



# DIGITAL DESIGN AND COMPUTER ORGANIZATION

## Finite State Machines - 3

---

**Reetinder Sidhu**

Department of Computer Science and Engineering

# DIGITAL DESIGN AND COMPUTER ORGANIZATION

---

## Finite State Machines - 3

**Reetinder Sidhu**

Department of Computer Science and  
Engineering

- Digital Design
  - ▶ Combinational logic design
  - ▶ Sequential logic design
    - ★ **Finite State Machines - 3**
- Computer Organization
  - ▶ Architecture (microprocessor instruction set)
  - ▶ Microarchitecture (microprocessor operation)

### Concepts covered

- Finite State Machine Design Example (Mealy)

# FINITE STATE MACHINES - 3

## How to Design Synchronous Sequential Logic Circuits? (Mealy FSM)

- Determine inputs and outputs
- State transition diagram
- Encoding tables
  - ▶ State
  - ▶ Output
- State transition and output table
- Logic minimization of state transition and output table yields Boolean formulas for
  - ▶ next state logic
  - ▶ output logic
- Logic circuit construction

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

---



Source: *platformliftco*

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first



Source: *platformliftco*

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up



Source: *platformliftco*

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up
  - ▶ The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up



Source: *platformliftco*



# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up
  - ▶ The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The lift system has a sensor which indicates when the lift is stationary at one of the floors or is moving between them



Source: platformliftco

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up
  - ▶ The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The lift system has a sensor which indicates when the lift is stationary at one of the floors or is moving between them
  - ▶ The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors



Source: *platformliftco*

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up
  - ▶ The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The lift system has a sensor which indicates when the lift is stationary at one of the floors or is moving between them
  - ▶ The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Control logic should output signals to control the motor that takes the lift up and down



Source: *platformliftco*

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up
  - ▶ The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The lift system has a sensor which indicates when the lift is stationary at one of the floors or is moving between them
  - ▶ The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Control logic should output signals to control the motor that takes the lift up and down
  - ▶ Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
  - ▶ Signal *lift\_down* is 1 when the lift should move down and 0 otherwise



Source: platformliftco

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up
  - ▶ The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The lift system has a sensor which indicates when the lift is stationary at one of the floors or is moving between them
  - ▶ The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Control logic should output signals to control the motor that takes the lift up and down
  - ▶ Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
  - ▶ Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- Control logic should also output signals indicating which floor the lift is on



Source: platformliftco

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up
  - ▶ The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The lift system has a sensor which indicates when the lift is stationary at one of the floors or is moving between them
  - ▶ The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Control logic should output signals to control the motor that takes the lift up and down
  - ▶ Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
  - ▶ Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- Control logic should also output signals indicating which floor the lift is on
  - ▶ *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
  - ▶ *on\_first* is 1 when lift is on first floor and 0 otherwise



Source: platformliftco

# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up
  - ▶ The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The lift system has a sensor which indicates when the lift is stationary at one of the floors or is moving between them
  - ▶ The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Control logic should output signals to control the motor that takes the lift up and down
  - ▶ Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
  - ▶ Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- Control logic should also output signals indicating which floor the lift is on
  - ▶ *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
  - ▶ *on\_first* is 1 when lift is on first floor and 0 otherwise
- Elevator initially on ground floor



Source: platformliftco

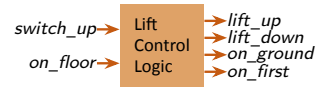
# FINITE STATE MACHINES - 3

## Lift (Elevator) Control Logic

- Problem is to design the control logic for a lift in a building of two floors: ground and first
- The lift has a switch with two positions, down and up, which respectively indicate that the lift should go down or up
  - ▶ The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The lift system has a sensor which indicates when the lift is stationary at one of the floors or is moving between them
  - ▶ The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Control logic should output signals to control the motor that takes the lift up and down
  - ▶ Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
  - ▶ Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- Control logic should also output signals indicating which floor the lift is on
  - ▶ *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
  - ▶ *on\_first* is 1 when lift is on first floor and 0 otherwise
- Elevator initially on ground floor



