## 4.4 Low Pass Filter (LPF)

Is a filter that passes signals with a frequency lower than a certain cut off frequency and attenuates signals with frequencies higher than the cutoff frequency, as shown in figure (4.6) . A low-pass filter is the opposite of a high-pass filter.

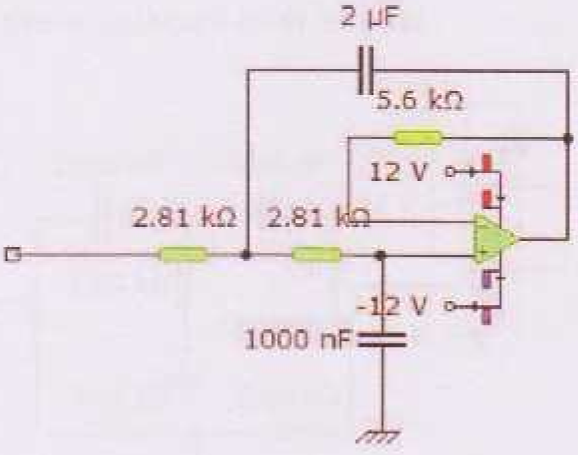

**Figure 4.4 :** Low Pass Filter

fc = 40 Hz

C2 = 2C1

Let C1 = 1 µF

C2 = 2 µF

$$fc = \frac{1}{2\pi R1\sqrt{C1C2}}$$

$$R1 = \frac{1}{2\pi fc\sqrt{C1C2}}$$

R1=2.813 KΩ

R2= R1 = 2.813 KΩ

R3= 5.6 KΩ

### 4.5 Band Reject Filter (BRF)

BRF filter passes all frequencies above and below a particular range set by the component values, as shown in figure (4.7) . Not surprisingly, it can be made out of a low-pass and a high-pass filter, just like the band-pass design, Chops out a specific frequency. This would be used to filter out the mains hum from the 60 Hz power line, for countries where power transmission is at 50Hz, the filter would have a 49–51 Hz range.

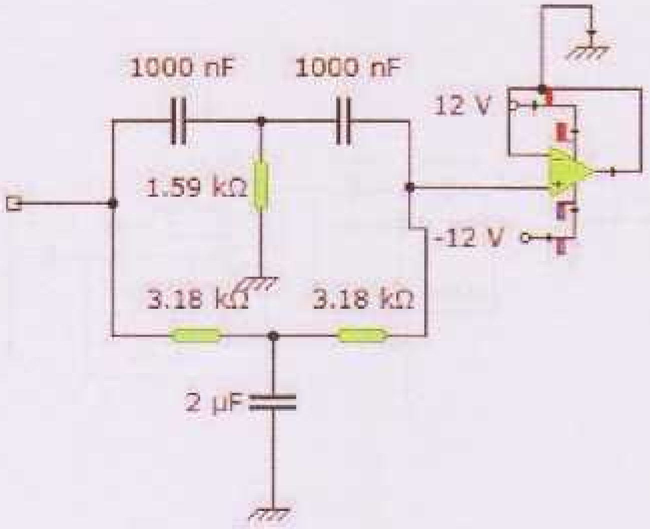

**Figure 4.5:** Band Reject Filter

let fc = 50 Hz

G = 1

$fc = \frac{1}{2\pi RC}$

$R = \frac{1}{50(2\pi C)} = 3.18\ k\Omega$

$C_1 = C_2 = C = 1\ \mu F$

$R_1 = R_2 = R = 3.18\ k\Omega$

$R_3 = \frac{1}{2} R_1 = 1.59 \text{ k}\Omega$

$C_3 = 2\ C_1 = 2\ \mu\text{F}$

### 4.6 Circuit Design

The all circuits that is used in this project are conected in this completed circuit as shown in figure (4.6) .

