

# CSE 460: VLSI Design (Lab)

Experiment 3, Part 2:  
Sequential Circuits Real-life Projects  
**Vending Machine**

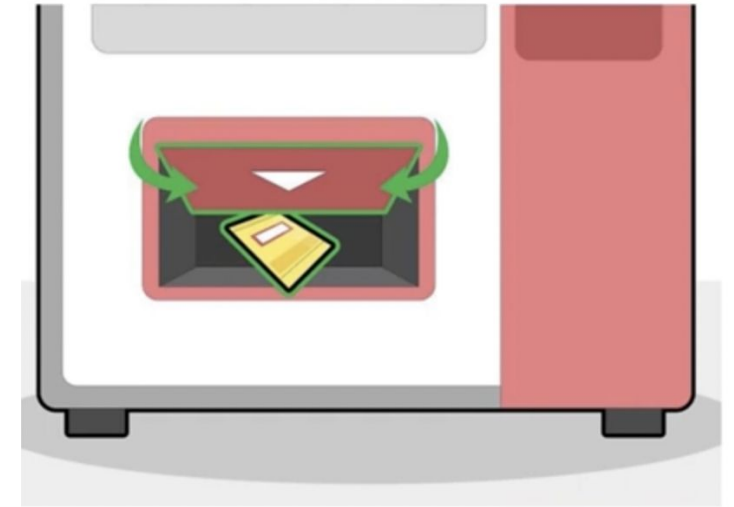


Inspiring Excellence

# What is Vending Machine?

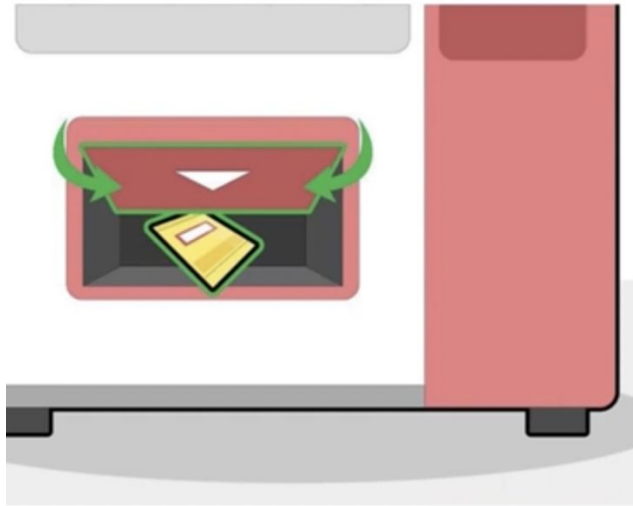


# What is Vending Machine?



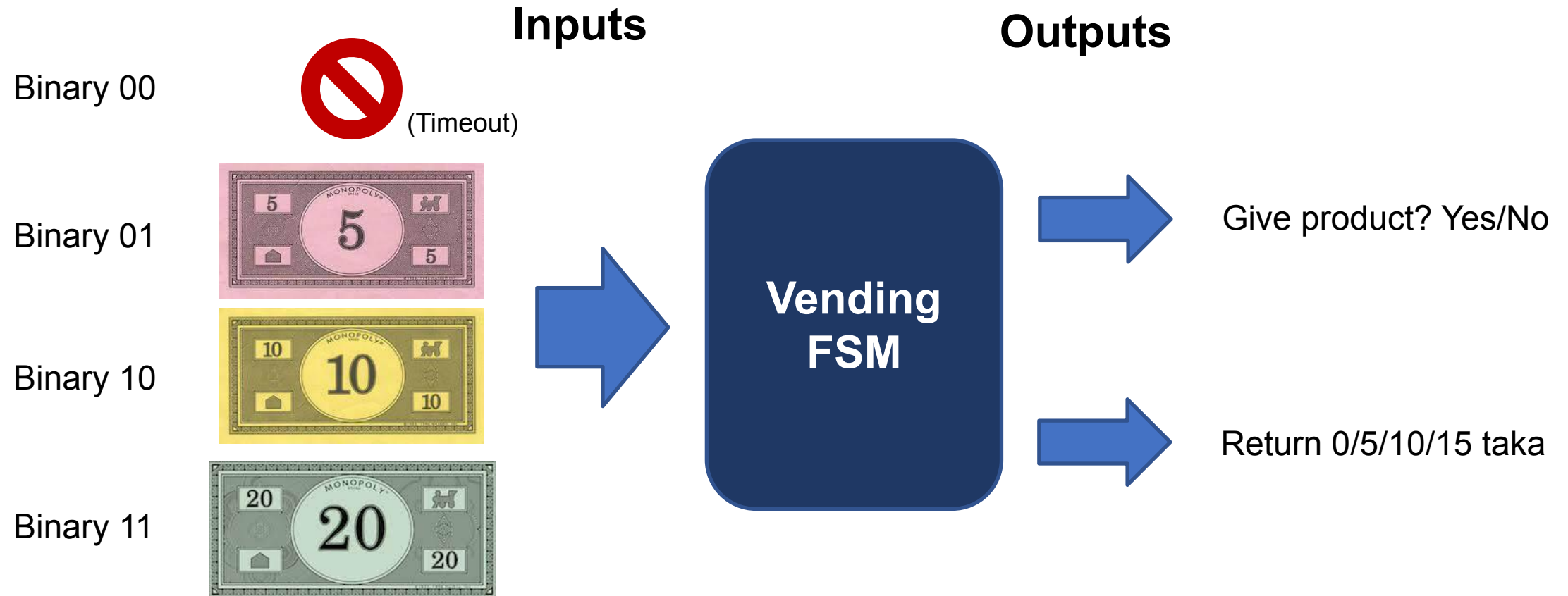


# What is Vending Machine?



Inspiring Excellence

# Application Description

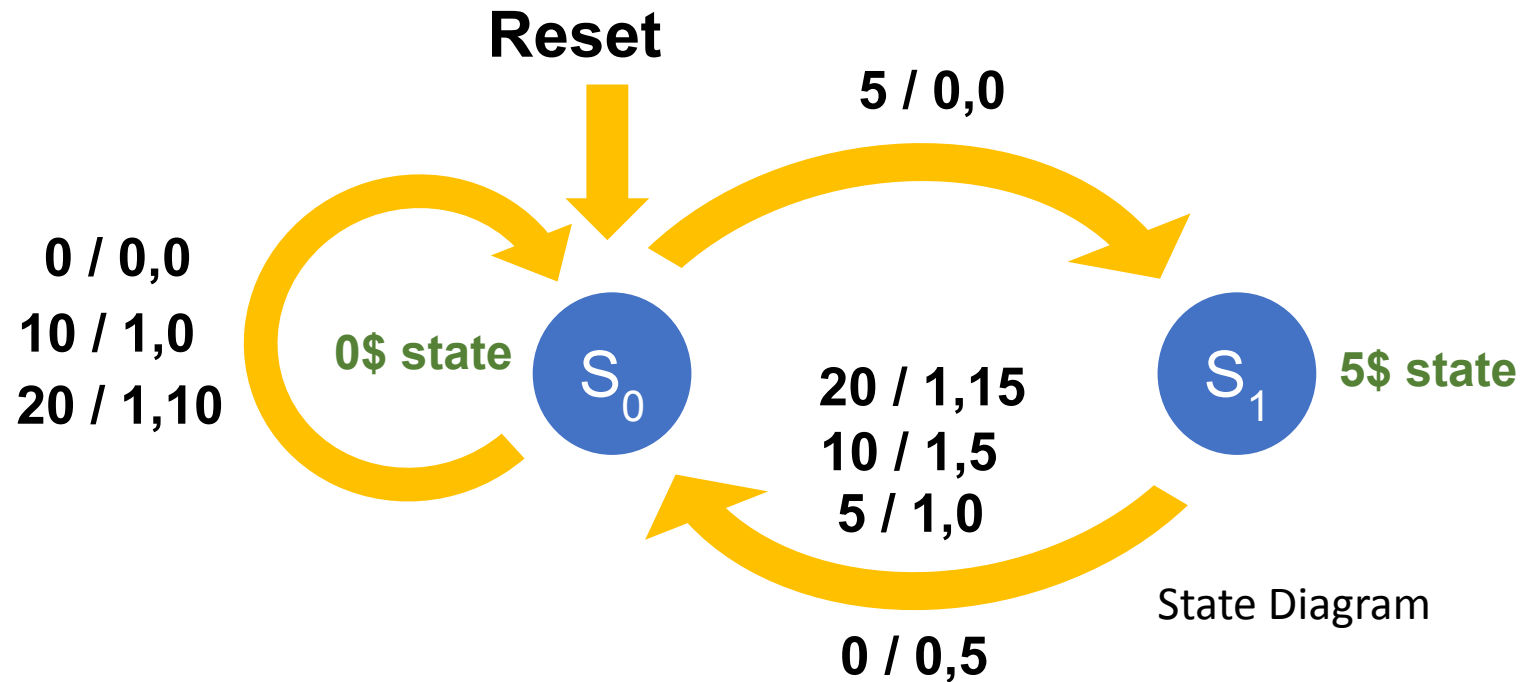




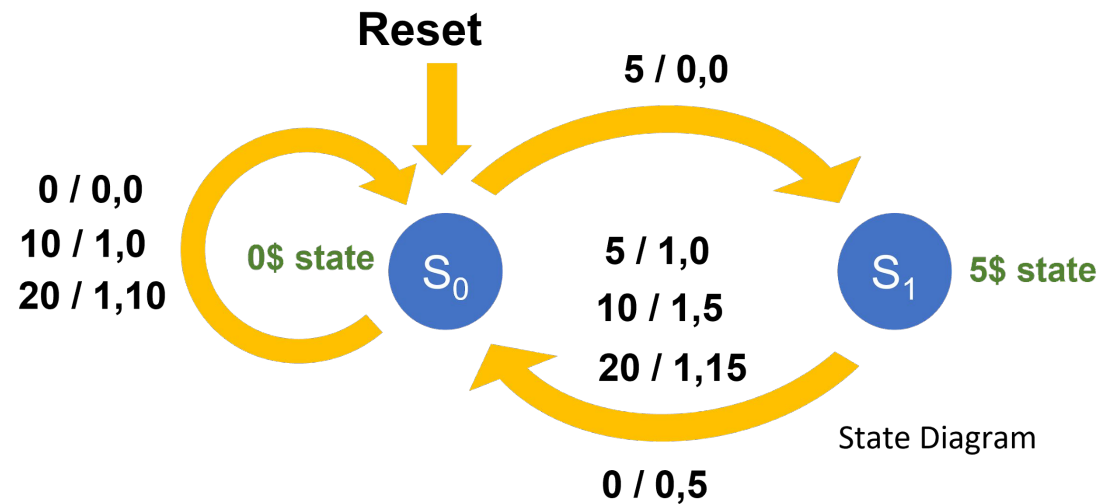
