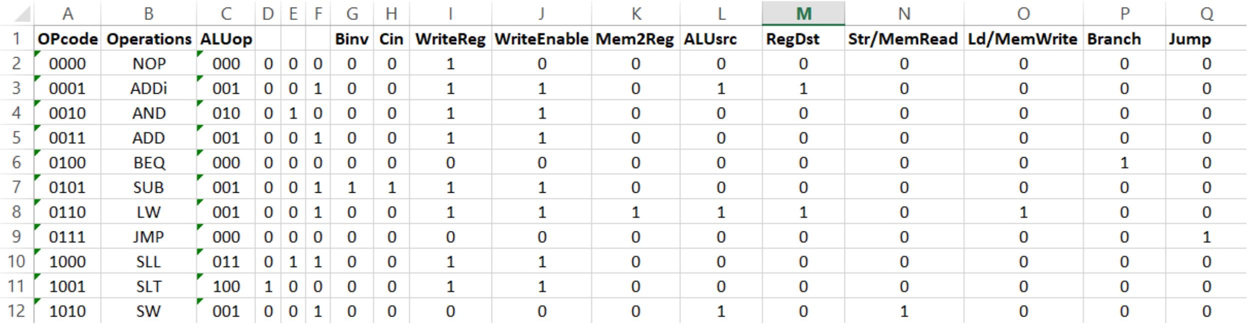
Objectives

Our objective is to design a 20 bit ISA which can solve problems such as addition, shifting, comparism, branch, etc.

Types of Operations

1. R-type: ADD, AND, SUB, SLL, SLT
2. I-type: ADDi, LW, SW
3. J-type: BEQ, JMP

Control Table



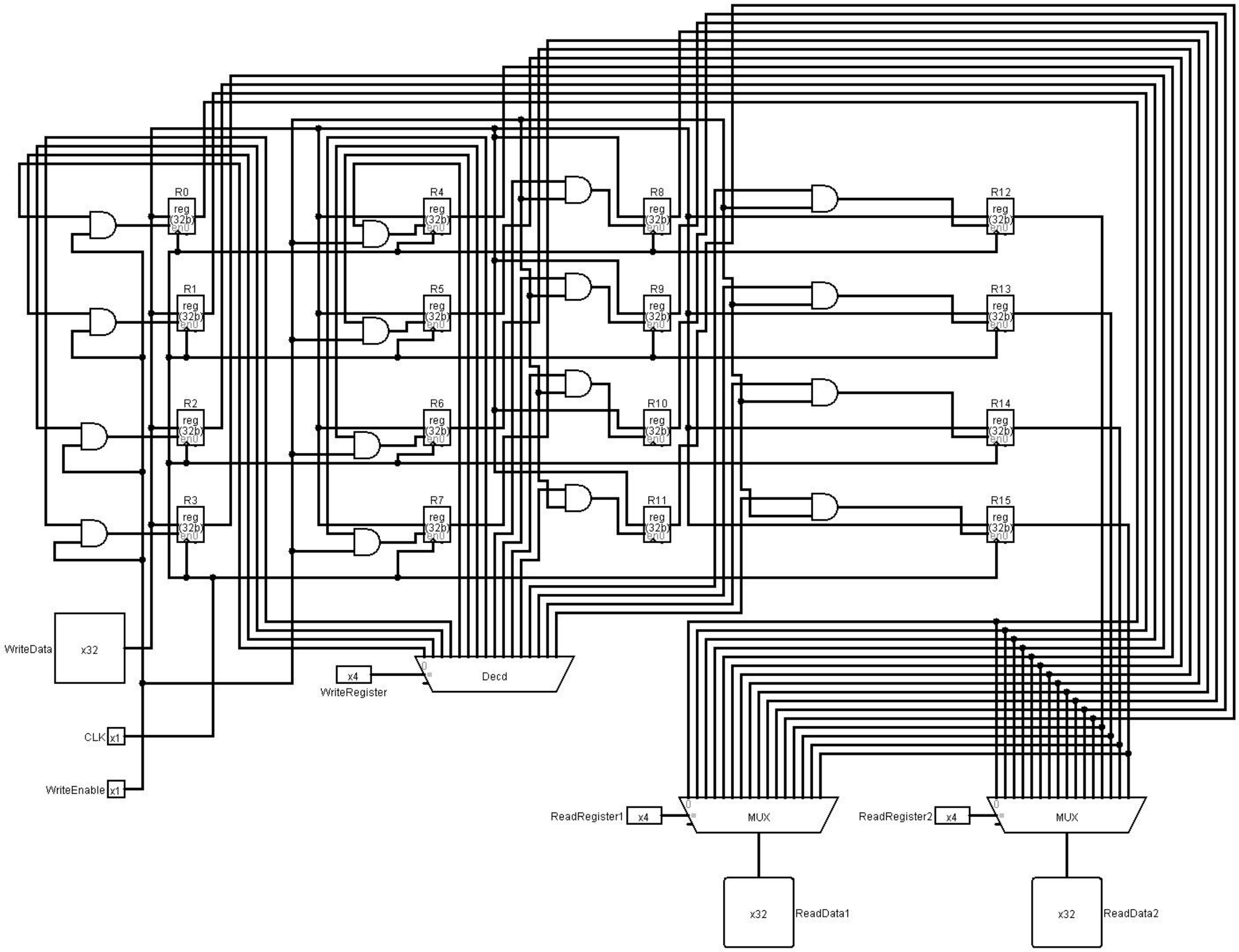
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	OPcode	Operations	ALUop				Binv	Cin	WriteReg	WriteEnable	Mem2Reg	ALUsrc	RegDst	Str/MemRead	Ld/MemWrite	Branch	Jump
2	0000	NOP	000	0	0	0	0	0	1	0	0	0	0	0	0	0	0
3	0001	ADDi	001	0	0	1	0	0	1	1	0	1	1	0	0	0	0
4	0010	AND	010	0	1	0	0	0	1	1	0	0	0	0	0	0	0
5	0011	ADD	001	0	0	1	0	0	1	1	0	0	0	0	0	0	0
6	0100	BEQ	000	0	0	0	0	0	0	0	0	0	0	0	0	1	0
7	0101	SUB	001	0	0	1	1	1	1	1	0	0	0	0	0	0	0
8	0110	LW	001	0	0	1	0	0	1	1	1	1	1	0	1	0	0
9	0111	JMP	000	0	0	0	0	0	0	0	0	0	0	0	0	0	1
10	1000	SLL	011	0	1	1	0	0	1	1	0	0	0	0	0	0	0
11	1001	SLT	100	1	0	0	0	0	1	1	0	0	0	0	0	0	0
12	1010	SW	001	0	0	1	0	0	0	0	0	1	0	1	0	0	0

