

# Arduino

*Una filosofia* intorno ad un  
 $\mu$ -controllore

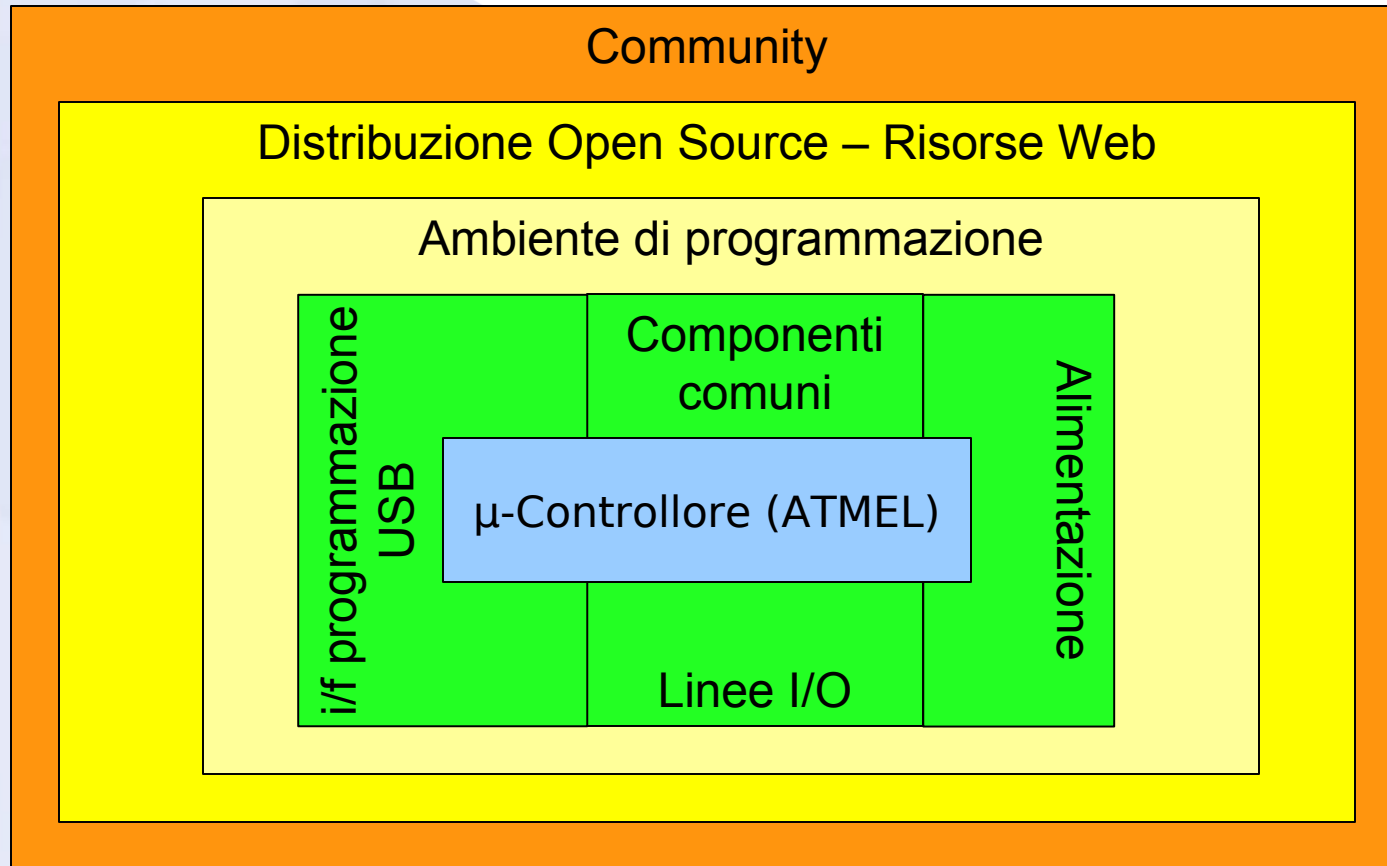
# Cos'è Arduino

- Piattaforma di prototipazione orientata all'automazione
- Basata su  $\mu$ -controllore ATMEL ATmega168/328P
- Progettato e distribuito OpenSource
- Non è uno strumento specifico per la radiocomunicazione

