# CS-226 : Digital Logic Design Project

| Name | Roll No. |
|---|---|
| Aditya Jain | 203050003 |
| Fenil Gaurang Mehta | 203050054 |
| Sagar Tyagi | 203050008 |
| Vipin Mahawar | 203050110 |
| Amit Hari | 203050115 |

## IITB_Proc

- 16 bit Computer System
- 8 Register (R0 - R7) : General Purpose
- Condition Code Register ( With 2 Flags - Carry Flag (c) and Zero Flag (z) )
- Use Point-to-point communication infrastructure
- Memory is word Addressible (Cell size is 16 bits)
- Supports 15 Instructions
- Instructions are of 3 Types.
  1) R Type Instruction format
  2) I Type Instruction format
  3) J Type Instruction format

## Design Document

1) R Type Instruction Format

| 15 - 12 | 11 - 9 | 8 - 6 | 5 - 3 | 2 | 1 - 0 |
|---|---|---|---|---|---|
| Opcode (4 bit) | Reg A (RA) (3 bit) | Reg B (RB) (3 bit) | Reg C (RC) (3 bit) | Unused (1 bit) | Condition (CZ) (2 bit) |

2) I Type Instruction format

| 15 - 12 | 11 - 9 | 8 - 6 | 5 - 0 |
|---|---|---|---|
| Opcode (4 bit) | Reg A (RA) (3 bit) | Reg C (RC) (3 bit) | Immediate (6 bit Signed) |

3) J Type Instruction format

| 15 - 12 | 11 - 9 | 8 - 0 |
|---|---|---|
| Opcode (4 bit) | Reg A (RA) (3 bit) | Immediate (9 bit Signed) |

# R Type Instructions Format

ADD, ADC, ADZ, NDU, NDC, NDZ (6 Instructions)

**$S_0$**

| Data Transfer | Control Signals |
|---|---|
| PC → Mem_Addr | • MR |
| Mem_Out → IR | • IR_W |

(MR - Memory Read
IR_W - Instruction Register Write
IR - Instruction Register

**$S_1$**

| | |
|---|---|
| $IR_{11-9}$ → RF_A1 | • T1_W |
| $IR_{8-6}$ → RF_A2 | • $T_2$ – W |
| RF_D1 → T1 | |
| RF_D2 → $T_2$ | |

(RF_A1 - Register file Address 1
RF_A2 - Register file Address 2
RF_D1 - Register file Data 1
RF_D2 - Register file Data 2
T1_W - T1 Write
T2_W - T2 Write

**$S_2$**

| | |
|---|---|
| $T_1$ → ALU_A | • ALU_OPN → AND if opcode is 0000 or 0001; NAND if opcode is 0010 |
| $T_2$ → ALU_B | • T3_W |
| ALU_C → $T_3$ | |
| $ALU\_C_{(16)}$ → C (Flag) | |
| $ALU\_C_{15-0} == 0$ → Z(Flag) | |

**$S_3$**

| | |
|---|---|
| $T_3$ → RF_D3 | • RF_W |
| $IR_{5-3}$ → RF_A3 | |

(RF_D3 - Register file Data 3
RF_A3 - Register file Address 3
RF_W - Register file Write

