HARDWIRED CONTROL UNIT

- Here control signals are produced by hardware.
- There are three types of Hardwired Control Units

STATE TABLE METHOD

1) It is the most basic type of hardwired control unit.
2) Here the behavior of the control unit is represented in the form of a table called the state table.
3) The rows represent the T-states and the columns indicate the instructions.
4) Each intersection indicates the control signal to be produced, in the corresponding T-state of every instruction.
5) A circuit is then constructed based on every column of this table, for each instruction.

T-STATES	INSTRUCTIONS			
	I_1	I_2	…	I_N
T_1	$Z_{1,1}$	$Z_{1,2}$	…	$Z_{1,N}$
T_2	$Z_{2,1}$	$Z_{2,2}$	…	$Z_{2,N}$
…	…	…	…	…
T_M	$Z_{M,1}$	$Z_{M,2}$	…	$Z_{M,N}$

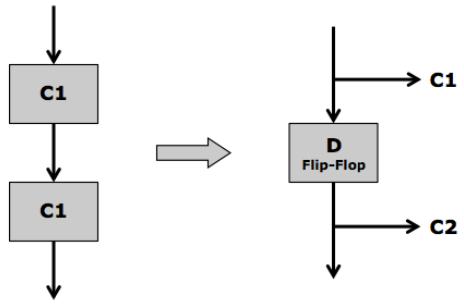
$Z_{1,1}$: Control Signal to be produced in T-state (T_1) of Instruction (I_1)

ADVANTAGE:
It is the simplest method and is ideally suited for very small instruction sets.

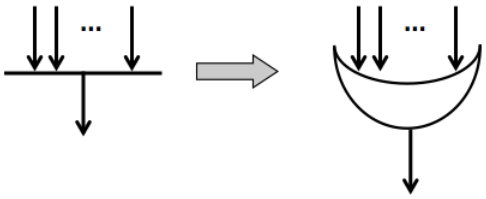
DRAWBACK:
As the number of instructions increase, the circuit becomes bigger and hence more complicated.
As a tabular approach is used, instead of a logical approach (flowchart), there are duplications of many circuit elements in various instructions.

DELAY ELEMENT METHOD

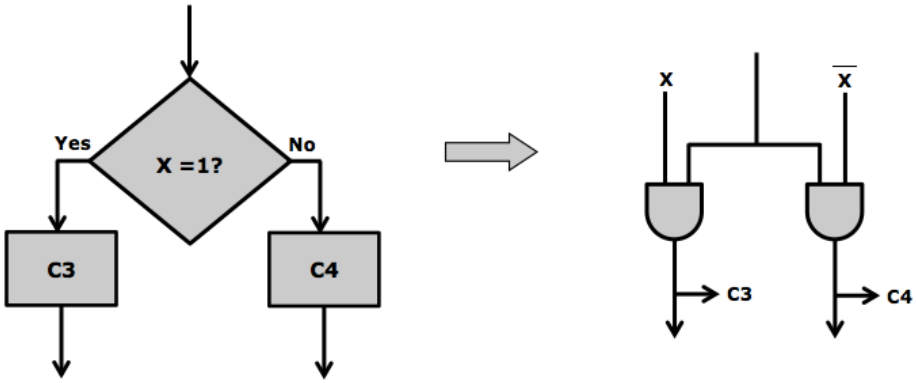
1) Here the behavior of the control unit is represented in the form of a flowchart.
2) Each step in the flowchart represents a control signal to be produced.
3) Once all steps of a particular instruction, are performed, the complete instruction gets executed.
4) Control signals perform Micro-Operations, which require one T-states each.
5) Hence between every two steps of the flowchart, there must be a delay element.
6) The delay must be exactly of one T-state. This delay is achieved by D Flip-Flops.
7) These **D Flip-Flops** are **inserted** between every two **consecutive control signals**.



8) Of all D Flip-Flops only one will be active at a time. So the method is also called **"One Hot Method"**.
9) In a **multiple entry point**, to combine two or more paths, we use an **OR gate**.



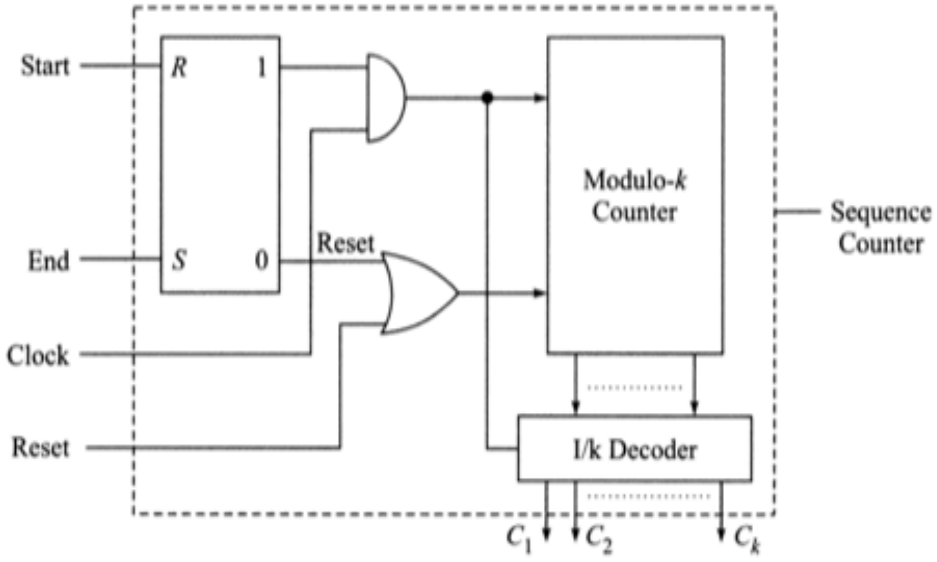
10) A **decision box** is replaced by a set of two complementing **AND gates**



ADVANTAGE:
As the method has a logical approach, it can **reduce the circuit complexity**.
This is done by re-utilizing common elements between various instructions.

DRAWBACK:
As the no of instructions increase, the number of **D Flip-Flops increase**, so the **cost increases**.
Moreover, **only one of those D Flip-Flops are actually active at a time.**

SEQUENCE COUNTER METHOD



1) This is the **most popular** form of hardwired control unit.
2) It follows the same **logical approach of a flowchart**, like the Delay element method, but does not use all those unnecessary D Flip-Flops.
3) First a flowchart is made representing the behavior of a control unit.
4) It is then **converted into a circuit** using the same principle of AND & OR gates.
5) We need a **delay of 1 T-state** (one clock cycle) between every two **consecutive control signals**.
6) That is achieved by the above circuit.
7) If there are **"k" number of distinct steps** producing control signals, we employ a **"mod k"** and **"k" output decoder**.
8) The counter will **start counting** at the beginning of the instruction.
9) The "clock" input via an AND gate ensures each count will be generated **after 1 T-state**.
10) The count is given to the **decoder** which **triggers the generation of "k" control signals**, each after a delay of 1 T-state.
11) When the **instruction ends**, the **counter is reset** so that next time, it begins from the first count.

ADVANTAGE:

Avoids the use of **too many D Flip-Flops**.

GENERAL DRAWBACKS OF A HARDWIRED CONTROL UNIT

1) Since they are based on hardware, as the instruction set increases, the **circuit becomes more and more complex**. For modern processors having hundreds of instructions, it is virtually impossible to create Hardwired Control Units.
2) Such large circuits are **very difficult to debug**.
3) As the **processor gets upgraded**, the entire Control Unit has to be **redesigned**, due to the **rigid nature** of hardware design.