An introduction to Arduino

Rita Pucci

pucci@di.unipi.it

Content

- **Introduction on Arduino world;**
  - Idea of Arduino project;
  - “Arduino” employment;
- **Arduino: the device;**
  - Models of devices;
  - Models enable for your projects;
  - Technical characteristics;
  - Device characteristics;
  - Sensors;
- **Arduino: development environment;**
  - How to prepare the environment;
  - IDE;
  - Sketch and its structure;
  - Language and libraries;
- **Arduino: Support;**
  - Libraries;
  - Forum and Support;
  - Interesting projects;
- **Examples;**
- **Try it;**

The Idea of Arduino

Arduino is an **open-source electronics prototyping platform** based on flexible, **easy-to-use** hardware and software.

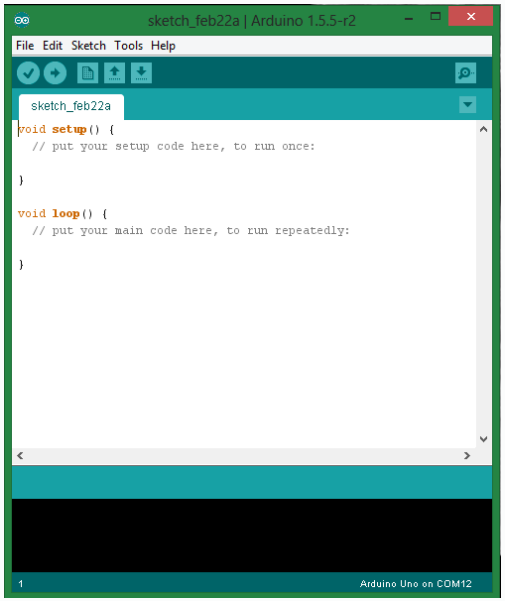
It's intended for artists, designers, hobbyists and anyone interested in **creating interactive objects or environments.**

“Arduino”

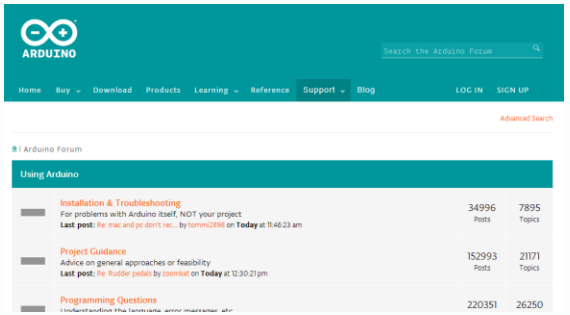
“Arduino” is employed for:

- Device

- IDE



- Forum



Arduino Hardware

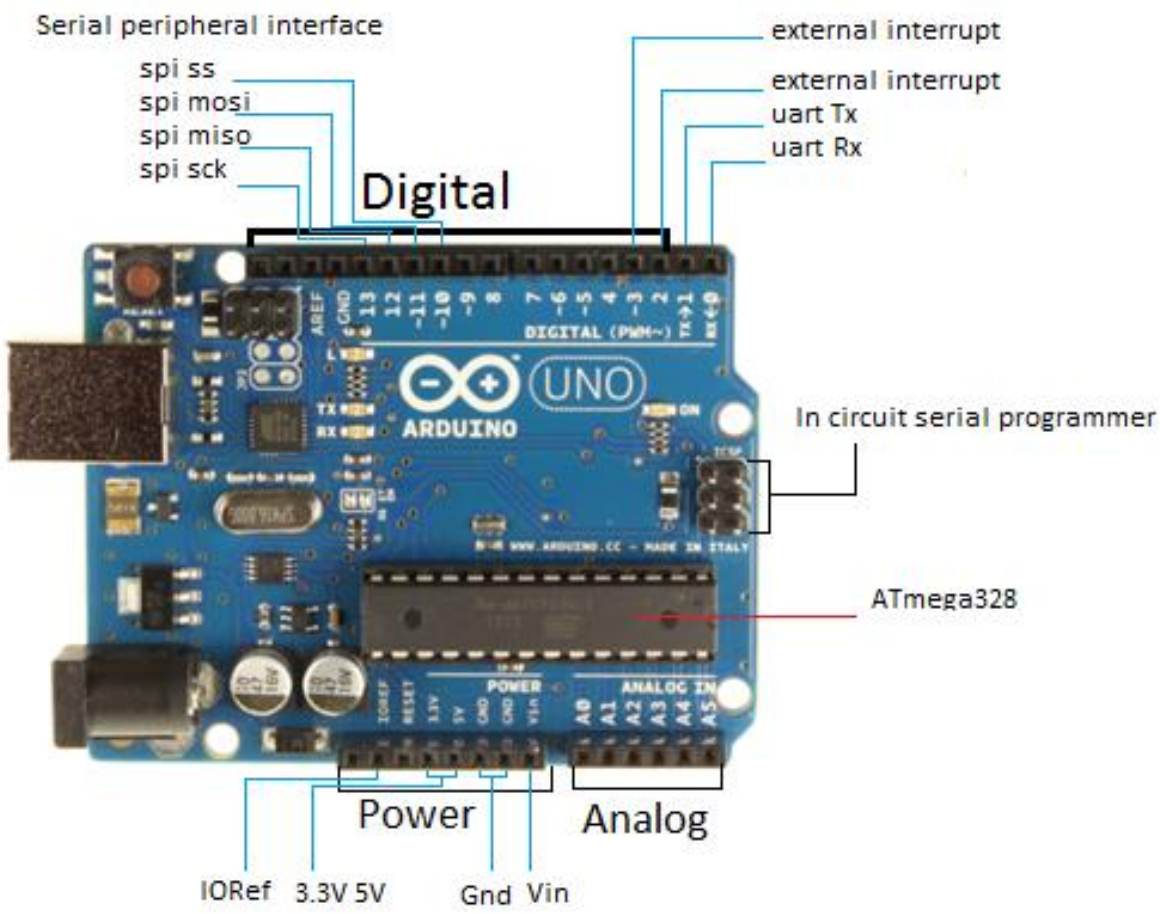
Models available for your projects

UNO

YÚN

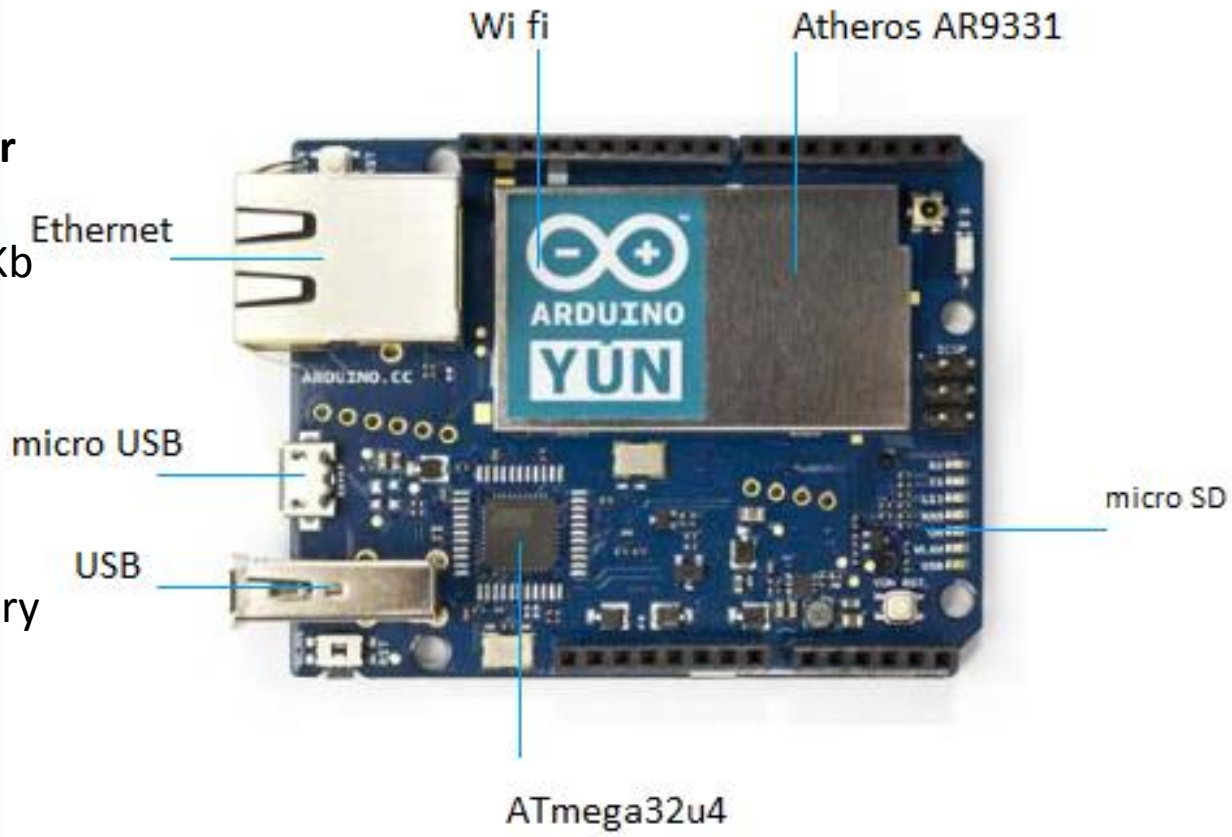
Arduino UNO

- **AVR Arduino microcontroller**
  - **Atmega328**
    - SRAM 2KB
    - EEPROM da 1KB
    - Flash memory 32 KB



Arduino YÚN

- **AVR Arduino microcontroller**
  - **Atmega32u4**
    - Flash memory 32 Kb
    - SRAM 2.5KB
    - EEPROM 1KB
- **Linux microprocessor**
  - **Atheros AR9331**
    - RAM 64 MB DDR2
    - 16MB Flash memory



Sensors, Actuators, and Shields

- Sensors
  - Accelerometer module
  - Tilt module
  - Button module
  - Linear potentiometer
  - Rotatory potentiometer
  - Joystick module
  - Hall sensor module
  - LDR sensor module
  - Temperature sensor module
  - Touch sensor module
  - Humidity sensor
  - GPS module
  - Piezo

- Actuators
  - Led (red, blue, green, yellow)
  - Power Led module
  - Servo motors
  - Stepper motors
  - Paper panel

