

# ELE206 Lab #1

William Ohley  
Jing Wang

## Abstract:

Arduino is an open-source microcontroller platform, which is designed for students, artists, designers, researchers, and hobbyists to learn programming, fast prototype and put their creative ideas into final professional production with quite low cost. Since the boot loader has been programmed into the on-board microcontroller, developers can upload the programs/sketches to on board flash memory without using a special programmer. Developers can compile and upload sketches with only one software environment, that is, the Arduino Integrated Development Environment (IDE).

Lab #1 is divided into two parts. Part 1 covers the basic software and hardware environment of Arduino by uploading example sketches into Arduino board. In part two, input from keyboard and how to control the blinking rate of LED using user input is explored.

## Methods:

Part 1 of lab #1 consists of uploading examples that are in the IDE to Arduino and examining the result.

For part 2, It is learned how to use the Serial library to get the input from the keyboard and change the blinking rate of the LED using the input values. The example sketch “SerialEvent” was examined. The combination of this example and example “Blink” could solve the problem. Basically, when any key is hit, Arduino can read the ASCII value for that key and save it into an integer type variable by “inChar”. This value can then control the blinking rate of LED. However, knowledge of the ASCII table is necessary to figure out what is adjusted in the blinking rate example. Thus, the code is modified and when the user types in the digit 1~9, the code will change the blinking rate from 1 second to 9 seconds. To achieve this, simply subtract ASCII '0' from inChar. It is a bit complicated to read two digits from keyboard input using Serial.read() and a loop. But the Serial library also provides a very easy-to-use function called “Serial.parseInt()”, which can read digits of any length from keyboard input. To simply the code, Serial.parseInt() is used in the lab.

The model used is Arduino UNO R2 and the hardware resources include: one Atmel Amega328 microcontroller, 14 digital I/O pins, six of which can produce pulse-width modulated (PWM) signals, and six analog inputs, SPI, external interrupts and serial communication RX/TX etc.. There are also a lot of commercial shields available and could be used to extend the function of basic Arduino board, such as Ethernet shield, Wi-Fi shields and blue-tooth shields etc. And only the basic Arduino board is needed in lab#1

## Results:

### Part 1:

Before a sketch is loaded into the Arduino, the “tools” menu is used to check the port to which the Arduino board is connected. Once the right port is selected, it is possible to upload sketches to Arduino. The example sketch uses the built-in LED connected to PIN 13 and controls the time that LED turns on and turns off, separately. It is also necessary to know how to use the other digital port to control LED.

Thus, the pin definition is changed from “int led = 13” to “int led = 9” and an LED is plugged into PIN 9 and GND, as illustrated in Fig.1

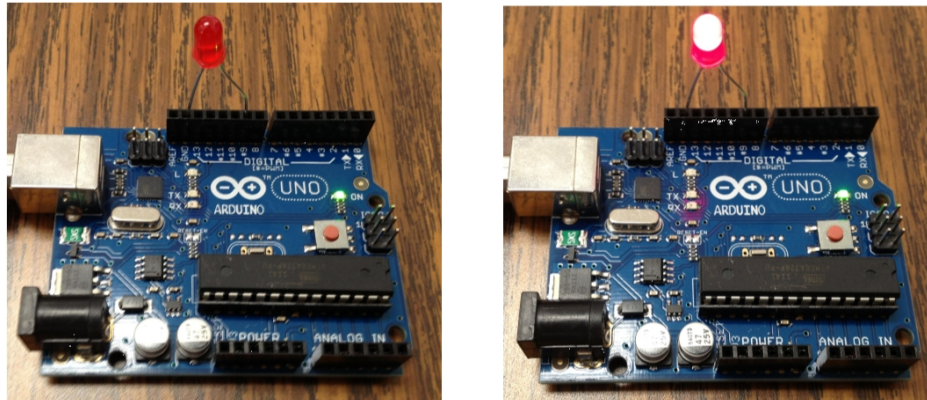


Fig.1. Connect the LED to PIN9 and GND

## Part 2:

As mentioned before, the two example sketches, “SerialEvent” and “Blink” are combined together and the new sketch works very well and can read single digits from keyboard and display the reading on serial monitor, a snapshot of serial monitor is shown in Fig.2. Serial.read() was replaced with Serial.parseInt() and the sketch works perfectly and it can read digits of any length as shown in Fig.3. The full listing of the sketch, including comments, is attached to this report.

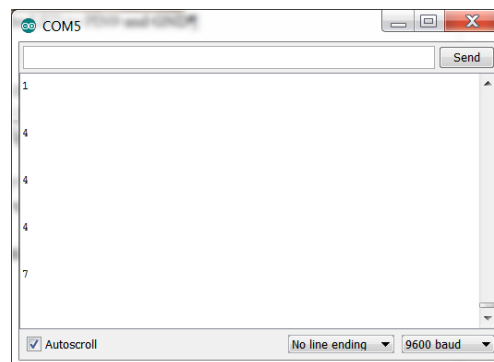


Fig.2. Part 2: Get one digit at time from keyboard using Serial.read()

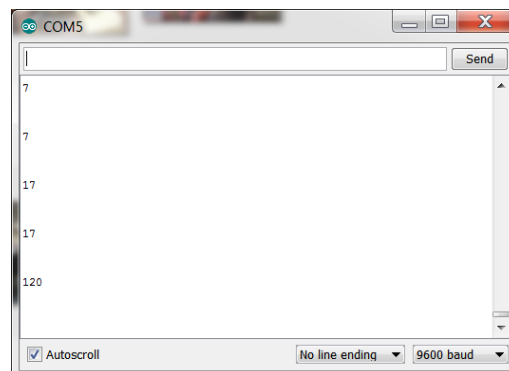


Fig.3. Part 2: Get digits of any length at time from keyboard using Serial.parseInt()

**Discussion:**

All the tasks in lab #1 were successfully completed. It was learned how to setup the IDE for Arduino and how to upload sketches. It was also learned how to use `Serial.read()` and `Serial.parseInt()` to read keyboard input and use the input value to change LED blinking rate. TA Jerry suggested to write code to read two digits from keyboard using `Serial.read()` and a loop, this will be tried later to practice programming skills.