LIN"
158 ASTO 01
159 "2: EXP"
160 ASTO 02
161 "3: LOG"
162 ASTO 03
163 "4: PWR"
164 ASTO 04
165 CLA
166 ARCL IND 07
167 AVIEW
168 PSE
169 "a="
170 ARCL 09
171 AVIEW
172 PSE
173 "b="
174 ARCL 08
175 AVIEW
176 PSE
177 "r="
178 ARCL 10
179 AVIEW
180 PSE
181 "r↑2="
182 RCL 10
183 X↑2
184 ARCL X
185 AVIEW
186 ADV
187♦LBL 07
188 FC? 55
189 RTN
190 "TO PLOT: R/S"
191 CF 12
192 PROMPT
193 "♦♦♦"
194 ASTO 26
195 "♦α    "
196 ASTO 27
197 "×ΩAβΩ♦"
198 ASTO 28
199 "♦×♦"
200 ASTO 31
201 7.011
202 ENTER↑
203 33.037
204 XROM "BE"
205 RCL 30
206 INT
207 STO Y
208 1
209 -
210 1 E-3
211 *
212 +
213 STO 24
214 FRC
215 39
216 +
217 STO 25
218 XROM "BX"
219 .01
220 *
221 STO 00
222 X<>Y
223 .01
224 *
225 STO 01
226 -
227 ABS
228 .25
229 *
230 ST+ 01
231 ST- 00
232 RCL 00
233 X<0?
234 0
235 STO 00
236 STO 04
237 RCL 01
238 -
239 -62
240 /
241 STO 32
242 "CRV"
243 ASTO 11
244 XEQ 16
245 RCL 25
246 XROM "S2"
247 STO 30
248 RCL 24
249 1
250 -
251 RCL IND X
252 INT
253 RCL 39
254 INT
255 -
256 RCL 29
257 /
258 STO 10
259 RCL 39
260 XEQ 00
261 X<>Y
262 X≠Y?
263 X>Y?
264 -
265 ABS
266 STO 08
267 RCL 24
268 1
269 -
270 RCL IND X
271 XEQ 00
272 3
273 *
274 +
275 STO 09
276 XROM "PRPLOTP"
277♦LBL "CRV"
278 FC?C 00
279 GTO 11
280 7.011
281 ENTER↑
282 33.037
283 XROM "BE"
284 XEQ 16
285 RCL 24
286 INT
287 .999
288 +
289 STO 30
290 FIX IND 38
291 BEEP
292 STOP
293♦LBL 11
294 FS? 02
295 GTO 08
296 STO [
297♦LBL 06
298 RCL IND 30
299 XEQ 00
300 2
301 /
302 RCL [
303 +
304 X>Y?
305 GTO 09
306 FS? 03
307 GTO 10
308 GTO 08
309♦LBL 00
310 INT
311 1 E2
312 /
313 RCL 10
314 RTN
315♦LBL 09
316 X<>Y
317 RCL [
318 RCL 10
319 2
320 /
321 -
322 X<=Y?
323 GTO 11
324 FS? 03
325 GTO 10
326♦LBL 08
327 RCL 31
328 STO 03
329 FS?C 02
330 SF 00
331 RCL [
332 GTO IND 33
333♦LBL 11
334 FS? 03
335 GTO 08
336 SF 03
337 ISG 30
338 GTO 06
339 SF 02
340♦LBL 10
341 1
342 ST- 30
343 CF 03
344 RCL 26
345 FS?C 01
346 GTO 15
347 RCL 27
348 FS?C 04
349 GTO 15
350 RCL 28
351♦LBL 15
352 CF 04
353 STO 03
354 RCL IND 30
355 FRC
356 1 E3
357 *
358 ISG 30
359 RTN
360 RTN
361♦LBL 08
362 1
363 ST- 30
364 RCL IND 30
365 FRC
366 ISG 30
367 RCL IND 30
368 FRC
369 -
370 1 E3
371 *
372 ABS
373 RCL 32
374 ISG 30
375 SF 04
376 X<Y?
377 SF 01
378 GTO 06
379♦LBL 02
380 RCL 34
381 *
382 E↑X
383 RCL 35
384 *
385 RTN
386♦LBL 03
387 LN
388♦LBL 01
389 RCL 34
390 *
391 RCL 35
392 +
393 RTN
394♦LBL 04
395 RCL 34
396 Y↑X
397 RCL 35
398 *
399 RTN
400♦LBL 16
401 RCL 25
402 STO 30
403♦LBL 05
404 RCL IND 30
405 STO Z
406 FRC
407 1 E5
408 *
409 STO Y
410 RCL Z
411 INT
412 1 E5
413 /
414 ST+ Y
415 RDN
416 STO IND 30
417 ISG 30
418 GTO 05
419 RTN
420♦LBL 17
421 FC? 24
422 FC? 55
423 RTN
424 0
425 /
426♦LBL b
427 1
428 -
429 100
430 *
431 STO 29
432 GTO 07
433♦LBL c
434 STO 38
435 GTO 12
436♦LBL d
437 SF 24
438 .END.
```

FORMULAS USED IN CV

Linear (Type 1):

(1) $y = b*x + a$

(2) $Y = B*X + A$ where Y=y, X=x, A=a, B=b

(3) $x = (y-a)/b$

Exponential (Type 2):

(4) $y = a*e^{b*x}$ (a>0, y>0)

(5) $Y = B*X + A$ where Y=ln(y), X=x, A=ln(a), B=b

(6) $x = [\ln(y) - \ln(a)]/b$

Logarithmic (Type 3):

(7) $y = b*\ln(x) + a$ (x>0)

(8) $Y = B*X + A$ where Y=y, X=ln(x), A=a, B=b

(9) $x = e^{[(\ln(y) - \ln(a))/b]}$

Power (Type 4):

(10) $y = a*x^b$ (a>0, x>0, y>0)

(11) $Y = B*X + A$ where Y=ln(y), X=ln(x), A=ln(a), B=b

(12) $x = e^{[(\ln(y) - \ln(a))/b]}$

The curve fit program determines the least squares fit for the equation Y = B*X + A.

(13) $B = (\Sigma XY - (\Sigma X)(\Sigma Y)/n)/(\Sigma X^2 - (\Sigma X)^2/n)$

(14) $A = (\Sigma Y - (B)(\Sigma X))/n$

(15) $$r^2 = \frac{(\Sigma XY - \Sigma X \Sigma Y/n)^2}{((\Sigma X^2 - (\Sigma X)^2/n)(\Sigma Y^2 - (\Sigma Y)^2/n))}$$

The standard four curve type equations (1), (4), (7), and (10) are all special cases of Y = B*X + A which is equations (2), (5), (8), and (11). Note the distinction between upper and lower case letters. The user inputs and outputs are always in terms of the lower case letters. For example, the data input sists of pairs of the form (x,y) and the fficients the program determines are a and b. In all four cases b=B. The upper case letters are the quantities that the program uses to "conceptually" work with all four curve types simultaneously.

Routine Listing For: CV

```
01♦LBL "CV"
02 GTO IND 06
03♦LBL A
04♦LBL 01
05 CF 10
06♦LBL 06
07 STO 09
08 X<>Y
09 STO 08
10 ΣREG 13
11 FC? 10
12 Σ+
13 FS? 10
14 Σ-
15 RDN
16 RCL 08
17 ENTER↑
18 X>0?
19 LN
20 ST* Z
21 RCL 09
22 X>0?
23 LN
24 ST* Z
25 X<>Y
26 ΣREG 19
27 FC? 10
28 Σ+
29 FS? 10
30 Σ-
31 R↑
32 FS? 10
33 CHS
34 ST+ 12
35 R↑
36 FS? 10
37 CHS
38 ST+ 11
39 X<> Z
40 SIGN
41 ST+ L
42 RCL 08
43 RCL 09
44 X<> L
45 RTN
46 RCL 08
47 RCL 09
48♦LBL a
49 SF 10
50 GTO 06
51♦LBL B
52♦LBL 02
53 CF 08
54 CF 09
55 STO 07
56 2
57 X<Y?
58 SF 09
59 /
60 FRC
61 X=0?
62 SF 08
63 8
64 ST+ 07
65 XEQ IND 07
66 RCL 17
67 RCL 13
68 RCL 15
69 STO 09
70 *
71 RCL 18
72 /
73 -
74 STO 10
75 RCL 14
76 RCL 13
77 X↑2
78 RCL 18
79 /
80 -
81 STO Z
82 /
83 STO 08
84 RCL 13
85 *
86 ST- 09
87 X<>Y
88 RCL 16
89 RCL 15
90 X↑2
91 RCL 18
92 ST/ 09
93 /
94 -
95 *
96 SQRT
97 ST/ 10
98 XEQ IND 07
99 8
100 ST- 07
101 RCL 10
102 RCL 09
103 FS? 08
104 E↑X
105 STO 09
106 RCL 08
107 RTN
108♦LBL 10
109 RCL 11
110 X<> 17
111 STO 11
112♦LBL 13
113 RCL 21
114 X<> 15
115 STO 21
116 RCL 22
117 X<> 16
118 STO 22
119♦LBL 09
120 RTN
121♦LBL 11
122 RCL 12
123 X<> 17
124 STO 12
125♦LBL 14
126 RCL 19
127 X<> 13
128 STO 19
129 RCL 20
130 X<> 14
131 STO 20
132 RTN
133♦LBL 12
134 RCL 23
135 X<> 17
136 STO 23
137 XEQ 14
138 GTO 13
139♦LBL C
140♦LBL 03
141 FS? 09
142 LN
143 RCL 08
144 *
145 RCL 09
146 FS? 08
```

Listing continued on page 118.

Listing continued from page 117.

Routine Listing For: CV	

```
147 LN
148 +
149 FS? 08
150 E↑X
151 RTN
152♦LBL D
153♦LBL 04
154 FS? 08
155 LN
156 RCL 09
157 FS? 08
158 LN
159 -
160 RCL 08
161 /
162 FS? 09
163 E↑X
164 RTN
165♦LBL e
166♦LBL 00
167 11.024
168 XROM "BC"
169  E
170 RTN
171♦LBL E
172♦LBL 05
173 .
174 STO 25
175 4
176 STO 07
177♦LBL 07
178 RCL 07
179 XEQ B
180 RCL 25
181 RCL 10
182 ABS
183 X<=Y?
184 GTO 15
185 STO 25
186 RCL 07
187 STO 26
188♦LBL 15
189 DSE 07
190 GTO 07
191 RCL 26
192 XEQ 02
193 RCL 26
194 END
```

LINE BY LINE ANALYSIS OF CV

Line 02 provides access to all numeric labels within CV.

Lines 03-45 perform the function of inputting the next data point. This is the "sigma plus" subroutine. All summations are updated when this routine is called. These summations include sums of

$x, x^2, y, y^2, xy, \ln(x), \ln(x)^2, \ln(y),$

$\ln(y)^2, \ln(x)\ln(y), x\ln(y), y\ln(x).$

Lines 48-50 perform the "sigma minus" function for deleting a data pair. Note that flag 10 is set and then a jump is made into the "sigma plus" function.

Lines 51-107 calculate the parameters a, b, and r, for the curve type using formulas (13), (14), and (15). b is stored in R08, a is stored in R09 and r is stored in R10.

Lines 108-138 are a series of intertwined subroutines which are called in the curve fit process. These routines simply perform a series of register exchanges which place the proper sums in the sigma registers for the calculation of the parameters a, b, and r depending on the curve type selected. Since the exchange is performed twice (once in line 65 and once in line 98) all registers are returned to their original state.

Lines 139-151 perform the calculation of the predicted y value using formulas (1), (4), (7), and (10).

Lines 152-164 perform the calculation of the predicted value using formulas (3), (6), (9), and (12).

Lines 165-170 perform the initialization for the program by clearing the data registers used to accumulate the sums the program requires.

Lines 171-194 perform the function of selecting the best curve type among the four curve types that the program handles.

NUMERIC LABELS/FUNCTIONS IN THE CV PROGRAM

The following list gives a correspondence between numeric labels and subroutines to be called as part of CV programs. To call a subroutine function from one of your own programs, first store the number corresponding to the desired function in data register R06. Then use the instruction XEQ "CV" as part of your program. The execution times for the more significant subroutines are in seconds and are shown in parentheses.

Numeric Label Number in R06	Keyboard Label	Subroutine Function
00	e	Initialize/clear sigma registers (3.2 sec)
01	A	Sigma Plus function (2.2 seconds)
02	B	Solve Type J (1: 2.9 seconds) (2: 3.5 seconds) (3: 3.4 seconds) (4: 4.3 seconds)
03	C	Predict y
04	D	Predict x
05	E	Solve Best Curve Type (15.7 seconds)
06	a	Sigma Minus function provided F10 is set
09	none	provides simple RTN so no register exchange takes place
10	none	exchange register pairs R11&R17, R15&R21, R16&R22
11	none	exchange register pairs R12&R17, R13&R19, R14&R20
12	none	exchange register pairs R23&R17, R13&R19, R14&R20 R15&R21, R16&R22
13	none	exchange register pairs R15&R21, R16&R22
14	none	exchange register pairs R13&R19, R14&R20

Note that to use the Sigma Minus function you must set flag 10 before calling label 06.

Note that labels 09-14 are not represented by functions on the keyboard. They are only internal subroutines within CV which would seem to perform no useful purpose outside of CV. However, they are documented here only because they may provide the truly curious PPC member with some "hidden" functions that they will no doubt find some application for. Those who never read this will never know what they are missing.

REFERENCES FOR CV

1. Gary M. Tenzer (1816) PPC Journal " Curve Fitting Made Easy": I V5N1P29
II V5N3P9
III V5N6P29
IV V6N3P16
V V7N2P20
VI V7N5P46

2. Cuthbert, Daniel and Wood, Fred, "FITTING EQUATIONS TO DATA," John Wiley, New York, 1971

CONTRIBUTORS HISTORY FOR CV

Curve fitting has long been a topic of great interest among PPC members. Gary M. Tenzer (1816) has written several articles introducing this topic and is responsible among others for the development of curve fit programs for the HP-67 and the HP-41C.

The key equation for this program, Y = B*X + A, was taken from a program written by Keith Jarrett (4360). This one equation unifies the four types of curves fit and greatly simplifies previous programs that accomplish the same functions, but less elegantly. John Kennedy (918) did the final coding of the CV program. Bill Barnett (1514) provided CVPL.

FINAL REMARKS FOR CV

CV is not the most powerful curve fit program ever written for a programmable calculator. Limited space did not allow a more comprehensive routine which would fit up to 16 different types of curves. Only the standard 4 types are present in CV. CV can be extended by those wishing to take advantage of its subroutines.

FURTHER ASSISTANCE ON CV

NOTES

TECHNICAL DETAILS

XROM: 20, 08 | **CV** | SIZE: 027

Stack Usage:
- 0 T: used
- 1 Z: used
- 2 Y: used
- 3 X: used
- 4 L: used

Alpha Register Usage:
- 5 M: not used
- 6 N: not used
- 7 O: not used
- 8 P: not used

Other Status Registers:
- 9 Q: not used
- 10 ⊦: not used
- 11 a: not used
- 12 b: not used
- 13 c: not used
- 14 d: not used
- 15 e: not used

ΣREG: used

Data Registers:

R00: not used

R06: function call #	
R07: CV type	R18: n
R08: b, x	R19: Σ ln(x)
R09: a, y	R20: Σ ln(x)^2
R10: r	R21: Σ ln(y)
R11: Σ x ln(y)	R22: Σ ln(y)^2
R12: Σ y ln(x)	R23: Σ ln(x)•ln(y)
R13: Σ x	R24: n
R14: Σ x^2	R25: best r
R15: Σ y	R26: best type
R16: Σ y^2	
R17: Σ xy	

Flag Usage:
- 04: not used
- 05: not used
- 06: not used
- 07: not used
- 08: used CV type
- 09: used CV type
- 10: used Σ- function
- 25: not used

Display Mode: not used

Angular Mode: not used

Unused Subroutine Levels: 4

Global Labels Called:

Direct	Secondary
BC	none

Local Labels In This Routine:

A, B, C, D, E, a, e, 00, 01, 02, 03, 04, 05, 06, 07, 09, 10, 11, 12, 13, 14, 15

Execution Time: see NUMERIC LABELS in the CV documentation

Peripherals Required: none

Interruptible? yes

Execute Anytime? no

Program File: CV

Bytes In RAM: 292

Registers To Copy: 42

Other Comments:
The subroutine which fits a given curve type should be allowed to run to its full conclusion so that a double register exchange is properly completed

SV - SOLVE ROUTINE

This routine is a simple root solving program which will approximate a solution to an equation of the form: f(x)=0 using the Secant Method (a simplified form of Newton's Method). **SV** will find only one root at a time. The program requires an initial guess and an initial step size. The output is an x value which most closely makes f(x)=0. A flag may be set to display the successive approximations as they converge to the final answer. Convergence depends on the initial guess. Accuracy depends on the display setting.

Example 1: Use **SV** to find the two roots of the quadratic equation $x^2 + 2*x - 15 = 0$.

1. Insure a minimum SIZE 010.
2. Select a display setting of SCI 5. The routine will end when two successive approximations are rounded and found to be equal according to the display setting.
3. Set flag F10 to view the successive approximations.
4. The function on the left side of the equation must be programmed as a subroutine. The input to this subroutine, namely x, is assumed to be in the X-register and can be recalled from R07. The output from this subroutine, namely f(x), is also to be left in the X-register. For this example the following routine may be programmed in RAM program memory.

```
01*LBL "FX1"
02 X↑2
03 LASTX
04 2
05 *
06 +
07 15
08 -
09 RTN
```

5. The name of the global label "FX1" should be stored in R06. Go into alpha mode and key "FX1" ASTO 06.
6. The initial guess (nonzero) is to be entered along with an initial step size which may be zero or may be a small number compared to x. If a 0 step size is entered then the program will calculate the first step as 1% of the intial guess x. The program will also accept a non-zero value as the intial step size. For most applications and for this example use 0 as the initial step size. Choose x=7 as the initial guess for x. Key 0 ENTER↑ 7
7. XEQ "**SV**". The following consecutive approximations will be displayed.

```
7.00000+00
6.93000+00
3.98682+00
3.30024+00
3.03190+00
3.00115+00
3.00000+00
```

The final solution is returned after about 8 seconds. The true answer is exactly x=3. Since the above quadratic has two roots we will key in another initial guess to search for the other root.
This time we will guess x = -10. Key 0 ENTER↑ 10 CHS and XEQ "**SV**". The following sequence of numbers will be displayed.

```
-1.00000+01
-9.90000+00
-6.36872+00
-5.47003+00
-5.06539+00
-5.00360+00
-5.00003+00
-5.00000+00
```

The true answer this time is exactly x = -5.

COMPLETE INSTRUCTIONS FOR SV

(Keyboard Operations):

To calculate a root of f(x)=0:

1) Select SIZE. The minimum size required by **SV** is SIZE 010. The storage requirements for constants and coefficients associated with the function f(x) may dictate a larger size.

2) Select display setting. The display setting will generally determine when **SV** ends. If an exact solution is found then **SV** will end on the next iteration, otherwise **SV** rounds the last two approximations and ends if those rounded values are equal. In general, a display setting of SCI n will produce (optimistically) a solution correctly rounded to n+1 significant figures. A display mode of SCI or ENG is generally preferred to a FIX mode.

3) Specify display option. Flag 10 controls a display option. If F10 is set then the successively calculated approximations will be displayed. In this manner the user may view the progress of the iterations. This is especially recommended in the **SV** routine since **SV** may fail to converge if the initial guess is too far away from an actual root. Even when the values stabilize they may oscillate and it is a simple matter for the user to manually stop the program. If F10 is clear only the final x-value is returned.

4) Program the function f(x)=0. The function f(x) represented by one side of the equation must be programmed as a subroutine in program memory which starts with a global label name and ends with a RTN or END instruction. The label name should be of six or less characters and should be stored in R06. The input x and the output f(x) are both assumed to be in the X-register. The input x may also be recalled from R07. Since global label search begins from the bottom of program memory, it is advisable to place f(x) near the bottom of program memory. The f(x) program should not use registers R06-R09 and should not disturb flag F10.

5) Store the global label name from step 4) (six or less characters) in R06. The function subroutine call will be made via an XEQ IND 06 instruction.

6) Specify initial step size and initial guess. **SV** requires two input values. The first input is the step size which the program uses to determine the approximation for the derivative at the initial guess. The second input is the initial guess and is used as the starting x value by the program. The closer the initial guess is to the true solution the quicker the solution is found. Do not use 0 as an initial guess.

If zero is entered as the initial step size then the program will automatically calculate 1% of x as the acutal step size. A zero step size should prove adequate for the majority of applications. However, the user may enter a non-zero step size which may be finer or coarser than 1% of the initial x.

These two values are keyed in as:

step size ENTER↑ guess

7. XEQ " SV ". If F10 is set the program will display the consecutive approximations. If a printer is plugged in and turned on these approximations will be printed. The final solution will be left in the X-register when the program ends.

MORE EXAMPLES OF SV

Example 2: Solve $f(x) = x^3 - x - 1 = 0$.

1) SIZE 010 minimum
2) Set display mode as SCI 6.
3) Set flag F10 to view the approximations.
4) Key the following routine for f(x) into program memory:

```
LBL*FX2
ENTER↑
 X↑2
  1
  -
  *
  1
  -
 RTN
```

5) Key "FX2" in the alpha register and press ASTO 06.
6) Key in the initial step size as 0 and key in the initial guess as x=4. Key 0 ENTER↑ 4.
7) XEQ " SV ".
The following sequence of approximations will be displayed:

```
4.000000+00
3.960000+00
2.731772+00
2.226515+00
1.780222+00
1.522190+00
1.382556+00
1.333776+00
1.325185+00
1.324722+00
1.324718+00
```

The final solution is returned after about 13 seconds. The solution is correct to the digits displayed.

Example 3: Solve $f(x) = x^3 - 3*x^2 + 4 = 0$.

1) SIZE 010 minimum
2) Set display mode as SCI 4.
3) Set flag F10 to view the approximations.
4) Key the following routine for f(x) into program memory:

```
LBL*FX3
 X↑2
LAST X
  3
  -
  *
  4
  +
 RTN
```

5) Key "FX3" in the alpha register and ASTO 06.
6) Key in an initial step size of 0 and an initial guess of -2. Key 0 ENTER↑ 2 CHS
7) XEQ " SV ".
The following approximations will be displayed:

```
-2.0000+00
-1.9800+00
-1.3283+00
-1.1289+00
-1.0229+00
-1.0018+00
-1.0000+00
```

The final answer is returned after about 8 seconds. The true solution is exactly x = -1.

The following examples contain abbreviated instructions.

Example 4: The following equation is known as Kepler's equation:

x - E*SIN(x) - m = 0

and plays an important role in astronomy and astrodynamics (space travel). It can be programmed as follows:

```
LBL*FX4
ENTER↑
 SIN
RCL 01        (Note: R01=E)
  *
  -
RCL 02        (Note: R02=m)
  -
 RTN
```

Set RADIANS angle mode. The equation can now be solved for any values of E (R01) and m (R02). When E=0.2 and m=0.8 the function has only one root (9.64334-01) which can be found with any initial guess.

Example 5: SV can be used to find maxima and minima of a function by solving for zeros of its derivative. For example, if $f(x) = \sin(x)/x$ then the derivative $f'(x) = [x*\cos(x) - \sin(x)]/x^2$
Zeros of f' occur where the numerator is 0. Consequently, solutions can be found by applying SV to the following function which represents the numerator $g(x) = x*\cos(x) - \sin(x)$

```
LBL*GX5
ENTER↑
 COS
  *
RCL 07
 SIN
  -
 RTN
```

Assuming SCI 6 display mode and RADIANS angle mode. Store "GX5" in R06. Key in small initial guesses using a step size of 0 and SV will find the first few roots as 0, ±4.49341+00, ±7.72525+00. (Note that instead of taking the derivative algebraically, the ROM routine FD might be used).

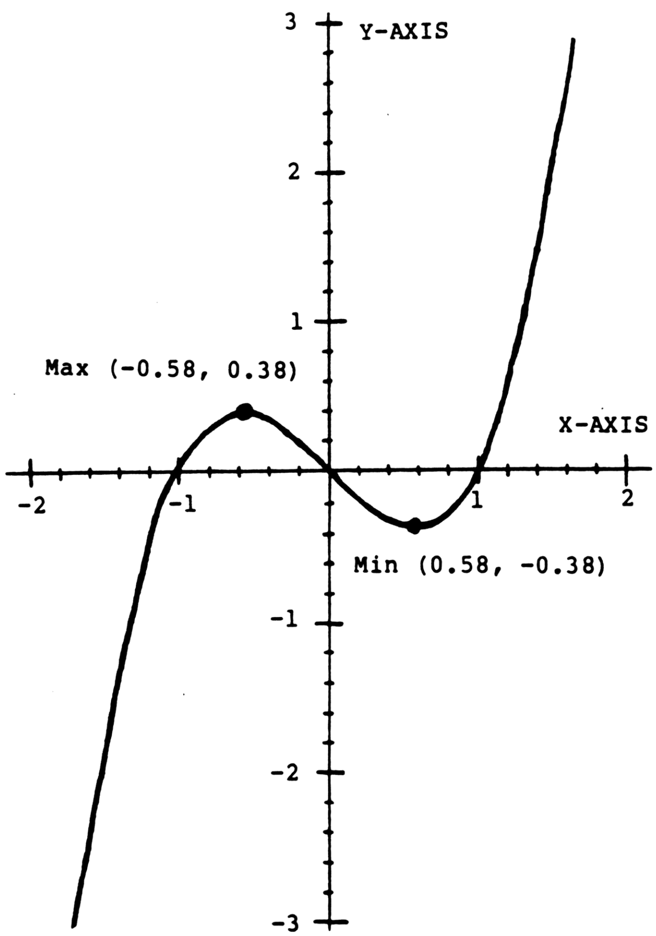
Example 6: Use SV to find all three roots of the cubic equation:

$$x*(x^2-1) = 0$$

The following should be programmed:

```
LBL*FX6
ENTER↑
X↑2
1
-
*
RTN
```

A sketch of the graph of this function will prove useful in understanding how the inital guesses determine which root is found.



GRAPH OF $Y = X^3 - X = X(X^2 - 1) = X(X+1)(X-1)$

The initial step size is 0 for each of the guesses suggested below. An initial guess greater than 0.6 for this example will find the root x=1.0, while a guess between -0.49 and +0.49 will find the root at x=0.0. However, guesses too close to the local peaks of the function at $x=\pm 1/\sqrt{3}$, where the slope of the tangent line is zero, lead to oscillations that fail to converge. Try an initial guess of x=0.58 to observe this behavior. This example also illustrates that the root found is not necessarily the one nearest to the initial guess. Try x=0.52 which finds the root x = -1.0.

Example 7: f(x) = x*LN(x) - 1.2 = 0

Display mode: SCI 6
Initial step size = 0
Initial guess = 3
See:

```
3.000000+00
2.970000+00
1.998929+00
1.902018+00
1.888327+00
1.888087+00
```

Result: 1.888087 after about 8 seconds

Example 8: f(x) = 3*x - COS(x) - 1 = 0

Display mode: SCI 6
Use RADIANS angle mode
Initial step size = 0
Initial guess = 2
See:

```
2.000000+00
1.980000+00
6.159990-01
6.078243-01
6.071024-01
6.071016-01
```

Result: 6.071016-01 after about 10 seconds

Example 9: $f(x) = x^2 + 4*SIN(x) = 0$

Display mode: SCI 6
Use RADIANS angle mode
Initial step size = 0
Initial guess = -4
See:

```
-4.000000+00
-3.960000+00
-2.210789+00
-2.037220+00
-1.946227+00
-1.934406+00
-1.933758+00
-1.933754+00
```

Result: -1.933754 after about 12 seconds

Example 10: $f(x) = x^4 - 26x^2 + 49x - 25 = 0$

Display mode: SCI 6
Initial step size = 0
Initial guess = 5
See:

```
5.000000+00
4.950000+00
4.310588+00
4.077156+00
3.927333+00
3.883035+00
3.876065+00
3.875777+00
3.875775+00
```

Result: 3.875775 after about 14 seconds

Continuing this same example for another root:
Initial step size = 0
Initial guess = -10
See:

```
-1.000000+01
-9.900000+00
-7.959395+00
-7.135530+00
-6.438967+00
-6.086301+00
-5.945260+00
-5.917701+00
-5.915863+00
-5.915842+00
```

Result: -5.915842+00 after about 14 seconds

FURTHER DISCUSSION OF SV

SV is not a sophisticated root solver and is subject to all the difficulties and error traps that confront all other root solvers. Limited space in the PPC ROM did not allow protection schemes to detect or rectify possible trouble areas. The method used, strictly speaking, is the Secant Method, however, it can be considered a form of Newton's Method where a numerical approximation is used for the derivative. A secant line is used to approximate the true tangent line. If SV fails to converge then another initial guess must be tried. SV can be effective as a subroutine in a program provided the user has knowledge of the range of appropriate values for the given function.

The display setting will help control the accuracy of the final result. When in SCI n display mode the final answer will (usually) be accurate to n+1 significant digits. However, sometimes this is not the case and the final answer will not be as accurate as the display setting would indicate. Every floating point operation in a computational process can give rise to rounding error which, once generated, may then be increased in subsequent operations.

For example, let $f(x) = x^2 - 6*x + 9$ and use SV to solve for f(x)=0. In FIX 9 the answer SV returns may be 3.000030072 which is accurate to five digits only. The true solution is a double root at x=3. Thus the display setting has not determined the accuracy in this example. This is basically caused by using only ten digits internally in the calculator, causing each and every calculation to have its solution rounded to ten digits. The root solver itself cannot then be held to ransom when the f(x) routine is affected by rounding errors.

This example highlights the action of SV when the secant line is horizontal, that is, when f(a)=f(b) where a and b are successive approximations. This situation may occur at multiple roots where the first derivative shares a root with the original function. (When the first derivative is zero the tangent line is horizontal). In general, do not select a display n value any larger than necessary. The use of SCI or ENG display modes are generally preferred to the FIX display mode. And do not blindly accept any solution given by this or any other root-solving program. Any potential real solution can be validated by applying the f(x) subroutine to see how close f(x) really is to 0.

Because the HP-34C calculator's SOLVE routine uses a similar method as SV (with many refinements), users are urged to study Reference 5. In that informative article some of the problems of root solving and a description of the mathematics of the secant method are discussed. Reference 1 provides a broad background to the subject. Further information may be found in most university and college libraries.

THE SELECTION OF A METHOD FOR SV

The oldest know method of root-solving is the Method of False Position, or Regula Falsi. Commencing with two estimates, lying on either side of the actual root, inverse linear interpolation is applied to produce a new estimate. Here, a linear function (a straight line) is used to approximate the true function f(x) over the interval of interest. This can be seen to be "reasonable" approximation so long as the interval is "small". As the two estimates must always straddle or bracket the root, convergence is always guaranteed. Of course, after calculating the new estimate, a decision has to be made as to which previous estimate is to be discarded.

The Method of False Position has a unity order of convergence, making the iteration time reasonably long. However, the solution is always obtained.

A method similar to False Position is the Secant Method. Although the mathematics of the two methods are identical, the difference lies in the fact that the Secant Method uses the approximations in strict sequence. Thus, the bracketing of the root is no longer necessary, and a secant is used to approximate the function f(x) over the interval of interest. Then, inverse linear interpolation is applied to produce the next estimate of the root. The Secant Method's order of convergence is approximately 1.62, which is higher than the Method of False Position, but convergence to the root is no longer guaranteed.

Both the Method of False Position and the Secant Method are in the class of two-point iterative methods, which, while the order of convergence is not high, nevertheless have high stability.

Another popular method derived from calculus using a Taylor Series is commonly known as the Newton-Raphson, or Newton's Method. In this method a tangent line to the function is used to determine the direction and amount of displacement to move from the current estimate to the new estimate. Newton's Method belongs to the class of one-point iterative methods. Newton's Method is the official and familiar name of tangent sliding philosophy, and has an order of convergence of two.

The mercurial properties of Newton's Method arise from its use of derivative information gathered at one point. Both the function and its derivative must be evaluated at each iteration. (This also results in a greater programming effort for two functions are really being evaluated). The time for an iteration is longer than the False Position and Secant Methods, which both require only one function evaluation per iteration. However, the convergence rate of Newton's Method is greater than both.

Comparison of Methods

A. Method of False Position.
   Advantages:
   1. Convergence is guaranteed.
   2. Only one function evaluation is needed at each iteration.
   Disadvantages:
   1. Low (unity) order of convergence.
   2. A decision is needed as to which estimate to discard to insure root-bracketing occurs.
   3. The root-bracketing requirement prevents its use at multiple, even-order roots.

B. The Secant Method.
   Advantages:
   1. Medium (approx. 1.62) order of convergence.
   2. Estimates are used in strict sequence, so no decision is needed on which estimate to discard.
   3. Only one function evaluation is needed at each iteration.
   4. Can be very stable.
   Disadvantages:
   1. Convergence is not always guaranteed.
   2. May have difficulty at multiple even-order roots.

C. Newton's Method.
   Advantages:
   1. High (2.0) order of convergence.
   Disadvantages:
   1. Convergence is not always guaranteed.
   2. Both the function and its derivative must be evaluated at each iteration, which increases the time per iteration.
   3. The derivative must be known explicitly.
   4. Can be very unstable.
   5. Has difficulty at multiple, even-order roots, where the function and its first derivative both have the same root.

Summary

From the above comparison, the Secant Method was considered to be the optimum algorithm, and was selected for SV . It combines a reasonable rate of convergence, a (usually) stable two-point step, uses the calculated approximations in strict sequence, and does not require evaluation of the derivative, dispensing with the need to provide the derivative explicitly.

ROOT SOLVING DIFFICULTIES - A PRIMER

One of the most frequently occuring problems in scientific work is to find the values of x for an expression f(x) which will make f(x)=0. Those values of x are called the roots of the equation f(x)=0. The function may be given explicitly as, for example, a polynomial, or as a transcendental function. In rare caes it may be possible to obtain the exact roots by algebraic manipulations. In general, however, we can hope to obtain only approximations to the roots relying on some iterative computational procedure to produce those approximations.

In the year 1225 Leonardo of Pisa studied the equation:

$$f(x) = x^3 + 2*x^2 + 10*x - 20$$

and was able to produce the root of 1.368808107. Nobody knows by what method Leonardo found this value, but it was a remarkable achievement for his time.

Simply put, all we require are those values of x which will make f(x)=0. It should be easy, so why all the fuss? The basic difficulty stems from the fact that our root solving methods tend only to use the function expression to numerically evaluate f(x) and have no analytical knowledge of the function. It they did, better starting guesses and better exit criteria could be selected. More importantly, the best root solving method to use for a particular function could be selected.

Easy though root solving may seem, and when coupled with one of the popular methods, e.g., Newton's Method, Secant Method, Bisection Method, it may come as a surprise to know that the search for the perfect root solver is no less difficult than the search for the Holy Grail! For every root finding method put forward, a situation can be provided which will cause the method to fail and deny us the solution.

What can be done? A root solver is simply a computational process which uses known facts (data) to calculate a better approximation to the root. The basic difference among root solving methods is the way in which the known facts are used to calculate the improved estimate.

If we know situations that cause our root solver to fail to find the root, we can provide assistance by enhancing the basic method with strategies to detect these problem situations and allow an escape from them. How many difficulties may confront a root solver? How many strategies do we design into it? One, five or one hundred? If we limit our root solver to solve only a specific class of problems we may be able to implement some strategies to overcome the typical problems encountered by that class.

Starting Values

All root solvers require starting values whether they be entered manually as in SV or use some set of values, e.g., 1 and 10. If the starting value is not "close" to the root, our selected method may step away from the root, making the problem worse.

Therefore the accuracy of the starting values (guesses) can be seen to be as important as the selected root solving method and its inbuilt strategies. Do we have strategies for determining an approximation to the root we seek? In some cases the answer is yes.

Exit Criteria.

When the root solver has located a root, it exits the iteration process, gives us the answer and stops. How does the root solver know when it really has found a root? Our inbuilt exit criteria must apply certain tests to the known facts and assess if a root has been found. Again, situations can be provided to fool the exit tests, and cause execution to stop when really no root has been found (i.e., no root exists).

Should we have several exit tests built into the program? Under what conditions should each be invoked? If only the answers were simple!

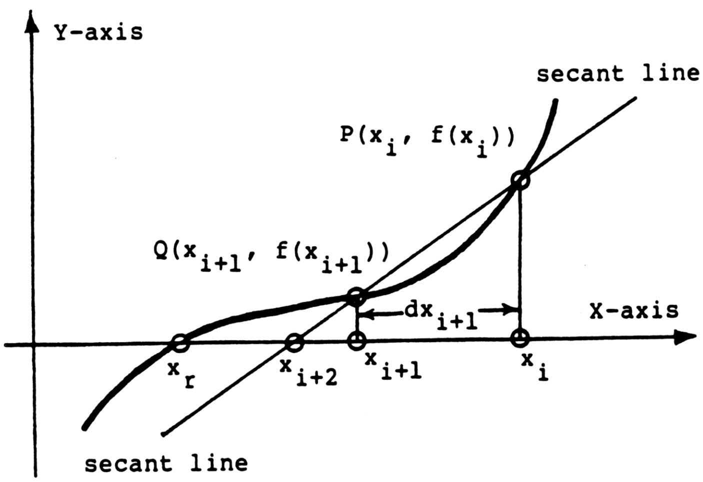
Numerical instability and round off errors.

Some root solvers can exhibit instability in certain conditions. Do we know how many of these conditions exist for any one root solver? How do we overcome these problems? As all floating point calculations are rounded to ten digits by the calculator a further difficulty arises due the propogation of these errors.

Because of the enormous difficulties which may confront a root solving process, **SV** has been written as an elementary routine, with no refinements or strategies included. It is left to the user to provide such strategies, by observing the behavior of the approximations, and taking action should divergence occur. A little experience with particular problems will provide guidance. Readers are urged to consult Reference 5.

FORMULAS USED IN SV

Let $f(x)$ be the function whose root x_r we desire. x_{i+1} and x_i are the two previous approximations.



GRAPH OF GENERAL FUNCTION WITH SECANT LINE SHOWING APPROXIMATION X_{i+2} FOLLOWING X_i AND X_{i+1}

The slope of the secant line through points P and Q is given by:

$$[f(x_{i+1}) - f(x_i)]/[x_{i+1} - x_i]$$

Letting $dx_{i+1} = x_{i+1} - x_i$ we have as the equation of the secant line through points P and Q:

$$y - f(x_{i+1}) = ([f(x_{i+1}) - f(x_i)]/dx_{i+1})*(x - x_{i+1})$$

Hence,

(1) $$x_{i+2} = x_{i+1} + dx_{i+2}$$

where

(2) $$dx_{i+2} = \frac{(dx_{i+1})*f(x_{i+1})}{f(x_i) - f(x_{i+1})}$$

The initial value x_0 is input by the user and dx_0 is usually taken as a small fractional part of x_0

For example, if we assume $f(x_{-1})=0$ and $dx_0=(.01x_0)$ then $x_1 = (.99)x_0$

Execution halts when x_{i+2} and x_{i+1} are rounded and found to be equal.

Routine Listing For: SV

```
 91*LBL C           108 ST* 09
 92*LBL "SV"        109 ST- 08
 93 STO 07          110 RCL 09
 94 E               111 RCL 08
 95 %               112 X≠0?
 96 RCL Z           113 /
 97 X=0?            114 STO 09
 98 X<>Y            115 X<> 07
 99 STO 09          116 ST+ 07
100 CLST            117 RND
101*LBL 04          118 RCL 07
102 RCL Z           119 RND
103 STO 08          120 X≠Y?
104 RCL 07          121 GTO 04
105 FS? 10          122 RCL 07
106 VIEW X          123 RTN
107 XEQ IND 06
```

LINE BY LINE ANALYSIS OF SV

Lines 91-101 initialize the program by storing the initial guess X_0 in R07 and store the initial step size in R09. Note that if the user has input 0 as the initial step size then lines 94 & 95 and lines 97 & (calculate and select the value $0.01*X_0$ as the actual step size.

Lines 101-121 are the main loop in the program. At LBL 04 X, Y, and T are assumed to be scratch and $f(X_i)$ is assumed to be in Z. The next approximation is calculated via formula (1) and is stored in R07 (line 116). Next, the two most recent approximations are rounded and tested for equality in line 120. A branch is then made back to LBL 04 unless the rounded values are equal.

Lines 122 & 123 recall the final solution and end the routine.

REFERENCES FOR SV

1. Forman S. Action, NUMERICAL METHODS THAT WORK, Harper and Row, New York, 1970
2. S. D. Conte and C. de Boor, ELEMENTARY NUMERICAL ANALYSIS, McGraw-Hill, 1972
3. John Kennedy (918) PPC JOURNAL "Method of Successive Bisections" V5N8P19. See also V6N5P10
4. Chris Stevens, PPC JOURNAL, V5N8P45, Sept.-Oct. 1978
5. William M. Kahan, "Personal Calculator Has Key To Solve any Equation f(x)=0", Hewlett-Packard Journal, December 1979.

CONTRIBUTORS HISTORY FOR SV

John Kennedy (918) wrote the SV program for the HP-41C from a previous HP-25 program. Graeme Dennes (1757) and Richard Schwartz (2289) made suggestions for improvements in the accuracy and overall program operation. Harry Bertucelli (3994) suggested register usage to allow SV to be used with IG. Graeme Dennes(1757) and Iram Weinstein (6051) contributed to the documentation of SV.

FINAL REMARKS FOR SV

SV needs improvement in almost all areas. SV is only a basic routine designed to be used primarily from the keyboard where the user may watch the convergence (or lack thereof) and take action to halt SV and make a new guess. SV lacks all of the sophistication of the SOLVE function on the HP-34C calculator.

FURTHER ASSISTANCE ON SV

NOTES

TECHNICAL DETAILS

XROM: 20, 10 — **SV** — SIZE: 010 minimum

Stack Usage:
- 0 T: used
- 1 Z: used
- 2 Y: used
- 3 X: used
- 4 L: used

Flag Usage:
- 04: not used
- 05: not used
- 06: not used
- 07: not used
- 08: not used
- 09: not used
- 10: displays successive approximations when F10 is set
- 25: not used

Alpha Register Usage:
- 5 M: not used
- 6 N: not used
- 7 O: not used
- 8 P: not used

Other Status Registers:
- 9 Q: not used
- 10 ⊦: not used
- 11 a: not used
- 12 b: not used
- 13 c: not used
- 14 d: not used
- 15 e: not used

Display Mode:
SCI n recommended controls accuracy

Angular Mode:
not used, but may be required by function

Unused Subroutine Levels:
4

ΣREG: not used

Data Registers:
- R00: not used
- R06: function LBL name
- R07: point x_i
- R08: $f(x_i)$
- R09: dx_i
- R10: not used
- R11: not used
- R12: not used

Global Labels Called:

Direct	Secondary
function LBL in R06	none

Local Labels In This Routine:
C, 04

Execution Time: variable, depends on function, display setting, and initial guess.

Peripherals Required: none

Interruptible? yes

Execute Anytime? no

Program File: IG

Bytes In RAM: 51

Registers To Copy: 43

Other Comments:

C56

BC - BLOCK CLEAR

This block clear routine may be used to store zeros in a block of registers. BC uses the complete form of the general block control word bbb.eeeII and can thus be used to clear blocks of consecutive registers or can be used to skip over registers within a block.

Example 1: Use BC to clear registers R05-R15.

Before clearing these registers we will first use the block increment routine BI to load them with data. (The PPC ROM routines BI and BV are extremely convenient for loading and viewing blocks of registers). Key 5.015 ENTER↑ 1 ENTER↑ and XEQ "BI". The consecutive integers from 1-11 should now be loaded in R05-R15 inclusively. To convince yourself, clear flags F09 and F10 and key 5.015 XEQ "BV". The block view routine BV should run through the registers and show the contents as described.

Now to clear these registers, simply key 5.015 and XEQ "BC". If you again use BV to view these registers you won't see anything because they have been cleared.

Example 2: Use BC to clear every 5th register starting with R07 and ending with R102.

Assuming you have the available memory and that the current size is at least 103, simply key in 7.10205 and XEQ "BC". The registers R07, R12, R17, R22, R27, R32, R37, R42, R47, R52, R57, R62, R67, R72, R77, R82, R87, R92, R97, and finally R102 should all contain zero. These registers may be inspected by keying in 7.10205 and XEQ "BV". Since BV skips over registers which are zero, none of the above registers will show up in the display; only a series of short TONES will be heard as BV runs through the registers.

COMPLETE INSTRUCTIONS FOR BC

1) The only input to BC is the block control word which is of the form bbb.eeeII.

2) BC contains an internal ISG loop that is controlled by bbb.eeeII. BC stores and uses this block control word in the Last X register. The Y, Z, and T registers are all preserved by BC. The X register contains 0 when BC ends. The following shows the stack contents on input/output from BC.

Input to BC:	Output from BC:
T: T	T: T
Z: Z	Z: Z
Y: Y	Y: Y
X: bbb.eeeII	X: 0
L: L	L: final control word

MORE EXAMPLES OF BC

Example 3: Use BC to clear registers R13-R49 inclusive.

Key 13.049 and XEQ "BC".

Example 4: Use BC to clear the even numbered registers from R02-R100.

Key 2.10002 and XEQ "BC".

Example 5: Use BC to clear the odd numbered registers from R01-R99.

Key 1.09902 and XEQ "BC".

Routine Listing For: BC
208◆LBL "BC"
209 SIGN
210 CLX
211◆LBL 13
212 STO IND L
213 ISG L
214 GTO 13
215 RTN

LINE BY LINE ANALYSIS OF BC

BC is a very short routine. The SIGN function at line 209 is used to store the block control word in LAST X which is used as an ISG counter in the loop in BC.

CONTRIBUTORS HISTORY FOR BC

The BC routine and documentation were written by John Kennedy (918) based on the suggestion from Richard Schwartz (2289) that where possible, the block routines should make full use of the block control word.

FINAL REMARKS FOR BC

The one area of improvement for a future BC routine would be greater speed.

FURTHER ASSISTANCE ON BC

NOTES

NOTES

TECHNICAL DETAILS

XROM: 20, 43 | **BC** | SIZE: depends on block used

Stack Usage:
- 0 T: not used
- 1 Z: not used
- 2 Y: not used
- 3 X: 0
- 4 L: ISG counter

Alpha Register Usage:
- 5 M: not used
- 6 N: not used
- 7 O: not used
- 8 P: not used

Other Status Registers:
- 9 Q: not used
- 10 ⊢: not used
- 11 a: not used
- 12 b: not used
- 13 c: not used
- 14 d: not used
- 15 e: not used

ΣREG: not used

Data Registers:

R00:

R06: the data registers used are those defined by the block
R07:
R08:
R09:
R10:
R11:
R12:

Flag Usage:
- 04: not used
- 05: not used
- 06: not used
- 07: not used
- 08: not used
- 09: not used
- 10: not used
- 25: not used

Display Mode: not used

Angular Mode: not used

Unused Subroutine Levels: 5

Global Labels Called:

Direct	Secondary
none	none

Local Labels In This Routine: 13

Execution Time: depends on block size, clears approximately 7.8 registers per second

Peripherals Required: none

Interruptible? yes

Execute Anytime? no

Program File: M

Bytes In RAM: 18

Registers To Copy: 61

Other Comments:

No special SIZE requirement is necessary provided the data block already exists

Program CV,BC,SV For PPC ROM Programs 366 bytes Page 1 of 3

	X	Y	Z	T	L

```
01♦LBL "CV"
02 GTO IND 06

03♦LBL A
04♦LBL 01
05 CF 10

06♦LBL 06
07 STO 09
08 X<>Y
09 STO 08
10 ΣREG 13
11 FC? 10
12 Σ+
13 FS? 10
14 Σ-
15 RDN
16 RCL 08
17 ENTER↑
18 X>0?
19 LN
20 ST* Z
21 RCL 09
22 X>0?
23 LN
24 ST* Z
25 X<>Y
26 ΣREG 19
27 FC? 10
28 Σ+
29 FS? 10
30 Σ-
31 R↑
32 FS? 10
33 CHS
34 ST+ 12
35 R↑
36 FS? 10
37 CHS
38 ST+ 11
39 X<> Z
40 SIGN
41 ST+ L
42 RCL 08
43 RCL 09
44 X<> L
45 RTN
46 RCL 08
47 RCL 09

48♦LBL a
49 SF 10
50 GTO 06
51♦LBL B
52♦LBL 02
53 CF 08
54 CF 09
55 STO 07
56 2
57 X<Y?
58 SF 09
59 /
60 FRC
61 X=0?
62 SF 08
63 8
64 ST+ 07
65 XEQ IND 07
66 RCL 17
67 RCL 13
68 RCL 15
69 STO 09
70 *
71 RCL 18
72 /
73 -
74 STO 10
75 RCL 14
76 RCL 13
77 X↑2
78 RCL 18
79 /
80 -
81 STO Z
82 /
83 STO 08
84 RCL 13
85 *
86 ST- 09
87 X<>Y
88 RCL 16
89 RCL 15
90 X↑2
91 RCL 18
92 ST/ 09
93 /
94 -
95 *
96 SQRT
97 ST/ 10
98 XEQ IND 07
99 8
100 ST- 07
101 RCL 10
102 RCL 09
103 FS? 08
104 E↑X
105 STO 09
106 RCL 08
107 RTN
```

```
108♦LBL 10
109 RCL 11
110 X<> 17
111 STO 11

112♦LBL 13
113 RCL 21
114 X<> 15
115 STO 21
116 RCL 22
117 X<> 16
118 STO 22

119♦LBL 09
120 RTN

121♦LBL 11
122 RCL 12
123 X<> 17
124 STO 12

125♦LBL 14
126 RCL 19
127 X<> 13
128 STO 19
129 RCL 20
130 X<> 14
131 STO 20
132 RTN

133♦LBL 12
134 RCL 23
135 X<> 17
136 STO 23
137 XEQ 14
138 GTO 13

139♦LBL C
140♦LBL 03
141 FS? 09
142 LN
143 RCL 08
144 *
145 RCL 09
146 FS? 08
147 LN
148 +
149 FS? 08
150 E↑X
151 RTN

152♦LBL D
153♦LBL 04
154 FS? 08
155 LN
156 RCL 09
157 FS? 08
158 LN
159 -
160 RCL 08
161 /
162 FS? 09
163 E↑X
164 RTN

165♦LBL e
166♦LBL 00
167 11.024
168 XEQ "BC"
169  E
170 RTN

171♦LBL E
172♦LBL 05
173 .
174 STO 25
175 4
176 STO 07

177♦LBL 07
178 RCL 07
179 XEQ B
180 RCL 25
181 RCL 10
182 ABS
183 X<=Y?
184 GTO 15
185 STO 25
186 RCL 07
187 STO 26

188♦LBL 15
189 DSE 07
190 GTO 07
191 RCL 26
192 XEQ 02
193 RCL 26
194 RTN
```

Program CV,BC,SV For ______ 366 bytes Page 3 of 3

```
195•LBL "BC"
196 SF 25
197 CLRGX
198 FS?C 25
199 RTN
200 SIGN
201 CLX

202•LBL 16
203 STO IND L
204 ISG L
205 GTO 16
206 RTN

207•LBL "SV"
208 STO 07
209  E
210 %
211 RCL Z
212 X=0?
213 X<>Y
214 STO 09
215 CLST

216•LBL 08
217 RCL Z
218 STO 08
219 RCL 07
220 FC? 00
221 VIEW X
222 XEQ IND 06
223 ST* 09
224 ST- 08
225 RCL 09
226 RCL 08
227 X≠0?
228 /
229 STO 09
230 X<> 07
231 ST+ 07
232 RND
233 RCL 07
234 RND
235 X≠Y?
236 GTO 08
237 RCL 07
238 END
```

C61

STRUCTURAL

PROGRAMS

Structural Engineer's Creed

Hope is stronger than truth.

Truth is stronger than steel.

I hope this thing stands up.

REINFORCED CONCRETE DESIGN/ANALYSIS

"In the field of structural design the effort to get intelligence through standardization has been carried pretty far. In reinforced concrete, for example, it has been necessary to set up elaborate standards. Out of this work came a narrowly circumscribed standardization of procedures, which is called "the theory of reinforced concrete" and to which unfortunate students are exposed. Few will question that the standardized theory of reinforced concrete is perhaps as complicated a bit of nonsense as has been conceived by the human mind. It does, however, work pretty well as a check on undiscriminating unintelligence."

Hardy Cross, Engineers And Ivory Towers

There are six programs and several subroutines in this module concerned with reinforced concrete. The programs are:

WSBM	Working stress rectangular beam design
LFBM	Load factor (ultimate strength) rectangular beam design
Ic	Working stress cracked-section beam analysis
LFAN	Load factor beam analysis with cracking and fatigue check
VST	Shear reinforcing design (WSD or LFD)
CORBL	Concrete corbel design (WSD or LFD)

The subroutines are listed on the next page. In addition, a listing for a rectangular concrete column analysis program is included for use in RAM. Shortage of space and a re-evaluation of needs prevented it from being included in this version of BRIDGEROM.

FLAGS

These programs follow AASHTO if flag 04 is clear, and ACI if it is set. There isn't a lot of difference between the two (there isn't any difference for the LFD versions of VST and CORBL); what differences there are are outlined in the individual program documentation.

All the programs follow the conventions for the "user flags" 00 and 02. Shortened output results if flag 00 is set. Flag 02 is also active in these programs. Warning BEEPs are enabled if flag 02 is set;

D1

they are silenced if it is clear. The programs all access the input subroutines N/Fc, which have warnings if the concrete and steel strengths are outside a certain range. Individual programs also have other warnings written in. Flags 01 and 03 have no effect on these programs.

CONCRETE COLUMN ANALYSIS

As noted, listings for CCOL and its subroutine CC are provided for the user to key into RAM. Space constraints made it impossible to include program CC in this version of BRIDGEROM or to provide a full concrete column analysis program.

CONSTANTS and STRESSES

The constant B is calculated (in subroutine N) as 0.85 for Fc' of 4000 psi or less, decreasing by 0.05 for each 1000 psi gain in concrete strength, with a minimum value of 0.65 for Fc' of 8000 psi or more.

An idiosyncracy resulting from the use of both working stress and ultimate strength methods is that the reinforcing strength is prompted differently for working stress (F-S 20,24) than for ultimate strength (F-Y 40,60); but the concrete prompt (Fc' KSI) always intends the ultimate concrete strength to be input. For working stress designs it is then factored down according to the status of flag 04.

SUBROUTINES

All of the following subroutines (except KJ) are accessed by the reinforced concrete programs and may find some use in user-written programs. KJ is a stand-alone program which may be useful for hand calculations and as a subroutine in other programs. For storage register and input requirements, see the individual program listings with stack trace.

Mc Calculates and displays 1.2 times the concrete cracking moment based on gross section and concrete modulus of rupture, 7.5 $\sqrt{f_c'}$. Falls into subroutine "U."

U Calculates the area of reinforcing needed for a given moment by ultimate strength methods, leaving the moment in the Y register. The moment should include the load factor but not the undercapacity factor phi. Falls into display routine AS.

AS Displays the area of steel in square inches.

N Prompts for reinforcing steel F-S (for WSD, FC 06) or F-Y (for LFD, FS 06). If by WSD, calculates

F-Y. Falls into routine Fc.

Fc Prompts for concrete strength f_c'. If flag 10 is set, it returns; if clear, calculates B_1, 1/m $(.85\ f_c'/F_y)$, and .75 Rho_{bal}; and leaves "N" in the alpha register for prompting by the calling program.

R Calculates and displays minimum reinforcing areas.

RH Calculates and displays maximum reinforcing areas.

VS "Port of entry" into shear reinforcing program VST

KJ Given (rho x N), calculates the working stress constants K and J for a singly reinforced rectangular section, using the formulae

$$K = \sqrt{(pn)^2 + 2pn} - pn \quad \text{and } J = 1 - K/3$$

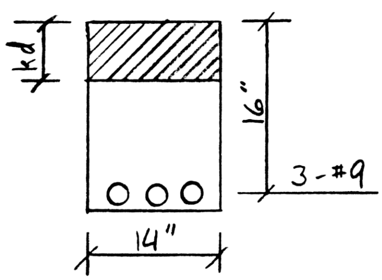
For these simple sections, it's often simpler to use KJ than the more comprehensive program Ic, solving for the missing variable:

$$M = A_s\ f_s\ J\ d/12$$

$$A_s = (M \times 12)/(f_s\ J\ d)$$

$$f_c = (2M \times 12)/(K\ J\ b\ d^2)$$

Example



$f_c' = 4.0$ ksi
$n = 8$
$f_s = 24$ ksi

$\rho = 3.0\,\text{in}^2/14 \times 16 = .0134$

$\rho n = .0134 \times 8 = .1071$

$k = \sqrt{2\rho n + (\rho n)^2} - \rho n = .3680$

N.A. position $= kd = .368 \times 16 = 5.89"$

$j = 1 - k/3 = .8773$

Allowable $M = 3.0\,\text{in}^2 \times 24\text{ ksi} \times .8773 \times 16"/12$

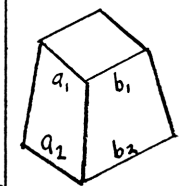
$= 84.22'^{k}$

D3

Program KJ, LPT, PYR For ______ 155 bytes Page 1 of 1

Step	X	Y	Z	T	L / Notes
01♦LBL "KJ"	$\rho \times n$				
02 ENTER↑					
03 ENTER↑					
04 ENTER↑	ρn	ρn	ρn	ρn	
05 *	$(\rho n)^2$				
06 +					
07 +	$(\rho n)^2+2\rho n$				
08 SQRT	$\sqrt{\ }$				
09 X<>Y	ρn				$K=\sqrt{(\rho n)^2+2\rho n}-\rho n$
10 -	K				
11 ENTER↑	K	K	ρn	ρn	$J=1-K/3$
12 SIGN	1				
13 LASTX	K				
14 3	3	K	1	K	
15 /					
16 -	J	K	K	K	
17 X<>Y	K	J	K	K	
18 RTN					
19♦LBL "LPT"					
20 XEQ "L*"					
21♦LBL 01					
22 "J↑D↑K↑L"					
23 PROMPT	L	K	D	J	
24 R↑					
25 STO 01	J				
26 R↑					
27 STO [	D				
28 ST+ 00					
29 R↑					
30 R↑	L	K	D	J	
31 XEQ "XY"	ΔX	ΔY			
32 R-P	d	θ			
33 X<> [	D	θ			
34 P-R	$\Delta X'$	$\Delta Y'$			
35 RCL 01	J				
36 XEQ "P"	Y_J	X_J	ΔX	ΔY	
37 "N"					
38 XEQ "SP"					
39 GTO 01					

Step	X	Y	Z	T	L / Notes
40♦LBL "PYR"	a_1	b_1	a_2	b_2	
41 *	A_1	a_2	b_2		
42 RDN	a_2	b_2		A_1	
43 *	A_2		A_1	A_1	
44 ST* Z			$A_1 \ast A_2$		
45 R↑	A_1	A_2		$A_1 \ast A_2$	
46 +	A_1+A_2				
47 R↑	$(A_1 \times A_2)$				
48 SQRT	$\sqrt{A_1 \times A_2}$				Vol. ≐ $\frac{ht.}{3}\times(A_1+A_2+\sqrt{A_1A_2})$
49 +	Σ				
50 3					
51 /	$\Sigma/3$				
52 RTN					
53♦LBL "P1"					
54 "⊢("					
55 RTN					
56♦LBL "P2"					
57 "⊢)"					
58 RTN					
59♦LBL "PP"					
60 "⊢⊢"					
61 RTN					
62♦LBL "MU"					
63 "⊢μ"					
64 RTN					
65♦LBL "AT"					
66 "⊢@"					
67 RTN					
68♦LBL "AM"					
69 "⊢&"					
70 END					



D4

REINFORCED CONCRETE BEAM DESIGN

xeq WSBM (SIZE 015) for working stress beam design
xeq LFBM (SIZE 011) for load factor beam design

SOFTKEYS (USER mode ON)

D New beam dimensions.

USER FLAGS

Flag 00	clear	Full output
	set	Only area of steel and warnings viewed
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 04	clear	AASHTO
	set	ACI

INPUT SUMMARY

Prompt	Input	Default
F-S 20,24 or F-Y 40,60	Reinforcing or yield stress for WSBM or LFBM, respectively, in ksi	24 or 60
Fc'KSI	Ultimate concrete strength, ksi	4.0 ksi
N (WSBM only)	Modular ratio E_{st}/E_c	No default
B"	Beam width, inches	Previous B
D"	Depth to reinforcing, inches.	Previous D
H"	Full beam depth (height), inches	Previous H
M,'K	Design moment in foot-kips. By-pass (or enter zero) to access shear design subroutine.	Zero

D5

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(AASHTO or ACI)	States which code is being followed. If not using a printer, press R/S to continue.
(1.2Mc)	1.2 times the concrete cracking moment, for AASHTO's minimum reinforcing calculation
(200/FY)	ACI's minimum reinforcing ratio
(A-S)	The minimum area of steel based on one of the above two criteria
(.75Pb)	The maximum reinforcing ratio, 75% of rho-balanced
(M/BD2)	The moment over bd^2, for checking hand calculations. Is actually $M/\emptyset bd^2$ for LFBM.
(Y or a)	The depth to the neutral axis (WSBM) or the depth of the compression block (LFBM), inches
(OVER)	(WSBM only) The design is over-reinforced by working stress rules (general practice discourages but does not prohibit this)
(RHO)	The required reinforcing ratio
P/Pb	The required reinforcing ratio divided by the balanced reinforcing ratio. Design may still be acceptable if compr. steel is present.
A-S	The required area of steel
4/3 A-S	4/3 times the area of steel required for strength, if A-S is less than the minimum

Note: shear input and output are detailed in documentation for program VST.

PROGRAM FLAGS

Flag 06	clear	WSBM
	set	LFBM
Flag 07	clear	Design is physically possible
	set	Design requires a neutral axis depth greater than the depth of the member
Flag 09	clear	WSBM or LFBM
	set	CORBL
Flag 10	clear	Subroutine N runs full length

STORAGE REGISTER USE

Register	WSBM	LFBM
00	.85 Fc'/Fy	.85 Fc'/Fy
01	Fy	Fy
02	.75Pb	.75Pb
03	B (width)	B
04	D (depth)	D
05	ϕbd^2	ϕbd^2
06	$\sqrt{Fc'}/253$	$\sqrt{Fc'}/253$ (for 1.2Mcr calc.)
07	H (member height)	H
08	Min. A_{st}	Min. A_{st}
09	N (E_{st}/E_c)	Not used
10	D/3 (trial Y); temporary .75Pb	Temporary .75 Pb while in shear design subroutine
11	Trial Y	
12	(.40 or .45) Fc'	
13	F_{st} (20 or 24 ksi)	
14	R_{ideal}	

D7

P8

REINFORCED CONCRETE BEAM DESIGN

Programs WSBM and LFBM calculate the amount of tension reinforcing necessary to resist a given moment (service or factored, respectively). They specify the amount of reinforcing steel needed for singly reinforced rectangular beams. The programs assume the design reinforcing to be in a single layer, and they do not recognize compression reinforcing. T-beams can be accommodated only if the neutral axis falls within the flange. For the purpose of checking this, the programs display the neutral axis depth, Y, in inches (WSBM), or the depth of the compression block, "a" (LFBM), if flag 00 is clear.

AASHTO/ACI MINIMUM A_s

Both programs design by AASHTO if flag 04 is clear, and by ACI if flag 04 is set. The main differences are in the allowable concrete compression stress for WSBM, and the minimum amount of reinforcing required. The programs calculate a maximum reinforcing ratio of $.75p_b$; $1.2M_{cr}$, AASHTO's cracking moment criterion for minimum reinforcing; $200/F_y$, ACI's minimum reinforcing ratio; and the minimum reinforcing based on one of these, depending on the status of flag 04. These are all displayed if flag 00 is clear. The reinforcing designed is checked against these minimum and maximum values and warnings are displayed if they are violated. Neither ACI nor AASHTO requires the maximum of $.75p_b$ for working stress designs. However, it is checked for WSBM as well as LFBM as a matter of good design practice. The AASHTO minimum reinforcing is calculated (with phi = 0.9) for $1.2M_{cr}$ based on ultimate strength concepts for both WSBM and LFBM; although the present AASHTO is rather vague on this, the Commentary to the 1982 Interim Specifications, where it was introduced, indicates this more clearly. Logically, all designs should have the same minimum and maximum steel requirements.

The programs are written to accept beam dimensions and material properties, calculate the above-mentioned (ad infinitum) requirements, then enter a flexural design loop. A design moment is prompted; the programs calculate M/bd^2, the neutral-axis or compression block depth, the required reinforcing ratio, and the corresponding amount of steel. If flag 00 is clear, all these are displayed (useful for checking hand calculations); if flag 00 is set, only the area of steel is displayed. The maximum and minimum requirements are checked next. If the design is above the maximum, the programs display "P/Pb=..."; 0.75 is the maximum allowed. If the design is less than minimum, the program calculates and displays 4/3 the design area

D9

of steel. If either program finds that a reinforcing ratio over 0.1 is required, the program displays NG (no good) and returns to the M,'K prompt.

OVER-REINFORCED BEAMS

If flag 00 is clear, WSBM displays a notice, "OVER," if its design yields an "over-reinforced" beam by working stress concepts, i.e. a design governed by the allowable concrete stress. This is acceptable by general practice but may produce deflection problems. It is possible to have an over-reinforced beam which has less than $.75p_b$, especially when designing by AASHTO. With its lower allowable concrete stress (.40Fc'), AASHTO will sometimes force an over-reinforced design that would have been under-reinforced by ACI.

STRESSES

WSBM has a couple of idiosyncracies concerning the allowable stresses input. First, although the input steel stress is a working stress (usually 20 or 24 ksi), the concrete stress to be input at the Fc'KSI prompt is the <u>ultimate</u> strength of the concrete, usually 3 or 4 ksi. The factor of .40 (AASHTO) or .45 (ACI) is applied by the program. Second, the input subroutine N calculates the reinforcing yield stress from the working stress input: $F_y = 5 \times (F_{st} - 12 \text{ ksi})$, giving 40 and 60 ksi from 20 and 24, respectively. If the designer is using a different allowable stress (e.g. some tank wall designs are done at 14 or 16 ksi to hold down cracking), the calculated maximum and minimum reinforcing ratios (which are based on ultimate strength concepts and yield stresses) will be invalid.

LOAD and UNDER-CAPACITY FACTORS

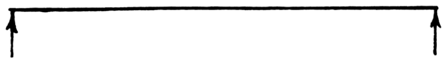
The designer must be careful to input the correct moment to LFBM. The moment must have the load factor(s) already applied; however, it must <u>not</u> include the undercapacity factor phi, which is applied by the program (O = 0.90). See the examples.

SHEAR DESIGN

Both LFBM and WSBM access a shear reinforcing design subroutine by bypassing the M,'K prompt (or keying in zero). This subroutine uses the beam dimensions and material properties input into the flexural design programs; the only new input required is the area of the stirrups and the shear. See the documentation for program VST. While in this subroutine, softkeys from the calling program are useless. VST has softkeys A and D; if either of these is accidentally used while in the shear subroutine, it will not be able to return to the calling program.

Reinforced Concrete Beam Design

Design a beam for the following conditions:



Span = 20'-0, simple
DL = 2400 #/' (heavy industrial)
LL = 3600 #/'

$f_c' = 4000$ psi
$f_y = 60$ ksi
$n = 8$

Moment

$DL = 2.40 \times 20^2/8 = 120.0'^k$
$LL = 3.60 \times 20^2/8 = 180.0'^k$

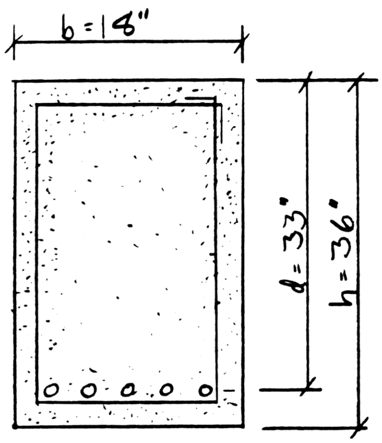
Shear

@ Reaction: $DL = 2.40 \times 20'/2 = 24.0$ k
$LL = 3.60 \times 20/2 = 36.0$ k

(Note: take @ reaction since beam depth for "d/2" is unknown)

@ 1/4 Pt.: $DL = 2.40 \times 5' = 12.0$ k
$LL = \frac{7.5'}{20'} \times (15' \times 3.60) = 20.25$ k

Try this configuration:



Design by:

1) AASHTO W.S.D.

2) ACI W.S.D. ("Alternate")

3) AASHTO L.F.D.

4) ACI L.F.D. ("Strength")

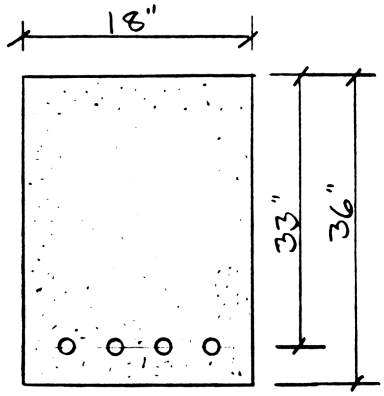
1) and 2) use "WSBM"

3) and 4) use "LFBM"

Calculate the amount of reinforcing necessary for the following loads:

DL $120^{\prime k}$ (including selfwt.)
LL $180^{\prime k}$
$300^{\prime k}$

$f_c' = 4000$ psi ; $n = 8$
$f_s = 24$ ksi (Grade 60)



Calculate Min. Steel

$$M_{cr} = S \times f_r$$

$$f_r = 7.5\sqrt{f_c'} = 7.5\sqrt{4000}$$

$$= 474.34 \text{ psi}$$

$$S = \frac{bh^2}{6} = 18'' \times 36^2/6 = 3888.0 \text{ in}^3$$

$$1.2 M_{cr} = 1.2(3888 \times .47434)/12$$

$$= \underline{144.42^{\prime k}}$$

Calc. reinf. by ult. strength methods:

$$\rho = \frac{1}{m}\left(1 - \sqrt{1 - \frac{2mR_u}{f_y}}\right)$$

$$m = \frac{f_y}{.85 f_c'} = \frac{60}{.85 \times 4} = 17.647$$

$$R_u = \frac{M}{\phi b d^2} = \frac{144.42 \times 12}{.9 \times 18'' \times (33'')^2}$$

$$= .1254$$

$$\rho = \frac{1}{17.647} \times \left(1 - \sqrt{1 - \frac{17.647 \times .1054 \times 2}{60 \text{ ksi}}}\right)$$

$$= .0021$$

Req'd $A_{s\text{-}min} = \rho b d = .0021 \times 18'' \times 33''$

$$= \underline{1.266 \,\square''}$$

Calculate Max. Steel

$$\rho_{bal} = \frac{.85\beta_1 f_c'}{f_y}\left(\frac{87}{87 + f_y}\right)$$

for $f_c' = 4000$ psi, $\beta_1 = .85$

$$.75\rho_b = .75 \times \frac{.85 \times .85 \times 4}{60} \times \left(\frac{87}{87 + 60}\right)$$

$$= \underline{.0214}$$

Max. $A_s = \rho b d = \underline{12.70 \,\square''}$

Note: neither AASHTO nor ACI require that $A_s < .75\rho_b$ for working stress design. However, it is included as a matter of good design practice. The reasons for limiting the reinf. to $.75\rho_b$ under L.F.D. are no less valid under W.S.D.; the beam doesn't know how it was designed.

D12

Calculate Ideal R

$$R_{ideal} = \frac{1}{2}(.40 f_c') \cdot j \cdot k$$

$$k = \frac{.40 f_c'}{(f_{sr}/n + .40 f_c')}$$

$$= \frac{1.6}{(24\text{ ksi}/8 + 1.6)} = \underline{.3478}$$

$$j = 1 - k/3 = .8841$$

$$R_{ideal} = \frac{1}{2}(1.60\text{ ksi}) \times .3478 \times .8841$$

$$= \underline{.2460}\text{ ksi}$$

Calculate Req'd Reinforcing

$$\text{Actual } R = \frac{M}{bd^2} = \frac{300'^{k} \times 12}{18 \times 33^2}$$

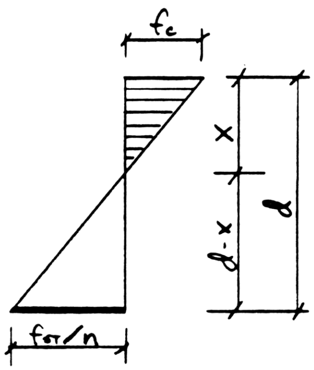
$$= .1837\text{ ksi}$$

$R_{act} < R_{ideal} \Rightarrow$ under-reinforced

∴ Reinf. @ 24 ksi ; Concrete stress unknown

$$M = C \times (d - x/3)$$

Max. concrete stress $f_c = \dfrac{(f_s/n) \times x}{d - x}$



$$C = \frac{1}{2} \times f_c \times x \times b$$

$$\therefore 300'^{k} \times 12 = \frac{1}{2}\left(\frac{(24/8)\,x}{33'' - x}\right) \cdot 18x \cdot \left(33'' - \frac{x}{3}\right)$$

$$3600'^{k} \times (33 - x) = \frac{1}{2}(3x)(18x)\left(33 - \frac{x}{3}\right)$$

$$9x^3 - 891x^2 - 3600x + 118{,}800 = 0$$

solving the cubic by iteration,

$$x = 10.1435'' \text{ (Neutral Axis)}$$

$$C = \frac{1}{2}\left(\frac{3 \times 10.1435''}{33 - 10.1435''}\right) \cdot 10.1435'' \times 18''$$

$$= 121.54\text{ k}$$

$$\text{Req'd } A_s = 121.54\text{ k}/24\text{ ksi}$$

$$= \underline{5.0643 \square''}$$

Check: $5.06 < 12.70 \square''$ $(.75\rho_b)$
$5.06 > 1.27$ (Min)

<u>OK</u>

USE 4 - #10

Check spacing

$$18'' - 2(2''_{\text{clear}} + \tfrac{3}{4}''_{\text{stirrup}}) - 1\tfrac{1}{2}''_{\text{bar}} \overset{\text{say}}{=} 11''$$

$$11''/_{3\text{ spa.}} - 1\tfrac{1}{2}''_{\text{bar}} = 2.17''\text{ clear}$$

<u>OK</u>

Note: WSBM does not check spacing or bar placement.

DB

Calculate Reinforcing needed for 450''k on same section.

$$R = \frac{450 \times 12}{18 \times 33^2} = .2755 \text{ ksi}$$

$R_{act} > R_{ideal} \Rightarrow$ Over-reinf.

$\therefore$ Conc. stress $= .40 f'_c$; $f_{st} < 24$ ksi

$$M = C \times (d - X/3)$$

$$450 \times 12 = \tfrac{1}{2}(1.60 \text{ ksi}) \times 18X \times (33 - \tfrac{X}{3})$$

$$5400''k = 475.20x - 4.80x^2$$

solving the quadratic directly,

$X = 13.0960''$

$$C = \tfrac{1}{2} \times 1.60 \text{ksi} \times 18'' \times 13.0960''$$

$$= 188.58 \text{ k}$$

$$f_s = n f_c \left(\frac{d-x}{x}\right)$$

$$= 8 \times (.4 \times 4 \text{ ksi}) \times \left(\frac{33'' - 13.0960''}{13.0960''}\right)$$

$$= 19.454 \text{ ksi}$$

$$\text{Req'd } A_s = \frac{188.58 \text{ k}}{19.454} = 9.6937 \square''$$

Check: $9.69\square'' < 12.70\square''$ ($.75\rho_b$)
$9.69 > 1.27$ (Min)

OK - USE 7-#11 ($A_s = 10.72 < 12.70$)

Check Spacing:

$$18'' - 2(2'' + \tfrac{3}{4}'') - 1\tfrac{1}{2}'' = 11''$$
(cl. st. bar)

$$\frac{11''}{6 \text{ spa}} - 1\tfrac{1}{2}'' \text{ (bar)} = .33'' \text{ clear}$$

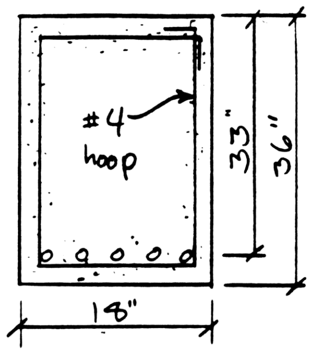
No Good

Will need 2 layers of reinforcing. Results of calculation (9.69 □") no longer valid. Can use program "Ic" to check capacity of two-layer estimate.

Note: WSBM does not check spacing or bar placement. As in this case, the area of steel calculated by WSBM may be physically impossible to place.

From the preceding:

$f_c' = 4000$ psi
$f_y = 60$ ksi
$f_{st} = 24$ ksi



@ Reaction:

DL = 24.0 k (service)
LL = 36.0 k
Σ = 60.0 k

@ 1/4 Point:

DL = 12.0 k
LL = 20.25 k
Σ = 32.25 k

$V_c = .95\sqrt{4000} \times 18'' \times 33'' = \underline{35.69\ k}$

@ Reaction,

$V_s = 60.0 - 35.69 = 24.31$ k

Use stirrups - #4 hoop

$A_v = 2 \times .20 = .40\ \square''$

$Spa. = \frac{A_v f_s d}{V_s} = \frac{.40\square'' \times 24\text{ksi} \times 33''}{24.31\ k} = 13.03''$

@ 1/4 Point

$V_s = 32.25 - 35.69$

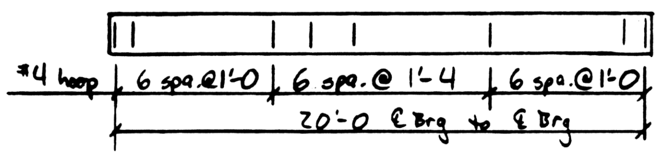
$V_s < 0 \Rightarrow$ Min. Steel

$\frac{1}{2} V_c = 17.84$ k

$32.25 > 17.84 \Rightarrow$ Need Stirrups

Max. Spa.:
$d/2 = 33/2 = 16.5''$ ← Governs
max $= 24''$

Use this layout:



Check Maximums

$2\sqrt{f_c}\, b\, d = 2\sqrt{4000} \times 18 \times 33$

$= 75.14$ k

$V_s < 2\sqrt{f_c'} \Rightarrow$ Spa. OK - AASHTO 8.15.5.3.8

$V_s < 4\sqrt{f_c'} \Rightarrow$ OK - AASHTO 8.15.5.3.9

$A_{v\text{-}min} = \frac{50\, b_w \cdot Spa}{f_y}$ AASHTO 8.19.1.2

$\therefore$ Max Spa. $= \frac{.40 \times 60{,}000}{50 \times 18''}$

$= 26.67''$ OK

```
        XEQ "WSBM"
 6:33:12 PM  12/09/88

TITLE:
TEST 1               RUN
************************
AASHTO
F-S 20,24
         24.0000     RUN
Fc'KSI
          4.0000     RUN
N
          8.0000     RUN
B"
         18.0000     RUN
D"
         33.0000     RUN
H"
         36.0000     RUN
1.2Mc=184.4111 'K
 A-S=1.2656 SQ"
200/FY=0.0033
.75Pb=0.0214
************************
M,'K
        300.0000     RUN
M/BD2=183.6547
Y=10.1435"
RHO=0.0085
P/Pb=0.2991-OK
A-S=5.0643 SQ"
-----------------------
M,'K
        450.0000     RUN
M/BD2=275.4821
OVER
Y=13.0960"
RHO=0.0163
P/Pb=0.5725-OK
A-S=9.6937 SQ"
-----------------------
M,'K
        100.0000     RUN
M/BD2=61.2182
Y=6.2049"
RHO=0.0027
P/Pb=0.0955-OK
A-S=1.6165 SQ"
-----------------------
```

Design by AASHTO

1.2 M_{crack} Min. Reinf.

Over-reinforced

```
M,'K
         60.0000     RUN
M/BD2=36.7309
Y=4.8884"
RHO=0.0016
P/Pb=0.0565-OK
A-S=0.9563 SQ"
4/3 A-S=1.2751 SQ"
-----------------------
M,'K
                     RUN
************************
SHEAR
Vc=35.6895 K
************************
A-V
           .4000     RUN
---------
V, K
         60.0000     RUN
SPA=13.0314"
---------
V, K
         32.2500     RUN
SPA=16.5000"(MAX)
---------
V, K
        120.0000     RUN
SPA=3.7575"
---------
V, K
         15.0000     RUN
.5Vc>V
---------
V, K
                     RUN
************************
A-V
                     RUN
B"
```

A_s < Min. A_s so display 4/3 A_s

Bypass to access shear design

Bypass

Bypass to return to WSBM - new beam dimensions

```
                    CF 04     Clear flag 04 → AASHTO
                    SF 00     Set flag 00 → Short Format

