MICROPROCESSORS

tutorialspoint
SIMPLY EASY LEARNING

About the Tutorial

A microprocessor is a controlling unit of a micro-computer, fabricated on a small chip capable of performing Arithmetic Logical Unit (ALU) operations and communicating with the other devices connected to it.

In this tutorial, we will discuss the architecture, pin diagram and other key concepts of microprocessors.

Audience

This tutorial is designed for all those readers pursing either Bachelor's or Master's degree in Computer Science. It will help them understand the basic concepts related to Microprocessors.

Prerequisites

In this tutorial, all the topics have been explained from elementary level. Therefore, a beginner can understand this tutorial very easily. However if you have a prior knowledge of computer architecture in general, then it will be quite easy to grasp the concepts explained here.

Disclaimer & Copyright

Table of Contents

Microprocessor

Microprocessor is a controlling unit of a micro-computer, fabricated on a small chip capable of performing ALU (Arithmetic Logical Unit) operations and communicating with the other devices connected to it.

Microprocessor consists of an ALU, register array, and a control unit. ALU performs arithmetical and logical operations on the data received from the memory or an input device. Register array consists of registers identified by letters like B, C, D, E, H, L and accumulator. The control unit controls the flow of data and instructions within the computer.

Block Diagram of a Basic Microcomputer
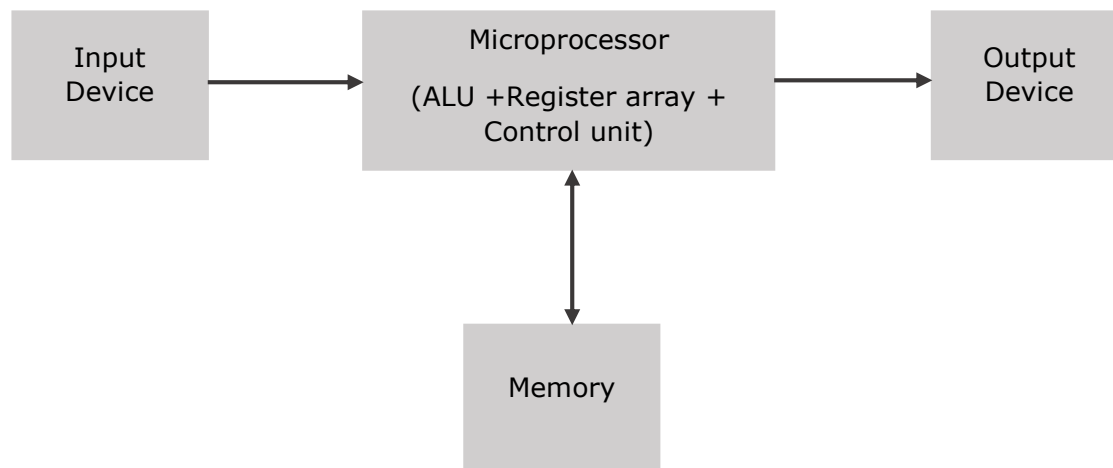
How does a Microprocessor Work?

The microprocessor follows a sequence: Fetch, Decode, and then Execute.

Initially, the instructions are stored in the memory in a sequential order. The microprocessor fetches those instructions from the memory, then decodes it and executes those instructions till STOP instruction is reached. Later, it sends the result in binary to the output port. Between these processes, the register stores the temporarily data and ALU performs the computing functions.

List of Terms Used in a Microprocessor

Here is a list of some of the frequently used terms in a microprocessor:

- **Instruction Set**: It is the set of instructions that the microprocessor can understand.

- **Bandwidth**: It is the number of bits processed in a single instruction.

- **Clock Speed**: It determines the number of operations per second the processor can perform. It is expressed in megahertz (MHz) or gigahertz (GHz). It is also known as Clock Rate.

- **Word Length**: It depends upon the width of internal data bus, registers, ALU, etc. An 8-bit microprocessor can process 8-bit data at a time. The word length ranges from 4 bits to 64 bits depending upon the type of the microcomputer.

