Interrupt Driven I/O

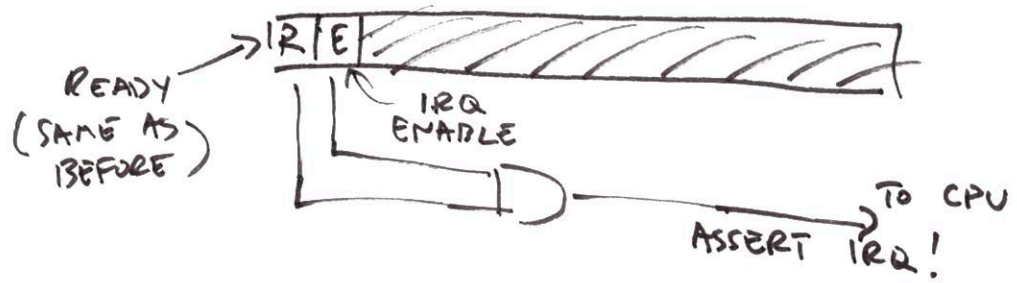
- Asynchronously notify CPU when I/O device needs service

- THIS IS DONE VIA AN INTERRUPT REQUEST LINE (IRQ). THINK OF IT AS A WIRE

- WE MUST GO TO THE SERVICE ROUTINE AFTER INTERRUPT SIGNAL IS RAISED. HOW?
  THINK: LINKAGE

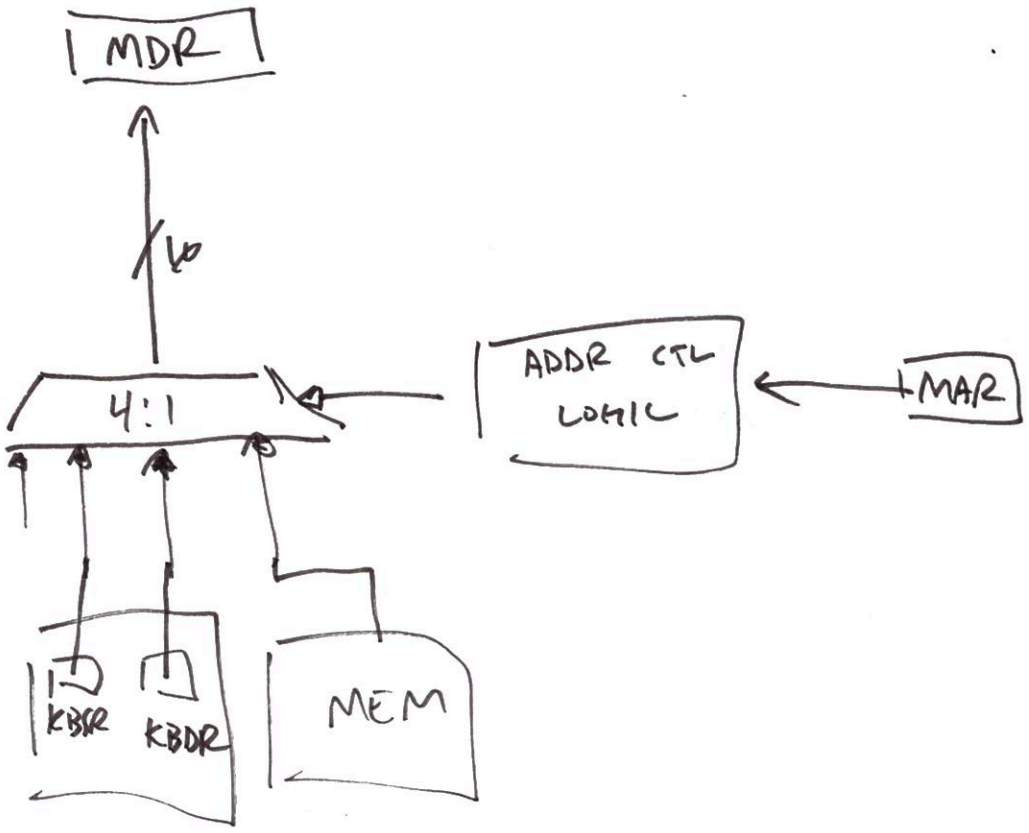
- WHEN DO WE CHECK FOR INTERRUPTS? (STORE RESULTS PHASE)

- MUST MODIFY OUR KBSR:



- WHAT DO WE NEED TO SAVE?: PC, GPRS, ...?

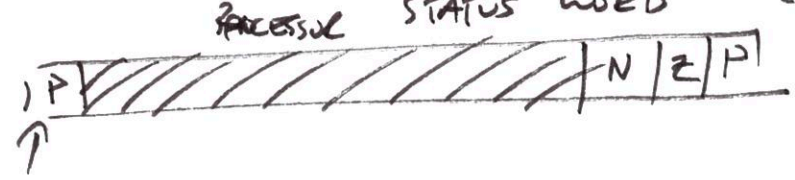
MMIO IN HARDWARE



- what if:

LD R2, Val ← IRQ!
Bzp Target

- CAN'T LOSE OUR CONDITION CODES! ALSO, WHAT PRIVILEGE ARE WE AT?



CURRENT PRIVILEGE LEVEL (0/1)
↑ USER ↑ OS

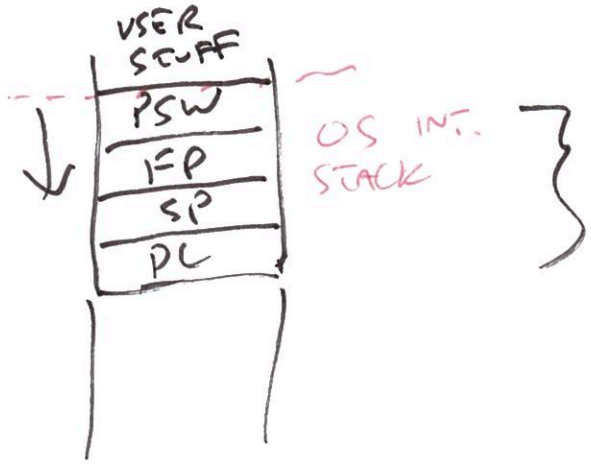
SOMETIMES CALLED "CPL"

- So: [PSW, PC, old SP, old FP]

REGS CAN BE SAVED BY SERVICE ROUTINE.

- WHERE DO WE SAVE THEM? THE STACK!

- How DO WE GET BACK? ADD A NEW INSTR:
RTI (RETURN FROM INTERRUPT)



RTI POPS THIS AND LOADS MACHINE REGISTERS

MMIO IN / OUT

- /proc/iomem ↔ /proc/ioports
- /proc/interrupts
- lspci