



DEASCUOLA

INSEGNARE SISTEMI E RETI

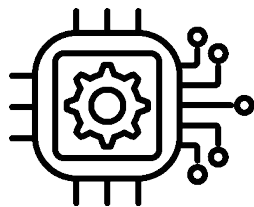
Insegnare con... Gateway

Terza edizione

La via di accesso al mondo delle reti

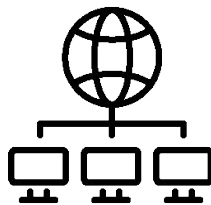
Relatori:
Paolo Macchi, Susanna Anelli





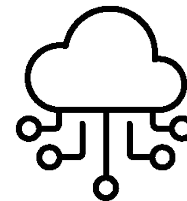
VOL 1

- Sistemi di elaborazione
- Physical Computing
- Modello TCP/IP



VOL 2

- LAN e WAN
- Protocolli di rete
- Progettazione di reti



VOL 3

- Livello applicazione
- Cybersecurity
- Cloud e IoT



COLLABORAZIONE

L'insegnamento di Sistemi e Reti nasce da un'esperienza didattica

ANALOGIE e «la Parola a...»

Utilizzo delle analogie nell'insegnamento dei sistemi informatici



*Il metodo è più importante
del contenuto*

PERCORSO

Percorso didattico tracciato dal flusso dei dati dell'IoT

IMMAGINI e LABORATORIO

Immagini e attività di laboratorio per favorire l'apprendimento



Collaborazione



Questo libro non è il risultato di un lavoro solitario svolto a tavolino, ma è piuttosto il frutto di un processo collaborativo tra studenti e docenti. È nato dall'esperienza condivisa, dallo scambio di idee e dalla volontà comune di creare qualcosa di significativo. Questa collaborazione ha portato ad alcune scelte:

- utilizzo di un linguaggio adeguato
- utilizzo di esempi e immagini
- apprendimento non formale
- condivisione di esperienze



Analogie: partiamo dalla realtà

Tema 1 - Architettura dell'elaboratore

1 Introduzione

Un computer deve poter interagire con il mondo esterno.

La comunicazione con le grandezze fisiche dell'ambiente e con gli esseri umani è realizzata attraverso le **interfacce di ingresso/uscita**.

Si tratta di circuiti hardware che hanno la funzione di collegare il data bus del computer con le periferiche.

Ciò è necessario per colmare la distanza tra il mondo analogico, rappresentato da grandezze fisiche continue, e il mondo digitale basato su grandezze discrete.

I computer hanno leggi proprie, diverse da quelle del mondo ordinario. Al loro interno i dati marcano su un bus come soldatini ben allineati. Il ritmo di lavoro è imposto da un orologio che batte il tempo a una frequenza altissima e i valori di tensione e corrente possono essere molto diversi da quelli utilizzati nella vita quotidiana.

Comprendi con l'analogia

I tempi della realtà esterna rispetto a quelli di un computer sono come quelli di un bradipo rispetto ai nostri.

I bradipi sono animali famosi per la loro lentezza. Si muovono a una velocità di qualche centimetro al secondo e possono trascorrere l'intera vita su un albero.

Per quanto riguarda i tempi di risposta, un computer avrebbe nei nostri confronti la stessa sensazione che proviamo noi verso i bradipi: nel tempo in cui un umano preme un tasto della tastiera, il computer vive una vita!



2 Le periferiche

Una Ferrari è un'auto meravigliosa. Il rombo del suo motore è musica per le orecchie di chi lo apprezza. Eppure senza il volante e il cambio sarebbe solo un bel monumento all'ingegno umano.

La stessa cosa vale per uno smartphone, un PC o un supercalcolatore: senza le periferiche non sapremmo cosa farcene.

Le **periferiche** sono i dispositivi che permettono lo scambio di informazioni tra il computer e il mondo esterno.

Hanno tre funzioni principali (Tabella 1):

- permettono di comunicare con l'utente (monitor, tastiera ecc.);
- permettono di interagire con il mondo fisico (sensori, attuatori ecc.);
- permettono di immagazzinare le informazioni su memorie permanenti (dischi magnetici e a stato solido, nastri ecc.).

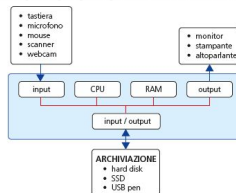
Tab. 1 Classificazione delle periferiche più comuni

Funzione	Esempi di periferiche
Colloquio con l'utente	• monitor touchscreen, tastiera, mouse, tavoletta grafica • stampanti inkjet e laser • videocamere e fotocamere • casse audio, cuffie • gamepad, joystick, sensori di movimento
Interazione con il mondo fisico	• apparati per la trasmissione dei dati in rete (modem) • sistemi elettromeccanici (relè, motori ecc.) • convertitori analogico-digitale e convertitori digitale-analogico
Memorizzazione permanente di informazioni	• dischi a stato solido, dischi magnetici (hard disk) • dischi ottici (CD, DVD) • nastri magnetici

Le periferiche di I/O - Unità 3

Dal punto vista del trasferimento dei dati, le periferiche si possono classificare come dispositivi di **input**, in cui le informazioni transitano dall'esterno verso il data bus del computer, e dispositivi di **output**, in cui le informazioni transitano dal data bus del computer verso l'esterno (Figura 1).

Le prime comprendono, per esempio, tastiera, mouse, touchscreen, microfono e sensori, le seconde monitor, stampanti, modem, sistemi audio, attuatori. Alcune periferiche, come i dischi o le schede di rete per la trasmissione dei dati, sono sistemi **bidirezionali** (input/output), perché possono sia inviare sia ricevere informazioni.



Pit Stop

3 Che cosa significa che una periferica è bidirezionale?

Fig. 1 Le periferiche di ingresso e uscita più comuni. Le periferiche per l'immagazzinamento delle informazioni sono bidirezionali (input/output), perché i loro dati possono essere sia scritti sia letti.

Le periferiche possono essere:

- **esterne** all'elaboratore, cioè collegate al sistema tramite cavi che si inseriscono nelle porte dell'interfaccia. Una porta connette elettricamente e meccanicamente una periferica con l'interfaccia;
- **interne** all'elaboratore, cioè collegate al sistema tramite bus interni al case del computer.

3 Le interfacce

Il punto di incontro tra le periferiche e il bus del sistema è costituito dalle **interfacce**, sistemi hardware e software, che utilizzano metodi standard per l'indirizzamento, il controllo, la sincronizzazione e lo scambio di dati con le periferiche. La Figura 2 mostra la struttura generale di una interfaccia, che comprende:

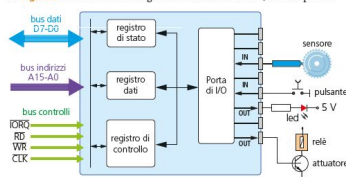


Fig. 2 Architettura di una generica interfaccia di ingresso/uscita. La parte di sinistra comprende il buffer di dati, il registro di stato e il registro di controllo necessario per la programmazione del dispositivo.

Analogie: partiamo dalla realtà

Tema 3 - Architettura di rete: l'accesso alla rete

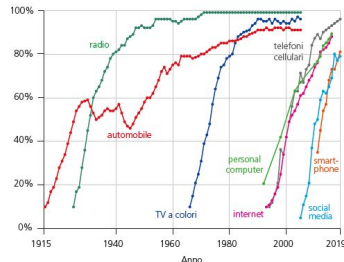
1 Introduzione

La convergenza dei sistemi di elaborazione e delle reti di telecomunicazione ha reso sempre più facile la comunicazione tra persone e cose. La conseguenza è che in ogni istante circolano in rete miliardi di terabyte di dati, con profonde conseguenze sociali.

A titolo di esempio, la **Figura 1** mostra la diffusione delle principali tecnologie negli ultimi decenni, paragonata a quella delle tecnologie meno recenti. I fenomeni legati alla comunicazione stanno diventando sempre più complessi, coinvolgendo molti aspetti. Per questo motivo negli ultimi anni i team di ricerca sono formati da persone esperte in diverse discipline quali, per esempio, la matematica, la fisica, l'informatica, la statistica, l'architettura, la filosofia, la medicina, la psicologia, l'economia.

In questa sede ci limiteremo ad approfondire gli aspetti tecnici legati alle reti informatiche e alle loro applicazioni, tenendo in considerazione le leggi scientifiche che regolano i sistemi complessi naturali e artificiali, di cui le reti, come per esempio Internet, sono parte.

→ Fig. 1 Quota di famiglie statunitensi che utilizzano specifiche tecnologie, dal 1915 al 2019. (Tratto con modifiche da Esteban Ortiz-Ospina, 2019, «The rise of social media». Pubblicato su OurWorldinData.org con licenza CC BY.)



2 I sistemi complessi

Un **sistema complesso** è un sistema composto da un numero molto grande di componenti; le interazioni e le relazioni di queste componenti definiscono il comportamento globale del sistema.

