# 3.1.1 Finite State Machine

**What is a finite state machine?**

If you have ever used a lift then you have used a finite state machine. A lift can only be at one of a number of finite floors or travelling between floors. In order to function correctly the system needs to maintain a record of the lifts current state including whether the lift doors are open or closed.

FSMs have inputs and output puts which include the following:

**Inputs**

1. A floor selection button has been pressed.
2. Either the open or close button has been pressed.
3. Sensor indicates the door is closed.
4. Sensor indicates the lift is correctly aligned with the floor.
5. Sensor indicates the door is open.

**Outputs**

1. Display information in the lift
2. Display information at each floor
3. Pulses to activate or deactivate motors

**Definition of a Finite State Machine**

This is a machine that can only be in one of a finite number of states at any given time. A set number of inputs are used to change the state of the machine. The next output of state is dependent upon the current state and the input.

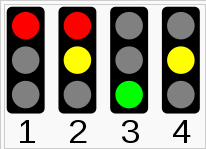
**Exercise 1 – The Finite State Machine**

1. Make a list of five Finite State Machines that can be found in a typical house.
2. Select one of your examples of a FSM and list the following:
   1. States of the machine
   2. Inputs
   3. Outputs

**State Transition Diagrams**

The behaviour of a FSM as it changes between states can be shown using a State Transition Diagram. This will be demonstrated by the use of an example.

Consider a set of traffic lights which can be in one of four states:

1. RED
2. RED & ORANGE
3. GREEN
4. ORANGE

The input for this FSM is a reading from a clock. If the clock is set to zero as soon as the lights turn to red then the subsequent changes can take place at the following times:

1. RED Change after 25 seconds
2. RED & ORANGE Change after 30 seconds
3. GREEN Change after 55 seconds
4. ORANGE Change after 60 seconds

25 seconds 30 seconds 55 seconds

60 seconds

Finite state machines can be represented using a state transition table. The four different states of the traffic lights are S1, S2, S3 and S4, the T is used to represent time. The different state can be represented using the State Transition Table below.

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Current**  **State** | **Output** | **Next**  **State** |
| 0 ≤ T < 25 | S1 | Red | S2 |
| 25 ≤ T < 30 | S2 | Red & Orange | S3 |
| 30 ≤ T < 55 | S3 | Green | S4 |
| 55 ≤ T < 60 | S4 | Orange  T set to 0 | S1 |

**Exercise 2 – Washing Machine**

Construct a State Transition Diagram and a State Transition Table for the operation of a simple washing machine. The washing cycle operates as follows:

* When the **Start Button** is pressed the cycle begins by turning the fill tap on. The **Wash Timer** and the **Rinse Timer** are both set to zero.
* Once **Water Level Sensor** indicates that the machine is full the machine stops filling.
* The **Water Heater** comes on and stays on until the **Thermometer** indicates that the temperature has reached 30 degrees Celsius and then the wash begins.
* Once the **Wash Timer** reaches 20 minutes the machine stops washing and starts to drain.
* Once the **Water Level Sensor** indicates that the machine is empty the machine stops draining and the machine starts to fill ready for the rinse.
* Once **Water Level Sensor** indicates that the machine is full the machine stops filling and because the **Wash Timer** is equal 20 minutes the machine starts to rinse.
* Once the **Rinse Timer** reaches 10 minutes the machine stops rinsing and starts to drain.
* Once **Water Level Sensor** indicates that the machine is empty and the **Rinse Timer** is 10, indicating that both the wash and rinse have taken place.
* The machine then spins for 3 minutes. Once the **Spin Timer** reaches 3 minutes the machine stops.

**Exercise 3 – Central Heating Boiler**

A central heating boiler has a thermostat, an over-ride button, a clock and a facility to hold timer settings, i.e. the times at which the boiler should come on and off during the day. Write down the inputs, outputs and states for the boiler and then construct a State Transition Diagram and a State Transition Table for the operation of the boiler.

**Decision Tables**

A decision table is a compact and precise way to model some complicated logic. Decision tables make it easy to observe that all possible conditions are accounted for.

Consider the code which could be used by an insurance company to calculate the cost of a life insurance policy.

If Gender = “Female” then

If Age <= 50 then

Price = 120.00

Else

Price = 160.00

End If

Else

If Age <= 50 then

Price = 150.00

Else

Price = 175.00

End If

End If

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **Rule or condition options:**  **Y = True, N = False** | | | |
| Conditions | Gender = Female | Y | Y | N | N |
| Age <= 50 | Y | N | Y | N |
| Actions | 120.00 | X |  |  |  |
| 150.00 |  | X |  |  |
| 160.00 |  |  | X |  |
| 175.00 |  |  |  | X |

**Exercise 3 – Decision Tables (Examination Grade)**

A Certificate in Programming is accessed through one examination and a coursework assignment. The pass mark for each assessment is 40%. In order to pass a student must achieve a pass in each assessment. If a student achieves a pass and an average mark of 60% or more they achieve a Distinction. Construct a decision table for this problem.

**Exercise 4 – Decision Tables (ABC Limited)**

ABC Limited sells merchandise to wholesale and retail customers. Wholesale customers receive a 2% discount on all orders. The company also encourages both wholesale and retail customers to pay cash on delivery by offering a further 2% discount for this method of payment. Another 2% discount is given on orders of £500.00 or more. Construct a decision table.