ECE/CS 252
Intro to Computer Engineering

Discussion

I/O Concepts 1

- **Input**
  - Getting input from the user or outside world
- **Output**
  - Sending information to the user or outside world
- The direction of input and output is from the **processor’s** point of view

I/O Concepts 2

- Memory-mapped IO versus isolated IO
    - To use IO devices, we need to be able to read/write from/to them using instructions
    - The LC-3 (and most processors) put the interface to IO devices into the memory, so we can access the devices using normal load/store instructions
        - This is **memory-mapped IO**
    - Some processors (most notably the Intel x86 family) have special instructions (and a different address space) to access IO devices
        - This is **isolated IO**

I/O Concepts 3

- Synchronous versus asynchronous
  - **Synchronous** = occurs at the same rate as the processor clock
  - **Asynchronous** = occurs at an unpredictable rate
- Since we don’t know when asynchronous events will occur, we need
  - A way to check if the event has occurred (a *synchronization mechanism*)
  - A way to transfer the acknowledge that we have seen the event (a *handshaking protocol*)

I/O Concepts 4

- Polling versus interrupts
  - **Polling** = the processor must continually check to see if an event has occurred
  - **Interrupts** = the processor hardware causes to automatically stop normal processing and run a different part of the program to *handle* the event
  - We will only discuss the use of polling today
- Does the telephone system use polling or interrupt?
  - How would the telephone function if it used polling?

Using Console Input

- Console input is a simulated keyboard
  - You can type characters on the "keyboard" in the PennSim window
- The interface to the keyboard uses two memory locations (aka memory-mapped registers)

Name	Address	Description
KBSR **K**ey**b**oard **S**tatus **R**egister	xFE00	KBSR[15] is 1 when new character is available, KBSR[15] is cleared by reading KBDR
KBDR **K**ey**b**oard **D**ata **R**egister	xFE02	KBDR[7:0] has most recently-typed ASCII character, KBDR[15:8] are 0

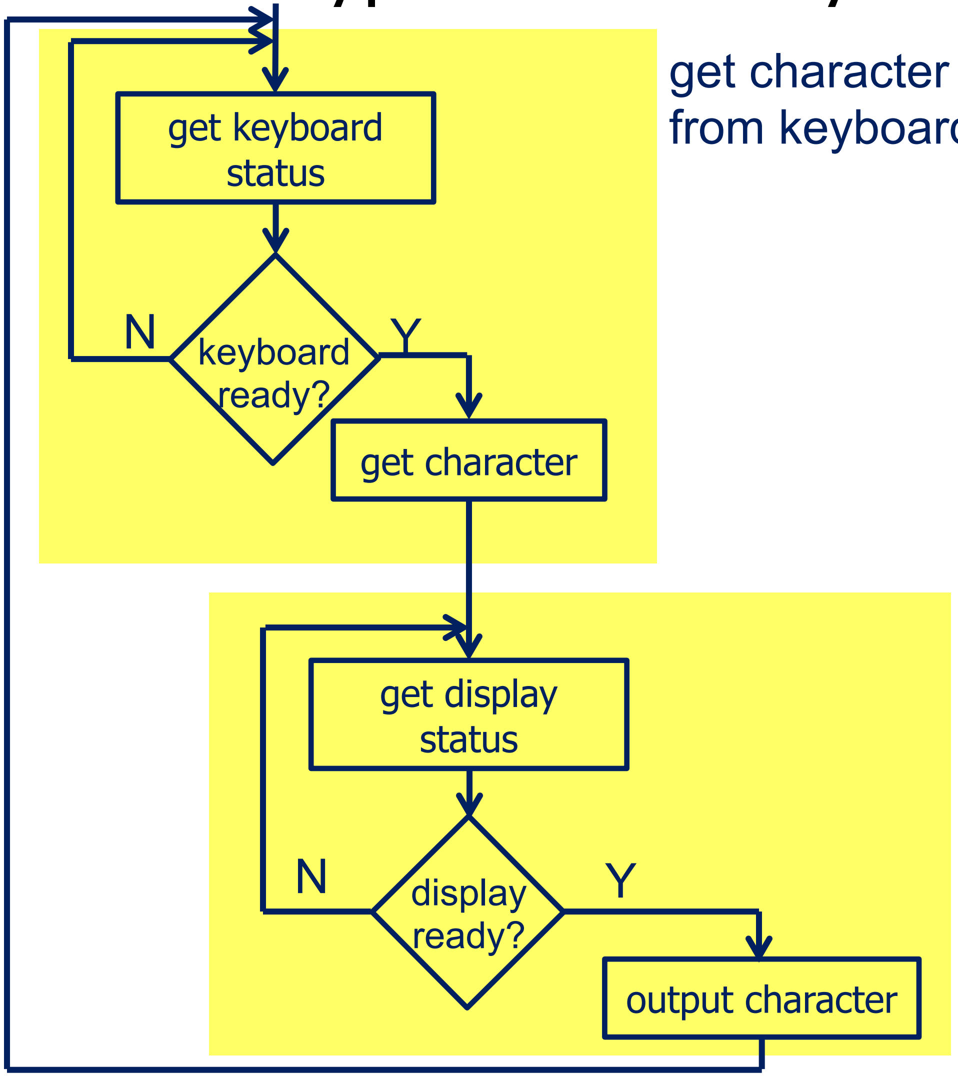
Using Console Output

- Console output is a simulated display screen
  - You can see the console output on the "terminal" window in PennSim
- The interface to the display uses two memory locations (aka memory-mapped registers)