Per molto tempo si è pensato che sarebbe bastato studiare le singole parti di un sistema per comprenderne il funzionamento complessivo, ma non è così.

In un sistema complesso le leggi che regolano le relazioni tra i componenti (ad esempio la struttura ad archi del ponte in **Figura 2** a pagina seguente) sono più importanti dei componenti stessi (le pietre).

Comprendi con l'analogia

Un sistema complesso è come un ponte, che esiste sia grazie alle singole pietre sia grazie agli archi che le connettono.

«Marco Polo descrive un ponte, pietra per pietra.

«Ma qual è la pietra che sostiene il ponte?» - chiede Kublai Kan.

«Il ponte non è sostenuto da questa o quella pietra, - risponde Marco - ma dalla linea dell'arco che esse formano.

Kublai Kan rimane silenzioso, riflettendo.

Poi soggiunge: - Perché mi parli delle pietre? È solo dell'arco che m'importa.

Polo risponde: - Senza pietre non c'è arco.»

Italo Calvino, «Le città invisibili», Mondadori, 1972

In questo colloquio tra Marco Polo e Kublai Kan, immaginato dallo scrittore Italo Calvino nel libro *Le città invisibili*, Marco Polo evidenzia a Kublai Kan che un ponte non è sostenuto dalle singole pietre di cui è fatto, ma dalla struttura formata dalle pietre stesse. Chi tenta di spiegare come possa esistere un ponte tenendo conto solo delle singole componenti (le pietre) non vi riesce: assume una prospettiva che viene chiamata riduzionismo. Ma una volta sottolineato il fatto che i sistemi complessi come i ponti esistono grazie agli archi, Marco Polo pone l'accento anche sul fatto complementare: senza le pietre non vi sarebbero gli archi che le connettono.

Le parti che costituiscono un sistema complesso, prese singolarmente, non riescono a spiegare il funzionamento globale. È la loro interazione che fa emergere un comportamento collettivo, che si presenta spontaneamente e non dipende da un controllo centrale.

Sistemi molto diversi tra loro, come lo stato del ghiaccio che fonde, il volo di uno stormo di uccelli, una rete sociale, una rete aeroportuale o una rete di computer, possono mostrare comportamenti simili, regolati dalle stesse leggi matematiche. Per comprendere i sistemi complessi, la cosa importante è scoprire le relazioni che si creano tra i componenti.

La parola a... Giorgio Parisi

Giorgio Parisi, premio Nobel per la fisica 2021, ha avuto il merito di portare all'attenzione del grande pubblico i «sistemi complessi». A partire dallo studio del volo degli stormi, Parisi ha studiato i «comportamenti collettivi complessi» che coinvolgono atomi e molecole, ma anche animali, persone, leggi economiche e siti web.

Lasciamo a lui la parola...

«Il volo degli stormi mi affascinava in maniera particolare perché [permetteva] di capire il comportamento di un sistema composto da un gran numero di componenti (attori) interagenti. Nella fisica, a seconda dei casi, gli attori possono essere elettroni, atomi, spini, molecole; hanno delle regole di comportamento molto semplici, ma presi tutti insieme danno luogo a un comportamento collettivo molto più complesso. [...] Le interazioni si possono schematizzare con regole semplici, ma i risultati dell'azione collettiva possono essere davvero inaspettati.»

Giorgio Parisi, «In un volo di stormi», Rizzoli, 2021

Le reti - Unità 10



Fig. 2 Quando osserviamo le pietre di un ponte non riusciamo a comprendere come esso possa reggersi. Per capire, dobbiamo osservare la struttura delle arcate. In generale, è la visione complessiva dell'opera che ci aiuta a capire le sue proprietà.

Pit Stop

- 1 Che cos'è un sistema complesso?
- 2 Perché è fondamentale l'interazione tra le parti di un sistema?

Analogie: partiamo dalla realtà

Tema 1 - Architettura dell'elaboratore

Pit Stop

11 Che cosa sono Meltdown e Spectre?



Fig. 7 Un processore Multi-Core Intel.

Pit Stop

12 Qual è il vantaggio dei processori Multi-Core?

In particolare, Meltdown è una vulnerabilità hardware legata alla progettazione di molte moderne CPU Intel x86 e ARM (Spectre è una vulnerabilità simile che affligge microprocessori più recenti). Meltdown consente a un processo di «scivolare» i confini della sicurezza imposti dall'hardware aggirando i normali controlli e accedendo allo spazio di memoria protetto, tramite un'analisi della cache dei dati durante l'elaborazione delle istruzioni nella pipeline e l'utilizzo della branch prediction. In questo modo un malintenzionato può ottenere dati riservati presenti nella memoria cache dei programmi in esecuzione: password, messaggi, foto, e-mail e documenti personali. Nel 2018 degli attacchi che sfruttavano questa vulnerabilità hanno colpito molti server e dispositivi intelligenti basati su ARM, rallentandone il funzionamento.

Multi-Core

L'idea di disporre di un unico chip contenente più processori (Multi-Core, Figure 7) è nata dalla necessità di raggiungere prestazioni sempre più elevate, cosa che l'aumento della frequenza di clock o del numero di bit, da 32 a 64 a 128, non era più in grado di assicurare. Infatti l'aumento della frequenza del clock, e il conseguente aumento di velocità nelle operazioni, porta con sé un crescente ed eccessivo consumo di potenza elettrica, con relativo calore generato e difficilmente dissipabile, soprattutto nei sistemi mobili di dimensioni ridotte.

I primi processori composti da **doppio core (Dual-Core)**, realizzati da IBM, risalgono al 2003. La tecnologia si è rapidamente diffusa quando, nel 2005, Intel e AMD (tra i principali produttori di CPU) hanno presentato i loro primi modelli. I processori Multi-Core di ultima generazione possono contenere decine di core e lavorare a frequenze che superano i 5 GHz, offrendo prestazioni elevate sia per creazioni lavorative che per esperienze di gioco immersive. Il numero di core installati è aumentato negli anni fino a superare i 70, un numero che, tuttavia, non è ancora sfruttato appieno dai sistemi software.

Due diverse filosofie di progettazione delle architetture: CISC e RISC

CISC e RISC sono modi di progettare l'architettura di un processore basati su filosofie diverse nel modo di intendere e gestire le istruzioni che possono essere eseguite da quei processore.

- Le architetture di tipo **CISC (Complex Instruction Set Computing)** sono caratterizzate da un insieme numero di istruzioni complesse che possono avvalersi di numerosi metodi di indirizzamento. La complessità delle istruzioni si riflette sulla complessità dell'hardware del processore e, di conseguenza, sui tempi necessari per l'esecuzione delle istruzioni stesse.

Un'altra caratteristica importante di queste architetture è che la dimensione delle istruzioni è variabile in funzione del metodo di indirizzamento utilizzato. Questo ha un impatto sulla fase di fetch, che risulta essere più complessa, poiché non è noto a priori il numero di byte da prelevare.

La complessità penalizza le prestazioni, ma aumenta la scelta tra le istruzioni, diminuendo il divario esistente tra il linguaggio ad alto livello utilizzato dai programmatori e il linguaggio macchina.

Evoluzione dei microprocessori - Unità 5

- Le architetture di tipo **RISC (Reduced Instruction Set Computing)** sono caratterizzate da poche istruzioni, semplici e di lunghezza fissa. Operazioni complesse devono essere scomposte in operazioni elementari e gli operandi possono essere reperiti solo dai registri e non dalla memoria. Esistono delle operazioni specifiche per il trasferimento dati da/a memoria e registri (load e store). L'hardware è semplificato, così come sono ridotti i tempi di esecuzione delle istruzioni, ma le istruzioni sono meno potenti. La dimensione fissa certamente semplifica la fase di fetch e favorisce la realizzazione del pipeline. Obiettivo di queste architetture è quello di avere a disposizione il minimo numero di istruzioni necessarie per risolvere qualsiasi problema, allo scopo di semplificare l'hardware e la gestione delle istruzioni.

Comprendi con l'analogia

Le architetture CISC e RISC possono essere comprese con i mattoncini delle costruzioni.

Utilizzare un'architettura RISC è un po' come avere a disposizione solo i mattoncini fondamentali delle costruzioni.

L'architettura CISC, invece, permette di avere anche pezzi di forme strane e diverse, le quali possono essere comunque ottenute combinando tra loro i pezzi base.



Architetture alternative nel gestire con la memoria dati e istruzioni

Sebbene negli anni siano state introdotte architetture diverse, come quelle del *transputer* per il calcolo parallelo, le architetture che hanno avuto il maggior successo sono quelle di von Neumann, ancora oggi la più usata, e di Harvard. La caratteristica dell'**architettura di von Neumann** è di avere un'unica memoria per dati e istruzioni e questo fatto potrebbe causare rallentamenti del sistema nelle operazioni sequenziali.

