

SOFTWARE DELAY ROUTINE IN 8051

In an 8051 microcontroller, it requires 12 cycles of the processor clock for executing a single instruction cycle. For an 8051 microcontroller clocked by a 12MHz crystal, the time taken for executing one instruction cycle is $1\mu\text{S}$ and it is according to the equation, *Time for 1 instruction cycle* = $12 / 12\text{MHz} = 1\mu\text{S}$. The shortest instructions will execute in $1\mu\text{S}$ and other instructions will take 2 or more micro seconds depending up on the size of the instruction. Thus a time delay of any magnitude can be generated by looping suitable instructions a required number of time. Any way, keep one thing in mind that software delay is not very accurate because we cannot exactly predict how much time its takes for executing a single instruction. Generally an instruction will be executed in the theoretical amount of time but some times it may advance or retard due to other reasons. Therefore it is better to use 8051 Timer for generating delay in time critical applications. How ever software delay routines are very easy to develop and well enough for less critical and simple applications.

Program to delay 1mS.

```
DELAY: MOV R6, #250D
        MOV R7, #250D
LABEL1: DJNZ R6, LABEL1
LABEL2: DJNZ R7, LABEL2
        RET
```

The above program roughly produces a delay of 1mS. The instruction `DJNZ Rx, LABEL` is a two cycle instruction and it will take $2\mu\text{S}$ to execute. So repeating this instruction 500 times will generate a delay of $500 \times 2\mu\text{S} = 1\text{mS}$. The program is written as a subroutine and it works this way. When called the sub routine `DELAY`, Registers R6 and R7 are loaded by 250D. Then `DJNZ R6, LABEL1` is executed until R6 becomes zero and then `DJNZ R7, LABEL2` is executed until R7 is zero. This creates a loop of `DJNZ Rx, LABEL` repeating 500 times and the result will be a 1mS delay. As I said earlier, this just a rough delay and when you test this program you may find slight differences in the output. You can make adjustments on the initial values of R6 and R7 to make the result more accurate.

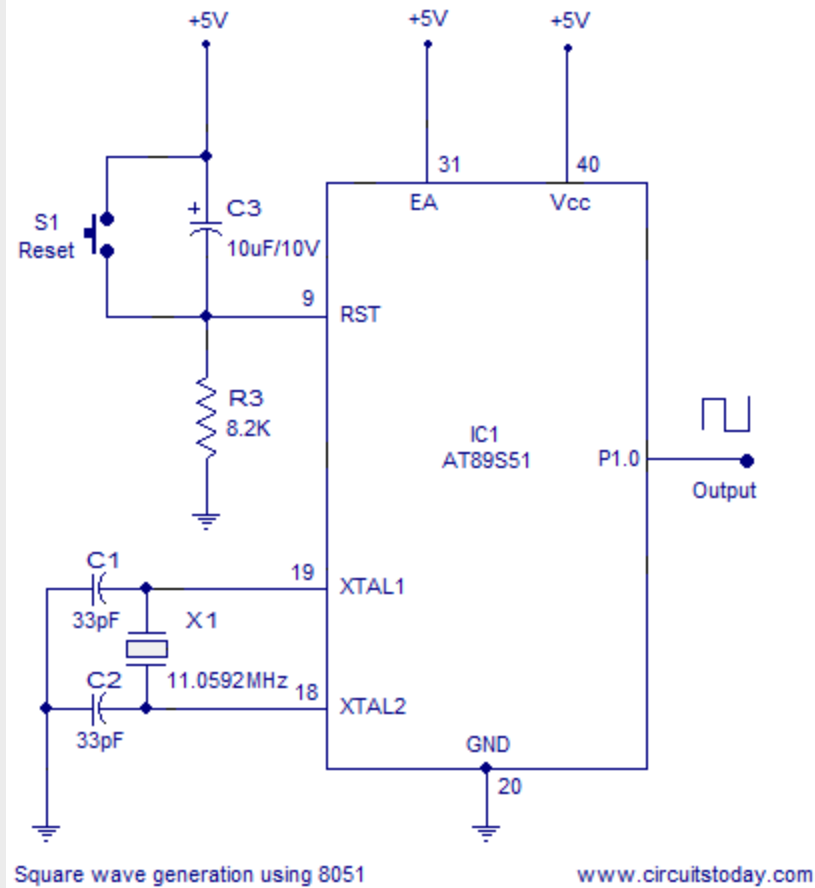
Program to delay 1 second.

The program shown below produces a delay of around 1 second. In this program subroutine for delaying 1mS (DELAY) is called 4 times back to back and the entire cycle is repeated 250 times. As result, a delay of $4 \times 1\text{mS} \times 250 = 1000\text{mS} = 1 \text{ second}$ is produced.

```
DELAY1: MOV R5, #250D
LABEL:  ACALL DELAY
        ACALL DELAY
        ACALL DELAY
        ACALL DELAY
        DJNZ R5, LABEL
        RET
DELAY:  MOV R6, #250D
        MOV R7, #250D
LOOP1:  DJNZ R6, LOOP1
LOOP2:  DJNZ R7, LOOP1
        RET
```

Square wave generation using 8051.

Using software delay subroutines square waves over a wide frequency range (limited by the crystal frequency) can be produced using 8051. The idea is very simple, run a subroutine with delay equal to half the time period of the square wave, complement any port pin after the delay routine is finished, repeat the delay subroutine again, complement the same port pin again and repeat the cycle again and again over time. This will result in a square wave of the required frequency at the corresponding port pin. Circuit diagram for generating square wave using 8051 is shown below. The same circuit can be used for generating any frequency but the program is different.



Square wave generation using 8051

Program for generating 1KHz square wave.

```

ORG 000H
MOV P1,#00000000B
MOV A,#00000000B
MAIN: MOV R6,#220D
LOOP1:DJNZ R6,LOOP1
      CPL A
      MOV P1,A
      SJMP MAIN
      END

```

Program for generating 2KHz square wave.

```

ORG 000H
MOV P1,#00000000B
MOV A,#00000000B

```

```
MAIN: MOV R6,#220D
      MOV R7,#183D
LOOP1:DJNZ R6,LOOP1
LOOP2:DJNZ R7,LOOP2
      CPL A
      MOV P1,A
      SJMP MAIN
      END
```

Program for generating 10KHz square wave.

```
ORG 000H
MOV P1,#00000000B
MOV A,#00000000B
MAIN: MOV R6,#20D
LOOP1:DJNZ R6,LOOP1
      CPL A
      MOV P1,A
      SJMP MAIN
      END
```

Source : <http://www.circuitstoday.com/software-delay-routine-in-8051>