# CSE 30321
# Computer Architecture I

## Lecture 19 - Control for Multi Cycle Machines

*Michael Niemier*
Department of Computer Science and Engineering



## Microcode: Trade-offs

- Distinction between specification and implementation is sometimes blurred
- Specification Advantages:
  - Easy to design and write
  - Design architecture and microcode in parallel
- Implementation (off-chip ROM) Advantages
  - Easy to change since values are in memory
  - Can emulate other architectures
  - Can make use of internal registers
- Implementation disadvantages, SLOWER now that:
  - Control is implemented on same chip as processor
  - ROM is no longer faster than RAM
  - No need to go back and make changes



## Bad Code

- Let's consider some code that should produce an exception - and think about what happens architecturally:

```
#include <stdio.h>
int main(void) {
   double k;
   k = 2 / 0;
}
```

(now, see what happens when you execute this…)



## Exceptions

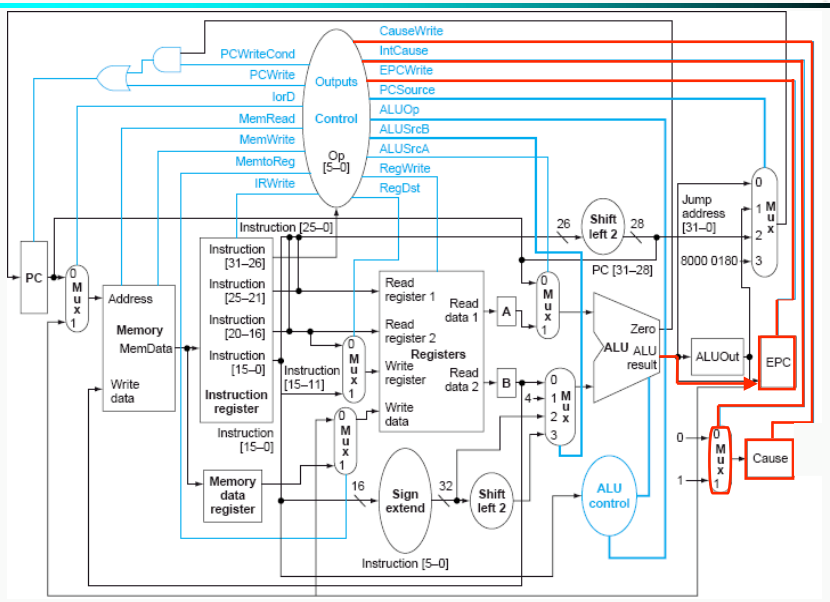
- Exceptions: unexpected events from within the processor
  - arithmetic overflow
  - undefined instruction
  - switching from user program to OS
- Interrupts: unexpected events from outside of the processor
  - I/O request
- Consequence: alter the normal flow of instruction execution
- Key issues:
  - detection
  - action
    - save the address of the offending instruction in the EPC
    - transfer control to OS at some specified address
- Exception type indication:
  - status register
  - interrupt vector

# Exception Handling

- Types of exceptions considered:
  - undefined instruction
  - arithmetic overflow
- MIPS implementation:
  - EPC: 32-bit register, EPCWrite
  - Cause register: 32-bit register, CauseWrite
    - undefined instruction: Cause register = 0
    - arithmetic overflow: Cause register = 1
  - IntCause: 1 bit control
  - Exception Address: C0000000 (hex)
- Detection:
  - undefined instruction: op value with no next state
  - arithmetic overflow: overflow from ALU
- Action:
  - set EPC and Cause register
  - set PC to Exception Address



# Datapath with Exception Handling






# FSM with Exception Handling