

Mealy Machines

NOTE: If we can compose (output to input of another machine) these machines, we could create complete simulations.

Motivation

Mealy Machines have an output function which is a function of both the input and current state. These systems can be considered to *consume* input events, *produce* output events and update its internal state. These systems are often used to model network protocols or other message passing systems.

Application

Consider the Transport Control Protocol (TCP) shown in the figure.

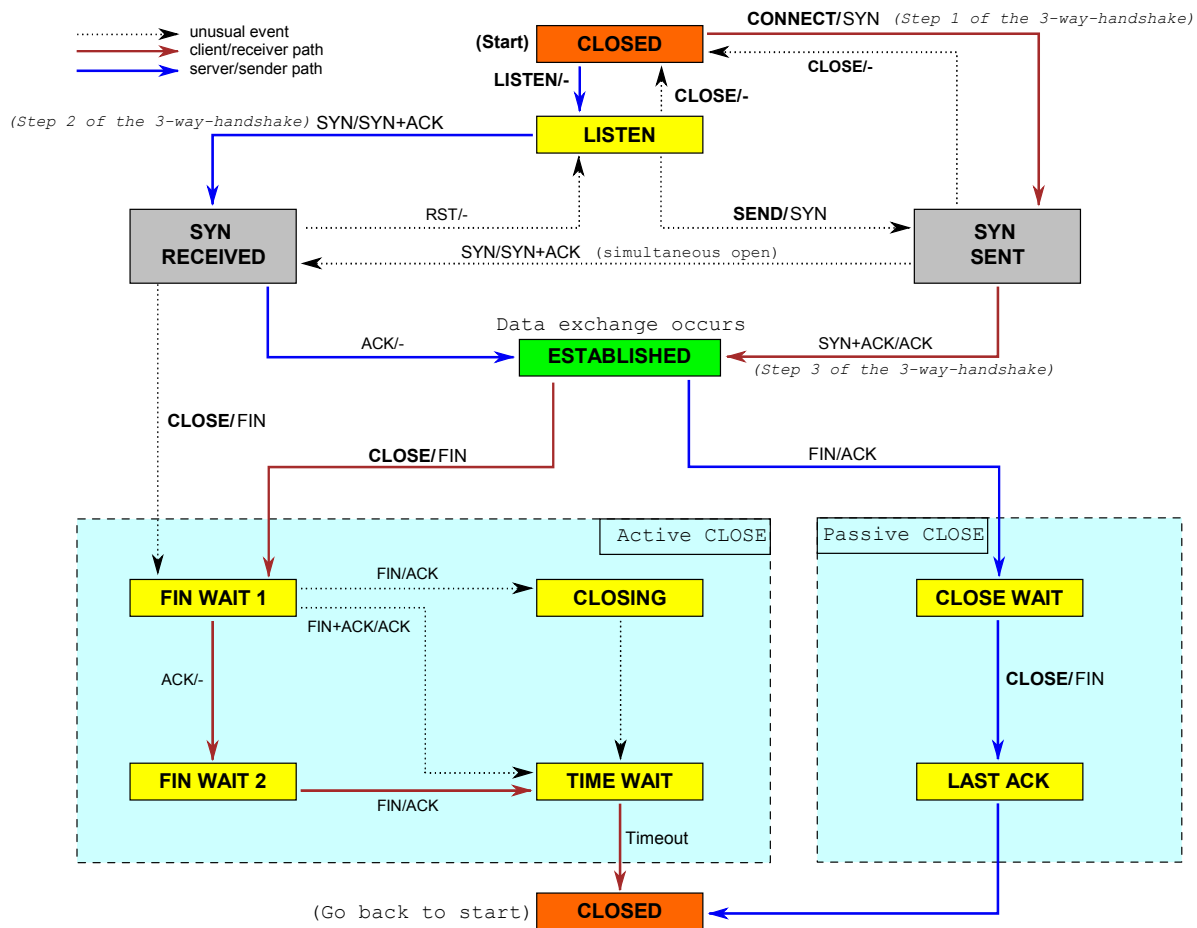


Illustration 1: http://upload.wikimedia.org/wikipedia/commons/a/a2/Tcp_state_diagram_fixed.svg

Questions Tasks

- 1) From the diagram extract all of the “events” or symbols in the input alphabet.
- 2) What are the output alphabet or “events”

The input events are from the following set:

{LISTEN,CLOSE,CONNECT,SEND,RST,ACK,SYN,SYN+ACK,FIN,FIN+ACK,TIMEOUT}

The output events are from the following set:

{-,FIN,SYN,ACK,SYN+ACK}

The states of the machine are:

{CLOSED, LISTEN, SYN_RECIEVED, SYN_SENT, ESTABLISHED,
FIN_WAIT1, FIN_WAIT2, CLOSING, TIME_WAIT, CLOSE_WAIT, LAST_ACK}

Questions Tasks

- 1) Using the above events and mapping the states to q_0, \dots, q_{11} create a Mealy machine.
(the following may be helpful:
http://tcpipguide.com/free/t_TCPOperationalOverviewandtheTCPFiniteStateMachineF-2.htm)
- 2) Use the resulting machine to discuss the following
 1. If the client attempts to CONNECT to a server, events must occur to reach the ESTABLISHED state? Are there any other ways to reach the ESTABLISHED state after receiving a CONNECT event?
 2. What sequence(s) of events occur after a server receives a LISTEN event to reach the ESTABLISHED state?
 3. What sequence(s) of events occur when the may occur to reach the CLOSED state?

