

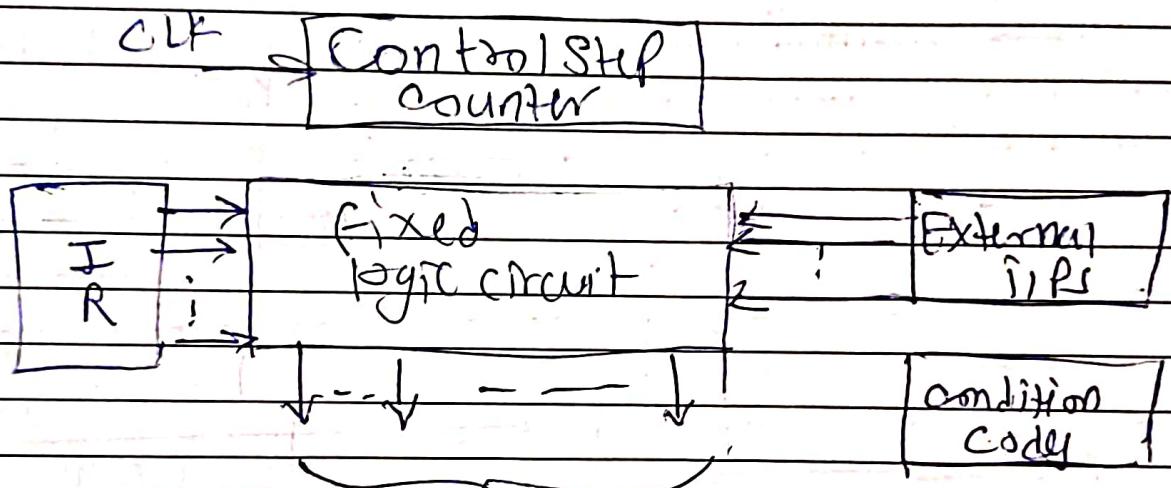
UNIT III

Processing Unit

Hardwired control

There are 2 different approaches used for control unit design.

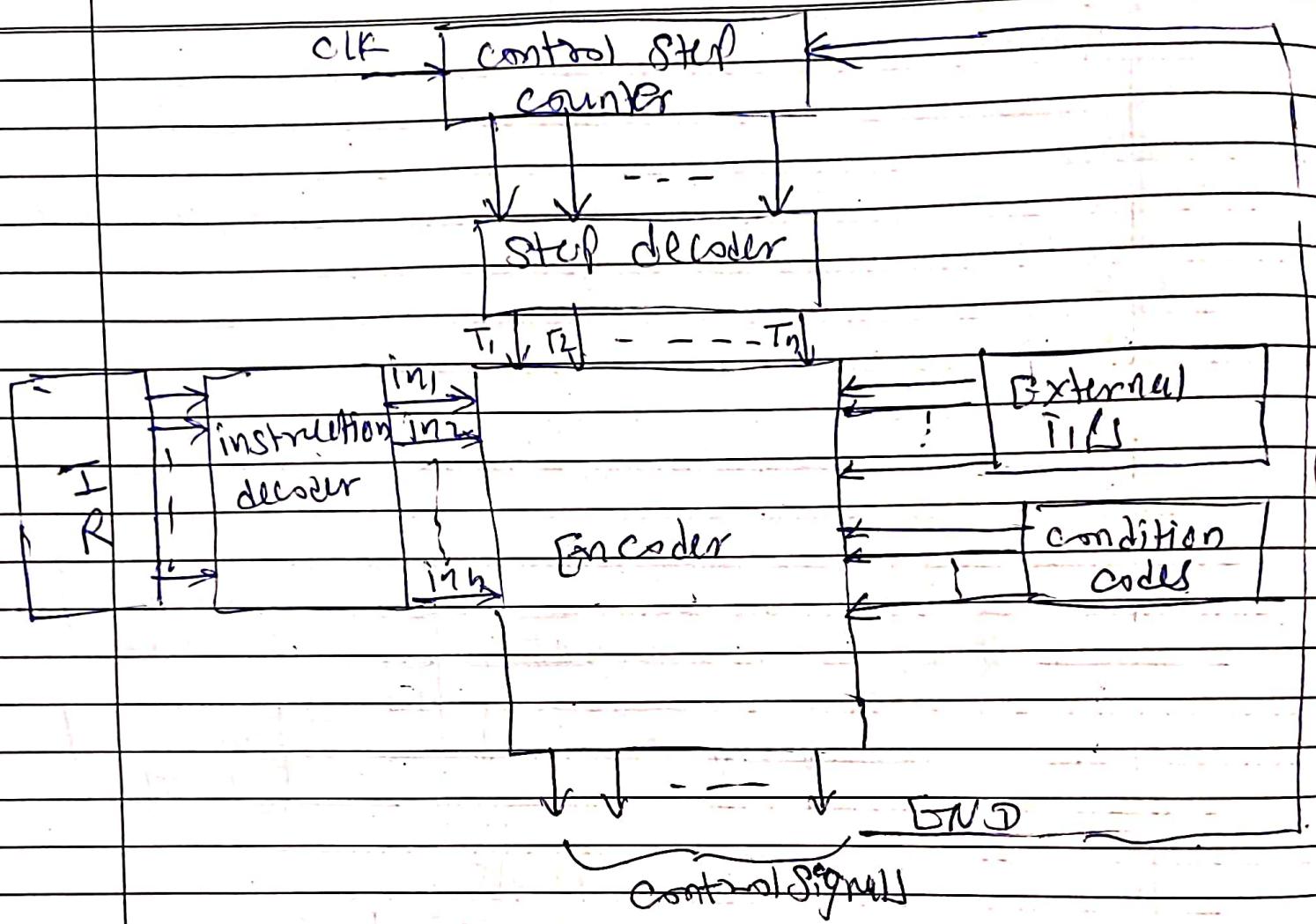
- i) Hardwired control
- ii) microprogrammed control.



