The Internet of Things:
Prototyping Boards & Languages

Course website: http://site.unibo.it/iot

Prof. Luciano Bononi
luciano.bononi@unibo.it

Prof. Marco Di Felice
marco.difelice3@unibo.it

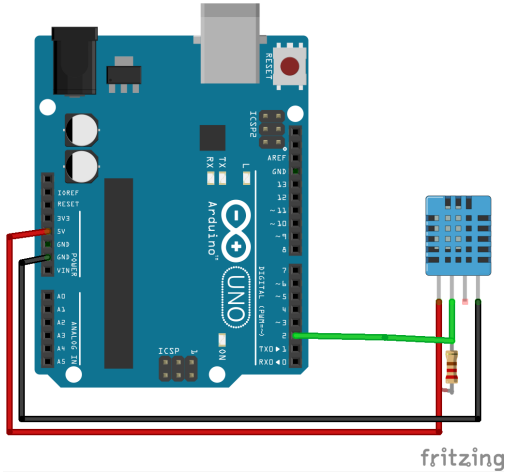
Luca Sciullo
luca.sciullo@unibo.it

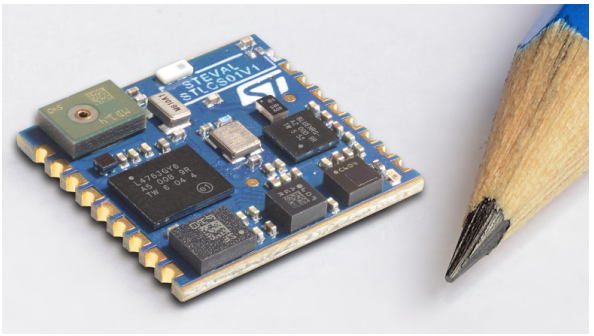
MASTER DEGREE IN COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, UNIVERSITY OF BOLOGNA, ITALY

Smart Things

- IoT world is made by **smart things**
- A smart thing is a physical object digitally augmented with one or more of the following:
  - (smart) Sensors (temperature, light, motion, and so on)
  - (smart) Actuators (displays, sound, motors, and so on)
  - Computation (can run programs and logic)
  - Communication interfaces (wired or wireless)

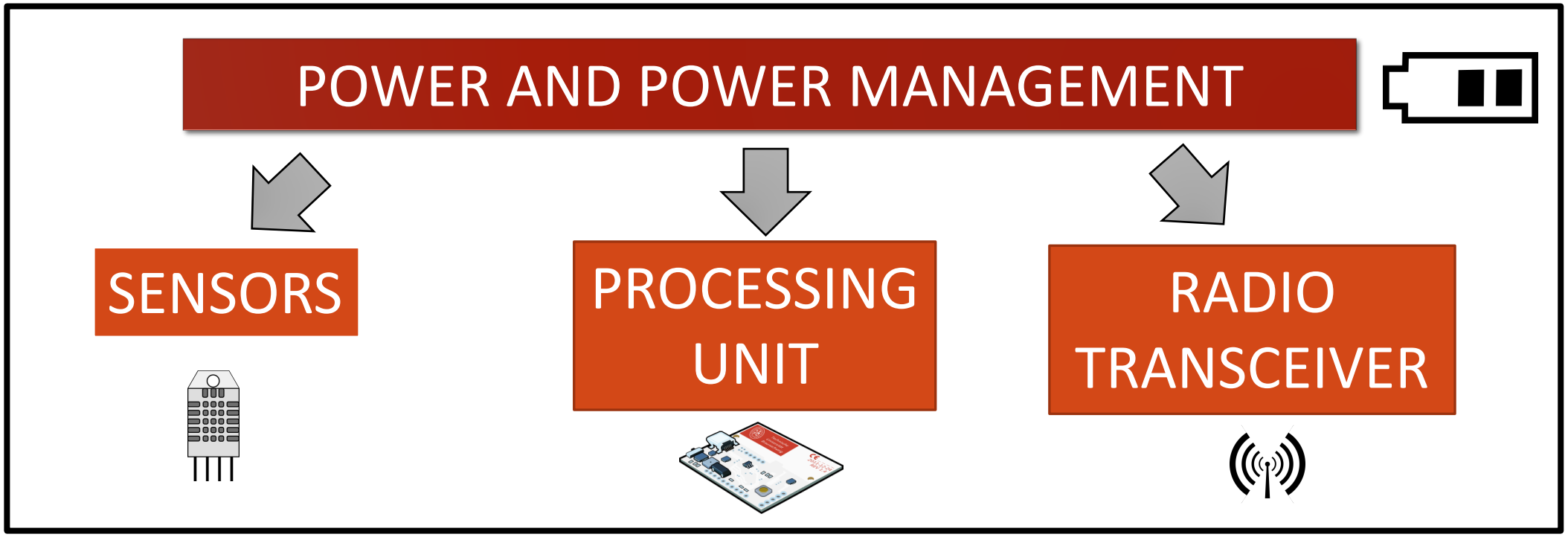
Smart Things





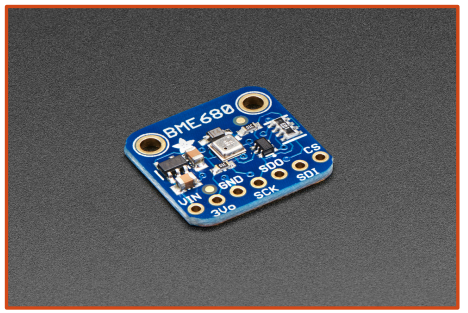
Smart Things

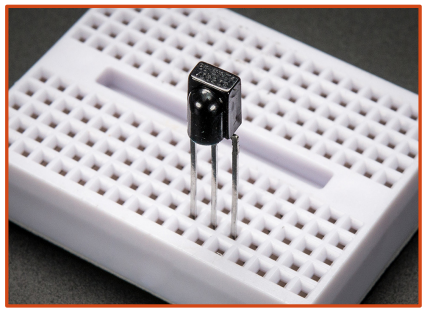
✧ Architecture of a **Smart Object**



Smart Sensors

- A sensor is a device that is able to detect events or changes in its physical environment.
- A smart sensor is a device capable of measuring analog inputs from the physical environment and making them digital by using some built-in resources

