

Corso di Percezione Robotica

-

Prof.ssa Cecilia Laschi

23/04/2009

Introduzione ai Microcontroller

Dr. Virgilio Mattoli
(mattoli@crim.sssup.it)

Processori Embedded

- ✓ I microprocessori embedded sono contenuti in tutto ciò che ci circonda.
- ✓ I primi microprocessori sono apparsi negli anni '70 → Intel 8080
- ✓ Oggi sono integrati praticamente in ogni apparecchio elettronico presente sulle mercato: lavatrici, forni a microonde, telefonini, autovetture, ...
- ✓ Ogni anno vengono venduti nel mondo **miliardi** di processori embedded

Processori Embedded

Mediamente un microprocessore per essere definito embedded deve avere le seguenti caratteristiche:

→ Deve essere dedicato al controllo real-time di uno specifico dispositivo o funzione.

→ Deve contenere il proprio programma operativo in qualche tipo di memoria non volatile

→ Deve essere trasparente all'utente (deve funzionare come un hardware dedicato)

Processori Embedded

Un *sistema embedded* deve contenere solitamente le seguenti componenti:

- ✓ Un microprocessore
- ✓ Memoria RAM (random access memory)
- ✓ Memoria non-volatile : ROM, EEPROM,, FLASH, ...
- ✓ I/O (interfaccia con l'ambiente)

In cosa un sistema embedded è diverso da un computer (PC)? **Risorse**

	PC	Embedded System
RAM	GB	Centinaia di GByte
ROM	Centinaia di GByte	KByte.

Processori Embedded

In cosa un sistema embedded è diverso da un computer (PC)? **Applicazione!**

→ Personal Computer devono poter svolgere una varietà virtualmente illimitata di funzioni e programmi e applicazioni;

→ I sistemi embedded devono svolgere un limitato numero di task (as. controllo della temperatura di un forno, controllo del tempo di lavaggio di una lavatrice,)

Processori Embedded

Perche usare un sistema a microprocessore embedded?

Costo. – Microprocessore embedded è molto vantaggioso rispetto all'implementazione hardware in componenti discreti

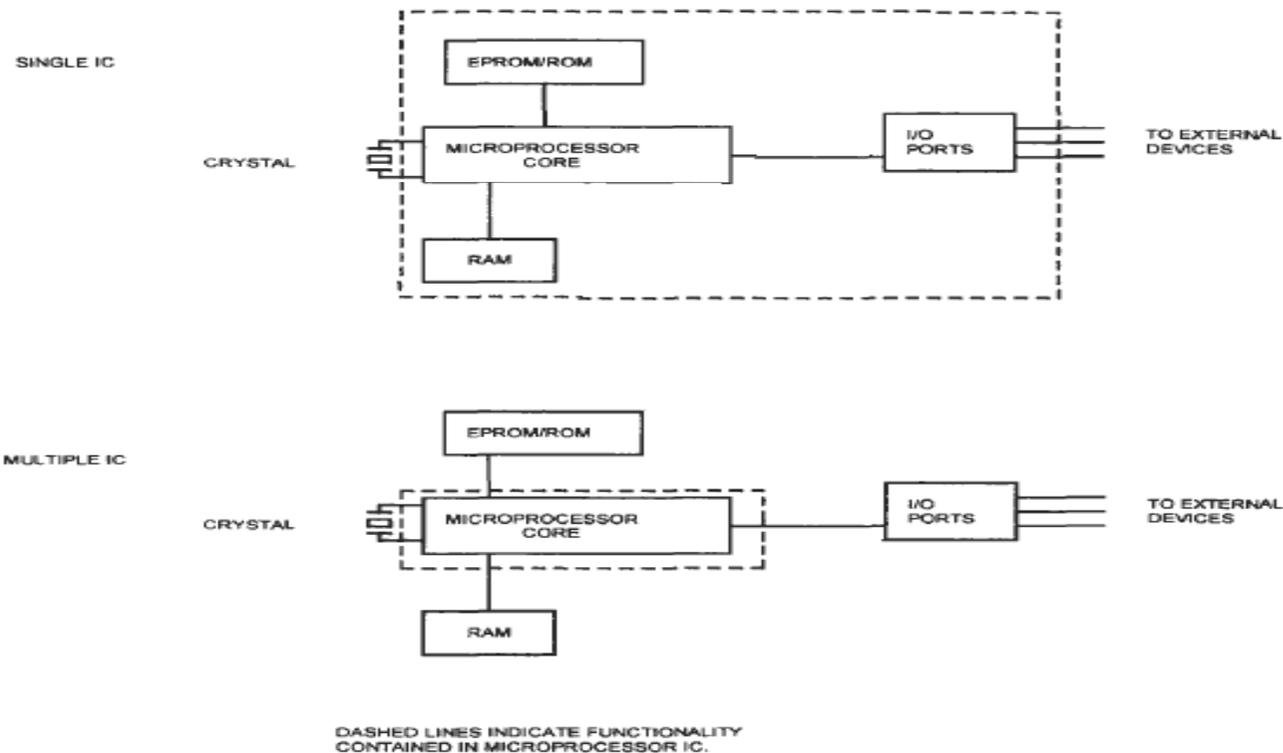
Programmabilità – La stessa piattaforma hardware permette di implementare differenti applicazioni

Flessibilità - Le funzionalità possono essere semplicemente ri-programmate in firmware

Adattabilità – Implementazione di sistemi intelligenti (“smart”) con capacità di adattarsi all'ambiente

Cosa è un microcontroller?

Microcontroller = Microprocessore Embedded in un singolo chip



Un microcontroller è progettato per minimizzare il numero dei componenti richiesti per la realizzazione di un sistema embedded, incorporando memoria e I/O.

Spesso sono specializzati per una certa applicazione (a scapito della flessibilità).

Cosa è un microcontroller?

