MODULE *counter*

This spec specefies the logic that is used to implement atomix counter primitive

EXTENDS *Integers*, *TLC*, *Naturals*

counter variable

VARIABLE *counter*

counter value before change

VARIABLE *preValue*

counter value after change

VARIABLE *nextValue*

counter change value

CONSTANT *Delta*

list of all possible values for a counter

CONSTANT *Value*

list of variables

$vars \triangleq \langle counter, preValue, nextValue \rangle$

$TypeInvariant \triangleq counter \in Int$

initialize variables

$$
\begin{aligned}
&Init \triangleq \\
&\quad \land counter = 0 \\
&\quad \land preValue = 0 \\
&\quad \land nextValue = 0
\end{aligned}
$$

set counter variable

$$
\begin{aligned}
&Set(val) \triangleq \\
&\quad \land counter' = val \\
&\quad \land \text{UNCHANGED } \langle preValue, nextValue \rangle
\end{aligned}
$$

increment counter variable

$$
\begin{aligned}
&Increment(delta) \triangleq \\
&\quad \land preValue' = counter \\
&\quad \land counter' = counter + delta \\
&\quad \land nextValue' = counter'
\end{aligned}
$$

decrement counter variable

$$
\begin{aligned}
&Decrement(delta) \triangleq \\
&\quad \land preValue' = counter \\
&\quad \land counter' = counter - delta \\
&\quad \land nextValue' = counter'
\end{aligned}
$$

next state

$$Next \triangleq$$
$$\lor \exists\, d \in Delta :$$
$$\quad Increment(d)$$
$$\lor \exists\, d \in Delta :$$
$$\quad Decrement(d)$$
$$\lor \exists\, v \in Value :$$
$$\quad Set(v)$$

$$Spec \triangleq Init \land \Box[Next]_{\langle vars \rangle}$$

* Modification History
* Last modified *Thu Feb* 13 19:17:59 *PST* 2020 by *adibrastegarnia*
* Created *Wed Feb* 05 20:49:41 *PST* 2020 by *adibrastegarnia*