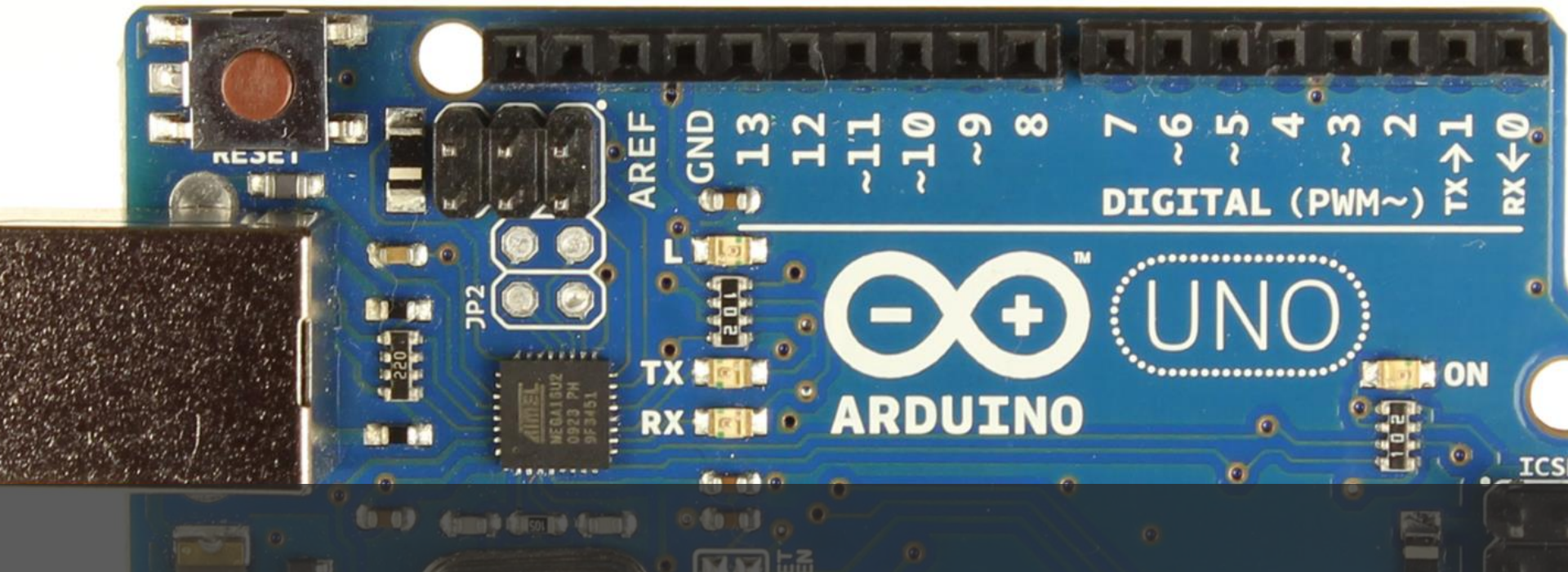




Mini curso Arduino Parte I

Prof. Rogério Neves



Conteúdo – Parte I

- O que é Arduino?
- Hardware e Software
- Dispositivos e conexões
- Interface e linguagem
- Entrada e saída de dados
- Prática de desenvolvimento 101

Parte II

- Projetos com Arduino

O que é Arduino?

- Proposta
- Vantagens
- Origens
- Modelos

Proposta

- Interface de desenvolvimento Open-Source com licença pública
- Disponibiliza uma linguagem simples e familiar (similar a C/C++)
- Permitir a hobbistas, artistas, estudantes e a comunidade “DIY”
 - ter acesso a prototipação rápida sem a necessidade de conhecimento aprofundado de eletrônica e microprocessadores
 - Compartilhar projetos na comunidade aberta (Forum)
 - Conectar uma variedade de instrumentos e sensores

<http://spectrum.ieee.org/geek-life/hands-on/the-making-of-arduino/o>

Vantagens

- Projeto aberto (hardware e software)
- Permite prototipação rápida
- Sem conhecimento de operação de micro controladores
- Conhecimento básico de eletrônica *
- Conhecimento básico de programação
- Gravação simples
 - Bootloader
 - Arduino IDE
- Bibliotecas prontas

* Conhecimento de eletrônica permite desenvolver projetos mais sofisticados

Origens

- Embarcados
- Basic Stamp, PIC, ATMEL AVR
- Projetos desenvolvidos no MIT (EUA) e na Interaction Design Institute de Ivrea (Itália) *
- Primeiro modelo comercial em 2005 (de acordo com Massimo Banzi)

