22 Feb '71

NEW STACK LOGIC

The manipulation of the call stack is being extensively revised. The most significant changes are:

1) When a subprocess does a system call, the p-counter in the stack will point at the XJ, not one beyond it;

2) A p-counter qualifier will indicate whether the subprocess was
   a) about to execute the inst at p-counter
   b) in the middle of the inst at p-counter (presumably an XJ)
   c) almost finished with the inst at p-counter "

3) A return ~~instruction~~ action which will modify the p-counter qualifier of the ~~stack entry returned to~~ previous stack entry as part of the action will be provided.

4) The interrupt inhibit bit ~~in the top~~ will always be set when a new top of stack is formed, so that the current, running subprocess will automatically have interrupts inhibited. An operation to explicitly set* and clear the bit will also be provided.

5) The forced f-return and interrupt flags will disappear. I would like to move the interrupt inhibit bit from its present position if no one objects.

A complete description of the display stack operation and an "internals" specification for the new stack should be available soon.

Is there any enthusiasm for an action to display stack entries from some other process?

*Oh my God, a split infinitive!