MODULE *SendInt2*

The example in Section 4.1 of the paper "Auxiliary Variables in TLA+" comprises this module and modules *SendInt*1 and *SendInt*1*P*.

The example explains the basic idea behind prophecy variables with a simple prophecy variable that can make at most a single prediction at a time. It consists of a trivial system in which a sender sends arbitrary integers to a receiver, where sending an integer v means setting the variable x to v, and receiving the integer means setting x to the non-integer value *NotInt*. This spec also contains a variable z that is initially equal to the first value to be sent and is set by the receive action to the value of the next integer to be sent.

EXTENDS *Integers*

This defines *NotInt* to be some particular constant value that is not an integer. The semantics of TLA+ do not determine what that particular value is, just that it isn't an integer. It is the same value for every possible behavior satisfying the spec. By default, when creating a model, *TLC* substitutes a model value of the same name for *NotInt*. (The definition has to have a particular syntactic form for it to do this.)

$NotInt \triangleq \text{CHOOSE } n : n \notin Int$

VARIABLE x, z

In general, a spec should define a formula that asserts type correctness of the variables. This helps the reader understand the spec, and you can catch simple "type" errors easily by having *TLC* check that the formula is an invariant. To save space, this is not done in the versions of the specifications in the paper "Auxiliary Variables in TLA+".

$$TypeOK \triangleq \begin{array}{l} \land\ x \in Int \cup \{NotInt\} \\ \land\ z \in Int \cup \{NotInt\} \end{array}$$

$$Init \triangleq \begin{array}{l} \land\ x = NotInt \\ \land\ z \in Int \end{array}$$

$$Send \triangleq \begin{array}{l} \land\ x = NotInt \\ \land\ x' = z \\ \land\ z' = NotInt \end{array}$$

$$Rcv \triangleq \begin{array}{l} \land\ x \in Int \\ \land\ x' = NotInt \\ \land\ z' \in Int \end{array}$$

$$Next \triangleq Send \lor Rcv$$

$$Spec \triangleq Init \land \Box[Next]_{\langle x,\, z\rangle}$$

* Modification History
* Last modified *Wed Oct* 19 02:48:18 *PDT* 2016 by *lamport*
* Created *Tue Sep* 06 02:16:47 *PDT* 2016 by *lamport*