- **Data Types**: The microprocessor has multiple data type formats like binary, BCD, ASCII, signed and unsigned numbers.

Features of a Microprocessor

Here is a list of some of the most prominent features of any microprocessor:

- **Cost-effective**: The microprocessor chips are available at low prices and results its low cost.

- **Size**: The microprocessor is of small size chip, hence is portable.

- **Low Power Consumption**: Microprocessors are manufactured by using metal-oxide semiconductor technology, which has low power consumption.

- **Versatility**: The microprocessors are versatile as we can use the same chip in a number of applications by configuring the software program.

- **Reliability**: The failure rate of an IC in microprocessors is very low, hence it is reliable.

A microprocessor can be classified into three categories:
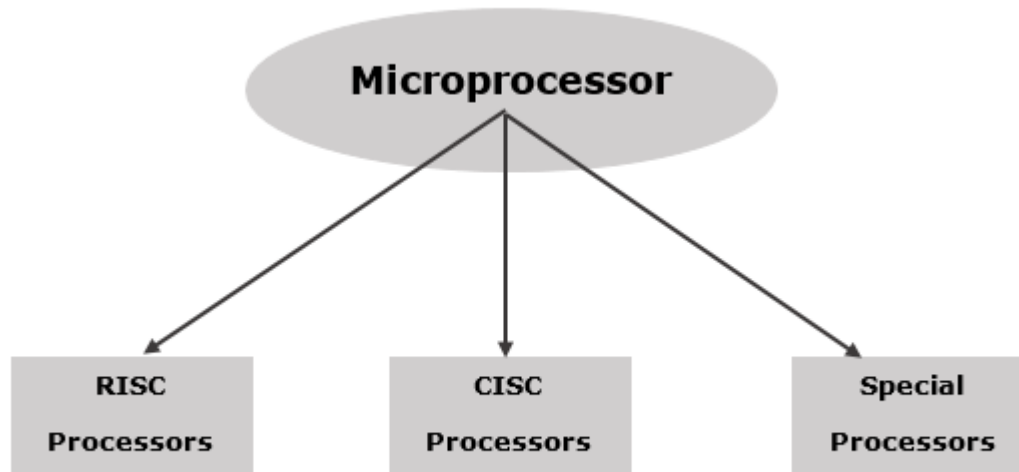
RISC Processor

RISC stands for **Reduced Instruction Set Computer**. It is designed to reduce the execution time by simplifying the instruction set of the computer. Using RISC processors, each instruction requires only one clock cycle to execute results in uniform execution time. This reduces the efficiency as there are more lines of code, hence more RAM is needed to store the instructions. The compiler also has to work more to convert high-level language instructions into machine code.

Some of the RISC processors are:

- Power PC: 601, 604, 615, 620
- DEC Alpha: 210642, 211066, 21068, 21164
- MIPS: TS (R10000) RISC Processor
- PA-RISC: HP 7100LC

