**Outline**

Write a program to blink the on-board LED based on user commands from the serial monitor. Parse commands to turn on and off the LED as well as blink it a specified number of times.

**Objectives**

**Prerequisites**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Prerequisite Module(s)** | **Level** | **Student Initial** | **Teacher Initial** | **Date** |
| None |  |  |  |  |

**Materials**

* Arduino Development Environment (IDE)
* Arduino proto board
* Web Documentation: “Smraza UNO Documentation” folder in the “ICS3C0” Repository
  + Getting started guide.pdf
  + Public\_materials🡪Ebook🡪Arduino book.pdf
  + Lessons Folder

**Level 0: Serial Write**

1. Research how to write messages to the Arduino Serial Monitor.
   1. Google “Arduino Serial Write”
   2. Google “Arduino Serial Monitor”
2. Modify the basic blink examples program to:
   1. Write “On” to the serial monitor when the LED turns on.
   2. Write “Off” to the serial monitor when the LED turns off.
   3. Print each message on a new line.

**Modified Code:**

/\*

\*/

void setup() {

Serial.begin(9600);

pinMode(LED\_BUILTIN, OUTPUT);

}

void loop() {

Serial.println();

digitalWrite(LED\_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)

Serial.println("LED ON " );

delay(1000);

Serial.println();

digitalWrite(LED\_BUILTIN, LOW);

Serial.println("LED OFF " );

delay(1000);

}

**Level 1: Serial Read**

1. Research how to read messages from the Arduino Serial Monitor into an Arduino program sketch.
   1. Google “Arduino Serial Read”
2. Modify the basic blink examples program to:
   1. Turn on the LED when the user types “On”.
   2. Turn off the LED when the user types “Off”.
   3. Write the status of the LED to the serial monitor.

**Modified code:**

/\*\*/void setup(){Serial.begin(9600);pinMode(LED\_BUILTIN, OUTPUT);}void loop() { int val = 0;//waiting for inputwhile(Serial.available()==0);val=Serial.parseInt(); //read Int or parseFloat for...Float...Serial.println(val);for(int i=0;i>val;i++){digitalWrite(LED\_BUILTIN, HIGH); delay(1000); digitalWrite(LED\_BUILTIN, LOW); delay(1000); }}

**Level 2: Number Blink**

1. Research how to read numbers from the Arduino Serial Monitor into an Arduino program sketch.
2. Modify the basic blink examples program to:
   1. Read a number from the serial monitor
   2. Blink the LED the same number of times indicated.
   3. Turn the LED off and wait for a new number.

**Modified code:**

/\*\*/void setup(){Serial.begin(9600);pinMode(LED\_BUILTIN, OUTPUT);}void loop(){ int val = 0; //waiting for input while(Serial.available()==0); val=Serial.parseInt(); //read Int or parseFloat for...Float... Serial.println(val); for(int i=0;i<val;i++) {digitalWrite(LED\_BUILTIN, HIGH); delay(1000); digitalWrite(LED\_BUILTIN, LOW); delay(1000); }}

**Level 3: Robustness**

1. Modify your Level 2 program to also check for “On” or “Off” commands as well as numbers.
2. Extension: Modify your program to read any string from the serial monitor and do the first valid command in the string.

**Level 4: Quiz**

1. Complete the quiz assigned by your teacher.

**Achievement Record**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attainment Level** | **Student Initial** | **Teacher Initial** | **Date** |
| Level 0: Serial Write |  |  |  |
| Level 1: Serial Read |  |  |  |
| Level 2: Number Blink |  |  |  |
| Level 3: Robustness |  |  |  |
| Level 4: Quiz |  |  |  |