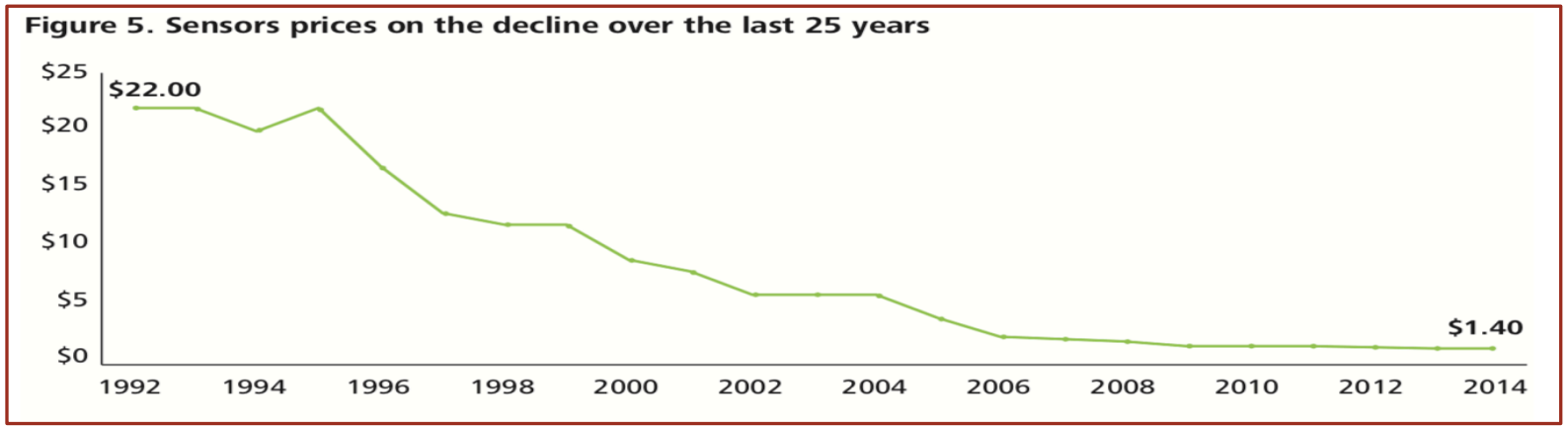




https://www.adafruit.com/

[3] https://www.figaro.co.jp/en/technicalinfo/miniaturization.html

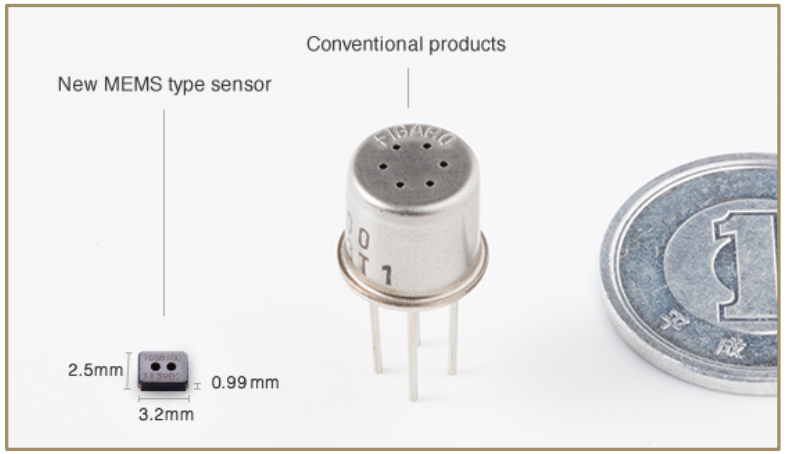
Smart Sensors

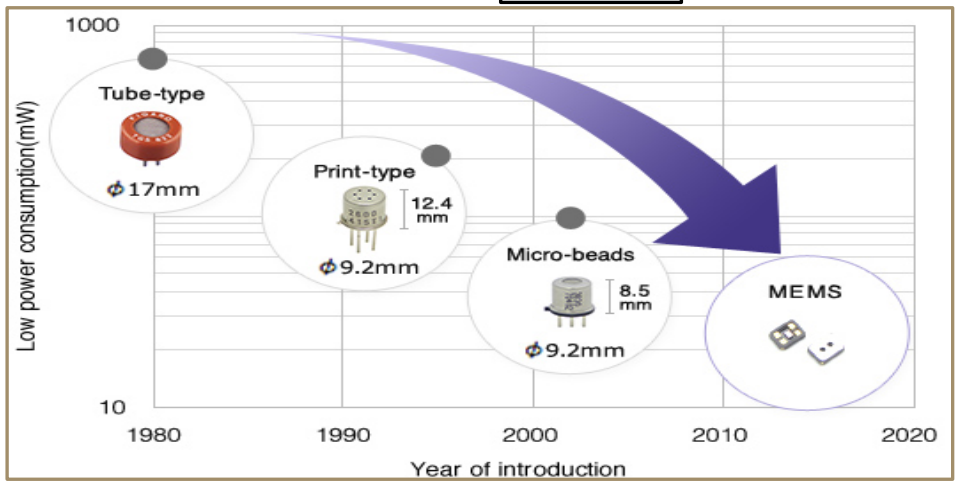


Sensors are getting **cheaper** and **cheaper** …

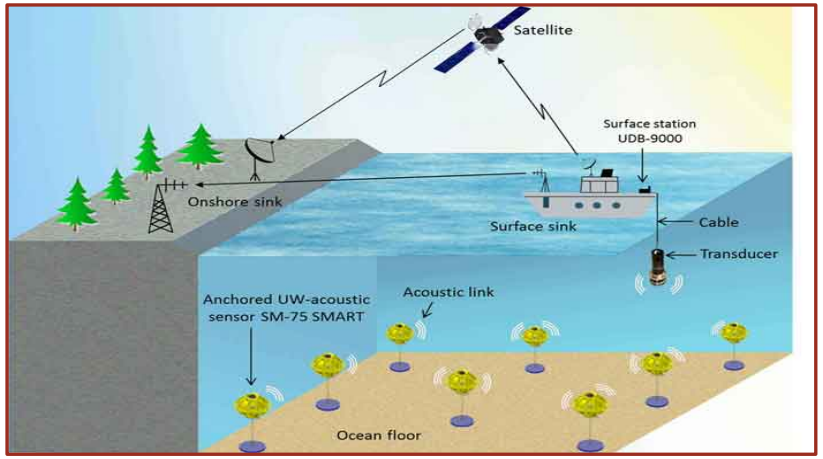
Source: [3]

Sensors are getting **smaller** and **smaller** …

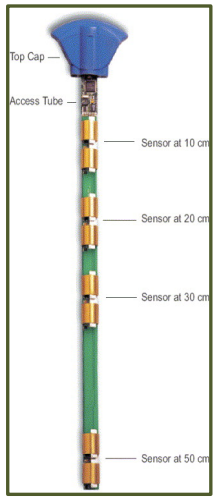


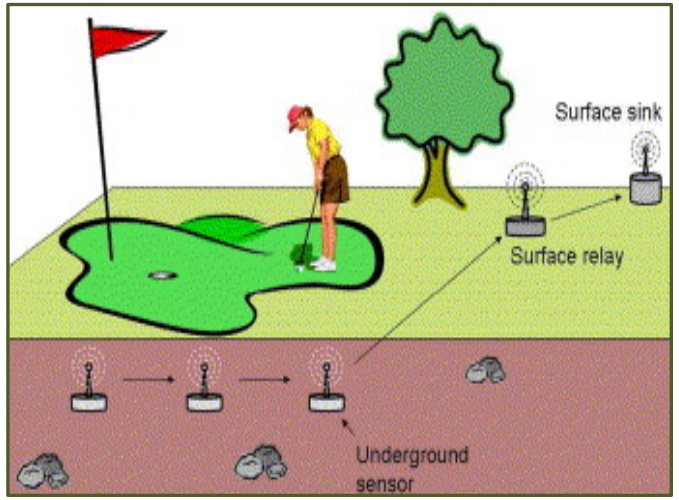


Smart Sensors

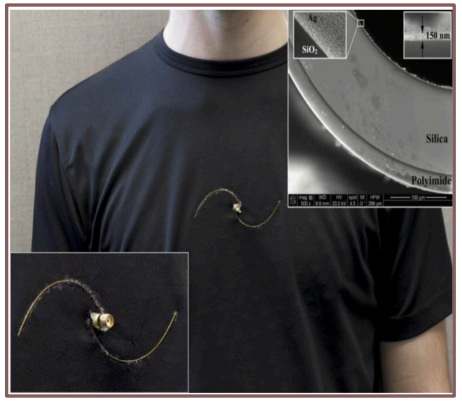


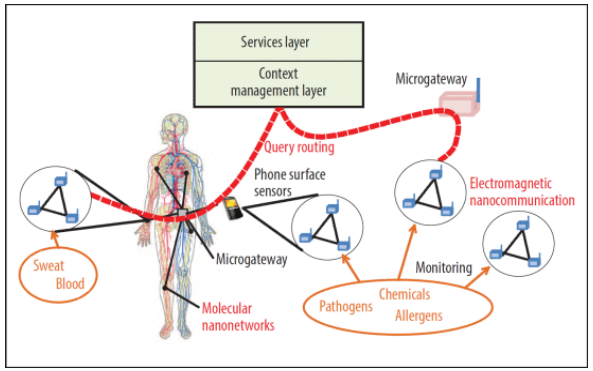
UNDERWATER **ACUSTIC** SENSOR NETWORKS





UNDERGROUND SENSOR NETWORKS





WEARABLE SENSORS
+ NANOSENSORS

Source: http://www.ece.neu.edu/wineslab/

[3] https://www.figaro.co.jp/en/technicalinfo/miniaturization.html

Smart Sensors

✧Btw, **current smartphones** embed a multitude of sensors that provide **context-aware information** about users.

SAMSUNG S9

- ✧ Iris (motion) sensor
- ✧ Pressure sensor
- ✧ Accelerometer
- ✧ Barometer
- ✧ Gyroscope
- ✧ Geomagnetic sensor
- ✧ Hall sensor
- ✧ HR sensor
- ✧ Proximity sensor
- ✧ RGB Light sensor
- ✧ …

MINING & LEARNING TECHNIQUES

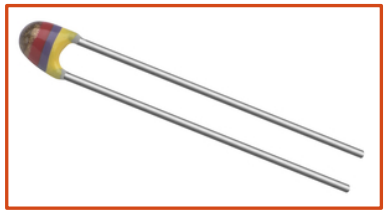
- ✧ 3D Indoor localization
- ✧ Transportation mode detection
- ✧ Human activity detection
- ✧ Human presence detection
- ✧ Health condition detection
- ✧ Gesture tracking
- ✧ …

Smart Sensors

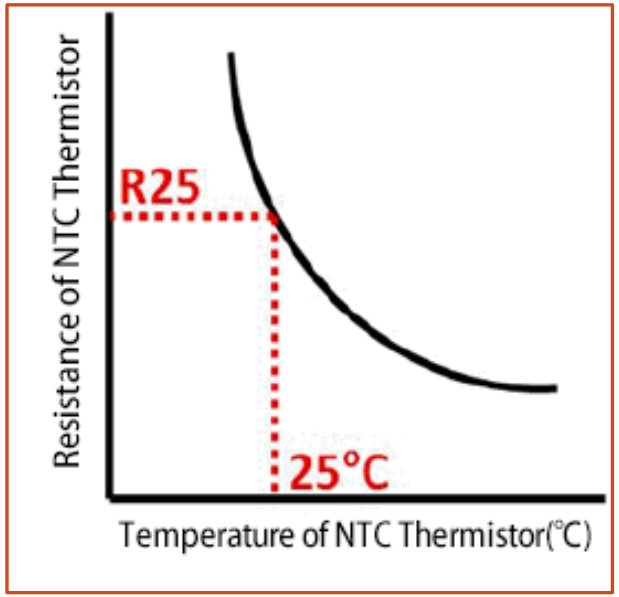
✧**Sensors** are mostly kinds of ***transducers***, i.e. they convert one form of energy (electrical or not) into another (electrical or not).

