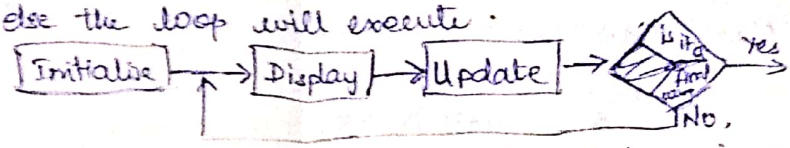
Microprocessor

Counter and Time delay

Counter:

A counter is designed simply by loading an appropriate number into one of the register and using the INR or DCR instructions.

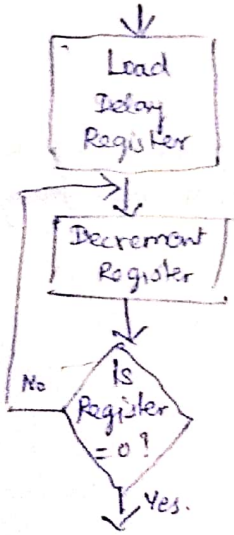
A loop is established to update the count, and each count is checked to determine whether it has reached that number; else the loop will execute.



Time Delay:

* The procedure used to design a specific delay is similar to that used to set up a counter.
* A register is loaded with a number, depending on the time delay required, and then the register is decremented until it reaches zero by setting up a loop with a condition loop instruction.

Time delay Using One Register:



Let us take an example instruction:

			T state
	MVI	C, FFH	7
Loop:	DCR	C	4
	JNZ	LOOP	10/7.

Usually, An 8085 based computer with 2 MHz clock frequency will execute the instructions MVI, DCR as follows.

Clock frequency of a system (8085) f = 2 MHz.

Clock period $T = 1/f = \frac{1}{2} \times 10^{-6}\,s = 0.5\,\mu s$

Time to execute MVI instruction

$T = 7 \text{ (T-states)} \times 0.5$

$= 3.5\,\mu s.$

If clock frequency has 1 MHz in system, and the μ
May require 7 μs to execute the same instruction.

Time delay depends on T-states and no. of time instruction is executed.

Time delay in the loop with 2 MHz clock frequency
Calculated as,

$T_L = T \times \text{loop T states} \times N_{10}$

T_L → Time delay T → System clock period

N_{10} → Equivalent number of hexa. count loaded

$T_L = 0.5\times10^{-6} \times (4+10) \times 255$

$= 0.5\times10^{-6} \times 14 \times 255$

$= 1785\ \mu s = 1.8 ms.$

$T_L = 1.8 ms$

A time delay to execute the loop,

$T_{LA} = T_L - (3\ T\text{states} \times \text{clock period})$

~~1.8ms~~

$= 1785\ \mu s - (1.5\ \mu s) = 1783.5 \mu s.$

~~3 T-states~~,

The total time delay,

$T_D = T_0 + T_{LA}$ [T_0 → Time delay outside the loop, T_{LA} → Time delay inside loop in ~~to execute~~ loop instruction]

$= (7 \times 0.5) + (1783.5)$

$= 3.5 + 1783.5$

$= 1787 \mu s \approx 1.8 ms.$

$T_D = 1.8 ms$

Time Delay Using a Register Pair.

Let us consider this example.

	Mnemonic		T states
	LXI	B, 2384H	10
Loop:	DCX	B	6
	MOV	A, C	4
	ORA	B	4
	JNZ	Loop	10/7.

Note: In LXI, 23H stores in B and 84H stores in C.

Increment in 2384 - 1 = 2383H.

~~let us take~~

We load 2384H into BC Register pair, such that

B → 23H C → 84H

As it is hexadecimal, let us convert into decimals,

$2384 = 2\times16^3 + 3\times16^2 + 8\times16^1 + 4\times16^0$

$= 9092_{10}$

T = 0.5 μs

$\therefore T_L = 0.5 \times 24 \times 9092_{10}$ μsec

$= 109$ msec (without last cycle)

Total time delay,

$T_D = T_L + T_0$

~~= 109t~~

= 109 ms (The instruction LXI adds only 5 μs.)