Source: platformliftco





# FINITE STATE MACHINES - 3

## State Transition Diagram



### State Transition Diagram

- A visual representation of an FSM
  - ▶ States represent by circles (called **nodes/vertices**)
  - ▶ Transitions between states represented by directed line segments (called **arcs/edges/arrows**)
  - ▶ Each edge is labeled with a Boolean formula of inputs as well as outputs corresponding to the transition
- In each clock cycle, only one state is **active**
- Active state transitions occur at the rising edge of the clock signal
- If a state is active in the current clock cycle, and it has an outgoing transition labeled with a Boolean formula which is 1 in that clock cycle, then the destination state of the transition will be the active state in the next clock cycle

# FINITE STATE MACHINES - 3

## State Transition Diagram

- The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

### Elevator Example State Transition Diagram

# FINITE STATE MACHINES - 3

## State Transition Diagram

- The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

### Elevator Example State Transition Diagram

- **State f0** Lift on ground floor or in transit to ground floor
- **State f1** Lift on first floor or in transit to first floor

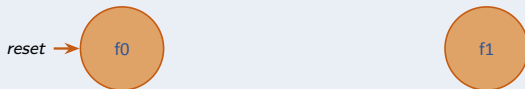
# FINITE STATE MACHINES - 3

## State Transition Diagram

- The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

### Elevator Example State Transition Diagram

- **State f0** Lift on ground floor or in transit to ground floor
- **State f1** Lift on first floor or in transit to first floor



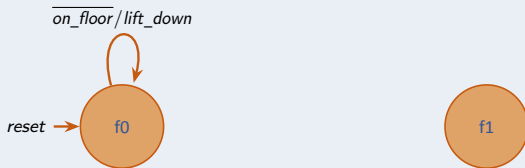
# FINITE STATE MACHINES - 3

## State Transition Diagram

- The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

### Elevator Example State Transition Diagram

- **State f0** Lift on ground floor or in transit to ground floor
- **State f1** Lift on first floor or in transit to first floor



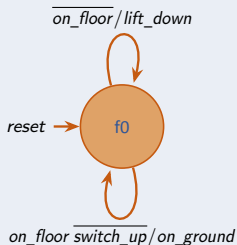
# FINITE STATE MACHINES - 3

## State Transition Diagram

- The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

### Elevator Example State Transition Diagram

- **State f0** Lift on ground floor or in transit to ground floor
- **State f1** Lift on first floor or in transit to first floor



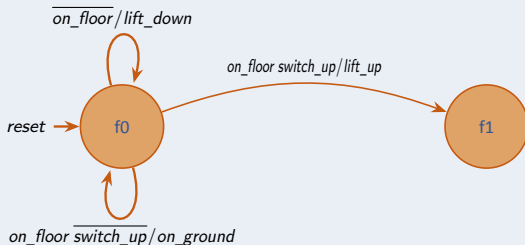
# FINITE STATE MACHINES - 3

## State Transition Diagram

- The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

### Elevator Example State Transition Diagram

- **State f0** Lift on ground floor or in transit to ground floor
- **State f1** Lift on first floor or in transit to first floor



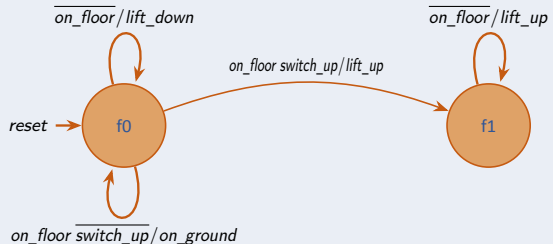
# FINITE STATE MACHINES - 3

## State Transition Diagram

- The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

### Elevator Example State Transition Diagram

- **State f0** Lift on ground floor or in transit to ground floor
- **State f1** Lift on first floor or in transit to first floor





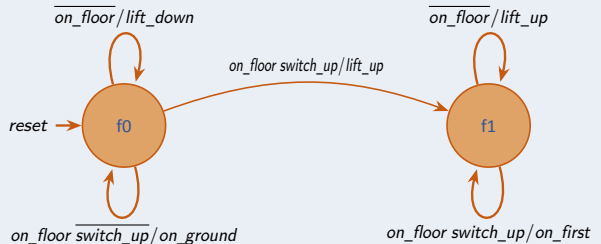
# FINITE STATE MACHINES - 3

## State Transition Diagram

- The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

### Elevator Example State Transition Diagram

- **State f0** Lift on ground floor or in transit to ground floor
- **State f1** Lift on first floor or in transit to first floor



