Topics

• Reminders
  – Homework due this Friday (10-10)
  – Exam 2 on next Friday (10-17)

• Memory

• Buses

• Microarchitecture level

Random Access Memory (RAM)

• Static RAM (SRAM)
  – Constructed from circuits similar to D flip-flop
  – Contents retained while power is on
  – Used in level 2 cache because of fast access times

• Dynamic RAM (DRAM)
  – Constructed with a transistor and capacitor
  – Contents must be “refreshed” due to charge leakage
  – Used in main memory because of high density
Read-Only Memory (ROM)

- **Programmable ROM (PROM)**
  - Programmed once using fuses

- **Erasable PROM (EPROM)**
  - Can be reused after erasing with UV light

- **Electrically Erasable PROM (EEPROM)**
  - Electrical pulses on special input pin erases device
  - Flash memory is a type of EEPROM that is block erasable

Computer Buses

- **Bus** – common electrical pathway between multiple devices
Bus Clocking

- **Synchronous**
  - Uses master clock driven by crystal oscillator
  - All transactions occur as multiples of “bus cycles”
  - Timing diagram defines the bus protocol (Fig 3-37)

- **Asynchronous**
  - Does not require a master clock
  - Transactions can use any length required
  - Full handshake defines the bus protocol (Fig 3-38)

Synchronous Timing
Asynchronous Timing

Example Buses

- **Industry Standard Architecture (ISA)**
  - Found in every Intel-based PC
  - Maximum bandwidth of 16.7 MB/sec

- **Peripheral Component Interconnect (PCI)**
  - Patents in public domain to encourage third-party use
  - Maximum bandwidth of 528 MB/sec

- **Universal Serial Bus (USB)**
  - Industry consortium design for low-speed I/O devices
  - Bus protocol similar to network with data packets contained in frames
  - Maximum bandwidth of 1.5 MB/sec
Registers

- D flip-flops combined to operate as a group
- Forms an 8-bit register (1 byte)
- Use additional chips in parallel for larger registers

Combining Bit Slices for ALU
Microarchitecture Level

- Purpose: implement the ISA level above it
- Design depends on several factors
  - The ISA being implemented (CISC vs. RISC)
  - Cost and performance goals of the computer

CISC Architectures

- S/W interpreter increased the computing power of simple H/W design

- Provided 200 – 300 instructions
  - But how many of those were really used?

- Must still provide support for legacy code
  - Intel processors are CISC/RISC hybrid
RISC Design Principles

- **All instructions directly executed by H/W**
  - Eliminate overhead of interpretation

- **Maximize rate instructions are issued**
  - Utilize parallelism within program

- **Instructions should be easy to decode**
  - Quickly determine resources required

- **Only loads and stores should reference memory**
  - Memory access is longer and unpredictable

- **Provide plenty of registers**
  - Avoid memory access penalty for flushing data set

Datapath of IJVM

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

- **Example microarchitecture**
  - Microprogram (in ROM)
  - Loop to fetch, decode, and execute instructions
Datapath Buses

- **A Bus**
  - Only connected to H register
  - Becomes the left input of the ALU

- **B Bus**
  - Connected to all other registers except MAR
  - Becomes the right input to the ALU
  - Only want one register enabled

- **C Bus**
  - Connected to all registers
  - Writes output to all enabled registers

Datapath Registers (Memory Control)

- **MAR**
  - Memory Address Register
  - Contains word addresses for memory references

- **MDR**
  - Memory Data Register
  - Contains data words to/from memory references

- **PC**
  - Program Counter
  - Points to next instruction to be executed

- **MBR**
  - Memory Buffer Register
  - 8-bit register used to read a single byte from memory
Datapath Registers (Stack)

- **SP**
  - Stack Pointer
  - Points to the highest word of the local variable frame
- **LV**
  - Local Variable pointer
  - Contains the address of the first location in the local variable frame
- **CPP**
  - Constant Pool Pointer
  - Contains the address of the first word of the constant pool
- **TOS**
  - Top Of Stack
  - Contains value on top of memory stack

Datapath Registers (Miscellaneous)

- **OPC**
  - Couldn’t find what it stood for!
  - A temporary (scratch) register without a preassigned use
- **H**
  - Holding register
  - Attached to the left input of the ALU
Datapath Control Signals

- **B Bus Enable**
  - Enable register onto B bus
- **C Bus Enable**
  - Write C bus into register
- **ALU Control**
  - Six signals from Fig 4-2
- **Shifter Control**
  - SLL8 and SRA1

- What are **N** and **Z** from the ALU?
  - Used for conditional tests

Control Signals for ALU Functions

![Control Signals for ALU Functions](image)
Mic-1 Simulator for Class

Java-based simulator which implements the Mic-1 microarchitecture described in Chapter 4

- [http://www.ontko.com/mic1/](http://www.ontko.com/mic1/)

*Programming assignments will use this!!!*