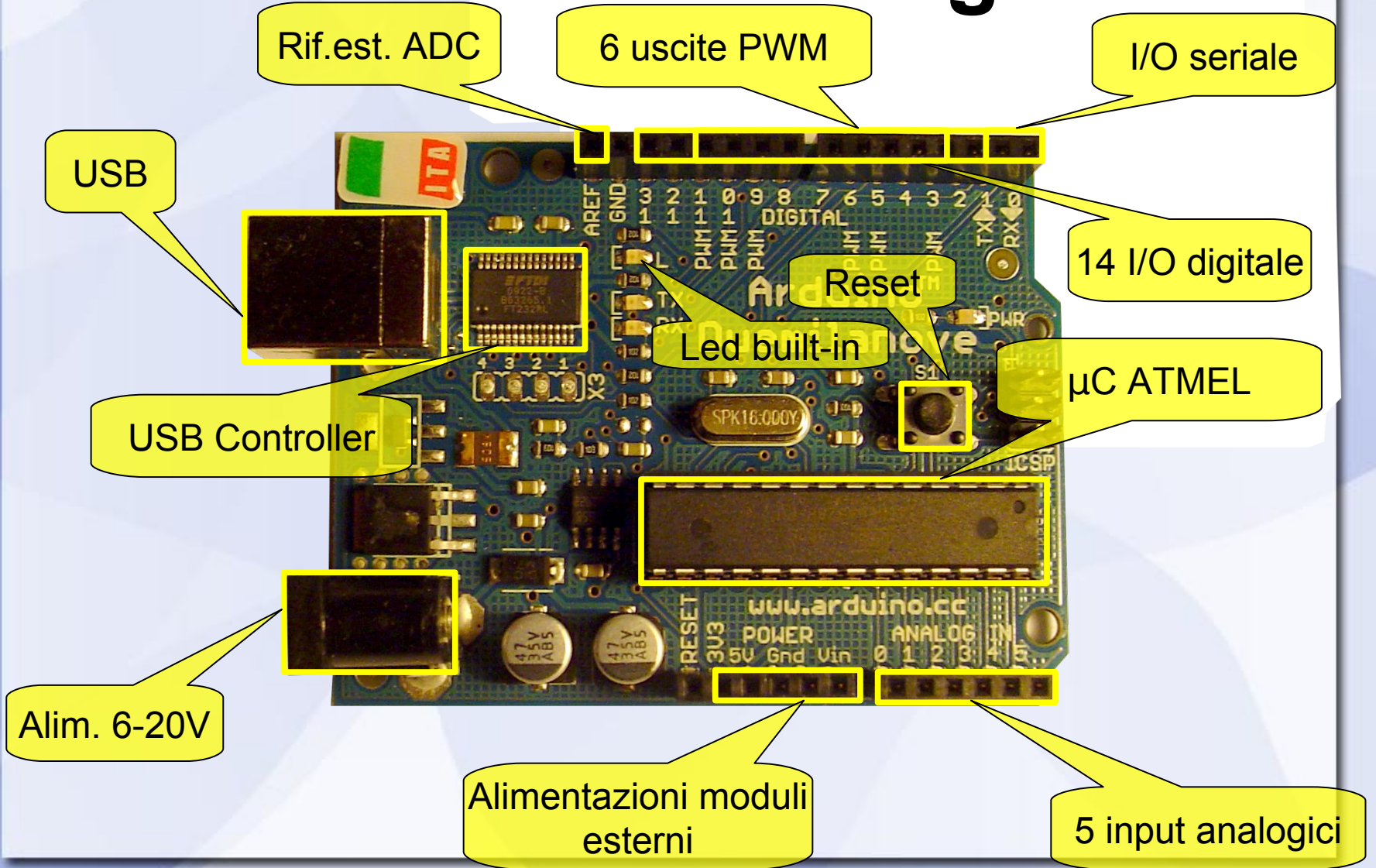
# Aspetti chiave

- E' uno strumento per il progetto di automazione
  - non è un componente nè un modulo.
- HW Open Source: è ammessa la riproduzione senza royalty
- Multiplatforma: Windows, Linux, Mac
- HW economico: la scheda costa circa € 26
- Vasta comunità di utenti, di progetti, di idee e soluzioni che non richiedono di partire da zero
- E' un idea *made in Italy*

# L'architettura di un idea



# Arduino in dettaglio



# Caratteristiche tecniche

*Riferite al modello "Duemilanove"*

