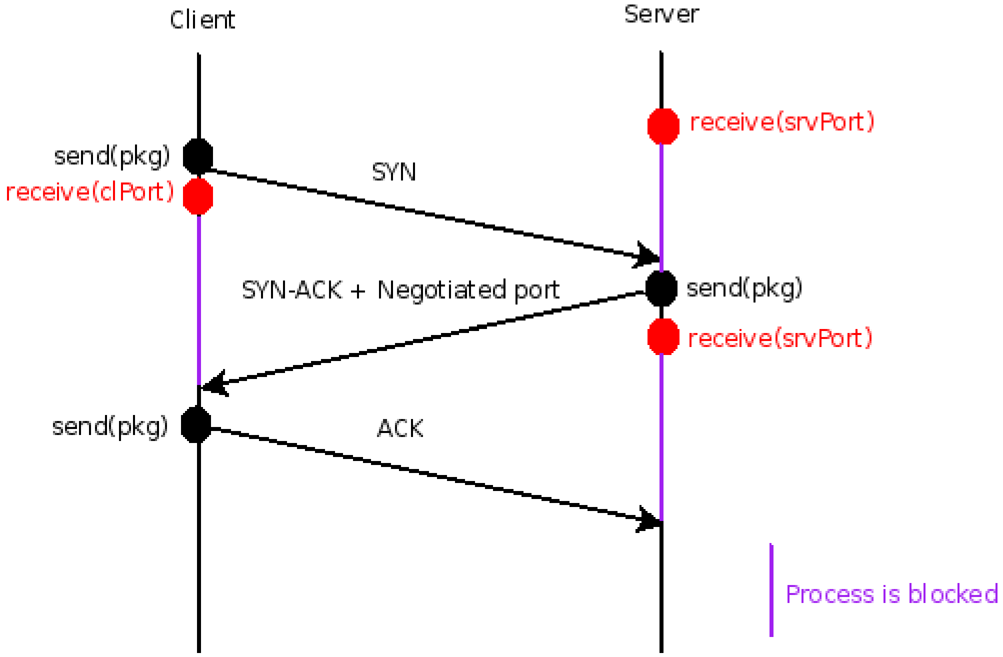
Message sequence charts

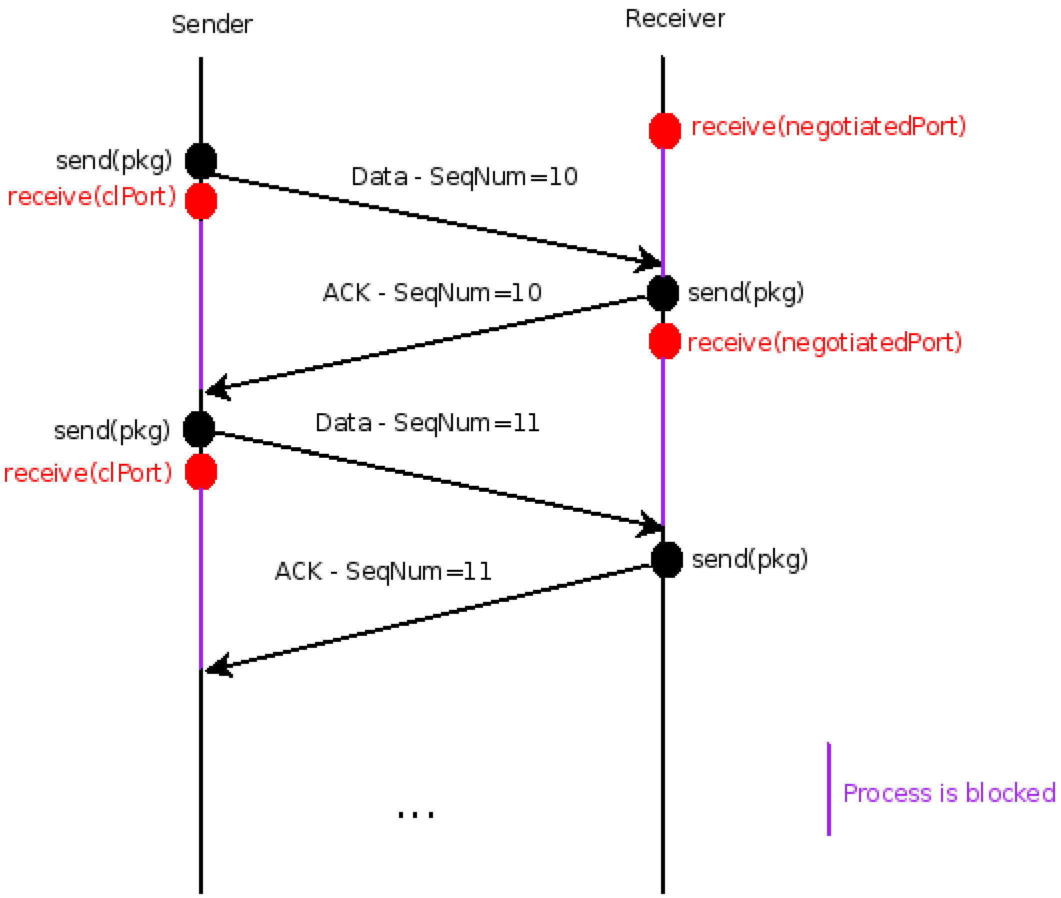
## Connection

The connection process concerns to the methods accept() and connect() of A1, and it is implemented with a three-way handshake using the methods send() and receive() of A2.
The flags are enabled through the set methods provided by the class KtnDatagram.
Once the connection is established, the client will connect with the server using another port which has been negotiated during this process.