- For high power
  - Mosfet module
  - Relay module

- Shields
  - Bluetooth
  - GSM
  - Zigbee

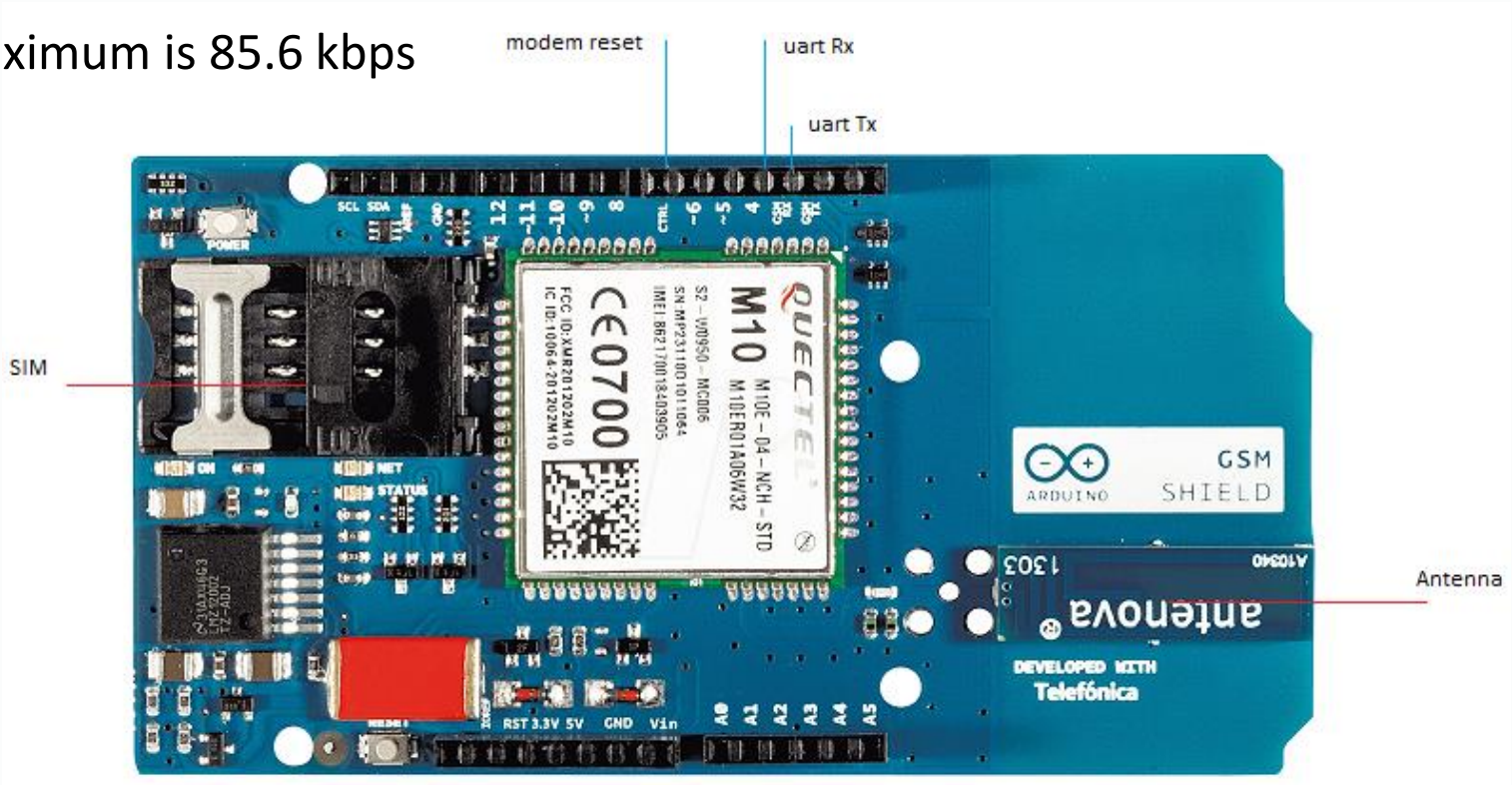
Bluetooth and Xbee module

- Bluetooth® version 2.1 module
- It supports the EDR (Enhanced Data Rate)
- Delivers up to a 3 Mbps data rate for distances up to 20 meters

- Xbee module series 1
- Standard 802.15.4
- Set as coordinator, router, end node
- 250kbps Max data rate
- 100m range

