MODIFICATIONS TO THE SDS 930 COMPUTER FOR THE IMPLEMENTATION OF TIME-SHARING

W. W. Lichtenberger

M. W. Pirtle

W. J. Sanders

University of California, Berkeley

Document No. 20.10.10

Issued January 22, 1965

Revised August 27, 1965

Contract No. SD-185

Office of Secretary of Defense

Advanced Research Projects Agency

Washington 25, D. C.

This document describes changes made to the SDS 930 used in the Berkeley Time-Sharing System.

1.0 GENERAL

Every multi-programming computer system must be able to provide isolation between the independent concurrent operating programs. Otherwise, such programs may interfere by overwriting each other with data, by transferring control to each other, by attempting to use the same input/output devices, or by halting or otherwise hanging up the computer. Memory protection, the trapping of I/O and illegal instructions, and centralized, system-controlled I/O will usually solve isolation problems. The problems of relocation of program areas and allocation of storage among the concurrent programs arise when main memory cannot contain all of the concurrent programs and it is necessary to move them to and from secondary storage. A mechanism which provides a solution to the relocation and allocation problems and also provides memory protection is discussed in Section 3.0.

2.0 PRIVILEGED INSTRUCTIONS

To insure mutual isolation of users' programs, it is necessary to restrict users to a subset of 930 orders. Forbidden orders are termed "privileged instructions". In essence, the absence of privileged instructions from the normal repertoire redefines the machine which the user has at his disposal. We therefore think of two computers (more precisely, two modes of operation of the 930) --- a user's mode and an executive or monitor mode. Because both modes entail changes in programming conventions in the 930, it is necessary to have a third or normal mode. The mode of the machine is set by an EOM and control transfers as described in Section 6.0.

The set of privileged instructions consists of all undefined order codes, halt, all input/output orders, and all sense orders except for overflow test. An attempt to execute a privileged instruction while in user mode will result in the execution of a NOP instruction and, subsequently, a trap* to location 40_8. The program counter (P counter) is not incremented during the execution of the NOP instruction. Consequently, the address stored by the BRM instruction in location 40_8 is that of the offending instruction.

Privileged operation codes are: 00, 02 (except 0 20 00001 and 0 20 00010), 03, 04, 05, 06, 07, 10, 11, 12, 13, 15, 21, 22, 24, 25, 26, 27, 30, 31, 32, 33, 34, 40 (except the combination 0 40 20001, the overflow test), 42, 44, 45 and 47.

Defined instructions included in the above list are 00 HLT, 02 EOM (except ROV and REO), 06 EOD, 10 MIY, 11 BRI (cf. Sect. 5.3), 12 MIW, 13 POT, 30 YIM, 32 WIM, 33 PIN, and all 40 SKS (except OVT).

* The term "trap" is to be distinguished from the interrupt defined by SDS. The trap is a forced transfer to a fixed location; hence a trap routine is interruptable by any other interrupt or trap condition.

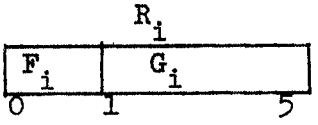
3.0 MEMORY RELABELING

The address field of the 930 consists of the rightmost 14 bits, permitting programs to access directly 16K of core. A memory extension register is provided to allow programs to access 32K. The use of this register is described in detail in the SDS 930 Computer Reference Manual.*

The standard SDS memory extension is not used in the time-sharing system. Rather, the following memory relabeling scheme has been implemented:

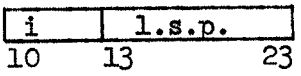
Eight relabeling registers of six bits each are laid out in two registers RL1 and RL2 as follows:

Each of these eight registers contains information as shown below:

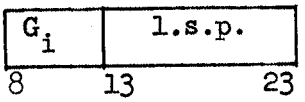


where F_i is a flag bit and G_i is the least significant 5 bits of R_i. Thus R_i may be thought of as a 5-bit register G_i with an associated flag F_i.

When relabeling, the contents of G_i, where i is the value of the three most significant bits of the address, are concatenated with the least significant eleven bits. Thus, the address



becomes



---

*Cf. SDS 930 Computer Ref. Manual. No. 900064B, Scientific Data Systems, Inc., Santa Monica, California, 1964, P. 4.

