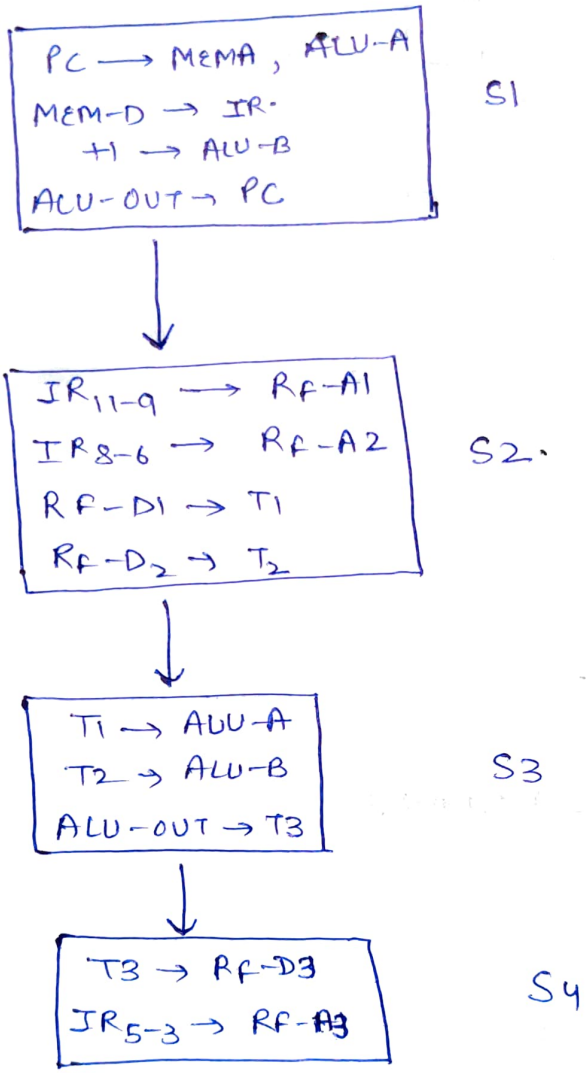
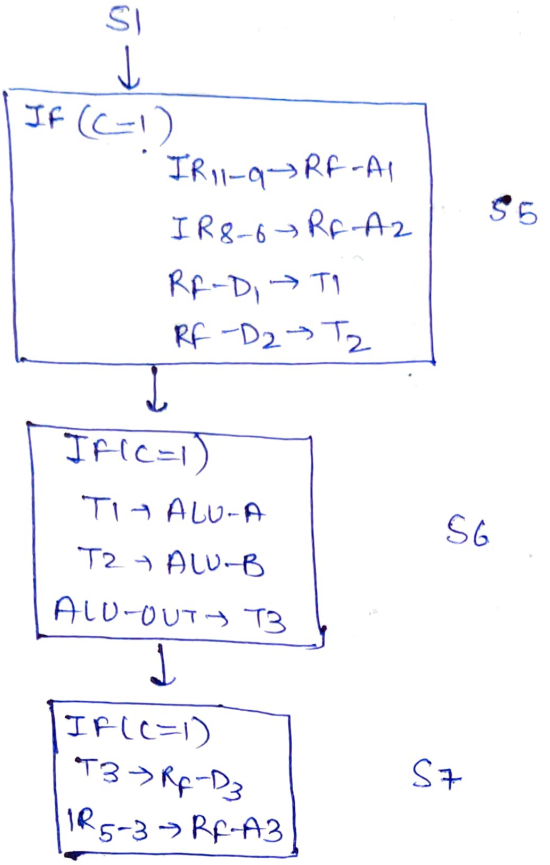
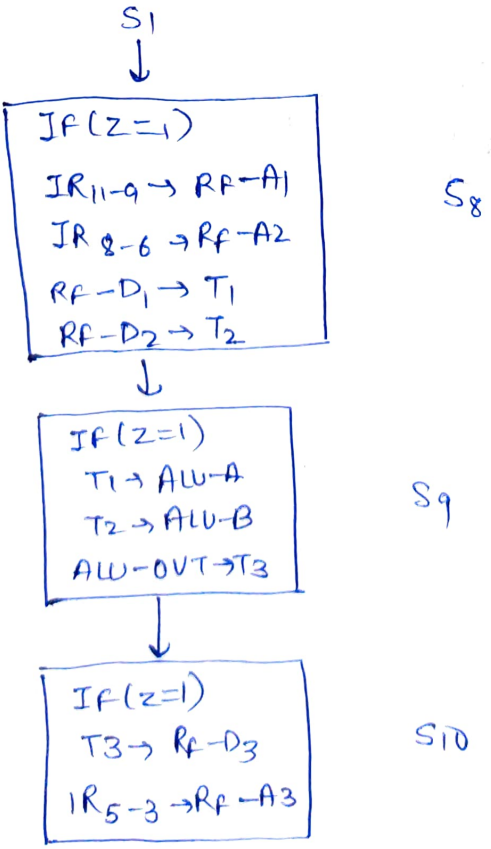
1. ADD RC, RA, RB / 6. NDU RC, RA, RB



