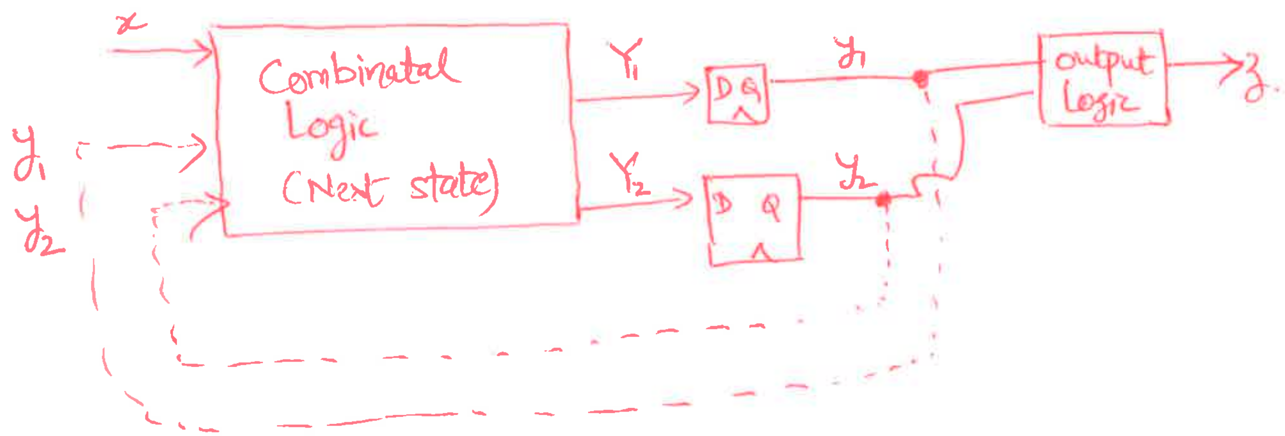
Finite State Machines (FSMs)

We study two types of FSMs.

(1) Moore m/c & (2) Mealy m/c.

The general structure of a Moore m/c.



Moore m/c. = $M = (\Sigma, O, S, S^0, \delta, \lambda)$.

Σ = input label, O = output alphabet

S = set of states, $S^0 \subset S$ = initial/reset state

$\delta : S \times \Sigma \longrightarrow S$ transition function

$\lambda : S \longrightarrow O$ output function.

$\{Y_1, Y_2 \ldots\}$ = Next state signals (output of next state logic)