               XEQ "WSBM"     No title block
F-S 20,24
          24.0000    RUN
Fc'KSI
           4.0000    RUN
N
           8.0000    RUN
B"
          18.0000    RUN
D"
          33.0000    RUN
H"
          36.0000    RUN      No minimums or maximums
************************
M,'K
         300.0000    RUN
A-S=5.0643 SQ"                Only A_s output
-----------------------
M,'K
         450.0000    RUN
A-S=9.6937 SQ"
-----------------------
M,'K
         600.0000    RUN
P/Pb=1.6172-NG                Warnings still output
A-S=27.3837 SQ"
-----------------------
M,'K
          60.0000    RUN
A-S=0.9563 SQ"
4/3 A-S=1.2751 SQ"            Minimums still output
-----------------------
M,'K
```

Calculate the reinforcing required by ACI for moments of 300'-k and 450'-k on the same section.

$f'_c = 4000$ psi ; $n = 8$
$f_{ST} = 24$ ksi (Grade 60)

$b = 18''$
$d = 33''$

Calculate Min. Steel

$$\rho_{min} = \frac{200}{f_y} = \frac{200}{60{,}000} = .0033$$

Min. $A_s = .0033 \times 18'' \times 33'' = \underline{1.98 \square''}$

Calculate Max. Steel

As for AASHTO, $.75\,\rho_b = .0214$

Max. $A_s = .0214 \times 18'' \times 33'' = \underline{12.70 \square''}$

Calculate Ideal R

$$R_{ideal} = \tfrac{1}{2}(.45 f'_c) \times j \times k$$

$$k = \frac{.45 \times 4.0}{(24\text{ksi}/n{=}8 + .45 \times 4.0)} = .3750$$

$$j = 1 - k/3 = .8750$$

$$R_{ideal} = \tfrac{1}{2}(1.8 \text{ ksi}) \cdot .375 \cdot .875$$

$$= \underline{.2953 \text{ ksi}}$$

Calculate Req'd Reinf. for M = 300'-k

$$\text{Actual } R = \frac{300 \times 12}{18 \times 33^2} = .1837 \text{ ksi}$$

$.1837 < .2953 \Rightarrow$ under-reinf.

∴ Calculations are as before:

$X = 10.1435''$
$C = 121.54$ k

Req'd $A_s = \underline{5.0643 \square''}$

check : $5.06 < 12.70$ $(.75\rho_b)$
$5.06 > 1.98 \square''$ (min)

OK - USE 4-#10

Calculate Req'd Reinf. for M = 450'-k

$$R = \frac{450 \times 12}{18'' \times 33^2} = .2755 \text{ ksi}$$

$.2755 < .2953 \Rightarrow$ under-reinforced

(was over-reinforced by AASHTO)

$$C = \tfrac{1}{2} f_c X b \;\; ; \;\; f_c = \left(\frac{(f_s/n) \cdot X}{d - X}\right)$$

$$f_c = \frac{24 \text{ ksi}/8 \cdot X}{33'' - X}$$

$$M = C \cdot (d - X/3)$$

$$450 \times 12 = \tfrac{1}{2}\left(\frac{24/8 \; X}{33 - X}\right) \cdot 18X \cdot \left(33 - \frac{X}{3}\right)$$

$$9X^3 - 891X^2 - 5400X + 178{,}200 =$$

solving by iteration, $\underline{X = 12.0283''}$

$$C = \frac{1}{2}\left(\frac{3 \times 12.0283''}{33 - 12.0283''}\right) \times 12.0243'' \times 18''$$

$$= 186.27 \text{ k}$$

$$\text{Req'd } A_s = \frac{186.27}{24 \text{ ksi}} = \underline{7.7611 \ \square''}$$

Check: $7.76 < 12.70 \quad (.75\rho_b)$
$7.76 > 1.98 \quad (\text{Min})$

OK - USE 5-#11

Check spacing

$$18'' - 2(2'' + 3/4'') - 1\tfrac{1}{2}'' = 11''$$
(clear, stirrup, bar)

$$\frac{11''}{4 \text{ spa.}} - 1\tfrac{1}{2}'' = 1.25'' \text{ clear}$$

OK

SHEAR DESIGN

$$V_c = 1.1\sqrt{4000} \times 18'' \times 33'' = 41.32 \text{ k}$$

@ Support

$$V_s = 60.0 - 41.32 = 18.68 \text{ k}$$

for $A_v = 2\text{-}\#4 = .40 \ \square''$,

$$\text{Spa.} = \frac{.40 \times 24 \text{ ksi} \times 33''}{18.68 \text{ k}} = 16.96''$$

USE #4 @ 16" ENTIRE BEAM

```
                SF 04
         XEQ "WSBM"
 6:40:21 PM  12/09/88

TITLE:
TEST 2               RUN
************************
ACI
F-S 20,24
          24.0000    RUN
Fc'KSI
           4.0000    RUN
N
           8.0000    RUN
B"
          18.0000    RUN
D"
          33.0000    RUN
H"
          36.0000    RUN
1.2Mc=184.4111 'K
200/FY=0.0033
 A-S=1.9800 SQ"
.75Pb=0.0214
************************
M,'K
         300.0000    RUN
M/BD2=183.6547
Y=10.1435"
RHO=0.0085
P/Pb=0.2991-OK
A-S=5.0643 SQ"
-----------------------
M,'K
         450.0000    RUN
M/BD2=275.4821
Y=12.0283"
RHO=0.0131
P/Pb=0.4583-OK
A-S=7.7611 SQ"
-----------------------
```

Set flag 04 for ACI

Design by ACI

Min. A_s figured for $200/F_y$

Not over-reinforced by ACI due to higher allowable f_c

```
M,'K
         100.0000    RUN
M/BD2=61.2182
Y=6.2049"
RHO=0.0027
P/Pb=0.0955-OK
A-S=1.6165 SQ"
4/3 A-S=2.1553 SQ"
-----------------------
M,'K
                     RUN
************************
SHEAR
Vc=41.3246 K
************************
A-V
          60.0000    RUN
---------
V, K
                     RUN
************************
A-V
            .4000    RUN
---------
V, K
          60.0000    RUN
SPA=16.5000"(MAX)
---------
V, K
```

A_s < Min. A_s for ACI so display 4/3 A_s

Bypass to access shear design

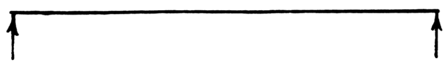
Oops - error

Bypass to return to A_v prompt

D20

Reinforced Concrete Beam Design

Design a beam for the following conditions :



Span = 20'-0, simple
DL = 2400 #/'
LL = 3600 #/' (heavy industrial)

f_c' = 4000 psi
f_y = 60 ksi
n = 8

Design by:

1) AASHTO W.S.D.

2) ACI W.S.D. ("Alternate")

3) AASHTO L.F.D.

4) ACI L.F.D. ("Strength")

1) and 2) use "WSBM"

3) and 4) use "LFBM"

Moment

$DL = 2.40 \times 20^2/8 = 120.0'^{k}$
$LL = 3.60 \times 20^2/8 = 180.0'^{k}$

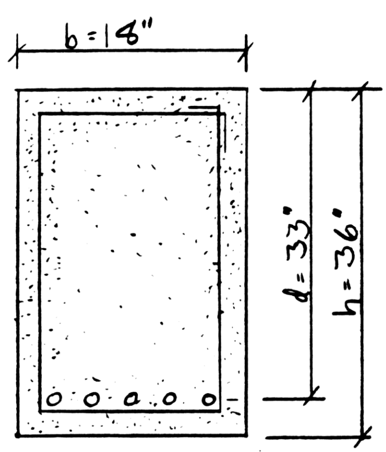
Shear

@ Reaction : $DL = 2.40 \times 20'/2 = 24.0$ k
$LL = 3.60 \times 20/2 = 36.0$ k

(Note : take @ reaction since beam depth for "d/2" is unknown)

@ 1/4 Pt. : $DL = 2.40 \times 5' = 12.0$ k
$LL = \frac{7.5'}{20'} \times (15' \times 3.60) = 20.25$ k

Try this configuration :



Calculate the amount of reinforcing necessary for the same loads by AASHTO:

	Serv.	Factored
DL	$120'^{k}$	$120'^{k}$
LL	$180'^{k} \times 5/3$	$= 300'^{k}$
		$420'^{k} \times 1.3 = \underline{546'^{k}}$

Note: undercapacity factor $\phi = .9$ not applied. ϕ is applied by LFBM.

$f_c' = 4000$ psi $b = 18"$
$f_y = 60$ ksi $d = 33"$

Calculate Min. and Max. Steel

Note: L.F.D. Min. and Max. steel are identical to W.S.D. limits. See preceding W.S.D. calculations.

$1.2 M_{cr} = 184.42'^{k}$

Min. $A_s = \underline{1.266 \square"}$

$.75 \rho_b = .0214$

Max. $A_s = \underline{12.70 \square"}$

Calculate Req'd Reinforcing

$$\rho = \frac{1}{m}\left(1 - \sqrt{1 - \frac{2 m R_u}{f_y}}\right)$$

$$m = \frac{f_y}{.85 f_c'} = \frac{60}{.85 \times 4} = 17.647$$

$$R_u = \frac{M}{\phi b d^2} = \frac{546'^{k} \times 12}{.9 \times 18" \times (33")^2}$$

$= .3714$ ksi

$$\rho = \frac{1}{17.647}\left(1 - \sqrt{1 - \frac{2 \times 17.647 \times .371}{60}}\right)$$

$= .0066$

Req'd $A_s = .0066 \times 18" \times 33"$

$= \underline{3.9031 \square"}$

check: $3.90 < 12.70\square"$ ($.75\rho_b$)
$3.90 > 1.266\square"$ (Min.)

OK - USE 4-#9

Check spacing

$18" - 2(2" + 3/4") - 1\frac{1}{4}" = 11.25"$
(clear, stirrup, say bar)

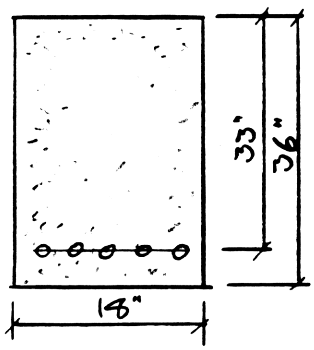
$\frac{11.25"}{3\text{ spa.}} - 1\frac{1}{4}"\text{ bar} = 2.50"$ clear

OK

Note: LFBM does not check bar placement or clearance; fatigue; Z-cracking; or deflection. Program LFAN is somewhat more complete.

(DZZ)

Design shear reinforcing for this beam:



$f_c' = 4000$ psi
$f_y = 60$ ksi

$V_c = 2\sqrt{4000} \times 18" \times 33" = \underline{75.14 \text{ k}}$

Use #4 stirrups - $A_v = 2 \times .20 = \underline{.40 \square"}$

Max. Spacing

$A_{v\text{-}min} = \dfrac{50\, b \cdot Spa}{f_y}$ (AASHTO 8.19.1.2)

$\therefore$ Max spa. $= \dfrac{.40 \times 60{,}000}{50 \times 18"}$ = 26.67"

or $\frac{d}{2} = \frac{33'}{2}$ = 16.5" ← Governs
or Max = 24"

@ Reaction

DL = 24.0
LL = 5/3 × 36.0 = 60.0
84.0 × 1.3 = 109.2 k

$V_s = 109.2/(\phi = .85) - 75.14 = 53.33$ k

$Spa. = \dfrac{A_v f_y d}{V_s} = \dfrac{.40 \times 60 \times 33"}{} = 14.85"$

@ 1/4 Point

DL = 12.0
LL = 5/3 × 20.25 = 33.75
45.75 × 1.3 = 59.48

$V_s = \dfrac{59.48}{(\phi = .85)} - 75.14 \text{ k} < 0$

⇒ Use Max. Spa. = 16.5"

```
XEQ "LFBM"
6:43:33 PM  12/09/88

TITLE:
TEST 3                  RUN
***************************
AASHTO
F-Y 40,60
          60.0000       RUN
Fc'KSI
           4.0000       RUN
B"
          18.0000       RUN
D"
          33.0000       RUN
H"
          36.0000       RUN
1.2Mc=184.4111 'K
 A-S=1.2656 SQ"
200/FY=0.0033
.75Pb=0.0214
***************************
M,'K
         546.0000       RUN
M/BD2=371.3907
a=3.8265"
RHO=0.0066
P/Pb=0.2305-OK
A-S=3.9031 SQ"
-------------------------
M,'K
                        RUN
***************************
SHEAR
Vc=75.1357 K
***************************
A-V
            .4000       RUN
---------
V, K
         109.2000       RUN
SPA=14.8496"
---------
V, K
          59.4800       RUN
SPA=16.5000"(MAX)
---------
```

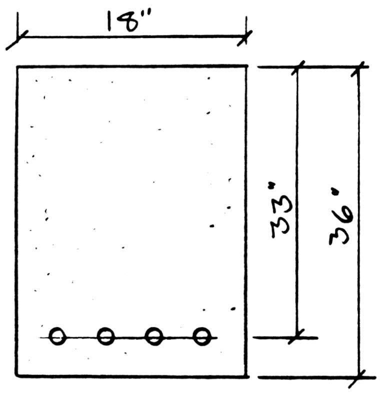
D23

Calculate the amount of reinforcing necessary for the following loads:

	Service		Factored	
DL	120'-k	× 1.4 =	168.0	(includes beam)
LL	140'-k	× 1.7 =	306.0	
			474.0'-k	

Note: undercapacity factor $\phi = 0.9$ not applied. ϕ is applied by LFBM.

$f_c' = 4000$ psi b = 18"
$f_y = 60$ ksi d = 33"



Calculate Min. and Max. Steel

Note: L.F.D. Min. and Max. limits are identical to W.S.D. limits. See preceding W.S.D. calculations.

$$\frac{200}{f_y} = .0033$$

Min. A_s = 1.98 □"

$$.75\,\rho_{bal} = .0214$$

Max. A_s = 12.70 □"

Calculate Req'd Reinforcing

$$\rho = \frac{1}{m}\left(1 - \sqrt{1 - \frac{2mR_u}{f_y}}\right)$$

$$m = \frac{f_y}{.85 f_c'} = \frac{60}{.85 \times 4} = 17.647$$

$$R_u = \frac{M}{\phi b d^2} = \frac{474'^{k} \times 12}{.9 \times 18" \times 33^2}$$

$$= .3224 \text{ ksi}$$

$$\rho = \frac{1}{17.647}\left(1 - \sqrt{1 - \frac{2 \times 17.647 \times .3224}{60 \text{ ksi}}}\right)$$

$$= .0057$$

Req'd $A_s = \rho b d = .0057 \times 18" \times 33"$

= 3.3596 □"

check: 3.36 □" < 12.70 □" ($.75\rho_b$)
3.36 > 1.98 □" (Min)

OK - USE 4-#9 - A_s = 4.00 □"

Check Spacing

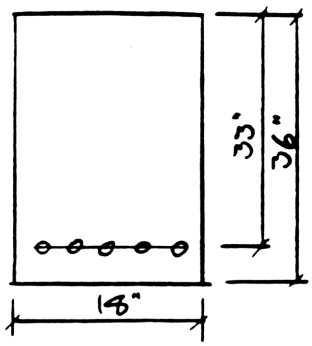
18" − 2 (2" c/c + 3/4" stirrup) − 1¼" (say bar) = 11.25"

$$\frac{11.25"}{3 \text{ spa}} - 1\tfrac{1}{4}" = 2.50" \text{ clear}$$

OK

Note: LFBM does not check bar placement or clearance as here. Neither does it check z-cracking deflection. Program LFAN checks the latter two.

Design shear reinforcing for this beam:



$f_c' = 4000$ psi
$f_y = 60$ ksi

$V_c = 2\sqrt{4000} \times 18" \times 33" = \underline{75.14\ k}$

Use #4 stirrups - $A_v = 2 \times .20 = \underline{.40 \square"}$

Max. Spacing

$A_{v\text{-}min} = \frac{50\ b \cdot Spa}{f_y}$ (ACI 11.5.5.3)

$\therefore$ Max spa. $= \frac{.40 \times 60,000}{50 \times 18"}$

$= 26.67"$

or $\frac{d}{2} = \frac{33"}{2} = 16.5"$ ← Governs

or Max $= 24"$

@Reaction

DL = 1.4 × 24.0 = 33.6
LL = 1.7 × 36.0 = 61.2
94.8 k

$V_s = 94.8/(\phi = .85) - 75.14 = 36.39$ k

$Spa. = \frac{A_v f_y d}{V_s} = \frac{.40 \times 60 \times 33"}{36.39} = 21.76"$

Use 16" Spacing Entire Beam

```
SF 04

XEQ "LFBM"
6:47:04 PM  12/09/88

TITLE:
TEST 4                RUN
*************************
ACI
F-Y 40,60
         60.0000      RUN
Fc'KSI
          4.0000      RUN
B"
         18.0000      RUN
D"
         33.0000      RUN
H"
         36.0000      RUN
1.2Mc=184.4111 'K
200/FY=0.0033
 A-S=1.9800 SQ"
.75Pb=0.0214
*************************
M,'K
        474.0000      RUN
M/BD2=322.4161
a=3.2937"
RHO=0.0057
P/Pb=0.1984-OK
A-S=3.3596 SQ"
-------------------------
M,'K
        150.0000      RUN
M/BD2=102.0304
a=1.0056"
RHO=0.0017
P/Pb=0.0606-OK
A-S=1.0257 SQ"
4/3 A-S=1.3676 SQ"
-------------------------
M,'K
                      RUN
*************************
SHEAR
Vc=75.1357 K
*************************
A-V
           .4000      RUN
---------
V, K
         94.8000      RUN
SPA=16.5000"(MAX)
---------
V, K
         30.0000      RUN
.5Vc>V
---------
V, K
```

Program WSBM/LFBM/CORBL For Concrete Design 752 bytes Page 1 of 4

Step	X	Y	Z	T	L
01♦LBL "WSBM"					
02 CF 06					
03 14					
04 GTO 03					
05♦LBL "LFBM"					
06 SF 06					
07 10					
08♦LBL 03					
09 CF 09					
10 XROM "S"					
11 GTO 03					
12♦LBL "CORBL"					
13 SF 09					
14 17					
15 XROM "W"					
16♦LBL 03					
17 CF 10					
18 XEQ "N"					
19 STO 09	f_c'				
20 FS? 06					
21 GTO 03					
22 FC? 04					
23 .4					
24 FS? 04					
25 .45					
26 *					
27 STO 12	$.45f_c'$				
28 8					
29 PROMPT	N				
30 X<> 09	f_c'				
31♦LBL 03					
32 FC? 09					
33 GTO D					
34 5					
35 /	$.2f_c'$				
36 .8	Absl				
37 X>Y?					
38 X<>Y	lesser				
39 STO 08					
40 .55					
41 FC? 04					
42 .45	WSD factors				
43 FC? 06					
44 ST* 08					
45 "MU 1,1.4"					
46 PROMPT					
47 STO 16	μ (friction factor)				
48♦LBL D					
49 .9					
50 FS? 09					
51 .85					
52 STO 05	ϕ				
53 XROM "BD"	D	B			
54 X=0?					
55 GTO D					
56 RCL 07					
57 "H="					
58 PROMPT					
59 STO 07	H				
60 RCL 04	d				
61 X↑2	d^2				
62 RCL 03	b				
63 *					
64 STO 10	bd^2				ϕbd^2 in R.05
65 ST* 05					
66 FS? 06					
67 GTO 00					
68 RCL 12	$.45f_c'$				
69 RCL 13	f_{st}				
70 RCL 09	N	f_{st}	$.45f_c'$		Ideal $k = \frac{all.f_c}{f_s/n + all.f_c}$
71 /	f_s/n				
72 RCL 12	$.45f_c'$				
73 +					
74 /	k				
75 ENTER↑	k	k			
76 CHS					
77 3	3	-k	k		
78 /	$-k/3$				
79 E	1				
80 +	j	k			$j = 1 - k/3$
81 *					
82 2					
83 /	$\frac{jk}{2}$				
84 RCL 12	$.45f_c'$				
85 *					
86 STO 14	R_{ideal}				Ideal $R = \frac{1}{2}(all.f_c)jk$
87♦LBL 00					
88 RCL 07	h				
89 RCL 04	d				
90 -	$h-d$				
91 FS? 09					
92 GTO 16					
93 RCL 07	h				
94 X↑2	h^2				
95 RCL 03	b				
96 *	bh^2				
97 XROM "Mc"	ρ_{min}				
98 RCL 03	b				
99 *	$\rho_{min}\cdot b$				
100 RCL 04	d	$\rho_{min}\cdot b$			
101 XROM "R"					
102 ARCL 02	$A_{s\text{-}min}$				
103 FC? 00					
104 AVIEW					
105 STO 08	A_{smin}				
106 SF 14					

D27

Program W5EM/11PP/... For Concrete Design 752 bytes Page 2 of 4

	X	Y	Z	T	L

```
107+LBL E
108 CF 07
109 XROM "LL"
110 CLA
111 CLX
112 XROM "M"
113 PROMPT
114 *
115 X=0?
116 GTO 37

117+LBL 07
118 "M/BD2="
119 RCL 03
120 .9
121 FC? 06
122 SIGN
123 RCL 04
124 X↑2
125 *
126 *
127 /
128  E3
129 *
130 ARCL X
131 FC? 00
132 FS? 09
133 X=0?
134 AVIEW
135 LASTX
136 /
137 FC? 06
138 GTO 05
139 X<>Y
140 XROM "U"
141 STO a
142 RCL 04
143 *
144 RCL 00
145 /
146 X<> a
147 "a="

148+LBL 19
149 FS? 09
150 RTN
151 ARCL a
152 "├""
153 FC? 00
154 AVIEW
155 .1
156 X>Y?
157 FS? 07
158 GTO 12
159 RDN
160 XEQ "RH"
161 RCL 03
162 R↑
163 *
164 RCL 04
165 *
166 CLA
167 XROM "AS"
168 XEQ "PY"
169 RCL 08
170 X<>Y
171 X>Y?
172 GTO E
173 BEEP
174 4
175 "4/3 "
176 3
177 /
178 *
179 XROM "AS"
180 XEQ "PY"
181 GTO E

182+LBL 12
183 "NG"
184 BEEP
185 AVIEW
186 GTO E

187+LBL 05
188 RCL 14
189 X>Y?
190 GTO 10
191 RCL Z
192 RCL 04
193 2
194 /
195 CHS
196 RCL 12
197 RCL 03
198 *
199 *
200 LASTX
201 6
202 /
203 XROM "Q"
204 X<>Y
205 STO a
206 ENTER↑
207 X↑2
208 RCL 03
209 RCL 09
210 /
211 *
212 "OVER"
213 RCL 04
214 R↑
215 -
216 X<=0?
217 SF 07
```

$a = A_s \cdot \frac{f_y}{.85 f'_c \cdot b}$

D28

Step	X	Y	Z	T	L
218 /	$\frac{y^2 b/n}{d-y}$				
219 2	2	$\frac{y^2 b/n}{d-y}$	y	y	
220 FC? 09					
221 FS? 00					
222 X=0?					
223 AVIEW					
224 GTO 08					
225♦LBL 10	R_{ideal}	R_{act}	M		
226 6					
227 R↑	M	6			
228 STO a					
229 *	6M				
230 RCL 09	N				
231 *	6nM				
232 RCL 13	F_{st}				
233 /					
234 RCL 03	b				
235 /					
236 STO \	$\frac{6nM}{bF_{st}}$				
237 RCL 04	d				
238 *					
239 STO]					
240 LASTX	d				
241 3					
242 *	3d				
243 STO [					
244 9					
245 /	d/3				
246 STO 10					
247 E					
248 %					
249 STO ↑	d/300	d/3	$\frac{6Mdn}{bF_{st}}$		
250 CLST					
251♦LBL 11					
252 RCL Z					
253 STO 11					
254 RCL 10	Y?				
255 ENTER↑					
256 ENTER↑					
257 ENTER↑					
258 RCL [	A				
259 -					
260 *					
261 RCL \	B				
262 -					
263 *					
264 RCL]	C				$x^3 - Ax^2 - Bx + C = 0$
265 +					
266 ST* ↑					
267 ST- 11					
268 RCL ↑					
269 RCL 11					
270 X≠0?					
271 /					
272 STO ↑					
273 X<> 10					

Step	X	Y	Z	T	L
274 ST+ 10					
275 RND					
276 RCL 10					
277 RND					
278 X≠Y?					
279 GTO 11					
280 ENTER↑	Y	Y			
281 X<> a	M	Y			
282 RCL 04	d	M	Y		
283 RCL Z	Y	d	M		
284 3					
285 /	Y/3	d	M	M	
286 -	jd	M			$A_s = \frac{M}{f_{st} j d}$
287 /	M/jd				
288 RCL 13	f_{st}				
289♦LBL 08					
290 /	A_s				
291 RCL 03	b				
292 /					
293 RCL 04	d				
294 /	ρ				
295 "Y="					
296 GTO 19					
297♦LBL 16					
298 STO 02	h"-d"				
299 RCL 01	F_y				
300 .85					
301 *	$.85F_y$				
302 FC? 06					
303 RCL 13	F_s				
304 STO 17	F_{sv}				
305 "ARM""					
306 PROMPT					
307 STO 15	Arm				
308 RCL 04	d				
309 X>Y?					
310 SF 14					
311 X>Y?					
312 GTO 34					
313 "ARM > D"					
314 BEEP					
315 PROMPT					
316 GTO D					
317♦LBL 34					
318 XROM "L"					
319 CLX					
320 "V, K"					
321 PROMPT					
322 X=0?					
323 GTO D					
324 STO 06	V				
325 RCL 08	allow. f_v				
326 RCL 03	b				
327 *					
328 RCL 04					

Program WSBM, etc. For Concrete Design 752 bytes Page 4 of 4

Step	X	Y	Z	T	L	
329 *	V_c	V				
330 "V>Vc"						
331 X<=Y?						
332 BEEP						
333 X<=Y?						
334 AVIEW						
335 X<=Y?						
336 GTO D						
337 CLX						
338 "TENS Nc, K"						
339 PROMPT						
340 STO 07	N_c					
341 "Nc>V"						
342 RCL 06	V	N_c				
343 X<=Y?						
344 BEEP						
345 X<=Y?						
346 AVIEW						
347 X<=Y?						
348 GTO 34						
349 5						
350 /	V/5	N_c				Min. N_c = V/5
351 X>Y?						
352 STO 07						
353 RCL 06	V					
354 RCL 17	f_{sv}					
355 /						
356 RCL 16	μ					
357 /						
358 "A-V="						
359 XEQ 01						
360 2						
361 *						
362 3						
363 /	$\frac{2}{3}A_v$					
364 X<> 06	V					$\frac{2}{3}A_v$ in R.06
365 RCL 15	Arm					
366 *	M_{vert}					
367 RCL 07	N_c					
368 RCL 02	h-d					
369 *	M_{nomin}					
370 +	M					
371 XEQ 07	ρ_M					
372 RCL 03	b					
373 *						
374 RCL 04	d					
375 *	A_f					
376 "A-F="						
377 XEQ 01						
378 RCL 07	N_c					
379 RCL 17	F_s					
380 /	A_n	A_f				
381 "A-N="						
382 XEQ 01						
383 +	A_n+A_f					
384 RCL 06	$\frac{2}{3}A_v$					
385 LASTX	A_n					
386 +	$\frac{2}{3}A_v+A_n$					

Step	X	Y	Z	T	L
387 X<=Y?					
388 X<>Y	Greater	Lesser			
389 LASTX	A_n				
390 X<>Y	Greater	A_n			
391 RCL 00	$.85f'_c/f_y$				
392 21.25	21.25	$.85f'_c/f_y$	Greater	A_n	
393 /	$.04f'_c/f_y$	Greater	A_n	A_n	
394 "RHO-MIN="					
395 ARCL X					
396 FC? 00					
397 AVIEW					
398 RCL 03	b				
399 *					
400 RCL 04	d				
401 *	$A_{s\,min}$	A_s	A_n		
402 X>Y?					
403 X<>Y	lesser	greater			
404 RDN	greater	A_n			
405 ADV					
406 "PRIM "					
407 XROM "AS"					
408 XEQ "PV"					
409 -	A_n-A_s				
410 CHS	A_s-A_n				
411 2					
412 /	$\frac{1}{2}(A_s-A_n)$				
413 "SEC "					
414 XROM "AS"					
415 XEQ "PV"					
416 GTO 34					
417♦LBL 37					
418 RCL 02	$.75\rho_b$				
419 STO 10					
420 RCL 03					
421 XEQ "VS"					
422 RCL 10					
423 STO 02					
424 GTO D					
425♦LBL 01					
426 ARCL X					
427 XROM "IS"					
428 FC? 00					
429 XEQ "PV"					
430 END					

LOAD FACTOR (ULTIMATE STRENGTH) CONCRETE BEAM ANALYSIS

xeq LFAN (SIZE 029)

SOFTKEYS (USER mode ON)

A	All new reinforcing input
C	Correct previous layer of reinforcing input
D	New beam dimensions
E	New acting moment

USER FLAGS

Flag 00	clear	Full output
	set	Only moment capacity and reinforcing limitation warnings
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 04	clear	AASHTO
	set	ACI

INPUT SUMMARY

Prompt	Input	Default
AASHTO or ACI	A reminder, not a prompt. Press R/S to continue if correct.	
F-Y 40,60	Steel yield stress in ksi	60 ksi
Fc' KSI	Concrete ultimate strength	4.0 ksi
FL W"	Top flange width in inches	No default
WEB W"	Web width in inches. Bypass for a rectangular beam. WEB W" is greater than FL W" for an invert-T beam.	FL W"
FL TH"	Top flange thickness in inches. Only prompted if WEB W" is different than FL W".	No default
H"	Total depth of the member (full concrete depth, not depth to reinforcing), in inches	No default

D31

Prompt	Input	Default
Z 130,170 (AASHTO) Z 145,175 (ACI)	Allowable Z-cracking constant in kips per inch	130
LF M,'K	Factored moment in foot-kips, <u>always positive</u>. Note: the program applies the under-capacity factor phi = 0.90.	No default
SERV M,'K	Service moment in foot-kips, <u>always positive</u>	No default
MIN FATG M,'K (AASHTO only)	Minimum moment for fatigue considerations (<u>Service DL + LL</u>). If reversal takes place, this is negative. Bypass or input zero to skip fatigue calculations.	Zero
A-S1	Area of first layer of tension reinforcing in square inches. Bypass to access shear design.	Shear design
D"	Depth to reinforcing, in inches	No default
A-S2, A-S3, etc.	Area of second (third, etc.) layer of tension reinforcing. Bypass (or enter zero) to end reinforcing input. Layers need be in no particular order.	Zero
MAX BAR #	Size of the largest tension bar; e.g. #8 for a 1" diameter bar	No default
A-S'	Area of compression reinforcing.	Zero
D-TOP"	Depth to compression steel	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(LF: EST A-S)	Estimated reinforcing necessary for the load factor moment (LF M) input
(WS: EST A-S)	Estimated reinforcing needed for the service moment (SERV M) input
(RHO)	The reinforcing ratio
P/Pb	Compares the actual reinforcing ratio to the "balanced" ratio.
AS<MIN	Warns that the area of steel is less than the calculated minimum. Design is still OK if the area of steel is greater than 4/3 A_{s-req}.

(Ie)	The "effective" moment of inertia based on service moment acting
(C)	The neutral axis location, measured from the top down
LF M	The "nominal" ultimate strength of the member, O M_u, in foot-kips.
SERV M	The allowable service moment on the member based on Z-cracking considerations, in ft-kips
(Z STRS)	The allowable service reinforcing stress in ksi, used to figure SERV M
(ACT STRS)	The acting service reinforcing stress in ksi
(Z)	The acting Z value
(ALL FAT STRS)	The allowable fatigue stress on the reinforcing, in ksi
(ACT FAT STRS)	The actual fatigue stress on the reinforcing
(OK or NG)	Approval or disapproval of the design, based on ultimate and service strength and fatigue. Does not look at maximum or minimum steel areas or constructibility of the design.

PROGRAM FLAGS

Flag 05	clear	True T-beam, based on neutral axis depth
	set	Rectangular, or neutral axis in flange
Flag 06	clear	Ic (WSD)
	set	LFAN (LFD)
Flag 07	clear	OK
	set	NG
Flag 08	clear	T-beam, based on configuration
	set	Rectangular beam
Flag 09	clear	do Z-cracking calculation
	set	don't do Z-cracking calculation
Flag 10	clear	All concrete properties in Fc

STORAGE REGISTER USE

Register	Value
00	.85 Fc'/F_y
01	F_y, ksi

Register	Value
02	$.75\ P_{bal}$ for T-beam
03	Web width, inches
04	Maximum reinforcing depth
05	$\phi\ bd^2$
06	$\sqrt{Fc'}/253$ for 1.2 M_{cr} calculation
07	Flange width
08	Present A_{st}; D' for compr. steel
09	N
10	Flange thickness
11	H (full beam depth)
12	Y (neutral axis position) from "Ic"
13	Z
14	H - 3" for reinforcing estimates; $\sum A_{st}$
15	"c" for $.75P_{bal}$
16	S_{st} (reinforcing section modulus)
17	I_{cr}
18	B_1 ($beta_1$)
19	Min A_{st}
20	counter; A_s'
21	$\sum A_{st} \times D$
22	M_{cr}, inch-kips
23	I_g (gross moment of inertia of the concrete alone)
24	$.75\ P_{bal}$ for rectangular beams
25	SERV M (inch-kips stored)
26	LF M (inch-kips stored)
27	MAX BAR #
28	M_{min} for fatigue

LOAD FACTOR (ULTIMATE STRENGTH) CONCRETE BEAM ANALYSIS

Program LFAN analyzes reinforced beams by ultimate strength methods, calculating the cracked-section moment capacity of a given beam configuration for different reinforcing options. It accepts rectangular, T, or invert-T beams, with or without compression steel (one layer only), and multiple layers of tension steel. It is not able to distinguish bars input as tension steel which are above the neutral axis, so don't include too many "side bars" as tension reinforcing. The program also calculates the gross moment of inertia (I_g) of the concrete section, and the "effective" moment of inertia (I_e) as defined by AASHTO 8.13.3 and ACI 9.5.2.3, using the service moment input.

AASHTO/ACI MINIMUM A_s

Like most of the structural programs, LFAN follows AASHTO if flag 04 is clear and ACI if it is set. The governing code is displayed at the beginning of each run. If it is the desired code, press R/S to continue. (If using a printer, you won't need to press R/S--it continues automatically.) If designing by the other code, change the status of flag 04 and start over. AASHTO calculates the minimum area of steel by means of a cracking-moment concept rather than with 200/Fy as ACI does. If designing by ACI, the $200/F_y$ ratio is based on average depth to reinforcing for multiple layers of reinforcing, and on the web width for T-beams. If working with AASHTO, on the other hand, the minimum reinforcing is that necessary for a "cracking moment" based on the concrete gross moment of inertia, assuming a homogenous material with a strength of 7.5 $\sqrt{f_c'}$ (the "modulus of rupture" of concrete per AASHTO 8.15.2.1). This reinforcing is calculated by ultimate strength methods, using an effective depth to reinforcing 3" less than the full concrete depth of the member, and is the same for both concrete flexural analysis programs, Ic and LFAN. AASHTO is not very clear in its present form that ultimate strength methods are to be used for all designs, WSD as well as LFD; however, the 1982 Interim Specifications, where it was introduced, spell it out a little better. Besides, it doesn't make sense to have a different minimum for WSD than for LFD. If the actual reinforcing is less than the minimum, the program displays AS<MIN; if the area of steel is greater than 4/3 of what is needed, it still meets code requirements.

MAXIMUM REINFORCING

LFAN also checks the reinforcing against a maximum of $0.75P_b$ (75% of the "balanced" reinforcing ratio based on ultimate-strength concepts) as required by both AASHTO and ACI. This check does take compression reinforcing into account, as well as the effect of T-beam flanges.

MOMENT CAPACITY

The moment capacity is found by calculating the moment causing full allowable stress in the tension steel only, as opposed to program Ic, which checks concrete stresses as well as tension and compression steel stresses. This is because, in ultimate strength design, concrete stresses will not exceed the allowable if the reinforcing ratio is held below $.75P_b$. If the reinforcing ratio goes above this maximum, the program displays a warning.

The output moment capacity (LF M=...) includes the undercapacity factor Phi = 0.90.

At the end of a run, the program displays OK or NG (no good) in reference to the beam configuration just run. This reflects ultimate and fatigue capacity and service load (Z-cracking) capacity. It does not refer to maximum or minimum steel checks or deflection. In other words, a beam with too much reinforcing will effect a warning AS>MAX during the run but may well display OK at the end. This means it has adequate ultimate, cracking, and fatigue strengths; however, the beam is not acceptable.

REINFORCING ESTIMATES

After displaying the maximum and minimum steel calculations, LFAN prompts for the acting load factor and service moments, as well as minimum moment for fatigue. Based on the load factor and service moments, it estimates the amount of reinforcing required for each; these two quantities usually bracket the actual reinforcing required. If fatigue is a consideration, or if the Z-cracking requirement is strict (Z = 130 or 145), the actual steel required will be closer to the WSD estimate. If the Z required is higher (170 for AASHTO or 175 for ACI for "moderate" exposure), the area of steel required will be closer to the ultimate requirement. Using multiple layers of steel, of course, raises the required area since the reinforcing is being used less efficiently. The program bases these estimates on the following three assumptions:

1) One layer of reinforcing
2) Effective depth D = H - 3"

3) For a T-beam, the neutral axis is within the flange.

The LFD estimate is a true design, given these assumptions. The WSD estimate uses these rules of thumb (from the ACI Working Stress Handbook) where M is in footkips and D is in inches:

A_s = M/(1.76 D) for f_s = 24 ksi (Grade 60)
A_s = M/(1.24 D) for f_s = 20 ksi (Grade 40)

These two areas of reinforcing are, of course, only estimates, but are often useful as a place to start or to check if a given beam configuration is reasonable for the load.

FATIGUE AND CRACKING

LFAN calculates the elastic (cracked) section properties of the given beam configuration in order to evaluate service (Z-cracking) and, under AASHTO (flag 04 clear), fatigue properties. These calculations are done by the working portion of program Ic. The section properties are not displayed by LFAN; however, they are stored and may be recalled by the user. The program checks reinforcing fatigue only; concrete fatigue (0.5 f_c') is not checked. According to ACI 343, no failures have ever been attributed to fatigue of concrete; and it never seems to govern. However, if felt necessary, this should be checked by the designer.

Fatigue calculations involving stress reversal are complicated in concrete analysis, since it is not correct to simply use the same cracked-section properties with signs reversed. Any stress calculations on tension steel under stress reversal usually involve assumptions concerning the opposite layer of reinforcing (formerly compression steel, now in tension). LFAN cavalierly makes a number of such assumptions, all based on concrete bridge slab design:

1) There is only a single layer of steel on each side.

2) Whether or not compression steel was input, the "top" steel (normally in compression, now in tension) is assumed to be twice the area of the steel being investigated for fatigue. This is predicated on the assumption of equal top and bottom mats (which is approximately true for continuous slab design) where the mat under investigation has had half its bars cut off (since most moment reversal problems happen at bar cutoffs).

D37

3) Under moment reversal, the new "cracked" properties are not calculated precisely. The program assumes that k' = 0.3 and j' = 0.9 (where j' = 1 - k'/3).

Since the compressive force due to moment reversal is usually very low (on the order of 2 ksi or less) these assumptions are close enough. However, they really have no merit except in continuous slab design, so use negative MIN FATG M's with caution in other designs.

The program also assumes the reinforcing deformation ratio r/h is 0.8, meaning the maximum allowable fatigue stress is calculated as 21.0 + 3(r/h) - $(f_{min})/3$, or 23.4 ksi - $(f_{min})/3$. Under moment reversal, f_{min} is negative, thus raising the allowable fatigue stress.

USER FLAGS

The program displays full output (including limit checks, the neutral axis position, I_g, I_e, and fatigue and Z-cracking calculations) if flag 00 is clear. If flag 00 is set, the program displays only the ultimate and service moment capacity. The limit checks (minimum and maximum areas of steel) and their warnings are not disabled by setting flag 00; only the display of the actual minimums and maximums is suppressed. When flag 00 is set, some quantities (the moments) are not prompted to be input. Thus it works to <u>set</u> flag 00 in the middle of a run, but <u>clearing</u> it will cause problems.

All the programs have warning BEEPS for material property limits and some errors, which sound if flag 02 is set; however, LFAN does <u>not</u> have BEEPs for infractions concerning minimum steel areas, only displayed warnings. The program is unaffected by flag 01 or 03. As mentioned, flag 04 determines whether AASHTO (clear) or ACI (set) is followed.

SOFTKEYS

Softkeys are used extensively by LFAN for modification or correction of input. To access the beam <u>d</u>imension prompts, press softkey D (in USER mode); to input new moments, use softkey E (<u>e</u>xternal loads, maybe?). Softkey C (<u>c</u>orrect) returns the program to the previous reinforcing prompt (for multiple layers of tension steel) <u>while still in the tension steel input section</u>. Softkey A, on the other hand, starts the reinforcing input from the beginning (<u>A</u>-S1, etc.) and may be used any time the program pointer is actually within LFAN; i.e., any time except while in the shear design subroutine (see next paragraph) or the initialization or material prompting subroutines at the beginning of the program.

SHEAR DESIGN

All four of the flexural programs access a shear reinforcing design subroutine, VS. This is a true design program, not an analysis program. LFAN is diverted to shear design by bypassing the A-S1 prompt. Unlike most other "defaults", the prompt <u>must</u> be bypassed; it doesn't work to input zero; and it must be the AS-1 prompt. Control returns to the flexural program by bypassing the A-V prompt in the shear design subroutine. To use the shear design subroutine, the designer must go through the flexural program at least once, since VS uses the beam width and reinforcing depth from the calling program. VS also uses the Flag 06 setting from the calling program to determine whether to design in WSD or LFD. In program LFAN, softkey A can be used to get to the A-S1 prompt, and thence to shear design, as long as at least one flexural analysis has been done; VS will use the beam configuration last run. Note that shear design is by the simplified equations of both codes and does not include the effect of moment or external tension or compression; this was done both to save programming space and because most designers usually use only the simplified methods anyway. For more information on this subroutine, see the documentation for VST, of which VS is a part.

T-BEAMS

The analysis of T-beams has a few peculiarities worth mentioning. First of all, the derivation of $.75P_{bal}$ is based on the actual beam configuration. The neutral axis position is calculated for $.75P_b$, and if it is found to be below the flange then everything is recalculated. This is shown in the T-beam example, to which reference should be made.

Second, rather than use the distance "c" to calculate the position of the neutral axis, LFAN uses the dimension "a", where "a", the compression block depth, is B_1 x c. On most sections, this makes no difference. However, on those sections where "c" is slightly greater than the flange thickness and "a" is slightly less, the usual analysis has the flanges participating their full depth while the web is only effective to depth "a." In this situation, the program will have a different "a" somewhere between the two, and a slightly greater neutral-axis position "c." The difference is minimal practically but philosophically makes more sense. The "true" neutral axis position (i.e. that calculated by normal means) is still used to check compression steel yield and the value for $.75P_b$.

Invert T-beams are easily handled. In this case the FL W" (flange width) is less than the web thickness;

the flange is defined as always being in compression. The lesser of the two dimensions (flange or web width) is always used as the width for shear design.

ALLOWABLE STRESSES

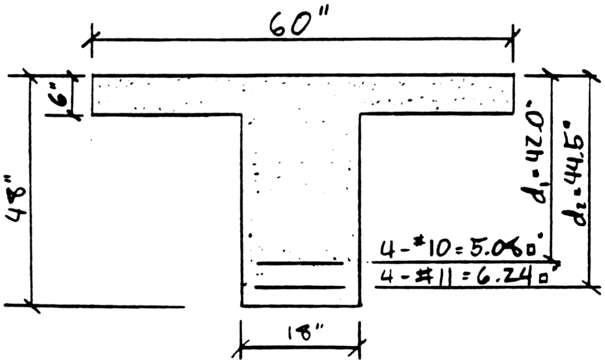
Program LFAN also has an idiosyncracy related to the value of N, the modular ratio, which is needed by the working-stress portion of the program but which is not input. LFAN calculates the modular ratio as N = 12-f_c' (in ksi), which is correct for the normal range of concrete strengths used. Note that using a concrete strength, for instance, of 3500 psi, results in an N value of 8.5; normal hand calculations would use 9. This may result in some discrepancies when checking hand calculations.

REINFORCING STRESSES under ULTIMATE LOAD

Tension steel yield is never checked directly since, as mentioned, yield must occur if the reinforcing ratio is less than P_b. However, this indirect check applies at the average depth to reinforcing; in beams with more than one layer of reinforcing, bars above this which may not be yielding are not checked. Practically, the limitation of 75% of P_b ensures yielding in most reasonable designs. However, bars placed as "side bars" in deeper beams may not yield. If deemed desirable to include these bars in an analysis, the designer must check that they reach yield strain; the neutral-axis depth is displayed for this purpose. If found not to yield, they should be left out, or their area factored down by the ratio of their actual stress to the yield stress.

Compression steel yield, on the other hand, is checked. If this steel is found not to yield, the program recalculates using a factored-down area as described in the preceding paragraph. This, of course, is an approximation, but the error is typically less than 1% and is conservative. Concrete displaced by compression steel is accounted for. By way of comparison, neglecting to do this is non-conservative, by about 10%.

Check the T-beam for the loads given by AASHTO W.S.D. criteria.



$f'_c = 4.0$ ksi
$n = 8$
$f_s = 24$ ksi (Grade 60)

Service DL = 700'·k
Service LL = 300'·k
1000'·k

Concrete

	A	d	Ad	$\bar{d}$	$A\bar{d}^2$	I_o
fl.	360	3"	1080	16.26	95,157	1080
web	756	27"	20,412	7.74"	45,313	111,132
	1116		21,492		140,470	112,212
					252,682 in⁴	

$\bar{d} = \frac{21,492}{1116} = 19.26"$

$S_{bottom} = \frac{252,682}{48"-19.26"} = 8791.4$

$f_r = 7.5\sqrt{f'_c} = 7.5\sqrt{4000} = 474.34$ psi

$1.2\,M_{cr} = 1.2 \times \frac{8791.4\,in^3 \times 474.34\,psi}{12,000} = \underline{417.01'\cdot k}$

$= 5004.15''\cdot k$

Calculate min. reinf. by ultimate str. methods:

$\rho = \frac{1}{m}(1-\sqrt{1-2mR_u/f_y}\,)$; $m = \frac{f_y}{.85f'_c} = 17.65$

$R_u = \frac{M}{\phi b d^2} = \frac{5,004.15''\cdot k}{0.9 \times 60 \times 45} = .0458$

Assumed (d = 45)

Thus, $\rho_{min} = \frac{1}{17.65} \times (1-\sqrt{1-2mR_u/60})$

$= .00077$

$A_{s\text{-}min} = .00077 \times \underline{60"} \times 45"$

$= \underline{2.07\,□"}$ - OK

Or, by ACI, $\rho_{min} = \frac{200}{F_y} = .0033$

$A_{s\text{-}min} = .0033 \times \underline{18"} \times 45"$

$= \underline{2.70\,□"}$ - OK
(much simpler)

<u>Check Max. A_s</u> $= .75\,\rho_{bal}$

Assume N.A. in flange @ max.:

$.75\rho_b = .75 \times \left(\frac{.85\beta_1 f'_c}{f_y} \cdot \frac{87}{87+f_y}\right) = .0214$

$A_{s\text{-}max?} = .0214 \times 60" \times 45 = 57.7\,□"$

$T = 57.7\,□" \times 60$ ksi $= 3463$ k
$C = 60" \times 6" \times (.85 \times 4) \times \beta_1 = 1040.4$ k

∴ N.A. is in web @ $.75\rho_{bal}$

Then, flange → equiv. A_s:

$(60-18) \times 6" \times \frac{(.85 \times 4.0\,ksi)}{60\,ksi} = 14.28\,□"$

Web: $\rho_b = \frac{.0214}{.75} \times 18" \times 45" = \underline{23.09\,□"}$

$\Sigma = 37.37\,□"$

$.75A_{s\text{-}bal} = .75 \times 37.37 = \underline{28.03\,□"}$

$.75\rho_{bal} = \frac{28.03}{60" \times 45"} = .0104$

(relative to flange width)

$A_s = (5.04 + 6.24) = 11.32 < 28.03$

<u>OK</u>

Section Properties

$60'' \times 6'' \cdot (6/2) = 1080\ in^3$

$(n=8) \cdot (5.08\,\square'' \cdot 36'' + 6.24\,\square'' \cdot 38.5'')$

$= 3385\ in^3 > 1080$

⇒ N.A. in web @ service

$\therefore (60-18) \cdot 6'' \cdot (X - \frac{6''}{2}) + 18X \cdot \frac{X}{2}$

$= (n=8)(5.08 \cdot (42-X) + 6.24(44.5-X))$

$252.0X - 756 + 9X^2 = (213.36 + 277.68 - 11.32X) \cdot 8$

$X = \underline{10.6785''}$ (Quadratic)

	Area	d	Ad^2	I_o
fl. $(60-18) \times 6$ =	252.0	7.68"	1934.98	756.0
web $18'' \times 10.68$ =	192.21	5.34"	5479.59	1826.53
$A_{s.1}$ 5.08×8 =	40.64	31.32"	39,869.22	–
$A_{s.2}$ $6.24 \cdot 8$ =	49.92	33.82	57,103.06	–
			117,309.77	2582.53

$I_{cr} = 119,892.30\ in^4$

Concrete: $S_c = \frac{I}{X} = 11,227.41\ in^3$

$M_c = \frac{(.40 \times 4\ ksi) \times 11,227.41}{12} = 1497.0'^k$

Steel: $S_{st} = \frac{119,892.30}{(44.5 - X) \cdot (n=8)} = 443.11\ in^3$

$M_{st} = \frac{24\ ksi \times 443.11}{12} = 886.21'^k$ ← Governs

$886.21 < 1000.0$ ⇒ **NG**

Section is inadequate by W.S.D.

```
           XEQ "Ic"
8:30:24 AM  02/17/89

TITLE:
T-BEAM                RUN
*************************
AASHTO
F-S 20,24
          24.0000     RUN
Fc'KSI
           4.0000     RUN
N
           8.0000     RUN
FL W"
          60.0000     RUN
WEB W"
          18.0000     RUN
FL TH"
           6.0000     RUN
H"
          48.0000     RUN
I-G=252,681.6777
1.2Mc=416.9831 'K
 A-S=2.0732 SQ"
200/FY=0.0033
.75Pb=0.0104
---------
SERV M,'K
       1,000.0000     RUN
MIN FATG M,'K
                      RUN

WS: EST A-S=12.5549 SQ"
-----------------------
A-S1
           5.0800     RUN
D"
          42.0000     RUN
A-S2
           6.2400     RUN
D"
          44.5000     RUN
A-S3
                      RUN
A-S'
                      RUN
RHO=0.0043

Ic=119,892.2980
Y=10.6785"
Sc=11,227.4086
S-ST=443.1073

ST M=886.2146 'K
Ie=125,463.8348

 NG
-----------------------
A-S1
```

D42

Check the same design for adequacy by AASHTO L.F.D criteria.

$f_c' = 4.0$ ksi
$F_y = 60$ ksi

$Z = 130 \frac{k}{in.}$ (severe exposure)

DL: 700
LL: $5/3 \times 300'^{k} = 500$
$1200 \times 1.3 = \underline{1560'^{k}}$

Strength

$(5.08 + 6.24) \times 60\,ksi \stackrel{?}{=} 60" \times (.85 \times 4.0\,ksi) \times a$

$a = 3.33"$

$c = \frac{a}{\beta_1 = .85} = 3.92" < 6" \Rightarrow$ N.A. in flange

$M = (\phi = .9)(5.08 \times (42 - \frac{3.33"}{2})$

$+ 6.24\square" \times (44.5 - \frac{3.33}{2})) \times \frac{60\,ksi}{12}$

$= \underline{2124.88'^{k}} > 1560'^{k}$ OK

Cracking

$Z = 130$; Max. bar = #11 = 1.56□"

$d = \frac{5.08 \times 42" + 6.24 \times 44.5"}{11.32\square"} = 43.38"$

equiv. number of #11 = $\frac{11.32}{1.56} = 7.256$ - #11 bars

$A_c = 2 \times (48 - 43.38") \times 18" / 7.256\,bars = 22.93\square"$

$d_c = 48" - 44.5" = 3.5"$

Allow. $f_s = \frac{Z = 130}{\sqrt[3]{3.5" \times 22.93}} = 30.14$ ksi

Act. $f_s = \frac{1000.0'^{k} \times 12}{S = 443.11} = 27.08$ ksi < 30.14 OK

Allow. serv. $M = \frac{30.14}{27.08} \times 1000'^{k} = \underline{1112.88'^{k}}$

Act. $Z = \frac{27.08}{30.14} \times 130 = 116.81$ k/in.

```
XEQ "LFAN"
8:34:40 AM  02/17/89

TITLE:
T-BEAM                  RUN
*************************
AASHTO
F-Y 40,60
            60.0000     RUN
Fc'KSI
             4.0000     RUN
FL W"
            60.0000     RUN
WEB W"
            18.0000     RUN
FL TH"
             6.0000     RUN
H"
            48.0000     RUN
I-G=252,681.6777
1.2Mc=416.9831 'K
 A-S=2.0732 SQ"
200/FY=0.0033
.75Pb=0.0104
---------
Z 130,170
           130.0000     RUN
LF M,'K
         1,560.0000     RUN
---------
SERV M,'K
         1,000.0000     RUN
MIN FATG M,'K
                        RUN

LF: EST A-S=7.9081 SQ"
WS: EST A-S=12.5549 SQ"
-----------------------
A-S1
             5.0800     RUN
D"
            42.0000     RUN
A-S2
             6.2400     RUN
D"
            44.5000     RUN
A-S3
                        RUN
MAX BAR #
            11.0000     RUN
A-S'
                        RUN
RHO=0.0043
Ie=125,463.8348
C=3.9170"

LF M=2,124.8799 'K
SERV M=1,112.8783 'K

Z STRS=30.1384 KSI
ACT STRS=27.0815 KSI
Z=116.8142

 OK
```

D44

WORKING STRESS CONCRETE BEAM ANALYSIS

xeq Ic (SIZE 029)

SOFTKEYS (USER mode ON)

A	All new reinforcing input
C	Correct previous layer of reinforcing input
D	New beam dimensions
E	New acting moment

USER FLAGS

Flag 00	clear	Full output
	set	Only moment capacity and reinforcing limitation warnings
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 04	clear	AASHTO
	set	ACI

INPUT SUMMARY

Prompt	Input	Default
AASHTO or ACI	A reminder, not a prompt. Press R/S to continue if not using a printer.	
F-S 20,24	Steel working stress in ksi	24 ksi
Fc' KSI	Concrete ultimate strength	4.0 ksi
N	Modular ratio, E_{st}/E_{conc}	8
FL W"	Top flange width in inches	No default
WEB W"	Web width in inches. Bypass for a rectangular beam. WEB W" is greater than FL W" for an invert-T beam.	FL W"
FL TH"	Top flange thickness in inches. Only prompted if WEB W" is different than FL W".	No default
H"	Total depth of the member (full concrete depth, not depth to reinforcing), in inches	No default

Prompt	Input	Default
SERV M,'K	Service moment in foot-kips, always positive	No default
MIN FATG M,'K (AASHTO only)	Minimum moment for fatigue considerations (Service DL + LL). If reversal takes place, this is negative. Bypass or input zero to skip fatigue calculations.	Zero
A-S1	Area of first layer of tension reinforcing in square inches. Bypass to access shear design.	Shear design
D"	Depth to A-S1, in inches	No default
A-S2, A-S3, etc.	Area of second (third, etc.) layer of tension reinforcing. Bypass (or enter zero) to end reinforcing input. Layers need be in no particular order.	Zero
D"	Depth to A-S2, etc.	No default
A-S'	Area of compression reinforcing. Bypass for no compression reinf.	Zero
D-TOP"	Depth to compression steel	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(WS: EST A-S)	Estimated reinforcing necesary for the design moment (SERV M) input
(RHO)	The reinforcing ratio
(Ic)	Cracked-section moment of inertia
(Y)	Neutral axis position measured from the compression side
(Sc)	Section modulus of the concrete
(S-ST)	Section modulus of the deepest layer of tension steel
ST M or CONC M	The service moment capacity, governed by steel or concrete, respectively
(Ie)	The "effective" moment of inertia
(ALL FAT STRS)	The allowable fatigue stress on the reinforcing, in ksi
(ACT FAT STRS)	The actual fatigue stress on the reinforcing

(OK or NG)	Approval or disapproval of the design, based on ultimate and service strength and fatigue. Does not look at maximum or minimum steel areas or constructibility of the design.

PROGRAM FLAGS

Flag 05	clear	True T-beam
	set	Rectangular, or neutral axis in flange
Flag 06	clear	Ic (WSD)
	set	LFAN (LFD)
Flag 07	clear	OK
	set	NG
Flag 08	clear	T-beam
	set	Rectangular beam
Flag 09	clear	do Z-cracking calculation
	set	don't do Z-cracking calculation
Flag 10	clear	All concrete properties in Fc

STORAGE REGISTER USE

Register	Value
00	$.85\ Fc'/F_y$
01	F_y, ksi
02	$.75\ P_{bal}$ for T-beam
03	Web width, inches
04	Maximum reinforcing depth
05	$\phi\ bd^2$
06	$\sqrt{Fc'}/253$ for 1.2 M_{cr} calculation
07	Flange width
08	Present A_{st}; D' for compr. steel
09	N
10	Flange thickness
11	H (full beam depth)

Register	Value
12	Y (neutral axis position) from Ic
13	Steel working stress, f_{st}
14	H - 3" for reinforcing estimates; $\sum A_{st}$
15	"c" for $.75P_{bal}$
16	S_{st} (reinforcing section modulus)
17	I_{cr}
18	β_1 (beta$_1$)
19	Min A_{st}
20	counter; A_s'
21	$\sum A_{st} \times D$
22	M_{cr}, inch-kips
23	I_g (gross moment of inertia of the concrete alone)
24	$.75\ P_{bal}$ for rect. beams
25	SERV M (inch-kips stored)
26	not used
27	MAX BAR #
28	M_{min} for fatigue

D48

WORKING STRESS CONCRETE BEAM ANALYSIS

Program Ic analyzes reinforced beams by working-stress methods, calculating the cracked-section properties and moment capacity of a given beam configuration for different reinforcing options. It accepts rectangular, T, or invert-T beams, with or without compression steel (one layer only), and multiple layers of tension steel. It is not able to handle bars input as tension steel which are above the neutral axis, so don't get greedy and include too many "side bars" as tension reinforcing. The program also calculates the gross moment of inertia (I_g) of the concrete section, and the "effective" moment of inertia (I_e) as defined by AASHTO 8.13.3 and ACI 9.5.2.3, using the service moment input.

AASHTO/ACI MINIMUM A_s

Like most of the structural programs, Ic follows AASHTO if flag 04 is clear and ACI if it is set. The governing code is displayed at the beginning of each run. If it is the desired code, press R/S to continue. (If using a printer, you won't need to press R/S--it continues automatically.) If designing by the other code, change the status of flag 04 and start over. AASHTO calculates the minimum area of steel by means of a cracking-moment concept rather than with 200/Fy as ACI does. If designing by ACI, the $200/F_y$ ratio is based on average depth to reinforcing for multiple layers of reinforcing, and on the web width for T-beams. If working with AASHTO, on the other hand, the minimum reinforcing is that necessary for a "cracking moment" based on the concrete gross moment of inertia, assuming a homogenous material with a strength of 7.5 $\sqrt{f_c'}$ (the "modulus of rupture" of concrete per AASHTO 8.15.2.1). This reinforcing is calculated by ultimate strength methods, using an effective depth to reinforcing 3" less than the full concrete depth of the member. AASHTO is not very clear in its present form that ultimate strength methods are to be used for all designs, WSD as well as LFD; however, the 1982 Interim Specifications, where it was introduced, spell it out a little better. Besides, it doesn't make sense to have a different minimum for WSD than for LFD. If the actual reinforcing is less than the minimum, the program displays AS<MIN; if the area of steel is greater than 4/3 of what is needed, it still meets code requirements.

MAXIMUM REINFORCING

Ic also checks the reinforcing against a maximum of $0.75P_b$ (75% of the "balanced" reinforcing ratio based

D49

on ultimate-strength concepts) as a matter of good practice, although neither AASHTO nor ACI requires it. This check does take compression reinforcing into account, as well as the effect of T-beam flanges.

MOMENT CAPACITY

The moment capacity is found by calculating the moment causing full allowable stress in the concrete, tension steel, and compression steel, if any. The least of these is the output capacity. The governing capacity is displayed by the program as either CONC M=... or ST M=... for concrete or steel, respectively. If concrete governs, it's usually best to increase the size of the beam or add compression reinforcing. Note that the concrete section modulus S_C already has been adjusted by N, the modular ratio. See the examples.

At the end of a run, the program displays OK or NG (no good) in reference to the beam configuration just run. This reflects fatigue capacity and service load capacity. It does not refer to maximum or minimum steel checks or deflection. In other words, a beam with too much reinforcing will effect a warning AS>MAX during the run but may well display OK at the end. This means it has adequate service and fatigue strengths; however, the beam is not acceptable.

REINFORCING ESTIMATE

After displaying the maximum and minimum steel calculations, Ic prompts for the acting service moment. Based on this, it estimates the amount of reinforcing required. Using multiple layers of steel, of course, raises the required area since the reinforcing is being used less efficiently. The program bases this estimate on the following assumptions:

1) One layer of reinforcing
2) Effective depth D = H - 3"
3) For a T-beam, the neutral axis is within the flange.

The WSD estimate uses these rules of thumb (from the ACI Working Stress Handbook) where M is in foot-kips and D is in inches:

$$A_S = M/(1.76\ D) \text{ for } f_S = 24 \text{ ksi}$$
$$A_S = M/(1.24\ D) \text{ for } f_S = 20 \text{ ksi}$$

This area of steel is, of course, only an estimate, but is often useful as a place to start or to check if a given beam configuration is reasonable for the load.

FATIGUE AND CRACKING

Ic calculates the elastic (cracked) section properties of the given beam configuration and, under AASHTO (flag 04 clear), fatigue properties. The program checks reinforcing fatigue only; concrete fatigue (0.5 f_c') is not checked. According to ACI 343, no failures have ever been attributed to fatigue of concrete; and it never seems to govern, even in heavily reinforced members. However, if felt necessary, this should be checked by the designer.

Fatigue calculations involving stress reversal are complicated in concrete analysis, since it is not correct to simply use the same cracked-section properties with signs reversed. Any stress calculations on tension steel under stress reversal usually involve assumptions concerning the opposite layer of reinforcing (formerly compression steel, now in tension). Ic cavalierly makes a number of such assumptions, all based on concrete bridge slab design:

1) There is only a single layer of steel on each side.

2) Whether or not compression steel was input, the "top" steel (normally in compression, now in tension) is assumed to be twice the area of the steel being investigated for fatigue. This is predicated on the assumption of equal top and bottom mats (which is approximately true for continuous slab design) where the mat under investigation has had half its bars cut off (since most moment reversal problems happen at bar cutoffs).

3) Under moment reversal, the new "cracked" properties are not calculated precisely. The program assumes that k' = 0.3 and j' = 0.9 (where j' = 1 - k'/3).

Since the compressive force due to moment reversal is usually very low (on the order of 2 ksi or less) these assumptions are close enough. However, they really have no merit except in continuous slab design, so use negative MIN FATG M's with caution in other designs.

The program also assumes the reinforcing deformation ratio r/h is 0.8, meaning the maximum allowable fatigue stress is calculated as 21.0 + 3(r/h) - $(f_{min})/3$, or 23.4 ksi - $(f_{min})/3$. Under moment reversal, f_{min} is negative, thus raising the allowable fatigue stress.

FLAGS

The program displays full output (including limit checks, cracked section properties, I_g, and I_e) if flag 00 is clear. If flag 00 is set, the program displays only the moment capacity. The limit checks (minimum and maximum areas of steel) and their warnings are not disabled by setting flag 00; only the display of the actual minimums and maximums is suppressed. All the programs have warning BEEPS for material property limits and some errors, which sound if flag 02 is set; however, Ic does not have BEEPs for infractions concerning minimum steel areas, only displayed warnings. The program is not affected by flags 01 and 03. As mentioned, flag 04 determines whether AASHTO (clear) or ACI (set) is followed.

SOFTKEYS

Softkeys are used extensively by Ic for modification or correction of input. To access the beam dimension prompts, press softkey D (in USER mode); to input new moments, use softkey E (external loads, maybe?). Softkey C (correct) returns the program to the previous reinforcing prompt (for multiple layers of tension steel) while still in the tension steel input section. Softkey A, on the other hand, starts the reinforcing input from the beginning (A-S1, etc.).

SHEAR DESIGN

All four of the flexural programs access a shear reinforcing design subroutine, VS. This is a true design program, not an analysis program. Ic is diverted to shear design by bypassing the A-S1 prompt. Unlike most other "defaults", the prompt must be bypassed; it doesn't work to input zero. And, it must be the A-S1 prompt. Control returns to the flexural program by bypassing the A-V prompt in the shear design subroutine. To use the shear design subroutine, the designer must go through the flexural program at least once, since VS uses the beam width and depth to reinforcing from the calling program. VS also uses the Flag 06 setting from the calling program to determine whether to design in WSD or LFD. In program Ic, softkey A can be used to get to the A-S1 prompt as long as at least one flexural analysis has been done; VS will use the beam configuration last run. Note that shear design is by the simplified equations of both codes and does not include the effect of moment or external tension or compression; this was done both to save programming space and because most designers usually use only the simplified methods anyway. For more information on this subroutine, see the documentation for VST, of which VS is a part.

D52

T-BEAMS

The analysis of T-beams has a few peculiarities worth mentioning. First of all, the derivation of $.75P_{bal}$ is based on the actual beam configuration. The neutral axis position is calculated for $.75P_b$, and if it is found to be below the flange then everything is recalculated. This is shown in the T-beam example, to which reference should be made.

Second, rather than use the distance "c" to calculate the position of the neutral axis, LFAN uses the dimension "a", where "a", the compression block depth, is B_1 x c. On most sections, this makes no difference. However, on those sections where "c" is slightly greater than the flange thickness and "a" is slightly less, the usual analysis has the flanges participating their full depth while the web is only effective to depth "a." In this situation, the program will have a different "a" somewhere between the two, and a slightly greater neutral-axis position "c." The difference is minimal practically but philosophically makes more sense. The "true" neutral axis position (i.e. that calculated by normal means) is still used to check compression steel yield and the value for $.75P_b$.

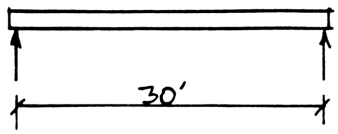
Invert T-beams are easily handled. In this case the FL W" (flange width) is less than the web thickness; the flange is defined as always being in compression. The lesser of the two dimensions (flange or web width) is always used as the width for shear design.

STRESSES

Program Ic (and the other working stress concrete programs, WSBM and VST) have a couple of idiosyncracies related to input of the allowable steel and concrete stresses. First, although the steel stress is a working stress (usually 20 or 24 ksi), the concrete stress to be input at the Fc'KSI prompt is the <u>ultimate</u> strength of the concrete, usually 3 to 4 ksi. The factor of .40 (AASHTO) or .45 (ACI) is applied <u>by the program.</u> Second, these programs (in subroutine N) calculate the reinforcing yield stress as Fy = 5 x (Fs-12 ksi), giving 40 and 60 ksi from 20 and 24 ksi respectively. If the designer is using a different allowable stress (for example, some tank wall designs are done at 14 or 16 ksi to hold down cracking),the calculated yield stress will be incorrect, invalidating the maximum and minimum reinforcing checks (all of which involve ultimate strength concepts and yield stresses).

Working Stress Bridge Slab

$f_c' = 4000\,psi$; $f_y = 60\,ksi$



Design a 30' slab for HS20. (24' Wide)

AASHTO App. A: $M = 242.1^{\prime k}/lane$

$Impact = 1 + \frac{50}{Span + 125} = 1.32 > 1.30$

$\therefore Impact = 1.30$

Distribution width (AASHTO 3.24.3.2)

$E = (4 + .06S) = 4 + .06 \times 30 = \underline{5.8'} < 7.0'$

$\therefore M = \frac{242.1^{\prime k}/lane}{2\ wheel/lane} / 5.8' = 24.319^{\prime k}/'$

check: $M \doteq 900S = .9 \times 30' = 27^{\prime k}/'$ OK

$\therefore LL\ M = 1.30 \times 24.319 = \underline{\underline{31.61^{\prime k}/'}}$

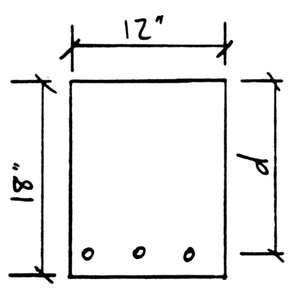
AASHTO 8.9.2 recommends $h \doteq 1.1 \times \frac{S + 10}{30}$

$h = 1.1 \times \frac{30' + 10}{30} / 12 = 17.6''$

USE 18" SLAB

DL per foot: Slab $1' \times 1.5' \times .15 = .225^{k}/'$
Rail $.44^{k}/' \div 13.67' = .032$
Topping $.02\,ksf \times 12'/13.67' = .018$
$.275^{k}/'$

DL $M = .275 \times \frac{30^2}{8} = 30.94^{\prime k}/'$
LL $M = 31.61$
$62.55^{\prime k}/'$



$I_g = 12 \times \frac{18^3}{12} = 5832\ in^4$

$S_g = \frac{5832}{9''}\quad 648.0\ in^3$

Check cracking moment:

$M_{cr} = f_r \cdot S_g$

$f_r = 7.5\sqrt{f_c'} = 7.5 \times \sqrt{4000} = 474.34\ psi$

$1.2\,M_{cr} = 1.2 \times .47434 \times 648.0\ in^3$

$= 368.85^{\prime\prime k} = \underline{\underline{30.7373^{\prime k}}}$

By ultimate strength methods,

$\rho = \frac{1}{m}\left(1 - \sqrt{1 - 2mR_u/f_y}\right)$

$m = \frac{f_y}{.85 f_c'} = 17.6471$

$R_u = \frac{M}{\phi b d^2}$; estimate $d = 18'' - 3'' = 15''$

$R_u = \frac{30.737 \times 12}{.9 \times 12'' \times 15^2} = .1518\ ksi$

$\rho = \frac{1}{17.6471} \cdot \left(1 - \sqrt{1 - 2 \times 17.65 \times .15/60}\right)$

$= .0026$

$Min\ A_s = \rho b d = .0026 \times 12 \times 15$

$= \underline{\underline{.4660\ \square''/'}}$

Check by ACI: $\rho = 200/F_y$

$Min.\ A_s = \frac{200}{60{,}000} \times 12 \times 15 = .60\ \square''/'$ OK

Estimate $A_{s\,req}$

$Approx.,\ A_s = \frac{M^{\prime k}}{1.76 \cdot d''} = \frac{62.55^{\prime k}}{1.76 \times 15''}$

$= 2.37\ \square''$

Try #10 @ 6" ; $A_s = 1.27\square'' \times \frac{12'}{6''}$

$= 2.54\,\square''/'$

Use $d = 18'' - 1.5''_{clear} - \frac{\#10}{8} \times \frac{1}{2} = 15.9''$

Check against min:

$2.54 > .4660^{\square''}$ OK

Check against max:

$$\rho_{bal} = \frac{.85\beta_1 f_c'}{f_y}\left(\frac{87}{87+f_y}\right)$$

for $f_c' = 4000$, $\beta_1 = .85$

$$\therefore .75\rho_{bal} = .75 \times \frac{.85 \times .85 \times 4}{60} \times \left(\frac{87}{87+60}\right)$$

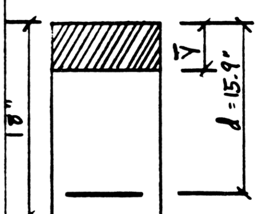
$$= .0214$$

Max $A_s = .0214 \times 12 \times 15.9'' = 4.08 \square''$

$2.54 \square'' < 4.08 \square''$ OK

Calculate Section Properties (elastic)

Use $n = 8$ for $f_c' = 4000$



$$\bar{Y} = \frac{n \cdot A_s}{b}\left(\sqrt{1 + \frac{2bd}{nA_s}} - 1\right)$$

$$= \frac{8 \cdot 2.54}{12}\left(\sqrt{1 + \frac{2 \cdot 12 \cdot 15.9}{8 \times 2.54}} - 1\right)$$

$$= 5.8376''$$

$$I_{cr} = 12'' \cdot \frac{(5.8376)^3}{3} + (n=8) \times 2.54 \times (15.9 - 5.8376)^2$$

$$= 2853.16 \text{ in}^4$$

S_{conc} : $2853.16 / 5.8376'' = 488.76 \text{ in}^3$

$$S_{ST} = \frac{2853.16}{(n=8) \times (15.9 - 5.8376'')} = 35.443 \text{ in}^3$$

Either concrete or steel may govern:

$M_{st} = 24 \text{ ksi} \times 35.443/12 = 70.89'^{k}$

check: $M_c = (.40 \times 4) \times 488.76/12 = 65.17'^{k} > 62.55'^{k}$ OK

D56

```
           XEQ "Ic"
 1:46:50 PM  02/07/88

TITLE:
BRIDGE SLAB          RUN
------------------------
AASHTO
N
            8.0000   RUN
F-S 20,24
           24.0000   RUN
Fc'KSI
            4.0000   RUN
FL W"
           12.0000   RUN
WEB W"
                     RUN
H"
           18.0000   RUN
I-G=5,832.0000
1.2Mc=30.7352 'K
 A-S=0.4660 SQ"
200/FY=0.0033
.75Pb=0.0214
------------------------
A-S1
            2.5400   RUN
D"
           15.9000   RUN
A-S2
                     RUN
A-S'
                     RUN
RHO=0.0133

Ic=2,853.1635
Y=5.8376"
Sc=488.7539
S-ST=35.4435

CONC M=65.1672 'K
------------------------
A-S1
             .4000   RUN
D"
           16.0000   RUN
A-S2
                     RUN
A-S'
                     RUN
RHO=0.0021
AS<MIN
---------

Ic=644.7407
Y=2.6667"
Sc=241.7778
S-ST=6.0444

ST M=12.0889 'K
------------------------
A-S1
```

Check fatigue (AASHTO)

Since this is a simple span, $M_{min} = M_{DL}$ and moment range = M_{LL}.

$f_{min} = (30.94^{'k} \times 12) / 35.443 in^3$

$= 10.475$ ksi

Allowable $f_f = 21 - \frac{f_{min}}{3} + 8(r/h)$

use $(r/h) = 0.3$ (deformation $\frac{radius}{height}$)

$f_f = 21 - \frac{10.475}{3} + 4 \times 0.3 = \underline{19.9082 \text{ ksi}}$

Actual stress range $= \frac{(31.61 \times 12)}{35.443} = \underline{10.7023 \text{ ksi}}$

Calculate $I_{eff} = (\frac{M_{cr}}{M_{act}})^3 \times I_g + (1 - \frac{M_{cr}}{M_{act}})^3 \times I_{cr} \leq I_g$

$I_e = (\frac{30.7373/1.2}{62.55})^3 \times 5832.0 + (1 - (\ )^3) \times 2853.16 = \underline{3057.72}$

Check design for same loads by ACI

Minimum $\rho = \frac{200}{F_y} = .0033$

$.0033 \times 12'' \times 15''_{say} = .60 \square'' < 2.54 \square''$ OK

Section properties are the same

$M_{ST} = 24 \text{ ksi} \times 35.443 in^3 / 12 = 70.49^{'k}$

$M_{CONC} = (.45 \times 4 \text{ ksi}) \times \frac{488.76}{12} = 73.31^{'k}$

Steel governs @ $\underline{70.49^{'k}} > 62.55$ OK

ACI has no fatigue req'ments

$.75 p_b = .0214$; act. $\rho = \frac{2.54}{12 \times 15.9}$ .0133 < .0214 OK

```
                    SF 04
                 XEQ "Ic"
  7:42:48 PM  02/07/88

TITLE:
ACI WSD EXAMPLE       RUN
-----------------------
ACI
N
           8.0000     RUN
F-S 20,24
          24.0000     RUN
Fc'KSI
           4.0000     RUN
FL W"
          12.0000     RUN
WEB W"
                      RUN
H"
          18.0000     RUN
I-G=5,832.0000
1.2Mc=30.7352 'K
200/FY=0.0033
 A-S=0.6000 SQ"
.75Pb=0.0214
---------
SERV M,'K
          62.5500     RUN

WS: EST A-S=2.3559 SQ"
-----------------------
A-S1
           2.5400     RUN
D"
          15.9000     RUN
A-S2
                      RUN
A-S'
                      RUN
RHO=0.0133

Ic=2,853.1635
Y=5.8376"
Sc=488.7539
S-ST=35.4435

ST M=70.8870 'K
Ie=3,057.6799

 OK
-----------------------
A-S1
```

Re-design bridge by LFD.

AASHTO: DL = 30.94
LL = 5/3 × 31.61 = 52.68
43.62 (sum as written)
× 1.3
108.71

Steel req'd for ultimate strength:

$$\rho = \frac{1}{m}\left(1-\sqrt{1-2mR_u/f_y}\right)$$

$$m = \frac{f_y}{.45f_c'} = \frac{60}{.45 \times 4} = 17.6471$$

$$R_u = \frac{M}{\phi b d^2} = \frac{108.71 \times 12}{.9 \times 12'' \times 15.9^2}$$

$$= .4778$$

$$\therefore \rho = \frac{1}{17.65}\left(1-\sqrt{1-2 \times 17.65 \times .4778/60}\right)$$

$$= .0086$$

$$A_s = .0086 \times 12 \times 15.9 = 1.644 \,\square''$$

Positive moment reinf. - bottom of slab ⇒ Use $z = 170$

∴ Expect A_s to be closer to LFD than WSD value (2.37 □")

Try #10 @ 9"; $A_s = 1.69$ □"/'

Calculate elastic properties:

$$\bar{Y} = \frac{n \cdot A_s}{b}\left(\sqrt{1+\frac{2bd}{nA_s}}-1\right) = 4.9641''$$

$$I_{cr} = 12'' \times \frac{4.9641^3}{3} + (n=8) \times 1.69 \times (15.9-4.9641)^2$$
$$= 2106.22 \text{ in}^4$$

$$S_{ST} = \frac{2106.22}{(15.9-4.9641) \times 8} = 24.0746 \text{ in}^3$$

Fatigue

$$f_{min} = \frac{M_{min}}{S_{ST}} = \frac{30.94'^{k} \times 12}{24.0746}$$
$$= 15.42 \text{ ksi}$$

$$\text{allow. } f = 21.0 - \frac{f_{min}}{3} + 8(r/h)$$

Use $(r/h) = 0.3$

∴ allow. f = 18.26 ksi

$$\text{Act. Fatg. stress} = \frac{\text{Mom Range}}{S_{st}}$$
$$= \frac{31.61'^{k} \times 12}{24.0746} = 15.756 \text{ ksi} \quad \text{OK}$$

Ultimate strength

$$12'' \times (.85 \times 4.0\text{ ksi}) \times a = 1.69\,\square'' \times 60\text{ ksi}$$

$$a = 2.4853''$$

$$c = \frac{a}{\beta_1} = \frac{2.4853}{0.85} = 2.9239''$$

$$\text{Ult. } M = \phi \times (A_s f_y) \times (d - a/2)$$
$$= .9 \times (1.69 \times 60) \times (15.9 - \frac{2.4853}{2})/12$$
$$= 111.47'^{k} > 108.71'^{k} \quad \text{OK}$$

Z-Cracking

$$A_e = 2 \times (18'' - 15.9) \times 12''/(\tfrac{12}{9}\text{ bars per foot})$$
$$= 37.80 \,\square''/\text{bar}$$

$$d_c = 18'' - 15.9'' = 2.10''$$

$$\text{Allow. } f_s = \frac{z = 170}{\sqrt[3]{37.80 \times 2.10}} = 39.556 \text{ ks}$$

⇒ Use f_s = 36.0 ksi

$$\text{Act. } f_s = \frac{62.55'^{k} \times 12}{24.0746} = 31.1781 \quad \text{OK}$$

$$\text{Act. } z = \frac{31.1781}{36.0} \times 170 = 147.23$$

Service Moment

Allowable $f_s = 36.00$ ksi

∴ Allowable serv. $M = f_s \cdot S_{ST}$

$= 36.0 \cdot 24.0746/12 = \underline{72.22}$ ^k^

OK - Use #10 @ 9"

```
              CF 04
         XEQ "LFAN"
SIZE<28
           SIZE 028
         XEQ "LFAN"
 3:13:18 PM  02/08/88

TITLE:
BRIDGE SLAB--LFD     RUN
-----------------------
AASHTO
F-Y 40,60
         60.0000     RUN
Fc'KSI
          4.0000     RUN
FL W"
         12.0000     RUN
WEB W"
                     RUN
H"
         18.0000     RUN
I-G=5,832.0000
1.2Mc=30.7352 'K
 A-S=0.4660 SQ"
200/FY=0.0033
.75Pb=0.0214
---------
Z 130,170
        170.0000     RUN
LF M,'K
        108.7100     RUN
---------
SERV M,'K
         62.5500     RUN
MIN FATG M,'K
         30.9400     RUN
RANGE,'K
         31.6100     RUN

LF: EST A-S=1.7629 SQ"
WS: EST A-S=2.3559 SQ"
-----------------------
A-S1
          1.6900     RUN
D"
         15.9000     RUN
```

Size checked by program

try again

estimates

```
A-S2
                    RUN
MAX BAR #
         10.0000    RUN
A-S'
                    RUN
RHO=0.0089
Ie=2,362.0159
C=2.9239"