(3). Time delay using a loop within a loop technique.

Example:

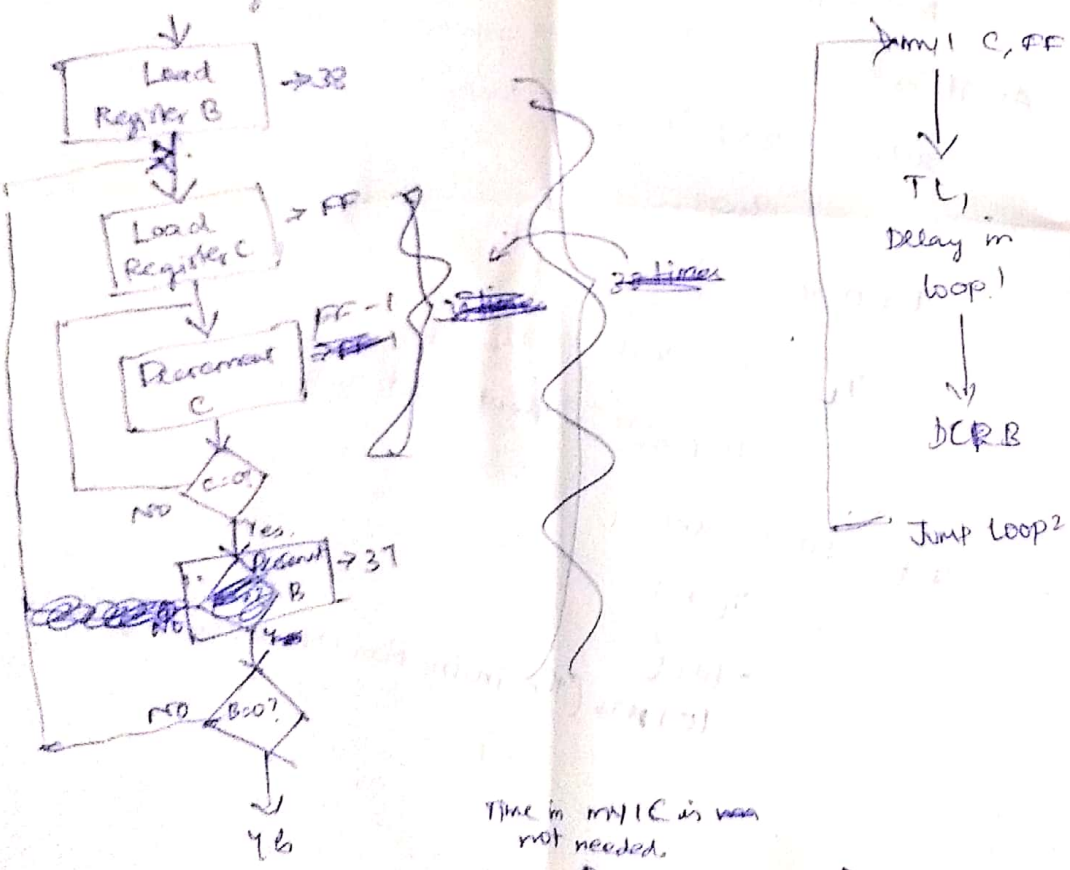
```
        MVI  B, 38H   (7T)
Loop 2: MVI  C, FFH   (7T)
Loop 1: DCR  C        (4T)
        JNZ  Loop 1   (10/7T)
        DCR  B        (4T)
        JNZ  Loop 2   (10/7T)
```

38
8 × 16^0 = 8
3 × 16^1 = 48
56

Delay Calculations

Delay in loop 1 is T_{L1} = 1783.5 μs (as in ① type)

This loop executes 56 times because of the count 38H in Register B [$38_H = 56_D$]



T_{L2} = 56 (T_{L1} + 21 T states × 0.5 μs)

= 56 (1783.5 μs + 10.5

= 100.4 ms.