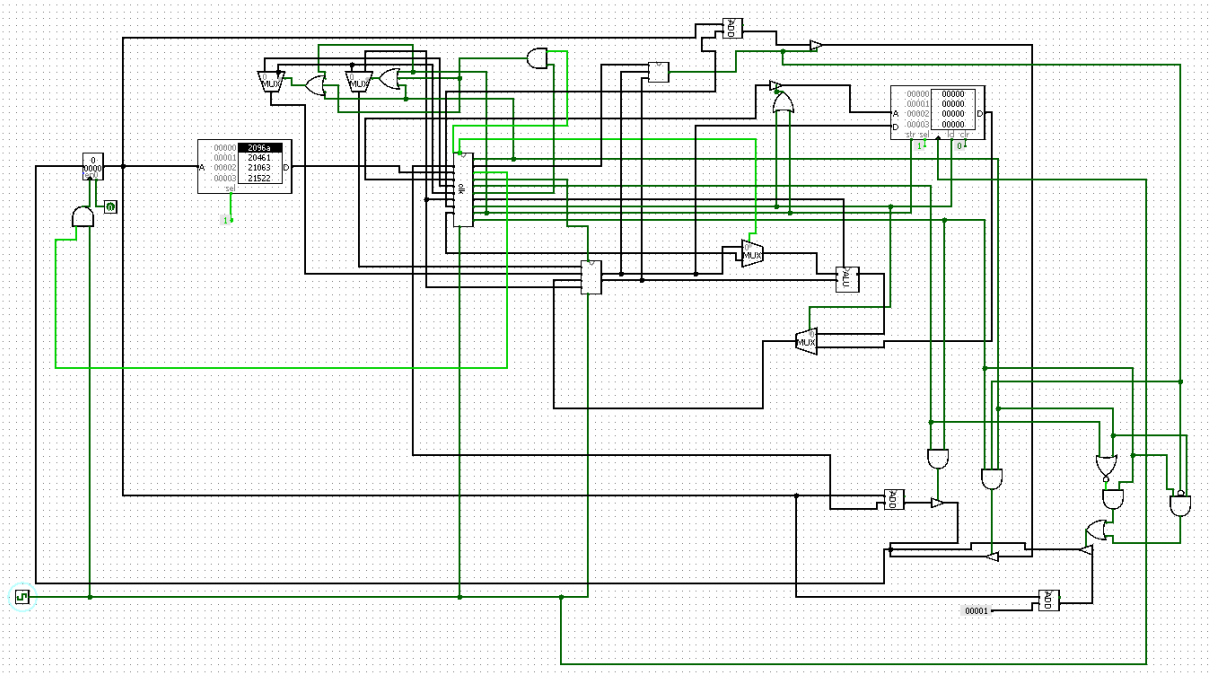
CSE3015
DIGITAL LOGIC DESIGN

SIMPLIFIED MICROPROCESSOR PROJECT REPORT

Serkan Aydın - 150114053

Simplified processor which supports instruction set: (AND, OR, ADD, LD, ST, ANDI, ORI, ADDI, XOR, XORI, JUMP, BEQ, BGT, BLT, BGE, BLE). Processor has 18 bits address width and 18 bits data width.There are 16 registers in processor.
The schematic view of datapath for simplified processor is below,

