

Smart Software Project

Lab: Week 1
Lab Introduction &
Arduino Overview

Prof. HyungJune Lee
hyungjune.lee@ewha.ac.kr



이화여자대학교
EWHA WOMANS UNIVERSITY

Today

- What is this lab session for?
 - Lab policy
- Arduino overview
 - Platform
 - Programming environment
- Announcement



Class Schedule

Week	Lecture Contents	Lab Contents
Week 1	Course introduction	Arduino introduction: platform & programming environment
Week 2	Embedded system overview & source management in collaborative repository (using GitHub)	Lab 1: Arduino Mega 2560 board & SmartCAR platform
Week 3	ATmega2560 Micro-controller (MCU): architecture & I/O ports, Analog vs. Digital, Pulse Width Modulation	Lab 2: SmartCAR LED control
Week 4	Analog vs. Digital & Pulse Width Modulation	Lab 3: SmartCAR motor control (Due: HW on creating project repository using GitHub)
Week 5	ATmega2560 MCU: memory, I/O ports, UART	Lab 4: SmartCAR control via Android Bluetooth
Week 6	ATmega2560 UART control & Bluetooth communication between Arduino platform and Android device	Lab 5: SmartCAR control through your own customized Android app (Due: Project proposal)
Week 7	Midterm exam	
Week 8	ATmega2560 Timer, Interrupts & Ultrasonic sensors	Lab 6: SmartCAR ultrasonic sensing
Week 9	Infrared sensors & Buzzer	Lab 7: SmartCAR infrared sensing
Week 10	Acquiring location information from Android device & line tracing	Lab 8: Implementation of line tracer
Week 11	Gyroscope, accelerometer, and compass sensors	Lab 9: Using gyroscope, accelerometer, and compass sensors
Week 12	Project	Team meeting (for progress check)
Week 13	Project	Team meeting (for progress check)
Week 14	Course wrap-up & next steps	
Week 15	Project presentation & demo I (Due: source code, presentation slides, & poster slide)	Project presentation & demo II
Week 16	Final week (no final exam)	



Today

- **What is this lab session for?**
 - Lab policy
- Arduino overview
 - Platform
 - Programming environment
- Announcement



Lab Session

- Practice in-lab programming exercises based on the lecture materials
- Upload source codes for lab assignments in Ewha Cyber Campus after the lab session
 - Due: 11:59pm on the lab day
- Once you are done, you can leave the session after checking with me or TA
- Or, continue to work on programming for other homework assignments



Lab Policy

- 1) Please check out your SmartCAR (& Nexus 7 tablet) as soon as you arrive at the classroom
- 2) Please complete lab assignments
- 3) Upload required files to Ewha Cyber Campus
- 4) Check with me or TA
- 5) Please **remove files that you created** in your computer after you are done
- 6) Please **shut down your computer** before you leave
- 7) Return the checked-out SmartCAR (& Nexus 7 tablet) to TA



Today

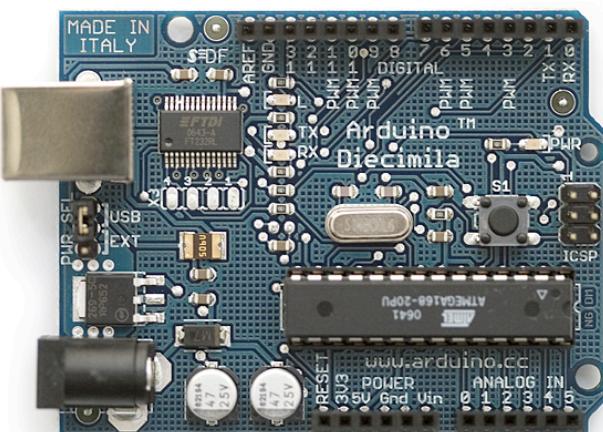
- What is this lab session for?
 - Lab policy
- **Arduino overview**
 - Platform
 - Programming environment
- Announcement



What is Arduino?

The word “Arduino” can mean 3 things

A physical piece
of hardware



A programming
environment

Arduino - 0010 Alpha

Blink §

```
* The basic Arduino example. Turns on an LED on for one second,
* then off for one second, and so on... We use pin 13 because,
* depending on your Arduino board, it has either a built-in LED
* or a built-in resistor so that you need only an LED.
*
* http://www.arduino.cc/en/Tutorial/Blink
*/
int ledPin = 13; // LED connected to digital pin 13

void setup() // run once, when the sketch starts
{
  pinMode(ledPin, OUTPUT); // sets the digital pin as output
}

void loop() // run over and over again
{
  digitalWrite(ledPin, HIGH); // sets the LED on
  delay(1000); // waits for a second
  digitalWrite(ledPin, LOW); // sets the LED off
  delay(1000); // waits for a second
}
```

18

A community
& philosophy

Arduino playground

Arduino playground

:: About the Arduino Playground ::

Welcome to the Arduino Playground, a wild where all the users of [Arduino](#) can contribute and benefit from their collective research.

This is the place to post and share your own code, circuit diagrams, tutorials, DIY instructions, tips and tricks, and after all the hard work, to show off your projects!

Arduino Playground is a **work in progress**. We can use all the help you can give, so please read the [Participate](#) section and get your fingers typing!

:: RoadMap: What Needs to be Done? ::

There is a lot to do...most of the pages are just stubs, simple placeholders waiting for you to fill them up. However here is a small roadmap of things I personally think should be developed first (again, this is a wiki so you are more than welcome to get in

Search: Go

Edit | History | Attachments | About Arduino playground

Projects Built with Arduino

Manuals

Arduino Tutorials

- Official tutorial page
- Information on the Arduino Mini
- Bluetooth tutorial
- Xbee (Zigbee) tutorial
- DIY Arduino Shield
- Serial LCD tutorial
- Learning Electronics
- Learning Programming
- Arduino and Arduino
- Let the Arduino Sleep
- Burning Bootloaders
- CourseWare
- Stand alone 3.3V 8 MHz
- Programming the Bluetooth WT11
- Retrofitting Diecimila AutoReset
- Bare Bones "Diecimila" Upgrade

Run Arduino on...

