BLOSSOMING POSSIBILITIES ONE HEART BEAT AT A TIME

v1.1 - Demo

GROUP 3
PHYSICS 173 LAB

Short Video: To Remember the Fallen Ones

PS: They’re not dead, as of now.

Problems in Philippine Agriculture

Weak Sustenance and Growth of Local Agricultural Output

Recent Problems in Crop Supplies and Mass Importation

Few Investments on Food Security and Innovation

Market Analysis

Currently, there are 5 million famers that cultivate 9.7 million hectares or about 30% of the land in the Philippines. By offering our services to them, it will generate us a huge amount of revenue.

What's The Big Idea?

What if there is a prototype for a product that can be used for small-scale purposes that can be used by people in their homes to help with their farming needs, enabling them to be sustainable and eco-friendly?

But How Do We Do It?

A good way to do this is by prototyping it with electronics, and what better way to do it with tiny computers that we call microprocessors?

Short Video: Arduino Explainer

Got this from YouTube

Ardu... what?

- Arduino means “strong friend” in Italian and is developed in 2005 in Ivrea, Italy by Massimo Banzi and David Cuartielles.

- Arduino boards and open-source hardware and uses an processor that is developed by 

.

- Coding with the Arduino board is accessible and transferrable with both compiler and interpreter languages.

Functions of Arduino

- Arduino is great for prototyping ideas as it has easy access to I/O.
- Arduino is flexible and open-source, and runs on low power.
- Arduino’s microprocessor unit can be used to make transducers and sensors that can be controlled.

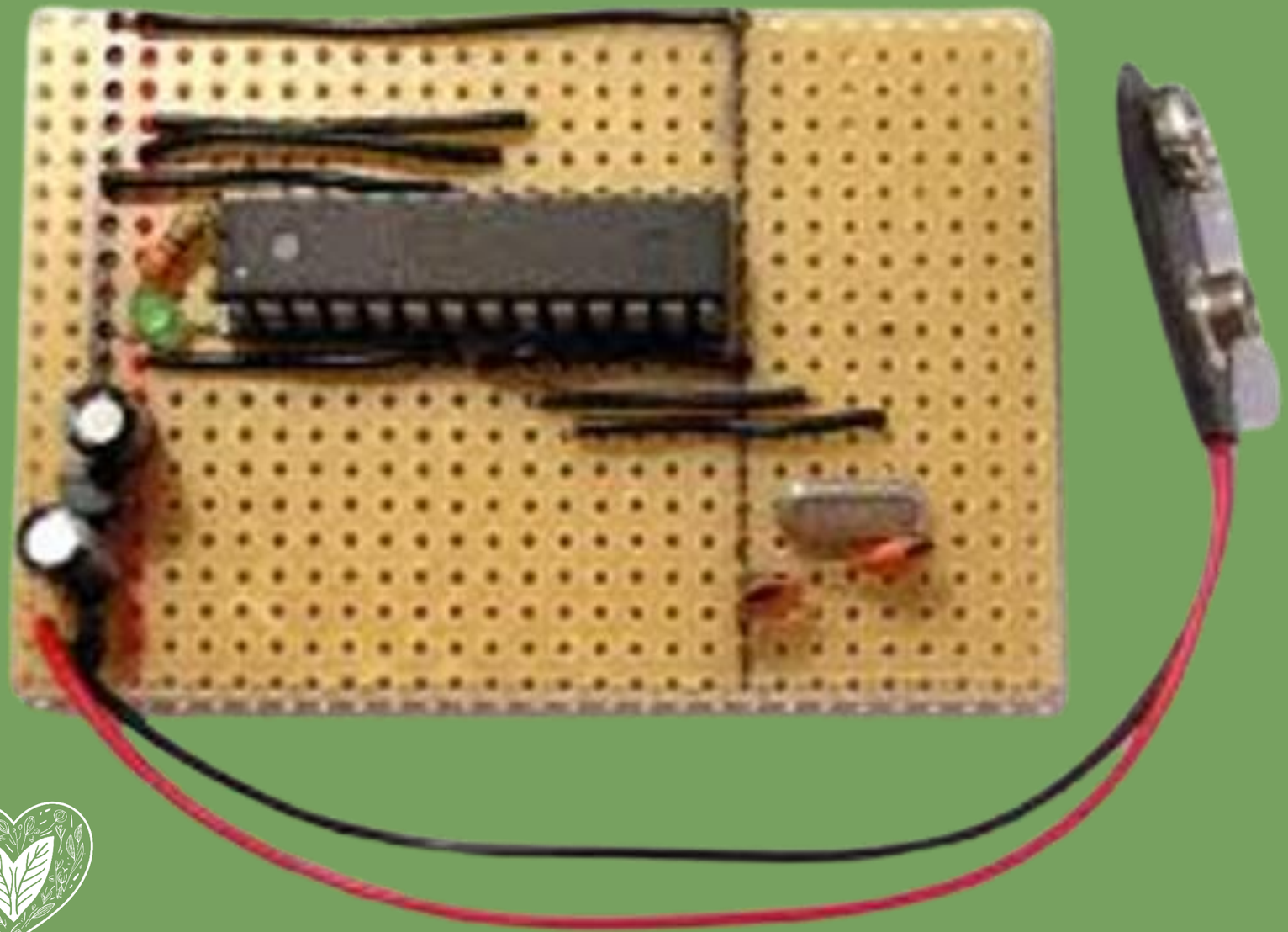
Benefits of Arduino

- Arduino is the go-to gear for artists, hobbyists, students, and anyone with a gadgetry dream.

- Arduino rose out of another formidable challenge: how to teach students to create electronics, fast.

- With Arduino, you can control almost everything around you be it simple LED or giant robots.

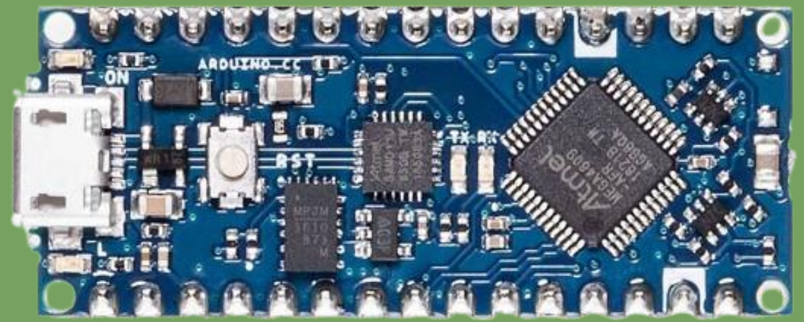
Types of Arduino



DIY VERSION ARDUINO (ATMEGA328P)