* <https://arduinohistory.github.io/>

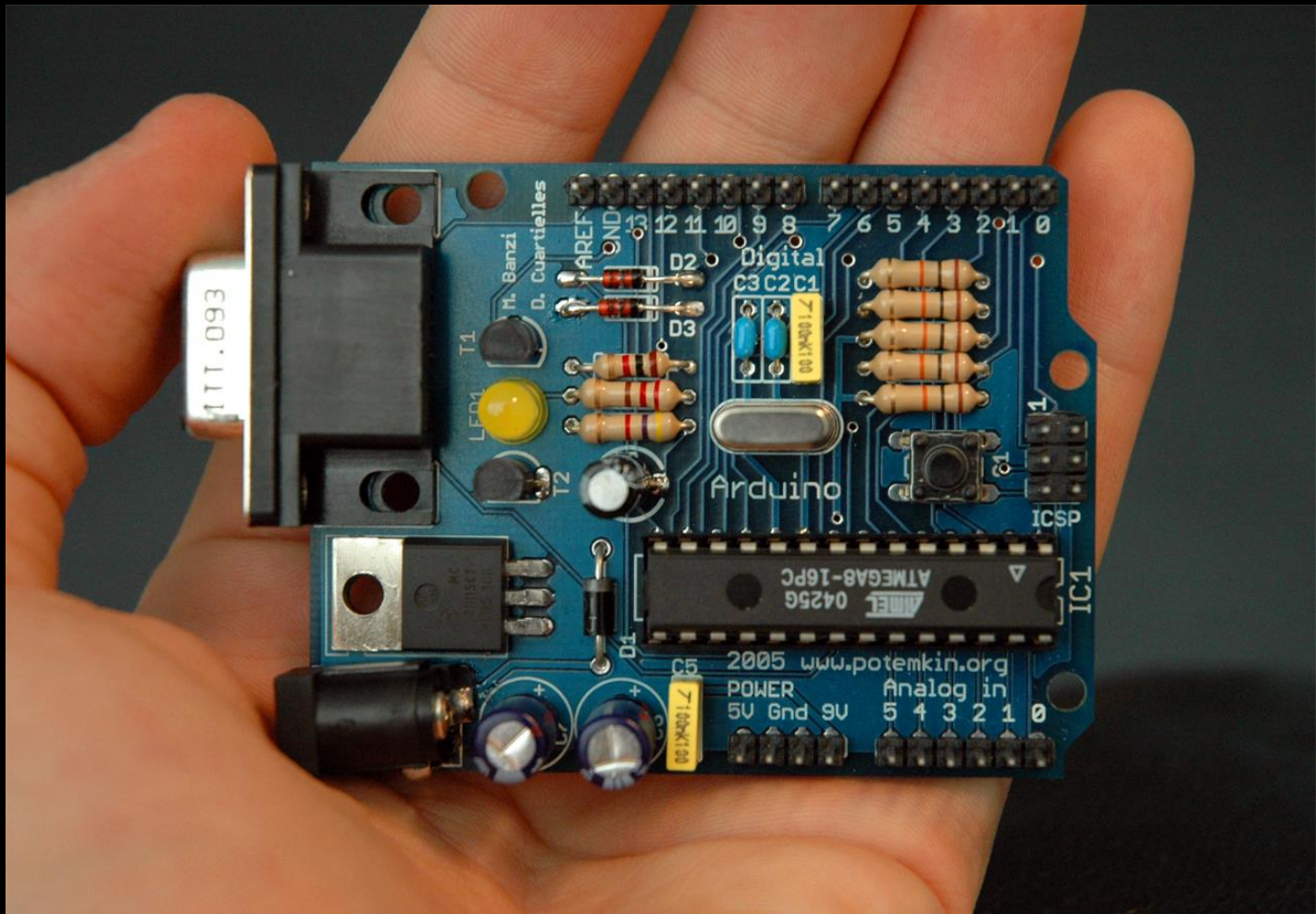
Sistemas embarcados

- Sistemas que contém microprocessadores dedicados
- Embutidos no aparelho/equipamento a que se destina
- Soluções para prototipação
 - PIC
 - Basic Stamp
 - Arduino
 - ESP8266 e ESP-12
 - Raspberry PI
 - Intel Galileo
 - Vários outros

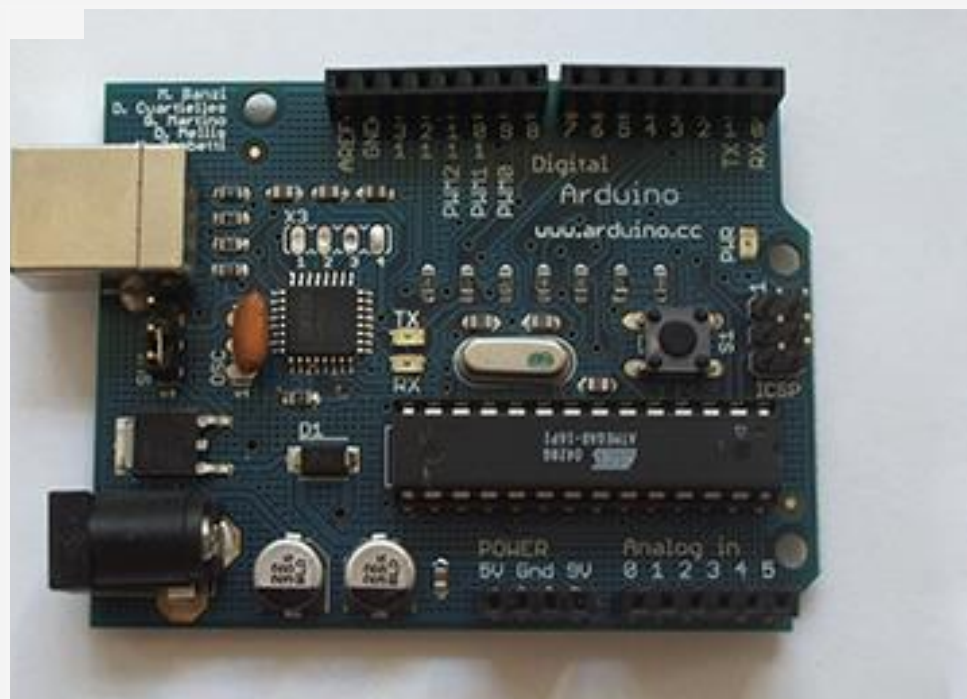
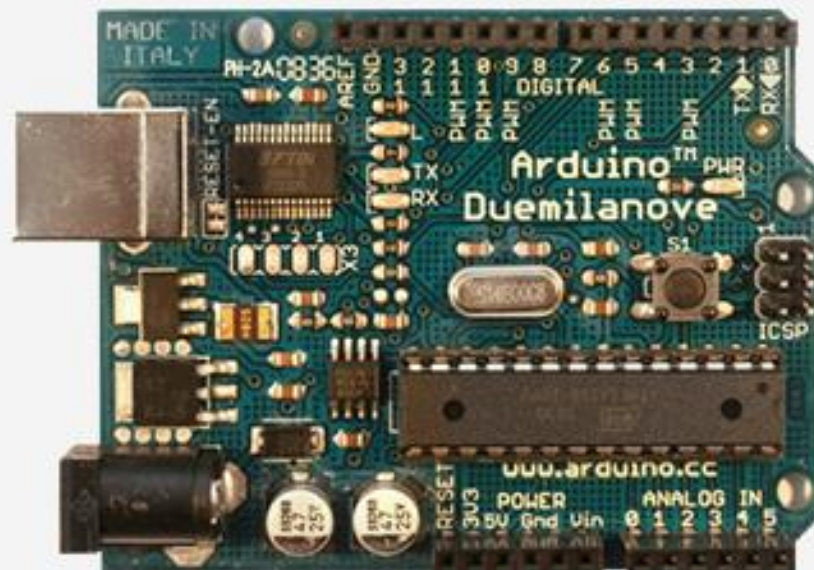
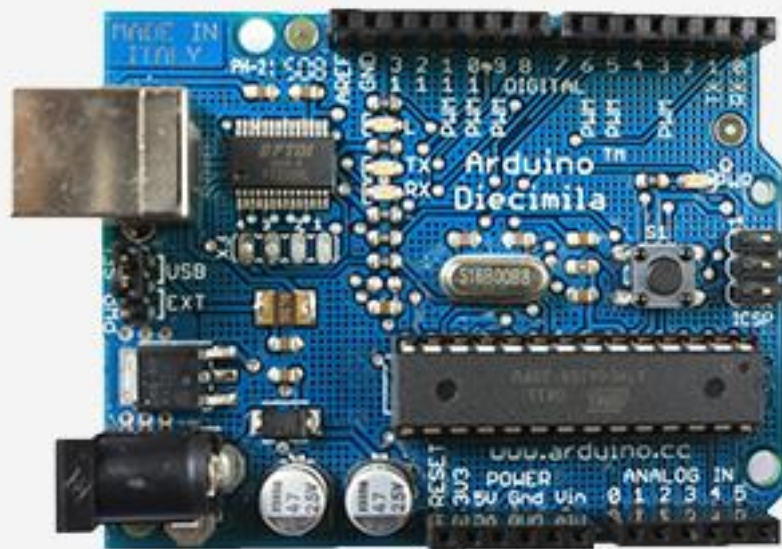
Hardware

