“Heaven’s Light is Our Guide”

Department of Computer Science & Engineering

RAJSHAHI UNIVERSITY OF ENGINEERING & TECHNOLOG

Lab Report-01

Date of Experiment: 12/05/2024

Date of Submission: 25/05/2024

Submitted By:
Name: Khandoker Sefayet Alam
Roll:2003121
Department: Computer Science & Engineering
Section-C

Course code: CSE 3106
Course name: Computer Interfacing and Embedded Systems Sessional

Submitted To:
Md. Nasif Osman Khansur
Lecturer
Department of Computer Science and Engineering,
RUET

Experiment-01:

Title: Blinking LED

Introduction:

The experiment of blinking LED light after constant interval time has been conducted with an aim to get introduced with microcontroller programming and understand the interface between software and hardware components.

Equipments:

1. Microcontroller
2. LED
3. Resistor
4. Breadboard
5. Jumper Wires

Procedure:

Using breadboard and jumper wires, the LED has been connected to the microcontroller through a resistor where the microcontroller programming has been done using Arduino IDE.

The Arduino code is as below:

```
void setup() {

// put your setup code here, to run once:
pinMode(PC13, OUTPUT);
}

void loop() {
// put your main code here, to run repeatedly:
digitalWrite(PC13, HIGH);
delay(1000);
digitalWrite(PC13, LOW);
delay(1000);
}
```

Explanation:

In setup() function, PC13 has been set up as the output pin. The LED has been connected to the PC13 pin.

In loop() function, the output pin has been directed to hold high voltage using digitalWrite function(pin no, status). The delay function has been used to hold the state and wait for 1000 ms or 1 second. Then another digitalWrite function has been used to put low voltage in the output pin. The delay function has been used again to wait for 1s. As the loop() is a repetitive function, it'll make the LED blink continuously.

Result and discussion:

The procedures successfully made the LED blink after 1 second interval each time. Throughout this experiment, an introduction to digital output and timing control using microcontroller programming has been successfully conducted.

Experiment-02:

Title: Turning On a LED

Introduction:

The experiment of turning on a LED has been conducted with an aim to get introduced with microcontroller programming and output showing on external hardwares.

Equipments:

1. Microcontroller
2. LED
3. Resistor
4. Breadboard
5. Jumper Wires

Procedure:

Using breadboard and jumper wires, the LED has been connected to the microcontroller through a resistor where the microcontroller programming has been done using Arduino IDE.

The Arduino code is as below:

```
void setup()
{
pinMode (PA15, OUTPUT);
// Set the LED pin as an output
digitalWrite(PA15, HIGH);
// Set the LED pin high
}
void loop()
{
// We are not doing anything in the loop!
}
```

Explanation:

In setup() function, PA15 has been set up as the output pin. The LED has been connected to the PA15 pin. the output pin has been directed to hold high voltage using digitalWrite function(pin no, status) In loop() function, nothing is written as the LED has been directed to be ON using setup function.

Result and discussion:

The procedures successfully made the LED turn ON. Throughout this experiment, an introduction to turning LED ON and using microcontroller programming has been successfully conducted.

Experiment-03:

Title: LED with changing Blink rate

Introduction:

The experiment of LED with changing Blink rate has been conducted with an aim to get introduced with how to change the rate of blinking of a LED using microcontroller programming.

Equipments:

1. Microcontroller
2. LED
3. Resistor
4. Breadboard
5. Jumper Wires

Procedure:

Using breadboard and jumper wires, the LED has been connected to the microcontroller through a resistor where the microcontroller programming has been done using Arduino IDE.

The Arduino code is as below:

```
void setup() {

  pinMode(PA15, OUTPUT);
  // Set the LED pin as an output
}
void loop() {
  for (int i = 100; i <= 1000; i = i + 100) {
    digitalWrite(PA15, HIGH);
    delay(i);
    digitalWrite(PA15, LOW);
    delay(i);
  }
}
```

Explanation:

In setup() function, PA15 has been set up as the output pin. The LED has been connected to the PA15 pin.

