EEL3701

# Menu

- State Machine Design
  - >Design example: Sequence Detector (using Moore Machine)
  - >Design example: Sequence Detector (using Mealy Machine)
  - >Implementation




EEL3701

# Classical Design

**[Example]** Design a *Sequence Detector/Acceptor* to accept X = 0 1 $0^*$ 1
where $0^*$ = { λ(nil), 0, 00, 000, 0000, ....... }

- Thus, Sequence {011}, {0101}, {01001}, {010001} are acceptable, but {0111} is not. Z is the output; X and CLK are the inputs.

X = 0 1 0 0 0 1 0 1 0 ......
Z = 0 0 0 0 0 1 0 1 0 ......
CLK= 1 2 3 4 5 6 7 8 9 ......

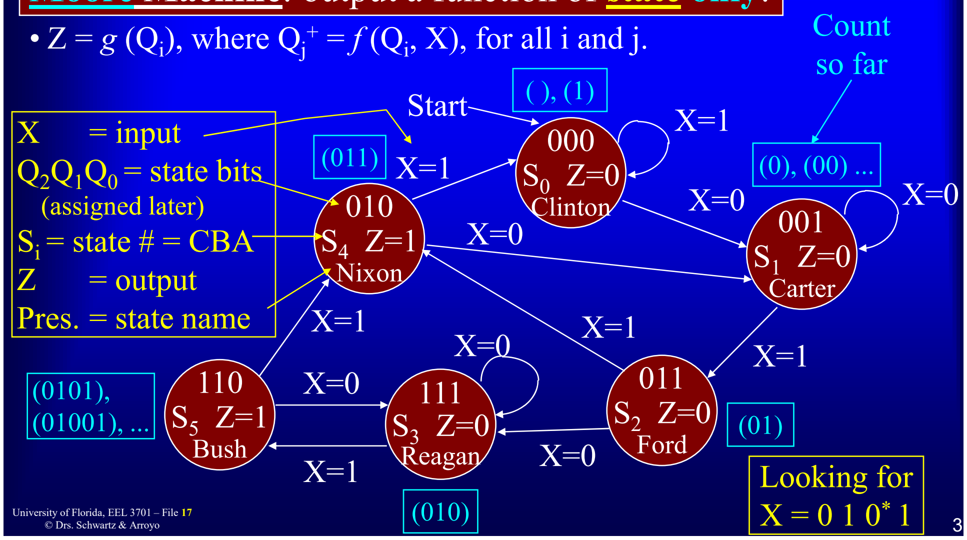
- **STEP 1**: State Diagram - This comes in two flavors:
  (1) Moore Machine - Outputs depend **only** on present state
  (2) Mealy Machine - Outputs depend on state **and** present inputs

EEL3701

## Moore Machine Example

**Moore Machine**: output a function of **state only**!

- $Z = g\,(Q_i)$, where $Q_j^+ = f\,(Q_i, X)$, for all i and j.






3

EEL3701

## State Machine (SM) Design: Background

- **Explanation of the state diagram**

(a) Start with a "start" state, $S_0$. We receive an input X, and we produce an output Z. If we get X=0, it might be a part of the desired sequence. However, if we get X=1, it is definitely not a part of the desired sequence. We are looking for X=0. Let's call the state where we got what we want (X=0) $S_1$ since it will be a different state from the original state. ($\therefore$ $S_0 \equiv$ "looking for X=0", $S_1 \equiv$ "we got it.") The output for $S_0$ is Z=0 because no detection has taken place.

Looking for X = 0 1 0* 1



4

EEL3701

## SM Design: Background

- **Explanation of the state diagram**

(b) Now in state S1. We are looking for a 1. If we get it, we might be detecting the sequence. However, if we get a 0, we stay at the state where we are looking for a 1. Call the state where we got the 1 we are looking for $S_2$. The output for $S_1$ is 0 since we've not detected all of the sequence yet.

(c) From $S_2$, if we get a 1, we succeed in detecting {011}. Call this state $S_4$ (success!). If we get a 0, we might still succeed in detecting a part of {010......01}. Call this state $S_3$. The output so far is still 0.

