COMPUTER SCIENCE

Computer Organization and Architecture

ALU & Control unit

Lecture_04

Vijay Agarwal sir

TOPICS TO BE COVERED

01 Control Unit

02 Control Unit Design

Working of Computer

ALU Data Path.

Micro operation

Micro Program.

Control-unit

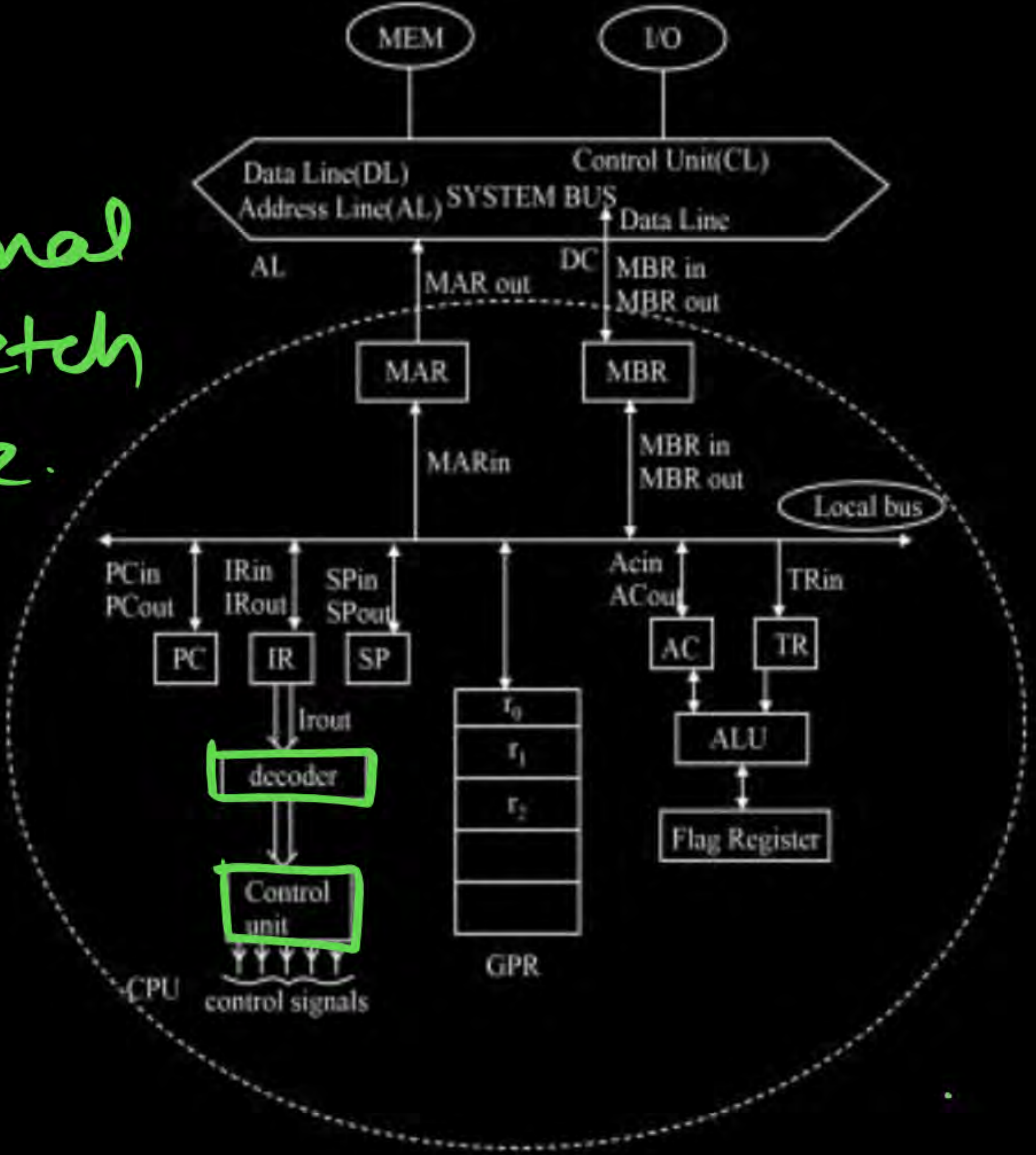
- Control Unit is the Supervisior in the System that Control Each & Every Activity.

- Control Unit takes various input but Produce Only One Output Called Control Signal (Sequence of Control Signal).

Control Unit Functional Requirements

- By reducing the operation of the processor to its most fundamental level we are able to define exactly what it is that the control unit must cause to happen
- Three step process to lead to a characterization of the control unit:
  - ① Define basic elements of processor
  - ② Describe micro-operations processor performs
  - ③ Determine the functions that the control unit must perform to cause the micro-operations to be performed
- The control unit performs two basic tasks:
  - Sequencing
  - Execution

Same Control Signal Used in Fetch & Decode.



Control Unit Functional Requirements

- By reducing the operation of the processor to its most fundamental level we are able to define exactly what it is that the control unit must cause to happen
- Three step process to lead to a characterization of the control unit:
  - Define basic elements of processor
  - Describe micro-operations processor performs
  - Determine the functions that the control unit must perform to cause the micro-operations to be performed
- The control unit performs two basic tasks:
  - Sequencing
  - Execution