- Linux
- Ubuntu Linux
- Debian Linux
- Gentoo Linux
- Slackware Linux
- FreeBSD
- Fedora Linux
- The command line

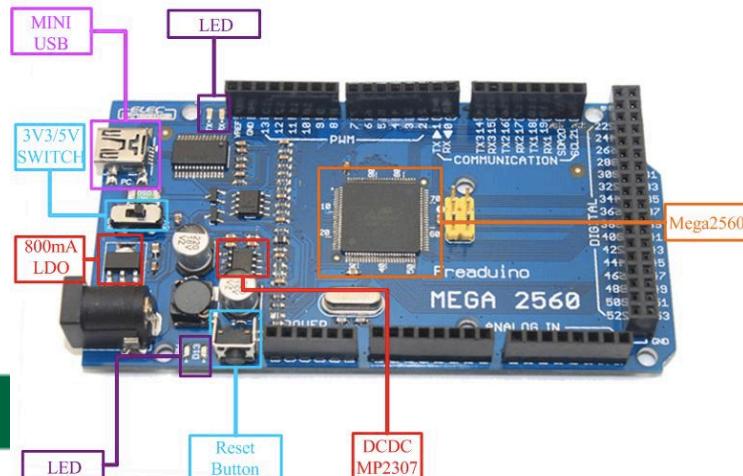
Interface Arduino with...

- Instant Arduino (X3D)
- Flash
- Processing
- PD (Pure Data)



Arduino Overview

- Started in 2005 as a project for students
- Founded by Massimo Banzi
- Open-source electronics prototyping platform
 - Open source hardware
 - Assembling/reconfiguring hardware units by hand
- Simple and cost effective development environment (starting from \$30)
- Easy to upload programs into the microcontroller memory



Arduino Philosophy & Community

- Open Source Physical Computing Platform
 - “Open Source Hardware”
 - Open source: free to inspect & modify
- Community-built
 - Examples wiki (the “playground”) editable by anyone
 - <http://playground.arduino.cc/>
 - Forums with lots of helpful people
 - <http://forum.arduino.cc/>



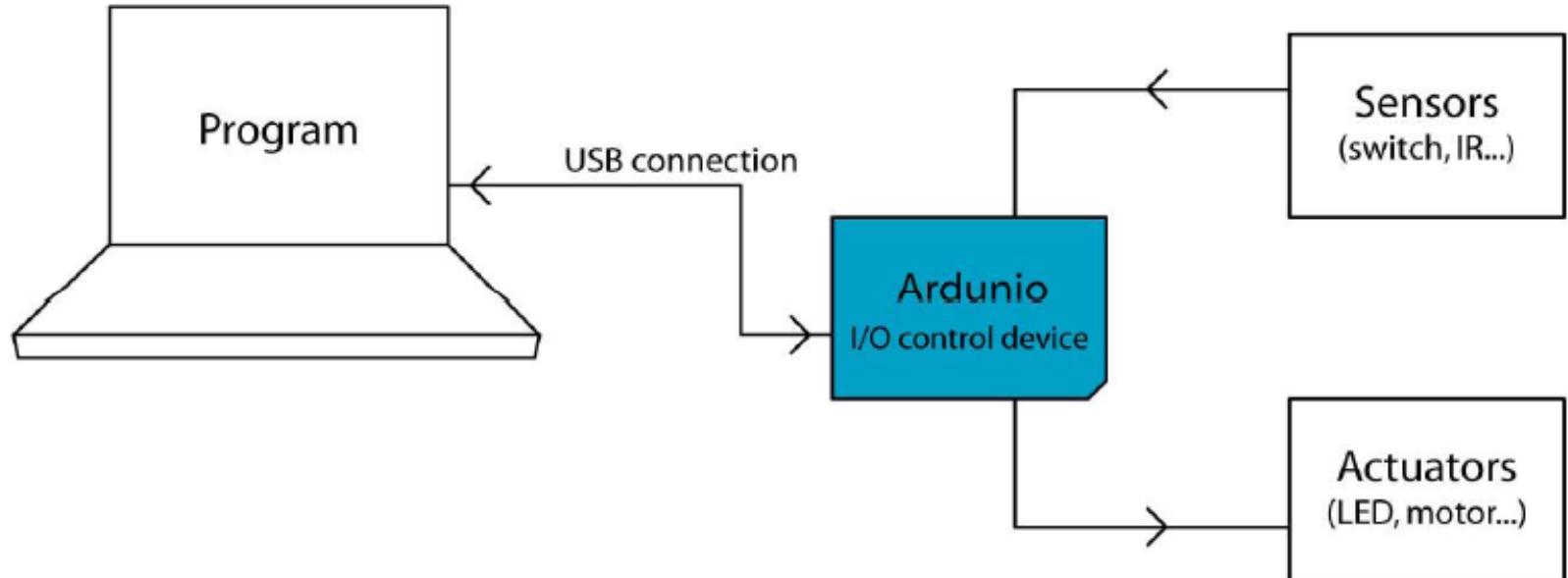
Arduino Terminology

- “**sketch**” – a program you write to run on an Arduino board
- “**pin**” – an input or output connected to something
 - Output to an LED
 - Input from temperature sensor
- “**digital**” – value is either HIGH or LOW
 - a.k.a. on/off, 1/0
- “**analog**” – value ranges, usually from 0 ~ 255
 - LED brightness, motor speed, etc.



