AppendixA

A1:lm35 Features

1-Calibrated directly in 1° Celsius (Centigrade)

2- Linear + 10.0 mV/°C scale factor

3-0.5°C accuracy guarantee able (at +25°C)

4-Rated for full range −55° to +150°C

5-Suitable for remote applications

6-Low cost due to wafer-level trimming

7-Operates from 4 to 30 volts

8-Less than 60 μA current drain

9-Low self-heating, 0.08°C in still air

10-Nonlinearity only ±1⁄4°C typical

11- Low impedance output, 0.1 Ω for 1 mA load

12-low cost

13 -200°C over range

A2:Pressure sensor Features

1-1.5% Maximum Error over 0ºCto 85ºC

2-Ideally suited for Microprocessor or Microcontroller–Based Systems

3-Patented Silicon Shear Stress Strain Gauge

4-Available in Absolute, Differential and Gauge Configurations

5-Durable Epoxy Unibody Element

6-Easy–to–Use Chip Carrier Option

A3:Humidity sensor Features HS1100/ HS 1101

Full interchangeabilitywith no calibration required in standard conditions

1-Instantaneous desaturation after long periods in saturation phase

2- Compatible with automatized assembly processes, including wave soldering**,**

3-Reflow and water immersion .

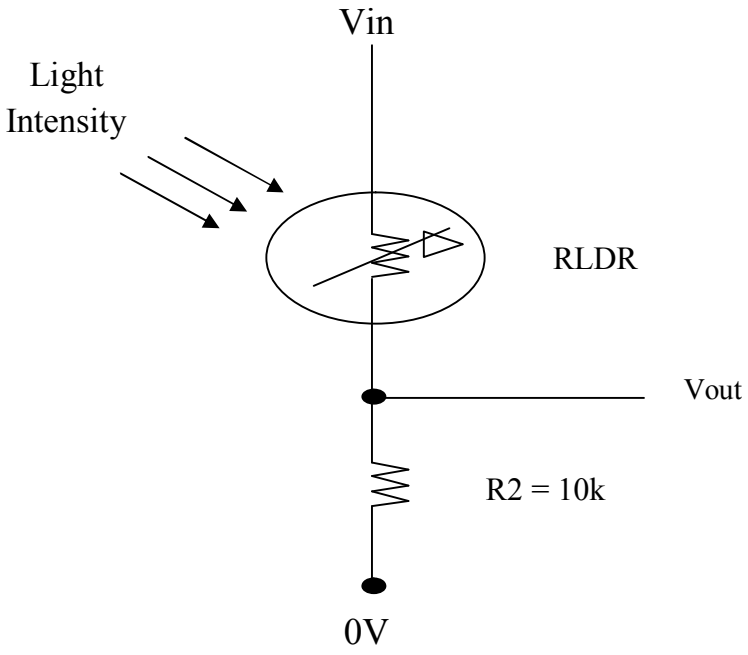
4- High reliability and long term stability.

5- Suitable for linear voltage or frequency output circuitry.

6- Fast response time.

7- Individual marking for compliance to stringent traceability requirements.

A4:Light Sensor Circuit and Equation



$$V_{out} = V_{in} \times \frac{R_2}{RLDR + R_2} \qquad (A1)$$

Appendix B

Table B.1:Pin description of LCD

Pin No	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	Vcc
3	Contrast adjustment; through a variable resistor	VEE
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable
7	data pin	DB0
8	data pin	DB1
9	data pin	DB2
10	data pin	DB3
11	data pin	DB4
12	data pin	DB5
13	data pin	DB6
14	data pin	DB7
15	Backlight VCC (5V)	Led+
16	Backlight Ground (0V)	Led-

Appendix C

Table C.1: Pin description of HT12Ddecoder

Pin number	Function	Name
1	8 Bit Address PINS input	A0
2		A1
3		A2
4		A3
5		A4
6		A5
7		A6
8		A7
9	Ground (0V)	Ground
10	4 Bit Data Address PINS for output	D0
11		D1
12		D2
13		D3
14	Serial data input	Input
15	Oscillator input	OSC2
16	Oscillator output	OSC1
17	Valid transmission,active high	VT
18	Supply voltage (5V) (2.4 – 12 v)	Vcc

