# CMSC 313 Lecture 24

- State Assignment Heuristics
- Using J-K Flip-Flops

# Last Time

- Mealy vs Moore finite state machines
- Vending machine example
- Sequence detector example
- State reduction algorithm

# Simplifying Finite State Machines

- State Reduction: equivalent FSM with fewer states
- State Assignment: choose an assignment of bit patterns to states (e.g., B is 010) that results in a smaller circuit
- Choice of flip-flops: use D flip-flops, J-K flip-flops or a T flip-flops? a good choice could lead to simpler circuits.

# The State Assignment Problem

- **Two state assignments for machine $M_2$.**

| P.S. \ Input | $X$ = 0 | $X$ = 1 |
|---|---|---|
| $A$ | $B$/1 | $A$/1 |
| $B$ | $C$/0 | $D$/1 |
| $C$ | $C$/0 | $D$/0 |
| $D$ | $B$/1 | $A$/0 |

Machine $M_2$

| $S_0S_1$ \ Input | $X$ = 0 | $X$ = 1 |
|---|---|---|
| $A$: 00 | 01/1 | 00/1 |
| $B$: 01 | 10/0 | 11/1 |
| $C$: 10 | 10/0 | 11/0 |
| $D$: 11 | 01/1 | 00/0 |

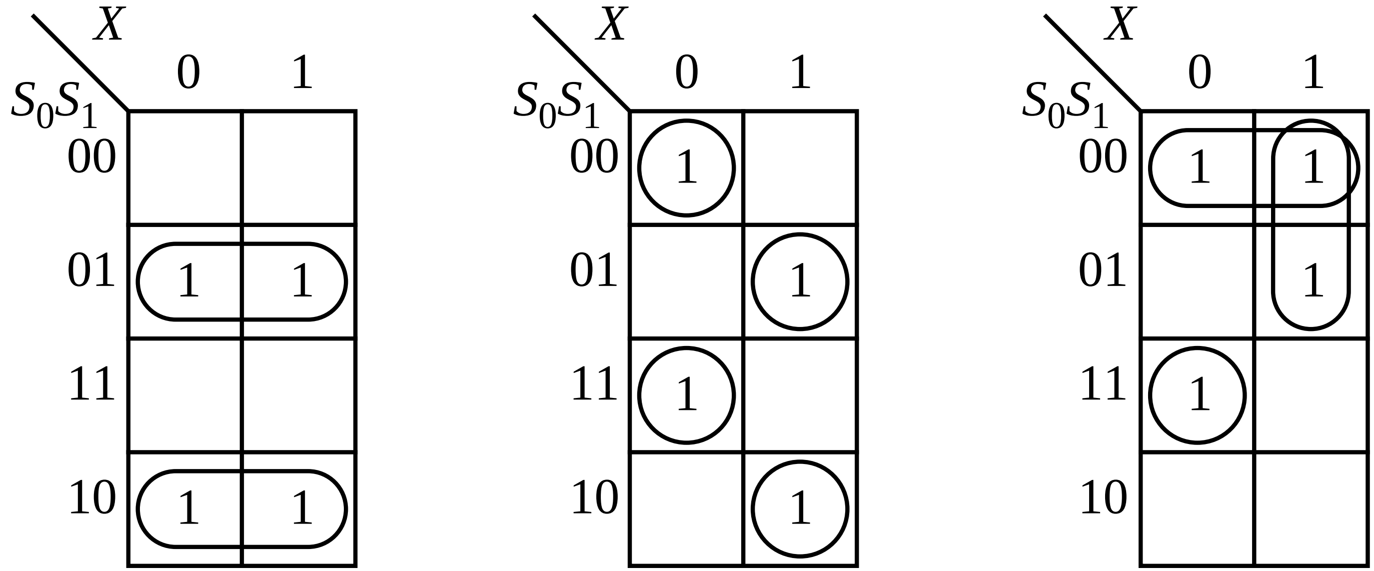
State assignment $SA_0$

| $S_0S_1$ \ Input | $X$ = 0 | $X$ = 1 |
|---|---|---|
| $A$: 00 | 01/1 | 00/1 |
| $B$: 01 | 11/0 | 10/1 |
| $C$: 11 | 11/0 | 10/0 |
| $D$: 10 | 01/1 | 00/0 |

State assignment $SA_1$

# State Assignment $SA_0$

- **Boolean equations for machine $M_2$ using state assignment $SA_0$.**



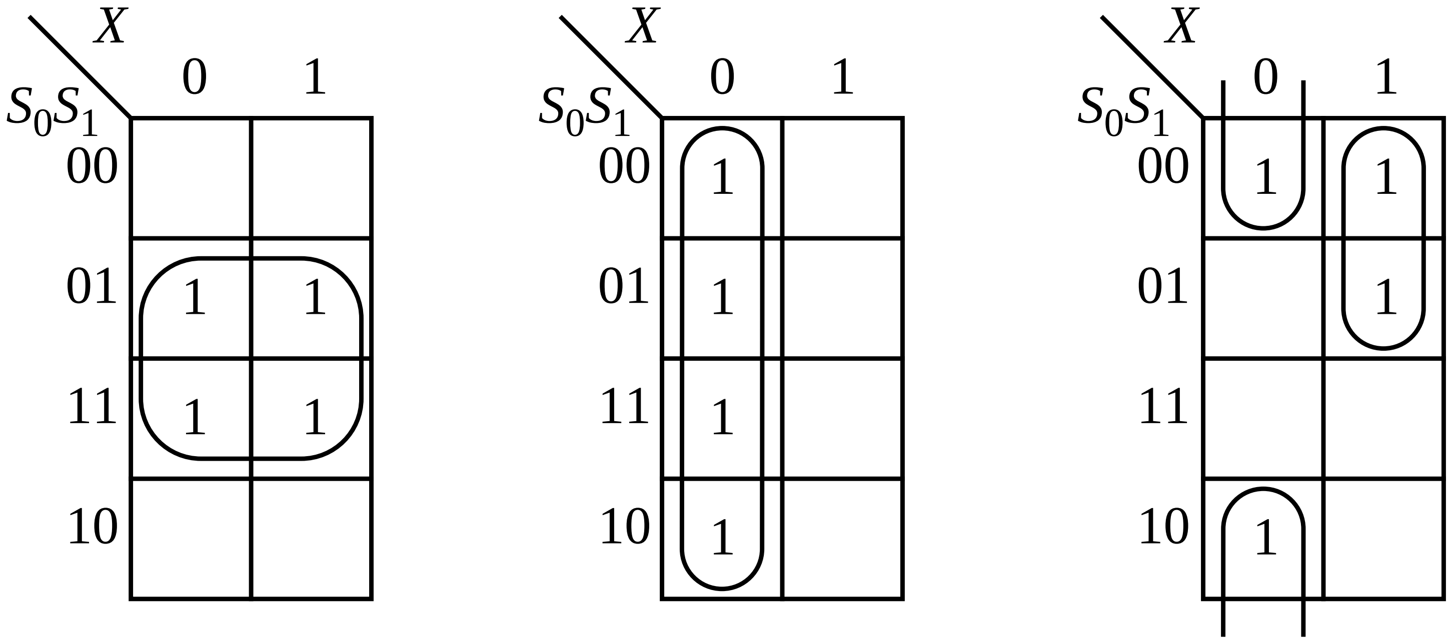

$$S_0 = \overline{S_0}S_1 + S_0\overline{S_1}$$

$$S_1 = \overline{S_0}\,\overline{S_1}\,\overline{X} + \overline{S_0}S_1X + S_0S_1\overline{X} + S_0\overline{S_1}X$$