Dimensioni e Packaging



P: 40-pin PDIP
(52.27 x 15.24 x 3.81 mm)



PF: 64-pin TQFP
(14 x 14 x 1 mm)



PT: 64-pin TQFP
(10 x 10 x 1 mm)



PF: 80-pin TQFP
(14 x 14 x 1 mm)



PF: 100-pin TQFP
(14 x 14 x 1 mm)



SO: 18-pin SOIC
(11.53 x 10.34 x 2.31 mm)



P: 18-pin PDIP
(22.81 x 7.95 x 3.3 mm)



S: 20-pin SSOP
(2 x 7.85 x 1.85 mm)



SO: 20-pin SOIC
(12.80 x 10.34 x 2.31 mm)



PT: 44-pin TQFP
(10 x 10 x 1 mm)



PT: 80-pin TQFP
(12 x 12 x 1 mm)



PT: 100-pin TQFP
(12 x 12 x 1 mm)



P: 20-pin PDIP
(26.24 x 7.87 x 3.3 mm)



ML: 28-pin QFN
(6 x 6 x 0.9 mm)



SO: 28-pin SOIC
(17.88 x 10.34 x 2.31 mm)



ML: 44-pin QFN
(8 x 8 x 0.9 mm)



SP: 28-pin SPDIP
(34.67 x 7.87 x 3.3 mm)



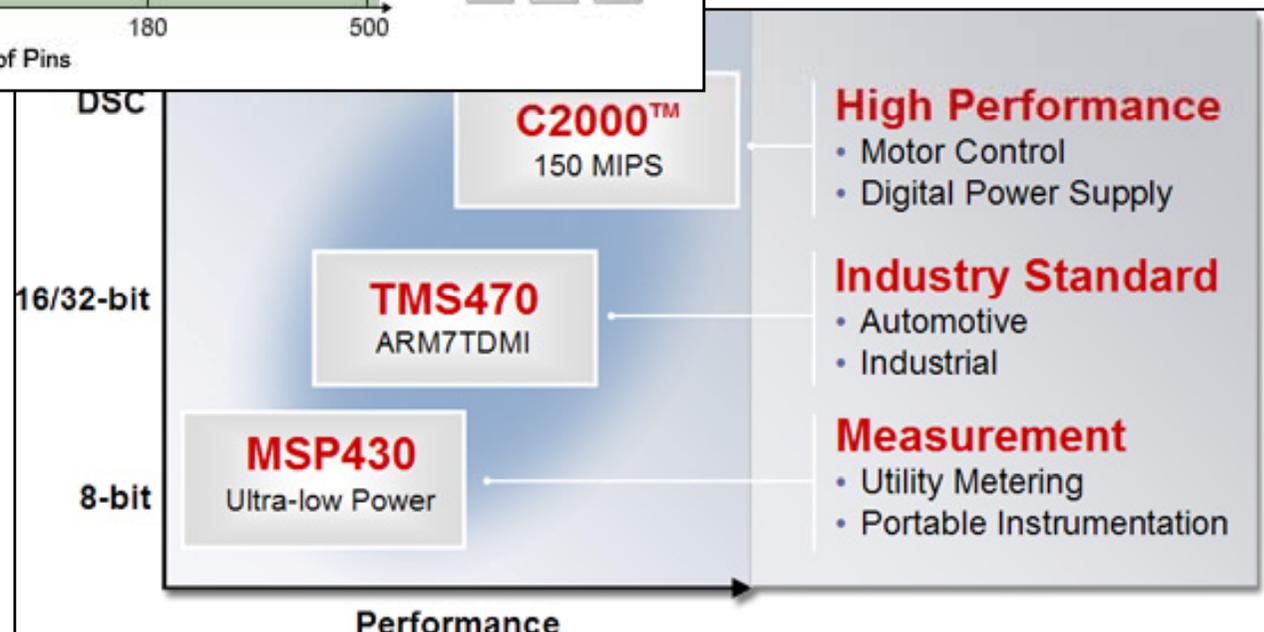
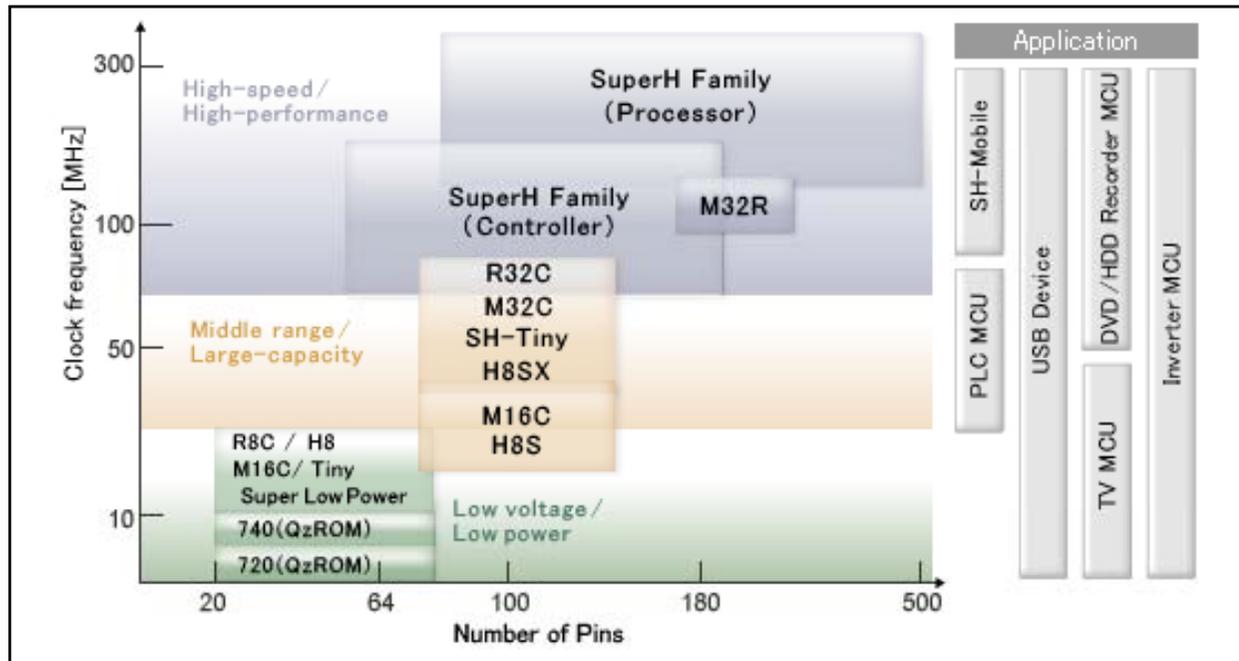
SS: 28-pin SSOP



MM: 28-pin QFN
(6 x 6 x 0.9 mm)

Cosa è un microcontroller?