Appendix D

Table D.1: Pin description of HT12Dencoder

Pin number	Function	name
1	8 Bit Address PINS input	A0
2		A1
3		A2
4		A3
5		A4
6		A5
7		A6
8		A7
9	Ground (0V)	Ground
10	4 Bit Data Address PINS for output	D0
11		D1
12		D2
13		D3
14	Transmission Enable (Active low)	TE
15	Oscillator output	OSC2
16	Oscillator Input	OSC1
17	Valid transmission, active high	VT
18	Supply voltage (5V) (2.4 – 12 v)	Vcc

Appendix E

Table E.1:Transmitter module pin description

Pin number	Function	Name
1	Ground (0V)	GND
2	Serial data input pin	Data
3	Supply voltage (5V)	Vcc
4	Antenna output pin	ANT

AppendixF

Table F.1:Receiver module pin description

Pin number	Function	Name
1	Ground (0V)	GND
2	Serial data output pin	Data
3	Linear output pin not connected	NC
4	Supply voltage (5V)	Vcc
5	Supply voltage (5V)	Vcc
6	Ground (0V)	GND
7	Ground (0V)	GND
8	Antenna input pin	ANT

AppendixG

Table G.1:MAX232 pin describtion

Name	Function	Pin No
1	Capacitor connection pins	Capacitor 1+
2		Capacitor 2+
3		Capacitor 3+
4		Capacitor 4+
5		Capacitor 5+
6		Capacitor 6+
7	Output pin; outputs the serially transmitted data at RS232 logic level; connected to receiver pin of PC serial port	T2 Out
8	Input pin; receives serially transmitted data at RS 232 logic level; connected to transmitter pin of PC serial port	R2 In
9	Output pin; outputs the serially transmitted data at TTL logic level; connected to receiver pin of controller	R2 Out
10	Input pins; receive the serial data at TTL logic level; connected to serial transmitter pin of controller	T2 In
11		T1 In
12	Output pin; outputs the serially transmitted data at TTL logic level; connected to receiver pin of controller	R1 Out
13	Input pin; receives serially transmitted data at RS 232 logic level; connected to transmitter pin of PC serial port	R1 In
14	Output pin; outputs the serially transmitted data at RS232 logic level; connected to receiver pin of PC serial port	T1 Out
15	Ground (0V)	Ground
16	Supply voltage; 5V (4.5V – 5.5V)	Vcc

AppendixH

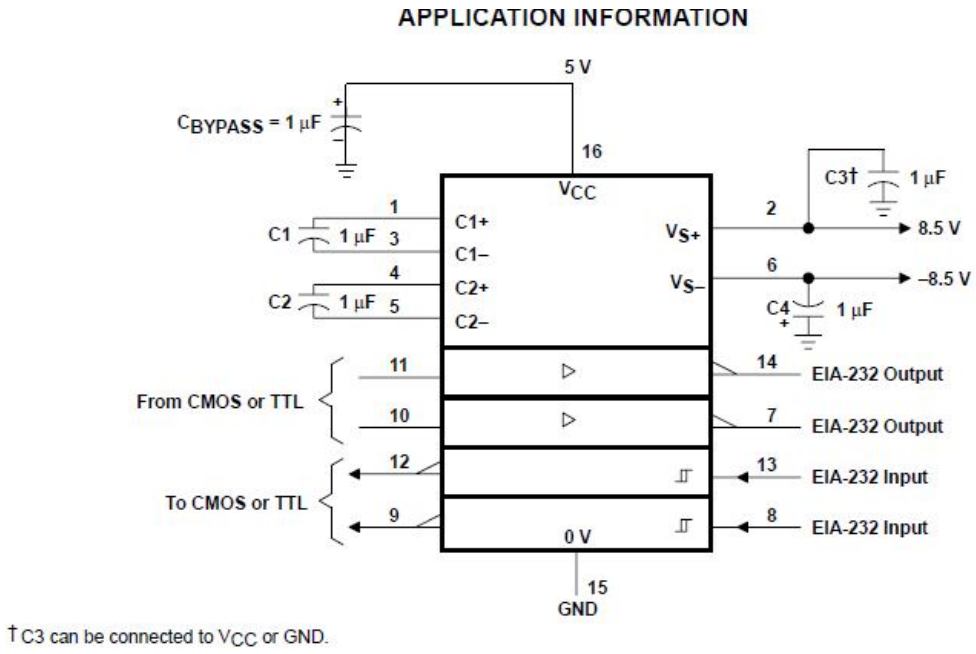


Figure H.1:Typicaloperating circuit of MAX232

Appendix I

Table I.1:The pindescribtion of RS232

PIN	Purpose	SIGNAL NAME
1	Data Carrier Detect	DCD
2	Received Data	RxData
3	Transmitted Data	TxData
4	Data Terminal Ready	DTR
5	Signal Ground	Gnd
6	Data Set Ready	DSR
7	Request To Send	RTS
8	Clear To Send	CTS
9	Ring Indicator	RI