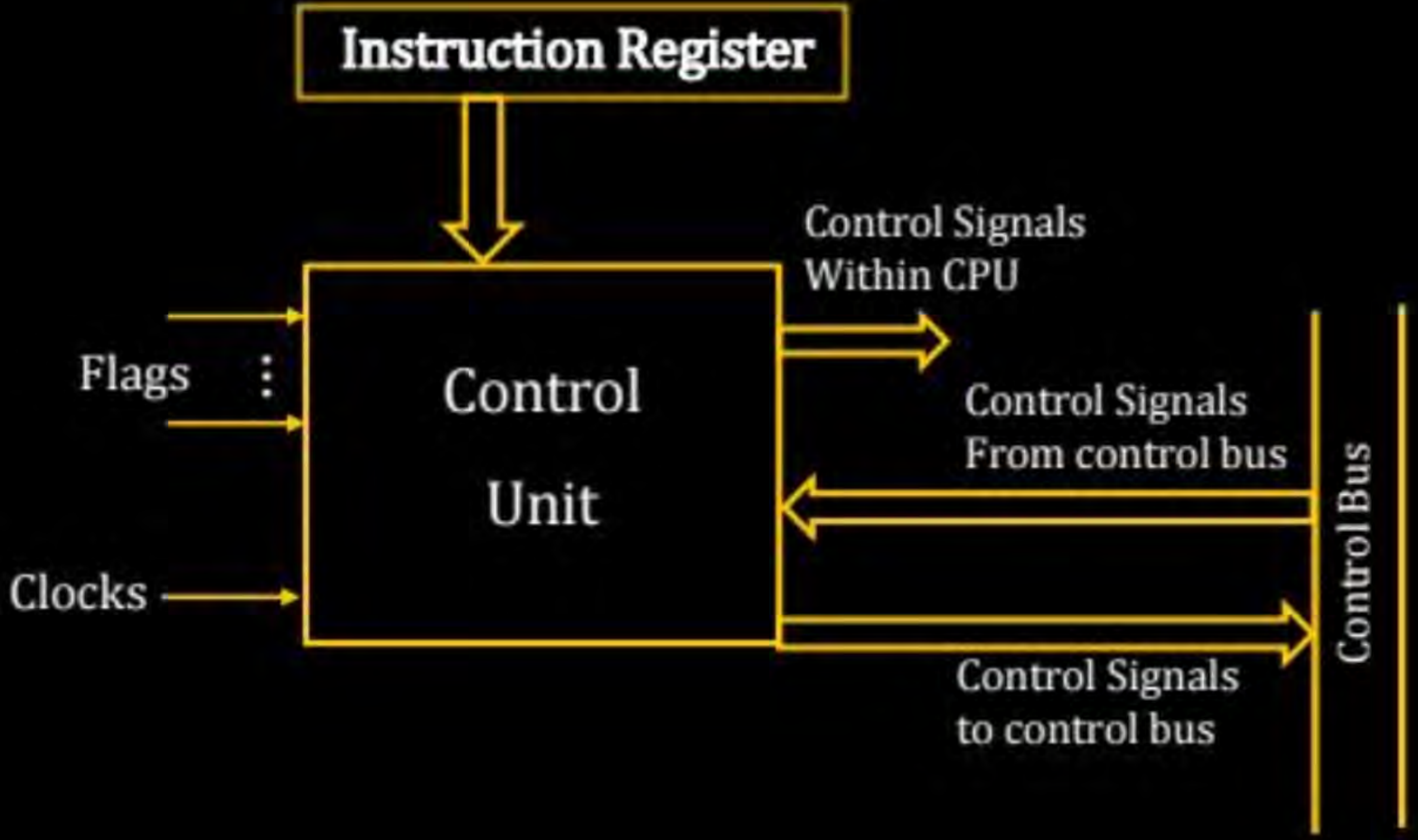
Control Unit

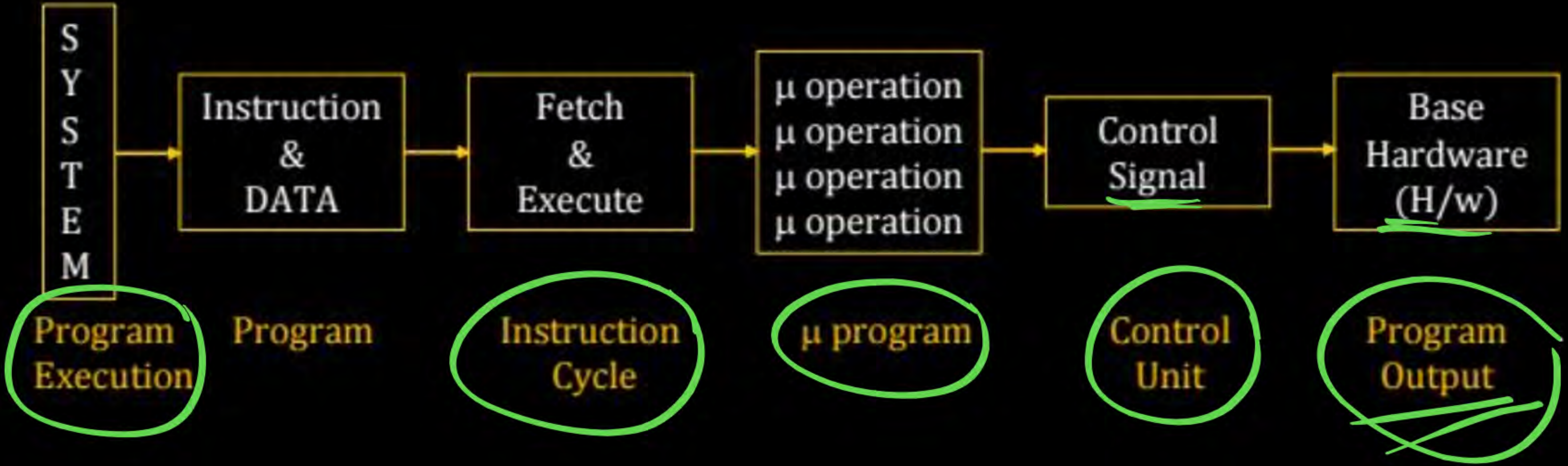
Control unit is the Supervisor in the System that control each & every activity.

- □ Control Signals are implement in a Control Unit.
- □ Control Signal are Required to execute the micro operation.
- □ Micro operation is the elementary operation in the hardware.
- □ Control unit generates the sequence of control Signal.
- □ Control Signal are Directly executed on a Base Hardware (H/W) So H/W generate the desired Response.

Computer System Functionality is Program Execution.

Block Diagram of the Control Unit





SYSTEM → Instruction & DATA → Fetch & Execute → µ operation µ operation µ operation µ operation → Control Signal → Base Hardware (H/w)

Program Execution

Program

Instruction Cycle

µ program

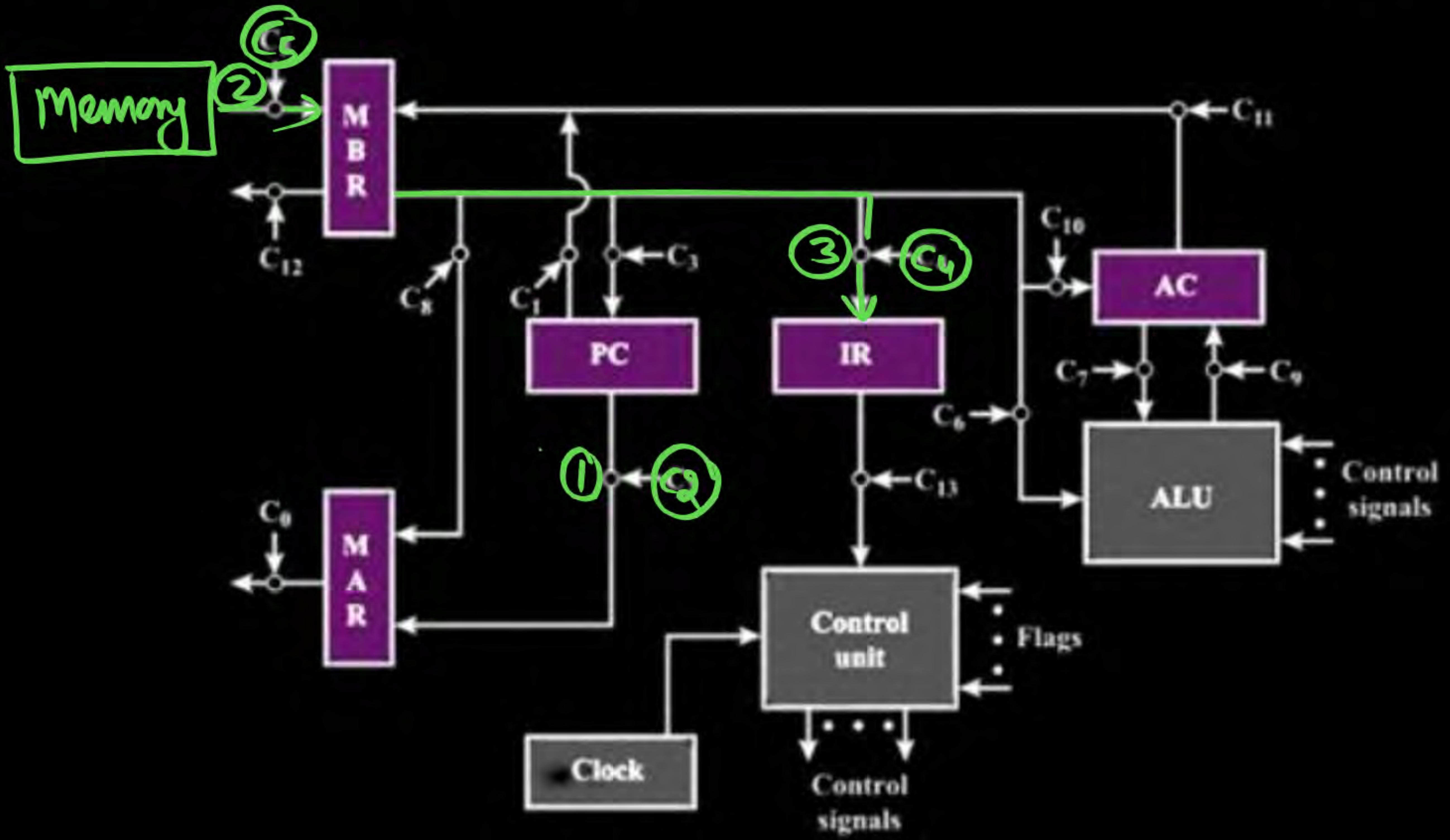
Control Unit

Program Output

Micro-operations & Control Signals

	Micro-operations	Active-control Signals
Fetch:	T_1: MAR ℜ (PC) (or) PC → MAR PCout MARin	C_2
	T_2: MBR ℜ Memory PC ℜ (PC) + 1 MB Rin	$\underline{C_5}$, C_R
	T_3: IR ℜ (MBR) MBR_{out} IRin	C_4
Indirect:	T_1: MAR ℜ (IR(Address))	C_8
	T_2: MBR ℜ Memory	C_5, C_R
	T_3: IR(Address) ℜ (MBR (Address))	C_4
Interrupt:	T_1: MBR ℜ (PC)	C_1
	T_2: MAR ℜ Save-address PC ℜ Routine-address	
	T_3: Memory ℜ (MBR)	C_{12}, C_W

Data Paths & Control Signals



Pre Requirement of the CU Design is as follow

1) How many Control Lines are present in the Hardware [S0, S1, S2, S3...]

