

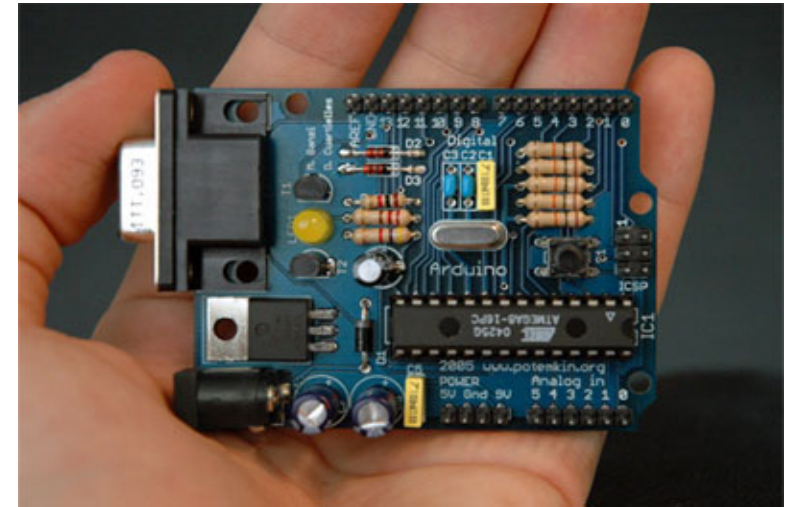
# **Microcontrollori: Arduino una piattaforma open-source**

*relatore Salvatore Balestrino*

# Cosa sono i micro-controllori

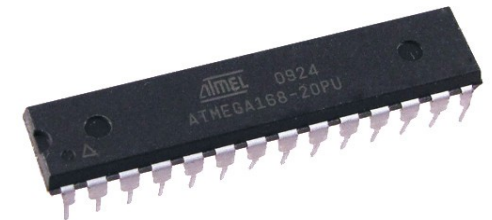
Il microcontrollore ( $\mu$ C) è un dispositivo logico digitale che raggruppa su di un unico chip un processore ed un insieme di dispositivi:

- CPU
- RAM
- EEPROM
- PORTE I/O e ADC
- BUS



E' integrato, ri-programmabile, costa poco, è piccolo ed ha un bassissimo consumo!

Sono adatti alla realizzazione di applicazioni di controllo e di acquisizione ed elaborazione dei segnali.



# Le differenze tra MCU e CPU

Le CPU dei nostri PC sono estremamente potenti ma hanno bisogno di molte integrazioni per poter funzionare.

Il microcontrollore invece è un dispositivo elettronico integrato e viene utilizzato generalmente in sistemi cosiddetti “**embedded**”, cioè per applicazioni specifiche di controllo digitale.

Le CPU consumano diverse decine di watt, i  $\mu$ C consumano milliWatt e sono progettati per operare in consumo estremamente ridotto adatto per operare con alimentazione a batteria.

Le CPU costano dai 50 € in su, i microcontrollori a partire da 0,50 €

I microcontrollori sono disponibili in tre fasce di capacità elaborativa: 8 bit, 16 bit e 32 bit.

# Linguaggi di programmazione

Uno dei punti di forza dei  $\mu\text{C}$  è la loro facilità nella riprogrammazione.

La programmazione dei  $\mu\text{C}$  è effettuata con specifici programmatori oppure direttamente tramite porta seriale o USB.

Il linguaggio più "*naturale*" per programmare un  $\mu\text{C}$  è l'assembler. Con l'assembler si lavora direttamente sul processore e i suoi registri, se ne ha il massimo controllo e la massima velocità a discapito della difficoltà!

Ci sono altri linguaggi molto usati ed estremamente validi come il C, il BASIC, il PASCAL e alcuni metodi di programmazione grafici.

**Il C è il linguaggio di programmazione attualmente il più usato**

# Funzioni speciali dei $\mu$ C

1. Ha diverse **porte digitali** in input e in output
2. Ha solitamente una decina di **ingressi analogici** a 10bit
3. Gestisce gli **interrupt** con diverse priorità
4. Ha diversi **timer** interni, a 8 bit, a 16 bit, etc.
5. Ha diversi **counter** interni, a 8 bit, a 16 bit, etc.
6. Può effettuare operazioni automatiche di comparazione **CCP**
7. Una memoria **EEPROM** dove salvare permanentemente i dati
8. **PWM**: pulse-width modulation. Modulazione a larghezza di impulso
9. Modalità risparmio energetico **SLEEP**
10. **WAKE on interrupt**: risveglio da modalità sleep tramite interrupt
11. Comunica con altri integrati tramite **SPI e I2C, RS232**
12. Supporto del protocollo **CAN**
13. Comunica con il mondo esterno tramite **USB e/o porta seriale**

# Cos'è Arduino

Arduino è una piccola scheda elettronica con  $\mu$ C **ATMEL** che rende facile la programmazione e la sperimentazione.

Oltre alla scheda elettronica, Arduino include anche un **ambiente di sviluppo** (IDE) nel quale è possibile scrivere i programmi (sketch). La scrittura dei programmi è estremamente semplice grazie anche alle librerie e ai molti esempi pratici disponibili in rete.

Attorno ad Arduino si è creata con il tempo una **community** che ha permesso a molte persone prive di conoscenze di elettronica e software di fare "innovazione" creando prodotti

**Arduino è un prodotto made in Italy!**

Arduino si è affermato come un vero e proprio standard, è usato nelle scuole, dai sviluppatori, dai creativi e dagli artisti di tutto il mondo.

E' un **grande risultato** soprattutto se si pensa che ciò è avvenuto al di fuori del mondo delle grandi case produttrici dell'universo IT.

# Open-hardware e open-source

Perchè Arduino è diventato uno dei riferimenti a livello internazionale?

**open hardware**  
**software open-source**  
**cross-platform**  
**economico, flessibile**  
**facilmente utilizzabile**  
**grande quantità di progetti e tutorial**  
**schede standard tra loro**  
**molti shield (addon) disponibili**

[www.arduino.cc](http://www.arduino.cc)