Fundamentals of Computer Systems

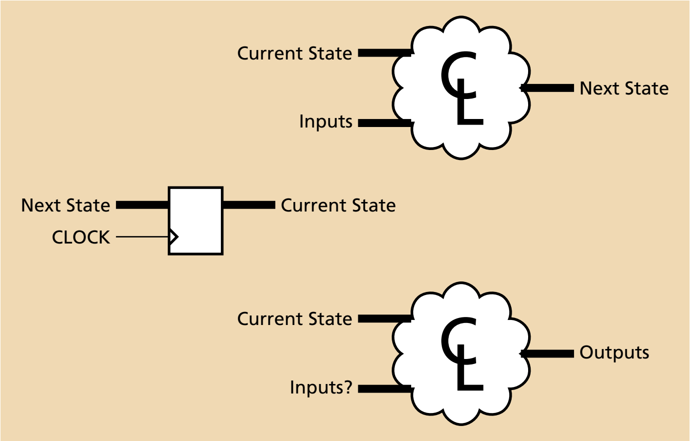
Finite State Machines

Stephen A. Edwards

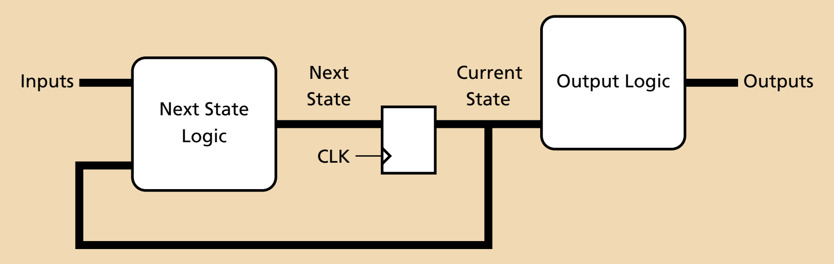
Columbia University

Summer 2020

Finite State Machine Components



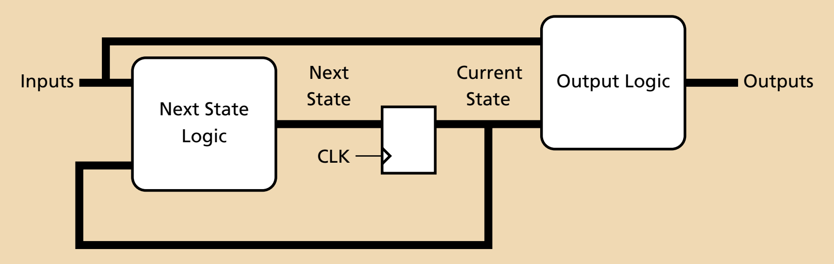
Moore and Mealy Machines



The Moore Form:

Outputs are a function of *only* the current state.

Moore and Mealy Machines

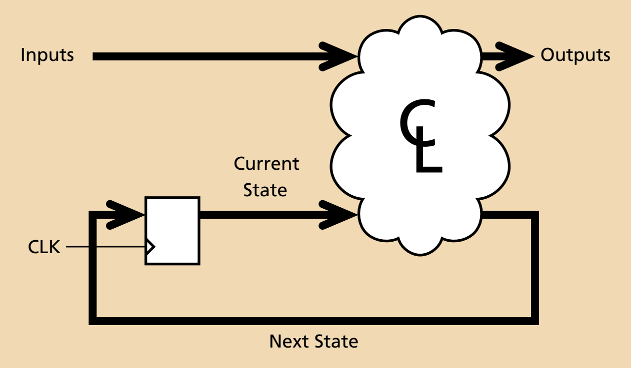


The Mealy Form:

Outputs may be a function of *both* the current state and the inputs.

A mnemonic: *Moore* machines often have *more* states.

Mealy Machines are the Most General

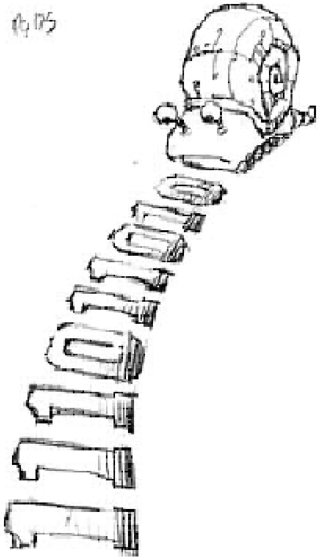


Another, equivalent way of drawing Mealy Machines

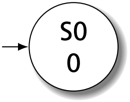
This is exactly the synchronous digital logic paradigm

Moore vs. Mealy FSMs

Alyssa P. Hacker has a snail that crawls down a paper tape with 1's and 0's on it. The snail smiles whenever the last four digits it has crawled over are 1101. Design Moore and Mealy FSMs of the snail's brain.

