Lecture No 7

Basic Instruction Timing

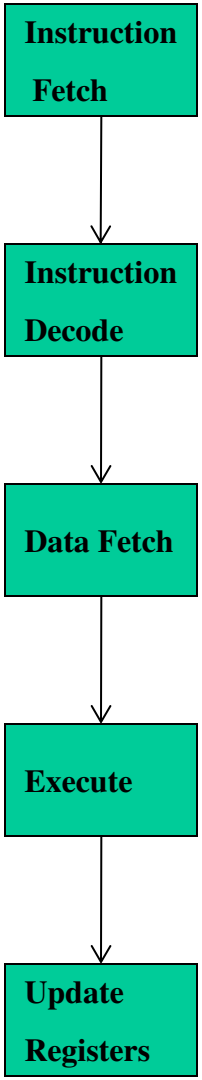
Basic Instruction Timing

A simple machine normally consists of following functional units.

- **Cache**
- **Memory**
- **ALU**
- **Address Generation Unit**
- **TLB**
- **Instruction decoder**

These units are accessed or employed for execution of an instruction . Access to different units occupies one or more cycles. The sequence of events happening in execution of an instruction will determine access to these units and addition of all the cycles required will give the time required to execute the particular instruction.

Basic Instruction Timing



Basic Instruction Timing

The Process of instruction execution for simple machines that executes instructions serially (called well mapped machines), consists of following events and sub events.

- **Instruction Fetch**
  - **Generate real address from value stored in PC to access the instruction.**
  - **Access the cache**
  - **Access Memory if *cache miss* occurs**
  - **Move the word (instruction) fetched from cache / memory (Available in SR Register) to the IR (Instruction Register).**
- **Instruction Decode**
  - **Determine instruction type and addressing mode**
  - **Fetch register operands**

Basic Instruction Timing

- **Data Fetch**
  - **Generate real address for data (Offset +Base / Index)**
  - **Access the cache**
  - **Access Memory if cache miss occurs.**
- **Execute**
  - **Use ALU to perform required operation on data. (Available in SR and other Registers)**
- **Update Registers**
  - **Adjust PC to point to next instruction**
  - **Store results of ALU operation in registers.**

Many other events (like Page fault) also might happen. Which would further prolong the execution of instructions. Details such as setting up of condition codes, memory bound checking etc are not mentioned to keep things basic and simple.