8, 16 o 32 bit



Cosa è un microcontroller?

Principali Produttori



<http://www.microchip.com>



<http://www.ti.com/>



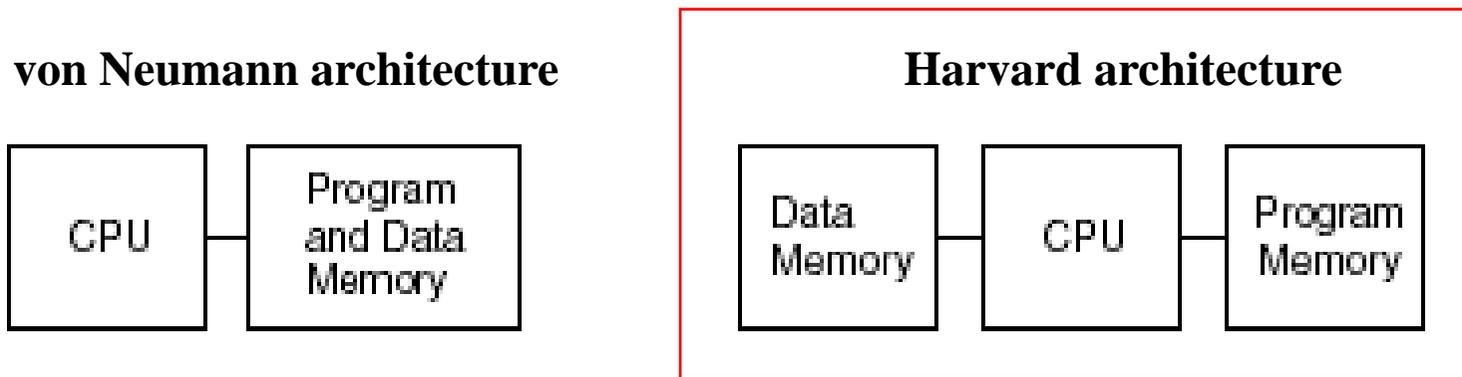
<http://www.atmel.com/>



<http://eu.renesas.com/>

Architettura di un microcontroller

I sistemi a microprocessore hanno solitamente un'architettura di tipo von Neumann con una singola memoria per il programma e i dati che permette la massima flessibilità di allocazione; i microcontroller hanno invece tipicamente un'architettura di tipo Harvard in cui la memoria di programma è separata da quella per i dati.



Il vantaggio dell'architettura Harvard per applicazioni embedded è dovuta alla possibilità di usare due tipi diversi di memoria per i dati e il programma

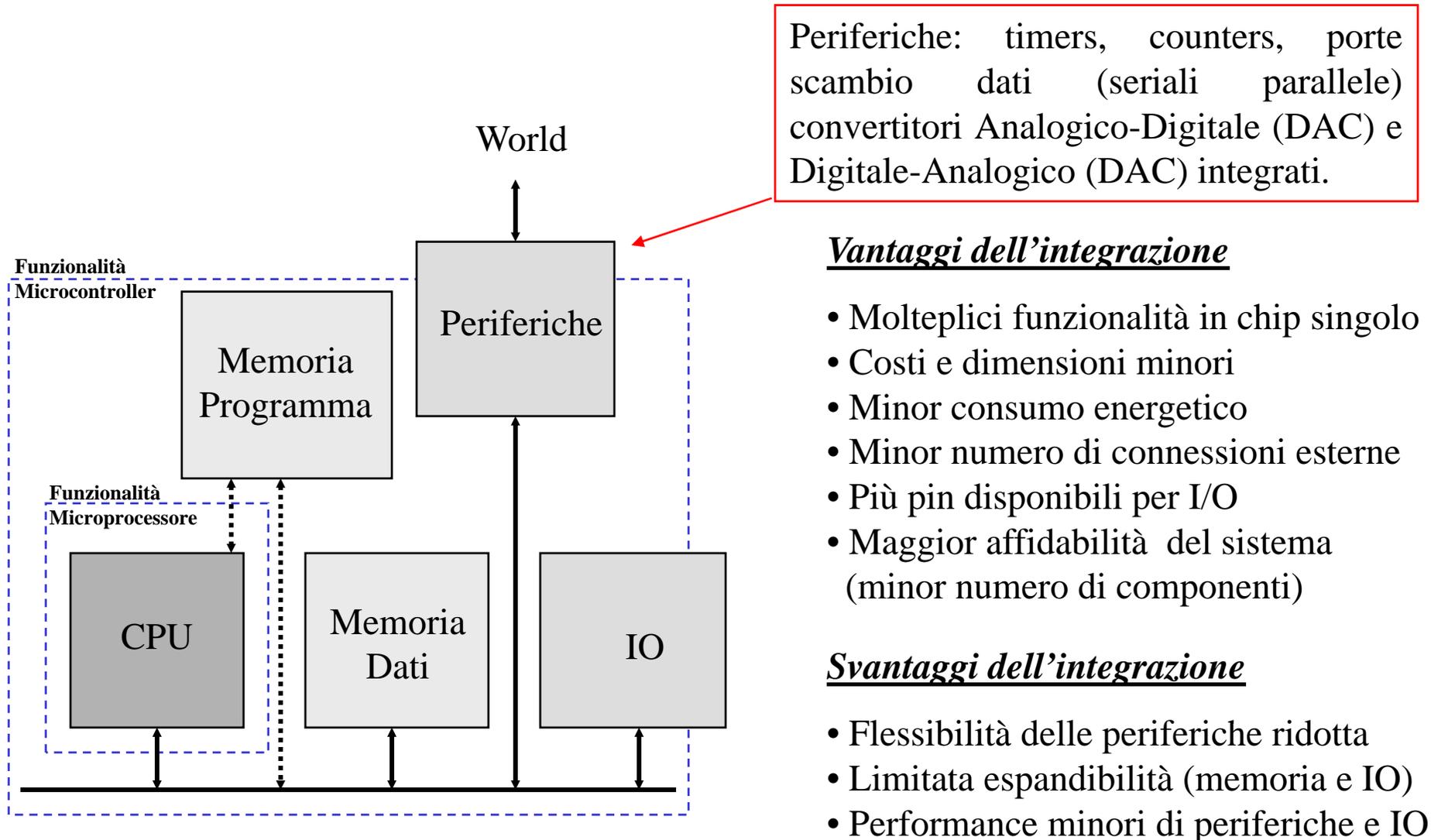
Programma → memoria non-volatile (ROM, programma non si perde allo spegnimento)