Looking for
$X = 0\ 1\ 0^*\ 1$

University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo

5

5

EEL3701

## SM Design: Background

- **Explanation of the state diagram**

(d) From $S_3$, if we get a 0, we might still be working on {010......01} so stay there. If we get a 1, we detected one of the good guys so go to $S_5$ (success again!). Why not $S_4$? $S_4$ was for {011}, but now we have a sequence of intervening 0's followed by a 1. This is not {011}. ***Treat them separately!*** Output is 0 because no sequence has been detected.

Looking for
$X = 0\ 1\ 0^*\ 1$

University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo

6

6

EEL3701

## SM Design: Background

- **Explanation of the state diagram**

(e) From $S_5$, if we get a 0, we can go back to $S_3$ because we would be detecting {0101} or {010......01}. If we get a 1, there was no intervening zeros, we have {011} (success! go to $S_4$). Output is 1 since we detected in $S_5$ the pattern {010......01}.

(f) From $S_4$, that is "after {011} is received," another 1 sends us to "looking for the 1st zero," that is $S_0$. A 0 sends us to "looking for a 1 after the 1st zero was detected," that is, $S_1$. Output is 1 since we detected {011}.

Looking for $X = 0\ 1\ 0^*\ 1$

University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo 7

7

EEL3701

## SM Design: Background

- We are ∴ finished with the state diagram since we have gone through all the possibilities. "Designing" is an ***art with science***. It is not totally recipe-driven.
- **STEP 2**: Assign State Identifiers using binary patterns and/or names.

  Since we have 6 states, we need 3 bits (3 FF's) to represent the $[(2^2{=}4) < 6 \le (2^3 = 8)]$ possibilities.

Looking for $X = 0\ 1\ 0^*\ 1$

University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo 8

8

EEL3701

## SM Design: Background

- Make a state assignment table.

**Important State bits**

| Present State | Names | $Q_2$ | $Q_1$ | $Q_0$ | Next State(s) |
|---|---|---|---|---|---|
| $S_0$ | Clinton | 0 | 0 | 0 | $(S_0)$ $S_1$ |
| $S_1$ | Carter | 0 | 0 | 1 | $(S_1)$ $S_2$ |
| $S_2$ | Ford | 0 | 1 | 1 | $S_3$ $S_4$ |
| $S_3$ | Reagan | 1 | 1 | 1 | $(S_3)$ $S_5$ |
| $S_4$ | Nixon | 0 | 1 | 0 | $S_0$ $S_1$ |
| $S_5$ | Bush | 1 | 1 | 0 | $S_3$ $S_4$ |

- *Turns out that the assignment of bits has an effect on the complexity of the hardware realization. The details are beyond the scope of EEL 3701.*

