Arduino

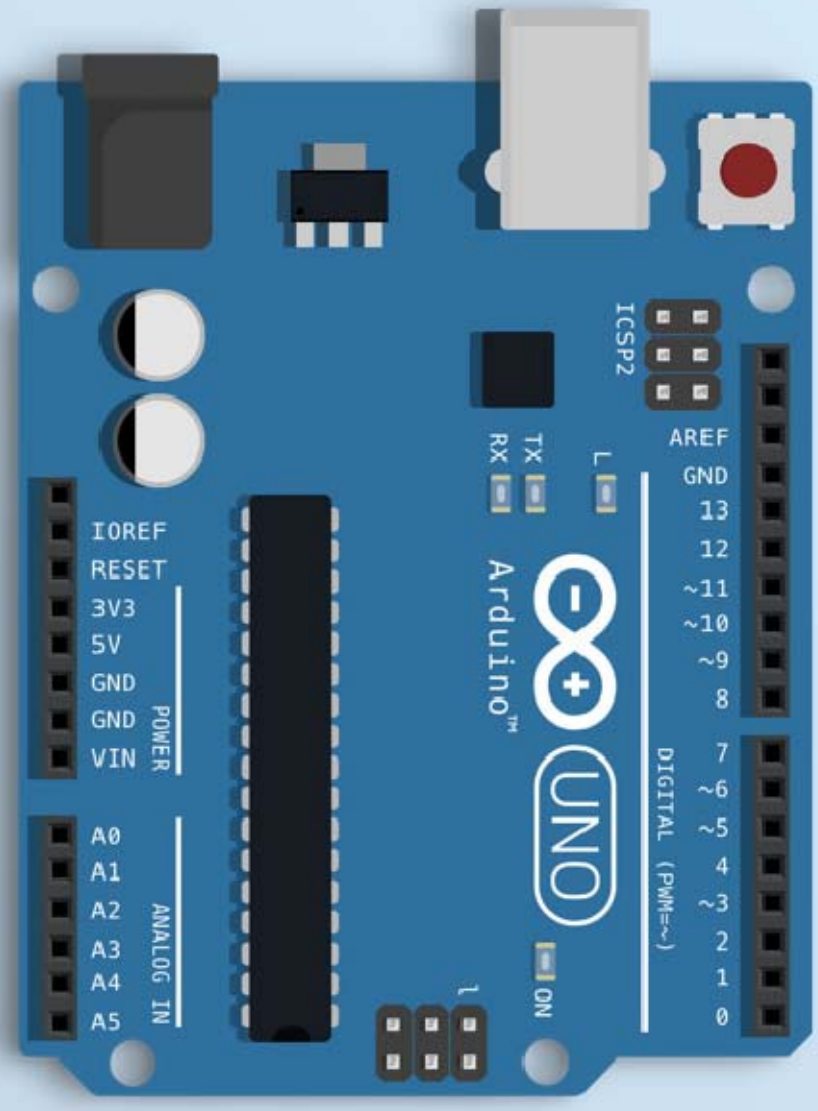
Topics:

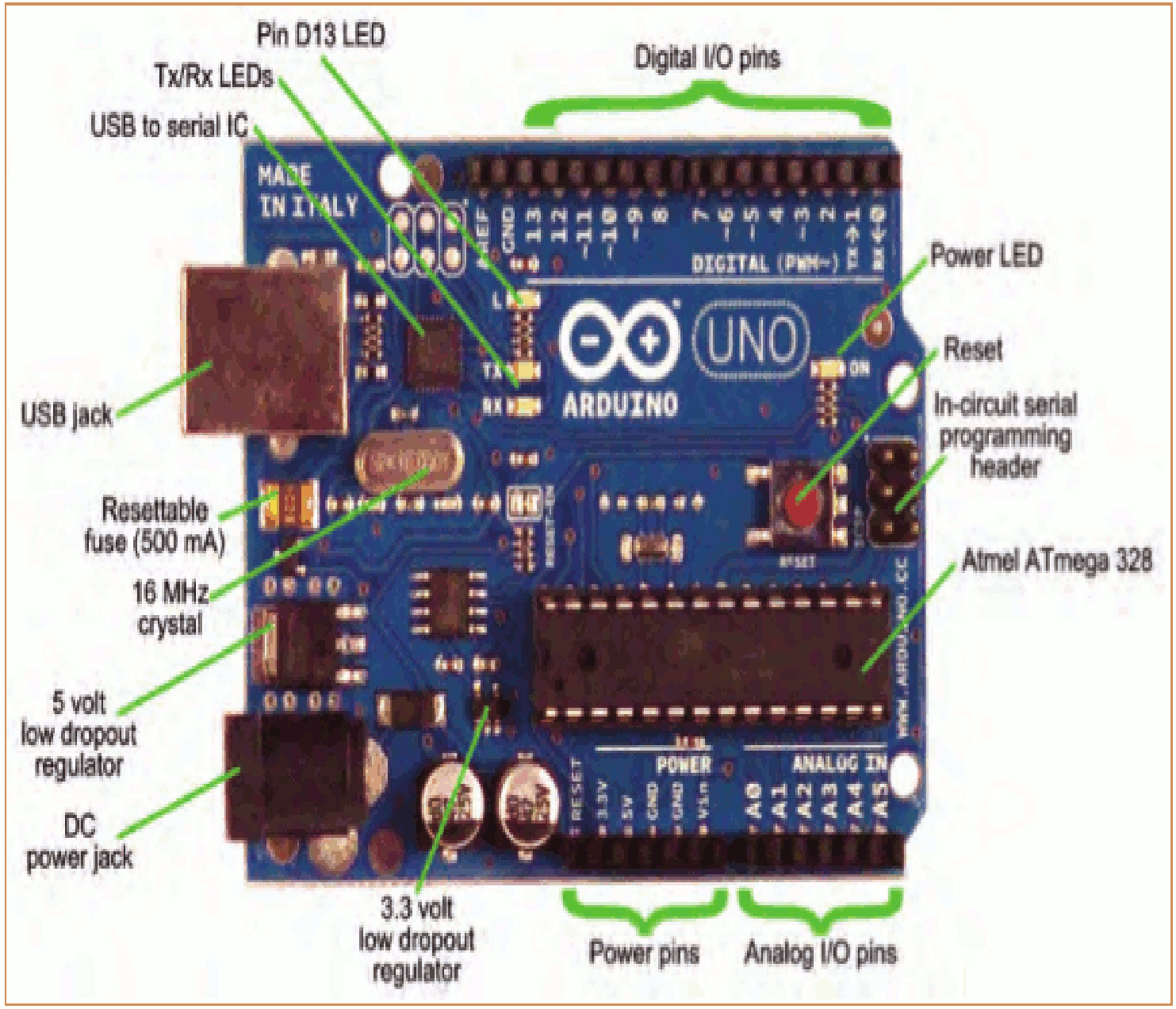
The Arduino

Digital IO

Analog IO

Serial Comm





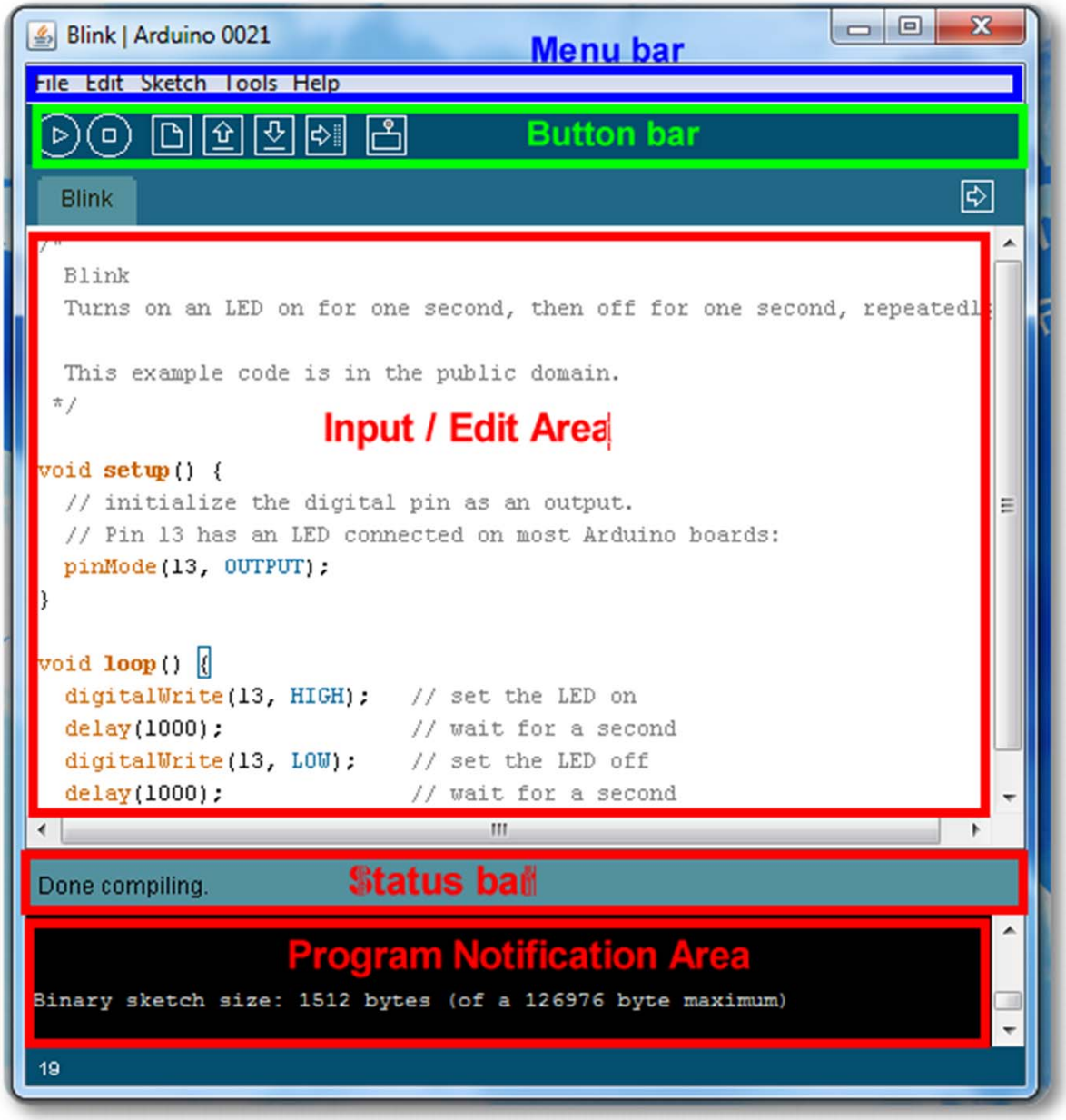
Topic 1: Meet Arduino Uno

Getting Started

- Download & install the Arduino environment (IDE)
- Connect the board to your computer via the USB cable
- If needed, install the drivers
- Launch the Arduino IDE
- Select your board
- Select your serial port
- Open the blink example
- Upload the program

