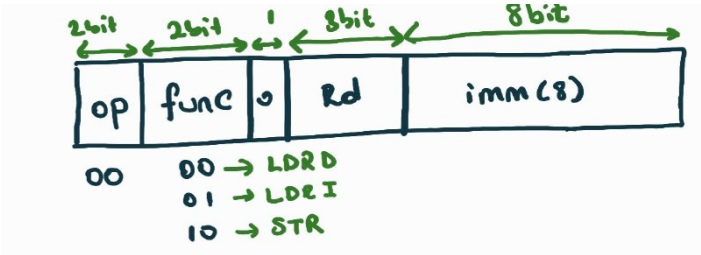
Zeynepnur ŞAHİNEL
2305399 05.06.2022

EE446 LABORATORY 4 PRELIMINARY WORK

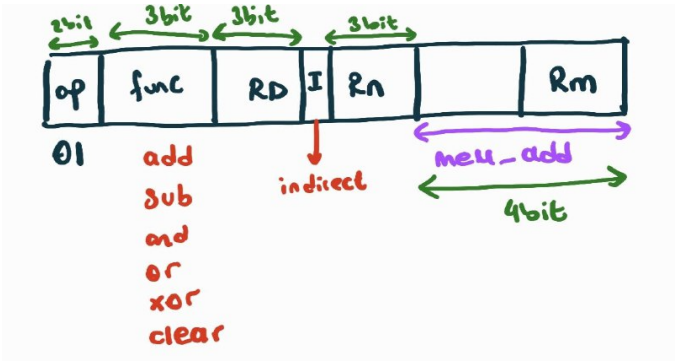
ISA & Datapath Design for Multi-Cycle CPU

Part 1) Instruction Set Architecture (ISA)

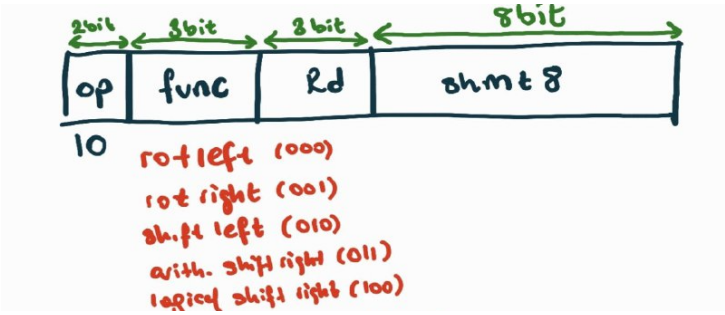
Instructions Classification:
-Memory/Register Instructions



-Arithmetic and Logical Instructions



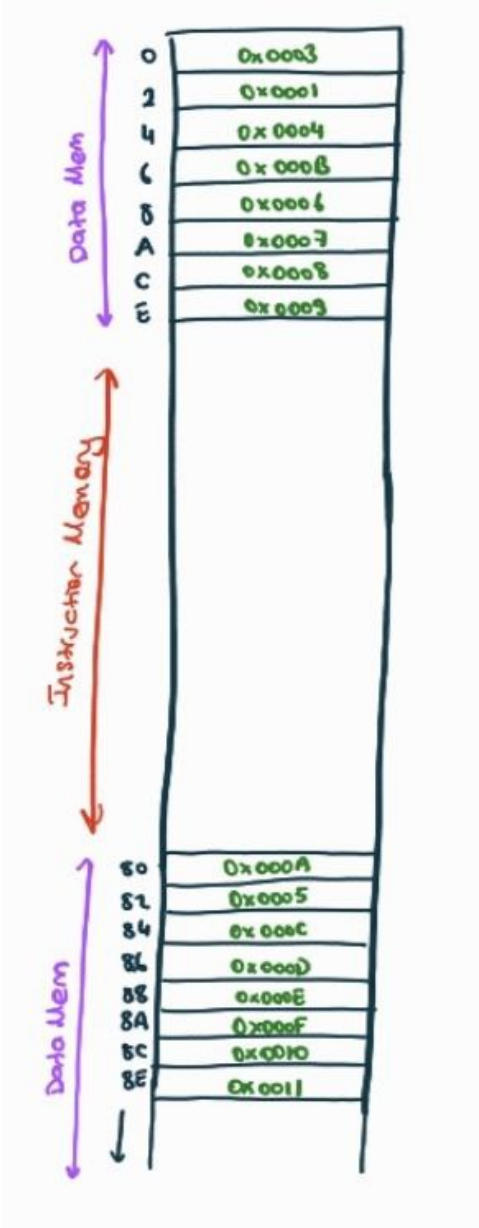
-Shift Instructions



Zeynepnur ŞAHİNEL
2305399 05.06.2022

-Branch Instructions

Instruction/Data Memory Seperation



Zeynepnur ŞAHİNEL
2305399 05.06.2022

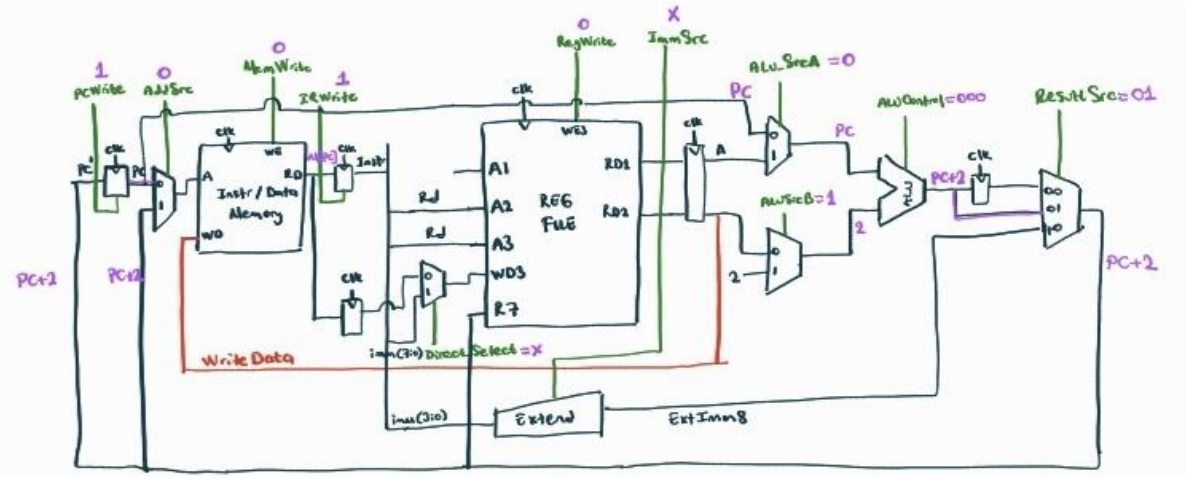
1.1 Memory/Register Transfer Instructions

