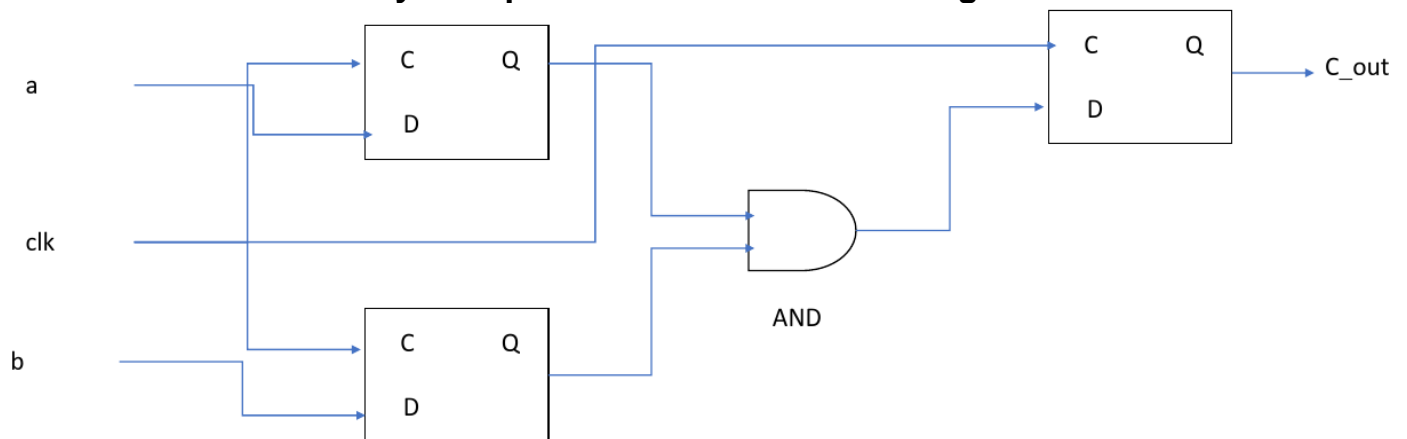
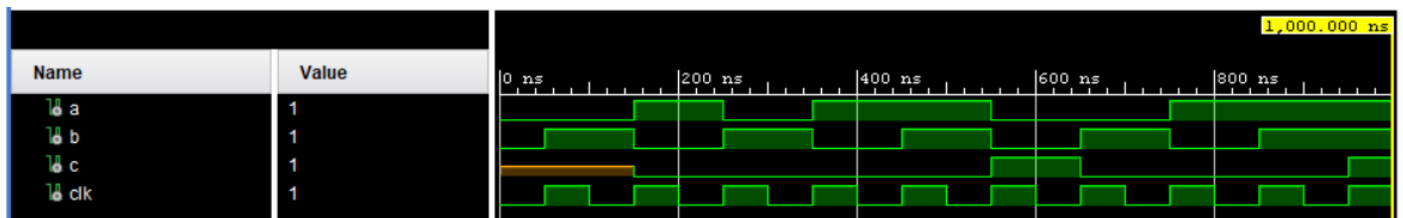


RC Lab Module 1, Task 1

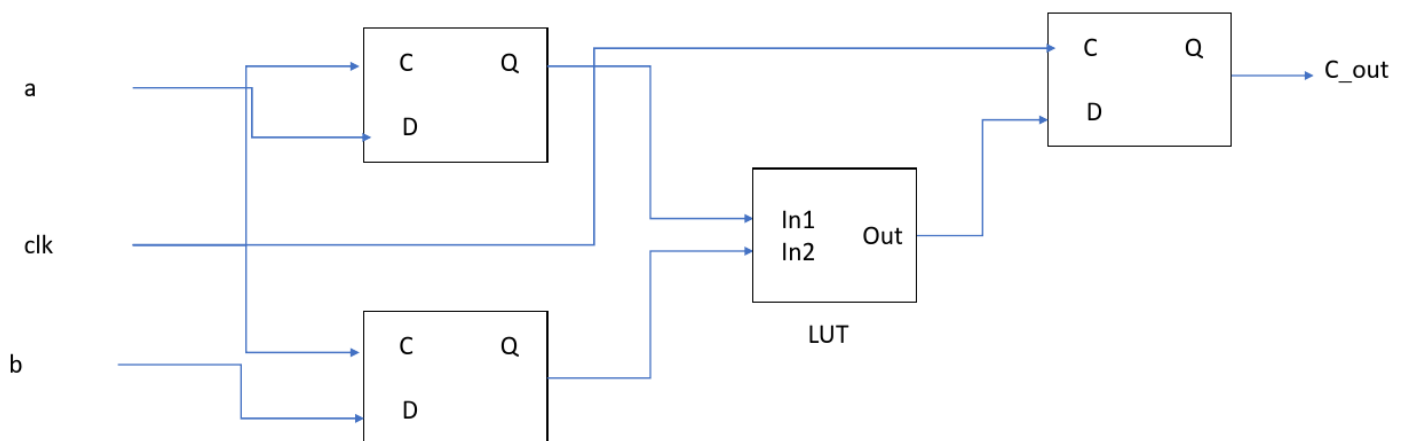
2.2 Draw the schematic you expect for the elaborated design