Architecture of RISC

RISC microprocessor architecture uses highly-optimized set of instructions. It is used in portable devices like Apple iPod due to its power efficiency.

```
┌─────────────────────────────────────────────────────┐
│                                                       │
│   ┌──────────────┐              ┌──────────────┐      │
│   │  Hardwired   │◄────────────►│  Data Path   │      │
│   │ Control Unit │              │              │      │
│   └──────┬───────┘              └──────┬───────┘      │
│          │                             │              │
│          ▼                             ▼              │
│   ┌──────────────┐              ┌──────────────┐      │
│   │ Instruction  │              │  Data cache  │      │
│   │    cache     │              │              │      │
│   └──────┬───────┘              └──────┬───────┘      │
│          │                             │              │
│          ▼                             ▼              │
│   ┌─────────────────────────────────────────────┐    │
│   │              (Instruction)                   │    │
│   │               + (Data)                       │    │
│   │                                              │    │
│   │              Main memory                     │    │
│   └─────────────────────────────────────────────┘    │
│                                                       │
└─────────────────────────────────────────────────────┘
```
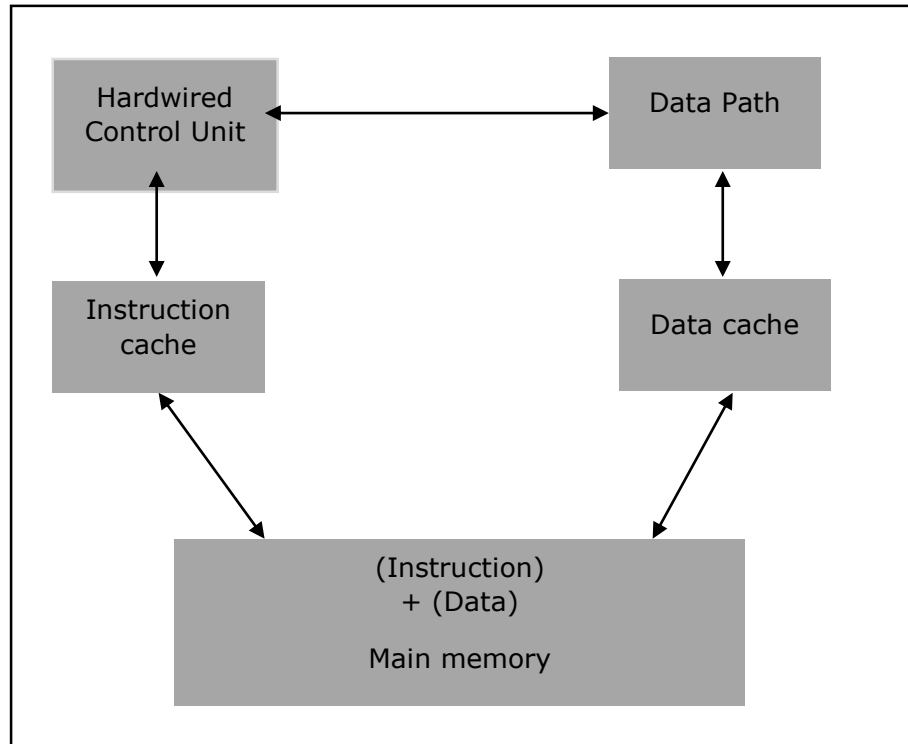
Characteristics of RISC

The major characteristics of a RISC processor are as follows:

- It consists of simple instructions.
- It supports various data-type formats.
- It utilizes simple addressing modes and fixed length instructions for pipelining.
- It supports register to use in any context.
- One cycle execution time.
- "LOAD" and "STORE" instructions are used to access the memory location.
- It consists of larger number of registers.
- It consists of less number of transistors.

9

tutorialspoint
SIMPLYEASYLEARNING

CISC Processor

CISC stands for **Complex Instruction Set Computer**. It is designed to minimize the number of instructions per program, ignoring the number of cycles per instruction. The emphasis is on building complex instructions directly into the hardware.

The compiler has to do very little work to translate a high-level language into assembly level language/machine code because the length of the code is relatively short, so very little RAM is required to store the instructions.

Some of the CISC Processors are:

- IBM 370/168
- VAX 11/780
- Intel 80486

Architecture of CISC

Its architecture is designed to decrease the memory cost because more storage is needed in larger programs resulting in higher memory cost. To resolve this, the number of instructions per program can be reduced by embedding the number of operations in a single instruction.
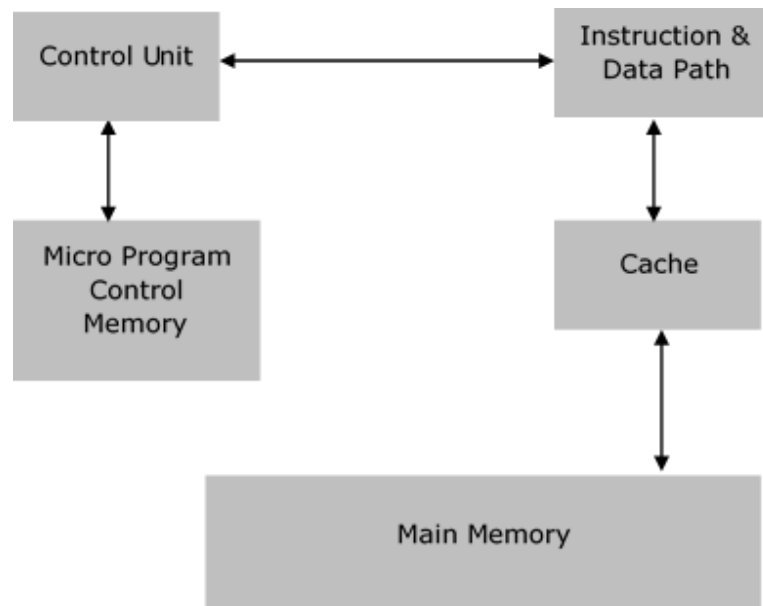
Characteristics of CISC

