CHECK YOUR PROGRESS

4. Fill the blanks-

(i) The routine executed in response to an interrupt request is known as _____.

(ii) Methods for handling multiple interrupts are _____ and _____.

(iii) The module that can handle the interrupt requests from various devices and allow one by one to the processor and is known as _____.

(iv) The processing required for executing a single instruction is known as _____.

(v) Reading the instruction from the memory to Instruction register, called _____, and executing the instruction from the instruction register, called _____.

4.9 HARDWIRED AND MICRO PROGRAMMED CONTROL

We know that the execution of a program is nothing but the execution of a sequence of instruction cycles, with one machine instruction per cycle. Again, each instruction cycle is made up of some small cycles and those are fetch cycle, indirect cycle, execute cycle and interrupt cycle, with only fetch and execute cycles always occurring. Now, each of the smaller cycles i.e. fetch cycle, indirect cycle, execute cycle and interrupt cycle, involves a series of steps, each of which involves the processor registers and these steps are called Micro-Operations. Therefore, a micro-operation is an elementary CPU operation, performed during one clock pulse and an instruction consists of a sequence of micro-operations.

Now, we will try to see the instruction cycle as a sequence of Micro-Operations. For that, first we have to take the fetch cycle, a sub-cycle of the instruction cycle. In the fetch cycle, the Micro-Operations are as follows-

µOp1- Move the content of Program Counter to Memory Address Register.
i.e. MAR ← (PC)

µOp2- Move the content of memory location specified by Memory Address register to Memory Buffer Register.
i.e., MBR ← Memory

µOp3- Increment the Program Counter by the length of instruction.
i.e., PC ← PC + *I*, where *I* is the instruction size.

µOp4- Move the content of the Memory Buffer Register to the Instruction Register.
i.e., IR ← (MBR)

Therefore, it is found that four Micro-Operations are needed for the fetch sub-cycle. Here, µOp1 will be done in time T_1, µOp2 and µOp3 will be done in time T_2 and µOp4 will be done in T_3. Or, µOp1 will be done in time T_1, µOp2 will be done in time T_2 and µOp3 and µOp4 will be done in T_3.

T1:	MAR←(PC)
T2:	MBR ← Memory PC ← (PC) + I
T3:	IR ← (MBR)

Or,

T1:	MAR←(PC)
T2:	MBR ← Memory
T3:	PC ← (PC) + I IR ← (MBR)

Next, Micro-Operations for the Indirect Cycle are as follows-

T1: MAR ← (IR (Address)) // Move the address field of instruction to MAR

T2: MBR ← Memory

T3: IR (Address) ← (MBR (Address)) // Address field of IR is updated from the MBR

Next, Micro-Operations for the Interrupt Cycle are as follows-

T1: MBR ← (PC)

T2: MAR ← Save_address

PC ← Routine_ Address

T3: Memory ← (MBR)

In the above interrupt Cycle, first the content of PC is moved to the MBR, so that it can be saved, as it will be needed after returning from the interrupt service routine. Then MAR is loaded with the address of the location where PC content is to be saved and PC will be loaded with the base address of the interrupt service routine. Finally, store the content of MBR into the memory as MBR is holding the old value of the PC.

Finally, for gaining the knowledge of Micro-Operations for the execute cycle, let us take an example of an ADD instruction for adding the content of location X with register R1, as follows-

ADD R1, X

So, for the execution of the above instruction, the following Micro-Operations may occurs-

T1: MAR ← (IR (Address))

T2: MBR ← Memory

T3: R1 ← (R1) + (MBR)
//add the content of R1 with MBR content.

At this point we are familiar with Micro-Operations and the two basic tasks of the control unit as follows-

- **Sequencing:** The control unit causes the processor to step through a series of Micro-Operation in a proper sequence, based on the program being executed.
- **Execution:** The control unit causes each Micro-Operation to be performed.

