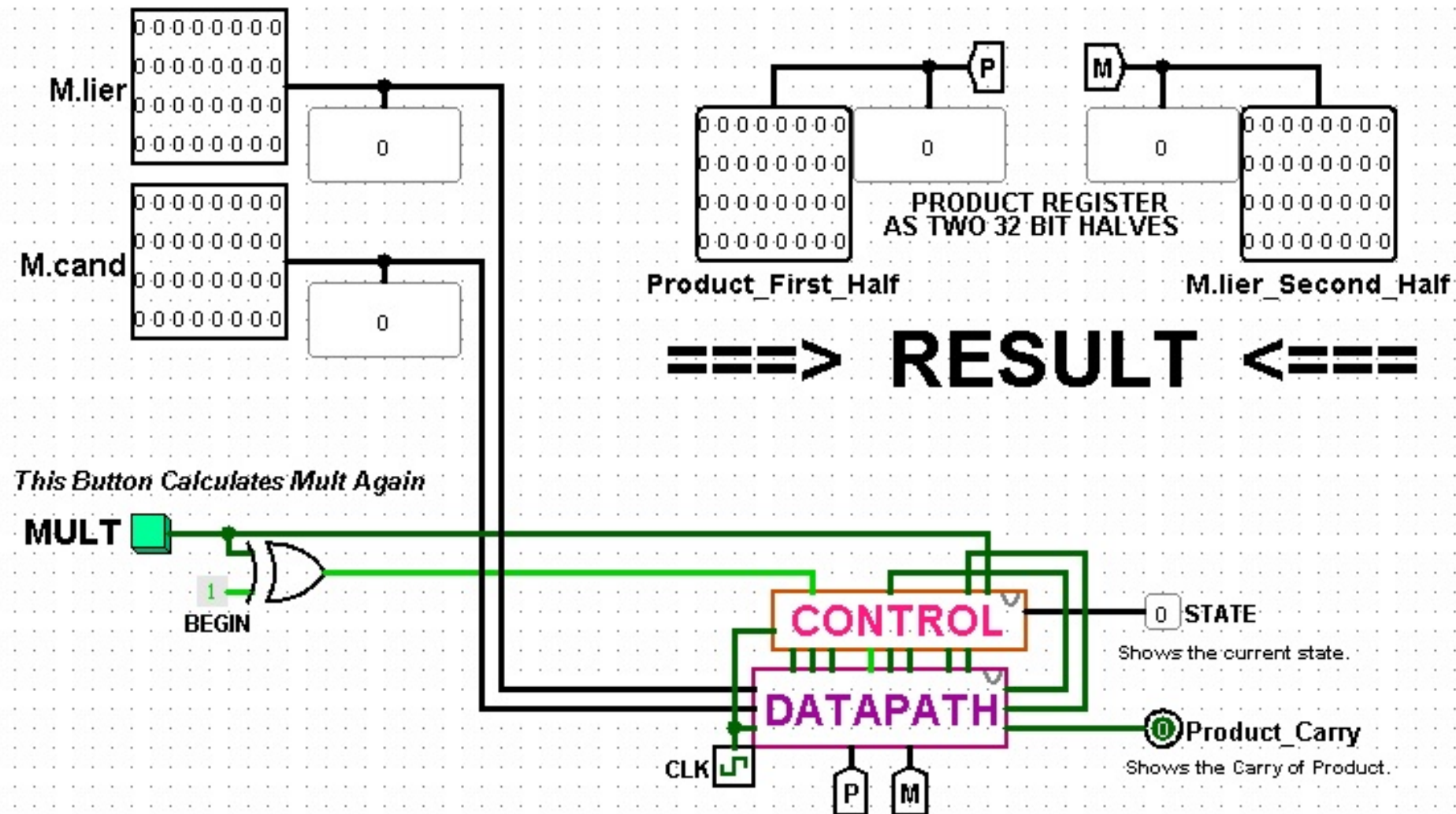


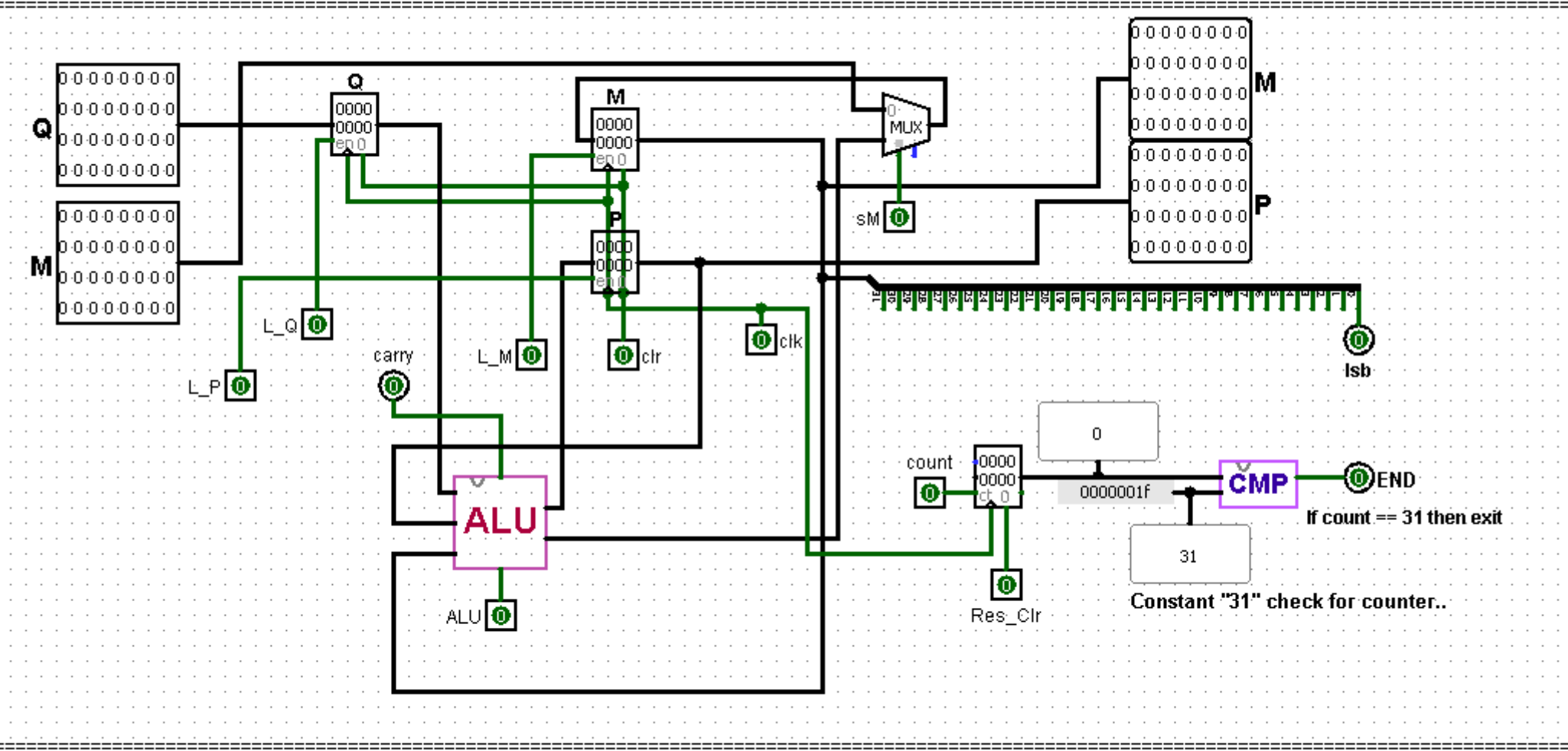
# MULT32 DESIGN

==> PRESS MULT AFTER ENTERING