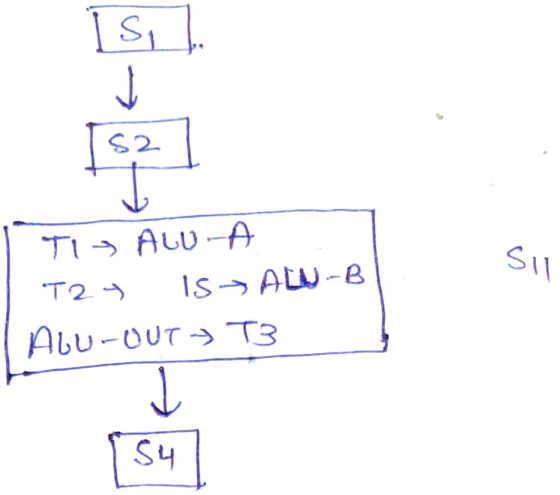
2. ADC R_C, R_A, R_B / 7. NDC R_C, R_A, R_B



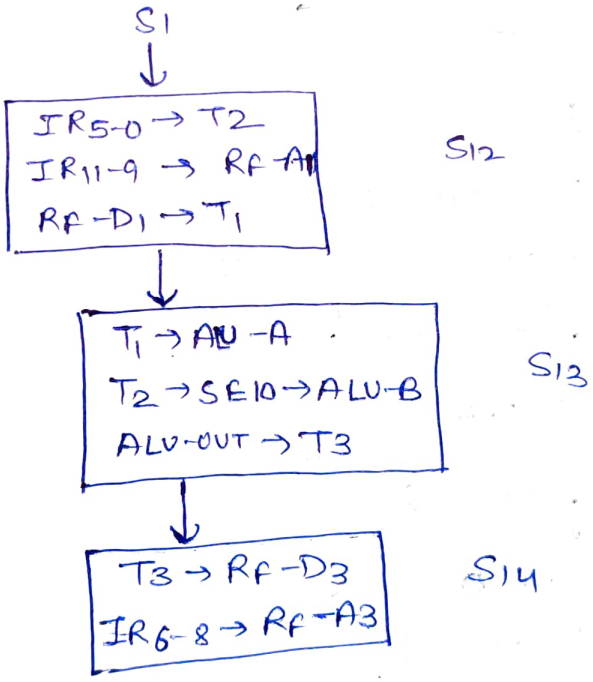
3. ADZ R_C, R_A, R_B / 8. NDZ R_C, R_A, R_B



