RISC-V Single Cycle Microarchitecture for S-Type Instructions (ISA)

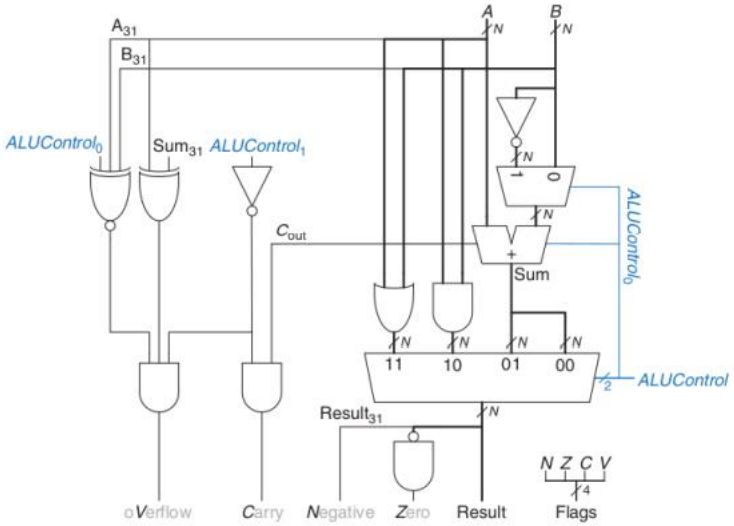
1. S-Type Instructions The name S-type is short for store-type. S-type instructions use two register operands and one immediate operand. Figure below shows the S-type machine instruction format. The 32-bit instruction has five fields: op, rs1, rs2, funct3 and imm. The first three fields op, rs1, and rs2 are like those of R-type instructions. The imm fields hold the 12-bit immediate. The 12-bit immediate is split into two sets as shown below. The operation is determined by the opcode and funct3, highlighted in blue. The operands are specified in the three fields rs1, rs2 and imm.

31:25	24:20	19:15	14:12	11:7	6:0	
$imm_{11:5}$	rs2	rs1	funct3	$imm_{4:0}$	op	**S-Type**

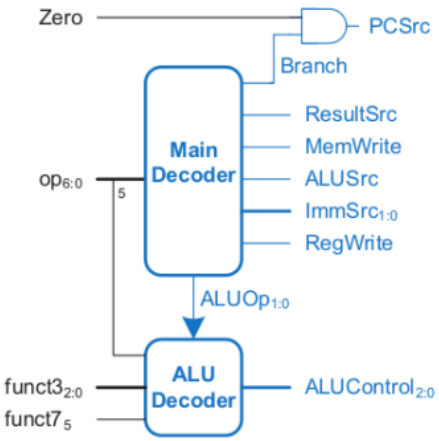
2. Instruction format for some of the I-type instructions are shown below:

imm[11:5]	rs2	rs1	000	imm[4:0]	0100011	SB
imm[11:5]	rs2	rs1	001	imm[4:0]	0100011	SH
imm[11:5]	rs2	rs1	010	imm[4:0]	0100011	SW

3. ALU



4. CONTROL UNIT



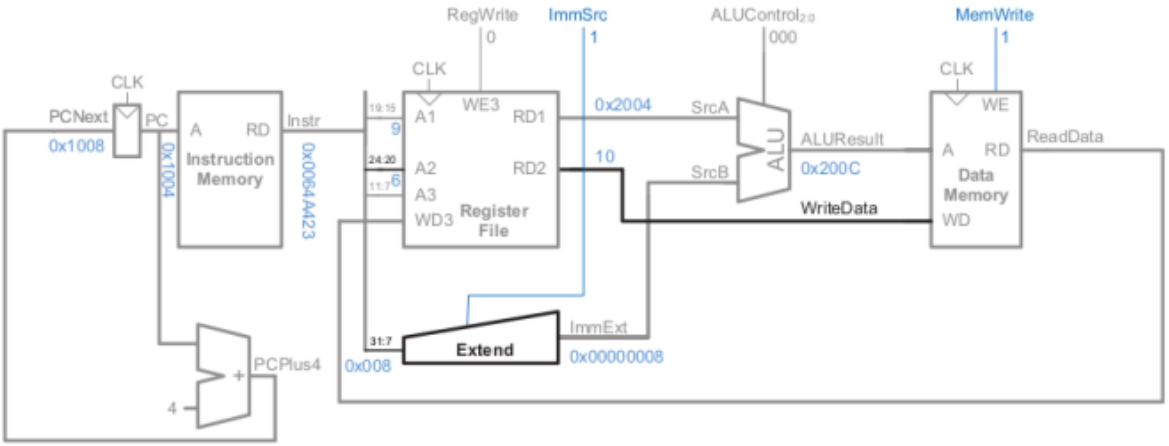
- MAIN Decoder The table below is a truth table for the main decoder that summarizes the control signals as a function of the opcode. All R-type instructions use the same main decoder values; they differ only in the ALU decoder output. Recall that, for instructions that do not write to the register file (e.g., S-type and B-type), the ResultSrc control signals is don't care (X); the address and data to the register write port do not matter because RegWrite is not asserted. The logic for the decoder can be designed using your favorite techniques for combinational logic design.