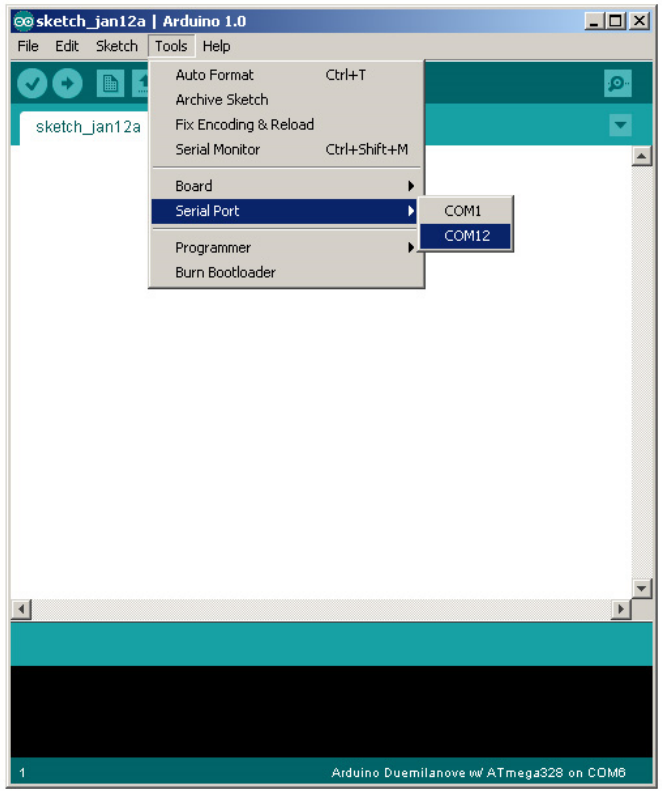
http://arduino.cc/en/Guide/HomePage

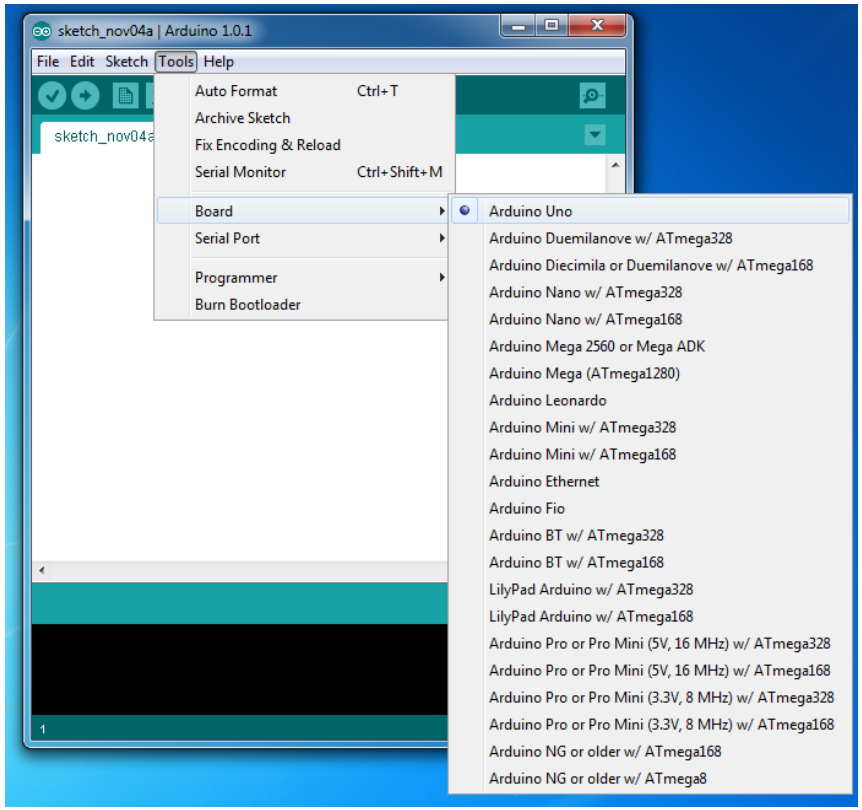
Arduino IDE



http://arduino.cc/en/Guide/Environment

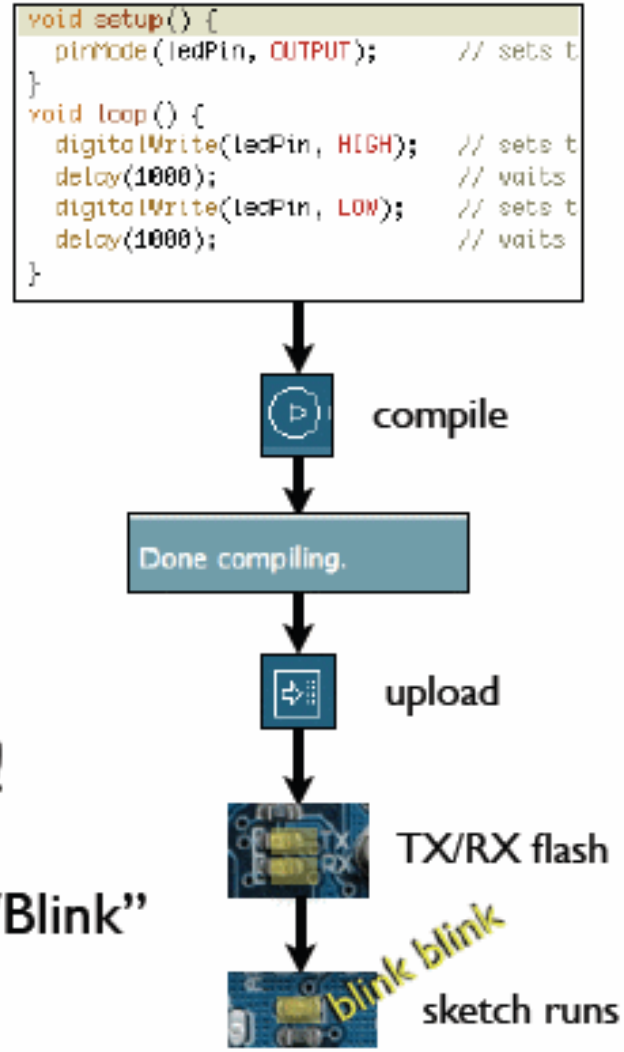
Select Serial Port and Board





Using Arduino

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

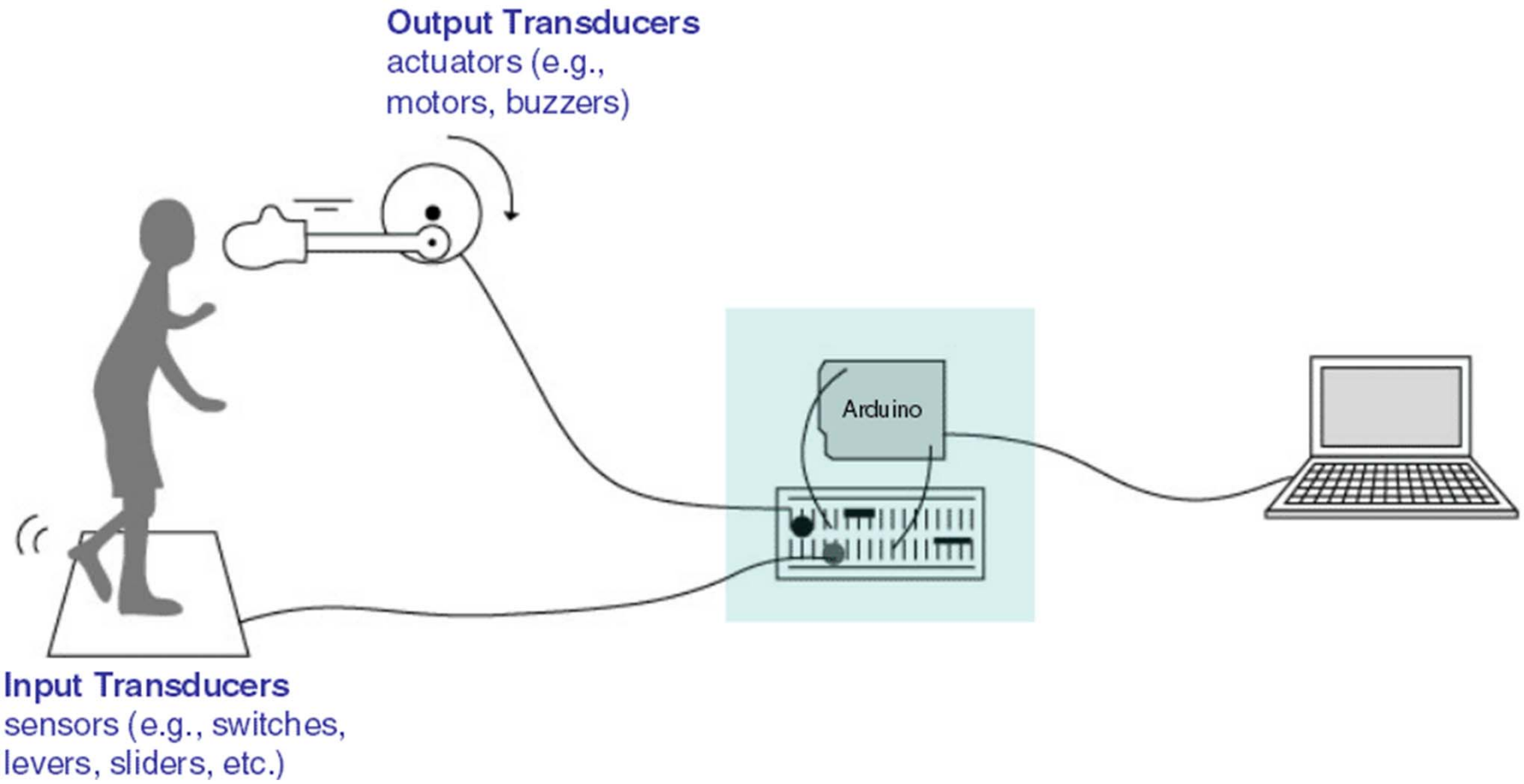


Try it out with the "Blink" sketch!

Load "File/Sketchbook/Examples/Digital/Blink"

