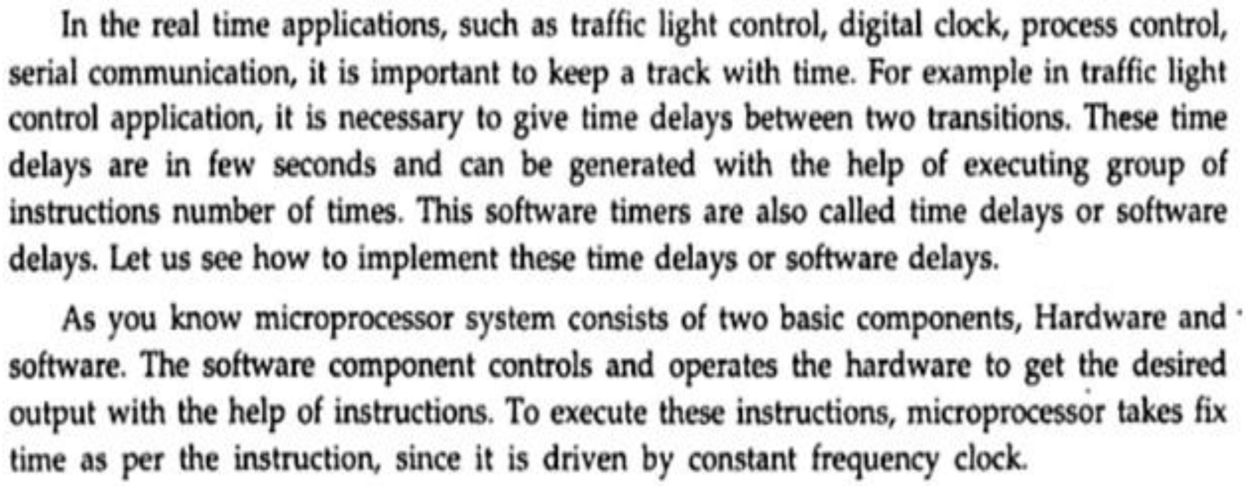
Microprocessor and its Applications Unit 3

Counters & Time Delays

3.1 Counters



In the real time applications, such as traffic light control, digital clock, process control, serial communication, it is important to keep a track with time. For example in traffic light control application, it is necessary to give time delays between two transitions. These time delays are in few seconds and can be generated with the help of executing group of instructions number of times. This software timers are also called time delays or software delays. Let us see how to implement these time delays or software delays.

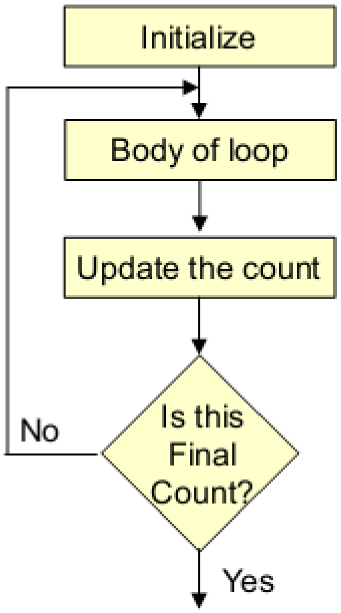
As you know microprocessor system consists of two basic components, Hardware and software. The software component controls and operates the hardware to get the desired output with the help of instructions. To execute these instructions, microprocessor takes fix time as per the instruction, since it is driven by constant frequency clock.

3.1.1 A Single register used Loop Counter

A loop counter is set up by loading a register with a certain value Then using the DCR (to decrement) and INR (to increment) the contents of the register are updated.

• A loop is set up with a conditional jump instruction that loops back or not depending on whether the count has reached the termination count.

• The operation of a loop counter can be described using the following flowchart.



Sample ALP for implementing a loop Using DCR instruction

```
        MVI C, 15H
LOOP : DCR C
        JNZ LOOP
```

3.1.2 Using a Register Pair as a Loop Counter

Using a single register, one can repeat a loop for a maximum count of 255 times.

• It is possible to increase this count by using a register pair for the loop counter instead of the

single register.

– A minor problem arises in how to test for the final count since DCX and INX do not modify the flags .

– However, if the loop is looking for when the count becomes zero, we can use a small trick by ORing the two registers in the pair and then checking the zero flag .

The following is an example of a loop set up with a register pair as the loop counter.

```
        LXI B, 1000H
LOOP: DCX B
        MOV A, C
        ORA B
       JNZ LOOP
```

3.2 Time Delays

•Each instruction passes through different combinations of Fetch, Memory Read, and Memory Write cycles.

What is T-state?