THE MULTIPLICAND-MULTIPLIER NUMBERS! <==

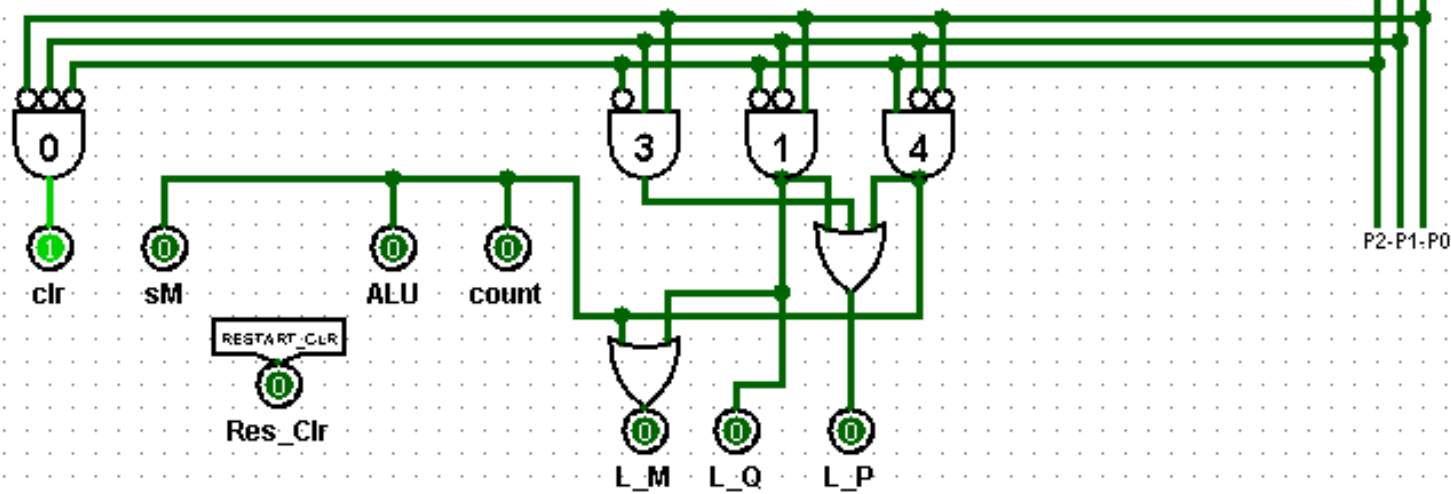
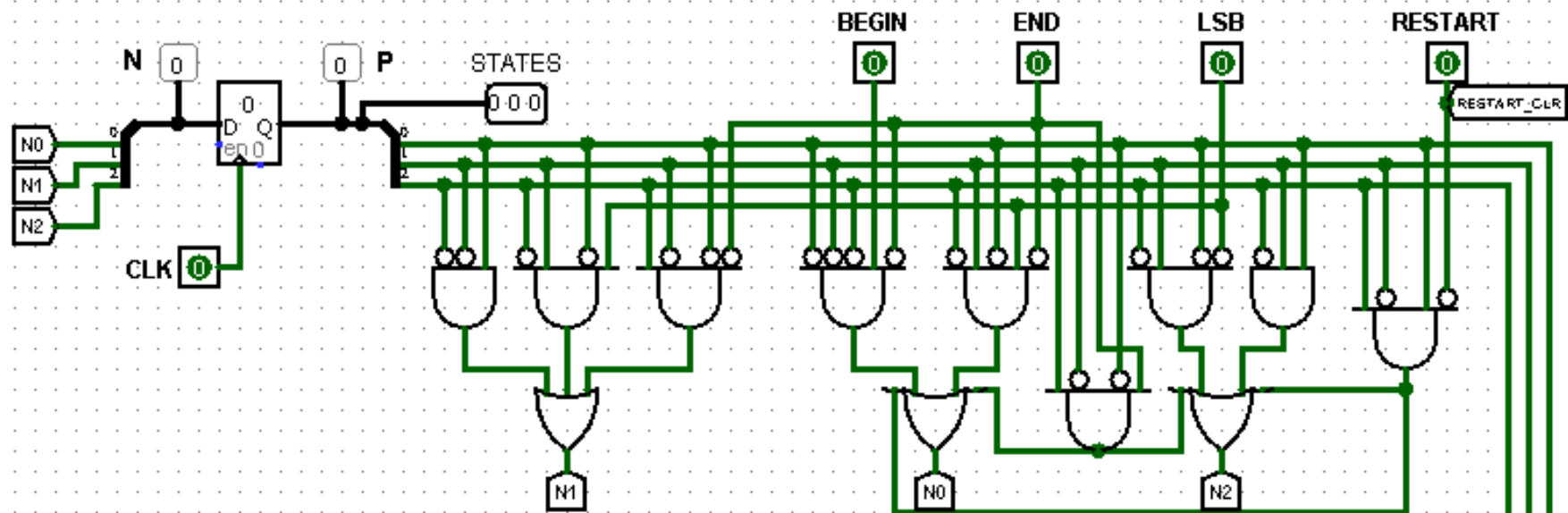