Nell'**architettura di Harvard** originale (Figure 8), che precede storicamente quella di von Neumann, i dati e le istruzioni risiedono in aree di memoria diverse. Questo fatto permette di leggere contemporaneamente un'istruzione e un dato parallelizzando le operazioni. La maggior velocità di esecuzione è a vantaggio di una elevata complessità.

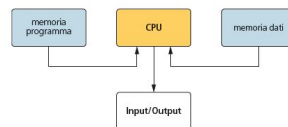


Fig. 8 L'originale architettura di Harvard. Successivamente l'architettura di Harvard è stata ripensata con un'area di memoria che contiene istruzioni e dati, e un'altra area di memoria che contiene solo dati.



Analogie: partiamo dalla realtà



L'analogia aiuta a :

- spiegare problemi complessi, partendo dall'esperienza quotidiana di studenti e studentesse
- comprendere le ragioni per cui un problema è stato posto («perché mi pongo questo problema? Perché devo studiare questa cosa?»)
- comprendere a livello intuitivo il problema

Nel libro:

- per esempio, nel volume 1 sono presenti **36 analogie** distribuite su tutte le unità

Comprendi con l'analogia

Le architetture CISC e RISC possono essere comprese con i mattoncini delle costruzioni

Utilizzare un'architettura RISC è un po' come avere a disposizione solo i mattoncini fondamentali delle costruzioni.

L'architettura CISC, invece, permette di avere anche pezzi di forme strane e diverse, le quali possono essere comunque ottenute combinando tra loro i pezzi base.



La parola a...

Tema 1 - Architettura dell'elaboratore

La Tabella 1 confronta i linguaggi ad alto e a basso livello, evidenziando le loro caratteristiche.

→ Tab. 1

Linguaggi ad alto livello	Linguaggi a basso livello (macchina/Assembly)
«Astraggono» dall'architettura dell'elaboratore; - offrono una scelta tra molti linguaggi; - la loro esecuzione è più lenta; - hanno istruzioni potenti, semplici e generali; - sono facili da imparare e di pronto uso, con molte librerie di programmi disponibili; - sono utilizzati per la realizzazione delle applicazioni software più comuni.	- Dipendono dall'hardware e dall'architettura; - la loro esecuzione è più veloce; - c'è una corrispondenza 1:1 tra istruzioni in Assembly e istruzioni in linguaggio macchina; - richiedono tempi lunghi e uno sforzo maggiore per l'apprendimento e per la realizzazione dei programmi nel linguaggio specifico del processore.

La parola a... Ada Byron Lovelace, la madre di tutti i programmatori

Ada Byron Lovelace (1815-1852) può essere considerata la «madre di tutti i programmatori». Fu lei che cento anni prima che il calcolatore fosse inventato ebbe l'intuizione che «rendere un meccanismo capace di combinare simboli generali, in sequenze illimitate per varietà ed estensione, vuol dire avere trovato un punto di unione fra le operazioni della materia e i processi astratti della mente». Ada prefigura una macchina che può rappresentare, oltre ai numeri, qualsiasi elemento simbolico della realtà, come testi e musica. Non solo: Ada immagina la capacità di apprendere delle macchine, anticipando le soluzioni di Intelligenza Artificiale e mostrandone i limiti: «la macchina analitica non ha alcuna pretesa di concepire alcunché», ma solo di eseguire i compiti che noi le assegniamo.



Comprendi con l'analogia

I codici di un computer sono come quelli della natura

Dal diario di Giulia Mercuri che realizza applicazioni web.

Stasera, a tavola, mia figlia di terza superiore mi chiede: «Oggi a scuola parlavano di DNA, il codice della vita. C'entra qualcosa con i linguaggi dei computer?» La domanda mi sorprende e non so rispondere. «Mangia che si raffredda!» taglio corto e cambio discorso, anche se la cosa mi incuriosisce. Dopo cena provo a fare una ricerca in rete e... scopro delle somiglianze stupefacenti tra il DNA e il codice binario, a me più familiare.

Nella memoria del computer gli «0» e gli «1» sono memorizzati in forma di cariche elettriche, mentre in natura le informazioni sono immagazzinate come catene del DNA.

L'informazione, in un computer, è codificata con i simboli binari, mentre l'informazione biologica è trasmessa da una successione di 4 simboli (figura 3).

Il computer possiede un disco a stato solido. Il DNA è come un supporto di memoria ridondante e affidabile. Per elaborare un programma, il computer deve trasferirlo dal disco alla RAM, mentre il DNA si serve dell'RNA messaggero.



Il software (con DigiCine) - Unità 4

GAGAAVAAAC	VAGVAVVCFV	CVGGVCCCA	CAGACVAGA	GAGAACCCG	50
CACCAVGVFC	GVGGVCCVGG	VGCGVCCG	VGCGVGVCC	AGCCAGGVGG	100
VGACGVVAC	CACCGAGACA	CAGCGVCCV	CAGCGVAC	CACAGGVVFC	150
ACCAAGGGC	GVGAVVACCC	CGACAGGGV	VVCAAGVCA	GGVGVVCA	200
CVCAVCCAG	CAGCVGVFC	VGCGVGVFC	CAGAACGGV	ACCGGVVCC	250
ACCGCAVCA	CGVGVCCGG	ACCAVGGCA	CGAGAGGVV	CGACACCCG	300
GVGGVCCV	VGACGVVCC	GVGGVGVFC	CGACAGVCA	AGAGVCA	350
CAVCAVCA	GGVGVVFC	VGGCAGCAC	ACVGGACAG	AGAGCCAG	400
GGVGVGVV	CGVGAACAC	CGACAGVCA	AGVGVGGAG		450
VVCAVGVFC	CGACAGVCC	CVGGVGGG	GVGAVVAC	ACAGAGCA	500

Fig. 3 Un esempio di una sequenza di 4 simboli che trasmettono l'informazione biologica: i primi 500 caratteri del codice del vaccino mRNA BNT162b2 (Fonte: World Health Organization).

4 Dal codice sorgente al codice eseguibile

Un programma è scritto e salvato su un file, ma per essere eseguito deve essere tradotto in binario e caricato in memoria (figura 4). I software di base si occupano di tradurre un codice da codice sorgente a codice eseguibile:

- il **codice sorgente** è salvato su un file: *file sorgente*;
- il **compilatore** (o **Assembler**) effettua l'analisi sintattica del file sorgente e lo traduce in binario, generando il *file oggetto*;
- il **linker** «cuce» insieme diversi file oggetto in un unico file, generando il *file eseguibile* che è caricato in memoria per essere eseguito.

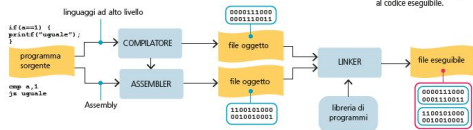


Fig. 4 Dal codice sorgente al codice eseguibile.

Per ogni linguaggio esiste un programma traduttore specifico; per alcuni esiste un interprete, per altri un compilatore.

Un **compilatore** effettua l'analisi sintattica di ogni istruzione, costruisce la **symbol table** (tabella dei simboli) che associa i nomi delle variabili agli indirizzi di memoria, e genera un **file oggetto** nel codice macchina del processore.

Un **interprete** effettua l'analisi sintattica dell'istruzione che, se corretta, viene subito eseguita in tempo reale. La traduzione di ogni istruzione comporta una maggiore lentezza nell'esecuzione rispetto a un programma che viene compilato.

notabene

Un esempio di Symbol Table è presente in diverse attività di laboratorio.

La parola a...



La sezione «La parola a...» dà voce ai protagonisti che hanno contribuito direttamente alla nascita e allo sviluppo delle teorie e delle applicazioni che vengono spiegate nel libro.

- Fornisce un inquadramento storico dei problemi citati, presentandoli attraverso le storie personali dei personaggi coinvolti.
- Risulta molto utile per comprendere il contesto e l'importanza delle scoperte e dei contributi dei diversi personaggi.
- Per esempio, nel volume 1 sono presenti **10** sezioni dedicate ai personaggi che hanno fatto la storia dell'informatica e delle reti.

La parola a... Ada Byron Lovelace, la madre di tutti i programmatori

Ada Byron Lovelace (1815-1852) può essere considerata la «madre di tutti i programmatori». Fu lei che cento anni prima che il calcolatore fosse inventato ebbe l'intuizione che «rendere un meccanismo capace di combinare simboli generali, in sequenze illimitate per varietà ed estensione, vuol dire avere trovato un punto di unione fra le operazioni della materia e i processi astratti della mente». Ada prefigura una macchina che può rappresentare, oltre ai numeri, qualsiasi elemento simbolico della realtà, come testi e musica. Non solo: Ada immagina la capacità di apprendere delle macchine, anticipando le soluzioni di Intelligenza Artificiale e mostrandone i limiti: «la macchina analitica non ha alcuna pretesa di concepire alcunché», ma solo di eseguire i compiti che noi le assegniamo.





Fig. 1. L'ecosistema Internet of Things.