- Variety of addressing modes.
- Larger number of instructions.

- Variable length of instruction formats.

- Several cycles may be required to execute one instruction.

- Instruction-decoding logic is complex.

- One instruction is required to support multiple addressing modes.

Special Processors

These are the processors which are designed for some special purposes. Few of the special processors are briefly discussed:

Coprocessor

A coprocessor is a specially designed microprocessor, which can handle its particular function many times faster than the ordinary microprocessor.

For example: Math Coprocessor.

Some Intel math-coprocessors are:

- 8087-used with 8086

- 80287-used with 80286

- 80387-used with 80386

Input/Output Processor

It is a specially designed microprocessor having a local memory of its own, which is used to control I/O devices with minimum CPU involvement.

For example:

- DMA (direct Memory Access) controller

- Keyboard/mouse controller

- Graphic display controller

- SCSI port controller

Transputer (Transistor Computer)

A transputer is a specially designed microprocessor with its own local memory and having links to connect one transputer to another transputer for inter-processor communications. It was first designed in 1980 by Inmos and is targeted to the utilization of VLSI technology.

A transputer can be used as a single processor system or can be connected to external links, which reduces the construction cost and increases the performance.

11

For example:16-bit T212, 32-bit T425, the floating point (T800, T805 & T9000) processors.

DSP (Digital Signal Processor)

This processor is specially designed to process the analog signals into a digital form. This is done by sampling the voltage level at regular time intervals and converting the voltage at that instant into a digital form. This process is performed by a circuit called an analogue to digital converter, A to D converter or ADC.

A DSP contains the following components:

- **Program Memory**: It stores the programs that DSP will use to process data.

- **Data Memory**: It stores the information to be processed.

- **Compute Engine**: It performs the mathematical processing, accessing the program from the program memory and the data from the data memory.

- **Input/Output**: It connects to the outside world.

Its applications are:

- Sound and music synthesis
- Audio and video compression
- Video signal processing
- 2D and 3d graphics acceleration.

For example: Texas Instrument's TMS 320 series, e.g., TMS 320C40, TMS320C50.

8085 Microprocessor

8085 is pronounced as "eighty-eighty-five" microprocessor. It is an 8-bit microprocessor designed by Intel in 1977 using NMOS technology.

It has the following configuration:

- 8-bit data bus
- 16-bit address bus, which can address upto 64KB
- A 16-bit program counter
- A 16-bit stack pointer
- Six 8-bit registers arranged in pairs: BC, DE, HL
- Requires +5V supply to operate at 3.2 MHZ single phase clock

It is used in washing machines, microwave ovens, mobile phones, etc.

8085 Microprocessor – Functional Units

8085 consists of the following functional units:

Accumulator

It is an 8-bit register used to perform arithmetic, logical, I/O & LOAD/STORE operations. It is connected to internal data bus & ALU.

Arithmetic and logic unit

As the name suggests, it performs arithmetic and logical operations like Addition, Subtraction, AND, OR, etc. on 8-bit data.

General purpose register

There are 6 general purpose registers in 8085 processor, i.e. B, C, D, E, H & L. Each register can hold 8-bit data.
These registers can work in pair to hold 16-bit data and their pairing combination is like B-C, D-E & H-L.