LF M=111.4692 'K
SERV M=72.2237 'K

Z STRS=36.0000 KSI
ACT STRS=31.1781 KSI
Z=147.2301

ALL FAT STRS=18.2593 KSI
ACT FAT STRS=15.7561 KSI

 OK
-----------------------
A-S1
          4.2000    RUN
D"
         15.9000    RUN
A-S2
                    RUN
MAX BAR #
         11.0000    RUN
A-S'
                    RUN
RHO=0.0220
P/Pb=0.7722
---------
Ie=4,156.7368
C=7.2664"

LF M=242.1424 'K
SERV M=170.7604 'K

Z STRS=36.0000 KSI
ACT STRS=13.1869 KSI
Z=62.2715

ALL FAT STRS=21.2257 KSI
ACT FAT STRS=6.6641 KSI

 NG
-----------------------
```

Too much steel

Check design for same loads by ACI

DL: $30.94 \times 1.4 = 43.32$
LL: $31.61 \times 1.7 = 53.74$
$97.06^{\prime k}$

Ultimate strength

Same: $M_{cap} = 111.47 > 97.06^{\prime k}$ OK

Fatigue

ACI has no fatigue requirements

Z-Cracking

$A_c = 37.80 \, \square''/\text{bar}; \; d_c = 2.10''$

Ext. exposure ⇒ Use Z = 145

$$\text{Allow. } f_s = \frac{145}{\sqrt[3]{37.80 \cdot 2.10}} = 33.74 \text{ ksi}$$

$$\text{Act. } f_s = \frac{62.55 \cdot 12}{24.0746 \text{ in}^3} = 31.18 \text{ ksi} \quad \text{OK}$$

$$\text{Act. } Z = \frac{31.18}{33.74} \cdot 145 = 133.99$$

$$\text{Allow. Serv. } M = \frac{33.74}{31.18} \times 62.55 = 67.69^{\prime k}$$

Area of steel could probably be reduced for ACI design

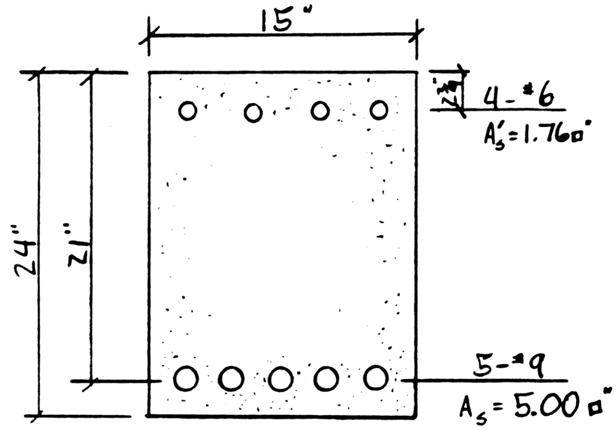
```
                 SF 04
            XEQ "LFAN"
10:13:58 AM  02/09/88

TITLE:
ACI LFD EXAMPLE      RUN
-----------------------
ACI
F-Y 40,60
         60.0000     RUN
Fc'KSI
          4.0000     RUN
FL W"
         12.0000     RUN
WEB W"
                     RUN
H"
         18.0000     RUN
I-G=5,832.0000
1.2Mc=30.7352 'K
200/FY=0.0033
 A-S=0.6000 SQ"
.75Pb=0.0214
---------
Z 145,175
        145.0000     RUN
LF M,'K
         97.0600     RUN
---------
SERV M,'K
         62.5500     RUN

LF: EST A-S=1.5567 SQ"
WS: EST A-S=2.3559 SQ"
-----------------------
A-S1
          1.6900     RUN
D"
         15.9000     RUN
A-S2
                     RUN
MAX BAR #
         10.0000     RUN
A-S'
                     RUN
RHO=0.0089
Ie=2,362.0159
C=2.9239"

LF M=111.4692 'K
SERV M=67.6430 'K

Z STRS=33.7168 KSI
ACT STRS=31.1781 KSI
Z=134.0826

 OK
-----------------------
A-S1
```

Check adequacy of this beam by Strength Design.

Span = 32'-0

$f_c' = 3.5$ ksi
$F_y = 60$ ksi
$n = 8.5$ for elastic prop.

Design for interior exposure by ACI → $z = 175$



$d = 24'' - \underset{cl.}{2''} - \underset{stir.}{\tfrac{1}{2}''} - \underset{bar}{\tfrac{1}{2}''} = 21''$

$d' = 2\tfrac{3}{4}''$

Loads

Beam $\frac{15'' \times 24''}{144} \times .15 \times \frac{32^2}{8} = 48.00^{\prime k}$

Partitions, Floor, etc.: $.70^{k/\prime} \times \frac{32^2}{8} = 89.60$

DL = $137.60^{\prime k}$

Live Load $0.75^{k/\prime} \times 32^2/8 = 96.00^{\prime k}$

Load	Service		Ultimate
DL	137.60	× 1.4 =	192.64
LL	96.00	× 1.7 =	163.20
Σ	$233.60^{\prime k}$		$355.84^{\prime k}$

Section Properties

Area = 15 × 21" = 315.00 □"

$\rho_{min} = \frac{200}{F_y} = .0033$

$\therefore A_{s.min} = .0033 \times 315.0 = \underline{1.05\square''}$

Compare with AASHTO minimum:

$I_g = \frac{15 \times 24^3}{12} = 17{,}280 \text{ in}^4$

$S = \frac{17280}{12''} = 1440.0 \text{ in}^3$

$M_{cr} = f_r \times S/12$

$f_r = 7.5\sqrt{f_c'} = 7.5\sqrt{3500}$

$= 443.71$ psi

$\therefore 1.2M_{cr} = 1.2 \times .44371 \times 1440/12$

$= \underline{63.89^{\prime k}}$

Neglect effect of compression steel:

$\rho = \frac{1}{m}\left(1 - \sqrt{1 - 2mR_u/F_y}\right)$

$m = \frac{f_y}{.85f_c'} = 20.17$

$R_u = \frac{M}{\phi b d^2} = \frac{63.89 \times 12}{.9 \times 15 \times 21^2}$

$= .1288$ ksi

$\rho = \frac{1}{20.17} \times \left(1 - \sqrt{1 - 2 \times 20.17 \times .1288/60}\right)$

$= .0022$

AASHTO Min. $A_s = .0022 \times 315.0$

$= \underline{.69\square''}$

$A_s = 5.00\square'' >$ Min. A_s OK

Check Max. A_s :

$$\rho_{bal} = \frac{.85\beta_1 f_c'}{F_y} \times \frac{87}{87+F_y}$$

for $f_c' = 3500$ psi, $\beta_1 = .85$

$$\rho_{bal} = \frac{.85^2 \times 3.5}{60} \times \frac{87}{87+60} = .0249$$

Act. $\rho = (5.0 - 1.76)/_{15\times21} = .0103$

$$\frac{\rho}{\rho_{bal}} = \frac{.0103}{.0249} = .41 < .75 \text{ OK}$$

Calculate Elastic Section Properties

Neutral axis position, Y :

$$15'' \cdot Y \cdot Y/_2 + [2\cdot(n=8.5)-1] \times 1.76_{\square}'' \cdot (Y - 2.75'')$$

$$= (n=8.5) \times 5.0_{\square}'' \times (21'' - Y)$$

$$7.5Y^2 + 28.16Y - 77.44 = 892.50 - 42.5Y$$

$$7.5Y^2 + 70.66Y - 969.94 = 0$$

$$Y = 7.5985''$$

	Area	d	Ad^2
Conc.: 15" × 7.60" =	113.98	3.80"	1645.19
A_s' : (2×8.5−1)×1.76 =	28.16	4.85"	661.98
A_s : (n=8.5)×5.0 =	42.50	13.40"	7633.01
			9940.18
Conc. I_o = 15×7.5985³/12		=	548.40
			10,488.57

$$S_{ST} = \frac{10,488.57/(n=8.5)}{21'' - \bar{Y}} = \underline{92.076 \text{ in}^3}$$

Calculate Ultimate Section Properties

1st iteration - assume compr. f_s = 60 ksi:

$$(.85\times3.5\text{ksi}) \times 15'' \times a$$

$$= (5.0\times60) - (1.76\times(60 - (.85\times3.5)))$$

$$a = 4.47''$$

Then compr. $f_s = 87 \times \frac{4.47 - 2.75\times(\beta_1=.85)}{4.47}$

$$= 41.54 \text{ ksi}$$

(87 = conc. strain, .003, × E_{st} = 29,000)

2nd iteration - use f_s' = 41.54 ksi

$$(.85\times3.5)\times15\times a = (5.0\times60)$$

$$- 1.76\times(41.54 - .85\times3.5)$$

$$a = 5.20''$$

$$M = (\phi=.9)\times\left[(5.20''\times15''\times.85\times3.5)\times(21 - \tfrac{5.20}{2})\right.$$

$$\left. + 1.76\times(41.54 - .85\times3.5)\times(21'' - 2.75'')\right]$$

$$= (\phi=.9)\times(4270.85 + 1238.71) = 4958.61^{\text{in-k}}$$

$$= \underline{\underline{413.22'^{k}}} > 355.84'^{k} - \underline{\text{OK}}$$

Note: LFAN uses this two-iteration process. By way of comparison, an exact solution is given below:

$$(.85\times3.5)\times15\times a + 1.76\left(87\,\frac{a - (\beta_1=.85)\times2.75''}{a}\right.$$

$$\left. - .85\times3.5\right) = 5.0\times60$$

Solving, $a = 5.01''$; $f_s' = 46.41$ ksi

$M = 411.88'^{k} = 99.68\%$ of 413.22

Neglecting displaced concrete: $a = 5.59''$; $f_s' = 50.61$ ksi ; $M = 445'^{k}$

Calculate allowable service moment

Allowable Z = 175

$A_e = 2 \times (24"-21") \times 15"/5 \text{ bars} = 18.0 \square"/\text{bar}$

$d_c = 24"-21" = 3"$

$\text{Allow. } f_s = \dfrac{Z = 175}{\sqrt[3]{3" \times 18.00"}} = 46.30 \text{ ksi} > 36.0$

⇒ use allow. $f_s = 36.0$ ksi

Elastic $S_{st} = 92.671 \text{ in}^3$; act. M = 233.60'k

$\text{Act. } f_{st} = \dfrac{233.60 \times 12}{92.076} = 30.44 \text{ ksi} < 36.0$ OK

$\text{Acting } Z = \dfrac{30.44}{36.0} \times 175 = 148.0$

Allow. Serv. M:

$36.0 \text{ ksi} \times \dfrac{92.076}{12} = 276.23'k$

∴ Design is OK

Set flag 00 for Short Form run

```
             SF 00
        XEQ "LFAN"
F-Y 40,60
          60.0000    RUN
Fc'KSI
           3.5000    RUN
FL W"
          15.0000    RUN
WEB W"
                     RUN
H"
          24.0000    RUN
---------
Z 145,175
         175.0000    RUN
-----------------------
A-S1
           5.0000    RUN
D"
          21.0000    RUN
A-S2
                     RUN
MAX BAR #
           9.0000    RUN
A-S'
           1.7600    RUN
D-TOP"
           2.7500    RUN

LF M=413.2223 'K
SERV M=276.2266 'K
-----------------------
A-S1
```

```
                 XEQ "LFAN"
 9:39:09 AM  02/15/89

TITLE:
DOUBLE-REINF. BEAM   RUN
************************
ACI
F-Y 40,60
          60.0000    RUN
Fc'KSI
           3.5000    RUN
FL W"
          15.0000    RUN
WEB W"
                     RUN
H"
          24.0000    RUN
I-G=17,280.0000
1.2Mc=63.8892 'K
200/FY=0.0033
 A-S=1.0500 SQ"
.75Pb=0.0187
---------
Z 145,175
         175.0000    RUN
LF M,'K
         355.8400    RUN
---------
SERV M,'K
         233.6000    RUN

LF: EST A-S=4.3795 SQ"
WS: EST A-S=6.2846 SQ"
-----------------------
A-S1
           5.0000    RUN
D"
          21.0000    RUN
A-S2
                     RUN
MAX BAR #
           9.0000    RUN
A-S'
           1.7600    RUN
D-TOP"
           2.7500    RUN
RHO=0.0103
P/Pb=0.4124 OK
Ie=10,568.9777
C=6.1195"

LF M=413.2223 'K
SERV M=276.2266 'K

Z STRS=36.0000 KSI
ACT STRS=30.4446 KSI
Z=147.9944

 OK
-----------------------
A-S1
```

Calculate Working Stress Capacity

$S_{st} = 92.076\ in^3$

$M_{st} = 92.076 \times 24\ ksi/12 = \underline{184.15'^{k}}$

$S_c = \frac{10{,}488.57}{7.5985''} = 1380.35\ in^3$

$M_{conc} = 1380.35 \times (.45 \cdot 3.5\ ksi)/12$

$= \underline{181.17'^{k}}$

$S'_{st} = \frac{10{,}488.57/8.5}{7.5985 - 2.75''} = 254.50\ in^3$

$M'_{ST} = 254.50 \times 24\ ksi/12 = \underline{509.00'^{k}}$

Concrete governs: $M = 181.17'^{k}$

∴ Section is <u>No Good</u> by W.S.D.

```
                 XEQ "Ic"
11:28:41 AM  02/15/89

TITLE:
DOUBLY-REINF. BEAM    RUN
************************
ACI
F-S 20,24
          24.0000     RUN
Fc'KSI
           3.5000     RUN
N
           8.5000     RUN
FL W"
          15.0000     RUN
WEB W"
                      RUN
H"
          24.0000     RUN
I-G=17,280.0000
1.2Mc=63.8892 'K
200/FY=0.0033
 A-S=1.0500 SQ"
.75Pb=0.0187
---------
SERV M,'K
         233.6000     RUN

WS: EST A-S=6.2846 SQ"
-----------------------
A-S1
           5.0000     RUN
D"
          21.0000     RUN
A-S2
                      RUN
A-S'
           1.7600     RUN
D-TOP"
           2.7500     RUN
RHO=0.0103
P/Pb=0.4124 OK

Ic=10,488.5732
Y=7.5985"
Sc=1,380.3469
S-ST=92.0755

CONC M=181.1705 'K
Ie=10,568.9777

 NG
-----------------------
A-S1
```

Program LFAN/Ic For Reinf. Concrete Beam Analysis 1148 bytes

Step	X	Y	Z	T	L
01♦LBL "LFAN"					
02 SF 06					
03 CF 09					
04 GTO 00					
05♦LBL "Ic"					
06 CF 06					
07 SF 09					
08♦LBL 00					
09 28					
10 XROM "S"					
11 CF 10					
12 XEQ "N"	f_c'	β_1			
13 X<>Y					
14 STO 18	β_1				
15 12	12	β_1	f_c'		
16 RCL Z	f_c'				
17 -					
18 FC? 06					
19 PROMPT					
20 STO 09	N				
21 RCL 01	F_y				
22 41					
23 X>Y?					
24 SF 09					
25 RCL 02	$.75\rho_{bal}$				
26 STO 24					
27♦LBL D					
28 CF 08					
29 "FL W="					
30 PROMPT					
31 STO 07					
32 "WEB W="					
33 PROMPT					
34 STO 03					
35 RCL 07	Fl.W	Web W			
36 X=Y?					
37 SF 08					
38 CLX					
39 "FL TH="					
40 FC? 08					
41 PROMPT					
42 STO 10	Fl.Th				
43 "H="					
44 PROMPT	H				
45 STO 11					
46 3					
47 -					
48 STO 14	Est d				
49 X↑2	d^2				
50 RCL 07	Fl.W				
51 *					
52 .9	ϕ				
53 *					
54 STO 05	ϕbd^2				

Step	X	Y	Z	T	L
55 CLA					
56 RCL 11	H				
57 RCL 10	Fl.Th				
58 -	Web d				
59 RCL 03	Web W				
60 RCL 07	Fl.W				
61 LASTX	Fl.Th	Fl.W	Web d	Web W	
62 XROM "AY"	A	$\bar{Y}$	S_b	I	
63 R↑					
64 STO 23	I				
65 "I-G="					
66 ARCL X					
67 FC? 00					
68 AVIEW					
69 R↑	S_b				
70 6					
71 *	$6S_b$				
72 XROM "Mc"	ρ				
73 X<>Y	M_{cr}				
74 1.2					
75 /					
76 STO 22	M_{cr}				
77 X<>Y	ρ				
78 RCL 07	Fl.W.				
79 *					
80 RCL 14	d				
81 XROM "R"					
82 STO 19	Min A_s				
83 RCL 24	$.75\rho_{bal}$				
84 RCL 14	est. d				
85 RCL 18	β_1				
86 RCL 00	$.85f_c'/f_y$	β_1	d	$.75\rho_{bal}$	
87 *					
88 /					
89 *	C" Arr				
90 STO 15	$.75\rho_{bal}$				
91 FS? 08					
92 GTO 10	C	$.75\rho_b$	$.75\rho_b$	$.75\rho_b$	
93 RCL 10	Fl.Th	C			
94 X>Y?					
95 GTO 10					
96 RCL 00	$.85f_c'/f_y$	Fl.Th.			
97 *					
98 RCL 07	Fl.W				
99 RCL 03	Web W	Fl.W		C	
100 -	Width				
101 *	Equiv A_s				
102 RCL 14	est. d				
103 /	Fl.A/d				
104 .75					
105 *	$.75A_s$ fl/d				
106 RCL 24	$.75\rho_{bal}$ web				
107 RCL 03	Web W				
108 *	$\rho_w \cdot b_w$	$\rho_f \cdot b_f$			$A_{s.fl} \cdot d = \rho \times b$
109 +	$\rho \cdot b$				
110 RCL 07	Fl. width				
111 /	$\rho_{equiv.}$				
112 RDN				ρ	

D65

Program LFAN/IC For ________ ______ bytes

Step	X
113◆LBL 10	
114 R↑	.75ρbal
115 STO 02	
116 ARCL X	
117 FC? 00	
118 AVIEW	
119◆LBL E	
120 CLX	
121 STO 25	Serv. M
122 STO 26	LF M
123 FS? 09	
124 GTO 13	
125 XROM "L"	
126 130	
127 "Z 130,170"	
128 FS? 04	
129 "Z 145,175"	
130 PROMPT	
131 STO 13	Z
132 FS? 00	
133 GTO 23	
134 "LF"	
135 XEQ 10	
136 STO 26	L.F. M
137 XROM "U"	
138 STO 14	Est. ρ
139◆LBL 13	
140 FS? 00	
141 GTO 23	
142 XROM "L"	
143 "SERV"	
144 XEQ 10	
145 STO 25	Serv M
146 "MIN FATG"	
147 FC? 04	
148 XEQ 10	Fatg M
149 STO 28	
150 ADV	
151 RCL 14	Est. ρ
152 RCL 07	b_{e1}
153 *	
154 RCL 11	H
155 3	3"
156 -	Est d
157 *	Est A_s
158 "LF"	
159 FS? 06	
160 XEQ 03	
161 RCL 25	Serv M
162 LASTX	d
163 /	M/d
164 RCL 01	f_y
165 5	
166 /	
167 9.24	
168 +	*

Step	X
169 /	Est. A_s
170 "WS"	
171 XEQ 03	
172◆LBL 23	
173◆LBL A	
174 E	1
175 STO 20	counter
176 CLX	
177 STO 14	
178 STO 21	
179 STO 17	
180 X<> 04	Max d
181 XROM "LL"	
182 CF 22	
183◆LBL 01	
184 "A-S"	
185 RCL 20	counter
186 XROM "AX"	
187 CLX	
188 PROMPT	A_s
189 FC? 22	} shear
190 GTO 04	
191 X=0?	} end of
192 GTO 02	} A_s input
193 ST+ 14	
194 STO 08	A_s
195 CF 07	
196 ISG 20	
197 ""	
198 "D""	
199 PROMPT	D
200◆LBL 12	
201 *	$A_s d$
202 ST+ 21	$\Sigma A_s d$
203 LASTX	d
204 *	$A_s d^2$
205 ST+ 17	$\Sigma A_s d^2$
206 SF 07	
207 RCL 04	Max d
208 LASTX	d
209 X>Y?	
210 STO 04	Max d
211 GTO 01	
212◆LBL 04	
213 RDN	d_{max}
214 STO 04	
215 RCL 03	b_w
216 RCL 07	b_e
217 X>Y?	
218 X<>Y	b_{min}
219 XEQ "VS"	
220 GTO E	

$A_s = \frac{M^{\prime\prime k}}{1.76 d^{\prime\prime}}$ or $= \frac{M^{\prime\prime k}}{1.43 d^{\prime\prime}}$

* 1.76 × 12 for f_y=60
1.43 × 12 for f_y=40

D66

Step	X	Y	Z	T	L
221♦LBL C					
222 DSE 20					
223 ""					
224 RCL 08	A_s				
225 ST- 14					
226 CHS	$-A_s$				
227 FC? 07					
228 .	0 or $-A_s$				
229 LASTX	d	$-A_s$			
230 GTO 12					
231♦LBL 02					
232 "MAX BAR #"					
233 FC? 09					
234 PROMPT	Max #				
235 STO 27					
236 CF 07					
237 CLX					
238 "A-S'"					
239 PROMPT					
240 STO 20	A_s'				
241 ENTER↑					
242 "D-TOP"					
243 X>0?					
244 PROMPT	d'				
245 STO 08					
246 RCL 15	"c"				c for .75ρ_{bal}
247 -	$d-c$				
248 LASTX	c				
249 /					
250 87					
251 *					
252 RCL 01	f_y				
253 /	$\frac{87(d'-c)}{c\,f_y}$				
254 E					
255 CHS	-1				
256 X>Y?					
257 X<>Y					
258 RDN	max	A_s'			
259 *	eff. A_s'				
260 RCL 14	ΣA_{st}				
261 +					
262 LASTX					
263 *	$\frac{A_s-A_s'}{A_s}$				
264 RCL 21	$\Sigma A_s d$				
265 /	$(A_s-A_s')/Ad$				
266 RCL 07	b_{el}				
267 /	eff. ρ				
268 XEQ "RH"					
269 RCL 19	Min A_s				
270 RCL 14	ΣA_s				
271 "AS<MIN"					
272 X<=Y?					
273 AVIEW					
274 X<=Y?					
275 XROM "L"					
276 RCL 09	N				
277 *	nA_s				

Step	X	Y	Z	T	L
278 LASTX	N				
279 ST* 17	$(\Sigma A_s d^2)n$				
280 ST+ X	$2n$				
281 E	1				
282 -	$2n-1$				
283 RCL 20	A_s'				
284 *					
285 STO]	$(2n-1)A_s'$	nA_s			
286 +	ΣnA_s				
287 STO [					
288 RCL 21	$\Sigma A_s d$				
289 RCL 09	n				
~~290 *~~					
~~291 RCL 08~~					
290 *	$nA_s d$				
291 RCL 08	d'				
292 RCL]	$(2n-1)A_s'$				
293 *					
294 +	$\Sigma nA_s d$				
295 STO \					
296 FS? 08					
297 GTO 04					
298 RCL 07	b_e				
299 RCL 10	Fl.Th				
300 X↑2	th^2				
301 *	Ad				
302 2					
303 /	Ad/2				
304 RCL 10	Fl.Th.				
305 RCL 08	D'				
306 -					
307 RCL]	$(2n-1)A_s'$				
308 *	$A_s'd$				
309 +	ΣAd				
310 RCL 21	$\Sigma A_s d$				
311 RCL 14	ΣA_s				
312 RCL 10	Fl.Th				
313 *					
314 -	$A_s(Fl-d)$				
315 RCL 09	n				
316 *					
317 X<=Y?					x<y? → N.A. in flange
318 GTO 04					
319 CF 05					
320 RCL 07	Fl.W				
321 RCL 03	Web W				
322 -	Fl. W				
323 RCL 10	Fl.Th				
324 *	A_f				
325 ST+ [	(ΣA)				
326 LASTX	Th				
327 *	Ad				
~~328 2~~					
329 /	Ad/2				
330 ST+ \	(ΣAd)				
331 RCL 03	Web W				
332 GTO 05					

Step	X	Y	Z	T	L
333◆LBL 04					
334 RCL 07	fl.w.				
335 SF 05					
336◆LBL 05	b				
337 2					
338 /	b/2				
339 RCL [	ΣA				
340 RCL \	ΣAd				
341 CHS	-ΣAd				
342 X<> Z	b/2	ΣA	ΣAd		
343 XROM "Q"					
344 STO 12	Ȳ				Elastic N.A. position
345 FS? 08					
346 GTO 08					
347 RCL 10	Fl.Th.				
348 2					
349 /	Th/2	Ȳ			
350 X>Y?					
351 GTO 08					
352 -	Ȳ-Th/2				
353 X↑2	d^2				
354 RCL 10	Th				
355 X↑2	Th^2				
356 12					
357 /	$Th^2/12$				
358 +					
359 RCL 10	Fl.th.				
360 *					
361 RCL 07	Fl.w				
362 *	I_o+Ad^2				
363 RCL 03	WebW				
364 RCL 12	Ȳ				
365 RCL 10	Fl.th.				
366 -	Ȳ-Th.	WebW	I_o+Ad^2		
367 GTO 09					
368◆LBL 08					
369 .	0				
370 RCL 07	Fl.W				
371 RCL 12	Ȳ				
372◆LBL 09	Ȳ	b	I_o+Ad^2		
373 3					
374 Y↑X	y^3				
375 LASTX					
376 /	$y^3/3$	b	I_o+Ad^2		
377 *	I_w				
378 +	$I_{conc.}$				
379 RCL 08	d'				
380 RCL 12	Ȳ				
381 -					
382 X↑2	d^2				
383 RCL]	$(2n-1)A_s'$				
384 *	$A_s'd^2$				
385 +	I				
386 RCL 14	ΣA_s				

Step	X	Y	Z	T	L
387 RCL 09	n				
388 *					
389 RCL 12	Ȳ				
390 X↑2	y^2	nA_s			
391 *	Ay^2				
392 +	I				
393 RCL 21	$\Sigma A_s d$				
394 RCL 09	n				
395 *	$nA_s d$				
396 RCL 12	Ȳ				
397 *					
398 ST+ X	$2nA_s d\bar{Y}$				
399 -	I_{conc}				
400 ST+ 17					
401 RCL 17	I_{cr}				
402 RCL 09	n				
403 /					
404 RCL 04	Max d				
405 RCL 12	Ȳ				
406 -	d	I/n			
407 /	S_{st}				
408 STO 16					
409 FS? 06					
410 GTO 06					
411 .4					
412 FS? 04					
413 .45					
414 RCL 00	$.45f_c'/f_c$				
415 *					
416 .85	.45				
417 /					
418 RCL 01	f_y				
419 *	$.45f_c'$				
420 RCL 17	I_{cr}				
421 RCL 12	Ȳ				
422 /	S_c	$.45f_c'$			
423 *	M_c				
424 FS? 00					
425 GTO 11					
426 ADV					
427 "Ic="					
428 ARCL 17					
429 AVIEW					
430 "Y="					
431 ARCL 12					
432 "⊢" "					
433 AVIEW					
434 "Sc="					
435 ARCL L					
436 AVIEW					
437 "S-ST="					
438 ARCL 16					
439 AVIEW					
440◆LBL 11	M_c				
441 ADV					
442 RCL 13	f_{st}				
443 RCL 16	S_{st}	f_s	M_c		

068

Step	X	Y	Z	T	L
444 *	M_{sr}	M_c			
445 "CONC"					
446 X<=Y?					
447 "ST"					
448 X>Y?					
449 X<>Y	M_{min}				
450 RCL 25	ServM				
451 XEQ 04	M_{min}				
452♦LBL 06					
453 FS? 00					
454 GTO 06					
455 RCL 23	I_{gross}				
456 RCL 22	M_{cr}				
457 RCL 25	ServM				
458 X=0?					
459 GTO 06	(M_{cr}/M)	I_{gr}			
460 /					
461 3					
462 Y↑X	$(\ )^3$				
463 *	$I_{gr}(\ )^3$				
464 E	1				
465 LASTX	$(\ )^3$				
466 -	$1-(\ )^3$				
467 RCL 17	I_{cr}				
468 *	$I_{cr}(1-(\ )^3)$	$I_g(\ )^3$			
469 X<0?					
470 CLX					
471 +	I_e				
472 RCL 23	I_g				
473 X>Y?					
474 X<>Y	I_e				
475 "Ie="					
476 ARCL X					
477 AVIEW					
478♦LBL 06					
479 CLA					
480 FC? 06					
481 GTO 25					
482 RCL 10	Fl.Th				
483 RCL 14	ΣA_s	•			
484 RCL 20	A_s'		•		
485 -	A_s-A_s'	•			
486 RCL 00	$.85f_c'/f_y$		•		
487 /	A_sf_y/f_c'	•			
488 RCL 20	A_s'		•		
489 +		•			
490 RCL 07	b_e		•		
491 STO \					
492 /	a	Fl.Th			
493 ENTER↑					
494 CHS	-a	a	Fl.Th		
495 RCL 18	β_1			•	
496 /	c		•		
497 RCL 08	d'			•	
498 X<>Y	$-a/\beta_1$	d'	a		
499 +	d'-c		•		

Step	X	Y	Z	T	L
500 LASTX	-c			•	
501 /	$\frac{c-d'}{c}$	a	•		
502 87				•	
503 *	f_s'		•		
504 RCL 01	f_y	f_s'	a		
505 X<=Y?					
506 X<>Y	Max	Min	a	Fl.Th	
507 RDN	Min	a			
508 STO a					
509 RDN	a	Fl.Th			
510 X<=Y?					
511 CLX					
512 X<=Y?					
513 GTO 07					
514 RCL 10 (Ultimate N.A. in Web: 514–522)	Fl Th				
515 RCL 03	Web W				
516 STO \					
517 RCL 07	Fl. W				
518 -	-Fl. W				
519 *	-Fl. Area				
520 RCL 00	$.85f_c'/f_y$				
521 *					
522 STO [	$Fl.A.f_c'/f_y$				
523♦LBL 07					
524 RCL 14	ΣA_s				
525 +	ΣA				
526 RCL 00	$.85f_c'/f_y$				
527 /	Af_y/f_c				
528 RCL 20	A_s'				
529 RCL 00					
530 RCL 01	f_y				
531 *	$.85f_c'$				
532 ST- a	$(f_s'-.85f_c')$				
533 /	A_s'/f_c'				
534 RCL a	f_s'				
535 *	$A_s'f_s'/f_c'$	$\Sigma Af_y/f_c'$			
536 -	Af_y/f_c'				
537 RCL \	b				
538 /	"a"				
539 RCL 21	ΣA_sd				
540 RCL 14	ΣA_s				
541 /	d				
542 STO]					
543 RCL 10	Fl.Th				
544 2	2	Fl.Th	d	a	
545 /					
546 -	$d-Fl/2$	a	a	a	
547 RCL [	$Fl.A.f_c/f_y$				
548 *	Adf_c/f_y	a	a	a	
549 CHS					
550 R↑	a				
551 2					
552 /	a/2				
553 RCL]	d	a/2	$-Adf_c/f_y$	a	
554 -	a/2-d				
555 R↑	a				
556 *	a·d				

Step	X	Y	Z	T	L
557 RCL \	b				
558 *	A·d				
559 RCL 00	.85f'c				
560 *					
561 -	-ΣAd f'c				
562 RCL 01	fy				
563 *	-ΣAd fy				
564 RCL]	d				
565 RCL 08	d'				
566 -					
567 RCL 20	A's				
568 *	A's(d-d')	ΣAd			
569 RCL a	f's				
570 *					
571 +					
572 .9	φ				
573 *	M				
574 X<>Y	a				
575 RCL 18	β1				
576 /	C	M			
577 "C="					
578 ARCL X					
579 "├""					
580 FC? 00					
581 AVIEW					
582 ADV					
583 X<>Y	M				
584 "LF"					
585 RCL 26	LF M				
586 XEQ 04					
587 FS? 09					
588 GTO 25					
589 RCL 13	Z				
590 RCL d	d	Z			
591 FIX 2					
592 RCL 27	Max #				
593 X↑2	#²				
594 81	81	#²	d	Z	
595 X≠Y?					
596 X>Y?					
597 GTO 07					
598 CLX					
599 1.1					
600 Y↑X					
601 125					
602♦LBL 07					
603 /	Area				Area of one bar
604 RND					
605 X<>Y	d	Area	Z		
606 STO d					
607 CLX					
608 RCL 21	ΣAs d				
609 RCL 14	ΣAs	As d	Area	Z	
610 /	d	Area	Z		
611 RCL 11	H				
612 -	d-H				
613 ST+ X	d	Area			

Step	X	Y	Z	T	L
614 *	d Bar	Z			
615 RCL 04	Max d				
616 RCL 11	H				
617 -	dc				
618 *					
619 RCL 03	bw				
620 *	dc A·Zs	Z			
621 RCL 14	ΣAs				
622 /	dc A				
623 3					
624 1/X					
625 Y↑X	∛(dc A)	Z			
626 /	All. Strs				
627 RCL 01	fy				
628 .6	.6 fy				
629 *					
630 X>Y?					
631 X<>Y	All. Strs				
632 ENTER↑	All.	All.			
633 "SERV"					
634 RCL 16	Sst				
635 *	All M				
636 RCL 25	Serv M				
637 XEQ 04					
638 FS? 00					
639 GTO 23					
640 ADV					
641 CLX					
642 RCL 13	Z	All Strs			
643 X<>Y	All Strs	Z			
644 "Z"					
645 XEQ 11					
646 RCL 25	Serv M				
647 RCL 16	Sst				
648 /	Act strs	All Strs	Z		
649 "ACT"					
650 XEQ 11					
651 /	All/Act	Z			
652 /	Act Z				
653 "Z="					
654 ARCL X					
655 AVIEW					
656♦LBL 25					
657 FS? 00					
658 GTO 23					
659 23.4	23.4				
660 RCL 28	Mmin				
661 FC? 04					
662 X=0?					
663 GTO 07					
664 RCL 16	Sst				
665 /	fmin	23.4			
666 X>0?					
667 GTO 06					

D70

Program LFAN/Ic For ______ ______ bytes Page 7 of 7

Stress Reversal (steps 668–678)

Step	X	Y	Z	T	L
668 RCL 11	H				
669 RCL 04	Max d				
670 -	H-d				
671 LASTX	d				
672 /	(H-d)/d	f_{min}	23.4		
673 .3					
674 -					
675 *					
676 1.4					
677 /					
678 CHS	f_{min}'				
679♦LBL 06					
680 STO 04	f_{min}				
681 3					
682 /	f/3	23.4			
683 -	All.f				
684 ADV					
685 "ALL"					
686 XEQ 00					
687 RCL 25	Serv M				
688 RCL 16	S_{ST}				
689 /	Act.f				
690 RCL 04	f_{min}				
691 -					
692 "ACT"	Act.f	All.f			
693 XEQ 00					
694 X>Y?					
695 SF 07					
696♦LBL 07					
697 ADV					
698 CLA					
699 XROM "OK"					
700 AVIEW					
701 GTO 23					
702♦LBL 00					
703 "⊢ FAT"					
704♦LBL 11					
705 XROM "K"					
706 XEQ "PV"					
707 RTN					

Step	X	Y	Z	T	L
708♦LBL 04	Minpod	Mcalc			
709 X>Y?					
710 SF 07					
711 RDN	Mcalc				
712 XROM "M="					
713 XEQ "PV"					
714 RTN					
715♦LBL 10					
716 "⊢ "					
717 CLX					
718 XROM "M"					
719 PROMPT	M'-k	12			
720 *	M''-k				
721 RTN					
722♦LBL 03					
723 "⊢: EST "					
724 XROM "AS"					
725 X≠0?					
726 XEQ "PV"					
727 END					

D71

CONCRETE SHEAR REINFORCING DESIGN

xeq VST (SIZE 015)

SOFTKEYS (USER mode ON)

A	New area of shear reinforcing
D	New dimensions

USER FLAGS

Flag 00	clear	Full output
	set	Title block not printed (output is not affected by flag 00)
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 04	clear	AASHTO
	set	ACI

INPUT SUMMARY

Prompt	Input	Default
WSD?	Asking if by working stress design. Answer 1 (yes) for WSD, 0 (no) for LFD.	1 (WSD)
F-S 20,24 or F-Y 40,60	Reinforcing working or yield stress for WSD or LFD, respectively, in ksi.	24; 60
Fc'KSI	Concrete ultimate strength (for either WSD or LFD)	4.0 ksi
B"	Beam width (or web thickness), in inches	Previous B
D"	Beam depth to outermost layer of tension reinforcing, in inches	Previous D
A-V	Area of shear reinforcing, sq. inches; e.g. for one #5 stirrup input 0.62. Bypass (or input zero) to get new dimension prompts in VST or return to flexural program.	Zero
V, K	Design shear, in kips. For LFD, this must have the load factor included; however, <u>the program applies the undercapacity factor phi = 0.85.</u>	Zero

OUTPUT SUMMARY	(Items in parenthesis are displayed only if flag 00 is clear)
Vc	The portion of the total shear taken by the concrete, in kips.
SPA	The required stirrup spacing, in inches. Appends (MAX) if the spacing is governed by maximum spacing rules rather than shear requirements.
.5Vc>V	States that the design shear is less than half the allowable concrete shear; no shear reinforcing is necessary.
V-Vc NG	Design shear is too large for the given section. Increase section (or concrete strength).

PROGRAM FLAGS

Flag 06	clear	WSD
	set	LFD
Flag 10	set	Quick return from subroutine N

STORAGE REGISTER USE

Register	Value
00	Fc', ksi
01	Fy, ksi
02	Vc, kips (concrete shear)
03	B"
04	D"
05-07	Not used
08	$\sqrt{Fc'}$, ksi
09-12	Not used
13	F_{st}, ksi (WSD only)
14	B"

D74

CONCRETE SHEAR REINFORCING DESIGN

Program VST designs the shear reinforcing for rectangular concrete beams. It can also be used for T-beams by simply inputting the web thickness at the B" (width, inches) prompt. VST will design by either working stress (WSD) or load factor (LFD) methods. It uses the ACI code if flag 04 is set; AASHTO is followed if flag 04 is clear. In LFD there is no difference between the two; in WSD, ACI allows a somewhat higher concrete shear than AASHTO. No matter which code is followed, VST uses the simplest shear design formula,

$$SPA = (F_{st} \text{ or } F_y) \times D'' \times A_V/(V - V_C).$$

It does not use the more complex moment-interaction formula.

After the dimensions and material properties are input, VST calculates and displays the concrete shear contribution, Vc, in kips. It then prompts for the area of shear reinforcing to be used,and then repeatedly calculates the stirrup spacing required for shear loads input. This allows the designer to quickly find the necessary spacings for varying shears along a beam. To input a new stirrup area (A-V), bypass the V, K (shear, in kips) prompt or press softkey A in USER mode. To input new dimensions, bypass the A-V prompt or press softkey D in USER mode.

MAXIMUM SPACING

The program follows all code provisions for maximum stirrup spacing (d/2 or 24") and maximum and minimum shear loads. In the case of small heavily loaded beams with large stirrup areas, the maximum spacing is cut in half (AASHTO 8.15.5.3.8/8.16.6.3.8 and ACI 11.5.4.3) to d/4 or 12". The recommended procedure if this happens is to decrease the stirrup size and run it again. The minimum area of shear reinforcing (per AASHTO 8.19.1.2 or ACI 11.5.5.3) is also checked. This sometimes governs the design of large, lightly loaded beams. See the examples for clarification.

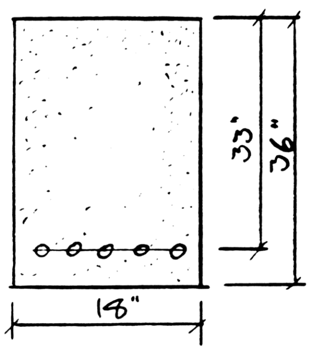
FOOTINGS and SLABS

Footing and slab designs are treated differently; they require no shear reinforcing for $V < V_C$. For these, any minimum shear reinforcing output (i.e. any SPA with (MAX) appended to it) is incorrect.

FLEXURAL SUBROUTINE

The main shear design program VST contains a subroutine, VS, which is called by the four concrete flexural programs (WSBM, LFBM, Ic, and LFAN). This subroutine is identical to VST except it uses the material properties and dimensions of the beam being investigated by the calling program. In the case of the analysis programs LFAN and Ic, which accept T-beams, the width used is the smaller of the web thickness or the flange width (the latter would be chosen for invert T-beams). The depth used is to the deepest layer of reinforcing (if there is more than one layer of tension reinforcing). To return to the calling program, bypass the V, K prompt and then the A-V prompt. The analysis programs return to the moment input sequence, while the design programs prompt for a new section.

Design shear reinforcing for this beam:



f'_c = 4000 psi
f_y = 60 ksi

$V_c = 2\sqrt{4000} \times 18" \times 33" =$ <u>75.14 k</u>

Use #4 stirrups - $A_v = 2 \times .20 =$ <u>.40 □"</u>

<u>Max. Spacing</u>

$A_{v\text{-}min} = \dfrac{50\ b \cdot Spa}{f_y}$ (AASHTO 8.19.1.2)

$\therefore$ Max spa. = $\dfrac{.40 \times 60{,}000}{50 \times 18"}$ = 26.67"

or $\frac{d}{2} = \frac{33"}{2}$ = 16.5" ← Governs
or Max = 24"

@ <u>Reaction</u>

DL = 24.0
LL = 5/3 × 36.0 = <u>60.0</u>
84.0 × 1.3 = 109.2 k

V_s = 109.2/(ϕ=.85) − 75.14 = 53.33 k

Spa. = $\dfrac{A_v f_y d}{V_s} = \dfrac{.40 \times 60 \times 33"}{53.33}$ = 14.85"

@ <u>1/4 Point</u>

DL = 12.0
LL = 5/3 × 20.25 = <u>33.75</u>
45.75 × 1.3 = 59.48

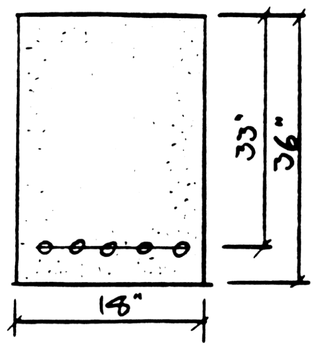
$V_s = \dfrac{59.48}{(\phi = .85)} - 75.14\ k < 0$

⇒ <u>Use Max. Spa. = 16.5"</u>

```
                 XEQ "VST"
 3:52:40 PM  12/12/88

TITLE:
RECT. BEAM            RUN
*************************
WSD?
            0.0000    RUN     0→"no"→LFD
AASHTO
F-Y 40,60
           60.0000    RUN
Fc'KSI
            4.0000    RUN
B"
           18.0000    RUN
D"
           33.0000    RUN
Vc=75.1357 K
*************************
A-V
             .4000    RUN
---------
V, K
          109.2000    RUN
SPA=14.8496"
---------
V, K
           59.4800    RUN
SPA=16.5000"(MAX)
---------
V, K
           30.0000    RUN
.5Vc>V
---------
V, K
```

ACI L.F.D. Shear Design

Design shear reinforcing for this beam:



$f_c' = 4000$ psi
$f_y = 60$ ksi

$V_c = 2\sqrt{4000} \times 18" \times 33" = \underline{75.14\ k}$

Use #4 stirrups - $A_v = 2 \times .20 = \underline{.40 \square"}$

Max. Spacing

$A_{v\text{-}min} = \dfrac{50\ b \cdot Spa}{f_y}$ (ACI 11.5.5.3)

$\therefore$ Max spa. $= \dfrac{.40 \times 60{,}000}{50 \times 18"}$

$= 26.67"$

or $\dfrac{d}{2} = \dfrac{33"}{2} = 16.5"$ ← Governs

or Max $= 24"$

@ Reaction

DL = 1.4 × 24.0 = 33.6
LL = 1.7 × 36.0 = 61.2
94.8 k

$V_s = 94.8/(\phi = .85) - 75.14 = 36.39$ k

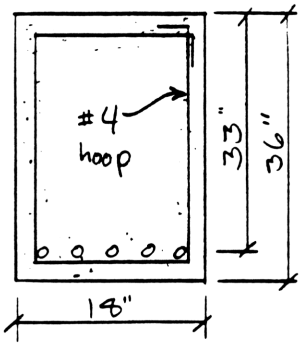
$Spa. = \dfrac{A_v f_y d}{V_s} = \dfrac{.40 \times 60 \times 33"}{36.39} = 21.76"$

Use 16" Spacing Entire Beam

```
                     SF 04        ACI

                  XEQ "VST"
 3:54:16 PM  12/12/88

TITLE:
RECT. BEAM 2           RUN
************************
WSD?
             0.0000    RUN        0 → "no" → LFD
ACI
F-Y 40,60
            60.0000    RUN
Fc'KSI
             4.0000    RUN
B"
            18.0000    RUN
D"
            33.0000    RUN
Vc=75.1357 K
************************
A-V
              .4000    RUN
---------
V, K
            94.8000    RUN
SPA=16.5000"(MAX)
---------
V, K
                       RUN        Bypass
************************          prompt to
A-V                               Try #3 hoop
              .2200    RUN
---------
V, K
            94.8000    RUN
SPA=11.9691"
---------
V, K
```

From the preceding:

$f'_c = 4000$ psi
$f_y = 60$ ksi
$f_{st} = 24$ ksi



@ Reaction:

DL = 24.0 k (service)
LL = 36.0 k
Σ = 60.0 k

@ ¼ Point:

DL = 12.0 k
LL = 20.25 k
Σ = 32.25 k

$V_c = .95\sqrt{4000} \times 18" \times 33" = \underline{35.69\ k}$

@ Reaction,

$V_s = 60.0 - 35.69 = 24.31$ k

Use stirrups - #4 hoop

$A_v = 2 \times .20 = .40\ \square"$

$Spa. = \dfrac{A_v f_s d}{V_s} = \dfrac{.40\square" \times 24\ ksi \times 33"}{24.31\ k} = 13.03"$

@ ¼ Point

$V_s = 32.25 - 35.69$

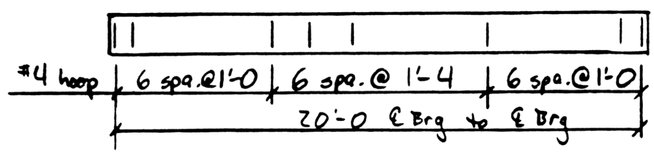
$V_s < 0 \Rightarrow$ Min. Steel

$\frac{1}{2} V_c = 17.84$ k

$32.25 > 17.84 \Rightarrow$ Need Stirrups

Max. Spa.:
$d/2 = 33/2 = 16.5"$ ← Governs
max = 24"

Use this layout:



Check Maximums

$2\sqrt{f_c}\, b\, d = 2\sqrt{4000} \times 18 \times 33$

$= 75.14$ k

$V_s < 2\sqrt{f'_c} \Rightarrow$ Spa. OK - AASHTO 8.15.5.3.8

$V_s < 4\sqrt{f'_c} \Rightarrow$ OK - AASHTO 8.15.5.3.9

$A_{v\text{-}min} = \dfrac{50\, b_w \cdot Spa}{f_y}$ AASHTO 8.19.1.2

$\therefore$ Max Spa. $= \dfrac{.40 \cdot 60{,}000}{50 \cdot 18"}$

$= 26.67"$ OK

CF 04 AASHTO

```
            XEQ "VST"
 3:56:10 PM  12/12/88

TITLE:
RECT. BEAM 3          RUN
*************************
WSD?
           1.0000     RUN
AASHTO
F-S 20,24
          24.0000     RUN
Fc'KSI
           4.0000     RUN
B"
          18.0000     RUN
D"
          33.0000     RUN
Vc=35.6895 K
*************************
A-V
            .4000     RUN
---------
V, K
          24.3100     RUN
SPA=16.5000"(MAX)
---------
V, K
          60.0000     RUN
SPA=13.0314"
---------
V, K
          32.2500     RUN
SPA=16.5000"(MAX)
---------
V, K
```

W.S.D. → 1 = "yes"

$.95\sqrt{f_c'} \cdot b \cdot d$

Oops - wrong number

Program VST For Shear Reinforcing Design 263 bytes Page 1 of 2

Step	X	Y	Z	T	L
01♦LBL "VST"					
02 14					
03 XROM "W"					
04 SF 10					
05 XEQ "N"					
06 STO 00	f'c				
07♦LBL D					
08 XROM "BD"	D				
09 RCL 03	B				
10 STO 14					
11 RCL 00	f'c				
12 XEQ 02					
13 GTO D					
14♦LBL "VS"	B				
15 STO 14					
16 XROM "L*"					
17 "SHEAR"					
18 AVIEW					
19 RCL 00	.85f'c/fy				
20 RCL 01	fy				
21 *					
22 .85					
23 /	f'c				
24♦LBL 02					
25 E3	1000				
26 /					
27 SQRT	√f'c ksi				
28 STO 08					
29 .95					
30 FS? 04					
31 1.1					
32 FS? 06					
33 2	factor				
34 RCL 08					
35 *	Vc				
36 RCL 14	B				
37 *					
38 RCL 04	D				
39 *					
40 STO 02	Vc, k				
41 "Vc="					
42 ARCL X					
43 "⊢ K"					
44 FC? 00					
45 AVIEW					
46♦LBL A					
47 XEQ "L*"					
48 CLX					
49 "A-V"					
50 PROMPT					
51 X=0?					
52 RTN					
53 STO a	Av				

Step	X	Y	Z	T	L	Notes
54♦LBL 01						
55 RCL a						
56 XROM "L"						
57 STO a	Av					
58 CLX						
59 "V, K"						
60 PROMPT	V					
61 X=0?						
62 GTO A						
63 24	24					
64 RCL 04	d					
65 2	2	d	24	V		
66 /	d/2	24				
67 X>Y?						
68 X<>Y	max spa					
69 STO [						
70 R↑	V					
71 .85	φ					
72 FC? 06						
73 SIGN						
74 /	V					
75 RCL 02	Vc					
76 2						
77 /	Vc/2	V				
78 X>Y?						
79 ".5Vc>V"						no shear reinf. needed
80 X>Y?						
81 AVIEW						
82 X>Y?						
83 GTO 01						
84 ST+ X	Vc	V				
85 -	Vs					
86 RCL 08	√f'c					
87 FS? 06						
88 ST+ X						
89 RCL 14	b					
90 RCL 04	d	b	√f'c	Vst		
91 *	bd					
92 *	VLimit	VST				
93 /	VST/VLimit					
94 2	2					
95 X>Y?						
96 SIGN						
97 ST/ [						Max.Spa. in M…
98 CLX						
99 1.1						1.1 only for ACI
100 FC? 06						FS?06 or FC?04
101 FC? 04						
102 SIGN	1 or 1.1					
103 4						
104 *	4 or 4.4	VST/VLimit	VST	VST		
105 X>Y?						
106 GTO 00						
107 "V-Vc NG"						
108 BEEP						
109 AVIEW						
110 GTO 01						

D81

Program VST For Shear Reinforcing Design 263 bytes

Step	X	Y	Z	T	L	
111♦LBL 00						
112 RCL a	A_v					
113 R↑	V_{st}					
114 X<0?						
115 CLX						
116 X≠0?						
117 /						
118 FS? 06						
119 RCL 01	f_y					$Spa. = \frac{A_v f_{st} d}{V_{st}}$
120 FC? 06						
121 RCL 13	f_s					
122 *						
123 RCL 04	d					
124 *	Spa					
125 X=0?						
126 E3	1000 or Spa					
127 RCL a	A_v					
128 RCL 01	f_y					
129 RCL 14	b	f_y	A_v	Spa		$Max. Spa. = \frac{A_v f_y \cdot 20}{b}$
130 /						
131 *						
132 20						
133 *	Max Spa	Spa	Spa	Spa		
134 X>Y?						
135 X<>Y	Spa					
136 RCL [	Max					
137 X>Y?						
138 X<>Y	Spa.?	Max	Max'	Spa		
139 R↑	Spa.					
140 X<>Y	Spa?	Spa	Max	Max'		
141 "SPA="						
142 ARCL X						
143 "⊢ "						
144 X≠Y?						
145 "⊢(MAX)"						
146 AVIEW	Spa.					
147 GTO 01						
148 END						

D82

CONCRETE CORBEL DESIGN

xeq CORBL (SIZE 018)

SOFTKEYS (USER mode ON)

d	Estimate required depth given width and load
D	New corbel dimensions

USER FLAGS

Flag 00	clear	Full output
	set	Only PRIM A-S and SEC A-S
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 04	clear	AASHTO
	set	ACI

INPUT SUMMARY

Prompt	Input	Default
WSD?	Asking if by working stress design. Key in "1" for yes (WSD), "0" for no (LFD)	1 (WSD)
AASHTO or ACI	A reminder, not a prompt; push R/S	
F-Y 40,60 (LFD only)	Reinforcing yield stress, ksi	60 ksi
N (WSD only)	Modular ratio, E-st/E-conc	8
F-S 20,24 (WSD only)	Reinforcing working stress, ksi	24 ksi
Fc'KSI	Concrete strength in ksi (always ultimate strength)	4 ksi
MU 1,1.4	Friction coefficient. 1.4 for integrally cast concrete, 1.0 for "intentionally roughened" cold joint, 0.6 for cold joint	No default
B"	Corbel width, inches	Previous B
D"	Depth from top reinforcing to bottom of corbel, in inches	Previous D
H"	Total depth of corbel, in inches	Previous H

Prompt	Input	Default
ARM"	Distance from face of supporting member to centroid of vertical load, in inches	No default
V, K	Vertical load, in kips. If bypassed or zero input, prompts for new dimensions	Zero
TENS Nc, K	Horizontal tension force due to temperature, shrinkage, etc.	V/5

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(A-V)	The area of steel for shear
(A-F)	The area of steel for bending
(A-N)	The area of steel for horizontal tension
(RHO-MIN)	Minimum reinforcing ratio, .04 Fc/Fy, for PRIM A-S
PRIM A-S	Primary (top) layer of reinforcing
SEC A-S	Secondary layers of reinforcing

PROGRAM FLAGS

Flag 06	clear	WSD
	set	LFD
Flag 07	clear	bending moment OK
	set	NG--too much bending moment in moment reinforcing subroutine
Flag 09	clear	LFBM or WSBM
	set	CORBL
Flag 10	clear	subroutine N runs full length

STORAGE REGISTER USE

Register	Value
00	.85 Fc'/Fy
01	Fy, ksi
02	(H") - (D"), to calculate moment due to TENS Nc, K
03	B" (width, inches)
04	D" (depth from primary reinf. to bottom of corbel)
05	ϕBD^2 (ϕ = .85 for shear and corbel moment)
06	V, K (vertical load, kips); 2/3 Av
07	H"; TENS Nc, K (horizontal tension force, kips)
08	Allowable concrete shear stress, ksi
09	N (modular ratio) (WSD only)
10	D/3 (WSD trial depth for neutral axis iteration)
11	Trial for iterative solution to neutral axis position (WSD only)
12	.45 Fc' (ACI) or .40 Fc' (AASHTO) (WSD only)
13	F-S KSI (allowable steel stress) (WSD only)
14	R_{ideal} (see example for equations) (WSD only)
15	ARM" (distance from center of load to column face)
16	Mu (friction coefficient)
17	ϕ Fy (LFD) or Fs (WSD) for horizontal tension

CONCRETE CORBEL DESIGN

Program CORBL designs the reinforcing for concrete corbels using either load factor (ultimate strength) or working stress (service) design. It follows AASHTO if flag 04 is clear, and ACI if flag 04 is set. AASHTO uses somewhat lower allowable stresses for WSD; for LFD the two codes are identical (except for factoring the loads, which is done by the user).

If flag 00 is set the display of A-F, A-N, A-V (the three component reinforcing areas), and RHO-MIN (the minimum primary reinforcing ratio) is suppressed; the program outputs only the necessary design quantities PRIM A-S and SEC A-S, the primary and secondary areas of steel.

WARNINGS

CORBL checks several items and displays the following warnings (with BEEPs if flag 02 is set) if they are violated:

V > Vc	Shear is greater than allowable concrete shear stress; new dimensions are prompted
Nc > V	Horizontal tension is greater than shear; new loads are prompted
ARM > D	The position of the load with respect to the face of the supporting member is greater than the depth of the corbel; new dimensions are prompted

The program also checks that the calculated primary (top) area of reinforcing is greater than the specified minimum.

LOAD and UNDER-CAPACITY FACTORS

The designer must be careful to input the correct loads to CORBL. If using LFD (Strength) design methods, the forces must have the load factor applied <u>by the user.</u> However, they must <u>not</u> include the under-capacity factor Phi (0.85), which is applied by the program. If using working stress design, this is not a problem.

HORIZONTAL TENSION

Both codes specify a minimum amount of horizontal tension on the corbel (due to thermal effects on the beam, concrete shinkage, wind, etc.) of at least one-fifth the vertical load. If a "TENS Nc, K" force less than this is input, the program uses one-fifth V. CORBL also checks that this horizontal force is no greater than the vertical force, as mentioned above.

LISTING

CORBL is within the same program as WSBM and LFBM; the program listing has been placed with the documentation for these two programs.

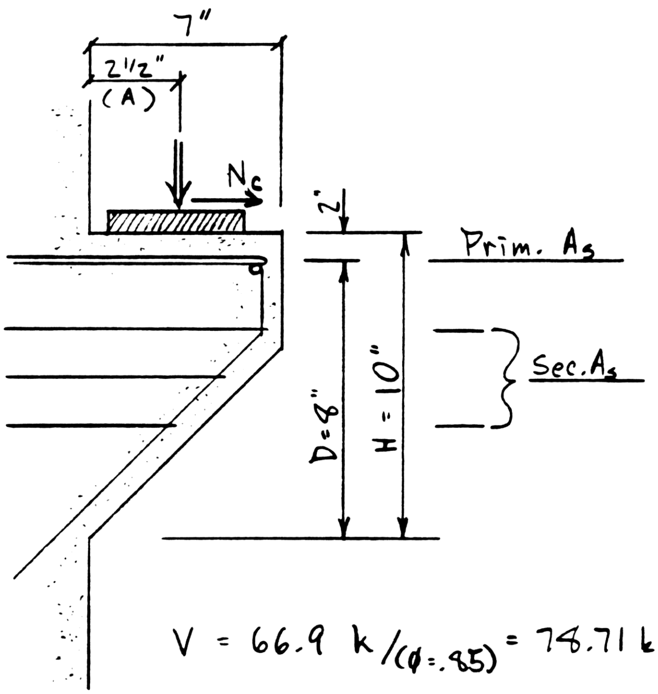
Design a corbel on a 14" wide column by LFD for the following loads, by AASHTO:

DL 30.3 k
LL $\frac{5}{3} \times 12.7 = 21.17$
$51.47 \cdot 1.3 = \underline{66.9 \text{ k}}$

$f'_c = 4.0$ ksi
$f_y = 60$ ksi
$\phi = 0.85$

Cast monolithicly $\Rightarrow \mu = 1.4$ (friction factor)

Try as follows:



$V = 66.9 \text{ k} / (\phi = .85) = 78.71 \text{ k}$

Horizontal Tension $N_c = 0 \Rightarrow$ Use min.

$N_{uc} = \frac{V_u}{5} = 15.74$ k (AASHTO 8.16.6.8.3 d)

Width B = 14" (same as column)

Shear Friction Reinforcement

V_n : $.2 f'_c = .2 \times 4000 = 800$

$800 b_w d = .80 \times 14" \times 8" = 89.6$ k

$89.6 \text{k} > 78.71 \text{ k}$ OK

AASHTO 8.16.6.4: $A_{vf} = \frac{V}{\phi f_y \mu}$

$A_{vf} = \frac{78.71 \text{ k}}{60 \text{ ksi} \times 1.4} = \underline{.9370 \square"}$

Moment Reinforcement

$M = V_u A + N_{uc} (h - d)$

$= 78.71 \times 2.5" + 15.74 \cdot 2"$

$= 228.26"^{\cdot k} = 19.02'^{\cdot k}$

$\frac{M}{\phi b d^2} = \frac{228.26 \times 1000}{.85 \times 14" \cdot (8")^2} = 254.75$

$m = \frac{f_y}{.85 f'_c} = 17.6471$

$\rho = \frac{1}{m} \left(1 - \sqrt{1 - \frac{2m (M/\phi b d^2)}{f_y}}\right)$

$= .0044$

$A_f = .0044 \times 8" \times 14" = \underline{.4948 \square"}$

Tension Reinforcement

$A_n = \frac{N_{uc}}{\phi f_y} = \frac{15.74 \text{ k}}{60 \text{ ksi}} = \underline{.2623 \square"}$

Primary A_s

$A_f + A_n = .4948 + .2623 = \underline{.7572\ \square''}$

$\frac{2}{3} A_{vf} + A_n = \frac{2}{3}(.9370) + .2623$

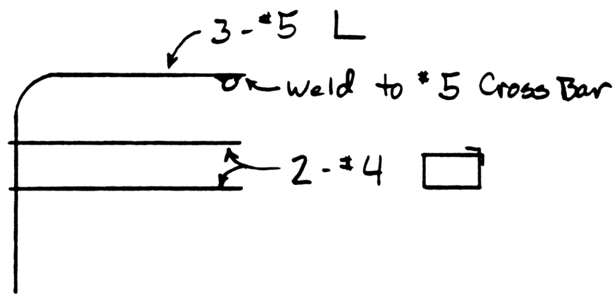
$= \underline{.8870\ \square''}$ ← Governs

Secondary A_s

$A_h = \frac{1}{2}(.8870 - .2623) = \underline{.3123\ \square''}$

Prim. A_s - provide <u>3-#5</u> ($A_s = .93\square'' > .887$)

Sec. A_s - provide <u>4-#4</u> ($A_s = .80\square'' > .3123$)



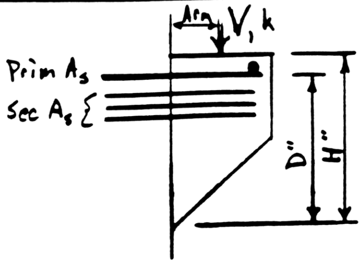
Check Minimum Reinforcing

AASHTO 8.16.6.8.5 :

$\rho_{min} = .04 \frac{f_c'}{f_y} = .04 \cdot \frac{4}{60} = \underline{.0027}$

$A_{s\cdot min} = .0027 \times 8'' \times 14'' = .299\ \square''$

$.8870 > .299$ <u>OK</u>




```
          XEQ "CORBL"
 7:27:07 AM  07/16/87

TITLE:
DEPTH EST.              RUN
************************
WSD?
            0.0000      RUN

AASHTO
F-Y 40,60
           60.0000      RUN
Fc'KSI
            4.0000      RUN
MU 1,1.4
            1.4000      RUN
---------
B"
           14.0000      RUN
D"
            0.0000      RUN
V
           66.9000      RUN
TRY H=9.9732"
---------
B"
           14.0000      RUN
D"
            8.0000      RUN
H"
           10.0000      RUN
ARM"
            2.5000      RUN
************************
V, K
           66.9000      RUN
TENS Nc, K
            0.0000      RUN
A-V=0.9370 SQ"
A-F=0.4948 SQ"
A-N=0.2624 SQ"
RHO-MIN=0.0027

PRIM A-S=0.8870 SQ"
SEC A-S=0.3123 SQ"
---------
V, K
           90.0000      RUN
V>Vc
---------
```

Load Factor Design (WSD? 0.0000)

friction coeff. (MU 1.4000)

depth unknown (D" 0.0000)

Estimated Depth (TRY H=9.9732")

Concrete Overstress for V=90.0 k

```
B"
           14.0000      RUN
D"
           10.0000      RUN
H"
           12.0000      RUN
ARM"
            2.5000      RUN
************************
V, K
           90.0000      RUN
TENS Nc, K
            0.0000      RUN
A-V=1.2605 SQ"
A-F=0.5294 SQ"
A-N=0.3529 SQ"
RHO-MIN=0.0027

PRIM A-S=1.1933 SQ"
SEC A-S=0.4202 SQ"
---------
V, K
           66.9000      RUN
TENS Nc, K
            0.0000      RUN
A-V=0.9370 SQ"
A-F=0.3900 SQ"
A-N=0.2624 SQ"
RHO-MIN=0.0027

PRIM A-S=0.8870 SQ"
SEC A-S=0.3123 SQ"
---------
V, K
                        RUN
---------
B"
```

Try new section (B" 14.0000)

Try new section (final B")

```
          XEQ "CORBL"
 7:33:55 AM  07/16/87

TITLE:
CORBEL DESIGN-PED RAMP
                      RUN
************************
WSD?
         1.0000     RUN
```

Working Stress Design

```
AASHTO
N
         8.0000     RUN
F-S 20,24
        24.0000     RUN
Fc'KSI
         4.0000     RUN
MU 1,1.4
         1.4000     RUN
---------
B"
        14.0000     RUN
D"
         8.5000     RUN
H"
        10.5000     RUN
ARM"
         2.5000     RUN
************************
V, K
        43.0000     RUN
V>Vc
---------
```

Corbel too small

```
B"
         4.0000     CLX
```

oops

```
        14.0000     RUN
D"
        10.5000     RUN
H"
        12.5000     RUN
ARM"
         2.5000     RUN
************************
V, K
        43.0000     RUN
TENS Nc, K
         0.0000     RUN
A-V=1.2798 SQ"
A-F=0.5328 SQ"
A-N=0.3583 SQ"
RHO-MIN=0.0027

PRIM A-S=1.2115 SQ"
SEC A-S=0.4266 SQ"
---------
V, K
        43.0000     RUN
TENS Nc, K
        44.0000     RUN
Nc>V
---------
```

Tension greater than allowable

```
V, K
        43.0000     RUN
TENS Nc, K
         0.0000     RUN
A-V=1.2798 SQ"
A-F=0.5328 SQ"
A-N=0.3583 SQ"
RHO-MIN=0.0027

PRIM A-S=1.2115 SQ"
SEC A-S=0.4266 SQ"
---------
V, K
                    RUN
---------
B"
```

no more loads - new section

SHORT FORM, with flag 00 set:

```
                SF 00
          XEQ "CORBL"
WSD?
         1.0000     RUN
```

no title block

```
N
         8.0000     RUN
F-S 20,24
        24.0000     RUN
Fc'KSI
         4.0000     RUN
MU 1,1.4
         1.4000     RUN
---------
B"
        14.0000     RUN
D"
        10.5000     RUN
H"
        12.5000     RUN
ARM"
         2.5000     RUN
************************
V, K
        43.0000     RUN
TENS Nc, K
                    RUN

PRIM A-S=1.2115 SQ"
SEC A-S=0.4266 SQ"
---------
V, K
```

no breakdown of reinf. calcs - only final steel areas

REINFORCING BAR WEIGHT SUMMATION

xeq BWT (SIZE 004)

SOFTKEYS (USER mode ON)

A Gives total weight of reinforcing (∑), then returns to program. Can be checked at any time.

B Common BAR size for succeeding bar sets. Bypass the prompt to return the program to the BAR # prompt at each bar set.

C Correct an error. Subtracts the weight of the last bar set input from the sum of the weights.

c Clears the sum (and the common BAR size, if any) and starts over. (The total is also zeroed every time BARWT is executed.)

USER FLAGS

Flag 00	clear	Full title block if printing.
	set	Title block is skipped if printing; output is not affected by flag 00.

INPUT SUMMARY

Prompt	Input	Default
BAR #	Standard CRSI bar sizes 3-11, 14, and 18. Also recognizes 1/4" pencil rod (#2).	No default
QUANT	The quantity of bars in a given bar set	No default
L FT.IN	Length of one bar in feet and <u>inches.</u> Thus 12'-7 is input 12.07. For a variable bar length, key in the shortest length, enter/, then the longest length.	No default
BAR	(From softkey B) The bar size common to succeeding bar sets. Bypass to get BAR# prompt back for each bar set.	Zero

OUTPUT SUMMARY (Flag 00 has no effect on program BWT)

WT	Weight of a given set of bars.
Σ	(From softkey A) Total weight thus far.
Σ - Weight = New Σ	(From softkey C) The corrected total after subtracting an incorrect entry.

PROGRAM FLAGS

Flag 21	set	Controls display/printing
Flag 27	set	Turns USER mode on

STORAGE REGISTER USE

Register	Value
00	Bar weight for a given size bar; Weight x Quantity
01	Current bar set weight (for softkey C)
02	Summation of weights
03	Common bar size

D94

REINFORCING BAR WEIGHT SUMMATION

Program BWT is used for calculating and summing the weights of reinforcing bars. Upon execution, BWT prompts for the bar size (BAR #), the number of bars of that type (QUANT), and the length of each bar (L FT.IN) in feet and inches. Six feet eleven inches is input as 6.11; 12'-3 is 12.03. See the notes concerning length and variable bar sets on the next page.

BAR SIZE

For regular reinforcing bar sizes, answer the BAR # prompt with the bar size (3-11, 14, 18); for 1/4" pencil rod, respond with a 2. Responding with other numbers will give strange results because the program uses a weight calculation algorithm based on the bar number to figure a given bar's weight. For instance, the program calculates the weight of a #12 bar as 6.428 pounds per foot, between that of a #11 and #14 bar.

If the bar list has many bars of the same size (diameter), press softkey B and key in that size in response to the BAR prompt. The BAR# prompt will be skipped for succeeding bar sets, and the program will use the BAR size input for softkey B. This feature saves a lot of repetitive input. To get the BAR# prompt back (e.g. if the bar size changes), press softkey B again and bypass the BAR prompt (press R/S without keying anything in).

WEIGHT SUMMATION

As mentioned, the program adds all weights to register 02, which is zeroed at the beginning of the program. This can also be accomplished by pressing softkey c (shift, SQRT). The program gives no evidence that it has cleared anything; the first time through, the user may wish to use softkey A to check it. See the final paragraph, regarding accidental clearing.

BWT actually sums the <u>rounded-off</u> weights, as determined by the display format. Since bar lists usually state the weights to the nearest pound or tenth of a pound, it is suggested to set the calculator to FIX 0 or FIX 1 format, respectively, before starting (or soon thereafter). Then the sum figured by the program will be the same as the total of the weights on the bar list; there is no round-off difference.

In order to get the total of the weights (which is, after all, why we're doing this) press softkey A. This can be done at any time, as many times as desired. After viewing the sum, pressing R/S gets you back into the input sequence, carrying your total. As mentioned above, press softkey c (lowercase; shifted) to clear the total and start over.

LENGTHS

As mentioned, lengths are input in feet and inches, which is usually how bar lists state them. If a length is mistakenly input in decimal feet (e.g. 6.17 for 6'-2"), the program will catch it and shunt you back to the L FT.IN prompt for another go. It does this by checking that the fractional part of the length is less than .12; the only thing it can't catch is an input of 6'-1", for instance, as 6.08, which it sees as 6'-8".

VARIABLE BAR SETS

For a group of variable bars, key in the length of the longest bar, enter/ , then key in the length of the shortest bar and press R/S. Again, input must be in feet and inches. The program takes the average of these lengths, times the number of bars (QUANT) input. It's good to try this a few times to get comfortable with it.

ERROR CORRECTION

There are several ways to make mistakes with this program. The first and easiest, of course, is to simply key in the wrong number in response to a prompt. This is fixed by pressing softkey C (for "correct") after the incorrect weight has been displayed. The program will remove this weight from the total and return you to the input sequence. This can be done any time between when the incorrect weight is displayed and when the next weight is calculated and displayed (i.e. any time before you key in the next L FT.IN and press R/S). If you realize too late and finish inputting another bar, re-input the incorrect bar (incorrectly, as before) and then press softkey C twice, thus subtracting the incorrect weight twice. All of this, of course, hinges on realizing that a mistake has been made in the first place, which is the really tricky part.

Another mistake that happens when using the softkey B feature is to get used to bypassing the bar size, and then forget to input it when you're supposed to. Then, if you're paying more attention to the bar list than the calculator (the usual case), the BAR # prompt gets the quantity input and the QUANT prompt gets the length input. If the length happened to be an even number of feet, you'll be surprised by a L FT.IN prompt instead of a weight. The easiest way to take care of this is to press softkey B. If you try to input a non-integer number for the QUANT prompt (i.e. the length had inches) the program will realize what happened and do

D96

this for you. This mistake happens the other way around just as often--inputting the BAR# at the QUANT prompt, and the QUANT at the Length prompt. Use softkey C to fix this one.

Potentially the worst mistake is to press softkey c (shifted, to clear) instead of C (unshifted, to Correct). The shifted key was chosen for "clear" because it takes more effort to do, but in spite of this innovative protection scheme the mistake still happens. When it does, you'll be surprised by a BAR# prompt instead of the reassuring " Σ - Weight = New Σ" display. However, before the program zeroes the total weight, it recalls it. To continue working, press RDN (roll down) to bring the total into the X-register, store it in register 02 (STO 02), then press softkey B and take it from there.

D97

BILL OF BARS - SLAB

BAR	NO.	LENGTH	WEIGHT
S501	54	2'-7	145.5
S602	10	20'-10	312.9
S503	12	Varies 2'-0 / 7'-10	61.5
S1004	32	46'-0	6334.0

501: 54 × 2.583' × 1.043 = 145.5 lbs.
602: 10 × 20.833' × 1.502 = 312.9
503: 12 × ½(2.0 + 7.83') × 1.043 = 61.5
1004: 32 × 46.0' × 4.303 = 6334.0

6853.9

```
                    FIX 1
               XROM "BWT
12:43 PM  04/10

TITLE:
SLAB                  RUN
************************
BAR #
               5.0    RUN
QUANT
              54.0    RUN
L FT,IN
             145.5    RUN
L FT,IN
              2.07    RUN
WT=145.5
BAR #
               6.0    RUN
QUANT
              10.0    RUN
L FT,IN
             20.10    RUN
WT=312.9
BAR #
               5.0    RUN
QUANT
              12.0    RUN
L FT,IN
               2.0 ENTER^
              7.10    RUN
WT=61.5
BAR #
              10.0    RUN
QUANT
              32.0    RUN
L FT,IN
              46.0    RUN
WT=6,334.0
BAR #
                    XEQ A
Σ=6,853.9
BAR #
```

D96

Program BWT For Reinforcing Bar Weight Calculation 184 bytes Page 1 of 2

Step	X	Y	Z	T	L	Notes
01♦LBL "BWT"						
02 3						
03 XROM "S"						
04 SF 27	USER	mode				
05♦LBL c						
06 SF 21						
07 CLX						
08 STO 03	0					
09 X<> 02	Old Σ					
10♦LBL 02						
11 "BAR #"						
12 RCL 03	Common					
13 X=0?						
14 PROMPT	Bar #					
15 8	8					
16 X<Y?						
17 GTO 01						
18 /	Dia."					
19 2						
20 /	Radius					
21 X↑2	r^2					
22 PI	π					
23 *	Area					
24 GTO 00						
25♦LBL 01						
26 X<>Y	Bar #	8				
27 13	13					
28 X<Y?						
29 GTO 01						
30 SIGN	1	Bar #	8			
31 GTO 02						
32♦LBL 01	13	Bar #	8			
33 RDN						
34 2	2	Bar #	8			
35♦LBL 02						
36 -	Bar'	8				
37 X<>Y	8					
38 /	Dim					Dim = Side of equiv. square bar
39 X↑2	Area					
40♦LBL 00	Area					
41 3.4						$\frac{490\ pcf}{144\ \square''/\square'} = 3.4$
42 *	Wt.					
43 RCL d	flags					
44 X<>Y	Wt	flags				
45 FIX 3						
46 RND	Weight	flags				
47 STO 00	Wt.					
48 RDN	flags					Req.d the flag register.

Step	X	Y	Z	T	L	Notes
50 CLX	0					
51 "QUANT"						
52 PROMPT	Quant.					
53 ST* 00						Check if QUANT prompt was bypassed.
54 X=0?						
55 GTO 02						
56 FRC	Frc=0?					Check if L FT.IN accidentally input. Quantity must be an integer
57 X≠0?						
58 GTO 02						
59♦LBL 07						
60 .	0					
61 "L FT.IN"	L Ft.In					
62 PROMPT	L_1	L_2?				
63 XEQ 00	L					
64 ENTER↑	L_1	L_1	L_2?	L_2?		
65 R↑	L_2?	L_1	L_1	L_2?		
66 X>0?						
67 XEQ 00	L_2?					
68 X=0?						
69 RDN	L_1 or L_2	L_1				
70 +						
71 2						
72 /	L_1 or Avg.					
73 RCL 00	Wt×Quant					
74 *	Wt					
75 RND	Rounded					
76 ST+ 02	(Σ)					
77 STO 01	Weight					
78 "WT"						
79 GTO 04						
80♦LBL B						
81 .						
82 "BAR"						
83 PROMPT						
84 STO 03	Common Bar					
85 GTO 02						
86♦LBL A						
87 "Σ"						
88 RCL 02						
89 GTO 04						
90♦LBL C						
91 "Σ-"						
92 ARCL 01						
93 RCL 02	Σ					
94 RCL 01	Last Wt.					
95 -	Prev. Σ					
96 STO 02	Σ					

D99

Program BWT For ______ 144 bytes Page 2 of 2

Step	X	Y	Z	T	L
97♦LBL 04					
98 "⊢="					
99 ARCL X					
100 AVIEW					
101 GTO 02					
102♦LBL 00	3.07	Y			
103 INT	3				
104 LASTX	3.07				
105 FRC	.07	3	Y		
106 .12	.12	.07	3	Y	
107 X<=Y?					
108 GTO 07					
109 /	.583	3	Y	Y	
110 +	3.583	Y	Y	Y	
111 END					

Use example length
3.07 = 3'-7"

D100

<u>**RECTANGULAR CONCRETE COLUMN ANALYSIS**</u>

Program CC is a subroutine which calculates points on the load-moment interaction diagram for a rectangular reinforced concrete column. The points calculated represent levels of force in the tension-side reinforcing, and are ultimate (factored) strengths only. The following points are calculated:

MAX	The maximum usable axial load and the corresponding moment
0T	The eccentricity producing zero force in the tension-side reinforcing
25%T	The eccentricity producing a stress of 15 ksi in the tension reinforcing
50%T	The eccentricity producing a stress of 30 ksi in the tension reinforcing
BAL	The eccentricity producing a "balanced" condition, with concrete and steel reaching their ultimate strengths simultaneously

Output consists of axial loads in kips and moments in foot-kips, stored in numbered storage registers. CC has no displayed output; it is solely a computational subroutine requiring its input in the expected storage registers. Lack of space in the ROM prevented CC and its calling program CCOL from being included. However, a listing of both programs is included for use or as an example, to be keyed in and run in main memory (RAM). CCOL is a very simple input/output shell program for subroutine CC; however, its results are useful and the program can be expanded by the user.

Program CC uses modified versions of the CRSI Universal Strength Formulas, and includes all the limitations and assumptions of these formulas.

As mentioned, CC calculates only ultimate strengths; all values have been reduced by a phi-factor of 0.70, appropriate for strength design of tied columns. However, program CCOL is written to analyze by either strength or allowable stress (LFD or WSD) methods. In working stress design, CCOL uses O = 1.0 but factors the allowable loads and moments down to 0.35 (AASHTO) or 0.40 (ACI) of their ultimate levels.

D101

RECTANGULAR CONCRETE COLUMN ANALYSIS

Note: this document serves as the Summary Sheet for both subroutine "CC" and main program "CCOL", which uses CC. Both CCOL and CC must be keyed into memory by the user; space constraints prevented them from being included in the ROM. The only sections which pertain directly to CC are the SIZE requirement and storage register information.

(SIZE 019)

SOFTKEYS (USER mode ON)

A	New area of steel
D	New concrete dimensions

USER FLAGS

Flag 00	clear	Full output
	set	Title block skipped
Flag 04	clear	AASHTO
	set	ACI

INPUT SUMMARY

Prompt	Input	Default
WSD?	Asking if working stress or load factor design. Answer 1 (yes) or bypass for working stress design, 0 (no) for load factor (strength).	1 (WSD)
AASHTO or ACI	A reminder, not a prompt. Press to continue if not using a printer.	
F-S 20,24 or F-Y 40,60	Reinforcing working or yield stress in ksi	60 ksi
Fc' KSI	Ultimate concrete strength	4.0 ksi
B"	Column width in inches	Previous B
D"	Column depth (full dimension, not depth to reinforcing) in direction of bending.	Previous D
D" TO CL BAR	Distance from concrete surface to center of reinforcing bar (cover + tie + 1/2 bar diam.)	No default

D102

Prompt	Input	Default
#BARS	Total number of bars in column (all bars must be the same size)	No default
BAR A SQ"	The area of one bar in square in.	No default
#SIDE BARS	The number of bars not in either the tension or compression face	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

Output for CCOL consists of the following five points on the column's load-moment interaction diagram, each having an allowable axial load P in kips, and a corresponding moment M in foot-kips. The reinforcing ratio is also calculated, displayed, and checked to be between .01 and .08; and the value of 0.1 fc' x A_g is displayed.

BAL	The point of balanced stress between concrete and tension reinforcing.
MAX	The point of maximum usable axial load (0.80 P_0)
0T	The point producing zero stress in the tension reinforcing
25%T	The point producing one quarter of the allowable stress in the tension reinforcing
50%T	The point producing half the allowable stress in the tension reinforcing

PROGRAM FLAGS

Flag 06	clear	Working stress design (.35 or .40 times the ultimate strength)
	set	Ultimate strength (load factor) design

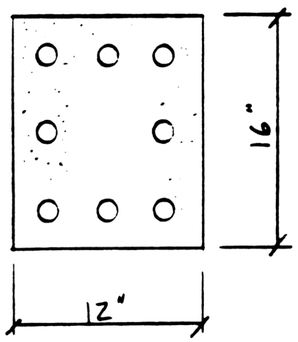
STORAGE REGISTER USE

Register	CC Value	Add'l CCOL Value
00	Fc', ksi	
01	β_1	
02	F-Y KSI	
03	B"	
04	D"	
05	# Side Bars	Counter

Register	CC Value	Add'l CCOL Value
06	"g" (bar separation/depth)	
07	Non-side-bar A_s/bd	
08	Side-bar A_s/bd	(.35 or .40)/ϕ or 1.0
09	#BARS; BAL P	
10	1 bar area; BAL M	
11	MAX P (.80 $f_c'A_g$ + A_s F_y whether by WSD or LFD)	
12	MAX M	
13	0T P	
14	0T M	
15	25%T P	
16	25%T M	
17	50%T P	
18	50%T M	

D104

Analyze this column by "strength" design and compare to tabular values in CRSI Handbook.



8 - #8 bars

$f_y = 60$ ksi
$f_c' = 5.0$ ksi
$\phi = .70$

Cover	1½"
tie	3/8"
½ bar	½"
	2.38" = d′

\# Side bars = 2

CRSI (Major Axis):

Point	P, k	e (in)	M'-k
Balanced	224 k	9.73"	181.6'-k
Max (.80 P_o)	654 k	1.43"	77.9'-k
Zero Tension (Interp.)	516.2	2.93"	126.0'-k