$$Z = \overline{S_0}\,\overline{S_1} + \overline{S_0}X + S_0S_1\overline{X}$$

# State Assignment $SA_1$

- **Boolean equations for machine $M_2$ using state assignment $SA_1$.**

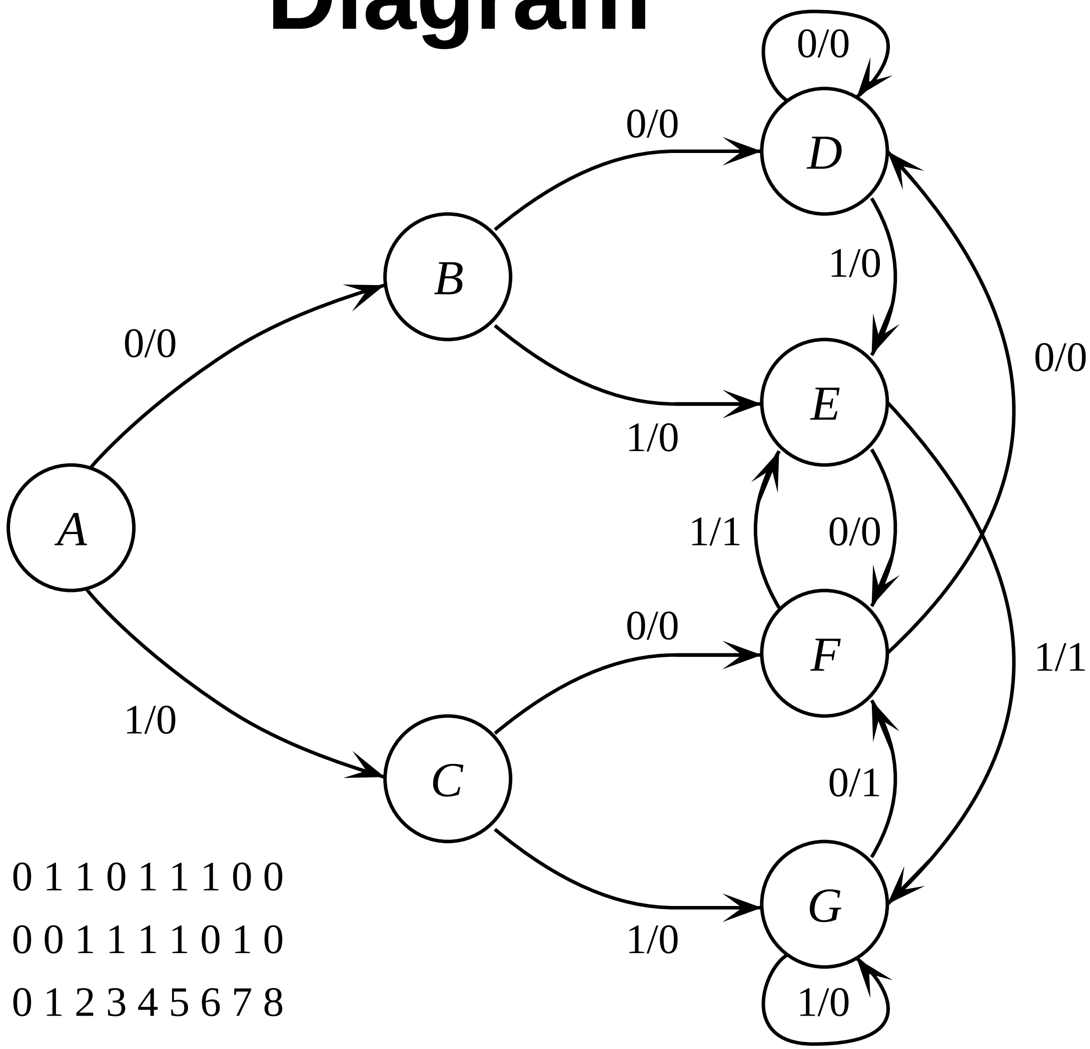



$S_0 = S_1$ $\qquad$ $S_1 = \bar{X}$ $\qquad$ $Z = \overline{S_1}\bar{X} + \overline{S_0}X$

# State Assignment Heuristics

- **No known efficient alg. for best state assignment**
- **Some heuristics (rules of thumb):**
  - The initial state should be simple to reset — all zeroes or all ones.
  - Minimize the number of state variables that change on each transition.
  - Maximize the number of state variables that don't change on each transition.
  - Exploit symmetries in the state diagram.
  - If there are unused states (when the number of states s is not a power of 2), choose the unused state variable combinations carefully. (Don't just use the first s combination of state variables.)
  - Decompose the set of state variables into bits or fields that have well-defined meaning with respect to the input or output behavior.
  - Consider using more than the minimum number of states to achieve the objectives above.

# Sequence Detector State Transition Diagram




| | |
|---|---|
| Input: | 0 1 1 0 1 1 1 0 0 |
| Output: | 0 0 1 1 1 1 0 1 0 |
| Time: | 0 1 2 3 4 5 6 7 8 |

# Sequence Detector State Table

| Present state \ Input | X: 0 | X: 1 |
|---|---|---|
| $A$ | $B/0$ | $C/0$ |
| $B$ | $D/0$ | $E/0$ |
| $C$ | $F/0$ | $G/0$ |
| $D$ | $D/0$ | $E/0$ |
| $E$ | $F/0$ | $G/1$ |
| $F$ | $D/0$ | $E/1$ |
| $G$ | $F/1$ | $G/0$ |

