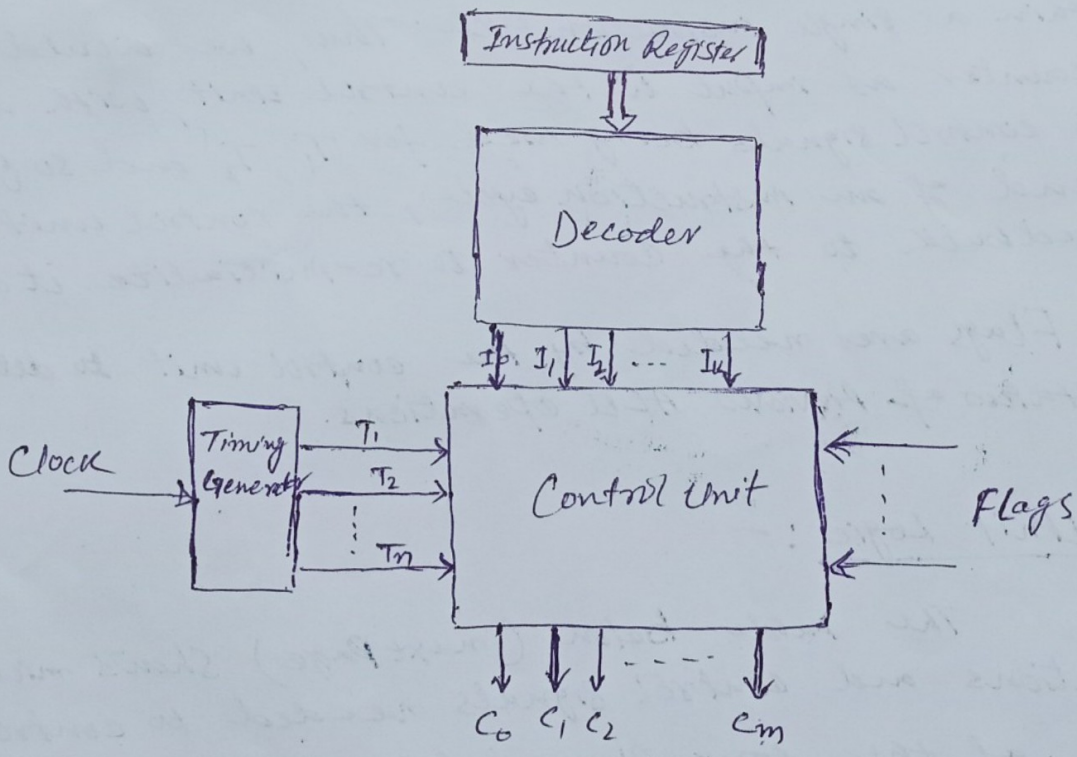
Design of Control Unit

Hardwired Control Unit :- Hardwired control unit is implemented with the help of logic gates, flip-flops, decoders etc. This organization is very complicated if we have large control unit. In this organization, if the design has to be modified or changed, requires changes in the wiring among the various components. Thus, the modification of all the combinational circuits may be very difficult.

In a hardwired implementation, the control unit is essentially a state machine circuit. Its input logic signals are transformed into a set of output logic signals, which are control signals.

Block-Diagram :-



Control Unit Inputs :- The key inputs are the instruction register, the clock and the flags.

First consider the instruction register. The control unit makes use of opcode and will perform different actions (issue a different combination of control signals) for different instruction. To simplify the control unit logic, there should be a unique logic input for each opcode. This function can be performed by a decoder, which takes an encoded input and produces a single output. In general, a decoder will have n binary inputs and 2^n outputs. Each of the 2^n different input patterns will activate a single unique output.

The clock portion of the control unit issues a repetitive sequence of pulses. This is useful for measuring the duration of micro-operations. Essentially, the period of clock pulses must be long enough to allow the propagation of signals along data path and through processor circuitry. However, the control unit emits different control signals at different time units within a single instruction cycle. Thus, we would like a counter as input to the control unit, with a different control signals being used for T_1, T_2 and so forth. At the end of an instruction cycle, the control unit must feedback to the counter to re-initialize it at T_1.

Flags are needed by the control unit to determine the states of previous ALU operations.

Control Unit Logic :-

The table below (next page) shows micro-operations and control signals needed to control three of the four phases of the instruction cycle.

	Micro-Operations	Active Control Signals
Fetch	t_1: MAR ← PC	C_2
	t_2: MBR ← Memory PC ← (PC)+1	C_5, C_R
	t_3: IR ← (MBR)	C_4
Indirect	t_1: MAR ← (IR(address))	C_8
	t_2: MBR ← Memory	C_5, C_R
	t_3: IR(address) ← (MBR(add.))	C_4
Interrupt	t_1: MBR ← ($\overline{PC}$)	C_1
	t_2: MAR ← Save-addr. PC ← Routine addr.	
	t_3: Memory ← (MBR)	C_{12}, C_W

C_R = Read control signal to system bus

C_W = Write control signal to system bus

Fetch, indirect, interrupt and execute cycles are represented by different combinations of variable P, Q as

P	Q	
0	0	Fetch cycle
0	1	Indirect cycle
1	0	Interrupt cycle
1	1	Execute cycle

Then, following Boolean expression defines C_5:

$$C_5 = \bar{P}\bar{Q}T_2 + \bar{P}QT_2$$

Similarly we can define the Boolean expression for different control signals. As a result

their would be a set of Boolean equations that defines the behaviour of the control unit and hence of the Processor.

Advantages:

(i) Hardwired control unit is fast because control signals are generated by combinational circuits.

Disadvantages :-

(i) More is the control signals required by CPU; more complex will be the design of control unit.

(ii) Modifications in control signals are very difficult. That means it requires re-arranging of wires in the hardware circuit.