

Please Check

CUSTOMER NOTES ON 2200A UPGRADE

Your 2200 has just been upgraded to provide a number of excellent operational improvements and to correct some previous problems. Most of these changes will not effect the operation or programming of your 2200. In a few instances the improvements may require slight changes to your operational procedures and/or programs, depending upon how you may be using the system. These areas are described below.

1. Use of Special Function Keys Immediately After Loading A Program

PREVIOUS OPERATION:

The 2200 was originally designed to undergo resolution processing each time a special function key entry was made to the program. With this processing the program was prepared for execution. That is, the entire program was scanned to insure that the program syntax was correct, and all program variables were setup in a Variable Table if not already there.

IMPROVEMENT:

As we gained more experience with 2200 applications, it became apparent that in most cases the resolution processing for special function entry was undesirable. It consumed a substantial amount of processing time, which was unnecessary since in vast majority of times the special function key operations are used, they are used after a program has been initially executed (and resolved). We therefore, eliminated resolution processing with special function entries.

OPERATIONAL CHANGES:

This change provides substantial improvement in the speed of special function operations. It does, however, place one additional restriction on system operation. If a program has been loaded by a LOAD command, it can no longer be initially executed with a special function key, (since resolution processing must be done once). If you setup programs to be initially started in this manner, the following simple changes can insure proper start of execution:

- a. Insert a STOP statement in your present programs as the very first statement. (Can be placed before COM, DIM).
- b. Change operation procedures to include depressing:
 RUN (Execute/Return)
after initially loading the program
- c. Thereafter normal special function operating procedures can be as followed.

Alternatively, instead of a STOP, a GOTO could be inserted as the first program statement, which would branch to the section of program that is to be initially execute.

~~ALTERNATIVELY~~

It should be noted that program loaded via overlays, (i.e., LOAD statements within a program) do not require any initialization or changes in procedure.

2. Use of Special Function Keys After Program Execution

PREVIOUS OPERATION:

Because of the previously discussed resolution processing associated with special function key entries to a program, it was illegal to use special function keys when the system was awaiting data requested by an INPUT or MAT INPUT. In addition the special function entry cleared previously stacked information on uncompleted Subroutines and FOR LOOPS. The special function entry acted like a subroutine called from the keyboard (except for TEXT ENTRY), and stacks return information. When RETURN was executed, program execution terminated and the system returned to CONSOLE INPUT mode.

IMPROVEMENT:

A special function key can now be entered while the system is awaiting input to a INPUT or MAT INPUT statement. When RETURN from the special entry is made, the INPUT or MAT INPUT statement is re-executed. Special Function Entry from CONSOLE INPUT mode is similar to the previous operation. Stacked incompleted Subroutines and FOR LOOPS information is no longer cleared. *However,*

OPERATIONAL CHANGES:

Special function entries other than TEXT ENTRY act similar to a subroutine call (i.e., an entry is made in a system storing return information for the subroutine call). When a RETURN is made (with a RETURN statement), either the system stops execution and returns to CONSOLE INPUT mode or re-executes a pending INPUT or MAT INPUT statement. Because in the previous design each special function entry cleared previous active Subroutine and FOR LOOP information, it was possible to make many successive special function entries without executing a RETURN. (Although *the manual stated a RETURN was necessary* ~~this was not recommended~~ in the manual). Because the previous subroutine information is now no longer cleared, it is now a firm requirement that special function entries exit by a RETURN, if they are done repetatively. (*otherwise, the stack will overflow* ~~Other standard~~ subroutine information will eventually fill a table and cause ERROR 02, table overflow).