Looking for $X = 0\ 1\ 0^*\ 1$



9

EEL3701

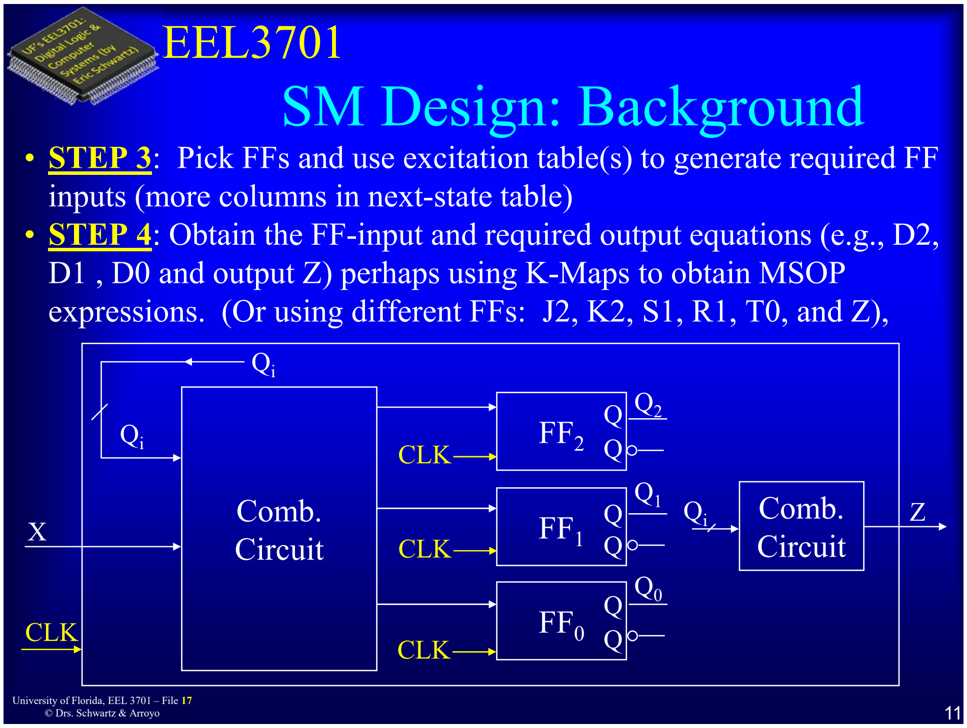
## SM Design: Background

- **Heuristic** -- Try to pick the state assignments so that only 1 bit changes from one state to the next state (the same used in Gray Code). This is not always possible, but we can often get close.
- You typically start by assigning $S_0$ to an arbitrary pattern, say, 000 and pick from that point on.
- This is **NOT** required in 3701, but you will likely see it if you take EEL4712.

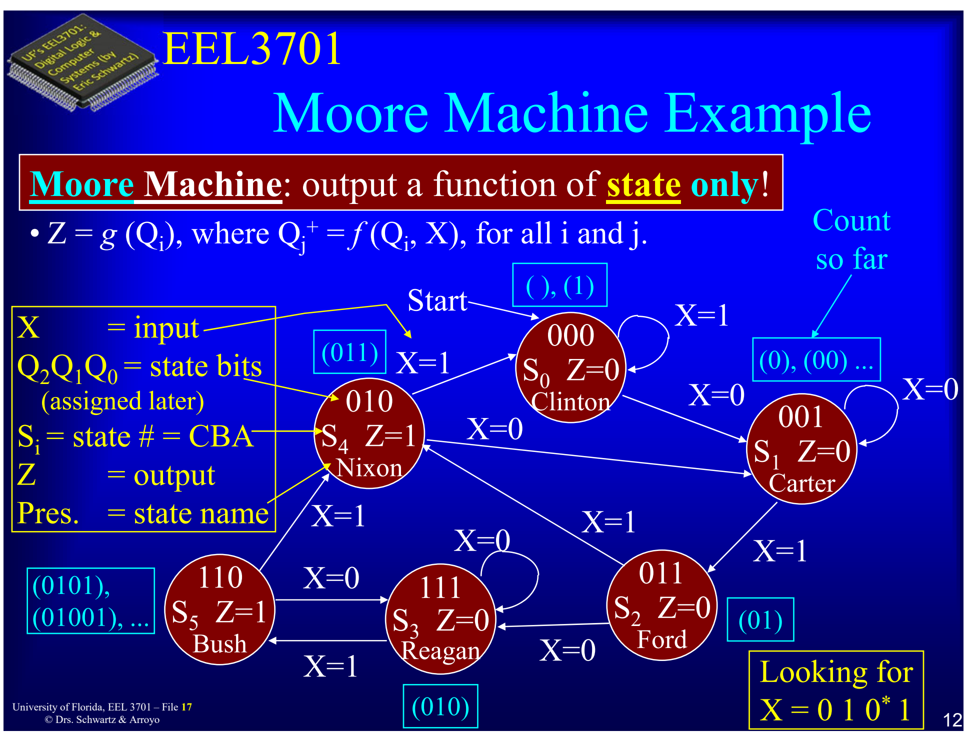
**NOTE**: Using the Gray Code-like table, we do not use code 100 or 101 because we had only 6 states out of 8 patterns (000 ~ 111). These codes, 2 leftovers, cannot happen.
∴ *They are "Don't Cares" in the K-Maps.*



10


EEL3701
SM Design: Background
• STEP 3: Pick FFs and use excitation table(s) to generate required FF inputs (more columns in next-state table)
• STEP 4: Obtain the FF-input and required output equations (e.g., D2, D1 , D0 and output Z) perhaps using K-Maps to obtain MSOP expressions. (Or using different FFs: J2, K2, S1, R1, T0, and Z),
Qi
Qi
X
CLK
Comb. Circuit
CLK
CLK
CLK
FF2
FF1
FF0
Q
Q2
Q1
Q0
Qi
Comb. Circuit
Z
University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo
11