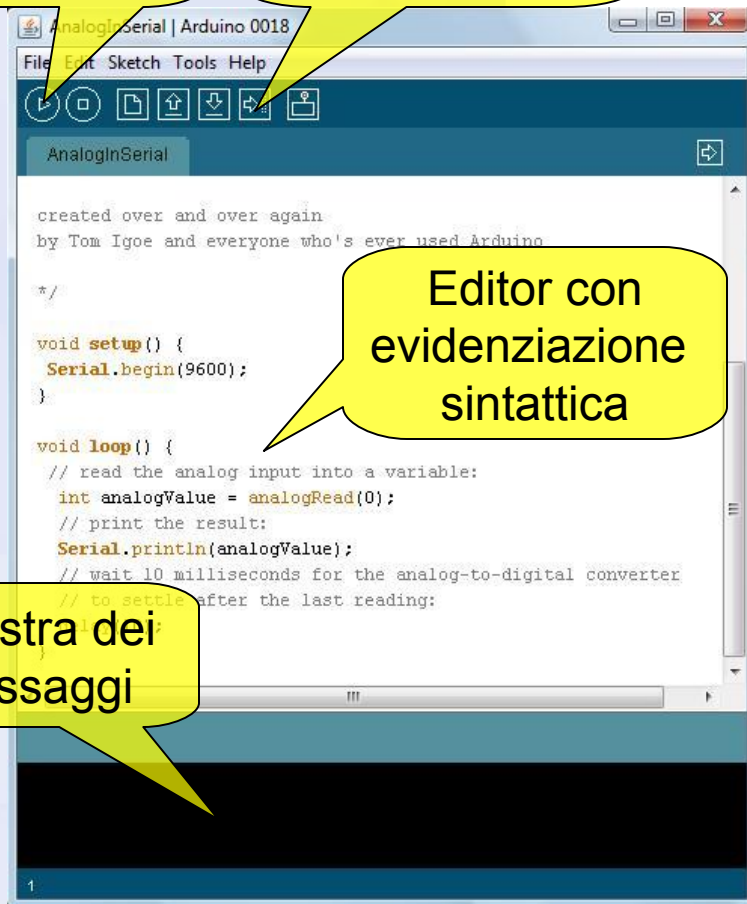
- Microcontroller  
ATMega168/328
  - Tensione su scheda: 5V
  - Corrente Max I/O Pin: 40 mA
  - Flash Memory 16 KB  
(ATmega168) o 32 KB  
(ATmega328) – 2K usati per  
bootloader
  - SRAM : 1 KB (ATmega168) o  
2 KB (ATmega328)
- EEPROM: 512 bytes  
(ATmega168) o 1 KB  
(ATmega328)
- Clock a 16 MHz
- 2 interrupt esterni (pin 2 e 3)
- ICSP disponibile
- Comunicazione I<sup>2</sup>C (pin 4 e 5)
- Comunicazione seriale
- PWM integrato a 490Hz
- USB



