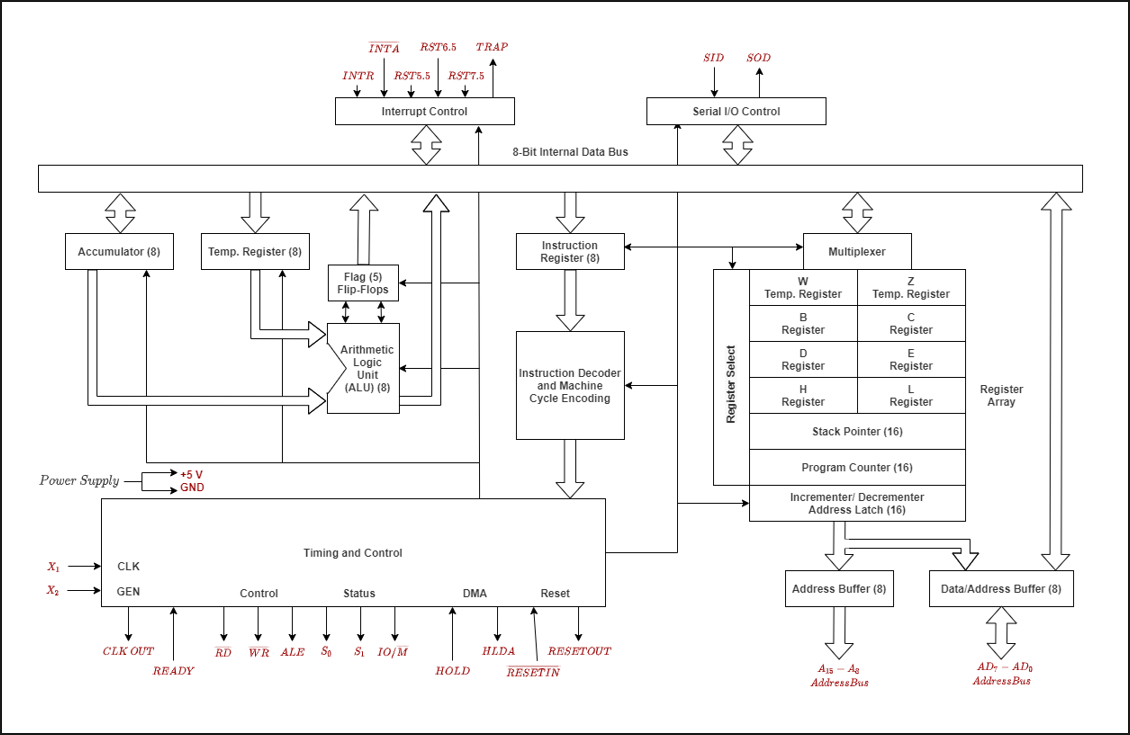
**Practical-6**

**Aim: Study of 8085 Microprocessor Architecture. (Including types of instruction, addressing modes, flag register)**

**What is the 8085 Microprocessor?**

* The 8085 is an 8-bit microprocessor, and it was launched by the Intel team in the year of 1976 with the help of NMOS technology.
* This processor is the updated version of the microprocessor. The configurations of 8085 microprocessor mainly include data bus-8-bit, address bus-16 bit, program counter-16-bit, stack pointer-16 bit, registers 8-bit, +5V voltage supply, and operates at 3.2 MHz single segment CLK.
* The applications of 8085 microprocessor are involved in microwave ovens, washing machines, gadgets, etc.

**Architecture:**

****

**Types of instruction:**

1. **Control Instructions**: These instructions are mainly used to control the microprocessor operations.

|  |  |
| --- | --- |
| Opcode | Meaning |
| NOP | No operation |
| HLT | Halt and enter wait state |
| DI | Disable interrupts |
| EI | Enable interrupts |
| RIM | Read interrupt mask |
| SIM | Set interrupt mask |

1. **Logical Instructions:** Logical instructions are mainly used to perform different operations like logical or Boolean over the data available in either memory or register. These instructions will modify the flag bits based on the operation executed.

|  |  |
| --- | --- |
| Opcode | Meaning |
| CMP | Compare the register or memory with the accumulator |
| CPI | Compare immediate with the accumulator |
| ANA | Logical AND register or memory with the accumulator |
| ANI | Logical AND immediate with the accumulator |
| XRA | Exclusive OR register or memory with the accumulator |
| XRI | Exclusive OR immediate with the accumulator |
| ORA | Logical OR register or memory with the accumulator |
| ORI | Logical OR immediate with the accumulator |
| RLC | Rotate the accumulator left |
| RRC | Rotate the accumulator right |
| RAL | Rotate the accumulator left through carry |
| RAR | Rotate the accumulator right through carry |
| CMA | Complement accumulator |
| CMC | Complement carry |
| STC | Set Carry |

1. **Branching Instructions:** These types of instructions are mainly used to transfer or switch the microprocessor from one location to another. So, it simply changes the general sequential flow.

