

**ISTANBUL TECHNICAL UNIVERSITY
COMPUTER ENGINEERING DEPARTMENT**

**BLG 242E
DIGITAL CIRCUITS LABORATORY
EXPERIMENT REPORT**

EXPERIMENT NO : 7
EXPERIMENT DATE : 05.04.2019
LAB SESSION : FRIDAY - 14.00
GROUP NO : G13

GROUP MEMBERS:

150180704 : CİHAT AKKIRAZ
150180707 : FATİH ALTINPINAR
150180734 : SİNAN ŞAR

SPRING 2019

ETHIC FORM for BLG242E Logic Circuits Laboratory

As a student of

Istanbul Technical University Faculty of Computer and Informatics Engineering;

1. I will not attempt to cheat in quizzes and final exam,
2. I will not use disallowed sources or tools (mobile phone, calculator etc.) during the exam,
3. I will not write any information (formula, text, figure etc.) on the table, sheets or books that are allowed to be used during the exam,
4. I will give reference when using printed or online published sources,
5. I will not use the results in a source as they are, or by changing a part of them without giving a reference,
6. I will not show unused sources as used,
7. I will not present someone else's idea as my own idea,
8. I will not make someone do my homework, project or thesis for money or anything else,
9. I will not take an exam or enter a lecture on behalf of others,
10. I will not make excuses for not attending exams or lessons by taking reports from someone I know (medical doctor parents or relatives),
11. I will refrain from deliberately harming the public materials at our university,
12. I will comply with the safety rules in laboratory work,
13. I will behave in accordance with the rules of respect for the lecturers and teaching assistants

signed by

150180704 : CİHAT AKKIRAZ

150180707 : FATİH ALTINPINAR

150180734 : SİNAN ŞAR

Contents

ETHICS	1
FRONT COVER	
CONTENTS	
1 INTRODUCTION	1
.....	1
2 REQUIREMENTS	1
2.1 PART 1	1
.....	1
2.2 PART 2	2
.....	2
2.3 PART 3	3
.....	3
2.4 PART 4	4
.....	4
.....	4
.....	4
.....	5
.....	5
.....	6
.....	6
3 INTERPRETATION OF THE RESULTS	7
.....	7
4 CONCLUSION	7
.....	7
REFERENCES	8

1 INTRODUCTION

In this experiment series of circuits are built in order to understand how latches, flip flow behave and how to manipulate a signal by a circular shifter.

2 REQUIREMENTS

Tools Used[1]

- C.A.D.E.T
- 74000 series ICs
 - 74xx¹00 - Quadruple 2-input Positive NAND Gates
 - 74xx¹02 - Quadruple 2-input Positive NOR Gates
 - 74xx¹04 - Hex Inverters
 - 74xx¹75 - Quadruple Bistable D Type Latches
 - 74xx¹165 - 8-Bit Parallel Input/Serial Output Shift Register
- Oscilloscope
- Function Generator

2.1 PART 1

In this part of the experiment, a S-R latch has to be built with NOR gates. In order to create the base of S-R latch, a stable state needs to be obtained. A stable state can be briefly explained as a circuit that does not oscillate. With the addition of Set and Reset inputs to the leftover inputs of NOR gates, a functioning S-R latch is obtained(see Figure 1). As it can be seen in Table 1, if given zeroes to the inputs of the latch, state remains the same. Set input sets state to 1, reset sets it to 0 as expected. However if high logic applied to both inputs at once, latch loses its stable state and oscillates too fast. Output is observed as 0 in both Q and \bar{Q} . Applying 1 to both inputs is forbidden since it makes the circuit unpredictable, yielding different values. Characteristic function of S-R latch can be seen from Table 1:

- $Q(t+1) = S + \bar{R} \cdot Q(t)$

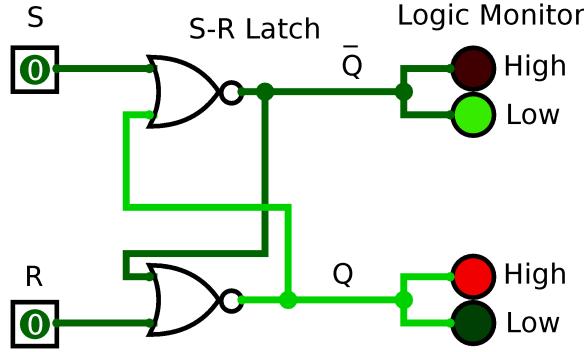


Figure 1: Circuit of S-R Latch

S	R	$Q(t+1)$
0	0	$Q(t)$
0	1	0
1	0	1
1	1	ϕ Forbidden

Table 1: Truth table of S-R Latch

2.2 PART 2

With the addition of an enable input, another S-R latch is implemented as seen in Figure 2. This addition allows S-R latch to change its state only if the enable input is on (high logic state). If given 0 to enable input (C) no change can be observed, otherwise everything is same as it is in Part 1. Except, if given 1 to both Set and Reset inputs to an enabled latch, output is observed on Q and \bar{Q} were 1 rather than zero. This difference is caused by the difference in gates used. The latch is still unpredictable and oscillating very fast.