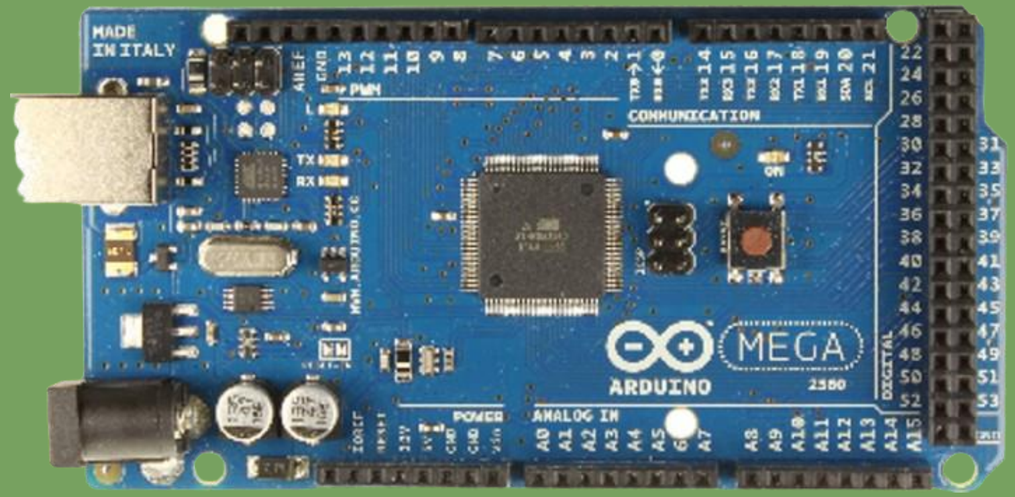
Types of Arduino

UNO



NANO

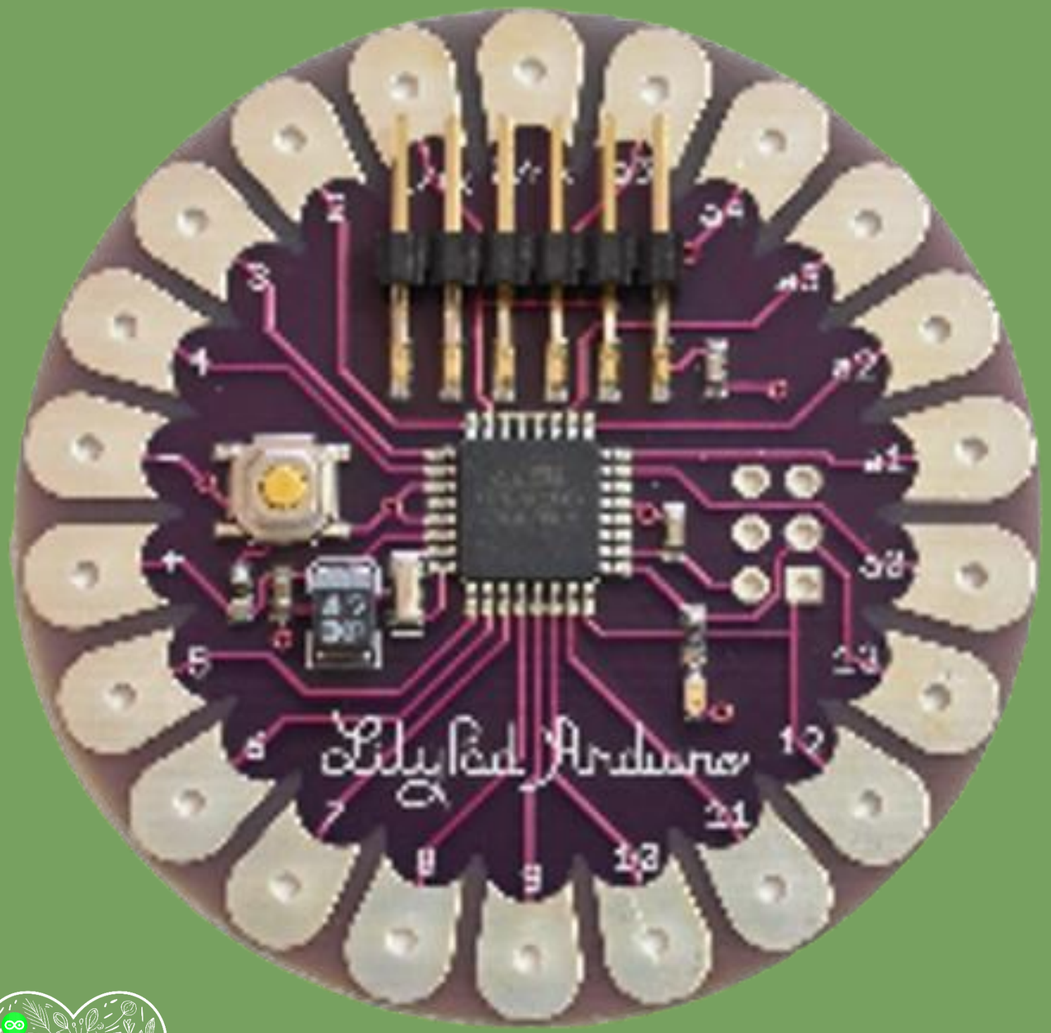
ARDUINO



MEGA

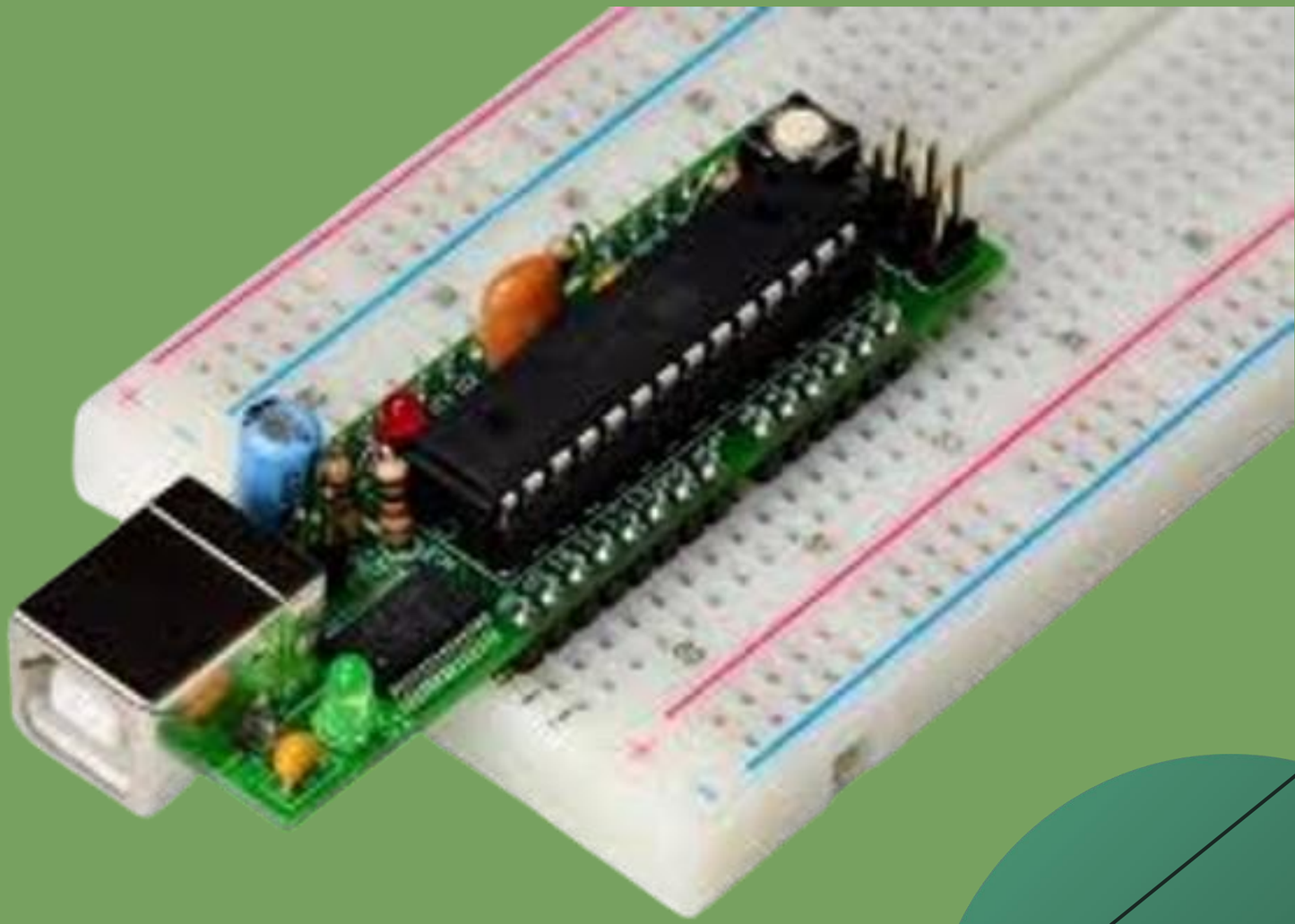
RS-232

Types of Arduino



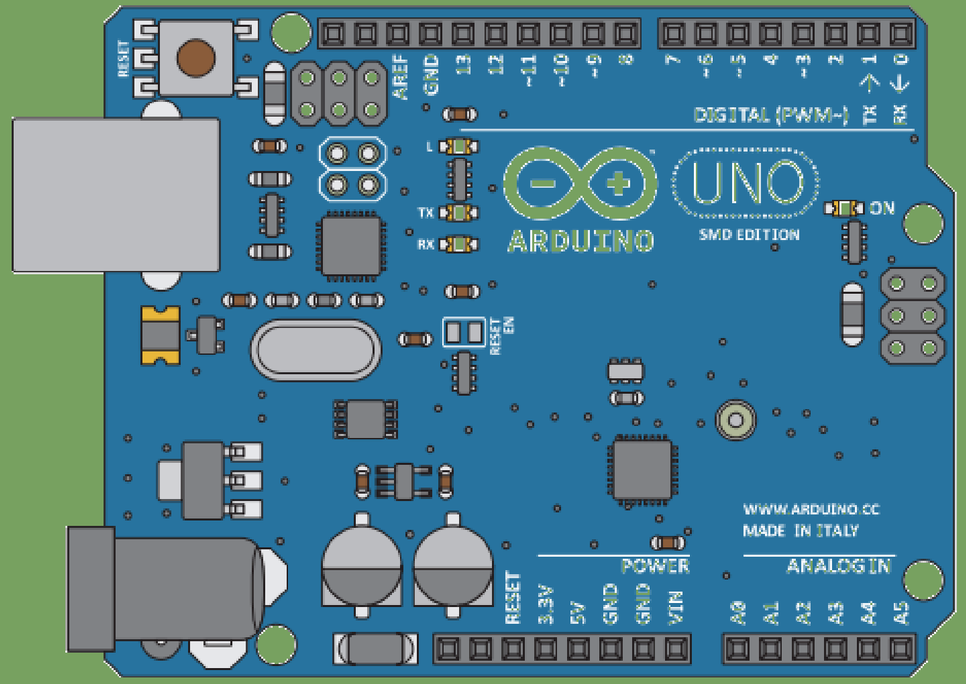
LILYPAD

BOARDUINO

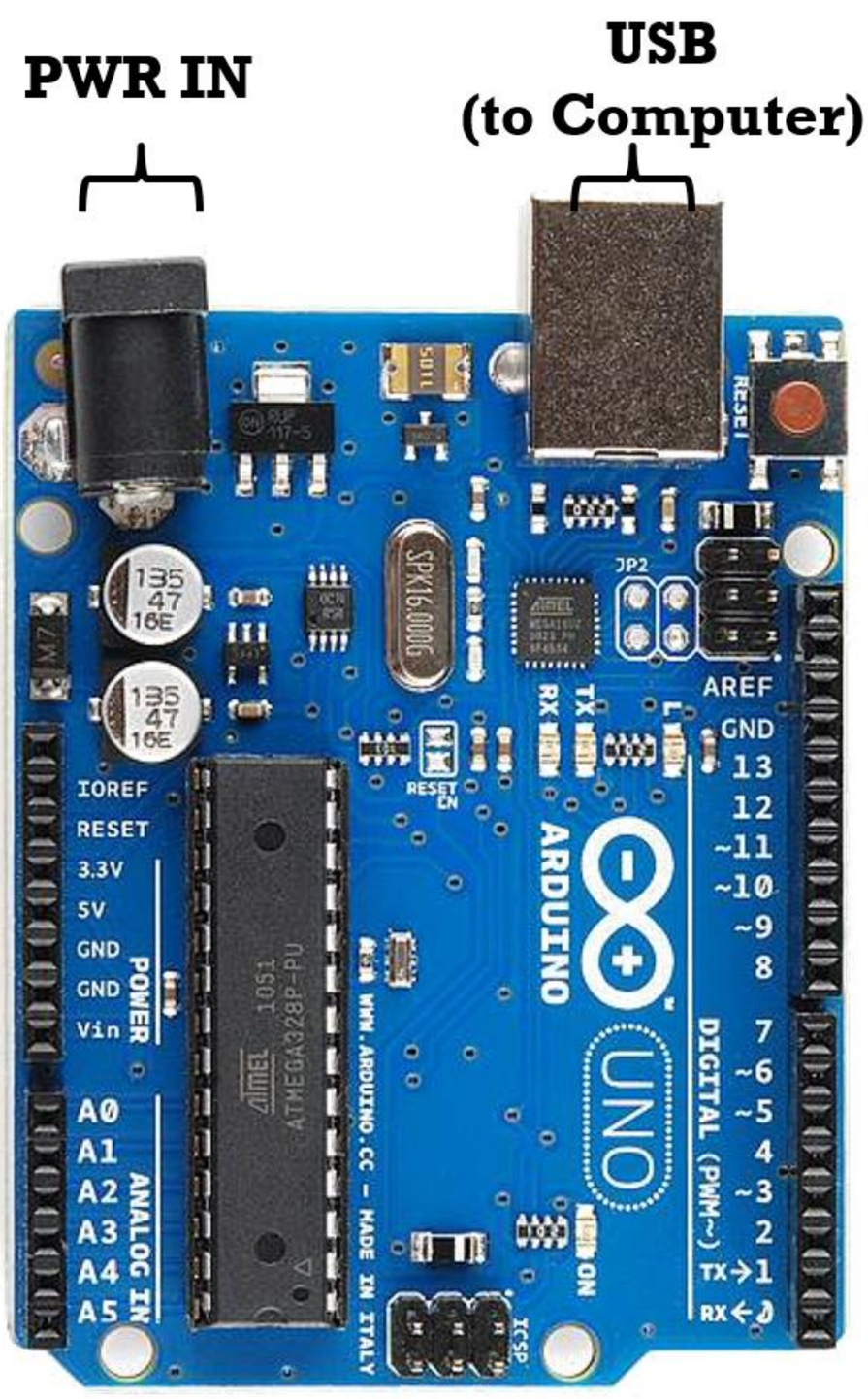


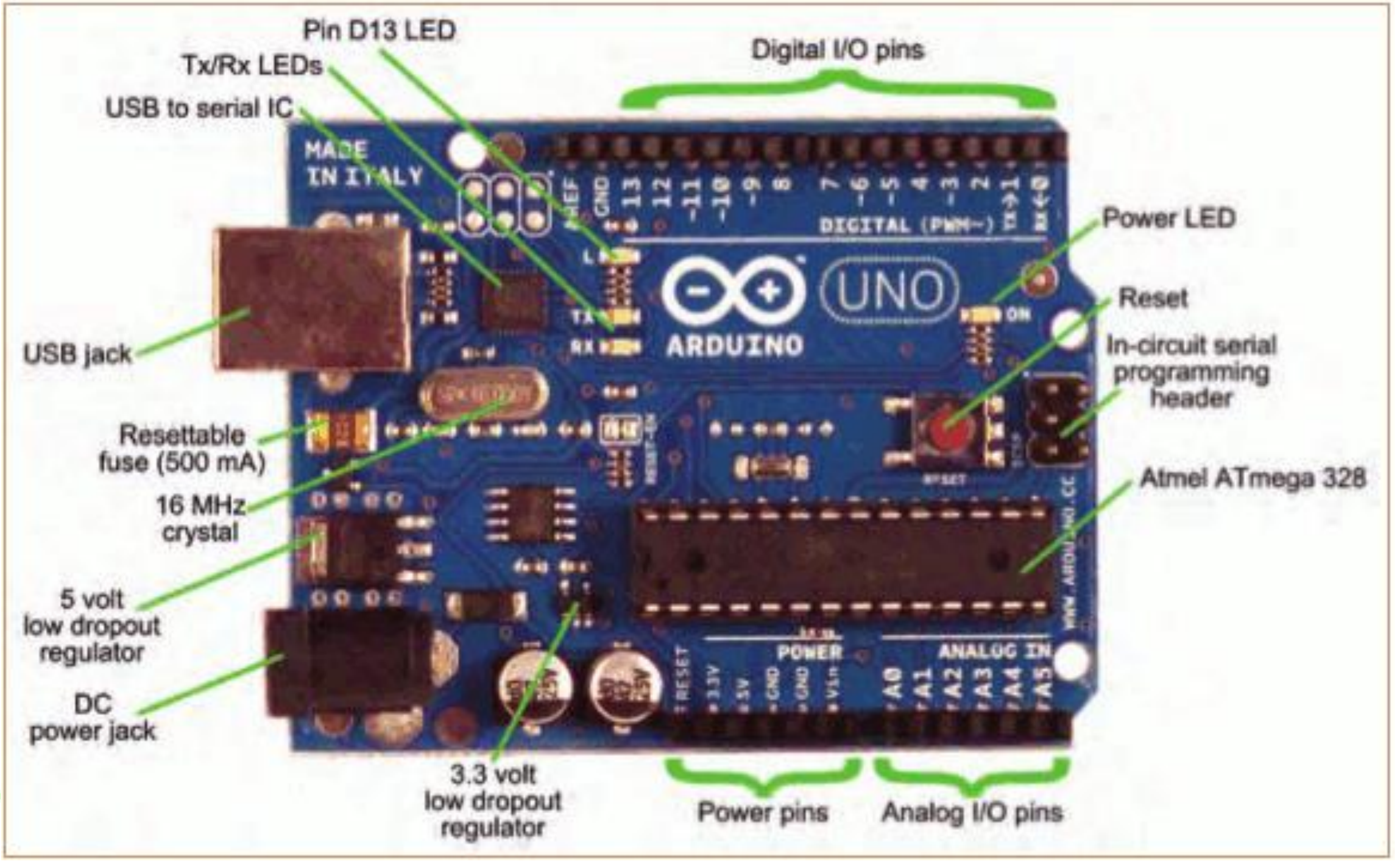
The Standard Uno

- **What does it have?**
  - **14 Digital In/Out pins (6 can be used as PWM)**
  - **6 Analog Inputs**
  - **A USB Connection**
  - **A Power Jack**
  - **Reset Button**
  - **On-board LED**
  - **SCL/SDA pins (Serial Clock/ Serial Data pins)**

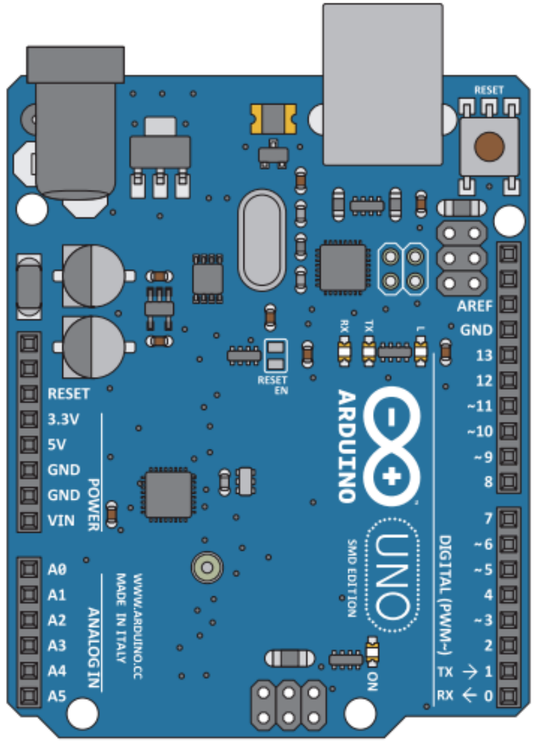


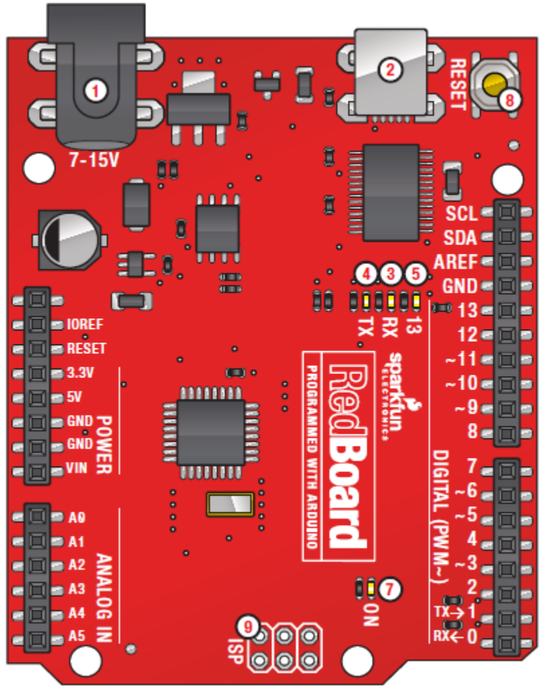
- **In short, it contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.**

