David Elliott

Evan Yeh

EE125 Homework #4

Exercise 1:

1. Code:

01 -------------------------------------------------------------------------

02 **library** ieee**;**

03 **use** ieee**.**std\_logic\_1164**.all;**

04 **use** ieee**.**numeric\_std**.all;**

05 **use** ieee**.**math\_real**.all;**

06 -------------------------------------------------------------------------

07 **entity** log2\_prob **is**

08 **generic** **(**BITS**:** natural **:=** 8**);**

09 **port** **(**x**:** **in** std\_logic\_vector**(**BITS **-** 1 **downto** 0**);**

10 -- generically:

11 -- y: out std\_logic\_vector(ceil(log2(BITS)) downto 0);

-- not synthesizeable

12 -- for simulation:

13 y**:** **out** std\_logic\_vector**(**3 **downto** 0**));**

14 **end** **entity;**

15 -------------------------------------------------------------------------

16 **architecture** problem\_one **of** log2\_prob **is**

17 **function** ceil\_log2 **(**input**:** positive**)** **return** integer **is**

18 **variable** i**:** integer **:=** 0**;**

19 **begin**

20 **while** **(**2 **\*\*** i **<** input and i **<=** BITS**)** **loop**

21 i **:=** i **+** 1**;**

22 **end** **loop;**

23 **return** i**;**

24 **end** **function** ceil\_log2**;**

25

26 **signal** x\_int**:** integer**;**

27 **begin**

28 x\_int **<=** positive**(to\_integer(**unsigned**(**x**)));**

29 -- generically:

30 -- y <= std\_logic\_vector(to\_unsigned(ceil\_log2(x\_int), ceil\_log2(BITS));

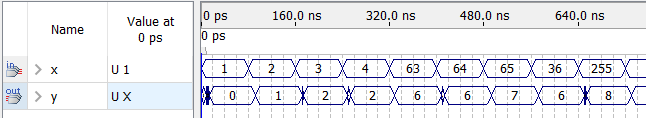
31 -- for simulation:

32 y **<=** std\_logic\_vector**(to\_unsigned(**ceil\_log2**(**x\_int**),** 4**));**

33 **end** **architecture;**

34 -------------------------------------------------------------------------

1. Simulation:



Exercise 2:

Exercise 3:

1. Code:

01 ---- tflipflop:----------------------------------------------------------

02 **library** ieee**;**

03 **use** ieee**.**std\_logic\_1164**.all;**

04 -------------------------------------------------------------------------

05 **entity** tflipflop **is**

06 **port** **(**a**,** b**,** clk**:** **in** std\_logic**;**

07 x**:** **out** std\_logic**;**

08 q**:** **buffer** std\_logic**);**

09 **end** **entity;**

10 -------------------------------------------------------------------------

11 **architecture** tflipflop **of** tflipflop **is**

12 **begin**

13 **process(**clk**)**

14 **begin**

15 x **<=** a and b**;**

16 **if** **rising\_edge(**clk**)** **then**

17 q **<=** **(**a and b**)** xor q**;**

18 **end** **if;**

19 **end** **process;**

20 **end** **architecture;**

21 ----Main code:-----------------------------------------------------------

22 **library** ieee**;**

23 **use** ieee**.**std\_logic\_1164**.all;**

24 **entity** sync\_counter **is**

25 **generic** **(**BITS**:** natural **:=** 4**);**

26 **port** **(**a**,** b**,** clk**:** **in** std\_logic**;**

27 q\_vec**:** **buffer** std\_logic\_vector**(**BITS **-** 1 **downto** 0**));**

28 **end** **entity;**

29 -------------------------------------------------------------------------

30 **architecture** sync\_counter **of** sync\_counter **is**

31 **signal** x\_vec**:** std\_logic\_vector**(**BITS **-** 1 **downto** 0**);**

32

33 **component** tflipflop **is**

34 **port** **(**a**,** b**,** clk**:** **in** std\_logic**;**

35 x**:** **out** std\_logic**;**

36 q**:** **buffer** std\_logic**);**

37 **end** **component;**

38

39 **begin**

40 cell**:** tflipflop **PORT** **MAP** **(**a**,** b**,** clk**,** x\_vec**(**0**),** q\_vec**(**0**));**

41 gen**:** **for** i **in** 1 **to** BITS **-** 1 **generate**

42 cells**:** tflipflop **PORT** **MAP** **(**x\_vec**(**i**-**1**),** q\_vec**(**i**-**1**),** clk**,** x\_vec**(**i**),**

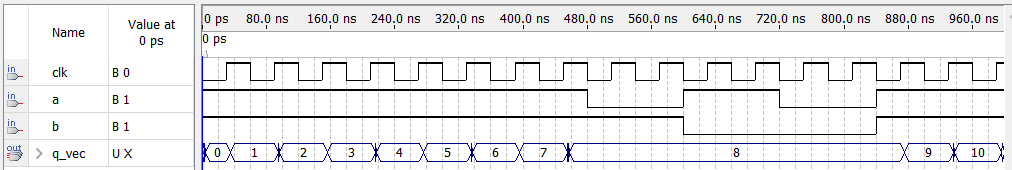
q\_vec**(**i**));**

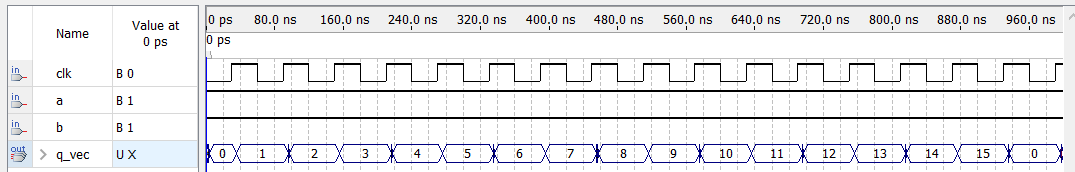
43 **end** **generate** gen**;**

44

45 **end** **architecture;**

1. Simulation:





Exercise 4