- Inicialmente utilizava micro controladores ATMEL (ATMEGA8)
- Utiliza hoje uma variedade de micro controladores
- Os modelos mais populares utilizam o chip ATMEGA328 (Arduino UNO, Duemilanove)
- Os novos modelos usam ARM CORTEX de 32 bits
- Varias variações, clones e projetos feitos pela comunidade

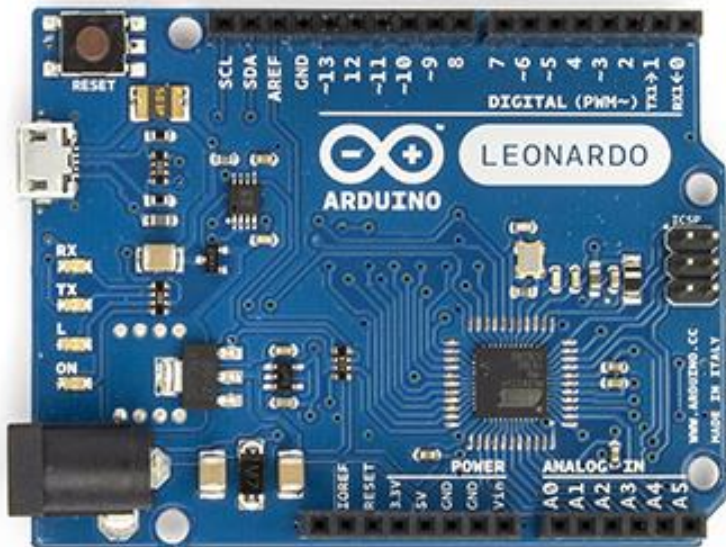
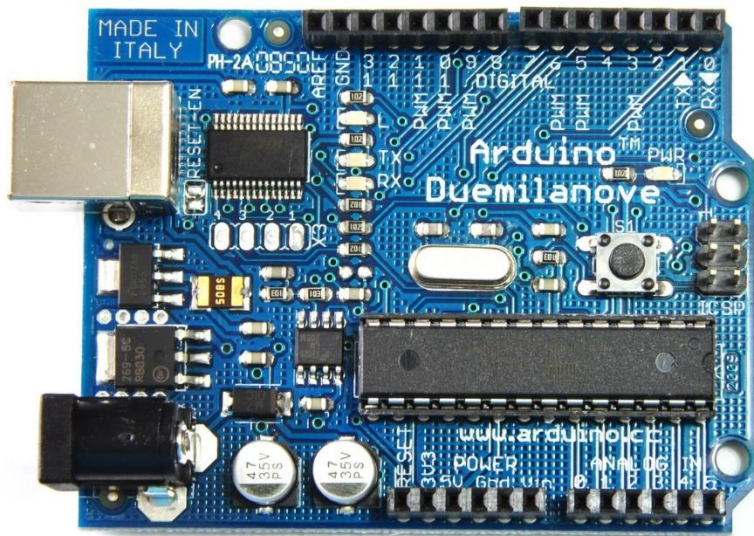
Modelo original



Arduino de Massimo Banzi (2005)



Diecimila, Duemilanove, Extreme 2, NG Rev. C, NG, Extreme, NG, USB 2.0, USB



MAIS VENDIDOS

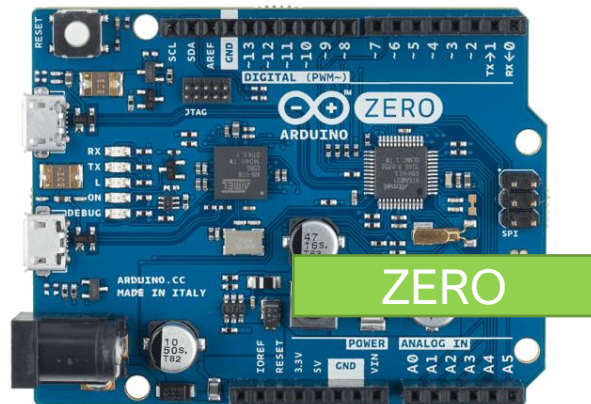
<https://www.arduino.cc/en/Main/Products>