T-state is defined as one subdivision of operation performed in one clock period. These subdivisions are internal states synchronized with the system clock, and each T-state is precisely equal to one clock period.

What is an instruction cycle?

The sequence of operations that a processor has to carry out while executing the instruction is called instruction cycle. Each instruction cycle of processor contains a number of machine cycles

What is a machine cycle?

Machine cycle is defined as the time required to complete one operation of accessing memory input/output, or acknowledging an external request. This cycle may consist of three to six T-states.

• Knowing the combinations of cycles, one can calculate how long such an instruction would require to complete.

• Consider the values of B/M/T for each instruction where:

– B for Number of Bytes

– M for Number of Machine Cycles

– T for Number of T-State.

Knowing how many T- States an instruction requires, and keeping in mind that a T- State is one clock cycle long, we can calculate the time using the following formula:

Delay = No. of T - States / Frequency

• For example a "MVI" instruction uses 7 T- States. Therefore, if the Microprocessor is running at 2 MHz, the instruction would require 3.5μ Seconds to complete

3.2.1 Delay loops

• We can use a loop to produce a certain amount of time delay in a program.

The following is an example of a delay loop:

```
          MVI C,FFH      ----------→7T-States
LOOP      DCR C          ----------→4 T-States
          JNZ LOOP       ----------→10 T-States   sometimes (10/7 T-States)
```

The first instruction (MVI C,FFH) initializes the loop counter and is executed only once requiring only 7 T-States.

The following two instructions form a loop that requires 14 T-States to execute and is repeated 255 times until C becomes 0.

We need to keep in mind though that in the last iteration of the loop, the JNZ instruction will fail and require only 7 T-States rather than the 10.

Therefore, we must deduct 3 T-States from the total delay to get an accurate delay calculation.

• To calculate the delay, we use the following formula:

$$T_{delay} = T_O + T_L$$

– T_{delay} = total delay

– T_O = delay outside the loop

– T_L = delay of the loop

T_O is the sum of all delays outside the loop.

Using these formulas, we can calculate the time delay for the previous example:

T_O = 7 T- States –> Delay of the MVI instruction

T_L = (14 X 255) - 3 = 3567 T- States

– 14 T-States for the 2 instructions repeated 255 times ($FF_{16} = 255_{10}$) reduced by the 3T-States for the final JNZ.

T_{Delay} = T_O + T_L = 7 T + 3567 = 3574 T- States

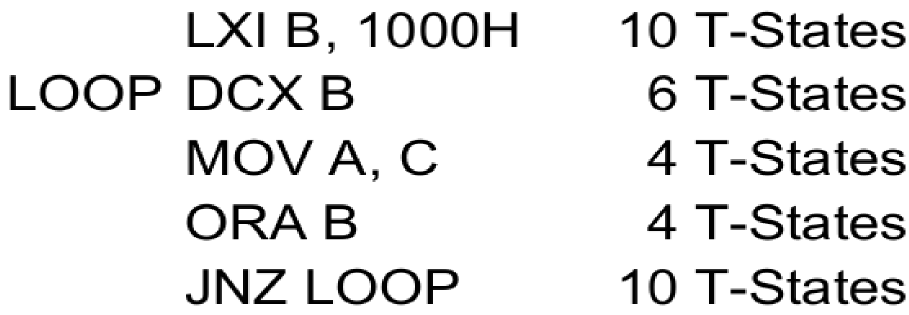
If the Microprocessor frequency is 3Mh what is the delay time?

1T-state =1/f =1/3x10^{-6} seconds = 333n.sec

Delay time is= 3574 x 333n.sec
= 3.574 x 0.333 x 10^{-3} seconds
= 1.190142 x 10^{-3} seconds
= 1. 190142 milliseconds

3.2.2 Using a Register Pair as a Loop Counter

• The following is an example of a delay loop set up with a register pair as the loop counter.



```
     LXI B, 1000H    10 T-States
LOOP DCX B            6 T-States
     MOV A, C         4 T-States
     ORA B            4 T-States
     JNZ LOOP        10 T-States
```

Using the same formula from before, we can calculate:

T_O = 10 T- States

– The delay for the LXI instruction

T_L= (24 X 4096) - 3 = 98301 T- States

– 24 T -States for the 4 instructions in the loop repeated 4096 times (1000_{16} = 4096_{10}) reduced by the 3 T-States for the JNZ in the last iteration.

Total delay = T_O + T_L = 10 + 98301 =98311 T- States

If the operating frequency is 2MHz

What is the maximum delay time =?

