

Unit 1: Biomedical Application – ECG Signal Acquisition and Processing

What Will You Learn in Unit 1?

- 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

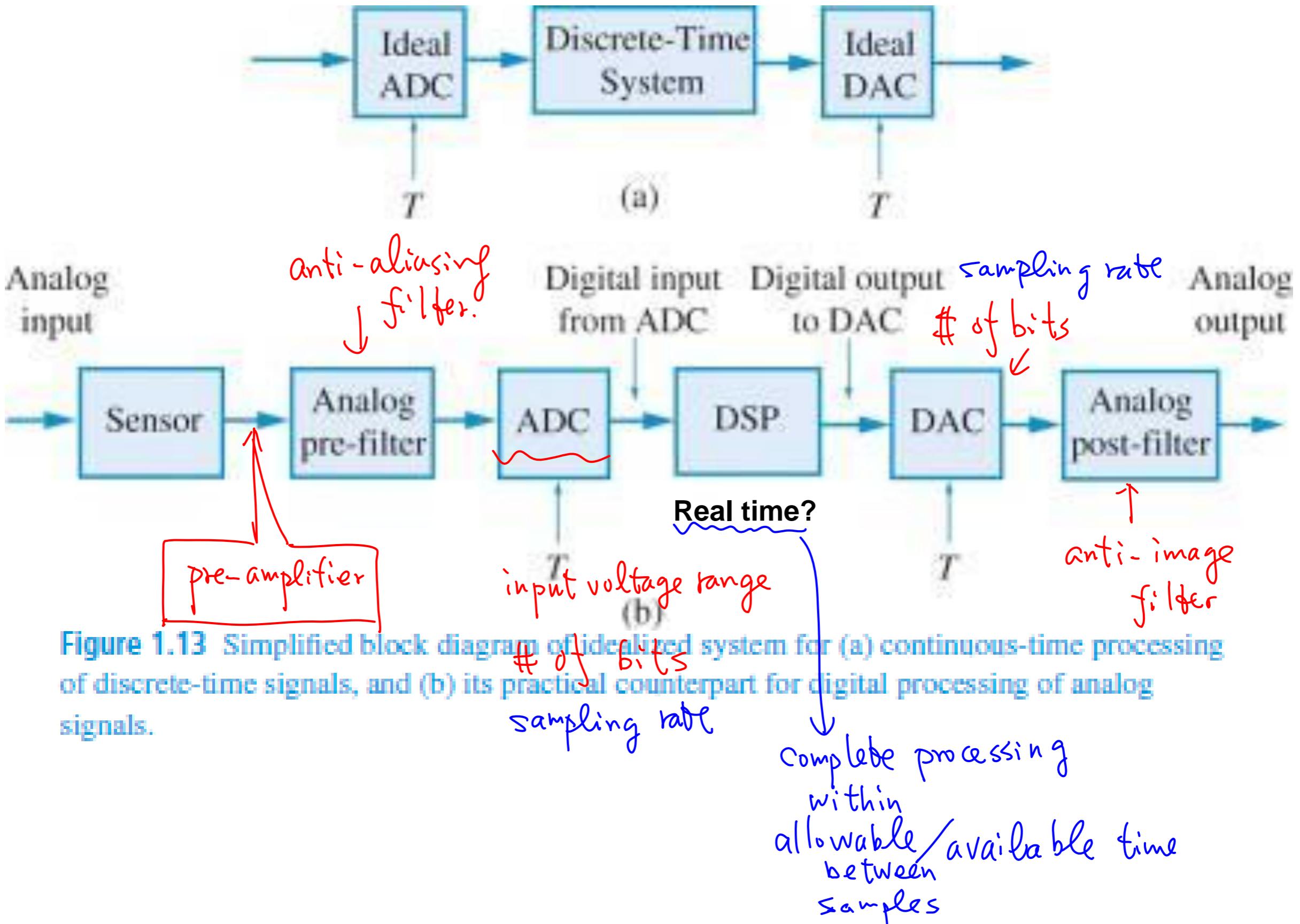
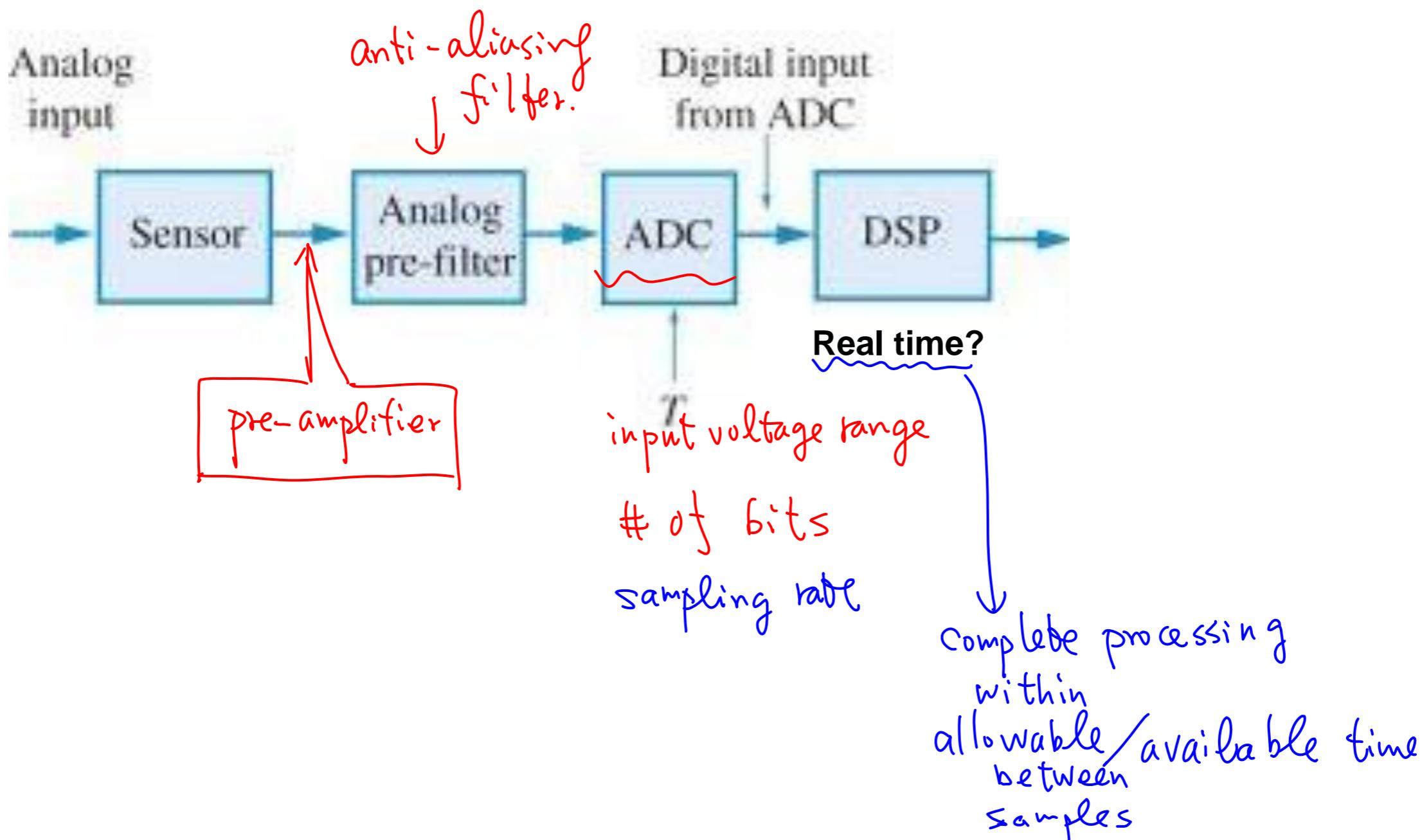
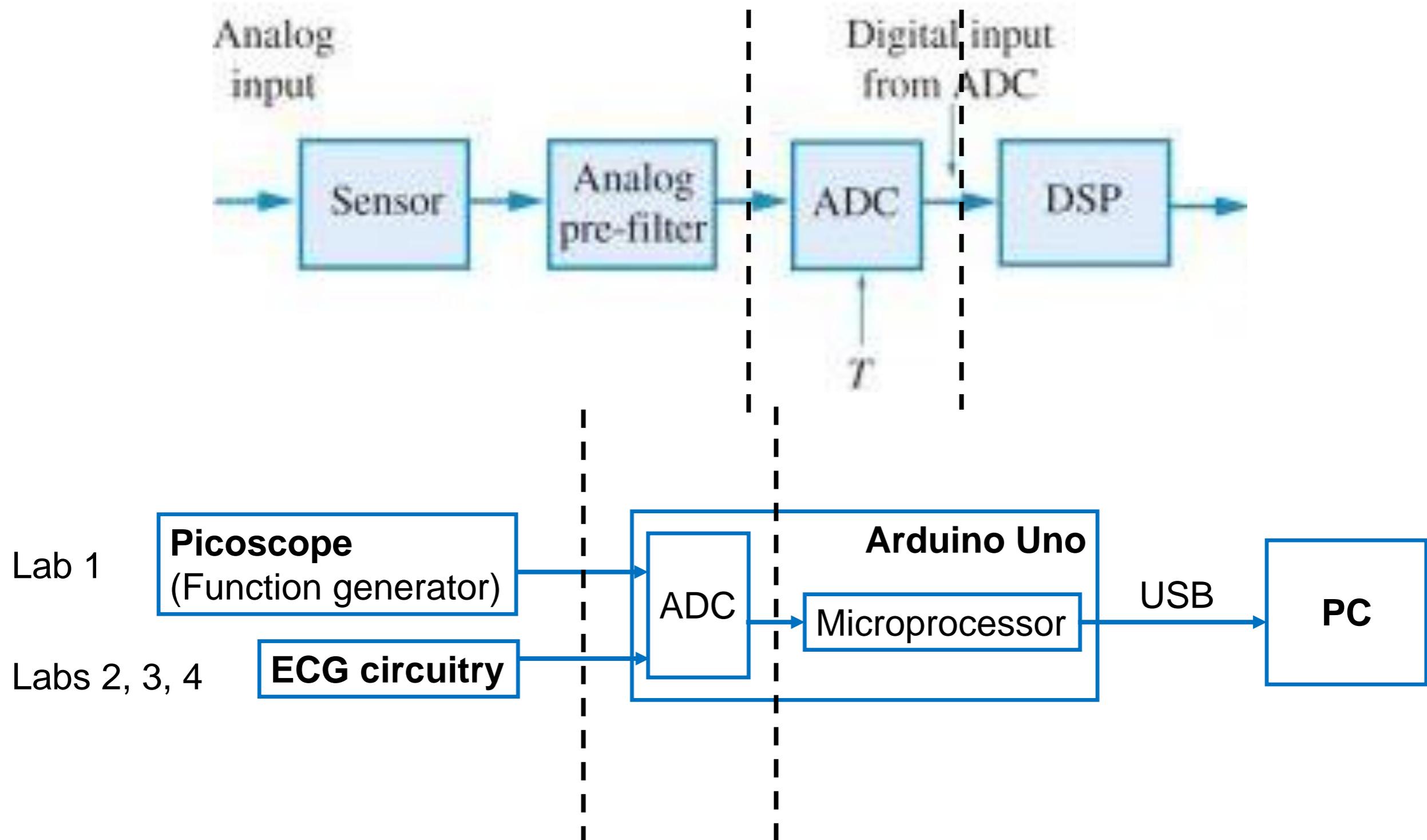