2) How many Instruction are implemented in the Hardware [I1, I2, I3....]

3) How many Micro operation are required for each Instruction [T1, T2, T3...]

4) What the control Signal Required for each micro operation for each Instruction.

③ Fetch

T_1 PC → MAR

T_2 M(MAR) → MBR

T_3 MBR → IR

Execute

Direct AM	Indirect AM
T_1	T_1
T_2	T_2
T_3	T_3
	T_4
	T_5

Working

- Control Unit generate the Control Signals.
- These Control Signals are same for Fetch & Decode.
- After the Decode, Different type Control Signal (Control Word) generated According to Different type of operations.

- At the Control Unit Design Time, Designer decide Which Control are generated in Which Cycle (T_1, T_2, T_3..T_n) for Different type of Instruction. that Stored in a Table. (Logic)

later we Make a Boolean function for implementation.

Q. Consider the following hypothetical CPU which uses 3 data register A, B & C and supports 3 instruction I_1, I_2 & I_3. Obtain the logic function that will generate the hardwired control for the signal Ain & Bout with the following data.

I_1

T_1: B → A

Bout Ain

	I_1	I_2	I_3
T_1	Ain, Bout	Ain, Cin, Bout	Bin, Bout
T_2	Bin, Cin, Aout	Ain, Aout	Ain, Bin, Cout
T_3	Bin, Bout	Bin, Bout	Bin, Bout
T_4	Cin, Aout	Bin, Aout	Ain, Aout
T_5	End	End	End

Q.

A hardwired CPU use 10 control signals SI to S10 in various time steps T1 to T5 implement 4 instructions I1 to I4 as shown below.

	T1	T2	T3	T4	T5
I1	S1, S3, S5	S2, S4, S6	S1, S7	S10	S3, S8
I2	S1, S3, S5	S8, S9, S10	S5, S6, S7	S6	S10
I3	S1, S3, S5	S7, S8, S10	S2, S6, S9	S10	S1, S3
I4	S1, S3, S5	S2, S6, S7	S5, S10	S6, S9	S10

T_1 Cycle

S_1

S_3

S_5

Control Signals will be Implemented into the Control Unit by using following Approach:

1) HARDWIRED CU Design

2) MICRO-PROGRAMMED CU Design

HARDWIRED CU DESIGN

- In the Hardwired CU Design Control Signals are Expressed in S.O.P Expression (Sum of Product).
- They are Directly realized on a Hardwired.

- In Hardwired Control Unit they used fixed Logic Circuit to Interpret the Instruction & generate the Control Signal.

Note Hardwired CU is Fastest CU Design.

Note RISC is a Hard wired CU.

HARDWIRED CU DESIGN

Dis Advantage :

- It is Not Flexible.
- Even a Minor Modification Require Redesigning & Rewiring.
- It does not Support New operation. (Once designed)

Q. Consider the following hypothetical CPU which uses 3 data register A, B & C and supports 3 instruction I_1, I_2 & I_3. Obtain the logic function that will generate the hardwired control for the signal Ain & Bout with the following data.

	I_1	I_2	I_3
T_1	Ain, Bout	Ain, Cin, Bout	Bin, Bout
T_2	Bin, Cin, Aout	Ain, Aout	Ain, Bin, Cout
T_3	Bin, Bout	Bin, Bout	Bin, Bout
T_4	Cin, Aout	Bin, Aout	Ain, Aout
T_5	End	End	End

$$B_{out} = T_1I_1 + T_1I_2 + T_3 + T_3I_1 + T_3I_2 + T_3I_3$$

$$\Rightarrow T_1(\underbrace{I_1 + I_2 + I_3}_{1}) + T_3(\underbrace{I_1 + I_2 + I_3}_{1})$$

$$\boxed{B_{out} = T_1 + T_3} \text{ Ans}$$

$$A_{in} = T_1I_1 + T_1I_2 + T_2I_2 + T_2I_3 + T_4I_3$$

$$\boxed{A_{in} = T_1I_1 + (T_1 + T_2)I_2 + (T_2 + T_4)I_3} . \text{ Ans}$$

Process :

✓ Step 1 : Search where the control signals Ain & Bout are present.

✓ Step 2 : Options are in I.T format or T.I format.

✓ Step 3 : For any particular time interval. Is the control signal presents for all the instructions?

$I.T$

$I_1T_2 + I_1T_3$

or

$T.I$

$T_2 I_1 + T_3 I_1$

$T_1 \underbrace{(I_1 + I_2 + I_3)}_{L}$

$T.L = \boxed{T_1}$

Step 1 : Search where the control signals Ain & Bout are present.

Step 2 : Options are in I.T format or T.I format.

Step 3 : For any particular time interval. Is the control signal presents for all the instructions?

$\text{Ain} = T_1\ I_1 + (T_1 + T_2)\ I_2 + (T_2 + T_4)\ I_3$

$\text{Bout} = T_1 + T_3$

↓

Bout is present for all instruction during T_1 & T_3

Q

GATE

A hardwired CPU use 10 control signals SI to S10 in various time steps T1 to T5 implement 4 instructions I1 to I4 as shown below.

S_5 & S_{10}

	T1	T2	T3	T4	T5
I1	S1, S3, S5	S2, S4, S6	S1, S7	S10	S3, S8
I2	S1, S3, S5	S8, S9, S10	S5, S6, S7	S6	S10
I3	S1, S3, S5	S7, S8, S10	S2, S6, S9	S10	S1, S3
I4	S1, S3, S5	S2, S6, S7	S5, S10	S6, S9	S10

$$S_5 = T_1(I_1+I_2+I_3+I_4) + T_3(I_2+I_4)$$

$$\boxed{S_5 = T_1 + (I_2+I_4)T_3}$$

Ans

$$S_{10} = T_2I_2 + T_2I_3 + T_3I_4 + T_4I_1 + T_4I_3 + T_5I_2 + T_5I_4$$

$$\boxed{S_{10} = (I_2+I_3)T_2 + I_4T_3 + (I_1+I_3)T_4 + (I_2+I_4)T_5}$$

Ans

Which of the following pairs of expressions represent the circuit for generating control signals S5 and S10 respectively [(IJ + Ik) Tn indicates that the control signal should be generated in time step Tn if the instruction being executed is [IJ to IK]?

(a) S5 = Tl + I2.T3 and S10 = (I1 + I3).T4 + (I2 + I4).T5

(b) S5 = TI + (I2 + I4).T3 and S10 = (I1 + I3).T4 + (I2 + I4).T5

(c) S5 = T1 + (I2 + I4).T3 and S10 = (I2 + I3 + I4).T2 + (I1 + I3).T4 + (I2 + I4).T5

[✓ (d)] S5 = T1 + (I2 + I4).T3 and S10 = (I2 + I3).T2 + I4.T3 + (I1+I3).T4 + (I2+ I4).T5

Ans (D).

Q.

A CPU has only three instructions I1, 12 and 13, which use the following signals in time steps T I—T5:

I1: Tl : Ain, Bout, Cin
T2 : PCout, Bin
T3 : Zout, Ain
T4 : PCin, Bout
T5 : End

I2: T1 : Cin, Bout, Din
T2 : Aout, Bin
T3 : Zout, Ain
T4 : Bin, Cout
T5 : End

I3: T1: Din, Aout
T2 : Ain, Bout
T3 : Zout, Ain
T4 : Dout, Ain
T5 : : End

$$Ain = I_1T_1 + I_1T_3 + I_2T_3 + I_3T_2 + I_3T_3 + I_3T_4$$

$$\Rightarrow T_1I_1 + T_3(I_1+I_2+I_3) + T_2I_3 + T_4I_3$$

$$\boxed{Ain = T_1I_1 + T_2I_3 + T_4I_3 + T_3.}$$ Ans

Which of the following logic functions will generate the hardwired control for the signal Ain?

[GATE CSE 2004]

[✓ (a)] T1. I1 + T2. I3 + T4.I3 +T3

(b) (T1 + T2 + T3).I3 + T1.I1

(c) (T1 + T2).I1 + (T2 + T4).I3 + T3

(d) (T1 + T2).I2 + (T1 + T3). I1 + T3

✓ ① Hardwired CU Design.

✓ ② Micro Programmed CU Design.

