Embedded Systems and IoT Laboratory

|  |  |  |  |
| --- | --- | --- | --- |
| Course Code | 18CSL68 | Credits | 1.5 |
| Course type | LAB | CIE Marks | 25 marks |
| Hours/week: L-T-P | 0-0-3 | SEE Marks | 25 marks |
| Total Hours: | 36 | SEE Duration | 3 Hours for 50 marks |

List of experiments along with connections

|  |  |
| --- | --- |
| 1. | Develop an 8051 ‘C’ program to implement MOD-4 (UP/ DOWN) counter on LEDs connected to Port 2. Include 1second delay between each count. Generate delay using *for loop*.    Connections:  Port 2 to CN11 of Microcontroller Evaluation board    Program:  #include "at89c51ed2.h"    void delay(unsigned int);    void main(void){ while(1){ P2 = 0x00; delay(1000); P2 = 0x10; delay(1000); P2 = 0x20; delay(1000); P2 = 0x30;  delay(1000);  }  }    void delay(unsigned int itime){  unsigned int i,j; for(i=0;i<itime;i++){  for(j=0;j<1275;j++);  }  } |
| 2. | Develop an 8051 ‘C’ program to implement MOD-4 (UP/ DOWN) counter on LEDs connected to Port 2. Include 0.5 second delay between each count. Generate delay using for loop.    Connections:  Port 2 to CN11 of Microcontroller Evaluation board    Program:  #include "at89c51ed2.h"    void delay(unsigned int);    void main(void){ while(1){ P2 = 0x00; delay(500); P2 = 0x10; |
|  | delay(500); P2 = 0x20; delay(500); P2 = 0x30; delay(500);  }  }    void delay(unsigned int itime){  unsigned int i,j; for(i=0;i<itime;i++){  for(j=0;j<1275;j++);  }  } |
| 3. | Develop an 8051 ‘C’ program to implement MOD-4 counter on LEDs connected to Port 2 using Hardware delay. Use Timer1 in Mode 1 to generate a delay of ---- ms.    Connections:  Port 2 to CN11 of Microcontroller Evaluation board    Program:  #include "at89c51ed2.h"    void T1M1delay();    void main(void){ while(1){ P2 = 0x00;  T1M1delay();  P2 = 0x10;  T1M1delay();  P2 = 0x20;  T1M1delay();  P2 = 0x30;  T1M1delay();  }  }    void T1M1delay(){  TMOD = 0x10;  TL1 = //Calculate as per delay required;  TH1 = //Calculate as per delay required;  TR1 = 1;  while(TF1 == 0);  TF1 = 0;  TR1 = 0;  } |
| 4. | Develop an 8051 ‘C’ program to implement MOD-4 counter on LEDs connected to Port 2 using Hardware delay. Use Timer1 in Mode 2 to generate a delay of ---- ms.    Connections:  Port 2 to CN11 of Microcontroller Evaluation board    Program:  #include "at89c51ed2.h"    void T1M2delay(); |
|  | void main(void){ while(1){ P2 = 0x00;  T1M2delay();  P2 = 0x10;  T1M2delay();  P2 = 0x20;  T1M2delay();  P2 = 0x30;  T1M2delay();  }  }    void T1M2delay(){  TMOD = 0x20;  TH1 = //Calculate as per delay required;  TR1 = 1;  while(TF1 == 0);  TF1 = 0;  TR1 = 0;  } |
| 5. | Develop an 8051 ‘C’ program to generate the following waveforms using DAC 0800 interface i) Square  ii) Triangular    Connections:  Port 0 to CN15 of Microcontroller Evaluation board    Program:  i)Square wave  #include "at89c51ed2.h"    void delay(unsigned int);    void main(){ while(1){ P0 = 0x00; delay(200) P0 = 0xFF;  delay(200)  }  }    void delay(unsigned int itime){  unsigned int i,j; for(i=0;i<itime;i++){  for(j=0;j<1275;j++);  }  }    ii) Triangular  #include "at89c51ed2.h"    void main(){  unsigned char count; while(1){  for(count=0; count!=0xFF; count++){ P0 = count; |
|  | }  for(count=0xFF; count!=0; count--){  P0 = count;  }  }  } |