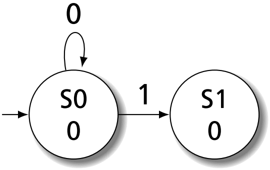


State Transition Diagrams: Looking for "1101"



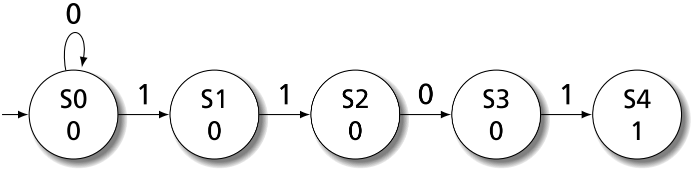
Moore Machine: States indicate output

State Transition Diagrams: Looking for “1101”



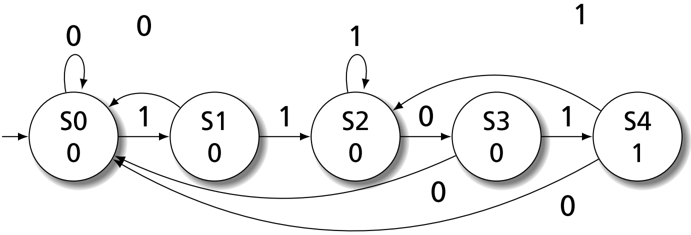
Moore Machine: States indicate output

State Transition Diagrams: Looking for "1101"



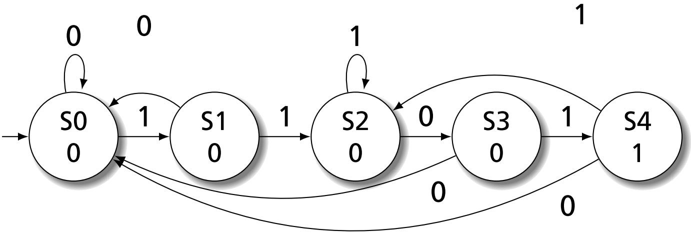
Moore Machine: States indicate output

State Transition Diagrams: Looking for “1101”

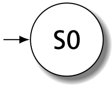


Moore Machine: States indicate output

State Transition Diagrams: Looking for "1101"

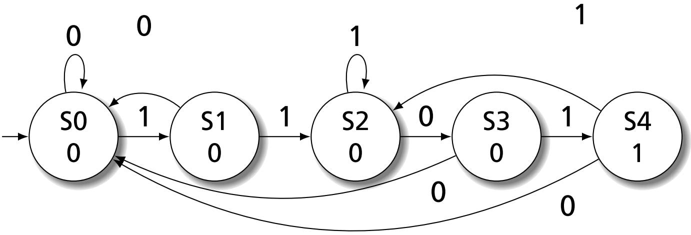


Moore Machine: States indicate output

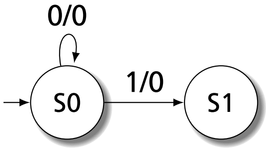


Mealy Machine: Arcs indicate input/output

State Transition Diagrams: Looking for "1101"