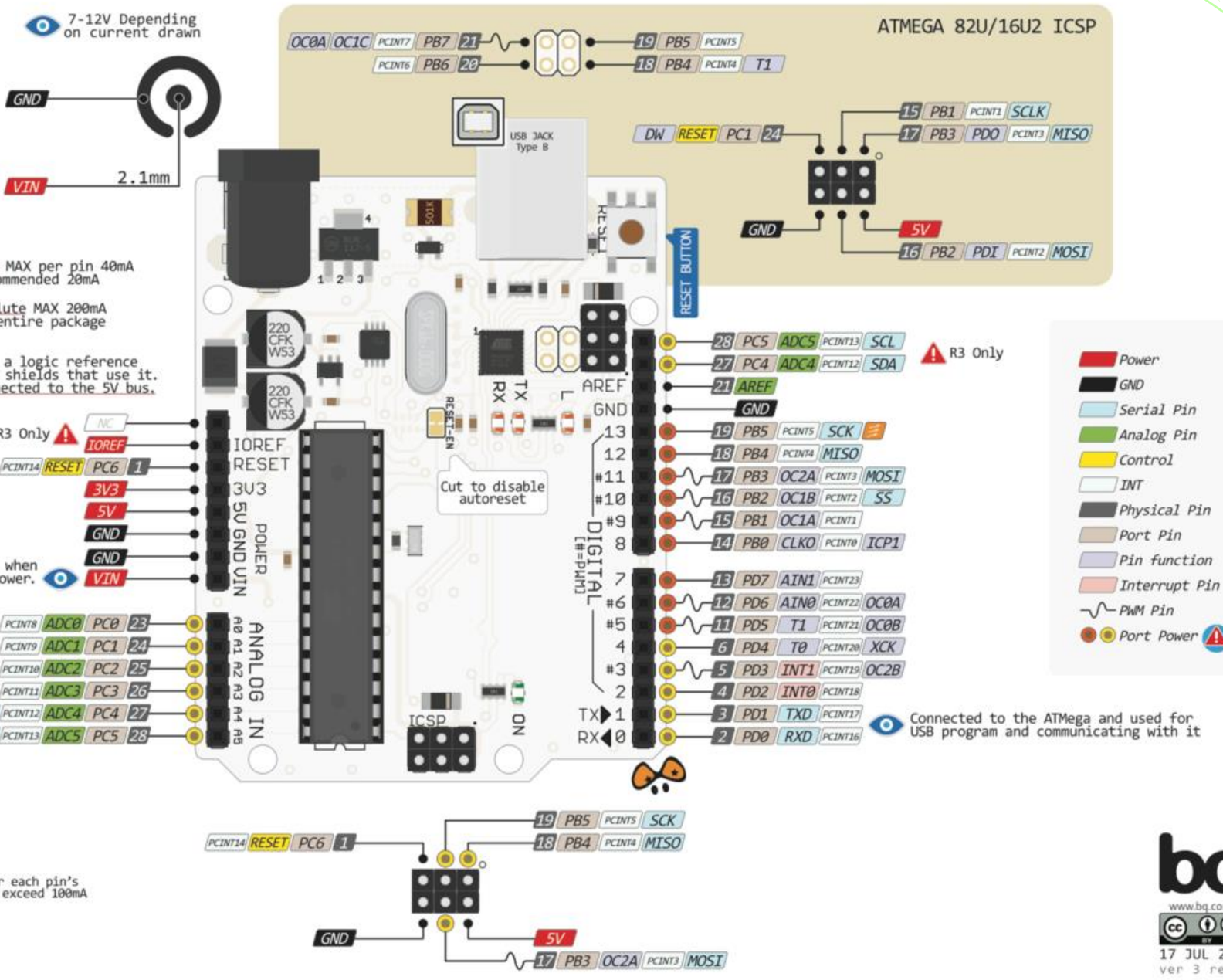

PWR IN
USB
(to Computer)
RESET
SCL\SDA
(I2C Bus)
POWER
5V / 3.3V /
GND
Analog
INPUTS
Digital I\O
PWM(3, 5, 6, 9, 10,
11)
IOREF
RESET
3.3V
5V
GND
GND
Vin
POWER
A0
A1
A2
A3
A4
A5
ANALOG IN
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
ARDUINO
UNO
ATMEGA328P-PU
WWW.ARDUINO.CC – MADE IN ITALY
ON
ICSP
JP2
RESET
TX
RX


Pin D13 LED
Tx/Rx LEDs
USB to serial IC
Digital I/O pins
Power LED
Reset
In-circuit serial programming header
USB jack
Resettable fuse (500 mA)
16 MHz crystal
Atmel ATmega 328
5 volt low dropout regulator
DC power jack
3.3 volt low dropout regulator
Power pins
Analog I/O pins
MADE IN ITALY
DIGITAL (PWM~)
ARDUINO
UNO
POWER
ANALOG IN

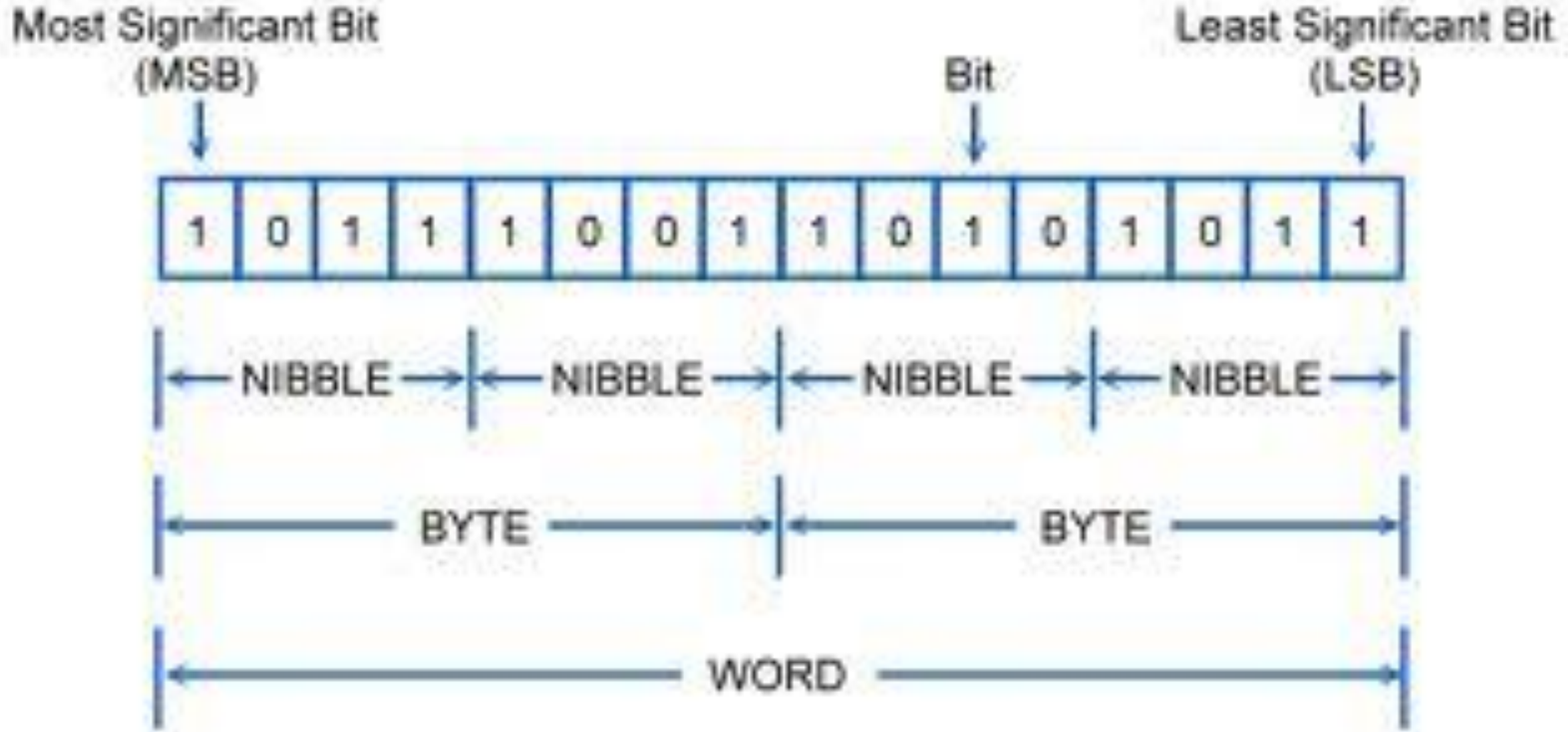
UNO PINOUT



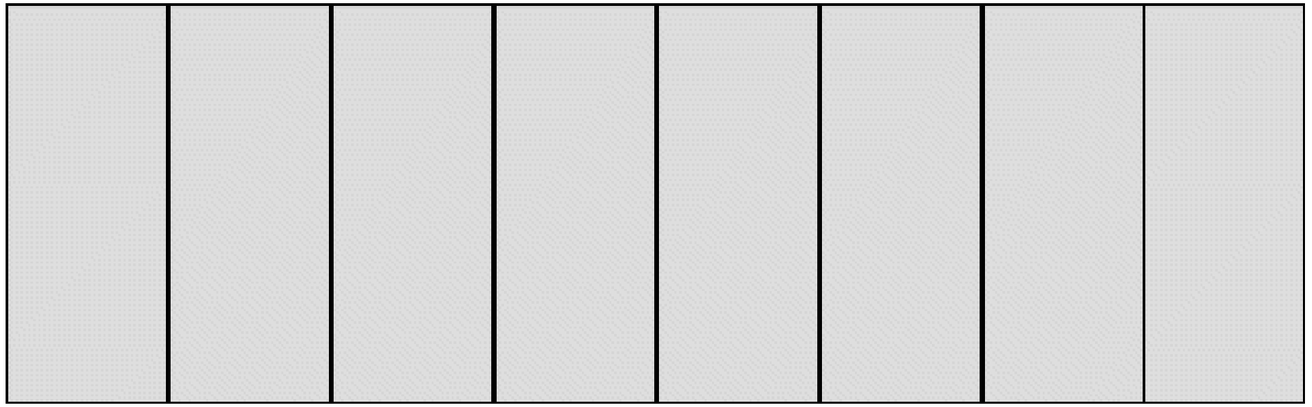


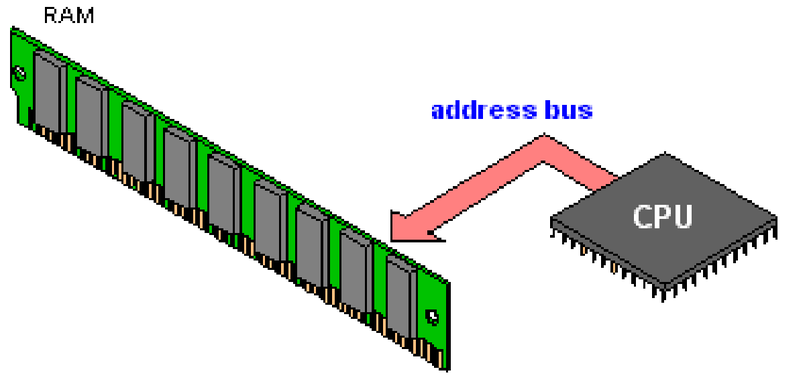


bq
www.bq.com
17 JUL 2014
ver 3 rev 0


Most Significant Bit
(MSB)
Bit
Least Significant Bit
(LSB)
1 0 1 1 1 0 0 1 1 0 1 0 1 0 1 1
NIBBLE
NIBBLE
NIBBLE
NIBBLE
BYTE
BYTE
WORD