ARDUINO TRE



MEGA 2560



ZERO



ARDUINO 101



ARDUINO PRO

Compact



ARDUINO MICRO

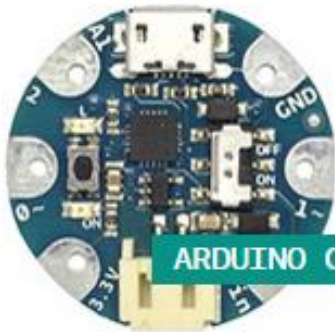


ARDUINO PRO MINI



ARDUINO NANO

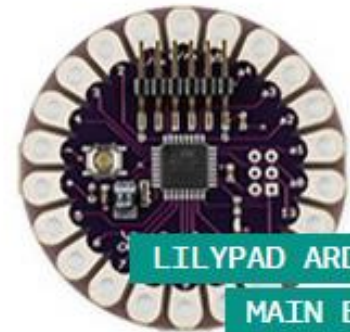
Wearable



ARDUINO GEMMA



LILYPAD ARDUINO
SIMPLE



LILYPAD ARDUINO
MAIN BOARD



LILYPAD ARDUINO USB



LILYPAD ARDUINO
SIMPLE SNAP

Descontinuados



ARDUINO MEGA ADK



ARDUINO ETHERNET



ARDUINO ROBOT



