bcc	title CPU INTERRUPTABILITY	prefix/class-number.revision CPUINT/W- 20	
checked Charles Simonyi	authors Butler W. Lampson Butler W Lampson	approval date 8/15/69	revision date
checked		classification Working Paper	
approved Mel		distribution Company Private	pages 4

ABSTRACT and CONTENTS

The behavior of the CPU with respect to attention, single-step, and crash signals, the CPU protect signal (4); the timer trap, the XMON and XUTIL traps, and initialization is described. This document will be incorporated into a revision of the CPU manual.

The relevant information is:

A) Some information in the state

1) The ring in which the P-counter is contained.

2) The XMON trap bit in SR

3) The XUTIL trap bit in SR

4) The sign bit of the interval timer, which we call TO.

B) Some flip-flops in the microprocessor which are not part of the CPU state

1) STROBE, which may be set by another microprocessor, normally the microscheduler

2) STEP and PROCEED, which may be set by some external device, which will be called the test processor (TP), to make the CPU operate in single-step mode.

3) LOCKED, which is not accessible to external devices

4) ALARM, which is set when a system crash is impending

C) The state of protect 4, which will be called CPUPRO

Idle State

When it is initialized (by setting O to Ø) the CPU goes into idle state.

```
IDLE:    GOTO IDLE IF NOT STROBE;
         Clear STROBE.
PWAIT:   P ← contents of absolute cell ( 6 + CPU#(=Ø or 1 and
              wired into microcode))
         Goto PWAIT if P = Ø;
         Clear absolute cell (6 + CPU#);
         Clear LOCKED;
         Find the page with real name in (P), (P+1) and (P+2).
              Take it as a context block and load the state from
              locations 2764B-2777B in it (called the SAVE area).
         Start executing instructions at the location given by
              the P-counter;
```

The CPU returns to the idle state whenever it dumps the state of a process.

Interruption of program execution

At the start of every instruction, the truth of any of the following conditions will stop execution and cause the indicated action to be taken. The conditions are tested in the order in which they are listed.

1) NOT LOCKED AND TO: cause the timer overflow trap.

2) NOT LOCKED AND STROBE: dump the state into the SAVE area,. send a RETURN message to the μscheduler and go into idle state.

3) STEP OR ALARM: dump the state into the SAVE area, clear STEP. Wait for PROCEED, clear it, reload the state from the SAVE area and proceed.

At every step of indirection, every start of an instruction which is the target of EXU, every parameter of a BLL and in all other places where the CPU might be held up for more than a few microseconds, conditions 1 and 2 are tested and their indicated actions taken.

Setting the bits

XMON and XUTIL are part of SR and may be set or cleared with SRS, LOADS or XSA. Like the rest of SR, both are cleared when an MCALL, UCALL or fixed trap occurs (after storing the state in the latter case).

LOCKED is set by MCALL or fixed trap. It can also be set by SLOK. It is cleared by any BLL or LOADS which leaves the monitor ring (BLL, here and elsewhere, includes all variants: UCALL, MCALL, POP, GRET), and can also be cleared by RLOK.

TO can be changed by loading a state from the SAVE area or by the OPRs to set the interval timer.

The X traps

At every BLL or LOADS a check is made for transition into a lower ring. If there is a transition from monitor to utility or user rings, the XMON trap is caused if the XMON bit is set. Then if there is a transition from utility to user ring, the XUTIL trap is caused if the XUTIL bit is set.

The CPUPRO signal

This protect is set automatically at each point where LOCKED is set and cleared at each point where LOCKED is cleared. The programmer can set it himself with the PRO operate, but this is probably unwise.