# Computer Architecture

Spring 2020

**Hamed Farbeh**

**farbeh@aut.ac.ir**

Department of Computer Engineering

Amirkabir University of Technology

## BASIC COMPUTER ORGANIZATION AND DESIGN

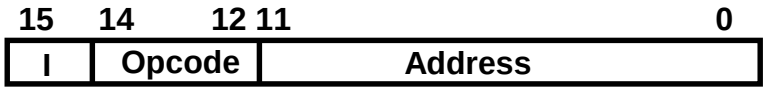
# Outlines

- Instruction Codes
- Computer Registers
- **Computer Instructions**
- **Timing and Control**
- **Instruction Cycle**
- **Memory Reference Instructions**
- **Input-Output and Interrupt**
- **Complete Computer Description**
- **Design of Basic Computer**
- **Design of Accumulator Logic**

# BASIC COMPUTER INSTRUCTIONS

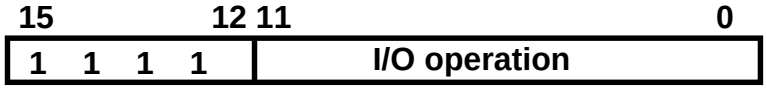
- **Basic Computer Instruction Format**

**Memory-Reference Instructions** **(OP-code = 000 ~ 110)**




**Register-Reference Instructions** **(OP-code = 111, I = 0)**


**Input-Output Instructions** **(OP-code =111, I = 1)**



# BASIC COMPUTER INSTRUCTIONS

| Symbol | Hex Code I = 0 | Hex Code I = 1 | Description |
|---|---|---|---|
| AND | 0xxx | 8xxx | AND memory word to AC |
| ADD | 1xxx | 9xxx | Add memory word to AC |
| LDA | 2xxx | Axxx | Load AC from memory |
| STA | 3xxx | Bxxx | Store content of AC into memory |
| BUN | 4xxx | Cxxx | Branch unconditionally |
| BSA | 5xxx | Dxxx | Branch and save return address |
| ISZ | 6xxx | Exxx | Increment and skip if zero |
| CLA | 7800 | | Clear AC |
| CLE | 7400 | | Clear E |
| CMA | 7200 | | Complement AC |
| CME | 7100 | | Complement E |
| CIR | 7080 | | Circulate right AC and E |
| CIL | 7040 | | Circulate left AC and E |
| INC | 7020 | | Increment AC |
| SPA | 7010 | | Skip next instr. if AC is positive |
| SNA | 7008 | | Skip next instr. if AC is negative |
| SZA | 7004 | | Skip next instr. if AC is zero |
| SZE | 7002 | | Skip next instr. if E is zero |
| HLT | 7001 | | Halt computer |
| INP | F800 | | Input character to AC |
| OUT | F400 | | Output character from AC |
| SKI | F200 | | Skip on input flag |
| SKO | F100 | | Skip on output flag |
| ION | F080 | | Interrupt on |
| IOF | F040 | | Interrupt off |

# INSTRUCTION SET COMPLETENESS

**A computer should have a set of instructions so that the user can construct machine language programs to evaluate any function that is known to be computable.**

- **Instruction Types**

  **Functional Instructions**
  - **Arithmetic, logic, and shift instructions**
  - **ADD, CMA, INC, CIR, CIL, AND, CLA**

  **Transfer Instructions**
  - **Data transfers between the main memory and the processor registers**
  - **LDA, STA**

  **Control Instructions**
  - **Program sequencing and control**
  - **BUN, BSA, ISZ**

  **Input/Output Instructions**
  - **Input and output**
  - **INP, OUT**

to be continued