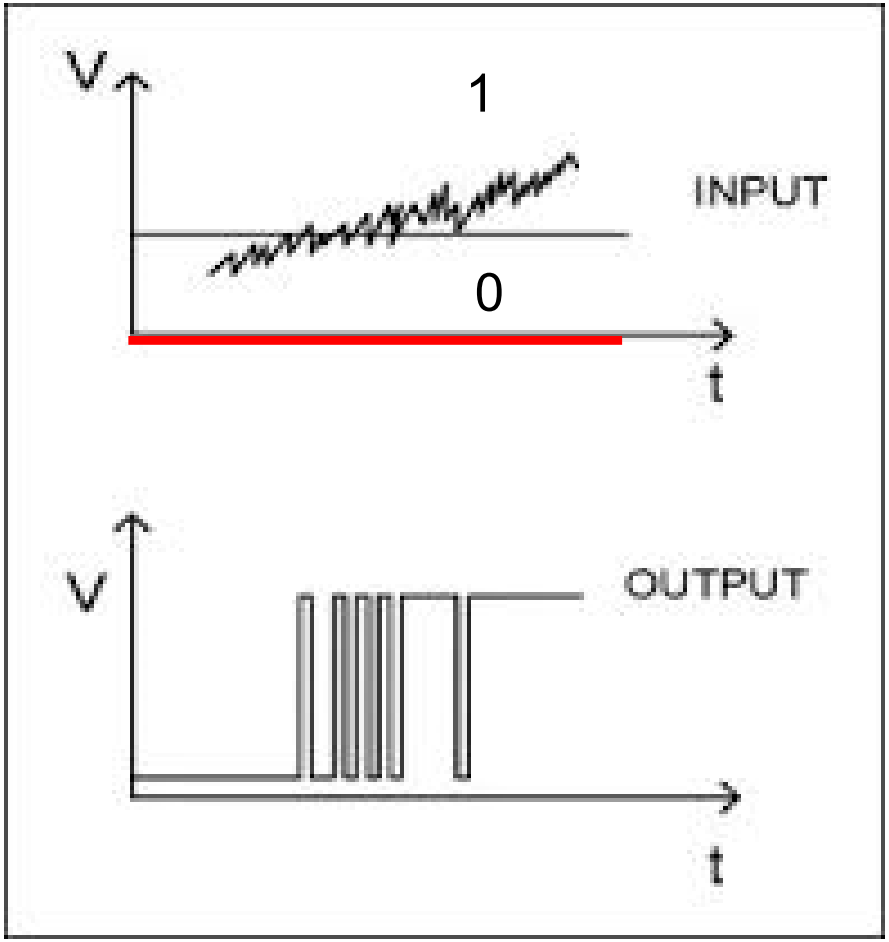
Input/Output

Image from *Theory and Practice of Tangible User Interfaces* at UC Berkley

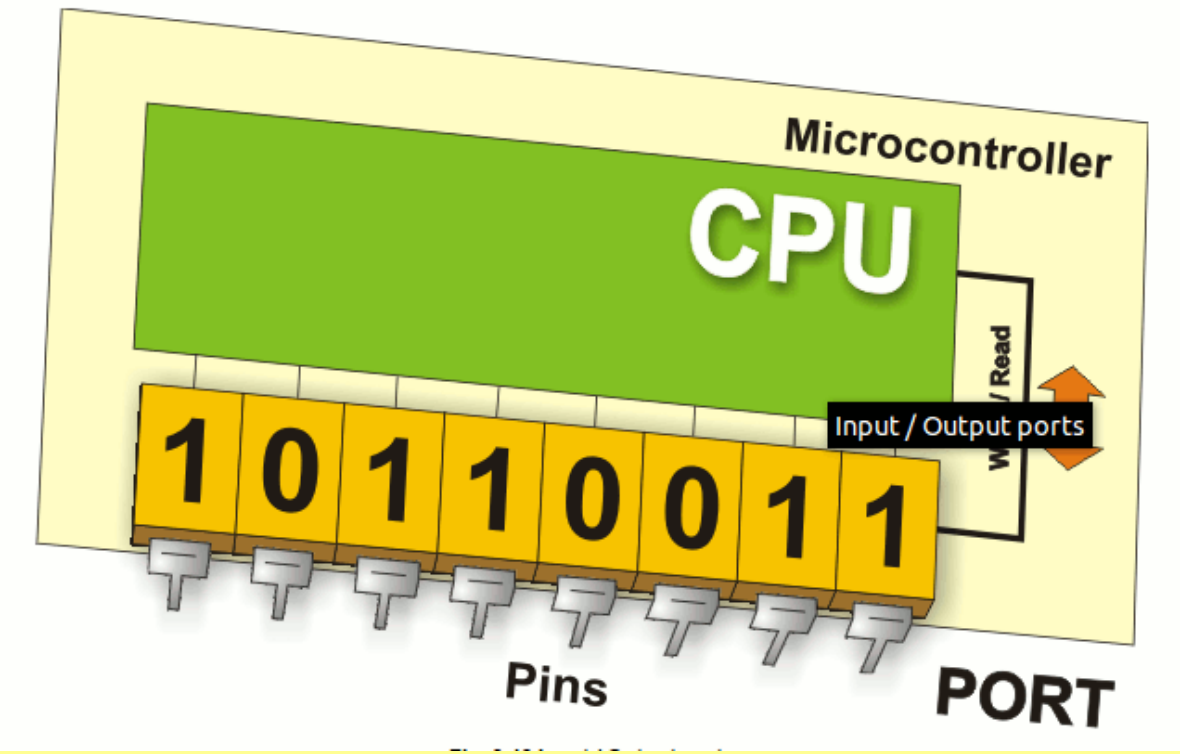


Topic 2: Digital Input/Output

- Digital IO is binary valued—it's either *on* or *off*, 1 or 0
- Internally, all microprocessors are digital, **why**?



Arduino Digital I/0



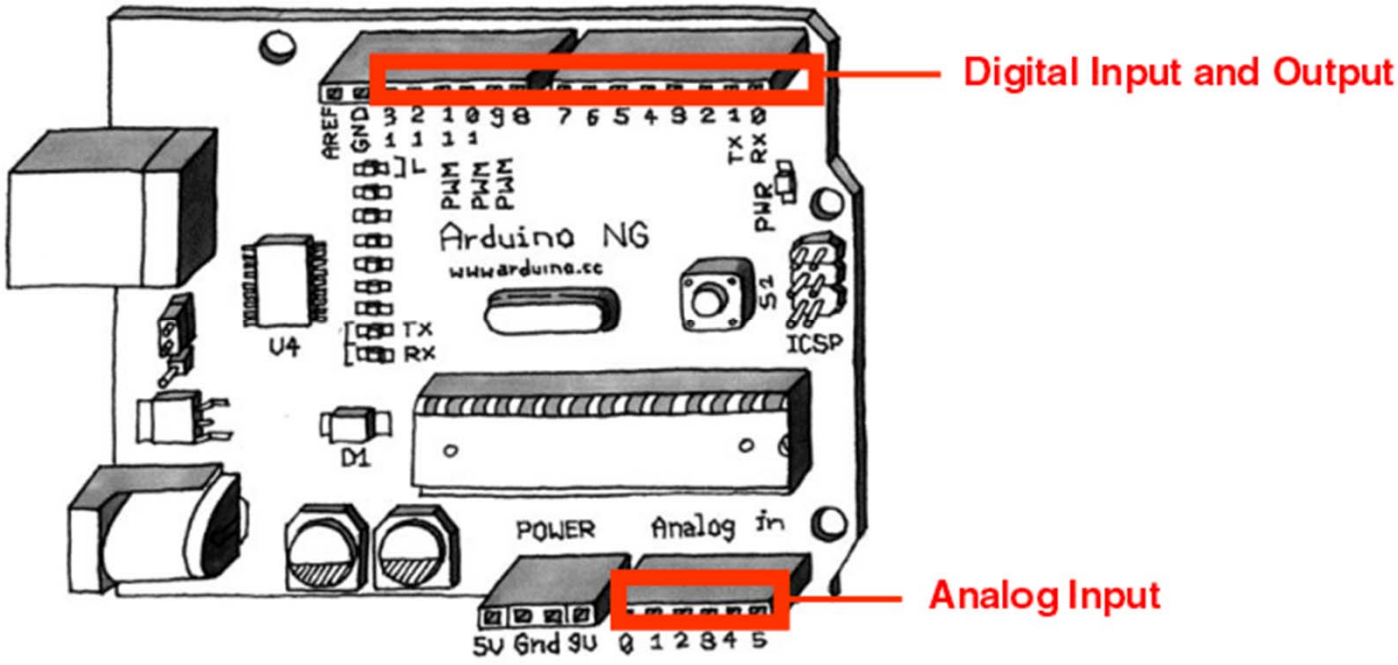
- pinMode(*pin*, *mode*)
  - Sets pin to either INPUT or OUTPUT
- digitalRead(*pin*)
  - Reads HIGH or LOW from a pin
- digitalWrite(*pin*, *value*)
  - Writes HIGH or LOW to a pin

Our First Program

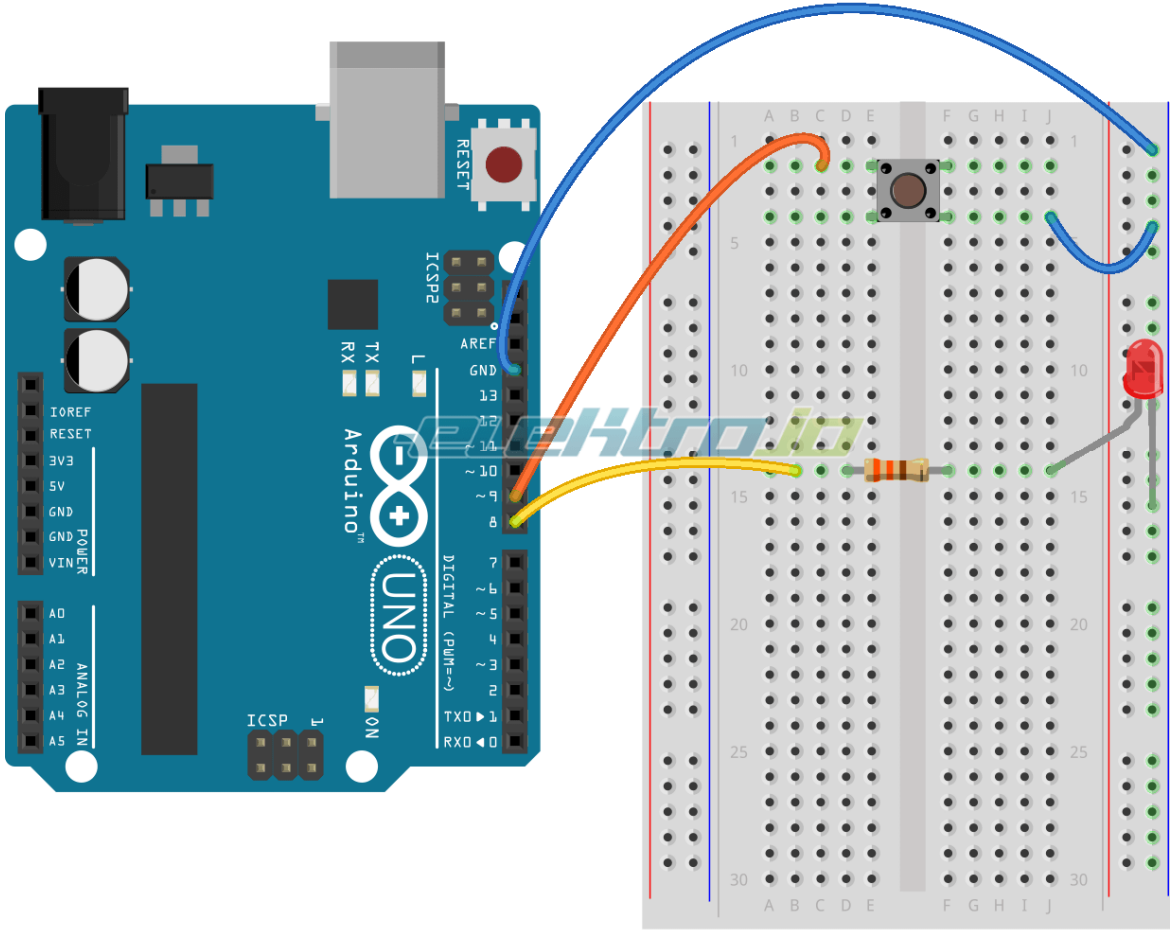
IO Pins

Image from *Theory and Practice of Tangible User Interfaces* at UC Berkley

Two states (binary Signal) vs multiple states (Continuous Signal)