# L'ambiente di sviluppo

Compilazione

Upload al micro



Editor con evidenziazione sintattica

Finestra dei messaggi

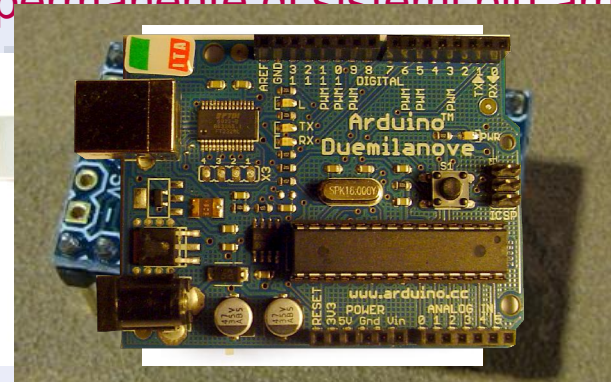
- Semplice ed intuitivo
- Realizzato in Java, identico su qualsiasi S.O.
- Pochi click per programmare la scheda
- Il programma viene chiamato *sketch* (ingl.: abbozzo, schizzo)

# Tanti modelli di Arduino

- Il “*Duemilanove*” è il più diffuso, con l'ATMega 328 (ultimi modelli)
- Il “*BT*” è una versione dotata di modulo Bluetooth
- Il “*Mega*” è una versione più potente, con l'ATMega1280, con 54 linee I/O, 16 analogiche, 4 seriali, 128K Flash, 8K SRAM e 4K di EEPROM

Il “*Nano*” è una versione compatta del “*Duemilanove*” senza alimentazione esterna e con mini-USB

- Il “*Mini*” è la più piccola versione, con ATMega168 e solo le linee di I/O e analogiche
- Il “*LilyPad*” è una versione miniaturizzata per essere integrata in tessuti e materiali sottili, alimentato a 3.3V e 8MHz di clock
- Il “*Pro*” è una versione su scheda modulare che permette l'uso di Arduino come modulo permanente di sistemi più ampi





# La Storia

- Nasce nel 2005 da un'idea di **Massimo Banzi**, ingegnere insegnante in un istituto tecnico di Ivrea (“Arduino” era un pub vicino alla scuola di Banzi)

Rispondeva alla necessità di studenti di design pubblicitario di realizzare piccole automazioni in breve tempo

- La voce si diffuse tra i vari ambienti di design pubblicitario ed architettonico e in pochi mesi vendettero le prime 200 schede – Era stato scoperto un *mercato!*

- Fu creata una piccola società **Tinker.it** che produceva la scheda e la distribuiva con licenza *Attribution Share Alike* (*CreativeCommons*)

- L'unico elemento protetto da copyright è il nome “**Arduino**”

- La licenza CC impone solo l'obbligo di mostrare nella riproduzione il riferimento esplicito al gruppo **Tinker.it**

# La Storia - 2

Nasce così una nutrita comunità di sperimentatori in tutto il mondo che contribuisce continuamente al suo miglioramento e diffusione – *a costo 0*, e fa continuo riferimento alla community ufficiale

- Nel 2006 erano state *vendute* +5000 copie, nel 2007, +30000

- Tentativi di sfruttare l'idea in modo proprietario (Cina...) fallivano per la minore qualità e i margini troppo esigui per competere utilizzando un modello industriale classico
- Oggi [Tinker.it](http://Tinker.it) è un affermato studio di design che fa di Arduino il suo cavallo di battaglia
- Arduino continua a diffondersi e a creare comunità di sperimentazione

# Open Source nell'HW

- Il principio di business è che il servizio offre maggior guadagno della semplice produzione
- I ritorni della comunità degli utenti permette un processo di miglioramento rapido e a costo quasi nullo
- La semplice copia senza la stessa comunità di utenti non porta gli stessi benefici
- Quello che conta è associare il *nome* alla qualità
  - Il *nome* è l'unica cosa protetta da copyright

