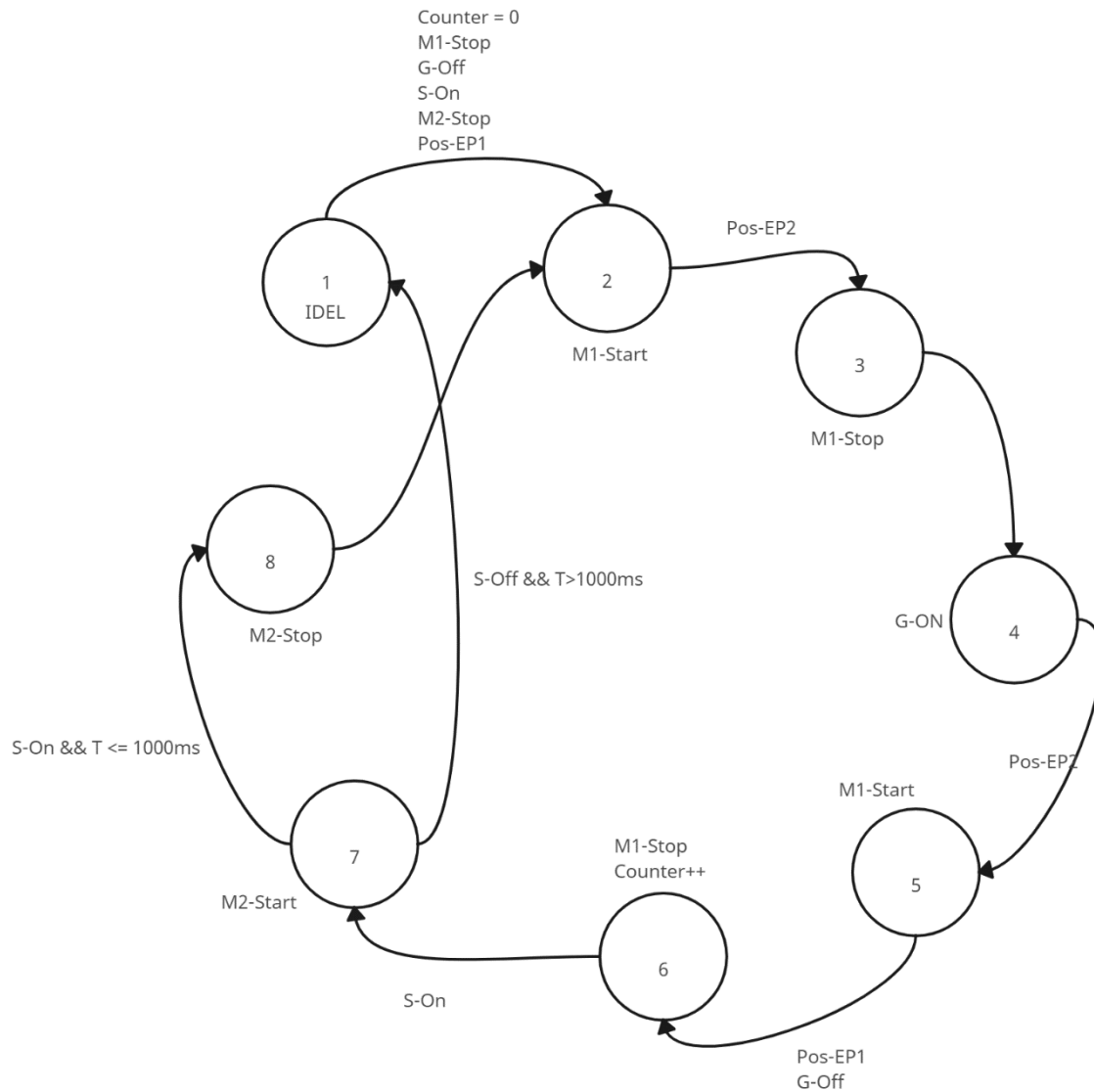


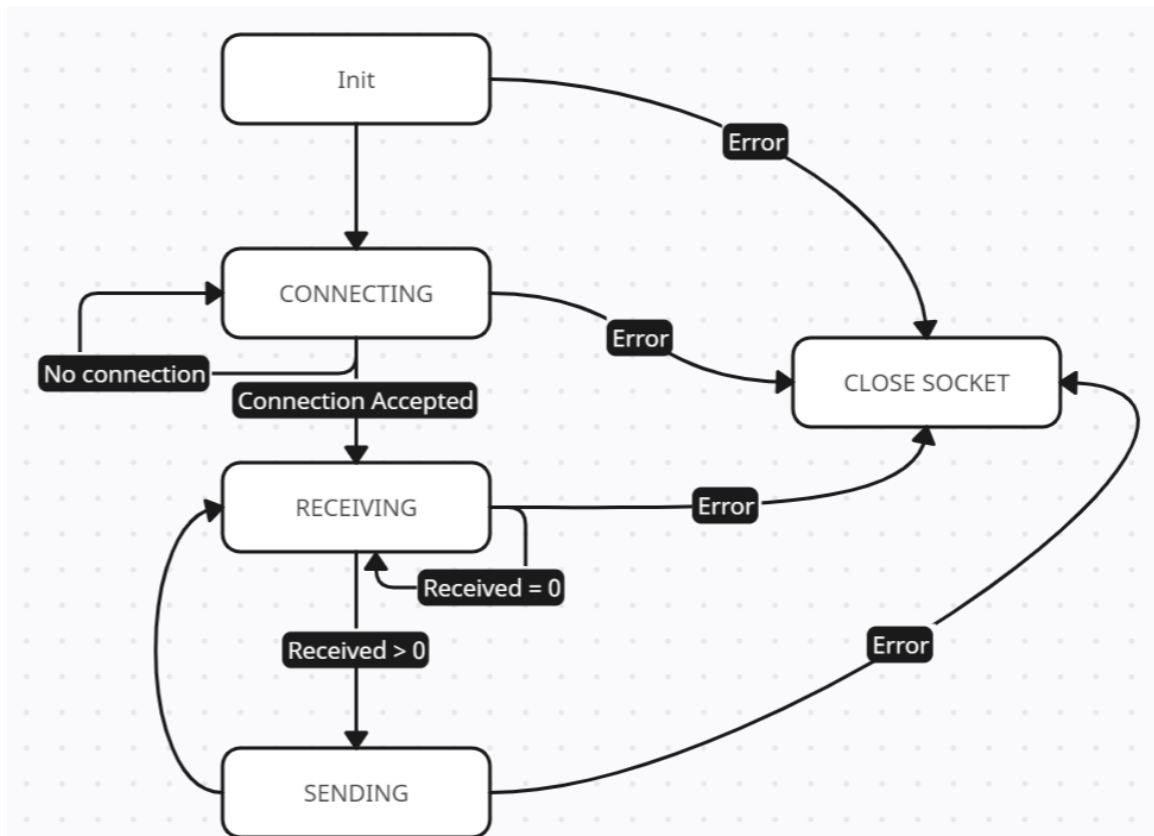
State Diagram implementation for Cylinder Pick and Place



Description

1. Ideal - Initial home position
2. Counter – pick and place cycle counter
3. M1 – linear axis slide motor
4. M2 – spinning disk motor
5. G – gripper
6. S – NO (NPN) cylinder presence sensor
7. Pos EP1 – End position linear axis 1 (place side)
8. Pos EP2 – End position linear axis 2 (pick side)
9. T – Cycle time between each cylinder

State Diagram implementation for TwinCAT_TCP_IP_Server and Python client signal exchange



How it works

- 1.The function block FB_TcpServer will handle block parameters and sockets for FB_SocketListen, FB_SocketAccept, FB_SocketSend and FB_SocketReceive.
- 2.Open Listener socket using FB_SocketListen then TwinCAT TCP/IP Connection Server can 'listen' for incoming connection requests from remote clients.
- 3.Python client will connect to the host and the port.
- 4.The connection handle of the listener sockets is transferred to the function block FB_SocketAccept. FB_SocketAccept will then return the connection handles of the remote clients.
- 5.Send and receive the data packets are managed by 100ms timer which FB_SocketAccept accepts the incoming remote client connection requests, opens a new remote client socket and returns the associated connection handle.
- 6.The connection handle is then transferred to the function blocks FB_SocketSend and/or FB_SocketReceive, in order to be able to exchange data with the remote clients;

7.FB_SocketReceive will copy received client data and send it back via FB_SocketSend and will continue back and forth between FB_SocketSend & FB_SocketReceive while there are connection handles from FB_SocketAccept

8.On error listener socket connection handle will be transferred to the function block FB_SocketClose, which closes the listener socket