What is in a Processor?

- Arithmetic Logic Unit (ALU)
- Cache
- Registers
  - User registers
  - Program Counter
  - Instruction Register
  - Memory Address Register
  - Memory Data Register
- Internal Bus
- Control Logic

Internal Bus

- The processor has an internal bus that connects all of the CPU components.
- This bus is completely different from the bus that connects the CPU to the memory and I/O controllers.
- Some processors may have multiple internal buses.
- The bus is primarily a set of wires that are charged to hold a value or address.

Register Organization

- A register is a set of flip-flops. There is one flip-flop for each bit.
- The output of a D flip-flop is the same as the input the last time the clock was true.

D

\[ Q = \begin{cases} D & \text{if clock is true} \\ \text{last value of } D & \text{if clock is false} \end{cases} \]

Connecting Register to Bus

- The diagram looks simple, but what is really happening?
- The connection is shown as bidirectional, but the implementation is usually two unidirectional connections

CPU Organization

- The processor has an internal bus that connects all of the CPU components.
- This bus is completely different from the bus that connects the CPU to the memory and I/O controllers.
- Some processors may have multiple internal buses.
- The bus is primarily a set of wires that are charged to hold a value or address.

Register Organization

- A register is a set of flip-flops. There is one flip-flop for each bit.
- The output of a D flip-flop is the same as the input the last time the clock was true.

D

\[ Q = \begin{cases} D & \text{if clock is true} \\ \text{last value of } D & \text{if clock is false} \end{cases} \]
Register Organization
- Below is a 4 bit register.
- When the clock is true, the output of all flip-flops is set to the input value.
- The output is always the last input value.

Switch
- Sometimes you can use an AND gate as a switch
- An AND will NOT work to connect the register to the bus because it will force the bus to false with the toggle is false.

Transistor Switch
- A single transistor can work as a switch.
- When the gate has no voltage, the source and drain are disconnected.

Register Copy
To move a value from register A to register B (which can be any user or system registers).
- Turn on the “register out” switch for register A
- The output of the register A flip-flops sets the value on the bus.
- Turn on the “register in” switch to set the clock input of the register B flip-flops to True.
- The values on the bus (from register A) connect to the input of register B’s flip-flops.
- Turn off “register in” and “register out”.

Fetch Execute Cycle
- Fetch the instruction from memory
- Execute the instruction
**Fetch Execute Cycle** *(more detail)*

1. Fetch the instruction from the memory address in the Program Counter register
2. Increment the Program Counter
3. Decode the type of instruction
4. Fetch the operands
5. Execute the instruction
6. Store the results

**Fetch Execute Cycle** *(even more)*

1. Fetch the instruction from the memory address in the Program Counter register
   - Copy the program counter to the Memory Address Register
   - Tell the memory system to read.
   - Wait for the read to complete
   - Copy from the Memory Data Register to the Instruction Register

2. Increment the Program Counter
   - Copy the Program Counter to the ALU input register.
   - Set the ALU function to increment.
   - Copy the ALU output register to the Program Counter

4. Fetch the operands
   *Assuming direct addressing*
   - Copy the address portion of the instruction register to the Memory Address Register.
   - Tell the memory system to read.
   - Wait for the read to complete
   - Copy from the Memory Data Register to the appropriate data or ALU Register

5. Execute the instruction
   *Assume an arithmetic instruction*
   - Copy the operand from a data register to an ALU input register.
   - Set the ALU function according to the opcode field of the instruction register

6. Store the results
   - Copy the output ALU register to the appropriate data register.
Layers

- Applications
- Middleware
- High level languages
- Machine Language
- Microcode
- Logic circuits
- Gates
- Transistors
- Silicon structures