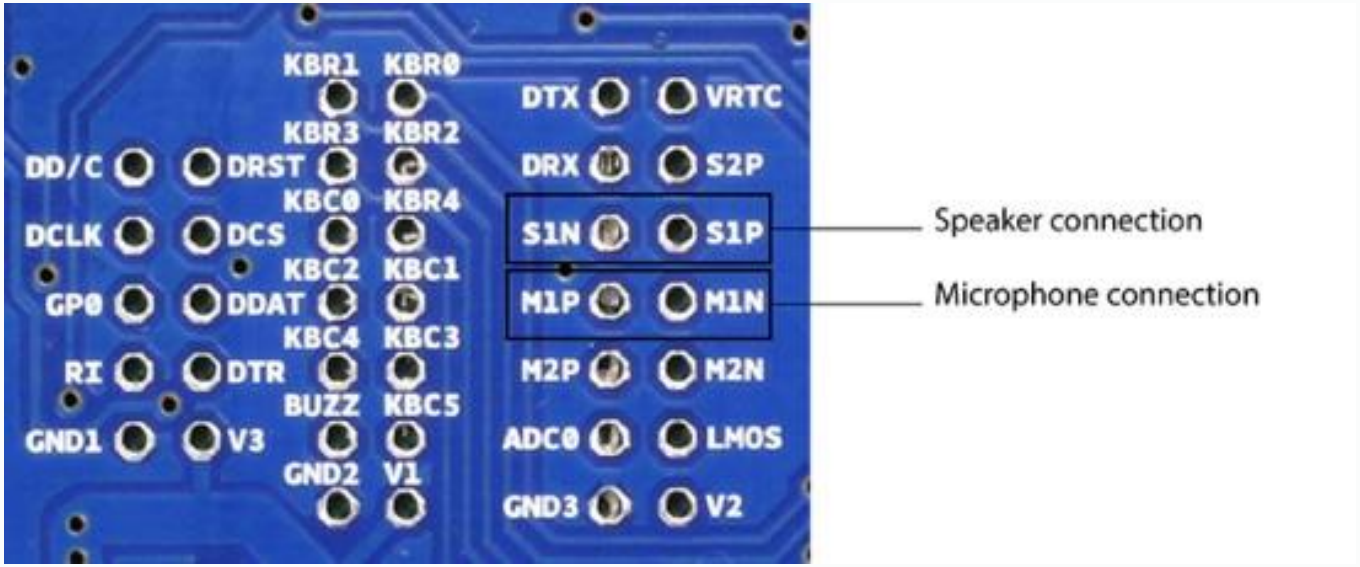
GSM shield

- Quad-band GSM/GPRS modem
- Supports TCP/UDP and HTTP
- Speed maximum is 85.6 kbps



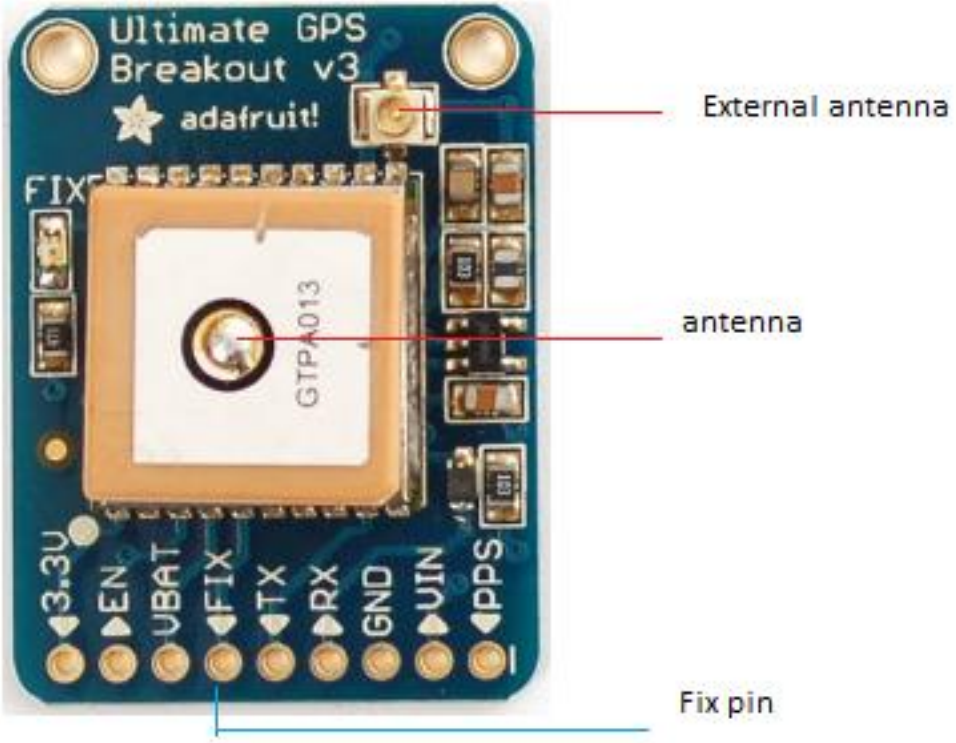
GSM shield

Through the modem, it is possible to make voice calls.



GPS module

- Power usage is incredibly low
- Ultra-low dropout 3.3V regulator so you can power it with 3.3-5VDC in, 5V level safe inputs
- Position accuracy of 1.8 meters
- Velocity accuracy of .1 meters per second



