BRAINSTORM-2005
State Level Hardware Design Contest
SDM College of Engineering & Technology

Hardware Problem:

Hello!! I am a Scientist at ISRO working for the Satellite Control laboratory. A part of my work deals with measuring frequencies of two signals (sum/difference frequencies). I need a device, which can take two signals as input and generate and show me the sum/difference frequencies.

This is my problem:

1. Design a circuit, which generates and displays the sum and difference of frequencies between two signals.
2. If f1 and f2 are the two input frequencies, with one frequency as reference, cases to be considered:
   i. $f_1 > f_2$
   ii. $f_1 = f_2$
   iii. $f_1 < f_2$
3. Use any means necessary.

I hope some of you young engineers will brainstorm to provide me a good design.

JSS Mahavidyapeetha

SRI JAYACHAMARAJENDRA COLLEGE OF ENGINEERING Mysore - 570 006, India

- Affiliated to University of Mysore and Visveswaraiah Technological University, Belgaum
- Approved by A.I.C.T.E. and Accredited by National Board of Accreditation
- Governed by the Grant-in-Aid Rules of Government of Karnataka

Telephone Office : 0821 - 2512568 / 2512292 / 2511383 / 2515944 / 2515751
Fax : 0821 - 2515770
e-mail : admn@sjce.ac.in

DEPARTMENT OF ELECTRONICS & COMMUNICATION

Date: 28th March 2004

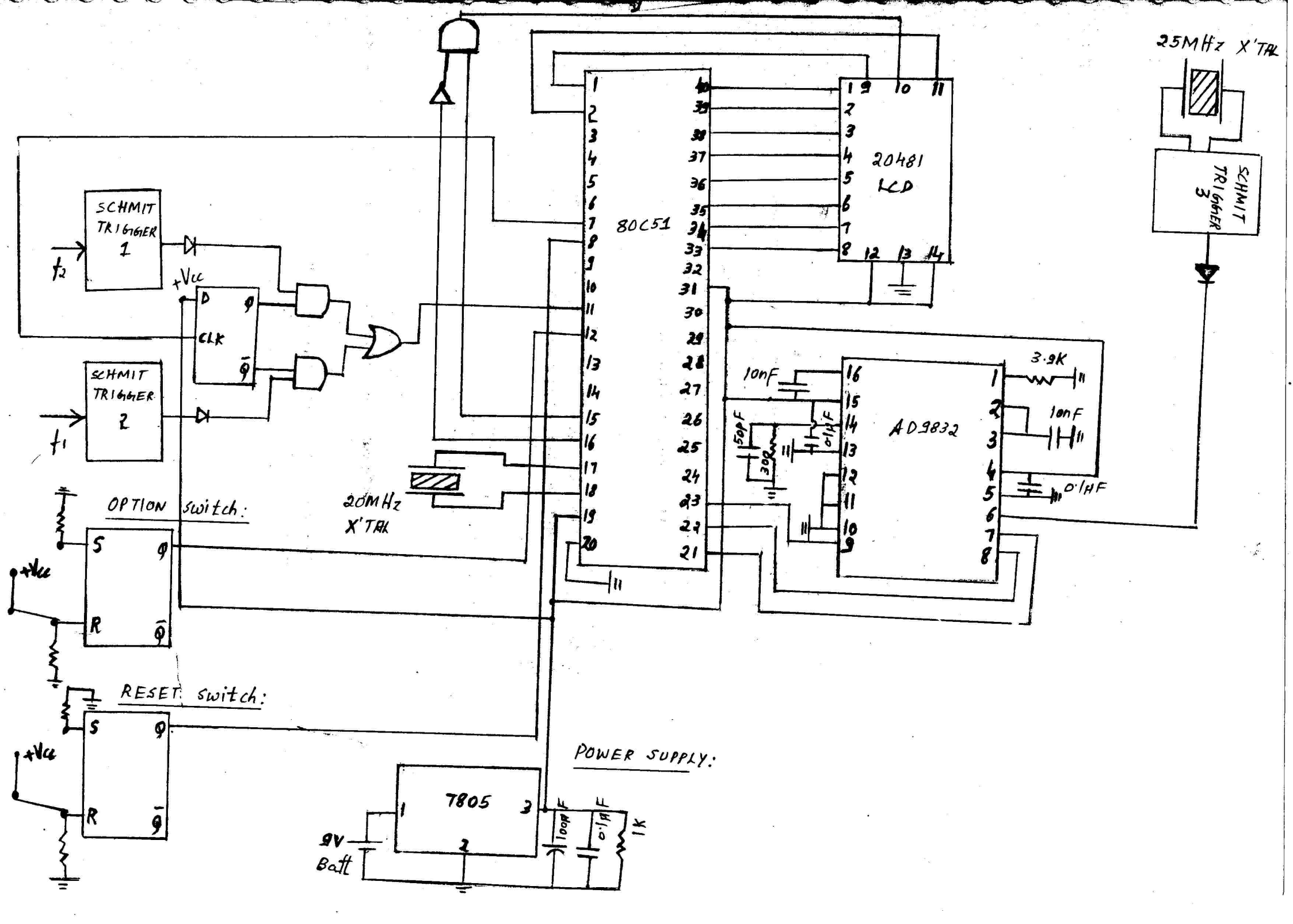
To:
The Organisers
BrainStorm 2005
SDM College of Engineering & Technology
Dharwad