BEGIN = Constant 1 ==> provides used to process multiple time

# DATAPATH DESIGN

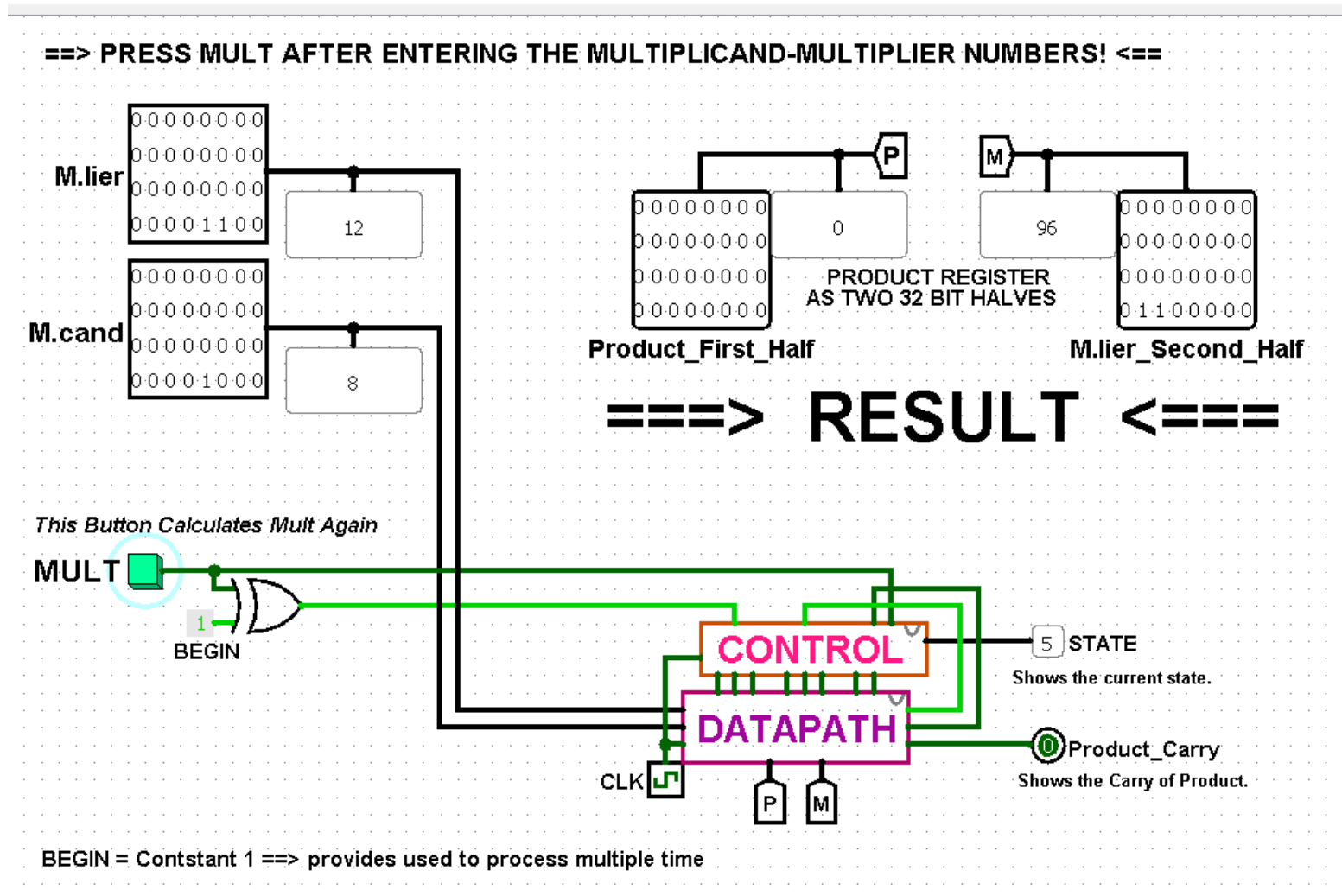


# CONTROL UNIT DESIGN



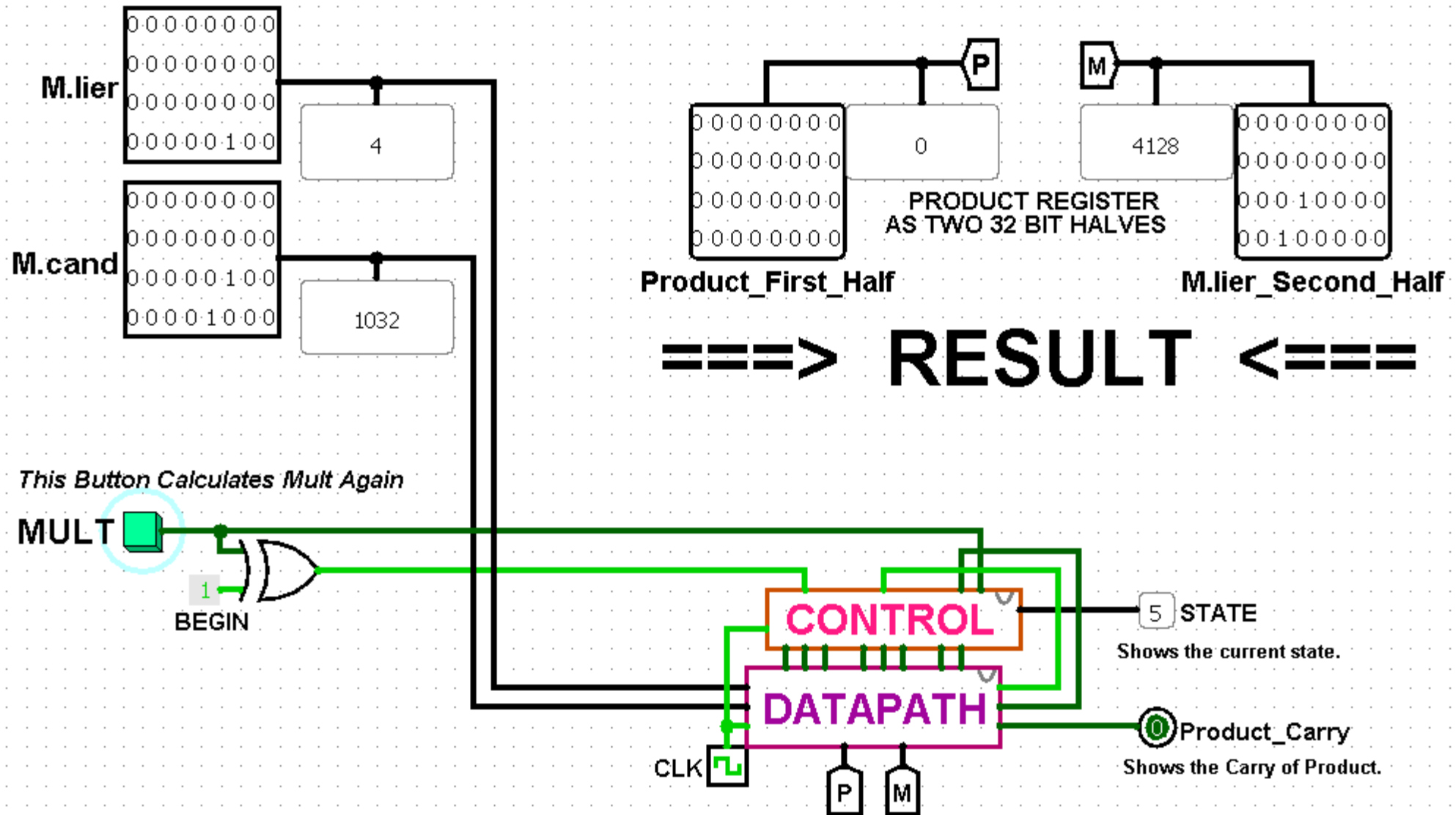
- The Followings are the screenshots(6) of Multiplication process on logisim.
- (from mult32.circ)

