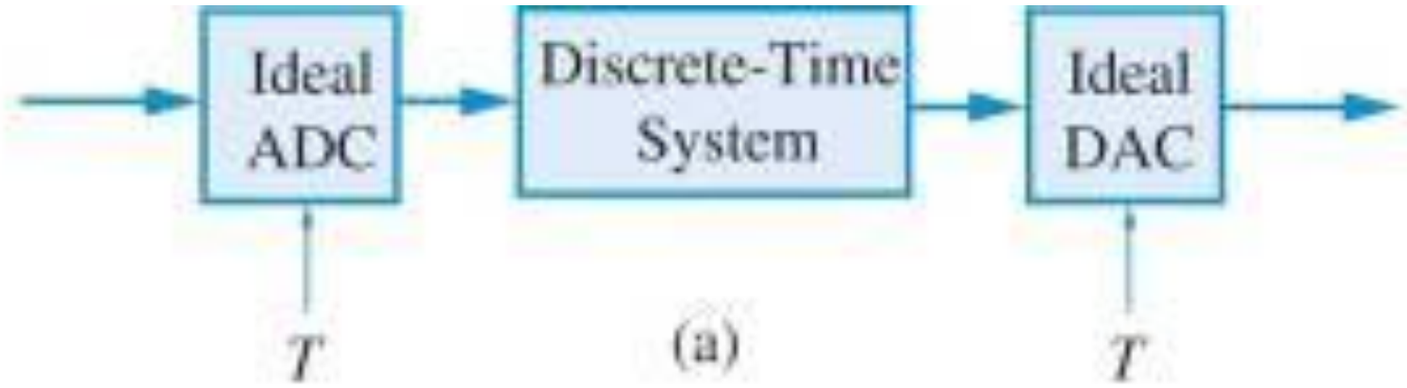
Unit 2: Biomedical Application – ECG Signal Acquisition and Processing

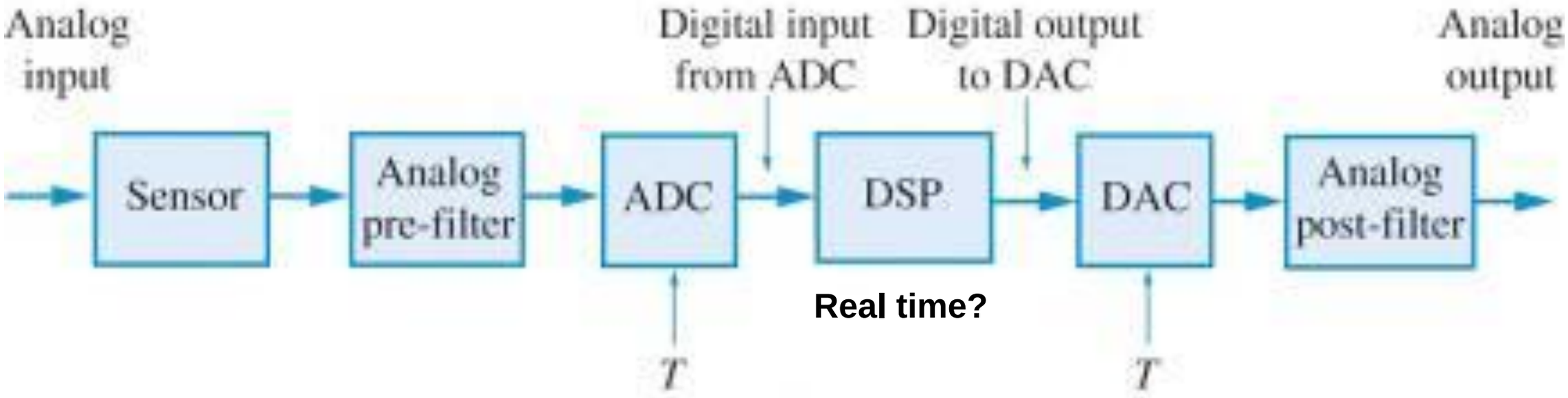
What Will You Learn in Unit 2?

- Experience DSP flow via ECG signal acquisition and processing
- Implement real time ECG display (i.e., build up a digital oscilloscope acquiring and displaying ECG) and heart rate estimation

DSP Flow Nowadays



(a)

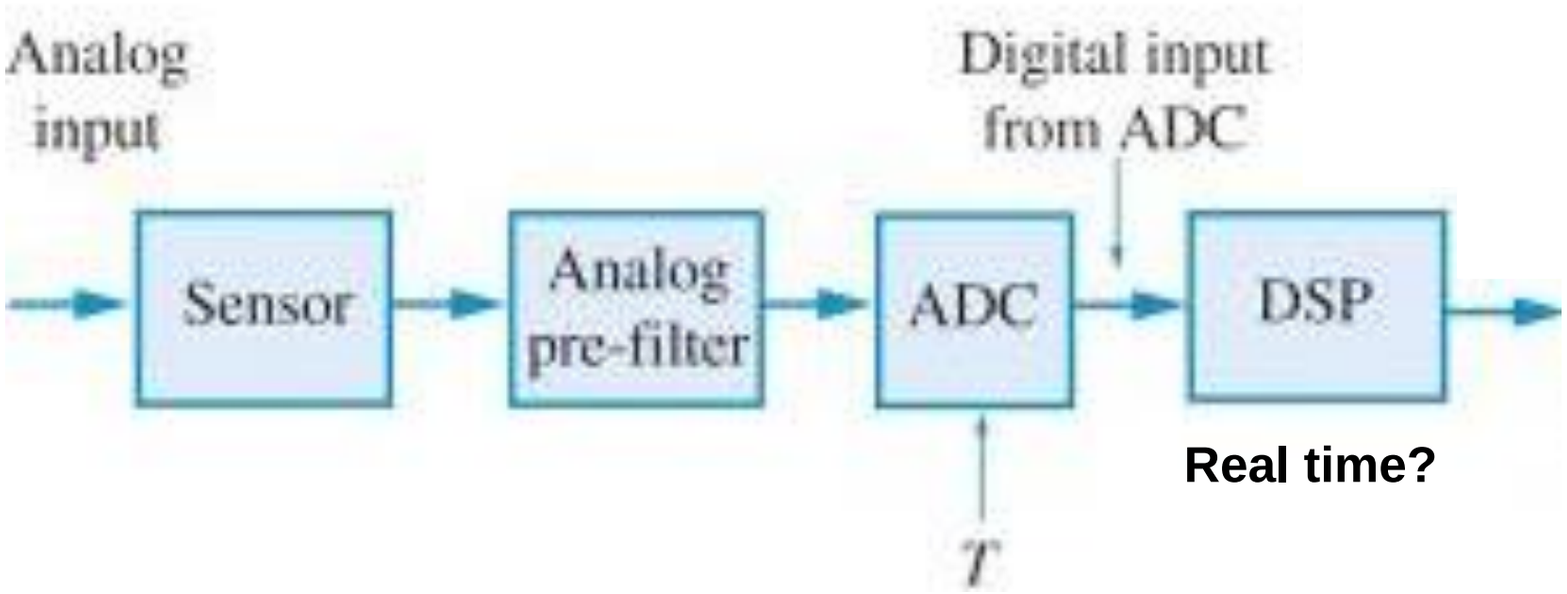


Real time?