```
          XEQ "CCOL"
 5:01:07 PM  01/24/89

TITLE:
CRSI TRIAL           RUN
************************
WSD?
          0.0000     RUN
AASHTO
F-Y 40,60
         60.0000     RUN
Fc'KSI
          5.0000     RUN
B"
         12.0000     RUN
D"
         16.0000     RUN
D" TO CL BAR
          2.3800     RUN
---------
# BARS
          8.0000     RUN
BAR A  SQ"
           .7900     RUN
# SIDE BARS
          2.0000     RUN
-----------------------
RHO=0.0329 OK
.1Fc'A=96.0000 K

BAL: P=225.9900 K
 M=181.7334 'K

MAX: P=654.2704 K
 M=69.3719 'K

0T: P=518.1019 K
 M=127.1233 'K

25%T: P=428.1258 K
 M=144.7068 'K

50%T: P=352.9934 K
 M=160.7045 'K
************************
# BARS
```

Note: for reference or use in RAM - not resident in ROM

Program CCOL For Concrete Column Analysis 246 bytes Page 1 of 1

```
01♦LBL "CCOL"
02 18
03 XROM "W"
04 CF 10
05 XROM "N"
06 STO 00
07 RDN
08 X<> 01
09 STO 02

10♦LBL D
11 XROM "BD"
12 "D" TO CL BAR"
13 PROMPT
14 ST+ X
15 RCL 04
16 -
17 LASTX
18 /
19 CHS
20 STO 06

21♦LBL A
22 XROM "L"
23 "# BARS"
24 PROMPT
25 STO 09
26 "BAR A "
27 XROM "IS"
28 PROMPT
29 STO 10
30 "# SIDE BARS"
31 PROMPT
32 STO 05
33 STO 08
34 CHS
35 RCL 09
36 +
37 STO 07
38 RCL 10
39 RCL 03
40 /
41 RCL 04
42 /
43 ST* 07
44 ST* 08
45 XEQ "CC"
46 XROM "LL"
47 9.1
48 STO 05
49 4
50 "RHO="
51 7
52 /
53 FC? 04
54 .5
55 FS? 06
56 SIGN
57 X<> 08
58 RCL 07
59 +
60 CF 07
61 .01
62 X>Y?
63 SF 07
64 8
65 *
66 X<Y?
67 SF 07
68 RDN
69 ARCL X
70 XROM "OK"
71 AVIEW
72 ".1Fc'A="
73 RCL 00
74 RCL 03
75 *
76 RCL 04
77 *
78 10
79 /
80 ARCL X
81 "⊦ K"
82 AVIEW
83 "BAL"
84 XEQ 01
85 "MAX"
86 XEQ 01
87 "0T"
88 XEQ 01
89 "25"
90 XEQ 02
91 "50"
92 XEQ 02
93 SF 14
94 GTO A

95♦LBL 02
96 "⊦%T"

97♦LBL 01
98 ADV
99 "⊦: P="
100 RCL IND 05
101 RCL 08
102 *
103 ARCL X
104 "⊦ K"
105 AVIEW
106 ISG 05
107 CLA
108 RCL IND 05
109 RCL 08
110 *
111 XROM "M="
112 AVIEW
113 ISG 05
114 END
```

X	Y	Z	T	L

D106

Program CC For Rect. Conc. Column Analysis 324 bytes Page 1 of 2

	X	Y	Z	T	L

```
01♦LBL "CC"
02 E
03 RCL 07
04 RCL 08
05 +
06 -
07 LASTX
08 RCL 02
09 *
10 X<>Y
11 .85
12 *
13 RCL 00
14 *
15 +
16 .56
17 *
18 RCL 03
19 RCL 04
20 *
21 STO [
22 *
23 STO 11
24 1.8
25 ENTER↑
26 SIGN
27 XEQ 09
28 STO 13
29 4
30 1/X
31 XEQ 07
32 STO 14
33 .28
34 XEQ 08
35 4
36 /
37 ST+ 14
38 2.7
39 RCL 06
40 1.4
41 *
42 -
43 RCL 06
44 *
45 XEQ 05
46 2
47 /
48 ST+ 13
49 RCL 11
50 .8
51 /
52 STO Y
53 RCL 13
54 -
55 /
56 RCL 14
57 5
58 /
59 *
60 STO 12
61 RCL 06
62 .5
63 +
64 XEQ 05
65 16
66 /
67 6
68 *
69 STO 15
70 1.5
71 LASTX
72 7
73 /
74 XEQ 09
75 ST+ 15
76 .22
77 XEQ 07
78 STO 16
79 4
80 1/X
81 XEQ 08
82 3.2
83 /
84 ST+ 16
85 RCL 06
86 1.8
87 *
88 XEQ 05
89 4
90 /
91 STO 17
92 1.3
93 .75
94 XEQ 09
95 ST+ 17
96 .19
97 XEQ 07
98 STO 18
99 .24
100 XEQ 08
101 3
102 *
103 8
104 /
105 ST+ 18
106 RCL 06
107 3
108 *
109 2
110 -
111 .
112 XEQ 06
113 LASTX
```

D107

Program CC For ______ ______ bytes Page 2 of 2

```
114 /
115 .15
116 *
117 STO 09
118 SIGN
119 .6
120 XEQ 09
121 ST+ 09
122 .15
123 XEQ 07
124 STO 10
125 .2
126 XEQ 08
127 2
128 /
129 ST+ 10
130 RTN

131♦LBL 05
132 RCL 07

133♦LBL 06
134 X<>Y
135 RCL 08
136 *
137 X<0?
138 CLX
139 +
140 GTO 10

141♦LBL 08
142 RCL 06
143 *
144 .19
145 +
146 .37
147 X>Y?
148 X<>Y
149 STO \
150 RCL 05
151 4
152 X>Y?
153 CLX
154 X=0?
155 GTO 00
156 RDN
157 6
158 X<=Y?
159 SIGN
160  E
161 X=Y?
162 GTO 00
163 R↑
164 R↑
165 12
166 /
167 1/X
168 INT
169 1/X

170♦LBL 00
171 RCL \
172 *
173 RCL 08
174 *
175 RCL 07
176 +
177 RCL 06
178 *
179 RCL 04
180 *

181♦LBL 10
182 RCL 02
183 *

184♦LBL 12
185 RCL [
186 *
187 .7
188 *
189 RTN

190♦LBL 09
191 RCL 06
192  E
193 +
194 *
195 X<>Y
196 RCL 08
197 *
198 -
199 RCL 01
200 *
201 RCL 07
202 -
203 RCL 00
204 *
205 .425
206 *
207 GTO 12

208♦LBL 07
209 RCL 01
210 *
211 RCL 06
212  E
213 +
214 *
215 .5
216 -
217 CHS
218 RCL 04
219 *
220 *
221 END
```

0108

STEEL DESIGN PROGRAMS

This module contains three programs concerned with the design of steel structures:

COMP Composite steel/concrete beam analysis
PLG Slender built-up plate girder analysis
SCOL Steel column analysis (axial loads only)

In addition, another program, BM, which calculates elastic section properties of I- and T-shapes, is also frequently useful for steel design problems.

FLAGS

All the programs recognize flag 00; they are shorter and run faster when it is set. They also recognize flag 04. When flag 04 is set, the AISC Code is followed; when it is clear, AASHTO is used. AASHTO and AISC have much greater differences than do AASHTO and ACI (concrete design) or NDS (timber design). It is therefore very important that the status of flag 04 be checked before using these programs.

Flag 02 is also used by PLG and COMP. These programs have a BEEP at erors and warnings; when flag 02 is clear these BEEPs are disabled. Flags 01 and 03 have no effect on these programs.

STRESSES

All the programs work with stresses in KSI (kips per square inch); this changes the form of some AASHTO equations, making the program listings a little more difficult to trace through.

SUBROUTINES

As mentioned, SCOL deals only with axially loaded columns. However, the working calculation portion of the program has been given a global label, SFa, so that it may be called from a user-written program. In addition, program PLG contains a subroutine dealing with bending stresses, SFb. These are called in a program analyzing steel beam-columns named SBCOL, a listing of which follows. This may be keyed into main memory by the user. Unfortunately, space constraints prevented this from being included in the ROM.

D109

I-BEAM SECTION PROPERTIES

Program BM is used to calculate elastic section properties for I- and T-shaped homogenous beams. It prompts for web depth and thickness, and the width and thickness of each flange, then calculates the moment of inertia, neutral axis position, section moduli of the top and bottom flanges, and the cross-sectional area. In addition, if flag 06 is set, the plastic modulus Z is calculated. If flag 00 is clear, all of these are displayed; if it is set, none of them are.

BM is intended for use as a subroutine. It may be used as a stand-alone program without any problems; however, it differs from most programs in this package on three points:

1) It does not use subroutine S; there is no SIZE check or title block. The user must verify a SIZE of at least 015.

2) Once finished, the program simply stops. It does not return for another section to be input.

3) If flag 00 is set there isn't just minimal output; there is no output at all.

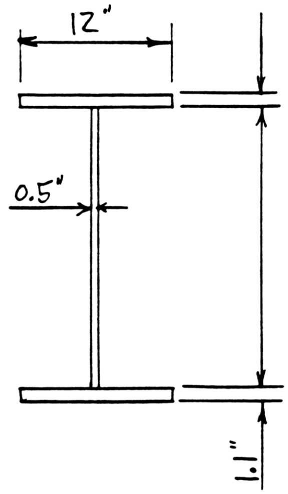
USE as SUBROUTINE

Program BM is used as a subroutine by COMP; all but the web input portion is used by PLG throuh subroutine SBM. Both BM and SBM in turn use subroutine AY (so called because the output is in the stack as A, Y, I, S_b) to actually calculate the section properties from the dimensions input. AY is also used by PIER and LFAN/Ic. AY is one of the three routines in this module that cannot be single-stepped through, due to its use of synthetic register P. See "Synthetic Programming and Other Loose Ends" for more details.

SOFTKEYS

This program uses no softkeys itself. However, it is a part of program COMP and thus will recognize softkeys fron that program. While within COMP (or BM), softkey B will execute BM from the beginning. However, it will then return to COMP. The listing, and storage register use, of program BM are within the documentation for COMP.

"BM": Elastic Section Properties




```
                XEQ "BM"
WEB D"
          34.0000    RUN
TH"
            .5000    RUN
COMP FL W"
          12.0000    RUN
TH"
           1.1000    RUN
TENS FL W"
          12.0000    RUN
TH"
           1.1000    RUN

I=9,771.5947
Y=18.1000"
S-C=539.8671
S-Te=539.8671

AREA=43.4000 SQ"
---------
```

	Area	d	Ad	$\bar{d}$	$A\bar{d}^2$	I_o
Top Fl.	13.2	.55	7.26	17.55	4065.633	1.331
Web	17.0	18.10	307.70	0	—	1637.667
Bottom Fl.	13.2	35.65	470.58	17.55	4065.633	1.331
	43.4	18.10	785.54		8131.266	1640.329
					9771.595	

$$S_{comp} = 9771.595 / 18.10" = 539.867$$

STEEL COLUMN ANALYSIS

xeq SCOL   (SIZE 007)

SOFTKEYS

D	New effective lengths

USER FLAGS

Flag 00	clear	Outputs allowable stress and maximum axial load
	set	Outputs only allowable axial load
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 04	clear	AASHTO
	set	AISC

INPUT SUMMARY

Prompt	Input	Default
WSD?	Asking if the analysis is by working stress (ASD) methods. Answer "0" (no) to analyze by AASHTO LFD or AISC LRFD rules.	1 (WSD)
F-Y KSI	Steel yield stress, ksi	36 ksi
KL-X"	Major axis effective length, in.	No default
KL-Y"	Minor axis effective length, in.	KL-X
AREA	Trial member area, sq. inches. Bypass to re-use the previous section (e.g. after softkey D).	Previous Section
R-X	Major axis rad. of gyration, in.	No default
R-Y	Minor axis radius of gyration	R-X

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(Fa)	Allowable axial stress, ksi
ALL P	Allowable axial load, kips

In addition, the Euler stress is held in the Y-register.

D113

PROGRAM FLAGS

Flag 06	clear	WSD (ASD)
	set	LFD (LRFD)

STORAGE REGISTER USE

Register	Value
00	not used
01	KL-X (inches stored)
02	KL-Y (inches stored)
03	R-X
04	AREA, square inches
05	R-Y
06	F-Y KSI

D114

STEEL COLUMN ANALYSIS

Program SCOL analyzes steel columns of any shape by the AISC or AASHTO codes for allowable service or ultimate (factored) stresses, using program SFa as a subroutine. SCOL prompts for steel yield stress, axial load, and effective lengths in each direction. It then asks for the area and radii of gyration of a trial section, calculates the allowable compressive stress and corresponding load, and returns for a new trial section.

The program displays a warning and BEEPs if the KL/r value is too high (above 120 per AASHTO 10.7.1; and 200 per AISC ASD B7. (1.8.4) or LRFD B.7). It does still output an allowable stress by the appropriate formulae; it is up to the judgement of the designer if the design is acceptable.

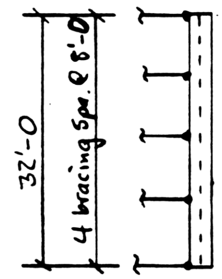
SCOL is not written to analyze beam-columns; however, the allowable-stress subroutine, SFa, was given a global label in order to allow its inclusion in user-written programs for interaction analysis. SFa needs only the effective length Kl/r in the X-register, and F-Y KSI in register 06. It outputs the allowable stress in ksi in the X-register and the Euler stress in the Y-register. The WSD versions of the program for both AASHTO and AISC incorporate the proscribed safety factor (2.12 and 23/12, respectively) into the Euler stress value, while the AASHTO LFD and AISC LRFD versions do not. This follows the way the Euler values are used in the interaction equations. See the examples.

The program is written to analyze prismatic members only; it is unable to account for tapered columns. Built-up members can be analyzed if the separate parts can be considered integrally connected. The user must calculate the radii of gyration. Program BM may be of help for this.

SCOL/SFa assume the flanges and web to be compact, or at least more so than the overall length. The only stress/strength reduction taken is for unbraced length.

D116

Calculate the allowable load on the column shown below.



$KL_y = 8'\text{-}0'' = 96''$; $KL_x = 32'\text{-}0 = 384''$

W6×20: $A = 5.87\ \square''$
$r_x = 2.66''$
$r_y = 1.50''$

$F_y = 36$ ksi

Effective Length

$(Kl/r)_x = \dfrac{384}{2.66} = 144.36$ ← Governs

$(Kl/r)_y = \dfrac{96}{1.50} = 64.0$

$C_c = \sqrt{\dfrac{2\pi^2 E}{F_y}} = \sqrt{\dfrac{2\pi^2 \cdot 29{,}000}{36}} = 126.10$

$144.36 > 126.10 \Rightarrow$ "long"

$\therefore F_a = \dfrac{12\pi^2 E}{23(Kl/r)^2} = 7.166$ ksi

Allowable $P = 7.166 \times 5.87\square'' = \underline{42.06\ k}$

Add a brace at midlength - $kl = 192''$

$(Kl/r)_x = 72.18 < C_c \Rightarrow$ "short"

$\therefore F_a = \left(1 - \dfrac{(kl/r)^2}{2C_c^2}\right) \times \dfrac{F_y}{F.S.}$

$F.S. = \dfrac{5}{3} + \dfrac{3}{8}\left(\dfrac{kl/r}{C_c}\right) - \dfrac{1}{8}\left(\dfrac{kl/r}{C_c}\right)^3$

$= 1.858$

$F_a = \left(1 - \frac{1}{2}\left(\dfrac{kl/r}{C_c}\right)^2\right) \times 36/1.858$

$= 16.20$ ksi

Allowable $P = 16.20 \times 5.87\square''$

$= \underline{95.11\ k}$

For interaction equation,
Euler stress $F_e = \dfrac{12}{23}\dfrac{\pi^2 E}{(kl/r)^2} = 28.66$ ksi

```
            SF 04
         XEQ "SCOL"
3:14:49 AM  01/01/00

TITLE:
ASD                  RUN
************************
AISC
WSD?
          1.0000     RUN
F-Y KSI
         36.0000     RUN
KL-X"
        384.0000     RUN
KL-Y"
         96.0000     RUN
---------
AREA
          5.8700     RUN
R-X
          2.6600     RUN
R-Y
          1.5000     RUN
Fa=7.1655 KSI
Pa=42.0613 K
---------
AREA
                   XEQ D
KL-X"
        192.0000     RUN
KL-Y"
         96.0000     RUN
---------
AREA
                     RUN
Fa=16.2025 KSI
Pa=95.1087 K
---------
AREA
```

Set flag 04 to use AISC

1 = "yes" ⇒ A.S.D.

32'-0 = 384"

8'-0 = 96"

W6×20

Press softkey "D" for new length (Dimension) prompts

Bypass to use same section as previous trial

Same column - W6×20

$A = 5.87\ \square''$
$r_x = 2.66''$
$r_y = 1.50''$

First Case

$K\ell_x = 384''$; $K\ell/r = 144.36$

$K\ell_y = 96''$; $K\ell/r = 64.0$

Check flange b/t: $\frac{6.02''}{2\times.365''} = 8.25$

$\lambda_r = \frac{95}{\sqrt{F_y}} = \frac{95}{\sqrt{36}} = 15.8 > 8.25$ ∴ OK

∴ Can use full allowable stresses

Note: The program does not check flange slenderness (λ_r). The user must check this to determine if SCOL / SFa may be used.

$\lambda_c = \frac{K\ell}{\pi r}\sqrt{\frac{F_y}{E}} = 1.619 > 1.5$

$\Rightarrow F_{cr} = \frac{.877}{\lambda_c^2} \times F_y = 12.04$ ksi

$F_a = (\phi = .85) \times F_{cr} = 10.24$ ksi

Allowable $P = 10.24 \times 5.87 = 60.10$ k

(Factored Load)

Euler stress $F_e = \frac{F_y}{\lambda_c^2} = 13.734$ ksi

for interaction equation

Second Case - Brace @ Midpoint

$K\ell_x = 16' = 192''$

$\lambda_c = \frac{K\ell}{\pi r}\cdot\sqrt{\frac{F_y}{E}} = .8095 < 1.5$

$\Rightarrow F_{cr} = (.658^{(\lambda_c^2)})\, F_y$

$= .658^{0.6553} \cdot F_y$

$= 27.364$ ksi

$F_a = (\phi = .85) \times 27.364 = 23.26$ ksi

Allow. $P = 23.26 \times 5.87\square'' = 136.53$ k

Euler stress $= F_y/\lambda_c^2 = 54.94$ ksi

```
                SF 04
           XEQ "SCOL"
 3:16:36 AM  01/01/00          Set flag 04
                               to use AISC
TITLE:
LRFD                  RUN
************************
AISC
WSD?
              0.0000  RUN      0="no"→LRFD
F-Y KSI
             36.0000  RUN
KL-X"
            384.0000  RUN
KL-Y"
             96.0000  RUN
---------
AREA
              5.8700  RUN
R-X
              2.6600  RUN
R-Y
              1.5000  RUN
Fa=10.2380 KSI
Pa=60.0973 K
---------
AREA
               XEQ D           Softkey "D"
KL-X"
            192.0000  RUN
KL-Y"
             96.0000  RUN
---------
AREA                           Bypass to use
                      RUN      same section
Fa=23.2597 KSI
Pa=136.5343 K
---------
```

D16

Calculate the allowable load on the same column (W6×20) by AASHTO working stress criteria.

W6×20: $A = 5.87$ □"
$r_x = 2.66"$
$r_y = 1.50"$

<u>First Case</u>

$Kl_x = 384"$; $Kl/r = 144.36$ ← Use

$Kl_y = 96"$; $Kl/r = 64.0$

$$C_c = \sqrt{\frac{2\pi^2 E}{F_y}} = 126.10$$

$144.36 > 126.10 \Rightarrow$ "Long"

$$\therefore F_a = \frac{\pi^2 E}{2.12(Kl/r)^2} = 6.478 \text{ ksi}$$

Allowable $P = 5.87 \times 6.48 =$ <u>38.03 k</u>

Euler stress $F_e = 6.478$ ksi

(Note: this column violates AASHTO 10.7.1)

<u>Second Case</u> - Brace @ Mid height

$Kl_x = 192"$; $Kl/r_x = 72.18 > 64.0$

$\therefore Kl/r = 72.18 < (C_c = 126.10)$

$\Rightarrow$ "Short"

$$F_a = \frac{F_y}{2.12} \times \left(1 - \frac{(Kl/r)^2 F_y}{4\pi^2 E}\right) = 14.20 \text{ ksi}$$

Allowable $P = 5.87 \times 14.20 =$ <u>83.35 k</u>

$$\text{Euler stress} = \frac{\pi^2 E}{2.12(Kl/r)^2} = 25.91 \text{ ksi}$$

```
                XEQ "SCOL"
3:13:09 AM  01/01/00

TITLE:
WSD                    RUN
************************
AASHTO
WSD?
          1.0000       RUN     1="yes" → WSD
F-Y KSI
         36.0000       RUN
KL-X"
        384.0000       RUN
KL-Y"
         96.0000       RUN
---------
AREA
          5.8700       RUN
R-X
          2.6600       RUN
R-Y
          1.5000       RUN
KL/R>120.0000                  Warning: AASHTO 10.7.1
Fa=6.4783 KSI
Pa=38.0278 K
---------
AREA
                     XEQ D     Softkey "D"
KL-X"
        192.0000       RUN
KL-Y"
         96.0000       RUN
---------
AREA                           Bypass for
                       RUN     same section
Fa=14.1992 KSI
Pa=83.3492 K
---------
AREA
```

D119

Calculate the allowable load on the same column (W6×20) by AASHTO load factor criteria.

W6×20 : $A = 5.87 \square''$
$r_x = 2.66''$
$r_y = 1.50''$

First Case:

$K\ell_x = 384''$; $K\ell/r = 144.36$ ← Governs

$K\ell_y = 96''$; $K\ell/r = 64.0$

Note: $K\ell/r = 144.36 > 120$; this column violates AASHTO 10.7.1. It is up to the discretion of the engineer whether the results are acceptable.

$C_c = 126.10 < 144.36$ ⇒ "Long"

$$F_{cr} = \frac{\pi^2 E}{(K\ell/r)^2} = 13.734 \text{ ksi}$$

$F_a = (\phi = .85) \times 13.734 = 11.674$ ksi

Allowable P = 11.674 × 5.87 = **68.53 k**

Second Case - Brace @ Midheight

$K\ell_x = 192''$; $K\ell/r_x = 72.18$ ← Governs
$K\ell/r_y = 64.0$

$72.18 > C_c$ (= 126.10) ⇒ "Short"

$$F_{cr} = \left(1 - \frac{(K\ell/r)^2 F_y}{4\pi^2 E}\right) \times F_y = 30.10 \text{ ksi}$$

$F_a = (\phi = .85) \times F_{cr} = 25.59$ ksi

Allowable P = 25.59 × 5.87 = **150.20 k**

(D120) Euler stress $= \frac{\pi^2 E}{(k\ell/r)^2} = 54.94$ ksi

```
XEQ "SCOL"
3:11:30 AM  01/01/00

TITLE:
LFD                     RUN
*************************
AASHTO
WSD?
          0.0000        RUN     0 ⇒ "no" ⇒ L.F.D.
F-Y KSI
         36.0000        RUN
KL-X"
        384.0000        RUN
KL-Y"
         96.0000        RUN
---------
AREA
          5.8700        RUN
R-X
          2.6600        RUN
R-Y
          1.5000        RUN
KL/R>120.0000                   Warning
Fa=11.6739 KSI
Pa=68.5260 K
---------
AREA
                      XEQ D     Softkey "D"
KL-X"
        192.0000        RUN
KL-Y"
         96.0000        RUN
---------
AREA                            Bypass to use
                        RUN     existing section
Fa=25.5869 KSI
Pa=150.1952 K
---------
AREA
```

Program SCOL For Steel Column Analysis 286 bytes Page 1 of 2

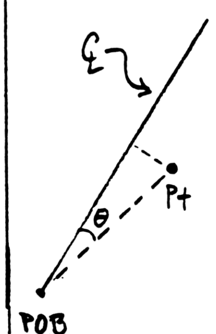
Step	X	Y	Z	T	L
01♦LBL "SCOL"					
02 6					
03 "AISC"					
04 XROM "AA"					
05 CF 06					
06 E	1				
07 "WSD?"					
08 PROMPT					
09 X=0?					
10 SF 06					
11 XROM "FY"	F_y				
12♦LBL D					
13 "KL-X""					
14 PROMPT					
15 STO 01	$K\ell_x$				
16 "KL-Y""					
17 PROMPT					
18 STO 02	$K\ell_y$				
19♦LBL 10					
20 XROM "L"					
21 CF 22					
22 "AREA"					
23 PROMPT					
24 FC? 22					
25 GTO 09					
26 STO 04	Area				
27 "R-X"					
28 PROMPT					
29 STO 03	r_x				
30 "R-Y"					
31 PROMPT					
32 STO 05	r_y				
33♦LBL 09					
34 RCL 02	$K\ell_y$				
35 RCL 05	r_y				
36 /					
37 RCL 01	$K\ell_x$				
38 RCL 03	r_x				
39 /					
40 X<Y?					
41 X<>Y	$K\ell/r$				
42 XEQ "SFa"	F_a	F_e			
43 "Fa"					
44 XROM "KS"					
45 FC? 00					
46 XEQ "PY"					
47 RCL 04	Area				
48 *	P, k	F_e			
49 "Pa="					
50 ARCL X					
51 "├ K"					
52 AVIEW					
53 GTO 10					

Step	X	Y	Z	T	L
54♦LBL "SFa"	$K\ell/r$				$\lambda_c = \frac{K\ell}{\pi r}\sqrt{\frac{F_y}{E}} \overset{?}{>} 1.5$
55 8					$\rightarrow \frac{K\ell}{r} > \sqrt{\frac{2.25\pi^2 E}{F_y}}$
56 "KL/R>"					
57 9					
58 /	8/9				LRFD: $C_c = \sqrt{\frac{2.25\pi^2 E}{F_y}}$
59 FS? 04					$2.25 = \frac{9}{8} \cdot 2$
60 FC? 06					
61 SIGN	1 ?				
62 2					
63 /	1/2				
64 X<>Y	$K\ell/r$	1/2			
65 FC? 04					
66 120	limit				AASHTO 10.7.1
67 FS? 04					AISC 1.8.4
68 200					LRFD B.7
69 ARCL X					
70 X<Y?					
71 BEEP					
72 X<Y?					
73 AVIEW					
74 RDN	$K\ell/r$	1/2			
75 PI	π				
76 /					
77 X↑2	$\left(\frac{K\ell/r}{\pi}\right)^2$				
78 29 E3	E				
79 /					
80 RCL 06	F_y				
81 *	$\frac{(K\ell/r)^2}{\pi^2 E/F_y}$	1/2			
82 STO Z			•		
83 *	$\frac{K\ell/r^2}{C_c^2}$	$\frac{(K\ell/r)^2}{\pi^2 E/F_y}$			
84 CF 19					
85 E	1				
86 X<Y?					
87 SF 19	⇒ "long"				
88 X<> Z	$\frac{K\ell/r^2}{\pi^2 E/F_y}$	$\frac{K\ell/r^2}{C_c^2}$			
89 ENTER↑					
90 1/X	F_e	λ_c^2	$\frac{K\ell/r^2}{C_c^2}$		
91 STO [					
92 FS? 06					
93 GTO 11					
94 FS? 04					
95 1.9167	F.S.	F_e	λ_c^2	$\frac{K\ell/r^2}{C_c^2}$	
96 FC? 04					
97 2.12					
98 /	$F_{e\text{-WSD}}$	λ_c^2	•	•	
99 STO [					
100 FS? 19					
101 GTO 00					
102 X<> L	2.12	λ_c^2			
103 R↑	$\frac{K\ell/r^2}{C_c}$				
104 SQRT	$K\ell/r/C_c$				
105 ST* L					$(\ )^3$
106 3	3	( )	2.12	λ_c^2	$(\ )^3$
107 ST* Y		3( )			
108 X<> L	$(\ )^3$	3( )	2.12	λ_c^2	
109 -	$3(\ )-(\ )^3$	2.12	λ_c^2	λ_c^2	

D121

Program SCOL For ____________ 385 bytes Page 2 of 2

Step	X	Y	Z	T	L
110 8	8				
111 /					
112 .6					
113 1/X	5/3				
114 +	A15° F.S.	2.12	λ_c^2	λ_c^2	
115 FC? 04					
116 X<>Y	F.S.		λ_c^2	λ_c^2	
117 1/X					
118 R↑	$\frac{(kl/r)^2}{\pi^2 E/F_y}$	F.S.			
119◆LBL 01					
120 4					
121 /					
122 SIGN	1				
123 LASTX	$\frac{(kl/r)^2}{4\pi^2 E/F_y}$	1	F.S.		
124 -	1-( )	F.S.			
125 *					
126 GTO 00	F_a				
127◆LBL 11	F_e	λ_c^2			
128 .877	.877				
129 FC? 04					
130 SIGN	1?	F_e	λ_c^2		
131 *					
132 .85	ϕ				
133 *	F_e				
134 FS? 19					
135 GTO 00					
136 X<> L	ϕ=.85	λ_c^2			
137 X<>Y	$\frac{(kl/r)^2}{\pi^2 E/F_y}$	.85			
138 FC? 04					
139 GTO 01					
140 .658	.658	λ_c^2	.85		
141 X<>Y	λ_c^2	.658	.85		
142 Y↑X					
143 *	$\phi \cdot .658^{\lambda_c^2}$				
144◆LBL 00	F_a				
145 RCL [	F_e				
146 X<>Y	F_a	F_e			
147 RCL 06	F_y				
148 ST* Z					
149 *	F_a	F_e			
150 RTN					
151◆LBL "POB"					
152 SF 21					
153 RDN					
154 FS?C 11					
155 GTO IND T					
156 SF 14					
157 "POB"					
158 PROMPT					
159 STO 00	POB				
160 CF 22					
161 "PT ON CL"	Pt	POB			
162 PROMPT					
163 FS? 22					
164 XROM "XY"	ΔX	ΔY			
165 X<>Y	Y	X			
166 R-P	d	Az			
167 X<>Y	Az	d			
168 "AZ"					
169 FC? 22					
170 XROM "Pd"					
171 STO 01	Az				
172◆LBL 04					
173 XROM "L"					
174 "PT"					
175 PROMPT	Pt				
176 XEQ 02	Sta	offset			
177 XROM "PY"					
178 X<>Y	offset				
179 RND					
180 CHS	offset				
181 "OS"					
182 XROM "Y"					
183 GTO 04					
184◆LBL 02					
185 RCL 00	POB				
186 X<>Y	Pt	POB			
187 XROM "XY"	ΔX	ΔY			
188 X<>Y	Y	x			
189 R-P	d	Az			
190 X<>Y	Az	d			
191 RCL 01	℄ Az				
192 -	θ				
193 X<>Y	d	θ			
194 P-R	Sta	Offset			
195◆LBL 03					
196 ENTER↑	Sta	Sta	Offset		
197 "STA="					
198 E2	100	sta	sta	OS	
199 /					
200 INT	sta/100	sta	OS	OS	
201 XROM "AX"					
202 "⊢+"					
203 X<> L	sta/100				
204 FRC	2nd Part				
205 E2	100				
206 *	2nd Part				
207 ARCL X					
208 RDN	sta	offset	offset		
209 END					



STEEL BEAM-COLUMN ANALYSIS

xeq SBCOL (SIZE 032)

Note: SBCOL is not resident in ROM and must be keyed into main memory by the user. A listing follows. The program requires bytes (registers) of memory in RAM.

SOFTKEYS

D New effective lengths

USER FLAGS

Flag 00	clear	Outputs allowable stress and maximum axial load
	set	Outputs only allowable axial load
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 04	clear	AASHTO
	set	AISC

INPUT SUMMARY

Prompt	Input	Default
WSD?	Asking if the analysis is by working stress (ASD) methods. Answer "0" (no) to analyze by AASHTO LFD or AISC LRFD rules.	1 (WSD)
F-Y KSI	Steel yield stress, ksi	36 ksi
AX P, K	Axial load on the member in kips	No default
M,'K	Moment on the member in footkips	No default
KL-X"	Major axis effective length, in.	No default
KL-Y"	Minor axis effective length, in.	KL-X
AREA	Trial member area, sq. inches. Bypass to re-use the previous section (e.g. after softkey D).	Previous Section
R-X	Major axis rad. of gyration, in.	No default
R-Y	Minor axis radius of gyration	R-X

OUTPUT SUMMARY (Items in parenthesis are displayed only if fla 00 is clear)

INTRX	Interaction in the X or Y direction. Should be less than unity.

PROGRAM FLAGS

Flag 06	clear	WSD (ASD)
	set	LFD (LRFD)
Flag 18	clear	WSD (ASD)
	set	LFD (LRFD)

STORAGE REGISTER USE

Register	Value
00	not used
01	KL-X (inches stored)
02	KL-Y (inches stored)
03	R-X
04	AREA, square inches
05	R-Y
06	F-Y KSI
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	

SBCOL FR -3

D 125

SBCCL 1

D 126

SBCOL · 2

D 127

SBCOL ex.

D128

===== Remove using the output option in the Executive =====

3BCE2 ex

D129

SBCOL List 1

D130

===== Remove using the output option in the Executive =====

SBCOL List 2

D131

SBC02 List 3

D132

===== Remove using the output option in the Executive =====

COMPOSITE BEAM ANALYSIS

xeq COMP (SIZE 028)

SOFTKEYS (USER mode ON)

B	New beam input
D	New slab dimension input
E	New moment input

Note: if designing by AASHTO, do not use these softkeys during "N=3N" calculations; allow the program to return to a new beam prompt first.

USER FLAGS

Flag 00	clear	Full output
	set	Only live load capacity, or sums of stresses on flanges
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 04	clear	AASHTO
	set	AISC/ACI

INPUT SUMMARY

Prompt	Input	Default
WSD?	Asking if designing by working stress or load factor design. Respond 1 (yes) for WSD, 0 (no) for LFD/LRFD.	1 (WSD)
PLAST? (LFD/LRFD only)	Asking if section is compact enough to be analyzed by plastic rather than elastic methods. Respond 1 for "yes", 0 for "no." Elastic section properties will still be computed but capacity will be based on plastic properties.	0 (no)
N	Modular ratio, E_{st}/E_{conc}	8
Fc'KSI	Ultimate concrete cylinder strength in kips per square inch	4.0 ksi
F-Y KSI	Steel yield stress, for shear and plastic calculations	36 ksi
Fb KSI (Elastic analysis only)	Allowable steel bending stress. If by LFD, will be near or equal to yield stress.	F_y

D133

Prompt	Input	Default
(STUD D")	Shear stud diameter in inches. Bypass to input capacity directly.	CAP prompt
(STUD CAP, K)	Shear capacity of one shear stud from AISC table 1.11.4, or as calculated by the user for special connectors (e.g. channels) or for reduced capacity due to formed metal deck ribs	No default
SL TH"	Slab thickness, inches	No default
BM SPA'	Beam spacing in feet	No default
SL Be=..."	Displays calculated effective slab width. If OK, press R/S to continue. If not, key in the correct width and press R/S. Does not include slab thickness (B_t + 16 x Th) for AISC ASD.	Displayed
H"	Haunch or stool between the top of the top flange and bottom of slab, or the rib dimension for transverse metal deck, in inches	Zero
NCDL M,'K	Non-composite dead load moment (load applied before slab has achieved its strength), in foot-kips	Zero
CDL M,'K	Composite dead load moment (e.g. from guard rail, future wearing surface, sidewalks, etc.)	Zero
LL M,'K	Live load moment in foot-kips. Bypass (or enter zero) to find remaining live load capacity.	Zero
WEB D"	Web depth (inside of flanges) in inches. Bypass for rolled beam input.	Zero

If inputting a built-up plate member:

WEB TH"	Web thickness in inches.	No default
COM FL W"	Compression flange width, inches. Bypass for rolled beam input.	Zero
TH"	Compression flange thickness, in.	No default
TENS FL W"	Tension flange width in inches	No default
TH"	Tension flange thickness, inches	No default

If inputting a rolled beam:

BM A SQ"	Beam cross-sectional area in square inches	No default
BM D"	Beam depth in inches (out to out)	No default
Y"	Neutral axis position, from the top flange. Used only for cover-plated or rivetted beams; bypass for symmetrical (rolled) beams.	(BM D")/2
I	Moment of inertia of the beam, in $inches^4$	No default

All beams:

#STUDS	Number of shear studs to be used between points of inflection. For a simple span, this is the number of studs on the entire beam. If input is greater than #STUDS REQD or prompt is bypassed, full composite action is assumed.	#STUDS REQD

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(AASHTO or AISC)	Displays code being followed. If not right, change flag 04 and start again. If OK, press R/S to continue.
SL Be	Effective slab width in inches. If OK, press R/S to continue. If not, key in the correct width and press R/S.
(HOR V)	Horizontal shear, in kips (H/2 for AISC ASD)
(#STUDS REQD)	The minimum number of shear studs required to resist the HOR V, taken between points of inflection. See softkey note for HOR V.

If a built-up beam: (if WEB D" prompt was answered)

(I)	Moment of inertia of the steel beam
(Y)	Neutral axis position of the bare beam, in inches, measured from the compression flange
(COM S)	Section modulus of the compression flange, in $inches^3$
(TEN S)	Section modulus of the tension flange
(AREA)	Area of the bare beam, in square inches
(BM Z)	The plastic section modulus of the bare beam (only if by LFD/LRFD)

All beams:

(a)	The effective slab depth used in calculating section properties, as limited by the number of shear studs used.
(Z)	The composite plastic section modulus, in $inches^3$, if plastic analysis was chosen
PLAST M	The full plastic moment capacity of the composite beam, multiplied by O = 0.85 if by AISC LRFD (if plastic analysis was chosen)
LL M	The live load moment capacity of the composite beam (if plastic analysis was chosen)
(N)	The modular ratio (n) used to calculate the (elastic) section properties which follow
(I-COMP)	Composite moment of inertia, $inches^4$
(Y)	Neutral axis position, measured from the top of the slab, in inches
(Sc)	Section modulus of the concrete. Do <u>not</u> divide by N to use. See the examples.
(TF S)	Section modulus of the top (compression) flange, in $inches^3$
(BF S)	Section modulus of the bottom (tension) flange
(EFF S) (AISC ASD only)	The effective section modulus of the bottom flange, reduced due to insufficient shear connectors, by AISC ASD equation I2-1.
(Q) (AASHTO only)	Statical moment of the slab about the neutral axis, in cubic inches

Note: if designing by AASHTO, the above elastic section properties are repeated for a modular ratio of 3N per AASHTO 10.38.1.4. The values for Q, HOR V, and MIN # STUDS are not repeated.

If a live load moment was input:

(B FL) (T FL) (SLAB)	The stresses which follow are on the bottom flange (or coverplate, if there is one), top flange, or slab, respectively.

(NCDL STRS) (CDL STRS) (LL STRS) $\sum$ BF STRS $\sum$ TF STRS $\sum$ SL STRS	Stress due to non-composite dead load, composite dead load, live load, and the summation of these stresses, respectively, in ksi, on the part indicated (top or bottom flange or slab).

If a live load moment of zero was input:

WS or LF LL M	The remaining allowable live load moment, by working stress or load factor design methods, governed by stresses on the bottom flange, top flange, or slab; in foot-kips. For example, WS LL M=107.95 'K means the remaining moment, calculated by working stress design, is 107.95 foot-kips. This will not be displayed if a plastic section is being used under LFD/LRFD.

PROGRAM FLAGS

Flag 05	clear	Have a live load moment
	set	Solve for live load moment capacity
Flag 06	clear	WSD/ASD
	set	LFD/LRFD
Flag 07	clear	Modular ratio equals N
	set	Modular ratio equals 3N (AASHTO only)
Flag 08	clear	Elastic analysis only
	set	Plastic analysis permitted by user (PLG)
Flag 09	clear	Composite
	set	Non-composite (from PLG)
Flag 10	clear	AISC LRFD, or AASHTO
	set	AISC ASD
Flag 11	set	Subroutine return
Flag 18	clear	Elastic analysis only
	set	Plastic analysis permitted by code (PLG)

STORAGE REGISTER USE

Register	COMP	BM
00	Beam Depth	Beam depth
01	Beam Area; Allowable LL Moment	Beam area
02	Beam I (moment of inertia)	Beam I
03	Compr. Flange Width; Beam S_{bf}; "TF" or "BF"	Comp. Fl. W.

D137

Register	COMP	BM
04	Compr. Flange Th.; S_{tf} (N = N)	Comp. Fl. Th.
05	Y (neutral axis) of bare beam	Y
06	F_y	
07	Web Depth (built-up beams)	Web Depth
08	Web Thickness (built-up beams and some plastic sections)	Web Thickness
09	Tens. Fl. Width; Composite Y (compos. neutral axis position)	Tens. Fl. W.
10	Tens. Fl. Th.; Slab Compr. Force; S_{conc} (N = N)	Tens. Fl. Th.
11	(Slab + Haunch + Beam Y); $S_{bottom\ flange}$ (N = 3N)	Beam S_{tf}
12	Effective slab thickness (reduced for inadequate number of shear connectors)	Beam Z
13	Slab thickness	
14	Effective slab area/N	Beam S_{bf}
15	H + S (Haunch or stool + Slab thickness)	
16	Fc' x (.40, .45, or .85)	
17	NCDL Moment (inch-kips stored)	
18	Fb (allowable bending stress) or OF_y, ksi	
19	$S_{top\ flange}$ (N = 3N)	
20	N (modular ratio, E_{st}/E_{conc})	
21	CDL Moment (inch-kips stored)	
22	Number of studs required/provided	
23	$S_{bottom\ flange}$ (N = N)	
24	.85 Fc'	
25	One half of shear stud capacity, kips	
26	Live load moment (inch-kips stored)	
27	Slab Area	

D134

COMPOSITE BEAM ANALYSIS

Program COMP analyzes composite concrete slab/steel beam sections. It will follow AISC (ASD or LRFD) or AASHTO (WSD or LFD) depending on the status of flag 04, and the user's response to the "WSD?" prompt. It will accept rolled beams with known section properties or built-up sections. If a built-up (welded plate) beam is used, COMP calculates and displays the section properties of the steel beam before calculating the composite properties. Only elastic section properties are calculated for WSD designs; however, under LFD/LRFD designs, the program will calculate plastic properties if desired.

COMP does not check compactness, slenderness, or strength (allowable stress) criteria; these must be checked or calculated by hand or by some other means (for instance, program PLG in this package). For plastic design, COMP is thus more useful for rolled beams, which are usually compact, while PLG is usually used for welded beams.

Code references are given throughout this documentation. For portions dealing with AISC ASD, the specification number given is for the Ninth Edition (green book), with the Eighth Edition (red book) specification in parenthesis following.This should help ease the transition to the new format followed in the Ninth Edition.

BEAM CAPACITY

COMP can output beam strength in two different ways. If the live load moment prompt LL M 'K is bypassed (or answered zero) the program calculates the composite section properties, subtracts the dead load stresses from the allowable stresses (yield stress for compact beams), and then finds the allowable live load moment which can be applied. Method of calculation of dead load stresses is addressed in the following section.

If the actual live load moment is input, on the other hand, the program calculates the stresses due to NCDL (non-composite dead load), CDL (composite dead load) and LL (live load), and checks the sum of these (on the bottom flange, top flange, and slab--see following paragraphs) against the allowable stress input. The former mode is useful for rating of existing structures and preliminary rough-out work; the latter mode is intended for detailed final design of beams.

Elastic Design: AASHTO/AISC

AASHTO and AISC treat live load capacity somewhat differently under WSD rules. For design by AISC ASD, the program considers the entire composite section under the combination of dead and live loads, and CDL stress is figured on the same section as the LL stress. The output stresses for NCDL are those on the bare beam, but the stresses for combined dead and live load (BF and _ TF STRS) are calculated with the full composite section acting. Thus, the individual stresses on the beam due to NCDL, CDL, and LL will not add up to the summation given. Although this seems rather awkward at first, it is useful because it gives both the NCDL stress on the bare beam (to help the designer check the beam adequacy under NCDL) and the stress produced by all loads, acting on the composite section, for checking the stresses under full load (or checking the remaining live load capacity). COMP does not check Eighth Edition AISC equation 1.11-2; if the ratio of live to dead load is too high, the results will be incorrect and unconservative. The user must verify that this ratio is within the limits prescribed. The Ninth Edition ASD specification has dropped this rule and requires only that the superimposed stresses be less than .9 F_y. The user must also check this, by adding up the output stresses.

For design by AASHTO, the procedure is a little more logical, if somewhat more involved. NCDL moment is applied to the bare beam. CDL moment is applied to a composite section figured with a modular ratio of 3N for the top and bottom flanges, and N for the slab (per AASHTO 10.38.1.4, to account for creep). The live load moment is applied to, or figured from, the composite section calculated using a modular ratio of N. The sum of the stresses on any part (top flange, bottom flange, or slab) is calculated directly by adding these. Thus, design by AASHTO involves calculating two sets of composite properties; for AISC, only one set is needed. For an AASHTO design, the further refinement of considering two different slabs, one including a topping thickness (for live load, N = N) and one not including this extra thickness (for CDL, N = 3N), was not instrumented. This was left out both for simplicity, and because it's usually slightly conservative to consider only the final slab dimensions. Since contractors are usually left the option of constructing a slab as a single pour, it would be slightly unconservative to consider two different sets of slab dimensions.

For LFD/LRFD, the program also uses elastic properties if the PLAST? prompt was answered 0 (no) or bypassed (i.e. for non-compact beams). The same methodology is used as described above with one exception: under AISC LRFD rules, the summation of stresses is figured from the superposition of elastic stresses. That is, the dead load (NCDL) is taken on the bare beam instead of being combined with live load and composite dead load on the composite section. An undercapacity (phi) factor of 0.90 is applied to the resulting LL capacity if the LL M input prompt was bypassed. (Actually, the factor of 0.90 is applied by the program only to the allowable steel stress; if the concrete strength governs, the results will be incorrect by this amount. This happens only rarely.)

For AASHTO LFD analysis for non-compact beams, the methodology described above is followed exactly; the strength is taken as the allowable steel stress (usually close to yield) or the concrete strength (0.85 f_c') times the appropriate elastic section modulus. No reduction (phi) factor is applied.

Plastic Design: AASHTO/AISC

If the "PLAST?" prompt is answered 1 (yes), COMP figures the strength based on a fully plastic stress distribution over the composite section. All loads are taken on this composite section. AISC LRFD uses an undercapacity (phi) factor of 0.85; AASHTO uses 1.0. As mentioned, COMP leaves it up to the user to determine that the section meets all compactness requirements. (When portions of COMP are called as subroutines by program PLG, the calling program determines this.)

For built-up beams, the program also calculates the plastic section modulus, BM Z, of the bare beam, for possible use by the designer. However, program COMP makes no use of this value (again, PLG does). Also, the program assumes that the plastic neutral axis falls within the web. For very unbalanced beams where one flange contains over half the area, this assumption is invalid, and COMP will not output a BM Z value.

For those following the examples in the AISC LRFD manual, the refinement of using the beam's fillet area (e.g. as shown on page 4-4) is not included in this program. However, for the condition of the plastic neutral axis (P.N.A.) falling within the web or flange of a rolled beam, the program will prompt for the flange and web dimensions in order to calculate the P.N.A. position. For built-up plate beams, this is not done since the program already has these dimensions.

INPUT SEQUENCE

The program is written to check several different beams in succession with a given slab configuration. To enter a built-up (welded) beam, answer the web and flange dimension prompts. To enter a rolled beam, bypass the WEB D" prompt (or enter zero) and the program will prompt for the tabulated elastic section properties. COMP is not able to directly figure the properties of riveted beams built up with angle (L) shapes in the flanges, or rolled beams with coverplates; for these beams, the designer must calculate the section properties by hand and enter them as for a rolled beam. (The program's neutral axis prompt, Y", is expressly for this purpose. For symmetric rolled beams, it can be bypassed.)

COMP prompts for slab thickness, span length, and beam spacing. Then it calculates the effective slab width (for an <u>interior</u> beam) from these parameters and displays it, using the minimum of:

Span/4
Beam spacing
12 x Slab thickness (AASHTO only)

Note that the Eighth Edition AISC ASD criteria (Flange width plus 16 x Slab thickness) is not used, at the recommendation of LRFD Commentary section I3.1 and the Ninth Edition ASD.

If the effective width is correct, press R/S to continue; if not (for instance, for an edge beam) key in the correct value and press R/S. For a given beam, if the calculated neutral axis is below the bottom of the slab, the entire slab is considered effective in compression; if the neutral axis is within the slab, only that portion above the axis is considered. The number of shear studs used also can affect the neutral axis position; see the following paragraphs.

If the neutral axis is above the top flange of the beam (i.e. in the slab or haunch), the composite section modulus of the top flange calculates out as a negative number, and the stress on the top flange due to live load is subtracted from the dead load stress. Slab reinforcing is not taken into account, nor is any concrete in the haunch between the beam and slab considered (although the separation between beam and slab due to the haunch is taken into account). Thus, transverse formed metal deck may be accomodated by entering the height of the ribs for the H" prompt. The slab thickness to be input would then be that amount above the ribs. Formed metal deck with the ribs running <u>along</u> the beam is a little more difficult. The simplest procedure is probably to ignore the concrete in the ribs and treat them as a haunch, the same as for transverse ribs. This is conservative.

If the beam is slightly embedded in the slab (i.e. a negative haunch), the "negative area" of concrete in the haunch is still neglected. That is, the program

DWR

considers there to be concrete and a beam flange in the same place. This, of course, is slightly unconservative if the neutral axis falls beneath the top of the top flange. For embedded beams with the N.A. in the beam, the user should reduce the input slab thickness to obtain zero embedment.

SHEAR CONNECTORS

COMP also calculates shear connector requirements using ultimate strengths of the slab and beam. If designing by AASHTO or AISC LRFD, the stud diameter is input, and the capacity and required number of studs is calculated by AASHTO equation 10-66 or LRFD equation I5-1. Research has shown that concrete strengths above 4000 psi do not increase connector strength. However, the program does not take this into account. If using concrete stronger than 4000 psi, see the next paragraph.

If designing by AISC ASD, the shear connector capacity is input directly from Table I4.1 (1.11.4) at the STUD CAP (connector capacity) prompt. When designing by LRFD or AASHTO,if using something other than standard shear studs at full strength (e.g. channel connectors, or studs at reduced strength due to formed metal deck ribs); or if the input slab concrete strength is greater than 4000 psi; bypass the STUD D" (diameter) prompt in order to get the STUD CAP prompt and key in a strength value directly. (That is, the program does not calculate a reduction per LRFD I3-1.)

The program does not check AASHTO fatigue requirements; however, the statical inertia Q is displayed if flag 04 is clear. For partial connection, this value of Q is still figured on the full slab area if the slab governs. If flag 00 is set, this shear output is suppressed.

The requirements of AISC ASD equation I4-5 (1.11-7) regarding moments due to concentrated loads are not checked, nor are those of AASHTO 10.38.5.1.3 regarding negative moment regions. The designer must check vertical shear on the steel beam web; COMP does not do this.

Partial shear connection is handled differently depending on which code is being followed. If designing by AISC LRFD or AASHTO, the program simply reduces the effective slab area by the ratio of actual to required shear connectors. This is done by reducing the effective thickness of the slab rather than the width; the haunch dimension is increased accordingly. The difference between the two methods is small.

If designing by AISC ASD, formula I4-4 (1.11-1) is used, interpolating the strength between that of the fully composite section and the bare beam. However, the slab used to figure the "fully composite section" is first reduced as explained in the preceding paragraph. This minor error will cause a small discrepancy when checking hand calculations. The error is conservative and typically is very small, on the order of two percent.

ALLOWABLE STRESSES

As with the other structural programs in this module, the concrete strength prompted for, Fc'KSI, is the ultimate strength, whether designing by WSD or LFD. The steel stress Fb KSI is the working stress to be used; for load factor designs it will be close to or equal to the yield stress. F-Y KSI (yield) is used only for the shear connector design and plastic design.

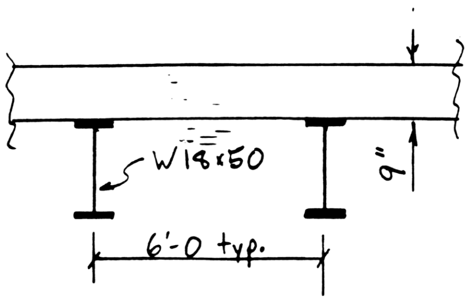
OTHER CONSIDERATIONS

Shored construction can be analyzed by simply considering the dead loads (NCDL and CDL) as part of the live load, and inputting them as zero.

COMP does not check span/depth ratios per AASHTO 10.5; the designer should do this. It also will not handle negative-moment areas (i.e. a beam acting compositely with slab reinforcing).

When using program PLG to analyze a welded composite beam, the flange width to thickness requirement is assumed to be met. The AASHTO requirements for $b/2t_f$ related to dead load stress (10.34.2.1.5 and 10.50c) are not checked in this situation by either program COMP or PLG and should be reviewed by the designer. AASHTO 10.50(d), related to dead load moment, also must be checked by the designer. Program PLG calculates the constant R_b for this purpose.

COMP does not check vertical shear on the web. Program PLG may be used for this. Neither program checks AASHTO 10.50(e), concerning tension field action in the web prior to hardening of the slab.



$F_y = 36$ ksi ; $f_s = 24$ ksi

$f_c' = 4.0$ ksi ; $n = 8$

Solve for Live Load capacity by AISC, Span = 26'-0.

W18×50 : $d = 17.99"$
$A = 14.7\,\square"$
$I = 800\ in^4$

Effective slab width :

Span/4 = 26'/4 = 78"
Beam Spa. = 72" ← Governs
Slab 16×9" + 7½" = 151.5"

DL : 6'×.75'×.15 = .675 k/'
Beam .05
.725

DL M = $.725^{k/'} \times \frac{(26')^2}{8} = 61.26'^{k}$

Check erection stress :

$(61.26 \times 12) \cdot \left(\frac{17.99"}{2}\right) / 800_{in^4} = 8.27$ ksi OK

Composite Section Properties

		A	d	Ad
Slab	72"×9"/n=8 =	81.00	4.5"	364.5
Beam	9" + 17.99/2	14.7	17.995	264.53
		95.7	6.57"	629.03

Trial N.A. = 6.57" < 9" ⇒ Cracked section

$\frac{72"}{n=8} \times Y \cdot (\tfrac{1}{2} Y) = 14.7\square" \cdot \left(\frac{17.99}{2} + 9" - Y\right)$

$4.5 Y^2 = -14.7 Y + 264.53$

$Y = \frac{-14.7 + \sqrt{14.7^2 - 4 \times 4.5 \times (264.53)}}{2 \times 4.5}$

$= 6.2058"$

Re-Calc. Composite Section Properties

	A	d	Ad^2	I_o
Slab	55.85	3.10	537.73	179.25
Beam	14.7	11.79"	2043.10	800.0
			2580.83	979.25
			3560.08 in^4	

$S_{slab} = \frac{3560.08}{6.2058} \times (n=8) = 4589.38\ in^3$

$S_{T\cdot Fl.} = \frac{3560.08}{6.2058 - 9"} = -1274.08\ in^3$

(minus sign indicates top fl. in tension)

$S_{B\cdot Fl.} = \frac{3560.08}{17.99" + 9" - 6.2058"} = 171.29\ in^3$

Live Load Capacity

$\left(\frac{24\ ksi \times 171.29}{12}\right) - 61.26'^{k}$

$= 281.31'^{k}$

D145

Design Shear Studs

Slab: $(.85 \times 4\,ksi)\ \frac{72''}{4} \times 9''/2 = 137.70\ k$

Beam: $14.7\,\square'' \times 36\,ksi/2 = 264.60\ k$ ← Governs

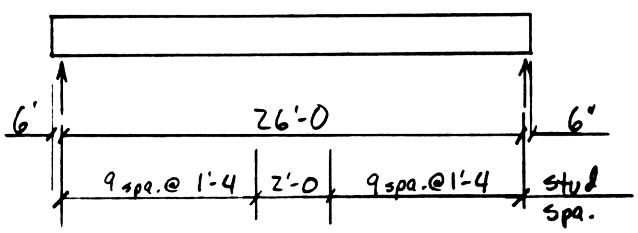
Use 3/4" φ studs; capacity = 13.3 k/stud (Table 1.11.4)

Min. # studs = 19.89 each side of ℄

USE TWO ROWS OF 20 - 3/4" φ STUDS

Check diameter: flange th. = .57"

$.57'' \times 2\frac{1}{2} = 1.425'' < 3/4''\,\phi$ OK



Check w/ 2 rows of 15 studs

$$S_{eff} = 48.94\,in^3 + \sqrt{\frac{15}{19.89}}\,(171.29 - 48.94)$$

$$= 160.45\ in^3$$

Live Load: $160.45 \times 24\,ksi/12 = 320.90$
$-\ 61.26$
$259.64'^{-k}$

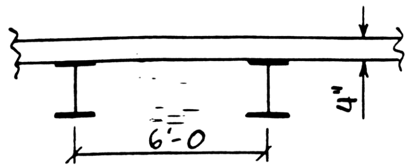
```
            XEQ "COMP"
 8:55:58 AM  05/16/89

 TITLE:
                     RUN
 ************************
 AISC
 WSD?
                     RUN
 N
          8.0000     RUN
 Fc'KSI
          4.0000     RUN
 STUD CAP, K
         13.3000     RUN
 F-Y KSI
         36.0000     RUN
 Fb KSI
         24.0000     RUN
 ---------
 SL TH"
          9.0000     RUN
 BM SPA'
          6.0000     RUN
 SPAN'
         26.0000     RUN
 SL Be=72.0000"
                     RUN
 H"
          0.0000     RUN
 ---------
 NCDL M,'K
         61.2600     RUN
 CDL M,'K
          0.0000     RUN
 LL M,'K
                     RUN
 ************************
 WEB D"
                     RUN
 BM A SQ"
         14.7000     RUN
 BM D"
         17.9900     RUN
 Y"
                     RUN
 I
        800.0000     RUN
 HOR V=264.6000 K
 #STUDS REQD=39.7895

 #STUDS
                     RUN
 ---------
```

```
 N=8

 I-COMP=3,560.0754
 Y=6.2058"
 Sc=4,589.3792
 TF S=-1,274.0782
 BF S=171.2873
 ------------------------
 B FL
 NCDL STRS=8.2655 KSI
 ---------
 T FL
 NCDL STRS=8.2655 KSI
 ---------
 ---------
 WS LL M=281.3146 'K
 ---------
 ************************
 WEB D"
                     RUN
 BM A SQ"
         14.7000     RUN
 BM D"
         17.9900     RUN
 Y"
                     RUN
 I
        800.0000     RUN
 HOR V=264.6000 K
 #STUDS REQD=39.7895

 #STUDS
         30.0000     RUN
 ---------
 N=8

 I-COMP=3,560.0754
 Y=6.2058"
 Sc=4,589.3792
 TF S=-1,274.0782
 BF S=171.2873

 EFF S=160.4430
 ------------------------
 B FL
 NCDL STRS=8.2655 KSI
 ---------
 T FL
 NCDL STRS=8.2655 KSI
 ---------
 ---------
 WS LL M=259.6261 'K
 ---------
 ************************
 WEB D"
```

Use less than full shear connection

Effective S for reduced number of studs

D148



Span = 30'-0
Bm Spa. = 6'-0
Slab Th. = 4"

f'_c = 3.0 ksi
n = 9
f_y = 36 ksi
f_{st} = 24 ksi

Live Load = 100 psf
Partition + Ceiling Load = ~30 psf

Loads: Slab $.33 \times 6' \times .15 = .30$ k/'
Bm say .04 k/'
Σ NCDL .34 k/'

NCDL M = $.34\text{ k/'} \times \frac{30^2}{8}$ = 38.25'-k

CDL M = $(.03\text{ ksf} \times 6') \times \frac{30^2}{8}$ = 20.25'-k

LL M = $(.10 \times 6') \times \frac{30^2}{8}$ = 67.50'-k

Shear = $(.34\text{ k/'} + (.03 + .10) \times 6') \times 15'$
= 16.40 k

Effective Slab Width

Span/4 = 30'×12/4 = 90"
Bm Spa. = 6'×12 = 72"
16 t_{sl} + Flange = 16×4" + 6" (say) = 70" ←

b_{eff} = 70"; effective Area = $\frac{70"}{n=9} \times 4"$
= 31.11 □"

TRY W 16×31

A = 9.12 □"
d = 15.88"
I_x = 375 in⁴
S_x = 47.2 in³

t_w = .275"
b_f = 5.53" ≈ 6" OK

Check Shear

$\frac{16.40\text{ k}}{.275" \times 15.88"}$ = 3.85 ksi < .40 f_y OK

Note: COMP does not check shear.

Check DL on bare beam

$\frac{38.25'^{k} \times 12}{47.2\text{ in}^3}$ = 9.72 ksi < 24 ksi OK

Calculate Composite Properties

	A	d	Ad	$\bar{d}$	$A\bar{d}^2$	I_o
Slab	31.11	2"	62.22	2.25"	157.96	41.48
Beam	9.12	11.94"	108.89	7.69"	538.86	375
	40.23		171.11		696.82	416.48
						1113.30

$\frac{171.11}{40.23}$ = 4.25" ⇒ N.A. below slab

I_{comp} = 1113.30 in⁴

$S_{conc} = \frac{1113.30}{4.25} \cdot (n=9)$ = 2355.75 in³

$S_{TF} = \frac{1113.30}{4.25" - 4"}$ = 4395.18 in³

$S_{BF} = \frac{1113.30}{15.88 + 4" - 4.25"}$ = 71.24 in³

Σ M = 38.25 NCDL
20.25 CDL
67.50 LL
126.00'-k

Slab: $\frac{(20.25 + 67.50) \times 12}{2355.75}$ = .447 ksi < .45 f'_c OK

Bottom Fl: $\frac{126.0'^{k} \times 12}{71.24}$ = 21.22 ksi < 24 OK

Top Fl: OK by inspection

Check: $S_{max} = (1.35 + .35 \frac{(67.50 + 20.25)}{38.25}) \times 47.2"$
= 101.62 > 71.24 OK

Design Shear Connectors

Use 3/4" × 3" Studs (2 lines)

Max stud $\phi = 2.5 \times t_{fl} = 2.5 \times .440"$

$= 1.1"$ OK

Horiz. Shear (do not use part. comp. action)

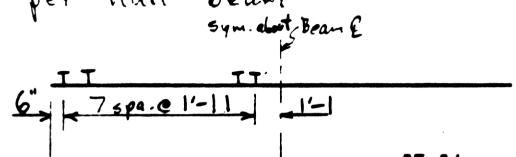
$(.85 \times 3.0 ksi) \times 70" \times 4"/2 = 357.0 k$

$36 ksi \times 9.120" / 2 = 164.16 k$ ←

3/4" × 3" studs ⟶ 11.5 kips/stud

$\frac{164.16 k}{11.5} = 14.27$ studs per half

∴ Use two rows of 8 studs per half beam

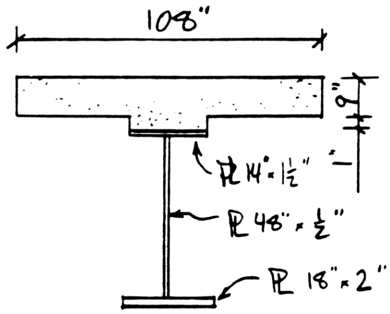



```
                SF 04
           XEQ "COMP"
  8:41:50 AM  05/16/89

TITLE:
OFFICE BLDG FLOOR DESIGN
                     RUN
************************
AISC
WSD?
          1.0000     RUN
N
          9.0000     RUN
Fc'KSI
          3.0000     RUN
STUD CAP, K
         11.5000     RUN
F-Y KSI
         36.0000     RUN
Fb KSI
         24.0000     RUN
---------
SL TH"
          4.0000     RUN
BM SPA'
          6.0000     RUN
SPAN'
         30.0000     RUN
SL Be=72.0000"
         70.0000     RUN
H"
          0.0000     RUN
---------
NCDL M,'K                   Non-composite DL
         38.2500     RUN
CDL M,'K                    Composite DL
         20.2500     RUN
LL M,'K                     Live load
         67.5000     RUN
************************
WEB D"
                     RUN    by pass for rolled beam
BM A SQ"
          9.1200     RUN
BM D"
         15.8800     RUN
Y"
                     RUN    can bypass for symm. beam
I
        375.0000     RUN
HOR V=164.1600 K
#STUDS REQD=28.5496

#STUDS
                     RUN
---------
N=9

I-COMP=1,113.3023
Y=4.2533"
Sc=2,355.7516
TF S=4,395.1765
BF S=71.2436
------------------------
B FL
NCDL STRS=9.7186 KSI        NCDL stress on bare beam
CDL STRS=3.4108 KSI
LL STRS=11.3694 KSI

Σ B FL STRS=21.2230 KSI     Σ on composite section
---------
T FL
NCDL STRS=9.7186 KSI
CDL STRS=0.0553 KSI
LL STRS=0.1843 KSI

Σ T FL STRS=0.3440 KSI
---------
SLAB
CDL STRS=0.1032 KSI
LL STRS=0.3438 KSI

Σ SLAB STRS=0.4470 KSI
---------
************************
WEB D"
```

D150

Calculate plastic section properties of this beam. Find allowable load by AASHTO and AISC LRFD.

$F_y = 36$ ksi
$f_c' = 4.0$ ksi



Check Compactness - $D/t_w = \frac{48}{.5} = 96.0$

If by AASHTO, $\frac{D}{t_w} < \frac{19{,}230}{\sqrt{F_y}} = 101.35$ - **OK**

If by AISC, $\frac{H_c}{t_w} < \frac{640}{\sqrt{F_y}} = 106.67$ - **OK**

Composite section is considered compact by either code. Non-composite section would require additional compression flange checks.

Note: Program COMP does not check compactness; the user must do this. Program PLG does check compactness.

Calculate Z of bare beam:

$14 \times 1.5 = 21.0$
$48 \times 0.5 = 24.0$
$18 \times 2.0 = 36.0$
Area = 81.0 □"

$\frac{81.0}{2} - 21.0\square" = 19.5\square"$

$\frac{19.5}{0.5} = 39.0", + 1½" = 40.5"$ to N.A.

Bare Beam (cont'd)

Top half C.G. : $21.0 \times .75" = 15.75$
$19.5 \times 21.0" = 409.50$
$40.5 \quad 10.5" \quad 425.25$

Bottom half C.G. $36.0 \times 1.0" = 36.0$
$4.5\square" \times 6.5" = 29.25$
$40.5 \quad 1.611" \quad 65.25$

$Z = 40.5\square" \times (51.5" - 1.611" - 10.5")$

$= 1595.25 \text{ in}^3$

Composite Beam Z:

$C = 108" \times 9" \times (.85 \times 4.0 \text{ ksi}) = 3304.8k$

$T = 81.0\square" \times 36 \text{ ksi} = 2916.0k$

∴ N.A. is in slab

$a = \frac{2916.0}{3304.8} \times 9" = 7.9412"$

Slab $\frac{1}{2} \times (7.9412^2) \times 108" \times \frac{(.85 \times 4)}{36 \text{ ksi}} = 321.62$

Top Fl. $21\square" \times (1.0588" + 1" + 3/4") = 58.99$

Web $24\square" \times (1.0588 + 1" + 1½" + 24") = 661.41$

Bott. Fl. $36\square" \times (1.0588 + 50.5" + 1") = 1892.12$

$Z = 2934.13 \text{ in}^3$

AASHTO Capacity $= F_y Z = 36.0 \times 2934.13/12$
$= 8802.40'^{k}$

Capacity by AISC LRFD $= \phi F_y Z$
$= .85 \times 8802.40 = 7482.04'^{k}$

D51

2/4

SF 04

XEQ "COMP"
8:12:15 AM 05/16/89

TITLE:
PLASTIC SECTION TRIALS
RUN
****
AISC
WSD?
0.0000 RUN
PLAST?
1.0000 RUN
N
8.0000 RUN
Fc'KSI
4.0000 RUN
STUD D"
1.0000 RUN
F-Y KSI
36.0000 RUN

SL TH"
9.0000 RUN
BM SPA'
9.0000 RUN
SPAN'
100.0000 RUN
SL Be=108.0000"
RUN
H"
1.0000 RUN

NCDL M,'K
1,610.0000 RUN
CDL M,'K
800.0000 RUN
LL M,'K
RUN
****
WEB D"
48.0000 RUN
WEB TH"
.5000 RUN

COM FL W"
14.0000 RUN
TH"
1.5000 RUN
TEN FL W"
18.0000 RUN
TH"
2.0000 RUN
I=38,202.6875
Y=30.1944"
COM S=1,265.2224
TEN S=1,793.0857

BM Z=1,595.2500

AREA=81.0000 SQ"

HOR V=2,916.0000 K
#STUDS REQD=123.7154

#STUDS
RUN

Z=2,934.1324
PLAST M=7,482.0375 'K
LL M=5,072.0375 'K

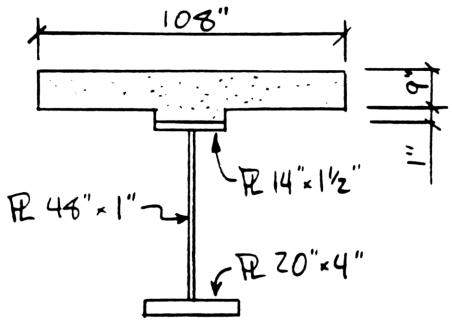
N=8

I-COMP=100,943.7500
Y=18.7778"
Sc=43,005.6213
TF S=11,499.9209
BF S=2,362.7926
****

D152

Calculate composite plastic section properties of this beam, and capacities by AASHTO and AISC LRFD.

$F_y = 36$ ksi
$f_c' = 4.0$ ksi



Bm Area : 14 × 1½ = 21.0
48 × 1 = 48.0
20 × 4 = 80.0
149.0 □"

<u>Note:</u> COMP will not figure a BM Z ~~for~~ this beam since one flange contains over half the beam area.

$C = 108" \times 9" \times (.85 \times 4.0) = 3304.8$

$T = 149.0\,□" \times 36 = 5634.0$

$C' = (5634.0 - 3304.8)/2 = 1029.6$ k

$\frac{1029.6}{36\,ksi} = 28.6\,□" > 21.0\,□"$

⇒ N.A. in Web

$\frac{28.6 - 21.0}{1"} = 7.60"$

Slab	$\left(\frac{3304.8}{36\,ksi}\right)$	× (7.6" + 1.5 + 1 + 4.5")	=	1340.28
Top Fl.	21.0	× (7.60" + .75")	=	175.35
Web	7.60" × 1"	× (7.60/2)	=	28.88
Web	40.4 × 1"	× (40.4/2)	=	816.08
B.Fl.	80 □"	× (40.4 + 2")	=	3392.00
		Comp. Z	=	5752.59 in³

AASHTO Capacity = $F_y Z$

= 36.0 × 5752.59/12

= **17,257.8 '·k**

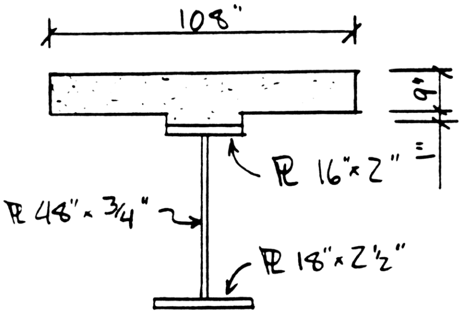
AISC Capacity = $\phi F_y Z$

= .85 × 17,257.8 = **14,669.1 '·k**

```
WEB D"
          48.0000    RUN
WEB TH"
           1.0000    RUN
COM FL W"
          14.0000    RUN
TH"
           1.5000    RUN
TEN FL W"
          20.0000    RUN
TH"
           4.0000    RUN
I=59,932.2955
Y=35.9715"
COM S=1,666.1061
TEN S=3,419.1297

AREA=149.0000 SQ"
---------
HOR V=3,304.8000 K
#STUDS REQD=140.2108

#STUDS
                     RUN
---------
Z=5,752.5900
PLAST M=14,669.1045 'K
LL M=12,259.1045 'K
---------
N=8

I-COMP=175,857.4426
Y=27.3438"
Sc=51,450.7545
TF S=10,139.4945
BF S=4,863.8264
************************
WEB D"
```

Calculate composite plastic section properties of this beam, and capacities by AISC LRFD and AASHTO.

$F_y = 36$ ksi
$f'_c = 4.0$ ksi



$C = 9" \times 108" \times (.85 \times 4) = 3304.8$

Bm Area = $16 \times 2 = 32.0$
$48 \times 3/4 = 36.0$
$18 \times 2\frac{1}{2} = 45.0$
113.0 □"

$T = 113.0 \times 36$ ksi $= 4068.0$ k

$T > C \Rightarrow$ N.A. in beam

$$C' = \frac{4068 - 3304.8}{2} = 381.6 \text{ k}$$

$$\frac{381.6 \text{ k}}{36 \text{ ksi} \times 16"} = .6625" < 2"$$

∴ N.A. is in flange

Slab	$\frac{3304.8 \text{ k}}{36 \text{ ksi}}$	= 91.8□"	× 6.1625"	= 565.72 in³
Fl.	.6625 × 16	= 10.6□"	× .3312"	= 3.51
Fl.	1.3375 × 16	= 21.4□"	× .6688"	= 14.31
Web	48 × 3/4	= 36.0□"	× 25.3375	= 912.15
B.Fl.	18 × 2½	= 45.0□"	× 50.5875	= 2276.44

Composite Z = 3772.13

By AASHTO, capacity = $F_y Z$

$= 36 \times 3772.13/12$

$= \underline{11,316.4'^{k}}$

By AISC LRFD, capacity = $\phi F_y Z$

$= .85 \times 11,316.4 = \underline{9618.9'^{k}}$

```
WEB D"
        48.0000    RUN
WEB TH"
          .7500    RUN
COM FL W"
        16.0000    RUN
TH"
         2.0000    RUN
TEN FL W"
        18.0000    RUN
TH"
         2.5000    RUN
I=54,635.8497
Y=28.9757"
COM S=1,885.5772
TEN S=2,322.5246