C	S	R	$Q(t+1)$
0	X	X	$Q(t)$
1	0	0	$Q(t)$
1	0	1	0
1	1	0	1
1	1	1	ϕ Forbidden

Table 2: Truth table of S-R Latch with enable

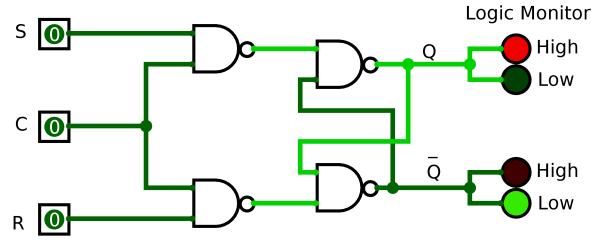


Figure 2: Circuit of S-R Latch with enable

2.3 PART 3

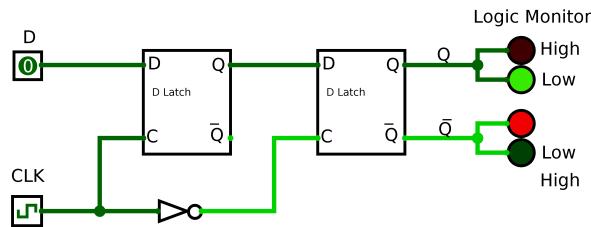


Figure 3: Circuit of falling edge D flip-flop

In this part, a D type flip flop is constructed (see Figure 3), using D type latches. D type latch is a latch that has the characteristic function $Q = D$ if the latch is enabled, otherwise it keeps its state. As it can be seen in Table 3, state only changes when clock is changing from a high state to low state. Which makes this circuit, a falling edge flip flop.

D	CLK	$Q(t+1)$
0	\downarrow	0
1	\downarrow	1
x	0	$Q(t)$
x	\uparrow	$Q(t)$
x	1	$Q(t)$

Table 3: Truth table of falling edge D flip-flop

2.4 PART 4

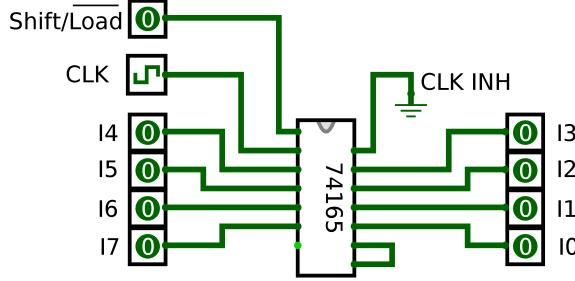


Figure 4: Circuit built to manipulate input signal

In the final part of the experiment, pulse generator is implemented using a shift register. Q_H output of the shift register (bit that is shifted out from I_7) is connected to SER (new value of I_0).Results are observed for different frequency values and pulse/gap duration rates using the oscilloscope provided in the laboratory. As input, 8 kHz square wave is used since register have 8 bits. This produced nicer graphs in oscilloscope.

In every figure, blue wave is the 8 kHz signal that is produced by signal generator, yellow is the signal that is changed by the shift register.

To obtain 1/2 frequency from the input signal, 10101010 bits are loaded to the shift register. Output of the signal can be seen in Figure 5. In the Figure, a single, full cycle of the wave that is produced by the register is as long as 2 full cycles of the wave that is produced by the signal generator. This shows yellow wave has 2 times of the wavelength of blue wave.

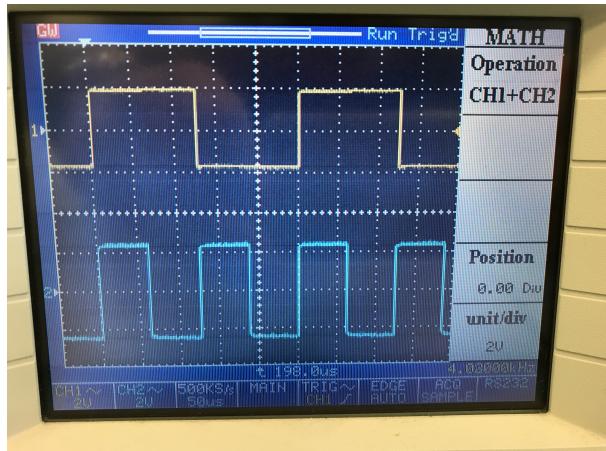


Figure 5: 1/2 frequency of input

To obtain 1/4 frequency from the input signal, 11001100 bits are loaded to the shift register. Output of the signal can be seen in Figure 6. As it can be counted from the figure, a full cycle of the yellow wave is as long as 4 blue cycles.