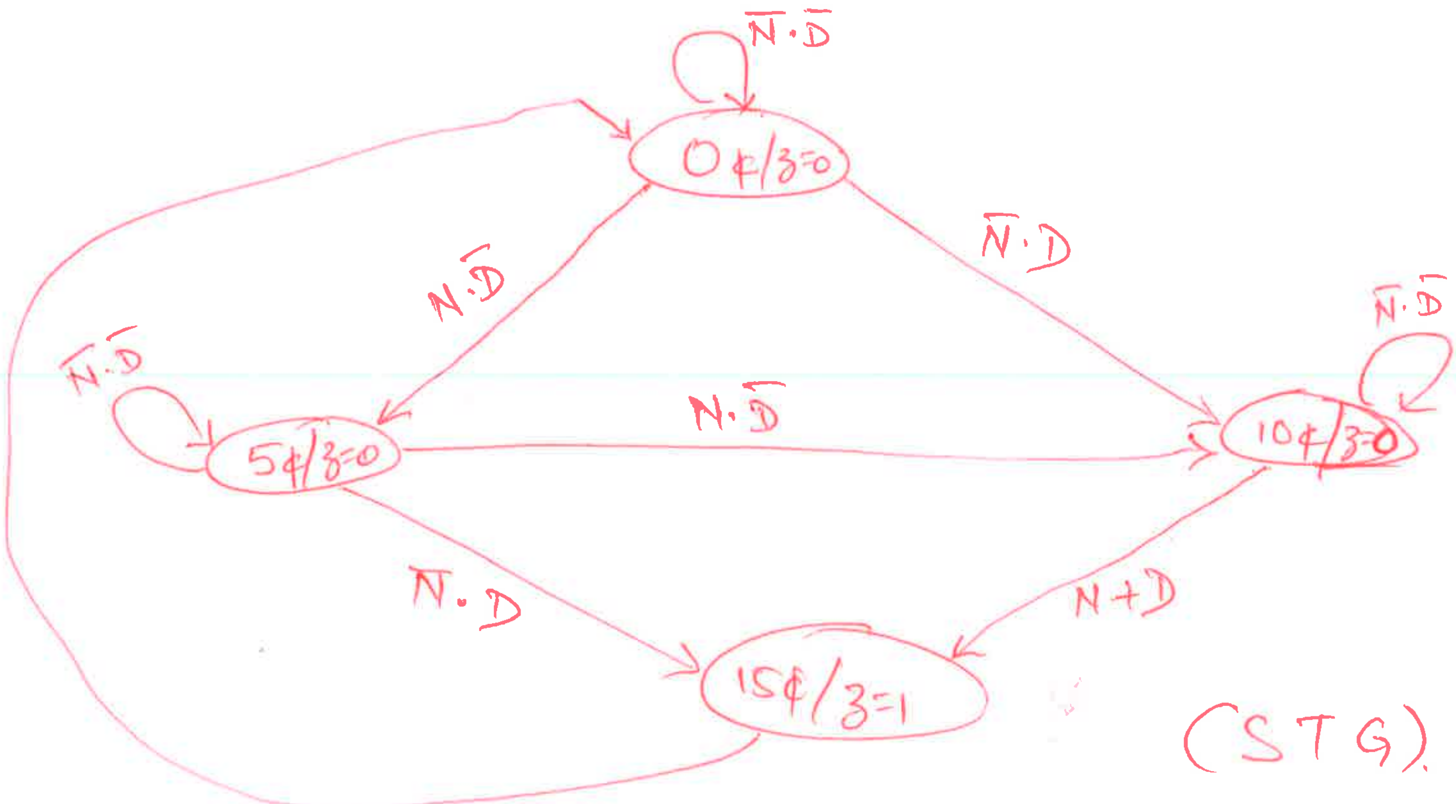
$\{y_1, y_2 \ldots\}$ = Present state signals (input to next state logic)

Next state = function of present-state & inputs.

Output = " " present state only.

Moore FSM Design example: ②

Vending m/c controllers, takes Nickels (N) & Dimes (D) as inputs, output $z=1$ (dispenser) when 15¢ are deposited.



(STG)

State encoding: 4 states = 2 FFs. (Y_1, y_1) (Y_2, y_2)

0¢ = 00, 5¢ = 01, 10¢ = 10, 15¢ = 11

State Table.

P.S. $(y_1 y_2)$	N.S. $(Y_1 Y_2)$ $\bar{N}\cdot\bar{D}$	$\bar{N}D$	$N\bar{D}$	ND	Output (z)
0¢ = 00	00	10	01	x	0
5¢ = 01	01	11	10	x	0
10¢ = 10	10	11	11	x	0
15¢ = 11	00	00	00	x	1

Assumption: N & D cannot be input at the same time as there is only 1 coin-slot. Also, no change is given by m/c.

Logic for Y_1

y_1y_2 \ ND	00	01	11	10
00		1	X	
01		1	X	1
11		0	X	
10	1	1	X	1

$$Y_1 = y_1\bar{y}_2 + \bar{y}_1 D + \bar{y}_1 y_2 N$$

Y_2

y_1y_2 \ ND	00	01	11	10
00			X	1
01	1	1	X	
11			X	
10		1	X	1

$$Y_2 = \bar{y}_1\bar{y}_2 N + \bar{y}_1 y_2 \bar{N} + y_1\bar{y}_2 D + y_1\bar{y}_2 N$$