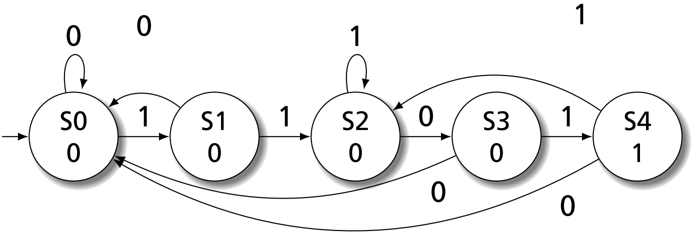


Moore Machine: States indicate output

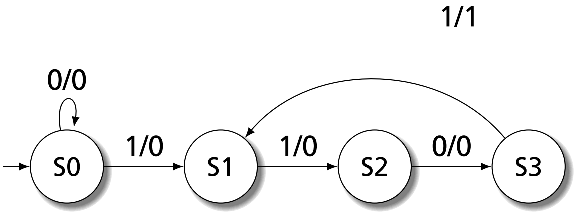


Mealy Machine: Arcs indicate input/output

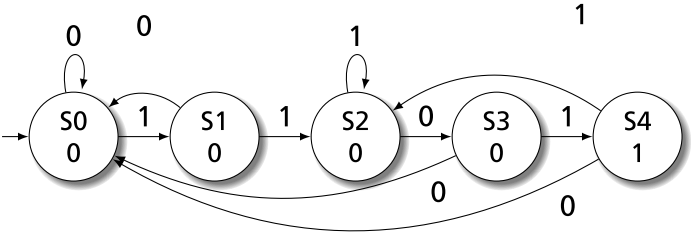
State Transition Diagrams: Looking for “1101”



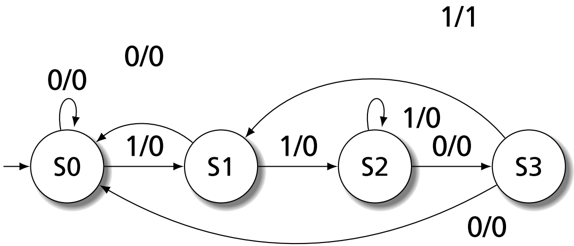
Moore Machine: States indicate output



State Transition Diagrams: Looking for "1101"



Moore Machine: States indicate output



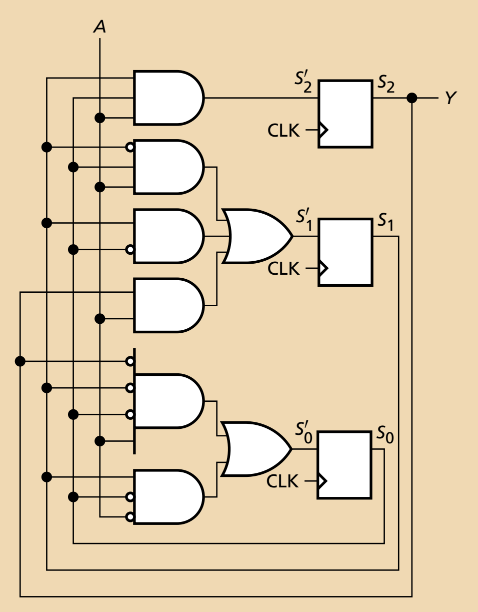
Moore Machine

Next State		
S	**A**	**S'**
S0	0	S0
S0	1	S1
S1	0	S0
S1	1	S2
S2	0	S3
S2	1	S2
S3	0	S0
S3	1	S4
S4	0	S0
S4	1	S2

Output	
S	**Y**
S0	0
S1	0
S2	0
S3	0
S4	1

Moore Machine

Next State			Output	
S	**A**	**S′**	**S**	**Y**
000	0	000	000	0
000	1	001	001	0
001	0	000	010	0
001	1	010	011	0
010	0	011	100	1
010	1	010		
011	0	000		
011	1	100		
100	0	000		
100	1	010		

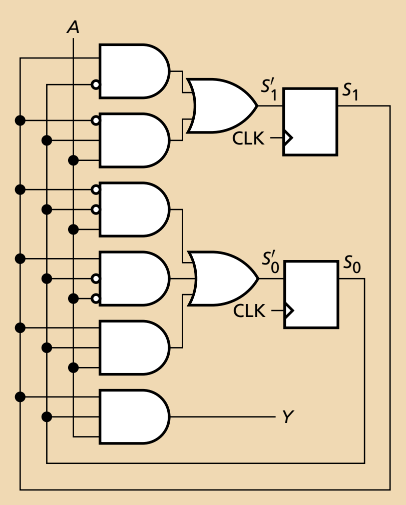


Mealy Machine

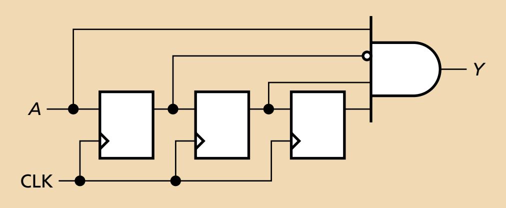
S	A	S'	Y
S0	0	S0	0
S0	1	S1	0
S1	0	S0	0
S1	1	S2	0
S2	0	S3	0
S2	1	S2	0
S3	0	S0	0
S3	1	S1	1