Bit positions: 8 7 6 5 4 3 2 1





$2^0 = 1$

$2^1 = 2$

$2^2 = 4$

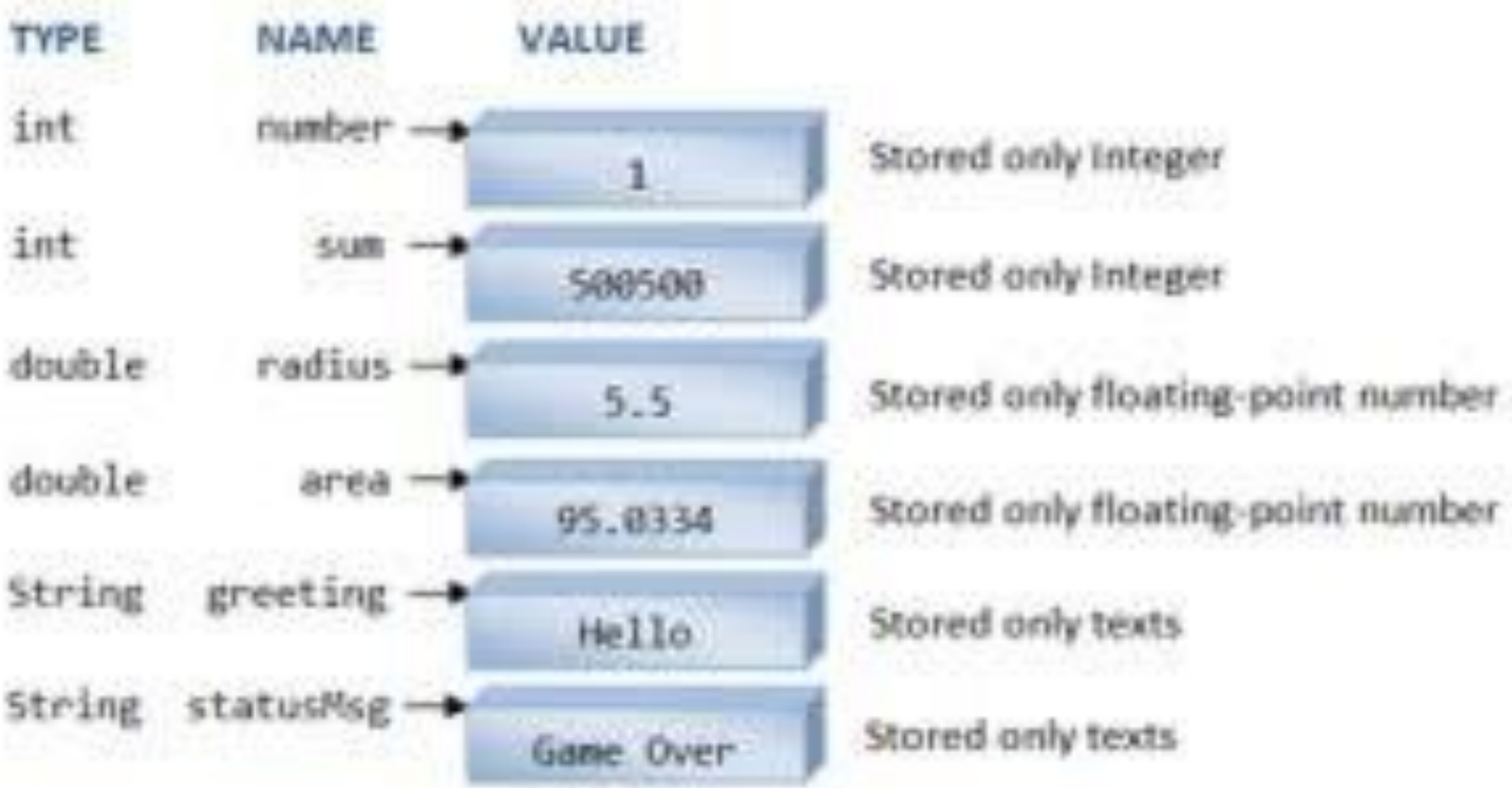
$2^3 = 8$

$2^4 = 16$

$2^5 = 32$

$2^6 = 64$

$2^7 = 128$


TYPE
NAME
VALUE
int
number
1
Stored only integer
int
sum
500500
Stored only integer
double
radius
5.5
Stored only floating-point number
double
area
95.0334
Stored only floating-point number
String
greeting
Hello
Stored only texts
String
statusMsg
Game Over
Stored only texts
A variable has a name, stores a value of the declared type.

How To Code???

- You need to download Arduino IDE (Integrated Development Environment).

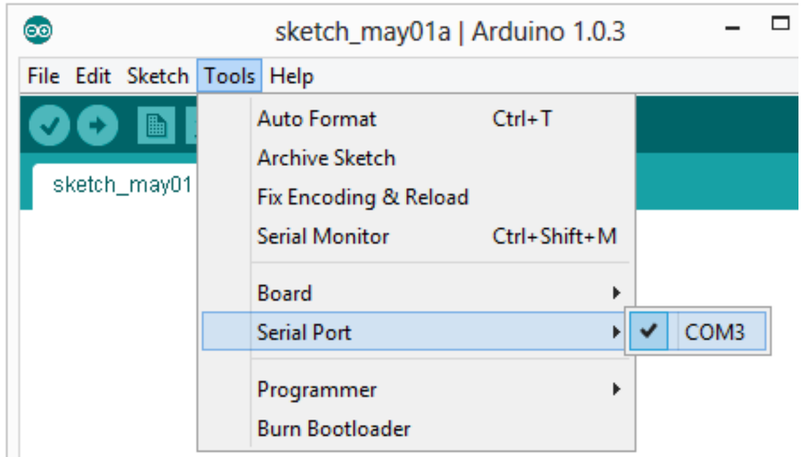
- Arduino IDE is available for all Mac, Windows, and Linux. There is also a direct to shell version (Arduino CLI, available on Github and an IOT version.

- Arduino IDE has Legacy (no dark mode available but can handle discontinued official hardware) and the new IDE (has dark mode and can handle newer official hardware).

How To Code???

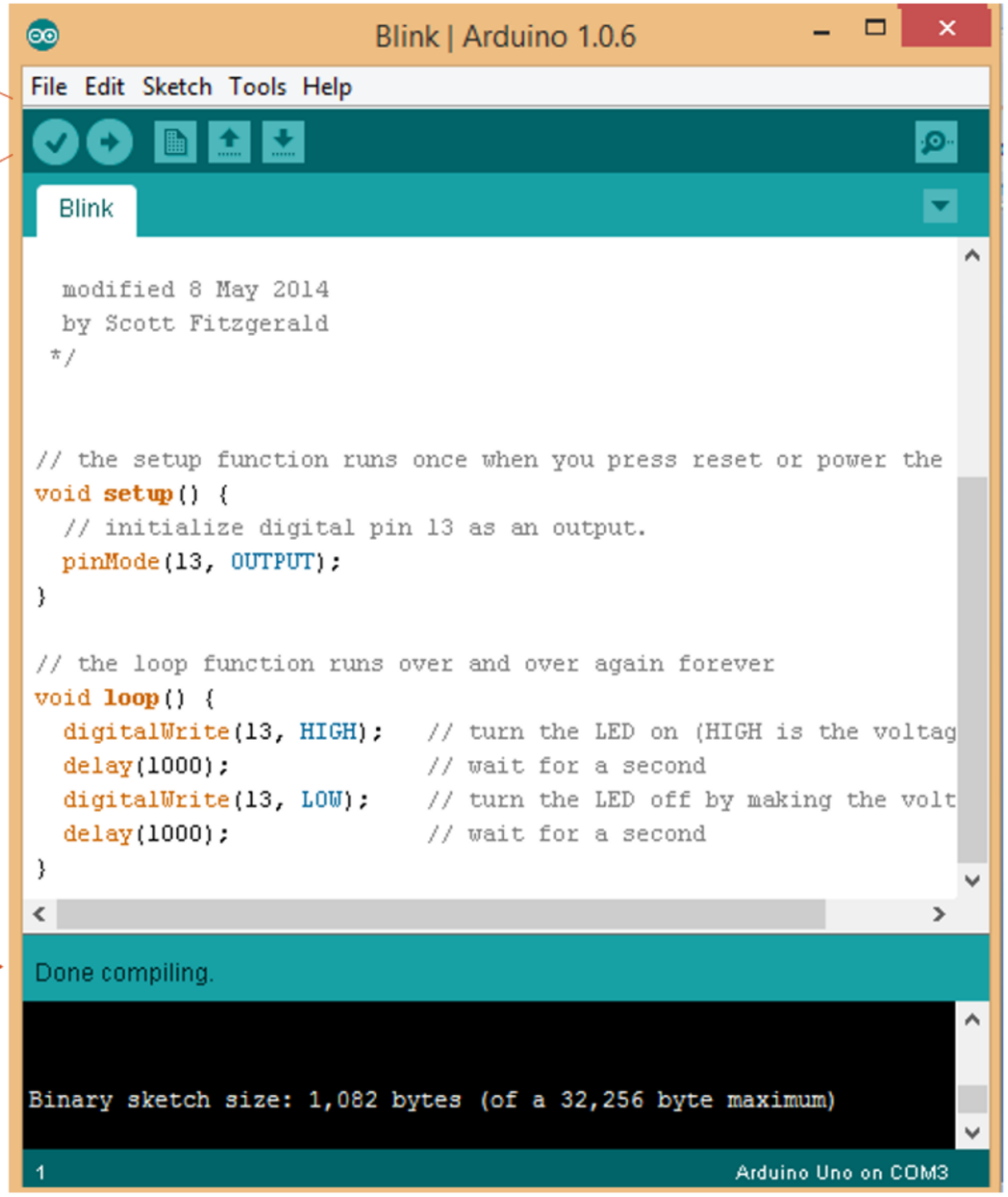
- You need to familiarize yourself with the IDE and what it can offer to your board (or other open-source boards) to hav it easily detected by your computer to install it.

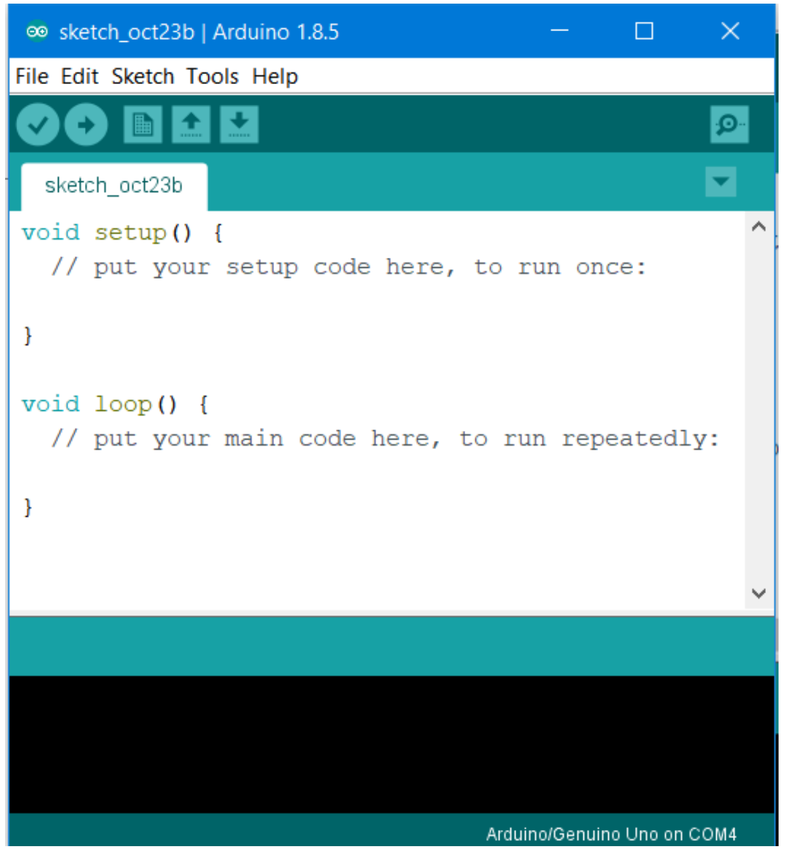
- To use transducers and sensors, make sure that it is compatible with Arduino or other MCU's like Arduino clones and Arduino Alternatives (i.e. Teensy, NodeMCU, ESP32, etc.)