11


EEL3701
Moore Machine Example
Moore Machine: output a function of state only!
• Z = g (Qi), where Qj+ = f (Qi, X), for all i and j.
Count so far
X = input
Q2Q1Q0 = state bits
(assigned later)
Si = state # = CBA
Z = output
Pres. = state name
Start
( ), (1)
(011)
(0), (00) ...
000
S0 Z=0
Clinton
001
S1 Z=0
Carter
010
S4 Z=1
Nixon
011
S2 Z=0
Ford
110
S5 Z=1
Bush
111
S3 Z=0
Reagan
X=1
X=0
(0101), (01001), ...
(01)
(010)
Looking for X = 0 1 0* 1
University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo
12

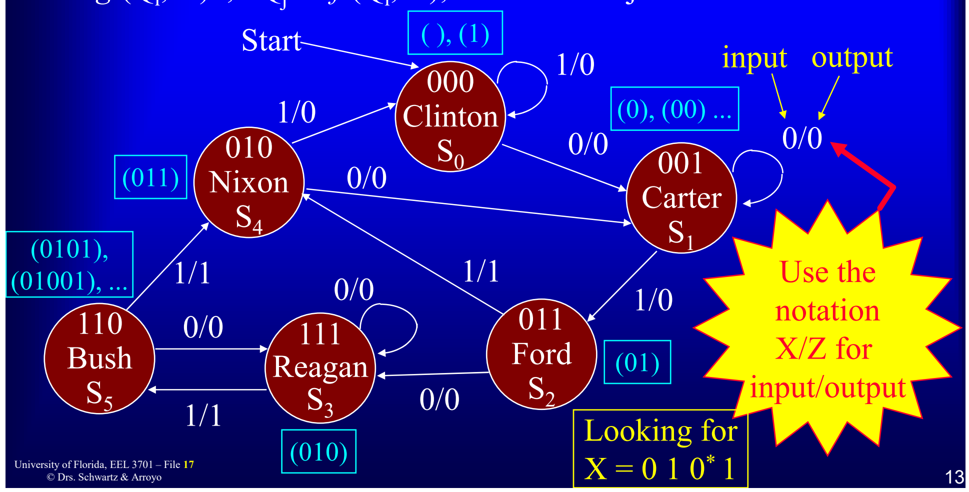
12

EEL3701

## Mealy Machine Example

**Mealy Machine**: output is a function of **state and inputs**!

- $Z = g(Q_i, X)$ , $Q_j^+ = f(Q_i, X)$, for all i and j






13

EEL3701

## Mealy Design Example

- In a Mealy circuit, ***the output Z is a function of the input & the state*** (where you are in the diagram)

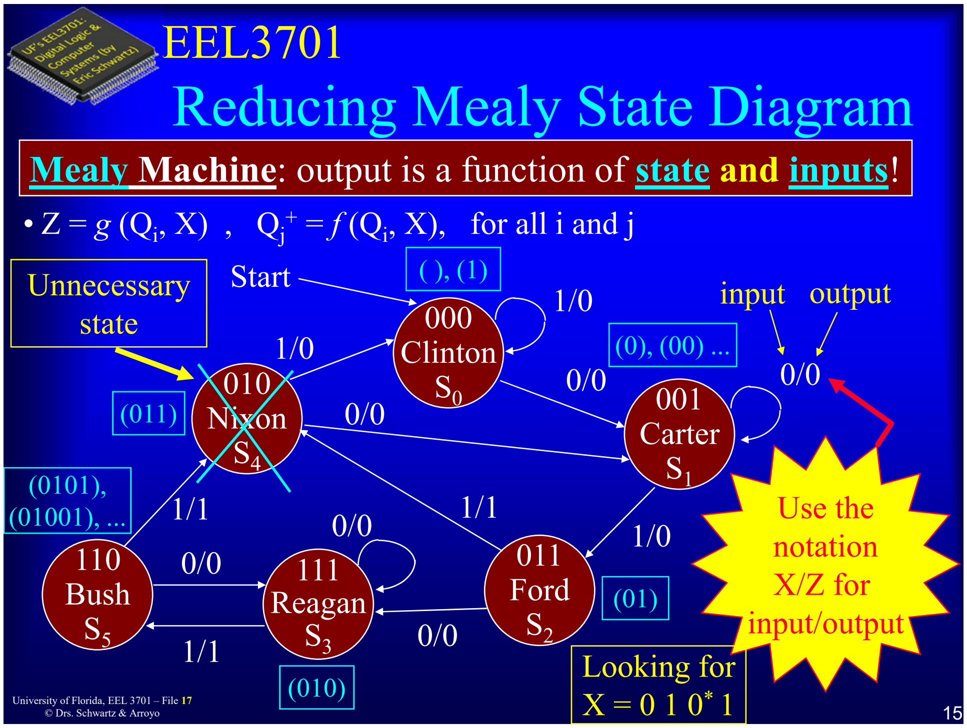
$Z = g(Q_i, X)$ , $Q_j^+ = f(Q_i, X)$, for all i and j