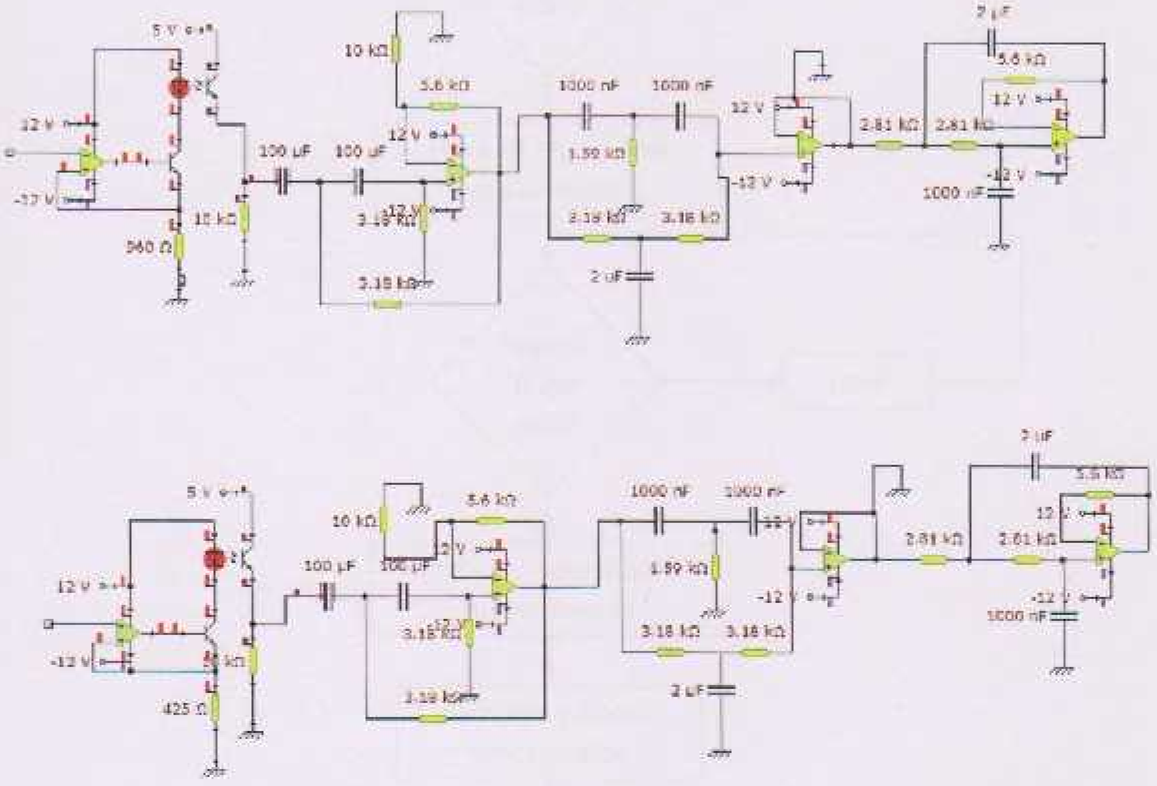

**Figure 4.6 :** Complete System Schematic

## 4.7 System Flow chart

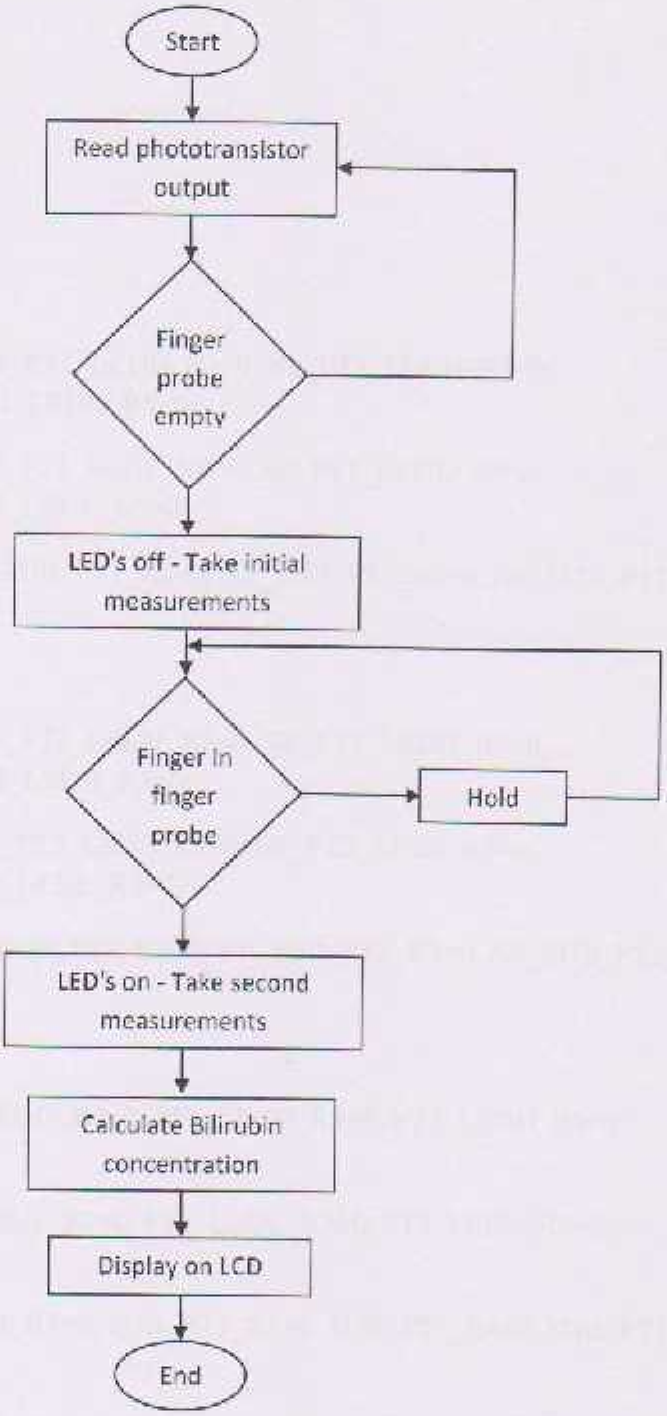
Start
Read phototransistor output
Finger probe empty
LED's off - Take initial measurements
Finger in finger probe
Hold
LED's on - Take second measurements
Calculate Bilirubin concentration
Display on LCD
End