### TEST-1 ➡



## TEST-2 ➡

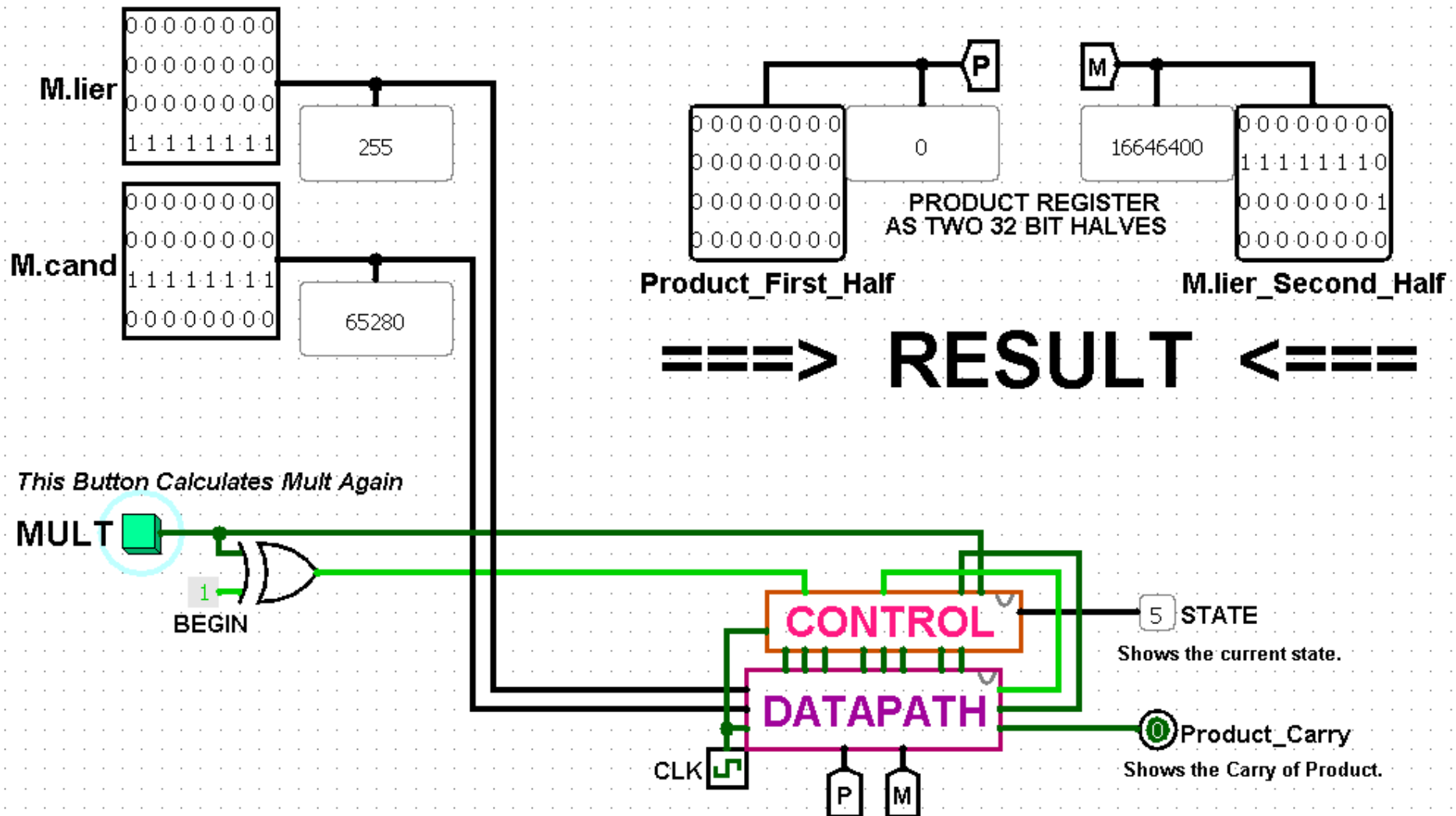
==> PRESS MULT AFTER ENTERING THE MULTIPLICAND-MULTIPLIER NUMBERS! <==



BEGIN = Constant 1 ==> provides used to process multiple time

## TEST-3 ➡

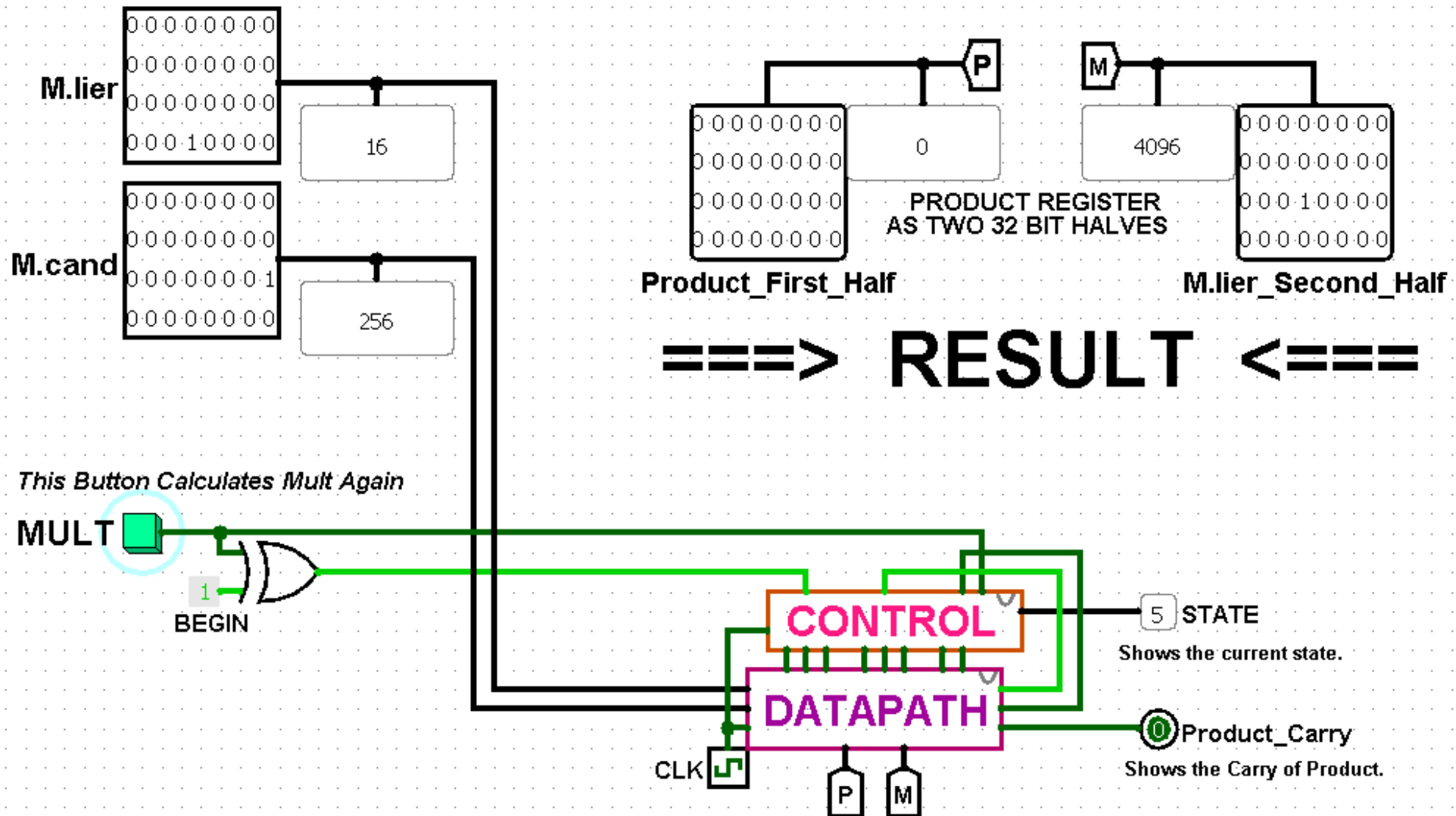
==> PRESS MULT AFTER ENTERING THE MULTIPLICAND-MULTIPLIER NUMBERS! <==