# 10 Tk Product Case (State Diagram)

Money in: 0, 5, 10, 20 Taka

Money in/Purchase, Change



# 10 Tk Product Case (State Table)



Present State	Next State				Output							
	$Y_2Y_1$				$z$				$C_2C_1$			
$(y_2y_1)$	$(w_2w_1)$				$(w_2w_1)$				$c(w_2w_1)$			
	00	01	10	11	00	01	10	11	00	01	10	11
S0	S0	S1	S0	S0	0	0	1	1	00	00	00	10
S1	S0	S0	S0	S0	0	1	1	1	01	00	01	11

State Table

Present State	Next State				Output							
	$Y_2Y_1$				$z$				$C_2C_1$			
$(y_2y_1)$	$(w_2w_1)$				$(w_2w_1)$				$c(w_2w_1)$			
	00	01	10	11	00	01	10	11	00	01	10	11
00	00	01	00	00	0	0	1	1	00	00	00	10
01	00	00	00	00	0	1	1	1	01	00	01	11

State Assigned Table



# 10 Tk Product Case (Code Snippet)

Present State (y <sub>2</sub> y <sub>1</sub> )	Next State Y <sub>2</sub> Y <sub>1</sub>				Output							
	Y <sub>2</sub> Y <sub>1</sub>				z				C <sub>2</sub> C <sub>1</sub>			
	(w <sub>2</sub> w <sub>1</sub> )				(w <sub>2</sub> w <sub>1</sub> )				c(w <sub>2</sub> w <sub>1</sub> )			
	00	01	10	11	00	01	10	11	00	01	10	11
00	00	01	00	00	0	0	1	1	00	00	00	10
01	00	00	00	00	0	1	1	1	01	00	01	11