1 Introduzione

Il **Physical Computing** è costituito da sistemi hardware e software in grado di **percepire**, grazie alla rilevazione di parametri fisici tramite sensori, il mondo che li circonda e di **interagire** con esso.

La rilevazione dei parametri fisici tramite sensori non è una novità, ma in questi ultimi anni, grazie alla convergenza delle nuove tecnologie con Internet, ha ripreso vigore e sta cambiando lo scenario mondiale. L'introduzione di progetti open source e delle schede a microcontrollore a basso consumo (come Raspberry Pi e Arduino) ha accelerato il processo di cambiamento.

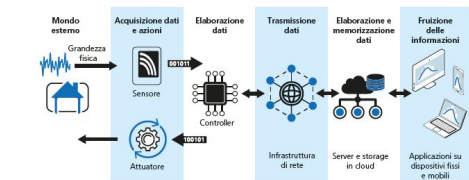
L'utilizzo dei dispositivi è passato da un contesto prevalentemente locale a una rete di oggetti che interagiscono dando vita a un ecosistema, chiamato **Internet delle cose (IoT, Internet of Things)** di cui fanno parte anche gli esseri umani (Figura 1); tutti gli elementi dell'ecosistema sono connessi tramite reti di comunicazione.

2 Introduzione all'Internet of Things (IoT)

Le «cose» dell'IoT sono gli **oggetti intelligenti (smart objects)**, dispositivi di varia natura in grado di collegarsi alla rete Internet e di interagire con il mondo esterno.

Lo scenario è quello di un ecosistema costituito da comunità di oggetti e persone che interagiscono tra loro e con l'ambiente che li circonda. Elemento centrale di questo complesso sistema sono i **dati**. I vari elementi dell'IoT possono essere molto diversi tra loro, ma tutti hanno a che fare con i dati. In generale, una catena completa parte dalla raccolta dei dati. Questi possono essere elaborati localmente, al fine di agire sul mondo esterno nell'immediato, oppure possono essere trasmessi a dispositivi remoti in grado di memorizzarli, elaborarli, analizzarli, al fine di ottenere informazioni utili (Figura 2).

L'IoT è uno dei modi per generare dati. Altre fonti sono, per esempio, i social media, le transazioni aziendali, che usano database (per esempio lo scontrino della spesa, l'acquisto di un biglietto aereo, il registro elettronico), la navigazione nel web, gli esperimenti scientifici. Tutti questi dati sono oggetto di una importante disciplina: la **scienza dei dati**.



La possibilità di trarre informazioni e conoscenza dai dati sta avendo un forte impatto sulla nostra vita quotidiana. Siamo in grado di intervenire meglio sull'ambiente, di risparmiare energia, di prendersi cura in modi nuovi delle persone, di facilitare il traffico, di rendere più efficienti i parcheggi, di rendere più

confortevoli le abitazioni. Il punto centrale rimane il limite, sempre incerto, tra la possibilità di intervenire in modo positivo su vari aspetti della nostra esistenza e il rischio di perdere la nostra identità e la nostra libertà.

In questa unità studieremo come funziona la parte iniziale della catena dei dati: il passaggio dal mondo fisico in cui viviamo alla sua trasformazione in dati tramite i **sensori** e le **azioni di controllo**.

3 Il controllo delle grandezze fisiche

Elementi base del Physical Computing sono i **sensori**, gli **attuatori** e i **controllori**. Le grandezze fisiche, cioè gli elementi misurabili che caratterizzano i fenomeni fisici, come per esempio la temperatura o la pressione, vengono rilevate tramite sensori, convertite in segnali digitali e inviate al controllore (microcontrollore) per la loro elaborazione. I risultati ottenuti dall'elaborazione possono essere usati per pilotare, tramite attuatori, dispositivi esterni come, per esempio, sistemi elettromeccanici e motori (Figura 3).



Pit Stop

4 Descrivi la funzione di ognuno dei tre elementi base del Physical Computing.

Sensori

Un sensore è un dispositivo che:

- **rileva e misura** le proprietà fisiche dell'ambiente (per esempio movimento, posizione, distanza, luminosità, suono, temperatura, umidità, pressione);
- **trasforma** le proprietà fisiche in un segnale di natura elettrica;
- **fornisce** il segnale di natura elettrica al sistema di elaborazione.

Esempi di sensori sono:

- il microfono, che converte l'energia dell'onda sonora in un segnale elettrico;
- una termoresistenza, che sfrutta la resistività di un conduttore, che a sua volta - variando al variare della temperatura - consente di risalire alla temperatura del corpo;
- i molti sensori contenuti in un cellulare: per esempio, l'accelerometro (un sensore che misura l'accelerazione), il giroscopio (un sensore che serve a misurare l'orientamento spaziale), la bussola (un sensore che misura la forza e la direzione di un campo magnetico), il GPS (un sensore che fornisce a un ricevitore informazioni sulle sue coordinate geografiche), il sensore di luminosità.

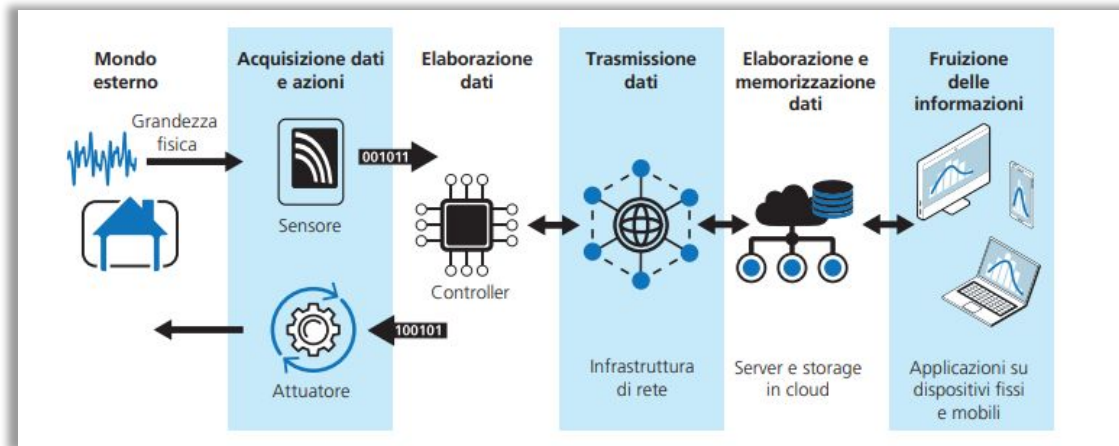
Pit Stop

5 Quali sono le funzioni di un sensore?

Percorso



- Il mondo dell'informatica è centrato sui DATI.
- Il percorso dell'opera è guidato dal flusso delle operazioni sui dati: raccolta, trasmissione, analisi, elaborazione e visualizzazione dei risultati.



Immagini e laboratorio

Tema 1 - Architettura dell'elaboratore

Per alcuni linguaggi esiste sia un interprete sia un compilatore. Per esempio:

- C e C++ sono linguaggi compilati, perché usano un compilatore;
- Javascript è un linguaggio interpretato;
- Java utilizza un compilatore che traduce il programma sorgente in un linguaggio intermedio (bytecode), che poi viene interpretato dalla Java Virtual Machine.

5 Il linguaggio Assembly

Studiare il linguaggio Assembly è importante per capire veramente come è fatto e come funziona un sistema di elaborazione. Usare l'Assembly è l'unico modo per interagire direttamente con la macchina.

Una generica istruzione in linguaggio macchina è formata da due parti:

<codice operativo> e <operandi>

Il **codice operativo** (operation code) indica l'istruzione che la CPU deve eseguire, gli **operandi** specificano i dati su cui l'istruzione deve operare.

Nel linguaggio Assembly il codice operativo del linguaggio macchina viene sostituito da un codice mnemonico che ricorda il significato dell'istruzione. Per esempio, il codice operativo della somma può essere rappresentato con `add`, il codice della differenza con `sub`, un salto con `jmp` e un trasferimento dati con `mov`. Un'istruzione può avere uno o più operandi, oppure non averne. In presenza di due operandi uno è la sorgente, l'altro la destinazione, cioè dove viene posto il risultato. Ogni Assembly ha la sua sintassi. Consideriamo una istruzione Assembly con due operandi, in cui il primo è la destinazione e il secondo la sorgente:

```
mov A,4
```

Il suo significato è: trasferisci il valore 4 (sorgente) nel registro A (destinazione). In genere un set di istruzioni Assembly è diviso in categorie in base al tipo di istruzione:

- istruzioni di trasferimento;
- istruzioni aritmetico-logiche;
- istruzioni di salto;
- istruzioni di I/O.

6 Il microprocessore DuplOne e il suo linguaggio Assembly

Esistono molti linguaggi Assembly, uno per ogni architettura, ma inizieremo lo studio del linguaggio a partire da un **microprocessore** appositamente inventato dagli Autori del corso per scopi didattici: il **DuplOne**. Il suo set di istruzioni è pensato esclusivamente per capire il funzionamento di base di un microprocessore, ma è verosimile, nel senso che utilizza istruzioni simili a quelle dei microprocessori che si trovano in commercio.