Figure 1.13 Simplified block diagram of idealized system for (a) continuous-time processing of discrete-time signals, and (b) its practical counterpart for digital processing of analog signals.

DSP Flow in Unit 1



Unit 1 DSP Flow vs. Unit 1 Setup



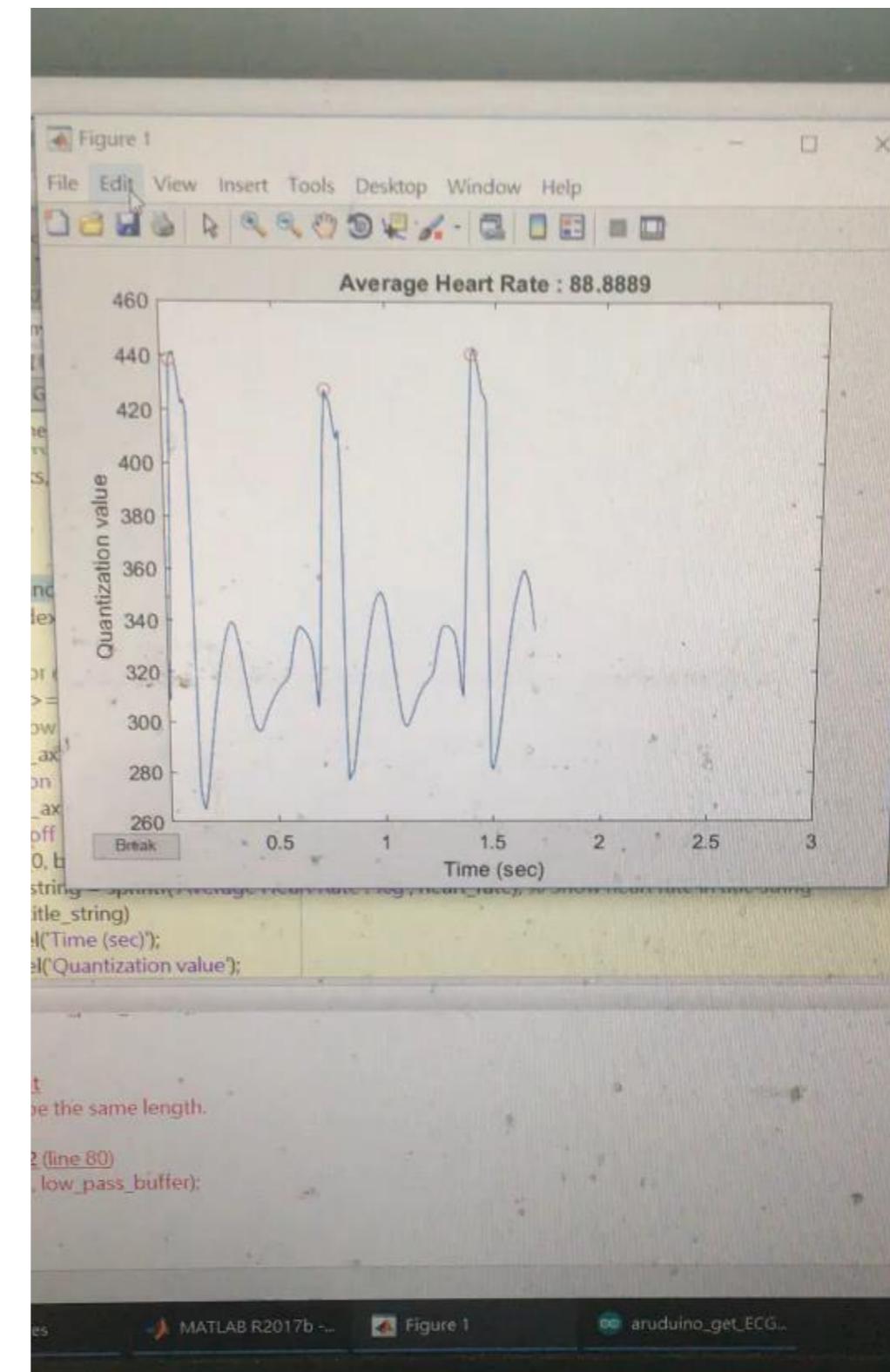
09/27: Lab 1 Introduction to Arduino Platform