State Assigned Table

```
state0: if(cash_in == 2'b00)
    begin
        next_state = state0;
        purchase = 0;
        cash_return = 0;
    end
else if(cash_in == 2'b01)
    begin
        next_state=state1;
        purchase = 0;
        cash_return = 0;
    end
else if(cash_in == 2'b10)
    begin
        next_state=state0;
        purchase=1;
        cash_return=0;
    end
else if(cash_in == 2'b11)
    begin
        next_state = state0;
        purchase = 1;
        cash_return = 2'b10;
    end
end
```



# 10 Tk Product Case (Code Snippet)

Present State (y <sub>2</sub> y <sub>1</sub> )	Next State Y <sub>2</sub> Y <sub>1</sub>				Output							
	Y <sub>2</sub> Y <sub>1</sub>				z				C <sub>2</sub> C <sub>1</sub>			
	(w <sub>2</sub> w <sub>1</sub> )				(w <sub>2</sub> w <sub>1</sub> )				c(w <sub>2</sub> w <sub>1</sub> )			
	00	01	10	11	00	01	10	11	00	01	10	11
00	00	01	00	00	0	0	1	1	00	00	00	10
01	00	00	00	00	0	1	1	1	01	00	01	11

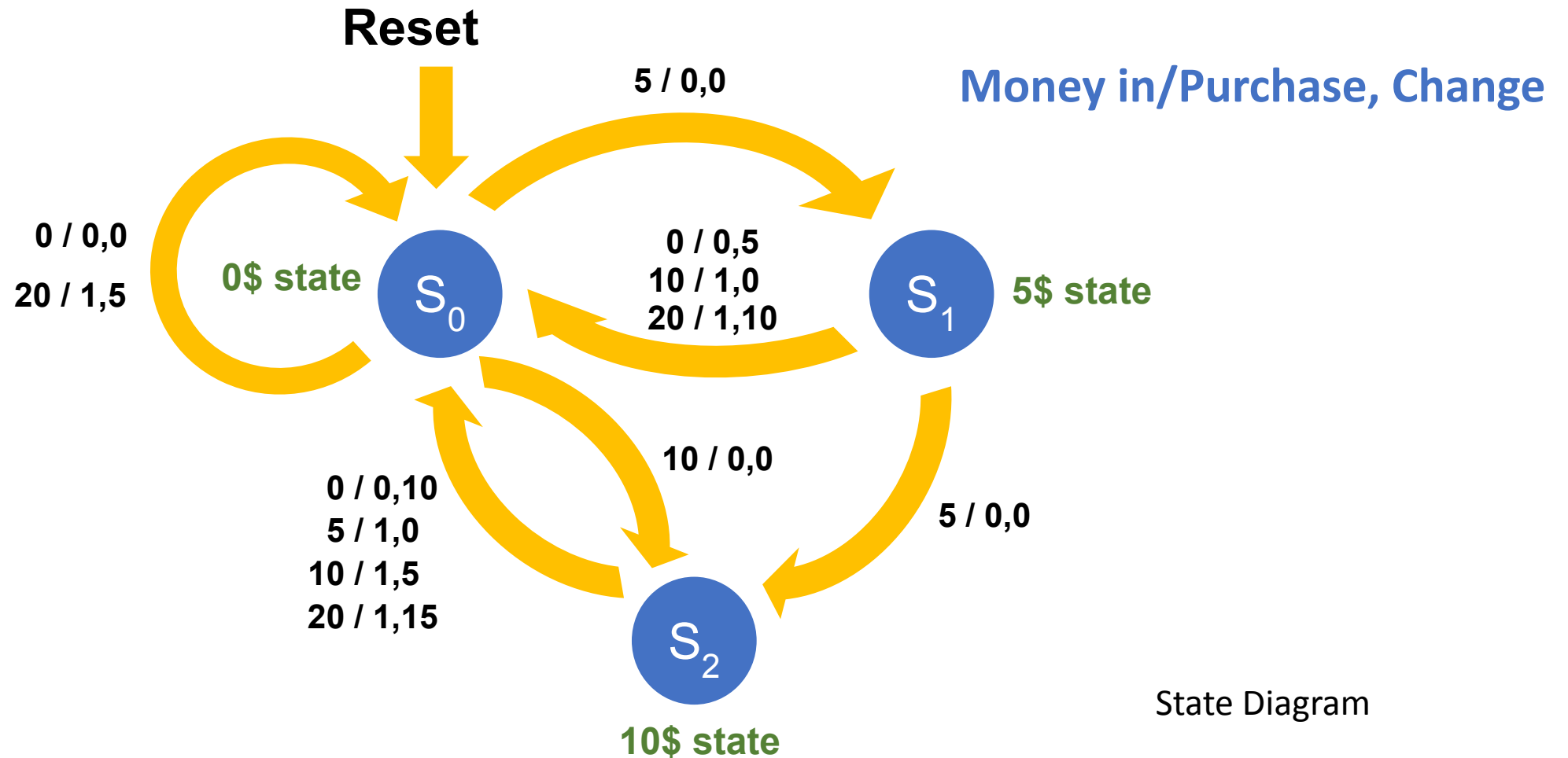
State Assigned Table

```
state1: if (cash_in == 2'b00)
begin
    next_state = state0;
    purchase = 0;
    cash_return = 2'b01;
end
else if (cash_in == 2'b01)
begin
    next_state = state0;
    purchase = 1;
    cash_return = 0;
end
else if (cash_in == 2'b10)
begin
    next_state = state0;
    purchase = 1;
    cash_return = 2'b01;
end
else if (cash_in == 2'b11)
begin
    next_state = state0;
    purchase = 1;
    cash_return = 2'b11;
end
endcase

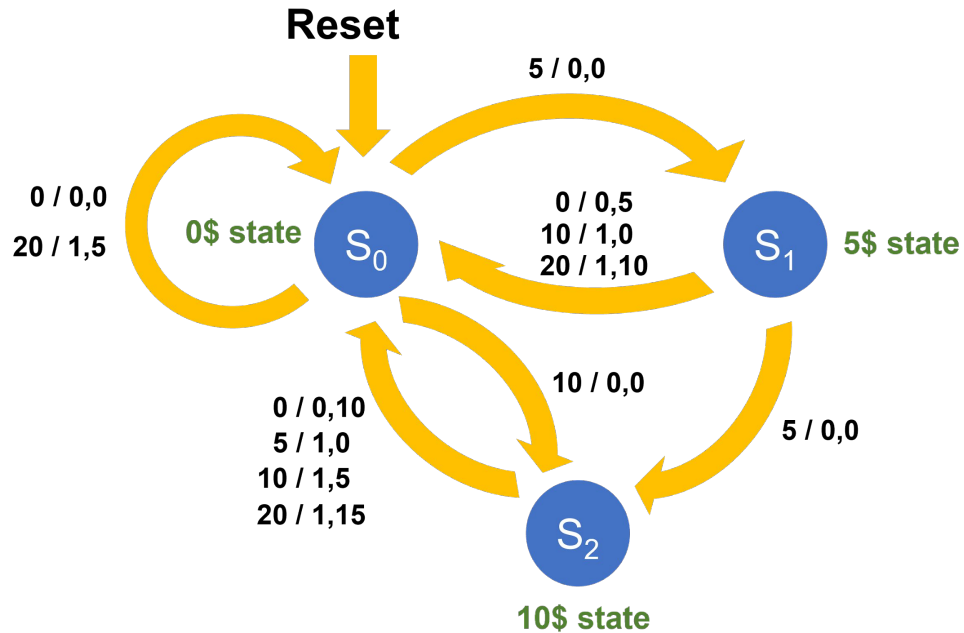
always @(posedge rst, posedge clk)
if (rst == 1)
    current_state <= state0;
else
    current_state <= next_state;
endmodule
```