Now, come to the topic Control Unit Implementation. There are several implementation techniques and most of them fall into the following two categories-

- Hardwired Control
- Micro programmed Control

4.9.1 HARDWIRED CONTROL

In this technique, the control unit is a combinational circuit, where the input logic signals are transformed into a set of output logic signals.

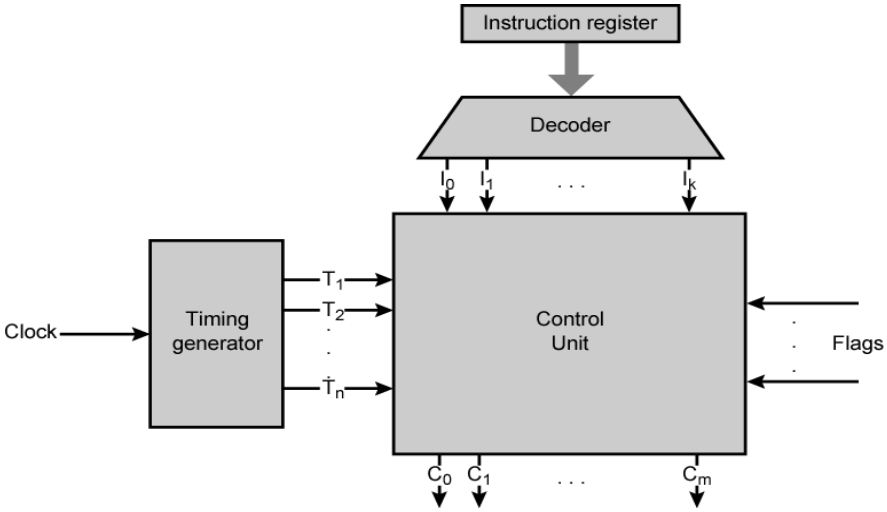

Figure- 4.7

Control unit takes instruction register, the clock, flags and control bus signals as input. Actually control unit does the use of opcode and will perform different actions for different instructions. Here, as shown in the **figure-4.7**, the opcode of the instructions available in the Instruction Register will be the input to the decoder and will be decoded by the decoder. So, the decoded output will be the input to the control unit and hence there would be a unique logic input for each opcode.

The clock is also input to the control unit that issues a repetitive sequence of pulses for measuring the duration of micro-operations. The control unit produces outputs which are control signals at different time units within a single instruction cycle that drives various components in the computer. These control signals are the sequence of events to be executed under the control of a system clock.

4.9.2 MICRO PROGRAMMED CONTROL

This technique do not use the interconnection of the basic logic elements to implement a control unit, because there should have a logic for sequencing through micro operations, for executing micro operations, for interpreting opcodes and for making decisions based on the ALU flags, but it is not easy task. So, micro programmed control technique uses a different way based on micro programming language.

We know that for any micro operation (termed as micro instruction), each control line output from the control unit is either 1 or 0. So, it is possible to construct a control word in which each bit represents one control line. Now, suppose we put together a sequence of control words to represents the sequence of micro operations performed by the control unit. Then, place the control words in a memory (known as control memory), with each word

having a unique address. Now, add an address field to each control word, indicating the location of the next word to be executed if a certain condition is true. For specifying the condition, add some other bits in each control word, and add one bit for each Internal CPU control signals, one bit for each system bus control signals. The representation of the control word after adding this address field and some other bits as mentioned is known as Horizontal Microinstruction and it will be executed in the following fashion-

1. For executing this microinstruction, turn on all the control lines indicated by a 1 bit; and leave all other control lines off indicated by a 0 bit. The resulting control signals will cause one or more micro-operations to be performed.
2. If the condition indicated by the condition bits is false, then execute the next microinstruction in sequence, else next instruction to be executed is indicated in the address field.

Now, let us examine the Functioning of Micro programmed Control Unit in the following **figure- 4.8**.