BM Z=2,311.9167

AREA=113.0000 SQ"
---------
HOR V=3,304.8000 K
#STUDS REQD=140.2108

#STUDS
                   RUN
---------
Z=3,772.1275
PLAST M=9,618.9251 'K
LL M=7,208.9251 'K
---------
N=8

I-COMP=125,044.4220
Y=21.1130"
Sc=47,381.0009
TF S=11,252.0787
BF S=3,021.3459
************************
```

(D134)

Program COMP For Composite Beam Analysis 1487 bytes Page 1 of 8

Step	X	Y	Z	T	L
01♦LBL "COMP"					
02 CF 09					
03 CF 11					
04 27					
05 XEQ 31	F_y				
06 "Fb KSI"					
07 FC? 18					
08 PROMPT					
09 .9	φ				
10 FS? 04					
11 FC? 06					
12 SIGN	1.0				
13 *					
14 STO 18	F_b				
15♦LBL D					
16 XROM "L"					
17 "SL TH""					
18 PROMPT	Sl.Th.				
19 STO 13					
20 "BM SPA·"					
21 PROMPT	spa.				
22 "SPAN·"	span				
23 PROMPT					
24 4					
25 /	span/4	spa.			
26 X>Y?					
27 X<>Y	min				
28 ENTER↑	min	min			
29 FC? 04					
30 RCL 13	sl.th.				
31 X>Y?					
32 X<>Y	width,ft				
33 12					
34 *	width,in.				
35 "SL Be="					
36 ARCL X					
37 XEQ "AI"					
38 PROMPT					
39 RCL 13	sl.th.				
40 *					
41 STO 27	sl.area				
42 CLX					
43 "H""					
44 PROMPT	stool				
45 RCL 13	sl.th.				
46 +					
47 STO 15	h+sl.				
48♦LBL E					
49 XROM "L"					
50 "N"					
51 XEQ 13					
52 STO 17	NCDL				
53 CLA					

Step	X	Y	Z	T	L
54 XEQ 13	CDL				
55 STO 21					
56 CF 05					
57 "LL "					
58 XEQ 14					
59 X=0?					
60 SF 05					
61 STO 26	LLM				
62 FS? 11					
63 RTN					
64♦LBL B					
65 XROM "L*"					
66 CF 07					
67 XEQ "BM"					
68 X≠0?					
69 GTO 39					
70 "BM A"					
71 XROM "IS"					
72 PROMPT					
73 STO 01	BmA				
74 "BM D""					
75 PROMPT					
76 STO 00	d				
77 2					
78 /	$\bar{Y}$				
79 "Y""					
80 PROMPT					
81 STO 05	$\bar{Y}$				
82 "I"					
83 PROMPT					
84 STO 02	I				
85♦LBL 39					
86 RCL 24	$.85f'_c$				
87 RCL 27	sl. area				
88 *					
89 STO 10	sl. force				
90 LASTX	area				
91 RCL 20	n				
92 /					
93 STO 14	a/n				
94 X<>Y	force				
95 RCL 06	F_y				
96 RCL 01	A_{bm}				
97 *	Tens.	Compr.			
98 X>Y?					
99 X<>Y	force				
100 2					
101 FC? 10					AISC ASD
102 SIGN					uses H =
103 /	Hor. Shear				$A \cdot F_y / 2$
104 "HOR V="					
105 ARCL X					
106 "⊢ K"					
107 FC? 00					
108 XEQ "PY"					

D155

Step	X	Y	Z	T	L / Notes
109 RCL 25	Stud cap				
110 X≠0?					
111 /					
112 "#STUDS REQD"	#Studs				
113 X≠0?					
114 XEQ 02					
115 ADV					
116 ENTER↑	#studs	#studs			
117 ASTO 23					"#STUDS"
118 CLA					
119 ARCL 23					
120 X≠0?					
121 PROMPT	#avail	#req'd			
122 X≠0?					
123 /	Req/avail				
124 SIGN	1				
125 LASTX	Req/avail	1			If more studs are furnished than are req'd, consider only the number req'd.
126 X<Y?					
127 SIGN	#req/avail				
128 STO 22					
129 ST/ 10	*				
130 ST/ 14	*				
131 RCL 13	sl. th.				
132 X<>Y					
133 /					*Reduces slab area and compr. force if less studs are furnished than are required.
134 STO 12	eff. sl. th.				
135 "a="					
136 FC? 10					
137 X=Y?					
138 X=0?					
139 XEQ 23					
140 XROM "L"					
141 FC? 18					
142 GTO 29					
143 RCL 15	sl+h				
144 RCL 01	Bm A				
145 RCL 10	sl. compr.				
146 RCL 06	Fy				
147 /					
148 STO 10	*	Bm A			*Slab area transformed to equiv. steel area
149 X<Y?					
150 GTO 01					
151 /	Bm A / Sl. A	sl+h	sl+h	sl+h	
152 RCL 12	sl. th.				Neutral axis is in flange
153 *	eff. sl. th.				
154 2					
155 /					
156 -	arm				
157 RCL 05	Bm ȳ				
158 +	arm				
159 RCL 01	Bm A				
160 *	Mom cap				
161 GTO 09					
162♦LBL 13					
163 "⊢CDL "					

Step	X	Y	Z	T	L / Notes
164♦LBL 14					
165 CLX					
166 XROM "M"					
167 PROMPT					
168 *					
169 RTN					
170♦LBL 01	Sl. A	Bm A			Neutral axis is in beam
171 RCL 04	T. Fl. Th.?				
172 X>0?					If there is a value for flange thickness, GTO 01. Otherwise (i.e. rolled beam) prompt for it.
173 GTO 01					
174 XEQ 11					
175 STO 03	T. Fl. W.				
176 PROMPT					
177 STO 04	T. Fl. Th.				
178♦LBL 01	T. F. Th.				
179 RCL 03	T. F. W.				
180 *	Fl. Area				
181 RCL 01	Bm A				
182 RCL 10	Sl. A				
183 -	ΔA				
184 2					
185 /	ΔA/2	Fl. A.			
186 X<Y?					
187 GTO 01					
188 -	Δ				Neutral axis in web
189 STO Y	Δ	Δ			
190 RCL 08	Web th.				
191 /	N.A.				
192 RCL 05	Bm N.A.				
193 RCL 04	T. Fl Th.	Bm NA	Pl. N.A.	Δ	
194 -					
195 ST+ X					
196 +	Ze"	Δ			
197 *	3rd term				
198 CHS					
199 RCL 05	Bm N.A.				
200 ST+ X					
201 RCL 04	T. F. Th.				
202 -	Ze'				
203 RCL 03	Fl. W.				
204 LASTX	F. Th.	Fl. W.	Ze'	3rd term	
205 *	A Fl.	Ze'	3rd term		
206 *	2nd term	3rd term			
207 +	Steel term				
208 GTO 03					
209♦LBL 01					Neutral axis in flange
210 STO Y	ΔA/2	ΔA/2			
211 RCL 03	Fl. W.				
212 /					
213 CHS					
214 RCL 05	Bm. N.A.				
215 ST+ X					
216 +	Ze'	ΔA/2			
217 *	Steel term				

Program COMP For Composite Beam Analysis ______ bytes Page 3 of 8

Step	X	Y	Z	T	L
218♦LBL 03	Steel term				
219 RCL 15	sl th				
220 RCL 12	eff. sl. th.				
221 2	2	th.	sl th	steel term	
222 /					
223 -	e_c				
224 RCL 05	$\bar{Y}$				
225 +	e				
226 RCL 10	sl. force				
227 *	conc. term	steel term			
228 +	Z				
229♦LBL 09	Z				
230 "Z"					
231 XEQ 02					
232 RCL 06	F_y				
233 *	M				
234 .85	φ				
235 FC? 04					
236 SIGN					
237 *	φM				
238 "PLAST"					
239 XROM "M="					
240 XEQ "PY"					
241 LASTX	12				
242 *	φM"-k				
243 RCL 21	NCDL				
244 -					
245 RCL 17	CDL				
246 -	LLM				
247 CLA					
248 XEQ 05					
249♦LBL 29					
250 RCL 15	sl th				
251 RCL 05	$\bar{Y}$				
252 +					
253 STO 11	sl th $\bar{Y}$				
254 RCL 01	Bm A				
255 *	A d_{Bm}				
256 RCL 14	sl A/n				
257 RCL 12	eff sl th				
258 2					
259 /	$d_{sl.}$	$A_{sl.}$	A·d_{Bm}		
260 *	A·d_{sl}				
261 +	ΣAd				
262 RCL 01	Bm A				
263 RCL 14	sl. A/n				
264 +	ΣA				
265 /	trial $\bar{Y}$				
266 STO 09	$\bar{Y}$				
267 RCL 12	eff sl. th.				
268 X<=Y?					
269 GTO 06					
270 RCL 11	sl th $\bar{Y}$				Neutral Axis in slab
271 RCL 01	Bm A				

Step	X	Y	Z	T	L
272 *	A·d_{Bm}				
273 CHS	-A·d				
274 LASTX	Bm A				
275 RCL 14	sl. A/n	Bm A	-Bm A d	sl. th.	
276 R↑					
277 ST+ X	2·sl. th	sl. A/n	Bm A	-Bm A·d	
278 /	Eff. W / 2	Bm A	-A·d		
279 XROM "Q"					
280 STO 09	$\bar{Y}$				
281♦LBL 06	*				* $\bar{Y}$ or sl. thickness (uncracked portion of slab)
282 ENTER↑	*	*			
283 X↑2	$*^2$				
284 12					
285 /	$*^2/12$	*			
286 RCL 14	sl. A/n		$\bar{Y}^2/12$	$\bar{Y}$	
287 RCL 12	sl. th.	A/n	$\bar{Y}^2/12$	$\bar{Y}$	
288 /	eff. W				
289 R↑	$\bar{Y}$	eff. w	$\bar{Y}^2/12$	$\bar{Y}$	
290 *	eff. sl. A	$Y^2/12$	*	*	
291 R↑	*				
292 2	2	*	eff. sl. A	$Y^2/12$	
293 /	*/2				
294 RCL 09	$\bar{Y}$				
295 -	d				
296 X↑2	d^2	eff. A	$Y^2/12$	$Y^2/12$	
297 R↑	$Y^2/12$	d^2	eff. A		
298 +					
299 *	$I_{sl.}$				
300 RCL 09	$\bar{Y}$				
301 RCL 11	sl. th + $\bar{Y}$				
302 -	bm d				
303 X↑2	d^2				
304 RCL 01	Bm A				
305 *	A d^2	sl. I			
306 +	Σ I				
307 RCL 02	Bm I				
308 +	I				
309 ENTER↑	I	I			
310 "N="					
311 RCL 20	n	I	I		
312 XROM "AX"					
313 FC? 00					
314 AVIEW					
315 ADV					
316 X<>Y	I	n	I		
317 "I-COMP"					
318 XEQ 02					
319 RCL 09	$\bar{Y}$	I	n	I	
320 "Y="					
321 XEQ 23					
322 /	S_c	n	I	I	
323 *	S_c·n	I	I	I	
324 "Sc"					
325 FC? 07					
326 STO 10					
327 XEQ 02					

Program COMP For Composite Beam Analysis ______ bytes

Step	X	Y	Z	T	L
328 RDN	I	I	I	S_c	
329 RCL 09	$\bar{Y}$				
330 RCL 15	sl.th				
331 -	T.Fl.d	I	I	I	
332 /	T.Fl.S				
333 STO 19					
334 FC? 07					
335 STO 04					
336 "TF"					
337 XEQ 22					
338 RDN	I				
339 RCL 00	Bm d				
340 LASTX	T.Fl. d				
341 -	B.Fl.d	I			
342 /	B.Fl.S (comp)	I	I	I	
343 STO 11					
344 "BF"					
345 XEQ 22					
346 FS? 07					
347 GTO 01					
348 STO 23					
349 RCL 02	Bm I				
350 RCL 00	Bm d				
351 RCL 05	Bm $\bar{Y}$				
352 -	B.Fl. d_{Bm}				
353 /	B.Fl.S (bare beam)				
354 STO 03					
355 FS? 04					
356 GTO 06					
357 ADV					
358 RCL 09	Compos $\bar{Y}$				
359 RCL 13	sl.th*				* Full slab thickness, regardless of studs
360 2					
361 /					
362 -	Slab d				
363 RCL 14	sl.A/n				
364 *	Q				
365 "Q"					
366 XEQ 02					
367 GTO 01					
368♦LBL 06	S_{Bm}	S_{compos}			
369 -	ΔS				
370 RCL 22	*				* #Studs req'd / # avail., or 1; if 1, then $S_{eff} = S_{compos}$
371 SQRT	√*	$S_{tr} - S_{Bm}$			
372 /					
373 RCL 03	S_{BM}				
374 +	$S_{eff.}$				
375 STO 11					
376 X<> 23	S_{compos}				
377 RCL 11	S_{eff}				
378 X<Y?					
379 ADV					
380 "EFF"					
381 X<Y?					
382 XEQ 22	$S_{eff.}$	$S_{compos.}$			

Note	Step	X	Y	Z	T	L
	383♦LBL 01					
FC? 07 and FC? 04	384 FC? 07					
	385 FS? 04					
	386 X<0?					
	387 GTO 16					
plastic?	388 FS? 18					
	389 GTO 16					
	390 XROM "LL"					
	391 RCL 17	NCDL M				
	392 RCL 03	S_{BF}				Bottom Flange
	393 "B"					
	394 RCL 11	S_{3n}*				* Both are S_{BF} for AISC
	395 RCL 23	S_n *	S_{3n}	S_{Bm}	NCDL	
	396 XEQ 20	Σf_b				
	397 RCL 18	allow F_b				
	398 XEQ 21	act. f_b	allow F_b			
	399 -	ΔF_b				
	400 RCL 23	S_n				
	401 *	LL M				
	402 STO 01	allow LL M				
	403 RCL 17	NCDL M				
	404 RCL 02	Bm I				
	405 RCL 05	Bm $\bar{Y}$				
	406 /	Bm T.Fl.S				
	407 "T"					
	408 RCL 19	S_{3n}				
	409 RCL 04	S_n	S_{3n}	S_{Bm}	NCDL M	Top Flange
	410 XEQ 20	act. f_b				
	411 RCL 18	allow F_b				
	412 XEQ 21	act f_b	F_b			
	413 RCL 04	$S_{TF(n)}$				If compos. S_{TF} is negative (i.e. N.A. in slab) then DL stress counteracts LL stress
	414 SIGN	±1				
	415 *	± act f_b				
	416 -	ΔF_b				
	417 RCL 04	S_{TF}				
	418 *	Allow. LLM				
	419 ABS					
	420 RCL 01	BF LLM				
	421 X>Y?					
	422 X<>Y	min.				
	423 STO 01	allow LLM				
	424 RCL 10	$S_{c \cdot n}$				Slab
	425 .	0 *				* S_c for $n=\infty$ (NCDL)
	426 RCL 10	$S_{c \cdot n}$				
	427 "SLAB"					
	428 RCL 21	CDL M				Slab stress calculations do not use n:3n modulus
	429 ASTO 03	"SLAB"				
	430 FS? 00	CDL M	S_c	0	S_c	
	431 GTO 01					
	432 FS? 05					
	433 X≠0?					
	434 AVIEW					
	435♦LBL 01					
	436 X<>Y	S_c	CDL M	0	S_c	
	437 XEQ 08	act. f_b				

Step	X	Y	Z	T	L
438 RCL 16	all. fc				
439 XEQ 21	act. fc	all fc			
440 FC? 05					
441 GTO 16					
442 -	Δfc				
443 RCL 10	Sc				
444 *	all. LL M				
445 "WS "					
446 FS? 06					
447 "LF "					
448 RCL 01	steel LL M				
449 X>Y?					
450 X<>Y	LL M				
451 XEQ 05					
452♦LBL 16					
453 FS? 04					
454 GTO 12					
455 3	3				
456 FS? 07					
457 1/X	3 or ⅓				
458 ST* 20	(N)				
459 ST/ 14	(sl. A/n)				
460 FS?C 07					
461 GTO 12					
462 SF 07					
463 GTO 29					
464♦LBL 12					
465 FS? 11	*				
466 RTN					
467 GTO B					
468♦LBL 20	Sn	S3n*	SBm	MNCDL	
469 "⊢ FL"					
470 ASTO 03					
471 FC? 00					
472 AVIEW					
473 "NCDL"					
474 R↑					
475 R↑	SBm	M	Sn	S3N	
476 /	fb-1				
477 XEQ 15					
478 FC? 10					
479 GTO 04					
480 LASTX	SBm				
481 *	MNCDL	Scompos	Scompos		
482 R↑	Scompos				
483 /	fb-1				
484♦LBL 04	fb-1	Sn	S3n		
485 RCL 21	CDL M	fb-1	Sn	S3n	
486 R↑	S3-N	CDL M	fb-1	Sn	
487♦LBL 08					
488 "CDL"					
489 /	fb-2				

Step	X	Y	Z	T	L
490 X≠0?					
491 XEQ 15					
492 +	ΣDL fb				
493 FS? 05					
494 RTN					
495 RCL 26	LL M	ΣDL fb	Sn	Sn	
496 R↑	Sn				
497 /	LL fb				
498 "LL"					
499 XEQ 15					
500 +	Σfb				
501 FC? 00					
502 ADV					
503 RTN					
504♦LBL 15					
505 FS? 00					
506 RTN					
507 XROM "K"					
508 XEQ "PV"					
509 RTN					
510♦LBL 21	allow Fb	act fb			
511 X<>Y	act fb	allow			
512 FS? 05					
513 GTO 08					
514 "Σ "					
515 ARCL 03					
516 XROM "K"					
517 GTO 01					
518♦LBL 05					
519 "⊢LL"					
520 XROM "M="					
521 RCL 26					
522♦LBL 01					
523 XEQ "PV"					
524 FC? 11					
525 X=0?					
526 GTO 08					
527 "NG"					
528 X>Y?	act. fb	allow Fb			
529 BEEP					
530 X>Y?					
531 AVIEW					
532♦LBL 08					
533 FC? 00					
534 XROM "L"					
535 RTN					
536♦LBL "BM"					
537 CLX					
538 STO 04	0				
539 "WEB D-"					
540 PROMPT					

Step	X	Y	Z	T	L
541 X=0?					
542 RTN					
543 STO 07	Web d				
544 XEQ 11	Compr Fl.W	Web th			
545 GTO 05					
546♦LBL "SBM"					From PLG
547 SF 11					
548 CF 27					
549 GTO IND X					
550♦LBL 01					
551 GTO 39					
552♦LBL 03					
553 GTO D					
554♦LBL 04					
555 GTO E					
556♦LBL 00					
557 STO 12					
558 XEQ 09					
559♦LBL 05					
560 DEG					
561 X=0?					
562 RTN					
563 STO 03	Compr.Fl.W				
564 PROMPT					
565 STO 04	Comp.Fl.Th.				
566 "TEN"					
567 XEQ 10					
568 STO 09	Tens. W				
569 PROMPT					
570 STO 10	Tens.Th.				
571 STO [					
572 RCL 04	Comp Th				
573 +					
574 RCL 07	Web d				
575 +	Bm d				
576 STO 00					
577 RCL 09	Tens.W				
578 STO ↑					
579 RCL 07	Web d				
580 RCL 08	Web th				
581 RCL 03	Comp Fl.W				
582 RCL 04	Comp Fl Th.				
583 XEQ "AY"	A	Y	S_b	I	
584 STO 01	A				
585 R↑					
586 SF 21					
587 "I"	I				
588 STO 02					
589 XEQ 02					
590 R↑	S_b	I	A	Y	
591 X<> T	Y	I	A	S_b	

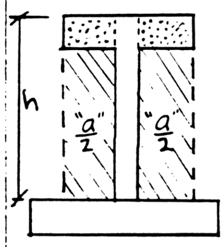
D160

Step	X	Y	Z	T	L
592 STO 05	Y				
593 "Y="					
594 XEQ 23	S_T	A	S_b	S_b	
595 /					
596 STO 11					
597 "COM"					
598 XEQ 22	S_{comp}				
599 "TEN"					
600 R↑	S_{trans}				
601 STO 14					
602 XEQ 22					
603 FC? 06					
604 GTO 03					
605 CLA					
606 RCL 07	Web d				
607 RCL 09	Tens W				
608 RCL 10	Tens Th				
609 *	Tens A				
610 RCL 03	Comp W				
611 RCL 04	Comp Th	Comp W	Tens A	Web d	
612 *	Comp A	Tens A	Web d		
613 -	ΔA				
614 RCL 08	Web th.				
615 /	ΔWeb d				
616 -					
617 2					
618 /	N.A. pos	Web d			
619 X<=Y?					
620 X<0?					
621 GTO 03					
622 ADV					
623 STO 12	N.A.				
624 RCL 08	Web th				
625 RCL 09	Tens W				
626 RCL 10	Tens Th	Tens W	Web th	d	
627 RAD					
628 XEQ "AY"	A d				
629 RCL \	ΣAd				
630 RCL]	ΣA				
631 /	$\bar{d}$				
632 X<> 12	N.A.				
633 RCL 07	Web d				
634 -	new N.A.				
635 CHS					
636 RCL 08	Web th				
637 RCL 03	Comp W				
638 RCL 04	Comp Th	W	Web th	d	
639 XEQ "AY"					
640 DEG					
641 RCL \	ΣAd				
642 RCL]	ΣA				
643 /	$\bar{d}_{compr}$				
644 RCL 12	$\bar{d}_{trans.}$				
645 +					
646 2					
647 /	$\bar{d}/2$				
648 CHS					

(Note beside steps 619–621: X>Y? or X<0? (i.e. N.A. in flange?))

Program COMP For Composite Beam Analysis ______ bytes Page 7 of 8

Step	X	Y	Z	T	L
649 RCL 00	d				
650 +	d				
651 RCL 01	A_{bm}				
652 *	Z×Z				
653 2					
654 /	Z				
655 "BM Z"					
656 STO 12					
657 XEQ 02					
658♦LBL 03					
659 FS? 00					
660 RTN					
661 ADV					
662 "AREA="					
663 ARCL 01					
664 XROM "IS"					
665 AVIEW					
666 XROM "L"					
667 RTN					
668♦LBL 22					
669 "⊦ S"					
670♦LBL 02					
671 "⊦="					
672 ARCL X					
673 FC? 00					
674 XEQ "PY"					
675 RTN					
676♦LBL 11					
677 "WEB TH"					
678 PROMPT					
679 STO 08	Web th				
680♦LBL 09					
681 "COM"					
682♦LBL 10					
683 "⊦ FL W"	Fl. W				
684 PROMPT					
685 "TH"					
686 RTN					
687♦LBL 23					
688 ARCL X					
689 XEQ "AI"					
690 FC? 00					
691 AVIEW					
692 RTN					



Step	X	Y	Z	T	L
693♦LBL "AY"	Comp Th	Comp W	Web Th	Web d	
694 STO]	Th				
695 R↑	Web d				
696 +	"h"	Comp W	Web th	Web th	
697 X<> T	Web th	Comp W	Web th	h	
698 -	"a"	Web th	h	h	
699 RCL]	Comp Th	a	web th	h	
700 *	A				
701 STO]	A				
702 LASTX	Comp Th				
703 *	Ad	web th	h	h	
704 STO \					
705 LASTX					
706 *	Ad^2	web th	h	h	
707 RDN	web th				
708 X<>Y	h	web th	h	Ad^2	
709 *	A_{web}				
710 ST+]	(ΣA)				
711 LASTX	h				
712 *	Ad				
713 ST+ \	(ΣAd)	h	Ad^2	Ad^2	
714 FS? 43					
715 RTN					
716 LASTX	h				
717 *	Ad_w^2	h	Ad_{Fl}^2	Ad_{Fl}^2	
718 R↑	Ad_{Fl}^2				
719 +	ΣAd^2				
720 RCL ↑	Tens W	ΣAd^2	h		
721 RCL Z	h	Tens W	ΣAd^2	h	
722 *	A				
723 ST-]	(ΣA)				
724 LASTX	h				
725 *	Ad				
726 ST- \	(ΣAd)				
727 LASTX	h				
728 *	Ad^2	ΣAd^2	h	h	
729 -	ΣAd^2	h	h	h	
730 RCL ↑	Tens Fl. W				
731 RCL [	Tens Fl Th				
732 R↑	h	Th	W	ΣAd^2	
733 +	d	W			
734 STO [					
735 *	A				
736 ST+]					
737 LASTX	d				
738 *	Ad				
739 ST+ \					
740 LASTX	d				
741 *	Ad^2	ΣAd^2			
742 +	ΣAd^2				
743 3					
744 /	I				
745 RCL \	ΣAd				
746 RCL]	ΣA				
747 ST+ X					
748 /	$\bar{Y}$				
749 X↑2	$\bar{Y}^2$	I			

D161

Program COMP For Composite Beam Analysis ______ bytes

Step	X	Y	Z	T	L
750 LASTX	Y	Y^2	I		
751 X<>]	ΣA	Y^2	I		
752 *	$ΣAY^2$	I			
753 LASTX	A				
754 RDN	AY^2	I		A	
755 +	I				
756 RCL]	Y	I		A	
757 R↑	A	Y	I		
758 X<> \	ΣAd	Y			
759 *	$ΣAd\bar{Y}$	I			
760 -	I				
761 ENTER↑					
762 ENTER↑	I	I	I		
763 RCL [	d				
764 RCL]	$\bar{Y}$				
765 -	Y_{bottom}	I	I	I	
766 /	S_b				
767 RCL]	Y				
768 RCL \	A	Y	S_b	I	
769 RTN					
770◆LBL 31					
771 "AISC"					
772 XROM "AA"					
773 CF 06					
774 .4					
775 FS? 04					
776 .45					
777 STO 16	.45				
778 "WSD?"					
779 PROMPT					
780 X=0?					
781 SF 06					
782 CF 08					
783 CLX					
784 "PLAST?"					
785 FS? 06					
786 PROMPT					
787 X>0?					
788 SF 08					
789 .85					
790 STO 24	.85				
791 FS? 06					
792 STO 16	.45?				
793 8					
794 "N"					
795 FC? 09					
796 PROMPT					
797 STO 20	n				
798 SF 10					
799 CLA					
800 FC? 09					
801 XEQ "Fc"	f_c'				
802 ST* 24	$(.85f_c')$				
803 ST* 16	$(.45f_c')$				
804 SF 10					

Step	X	Y	Z	T	L
805 FS? 04					
806 FS? 06					
807 CF 10					
808 CF 22					
809 FC? 09					
810 FS? 00					
811 GTO 05					
812 "STUD "					
813 FS? 10					
814 GTO 07	f_c'				
815 .75					
816 Y↑X	$(f_c')^{3/4}$				
817 7.25	$7.25()^{.75}$				
818 *					
819 "⊢D="					
820 PROMPT	Stud dia.				
821 X↑2	d^2				
822 *	Stud cap.				
823 .87					
824 FC? 04					
825 SIGN					
826 /	Stud cap.				
827 FS? 22					
828 GTO 05					
829 CLA					
830◆LBL 07					
831 "⊢CAP, K"					
832 PROMPT	Stud Cap				
833 2					(for AISC
834 /					ASD)
835◆LBL 05					
836 FC? 22					
837 CLX					
838 STO 25	Stud Cap.				
839◆LBL "FY"					
840 36					
841 "F-Y KSI"					
842 PROMPT	F_y				
843 STO 06					
844 66 *					* AISC
845 X<>Y	F_y				LRFD
846 X>Y?					A5.1
847 BEEP					
848 X>Y?					
849 CF 08					
850 CF 18					
851 FS? 08					
852 SF 18					
853 END	F_y				

D162

PLATE GIRDER ANALYSIS

xeq PLG   (SIZE 032)

SOFTKEYS (USER mode ON)

- a Input intermediate stiffener spacing
- b Input new flanges and find moment capacity
- c Input new moments and lateral bracing spacing
- d Input new web section
- e Input new shear within a panel (e.g. for a new stiffener location)

Note: do not use softkey "b" before having begun flange input once already, since web and stiffener information must be processed before the beam's section and strength properties can be calculated.

USER FLAGS

Flag 00	clear	Full output
	set	Reduced output-many intermediate calculations not displayed
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 04	clear	AASHTO
	set	AISC

INPUT SUMMARY Note: The following summary is for a non-composite plate girder. If analyzing a composite plate girder, many of the prompts encountered are summarized in the documentation for program COMP and are not repeated here. The user should become familiar with COMP before using PLG on a composite beam.

Prompt	Input	Default
AASHTO or AISC	A reminder, not a prompt. Push R/S to continue.	
WSD?	Asking if by working stress method. Respond 1 for yes (WSD/ASD), 0 for no (LFD/LRFD).	1 (WSD)
F-Y KSI	Steel yield stress, ksi.	No default
M1,'K	Greatest moment within panel being analyzed, in foot-kips. For composite designs, the sum of the NCDL, CDL, and LL moments input is used for M1 and it is not prompted.	No default

<u>Prompt</u>	<u>Input</u>	<u>Default</u>
M2,'K	Smallest moment within the panel in question. If moment reversal occurs, M2 is negative. <u>For an end panel, input M2 as zero.</u>	M1,'K
V, K	Shear at the location of interest within the panel.	No default
LAT BR SPA'	Spacing of the lateral bracing, or unbraced length of the compression flange, in <u>feet</u>. Assumed zero and not prompted for composite beams.	No default
WEB D"	Depth of the web plate, inches. Bypass (or enter zero) to accept last web configuration entered and begin stiffener calculations (if stiffeners are needed) or flange trials (if stiffeners aren't needed).	Zero
TH"	Web thickness, inches.	No default
END or INT PANEL:	A message, not a prompt. Push R/S.	
ST SPA"	Stiffener spacing, inches. Bypass (or enter zero) to accept last spacing and begin flange calculations. Press softkey e (lowercase) to input a new shear value (i.e. to look at another location within the panel).	Zero
COMP FL W"	Compression flange width, inches. If satisfied with previous flange, bypass (or key in zero) to enter new loads and begin analyzing a new panel.	Zero
TH"	Compression flange thickness.	No default
TENS FL W"	Tension flange width, inches.	No default
TH"	Tension flange thickness, inches.	No default

<u>OUTPUT SUMMARY</u> (Items in parenthesis are displayed only if flag 00 is clear)

(MAX D/TW)	Maximum allowable web slenderness
D/TW	Actual web slenderness

ACT FV	Acting shear stress on the trial web
STIFF REQD or NO STIFF REQD	Telling whether or not intermediate stiffeners are needed for shear
MAX ST SPA"	Maximum allowable stiffener spacing, determined by either spacing (for interior panels) or shear requirements (for end panels). If an interior panel, shear requirements may dictate a spacing considerably less than this calculated maximum.
TENS FIELD or NO TENS FIELD	Telling whether or not tension field action is being utilized. The program only uses t.f.a. when allowed by code; and for AISC designs, only when needed.
CV	Shear constant.
ALL FV	Allowable shear stress, ksi. If greater than ACT FV, displays OK; otherwise NG (no good).
(I, Y, S-C, S-Te, A)	Elastic section properties of the beam: moment of inertia, neutral axis depth, section moduli of the compression and tension flanges, and area of the section.
(Z)	Plastic section modulus of the beam, in inches cubed (only if by LFD/LRFD)
b/T	Compression flange width to thickness ratio.
Cb	Bending constant, to account for moment gradient on allowable bending stress.
ALL L/b	Allowable compression flange unbraced length to width ratio. See "FLANGES" section.
L/b OK or NG (AASHTO WSD only)	Checks compression flange L/b.
(L/r_T)	Unbraced length property for AISC designs.
(K_c) (AISC ASD)	Web/flange slenderness interaction constant
(R_b) (AASHTO LFD)	Composite dead load factor
TENS and COMP Fb	Allowable bending stresses for tension and compression flanges, in ksi (WSD/ASD only).
M	Allowable moment capacity.

D165

PROGRAM FLAGS

Flag 05	clear	Interior web panel
	set	End panel-no tension field action
	clear	Have live load moment in COMP
Flag 06	clear	WSD (working stress design)
	set	LFD (load factor design)
Flag 07	clear	OK
	set	NG (no good)
	clear	No stiffener required
	set	Stiffener is required
	clear	L/b_{fl} is under limit (OK) (AASHTO WSD)
	set	L/b_{fl} is over limit (NG)
	clear	h/t_w is under limit (beam) (AISC LRFD)
	set	h/t_w is over limit (plate girder)
Flag 08	clear	Elastic design only
	set	Plastic design permitted by designer
Flag 09	clear	Composite beam
	set	Non-composite beam
Flag 10	clear	AISC LRFD or AASHTO
	set	AISC ASD
Flag 11	set	Within a subroutine
Flag 18	clear	Elastic design only
	set	Plastic design allowed by code (compact)
Flag 19	clear	Symmetrical beam (identical flanges)
	set	Asymmetrical beam
Flag 20	clear	Tension field action allowed
	set	No tension field action allowed

STORAGE REGISTER USE

Register	Value
00	Total beam depth (in SBM)
01	C or Cv; Beam area (in SBM); Allow. LL Moment
02	Beam I (in SBM);
03	Max. D/tw; Compr. flange width; Beam S_{bf}; "B FL"
04	$(D/tw)^2$; Compression flange thickness; S_{tf} (n=n)
05	$(1 + (D/do)^2)/(D/tw)^2 Fy$; Beam N.A.

Register	Value
06	Fy, ksi
07	Web depth
08	Web thickness
09	Allowable Fv; Tension flange width (SBM); $1/r_T$; Composite N.A.
10	Actual D/tw; Tension flange thickness; M_R; Concrete compr. force; S_c (n=n)
11	D/t_w; Top flange S; (Slab + Haunch + Beam N.A.); S_{bf} (n=3n)
12	Beam Z; Allowable capacity M_u (in SFb); Effective slab thickness
13	Slab Thickness
14	Beam S_{bf}; Slab area/N
15	Slab thickness + Haunch
16	.40, .45, or .85 x F_c'
17	NCDL M ("K stored)
18	Allowable tension flange stress
19	M_p (in SFb); S_{tf} (n=3n)
20	N
21	CDL Moment
22	Actual F_v; Number of studs required/available
23	r_y; r_y X_1; "#STUDS"; S_{bf} (n=n)
24	.85 Fc'
25	Stud capacity/2
26	Live load moment
27	Slab area
28	M_1/M_2
29	Shear interaction reduction factor
30	Lateral bracing spacing (inches stored)
31	M_2

PLATE GIRDER ANALYSIS

Program PLG analyzes a plate girder design for compliance with code requirements, and determines allowable stresses. It considers a beam one section at a time, considering shear, intermediate stiffener spacing, moment gradient, and section properties within each bracing bay or section of interest. The program can analyze a girder by one of four codes:

AISC ASD (Ninth Edition)
AISC LRFD (First Edition)
AASHTO WSD (1989)
AASHTO LFD (1989)

AISC is followed if flag 04 is set. If flag 04 is clear, AASHTO is followed. At the beginning of a run, the designer is prompted "WSD?" Answering this with a 1 (yes) sets the program for ASD/ WSD; responding with a 0 (no) results in the use of LFD/LRFD rules.

For each bracing bay or stiffener panel analyzed, the program steps through the following sequence:

1) Prompt maximum shear in panel, moments at each end, and spacing of lateral bracing.

2) Try web configurations; check D/t_w and acting shear stress; see if intermediate stiffeners are needed.

3) Using last web configuration input, calculate minimum stiffener spacing; try different spacings, calculating allowable shear stress for each.

4) Using last web and stiffener input, try different flange sizes; calculate section properties; check B/t_f and bracing requirements.

5) Calculate allowable stresses (ASD/WSD) or ultimate moments (LFD/LRFD) for each flange trial.

6) Start sequence for new panel.

Each step of the sequence repeats itself for new trials until the user bypasses the first prompt of that step, thus accepting the last parameter input. See the examples for clarification.

As mentioned, PLG analyzes a girder one section or bracing bay at a time; each section can be different. It prompts for moments M1 and M2 at the ends of the bay or section, the maximum shear V within the section, and the unbraced length if non-composite (the length of the

D168

bracing bay for noncomposite beams). If a uniform section is desired, it's still not a good idea to input the maximum moment and maximum shear as occurring at one point if they don't. For one thing, the lower shear coincident with the maximum moment may give a higher allowable bending stress. Also, the web stiffener analysis will change with the stiffener spacing, which presumably will change.

The value of the shear is a design variable affecting the intermediate stiffener area requirements and spacing and the allowable flange stress calculations, among other things. As such it must be calculated and input carefully. The moment values M1 and M2 are a little less critical. They are used for three things. First, if M2 is input as zero, the program assumes an end panel and the shear and stiffener calculations are done accordingly.

Second, the program uses the moments to figure C_b (for AISC) or to check for the 20% increase in bending stress allowed by AASHTO LFD. If designing by AISC, and the panel has a higher moment in the middle than at the ends, M1 and M2 must be input the same to force C_b=1.0.

Third, for AASHTO LFD, the maximum unbraced length for full plastic moment capacity is based on the moment ratio.

The sign convention used by PLG is that moments inducing single curvature are both positive; if they induce reverse curvature, the signs are opposite (either M1 or M2 is negative). Note that this is opposite the convention used by AISC.

Some of the things PLG does <u>not</u> do are:

1) Analyze hybrid girders. The web and flanges must be the same yield strength. Program BM may be used to calculate the section properties of a hybrid girder.

2) Design or analyze bearing or intermediate stiffeners, <u>unless</u> RAM program "STIF" is keyed into main memory; a listing follows. Space constraints prevented this from being included with PLG in the ROM.

3) Consider fatigue. The designer must do this.

4) Check stresses (yield on gross section against ultimate on net section) at splices or other bolt holes, or check that bolt hole area is less than 15% of gross area.

5) Check deflection or camber (although it does calculate moment of inertia).

6) Analyze beam-columns. Direct axial stress P/A is assumed to be negligible. See program SBCOL.

The following information concerning what the program <u>does</u> do, and how it does it, is broken down into seven sections, concerned respectively with:

softkeys
plastic analysis
the web
intermediate stiffener spacing
the flanges
allowable stresses and moment capacity
composite design
stiffener analysis

<u>SOFTKEYS</u>

Softkeys are used in PLG mainly for jumping from one area of analysis (for instance, stiffener spacing), to another (e.g. flanges). Softkey operation is complicated by the fact that PLG uses many subroutines outside the main program. However, most of these are either general-purpose subroutines such as "S", which have no softkeys of their own; or portions of program COMP (for input and composite beam design). All of the softkeys in COMP are uppercase (unshifted); all of PLG's softkeys are lowercase (shifted). Thus, to use a softkey in PLG, <u>hold it down</u> so that you can read its function in the display. If the softkey calls its native function (X↑2, Y↑X, etc.) instead of XEQ a or XEQ c, it means that you aren't within PLG and the softkey is inaccessible at the moment; continue holding it until it reads NULL. Then continue running the program until you're back in the main body, and try again.

Note that when analyzing composite beams, the program spends a lot of time within program COMP; but when running non-composite beams, most of the external subroutines are simply short output and display routines which can be gotten through quickly with a couple of SST's. As a quick check, USER mode is turned off while within COMP and turned back on upon re-entry into PLG.

<u>PLASTIC DESIGN</u>

Another potentially confusing item is the PLAST? prompt near the end of the initial input sequence. For design by LFD/LRFD, this prompt gives the user the option of disallowing plastic section analysis (the prompt does not appear for WSD/ASD designs). In other words, the maximum moment capacity is computed using

D170

elastic section properties at first yield. This "veto" ability was initiated for several reasons:

1) AASHTO eq. 10-94 involving flange-web slenderness interaction is not checked. For right-to-the-wire designs where this might be a problem, the user must check it and disallow plastic properties if necessary.

2) AASHTO 10.48.3 involving web thickness transitions is ignored, i.e. the allowable moment capacity is interpolated between elastic and plastic values. If the user is not comfortable with this, PLAST? can be disallowed for these "in-between" designs.

3) PLG automatically disallows plastic design for steel yield strengths above 65 ksi per AISC LRFD A5.1. This is assumed to meet the requirements of AASHTO 10.48.1.2.

4) For composite beams, flange b/t ratios are not checked since the flange is assumed restrained from local buckling by the shear connectors. For thin flanges with wide connector spacings (or nervous designers) it may be desireable to override this assumption.

5) Inevitably, some agencies will refuse to allow full utilization of plastic strength properties on their structures.

Answering 1 (yes) to the PLAST? prompt does not mandate plastic section analysis; it merely allows it. Applicable code provisions are still checked.

The above discussion is concerned with plastic section analysis; PLG assumes that loads are derived from an elastic structure analysis. AASHTO 10.44.1 is assumed to be met, with the 9/10 exception allowed by 10.48.13; for AISC LRFD designs, equation F1-4 is checked, not equation F1-1. If the user verifies that a section meets the requirements of F1-1, then PLG may be used for a plastically analyzed structural member by LRFD (i.e. the loads may be from a plastic structural analysis). Doing this generally results in a significant decrease in the design loading and thus in the level of conservatism.

Note that the status initially set by the PLAST? prompt can be changed manually at any time by changing the setting of flag 08. This should be done before the program begins calculating the allowable capacity; before or during input of the flange dimensions is a good time.

D171

WEB

The effective depth of the web used by PLG is the distance between the flanges rather than the full member depth, both for web slenderness and shear stress calculations. This is in keeping with AASHTO criteria and is in the spirit of AISC ASD rules (assuming web depth is 95% of full depth) for web slenderness; it is contrary to the liberalization allowed by AISC LRFD F2.1.

The value of h_c is correctly taken as twice the neutral axis depth for asymmetrical non-composite beams; however, for composite beams, the value used is that of the bare beam. This is conservative compared to the codes, and seems prudent given the uncertainty of construction loads prior to hardening of the slab. These limits are concerned primarily with slender web buckling. Putting a composite slab on a beam with its web already undergoing flexural buckling does not reverse the buckling.

AISC rules require calculation of a bending stress reduction factor, R_{PG}, for slender webs. PLG uses web depth per ASD section G2 rather than h_c as called for by LRFD A-G2-3.

The shear constant C_v is the ratio of a panel's buckling strength to its yield strength in shear. C_v may calculate and be displayed as greater than 1.0; however, for AASHTO and LRFD designs, a maximum of 1.0 is used since the web will yield in shear before it buckles. AISC ASD effectively liberalizes this limit to (0.4 x 2.89) or 1.156; i.e. F_v is never greater than 0.40 F_y.

The maximum web depth:thickness ratio for <u>AISC</u> is figured assuming stiffeners spaced at 1.5 times the depth; this ratio is only slightly greater than that for an unstiffened girder. However, maximum stiffener spacing/depth is calculated as the greater of $(260tw/h)^2$ or 3.0, not 1.5. This will very seldom cause conflict, since for the large h/t ratios typically involved, either the $(260t_w/h)^2$ requirement or the shear stress requirements will dictate spacings less than 1.5h.

Maximum D/tw requirements for <u>AASHTO</u> are based on AASHTO 10.34.3.1 and 10.48.5.1, and similarly assume maximum stiffener spacing of 3.0 D, or 1.5 D for end panels (panels where one of the moments is input as zero). The increase allowed by 10.34.3.1.1 for bending stress less than allowable is not used, since at this point in the program the bending stress is not known. Stiffeners are required for D/t_w greater than 150, and to meet shear requirements.

D172

The web slenderness is checked against the AASHTO value of $608/\sqrt{F_y}$ over the web depth instead of $640/\sqrt{F_y}$ over the full depth, assuming the flanges comprise 5% of the depth (608 = .95 x 640). This is within the spirit of ASD rules and is conservative for LRFD. Another minor adjustment is made to the AISC ASD shear strength constant of 190 used for the calculation of C_v. The program uses $\sqrt{36000}$, which is 189.74, to match the AASHTO wording. For the same reason, PLG uses 186.95 instead of 187 in this portion of the LRFD code. These differences are obviously insignificant but may cause very minor differences between computed and hand-calculated versions of the same design.

PLG does not check web crippling; this must be investigated by the designer. Neither does it does check that the web is at least 5/16" thick per AASHTO 10.8.1 if designing by AASHTO. No thickness minimum is required for AISC designs.

INTERMEDIATE STIFFENERS

Stiffener spacing is checked and a warning displayed if it is greater than 3.0 D. If the limit is violated, the program still continues; however, the user may wish to use softkey "a" to input a new stiffener spacing. If an end panel is being analyzed by AASHTO, the limit used is 1.5 rather than 3.

Thicker webs may not require any intermediate stiffeners. However, if the actual/allowable shear stress is greater than 0.6 they may be desirable to eliminate flange stress reduction. Thus, even if the program finds NO STIFF REQD, the stiffener input sequence will still be entered. Or, press softkey (shift) a (in USER mode) to get to the stiffener input sequence and input a new stiffener spacing from elsewhere in the program. If no stiffeners are desired, simply bypass the STIFF SPA prompt (press R/S with no numeric entry) to get on to the flange design portion.

The portion of the program related to the analysis of the actual stiffeners had to be left out of this version of BRIDGEROM due to severe space constraints (largely due to the addition of AISC LRFD provisions, absent in the previous version). However, a listing of RAM program "STIF" is included. If this is present in main memory, it will be accessed as a subroutine by PLG and will analyze both intermediate and bearing stiffeners. NOTE: if you have any programs presently in main memory with the Label STIF, please change the title if intending to use PLG. For more information, see page D178.

FLANGES

For composite designs, PLG assumes an unbraced length of zero and Lb' (lateral bracing spacing) is never prompted.

Compression flange b/t ratios are checked using full flange width. For composite beams before hardening of the slab, the requirements of AASHTO 10.34.2.1.5 or 10.50(c), and AISC LRFD I3.4, regarding width to thickness ratios as a function of dead load stress, are not checked by the program and should be verified by the designer.

AASHTO WSD has upper limits on compression flange L/b, while AASHTO LFD and AISC simply keep reducing the allowable (compressive) bending stress. If an AASHTO WSD run finds "L/B NG", it is a Code violation, and the program returns for a new beam section. For situations where neither the unbraced length nor the flange width can be changed (i.e. rating of existing structures), the user is left with the choice of ignoring the limit and extrapolating the parabolic reduction equation beyond that point (as the other codes do) or pronouncing the structure impossible to rate or possessing no strength. PLG will be of no help in any of these situations, since it does acknowledge the code cutoff point.

For very wide thin flanges, AISC LRFD equation A-B5-3 is identical to the Eighth Edition ASD equation C2-3. This has been changed in the Ninth Edition ASD spec., appearing as ASD equation A-B5-3; the equations are similar except for the introduction of a web slenderness interaction constant K_c. PLG incorporates K_c into ASD equation F1-4; however the infrequently used Appendix B equation is the older Eighth Edition ASD C2-3 (LRFD A-B5-3). For designers who like to stretch the bounds of present knowledge concerning very wide thin flanges, the newer Ninth Edition equation should be checked by hand.

The program makes a few minor adjustments to code constants in the interest of efficient use of code. The LRFD constant of 11,200 in the b/2t check is taken as 11,250. Also, for AISC ASD designs, the program assumes $b/2t < 195//F_y\ K_c$; it doesn't check equation A-B5-4 in appendix B.

ALLOWABLE STRESSES and MOMENT CAPACITY

The AISC LRFD code explicitly considers the strength of a beam as the lowest of three values based on analysis of web slenderness (WLB), flange slenderness (FLB), and overall (unbraced length) buckling (LTB). Program PLG uses this approach for all four of the

D174

codes accepted (AASHTO WSD/LFD and AISC ASD/LRFD); the moment capacity of the beam is calculated based on each of the three criteria (WLB is not considered for LRFD "plate girders" (very slender-webbed members) and for AASHTO WSD analysis, which has no such provisions). The final moment capacity output is the least of these two or three values. If by LRFD, this final capacity is multiplied by the undercapacity factor phi (which is why it is not the same as any of the preliminary values).

Except for AASHTO WSD, the codes recognize three levels of "compactness" for each of the three beam criteria: compact
intermediate between compact and non-compact
non-compact

"Compact" beams have a moment capacity (allowable stress) equal to the plastic moment of the beam (0.66 F_y for AISC ASD). "Non-compact" beams have a capacity (allowable stress) less than or equal to the elastic strength at first yield (0.60 F_y for ASD). And, "intermediate" beams have a strength (stress) interpolated between these levels.

ALLOWABLE STRESSES: WSD/ASD

For WSD/ASD, the program uses the actual value of 0.55, .60, or .66 F_y rather than the rounded-off values often used; i.e. for compact AISC designs it outputs 21.6 ksi instead of the familiar 22 ksi (Grade 36 steel). These values are then reduced for lateral-bracing, flange-slenderness, and shear-interaction requirements; AISC designs also are reduced for web slenderness.

For reduction in the allowable bending stress due to high shear stress in the web (shear interaction), AISC reduces only the allowable tension web stress, while AASHTO simply reduces all stresses. PLG reduces both tension and compression flange stress, which is correct for AASHTO and makes no difference for most AISC designs. In the unlikely event that AISC would allow a higher compression than tension stress, the output allowable compression stress will thus be slightly conservative.

AISC only makes this reduction if tension field action is being utilized; AASHTO considers it for all high shear areas. Thus, when designing by AISC, the REDX=... display only appears if tension field action was necessary to carry a shear load with a given web and stiffener spacing. If the web has sufficient shear capacity without relying on the tension field, no reduction is applied. AASHTO takes a simpler view of things; tension field action is always assumed used and

D175

the reduction is always investigated. For simplicity, PLG takes the constants in AASHTO WSD equation 10-29 in 10.34.4.4 as .55 x (1.375 and .625) instead of .754 and .34; the difference is negligible.

This reduction is not additive with reductions in compressive flange stress due to lateral bracing requirements for WSD/ASD designs. However, due to the way the equations are written, the reduction is additive with other reductions for LFD/LRFD. Program PLG takes this shear-interaction reduction as additive in all cases and is thus slightly conservative for ASD/WSD designs. While the AASHTO WSD code is a generally conservative code, AISC ASD often results in members which are lighter than the other codes would allow. Thus this small degree of conservatism is not inappropriate.

MOMENT CAPACITY: LFD/LRFD

As mentioned, moment capacities for beams with compactness properties in the "intermediate" range are figured by interpolating between the elastic moment at first yield and the plastic moment. For WLB (slender web) calculations, the elastic moment is figured with the section modulus taken with respect to the compression flange, and when this flange is smaller than or equal to the tension flange this calculation is straightforward. However, when the compression flange is distinctly larger than the tension flange, its section modulus may be greater than the plastic modulus Z of the beam, which, of course, represents a physically impossible situation. In this case, neither AASHTO nor AISC offers much advice. Whether the intent is to use the moment corresponding to first yield of the compression flange with the tension zone partially yielded (a difficult computation) or the much lower moment at tension flange yield, or some other value, is unknown. In this instance, PLG uses the full compression section modulus, but with an upper limit of the plastic modulus. In other words, the program may interpolate between Z and Z. Please note that this applies only to WLB calculations.

For the case of flange slenderness (FLB) calculations, the elastic moment is figured differently for AASHTO LFD and AISC LRFD designs. For thin-webbed "plate girder" design, LRFD appendix G2 is quite straightforward, and the program follows it directly. For stout-webbed "beam" design, LRFD reduces the yield stress by F_r, the assumed residual stress; PLG uses the compression flange S_x for this case. AASHTO also specifies the full compression flange S_x, but with the full yield stress. In this case, as for web slenderness, PLG uses this moment up to a limiting value of the plastic moment.

0176

COMPOSITE BEAMS

Note: this section mostly reiterates information concerning composite built-up beams from the preceding material. The documentation for program COMP concerns itself with all composite beams and should be consulted along with this documentation for composite plate girders.

The stability of a composite beam during construction is not checked directly by the program. To do this a separate run must be made with the COMP? prompt answered 0 (no). The designer should input the weight of the wet slab and beam as dead load, and whatever construction live load is likely to be applied. Depending on the degree of restraint afforded by the slab falsework, the unbraced length of the compression flange will probably be the distance between cross bracing. AASHTO 10.50(e), forbidding tension field action in a composite beam before hardening of the slab, is not checked by the program and must be checked by the designer. AASHTO 10.50(d) is not checked, but the constant R_b is calculated for use by the designer.

Non-composite beams are checked on the basis of three criteria: web slenderness (WLB), flange slenderness (FLB), and unbraced length (LTB). For composite beams, only web slenderness is checked. For plastic design, if the web is compact, a fully plastic composite section is used. However, if the web falls between the compact and non-compact limits, PLG does not interpolate between the plastic and elastic (first yield) levels of strength; the moment capacity output is that of the elastic section at first yield, as directed by both AASHTO 10.50.1.2 and AISC LRFD I3.2.

The value of h_c is correctly taken as twice the neutral axis depth for asymmetrical non-composite beams; however, for composite beams, the value used is that of the bare beam. This is conservative compared to the codes, and seems prudent given the uncertainty of construction loads prior to hardening of the slab.

For composite beams, flange b/t ratios are not checked since the flange is assumed restrained from local buckling by the shear connectors. For thin flanges with wide connector spacings, the designer should check by hand that FLB does not govern the design. For composite beams before hardening of the slab, the requirements of AASHTO 10.34.2.1.5 or 10.50(c), and AISC LRFD I3.4, regarding flange width to thickness ratios as a function of dead load stress, are not checked by the program and should be verified by the designer.

The flange is also assumed restrained against LTB by the slab and shear connectors. For composite designs, PLG assumes an unbraced length of zero and Lb' (lateral bracing spacing) is never prompted.

STIFFENER ANALYSIS

Program PLG does not analyze stiffeners by itself. However, it will access a subroutine in Main Memory named STIF if such a program exists. (Note: If by chance you have any programs in memory named "STIF", change the name if using program PLG.) A listing of this program follows. It uses the same flag conventions and storage registers as PLG, and requires 393 bytes (57 registers) of memory in RAM.

STIF analyzes both bearing and intermediate stiffeners. Bearing stiffeners are assumed to consist of one plate on each side of the web; intermediate stiffeners may be one or two plates. All applicable provisions of AASHTO or AISC are checked, with these exceptions:

1) AASHTO LFD bearing stiffener analysis gives an allowable factored load instead of following WSD criteria as specified.

2) AASHTO WSD intermediate stiffener analysis uses the minimum area requirement from the LFD spec.

3) Shear transfer (i.e. weld size) is not addressed. Neither is intermediate stiffener length.

4) All bearing stiffeners are analyzed as columns with effective length equal to web depth. This is contrary to the 75% criterion of AISC.

5) Single intermediate stiffener moment of inertia is calculated from center of web instead of edge of stiffener plate.

D174

Program STIF For ________ ________ bytes

X	Y	Z	T	L

```
01♦LBL "STIF"
02 CF 25
03 XROM "LL"
04 CLX
05 CLA
06 XROM "NO"
07 "├ INT ST 1,2"
08 PROMPT
09 X=0?
10 RTN
11 1
12 X=Y?
13 2.4
14 STO 00
15 RCL 11
16 1/X
17 ENTER↑
18 X↑2
19 ENTER↑
20 SIGN
21 LASTX
22 +
23 SQRT
24 /
25 -
26 RCL 07
27 RCL 08
28 *
29 RCL 00
30 *
31 2
32 /
33 1
34 RCL 01
35 -
36 *
37 RCL 09
38 /
39 RCL 22
40 *
41 FC? 10
42 GTO 00
43 *
44 RCL 07
45 50
46 /
47 X↑2
48 X↑2
49 GTO 03

50♦LBL 00
51 .3
52 *
53 RCL 08
54 X↑2
55 18
56 *
57 -
58 RCL 11
59 X↑2
60 2.5
61 *
62 2
63 -
64 .5
65 X>Y?
66 X<>Y
67 RDN
68 "J"
69 XROM "V"
70 RCL 07
71 RCL 11
72 [illegible]
73 *
74 RCL 08
75 3
76 Y↑X
77 *

78♦LBL 03
79 STO 10
80 "MIN I"
81 XROM "V"
82 "MIN A"
83 X<>Y
84 X<0?
85 "├<0 "
86 STO 12
87 XROM "V"
88 256
89 RCL 06
90 *
91 FS? 06
92 6760
93 CLA
94 XEQ 14
95 SF 14

96♦LBL 07
97 XROM "L"
98 CF 07
99 "INT"
100 XEQ 11
101 X=0?
102 GTO 10
103 RCL 10
104 R↑
105 RCL 00
106 INT
107 ST/ 02
108 /
109 *
110 3
111 /
112 STO 05
```

X	Y	Z	T	L

B a/d *

Program STIF For Plate Girder Stiffeners ______ bytes Page 2 of 3

	X	Y	Z	T	L

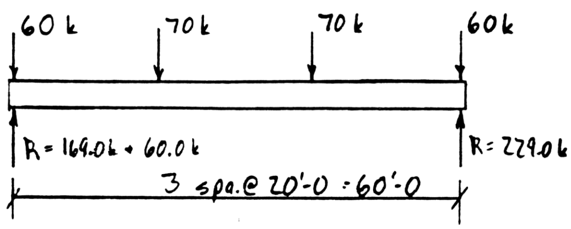
```
113 RCL 04
114 RCL 07
115 30
116 /
117 2
118 +
119 FS? 04
120 CLX
121 X>Y?
122 SF 07
123 95
124 RCL 06
125 SQRT
126 /
127 FC? 04
128 16
129 RCL 04
130 RCL 03
131 /
132 X>Y?
133 SF 07
134 RCL 02
135 RCL 12
136 X>Y?
137 SF 07
138 RCL 05
139 RCL 10
140 X>Y?
141 SF 07
142 "STIFF"
143 XROM "OK"
144 AVIEW
145 GTO 07

146♦LBL 10
147 XROM "L*"
148 "BS "
149 4752
150 XEQ 14

151♦LBL 03
152 XROM "L"
153 "BRG"
154 XEQ 11
155 X=0?
156 RTN
157 R↑
158 STO 00
159 25
160 FC? 04
161 18
162 "WEB W="
163 ARCL X
164 "⊦ TW?"
165 PROMPT
166 RCL 08
167 *
168 RCL 04
169 -
170 RCL 08
171 *
172 ST+ 02
173 RCL 07
174 RCL 00
175 RCL 02
176 /
177 SQRT
178 /
179 XROM "SFa"
180 RCL 02
181 *
182 "ALL R="
183 ARCL X
184 "⊦ K"
185 AVIEW
186 GTO 03

187♦LBL 11
188 CF 07
189 CLX
190 "⊦ ST W"
191 XROM "IN"
192 PROMPT
193 STO 03
194 "TH"
195 XROM "IN"
196 PROMPT
197 STO 04
198 RCL 03
199 ST+ X
200 RCL 08
201 +
202 *
203 STO 02
204 LASTX
205 X↑2
206 *
207 12
208 /
209 RCL 09

210♦LBL 04
211 RCL 03
212 RCL 04
213 /
214 "b/T="
215 ARCL X
216 X>Y?
217 SF 07
218 XROM "OK"
219 AVIEW
220 RTN
```

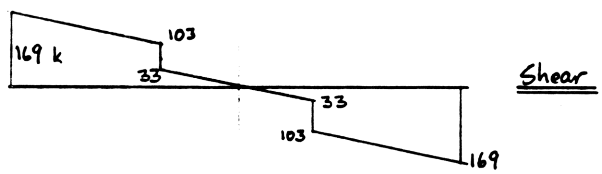
```
221♦LBL 14
222 SQRT
223 FS? 04
224 95
225 RCL 06
226 SQRT
227 /
228 STO 09
229 "⊢MAX b/T"
230 XROM "Y"
231 R↑
232 END
```

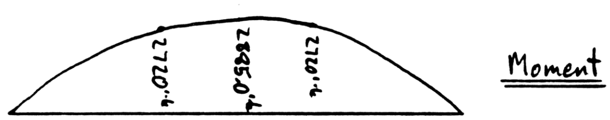
Design a Welded Plate Girder in A36 steel to serve as a transfer beam. Design by AISC ASD.

$F_y = 36$ ksi

Uniform load = 3.1 k/' + Selfwt.
(Assume 200 lbs/' selfwt.)
Consider braced only at 1/3 pts.







Select web depth

Try $l/10 = 72''$; try 1/4" web th.

$h/t = 288$; $\frac{760}{\sqrt{22}} = 162 < 288 \Rightarrow$ Plate Girder

check web h/t by AISC G1-1

$$\frac{14{,}000}{\sqrt{36(36+16.5)}} = 322 > 288 \ \underline{OK}$$

Intermediate stiffeners

$h/t > 260 \Rightarrow$ Int. stiff. req'd (AISC F5.)

Shear stress @ end: $f_v = \frac{169\ k}{72'' \cdot 1/4''} = 9.39$ ksi

Max. Stiffener spacing: $\frac{a}{h} < \left(\frac{260}{h/t}\right)^2 \le 3.0$

Max. $a = 72'' \times \left(\frac{260}{72/1/4}\right)^2 = \underline{58.68''} \le 216.0''$

In end panel, spacing will be governed by shear stress considerations. (AISC 1.10.5.2)

Try first space @ 24":

$a/h = 24/72 = .333$

$a/h < 1 \Rightarrow k = 4.0 + \frac{5.34}{(a/h)^2} = 52.06$

Try $C_v = \frac{190}{h/t}\sqrt{\frac{k}{F_y}} = \frac{190}{288} \cdot \sqrt{\frac{52.06}{36}}$

$= .7933$

$C_v < 0.8 \Rightarrow$ re-calculate

$$C_v = \frac{45{,}000 \cdot k}{F_y \cdot (h/t)^2} = \frac{45{,}000 \times 52.06}{36 \times 288^2}$$

$= .7846$

Allow. $F_v = \frac{F_y}{2.89}(C_v) \le .40 F_y$

$\frac{36}{2.89} \times .7446 = \underline{\underline{9.7731 \text{ ksi}}}$ ← Governs

$.40 \times 36 = 14.40$ ksi

9.77 ksi < 9.39 actual <u>OK</u>

Locate first int. stiff. @ 24"

Thereafter, use tension field action (AISC eq. G3-1). Spacing probably governed by minimum 58.68".

Between first int. stiff. and bearing stiffeners under 70 k loads, have:

20' × 12 =	240"
1st spa	24"
	216"

Int. Stiff. (Cont'd)

$216''/54.64'' = 3.68$

Try 4 spa. @ 54"

$\frac{a}{h} = \frac{54}{72} = .75 < 1.0$

$\Rightarrow k = 4.0 + \frac{5.34}{(a/h)^2} = 13.49$

$C_v = \frac{45{,}000 \cdot k}{36 \cdot (288)^2} = .2034 < 0.8$ OK

$F_v = \frac{F_y}{2.89}\left(C_v + \frac{1 - C_v}{1.15\sqrt{1+(a/h)^2}}\right) < .40 F_y$

$.40\ F_y = 14.40$ ksi

$\frac{36}{2.89} \times \left(.2034 + \frac{1 - .2034}{1.15\sqrt{1+.75^2}}\right) = 9.44$ ksi

Allow. $F_v = 9.44$ ksi > 9.39 ksi OK

Center Panel

$\frac{h}{t} = 288 > 260 \Rightarrow$ stiff. still req'd.

However, shear stress unlikely to govern. Max spa. = 54.68" (see previous page).

$(20' \cdot 12 - 14'')/54.68 = 3.45$

Space @ $\frac{226''}{4} = 56\frac{1}{2}''$

Shear stress OK by inspection in center panel.

Intermediate Stiffener Design

Try single stiffener (AISC 1.10.5.4)

$\text{Min. } A_{st} = \frac{1-C_v}{2}\left(\frac{a}{h} - \frac{(a/h)^2}{\sqrt{1+(a/h)^2}}\right) h \cdot t \cdot D \cdot \frac{f_v}{F_v}$

D = 2.4 for single plate stiff.

1st stiffener: $\frac{1-.7446}{2} \times \left(.333 - \frac{.333^2}{\sqrt{1+.333^2}}\right)$

$= .0245$

Center panel won't govern due to low shear.

Rest of end panel: $\frac{1-.2034}{2}\left(.75 - \frac{.75^2}{\sqrt{1+.75^2}}\right)$

$= .1195$ ← Governs

$\text{Min } A_{ST} = .1195 \times (72 \cdot \tfrac{1}{4}) \times 2.4 \cdot \frac{9.39}{9.44}$

$= 5.13\ \square''$

Try 7" × 3/4" stiff.

$A_{st} = 5.25\ \square''$ OK

$I_{st} = \frac{7^3 \times .75}{3} = 85.8\ in^4$

$\text{Min I} = (h/50)^4 = \left(\frac{72}{50}\right)^4 = 4.30\ in^4$ OK

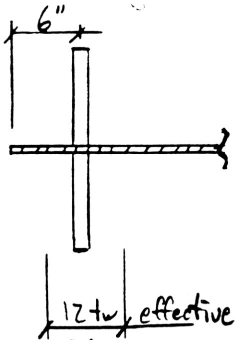
Slenderness: $b/T = \frac{7}{.75} = 9.33$

$\frac{95}{\sqrt{36}} = 15.83 > 9.33$ OK

Use single PL 7 × 3/4

Note: PLG requires Main Memory program "STIF" to analyze intermediate and bearing stiffeners.

End Bearing Stiffeners



effective web (AISC K1.8)

Reaction = 229.0 k

Estimate 20" flanges ⇒ Use 8" plates

Size for ~ 20 ksi allowable.

$\frac{229.0\ k}{20\ ksi} = 11.45\ \square''$

$11.45 - 3'' \times 1/4'' = 10.70\ \square''$

$10.70/2 \times 8'' = .67''$

⇒ $\frac{\text{width}}{\text{thickness}}$ shouldn't govern

(AISC B5.1) $\frac{b}{t} < \frac{95}{\sqrt{F_y}} = \frac{95}{\sqrt{36}} = 15.83$

∴ Use 2 PL 8" × 3/4"

$b/t = 10.7 < 15.83$ OK

$I = 16.25^3 \times .75/12 = 268.19\ in^4$

Area = $2 \times 8'' \times .75'' + 3'' \times 1/4'' = 12.75\ \square''$

$r = \sqrt{I/A} = 4.586''$

Use eff. length = d = 72" (AISC K1.8 allows 54")

From AISC App. A, Table 3-36 for $\frac{Kl}{r} = 15.7$, $F_a = 20.85$ ksi

Allow. P = 20.85 ksi × 12.75 □"
= 265.9 k

265.9 k > 229.0 k **OK**

Since third-pt. stiffeners have greater effective web width and smaller load, and since a thinner stiffener will impinge on width/4th limit,

USE SAME STIFF. @ 1/3 PTS.

FLANGE DESIGN

Check allowable comp. stress (AISC G2)

Web $h/t = \frac{72}{1/4} = 288$

Max. $h/t = \frac{760}{\sqrt{F_b}} = \frac{760}{\sqrt{21.6}} = 163.5$

288 >> 163.5 ⇒ reduce allow. F_b

Assume F_b = 20 ksi for trial Flange

@ Center, M = 2885.0'k

$A_f \approx \frac{M}{F_b h} - \frac{A_w}{6} = \frac{2885.0 \times 12}{20 \times 72} - \frac{72 \times 1/4}{6}$

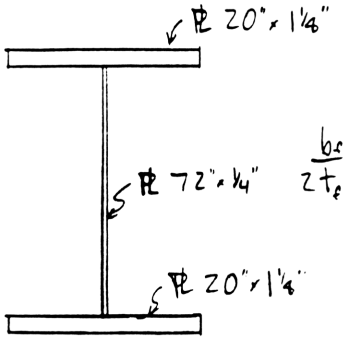
$= 21.04\ \square''$

Long unbraced length ⇒ desire wide flanges.

(AISC B5.1) $\frac{b_f}{2t_f} < \frac{95}{\sqrt{F_y}} = 15.83$

$b_f < 31.66\ t_f$

TRY 20" × 1 1/8" FLANGES



$\frac{b_f}{2t_f} = 8.49 < 15.43$

OK

Web: $72^3 \times 1/4 / 12$ = 7776.0

Fl. $2 \times 1.125^3 \times 20/12$ = 4.75

$2 \times (36.5625)^2 \times 20 \times 1.125$ = 60,156.74

67,937.49 in^4

$S = \frac{67{,}937.49}{37.125''} = 1829.97$

Act. $f_b = \frac{2445.0 \times 12}{1829.97} = 16.92$ ksi

Flange stress reduction in center panel

$I_{fl} = \frac{20^3 \times 1.125}{12} = 750.0$

$A_f + \frac{A_w}{6} = 20 \times 1.125 + \frac{1}{6} \times 72'' \times \frac{1}{4}'' = 25.50 \text{ □}''$

$r_T = \sqrt{\frac{I}{A}} = 5.4233''$

Center Panel: $M_1 = M_2 \Rightarrow C_b = 1.0$

$\sqrt{\frac{102 \times 10^3 \times C_b}{F_y}} = 53.23$ (AISC 1.5.1.4.5(2))

$\frac{l}{r_T} = \frac{20' \times 12}{5.42''} = 44.25$

$\frac{l}{r_T} < 53.23 \Rightarrow F_b = .60 F_y = 21.6$ ksi

Reduction due to web slenderness

$F_b' = F_b \left[1 - .0005 \frac{A_w}{A_f} \left(\frac{h}{t} - \frac{760}{\sqrt{F_b}}\right)\right]$ (AISC G2.)

$F_b' = 21.6\left(1 - .0005 \frac{18.0}{22.5} \left(288 - \frac{760}{\sqrt{22}}\right)\right)$

$= 20.511$ ksi (Comp. Flange)

Max. Tens. Flange $F_b = 21.6$ ksi

Max. $M = 20.511 \text{ ksi} \times 1829.97 \text{ in}^3/12$

$= 3127.90^{'k} > 2445$ OK

Flange Stress in End Panels

$M_1 = 0$; $M_2 = 2720^{'k}$

$\frac{M_1}{M_2} = 0$

$C_b = 1.75 + 1.05\frac{M_1}{M_2} + 0.3\left(\frac{M_1}{M_2}\right)^2 = 1.75$

$\sqrt{\frac{102 \times 10^3 \times 1.75}{36}} = 70.42$

Reduce flanges to 20" x 3/4"

at approx. 10' from supports

$\frac{b_f}{2t_f} = 13.33 < 15.43$ OK

I: web 7776.0

fl $2 \times .75^3 \times 20/12$ = 1.41

$2 \times (36.375)^2 \times 20 \times .75$ = 39,694.22

I = 47,471.63

$S = \frac{I}{36.75''} = 1291.74 \text{ in}^3$

Consider small fl. entire 20' for l/r_T

$I_{fl.} = \frac{20^3 \times .75}{12} = 500 \text{ in}^4$

$A_f + \frac{A_w}{6} = 20 \times .75 + 3.0 = 18.0$

$r_T = \sqrt{I/A} = 5.2705''$

Reduced Flange

$$\frac{l}{r_T} = \frac{20' \times 12}{5.27} = 45.54$$

45.54 < 70.42 **OK**

Tens. $F_b = .60\,F_y = 22$ ksi

Comp. $F_b = 21.6\left(1 - .0005\,\frac{18.0}{15.0}\left(244 - \frac{760}{\sqrt{22}}\right)\right)$

$= 19.967$ ksi

Max. M @ flange reduction

$= 19.967 \times 1291.74\ in^3/12$

$= 2149.37^{\,1\text{-}k}$

Approximately, $\frac{2149}{2720} = \frac{x}{20'}$

$x = 15.8'$

Locate Flange Reduction

15'-0 from ℄ Reaction

Check shear interaction @ Conc. Load

$$f_v = \frac{103\ k}{72 \times 1/4} = 5.722 \text{ ksi}$$

$F_v = 9.44$ ksi (see sh. 2)

$\left(.825 - .375\,\frac{f_v}{F_v}\right) \times F_y = .598 \times F_y \approx .60\,F_y$

⇒ **Use F_b as above**

(AISC G5-1)

Check Web Crippling

Under 3.1 k/' uniform load, with flange restrained against rotation

$$\frac{3.1\ k/'}{12'' \times 1/4''} = \mathbf{1.03\ ksi}$$

Allowable comp. stress (Eighth Edition AISC 1.10-10)

$$\left(5.5 + \frac{4}{(a/h)^2}\right) \times \frac{10{,}000}{(h/t)^2} = \left(5.5 + \frac{4}{(.7447)^2}\right) \times \frac{10{,}000}{244^2}$$

$= 1.45$ ksi **OK**

Note: Program PLG does not check web crippling. This must be done by the engineer. Ninth Edition AISC is not particularly clear on this requirement.

Check Weight

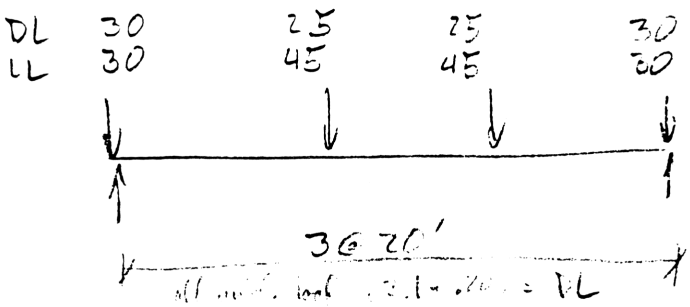
Heavy section	= 214.2 lb/' × 30' =	6426#
Light section	= 163.2 lb/' × 30' =	4896
		11,322

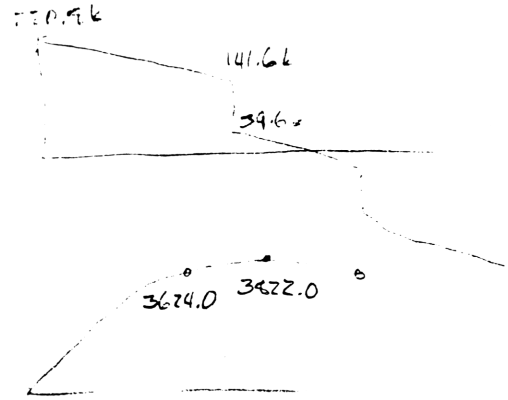
$\frac{11{,}322}{60'} = 188.7$ lb/' avg.

⇒ 200 lb/' assumed is **OK**

ASV or. 6/6

Same Design by LRFD





DL: $M_{1/3} = 1.2\,(25 \times 20' + 3.3^{k/\prime} \times 20' \times 40'/2)$

$= 1.2 \times 1820 = 2184^{\prime k}$

LL: $M_{1/3} = 1.6\,(45 \times 20') = 1440^{\prime k}$

$3624.0^{\prime k}$ @ 1/3 pt

DL: $M_{mid} = 1.2 \times (25 \times 20' + 3.3 \times \frac{60^2}{4})$

$= 1.2 \times 1985 = 2382.0$

LL $\quad 1440$

$3822.0^{\prime k}$ @ mid

Shear $\quad 1.2\,(3.3 \times 30' + 25k) = 148.8$

$1.6 \times 45k = 72.0$

220.8

$1.2 + 1.6 \times 30 = 44.0$

304.8

$1.2 \times 3.3 \times 10' = 39.6k$

$1.2 \times 25k = 30.0$

$1.6 \times 45 = 72.0$

$141.6\,k$

~~(1.2 + 1.6) × 30k = 84.0k~~

225.6

Max. end spa.

$= 72'' \times \sqrt{\dfrac{5}{\dfrac{f_v / .6\phi}{44{,}000/(h/t)^2} - 5}} = 26.18''$

Use same web $72'' \times 1/4''$

end shear stress:

$f_v = \dfrac{220.8}{72 \times 1/4} = 12.27$ ksi

Try @ 24": $a/h = \frac{24}{72} = .333$

$k = 5 + \dfrac{5}{(a/h)^2} = 50.0$

$234\sqrt{\dfrac{k}{F_y}} = 275.8$

$\dfrac{h}{t_w} = \dfrac{72}{1/4} = 288 > 275$

$\therefore C_v = \dfrac{44{,}000\,k}{(h/t_w)^2 F_y} = .7368$

all. F_v = $.6\phi A_w C_v F_y$

$= .6 \times (\phi = .9) \times 72 \times 1/4 \times .7368 \times 36$

~~257.8 k > 220.8 OK~~

$= 14.32$ ksi > 12.27 OK

Note: PLG rounds slightly to

$C_v = .7316$

$V = 14.2218 \times 72 \times 1/4 = 256.0\,k$

LRFD ex 2/6

LRFD ex. 3/6

CKFD ex 4/6

LRFD ex. 5/6

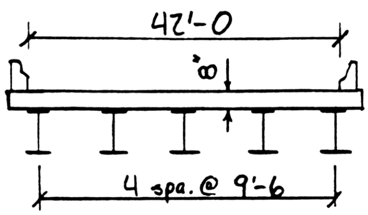
LRFD ex. 6/6

Design the interior girders for a 90'-0 simple span bridge

Design Load: HS20

$F_y = 50$ ksi (A588)
$f'_c = 4.0$ ksi

Use AASHTO Working Stress Design



LOADS

NCDL: Beam est. .20 k/'
Deck .67' × 9.5' × .15 = .95 k/'
1.15 k/'

NCDL $M = \frac{1.15 \times 90^2}{8} = 1164.38$ 'k

CDL (Composite Dead Load)
Rails 2 × .44 k/' / 5 beams = .18 k/'
FWS 42' × .02 ksf / 5 = .17 k/'
.35 k/'

CDL $M = \frac{.35 \times 90^2}{8} = 354.38$ 'k

Live Load

From AASHTO App. A for 90' span:

M = 1344.4 'k/lane ; V = 64.5 k/lane

$\text{Impact} = \frac{50}{L+125} = \frac{50}{90'+125} = .2326$

$\text{Distribution} = \frac{\text{Bm Spa}}{5.5} = \frac{9.5}{5.5} = 1.7273 \frac{\text{wheel}}{\text{beam}}$

∴ M = 1344.4 'k/lane × (1.2326 × $\frac{1 \text{ lane}}{2 \text{ wheel}}$ × 1.7273)

LL M = 1344.4 × 1.0645 = 1431.09 'k

LL V = 64.5 × 1.0645 = 68.66 k
CDL V = .35 k/' × 45' = 15.75
NCDL V = 1.15 k/' × 45' = 51.75
136.16 k

AASHTO 10.5.2 recommends L/d < 30 for compos. constr.

Simple span ⇒ deflection problems

Use 48" Web ; L/d = 22.5 OK

Max. D/t_w = 140 for A588 (10.34.3.1.2)

AASHTO 10.8.1 limits min. thick. = 5/16"

Try 3/8" thick web

$f_v = \frac{136.16 \text{ k}}{48" \times 3/8"} = 7.56$ ksi

$D/t_w = \frac{48"}{3/8"} = 128 < 150$ OK

(10.34.4.1)

$F_v = \frac{5.625 \times 10^4}{(D/t_w)^2} = 3.43$ ksi < 7.56 NG

∴ Int. Stiffeners Req'd

First Stiffener Spa. ≤ D/2

Try 24"

$F_v = \frac{7 \times 10^4 [1 + (D/d_o)^2]}{(D/t_w)^2} \le \frac{F_y}{3}$

$= \frac{7 \times 10^4 \cdot [1 + (48/24)^2]}{128^2} = 21.36$ ksi

$F_y/3 = 16.67$ ksi

16.67 > 7.56 ksi ⇒ 24" OK

Locate remaining stiffeners

Max. spa. = 1.5D = 72" (10.34.4.2)

Try spa. @ max : 72"

$$C = \frac{2.2 \times 10^5 [1 + (D/d_o)^2]}{F_y (D/t_w)^2} \leq 1$$

$$C = \frac{2.2 \times 10^5 [1 + (48/72)^2]}{50 (128)^2} = .3479$$

$$\text{Allow. } F_v = \frac{F_y}{3}\left[C + \frac{.87(1-C)}{\sqrt{1+(d_o/D)^2}}\right] = \underline{11.38 \text{ ksi}}$$

11.38 > 7.56 ksi OK

Check 10.34.4.4 : $\frac{7.56}{11.38} = .66 > .60$

⇒ reduce bending stress. (10.34.4.4)

$$\text{Max. } F_b = (.754 - .34 \cdot \frac{7.56}{11.38}) \times 50 = \underline{26.42 \text{ ksi}}$$

Design Int. Stiffeners

Use 1-side (single) stiffeners

Min. I = $d_o t_w^3 J$

$J = 2.5 (D/d_o)^2 - 2 \geq 0.5$

First Stiff. : $J = 2.5 \times (\frac{48}{24})^2 - 2 = 8.0$

Min. I = 24" · $(.375)^3$ · 8.0 = 10.125 in⁴

Rest of Stiff. : $J = 2.5 \times (\frac{48}{72})^2 - 2 = -.89$

⇒ J = .5

Min. I = 72" × $(.375)^3$ × .5 = 1.898 in⁴

∴ Min I = 10.125 in⁴ (10.34.4.7)

Min. Stiff. Width = $2" + \frac{D}{30}$

$= 2" + \frac{48}{30} = 3.6"$ (10.34.4.10)

Try 5" × 5/16" Stiffeners

$$I = \frac{5^3 \times 5/16}{3} = 13.02 \text{ in}^4 \ \underline{OK}$$

$$\frac{b}{t} = \frac{5}{5/16} = 16 \ \underline{OK}$$

USE 5" × 5/16" SINGLE STIFF.

BEARING STIFFENERS

Design for reaction = 136.16 k

Assume 14" Wide Top Flange

⇒ try 6½" stiffeners

(Use A588 for brg. stiff.)

$$\text{Min. th.} = \frac{6\frac{1}{2}"}{12} \cdot \sqrt{\frac{50 \text{ ksi}}{33}} = .67"$$

Try 3/4" × 6½" Stiff.

(10.34.6.1) A_{web} = (18 × 3/8) × 3/8

= 2.53 □"

A_{stiff} = 2 × 6½ × 3/4 = 9.75

12.28 □"

$$I = (2 \cdot 6\tfrac{1}{2} + \tfrac{3}{8})^3 \times \frac{3/4}{12} = 149.54 \text{ in}^4$$

$$r = \sqrt{I/A} = 3.49"$$

Use L = 48" ; assume K = 1

$$\frac{Kl}{r} = 13.755$$

$$C_c = \sqrt{\frac{2\pi^2 E}{F_y}} = 107.0$$

$$\frac{Kl}{r} < C_c \Rightarrow F_a = 23.54 - .00103\left(\frac{Kl}{r}\right)^2$$

$$= 23.385 \text{ ksi}$$

$$\text{Act } f_a = \frac{136.16k}{12.28 \square''} = 11.09 \text{ ksi } \underline{OK}$$

Note: PLG does not check bearing stiff. design

FLANGES

Desire ≈ 14" min. width of top flange, for shear stud placement, erection stability.

$$\text{Min. th.} = \frac{b}{20} = \frac{14''}{20} = .70''$$

(10.34.2.1.4)

Try 14" × 3/4" top flange

Approximately, bottom flange has moment arms of:

50" ± for NCDL
57" ± for CDL + LL + I

Assume $f_b \approx 26$ ksi

$$\therefore A_{bf} = \frac{12}{26 \text{ ksi}}\left(\frac{1164.38'^k}{50''} + \frac{354.34 + 1431.09'^k}{57''}\right)$$

$$= 25.2 \square''$$

Try 18" × 1½" bottom flange

Beam Properties:

	A	d	Ad	Ad^2	I_o
Top Fl	10.5	.375	3.94	10,620.73	0.49
Web	14.0	24.75	445.5	993.44	3456.0
B Fl	27.0	49.5	1336.5	8100.41	5.06
	55.5		1785.94	19714.58	3461.55
				23,176.13	

$$\bar{y} = \frac{1785.94}{55.5} = 32.18''$$

$$S_{top} = \frac{23,176.13}{32.18} = 720.22 \text{ in}^3$$

$$S_{Bott} = \frac{23,176.13}{50.25'' - 32.18''} = 1282.51 \text{ in}^3$$

Check erection stresses

Bracing @ ¼ pts. = 22.5'

$L_b < 25'$ OK (10.

Top flange governs by inspection

$$\text{Table 10.32.1 A}: F_a = 27{,}000 - 14.4\left(\frac{l}{b}\right)^2$$

$$= 27{,}000 - 14.4\left(\frac{22.5' \times 12}{14''}\right)^2$$

= 21,644 psi (*Note: PLG solves as below)

Check NCDL on bare beam; increase by 10% for constr. loads

$$f_b = \frac{(1.10 \times 1164.38 \times 12)}{720.22} = 21.34 \text{ ksi}$$

OK

Note: decrease in F_b due to web slenderness (sheet 2) does not apply here due to lower shear.

* PLG solves for F_b as follows:

$$F_b = 27{,}000 \times \left[1 - \frac{12\left(\frac{l}{b}\right)^2 \times F_y}{4\pi^2 E}\right]$$

$$= 21{,}737.04 \text{ psi}$$

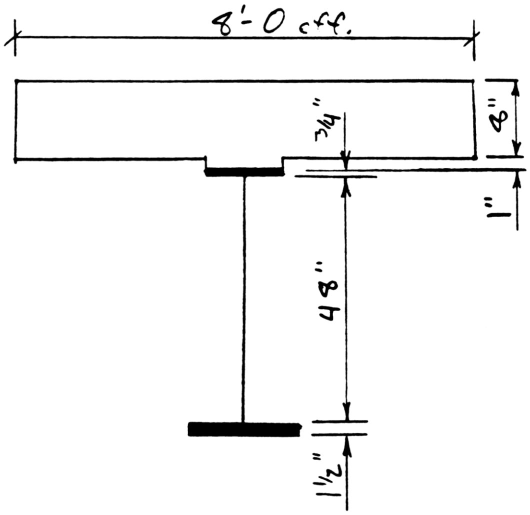
Calculate composite properties

Effective slab width:

Bm Spa. : 9.5'
Span /4 = 90/4 = 22.5
12 × th = 12 × 4" = 4.0' ← Governs

∴ For n = 8, eff. width = $\frac{12 \times 8"}{n=8}$ = 12"

Assume 1" stool



For n = 8

	A	d	Ad	Ad²	I₀
Slab 12" × 8" =	96.0	4"	384.0	17,808.60	512.0
Beam	55.5	41.1791"	2285.44	30,804.05	23,176.13
	151.5□"	17.6201"	2669.44	48,612.65	23,688.13
				72,300.78	

$\bar{y}$ = 17.62" > 8" ⇒ entire slab effective

$S_{slab} = \frac{72,300.78}{17.62"} \times (n=8) = 32,826.57 \ in^3$

$S_{TF} = \frac{72,300.78}{(17.62" - 9")} = 8387.50 \ in^3$

$S_{BF} = \frac{72,300.78}{(59.25 - 17.62")} = 1736.75 \ in^3$

For n = 24 (Per 10.38.1.4)

	A	d	Ad	Ad²	I₀
Slab 4" × 8"	32.0	4"	128.0	17,795.80	170.67
Bm	55.5		2285.44	10,260.64	23,176.13
	87.5	27.5822"	2413.44	28,056.45	23,346.80
				51,403.24	

n = 24:

$S_{slab} = \frac{51,403.24}{27.58"} \times (n=24) = 44,727 \ in^3$

$S_{TF} = \frac{51,403.24}{27.58 - 9"} = 2766.27 \ in^3$

$S_{BF} = \frac{51,403.24}{(59.25 - 27.58)} = 1623.20 \ in^3$

Actual Stresses:

Slab Use S_{s1} for n = 8 (conserv.)

$(354.38 + 1431.09) \times \frac{12}{32,826.57}$

= .65 ksi

(Note: PLG does not calculate composite section properties or stresses thereon. Program COMP can be used for this.)

Stresses on comp. section

Top Flange Since the neutral axis is below the top flange for both n=9 and n=24, stresses are additive.

NCDL	1164.39×12 / 720.22	=	19.40 ksi
CDL	354.36×12 / 2766.27	=	1.54
LL+I	1431.09×12 / 4387.30	=	2.05
			22.99 ksi

Bottom Flange

NCDL	1164.39×12 / 1242.51	=	10.49 ksi
CDL	354.34×12 / 1623.20	=	2.62
LL+I	1431.09×12 / 1736.75	=	9.89
			23.40 ksi

Allowable stresses on Comp. section

From Table 10.32.1.A, F_{tens} = 27 ksi

and for $L/b \approx 0$ (due to slab), F_{comp} = 27 ksi (assuming no splice)

∴ Max F_b = 26.42 ksi (see sh. 2) OK

(Actually this reduction is only true at high shear areas, i.e. the ends. F_b = 27.0 ksi where shear drops below .6 F_v :

.6 × 11.39 = 6.83 ksi

or 6.83 × 48 × 3/4" = 122.90 k.

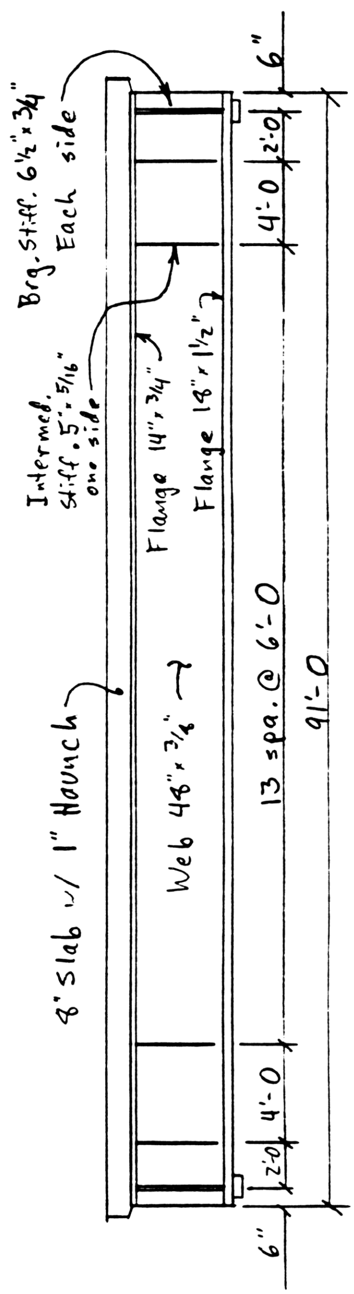
However, erection stresses govern this design so advantage cannot be taken of the full allowable.

∴ USE SECTION AS DESIGNED

Note: Ordinarily the bottom flange would be reduced for the end 20' or so. This example does not consider this reduction.

This example also excludes design of shear connectors; consideration of field splices; and deflection and camber calculations.

LAYOUT



XEQ "PLG"
9:12:59 AM 03/19/88

TITLE:
90 FT SIMPLE SPAN BRIDGE
RUN
****
AASHTO
WSD?
1.0000 RUN — Working stress
F-Y KSI
50.0000 RUN

M1,'K
2,000.0000 RUN — M1 is arbitrary due to composite action
M2,'K
0.0000 RUN — 0 ⇒ check end panel
V, K
136.1600 RUN
LAT BR SPA'
0.0000 RUN — fully braced by slab
MAX D/TW=140.0000

WEB D"
48.0000 RUN
TH"
.3750 RUN
D/TW=128.0000 OK
ACT FV=7.5644 KSI
STIFF REQD

WEB D"
RUN — New web trial? Bypass
****
END PANEL:
MAX ST SPA=24.0000"

ST SPA"
24.0000 RUN — First stiffener space
NO TENS FIELD
ALL FV=16.6667 KSI OK

ST SPA"
RUN — New spacing trial? Bypass

#STIFF
1.0000 RUN — single stiffener
J=8.0000
MIN I=10.1250
****
INT PANEL:
MAX ST SPA=72.0000"

ST SPA"
72.0000 RUN
CV=0.3879
TENS FIELD
ALL FV=11.3883 KSI OK

ST SPA"
RUN

#STIFF
1.0000 RUN
J=0.5000
MIN I=1.8984
****
STIFF W"
4.0000 RUN
TH"
.2500 RUN
STIFF NG — I < Min

STIFF W"
6.0000 RUN
TH"
.3125 RUN
STIFF NG — $\frac{b}{t} > 16$

STIFF W"
5.0000 RUN
TH"
.3125 RUN
STIFF OK — OK

STIFF W"
RUN — Go on
****
COMP FL W"
14.0000 RUN
TH"
.7500 RUN
TENS FL W"
18.0000 RUN
TH"
1.5000 RUN
I=23,176.1267
Y=32.1791"
S-C=720.2240
S-Te=1,282.5077

A=55.5000
----
b/T=18.6667 OK — Check flange $\frac{b}{t}$ and slenderness
ALL L/b=30.0000
L/b OK