Instruction	Op	RegWrite	ImmSrc	ALUSrc	MemWrite	ResultSrc	Branch	ALUOp
lw	0000011	1	00	1	0	1	0	00
sw	0100011	0	01	1	1	x	0	00
R-type	0110011	1	xx	0	0	0	0	10
beq	1100011	0	10	0	0	x	1	01

- ALU Decoder

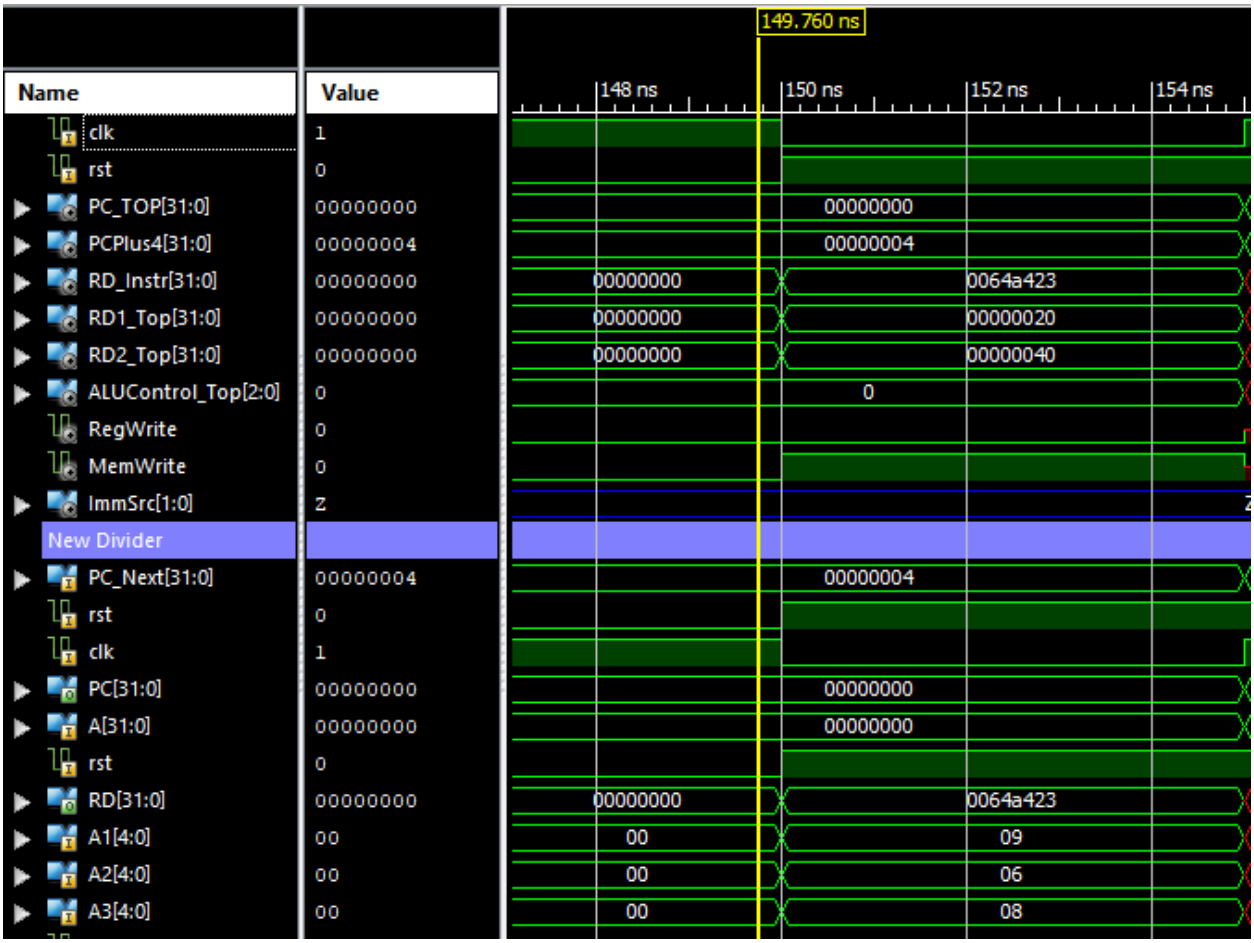
ALUOp	funct3	{op5, funct75}	ALUControl	Instruction
00	x	x	000 (add)	lw, sw
01	x	x	001 (subtract)	beq
10	000	00, 01, 10	000 (add)	add
	000	11	001 (subtract)	sub
	010	x	101 (set less than)	slt
	110	x	011 (or)	or
	111	x	010 (and)	and

