

WORKSHOP ARDUINO

Lcm OpenLab

Stefano Balzan <stefano.balzan@gmail.com>
Francesco Maiocchi <frasix96@gmail.com>

Università degli Studi di Milano
Laboratorio di Calcolo e Multimedia

14.04.2016 / 18.04.2016



Cosa faremo oggi?

Vedremo

- cos'è Arduino
- cosa "nasconde" all'interno e le caratteristiche tecniche
- cosa permette di realizzare e quali sono le potenzialità
- perché interessa a noi, studenti di fisica

Sperimenteremo con

- led e resistenze
- display lcd
- strumenti di misura
- attuatori meccanici
- interfacce Bluetooth

Arduino

Qualche informazione di base...

- Cos'è di preciso?
- Cosa permette di realizzare?
- Qual è il segreto del suo successo?
- Perché potrebbe interessarmi?

La piattaforma Arduino

cos'è e cosa offre



Una scheda elettronica con un microcontrollore programmabile che può pilotare dell'hardware

Un'ambiente di sviluppo software e delle librerie che rendono rapido e semplice programmare la scheda

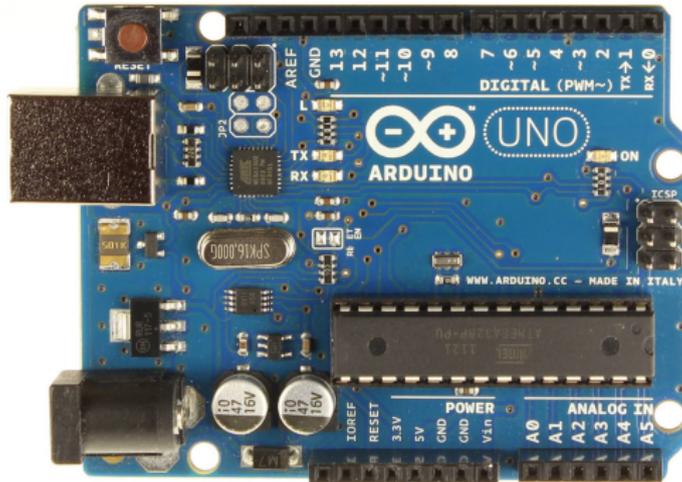
```
Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 * This example code is in the public domain.
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // set the LED on
  delay(1000);            // wait for a second
  digitalWrite(13, LOW);  // set the LED off
  delay(1000);           // wait for a second
}
```

L'hardware

Cosa c'è dentro Arduino?



- un microcontrollore(μ C) programmabile Atmega328p
- un circuito per la programmazione seriale (USB)
- circuiteria di alimentazione e collegamento

Un po' impropriamente si direbbe che Arduino costituisce un "wrapper" del μ C Atmega328p, "newbie friendly" e "Low level agnostic".

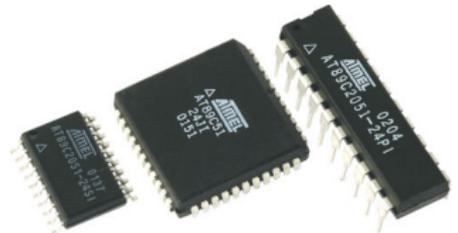
Cos'è un microcontrollore?



- piccola unità di controllo e elaborazione
- si programma in assembly o al più in C
- dotato di pin che possono leggere un segnale in ingresso o controllare hardware esterno

l'Atmega328p:

- CPU 8-bit
- pochi kB di memoria
- Porte di input/output (I/O) analogiche e digitali



Interfacciarsi con il mondo

cosa può controllare un μ C come l'Atmega?

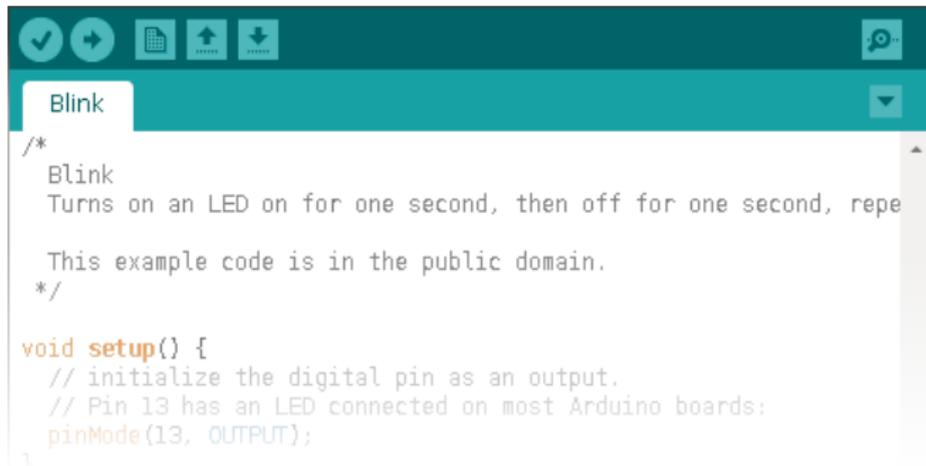
- semplici sensori come fotoresistenze o termocoppie
- led, piccoli buzzers, carichi pesanti attraverso un transistor
- motori DC, motori passo-passo, servocomandi, attuatori idraulici
- display LCD, TFT, touchscreen
- interfacce di rete (ethernet, wifi e bluetooth)
- collegamento a computer o qualsiasi macchina dotata di interfacce seriali/I2C
- ...