74

Il software (con DuplOne) - Unità 4

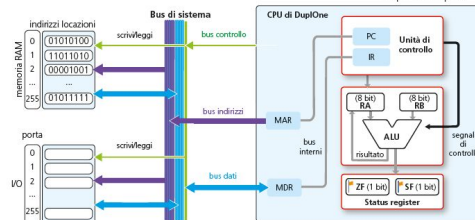
In seguito prenderemo in considerazione linguaggi Assembly reali come quello che sta alla base della famiglia x86, relativa ai microprocessori Intel, e il linguaggio dei microprocessori ARM che è attualmente il più diffuso, soprattutto nei dispositivi mobili e nelle single board come Raspberry Pi.

Architettura di DuplOne

L'architettura del microprocessore DuplOne è ridotta all'essenziale e prevede (Figura 5):

- **bus:**
 - bus dati a 8 bit;
 - bus indirizzi a 8 bit;
 - bus controllo;
- **registri:**
 - due registri di uso generale a 8 bit: registro A (accumulatore) e registro B;
 - Status Register con Flag di Zero, Flag di Segno;
 - Stack Pointer;
 - Program Counter;
 - MDR, MAR, IR;
- **organizzazione della memoria:**
 - locazioni di memoria di 1 byte;
 - è indirizzabile il singolo byte;
 - memoria indirizzabile: $2^8 = 256$ byte.

4 Fig. 5 L'architettura del microprocessore DuplOne.



Il linguaggio Assembly DuplOne

Il codice operativo di ogni istruzione nel linguaggio Assembly DuplOne è formato da 8 bit.

75



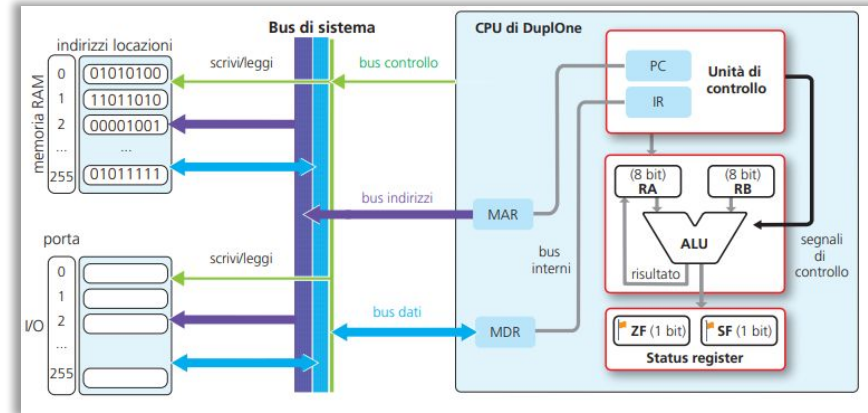
Immagini e laboratorio



Infografica e immagini

- sintetizzano le informazioni
- forniscono una visualizzazione di informazioni, problemi, flussi informativi, ...
- facilitano la comprensione, la memorizzazione e l'apprendimento

Nel libro: per esempio, nel volume 1 sono presenti circa **160 immagini** e **40 tabelle analogie** distribuite su tutte le unità



Immagini e laboratorio

Tema 1 - Architettura dell'elaboratore

Tab. 2

#	Istruzione Assembly - mnemonica	Codice Operativo (codice macchina binario)	hex	dec	Descrizione
Istruzioni di trasferimento dalla memoria in un registro					
1	mov A,[indirizzo]	0001 0011 nnnnnnnn (indirizzo)	13	19	Carica nel registro A il contenuto della locazione di memoria di indirizzo specificato
2	mov B,[indirizzo]	0001 1110 nnnnnnnn (indirizzo)	1E	30	Carica nel registro B il contenuto della locazione di memoria posta all'indirizzo specificato
Istruzioni di trasferimento da un registro alla memoria					
3	mov [indirizzo],A	0001 1001 nnnnnnnn (indirizzo)	19	25	Carica il contenuto del registro A nella locazione di memoria posta all'indirizzo specificato
4	mov <var>,B	0001 1010 nnnnnnnn (indirizzo)	1A	26	Carica il contenuto del registro B nella locazione di memoria posta all'indirizzo specificato
Istruzioni di trasferimento tra registri					
5	mov B,A	0001 0010	12	18	Carica il contenuto del registro A nel registro B
6	mov A,B	0001 0001	11	17	Carica il contenuto del registro B nel registro A
Istruzioni di trasferimento di un valore immediato in un registro					
7	mov A,n	0001 0110 nnnnnnnn (valore)	16	22	Carica nel registro A il valore n
8	mov B,n	0001 0111 nnnnnnnn (valore)	17	23	Carica nel registro B il valore n
Istruzioni aritmetico-logiche					
9	add A,B	0010 0001	21	33	A ← A + B Somma al contenuto del registro A il contenuto del registro B e pone il risultato nel registro A
10	add B,A	0010 0010	22	34	B ← B + A Somma al contenuto del registro B il contenuto del registro A e pone il risultato nel registro B
11	add A,n	00100011 nnnnnnnn (valore)	23	35	A ← A + nnnnnnn Somma al contenuto del registro A il valore n e pone il risultato nel registro A
12	sub A,B	0011 0001	31	49	A ← A - B Sottrae al contenuto del registro A il contenuto del registro B e pone il risultato nel registro A
13	inc A	0100 0001	41	67	A ← A + 1 Incrementa di 1 il valore di A
14	inc B	0101 0010	52	82	B ← B + 1 Incrementa di 1 il valore di B
15	dec A	0110 0001	61	97	A ← A - 1 Decrementa di 1 il valore di A
16	dec B	0111 0010	72	144	B ← B - 1 Decrementa di 1 il valore di B
17	cmp A,n	0011 0010 nnnnnnnn (valore)	32	50	Confronta (compara) il valore contenuto nel registro A con il valore n. È una sottrazione A - n

Il software (con Duplicare) - Unità 4

#	Istruzione Assembly - mnemonica	Codice Operativo (codice macchina binario)	hex	dec	Descrizione
18	not A	0011 0100	34	52	Pone nel registro A il valore di A negato
19	or A,B	0011 0101	35	53	Effettua l'OR logico bit a bit tra il contenuto del registro A e il contenuto del registro B e pone il risultato in A
20	and A,B	0011 0110	36	54	Effettua l'AND logico bit a bit tra il contenuto del registro A e il contenuto del registro B e pone il risultato in A
Istruzioni di salto					
21	jnz <indirizzo/ etichetta di salto>	1001 0010 nnnnnnnn (et/ind)	92	146	Salto condizionato: salta all'indirizzo di memoria indicato in <indirizzo/etichetta> se il flag Zero è a zero, cioè se è NON-Zero (NZ)
22	js <indirizzo/ etichetta di salto>	1001 0001 nnnnnnnn (et/ind)	91	145	Salto condizionato: salta all'indirizzo di memoria indicato in <indirizzo/etichetta> se il flag Zero è a 1, cioè se è Zero (Z)
23	jmp <indirizzo/ etichetta di salto>	1001 0000 nnnnnnnn (et/ind)	90	144	Salto non condizionato: salta all'indirizzo di memoria indicato in <indirizzo/etichetta>
Istruzioni che coinvolgono lo stack					
24	call <indirizzo/ etichetta di salto>	1001 1000 nnnnnnnn (et/ind)	98	152	Chiamata di routine: salva il PC nello stack tramite push e salta all'indirizzo di memoria indicato in <indirizzo/etichetta>
25	ret	1001 1001	99	153	Ripristina il PC, prelevandolo dallo stack tramite pop
26	push A	1001 0000	A0	160	Carica il contenuto del registro A sulla cima dello stack (locazione puntata dallo SP) e decrementa SP (SP ← SP - 1)
27	push B	1001 0001	A1	161	Carica il contenuto del registro B sulla cima dello stack (locazione puntata dallo SP) e decrementa SP (SP ← SP - 1)
28	pop A	1000 0000	80	128	Incrementa SP (SP ← SP + 1) e carica il contenuto della cima dello stack (locazione puntata da SP) nel registro A
29	pop B	1000 0001	81	129	Incrementa SP (SP ← SP + 1) e carica il contenuto della cima dello stack (locazione puntata da SP) nel registro B
Istruzioni di Ingresso / Uscita (Input/Output)					
30	in A, <indirizzo unità ingresso>	0000 1000 nnnnnnnn (indirizzo)	08	08	Carica nel registro A il contenuto della periferica posta all'indirizzo <indirizzo unità di ingresso> (parametro)
31	out A, <indirizzo unità>	0000 1001 nnnnnnnn (indirizzo)	09	09	Carica nella periferica posta all'indirizzo <indirizzo unità di uscita> il contenuto del registro A
Altre Istruzioni					
32	nop	0000 0000	00	00	Nessuna operazione
33	halt	1111 1111	FF	255	Ferma il programma



Immagini e laboratorio



L'attività di laboratorio consente di

- Apprendere in ambiente protetto
- Collaborare tra pari
- Risolvere problemi
- Utilizzare simulatori
- Sviluppare competenze
- Riflettere su esperienze pratiche
- Orientare al futuro



Immagini e laboratorio

Tema 3 - Architettura di rete: l'accesso alla rete

ATTIVITÀ 2

Mezzi fisici di trasmissione disponibili in Packet Tracer

In questa attività consideriamo i mezzi fisici di trasmissione disponibili per progettare una rete con Packet Tracer. La Figura 2 mostra una rete in cui sono utilizzati diversi mezzi fisici:

- USB;
- IoT;
- cavo Fast Ethernet (Fa) e Gigabit Ethernet (Gig) diretto e incrociato;
- coassiale (Coax);
- seriale (Se);
- fibra ottica (Fa e Gig);
- collegamento Wi-Fi;
- torre per telefoni cellulari (Cell Tower).