-Load to register from memory

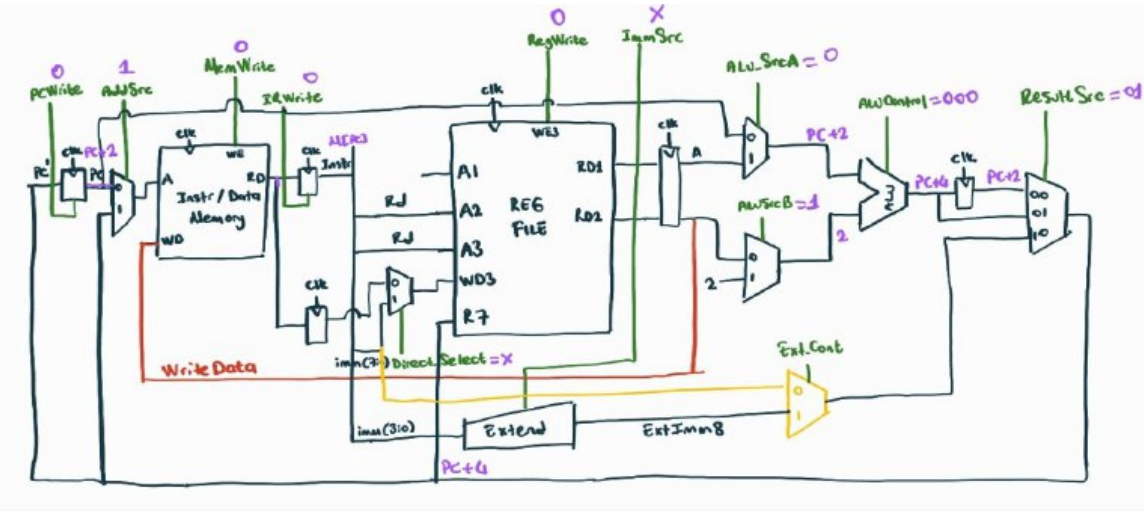
Load with direct addressing

R0 ←MEM1

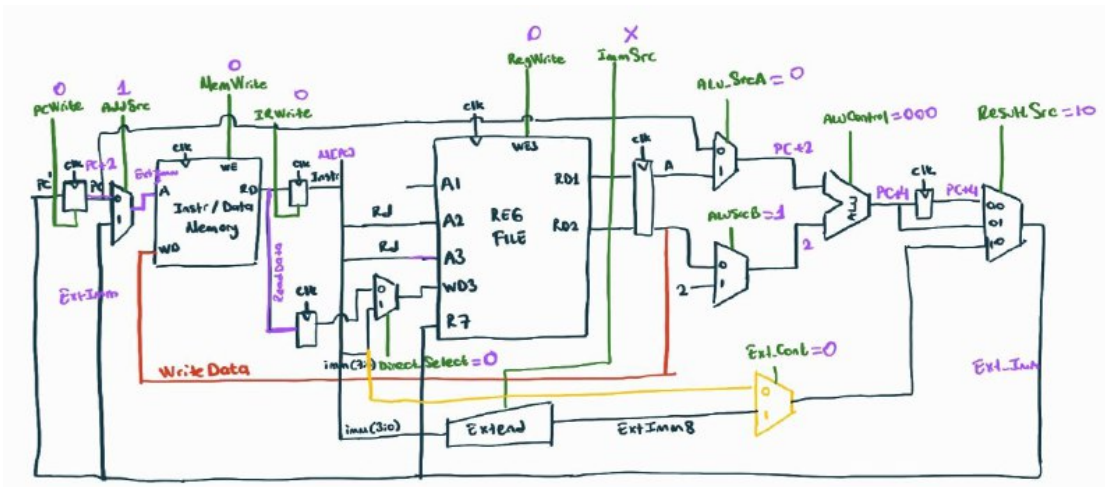
Cycle 1: Fetch



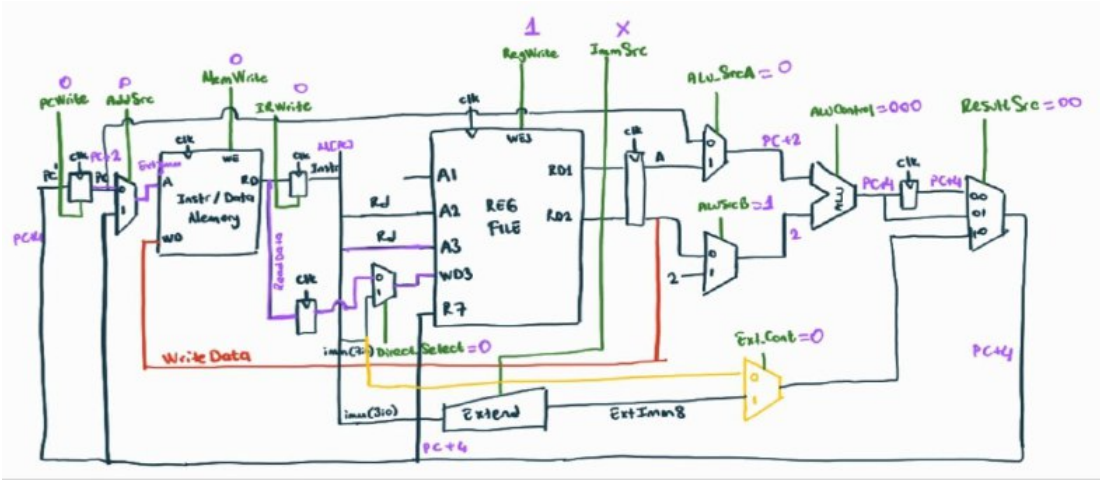
Cycle 2: Decode



Cycle 3: MemRead/Execute ALU



Cycle 4:Writeback



-Load immediate to register

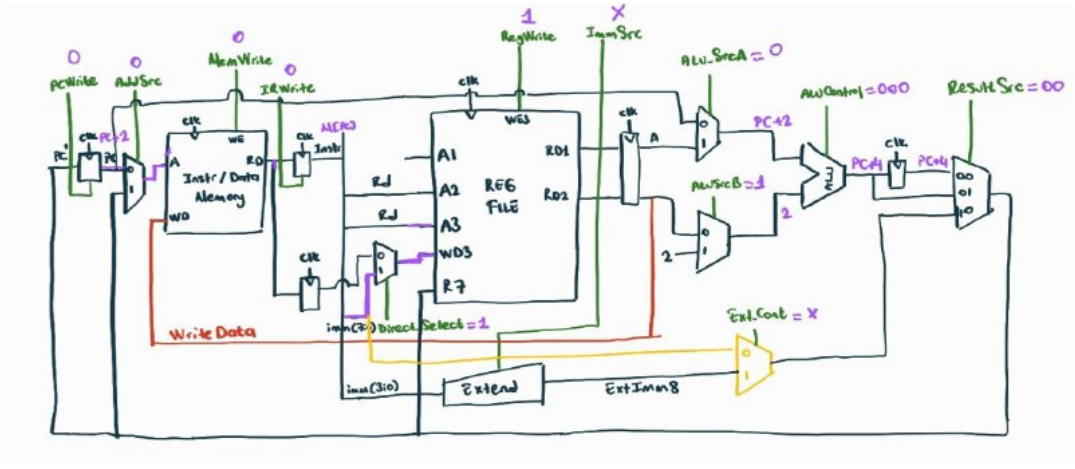
Load with immediate addressing

R3 ←DATA

Cycle 1: Fetch (same as the previous one)

Cycle 2: Decode (same as the previous one)