Upload

Compile

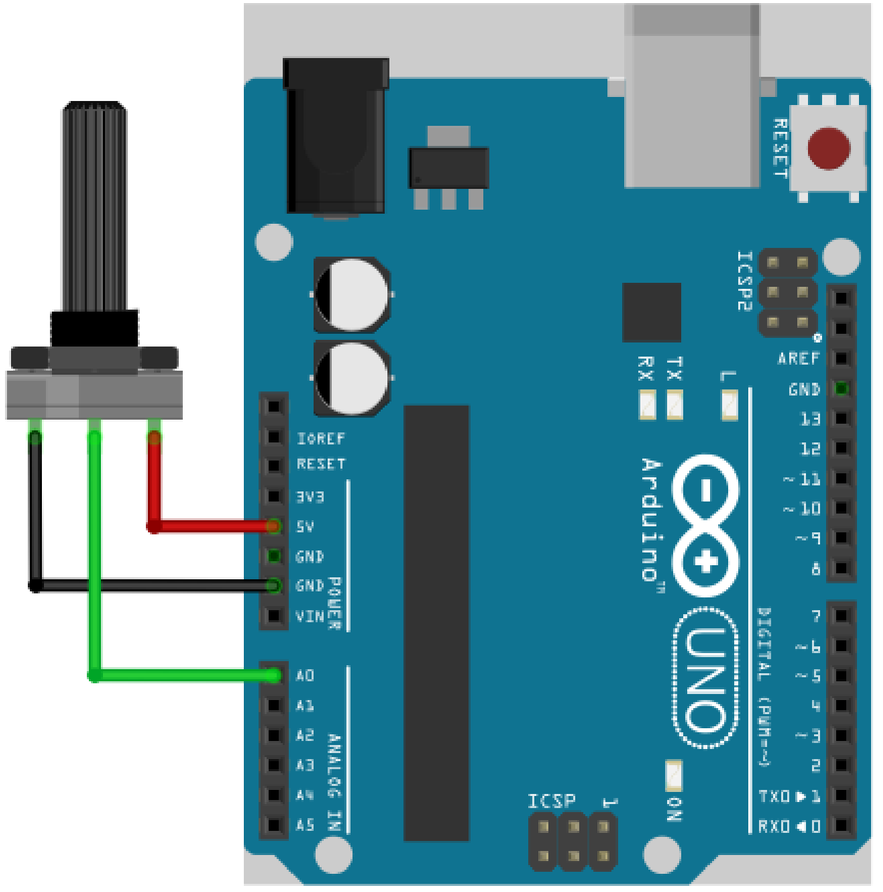


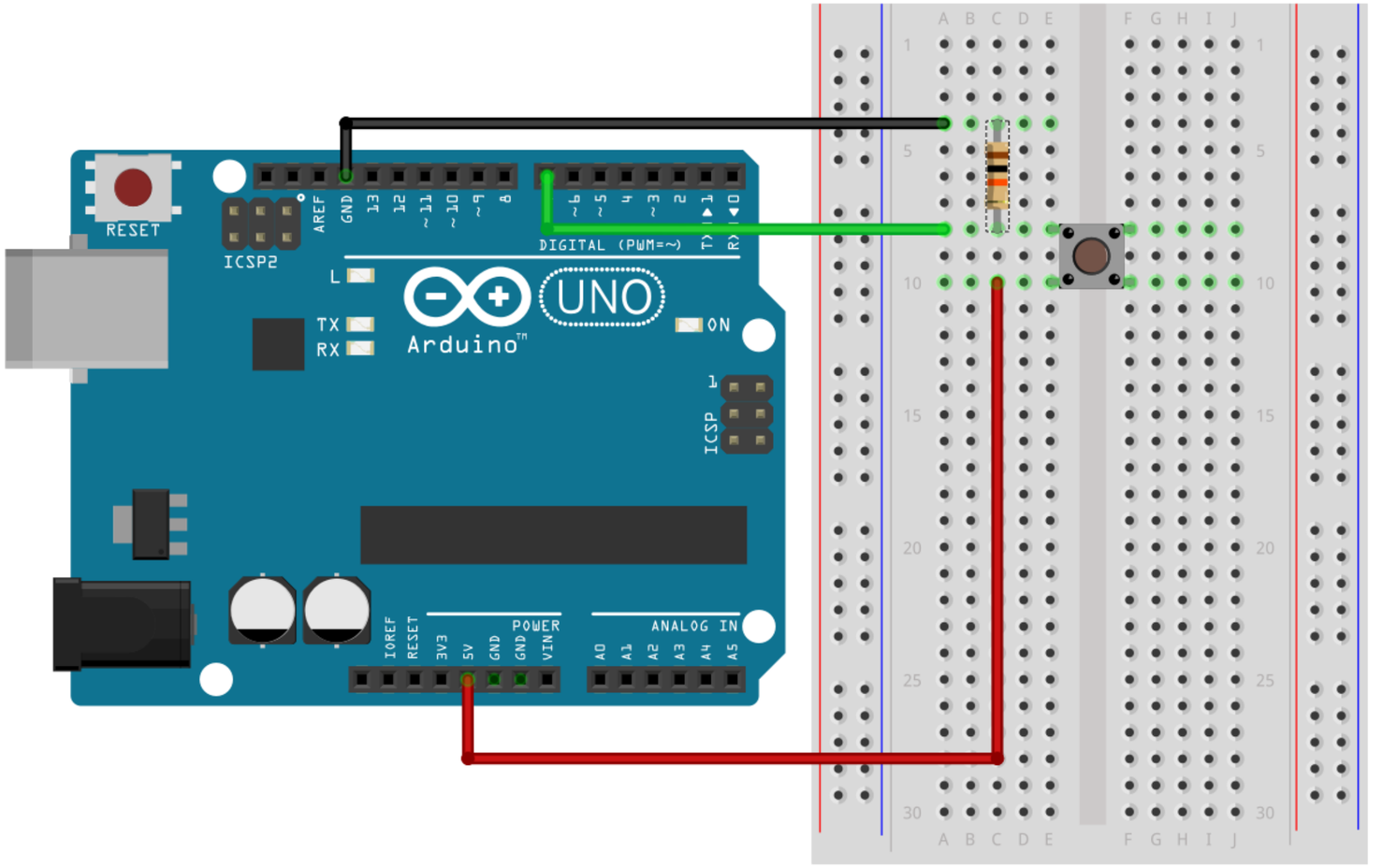


Status Message

Analog Versus Digital

- All Arduino signals are either Analog or Digital
- All computers including Arduino, only understand Digital
- It is important to understand the difference between Analog and Digital signals since Analog signals require an Analog to Digital conversion





"*sketch*" – a program you write to run on an Arduino board

"*pin*" – an input or output connected to something.

e.g. output to an LED, input from a knob.

"*digital*" – value is either HIGH or LOW.

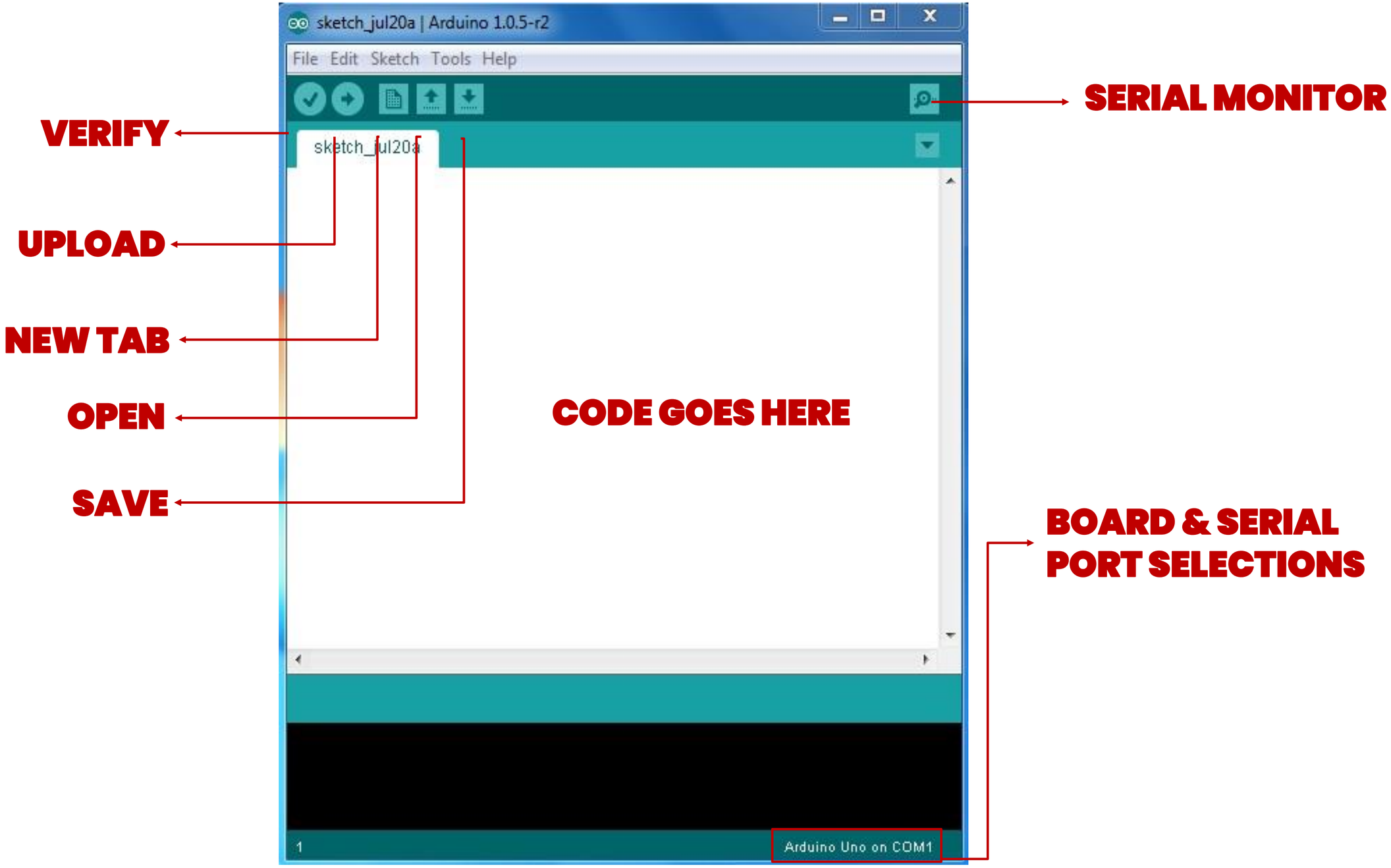
(aka on/off, one/zero) e.g. switch state

"*analog*" – value ranges, usually from 0-255.

e.g. LED brightness, motor speed, etc.

Serial But Not Killer...

- “Serial” because data is broken into bits, each sent one after another in a single wire.
- 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.


sketch_jul20a | Arduino 1.0.5-r2
File Edit Sketch Tools Help
sketch_jul20a
1
Arduino Uno on COM1
VERIFY
UPLOAD
NEW TAB
OPEN
SAVE
CODE GOES HERE
SERIAL MONITOR
BOARD & SERIAL PORT SELECTIONS

Eh, No Naptime?

- The Arduino is usually programmed with Embedded C, this is not necessarily similar to C, C++, or C#, but instead is a modified language of its own that is compatible mainly with Arduino boards and Arduino clones.

- The void setup part sets up the different transducers or sensors to initiate a working code, while the void loop part presents the code indefinitely.

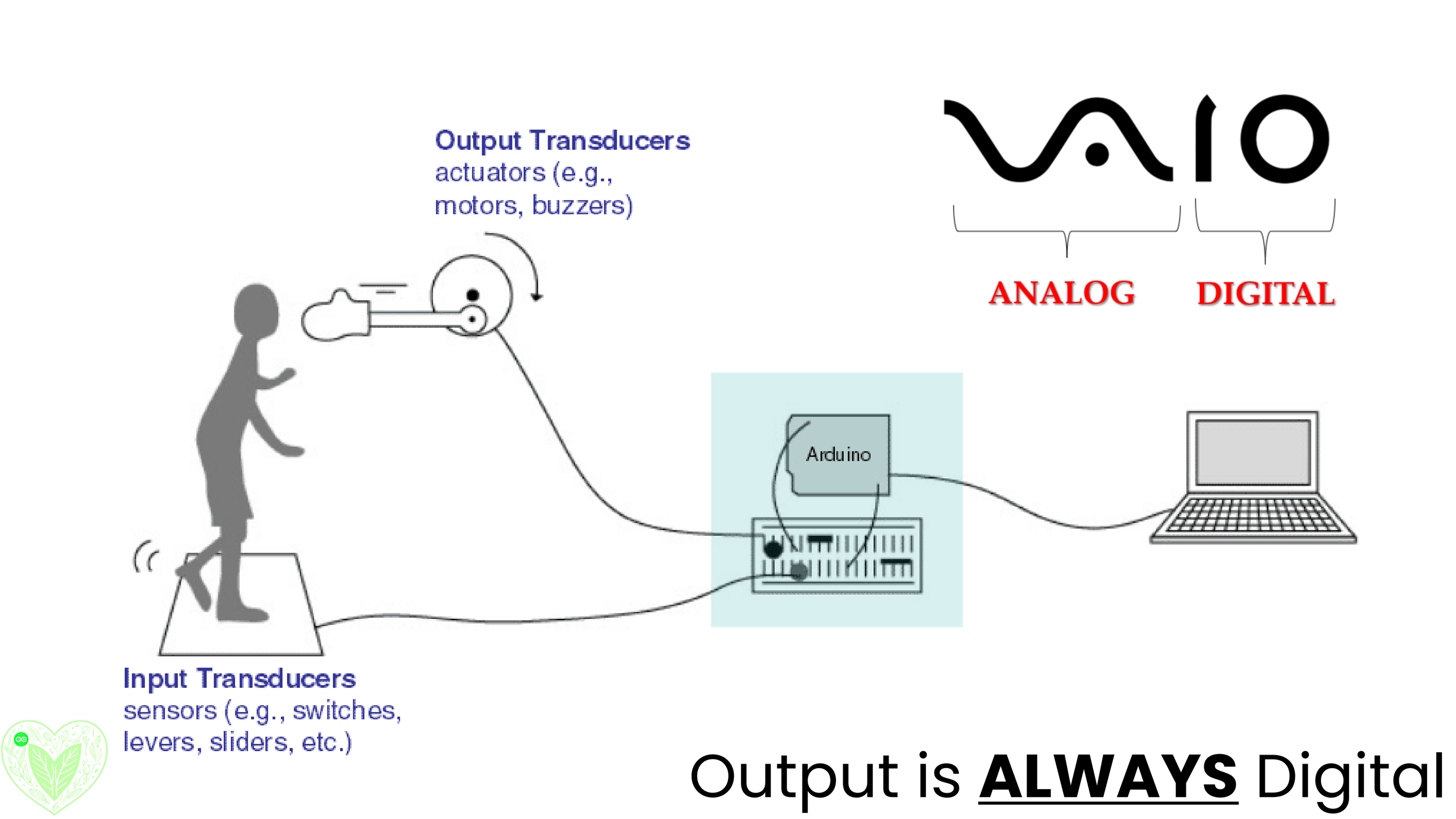
Eh, No Naptime?