(b)

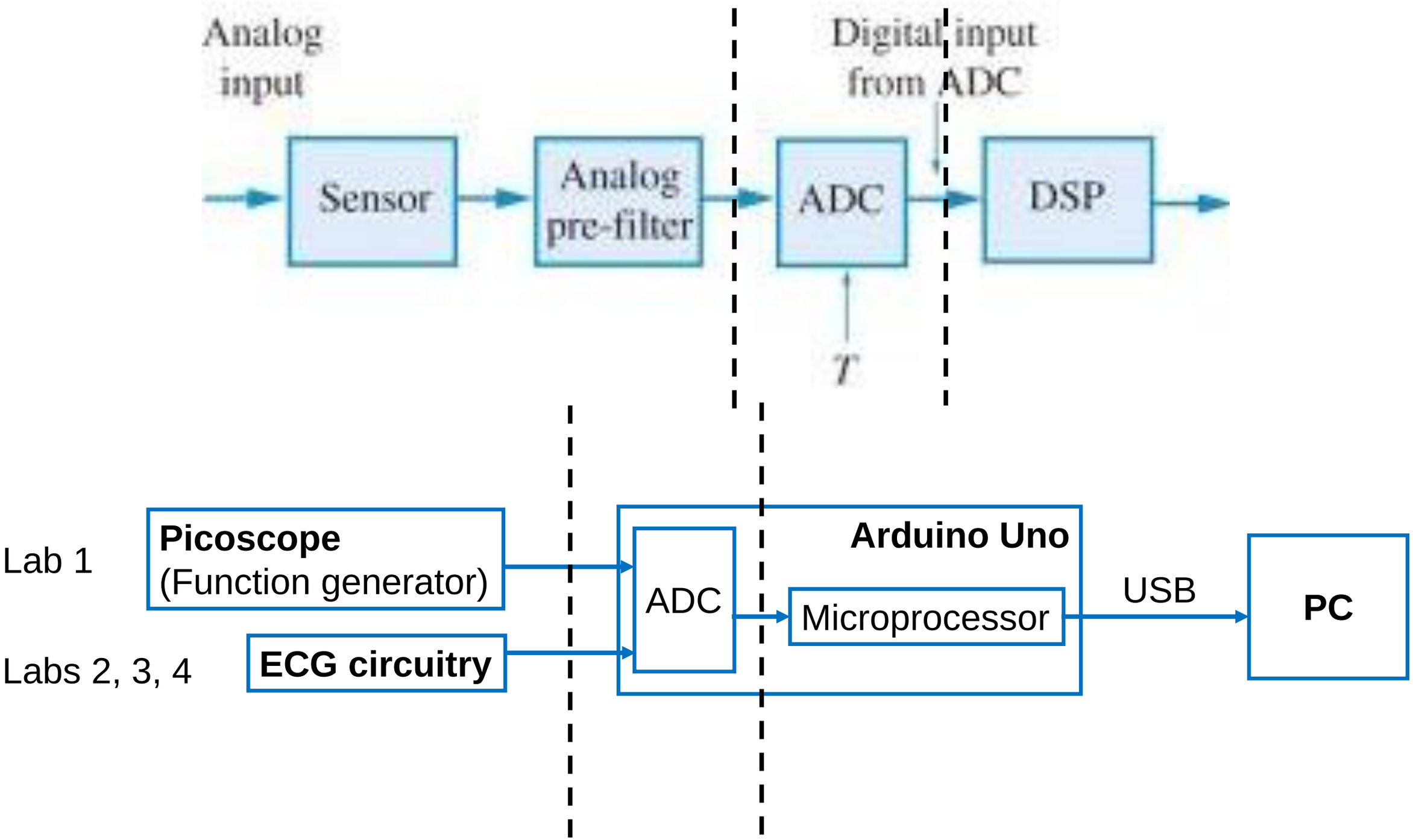
Figure 1.13 Simplified block diagram of idealized system for (a) continuous-time processing of discrete-time signals, and (b) its practical counterpart for digital processing of analog signals.

DSP Flow in Unit 2



Real time?