# 15 Tk Product Case (State Diagram)



# 15 Tk Product Case (State Table)



Present State	Next State				Output							
	Y <sub>2</sub> Y <sub>1</sub>				z				C <sub>2</sub> C <sub>1</sub>			
	(y <sub>2</sub> y <sub>1</sub> )				(w <sub>2</sub> w <sub>1</sub> )				c(w <sub>2</sub> w <sub>1</sub> )			
	00	01	10	11	00	01	10	11	00	01	10	11
S <sub>0</sub>	S <sub>0</sub>	S <sub>1</sub>	S <sub>2</sub>	S <sub>0</sub>	0	0	0	1	00	00	00	01
S <sub>1</sub>	S <sub>0</sub>	S <sub>2</sub>	S <sub>0</sub>	S <sub>0</sub>	0	0	1	1	01	00	00	10
S <sub>2</sub>	S <sub>0</sub>	S <sub>0</sub>	S <sub>0</sub>	S <sub>0</sub>	0	1	1	1	10	00	01	11

State Table

# 15 Tk Product Case (State Assigned Table)

Present State	Next State				Output							
	$Y_2Y_1$				$z$				$C_2C_1$			
$(y_2y_1)$	$(w_2w_1)$				$(w_2w_1)$				$c(w_2w_1)$			
	00	01	10	11	00	01	10	11	00	01	10	11
$S_0$	$S_0$	$S_1$	$S_2$	$S_0$	0	0	0	1	00	00	00	01
$S_1$	$S_0$	$S_2$	$S_0$	$S_0$	0	0	1	1	01	00	00	10
$S_2$	$S_0$	$S_0$	$S_0$	$S_0$	0	1	1	1	10	00	01	11

State Table

Present State	Next State				Output							
	$Y_2Y_1$				$z$				$C_2C_1$			
$(y_2y_1)$	$(w_2w_1)$				$(w_2w_1)$				$(w_2w_1)$			
	00	01	10	11	00	01	10	11	00	01	10	11
<b>00</b>	00	01	10	00	0	0	0	1	00	00	00	01
<b>01</b>	00	10	00	00	0	0	1	1	01	00	00	10
<b>10</b>	00	00	00	00	0	1	1	1	10	00	01	11
<b>11</b>	d	d	d	d	d	d	d	d	d	d	d	d

State Assigned Table





# 15 Tk Product Case (Code Snippet)

Present State	Next State				Output							
	$Y_2Y_1$				$z$				$C_2C_1$			
$(y_2y_1)$	$(w_2w_1)$				$(w_2w_1)$				$(w_2w_1)$			
	00	01	10	11	00	01	10	11	00	01	10	11
00	00	01	10	00	0	0	0	1	00	00	00	01
01	00	10	00	00	0	0	1	1	01	00	00	10
10	00	00	00	00	0	1	1	1	10	00	01	11
11	d	d	d	d	d	d	d	d	d	d	d	d

State Assigned Table

```

state0: if(cash_in == 2'b00)
    begin
        next_state = state0;
        purchase = 0;
        cash_return = 0;
    end
else if(cash_in == 2'b01)
    begin
        next_state = state1;
        purchase = 0;
        cash_return = 0;
    end
else if(cash_in == 2'b10)
    begin
        next_state = state2;
        purchase = 0;
        cash_return = 0;
    end
else if(cash_in == 2'b11)
    begin
        next_state = state0;
        purchase = 1;
        cash_return = 2'b01;
    end

```





# 15 Tk Product Case (Code Snippet)

Present State	Next State				Output							
	Y <sub>2</sub> Y <sub>1</sub>				z				C <sub>2</sub> C <sub>1</sub>			
(y <sub>2</sub> y <sub>1</sub> )	(w <sub>2</sub> w <sub>1</sub> )				(w <sub>2</sub> w <sub>1</sub> )				(w <sub>2</sub> w <sub>1</sub> )			
	00	01	10	11	00	01	10	11	00	01	10	11
00	00	01	10	00	0	0	0	1	00	00	00	01
01	00	10	00	00	0	0	1	1	01	00	00	10
10	00	00	00	00	0	1	1	1	10	00	01	11
11	d	d	d	d	d	d	d	d	d	d	d	d