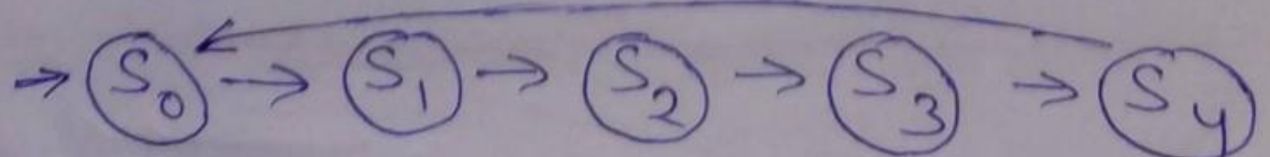
**$S_4$**

| | |
|---|---|
| PC → ALU_A | • ALU_ADD |
| +1 → ALU_B | • PC_W |
| ALU_C → PC | |

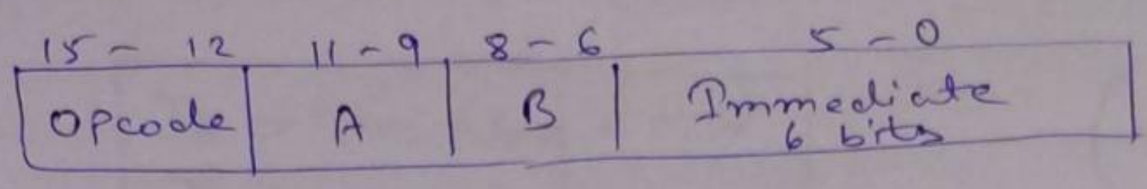
∵ Memory is word Addressible

(PC_W - PC Write)

State transition Diagram



# I Type Instruction format




## 1) ADI

| | Data Transfer | Control Signals |
|---|---|---|
| $S_5$ | $IR_{11-9} \rightarrow RF\_A1$<br>$RF\_D1 \rightarrow T_1$<br>$IR_{5-0} \rightarrow SE6 \rightarrow T_2$ | • $T_1 - W$<br>• $T_2 - W$ |

(SE6 – Sign Extender 6 bits)

| | | |
|---|---|---|
| $S_6$ | $T_3 \rightarrow RF\_D3$<br>$IR_{8-6} \rightarrow RF\_A3$ | • RF_W |

**State Transition Diagram**

→ $S_0$ → $S_5$ → $S_2$ → $S_6$ → $S_4$

## 2) LW

| | | |
|---|---|---|
| $S_7$ | $T_2 \rightarrow ALU\_A$<br>$IR_{5-0} \rightarrow SE6 \rightarrow ALU\_B$<br>$ALU\_C \rightarrow T_2$ | • ALU_ADD<br>• $T_2 - W$ |
| $S_8$ | $T_2 \rightarrow Mem\_Addr$<br>$Mem\_Out \rightarrow T_3$<br>$(T_3 == 0) \rightarrow Z$ | • MR<br>• $T_3 - W$ |
| $S_9$ | $IR_{11-9} \rightarrow RF\_A3$<br>$T_3 \rightarrow RF\_D3$ | RF_W |

($T_3 - W - T_3$ Write)

**State Transition Diagram**

→ $S_0$ → $S_1$ → $S_7$ → $S_8$ → $S_9$ → $S_4$

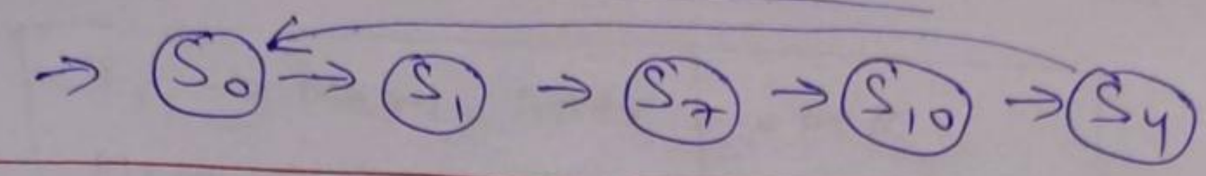
3) SW

$S_{10}$

| | |
|---|---|
| $T_2 \Rightarrow Mem\_Addr$<br>$T_1 \Rightarrow Mem\_In$ | $\cdot MW$ |

(MW - Memory Write)

State Transition Diagram

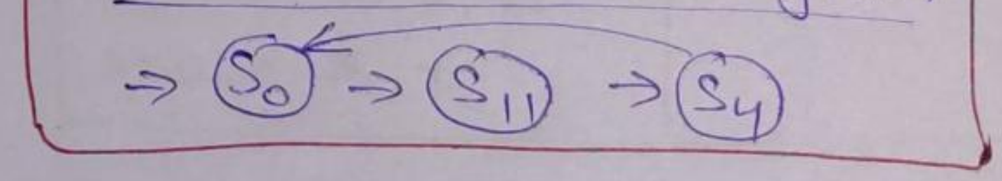



4) LHI (I Type Instruction)

