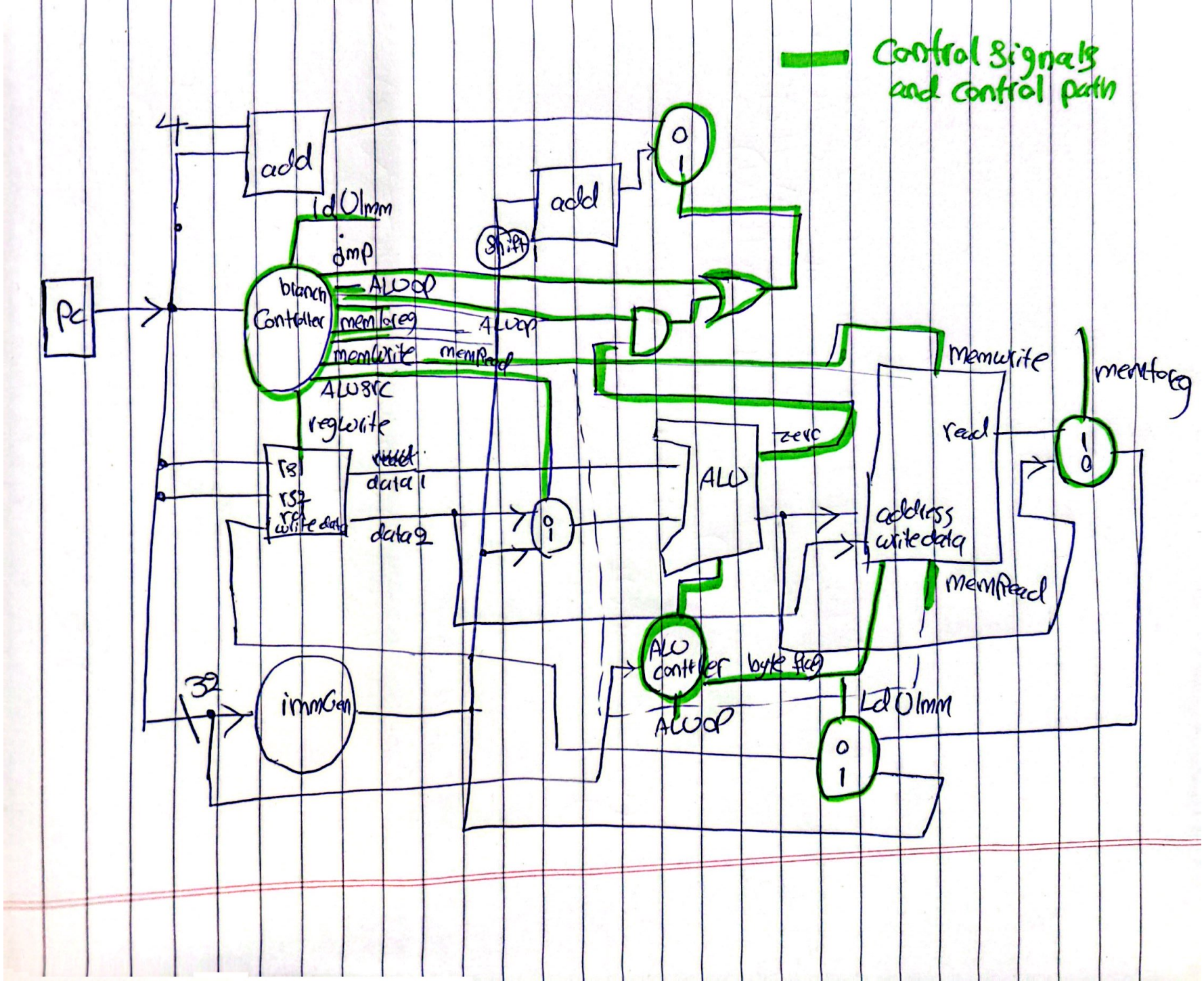
Control Signals and control path
add
4
ldUImm
jmp
branch Controller
ALUop
memToReg
memWrite
ALUSrc
regWrite
Pc
add
shift
0
1
D
memWrite
memToReg
read
address
writedata
memRead
zero
ALU
rs1
rs2
rd
writedata
data1
data2
ALU controller
byte flag
ALUop
ldUImm
32
immGen

		RegWrite	ALUSrc	Branch	Jump	MemRe	MemWr	MemtoReg	LdUimm	ALUop
R-type	ADD	1	0	0	0	0	0	0	0	10
	XOR	1	0	0	0	0	0	0	0	10
I-type	ORI	1	1	0	0	0	0	0	0	11
	SRAI	1	1	0	0	0	0	0	0	11
	LB	1	1	0	0	1	0	1	0	00
	LW	1	1	0	0	1	0	1	0	00
S-type	SB	0	1	0	0	0	1	0	0	00
	SW	0	1	0	0	0	1	0	0	00
B-Type	BEQ	0	0	1	0	0	0	0	0	01
J-type	JAL	1	0	0	1	0	0	0	0	XX
U-Type	LUI	1	0	0	0	0	0	0	1	XX

SB, LB have byte_flag that ALU_controller generates, because it receives func3. For LUI, shift by 12 is supported inside Immgen so ALU doesn't do anything and ALUop doesn't matter.