Unit 2 DSP Flow vs. Unit 2 Setup



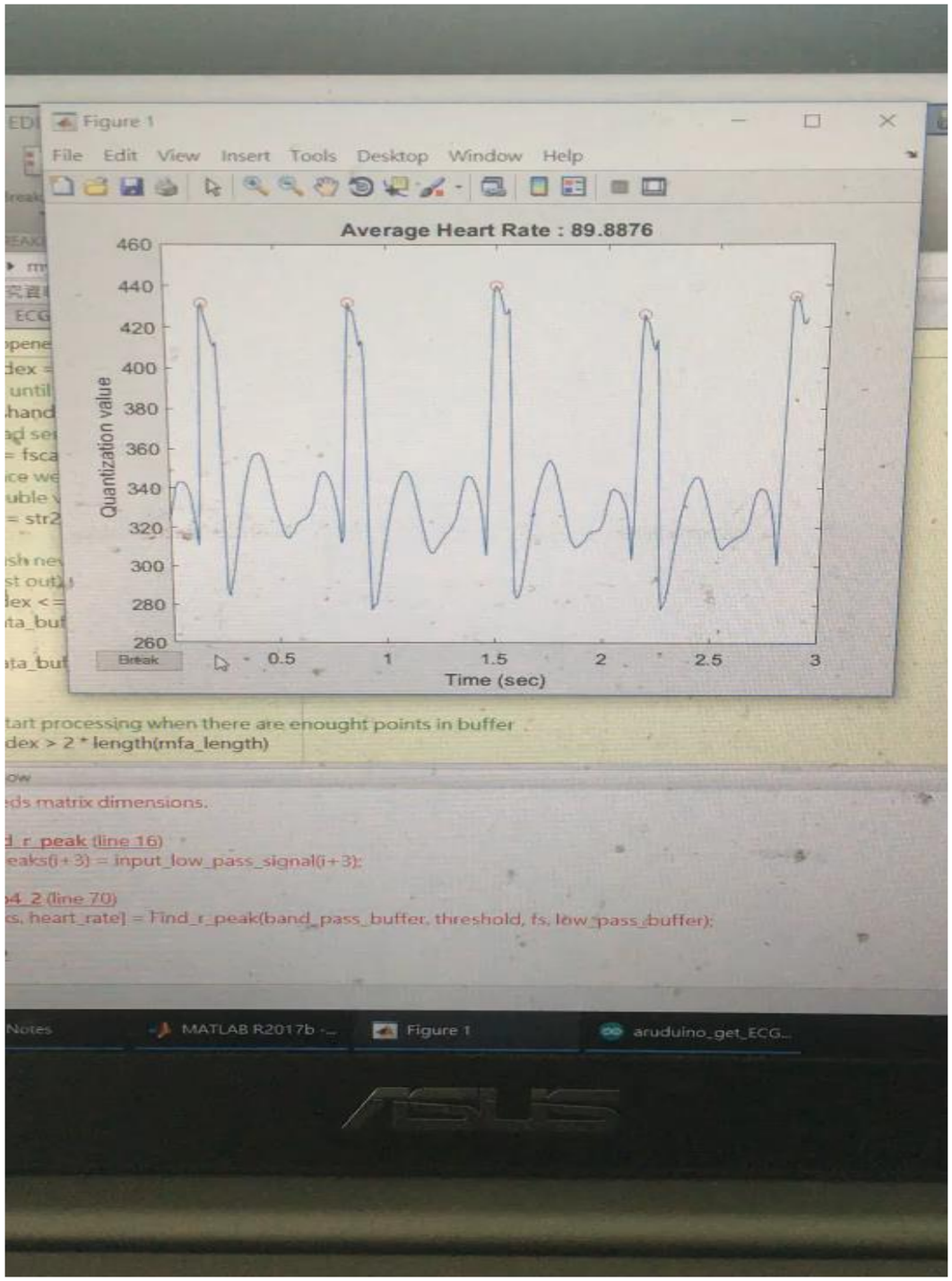
10/24: Lab 1 Introduction to Arduino Platform

