MODULE *heartbeat*

EXTENDS *TLC*, *Integers*, *FiniteSets*

```
--algorithm heartbeat
variables alive = 1 .. 3, replOwner = [x ∈ 1 .. 3 ↦ x], replStuck = [x ∈ 1 .. 3 ↦ FALSE], killed = {} ;
fair process node ∈ 1 .. 3
begin P:
    while self ∈ alive do
        CheckIfStuck:
            if {x ∈ 1 .. 3 : replOwner[x] = self ∧ replStuck[x]} ≠ {} then
                killed := killed ∪ {x ∈ 1 .. 3 : replStuck[x] ∧ replOwner[x] = self} ;
            end if ;
        RestartReplicator:
            if {x ∈ 1 .. 3 : replOwner[x] = self ∧ replStuck[x]} ≠ {} then
                replStuck := [x ∈ 1 .. 3 ↦ CASE replOwner[x] = self → FALSE□OTHER → replStuck[x]] ;
                killed := killed \ {x ∈ 1 .. 3 : replOwner[x] = self} ;
            end if ;
    end while ;
    NodeDown:
    await self ∈ alive ;
    goto P ;
end process

fair + process orchestrator = 0
begin Orchestrator:
    while alive ≠ {} do
        either
            RebootNode:
                if Cardinality(alive) > 1 then
                    with x ∈ alive do
                        alive := alive \ {x} ;
                        replOwner := [z ∈ 1 .. 3 ↦
                            CASE replOwner[z] = x → CHOOSE y ∈ alive : x ≠ y
                            □OTHER → replOwner[z]
                        ] ;
                    end with ;
                end if ;
        or
            MakeReplicatorStuck:
                with x ∈ 1 .. 3 do
                    replStuck[x] := TRUE ;
                end with ;
        end either ;
    end while ;
end process
```

end algorithm ;

BEGIN TRANSLATION ($chksum(pcal) =$ "23f8c013" $\land\ chksum(tla) =$ "d5065706")

VARIABLES *alive*, *replOwner*, *replStuck*, *killed*, *pc*

$vars \triangleq \langle alive, replOwner, replStuck, killed, pc \rangle$

$ProcSet \triangleq (1 \,..\, 3) \cup \{0\}$

$Init \triangleq$ Global variables
$\land\ alive = 1 \,..\, 3$
$\land\ replOwner = [x \in 1 \,..\, 3 \mapsto x]$
$\land\ replStuck\ = [x \in 1 \,..\, 3 \mapsto \text{FALSE}]$
$\land\ killed = \{\}$
$\land\ pc = [self \in ProcSet \mapsto \text{CASE } self \in 1 \,..\, 3 \to$ "P"
$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \square\quad self = 0 \to$ "Orchestrator"$]$

$P(self) \triangleq \land\ pc[self] =$ "P"
$\qquad\land$ IF $self \in alive$
$\qquad\qquad$ THEN $\land\ pc' = [pc$ EXCEPT $![self] =$ "CheckIfStuck"$]$
$\qquad\qquad$ ELSE $\land\ pc' = [pc$ EXCEPT $![self] =$ "NodeDown"$]$
$\qquad\land$ UNCHANGED $\langle alive, replOwner, replStuck, killed \rangle$

$CheckIfStuck(self) \triangleq \land\ pc[self] =$ "CheckIfStuck"
$\qquad\qquad\land$ IF $\{x \in 1 \,..\, 3 : replOwner[x] = self \land replStuck[x]\} \neq \{\}$
$\qquad\qquad\qquad$ THEN $\land\ killed' = (killed \cup \{x \in 1 \,..\, 3 : replStuck[x] \land replOwner[x] = self\})$
$\qquad\qquad\qquad$ ELSE $\land$ TRUE
$\qquad\qquad\qquad\qquad\land$ UNCHANGED $killed$
$\qquad\qquad\land\ pc' = [pc$ EXCEPT $![self] =$ "RestartReplicator"$]$
$\qquad\qquad\land$ UNCHANGED $\langle alive, replOwner, replStuck \rangle$

$RestartReplicator(self) \triangleq \land\ pc[self] =$ "RestartReplicator"
$\qquad\qquad\land$ IF $\{x \in 1 \,..\, 3 : replOwner[x] = self \land replStuck[x]\} \neq \{\}$
$\qquad\qquad\qquad$ THEN $\land\ replStuck' = [x \in 1 \,..\, 3 \mapsto$ CASE $replOwner[x] = self \to$ FALSE$\square$
$\qquad\qquad\qquad\qquad\land\ killed' = killed \setminus \{x \in 1 \,..\, 3 : replOwner[x] = self\}$
$\qquad\qquad\qquad$ ELSE $\land$ TRUE
$\qquad\qquad\qquad\qquad\land$ UNCHANGED $\langle replStuck, killed \rangle$
$\qquad\qquad\land\ pc' = [pc$ EXCEPT $![self] =$ "P"$]$
$\qquad\qquad\land$ UNCHANGED $\langle alive, replOwner \rangle$