Cycle 3: Writeback



-Store from register to memory

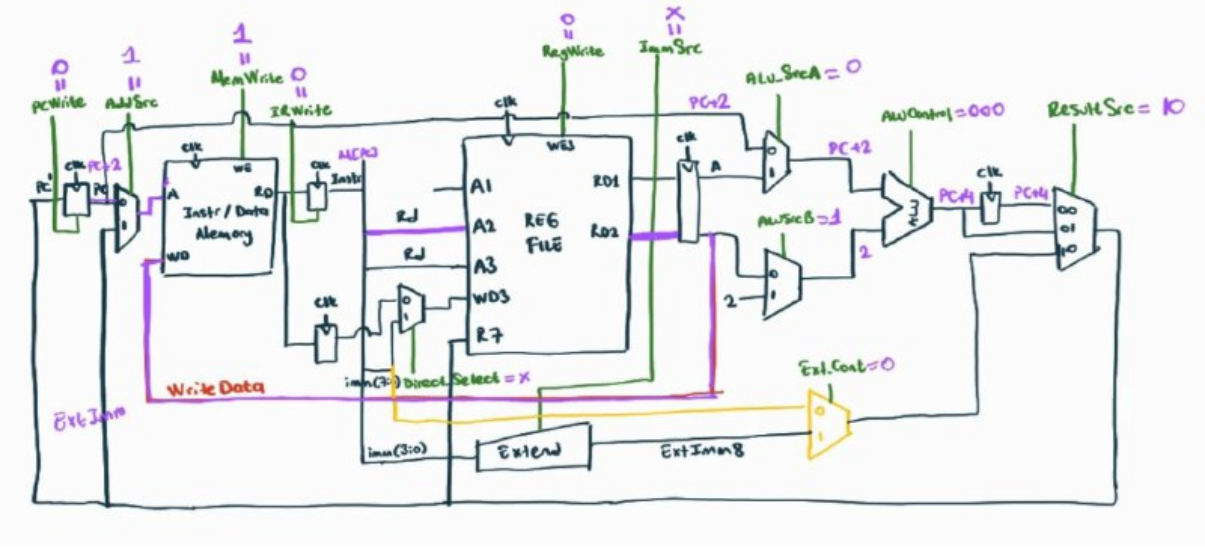
Data store into memory

MEM2 ←R2

Cycle 1: Fetch

Cycle 2: Decode

Cycle 3: MemWrite



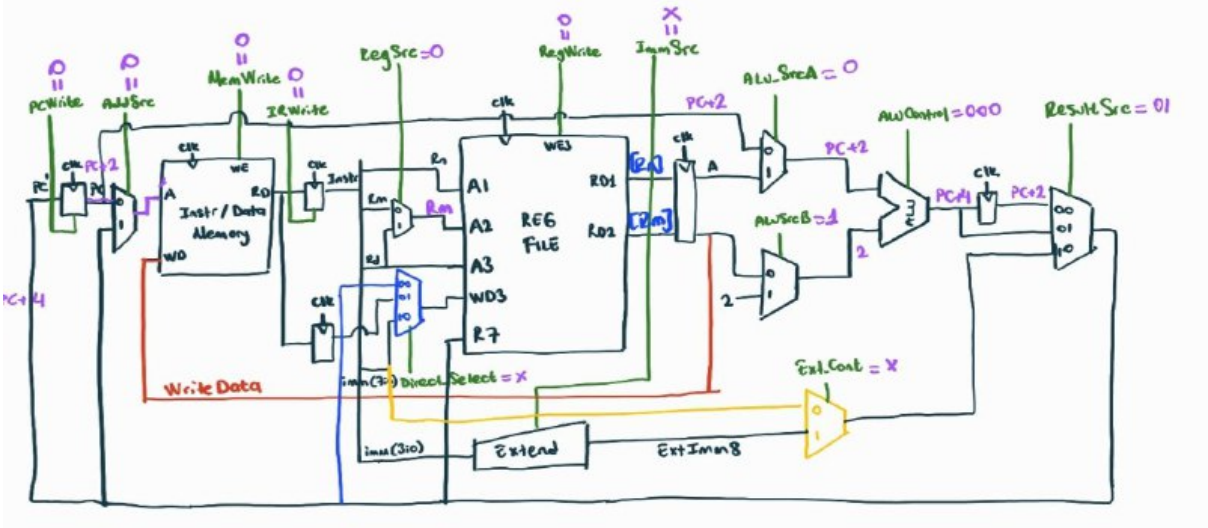
1.2 Arithmetic Instruction

-Direct Addition

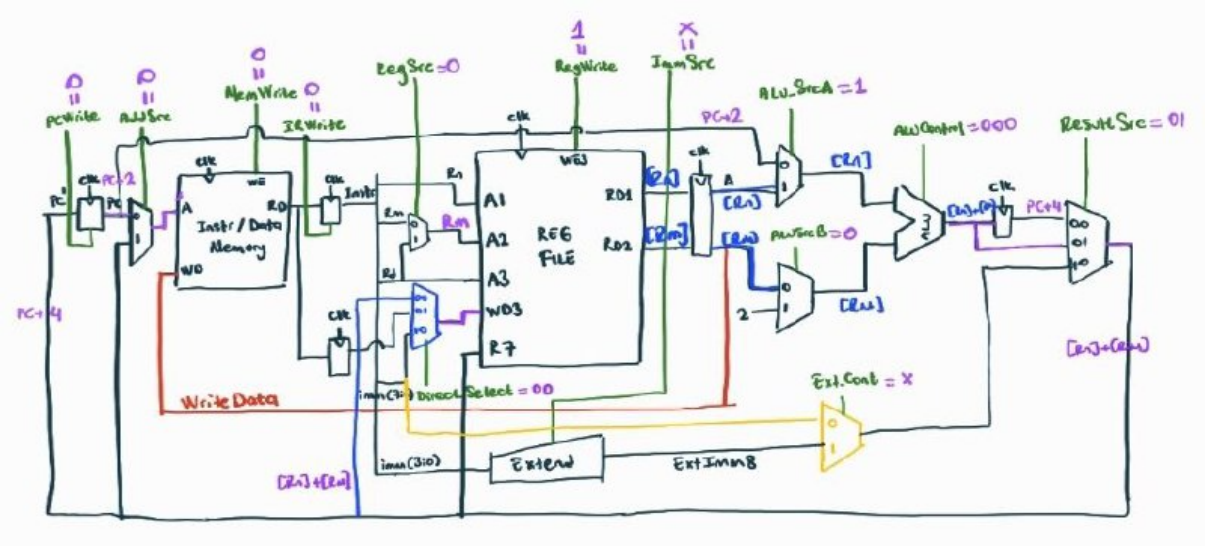
Rd←Rn+Rm

Cycle 1: Fetch

Cycle 2: Decode



Cycle 3: Execute ALU & Writeback



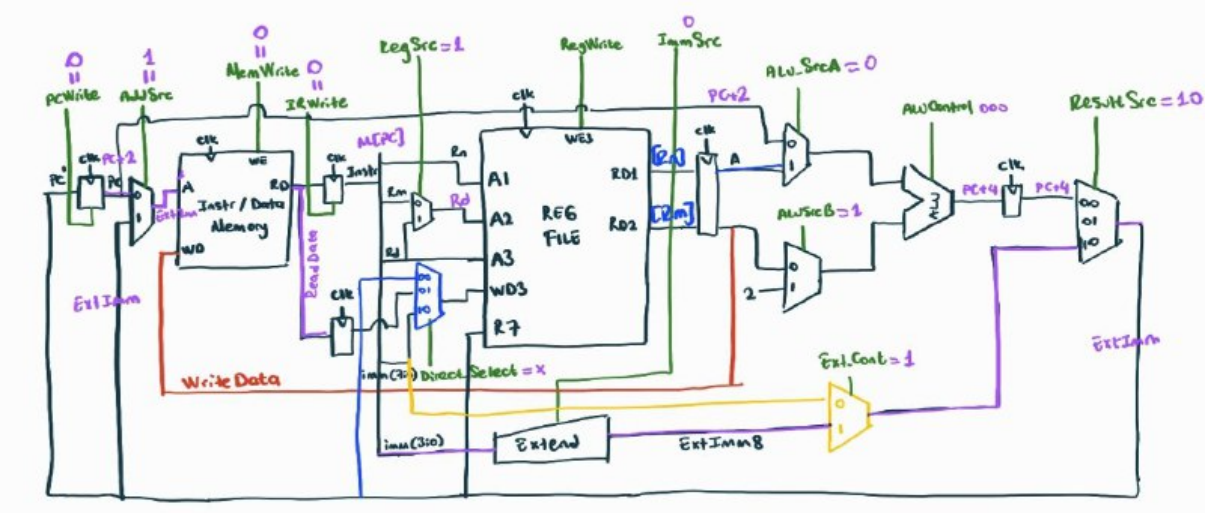
-Indirect Addition

Rd←Rn+MEM

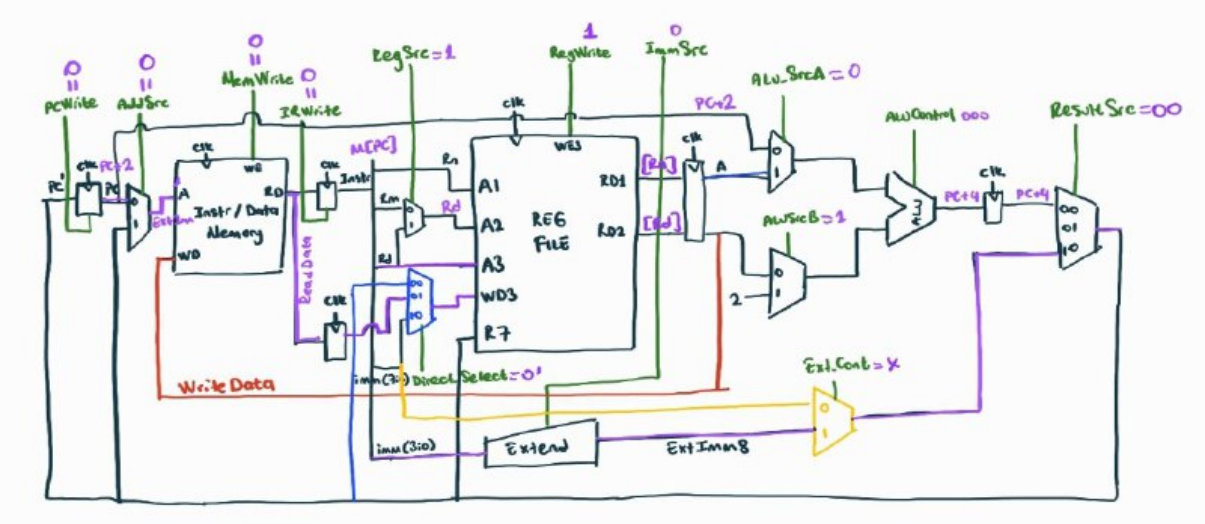
Cycle 1: Fetch

Cycle 2: Decode

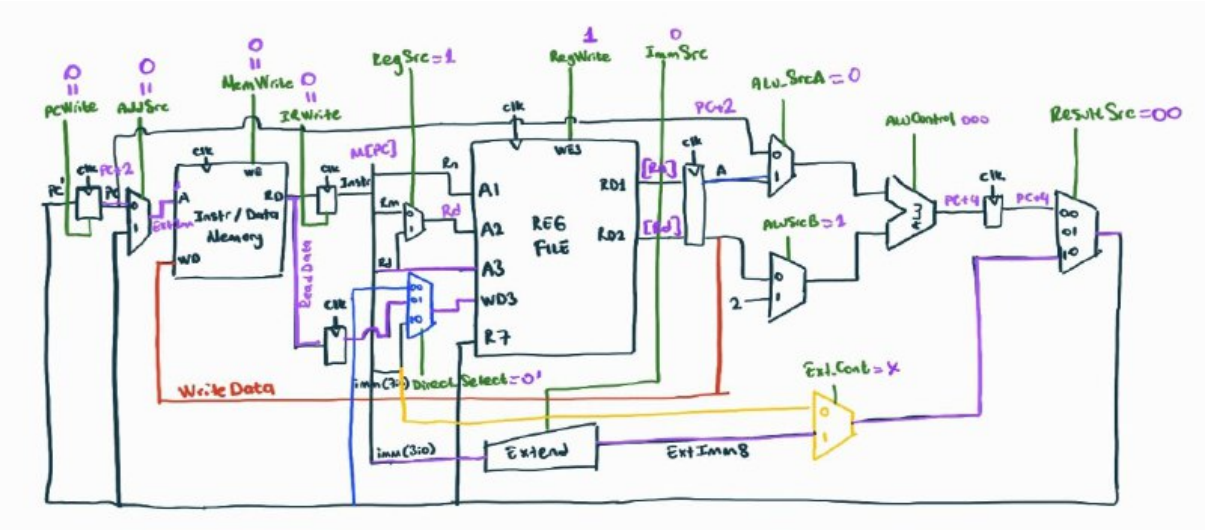
Cycle3: MemRead



Cycle 3: Writeback (1)



Cycle 4: Writeback (2)



-Subtraction: Same procedure with addition, only ALU_Control differs.
ALU_Control=001

-Indirect Subtraction: Same procedure with indirect addition, only ALU_Control differs.
ALU_Control=001

1.3 Logic Instructions

-AND: Same procedure with addition, only ALU_Control differs.
ALU_Control=100

-OR: Same procedure with addition, only ALU_Control differs.
ALU_Control=101

-XOR: Same procedure with addition, only ALU_Control differs.
ALU_Control=110

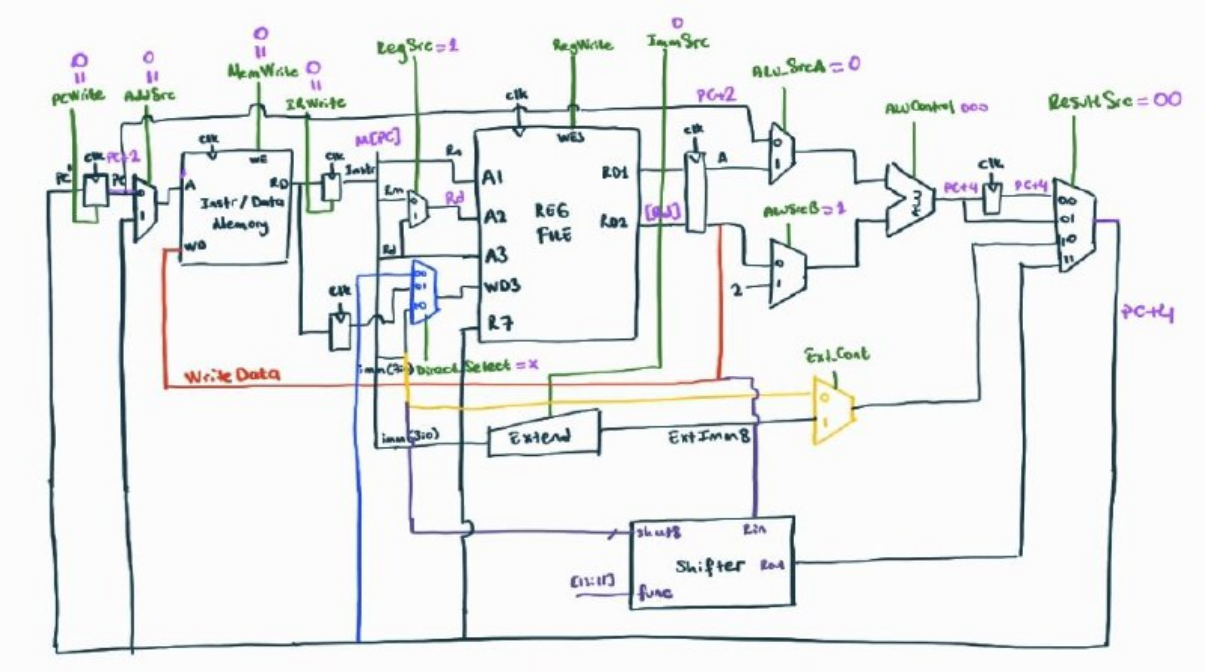
-CLEAR

1.4 Shift Instructions

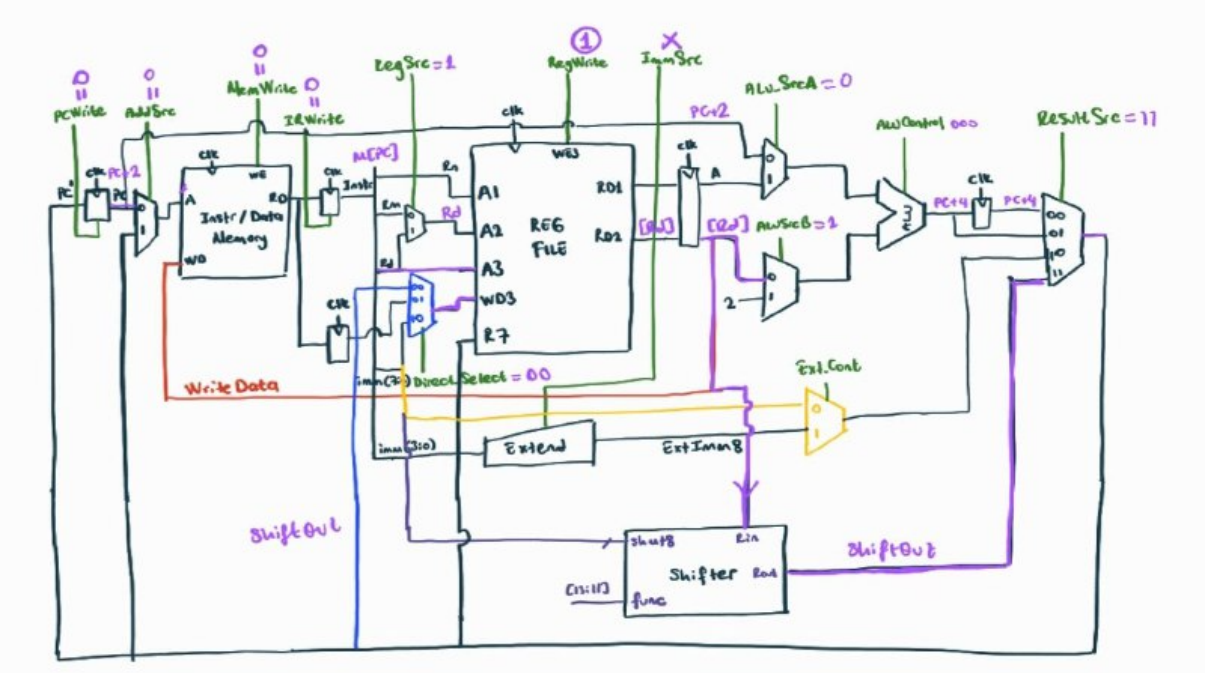
-Rotate Left (func=000)

