**Input/Output**

**Memory-Mapped I/O vs I/O-Mapped**
Memory-mapped I/O is when I/O device registers are mapped to memory addresses. Therefore, I/Os can be performed using the same load and store memory access instructions.

**I/O-Mapped**
I/O device registers have their own address space. Special I/O instructions are needed to perform I/O.

The LC3 uses memory-mapped I/O. Addresses xFE00 to xFFFF are reserved for I/O device registers.

**Asynchronous vs Synchronous**
I/O devices usually operate at speeds much slower than a microprocessor, and not in lockstep. Operating not in lockstep is referred to as asynchronous. To control processing in an asynchronous world requires some protocol or handshaking mechanism. For example, the keyboard needs to tell the microprocessor when a user has pressed a key using a ready bit flag. Each time the typist types a character, the ready bit is set. Each time the microprocessor reads a character, it clears the ready bit.

The LC3 uses asynchronous communication.

**Interrupt-Driven vs Polling**
Continuing with the above keyboard example, the question is how does the microprocessor know when the ready bit has been set? One way is by polling where the microprocessor is continuously checking to see if the ready bit has been set or not. If it is set then it will go and read in the key. This method does not require any extra hardware support but waste a lot of CPU time for the microprocessor to continually check the ready bit. A more efficient method, but requires extra hardware support, is to use an interrupt. The microprocessor is doing its own thing until it is interrupted by the keyboard, at which time it will then go and read in the key.

The LC3 uses the polling method.
Keyboard
Two registers are used for communication between the keyboard and the microprocessor.

- Keyboard data register (KBDR) is assigned the memory address xFE02, bits 7 downto 0
- Keyboard status register (KBSR) is assigned the memory address xFE00, bit 15

The status bit KBSR[15] controls the synchronization of the fast processor and the slow keyboard. When a key is pressed, the ASCII code for that key is loaded into KBDR[7:0] and the status bit KBSR[15] is automatically set to a 1 by the keyboard electronics. When the microprocessor reads KBDR, the status bit KBSR[15] is automatically cleared by the keyboard hardware.

```
GETC     LDI   R1, KBSR    ; Test for
BRzp    GETC    ; character input
LDI     R0, KBDR    ; read in character
BRnzp   continue
KBSR    .FILL xFE00
KBDR    .FILL xFE02
continue ...
```

Display
Two registers are used for communication between the display and the microprocessor.

- Display data register (DDR) is assigned the memory address xFE06, bits 7 downto 0
- Display status register (DSR) is assigned the memory address xFE04, bit 15

The status bit DSR[15] controls the synchronization of the fast processor and the slow monitor. When the LC3 processor transfers an ASCII code to DDR[7:0] for outputting, the electronics of the monitor automatically clears DSR[15] to notify the monitor that there is valid data in DDR[7:0] to be processed. When the monitor finishes outputting the character on the screen, the monitor automatically sets DSR[15] to signal to the processor that the processor can transfer another character.

```
OUT     LDI   R1, DSR    ; Test to see if
BRzp    OUT    ; output register is ready
STI     R0, DDR    ; output character
BRnzp   continue
DSR    .FILL xFE04
DDR    .FILL xFE06
continue ...
```