$S_{11}$

| | |
|---|---|
| $\vec{O}_{7bits} \rightarrow RR\_D_3$<br>$IR_{8-0} \Rightarrow RF\_D_3$ (15-7)<br>$IR_{11-9} \Rightarrow RF\_A_3$ | $RF\_W$ |

State Transition Diagram

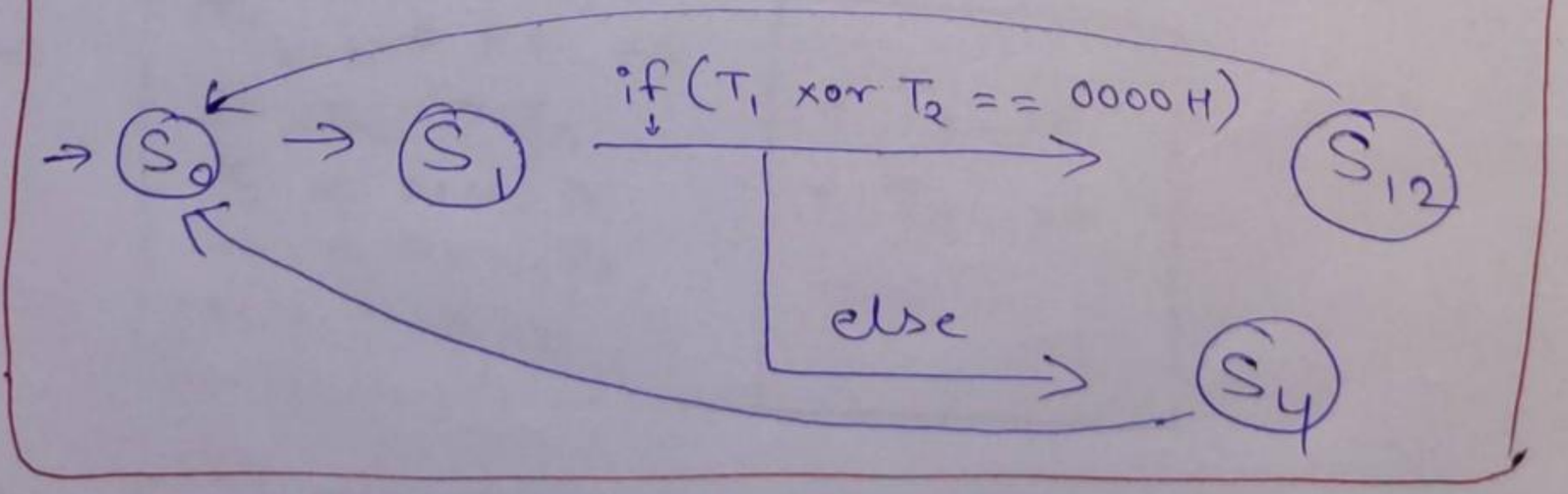



5) BEQ

$S_{12}$

| | |
|---|---|
| $PC \rightarrow ALU\_A$<br>$IR_{5-0} \Rightarrow SE_6 \Rightarrow ALU\_B$<br>$ALU\_C \Rightarrow PC$ | $ALU\_Add$<br>$T_3 - W$ |

State Transition Diagram



# J Type Instruction Formats

## 1) LA

**$S_{13}$**

| Data Transfer | Control Signals |
|---|---|
| $\vec{0} \rightarrow T_2$ | • $T_1$ – W |
| $IR_{11-9} \rightarrow RF\_A1$ | • $T_2$ – W |
| $RF\_D1 \rightarrow T_1$ | |