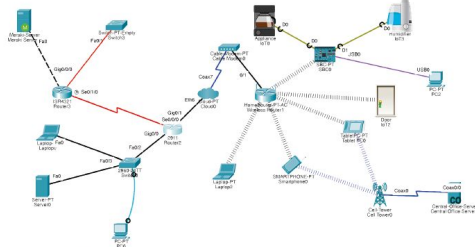


Fig. 2

La tabella seguente mostra alcuni cavi disponibili in Packet Tracer.

Console	Copper Straight-through	Copper Cross-over	Fibra Ottica	Phone	Coaxial	Seriale (per WAN)	Cavo ottico (audio-video)	USB	IoT

356

Il Livello Fisico - Unità II

La tabella seguente riporta la legenda e la descrizione dei diversi mezzi di trasmissione.

Tipo di cavo	Immagine	Descrizione
Console		Cavo per effettuare connessioni seriali asincrone (7/ 8 bit con parità e bit di Start/Stop) da un PC terminale (console) a un router o uno switch.
Copper Straight-through (cavo di rame «dritto»)		Cavo Ethernet standard in rame, per il collegamento alle interfacce dei dispositivi di rete (PC, router e switch) con velocità diverse: 10 Mbps (Ethernet), 100 Mbps (Fast Ethernet) e 1000 Mbps (Gigabit Ethernet).
Copper Cross-over (cavo di rame incrociato)		Cavo con le stesse funzioni di quello «dritto», ma con i pin di RX (ricezione) e TX (trasmissione) incrociati. Utile per collegamenti tra dispositivi simili (DTE-DTE, DCE-DCE), ad esempio PC-PC o Router-Router (DTE - Data Terminal Equipment: un qualsiasi dispositivo che è la sorgente o la destinazione di una comunicazione dati. DCE - Data Communication Equipment: un qualsiasi dispositivo utilizzato per interfacciare le linee di comunicazione).
Fibra Ottica		Supporto in fibra ottica per connessioni con porte in fibra (100 Mbps o 1000 Mbps).
Phone		Cavo (telefonico) per connessioni con modem.
Coaxial (coassiale)		Cavo coassiale.
Seriale (per WAN)		Cavi seriali per collegamenti WAN con clock o senza.
IoT custom cable		Cavo per collegare dispositivi intelligenti con microcontrollori (MCU) e computer a scheda singola (SBC). Il cavo raggruppa le linee dati, alimentazione e massa.
USB		Cavo USB per collegare porte seriali USB presenti nei diversi dispositivi.
Wireless		Collegamenti wireless tra «access point» e dispositivi finali (PC, server e stampanti).
Cell-Tower		Le torri per telefoni cellulari (stazioni base) sono strutture con un'antenna e dei trasmettitori e dei ricevitori situati nella parte alta.



ATTIVITÀ 3

La gestione Router Switch con Console cable

Disponibile nell'eBook.

ATTIVITÀ 4

Montare un cavo UTP per rete Ethernet

Il cavo di rete Ethernet è un doppino (TP, *Twisted Pair*) particolare che garantisce velocità dell'ordine dei 100/1000 Mbps. Come mostra la Figura 3, il TP è costituito da 4 coppie intrecciate di fili, ogni coppia è intrecciata per ridurre l'interferenza causata da disturbi elettromagnetici (EMI) e da segnali radio (RFI) prodotti dalle altre coppie o da altre apparecchiature elettriche.

I cavi UTP sono divisi in categorie corrispondenti a diverse velocità trasmissive supportate: la categoria 5e raggiunge i 100 Mbps, le categorie 6 e 7 consentono velocità dell'ordine dei Gbps. I fili che trasportano i dati hanno nomi e colori definiti dagli standard, e i connettori hanno piedinature anch'esse definite dagli standard (RJ45 - ISO 8877) (Figura 4).



Fig. 3 Il connettore RJ45 (sopra) e i cavi intrecciati del doppino TP (sotto).

357

Immagini e laboratorio



Nel libro: per esempio, nel volume 1 sono presenti **55 attività di laboratorio tutte svolte**.

Un laboratorio è relativo a una unità.
Ogni laboratorio è suddiviso in attività.

Per ogni attività sono specificati:

- il titolo (per il riferimento alla relativa teoria)
- l'obiettivo
- lo svolgimento passo passo



Immagini e laboratorio



ATTIVITÀ 11

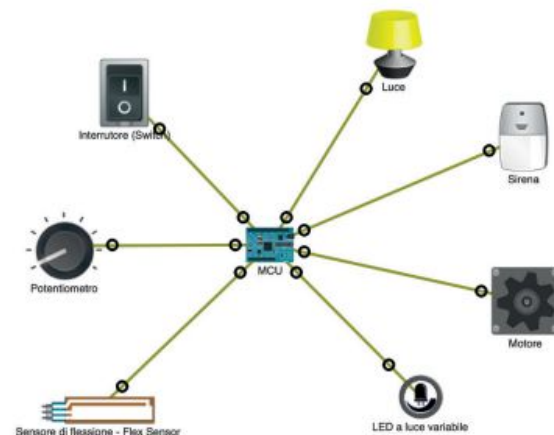
Automazione garage

Consegna

Vogliamo realizzare l'automazione di alcuni elementi di un piccolo garage per autovetture.

Svolgimento

Un interruttore (Figura 21) accende una lampada, un potenziometro agisce su un LED a luce variabile (*dimnable LED*), e l'apertura della porta del garage è ottenuta premendo con il piede un sensore di flessione a membrana (*flex sensor*) che ne rileva la pressione: il valore (analogico) aumenta proporzionalmente alla pressione esercitata sul sensore. Un allarme segnala l'apertura della porta. Il sistema è realizzato con una scheda MCU (Arduino) che gestisce, in linguaggio Python attraverso l'uso di funzioni, ingressi e uscite analogiche e digitali, come mostra la



.....
.....

Immagini e laboratorio



Interfaccia del
simulatore
DuplOne:

DuplOne Assembly Simulator

Load Reload Reset Step back Single step Run Scarica Pdf Memory Dump

Ind	Val	Binario	Istruzioni
00:	22	00010110	00: MOV A, constant
01:	16	00010000	01: 16
02:	23	00010111	02: MOV B, constant
03:	32	00100000	03: 32
04:	33	00100001	04: ADD A, B
05:	22	00010110	05: MOV A, constant
06:	3	00000011	06: 3
07:	23	00010111	07: MOV B, constant
08:	-1	11111111	08: -1

A	0	00000000
B	0	00000000
PC	0	00000000
SP	255	11111111
SR	0	0
	ZERO	CARRY



Approfondimento - Tecnologia

Sensori e MEMS

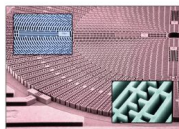


Fig. 1 Dettaglio di un sensore inerziale di STMicroelectronics. Mediante uno specifico processo produttivo (bulk micromachining) si utilizza la luce per scavare il silicio in modo da realizzare strutture tridimensionali. Al contrario, attraverso la tecnologia surface micromachining si riduce il wafer di silicio per costruirvi sopra dei filari stratificati. Fonte: STMicroelectronics.

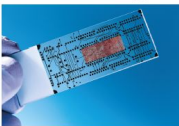


Fig. 2 Labo Chip (LOC) è un dispositivo che integra le funzioni di laboratorio su un nanochip. Il chip contiene un vero e proprio laboratorio in miniatura, in cui migliaia di reazioni possono essere eseguite in parallelo. I dispositivi LOC appartengono alla famiglia dei MEMS.

La **nanoelettronica** è una delle tecnologie più pervasive e ha un'enorme influenza sulla nostra vita quotidiana, sul nostro modo di lavorare, di divertirci e di comunicare. Ha alimentato e continuerà ad alimentare nel futuro lo sviluppo dell'informatica, dell'elettronica di consumo, delle telecomunicazioni e del settore industriale (basti pensare a quanta micro- e nanoelettronica ci sia oggi in un'automobile).