BEGIN = Constant 1 ==> provides used to process multiple time

## TEST-4 ➡

==> PRESS MULT AFTER ENTERING THE MULTIPLICAND-MULTIPLIER NUMBERS! <==

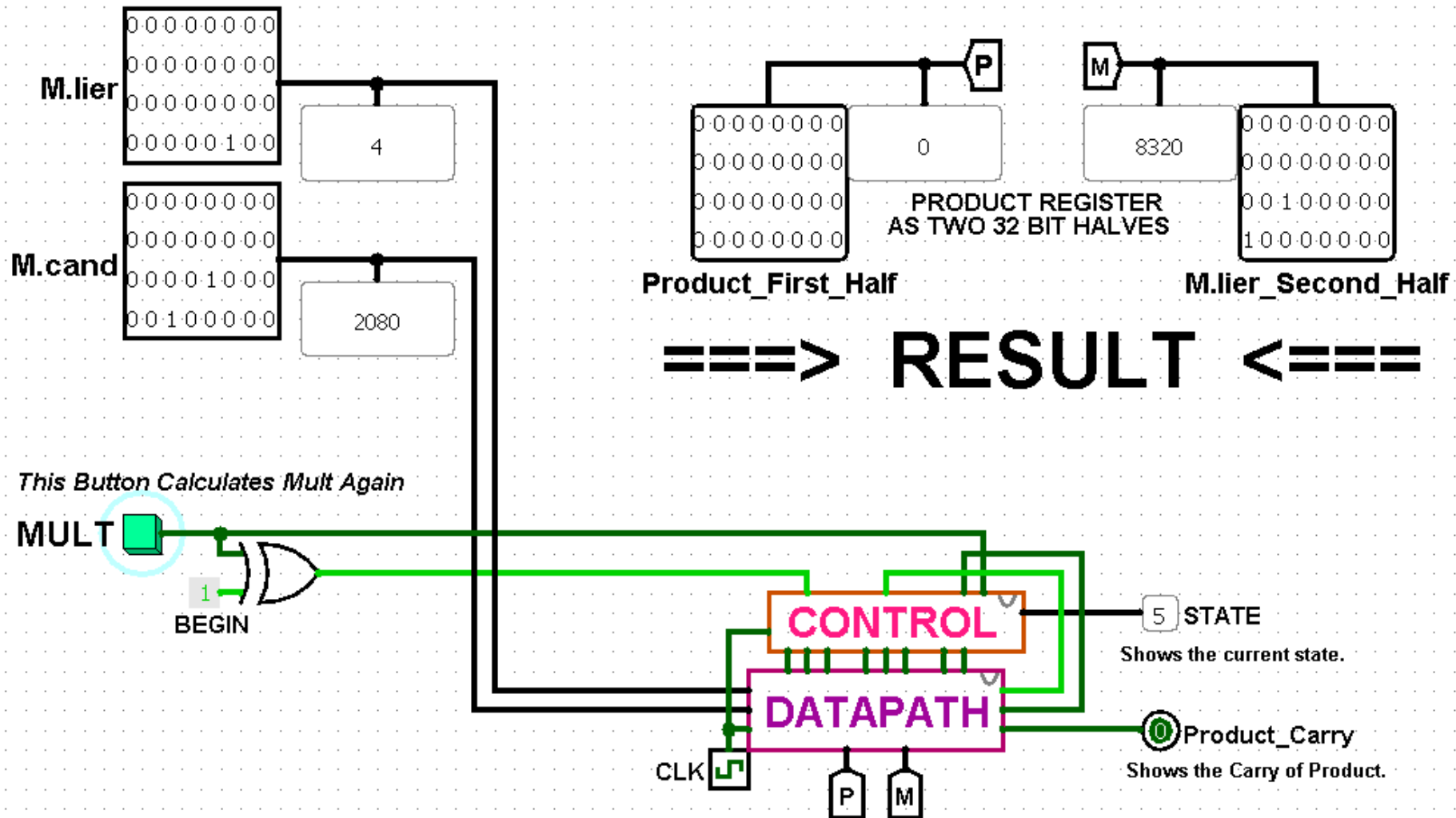


==> **RESULT** <==

BEGIN = Constant 1 ==> provides used to process multiple time

