```
 yZero :

  |-------
{1}   is_P_reachable_valid(G(assertion_to_TP[STATE[N]](yZero)), pfs)

Rule? (BINV "yZero")
```

```
 ;;;The inductive step of the BINV rule
{-1,(rho)}
    loc(current!1)(p!1) = 0 AND y(next!1) = y(current!1) AND
    loc(next!1) = loc(current!1) WITH [(p!1) := 1]
 OR loc(current!1)(p!1) = 1 AND
    (EXISTS (m: nat): (FORALL (q: PROC_ID):
      y(current!1)(q) < m) AND y(next!1) = y(current!1) WITH [(p!1) := m])
      AND loc(next!1) = loc(current!1) WITH [(p!1) := 2]
 OR loc(current!1)(p!1) = 2 AND
    (FORALL q: q /= p!1 IMPLIES
      y(current!1)(q) = 0 OR y(current!1)(p!1) <= y(current!1)(q)) AND
    y(next!1) = y(current!1) AND
    loc(next!1) = loc(current!1) WITH [(p!1) := 3]
 OR loc(current!1)(p!1) = 3 AND y(next!1) = y(current!1) AND
    loc(next!1) = loc(current!1) WITH [(p!1) := 4]
 OR loc(current!1)(p!1) = 4 AND
    y(next!1) = y(current!1) WITH [(p!1) := 0] AND
    loc(next!1) = loc(current!1) WITH [(p!1) := 0]
{-2,(yZero invariant)} FORALL (i: PROC_ID):
    (y(current!1)(i) = 0 IFF (loc(current!1)(i) = 0 OR loc(current!1)(i) = 1))
  |-------
{1,(rtp)} (y(next!1)(i!1) = 0 IFF (loc(next!1)(i!1) = 0 OR loc(next!1)(i!1) = 1))
Rule? (split-rho)
```

```
 this yields  5 subgoals:
yZero.1 :
{-1,(rho)}
      loc(current!1)(p!1) = 0
{-2,(rho)}
      y(next!1) = y(current!1)
{-3,(rho)}
      loc(next!1) = loc(current!1) WITH [(p!1) := 1]
{-4,(yZero invariant)}
      FORALL (i: PROC_ID):
        IF y(current!1)(i) = 0
          THEN (loc(current!1)(i) = 0 OR loc(current!1)(i) = 1)
        ELSE NOT (loc(current!1)(i) = 0 OR loc(current!1)(i) = 1)
        ENDIF
  |-------
{1,(rtp)}
      IF y(current!1)(i!1) = 0
        THEN (loc(current!1) WITH [(p!1) := 1](i!1) = 0 OR
               loc(current!1) WITH [(p!1) := 1](i!1) = 1)
      ELSE NOT (loc(current!1) WITH [(p!1) := 1](i!1) = 0 OR
                 loc(current!1) WITH [(p!1) := 1](i!1) = 1)
      ENDIF

Rule?  (inst - "i!1")




 [-1,(rho)]
      loc(current!1)(p!1) = 0
[-2,(rho)]
      y(next!1) = y(current!1)
[-3,(rho)]
      loc(next!1) = loc(current!1) WITH [(p!1) := 1]
{-4,(yZero invariant)}
      IF y(current!1)(i!1) = 0
        THEN (loc(current!1)(i!1) = 0 OR loc(current!1)(i!1) = 1)
      ELSE NOT (loc(current!1)(i!1) = 0 OR loc(current!1)(i!1) = 1)
      ENDIF
  |-------
[1,(rtp)]
      IF y(current!1)(i!1) = 0
        THEN (loc(current!1) WITH [(p!1) := 1](i!1) = 0 OR
               loc(current!1) WITH [(p!1) := 1](i!1) = 1)
      ELSE NOT (loc(current!1) WITH [(p!1) := 1](i!1) = 0 OR
                 loc(current!1) WITH [(p!1) := 1](i!1) = 1)
      ENDIF

Rule? (split-all)
Split-all if-then-else consequents,

This completes the proof of yZero.1.
```

```
 yZero.2 :
{-1,(rho)}
      loc(current!1)(p!1) = 1
{-2,(rho)}
      FORALL (q: PROC_ID[5, N]): y(current!1)(q) < m!1
{-3,(rho)}
      y(next!1) = y(current!1) WITH [(p!1) := m!1]
{-4,(rho)}
      loc(next!1) = loc(current!1) WITH [(p!1) := 2]
{-5,(yZero invariant)}
      FORALL (i: PROC_ID):
        IF y(current!1)(i) = 0
          THEN (loc(current!1)(i) = 0 OR loc(current!1)(i) = 1)
        ELSE NOT (loc(current!1)(i) = 0 OR loc(current!1)(i) = 1)
        ENDIF
  |-------
{1,(rtp)}
      IF y(current!1) WITH [(p!1) := m!1](i!1) = 0
        THEN (loc(current!1) WITH [(p!1) := 2](i!1) = 0 OR
               loc(current!1) WITH [(p!1) := 2](i!1) = 1)
      ELSE NOT (loc(current!1) WITH [(p!1) := 2](i!1) = 0 OR
                 loc(current!1) WITH [(p!1) := 2](i!1) = 1)
      ENDIF
Rule? (split-all-inst ("i!1"))
This completes the proof of yZero.2.
```

```
 ;;;The inductive step of the BINV rule
{-1,(rho)}
    loc(current!1)(p!1) = 0 AND y(next!1) = y(current!1) AND
    loc(next!1) = loc(current!1) WITH [(p!1) := 1]
 OR loc(current!1)(p!1) = 1 AND
    (EXISTS (m: nat): (FORALL (q: PROC_ID):
      y(current!1)(q) < m) AND y(next!1) = y(current!1) WITH [(p!1) := m])
      AND loc(next!1) = loc(current!1) WITH [(p!1) := 2]
 OR loc(current!1)(p!1) = 2 AND
    (FORALL q: q /= p!1 IMPLIES
      y(current!1)(q) = 0 OR y(current!1)(p!1) <= y(current!1)(q)) AND
    y(next!1) = y(current!1) AND
    loc(next!1) = loc(current!1) WITH [(p!1) := 3]
 OR loc(current!1)(p!1) = 3 AND y(next!1) = y(current!1) AND
    loc(next!1) = loc(current!1) WITH [(p!1) := 4]
 OR loc(current!1)(p!1) = 4 AND
    y(next!1) = y(current!1) WITH [(p!1) := 0] AND
    loc(next!1) = loc(current!1) WITH [(p!1) := 0]
{-2,(yZero invariant)} FORALL (i: PROC_ID):
    (y(current!1)(i) = 0 IFF (loc(current!1)(i) = 0 OR loc(current!1)(i) = 1))
  |-------
{1,(rtp)} (y(next!1)(i!1) = 0 IFF (loc(next!1)(i!1) = 0 OR loc(next!1)(i!1) = 1))
Rule? (split-rho-all ("i!1"))
Q.E.D.
```