EXAMPLE: TEMPERATURE SENSOR

Negative Temperature Coefficient →Variable resistor that changes its resistance with change of the temperature: the resistance decreases with increase of the temperature.



Source: https://home.roboticlab.eu/

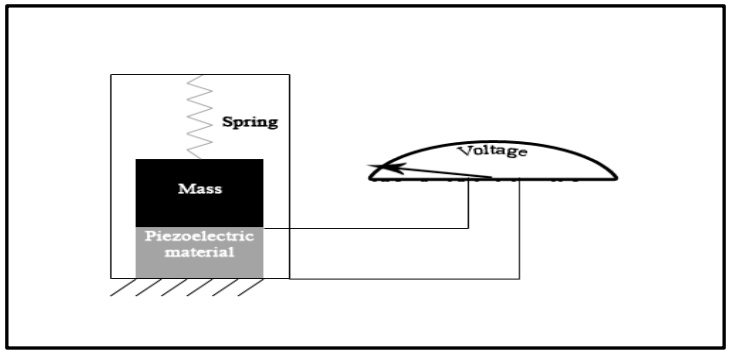


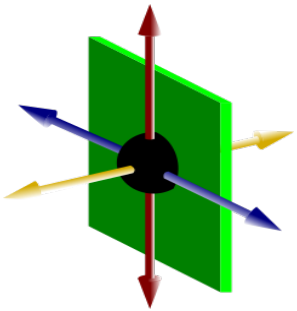
Smart Sensors

✧**Sensors** are mostly kinds of ***transducers***, i.e. they convert one form of energy (electrical or not) into another (electrical or not).

EXAMPLE: ACCELEROMETER SENSOR

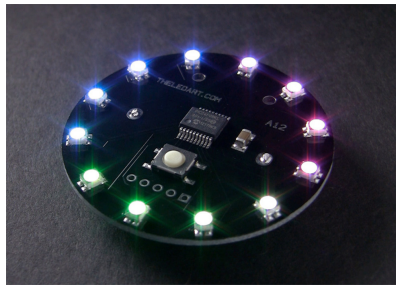
- ✧ Accelerometers are electromechanical devices that sense either static or dynamic forces of acceleration.
- ✧ Accelerometers can measure acceleration on 1,2 or 3 axes.
- ✧ Acceleration is derived by the displacement of an **internal spring** (Inertial Equation), often measured via capacitive plates.



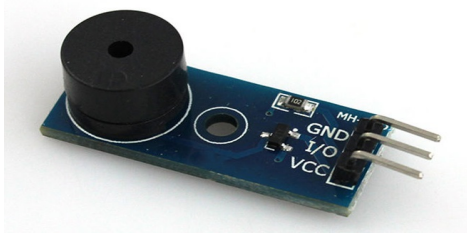


Smart Actuators

https://www.adafruit.com/



- An actuator is a device that converts energy into motion.
- A smart actuator is a device capable of turning digital inputs into physical actions.



http://www.utteducation.com

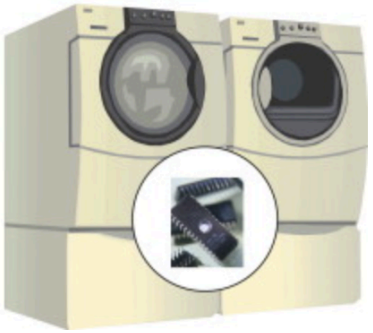
[1] https://www.sciencedirect.com/topics/computer-science/embedded-systems

Embedded Systems

- Integrated Microprocessor-based **hardware system** + **software/firmware** that is designed to perform a dedicated function/specific purpose.

Some examples:

- Industrial systems
- Home appliances
- Avionics
- Network devices (e.g. Routers, switches)
- Cache machines
- …



Source: [1]

[1] https://www.sciencedirect.com/topics/computer-science/embedded-systems

Embedded Systems

- **Embedded systems programming** often poses some unique challenges (not so common in general-purpose programming):
  - Real-time vs non-real time systems
    - ✓ Hard-real time vs soft real-time vs firm real-time
  - Constrained resources (CPU, power battery, …)
  - Limited memory for data loading and data operations
  - Limited memory for storing the program source code
  - Limited multitasking (i.e. thread/process support)

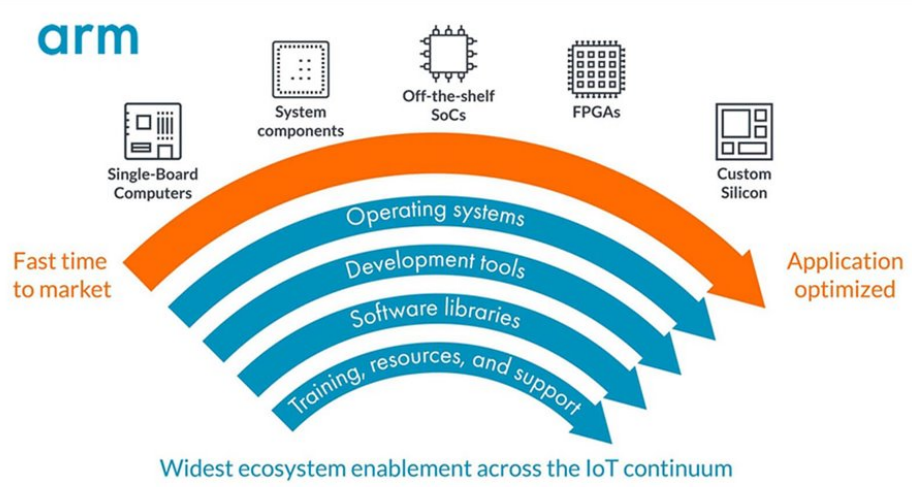
WE WILL PROVIDE EVIDENCES OF THESE CHALLENGES THROUGH A DEMO

[1] https://community.arm.com/iot/b/internet-of-things/posts/the-arm-internet-of-things-iot-continuum

IoT Prototyping Boards

Hardware platforms that are most commonly used to build **prototypes** of IoT projects and IoT embedded systems.

- ❑ Validate the feasibility of a product
- ❑ Build a proof-of-concept
- ❑ Do-It-yourself (DIY) projects
- ❑ Pre-production phase of a new product
  - ✓ Choose **hardware** components
  - ✓ Deploy and **test** the software

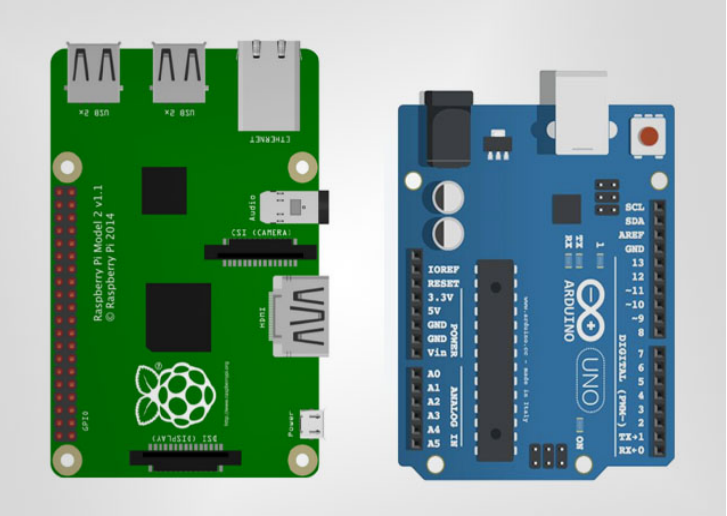


Source: [1]

IoT Prototyping Boards

IoT Prototyping boards can be often be classified according to their hardware/software characteristics into **two main families**:

- ❑ Microcontroller-based boards
- ❑ Single-board computers



https://mybroadband.co.za

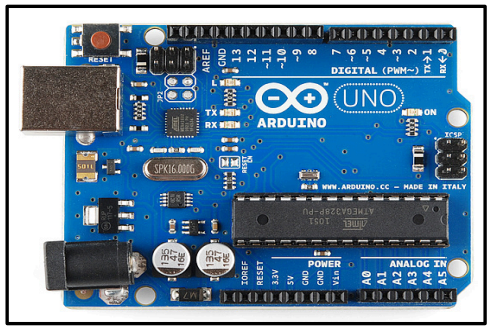
Microcontroller Boards

It is a system on a chip (SoC) that contains processing cores, RAM and EPROM for the storage of custom programs that are executed on the microcontroller. It is a PCB with added circuitry.