②Micro Programmed Control Unit

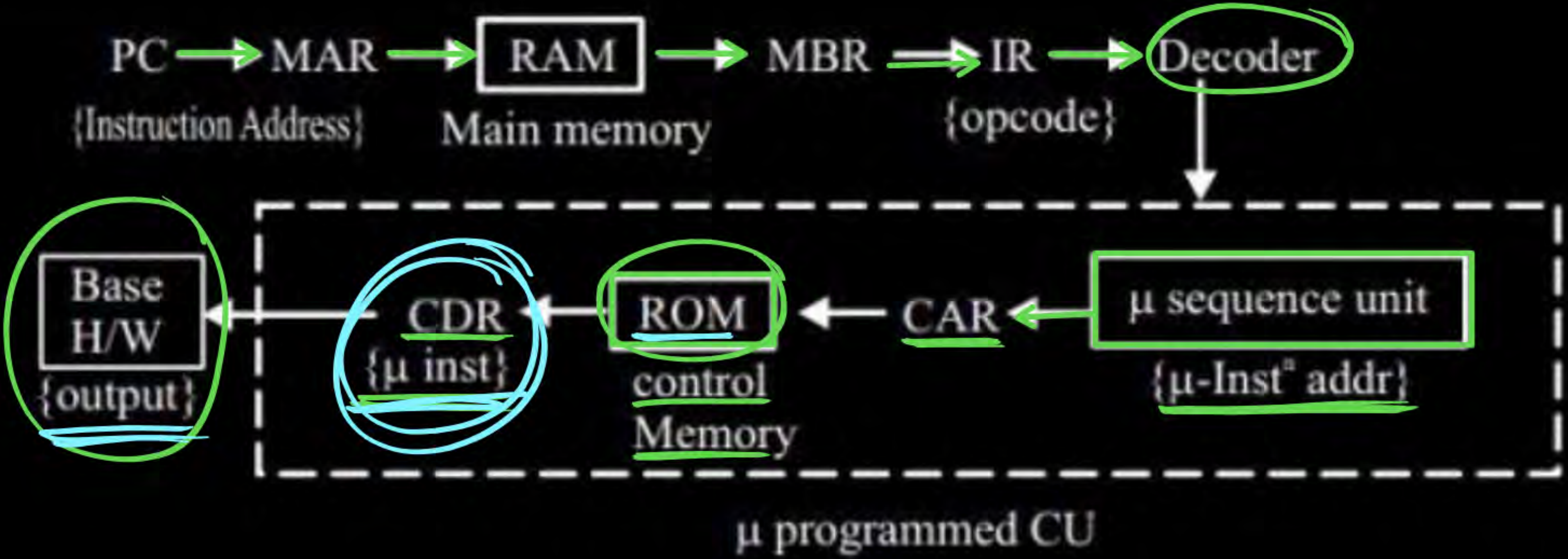
: In Micro-Programmed CU Design Control words are stored in the Control Memory.

- According to the type of operation Control Signals are generated.

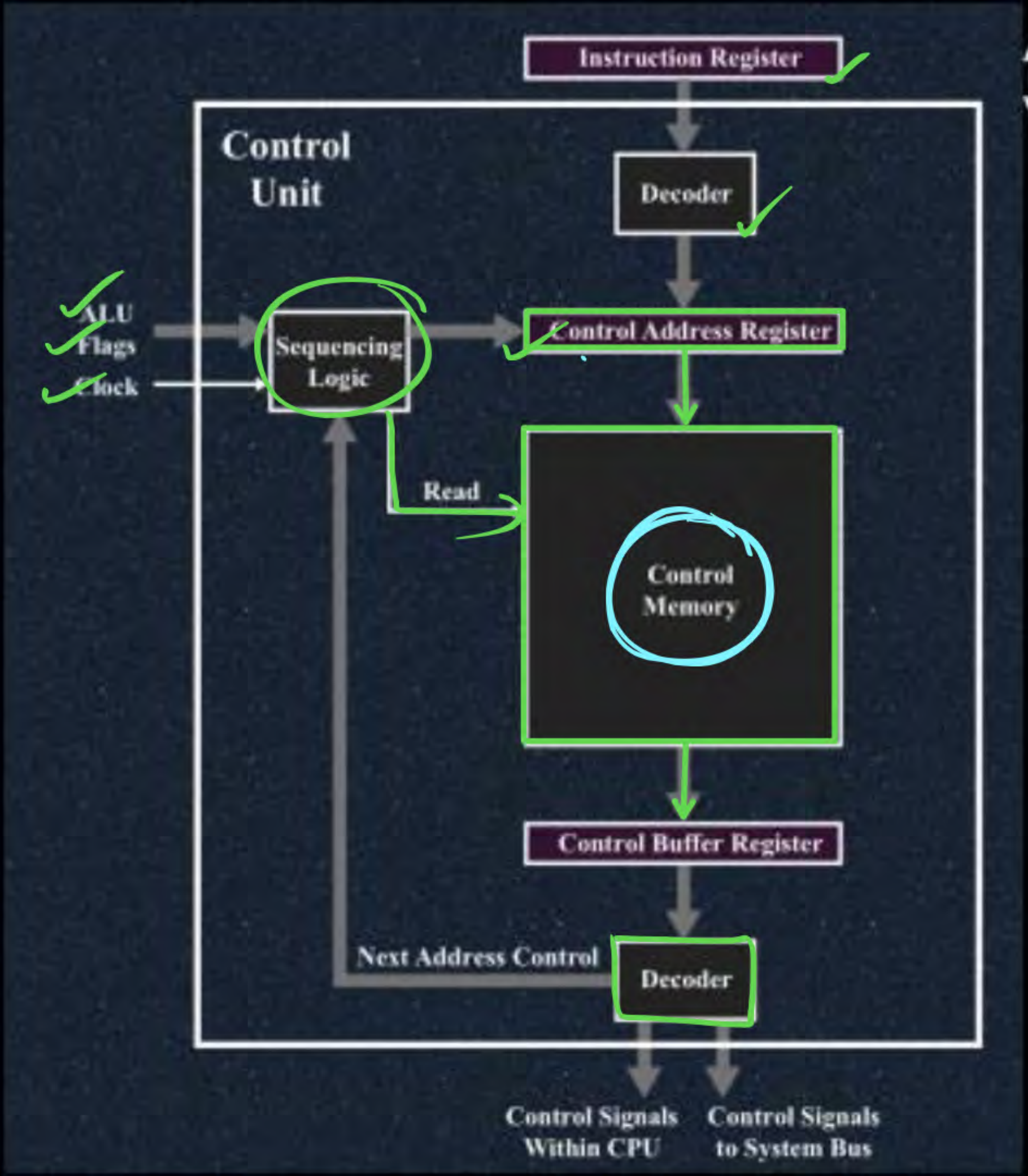
- Control Memory is associated with CAR & CDR Register. to Contain Control Memory Address & Data Respectively.

CAR : Control Address Register
CDR : Control Data Register.