10/31: Lab 2 ECG circuits, Signal Sampling and Digitization

11/07: Lab 3 Pre-processing of ECG Signals

11/14: Lab 4 Heart Rate Estimation

Goal: Real Time ECG Display and Heart Rate Estimation



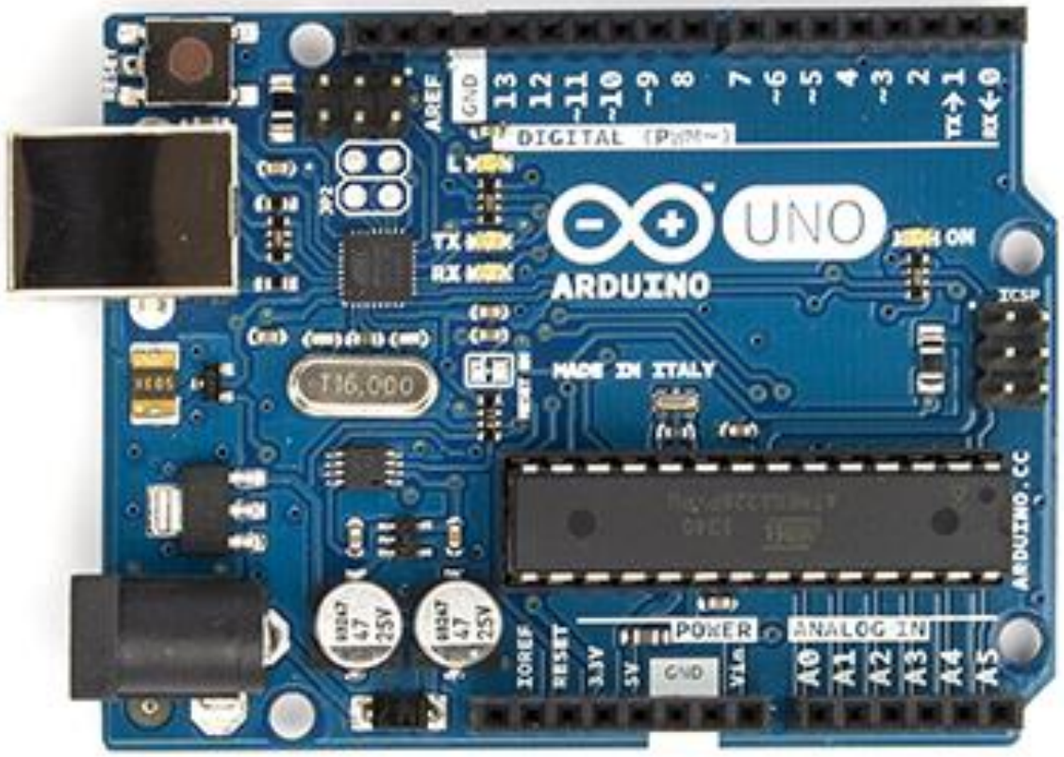
Lab 1:
Introduction to Arduino Platform

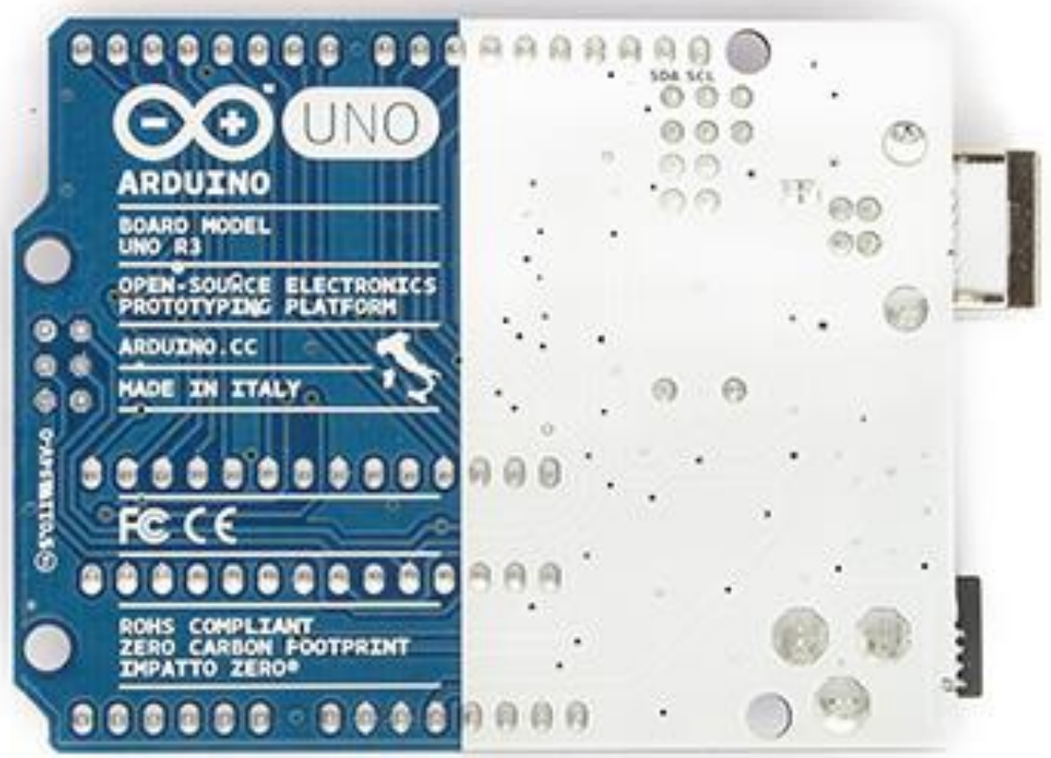
What Is Arduino

- Arduino is an open-source electronics platform based on easy to use hardware and software.
- You can find all the documentation at http://arduino.cc/
- http://arduino.cc/en/Guide/Introduction

The Board We Choose Is Arduino Uno

- http://arduino.cc/en/Main/ArduinoBoardUno





The Board We Choose Is Arduino Uno

Name	Processor	Operating Voltage/Input Voltage	CPU Speed	Analog In/Out	Digital IO/PWM	EEPROM [KB]	SRAM [KB]	Flash [KB]	USB	UART
Uno	ATmega328	5 V/7-12 V	16 Mhz	6/0	14/6	1	2	32	Regular	1
Due	AT91SAM3X8E	3.3 V/7-12 V	84 Mhz	12/2	54/12	-	96	512	2 Micro	4
Leonardo	ATmega32u4	5 V/7-12 V	16 Mhz	12/0	20/7	1	2.5	32	Micro	1
Mega 2560	ATmega2560	5 V/7-12 V	16 Mhz	16/0	54/15	4	8	256	Regular	4
Mega ADK	ATmega2560	5 V/7-12 V	16 Mhz	16/0	54/15	4	8	256	Regular	4
Micro	ATmega32u4	5 V/7-12 V	16	12/0	20/7	1	2.5	32	Micro	1

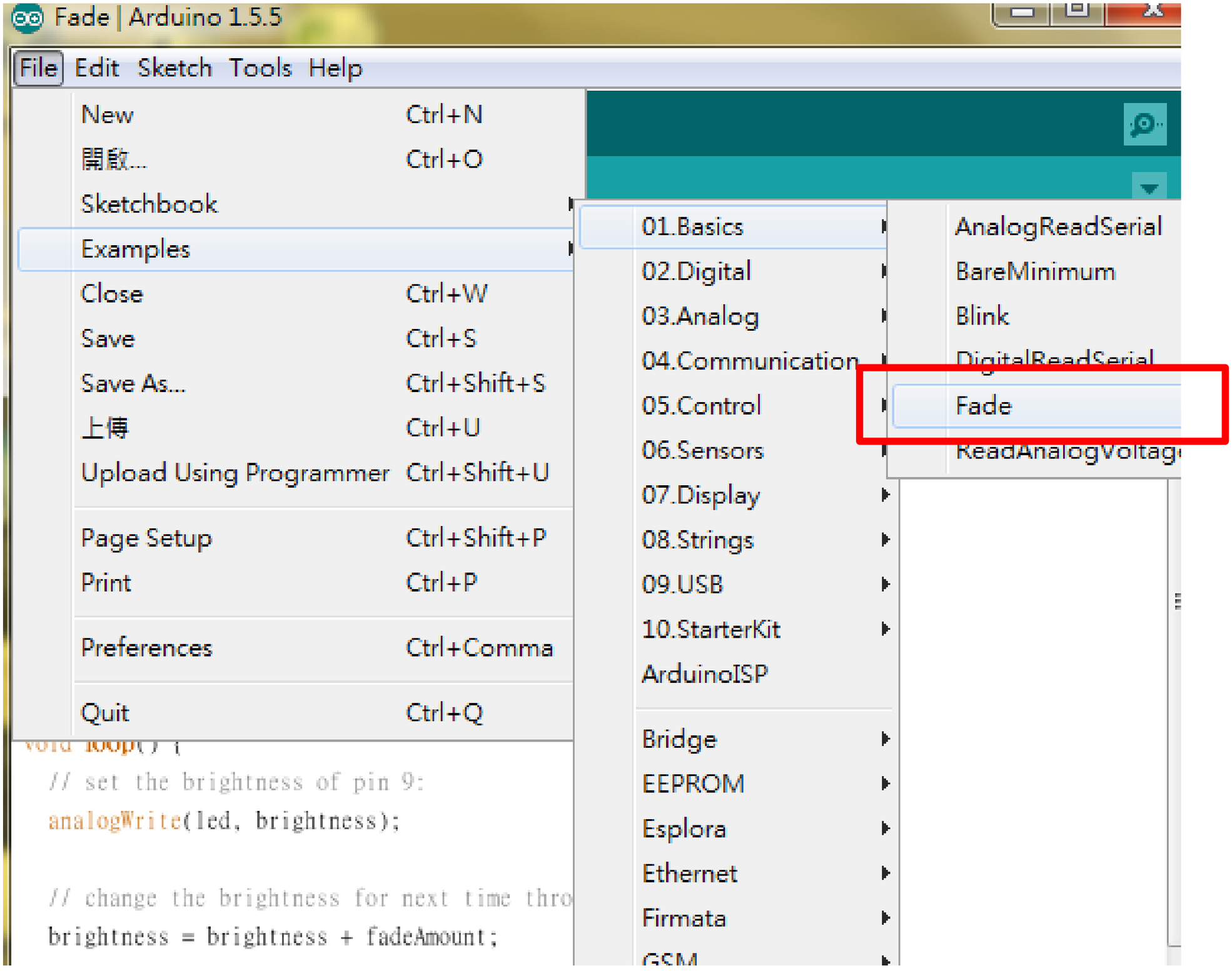
- DC current per I/O pin: 40 mA
-0.5 KB Flash memory used by bootloader
-6 analog inputs (A0 to A5), each of which provide 10 bits of resolution (i.e. 1024 different values) from ground to 5 volts by default (http://arduino.cc/en/Reference/AnalogRead)
-It is possible to change the upper end of their range using the AREF pin and the analogReference() function

