Lecture No 2

The Machine: Interpretation & Microprogramming
Management of *Interpretation* Process is responsibility of decoder. The *Interpretation Process* begins with the decoding of opcode field from the *Instruction*.

OP Code field being decoded by the *Decoder*.

<table>
<thead>
<tr>
<th>OP CODE</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
</table>

The *Instruction*

The *Decoder* activates *Registers* for a series of state transitions that correspond to the action of OP Code.
The Registers used in Instructions can be both

*Explicit and Implicit.*

**Explicit Registers Include:**

- General Purpose Registers (GPR)
- Accumulators (ACC)
- Address Registers (Index or Base Registers)
The Machine: Interpretation & Microprogramming

The * Registers used in *Instructions can be both

*Explicit* and *Implicit*.

**Implicit Registers Include:**

- **PC (Program Counter):**
  Contains address of next instruction in sequence.

- **Instruction Register:**
  This register holds the Instruction being interpreted or executed.

- **Memory Address Register (MAR):**
  Contents of this register are used as address to locate information in the memory.
Decoder controls the *Data Paths* consisting of combinational logic.

**What is Data Path?**
“Data path connect the output of one register to another register.”

Each OP Code defines which of the various *data paths* will be used in its *Execution*.

![Diagram of Data Path](image)
Decoder that manage interpretation is Direct or Microprogrammed Decoder

1. **Direct Decoder:**

Direct Decoders are designed using combinational logic to represent the various desired control point actions.

![Diagram](image)
Decoder that manage interpretation is Direct or Microprogrammed Decoder

2. **Microprogrammed Decoder**: are designed using ROM. The OP Code provides an initial address to an entry which specifies the control point values as well as the address of the next micro instruction.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Direct Decoders</th>
<th>Micro Programmed Decoder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>Fast</td>
<td>Slower</td>
</tr>
<tr>
<td>Chip Area Efficiency</td>
<td>Uses Least area</td>
<td>Uses More Area</td>
</tr>
<tr>
<td>Ease Of Change</td>
<td>Somewhat Difficult</td>
<td>Easier</td>
</tr>
<tr>
<td>Large/Complex Instruction Sets</td>
<td>Somewhat Difficult</td>
<td>Easier</td>
</tr>
<tr>
<td>Support of Operating Systems and Diagnostic Features</td>
<td>Very Difficult</td>
<td>Easy</td>
</tr>
<tr>
<td>Where Used</td>
<td>Mostly RISC M/C</td>
<td>Main Frames / Microprocessors</td>
</tr>
<tr>
<td>Instruction set size</td>
<td>Usually under 100</td>
<td>Usually over 100</td>
</tr>
</tbody>
</table>