000000000 01010100 30011100 00002020 20202E4F 52494720 20207833 3030300A E0001300 20204C45 41202052 1C3015C0 794C696E 6509E200 13000000 20202020 4C454120 2052312C 206D794C 696E6540 4F502020 60001600 00004C4F 52205230 2C205231 2C202330 21F00010 00000020 20202020 20202054 52415020 78323105 24001400 00002020 20204C44 20204C44 20205232 2C207465 726D8014 00160000 00202020 20202020 20414444 2052322C 20523002 20202020 00002020 20202020 20204252 7A201854 (F506 12 00 5) 00 02 00120000 00202020 20202020 20421 00 02 16 4F155 16 10 0 2020 20202020 20414444 2052312C 2052312C 04001000 20204841 4C54D0FF 2031F90F 00746572 6D202020 202E4649 4C4C2020 20784646 44306900 00010000 00697400 Lecture x0001 - 01/16 00324000 00010000 00010000 00627200 00010000 00010000 00010000 002D6500 00010000 00666100 00010000 0001000000613200 00010000 00010000 00653200 00010000 00323200 00010000 00323000 00010000 002A0000 00636500 00300000 00010000 202E5354 52494E47 5A202020 20226974 61627261 68324066 6132332D 32302200 

Slides based on material by: Yuting Chen, Yih-Chun Hu & Ujjal Bhowmik

**Dr. Ivan Abraham** ECE 220 - Spring '24



# Memory mapped I/O

- How do we communicate with the computer?
- Memory-mapped I/O: Hardware devices (i.e. their registers) are treated the  $\bullet$ same as the computer's main memory and addressable the same way
  - Memory of *peripherals* is *physically* separate from main memory
- Alternative: Port mapped I/O (requires having more specialized) instructions)
- In LC3: KBDR, KBSR, DSR, DDR are used for [K]eyboard and [D]isplay respectively.



# LC3 - Input/Output (IO)



Figure A.1 - P&P 3rd Ed.

ster Name	I/O Register Function
tatus register 3SR)	The ready bit (bit[15]) indicates if the keyboard has received a new character
data register 3DR)	Bits [7:0] contain the last character typed on the keyboard
atus register )SR)	The ready bit (bit[15]) indicates if the display device is ready to receive another character to print on the screen
ata register DR)	A character written in bits [7:0] will be displayed displayed on the screen



# LC3 - Input/Output (IO)



ster Name	I/O Register Function
tatus register 3SR)	The ready bit (bit[15]) indicates if the keyboard has received a new character
data register 3DR)	Bits [7:0] contain the last character typed on the keyboard
atus register )SR)	The ready bit (bit[15]) indicates if the display device is ready to receive another character to print on the screen
ata register DR)	A character written in bits [7:0] will be displayed displayed on the screen



### LC3 - Input from keyboard **Basic routine** Handshaking is performed using KBSR & KBDR

- When user presses a key
  - Its ASCII code is placed in **KBDR[0:7**]
  - **KBSR**[15] is set to 1 (ready bit)
  - Keyboard is disabled, i.e., any further keypress is ignored
- When **KBDR** is read by CPU
  - KBSR[15] is set to 0
  - Keyboard is enabled



## LC3 - Input from keyboard Basic routine



.ORIG x3000

;Create a loop to check KBSR

;If ready bit unset loop again

;If ready bit set, read KBDR into R0

KBSR .FILL xFE00
KBDR .FILL xFE02



### LC3 - Display to console **Basic routine** Handshaking is performed using DSR & DDR

- When display is ready to present a character
  - DSR[15] is set to 1 (ready bit)
- When a new character is written to DDR
  - DSR[15] is set to 0
  - Any other chars written to DDR are ignored
  - DDR[7:0] is displayed



## LC3 - Display to console Basic routine



.ORIG x3000

;Create a loop to check DSR

;If ready bit unset loop again

;If ready bit set, write R0 into DDR

DSR .FILL xFE04 DDR .FILL xFE06



## Exercise

• Write a program to display "ECE 220 is fun!" to the console. You can use the pseudo-op **.**STRINGZ to store string to memory. Do not use **TRAP** codes (if you know what they are).



## **Issues**?

- Limited amount of GPRs polling display & keyboard uses up two of them
- Code often repeated inefficient to keep inserting same code over • & over again
- Human error keeping track of registers & having direct access to hardware registers is recipe for unforced errors & bugs



## Next time

- Subroutines & repeated code
  - Also called *functions*
- TRAP routines

ILLINOIS