# La filosofia di Arduino

- **Fare** progetti - anzichè limitarsi a parlarne...
- “Cortocircuitare” la rigidità della progettazione elettronica classica e partire da quello che si vede, sente e tocca
- Liberare la mente dall'apprendimento non necessario, quando l'obiettivo è più importante della conoscenza dello strumento
- L'approccio accademico o professionale non è sempre necessario e non deve essere un ostacolo alla voglia di provare

# Le parole chiave di Arduino

- La filosofia di Arduino ha i suoi concetti fondamentali
  - Tinkering
  - Patching
  - Circuit bending
  - Physical Computing
  - Collaborazione



# Tinkering

- *Tinkering* è la definizione che **Banzi** dà dei suoi primi approcci giovanili all'elettronica.
- **To tinker**: armeggiare, cercare di riparare
- *“Tinkering è quanto accade quando si cerca di fare qualcosa che non si sa bene come fare, guidati dal capriccio, dall'immaginazione e dalla curiosità... Tinkering, fondamentalmente, è un processo che coniuga gioco e ricerca.”* [www.exploratorium.edu/tinkering](http://www.exploratorium.edu/tinkering)
- *Tinkering* è intraprendere un percorso di esplorazione, senza un obiettivo preciso, per scoprire meccanismi che nemmeno ci si aspettava

# Molti di noi all'inizio sono stati *tinkerers*

- La scoperta della radio “amatoriale” che nasce dal voler vedere *cosa c'è dentro...*
- ...e dal *ricombinare* i diversi componenti per vedere cosa succede...
- ...e alla fine scoprire vecchie e nuove cose con la propria esperienza

# Patching

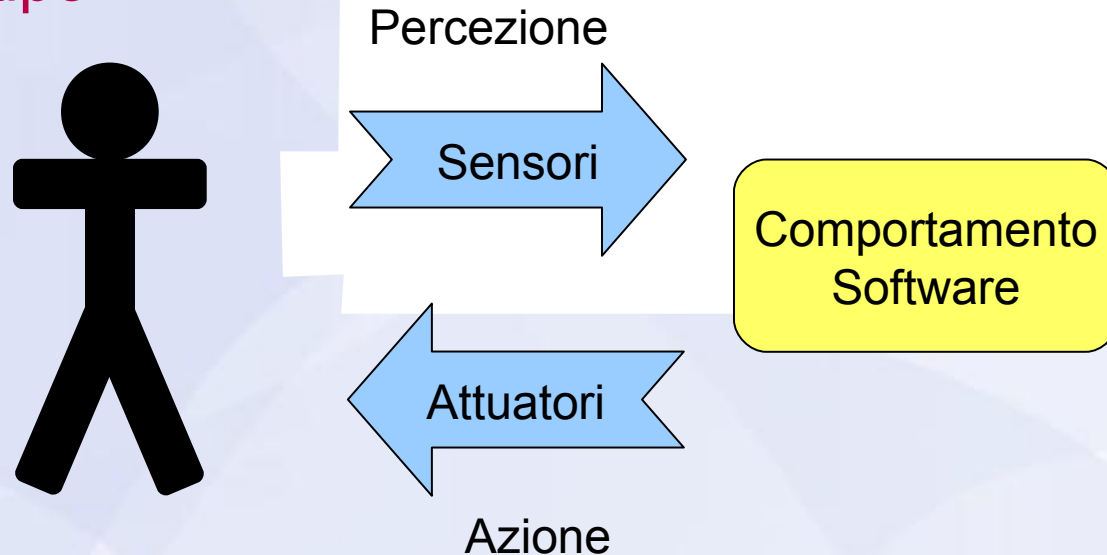
- *Patching* = collegamento di interfacce
- Sperimentazione attraverso il collegamento di moduli e adattamento di interfacce
- Sperimentazione più potente se possiamo saltare il dettaglio componentistico dei singoli moduli

