Design and Synthesis of a 32-bit RISC Processor

CS39001: Computer Organization and Architecture Laboratory

Group 06: Harshit Jain (22CS10030), Sachish Singla (22CS30046)

22nd October 2024

Contents

1 Instruction Format and Encoding

1.1 R-Type Instructions

opcode	rs	rt	rd	Don't Care	func
6 bits	5 bits	5 bits	5 bits	6 bits	5 bits

Table 1: R-type Instruction Format

Instruction	Usage	Opcode	Function
ADD	ADD rd,rs,rt	000000	00001
SUB	SUB rd,rs,rt	000000	00010
AND	AND rd,rs,rt	000000	00011
OR	OR rd,rs,rt	000000	00100
XOR	XOR rd,rs,rt	000000	00101
NOR	NOR rd,rs,rt	000000	00110
SL	SL rd,rs,rt	000000	00111
SRL	SRL rd,rs,rt	000000	01000
SRA	SRA rd,rs,rt	000000	01001
SLT	SLT rd,rs,rt	000000	01010
SGT	SGT rd,rs,rt	000000	01011
NOT	NOT rd,rs	000000	01100
INC	INC rd,rs	000000	01101
DEC	DEC rd,rs	000000	01110
HAM	HAM rd,rs	000000	01111
MOVE	MOVE rd,rs	010100	10000
CMOV	CMOV rd,rs,rt	010101	10001

Table 2: Opcodes and function codes for R-type instructions

1.2 I-Type Instructions

opcode	rs	rt	immediate
6 bits	5 bits	5 bits	16 bits

Table 3: I-type Instruction Format

Instruction	Usage	Opcode
ADDI	ADDI rt,rs,imm	000001
SUBI	SUBI rt,rs,imm	000010
ANDI	ANDI rt,rs,imm	000011
ORI	ORI rt,rs,imm	000100
XORI	XORI rt,rs,imm	000101
NORI	NORI rt,rs,imm	000110
SLI	SLI rt,rs,imm	000111
SRLI	SRLI rt,rs,imm	001000
SRAI	SRAI rt,rs,imm	001001
SLTI	SLTI rt,rs,imm	001010
SGTI	SGTI rt,rs,imm	001011
NOTI	NOTI rt,imm	001100
INCI	INCI rt,imm	001101
DECI	DECI rt,imm	001110
HAMI	HAMI rt,imm	001111
LUI	LUI rt,imm	010000
LD	LD rt,imm(rs)	010001
ST	ST rt,imm(rs)	010010
BMI	BMI rs,imm	100001
BPL	BPL rs,imm	100010
BZ	BZ rs,imm	100011

Table 4: Opcodes for I-type instructions

1.3 J-Type Instructions

opcode	immediate
6 bits	26 bits

Table 5: J-type Instruction Format

Instruction	Usage	Opcode
BR	BR imm	100000

Table 6: Opcodes for J-type instructions with register arguments

1.4 Program Control Instructions

opcode	Don't Care
6 bits	26 bits

Table 7: Program Control Instruction Format

Instruction	Opcode
HALT	100100
NOP	100101
CALL	100110

Table 8: Opcodes for program control instructions

2 Register Usage Convention

The register file contains 18 registers, each of size 32 bits.

Register	Function	Register Number
$R0	Zero Register	00000
$R1-$R15	General Purpose Registers	00001-01111
$pc	Program Counter	10000
$ret	Return Address	10001

Table 9: Register Usage Convention

3 Datapath

The datapath consists of the following components:

- 32-bit general-purpose registers.
- ALU for arithmetic and logic operations.
- Program Counter (PC) and incrementer for the same
- Control Unit: A hardwired control unit that generates control signals.
- Memory: Byte-addressable memory, supporting 32-bit aligned access.