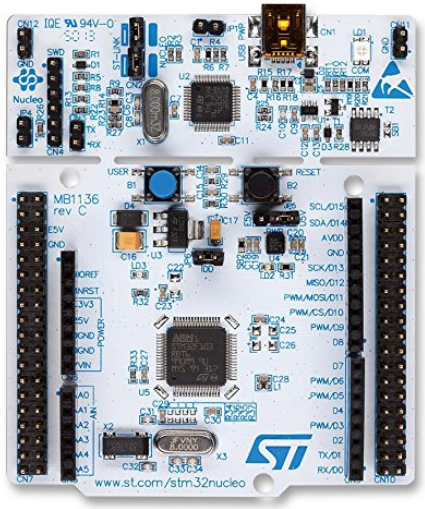
- Available on the market since 1975
- Often composed by a single CPU + RAM memory + ROM memory (EPROM, EEPROM, …)
- Limited processing power (clock speed <100 MHz)
- 8-bit/16-bit/32-bit architectures
- Single program or limited multitasking
- No operating system

Microcontroller Boards

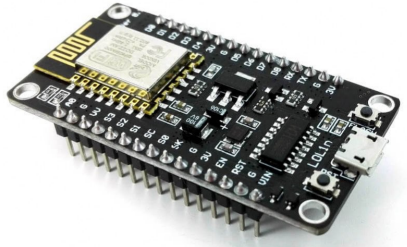
Some examples (available in our lab):



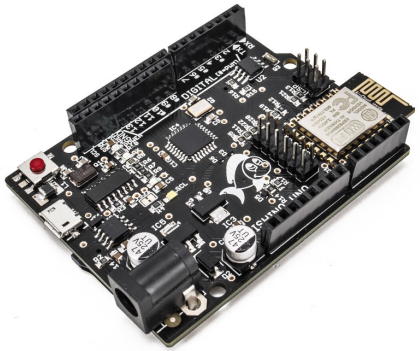
ARDUINO UNO



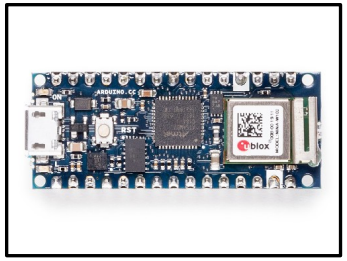
NUCLEO STM32



NODEMCU

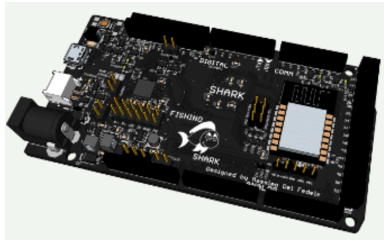


FISHINO UNO



ARDUINO NANO IOT

ESP32



FISHINO SHARK

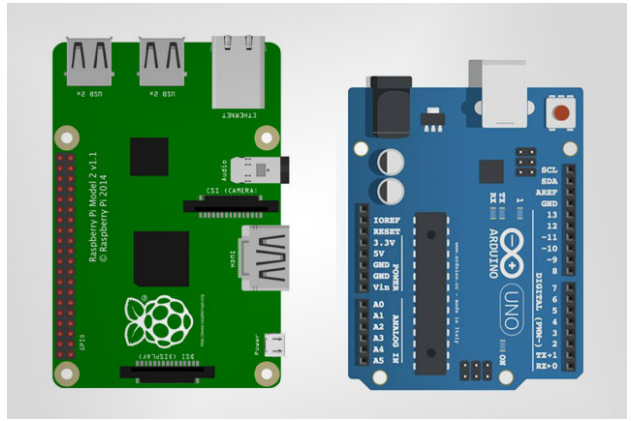
Microcontroller Boards

IoT Programming Languages for microcontroller-based boards

- **C/C++** and variants [C tutorial from W3C community LINK]
- **LUA** [Tutorial: LINK]
- **Wiring** (Arduino, actually a dialect of C/C++)
  Tutorial: Sistemi Embedded Teoria e Pratica [LINK]
- **Micropython**
  Tutorial for different IoT boards [LINK]
- **Javascript** libraries [e.g. CYLON.JS, JOHNNY FIVE]
- ....

Single board Computers

- It is a complete computer on a single circuit board, including microprocessor(s), memory, input/output (I/O) and other features required by a functional computer.
- SBC computers typically provide a fan-less, low-power computing solution and a low profile architecture.

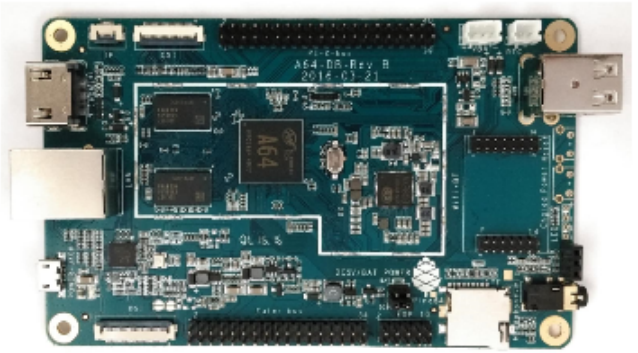


https://mybroadband.co.za

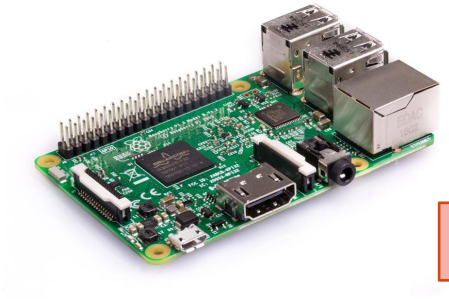
Single board Computers

Some examples (available in our lab):

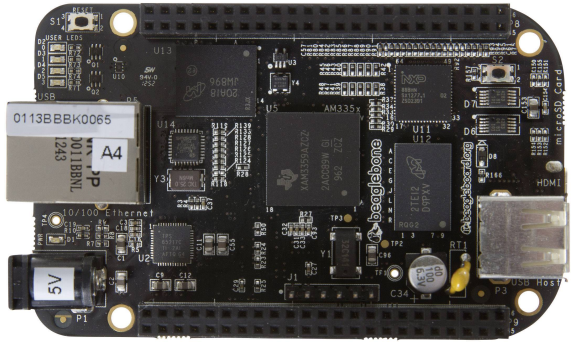
RASPBERRY PI **ZERO**



PINE64



RASPBERRY PI **3**

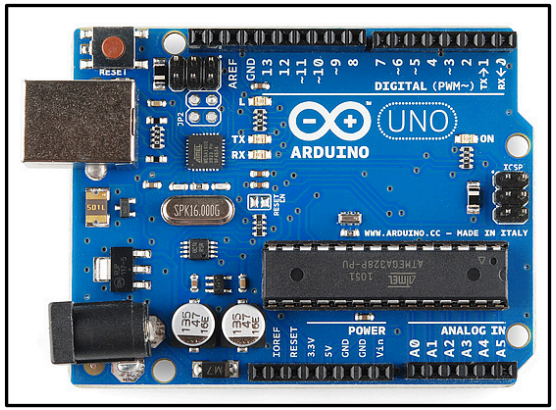


BEAGLEBONE

EXAMPLE OF MICRO-CONTROLLER BOARD: ARDUINO UNO

The Arduino Board

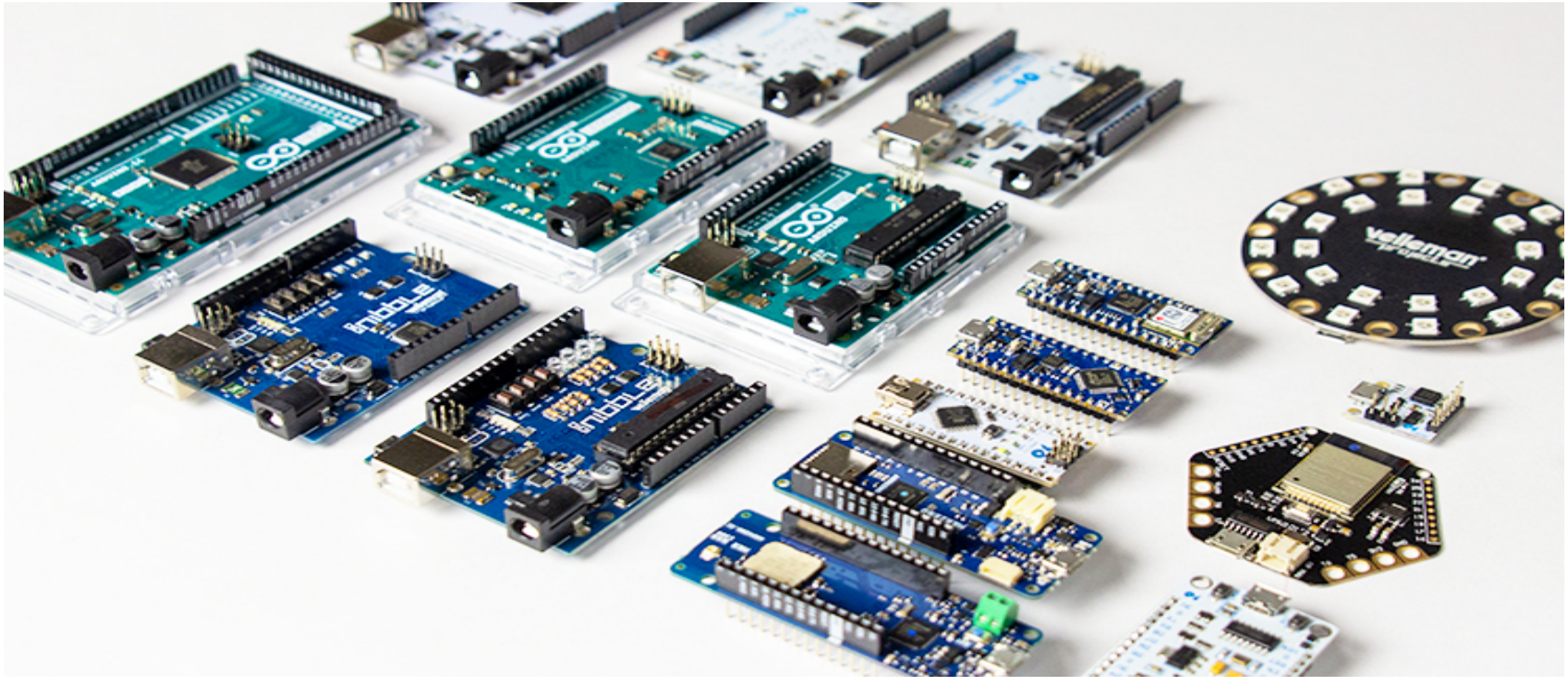
- Project started in 2003 as a program for students at the **Interaction Design Institute** (Ivrea, Italy).
- IDEA: Create simple, low cost instruments also for non-engineers, aimed to work on digital projects.
- Around 700.000 units shipped (in 2013).
- Partnership with **ARM Holdings** (2017).