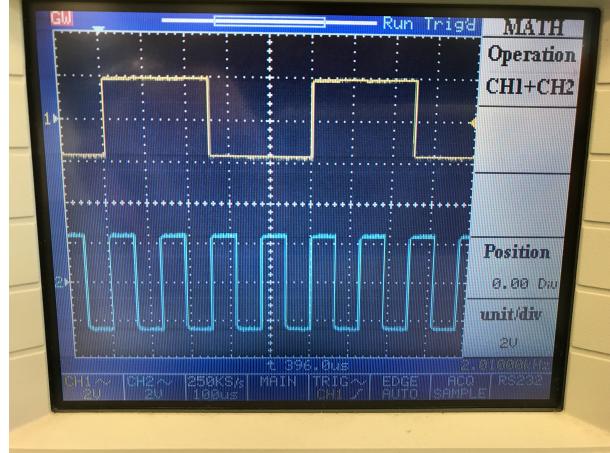


Figure 6: 1/4 frequency of input

In order to obtain 1/8 frequency from the input signal, 11110000 bits are loaded to the shift register. From the Figure 7, it is clear that, yellow wave's wavelength is 8 times the wavelength of the blue signal.

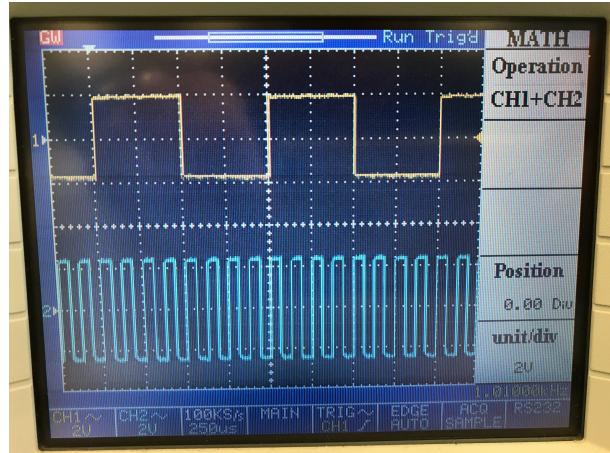


Figure 7: 1/8 frequency of input

In order to produce a signal that has 1/3 pulse-gap ratio, 10001000 bits are loaded to the shift register. This gives a signal that can be seen in Figure 8 which stays in low logic longer than high logic state. It takes 1 cycle from the blue wave to end high logic, however low logic level takes 3 cycles. This concludes that signal is manipulated correctly.

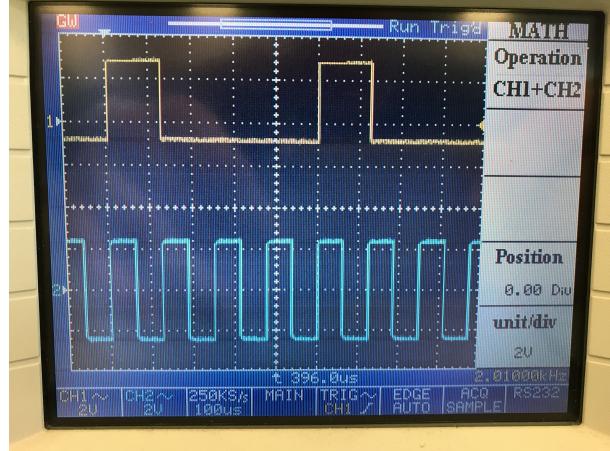


Figure 8: 1/3 pulsegap duration rate

Loading 10000000 bits to the shift register produces the wave in Figure 9. This time high level takes 1 cycle and low logic level 7 cycles from the blue wave to complete.

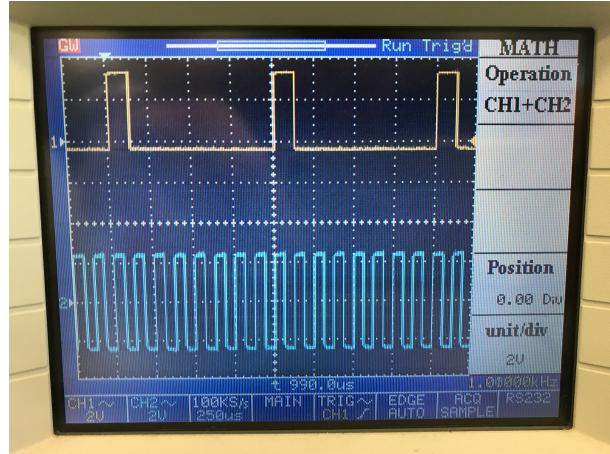


Figure 9: 1/7 pulsegap duration rate

3 INTERPRETATION OF THE RESULTS

There weren't any results which contradicted with the theoretical result we have calculated before the experiment. For forbidden inputs, we observed different, unpredictable result which differ because of the different gates used in different circuits as explained above. Also, we have learned that input signal can be manipulated by using a circular shift register.

4 CONCLUSION

We didn't face any particular difficulties that were worth noting throughout the experiment. We were able to complete the given tasks quite swiftly. All in all, this experiment helped us improve our abilities to implement latches and flip-flops in a multiplicity of different situations.

REFERENCES

- [1] Istanbul Technical University Department of Computer Engineering. Blg 242e logic circuits laboratory experiments booklet version 1.9, Spring 2019.
- [2] Overleaf documentation <https://tr.overleaf.com/learn>.
- [3] Detailed info on writing reports <https://projects.ncsu.edu/labwrite/res/res-studntintro-labparts.html>.