The sky is the limit!

Perché la board Arduino allora?

- fornisce una tensione di 5V e 3.3V regolata attraverso una porta USB
- contiene già il circuito di alimentazione e gli oscillatori per l'Atmega
- permette di programmare il μ C con estrema semplicità
- espone in maniera semplice ed ordinata i pin

Il software cosa fa funzionare Arduino?



The screenshot shows the Arduino IDE interface. At the top, there is a dark teal header bar with several icons: a checkmark, a right-pointing arrow, a grid icon, two upload/download icons, and a speech bubble icon. Below the header, a white box contains the text "Blink" and a dropdown arrow. The main area is a white text editor with a light gray scrollbar on the right. It contains the following code:

```
/*  
  Blink  
  Turns on an LED on for one second, then off for one second, repe  
  
  This example code is in the public domain.  
  */  
  
void setup() {  
  // initialize the digital pin as an output.  
  // Pin 13 has an LED connected on most Arduino boards:  
  pinMode(13, OUTPUT);  
}
```

- un ambiente semplice e completo per scrivere codice in C/C++
- un meccanismo di compilazione e upload semplice e immediato
- funzioni e librerie di alto livello e "newbie friendly"

Alcune librerie...

"standard libraries"

- [EEPROM](#) - reading and writing to "permanent" storage
- [Ethernet](#) - for connecting to the internet using the Arduino Ethernet Shield
- [Firmata](#) - for communicating with applications on the computer using a standard serial protocol.
- [GSM](#) - for connecting to a GSM/GRPS network with the GSM shield.
- [LiquidCrystal](#) - for controlling liquid crystal displays (LCDs)
- [SD](#) - for reading and writing SD cards
- [Servo](#) - for controlling servo motors
- [SPI](#) - for communicating with devices using the Serial Peripheral Interface (SPI) Bus
- [SoftwareSerial](#) - for serial communication on any digital pins. Version 1.0 and later of Arduino incorporate [Mikal Hart's NewSoftSerial](#) library as SoftwareSerial.
- [Stepper](#) - for controlling stepper motors
- [TFT](#) - for drawing text , images, and shapes on the Arduino TFT screen
- [WiFi](#) - for connecting to the internet using the Arduino WiFi shield
- [Wire](#) - Two Wire Interface (TWI/I2C) for sending and receiving data over a net of devices or sensors.

Mettiamoci al lavoro

Pilotiamo un semplice diodo led

Dovrete dividervi in quattro gruppi, a ciascuno verrà assegnato un kit.
Ci servirà parte del materiale che trovate all'interno.

Adesso:

- progettiamo il circuito e poi lo realizziamo fisicamente
- scriviamo quindi il codice per controllarlo
- colleghiamo l'arduino al cavo USB e lo programiamo
- inseriamo l'Arduino nel circuito e lo alimentiamo

Il codice

Cosa abbiamo scritto?

```
int led = 13;

void setup()
{
  pinMode(led, OUTPUT);
}

void loop()
{
  digitalWrite(led, HIGH);
  delay(1000);
  digitalWrite(led, LOW);
  delay(1000);
}
```

- due sezioni: setup e loop
- alcune variabili
- diverse chiamate a funzione

Un altro esempio

controlliamo un display LCD

Ora che abbiamo preso confidenza, vediamo qualcosa di più avanzato.

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  lcd.begin(16, 2);
  lcd.print("Hello Arduino!");
}

void loop()
{
  lcd.setCursor(0, 1);
  lcd.print(millis()/1000);
}
```

L'approccio è lo stesso di prima:

- vediamo come funziona un display LCD
- progettiamo il circuito e realizziamolo
- scriviamo il codice
- carichiamolo sull'Arduino e proviamolo

Volete vedere qualcosa di simpatico? Caricate lo sketch lcdDemo.ino

L'attività di oggi

Nel tempo restante ciascun gruppo lavorerà ad un progetto differente.

Realizzeremo:

- conduttivimetro per liquidi
- misuratore di distanza ad ultrasuoni
- stabilizzatore a 1 asse
- display remoto via Bluetooth

Tenete i display LCD collegati, vi serviranno. All'interno della scatola trovate un foglio con le istruzioni per montare i sensori aggiuntivi. Il codice sorgente è già stato scritto ed è presente tra i file forniti.

Suggerimenti e risorse utili

Internet è pieno di risorse su ogni tema o concetto affrontato da questo workshop, se qualcosa vi incuriosisce basta aprire Google!

Alcuni link o fonti utili:

- sito ufficiale Arduino: arduino.cc
- [arduino.cc Tutorials](http://arduino.cc/Tutorials) e reference
- blog e siti web di makers e appassionati
- comunità online: [forum Arduino](http://forum.arduino.cc), IRC, mailing list ...
- piattaforme web dedicate al making: instructables.com, hackaday.io
- Github (sorgenti)

Bonus tip: cercate in lingua inglese, produce risultati migliori

Bonus tip 2: internet è bello, ma magari qualcuno che vi consiglia o che collabori con voi potreste trovarlo anche in real life!

Fantastico! Vorrei assolutamente un Arduino!

Dove lo compro? Quanto costa? Mi sarà veramente utile?