Sol: 1T-state =1/f = $1/2X10^6$ = 0.5us

Maximum count will happen for $FFFF_{16}$ or 65535_{10} so,

T_L= (24 X 65535) - 3 = 1572837 T- States

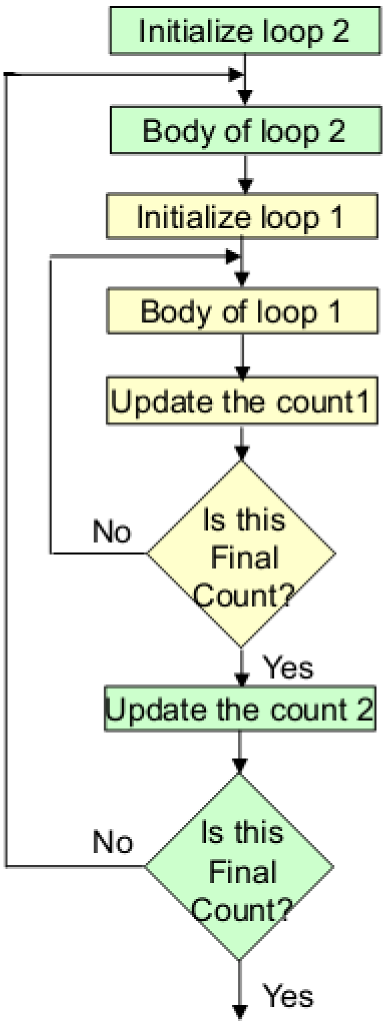
Total delay time = (10 + 1572837 T- States) x0.5us

0.786423Sec

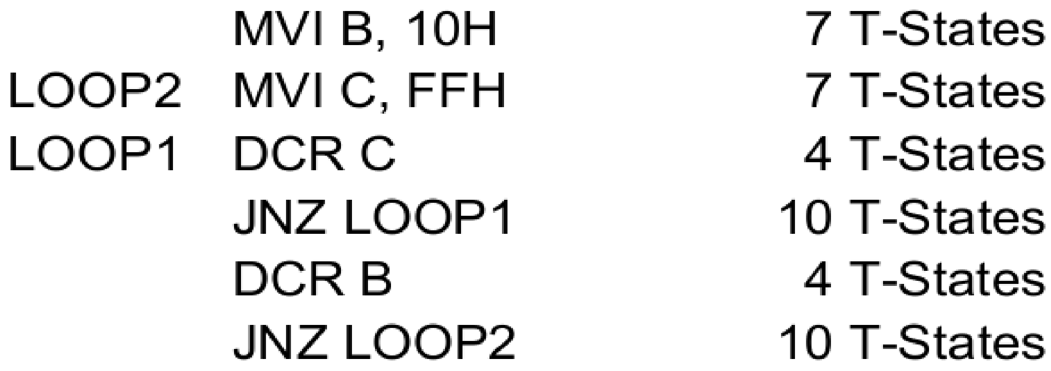
3.2.3 Nested Loops

Nested loops can be easily setup in Assembly language by using two registers for the two loop counters and updating the right register in the right loop.

– In the figure, the body of loop 2 can be before or after loop 1.



Instead (or in conjunction with) Register Pairs, a nested loop structure can be used to increase the total delay produced beyond the maximum delay of 16 bits register delay.



```
        MVI B, 10H         7 T-States
LOOP2   MVI C, FFH         7 T-States
LOOP1   DCR C              4 T-States
        JNZ LOOP1         10 T-States
        DCR B              4 T-States
        JNZ LOOP2         10 T-States
```

Delay Calculation of Nested Loops

• The calculation remains the same except that it the formula must be applied recursively to each loop.

– Start with the inner loop, and then plug that delay in the calculation of the outer loop

i) Delay of inner loop

– T_{O1}= 7 T-States

• MVI C, FFH instruction

– T_{L1} = (255 X 14) - 3 = 3567 T-States

• 14 T-States for the DCR C and JNZ instructions repeated 255 times ($FF_{16} = 255_{10}$) minus 3 for the final JNZ

T_{LOOP1} = 7 + 3567 =3574 T-states

ii) Delay of outer loop

– T_{O2} = 7 T-States

• MVI B, 10H instruction

– T_{L2}= (16 X (14 + 3574)) - 3 = 57405 T-States

• 14 T-States for the DCR B and JNZ instructions and 3574

T-States for loop 1 repeated 16 times ($10_{16} = 16_{10}$) minus 3 for the final JNZ.

– T_{Delay} = 7 + 57405 = 57412 T-States

• Total Delay

– T_{Delay} = 57412 X 0 .5 µ Sec = 28.706 mSec

Increasing the delay

• The delay can be further increased by using register pairs for each of the loop counters

in the nested loops setup.

- It can also be increased by adding dummy instructions (like NOP) in the body of the loop.