MODULE *TLABarrier*

EXTENDS *Integers*

CONSTANT *threads*, *num_threads*

VARIABLE *TState*, *n*, *BState*

$$
\begin{aligned}
Init \triangleq\ & \land n = num\_threads \\
& \land TState = [t \in threads \mapsto \text{“pre”}] \\
& \land BState = \text{“init”}
\end{aligned}
$$

$$
\begin{aligned}
First(r) \triangleq\ & \land BState = \text{“init”} \\
& \land TState \in [threads \to \{\text{“pre”}\}] \\
& \land BState' = \text{“stop”} \\
& \land TState' = [TState \text{ EXCEPT } ![r] = \text{“at”}] \\
& \land \text{UNCHANGED } \langle n \rangle
\end{aligned}
$$

$$
\begin{aligned}
AdvanceTo(r) \triangleq\ & \land BState = \text{“stop”} \\
& \land TState[r] \neq \text{“at”} \\
& \land (\exists\, t \in threads : TState[t] \neq \text{“at”}) \\
& \land TState' = [TState \text{ EXCEPT } ![r] = \text{“at”}] \\
& \land \text{UNCHANGED } \langle n, BState \rangle
\end{aligned}
$$

$$
\begin{aligned}
AdvanceToLast(r) \triangleq\ & \land BState = \text{“stop”} \\
& \land TState[r] \neq \text{“at”} \\
& \land (\forall\, t \in threads : \lor t = r \\
& \qquad\qquad \lor TState[t] = \text{“at”}) \\
& \land TState' = [TState \text{ EXCEPT } ![r] = \text{“at”}] \\
& \land BState' = \text{“pass”} \\
& \land \text{UNCHANGED } \langle n \rangle
\end{aligned}
$$

$$
\begin{aligned}
AdvancePast(r) \triangleq\ & \land BState = \text{“pass”} \\
& \land TState' = [TState \text{ EXCEPT } ![r] = \text{“thru”}] \\
& \land \text{UNCHANGED } \langle n, BState \rangle
\end{aligned}
$$

$$
\begin{aligned}
Done \triangleq\ & \land TState \in [threads \to \{\text{“thru”}\}] \\
& \land \text{UNCHANGED } \langle TState, BState, n \rangle
\end{aligned}
$$

$$
\begin{aligned}
TNext(r) \triangleq\ & \lor First(r) \\
& \lor AdvanceTo(r) \\
& \lor AdvanceToLast(r) \\
& \lor AdvancePast(r)
\end{aligned}
$$

$$
\begin{aligned}
TypeOK \triangleq\ & \land TState \in [threads \to \{\text{“pre”}, \text{“at”}, \text{“thru”}\}] \\
& \land BState \in \{\text{“init”}, \text{“stop”}, \text{“pass”}\}
\end{aligned}
$$

$$
\begin{aligned}
BarrierNext \triangleq\ & \lor Done \\
& \lor (\exists\, t \in threads : TNext(t))
\end{aligned}
$$

$$Spec \triangleq Init \wedge \Box[BarrierNext]_{\langle TState,\, n,\, BState \rangle}$$

* Modification History
* Last modified *Tue Mar* 19 16:12:18 *PDT* 2019 by *JAlbers*