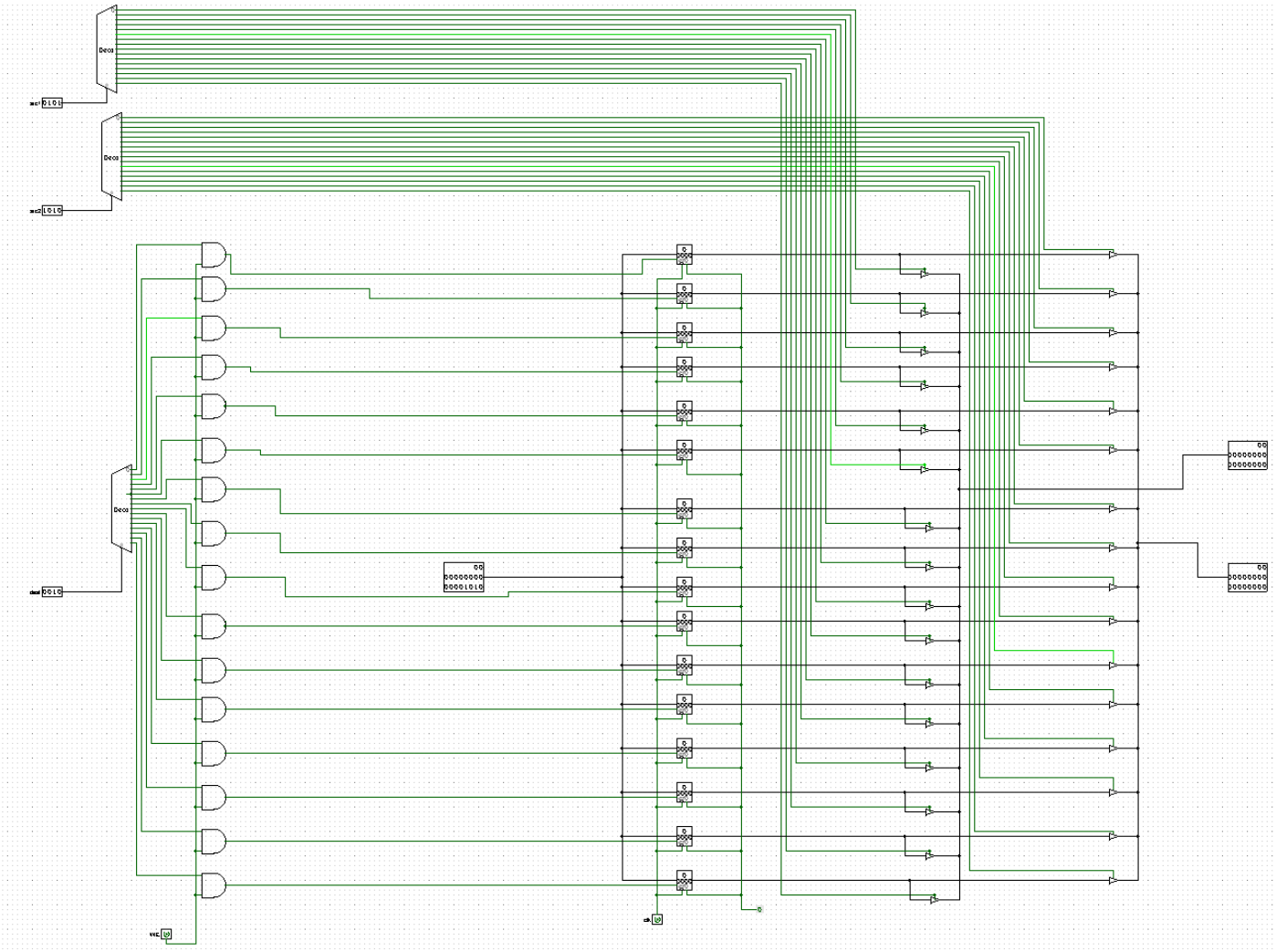


Instruction set architecture table;

OP	Opcode[17:14]	[13:10]	[9:6]	[5]	[4]	[3:0]
BEQ	0010	OP1	OP2	ADDR		
BLT	0011	OP1	OP2	ADDR		
BGT	0100	OP1	OP2	ADDR		
BLE	0101	OP1	OP2	ADDR		
BGE	0110	OP1	OP2	ADDR		
ADD	1000	DST	SRC1	0	0	SRC2
ADDI	1000	DST	SRC1	1	İmm5	
AND	1001	DST	SRC1	0	0	SRC2
ANDI	1001	DST	SRC1	1	İmm5	
OR	1010	DST	SRC1	0	0	SRC2
ORI	1010	DST	SRC1	1	İmm5	
XOR	1011	DST	SRC1	0	0	SRC2
XORI	1011	DST	SRC1	1	İmm5	
JUMP	1100	ADDR				
ST	1101	SRC	ADDR			
LD	1110	DST	ADDR			

Register File

There are 16 registers in *Register File*. It has 4 input port SRC1, SRC2, DEST,input and 2 output port output1 and output2.Writing operation to a register is controlled by WE signal. When WE is 1, data at the input port will be stored to DEST. When WE is 0, no register can be changed.

