M1 : 0-from PC+1 1-from NPC

M2: 0-from PE 1-from Instr Memory

\*\*\*We will have to change implementation of PE such that index contains the whole 16-bit instruction

M3: 1-from Control Signals 0-from “all zeroes” (control signal is a flag raised by HDU in case of need for stall)

M4: 0-from I(8-6) 1-from addr in PE

M5: 0-from I(5-3) 1-from I(11-9)

M6: 0-from I(11-9) 1-from addr in PE

M7: 00-from SE10 01-from SE7 10-from ZE

M8: 0-from +1 adder of PE 1-from D2

ALU : 00- add 01-equality 10-nand

M9: 00- from D2; 01-from PC in pipeline register; 11 - from Imm16

M10: 00- from D1; 01- from PC in pipeline register; 10: - from +1 ;11 - 0 line

M11:00 - from M9; 01 - from ALU result pipeline register; 11 -- from DATA WB pipeline register

M12: 00:- from M10; 10: from ALU result pipeline register; 11: from DATA WB pipeline register;

M13: - 00: - ALU out; 01: - from M12 ; 11: - from adder for PC + IMM6

M14: 0: from D2 of EX/MEM pipeline register, 1: from D1 of EX/MEM pipeline register

M15: 0: - Data memory output , 1: from alu result from EX/MEM pipeline register

M16: 00:- NPC write back ; 01: from Data in MA/WB pipeline register; 11: -from PC

M17: 00-from IMM 01-from +1 10-from +0

M18: 0- from ALU 1-from small adder

M19: 0-from PC 1-from M11\_out

M20: 0-from ALUResult 1-from D2