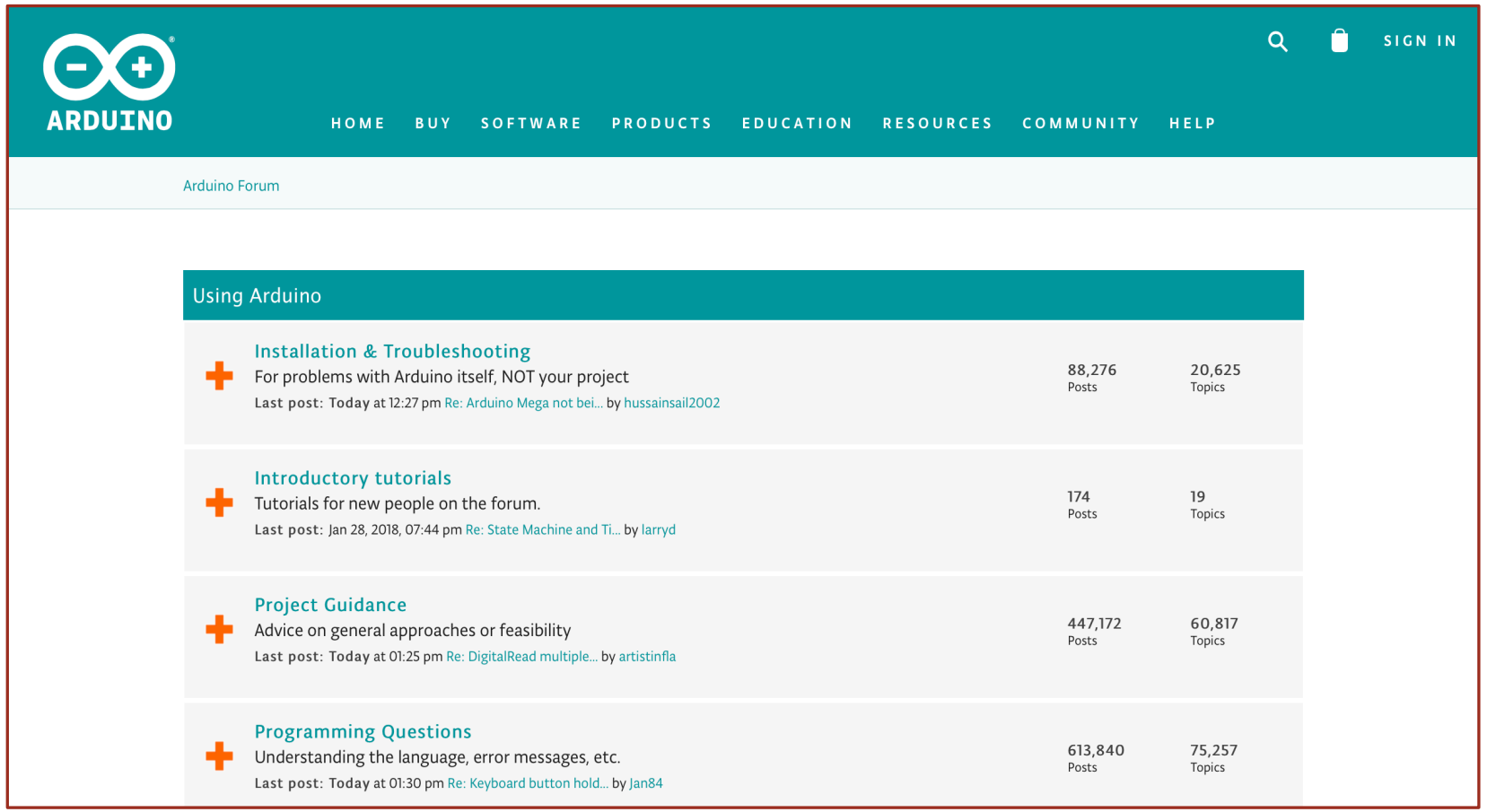
www.arduino.cc

Source: https://www.vellemanformakers.com/the-ultimate-arduino-guide/

The Arduino Board

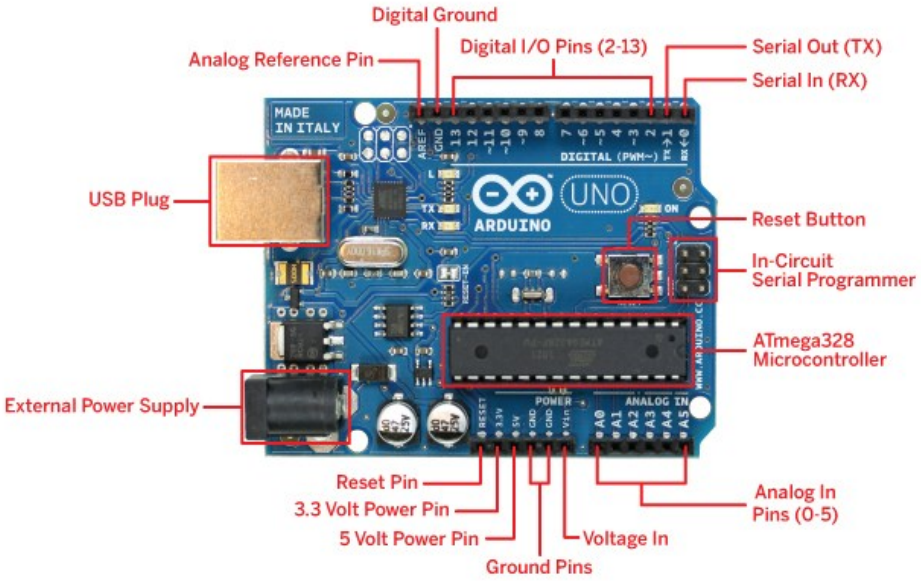


The Arduino Board



The Arduino Board

- **Single-board microcontroller**.
- Sets of digital and analog input/output (I/O) pins that can be interfaced to various expansion boards (shields) and other circuits

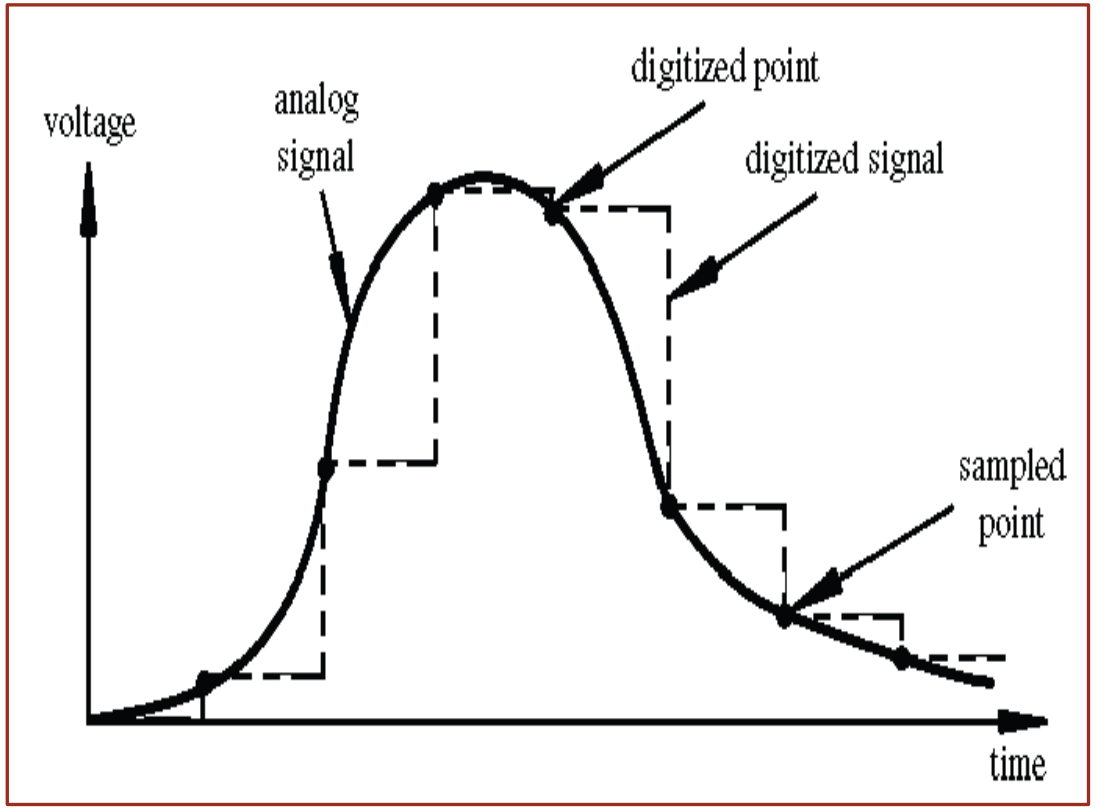


http://arduinoarts.com/2011/08/the-arduino-uno-anatomy/

The Arduino Board

❑ 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).
- ✧ 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.