Appendix J

Table J.1: Pin Description of NE555 Timer

Pin	Name	Purpose
1	GND	Ground reference voltage, low level (0 V)
2	TRIG	The OUT pin goes high and a timing interval starts when this input falls below 1/2 of CTRL voltage (hence TRIG is typically 1/3 VCC, CTRL being 2/3 VCC by default, if CTRL is left open).
3	OUT	This output is driven to approximately 1.7 V below +VCC or GND.
4	RESET	A timing interval may be reset by driving this input to GND, but the timing does not begin again until RESET rises above approximately 0.7 volts. Overrides TRIG which overrides THR.
5	CTRL	Provides "control" access to the internal voltage divider (by default, 2/3 VCC)
6	THR	The timing (OUT high) interval ends when the voltage at THR is greater than that at CTRL (2/3 VCC if CTRL is open).
7	DIS	Open collector output which may discharge a capacitor between intervals. In phase with output.
8	VCC	Positive supply voltage, which is usually between 3 and 15 V depending on the variation.

AppendixK

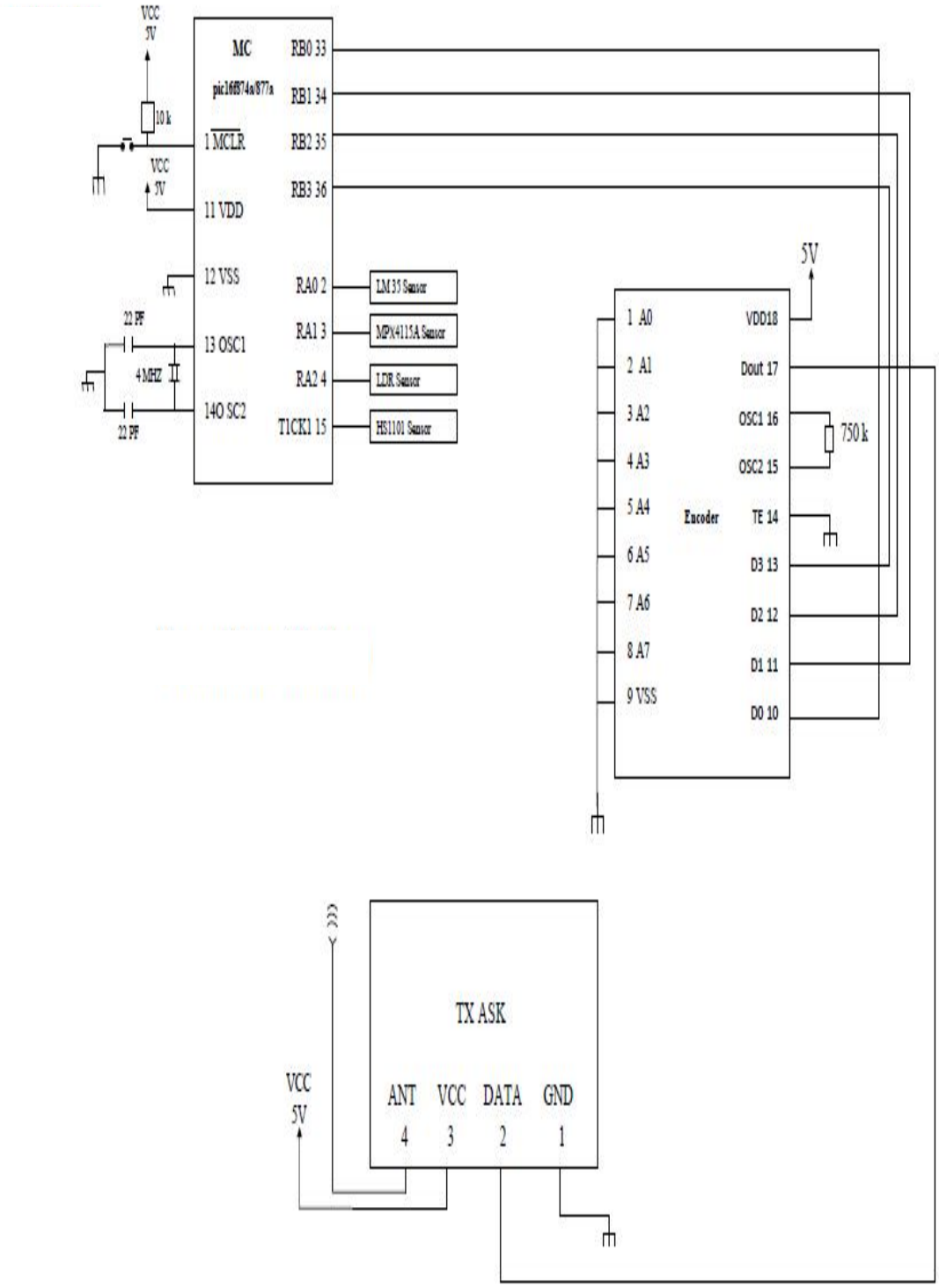


Figure K1: Transmission Station

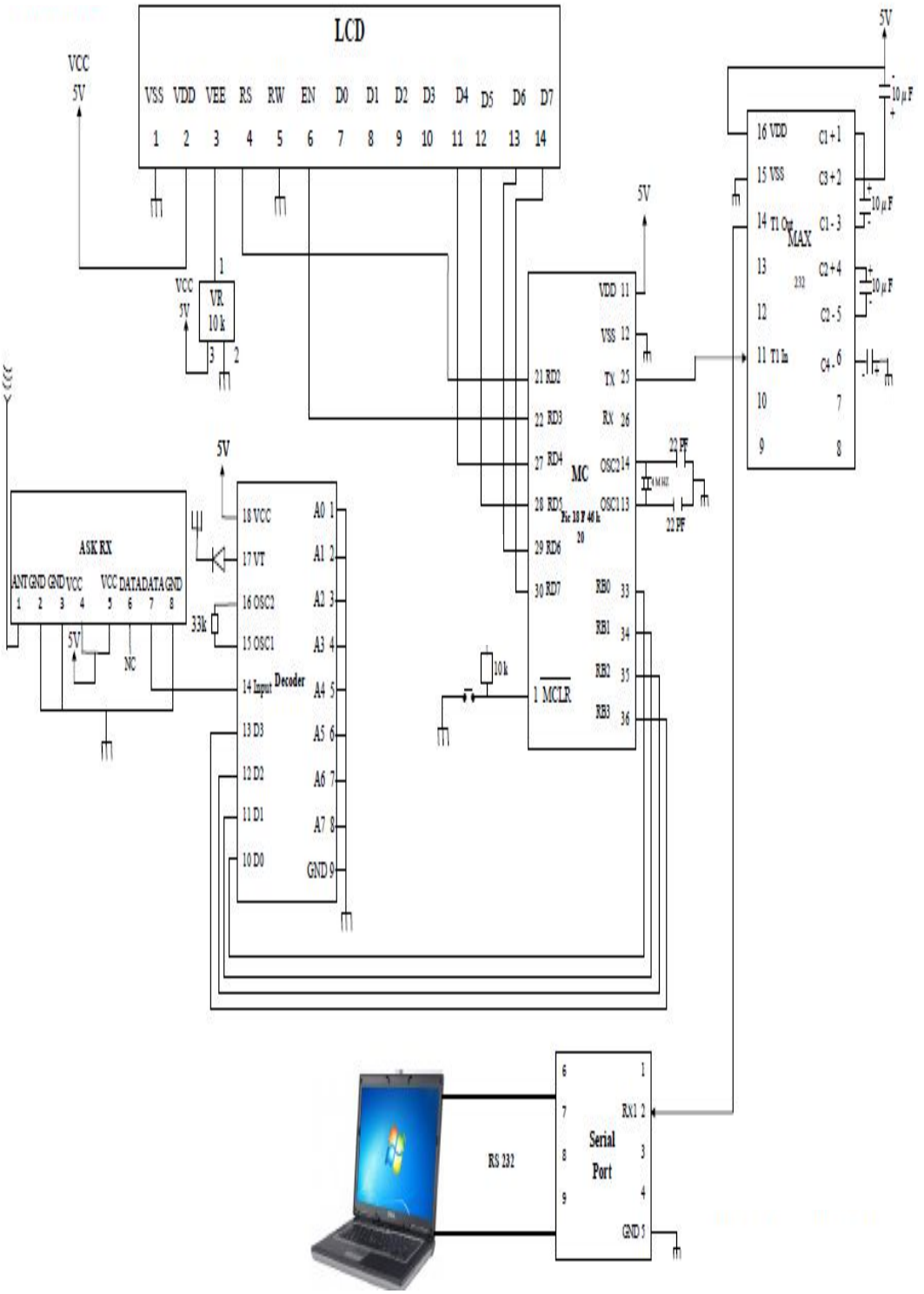


Figure K2: Receiver station

AppendixL

L1: Transmitter Code

```
unsignedint Temp;   // Tempreature Sensor variables
unsignedint pressure ;// Pressure Sensor variables
unsignedint light ; // Pressure Sensor variables
unsigned long T1;// save frequency
unsignedint Humidity;
bit S0,S1,S2,S3,S4,S5,S6,S7,S8,S9,S10,S11,S12,S13,S14,S15;