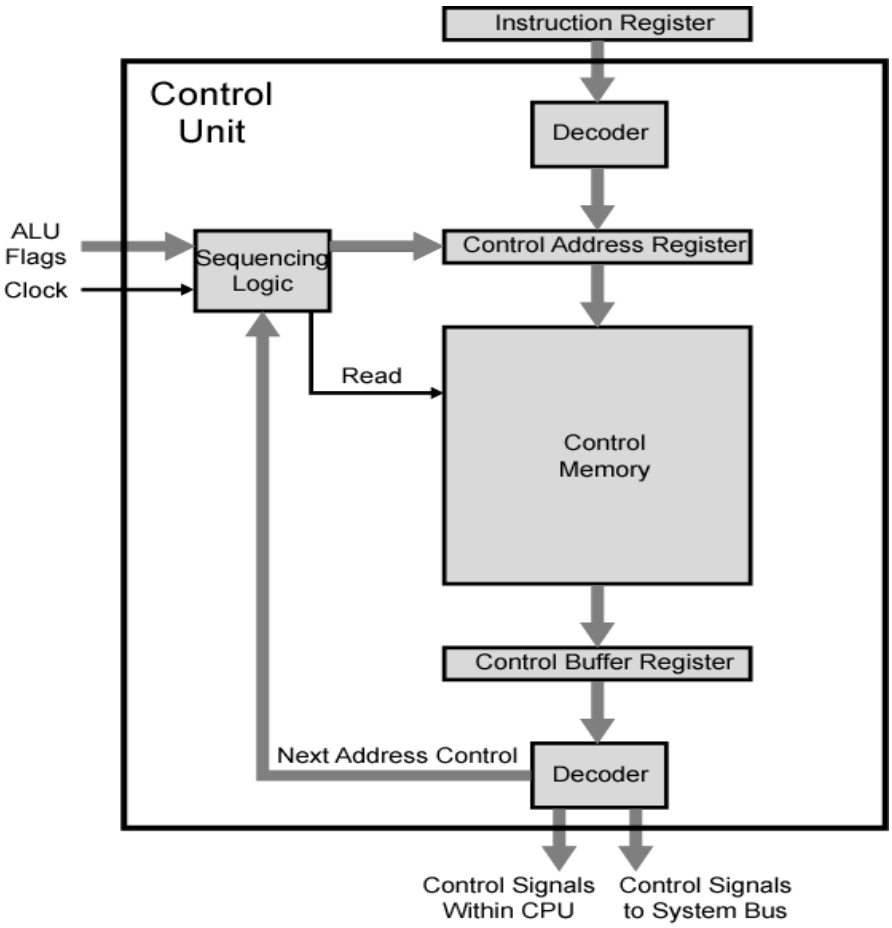

Figure- 4.8

In the figure, the set of micro instructions is stored in the Control Memory. The Control Address Register is for storing the address of the control memory, generated by the micro program sequencer,

or for storing the address of the next micro instruction to be read. Then Control Buffer Register is for holding the micro instruction transferred from the control memory after reading it.

Now, these are the steps in the following that happens during one clock pulse, as the functioning of micro programmed control unit-

1. For executing an instruction, the sequencing logic issues a READ control signal to the control memory.
2. The micro instruction or the control word, whose address is specified by the control address register, is then transferred to the control buffer register.
3. The content of the control buffer register generates control signals and next address information for the sequencing logic unit.
4. The sequencing logic unit loads a new address into the control address register based on the next control word address field from the control buffer register and the ALU flags. The new address may be next instruction address or may be a jump to a new routine based on a jump micro instruction or may be jump to a machine instruction routine.

4.9.3 HARDWIRED vs. MICRO PROGRAMMED

Both the techniques are having advantages and disadvantages over each other.

1. In micro programmed scheme, implementing control operation is through micro programmed organization where the control unit is implemented through programming.

   But, in hardwired scheme, implementing the control operation is through sequential circuits.

2. In micro programmed, speed of operation is low as it involves memory access

   But, in hardwired, speed of operation is high.

3. In micro programmed, changes in the control behavior are easy by modifying the micro instructions in control memory.

   But, in hardwired scheme, it is possible if the entire unit gets redesigned.