ADC (input range and number of bits)

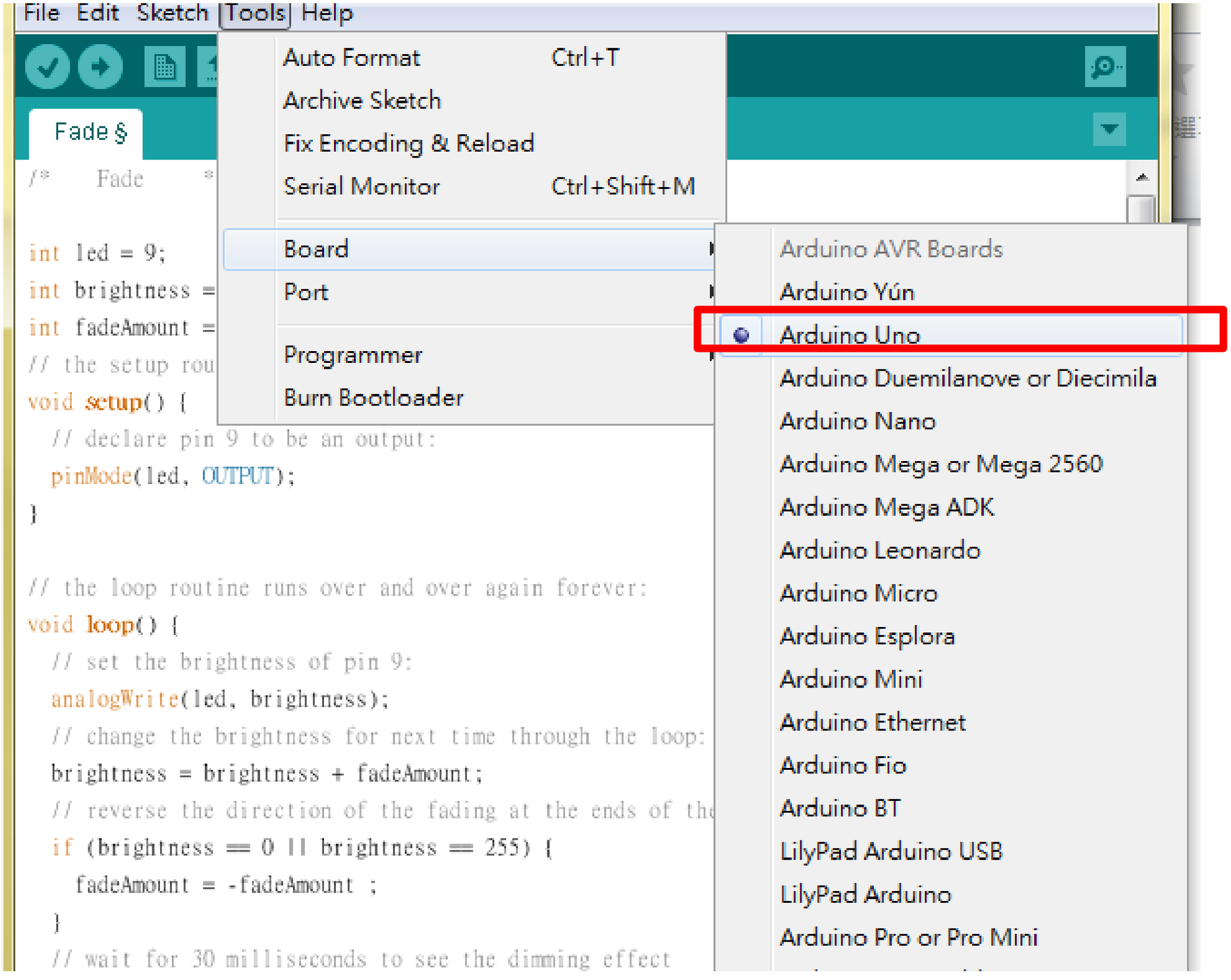
Arduino IDE Software

- Download the Arduino software
  http://arduino.cc/en/Main/Software
- Getting started with Arduino
  http://arduino.cc/en/Guide/Windows