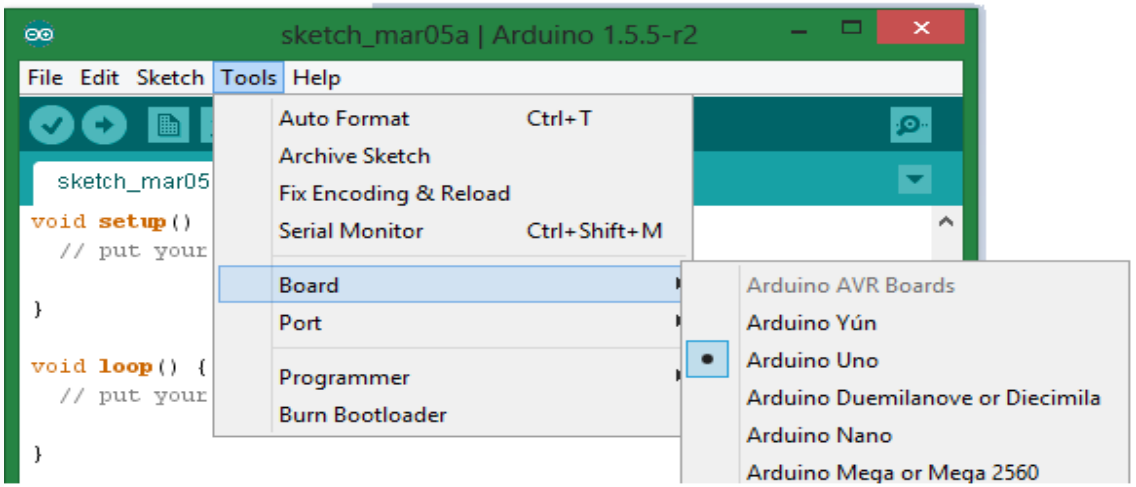
Arduino Software

How to prepare the environment

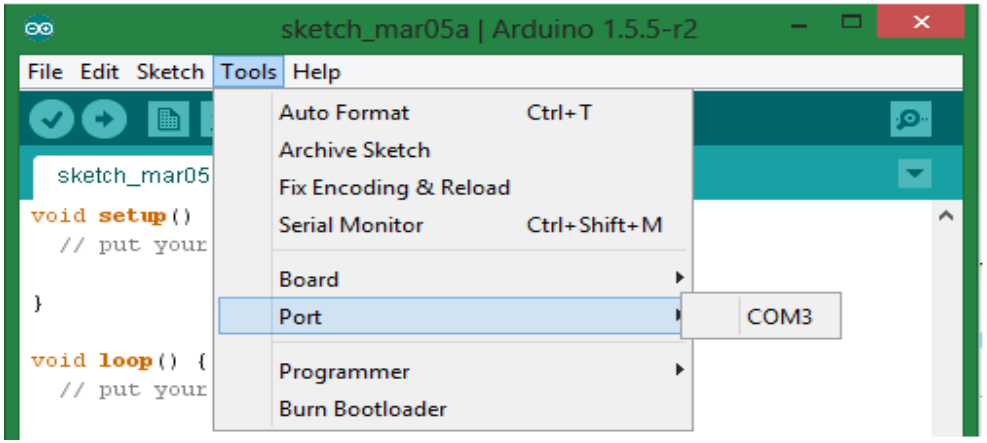
The **open-source** Arduino **environment** makes it **easy to write code** and **upload it to the I/O board**. It runs on **Windows, Mac OS X, and Linux**. The environment is written in Java and based on Processing, avr-gcc, and other open source software.

Arduino IDE can be downloaded at www.arduino.cc

Selection Location and Type



Select your arduino



Select the location of device

Terminology

- "*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 0-1023.
  - e.g. LED brightness, motor speed, etc.

