Elettronica Biomedica II

Analog and digital signal processing

Processare un segnale, dal punto di vista analogico, significa trattarlo con componenti appunto analogiche. Si tratta di un processamento:

- A basso costo (amplificatori e filtri sono facilmente reperibili)
- Semplice ed immediato (non è necessaria una conversione)
- Dato che non si ha il problema della frequenza di campionamento, possono essere utilizzate anche bande elevate
- La risoluzione, sia in ampiezza che in frequenza sono infinite (per cui non si ha neanche il problema dell'aliasing)

Tuttavia, un segnale trattato in modo analogico:

- Non è riproducibile, poiché le resistenze sono sempre caratterizzate da una certa tolleranza
- Non è ripetibile
- Nel caso in cui si vogliano effettuare delle modifiche è necessario riprogettare tutta la scheda
- I componenti sono sensibili alle condizioni ambientali e soggetti ad invecchiamento

Per quanto riguarda invece il processing digitale, esso mostra numerosi vantaggi:

- Programmabilità (facilmente modificabile senza dover reingegnerizzare la scheda)
- Performance più elevate (costi, tempi di sviluppo, minore dissipazione di potenza)
- Stabilità (le performance non variano nel tempo o in funzione di condizioni ambientali)
- Ripetibilità
- Segnale immagazzinabile e trasmissibile

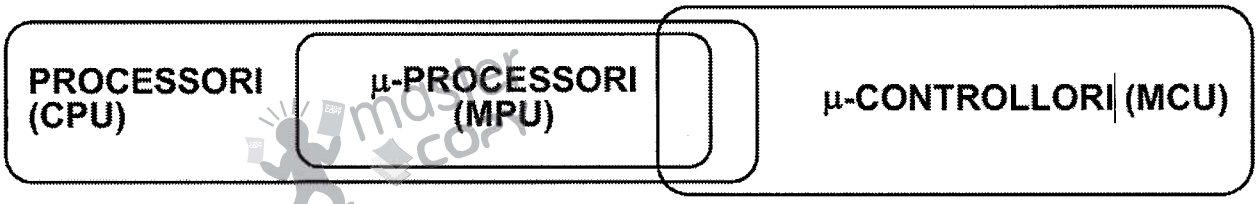
Tuttavia presenta anche alcuni svantaggi:

- Rumore di quantizzazione
- Frequenza di campionamento finita, con conseguenti problemi di aliasing
- Banda limitata da vincoli real-time e dalla velocità dei convertitori ADC e DAC

Microcontrollore

Si **definisce processore**, un dispositivo hardware dedicato all'esecuzione di istruzioni.

Si definisce **micro-processore**, un tipo di processore in cui la struttura hardware è interamente contenuta in un circuito integrato. Un **microcontrollore** contiene un microprocessore con un set di istruzioni ridotto e alcune periferiche dedicate.



Quindi, a differenza del processore, oltre alla CPU sono presenti anche altri componenti.

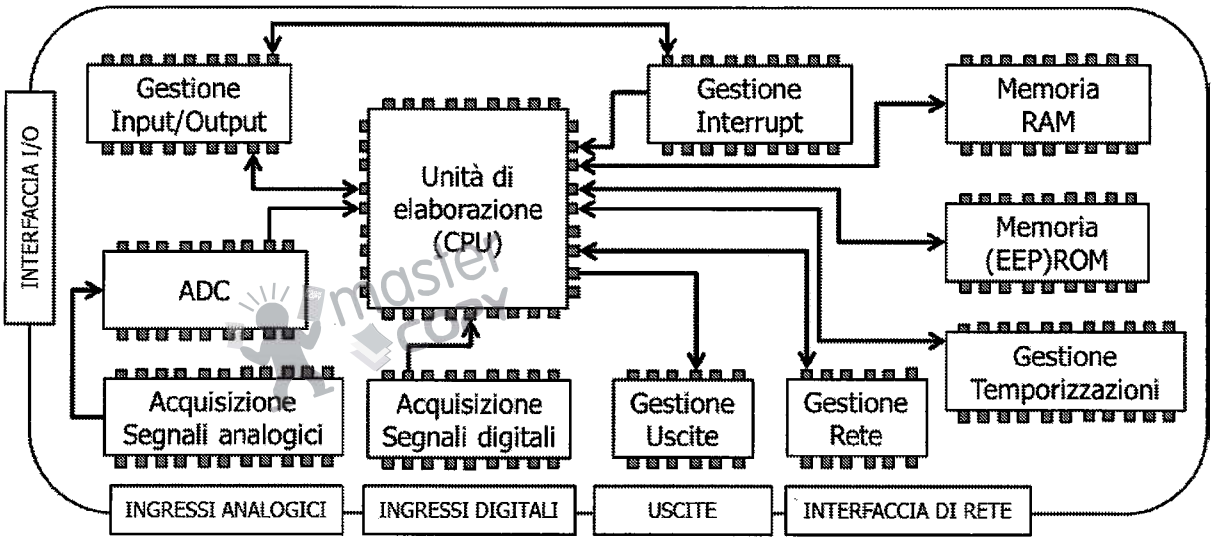
Le differenze principali tra un microprocessore e un microcontrollore, stanno nel fatto che:

- Un microprocessore è costituito da un unico chip che però ha bisogno di numerose integrazioni esterne (memoria, clock, periferiche di ingresso e di uscita) (esempio: CPU dentro il pc e tutte le periferiche, come stampanti, mouse, fuori)
- Un microcontrollore riunisce tutti gli elementi all'interno di un piccolo contenitore (memoria ROM, RAM, clock, circuito di reset e periferiche), si tratta di fatto di un minicomputer. Questo d'altra parte porta ad avere capacità di calcolo molto minori e quindi vengono utilizzati solo per un piccolo insieme di funzioni specifiche.

Il microcontrollore viene utilizzato in sistemi così detti ***embedded*** ovvero integrati, per applicazioni di controllo digitale. Interagisce direttamente con l'esterno attraverso un programma residente nella propria memoria interna, ed è disponibile da 8, 16 o 32 bit.

A differenza dei microprocessori, che sfruttano una memoria volatile o di massa, un microcontrollore esegue un programma applicativo detto **firmware** che di solito è memorizzato su una memoria ROM e non è riprogrammabile.

Struttura di un microcontrollore



Interrupt: è un segnale di hardware (come un tasto premuto) che gestito a livello software, avvisa che c'è stato un evento esterno. Ad esempio, il MC svolge una certa funzione e in quel momento viene premuto un tasto sulla tastiera: a seconda del livello di priorità prevarrà un evento oppure un altro.

Essi hanno moltissimi campi di applicazione:

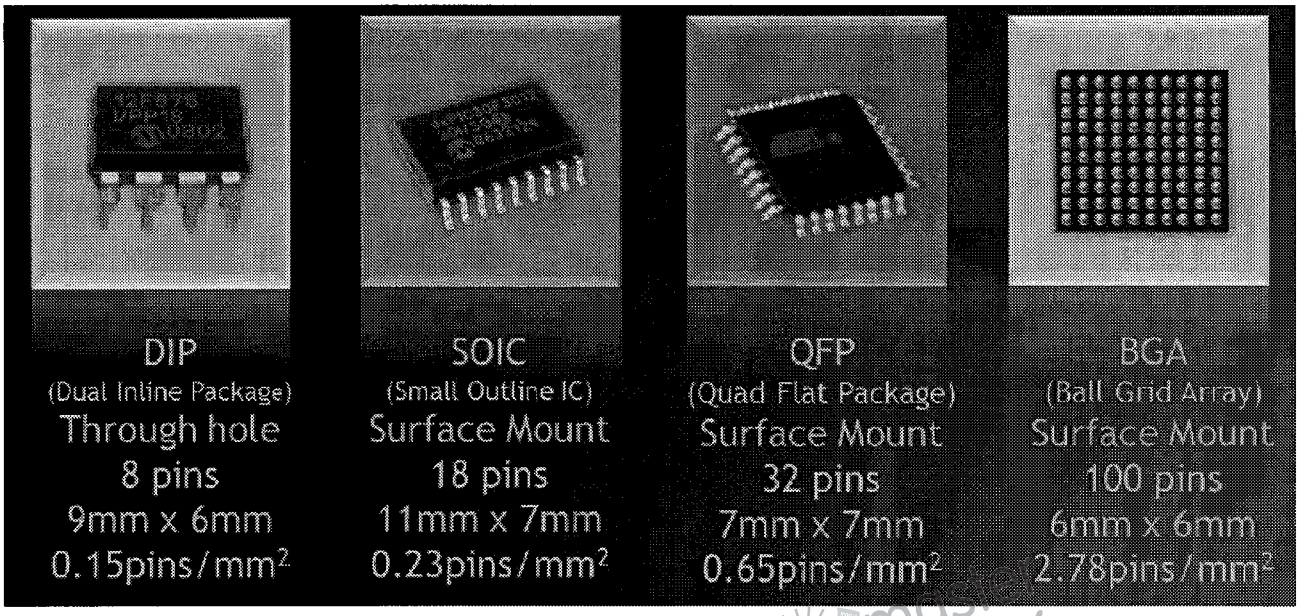
- Convertitori A/D e D/A
- PWM
- Timer, contatori
- Pin di I/O

Inoltre sono adatti alla realizzazione di applicazioni di controllo e coordinamento, e di acquisizione ed elaborazione del segnale.