| 6. | Develop an 8051 ‘C’ program to generate the following waveforms using DAC 0800 interface i) Rectangular  ii) Positive Ramp    Connections:  Port 0 to CN15 of Microcontroller Evaluation board    Program:  i)Rectangular  #include "at89c51ed2.h"    void delay(unsigned int);    void main(){ while(1){ P0 = 0x00; delay(100) P0 = 0xFF;  delay(200)  }  }    void delay(unsigned int itime){  unsigned int i,j; for(i=0;i<itime;i++){  for(j=0;j<1275;j++);  }  }    ii)Positive Ramp #include "at89c51ed2.h"    void main(){  unsigned char count; while(1){  for(count=0; count!=0xFF; count++){  P0 = count;  }  P0 = 0;  }  } |
| 7. | Develop an 8051 ‘C’ program to generate the following waveforms using DAC interface i) Square ii) Negative Ramp    Connections:  Port 0 to CN15 of Microcontroller Evaluation board    Program: i)Square  #include "at89c51ed2.h" |
|  | void delay(unsigned int);    void main(){ while(1){ P0 = 0x00; delay(200) P0 = 0xFF;  delay(200)  }  }    void delay(unsigned int itime){  unsigned int i,j; for(i=0;i<itime;i++){  for(j=0;j<1275;j++);  }  }    ii) Negative Ramp: #include "at89c51ed2.h"    void main(){  unsigned char count; while(1){  for(count=0xFF; count!=0; count--){  P0 = count;  }  P0 = 0xFF;  }  } |
| 8. | Develop an 8051 ‘C’ program to interface 2x16 LCD display and to display the following two strings.  (Start displaying from 1st position on both lines)   1. KLS GIT 2. ESIoT LAB     Connections:  Port 2 to CN6 of Microcontroller Evaluation board    For KLS GIT → temp1 = 0x80  For ESIoT LAB → temp1 = 0xC0    Program:  #include "at89c51ed2.h"  #include<intrins.h>    void lcd\_init(); void lcd\_comm();  void lcd\_data();    unsigned char xdata arr1[16]={"KLS GIT"} unsigned char xdata arr2[16]={"ESIoT LAB"} unsigned char temp1 = 0x00; unsigned char temp2;  unsigned int i=0;    void main(void){  AUXR = 0x10; //To access full external RAM lcd\_init(); |
|  | temp1 = 0x80 // First line first position lcd\_comm(); for(i=0; i<7; i++){ temp2 = arr1[i]; lcd\_data();  }  temp1 = 0xC0 //Second line first position for(i=0; i<9; i++){ temp2=arr2[i]; lcd\_data();  }  } |
| 9. | Develop an 8051 ‘C’ program to interface 2x16 LCD display and to display the following two strings.  (Start displaying from 6th position on both lines) iii) CSE iv) BRANCH    Connections:  Port 2 to CN6 of Microcontroller Evaluation board    For CSE → temp1 = 0x85  For BRANCH → temp1 = 0xC5    Program:  #include "at89c51ed2.h"  #include<intrins.h>    void lcd\_init(); void lcd\_comm();  void lcd\_data();    unsigned char xdata arr1[16]="CSE"; unsigned char xdata arr2[16]="BRANCH"; unsigned char temp1 = 0x00; unsigned char temp2;  unsigned int i=0;    void main(void){  AUXR = 0x10; //To access full external RAM lcd\_init();  temp1 = 0x85; //First line 6th position lcd\_comm(); for(i=0; i<3; i++){ temp2 = arr1[i]; lcd\_data();  }  temp1 = 0xC5; //Second line 6th position for(i=0; i<6; i++){ temp2 = arr2[i]; lcd\_data();  }  } |
| 10. | Develop an Embedded ‘C’ program to blink the LEDs connected to Arduino SBC upon pressing the push buttons.    buttonPin1 – 13, buttonPin2 – 12, buttonPin3 – 11, buttonPin4 – 10  ledPin1 – A5, ledPin2 – A4, ledPin3 – A3, ledPin4 – A2 Connections: CN9 to CN4 |