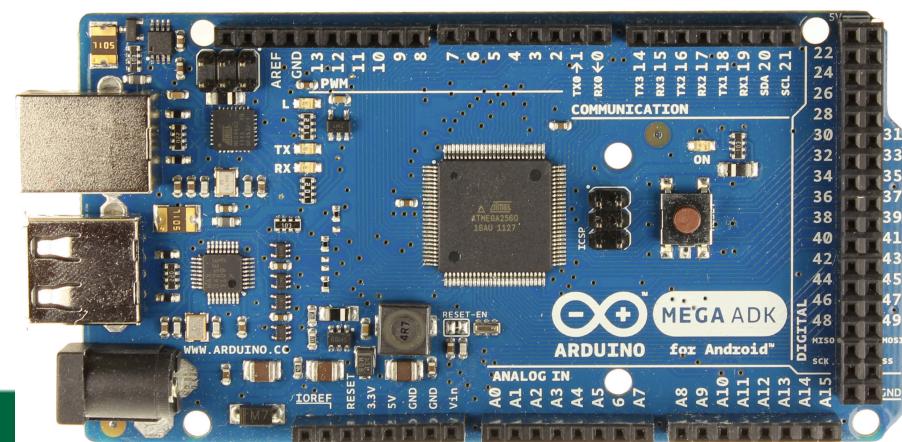
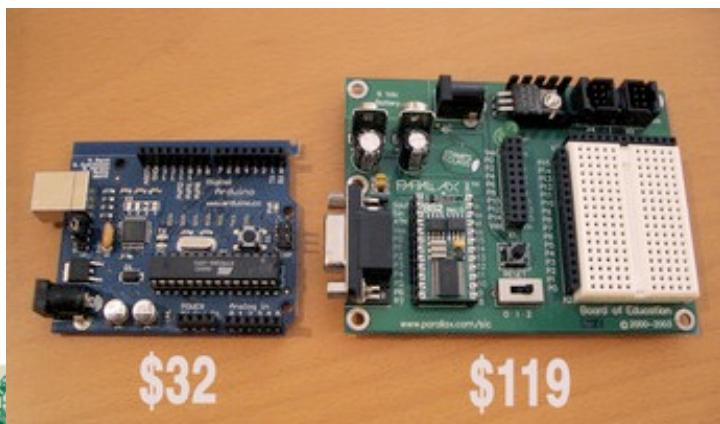
Arduino Platform

- Arduino Platform Basic Configuration



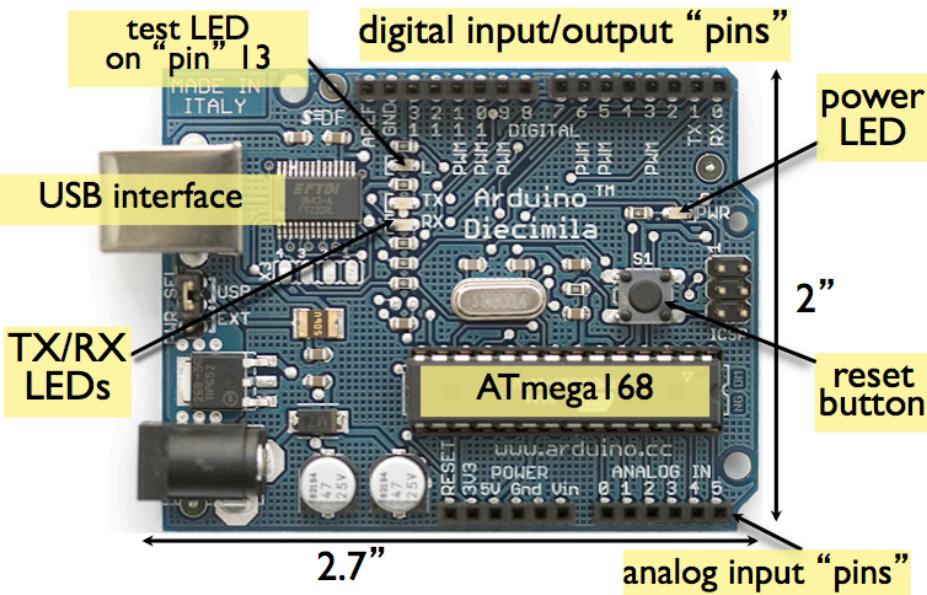
Arduino Hardware

- Cheaper, faster, and open
- Uses AVR Atmega 8-bit RISC-based microcontroller chip
 - What is RISC?
 - This chip is designed to be used with C language



