# Lecture 7

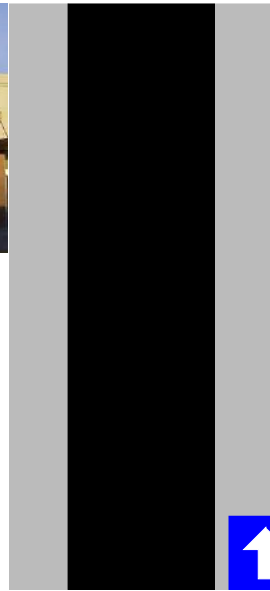
## Digital system design

Finite State Machines

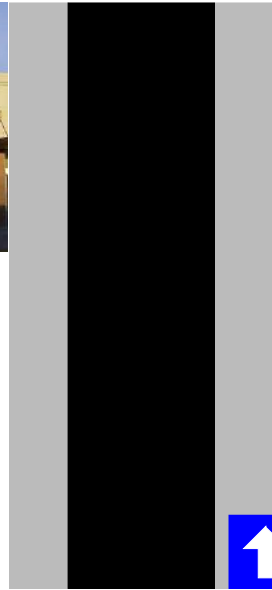
*CS173 - Conception de systèmes numériques*
*November 2016*

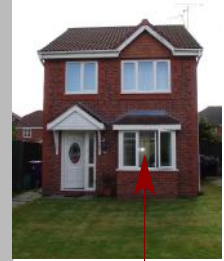
Prof. Dr. Theo Kluter
Bern University of Applied Sciences

# Introduction

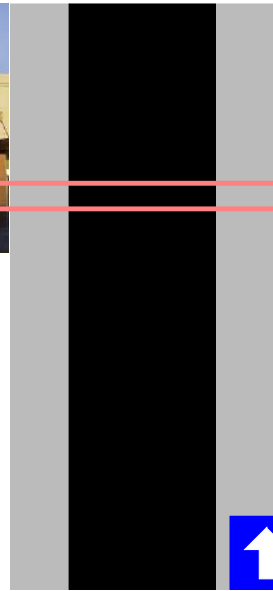



# Introduction

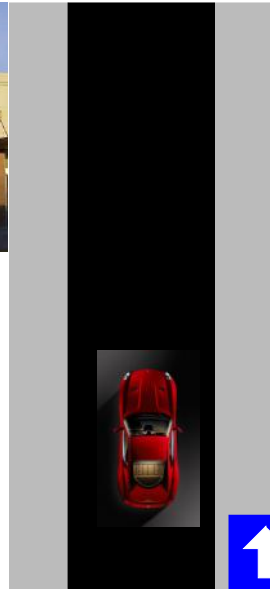



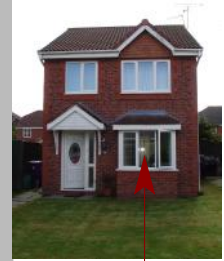


# Introduction

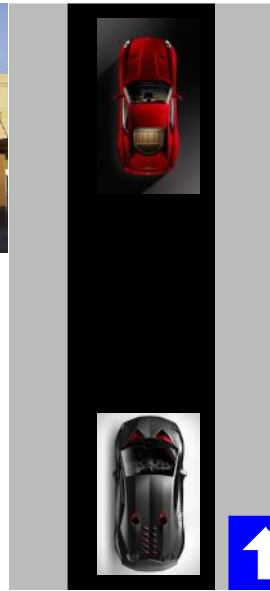



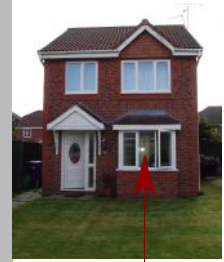
# Introduction

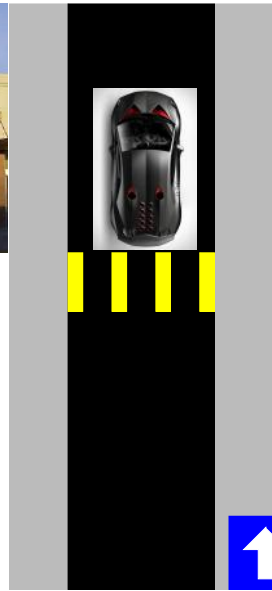


# Introduction

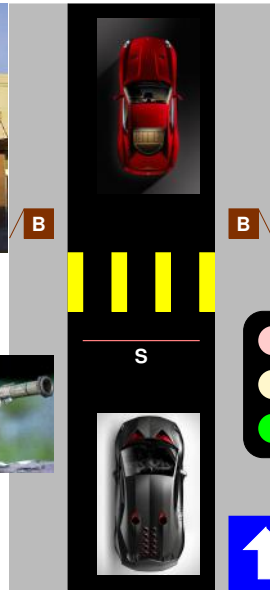


# Introduction

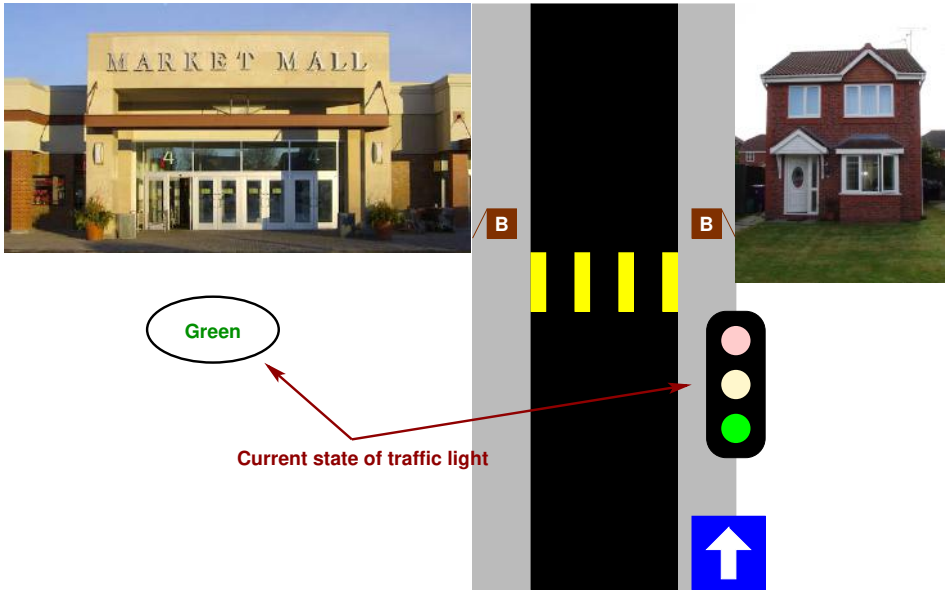



# Introduction

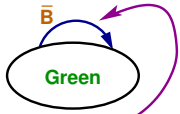

# Introduction

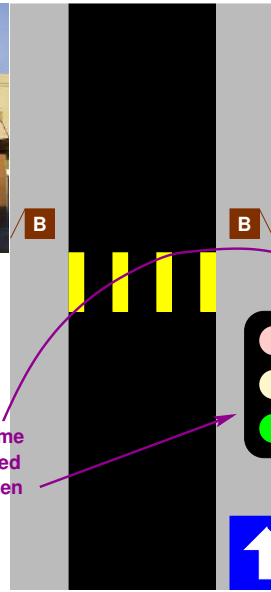




# Sequence



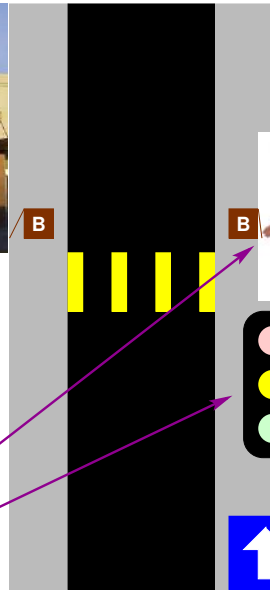
# Sequence

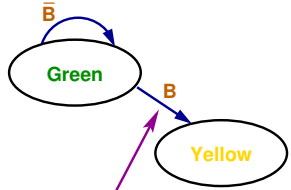




grandmother still at home
hence button not pushed
traffic light stays at green

# Sequence

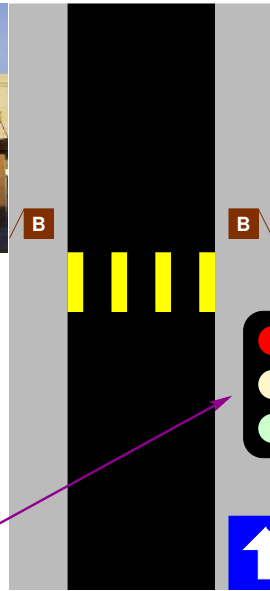



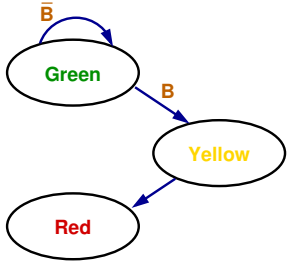



grandmother pushes button
traffic light goes to yellow

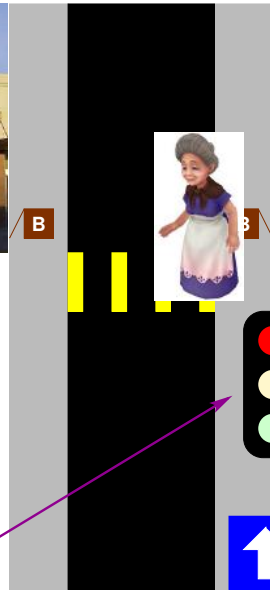
# Sequence

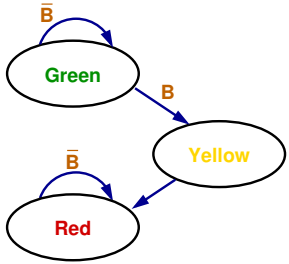




traffic light goes to red

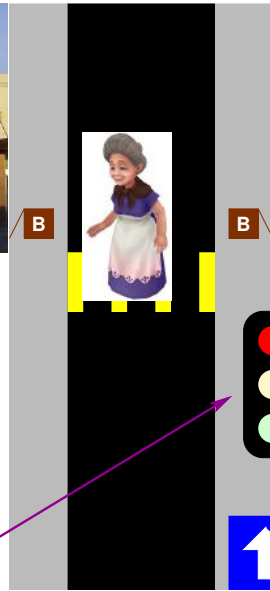
# Sequence

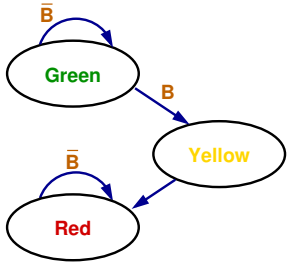




**As long as the button is not pressed the traffic light stays on red**

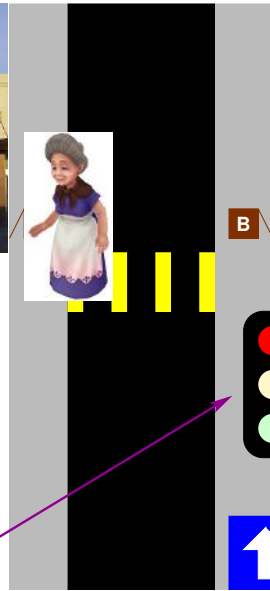
# Sequence

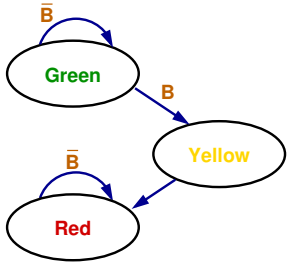




**As long as the button is not pressed the traffic light stays on red**

# Sequence


**As long as the button is not pressed the traffic light stays on red**

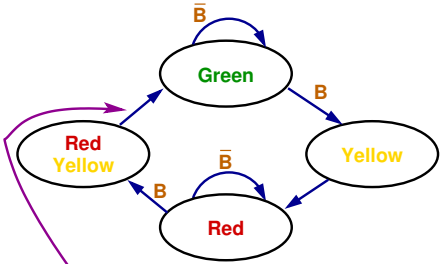
# Sequence

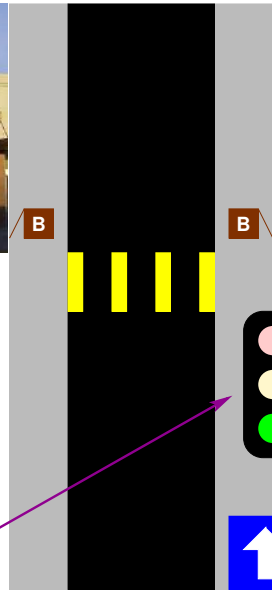

MARKET MALL




B
B
B̄
Green
B
Red
Yellow
Yellow
B̄
B
Red
When the button is again pressed
the traffic light goes to red–yellow

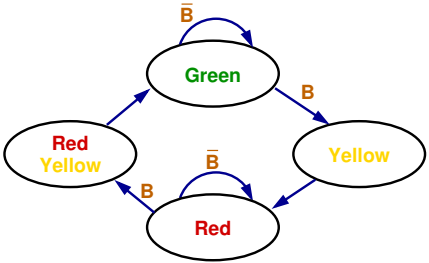
# Sequence

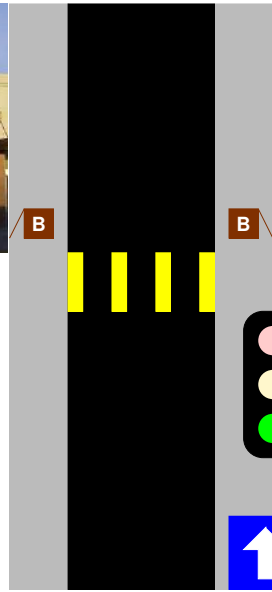




and finally back to green

# Sequence

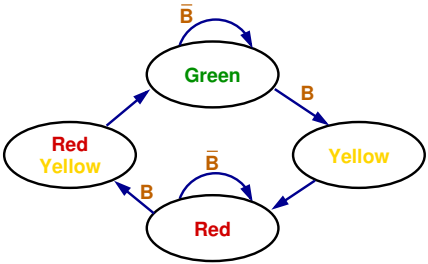




**This sequence repeats itself**

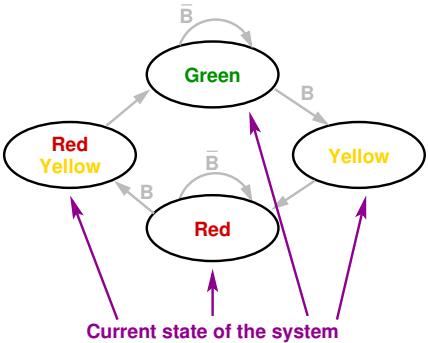
# Basic Structure

**State Diagram:**



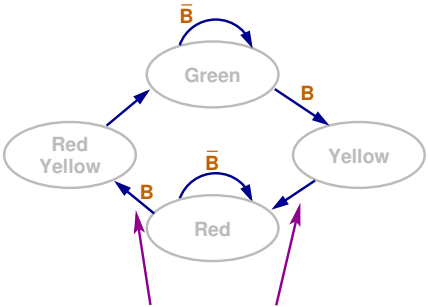
# Basic Structure

State Diagram:



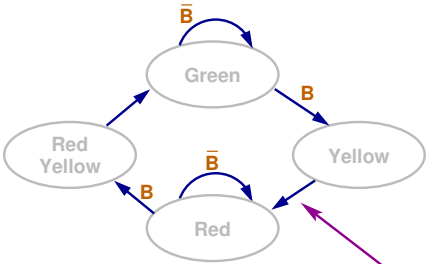
# Basic Structure

**State Diagram:**

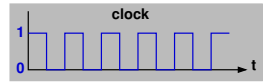


# Basic Structure

**State Diagram:**

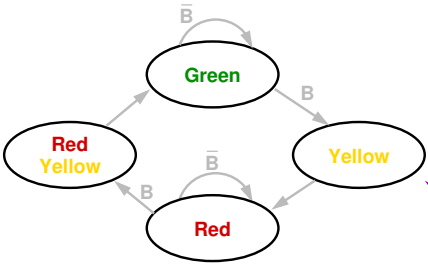



**To make a sequence (perform transitions) we require a repeating signal (clock)**

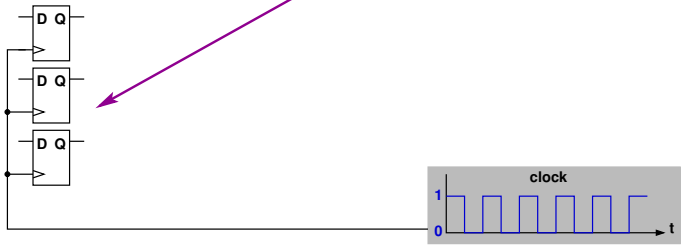


# Basic Structure

**State Diagram:**

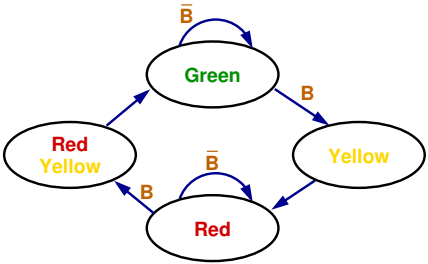



**To store the current state we require memory elements**

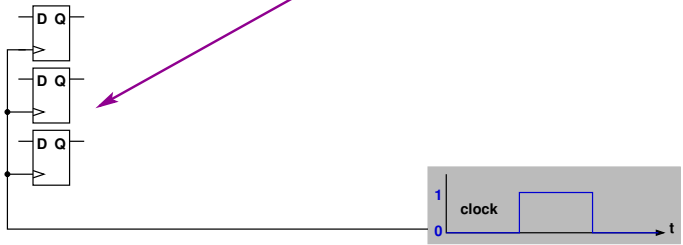


# Basic Structure

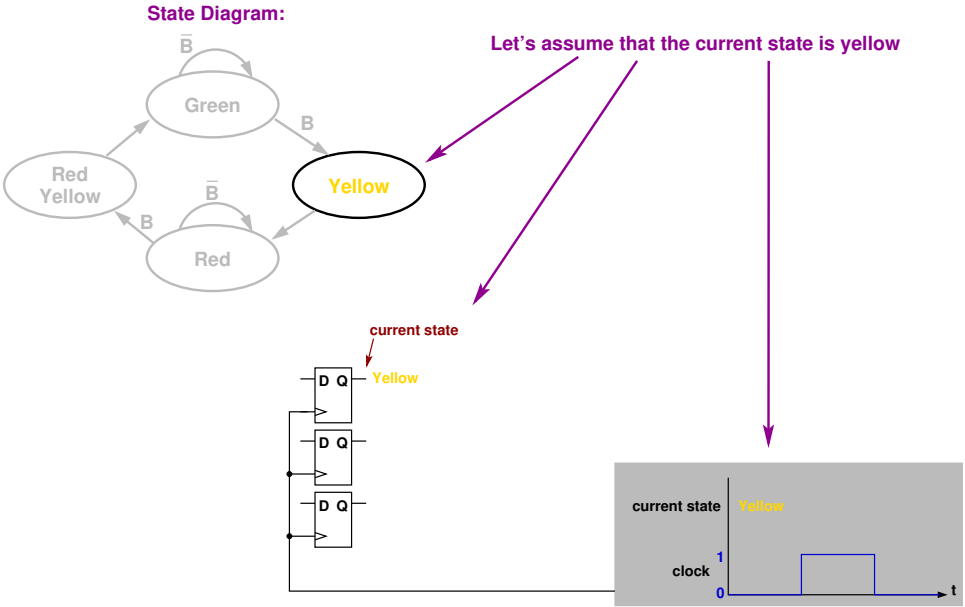
**State Diagram:**



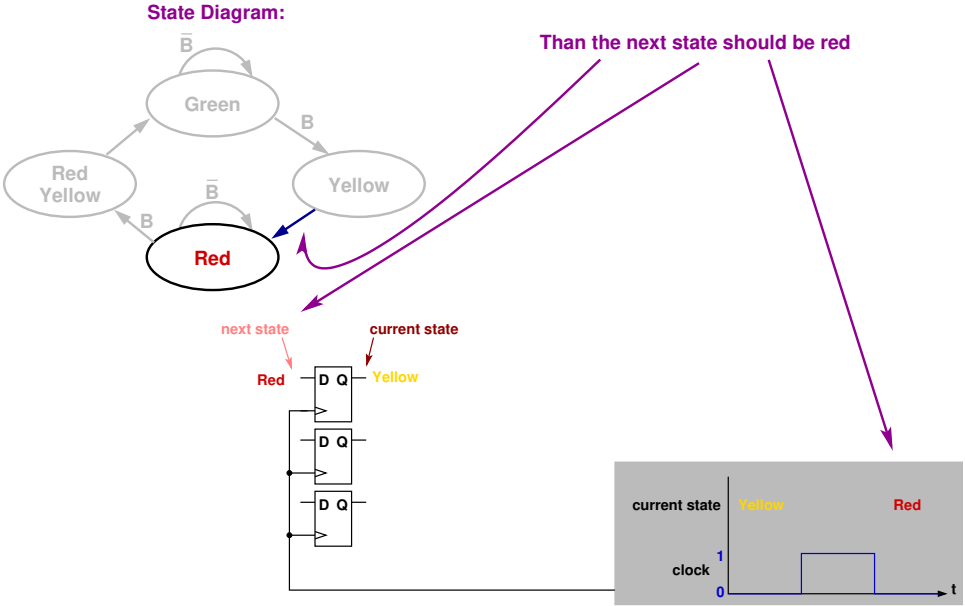

**Why D–flipflops?**



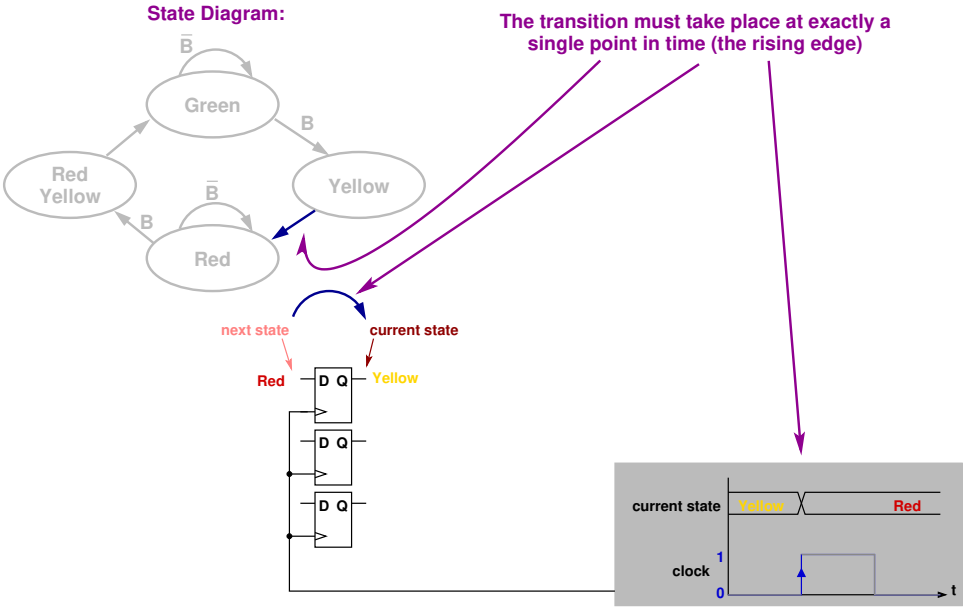
# Basic Structure



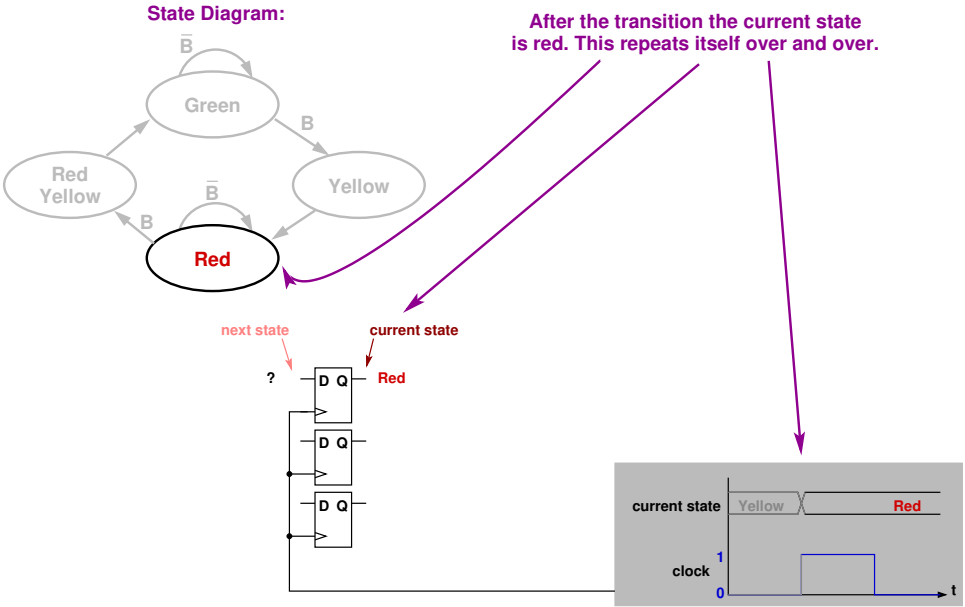
# Basic Structure



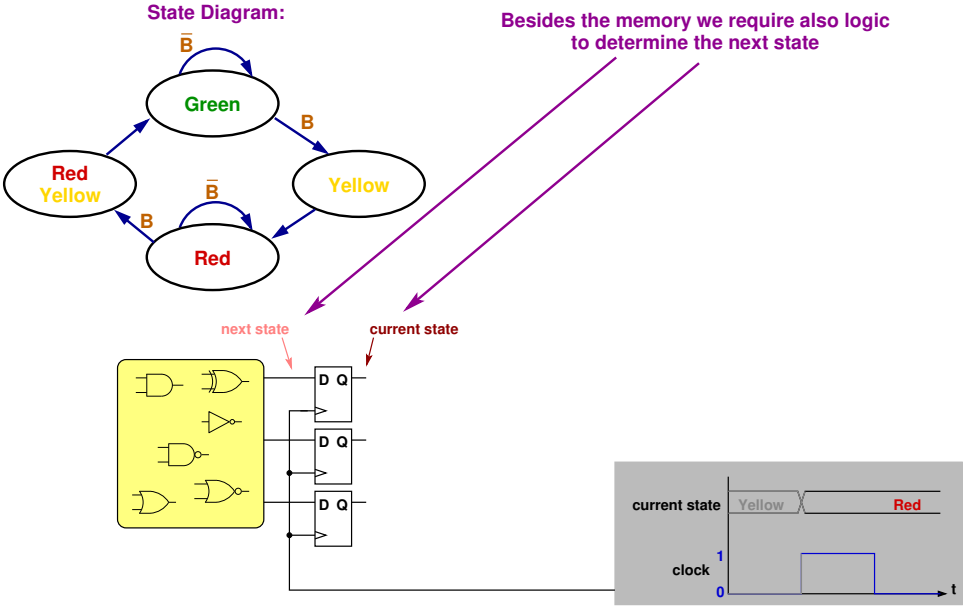
# Basic Structure



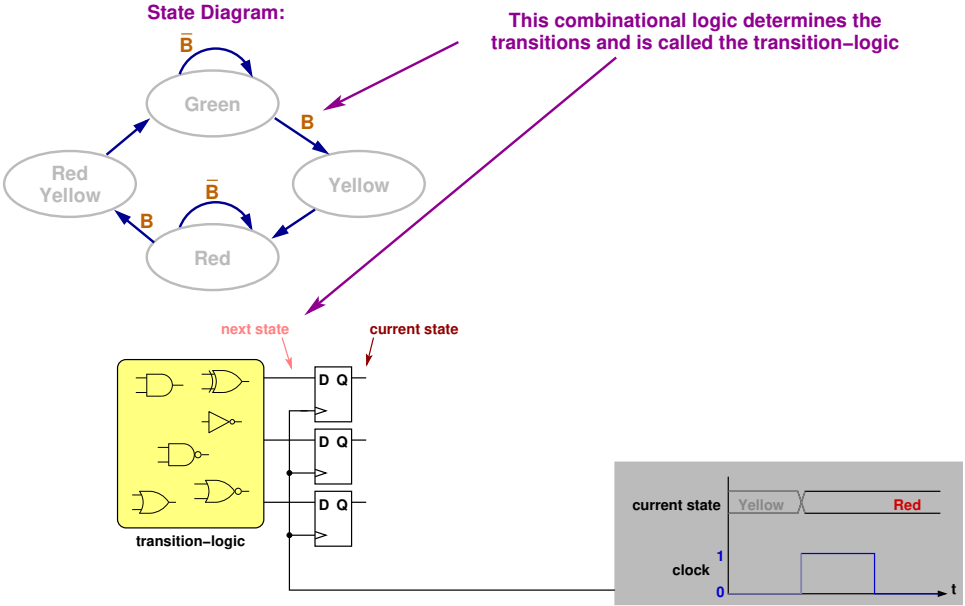
# Basic Structure



# Basic Structure

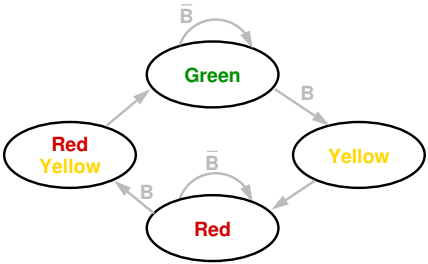


# Basic Structure

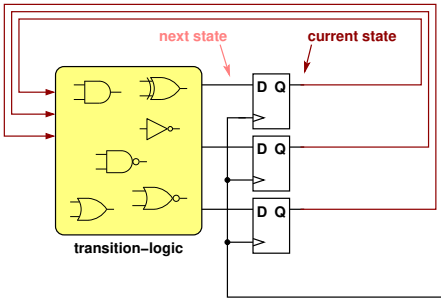


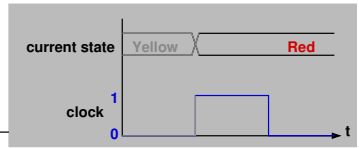
# Basic Structure

**State Diagram:**



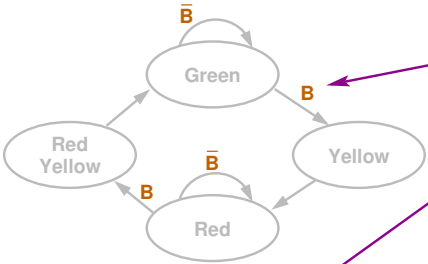

**The transition–logic needs to know what the current state is**



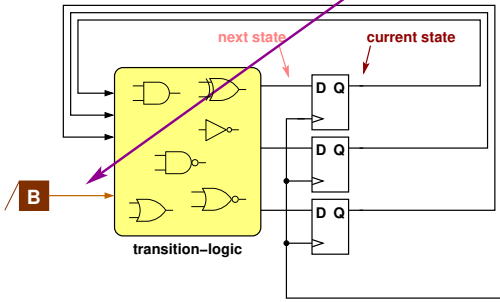



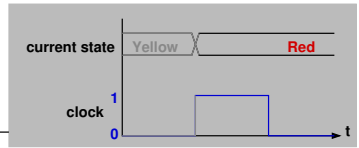
# Basic Structure

State Diagram:

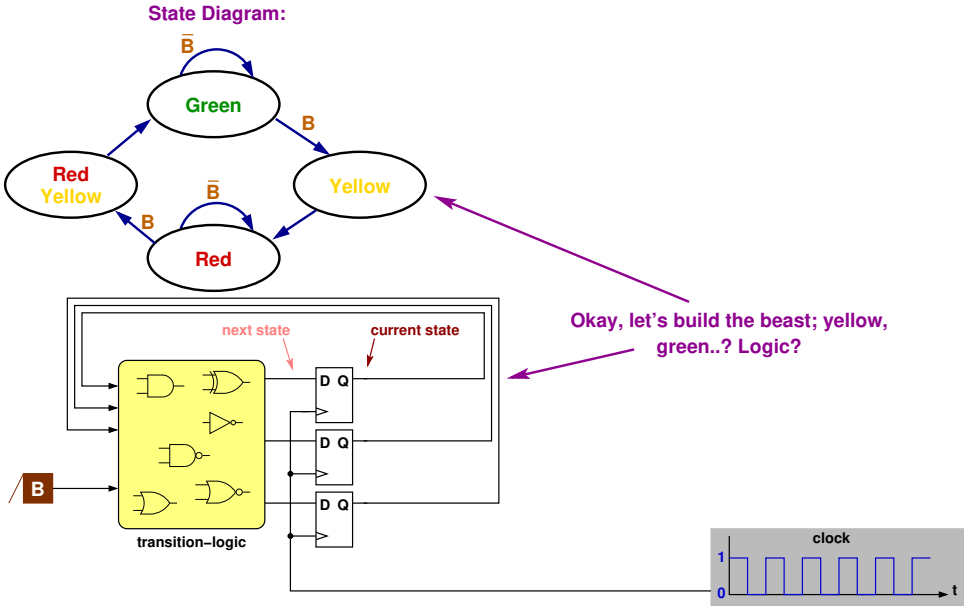



The transition–logic also needs to know the state of the system inputs

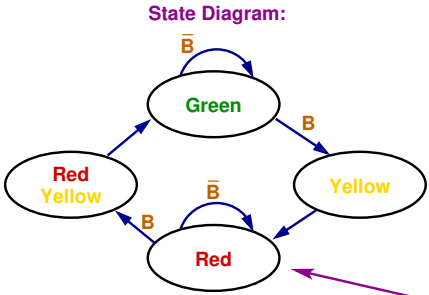


# Medvedev FSM



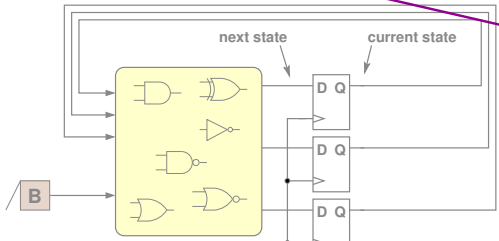
# Medvedev FSM



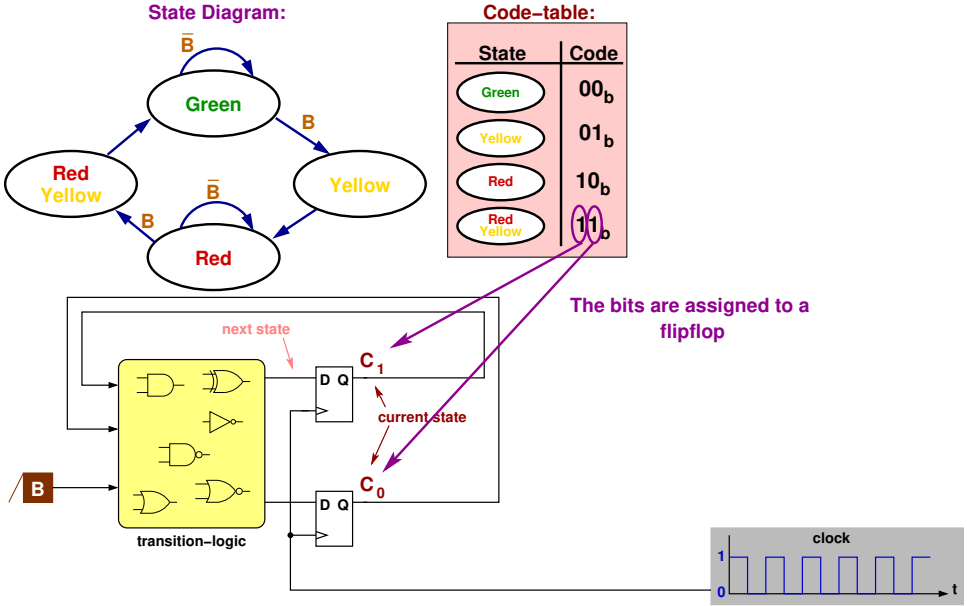

Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

We need bits!
Hence, we have to invent a code
The code table

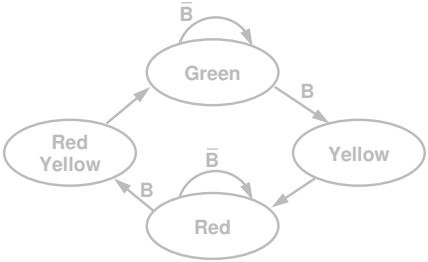



# Medvedev FSM



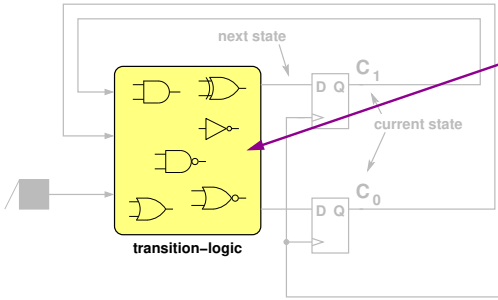
# Medvedev FSM

State Diagram:




Code–table:

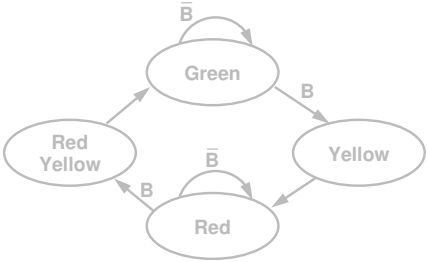
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |




Now we can determine the transition logic

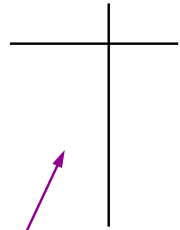
# Medvedev FSM

State Diagram:

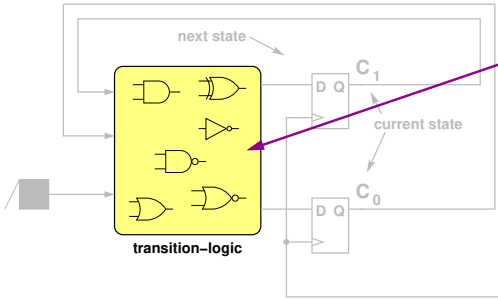



Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

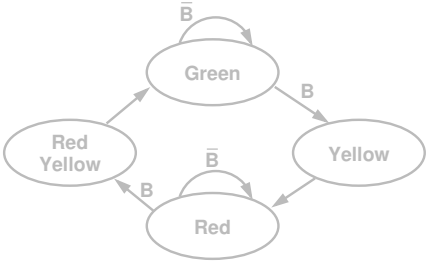


Now we can determine the transition logic
We make a truthtable!



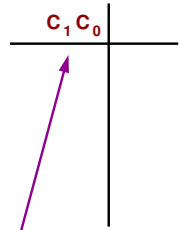

# Medvedev FSM

State Diagram:

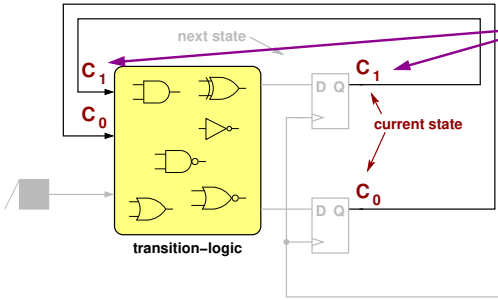



Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

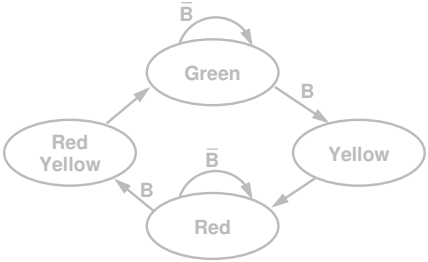



The inputs of the transition logic are the current state



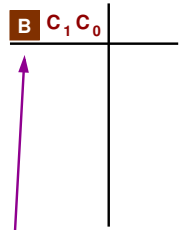

# Medvedev FSM

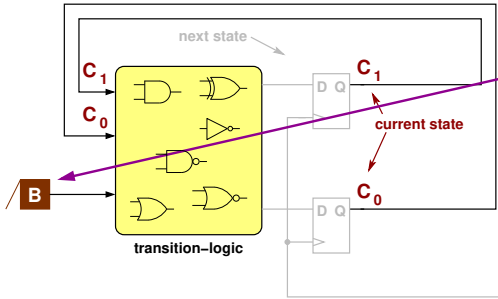
State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

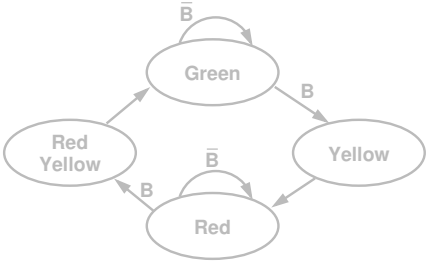



The inputs to the transistion logic are the current state and the external input(s).

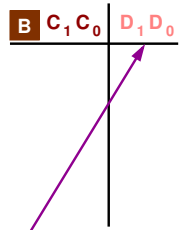
# Medvedev FSM

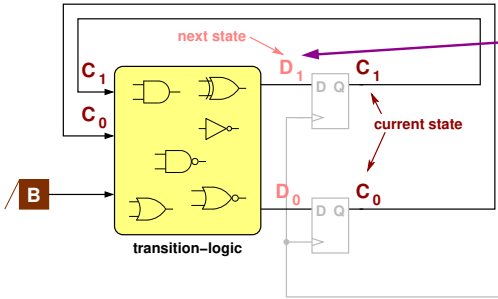
State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

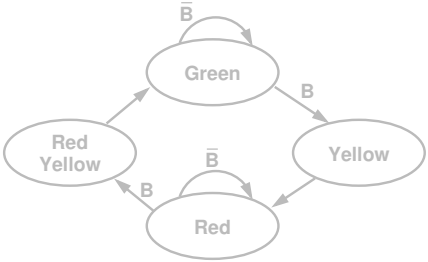



The output(s) of the transition logic is the next state.
We have to name them.

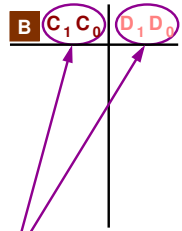
# Medvedev FSM

State Diagram:

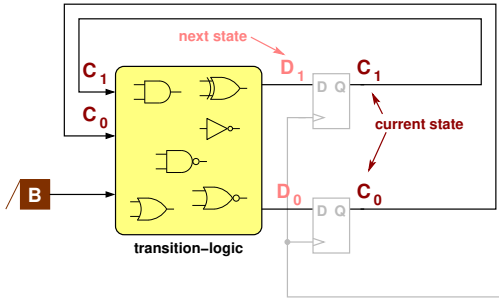



Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

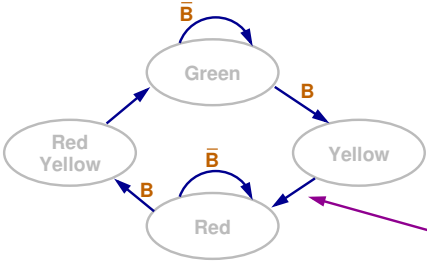



**IMPORTANT:**
**Please note the order of the state bits!**
**Always: MSB...LSB!**



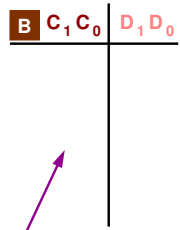

# Medvedev FSM

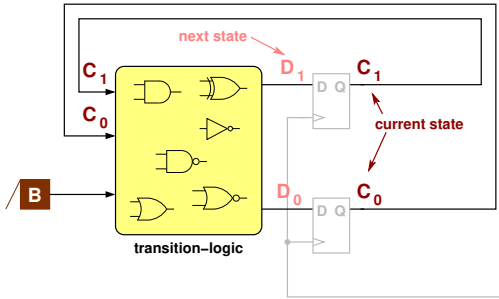
**State Diagram:**




**Code–table:**

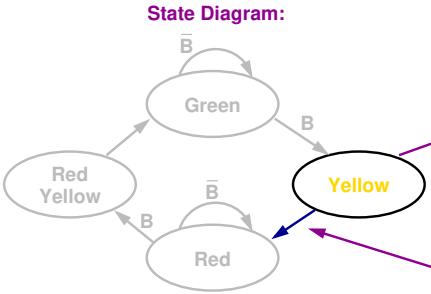
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |



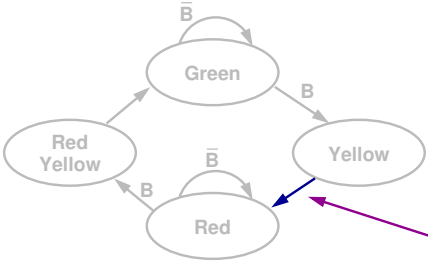

**The transition logic defines the transitions in the state diagram. Each transition represents one line.**

# Medvedev FSM



# Medvedev FSM

**State Diagram:**

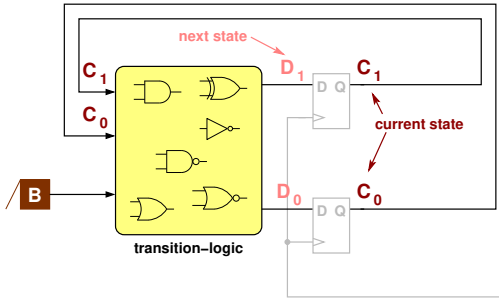



**Code–table:**

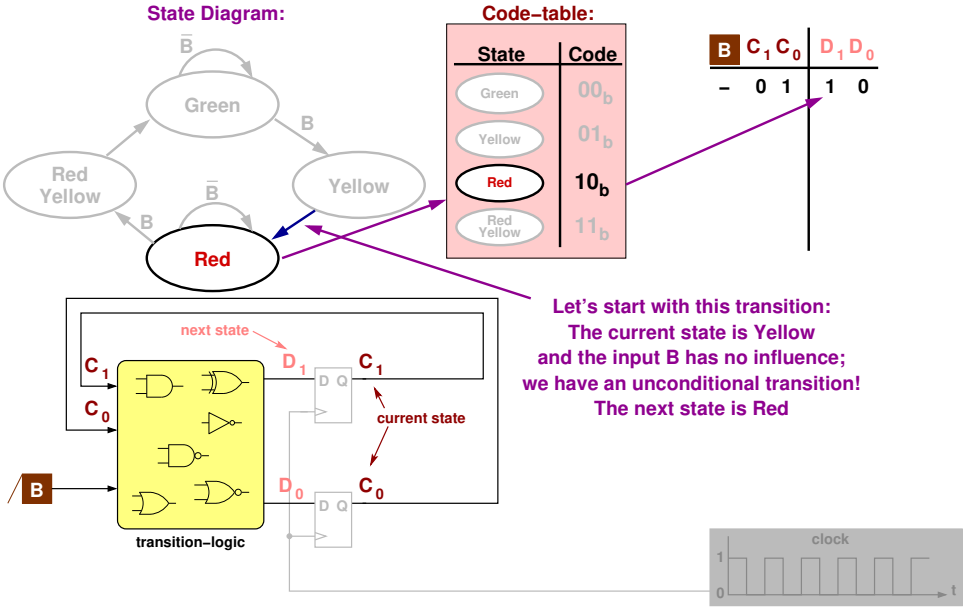
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

| B | $C_1 C_0$ | $D_1 D_0$ |
|---|---|---|
| – | 0 1 | |

**Let's start with this transition:**
**The current state is Yellow**
**and the input B has no influence;**
**we have an unconditional transition!**