MICRO-PROGRAMED CU DESIGN



Functioning of Microprogrammed Control Unit

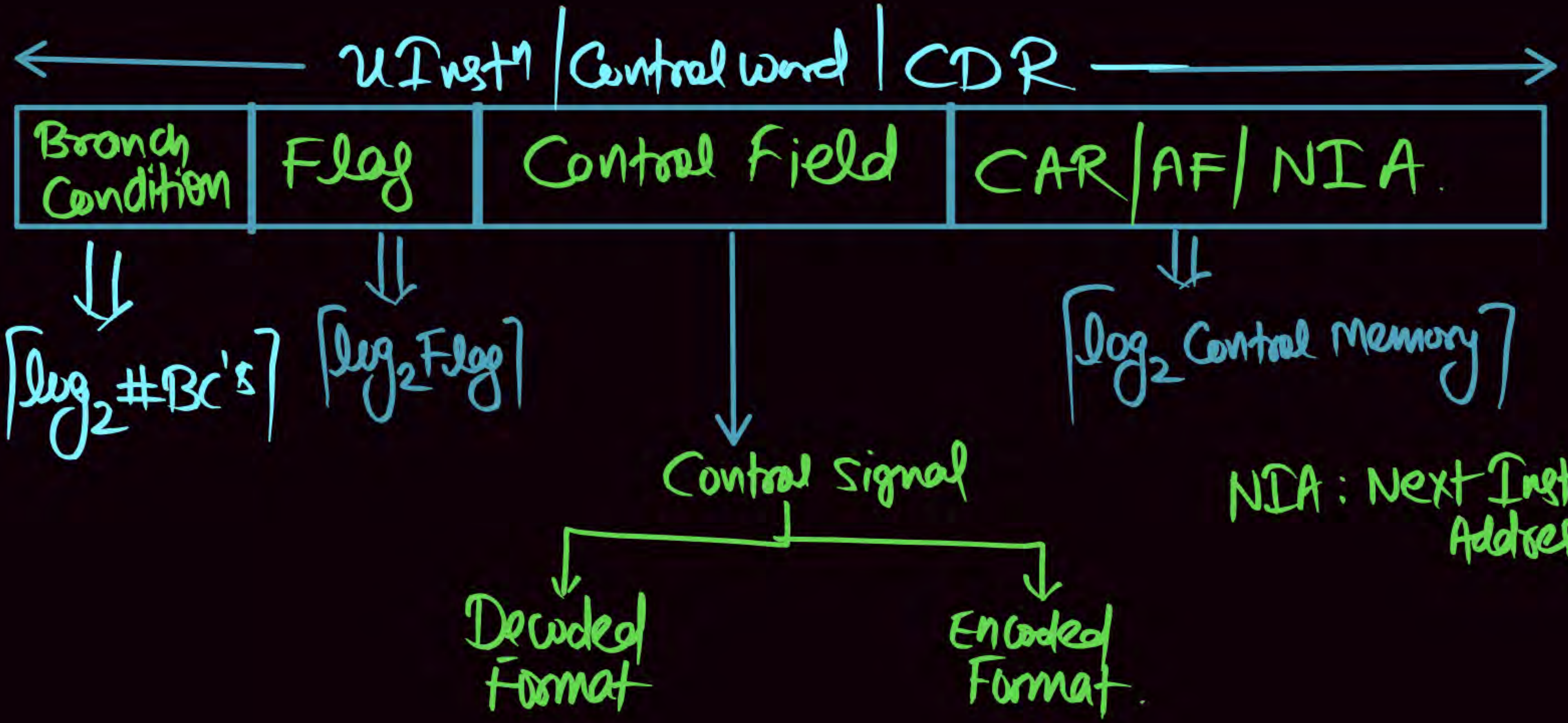


- The Set of Micro Instruction Stored in the Control Memory.
- The [CAR] Control Address Contain the Address of Next μInst'. Micro Instruction to be Read.
- The a micro Instruction is Read from the Control Memory & its transfered to Control Buffer Register.
- Control Buffer Register is Connected to the Control Line.

So in this Micro Instn Execution occur -

Instruction format ⇒ IR

Now we will see Micro Instruction Format.

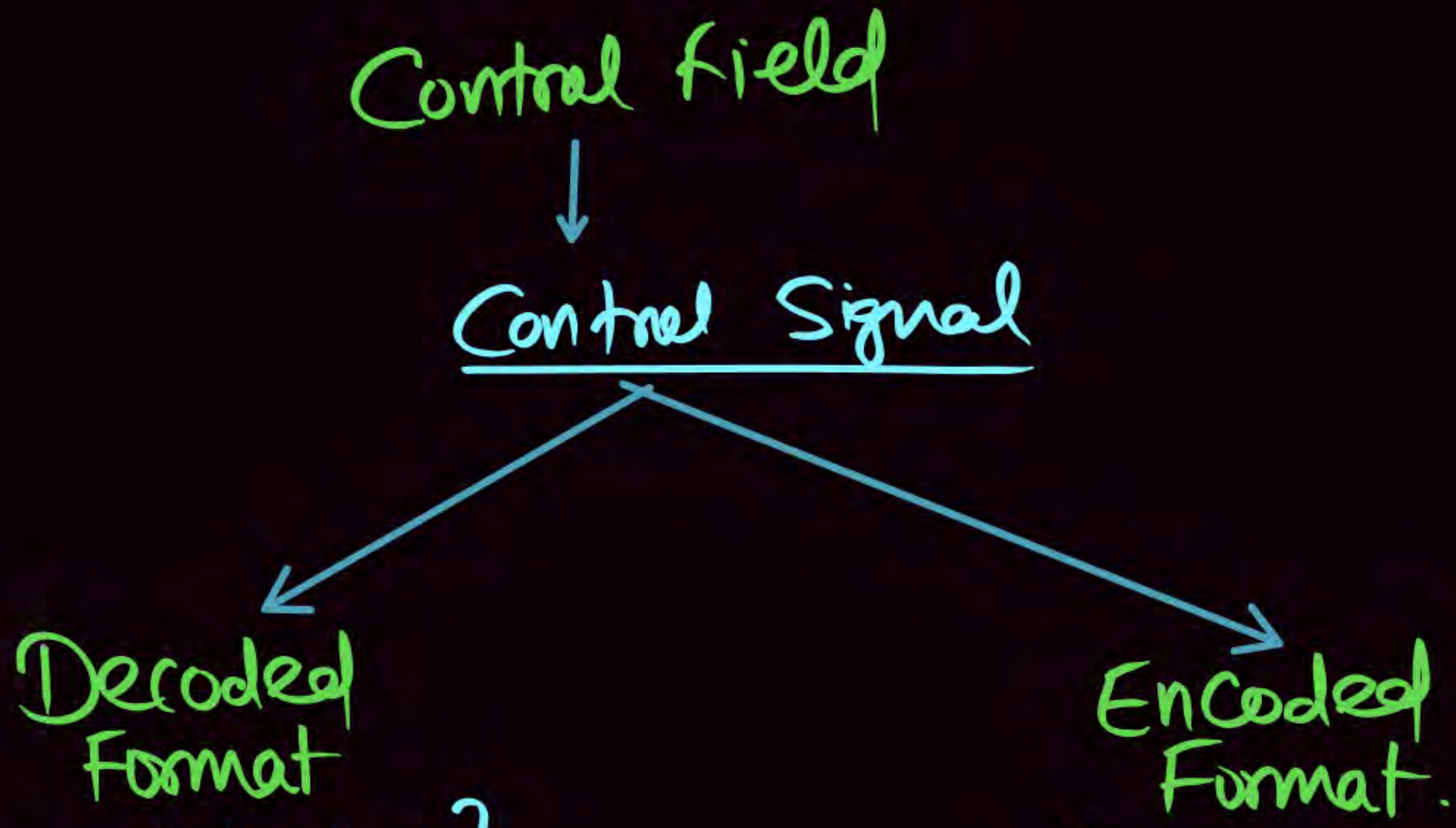


UInstn | Control word | CDR
Branch Condition
Flag
Control Field
CAR | AF | NIA
$\lceil \log_2 \#BC's \rceil$
$\lceil \log_2 Flag \rceil$
$\lceil \log_2 \text{Control Memory} \rceil$
Control Signal
Decoded Format
Encoded Format

eg 4BC

Z
NZ
C
NC

NIA : Next Instn Address.



[Horizontal µprog]

$$\left[\begin{array}{l} NCS \Rightarrow N\,bit \\ n\,bit \Rightarrow n\,CS \end{array}\right]$$

$$\left[\begin{array}{l} 8CS \Rightarrow 8\,bit \\ 3bit \Rightarrow 3CS \end{array}\right]$$

{Vertical µprog.}

$$\left[\begin{array}{l} NCS \Rightarrow \lceil \log_2 N \rceil\,bit \\ n\,bit \Rightarrow 2^n\,CS \end{array}\right]$$

$$\left[\begin{array}{l} 8CS \Rightarrow \lceil \log_2 8 \rceil = 3\,bit \\ 3bit \Rightarrow 2^3 = 8CS \end{array}\right]$$

Control Signal

① Hardwired CU Design

- S.O.P expression
- Logic function
- Fastest (RISC)
- Not flexible

② Micro programmed CU

(microprogram Stored in Control
Memory)

Control Signal

(1bit / 1CS)

