Mandatory Assignment

**Table of content**

[1 Introduction 1](#_Toc20299969)

[2 Boolean Arithmetic 1 1](#_Toc20299970)

[2.1 TASK: write down the Arithmetic equation for the logic described in Table 1 and reduce it as much as possible 1](#_Toc20299971)

[2.2 TASK: Implement the circuit in the simulator using AND, OR and inverter gates (left click on the gates and select “Edit” if you want more than 2 inputs). Create a hyper link with the designed circuit and insert it in the assignment 2](#_Toc20299972)

[3 Boolean Arithmetic 2 2](#_Toc20299973)

[3.1 TASK: write down the Arithmetic equation for the logic described in Table 2 and reduce it as much as possible 2](#_Toc20299974)

[3.2 TASK: Implement the circuit in the simulator using AND, OR and inverter gates (left click on the gates and select “Edit” if you want more than 2 inputs). Create a hyper link with the designed circuit and insert it in the assignment 2](#_Toc20299975)

[4 Sequential Logic 3](#_Toc20299976)

[4.1 Level triggered latch 3](#_Toc20299977)

[4.2 Rising edge triggered latch 4](#_Toc20299978)

[5 CPU 4](#_Toc20299979)

# Introduction

Mandatory group assignment 2-3 people.

Deadline is 29 september 2019

# Boolean Arithmetic 1

The truth table of some combinational logic can be seen in Table 1

Table 1: Truth table

|  |  |  |  |
| --- | --- | --- | --- |
| **A** | **B** | **C** | **Out** |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

## TASK: write down the Arithmetic equation for the logic described in Table 1 and reduce it as much as possible

## TASK: Implement the circuit in the [simulator](http://falstad.com/circuit/circuitjs.html?cct=$+1+0.000005+1.500424758475255+50+5+50%0A150+272+160+416+160+0+2+0+5%0AL+128+160+80+160+2+0+false+5+0%0AL+128+224+80+224+2+0+false+5+0%0AM+528+208+576+208+2+2.5%0AI+208+224+336+224+0+0.5+5%0A152+256+304+416+304+0+2+0+5%0AL+128+288+80+288+2+0+false+5+0%0Ax+539+178+577+181+4+24+Out%0Ax+53+140+69+143+4+24+A%0Ax+50+212+66+215+4+24+B%0Ax+51+276+68+279+4+24+C%0A) using AND, OR and inverter gates (left click on the gates and select “Edit” if you want more than 2 inputs). Create a hyper link with the designed circuit and insert it in the assignment

# Boolean Arithmetic 2

The truth table of some combinational logic can be seen in Table 2

Table 2: Truth table

|  |  |  |  |
| --- | --- | --- | --- |
| **A** | **B** | **C** | **Out** |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

## TASK: write down the Arithmetic equation for the logic described in Table 2 and reduce it as much as possible

## TASK: Implement the circuit in the [simulator](http://falstad.com/circuit/circuitjs.html?cct=$+1+0.000005+1.500424758475255+50+5+50%0A150+272+160+416+160+0+2+0+5%0AL+128+160+80+160+2+0+false+5+0%0AL+128+224+80+224+2+0+false+5+0%0AM+528+208+576+208+2+2.5%0AI+208+224+336+224+0+0.5+5%0A152+256+304+416+304+0+2+0+5%0AL+128+288+80+288+2+0+false+5+0%0Ax+539+178+577+181+4+24+Out%0Ax+53+140+69+143+4+24+A%0Ax+50+212+66+215+4+24+B%0Ax+51+276+68+279+4+24+C%0A) using AND, OR and inverter gates (left click on the gates and select “Edit” if you want more than 2 inputs). Create a hyper link with the designed circuit and insert it in the assignment

# Sequential Logic

## Level triggered latch

A level triggered latch can be seen on Figure 1.

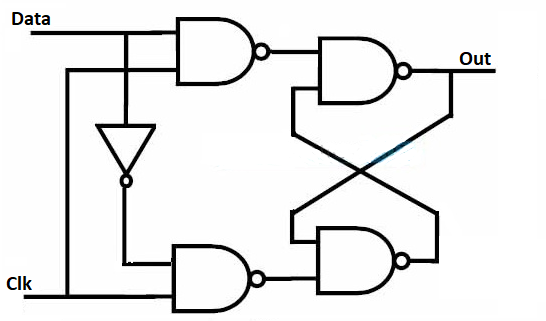


Figure 1 Data Flip Flop (DFF):