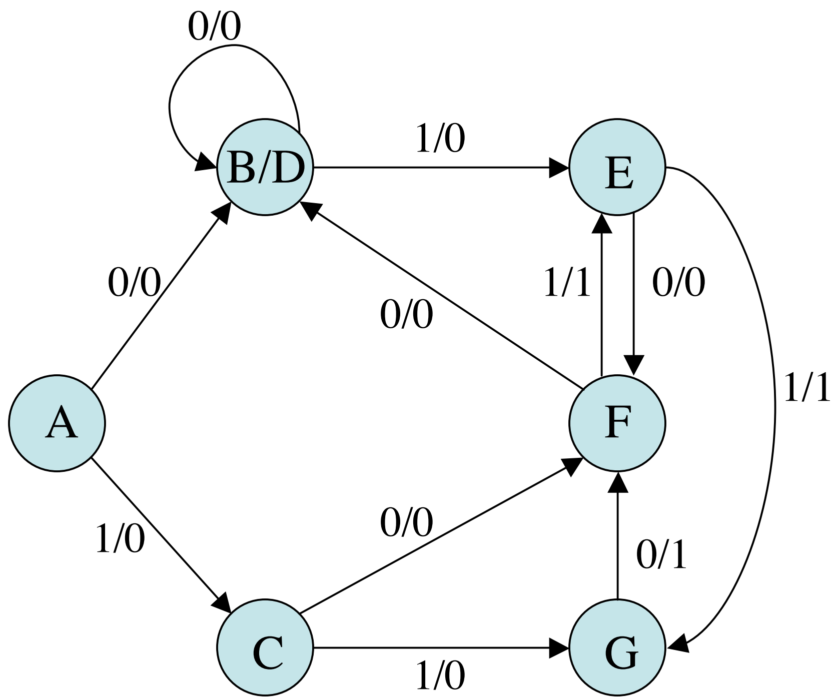
# Sequence Detector State Reduction Table

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| A | | x | x | x | x | x | x |
| B | | | x | | x | x | x |
| C | | | | x | x | x | x |
| D | | | | | x | x | x |
| E | | | | | | x | x |
| F | | | | | | | x |
| G | | | | | | | |

# Sequence Detector Reduced State Table

| Present state \ Input | X = 0 | X = 1 |
|---|---|---|
| $A: A'$ | $B'/0$ | $C'/0$ |
| $BD: B'$ | $B'/0$ | $D'/0$ |
| $C: C'$ | $E'/0$ | $F'/0$ |
| $E: D'$ | $E'/0$ | $F'/1$ |
| $F: E'$ | $B'/0$ | $D'/1$ |
| $G: F'$ | $E'/1$ | $F'/0$ |

# 6-State Sequence Detector



# Sequence Detector State Assignment

| Present state \ Input | $X$ = 0 | $X$ = 1 |
|---|---|---|
| $S_2S_1S_0$ | $S_2S_1S_0Z$ | $S_2S_1S_0Z$ |
| $A'$: 000 | 001/0 | 010/0 |
| $B'$: 001 | 001/0 | 011/0 |
| $C'$: 010 | 100/0 | 101/0 |
| $D'$: 011 | 100/0 | 101/1 |
| $E'$: 100 | 001/0 | 011/1 |
| $F'$: 101 | 100/1 | 101/0 |

# Sequence Detector K-Maps

- **K-map reduction of next state and output functions for sequence detector.**

| $S_0X$ \ $S_2S_1$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | | $d$ | 1 |
| 01 | | 1 | $d$ | 1 |
| 11 | 1 | 1 | $d$ | 1 |
| 10 | 1 | | $d$ | |

$$S_0 = \overline{S_2}\,\overline{S_1}\,\overline{X} + S_0X + S_2\overline{S_0} + S_1X$$

| $S_0X$ \ $S_2S_1$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | | $d$ | |
| 01 | 1 | | $d$ | 1 |
| 11 | 1 | | $d$ | |
| 10 | | | $d$ | |

$$S_1 = \overline{S_2}\,\overline{S_1}X + S_2\overline{S_0}X$$

| $S_0X$ \ $S_2S_1$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | 1 | $d$ | |
| 01 | | 1 | $d$ | |
| 11 | | 1 | $d$ | 1 |
| 10 | | 1 | $d$ | 1 |

$$S_2 = S_2S_0 + S_1$$

| $S_0X$ \ $S_2S_1$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | | $d$ | |
| 01 | | | $d$ | 1 |
| 11 | | 1 | $d$ | |
| 10 | | | $d$ | 1 |

$$Z = S_2\overline{S_0}X + S_1S_0X + S_2S_0\overline{X}$$

# Improved Sequence Detector?

- **Formulas from the 7-state FSM:**

$$s2' = (\overline{s0} + x)(s2 + s1 + s0)$$

$$s1' = \overline{s0}\ x + s0\ \overline{x} = s0 \text{ xor } x$$

$$s0' = \overline{x}$$

$$z = s2\ \overline{s1}\ x + s2\ s1\ \overline{x}$$

- **Formulas from the 6-state FSM:**

$$s2' = s2\ s0 + s1$$

$$s1' = \overline{s2}\ \overline{s1}\ x + s2\ \overline{s0}\ x$$

$$s0' = \overline{s2}\ \overline{s1}\ \overline{x} + s0\ x + s2\ \overline{s0} + s1\ x$$

$$z = s2\ \overline{s0}\ x + s1\ s0\ x + s2\ s0\ \overline{x}$$

# Sequence Detector State Assignment

## 7-state

| | s2 | s1 | s0 | x | s2' | s1' | s0' | z |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 5 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 6 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 7 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 9 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 10 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 11 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 12 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 13 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 14 | 1 | 1 | 1 | 0 | d | d | d | d |
| 15 | 1 | 1 | 1 | 1 | d | d | d | d |

A = 000
B = 001
C = 010
D = 011
E = 100
F = 101
G = 110

## new 6-state

| | s2 | s1 | s0 | x | s2' | s1' | s0' | z |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 5 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 6 | 0 | 1 | 1 | 0 | d | d | d | d |
| 7 | 0 | 1 | 1 | 1 | d | d | d | d |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 9 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 10 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 11 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 12 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 13 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 14 | 1 | 1 | 1 | 0 | d | d | d | d |
| 15 | 1 | 1 | 1 | 1 | d | d | d | d |

A = 000
B/D = 001
C = 010
~~D = 011~~
E = 100
F = 101
G = 110

# 6-State Sequence Detector

## 7-state

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 1 | 1 | 1 |
| 01 | 0 | 1 | 1 | 1 |
| 11 | 1 | 1 | d | 1 |
| 10 | 0 | 0 | d | 0 |

$$s2' = (\overline{s0} + x)(s2 + s1 + s0)$$

## new 6-state

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 1 | 1 | 1 |
| 01 | 0 | 1 | 1 | 1 |
| 11 | 1 | d | d | 1 |
| 10 | 0 | d | d | 0 |

$$s2' = (\overline{s0} + x)(s2 + s1 + s0)$$

# 6-State Sequence Detector

## 7-state

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | | | |
| 01 | 1 | 1 | 1 | 1 |
| 11 | | | d | |
| 10 | 1 | 1 | d | 1 |

$$s1' = \overline{s0}\ x + s0\ \overline{x}$$

## new 6-state

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | | | |
| 01 | 1 | 1 | 1 | 1 |
| 11 | | d | d | |
| 10 | 0 | d | d | 0 |