Looking for
$X = 0\ 1\ 0^*\ 1$

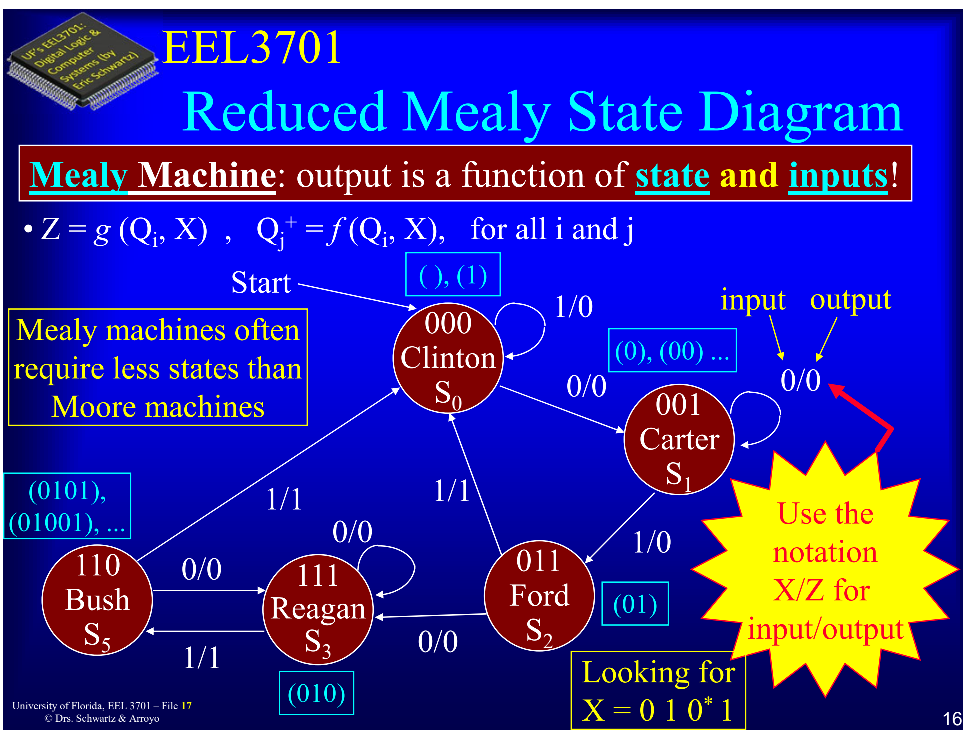
For Moore machines, $Z = g(Q_i)$, where $Q^+ = f(Q_i, X)$, for all i



14


EEL3701
Reducing Mealy State Diagram
Mealy Machine: output is a function of state and inputs!
• Z = g (Qi, X) , Qj+ = f (Qi, X), for all i and j
Unnecessary state
Start
( ), (1)
000 Clinton S0
1/0
(0), (00) ...
input output
0/0
1/0
0/0
(011)
010 Nixon S4
0/0
001 Carter S1
(0101), (01001), ...
1/1
1/1
0/0
110 Bush S5
0/0
111 Reagan S3
011 Ford S2
1/0
(01)
Use the notation X/Z for input/output
1/1
0/0
(010)
Looking for X = 0 1 0* 1
University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo
15

15


EEL3701
Reduced Mealy State Diagram
Mealy Machine: output is a function of state and inputs!
• Z = g (Qi, X) , Qj+ = f (Qi, X), for all i and j
Start
( ), (1)
Mealy machines often require less states than Moore machines
000 Clinton S0
1/0
(0), (00) ...
input output
0/0
0/0
001 Carter S1
(0101), (01001), ...
1/1
1/1
0/0
110 Bush S5
0/0
111 Reagan S3
011 Ford S2
1/0
(01)
Use the notation X/Z for input/output
1/1
0/0
(010)
Looking for X = 0 1 0* 1
University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo
16