Name	Address	Description
DSR **D**isplay **S**tatus **R**egister	xFE04	DSR[15] is 1 when display is ready to accept a new character, DSR[15] is 0 while display is not ready
DDR **D**isplay **D**ata **R**egister	xFE06	DDR[7:0] is ASCII character to send to display, DDR[15:8] must be 0

Example 0

- Echo each character typed on the keyboard to the display.

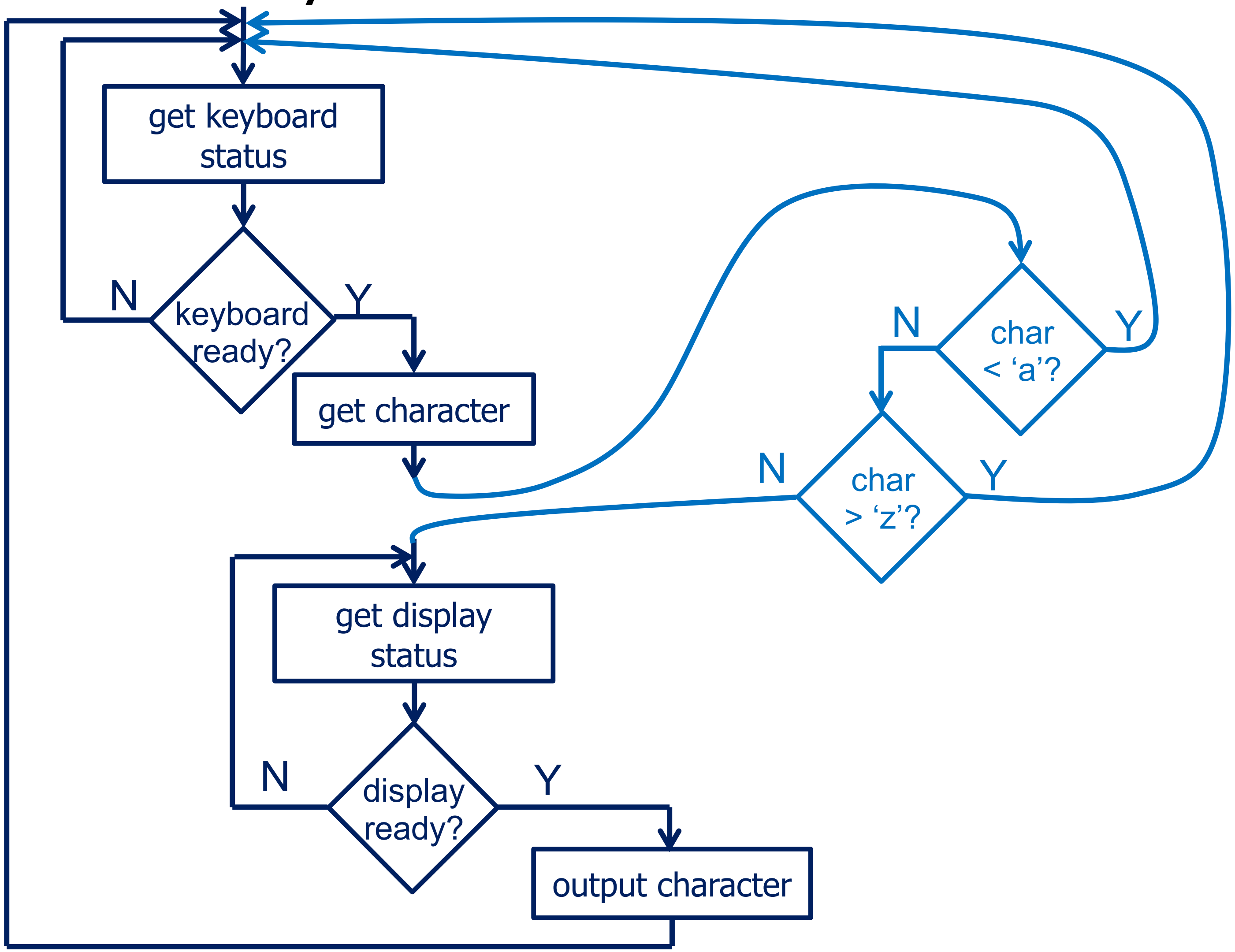


get character from keyboard

output character to display

Example 1

- Echo only lowercase letters.



Wrapping Up

- **The best way to learn programming is to DO programming!**