Typical hardwired control logic

In the hardwired control, the control units use fixed logic circuit to interpret instructions and generate control signal from them.

The fixed logic circuit blocks include combinational circuit that generates the required control bits for decoding and encoding functions. By separating the decoding and encoding functions - detail block diagram for hardwired control unit.



Detail block diagram for Hardwired control unit

The instruction decoder decodes the instruction loaded in the IR. If IR is an 8-bit register then instruction decoder generates 2^8 i.e. 256 lines one for each instruction. According to code in the IR, only one line amongst all 256 lines of decoder goes high i.e. set to 1. and all other lines are set to 0. The Step decoder provides a separate signal line for each step or time slot, in a control sequence. The encoder gets its input from instruction decoder, step decoder, external ife and condition codes. it uses all

These if's to generate the individual control signals. After execution of each instruction end signal is generated which reset control step counter and make it ready for generation of control step for next instruction.

Designed methods for Hardwired control unit.

I State Table Method

It is standard algorithmic approach to sequential ckt design.

II Delay Element method —

It is a heuristic method based on the use of clocked delay elements for control signal timing.

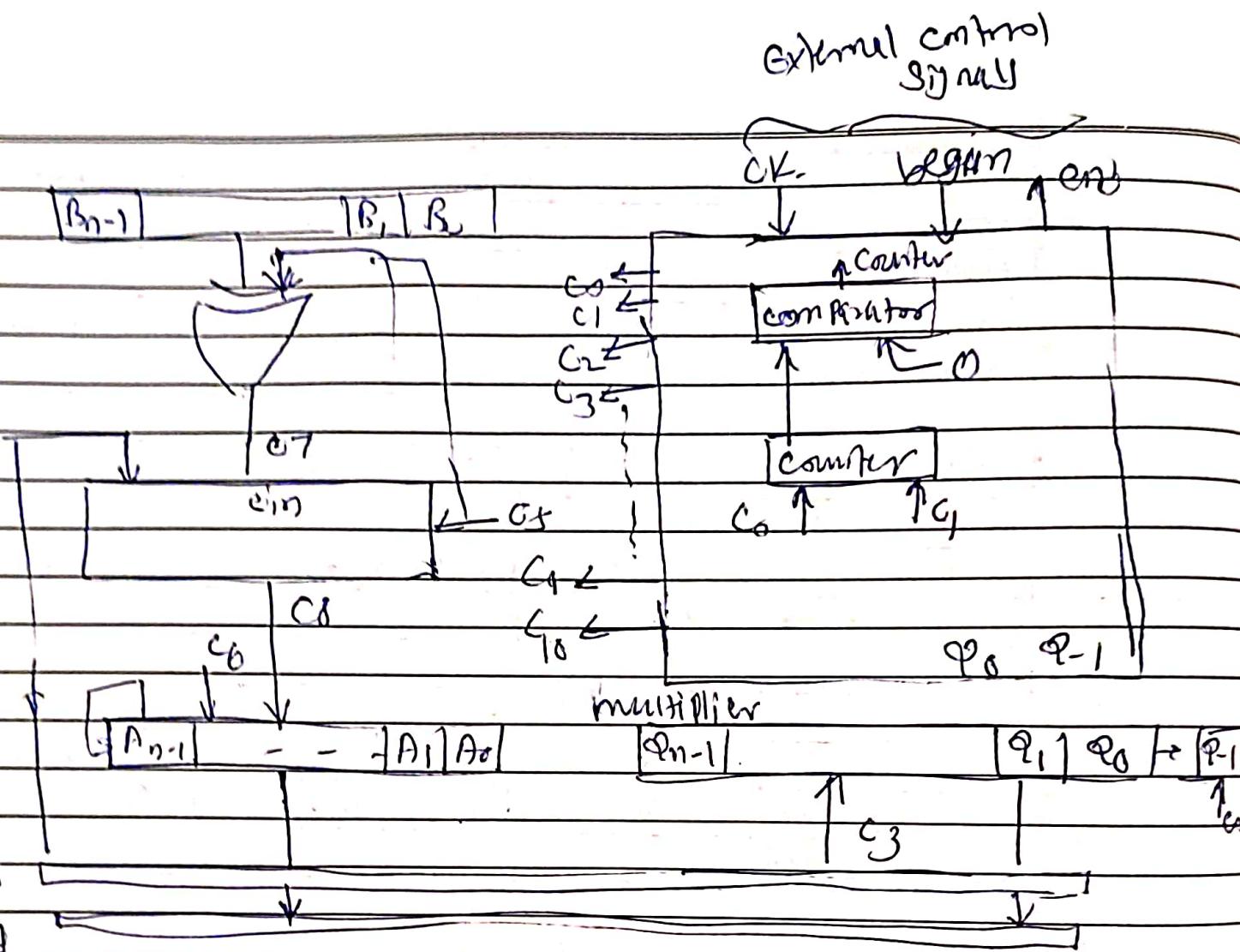
III Sequence - Counter method

It uses counters for timing purposes.

IV PLA method —

It uses Programmable logic array.

The 2's complement multiplier control circuit for illustration of above designing method. The Hardwired control unit for 2's complement multiplication is already introduced.



multiplier control circuit with control signals
control signal

C_0

clear $A, Q_1 : R \text{ count} \leftarrow n$

C_1

