MARMARA UNIVERSITY
FACULTY OF ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING

DIGITAL LOGIC DESIGN
PROJECT REPORT

Students:

Oğuzcan ÇELİK 150114062

Mehmet Ali SAKALLI 150114080

a) ASSEMBLER PART

In this part of project we took strings from the input file and convert them to the hexadecimal and binary numbers according to the our isa file. We suppose to generate 20 bit numbers but by mistake we made this implementation only for binary numbers and our hexadecimal numbers become 16 bit. Here is the way look the output file after convertion regarding to the input file.

0x4100

0x420A

0x430D

0x4900

0xA300

0x9400

0x4449

0x4540

0x411F

0xE210

0xA500

0x9700

0x4776

0x4870

0x4111

0xD910

0x8300

0x0A21

0x8600

0x4B20

0x4C20

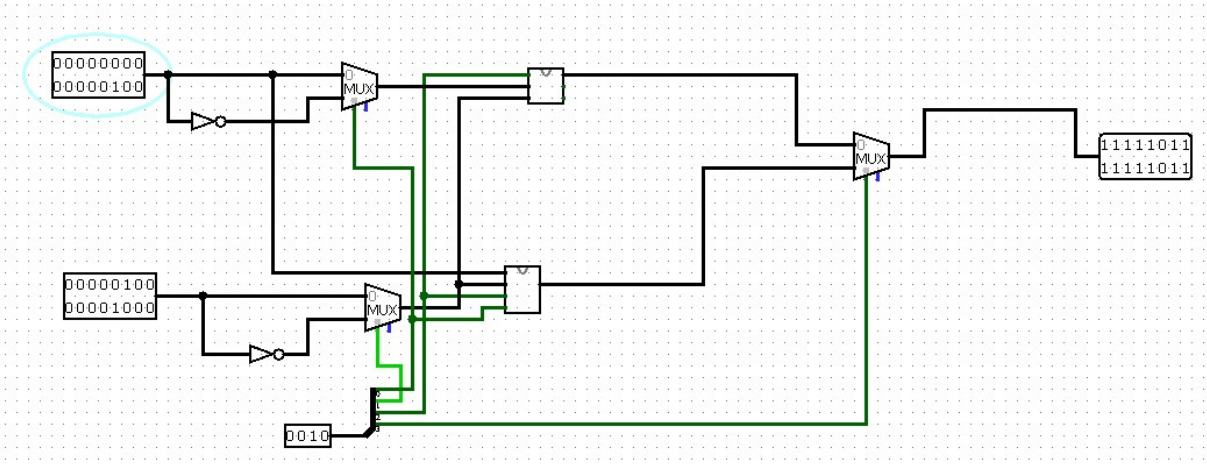
0x4D20

0x4E2

b) LOGISM PART

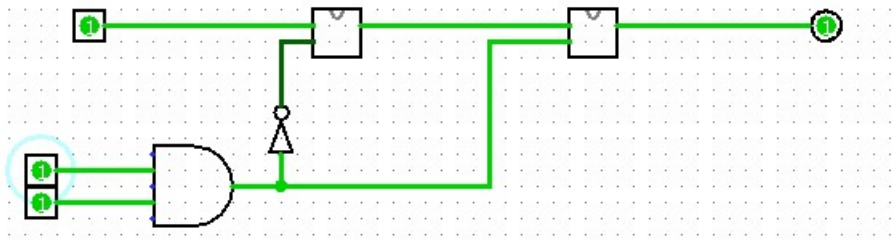
1) ALU

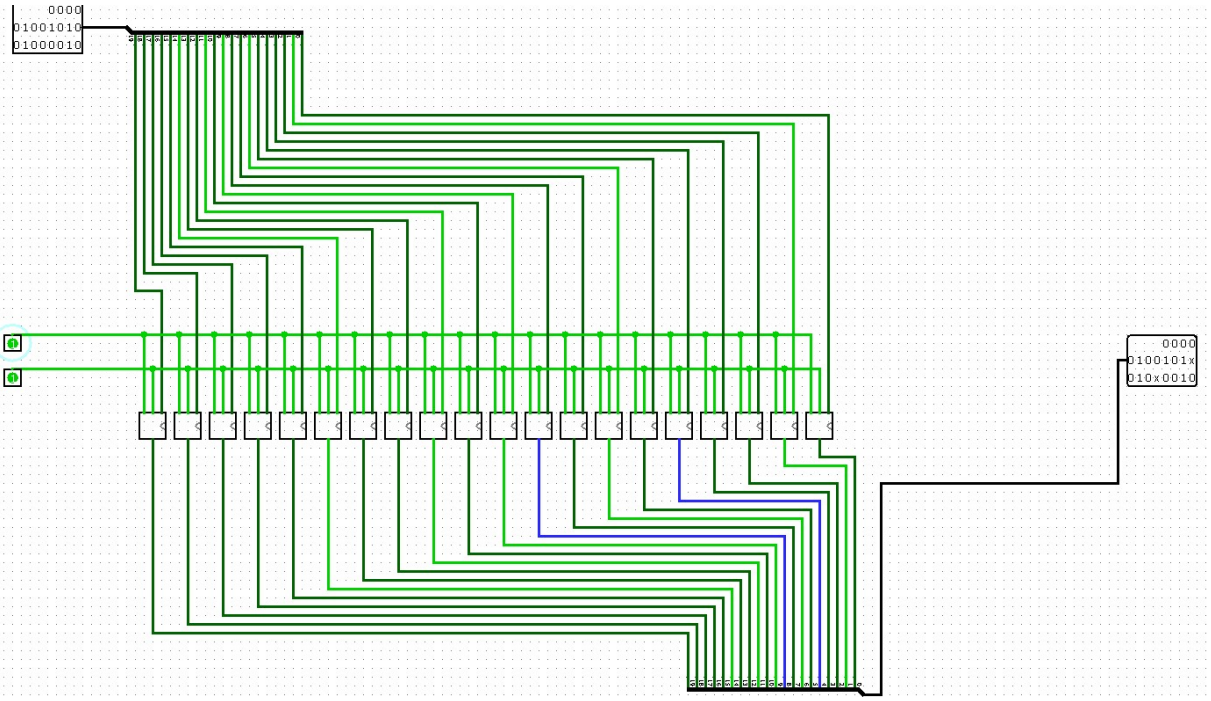
Arithmetic Logic Unit is the central component of the cpu. It performs all the arithmetic and logical operations on the data. All other components of our circuit design to provide data for ALU. To generate 20 bit ALU we create 20 bit Arithmetic Unit and 20 bit Logic Unit and combine them to get 20 bit ALU output. Here is our ALU circuit.



2) REGISTER

Registers are the memory of the circuits. They keep the data by using flip flops. In our design we used D flip flops to generate 20 bit register.First we create a DFF and after we combined 20 of them. After each clock cycle register keeps 20 bit data for 1 cycle and then passes it to the real memory. Here is the our DFF and Register implementation.





3) CONTROL UNIT (CU)

Control Unit takes the opcode from the instruction, and sets control wires which will control how the CPU will process the instruction. Since the setting of the control wires will only make sense once the use of the control wires is understood so it simply design related to the architecture and bit sizes for code part we want to provide. Here is the our CU implementation in logism.