Tradizionalmente la **microelettronica** si occupava solo della parte centrale degli elaboratori, mentre la componentistica di altro tipo (elettromeccanica, tubi a vuoto, dispositivi discreti di potenza) si occupava delle interfacce verso il mondo fisico; ormai la microelettronica, o quantomeno l'infrastruttura tecnologica basata sul silicio, ha iniziato a presidiare pesantemente anche questa parte dei sistemi. Da circa vent'anni, nella microelettronica esistono due filoni di sviluppo: uno continua a perseguire la miniaturizzazione e l'aumento di densità come sua forza trainante, ed è chiamato «**more of Moore**», l'altro, chiamato «**more than Moore**», sfrutta le tecnologie microelettroniche per implementare funzioni diverse.

Una delle più importanti tecnologie del filone **more than Moore** è quella dei **MEMS** (**Micro Electro Mechanical Systems**, sistemi micro-elettro-meccanici). Oltre alle nuove proprietà elettriche, il silicio ha infatti anche ottime proprietà meccaniche; l'infrastruttura tecnologica della microelettronica consente di sfruttare queste proprietà per realizzare sistemi meccanici su scala micrometrica e permette anche di integrare o interfacciare direttamente questi microsistemi con l'elettronica di controllo.

Si possono realizzare per esempio oggetti come quello mostrato in **Figura 1**, molto apprezzabile anche dal punto di vista estetico.

È un esempio di **sensore inerziale**, in cui ci sono due parti interdigitate, una fissa e l'altra sospesa; in presenza di un'accelerazione angolare, la distanza relativa tra le due parti si modifica a causa dell'inerzia della parte sospesa. La variazione della distanza tra le due parti può essere rilevata misurando la capacità elettrica della struttura. In questo modo si realizza un **sensore di rotazione**.

Grazie a sensori inerziali MEMS, le console dei videogiochi consentono all'utente di giocare, simulando movimenti reali. Gli smartphone e i tablet sono dotati di sensori inerziali che consentono di mantenere verticale l'immagine sullo schermo quando si ruota l'apparecchio.

Giroscopi micromeccanici permettono ai telefoni cellulari di non perdere la localizzazione quando si cammina all'interno degli edifici. La tecnologia MEMS, basata sulle nanotecnologie, consente inoltre di integrare funzioni elettroniche, meccaniche, ottiche, biologiche o chimiche in spazi ridottissimi in un unico componente.

Un dispositivo MEMS contiene, su un minuscolo chip di silicio, microcircuiti integrati con dispositivi di tipo meccanico o sensori (**Figura 2**).

ORIENTAMENTO
STEM

In questo modo è possibile disporre di **sensori microelettronici**: giroscopi, magnetometri, accelerometri, bussole digitali, sensori per ultravioletti, sensori tattili, rilevatori di pressione e di umidità.

Altre applicazioni interessanti dei MEMS si hanno nella **robotica**, dove si fa largo uso di giroscopi e accelerometri per determinare la posizione nello spazio del sistema, la sua accelerazione o inclinazione. Poter controllare il movimento di un sistema, compresa la sua accelerazione o inclinazione, permette di ottenere applicazioni intuitive, realistiche, interattive e totalmente integrate (*embedded*). In **campo automobilistico** possiamo trovare i MEMS ovunque: airbag, sistemi di navigazione assistita, scatole nere. I MEMS giocano un ruolo fondamentale nei sistemi di **antifurto**, proprio perché in grado di rilevare gli spostamenti. Nel campo industriale e dei beni di consumo (per esempio gli elettrodomestici) i MEMS possono essere utilizzati per rilevare vibrazioni anomale e individuare usure nelle parti meccaniche.

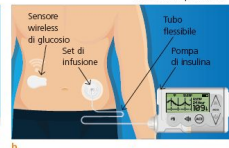
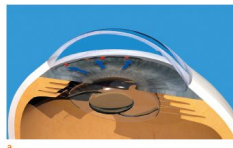


Fig. 3 L'applicazione dei MEMS al corpo umano come biosensori per la rilevazione della pressione oculare (a), o per la rilevazione del glucosio e la somministrazione dell'insulina con le pompe di insulina (b), aprono nuovi scenari in campo medico.

Anche in **ambito medicale** i MEMS trovano vaste applicazioni. Per esempio, vengono usati per il controllo della pressione oculare in patologie quali il diabete (**Figura 3a**), oppure nelle protesi per il movimento degli arti e per la somministrazione intelligente dell'insulina (**Figura 3b**).

Un'altra applicazione di successo delle tecnologie **more than Moore** è l'**imaging**. Anche le macchine fotografiche di alta gamma sono infatti dotate di sensori digitali realizzati su silicio: i sensori delle fotocamere sono matrici di transistor e condensatori, coperte da strati di materiali dielettrici trasparenti, opportunamente lavorati, che fanno da filtro di colore e da lente per ciascun pixel. Nelle videocamere è inoltre possibile compensare il movimento e stabilizzare le immagini. Nei proiettori di nuova generazione si trovano infine applicazioni per la gestione di microspecchi.

Numerose **app didattiche** per lo studio delle scienze sfruttano in modo accattivante e divertente i sensori interni degli smartphone (**Figura 4**). Le prospettive e i vantaggi sono enormi, anche se devono tenere presenti alcuni problemi legati a questi dispositivi, come il fatto di dover trattare segnali analogici di bassissima energia, con una differenza minima tra segnale e rumore. Anche la progettazione e la produzione di questi microsensori presenta problemi, per esempio legati a stress termici o di altra natura.

La produzione dei MEMS in grandi quantità permette la riduzione dei costi e soprattutto, nei prossimi anni, li renderà particolarmente convenienti e versatili, aprendo scenari tutti ancora da esplorare.



Fig. 4 L'app Science consente di utilizzare i sensori interni dello smartphone per misurare e monitorare luce, suono, movimento, campi magnetici.

Educazione civica

Educazione civica

Lo smart working

Spunti di riflessione

Che cos'è lo smart working

Il telelavoro, inteso come attività lavorativa svolta in una sede diversa dai locali aziendali e supportata dalla tecnologia, ha avuto una forte accelerazione durante il periodo della pandemia di Covid-19 perché contribuiva a garantire la continuità dei processi produttivi evitando il contatto tra le persone. In Italia questa modalità di lavoro era poco diffusa e stentava a diffondersi a causa principalmente di vuoti normativi, di aspetti culturali, di carenza di infrastrutture e di competenze tecniche. Durante la pandemia, per necessità, ci si è trovati costretti a improvvisare ambienti di lavoro remoti e a organizzarsi in pochissimo tempo, utilizzando i mezzi e gli strumenti a disposizione.

Con il passare del tempo, la situazione si è evoluta: dal telelavoro si è passati allo smart working.

Lo smart working (detto anche lavoro agile) è una nuova concezione del lavoro, che diventa flessibile in termini di orari e spazi. È un modello organizzativo che può avere notevoli vantaggi, ma richiede responsabilità e capacità organizzativa, sia da parte del lavoratore che da parte dell'azienda. È una sfida culturale, perché lo smart working esce dagli ambienti di nicchia nei quali è nato e si diffonde tra i lavoratori, diventando popolare.



Che cosa è necessario per poter lavorare in modalità smart working?

Per poter far parte di questa rivoluzione culturale è necessario:

- **Essere aperti a un cambiamento culturale al lavoro.** Il lavoratore deve essere responsabile, organizzato e motivato al raggiungimento degli obiettivi; il datore di lavoro deve trovare una modalità adeguata a misurare e valutare il lavoro svolto dal lavoratore, abbandonando l'idea ormai obsoleta di valutare il lavoro in termini di ore di presenza in ufficio. Anche se può apparire scontato, questo cambio di mentalità è lento e non sempre accettato.
- **Disporre di una dotazione tecnologica adeguata.** Tante sono le tecnologie e gli strumenti necessari per poter lavorare in modalità smart working in sicurezza, ad esempio: una rete internet con elevata velocità di trasmissione, dispositivi hardware come PC e schermi ad alta risoluzione, una rete VPN, software applicativi, software per la collaborazione.
- **Possedere solide competenze digitali.** La tecnologia deve essere utilizzata in modo corretto e consapevole, prestando particolare attenzione alla sicurezza di un sistema informatico.
- **Disporre di spazi adeguati.** Il lavoratore, anche fuori dai locali aziendali, deve sempre disporre di una postazione di lavoro che rispetti i criteri di sicurezza e preservi il benessere fisico. Il datore di lavoro può organizzare e gestire in modo molto flessibile l'ambiente di lavoro nella sede aziendale, superando la logica della postazione fissa.