Arduino IDE: Open the Example

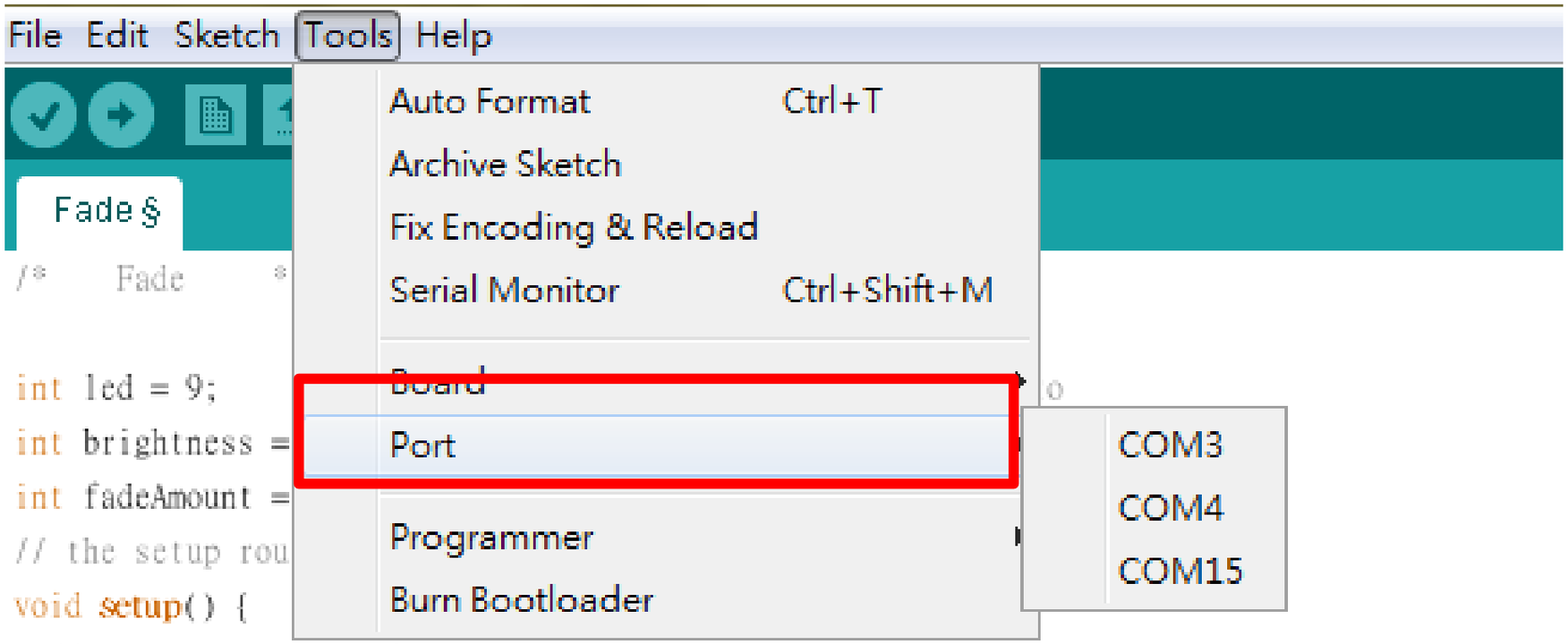


Arduino IDE: Select Your Board



Arduino IDE: Select Your Serial Port

- Check your Windows device manager



Arduino IDE: Write Your Own Codes

- http://arduino.cc/en/Reference/HomePage

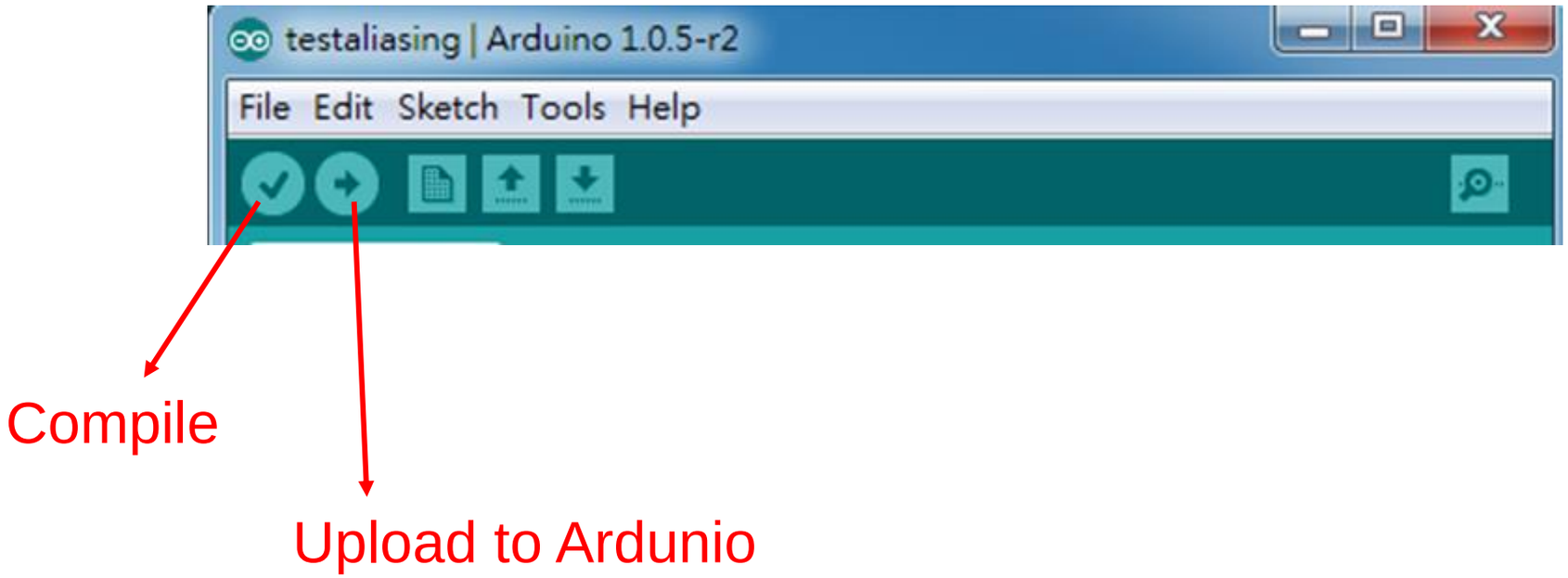
```
/*    Fade    */

int led = 9;           // the pin that the LED is attached to
int brightness = 0;    // how bright the LED is
int fadeAmount = 5;    // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);
  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;
  // reverse the direction of the fading at the ends of the fade:
  if (brightness == 0 || brightness == 255) {
    fadeAmount = -fadeAmount ;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```

Arduino IDE: Compile and Upload



Arduino IDE: Serial Communication with PC

- http://arduino.cc/en/Reference/Serial

- Serial.begin()
- Serial.end()

- Serial.read()
- Serial.write()

- Serial.print()
- Serial.println()

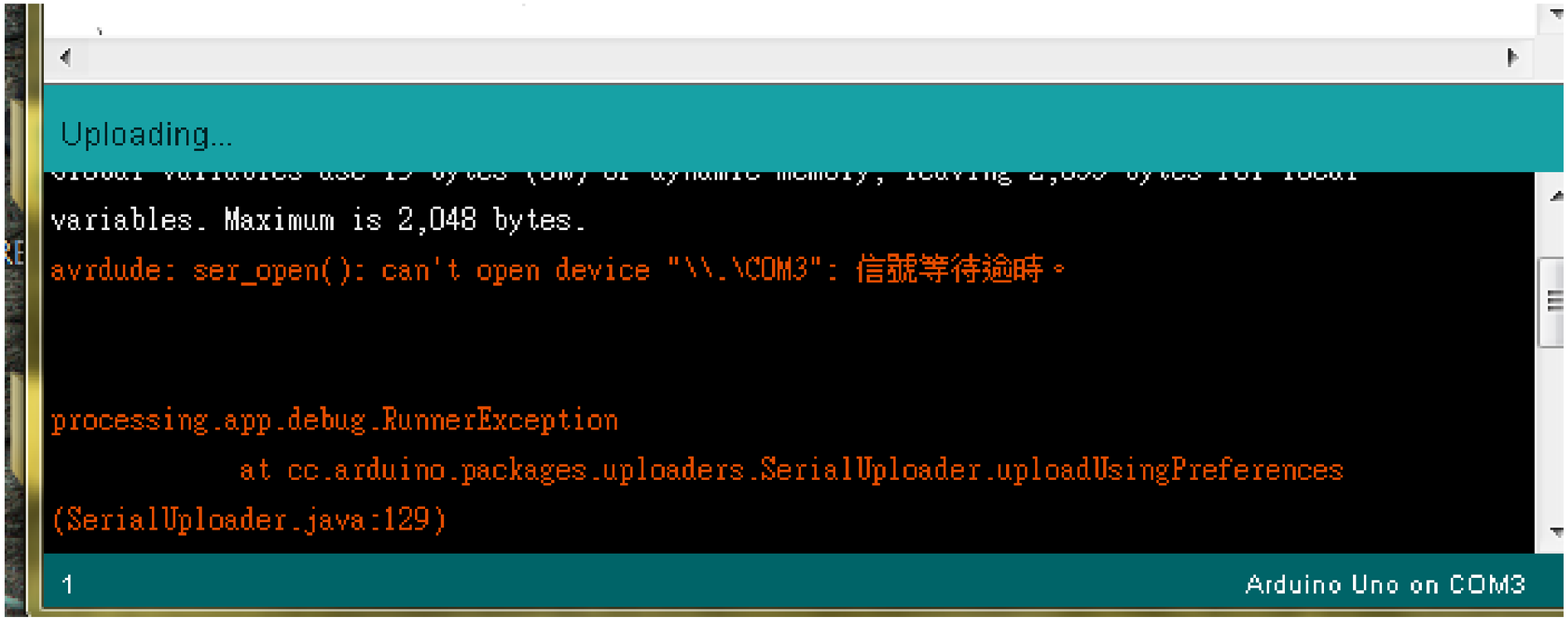
```
// the setup routine runs once when you p
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
  Serial.begin(115200) ;
}