**$S_{14}$**

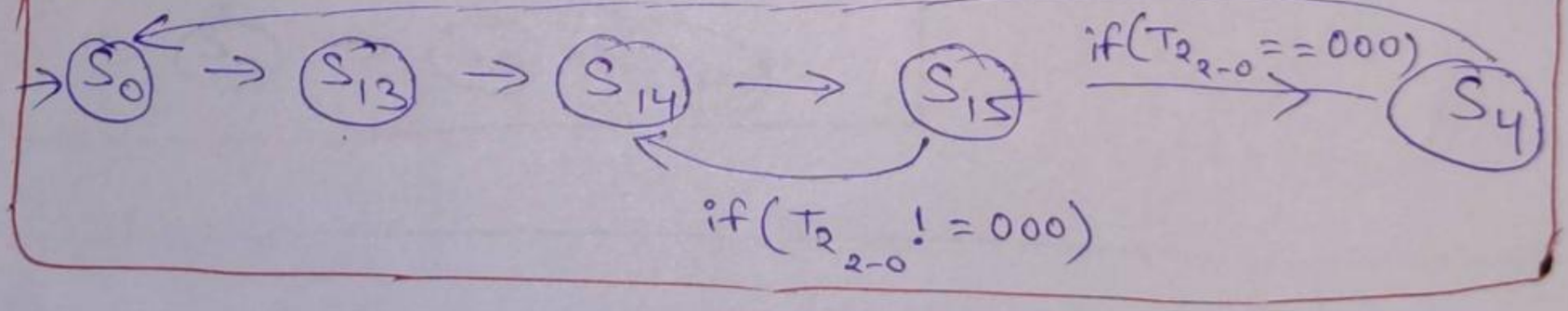
| | |
|---|---|
| $T_1 \rightarrow Mem\_Addr$ | • MR |
| $Mem\_Out \rightarrow T_3$ | • $T_3$ – W |
| $T_1 \rightarrow ALU\_A$ | • $T_1$ – W |
| $+1 \rightarrow ALU\_B$ | |
| $ALU\_C \rightarrow T_1$ | |

(Writing occurs at end of clock cycle so no problem in reading & writing $T_1$)

**$S_{15}$**

| | |
|---|---|
| $T_{2_{2-0}} \rightarrow RF\_A3$ | • RF_W |
| $T_3 \rightarrow RF\_D3$ | • $T_2$ – W |
| $T_2 \rightarrow ALU\_A$ | |
| $+1 \rightarrow ALU\_B$ | |
| $ALU\_C \rightarrow T_2$ | |

### State Transition Diagram




## 2) SA

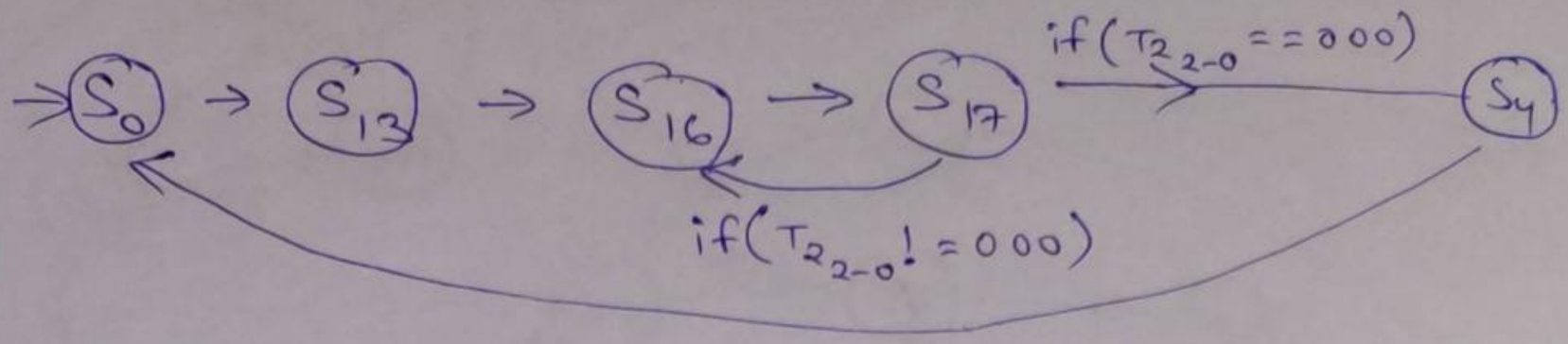
**$S_{16}$**

| | |
|---|---|
| $T_{2_{2-0}} \rightarrow RF\_A1$ | • $T_3$ – W |
| $RF\_D_1 \rightarrow T_3$ | • $T_2$ – W |
| $T_2 \rightarrow ALU\_A$ | |
| $+1 \rightarrow ALU\_B$ | |
| $ALU\_C \rightarrow T_2$ | |

