Exceptions and Interrupts

• Normal operation of a computer evaluates instructions one after another without pause
• What happens in unusual conditions? Two kinds:
  – Exceptions are unexpected events in execution of the instruction, i.e. mainly errors
  – Interrupts are unexpected changes in control flow caused by events outside the processor, i.e. mainly IO devices
• This distinction is slightly artificial ... there are other views
  – They’re all interrupts
  – Interrupts are external; exceptions are internal; syscall?
Handling Interrupts

• The executing program reports the exception/interrupt; the OS handles it

• Two pieces of information are needed for the OS to help: What happened and where.
  – Add a “Cause” register to store the reason of except/interrupt
  – Add a Exception PC (EPC) register to store the address of the offending instruction

• Set up a means of returning to the operating system
  – Force a branch to fixed memory address (where OS handler code resides)

• Vectored Interrupts are alternative to “Cause”
Multicycle with Interrupt Handling Capability
Revised Control
Review