Topics

- Datapath
- Memory operations
- Microinstructions

Mic-1 Simulator for Class

Java-based simulator which implements the Mic-1 microarchitecture described in Chapter 4
- http://www.ontko.com/mic1/

Programming assignments will use this!!!
Block Diagram of Microarchitecture (Mic-1)

Datapath
Part of CPU containing ALU, its inputs, and its outputs

Control Section
Part of CPU containing the H/W necessary to direct the datapath

Microarchitecture Level

Level 2  Instruction set architecture level
          Interpretation (microprogram) or direct execution
Level 1   Microarchitecture level
          Hardware
Level 0   Digital logic level

- Purpose: implement the ISA level above it
- Use digital logic “building blocks” as foundation
Datapath of IJVM

Example ISA: IJVM
- Subset of Java Virtual Machine (JVM)
- Only integer instructions

Example microarchitecture
- 32-bit architecture (4 bytes/word)
- Microprogram (in ROM)
- Loop to fetch, decode, and execute instructions

Datapath Registers

- Memory Address Register
- Memory Data Register
- Program Counter
- Memory Buffer Register
- Stack Pointer
- Local Variable pointer
- Constant Pool Pointer
- Top Of Stack
- OPC
- Holding register
Datapath Registers

- Only accessible at the microarchitecture level by the microprogram
- Maintains the state of the computer
- Usually hold a value corresponding to an ISA variable of the same name

Datapath Control Signals

- **B Bus Enable**
  - Enable one register onto B bus
- **C Bus Enable**
  - Write C bus into register(s)
- **ALU Control**
  - Six signals from Fig 4-2
- **What are N and Z from the ALU?**
  - Status flags used to determine next microinstruction
- **Shifter Control** (p. 207)
  - Shift Left Logical 8
  - Shift Right Arithmetic 1

Can I read and write the same register in one cycle?
Datapath Timing

- Has a finite propagation time
  - Signals travel along wires, through transistors, etc.

- Implicit clock subcycles
  - Set up control signals to drive datapath
  - Register loaded onto B bus
  - ALU and shifter operate
  - Result propagates along C bus to registers

Clocks

- Circuit that emits a series of pulses
  - Precise pulse width
  - Precise interval between consecutive pulses
    - Clock cycle time – time interval between corresponding edges of two consecutive pulses

- Events can be triggered by

<table>
<thead>
<tr>
<th>Rising edge</th>
<th>Falling edge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock high</td>
<td>Clock low</td>
</tr>
</tbody>
</table>

One Cycle
Figure 3-21. (a) A clock. (b) The timing diagram for the clock. (c) Generation of an asymmetric clock.

Timing Diagram for One Datapath Cycle

Cycle 1 starts here
Clock cycle 1
Clock cycle 2

Shifter

Set up
signals
to drive
data path

Δw

Δx

Δy

Δz

ALU

MPC

available
here

MPC

used
to
load
MIU
next
microinstruction
here

New MPC

Registers
loaded
instantaneously
from
C bus
and
memory
on
rising
dege
clock

Drive H
and
B bus

Propagation
from
shifter
to
registers

Set up
signals
to drive
data path

Δw

Δx

Δy

Δz

ALU

MPC

available
here

New MPC

used
to
load
MIU
next
microinstruction
here

Cycle 1

Clock cycle 1

Clock cycle 2
Single Stage Datapath

• **Making the design work requires:**
  – Rigid timing
  – Long clock cycle
  – Known minimum propagation time through ALU
  – Fast load of registers from C Bus

• **ALU runs continuously**
  – Inputs and outputs aren’t valid until right “subcycle”

• **Only explicit signals driving datapath**
  – Falling edge and rising edge of clock

---

Memory Operation

• **Memory Address Register**
• **Memory Data Register**
• **Program Counter**
• **Memory Buffer Register**

• Registers driven by control signals
• Two ways to communicate with memory
  – 32-bit port (word addressable)
    (MAR, MDR)
  – 8-bit read-only port (byte addressable)
    (PC, MBR)
• Actual memory is byte oriented
Mapping the MAR to Address Bus

- Unsigned
  - MBR value in low-order 8 bits
  - Zeros placed in upper 24 bits
  - Used to index into a table
  - Used to assemble a 16-bit integer from 2 consecutive (unsigned) bytes in instruction stream

- Signed
  - Use MBR as a value between -128 and +127
  - Sign extension duplicates leftmost bit in MBR
Microinstructions

- Control the data path

- Five functional groups:
  - 9 control writing data from C bus to registers
  - 9 control driving registers onto B bus
  - 8 control ALU and shifter functions
  - 2 indicate memory read or write (MAR/MDR) (*not shown*)
  - 1 indicates memory fetch (PC/MBR) (*not shown*)

Reduce Control Signals for B Bus

- **B Bus**
  - Connected to all other registers except MAR
  - MBR has two enables for signed and unsigned versions
  - Becomes the right input to the ALU
  - Only want one register enabled

- Use a 4:16 Decoder
Microinstruction Format

- **Use two additional sets of signals**
  - 9 for address of next microinstruction
  - 3 to determine how next microinstruction is selected (JAM)

- **Total of 36 control signals (bits)**

Block Diagram of Microarchitecture (Mic-1)