Mealy Machine

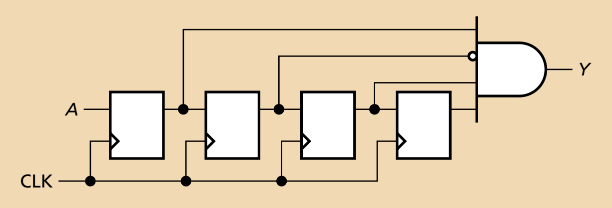
S	A	S'	Y
00	0	00	0
00	1	01	0
01	0	00	0
01	1	10	0
10	0	11	0
10	1	10	0
11	0	00	0
11	1	01	1



More Intuitive Solutions using Shift Registers

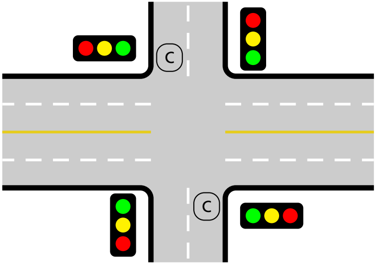


Mealy Form: Output Depends on Input Immediately



Moore Form: Output Depends Only on State

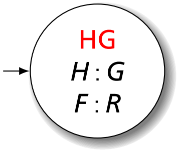
FSM Example: A Traffic Light Controller



This controls a traffic light at the intersection of a busy highway and a farm road. Normally, the highway light is green but if a sensor detects a car on the farm road, the highway light turns yellow then red. The farm road light then turns green until there are no cars or after a long timeout. Then, the farm road light turns yellow then red, and the highway light returns to green. The inputs to the machine are the car sensor, a short timeout signal, and a long timeout signal. The outputs are a timer start signal and the colors of the highway and farm road lights.

Source: Mead and Conway, *Introduction to VLSI Systems*, 1980, p. 85.

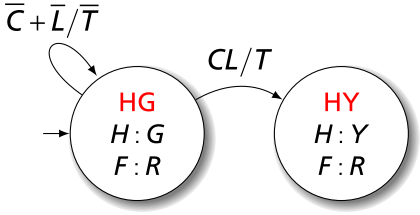
State Transition Diagram for the TLC



Inputs:
C: Car sensor
S: Short Timeout
L: Long Timeout

Outputs:
T: Timer Reset
H: Highway color
F: Farm road color

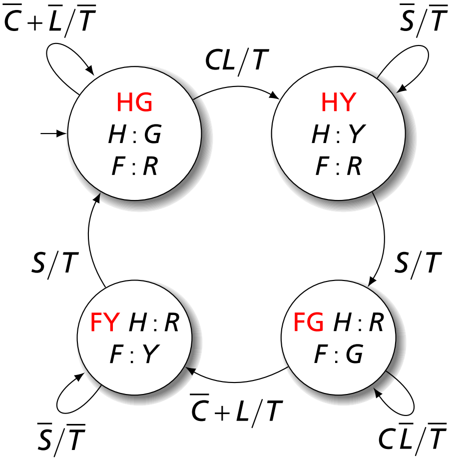
State Transition Diagram for the TLC



Inputs:
C: Car sensor
S: Short Timeout
L: Long Timeout

Outputs:
T: Timer Reset
H: Highway color
F: Farm road color

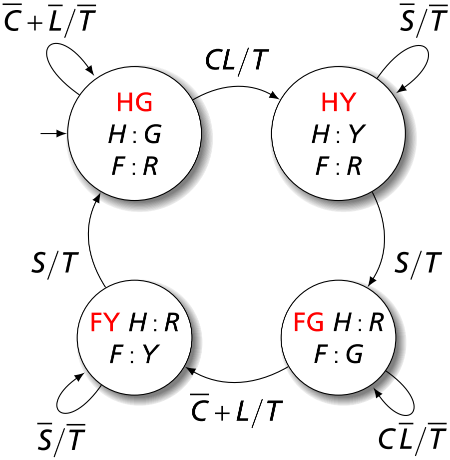
State Transition Diagram for the TLC



Inputs:
C: Car sensor
S: Short Timeout
L: Long Timeout

Outputs:
T: Timer Reset
H: Highway color
F: Farm road color

State Transition Diagram for the TLC



S	C	S	L	T	S'
HG	0	X	X	0	HG
HG	X	X	0	0	HG
HG	1	X	1	1	HY
HY	X	0	X	0	HY
HY	X	1	X	1	FG
FG	1	X	0	0	FG
FG	0	X	X	1	FY
FG	X	X	1	1	FY
FY	X	0	X	0	FY
FY	X	1	X	1	HG