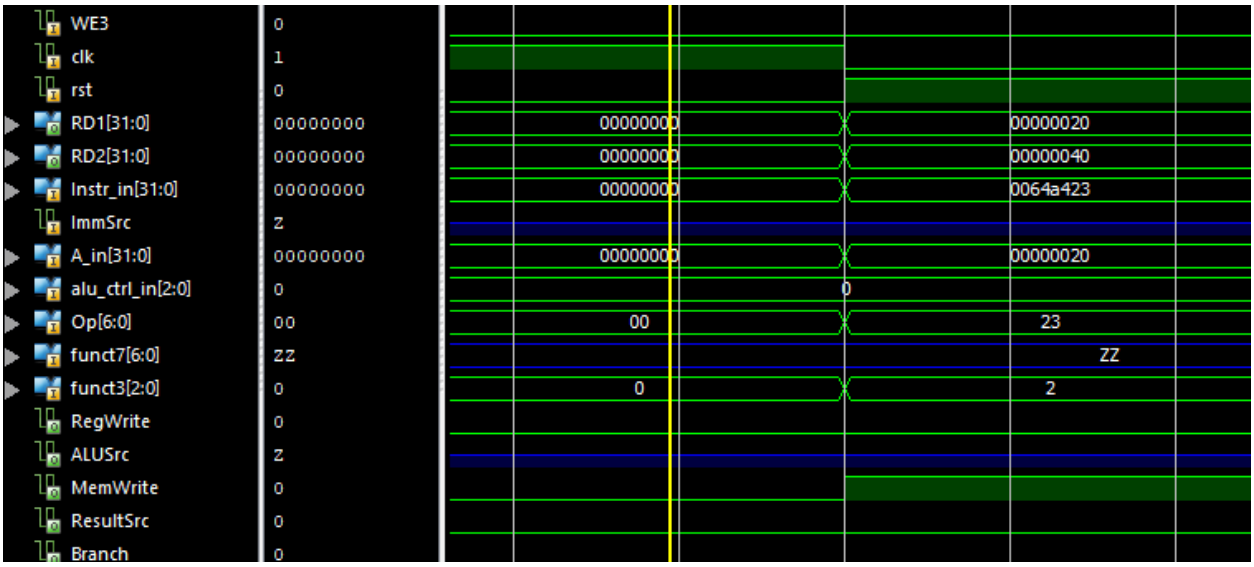
5. Store instructions are S-type instructions. The store word instruction, sw, copies data from a register to memory. The register is not changed. The memory address is specified using a base/register pair. The name S-type is short for Store-type. S-type instructions use two register operands and one immediate operand. The figure below shows the S-type machine instruction format. The 32-bit instruction has five fields: op, rs1, rs2, funct3, and imm. The first four fields, op, rs1, rs2, and funct3 are like those of R-type instructions. The imm field holds the 12-bit immediate. The operation is determined by the opcode and funct3, highlighted in blue. The operands are specified in the three fields rs1, rs2, and imm. rs and imm are always used as source operands. rs2 is used as another source for store instructions.



Address	Instruction	Type	Fields						Machine Language
			$imm_{11:5}$	rs2	rs1	f3	$imm_{4:0}$	op	
0x1004	sw x6, 8(x9)	S	0000000	00110	01001	010	01000	0100011	0064A423

6. SIMULATION RESULTS


149.760 ns
Name
Value
148 ns
150 ns
152 ns
154 ns
clk
1
rst
0
PC_TOP[31:0]
00000000
PCPlus4[31:0]
00000004
RD_Instr[31:0]
00000000
RD1_Top[31:0]
00000000
RD2_Top[31:0]
00000000
ALUControl_Top[2:0]
0
RegWrite
0
MemWrite
0
ImmSrc[1:0]
Z
New Divider
PC_Next[31:0]
00000004
rst
0
clk
1
PC[31:0]
00000000
A[31:0]
00000000
rst
0
RD[31:0]
00000000
A1[4:0]
00
A2[4:0]
00
A3[4:0]
00
0064a423
00000020
00000040
09
06
08


WE3
0
clk
1
rst
0
RD1[31:0]
00000000
RD2[31:0]
00000000
Instr_in[31:0]
00000000
ImmSrc
Z
A_in[31:0]
00000000
alu_ctrl_in[2:0]
0
Op[6:0]
00
funct7[6:0]
ZZ
funct3[2:0]
0
RegWrite
0
ALUSrc
Z
MemWrite
0
ResultSrc
0
Branch
0
00000020
00000040
0064a423
23
2