

Published: 11/17/67

Identification

Fault-Interrupt Utility for Initialization

set\_vector

D. R. Widrig

Purpose

Many Multics initialization programs require the setting of the attached processor's fault and interrupt vector so that faults and interrupts may be directed to the proper interceptors. Set\_vector was written to facilitate straightforward patching of the proper vectors.

Implementation

To patch a particular fault or interrupt pair, the following call is made:

```
call set_vector(scuptr, traptr, cellno, int_sw)
```

where the arguments are declared as follows:

```
    decl (scuptr,          /* ITS pointer to location to
                           store SCU data */
          traptr) ptr,    /* ITS pointer to interceptor
                           receiving control */
          (cellno,        /* pair # to be patched */
          int_sw) fixed,  /* 1 if interrupt vector,
                           0 if fault vector */
```

Set\_vector calls the SLT manager, (BL.2.02), to get a pointer to the processor's fault vector. The standard fault-interrupt vector arrangement outlined in BL.4.01 is assumed. Depending on the setting of "int\_sw", the proper ITS pairs for "cellno" are patched in the appropriate SCU-TRA blocks. "Cellno" must be in the range 0-31. Errors in patching the vector result in a call to "panic".

EXAMPLE

To patch the fault vector to allow linkage faults, (linkage fault = fault 16 octal), the following call is made:

```
call set_vector (scuptr, traptr, 14, 0);
```

After successful return from the call, all further linkage faults will store the SCU data in the area pointed at by "scuptr" and control will transfer to the location pointed at by "traptr".