Program counter

It is a 16-bit register used to store the memory address location of the next instruction to be executed. Microprocessor increments the program whenever an instruction is being executed, so that the program counter points to the memory address of the next instruction that is going to be executed.

Stack pointer

It is also a 16-bit register works like stack, which is always incremented/decremented by 2 during push & pop operations.

Temporary register

It is an 8-bit register, which holds the temporary data of arithmetic and logical operations.

Flag register

It is an 8-bit register having five 1-bit flip-flops, which holds either 0 or 1 depending upon the result stored in the accumulator.

These are the set of 5 flip-flops:

- Sign (S)
- Zero (Z)
- Auxiliary Carry (AC)
- Parity (P)
- Carry (C)

Its bit position is shown in the following diagram:

D7	D6	D5	D4	D3	D2	D1	D0
S	Z		AC		P		CY

Instruction register and decoder

It is an 8-bit register. When an instruction is fetched from memory then it is stored in the Instruction register. Instruction decoder decodes the information present in the Instruction register.

Timing and control unit

It provides timing and control signal to the microprocessor to perform operations. Following are the timing and control signals, which control external and internal circuits:

- Control Signals: READY, RD', WR', ALE
- Status Signals: S0, S1, IO/M'
- DMA Signals: HOLD, HLDA
- RESET Signals: RESET IN, RESET OUT

Interrupt control

As the name suggests it controls the interrupts during a process. When a microprocessor is executing a main program and whenever an interrupt occurs, the microprocessor shifts the control from the main program to process the incoming request. After the request is completed, the control goes back to the main program.

There are 5 interrupt signals in 8085 microprocessor: INTR, RST 7.5, RST 6.5, RST 5.5, TRAP.

Serial Input/output control

It controls the serial data communication by using these two instructions: SID (Serial input data) and SOD (Serial output data).

Address buffer and address-data buffer

The content stored in the stack pointer and program counter is loaded into the address buffer and address-data buffer to communicate with the CPU. The memory and I/O chips are connected to these buses; the CPU can exchange the desired data with the memory and I/O chips.

Address bus and data bus

Data bus carries the data to be stored. It is bidirectional, whereas address bus carries the location to where it should be stored and it is unidirectional. It is used to transfer the data & Address I/O devices.
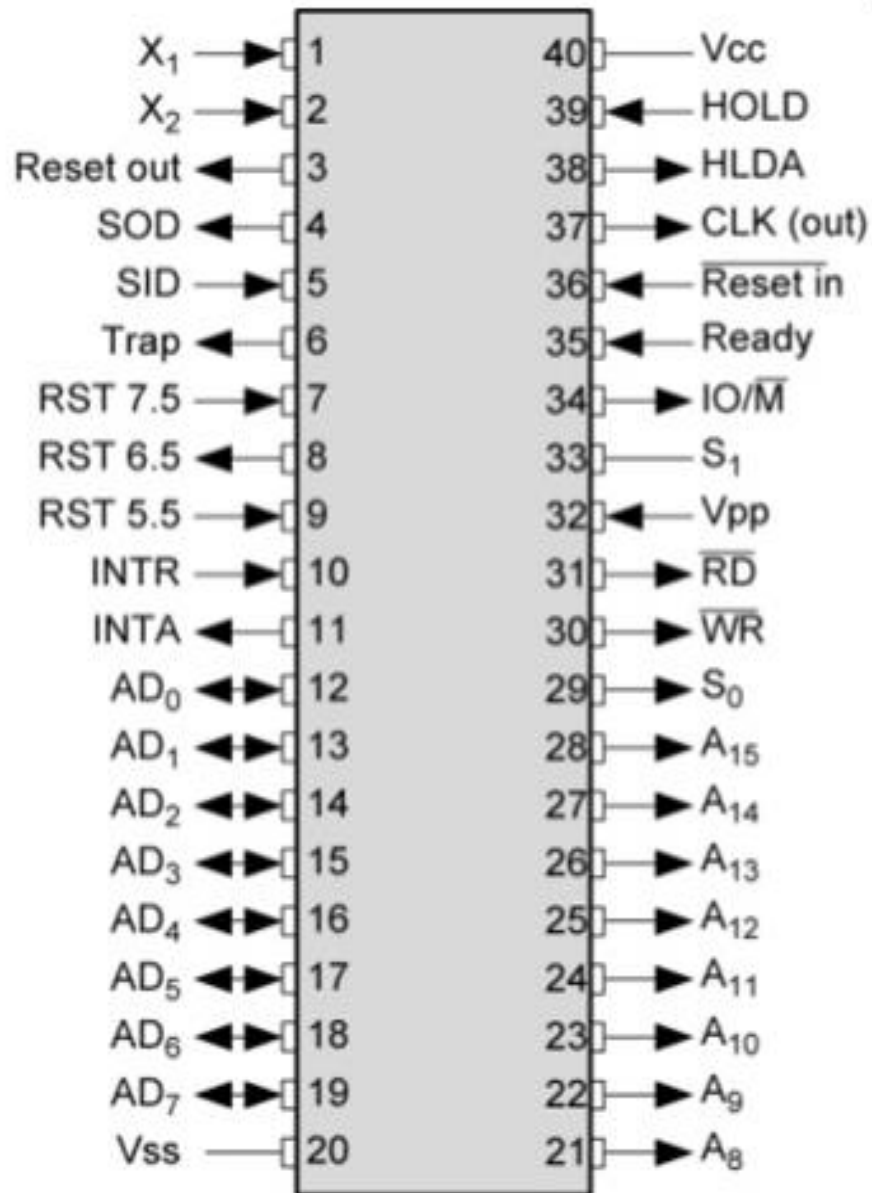
8085 Architecture