① Decoded Format

Horizontal Prog

100CS ⇒ 100 bit

16bit ⇒ 16CS

② Encoded Format

Vertical Prog

100CS ⇒ $\lceil \log_2 100 \rceil = 7$bit

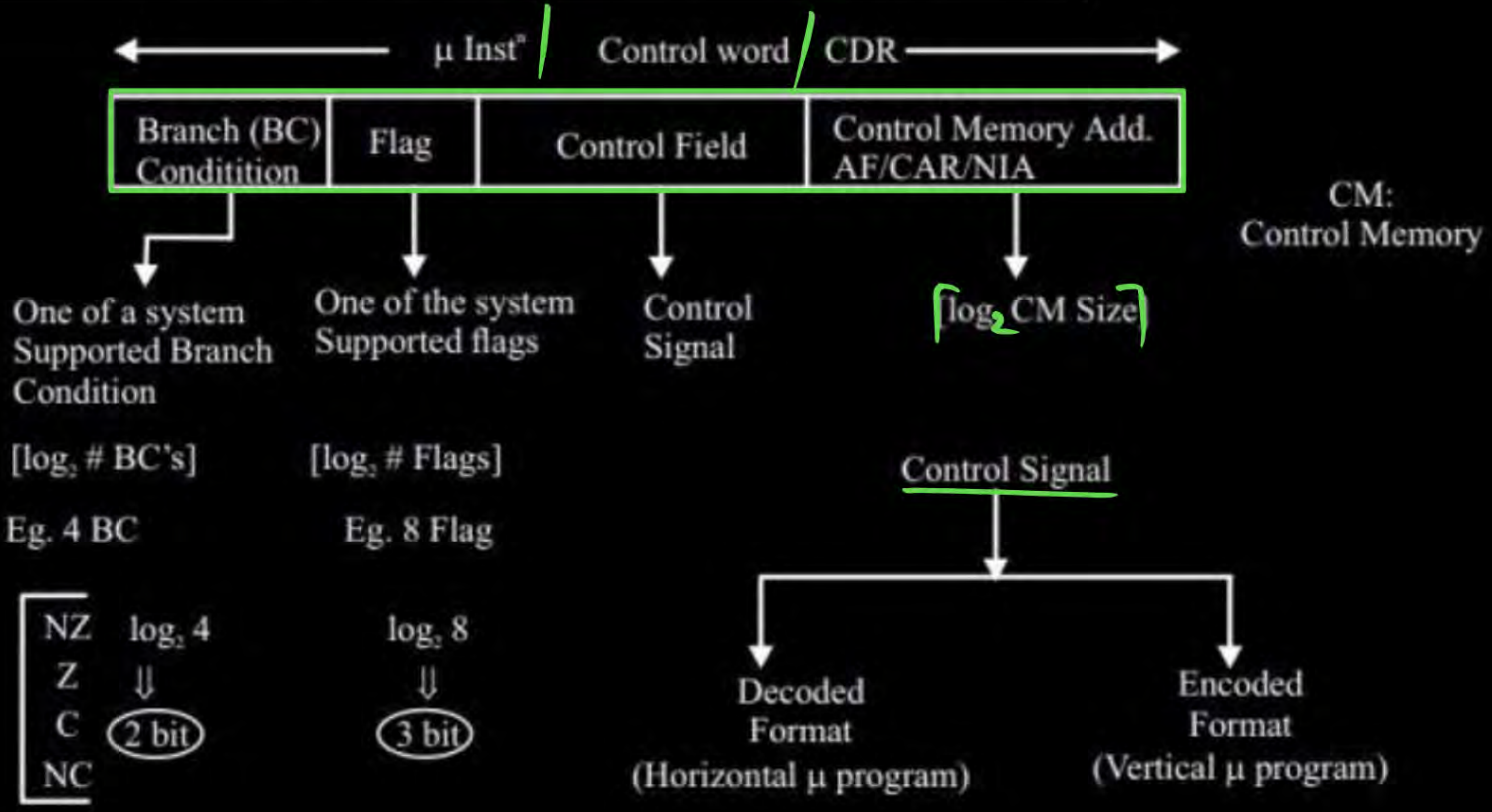
16bit ⇒ 2^{16}CS ⇒ 64K CS

Micro Instruction are Imblemented by 2 Abbroach.

1. Horizontal u programming

2. Vertical u programming.

MICRO INSTRUCTION FORMAT



$\longleftarrow$ μ Inst^n / Control word / CDR $\longrightarrow$

Branch (BC) Condidition	Flag	Control Field	Control Memory Add. AF/CAR/NIA

CM: Control Memory

One of a system Supported Branch Condition

$[\log_2 \# \text{BC's}]$

Eg. 4 BC

NZ
Z
C
NC

$\log_2 4$

$\Downarrow$

$\boxed{2\text{ bit}}$

One of the system Supported flags

$[\log_2 \# \text{Flags}]$

Eg. 8 Flag

$\log_2 8$

$\Downarrow$

$\boxed{3\text{ bit}}$

Control Signal

$\lceil \log_2 \text{CM Size} \rceil$

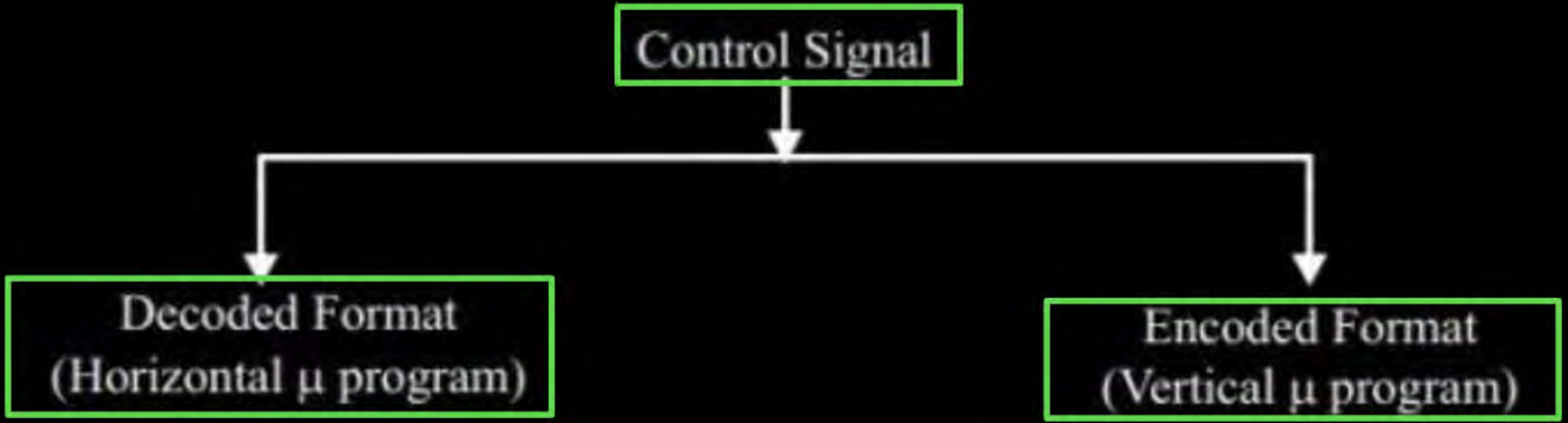
Control Signal

Decoded Format (Horizontal μ program)

Encoded Format (Vertical μ program)

① Horizontal μ programming.

② Vertical μ programming.



Control Signal

Decoded Format (Horizontal μ program)

Encoded Format (Vertical μ program)

[1 bit / 1CS]
OR
[NCS / N bits]

Eg 3bits ⇒ 3C.S
or
8 C.S ⇒ 8 bit

n bit $= 2^n$ C.S
NCS $= \log_2 N$ bits
⇓
N control Signal

3 bits ⇒ 2^3 ⇒ 8C.S
or
8C.S ⇒ 3bit

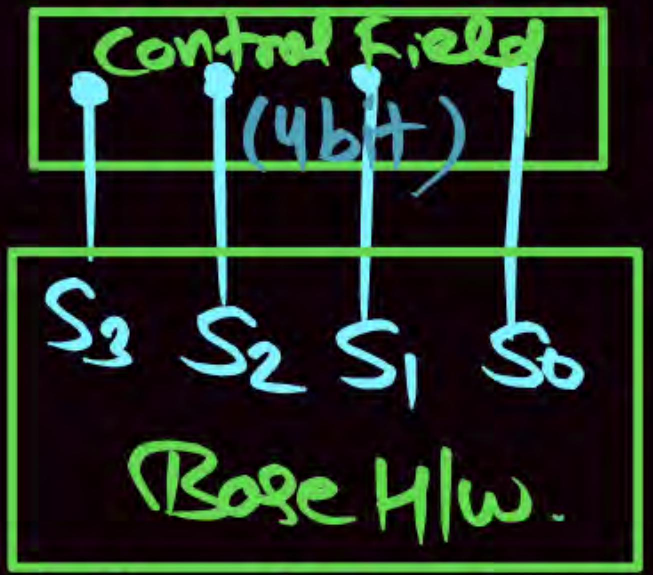
Ⓘ Horizontal Programming:

① Number(#) of Control in the Hardware = [$S_0\ S_1\ S_2\ S_3$].