Dati variabili → RAM volatile

Un altro potenziale vantaggio dell'architettura Harvard è dato dal fatto che il trasferimento dei dati e delle istruzioni di programma avviene in parallelo (velocità doppia).

Architettura generica di un microcontroller

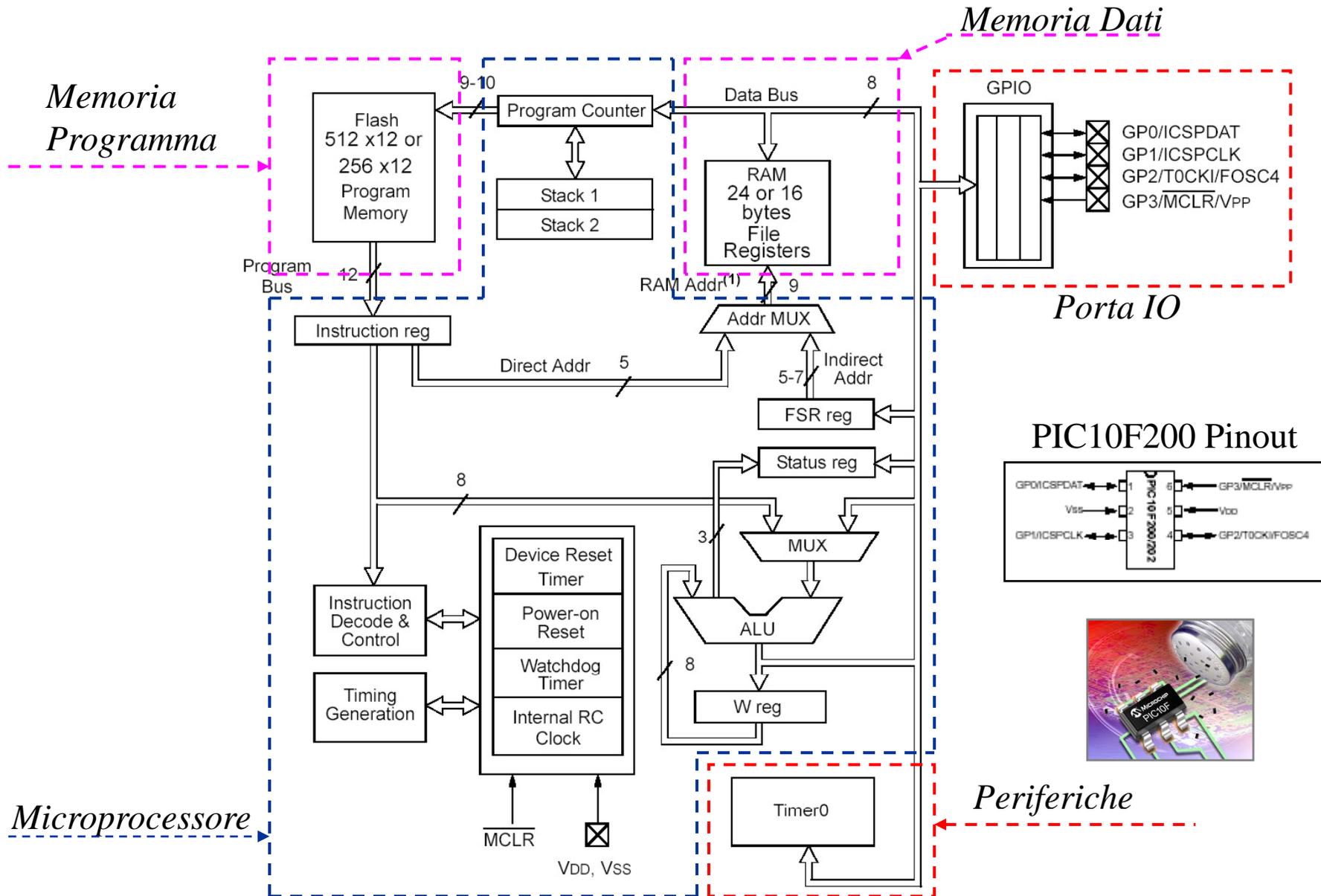
Tipica architettura di un microcontroller: CPU, memorie, I/O e periferiche per l'interfaccia con l'esterno. Componenti collegate da un Bus comune!



