EE-309 Microprocessors

REPORT TITLE

MULTI-CYCLE PROCESSOR IITB RISC-22

MEMBERS

Shrey Viral Ganatra :20d070074

Jahnvi Singh Rohela :20d070040

Kaushal Sunil Jadhav :200070026

Gnanendar Reddy Palagiri :200070053

Introduction

In this project we are designing an ISA for a simple multi-cycle RISC,which is a 16-bit computer system with 8 registers and the register R7 acts as both PC and R7.

Data-Path

Instructions:

We have a total of 17 instructions which consist of R,J,I type instructions.we were able to implement 17 given instructions with 22-unique states from S0 to S21.

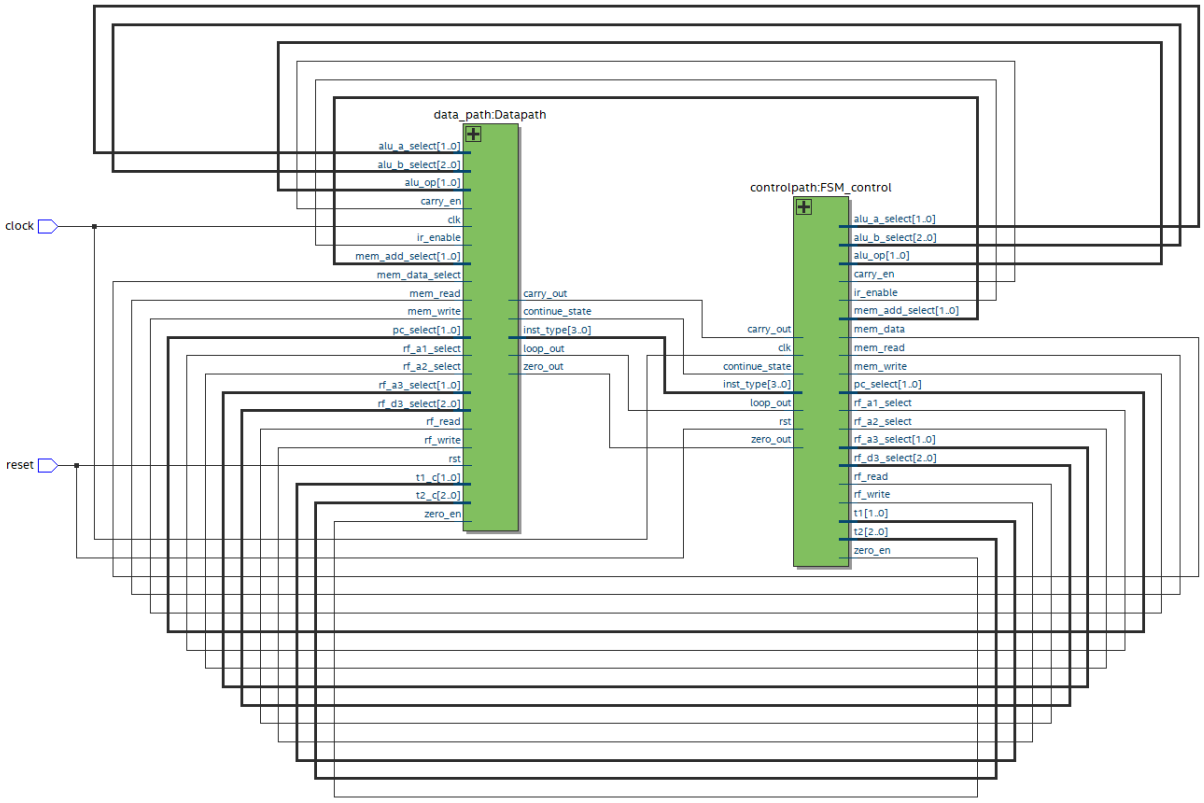
Components:

We were able to implement these 22 unique states using the following components : **Memory,Instruction Decoder,Data Extender,Register File,SE6,SE9,LSI,ALU** ,**PE** and 3 temporary registers **t1,t2&t3.**

We made the connections as we mentioned in the datapath diagram present in the hand-written part.There are many muxes as shown in the hand-written part which are used to select the inputs to various components in the data-path.

