Computation Center
March 9, 1964

FROM: Thomas N. Hastings

SUBJ: Requirements of the FAPBUG Program

The following memo describes changes or additions to the present time sharing system required in order to allow the FAPBUG program to operate as described in CC Memo-225.(1) All of themodifications are of a general nature and may be used by any program. These modifications will be especially useful to any B-core supervisory programs such as FAPBUG, classroom monitors, or a new, undebugged version of the time sharing system. In fact, the changes outlined here will permit a user to set up an arbitrary number of B-core supervisors in a hierarchical relationship below the A-core supervisor. Examples of the need for this capability are: the user using FAPBUG to debug a classroom monitor, a student using FAPBUG to debug his program while running under the classroom monitor, and a staff member debugging a new version of the time-sharing system by running it in core B.

From the point of view of the supervisor, FAPBUG is no different from any other user program. However, FAPBUG is a lower level supervisor from the point of view of the user's program. Thus any supervisory functions which the user's program may call should be reflected through FAPBUG, the B-core supervisor. This could be accomplished by a small bootstrap which would be placed in B-core along with the user's program. This bootstrap would coordinate the time-sharing in B-core of the FAPBUG program and the user's program, as only one of them would actually be in core at any given time. Some supervisor functions would be modified slightly by the bootstrap to make the user's program completely unaware of the presence of the bootstrap. The only restriction which would be placed on the user is that he could not use the 256 to 512 words of memory occupied by the bootstrap.

The above considerations lead to the following requirements.

1. All user pseudo-machine conditions such as:

   interrupt level
   break characters
   acknowledge mode
   BCD or 12 bit mode
   relocation register
   lower memory protection register
   active files
   etc.

---

(1) T. N. Hastings, "A Manual for the FAPBUG Symbolic Debugging Program", November 22, 1963.

must be testable as well as settable from B-core. FAPBUG will save all of these options, reset them so the user can type into FAPBUG, and restore them before transferring control back to the user's program.

2. The user must be able to set the relocation register and the lower memory protection register. The FAPBUG bootstrap would set the relocation register so that the user's core image would appear to start in location 0. Also the bootstrap would set the lower memory protection register so that the user could not destroy the bootstrap accidentally. See Programming Staff Note 25. "Proposed Memory Protection Simulation for User Programs" for a suggested implementation.

3. Another user option must be the ability to reflect all protection violations (including TIA's, HTR's, and interrupt signals) to a specified location in B-core. The relocation and lower protection registers would be restored in such an event to a specified setting, so that the transfer location could be memory protected. The FAPBUG bootstrap would set this option so that all protection violations would be reflected into it. Any legitimate supervisor calls would then be passed onto the supervisor and would be executed memory protected. The following supervisor entries would require special processing by the bootstrap:

SETMEM — setting the relocation register
GETMEM — setting the lower protection register
reflecting protection violations

so that the user's program would operate as if the bootstrap were not present.

4. Since the bootstrap must do some post processing after execution of an A-core subroutine requested by the user, it must be possible to request the supervisor to reflect all A-core subroutine returns to a particular location in B-core.

5. The relocatable loader should write the M∅VIE) table out on file M∅VIE) BINARY, deleting all previous copies. This is so that there is no possibility of destroying the loading information. The M∅VIE) table must include the file name from which the programs were loaded as well as the origin and entry points.

6. FAPBUG expects that every program has a unique file name associated with it. This gives the user a convenient way of specifying any one of his programs.

7. FAPBUG will need to have the translators generate symbol tables according to the new format as specified in Programming Staff Note 17.

8. In order to allow the user to type in 12-bit Hollerith information to FAPBUG from his console, the special control character scheme would be best. This special character would be interpreted in the break processor and would set the console input to 12 bit mode until the next occurrence of the special characte