```
─────────────────── MODULE GreyCodeCounter ───────────────────
EXTENDS Sequences, Integers, TLC

  Grey Code is a way of encoding binary information such that no two bits flip in one step.
 This is used in Flash Memory to minimize bit flips during self-discharge
  --algorithm GreyCode{

variables
     flashCell ∈ {⟨0, 0⟩, ⟨0, 1⟩, ⟨1, 1⟩, ⟨1, 0⟩} ;
     actions = [before ↦ ⟨⟩, after ↦ ⟨⟩] ;
{

while ( TRUE ) {
     actions.before := flashCell ;
     if ( flashCell = ⟨0, 0⟩ ) {

           flashCell := ⟨0, 1⟩ ;
           actions.after := ⟨0, 1⟩ ;

      } else if ( flashCell = ⟨0, 1⟩ ) {
            flashCell := ⟨1, 1⟩ ;
            actions.after := ⟨1, 1⟩ ;

      } else if ( flashCell = ⟨1, 0⟩ ) {

            flashCell := ⟨0, 0⟩ ;
            actions.after := ⟨0, 0⟩ ;

      } else {

            flashCell := ⟨1, 0⟩ ;
            actions.after := ⟨1, 0⟩ ;

      }

 }
 }
 }
  end algorithm;
  BEGIN TRANSLATION (chksum(pcal) = "f604e792" ∧ chksum(tla) = "ce051695")
VARIABLES flashCell, actions, pc

vars ≜ ⟨flashCell, actions, pc⟩

Init ≜  Global variables
          ∧ flashCell ∈ {⟨0, 0⟩, ⟨0, 1⟩, ⟨1, 1⟩, ⟨1, 0⟩}
          ∧ actions = [before ↦ ⟨⟩, after ↦ ⟨⟩]
          ∧ pc = "Lbl_1"
```

```
$Lbl\_1 \triangleq \land pc = \text{"Lbl\_1"}$
           $\land actions' = [actions \text{ EXCEPT } !.before = flashCell]$
           $\land \text{IF } flashCell = \langle 0, 0\rangle$
                  $\text{THEN } \land flashCell' = \langle 0, 1\rangle$
                         $\land pc' = \text{"Lbl\_2"}$
                  $\text{ELSE } \land \text{IF } flashCell = \langle 0, 1\rangle$
                                $\text{THEN } \land flashCell' = \langle 1, 1\rangle$
                                       $\land pc' = \text{"Lbl\_3"}$
                                $\text{ELSE } \land \text{IF } flashCell = \langle 1, 0\rangle$
                                              $\text{THEN } \land flashCell' = \langle 0, 0\rangle$
                                                     $\land pc' = \text{"Lbl\_4"}$
                                              $\text{ELSE } \land flashCell' = \langle 1, 0\rangle$
                                                     $\land pc' = \text{"Lbl\_5"}$

$Lbl\_2 \triangleq \land pc = \text{"Lbl\_2"}$
           $\land actions' = [actions \text{ EXCEPT } !.after = \langle 0, 1\rangle]$
           $\land pc' = \text{"Lbl\_1"}$
           $\land \text{UNCHANGED } flashCell$

$Lbl\_3 \triangleq \land pc = \text{"Lbl\_3"}$
           $\land actions' = [actions \text{ EXCEPT } !.after = \langle 1, 1\rangle]$
           $\land pc' = \text{"Lbl\_1"}$
           $\land \text{UNCHANGED } flashCell$

$Lbl\_4 \triangleq \land pc = \text{"Lbl\_4"}$
           $\land actions' = [actions \text{ EXCEPT } !.after = \langle 0, 0\rangle]$
           $\land pc' = \text{"Lbl\_1"}$
           $\land \text{UNCHANGED } flashCell$

$Lbl\_5 \triangleq \land pc = \text{"Lbl\_5"}$
           $\land actions' = [actions \text{ EXCEPT } !.after = \langle 1, 0\rangle]$
           $\land pc' = \text{"Lbl\_1"}$
           $\land \text{UNCHANGED } flashCell$

$Next \triangleq Lbl\_1 \lor Lbl\_2 \lor Lbl\_3 \lor Lbl\_4 \lor Lbl\_5$

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

$\Box[NextVariables]$ "OR" $\_unchanged\_vars$

END TRANSLATION

$TypeOk \triangleq flashCell[1] \in \{0, 1\} \land flashCell[2] \in \{0, 1\}$ They are 1 indexed!

Only 1 bit can change between two states. $\langle 0, 0\rangle \rightarrow \langle 1, 1\rangle$ is an illegal transition. Hence, "distance" between any two states is $|\{0, 1\}|$

$OneBitAtATime \triangleq \text{IF } \; actions.before \neq \langle\rangle \land actions.after \neq \langle\rangle$

$\textsc{then}$

$\quad (\mathit{actions}.\mathit{after}[1] + \mathit{actions}.\mathit{after}[2]) - (\mathit{actions}.\mathit{before}[1] + \mathit{actions}.\mathit{before}[2]) \ \in \{-1, 0, 1\}$

$\textsc{else} \ \ \textsc{true}$

$$\mathit{OurFriendship} \ \triangleq \ \mathit{HackingDay} \Rightarrow \Box(\mathit{Friends})$$

$$\mathit{OneBitAtATime} \ \triangleq \ (\mathit{flashCell}'[1] + \mathit{flashCell}'[2] - (\mathit{flashCell}[1] + \mathit{flashCell}[2])) \ \in \{-1, 0, 1\}$$

$$\mathit{AlwaysOneBitAtATime} \ \triangleq \ \Box([\mathit{OneBitAtATime}]\_\mathit{flashCell})$$

\* Modification History

\* Last modified *Fri Mar* 05 10:26:11 *MST* 2021 by *jeremy*

\* Created *Tue Mar* 02 16:05:58 *MST* 2021 by *jeremy*