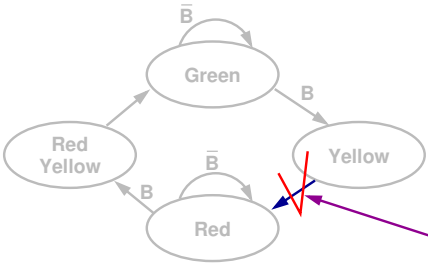



# Medvedev FSM



# Medvedev FSM

**State Diagram:**

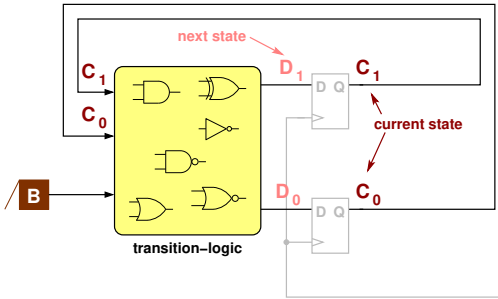



Code–table:

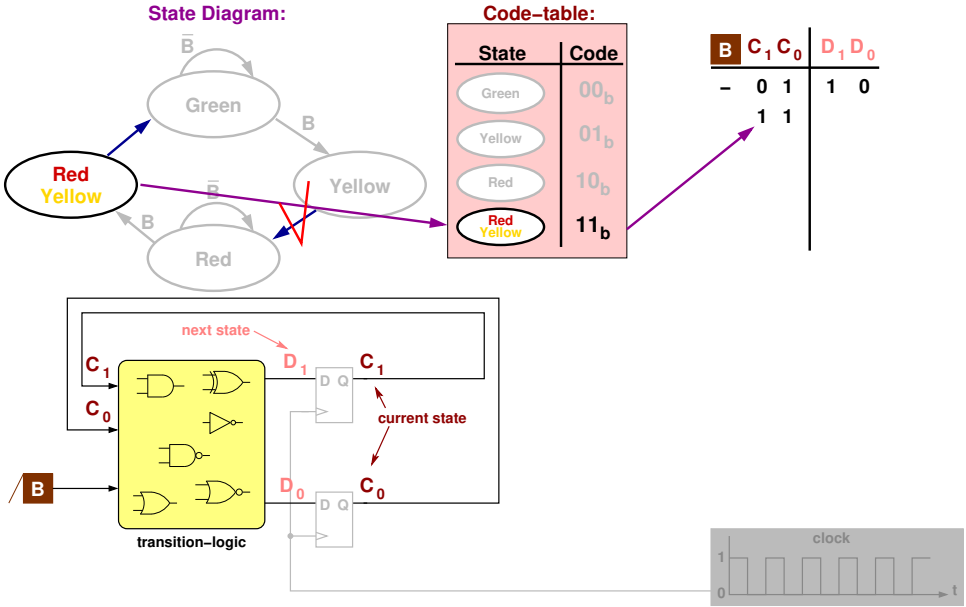
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

| B | $C_1 C_0$ | $D_1 D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |

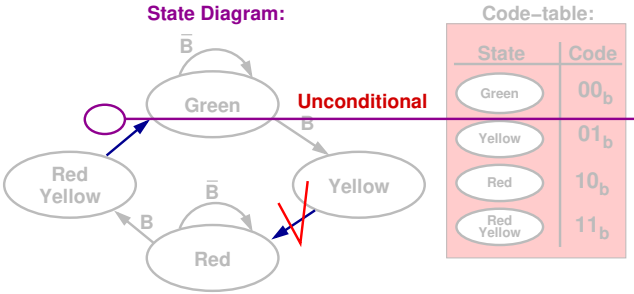
We are done for this transition;
we have to continue for all the others