Cycle 1: Fetch

Cycle 2: Decode



Cycle 3: Shift and Writeback

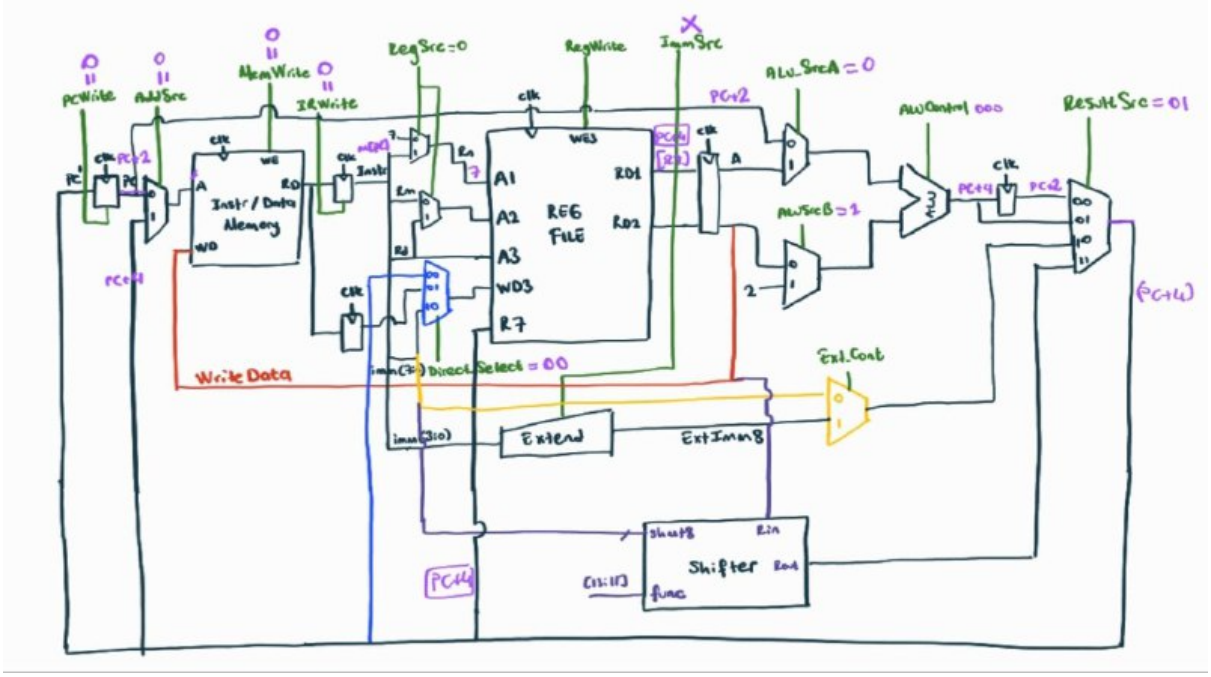


-Rotate right
func=001
-Shift left
func=010
-Arithmetic shift right
func=011
-Logical shift right
func=100

