Basic CPU architecture, Busses and Registers

January 4, 2019

---

1 Microprocessor Simulation

- SMS32 is an 8-bit CPU Similar to the x86 family
- Only has 256 bytes of memory
- Some of the features of sms32
  - 8-bit CPU
  - Up to 256 I/O ports (only a few are realistically ever used)

---

2 The CPU

- The brains of the computer
- All calculations and decisions take place here
- Has small pieces of storage called ***registers***
- Has ***Arithmetic and Logic Unit (ALU)*** where calculations are performed
- The ***ALU*** reads information from the registers, calculates and puts the results back into registers
- MOV commands are used to move data between registers in the CPU and memory outside the CPU

---

3 General Purpose Registers

- SMS32 has four general purpose registers
  - AL
  - BL
  - CL
  - DL
- The names come from the real x86 CPU
- Each of these registers is 8-bits(one byte) wide
- Each register can hold:
  - Unsigned numbers from 0 to 255
  - Signed numbers from -128 to +127
  - These registers are temporary storage locations

---

4 Special Registers

- the ***Program Counter (PC)***
  - Also called the ***Instruction Pointer (IP)***
  - Instructions (code) are stored in memory
  - This register tells the CPU where in memory the next instruction is to be taken from
  - When the CPU gets an instruction, the PC is changed to point to the next instruction
  - Some instructions such as CALL and INT can change the PC more drasctically
- The ***Status Register (SR)***
  - This should be viewed as 8 individual bits, or switches
  - Each of these bits has a special meaning (though some are unused)
  - The Z bit (Zero Flag) is set to 1 if the answer for a calculation is zero
  - The S bit (Sign Flag) is set to 1 if the answer for a calculation is negative
  - The O bit (Overflow Flag) is set to 1 if the answer for a signed calculation was too big

- The ***Stack Pointer (SP)***
  - The stack is an area of memory which uses a Last-In-First-Out)(LIFO) rule
  - The SP *points* to the next free location in memory
  - The simulator's stack starts at memory location BF
  - The stack *grows* towards memory location 0
  - Adding data to the stack is called a ***Push***
  - Removing data from the stack is called a ***Pop***
  - Each time a push or pop happens, the SP is updated

---

5 Random Access Memory

- This simulator has only 256 locations in RAM
- Each location can only store one byte
- The locations are numbered from 00 to FF in Hex
- In the assembler, square brackets ([ ]) around a number have a special meaning
  - 45 means the number 45
  - [45] means the number stored at memroy location 45

---

6 Busses

- These are a set of wires used to cary out information around in the computer
- You can see them on the printed circuit board (PCB) as parallel tracks of copper
- For example, between the CPU and memory, there is a set of wires called the Data Bus
- There are three busses we consider:
  - Data Bus
  - Address Bus
  - Control Bus

---

7 The Data Bus

- Used to transfer data between the CPU and memory
- **Usually** related to the size of registers
- A CPU with 8-bit registers will have 8 lines on the data bus
- There are exceptions to this rule
  - The 32-bit Pentium had a 64-bit data bus
- The width of the data bus (number of lines) conrols how much data can be transferred at once between CPU and memory

---

8 The Address Bus

- When the CPU wants to send information to memory, it needs to specify:
  - *What* information to store
  - *Where* the information should be stored
- When the CPU wants to read from memory, it has to specify:
  - *Where* to read from
- The address bus is how the CPU conveys these memory locations
- The address bus differs from the data bus because:
  - Data can be sent in both directions on the data bus (bi-directional)
  - The width of the address bus determines the maximum number of locations the CPU can send out
  - One address line can specify 2^1 addresses
    - 0 or 1
  - Two address lines can specify 2^2 addresses
    - 00, 01, 10, or 11
  - Three address lines can specify 2^3 addresses
    - 000, 001, 010, 011, 100, 101, 110, 111

---

9 The Control Bus

- Has a wire to determine whether to access RAM or IO ports
- Also has a wire to determine whether data is being read or written
- The CPU is reading data when it flows to the CPU
- The CPU is writing data when it flows away from the CPU to RAM or the IO ports
- Has a system clock wire
- Carries regular pulses to allow synchronization of various components
- Clock speeds around 2-4 billion cycles per second are typical

---

10 Hardware Interrupts

- Hardware Interrupts require at least one wire
- These wires are considered part of the control bus
- These enable the CPU to respond to external events
  - For Example, printers running out of paper
- When the interrupt happens
  - The CPU pauses it's current task
  - It runs some machine code in response to the interrupt
  - It then *(Usually)* continues with the original task
