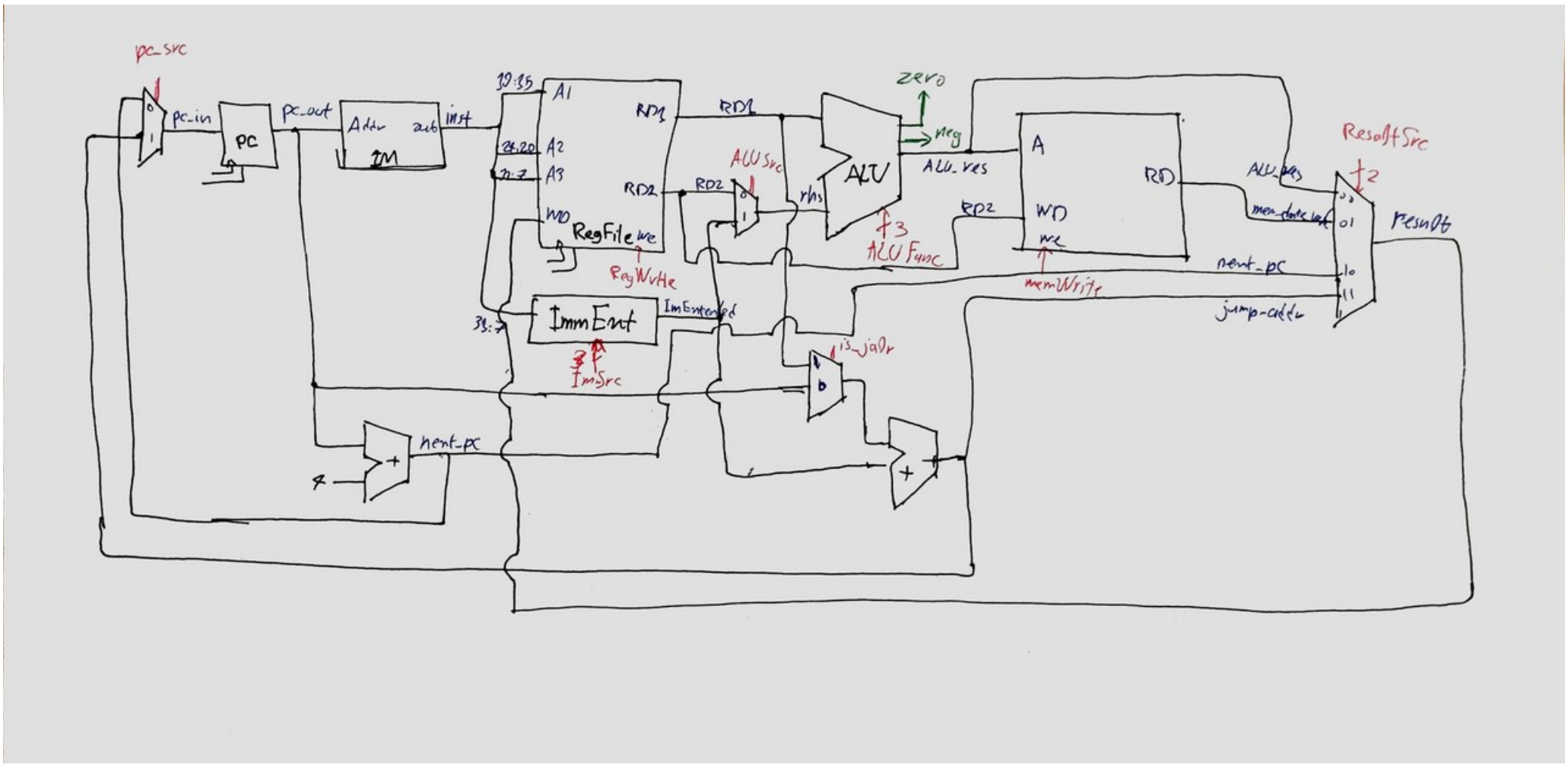
Computer Assignment 2

RISC-V single cycle implementation

Full Name	Student ID
Kourosh Alinaghi	810101476
Arshia Motaghian	810101503

Datapath

The design of the datapath:



Controller

Controller has two modules:

Main Controller

This module issues all of the controlling signals that are directly ued in datapath, except for `ALU_func`. It also has a `ALU_op` output which is used in ALU decoder to generate `ALU_func`.

Instruction type	`PC_src`	`reg_write`	`ALU_src`	`mem_write`	`result_src`	`imm_src`	`ALU_op`	`is_jalr`
Load word	0	1	1	0	01	`I_TYPE`	`ADD_ANYWAY`	0
Store word	0	0	1	1	00	`S_TYPE`	`ADD_ANYWAY`	0
Branch	depends on `f3`, `neg` and `zero`	0	0	0	00	`B_TYPE`	`SUB_ANYWAY`	0
I-Type	0	1	1	0	00	`I_TYPE`	`I_TYPE`	0
Load upper immidiate	0	1	1	0	11	`U_TYPE`	`I_TYPE`	0
JAL	1	1	0	0	10	`I_TYPE`	`ADD_ANYWAY`	0
JALR	1	1	1	0	10	`I_TYPE`	`ADD_ANYWAY`	1

Branch `PC_src`

When the inctruction type is `B_TYPE`, the value of `PC_src` is determines using `zero` and `neg` inputs from datapath like this:

Instruction	`PC_src`
`beq`	`zero`
`bne`	`~zero`
`blt`	`neg`
`bqe`	`~neg or zero`

ALU Decoder

This modules determines the value of `ALU_func` based on `ALU_op`, `f3` and `f7` inputs.

`ALU_OP`	`f7`	`f3`	`ALU_func`
`ADD_ANYWAY`	_	_	`ADD_FUNC`
`SUB_ANYWAY`	_	_	`SUB_FUNC`
`R-TYPE`	`0000000`	`000`	`ADD_FUNC`
`R-TYPE`	`0100000`	`000`	`SUB_FUNC`
`R-TYPE`	`0000000`	`111`	`AND_FUNC`
`R-TYPE`	`0000000`	`110`	`OR_FUNC`
`R-TYPE`	`0000000`	`010`	`SLT_FUNC`
`R-TYPE`	`0000000`	`011`	`SLTU_FUNC`
`I-TYPE`	_	`000`	`ADD_FUNC`
`I-TYPE`	_	`100`	`XOR_FUNC`
`I-TYPE`	_	`110`	`OR_FUNC`
`I-TYPE`	_	`010`	`SLT_FUNC`
`I-TYPE`	_	`011`	`SLTU_FUNC`