The reader will note that this scheme permits ultimate access to 64K of memory in 2K blocks. Because any combination of bits can be used in the eight registers, a user's program may occupy as much as 16K located randomly in non-contiguous blocks of 2K throughout the memory. The substitution of bits (or relabeling) is performed on the address presented to the memory by the machine, hence the user's program is effectively connected together into one strip of continuous memory beginning at (local) location 0. The problem of relocation is thus eliminated and the problem of allocation is greatly simplified.

Memory protection reduces to allotting memory to each user in multiples of 2K and detecting when the user attempts to exceed his allotment. In our scheme, a memory reference pointing to an R_i with the contents 1000000_2 is an indication that the block of memory involved has not been assigned, and it results in a NOP and a trap to location 00041_8. At the occurrence of the trap, the P counter contains the location of the offending instruction, except in the case of an attempted jump to an out of bounds location, in which case it contains the following information:

Notation: In q: OPN α, q is the unrelabeled location of the operation and α is the effective unrelabeled address.

q: BRM α $\begin{cases} 1) \ \alpha \text{ illegal, } (P) = q \\ 2) \ \alpha \text{ legal but } \alpha+1 \text{ illegal, } (P) = \alpha + 1 \end{cases}$

q: BRR α $\begin{cases} 1) \ \alpha \text{ illegal, } (P) = q \\ 2) \ \ (\alpha)+1 \text{ illegal, } (P)=(\alpha)+1 \end{cases}$

q: POP $\quad (P)=q$

q: $\begin{Bmatrix} \text{BRU} \\ \text{BRX} \end{Bmatrix}$ $\quad (P) = \alpha$

An intermediate level of memory protection is afforded by the flag bits F_i. Reading and writing in any assigned block (i.e., $(R_i) \neq 1000000_2$) of memory is permitted if the associated $F_i = 0$. If $F_i = 1$, the associated block is read-only. An attempt to store information in a read-only block results in a NOP and a trap to location 00043_8. The P counter contains the same information as it would in the case of an absolute protection violation.

To set RL1 it is necessary to execute an EOM 21000, which clears the register, followed by the execution of a POT instruction. To set RL2 an EOM 20400 is executed.

Normal addressing is also used under certain conditions. When the relabeling registers are used, however, special addressing is said to apply.

4.0 USER MODE

The machine which the users program, i.e., the 930 in the user mode, is as described in the computer manual except for the following changes:

4.1 All privileged instructions are forbidden.

4.2 A new class of operations called system programmed operators (SYSPOP) is provided. Although system programmed operators are, in fact, ordinary programmed operators, the user thinks of them as new and more powerful machine instructions since he does not have to allocate any of his own storage for them. In addition, the user may define his own set of programmed operators as he desires and exactly as explained in the manual. The distinction between system and local programmed operators is described in detail in Section 6.0.

4.3 Special addressing applies to all instructions in user mode.

5.0 MONITOR MODE

In monitor mode, the 930 has its full complement of orders including the privileged instructions. Addressing is normal, and the memory extension register may even be used if desired. Two changes distinguish this mode from the normal mode.

5.1 If an instruction is executed in which the sign bit (which is normally unused) is one, special addressing (relabeling) applies for that instruction only. Monitor programs can thus conveniently access information in user areas. Special addressing will also apply to any instruction for which the sign bit of any word fetched during the determination of an effective address is equal to one. More precisely, relabeling becomes effective when the sign bit is detected, and the machine will remain in this mode for the duration of the current instruction. Thus, if the sign bit of a word fetched during indirect addressing is equal to one, all further references to memory made by this instruction will be relabeled.

5.2 Because of the technique adopted for changing modes, it is necessary to modify the convention for storing the contents of the overflow indicator at the time of performing subroutine entries. Normally, the state of the overflow indicator is stored in the sign bit of the subroutine link. Since the sign bit is now reserved to indicate special addressing in monitor mode, it is necessary to move the state of the overflow indicator to Bit 2 of the link. Note that this applies <u>only</u> in the case of monitor mode and is <u>not</u> true in normal mode or user mode.