Arduino Mega 2560: ~ \$55

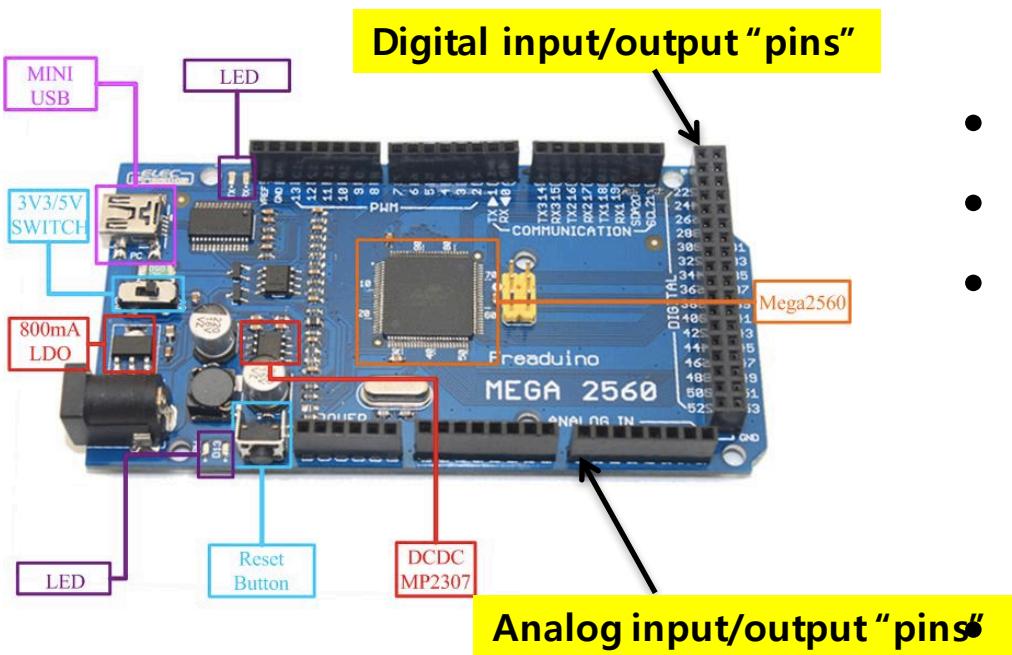
Arduino Diecimila Board



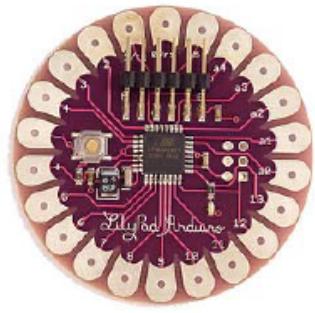
- 16 kBytes of Flash program memory
- 1 kByte of RAM
- 16 MHz (Apple II: 1 MHz)
- Inputs and Outputs
 - 13 digital input/output pins
 - 5 analog input pins
 - 6 analog output pins
- Completely stand-alone: does not need a computer once programmed