2.3 Simulate your Design



2.5 Synthesize the Design



How many LUTs (total) are available on the FPGA? **63400**

How many LUTs did you use as logic? **1**

How many LUTs did you use as memory? **0**

Is the resource utilization after synthesis accurate? **Yes**

How many DSPs are available on the FPGA? **0**

Let Vivado create the Timing Summary (you can use the default parameters).

What is the worst negative slack in your design? **8.998 ns**

What is the worst hold slack in your design? **0.122 ns**

2.6 Implement the Design

What is the worst negative slack in your design? **9.069 ns**

What is the worst hold slack in your design? **0.192 ns**

Compare the slack values to the results from synthesis. Are the values equal to the results

from synthesis? If yes, why? If no, why?

No. Synthesis design creates a netlist based on registers, clocks and gates in the required design, whereas implementation actually places the design on fpga board and creates routing between components. Thus, due to constraints imposed by fpga on such placement, timing values can be generally expected to be higher than those of synthesis results.

2.7 Custom Placement and Routing

Since no input or output delays have been specified: What is the delay you can focus on to destroy the timing? **Route delay due to placement of LUT.**

Modifying the placement:

What are you effectively modifying when you modify the placement?

Register A

Register B

Register C

LUT

Modifying the routing:

Re-run the implementation.

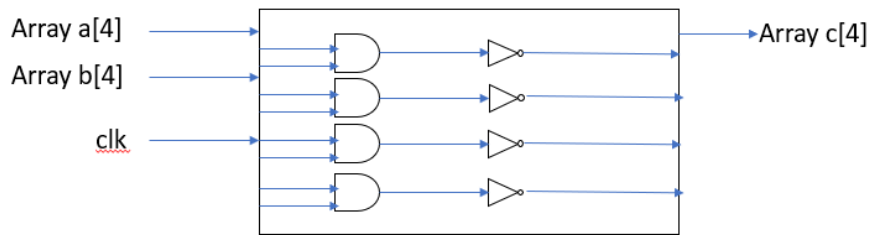
What is the worst negative slack in your design? **3.712 ns**

What is the worst hold slack in your design? **2.472 ns**

RC Lab Module 1, Task 2

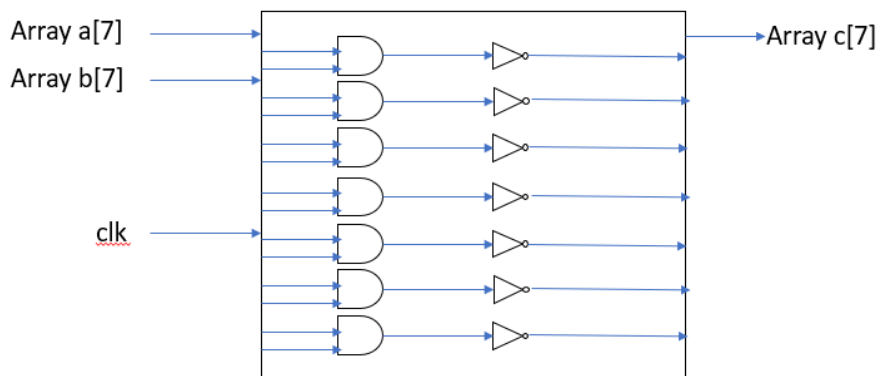
Assuming gates with two inputs to be used by Vivado.