|  |  |
| --- | --- |
| Opcode | Meaning |
| JMP | Jump unconditionally |
| |  |  |  | | --- | --- | --- | | Opcode | Description | Flag Status | | JC | Jump on Carry | CY=1 | | JNC | Jump on no Carry | CY=0 | | JP | Jump on positive | S=0 | | JM | Jump on minus | S=1 | | JZ | Jump on zero | Z=1 | | JNZ | Jump on no zero | Z=0 | | JPE | Jump on parity even | P=1 | | JPO | Jump on parity odd | P=0 | | Jump conditionally |
| |  |  |  | | --- | --- | --- | | Opcode | Description | Flag Status | | CC | Call on Carry | CY=1 | | CNC | Call on no Carry | CY=0 | | CP | Call on positive | S=0 | | CM | Call on minus | S=1 | | CZ | Call on zero | Z=1 | | CNZ | Call on no zero | Z=0 | | CPE | Call on parity even | P=1 | | CPO | Call on parity odd | P=0 | | Unconditional subroutine call |
| RET | Return from subroutine unconditionally |
| |  |  |  | | --- | --- | --- | | Opcode | Description | Flag Status | | RC | Return on Carry | CY=1 | | RNC | Return on no Carry | CY=0 | | RP | Return on positive | S=0 | | RM | Return on minus | S=1 | | RZ | Return on zero | Z=1 | | RNZ | Return on no zero | Z=0 | | RPE | Return on parity even | P=1 | | RPO | Return on parity odd | P=0 | | Return from subroutine conditionally |
| PCHL | Load the program counter with HL contents |
| RST | Restart |

1. **Arithmetic Instructions:** The arithmetic instructions perform different operations like addition, subtraction, increment & decrement on the data within memory & register in the 8085 microprocessor.

|  |  |
| --- | --- |
| Opcode | Meaning |
| ADD | Add register or memory, to the accumulator |
| ADC | Add register to the accumulator with carry |
| ADI | Add the immediate to the accumulator |
| ACI | Add the immediate to the accumulator with carry |
| LXI | Load the register pair immediate |
| DAD | Add the register pair to H and L registers |
| SUB | Subtract the register or the memory from the accumulator |
| SBB | Subtract the source and borrow from the accumulator |
| SUI | Subtract the immediate from the accumulator |
| XCHG | Exchange H and L with D and E |
| INR | Increment the register or the memory by 1 |
| INX | Increment register pair by 1 |
| DCR | Decrement the register or the memory by 1 |
| DCX | Decrement the register pair by 1 |
| DAA | Decimal adjust accumulator |

1. **Data-transfer Instructions**: An instruction that is used to transfer the data from one register to another is known as data transfer instruction. So, the data transfer can be done from source to destination without changing the source contents. Data transfer mainly occurs from one register to another register, from memory location to register, register to memory, and between an I/O device & accumulator.

|  |  |
| --- | --- |
| Opcode | Meaning |
| MOV | Copy from the source (Sc) to the destination(Dt) |
| MVI | Move immediate 8-bit |
| LDA | Load the accumulator |
| LDAX | Load the accumulator indirect |
| LXI | Load the register pair immediate |
| LHLD | Load H and L registers direct |
| STA | 16-bit address |
| STAX | Store the accumulator indirect |
| SHLD | Store H and L registers direct |
| XCHG | Exchange H and L with D and E |
| SPHL | Copy H and L registers to the stack pointer |
| XTHL | Exchange H and L with top of stack |
| PUSH | Push the register pair onto the stack |
| POP | Pop off stack to the register pair |
| OUT | Output the data from the accumulator to a port with 8bit address |
| IN | Input data to accumulator from a port with 8-bit address |

**Addressing Mode**

Each instruction requires some data on which it has to operate. There are different techniques to specify data for instructions. These techniques are called addressing modes.

**Direct Addressing:** In this addressing mode, the address of the operand (data) is given in the instruction itself.

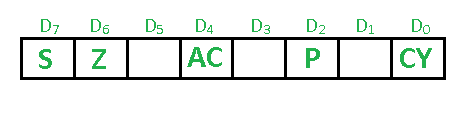
**Register Addressing:** In register addressing mode, the operand is in one of the general purpose registers. The opcode specifies the address of the register(s) in addition to the operation to be performed.

**Register Indirect Addressing:** In Register Indirect mode of addressing, the address of the operand is specified by a register pair.

**Immediate Addressing**: In this addressing mode, the operand is specified within the instruction itself.

**Implicit Addressing:** There are certain instructions which operate on the content of the accumulator. Such instructions do not require the address of the operand.

**Flag Register:** The Flag register is a Special Purpose Register. Depending upon the value of the result after any arithmetic and logical operation, the flag bits become set (1) or reset (0). In 8085 microprocessor, the flag register consists of 8 bits and only 5 of them are useful.



**Sign flag (S):-** The Sign flag is set if D7 bit is negative otherwise 0 reset.

**Zero flag (Z):-** The Zero flag is set if result of ALU operation is zero. Otherwise reset.

**Auxiliary Carry (AC):-** This flag is set when a carry is generated from bit D3 and passed to D4.

**Parity flag (P):-** After an ALU operation is the result has an even number of 1’s the Parity flag is set. Otherwise reset.

**Carry flag (CY):-** Carry flag is set when result generates a carry. Otherwise reset.