# ***Circuit bending***

- E' l'aspetto meno ortodosso di questa filosofia
- Si tratta di creare collegamenti non previsti in un circuito esistente per scoprirne comportamenti nuovi
- Potrebbe risolversi in una nuvola di fumo... Ma a volte si fanno scoperte entusiasmanti

# Physical Computing

- *Physical Computing* significa applicare procedure software per interagire con grandezze fisiche di qualsiasi tipo



- *Sensore ed attuatore* diventano gli elementi chiave di progettazione e influenzano il *comportamento* software



# Automazione per “non addetti ai lavori”

- Arduino permette di realizzare automazioni senza avere una profonda conoscenza di elettronica digitale
  - Assembler sostituito dal *Processing*, un linguaggio simile a Java e al C
  - Programmazione del  $\mu$ C completamente gestita dal programma: colleghi la USB e clicchi!
  - Funzioni comuni già disponibili come librerie facilmente utilizzabili (LCD, comunicazione seriale, ...),
  - ...il resto probabilmente lo trovi su Internet!

# Automazione subito

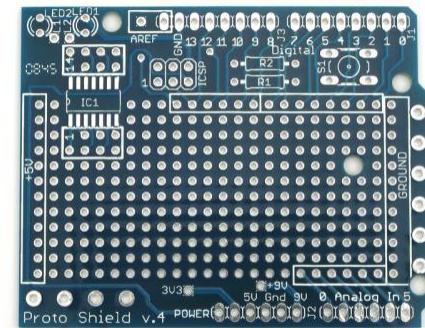
- Nella maggioranza dei casi serve solo una conoscenza di programmazione di base
- ...E tanta voglia di provare e riprovare
- Numerosi esempi funzionanti
- **IMPOSSIBILE** che non venga voglia di provare qualcosa!

# Librerie disponibili

- Ethernet (IP)
  - Serial
  - LiquidCristal
  - Led control
  - Servo
  - Stepper e Servo
  - Wire (I<sup>2</sup>C)
- Messenger
  - PS2 keyboard
  - Webduino
  - XBee
  - SerialControl
  - altre...*

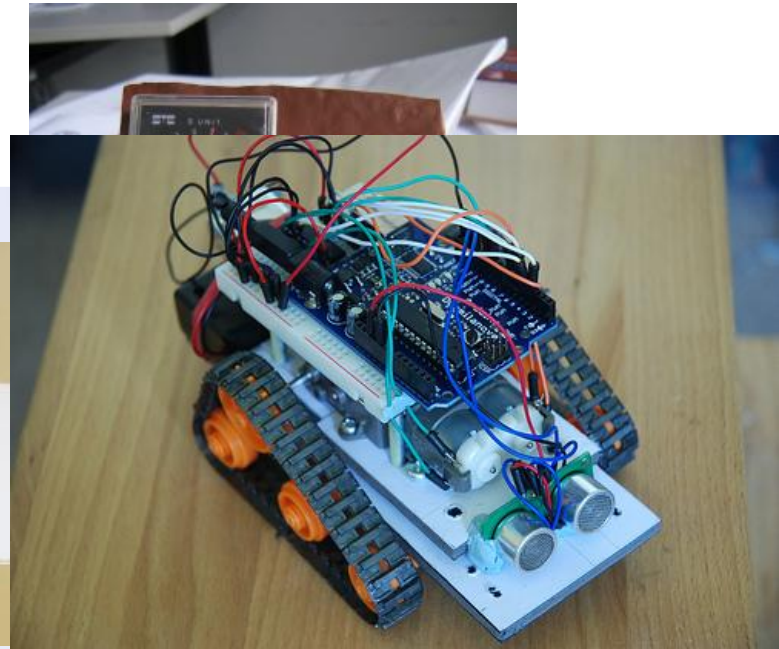
# Gli Shields

- Gli Shields sono estensioni hardware che si collegano ai connettori in linea
- Estendono le funzioni base con funzioni specializzate
  - *Ethernet*, con la libreria IP
  - *Motor driver*, per 2 motori in CC
  - *GPS*, di fatto un connettore per il modulo della US GlobalSat (EM-406)
  - *XBee*, per la connessione ZigBee (IEEE 802.15.4, wireless personal area networks - WPAN).
  - *MP3*
  - *Proto*, universale, fai da te...