$$s1' = \overline{s0}\ x$$

# 6-State Sequence Detector

## 7-state

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | 1 | 1 |
| 01 | | | | |
| 11 | | | d | |
| 10 | 1 | 1 | d | 1 |

$$s0' = \overline{x}$$

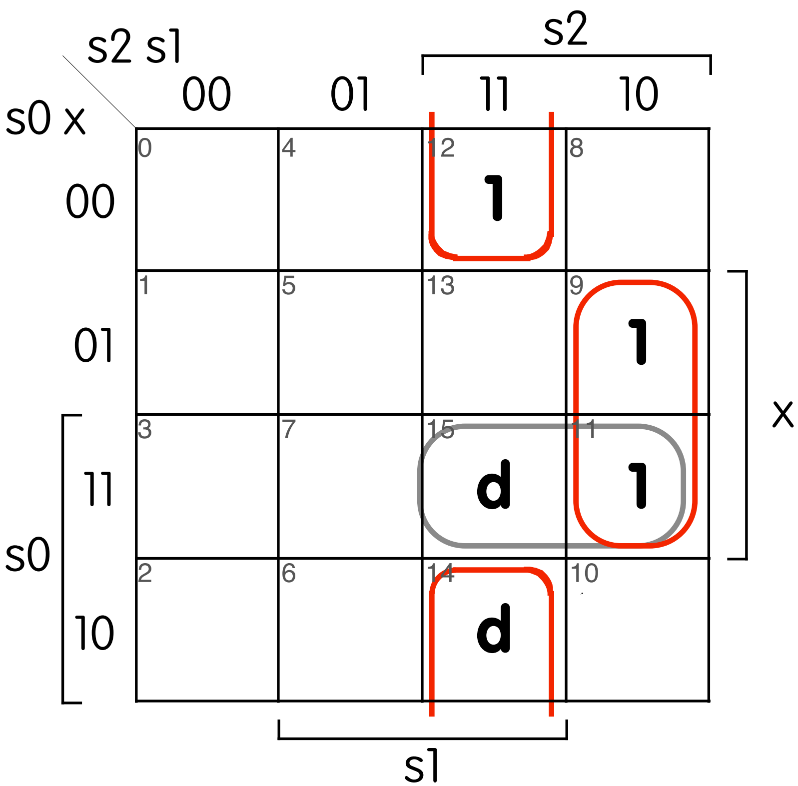
## new 6-state

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | 1 | 1 |
| 01 | | | | |
| 11 | | d | d | |
| 10 | 1 | d | d | 1 |

$$s0' = \overline{x}$$

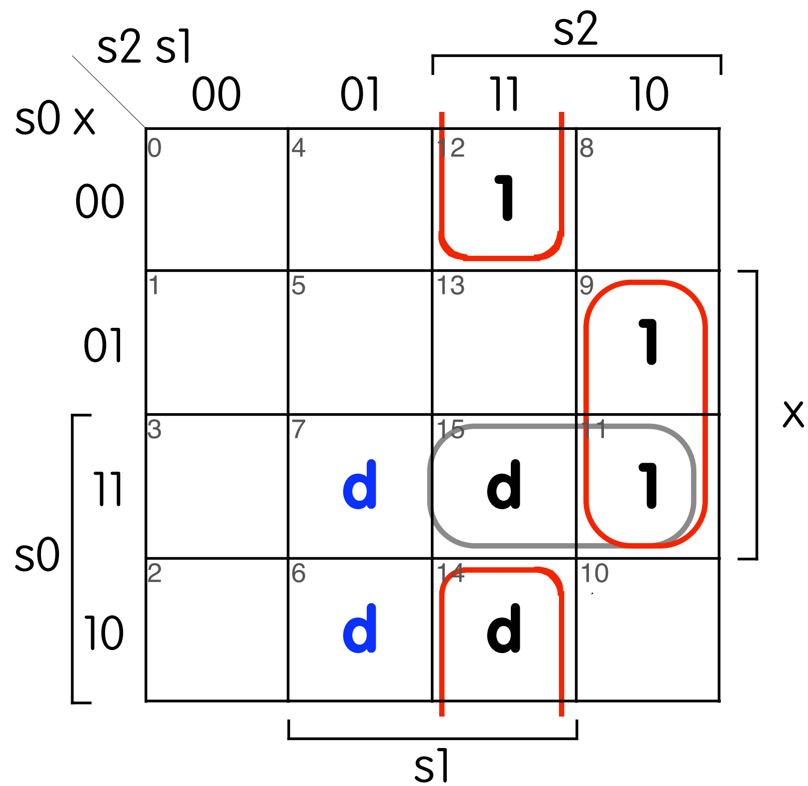
# 6-State Sequence Detector

## 7-state




$$z = s2\ \overline{s1}\ x + s2\ s1\ \overline{x}$$

## new 6-state




$$z = s2\ \overline{s1}\ x + s2\ s1\ \overline{x}$$

# Improved Sequence Detector

- **Textbook formulas for the 6-state FSM:**

$$s2' = s2\ s0 + s1$$

$$s1' = \overline{s2}\ \overline{s1}\ x + s2\ \overline{s0}\ x$$

$$s0' = \overline{s2}\ \overline{s1}\ \overline{x} + s0\ x + s2\ \overline{s0} + s1\ x$$

$$z = s2\ \overline{s0}\ x + s1\ s0\ x + s2\ s0\ \overline{x}$$

- **New formulas for the 6-state FSM:**

$$s2' = (\overline{s0} + x)(s2 + s1 + s0)$$

$$s1' = \overline{s0}\ x$$

$$s0' = \overline{x}$$

$$z = s2\ \overline{s1}\ x + s2\ s1\ \overline{x}$$

# Excitation Tables

- Each table shows the settings that must be applied at the inputs at time *t* in order to change the outputs at time *t*+1.