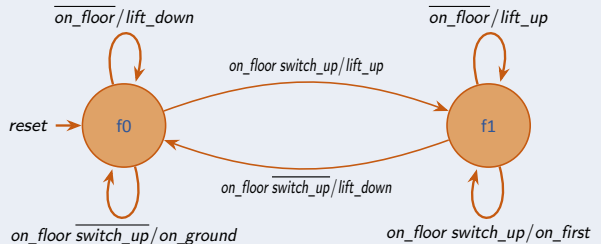
# FINITE STATE MACHINES - 3

## State Transition Diagram

- The switch signal *switch\_up* is 0 when the switch indicates the lift should go down and 1 when switch indicates it should go up
- The sensor signal *on\_floor* is 1 when the lift is at ground or first floor, and 0 when the lift is moving between floors
- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

### Elevator Example State Transition Diagram

- **State f0** Lift on ground floor or in transit to ground floor
- **State f1** Lift on first floor or in transit to first floor



# FINITE STATE MACHINES - 3

## State and Output Encoding Tables

---

- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

# FINITE STATE MACHINES - 3

## State and Output Encoding Tables

- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

Elevator State Encoding Table

State	Encoding (s)
f0	0
f1	1

# FINITE STATE MACHINES - 3

## State and Output Encoding Tables

- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

Elevator State Encoding Table

State	Encoding (s)
f0	0
f1	1

Elevator Output Encoding Tables

# FINITE STATE MACHINES - 3

## State and Output Encoding Tables

- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

Elevator State Encoding Table

State	Encoding (s)
f0	0
f1	1

Elevator Output Encoding Tables

- *on\_ground*

Meaning	Encoding
Lift on ground floor	1
Lift anywhere else	0

# FINITE STATE MACHINES - 3

## State and Output Encoding Tables

- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

Elevator State Encoding Table

State	Encoding (s)
f0	0
f1	1

Elevator Output Encoding Tables

- *on\_ground*

Meaning	Encoding
Lift on ground floor	1
Lift anywhere else	0

- *on\_first*

Meaning	Encoding
Lift on first floor	1
Lift anywhere else	0

# FINITE STATE MACHINES - 3

## State and Output Encoding Tables

- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

Elevator State Encoding Table

State	Encoding (s)
f0	0
f1	1

Elevator Output Encoding Tables

- *on\_ground*

Meaning	Encoding
Lift on ground floor	1
Lift anywhere else	0

- *on\_first*

Meaning	Encoding
Lift on first floor	1
Lift anywhere else	0

- *lift\_up*

Meaning	Encoding
Lift going from ground to first floor	1
Lift anywhere else	0



# FINITE STATE MACHINES - 3

## State and Output Encoding Tables

- Signal *lift\_up* is 1 when the lift should move up and 0 otherwise
- Signal *lift\_down* is 1 when the lift should move down and 0 otherwise
- *on\_ground* is 1 when the lift is on ground floor and 0 otherwise
- *on\_first* is 1 when lift is on first floor and 0 otherwise

Elevator State Encoding Table

State	Encoding (s)
f0	0
f1	1

Elevator Output Encoding Tables

- *on\_ground*

Meaning	Encoding
Lift on ground floor	1
Lift anywhere else	0

- *on\_first*

Meaning	Encoding
Lift on first floor	1
Lift anywhere else	0

- *lift\_up*

Meaning	Encoding
Lift going from ground to first floor	1
Lift anywhere else	0

- *lift\_down*

Meaning	Encoding
Lift going from first to ground floor	1
Lift anywhere else	0

# FINITE STATE MACHINES - 3

## State Transition and Output Table

Elevator Example State Transition and Output Table

Current State	Inputs		Next State
$s$	$on\_floor$	$switch\_up$	$s'$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

# FINITE STATE MACHINES - 3

## State Transition and Output Table

Elevator Example State Transition and Output Table

Current State	Inputs		Outputs			
<i>s</i>	<i>on_floor</i>	<i>switch_up</i>	<i>on_ground</i>	<i>on_first</i>	<i>lift_up</i>	<i>lift_down</i>
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

# FINITE STATE MACHINES - 3

## State Transition and Output Table



Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
<i>s</i>	<i>on_floor</i>	<i>switch_up</i>	<i>s'</i>	<i>on_ground</i>	<i>on_first</i>	<i>lift_up</i>	<i>lift_down</i>
0	0	0					
0	0	1					
0	1	0					
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					