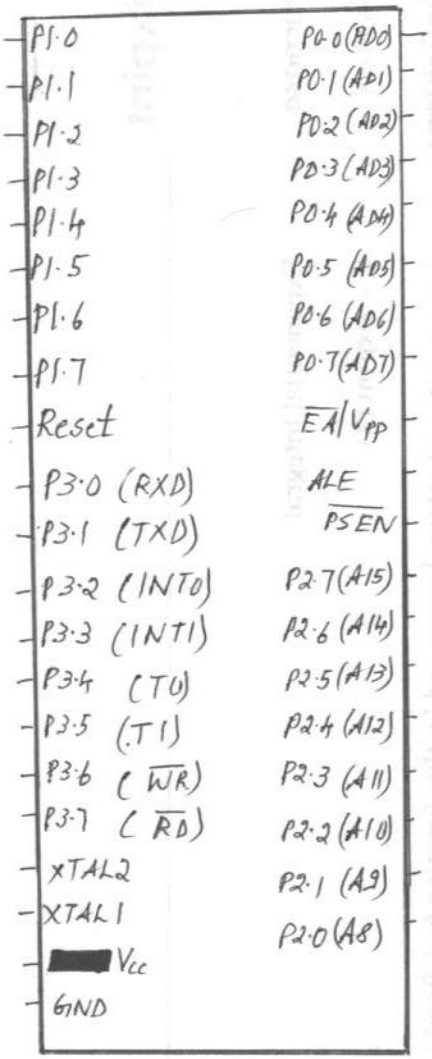
BONAFIDE CERTIFICATE

Certified that Amogh.R (Reg.No. 4JC02EC006) is a bonafide student of VI Semester in Electronics & Communication department proposing to present design solution for the hardware design problem to be held on 8th April 2005.

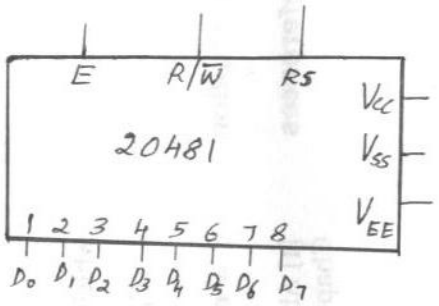
28/3/05

(Dr.R.D.Sudhaker Samuel)
Professor. & Head

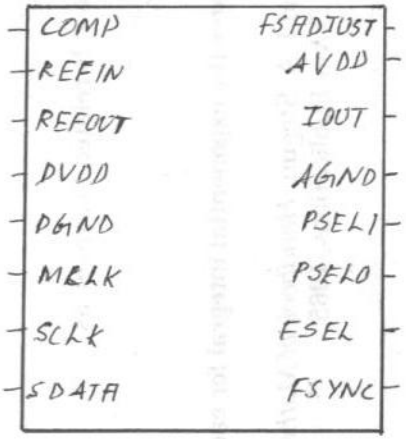

SCHMIT TRIGGER 1
SCHMIT TRIGGER 2
f_2
f_1
+Vcc
D
CLK
Q
$\bar{Q}$
OPTION switch:
RESET switch:
S
R
20MHz X'TAL
80C51
20481 LCD
AD9832
25MHz X'TAL
SCHMIT TRIGGER 3
10nF
50pF
0.1μF
39Ω
3.9K
POWER SUPPLY:
7805
9V Batt
100μF
1K