decrement count

C_2

Transfer word on index to B

C_3

- - - - -

to Q

C_4

Shift-right register $A \rightarrow Q \oplus Q_1$

C_5

2^{nd} complement multiplicand

C_6

Transfer A to left i/p of adder

C_7

- - - - - B to right i/p of adder

C_8

Transfer adder o/p to A

C_9

Transfer A to output

C_{10}

Transfer Q to output

In general, a control point is associated with each 'distinct' action / operation. A common signal is used for operation which can take place simultaneously for eg. register A & Q, can be reset and count can be loaded in simultaneously. Hence only one control signal C_5 is assigned for three operations. Control signal C_7 is used to decrement the iteration count. Control signals C_2 & C_3 transfer a data word from the 16 bit buses to registers B and Q.

In case of subtraction, control signal C_5 does the 2's complement conversion of the multiplicand we get 2's control signals C_6 & C_7 transfer a data from A and R registers to the left and right side of adder respectively. Control signal C_8 is used to transfer adder O/P to A register. To transfer final result from A & Q registers on the O/P bus. OUTBUS. Control signals C_9 & C_{10} are used. The control signal COUNT is set to 0 when counter has non zero. Otherwise it is set to logic 1. BEGIN and COUNT signals, the right most bit Q_0 of the Q register and register serve as the primary ticks of the control unit.

Design methods for Hardwired control unit:

I State table method -

It is necessary to construct a state table for the control unit. We associate a state with every microoperation block, giving nine states labelled S_0 through S_8 . S_0 state represents the idle or waiting state of the control unit.

There are 4 primary i/p signals BEGIN, COUNT, Q_0 and Q_1 . Hence there are 16 possible i/p combinations. Each entry indicates the next state followed by a list of the control signals that are activated.