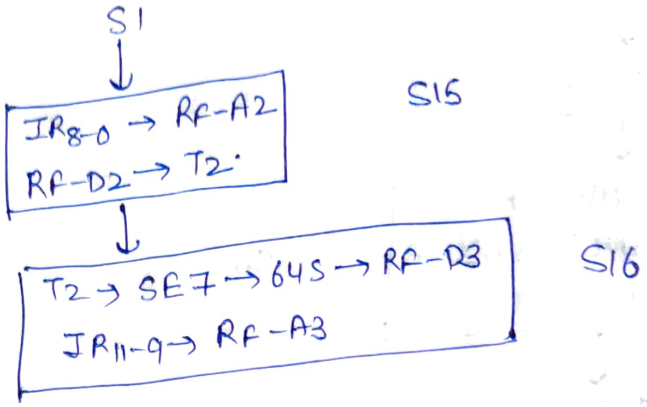
4. ADL R_C, R_A, R_B



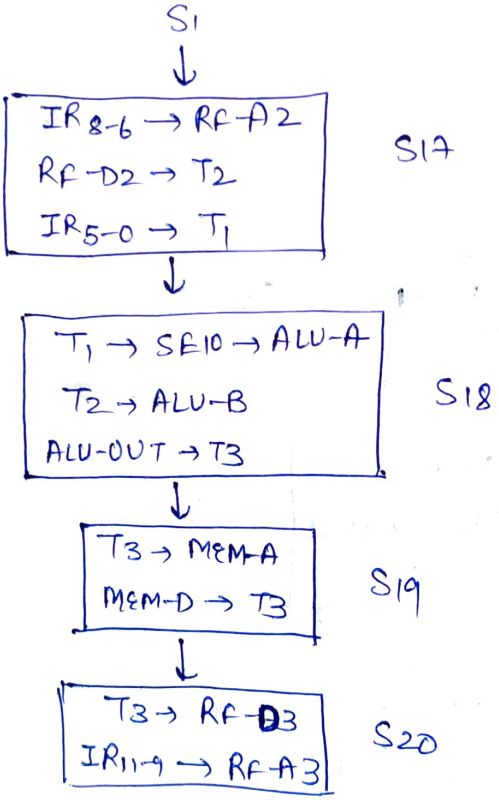
5. ADI $R_B, R_A, IMM6$



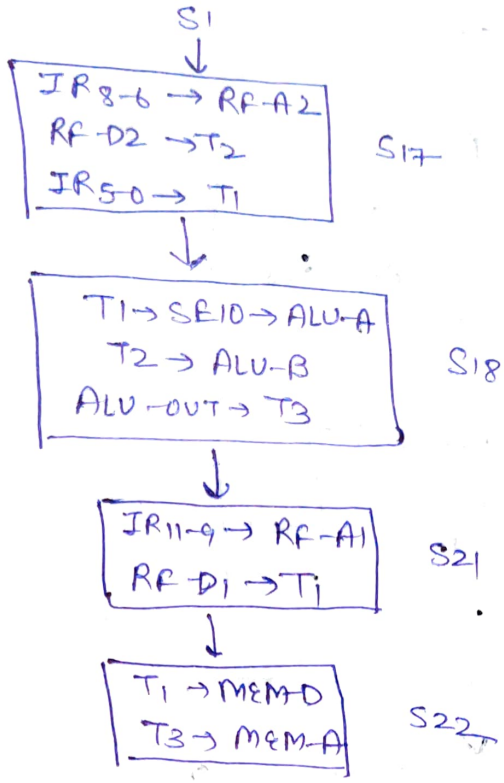
9. LHI RA, IMM



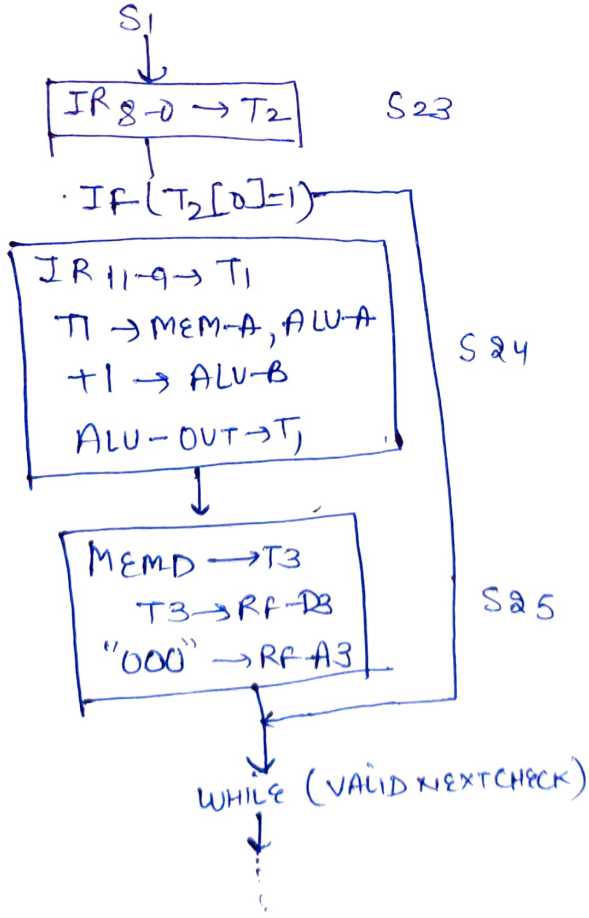
10. LW RA, RB, IMM



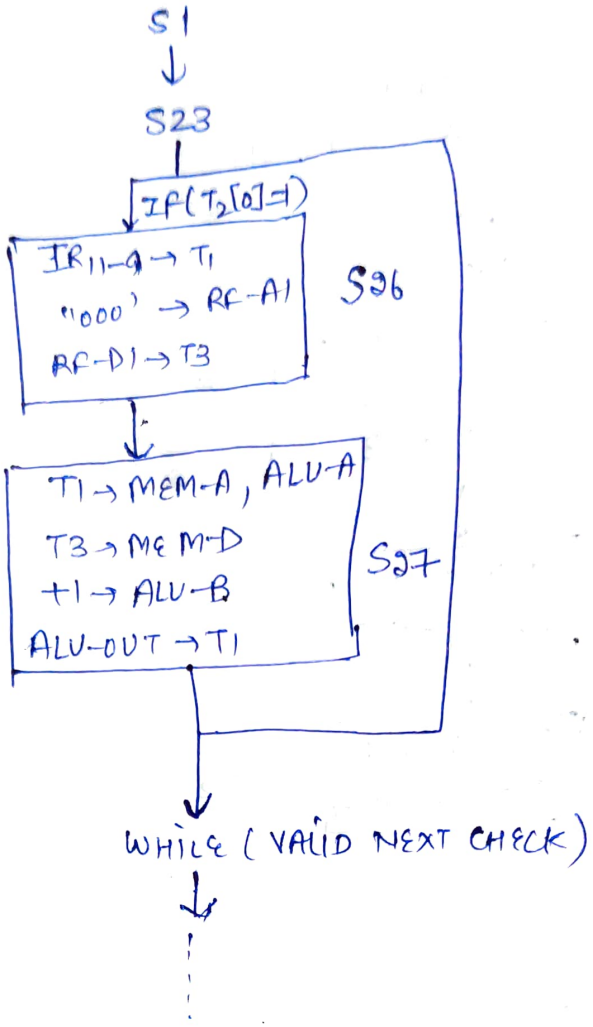
11. SW RA, RB, IMM



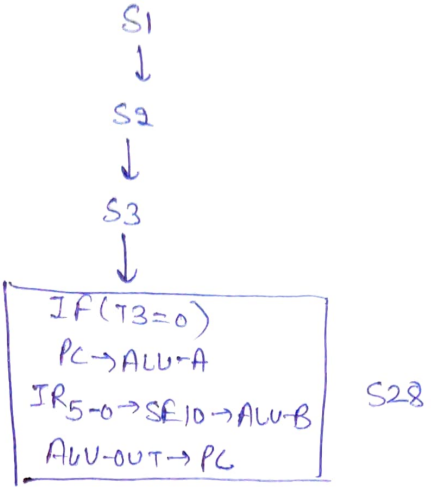
12. LM RA, IMM



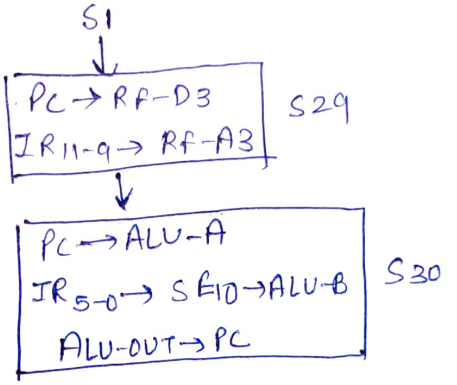
13. SM RA, IMM


S1
S23
IF(T2[0]=1)
IR11-9 → T1
"000" → RF-A1
RF-D1 → T3
S26
T1 → MEM-A, ALU-A
T3 → MEM-D
+1 → ALU-B
ALU-OUT → T1
S27
WHILE (VALID NEXT CHECK)

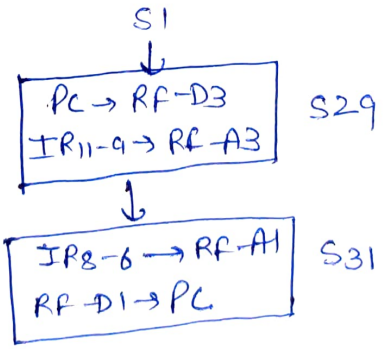
14. BEQ RA, RB, IMM



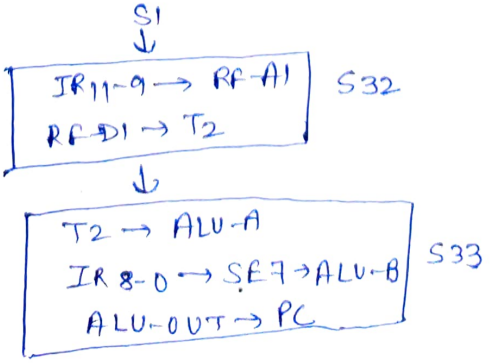
15. JAL RA, IMM



16. JLR RA, RB



17. JRI RA, IMM



CONTROL STORE

1) ALU

The ALU performs basic operations like add, OR, XOR, AND etc. by taking two inputs ALU-A & ALU-B of 16 bits each. It's output is represented as ALU-out. The instructions are given directly & multiple control lines are used to control the activities inside ALU.

