EE-739: Processor Design

Aashish Tamrakar
193079034
30 July 2020

Problem Statement

Design a 6 stage pipelined processor, IITB-RISC, whose instruction set architecture is provided. IITB-RISC is a 16-bit very simple computer developed for the teaching that is based on the Little Computer Architecture.

The IITB-RISC is have following features

- 8-register
- 16-bit computer system
- 6 stage pipelines
  - Instruction fetch
  - instruction decode
  - Register read
  - Execute,
  - Memory access
  - Write back
- Hazard mitigation techniques.
  - Forwarding and branch
  - Prediction technique.

IITB-RISC Instruction Set Architecture

- 8 general-purpose registers (R0 to R7)
  - Register R7 is always stores Program Counter. All addresses are short word addresses .
- This architecture uses condition code register 2 Flags
  - Carry flag (C)
  - Zero flag (Z)
- The architecture allows
  - Predicated instruction execution
  - Multiple load and store execution.
- There are three machine-code instruction formats (R, I, and J type) and a total of 14 instructions.

Instruction Format

- R Type Instruction format

Opcode (4 bit)	Register A (RA) (3 bit)	Register B (RB) (3-bit)	Register B (RB) (3-bit)	Unused (1 bit)	Condition (CZ) (2 bit)

- I Type Instruction format

Opcode (4 bit)	Register A (RA) (3 bit)	Register C (RC) (3-bit)	Immediate (6 bits signed)

- J Type Instruction format

Opcode (4 bit)	Register A (RA) (3 bit)	Immediate (9 bits signed)

Instruction Encoding

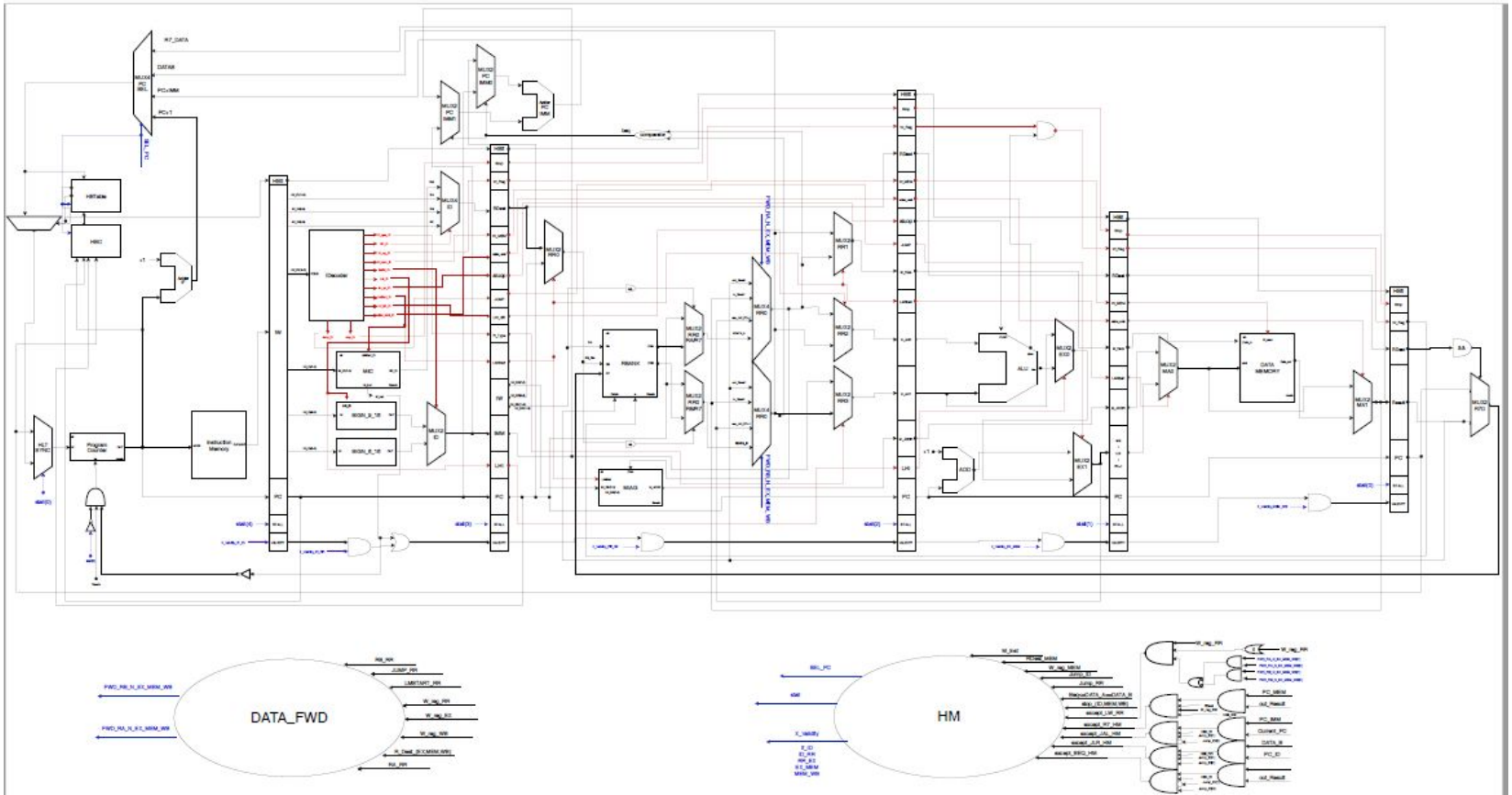
ADD:	00_00	RA	RB	RC	0	00
ADC:	00_00	RA	RB	RC	0	10
ADZ:	00_00	RA	RB	RC	0	01
ADI:	00_01	RA	RB	6 bit Immediate		
NDU:	00_10	RA	RB	RC	0	00
NDC:	00_10	RA	RB	RC	0	10
NDZ:	00_10	RA	RB	RC	0	01
LHI:	00_11	RA	9 bit Immediate			
LW:	01_00	RA	RB	6 bit Immediate		
SW:	01_01	RA	RB	6 bit Immediate		
LM:	01_10	RA	0 + 8 bits corresponding to Reg R7 to R0			
SM:	01_11	RA	0 + 8 bits corresponding to Reg R7 to R0			
BEQ:	11_00	RA	RB	6 bit Immediate		
JAL:	10_00	RA	9 bit Immediate offset			
JLR:	10_01	RA	RB	000_000		