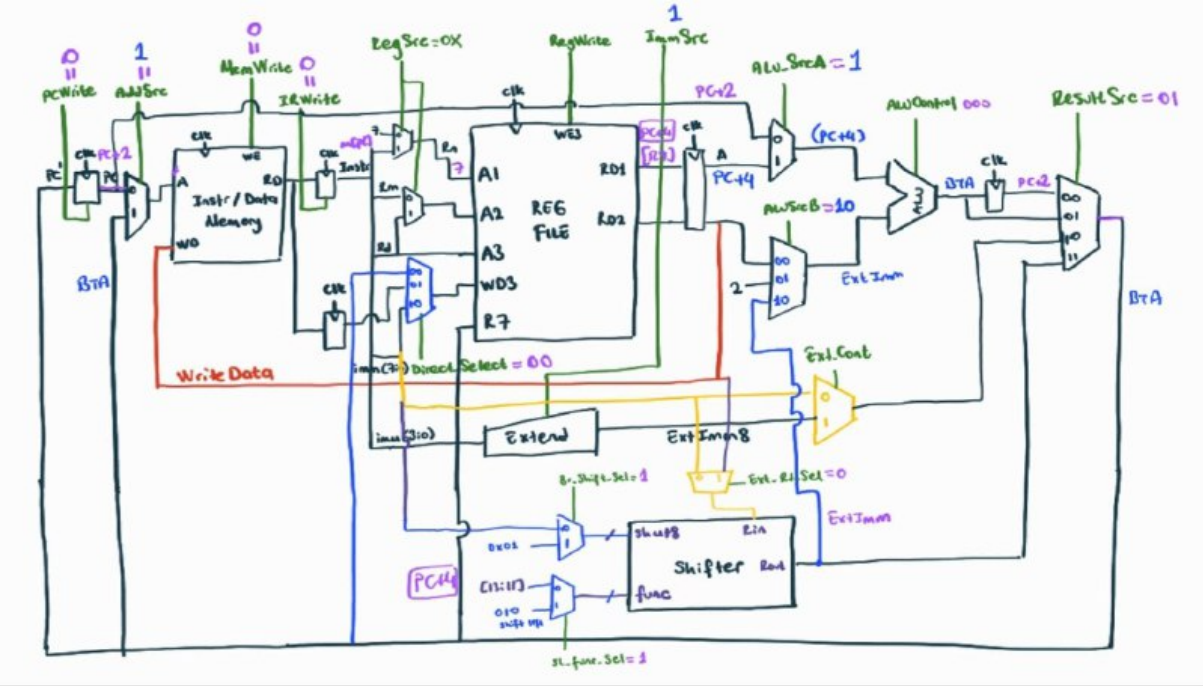
1.5 Branch Instructions

- Branch unconditional

Cycle 1: Fetch
Cycle 2: Decode



Cycle 3: Branch Execution

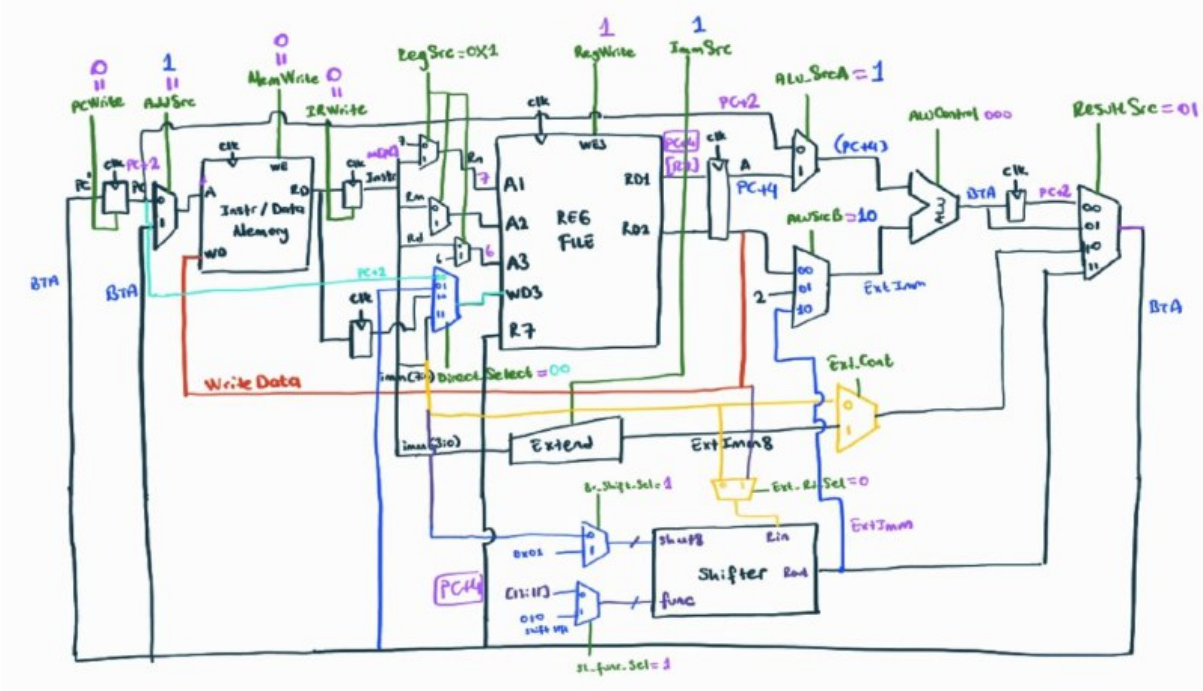


- Branch with link

Cycle 1: Fetch

Cycle 2: Decode

Cycle 3: Execution

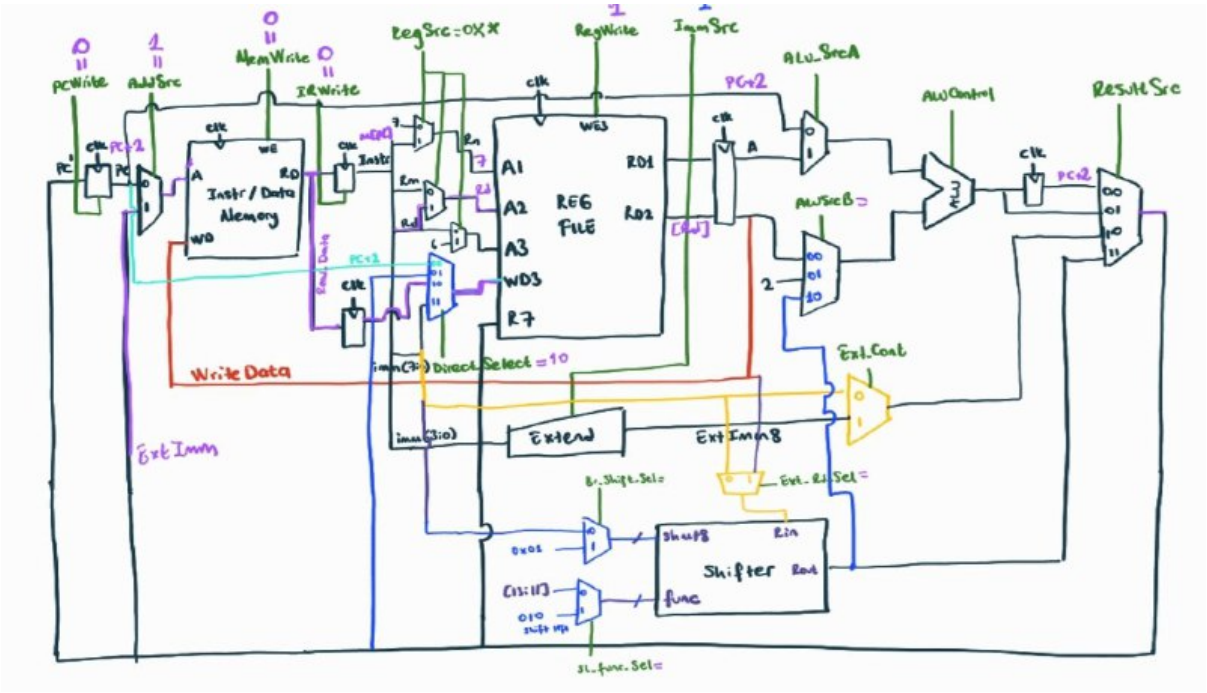


- Branch indirect

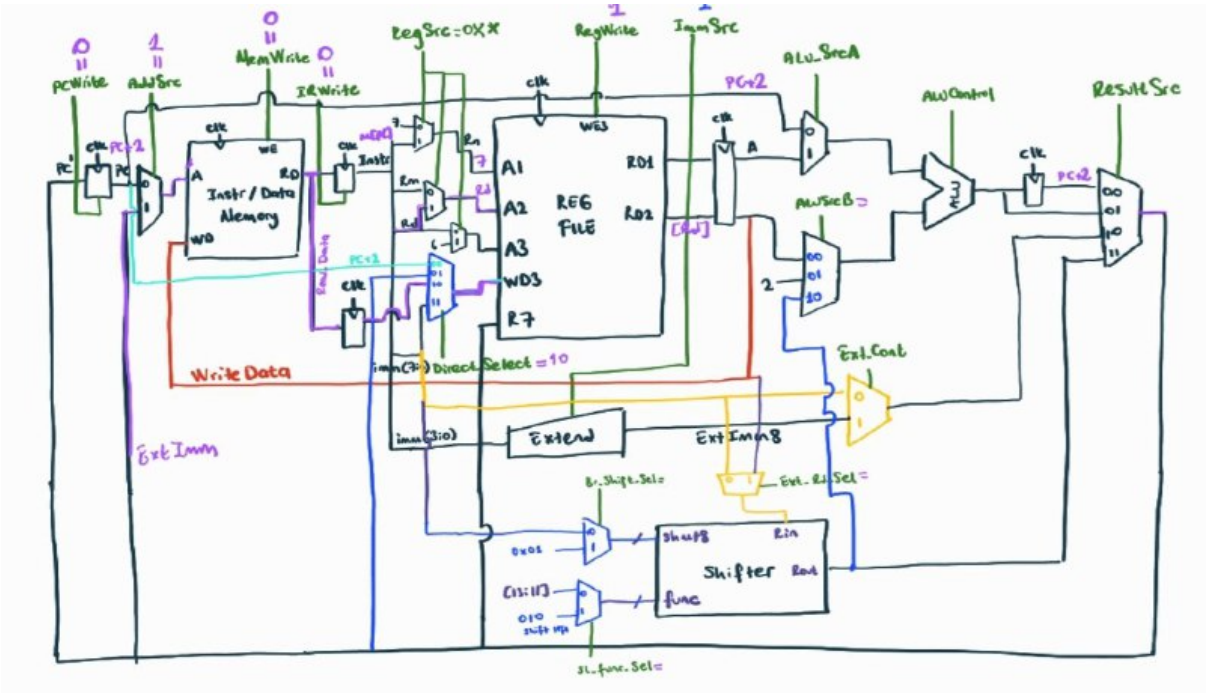
Cycle 1: Fetch

Cycle 2: Decode

Cycle 3: MemRead

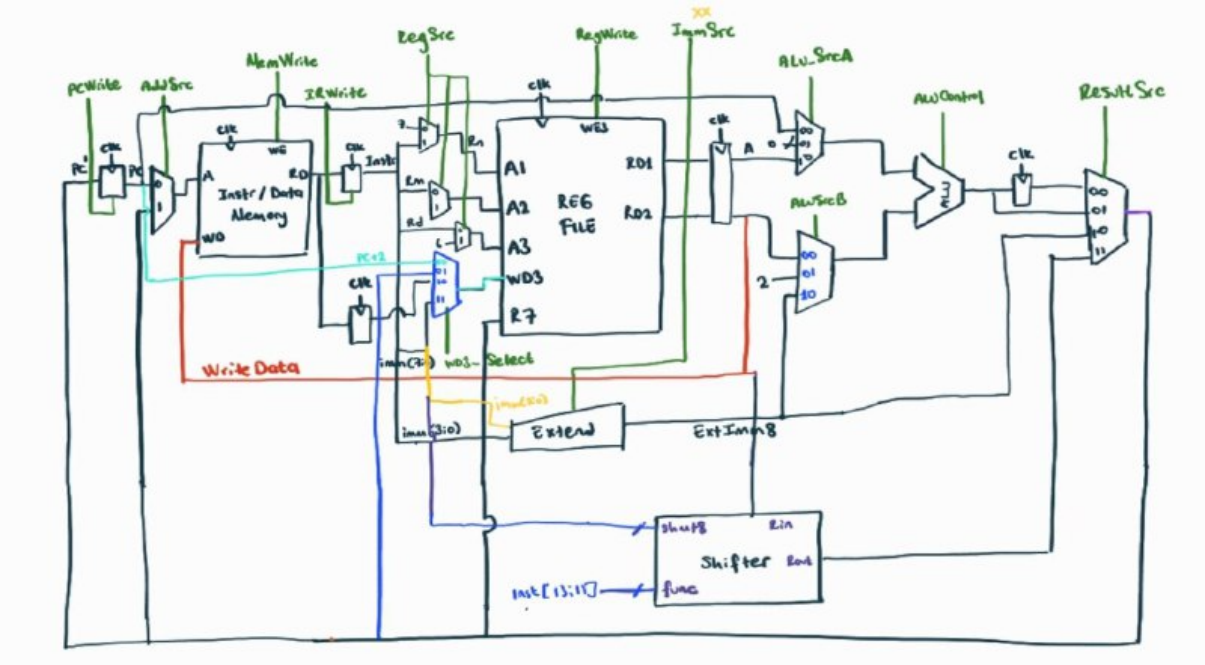


Cycle 4: Writeback



- **Branch if zero**
- **Branch if not zero**
- **Branch if carry set**
- **Branch if carry clear**

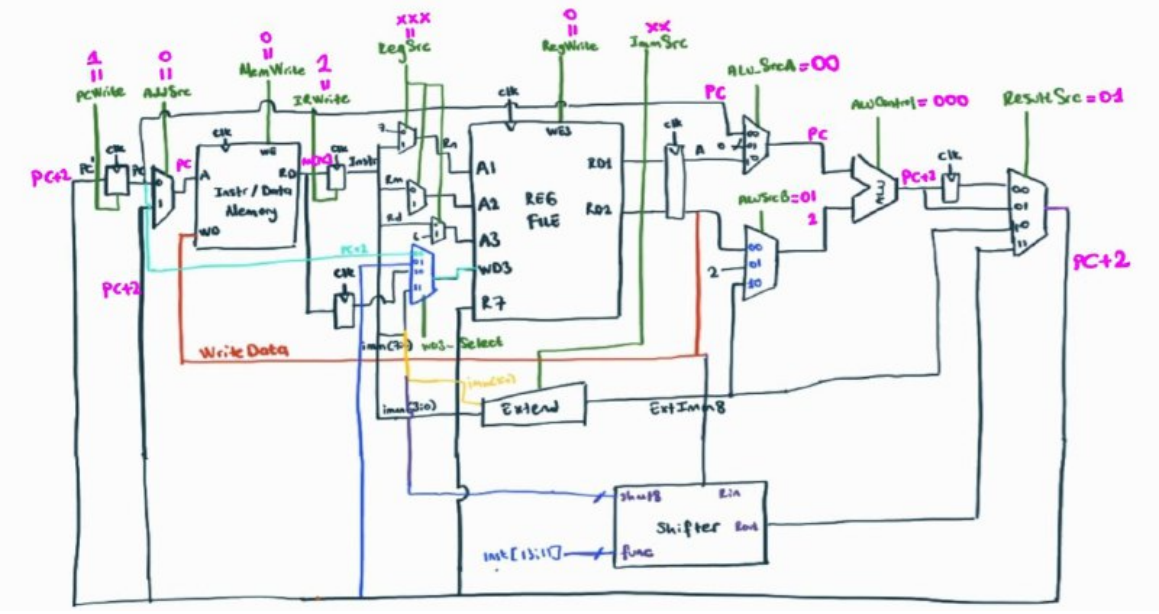
Part 2) Multi-cycle Datapath Design



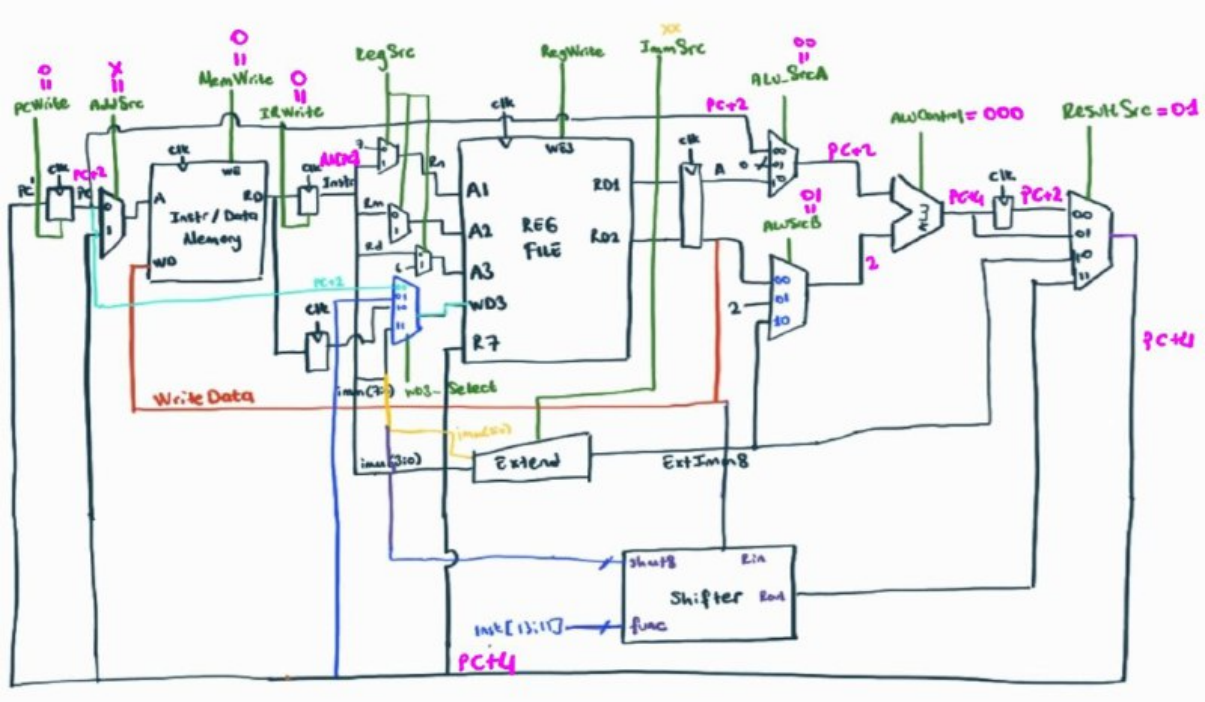
Last version of the datapath without branch conditions

Later, I realized that the logical shift left (imm8 <<1) for the BTA calculation could be handled inside the Extender. Thus, it was modified as the above figure. Extra multiplexers were destroyed.