10/04: Lab 2 ECG circuits, Signal Sampling and Digitization

10/18: Lab 3 Pre-processing of ECG Signals

10/25: Lab 4 Heart Rate Estimation

Goal: Real Time ECG Display and Heart Rate Estimation



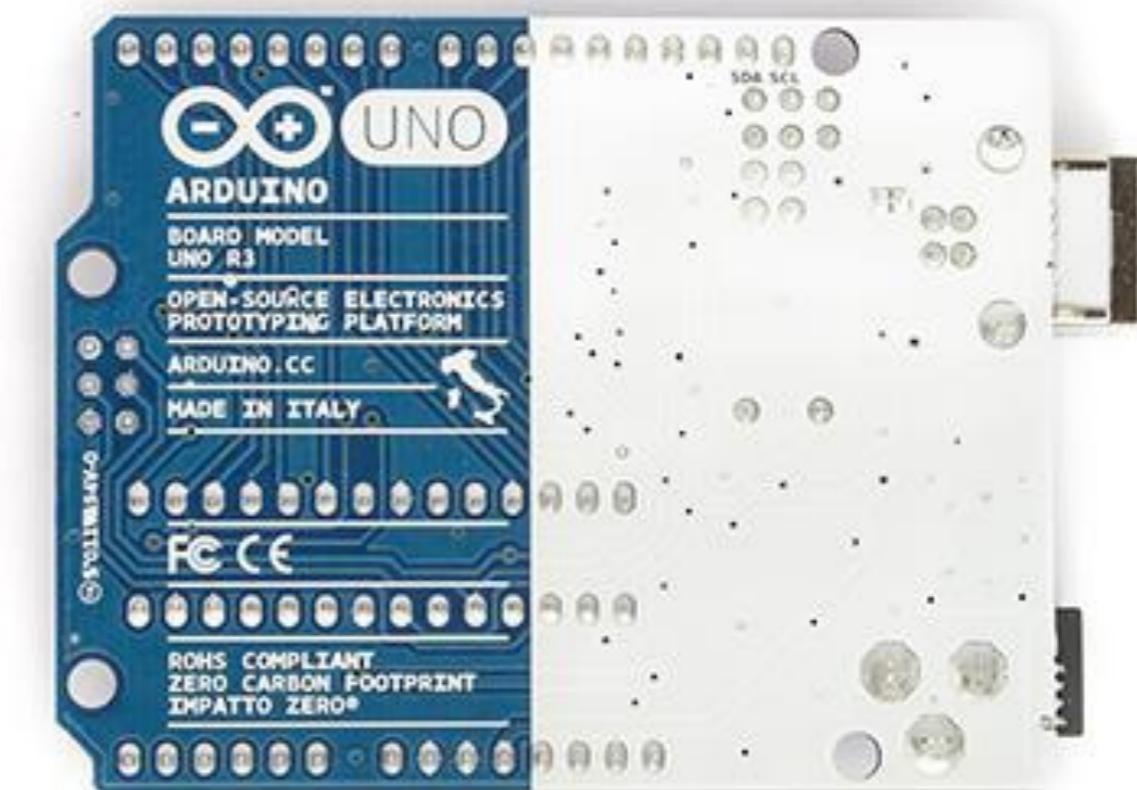
Lab 1: Introduction to Arduino Platform

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

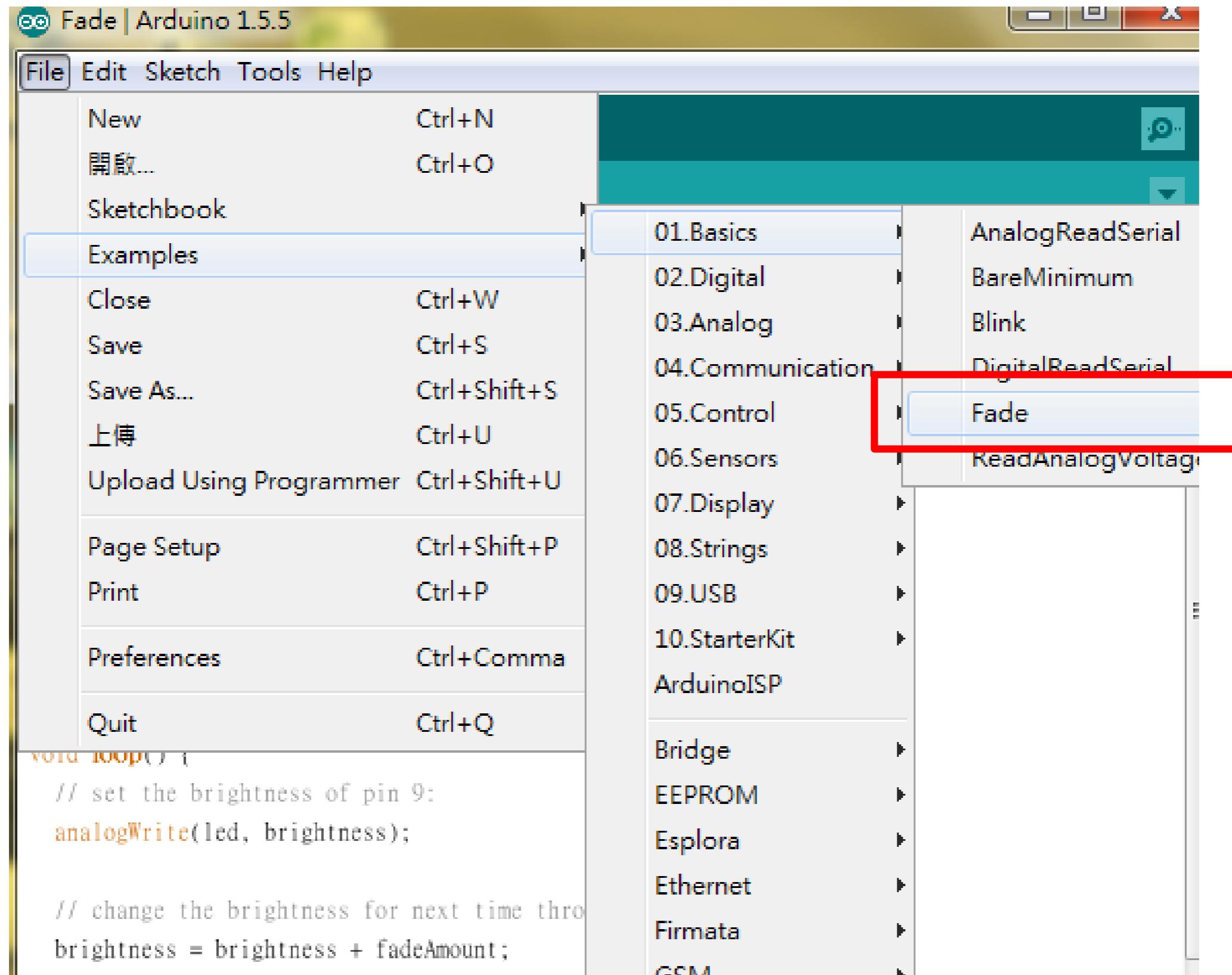
