MODULE *SimpleElevatorControl*

1. Loop through elevators (currently only one) a. Is elevator available
   i. Yes 1. Is a request made?
      a. Yes i. Make elevator unavailable. ii. Move elevator to service request
      b. No, do nothing
   ii. No, do nothing for that elevator because it is servicing a request

This module only has a single elevator but it works more like a normal elevator, in that it increment and decrements one floor at a time. Statistics:
Diameter = 12 States Found = 166 Distinct States = 102

I believe that additional improvements could be made to this to use invariants in order to check to every request is serviced and so forth. This in itself was nearly 20 hours of work to refine it and get it to its current state, so I will accept it as a imperfect design but much learned.

EXTENDS *Naturals*, *TLC*

CONSTANT *TotalFloors*

VARIABLE *elevator*, *request*

$$
\begin{array}{l}
\mathit{TypeInvariant} \triangleq \\
\quad \land\ \mathit{elevator} \in [\mathit{floor} : (1 \,..\, \mathit{TotalFloors}), \\
\qquad\qquad\ \ \mathit{available} : \{\text{TRUE}, \text{FALSE}\}, \\
\qquad\qquad\ \ \mathit{requestedFloor} : (0 \,..\, \mathit{TotalFloors})] \\
\quad \land\ \mathit{request} \in [(1 \,..\, \mathit{TotalFloors}) \to \{\text{TRUE}, \text{FALSE}\}]
\end{array}
$$

$$
\begin{array}{l}
\mathit{Init} \triangleq \\
\quad \land\ \mathit{TypeInvariant} \\
\quad \land\ \mathit{elevator}.\mathit{floor} = 1 \\
\quad \land\ \mathit{elevator}.\mathit{available} = \text{TRUE} \\
\quad \land\ \mathit{elevator}.\mathit{requestedFloor} = 0 \\
\quad \land\ \mathit{request} = [\mathit{req} \in (1 \,..\, \mathit{TotalFloors}) \mapsto \text{TRUE}]
\end{array}
$$

$$
\begin{array}{l}
\mathit{NextElevator}(\mathit{req}) \triangleq \\
\quad \land\ \mathit{elevator}.\mathit{available} = \text{TRUE} \\
\quad \land\ \mathit{elevator}' = \text{IF } \mathit{request}[\mathit{req}] = \text{TRUE} \\
\qquad\qquad\qquad\qquad \text{THEN } [\mathit{elevator} \text{ EXCEPT } !.\mathit{available} = \text{FALSE}, !.\mathit{requestedFloor} = \mathit{req}] \\
\qquad\qquad\qquad \text{ELSE } \mathit{elevator} \\
\quad \land\ \mathit{request}' = \text{IF } \mathit{request}[\mathit{req}] = \text{TRUE} \\
\qquad\qquad\qquad\qquad \text{THEN } [\mathit{request} \text{ EXCEPT } ![\mathit{req}] = \text{FALSE}] \\
\qquad\qquad\qquad \text{ELSE } \mathit{request}
\end{array}
$$

$$
\begin{array}{l}
\mathit{NextFloor} \triangleq \\
\quad \land\ \mathit{elevator}.\mathit{available} = \text{FALSE} \\
\quad \land\ \mathit{elevator}' = \text{IF } \mathit{elevator}.\mathit{floor} = \mathit{elevator}.\mathit{requestedFloor} \\
\qquad\qquad\qquad\qquad \text{THEN } [\mathit{elevator} \text{ EXCEPT } !.\mathit{available} = \text{TRUE}, !.\mathit{requestedFloor} = 0] \\
\qquad\qquad\qquad \text{ELSE IF } \mathit{elevator}.\mathit{floor} > \mathit{elevator}.\mathit{requestedFloor} \\
\qquad\qquad\qquad\qquad\qquad \text{THEN } [\mathit{elevator} \text{ EXCEPT } !.\mathit{floor} = (\mathit{elevator}.\mathit{floor} \% \mathit{TotalFloors}) - 1]
\end{array}
$$

$$
\begin{array}{l}
\qquad\qquad\quad \text{ELSE IF } elevator.floor < elevator.requestedFloor \\
\qquad\qquad\qquad\quad \text{THEN } [elevator \text{ EXCEPT } !.floor = (elevator.floor \% TotalFloors) + 1] \\
\qquad\qquad\quad \text{ELSE } elevator \\
\quad \land \text{UNCHANGED } request \\
\\
RequestMade \triangleq \\
\quad \forall\, req \in (1 \,..\, TotalFloors) : request[req] = \text{TRUE} \\
\\
NextRequest(req) \triangleq \\
\quad \land\ elevator.available = \text{TRUE} \\
\quad \land\ request' = \text{IF } RequestMade = \text{FALSE THEN } [request \text{ EXCEPT } ![req] = \text{TRUE}] \\
\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{ELSE } request \\
\quad \land \text{UNCHANGED } elevator \\
\\
Next \triangleq \\
\quad \land\ \lor \exists\, req \in (1 \,..\, TotalFloors) : NextElevator(req) \\
\qquad \lor NextFloor \\
\quad \land\ PrintT(elevator) \\
\\
Spec \triangleq \\
\quad Init \land \Box[Next]_{\langle elevator,\, request \rangle}
\end{array}
$$

THEOREM $Spec \Rightarrow TypeInvariant$

---

* Modification History
* Last modified *Fri* May 29 23:25:42 *PDT* 2015 by Me
* Created *Tue* May 26 16:13:01 *PDT* 2015 by Me