In-class Exercise 1: Digital IO

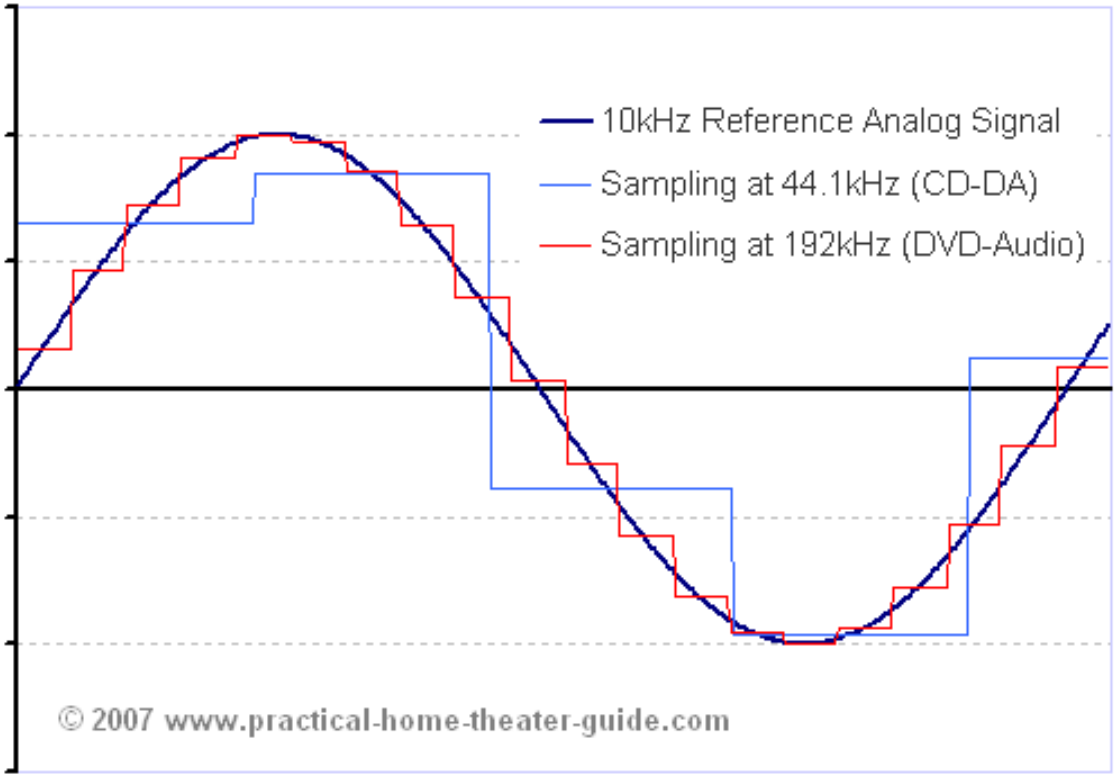


Use a push-button to turn ON/Off LED

Topic 3: Analog Input

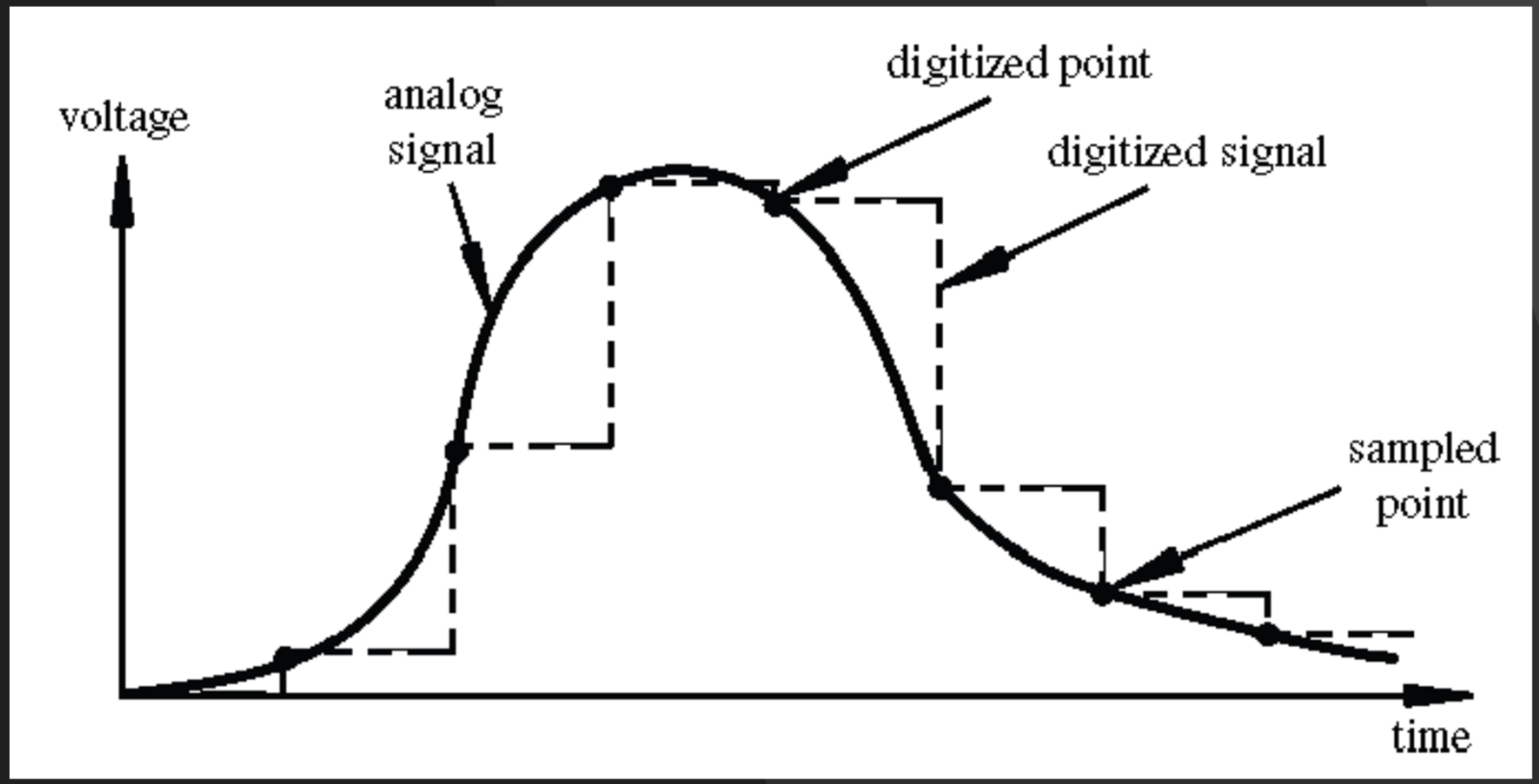
Think about music stored on a CD---an analog signal captured on digital media

- Sample rate
- Word length



Arduino Analog Input

- *Resolution*: the number of different voltage levels (i.e., *states*) used to discretize an input signal
- Resolution values range from 256 states (8 bits) to 4,294,967,296 states (32 bits)
- The Arduino uses 1024 states (10 bits)
- Smallest measurable voltage change is 5V/1024 or 4.8 mV
- Maximum sample rate is 10,000 times a second