The Arduino Board

❑ Pulse Width Modulation (PWM)

- ✧ Digital pins can only directly supply 3V or 5V, but they can also pulse the output on and off really fast to produce the same effect of a different voltage supply.
- ✧ The on-off pulsing is so fast that the connected output device “perceives” the result as a reduction in the voltage.

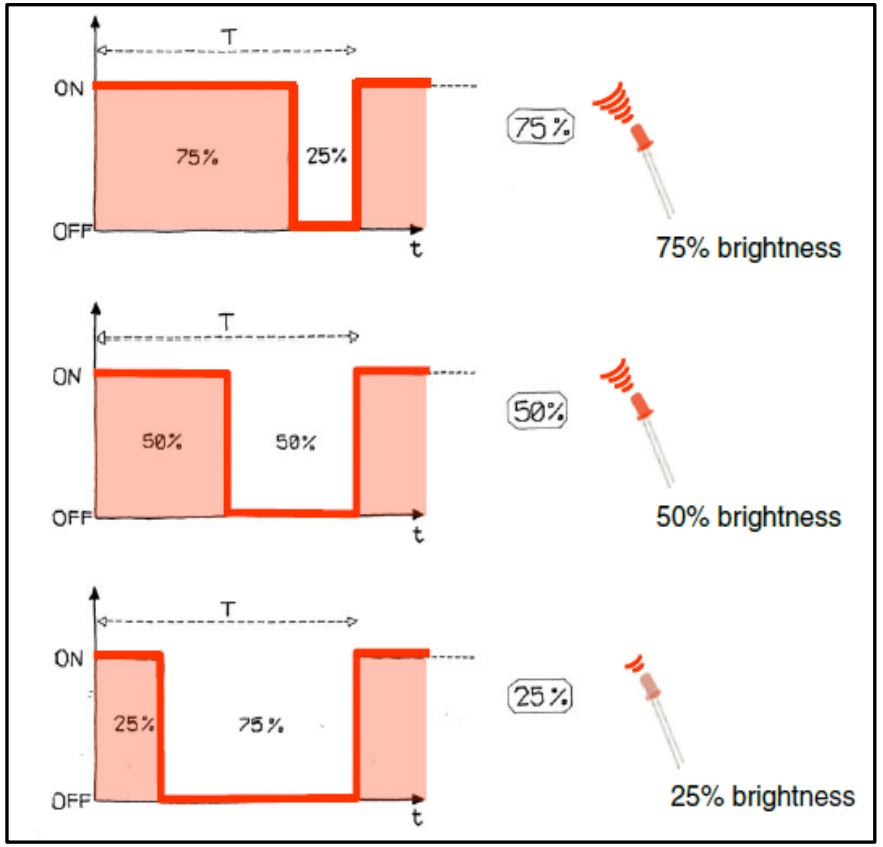


Image from *Theory and Practice of Tangible User Interfaces* at UC Berkley

The Arduino Board

output voltage = (on_time / cycle_time) * 5V

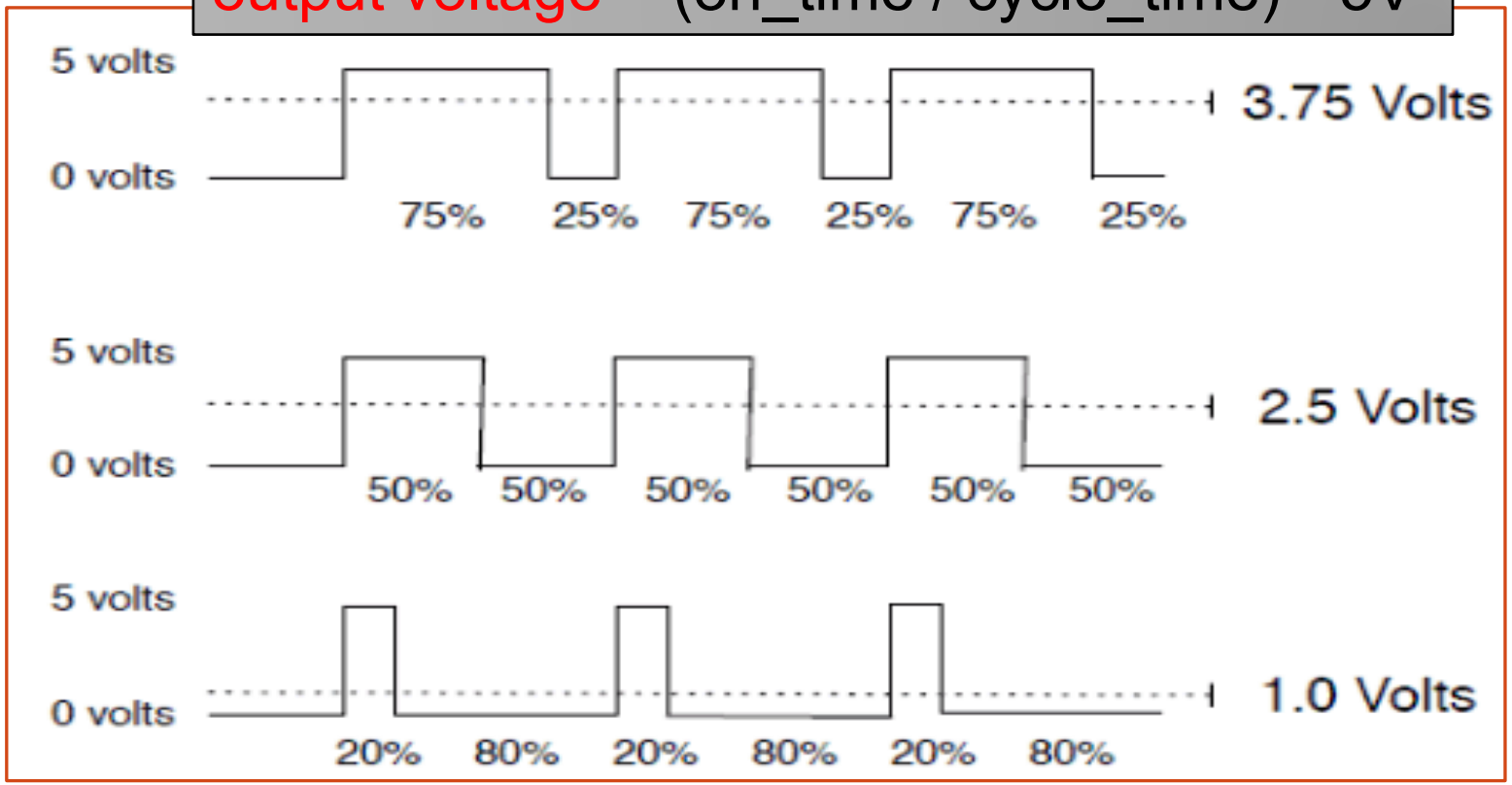
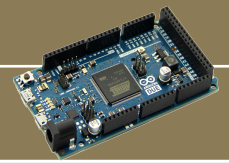
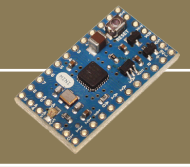
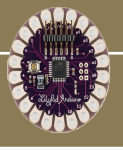


Image credit: Tod Kurt

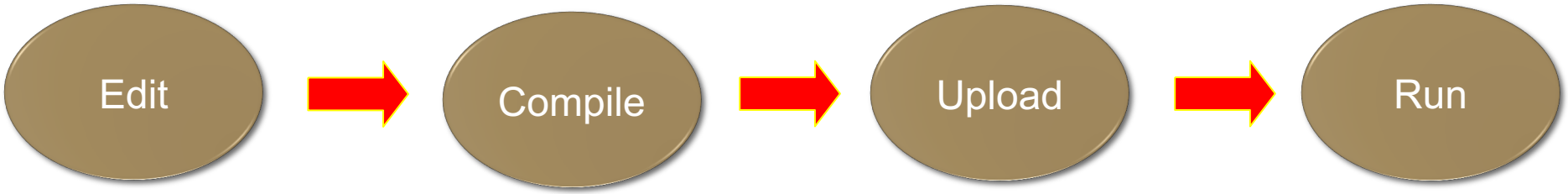
Fixed cycle length; constant number of cycles/sec

