Detecting and Handling Exceptions and Interrupts

The datapath has to be prepared to handle unusual situations -- it cannot stop, but must keep going and recover

---

Conditions Arise ...

Classify the unusual things that can happen
- Exceptions -- unusual events that affect the datapath, regardless of whether they are internally or externally generated
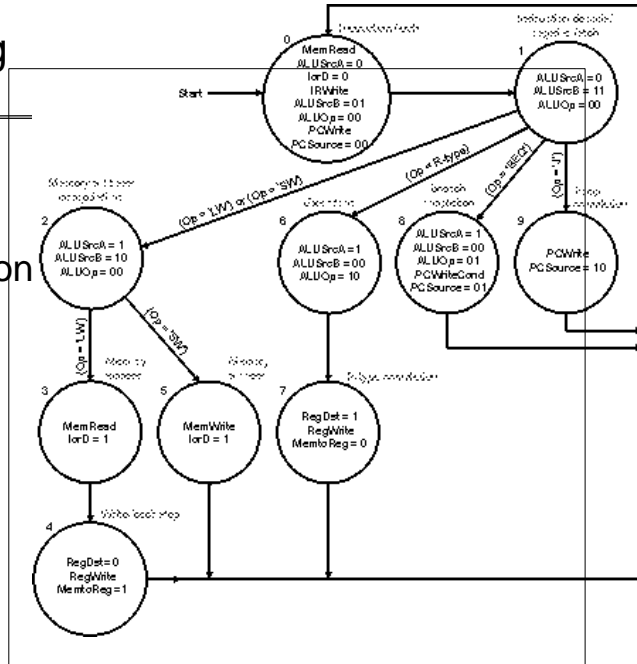- Interrupts -- externally generated events

Examples …
- I/O device requests
- Invoke operating system from user program
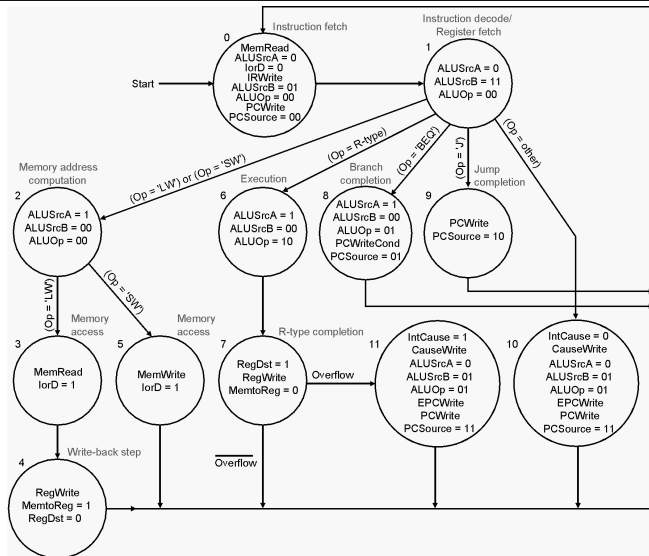- Arithmetic overflow
- Undefined instruction
- Errors

Abstracting Control

Finite state machine representation of datapath

Detection

Instruction fetch

0
MemRead
ALUSrcA = 0
IorD = 0
IRWrite
ALUSrcB = 01
ALUOp = 00
PCWrite
PCSource = 00

Start

Instruction decode/
Register fetch

1
ALUSrcA = 0
ALUSrcB = 11
ALUOp = 00

Memory address
computation

2
ALUSrcA = 1
ALUSrcB = 00
ALUOp = 00

(Op = 'LW') or (Op = 'SW')

(Op = R-type)

Execution

6
ALUSrcA = 1
ALUSrcB = 00
ALUOp = 10

(Op = 'BEQ')

Branch
completion

8
ALUSrcA = 1
ALUSrcB = 00
ALUOp = 01
PCWriteCond
PCSource = 01

(Op = 'J')

Jump
completion

9
PCWrite
PCSource = 10

(Op = other)

(Op = 'LW')