void send(){
if(Temp<35){
  S1=1;
  PORTB.F0=0; PORTB.F1=0; PORTB.F2=0; PORTB.F3=0;  }   //0000
else S1=0;
if(Temp<40 && S1!=1 ){
  S2=1;
  PORTB.F0=1; PORTB.F1=0; PORTB.F2=0; PORTB.F3=0; }  //1000
else S2=0;

if(Temp<45 && S1!=1 && S2!=1 ){
  S3=1;
  PORTB.F0=0; PORTB.F1=1; PORTB.F2=0; PORTB.F3=0;}  //0100
else S3=0;
if(Temp<50 && S1!=1 && S2!=1 && S3!=1 ){
  S4=1;
   PORTB.F0=1; PORTB.F1=1; PORTB.F2=0; PORTB.F3=0; }  //1100
else S4=0;
Delay_ms(500);
  ////////////////////Humidity Sensor//////////////////////////
if(Humidity<80){
  S5=1;
  PORTB.F0=0; PORTB.F1=0; PORTB.F2=1; PORTB.F3=0;  }  //0010
else {
  S5=0;   }
if(Humidity<90 && S5!=1 ){
  S6=1;
  PORTB.F0=1; PORTB.F1=0; PORTB.F2=1; PORTB.F3=0; }  //1010
else S6=0;
if(Humidity<100 && S5!=1 && S6!=1 ){
  S7=1;
  PORTB.F0=0; PORTB.F1=1; PORTB.F2=1; PORTB.F3=0;}  //0110
else S7=0;
```

```
if(Humidity<120 && S5!=1 && S6!=1 && S7!=1){
  S0=1;
  PORTB.F0=1; PORTB.F1=1; PORTB.F2=1; PORTB.F3=0;}  //1110
else S0=0;
Delay_ms(500);
    ////////////////////Pressure Sensor ////////////////////
if(pressure<50){
  S8=1;
  PORTB.F0=0; PORTB.F1=0; PORTB.F2=0; PORTB.F3=1;  }  //0001
else {
  S8=0;   }
if(pressure<60 && S8!=1 ){
  S9=1;
  PORTB.F0=1; PORTB.F1=0; PORTB.F2=0; PORTB.F3=1;}  //1001
else S9=0;
if(pressure<70 && S8!=1 && S9!=1 ){
  S10=1;
  PORTB.F0=0; PORTB.F1=1; PORTB.F2=0; PORTB.F3=1;}  //0101
else S10=0;
if(pressure<100 && S8!=1 && S9!=1 && S10!=1 ){
  S11=1;