**TASK: Draw the Output on Figure 2.**

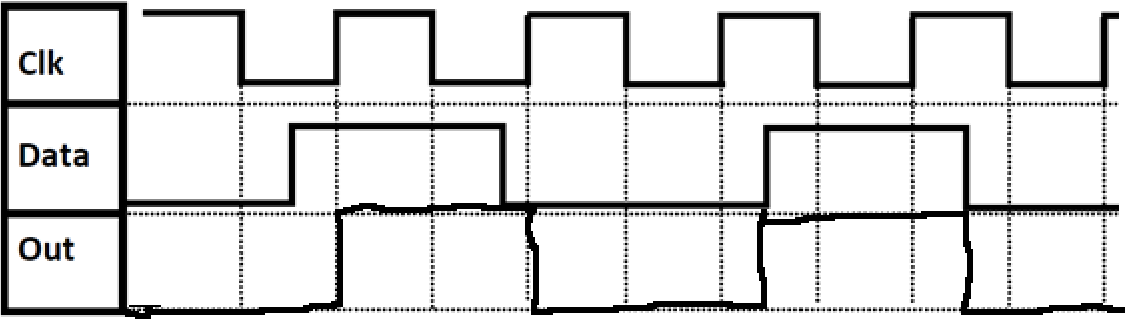


Figure 2: DFF timing diagram

## Rising edge triggered latch

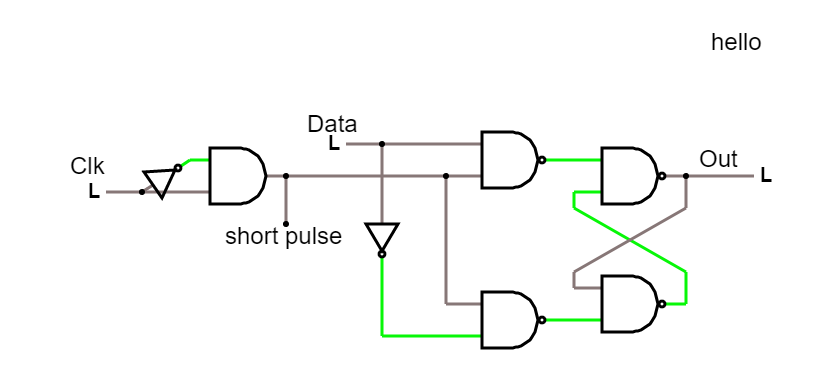


Figure 3: Rising edge latched

**TASK: Draw the Output on Figure 4.**

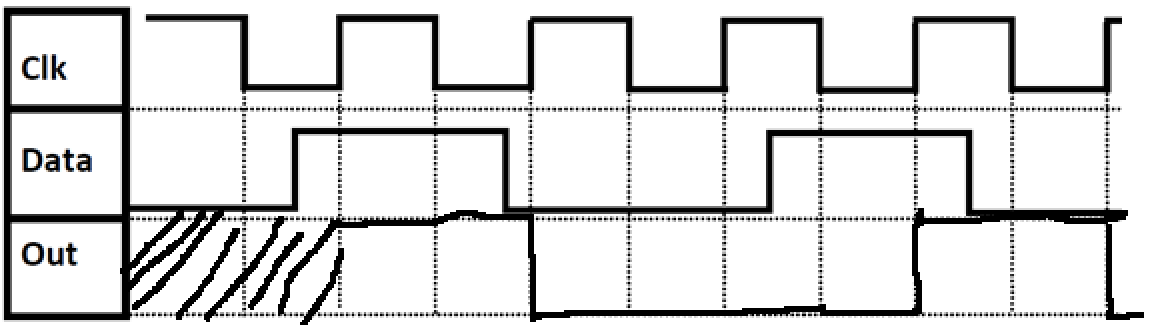


Figure 4: DFF timing diagram

# CPU

The 8-bit AVR CPU is running a program stored in the RAM. The first 3 instructions can be seen in Figure 5.

|  |  |  |  |
| --- | --- | --- | --- |
| **Address** | **Opcode** | **Hint** | **Explanation** |
| **0** | 1110 0001 0011 1111 | Look at page 115 in the [instructionset](http://ww1.microchip.com/downloads/en/devicedoc/atmel-0856-avr-instruction-set-manual.pdf) | LDI, value = 31, R19  R19 = 31 |
| **1** | 0110 1100 0011 0100 | Look at page 133 | ORI, Value = 196, R19,  R19= 223 |
| **2** | 0111 1010 0011 1100 | Look at page 36 | ANDI, Value = 172, R19  R19 = 140 |

Figure 5: Three instruction, that are fetched and executed by the CPU

The instructions set can be found [**Here**](http://ww1.microchip.com/downloads/en/devicedoc/atmel-0856-avr-instruction-set-manual.pdf)

**TASK: Explain what happens in each of the instructions, and write down the value of R19 after each of the instructions.**

**(HINT if done correctly register19 should contain the value 140 in the end)**