State Assigned Table

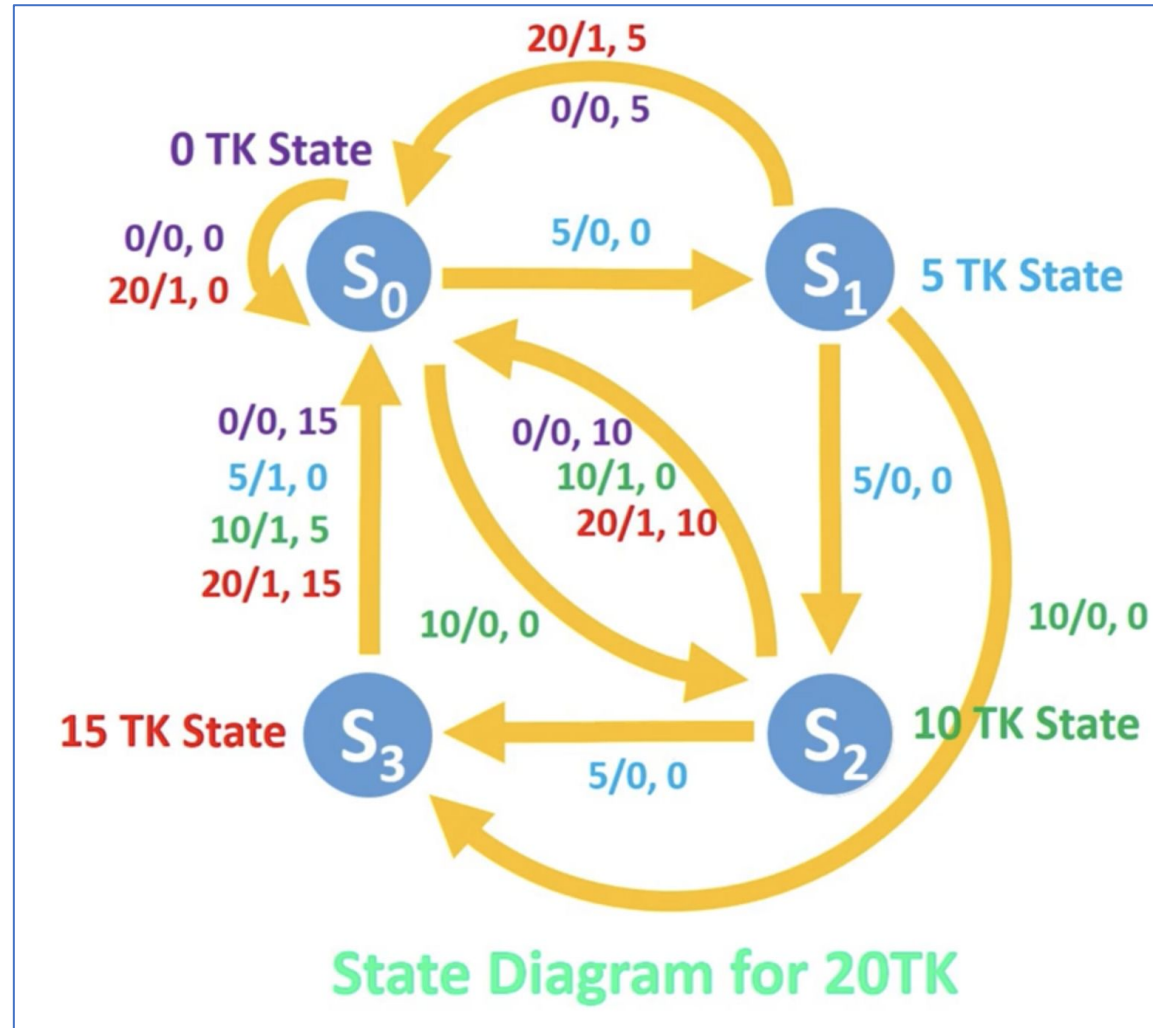
```

state2: if(cash_in == 2'b00)
    begin
        next_state = state0;
        purchase = 0;
        cash_return = 2'b10;
    end
else if(cash_in == 2'b01)
    begin
        next_state = state0;
        purchase = 1;
        cash_return = 0;
    end
else if(cash_in == 2'b10)
    begin
        next_state = state0;
        purchase = 1;
        cash_return = 2'b01;
    end
else if(cash_in == 2'b11)
    begin
        next_state = state0;
        purchase = 1;
        cash_return = 2'b11;
    end

```



# 20 Tk Product Case (State Diagram)



# Thank you!



Inspiring Excellence

7/1/2022

CSE 460: VLSI Design

17