We have tried to depict the architecture of 8085 with this following image:
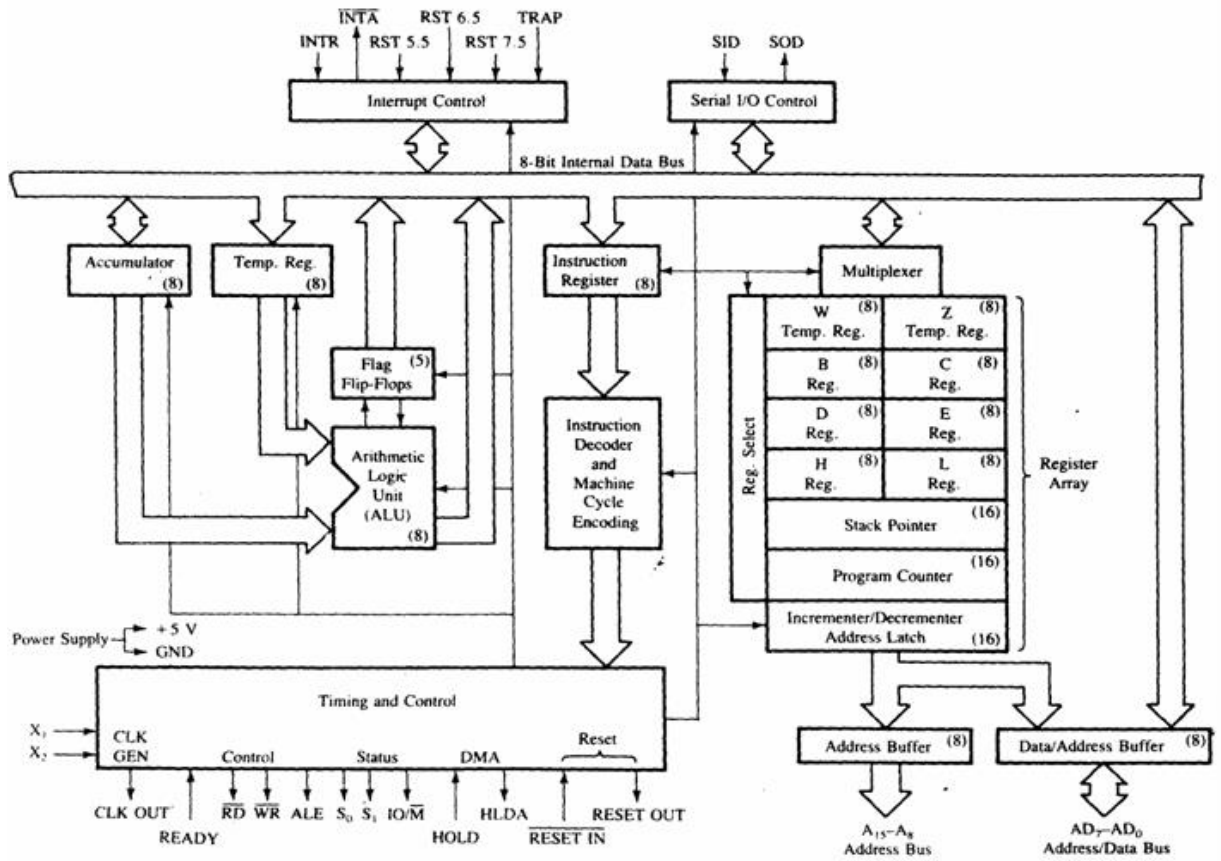
The following image depicts the pin diagram of 8085 Microprocessor:

The pins of a 8085 microprocessor can be classified into seven groups:

Address bus

A15-A8, it carries the most significant 8-bits of memory/IO address.

Data bus

AD7-AD0, it carries the least significant 8-bit address and data bus.

Control and status signals

These signals are used to identify the nature of operation. There are 3 control signal and 3 status signals.

Three control signals are RD, WR & ALE.

- **RD**: This signal indicates that the selected IO or memory device is to be read and is ready for accepting data available on the data bus.

- **WR**: This signal indicates that the data on the data bus is to be written into a selected memory or IO location.

- **ALE**: It is a positive going pulse generated when a new operation is started by the microprocessor. When the pulse goes high, it indicates address. When the pulse goes down it indicates data.

Three status signals are IO/M, S0 & S1.

IO/M

This signal is used to differentiate between IO and Memory operations, i.e. when it is high indicates IO operation and when it is low then it indicates memory operation.

S1 & S0

These signals are used to identify the type of current operation.

Power supply

There are 2 power supply signals: VCC & VSS. VCC indicates +5v power supply and VSS indicates ground signal.

Clock signals

There are 3 clock signals, i.e. X1, X2, CLK OUT.

- **X1, X2**: A crystal (RC, LC N/W) is connected at these two pins and is used to set frequency of the internal clock generator. This frequency is internally divided by 2.

- **CLK OUT**: This signal is used as the system clock for devices connected with the microprocessor.

Interrupts & externally initiated signals

Interrupts are the signals generated by external devices to request the microprocessor to perform a task. There are 5 interrupt signals, i.e. TRAP, RST 7.5, RST 6.5, RST 5.5, and INTR. We will discuss interrupts in detail in interrupts section.

- **INTA**: It is an interrupt acknowledgment signal.

- **RESET IN**: This signal is used to reset the microprocessor by setting the program counter to zero.
- **RESET OUT**: This signal is used to reset all the connected devices when the microprocessor is reset.

- **READY**: This signal indicates that the device is ready to send or receive data. If READY is low, then the CPU has to wait for READY to go high.

- **HOLD**: This signal indicates that another master is requesting the use of the address and data buses.

- **HLDA** (HOLD Acknowledge): It indicates that the CPU has received the HOLD request and it will relinquish the bus in the next clock cycle. HLDA is set to low after the HOLD signal is removed.

Serial I/O signals

There are 2 serial signals, i.e. SID and SOD and these signals are used for serial communication.

- **SOD** (Serial output data line): The output SOD is set/reset as specified by the SIM instruction.

- **SID** (Serial input data line): The data on this line is loaded into accumulator whenever a RIM instruction is executed.

End of ebook preview
If you liked what you saw…
Buy it from our store @ **https://store.tutorialspoint.com**