*S-R flip-flop*

| $Q_t$ | $Q_{t+1}$ | $S$ | $R$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 |

*D flip-flop*

| $Q_t$ | $Q_{t+1}$ | $D$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

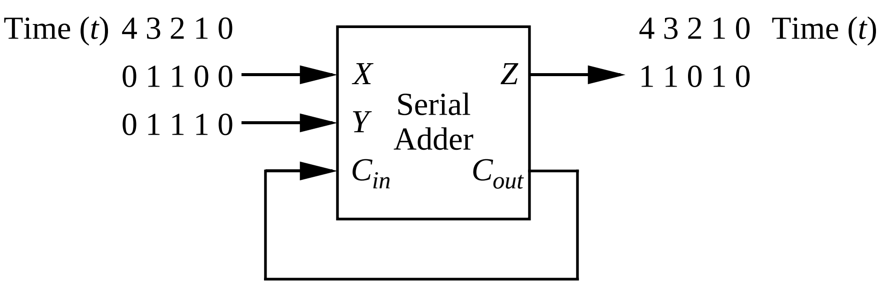
*J-K flip-flop*

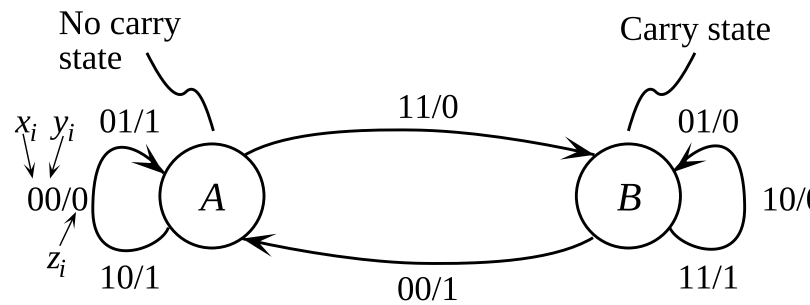
| $Q_t$ | $Q_{t+1}$ | $J$ | $K$ |
|---|---|---|---|
| 0 | 0 | 0 | $d$ |
| 0 | 1 | 1 | $d$ |
| 1 | 0 | $d$ | 1 |
| 1 | 1 | $d$ | 0 |

*T flip-flop*

| $Q_t$ | $Q_{t+1}$ | $T$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

# Serial Adder




- **State transition diagram, state table, and state assignment for a serial adder.**

| Present state \ Input | XY 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| *A* | A/0 | A/1 | A/1 | B/0 |
| *B* | A/1 | B/0 | B/0 | B/1 |

Next state Output

| Present state ($S_t$) \ Input | XY 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| *A*:0 | 0/0 | 0/1 | 0/1 | 1/0 |
| *B*:1 | 0/1 | 1/0 | 1/0 | 1/1 |

# Serial Adder Next-State Functions

- Truth table showing next-state functions for a serial adder for D, S-R, T, and J-K flip-flops. Shaded functions are used in the example.

| $X$ | $Y$ | Present State $S_t$ | $D$ | (Set) $S$ | (Reset) $R$ | $T$ | $J$ | $K$ | $Z$ |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | $d$ | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 1 | $d$ | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | $d$ | 1 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | $d$ | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | $d$ | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | $d$ | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | $d$ | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | $d$ | 0 | 1 |