RA: Register A

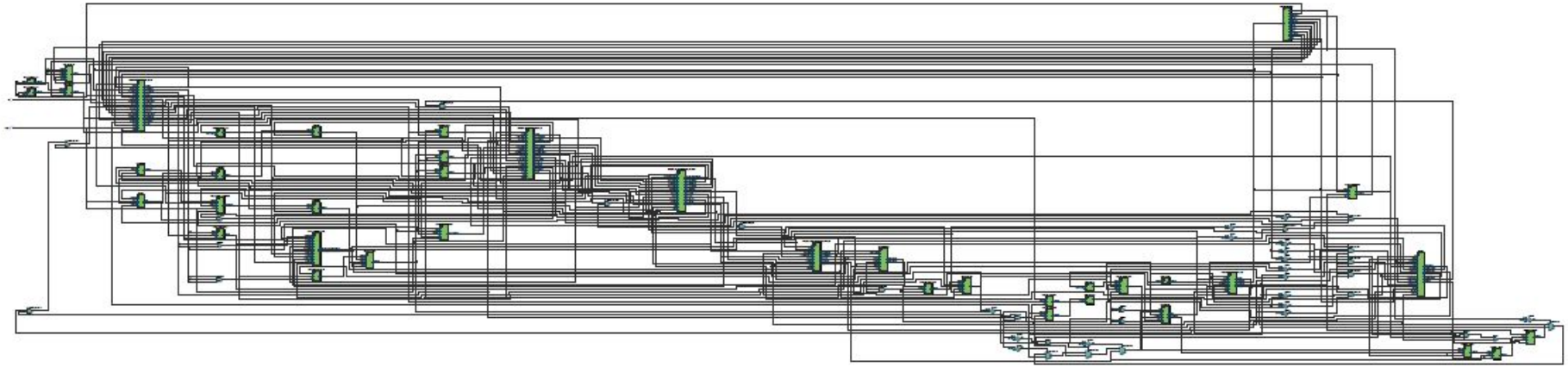
RB: Register B

RC: Register C

Datapath of Design with Prediction (Planned Manually)



Datapath of Design with Prediction (Quartus)



Instruction Decoder(Controller)

Instruction Decode Logic Table													
Levels_required		1	4	4	3	2	0	2	3,3	1	3	2	4
opcode	CODE (16bit)	R_type	RD	W_reg	W_mem	LHI	sel69	alu_op	Lmstart	lwsr	mem_ans	Jump	Stop
add r3,r1,r2	0000 ra rb rc 0 00	0	10	1	0	0	0	001	00	0	1	00	0
adc r3,r1,r2	0000 ra rb rc 0 10	0	10	1	0	0	0	010	00	0	1	00	0
adz r3,r1,r2	0000 ra rb rc 0 01	0	10	1	0	0	0	011	00	0	1	00	0
adi r2,r1,D6	0001 ra rb UUU_UUU	1	01	1	0	0	0	001	00	0	1	00	0
ndu r3,r1,r2	0010 ra rb rc 0 00	0	10	1	0	0	0	100	00	0	1	00	0
ndc r3,r1,r2	0010 ra rb rc 0 10	0	10	1	0	0	0	101	00	0	1	00	0
ndz r3,r1,r2	0010 ra rb rc 0 01	0	10	1	0	0	0	110	00	0	1	00	0
Lhi r1,D9	0011 ra UUU_UUU_UUU	1	00	1	0	1	1	111	00	0	1	00	0
lw r1,r2,imm	0100 ra rb UUU_UUU	1	00	1	0	0	0	000	00	1	0	00	0
sw r1,r2,imm	0101 ra rb UUU_UUU	1	00	0	1	0	0	000	00	1	0	00	0
LM ra,D6	0110 ra 0 UUU_UUU_UU	1	11	1	0	0	0	111	10	0	0	00	0
SM ra,D6	0111 ra 0 UUU_UUU_UU	1	11	0	1	0	0	111	11	0	0	00	0
beq ra,rb,d6	1100 ra rb UUU_UUU	1	00	0	0	0	0	111	00	0	0	01	0
jal ra,imm	1000 ra UUU_UUU_UUU	1	00	1	0	0	1	111	00	0	1	10	0
jlr ,ra,rb	1001 ra rb 000_000	0	00	1	0	0	0	111	00	0	0	11	0
NOP	1111 1111 1111 1110	0	00	0	0	0	0	000	00	0	0	0	0
HALT	1111 1111 1111 1111	0	00	0	0	0	0	000	00	0	0	0	1

Note:- 1.The "U" will be filled as per our interest as Given in Project Statement

Level=Pipeline Stage

RTL Simulation (on ModelSim)