  PORTB.F0=1; PORTB.F1=1; PORTB.F2=0; PORTB.F3=1;}  //1101
else S11=0;
Delay_ms(500);
  ////////////////////Light Sensor////////////////////
if(light<10){
  S12=1;
  PORTB.F0=0; PORTB.F1=0; PORTB.F2=1; PORTB.F3=1;  }  //0011
else {
  S12=0;   }
if(light<15 && S12!=1 ){
  S13=1;
  PORTB.F0=1; PORTB.F1=0; PORTB.F2=1; PORTB.F3=1; }  //1011
else S13=0;
if(light<20 && S12!=1 && S13!=1 ){
  S14=1;
 PORTB.F0=0; PORTB.F1=1; PORTB.F2=1; PORTB.F3=1; }  //0111
else S14=0;
if(light<25 && S12!=1 && S13!=1 && S14!=1 ){
  S15=1;
  PORTB.F0=1; PORTB.F1=1; PORTB.F2=1; PORTB.F3=1; }  //1111
else S15=0;
Delay_ms(500);
```

```
  }
void main() {
  TRISB=0X00; // PORTB OUTPUT
  PORTB=0X00; // PORTB LOW
  T1CON = 0b00000011; // Prescaler (1:1), T1CS =1 for counter mode
ADC_Init();// CONFIG ADC
while(1){// infinity loop
  ////////////////Humidity sensor////////////////////
  TMR1L=0;
  TMR1H=0;
delay_ms(1000);
  T1= TMR1H;
  T1=T1<<8;
  T1=T1|TMR1L;  // OR OPERATION
  Humidity=569.91301651-0.0775253895*T1 ;
   ////////////////Tempreature sensor////////////////////
  Temp= ADC_Read(0); //Reading ADC channel 0.
  Temp=(Temp*4.88)/10; //Using the equation
  ////////////////pressure sensor////////////////////
pressure= ADC_Read(1); //Reading ADC channel 1.
pressure= 0.264909784*pressure-3.45-10.840; //Using the equation
   /////////////////Light Sensor/////////////////
light= ADC_Read(2); //Reading ADC channel 1.
light= (light/15)/2; //Using the equation
send();
  }}
```

