Finite State Machines (FSMs)

A **Finite State Machine** is a computational model defined by the 5-tuple:

$$M = (Q, \Sigma, \delta, q_0, F)$$

where:

- Q is a finite set of states.
- Σ is a finite input alphabet.
- $\delta : Q \times \Sigma \to Q$ is the state transition function.
- $q_0 \in Q$ is the initial state.
- $F \subseteq Q$ is the set of accepting (final) states.

Moore Machine

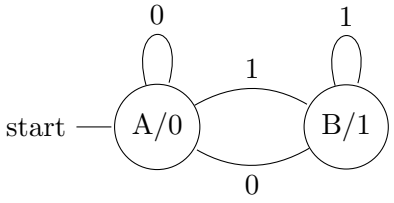
A *Moore machine* is an FSM where the outputs depend only on the current state. It is a 6-tuple:

$$M_{Moore} = (Q, \Sigma, \Lambda, \delta, \lambda, q_0)$$

where:

- Q, Σ, q_0 as before.
- Λ is the finite output alphabet.
- $\delta : Q \times \Sigma \to Q$ is the transition function.
- $\lambda : Q \to \Lambda$ is the output function.

Example diagram:



Mealy Machine

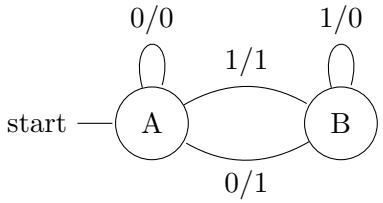
A *Mealy machine* is an FSM where the outputs depend on both the current state and the current input. It is defined as:

$$M_{Mealy} = (Q, \Sigma, \Lambda, \delta, \omega, q_0)$$

where:

- Q, Σ, q_0 as before.
- Λ is the finite output alphabet.
- $\delta : Q \times \Sigma \to Q$ is the state transition function.
- $\omega : Q \times \Sigma \to \Lambda$ is the output function.

Example diagram:



Resources

- YouTube playlist on FSMs: Finite State Machines Playlist