Tipologie di microcontroller

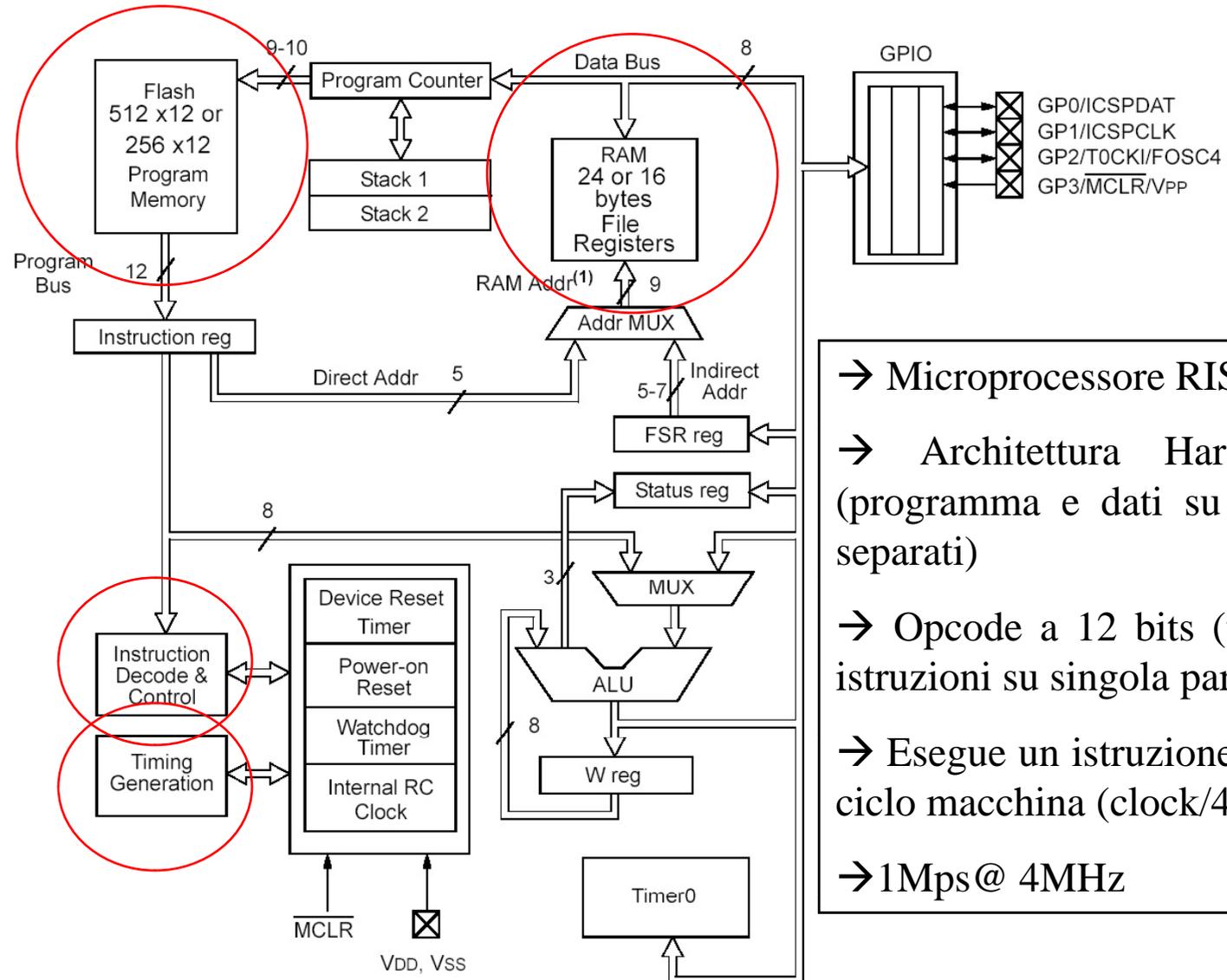
- I microprocessori così come i microcontrollori possono essere suddivisi in due grandi categorie: CISC e RISC.
 - CISC: Complex Instruction Set Computers, dotati di un repertorio molto ampio e molto variato di istruzioni (oltre 300)
 - RISC: Reduced Instruction Set Computers, dotati di un set ristretto di istruzioni

Architettura di un semplice microcontrollore - PIC10F200 (Microchip)