L2: Receiver Code

```
//LCD Pins Definition
sbit LCD_RS at RD2_bit;
sbit LCD_EN at RD3_bit;
sbit LCD_D7 at RD7_bit;
sbit LCD_D6 at RD6_bit;
sbit LCD_D5 at RD5_bit;
sbit LCD_D4 at RD4_bit;
sbitLCD_RS_Direction at TRISD2_bit;
sbitLCD_EN_Direction at TRISD3_bit;
sbit LCD_D7_Direction at TRISD7_bit;
sbit LCD_D6_Direction at TRISD6_bit;
sbit LCD_D5_Direction at TRISD5_bit;
sbit LCD_D4_Direction at TRISD4_bit;
////////////////////////
bit S1,S2,S3,S4;
void main() {
```

```
S1=1;S2=0;S3=0;S4=0;
TRISB=0XFF;// portb input
TRISD=0X00; // portd output
UART1_Init(9600); // baud rate
Lcd_Init();                    // Initialize LCD
Lcd_Cmd(_LCD_CLEAR);           // Clear display
Lcd_Cmd(_LCD_CURSOR_OFF);      // Cursor off
while(1){
while(S1==1 && PORTB.F0==0 && PORTB.F1==0 &&
PORTB.F2==0 && PORTB.F3==0){
LCD_out(1,1,"Temp:");
Lcd_Out_Cp("30");      //Prints the result.
lcd_chr_cp(223);
lcd_out_cp("C");
  UART1_Write_Text("30");  // send 30degree to pc
Delay_ms(500);
  S1=0;S2=1;}
while(S1==1 && PORTB.F0==1 && PORTB.F1==0 &&
PORTB.F2==0 && PORTB.F3==0){
LCD_out(1,1,"Temp:");
Lcd_Out_Cp("36");      //Prints the result.
lcd_chr_cp(223);
lcd_out_cp("C");
  UART1_Write_Text("36");
Delay_ms(500);
  S1=0;S2=1;}
while(S1==1 && PORTB.F0==0 && PORTB.F1==1 &&
PORTB.F2==0 && PORTB.F3==0){
LCD_out(1,1,"Temp:");
Lcd_Out_Cp("42");      //Prints the result.
lcd_chr_cp(223);
lcd_out_cp("C");
  UART1_Write_Text("42");
Delay_ms(500);
  S1=0;S2=1;}
while(S1==1 && PORTB.F0==1 && PORTB.F1==1 &&
PORTB.F2==0 && PORTB.F3==0){
LCD_out(1,1,"Temp:");
Lcd_Out_Cp("48");      //Prints the result.
lcd_chr_cp(223);
lcd_out_cp("C");
  UART1_Write_Text("48");
Delay_ms(500);
```

```
  S1=0;S2=1;}
  ////////////////////////////////////////////////////
while(S2==1 && PORTB.F0==0 && PORTB.F1==0 &&
PORTB.F2==1 && PORTB.F3==0){
Lcd_Out(2,9,"Humi:");
Lcd_Out_Cp("78");      //Prints the result.
  UART1_Write_Text("78");
Delay_ms(500);
  S2=0;S3=1;}
while(S2==1 && PORTB.F0==1 && PORTB.F1==0 &&
PORTB.F2==1 && PORTB.F3==0){
Lcd_Out(2,9,"Humi:");
Lcd_Out_Cp("88");      //Prints the result.
  UART1_Write_Text("88");
Delay_ms(500);
  S2=0;S3=1;}