16

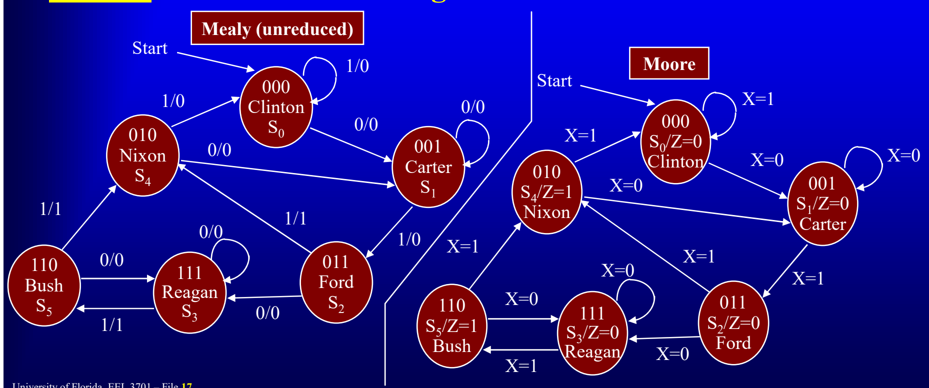
EEL3701

## Moore/Mealy Comparison

**[Example]** Design a *Sequence Detector/Acceptor* to accept $X = 010^{*}1$

where $0^{*}$ = { λ(nil), 0, 00, 000, 0000, ....... }

- **STEP 1: Draw the State Diagram.**






17

EEL3701

## Assigning State Bits

- **STEP 2: Assign state patterns or Names**

- 6 states require 3-FF's to represent $Q_2$, $Q_1$, $Q_0$.

state bits

| States | Names | Names | $Q_2$ | $Q_1$ | $Q_0$ |
|---|---|---|---|---|---|
| $S_0$ | Clinton | George | 0 | 0 | 0 |
| $S_1$ | Carter | John | 0 | 0 | 1 |
| $S_2$ | Ford | Thomas | 0 | 1 | 1 |
| $S_3$ | Reagan | Abe | 1 | 1 | 1 |
| $S_4$ | Nixon | Teddy | 0 | 1 | 0 |
| $S_5$ | Bush | Franklin | 1 | 1 | 0 |

- The above is called a state assignment table and the state numbers, state names, and state bits are **arbitrarily** assigned here



18

## EEL3701

# Use Table To Get K-Maps

- **STEPS 3 & 4: Obtain the next state K-Maps (using D-FFs).**

| States | Current $Q_2$ | Current $Q_1$ | Current $Q_0$ | Next for X=0 $Q_2^+$ | Next for X=0 $Q_1^+$ | Next for X=0 $Q_0^+$ | Next for X=1 $Q_2^+$ | Next for X=1 $Q_1^+$ | Next for X=1 $Q_0^+$ |
|---|---|---|---|---|---|---|---|---|---|
| $S_0$ | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| $S_1$ | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| $S_4$ | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| $S_2$ | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| unused | 1 | 0 | 0 | X | X | X | X | X | X |
| unused | 1 | 0 | 1 | X | X | X | X | X | X |
| $S_5$ | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| $S_3$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

K-Map for $D_2=Q_2^+$

| $Q_1Q_0$ \ $Q_2X$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | | | |
| 01 | | | | |
| 11 | | | | |
| 10 | | | | |

Need 4 variable K-Maps because the inputs are: X, $Q_2$, $Q_1$, $Q_0$ (4 variables).