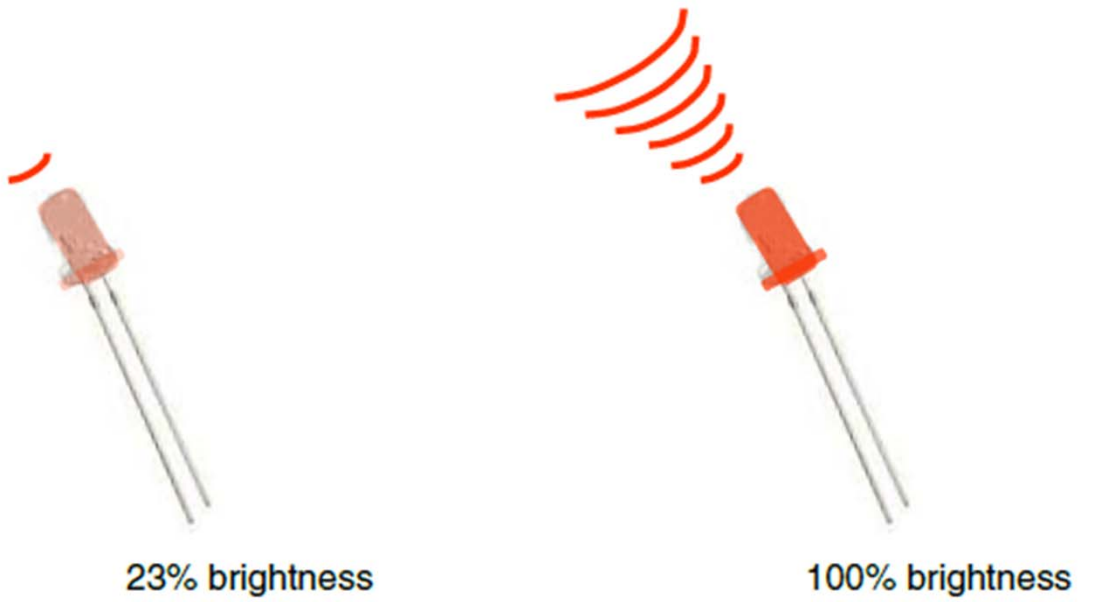


Image from *Theory and Practice of Tangible User Interfaces* at UC Berkley

Topic 3: Analog Output

- Can a digital devise produce analog output?
- Analog output can be simulated using pulse width modulation (PWM)

Pulse Width Modulation

- Can't use digital pins to directly supply say 2.5V, but can pulse the output on and off really fast to produce the same effect
- The on-off pulsing happens so quickly, the connected output device "sees" the result as a reduction in the voltage