Instruction Format

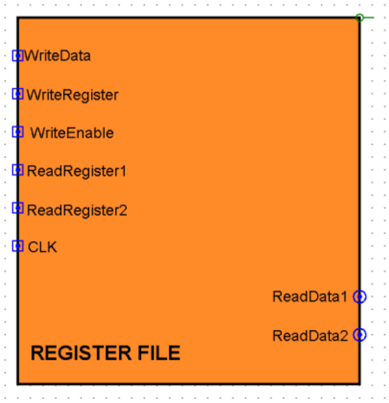
Immediate/R3 (5 bits) R2 (5 bits) R1 (5 bits) Opcode (5 bits)

Since there are 4-bit select keys to choose specific registers and decoder output in control unit, these 5 bits are bit-extended to 4 bits.

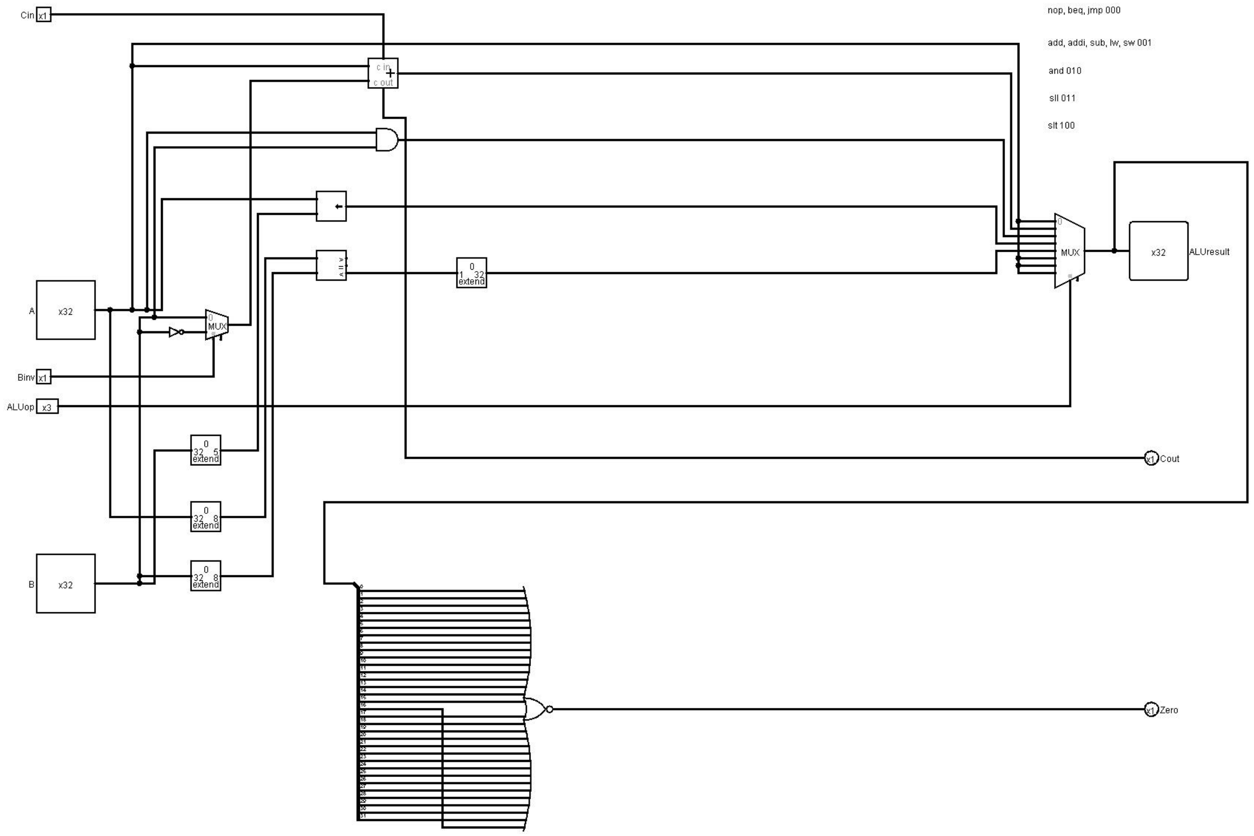
Circuit of Register File



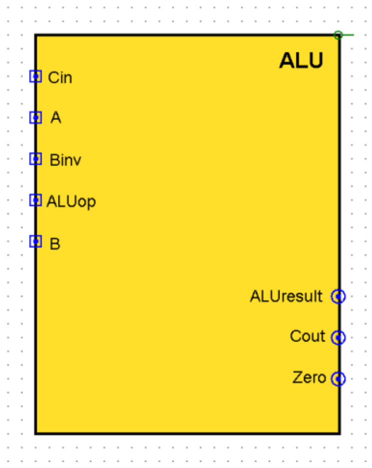
Subcircuit of Register File