RTL-View:

We connected the control flow and data path in a new .vhd file which acts as a top level entity.The RTL-View of top level entity is as follows:



Memory:

It has Read,Write,address in,data_in&data_out ports with control signals R & W we control when to write and when to read into memory.

We used 128 memory blocks each of 16 bits to have the required memory.

Instruction Decoder:

It takes the 16 bit instruction and decode(split) it into required instruction parts like Ir_(3-5),Ir_(6-8),Ir_(9-11),Ir_(0-5),Ir_(0-8),Ir_(0-7),etc.

Data Extender:

It takes the 8 bit input and gives o/p of 16 bits with most significant 8 bits are the i/p 8 bits and least significant 8 bits are '0'.

Register File:

It have 3 addresses a1,a2,a3 a1&a2 takes address for rfd1 &rfd2 which gives data from respective registers specified by a1 and a2 and a3 corresponds to rfd3 which takes input data and place it into the register whose address is given by a3.

SE6&SE9:

These components respectively take 6 and 9 bits data and sign extend it to 16 bits example

100101 is extended as **1111111111100101.**

LSI:

It shifts the 16 bits 1bit left and the initial bit LSB is now '0'.

ALU:

ALU has two inputs alu-a,alu-b which take input and compute output based on the control word like addition or operation based on op-code and store o/p into t3.It is controlled by three control bits.

t1,t2&t3:

These are temporary registers to store values till the next state . t1 and t2 are the temporary registers before the two inputs of alu and t3 is the temporary register that is directly connected to the o/p of ALU.

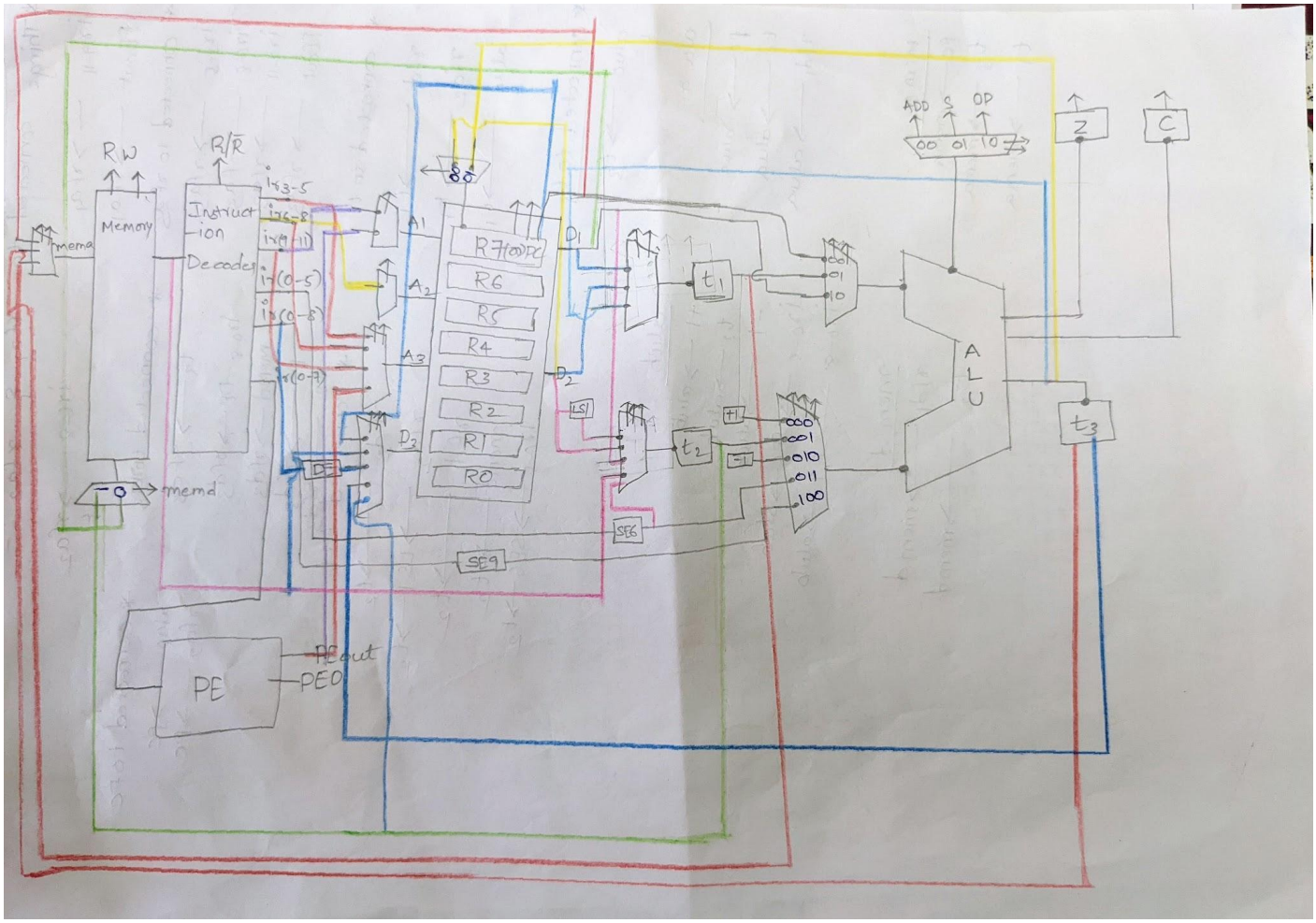
PE:

Priority Encoder takes into input 8 bits (the last eight bits of LM/SMinstruction) and gives the 3-bit index of the first non-zero bit. This is helpful in choosing which Register to Store/Load from.

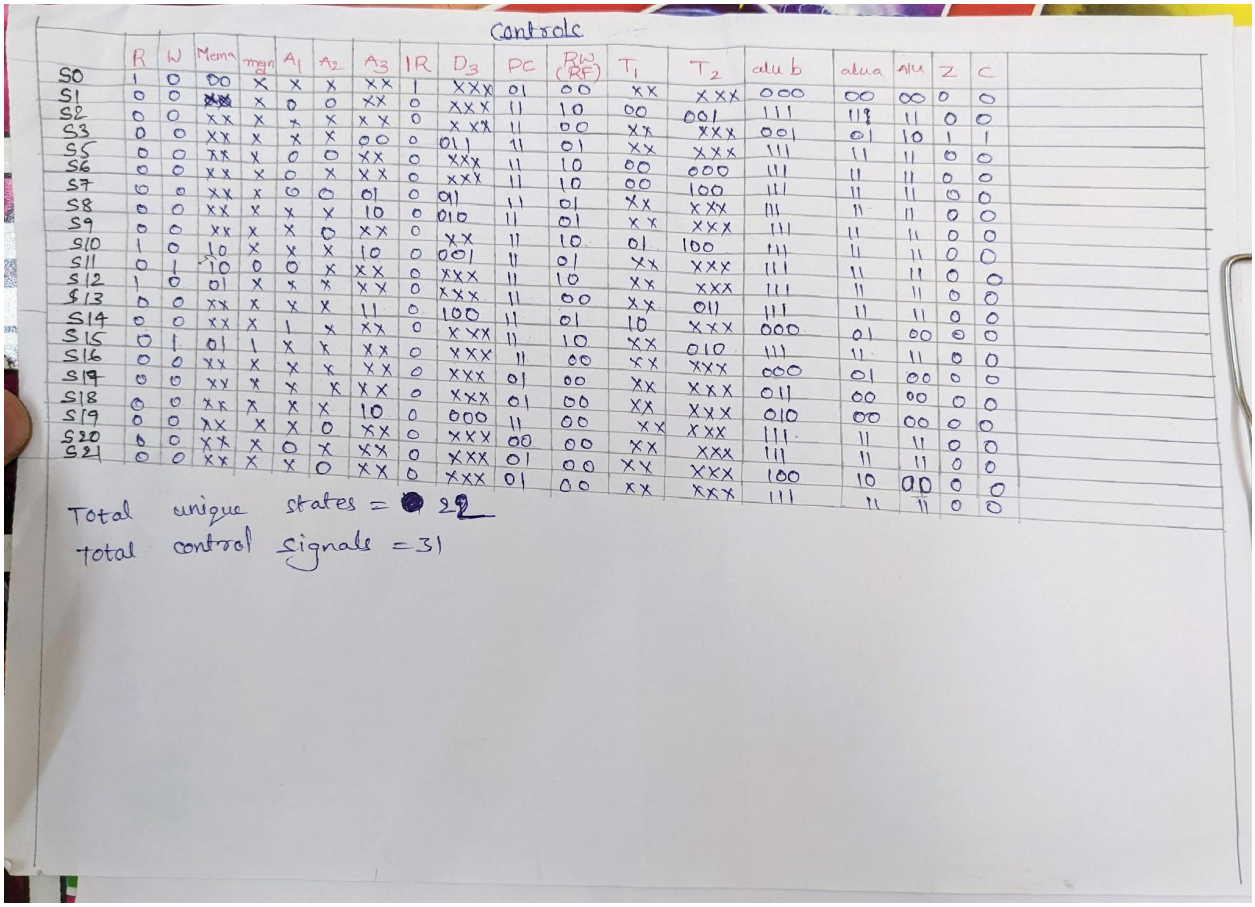
PE MODIFIER :

