MODULE *crond2*

EXTENDS *Naturals*, *FiniteSets*

CONSTANTS *MAXAPROGS*, *PROGS*
ASSUME $MAXAPROGS \in Nat \land MAXAPROGS > 0$

VARIABLES *crontab*, *timers*, *now*, *aprocs*

$nullp \triangleq$ CHOOSE $x : x \notin PROGS$
$TypeInv \triangleq \land timers = [n \in Nat \to [t \to now,\ l \to 0,\ r \to \text{“no”}]]$
$\land IsFiniteSet(aprocs)$
$\land crontab \in [Nat \to [time : Nat,$
$prog : PROGS \cup nullp,$
$status : \{\text{“none”}, \text{“set”}, \text{“no”}, \text{“yes”}, \text{“run”}\}]]$
$\land now \in Nat$
$Ts \triangleq$ INSTANCE *Timers*
$av \triangleq \langle crontab, timers, aprocs, now\rangle$
$sv \triangleq \langle crontab, timers, aprocs\rangle$

$Init \triangleq Ts!TInit$

$AddJob(t, p) \triangleq$ LET $i =$ CHOOSE $w : w \in Nat \land crontab[w].status = \text{“none”}$
IN $\land p \in PROGS$
$\land t > 0$
$\land crontab' = [crontab$ EXCEPT
$![i] = [time \to t,\ prog \to p,\ status \to \text{“set”}]]$
$\land$ UNCHANGED $\langle now, aprocs\rangle$

$SetJob(i) \triangleq \land crontab[i].status = \text{“set”}$
$\land Ts!Set(i,\ crontab[i].time)$
$\land crontab' = [crontab$ EXCEPT
$![i] = [time \to @.t,\ prog \to @.prog,\ status \to \text{“no”}]]$
$\land$ UNCHANGED $\langle now, aprocs\rangle$

$Start(i) \triangleq \land crontab[i].status = \text{“no”}$
$\land Ts!Start(i)$
$\land crontab' = [crontab$ EXCEPT
$![i] = [time \to @.time,\ prog \to @.prog,\ status \to \text{“yes”}]]$
$\land$ UNCHANGED $\langle aprocs, now\rangle$

$Sched(i) \triangleq \land crontab[i].status = \text{“yes”}$
$\land Ts!Timeout(i)$
$\land crontab' = [crontab$ EXCEPT
$![i] = [time \to @.time,\ prog \to @.prog,\ status \to \text{“run”}]]$
$\land$ UNCHANGED $\langle aprocs, now\rangle$

$Exec(i) \triangleq \land crontab[i].status = \text{“run”}$

$$
\begin{array}{l}
\quad \land\ Cardinality(aprocs) < MAXAPROGS \\
\quad \land\ crontab' = [crontab \text{ EXCEPT} \\
\qquad\qquad\qquad ![i] = [time \to @.time,\ prog \to @.prog,\ status \to \text{“no”}]] \\
\quad \land\ aprocs' = aprocs \cup \{crontab[i].prog\} \\
\quad \land \text{UNCHANGED } \langle timers,\ now \rangle
\end{array}
$$

$$
\begin{array}{l}
RemoveJob(i) \triangleq\ \land\ \lor\ \land\ crontab[i].status = \text{“yes”} \\
\qquad\qquad\qquad\qquad\quad \land\ Ts\,!\,Stop(i) \\
\qquad\qquad\qquad\qquad\quad \land \text{UNCHANGED } \langle aprocs,\ now \rangle \\
\qquad\qquad\qquad\qquad \lor\ \land\ crontab[i].status = \text{“no”} \\
\qquad\qquad\qquad\qquad\quad \land \text{UNCHANGED } \langle aprocs,\ now,\ timers \rangle \\
\qquad\qquad\qquad\quad \land\ crontab' = [crontab \text{ EXCEPT} \\
\qquad\qquad\qquad\qquad\qquad\qquad ![i] = [time \to 0,\ prog \to nullp,\ status \to \text{“none”}]]
\end{array}
$$

$$
\begin{array}{l}
Next \triangleq\ \lor\ (\exists\, t \in Nat,\ p \in PROGS : AddJob(t,\ p)) \\
\qquad\quad \lor\ (\exists\, i \in Nat : \lor\ Start(i) \lor Sched(i) \lor Exec(i) \lor RemoveJob(i) \lor SetJob(i)) \\
Spec \triangleq\ Init \land \Box[Next]_{sv} \land (\forall\, i \in Nat : \mathrm{WF}_{av}(Start(i)) \land \mathrm{SF}_{av}(Exec(i)))
\end{array}
$$

THEOREM $Spec \Rightarrow \Box\, TypeInv$