Draw the schematic you expect for the elaborated design with $N = 4$:



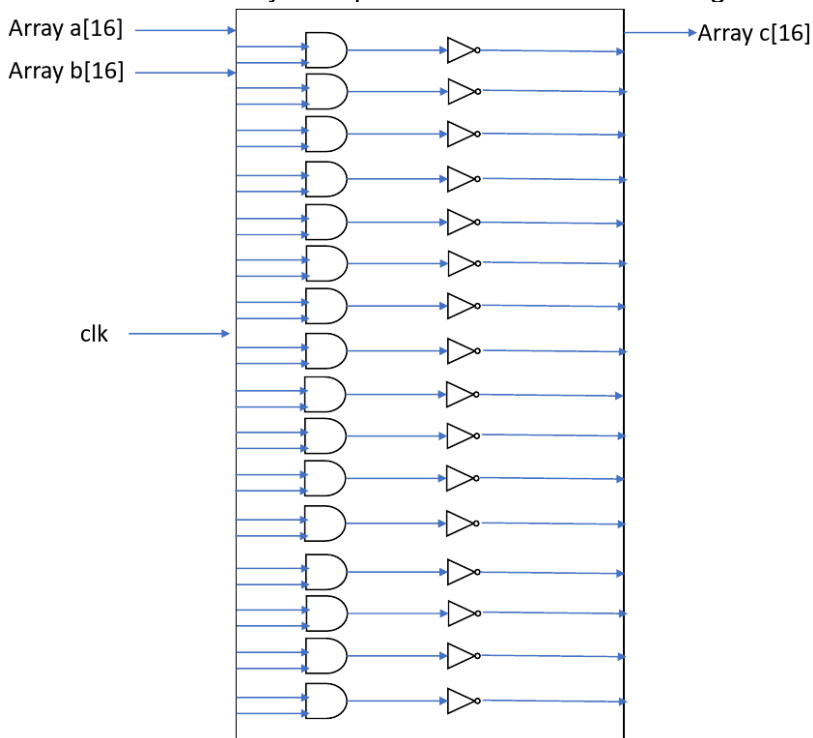
How many logical gates are you expecting to be used with $N = 4$? **8**

Draw the schematic you expect for the elaborated design with $N = 7$:



How many logical gates are you expecting to be used with $N = 7$? **14**

Draw the schematic you expect for the elaborated design with $N = 16$:



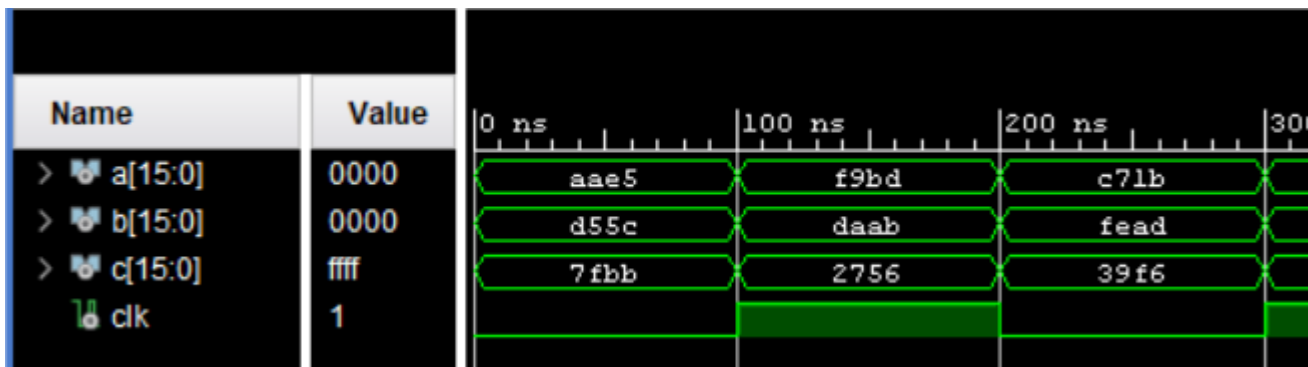
How many logical gates are you expecting to be used with $N = 16$? **32**

Verify the correctness of your design by comparing your expected schematic to the schematic generated by Vivado. Do you see structural differences in the schematics? If yes, why?

Ans

All 16 'And + Not gates' are generated irrespective of N value, but we expected that it would correspond to the value of N.