The Arduino Board

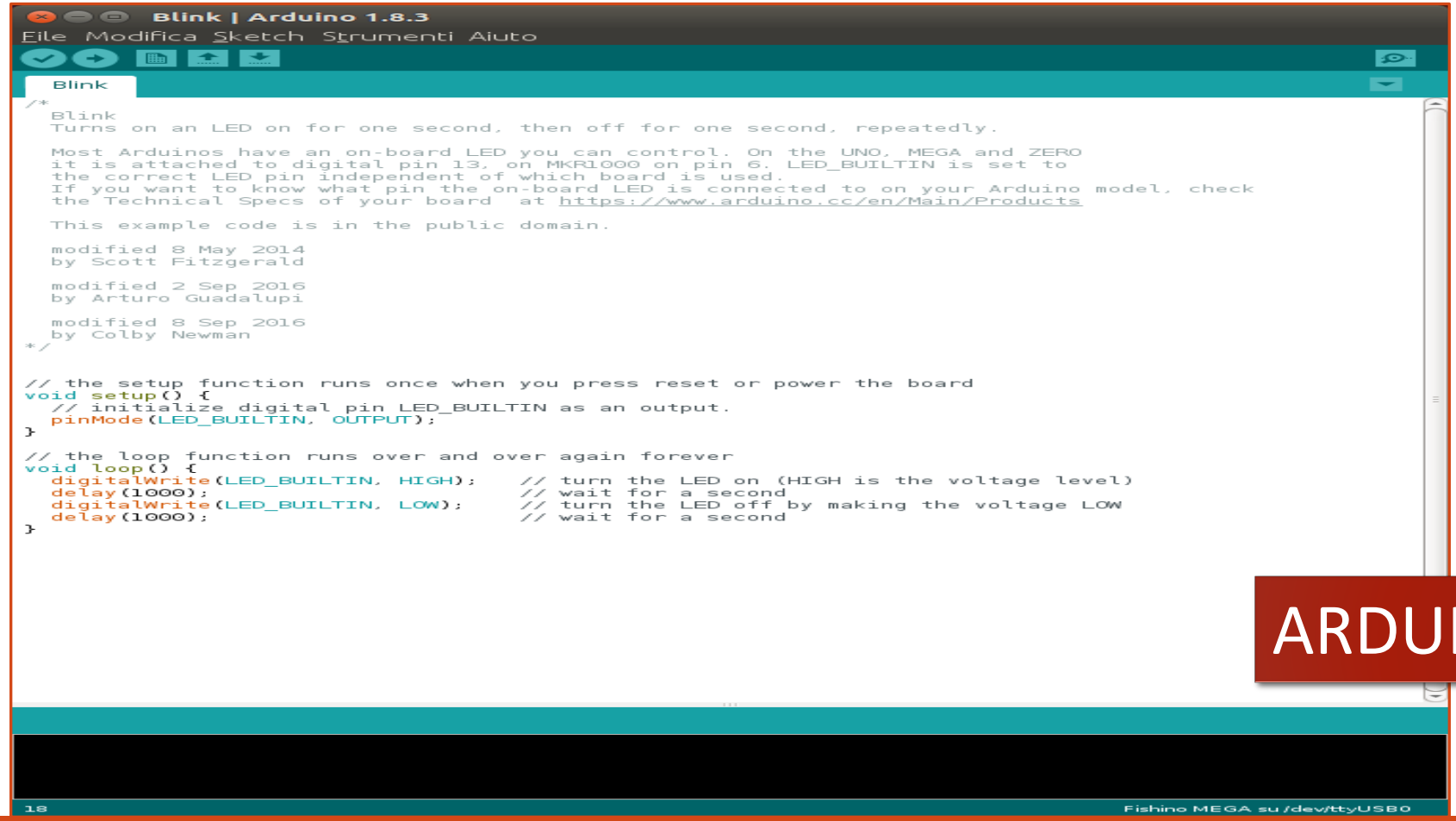
	Arduino Uno	Arduino Mega 2560	Arduino Due	Arduino Mini	Lily Pad
					
Processor	**ATmega328**	**ATmega2560**	**AT91SAM3X8E**	**ATmega328**	**ATmega168V ATmega328V**
CPU speed[MHz]	16	16	84	16	8
Analog IN/OUT	6/0	16/0	12/2	8/0	6/0
Digital IO/PWM	14/6	54/15	54/12	14/6	14/6
SRAM [KB]	2	8	96	2	1
Flash [KB]	32	256	512	32	16
UART	1	4	4	-	-
Price	~20€	~22€	~50€	~15€	~45€

The Arduino Board

- Sketches:
  - `.ino` files, `.cpp` and .h
  - `setup()` → initialization of the initial values (pins, variables, etc..)
  - `loop()` → infinite repetition of the main code
- Development cycle:



The Arduino Board



ARDUINO IDE

The Arduino Board

CLASSICAL **BLINK** EXAMPLE

```
#define LED_PIN 13

void setup() {
  // initialize digital pin LED.
  pinMode(LED_PIN, OUTPUT);
}
void loop() {
  digitalWrite(LED_PIN, HIGH);   // turn the LED on (HIGH voltage level)
  delay(1000);                   // wait for a second
  digitalWrite(LED_PIN, LOW);    // turn the LED off (voltage LOW)
  delay(1000);                   // wait for a second
}
```

The Arduino Board

- **boolean** (8 bit) - simple logical true/false.
- **byte** (8 bit) - unsigned number from 0-255.
- **char** (8 bit) - signed number from -128 to 127.
- **unsigned char** (8 bit) - same as ‘byte’; if this is what you’re after you should use ‘byte’ instead.
- **word** (16 bit) - unsigned number from 0-65535.
- **unsigned int** (16 bit) - the same as ‘word’. Use ‘word’ for clarity and brevity.
- **int** (16 bit)- signed number from -32768 to 32767.
- **unsigned long** (32 bit)- unsigned number from 0-4,294,967,295.
- **long** (32 bit) - signed number from -2,147,483,648 to 2,147,483,647.
- **float** (32 bit) - signed number from -3.4028235-E38 to 3.4028235-E38. Floating point is not native; the compiler has to jump through hoops to make it work. If you can avoid it, you should.

The Arduino Board

Digital I/O

- `digitalRead()`
- `digitalWrite()`
- `pinMode()`

Analog I/O

- `analogRead()`
- `analogReference()`
- `analogWrite()`

Communication

- Serial
- stream

Time

- `delay()`
- `delayMicroseconds()`
- `micros()`
- `millis()`

The Arduino Board

- Audio
- Cloud
- Communications
- Robotics
- Sensors
- Storage and memory
- Etc..

EXAMPLE of AVAILABLE LIBRARIES

https://playground.arduino.cc/Main/LibraryList

Firmata

- Protocol for communication between software on a host (computer, smartphone, etc..) and microcontrollers
- Firmata is based on the MIDI message format in that commands bytes are 8 bits and data bytes are 7 bits
- **Firmata libraries**:
  - Python
  - Perl
  - Javascript
  - Java
  - ruby

Firmata Message Type

type	command	MIDI channel	first byte	second byte
analog I/O message	0xE0	pin #	LSB(bits 0-6)	MSB(bits 7-13)
digital I/O message	0x90	port	LSB(bits 0-6)	MSB(bits 7-13)
report analog pin	0xC0	pin #	disable/enable(0/1)	- n/a -
report digital port	0xD0	port	disable/enable(0/1)	- n/a -
start sysex	0xF0			
set pin mode(I/O)	0xF4		pin # (0-127)	
set digital pin value	0xF5		pin # (0-127)	
sysex end	0xF7			
protocol version	0xF9		major version	minor version
system reset	0xFF			

Firmata Message Type

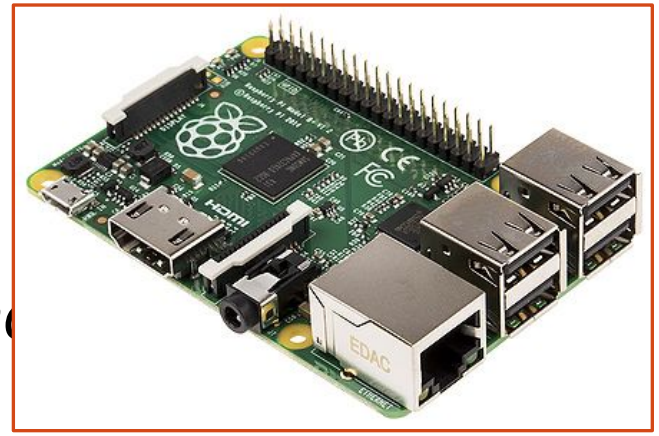
- Firmata library written in **Javascript**
- Platform Support:
  - Arduino
  - Raspberry Pi
  - Intel platforms
  - Etc…
- APIs for using the following sensors/actuators:
  - Accelerometer, GPS, Joystick, Keypad, LCD, Led, Leds, Motor, Pin, Thermometer, etc..