For ex. S_1 the next state is S_2 and it is reached by activating control signals C_0 and C_2 . Certain state and i/p signal combinations should not occur during normal operation, so the corresponding table entries are left blank. For example the BEGIN signal should assume the 1 value only when the control unit is in the idle state S_0 , similarly COUNT.

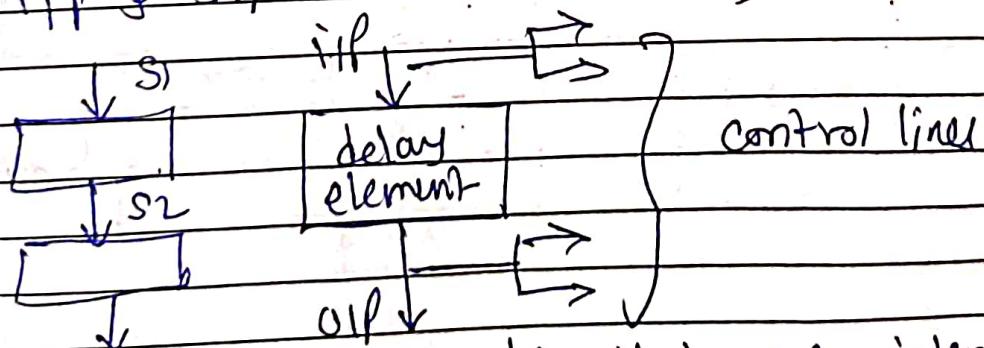
i/p	States								
BEGIN COUNT $Q_0 Q_1$	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8
0 0 0 0	S_0	S_2, C_0	S_4, C_3						
0 0 0 1	S_0	S_2, C_0, C_2						S_4, C_1, C_4	

II Delay Element Method

The control signals or groups of control signals from the control unit are activated in a proper sequence. There is a specific time delay between activation of two conjugative control signals or groups of control signals. A sequence of delay elements can be used to generate control signals one after the another. To ensure synchronous operation, the delay elements are implemented by flip-flops and controlled by a common clock signal.

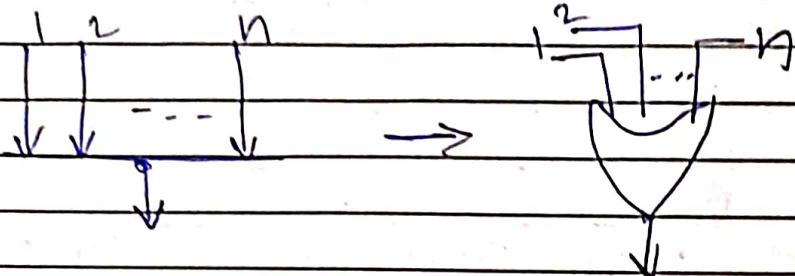
A control unit using delay elements can be constructed directly from flowchart that specifies required control signal sequences. Before going to see the implementation of multiplier control unit we see the simple rules to drive control circuit from the flowchart.

Rule 1. Each sequence of two successive micro-operation require a delay element. The signals that activate the control lines are taken directly from the if & ofp lines of the delay element.

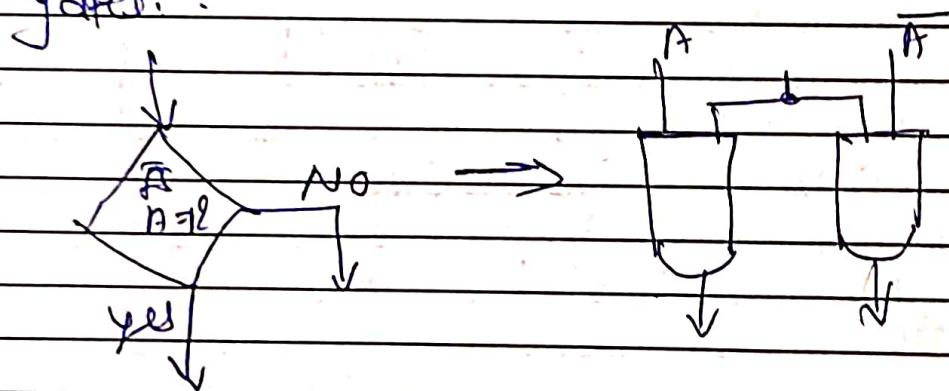


The signals that are intended to activate some control line are logically ORed to get one common ofp signal or