ARDUINO LEONARDO

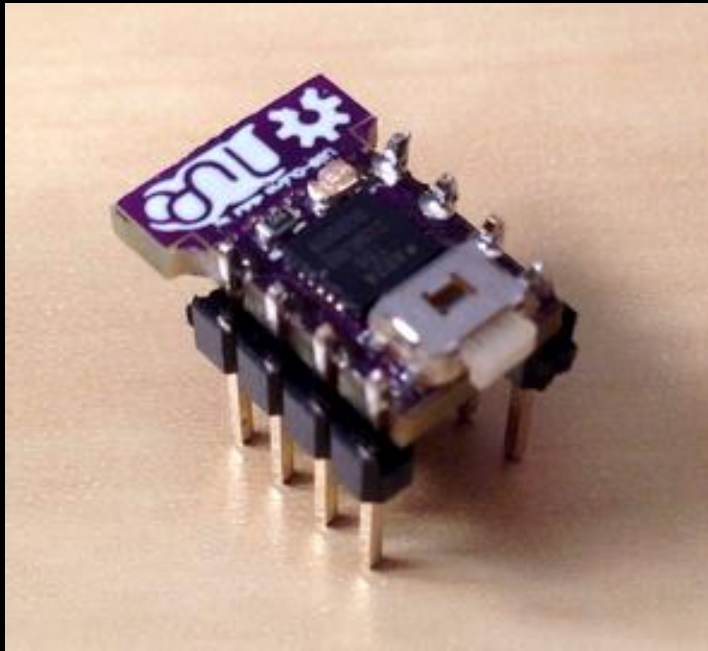


ARDUINO ESPLORA



ARDUINO MINI

Modelos da comunidade

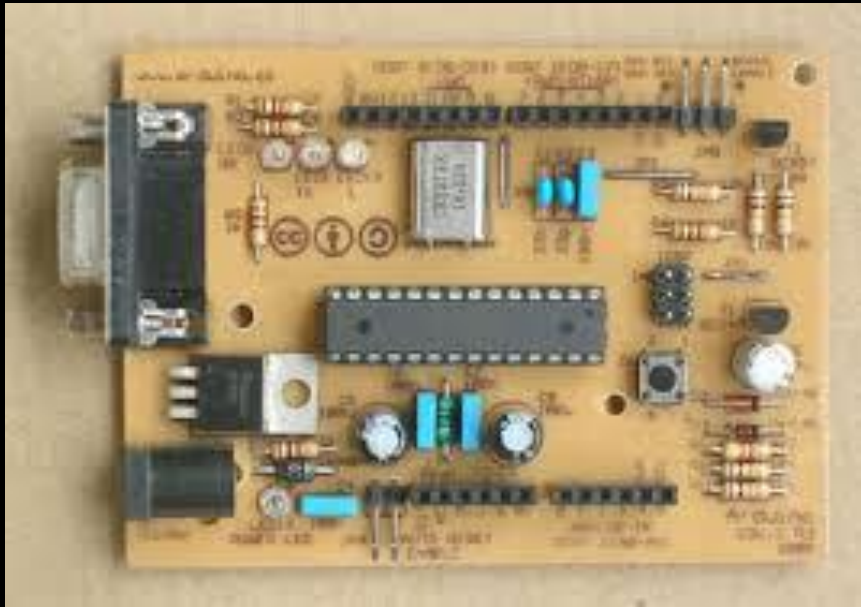


DILduino

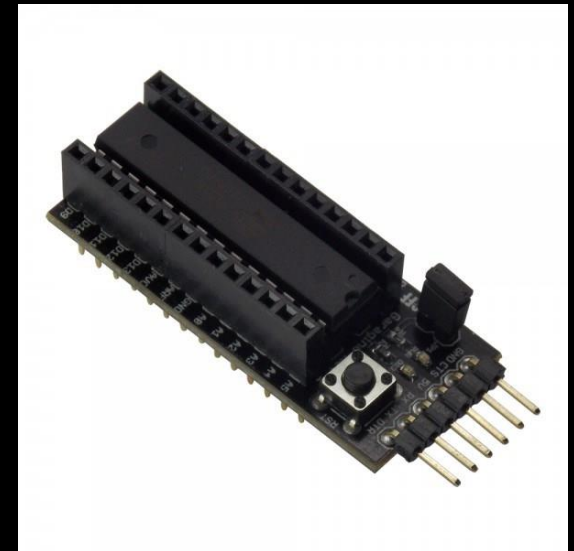


Latuino

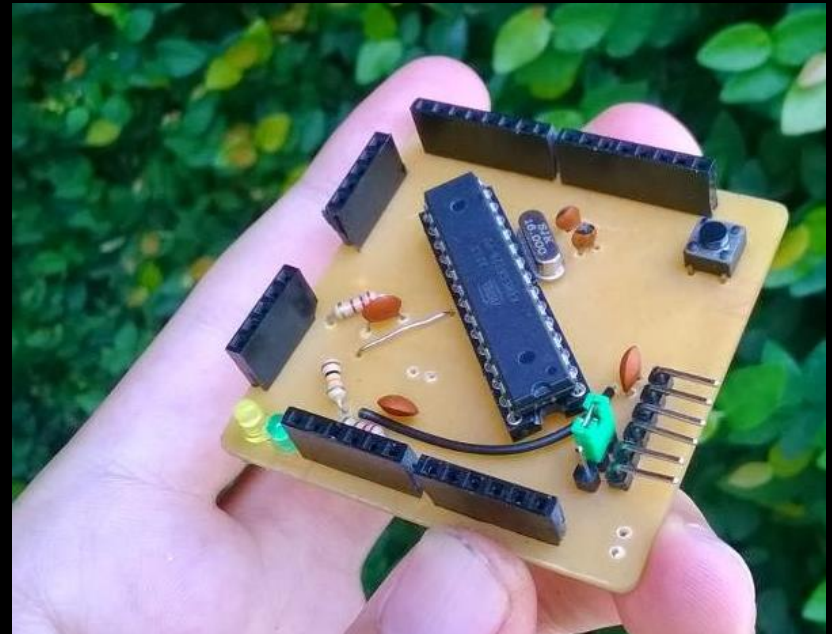
Versões nacionais



Severino

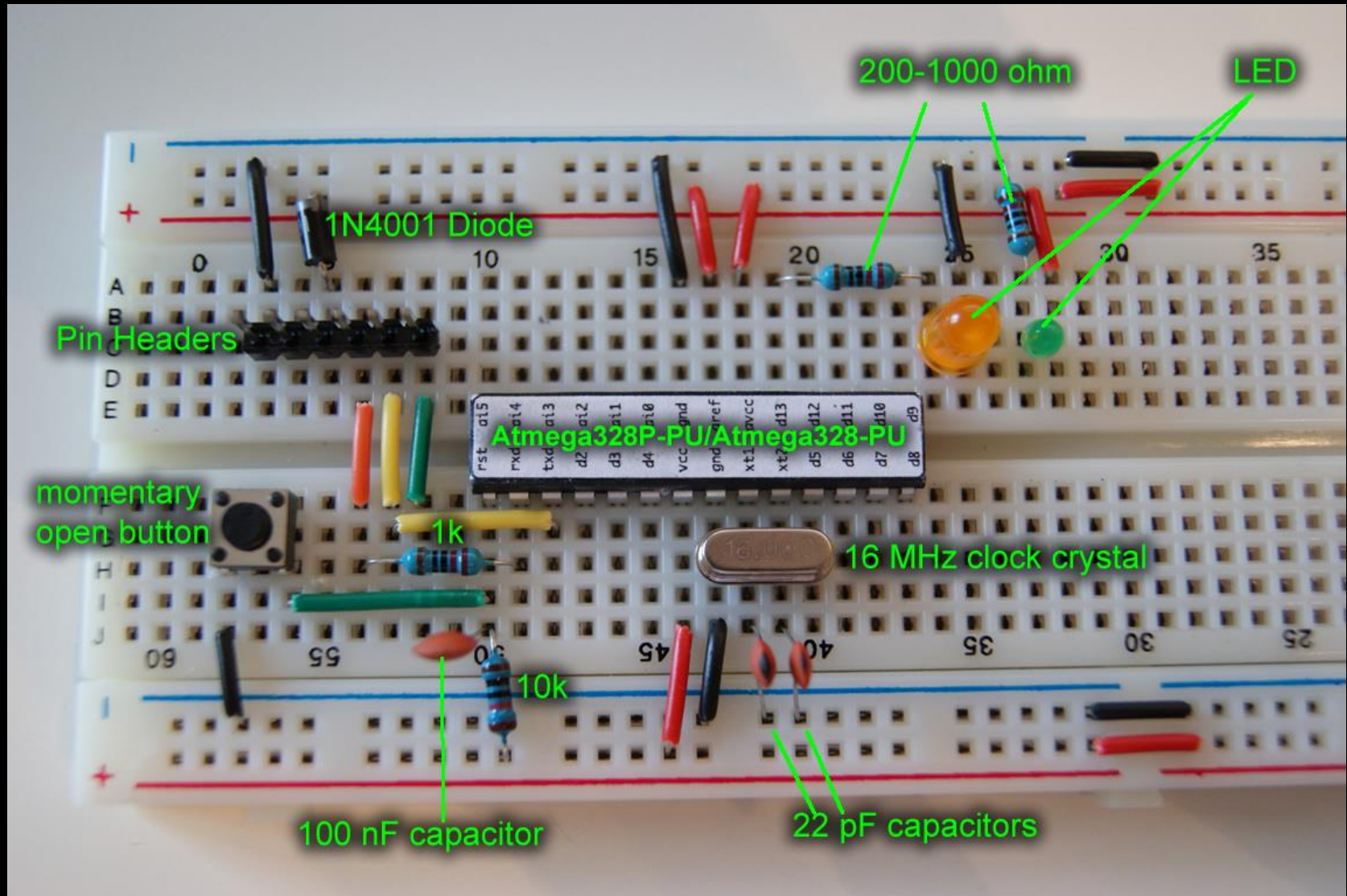


Garagino

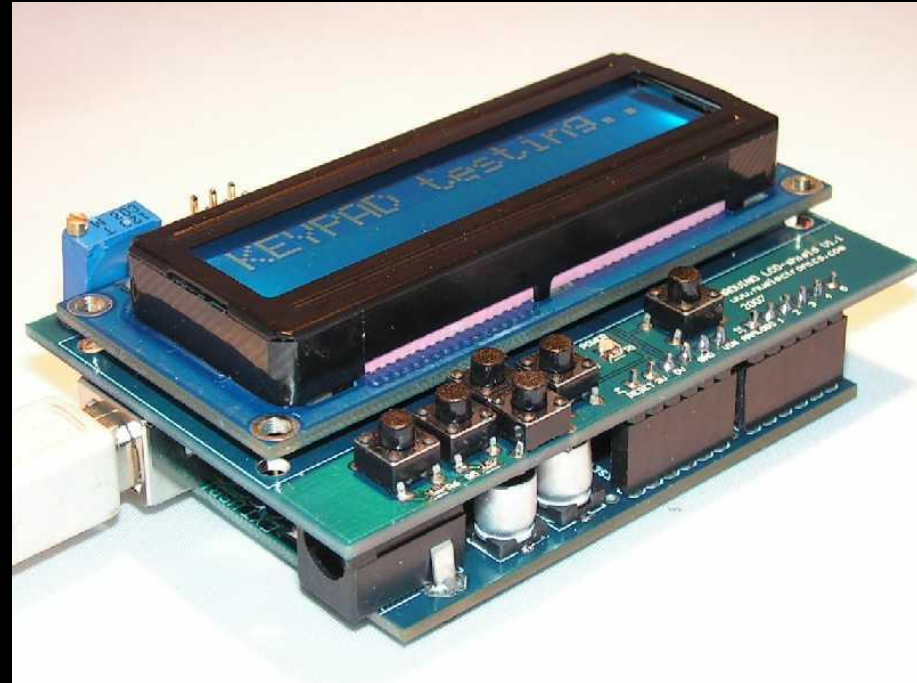
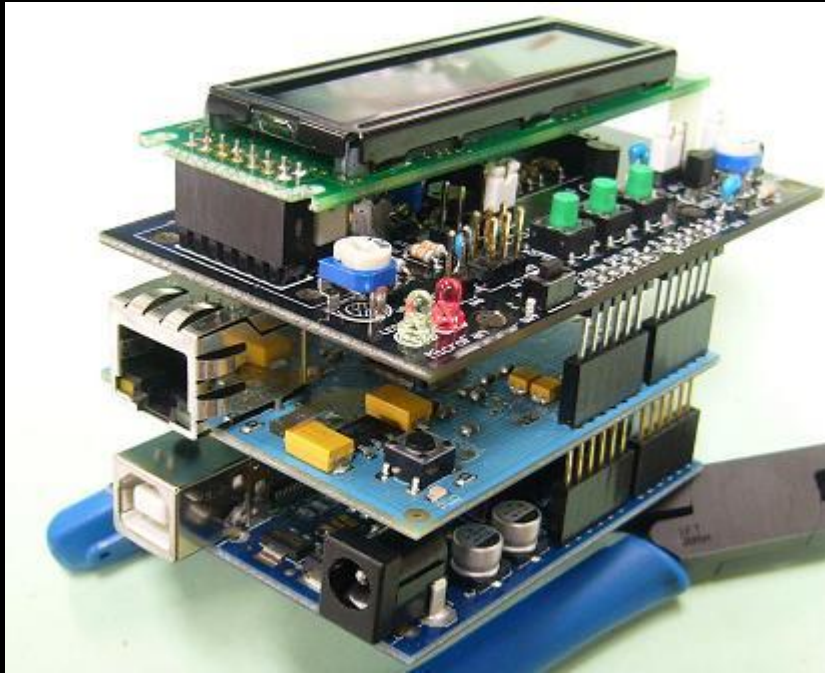