In generale, un microcontrollore deve rispondere ai seguenti requisiti:

Requisito	Caratteristica	Beneficio
Gestione ingressi / uscite	Porte di I/O con controllo fino al singolo bit	Controllo efficiente di dispositivi esterni quali **attuatori**, teleruttori etc.
Comunicazione Periferiche	Porta seriale, SPI, I²C, UART, CAN, etc.	Estensione delle funzionalità con l'uso di **periferiche esterne**
Controllo di motori ed attuatori	Timer, Contatori, PWM	Facilità di **programmazione**
Gestione di programmi logico sequenziali	Salti condizionati, istruzioni logiche, etc.	Facilità di **realizzazione** del software **logico-sequenziale**
Reazione ad eventi	Gestione degli IRQ prioritari	Facilità nella **realizzazione** di sistemi **Real Time**
Acquisizione dati da sensori	ADC	Facilità di **istallazione** in ambienti **pre-esistenti**

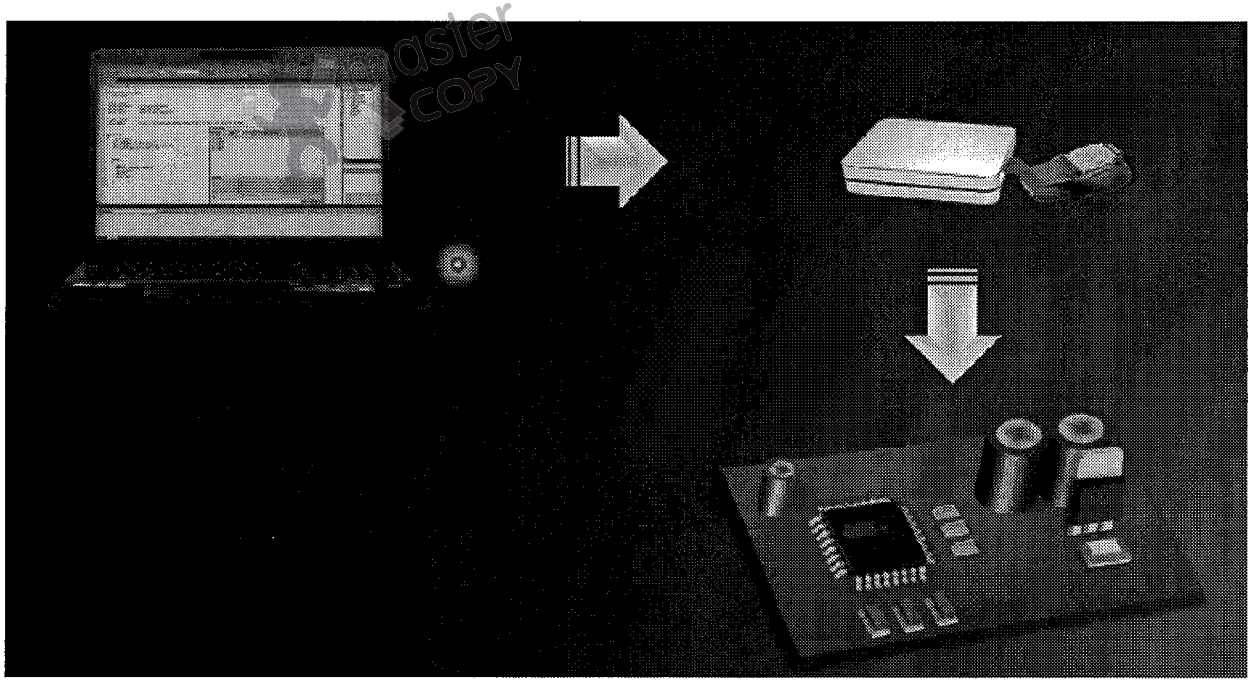
Esistono vari tipi di packaging: DIP,SOIC,QFP,BGA.



In generale, esistono due tipi di montaggio: pin che entra nel foro dove viene saldato e però risulta ingombrante, oppure si hanno dei pin piegati sulla punta, con una notevole riduzione dell'ingombro. Avere più pin significa avere più I/O e quindi più periferiche.

Programmare il microcontrollore

Per programmare un microcontrollore, esso dovrà essere inserito in una scheda che è interfacciata con un computer dove utilizzo un software SDK, ovvero un kit di sviluppo con un linguaggio di programmazione. Il programma verrà scritto nella memoria flash del microcontrollore. Il programma dovrà essere poi simulato per verificare che funzioni correttamente, e infine verrà inserito nella scheda.



Microcontrollore vs DSP

Un microcontrollore generico può essere molto performante nell'elaborazione di fogli elettronici, tuttavia può avere difficoltà nell'elaborare segnali audio e video. Per queste applicazioni si ricorre di solito al DSP (digital signal processor), caratterizzato da una diversa architettura che consente di ottenere capacità di calcolo molto più elevate. Ad esempio, per fare la moltiplicazione tra due numeri a 4 bit il microcontrollore generico impiega 5 cicli di clock, mentre il DSP ricorrendo ad una moltiplicazione hardware che sfrutta una somma di moltiplicazioni (moltiplicazione MAC), impiega un solo ciclo di clock.

master COPY