```
TENS Fb=26.3961 KSI
COMP Fb=26.3961 KSI
------------------------
COMP FL W"
                   RUN
************************
M1,'K
    1,280.8200     RUN
M2,'K
                   RUN
V, K
       56.9300     RUN
LAT BR SPA'
       22.5000     RUN
MAX D/TW=140.0000
---------
WEB D"
                 XEQ B
------------------------
COMP FL W"
       14.0000     RUN
TH"
         .7500     RUN
TENS FL W"
       18.0000     RUN
TH"
        1.5000     RUN
I=23,176.1267
Y=32.1791"
S-C=720.2240
S-Te=1,282.5077

A=55.5000
---------
b/T=18.6667 OK
ALL L/b=30.0000
L/b OK

TENS Fb=27.0000 KSI
COMP Fb=21.7370 KSI
------------------------
COMP FL W"
```

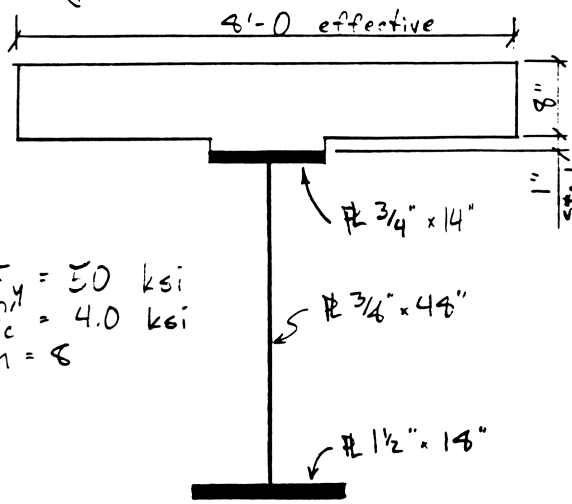
Allowables for fully supported flange - governed by shear reduction

Bypass flange input for new load and support conditions - check erection stresses

Web dimensions and shear properties stay the same - check new flange allowables

New allowable stresses for unbraced Length of 22.5'

Re-analyze this design using LOAD FACTOR DESIGN



F_y = 50 ksi
f'_c = 4.0 ksi
n = 8

Span = 90'-0
Braced @ ¼ Pts. ⇒ L_b = 22'-6

LOADS

Moments:	Serv.		LF
NCDL	1164.38	× 1.3	= 1513.69'k
CDL	354.38	× 1.3	= 460.69'k
LL+I	1431.09	× 1.3 × 5/3	= 3100.70'k

Shear	Serv.		LF
NCDL	51.75	× 1.3	= 67.28 k
CDL	15.75	× 1.3	= 20.48
LL+I	64.66	× 1.3 × 5/3	= 148.76
			236.52 k

For construction, consider 10% increase in NCDL.

LF M = 1.1 × 1513.69 = 1665.06'k
LF V = 1.1 × 67.28 = 74.01 k

Check compactness req'ments (10.48.2.1)

a) $\frac{b'}{t} = \frac{7''}{.75''} = 9.33$

$\frac{2200}{\sqrt{50{,}000}} = 9.84 > \frac{b'}{t}$ OK

b) $\frac{D}{t_w} = \frac{48''}{3/8''} = 128 < 150$ OK

c) $\frac{20{,}000 \cdot (14'' \cdot .75'')}{50\ ksi \times 50.25''} = 83.6''$

$L_b = 22.5' \times 12 = 270'' > 83.6''$

NG during erection

$L_b = 0'' < 83.6''$ OK for service

d) Max. axial comp. = 0 OK

e) $1.015 \times 10^5 \frac{t_w^3}{D} = 1.015 \times 10^5 \frac{(3/8)^3}{48}$

= 111.5 k

Shear = 236.52 k > 111.5 NG

⇒ transverse stiff. req'd

<u>Check stiffened web</u> (10.48.5.1)

$\frac{D}{t_w} = \frac{48''}{3/8} = 128 < 163$ <u>OK</u>

<u>Space first stiffener</u> (10.48.5.3)

Max $= \frac{D}{2} = 24''$; check allowable @ 24"

$v_p = .58\ F_y = 29.0$ ksi

Allow. $v = 1.2\times10^5 \left[1 + \left(\frac{D}{d_1}\right)^2\right] \cdot \frac{t^2}{D^2} = 36.62$ ksi

$36.62 > 29.0 \Rightarrow$ allow. v = <u>29.0 ksi</u>

Act. $v = \frac{236.52\ k}{48 \times .375}$ = <u>13.14 ksi</u> < 29.0 <u>OK</u>

<u>Remaining stiffeners</u>

Max. spa. = 1.5 D = 72" (10.48.5.5)

<u>Check v @ 72"</u> ; $D/d_0 = 2/3$

$C = 18{,}000\left(\frac{t_w}{D}\right)\sqrt{\frac{1 + (D/d_0)^2}{F_y}} - 0.3 < 1.0$

(10.48.5.3)

$= 18{,}000\left(\frac{.375''}{48''}\right)\sqrt{\frac{1 + (.667)^2}{50{,}000}} - 0.3$

$= .4558$

$v_u = v_p\left(C + \frac{.87(1-C)}{\sqrt{1 + (d_0/D)^2}}\right) = 29.0 \times .7184$

$= 20.43$ ksi

20.43 > 13.14 ksi <u>OK</u>

<u>Use 1 spa. @ 24" then spa. @ 72"</u>

<u>Check shear interaction</u> (10.48.5.4)

$\frac{V}{V_u} = \frac{13.14}{20.43} = .63 > .60$

∴ limit bending stress

Note: AISC limits only the <u>tension</u> bending stress, while AASHTO does not make the distinction.

As a simplification, PLG applies this reduction to both tensile and comp. stresses no matter which code is being followed.

$\frac{F_b}{F_y} = 1.375 - .625\frac{V}{V_u}$

$F_b = 50\left(1.375 - .625 \cdot \frac{13.14}{20.43}\right)$

= <u>49.04 ksi</u>

<u>Check bending stresses</u>

<u>Bottom Flange:</u>

NCDL	$1513.69^{ft\text{-}k} \times 12 / 1282.51\ in^3$ =	14.163 ksi
CDL	$460.69^{ft\text{-}k} \times 12 / 1623.20$ =	3.406
LL+I	$3100.70 \times 12 / 1736.75$ =	21.424
		34.993 ksi

34.99 < 49.04 <u>OK</u>

<u>Top Flange:</u>

NCDL	1513.69×12/720.22 =	25.22
CDL	460.69×12/2766.27 =	2.00
LL+I	3100.70×12/8387.50 =	4.44
		31.66 ksi

31.66 < 49.04 <u>OK</u>

Slab:

CDL 460.69 × 12 / 32,426.57 = .164
LL+I 3100.70 × 12 / 32,426.57 = 1.134
1.302

Allow. $f_c = .45f_c' = 3.40$ ksi > 1.302 OK

Check erection stresses

$L_b = 22.5' \cdot 12 = 270'' > 43.6''$

$\therefore F_b = F_y \left[1 - \frac{3F_y}{4\pi^2 E}\left(\frac{L_b}{.9b'}\right)^2\right]$

$= 50 \cdot \left[1 - \frac{3 \cdot 50}{4\pi^2 \cdot 29{,}000}\left(\frac{270''}{.9 \cdot 7''}\right)^2\right]$

= 37.968 ksi (10.48.4.1)

@ Midspan, $\frac{M_1}{M_2} \approx 1 \Rightarrow$ no increase

Act. $f_b = \frac{1665.06^{k \cdot '} \cdot 12}{720.22 \text{ in}^3} = 27.74$ ksi

27.74 < 37.97 OK

Note: Under LFD this section could have been considerably smaller than under WSD and would normally be re-designed. Deflection could become an important consideration.

Bearing Stiffener Design

Use WSD design (10.48.7)

Intermediate Stiff. Design (10.48.5.3)

Min. $A = (.15 B \cdot D \cdot t_w (1-C) \cdot \frac{V}{V_u} - 18 t_w^2) \times Y$

Y = 1.0 (non-hybrid)
B = 2.4 (single plate)
C = .4558

Min $A = (.15 \cdot 2.4 \cdot 46 \cdot \frac{7}{16}(1 - .4558) \times \frac{13.14}{20.43}$
$- 18 \cdot \frac{3}{4}^2)$

= −.3071 (for internal panels)

For end panel, C = 1.0

$\Rightarrow$ Min. A = $-18(\frac{3}{4})^2 = -2.5313$

Both are < 0 $\Rightarrow$ no min. area req'ment (stiffened portion of web alone is adequate)

Min I is calculated as for WSD:

End Panel Min. I = 10.125 in^4
Int. Panel Min I = 1.498 in^4

$\frac{b'}{t_{st}} < \frac{2600}{\sqrt{50{,}000}} = 11.63$

Try 5", 5/16":

$\frac{b'}{t} = \frac{5}{5/16} = 16 > 11.63$ NG

Try 4" × 1/2": $\frac{b'}{t} = 9.14 < 11.63$ OK

$I = \frac{4^3 \times .5}{3} = 10.67 > 10.125$ OK

USE 4" × 1/2" INT. STIFF.

LAYOUT will be the same as WSD except for size of int. stiffeners. As mentioned, the flanges (esp. the tension flange) would most likely be downsized.

AASHTO LFD

11/12

AASHTO LTD

12/12

Program PLG For Plate Girder Analysis ______ bytes Page 1 of 9

X	Y	Z	T	L

```
01♦LBL "PLG"
 02 SF 09
 03 E
 04 "COMP?"
 05 PROMPT
 06 X>0?
 07 CF 09
 08 31
 09 XEQ "SBM"
 10 3
 11 GTO 00

 12♦LBL c
 13 XROM "L*"
 14 4

 15♦LBL 00
 16 FC? 09
 17 XEQ "SBM"
 18 CF 05
 19 RCL 17
 20 RCL 21
 21 +
 22 RCL 26
 23 +
 24 "M1,↑K"
 25 FS? 09
 26 PROMPT
 27 ENTER↑
 28 "M2,↑K"
 29 PROMPT
 30 STO 31
 31 X≠0?
 32 /
 33 X=0?
 34 SF 05
 35 SIGN
 36 E
 37 LASTX
 38 ABS
 39 X<=Y?
 40 X<>Y
 41 /
 42 *
 43 STO 28

 44♦LBL d
 45 CF 07
 46 990
 47 FS? 06
 48 1157
 49 FS? 04
 50 2 E3
 51 RCL 06
 52 SQRT
 53 /
 54 INT
 55 STO 03
 56 "MAX D/TW"
 57 XROM "Y"
 58 SF 14

 59♦LBL 06
 60 XROM "L"
 61 CLX
 62 "WEB D="
 63 PROMPT
 64 X=0?
 65 GTO a
 66 STO 07
 67 "TH="
 68 PROMPT
 69 STO 08
 70 RCL 03
 71 RCL 07
 72 RCL 08
 73 /
 74 STO 10
 75 "D/TW="
 76 ARCL X
 77 CF 07
 78 FC? 00
 79 XEQ 05
 80 X>Y?
 81 GTO 06
 82 XEQ 12
 83 14 E3
 84 RCL 06
 85 16.5
 86 +
 87 RCL 06
 88 *
 89 SQRT
 90 /
 91 260
 92 X>Y?
 93 X<>Y
 94 FC? 04
 95 150
 96 RCL 10
 97 X>Y?
 98 SF 07
 99 XEQ 01
100 FS? 10
101 .4
102 FS? 10
103 83149
104 RCL 10
105 X↑2
106 /
107 XEQ 08
108 CLA
```

Handwritten notes: line 03 — "default response"; line 09 — "Input subroutine"; line 17 — "Slab dimension and LL input subroutine"; line 81 — "adv"; line 106 — "X<>Y, rcl 06, *, X>Y?, X<>Y"

Program PLG For ____________ ____________ ________ bytes Page 2 of 9

Step	X	Y	Z	T	L
109 FC? 07					
110 "NO "					
111 "⊢STIFF REQD"					
112 AVIEW					
113 GTO 06					
114♦LBL 12					
115 CF 07					
116 "V, K"					
117 PROMPT	V				
118 RCL 07	d				
119 /					
120 RCL 08	t_w				
121 /	f_v				
122 "ACT FV"					
123 STO 22					
124 XROM "KS"					
125 XEQ "PV"					
126 RTN					
127♦LBL e					
128 XROM "LL"					
129 XEQ 12					
130♦LBL a					
131 XEQ 01					
132 X<>Y					
133 1/X					
134 FS? 10					
135 2.89					
136 STO 04	all. F_v/F_y				
137 XROM "L*"					
138 "MAX INT"					
139 FS? 05					
140 "MAX END"					
141 260					
142 RCL 07	d				
143 RCL 08	t_w				
144 /					
145 STO 10	d/t_w				
146 /					
147 X↑2	$(260/d/t_w)^2$				
148 3	3	$(\ )^2$			
149 FC? 05					
150 GTO 03					
151 /					
152 ST+ X	$\frac{45000}{(d/t_w)^2}$				
153 5					
154 RCL 04	2.89				
155 RCL 22	act f_v	2.89	5	$\frac{45{,}000}{(d/t)^2}$	
156 R↑					
157 /					
158 *					
159 5					
160 -					
161 /	min. spa.				

Step	X	Y	Z	T	L
162 X<0?					
163 CLX					
164 SQRT	min. spa				
165 X=0?					
166 3					
167 ENTER↑	min.	min.			
168 FC? 04					
169 1.5	1.5 ?	min			
170♦LBL 03					
171 X>Y?					
172 X<>Y	min.				
173 RCL 07	d				
174 *	min spa				
175 "⊢ ST SPA="					
176 STO 00	min.				
177 ARCL X					
178 "⊢"					
179 FC? 00					
180 XEQ "PV"					
181 CF 22					
182 E	1				
183 STO 29	redux				
184♦LBL 21					
185 CF 07					
186 XROM "L"					
187 CLX					
188 "STIFF SPA"					
189 PROMPT					
190 FS? 22					
191 X=0?					
192 GTO 17					
193 STO 11	stiff spa				
194 CF 20					
195 FS? 05					
196 SF 20					
197 RCL 00					
198 X<>Y	spa.	min			
199 X>Y?					
200 XEQ 05					
201 RCL 07	d				
202 ST/ 11					
203 5	5				
204 ENTER↑	5				
205 FS? 10					
206 5.34	5.34				
207 FS? 10					
208 4	4	5.34			
209 RCL 11	a/d				
210 E	1	a/d	4	5.34	
211 X>Y?					
212 XEQ "ZT"	1	a/d	5.34	4	
213 RDN					
214 X↑2	$(a/d)^2$	5.34	4		
215 /					
216 +	K				
217 "K"					

Program PLG For ________ ________ ________ bytes Page 3 of 9

X	Y	Z	T	L

```
218 XROM "Y"
219 .8
220 X<>Y
221 RCL 06
222 /
223 36 E3
224 *
225 1.03
226 FS? 04
227 FC? 06
228 SIGN
229 /
230 SQRT
231 RCL 10
232 /
233 X>Y?
234 ENTER↑
235 X↑2
236 X<>Y
237 /
238 SIGN
239 LASTX
240 X>Y?
241 SF 20
242 "CV"
243 XEQ "Y"
244 SIGN
245 FS? 10
246 1.156
247 LASTX
248 X>Y?
249 X<>Y
250 STO 01
251 ENTER↑
252 SIGN
253 LASTX
254 -
255 .87
256 *
257 RCL 11
258 X↑2
259  E
260 +
261 SQRT
262 /
263 X<0?
264 CLX
265 X=0?
266 SF 20
267 X<>Y
268 RCL 06
269 RCL 04
270 /
271 ST* Z
272 *
273 RCL 22
274 X<Y?
275 FC? 04
276 X<0?
277 SF 20
278 RDN
279 FC? 20
280 +
281 CLA
282 FS? 20
283 "NO "
284 "├TENS FIELD"
285 FC? 00
286 AVIEW
287 XEQ 08
288 GTO 21

     LIST 122

289♦LBL 01
290 3
291 1/X
292 733 E2
293 FS? 06
294 .58
295 FS? 06
296 127 E3
297 FS? 04
298 .54
299 FS? 04
300 1188 E2
301 RTN

302♦LBL 08
303 STO 09
304 "ALL FV"
305 XROM "KS"
306 RCL 22
307 XEQ 05
308 X>Y?
309 RTN
310 FS? 04
311 FC? 20
312 X<0?
313 RTN
314 11
315 "REDX"
316 5
317 R↑
318 /
319 R↑
320 *
321 -
322 8
323 /
324  E
325 X>Y?
326 X<>Y
327 STO 29
```

Handwritten notes: step 220: X = .4, Y = K; step 221: X = .4/Fy, Y = K; steps 310–312: "FS? 04 and FS? 20"; step 320: "* 1.375 − .625 fv/Fv"; step 303: all. fv; step 306: act fv; step 313: act fv, allow. Fv; step 314: 11; step 316: 5, 11, act, all Fv; step 317: Fv; step 318: 5/Fv, 11, act, act; step 319: act fv; step 320: 5fv/Fv, 11; step 323: *; step 324: 1; step 326: reduction

```
328♦LBL "Y"
329 FS? 00
330 RTN
331 "⊢="
332 ARCL X

333♦LBL "PY"
334 SF 25
335 PRA
336 FC?C 25
337 AVIEW
338 RTN

339♦LBL 05
340 X>Y?
341 SF 07
342 XROM "OK"
343 XEQ "PY"
344 RTN

345♦LBL 17
346 FS?C 05
347 GTO a
348 SF 25
349 FS? 22
350 XEQ "STIF"
351 CF 25

352♦LBL b
353 XROM "L*"
354 CLX
355 "Lb'"
356 FS? 09
357 PROMPT
358 12
359 *
360 STO 30
361 CLX
362 XEQ "SBM"
363 X=0?
364 GTO c
365 RCL 00
366 CF 19
367 RCL 05
368 ST+ X
369 X≠Y?
370 SF 19
371 RCL 03
372 3
373 Y↑X
374 RCL 04
375 *
376 RCL 10
377 RCL 09
378 3
379 Y↑X
380 *
381 +
382 12
383 /
384 RCL 01
385 /
386 SQRT
387 STO 23
388 "RY"
389 XEQ "Y"
390 FC? 04
391 GTO 13
392 RCL 08
393 RCL 07
394 *
395 RCL 03
396 /
397 RCL 04
398 /
399 6
400 +
401 ST+ X
402 SQRT
403 RCL 03
404 /
405 1/X
406 STO 09
407 "RT"
408 XROM "Y"

409♦LBL 13
410 ADV
411 XEQ 11
412 RCL 07
413 2 E3
414 RCL 12
415 FS? 06
416 GTO 02
417 760
418 XEQ 00
419 FS? 04
420 AVIEW
421 FS? 04
422 SIGN
423 *
424 *
425 "COMP"
426 XEQ 09
427 "TENS"
428 RCL 18
429 RCL 29
430 *
431 XEQ 09
432 CF 18
433 FC? 09
434 GTO 13
435 RCL 14
436 *
```

```
437 X<>Y
438 RCL 11
439 *
440 X>Y?
441 X<>Y
442 GTO 07

443♦LBL 02
444 RCL 11
445 /
446 970
447 XEQ 00
448 FS? 04
449 AVIEW
450 *
451 *
452 .9
453 FC? 04
454 SIGN
455 FC? 09
456 GTO 02
457 *

458♦LBL 07
459 CLA
460 XROM "M="
461 AVIEW
462 GTO b

463♦LBL 02
464 RCL 06
465 *
466 STO 12
467 RCL 05
468 RCL 04
469 -
470 5 E2
471 RCL 06
472 395
473 XEQ 00
474 LASTX
475 "Rb"
476 FC? 04
477 XEQ "Y"
478 *
479 *
480 "ALL"
481 FC? 18
482 XEQ 09

483♦LBL 13
484 CF 05
485 XROM "L"
486 E
487 XEQ "SBM"
488 GTO b
```

```
489♦LBL 00
490 X<>Y
491 SQRT
492 /
493 RCL 08
494 *
495 R↑
496 -
497 LASTX
498 *
499 RCL 03
500 /
501 RCL 04
502 /
503 X>0?
504 CLX
505 X<>Y
506 /
507 E
508 +
509 "R-PG="
510 ARCL X
511 FC? 04
512 SIGN
513 RCL 12
514 RCL 29
515 RTN

516♦LBL 11
517 CLX
518 SF 18
519 FS? 04
520 FS? 20
521 X>0?
522 CF 18
523 CF 07
524 FC? 04
525 FS? 06
526 X>0?
527 GTO 10
528 .66
529 FC? 10
530 RCL 12        Z
531 RCL 06        Fy
532 *
533 STO 12        Mp
534 STO 18
535 STO 19
536 "PL"
537 XROM "M="
538 FS? 06
539 FC? 18
540 X=0?
541 AVIEW
542 ADV
543 RCL 11        ST
```

Program PLG For ______ ______ bytes Page 6 of 9

	X	Y	Z	T	L

```
544 FS? 10
545 .6
546 RCL 06
547 *
548 RCL 12
549 X>Y?
550 X<>Y
551 STO 10
552 FC? 20
553 STO 12
554 608
555 RCL 05
556 RCL 04
557 -
558 ST+ X
559 RCL 08
560 /
561 RCL 06
562 SQRT
563 *
564 X>Y?
565 CF 18
566 150
567 LASTX
568 *
569 970
570 FS? 04
571 X<>Y
572 RDN
573 FS? 04
574 X>Y?
575 X=0?
576 SF 07
577 FC? 06
578 CF 07
579 XEQ 07
580 RCL 10
581 X<=Y?
582 X<>Y
583 FC? 08
584 FC? 06
585 X<0?
586 CF 18
587 FC? 18
588 STO 12
589 FC? 09
590 RTN
591 "D/TW"
592 RCL 12
593 FC? 07
594 XEQ 14
595 SF 18
596 65
597 RCL 03
598 RCL 04
599 /
600 2
```

F_y M_R M_R .95×640 $\bar{Y}$ T Fl.Th h_c t_w w/t

If tens. field action, no plastic moment

```
601 /
602 RCL 06
603 SQRT
604 *
605 X>Y?
606 CF 18
607 FC? 07
608 X>Y?
609 X=0?
610 GTO 02
611 RCL 07
612 RCL 08
613 /
614 70
615 X>Y?
616 GTO 11
617 RDN
618 .46
619 CHS
620 Y↑X
621 4.05
622 *
623 "Kc"
624 FS? 10
625 XEQ "Y"
626 FC? 10
627 SIGN
628 SQRT
629 *
630 R↑
631 R↑

632♦LBL 11
633 R↑
634 R↑
635 FC? 06
636 95
637 FS? 06
638 70
639 FS? 06
640 FC? 04
641 GTO 03
642 RDN
643 FC? 07
644 GTO 04
645 RCL 10
646 STO 19
647 2
648 /
649 STO 10
650 RDN
651 150
652 GTO 03

653♦LBL 04
654 RCL 06
655 16.5
```

Step	X	Y	Z	T	L
656 -					
657 RCL 11					
658 X<>Y					
659 *					
660 STO 10					
661 RDN					
662 RCL 06					
663 LASTX					
664 /					
665 SQRT					
666 106					
667 *					
668♦LBL 03					
669 X<=Y?					
670 GTO 03					
671 XEQ 07					
672 GTO 02					
673♦LBL 03					
674 FC? 06					
675 GTO 11					
676 X<>Y					
677 /					
678 X↑2					
679 RCL 10					
680 *					
681 GTO 02					
682♦LBL 11					
683 1.415					
684 R↑					
685 229					
686 /					
687 -					
688 *					
689♦LBL 02					
690 "b/2T"					
691 XEQ 14					
692♦LBL "SFb"					
693 SF 18					
694 RCL 10	M_R				
695 RCL 14	S_{bP}				
696 RCL 06	F_y				
697 *					
698 X<=Y?					
699 STO 10					
700 2 E4	20000				
701 RCL 03					
702 *					
703 RCL 04					
704 *					
705 RCL 00	d				
706 /					
707 RCL 06					
708 /	$\frac{20000 A_f}{d F_y}$				
709 FS? 10					
710 GTO 02					
711 FC? 04					
712 GTO 03					
713 RCL 09	r_T				
714 FC? 07					
715 RCL 23	r_Y				
716 3 E2	300				
717 RCL 06	F_y	300	$r_{Y or T}$		
718 SQRT					
719 /					
720 *					
721 STO a					
722 RCL 30	L_b''				
723 X<=Y?					
724 GTO 05					
725 CF 18					
726 "X1"					
727 FC? 07					
728 PROMPT					
729 RCL 23					
730 *					
731 STO 23					
732 "X2"					
733 FC? 07					
734 PROMPT					
735 STO \					
736 RCL 06	F_y				
737 16.5					
738 -	$F_y - 16.5$				
739 ST/ Z					
740 X↑2					
741 *					
742 E					
743 +					
744 SQRT					
745 E					
746 +					
747 SQRT					
748 *					
749 756					
750 RCL 06					
751 SQRT					
752 /					
753 RCL 09					
754 /					
755 FC? 07					
756 X<>Y	λ_R				
757 RCL 30	L_b				
758 X>Y?					
759 GTO 04					
760 RCL a	λ_p	L_b	λ_r		
761 X<> Z	λ_r	λ	λ_f		
762 XEQ 07					
763 XEQ 08					
764 X>Y?					
765 X<>Y					
766 GTO 05					

X	Y	Z	T	L

```
767♦LBL 04
768 RCL 23
769 X<>Y
770 /
771 ENTER↑
772 X↑2
773 RCL \
774 *
775 2
776 +
777 SQRT
778 *
779 286 E3
780 RCL 30
781 RCL 09
782 /
783 X↑2
784 /
785 FC? 07
786 X<>Y
787 RCL 23
788 *
789 XEQ 08
790 GTO 05

791♦LBL 02
792 RCL 03
793 76
794 *
795 RCL 06
796 SQRT
797 /
798 X>Y?
799 X<>Y
800 RCL 30
801 X<=Y?
802 GTO 05
803 CF 18
804 RCL 10
805 RCL 12
806 X>Y?
807 X<>Y
808 STO 18
809 RCL 09
810 R↑
811 /
812 51 E4
813 XEQ 08
814 RCL 06
815 /
816 X<>Y
817 X↑2
818 *
819 SIGN
820 LASTX
821 X<=Y?
822 GTO 02
```

X	Y	Z	T	L

```
823 /
824 2
825 -
826 CHS

827♦LBL 02
828 RCL 06
829 *
830 3
831 /
832 RCL 10
833 X>Y?
834 X<>Y
835 GTO 05

836♦LBL 03
837 FC? 06
838 GTO 10
839 36
840 RCL 31
841 RCL 19
842 -
843 22
844 *
845 -
846  E2
847 *
848 RCL 06
849 /
850 RCL 23
851 *
852 RCL 30
853 X<=Y?
854 GTO 05
855 CF 18
856 R↑
857 X>Y?
858 GTO 04
859 XEQ 07
860 GTO 05

861♦LBL 07
862 RCL 19
863 X<> T
864 ST- Z
865 -
866 /
867 X<>Y
868 RCL 10
869 -
870 *
871 -
872 RTN

873♦LBL 04
874 XEQ 03
875 *
876 RCL 28
```

X	Y	Z	T

```
877 .7
878 X>Y?
879 1.2
880 X<=Y?
881  E
882 XEQ "YT"
883 XEQ 02
884 RCL 06
885 X>Y?
886 X<>Y
887 CLA
888 XEQ 09
889 RCL 14
890 *

891♦LBL 05
892 "Lb"

893♦LBL 14
894 FS? 18
895 RCL 19
896 FC? 06
897 XEQ 09
898 FC? 06
899 GTO 14
900 XROM "M="
901 LASTX
902 *
903 FC? 00
904 XEQ "PV"

905♦LBL 14
906 RCL 12
907 X>Y?
908 X<>Y
909 STO 12        Mmax
910 RTN

911♦LBL 08
912 3.5
913 RCL 28
914 *
915 CHS
916 LASTX
917 X↑2
918 +
919 .3
920 *
921 1.75
922 +
923 2.3
924 X<=Y?
925 X<>Y
926 RDN           Cb

927♦LBL 02
928 "Cb"
929 XEQ "V"
930 *
931 RTN
```

X	Y	Z	T	L

```
932♦LBL 09
933 "⊢ Fb"
934 XROM "KS"
935 XEQ "PV"
936 RTN

937♦LBL 10
938 CLX
939 STO 12
940 STO 18
941 139
942 RCL 06
943 SQRT
944 /
945 RCL 03
946 RCL 04
947 /
948 "b/T="
949 ARCL X
950 XEQ 05
951 X>Y?
952 RTN
953 219
954 RCL 06
955 SQRT
956 /
957 INT
958 "ALL L/b"
959 XEQ "V"

960♦LBL 03
961 RCL 30
962 RCL 03
963 /
964 .9
965 FS? 06
966 FC? 19
967 SIGN
968 /
969 FS? 06
970 ENTER↑
971 "L/b="
972 ARCL X
973 AVIEW
974 X>Y?
975 RTN
976 X↑2
977 95406
978 /
979 RCL 06
980 *
981 SIGN
982 LASTX
983 -
984 RCL 06
985 FS? 06
986 RTN
987 .55
988 *
989 STO 18
990 *
991 STO 12
```

TIMBER BEAM DESIGN

xeq WBM for wood beam analysis (SIZE 012)

SOFTKEYS (USER mode ON)

D	New dimensions
E	New LDF
F	New material properties (skips dimension and load/LDF prompts to re-evaluate same trial with new allowable stresses)

USER FLAGS

Flag 00	clear	full output
	set	short format
Flag 04	clear	AASHTO
	set	NDS/UBC

INPUT SUMMARY

Prompt	Input	Default
F-V PSI	Allowable horizontal shear stress	No default
Fb PSI	Allowable bending stress	No default
E PSI	Modulus of elasticity	No default
LDF	Load Duration Factor	1.0
Le-b"	Effective unbraced length in the "B" direction, in inches	Previous Le-b
B"	B dimension, inches. Enter zero for round.	Previous B
D"	D dimension, inches. Enter zero for round members.	Previous D
DIA" (round only)	Diameter, inches	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

(C-K)	Slenderness limit, dependent on E and F_b values
(S)	Section modulus in inches cubed
(C-S)	Effective length (slenderness) constant, $L_e\ d/b^2$
(Fb')	Allowable bending stress in psi
ALL M	Allowable moment on the section
ALL V	Allowable shear on the section

PROGRAM FLAGS

Flag 06	clear	check slenderness
	set	no slenderness check necessary
Flag 07	clear	rectangular
	set	round
Flag 08	clear	new material properties input via softkey "E"; use same loads and dim.
	set	normal input sequence
Flag 09	clear	column analysis (WCOL)
	set	beam analysis (WBM)

STORAGE REGISTER USE

Register	WBM
00	S (in3)
01	Fb
02	Le (inches stored)
03	B or Diameter, in.
04	D, inches
05	LDF
06	Area
07	E, psi
08	C-F (form)
09	F-V, psi
10	C-K
11	not used (column effective length)

D206

TIMBER BEAM ANALYSIS

Program WBM analyzes the strength of timber beams. It calculates and displays the section properties and allowable stresses, as well as all factors and coefficients used to figure these properties.

MEMBER SECTION

The program is intended for use with solid or glue-laminated (glulam) members only; the analysis of built-up members (e.g. nail-laminated or spaced members) or any solid shape other than rectangular or round, is not covered.

The program is not able to make allowance for bolt holes or notches; a uniform section is assumed. However, in many cases (e.g. slender members with bolt holes at the ends) the allowable stress may still be used. This is a matter requiring engineering judgement. Deflection is not calculated or checked.

FLAGS

WBM will analyze by AASHTO (1988) if flag 04 is clear, and by NDS (National Design Specification for Wood Construction, 1986, upon which AITC and UBC base their specs) if flag 04 is set. As with other structural programs, the code being followed is displayed at the beginning of each run. The differences between the two codes are minor; C_k and the slender beam stress formula for bending are slightly different between the codes. Also, of course, some of the allowable stresses are different for different species and grades of timber. Actually, since AASHTO bases its recommendations on the NDS, any differences merely reflect AASHTO's slowness in changing.

As with other programs in this package, setting flag 00 disables the output of intermediate calculations and constants (as well as the title block for printed output), shortening the output.

DIMENSIONS

The dimension prompts for B" and D" (B and D, in inches) assume D is the dimension in the direction of applied side load (or eccentricity for column design). Thus if a member is being used in weak-axis bending, B will be greater than D. For columns under pure axial load, B and D are interchangeable as long as the effective unbraced length and the member dimension correlate with each other in each direction. After being input once, the prompts for Le(b), Le(d), B, and D all default to their previous values. See the "Input Summary" section on the front sheet of this documentation.

ALLOWABLE STRESSES

Engineers using the NDS should note that tabulated allowable stresses have decreased in recent editions; an attempt should be made to keep up to date. The higher equation constants allowed for machine-stress-rated (MSR or E-rated) timber are not implemented in these programs.

Factors affecting allowable stresses are input and used in different ways. The following factors are the most common.

CUF	Condition-of-use or wet-use factor
C-FR	0.9 for fire-retardant treated wood
C-FU	Flat-use factor

The above three factors must be chosen by the designer and applied to the tabulated allowable bending stress (Fb) values <u>before</u> input; CUF and C-FR must be applied to the tabulated Fv and E values as well. (CUF for these is 1.0 for sawn timber but not for glulam.)

C-F	Form factor (1.0 for rect. and 1.18 for round members)
C-f	Size factor for deep beams

C-F and C-f are calculated <u>by the program</u> and applied to the allowable bending stress input. C-f is calculated by the basic exponential formula; no adjustment is made for different span/depth or loading conditions. A note of caution: tabulated stresses for poles or piles (NDS 6.2, UBC ch. 25) already have the form factor (1.18) included. If these values are used they must be reduced by 1.18 before input.

LDF	Load duration factor

The LDF is input by the user. It is applied where appropriate by the program (e.g. beams or columns which have a "short" or "intermediate" effective length) and must not be included in the allowable stresses input.

C-S	Slenderness factor

The slenderness factor is calculated and applied by the program. It is used in conjunction with the CUF, C-FR, and C-FU factors (obviously, since the allowable stresses are multiplied by them before input); C-F and C-f are also used in conjunction with these factors. However, C-S is not used with C-F and C-f. The allowable bending stress is calculated based on both

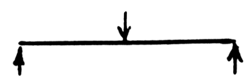
D208

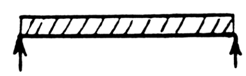
slenderness (C-S) and size (C-F x C-f). The lesser of these, multiplied by the LDF, is the allowable stress which is output. This will hopefully be made more clear in the examples: Fb_0 is the (user-modified) tabulated allowable stress to be input. Fb_1 is the calculated allowable stress involving size and form. Fb_2 is the calculated allowable stress involving slenderness. And finally, Fb is the least of these two, multiplied by the LDF, which is output.

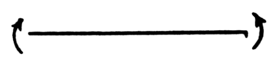
WBM is applicable to round, sawn, or (rectangular) glulam beams. However, for heavily cambered or curved glulam beams the user must modify the allowable stress by the curvature factor.

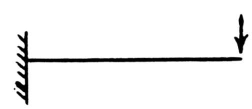
EFFECTIVE LENGTH

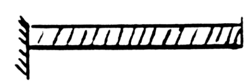
For rectangular beams which are as wide or wider than they are deep, and for round members used as beams, no reduction for slenderness is made; i.e. C-S = 1.0. For deep rectangular beams which are not fully braced, C-S depends on the effective length, which in turn is a multiple of the unbraced length depending on the loading condition. For those cases where the unbraced length is not zero, the following multipliers apply:

$1.37\ L_u + 3D$ 

$1.63\ L_u + 3D$ 

$1.84\ L_u$ 

$1.44\ L_u + 3D$ 

$0.90\ L_u + 3D$ 

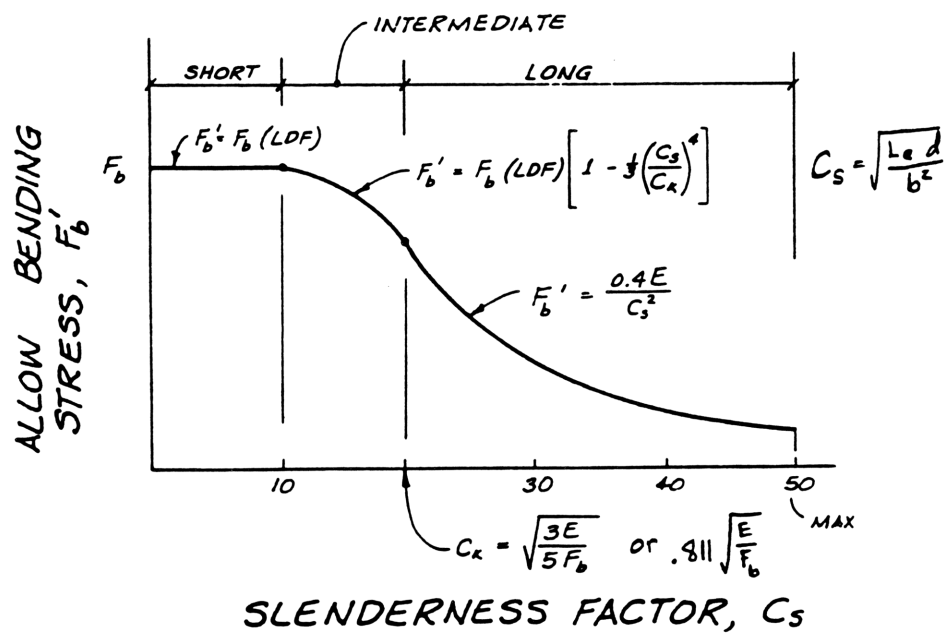
For C-S greater than 50 the program calculates an allowable stress of zero. See the next page for a graph showing the effect of unbraced length on allowable bending stresses.

SHEAR

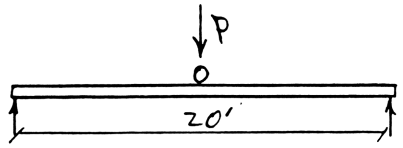
Shear analysis is a little simpler: the only factors which pertain are the CUF and C-FR, which the user must factor into the allowable stress input; and the LDF, which the user inputs directly. The actual shear stress compared is VQ/Ib, which amounts to the average shear stress times 3/2 for rectangular members and 4/3 for round beams.

ALLOWABLE BENDING STRESS considering EFFECTIVE LENGTH

The effect of unbraced length is shown graphically by the following illustration, taken by permission from Design of Wood Structures by Donald E. Breyer (McGraw-Hill, 1980)



D210



Design a sawn S4S beam to support a movable equipment hoist.

$$\text{Crane DL} = 2.90\ k$$
$$\text{Max. Expected LL} = \underline{1.20\ k}$$
$$P = 4.10\ k$$

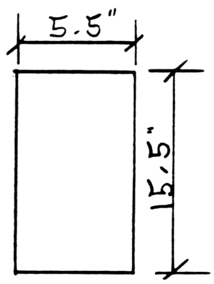
$$M_p = 4.10\,k \times \frac{20'}{4} = 20.50^{'\cdot k}$$

Beam requires fire-retardant treatment.

$$\therefore C_{FR} = 0.90$$

CUF = 1.0 (Indoor use)

TRY 6" × 16" – #1 Hem-Fir ("B+S")



NDS: $F_b = 1050\ psi \times C_{FR} \times CUF$
$F_v = 70\ psi \times C_{FR}$
$E = 1{,}300{,}000\ psi \times C_{FR}$

$$S = \frac{5.5 \times (15.5)^2}{6} = 220.229\ in^3$$

Beam DL (use 50 pcf.):

$$\frac{5.5'' \times 15.5''}{144} \times 50\ pcf \times \frac{(20')^2}{8} / 1000 = 1.48^{'\cdot k}$$

Check loading combinations

$$DL: \left(1.48^{'\cdot k} + \frac{2.90 \times 20'}{4}\right) / (LDF = 1.0) = 15.98^{'\cdot k}$$

$$LL: (1.48^{'\cdot k} + 20.50^{'\cdot k}) / 1.33 = 16.53^{'\cdot k}$$

∴ Governing load is LL:

$$DL + LL = 21.98^{'\cdot k}$$

$$\text{Shear} = (29.6\ ^{lb}/_{ft} \times 10') + \frac{(20' - 15.5'')}{20'} \times 4.10\ k = 4.13\ k$$

$$F_{b \cdot 0} = 1050 \times \overset{C_{FR}}{0.9} \times \overset{CUF}{1.0} = 945\ psi$$

$$F_{v \cdot 0} = 70 \times 0.9 \times 1.0 = 63\ psi$$

$$E_0 = 1{,}300{,}000 \times 0.9 \times 1.0 = 1{,}170{,}000$$

rect. beam ⇒ Form Factor $C_F = 1.0$

Size Factor $C_f = \left(\frac{12}{d}\right)^{1/9} = \left(\frac{12}{15.5}\right)^{1/9} = .9720$

$$F_{b \cdot 1} = 945 \times \overset{LDF}{1.33} \times .9720 = \underline{1221.7\ psi}$$

$L_u = 20.0'$ (unbraced); $L_{eff} = 1.61 \times 20' = 32.20'$

$$C_s = \sqrt{\frac{L_e d}{b^2}} = \sqrt{\frac{32.20' \times 12 \times 15.5}{(5.5)^2}} = 14.071 > 10 \Rightarrow \text{not "Short"}$$

$$C_K = .811\sqrt{\frac{E}{F_b}} = .811\sqrt{\frac{1{,}170{,}000}{945}} = 28.54$$

14.071 < 28.54 ⇒ "Intermediate"

$$\therefore F_{b \cdot 2} = 945 \times \overset{LDF}{1.33} \times \left[1 - \frac{1}{3}\left(\frac{C_s}{C_K}\right)^4\right] = \underline{1232.1\ psi}$$

$F_{b \cdot 1}$ Governs @ 1221.7 psi

Sawn Beam Design

Moment Capacity

$M = F_b \cdot S = \frac{1221.7 \text{ psi} \times 220.229 \text{ in}^3}{12{,}000}$

$= \underline{22.42'^{k}} > 21.94'^{k}$ OK

Shear Capacity

Shear stress $= \frac{3}{2} \times \frac{4.13 \text{ k}}{5.5'' \times 15.5''} = .0727$ ksi

Allowable stress $= .063 \text{ ksi} \times 1.33$ (LDF)

$= .0838$ ksi - OK

Allowable shear $= \frac{.0838}{.0727} \times 4.13 = \underline{4.76 \text{ k}}$

Short Form - Set Flag 00

no title block

```
            SF 00
       XROM "WBM"
F-V PSI
      63.0000    RUN
Fb PSI
     945.0000    RUN
E PSI
 1,170,000.000   RUN
----------
LDF
       1.3300    RUN
-------------------------
Le-b"
     386.4000    RUN
B"
       5.5000    RUN
D"
      15.5000    RUN

ALL M=22.4196 'K
ALL V=4.7621 K
-------------------------
Le-b"
```

no intermediate calculations

Long Form

← NDS

```
                SF 04
            XEQ "WBM"
SIZE<12

             SIZE 012

            XEQ "WBM"
 6:11:14 PM  11/02/88

TITLE:
CRANE RAIL         RUN
************************
NDS
F-V PSI
        63.0000    RUN
Fb PSI
       945.0000    RUN
E PSI
   1,170,000.000   RUN
----------
LDF
         1.3300    RUN
C-K=28.5363
-------------------------
Le-b"
       386.4000    RUN
B"
         5.5000    RUN
D"
        15.5000    RUN
S=220.2292
C-S=14.0709
Fb'=1,221.6123 PSI

ALL M=22.4196 'K
ALL V=4.7621 K
-------------------------
Le-b"
```

Use Adjusted values

D212

TIMBER COLUMN and BEAM-COLUMN DESIGN

xeq WCOL for wood column and beam-column analysis (SIZE 019)

SOFTKEYS (USER mode ON)

- D New dimensions
- E New loads and LDF
- F New material properties (skips dimension and load/LDF prompts to re-evaluate same trial with new allowable stresses)

USER FLAGS

Flag 00	clear	full output
	set	short format
Flag 02	clear	BEEP disabled
	set	BEEP enabled
Flag 04	clear	AASHTO
	set	NDS

INPUT SUMMARY

Prompt	Input	Default
Fc PSI	Allowable compressive stress	No default
Fb PSI	Allow. bending stress	No default
E PSI	Modulus of elasticity	No default
P, K	Axial load, kips	No default
ECCd"	Eccentricity of axial load in the "D" direction, in inches	0
LATd M,'K	Moment due to lateral load, in the "D" direction, in foot-kips	No default
LDF	Load Duration Factor	1.0
Le-b"	Effective unbraced length in the "B" direction, in inches	Previous Le-b

DZ13

Prompt	Input	Default
Le-d"	Effective unbraced length in the "D" direction, in inches	Previous Le-d
B"	B dimension, inches. Zero for round members.	Previous B
D"	D dimension, inches. Zero for round members.	Previous D
DIA" (round only)	Diameter, inches	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear. Asterisked items are displayed only if the member is in bending.)

(NDS or AASHTO)	A reminder of the code used. Press R/S to continue.
(C-K) *	Bending slenderness limit, based on E and F_b
(K)	Slenderness limit, based on E, F_c, and LDF
(S) *	Section modulus
(C-S) *	Effective length (slenderness) constant
(Fb') *	Allowable bending stress in psi
(LONG, INTRM, or SHORT)	The length category of the column in each direction
(Fc'd and Fc'b)	Allowable axial compressive stresses in the D and B axes, respectively, in psi
ALL P	The allowable axial load in kips, if no bending stress is present
(J) *	Interaction (P-delta) constant
INTRXN-b or -d (*)	The governing interaction value. Must be less than unity to be acceptable.

PROGRAM FLAGS

Flag 06	clear	check slenderness
	set	no slenderness check necessary
Flag 07	clear	rectangular
	set	round

D214

PROGRAM FLAGS

Flag 08	clear	new material properties input via softkey "E"; use same loads and dim.
	set	normal input sequence
Flag 09	clear	column analysis (WCOL)
	set	beam analysis (WBM)
Flag 10	clear	beam-column
	set	pure axial load

STORAGE REGISTER USE

Register	WCOL	WFc
00	S (in3)	
01	Fb	
02	Lb (inches stored)	
03	B or Diameter, in.	
04	D, inches	
05	LDF	LDF
06	Area	
07	E, psi	E, psi
08	C-F (form)	
09	Fc, psi	Fc, psi
10	C-K	
11	Ld (inches stored);	
12	P, kips	
13	M (inch-kips stored)	
14	ECC(d), inches	
15	Axial stress due to P	Length Category (SHORT, INTRM, LONG)
16	Fc'(b), psi	
17	Fb', psi	
18	K	K

DZ15

D216

TIMBER COLUMN ANALYSIS

Program WCOL analyzes the properties and strength of timber columns and beam-columns. It calculates and displays the section properties and allowable stresses, as well as all factors and coefficients calculated to arrive at these properties.

MEMBER SECTION

The program is intended for use with solid or glue-laminated (glulam) members only; the analysis of built-up members (e.g. nail-laminated or spaced members) or any solid shape other than rectangular or round, is not covered.

The program is not able to make allowance for bolt holes or notches; a uniform section is assumed. However, in many cases (e.g. slender members with bolt holes at the ends) the allowable stress may still be used. This is a matter requiring engineering judgement.

FLAGS

The program will analyze by AASHTO (1988) if flag 04 is clear, and by NDS (National Design Specification for Wood Construction, 1986, upon which AITC and UBC base their specs) if flag 04 is set. As with other structural programs, the code being followed is displayed at the beginning of each run. The differences between the two codes are minor. Also, of course, some of the allowable stresses are different for different species and grades of timber. Actually, since AASHTO bases its recommendations on the NDS, any differences merely reflect AASHTO's slowness in changing.

Beam-column interaction analysis is based entirely on the NDS no matter what the status of flag 04. However, the allowable stresses in bending and compression are figured using the appropriate code.

As with other programs in this package, setting flag 00 disables the output of intermediate calculations and constants (as well as the title block for printed output), shortening the output.

DIMENSIONS

The dimension prompts for B" and D" (B and D, in inches) assume D is the dimension in the direction of applied side load or eccentricity for beam-column design. Thus if a member is being used in weak-axis bending, B will be greater than D. For columns under

D217

pure axial load, B and D are interchangeable as long as the effective unbraced length and the member dimension correlate with each other in each direction. After being input once, the prompts for Le(b), Le(d), B, and D all default to their previous values. See the "Input Summary" section on the front sheet of this documentation.

MATERIAL PROPERTIES

The program prompts for material properties Fc, Fb, and E. Fb is not needed for axial column analysis; E is not needed for "short" columns. Thus if the design fits one of these situations, the unnecessary input prompt can be bypassed (hit R/S without answering the prompt). Constants calculated using these values (e.g. C-K and K) will be displayed, but if the estimate of length category was correct or the column is axially loaded, these incorrect constants will not enter into the calculation. If the lenght category estimate was not correct, however, garbage results. It's usually best to key in the correct values.

See the WBM documentation for information on factors affecting allowable stress input for beam-columns.

ALLOWABLE STRESSES

Engineers using the NDS should note that tabulated allowable stresses have decreased in recent editions; an attempt should be made to keep up to date. The higher equation constants allowed for machine-stress-rated (MSR or E-rated) timber are not implemented in these programs.

The program only checks column buckling on the gross section. It is left to the user to check crushing at the net section (i.e. at bolt holes). Shear is not checked in beam-columns. For truss chords the user must calculate C_t and apply it to E before input.

WCOL calculates and displays the allowable compressive stress about each axis, as well as the length category. For axially loaded columns it then picks the governing (least) of these and calculates the allowable axial load. For beam-columns the program calculates and displays the allowable compressive and bending stress, and the P-delta constant J. Then it calculates the interaction formula for each axis (which for the B axis is simply the actual/allowable compressive stress) and displays the governing (largest) one. If this is less than one the design is OK.

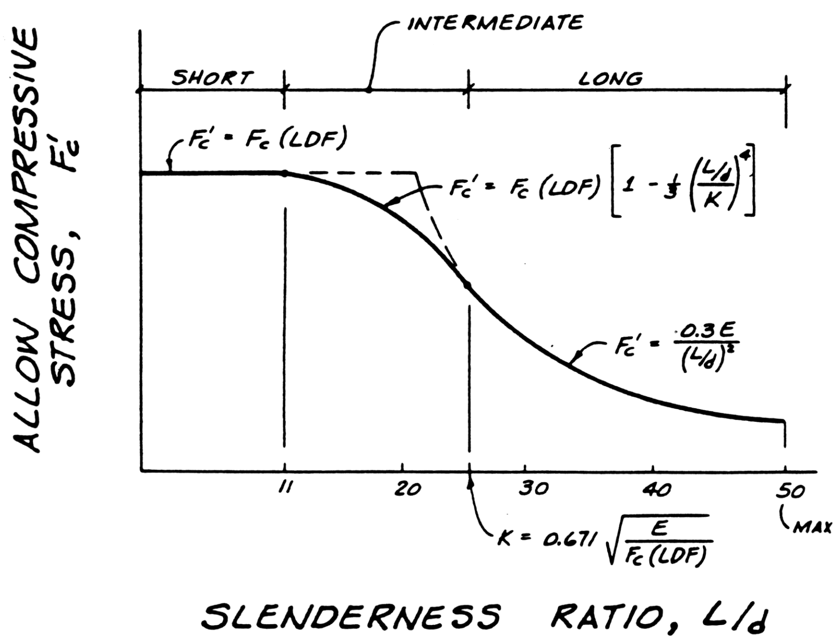
D218

EFFECTIVE LENGTH

The effective length for columns depends on the end fixity and the presence or absence of sidesway; such conditions are a matter of engineering judgement, but for conditions where sidesway is prevented, Le < Lu. For unbraced beams, on the other hand, Le > Lu. For the design of beam-columns, then, the choice of effective length probably should be based on whether the member is more beam or column. If doing the calculations by hand, both lengths could be used; the program, however, accepts only one effective length in each direction. Most members which carry side loading are braced laterally by the member delivering the load so this isn't a problem. For those infrequent cases where beam slenderness is a factor, it is up to the judgement of the user whether the program's results are acceptable.

If Le/D is greater than 50, the allowable stress output is zero.

All this is shown grapically by the following illustration, taken by permission from Design of Wood Structures by Donald E. Breyer (McGraw-Hill, 1980)



SUBROUTINE "WFc"

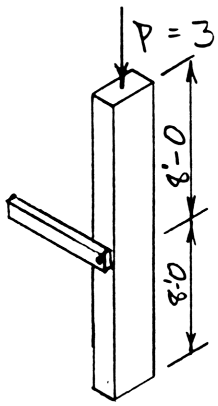
As mentioned, spaced column analysis is not covered. However, the allowable column stress subroutine of the program has been given a global label, "WFc," so that

DZ19

it may be called from user-written programs. The user's program must store the values of K, E, Fc, and LDF in the proper registers and D and Le must be in the stack. See the program listing for WFc (within WBM/WCOL, line 345) and the portion of WCOL which calls WFc (lines 238-243) for clarification.

BIAXIAL BENDING

WCOL does not analyze columns under biaxial bending; only uni-axial bending is considered. The moment prompted for is only that due to side loading, not axial eccentricity. The moments due to side loading and eccentricity are assumed to be additive. If they are of opposite sign, tending to cancel each other, the interaction is evaluated in a mathematically correct way; however, the intent of the Specification is rather vague on this point. Normally it's not a problem since the axial load and any side load are usually from different sources (i.e. side loads are usually from wind) so one of them would simply be neglected.



Design column for axial load = 30 k (including snow)

Sidesway prevented
Free ("Simple") Ends } Use $L_e = L_u$

Snow Load ⇒ LDF = 1.15

Design for wet conditions:

CUF = .91 for F_c
= 1.0 for E
= 1.0 for F_b

Use #1 DougFir, full cut : $F_c = 925$ psi
$E = 1{,}600{,}000$ psi
$F_b = 1300$ psi

(Assume "B+S" category due to bracing)

$F_{c.o} = \overset{CUF}{.91} \cdot 925 = \underline{841.75 \text{ psi}}$

__Try 5" × 8"__ : $f_c = \frac{30{,}000}{5 \times 8} = 750$ psi

$K = .671\sqrt{\frac{E}{F_c \cdot LDF}} = .671 \cdot \sqrt{\frac{1{,}600{,}000}{841.75 \times 1.15}} = 27.28$

$\frac{L_b}{D} = \frac{8' \times 12}{5} = 19.2$; $\frac{L_d}{D} = \frac{16' \times 12}{8} = \underline{\underline{24.0}}$

$11 < 24.0 < 27.28$ ⇒ Intermediate

$\therefore F_c' = F_{c.o} \cdot \left[1 - \frac{1}{3}\left(\frac{L/d}{K}\right)^4\right] \times \overset{LDF}{1.15} = 774.7$ psi

$774.7 > 750$ __OK__

Allowable P = 774.7 × 5" × 8" = __30.988 k__

As an example, try 10" × 5" oriented other way

$\frac{L_b}{B} = \frac{8' \times 12}{10"} = 9.6 < 10$ ⇒ Short ; $F_{c.b} = F_{c.o} \cdot LDF = 968.01$ psi

$\frac{L_d}{D} = \frac{16' \times 12}{5"} = 38.4 > 27.28$ ⇒ Long

$F_{c.d} = \frac{0.30E}{(L/d)^2} = \frac{.30 \times 1{,}600{,}000}{(38.4)^2} = 325.52$ psi ← Governs

Allow. P = 325.52 × 5" × 10" = __16.276 k__ __NG__

```
             XEQ "WCOL"
 6:17:39 PM  11/02/88

TITLE:
ROOF SUPPORT          RUN
*************************
NDS
Fc PSI
         841.7500     RUN
Fb PSI
                      RUN
E PSI
   1,600,000.000      RUN
---------
P, K
          30.0000     RUN
ECCd"
           0.0000     RUN
LATd M,'K
           0.0000     RUN
LDF
           1.1500     RUN
K=27.2799
*************************
Le-b"
          96.0000     RUN
Le-d"
         192.0000     RUN
B"
           5.0000     RUN
D"
           8.0000     RUN
INTRM
Fc'b=888.8360 PSI

INTRM
Fc'd=774.7104 PSI

ALL P=30.9884 K-OK
-------------------------
Le-b"
                      RUN
Le-d"
                      RUN
B"
          10.0000     RUN
D"
           5.0000     RUN
SHORT
Fc'b=968.0125 PSI

LONG
Fc'd=325.5208 PSI

ALL P=16.2760 K-NG
-------------------------
Le-b"
```

F_b not needed - can bypass

input same as before - can bypass

D221

Round Column

Redesign for round column. Remove brace.

$L_b = L_d = 16.0'$

Same material: $F_{c\cdot o} = 841.75$ psi

$E = 1{,}600{,}000$

Use "equiv. D" from rad. of gyration:

$D = r \times \sqrt{12}$

$r = \frac{Diam}{4} \Rightarrow D = .866 \cdot Diam$

Try 8" Ø member

$\frac{L_d}{D} = \frac{16.0' \times 12}{.866 \cdot 8"} = 27.71 > 27.28$

$\Rightarrow$ Long

$\therefore F_c' = \frac{0.30E}{(L/d)^2} = \frac{.30 \cdot 1{,}600{,}000}{(27.71)^2} = 624.96$ psi

Area $= \frac{\pi}{4}(8")^2 = 50.26 \square"$

Allowable P $= 50.26 \cdot 624.96 =$ **31.414 k**

$31.414 > 30$ **OK**

Or, analyze as "equiv. square" member:

$D = \sqrt{50.26} = 7.09"$

$\frac{L}{D} = \frac{16' \cdot 12}{7.09} = 27.08 < 27.28$

$\Rightarrow$ Intermediate (barely)

$\therefore F_c' = F_{c\cdot o} \cdot LDF \cdot \left[1 - \frac{1}{3}\left(\frac{L/D}{K}\right)^4\right]$

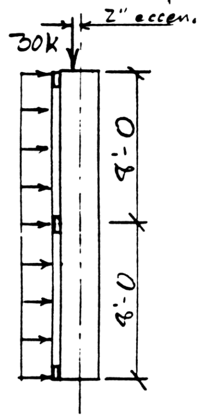
$= 841.75 \cdot 1.15 \cdot \left[1 - \frac{1}{3}\left(\frac{27.08}{27.28}\right)^4\right] = 654.65$ psi

Allowable P $= 50.26 \times 654.65 =$ **32.906 k**

(DZZL) Results are about the same

```
             XEQ "WCOL"
 6:21:25 PM  11/02/88

TITLE:
RE-DESIGN: ROUND COLUMN
                    RUN
************************
NDS
Fc PSI
        841.7500    RUN
Fb PSI
                    RUN
E PSI
   1,600,000.000    RUN
---------
P, K
         30.0000    RUN
ECCd"
                    RUN
LATd M,'K
                    RUN
LDF
          1.1500    RUN
K=27.2799
************************
Le-b"
        192.0000    RUN
Le-d"
        192.0000    RUN
B"
          0.0000    RUN
D"
          0.0000    RUN
DIA"
          8.0000    RUN
LONG
Fc'b=624.9633 PSI

LONG
Fc'd=624.9633 PSI

ALL P=31.4141 K-OK
------------------------
Le-b"
```

Design a wall beam-column for the same structure. Assume wind @ 40 psf. Columns @ 10' spa.



Use same wood:
$F_{c \cdot o} = 441.75$ psi
$E_o = 1{,}600{,}000$ psi
$F_{b \cdot 0} = 1300$ psi

$L_{e \cdot b} = 8.0' = 96"$
$L_{e \cdot d} = 16.0' = 192"$

With wind, LDF = 1.33
Without wind, LDF = 1.15 (Snow)

Wind Moment $= (40\text{ psf} \times 10' \times 8') \times \frac{16'}{4}$

$= 12.80^{'k}$

Try 6"×10" $f_c = \frac{30{,}000}{6 \times 10} = 500$ psi

Check minor-axis interaction:
(being the easiest to check)

$K = .671\sqrt{\frac{E}{f_{c \cdot o} \cdot LDF}} = .671\sqrt{\frac{1{,}600{,}000}{441.75\,(1.15 \text{ or } 1.33)}}$

$K_{wind} = 25.367$

$K_{snow} = 27.240$

∴ Check without wind

$\frac{L_b}{B} = \frac{96"}{6"} = 16.0$

$11 < 16.0 < 27.240 \Rightarrow$ Intermed.

$F'_{c \cdot b} = 441.75 \cdot 1.15\left[1 - \frac{1}{3}\left(\frac{16}{27.28}\right)^4\right] = 929.83$ psi

$\text{Interaction}_b = \frac{500 \text{ psi}}{929.83} = .5377 < 1.0$

Minor Axis **OK**

Check major axis, **with wind:**

$S = \frac{6" \times 10^2}{6} = 100.0 \text{ in}^3$; $C_f \cdot C_F = 1.0$

$C_K = .811\sqrt{E/F_{b \cdot o}} = .811\sqrt{\frac{1{,}600{,}000}{1300}} = 28.45$

$C_S = \sqrt{\frac{L_{e \cdot b} \cdot d}{b^2}} = \sqrt{\frac{8' \times 12 \times 10"}{6^2}} = 5.16$

$5.16 < 10 \Rightarrow$ "Short" for bending

∴ $F'_{b \cdot 1} = F'_{b \cdot 2} = 1300 \cdot \overset{LDF}{1.33} = \underline{1729 \text{ psi}}$

In compression, $\frac{L_d}{D} = \frac{16' \times 12}{10"} = 19.20$

$11 < 19.2 < 25.367 \Rightarrow$ "Intermed." for col.

$F'_{c \cdot d} = 441.75 \times \overset{LDF}{1.33} \times \left[1 - \frac{1}{3}\left(\frac{19.2}{25.367}\right)^4\right] = 997.05$

$J = \frac{L/d - 11}{K - 11} = \frac{19.2 - 11}{25.367 - 11} = .5708$

$f_b = \frac{12.80^{'k} \times 12}{100 \text{ in}^3} = 1.536$ ksi this doesn't look good

$\text{Interaction} = \frac{f_c}{F'_c} + \frac{f_b + f_c(6 + 1.5J)(e/D)}{F_b - J \cdot f_c}$

$= \frac{500}{997.05} + \frac{1536 + 500(6 + 1.5 \cdot .5708)(2"/10")}{1729 - .5708(500)}$

$= 2.0404 > 1.0$ **NG**

To see which governs, check without wind:

$J = \frac{19.2 - 11}{27.24 - 11} = .5037$; $f_b = 0$

$F'_b = 1300 \times \overset{LDF}{1.15} = 1495$ psi

$F'_{c \cdot d} = 441.75 \times 1.15 \times \left[1 - \frac{1}{3}\left(\frac{19.2}{27.240}\right)^4\right] = 444.4365$ psi

$\text{Interaction} = \frac{500}{~~444.44~~} + \frac{500(6 + 1.5 \times .5037)(2"/10")}{1495 - .5037 \cdot 500}$

$= 1.1060 > 1.0$

Clearly, Loading with wind governs

D223

Wall Column

Try 6" × 14"

$f_c = \frac{30,000}{6 \times 14} = 357.143$ psi

$S = \frac{14^2 \times 6"}{6} = 196.0 \text{ in}^3$

$f_b = \frac{12.80 \times 1.4 \times 12}{196.0} = 783.673$ psi

K = 25.367 ; C_K = 24.45 (previous page)

$C_S = \sqrt{\frac{Le \cdot b \times D}{B^2}} = \sqrt{\frac{8' \times 12 \times 14"}{6^2}} = 6.11$

6.11 < 10.0 ⇒ "Short" for bending

$\therefore F_{b2} = 1300 \times 1.33 = 1729.0$ psi

$C_F = 1.0$ (rect.) ; $C_f = \left(\frac{12}{14}\right)^{1/9} = .9830$

$\therefore F_{b1} = 1300 \times 1.33 \times .9830 = 1699.64$

F_{b1} governs @ 1699.64 psi

$\frac{Le \cdot d}{D} = \frac{16' \times 12}{14} = 13.714$

11 < 13.714 < K = "Intermediate"

$\therefore F_c' = 841.75 \times 1.33 \times \left[1 - \frac{1}{3}\left(\frac{13.714}{25.367}\right)^4\right]$

$= 1087.647$ psi

$J = \frac{L/D - 11}{K - 11} = \frac{13.714 - 11}{25.367 - 11} = .1889$

$\text{Interaction} = \frac{f_c}{F_c'} + \frac{f_b + f_c(6 + 1.5J)(e/d)}{F_{b1}' - J f_c}$

$= 1.0049 \approx 1.0$ say OK

(less than 1% over)

(D224)

```
          XEQ "WCOL"
4:45:47 PM  01/25/88

TITLE:
WALL BEAM-COLUMN      RUN
************************
NDS
Fc PSI
          841.7500    RUN
Fb PSI
        1,300.0000    RUN
E PSI
    1,600,000.000     RUN
---------
P, K
           30.0000    RUN
ECCd"
            2.0000    RUN
LATd M,'K
           12.8000    RUN
LDF
            1.3300    RUN
C-K=28.4518
K=25.3668
************************
Le-b"
           96.0000    RUN
Le-d"
          192.0000    RUN
B"
            6.0000    RUN
D"
           10.0000    RUN
S=100.0000
C-S=5.1640
Fb'=1,729.0000 PSI

INTRM
Fc'b=1,060.4621 PSI

INTRM
Fc'd=997.0495 PSI

J=0.5708
INTRXN-d=2.0404-NG
------------------------
Le-b"
                      RUN
Le-d"
                      RUN
B"
            6.0000    RUN
D"
           14.0000    RUN
S=196.0000
C-S=6.1101
Fb'=1,699.6381 PSI

INTRM
Fc'b=1,060.4621 PSI

INTRM
Fc'd=1,087.6455 PSI

J=0.1889
INTRXN-d=1.0049-NG
------------------------
Le-b"
```

Say OK

Program WBM,WCol For Timber Design 681 bytes Page 1 of 4

	X	Y	Z	T	L
01♦LBL "WBM"					
02 SF 09					
03 11					
04 GTO 00					
05♦LBL "WCOL"					
06 CF 09					
07 18					
08♦LBL 00					
09 SF 08					
10 "NDS"					
11 XROM "AA"					
12♦LBL F					
13 "Fc"					
14 FS? 09					
15 "F-V"					
16 "⊢ PSI"					
17 PROMPT	F_c or F_v				
18 STO 09					
19 "Fb PSI"					
20 PROMPT					
21 STO 01	F_b				
22 "E PSI"					
23 PROMPT					
24 STO 07	E				
25 FC? 08					
26 GTO 10					
27♦LBL E					
28 CF 10					
29 XROM "L"					
30 FS? 09					
31 GTO 00					
32 "P, K"					
33 PROMPT					
34 STO 12					
35 CLX					
36 "ECCd""					
37 PROMPT					
38 STO 14					
39 "LATd "					
40 XROM "M"					
41 PROMPT					
42 *					
43 STO 13	$M_d''^{k}$				
44 RCL 12	P				
45 RCL 14	Ecc_d	P	M		
46 *					
47 +	ΣM				
48 X=0?					
49 SF 10					

	X	Y	Z	T	L
50♦LBL 00					
51 E					
52 "LDF"					
53 PROMPT					
54 STO 05					
55 .6					
56 SQRT	$\sqrt{.6}$				
57 FS? 04					
58 .811					
59 RCL 07					
60 RCL 01	F_b	E			
61 /					
62 SQRT					
63 *	C_K				$C_K = .811\sqrt{E/F_b}$
64 STO 10					
65 "C-K"					
66 FC? 10					
67 XEQ 05					
68 FS? 09					
69 GTO D					
70 RCL 07	E				
71 RCL 09	F_c				
72 /					
73 RCL 05	LDF				
74 /					
75 SQRT					
76 .671					
77 *	K				$K = .671\sqrt{\frac{E}{LDF \cdot F_c}}$
78 STO 18					
79 "K"					
80 XEQ 05					
81 SF 14					
82♦LBL D					
83 XROM "LL"					
84 SF 06					
85 CF 07					
86 "Le-b""					
87 RCL 02	Le_b				Default - Previous L_e
88 PROMPT					
89 STO 02					
90 RCL 11					
91 "Le-d""					
92 FC? 09	Le_d				
93 PROMPT					
94 STO 11					
95 XROM "BD"	D	B			
96 X=0?					
97 GTO 02					
98 X↑2	D^2				
99 STO 08					

D225

Step	X	Y	Z	T	L
100 LASTX	D				
101 RCL 03	B				
102 X<Y?					
103 CF 06					
104 *	Area				
105 STO 06					
106 RCL 04	D				
107 *					
108 6					
109 /	S_x				
110 E	1				
111 GTO 03					
112♦LBL 02					
113 SF 07					
114 "DIA="					
115 PROMPT					
116 STO 03					
117 .866					
118 *	eff. d				
119 STO 04					
120 X<> 03	diam.				
121 ENTER↑					
122 X↑2	d^2	d			
123 PI					
124 *					
125 4					
126 /	area				
127 STO 06					
128 STO 08					
129 8	8	area	d		
130 /					
131 *	S				
132 1.18	1.18				$S=\frac{\pi d^3}{32}$
133♦LBL 03					
134 X<>Y	S	C_F			
135 "S"					
136 STO 00					
137 FC? 10					
138 XEQ 05					
139 12	12				
140 RCL 08	d^2	12	S	C_F	
141 SQRT	eff. d				
142 /	12/d				
143 9					
144 1/X	1/9	12/d	S	C_F	
145 Y↑X					
146 E	1	$(12/d)^{1/9}$	S	C_F	
147 X>Y?					
148 X<>Y	C_F				
149 R↑	C_F				
150 *	$C_F \cdot C_F$				
151 STO 08					
152 CF 08					

Step	X	Y	Z	T	L
153♦LBL 10					
154 FS? 10					
155 GTO 12					
156 RCL 02	$L_e \cdot b$				
157 RCL 04	d				
158 *					
159 RCL 03	b^2				
160 X↑2					
161 /					
162 SQRT	C_s				$C_s=\sqrt{\frac{L_e \cdot b \cdot d}{b^2}}$
163 "C-S"					
164 XEQ 05					
165 10					
166 X<>Y	C_s	10			Short
167 X<=Y?					
168 FC? 06					
169 .					
170 RCL 10	C_K				
171 X<=Y?					
172 GTO 00					
173 XEQ 04	Factor				Intermed.
174 RCL 08	$C_F \cdot C_f$				
175 X>Y?					
176 X<>Y					
177 RCL 01	F_b				
178 *	F_b'				
179 RCL 05	LDF				
180 *					
181 GTO 01					
182♦LBL 00					
183 CLX					
184 50	50	C_s			
185 X<Y?					
186 CLX	0?				
187 X=0?					
188 GTO 01					
189 RCL 07	E				Long
190 FC? 04					
191 .4	.4	E	50	C_s	
192 FS? 04					
193 .438					
194 *	.4E				
195 R↑	C_s				
196 X↑2					
197 /	$\frac{.4E}{(C_s)^2}$				
198♦LBL 01					
199 "Fb'="					
200 XEQ 06					
201 FC? 09					
202 GTO 12					
203 RCL 00	S	F_b'			
204 *					
205 XROM "M="					

D226

Step	X	Y	Z	T	L
206 2					
207 AVIEW					
208 3					
209 FS? 07					
210 4					
211 /	$\frac{2}{3}$ or $\frac{3}{4}$				
212 E3	1000				
213 /					
214 RCL 09	F_v psi				
215 *					
216 RCL 05	LDF				
217 *					
218 "ALL V="	All. V				
219 XEQ 07					
220 AVIEW					
221 GTO D					
222♦LBL 12					
223 STO 17	F_b				
224 RCL 02	L_b				
225 RCL 03	b	L_b			
226 "Fc'b"					
227 XEQ "WFc"					
228 STO 16	F_{c-b}				
229 RCL 11	L_d				
230 RCL 04	d	L_d			
231 "Fc'd"					
232 XEQ "WFc"	F_{c-d}				
233 FC? 10					
234 GTO 00					
235 RCL 16	F_{c-b}				
236 X>Y?					
237 X<>Y	lesser				
238 "├ P="					
239 XEQ 07	all. P				
240 RCL 12	act. P	allow. P			
241 GTO 11					
242♦LBL 07	stress				
243 RCL 06	area	stress			
244 *	force				
245 ARCL X					
246 "├ K"					
247 RTN					
248♦LBL 00	F_{c-d}				
249 RCL 11	L_d				
250 RCL 04	d				
251 /	L/d				
252 11					
253 -	L/d - 11	F_{c-d}			
254 X<0?					
255 CLX					
256 "J"					

Rectangular section (d × b): $f_v = \frac{3}{2}\frac{V}{bd}$

Round section (dia.): $f_v = \frac{4}{3}\frac{V}{area}$

Note: if from line 155, R.17 does not contain F_b. In this case (no moment), F_b is not used.

Step	X	Y	Z	T	L
257 RCL 18	K				
258 LASTX	11	K	L/d - 11	F_{c-d}	
259 -	11 - K				
260 /	-J				
261 ABS	J				
262 SIGN	1				
263 LASTX	J	1	F_{c-d}		
264 X<=Y?					
265 X<>Y	greater	lesser			
266 RDN	J	F_{c-d}			
267 XEQ 05					
268 STO \	J	F_{c-d}	•		
269 1.5			•		
270 *		•	•		
271 6			•		
272 +	( )	•	•		
273 RCL 14	eccen.		•		
274 RCL 04	d			•	
275 /	e/d		•		
276 *	( )e/d	•			
277 RCL 12	P		•		
278 RCL 06	area			•	
279 /	f_c		•		
280 STO 15					
281 *		•			
282 RCL 13	M		•		
283 RCL 00	S			•	
284 /	f_b	$f_c( )\frac{e}{d}$	•		
285 +	numer.	•			
286 RCL \	J		•		
287 RCL 15	f_c			•	
288 *	J f_c		•		
289 RCL 17	F_b			•	
290 -			•		
291 CHS	denom		•		
292 /	2nd term	•			
293 RCL 15	f_c	2nd term	F_{c-d}	F_{c-d}	
294 R↑	F_{c-d}	f_c			
295 /	f_c/F_c				
296 +	Interaction-d				
297 RCL 15	f_c				
298 RCL 16	F_{c-d}				
299 /	Interaction-b				
300 "INTRXN-"					
301 X<=Y?					
302 "├d="					
303 X>Y?					
304 "├b="					
305 X<=Y?					
306 X<>Y	greater				
307 SIGN	1				
308 LASTX	Interaction				
309 ARCL X					

Interaction = $\frac{f_c}{F_c} + \frac{f_b + f_c(6+1.5J)\frac{e}{d}}{F_b - J f_c}$

Step	X	Y	Z	T	L	Notes
310♦LBL 11						
311 CF 07						
312 X>Y?						
313 SF 07						
314 XROM "OK"						
315 AVIEW						
316 GTO D						
317♦LBL 04	K	$\frac{L}{b}$ or 0	11	F_b		
318 /	$\frac{L/b}{K}$					
319 X↑2						
320 X↑2	$(\ )^4$					$F_b' = F_b * LDF * (1-(\frac{C_s}{C_k})^4)$
321 3						
322 /						$F_c' = F_c * LDF * (1-(\frac{L/b}{K})^4)$
323 SIGN	1					
324 LASTX						
325 -	$1-(\ )^4$	11	F_b	F_b		
326 RTN						
327♦LBL 05						
328 "⊢="						
329 ARCL X						
330 FC? 00						
331 AVIEW						
332 RTN						
333♦LBL "WFc"	b	L_b				
334 ASTO 15						"F_c' b" stored
335 "LONG"						
336 /	L/b					
337 50	50	L/b				
338 X<=Y?						
339 CLX						
340 X<=Y?						$\frac{L}{b} > 50 \Rightarrow$ Allow. stress = 0
341 GTO 08						
342 CLX		L/b				
343 11	11					
344 X<>Y	L/b	11				
345 X<=Y?						
346 .	0					
347 RCL 18	K	0 or L/b	$\frac{L}{b}$ or 11			
348 X<=Y?						If $\frac{L}{b} < 11$, $K > 0 \Rightarrow$ don't GTO 09
349 GTO 09						If $\frac{L}{b} > 11$, then if $K < L/b$, GTO 09
350 XEQ 04	$1-(\ )^4$					
351 "INTRM"						
352 E	1					
353 X<>Y	$1-(\ )^4$	1				
354 X=Y?						
355 "SHORT"						
356 RCL 09	F_c					
357 *						
358 RCL 05	LDF					
359 *	Allow. F_c'					
360 GTO 08						

Step	X	Y	Z	T	L	Notes
361♦LBL 09	K	L/b				"LONG" colu[redacted]n
362 RCL 07	E					
363 .3	.3	E	K	L/b		
364 *	.3E					
365 R↑	L/b					
366 X↑2						Allow. $F_c' = \frac{.3E}{(L/b)^2}$
367 /	F_c'					
368♦LBL 08						
369 FC? 00						
370 AVIEW						
371 CLA						
372 ARCL 15	F_{c-b}'					"F_c'b"
373♦LBL 06						
374 "⊢="						
375 ARCL X						
376 E3	1000					
377 /	F_c' ksi					
378 "⊢ PSI"						
379 FC? 00						
380 AVIEW						
381 ADV						
382 "ALL"						
383 END	F_c' ksi					

DZZ4

SIMPLE SPAN TRUCK MOMENTS

xeq T2 for Type 2 (H15 or H20) truck* (SIZE 013)
T3 for Type 3 straight truck
T4 for Type 4 straight truck (triple axle)
2S1 for HS20 configuration*
3S2 for Type 3S2 semi-trailer truck
3S3 for Type 3S3 semi-trailer truck (triple axle)

*Generic trucks only--lane loading not calculated by these routines. RAM program "TRUCK" contains a routine which calculates and compares lane loadings.

SOFTKEYS (USER mode ON)

No softkeys are used by the truck routines

USER FLAGS

No user flags are used by the truck routines

INPUT/OUTPUT SUMMARY

These routines have no prompts or output explanation. All the required data must be stored in the following registers by a calling program, and the results displayed by that program. The truck routines are simple numerical calculation routines; output consists of the governing moment for that truck in the X-register and register 12. An example calling program which generates the moments for Minnesota trucks is included in the main documentation for use or reference.

Register	Input
00	Span in feet
01	P1 (front wheel load), kips
02	P2 (second wheel load), kips
03	P3 (third wheel load), kips
04	L2 (distance between the second and third wheel of a type 2S1 truck) in feet
05	L1 (distance between the front two wheels), feet
06	L2 (distance between the middle and rear sets of wheels), feet

PROGRAM FLAGS

No program flags are used by the truck routines

(D229)

STORAGE REGISTER USE

Register	Value
00	Span in feet
01	P1 (front wheel load), kips
02	P2 (second wheel load), kips
03	P3 (third wheel load), kips
04	L2 (distance between the second and third wheel of a type 2S1 truck) in feet
05	L1 (distance between the front two wheels), feet
06	L2 (distance between the middle and rear sets of wheels), feet
07	P x L
08	Minimum P
09	$-(\sum P \times L)/\sum P$
10	Maximum P
11	$\sum P$
12	Governing Moment

D230

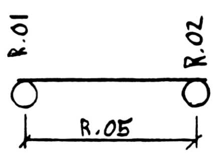
Storage Register Use - Input

Each truck type, and corresponding program label, is named for the number of axles. Semi-trailer truck designations have the number of axles in both the tractor and trailer.

Storage register use is as shown below. The user-written calling program must fill these registers with the appropriate values. Where a dimension is assumed by the programs, it is enclosed in parenthesis.

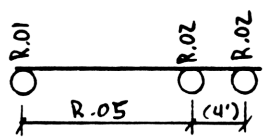
Truck — **Assumptions made**

T2



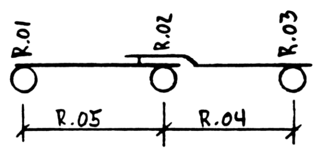
None.

T3



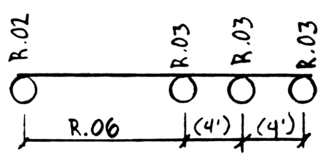
Rear wheels identical, 4' apart. (But, see 2S1)

2S1



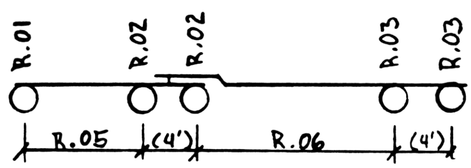
None.

T4



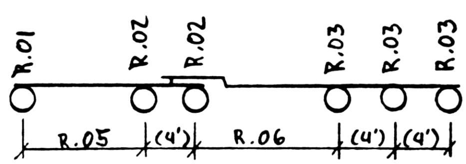
Rear wheels identical, 4' apart. R.06 length assumed long enough that last 3 axles govern over first 3.

3S2



Wheels which are 4' apart are identical.

3S3



Wheels which are 4' apart are identical.

The span in feet must be stored in R.00 for all routines.

Note: for reference or use in RAM - not resident in ROM

Program TRUCK For Minnesota Truck Moments 194 bytes Page 1 of 1

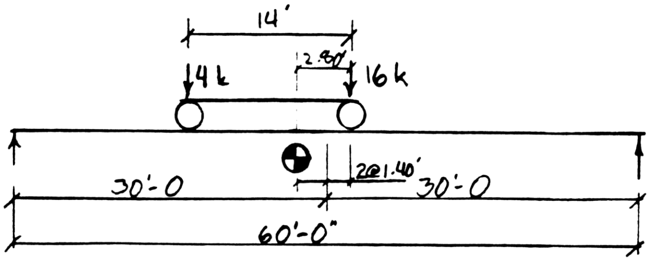
Step	X	Y	Z	T	L
01♦LBL "TRUCK"					
02 CF 07					
03 12					
04 XROM "S"					
05♦LBL 00					
06 "SPAN'"					
07 PROMPT					
08 STO 00	Span				
09 "I="					
10 50					
11 X<>Y					
12 125	125	span	50		
13 +					
14 /	50/(125+S)				
15 .3					
16 X>Y?					
17 X<>Y	Impact				
18 ARCL X					
19 FC? 00					
20 AVIEW					
21 ADV					
22 7					
23 STO 01	P_1				
24 8.5					
25 STO 02	P_2				
26 12					
27 STO 05	l				
28 XEQ "T3"	M_3				
29 "T3"					
30 XEQ 01					
31 4.65	P_1				
32 STO 01					
33 8					
34 STO 02	P_2				
35 STO 03	P_3				
36 10					
37 STO 05	l_1				
38 25					
39 STO 06	l_2				
40 XEQ "3S2"	M_{3S2}				
41 "3S2U"					
42 XEQ 01					
43 4.5					
44 STO 01	P_1				P_2 = 8 - still stored
45 6.5					
46 STO 03	P_3				
47 XEQ "3S3"	M_{3S3}				
48 "3S3"					
49 XEQ 01					
50 ADV					
51 4	P_1				
52 STO 01					
53 16	P_2				
54 STO 02					
55 STO 03	P_3				
56 14	l_1				
57 STO 04					
58 STO 05	l_2				
59 XEQ "T2"	M_2				
60 "H 20"					
61 XEQ 03					
62 XEQ "2S1"	M_{2S1}				
63 "HS20"					
64 XEQ 03					
65 XROM "LL"					
66 GTO 00					
67♦LBL 03					
68 CF 07					
69 RCL 00	Span				
70 4					
71 *	4S				Lane Load eq:
72 225					$M = \frac{1}{2} \cdot (.64 \times S^2/8$
73 +					$+ 18k \cdot S/4)$
74 E2					
75 /					
76 RCL 00	span				
77 *	Lane M				
78 RCL 12	Truck M				
79 X>Y?					
80 GTO 01					
81 X<>Y					
82 STO 12	Lane M				
83 SF 07					
84♦LBL 01					
85 "⊢: M="					
86 ARCL X					
87 XROM "FK"					
88 FS? 07					
89 "⊢(L)"					(L) → Lane loading governs
90 FC? 00					
91 AVIEW					
92 END					