Marminino

Arduino com ATMEGA 328-P/-PU



Shields



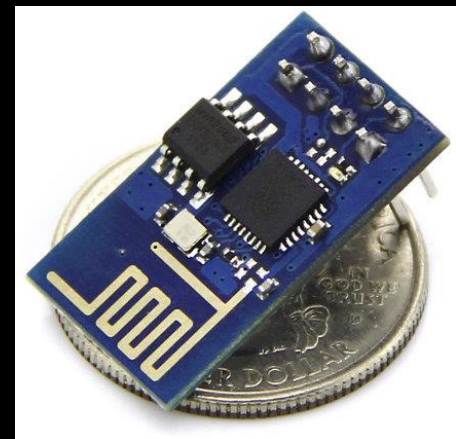
Módulos





ESP8266

Dedicado a IOT
Compatível com Arduino



Interface e Linguagem

Processing

- Linguagem baseada em C/C++
- Open-source
- Escrita em Java, Multi-plataforma
- Exigido conhecimento de programação (C, Java ou similar)
- Extensões para controle de entrada/saída e fluxo de dados

Wiring

- Projeto open-source derivado de Processing voltado para prototipação de circuitos eletrônicos
- Permite remapeamento e chaveamento dinâmico de entradas e saídas por comandos de software

Interface Arduino (Software)

- IDE – Integrated Development Interface
- Derivada de Wiring
- Escrita em Java, cross-platform
- Editor de código de programa verificador de sintaxe
- Incorporada comunicação com os dispositivos suportados
- Traz bibliotecas para operação de módulos e shields
- Suporte a múltiplos dispositivos/compiladores
- Integração com compilador/gravador AVR (AVRDude)



File Edit Sketch Tools Help



sketch_mar28b



```
void setup() {  
  // código de abertura, roda uma só vez:  
  
}  
  
void loop() {  
  // código de repetição, roda indefinidamente:  
  
}
```

Uploading...



Sketch uses 450 bytes (1%) of program storage space. Maximum is 30,720 bytes.
Global variables use 9 bytes (0%) of dynamic memory, leaving 2,039 bytes for local