|  | Program:  const int buttonPin1 = 13;  int buttonState1 = LOW;    const int buttonPin2 = 12;  int buttonState2 = LOW;    const int buttonPin3 = 11;  int buttonState3 = LOW;    const int buttonPin4 = 10;  int buttonState4 = LOW;    const int ledPin1 = A5; const int ledPin2 = A4; const int ledPin3 = A3; const int ledPin4 = A2;    void setup() {  // put your setup code here, to run once:  pinMode(buttonPin1, INPUT); pinMode(buttonPin2, INPUT); pinMode(buttonPin3, INPUT); pinMode(buttonPin4, INPUT); pinMode(ledPin1, OUTPUT); pinMode(ledPin2, OUTPUT); pinMode(ledPin3, OUTPUT); pinMode(ledPin4, OUTPUT);  Serial.begin(9600);  }    void loop() {  // put your main code here, to run repeatedly:  buttonState1 = digitalRead(buttonPin1); buttonState2 = digitalRead(buttonPin2); buttonState3 = digitalRead(buttonPin3); buttonState4 = digitalRead(buttonPin4);    if(buttonState1 == HIGH)  digitalWrite(ledPin1,HIGH); else  digitalWrite(ledPin1,LOW);  Serial.println(buttonState1);    if(buttonState2 == HIGH)  digitalWrite(ledPin2,HIGH); else  digitalWrite(ledPin2,LOW);  Serial.println(buttonState2);    if(buttonState3 == HIGH)  digitalWrite(ledPin3,HIGH); else  digitalWrite(ledPin3,LOW);  Serial.println(buttonState3);    if(buttonState4 == HIGH)  digitalWrite(ledPin4,HIGH); else  digitalWrite(ledPin4,LOW); |
|  | Serial.println(buttonState4);    } |
| 11. | Develop an Embedded ‘C’ program to interface the sensor DHT11 to Arduino SBC and display the data acquired from sensors on serial monitor.    #define DHT11\_PIN 4    Sketch → Include Library → Add Zip Library of DHT (#include<dht.h>) Connections:  RM2 to RM19    Program:  #include<dht.h>  #define DHT11\_PIN 4    dht DHT; void setup() {  // put your setup code here, to run once:  Serial.begin(9600);  }    void loop() {  // put your main code here, to run repeatedly:  int chk = DHT.read11(DHT11\_PIN);  Serial.print("Temperature: ");  Serial.println(DHT.temperature);  Serial.print("Humidity: "); Serial.println(DHT.humidity); delay(2000);  } |
| 12. | Develop an Embedded ‘C’ program to control the relay through Arduino UNO.    Connections:  RM17 to RM9    Program:  int relay\_pin = 8;    void setup() {  // put your setup code here, to run once:  pinMode(relay\_pin,OUTPUT); Serial.begin(9600);  digitalWrite(replay\_pin,HIGH);  }    void loop() {  // put your main code here, to run repeatedly:  digitalWrite(relay\_pin,LOW); Serial.println("Relay is OFF"); delay(1000);  digitalWrite(relay\_pin,HIGH); Serial.println("Relay is ON"); delay(1000);  } |
| 13. | Develop an Embedded ‘C’ program to interface the sensor LDR to Arduino SBC and display the data acquired from sensor on serial monitor. |
|  | light\_data → TRUE → indicates Darkness light\_data → FALSE → indicates Brightness    Connections:  RM3 to RM20    Program:  int light\_pin = 5;    void setup() {  // put your setup code here, to run once:  pinMode(light\_pin,INPUT);  Serial.begin(9600);  }    void loop() {  // put your main code here, to run repeatedly: int light\_data = digitalRead(light\_pin) if(light\_data)  Serial.println("Light not detected"); else  Serial.println("Light detected"); delay(1000);  } |
| 14. | Develop an Embedded ‘C’ program to blink the LEDs connected to Arduino SBC upon pressing the push buttons.    Connections:  CN9 to CN4    Same as Experiment 10 |
| 15. | Develop an Embedded ‘C’ program to interface the sensor DHT11 to Arduino SBC and display the data acquired from sensors on serial monitor. Turn ON the relay when temperature is greater than 22 degree centigrade.    Connections:  RM2 to RM19  RM17 to RM9      #include<dht.h>  #define DHT11\_PIN 4    int relay\_pin = 8;    void setup() {  pinMode(relay\_pin, OUTPUT); digitalWrite(relay\_pin,LOW);  Serial.begin(9600);  }    Void loop() {  Int chk = DHT.read11(DHT11\_PIN);  Serial.print(“Temperature: ”);  Serial.println(DHT.temperature);    if( DHT.temperature > 22) digitalWrite(relay\_pin,HIGH); |
|  | else  digitalWrite(relay\_pin,LOW);    delay(2000);  } |
|  |  |