Sending and receiving

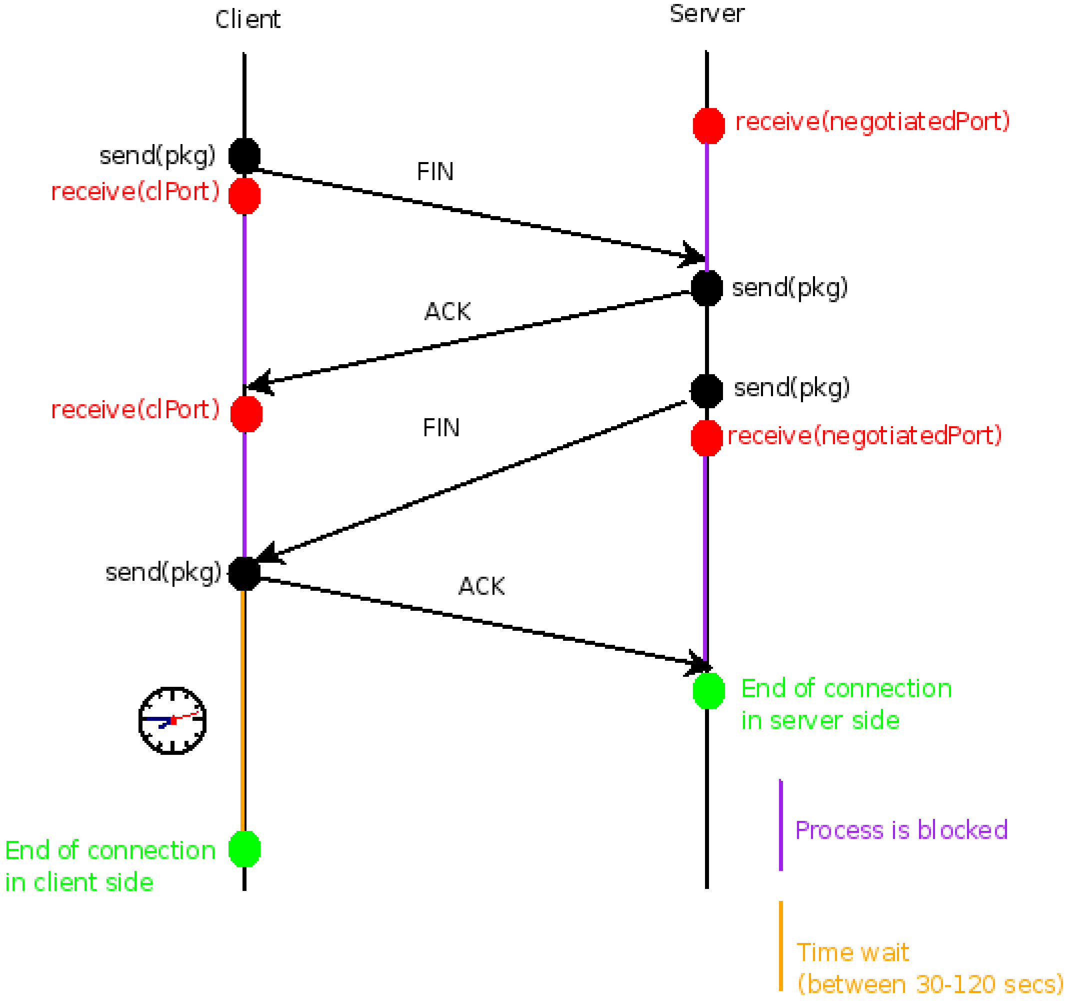
The methods send() and receive() of A1 are implemented by send() and receive() of A2 in an scenario without errors as is shown in this chart (errors handling is explained in subsequent sections).
As well as the flags, the access to the sequence number is provided by KtnDatagram methods.



This example suppose the server as receiver to illustrate the use of the negotiated port in next stages, but the server could also be the sender.

Disconnection

As in the termination of the connection process in TCP, we are going to implement a four-way handshake, including a TIME_WAIT state where the client retransmits the final ACK in case the ACK is lost. This scenario refers to the method close() in A1.


Client
Server
receive(negotiatedPort)
send(pkg)
receive(clPort)
FIN
send(pkg)
ACK
send(pkg)
receive(clPort)
FIN
receive(negotiatedPort)
send(pkg)
ACK
End of connection
in server side
Process is blocked
End of connection
in client side
Time wait
(between 30-120 secs)