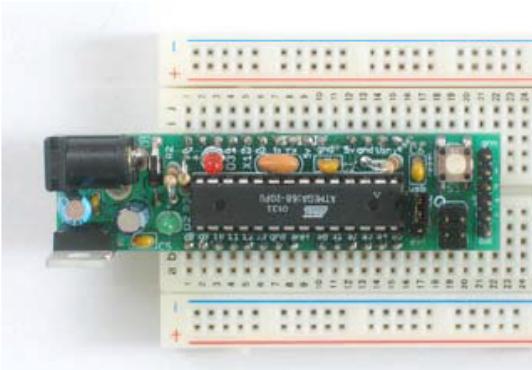
Arduino Mega 2560 Board



Arduino Hardware Variety



Lilypad Arduino



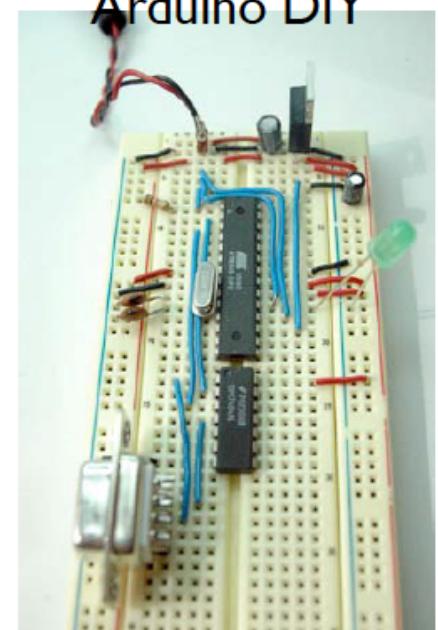
Breadboard Arduino



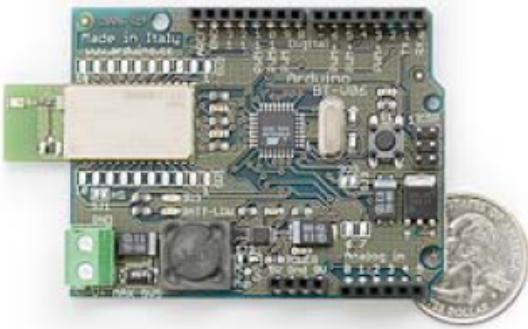
Arduino Mini



Arduino Nano



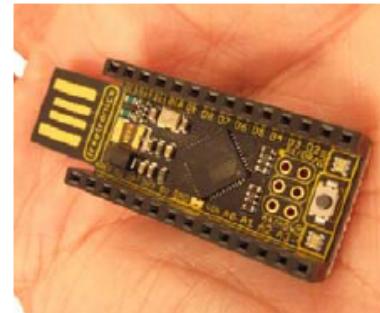
Arduino DIY



Arduino Bluetooth



Arduino Ethernet

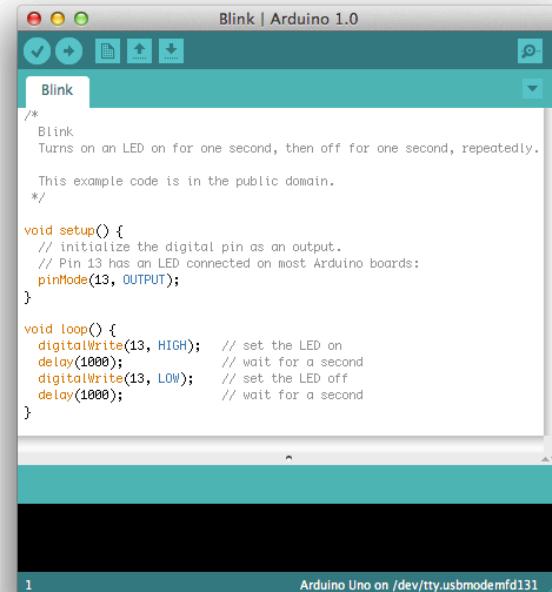


Arduino Notebook



Programming Environments

- Arduino IDE
 - Windows
 - Mac OS X
 - Linux



The screenshot shows the Arduino IDE interface with the title bar "Blink | Arduino 1.0". The code editor contains the "Blink" example sketch. The code is as follows:

```
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.
*/
void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

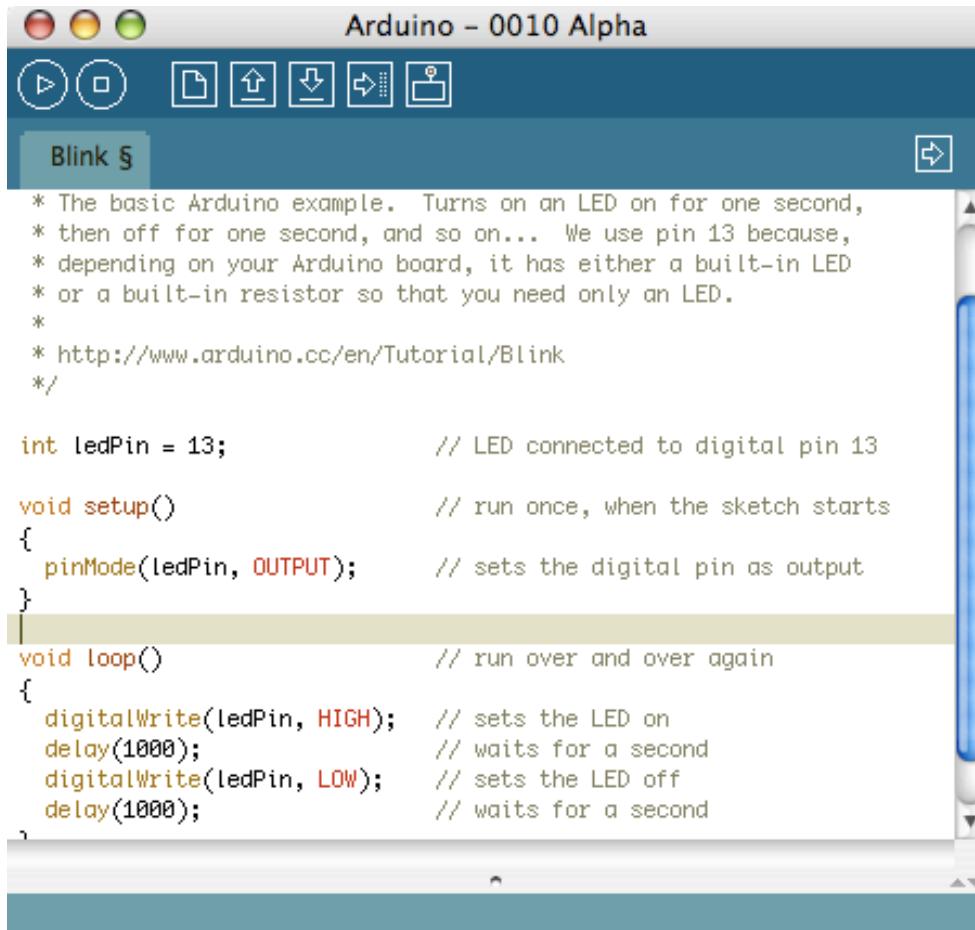
void loop() {
  digitalWrite(13, HIGH);    // set the LED on
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);     // set the LED off
  delay(1000);              // wait for a second
}
```

The status bar at the bottom right indicates "Arduino Uno on /dev/tty.usbmodemfd131".

- Embedded programming
 - Arduino Mega 2560-based SmartCAR Robot platform
 - Nexus 7 Android tablet



Arduino Software



The screenshot shows the Arduino IDE interface with the title bar "Arduino - 0010 Alpha". The toolbar contains icons for upload, download, and serial communication. The code editor window displays the "Blink" sketch. The code is as follows:

```
* The basic Arduino example. Turns on an LED on for one second,
* then off for one second, and so on... We use pin 13 because,
* depending on your Arduino board, it has either a built-in LED
* or a built-in resistor so that you need only an LED.
*/
* http://www.arduino.cc/en/Tutorial/Blink
*/
int ledPin = 13; // LED connected to digital pin 13

void setup() // run once, when the sketch starts
{
  pinMode(ledPin, OUTPUT); // sets the digital pin as output
}

void loop() // run over and over again
{
  digitalWrite(ledPin, HIGH); // sets the LED on
  delay(1000); // waits for a second
  digitalWrite(ledPin, LOW); // sets the LED off
  delay(1000);
}
```

- Like a text editor
- View/write/edit “sketches”
- Then you program them into hardware



Standard Library

- Main Functions
 - setup()
 - Used for one-time function calls
 - Initializing pins/serial ports
 - Takes and returns no arguments
 - loop()
 - Continuous loop function (similar to while(1))
 - Takes and returns no arguments