**Figure 4.7 :** System Flow Chart

### 4.8 System Software

```
#include <LiquidCrystal.h>

int LED455 = 8;

int LED575 = 9;

int PhoTrans1 = A0;

int PhoTrans2 = A1;

float NF_PT1_LEDH_R1=0, NF_PT1_LEDH_R2=0, NF_PT1_LEDH_R3=0,
NF_PT1_LEDH_R4=0, NF_PT1_LEDH_R5=0;

float NF_PT1_LEDL_R1=0, NF_PT1_LEDL_R2=0, NF_PT1_LEDL_R3=0,
NF_PT1_LEDL_R4=0, NF_PT1_LEDL_R5=0;

float NF_SUB_PT1_R1=0, NF_SUB_PT1_R2=0, NF_SUB_PT1_R3=0, NF_SUB_PT1_R4=0,
NF_SUB_PT1_R5=0;

float NF_AVG_PT1=0;

float NF_PT2_LEDH_R1=0, NF_PT2_LEDH_R2=0, NF_PT2_LEDH_R3=0,
NF_PT2_LEDH_R4=0, NF_PT2_LEDH_R5=0;

float NF_PT2_LEDL_R1=0, NF_PT2_LEDL_R2=0, NF_PT2_LEDL_R3=0,
NF_PT2_LEDL_R4=0, NF_PT2_LEDL_R5=0;

float NF_SUB_PT2_R1=0, NF_SUB_PT2_R2=0, NF_SUB_PT2_R3=0, NF_SUB_PT2_R4=0,
NF_SUB_PT2_R5=0;

float NF_AVG_PT2=0;

float PT1_LEDH_R1=0, PT1_LEDH_R2=0, PT1_LEDH_R3=0, PT1_LEDH_R4=0,
PT1_LEDH_R5=0;

float PT1_LEDL_R1=0, PT1_LEDL_R2=0, PT1_LEDL_R3=0, PT1_LEDL_R4=0,
PT1_LEDL_R5=0;

float SUB_PT1_R1=0, SUB_PT1_R2=0, SUB_PT1_R3=0, SUB_PT1_R4=0, SUB_PT1_R5=0;

float AVG_PT1=0;

float PT2_LEDH_R1=0, PT2_LEDH_R2=0, PT2_LEDH_R3=0, PT2_LEDH_R4=0,
PT2_LEDH_R5=0;
```

```
digitalWrite(LED455,LOW);
 NF_PT1_LEDL_R2=analogRead(PhoTrans1);
delay(500);
 NF_SUB_PT1_R2= NF_PT1_LEDH_R2-NF_PT1_LEDL_R2;
digitalWrite(LED455,HIGH);
 NF_PT1_LEDH_R3=analogRead(PhoTrans1);
delay(500);
digitalWrite(LED455,LOW);
 NF_PT1_LEDL_R3=analogRead(PhoTrans1);
delay(500);
 NF_SUB_PT1_R3= NF_PT1_LEDH_R3-NF_PT1_LEDL_R3;
digitalWrite(LED455,HIGH);
 NF_PT1_LEDH_R4=analogRead(PhoTrans1);
delay(500);
digitalWrite(LED455,LOW);
 NF_PT1_LEDL_R4=analogRead(PhoTrans1);
delay(500);
 NF_SUB_PT1_R4= NF_PT1_LEDH_R4-NF_PT1_LEDL_R4;
digitalWrite(LED455,HIGH);
 NF_PT1_LEDH_R5=analogRead(PhoTrans1);
delay(500);
digitalWrite(LED455,LOW);
 NF_PT1_LEDL_R5=analogRead(PhoTrans1);
delay(500);
 NF_SUB_PT1_R5=NF_PT1_LEDH_R5-NF_PT1_LEDL_R5;