2) TEMPERORY REGISTERS (T_1, T_2, T_3)

Temporary registers store the values for future use. They are all of 16-bits each.

3. MEMORY (MEM)

Memory stores various data at particular locations called addresses. We can access data by providing the address as input (16-bit) through MEM-A & accessing it through MEM-D. We can also store data in memory by providing data to be stored in MEM-D and address in MEM-A.

4. SE7, SE10 (SIGN EXTENDER)

The sign extender places the corresponding number of zeros (7 & 10 resp) to the left of input so as to make it a 16-bit input.

5. IS & 6US

This is basically used to shift the input by n bits to the left (1 & 7 resp). The digits which are pushed off are eliminated & zeros are added to the right.

6. REGISTER FILE (RF)

It includes all the registers (R0 to R7). It has 3 address selection lines (A_1, A_2, A_3) and 3 data output lines D_1, D_2 & D_3. To access a values inside register, we have to pass the address of that register in one of the two address lines A_1 or A_2. and the corresponding output is received at D_1 or D_2

To store a data in a register the data is sent to the D_3 selection line & the address of the register is sent as input in A_3.

R7 register stores the value of Program Counter (PC)

7. INSTRUCTION REGISTER (IR)

IR is a special purpose register, which is used to receive the 16 bits of instruction. There are three different types of Instruction format (R, I & J type) which allot bits differently to various operands. Following is the bit distribution

R-type

$R_C \rightarrow IR_{5-3}$

$R_B \rightarrow IR_{8-6}$

$R_A \rightarrow IR_{11-9}$

I-type

$IMM \rightarrow IR_{0-5}$

$R_B \rightarrow IR_{8-6}$

$R_A \rightarrow IR_{11-9}$

J-type

$IMM \rightarrow IR_{8-0}$

$R_A \rightarrow IR_{11-9}$