## TEST-5 ➡

==> PRESS MULT AFTER ENTERING THE MULTIPLICAND-MULTIPLIER NUMBERS! <==

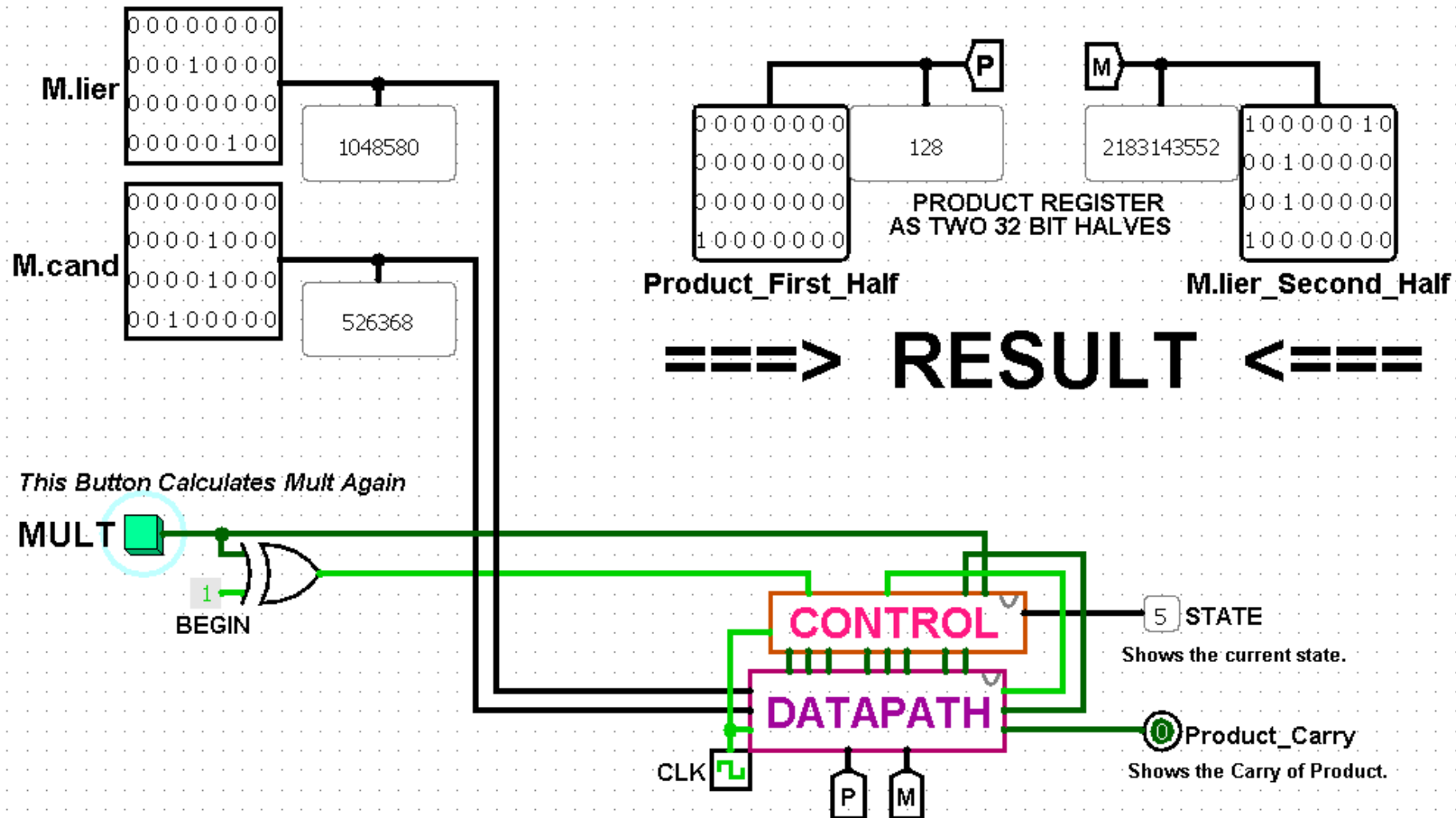


BEGIN = Constant 1 ==> provides used to process multiple time



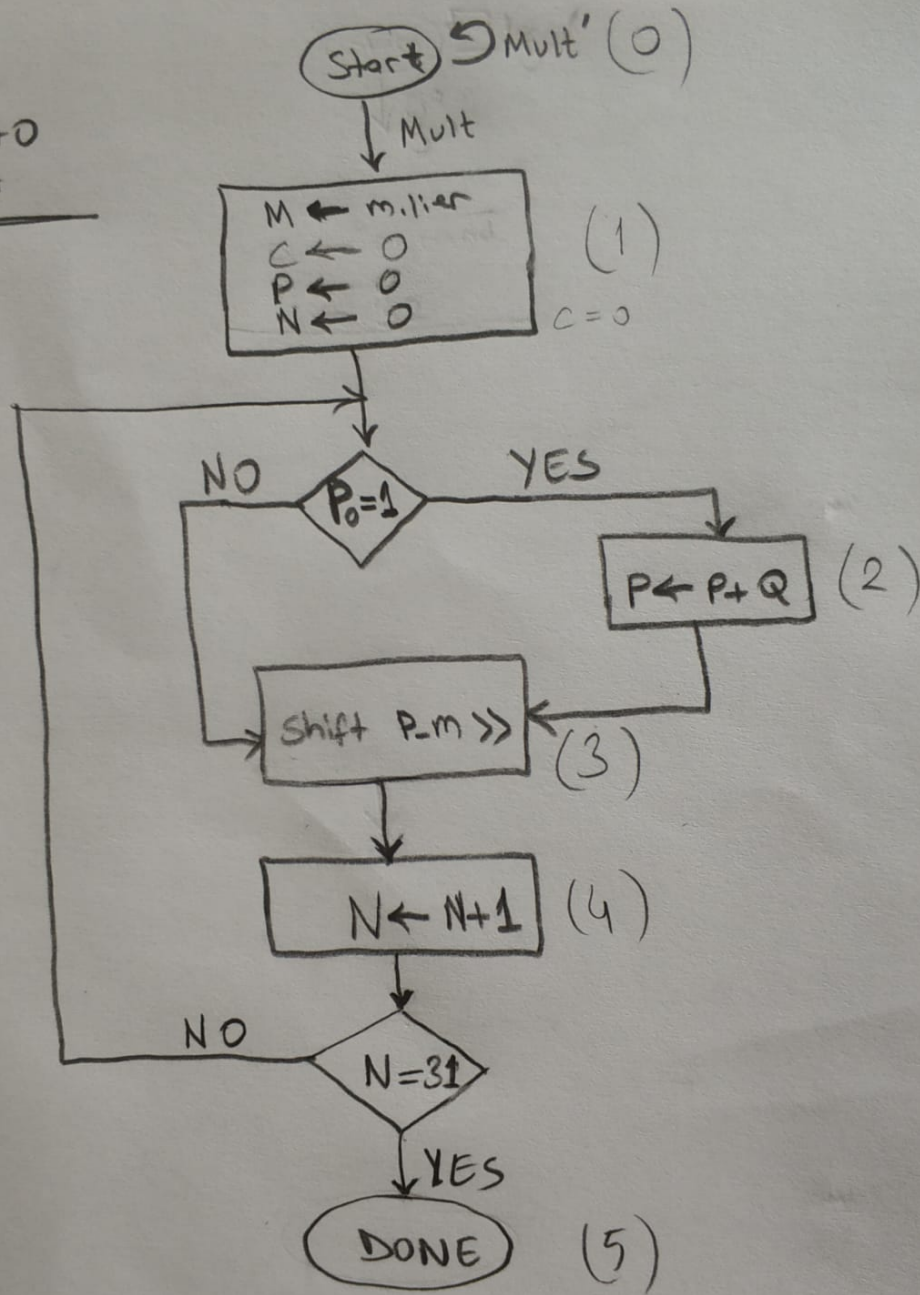
## TEST-6 ➔

==> PRESS MULT AFTER ENTERING THE MULTIPLICAND-MULTIPLIER NUMBERS! <==

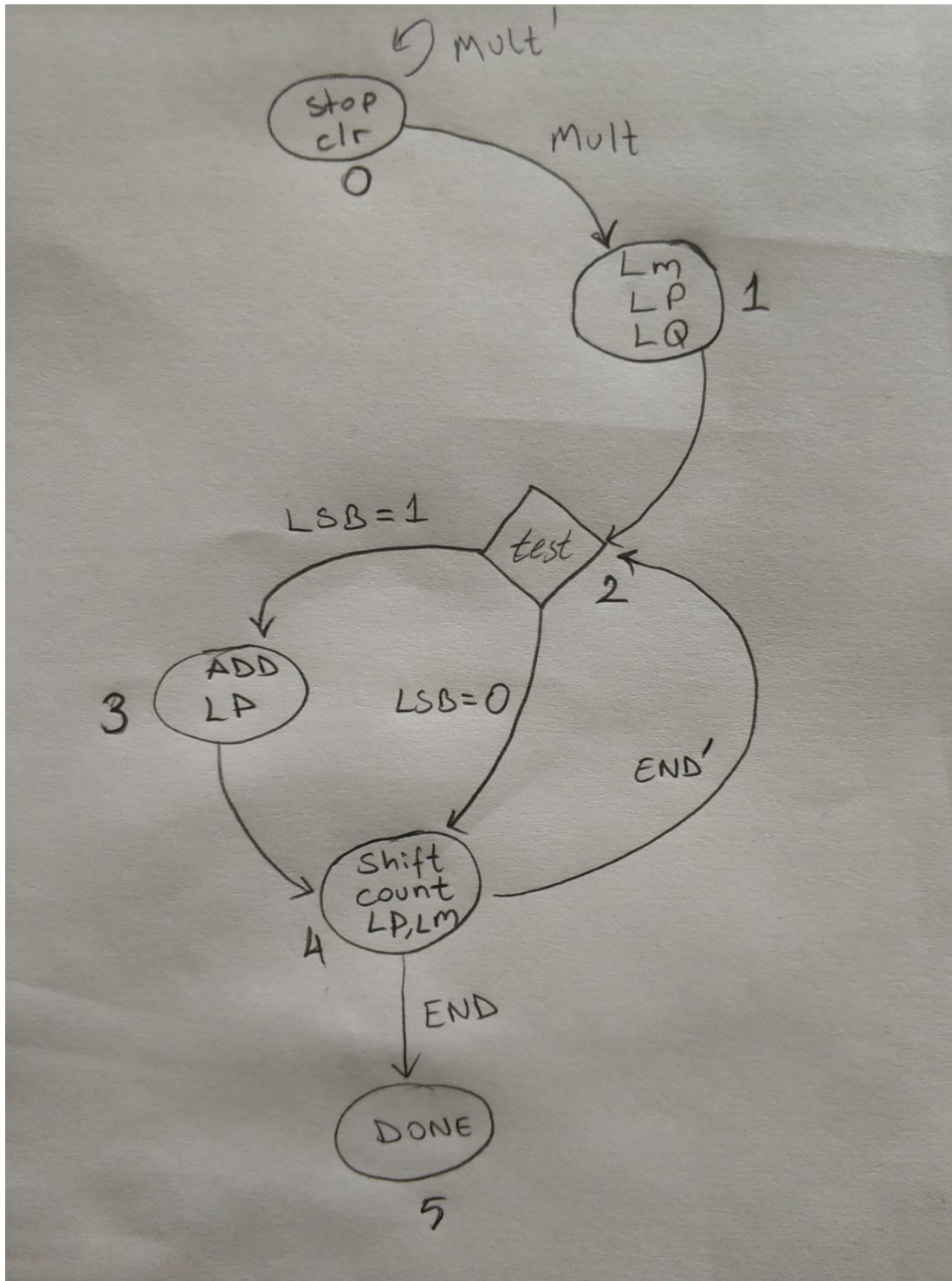


BEGIN = Constant 1 ==> provides used to process multiple time

M = multiplier  
Q = multiplicand  
P = Product  
LSB = Product + 0  
N = 32 bit



This is the first state diagram from ASM in PDF.



This is the second state diagram from after create the Datapath. This diagram has the datapath and control unit signals.