- Download the Arduino software

<http://arduino.cc/en/Main/Software>

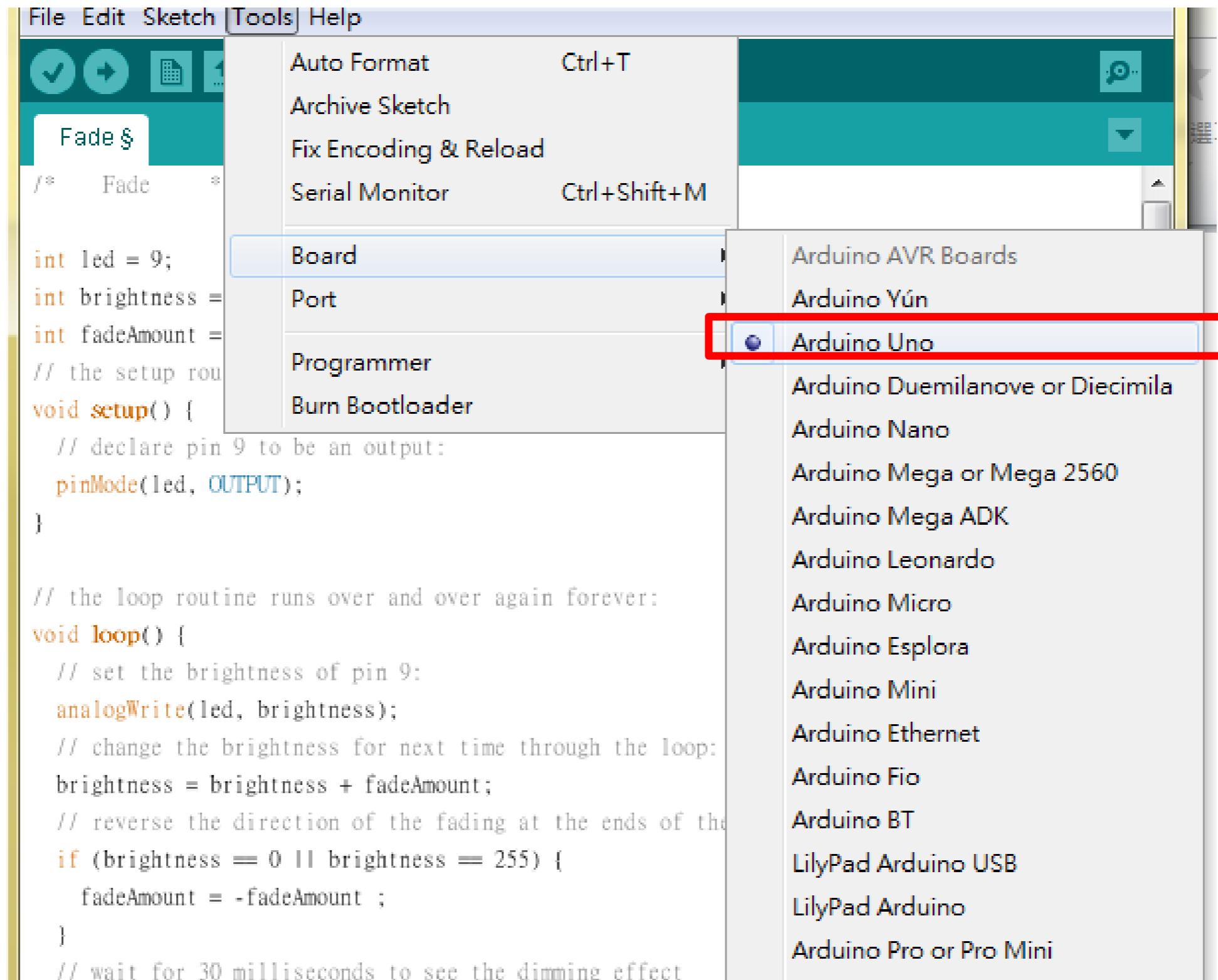
- Getting started with Arduino

<http://arduino.cc/en/Guide/Windows>

Arduino IDE: Open the Example

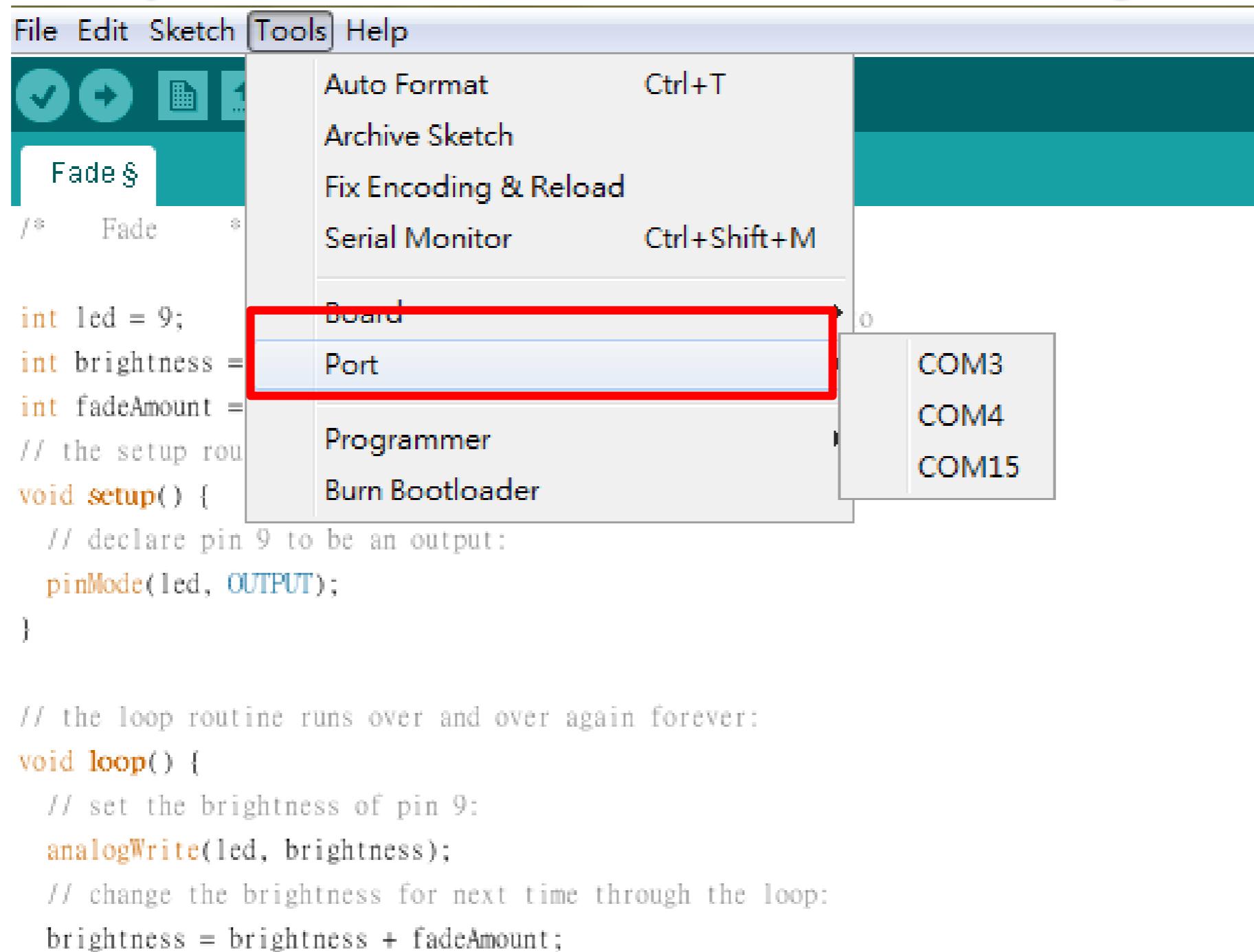


Arduino IDE: Select Your Board



Arduino IDE: Select Your Serial Port