(ii) Decoded for CS ⇒

Enable = 1

Disable = 0



4 CS ⇒ then

$\boxed{C.F = 4\text{bit}}$

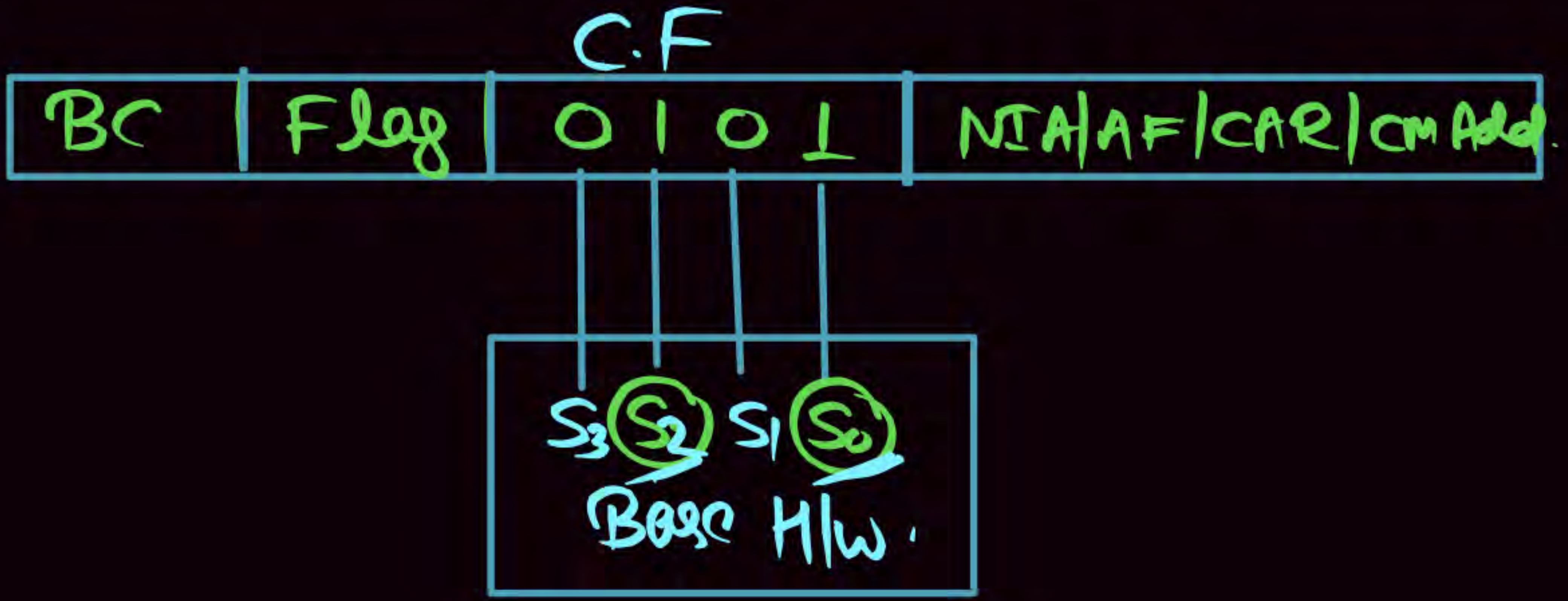
S_3	S_2	S_1	S_0
0	1	0	1

(iii) Design a Horizontal µ Instⁿ for C.S = ($\underline{S_0}$ & $\underline{S_2}$)

S_2 S_0

BC	Flag	0101	CMAdd	CAR

(iv) Operational State:



② vertical Microprogrammed CU:-

(i) Number (#) of Control Signal in H/w = $(S_0\ S_1\ S_2\ S_3)$

(ii) Encoded form of CS ⇒ 4CS ⇒ $\lceil \log_2 4 \rceil = 2$ bit

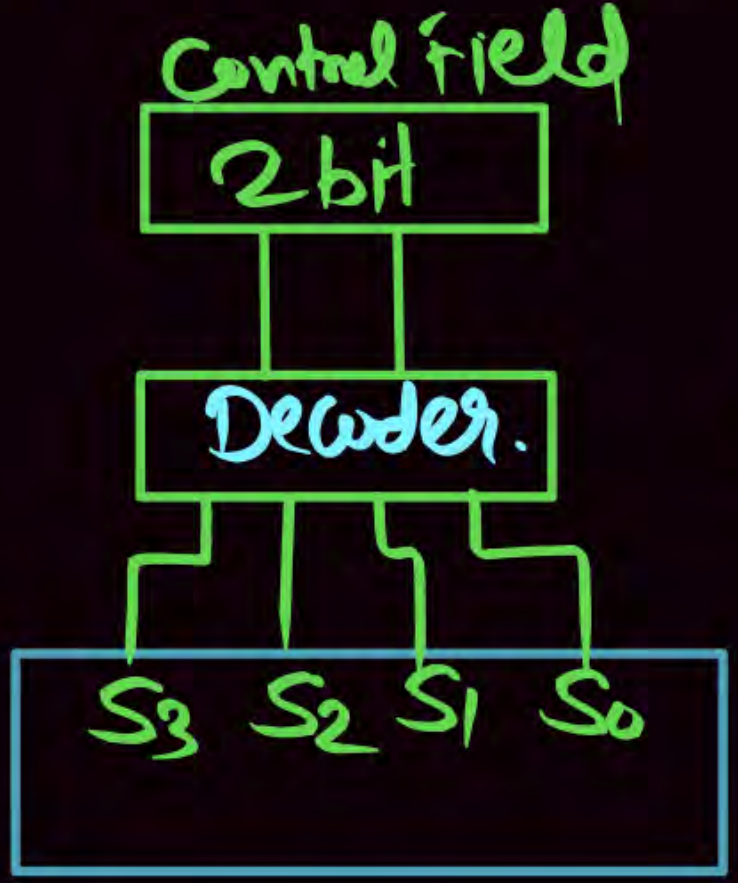
$\boxed{\text{Control field} = 2\text{bit}}$