- **The newest IDE doesn’t have all the files necessary to run a specific sensor, at this point you as the programmer should know where to find these files in order to run a specific sensor.**

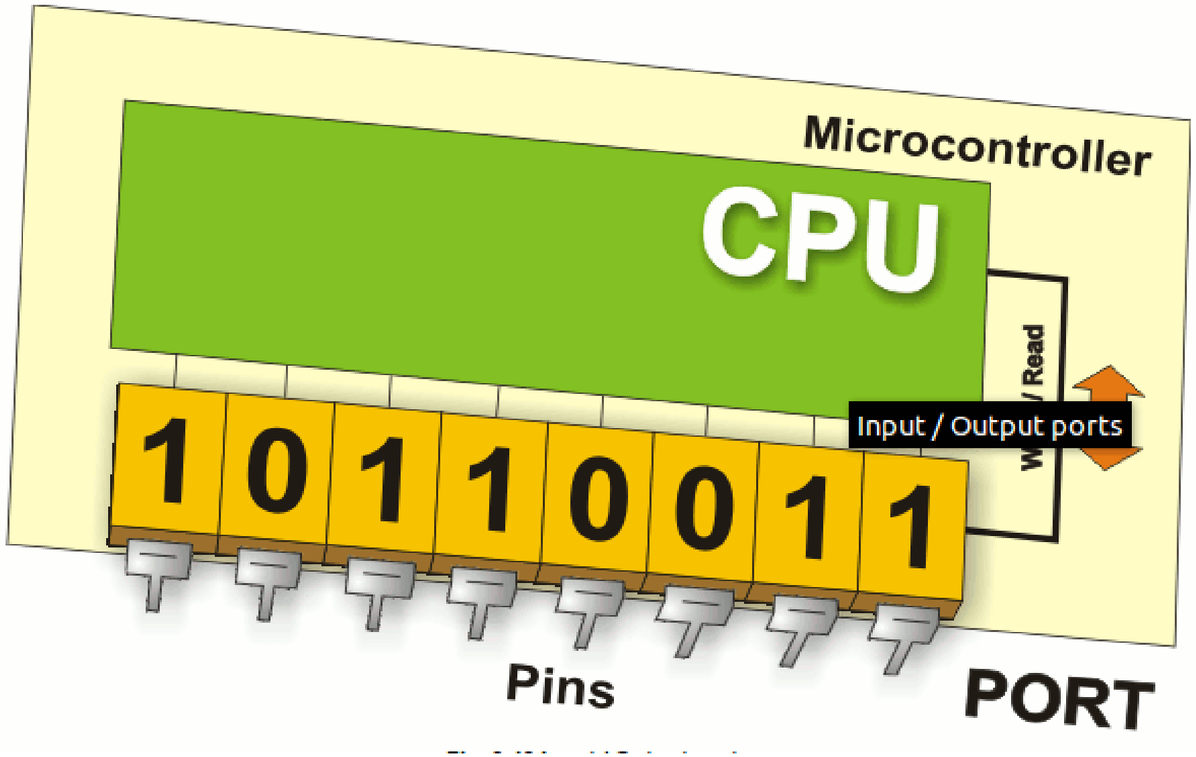
- **It is a good practice to be at least familiar with what the transducer or sensor can do before doing the coding part, like testing if the sensor in question works or not.**

Input Versus Output

- Everything is referenced from the perspective of the microcontroller.
- Inputs is a signal going into the board.
- Output is any signal exiting an electrical system.
- Almost all systems that use physical computing will have some form of output
- Often – Outputs include LEDs, a motor, a servo, a piezo element, a relay and an RGB LED



Output is **ALWAYS** Digital



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

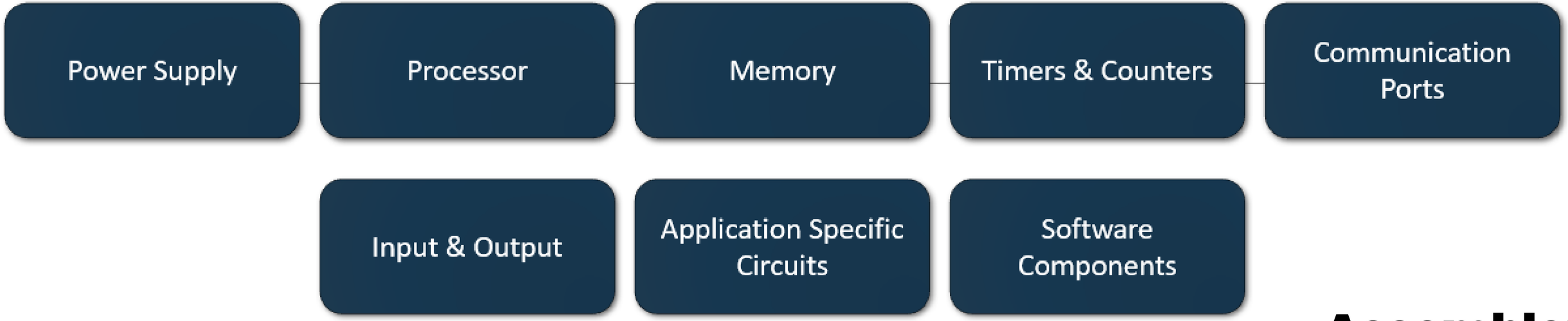
Electronic stuff
Output pins can provide 40 mA of current
Writing HIGH to an input pin installs a 20KΩ pullup

To output a signal that pretends to be Analog use this code:

analogWrite (*pinNumber, value*);

- Where pin is one of the analog output pins: 3, 5, 6, 9, 10, 11

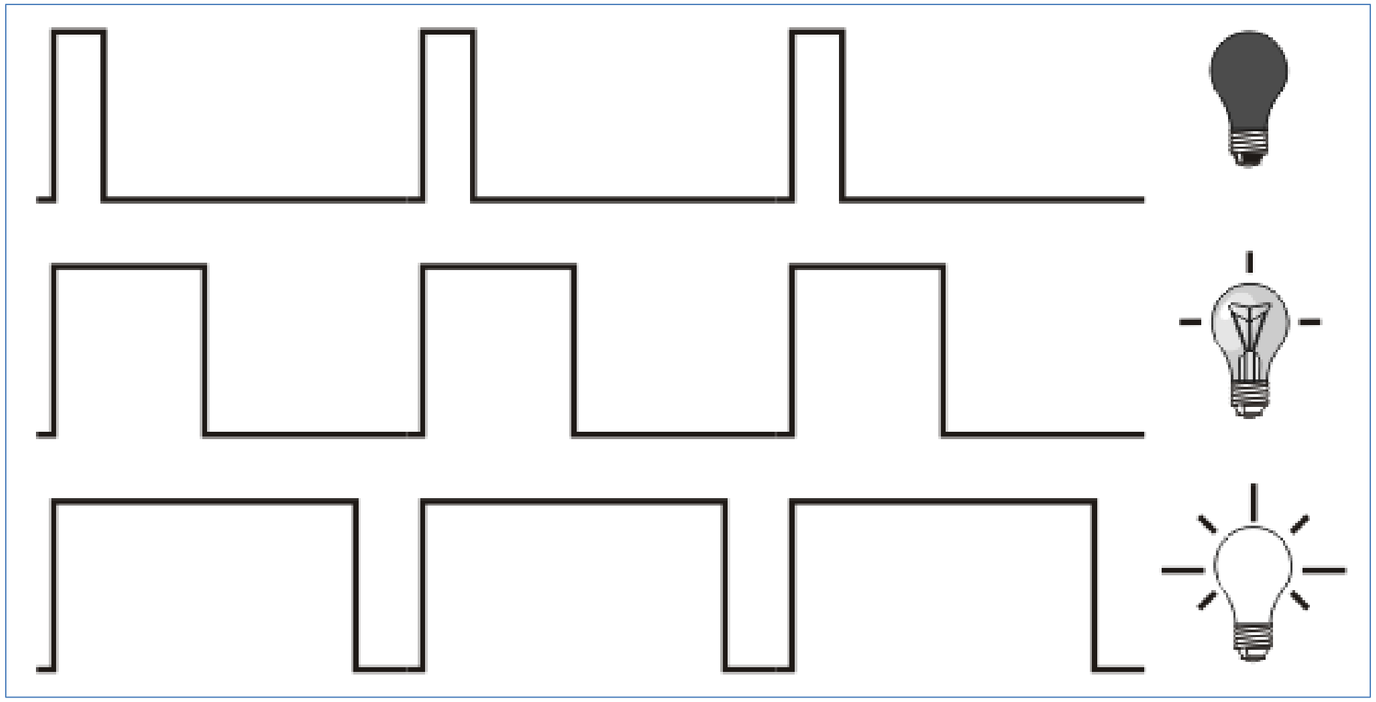
Where value is a number ranging from: 0 – 255.



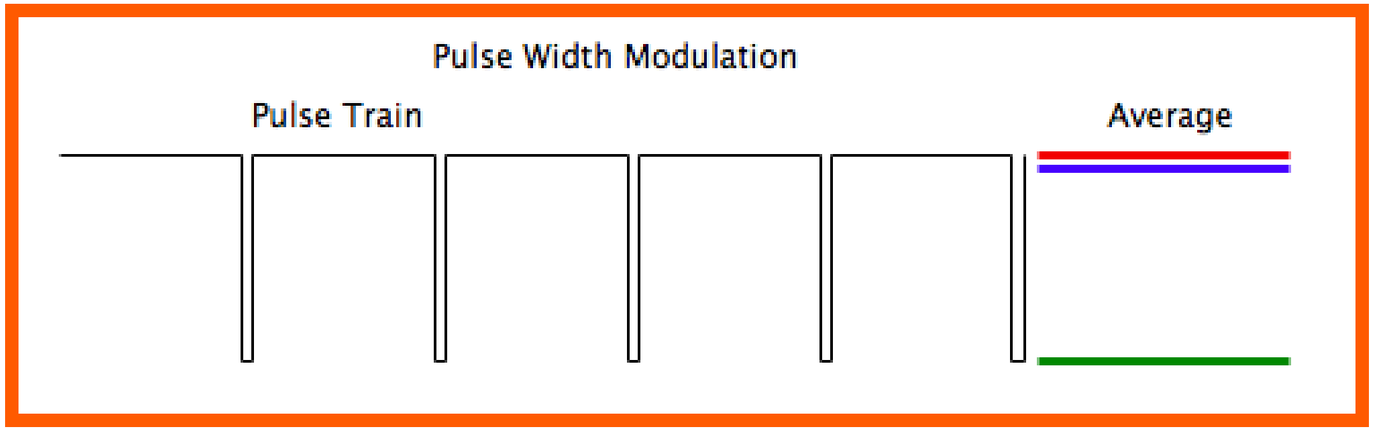
Embedded Systems

- **Assembler**
- **Emulator**
- **Debugger**
- **Compiler**

*"An **embedded system** is a computer system with a dedicated function within a larger mechanical or electrical system, often with real-time computing constraints. It is embedded as part of a complete device often including hardware and mechanical parts. Embedded systems control many devices in common use today."*



D: 0%



Pulse Width Modulation

Hackerman Time

- ***setup()* function**
  - Called when a sketch starts.
  - To initialize variables, pin modes, start using libraries, etc.
  - Will only run once, after each power-up or reset of the Arduino board.
- ***loop()* function**
  - Loops consecutively.
  - Code in the loop() section of the sketch is used to actively control the Arduino board.
- ***Commenting***
  - Any line that starts with two slashes (//) will not be read by the compiler, so you can write anything you want after it.

Hackerman Time

- ***pinMode()***
  - Instruction used to set the mode (INPUT or OUTPUT) in which we are going to use a pin.
  - Eg: pinMode (13, OUTPUT);
  - i.e. setting pin13 as output.

- ***digitalWrite()***
  - Write a HIGH or a LOW value to a digital pin.
  - Eg: digitalWrite (11, HIGH);
  - i.e. setting pin 11 to high.

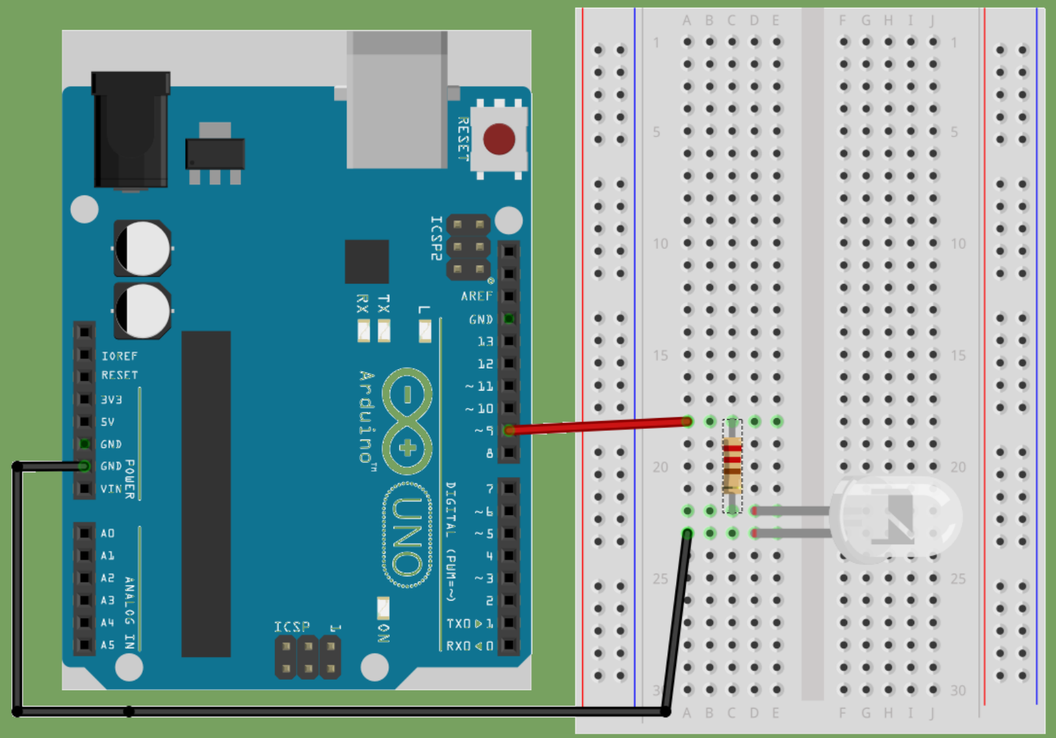
Hackerman Time

- ***digitalRead()***
  - Reads the value from a specified digital pin, either HIGH or LOW
  - Eg: int inPin=7;
    val = digitalRead(inPin);
  - ie. reads the value from inPin and assigns it to val.