// the loop routine runs over and over ag
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);
  Serial.println(brightness)
  // change the brightness for next time
  brightness = brightness + fadeAmount;
```

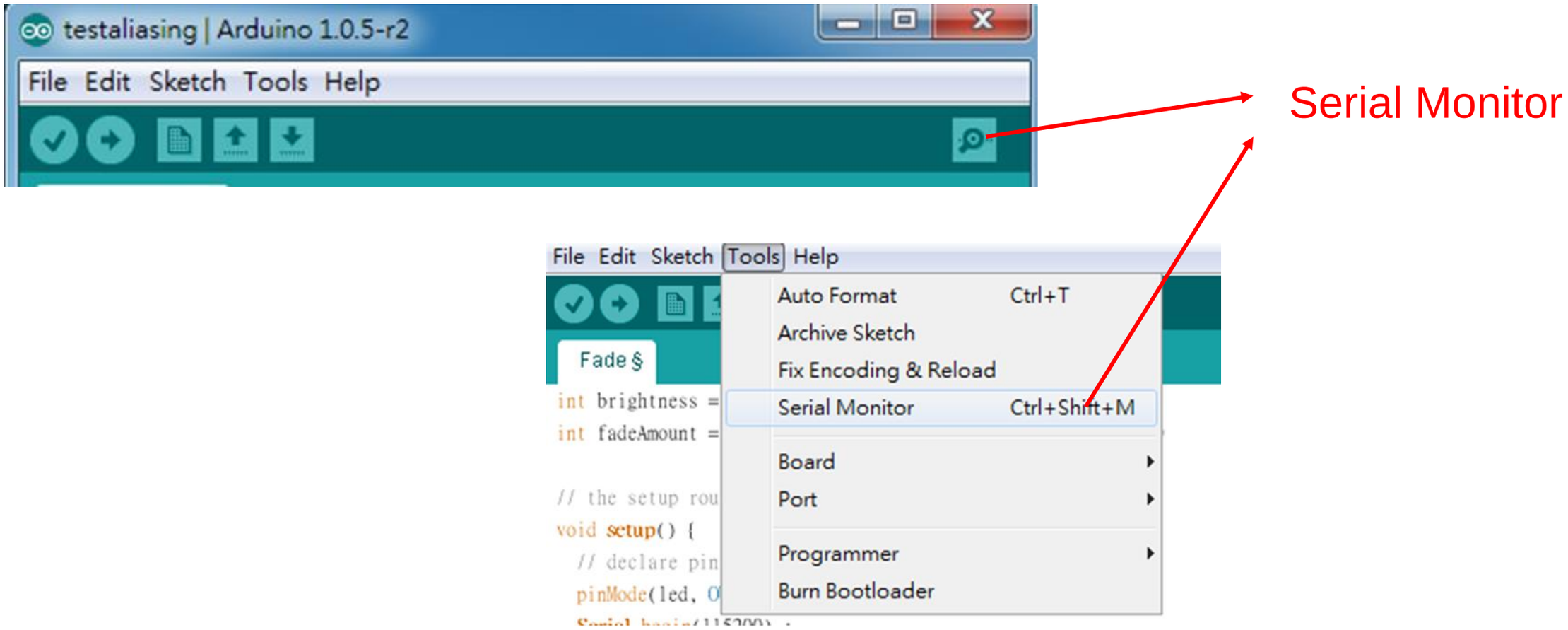
Baud rate/data rate: bps

Arduino IDE: Common Error

- Choose the Wrong COM



Arduino IDE: Serial Monitor



Arduino IDE: Serial Monitor

Lab 1 Today

0. Pickup Your Arduino Uno and Materials of ECG circuitry

- Arduino Uno、傳輸線、麵包板、單芯線、 LED、電阻、可變電阻、按鍵、蜂鳴器、七段顯示器、電晶體、伺服馬達等。

- ECG Circuitry：
  Resistor：10M Ω *10、10K Ω *8、2K Ω *3、220 Ω *1、30 Ω *1

  Capacitor：0.22uF*4、0.047uF(+-20%)*4、0.047uF(+80%-20%)*1、330nF*2、3.3nF*2

  ICs：LM317T、LM324N

1. Try to Be Familiar with Arduino: Training Manual

www.PlayRobot.com

Arduino 輕鬆上手18堂課　目錄

飛颯機器人 普特企業有限公司 www.PlayRobot.com

1. Try to Be Familiar with Arduino: Training Manual

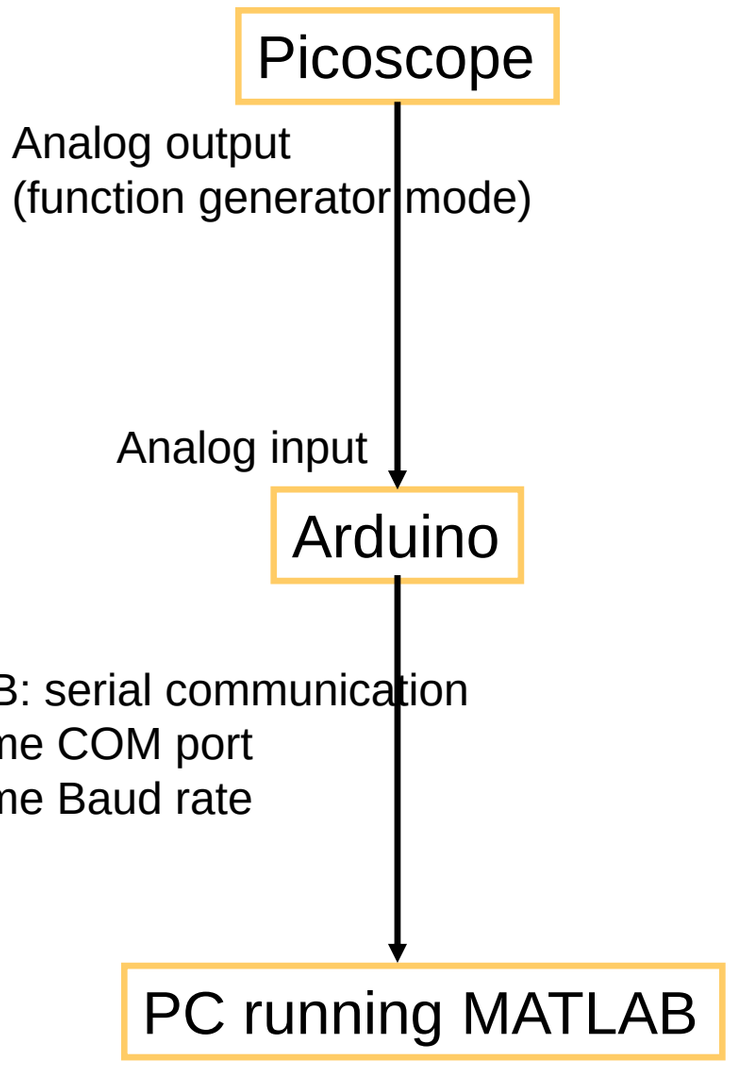
- Reference: http://coopermaa2nd.blogspot.tw/2011/05/arduino.html
- No demo is required.
- Please simply describe the exp content, the problems you met and how you solve the problems in your report

2. Sampling and Aliasing

- Connect the function generator (Picoscope, see PICOSCOPE.pdf) to the ADC of the Arduino board. Set sine wave as the ADC input waveform with frequency of 100 Hz. Use the following frequencies (500 Hz, 200 Hz, 100 Hz, 80 Hz) to sample the input waveform.
  - Plot the waveforms with the provided Matlab codes.
  - Use FFT to show the spectra of the saved samples.
  - Discuss about the aliasing issues.
- Demo: 500 Hz, and 80 Hz

2. Sampling and Aliasing – Experimental Setup

Setup



Function

1. **Function generator** – generate signals, e.g., sine wave
2. **Oscilloscope** – check the generated signal

1. **ADC** – determine sampling rate Fs (see Arduino_samplecode.txt), sampling the signal from the Picoscope

2. **Data transfer to PC** – Serial.println(val,DEC); in Arduino_samplecode.txt and need to setup COM port and Baud rate for serial port communication ()

1. **Receive data from Arduino** – data=fscanf(s1);in matlab_samplecode.m, and need to setup COM port and Baud rate for serial port communication
2. **Waveform display** –
default display buffer: 1000 points (see display_buffer=nan(1,display_length); in matlab_samplecode.m)
NSample in matlab_samplecode.m: sample the signal NSample times, and display and refresh the plot NSample times

2. Sampling and Aliasing: Setup ADC Sampling Frequency

- Take fs=200Hz for example

testaliasing | Arduino 1.0.5-r2

File Edit Sketch Tools Help

testaliasing §