5.3 To enable interrupt routines to restore the overflow indicator properly on return, a new instruction BRI (01100000) has been added. BRI (Branch and Return from Interrupt routine) functions in a manner similar to BRR with the following exceptions:

1. It does not increment the return address.
2. It first clears the overflow indicator and then sets it with the contents of Bit 2 in the return address word. (BRR simply merges the two indicators.)
3. It terminates the current priority interrupt level. BRI is a privileged instruction and hence cannot be executed in user mode. It should be noted that in monitor mode, the termination of interrupt levels is no longer accomplished by BRU*, hence it is legal to do a BRU* in an interrupt routine. Furthermore, BRI* may be executed to any depth. In normal mode, termination of interrupt levels is accomplished both by BRU* and BRI. The existence of a new instruction, BRI, in normal mode is a departure from the design goal of preserving normal SDS 930 operation in normal mode; BRI is, however, otherwise an undefined instruction, and it is advantageous to be able to run hardware diagnostics in both monitor and normal modes.

6.0 CHANGING MODES

Pushing the start button on the console forces the machine into normal mode. This is the only manner in which the transition to normal mode can be made. The transition from normal to monitor mode is made by executing an EOM 22000. The transition from monitor to user mode is made by executing any jump to a relabeled location. The user can cause a transition from user to monitor mode by executing a SYSPOP. There is no means of going directly from normal mode to user mode.

It should be noted that, although the above-mentioned means of making mode transitions exhaust the possibilities available to the programmer, there are two other causes of such transitions. First, the occurrence of an interrupt or trap when in user mode will cause a transition to monitor mode. Secondly, following the execution of a single instruction interrupt routine, a transition to user mode will occur if the machine was in user mode at the time that the interrupt occurred.

In order that system subroutines be able to serve both the user and the system itself, an indication of the mode before entry is preserved in the subroutine link. Bit 0=1 implies a transfer from user mode, and Bit 0=0 implies an entry from the system. Bit 0 is used for this purpose in order to make data access independent of mode (cf. Section 7.0) and to restore the proper mode upon return.

When attempting to execute a transfer from monitor mode to relabeled memory (and thus to user mode) which is out of bounds, the resulting trap forces Bit 0 of the link to a 1. The monitor must take this effect into account.

7.0 PROGRAMMED OPERATORS

In his program, the user may execute one of two types of programmed operators. An instruction in which Bit 0 is 0 and Bit 2 is 1 is a normal programmed operator, local to the user's area of memory. As such, the user must allocate space in local locations 100 through 177 for transfers to programmed operator subroutines in his own memory. If, however, Bit 0 of the instruction is 1 and Bit 2 is 1, the machine changes to monitor mode before executing the programmed operator. Thus, the user is sent automatically to actual locations 100 through 177, where system programmed operators service his program.

System programmed operators are included in the system routines mentioned in Section 6.0. The link for a programmed operator is location 0. If a user executes a SYSPOP, Bit 0 of the link is 1. Since programmed operators refer to their data indirectly via their link, special addressing is applicable and the user's data will be accessed. On the other hand, if the system programmed operator is used by the system itself, Bit 0 will be 0, and normal addressing will apply. Bit 0 may be inspected by the system to determine at interrupts whether the user was in his own program or whether he was in a system programmed operator.

Programmers should realize that in user's mode, Bit 0 has significance in the case of programmed operators. It is an error, then, to use Bit 0 of a programmed operator as storage for any purpose. Bit 0 is otherwise unrestricted for the user.

8.0 OTHER CHANGES

The following changes are not visible to the user, but serve to provide for security of the system from user action. The occurrence of an interrupt request from the interrupt priority logic during the execution of an "execute" instruction (or a long chain of EXUs) results in the termination of the process and the execution of a NOP. At the completion of the NOP, the highest priority interrupt request is honored, and the P counter contains the address of the interrupted instruction; hence, the normal interrupt routine exit will return to the interrupted instruction which will begin execution anew.

Similarly, when relabeling, the execution of instructions involving indirect addressing is interrupted when an interrupt request occurs during the indirect addressing phase of the execution.

Also, an interrupt request at the completion of a BRX instruction which calls for a jump causes the execution of a NOP (at the completion of which the interrupt can occur). In this case the P counter contains the location specified by the jump.

Finally, when in user mode interrupt requests are honored immediately following the execution of ROV and REO instructions.

Each of the features described above is effective both in monitor and user mode; in normal mode none function -- the CPU behaves exactly as a normal SDS 930.