**8-Bit RISC CPU Design**

M.Sai Dinesh and A.V.K.Chaitanya

(16CS01026) (16CS01027)

IIT BHUBANESWAR

# **CONTENTS**

1)Circuit Libraries

2)Instruction format

3)Summary

**PROGRAM COUNTER**

The Program Counter consists of an adder circuit that increments the PC value by one for every input clock cycle. There is a multiplexer which selects between 1 and Branch offset to be added to the PC in case of a jump instruction.Set-PC pin sets the program counter value to zero. PC enable enables the program counter circuit.

**REGISTER-FILE**

The Register file consists of four registers R0,R1,R2 and R3. The output from the register file is taken from two outputs Ra and Rb which are selected using two multiplexers mux1 and mux2. The values in the registers are stored using input c which is connected through a Demultiplexer.

**ALU**

The ALU is used for all arithmetic and logical operations. It consists of two inputs input 1 and input 2 on which computation is performed and a data input which selects the operation to be performed.

**CPU**

The general RISC cpu consists of five stages namely instruction fetch, decode,compute,memory and store.

During the instruction fetch stage the program counter increments by value 1 and the corresponding instruction is stored in instruction register and if there is any load/store/jump operation the program counter still increments by one and stores the corresponding address in address register to perform load/store/jump operation.

During the decode stage the instruction is decoded and corresponding values of registers are stored in Ra and Rb registers. The mux B either selects Rb or immediate value to be sent to ALU. The immediate value occurs in case of Load/store operation.

During the compute stage the ALU performs arithmetic and logical operation if there is any need otherwise it directly sends out the inputs to register Rz and the value in register Ra in previous stage is transferred to register Rm.

During the memory stage if there is any Load/Store operation the address will be present in register Rz. In case of Load operation the address will be sent to memory and the data is sent to register Ry through muxY. In case of Store operation the value in register Rm is stored in address pointed by Rz.

During the store operation the value in Register Ry will be stored in corresponding register if there is any need to store value in registers.

**CONTROL CIRCUIT**

The operation of the processor’s hardware components is governed by control signals. These signals determine which multiplexer input is selected, what operation is performed by the ALU, and so on. The control circuit consists of a counter with 6 stages with each stage representing each stage of CPU instruction cycle. The required control signals are generated by using the counter stage output and the current executing instruction. The control signals generated here are

PC\_ENABLE=T0+T1+T5(JUMP)

STORE2(WRITE DATA IN INSTRUCTION REGISTER)=T1

STORE4(WRITE DATA IN ADDRESS REGISTER)=T2

REG\_SELECT(TO WRITE DATA IN RA AND RB)=T3

MUX\_SELECT(TO SELECT RB OR IMMEDIATE VALUE)=(T3+T4)(LOAD+STORE)

RZ\_ENABLE=T4

MUX\_Y(TO SELECT VALUE TO BE LOADED IN RY)=T5(LOAD)

REG\_Y=T5(ALU+MOV+LOAD)

C\_SELECT=T6(ALU+MOV+LOAD)

STORE1(TO STORE INTO MEMORY)=T5(STORE)

STORE3(SELECT PC OR MEMORY ADDRESS)=T5(LOAD+STORE)

**INSTRUCTION FORMAT**

ADD 0000

SUB 0001

MUL 0010

AND 0011

OR 0100

XOR 0101

MOV 0110

LOAD 0111

STORE 1000

JMP 1001

HLT 1010

Every instruction needs two memory blocks. The first block stores opcode and the registers and second block stores address in case of load/store/jump operation(If there is no address leave it as 00)

Give the instruction in hexadecimal format

Eg - ADD R1 R2 == 0000 01 10== 06 and address 00

In case of LOAD/STORE operation

Eg- LOAD R1 06 == 0111 01 00 06 == 74 and 06

**SUMMARY**

CPU is a processing unit, which executes machine-language instructions and coordinates the activities of other units in a computer. During this project we examined its internal structure and showed how it performs the tasks of fetching, decoding, and executing the instructions.