

Project Helper file

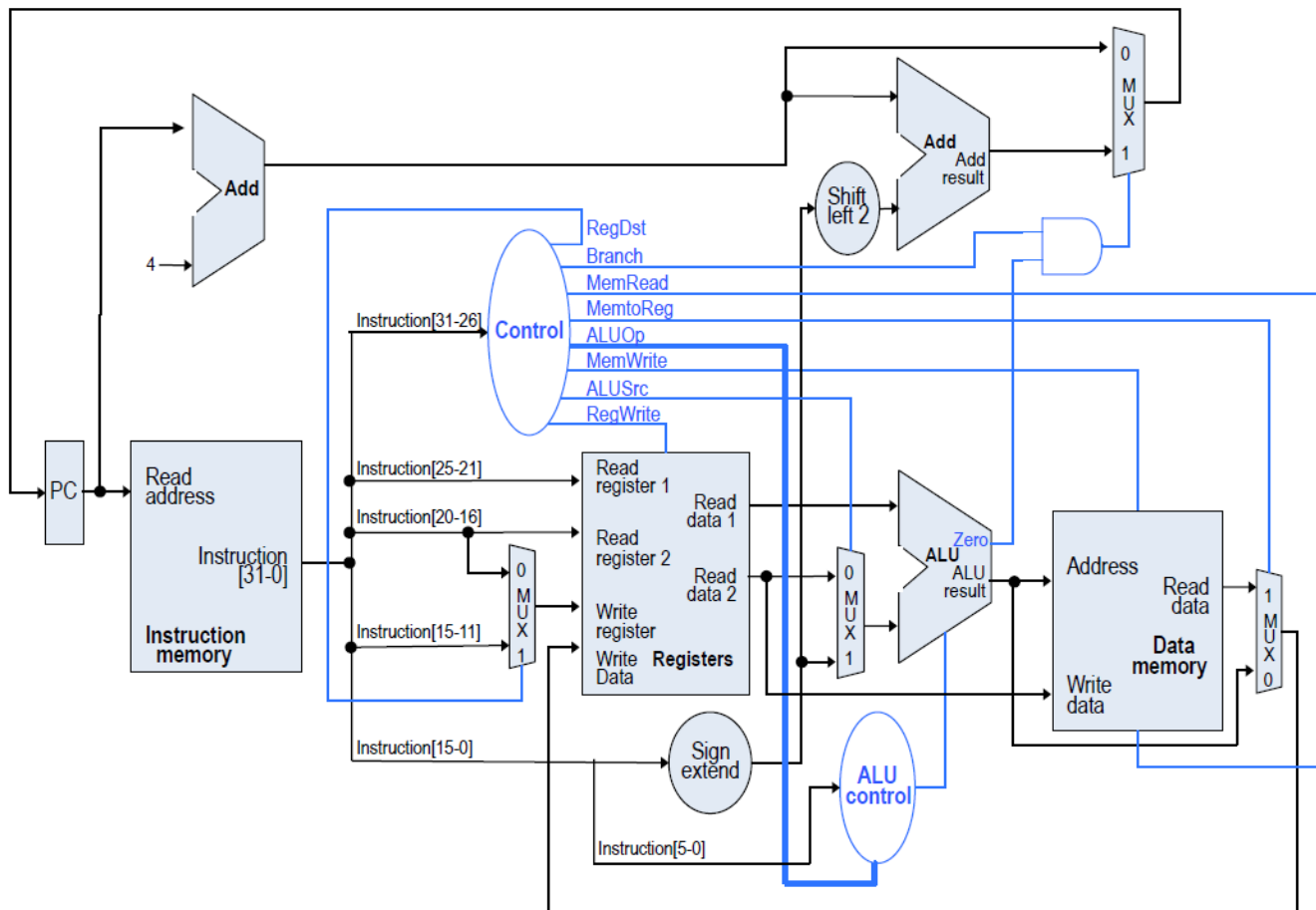
Instructions opcodes:

Type	Instruction	Opcode	Funct
R-Type	ADD	000000	100000
R-Type	SUB	000000	100010
R-Type	AND	000000	110110
R-Type	OR	000000	110111
R-Type	XOR	000000	111000
I-Type	ADDI	001000	
I-Type	SUBI	100100	
I-Type	LW	100011	
I-Type	SW	101011	
I-Type	BEQ	000100	

Control signal table:

Operation	RegDst	RegWrite	ALUSrc	ALUOp	MemWrite	MemRead	MemToReg	branch
ADD	1	1	0	010	0	0	0	0
SUB	1	1	0	110	0	0	0	0
AND	1	1	0	000	0	0	0	0
OR	1	1	0	001	0	0	0	0
XOR	1	1	0	011	0	0	0	0
LW	0	1	1	010	0	1	1	0
SW	X	0	1	010	1	0	X	0
BEQ	X	0	0	110	0	0	X	1
ADDI	0	1	1	010	0	0	0	0
SUBI	0	1	1	110	0	0	0	0

MIPS Processor:



Notes:

- We won't implement the ALU control, you will just insert ALUOp variable as an input to the ALU directly.
- You will import the data in the instruction file included to the instruction memory.

Instruction Format:

R-type	op	rs	rt	rd	shamt	funct
--------	----	----	----	----	-------	-------

Arithmetic instruction format

I-type	op	rs	rt	address/immediate
--------	----	----	----	-------------------

Transfer, branch, immediate.

Field size	6 bits	5bits	5bits	5bits	5bits	6 bits
------------	--------	-------	-------	-------	-------	--------