Architettura di un semplice microcontrollore - PIC10F200

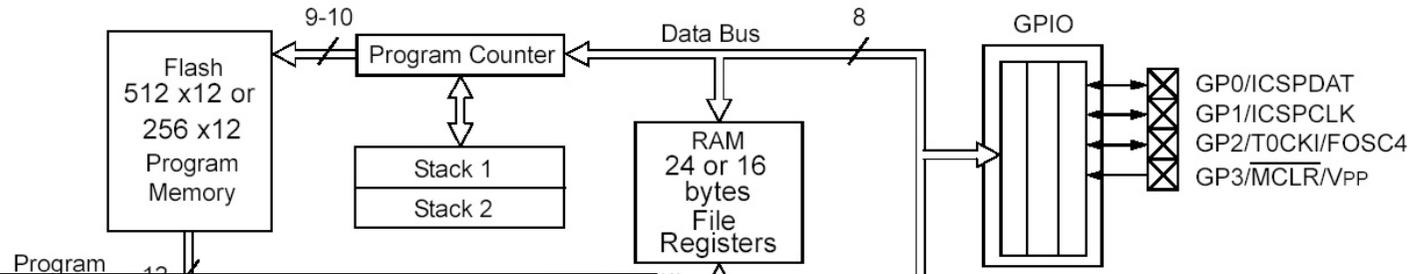
CPU



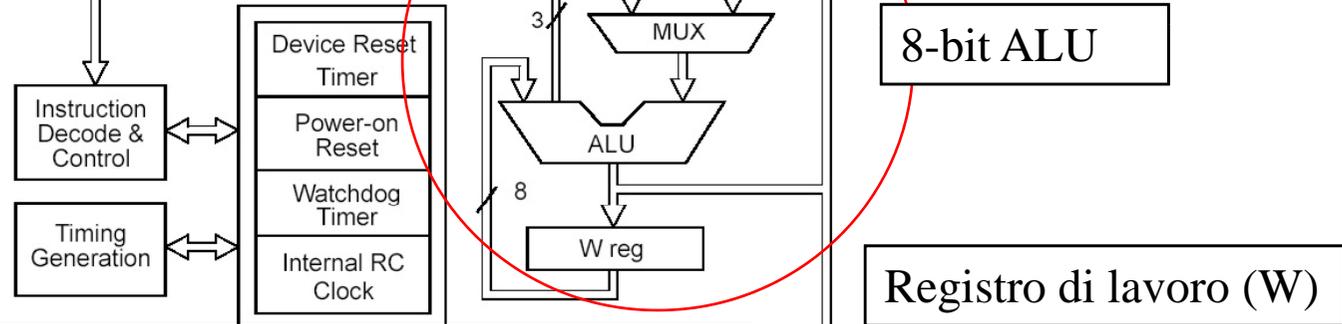
- Microprocessore RISC
- Architettura Harvard (programma e dati su bus separati)
- Opcode a 12 bits (tutte istruzioni su singola parola)
- Esegue un'istruzione per ciclo macchina (clock/4)
- 1Mps @ 4MHz

Architettura di un semplice microcontrollore - PIC10F200

CPU



La ALU può operare su tutti i registri e W. Se l'operazione richiede 2 operandi uno è sempre W
→ Il risultato dell'operazione va sempre in W



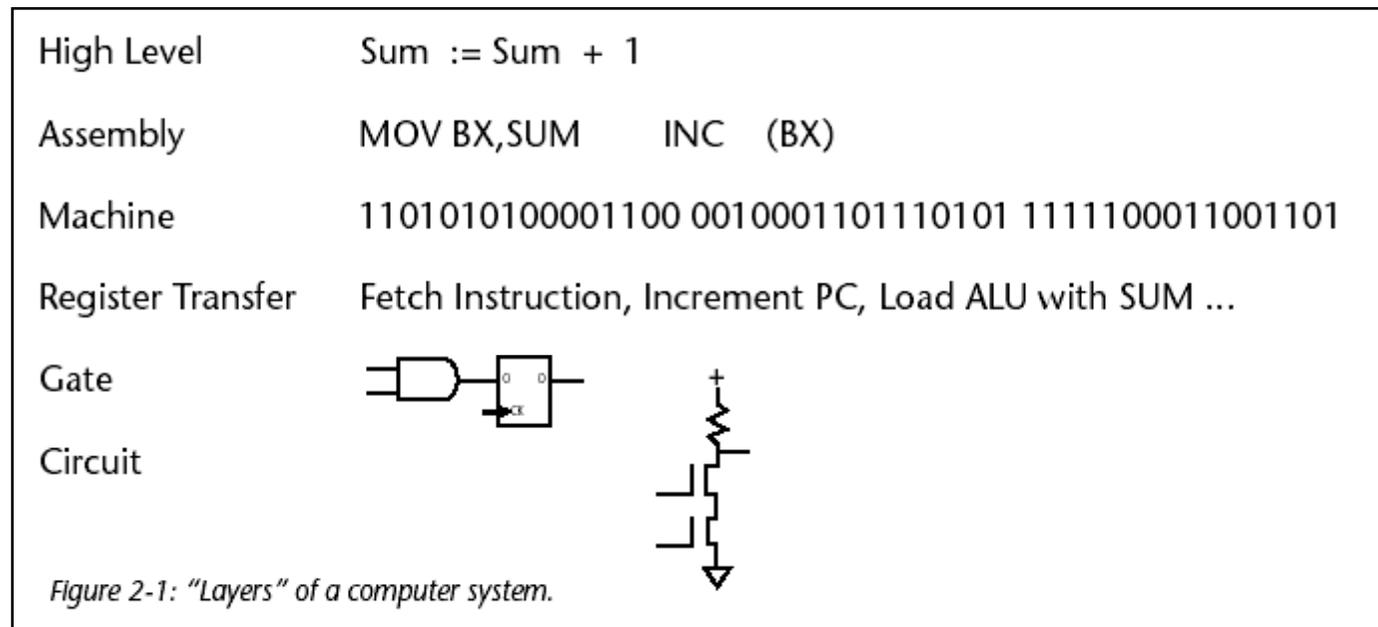
Operazioni supportate: +, -, AND, OR, XOR, NOT, shift.
Nello "Status register" si trovano i bit di Carry (C,) e Zero (Z) che vengono influenzati dalle operazioni della ALU

The Development Language

Linguaggio di alto livello vs. Assembler

- Permette applicazioni più complesse
- Riusabilità
- Portabilità
- Richiede maggiori risorse

- Velocità
- Miglior controllo
- Strumenti di sviluppo gratuiti
- Specifico per il processore



La scelta del microcontroller

Solitamente più scelte diverse sono possibili: nello specifico vanno tenuti in considerazione i seguenti parametri sulla base dell'applicazione finale.