```
            XEQ "TRUCK"
 9:03:25 AM  02/08/89

TITLE:
MINNESOTA TRUCKS      RUN
************************
SPAN'
           60.0000    RUN
I=0.2703

T3: M=301.4340 'K
3S2U: M=297.9510 'K
3S3: M=292.8094 'K

H 20: M=279.0000 'K(L)
HS20: M=403.2667 'K
-----------------------
SPAN'
```

D232

Truck T2

Analyze the moment due to an H20 truck configuration on a 60'-0" span.



One wheel

$$\frac{16k \times 60'}{4} = \underline{240.0'^{k}}$$

Two Wheels

$$C.G. = \frac{4k \times 14'}{4k + 16k} = 2.80' \text{ from } 16k \text{ wheel}$$

$$R_{Right} = \left(\frac{60'}{2} - \frac{2.80'}{2}\right) \times \frac{20k}{60'} = 9.53\ k$$

$$M = \left(\frac{60'}{2} - \frac{2.80'}{2}\right) \times 9.53\ k = \underline{272.65'^{k}}$$ ← Governs

Check lane loading:

W = .64 k/foot per lane
P = 18.0 k/lane

Per wheel: $.32 \times 60^2/8 = 144.0$
$9.0 \times 60/4 = \underline{135.0}$
279.00 '^k/wheel - Lane load governs

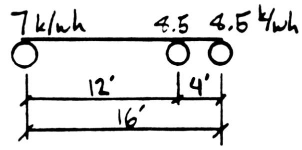
Note: Routine T2 does not figure lane loads. However, user program TRUCK does.

D233

Analyze the moment due to a Minnesota Type 3 truck as shown.

Truck T3

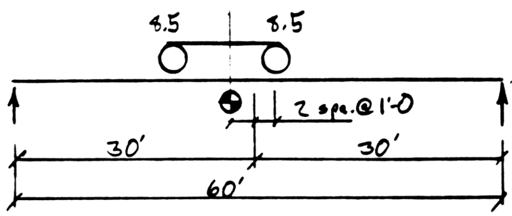
Span = 60'-0, simple.



One wheel

$$M = \frac{Pl}{4} = \frac{8.5\,k \times 60'}{4} = \underline{127.50'^{k}}$$

Two wheels



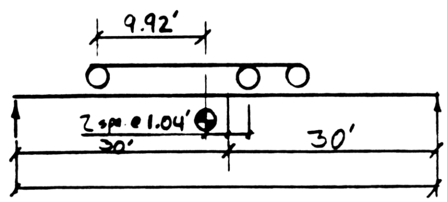
$$C.G. = \frac{8.5\,k \times 4'}{2 \times 8.5\,k} = 2.0'$$

$$R_R = (30' - \tfrac{2.0'}{2}) \times 17.0\,k / 60' = 8.22\,k$$

Max. M under right tire:

$$M = 8.22\,k \times (30' - \tfrac{2.0'}{2}) = \underline{238.28'^{k}}$$

Three wheels



$$C.G. = \frac{8.5\,k \times 12' + 8.5\,k \times 16'}{2 \times 8.5\,k + 7} = 9.92'$$

$$R_R = \frac{(30' - 2.08/2)}{60'} \times 24\,k = 11.583\,k$$

Max. M under middle tire:

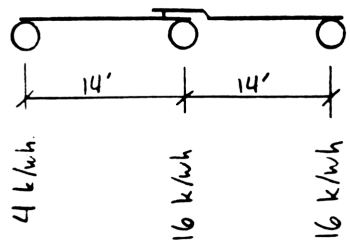
$$M = 11.583\,k \times (30' - \tfrac{2.08'}{2}) - 8.5\,k \times 4' = \underline{301.43'^{k}}$$

Max. = 301.43'k/wheel line

(D234)

Analyze the moment due to an HS20 truck configuration as shown.

Span = 60'-0, simple.

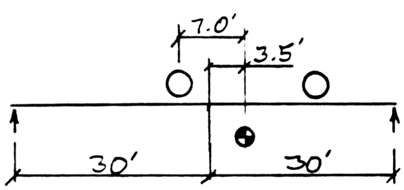


Truck 2S1

One wheel

$$M = \frac{P\ell}{4} = \frac{16\,k \times 60'}{4} = 240.0'^{k}$$

Two wheels

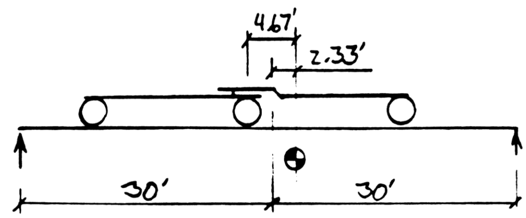


By inspection, C.G. = 7.0'

$$R_L = (30' - \frac{7.0'}{2}) \times 32k/60' = 14.13\,k$$

$$M = (30' - 3.5') \times 14.13 = \underline{374.53'^{k}}$$

Three wheels



$$C.G. = \frac{14' \times 16k - 14' \times 4k}{36\,k} = 4.67'$$

$$R_L = (30' - \frac{4.67'}{2}) \times \frac{36\,k}{60'} = 16.60\,k$$

$$M = (30' - \frac{4.67'}{2}) \times 16.60k - 4k \times 14'$$

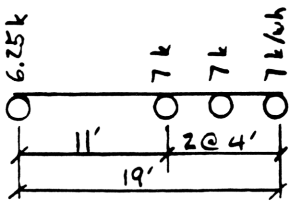
$$= \underline{403.27'^{k}}$$

Max. M = 403.27'^k

Analyze the moment due to an Iowa Type 4 truck as shown.

Truck T4

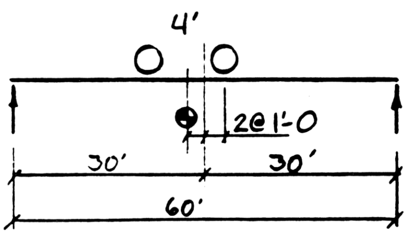
Span = 60'-0, simple.



One wheel

$\frac{Pl}{4} = 7k \times 60/4 = \underline{105.00'^{k}}$

Two wheels



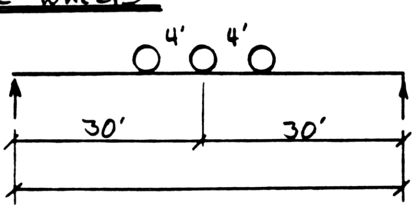
By inspection, C.G.@2.0'

$R_R = \frac{(30' - \frac{2.0'}{2})}{60'} \times (2 \cdot 7k) = 6.76\ k$

Max. M under right tire:

$M = (30' - \frac{2.0'}{2}) \times 6.76\ k = \underline{196.23'^{k}}$

Three wheels

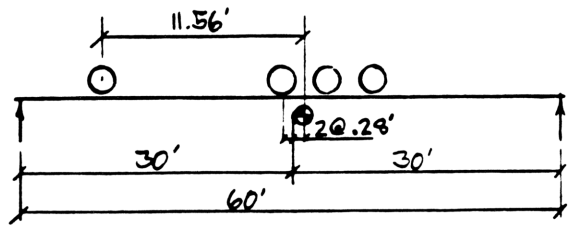


By inspection, $R_R = R_L = \frac{3 \times 7k}{2} = 10.5\ k$

$M = 30' \times 10.5\ k - 4' \times 7k = \underline{287.00'^{k}}$

D236

Four wheels



$C.G. = \frac{21k \times 15'}{27.25k} = 11.56'$ from front

$R_L = (30' - \frac{.56}{2}) \times 27.25k / 60'$

$= 13.50\ k$

Moment under 2nd tire:

$M = (30' - .56/2) \times 13.50 = 401.16$
$- 6.25k \times 11' = -68.75$
$332.41'^{k}$

Check for max. M under 3rd tire.

$R_R = (30' - \frac{3.44'}{2}) \times \frac{27.25k}{60'} = 12.84\ k$

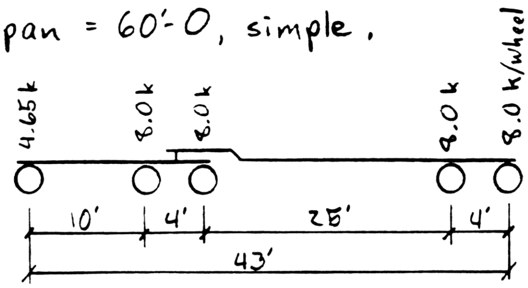
$M = (30 - \frac{3.44}{2}) \times 12.84 - 7k \times 4' = \underline{335.22'^{k}}$

Max. M = 335.22'^k/wheel line

Analyze the moment due to a Minnesota Type 3S2 truck as shown.

Truck 3S2

Span = 60'-0, simple.



In the interest of brevity, calculations similar to those of previous pages will be omitted or shortened.

One wheel $M = 120.0'^{k}$

Two wheels $M = 224.27'^{k}$

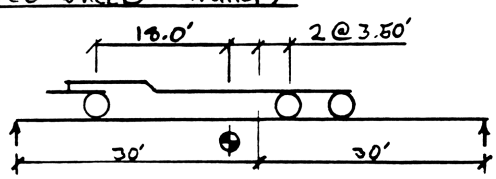
Three wheels (tractor)

$$C.G. = \frac{8.0 \times (10' + 14')}{20.65\,k} = 9.30' \text{ from front wh.}$$

$$M = \left(30' - \frac{.70}{2}\right)^2 \times \frac{20.65\,k}{60'} - 8\,k \times 4'$$

$$= \underline{270.54'^{k}}$$

Three wheels (trailer)



$$C.G. = \frac{8.0 \times (25' + 29')}{24.0\,k} = 14.0'$$

$$R_R = \left(30' - \frac{7.0'}{2}\right) \times \frac{24.0\,k}{60'} = 10.60\,k$$

$$M = (30' - 3.5') \times 10.60 - 8\,k \times 4' = \underline{248.90'^{k}}$$

Four wheels (rear)

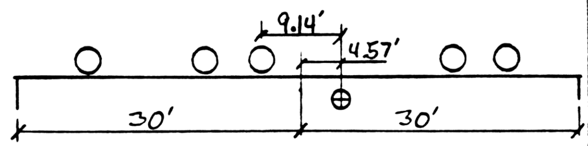
By inspection, C.G. = 16.5' from front

$$R_{min} = \left(30' - \frac{12.5'}{2}\right) \times \frac{32.0\,k}{60'} = 12.67\,k$$

$$M = \left(30' - \frac{12.5}{2}\right) \times 12.67\,k - 8 \times 4' = \underline{268.93'^{k}}$$

Five wheels

Max. M under 3rd wheel.



$$C.G. = \frac{8.0 \times (25' + 29') - 8 \times 4 - 4.65 \times 14}{36.65}$$

$$= 9.14'$$

$$R_L = \left(30' - \frac{9.14'}{2}\right) \times \frac{36.65\,k}{60'} = 15.53\,k$$

$$M = \left(30' - \frac{9.14}{2}\right) \times 15.53\,k - (8\,k \times 4' + 4.65\,k \times 14')$$

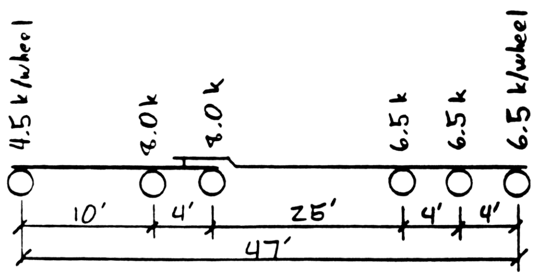
$$= \underline{297.95'^{k}}$$

$$\text{Max. } M = 297.95'^{k}$$

Analyze the moment due to a Minnesota Type 3S3 truck as shown.

Truck 3S3

Span = 60'-0, simple.



In the interest of brevity, calculations similar to those of previous pages will be omitted or shortened.

One wheel M = 120.0'·k

Two wheels M = 224.27'·k

Three wheels (tractor)

$$C.G. = \frac{8.0 \cdot (10' + 14')}{20.50\,k} = 9.37'$$

$$M = (30' - \frac{.63'}{2})^2 \times \frac{20.50k}{60'} - 8k \cdot 4'$$

$$= \underline{269.03'^{\cdot k}}$$

Rear three wheels

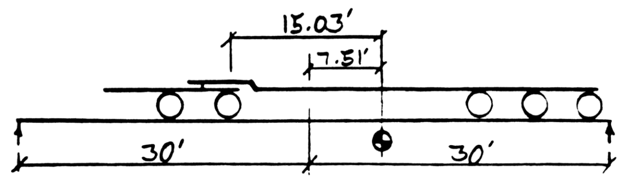
$$M = 30' \cdot \frac{3 \cdot 6.5k}{2} - 6.5k \cdot 4' = \underline{266.50'^{\cdot k}}$$

Rear four wheels

(as for Truck T4) : $M = \underline{278.66'^{\cdot k}}$

(DZ38)

Rear five wheels



Under 2nd wheel:

$$C.G. = \frac{3 \cdot 6.5k \cdot 29' - 8k \cdot 4'}{35.5\,k} = 15.03'$$

$$R_L = (30' - \frac{15.03'}{2}) \times 35.5k/60' = 13.30\,k$$

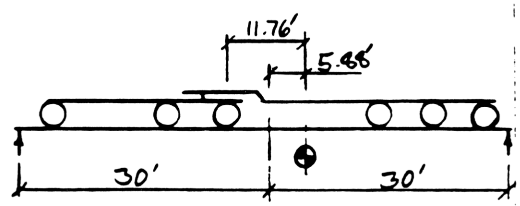
$$M = (30' - 7.51') \cdot 13.30k - 8k \cdot 4' = \underline{267.16'^{\cdot k}}$$

Under 3rd wheel : C.G. = 25' - 15.03' = 9.97'

$$M = (30' - \frac{9.97'}{2})^2 \times \frac{35.5k}{60'} - 6.5k \cdot (4' + 8')$$

$$= \underline{292.21'^{\cdot k}}$$

Six wheels - Max. under 3rd wheel



$$C.G. = \frac{3 \cdot 6.5 \cdot 29' - 8 \cdot 4 - 4.5 \cdot 14}{40.0\,k} = 11.76'$$

$$R_L = (30' - 5.88') \cdot \frac{40\,k}{60'} = 16.08\,k$$

$$M = (30' - 5.88') \cdot 16.08 - (8 \cdot 4' + 4.5 \cdot 14')$$

$$= \underline{292.81'^{\cdot k}}$$

Max. M = 292.81'·k

Program Trucks For Simple Span Live Load Moments 459 bytes

Step	X	Y	Z	T	L
01♦LBL "3S2"					
02 XEQ 13	M_1				
03 XEQ 03	M_3				M_3 for front 3 wheels (tractor)
04 RCL 03	P_3				
05 RCL 02	P_2				
06 X>Y?					
07 X<>Y	P_{min}				
08 STO 08					
09 X<>Y					
10 STO 10	P_{max}	P_{min}			
11 XEQ 12	M_2				
12 RCL 10	P_{max}				
13 RCL 10	✓				
14 RCL 08	P_{min}				
15 RCL 06	l_2	P_{min}	P_{max}	P_{max}	
16 XEQ 14	M_3'				M_3' for rear 3 wheels
17 RCL 10					
18 ST+ X	$2P_{max}$				
19 RCL 06	l_2				
20 2					
21 +	l_2+2				
22 RCL 08	P_{min}				
23 *					
24 -	$-\Sigma Pd$				
25 RCL 10					
26 RCL 08					
27 +	ΣP	$-\Sigma P \cdot d$			
28 XEQ 10	$M'/2$				
29 ST+ X	M'				
30 RCL 10	P_{max}				
31 4	4'				
32 *	$4' \cdot P$				
33 -	M_4				M_4 for rear 4 wheels
34 XEQ 22					
35 RCL 02	P_2				
36 4					
37 *	$4P_2$				
38 RCL 05	l_1				
39 LASTX	4				
40 +	l_1+4				
41 RCL 01	P_1				
42 *					
43 +	$P_1(l_1+4)+4P_2$				
44 STO 07	$\Sigma P \cdot d_{front}$				
45 RCL 06	l_2				
46 ST+ X					
47 4					
48 +					
49 RCL 03	P_3				
50 *	$P_3(2l_2+4)$				
51 -	$\Sigma P \cdot d$				
52 RCL 11	$\frac{1}{2}\Sigma P_4$				
53 ST+ X	ΣP_4				
54 RCL 01	P_1				
55 +	ΣP_5				
56 XEQ 10	M_5'				

Step	X	Y	Z	T	L
57 RCL 07	ΣP_{front}				
58 -	M_5				
59 GTO 22					
60♦LBL "3S3"					
61 XEQ 13	M_1				
62 XEQ 03	M_3				
63 XEQ 04	M_4				M_4 for rear 3 wheels (trailer)
64 RCL 03	P_3				
65 6	(4'+2')				
66 *	$\Sigma P_3 l_3$				
67 RCL 06	l_2				
68 2					
69 +					
70 RCL 02	P_2				
71 *					
72 -	$\Sigma P \cdot d_5$				
73 ST+ X					
74 RCL 11	$\Sigma P_{rear\,4}$				
75 RCL 02	P_2				
76 +	$\Sigma P_{rear\,5}$	$\Sigma P d_5$			
77 XEQ 10	M_5'				
78 RCL 03	P_3				
79 12	(8'+4')				
80 *					
81 -	M_{5A}				M_{5A} under 1st axle of triple
82 XEQ 22					
83 RCL 06	l_2				
84 RCL 09	$-d_{5A}$				
85 +					
86 CHS	d_{5B}'				
87 XEQ 05	M_{5B}'				
88 RCL 02	P_2				
89 4					
90 *	$4' \cdot P_2$				
91 STO 07					
92 -	M_{5B}				M_{5B} under 2nd axle of tractor dual
93 XEQ 22					
94 RCL 05	l_1				
95 4					
96 +					
97 RCL 01	P_1				
98 *	$P_1(l_1+4)$				
99 ST+ 07					ΣPd_{front} in R.0
100 RCL 09	$-d_{5A}$				
101 RCL 06	l_2				
102 +	d_{5B}				
103 RCL 11	ΣP_5				
104 *					
105 -	$-\Sigma Pd_6$				
106 RCL 11	ΣP_5				
107 RCL 01	P_1				
108 +	ΣP_6				
109 XEQ 10	M_6'				
110 RCL 07					
111 -					
112 GTO 22	M_6				

D239

Program Trucks For ________ ______ bytes Page 2 of 3

Step	X	Y	Z	T	L
113◆LBL "T4"					
114 RCL 03	P_2				
115 XEQ 11	M_1				
116◆LBL 04					
117 RCL 03	P_2				
118 XEQ 12	M_2				
119 RCL 03	P_2				
120 RCL 00	Span				
121 3					
122 *					
123 4					
124 /	$3S/4$	P_2			
125 LASTX	4				
126 -	$\frac{3S}{4}-4$				$M_{triple} = \frac{(3P)\cdot \ell}{4} - 4'\cdot P$
127 *					
128 XEQ 22	M_{triple}				
129 RCL 02	P_1				
130 RCL 06	ℓ				
131 4					
132 +	$\ell-4$				
133 *	$P_1(\ell-4)$				
134 CHS	$-P\cdot d$				
135 RCL 03	P_2				
136 3					
137 *	ΣP_{triple}				
138 RCL 02	P_1				
139 +	ΣP	$-\Sigma P\cdot d$			
140 XEQ 10	M'				
141 RCL 03	P_2				
142 4					
143 *	$4P$				
144 -	M_{4A}				$M_{4\cdot A}$ under 3rd wheel
145 XEQ 22					
146 RCL 09	$-\ell_A$				
147 4					
148 +	$-\ell_B$				
149 CHS					
150 XEQ 05	M'				
151 RCL 02	P_1				
152 RCL 06	ℓ_1				
153 *	$P_1\ell_1$				
154 -	M_{4B}				$M_{4\cdot B}$ under 2nd wheel
155 GTO 22					
156◆LBL "T3"					
157 RCL 01					
158 XEQ 11	M_1				
159◆LBL 03					
160 4					
161 STO 04	ℓ_2				
162 RCL 02	P_2				
163 XEQ 12	M_2				
164 RCL 02					
165 RCL 02					
166 RCL 01					
167 RCL 05	ℓ_1	P_1	P_2	P_2	

Step	X	Y	Z	T	L
168◆LBL 14	ℓ_1	P_1	P_2	P_{2+3}	
169 X<>Y	P_1	ℓ_1			
170 ST+ Z			ΣP		
171 *	$P_1\ell_1$	ΣP	P_{2+3}	P_{2+3}	
172 STO 07					
173 RCL 04	ℓ_2				
174 R↑	P_{2+3}	ℓ_2	$P_1\ell_1$	ΣP_{1-2}	
175 ST+ T				ΣP	
176 *	$P_2\ell_2$	$P_1\ell_1$			
177 -	$-\Sigma P\ell$	ΣP			
178 X<>Y	ΣP	$-\Sigma P\ell$			
179 XEQ 10	M_3'				
180 RCL 07	$P_1\ell_1$				
181 -	M_3				
182 GTO 22					
183◆LBL "2S1"					
184 XEQ 13	M_1				
185 RCL 03					
186 RCL 02					
187 RCL 01					
188 RCL 05	ℓ_1	P_1	P_2	P_3	
189 XEQ 14	M_3				
190 RCL 03					
191 RCL 02					
192 RCL 04	ℓ_2	P_2	P_3		
193 XEQ 02	$M_{2\cdot rear}$				
194 RCL 01					
195 RCL 02					
196 RCL 05	ℓ_1	P_2	P_1		
197 GTO 02					
198◆LBL 12	P				
199 ENTER↑					
200 ENTER↑	P	P	P		
201 4	4	P	P		
202 GTO 02					
203◆LBL "T2"					
204 RCL 01	P_1				
205 XEQ 11					
206 RCL 01	P_1				
207 RCL 02	P_2				
208 RCL 05	ℓ				
209◆LBL 02					
210 RDN	P_2	P_1		ℓ	
211 X>Y?					
212 X<>Y	P_{min}	P_{max}		ℓ	
213 ST+ Y	P_{min}	ΣP			
214 R↑	ℓ				
215 *	$P_{min}\cdot\ell$	ΣP			
216 CHS					
217 X<>Y	ΣP	$-P_1\ell$			
218 XEQ 10	M_2				

D240

	X	Y	Z	T	L
219♦LBL 22	M				
220 RCL 12	M_{max}	M			
221 X<=Y?					
222 X<>Y					
223 STO 12	M_{max}				
224 RTN					
225♦LBL 10					
226 STO 11	ΣP	-Pℓ			
227 /	-ℓ				
228 STO 09					
229♦LBL 05					
230 RCL 00					
231 +	Span-ℓ				
232 2					
233 /	$\frac{S}{2}-\frac{\ell}{2}$				
234 X>0?					
235 X↑2	$(\ )^2$				
236 RCL 11	ΣP				
237 *					
238 RCL 00	span				
239 /					
240 RTN	M′				
241♦LBL 13					
242 RCL 01	P_1				
243 RCL 03	P_3				
244 X<Y?					
245 X<>Y	P_{max}				
246♦LBL 11	P				
247 RCL 02	P_2				
248 X<Y?					
249 X<>Y	P_{max}				
250 RCL 00	Span				
251 *					
252 4					
253 /	Pℓ/4				
254 STO 12	M_1				
255 RTN					
256♦LBL "CO"					
257 XEQ 09	dec.deg				
258 COS					
259 GTO 07					
260♦LBL "SI"					
261 XEQ 09	dec.deg				
262 SIN					
263 GTO 07					
264♦LBL "TA"					
265 XEQ 09	dec.deg				
266 TAN					
267 GTO 07					

	X	Y	Z	T	L
268♦LBL "ACO"					
269 STO ↑					
270 ACOS					
271 GTO 08					
272♦LBL "ASI"					
273 STO ↑					
274 ASIN					
275 GTO 08					
276♦LBL "ATA"					
277 STO ↑					
278 ATAN					
279♦LBL 08					
280 FS? 03					
281 HMS					
282♦LBL 07	result				angle
283 X<> L	angle-dec				result
284 CLX					
285 X<> ↑	orig.angle				note: clears Reg. P
286 X<> L	result				orig. angle
287 RTN					
288♦LBL 09					
289 STO ↑	angle				
290 FS? 03					
291 HR	Dec.deg				
292 END					

DZ41

D242

RETAINING WALL ANALYSIS

xeq RW (SIZE 016)

SOFTKEYS (USER mode ON)

d	Jump directly to wall DEPTH prompt
F	New footing dimensions
H	New height and dimensions

Note: softkeys A, B, and D are used by program SSD and are active while RW is being run

USER FLAGS

Flag 00	clear	Full output
	set	Partial vert. loads and moments skipped
Flag 03	clear	Slope and Phi angles in decimal degrees
	set	Angles in degrees-minutes-seconds

INPUT SUMMARY

Prompt	Input	Default
SLOPE	Slope angle from horizontal	Zero
PHI	Angle of internal friction	No default
VERT E PCF	Vertical earth weight in pounds per cubic foot	No default
SUR PSF	Vertical surcharge in pounds per square foot	No default
LAT E PCF	Lateral earth load expressed as an equivalent fluid, in pcf	C_a x Vert.
SUR PSF	Lateral surcharge in psf	No default
FRIC μ	Coeff. of sliding friction, mu	No default
HT'	Height from bottom of footing to top of backfill, in feet	No default
F TH'	Footing thickness in feet	No default
COV'	Soil depth over the toe (front) of the footing, measured from the top of the footing	No default
T/W TH' B/W TH'	Thickness of the cantilever wall at top and bottom of the wall	No default

D243

Prompt	Input	Default
P, K	Additional vertical load applied to the top of the stem, in kips (e.g. rail DL or part of the wall above top of backfill)	No default
P CG'	Location of the load P, measured back from the front face of the stem, in feet	No default
FTG W'	Footing width in feet	No default
TOE'	Distance from the front of the footing to the front face of the stem (wall)	No default
D'	Location along the wall (depth) at which to figure moment and shear, measured down from top of backfill	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

Note: all moments caused by vertical loads are taken about the toe of the footing.

C_a	Coefficient of active pressure
(CONC: V,M)	Vertical load and moment due to the footing and stem, in kips and foot-kips
(P: V,M)	Vertical load and moment due to the load applied to the top of the wall
(VERT E: V,M)	Vertical load and moment due to vertical earth load
(SUR: V,M)	Vertical load and moment due to vertical surcharge. Only if prompt was answered.
(ΣV, M, RES)	Sum of the vertical loads and the moments due to these loads about the toe of the footing, and the position of the resultant measured in feet from the toe.
SLIDE FS	Sliding factor of safety, neglecting passive resistance and any key present
TOE P HEEL P	Pressure under the heel and toe for a spread footing
LAT E: H,M SUR: H,M Σ H,M	Horizontal shear and moment due to lateral earth pressure and surcharge, and their summations.

D244

PROGRAM FLAGS

RW uses no program flags.

STORAGE REGISTER USE

Register	Value
00	Summation of vertical forces
01	Summation of moments
02	Lateral earth equiv. fluid pressure (k/cf stored)
03	HT' (height from bottom of footing)
04	F TH'
05	Footing width
06	TOE' (distance from front of footing to the stem)
07	ST TH' (stem thickness)
08	Lateral surcharge (ksf stored)
09	Coefficient of friction, mu
10	P, K (rail DL)
11	C.G. P (point of application of rail DL from the front face of the wall)
12	COV' (Soil cover over the toe)
13	Vertical earth weight (k/cf stored)
14	Vertical surcharge (ksf stored)
15	Working register

D245

RETAINING WALL ANALYSIS

Program RW calculates the loads and reactions due to selfweight, soil, and surcharge (if any) on cantilever retaining walls with cohesionless backfill. Input consists of the wall and footing dimensions, soil parameters, and surcharge loadings. RW calculates the resulting moments and horizontal and vertical forces at bottom of footing, bottom of wall, and any desired points along the wall, as well as the heel and toe reactions and sliding factor of safety (assuming a spread footing).

SOIL PROPERTIES

RW prompts first for the backfill angle from the horizontal (see sketch), the internal friction angle O (phi) of the soil, and the coefficient of friction (mu) between soil and footing concrete. Typical values for mu, from the 1958 AREA specifications, are as follows:

Soil	Mu
Sound rock, rough surface	0.60
Coarse-grained, no silt	0.55
Coarse-grained, with silt	0.45
Silt	0.35

The program solves the Rankine/Coulomb formula for the active pressure coefficient C_a for lateral loading. This value is not used directly by the program; however, it is multiplied by the soil density (usually about 120 pcf) to calculate the default value for the equivalent fluid density used to calculate lateral pressure (usually about 35 pcf). If the designer agrees with the assumptions involved in this formula (cohesionless soil; essentially vertical pressure surface with a prescribed amount of friction between soil and wall), the default lateral earth pressure may be used. The formula used is:

$$C_a = \cos(\beta) \times \frac{\cos(\beta) - \sqrt{\cos(\beta) - \cos(\phi)}}{\cos(\beta) + \sqrt{\cos(\beta) - \cos(\phi)}}$$

β = backfill angle (0° = horizontal)
ϕ = angle of internal fricion
cohesion assumed zero

The user inputs the actual soil weight for vertical loading, and the equivalent fluid weight for horizontal loading, using the default if desired. If the user wants to check a range of soil values (e.g. a lighter soil weight for potential heel uplift), separate runs must be made.

D246

The user may also input surcharge loading, both vertical and horizontal, if desired. This is input in psf.

OUTPUT

The program calculates a force (vertical or horizontal) and a moment due to each component of load. These are output separately (in order to allow for either working stress or load factor methods of design) and then summed. If flag 00 is set, only the sums are displayed. An average wall thickness is used in the calculations; this leads to a minor error in the calculation of vertical surcharge loading.

After calculating vertical and horizontal forces and moments about the bottom of footing, the program calculates the sliding factor of safety and the toe and heel bearing stresses. These are valid, of course, only for spread footings. The sliding factor of safety is calculated using the input friction factor "mu" (μ) or the value of tan Ø, whichever is less. The former represents sliding between the footing and the soil; the latter represents shear failure of the soil under the footing. (Thus, the PHI prompt should not be bypassed, since it affects sliding as well as the default lateral load value.) The safety factor does not include passive resistance in front of the toe, or the effect of any key. RW ignores all vertical friction forces.

After outputting resulting forces, moments, and reactions at top and bottom of footing, the program calculates the horizontal force and moment at any point on the wall, e.g. to aid in reinforcing design and cutoff. RW will continue to do this until the DEPTH' prompt is bypassed, at which point a new footing size will be prompted for. (Softkey F will do the same thing.) Thus, many footing trials may be run easily.

SOFTKEYS

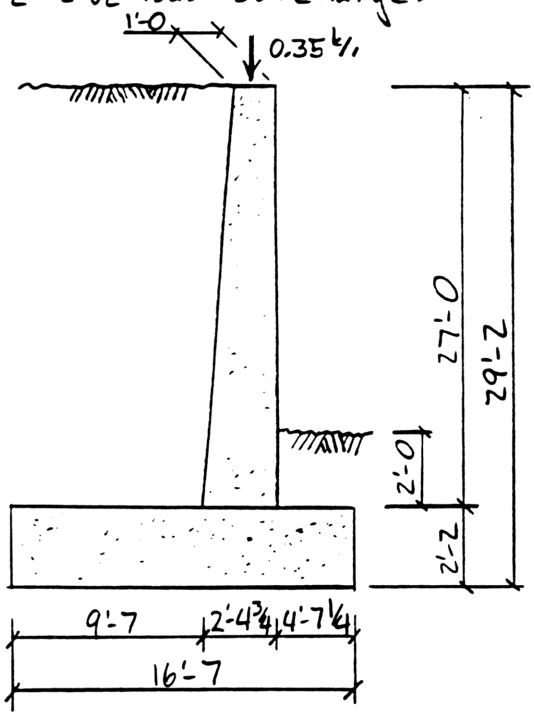
For design of sheetpile walls or to check reinforcing in a concrete wall, allow the program to prompt for soil parameters, then use softkey d (lowercase) to jump to the DEPTH' prompt for different points along the wall.

As mentioned, softkey F will bring up prompts for a new footing. Softkey H (for Height) will bring up prompts for a whole new wall configuration. This is the same as re-executing the program except it saves re-inputting soil parameters.

D247

Retaining Wall Analysis

Analyze this wall for
2' Live load surcharge.



Vert. E = 120 pcf
Vert. Surch. = 0

Horiz. E = 33 pcf
Horiz. Surch. = 66 psf (LL)

ϕ = 35°
Backslope is flat
Friction μ = .45

Parapet DL = 2.33' × 1' × .15 = .35 k/'

Vert. Load ΣM about toe

	V		M
Ftg. 2.17' × 16.58' × .15 =	5.39 k	× $\frac{16.58}{2}$ =	44.69 '-k
Wall ½(1' + 2.40') × 27' × .15 =	6.89	× 5.45' =	37.52
(Partial Summation)	(12.28)		(82.21)
Heel E avg. 10.28' × 27' × .12 =	33.32	× 11.44' =	381.21
Toe E 4.60' × 2' × .12 =	1.10	× 2.30' =	2.54
Parapet	.35	× 5.10' =	1.78
	47.05 k		467.74 '-k

eccentricity = $\frac{467.74}{47.05}$ = 9.94'

Horiz. Load ΣM about bottom/ftg

		H		M
Earth	$\frac{1}{2}$ × .033 × 29.17² =	14.04	× $\frac{29.17}{3}$ =	136.47
Surch	.066 × 29.17 =	1.93	× $\frac{29.17}{2}$ =	28.07
		15.97		164.54

Overturning Moments about toe

$$F.S. = \frac{\text{Resist. M}}{\text{O.T. M}} = \frac{467.74}{164.54} = 2.84 > 2.0$$

OK

Note: "RW" does not calculate overturning F.S. The designer must do this.

Sliding Neglect passive toe resist.

silty coarse-grained soil ⇒ μ = .45

tan(ϕ = 35°) = .70 > .45 OK

$$F.S. = \frac{\mu \times V}{H} = \frac{.45 \times 47.05}{15.97} = 1.33$$

1.33 < 1.50 desired but say OK

with passive resist. and shear key

Bearing

Ftg $S = \frac{16.58^2}{6}$ = 45.83 ft³/ft

Vert. 47.05/16.58	= 2.84
Vert. M 47.05 × (9.94' − $\frac{16.58}{2}$)/45.83	= −1.69
Horiz. M 164.54/45.83	= 3.59

Toe → 2.84 − 1.69 + 3.59 = 4.74 ksf

Heel → 2.84 + 1.69 − 3.59 = 0.94 > 0

Need ~ 5.0 ksf bearing

Stem Analysis

@ Top of Ftg. :

		H		M
E	$\frac{1}{2} \times .033 \times 27^2 =$	12.03 k	$\times \frac{27}{3} =$	108.26 'k
Surch	$.066 \times 27 =$	1.78	$\times \frac{27}{2} =$	24.06
		13.81 k		132.32 'k

Design stem reinf. for 132.32 'k

Check for 13.81 k shear

@ 10' Depth (say) for bar cutoff

		H		M
E	$\frac{1}{2} \times .033 \times 10^2$	= 1.65	$\times \frac{10}{3}$	= 5.50
Surch.	$.066 \times 10$	= .66	$\times \frac{10}{2}$	= 3.30
		2.31 k		8.80 'k

Run 'RW' with Flag 00 set (Short Form)

```
           SF 00
        XEQ "RW"
SLOPE (dd)
             RUN
PHI (dd)
   35.0000   RUN
Ca=0.2710
---------
VERT E PCF
  120.0000   RUN
SUR PSF
    0.0000   RUN
LAT E PCF
   33.0000   RUN
SUR PSF
   66.0000   RUN
FRIC μ
     .4500   RUN
---------
```

No title block

Input soil properties

```
HT'
   29.1667   RUN
F TH'
    2.1667   RUN
COV'
    2.0000   RUN
T/W TH'
    1.0000   RUN
B/W TH'
    2.4000   RUN
P, K
     .3500   RUN
P CG'
     .5000   RUN
************************
FTG W'
   16.5833   RUN
TOE'
    4.6000   RUN
---------
```

Input wall parameters

```
---------
ΣV=47.0465 K
M=467.7482 'K
RES@9.9422'
------------------------
B/FTG:

ΣH=15.9615 K
M=164.5390 'K

SLIDE FS=1.3264

TOE P=4.7326 KSF
HEEL P=0.9414 KSF
---------
T/FTG:

ΣH=13.8105 K
M=132.3135 'K
------------------------
D'
   10.0000   RUN

ΣH=2.3100 K
M=8.8000 'K
------------------------
D'
```

Formatting is a bit loose with Flag 00 set

No partial sums

Full output - Flag 00 clear

```
            XEQ "RW"
 4:45:09 PM  02/02/90

TITLE:
LARP. AVE. WALL       RUN
************************
SLOPE (dd)
            0.0000    RUN
PHI (dd)
           35.0000    RUN
Ca=0.2710
---------
VERT E PCF
          120.0000    RUN
SUR PSF
            0.0000    RUN
LAT E PCF
           33.0000    RUN
SUR PSF
           66.0000    RUN
FRIC μ
             .4500    RUN
---------
HT'
           29.1667    RUN
F TH'
            2.1667    RUN
COV'
            2.0000    RUN
T/W TH'
            1.0000    RUN
B/W TH'
            2.4000    RUN
P, K
             .3500    RUN
P CG'
             .5000    RUN
************************
FTG W'
           16.5833    RUN

TOE'
            4.6000    RUN
---------
CONC: V=12.2747 K
M=82.2124 'K

P: V=0.3500 K
M=1.7850 'K

TOE E: V=1.1040 K
M=2.5392 'K

HEEL E: V=33.3179 K
M=381.2117 'K

---------
ΣV=47.0465 K
M=467.7482 'K
RES@9.9422'
-----------------------
B/FTG:
LAT E: H=14.0365 K
M=136.4660 'K

SUR: H=1.9250 K
M=28.0730 'K

ΣH=15.9615 K
M=164.5390 'K

SLIDE FS=1.3264

TOE P=4.7326 KSF
HEEL P=0.9414 KSF
---------
T/FTG:
LAT E: H=12.0285 K
M=108.2565 'K

SUR: H=1.7820 K
M=24.0570 'K

ΣH=13.8105 K
M=132.3135 'K
---------
D'
           10.0000    RUN
LAT E: H=1.6500 K
M=5.5000 'K

SUR: H=0.6600 K
M=3.3000 'K

ΣH=2.3100 K
M=8.8000 'K

---------
```

Program RW For Retaining Wall Analysis ______ bytes Page 1 of 3

	X	Y	Z	T	L

```
01♦LBL "SSD"
  ⋮
114 GTO 06

115♦LBL "RW"
116 15
117 XROM "S"
118 CLX
119 "SLOPE"
120 XROM "Pd"
121 COS
122 ENTER↑
123 ENTER↑
124 X↑2
125 "PHI"
126 XROM "Pd"
127 STO 09
128 COS
129 X↑2
130 -
131 SQRT
132 ST- Z
133 +
134 /
135 *
136 STO 05
137 "Ca="
138 ARCL X
139 AVIEW
140 XROM "L"
141 "VER"
142 XEQ 04
143 STO 14
144 RDN
145 STO 13
146 LASTX
147 *
148 RCL 05
149 *
150 "LA"
151 XEQ 04
152 STO 08
153 RDN
154 STO 02
155 "FRIC µ"
156 PROMPT
157 RCL 09
158 TAN
159 X>Y?
160 X<>Y
161 STO 09
162♦LBL H
163 XROM "L"
164 "HT'"
165 PROMPT
166 STO 03
167 "F TH'"
168 PROMPT
169 STO 04
170 "COV'"
171 PROMPT
172 STO 12
173 "T/W TH'"
174 PROMPT
175 STO 07
176 "B/W TH'"
177 PROMPT
178 RCL 07
179 +
180 2
181 /
182 STO 07
183 "P, K"
184 PROMPT
185 STO 10
186 "P CG'"
187 PROMPT
188 STO 11

189♦LBL F
190 XROM "L*"
191 "FTG W'"
192 PROMPT
193 STO 05
194 "TOE'"
195 PROMPT
196 STO 06
197 XROM "L"
198 RCL 04
199 RCL 05
200 *
201 STO 00
202 LASTX
203 *
204 RCL 03
205 RCL 04
206 -
207 RCL 07
208 *
209 ST+ 00
210 RCL 06
211 ST+ X
212 LASTX
213 +
214 *
215 +
216 .15
```

D251

Program RW For Retaining Wall Analysis ______ bytes Page 2 of 3

	X	Y	Z	T	L

```
217 ST* 00
218 *
219 2
220 /
221 STO 01
222 RCL 00
223 X<>Y
224 "CONC"
225 XEQ 14
226 ADV
227 RCL 10
228 ST+ 00
229 RCL 06
230 RCL 11
231 +
232 RCL 10
233 *
234 ST+ 01
235 "P"
236 XEQ 14
237 ADV
238 RCL 13
239 RCL 12
240 *
241 RCL 06
242 *
243 ENTER↑
244 ST+ 00
245 LASTX
246 *
247 2
248 /
249 ST+ 01
250 "TOE E"
251 XEQ 14
252 RCL 03
253 RCL 04
254 -
255 RCL 13
256 *
257 "HEEL E"
258 XEQ 11
259 RCL 14
260 "SUR"
261 XEQ 11
262 XROM "L"
263 "ΣV="
264 RCL 01
265 RCL 00
266 XEQ 13
267 /
268 "RES@"
269 ARCL X
270 XEQ "FT"
271 AVIEW
```

	X	Y	Z	T	L

```
272 RCL 05
273 2
274 /
275 -
276 RCL 00
277 *
278 STO 01
279 XROM "LL"
280 "B"
281 RCL 03
282 XEQ 05
283 RCL 00
284 RCL 09
285 *
286 X<>Y
287 /
288 ADV
289 "SLIDE FS="
290 ARCL X
291 AVIEW
292 ADV
293 RCL 00
294 RCL 05
295 /
296 RCL 01
297 6
298 RCL 05
299 X↑2
300 /
301 *
302 ST+ Z
303 -
304 "TOE"
305 XEQ 10
306 X<>Y
307 "HEEL"
308 XEQ 10
309 XROM "L"
310 RCL 03
311 RCL 04
312 -
313 "T"
314 XEQ 05

315♦LBL d
316 XROM "LL"
317 "D'"
318 CLX
319 PROMPT
320 X=0?
321 GTO F
322 XEQ 02
323 GTO d
```

D252

Program RW For Retaining Wall Analysis ______ bytes Page 3 of 3

	X	Y	Z	T	L		X	Y	Z	T	L

```
324♦LBL 05
325 "⊦/FTG:"
326 AVIEW

327♦LBL 02
328 ENTER↑
329 ENTER↑
330 X↑2
331 2
332 /
333 STO 15
334 RCL 02
335 *
336 ST* Y
337 X<>Y
338 3
339 /
340 "LAT E"
341 XEQ 12
342 ADV
343 R↑
344 RCL 08
345 *
346 RCL 15
347 ST* L
348 X<> L
349 "SUR"
350 XEQ 12
351 ADV
352 ST+ Z
353 X<> T
354 +
355 X<>Y
356 "ΣH="
357 GTO 13

358♦LBL 11
359 RCL 05
360 RCL 06
361 -
362 RCL 07
363 -
364 STO T
365 *
366 ST+ 00
367 RCL 05
368 R↑
369 2
370 /
371 -
372 *
373 ST+ 01
374 ADV
```

```
375♦LBL 14
376 X<>Y
377 FC? 00
378 X=0?
379 RTN
380 "⊦: V="
381 GTO 13

382♦LBL 12
383 ST- 01
384 X<>Y
385 FC? 00
386 X=0?
387 RTN
388 "⊦: H="

389♦LBL 13
390 ARCL X
391 "⊦ K"
392 AVIEW
393 "M="
394 ARCL Y
395 XROM "FK"
396 AVIEW
397 RTN

398♦LBL 04
399 "⊦T E PCF"
400 PROMPT
401 "SUR PSF"
402 PROMPT
403  E3
404 ST/ Z
405 /
406 RTN

407♦LBL 10
408 "⊦ P="
409 ARCL X
410 "⊦ KSF"
411 X<0?
412 "⊦:UPL"
413 AVIEW
414 END
```

D253

Program RW For Retaining Wall Analysis ______ bytes Page 1 of 2

Step	X	Y	Z	T	L
01♦LBL "SSD"					
⋮					
114 GTO 06					
115♦LBL "RW"					
116 15					
117 XROM "S"					
118 "LA"					
119 XEQ 04					
120 STO 08	Lat Sur				
121 RDN					
122 STO 02	Lat E				
123 FS? 00					
124 GTO 03					
125 "VER"					
126 XEQ 04					
127 STO 14	Vert Sur.				
128 RDN					
129 STO 13	Vert E.				
130♦LBL H					
131 XROM "L"					
132 "HT'"					
133 PROMPT					
134 STO 03	Ht.				
135 "F TH'"					
136 PROMPT					
137 STO 04	F th				
138 "COV'"					
139 PROMPT					
140 STO 12	cover				
141 "ST TH'"					
142 PROMPT					
143 STO 07	Stem th				
144 "P, K"					
145 PROMPT					
146 STO 10	P				
147 "P CG'"					
148 PROMPT					
149 STO 11	C.G.				
150♦LBL 01					
151 XROM "L"					
152 "FTG W'"	Ftg W				
153 PROMPT					
154 STO 05					
155 "TOE'"					
156 PROMPT					
157 STO 06	Toe				
158 XROM "LL"					
159 RCL 04	Ftg th				
160 RCL 05	Ftg W				
161 *					
162 STO 00	Ft. A				
163 LASTX	Ftg W				
164 *	M'				
165 RCL 03	Ht				
166 RCL 04	F th				
167 -	ht'				
168 RCL 07	Stem Th				
169 *	Stem A				
170 ST+ 00					
171 RCL 06	Toe				
172 ST+ X	2·Toe				
173 LASTX	Stem Th				
174 +	Dist.				
175 *	Stem M				
176 +	M				
177 .15	conc. wt.				
178 ST* 00					
179 *					
180 2					
181 /	M				
182 STO 01					
183 RCL 00	V	M			
184 "CONC"					
185 XEQ 14					
186 RCL 10	P				
187 ST+ 00	(ΣV)				
188 RCL 06	Toe				
189 RCL 11	C.G.				
190 +	C.G.				
191 *	M				
192 ST+ 01					
193 RCL 10	P	M			
194 "P"					
195 XEQ 14					
196 RCL 13	.120				
197 RCL 12	Cover				
198 *					
199 RCL 06	Toe				
200 *	Toe V				
201 STO 15					
202 LASTX	Toe	V			
203 *	Toe M				
204 STO 09					
205 RCL 03	Ht.				
206 RCL 04	Ftg Th.				
207 -	ht.				
208 RCL 13	.120				
209 *	Earth V				
210 "VERT E"					
211 XEQ 11					
212 CLX					
213 STO 09					
214 STO 15					
215 RCL 14	Sur V				
216 "SUR"					
217 XEQ 11					
218 XROM "L"					
219 "ΣV="					
220 RCL 01	M				
221 RCL 00	V				
222 XEQ 13					
223 /	Res.				

Program RW For Retaining Wall Analysis ______ bytes

Step	X	Y	Z	T	L
224 "RES@"					
225 ARCL X					
226 AVIEW					
227 XROM "LL"					
228 "B"					
229 RCL 03	Ht.				
230 XEQ 05					
231 RCL 03	Ht.				
232 RCL 04	Ftg th				
233 -	Stem Ht				
234 "T"					
235 XEQ 05					
236♦LBL 03					
237 "D'"					
238 CLX					
239 PROMPT	Depth				
240 X=0?					
241 SF 14					
242 X=0?					
243 GTO 01					
244 XEQ 02					
245 GTO 03					
246♦LBL 05					
247 "⊢/FTG:"					
248 AVIEW					
249♦LBL 02	h				
250 ENTER↑	h				
251 ENTER↑	h	h	h		
252 X↑2	h^2				
253 2	2	h^2	h	h	
254 /	$h^2/2$				
255 STO 15					
256 RCL 02	Lat.E	$h^2/2$	h		
257 *	$wh^2/2$				
258 ST* Y		$wh^3/2$	h	h	
259 X<>Y					
260 3					
261 /	$wh^3/6$	$wh^2/2$	h	h	
262 "LAT E"					
263 XEQ 12					
264 R↑	h				
265 RCL 15	$h^2/2$				
266 RCL 08	Lat Sur	$h^2/2$	h		
267 ST* Z			wh		
268 *	$wh^2/2$	wh			
269 "SUR"					
270 XEQ 12					
271 XROM "L"					
272 RTN					

Step	X	Y	Z	T	L
273♦LBL 11	W				
274 RCL 05	FtgW				
275 RCL 06	Toe				
276 -					
277 RCL 07	StemTh				
278 -	Heel				
279 STO T	Heel	W		heel	
280 *	V				
281 ST+ 15	(ΣV)				
282 RCL 05	FtgW				
283 R↑	heel				
284 2					
285 ST/ 09					
286 /	heel/2	Ftg W	V		
287 -	Arm	V			
288 *	M				
289 RCL 09	M_{Toe}				
290 +	ΣM				
291 ST+ 01					
292 RCL 15	ΣV				
293 ST+ 00					
294♦LBL 14					
295 X=0?					
296 RTN					
297 ADV					
298 AVIEW					
299 "V="					
300 GTO 13					
301♦LBL 12	M	H			
302 X<>Y	H	M			
303 X=0?					
304 RTN					
305 AVIEW					
306 "H="					
307♦LBL 13	V or H	M			
308 ARCL X					
309 "⊢ K"					
310 AVIEW					
311 "M="					
312 ARCL Y					
313 XROM "FK"					
314 AVIEW					
315 RTN					
316♦LBL 04					
317 "⊢T E PCF"					
318 PROMPT					
319 "SUR PSF"	Surch	Earth			
320 PROMPT					
321 E3	1000				
322 ST/ Z					
323 /	Surch	Earth			
324 END					

ELASTOMERIC BEARING PAD ANALYSIS

xeq NEO (SIZE 017)

SOFTKEYS (USER mode ON)

D Input new shim thickness and pad dimensions

USER FLAGS

Flag 00	clear	Full output
	set	Skips output of shape factor, dead load stress, and reduced pier forces
Flag 02	clear	Warning BEEP disabled
	set	Warning BEEP enabled
Flag 03	clear	Pier SKEW in decimal degrees
	set	SKEW in degrees-minutes-seconds
Flag 04	clear	AASHTO
	set	AASHTO (ignores flag 04)

INPUT SUMMARY

Prompt	Input	Default
HARD	Neoprene durometer hardness	No default
LOW G PSI HI G PSI	Low- and high-end values of shear modulus. Default per AASHTO table 14.2.2A.	AASHTO
DEL-T"	Movement due to temperature (one way), in inches (zero for a fixed pad)	No default
DL K/PAD	Dead load, kips per pad (reduced by live load uplift, if any)	No default
D+L K/PAD	Maximum dead and live load	No default
# PADS	Number of bearing pads on one pier. If bypassed (or one is entered), pier properties are not prompted and pier flexibility is neglected.	One
PIER HT'	Pier height from point of fixity to beam seat, in feet. If zero is entered, pier properties are ignored.	No default

D255

Prompt	Input	Default
SKEW (dd or DMS)	Skew of the pier	
# COLS	Number of pier columns. Zero (or bypass) for a T-pier.	0 (T-pier)
DIA' (If #COLS is not zero)	Column diameter in feet	No default
SH TH' (T-piers)	Shaft thickness in feet	No default
W' (T-piers only)	Shaft width, feet	No default
SHIM TH 16'S	Metal shim thickness in six-teenths of an inch. May bypass for unreinforced pads. Answer 2 for 1/8"shims, 3 for 3/16".	Two (1/8")
PAD L"	Pad length in inches	No default
W"	Pad width in inches. Inter-changeable with length.	No default
#SHIMS	Number of shims. Bypass for unreinforced pads.	No shims
LAM TH"	Lamination thickness in inches (for the lams between the shims). Program assumes 1/4" thick cover layers outside the shims.	No default
TH"	Thickness of the (unreinforced) pad in inches, if #SHIMS prompt was answered zero or bypassed	No default

OUTPUT SUMMARY (Items in parenthesis are displayed only if flag 00 is clear)

TH	Thickness of the (reinforced) pad in inches if SHIM TH was not answered zero (assumes 1/4" cover layers)
TH>MAX ***	Warns that bearing is too thick per AASHTO 14.2.7 (TH > W/3 or W/5)
(SH F)	Shape factor (Area/Perimeter~~*~~ th.)
ALL STRS	Allowable compressive stress in kips per square inch. Fixed bearings are allowed a 10% increase (DEL-T" input as zero).
(DL STRS)	Dead load stress acting.
D+L STRS	Combined dead and live load stress acting.

D256

DL>MAX ***	Warns that dead load stress is more than 500 psi (guideline carried over from <u>1978</u> AASHTO)
DL<.2 ***	Warns that dead load is less than 200 psi (guideline carried over from 1978 AASHTO concerning anchorage of pads)
D+L>MAX ***	Warns that compressive stress is greater than calculated allowable
STRN	Compressive strain in percent. DuPont recommends strain be kept under 7%.
STR>7% ***	Warns against high strain. Reduce stress or increase shape factor.
TH<2T ***	Warns pad is too thin for shear deformation needed, per AASHTO 14.2.6
BOND ***	Warns pad needs to be secured against horizontal movement per AASHTO 14.2.9. No account is taken of pier flexibility for this warning.
F	Shear force in kips developed by the thermal deflection (DEL-T") input, considering the pier rigid
(Fe-11 and Fe-⊥)	Effective shear force developed parallel and perpendicular to the long axis of the pier, respectively, <u>taking into account the flexibility of the pier</u>
OK or NG	Trial pad design is OK or no good with respect to the above-mentioned checks

PROGRAM FLAGS

Flag 05	clear	Thermal movement
	set	Fixed pad (DEL-T" = 0)
Flag 06	clear	No pier properties
	set	Consider pier flexibility
Flag 07	clear	OK
	set	NG
Flag 09	clear	Multiple shims
	set	One shim (used rarely)

STORAGE REGISTER USE

Register	Value
00	DL K/PAD (Minimum load combination)
01	D+L K/PAD (Maximum load combination)
02	Lesser of pad length or width; pad area
03	Low-end shear modulus G (ksi stored)
04	Number of shims; 3 or 5 x Total thickness
05	Lamination thickness
06	Durometer hardness
07	Combined DL + LL stress
08	DEL-T (Thermal movement, one way)
09	Total elastomer thickness
10	High-end shear modulus x 1.9 (cold) x 5
11	Greater of pad length and width; shape factor
12	Skew (decimal degrees stored)
13	Pier height; $L^3/3EI$ of pier shaft or columns (C_p)
14	Number of pads per pier
15	Allowable compressive stress
16	Shim thickness (inches stored)

D254

ELASTOMERIC BEARING PAD ANALYSIS

Program NEO calculates properties and allowable stresses for elastomeric (neoprene) bearing pads by AASHTO Specifications, 1985 Revision. It uses AASHTO's compressive stress:strain curves. Default shear moduli are the high- and low-end values from AASHTO Table 14.2.2A; a cold-weather constant of 1.9 (recommended by DuPont for cold regions with temperatures down to $-20^{o}F$) is included in the shear-force calculations (high-end values) but not the allowable stress calculations (low-end values). The program allows elastomer durometer hardness values between 50 and 70; any other values may be assigned incorrect shear moduli.

PIER INTERACTION

NEO will analyze a bearing pad alone, or it will analyze a system of bearing pads and pier. If a thermal movement DEL-T" ($delta_{temp}$, inches) of zero is input, the bearing is assumed fixed and its allowable compressive stress is increased by 10%. If DEL-T" greater than zero is input, the program prompts for pier dimensions in order to calculate the effective shear force considering pier flexibility. The following assumptions are made:

1) Superstructure movement is purely longitudinal; deck does not rotate or deflect laterally on skewed piers.

2) Deck and beams are rigid in all directions.

3) Pier deflects only perpendicular to its long axis; it is assumed rigid along its long axis (parallel direction).

4) Pier is fixed at bottom. Definition of "bottom" depends on PIER HT' (pier height in feet) input by user, but is usually taken from top of footing.

5) The elastomeric material is at a temperature of $-20^{o}F$ and its stiffness has increased 90% over tabulated values. This may be somewhat over-conservative for warmer climates.

6) The pier concrete has a proportionality constant of N=8, and the gross moment of inertia of the pier shaft or columns is used.

7) The program assumes that a single pier shaft is rectangular, and that multiple columns are round. It considers the shaft or columns to be of uniform

D259

stiffness all the way to the bearings, ignoring the effect of the pier cap perpendicular to the long axis of the pier. (See the INPUT SUMMARY section.)

All but the last of these is conservative. If the bridge design involves a short T-pier, the designer may wish to input a PIER HT less than actual in recognition of the greater stiffness of the cap. For pier concrete of somewhat greater stiffness than indicated by N=8, assumption 6) is probably still conservative since the columns usually suffer at least minor cracking, decreasing their stiffness. The assumption of rectangular shape is pretty good for round-nosed T-pier shafts; just input a shaft length less than the actual by about one-half the shaft width. For rectangular column bents the user will have to calculate the DIA' (diameter in feet) of a round column of equal stiffness. For a single round column, answer the #COL prompt with 1; for a rectangular or roundnosed shaft respond with a zero.

DEFAULT SHEAR MODULI

The default values of G in psi are from AASHTO table 14.2.2A (except for 70 duro) and are as follows:

Duro	Low G	High G
50	85	110
55	102.5	132.5
60	120	155
70	155	200
(AASHTO 70)	(160)	(260)

SHEAR FORCE

For a non-fixed pier (one whose bearings absorb movement), NEO first calculates the shear force assuming the pads take all the thermal movement, i.e. the substructure is assumed rigid. If the #PADS prompt is bypassed (R/S is hit without keying in a number) or the PIER HT' prompt is answered zero, the program leaves its shear-force calculations at that. This, for instance, is appropriate for abutment bearings and very conservative pier designers. If these prompts are answered, however, NEO goes on to calculate the effective force developed considering the piers flexible using the above-noted assumptions.

Pad anchorage and shear checks are made for non-fixed piers using the full rigid-pier shear force (i.e. pier flexibility is not taken into account) whether or not pier information has been input. These checks are not made for fixed piers since the anchor bolts or

D260

other "fixing" mechanism are assumed to absorb the entire shear force.

The program assumes that elastomeric material harder than 60 durometer is used only in fixed, unreinforced pads, in accordance with AASHTO 14.2.2. If an expansion bearing (assumed reinforced) of 70 durometer material is analyzed using the default high-end shear modulus, the shear-force calculations will be about 25% low due to an error in the shear modulus calculation algorithm. If the user chooses to ignore 14.2.2, the default high-end modulus should not be used. The preceding section lists the defaults.

PAD GEOMETRY

The program works only with rectangular bearing pads, not round ones. It is not able to accommodate holes in pads. Pad length and width are interchangeable.

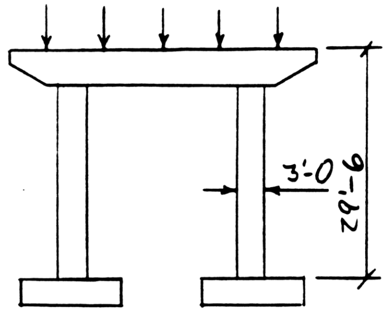
EXCEPTIONS

All AASHTO specifications are checked except 14.2.5, dealing with rotation, and 14.2.2, which forbids laminated pads of greater than 60 durometer hardness. If a pad fails to meet a given criterion, a message is displayed. At the end of the run, if a design has met all the checks, the program displays OK. If it has failed one or more of the checks, the program displays NG (no good) and, if flag 02 is set, a BEEP sounds.

OBSOLETE CHECKS

NEO checks against dead load stress of over 500 psi or under 200 psi; and total compressive strain of over 7%. The former are carried over as a guide from the 1978 AASHTO Specifications, while the latter comes from DuPont's 1959 guide to elastomeric bearing pad design. The program will label a design "NG" if it violates either of these; however, the design may still be valid by present (1989) AASHTO Specifications.

Design pads for 90' span (steel girders)



One-way $\Delta_T = (90' \times 12) \times (6.5 \cdot 10^{-6}) \times 75°F$
$= .5265"$

DL = 169.5 k/beam
Max LL = 107.99 k/beam
277.49

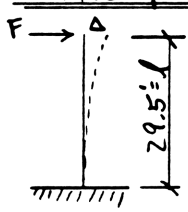
No uplift ⇒ min. LL = 0.0

5 beams / pier ; pier @ 10° skew

Assume:
1) columns fixed @ Top/Ftg.
2) deck rigid (beam spacing and orientation constant; no lateral movement by beams)
3) pier rigid parallel to frame
4) constant column EI all the way to the top.

*1-3 are conservative; *4 is slightly non-cons.

Derive pier flexural properties



$\Delta = F \times \frac{l^3}{3EI}$

$l = 29.5' = 354"$

Use $E = \frac{29{,}000}{n=8} = 3625$ ksi

$I = 2\text{ col.} \times \frac{\pi}{64}(36)^4 = 164{,}896"$

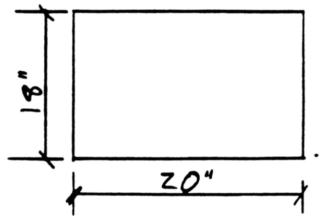
$\therefore \Delta_{pier} = F \times \frac{(354)^3}{3 \cdot 3625 \times I}$

$= F_{kips} \times .0247"/k$

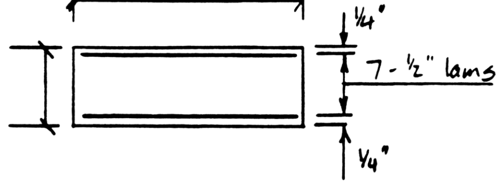
D262

Use 55-durometer per Mn/DOT specs.

Try 18"×20" pad w/ 3/16" shims, 7-1/2" lams



Neoprene: 2 × 1/4" = 1/2"
7 × 1/2" = 3 1/2"
4"
Total 8 × 3/16 = 1 1/2
5 1/2"



Stability (AASHTO 14.2.8) $\frac{W}{3} = 6" > 5\frac{1}{2}"$ OK

Anchorage (AASHTO 14.2.9) $F_{Horiz} = G \times A \times \Delta_T / T$

High Values
55 duro ⇒ $G = \frac{1}{2}(110+155) = 132.5$ psi

$A = 18" \times 20" = 360\,\square"$; $\Delta_T = .5265"$

T = neoprene thickness = 4"

$\therefore F_H = \frac{132.5 \times 360 \times .5265 \times 1.9^*}{4"} = 11.93$ k

Minimum Vert. F = 169.5 k > 5 × 11.93 OK

Shear (AASHTO 14.2.6) T = 4" > 2 × .5265" OK

Comp. Stress (AASHTO 14.2.3) Actual = 277.49 k/360 □"
= .7708 ksi

Shape Factor $= \frac{18" \times 20"}{2 \cdot (18+20) \cdot \frac{1}{2}} = 9.4737$

Low Values
Allowable stress $= \frac{\frac{1}{2}(85+120) \times 9.4737}{(\beta=1.0) \times 1000} = .9711$ ksi

.9711 > .7708 OK

* Cold-weather factor for −20°F.

Comp. Strain (AASHTO 14.2.4)

From tables 14.2.4, for stress of 770 psi:

50 duro → 3.70%
60 duro → 3.25%
avg. ~ 3.48%

As a guide, old AASHTO allowed max. 7%

3.48% < 7% **OK**

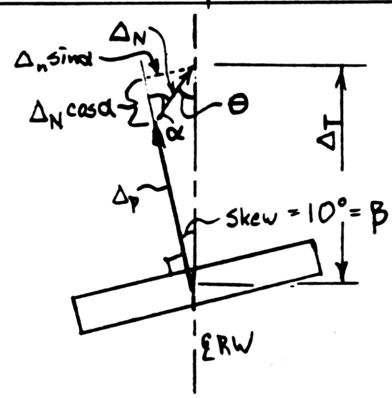
Also as a guide, check DL stress against 500 psi:

$DL = \frac{169.5\,k}{360\,\square''} = .47\,ksi < .5$ **OK**

all OK -

∴ **USE 18" × 20" × 5½" PAD**

Calculate actual pier force



$$\Delta_{Neo} = F \times \left(\frac{1000 \times 4''}{132.5 \times 360 \times 1.9 \times 5\,pads}\right) = F \times .0088\,''/k$$

$$\Delta_{Pier} = F \times .0247$$

$$\Delta_N \sin\alpha = \Delta_T \sin(\text{skew}) = .5265'' \times \sin 10° = .0914''$$

$$F_{e\text{-}\parallel} = .0914''/.0088''/k = 10.357\ k$$

$$\Delta_N \cos\alpha = \Delta_T \cos\beta - \Delta_p$$

and $\frac{\Delta_n \cos\alpha}{C_n} = \frac{\Delta p}{C_p}$

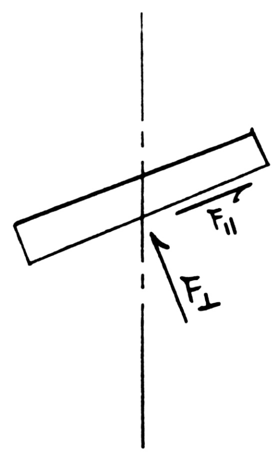
or $\Delta p = \frac{C_p \cdot \Delta_N \cos\alpha}{C_N}$

$$\therefore \Delta_T \cos\beta = \Delta_N \cos\alpha \left(1 + \frac{C_p}{C_N}\right)$$

$$C_N \Delta_T \cos\beta = \Delta_N \cos\alpha\,(C_N + C_p)$$

$$F_{e\text{-}\perp} = \frac{\Delta_N \cos\alpha}{C_N} = \frac{\Delta_T \cos\beta}{C_N + C_p} = \frac{.5265'' \times \cos 10°}{.0247 + .0088\,''/k} = 15.447\ k$$

Design pier for these effective thermal forces



D263

```
          XEQ "NEO"
 5:36:25 PM  02.03.89

TITLE:
I 394  PAD E-5        RUN
************************
HARD
          55.0000    RUN
DEL-T"
            .5265    RUN
DL K/PAD
         169.5000    RUN
MAX LL K/PAD
         107.9900    RUN
MIN LL K/PAD
                     RUN
# PADS
           5.0000    RUN
PIER HT'
          29.5000    RUN
SKEW (dd)
          10.0000    RUN
# COLS
           2.0000    RUN
DIA'
           3.0000    RUN
SHIM TH 16'S
           3.0000    RUN
-----------------------
PAD L"
          18.0000    RUN
W"
          20.0000    RUN
#3/16"SHIMS
           8.0000    RUN
LAM TH"
            .5000    RUN
TH=5.5000"
---------
SH F=9.4737
ALL STRS=0.9711 KSI

DL STRS=0.4708 KSI
D+L STRS=0.7708 KSI
STRAIN=3.4549%

F=11.9315 K/PAD
 =59.6577 K/PIER
Fe-11=10.3595 K/PIER
Fe-+=15.4179 K/PIER

 OK
-----------------------
```

total thickness (TH)
shape factor (SH F)
allowable (ALL STRS)
est. strain (STRAIN)
} equivalent forces taking pier flex. into account (Fe)
design is OK

```
PAD L"
          18.0000    RUN
W"
          20.0000    RUN
#3/16"SHIMS
           2.0000    RUN
LAM TH"
            .5000    RUN
TH=1.3750"
---------
SH F=9.4737
ALL STRS=0.9711 KSI

DL STRS=0.4708 KSI
D+L STRS=0.7708 KSI
STRAIN=3.4549%

TH<2T ***
---------
BOND ***
---------
F=47.7262 K/PAD
 =238.6309 K/PIER
Fe-11=41.4378 K/PIER
Fe-+=19.1960 K/PIER

 NG
-----------------------
```

Try another design
too thin
∴ design is no good

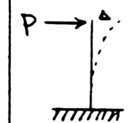
```
PAD L"
          18.0000    RUN
W"
          20.0000    RUN
#3/16"SHIMS
           4.0000    RUN
LAM TH"
            .7500    RUN
TH=3.5000"
---------
SH F=6.3158
ALL STRS=0.6474 KSI

