Questions about supporting interrupts.

1. Discuss the following sentence: "Interrupts are an unpleasant fact of life" [Andrew S. Tanenbaum, Modern Operating Systems, p. 212].

2. What is the difference between the following two sub-types of aborting interrupts:

   - external - causes all processes to abort.
   - internal - aborts the current process.

   What should an aborting interrupt do with the interrupt stack?

3. Define: (a) handling interrupts precisely (b) response time of interrupt handling. Your definition should address issues such as: distinction between external and internal interrupts, reconstructing the state of the machine, distinction between aborting interrupts and interrupts of type continue or repeat.

4. Why aren't the contents of CA and MAR registers restored during return from exception?

5. Consider the following suggestion for handling non-aborting external interrupts. Every external device (i.e. external interrupt) has a dedicated process. The mechanism for handling external interrupts uses a synchronization mechanism (e.g. semaphores) to signal that an interrupt occurs or has been serviced. Whenever a process has an activated event signal (for an external interrupt), a kernel procedure is invoked which signals via the synchronization mechanism that the external interrupt has occurred. (Of course, signaling in the synchronization mechanism is done in an atomic way). After that, the kernel checks to see which process should be activated according to priorities.

   The following questions refer to the suggestion above:

   (a) Can one integrate the handling of external interrupts with the context switching mechanism?
   (b) Is an interrupt stack required for handling external interrupts?
   (c) Where are priorities defined (if at all)?
   (d) How does one deal with nesting of external interrupts?
   (e) How can one deal with internal interrupts?
   (f) Compare this suggestion with the mechanism taught in class. Address the following issues:
       - Simplicity. What is demanded from the hardware and the operating system?
       - Preciseness.
       - Response time.
       - Nesting of interrupts.