It takes into input the output of Priority Encoder. It also takes the same 8 bits as input. It returns eight bit with the bit corresponding to Priority Encoder Output set to zero.Thus using the combination of Priority Encoder and Priority Encoder Modifier, we can iteratively tackle the LM and SM instructions.

The hand-written data-path :



Control-Word:



Controls

	R	W	Mem	mgn	A_1	A_2	A_3	IR	D_3	PC	RW (RF)	T_1	T_2	alu b	alua	alu	Z	C
S0	1	0	00	X	X	X	XX	1	XXX	01	00	XX	XXX	000	00	00	0	0
S1	0	0	XX	X	0	0	XX	0	XXX	11	10	00	001	111	11	11	0	0
S2	0	0	XX	X	X	X	XX	0	XXX	11	00	XX	XXX	001	01	10	1	1
S3	0	0	XX	X	X	X	00	0	011	11	01	XX	XXX	111	11	11	0	0
S5	0	0	XX	X	0	0	XX	0	XXX	11	10	00	000	111	11	11	0	0
S6	0	0	XX	X	0	X	XX	0	XXX	11	10	00	100	111	11	11	0	0
S7	0	0	XX	X	0	0	01	0	01	11	01	XX	XXX	111	11	11	0	0
S8	0	0	XX	X	X	X	10	0	010	11	01	XX	XXX	111	11	11	0	0
S9	0	0	XX	X	X	0	XX	0	XX	11	10	01	100	111	11	11	0	0
S10	1	0	10	X	X	X	10	0	001	11	01	XX	XXX	111	11	11	0	0
S11	0	1	10	0	0	X	XX	0	XXX	11	10	XX	XXX	111	11	11	0	0
S12	1	0	01	X	X	X	XX	0	XXX	11	00	XX	011	111	11	11	0	0
S14	0	0	XX	X	1	X	XX	0	100	11	01	10	XXX	000	01	00	0	0
S15	0	1	01	1	X	X	XX	0	XXX	11	10	XX	010	111	11	11	0	0
S16	0	0	XX	X	X	X	XX	0	XXX	11	00	XX	XXX	000	01	00	0	0
S19	0	0	XX	X	X	X	XX	0	XXX	01	00	XX	XXX	011	00	00	0	0
S18	0	0	XX	X	X	X	10	0	000	11	00	XX	XXX	010	00	00	0	0
S19	0	0	XX	X	X	0	XX	0	XXX	00	00	XX	XXX	111	11	11	0	0
S20	0	0	XX	X	0	X	XX	0	XXX	01	00	XX	XXX	111	11	11	0	0
S21	0	0	XX	X	X	0	XX	0	XXX	01	00	XX	XXX	100	10	00	0	0
													XXX	111	11	11	0	0

Total unique states = 22

Total control signals = 31

There are total 22 unique states and 31 control signals including the controls to muxes.