Standard Library

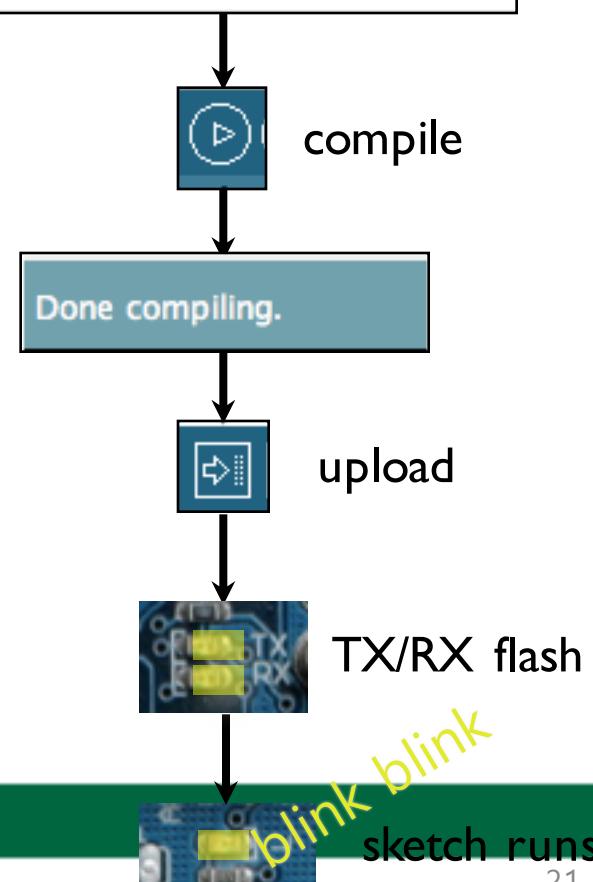
- I/O Functions
 - Set pin mode as input or output
 - `pinMode(pin, value)` – set a pin as INPUT or OUTPUT (in value)
 - Analog I/O
 - `analogRead(pin)` – read an analog pin through ADC
 - `analogWrite(pin, value)` – write an “analog” value using PWM
 - Digital I/O
 - `digitalRead(pin)` – read a digital pin’s state
 - `digitalWrite(pin, value)` – set a digital pin to HIGH or LOW
 - Timing
 - `delay(time)` – wait an amount of time in millisecond
 - `millis()` – get the number of milliseconds after beginning running the current program
 - Serial Communication
 - Begin, read, write, print, etc.



Using Arduino

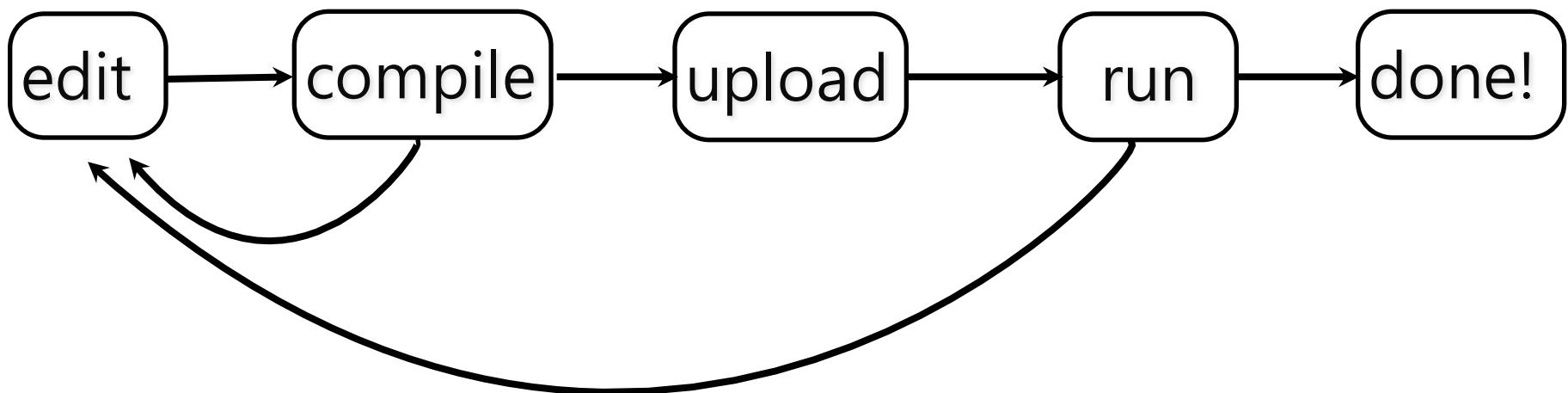
- Write your sketch
- Press Compile button
(to check for errors)
- Press Upload button to
program Arduino board
with your sketch

```
void setup() {  
    pinMode(ledPin, OUTPUT);      // sets t  
}  
void loop() {  
    digitalWrite(ledPin, HIGH);   // sets t  
    delay(1000);                // waits  
    digitalWrite(ledPin, LOW);    // sets t  
    delay(1000);                // waits  
}
```