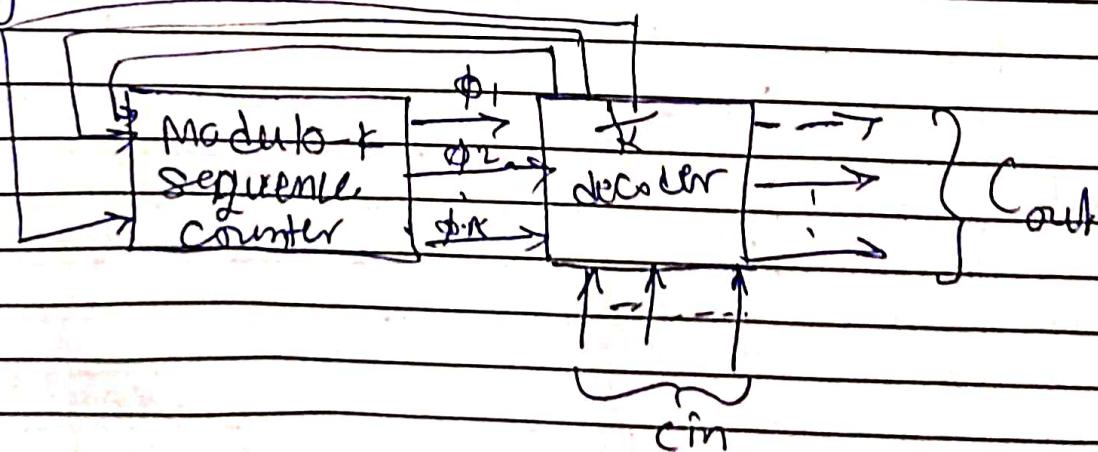
Rule 2 n lines in the flowchart merge to a common line are transformed into i/p OR gate.



Rule 3 A decision box can be implemented by two AND gates.

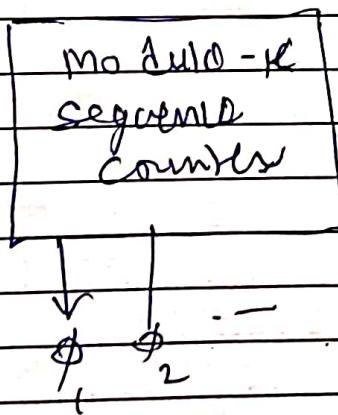


III Sequence counter Method



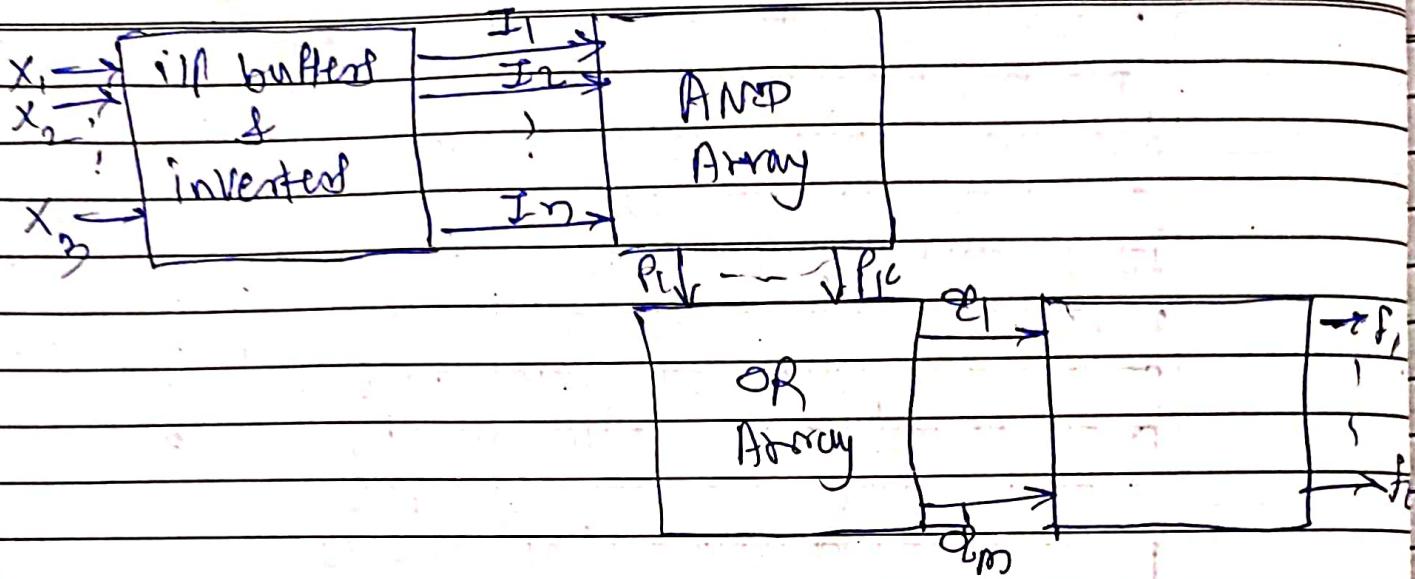
The sequence counter consists of modulo- k counter, decoder and if signals to control the of ϕ_i of ϕ_k of modulo- k counter. The ϕ_i of the modulo- k