# Progetti di interesse

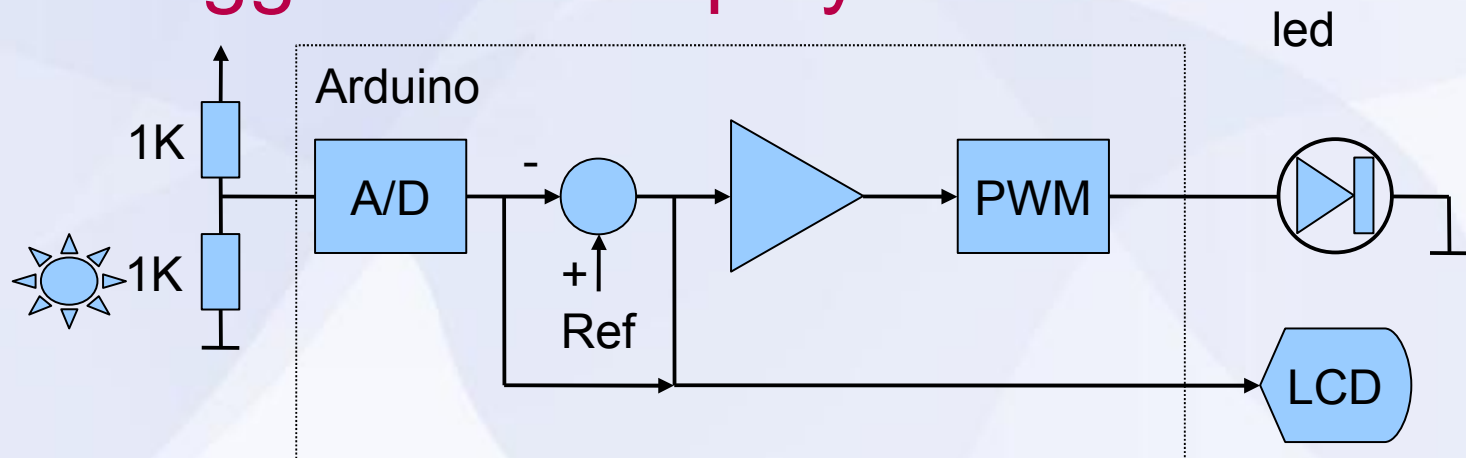
- Ci sono tantissimi progetti basati su Arduino nel mondo
- <http://www.arduino.cc/playground/Projects/ArduinoUsers>
- <http://hacknmod.com/hack/top-40-arduino-projects-of-the-web/>
- Fluidforms: design di paralumi con i guantoni!
- InboxMeter: mail monitor interattivo
- LarryBot: uno dei tanti robot basati su Arduino





# Un esempio di realizzazione

- Tipico esempio di physical computing
- Lettura di sensori analogici
- Pilotaggio di uscite PWM
- Pilotaggio di un display LCD



# Riferimenti

- Sito ufficiale Arduino: [www.arduino.cc](http://www.arduino.cc)
- Sito di Tinker.it: [www.tinker.it](http://www.tinker.it)
- **“Arduino – La guida ufficiale”** - Massimo Banzi – O'Reilly/Tecniche nuove
- **“Hai un’idea geniale? Fai come Mr Arduino, regalala”**  
su <http://www.wired.it/magazine/archivio/2009/01.aspx>