- Check your Windows device manager



Arduino IDE: Write Your Own Codes

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

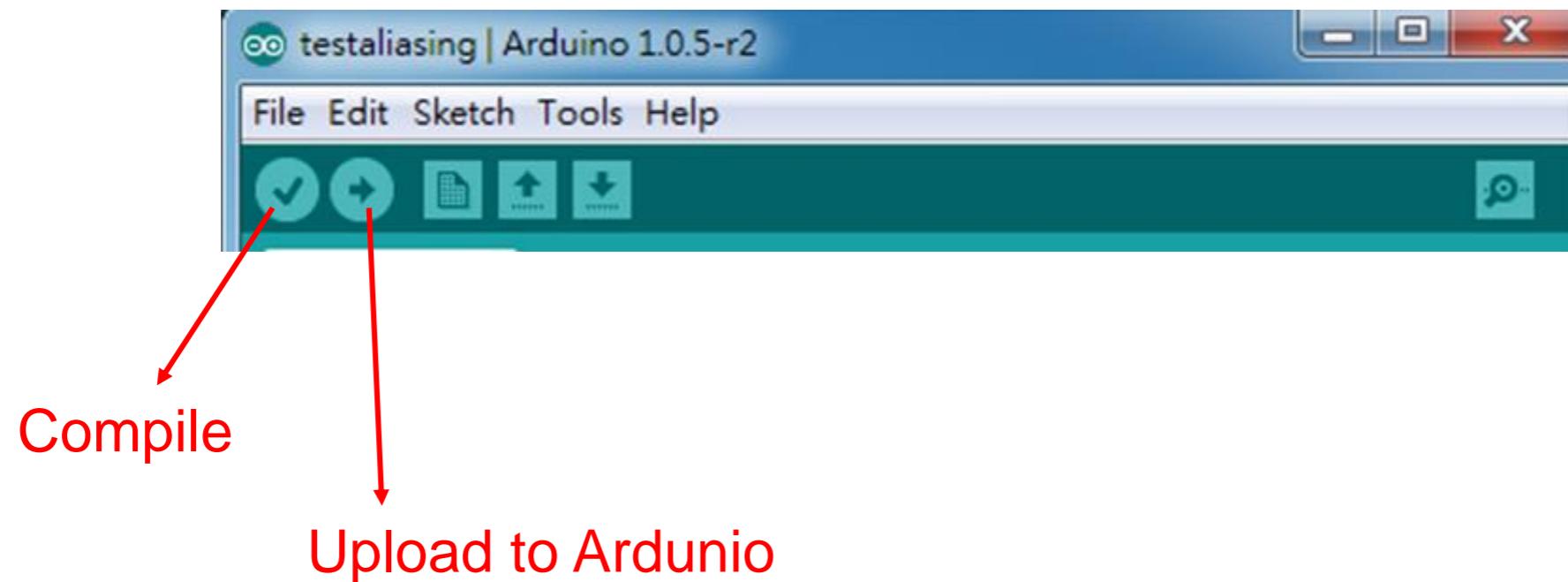
```
/* Fade */

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>

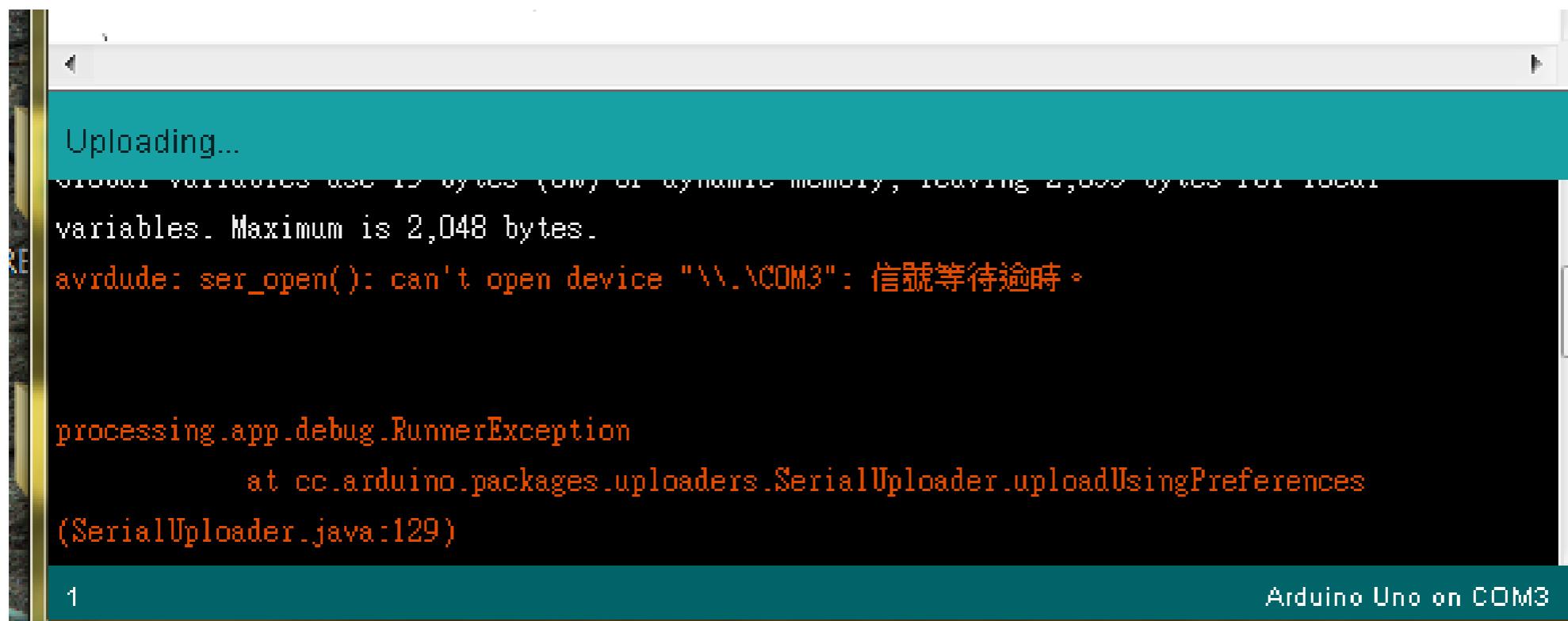
- Serial.begin()
- Serial.end()
- Serial.read()
- Serial.write()
- Serial.print()