If I had a longer paper, I'd use a 4-input (16-row) truth table

University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo

19

19

## EEL3701

# Moore K-Maps

| $Q_1Q_0$ \ $Q_2X$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | X | X |
| 01 | 0 | 0 | X | X |
| 11 | 1 | 0 | 1 | 1 |
| 10 | 0 | 0 | 0 | 1 |

$Q_2^+ = Q_2Q_0 + Q_2\,/X + Q_1Q_0\,/X$

| $Q_1Q_0$ \ $Q_2X$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | X | X |
| 01 | 0 | 1 | X | X |
| 11 | 1 | 1 | 1 | 1 |
| 10 | 0 | 0 | 1 | 1 |

$Q_1^+ = Q_2 + Q_1Q_0 + Q_0X$

| $Q_1Q_0$ \ $Q_2X$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 0 | X | X |
| 01 | 1 | 1 | X | X |
| 11 | 1 | 0 | 0 | 1 |
| 10 | 1 | 0 | 0 | 1 |

$Q_0^+ = /Q_1\,Q_0 + /X$

| $Q_1Q_0$ \ $Q_2$ | 0 | 1 |
|---|---|---|
| 00 | 0 | X |
| 01 | 0 | X |
| 11 | 0 | 0 |
| 10 | 1 | 1 |

$Z_{Moore} = Q_1\,/Q_0$

University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo

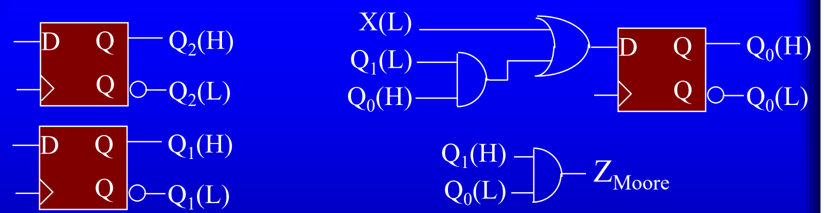
20

20

EEL3701

## Implement Design with D-FF's

- Chose D-FF so that $D_i = Q_i$. (Could have chosen T, JK, or SR FFs.)




**NOTE:** A characteristic of the Moore design is that $Z \neq f(\text{input})$, *i.e.* $Z = f(\text{state exclusively})$

- For **this** example, the **Mealy** design is identical except for the output Z. A map for Mealy Z

$Z_{Mealy} = Q_2X + Q_1Q_0X$

| $Q_1Q_0$ \ $Q_2X$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | X | X |
| 01 | 0 | 0 | X | X |
| 11 | 0 | 1 | 1 | 0 |
| 10 | 0 | 0 | 1 | 0 |

University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo

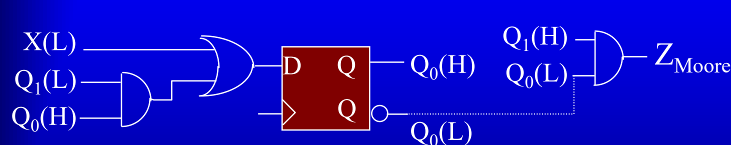
21

21

EEL3701

## Synch. Moore Machine Output

- Let us apply the same sequence X to the Moore machine. **Note** the output is generated not during the transition from state to state, but **"after"** you arrive at the new state.
- Recall, in the Moore machine, $Z = Q_1 /Q_0$