00 - S_0

01 - S_1

10 - S_2

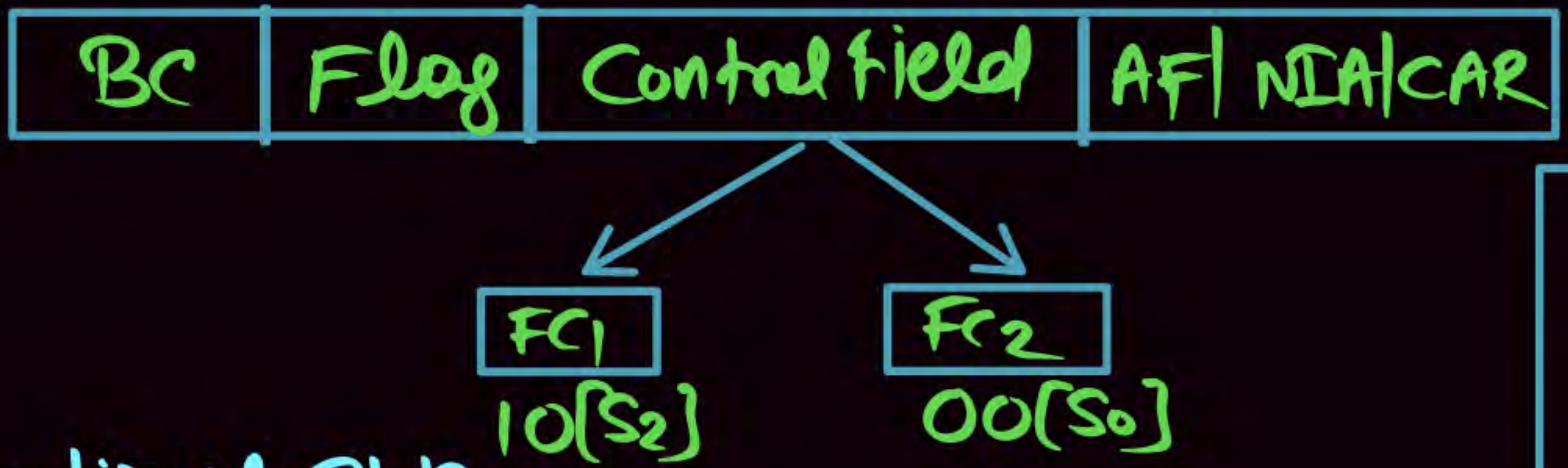
11 → S_3



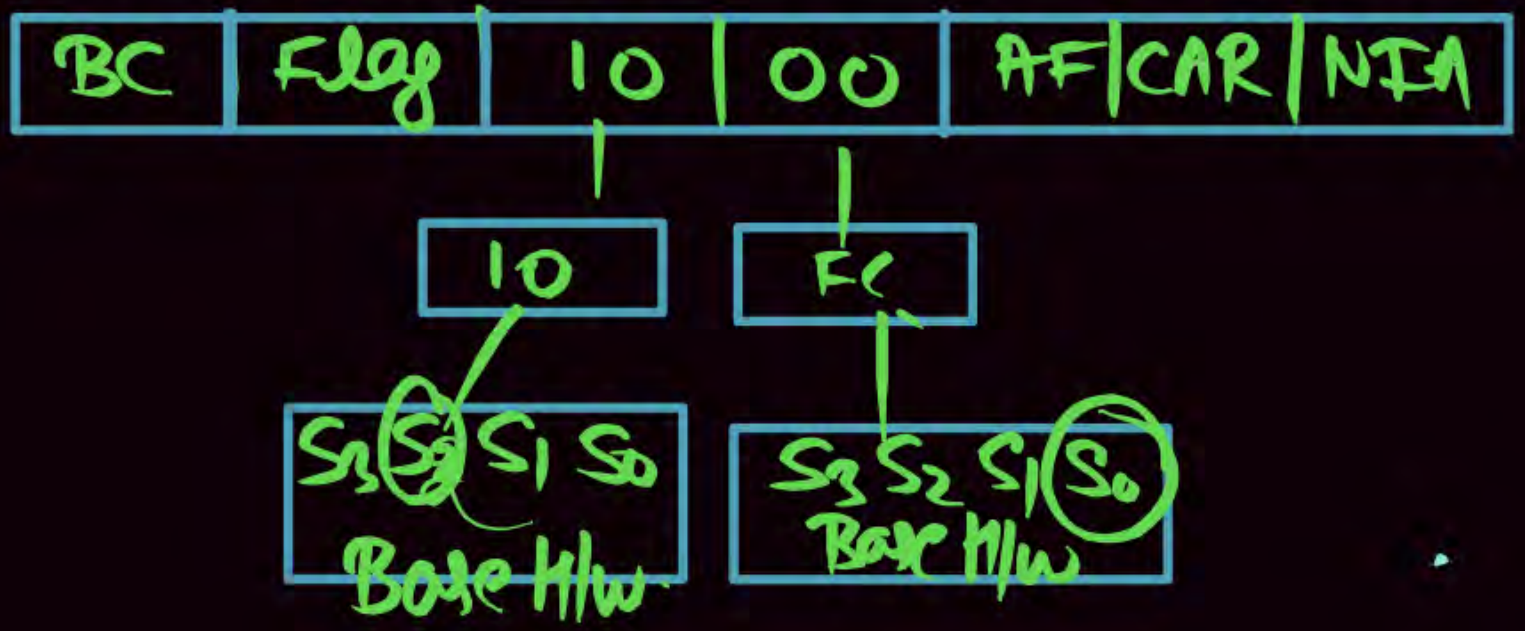
Note

∴ External Decoder is Required.

(iii) Design a vertical Micro Instⁿ for CS = [S0 S2].



(iv) operational State.



Function Code [FC] generated by the Control Unit to give the Signal to CPU for perform operation.

Horizontal Programming	vertical Programming.
① In this Control Signal are expressed in a Decode format	① In This Control signal are expressed in Encoded format
② $1\text{ bit} \Rightarrow 1\text{ CS}$ $N\text{CS} \Rightarrow N\text{ bit}$	② $n\text{ bit} \Rightarrow 2^n\text{ CS}$. $N\text{CS} \Rightarrow \lceil \log_2 N \rceil\text{ bits}$
③ It Support Longer Control Words.	③ It Support Shorter Control Word.
eg for 200 Control Signal We Required 200 bit in Control Field.	eg for 200 Control signal we Required 8 bit $\lceil \log_2 200 \rceil$ or 2^n in Control Field.
④ No External Decoder is Required to Generate the Control Signal	④ External Decoder is Required to generate the Control Signal.

Horizontal Programming	vertical Programming.
⑤ It is flexible compare to Hardwired CU.	⑤ It is More flexible.
⑥ It Support High Degree of Parallelism (None | More than 1)	⑥ It Support Low Degree of Parallelism. (None | One)

Note Default Micro Programmed CU is Vertical uprog CU. [CISC].

V.V.Imp.

Speed : Hardwired > Horizontal > Vertical

Time Consume : Vertical > Horizontal > Hardwired

Q.

GATE

Arrange the following configuration for CPU in decreasing order of operating speeds: Hardwired control, vertical micro-programming, horizontal micro-programming.

(a) Hardwired control, vertical micro programming, horizontal micro programming.

[✓ (b) Hardwired control, horizontal micro programming, vertical microprogramming]

(c) Horizontal micro programming, vertical micro programming. Hardwired control

(d) Vertical micro programming, horizontal micro programming, hardwired control

Q. GATE

Horizontal microprogramming.

✓ (a) does not require use of signal decoders

✓ (b) Results in larger sized microinstructions than vertical microprogramming

✓ (c) use one bit each control signal

✓ (d) All of the above

Ans (D).

Q.1

An instruction set of a processor has 125 signals which can be divided into 5 groups of mutually exclusive signals as follows:

Group 1 : 20 signals. Group 2 : 70 signals, Groups 3 : 2 signals. Groups 4 : 10 signals, Groups 5 : 23 signals.

How many bits of the control words can be saved by using vertical microprogramming over horizontal microprogramming?

(a) 0

(b) 103

(c) 22

(d) 55

Q.2

Consider a Hypothetical CU it has 1024 control words memory. Is support 48 control signals & 16 Flags. What is the size of the control words in bits & control memory in byte using

(i) Horizonal Programming?

(ii) Vertical Programming?

Q.3

Control field of a micro Instruction support 2 groups of control signal in which Group 1 Indicate None/One of 400 control signal & Group 2 (Horizontal) Indicate 6 signals. Hardwire contain 16 Flags & 32 Branch condition.

If CAR Register size is 20 bit then what is CDR in bits & control memory in bits?

Q.4

Consider a CPU where all the instructions require 7 clock cycles to complete execution. There are 140 instructions in the instruction set. It is found that 125 control signals are needed to be generated by the control unit. While designing the horizontal micro-programming control unit, single address field format is used for branch control logic. What is the minimum size of the control word and control address register?

[GATE IT 2008]

(a) 125, 7

(b) 125, 10

(c) 135, 7

(d) 135, 10

Q.5

A micro program control unit is required to generate a total of 25 control signals. Assume that during any microinstruction at most tow control signals are active. Minimum number of bits required in the control word to generate the required control signal.

[GATE CSE 1996]

(a) 2

(b) 2.5

(c) 10

(d) 12

Q.6

A Hypothetical processor contain word length of 12 bits. It support 1 word opcode. Each Instruction takes 8 cycles to compete the execution. Processor contain 256 control signal & 16 flags. Processor used Horizontal Control Unit. 64 Branch is used then what is the size of CAR & CDR in bits of control memory?

Q.

RISC Reduced Instruction set computer	CISC Complex Instruction set computer
1. It support less number of addressing Mode (AM)	1. It support more number of AM.
2. It support smaller Instruction set	2. It support larger Instruction set.
3. It support more number of Register	3. It support less number of Register
4. It support fixed length Instruction	4. It support variable length Instruction
5. It support 1 Instruction per cycle (CPI=1) (Cycle per Instruction =1)	5. It support number 1 Instruction Per cycle (CPI + 1)
6. It support pipeline successfully	6. It support unsuccessful Pipeline
7. It is the expensive processor used in Real Time application	7. It is the low expensive processor
8. It is a super computer	8. General Purpose computer
9. It uses hardwired control unit. (Motorola processor, power processer, ARM processor)	9. It uses microprogrammed (vertical) control unit (Pentium processer)

THANK YOU!