2020 Digital IC Design Homework 2: Divider

NAME	Tran Thi Ai
Student ID	P76087081

Simulation Result

Functional simulation	Pass	Gate-level simulation	Pass	Gate-level simulation time	2555914 (ns)

(your pre-sim result)

```
#       65525 data is correct
#       65526 data is correct
#       65527 data is correct
#       65528 data is correct
#       65529 data is correct
#       65530 data is correct
#       65531 data is correct
#       65532 data is correct
#       65533 data is correct
#       65534 data is correct
#       65535 data is correct
#       65536 data is correct
# --------------------PASS--------------------
# All data have been generated successfully!
```

(your post-sim result)

```
#       65523 data is correct
#       65524 data is correct
#       65525 data is correct
#       65526 data is correct
#       65527 data is correct
#       65528 data is correct
#       65529 data is correct
#       65530 data is correct
#       65531 data is correct
#       65532 data is correct
#       65533 data is correct
#       65534 data is correct
#       65535 data is correct
#       65536 data is correct
# --------------------PASS--------------------
# All data have been generated successfully!
```

Synthesis Result

Total logic elements	127 / 68,416 (< 1 %)
Total memory bit	0 / 1,152,000 (0 %)
Embedded multiplier 9-bit element	0 / 300 (0 %)

(your flow summary)

Flow Summary

Flow Status	Successful - Sat Apr 11 21:47:29 2020
Quartus II Version	10.0 Build 262 08/18/2010 SP 1 SJ Full Version
Revision Name	div
Top-level Entity Name	div
Family	Cyclone II
Device	EP2C70F896C8
Timing Models	Final
Met timing requirements	Yes
Total logic elements	127 / 68,416 (< 1 %)
Total combinational functions	127 / 68,416 (< 1 %)
Dedicated logic registers	0 / 68,416 (0 %)
Total registers	0
Total pins	25 / 622 (4 %)
Total virtual pins	0
Total memory bits	0 / 1,152,000 (0 %)
Embedded Multiplier 9-bit elements	0 / 300 (0 %)
Total PLLs	0 / 4 (0 %)

Description of your design

In my design, The first, we need to initialize P value equal 0. The a (in1) is dividend and the b (in2) is divisor. Load divisor and dividend into register a and b, respectively; clear partial-remainder register P. These variables are going to update during looping. Then, we start the for loop, it will run n time where n=n0 of bits. We implemented left shift register pair P and left shift a (in1) one bit. After that, compute P=P-b. Repeat, if P[8] == 1, a[0]=0; P= P + b. else a[0] = 1; Finally, the quotient = a.

```
`timescale 1ns / 10ps
`define CYCLE 39 // can be modified
module div_tb;
parameter width = 8;
```

This is the clock cycle modified when post-sim.

*Scoring = (Total logic elements + total memory bit + 9*embedded multiplier 9-bit element) × (gate-level simulation time in ns)*