S	H	F
HG	G	R
HY	Y	R
FG	R	G
FY	R	Y

Inputs:
C: Car sensor
S: Short Timeout
L: Long Timeout

Outputs:
T: Timer Reset
H: Highway color
F: Farm road color

State and Output Encoding

S	C	S	L	T	S'
HG	0	X	X	0	HG
HG	X	X	0	0	HG
HG	1	X	1	1	HY
HY	X	0	X	0	HY
HY	X	1	X	1	FG
FG	1	X	0	0	FG
FG	0	X	X	1	FY
FG	X	X	1	1	FY
FY	X	0	X	0	FY
FY	X	1	X	1	HG

S	H	F
HG	G	R
HY	Y	R
FG	R	G
FY	R	Y

First idea: use a binary encoding:

HG	00
HY	01
FG	10
FY	11

G	00
Y	01
R	10

State and Output Encoding

S	C	S	L	T	S'
00	0	X	X	0	00
00	X	X	0	0	00
00	1	X	1	1	01
01	X	0	X	0	01
01	X	1	X	1	10
10	1	X	0	0	10
10	0	X	X	1	11
10	X	X	1	1	11
11	X	0	X	0	11
11	X	1	X	1	00

S	H	F
00	00	10
01	01	10
10	10	00
11	10	01

$$T = \overline{S_1}\,\overline{S_0} C L + \overline{S_1} S_0 S + S_1 \overline{S_0} (\overline{C} + L) + S_1 S_0 S$$

$$S'_1 = \overline{S_1} S_0 S + S_1 \overline{S_0} + S_1 S_0 S$$

$$S'_0 = \overline{S_1}\,\overline{S_0} C L + \overline{S_1} S_0 \overline{S} + S_1 \overline{S_0} (\overline{C} + L) + S_1 S_0 \overline{S}$$

$$H_1 = S_1$$

$$H_0 = \overline{S_1} S_0$$

$$F_1 = \overline{S_1}$$

$$F_0 = S_1 S_0$$

State and Output Encoding

S	C	S	L	T	S'
00	0	X	X	0	00
00	X	X	0	0	00
00	1	X	1	1	01
01	X	0	X	0	01
01	X	1	X	1	10
10	1	X	0	0	10
10	0	X	X	1	11
10	X	X	1	1	11
11	X	0	X	0	11
11	X	1	X	1	00

S	H	F
00	00	10
01	01	10
10	10	00
11	10	01

$$T = \overline{S_1}\,\overline{S_0}CL + S_0 S + S_1\overline{S_0}(\overline{C} + L)$$

$$S_1' = S_0 S + S_1\overline{S_0}$$

$$S_0' = \overline{S_1}\,\overline{S_0}CL + S_0\overline{S} + S_1\overline{S_0}(\overline{C} + L)$$

$$H_1 = S_1$$

$$H_0 = \overline{S_1}S_0$$

$$F_1 = \overline{S_1}$$

$$F_0 = S_1 S_0$$

State and Output Encoding

S	C	S	L	T	S'
00	0	X	X	0	00
00	X	X	0	0	00
00	1	X	1	1	01
01	X	0	X	0	01
01	X	1	X	1	10
10	1	X	0	0	10
10	0	X	X	1	11
10	X	X	1	1	11
11	X	0	X	0	11
11	X	1	X	1	00

S	H	F
00	00	10
01	01	10
10	10	00
11	10	01

$$T = \overline{S_0}(\overline{S_1}CL + S_1(\overline{C} + L)) + S_0 S$$

$$S'_1 = S_0 S + S_1 \overline{S_0}$$

$$S'_0 = \overline{S_0}(\overline{S_1}CL + S_1(\overline{C} + L)) + S_0 \overline{S}$$