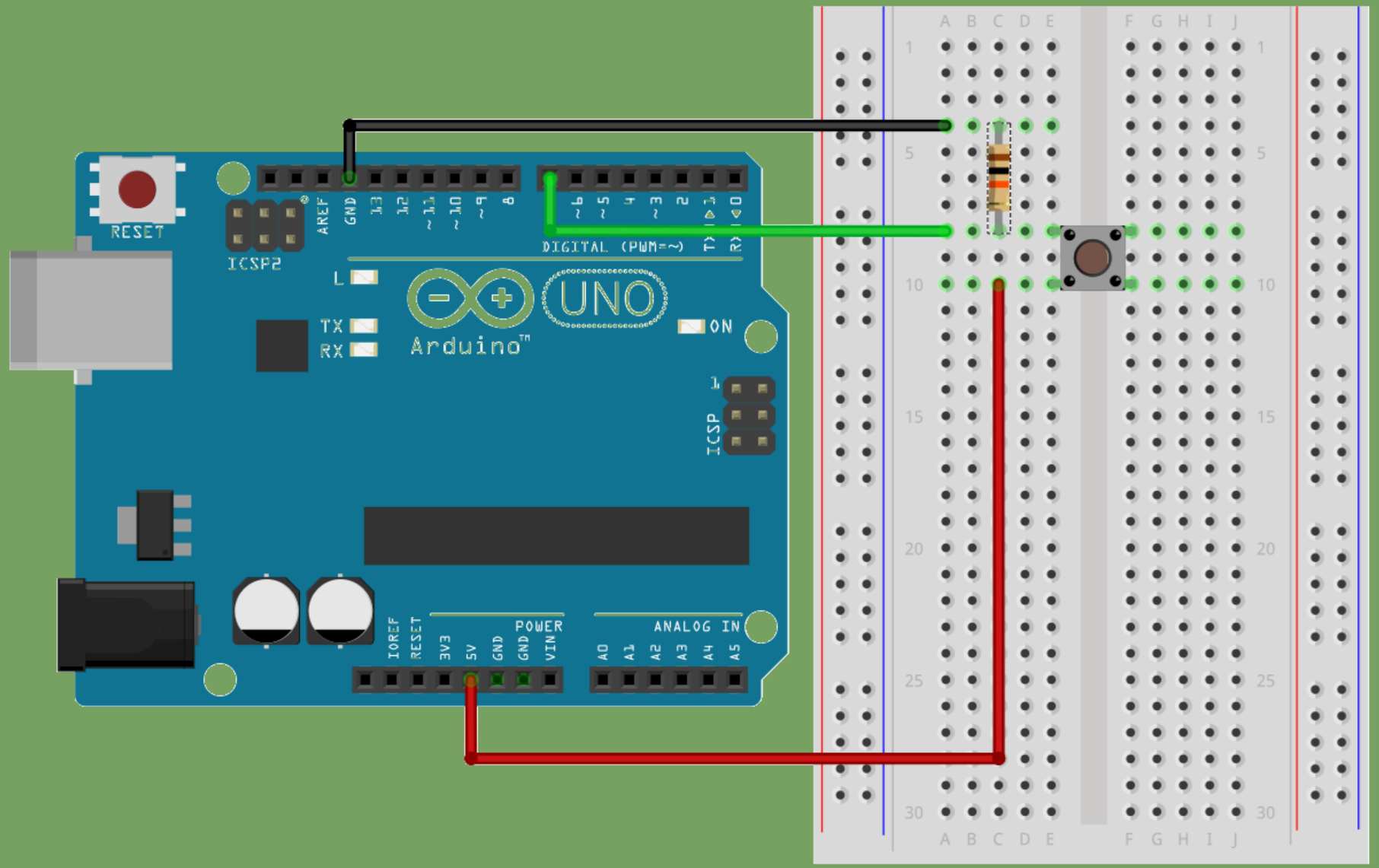
- ***delay()***
  - Pauses the program for the amount of time (in milliseconds) specified as parameter.
  - Eg: delay(1000);
  - i.e. waits for a second (1000 ms = 1 s)

Hackerman Time

Turn LED ON → Wait → Turn LED OFF → Wait → Rinse & Repeat



Digital Reading



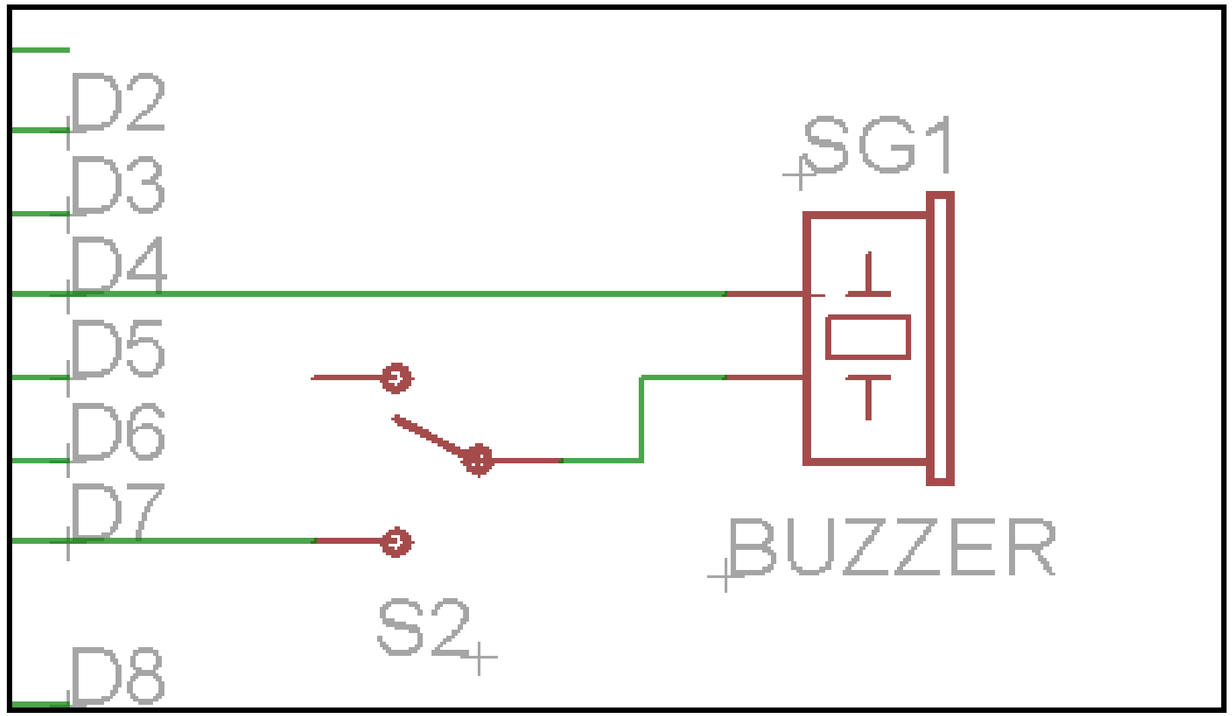
Digital Sensors

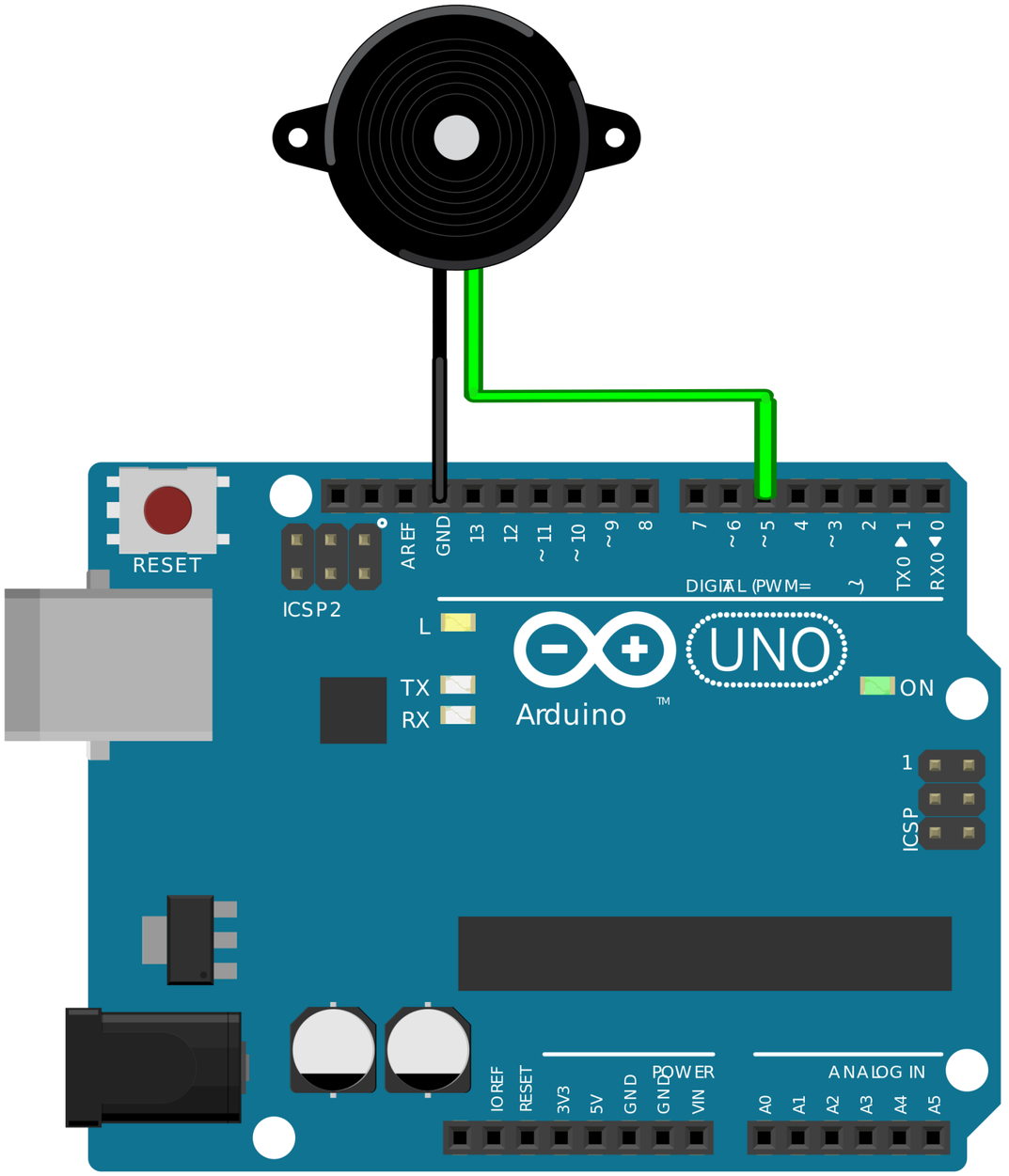
- Digital sensors are more straight forward than Analog
- No matter what the sensor there are only two settings: On and Off
- Signal is always either HIGH (On) or LOW (Off)
- Voltage signal for HIGH will be a little less than 5V on your Uno
- Voltage signal for LOW will be 0V on most systems

Make Some Noise

An example of a digital sensor is a BUZZER.

- **Code:** *tone(pin, freq, duration);*
- **pin** – the OUTPUT pin the buzzer is connected to
- **freq** – unsigned int (0 … 65,535)
- **duration** – unsigned long (0 … 2**32 - 1)





Buzzer for Arduino

Make Some Noise

- The Buzzer is connected between two digital pins.
- You must set both pins as OUTPUTs – pinMode(4, OUTPUT);
- pinMode(7, OUTPUT);

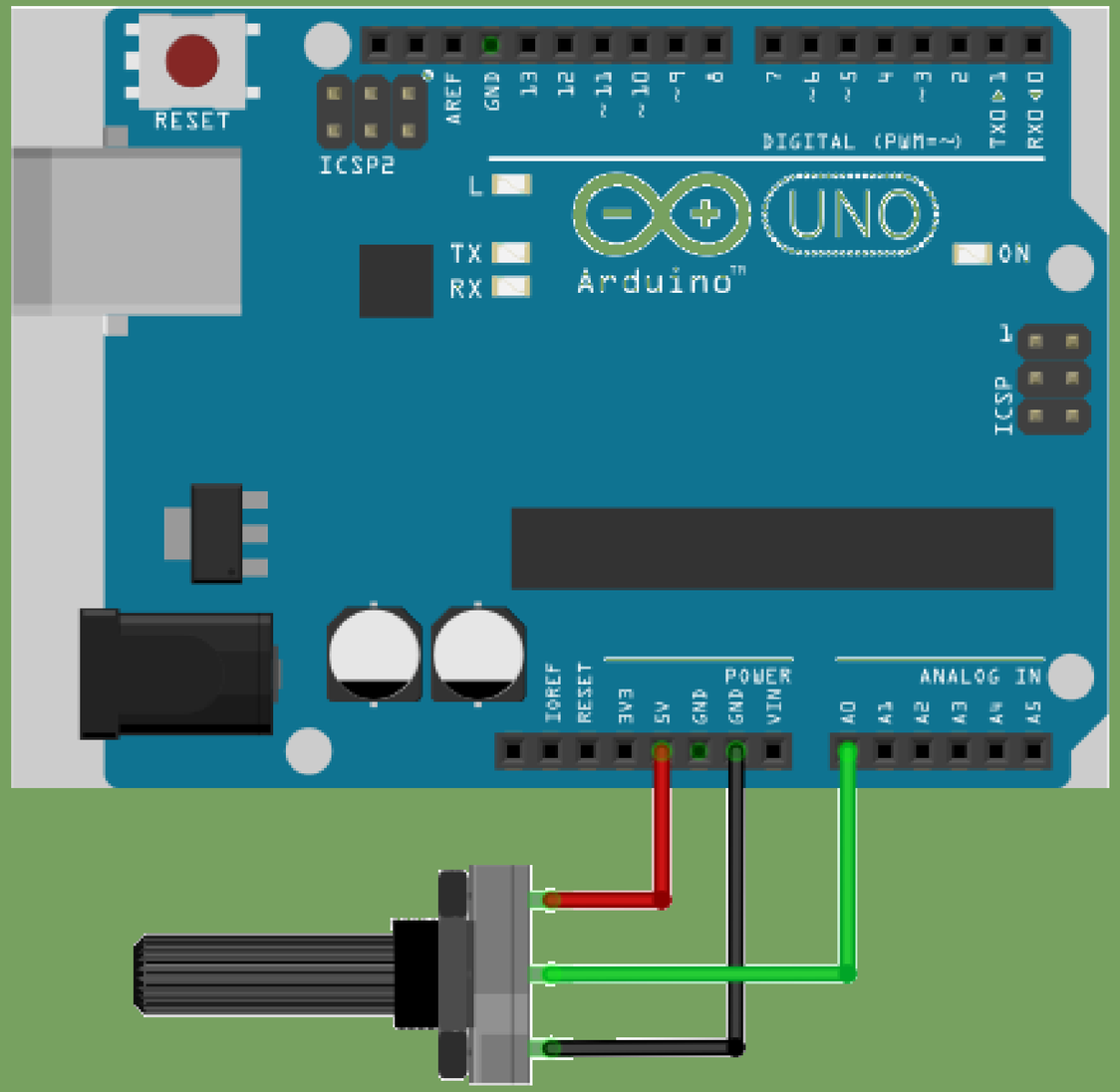
- Use tone(4, 440); to generate a 440 Hz sound.