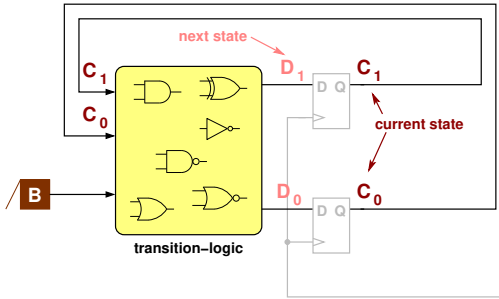


# Medvedev FSM

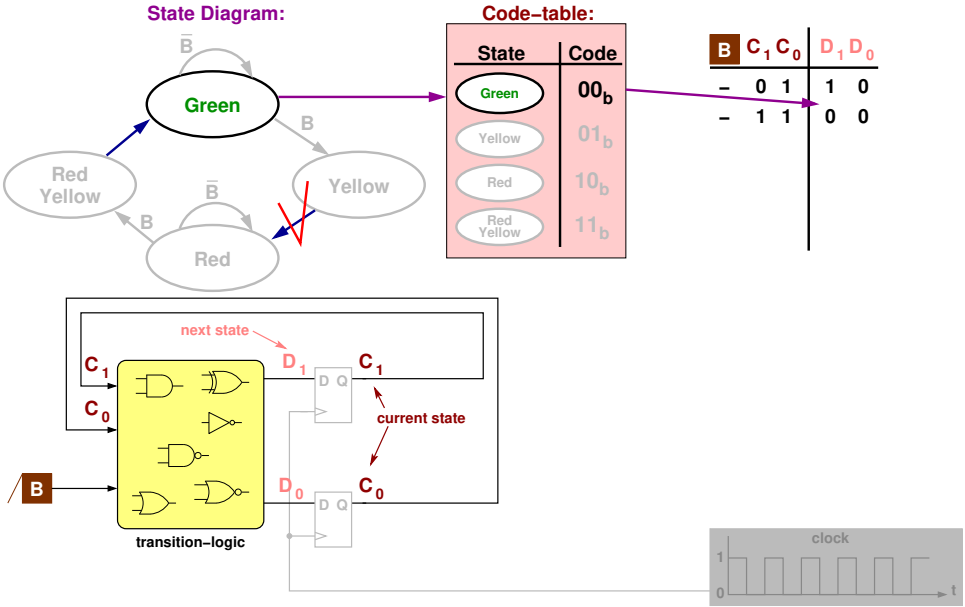


# Medvedev FSM

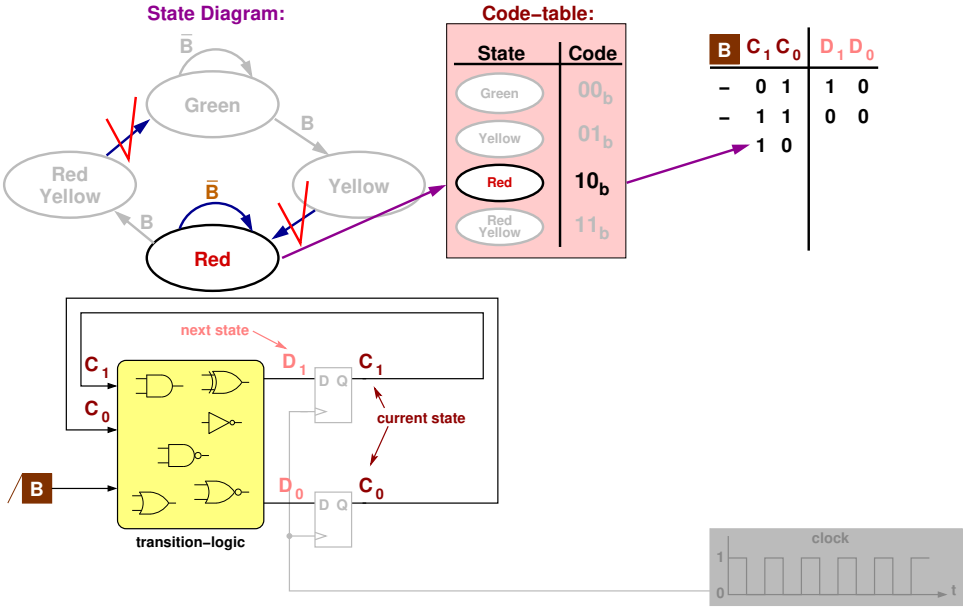


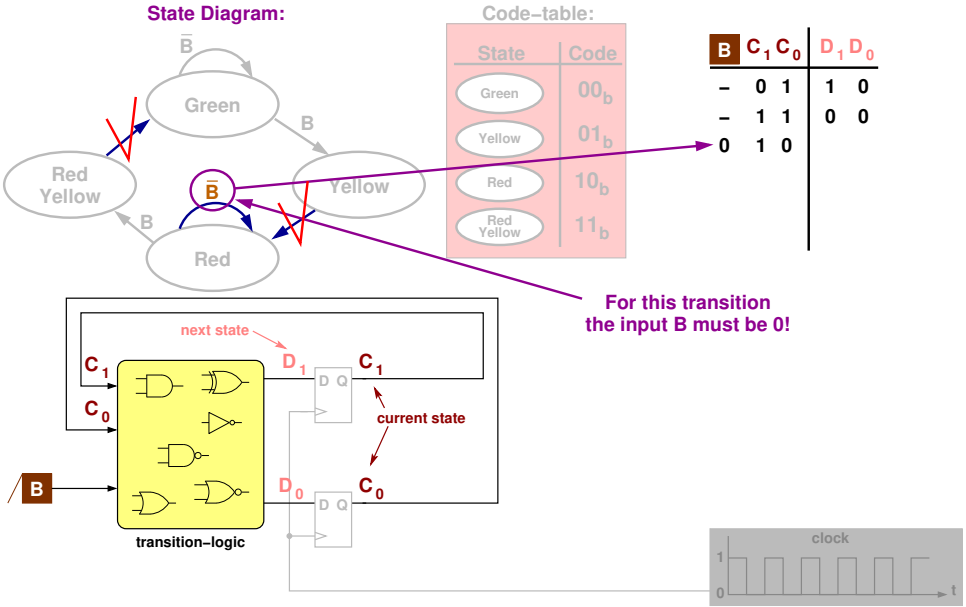

# Medvedev FSM



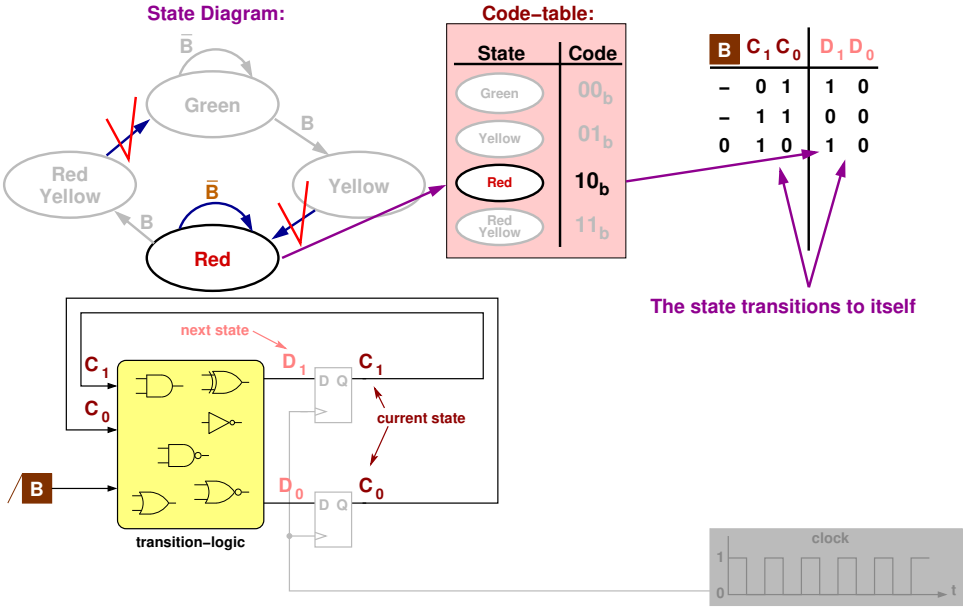
# Medvedev FSM



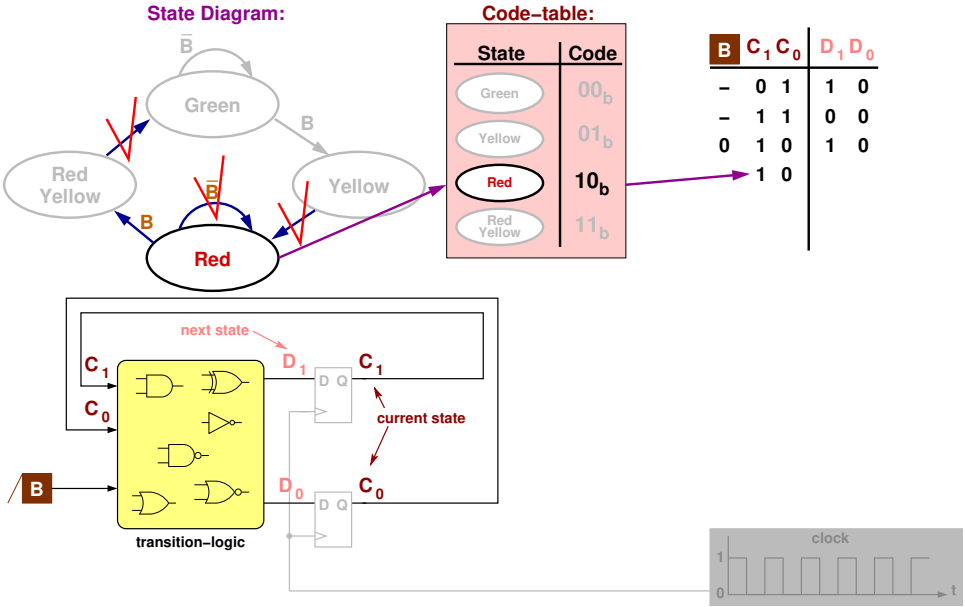
# Medvedev FSM



# Medvedev FSM

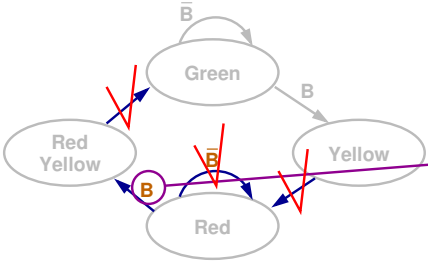


# Medvedev FSM



# Medvedev FSM

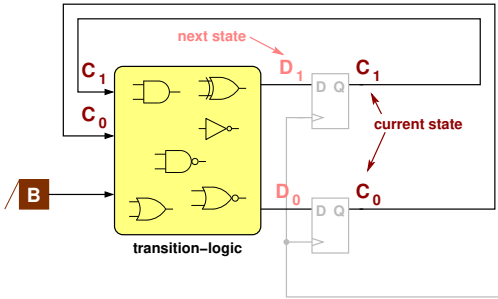
**State Diagram:**



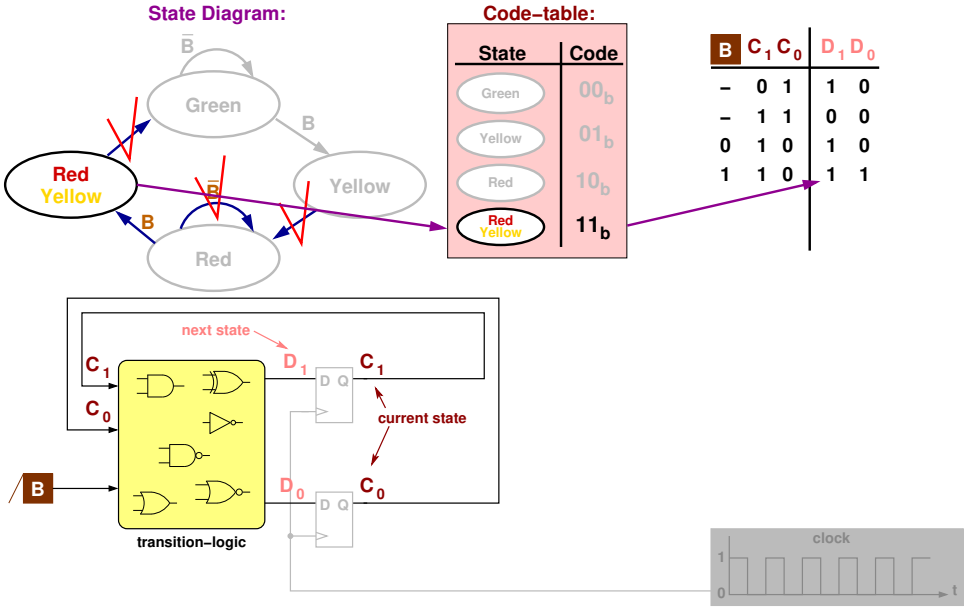

Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

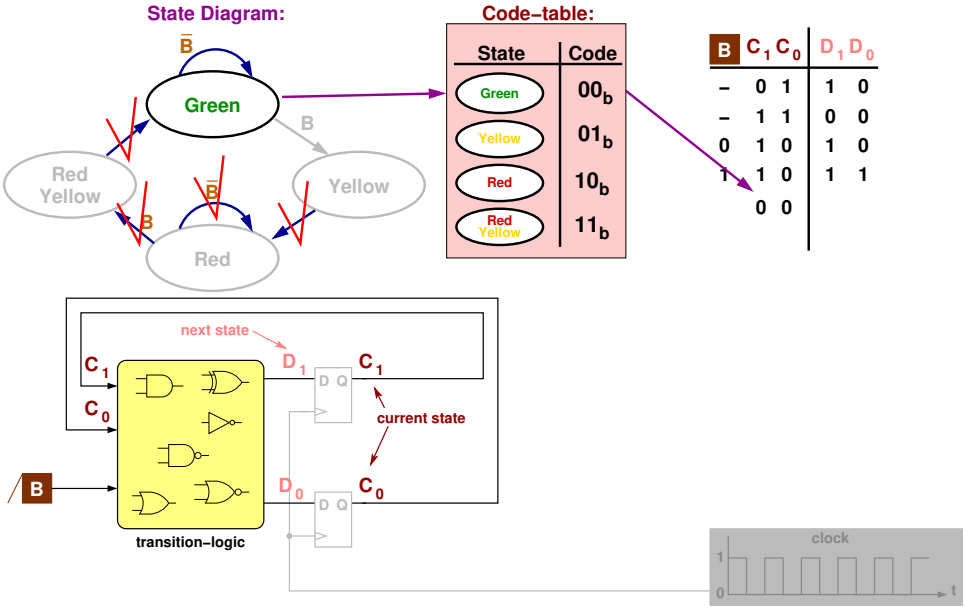
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | | |



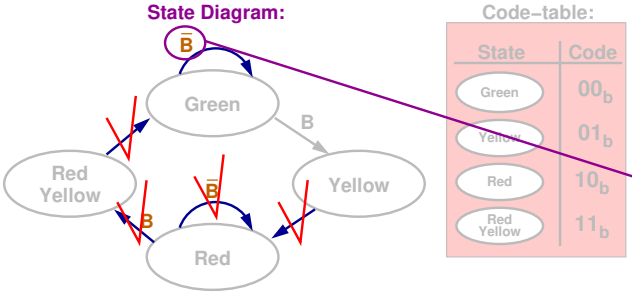
# Medvedev FSM



# Medvedev FSM



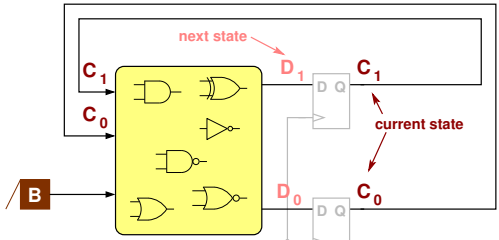
# Medvedev FSM



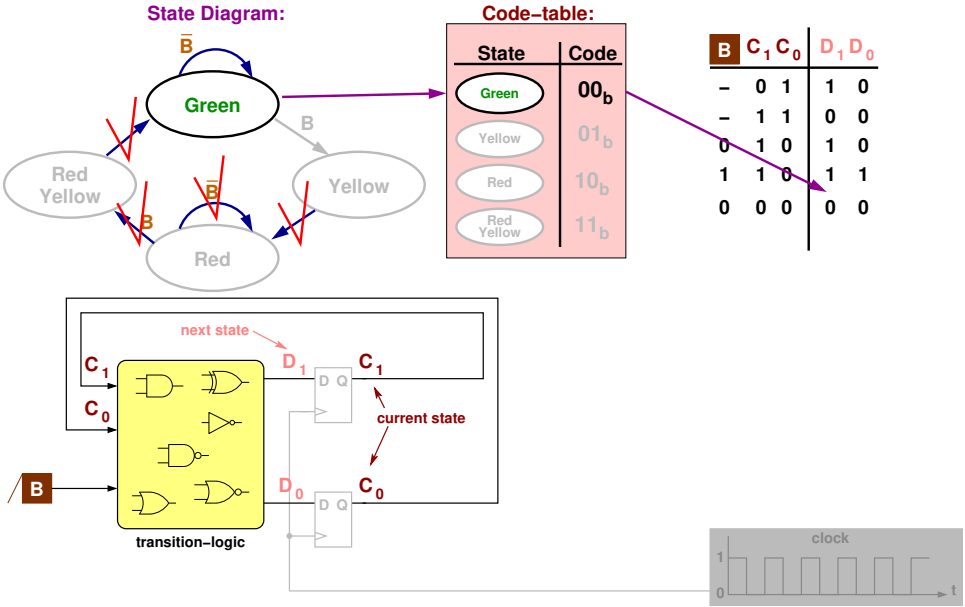

Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

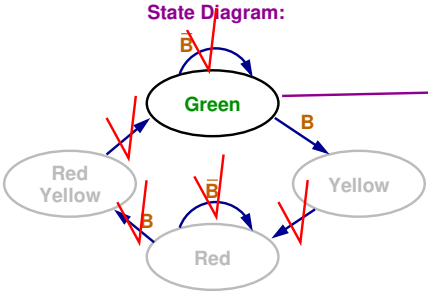
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | | |




# Medvedev FSM



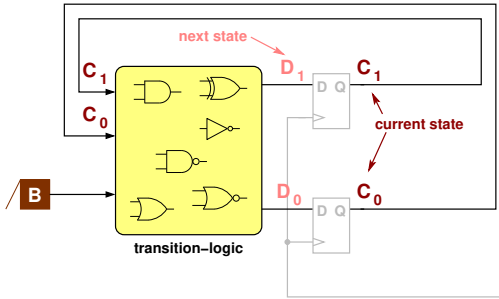
# Medvedev FSM



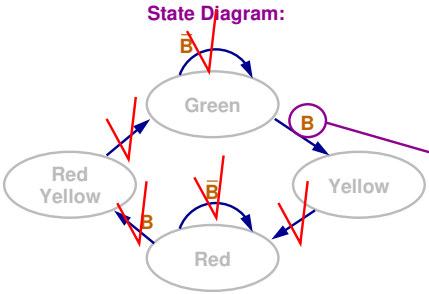

Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| | 0 | 0 | | |



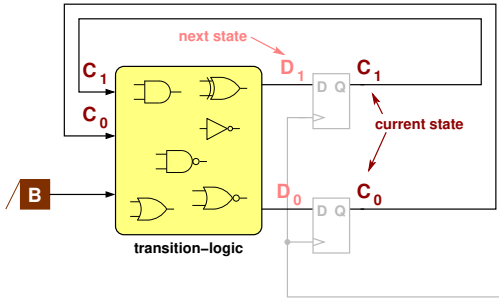
# Medvedev FSM



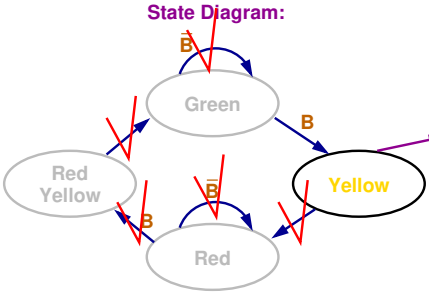

Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | | |



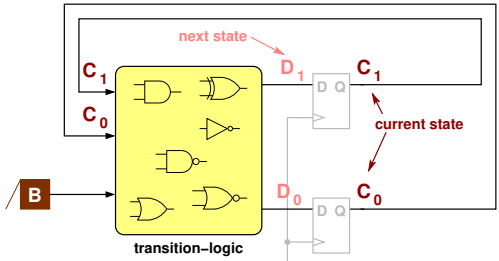

# Medvedev FSM




Code–table:

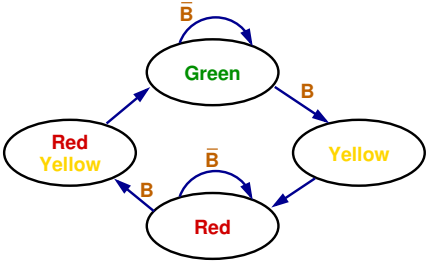
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |




# Medvedev FSM

**State Diagram:**



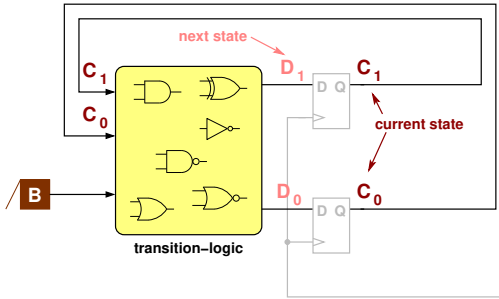

**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

**Transition–table:**

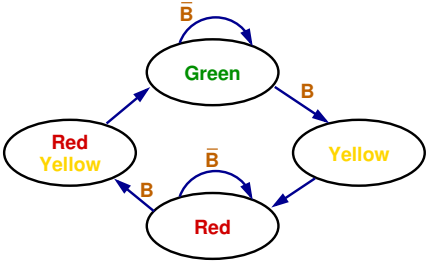
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

We are done!
This created truthtable is called the transition table of an FSM.



# Medvedev FSM

**State Diagram:**



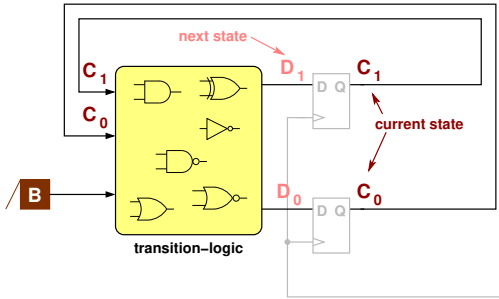

**Code–table:**

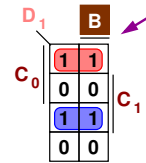
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

**Transition–table:**

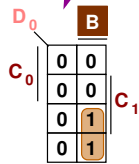
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

**We can build the circuit**



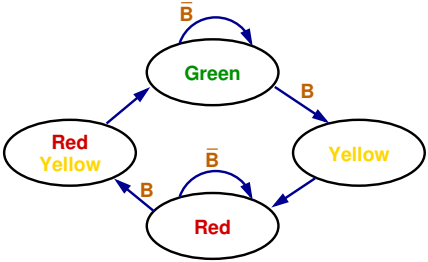

$D_1 = C_0 \oplus C_1$




$D_0 = \overline{C_0} \cdot B$

# Medvedev FSM

**State Diagram:**

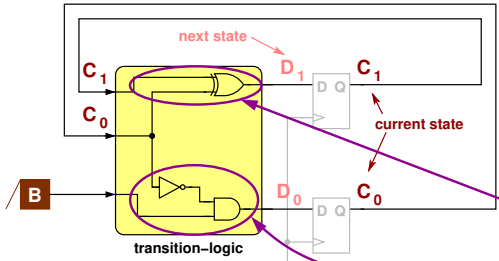


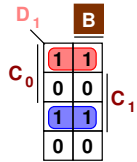

**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

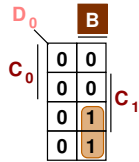
**Transition–table:**

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |



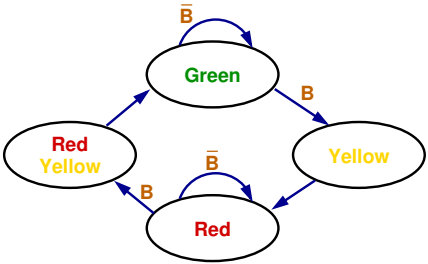

$D_1 = C_0 \oplus C_1$




$D_0 = \overline{C_0} \cdot B$

# Medvedev FSM

**State Diagram:**

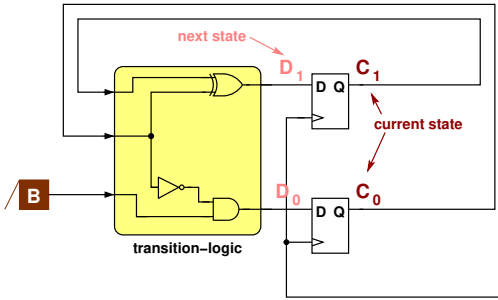



**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

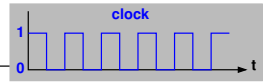
**Transition–table:**

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |



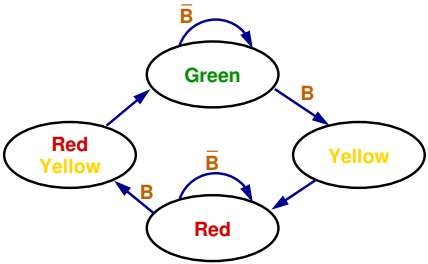

**This type of Finite State Machine (FSM) is called a:**

**Medvedev FSM**



# Moore FSM

**State Diagram:**

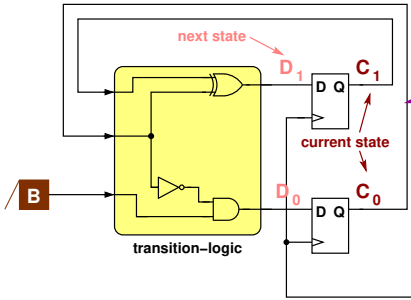


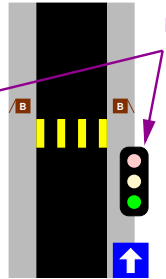

**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

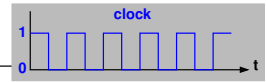
**Transition–table:**

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |



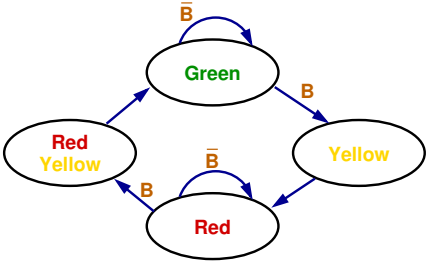




Now we have the sequence; but how to control the outputs of the system?