80C51 :

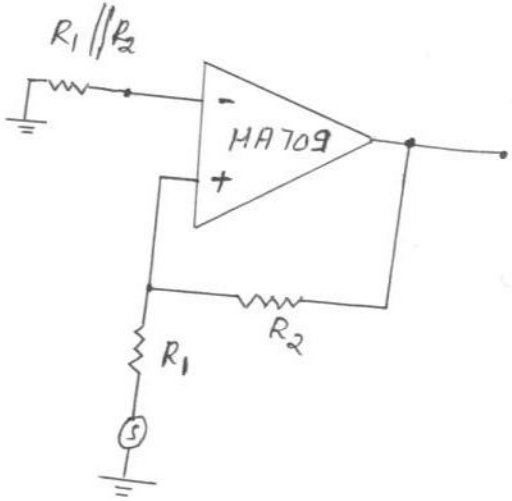


LCD:



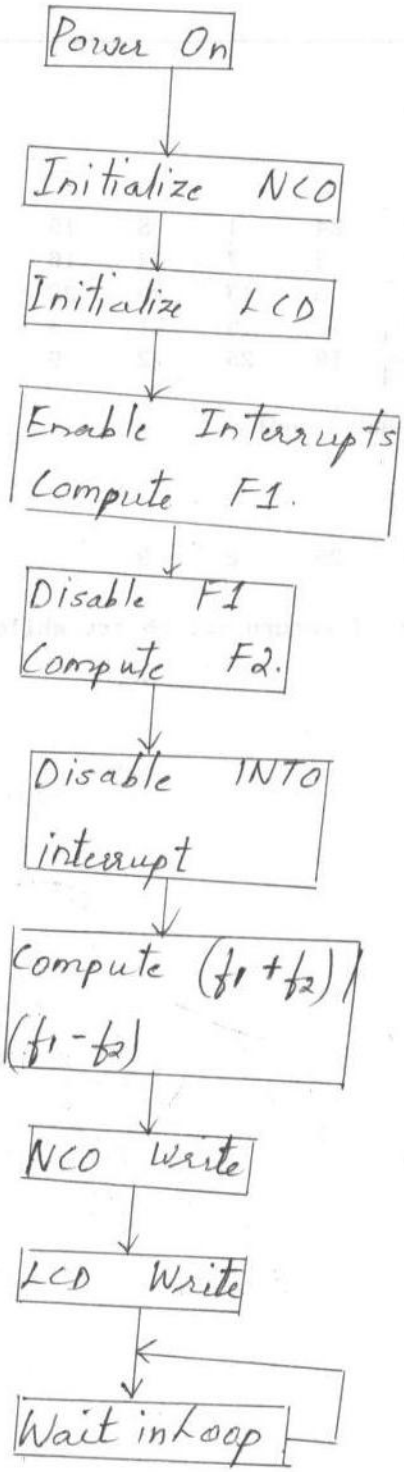
AD9832:-

SCHMIT TRIGGER:

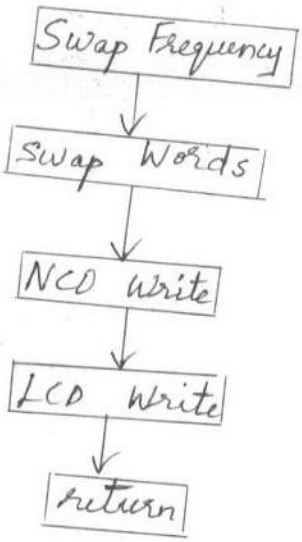


HA709: High Frequency Op-Amp.

Flowchart:



INT1 Interrupt:



COMPONENTS DESCRIPTION:

1. Schmit Trigger:
   It is used to convert input sinusoidal signals to square waves.

2. Diode:
   It is used to convert square waves to pulses.

3. D-FlipFlop, AND & OR gates :
   This circuit is used to select one of the input sinusoidal signals at a time.(Operation will be Explained later).

4. Option Switch :
   It is used to select either the sum frequency signal or the difference frequency signal.

5. Reset Switch:
   It is used to reset the entire device.

6. 7805 :
   It is a voltage regulator IC. It is used to provide +5V to 80C51, AD9832, LCD etc.

7. 80C51 :
   The Microcontroller is used to measure the two input signals frequencies i.e f1 and f2. Sum & Difference frequencies are also computed using the microcontroller. Then the sum Or difference frequency is written to AD9832, a Number Controlled Oscillator. Simultaneously, the frequency is also displayed at the LCD.