Entrada e saída

ICSP
for USB interface

Reset

(I2C) SCL
(I2C) SDA

(SPI) SCK
(SPI) MISO
(SPI) MOSI
(SPI) SS

Interrupt 1
Interrupt 0

USB
to computer

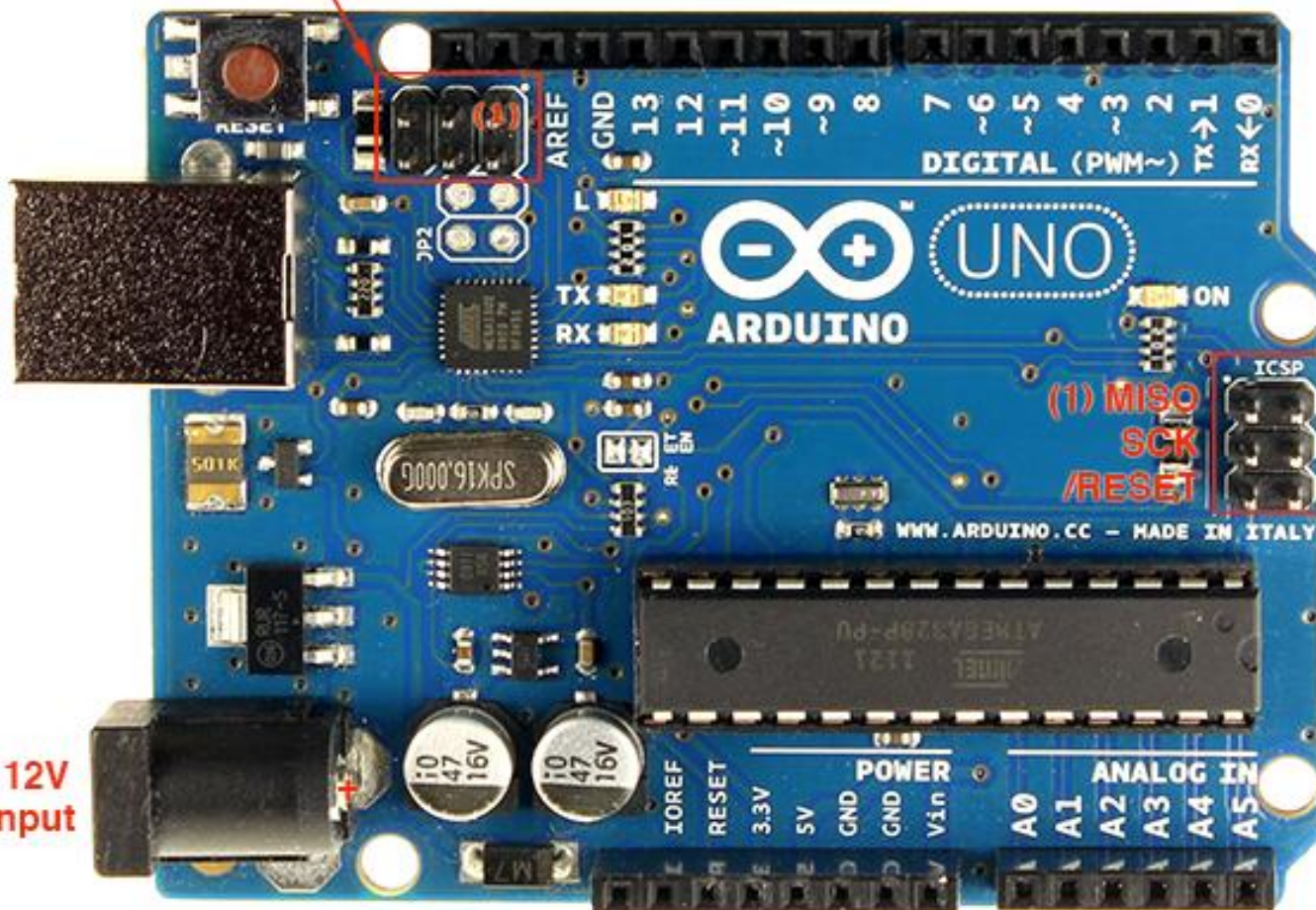
ICSP
for Atmega328

VCC
MOSI
GND

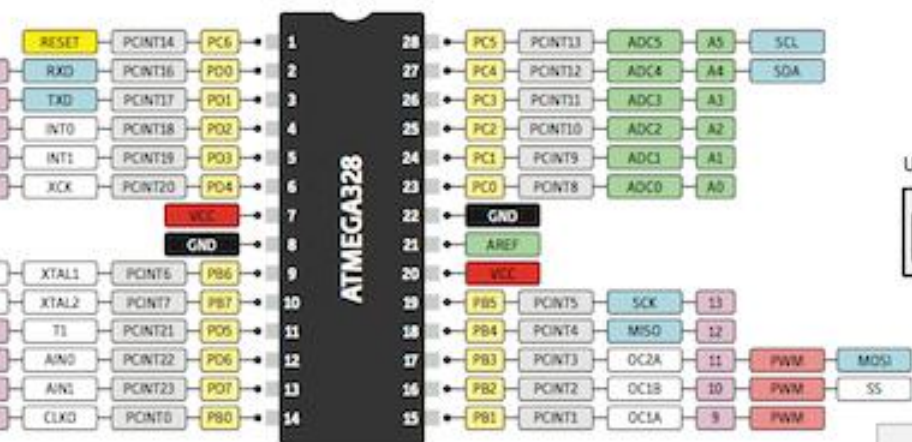
(1) MISO
SCK
/RESET

7 to 12V
DC input

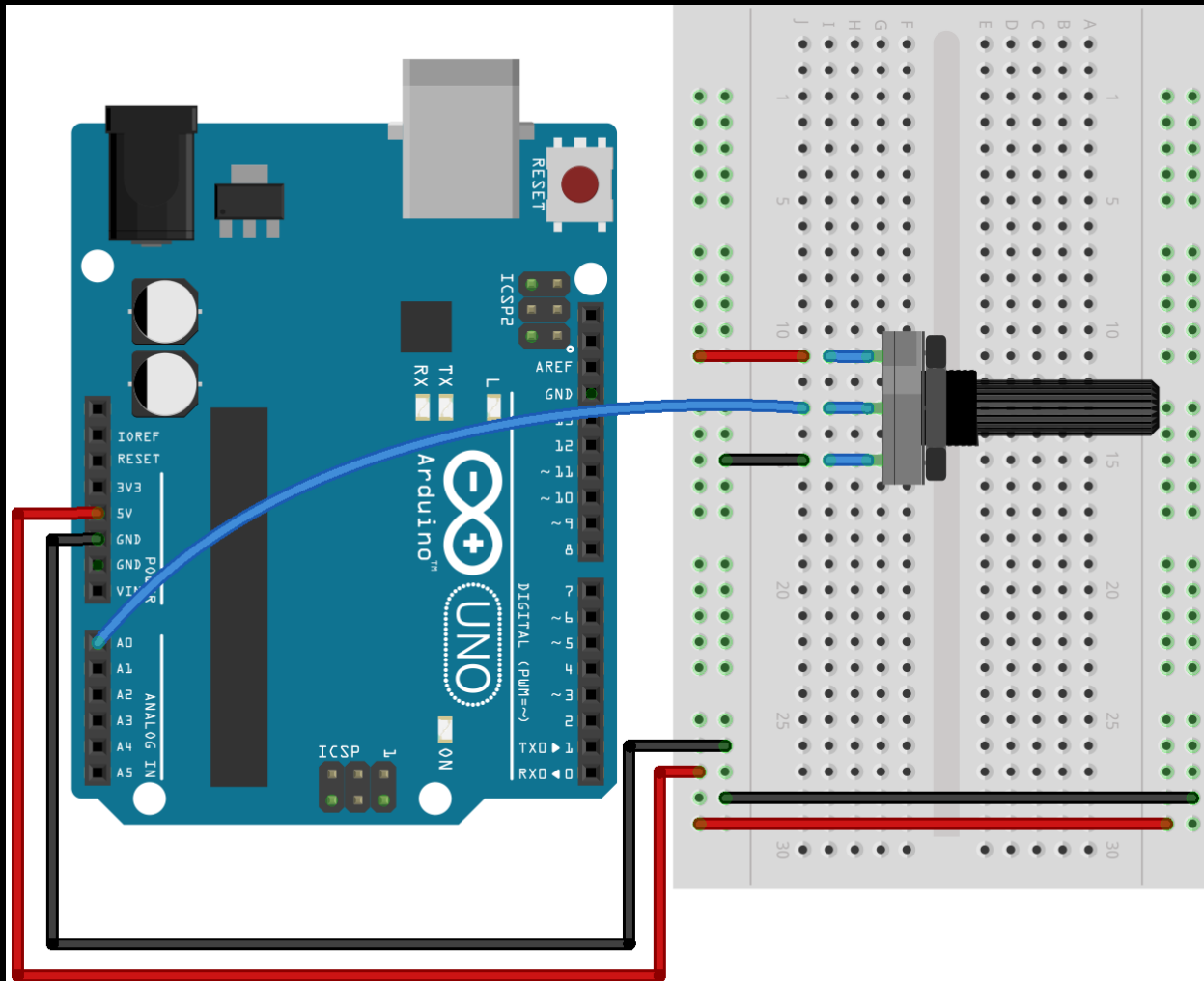
(I2C) SDA
(I2C) SCL



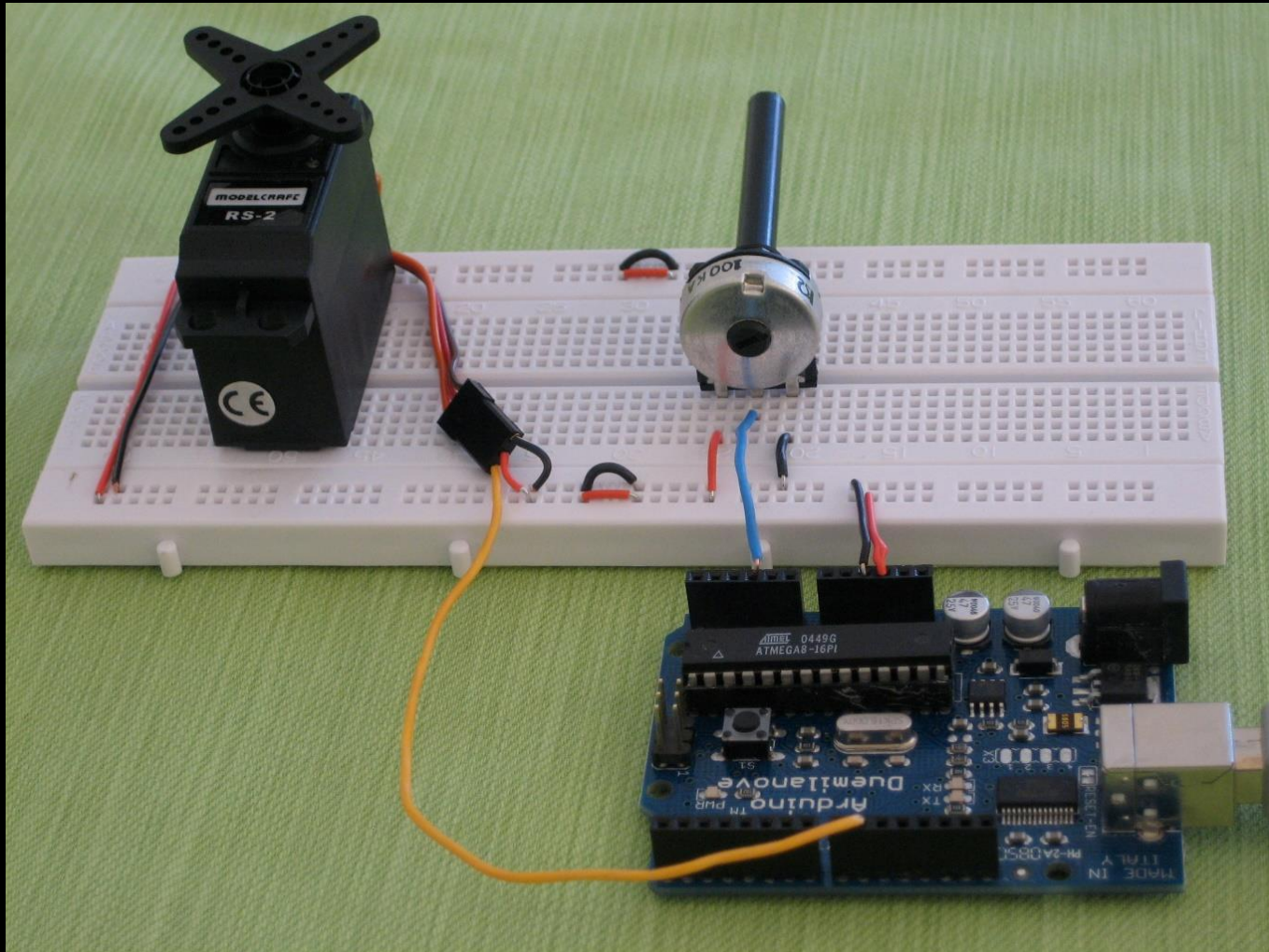




Exemplo de conexão: Potenciômetro



Exemplo: Servo



Código

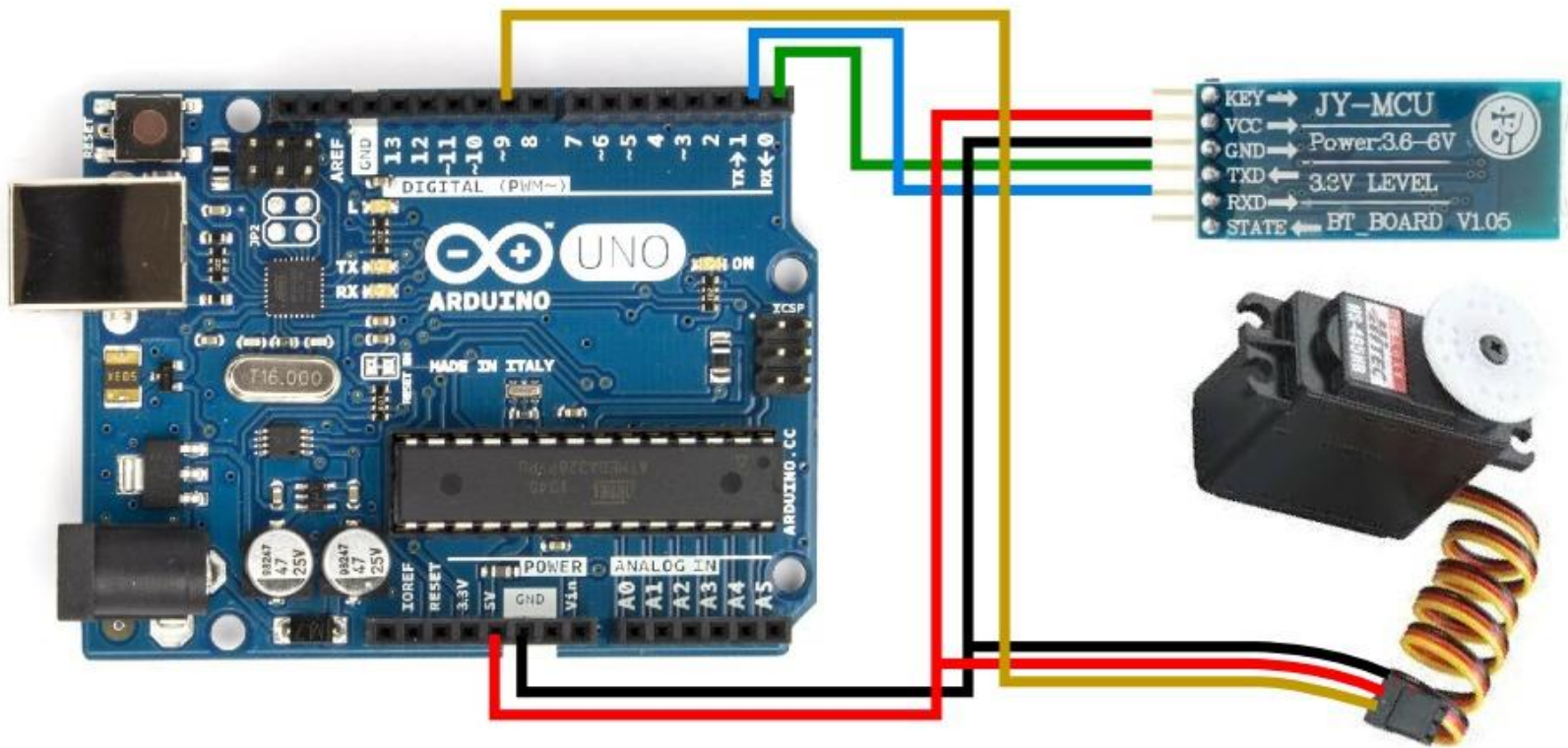


The screenshot shows the Arduino IDE window titled "Knob | Arduino 1.6.7". The menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar contains icons for checking, running, uploading, and downloading. The sketch name "Knob" is displayed in the top right. The code is as follows:

```
#include <Servo.h>
Servo myservo;
int potpin = 0;
int val;
void setup() {
  myservo.attach(9);
}
void loop() {
  val = analogRead(potpin);
  val = map(val, 0, 1023, 0, 180);
  myservo.write(val);
  delay(15);
}
```

A status message at the bottom of the code editor reads: "Code formatted for HTML has been copied to the clipboard." The bottom status bar shows "2" on the left and "Arduino Nano, ATmega328 on COM1" on the right.

Servo + Bluetooth



Exemplos de projetos

- <https://dheera.net/projects/sesame>
- https://youtu.be/nnbeROOT_dg
- <https://youtu.be/hqNZ1ul2BPc>
- <https://youtu.be/ySpeL8WkUEM>

Ferramentas

- Arduino IDE
- Fritzing
- <http://arduino.cc>

Prática 101

Basic Examples

Arduino IDE → Arquivos → Exemplos

- Basics/Blink

Libraries

Arduino IDE → Arquivos → Exemplos

- Servo/Knob