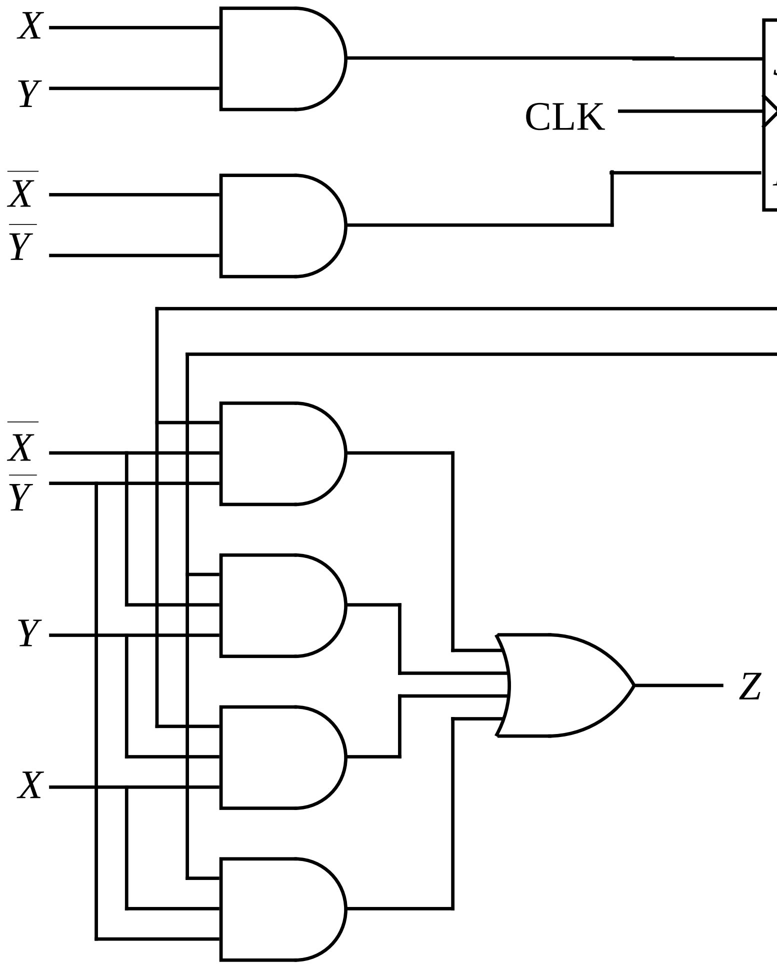
 

# J-K Flip-Flop Serial Adder Circuit

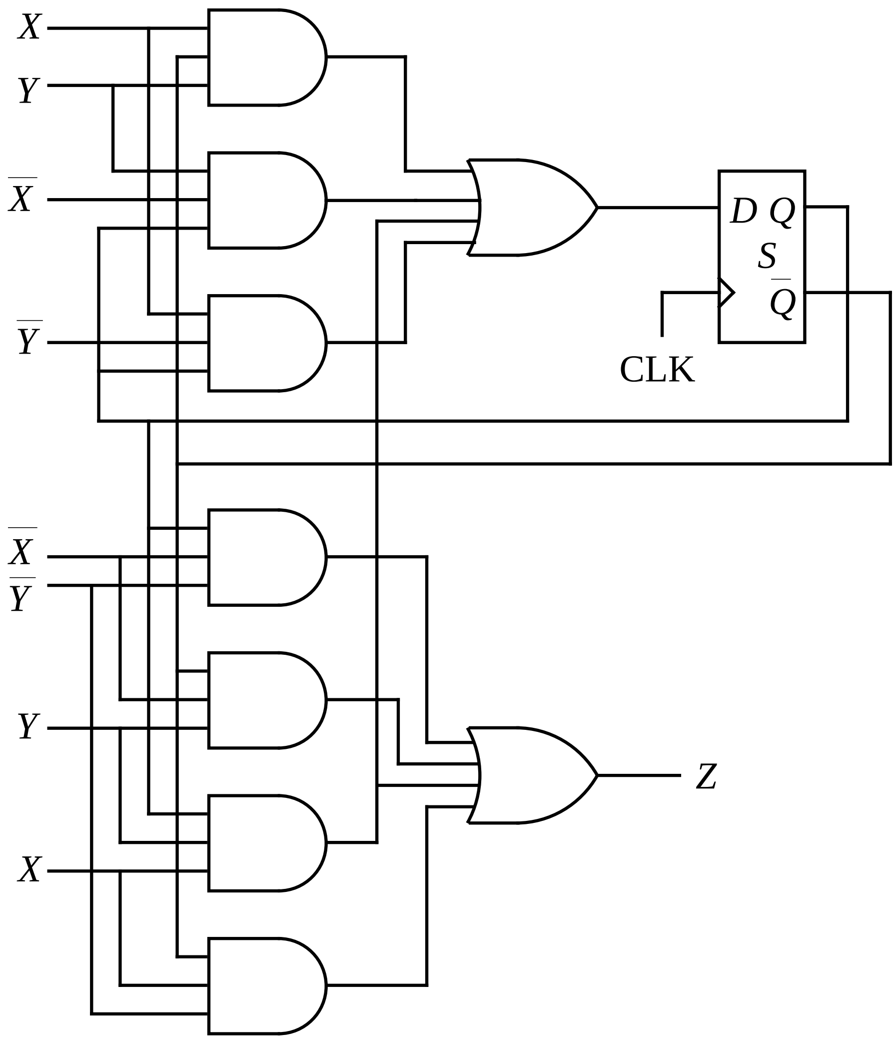
$$J = XY$$

$$K = \overline{X}\,\overline{Y}$$

$$Z = \overline{X}\,\overline{Y}S + \overline{X}Y\overline{S} + XYS + X\overline{Y}\,\overline{S}$$

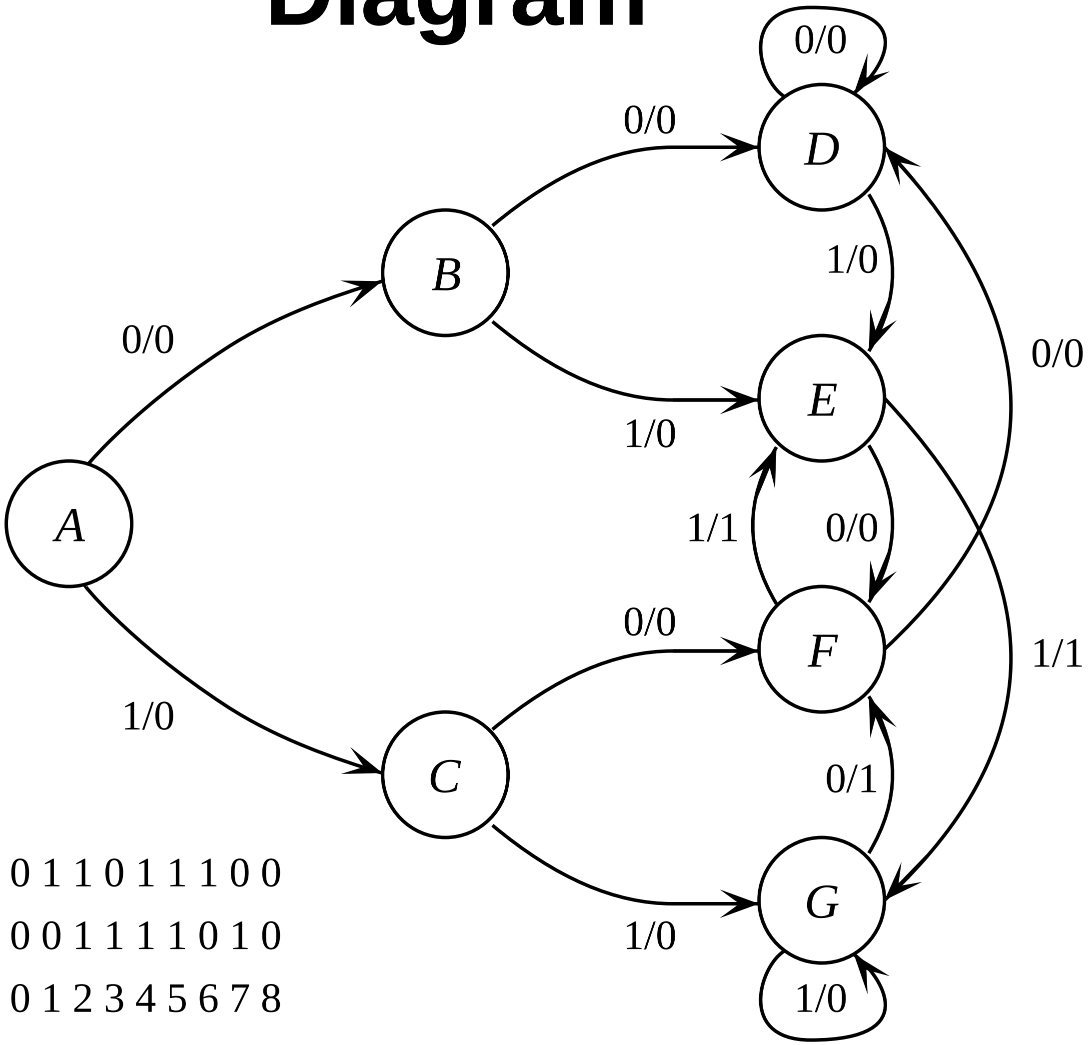


# D Flip-Flop Serial Adder Circuit



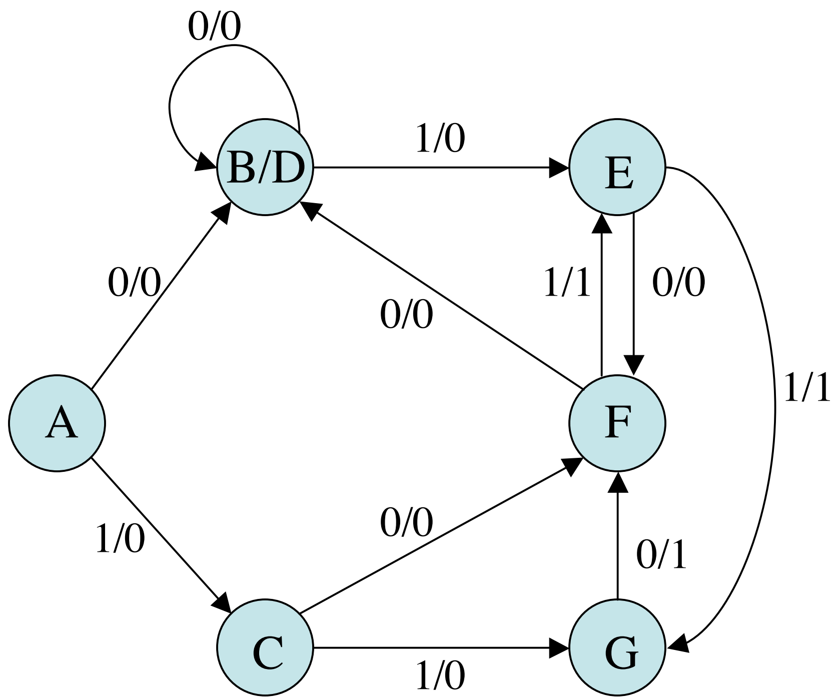

 

