Akhil Jain 14CS10003
Shubham Jain 14CS10043
Controller Specifications

Control Signals :-
reset : Restart the CPU.

after4goto6,after1goto21,after17goto3,after3goto12,after3goto14,after4goto15,after6goto20,
after6goto9,after12goto14,after8goto11,after6goto17,after14goto18,after10goto7 : Interior Signals for state transition.

ldPC : Load the value in Z-bus to Program Counter (PC).

ldIR : Load the value in (outdata = M[MAR]) to Instruction Register (IR).

ldMAR : Load the value of Z-bus to Memory Address Register (MAR).

rd_mem : Read the memory i.e., M[MAR] is loaded in memory output to be loaded in Memory Data Register (MDR) or IR.

wr_mem : Write the value in indata in M[MAR].

ldtmp : Load the value in Z-bus to the temporary register.

ldMDRZ : Load the value in Z-bus to MDR.

ldMDRdata : Load the value in memory output to MDR.

wr_reg : Write the value in Z-bus to the register in Write Port Address (wr_regA).

rd_reg : Read the value from the register in Read Port Address (rd_regA).

ldALU : Load the value to Z-bus according to the Z=ALU(X,Y,fsel).

ldXPC : Load the value in PC to the XY-bus.

ldYPC : Load the value in PC to the XY-bus, which on next clock would go to the Y Port of ALU.

ldXtmp : Load the value in temp register to the XY-bus.

ldYtmp : Load the value in temp to the XY-bus, which on next clock would go to the Y Port of ALU.

ldXreg : Load the value in read register output to the XY-bus.

ldYreg : Load the value in read register output to the XY-bus, which on next clock would go to the Y Port of ALU.

ldXmem : Load the value in MDR to the XY-bus.

ldYmem : Load the value in MDR to the XY-bus, which on next clock would go to the Y Port of ALU.

ldXtmp2 : Load the value 16'b0000000000000010 to the XY-bus.

ldYtmp2 : Load the value 16'b0000000000000010 to the XY-bus, which on next clock would go to the Y Port of ALU.

wr_regA : The value of the write port address in the register bank.

rd_regA : The value of the read port address in the register bank.

fsel : The value of the fsel in the ALU. (fsel-ALU operation : 0-Add 1-Sub 2-And 3-Or 4-Cmp 5-Mns 6-TransX 7-TransY).

C : The Carry signal output of the ALU.

V : The Overflow signal output of the ALU.

S : The Sign signal output of the ALU.

Z_det : The Zero detection signal output of the ALU.

opc : The value is IR[31:25]. Encoding given with the opcode encoding.

opd1 : IR[24:22]

opd2 : IR[21:19]

opd3 : IR[18:16]

state : State Encoding is given below :-
0 : Reset state.
1 : Y <= PC.
2 : MAR <= Z = ALU(Y).
3 : outdata <= M[MAR].
4 : (IR/MDR) <= outdata, PC <= Z = ALU(2 + Y).
5 : Decode Instruction according to the Opcode encoding.
6 : Y <= MDR.
7 : X = Ropd1, Ropd1 <= Z = ALU(X+Y).
8 : Y <= Ropd2.
9 : X = Ropd2, temp <= Z = ALU(X+Y).
10 : Y <= temp.
11 : X = Ropd3, MAR <= Z = ALU(X+Y).
12 : MDR <= outdata.
13 : X = MDR, temp <= Z = ALU(X).
14 : X = MDR, MAR <= Z = ALU(X)
15 : X = MDR, Ropd1 <= Z = ALU(X).
16 : X = Ropd2, Ropd1 <= Z = ALU(X).
17 : X = Ropd1, MAR <= Z = ALU(X+Y).
18 : X = Ropd2, MDR <= Z = ALU(X).
19 : M[MAR] <= MDR.
20 : X = PC, PC <= Z = ALU(X+Y).
21 : Ropd1 <= Z = ALU(Y), MAR <= Y.
22 : X = Ropd1, PC <= Z = ALU(X).
23 : Reset state.

next_state : next_state to go after clock edge, logic is according to the state,opcode and after-goto- signal.