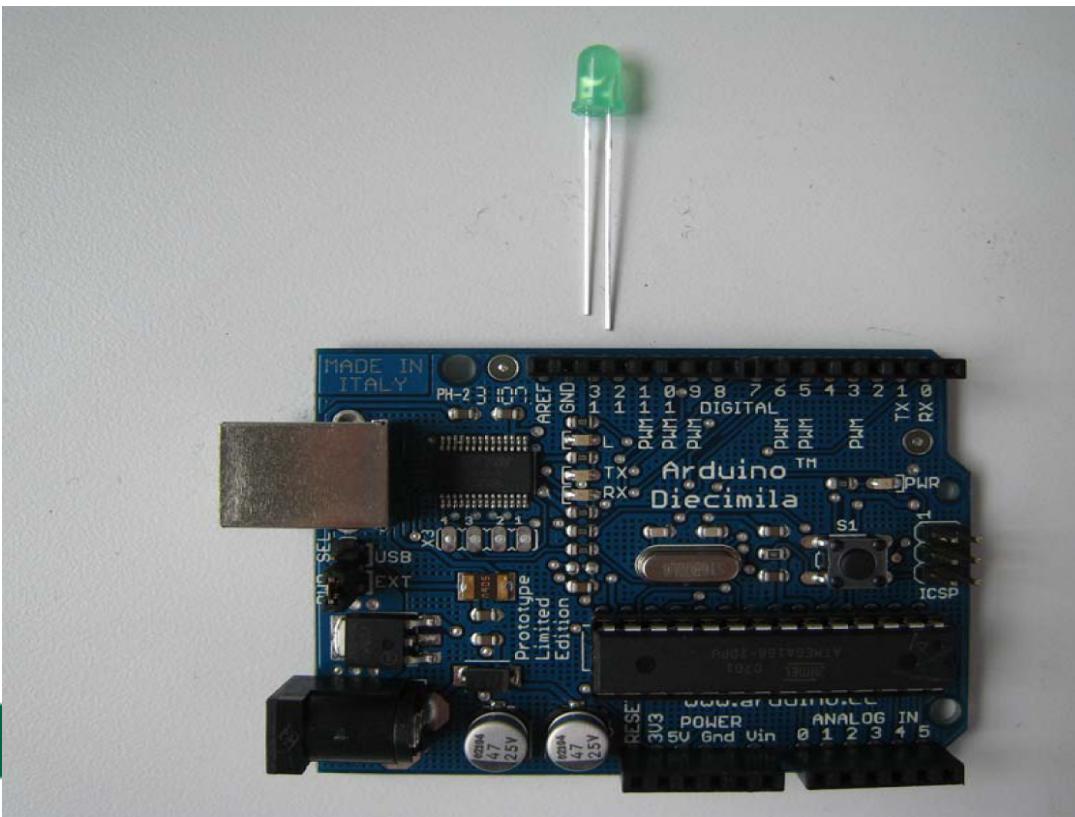
Development Cycle

- Make as many changes as you want
- edit → compile → upload → run



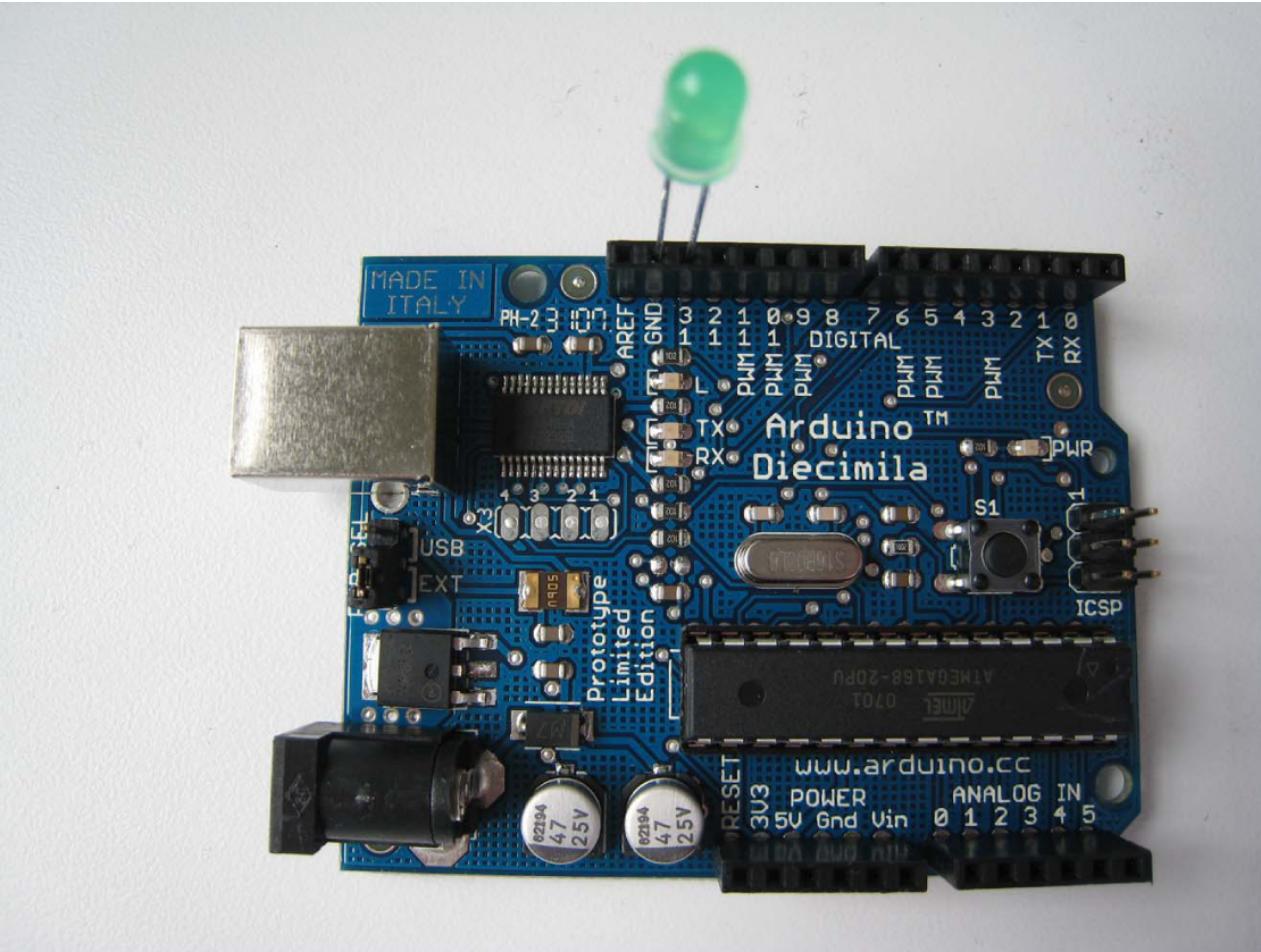
Exercise: Digital Output LED

- Will use pin 13
- On LEDs, polarity matters
 - Shorter lead is “negative” side, goes to ground