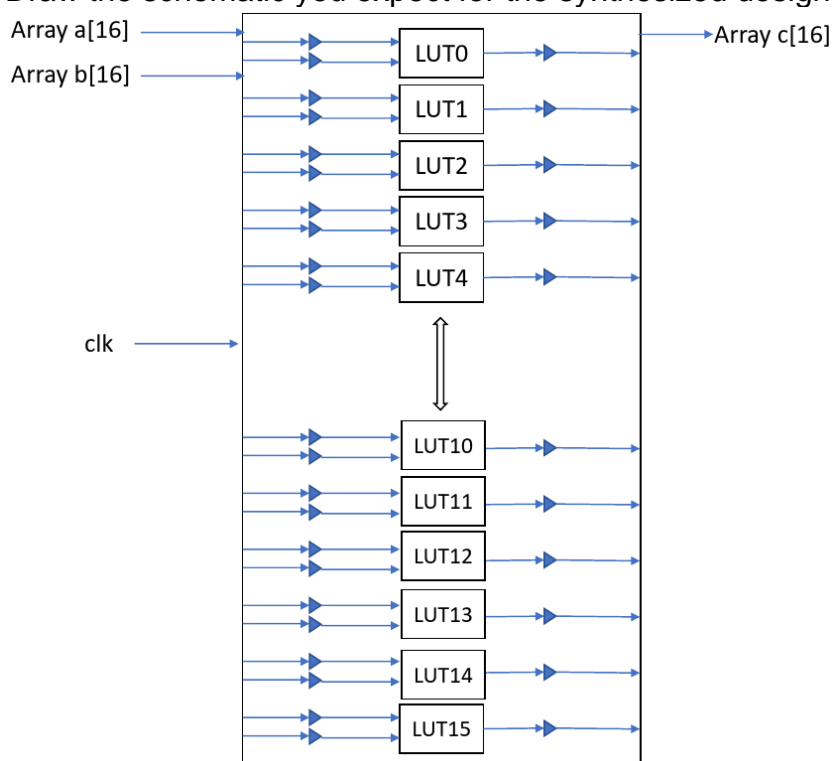
2.2 Simulate the Design



2.3 Synthesize the Design

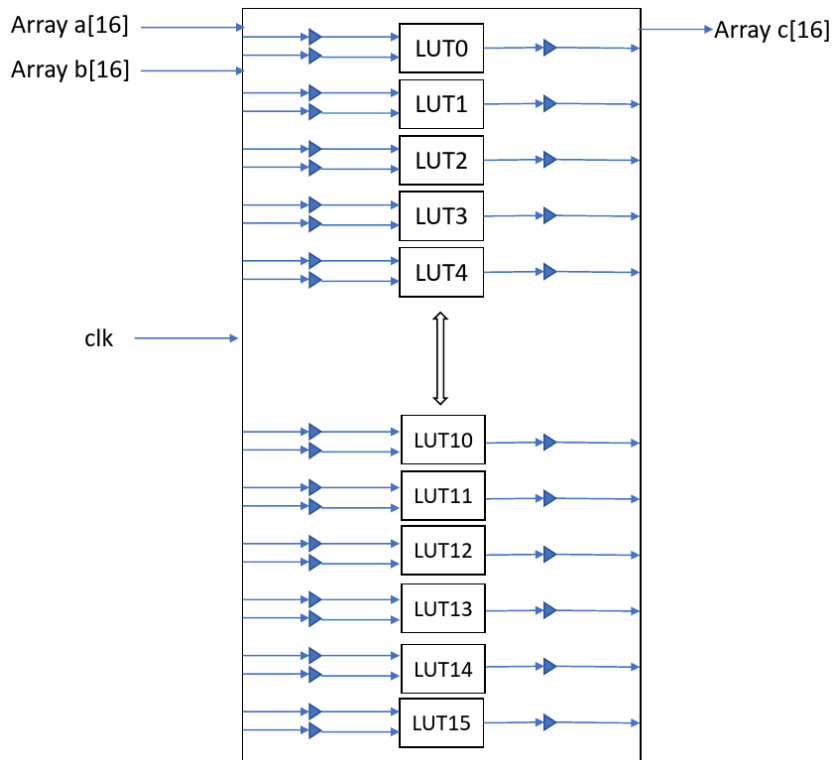
Determine the parameter K (#inputs per LUT) for the LUTs used in the FPGA of the Nexys board: **2** (verified via Truth table of LUT and from synthesised schematic)