# Sequence Detector State Transition Diagram




| | |
|---|---|
| Input: | 0 1 1 0 1 1 1 0 0 |
| Output: | 0 0 1 1 1 1 0 1 0 |
| Time: | 0 1 2 3 4 5 6 7 8 |

# 6-State Sequence Detector



# Improved Sequence Detector?

- **Formulas from the 7-state FSM:**

$$s2' = (\overline{s0} + x)(s2 + s1 + s0)$$

$$s1' = \overline{s0}\ x + s0\ \overline{x} = s0 \text{ xor } x$$

$$s0' = \overline{x}$$

$$z = s2\ \overline{s1}\ x + s2\ s1\ \overline{x}$$

- **Formulas from the 6-state FSM:**

$$s2' = s2\ s0 + s1$$

$$s1' = \overline{s2}\ \overline{s1}\ x + s2\ \overline{s0}\ x$$

$$s0' = \overline{s2}\ \overline{s1}\ \overline{x} + s0\ x + s2\ \overline{s0} + s1\ x$$

$$z = s2\ \overline{s0}\ x + s1\ s0\ x + s2\ s0\ \overline{x}$$

# Sequence Detector State Assignment

## 7-state

| | s2 | s1 | s0 | x | s2' | s1' | s0' | z |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 5 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 6 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 7 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 9 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 10 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 11 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 12 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 13 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 14 | 1 | 1 | 1 | 0 | d | d | d | d |
| 15 | 1 | 1 | 1 | 1 | d | d | d | d |

A = 000
B = 001
C = 010
D = 011
E = 100
F = 101
G = 110

## new 6-state

| | s2 | s1 | s0 | x | s2' | s1' | s0' | z |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 5 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 6 | 0 | 1 | 1 | 0 | d | d | d | d |
| 7 | 0 | 1 | 1 | 1 | d | d | d | d |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 9 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 10 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 11 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 12 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 13 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 14 | 1 | 1 | 1 | 0 | d | d | d | d |
| 15 | 1 | 1 | 1 | 1 | d | d | d | d |

A = 000
B/D = 001
C = 010
~~D = 011~~
E = 100
F = 101
G = 110

# Improved Sequence Detector

- **Textbook formulas for the 6-state FSM:**

$$s2' = s2\ s0 + s1$$

$$s1' = \overline{s2}\ \overline{s1}\ x + s2\ \overline{s0}\ x$$

$$s0' = \overline{s2}\ \overline{s1}\ \overline{x} + s0\ x + s2\ \overline{s0} + s1\ x$$

$$z = s2\ \overline{s0}\ x + s1\ s0\ x + s2\ s0\ \overline{x}$$

- **New formulas for the 6-state FSM:**

$$s2' = (\overline{s0} + x)(s2 + s1 + s0)$$

$$s1' = \overline{s0}\ x$$

$$s0' = \overline{x}$$

$$z = s2\ \overline{s1}\ x + s2\ s1\ \overline{x}$$

# 6-State Sequence Detector

| | s2 | s1 | s0 | x | s2' | s1' | s0' | z | j2 | k2 | j1 | k1 | j0 | k0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | *d* | 0 | *d* | 1 | *d* |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | *d* | 1 | *d* | 0 | *d* |
| 2 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | *d* | 0 | *d* | *d* | 0 |
| 3 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | *d* | 0 | *d* | *d* | 1 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | *d* | *d* | 1 | 1 | *d* |
| 5 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | *d* | *d* | 0 | 0 | *d* |
| 6 | 0 | 1 | 1 | 0 | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* |
| 7 | 0 | 1 | 1 | 1 | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | *d* | 0 | 0 | *d* | 1 | *d* |
| 9 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | *d* | 0 | 1 | *d* | 0 | *d* |
| 10 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | *d* | 1 | 0 | *d* | *d* | 0 |
| 11 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | *d* | 0 | 0 | *d* | *d* | 1 |
| 12 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | *d* | 0 | *d* | 1 | 1 | *d* |
| 13 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | *d* | 0 | *d* | 0 | 0 | *d* |
| 14 | 1 | 1 | 1 | 0 | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* |
| 15 | 1 | 1 | 1 | 1 | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* | *d* |

| Q | Q' | J | K |
|---|---|---|---|
| 0 | 0 | 0 | *d* |
| 0 | 1 | 1 | *d* |
| 1 | 0 | *d* | 1 |
| 1 | 1 | *d* | 0 |

# 6-State Sequence Detector

## J2

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 1 | d | d |
| 01 | 0 | 1 | d | d |
| 11 | 1 | d | d | d |
| 10 | 0 | d | d | d |

## K2

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | d | d | 0 | 0 |
| 01 | d | d | 0 | 0 |
| 11 | d | d | d | 0 |
| 10 | d | d | d | 1 |

$J2 = s1 + s0\,x$

$K2 = s0\,\overline{x}$

# 6-State Sequence Detector

## J1

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | $_{0}$ 0 | $_{4}$ d | $_{12}$ d | $_{8}$ 0 |
| 01 | $_{1}$ 1 | $_{5}$ d | $_{13}$ d | $_{9}$ 1 |
| 11 | $_{3}$ 0 | $_{7}$ d | $_{15}$ d | $_{11}$ 0 |
| 10 | $_{2}$ 0 | $_{6}$ d | $_{14}$ d | $_{10}$ 0 |

s2, s1, s0, x

$$J1 = \overline{s0}\ x$$

## K1

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | $_{0}$ d | $_{4}$ 1 | $_{12}$ 1 | $_{8}$ d |
| 01 | $_{1}$ d | $_{5}$ 0 | $_{13}$ 0 | $_{9}$ d |
| 11 | $_{3}$ d | $_{7}$ d | $_{15}$ d | $_{11}$ d |
| 10 | $_{2}$ d | $_{6}$ d | $_{14}$ d | $_{10}$ d |

s2, s1, s0, x

$$K1 = \overline{x}$$

# 6-State Sequence Detector

## J0

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | 1 | 1 |
| 01 | 0 | 0 | 0 | 0 |
| 11 | d | d | d | d |
| 10 | d | d | d | d |

$J0 = \overline{x}$

## K0

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | d | d | d | d |
| 01 | d | d | d | d |
| 11 | 1 | d | d | 1 |
| 10 | 0 | d | d | 0 |

$K0 = x$

# Improved Sequence Detector

- **Formulas for the 6-state FSM with D Flip-flops:**

$$s2' = (\overline{s0} + x)(s2 + s1 + s0)$$

$$s1' = \overline{s0}\ x$$

$$s0' = \overline{x}$$

- **Formulas for the 6-state FSM with J-K Flip-flops:**

$$J2 = s1 + s0\ x \qquad K2 = s0\ \overline{x}$$

$$J1 = \overline{s0}\ x \qquad K1 = \overline{x}$$

$$J0 = \overline{x} \qquad K0 = x$$

**Due: Tuesday, December 7, 2003**

1. (10 points) Question A.13, page 494, Murdocca & Heuring
2. (10 points) Question A.29, page 497, Murdocca & Heuring
3. (10 points) Question B.10, page 542, Murdocca & Heuring
4. (10 points) Question B.11, page 542, Murdocca & Heuring
5. (60 points) This problem asks you to take the steps involved in the design process of a finite state machine. You will design a finite state machine that has a one bit input $x$ and a one bit output $z$. The machine must output 1 for every input sequence ending in the string 0010 or 100. The output should be 0 in all other cases.