In loop() function, there's a loop using 'for'. The blinking rate has been changed by changing the delay time. An iteration of i=100 to i=1000 has been made to set different delay time and different blinking rate. In the for loop, the output pin has been directed to hold high voltage using digitalWrite function(pin no, status). The delay function has been used to hold the state and wait for i ms. Then another digitalWrite function has been used to put low voltage in the output pin. The delay function has been used again to wait for i ms.

Result and discussion:

The procedures successfully made the LED blink after i second interval each time for each iteration. Throughout this experiment, an introduction to control over blinking rate using microcontroller programming has been successfully conducted.

Experiment-04:

Title: Fading LED

Introduction:

The experiment of 'Fading LED' has been conducted with an aim to get introduced with how to make an LED fade in and out using microcontroller programming.

Equipments:

1. Microcontroller
2. LED
3. Resistor
4. Breadboard
5. Jumper Wires

Procedure:

Using breadboard and jumper wires, the LED has been connected to the microcontroller through a resistor where the microcontroller programming has been done using Arduino IDE.

The Arduino code is as below:

```
void setup() {

  pinMode(PB0, OUTPUT);
  //Set the LED pin as an output
}
void loop() {
  for (int i = 0; i < 256; i++) {
    analogWrite(PB0, i);
    delay(10);
  }
  for (
    int i = 255; i >= 0; i--) {
    analogWrite(PB0, i);
    delay(10);
  }
}
```

Explanation:

In setup() function, PB0 has been set up as the output pin. The LED has been connected to the PB0 pin.

In loop() function, there are 2 loops using ‘for’. The first for loop has been used analogWrite() the LED by pin no and intensity. As an iteration over i=0 to i=255 has been made where i has been used to fix the brightness/intensity of the LED. Firstly an iteration has been made in ascending order and this made the LED fade in. Using the second for loop an reverse iteration has been made so that the LED fade out over time and the brightness fades out. Proper delay functions has been used for user comfort.

Result and discussion:

The procedures successfully made the LED fade in and fade out. Throughout this experiment, an introduction to control over LED intensity using microcontroller programming has been successfully gained.

Experiment-05:

Title: Simple LED control with a button

Introduction:

The experiment of ‘Simple LED control with a button’ has been conducted with an aim to get introduced how to setup to get control over a LED with button.

Equipments:

1. Microcontroller
2. LED
3. Resistor
4. Breadboard
5. Jumper Wires

Procedure:

Using breadboard and jumper wires, the LED has been connected to the microcontroller through a resistor where the microcontroller programming has been done using Arduino IDE.

The Arduino code is as below:

```
const int LED = PB0;

// The LED is connected to pin 9
const int BUTTON = PC15;
// The Button is connected to pin 2
void setup() {
  pinMode(LED, OUTPUT);
  // Set the LED pin as an output
  pinMode(BUTTON, INPUT);
  // Set button as input (not required)
}
void loop() {
  if (digitalRead(BUTTON) == LOW) {
    digitalWrite(LED, LOW);
  } else {
    digitalWrite(LED, HIGH);
  }
}
```

Explanation:

At first some constants has been declared for further use. The input pin of button (PC15) and the output pin of LED (PB0) has been declared as ‘BUTTON’ and ‘LED’ respectfully.

In setup() function, pin ‘LED’ has been set up as the output pin and ‘BUTTON’ has beem set up as the input pin.

In loop() function, there’s a condition. If the digitalRead() function returns LOW for input from BUTTON pin, the LED has been turned off using digitalWrite function. Else, the LED has been turned on using digital write function.

Result and discussion:

The procedures successfully gave control over LED using a button. Throughout this experiment, an introduction to how control a LED using a button by microcontroller programming has been successfully conducted.