**$S_{17}$**

| | |
|---|---|
| $T_1 \rightarrow Mem\_Addr$ | • MW |
| $T_3 \rightarrow Mem\_In$ | • $T_1$ – W |
| $T_1 \rightarrow ALU\_A$ | |
| $+1 \rightarrow ALU\_B$ | |
| $ALU\_C \rightarrow T_1$ | |

State Transition Diagram

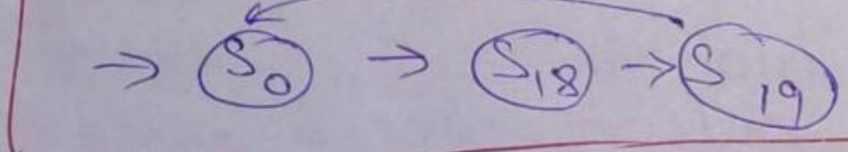



3) JAL

| | Data Transfer | Control Signal |
|---|---|---|
| $S_{18}$ | PC → RF_D3<br>$IR_{11-9}$ → RF_A3<br>$IR_{8-6}$ → RF_A2<br>RF_D2 → $T_2$ | • RF_W<br>• $T_2$_W |

| | | |
|---|---|---|
| $S_{19}$ | PC → ALU_A<br>$IR_{8-0}$ → SE9 → ALU_B<br>ALU_C → PC | • ALU_ADD<br>• PC_W |

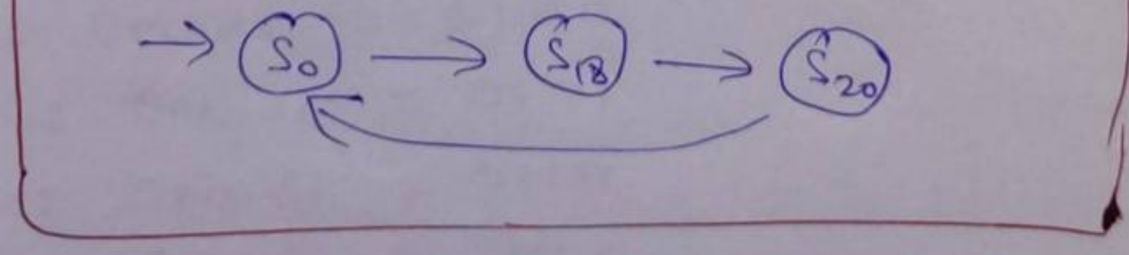
State Transition Diagram

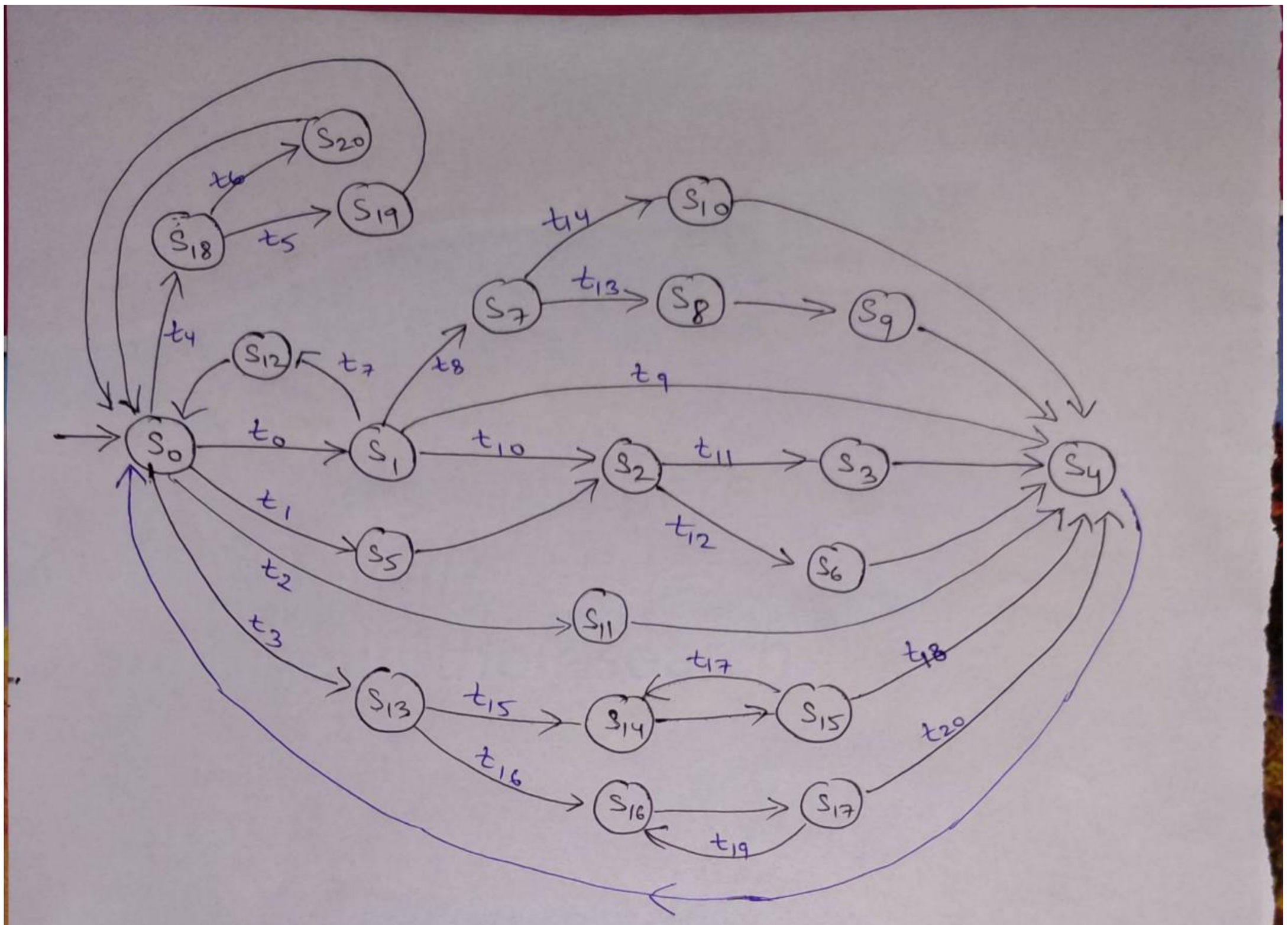



4) JLR

| | | |
|---|---|---|
| $S_{20}$ | $T_2$ → PC | PC_W |

State Transition Diagram






$t_0$ : Opcode = $\overline{0001 \text{ or } 0011 \text{ or } 0110 \text{ or } 1000 \text{ or } 0111 \text{ or } 1001}$

$t_1$ : Opcode = 0001

$t_2$ : Opcode = 0011

$t_3$ : Opcode = 0110 or 0111

$t_4$ : Opcode = 1000 or 1001

$t_5$ : Opcode = 1000

$t_6$ : Opcode = 1001

$t_7$ : Opcode = 1100 and ($T_1$ xor $T_2$ == 0000H)

$t_8$ : Opcode = 0100 or 0101

$t_9$ : [Opcode = 1100 and $\overline{(T_1 \text{ xor } T_2 == 0000H)}$] or $\overline{t_{10}}$

$t_{10}$ : (Opcode = 0000 or 0010) . [$IR_{1-0}$ == 0 or ($IR_{1-0}$ == 10) C or ($IR_{1-0}$ == 01) Z]

