



# **SUPERLEX**

FOR THE HP 71 COMPUTER

**TITAN SOFTWARE COMPANY**

## NOTICE

Titan Software Company makes no express or implied warranty with regard to the keystroke procedures and program material offered or their merchantability or their fitness for any particular purpose. The keystroke procedures and program material are made available solely on an "as is" basis, and the entire risk as to their quality and performance is with the user. Should the keystroke procedures or the program material prove defective, the user (and not Titan Software Company or any other party) shall bear the entire cost of all necessary correction and all incidental or consequential damages. Titan Software Company shall not be liable for any incidental or consequential damages in connection with or arising out of the furnishing, use, or performance of the keystroke procedures or program material.

## INTRODUCTION

The Superlex File is a set of BASIC keywords that will solve a variety of conversion problems. It also contains keywords that you can use to control the AC and ALARM annunciators, quick contrast and display scroll control, lowercase string conversion, a running clock keyword and more.

When the lex file is loaded into your HP-71 Computer these keywords become available for programs or keyboard use.

## Installing the Convert Lex File

To use the additional keywords supplied by this file it must be resident in your HP-71 Computer. Load it as you would any other file as per the Owner's Manual instructions.

Some general things to remember about lex files are:

The lex file must be in the machine to utilize its keywords.

If the file is removed and a program containing keywords from it is run, an error will result when the line containing the keyword is encountered.

With the file not in place, a program line containing keywords can be viewed (or edited) but the keyword will not appear. Instead you will see a line similar to that below.

```
110 DISP XFN820xx(A) (xx is the particular keyword number)
```

When the lex file is back in the computer the program will run normally. For more information on Lex files consult the HP-71 Owner's Manual, sections 8 and 9.

## How To Use The Keywords

This manual assumes that you are familiar with the operation of your HP-71 Computer. It also assumes you know how to create, edit, store, and run programs.

Many of the keywords in this lex file follow the same syntax. This manual will not show a syntax diagram for each word. Instead the keywords that are the same will be illustrated by one syntax diagram and then some examples of various ways to use them.

The following page contains a complete list of all the keywords and their function.

|          |  |
|----------|--|
| ACR2SQM  | acres to square miles                        |
| ATM2PSI  | atmospheres to pounds per square inch        |
| CAC      | clear AC annunciator                         |
| CALM     | clear ALARM annunciator                      |
| CC2CI    | cubic centimeters to cubic inches            |
| CF2CM    | cubic feet to cubic meters                   |
| CI2CC    | cubic inches to cubic centimeters            |
| CLOCK    | display running clock                        |
| CM2CF    | cubic meters to cubic feet                   |
| CM2IN    | centimeters to inches                        |
| CN2FH    | centigrade to fahrenheit                     |
| DATE2JUL | calendar date to julian date                 |
| DIFDAYS  | number of days between dates                 |
| DOFW     | calendar date to day of week                 |
| FH2CN    | fahrenheit to centigrade                     |
| FT2M     | feet to meters                               |
| FT2NM    | feet to nautical miles                       |
| FT2SM    | feet to statute miles                        |
| GAL2LTR  | U.S. liquid gallons to liters                |
| HI       | high contrast                                |
| HMS      | decimal hours to hours, minutes, and seconds |
| HR       | hours, minutes, and seconds to decimal hours |
| IN2CM    | inches to centimeters                        |
| JUL2DATE | julian date to calendar date                 |
| KG2LBM   | kilograms to pound mass, avoirdupois         |
| KM2LY    | kilometers to lightyears                     |
| KM2SM    | kilometers to statute miles                  |
| LBM2KG   | pound mass, avoirdupois to kilograms         |
| LO       | low contrast                                 |
| LTR2GAL  | liters to U.S. liquid gallons                |
| LWRC\$   | uppercase to lowercase                       |
| LY2KM    | lightyears to kilometers                     |
| ME       | medium contrast                              |
| M2FT     | meters to feet                               |
| ND       | no display delay and scroll                  |
| NM2FT    | nautical miles to feet                       |
| NM2SM    | nautical miles to statute miles              |
| PSI2ATM  | pounds per square inch to atmospheres        |
| REV\$    | reverse a string                             |
| SAC      | set AC annunciator                           |
| SALM     | set ALARM annunciator                        |
| SD       | standard (default) display delay and scroll  |
| SM2FT    | statute miles to feet                        |
| SM2KM    | statute miles to kilometers                  |
| SM2NM    | statute miles to nautical miles              |
| SQM2ACR  | square miles to acres                        |
| WKEY\$   | wait for a key                               |

The syntax of all the conversion keywords in this file is the same for e  
One example syntax diagram will be used for all of the these keywords.

Several of the keywords require special format for input. They will be c

In the following example we will use the keyword SM2KM (statute miles to  
kilometers). All the conversion keywords can be used in the same manner.

Syntax:

SM2KM(X)

Where X is any Real numeric expression.

Examples:

```
DISP SM2KM(Y/15)      A=SM2KM(SQR(H))
```

```
IF SM2KM(W*P+4) > 15 THEN GOTO 120
```

```
A$=STR$(SM2KM(S))
```

We can also nest one conversion factor inside another to obtain a  
new conversion. For example, to convert nautical miles to  
kilometers:

```
SM2KM(NM2SM(X))
```

Where X is the numeric expression representing nautical miles.