```
  //Serial.begin(9600);
  //Serial.begin(115200);
  Serial.begin(115200);
}

// the loop routine runs over and over again forever:
void loop() {
 start_times = micros();
 ...
 int value = analogRead(A0);
 ...
 time= micros();
 while(time-start_times<5000){
 time= micros();
 }
}
```

Sampling(every loop)

Fix 5000us(200Hz) for a loop

See the provided Arduino_samplecode.txt

2. Sampling and Aliasing: Receive Data via Serial Port

ECGr... × test.m ×

```
1 -     serialobj=instrfind;          %clear all serial port
2 -     if ~isempty(serialobj)
3 -         delete(serialobj)
4 -     end
5 -     clc;clear all;close all;
6
7 -     s1 = serial('COM8');          %define serial port
8 -     s1.BaudRate=9600;             %define baud rate
9 -     fopen(s1);
10 -    data=fscanf(s1);              %read data(string type)
11 -    fclose(s1);                   % close the serial port
12
```

Delete serial port object to prevent the problem of “serial port in use”，

Should be the same as the setting in Arduino IDE

http://www.matlabarduino.org/serial-communication.html

See the provided matlab_samplecode.m

2. Sampling and Aliasing: Real Time Plotting

```
% Initialize figure object
figure
h_plot = plot(nan,nan);
hold off

tic
for i = 1:NSample
    data = fscanf(s1); % Read from Arduino
    data = str2double(data);
    disp(data);

    % Add data to display buffer
    if i <= display_length
        display_buffer(i) = data;
    else
        display_buffer = [display_buffer(2:end) data]; % first in first out
    end

    % Update figure plot
    set(h_plot, 'xdata', time_axis, 'ydata', display_buffer)
    title('test');
    xlabel('Time');
    ylabel('Quantized value');
    drawnow;
end
toc

% Disconnect the serial port object from the serial port
fclose(s1);
```

You can change it to a while loop

Display butter

Updating the plot buffer

Drawing the curve

See the provided matlab_samplecode.m

3. Quantization

- Use 10 bits, 8 bits, 5 bits, and 3 bits to sample the oscillator waveform.
  - Plot the waveforms with the provided Matlab codes.
  - Use FFT to show the spectra of the saved samples.
  - Please discuss what you explore
- Demo: 10 bits and 3 bits
- Hint: You can try map()

More for Your Report …

- Please check the EECLASS elearning system
  - Week 1: Introduction to Arduino Platform

References

- Processor (ATmega328)的data sheet
  - http://www.atmel.com/Images/doc8161.pdf
- Detailed introduction of ATmega328 ADC
  - http://www.microsmart.co.za/technical/2014/03/01/advanced-arduino-adc/
  - http://www.openmusiclabs.com/learning/digital/atmega-adc/
- Advanced topic: how to modify ATmega328 ADC reference voltage and ADC sampling rate
  - http://coopermaa2nd.blogspot.tw/2011/07/7-adc.html
- AnalogRead used ADC spec.
  - http://arduino.cc/en/Reference/AnalogRead
  - http://arduino.cc/en/Reference/AnalogReference
- Color code of Resistor
  - http://zh.wikipedia.org/wiki/%E9%9B%BB%E9%98%BB%E8%89%B2%E7%A2%BC