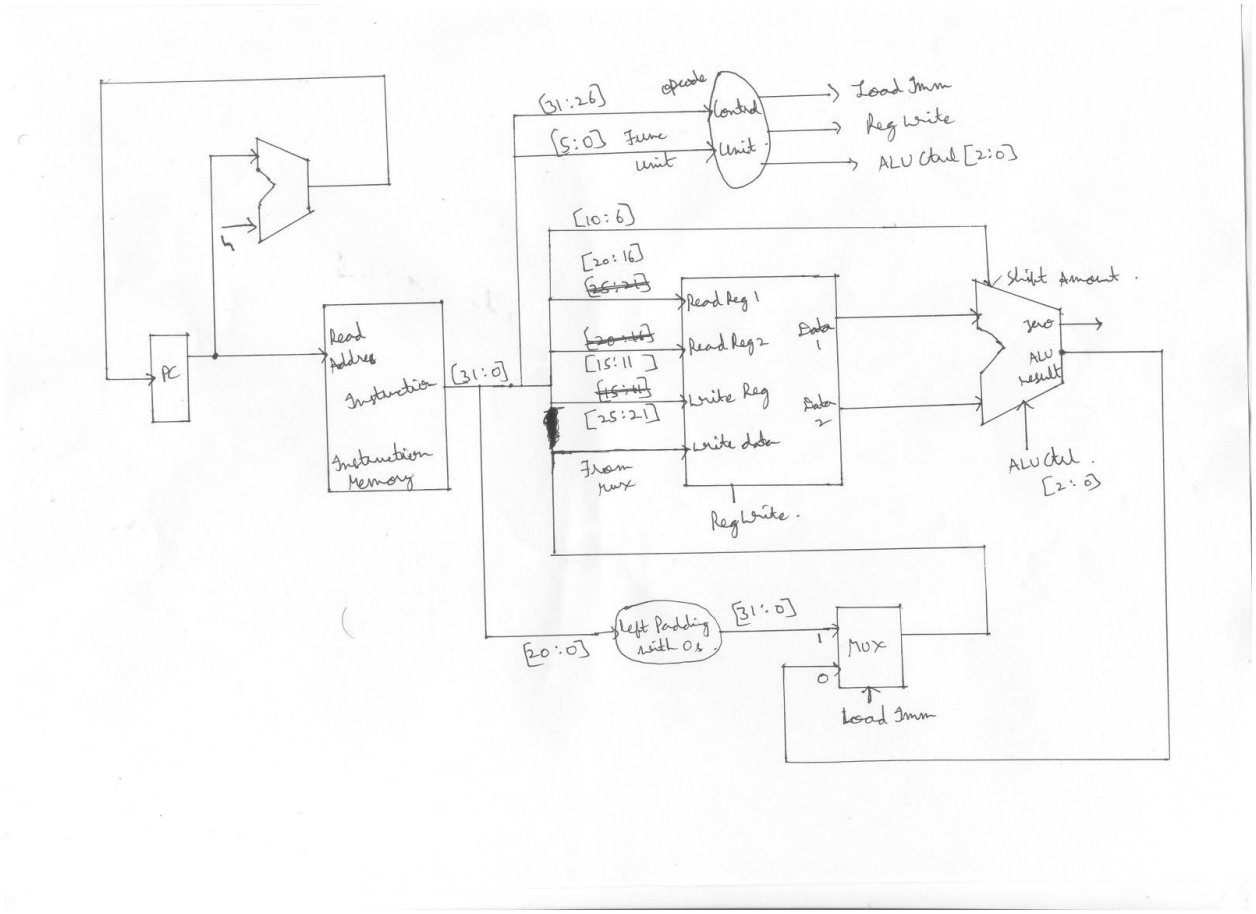
ASSIGNMENT 1

COMPUTER ARCHITECTURE

Name: P. SURYACHANDRA RASAD ID.NO 2014B2A30949G

6.1) block diagram



6.2) LIST THE DIFFERENT BLOCKS THAT WILL BE REQUIRED FOR IMPLEMENTATION OF ABOVE PROCESSOR

ANSWER: Instruction memory
Register file
ALU
Control unit
Left zero padding unit for li instruction

6.3) Most of the datapath blocks that are listed above have already been implemented as part of previous labs. Implement the blocks which have not been implemented in previous labs and copy the images of those verilog codes here.

Answer: only padding unit is new unit and ALU unit needs some modifications

Padding unit:

```
module padding(input [20:0] lmm,output [31:0] padded_result);
  assign padded_result ={11'b0,lmm};
endmodule
```

ALU:

```
module alu(input [31:0] a,input [31:0] b , input [2:0] ctrl, input [4:0] shiftamount ,output reg zero,output reg [31:0] result);

always@(a or b or ctrl)
begin
case(ctrl)

3'b000: result = a + b;
3'b001: result = a - b;
3'b010: result = a & b;
3'b011: result = a | b;
3'b100: result = a << shiftamount;
3'b101: result = a >> shiftamount;

default: result =32'bX;
endcase
if(result==0)
  zero=1;
else
  zero=0;
end
Endmodule
```

6.4) Assume main control unit generates all control signlas.List different control signals that will be required for above processor.Also specify the value of control signals for different instructions.