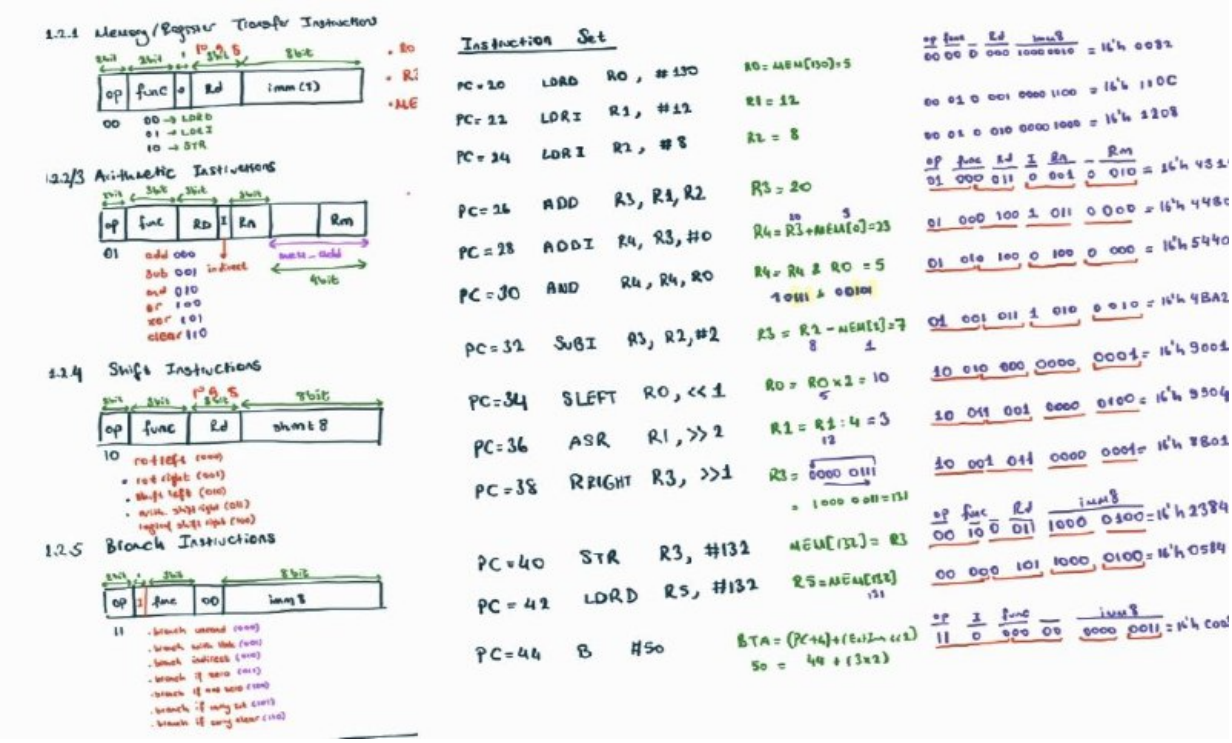
Fetch Cycle for all processes:



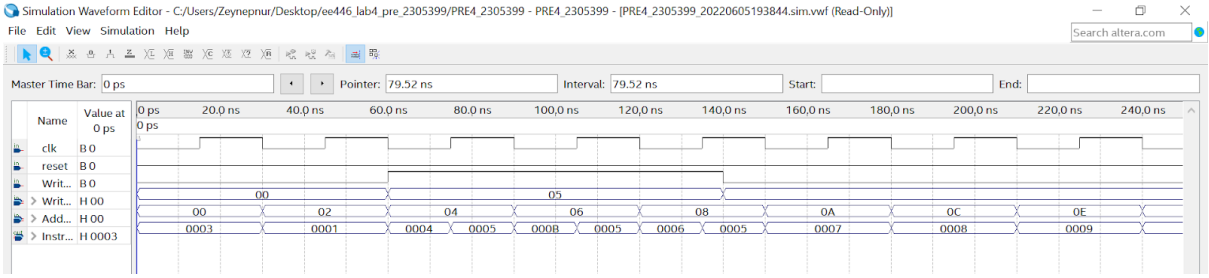
Decode Cycle for all processes



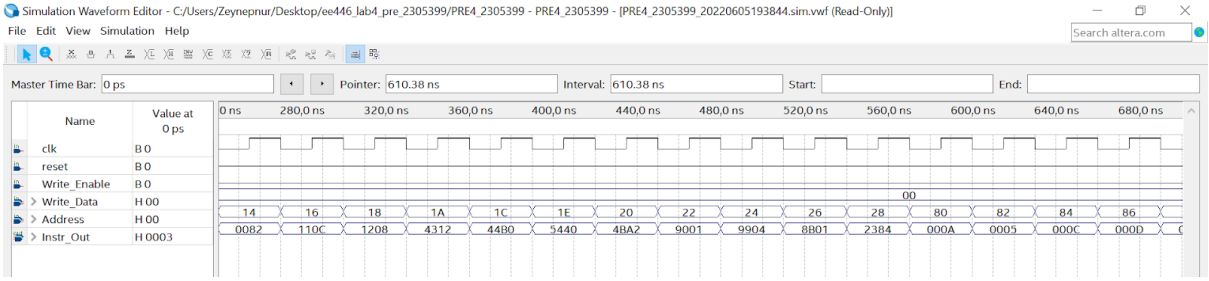
Appendix



Instruction-Data Memory Results

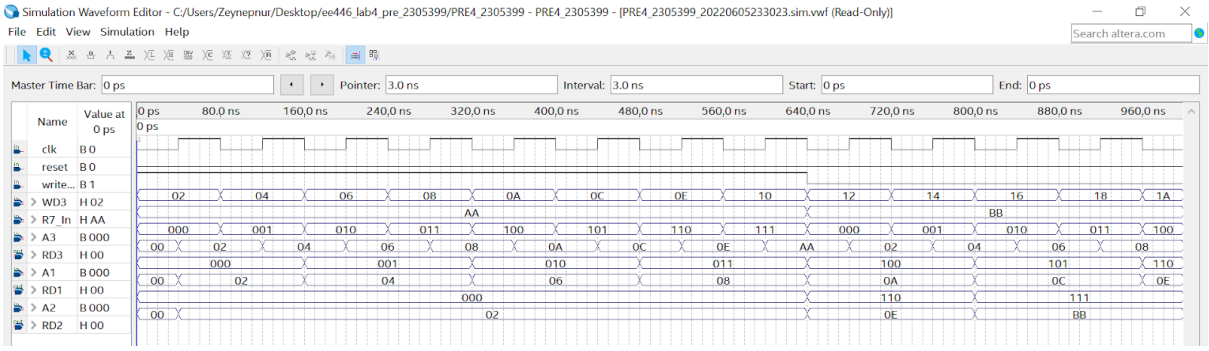


MEM[0] to MEM[14] (Data Memory)



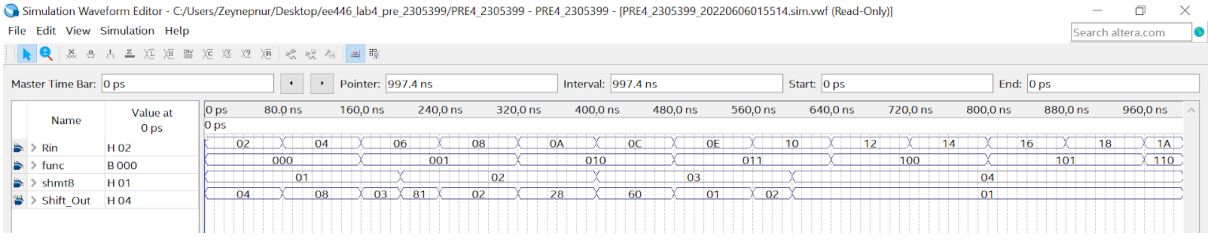
MEM[20]-MEM[28] (Instruction Memory)

Register File

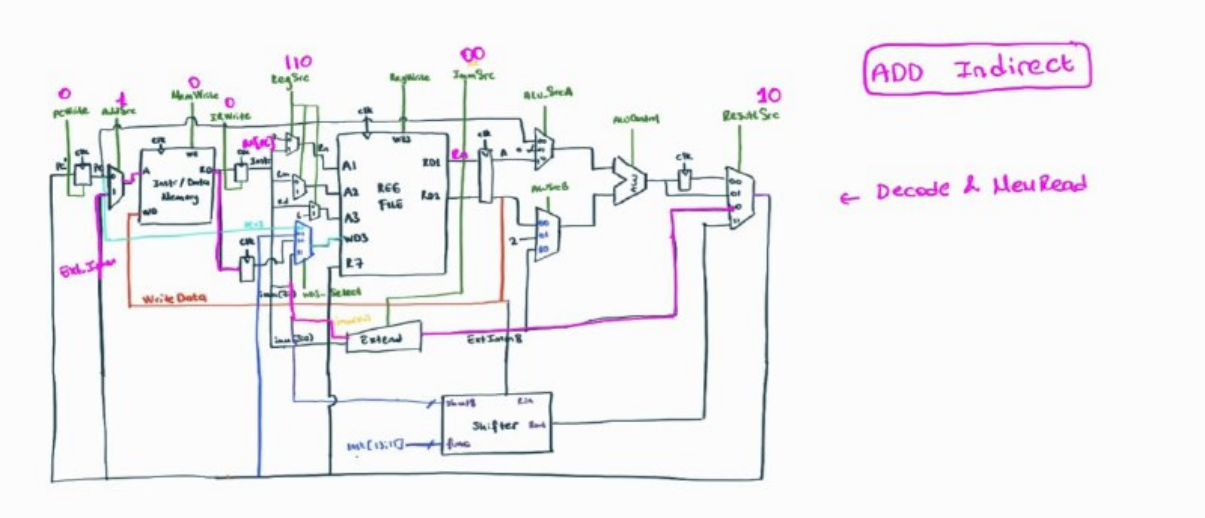


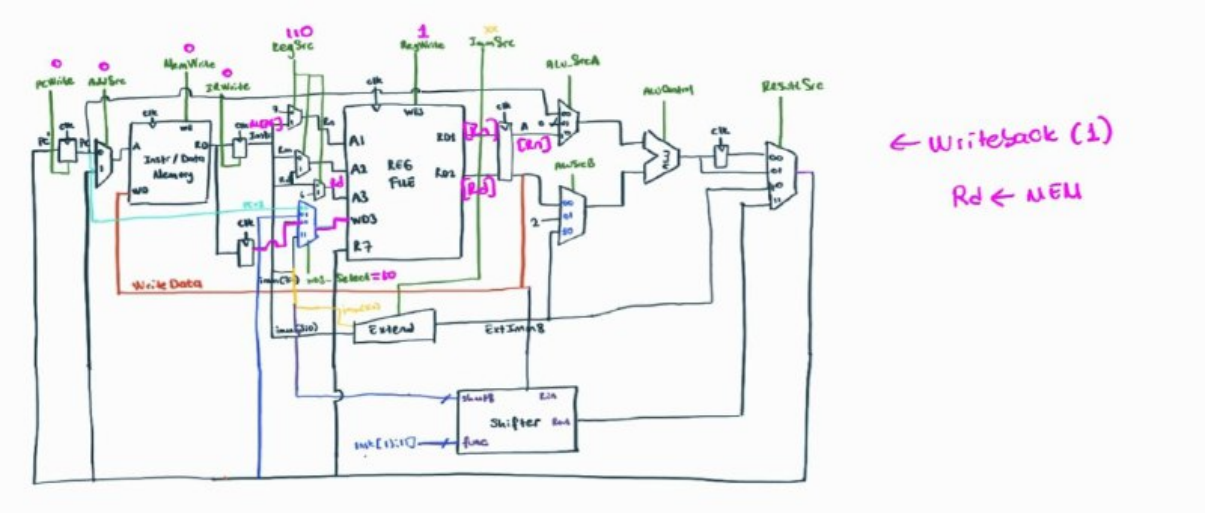
Register File Demonstration for manually given register numbers to A1, A2, A3