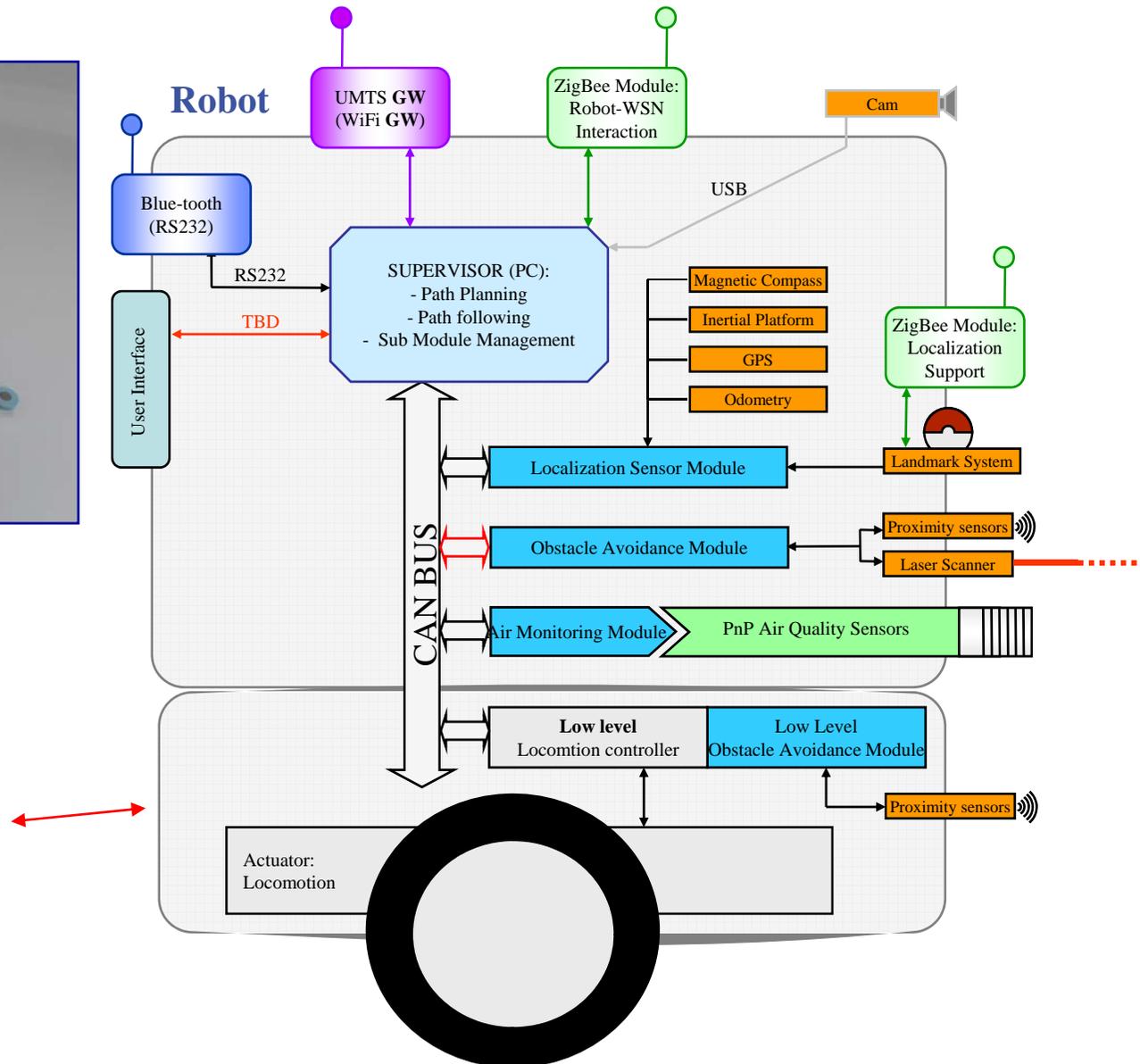
- ✓ Numero dei pin di I/O
- ✓ Interfacce
- ✓ Memoria RAM (quantità e tipo)
- ✓ Numero e tipo di interrupts
- ✓ Velocità del processore
- ✓ Consumi energetici
- ✓ Memoria di programma (quantità e tipo)
- ✓ Ambiente di sviluppo

Microprocessori in robotica

- ✓ Controllo dei motori/attuatori (basso livello)
- ✓ Controllo movimento (alto livello)
- ✓ Power management (carica/scarica batterie)
- ✓ Acquisizione/elaborazione dei segnali dei sensori (prossimità, encoders, odometria, contatto, visione)
- ✓ Interfacce

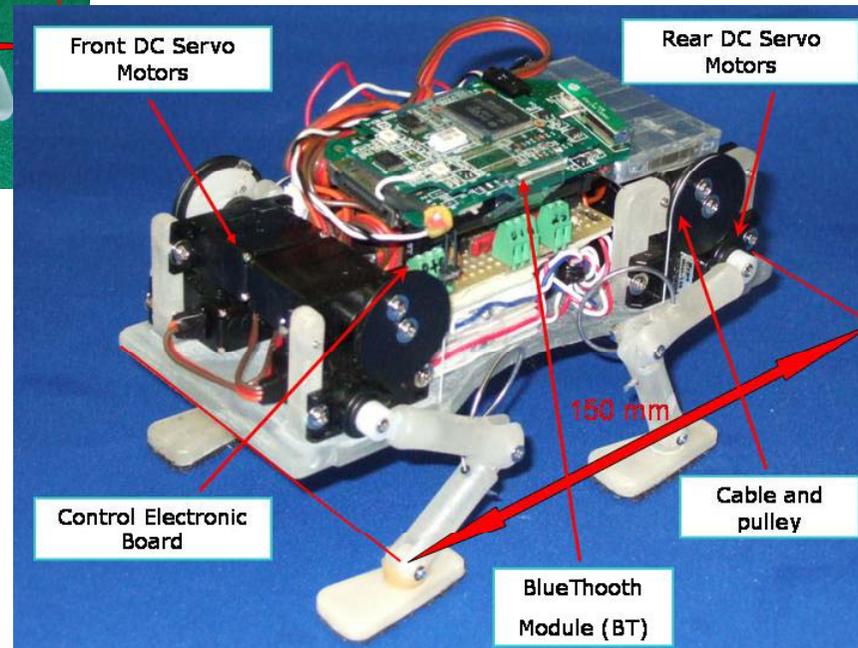
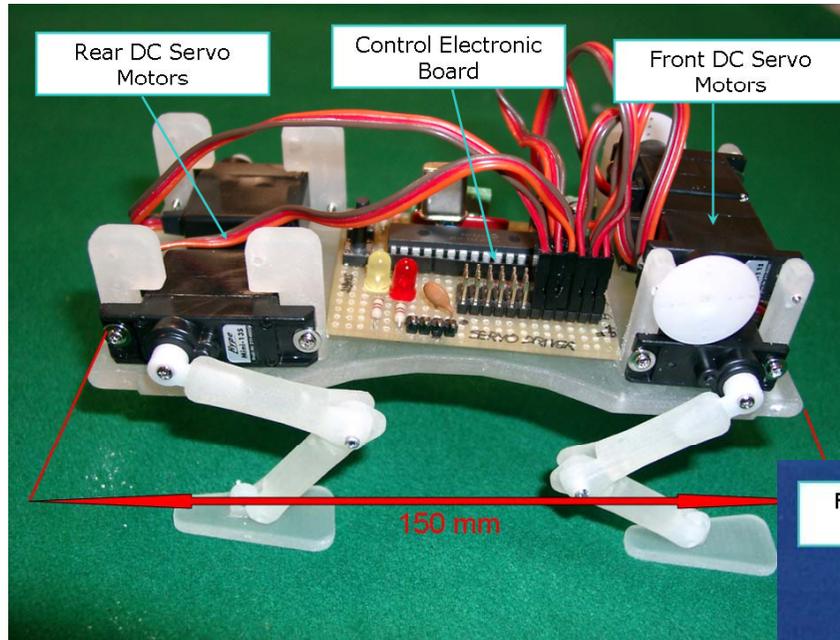
Microprocessori in robotica