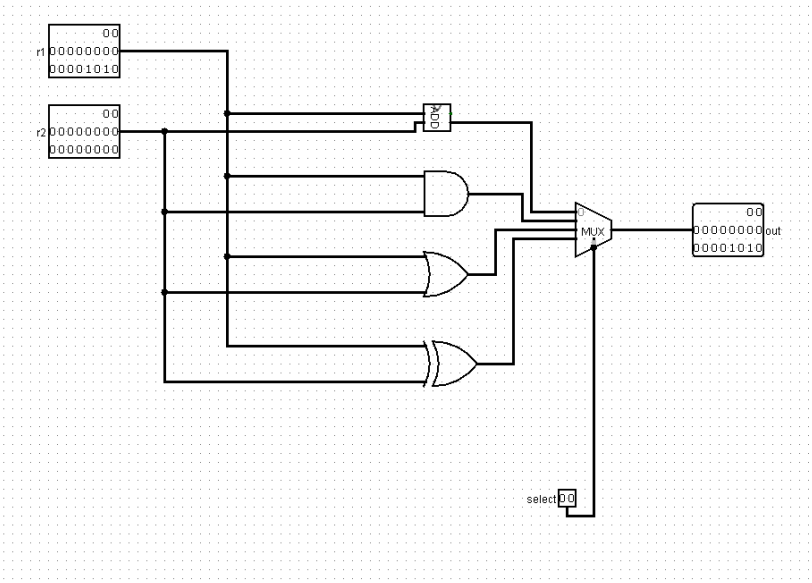


Arithmetic Logic Unit (ALU)

ALU has 2 inputs *r1* and *r2*. *r1* can be immediate value of instruction or *output1* which is directed by register file. *r2* is *output2* which is directed by register file
. It has one result as *out*.Signal *select* decides the operation to be done in *ALU*(*ADD,AND,OR,XOR)*

ALU select table;

00	ADD
01	AND
10	OR
11	XOR



Control Unit:

Control unit reads instruction memory then decides which operation will be executed and how the operation will be executed. There are 2 states which are executed for every instruction;

1-PC read operation which is called *Fetch1*

Instruction Memory[read_adress]<=*PC*

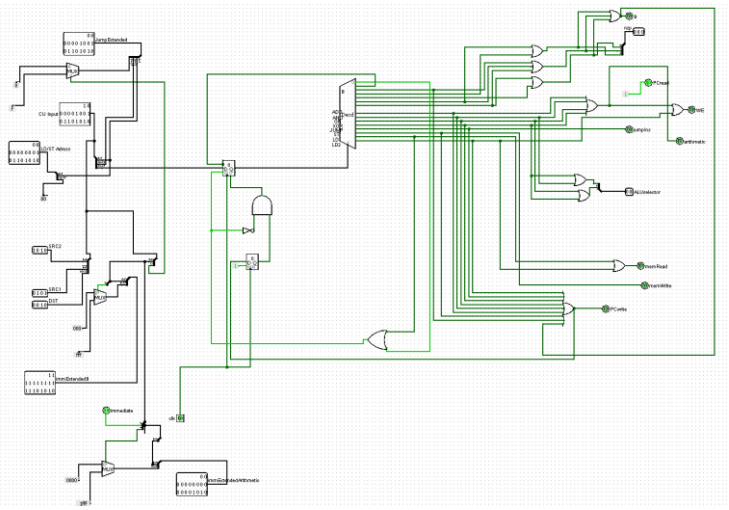
2- Instruction read operation which is called *Fetch2*
InstructionMemory[data_output]<=*InstructionMemory*[read_adress]

Except LD operation all the operations spend 3 clock cycle, LD operation spend 4 clock cycle

LD1	MemRead=1				
LD2	WE=1 PCwrite=1				
BEQ	B=1	nzp =010	PCwrite=1		
BLT	B=1	nzp =100	PCwrite=1		
BGT	B=1	nzp =001	PCwrite=1		
BLE	B=1	nzp =110	PCwrite=1		
BGE	B=1	nzp =011	PCwrite=1		
ADD	arithmetic =1	immediate =Instruction[5]	WE=1	ALUselector =00	PCwrite =1
AND	arithmetic =1	immediate =Instruction[5]	WE=1	ALUselector =01	PCwrite =1
OR	arithmetic =1	immediate =Instruction[5]	WE=1	ALUselector =10	PCwrite =1
XOR	arithmetic =1	immediate =Instruction[5]	WE=1	ALUselector =11	PCwrite =1
ST	memWrite =1 PCwrite =1				
JUMP	jumpIns =1 PCwrite=1				

PCread=1
InstructionRead = 1
B= BEQ | BLT | BGT | BLE | BGE
arithmetic= ADD | AND | OR | XOR
ALUselector[0] = AND | OR
ALUselector[0] = OR | XOR

nzp[0] = BGE | BGT
nzp[1] = BGE | BEQ | BLE
nzp[2] = BLE | BLT
immediate = Instruction[5]
memWrite = ST
memRead = LD1 | LD2
jumpIns = JUMP
PCwrite = BEQ | BLT | BGT | BLE | BGE | ADD | AND | OR | XOR | LD2 | JUMP



Branch:

Has 2 inputs from register file(op1,op2) and 1 input from control unit(nzp), 1 output Bout.Thecircuit looks to input nzp and comparison of inputs op1 and op2 then determines Bout.