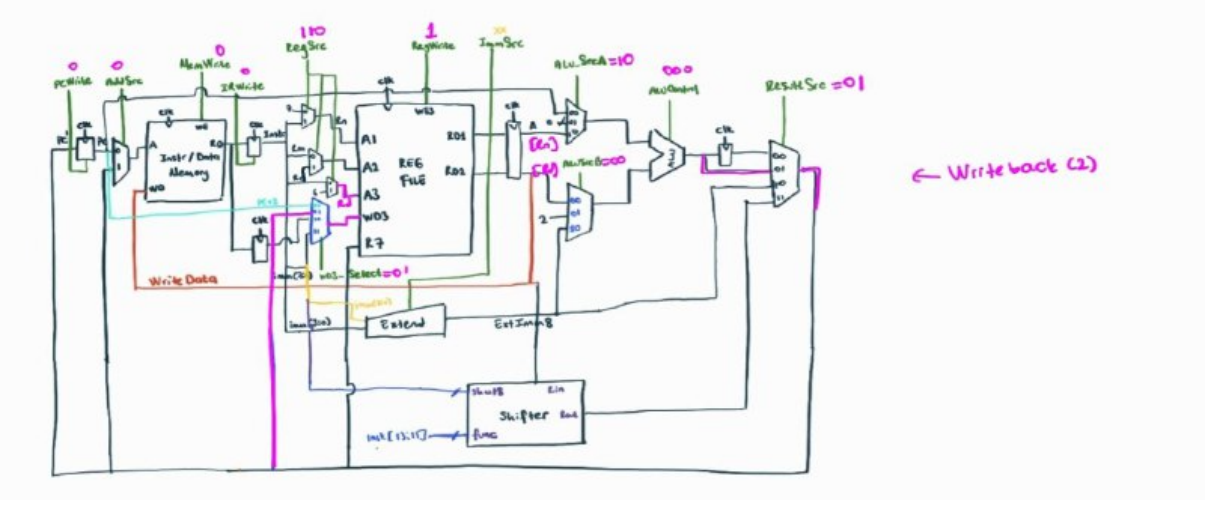
Barrel Shifter



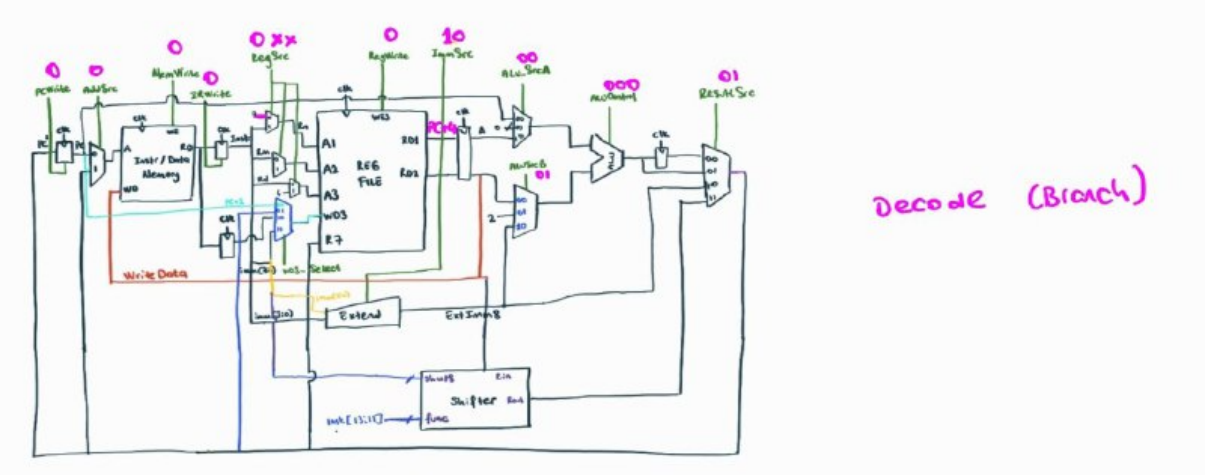
Modified Version of Indirect Addition



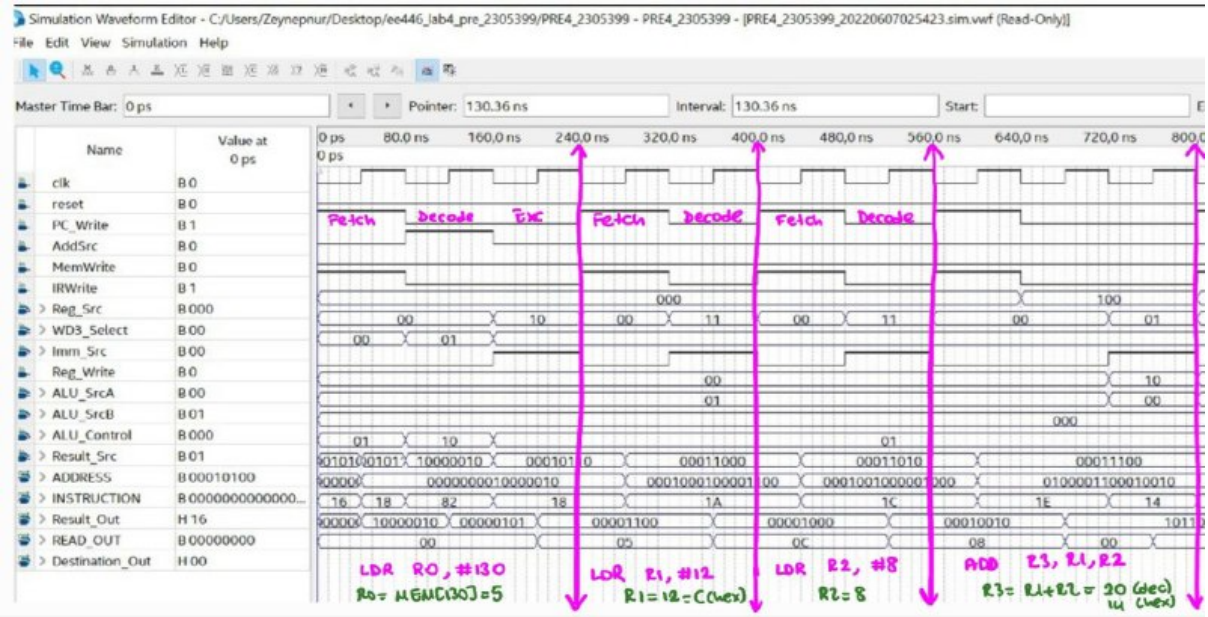


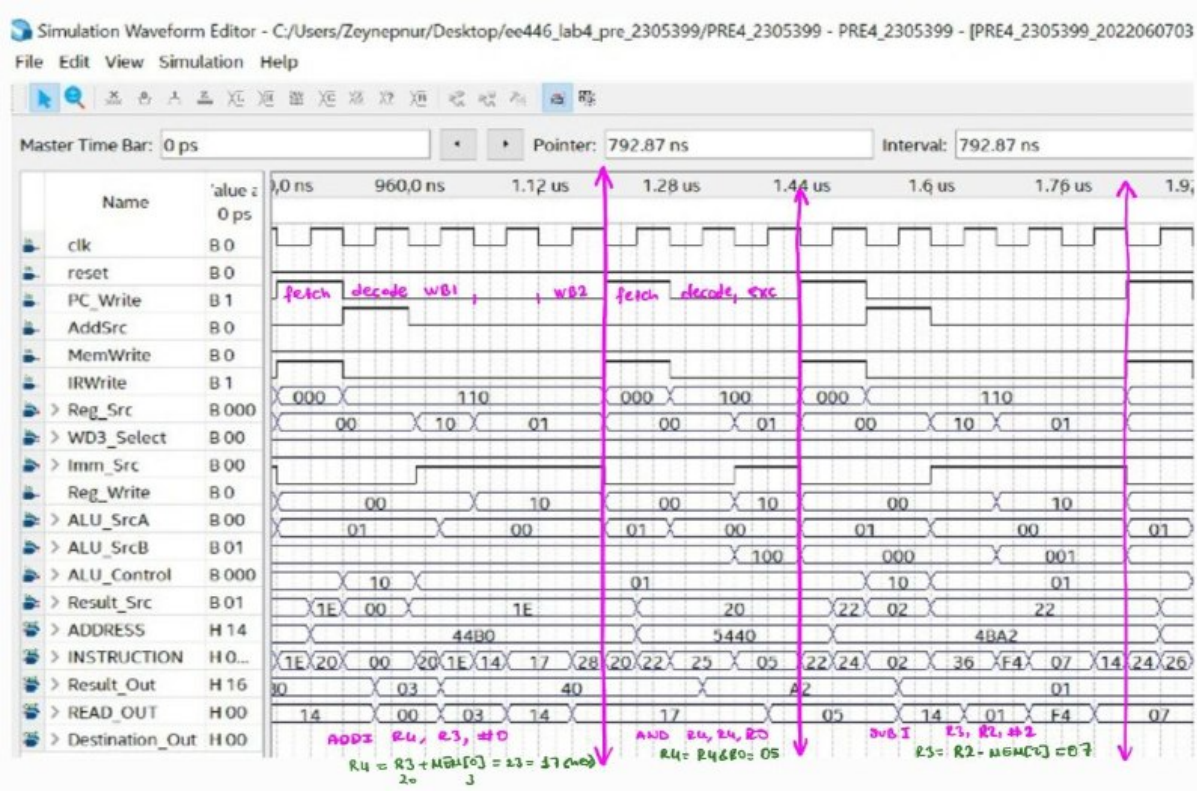


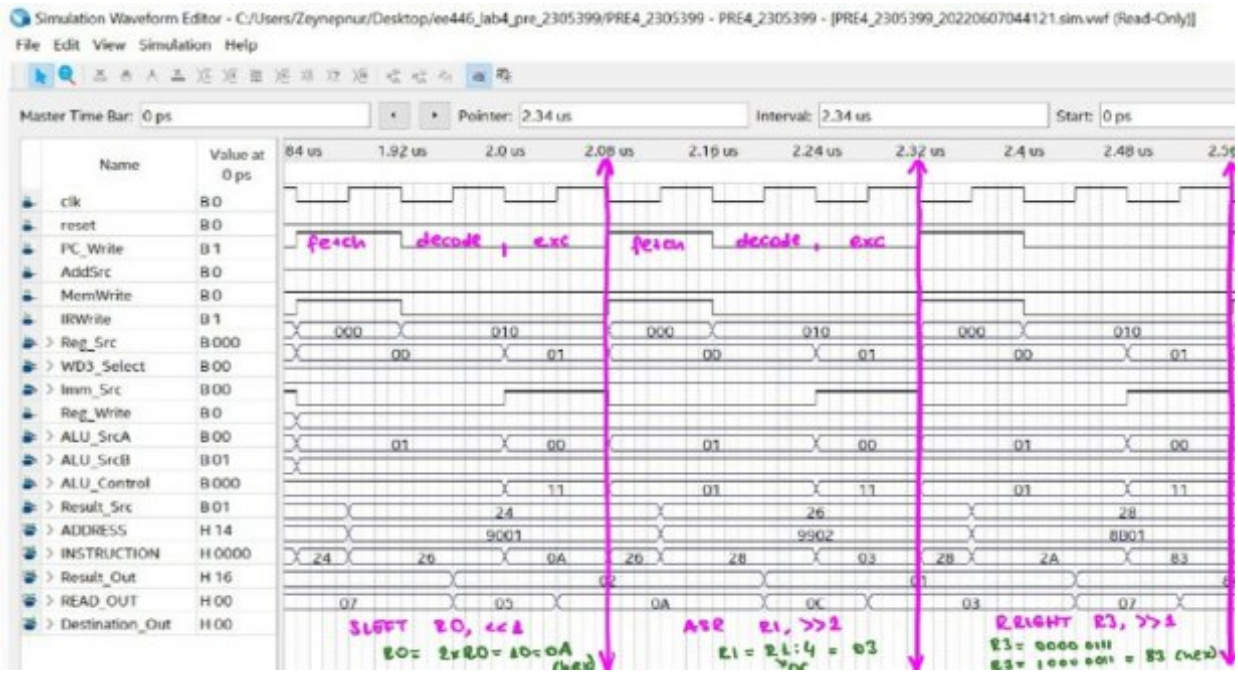
Modified Version of Branch

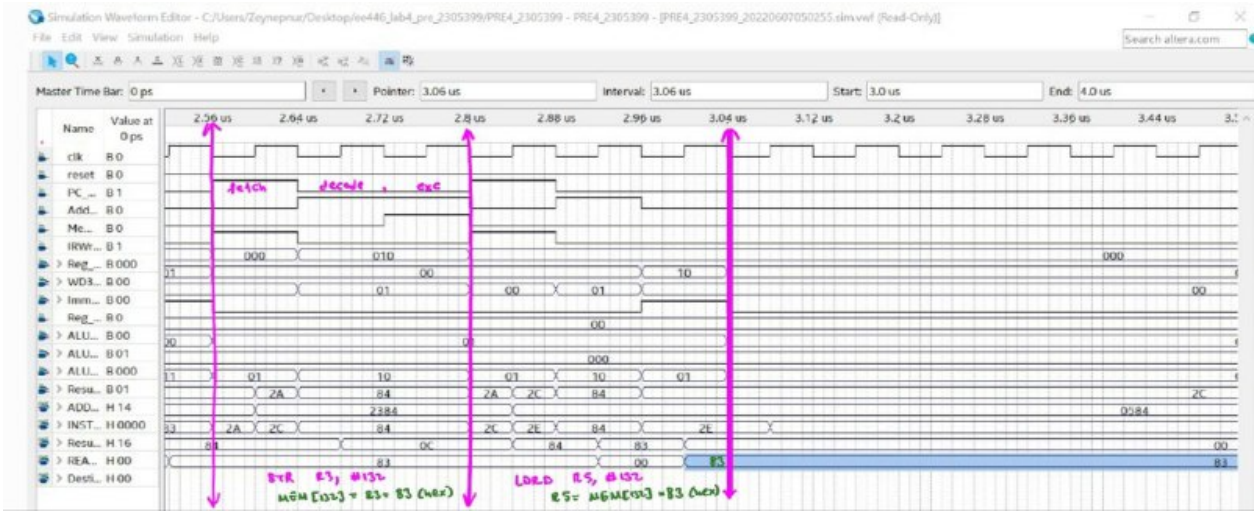


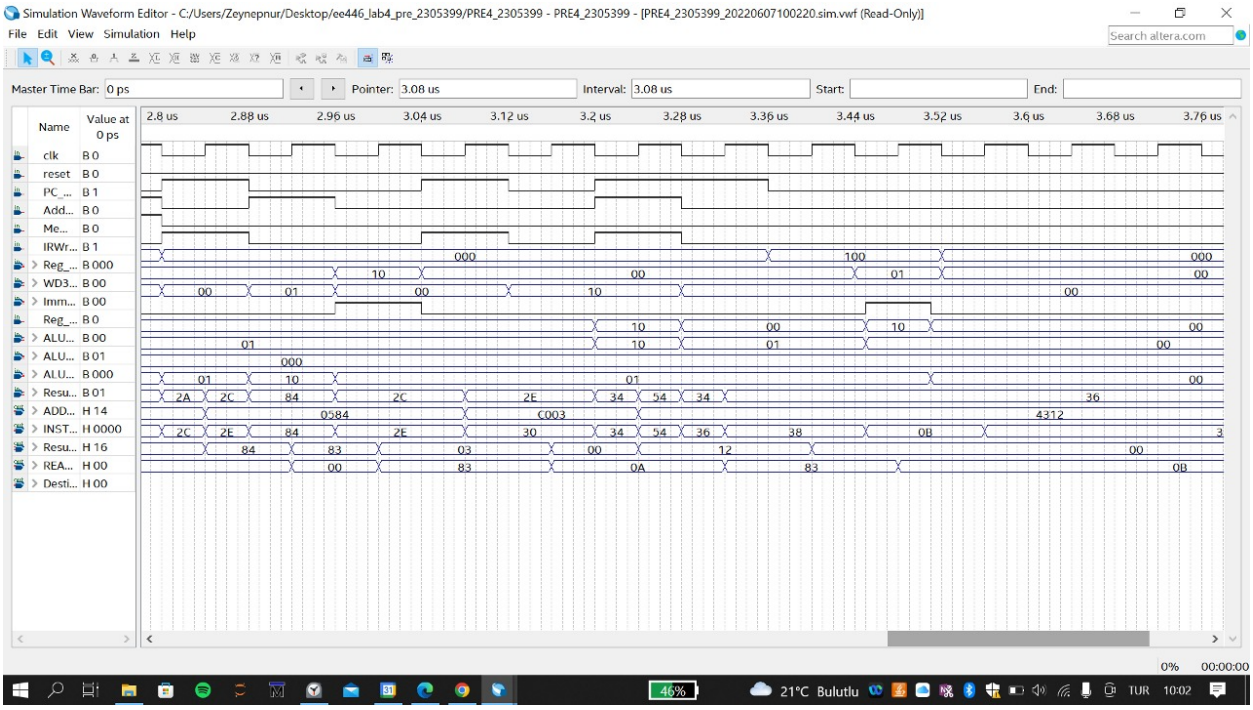
Datapath Waveforms

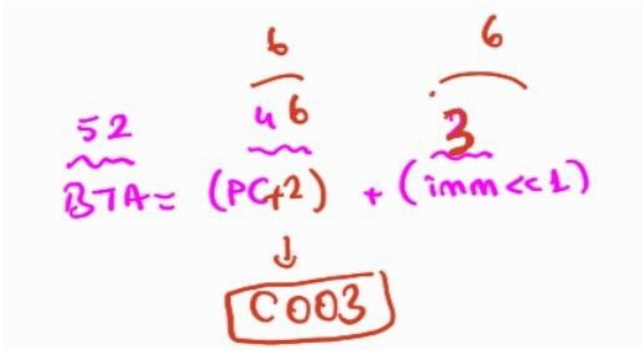



Simulation Waveform Editor - C:/Users/Zeynepnur/Desktop/ee446_lab4_pre_2305399/PRE4_2305399 - PRE4_2305399 - [PRE4_2305399_2022060703
File Edit View Simulation Help
Master Time Bar: 0 ps
Pointer: 792.87 ns
Interval: 792.87 ns
Name
clk
reset
PC_Write
AddSrc
MemWrite
IRWrite
Reg_Src
WD3_Select
Imm_Src
Reg_Write
ALU_SrcA
ALU_SrcB
ALU_Control
Result_Src
ADDRESS
INSTRUCTION
Result_Out
READ_OUT
Destination_Out
fetch
decode
WB1
WB2
exc
ADDI R4, R3, #0
AND R4, R4, R0
SUBI R3, R2, #2