8. 20481 LCD:
   It is used to display the sum or difference frequency.

9. AD9832 NCO:
   This Number Controlled Oscillator is used to generate the sum or difference frequency Signal.

WORKING PRINCIPLE:

Based on the flowchart , the device working can be explained as follows

Step1: Power ON

Step2: Initialize NCO by some appropriate instructions.

Step3: Initialize LCD by some appropriate instructions.

Step4: Enable INT0 interrupt in 80C51 . Measure the frequency F1 in a 1sec delay subroutine. In the 1sec delay subroutine , the signal F1 is made to interrupt the 80C51. The number of Interrupts in 1sec is equal to the frequency of the signal. Upon the Power ON , output of the D-FlipFlop is 0,therefore the output of the OR gate is the signal of frequency F1.

Step5: Now since the frequency F2 is also to be measured, after the 1sec delay subroutine the output of the D-FlipFlop is to be made 1, so that the signal F2 is selected. This is made by outputting a pulse to D-FlipFlop by some instructions.

Step6: By the same 1 sec delay subroutine, measure the frequency of the signal F2.

Step7: Disable the interrupt INT0 of the 80C51 after the two frequencies are measured.

Step8: Compute the sum & difference frequencies.

Step9: Using a subroutine , write to the NCO any one of the frequencies.

Step10: Using a subroutine , write to the LCD any one of the frequencies.

Step11: After all these steps, it will wait in a loop to be interrupted .

PROGRAM DETAILS:

Main Program:

```
mov R0,#80h
mov @R0,#F8h
inc R0
mov @R0,#0
mov DPTR, #80h
acall Ncowrite
mov R0,#80h
mov @R0,#90h
inc R0
mov @R0,#0
acall Ncowrite
mov R0,#80h
mov @R0,#C0h
inc R0
mov R0 ,#0
acall Ncowrite
mov R0,#F3h
acall 20delay
mov p1,#2
mov p0,#30h
mov p1,#0
mov R0,#D2h
acall 20delay
mov p1,#2
mov p0.#30h
mov p1,#0
mov R0,#7
acall 20delay
mov p1,#2
mov p0,#30h
mov p1,#0
mov R0,#F0h
acall 20delay
mov a,#38h
acall Lcdread
mov a,#8
acall Lcdread
mov a,#1
acall Lcdread
mov a,#7
acall Lcdread
mov a,#Ch
acall Lcdread

mov IE,#85h
mov TCON,#05h
mov R0,#80h
```

```
mov @R0,#0
inc R0
mov @R0,#0
inc R0
mov @R0,#0
dec R0
dec R0
acall 1secdelay
mov DPTR,#90h
mov R1,#3
repe: mov a,@R0
mov @DPTR,a
mov @R0,#0
inc R0
inc DPTR
djnz R1 repe
mov R0,#80h
mov p1,#0
mov p1,#80h
nop
mov p1,#0
acall 1secdelay
mov IE,84h
mov R2,#3
mov R1,#A0h
mov DPTR,#90h
clr c
rep: mov a,@DPTR
addc a,@R0
mov @R1,a
inc R1
inc R0
inc DPTR
djnz R2 rep
mov R2,#3
mov R1,#B0h
mov DPTR,#90h
mov R0,#80h
clr c
rep2:mov a, @DPTR
subb a, @R0
mov @R1,a
inc R1
inc R0
inc DPTR
djnz R2 rep2
 jnc  Noproblem
mov R2,#2
mov R1# b0h
mov a,@R1
cpl a
```

```
add a,#1
mov @R1,a
rep3:inc R1
mov a,@R1
cpl  a
addc a,#0
mov @R1,a
djnz R2 rep3
Noproblem: mov R0,#a0h
mov R1,#c0h
acall multiplier
mov R0,#B0h
acall multiplier
mov R0,#81h
mov @R0,#0
dec R0
mov @R0,#60h
mov DPTR,#80h
acall Ncowrite
 acall NcoFreqwrite
acall  LCDwrite
wait :  sjmp wait
```

SUBROUTINES:

1. Ncowrite : To write to the NCO.

```
Push b
Push dpl
Push dph
Mov  R0,#2
Start1: mov b,#1
Start: mov a, @DPTR
Mov  R1,b
Repe: rl a
Djnz  R1 repe
Anl a,#1
Orl  a,#2
Mov p2,a
Xrl  a,#2
Mov p2,a
Cjne a,#9 start
Inc DPTR
Djnz R0 start1
Mov a,#4
Mov p2 ,a
Pop dph
Pop dpl
Pop b
Ret
```

2.20 delay: This subroutine gives a delay of 20 micro secs.

```
Push acc
Start: mov A,#10
Repe: djnz a repe
Djnz R0 A Start
Pop acc
Ret
```

3. Lcdread : to read the status of the LCD.

```
  mov  p1,#2
  mov  p2,a
  loop: mov a,p2
  anl a,#80h
  jnz loop
 mov p1,#0
 ret
```

4. Multiplier: to multiply the frequency with offset.

```
Mov R2,#0
Mov B, #Ach
Mov DPTR,# 93h
Mov a,@R0
Mul ab
Mov @R1,a
Inc R1
Inc R0
Repe:  mov @DPTR, b
Inc DPTR
Mov B, #ach
Mov  A,@R0
Mul ab
Mov @DPTR, a
Inc DPTR
Djnz R2 repe
Clr  c
Mov DPTR,#93h
Mov  R2,#2
Repe2:mov a, @DPTR
Inc DPTR
Addc a, @DPTR
Mov 2R1,a
Inc DPTR
Inc R1
Djnz R2 repe2
Mov a,#0
Addc a, @DPTR
Mov a, @DPTR
```

```
Inc R1
Ret
```

5.NcoFreqwrite: to write the frequency to the NCO.

```
Mov R1,#c0h
Mov B,#20h
Mov R2, #4
Rep: mov @DPTR, b
Inc DPTR
Mov a, @R1
Mov @DPTR, a
Inc DPTR
Inc R1
Inc B
Djnz R2 rep
Mov R0,#91h
Mov @R0,#0
Dec R0
Mov @R0,#30h
Mov DPTR,#90h
acall  Ncowrite
mov DPTR,#80h
acall Ncowrite
inc DPTR
inc DPTR
acall Ncowrite
mov DPTR,#90h
acall Ncowrite
mov DPTR,#84h
acall Ncowrite
inc DPTR
inc DPTR
acall  Ncowrite
ret
```

6.1sec delay : to introduce 1sec delay.

```
Push acc
Mov a,#ffh
Repe: djnz a repe
Pop acc
Ret
```

7. LCDwrite: to write to the LCD.

```
Mov R0,#A2h
```

```
Mov R2,#3
Mov DPTR,#80h
Rep: mov a,@R0
acall ascii
dec R0
djnz R0 rep
mov R2,#6
mov DPTR,#80h
mov  p1,#2
mov p0,#98h
mov p1,#3
rep1:mov a, @DPTR
mov p0,a
inc DPTR
djnz R2 rep1
ret
```

8.Ascii: to convert Binary numbers to ascii codes.

```
Mov R3,#2
Mov b, a
Anl a,#f0h
Rr a
Rr a
Rr a
Rr  a
Sp: add a,#30h
Sub a,#3ah
Jc loop
Add a,#41h
Sjmp next
Loop: add a ,#3ah
Next: mov  @DPTR, a
Anl b,#0fh
Mov  a, b
Inc DPTR
Djnz R3 sp
ret
```

INTERRUPT SERVICE ROUTINES:

1. INT0 ISR: To measure the frequency of the interrupting signal.

```
Push  acc
Mov a,@R0
Add a,#1
Mov @R0,a
Jnc last
```

```
Inc R0
Mov A,@R0
Add a,#1
Mov @R0,a
Jnc last1
Inc R0
Mov A,@R0
Add a,#1
Mov @R0,a
Dec R0
Last1:dec R0
Last :pop acc
Reti
```

2. INT1 ISR: to swap between F1 and F2.

```
Mov R0,#a0h
Mov DPTR,#b0h
Mov R2,#3
Rep: mov a, @DPTR
Mov B,@R0
Mov @R0,a
Mov @DPTR,B
Inc R0
Inc DPTR
Djnz R2 rep
Mov R0,#c0h
Mov DPTR,#c4h
Mov R2,#4
Rep1: mov a, @DPTR
Mov b,@R0
Mov @R0,a
Mov @DPTR, b
Inc R0
Inc DPTR
Djnz R2 rep1
Acall NcoFreqwrite
Acall LCDwrite
Reti
```