counter consists of modulo- k counter. The output of the modulo- k counter is connected to the input of $1/k$ decoder. When the count enable input is connected to a clock source the counter cycles continually through its k states, & decoder generates k pulse signals (ϕ_i) on its output lines. These are called as phase signals. consecutive pulses of phase signals are separated by one clock period. Two additional input lines and a flip-flop are provided for training the counter on and off. A pulse on the begin line causes the counter to begin cycling through its states by logically connecting the count enable signal to the clock source. A pulse on the end line disconnects the clock and resets the counter.



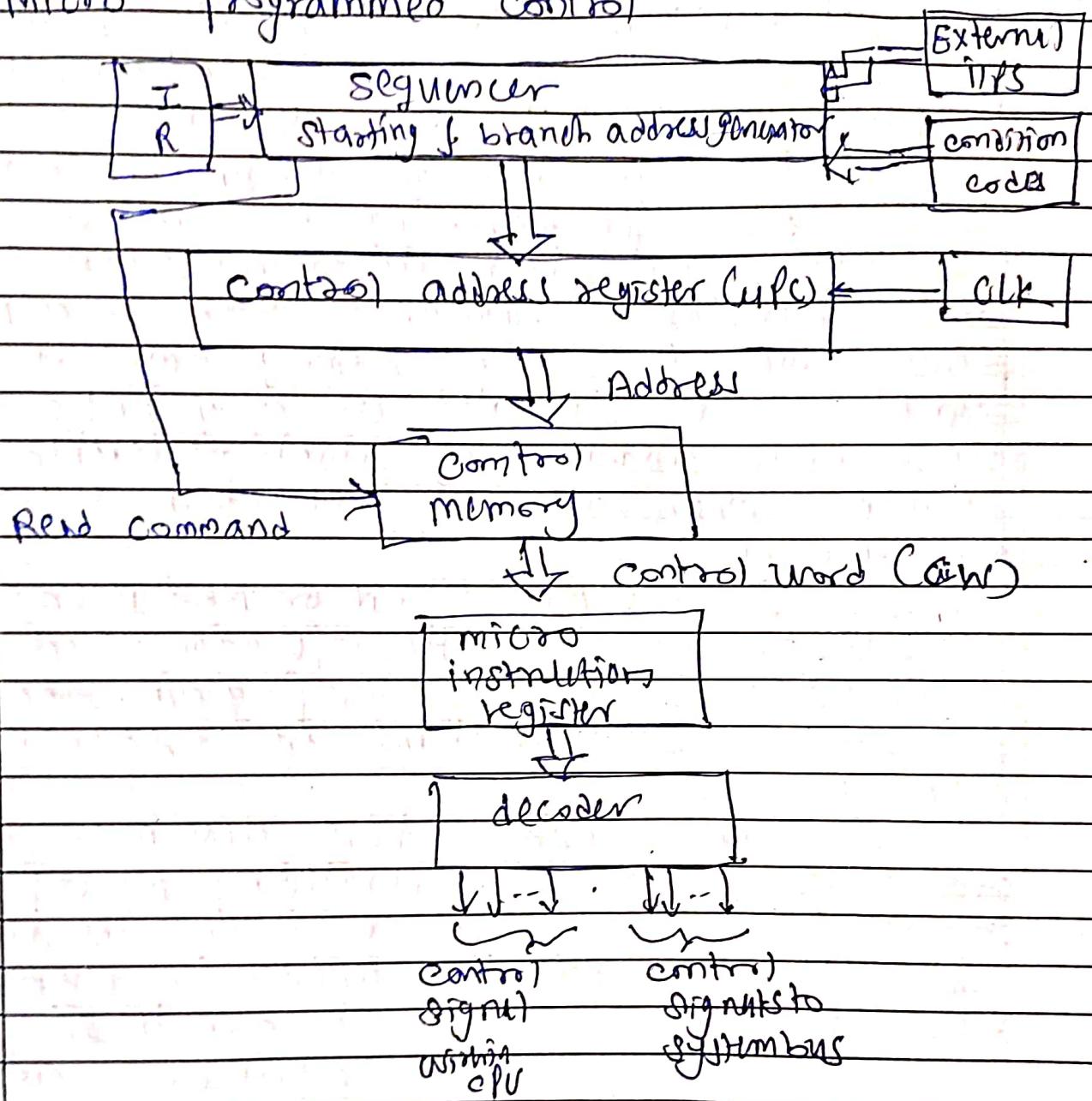
II PLA Method -

The Control Unit method suitable for small control unit due to their size and complexity. Programmable logic array (PLA) consist of an array of AND gates followed by array of OR gates. AND and OR arrays both can be programmed to implement combinations logic functions of several variables. x_1, x_2, \dots, x_n are input signals. F_1, F_2 the combinations of input signals are available as output.



Using Scan logic, we can give instruction code, contents of control step counter flags and condition codes as an ip to the PLA and programmed AND & OR arrays to get o/p as control signal.

Micro Programmed Control



Every instruction in a processor is implemented by a sequence of one or more sets of concurrent micro-operations. Each micro operation is associated with a specific bit of control lines which, when activated, causes that micro operation to take place. The number of instructions and control lines is often

in the hundreds, the complexity of hardwired control unit is very high. it is costly and difficult to design. the hardwired control unit is relatively inflexible because it is difficult to change the design, if one wishes to correct design error or modify the instruction set.

microprogramming is a method of control unit design in which the control signal selection and sequencing information is stored in a ROM or RAM called a control memory CM. The control signals to be activated at any time are specified by a micro instruction which is fetched from CM in much similar way an instruction is fetched from main memory.

A sequence of one or more microoperations designed to control specific operation such as addition, multiplication is called a microprogram. The microprogram for all instructions are stored in the control memory.