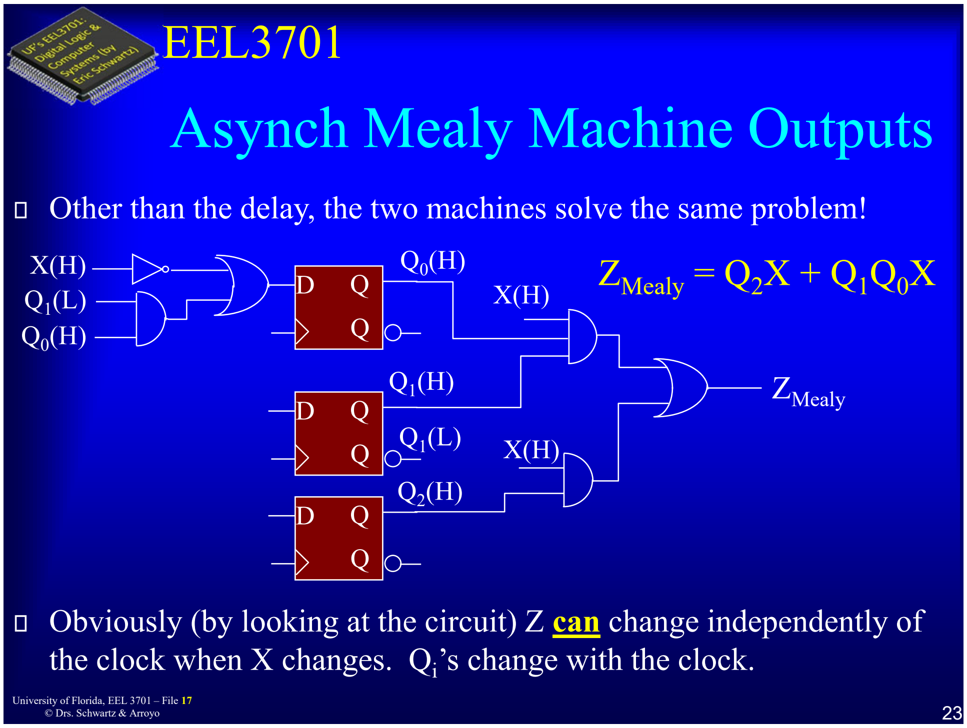

- Obviously (by looking at the circuit) Z will not change until $Q_0$ changes. $Q_0$ changes with the CLK (as do all $Q_i$'s) .
- The next X comes in, then the CLK, then $Q_0$ changes to reflect the new X, i.e., $Q_0$ (and therefore Z) lags (trails) X by up to 1 clock pulse.

This is characteristic of Moore Machines!!!

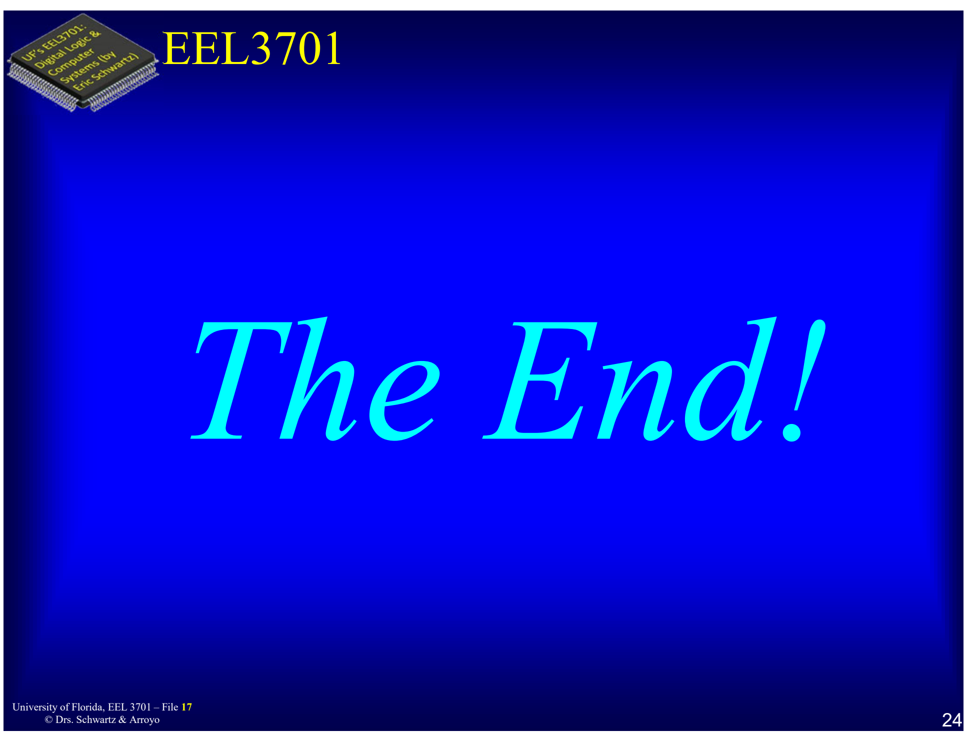
University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo

22

22


EEL3701
Asynch Mealy Machine Outputs
Other than the delay, the two machines solve the same problem!
$Z_{Mealy} = Q_2X + Q_1Q_0X$
X(H)
$Q_1$(L)
$Q_0$(H)
D Q
Q
$Q_0$(H)
X(H)
$Q_1$(H)
$Q_1$(L)
X(H)
$Q_2$(H)
$Z_{Mealy}$
Obviously (by looking at the circuit) Z **can** change independently of the clock when X changes. $Q_i$'s change with the clock.
University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo
23

23


EEL3701
The End!
University of Florida, EEL 3701 – File 17
© Drs. Schwartz & Arroyo
24

24