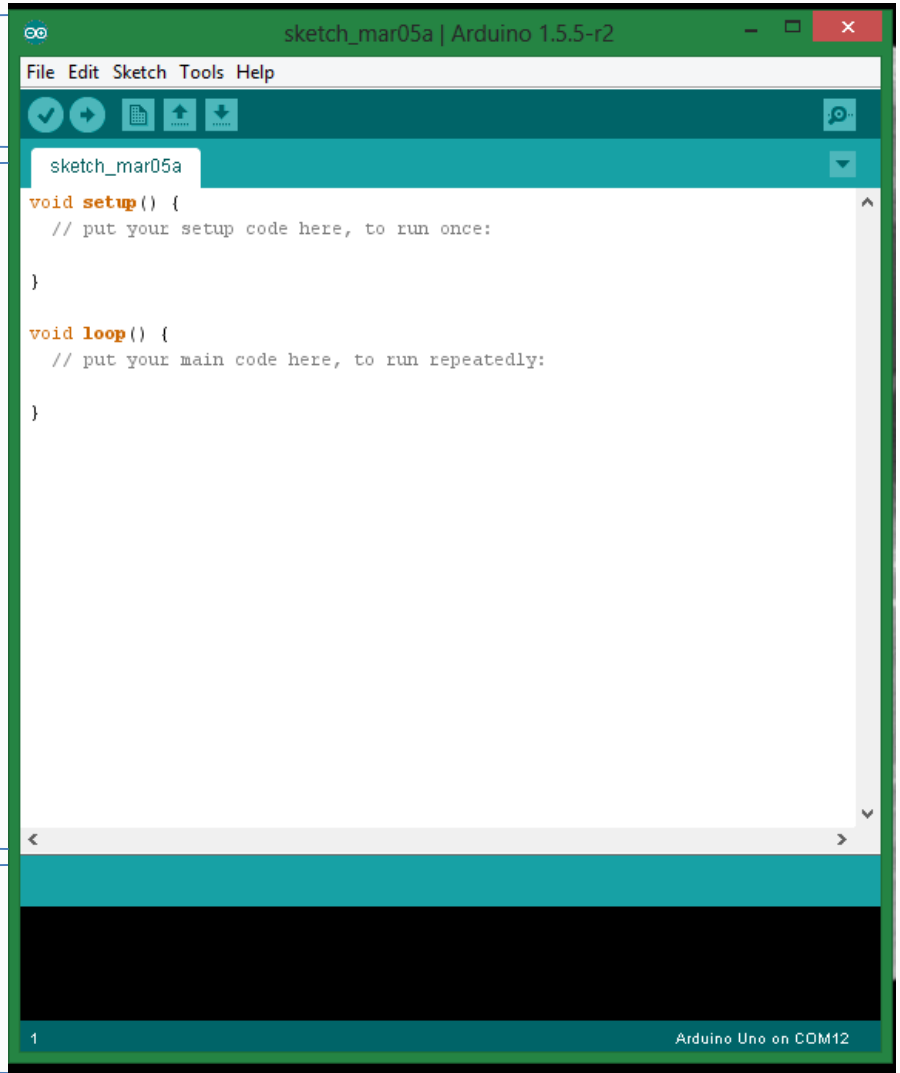
IDE

- Verify
- Upload
- New
- Open
- Save
- Serial monitor

Toolbar buttons

Sketch editor

Console display



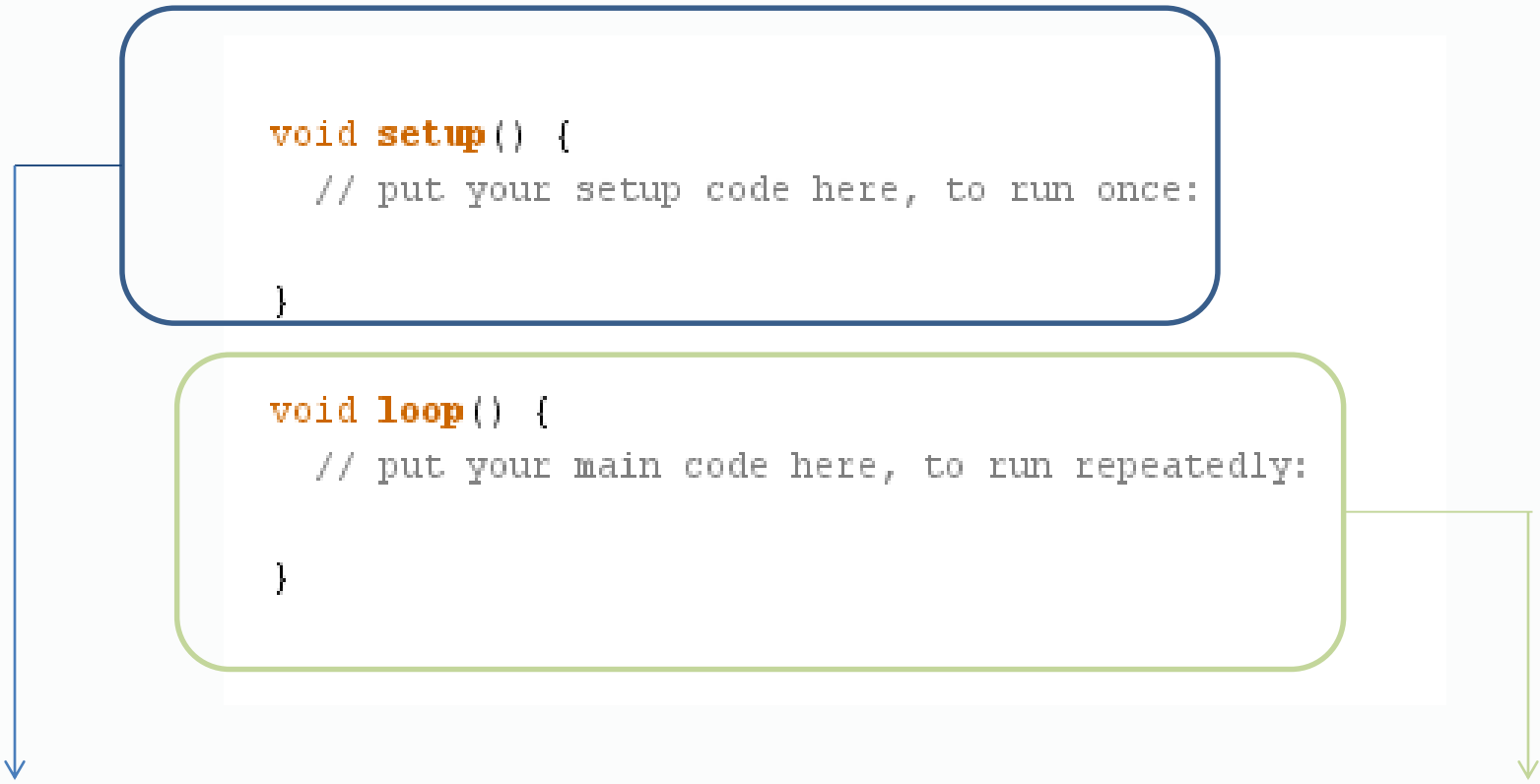
Language

The Arduino environment is based on Atmel Atmega microcontrollers. The AVR language is a "C" environment for programming Atmel chips.

The programs can be divided in three main parts:

- Sketch Structure
- Variables
- Functions

Sketch and its structure



Called when a sketch starts.
The setup function will only run once.

Does precisely what its name suggests, and loops consecutively.

Other structure functions

- Control Structures: if then else, for, switch, while, continue, return, goto …;
- Further Syntax: ;, {}, //, /**/, #include, #define;
- Arithmetic Operators: +, -, =, /, *, %;
- Comparison Operators: ==, !=, <, >, <=, >=;
- Boolean Operators: &&, ||, !;
- Pointer Access Operators: *, &;
- Bitwise Operators: &, |, ^, >>, <<, ~;
- Compound Operators: ++, --, ==, +=, -=, *=, /=, &=, |=;

Variables

- Constants: level of energy (HIGH; LOW); mode of pin(INPUT; OUTPUT; INPUT_PULLUP); led13(LED_BUILTIN);…;
- Types: word; String;…;
- Conversions: word();…;
- Variable scope and qualifiers: Volatile;…;
- Usefulness: sizeof();