## STATE TABLE DESIGN

**L\_P** → Load Product Register

**L\_M** → Load Multiplier Register

**L\_Q** → Load Multiplicand Register

**clr** → Clear all registers

**sM** → Select Multiplier Reg MUX

**Res\_Clr** → Resets the counter.

**ALU** → ALU operation Add-Shift

**Count** → count signal for counter

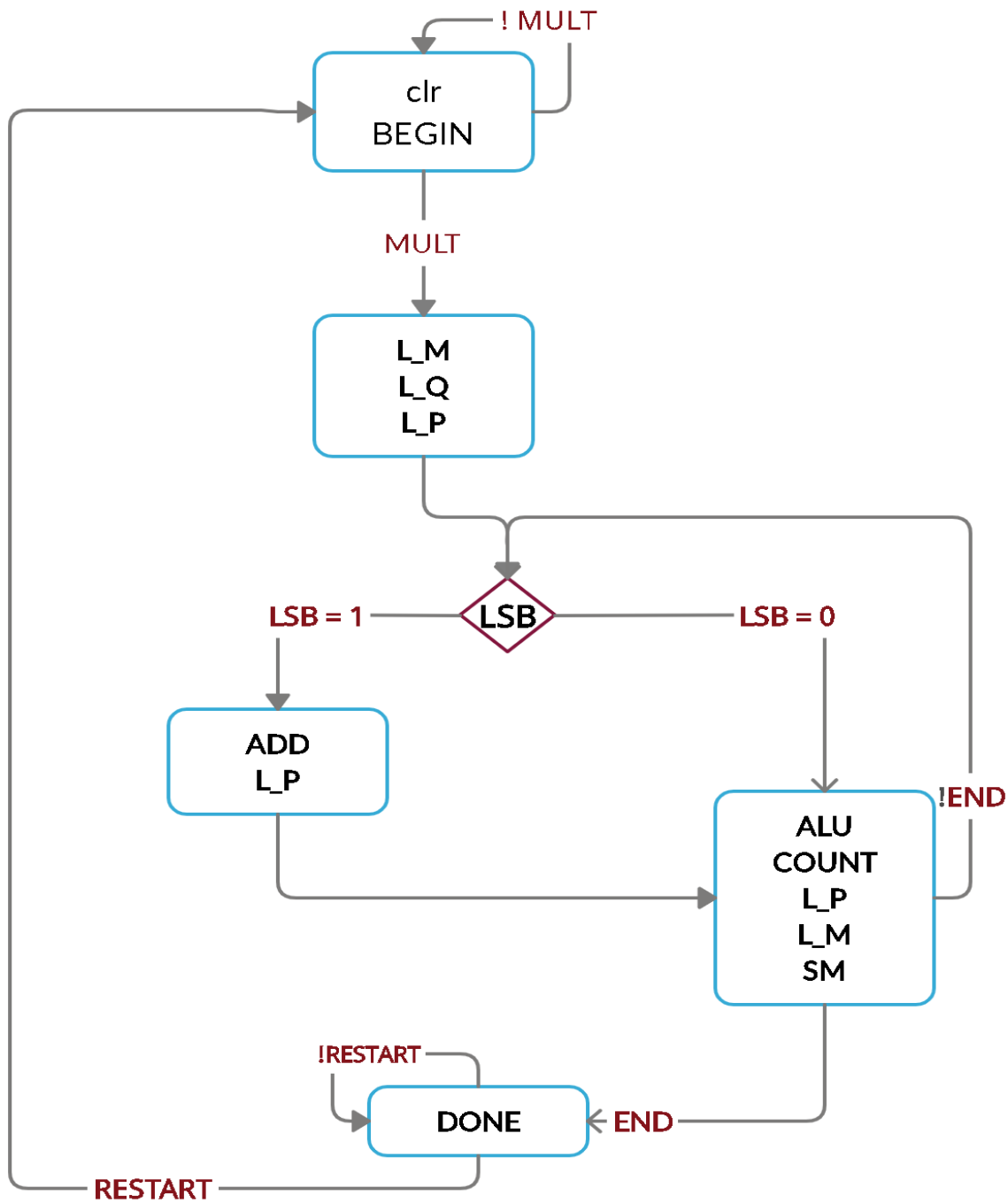
**LSB** → Least Significant Bit of Product-M.lier Reg.

**RESTART** → Controlled by MULT button to process

**END** → Check END with counter component in Datapath

**BEGIN** → Const 1: controlled by XOR with MULT Button

This is the last diagram with signals.



# STATE TABLE

*PRESENT*                      *INPUTS*                      *NEXT*

| P2 | P1 | P0 | START | RESTART | END | LSB | N2 | N1 | N0 |
|----|----|----|-------|---------|-----|-----|----|----|----|
| 0  | 0  | 0  | 0     | -       | 0   | -   | 0  | 0  | 0  |
| 0  | 0  | 0  | 1     | -       | 0   | -   | 0  | 0  | 1  |
| 0  | 0  | 1  | -     | -       | -   | -   | 0  | 1  | 0  |
| 0  | 1  | 0  | -     | -       | -   | 1   | 0  | 1  | 1  |
| 0  | 1  | 0  | -     | -       | -   | 0   | 1  | 0  | 0  |
| 0  | 1  | 1  | -     | -       | -   | -   | 1  | 0  | 0  |
| 1  | 0  | 0  | -     | -       | 0   | -   | 0  | 1  | 0  |
| 1  | 0  | 0  | -     | -       | 1   | -   | 1  | 0  | 1  |
| 1  | 0  | 1  | -     | 0       | -   | -   | 1  | 0  | 1  |



$$N_0 = \bar{P}_2 \bar{P}_1 \bar{P}_0 \cdot \text{BEGIN} \cdot \overline{\text{END}} + \bar{P}_2 P_1 \bar{P}_0 \cdot \text{LSB} \cdot \overline{\text{END}} + P_2 \bar{P}_1 \bar{P}_0 \cdot \overline{\text{END}}$$

$$N_1 = \bar{P}_2 \bar{P}_1 P_0 + \bar{P}_2 P_1 \bar{P}_0 \cdot \text{LSB} + P_2 \bar{P}_1 \bar{P}_0 \cdot \overline{\text{END}}$$

$$N_2 = \bar{P}_2 P_1 \bar{P}_0 \cdot \text{LSB} + \bar{P}_2 P_1 P_0 + P_2 \bar{P}_1 \bar{P}_0 \cdot \overline{\text{END}}$$



## OUTPUTS

| <u>states</u> | <u>clr</u> | <u>SM</u> | <u>L-P</u> | <u>L-m</u> | <u>L-Q</u> | <u>ALU</u> | <u>count</u> |
|---------------|------------|-----------|------------|------------|------------|------------|--------------|
| 0             | 1          | 0         | 0          | 0          | 0          | 0          | 0            |
| 1             | 0          | 0         | 1          | 1          | 1          | 0          | 0            |
| 2             | 0          | 0         | 0          | 0          | 0          | 0          | 0            |
| 3             | 0          | 0         | 1          | 0          | 0          | 0          | 0            |
| 4             | 0          | 1         | 1          | 1          | 0          | 1          | 1            |



