─────────────────── MODULE *Channel* ───────────────────

EXTENDS *Naturals*
CONSTANT *Data*
VARIABLE *chan*

$TypeInvariant \triangleq chan \in [val : Data, rdy : \{0, 1\}, ack : \{0, 1\}]$

───

$Init \triangleq \wedge TypeInvariant$
$\qquad\quad \wedge chan.ack = chan.rdy$

$Send(d) \triangleq \wedge chan.rdy = chan.ack$
$\qquad\qquad\quad \wedge chan' = [chan \text{ EXCEPT } !.val = d, !.rdy = 1 - @]$

$Rcv \triangleq \wedge chan.rdy \neq chan.ack$
$\qquad\quad \wedge chan' = [chan \text{ EXCEPT } !.ack = 1 - @]$

$Next \triangleq (\exists\, d \in Data : Send(d)) \vee Rcv$
$Spec \triangleq Init \wedge \Box[Next]_{chan}$

───

THEOREM $Spec \Rightarrow \Box\, TypeInvariant$

───