Draw the schematic you expect for the synthesized design with N = 4:



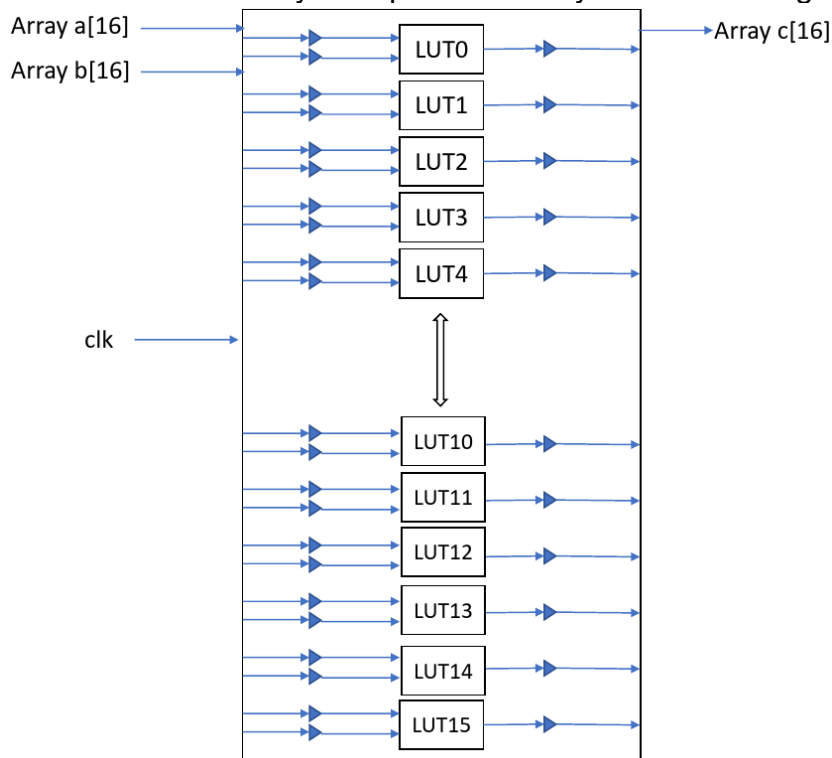
How many LUTs are you expecting to be used with N = 4? **4**

Draw the schematic you expect for the synthesized design with N = 7:



How many LUTs are you expecting to be used with $N = 7$? **7**

Draw the schematic you expect for the synthesized design with $N = 16$:



How many LUTs are you expecting to be used with $N = 16$? **16**

Synthesize the design and compare your schematic to the schematic after synthesis. Do the schematics differ from each other. If yes, why?

Ans

Yes, based on the difference in schematic of assignment 1, we used buffers in our expected schematic for this assignment 2. But, after the synthesis we understood that all lines of inputs are

not directly connected to LUT but through a common barrier/ connection which simplifies the design.

RC Lab Module 1, Task 3

2.2 Synthesize and Implement the Design for Different Sizes of the ROM

N_ROM_DATA		3	7	65	150	200	250	350
N_ROM_ADDR		2	3	7	8	8	8	9
Synthesis	LUT	4	6	13	19	18	19	26
	FF	36	38	53	56	56	56	50
	BRAM	0	0	0	0	0	0	0.5
Implementation	LUT	4	6	13	19	18	19	26
	FF	36	38	53	56	56	56	50
	BRAM	0	0	0	0	0	0	0.5

What can you observe when looking at the resource utilization?

Ans

As per the general trend observed via table, *utilisation of LUT and FF increases as we increase N_ROM_DATA* (and N_ROM_ADDR). Specifically, BRAM is used only in the last case as number of LUT increases and N_ROM_DATA increases to 350.

Based on the schematic for the last case where BRAM is used, all the data outputs from previous level of registers are connected to BRAM and thereafter, data output is connected to next set of registers. This effectively, *reduces number of registers between 2 sets of LUTs as BRAM be a buffer.*

If we have *very few data points as opposed to maximum achievable value with N_ROM_ADDR*, then *BRAM is not used.* Because there are *lesser number of registers* in contrast to cases where utilization is maximum.