Mechatronics Design – Class#2

Liwei Lin

Professor, Dept. of Mechanical Engineering
Co-Director, Berkeley Sensor and Actuator Center
The University of California, Berkeley, CA94720
e-mail: lwlin@me.berkeley.edu
http://www.me.berkeley.edu/~lwlin
Outline

- Announcements
- Lab #1
- Project Team discussions
Lab #1

Labs are online:

- Professor Lin’s website: 
  http://www.me.berkeley.edu/~lwlin/me102B/2016fall.html
- ME102 course website: 
  http://courses.me.berkeley.edu/ME102B/labs.html
Lab #1

- ME102 course website:

- Datasheets here!
Lab #1

◆ Professor Lin’s website:
Lab #1

ME102 Lab 1: Getting Started

This lab will teach you how to setup the IDE (Integrated Development Environment) for the Arduino. These instructions have been tested for the computers available in the lab. However, it is just as easy to setup the IDE on your own laptop. You may wish to do this for future lab assignments.

1. Introduction to the Arduino
   - Kit contents check
   - Arduino IDE and some terminology
   - Hello Worlds
   - Anatomy of a sketch

2. Setting up your hardware buffers
   - Digital input buffering with the 74LS14
   - Digital output buffering with the 7417
   - ADC setup with the 4342

3. Clean up and check off
   - Blinking LED with the buffered setup

4. References

Learn the Arduino interface
## Lab #1

### ME102 Lab 1: Getting Started

This lab will teach you how to setup the IDE (Integrated Development Environment) for the Arduino. These instruction have been tested for the computers available in lab. However, it is just as easy to setup the IDE on your own laptop. You may wish to do this for future lab assignments.

1. Introduction to the Arduino
   - Kit contents check
   - Arduino IDE and some terminology
   - Hello Worlds
   - Anatomy of a sketch

2. Setting up your hardware buffers
   - Digital input buffering with the 74LS14
   - Digital output buffering with the 7417
   - ADC setup with the 4342

3. Clean up and check off
   - Blinking LED with the buffered setup

4. References

Set up **input** and **output** buffers  
Set up ADC **voltage** follower
Lab #1

ME102 Lab 1: Getting Started

This lab will teach you how to setup the IDE (Integrated Development Environment) for the Arduino. These instructions have been tested for the computers available in lab. However, it is just as easy to setup the IDE on your own laptop. You may wish to do this for future lab assignments.

1. Introduction to the Arduino
   - Kit contents check
   - Arduino IDE and some terminology
   - Hello Worlds
   - Anatomy of a sketch
2. Setting up your hardware buffers
   - Digital input buffering with the 74LS14
   - Digital output buffering with the 7417
   - ADC setup with the 4342
3. Clean up and check off
   - Blinking LED with the buffered setup

Test with blinking LED
Kit contents check

Before we continue any further, please make sure that you have at least the following in your lab kit.

- 1 Arduino Duemilanove
- 1 USB A-B cord
- 1 breadboard
- 5 LED’s of assorted colors
- 2 tactile switches
- 1 DC motor
- 1 tiny RC servo
- 1 potentiometer
- 1 optical encoder (datasheet)
- 1 pair of wire cutters
- 1 74LS14 (datasheet)
- 1 DM7417 (datasheet)
- 1 OPA4342 (datasheet) or TLV2374I (datasheet)

Input buffer
Output buffer
ADC voltage follower

If you have any problems identifying your parts, please notify a GSI
If you are missing any parts, please find Tom Clark
Arduino
Labs are online:

- Teensy Arduino website: https://www.pjrc.com/teensy/teensy31.html#specs
- Code libraries https://www.pjrc.com/teensy/usb_debug_only.html
- Professor Lin’s website: http://www.me.berkeley.edu/~lwlin/me102B/2016f.html

ME102 course website: http://courses.me.berkeley.edu/ME102B/labs.html
Arduino IDE

The largest benefit of the Arduino is the fact that you are not limited to programming in the lab. Instructions on how to install a copy of the Arduino IDE on your own machine can be found here (Mac, Windows and Linux).

Start the Arduino IDE by clicking the Arduino icon on the desktop.

For those of you who are familiar with microprocessor programming, the first thing you will notice is the Arduino’s minimalist approach.

There are 7 shortcut commands that you can use with the Arduino IDE:

- **Serial Monitor** opens the only debugging tool you have with the Arduino. The Serial Monitor displays information passed from the Duemilanove to the computer.
- **Upload** compiles your program (“sketch” in the parlance of the creators of the Arduino, and sends it to the board if there are no compile-time errors.
- **Save your sketch**
- **Open an existing sketch**
- **New** creates a new sketch
- **Stop** interrupts compilation of your code.
- **Compile** your sketch in order to check for compile-time errors. As your sketch grows in size, it’ll save time to verify that your code compiles before trying to upload it.
1. Tools → Board (select the right Arduino version)
2. Computer → Device Manager → Ports → USB Serial (COM3)
3. Tools ➔ Serial Port ➔ COM3
5. Files ➔ Examples ➔ Stubs ➔ HelloWorld (load the program)
6. Verify the Screen (successful communication)
6. File → Examples → Digital → Blink (load a new program)
Software Introduction – “HelloWorld”

```c
void setup() {
  Serial.begin(9600);
}

void loop() {
  Serial.println("Hello World!");
}
```

- where to use the board
- please find out this yourself
- serial communication
- never-ending loop
- serial print on screen

Liwei Lin, University of California at Berkeley
Software Introduction – “Blink”

```c
int ledPin = 13;  // LED connected to digital pin 13

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop()
{
  digitalWrite(ledPin, HIGH);
  delay(1000); // wait 1000 ms
  digitalWrite(ledPin, LOW);
  delay(1000);
}
```

Liwei Lin, University of California at Berkeley
Hardware - Setup

- Set 5Volt on one rail of the breadboard
- (the breadboard can also be powered by the Arduino board but the current is limited to 50mA by the Arduino)
- Connect the common ground (0Volt) to one rail of the breadboard
LAB #1
Digital output buffering with the DM7417 (datasheet) protects the Arduino. The DM7414 is a hex buffer with open collector high voltage output. An open-collector only guarantees that a LOW input results in a LOW output. There are no guarantees for a HIGH input.