Make Some Noise

Note	Frequency (Hz)
C_4	261
$C^{\#}_4/D^{b}_4$	277
D_4	293
$D^{\#}_4/E^{b}_4$	311
E_4	329
F_4	349
$F^{\#}_4/G^{b}_4$	369
G_4	392
$G^{\#}_4/A^{b}_4$	415
A_4	440
$A^{\#}_4/B^{b}_4$	466
B_4	493

Note	Frequency (Hz)
C_5	523
$C^{\#}_5/D^{b}_5$	554
D_5	587
$D^{\#}_5/E^{b}_5$	622
E_5	659
F_5	698
$F^{\#}_5/G^{b}_5$	739
G_5	783
$G^{\#}_5/A^{b}_5$	830
A_5	880
$A^{\#}_5/B^{b}_5$	932
B_5	987

Analog Reading

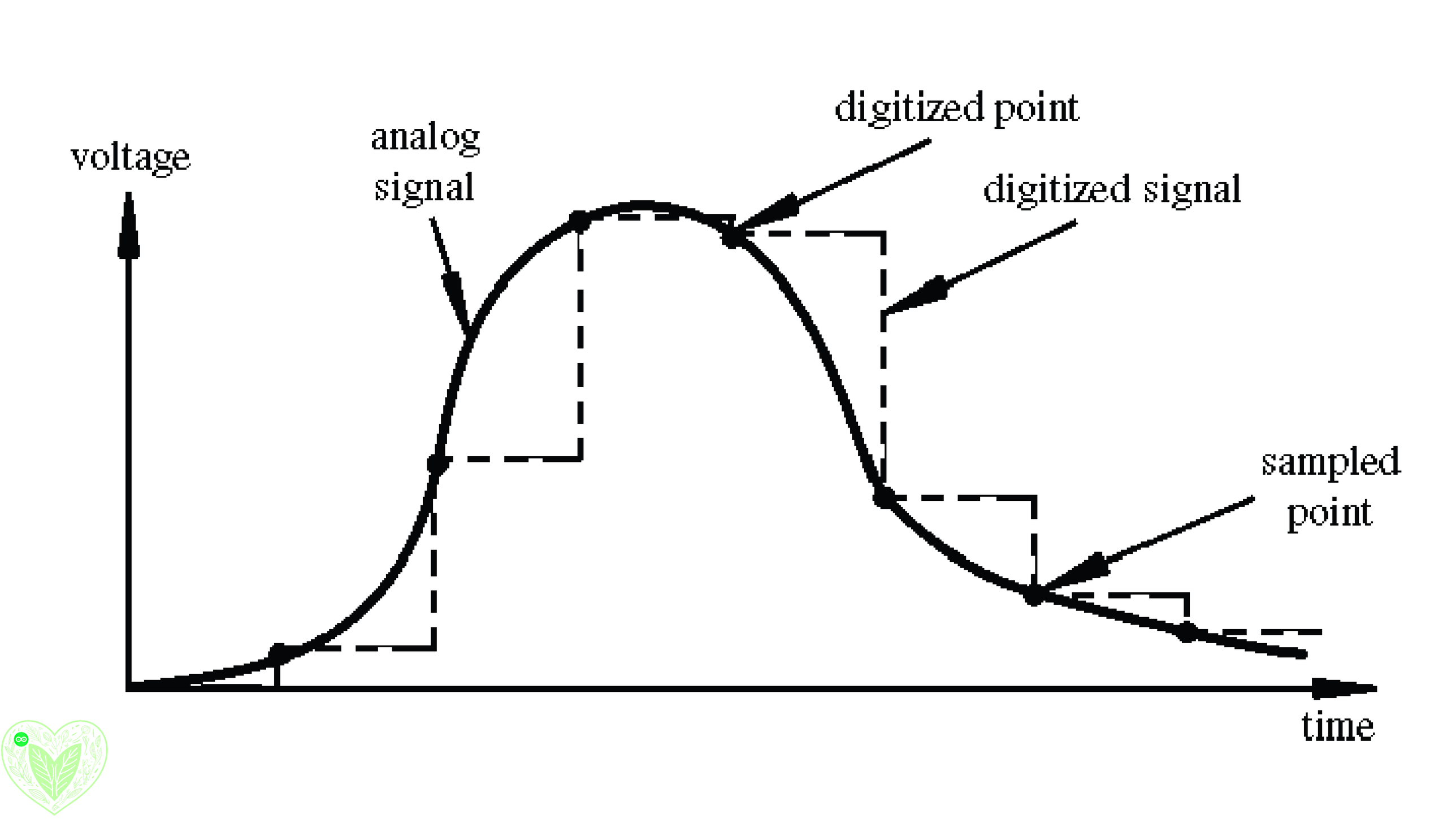


Analog Sensors

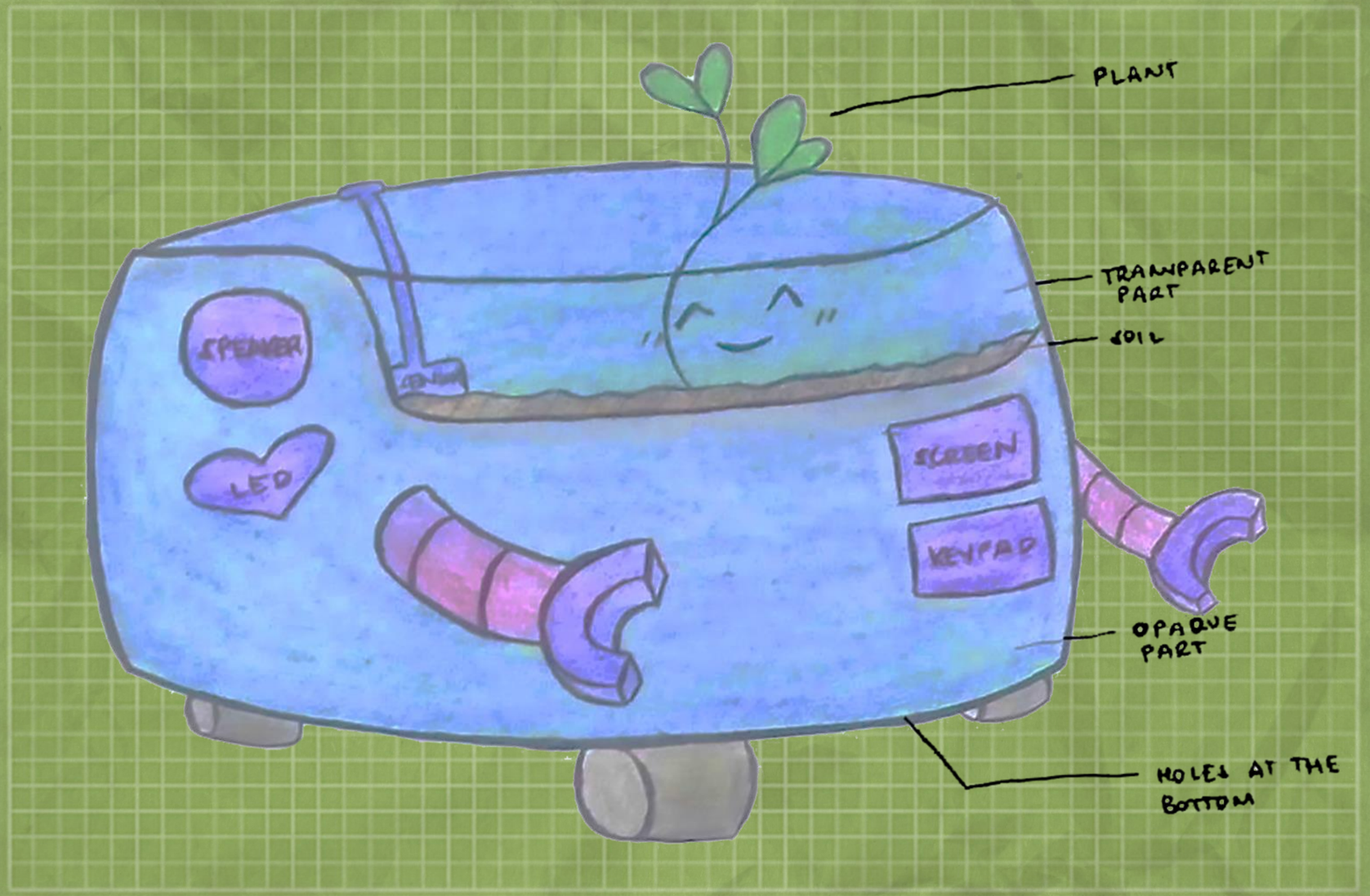
Sensors	Variables
Microphone	Sound, Volume
Photoresistor	Light Level
Potentiometer	Dial Position
Temp Sensor	Temperature
Flex Sensor	Bend
Accelerometer	Tilt / Acceleration

Analog Sensors

- Digital has two values: **on** and **off**
- Analog has many (infinite) values
- Computers don't really do analog, they ***quantize***
- Remember the 6 analog input pins---here's how they work


voltage
analog
signal
digitized point
digitized signal
sampled
point
time

Product Design



Project Roadmap

PLANNING PHASE: OCTOBER 2023

Brainstorming of Project HEART and its components using Arduino.

SIMULATION PHASE: NOVEMBER 2023

Simulation of Arduino components of Project HEART via Wokwi.

BUILD PHASE 1: NOVEMBER 2023

Building of Project HEART along with Arduino components.

Project Roadmap

BUILD PHASE 2: DECEMBER 2023

Building of Project HEART along with Arduino components.

TESTING PHASE: DECEMBER 2023

Testing of Project HEART along with Arduino components.

PRESENTATION: DECEMBER 2023

Final presentation of Project HEART and its capabilities.

Product Video: Blossoming Possibilities

Product pitch of Project HEART

Finished Product

PROFILES

Mary Heart Parena

3rd Year BS
Physics Student,
UP Baguio

Marc Andrie Bermundo

3rd Year BS
Physics Student,
UP Baguio

Danilo Earl John Sison

4th Year BS
Physics Student,
UP Baguio

Chloe Niña de la Cruz

4th Year BS
Physics Student,
UP Baguio

Contact Us

Mary Heart Jillbeth Parena
mlparena@up.edu.ph

Marc Andrie Bermundo
mmbermundo@up.edu.ph

Danilo Earl John Sison
dfsison@up.edu.ph

Chloe Niña de la Cruz
cydelacruz@up.edu.ph

FOR QUESTIONS OR CONCERNS

References

1. Barry, Jamie. “Everything You Need to Know About Protoboards.” Cool Components, 1 July 2021, https://coolcomponents.co.uk/blogs/news/everything-you-need-to-know-about-protoboards.
2. Braza, Jenn. “How Resistors Work.” Circuit Basics, 13 May 2020, https://www.circuitbasics.com/what-is-a-resistor/.
3. Dejan. “Ultrasonic Sensor HC-SR04 and Arduino - Complete Guide.” How To Mechatronics, 2 Feb. 2022, https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/.
4. “DHT11–Temperature and Humidity Sensor.” Components101, https://components101.com/sensors/dht11-temperature-sensor. Accessed 1 Nov. 2023.
5. “How to Make an Anemometer: 7 Steps (with Pictures).” wikiHow, https://www.wikihow.com/Make-an-Anemometer. Accessed 1 Nov. 2023.
6. Khan, Hamza. “Arduino Water Level Sensor.” Datasheet Hub, 21 June 2023, https://www.datasheethub.com/arduino-water-level-sensor/.
7. Lab, Microcontrollers. “MB102 Breadboard Power Supply Module - How to Use It.” Microcontrollers Lab, 15 Feb. 2021, https://microcontrollerslab.com/mb102-breadboard-power-supply-module-pinout-and-how-to-use-it/.
8. Makerspaces.com. “How To Solder: A Complete Beginners Guide.” Makerspaces.Com, 11 Apr. 2017, https://www.makerspaces.com/how-to-solder/.

References

9. “Micro SD Card Adapter Module.” Components101, https://components101.com/modules/micro-sd-card-module-pinout-features-datasheet-alternatives. Accessed 1 Nov. 2023.
10. “Philippines: Land Area Used for Agricultural Crop Cultivation 2022.” Statista, https://www.statista.com/statistics/1045556/land-area-used-for-agricultural-crop-cultivation-philippines/. Accessed 1 Nov. 2023.
11. seotechwriter. “What Are Jumper Wires: Colour, Types and Uses.” Wiltronics, 14 Apr. 2022, https://www.wiltronics.com.au/wiltronics-knowledge-base/what-are-jumper-wires/.
12. “Soil Moisture Sensor Module.” Components101, https://components101.com/modules/soil-moisture-sensor-module. Accessed 1 Nov. 2023.
13. “Sprout.” Brawl Stars Wiki, https://brawlstars.fandom.com/wiki/Sprout. Accessed 1 Nov. 2023.
14. What Is an Arduino Uno R3 Board? - Hand Tools for Fun. 11 Nov. 2020, https://handtoolsforfun.com/what-is-an-arduino-uno-r3-board/.
15. What Is an SD Card? https://www.computerhope.com/jargon/s/sdcard.htm. Accessed 1 Nov. 2023.
16. “World Bank Open Data.” World Bank Open Data, https://data.worldbank.org. Accessed 1 Nov. 2023.