Memory
access

3
MemRead
IorD = 1

(Op = 'SW')

Memory
access

5
MemWrite
IorD = 1

R-type completion

7
RegDst = 1
RegWrite
MemtoReg = 0

Overflow

11
IntCause = 1
CauseWrite
ALUSrcA = 0
ALUSrcB = 01
ALUOp = 01
EPCWrite
PCWrite
PCSource = 11

10
IntCause = 0
CauseWrite
ALUSrcA = 0
ALUSrcB = 01
ALUOp = 01
EPCWrite
PCWrite
PCSource = 11

Overflow

Write-back step

4
RegWrite
MemtoReg = 1
RegDst = 0

Exceptions in the Pipeline

- Handling exceptional conditions in the pipeline is difficult … multiple instructions are in process, but execution must be stopped at a coherent point
- The operating system will handle exceptions
- Two requirements …
  - Save the address of offending instruction in EPC, exception program counter
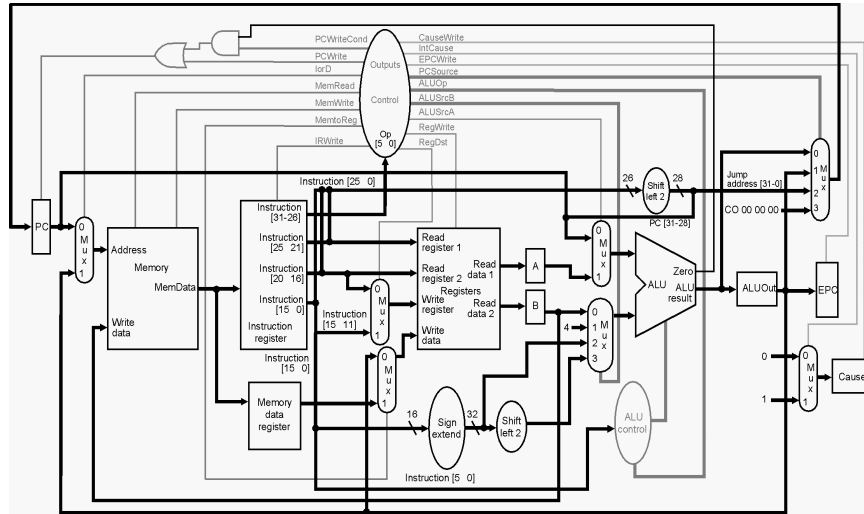 - Transfer control to operating system

Keeping Track of What Happened

- The machine must tell the OS what happened
- A "cause register" or "exception flags" are bit sequences that the processor sets indicating errors
- "Vectored interrupts" allow the processor to jump to different locations in the operating system depending on the exceptional condition
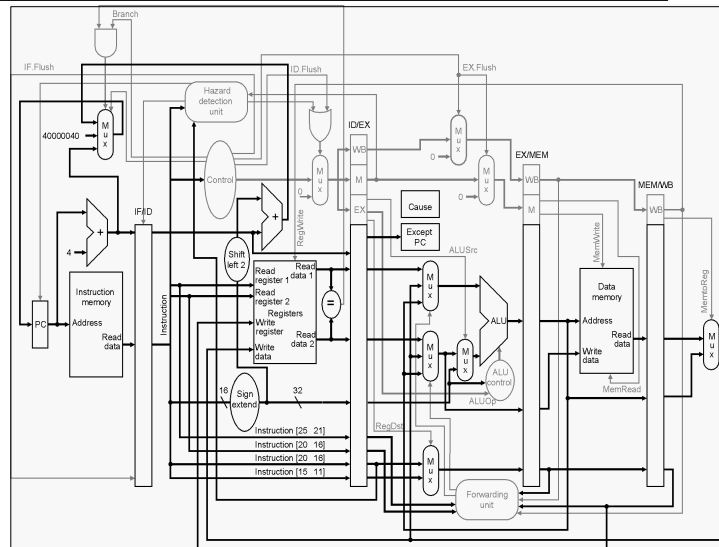
Add-ins for Exceptions

Controls