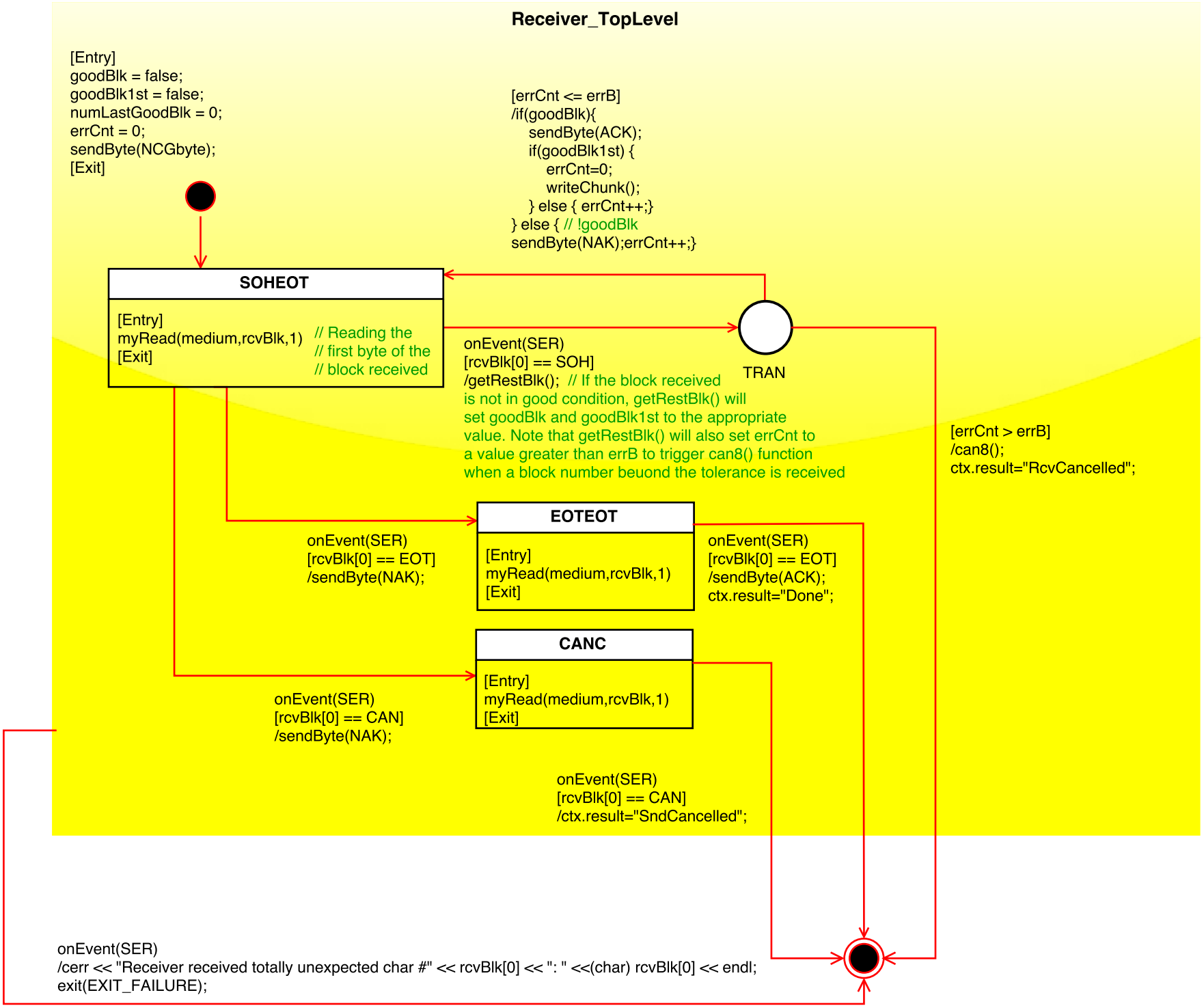
Receiver_TopLevel
[Entry]
goodBlk = false;
goodBlk1st = false;
numLastGoodBlk = 0;
errCnt = 0;
sendByte(NCGbyte);
[Exit]
[errCnt <= errB]
/if(goodBlk){
sendByte(ACK);
if(goodBlk1st) {
errCnt=0;
writeChunk();
} else { errCnt++;}
} else { // !goodBlk
sendByte(NAK);errCnt++;}
SOHEOT
[Entry]
myRead(medium,rcvBlk,1)
[Exit]
// Reading the
// first byte of the
// block received
TRAN
onEvent(SER)
[rcvBlk[0] == SOH]
/getRestBlk(); // If the block received
is not in good condition, getRestBlk() will
set goodBlk and goodBlk1st to the appropriate
value. Note that getRestBlk() will also set errCnt to
a value greater than errB to trigger can8() function
when a block number beuond the tolerance is received
[errCnt > errB]
/can8();
ctx.result="RcvCancelled";
EOTEOT
[Entry]
myRead(medium,rcvBlk,1)
[Exit]
onEvent(SER)
[rcvBlk[0] == EOT]
/sendByte(NAK);
onEvent(SER)
[rcvBlk[0] == EOT]
/sendByte(ACK);
ctx.result="Done";
CANC
[Entry]
myRead(medium,rcvBlk,1)
[Exit]
onEvent(SER)
[rcvBlk[0] == CAN]
/sendByte(NAK);
onEvent(SER)
[rcvBlk[0] == CAN]
/ctx.result="SndCancelled";
onEvent(SER)
/cerr << "Receiver received totally unexpected char #" << rcvBlk[0] << ": " <<(char) rcvBlk[0] << endl;
exit(EXIT_FAILURE);