Micro Programmed Control

In Micro programmed ctrl, generates ctrl signals by using a program similar to machine language ~~Pgms~~ Programs.

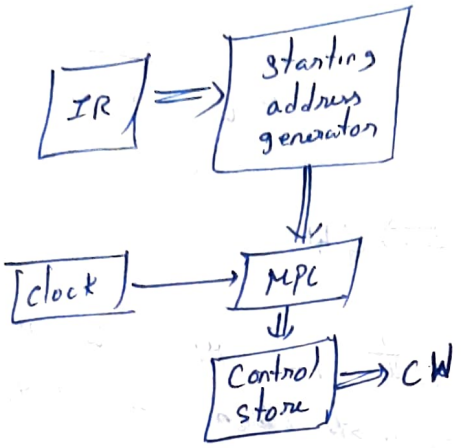

Fig 7.16
Basic organization of a micro programmed Control unit.

* Control Word (CW) ⇒ is a word, whose individual bit represent various control signals.

Each of the ctrl step in ctrl sequence is an unique instruction with combination of 1's & 0's. It is generated based on micro instructions / microroutine.

* Control store ⇒ microroutines of all instructions are stored in ctrl store.

* μPC (Micro-Program Counter) ⇒ To read CW sequentially from ctrl store μPC is used.

eg

Addrn	Microinstruction
0	PCout, MARin, Read, Select 4, Add, Zin
1	Zout, PCin, Yin, WMFC
2	MDRout, IRin
3	Branch to starting addrn of appropriate microroutine
...	...
25	If N=0, then branch to microinstruction 0
26	offset field-of-IRout, Select 4, Add, Zin
27	Zout, PCin, End

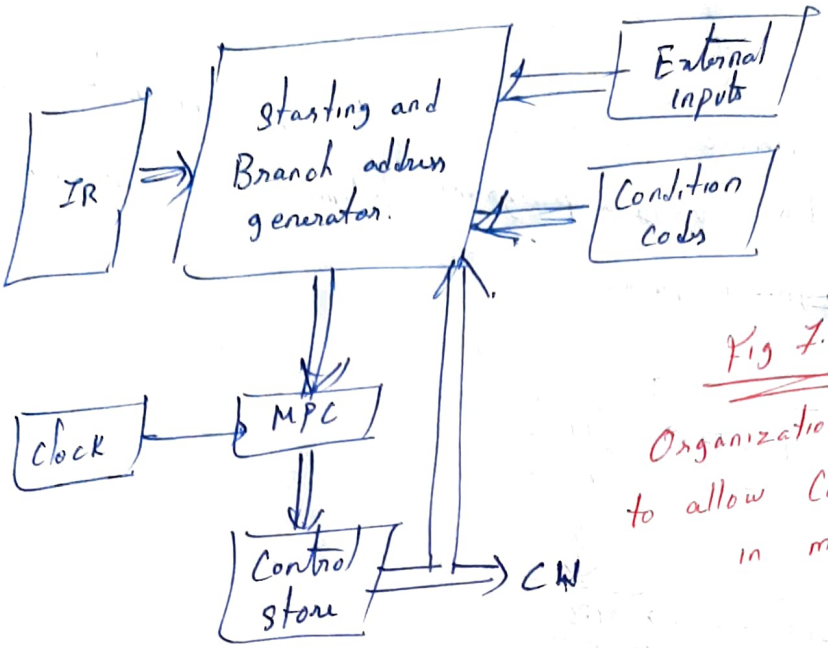

Fig 7.18
Organization of ctrl unit to allow Conditional branching in microprogram.

In this control unit, the μPC is incremented every time a new micro instruction is fetched from the micro program memory, except in the following situations -

1) When a new instruction is loaded into the IR, the μPC is loaded with the starting address of the microroutine for that instruction.

2) When a Branch microinstruction is encountered and the branch condition is satisfied, the μPC is loaded with the branch address.

3) When an End micro instruction is encountered, the μPC is loaded with the address of the first CW in the micro routine for the instruction fetch cycle.

Micro Instruction Organization -

* Highly encoded schemes use compact codes to specify only small number of ctrl functions in micro instructions are referred as Vertical Organization
* Minimally encoded schemes, control many resources in a single micro instruction is called as Horizontal Organization

≡ μBranch ≡ MAR (Micro instruction Address Register)
↓Loads Next address field in each micro instruction

Micro-Programmed Control

Step	Action
1	PC_{out}, MAR_{in}, Read, Select4, Add, Z_{in}
2	Z_{out}, PC_{in}, Y_{in}, WMFC
3	MDR_{out}, IR_{in}
4	$R3_{out}$, MAR_{in}, Read
5	$R1_{out}$, Y_{in}, WMFC
6	MDR_{out}, SelectY, Add, Z_{in}
7	Z_{out}, $R1_{in}$, End

Figure 7.6 Control sequence for execution of the instruction Add (R3),R1.

Micro-instruction	::	PC_{in}	PC_{out}	MAR_{in}	Read	MDR_{out}	IR_{in}	Y_{in}	Select	Add	Z_{in}	Z_{out}	$R1_{out}$	$R1_{in}$	$R3_{out}$	WMFC	End	::
1		0	1	1	1	0	0	0	1	1	1	0	0	0	0	0	0	
2		1	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	
3		0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	
4		0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	
5		0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	
6		0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	
7		0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	

Figure 7.15 An example of microinstructions for Figure 7.6.