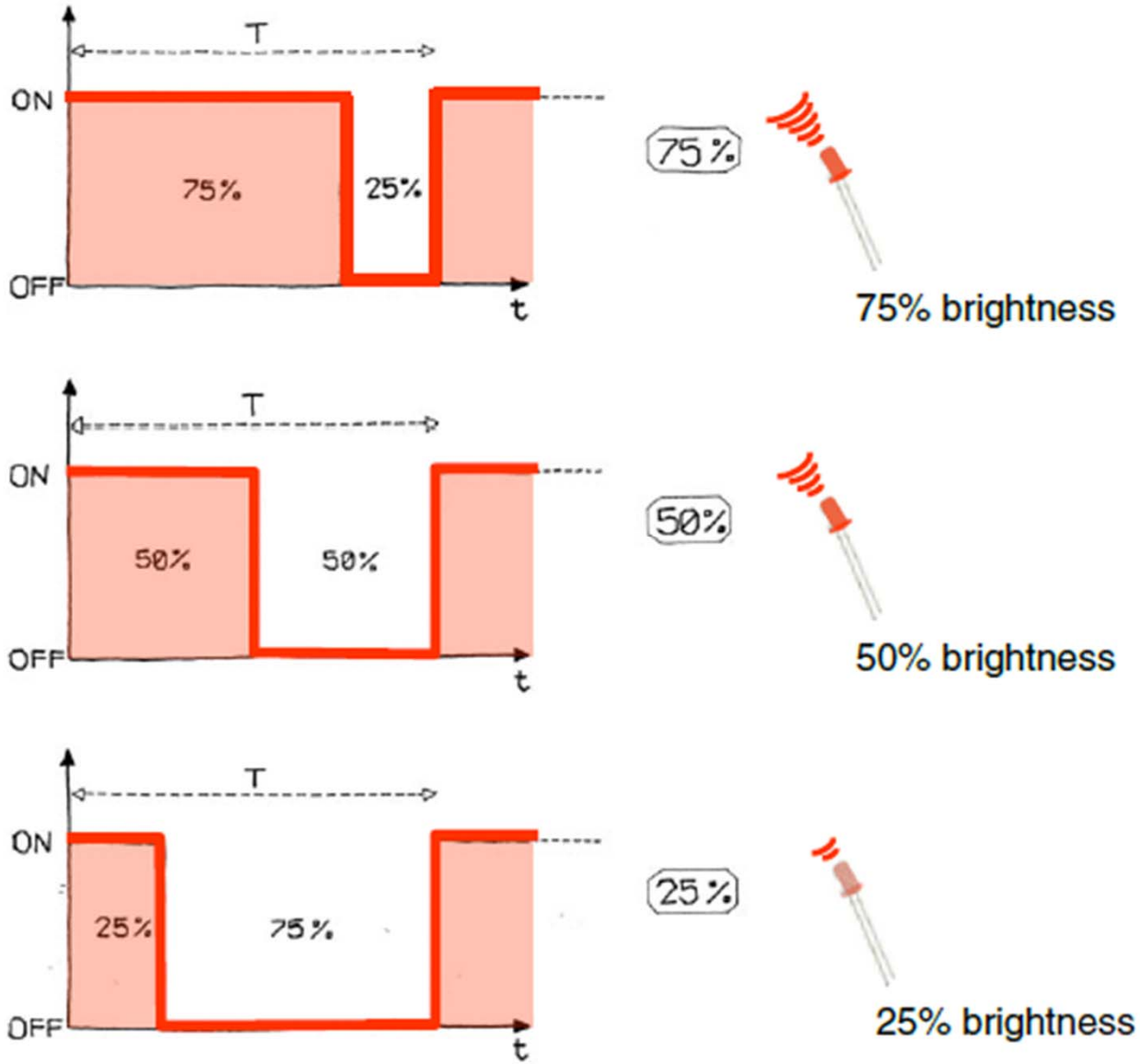


Image from *Theory and Practice of Tangible User Interfaces* at UC Berkley

PWM Duty Cycle

output voltage = (on_time / cycle_time) * 5V

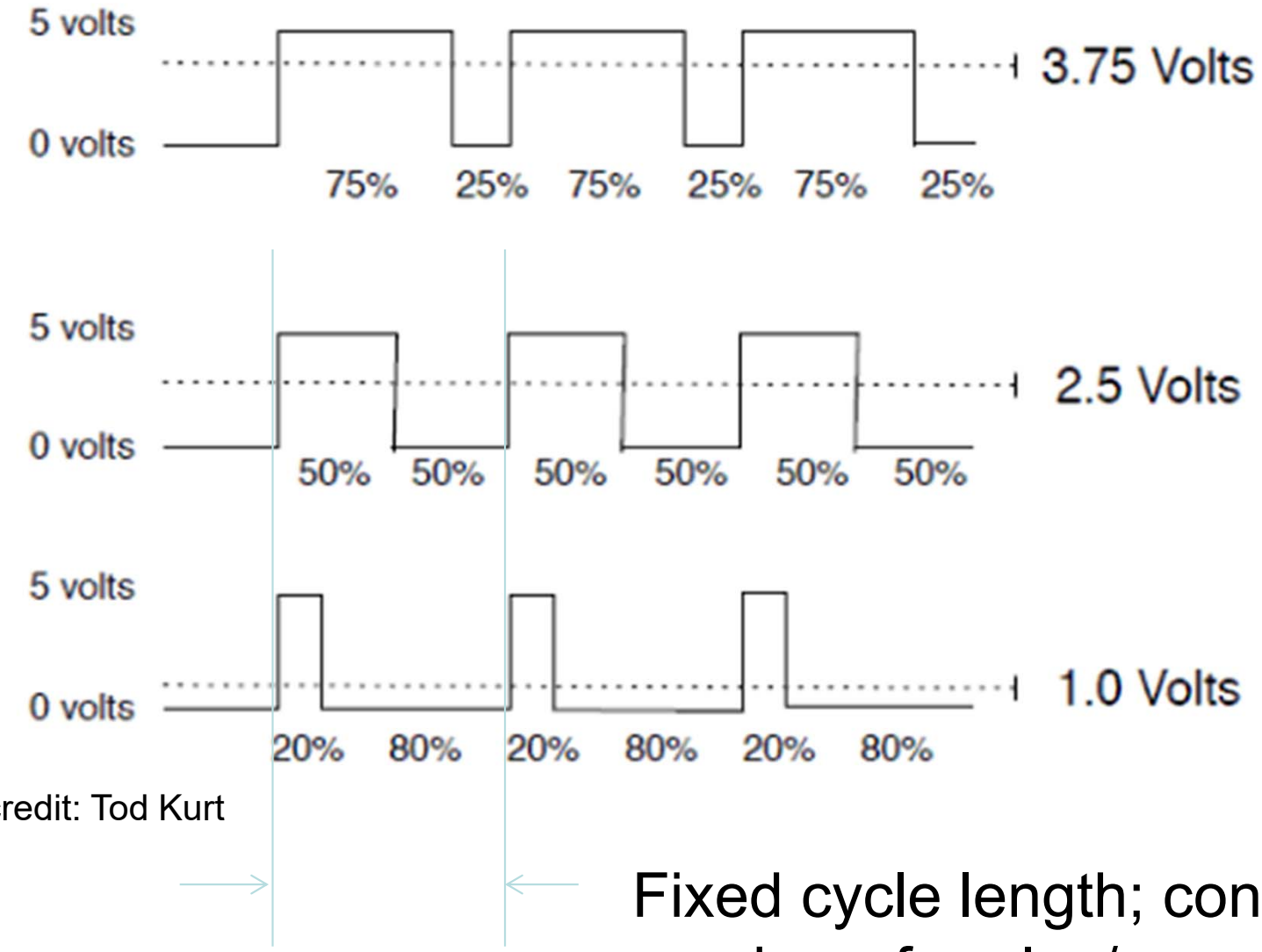


Image credit: Tod Kurt

Fixed cycle length; constant number of cycles/sec

PMW Pins

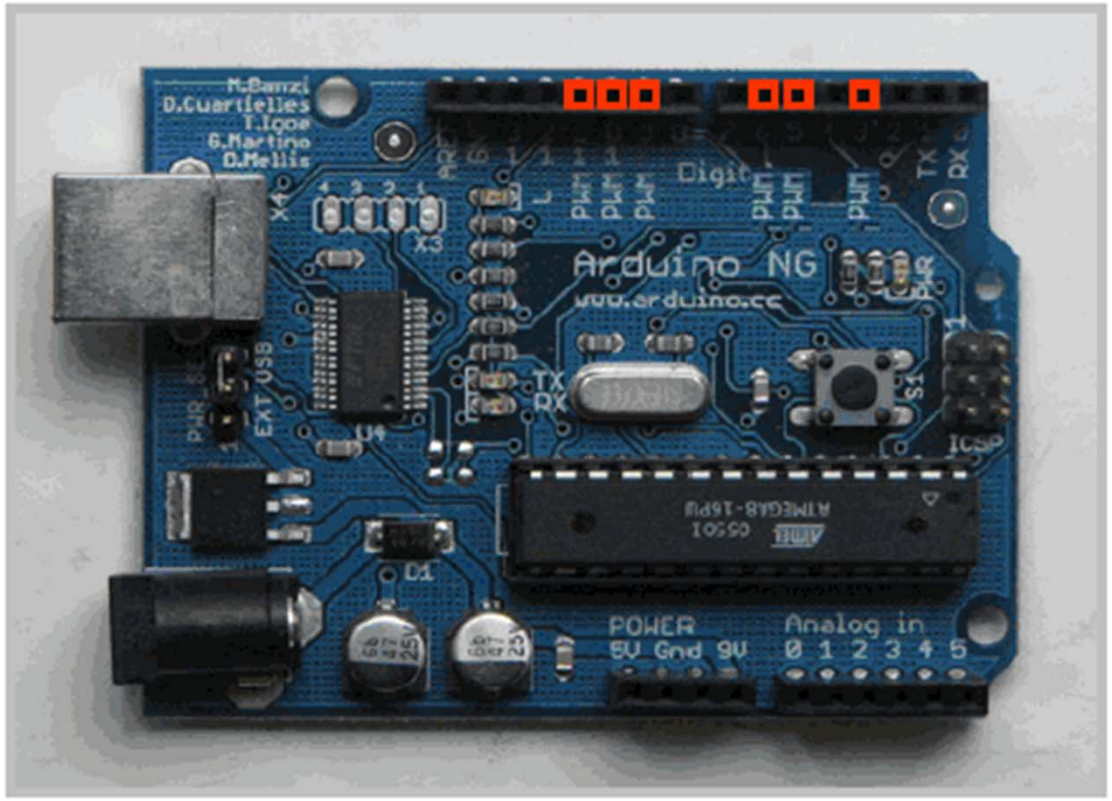
Command:
analogWrite(pin,value)

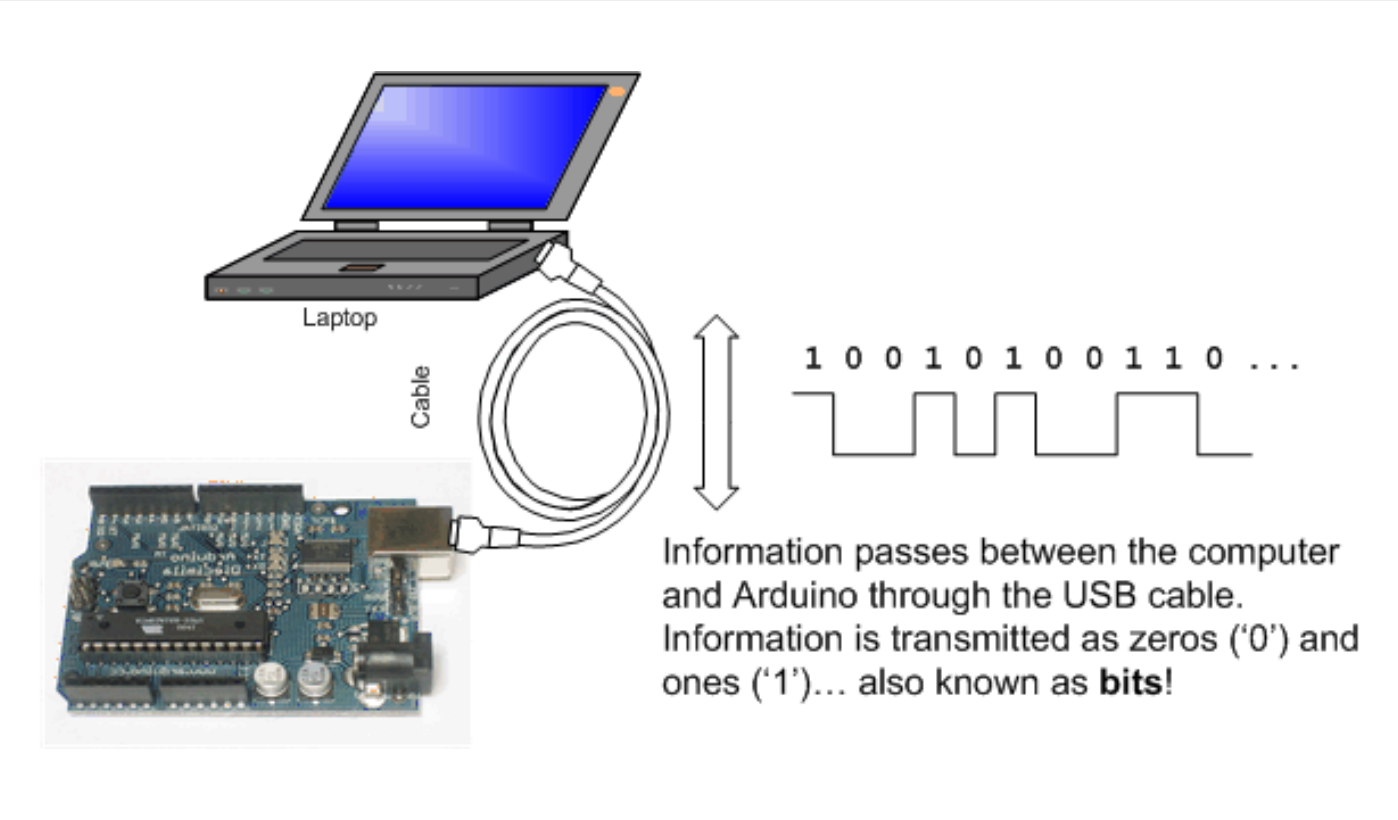
value is duty cycle: between 0 and 255

Examples:

- analogWrite(9, 128) for a 50% duty cycle
- analogWrite(11, 64) for a 25% duty cycle

Your Arduino board has built in PWM circuits, on pins 3, 5, 6, 9, 10, and 11





Topic 4: Serial Communication

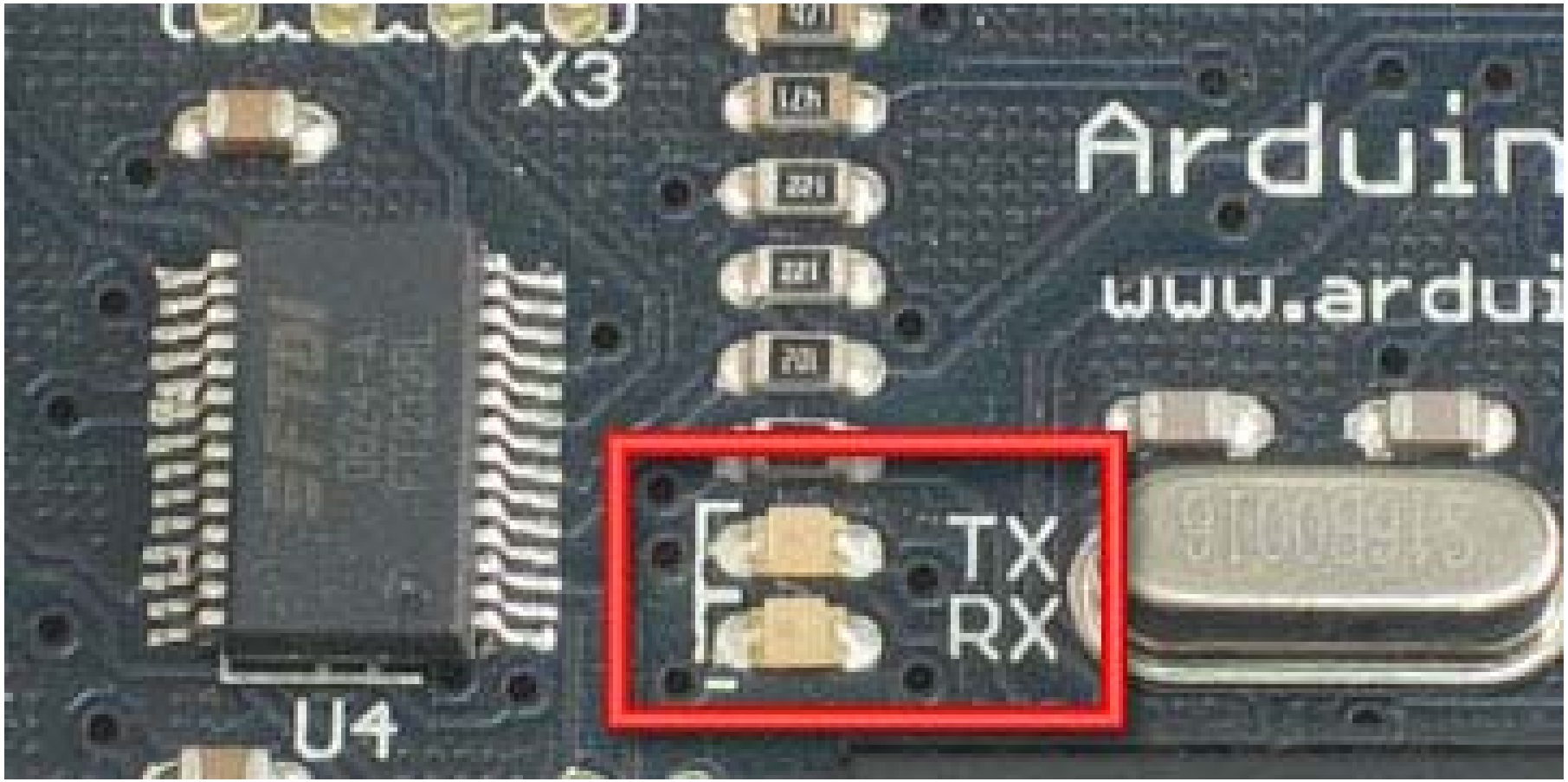
Image from http://www.ladyada.net/learn/arduino/lesson4.html

Serial Communications

- "Serial" because data is broken down into bits, each sent one after the other down a single wire.
- The single ASCII character 'B' is sent as:

```
'B' =  0 1 0 0 0 0 1 0
    =  L H L L L L H L
    =
```

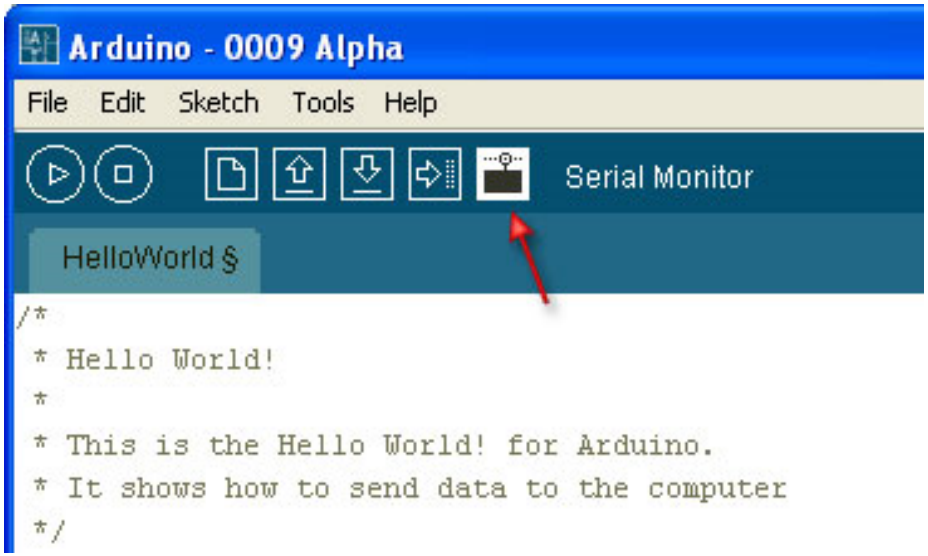
- Toggle a pin to send data, just like blinking an LED
- You could implement sending serial data with `digitalWrite()` and `delay()`
- A <u>single data wire</u> needed to send data. One other to receive.



Serial Communication

- ***Compiling*** turns your program into binary data (ones and zeros)
- ***Uploading*** sends the bits through USB cable to the Arduino
- The two LEDs near the USB connector blink when data is transmitted
  - **RX** blinks when the Arduino is receiving data
  - **TX** blinks when the Arduino is transmitting data

Open the Serial Monitor and Upload the Program



Some Commands

- Serial.begin()
  - Serial.begin(9600)
- Serial.print() or Serial.println()
  - Serial.print(value)
- Serial.read()
- Serial.available()
- Serial.write()
- Serial.parseInt()

Lab Exercise: Serial Communication

- Modify your program from in-class exercise 2-part 2 to control the intensity of the LED attached to pin 9 based on keyboard input.

- Use the Serial.parseInt() method to read numeric keyboard input as an integer.

- An input of 9 should produce full intensity and an input of 0 should turn the LED off.