Architettura di un sistema robotico - Esempio 1: DustBot



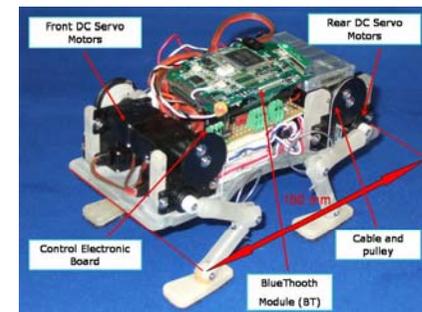
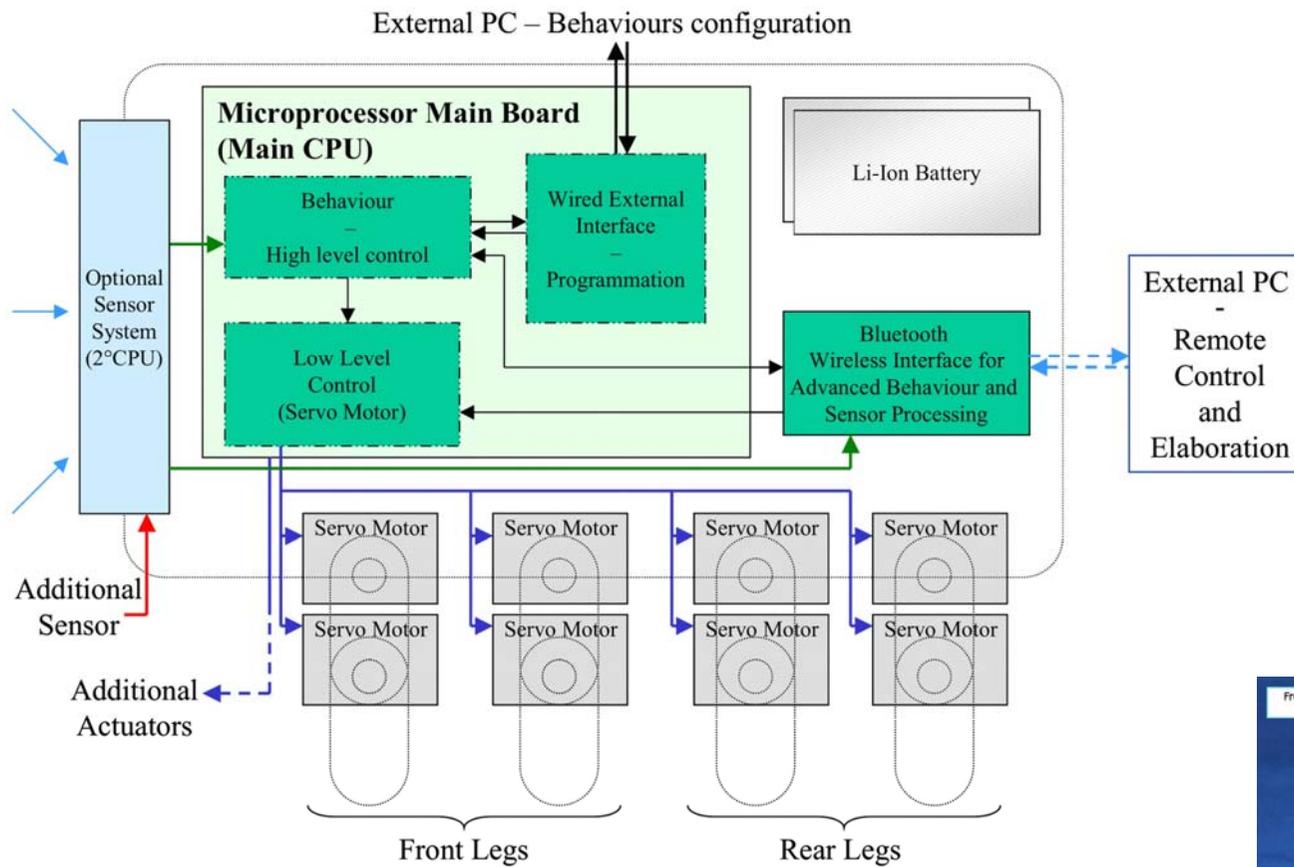
Microprocessori in robotica

Architettura di un sistema robotico - Esempio 2: Robot biomimetico a 4 zampe



Microprocessori in robotica

Architettura di un sistema robotico - Esempio 2: Ropbot biomimetico a 4 zampe



Microprocessori in robotica

Architettura di un sistema robotico - Esempio 2: Ropbot biomimetico a 4 zampe