   [Adapted from *Contemporary Logic Design*, Randy H. Katz, Benjamin/Cummings Publishing, 1994.]

   (a) (10 points) In the space provided on the next page, draw the minimum state-transition diagram for the finite state machine described above. You must use the state-minimization algorithm described in class to show that the finite state machine has the minimum number of states. (*Hint:* You should have fewer than 8 states in your machine.)

   (b) (5 points) Use the state assignment heuristics described in class and pick *two* different state assignments for your finite state machine. *Note:* the bit pattern for the initial state must be 000.

   (c) (40 points) For each of the two state assignments:

      i. Fill in the truth tables with values for D flip-flops, for the output bit and for J-K flip-flops.

      ii. Use the Karnaugh maps provided to minimize the formulas for each column of the truth table.

      iii. Count the number of gates needed for each implementation.

   (d) (5 points) Should you use your first or second state assignment? D flip-flops or J-K flip-flops?

   *Note:* Keep a copy of your work for the last question. You will need it for DigSim Assignment 3.

**CMSC 313 Homework 5** **Name: ____________________________________**

**Minimized State Transition Diagram (show work)**

**State Assignment:**

| | Assignment #1 | Assignment#2 |
|---|---|---|
| A | 000 | 000 |
| B | | |
| C | | |
| D | | |
| E | | |
| F | | |
| unused | | |
| unused | | |

# ASSIGNMENT #1

### Excitation Table for J-K Flip-Flops

| Q | Q' | J | K |
|---|---|---|---|
| 0 | 0 | 0 | *d* |
| 0 | 1 | 1 | *d* |
| 1 | 0 | *d* | 1 |
| 1 | 1 | *d* | 0 |

### Truth Table:

| | s2 | s1 | s0 | x | s2' | s1' | s0' | z | j2 | k2 | j1 | k1 | j0 | k0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | | | | | | | | | | |
| 1 | 0 | 0 | 0 | 1 | | | | | | | | | | |
| 2 | 0 | 0 | 1 | 0 | | | | | | | | | | |
| 3 | 0 | 0 | 1 | 1 | | | | | | | | | | |
| 4 | 0 | 1 | 0 | 0 | | | | | | | | | | |
| 5 | 0 | 1 | 0 | 1 | | | | | | | | | | |
| 6 | 0 | 1 | 1 | 0 | | | | | | | | | | |
| 7 | 0 | 1 | 1 | 1 | | | | | | | | | | |
| 8 | 1 | 0 | 0 | 0 | | | | | | | | | | |
| 9 | 1 | 0 | 0 | 1 | | | | | | | | | | |
| 10 | 1 | 0 | 1 | 0 | | | | | | | | | | |
| 11 | 1 | 0 | 1 | 1 | | | | | | | | | | |
| 12 | 1 | 1 | 0 | 0 | | | | | | | | | | |
| 13 | 1 | 1 | 0 | 1 | | | | | | | | | | |
| 14 | 1 | 1 | 1 | 0 | | | | | | | | | | |
| 15 | 1 | 1 | 1 | 1 | | | | | | | | | | |

## Assignment #1: Karnaugh Maps for D Flip-Flops and the output

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s2' =

# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s1' =

# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s0' =

# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

z =

# of gates =

Total # of gates for D flip-flops (don't count z) =

## Assignment #1: Karnaugh Maps for J-K Flip-Flops

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

j2 =

# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

k2 =

# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

j1 =

# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

k1 =

# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

j0 =

# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

k0 =

# of gates =

Total # of gates for J-K flip-flops (don't count z) =

# ASSIGNMENT #2

### Excitation Table for J-K Flip-Flops

| Q | Q' | J | K |
|---|---|---|---|
| 0 | 0 | 0 | *d* |
| 0 | 1 | 1 | *d* |
| 1 | 0 | *d* | 1 |
| 1 | 1 | *d* | 0 |

### Truth Table:

| | s2 | s1 | s0 | x | s2' | s1' | s0' | z | j2 | k2 | j1 | k1 | j0 | k0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | | | | | | | | | | |
| 1 | 0 | 0 | 0 | 1 | | | | | | | | | | |
| 2 | 0 | 0 | 1 | 0 | | | | | | | | | | |
| 3 | 0 | 0 | 1 | 1 | | | | | | | | | | |
| 4 | 0 | 1 | 0 | 0 | | | | | | | | | | |
| 5 | 0 | 1 | 0 | 1 | | | | | | | | | | |
| 6 | 0 | 1 | 1 | 0 | | | | | | | | | | |
| 7 | 0 | 1 | 1 | 1 | | | | | | | | | | |
| 8 | 1 | 0 | 0 | 0 | | | | | | | | | | |
| 9 | 1 | 0 | 0 | 1 | | | | | | | | | | |
| 10 | 1 | 0 | 1 | 0 | | | | | | | | | | |
| 11 | 1 | 0 | 1 | 1 | | | | | | | | | | |
| 12 | 1 | 1 | 0 | 0 | | | | | | | | | | |
| 13 | 1 | 1 | 0 | 1 | | | | | | | | | | |
| 14 | 1 | 1 | 1 | 0 | | | | | | | | | | |
| 15 | 1 | 1 | 1 | 1 | | | | | | | | | | |

# Assignment #2: Karnaugh Maps for D Flip-Flops and the output

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s2' =

\# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s1' =

\# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s0' =

\# of gates =

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

z =

\# of gates =

Total # of gates for D flip-flops (don't count z) =

## Assignment #2: Karnaugh Maps for J-K Flip-Flops

s2

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s1 s0 x

j2 =

# of gates =

s2

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s1 s0 x

k2 =

# of gates =

s2

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s1 s0 x

j1 =

# of gates =

s2

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s1 s0 x

k1 =

# of gates =

s2

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s1 s0 x

j0 =

# of gates =

s2

| s0 x \ s2 s1 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 4 | 12 | 8 |
| 01 | 1 | 5 | 13 | 9 |
| 11 | 3 | 7 | 15 | 11 |
| 10 | 2 | 6 | 14 | 10 |

s1 s0 x

k0 =

# of gates =

Total # of gates for J-K flip-flops (don't count z) =