The address where these microinstructions are stored in CM is generated by micro program sequencer / microprogram controller. The microprogram sequencer generates the address for microinstruction according to the instruction stored in the IR.

microprogrammed control unit consist of control memory, control address register, micro instruction sequencer and microprogram sequencer.

control Unit —
the address of the control address register (CAR) holds the address of the next microinstruction to be read. every time a new instruction is loaded into the IR the off of the block (addr "starting address generator")

is loaded into the MIFC. When address is available in control address register, the sequencer issues RBAQ command to the control memory. After issue of RBAQ command, the word from the addressed location is read into the microinstruction register. The MIFC is then automatically incremented by the clock, causing successive microinstructions to be read from the control memory. The content of the micro instruction register generates control signals which are delivered to various parts of the processor in the correct sequence.

Advantages of microprogrammed control

1. It simplifies the design of control unit. Thus it is both, cheaper and less error prone to implement.
2. Control functions are implemented in SW rather than hardware.
3. The design process is orderly and systematic.
4. more flexible, can be changed to accommodate new system specifications or to correct the design errors quickly and cheaply.
5. Complex function such as floating point arithmetic can be realized efficiently.

Disadvantage

1. A microprogrammed control unit is slower than the hardware control unit.
2. extra hardware cost.

Attribute	Hardwired control	microprogrammed control
speed	Fast	Slow
control functions	implemented in HW	Implemented in SW
Ability to handle complex instruction sets	difficult	Easier
Flexibility	not flexibility	more flexibility
Design process	complicated	orderly & systematic
Applications	RISC microprocessors	mainframes, microprocessors
instruction set size	usually under 100 instructions	over 100 instructions
ROM size	—	2K to 10K by 2s - 160 bit microinstructions
chip area efficiency	use less area	use more area

Application of micro Programming

- 1 Realization of computer
- 2 Simulation
- 3 operating system support
- 4 high level language support
- 5 micro diagnosis
- 6 new technology