```

```
  NF_PT2_LEDH_R4=analogRead(PhoTrans2);
  delay(500);
  digitalWrite(LED575,LOW);
  NF_PT2_LEDL_R4=analogRead(PhoTrans2);
  delay(500);
  NF_SUB_PT2_R4= NF_PT2_LEDH_R4-NF_PT2_LEDL_R4;
  digitalWrite(LED575,HIGH);
  NF_PT2_LEDH_R5=analogRead(PhoTrans2);
  delay(500);
  digitalWrite(LED575,LOW);
  NF_PT2_LEDL_R5=analogRead(PhoTrans2);
  delay(500);
  NF_SUB_PT2_R5=NF_PT1_LEDH_R5-NF_PT2_LEDL_R5;
NF_AVG_PT2=(NF_SUB_PT2_R1+NF_SUB_PT2_R2+NF_SUB_PT2_R3+NF_SUB_PT2_R4
+NF_SUB_PT2_R5)/5;
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Please Put");
  lcd.setCursor(0,1);
  lcd.print("your Finger");
  lcd.setCursor(0,2);
  lcd.print("in the Chamber");
  delay(5000);
  lcd.clear();
  delay(1000);
```

```
PT2_LEDH_R2=analogRead(PhoTrans2);
delay(500);
digitalWrite(LED575,LOW);
PT2_LEDL_R2=analogRead(PhoTrans2);
delay(500);
SUB_PT2_R2= PT2_LEDH_R2-PT2_LEDL_R2;
digitalWrite(LED575,HIGH);
PT2_LEDH_R3=analogRead(PhoTrans2);
delay(500);
digitalWrite(LED575,LOW);
PT2_LEDL_R3=analogRead(PhoTrans2);
delay(500);
SUB_PT2_R3= PT2_LEDH_R3-PT2_LEDL_R3;
digitalWrite(LED575,HIGH);
PT2_LEDH_R4=analogRead(PhoTrans2);
delay(500);
digitalWrite(LED575,LOW);
PT2_LEDL_R4=analogRead(PhoTrans2);
delay(500);
SUB_PT2_R4= PT2_LEDH_R4-PT2_LEDL_R4;
digitalWrite(LED575,HIGH);
PT2_LEDH_R5=analogRead(PhoTrans2);
delay(500);
digitalWrite(LED575,LOW);
PT2_LEDL_R5=analogRead(PhoTrans2);
```

```
 delay(500);
 SUB_PT2_R5=PT1_LEDH_R5-PT2_LEDL_R5;
 AVG_PT2=(SUB_PT2_R1+SUB_PT2_R2+SUB_PT2_R3+SUB_PT2_R4+SUB_PT2_R5)/5;
   Absorb455= log(NF_AVG_PT1/AVG_PT1);
   Absorb575= log(NF_AVG_PT2/AVG_PT2);
   BAbsorb= Absorb455-Absorb575;
   Billirubin_Concentration=BAbsorb/E;
 lcd.clear();
 lcd.setCursor(0,0);
 lcd.print("Your Bilirubin");
 lcd.setCursor(0,1);
 lcd.print("Concentration=");
 lcd.setCursor(0,2);
 lcd.print(Billirubin_Concentration);
 delay (10000);


}
```