Exercise: Digital Output LED

- 1) Connect LED to Arduino Board



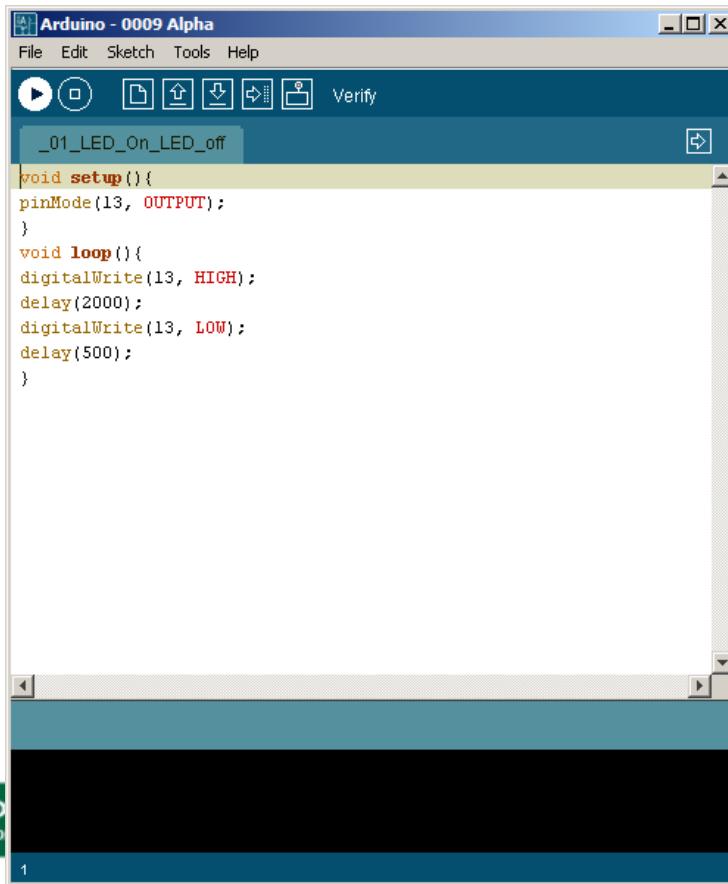
Exercise: Digital Output LED

- 2) LED should be turned ON for 1 second and OFF for 1 second, repeatedly
 - How to fill out **setup()**?
 - How to fill out **loop()**?



Exercise: Digital Output LED

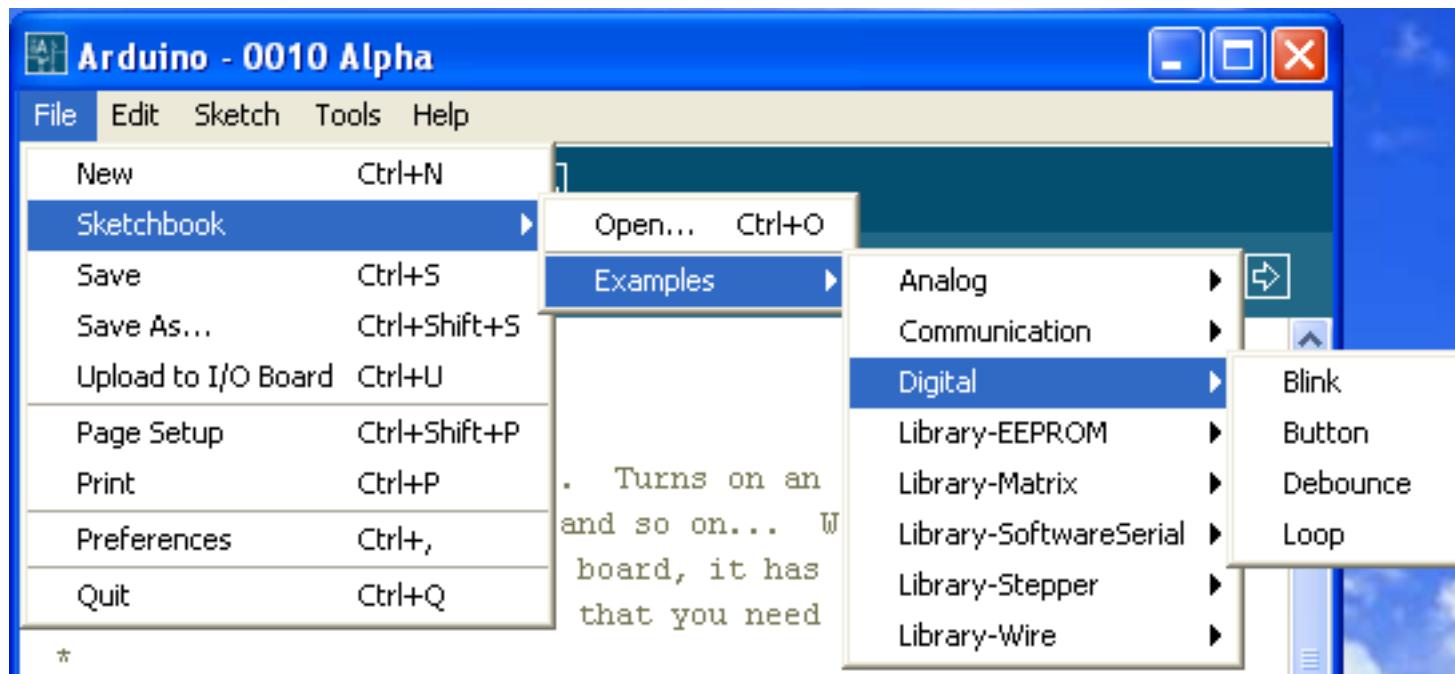
- 2) LED should be turned ON for 1 second and OFF for 1 second, repeatedly



```
void setup(){
  pinMode(13, OUTPUT);
}
```



Lots of Built-in Examples



- And more here:
 - <http://www.arduino.cc/en/Tutorial/HomePage>



Arduino Real-World Examples



Twitter Mood Light – The World's Mood in a Box



Arduino R/C Lawnmower



Turn signal biking jacket



Arduino Quadcopter



Power Laces – the Auto lacing shoe



Makey Makey: <http://makeymakey.com>



Course Announcement

- Next class, we will cover
 - Arduino Mega2560 board
 - Hardware (ATmega2560 MCU)
 - Software
- Next week, two students will form a team
 - Please feel free to find a team member
 - Otherwise, we can help you to match up with a student

