HP-75 I/O ROM
Programming Techniques Manual
Addendum

This document covers some problems introduced by the HP-75 I/O ROM. Work arounds are provided where appropriate.

Conditions That Can Cause a Reset

If a currently running program contains a DELETE#, INSERT#, or REPLACE# statement with a *file number* of zero, the program will modify itself. *This will lead to unpredictable results, usually a reset and loss of all data in system memory.* The problem also occurs if you specify the *file number* of the current program (previously assigned with ASSIGN#) in the DELETE#, INSERT#, or REPLACE# statement.

Certain situations involving the CALL statement can cause a reset of the HP-75, resulting in the loss of all data in system memory:

1. When you CALL an allocated program that expects parameters, you must provide the number and type of parameters that the program expects. Failure to provide the parameters, or providing incorrect parameters, will cause a reset.

2. If you use CALL with parameters to a program which does not have a SUB statement as the first line in the program, a reset will result.

3. If you call a non-BASIC file (for example, a TEXT file) from a mass-storage device (HP-IL or XMEM), a reset will result. Examples of such statements are: CALL "PRGM:M1" and CALL "PRGM:XMEM".

To avoid a reset and loss of data, avoid the above conditions.

Using SENDIO and ENTIO$ With Commands Containing HP-IL Addresses

The SENDIO and ENTIO$ statements do not properly process HP-IL commands that contain an HP-IL address number. These include: AADnn, AEPnn, AESnn, AMPnn, DDLnn, DDTnn, PPEnn, SADnn, LADnn, and TADnn. There are two aspects to this problem: 1) none of the commands will accept nn = 31; they all report "Bad Parameter," and 2) all of the commands perform a MOD 10 on nn; thus, the effective range of nn is 0 through 9. You can avoid this problem by using the SEND statement. The equivalent SEND forms of these commands follow:

Command:	SEND Form:
AADnn	RDY 128+nn
AEPnn	RDY 160+nn
AESnn	RDY 192+nn
AMPnn	RDY 224+nn
LADnn	LISTEN nn
TADnn	TALK nn
SADnn	SAD nn
PPEnn	CMD 128+nn
DDLnn	DDL nn
DDTnn	DDT nn

There is another work around for the LADnn and TADnn statements. Use the LAD# or TAD# forms, with the device name in the first parameter, instead of LADnn or TADnn.

Notes on Supporting Functions

The following notes relate to the supporting functions described in appendixes B and D of the *HP-75 I/O ROM Programming Techniques Manual*.

1. If AUTOLOOP is ON and the clock is in EXACT mode, HP-IL devices in standby mode may unexpectedly be powered up. To avoid this, don't use EXACT mode, or don't leave the devices in standby mode, when AUTOLOOP is ON.

2. If you change the definition of [SHIFT] [CTL] [CLR] using DEF KEY, the I/O ROM will cause the new definition to be *added* to the definitions of [TAB], [SHIFT] [TAB], and [CTL] [CLR]. For example, if you have redefined [SHIFT] [CTL] [CLR] to execute CLEAR LOOP, pressing [TAB] will move the cursor to the next tab position, then clear the loop. To avoid problems, don't redefine the [SHIFT] [CTL] [CLR] key sequence.

3. The INSERT# and REPLACE# statements do not correctly insert numeric data at positions greater than the ninth element in the data line. To avoid this problem, convert the numeric data to a string (using STR$) before inserting or replacing it. Use VAL to convert the data back to a number after reading it.

4. Do not use string constants in the conditional part of an IF statement. Such a statement, for example: A$="a" @ IF A$="a" THEN FOR X=1 TO 2 @ NEXT X will generate an error.

5. The LASTLN? function does not accept a *filename* with a period (".") as the first character. The work around is to temporarily rename the file.

6. The RPT$ statement generates unpredictable results if the *count* parameter is not an integer. Use the INT function to convert floating point numbers to integers. RPT$ generates a "NOT ENOUGH MEMORY" error if *count* is negative. Use ABS to insure that *count* is positive, or do a test for a negative number before executing RPT$.

7. The STATUS statement does not accept the thirteenth *flagset* character. Thus, the clock mode (am/pm—24-hour) cannot be set with STATUS. Truncate the output of the STATUS$ function to 12 characters before using it as *flagset* in the STATUS statement. Use the STATS command in TIME mode to set the am/pm—24-hour flag.

8. Subprograms (with a SUB statement as the first line) cannot be renumbered unless they are allocated. To renumber a subprogram that is not allocated, make the SUB statement a comment, execute RENUMBER, then remove comment status from the SUB statement.

9. If, in a TEMPLATE$ statement, *protect templ* contains no unprotected characters and *trail* is "U", the program will "hang" for about 20 seconds and the cursor will be misplaced.

10. [CTL] [DEL] should not be executed within a template created with TEMPLATE$. [CTL] [DEL] will not function correctly within the template, and if [CLR] is then pressed, the computer will "hang."