$t_{11}$ : Opcode = 0000 or 0010

$t_{12}$ : Opcode = 0001

$t_{13}$ : Opcode = 0100

$t_{14}$ : Opcode = 0101

$t_{15}$ : Opcode = 0110

$t_{16}$ : Opcode = 0111

$t_{17}$ : $T_{2_{2-0}}$ != 000

$t_{18}$ : $T_{2_{2-0}}$ == 000

$t_{19}$ : $T_{2_{2-0}}$ != 000

$t_{20}$ : $T_{2_{2-0}}$ == 000

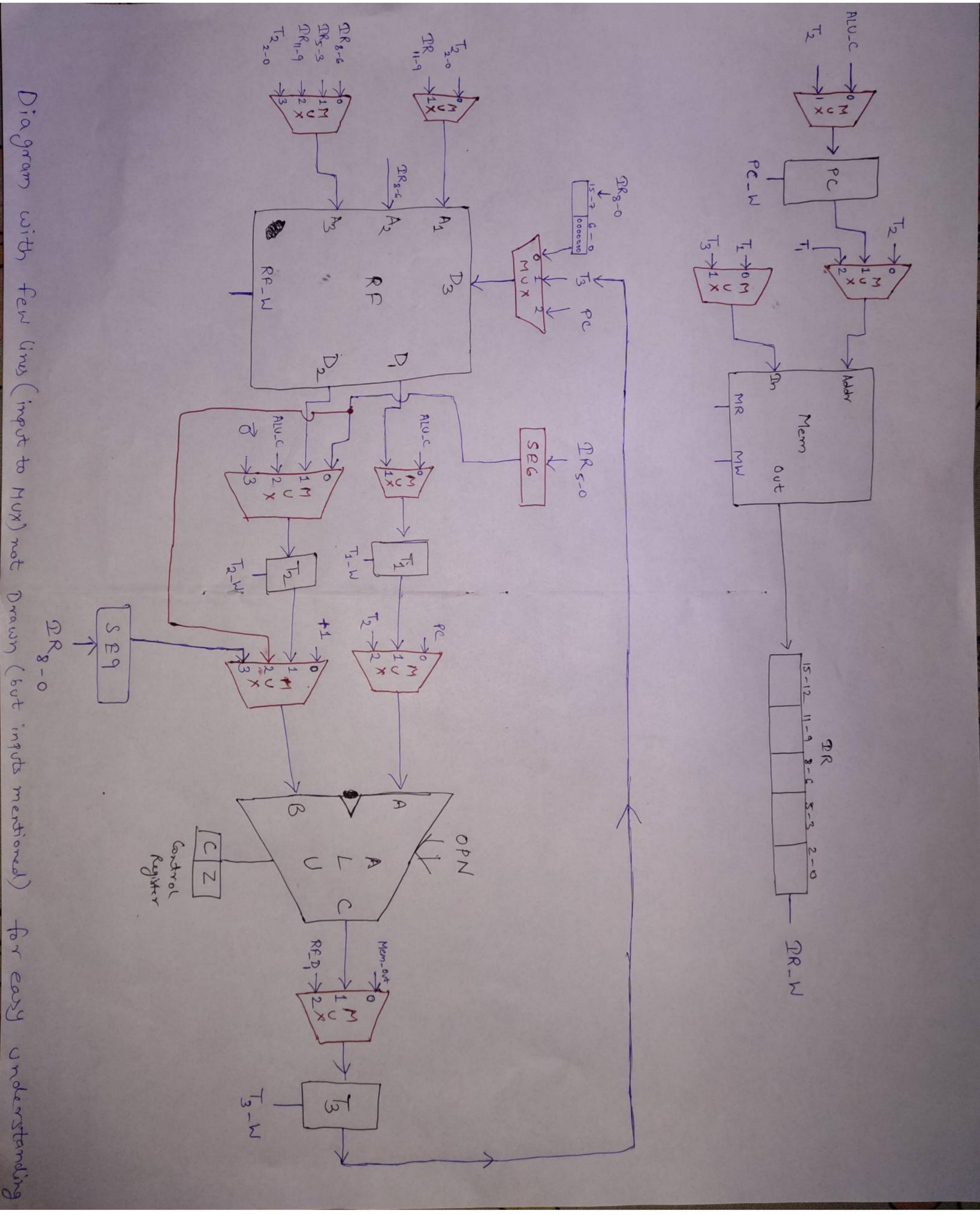



Diagram with few lines (input to MUX) not drawn (but inputs mentioned) for easy understanding

Diagram with every connections connected