- Serial.println()

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

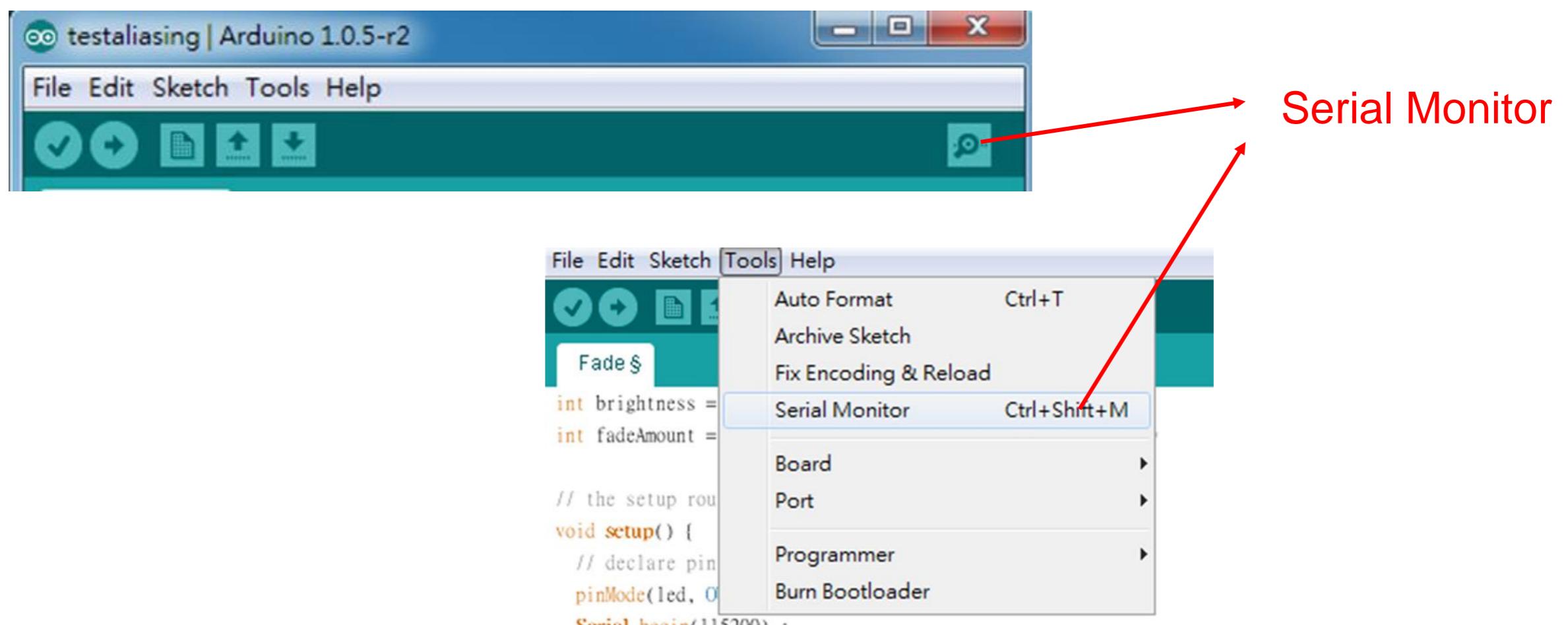
// the loop routine runs over and over ag
void loop() {
    // set the brightness of pin 9:
    analogWrite(led, brightness);
    Serial.println(brightness)
    // change the brightness for next time
    brightness = brightness + fadeAmount;
```