EXAMPLE OF SINGLE-BOARD COMPUTER: RASPBERRY PI

The Raspberry PI Board

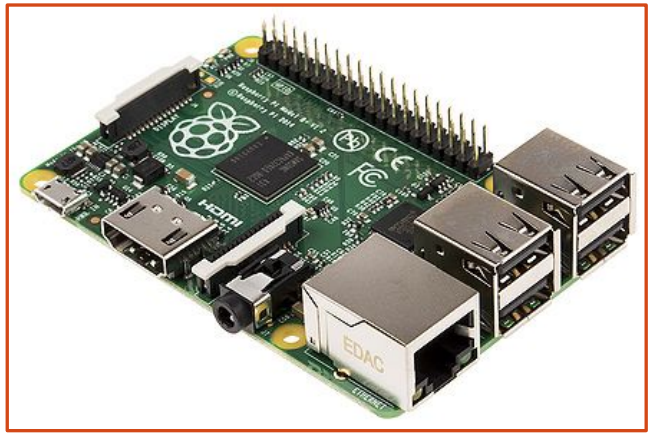
- **Developed in the United Kingdom by the Raspberry Pi Foundation** to promote the teaching of basic computer science in schools and in developing countries.
- "*The Raspberry Pi Foundation is a UK-based charity that works to put the power of digital making into the hands of people all over the world, so they are capable of understanding and shaping our increasingly digital world, able to solve the problems that matter to them, and equipped for the jobs of the future*."

https://www.raspberrypi.org/about/



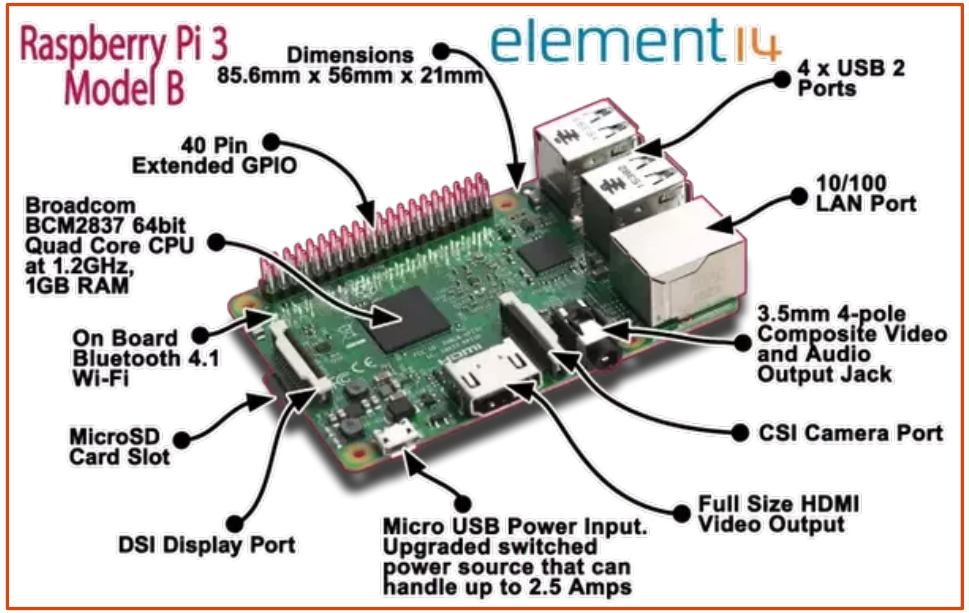
The Raspberry PI Board

- Foundation trustee Eben Upton assembled a group of teachers, academics and computer enthusiasts to devise a computer to inspire children.
- The computer is inspired by **Acorn's BBC Micro of 1981**. The Model A, Model B and Model B+ names are references to the original models of the British educational BBC Micro computer, developed by Acorn Computers.



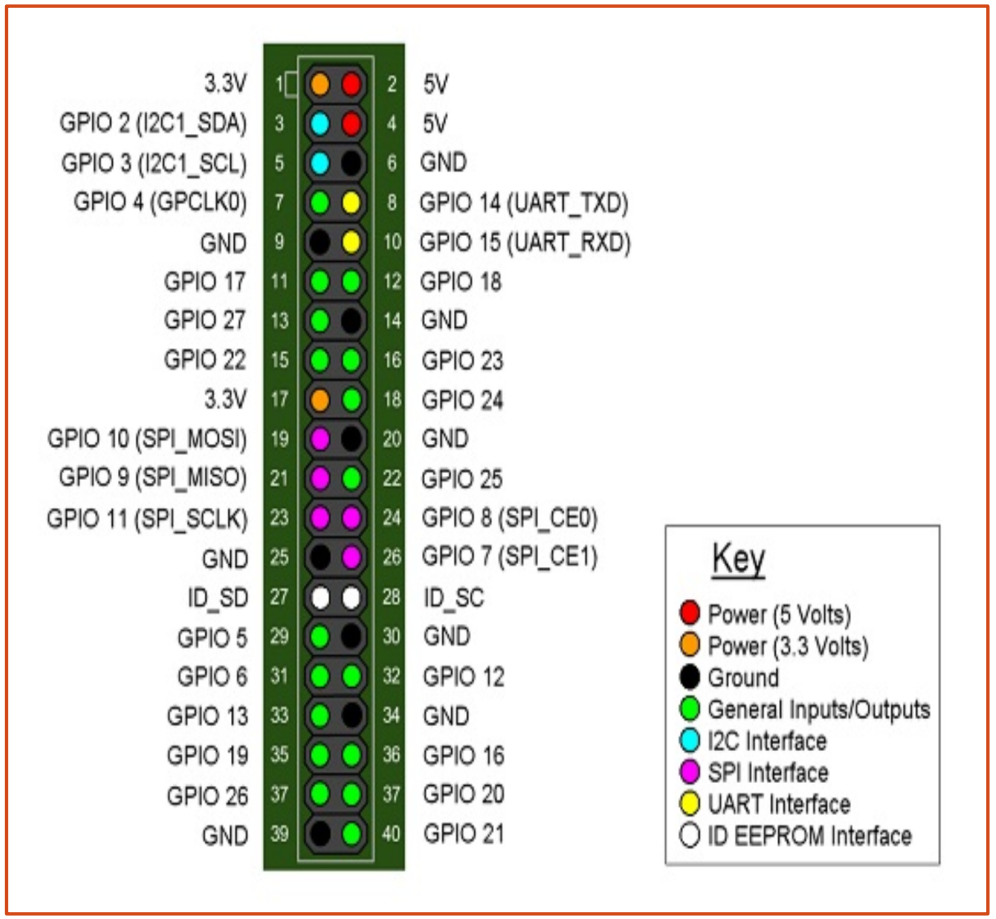
The Raspberry PI Board

- **Single-board computer** with a Broadcom system on a chip (SoC) with an integrated ARM compatible central processing unit (CPU) and on-chip graphics processing unit (GPU).



The Raspberry PI Board

- GPIO
  - 40 digital pins
  - PWM by software
  - UART
  - I2C
  - SPI



	PiA	PiB	Pi2B	Pi3B	PiZero
Release date	2014	2012	2015	2016	2015
CPU clock	700 MHz	700 MHz	900 MHz	1.2 GHz	1 GHz
Cores	1	1	4	4	1
Instruction set	ARMv6	ARMv6	ARMv7-A	ARMv8-A	ARMv6
SoC	Broadcom BCM2835	Broadcom BCM2835	Broadcom BCM2836	Broadcom BCM2837	Broadcom BCM2835
USB ports	1	2	4	4	1 Micro
Memory	256 MB	512 MB	1 GB	1 GB	512 MB
Storage	SDCard	SDCard	Micro SD card	Micro SD card	Micro SD card
Ethernet	None	100 Mbit/s	100 Mbit/s	100 Mbit/s	None
Wifi	None	None	None	802.11n	None
Bluetooth	None	None	None	4.1	None
GPIO	26 pins	26 pins	40 pins	40 pins	40 pins
Alimentation	5V/1 Amp	5V/1 Amp	5V/2 Amp	5V/2,5 Amp	5V/2 Amp
cost	20$	35$	35$	35$	5$

The Raspberry PI Board

❑ Software Distro

- ✧ Linux Distros:
  - Raspbian (Debian) and Ubuntu (at least ARMv7)
  - Archlinux, Fedora, Gentoo, etc..
  - Volumio, OpenElec
  - RISC OS
- ✧ Others:
  - Windows 10 IoT Core Edition (at least ARMv7)
  - Android

The Raspberry PI Board

- **Raspberry vs Arduino** Boards

Advantages of **Arduino**:	Advantages of **Raspberry**:
Robustness (no OS on it)	Powerfulness (multitasking)
Low Power Consumption	Networking on board
Price	All the OS functionalities

The Raspberry PI Board

- **Raspberry vs Arduino** Boards

Choose **Arduino** if:

- You want to learn basics of electronics
- You have a simple project to deploy
- Your projects does not involve so much software

Choose **Raspberry** if:

- You have a complex project, with a lot of software involved
- You need OS functionalities
- Networking is involved

DEMO