## HMS

The keyword HMS converts decimal hours into hours, minutes, and seconds. The input can be any Real numeric expression representing decimal hours. For example, 3.9 hours. To convert this to hours, minutes and seconds type (all examples assume STD display format in effect):

```
HMS(3.9) <endline>
```

The result being 3.54. The format of the result is hh.mmss where hh is the hour value, mm is the time in minutes, and ss is the time in seconds.

Let's try a couple more examples:

```
HMS(3.99) <endline> gives 3.5924 which reads:
```

3 hours, 59 minutes, and 24 seconds.

```
HMS(18.52417) <endline> gives 18.3127012 which reads:
```

18 hours, 31 minutes, 27 seconds, and 12 thousandths of a second.

Any digits following the seconds will be fractions of a second.



## HR

The keyword HR converts time in hours, minutes, and seconds into decimal hours. The input can be any Real numeric expression representing time in the format hh.mmss where hh is the time in hours, mm the time in minutes, and ss the time in seconds. For example 3 hours and 54 minutes; to convert this to decimal hours type (all examples assume STD display format):

```
HR(3.54) <endline>
```

The result is 3.9.

To convert 3 hours, 59 minutes, and 24 seconds to decimal hours type:

```
HR(3.5924) <endline> gives 3.99
```

Any digits entered after ss will be treated as fractions of a second. Convert 18 hours, 31 minutes, 27 seconds, and 12 thousands of a second.

```
HR(18.3127012) <endline> gives 18.52417.
```

## ADDITIONAL EXAMPLES

Here are some more examples of ways to use the conversion keywords. Note that the keywords can also be used in the CALC mode as well as in programs or the keyboard. All of the keywords are classified as functions and are IF.....THEN.....ELSE legal.

```
PRINT CCZCI(X/R)
```

```
PRINT "LIGHTYEARS=";KMZLY(C)
```

```
Z=MZFT(SIN(Q/E))
```

```
IF A THEN LTRZGAL(X1) ELSE LTRZGAL(X2)
```

```
PRINT HMS(TIME/60/60)
```

```
Z=KMZLY(SM2KM(NM2SM(D)))
```

```
W=CI2CC(P)
```

```
S=LY2KM(VAL(C$))
```

```
D=NM2SM(X)
```

The following keywords control the display annunciators:

**SAC** set AC annunciator, also sets flag -57

**CAC** clears AC annunciator, also clears flag -57

**SALM** set ALARM annunciator, also sets flag -60

**CALM** clear ALARM annunciator, also clears flag -60

The above keywords can be used in programs or from the keyboard. They will not function in CALC mode.

Example:

```
IF S>Y+A THEN SALM
```

The next group controls the display:

**HI** sets display contrast to High

**ME** sets display contrast to Medium

**LO** sets display contrast to Low

**SD** sets display delay and scroll to default

**ND** sets display delay and scroll to 0

The above can be used in programs or from the keyboard but not in CALC mode.

The following keywords perform calendar functions:

#### DATE2JUL(X)

Given a calendar date X this function returns the julian number for that date. The date must be  $\geq$  OCT 15.5, 1582 and is entered in the form:

YYYY.MMDD where YYYY is the year, MM is the month, DD is the day. fractions of a day may be used as another digit following DD.

Examples:

DATE2JUL(1985.0120) gives 2446085.5 as the julian date.

DATE2JUL(1846.10234) DATE2JUL(D)

-----

#### JUL2DATE(X)

Given a julian date X this function returns the calendar date in the form YYYY.MMDD. The julian date must be  $\geq$ 2299161. Its usage is similar to that of DATE2JUL.

Both of these keywords will generate errors if illegal dates are used (that is before the start of the Gregorian Calendar). Both can be used in CALC mode.

#### DIFDAYS(X,Y)

Given two calendar dates X and Y this function returns the number of days between them. Both dates must be  $\geq$ Oct 15.5, 1582. They must be in the usual form YYYY.MMDD.

#### DOFW(X)

Given a calendar date this function returns the day of the week as an integer.

0-Sunday, 1-Monday, etc. The date must be greater than Oct 15.5, 1582. Also for accurate results use OPTION ROUND POS. Rounding errors may occur with other settings.

The following keywords are string functions.

**LWRC\$** converts uppercase letters to lowercase letters

The above keyword is similar to **UPRC\$** (see reference manual) it only acts on letters. Other characters will remain unchanged.

Examples:

**LWRC\$('HELLO 2 YOU')** returns 'hello 2 you'

**LWRC\$(A\$) LWRC\$(X\$[1,3])**

**REV\$** reverses a string

Example:

**REV\$('HELLO')** returns 'OLLEH'

The above keywords can be used in programs or from the keyboard but will not function in **CALC** mode.

## WKEY\$

This keyword when encountered in a program will halt it and wait for a key to be pressed. It will then return the key and resume program execution. It returns a string character.

Example:

```
DISP "Input Yes or No" @ K$=WKEY$
```

In the above example a 'Y' or 'N' could be entered as a response and then K\$ could be tested for which character was entered. Notice also that an INPUT statement is not needed because WKEY\$ will halt the program and preserve the display.

## CLOCK

This keyword displays a running clock. Pressing any key will terminate the clock display. It can only be executed from the keyboard and is not programmable.

If you have any questions regarding this software please write:

Titan Software Company  
4721 Ambrose Ave.  
Los Angeles, CA 90027