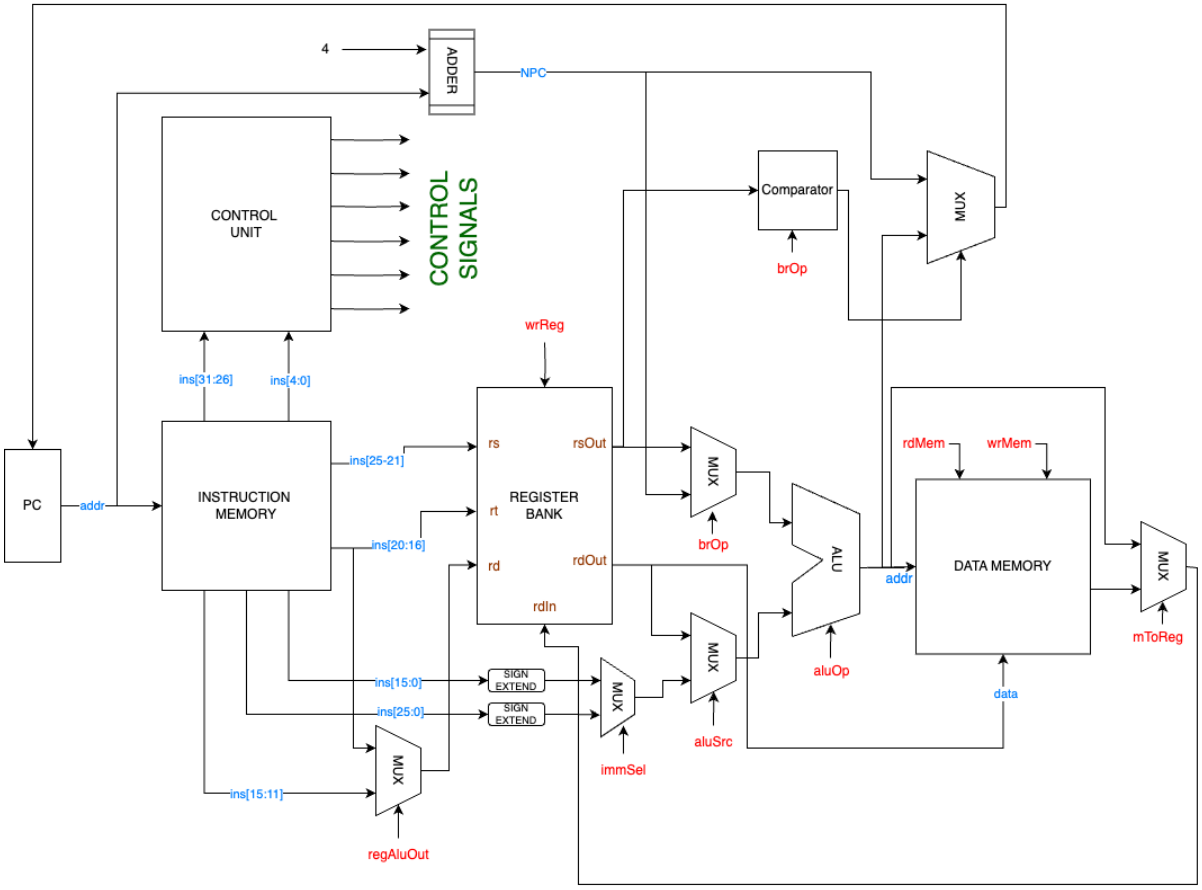

Figure 1: Processor Datapath

4 Control Unit

4.1 aluOp

Opcode	aluOp	Meaning
000000	$ins_{3:0} - 1$	Infer from func: ADD,...,HAM
000001-001111	$ins_{29:26} - 1$	Infer from opcode: ADD,...,HAM
010000	1111	LUI
010001-010010,100000-100100	0000	ADD
other	xxxx	Don't Care

Table 10: Specification of ALU operations

4.2 brOp

Opcode	brOp	Meaning
100000	001	BR
100001	010	BPL
100010	011	BMI
100011	100	BZ
other	000	Not a branching instruction

Table 11: Specification of Branch operations

4.3 aluSrc

Opcode	aluSrc	Meaning
000000	1	rt
other	0	imm

Table 12: Selection of 2nd source of ALU input

4.4 aluOut

Opcode	aluOut	Meaning
000000,010100-010101	1	rd
000001-010001	0	rt
other	x	Don't Care

Table 13: Selection of Destination Register

4.5 rdMem

Opcode	rdMem	Meaning
010001	1	READ from #imm(R[rs])
other	0	Don't Read

Table 14: Read from Memory

4.6 wrMem

Opcode	wrMem	Meaning
010010	1	WRITE to #imm(R[rs])
other	0	Don't Write

Table 15: Write to Memory

4.7 wrReg

Opcode	wrReg	Meaning
000000-010001,010100-010101	1	Write to dest reg
other	0	Don't Write

Table 16: Write to Register

4.8 mToReg

Opcode	mToReg	Meaning
000000-010000,010100-010101	0	Value from ALUOut
010001	1	Value from LMD
other	x	Don't Care

Table 17: Source of value to be written to dest reg

4.9 immSel

Opcode	immSel	Meaning
000001-010010,100001-100011	0	$(ins_{15})^{16}\#\#ins_{15:0}$
100000	1	$ins_{25:0}\#\#00$
other	x	Don't Care

Table 18: Selection of sign extended IMM (I-type or J-type)