DL STRS=0.4708 KSI
D+L STRS=0.7708 KSI
D+L>MAX ***
---------
STRAIN=3.9629%

F=17.3550 K/PAD
 =86.7749 K/PIER
Fe-11=15.0683 K/PIER
Fe-+=16.7952 K/PIER

 NG
-----------------------
PAD L"
```

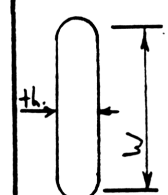
Try a third design
Load too high

Program NEO For Elastomeric Bearing Pad Design 712 bytes Page 1 of 4

Step	X	Y	Z	T	L
01♦LBL "NEO"					
02 CF 05					
03 16					
04 XROM "S"					
05 "HARD"	duro				
06 PROMPT					
07 STO 06					
08 .286					
09 /					
10 89.8					
11 -					
12 "LOW"	Low G				
13 XEQ 07					
14 STO 03	Low G				
15 RCL 06	duro				
16 4.5					
17 *					
18 115					
19 -					
20 "HI"					
21 XEQ 07	High G				1.9=cold stiffening constant
22 9.5	1.9×5				5 = DL friction coefficient: 14.2.9
23 *					
24 STO 10					
25 "DEL-T""					
26 PROMPT					
27 STO 08	Δ_T				One-way thermal movement
28 X=0?					
29 SF 05					
30 CF 06					
31 "DL"					
32 XEQ 03					
33 PROMPT					
34 STO 00	Min. load				
35 "D+L"					
36 XEQ 03					
37 PROMPT					
38 STO 01	Max. load				
39 FS? 05					fixed?
40 GTO D					Gto D
41 E					
42 ENTER↑	1	1			
43 "# PADS"					
44 PROMPT					
45 STO 14	#Pads	1			
46 X<=Y?					
47 GTO D					
48 "PIER HT""					
49 PROMPT					
50 STO 13	Height				
51 X=0?					
52 GTO D					

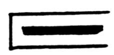
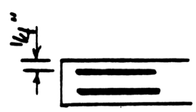
Step	X	Y	Z	T	L
53 SF 06					
54 22					
55 /	Ht/22				
56 3					
57 Y↑X	$(l/const)^3$				$\Delta = \frac{Pl^3}{3EI}$
58 STO 13					
59 "SKEW"					
60 XROM "Pd"					
61 STO 12	Skew				
62 CLX					
63 "# COLS"					
64 PROMPT					
65 X=0?					
66 GTO 00	#Col.				
67 "DIA""					
68 PROMPT					
69 X↑2					
70 X↑2	d^4	#col.			
71 *	Pier I				
72 .6	.6	Pier I			
73 GTO 13					
74♦LBL 07					
75 "⊢ G PSI"					
76 PROMPT	G psi				
77 E3	1000				
78 /					
79 RTN	G ksi				
80♦LBL 00					
81 "SH TH""					
82 PROMPT					
83 3					
84 Y↑X					
85 "W""					
86 PROMPT	W	th^3			
87♦LBL 13					
88 *	Pier I				
89 ST/ 13					$\frac{l^3}{3EI}$ in R.13
90♦LBL D					
91 2	2				
92 "SHIM TH 16'S"					
93 PROMPT	Sh.th.				
94 16					
95 /					
96 STO 16	th., inches				
97♦LBL 22					
98 XROM "LL"					
99 CF 07					
100 CF 09					
101 "PAD L""					
102 PROMPT					
103 STO 02	L				





D265

Program NEO For ______ 712 bytes

Step	X	Y	Z	T	L	Notes
104 "W="						
105 PROMPT	W					
106 RCL 02	L	W				
107 X>Y?						
108 X<>Y	lesser	greater				
109 STO 02						
110 X<>Y						
111 STO 11	greater					
112 CLX						
113 "#SHIMS"						
114 PROMPT	# of shims					
115 X=0?						
116 GTO 10						
117 STO 04	# shims					
118 E	1					
119 X=Y?						one shim → SF09
120 SF 09						
121 X<>Y	# shims					
122 RCL 16	sh. th.					
123 *	metal th					
124 ENTER↑	metal	metal				
125 "LAM TH="						
126 FC? 09						
127 PROMPT						
128 STO 05	Lam th.	metal				
129 RCL 04	# shims		•			
130 E	1			•		
131 -	# lams	lam. th.	•			
132 *	Σ lam	•				
133 .5	Z = 1/4"		•			
134 +	elast. th	•				
135 STO 09	elast. th.	metal th.				
136 +	total th					
137 "TH="						
138 ARCL X						
139 3						
140 *	3×th.				14.2.7	
141 STO 04						
142 "+="						
143 AVIEW						
144 XROM "L"						
145 4						
146 1/X	1/4					
147 FS? 09						
148 STO 05	cover layer					
149 E	1				β = 1	
150 STO 15	max				1000 psi max.	
151 GTO 11						
152♦LBL 10					Unreinf. pad	
153 "TH="	Th.					
154 PROMPT						
155 STO 05	Lam th.					
156 STO 09	elast. th					

Step	X	Y	Z	T	L	Notes
157 5						
158 *						
159 STO 04	5×total				14.2.7	
160 .8						
161 STO 15	.8				800 psi max	
162 E	1					
163 +	β				β = 1.8	
164♦LBL 11	β					
165 RCL 02	L					
166 RCL 11	W	L	β			
167 *	area					
168 LASTX	L					
169 RCL 02	W	L	area	β		
170 +	L+W					
171 ST+ X	Perim.	area	β			
172 /	area/P					
173 RCL 05	lam. th.					
174 /	Shape F	β				
175 X<> 11	L				Shape Factor in R.11	
176 RCL 02	W	L	β			
177 RCL 04	max. th.					
178 "TH"						
179 X>Y?					14.2.7	
180 XEQ 05						
181 RDN	W	L	β			
182 *	area	β				
183 STO 02	area					
184 RDN	β					
185 FS? 09					β = 1.0 for lam.	
186 1.4	β				= 1.4 for cover	
187 RCL 11	Sh. F				= 1.8 for plain	
188 "SH F="						
189 ARCL X						
190 FC? 00						
191 AVIEW						
192 X<>Y	β	S				
193 /						
194 RCL 03	Low G					
195 *	GS/β				Allow. compr. stress	
196 RCL 15	max allow				= GS/β	
197 X>Y?						
198 X<>Y	allow.				14.2.3	
199 1.1						
200 FC? 05					10% increase for fixe[redacted]	
201 SIGN						
202 *	allow.					
203 STO 15						
204 "ALL"						
205 XROM "K"						
206 XEQ "PY"						
207 ADV						
208 .5	500 psi					
209 RCL 00	DL					
210 RCL 02	area					
211 /	DL strs					

DZ66

Step	X	Y	Z	T	L	Notes
212 "DL"						
213 XROM "K"						
214 FC? 00						
215 XEQ "PY"						
216 "DL"						
217 X>Y?						
218 XEQ 05						Old AASHTO rules
219 "⊢<.2"						
220 .2	200psi	DL				
221 X>Y?						
222 XEQ 06						
223 RCL 01	Max DL+LL					
224 RCL 02	area					
225 /	Max. stress					
226 STO 07						
227 "D+L"						
228 XROM "K"						
229 XEQ "PY"						
230 "D+L"						
231 RCL 15	allow.	actual				
232 X<=Y?						
233 XEQ 05						
234 6	6					
235 RCL 11	Sh.F.					strain curve changes @ Sh.F ≐ 6
236 X<Y?						
237 GTO 09						
238 LN						
239 RCL 06	duro	ln(Sh.F)				
240 27.3						
241 /						
242 3.64						
243 -						
244 *						
245 RCL 06	duro					
246 7.16						
247 /						
248 -						
249 14.56						
250 +						
251 RCL 07	act. stress					
252 *						
253 .97						
254 +						
255 GTO 12	Strain%					
256♦LBL 09	Sh. factor					
257 18						
258 *						
259 214						
260 X<>Y						
261 -						
262 RCL 06	duro					
263 -						
264 .1						
265 *						
266 RCL 07	stress					
267 *	Strain%					

Step	X	Y	Z	T	L
268♦LBL 12					
269 "STRN="					
270 ARCL X					
271 "⊢%"					
272 AVIEW					Dupont rule limits strain to 7%
273 7	7%	Strain			
274 "STR>7%"					
275 X<=Y?					
276 XEQ 06					
277 FS? 05					
278 GTO 08					
279 ADV					
280 RCL 08	Δ_T				
281 ST+ X	$2\Delta_T$				
282 RCL 09	elast. th.				
283 "TH<2T"					14.2.6
284 X<=Y?					
285 XEQ 06					
286 RCL 00	DL				
287 RCL 10	$5 \times G_{ni}$				
288 RCL 02	area				
289 *					
290 RCL 08	Δ_T				$F = \frac{\Delta_T A G}{\text{elast.-th.}}$
291 *					
292 RCL 09	elast. th				
293 /	Temp. F	DL			
294 "BOND"					14.2.9
295 X>Y?					
296 XEQ 06					
297 5					
298 /	Temp. F				Force for one pad
299 "F="					
300 ARCL X					
301 XEQ 03					
302 XEQ "PY"					
303 FS? 06					FC? 06 or
304 FS? 00					FS? 00
305 GTO 08					
306 RCL 14	# Pads				
307 *	ΣF				Total force on pier if rigid
308 " ="					
309 XEQ 14	ΣForce				
310 RCL 08	Δ_T				
311 ENTER↑	Δ_T	Δ_T	F		see example:
312 "Fe-11="					
313 RCL 12	skew				
314 SIN	sin	Δ_T	Δ_T	F	
315 R↑	F				
316 *	$F_{e\text{-}11}$	Δ_T	Δ_T	Δ_T	
317 XEQ 14					
318 X<> L	F	Δ_T			
319 /	C_{neo}				
320 RCL 13	C_{pier}				
321 +	ΣC	Δ_T			

Program NEO For ______ bytes Page 4 of 4

	X	Y	Z	T	L		X	Y	Z	T	L
322 /	ΔT/℄C										
323 RCL 12	skew										
324 COS	cos										
325 *	Fe.⊢										
326 "Fe-⊢="											
327 XEQ 14											
328♦LBL 08											
329 ADV											
330 CLA											
331 XROM "OK"											
332 AVIEW											
333 GTO 22											
334♦LBL 03											
335 "⊢ K/PAD"											
336 RTN											
337♦LBL 14											
338 ARCL X	Force										
339 "⊢ K/PIER"											
340 XEQ "PV"											
341 RTN											
342♦LBL 05											
343 "⊢>MAX"											
344♦LBL 06											
345 "⊢ ***"											
346 AVIEW											
347 XROM "L"											
348 SF 07						SF 07 for any warning					
349 END											

D264

AASHTO T-PIER ANALYSIS

xeq PIER (SIZE 043)

SOFTKEYS (USER mode ON)

PIER uses no softkeys

USER FLAGS

PIER recognizes no user flags

INPUT SUMMARY

Prompt	Input	Default
T/C EL	Top of cap elevation	No default
CAP HT'	Cap height in feet	No default
FTG HT'	Footing height (thickness), feet	No default
B/F EL	Bottom of footing elevation	No default
DLW	Design low water elevation. Bypass (or enter zero) to signify dry-land construction.	Zero
DHW (if DLW prompt was answered)	Design high water elevation. Used to calculate ice elevation.	No default
GR EL (if DLW was bypassed)	Ground elevation, for landlocked piers	No default
SH TH'	Shaft thickness in feet	No default
SH L'	Shaft length measured along centerline of pier, in feet	No default
CAP TH'	Cap thickness in feet	No default
CAP L'	Cap length in feet	No default
WEDGE	Length of the triangular chamfer cut from the underside of the cap cantilever, in feet	No default
WEDGE HT'	Height of the cantilever chamfer which when subtracted from the CAP HT' leaves the height of the cap at the end of the cantilever	No default

Prompt	Input	Default
SPAN'	Span length in feet. For different spans about a pier, use the average of their lengths.	No default
DW O-O'	Deck width, out to out, in feet	No default
RW W'	Roadway width between curbs, for figuring live load lane eccentricities	No default
SKEW	Skew of the pier. A "square" bridge is defined to have 0^{o} skew.	No default
DECK EL	Elevation of the bridge deck	No default
BM+R HT'	Height from low beam to top of rail, for wind calculation, in feet	No default
BRG HT'	Bearing height; the difference in elevation between low beam and top of cap.	No default
ICE Fc PSI (if DLW prompt answered)	Ice crushing strength in psi	No default

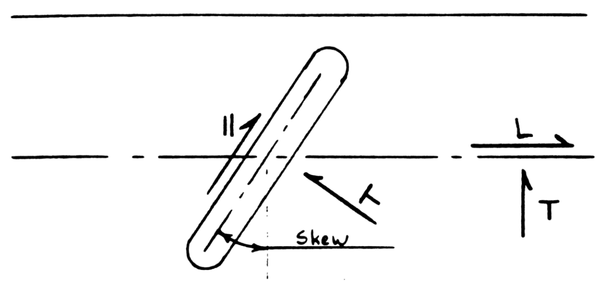
OUTPUT SUMMARY (Flag 00 has no effect on PIER)

MIN A-S	Minimum area of vertical reinforcing for the pier shaft (1/2 %) in square inches
SH DL	Shaft dead load in kips (assuming round ends)
CAP DL	Cap dead load in kips (assuming square ends)
SU DL EST	An estimate of the superstructure dead load
1L LL EST	An estimate of one lane of live load, without impact. Assumes continuous construction.
1L ARM 2L ARM	Eccentricity of one and two lanes of live load with respect to centerline of roadway (or CL pier) measured along the pier cap, in feet. Will not display 2-lane value if RW W' is less than 22'-0.
DLW ICE	Values which follow are for Design Low Water (or Ice) elevation. Press R/S to continue. If DLW prompt was bypassed (i.e. landlocked pier), no such identifier is displayed.
WOS	Wind on substructure. See next page for explanation of postscripts.

SH ARM	Position of center of action of load with respect to top of footing, for shaft design
FTG ARM	Position of center of action of load with respect to bottom of footing, for footing design; in feet.
ICE	Ice force along centerline of pier
WOF	Wind overturning force (uplift), in kips
WOF-11	Wind overturning moment, in foot-kips
W	Wind on superstructure
WL	Wind on live load

Postscripts (see below)

⊢	Perpendicular to the long axis of the pier
11	Parallel to the long axis of the pier
T	Transverse to centerline of roadway
L	Longitudinal to (along) centerline of roadway



PROGRAM FLAGS

Flag 07	clear set	Transverse force Longitudinal force
Flag 08	clear set	Landlocked pier (DLW prompt bypassed) Pier in water
Flag 09	clear set	WOS for DLW stage WOS for ICE stage, or if landlocked
Flag 10	clear set	WOS-⊢ (perpendicular) WOS-11 (parallel)

STORAGE REGISTER USE

Register	Value
00	Height of shaft
01	Top of cap elevation
02	DLW elevation or zero
03	ICE elevation or Ground elevation
04	Top of footing elevation
05	Footing thickness
06	Skew (degrees)
07	Shaft length
08	Cap height
09	Wedge height
10	Wedge length
11	Cap width (thickness)
12	Shaft thickness
13	Span
14	Deck width out to out
15	Cap length
16	Beam + rail (+ slab) height
17	WOS-11 (DLW Elev.)
18	WOS-11 Arm (DLW Elev.)
19	WOS-- (DLW Elev.)
20	WOS-- Arm (DLW Elev.)
21	WOS-11 (Ice or Ground Elev.)
22	WOS-11 Arm (Ice or Ground Elev.)
23	WOS-⊢ (Ice or Ground Elev.)
24	WOS-⊢ Arm (Ice or Ground Elev.)
25	Bearing height
26	Pier dead load (shaft + cap)
27	WOF, K
28	WOF-11, 'K
29	ICE force (11)
30	ICE arm

D272

Register	Value
31	Estimated Live load (one lane, no impact)
32	Estimated superstructure dead load
33	Height from top of cap to 6'-0 above top of deck (WL-T Arm)
34	W-T (11)
35	W-T (⊢)
36	W-T Arm
37	W-L (11)
38	W-L (⊢)
39	WL-T (11)
40	WL-T (⊢)
41	WL-L (11)
42	WL-L (⊢)

D273

D274

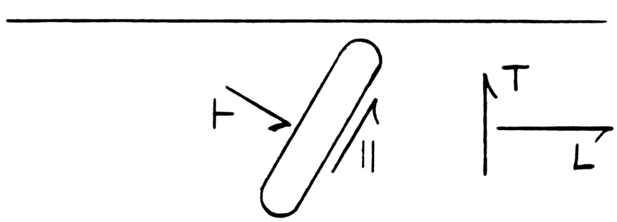
T-PIER ANALYSIS

Program PIER is a fast, simple way to calculate or check the AASHTO loads on a bridge pier. The program is written to analyze T-piers, but most of the output can also be used for column-bent piers by considering the columns as one shaft of equivalent width for purposes of wind load. In order to shorten the program, a few simplifying conditions were set:

1) The superstructure, pier, and footing are assumed to be symmetric with respect to the centerline of roadway.
2) The program does not recognize the presence of a sidewalk when calculating live load lane eccentricity, nor does it calculate sidewalk live load.
3) PIER does not calculate earth loads, loads due to earthquake or streamflow, or longitudinal forces due to live load.
4) Elastomeric bearing pads are assumed used. That is, the force due to temperature change is not the limit of the longitudinal force able to reach the pier; it is merely another additive force.

CONVENTION of DIRECTION of FORCES

This program, along with programs FTG and SHFTG (footing analysis programs) use a simple convention to differentiate directions of applied forces. Directions with respect to the superstructure or centerline of roadway are designated with a "T" for transverse forces and an "L" for longitudinal forces (forces acting along the centerline of roadway). Directions with respect to the centerline of the pier are designated with a "⊢" for perpendicular forces and a "11" for parallel forces (forces acting along the long axis of the pier). For bridges with no skew ("square" bridges), of course, $F_{\vdash} = F_L$ and $F_{11} = F_T$. For skewed bridges, the following diagram will provide the simplest description.



See the example for further clarification of the translation process between the two axes.

D275

POINT of APPLICATION of FORCES

Program PIER calculates the magnitude and point of action of the following forces, according to AASHTO Chapter 3:

W	Wind on superstructure
WOF	Wind overturning force
WOS	Wind on substructure
WL	Wind on live load
ICE	Ice force

The point of application of each component of these forces is given as a distance, or moment arm, both from bottom of footing (FTG ARM) for footing design; and from top of footing (SH ARM) for shaft design.

Wind on superstructure (W) is calculated by the simplified method of AASHTO 3.15.2.1.3; PIER does not check the requirement that the span be less than 125 feet. W and WL (wind on LL) are calculated as transverse and longitudinal forces with respect to centerline of roadway. Since a transverse force effectively is felt at its point of application while a-longitudinal force can act only through friction at the beam seat (due to the restraint of rotation by the beams longitudinally), different moment arms are given for each component. Then each of these forces is normalized (split) into components parallel (11) and perpendicular (⊢) to the long axis of the pier. See the example for clarification.

WOF (wind overturning force) is calculated and displayed as an uplifting force in kips and an overturning moment parallel to the long axis of the pier (WOF-11). There is no component perpendicular to the pier.

WOS (wind on substructure) is calculated as a force of 40 psf on each face of the pier. Those who would argue that the wind can't blow four directions at once will have to reduce these values with their sines and cosines. This program uses the simple reasoning that it's already blowing from two different directions on the superstructure; two more directions isn't that many more. WOS is actually calculated twice: once on the portion of the pier above DLW (design low water elevation), and once on the portion of the pier above the Ice elevation. PIER assumes that the bottom of the pier cap is above both these elevations; if this is not the case, the WOS values will be in error (they will be calculated with the full cap and a negative length of shaft exposed to the wind).

The ICE elevation is not input directly; it is assumed to be midway between DLW and DHW (design high water). Since the DHW elevation is not used for anything else, the ICE elevation can be set to any desired value by changing the DHW elevation. Although this may seem like a backward way of doing things, most hydrology information is presented in terms of DLW and DHW; only rarely is an ICE elevation given directly. The ice force is calculated as twelve inches of ice, of the crushing strength input, times the thickness of the pier shaft.

LANDLOCKED PIERS

For landlocked piers, bypass the DLW prompt and you will be prompted for GR EL, the ground elevation. WOS is calculated only once for this case, on the portion of the pier above ground. Again, the ground elevation must be lower than the top of the cap.

DEAD LOADS and ESTIMATES

Besides the AASHTO wind and ice loads, PIER also calculates the following quantities:

1) The pier dead load (shaft and cap separately), assuming a round-ended shaft, square-ended cap, and concrete at 150 pounds per cubic foot.
2) The minimum reinforcing required in the shaft (1/2% by AASHTO 4.4.9.4).
3) An estimate of the superstructure dead load, assuming a 7" thick deck, jersey rails, steel beams spaced at about 7'-0, and continuous construction. Naturally, all these assumptions place limits on the accuracy of such an estimate; still, experience with a given type of construction will lead to an expected deviation from this estimate, so it still can be useful as a check.
4) An estimate of reaction due to one lane of live load, assuming AASHTO lane loading, continuous construction, and no impact.
5) The effective moment arm (eccentricity) of one and two lanes of live load, measured along the pier cap from centerline of roadway. See the example for assumptions.

USE as SUBROUTINE

Although written as a stand-alone program, features are incorporated that allow PIER to be called as part of a user-written program; all of its output is stored for use by such a program. For example, a program to calculate and display AASHTO load groups could be written:

```
01   LBL GROUP
02   XROM "PIER" (called with XEQ)
03   "1 LANE LL"
04   PROMPT
     etc.
```

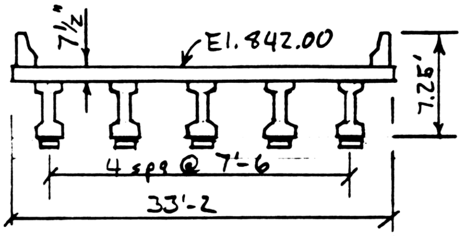
Such a program was not included in this module for two reasons: differing interpretations of several AASHTO

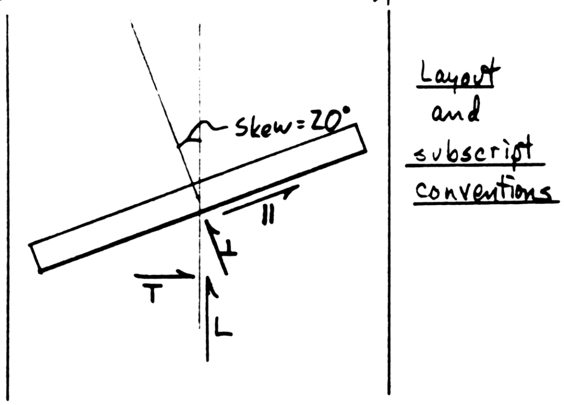
DZ77

provisions (for example, the application of WOS to a skewed pier was alluded to); and lack of memory space in the module (a working version took up about 750 bytes). Also important is the fact that output from PIER is essentially an intermediate step in the pier design procedure; an engineer has to digest it to get useful results, i.e. AASHTO load groups. If these results were calculated and output by a program written by a third party (this author), the results would be susceptible to use without checking or even understanding how they came about. As mentioned, many details of pier design by AASHTO are open to debate, and most design offices settle on certain methods; these would doubtless be different than the reasoning this author would employ.

Calculate loads for a T-Pier.

Span = 60'-0 Upstation and 40'-0 Downstation





Cap length: $4 \times 7.5'/\cos 20° = 31.93'$

USE 36'-0 CAP

Superst. DL:

Beams	5 × .45 k/' × 70'	= 157.50 k
Diaph	29' × 3.75' × 25' × .15	= 40.40
Int. Diaph	29' × .63' × 3' × .15	= 10.40
Deck	.625 × 33.17' × 70' × .15	= 217.70
FWS	.02 × 30' × 70'	= 42.0
Rails	2 × .37 k/' × 70'	= 51.4
		520.6 k

Est. Superst. LL

(.64 k/' × 70' × 1.25 continuity) + 26 k = 82.0 k/lane

Estimate cap depth by WSD:

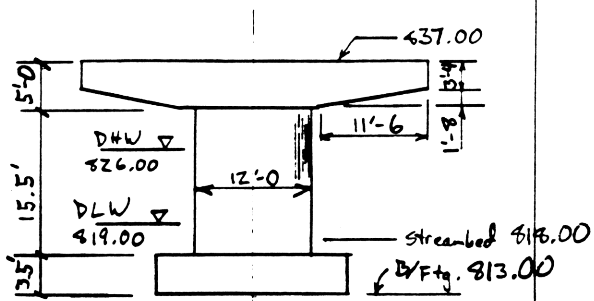
$\frac{2}{5}(520.6) + \frac{1}{2}(82.0) + 4'_{say} \times 3.33' \times 12' \times .15 = 273.24$ k

Allow. $V_c \doteq 150$ psi w/ stirrups

Use CAP TH = 3'-4

∴ Need $d \doteq \frac{273.24\text{ k}}{.15\text{ ksi} \times 3.33 \times 12} = 46"$

USE CAP HT. = 5'-0



Elevations

Beam	3.75'
Deck	.67'
Haunch	.17'
	4.58'

PG Elev.	842.00	
	4.58'	
Brg	.42'	
	837.00	T/Cap El.

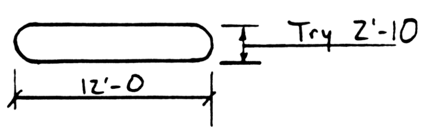
Streambed elev. ≈ 818.00

⇒ Set B/Ftg Elev. = 813.00

T/Cap	837.00	
Cap	5.0	
Ftg.: est.	3.5'	
B/Ftg	813.00	
	15.5'	= Shaft

Shaft

Cap = 36'-0 ⇒ USE 12'-0 SHAFT



(DZ79)

Cap DL

$(36' \times 5' - 11.5' \times 1.67') \times 3.33' \times .15 = \underline{\underline{40.42\ k}}$

Shaft DL

$(12' - 2.43') \times 2.43' = 25.97\ \square'$
$(2.43')^2 \times \pi/4 = 6.30$
32.27

$32.27\ \square' \times 15.5' \times .15 = \underline{\underline{75.04\ k}}$

Wind on Substructure

$WOS_\perp$:

	Area	d	A d
cap	160.0	2.5'	450
▷	−19.17	4.44'	−85.19
	160.83 ◻'		364.81

@ Design Low Water (DLW):

	Area	d	A d
	160.83		364.81
Shaft	156.00	11.5'	1794.00
	316.83 ◻'		2158.81

C.G. = 6.81' From T/Cap = 13.69' From T/Ftg. (DLW)

$WOS_\perp$ @ DLW = 316.83 ◻' × .040 = $\underline{\underline{12.67\ k}}$

@ Ice Elev. (= $\frac{DLW + DHW}{2}$ = 422.50)

	Area	d	A d
Cap	160.83		364.81
Shaft	114.00	9.75'	1111.50
	274.83		1476.31

C.G. @ 5.37' From T/Cap = 15.13' From T/Ftg (Ice)

$WOS_\perp$ @ Ice = 274.83 ◻' × .040 ksf = $\underline{\underline{10.99\ k}}$

$WOS_{\parallel}$ @ DLW

	Area	d	Ad
Cap	16.67 ◻'	2.5'	41.67
Shaft	36.83	11.5'	423.53
	53.50 ◻'		465.20

C.G. = 8.70' from T/Cap = 11.40' from T/Ftg.

$WOS_{\parallel}$ @ DLW = 53.50 ◻' × .040 = $\underline{\underline{2.14\ k}}$

$WOS_{\parallel}$ @ Ice Elev (= 422.50)

	Area	d	Ad
Cap	16.67		41.67
Shaft	26.92	9.75'	262.44
	43.59 ◻'		304.11

C.G. @ 6.98' from T/Cap = 13.52' from T/Ftg

$WOS_{\parallel}$ @ Ice = 43.59 ◻' × .04 = $\underline{\underline{1.74\ k}}$

Wind on Superstructure

Beam	3.75'
Haunch	.17'
Deck	.67'
Rail	2.67'
	7.25'

Area = $7.25' \times \frac{60' + 80'}{2}$ = 507.5'

W_T = 507.5 ◻' × .050 ksf = $\underline{\underline{25.38\ k}}$

$W_{T.\parallel}$ = 25.38 × cos20° = $\underline{\underline{23.84\ k}}$

$W_{T.\perp}$ = 〃 × sin20° = $\underline{\underline{8.68\ k}}$

Arm :

Shaft	15.5'
Cap	5.0'
Brg	.42'
Super. $\frac{7.25}{2}$ =	3.63'
	24.55' To T/Ftg

Wind on Super. (cont'd)

$W_L = 507.5 \square' \times .012 \text{ ksf} = \underline{6.09 \text{ k}}$

$W_{L\cdot\parallel} = 6.09 \times \sin 20° = \underline{2.04 \text{ k}}$

$W_{L\cdot\perp} = 6.09 \times \cos 20° = \underline{5.72 \text{ k}}$

Arm = 5.0' (cap) + 15.5' (sh) = **20.5' to T/Ftg.**

Wind on Live Load

$WLL_L = 70' \times .04 \text{ k/ft} = \underline{2.40 \text{ k}}$

$WLL_{L\cdot\perp} = 2.40 \times \cos 20° = \underline{2.63 \text{ k}}$

$WLL_{L\cdot\parallel} = 2.40 \times \sin 20° = \underline{.96 \text{ k}}$

Arm = **20.5' To T/Ftg**

$WLL_T = 70' \times .10 \text{ k/'} = \underline{7.00 \text{ k}}$

$WLL_{T\cdot\perp} = 7.0 \times \sin 20° = \underline{2.39 \text{ k}}$

$WLL_{T\cdot\parallel} = 7.0 \times \cos 20° = \underline{6.54 \text{ k}}$

Arm = 842.00 − 816.50 + 6.0'

= **31.50' to T/Ftg**

Wind Overturning Force

$WOF = 70' \times 33.17' \times .020 = \underline{46.43 \text{ k} \uparrow}$

$WOF_{\parallel} = 46.43 \text{ k} \times \frac{33.17'}{4} / \cos 20°$

$= \underline{409.72'^{\cdot k}}$

$WOF_{\perp} = 0.0$

Ice @ 300 psi (varies depending on state)

$Ice_{\parallel} = 2.83' \times 1' \times 144 \times .30 = \underline{122.26 \text{ k}}$

$Ice_{\perp} = .15 \times 122.26 = \underline{18.34 \text{ k}}$

Arm = 822.50 − 816.50 = **6.00' From T/Ftg**

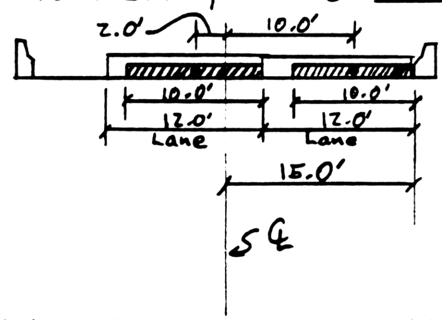
Thermal Forces

Obtain forces from Elastomeric Bearing Pad design.

Arm = 837.00 − 816.50 = **20.5'**

Live Load

From sh. 1, 1 lane = 82.0 k (est.)



1-Lane Arm = 10.0'/cos 20° = **10.64'**

2-Lane Arm = $\frac{10.0 - 2.0}{2} / \cos 20° =$ **4.26'**

Note: this is the method used by the program. Some states keep the second lane on its side of ℄. Thus the arm would be calculated:

2-Lane Arm = $\frac{10.0 - 5.0'}{2} / \cos 20° =$ **2.66'**

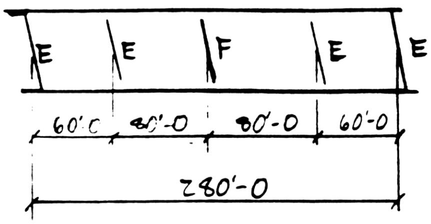
The program's method is the more conservative.

```
      XEQ "PIER"
 7:43:17 AM  09/27/88

TITLE:
PIERS 1 AND 3        RUN
************************
T/C EL                              Top of Cap
        837.0000     RUN
CAP HT'
          5.0000     RUN
FTG HT'                             Footing Thickness
          3.5000     RUN
B/F EL                              Bottom of Footing
        813.0000     RUN
DLW                                 Design Low Water El.
        819.0000     RUN
DHW                                 Design High Water El.
        826.0000     RUN
---------
SH TH'                              Shaft Thickness
          2.8333     RUN
SH L'                               Shaft Length
         12.0000     RUN
CAP TH'                             Cap Thickness
          3.3333     RUN
CAP L'                              Cap Length
         36.0000     RUN
WEDGE                               Triangular Wedge
         11.5000     RUN             Length
WEDGE HT'
          1.6667     RUN
---------
SPAN'                               Avg. span on pier
         70.0000     RUN
DW'O-O                              deck width out-out
         33.1667     RUN
RW W'                               Roadway width
         30.0000     RUN
SKEW
         20.0000     RUN
DECK EL
        842.0000     RUN
BM+R HT'                            Height of beam, rail,
          7.2500     RUN             and deck
BRG HT'                             Bearing Height
           .4167     RUN
ICE Fc PSI
        300.0000     RUN
------------------------
MIN A-S=23.2394 SQ"                 Min. shaft reinf.
SH DL=75.0439 K                     shaft DL
CAP DL=80.4157 K                    cap DL

SU DL EST=513.0727 K                } estimates
1L LL EST=82.0000 K                 }

1L ARM=10.6418'                     1 Lane } LL eccentricity
2L ARM=4.2567'                      2 Lane }
------------------------
```

```
DLW                                 Wind on Sub-
WOS-⊢=12.6733 K                     structure
 SH ARM=13.6863'                    @ Design Low Water
 FTG ARM=17.1863'

WOS-11=2.1400 K
 SH ARM=11.8037'
 FTG ARM=15.3037'
---------
ICE                                 WOS @ Ice
WOS-⊢=10.9933 K
 SH ARM=15.1283'
 FTG ARM=18.6283'

WOS-11=1.7433 K
 SH ARM=13.5225'
 FTG ARM=17.0225'
---------
ICE=122.3986 K                      Ice Force
 SH ARM=6.0000'
 FTG ARM=9.5000'
---------
WOF=46.4334 K                       Wind Overturning
 11 M=409.7196 'K
------------------------
W-T=25.3750 K                       Wind on
 F-11=23.8447 K                     Superstructure
 F-⊢=8.6788 K
 SH ARM=24.5417'
 FTG ARM=28.0417'

W-L=6.0900 K
 F-11=2.0829 K
 F-⊢=5.7227 K
 SH ARM=20.5000'
 FTG ARM=24.0000'
---------
WL-T=7.0000 K                       Wind on LL
 F-11=6.5778 K
 F-⊢=2.3941 K
 SH ARM=31.5000'
 FTG ARM=35.0000'

WL-L=2.8000 K
 F-11=0.9577 K
 F-⊢=2.6311 K
 SH ARM=20.5000'
 FTG ARM=24.0000'
************************
```

DZ82

Design Bearings for Piers 1,3



Skew 20° RHF
5 Beams @ 7'-6

$DL = \frac{520.6}{10 \text{ pads (avg.)}} = \underline{52.06\ ^{k}/_{pad}}$

Live Load

$\text{Dist.} = \frac{S}{5.5} = \frac{7.5}{5.5} = 1.3636\ ^{wh}/_{bm}$

$82.0\ ^{k}/_{lane} \cdot \frac{1 \text{ lane}}{2 \text{ wh}} \times 1.3636 \frac{wh}{bm} \times \frac{1 \text{ bm line}}{2 \text{ pads}} = \underline{27.95\ ^{k}/_{pad}}$

Uplift (from computer output or tables)

$4.70\ ^{k}/_{lane} \times \frac{1}{2} \times 1.3636 \times \frac{1}{2} = \underline{-1.60\ ^{k}/_{pad}}$

Temperature - One-way Δ_T

$(80' \times 12) \times (6.5 \times 10^{-6}) \times 45°F$ (Concrete beams)

$= .281''$

Pier height = (437.0-413.0)-3.5' = 20.5'

Prestr. bms → 10 brg. pads

Desire Length = 1'-4½ for prestr. beams

Use 55 durometer neoprene

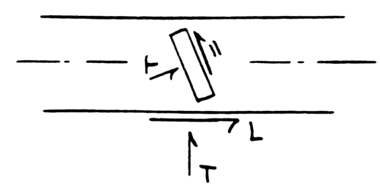
Try 10" × 16½" Pads w/ 4-½" lams

```
          XROM "NEO"
5:45:52 PM  09/18/88

TITLE:
PIERS 1 AND 3         RUN
*************************
HARD
          55.0000     RUN
DEL-T"
            .2810     RUN
DL K/PAD
          52.0600     RUN
MAX LL K/PAD
          27.9500     RUN
MIN LL K/PAD
          -1.6000     RUN
# PADS
          10.0000     RUN
PIER HT'
          20.5000     RUN
SKEW (DMS)
          20.0000     RUN
# COLS
                      RUN
SH TH'
           2.8333     RUN
W'
          11.0000     RUN
SHIM TH 16'S
           2.0000     RUN
------------------------
PAD L"
          16.5000     RUN
W"
          10.0000     RUN
#1/8"SHIMS
           5.0000     RUN
LAM TH"
            .5000     RUN
TH=3.1250"
----------
SH F=6.2264
ALL STRS=0.6382 KSI

DL STRS=0.3155 KSI
D+L STRS=0.4849 KSI
STRAIN=2.8641%

F=4.6736 K/PAD
 =46.7359 K/PIER
Fe-11=15.9846 K/PIER
Fe-+=28.5575 K/PIER

 OK
------------------------
PAD L"
```

D243

Calculations For FOOTING DESIGN

Skew = 20° RHF



	GROUP	P (k)	M_{ll} ('k)	M_t ('k)
	IA			
	Superst. DL	520.60	—	—
	Pier DL 75.04+80.42	155.46	—	—
	PierSh. Buoy 32.27'×2.5'×.0624	-5.03	—	—
	2 Lanes LL 2×82.0@4.26'	164.00	698.6	—
	Group IA @ 100%	835.03	698.6	—
2	**IB**			
	DL	671.03	—	—
	One Lane LL @10.64'	82.0	872.5	—
	Group IB @ 100%	753.03	872.5	—
3	**II**			
	DL	671.03	—	—
	WOS @ DLW	—	32.75	217.80
	WOF	-46.43	409.72	—
	W Longit. @ 24.0'	—	-50.00	137.34
	W Transv. @ 28.04'	—	668.64	243.37
	Group II @ 125%	624.60	1061.11	598.51
4	**III A**			
	Gr. IA	835.03	698.60	—
	.3 (W+WOF+WOS)	-13.93	318.33	179.55
	WLL Longit. @ 24.0'	—	-22.98	63.15
	WLL Transv. @ 35.0'	—	230.22	83.79
	Group III A @ 125%	821.10	1224.17	326.49
5	**III B**			
	Gr. IB	753.03	872.50	—
		-13.93	525.57	326.49
	Group III B @ 125%	739.10	1398.07	326.49

GROUP	P (k)	M_{ll} ('k)	M_t ('k)	
IV A				
Group IA	835.03	698.60	—	
Thermal (from NEO) @ 24'	—	383.63	685.38	
Group IV A @ 125%	835.03	1082.23	685.38	6
IV B				
Gr IB	753.03	872.50	—	
Thermal @ 24'	—	383.63	685.38	
Group IV B @ 125%	753.03	1256.13	685.38	7
V				
Group II	624.60	1061.11	598.51	
T	—	383.63	685.38	
Group V @ 140%	624.60	1444.74	1283.89	8
VI A				
Gr. 3A	821.10	1224.17	326.49	
T	—	383.63	685.38	
Group VI A @ 140%	821.10	1607.80	1011.87	9
VI B				
Gr. III B	739.10	1398.07	326.49	
T	—	383.63	685.38	
Group VI B @ 140%	739.10	1781.70	1011.87	10
VIII A				
Gr. IA	835.03	698.60	—	
Ice 122.40 × 9.5'	—	1162.80	174.42	
Group VIII A @ 140%	835.03	1861.40	174.42	11
VIII B				
Gr. IB	753.03	872.50	—	
	—	1162.80	174.42	
Group VIII B @ 140%	753.03	2035.30	174.42	12
IX Gr. II	624.60	1061.11	598.51	
Ice		1162.80	174.42	
Group IX @ 150%	624.60	2223.91	772.93	13

D284

Program PIER For AASHTO T-Pier Loads 407 bytes Page 1 of 4

Step	X	Y	Z	T	L
01♦LBL "PIER"					
02 42					
03 XROM "S"					
04 CF 09					
05 CF 10					
06 "T/C EL"					
07 PROMPT					
08 STO 01	T/C El				
09 "CAP"					
10 XEQ 08					
11 STO 08	Cap Wt.				
12 "FTG"					
13 XEQ 08	Ftg Th.				
14 STO 05					
15 "B/F EL"					
16 PROMPT					
17 RCL 05	Ftg Th.				
18 +					
19 STO 04	T/Ftg El				
20 CLX					
21 "DLW"					
22 PROMPT					
23 STO 02	DLW El.				
24 X=0?					
25 GTO 00					
26 CF 08					
27 "DHW"					
28 PROMPT	DHW				
29 RCL 02	DLW				
30 +					
31 2					
32 /	Ice El				
33 GTO 03					
34♦LBL 00					
35 "GR EL"					
36 PROMPT	Gr El.				
37 SF 08					
38 SF 09					
39♦LBL 03					
40 STO 03	Ice or Gr.				
41 XROM "L"					
42 "SH TH·"					
43 PROMPT					
44 STO 12	Sh. Th				
45 "SH L·"					
46 PROMPT					
47 STO 07	Sh. L				
48 "CAP TH·"					
49 PROMPT					
50 STO 11	Cap Th				
51 "CAP L·"					
52 PROMPT					
53 STO 15	Cap L				
54 "WEDGE"					
55 PROMPT					
56 STO 10	Wedge L				
57 XEQ 08					
58 STO 09	Wedge Wt				
59 XROM "L"					
60 "SPAN·"					
61 PROMPT					
62 STO 13	Span				
63 "DW'0-0"					
64 PROMPT					
65 STO 14	Deck W				
66 "RW W·"					
67 PROMPT					
68 STO 17	Roadway W				
69 "SKEW"					
70 PROMPT					
71 STO 06	Skew				
72 "DECK EL"					
73 PROMPT	Deck El				
74 RCL 04	T/Ftg El				
75 -					
76 6	6'				
77 +					
78 STO 33	Wt to LL				
79 "BM+R"					
80 XEQ 08					
81 STO 16	Bm-Rail Wt				
82 "BRG"					
83 XEQ 08					
84 STO 25	Brg Wt				
85 "ICE Fc PSI"					
86 FC? 08					
87 PROMPT					
88 STO 29	Ice Fc				
89 XROM "LL"					
90 RCL 07	Sh. L				
91 RCL 12	Sh. Th				
92 4.66					
93 /					
94 -					
95 RCL 12					
96 *	Sh. Area				
97 .72					
98 *	Min A_s				
99 "MIN "					
100 XROM "AS"					
101 AVIEW					
102 4.8					
103 /	Sh. Wt./ft.				
104 RCL 01	T/C EL				
105 RCL 04	T/Ftg El				
106 -					
107 STO 00	Shaft Arm				
108 RCL 08	Cap Wt.				
109 -					
110 STO 19	El. [illegible]				

$(\rho=.005)\cdot 144=.72$

$\frac{.15\ k/cf}{.72} = 1/4.8$

D245

Program PIER For ______ 407 bytes Page 2 of 4

Step	X	Y	Z	T	L
111 *	Sh. DL				
112 "SH DL"					
113 XEQ 10					
114 STO 26					
115 RCL 15	Cap L				
116 RCL 08	Cap Ht				
117 *					
118 RCL 09	Wedge Ht				
119 RCL 10	Wedge L				
120 *					
121 -	Cap Area				
122 RCL 11	Cap Th				
123 *					
124 .15					
125 *	Cap DL				
126 ST+ 26					Pier DL in R.26
127 "CAP DL"					
128 XEQ 10					
129 ADV					
130 RCL 13	Span				
131 75					
132 /					
133 .42					
134 -					
135 RCL 14					
136 *					
137 7					
138 /					
139 .87					
140 +					
141 RCL 14					
142 10					
143 /					
144 +					
145 RCL 13					
146 *					
147 RCL 14					
148 1.5					
149 *					
150 +					
151 STO 32					
152 "SU D"	Super DL Estimate				
153 XEQ 04					
154 RCL 13					
155 .8					.64k, ×1.25 = .8
156 *					
157 26	Lane Load/V				
158 +					
159 STO 31					
160 "1L L"					
161 XEQ 04	LL est.				
162 ADV					
163 RCL 17	Roadwidth				
164 2					
165 /					
166 5					
167 -					

Step	X	Y	Z	T	L
168 RCL 06	Skew				
169 COS					
170 /					
171 "1L"					
172 XEQ 13	Arm to 1	Lane LL			
173 6	Lane W/2				
174 LASTX	cos(Skew)				
175 /					
176 -					
177 "2L"	Arm to 2	Lanes LL			
178 X>0?					
179 XEQ 13					
180 XROM "LL"					
181 "DLW"					
182 FC? 08					
183 AVIEW					
184◆LBL 18					
185 RCL 02	DLW El				
186 FS? 09					
187 RCL 03	Ice El.				
188 RCL 01	T/Cap El.				
189 -					
190 CHS	Exposed Sh.				
191 RCL 08	Cap Ht.				
192 -	Exposed Sh. Ht.				
193 FS? 10					
194 RCL 12					
195 FC? 10					
196 RCL 07	Sh. Len Th.				
197 FC? 10					
198 RCL 15					
199 FS? 10					
200 RCL 11	Cap Len Th				
201 RCL 08	Cap Ht.	Cap Len Th	Sh Len Th	Sh. Ht.	
202 CLA					
203 XROM "AY"	Area	C.G. from top			
204 STO [					
205 *	A$\bar{y}$				
206 RCL 08	Cap Ht.				
207 RCL 09	Wedge Ht.				
208 3	C.G. pos				
209 /					
210 -					
211 RCL 09	Wedge Ht.				
212 RCL 10	Wedge L				
213 FS? 10					
214 CLX					
215 *	Wedge A				
216 ST- [					
217 *	Wedge A$\bar{y}$	A$\bar{y}$			
218 -	ΣA$\bar{y}$				
219 RCL [	ΣA				
220 /	$\bar{y}$				
221 LASTX	ΣA				
222 25					
223 /	.04·A	$\bar{y}$ from top			WOS = 40 psf

D286

Program PIER For ______ 407 bytes Page 3 of 4

Step	X	Y	Z	T	L	Note
224 RCL 00	Sh. Ht.					
225 RCL Z	$\bar{Y}$	Sh. Ht.	WOS	$\bar{Y}$		
226 -	WOS Arm					
227 "WOS-"						
228 FS? 10						
229 GTO 02						
230 "⊢⊢"						
231 STO 24	$Arm_{\perp}$					
232 X<>Y						
233 STO 23	$WOS_{\perp}$					
234 GTO 07						
235♦LBL 02						
236 "⊢11"						
237 STO 22	$Arm_{\parallel}$					
238 X<>Y						
239 STO 21	$WOS_{\parallel}$					
240♦LBL 07						
241 XEQ 10	WOS					
242 X<>Y						
243 XEQ 11	Arm					
244 FS?C 10						
245 GTO 14						
246 SF 10						
247 ADV						
248 GTO 18						
249♦LBL 14						
250 XROM "L"						
251 FS? 08						
252 GTO 06						
253 "ICE"						
254 FS?C 09						
255 GTO 05						
256 STO 18	$Arm_{\parallel}$					
257 X<>Y						
258 STO 17	$WOS_{\parallel}$					
259 RCL 23						
260 STO 19	$WOS_{\perp}$					
261 RCL 24						
262 STO 20	$Arm_{\perp}$					
263 SF 09						
264 AVIEW						
265 GTO 18						
266♦LBL 05						
267 RCL 29	Ice f'_c					
268 .144	[illegible]					
269 *						
270 RCL 12	Sh. Th.					
271 *						
272 STO 29	Ice Force, H					
273 XEQ 10						
274 RCL 03	Ice El.					
275 RCL 04	T/Ftg					
276 -						
277 STO 30	Ice Arm					
278 XEQ 11						
279 XROM "L"						
280♦LBL 06						
281 RCL 14	Deck W					
282 3						
283 *						
284 RCL 13	Span					
285 RCL 14	Deck W					
286 *	Area					
287 50						WOF = 20 psf
288 /						
289 STO 27	WOF A	Width/3				
290 "WOF"						
291 XEQ 10						
292 *						
293 RCL 06	Skew					
294 COS						
295 /						
296 " 11"						
297 XROM "M="						
298 STO 28						"M=" divides by 1
299 AVIEW						
300 XROM "LL"						
301 CF 07						
302 RCL 13	Span					
303 RCL 16	Bm Ht.					
304 *	Area					
305 20						Wind = 50 psf
306 /						
307 "W-T"						
308 XEQ 09						
309 STO 35	$W\text{-}T_{\perp}$					
310 X<>Y						
311 STO 34	$W\text{-}T_{\parallel}$					
312 RCL 00	Shaft Ht.					
313 RCL 16	Bm Ht.					
314 2						
315 /	Bm C.G.					
316 +						
317 RCL 25	Brg Ht.					
318 +	W-T Arm					
319 STO 36						
320 XEQ 11						
321 SF 07						
322 RCL 13						
323 RCL 16						
324 *	Area					
325 .012						Wind-L = 12 psf
326 *						
327 ADV						
328 "W-L"						
329 XEQ 09						
330 STO 38	$W\text{-}L_{\perp}$					

D287

Program PIER For ____________ 407 bytes

Step	X	Y	Z	T	L
331 X<>Y					
332 STO 37	W-L∥				
333 RCL 00	Sh.-Ht.				
334 XEQ 11					
335 CF 07					
336 RCL 13	Span				
337 .1					WL = 100 lb/ft.
338 *					
339 XROM "L"					
340 "WL-T"					
341 XEQ 09					
342 STO 40	WL-T⊥				
343 X<>Y					
344 STO 39	WL-T∥				
345 RCL 33	Arm				
346 XEQ 11					
347 SF 07					
348 RCL 13	Span				
349 25					
350 /					WL-L = 40 p/ft.
351 ADV					
352 "WL-L"					
353 XEQ 09					
354 STO 42	WL-L⊥				
355 X<>Y					
356 STO 41	WL-L∥				
357 RCL 00	Sh. Ht.				
358 XEQ 11					
359 XROM "L*"					
360 RTN					
361♦LBL 08					
362 "⊢ HT'"					
363 PROMPT					
364 RTN					
365♦LBL 04					
366 "⊢L EST"					
367 GTO 10					
368♦LBL 11					
369 " SH"	Sh.-Arm				
370 XEQ 13					
371 RCL 05	Ftg.-Th.				
372 +					
373 " FTG"	Ftg Arm				
374♦LBL 13					
375 "⊢ ARM="					
376 ARCL X					
377 XEQ "FT"					
378 AVIEW					
379 RTN					
380♦LBL 09	force				
381 XEQ 10					
382 RCL 06	Skew				
383 X<>Y	force	skew			
384 P-R	f∥	f⊥			
385 FS? 07					
386 X<>Y					
387 " F-11"					
388 XEQ 10					
389 X<>Y	f⊥	f∥			
390 " F-⊥"					
391♦LBL 10					
392 "⊢="					
393 ARCL X					
394 "⊢ K"					
395 AVIEW					
396 END					

D266

FOOTING ANALYSIS PROGRAMS

xeq FTG for iterative footing analysis (SIZE 4N+17 for N groups)

SOFTKEYS (USER mode ON)

C	Correct a load group input incorrectly
F	New footing configuration

USER FLAGS

Flag 00	clear	Full output, including MAX_{11} and $MAX_{\vdash}$
	set	Skips the MAX for each side

INPUT SUMMARY

Prompt	Input	Default
#GRPS	Number of load groups to be input and stored	One
P-DL	Vertical force due to dead load. Bypass to input Group 1 directly and skip the load breakdown.	No default
M-11	Moment parallel to the long axis of the pier	Zero
M-⊢	Moment perpendicular to the long axis of the pier	Zero
P-LL	Vertical force due to live load	No default
P-2, etc.	Vertical force (parallel moment, perpendicular moment) in Group 2	No default
DIV 1.25,1.4	The load group divisor (from column 14 of AASHTO Table 3.22.1A)	1.0
LIMIT	Allowable pile load or bearing pressure above which the resulting pressure calculated should be displayed. If bypassed (or zero is input), all results are displayed. If a value greater than zero is input, only displays MAX's greater than LIMIT and MIN values less than zero.	Zero
# PILES	Number of piles. Bypass for a spread footing.	Zero

Prompt	Input	Default
S-11 (piles)	Section modulus of a pile footing parallel to the long axis of the pier, in feet	No default
S-⊢ (piles)	Section modulus of a pile footing perpendicular to the long axis of the pier, in feet	No default
L' (11) (spread)	Length of the footing in the direction parallel to the long axis of the pier, in feet	No default
W' (⊢) (spread)	Width of the footing in the direction perpendicular to the long axis of the pier	No default
FTG DL, K	Footing deadload in kips. Should include the weight of earth on the footing and the effect of buoyancy on the footing, if any.	No default

OUTPUT SUMMARY (Items in brackets { } are displayed only if flag 00 is clear)

1 MAX	The maximum pressure or pile load due to Group 1, in kips per square foot or kips respectively
{(11)MAX}	The average maximum pressure on the long side of the footing
{(⊢)MAX}	The average maximum pressure on the short side of the footing
1 MIN	The minimum pressure or pile load due to Group 1; If less than zero, appends UPL to denote uplift.
GR _ GOV:	Displays the group producing the largest MAX
DL LL OT Σ	The dead load, live load, and overturning components and total load of the governing group, in <u>tons</u> per square foot or tons. This is the full unDIVided load.

PROGRAM FLAGS

Flag 05	clear	LIMIT of zero (all groups displayed)
	set	Only display MAX's greater than LIMIT and MIN's less than zero
Flag 06	clear	Inputting DL and LL
	set	DL and LL have been input

DZ90

Flag 08	clear	FTG
	set	SHFTG
Flag 09	clear	Spread footing
	set	Pile footing
Flag 10	clear	Have DL and LL--do breakdown
	set	No DL or LL--no breakdown

<u>STORAGE REGISTER USE</u>

Register	Value
00	LIMIT
01	# PILES or area
02	$S_{\vdash}$
03	S_{11}
04	FTG DL, K
05	Counter
06	Governing MAX
07	Governing Group
08	(11) MAX
09	(⊢) MAX
10	Governing MAX, unDIVided
11	DL P
12	DL M-11
13	DL M-⊢
14	LL P
15	LL M-11
16	LL M-⊢
17	Group 1 P
18	Group 1 M-11
19	Group 1 M-⊢
20	Group 1 L.F. (1.0)
21	Group 2 P
22	Group 2 M-11
23	Group 2 M-⊢
24	Group 2 L.F.
25	Group 3 P
	etc.

D291

FOOTING ANALYSIS PROGRAM

Program FTG is a general-purpose pier and column footing analysis program, an extended version of SHFTG. It analyzes both spread footings on rock and pile footings. It accepts any number of load groups, then prompts for and analyzes different footing configurations until a satisfactory solution is found. This is the biggest difference between FTG and SHFTG, which accepts one footing configuration and then analyzes different load groups on that footing.

FTG first asks for the number of load groups to be input, in order to check the SIZE (storage register) requirement. It then begins prompting for load groups. A load group consists of a vertical load, moments in two planes, and a "load factor" by which these are divided. See the example. The vertical load should not include the weight of the footing itself (FTG DL), since this is input with each trial footing's properties. This way, if the footing size changes from the original estimate the results are still correct.

LOAD GROUPS

FTG assumes that Load Group 1 consists of only dead load and live load. The vertical load and moments due to each of these are input separately and combined by the program to comprise its Group 1; it then prompts for Group 2 and so on. Group 1 is thus not input directly. The DL and LL are also stored separately so that a breakdown may be made of the governing load group into DL, LL, and the remainder, OT (overturning, i.e. wind and other lateral forces). After the program has analyzed a given footing configuration for all the load groups, it displays the governing group and calculates this breakdown for that group; these loads have not been divided by the DIV factor input. This is made clear in the example. If this breakdown is not wanted, bypass the dead load prompt (P-DL). The program will then prompt for Group 1 in its entirety, then Group 2 and so on as before. All this is also made more clear in the examples. Note that the program assumes the governing group includes the Live Load (axial and moments) input at the beginning of the run. If this is not true, the load breakdown at the end will be incorrect.

If there is only one load group to be analyzed, the P-DL prompt may be bypassed and the group entered directly, or (if it consists of just DL and LL) the component loads can be input, and the program will give the breakdown at the end.

D292

CORRECTING INPUT ERRORS

If a mistake is made when inputting load groups, press softkey C to correct it. This must be done before finishing inputting the next group. If C is pressed during the moment or DIV input, the program returns to the beginning prompt (P, K) for that load group; if before or during input of the vertical load, it returns to the previous group's prompt. Softkey C may be pressed more than once to get back to a previous load group. However, any load information input after that group will have to be re-input. Softkey C will not return to the P-DL or P-LL prompt from the P-2 prompt sequence; it will prompt for P-1, etc., not correcting the DL or LL values input.

If a mistake is made inputting footing properties, press softkey F. Softkey F can be used at any time; it will interrupt an analysis and jump immediately to the footing input section.

"LIMIT:" LONG and SHORT OUTPUT

FTG has two basic modes of operation, controlled by the user's input to the LIMIT prompt. If a "limit" of zero is input (the default value), the program calculates and displays the results of all the load groups upon the given footing, as explained in the next several paragraphs. If, on the other hand, the designer wants to check out several different footing configurations quickly, this full output becomes rather time-consuming. If, for instance, the maximum allowable pile load is 80 kips, input a LIMIT of, say, 75. Then the program will only display the load groups which cause a maximum pile load over 75 kips or a minimum pile load of less than zero (uplift). Thus, a number of footing trials can be run quickly. This also works, of course, for spread footings; the LIMIT is then in kips per square foot.

Assuming the LIMIT prompt is bypassed (or zero input), FTG calculates and displays the maximum and minimum corner or pile load for each load group; if the minimum is less than zero (i.e. an uplift exists) this is noted by UPL displayed after the pressure value. The program assumes that the footing is effective in uplift. That is, its calculations are based on the full footing properties whether or not uplift exists. In reality, of course, if a spread footing undergoes uplift, a neutral axis shift occurs and the maximum corner pressure increases. FTG does not calculate this increase.

If flag 00 is clear, the program also displays the average maximum load along each side as defined by

D293

AASHTO 4.4.6.1 and ACI 15.4.1 for reinforcing design. See the examples. As mentioned, it then displays the governing load group, and, if the dead load and live load forces and moments were input separately, the breakdown of forces for that group.

FLAG 00

If flag 00 is set, display of the side maximums (11)MAX and (⊢)MAX is skipped. This is the only function of flag 00; most of the output control is handled by the LIMIT feature. FTG is thus one of the few programs in this package whose normal mode of operation will probably be with flag 00 set.

SPREAD and PILE FOOTINGS

Spread footings and pile footings are handled differently in the input sequence. For a pile footing, the designer must calculate the section (moment) properties in each direction, and then input the number of piles (# PILES) and these properties. For a spread footing, the # PILES prompt is bypassed (press R/S without keying anything in). The program then prompts for the footing length and width, and figures the section properties itself. For either of these, the designer needs to calculate and input the FTG DL, the selfweight of the footing being analyzed (unless it was included in all the load groups--this takes a lot of confidence). An extra calculator is handy for this, as well as pile property calculations. However, these calculations can all be done on the HP-41 after the prompt if a few extra numbers on the printout aren't objectionable. For instance, for a 9'x12' footing 3'-0 thick, after the FTG DL prompt, key in:

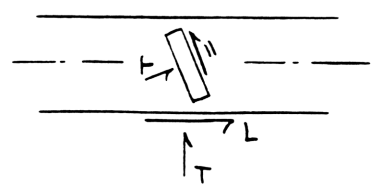
```
12
Enter↑
9
*
3
*
.15
*
R/S
```

If not using a printer, of course, this is no problem at all.

The calculations for pile group properties (S-11 and S-⊢) can also be done during a run, as no information is being carried in the stack. However, it is often more convenient to use another calculator or work out a few trials beforehand. Although this sounds awkward, in practice it works quite well.

Calculations For FOOTING DESIGN

Skew = 20° RHF



GROUP	P (k)	M_{ll} (′k)	M_t (′k)	
IA				
Superst. DL	520.60	—	—	
Pier DL 75.04·80.42	155.46	—	—	
PierSh. Buoy 32.27′·2.5′·.0624	-5.03	—	—	
2 Lanes LL 2×82.0426′	164.00	698.6	—	
Group IA @ 100%	435.03	698.6	—	
IB				2
DL	671.03	—	—	
One Lane LL @10.64′	82.0	872.5	—	
Group IB @ 100%	753.03	872.5	—	
II				3
DL	671.03	—	—	
WOS @ DLW	—	32.75	217.80	
WOF	-46.43	409.72	—	
W Longit. @ 24.0′	—	-50.00	137.34	
W Transv. @ 24.04′	—	668.64	243.37	
Group II @ 125%	624.60	1061.11	598.51	
III A				4
Gr. IA	435.03	698.60	—	
.3(W+WOF+WOS)	-13.93	318.33	179.55	
WLL Longit. @ 24.0′	—	-22.98	63.15	
WLL Transv. @ 35.0′	—	230.22	83.79	
Group III A @ 125%	821.10	1224.17	326.49	
III B				5
Gr. IB	753.03	872.50	—	
	-13.93	525.57	326.49	
Group III B @ 125%	739.10	1398.07	326.49	

GROUP	P (k)	M_{ll} (′k)	M_t (′k)	
IV A				
Group IA	435.03	698.60	—	
Thermal (from NEO) @24′	—	383.63	685.38	
Group IV A @ 125%	435.03	1082.23	685.38	6
IV B				
Gr IB	753.03	872.50	—	
Thermal @ 24′	—	383.63	685.38	
Group IV B @ 125%	753.03	1256.13	685.38	7
V				
Group II	624.60	1061.11	598.51	
T	—	383.63	685.38	
Group V @ 140%	624.60	1444.74	1283.89	8
VI A				
Gr. 3A	821.10	1224.17	326.49	
T	—	383.63	685.38	
Group VI A @ 140%	821.10	1607.80	1011.87	9
VI B				
Gr. III B	739.10	1398.07	326.49	
T	—	383.63	685.38	
Group VI B @ 140%	739.10	1781.70	1011.87	10
VIII A				
Gr. IA	435.03	698.60	—	
Ice 122.40×9.5′	—	1162.80	174.42	
Group VIII A @ 140%	435.03	1861.40	174.42	11
VIII B				
Gr. IB	753.03	872.50	—	
	—	1162.80	174.42	
Group VIII B @ 140%	753.03	2035.30	174.42	12
IX Gr. II	624.60	1061.11	598.51	
Ice		1162.80	174.42	
Group IX @ 150%	624.60	2223.91	772.93	13

DZ95

Design a footing for the T-Pier loads given. (See documentation for program PIER.)

Check Groups

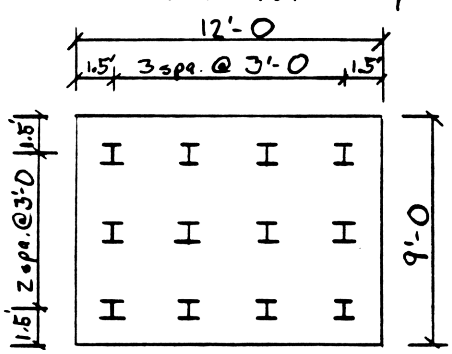
	IA	(1)
	II	(2)
	IIIA	(3)
	VIA	(4)
and	VIB	(5)

Use HP 12×53 piling @ 7.0 ksi:

15.5 □" × 7 = 108.5, say <u>110 kips</u>

Group I → $\frac{835\ k}{110}$ = 7.6 → 8 piles

<u>Estimate 12 piles:</u> try minimum footing configuration. Estimate 3.0' thick. <u>Neglect</u> <u>earth</u> on ftg. Consider under full buoyancy.



Footing DL = 12'×9'×3'× (.15-.0624)

= <u>28.34 k</u>

$S_{\parallel} = (6\times1.5^2 + 6\times4.5^2)/4.5' = 30.0'$

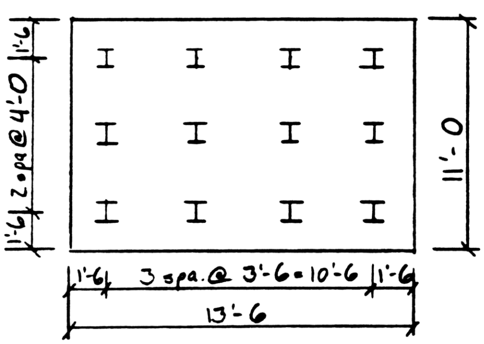
$S_{\perp} = 8 \times 3' = 24.0'$

```
                  XEQ "FTG"
# GRPS
             5.0000    RUN
 8:22:40 AM  03/21/89

TITLE:
T-PIER  20d SKEW     RUN
************************
P-DL K
           671.0300    RUN
 M-11 'K
             0.0000    RUN
 M-+ 'K
             0.0000    RUN
---------
P-LL K
           164.0000    RUN
 M-11 'K
           698.6000    RUN
 M-+ 'K
                       RUN
---------
P-2 K
           624.6000    RUN
 M-11 'K
         1,061.1100    RUN
 M-+ 'K
           598.5100    RUN
DIV 1.25,1.4
             1.2500    RUN
---------
P-3 K
           821.1000    RUN
 M-11 'K
         1,224.1700    RUN
 M-+ 'K
           326.4900    RUN
DIV 1.25,1.4
             1.2500    RUN
---------
P-4 K
           821.1000    RUN
 M-11 'K
         1,607.8000    RUN
 M-+ 'K
         1,011.8700    RUN
DIV 1.25,1.4
             1.4000    RUN
---------
P-5 K
           739.1000    RUN
 M-11 'K
         1,781.7000    RUN
 M-+ 'K
         1,011.8700    RUN
DIV 1.25,1.4
             1.4000    RUN
************************
LIMIT
```

Run preliminary trial with LIMIT of 100 k (slightly below allowable of 110 kips) to check for overload, uplift

```
*************************
LIMIT
          100.0000    RUN
# PILES
           12.0000    RUN
S-11
           30.0000    RUN
S-⊥
           24.0000    RUN
FTG DL, K
           28.3800    RUN
---------
2 MIN=-5.8933 K: UPL
3 MAX=100.1595 K
4 MAX=118.9604 K
4 MIN=-17.8318 K: UPL
5 MAX=118.2199 K
5 MIN=-26.8533 K: UPL
-----------------------
GR 4 GOV:
*************************
```

Both maximum downward load and uplift are too high - increase footing size and re-run. Since these were close, expect next trial to work ⇒ set LIMIT = 0.



Greater moment arm ⇒ make 3'-6 thick

Ftg DL = $13.5' \times 11' \times 3.5' \times (.15 - .0624) = 45.53k$

$S_{11} = 6 \times (1.75^2 + 5.25^2)/5.25' = 35.0'$

$S_{\perp} = 4 \times (4')^2/4' = 32.0'$

```
LIMIT
           0.0000    RUN
# PILES
          12.0000    RUN
S-11
          35.0000    RUN
S-⊥
          32.0000    RUN
FTG DL, K
          45.5300    RUN
---------
1 MAX=93.3400 K
 (11) MAX=93.3400 K
 (⊥) MAX=73.3800 K
1 MIN=53.4200 K

2 MAX=83.8920 K
 (11) MAX=68.9293 K
 (⊥) MAX=59.6381 K
2 MIN=5.4586 K

3 MAX=93.9186 K
 (11) MAX=85.7564 K
 (⊥) MAX=65.9376 K
3 MIN=21.6321 K

4 MAX=106.9837 K
 (11) MAX=84.3974 K
 (⊥) MAX=74.1715 K
4 MIN=-3.8135 K: UPL

5 MAX=105.6518 K
 (11) MAX=83.0654 K
 (⊥) MAX=69.2906 K
5 MIN=-12.2434 K: UPL
-----------------------
GR 4 GOV:
DL=29.8567 T
LL=16.8133 T
OT=28.2186 T
---------
Σ=74.8886 T
*************************
```

Check Group 4:

DL: $\dfrac{671.03 + 45.53}{12} = \underline{59.71\ k}$

LL: $\dfrac{164.00}{12} + \dfrac{698.6'^{k}}{35.0'} = \underline{33.63\ k}$

Σ: $\dfrac{421.1 + 45.53}{12} + \dfrac{1607.40'^{k}}{35.0'} + \dfrac{1011.87'^{k}}{32.0'}$

$= \underline{149.78\ k}, \div 1.40 = \underline{\underline{106.98\ k}}$

OT = difference = $149.78 - 59.71 - 33.63$
$= \underline{56.44\ k}$

Check the same footing as a spread footing.

$L = 13.5'$; $W = 11.0'$

```
LIMIT
                    SF 00
          0.0000    RUN
# PILES
                    RUN
L'(11)
         13.5000    RUN
W'(⊦)
         11.0000    RUN
FTG DL, K
         45.5300    RUN
---------
1 MAX=8.0205 KSF
1 MIN=3.8389 KSF

2 MAX=7.9095 KSF
2 MIN=-0.6892 KSF: UPL

3 MAX=8.5591 KSF
3 MIN=0.7783 KSF

4 MAX=10.2604 KSF
4 MIN=-1.9234 KSF: UPL

5 MAX=10.2377 KSF
5 MIN=-2.6896 KSF: UPL
------------------------
GR 4 GOV:
DL=2.4127 TSF
LL=1.5976 TSF
OT=3.1720 TSF
---------
Σ=7.1823 TSF
*************************
LIMIT
```

Set flag 00 to eliminate "side" loads

Again check group 4:

$$Area = 13.5' \times 11' = 148.50 \square'$$

$$S_{11} = 13.5^2 \times 11/6 = 334.125 \ ft^3$$

$$S_{\vdash} = 11^2 \times 13.5/6 = 272.25 \ ft^3$$

$$\Sigma: \frac{821.10 + 45.53}{148.50 \square'} = 5.836 \ ksf$$

$$\frac{1607.80'^{k}}{334.125} = 4.812$$

$$\frac{1011.87'^{k}}{272.25} = 3.717$$

$$Max\ \Sigma = 14.365 \ ksf$$
$$\div 1.40$$
$$10.26 \ ksf$$

$$Min\ \Sigma = 5.836 - (4.812 + 3.717)/1.40$$
$$= -2.69 \ ksf/1.40$$
$$= -1.92 \ ksf \ \text{(UPLIFT)}$$

(Note that for a spread footing, a corner in uplift renders the above analysis invalid. Most designers would accept only a small amount of uplift. This footing should thus be upsized)

Had flag 00 been clear, the average side loads would be:

$$(11)\ Max = (5.836 + 4.812)/1.40$$
$$= 7.61 \ ksf$$

$$(\vdash)\ Max = (5.836 + 3.717)/1.40$$
$$= 6.82 \ ksf$$

These are used for the methods described in ACI 15.4.1 and AASHTO 4.4.6.1 if these are used by the designer.

D298

FOOTING ANALYSIS PROGRAMS

xeq SHFTG for simple footing analysis (SIZE 004)

SOFTKEYS

SHFTG uses no softkeys. However, the softkeys C and F are active in FTG when SHFTG is in use, and should be avoided.

USER FLAGS

Flag 00	clear	Full output
	set	Title block not printed; no other effect

INPUT SUMMARY

Prompt	Input	Default
# PILES	Number of piles. Bypass for a spread footing.	Zero
S-11 (piles)	Section modulus of a pile footing parallel to the long axis of the pier, in feet	No default
S-⊢ (piles)	Section modulus of a pile footing perpendicular to the long axis of the pier, in feet	No default
L' (11) (spread)	Length of the footing in the direction parallel to the long axis of the pier, in feet	No default
W' (⊢) (spread)	Width of the footing in the direction perpendicular to the long axis of the pier	No default
P	Vertical force in kips	No default
M-11	Moment parallel to the long axis of the pier	Zero
M-⊢	Moment perpendicular to the long axis of the pier	M-11

OUTPUT SUMMARY

MAX	The maximum pressure or pile load, in kips per square foot or kips respectively
MIN	The minimum pressure or pile load; if less than zero, appends UPL to denote uplift.

PROGRAM FLAGS

Flag 08	clear	FTG
	set	SHFTG
Flag 09	clear	Spread footing
	set	Pile footing

STORAGE REGISTER USE

Register	Value
00	not used
01	# PILES or area
02	$S_{\vdash}$
03	S_{11}

COMPARISON BETWEEN PROGRAMS

Feature	SHFTG	FTG
FTG DL	Must be included in loads	Input separately
Divisor	Must be included in loads	Input separately
Loads	Not stored	Stored
Best use	Checking/analysis	Design
SIZE	004	4N + 17

SHORT FOOTING ANALYSIS PROGRAM

Program SHFTG is a simple pier and column footing analysis program, an much-simplified version of FTG. It analyzes both spread footings on rock and pile footings. It prompts for a footing configuration, then analyzes any number of load groups; these load groups, however, are not stored and must be re-input for each footing trial. This is the main difference between SHFTG and FTG, which accepts all the load groups up front, and then analyzes different footing configurations on all the groups input. SHFTG is thus best used for checking a given footing configuration, while program FTG is better suited for iterating to the best configuration. See the preceding page for a comparison.

INPUT

SHFTG begins by asking for the footing configuration; the input sequence is the same as for program FTG. It then begins prompting for load groups. A load group consists of a vertical load and moments in two planes. See the example. The vertical load must include the weight of the footing; SHFTG does not allow input of the footing deadload separately as FTG does. It must also have been divided by the "load factor" (1.0, 1.25, 1.40, or 1.50) as SHFTG does not do this itself. The moments should both be positive. If one is negative, pressures or pile loads will be at non-extreme corners of the footing; if both are negative, MAX and MIN will be switched.

MAXIMUMS and MINIMUMS

SHFTG calculates and displays the maximum and minimum corner or pile load for each load group; if the minimum is less than zero (i.e. an uplift exists) this is noted by UPL displayed after the pressure value. The program assumes that the footing is effective in uplift. That is, its calculations are based on the full footing properties whether or not uplift exists. In reality, of course, if a spread footing undergoes uplift, a neutral axis shift occurs and the maximum corner pressure increases. SHFTG does not calculate this increase.

FOOTING TYPES

Spread footings and pile footings are input differently. For a pile footing, the designer must calculate the section (moment) properties in each direction, and then input the number of piles and these properties; an extra calculator is handy for this. The

D301

calculations can be done on the HP-41 during the run; no data is being carried in the stack. See the program listing and stack analysis for clarification.

For a spread footing, on the other hand, the # PILES prompt is bypassed (press R/S without keying anything in). The program then prompts for the footing length and width, and figures the section properties itself.

FOOTING SHAPE

A spread footing must be rectangular in order that its section properties will be calculated correctly by the program. A pile footing, on the other hand, has all its properties input by the user. Thus, if a non-rectangular spread footing is to be analyzed, its properties must be calculated by hand and input as for a pile footing. All output will be in Kips and Tons, instead of KSF and TSF as for a normal spread footing.

Program FTG For Footing Analysis 724 bytes Page 1 of 4

Step	X	Y	Z	T	L
01♦LBL "FTG"					
02 CF 08					
03 E	1				
04 "# GRPS"					
05 PROMPT	# Groups				
06 4					
07 +					
08 4					
09 *	20,24,etc.				
10 ENTER↑	Reg.	Reg.			
11 XROM "S"	Reg.				
12 E3	1000				
13 /					
14 11					
15 +	11.00R				
16 STO 05	counter				
17 SF 06					
18 "DL"					
19 ASTO X					
20 CF 22					
21 XEQ 12					
22 XROM "L"					
23 "LL"					
24 ASTO X					
25 XEQ 12					
26 CF 10					
27 RCL 14	LL P				
28 RCL 11	DL P				
29 +					
30 STO 17	Gr.I P				
31 RCL 15	LL Mu				
32 RCL 12	DL Mu				
33 +					
34 STO 18	Gr.I Mu				
35 RCL 16	LL Mr				
36 RCL 13	DL Mr				
37 +					
38 STO 19	Gr.I Mr				
39 SIGN	1				
40 STO 20	Gr.I Div				
41 3					
42 ST+ 05					
43 ISG 05					
44 GTO 06					
45 GTO F					
46♦LBL C					
47 RCL 05	counter				
48 FRC	.00 Reg				
49 LASTX	counter	Frc.			
50 17.5					
51 -					
52 LASTX	17.5				
53 X<>Y	Count-17.5	17.5	Frc		
54 X<0?					
55 GTO 07					
56 4					
57 /					
58 INT					
59 4					
60 *	New count	17.5	Frc		
61 +					
62 INT	New Count	Frc			
63♦LBL 07	Counter				
64 +					
65 STO 05					
66 GTO 06					
67♦LBL 01					
68 SF 10					
69 6					
70 ST+ 05	Counter				
71♦LBL 06					
72 CF 06					
73 RCL 05	counter				
74 4					
75 /					
76 3					
77 -	Group#				
78 XROM "L"					
79♦LBL 12	Group#				
80 "P-"					
81 XROM "AX"					
82 "⊢ K"					
83 PROMPT	P				
84 FC? 22					
85 GTO 01					
86 STO IND 05					
87 ISG 05					
88 CLX					
89 XEQ 05	Mu				
90 STO IND 05					
91 ISG 05					
92 CLX					
93 PROMPT	Mr				
94 STO IND 05					
95 ISG 05					
96 FS? 06					
97 RTN					
98 SIGN	1				
99 "DIV 1.25,1.4"					
100 PROMPT	Divisor				
101 STO IND 05					
102 ISG 05					
103 GTO 06					
104♦LBL F					
105 XROM "L*"					
106 CF 05					
107 CLX					
108 STO 06	Max				

Program FTG For Footing Analysis ______ bytes

Step	X	Y	Z	T	L
109 "LIMIT"					
110 PROMPT					
111 STO 00	Limit				
112 X>0?					
113 SF 05					
114♦LBL 08					
115 CF 09					
116 CLX					
117 "# PILES"					
118 PROMPT					
119 X=0?					
120 GTO 09					
121 STO 01	#Piles				
122 SF 09					
123 "S-11"					
124 PROMPT					
125 STO 03	S_{11}				
126 "S-⊢"					
127 PROMPT					
128 STO 02	$S_{\vdash}$				
129 GTO 02					
130♦LBL 09					
131 "L'(11)"					
132 PROMPT	Length				
133 ENTER↑					
134 ENTER↑	L	L	L		
135 "W'(⊢)"					
136 PROMPT	W	L	L		
137 *	Area				
138 STO 01					
139 LASTX	W				
140 *	W^2L				
141 X<>Y	L				
142 RCL 01	Area				
143 *	WL^2				
144 6					
145 /	$WL^2/6$				
146 STO 03	S_{11}				
147 X<>Y	W^2L				
148 LASTX	6				
149 /					
150 STO 02	$S_{\vdash}$				
151♦LBL 02					
152 FS? 08					
153 RTN					
154 "FTG DL, K"					
155 PROMPT					
156 STO 04	Ftg DL				
157 XROM "L"					
158 RCL 05					
159 FRC	.00 Reg				
160 17					
161 +					
162 STO 05	Counter				
163 GTO 02					

D304

Step	X	Y	Z	T	L	Notes
164♦LBL 23						
165 FC? 05						
166 ADV						
167♦LBL 02						
168 RCL IND 05	P					
169 RCL 04	Ftg DL					
170 +	ΣP					
171 RCL 01	#Piles	or Area				
172 /						
173 STO 08	P/A					
174 STO 09	P/A					
175 ISG 05						
176 RCL IND 05	M_{11}					
177 RCL 03	S_{11}					
178 /	M/S_{11}					
179 ST+ 08	(Max 11)					
180 ISG 05						
181 RCL IND 05	$M_{\vdash}$					
182 RCL 02	$S_{\vdash}$	$M_{\vdash}$	M/S_{11}	P/A		
183 /	$M/S_{\vdash}$	M/S_{11}	P/A			
184 ST+ 09	(Max ⊢)					
185 +	ΣM/S	P/A	P/A			
186 ST- Z			Min			
187 +	Max	Min				
188 RCL 05	counter					
189 4	4	counter	Max	Min		
190 /						
191 4						
192 -						
193 CLA	Group#	Max	Min			
194 XROM "AX"	"Group#"					
195 RDN	Max	Min				
196 ISG 05						
197 RCL IND 05	Divisor	Max	Min			
198 ST/ 08						
199 ST/ 09						
200 ST/ Z			Min			
201 /	Max	Min				
202 RCL 00	Limit					
203 X>Y?						x>y? and FS?05 } gto 10
204 FC? 05						
205 X<0?						
206 GTO 10						
207 ASTO X	"Group#"	Max	Min			
208 X<>Y	Max	"#"				
209 XEQ 14						
210 RCL 06	MAX	Max	"Gr#"	Min		MAX is largest of Max's
211 X>Y?						
212 GTO 02						
213 RDN	Max	"Gr#"	Min			Max is new MAX
214 STO 06						
215 ST* L						Div.×(Max/Div) in LastX
216 RDN	"Gr#"	Min		Max		
217 STO 07	Asto'd					
218 LASTX	Max (undivided)		min			
219 STO 10						
220 R↑			"Gr#"	Min		

Program FTG For Footing Analysis ______ bytes Page 3 of 4

Step	X	Y	Z	T	L
221♦LBL 02					
222 FC? 05					
223 FS? 00					
224 GTO 02					
225 " (11)"					
226 X<> 08	Max $_{11}$				
227 XEQ 14					
228 " (⊢)"					
229 X<> 09	Max $_{\vdash}$				
230 XEQ 14					
231♦LBL 02			"Gr#"	Min	
232 CLA					
233 ARCL Z			"Gr#"		
234♦LBL 10					
235 R↑	Min				
236 X>0?					X < 0? or FC? 05 } xeq 03
237 FC? 05					
238 XEQ 03					
239♦LBL 11					
240 ISG 05					
241 GTO 23					
242 RCL 05	counter				All groups done - do breakdown of governing Group
243 FRC					
244 17					
245 +					
246 STO 05	counter				
247 XROM "LL"					
248 RCL 10	unDIVided Max				
249 RCL 04	Ftg DL				
250 RCL 11	DL P				
251 +	ΣDL P	Max			
252 RCL 01	Area				
253 /	DL P/A				
254 RCL 12	DL M_{11}				
255 RCL 03	S_{11}	M_{11}	DL P/A	Max	
256 /	DL M/S_{11}	P/A	Max		
257 +	Σ DL	Max			
258 RCL 13	DL $M_{\vdash}$				
259 RCL 02	$S_{\vdash}$	$M_{\vdash}$	ΣDL	Max	
260 /					
261 +	Σ DL				
262 ST- 10					
263 "GR "					
264 ARCL 07					
265 "⊢ GOV:"					
266 AVIEW					
267 FC? 10					
268 FS? 05					
269 GTO F					
270 "DL"					
271 XEQ 13	Σ DL				
272 RDN	Max				
273 RCL 14	LL P				
274 RCL 01	Area				
275 /	LL P/A				
276 RCL 15	LL M_{11}				
277 RCL 03	S_{11}	M_{11}	P/A	Max	
278 /	M/S	P/A	Max		
279 +	ΣLL				
280 RCL 16	LL $M_{\vdash}$				
281 RCL 02	$S_{\vdash}$	$M_{\vdash}$	ΣLL	Max	
282 /					
283 +	Σ LL	Max			
284 ST- 10					
285 "LL"					
286 XEQ 13	ΣLL				
287 RCL 10	Remainder				Remainder = Overturning
288 RND	OT	ΣLL	Max	Max	
289 "OT"					
290 XEQ 13					
291 XROM "L"					
292 "Σ"					
293 R↑	Max (UnDIVided)				
294 XEQ 13					
295 GTO F					
296♦LBL 05					
297 " M-11 'K"					
298 PROMPT	M_{11}				
299 " M-⊢ 'K"					
300 RTN					
301♦LBL "SHFTG"					
302 SF 08					
303 3	Reg.				
304 XROM "S"					
305 XEQ 08					Lbl 08 prompts Ftg properties
306 XROM "L"					
307♦LBL 04					
308 "P, K"					
309 PROMPT	P				
310 RCL 01	Area				
311 /	P/A				
312 XEQ 05	M_{11}				
313 RCL 03	S_{11}				
314 /	M/S_{11}	P/A			
315 PROMPT	$M_{\vdash}$				
316 RCL 02	$S_{\vdash}$	$M_{\vdash}$	M/S_{11}	P/A	
317 /	M/$S_{\vdash}$				
318 +	ΣM/S	P/A			
319 +	Max				
320 CLA					
321 XEQ 14					
322 LASTX	ΣM/S	Max			
323 ST+ X					
324 -	Min				
325 CLA					
326 XEQ 03					
327 XROM "LL"					
328 GTO 04					

D305

Step	X	Y	Z	T	L
329◆LBL 13	Load, k				
330 2					
331 /	Load, Tons				
332 "⊢="					
333 ARCL X					
334 "⊢ T"					
335 GTO 07					
336◆LBL 03					
337 "⊢ MIN="					
338 GTO 02					
339◆LBL 14					
340 "⊢ MAX="					
341◆LBL 02					
342 ARCL X					
343◆LBL 00					
344 "⊢ K"					
345◆LBL 07					
346 FC? 09					
347 "⊢SF"					
348 X<0?	Load				
349 "⊢: UPL"					
350 XEQ "PV"					
351 RTN					
352◆LBL "YZ"	x	Y	Z	T	
353 RDN	Y	Z	T	x	
354 X<>Y	Z	Y	T	x	
355 R↑					
356 RTN	x	Z	Y	T	
357◆LBL "YT"	x	Y	Z	T	
358 RDN	Y	Z	T	x	
359 X<> Z	T	Z	Y	x	
360 R↑					
361 RTN	X	T	Z	Y	
362◆LBL "ZT"	x	Y	Z	T	
363 R↑	T	x	Y	Z	
364 X<> T	Z	x	Y	T	
365 RDN					
366 RTN	X	Y	T	Z	
367◆LBL "RP"	X	Y			
368 R-P	d	θ'			
369 STO a	d				x
370 X<> L	X	θ'			d
371 X<>Y	θ'	X			
372 FS? 03					
373 HMS	θ	x			
374 X<>Y	x	θ			
375 X<> L	d or θ'	θ			X
376 CLX	0				
377 X<> a	d	θ			X
378 RTN					
379◆LBL "PR"	d	θ'			
380 X<>Y	θ'	d			
381 FS? 03					
382 HR	θ	d			
383 X<>Y	d	θ			
384 P-R	X	Y			d
385 END					

D306