while(S2==1 && PORTB.F0==0 && PORTB.F1==1 &&
PORTB.F2==1 && PORTB.F3==0){
Lcd_Out(2,9,"Humi:");
Lcd_Out_Cp("94");      //Prints the result.
  UART1_Write_Text("94");
Delay_ms(500);
  S2=0;S3=1;}
while(S2==1 && PORTB.F0==1 && PORTB.F1==1 &&
PORTB.F2==1 && PORTB.F3==0){
Lcd_Out(2,9,"Humi:");
Lcd_Out_Cp("99");      //Prints the result.
  UART1_Write_Text("99");
Delay_ms(500);
  S2=0;S3=1;}
    ////////////////////////////////////////////////////////
while(S3==1 && PORTB.F0==0 && PORTB.F1==0 &&
PORTB.F2==0 && PORTB.F3==1){
Lcd_Out( 2, 1,"Pres:");
Lcd_Out_Cp("50");      //Prints the result.
  UART1_Write_Text("50");
Delay_ms(500);
  S3=0;S4=1;}
while(S3==1 && PORTB.F0==1 && PORTB.F1==0 &&
PORTB.F2==0 && PORTB.F3==1){
Lcd_Out( 2, 1,"Pres:");
Lcd_Out_Cp("58");      //Prints the result.
```

```
  UART1_Write_Text("58");
Delay_ms(500);
  S3=0;S4=1;}
while(S3==1 && PORTB.F0==0 && PORTB.F1==1 &&
PORTB.F2==0 && PORTB.F3==1){
Lcd_Out( 2, 1,"Pres:");
Lcd_Out_Cp("64");      //Prints the result.
  UART1_Write_Text("64");
Delay_ms(500);
  S3=0;S4=1;}
while(S3==1 && PORTB.F0==1 && PORTB.F1==1 &&
PORTB.F2==0 && PORTB.F3==1){
Lcd_Out( 2, 1,"Pres:");
Lcd_Out_Cp("76");      //Prints the result.
  UART1_Write_Text("76");
Delay_ms(500);
  S3=0;S4=1;}
  ////////////////////////////////////////////////////////////////////
while(S4==1 && PORTB.F0==0 && PORTB.F1==0 &&
PORTB.F2==1 && PORTB.F3==1){
Lcd_Out( 1,10," Lig:");
Lcd_Out_Cp("10");      //Prints the result.
  UART1_Write_Text("10");
Delay_ms(500);
  S4=0;S1=1;}
while(S4==1 && PORTB.F0==1 && PORTB.F1==0 &&
PORTB.F2==1 && PORTB.F3==1){
Lcd_Out( 1,10," Lig:");
Lcd_Out_Cp("14");      //Prints the result.
```

```
  UART1_Write_Text("14");
Delay_ms(500);
  S4=0;S1=1;}
while(S4==1 && PORTB.F0==0 && PORTB.F1==1 &&
PORTB.F2==1 && PORTB.F3==1){
Lcd_Out( 1,10," Lig:");
Lcd_Out_Cp("17");      //Prints the result.
  UART1_Write_Text("17");
Delay_ms(500);
  S4=0;S1=1;}
while(S4==1 && PORTB.F0==1 && PORTB.F1==1 &&
PORTB.F2==1 && PORTB.F3==1){
Lcd_Out( 1,10," Lig:");
Lcd_Out_Cp("22");      //Prints the result.
  UART1_Write_Text("22");
Delay_ms(500);
  S4=0;S1=1;}
}}
```