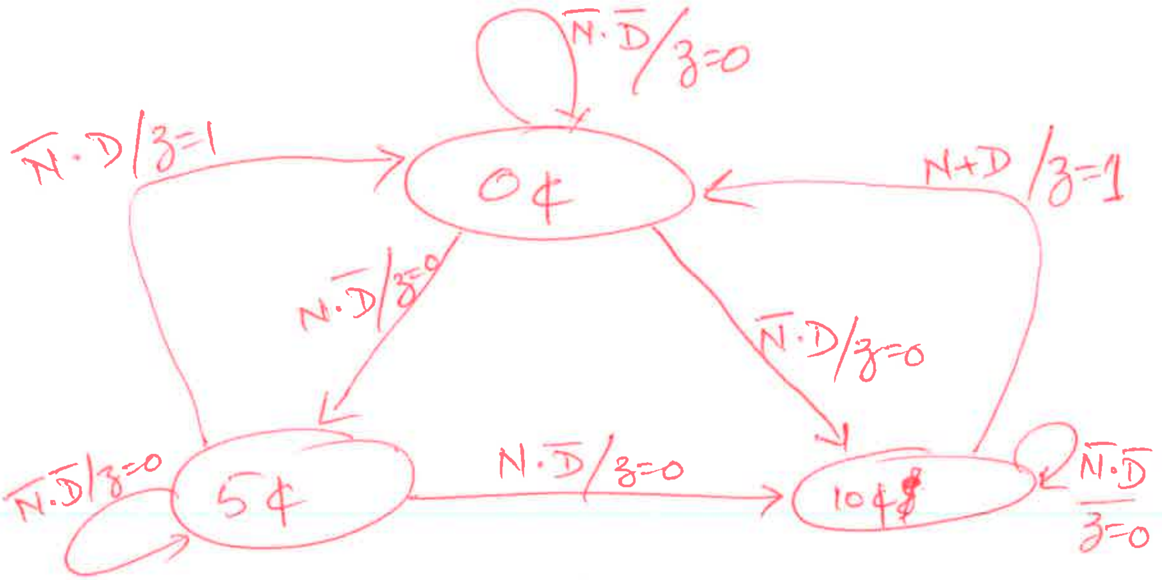
output $$z = Y_1 \cdot Y_2$$

The same m/c behaviour can be designed using a Mealy machine, by making the output depend ~~on~~ on both present states and present inputs.

May result in a reduction of the number of states, but also may complicate the output logic.

For the same vending machine controllers.

Mealy



Encoding choice 0¢ = 11, 5¢ = 00, 10¢ = 01

NS. (Y_1Y_2), z.

P.S. ($y_1 y_2$)	$\bar{N}\cdot\bar{D}$	$\bar{N}\cdot D$	$N\,\bar{D}$	$N\cdot D$
0¢ = 11	11, 0	01, 0	00, 0	X
5¢ = 00	00, 0	11, 1	01, 0	X
10¢ = 01	01, 0	11, 1	11, 1	X
11	xx, x	xx, x	xx, x	xx, x

11 → Sequential don't care ← unreachable state of the m/c.

Primary input D. C.

Next-state & output logic.

Y_1

y_1y_2 \ ND	00	01	11	10
00		1	X	
01		1	X	1
11	1		X	
10	X	X	X	X

$$Y_1 = y_1 \overline{N}\,\overline{D} + \overline{y}_1 D + \overline{y}_1 y_2 N$$

Y_2

y_1y_2 \ ND	00	01	11	10
00		1	X	
01	1	1	X	1
11	1	1	X	
10	X	X	X	X

$$Y_2 = D + y_2 \overline{N} + \overline{y}_1 N$$

z.

y_1y_2 \ ND	00	01	11	10
00		1	X	
01		1	X	1
11			X	
10	X	X	X	X

$$z = \overline{y}_1 D + \overline{y}_1 y_2 N$$

Try out yourself:

$0¢ = 00$
$5¢ = 01$
$10¢ = 10$
} encoding.

PS	$\bar{N}\bar{D}$	$\bar{N}D$	$N\bar{D}$	ND
00	00, 0	10, 0	01, 0	X
01	01, 0	00, 1	10, 0	x
10	10, 0	00, 1	00, 1	x
11	X	X	x	X

Synthesize logic for Y_1, Y_2 & z with <u>this</u> encoding & compare the SOP literal cost of implementation.

So a Mealy m/c $M = (\Sigma, O, S, S^0, \delta, \lambda)$

where $\lambda : S \times \Sigma \rightarrow O$