

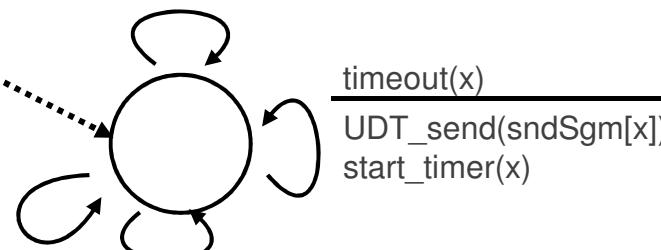
SR

```
RDT_send(data)
if (nextseqnum < send_base+N) {
    sndSgm[nextseqnum] = make_segment(nextseqnum,data)
    UDT_send(sndSgm[nextseqnum])
    start_timer(nextseqnum)
    nextseqnum++
}
else
    refuse_data(data)
```

```
send_base=1
nextseqnum=1
forEach y: isAckedSgmt[y]=false
```

```
rcvSgm=UDT_rcv() &&
( corrupted(rcvSgm) || !isACKinWindow(rcvSgm) )
```

Δ

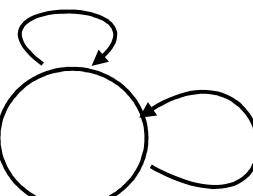


```
rcvSgm=UDT_rcv() && !corrupted(rcvSgm) && isACKinWindow(rcvSgm)
y=getacknum(rcvSgm)
isAckedSgmt[y]=true
stop_timer(y)
If (y==send_base)
    while (isAckedSgmt[send_base]==true && send_base!=nextseqnum) do
        {isAckedSgmt[send_base]=false; send_base++}
```

```
rcvSgm=UDT_rcv() && corrupted(rcvSgm)
```

Δ

```
rcv_base=1
forEach y: isReceivedSgmt[y]=false
```



```
rcvSgm=UDT_rcv() && !corrupted(rcvSgm)
y=seqN(rcvSgm)
if (isReceivedSgmt[y]==false)
    {rcvSgmt[y]=extract(rcvSgm); isReceivedSgmt[y]=true}
    sndSgm = make_segment(ACK, y)
    UDT_send(sndSgm)
if (y==rcv_base)
    while (isReceivedSgmt[rcv_base]==true) do {
        deliver_data(rcvSgmt[rcv_base])
        isReceivedSgmt[y]=false
        rcv_base++}
```