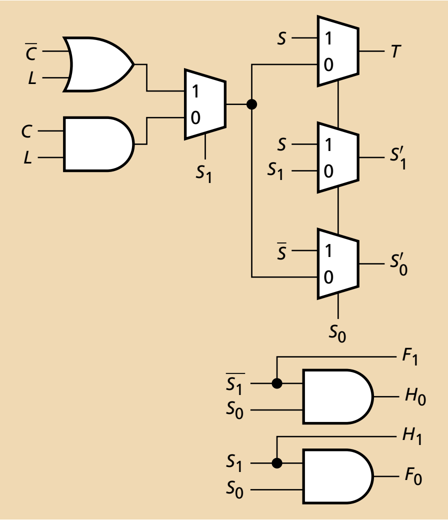
$$H_1 = S_1$$

$$H_0 = \overline{S_1} S_0$$

$$F_1 = \overline{S_1}$$

$$F_0 = S_1 S_0$$

State and Output Encoding



$$T = \overline{S_0}(\overline{S_1}\,CL + S_1(\overline{C} + L)) + S_0 S$$

$$S'_1 = S_0 S + S_1 \overline{S_0}$$

$$S'_0 = \overline{S_0}(\overline{S_1}\,C L + S_1(\overline{C} + L)) + S_0 \overline{S}$$

$$H_1 = S_1$$

$$H_0 = \overline{S_1}\,S_0$$

$$F_1 = \overline{S_1}$$

$$F_0 = S_1 S_0$$

State and Output Encoding

S	C	S	L	T	S'
HG	0	X	X	0	HG
HG	X	X	0	0	HG
HG	1	X	1	1	HY
HY	X	0	X	0	HY
HY	X	1	X	1	FG
FG	1	X	0	0	FG
FG	0	X	X	1	FY
FG	X	X	1	1	FY
FY	X	0	X	0	FY
FY	X	1	X	1	HG

S	H	F
HG	G	R
HY	Y	R
FG	R	G
FY	R	Y

Second idea: use a one-hot encoding:

HG	0001
HY	0010
FG	0100
FY	1000

G	001
Y	010
R	100

State and Output Encoding

S	C	S	L	T	S′
0001	0	X	X	0	0001
0001	X	X	0	0	0001
0001	1	X	1	1	0010
0010	X	0	X	0	0010
0010	X	1	X	1	0100
0100	1	X	0	0	0100
0100	0	X	X	1	1000
0100	X	X	1	1	1000
1000	X	0	X	0	1000
1000	X	1	X	1	0001

S	H	F
0001	001	100
0010	010	100
0100	100	001
1000	100	010

$$T = S_0 C L + S_1 S + S_2(\overline{C} + L) + S_3 S$$

$$S'_3 = S_2(\overline{C} + L) + S_3 \overline{S}$$

$$S'_2 = S_1 S + S_2 \overline{(\overline{C} + L)}$$

$$S'_1 = S_0 C L + S_1 \overline{S}$$

$$S'_0 = S_0 \overline{(C L)} + S_3 S$$

$$H_R = S_2 + S_3$$

$$H_Y = S_1$$

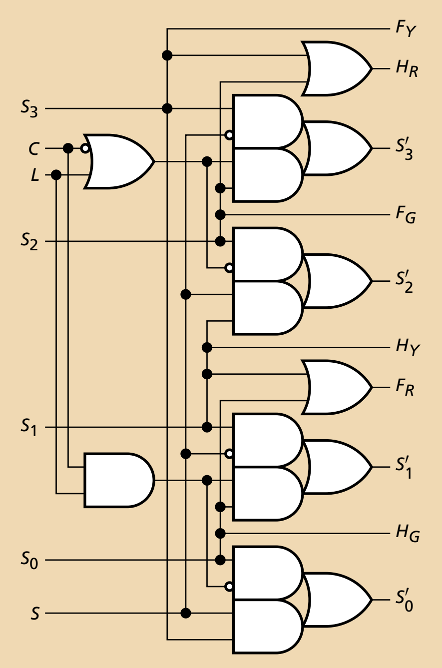
$$H_G = S_0$$

$$F_R = S_0 + S_1$$

$$F_Y = S_3$$

$$F_G = S_2$$

State and Output Encoding



$$T = S_0 C L + S_1 S + S_2(\overline{C} + L) + S_3 S$$

$$S'_3 = S_2(\overline{C} + L) + S_3 \overline{S}$$

$$S'_2 = S_1 S + S_2 \overline{(\overline{C} + L)}$$

$$S'_1 = S_0 C L + S_1 \overline{S}$$

$$S'_0 = S_0 \overline{(CL)} + S_3 S$$

$$H_R = S_2 + S_3$$

$$H_Y = S_1$$

$$H_G = S_0$$

$$F_R = S_0 + S_1$$

$$F_Y = S_3$$

$$F_G = S_2$$