

```

1  |----- MODULE AJupiterExtended -----|
   | AJupiter extended with JupiterCtx. This is used to show that AJupiter implements XJupiter. |
5  |-----|
   | EXTENDS JupiterCtx |
6  |-----|
7  VARIABLES cbuf, crec, sbuf, srec, cincomingXJ, sincomingXJ

9  commXJVars  $\triangleq$   $\langle \text{cincomingXJ}, \text{sincomingXJ} \rangle$ 
10 commXJ  $\triangleq$  INSTANCE CSComm WITH Msg  $\leftarrow$  Seq(Cop),
11      cincoming  $\leftarrow$  cincomingXJ, sincoming  $\leftarrow$  sincomingXJ

13 varsEx  $\triangleq$   $\langle \text{intVars}, \text{ctxVars}, \text{cbuf}, \text{crec}, \text{sbuf}, \text{srec}, \text{commXJVars} \rangle$ 

15 Msg  $\triangleq$  [ack : Int, cop : Cop, oid : Oid]
16 |-----|
17 TypeOKEx  $\triangleq$ 
18    $\wedge$  TypeOKInt
19    $\wedge$  TypeOKCtx
20    $\wedge$  Comm(Msg)! TypeOK
21    $\wedge$  commXJ! TypeOK
22    $\wedge$  crec  $\in$  [Client  $\rightarrow$  Int]
23    $\wedge$  srec  $\in$  [Client  $\rightarrow$  Int]
24    $\wedge$  cbuf  $\in$  [Client  $\rightarrow$  Seq(Cop)]
25    $\wedge$  sbuf  $\in$  [Client  $\rightarrow$  Seq(Cop)]
26 |-----|
27 InitEx  $\triangleq$ 
28    $\wedge$  InitInt
29    $\wedge$  InitCtx
30    $\wedge$  commXJ! Init
31    $\wedge$  Comm(Msg)! Init
32    $\wedge$  crec = [c  $\in$  Client  $\mapsto$  0]
33    $\wedge$  srec = [c  $\in$  Client  $\mapsto$  0]
34    $\wedge$  cbuf = [c  $\in$  Client  $\mapsto$   $\langle \rangle$ ]
35    $\wedge$  sbuf = [c  $\in$  Client  $\mapsto$   $\langle \rangle$ ]
36 |-----|
37 DoOp(c, op)  $\triangleq$ 
38   LET cop  $\triangleq$  [op  $\mapsto$  op, oid  $\mapsto$  [c  $\mapsto$  c, seq  $\mapsto$  cseq'[c], ctx  $\mapsto$  ds[c]]
39   IN    $\wedge$  crec' = [crec EXCEPT ![c] = 0]
40        $\wedge$  cbuf' = [cbuf EXCEPT ![c] = Append(@, cop)]
41        $\wedge$  state' = [state EXCEPT ![c] = Apply(op, @)]
42        $\wedge$  Comm(Msg)! CSend([ack  $\mapsto$  crec[c], cop  $\mapsto$  cop, oid  $\mapsto$  cop.oid])
43        $\wedge$  commXJ! CSend(cop)

45 DoEx(c)  $\triangleq$ 
46    $\wedge$  DoCtx(c)
47    $\wedge$  DoInt(DoOp, c)
48    $\wedge$  UNCHANGED  $\langle \text{sbuf}, \text{srec} \rangle$ 

```

```

50  $RevEx(c) \triangleq$ 
51    $\wedge Comm(Msg)!CRev(c)$ 
52    $\wedge commXJ!CRev(c)$ 
53    $\wedge crec' = [crec \text{ EXCEPT } ![c] = @ + 1]$ 
54    $\wedge \text{LET } m \triangleq Head(cincoming[c])$ 
55      $cBuf \triangleq cbuf[c]$ 
56      $cShiftedBuf \triangleq SubSeq(cBuf, m.ack + 1, Len(cBuf))$ 
57      $xcop \triangleq XformOpOps(COT, m.cop, cShiftedBuf)$ 
58      $xcBuf \triangleq XformOpsOp(COT, cShiftedBuf, m.cop)$ 
59   IN    $\wedge cbuf' = [cbuf \text{ EXCEPT } ![c] = xcBuf]$ 
60        $\wedge state' = [state \text{ EXCEPT } ![c] = Apply(xcop.op, @)]$ 
61    $\wedge RevCtx(c)$ 
62    $\wedge RevInt(c)$ 
63    $\wedge \text{UNCHANGED } \langle sbuf, srec \rangle$ 

65  $SRevEx \triangleq$ 
66    $\wedge Comm(Msg)!SRev$ 
67    $\wedge commXJ!SRev$ 
68    $\wedge \text{LET } m \triangleq Head(sincoming)$ 
69        $c \triangleq ClientOf(m.cop)$ 
70        $cBuf \triangleq sbuf[c]$ 
71        $cShiftedBuf \triangleq SubSeq(cBuf, m.ack + 1, Len(cBuf))$ 
72        $xcop \triangleq XformOpOps(COT, m.cop, cShiftedBuf)$ 
73        $xcBuf \triangleq XformOpsOp(COT, cShiftedBuf, m.cop)$ 
74   IN    $\wedge srec' = [cl \in Client \mapsto$ 
75       IF  $cl = c$  THEN  $srec[cl] + 1$  ELSE  $0]$ 
76        $\wedge sbuf' = [cl \in Client \mapsto$ 
77       IF  $cl = c$  THEN  $xcBuf$  ELSE  $Append(sbuf[cl], xcop)]$ 
78        $\wedge state' = [state \text{ EXCEPT } ![Server] = Apply(xcop.op, @)]$ 
79        $\wedge Comm(Msg)!SSend(c, [cl \in Client \mapsto [ack \mapsto srec[cl], cop \mapsto xcop, oid \mapsto xcop.oid]])$ 
80        $\wedge commXJ!SSendSame(c, xcop)$ 
81    $\wedge SRevCtx$ 
82    $\wedge SRevInt$ 
83    $\wedge \text{UNCHANGED } \langle cbuf, crec \rangle$ 

84 |-----|
85  $NextEx \triangleq$ 
86    $\vee \exists c \in Client : DoEx(c) \vee RevEx(c)$ 
87    $\vee SRevEx$ 

89  $FairnessEx \triangleq$  There is no requirement that the clients ever generate operations.
90    $WF_{varsEx}(SRevEx \vee \exists c \in Client : RevEx(c))$ 

92  $SpecEx \triangleq InitEx \wedge \Box[NextEx]_{varsEx} \wedge FairnessEx$ 
93 |-----|
94  $QC \triangleq$  Quiescent Consistency
95    $Comm(Msg)!EmptyChannel \Rightarrow Cardinality(Range(state)) = 1$ 

```

97 THEOREM $SpecEx \Rightarrow \Box QC$

98 |

* Modification History

* Last modified *Mon Dec 31 21:05:54 CST 2018* by *hengxin*

* Created *Thu Dec 27 21:15:09 CST 2018* by *hengxin*