# FINITE STATE MACHINES - 3

## State Transition and Output Table



Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$on\_floor$	$switch\_up$	$s'$	$on\_ground$	$on\_first$	$lift\_up$	$lift\_down$
0	0	0					
0	0	1					
0	1	0					
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					

# FINITE STATE MACHINES - 3

## State Transition and Output Table

$\overline{\text{on\_floor}} / \text{lift\_down}$

reset



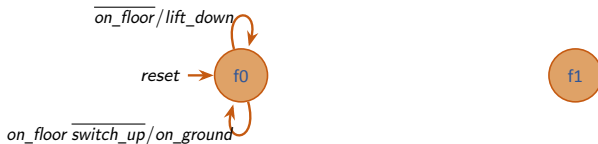
f1

### Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$\text{on\_floor}$	$\text{switch\_up}$	$s'$	$\text{on\_ground}$	$\text{on\_first}$	$\text{lift\_up}$	$\text{lift\_down}$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0					
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					

# FINITE STATE MACHINES - 3

## State Transition and Output Table

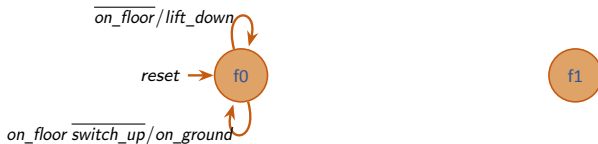


Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$on\_floor$	$switch\_up$	$s'$	$on\_ground$	$on\_first$	$lift\_up$	$lift\_down$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0					
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					

# FINITE STATE MACHINES - 3

## State Transition and Output Table



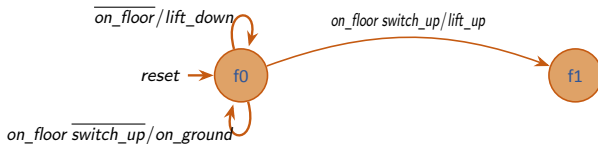
Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$on\_floor$	$switch\_up$	$s'$	$on\_ground$	$on\_first$	$lift\_up$	$lift\_down$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					



# FINITE STATE MACHINES - 3

## State Transition and Output Table

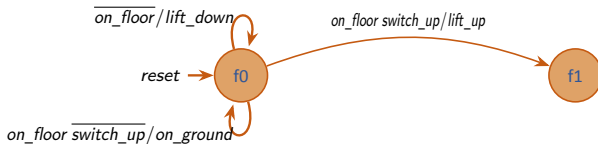


Elevator Example State Tansition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$\text{on\_floor}$	$\text{switch\_up}$	$s'$	$\text{on\_ground}$	$\text{on\_first}$	$\text{lift\_up}$	$\text{lift\_down}$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1					
1	0	0					
1	0	1					
1	1	0					
1	1	1					

# FINITE STATE MACHINES - 3

## State Transition and Output Table

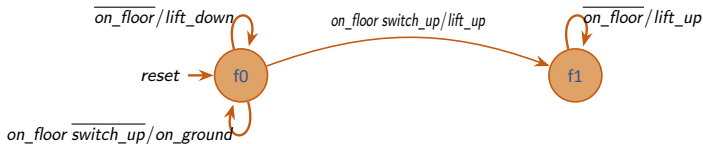


Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$on\_floor$	$switch\_up$	$s'$	$on\_ground$	$on\_first$	$lift\_up$	$lift\_down$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0					
1	0	1					
1	1	0					
1	1	1					

# FINITE STATE MACHINES - 3

## State Transition and Output Table

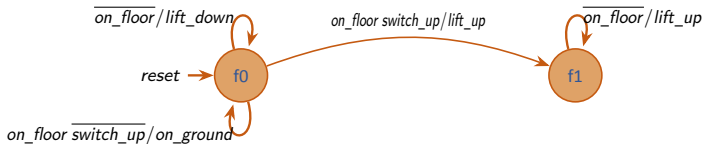


Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$\text{on\_floor}$	$\text{switch\_up}$	$s'$	$\text{on\_ground}$	$\text{on\_first}$	$\text{lift\_up}$	$\text{lift\_down}$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0					
1	0	1					
1	1	0					
1	1	1					

