

```

1  |----- MODULE JupiterInterface -----|
   |Interface of a family of Jupiter protocols.
5  |-----|
   |EXTENDS Op
6  |-----|
7  |VARIABLES
8      aop,      aop[r]: the actual operation applied at replica r ∈ Replica
9      state,    state[r]: state (the list content) of replica r ∈ Replica
10     chins     a set of chars allowed to insert; this is for model checking
12  intVars  $\triangleq$   $\langle aop, state, cincoming, sincoming, chins \rangle$ 
13  |-----|
14  SetNewAop(r, aopr)  $\triangleq$ 
15     aop' = [aop EXCEPT ![r] = aopr]
17  ApplyNewAop(r)  $\triangleq$ 
18     state' = [state EXCEPT ![r] = Apply(aop'[r], @)]
19  |-----|
20  TypeOKInt  $\triangleq$ 
21      $\wedge aop \in [Replica \rightarrow Op \cup \{Nop\}]$ 
22      $\wedge state \in [Replica \rightarrow List]$ 
23      $\wedge Comm!TypeOK$ 
24      $\wedge chins \subseteq Char$ 
26  InitInt  $\triangleq$ 
27      $\wedge aop = [r \in Replica \mapsto Nop]$ 
28      $\wedge state = [r \in Replica \mapsto InitState]$ 
29      $\wedge Comm!Init$ 
30      $\wedge chins = Char$ 
32  DoIns(DoOp(-, -), c)  $\triangleq$  Client c ∈ Client generates and processes an “Ins” operation.
33      $\exists ins \in Ins :$ 
34          $\wedge ins.pos \in 1 \dots (Len(state[c]) + 1)$ 
35          $\wedge ins.ch \in chins$ 
36          $\wedge ins.pr = Priority[c]$ 
37          $\wedge DoOp(c, ins)$ 
38          $\wedge chins' = chins \setminus \{ins.ch\}$  We assume that all inserted elements are unique.
40  DoDel(DoOp(-, -), c)  $\triangleq$  Client c ∈ Client generates and processes a “Del” operation.
41      $\exists del \in Del :$ 
42          $\wedge del.pos \in 1 \dots Len(state[c])$ 
43          $\wedge DoOp(c, del)$ 
44          $\wedge UNCHANGED chins$ 
46  DoInt(DoOp(-, -), c)  $\triangleq$  Client c ∈ Client generates an operation.
47      $\wedge \vee DoIns(DoOp, c)$  DoOp(c ∈ Client, op ∈ Op)
48      $\vee DoDel(DoOp, c)$ 

```

```

49  ∧ ApplyNewAop(c)

51  RevInt(ClientPerform(−, −), c) ≜ Client c ∈ Client receives and processes a message.
52  ∧ Comm!CRev(c)
53  ∧ ClientPerform(c, Head(cincoming[c])) ClientPerform(c ∈ Client, m ∈ Msg)
54  ∧ ApplyNewAop(c)
55  ∧ UNCHANGED chins

57  SRevInt(ServerPerform(−)) ≜ The Server receives and processes a message.
58  ∧ Comm!SRev
59  ∧ ServerPerform(Head(sincoming)) ServerPerform(m ∈ Msg)
60  ∧ ApplyNewAop(Server)
61  ∧ UNCHANGED chins
62  |
    /* Modification History
    /* Last modified Sun Jan 13 10:53:07 CST 2019 by hengxin
    /* Created Tue Dec 04 19:01:01 CST 2018 by hengxin

```