**Chapter 8**

**8.6 What problem could occur if a program does not check the Ready bit of the KBSR before reading the KBDR?**

Input devices usually operate at speeds very different from that of a microprocessor which is asynchronous. Therefore, a program checks the Ready bit of the KBSR before reading the KBDR to test if the program is available to input. When the KBSR[15] is 1, the ASCII code corresponding to the last key struck has not yet been read, and so the keyboard is disabled. If not, the program will read repeatedly from KBDR.

**8.9 What problem is likely to occur if the keyboard hardware does not check the KBSR before writing to the KBDR?**

The program may load in character before it is read, so that character will lost.

**8.12 Adam H. decided to design a variant of the LC-3 that did not need a keyboard status register. Instead, he created a readable/writable keyboard data and status register (KBDSR), which contains the same data as the KBDR. With the KBDSR, a program requiring keyboard input would wait until a nonzero value appeared in the KBDSR. The nonzero value would be the ASCII value of the last key press. Then the program would write a zero into the KBDSR indicating that it had read the key press. Modify the basic input service of Section 8.2.2 to implement Adam's scheme.**

START LDI R0,A

BRz START ;Branch if KBDSR

AND R1,R1,#0

STI R1,a ;Clear KBDSR

BR NEXT\_TASK

A .FILL xF000 ；Address of KBDSR

**8.14 An LC-3 Load instruction specifies the address xFE02. How do we know whether to load from the KBDR or from memory location xFE02?**

Addresses x0000 to xFDFF are allocated to memory location. Addresses to  xFE00 to xFFFF are reserved for I/O device registers.

**8.16**

Output ‘ABCDEFGHI’ to display.