# FINITE STATE MACHINES - 3

## State Transition and Output Table

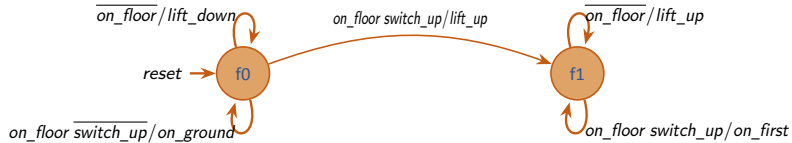


Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$\text{on\_floor}$	$\text{switch\_up}$	$s'$	$\text{on\_ground}$	$\text{on\_first}$	$\text{lift\_up}$	$\text{lift\_down}$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0					
1	1	1					

# FINITE STATE MACHINES - 3

## State Transition and Output Table

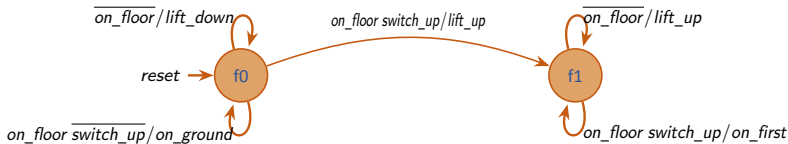


Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$\text{on\_floor}$	$\text{switch\_up}$	$s'$	$\text{on\_ground}$	$\text{on\_first}$	$\text{lift\_up}$	$\text{lift\_down}$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0					
1	1	1					

# FINITE STATE MACHINES - 3

## State Transition and Output Table

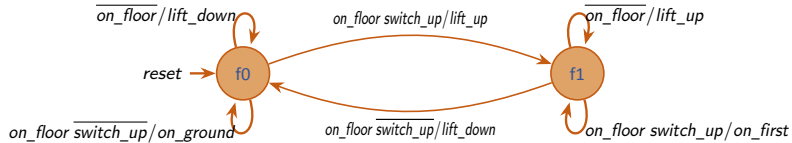


### Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$\text{on\_floor}$	$\text{switch\_up}$	$s'$	$\text{on\_ground}$	$\text{on\_first}$	$\text{lift\_up}$	$\text{lift\_down}$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0					
1	1	1	1	0	1	0	0

# FINITE STATE MACHINES - 3

## State Transition and Output Table

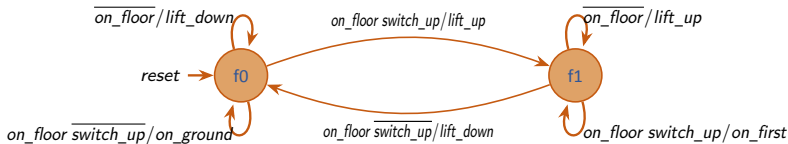


Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$\text{on\_floor}$	$\text{switch\_up}$	$s'$	$\text{on\_ground}$	$\text{on\_first}$	$\text{lift\_up}$	$\text{lift\_down}$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0					
1	1	1	1	0	1	0	0

# FINITE STATE MACHINES - 3

## State Transition and Output Table



### Elevator Example State Transition and Output Table

Current State	Inputs		Next State	Outputs			
$s$	$\text{on\_floor}$	$\text{switch\_up}$	$s'$	$\text{on\_ground}$	$\text{on\_first}$	$\text{lift\_up}$	$\text{lift\_down}$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	1
1	1	1	1	0	1	0	0



# FINITE STATE MACHINES - 3

## Logic Minimization

- State transition table:

Current State	Inputs		Next State	Outputs			
$s$	$on\_floor$	$switch\_up$	$s'$	$on\_ground$	$on\_first$	$lift\_up$	$lift\_down$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	1
1	1	1	1	0	1	0	0

Minimized Boolean Formula

# FINITE STATE MACHINES - 3

## Logic Minimization

- State transition table:

Current State	Inputs		Next State	Outputs			
$s$	$on\_floor$	$switch\_up$	$s'$	$on\_ground$	$on\_first$	$lift\_up$	$lift\_down$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	1
1	1	1	1	0	1	0	0

## Minimized Boolean Formula

- K-map for  $s'$ :

		$on\_floor \quad switch\_up$			
		00	01	11	10
$s$	0	0	0	1	0
	1	1	1	1	0

- Minimized formula for  $s'$ :  
 $s' = s \text{ on\_floor} + on\_floor \text{ switch\_up}$

# FINITE STATE MACHINES - 3

## Logic Minimization

- State transition table:

Current State	Inputs		Next State	Outputs			
$s$	$on\_floor$	$switch\_up$	$s'$	$on\_ground$	$on\_first$	$lift\_up$	$lift\_down$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	1
1	1	1	1	0	1	0	0

### Minimized Boolean Formula

- K-map for  $on\_ground$ :

		$on\_floor \ switch\_up$			
		00	01	11	10
$s$	0				
	1				

- Minimized formula  $on\_ground$ :  
 $on\_ground = \bar{s} \ on\_floor \ switch\_up$

# FINITE STATE MACHINES - 3

## Logic Minimization

- State transition table:

Current State	Inputs		Next State	Outputs			
<i>s</i>	<i>on_floor</i>	<i>switch_up</i>	<i>s'</i>	<i>on_ground</i>	<i>on_first</i>	<i>lift_up</i>	<i>lift_down</i>
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	1
1	1	1	1	0	1	0	0

## Minimized Boolean Formula

- K-map for *on\_first*:

		<i>on_floor switch_up</i>			
		00	01	11	10
<i>s</i>	0				
	1				

- Minimized formula *on\_first*:  
 $on\_first = s \text{ on\_floor } switch\_up$

# FINITE STATE MACHINES - 3

## Logic Minimization

- State transition table:

Current State	Inputs		Next State	Outputs			
$s$	$on\_floor$	$switch\_up$	$s'$	$on\_ground$	$on\_first$	$lift\_up$	$lift\_down$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	1
1	1	1	1	0	1	0	0

### Minimized Boolean Formula

- K-map for  $lift\_up$ :

	$on\_floor \ switch\_up$			
	00	01	11	10
$s$				
0	0	0	1	0
1	1	1	0	0

- Minimized formula for  $lift\_up$ :  
$$lift\_up = s \ on\_floor + \bar{s} \ on\_floor \ switch\_up$$

# FINITE STATE MACHINES - 3

## Logic Minimization

- State transition table:

Current State	Inputs		Next State	Outputs			
$s$	$on\_floor$	$switch\_up$	$s'$	$on\_ground$	$on\_first$	$lift\_up$	$lift\_down$
0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	1
1	1	1	1	0	1	0	0

## Minimized Boolean Formula

- K-map for  $lift\_down$ :

		$on\_floor \ switch\_up$			
		00	01	11	10
$s$	0	1	1	0	0
	1	0	0	0	1

- Minimized formula for  $lift\_down$ :  
 $lift\_down = \bar{s} \ on\_floor + s \ on\_floor \ switch\_up$

# FINITE STATE MACHINES - 3

## Logic Diagram

- Next state formulas:

- $s' = s \overline{on\_floor} + on\_floor switch\_up$

- Output formulas:

- $on\_ground = \overline{s} \overline{on\_floor} \overline{switch\_up}$
  - $on\_first = s \overline{on\_floor} switch\_up$
  - $lift\_up = s \overline{on\_floor} + \overline{s} on\_floor switch\_up$
  - $lift\_down = \overline{s} on\_floor + s on\_floor switch\_up$

# FINITE STATE MACHINES - 3

## Logic Diagram

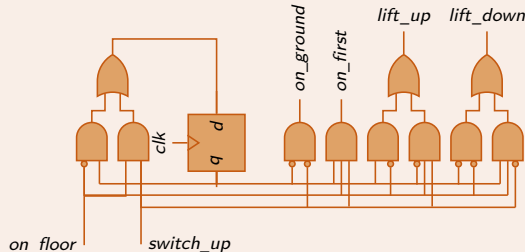
- Next state formulas:

- $s' = s \overline{\text{on\_floor}} + \text{on\_floor switch\_up}$

- Output formulas:

- $\text{on\_ground} = \bar{s} \text{ on\_floor } \overline{\text{switch\_up}}$
- $\text{on\_first} = s \text{ on\_floor } \overline{\text{switch\_up}}$
- $\text{lift\_up} = s \text{ on\_floor} + \bar{s} \text{ on\_floor switch\_up}$
- $\text{lift\_down} = \bar{s} \text{ on\_floor} + s \text{ on\_floor switch\_up}$

### Elevator Example Logic Diagram





# FINITE STATE MACHINES - 3

## Think About It

---



Design of a Moore and a Mealy FSM for the same problem (example 3.7 of your textbook)