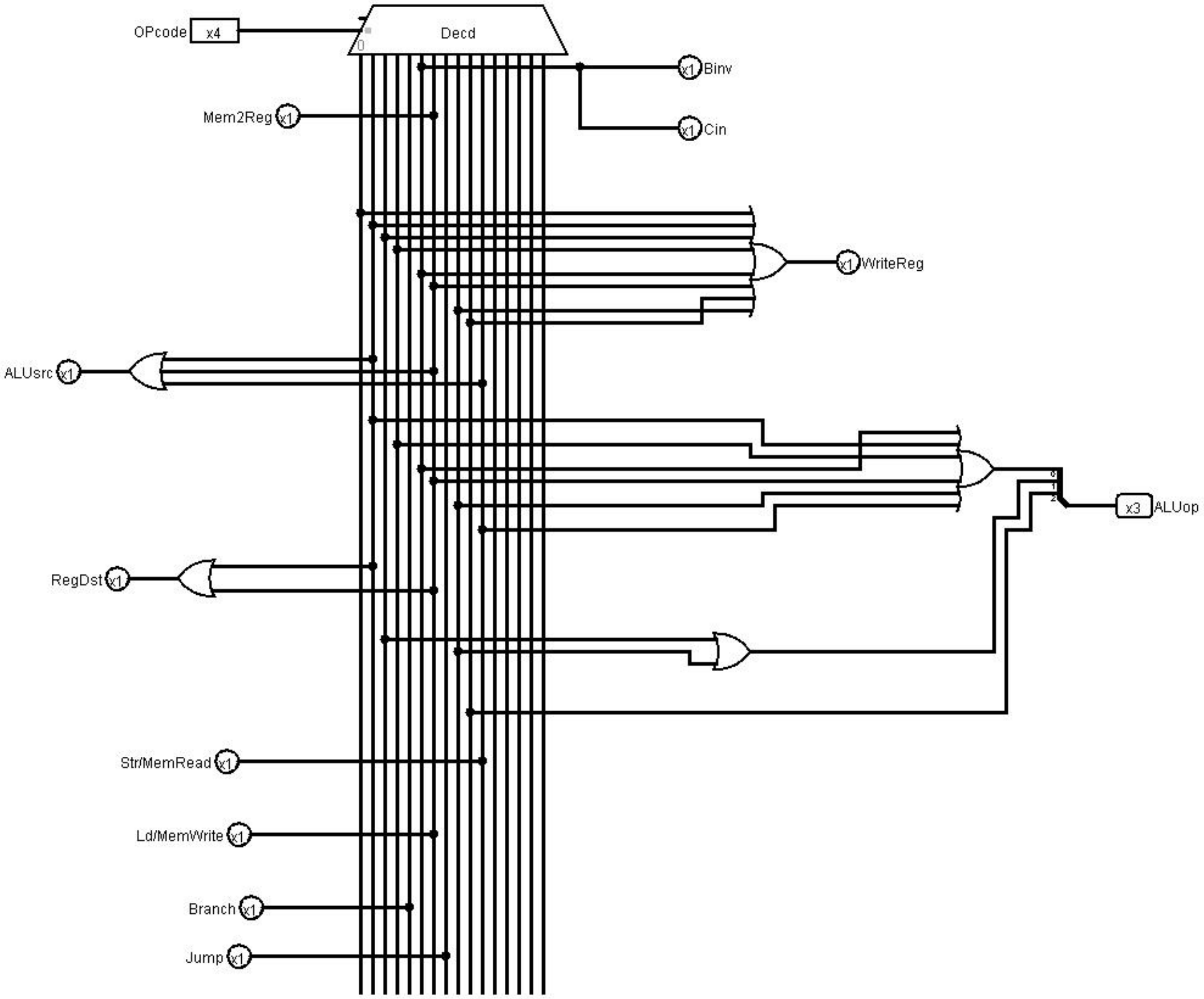
Circuit of ALU



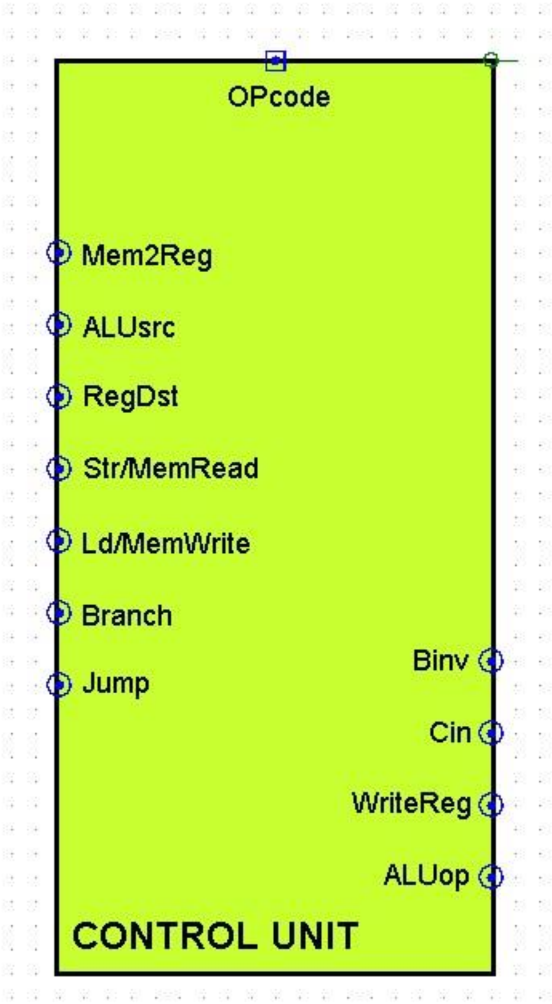
Subcircuit of ALU



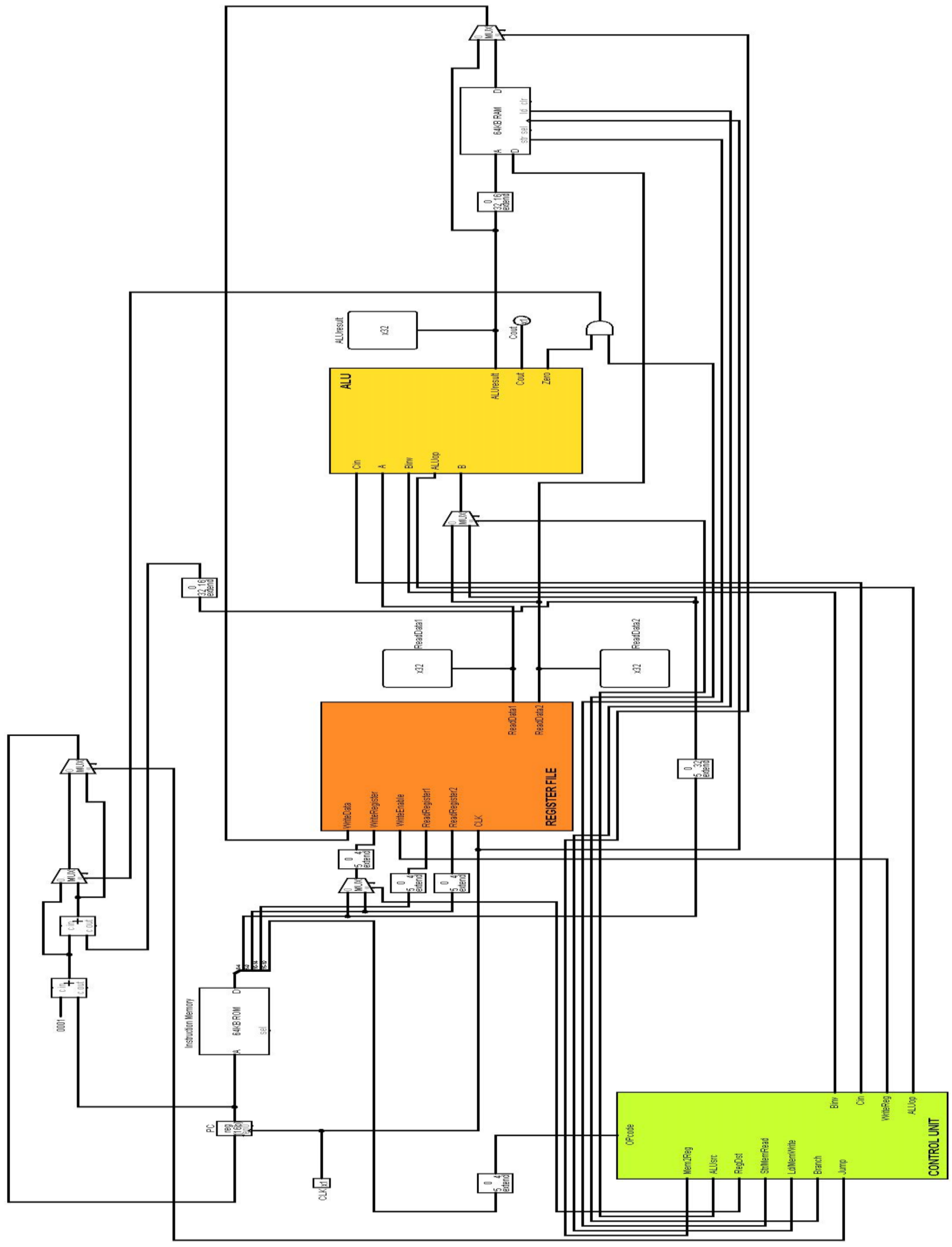
Circuit of Control Unit



Sub-circuit of Control Unit



Circuit of Datapath



Limitations

1. The second input of shift is 5 bits, whereas Read Data 2 is 32 bits. Although bit-extender is used, a better alternative is to build a separate shifting sub-circuit, so that more bits can be shifted.
2. Both inputs of the comparator device is 8 bits, whereas Read Data 1 and Read Data 2 are 32 bits. If the 9th bit or later causes one of them to not be same/less than/greater than, then it will not be detected in comparator. To overcome this limitation, a separate 32 bit comparator sub-circuit can be built.