# Moore FSM

**State Diagram:**

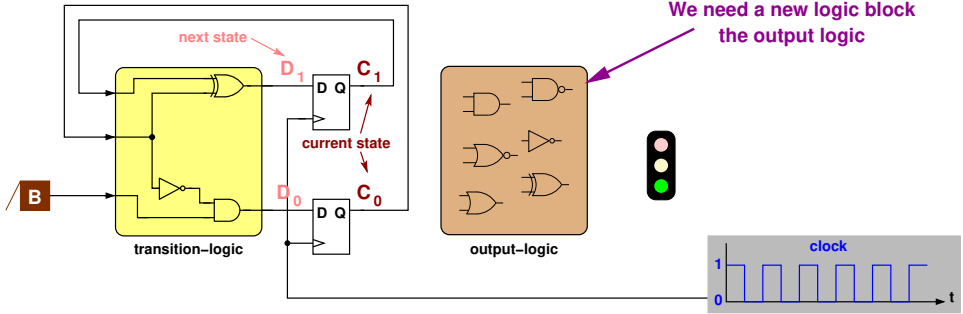



**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

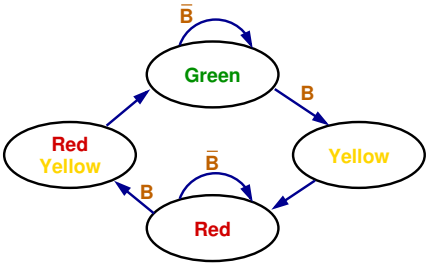
**Transition–table:**

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |



# Moore FSM

**State Diagram:**

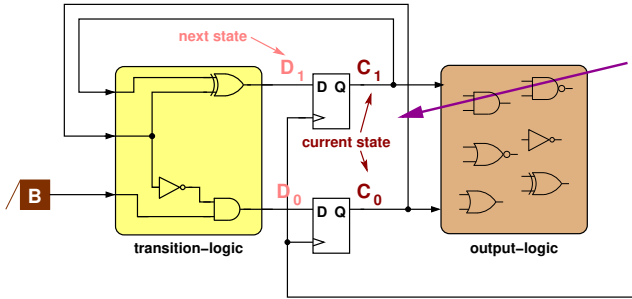



**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

**Transition–table:**

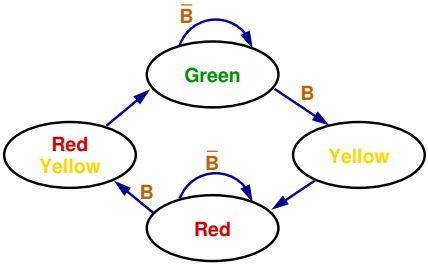
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |




We need a new logic block
the output logic
it depends on the current state

# Moore FSM

**State Diagram:**

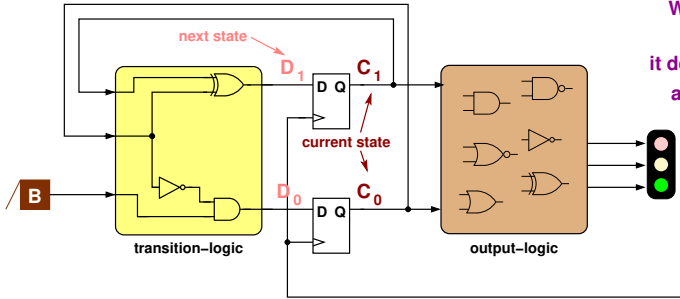



**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

**Transition–table:**

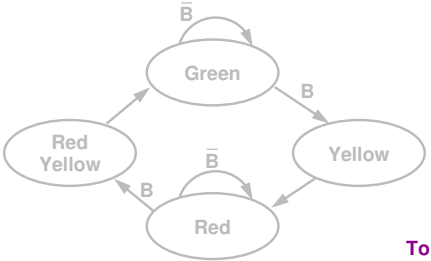
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |




We need a new logic block
the output logic
it depends on the current state
and controls the output(s)

# Moore FSM

State Diagram:



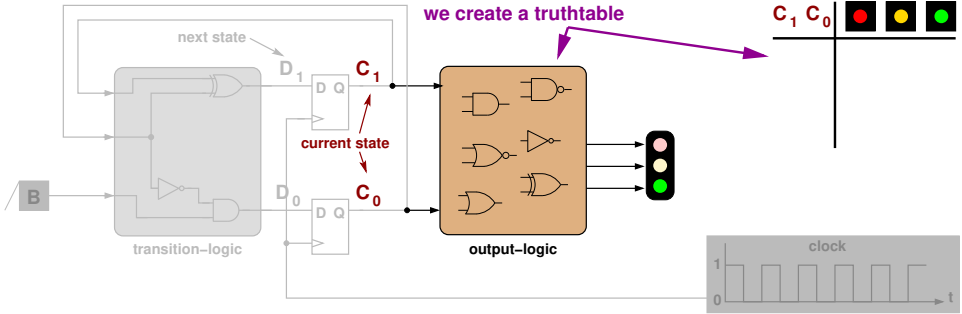

Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

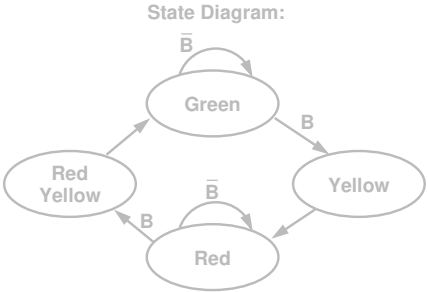
Transition–table:

| B | $C_1$ $C_0$ | $D_1$ $D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |
| – | 1 1 | 0 0 |
| 0 | 1 0 | 1 0 |
| 1 | 1 0 | 1 1 |
| 0 | 0 0 | 0 0 |
| 1 | 0 0 | 0 1 |

To realize the output logic we create a truthtable



# Moore FSM




**Code–table:**

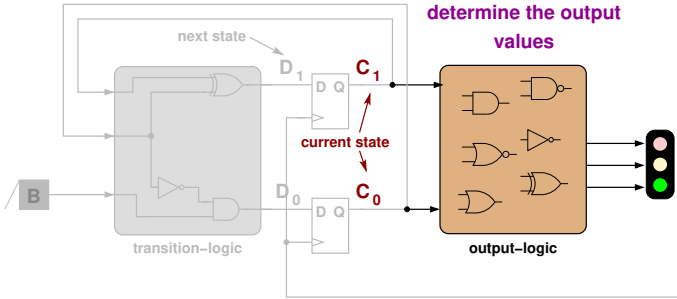
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

**Transition–table:**

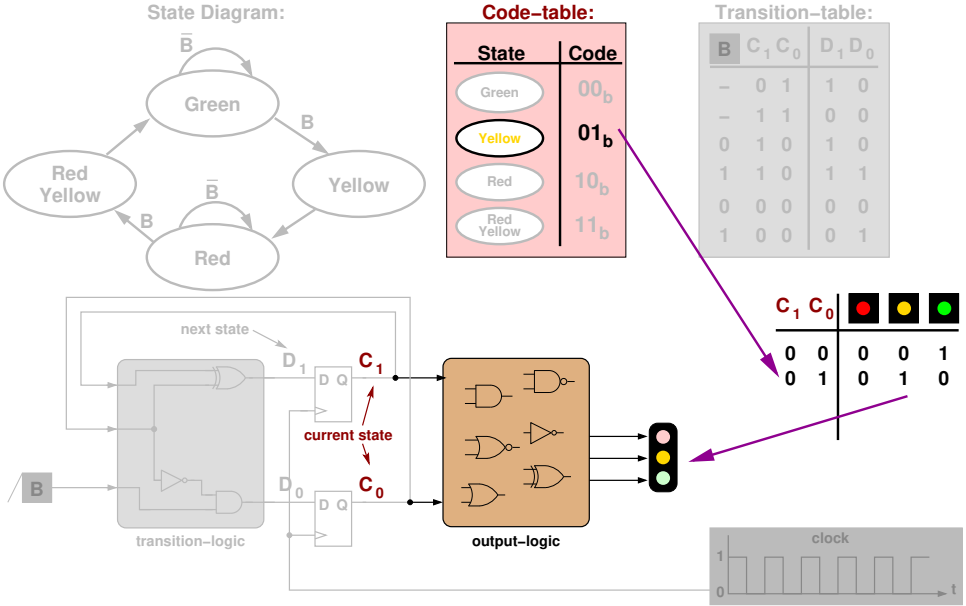
| B | $C_1$ $C_0$ | $D_1$ $D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |
| – | 1 1 | 0 0 |
| 0 | 1 0 | 1 0 |
| 1 | 1 0 | 1 1 |
| 0 | 0 0 | 0 0 |
| 1 | 0 0 | 0 1 |

**For each state we determine the output values**

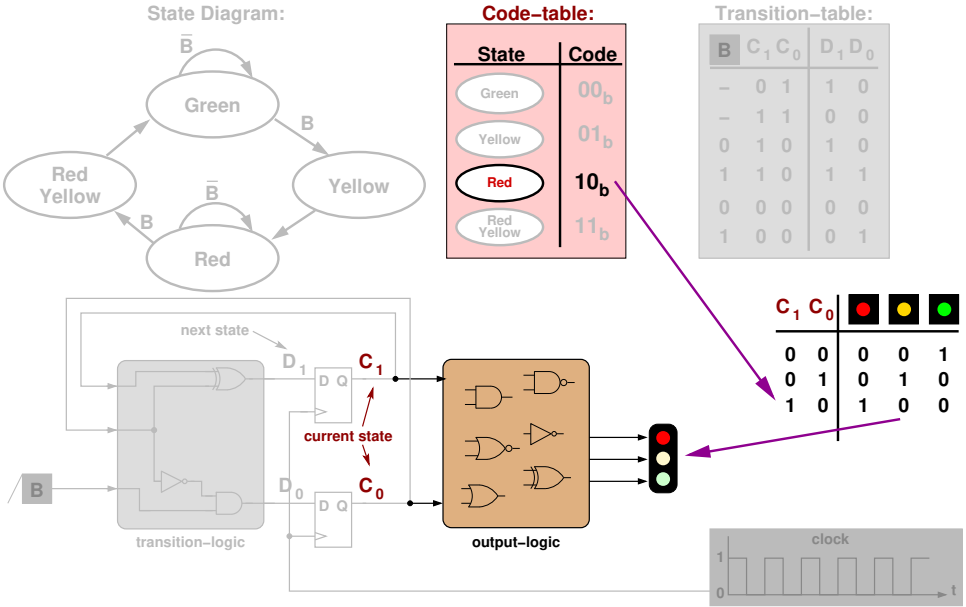
| $C_1$ $C_0$ | | | | |
|---|---|---|---|---|
| 0 0 | 0 | 0 | 0 | 1 |



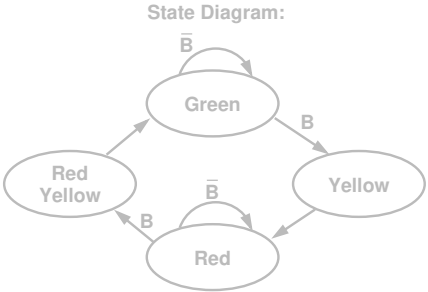
# Moore FSM


State Diagram:
Green
Yellow
Red
Red Yellow
B
Code–table:
State
Code
Green
00b
Yellow
01b
Red
10b
Red Yellow
11b
Transition–table:
B C1 C0 D1 D0
– 0 1 1 0
– 1 1 0 0
0 1 0 1 0
1 1 0 1 1
0 0 0 0 0
1 0 0 0 1
C1 C0
0 0 0 0 1
0 1 0 1 0
next state
D1
C1
current state
D0
C0
D Q
B
transition–logic
output–logic
clock
1
0
t

# Moore FSM


State Diagram:
Green
Yellow
Red
Red Yellow
Code–table:
State
Code
Green
00b
Yellow
01b
Red
10b
Red Yellow
11b
Transition–table:
B C1 C0 D1 D0
– 0 1 1 0
– 1 1 0 0
0 1 0 1 0
1 1 0 1 1
0 0 0 0 0
1 0 0 0 1
C1 C0
0 0 0 0 1
0 1 0 1 0
1 0 1 0 0
next state
D1
C1
current state
D0
C0
D Q
B
transition–logic
output–logic
clock
1
0
t

# Moore FSM

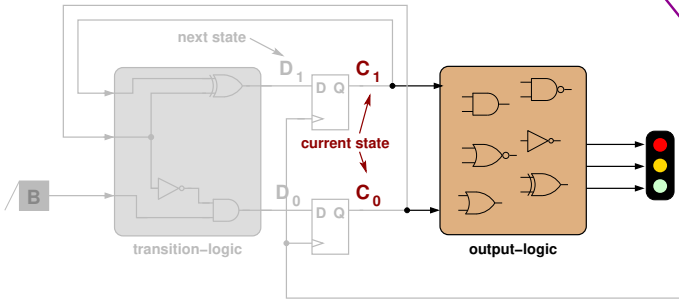



**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

**Transition–table:**

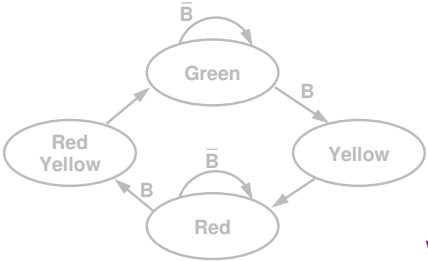
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |




| $C_1$ $C_0$ | | | | |
|---|---|---|---|---|
| 0 0 | 0 | 0 | 1 | |
| 0 1 | 0 | 1 | 0 | |
| 1 0 | 1 | 0 | 0 | |
| 1 1 | 1 | 1 | 0 | |

# Moore FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

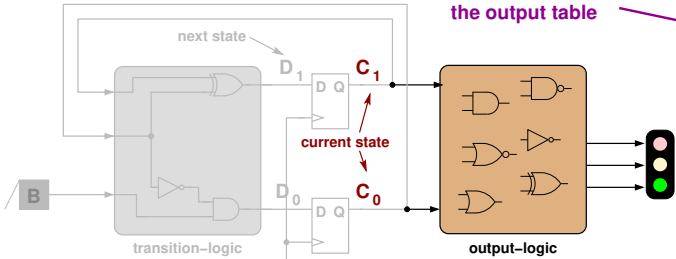
Transition–table:

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

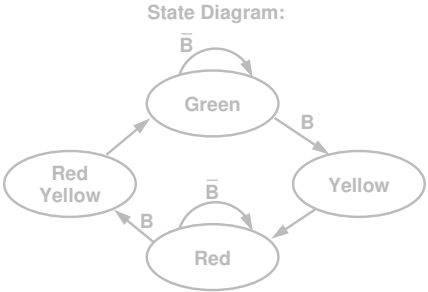
We call this truthtable the output table

Output–table:

| $C_1$ | $C_0$ | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |



# Moore FSM

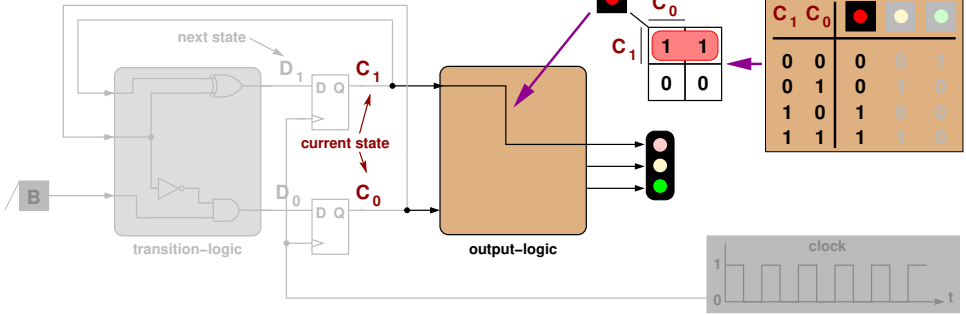



Code–table:

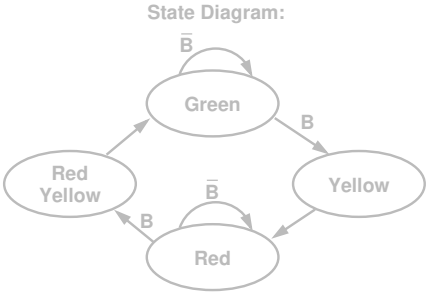
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

Transition–table:

| B | $C_1$ $C_0$ | $D_1$ $D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |
| – | 1 1 | 0 0 |
| 0 | 1 0 | 1 0 |
| 1 | 1 0 | 1 1 |
| 0 | 0 0 | 0 0 |
| 1 | 0 0 | 0 1 |



# Moore FSM

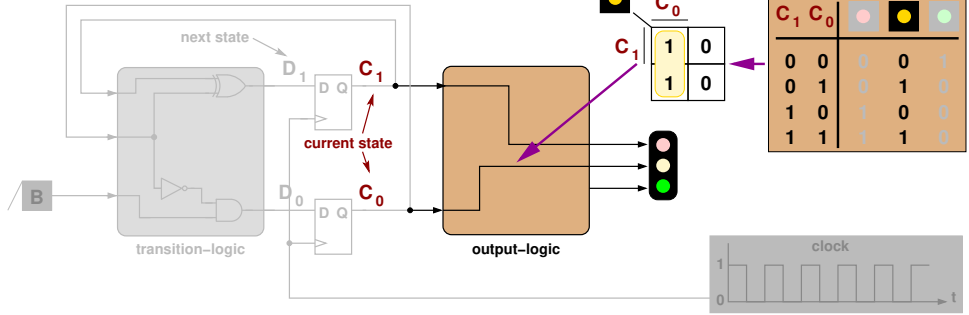



Code–table:

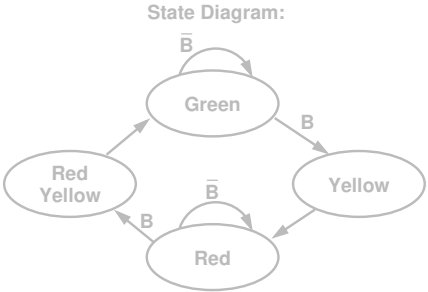
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

Transition–table:

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |



# Moore FSM

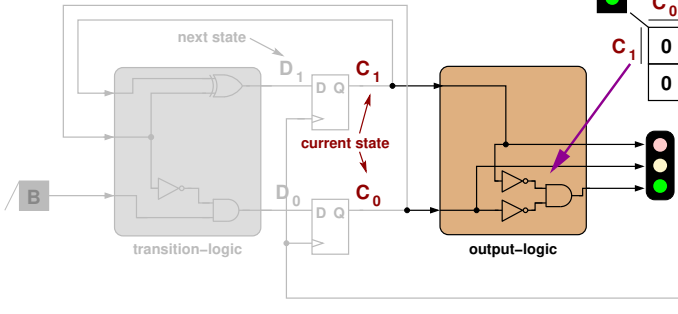



Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

Transition–table:

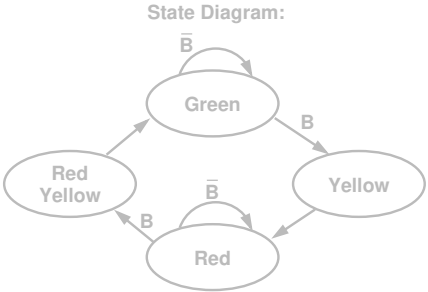
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |




Output–table:

| $C_1$ | $C_0$ | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

# Moore FSM

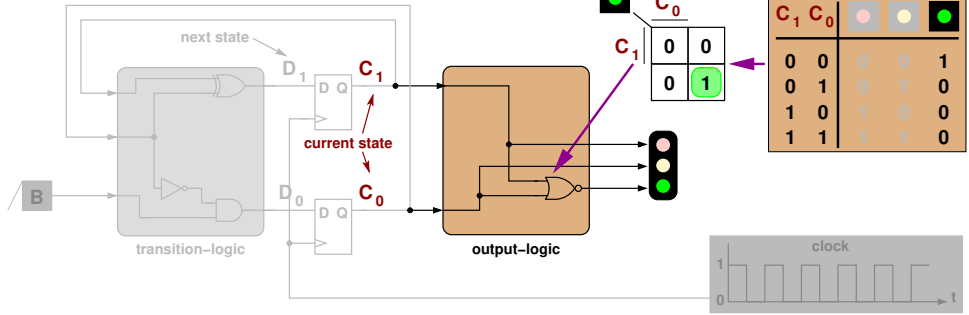



Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

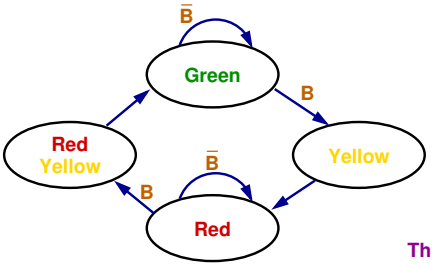
Transition–table:

| B | $C_1$ $C_0$ | $D_1$ $D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |
| – | 1 1 | 0 0 |
| 0 | 1 0 | 1 0 |
| 1 | 1 0 | 1 1 |
| 0 | 0 0 | 0 0 |
| 1 | 0 0 | 0 1 |



# Moore FSM

**State Diagram:**




**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

**Transition–table:**

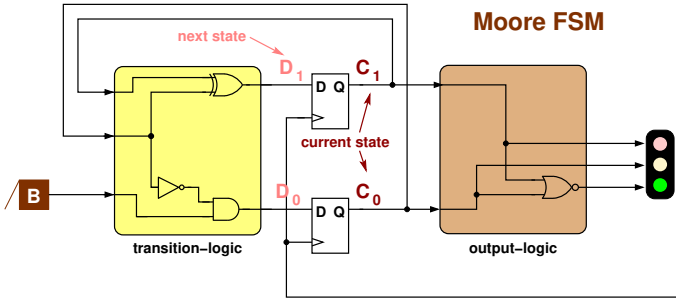
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

**This type of FSM is called a:**

**Moore FSM**

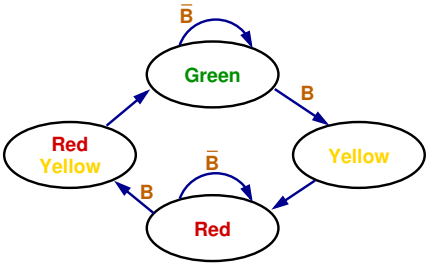
**Output–table:**

| $C_1$ | $C_0$ | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |



# Mealy FSM

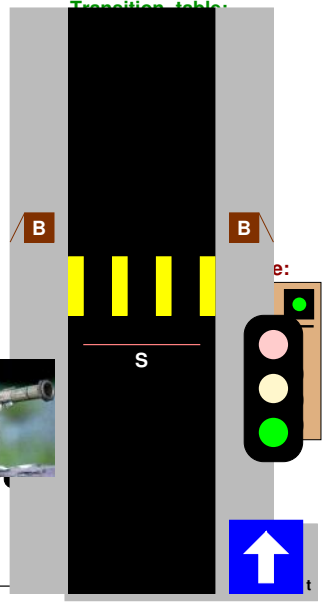
State Diagram:



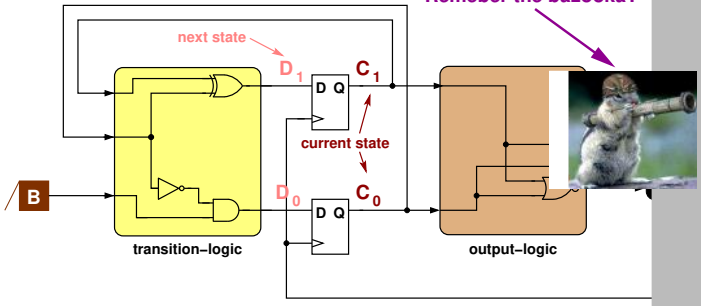

Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

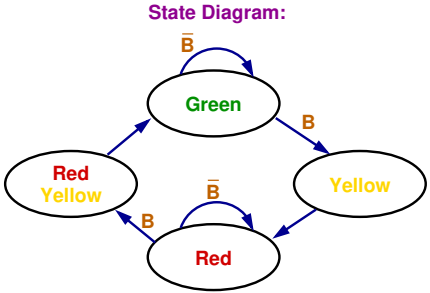
Transition–table:




Remeber the bazooka?

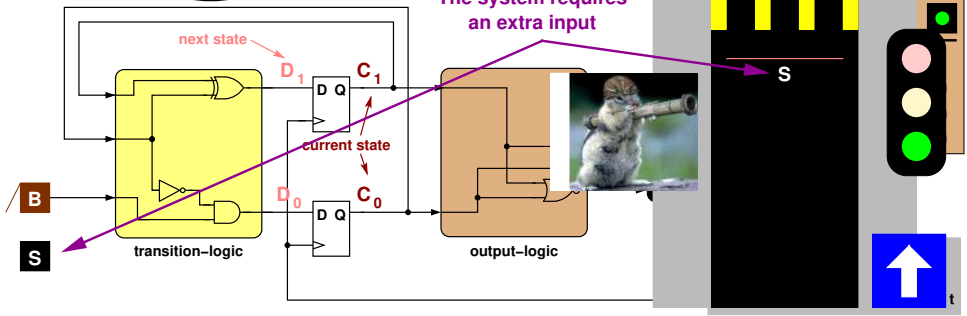


# Mealy FSM



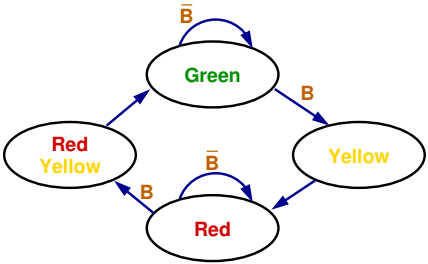

Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |



# Mealy FSM

**State Diagram:**




**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

**Transition–table:**

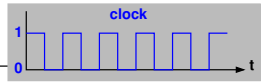
| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

**Output–table:**

| $C_1$ | $C_0$ | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

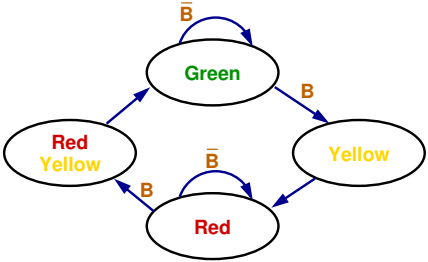
**And has an extra output**



# Mealy FSM

**State Diagram:**




**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

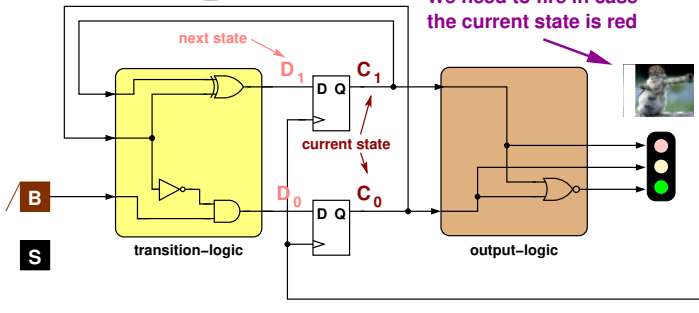
**Transition–table:**

| B | $C_1$ $C_0$ | $D_1$ $D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |
| – | 1 1 | 0 0 |
| 0 | 1 0 | 1 0 |
| 1 | 1 0 | 1 1 |
| 0 | 0 0 | 0 0 |
| 1 | 0 0 | 0 1 |

We need to fire in case the current state is red

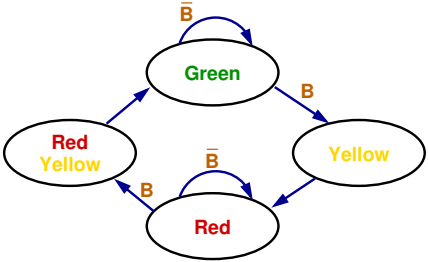
**Output–table:**

| $C_1$ $C_0$ | | | |
|---|---|---|---|
| 0 0 | 0 | 0 | 1 |
| 0 1 | 0 | 1 | 0 |
| 1 0 | 1 | 0 | 0 |
| 1 1 | 1 | 1 | 0 |



# Mealy FSM

**State Diagram:**




**Code–table:**

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

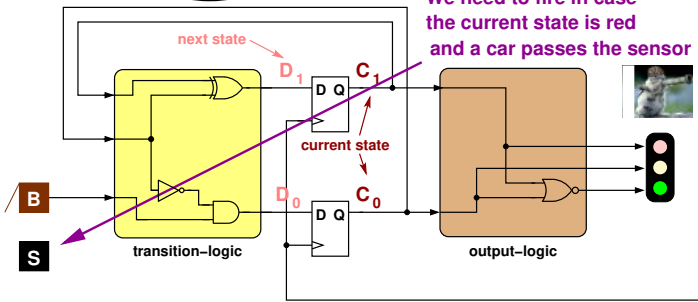
**Transition–table:**

| B | $C_1$ $C_0$ | $D_1$ $D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |
| – | 1 1 | 0 0 |
| 0 | 1 0 | 1 0 |
| 1 | 1 0 | 1 1 |
| 0 | 0 0 | 0 0 |
| 1 | 0 0 | 0 1 |

**Output–table:**

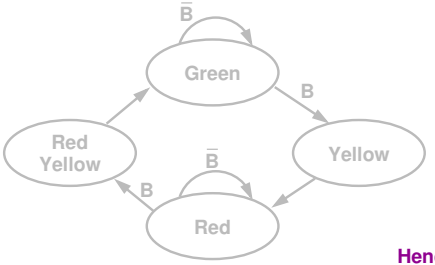
| $C_1$ $C_0$ | 🔴 | 🟡 | 🟢 |
|---|---|---|---|
| 0 0 | 0 | 0 | 1 |
| 0 1 | 0 | 1 | 0 |
| 1 0 | 1 | 0 | 0 |
| 1 1 | 1 | 1 | 0 |

We need to fire in case the current state is red and a car passes the sensor



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

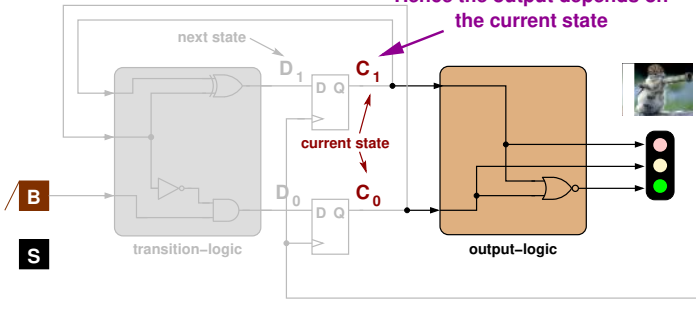
Transition–table:

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

Hence the output depends on the current state

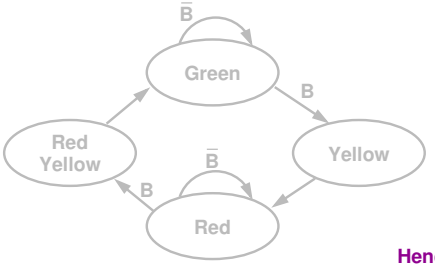
Output–table:

| $C_1$ | $C_0$ | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

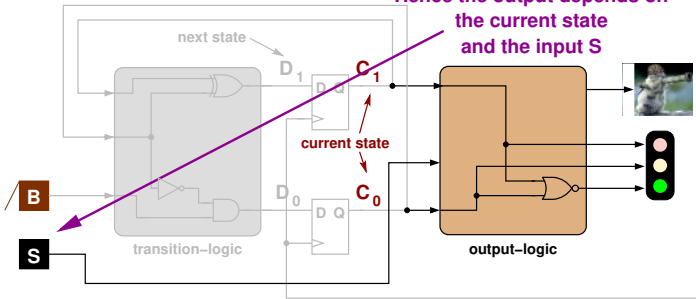
Transition–table:

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

Hence the output depends on the current state and the input S

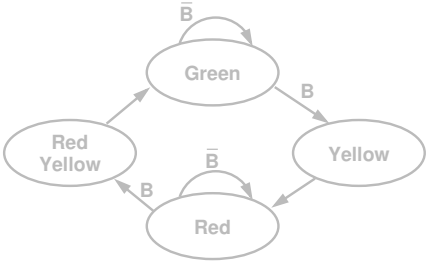
Output–table:

| $C_1$ | $C_0$ | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

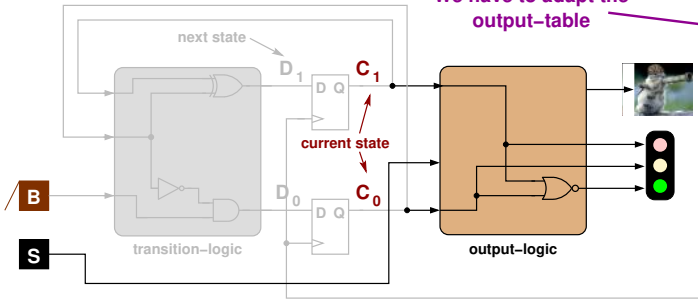
Transition–table:

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

We have to adapt the output–table

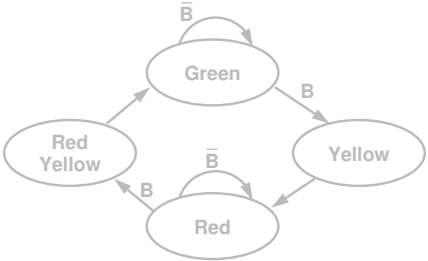
Output–table:

| $C_1$ | $C_0$ | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

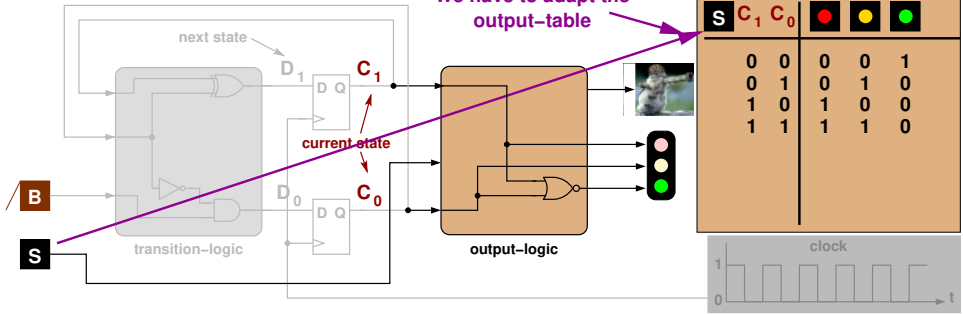
Transition–table:

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

We have to adapt the output–table

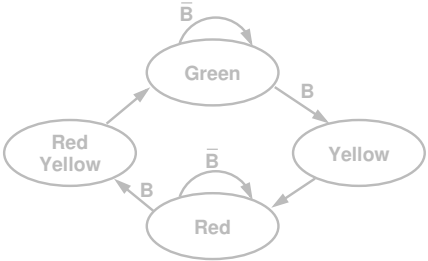
Output–table:

| S | $C_1$ | $C_0$ | | | |
|---|---|---|---|---|---|
| 0 | 0 | | 0 | 0 | 1 |
| 0 | 1 | | 0 | 1 | 0 |
| 1 | 0 | | 1 | 0 | 0 |
| 1 | 1 | | 1 | 1 | 0 |



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

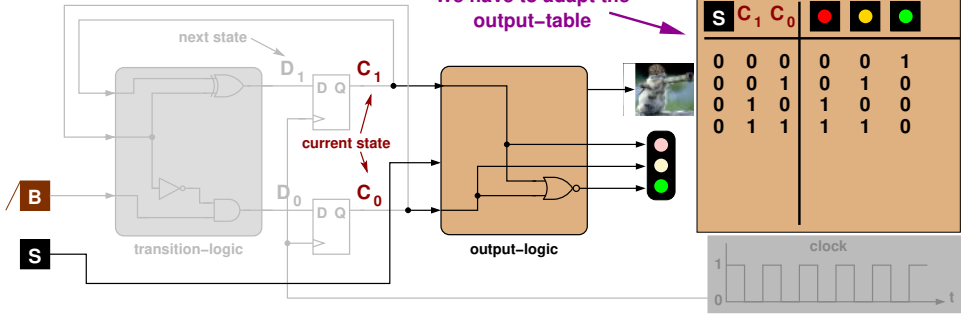
Transition–table:

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

We have to adapt the output–table

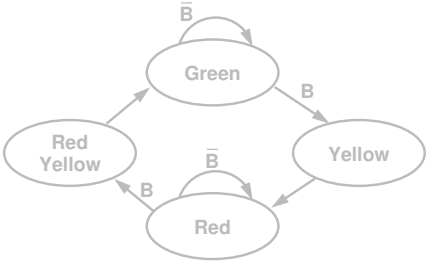
Output–table:

| S | $C_1$ | $C_0$ | | | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 |



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

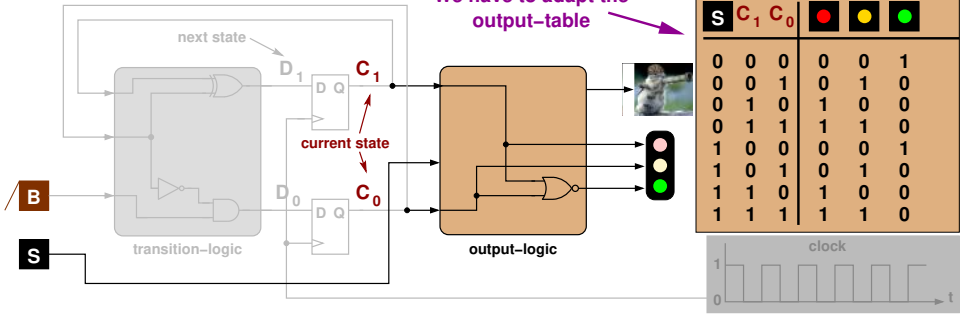
Transition–table:

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

We have to adapt the output–table

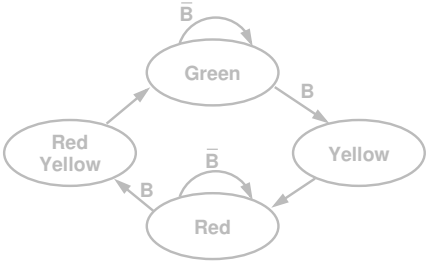
Output–table:

| S | $C_1$ | $C_0$ | | | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 |



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

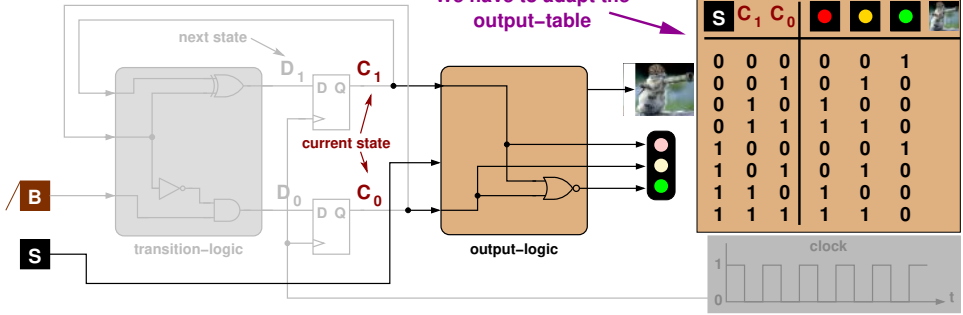
Transition–table:

| B | $C_1$ $C_0$ | $D_1$ $D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |
| – | 1 1 | 0 0 |
| 0 | 1 0 | 1 0 |
| 1 | 1 0 | 1 1 |
| 0 | 0 0 | 0 0 |
| 1 | 0 0 | 0 1 |

We have to adapt the output–table

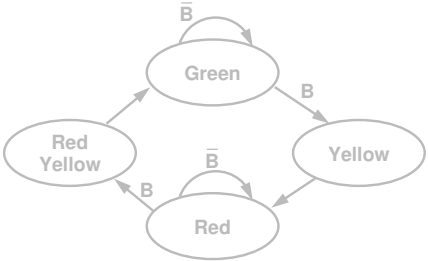
Output–table:

| S | $C_1$ | $C_0$ | | | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 |



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

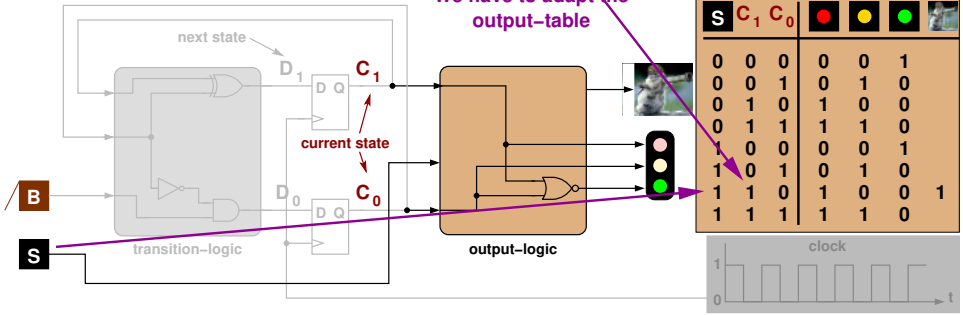
Transition–table:

| B | $C_1$ $C_0$ | $D_1$ $D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |
| – | 1 1 | 0 0 |
| 0 | 1 0 | 1 0 |
| 1 | 1 0 | 1 1 |
| 0 | 0 0 | 0 0 |
| 1 | 0 0 | 0 1 |

We have to adapt the output–table

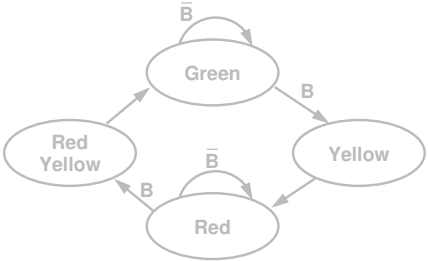
Output–table:

| S | $C_1$ $C_0$ | | | | |
|---|---|---|---|---|---|
| 0 | 0 0 | 0 | 0 | 1 | |
| 0 | 0 1 | 0 | 1 | 0 | |
| 0 | 1 0 | 1 | 0 | 0 | |
| 0 | 1 1 | 1 | 1 | 0 | |
| 1 | 0 0 | 0 | 0 | 1 | |
| 1 | 0 1 | 0 | 1 | 0 | |
| 1 | 1 0 | 1 | 0 | 0 | 1 |
| 1 | 1 1 | 1 | 1 | 0 | |



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

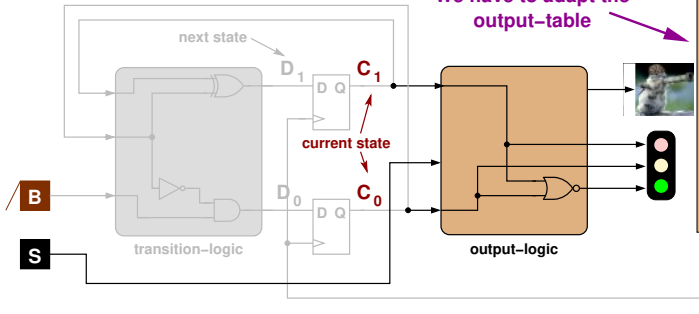
Transition–table:

| B | $C_1$ $C_0$ | $D_1$ $D_0$ |
|---|---|---|
| – | 0 1 | 1 0 |
| – | 1 1 | 0 0 |
| 0 | 1 0 | 1 0 |
| 1 | 1 0 | 1 1 |
| 0 | 0 0 | 0 0 |
| 1 | 0 0 | 0 1 |

We have to adapt the output–table

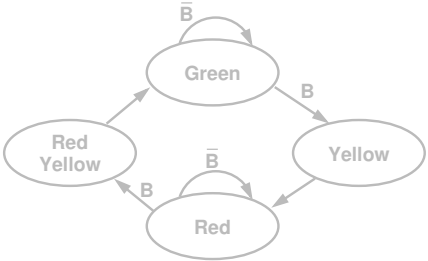
Output–table:

| S | $C_1$ | $C_0$ | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 |



# Mealy FSM

State Diagram:




Code–table:

| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

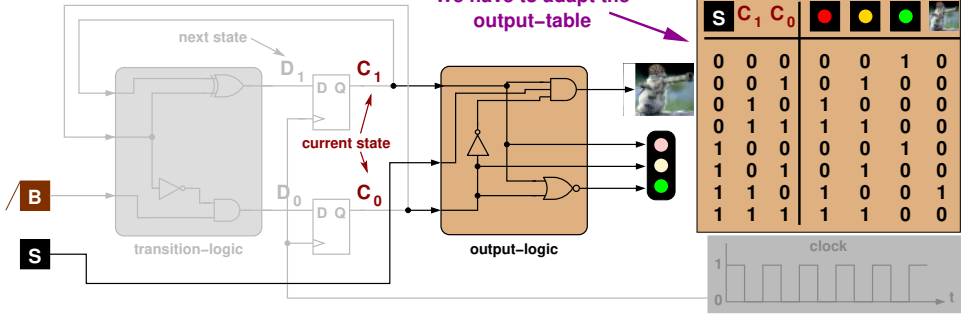
Transition–table:

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

We have to adapt the output–table

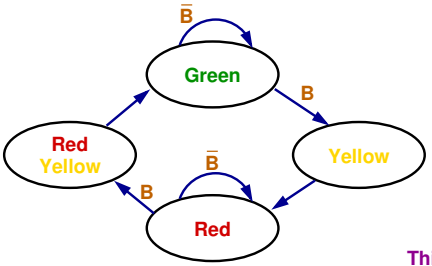
Output–table:

| S | $C_1$ | $C_0$ | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 |



# Mealy FSM

**State Diagram:**




**Code–table:**

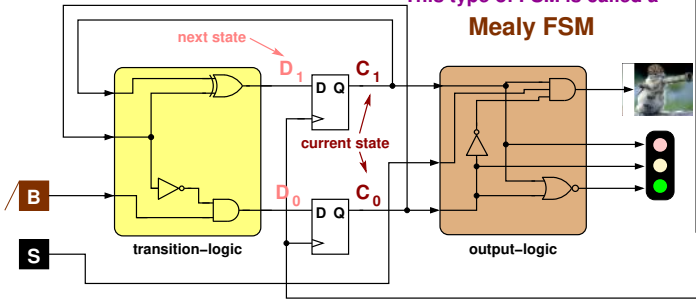
| State | Code |
|---|---|
| Green | $00_b$ |
| Yellow | $01_b$ |
| Red | $10_b$ |
| Red Yellow | $11_b$ |

**Transition–table:**

| B | $C_1$ | $C_0$ | $D_1$ | $D_0$ |
|---|---|---|---|---|
| – | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |

This type of FSM is called a

**Mealy FSM**




**Output–table:**

| S | $C_1$ | $C_0$ | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 |