ANSWER:

Control signal name	RegWrite	ALUctrl	LoadImm(ctrl signal for mux)
Li r1 , 8	1	xxx	1
Add r0,r1,r2	1	000	0
Sub r4,r5,r6	1	001	0
And r8,r9,r10	1	010	0
And r9,r8,r10	1	010	0
Or r9,r8,r10	1	011	0
Sll r11,r6,6	1	100	0
Srl r13,r9,10	1	101	0

6.5) Implement the main control unit and copy the image of verilog code of main control unit here.

Verilog code for control unit:

```
module control(input [5:0] opcode,input [5:0] funct,output reg LoadImm,output reg [2:0] ALUctrl,output reg RegWrite);

 always@(opcode or funct)
 begin
  if(opcode == 6'b000000)
    begin
      case (funct)
        6'h20: {LoadImm,RegWrite,ALUctrl}=5'b01000;

        6'h22: {LoadImm,RegWrite,ALUctrl}=5'b01001;
        6'h24: {LoadImm,RegWrite,ALUctrl}=5'b01010;
        6'h25: {LoadImm,RegWrite,ALUctrl}=5'b01011;
        6'h00: {LoadImm,RegWrite,ALUctrl}=5'b01100;
        6'h02: {LoadImm,RegWrite,ALUctrl}=5'b01101;
      endcase
    end
```

```
   else if (opcode == 6'b111111)
    {LoadImm,RegWrite,ALUctrl}=5'b11xxx;


  end
endmodule
```

6.6) Implement complete processor in verilog(instantiate all datapath blocks and main control unit as modules).copy the verilog code of the processor here.

Answer:

Verilog code :

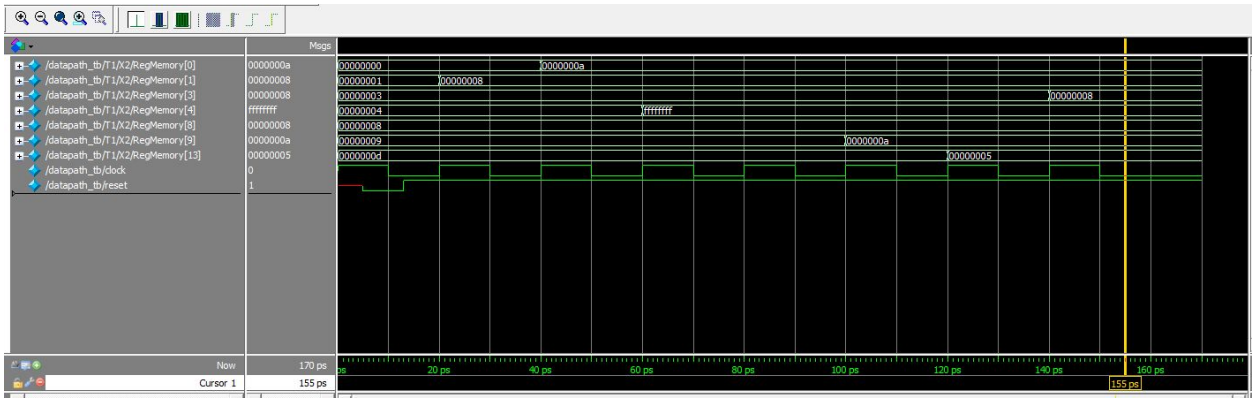
```
module datapath(input clock,input reset);

  wire [31:0] Instruction_Code;
  wire [2:0] ALUctrl;
  wire LoadImm;
  wire [31:0] muxout;
  wire [31:0] Read_Data_1,Read_Data_2;
  wire [31:0] result;
  wire RegWrite;
  wire zero;
  wire [31:0] padded_result;

  Instruction_Fetch X1(clock,reset,Instruction_Code);

  Register_file X2(Instruction_Code[20:16],Instruction_Code[15:11],Instruction_Code[25:21],muxout,Read_Data_1,
Read_Data_2,RegWrite,clock);

  alu X3(Read_Data_1,Read_Data_2,ALUctrl,Instruction_Code[10:6],zero,result);

  control X4(Instruction_Code[31:26],Instruction_Code[5:0],LoadImm,ALUctrl,RegWrite);

  padding X5(Instruction_Code[20:0],padded_result);

  mux2to1 X6(result,padded_result,LoadImm,muxout);
endmodule
```

6.7) Test the processor design by instializing the instruction memory with a set of instructions (atleast 5 instructions). List below the instructions

Instructions;

Li r1,8
Add r0,r1,r2
Sub r4,r5,r6
And r8,r9,r10
Or r9,r8,r10
Srl r13,r9,10
Li r3,8

6.8) Verify the register file is getting updated accordingly to your sequence of instructions (mentioned earlier)



Unrelated questions:

What were the problems you faced during implementation?

Answer: no problems.

Did you implement the processor on your own? If you took help from others whose help did you take? Which part of design?

Answer: Implemented on my own

Honor Code Declaration by student:

My answers to the above questions are my own work.

I have not shared the codes/answers written by me with any other students. (I might have helped clear doubts of other students).

I have not copied other's code/answers to improve my results. (I might have got some doubts cleared from other students).

Name: P. SURYACHANDRA PRASAD DATE:29-3-2018

ID.NO: 2014B2A30949G