Simulation Waveform Editor - C:/Users/Zeynepnur/Desktop/ee446_lab4_pre_2305399/PRE4_2305399 - PRE4_2305399 - [PRE4_2305399_20220607044121.sim.vwf (Read-Only)]
File Edit View Simulation Help
Master Time Bar: 0 ps
Pointer: 2.34 us
Interval: 2.34 us
Start: 0 ps
Name
clk
reset
PC_Write
AddSrc
MemWrite
IRWrite
Reg_Src
WD3_Select
Imm_Src
Reg_Write
ALU_SrcA
ALU_SrcB
ALU_Control
Result_Src
ADDRESS
INSTRUCTION
Result_Out
READ_OUT
Destination_Out
fetch
decode
exc
SLEFT R0, <<1
ASR R1, >>1
RIGHT R3, >>1


Simulation Waveform Editor - C:/Users/Zeynepnur/Desktop/ee446_lab4_pre_2305399/PRE4_2305399 - PRE4_2305399 - [PRE4_2305399_20220607050255.sim.vwf (Read-Only)]
Master Time Bar: 0 ps
Pointer: 3.06 us
Interval: 3.06 us
Start: 3.0 us
End: 4.0 us
fetch
decode
exe
BTA= (PC+2) + (imm<<1)
C003


Simulation Waveform Editor - C:/Users/Zeynepnur/Desktop/ee446_lab4_pre_2305399/PRE4_2305399 - PRE4_2305399 - [PRE4_2305399_20220607100220.sim.vwf (Read-Only)]
Master Time Bar: 0 ps
Pointer: 3.08 us
Interval: 3.08 us
Start:
End:


52
46
3
BTA= (PC+2) + (imm<<1)
C003