Functions

Functions are distinguished according to the pin:

- Digitals: pinMode(); digitalRead(); digitalWrite();
- Analogs: analogReference(); analogRead(); analogWrite();
- Advanced I/O: tone(); noTone(); shiftOut(); shiftIn(); pulseIn();
- Time: millis(); micros(); delay(); delayMicroseconds();
- Math: min(); max(); abs(); ...;
- Trigonometry: sin(); cos(); tan();
- Random Numbers: randomSeed(); random();
- Bits and Bytes: lowByte(); highByte(); bitRead(); bitWrite(); bitSet(); bitClear(); bit();
- External Interrupts: attachInterrupt() detachInterrupt()
- Interrupts: interrupts(); noInterrupts();
- Communication: Serial; Stream;

Arduino Support

Libraries

All Libraries for all Arduino shields and components are on: Library

Forum & Support

Support for arduino programmer:
http://forum.arduino.cc

Tutorial of Arduino Owner:

Arduino Tutorial

Starter projects with Arduino:

Starter Projects

Tutorial for AdaFruit component:

- GSM and GPS
- Adafruit products

Interesting projects

- Bare Conductive
- Smart citizen kit
- Little Robot Friends
- Little Bits
- Primo
- Earth Make
- Annikken Andee

Let’s try it

- Blink Led
- Potentiometer rotary + blink led
- Humid + Term with yun
- Volatile Button
- GPS paring