$NodeDown(self) \triangleq \land\ pc[self] =$ "NodeDown"
$\qquad\qquad\land\ self \in alive$
$\qquad\qquad\land\ pc' = [pc$ EXCEPT $![self] =$ "P"$]$
$\qquad\qquad\land$ UNCHANGED $\langle alive, replOwner, replStuck, killed \rangle$

$node(self) \triangleq P(self) \lor CheckIfStuck(self) \lor RestartReplicator(self)$
$\qquad\qquad\lor NodeDown(self)$

$Orchestrator \triangleq \land\ pc[0] =$ "Orchestrator"

$$
\begin{array}{l}
\quad \land \text{IF } alive \neq \{\} \\
\qquad \text{THEN } \land \lor \land pc' = [pc \text{ EXCEPT } ![0] = \text{“RebootNode”}] \\
\qquad\qquad\qquad \lor \land pc' = [pc \text{ EXCEPT } ![0] = \text{“MakeReplicatorStuck”}] \\
\qquad \text{ELSE } \land pc' = [pc \text{ EXCEPT } ![0] = \text{“Done”}] \\
\quad \land \text{UNCHANGED } \langle alive, replOwner, replStuck, killed \rangle
\end{array}
$$

$$
\begin{array}{lll}
RebootNode & \triangleq & \land pc[0] = \text{“RebootNode”} \\
 & & \land \text{IF } Cardinality(alive) > 1 \\
 & & \qquad \text{THEN } \land \exists\, x \in alive : \\
 & & \qquad\qquad \land alive' = alive \setminus \{x\} \\
 & & \qquad\qquad \land replOwner' = [z \in 1 \,..\, 3 \mapsto \\
 & & \qquad\qquad\qquad \text{CASE } replOwner[z] = x \to \text{CHOOSE } y \in alive' : x \neq y \\
 & & \qquad\qquad\qquad \square\text{OTHER } \to replOwner[z] \\
 & & \qquad\qquad] \\
 & & \qquad \text{ELSE } \land \text{TRUE} \\
 & & \qquad\qquad \land \text{UNCHANGED } \langle alive, replOwner \rangle \\
 & & \land pc' = [pc \text{ EXCEPT } ![0] = \text{“Orchestrator”}] \\
 & & \land \text{UNCHANGED } \langle replStuck, killed \rangle
\end{array}
$$

$$
\begin{array}{lll}
MakeReplicatorStuck & \triangleq & \land pc[0] = \text{“MakeReplicatorStuck”} \\
 & & \land \exists\, x \in 1 \,..\, 3 : \\
 & & \qquad replStuck' = [replStuck \text{ EXCEPT } ![x] = \text{TRUE}] \\
 & & \land pc' = [pc \text{ EXCEPT } ![0] = \text{“Orchestrator”}] \\
 & & \land \text{UNCHANGED } \langle alive, replOwner, killed \rangle
\end{array}
$$

$$
orchestrator \triangleq Orchestrator \lor RebootNode \lor MakeReplicatorStuck
$$

Allow infinite stuttering to prevent deadlock on termination.

$$
\begin{array}{lll}
Terminating & \triangleq & \land \forall\, self \in ProcSet : pc[self] = \text{“Done”} \\
 & & \land \text{UNCHANGED } vars
\end{array}
$$

$$
\begin{array}{lll}
Next & \triangleq & orchestrator \\
 & & \quad \lor (\exists\, self \in 1 \,..\, 3 : node(self)) \\
 & & \quad \lor Terminating
\end{array}
$$

$$
\begin{array}{lll}
Spec & \triangleq & \land Init \land \square[Next]_{vars} \\
 & & \land \forall\, self \in 1 \,..\, 3 : \mathrm{WF}_{vars}(node(self)) \\
 & & \land \mathrm{SF}_{vars}(orchestrator)
\end{array}
$$

$$
Termination \triangleq \Diamond(\forall\, self \in ProcSet : pc[self] = \text{“Done”})
$$

END TRANSLATION

$$
MyProperty \triangleq \forall\, x \in 1 \,..\, 3 : (\Diamond(replStuck[x] \rightsquigarrow \neg replStuck[x] \lor alive = \{x\} \lor x \in killed))
$$