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. Finish the Lab Till Class 7 of Arduino

Training Manual

www.PlayRobot.com

Arduino 輕鬆上手18堂課 目錄



Class 1 :	認識Arduino	Page- 2
Class 2 :	Arduino程式語言	Page- 17
Class 3 :	Arduino程式控制元件	Page- 34
Class 4 :	LED元件的運用	Page- 49
Class 5 :	數位輸入-按鍵的運用	Page- 65
Class 6 :	數位與類比	Page- 79
Class 7 :	可變電阻的運用	Page- 85
Class 8 :	七段顯示器的運用	Page- 91
Class 9 :	蜂鳴器的運用	Page- 97
Class 10 :	綜合練習一：使用按鍵玩音樂	Page-106
Class 11 :	光敏電阻的運用	Page-108
Class 12 :	伺服馬達的運用	Page-116
Class 13 :	電晶體的運用	Page-125
Class 14 :	數位電位計	Page-129
Class 15 :	IR 紅外線感測器	Page-133
Class 16 :	綜合練習二：使用光敏電阻使馬達追光	Page-148
Class 17 :	Processing & Arduino	Page-149
Class 18 :	Scratch& Arduino	Page-156
課外補充	各式感測器	Page-173

版權為飄機器人所有，僅供教師使用本教材之教學使用，不得拷貝翻印或從事任何形式之商業行為與用途！

1. Finish the Lab Till Class 7 of 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

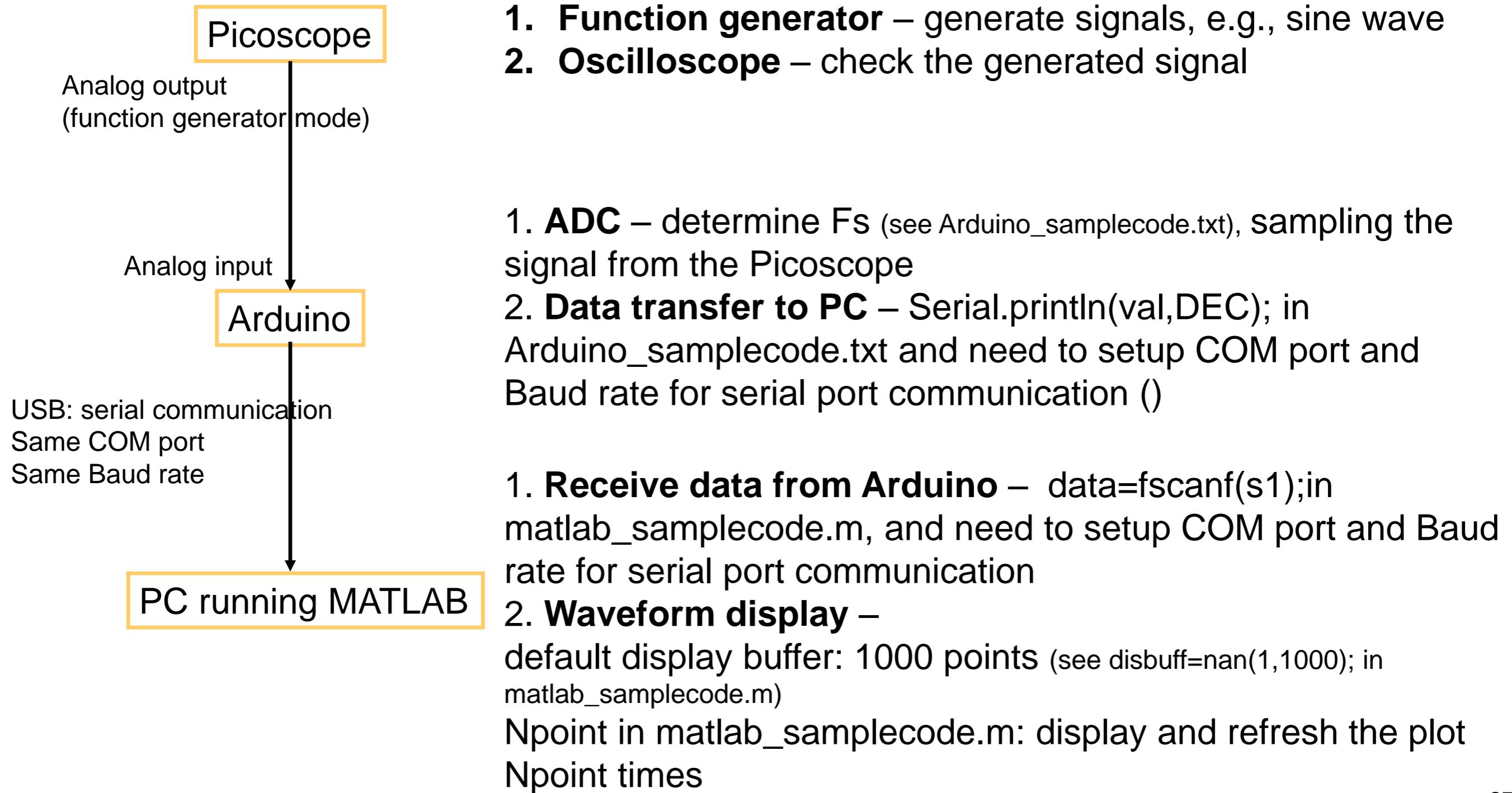
2. Sampling and Aliasing

- 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



2. Sampling and Aliasing: Setup ADC Sampling Frequency

- Take $f_s=200\text{Hz}$ 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); ← Sampling(every loop)
    ...
    time= micros();
    while(time-start_times<5000){ ← Fix 5000us(200Hz)
        time= micros();
    }
}
```

See the provided Arduino_samplecode.txt

2. Sampling and Aliasing: Receive Data via Serial Port

```
ECGr... x test.m x
1 -     serialobj=instrfind;           %clear all serial port
2 -     if ~isempty(serialobj)
3 -         delete(serialobj)          } Delete serial port object to
4 -     end                           prevent the problem of "serial
5 -     clc;clear all;close all;       port in use",
6
7 -     s1 = serial('COM8');           %define serial port
8 -     s1.BaudRate=9600;              %define baud rate } Should be the same as the
9 -     fopen(s1);                   %read data(string type)
10 -    data=fscanf(s1);             % close the serial port
11 -    fclose(s1);
12
```

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

See the provided matlab_samplecode.m

2. Sampling and Aliasing: Real Time Plotting

Editor - C:\Users\Alonso\Dropbox\master\secondgrade\dsp助教\testcode\test.m

ECGr... x test.m x

```
13 - figure
14 - h_plot=plot(nan,nan);
15 - hold off
16 -
17 - tic
18 - for i= 1:N_point
19 -     data=fscanf(s1);          %read data(string type)
20 -     y(i) = str2double(data);%convert string to double type
21 -     if i<=1000
22 -         disbuff(i)=y(i);
23 -     else
24 -         disbuff=[disbuff(2:end) y(i)];
25 -     end
26 -     if i>1
27 -         set(h_plot, 'xdata',time, 'ydata',disbuff)
28 -         title('ECG');
29 -         xlabel('Time');
30 -         ylabel('Quantization value');
31 -         drawnow;
32 -     end
33 -
34 -     fclose(s1);              % close the serial port
35 -
```

You can change it to a while loop

1000- point display buffer

Updating the plot buffer

Drawing the curve

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%B2%C0>