La normativa che regola lo smart working in Italia

In Italia lo smart working è chiamato "lavoro agile" ed è regolato dalla Legge n.81 del 22 maggio 2017 (anche detta Legge sul lavoro agile). La legge definisce gli aspetti giuridici relativi ai diritti dei lavoratori, agli strumenti tecnologici, alle modalità di lavoro. Di particolare rilievo sono gli aspetti legati alla sicurezza, sia in termini di salute del lavoratore, sia in termini di sicurezza informatica.

In particolare, la direttiva NIS (Direttiva sulla sicurezza delle reti e dei sistemi informativi dell'Unione, Direttiva NISUE 2016/1148) attuata nel nostro ordinamento con il Decreto Legislativo 18 maggio 2018, n. 65 e con il decreto del Presidente del Consiglio dei ministri 313/2020 afferma che investire nella formazione digitale del personale e nella sicurezza interna è lo strumento più efficace per tutte le imprese e gli operatori. I temi principali trattati in questa direttiva riguardano: il costante aggiornamento del software e dei

La cultura del lavoro

sistemi, la previsione di sistemi di autenticazione con password complesse, adeguati sistemi di backup dei dati, l'utilizzo di reti private virtuali (VPN) per la navigazione, test continui della sicurezza dei sistemi per individuare per tempo eventuali falle.

Attività per lo studente

Obiettivi

L'approfondimento di questo tema ti consentirà di:

- conoscere e comprendere le implicazioni sociali e tecnologiche dello smart working;
- migliorare la tua capacità di lavorare a distanza per obiettivi, in sicurezza, in gruppo.



Nuclei tematici fondamentali e suggerimenti per l'approfondimento

- Che cos'è lo smart working?
- Quali sono le normative vigenti?
- Quali sono le infrastrutture necessarie?
- Perché può essere considerato un cambiamento culturale?
- Quali sono i rischi legati alla sicurezza informatica?
- Che cosa significa "lavorare per obiettivi"?
- Quali vantaggi può avere lo smart working per il lavoratore?
- Quali risvolti negativi può avere lo smart working nella vita privata dei lavoratori?

Fasi operative

Fase	Attività	Fase	Attività
1 Analisi preliminare (individuale)	Ogni studente, lavorando individualmente: • ricerca sul web informazioni relative allo smart working; • sintetizza e organizza le informazioni raccolte, evidenziando i nuclei tematici fondamentali individuati.	3 Condivisione dell'elaborato (in team di 2/3 studenti)	Ogni gruppo presenta alla classe e affinisce il proprio elaborato
2 Produzione di un elaborato (in team di 2/3 persone)	La classe viene suddivisa in gruppi composti da 2-3 persone. Ogni gruppo produce un elaborato (documento testuale, documento multimediale, video, presentazione...) che spieghi i nuclei tematici individuati nella fase precedente. Per ogni nucleo tematico si può includere: • un breve video (2-3 minuti) tratto dal web; • un testo di 15-20 righe; • un'infografica; • immagini esplicative.	4 Riflessione condivisa (tutte le classi)	Per quali motivi il lavoro fatto sullo smart working ti fa essere un migliore cittadino attivo? Confrontati con il resto della classe e rifletti anche alla luce della vostra esperienza riguardo a: • organizzazione del tempo; • responsabilità; • capacità di ascolto; • rispetto dei turni di parola; • rispetto delle regole in rete.
	È opportuno alternare momenti di lavoro "in presenza" e momenti di lavoro "da remoto"; ad esempio: • organizzazione del gruppo, scelta del tipo di elaborato, suddivisione dei compiti → in presenza; • produzione dell'elaborato e preparazione dell'esposizione orale → da remoto (individualmente o in modo collaborativo); • controllo finale dell'elaborato prodotto e simulazione della sua presentazione alla classe → in presenza.	5 Valutazione	Ogni gruppo valuta il lavoro degli altri esprimendo un giudizio su completezza, chiarezza espositiva, originalità, collaborazione. L'insegnante valuta i lavori usando i criteri di valutazione definiti e tenendo conto delle valutazioni degli studenti. È possibile assegnare un bonus per particolari aspetti emersi dal lavoro presentato.



L'impatto dell'informatica sull'ambiente



Spunti di riflessione

Alcune delle tendenze in atto negli ultimi anni nell'ambito delle tecnologie informatiche hanno un impatto di segno opposto sull'ambiente: positivo per certi aspetti, negativo per altri. Consideriamo due esempi.

A partire dall'inizio del 2020, molte attività hanno decisamente incrementato la modalità on-line: ad esempio lo smart working o la didattica a distanza. Questa tendenza era già in atto, ma ha avuto una notevole accelerazione. In ambito energetico, sensori wireless e tecnologie di monitoraggio hanno permesso di sviluppare sistemi intelligenti che ottimizzano la gestione dell'energia attraverso il controllo di parametri come la temperatura o la luminosità.

Da un lato, il nuovo modo con cui lavoriamo e comunichiamo riduce l'impatto ambientale: basti pensare, per esempio, al minor inquinamento dovuto a un numero inferiore di spostamenti sui mezzi di trasporto. Dall'altro lato, la maggiore dipendenza dai servizi online e dai dispositivi intelligenti comporta una crescita notevole dei consumi energetici con la conseguente maggiore emissione di anidride carbonica, che è la principale causa del riscaldamento globale. La rete Internet, con i sistemi utilizzati per sostenerla, genera circa il 4% delle emissioni di CO₂, paragonabili ai consumi del settore aereo. Infatti, anche se il termine cloud ("nuvola informatica") rimanda a una sensazione di leggerezza, in realtà esso è un'infrastruttura fisica costituita da migliaia di cavi e fibre ottiche che collegano router e computer (che devono essere raffreddati) e richiede quantitativi di energia enormi.

Per quanto riguarda, in particolare, l'impatto del software sull'ambiente, sono state proposte linee guida che suggeriscono tecniche appropriate di progettazione e di programmazione.

Il modello Greensoft

Il "Modello Greensoft" considera il ciclo di vita di un "software verde e sostenibile" fornendo i **criteri di sostenibilità da applicare a un software dalla sua nascita al suo ritiro dal mercato**. L'obiettivo è consentire alle parti interessate, dagli sviluppatori agli utilizzatori, di valutare l'impatto del software in base a tre criteri: prodotto, processo di produzione e consumo.

Il green coding

Anche le tecniche di programmazione e, addirittura, il tipo di linguaggio, possono influire sui consumi energetici. La figura classifica i diversi linguaggi di programmazione in base alla loro efficienza:



Il "green coding" suggerisce pratiche di programmazione che si propongono di ridurre l'impatto ambientale delle attività di sviluppo del software. Ricordiamo alcune di queste "buone pratiche".

- Scegliere linguaggi di programmazione adatti al problema da risolvere, evitando sprechi di memoria e della capacità di calcolo della CPU.
- Utilizzare algoritmi efficienti che riducono il numero di operazioni e il tempo di esecuzione.
- Scrivere il codice in modo semplice e riutilizzabile.
- Migliorare il codice esistente eliminando parti superflue o ridondanti ("code refactoring").
- Abbassare, se possibile, la risoluzione e la qualità di immagini e video in modo da ridurre la quantità di dati che sono inviati in rete.
- Utilizzare il cloud computing per ottimizzare l'uso delle risorse.



- Preferire fonti di energia rinnovabili per l'alimentazione del server.

Queste pratiche, in ogni caso, devono rispecchiare le scelte consapevoli che vanno adottate a livello organizzativo e personale per uno sviluppo del software sostenibile e a basso impatto ambientale.

Attività per lo studente

Obiettivi

Approfondire il tema dell'impatto dell'informatica sull'ambiente ti consentirà di:

- comprendere le implicazioni sociali e tecnologiche dell'impatto ambientale derivante dall'utilizzo dell'hardware e del software;
- individuare le "buone pratiche" della progettazione software.



Nuclei tematici fondamentali e suggerimenti per l'approfondimento

- Che cosa si intende per sostenibilità ambientale?
- In che modo il software sostenibile permette di conciliare innovazione tecnologica e sostenibilità ambientale?
- Che cos'è il ciclo di vita per un software verde e sostenibile?
- Quali implicazioni personali e sociali comporta lo sviluppo di un software sostenibile?
- Elenca almeno quattro buone pratiche del green coding.

Fasi operative

Fase	Attività	Fase	Attività
1 Analisi preliminare (individuale)	Ogni studente, lavorando individualmente: • ricerca sui web informazioni sull'impatto ambientale del software e la sua sostenibilità; • sintetizza e organizza le informazioni raccolte, evidenziando i nuclei tematici fondamentali individuati.	3 Condivisione dell'elaborato (in team di 2/3 studenti)	Ogni gruppo presenta alla classe e all'insegnante il proprio elaborato.
2 Produzione di un elaborato (in team di 2/3 persone)	La classe viene suddivisa in gruppi composti da 2-3 persone. Ogni gruppo produce un elaborato (documento testuale, documento multimediale, video, presentazione...) che spieghi i nuclei tematici individuati nella fase precedente. Per ogni nucleo tematico si può includere: • un breve video (2-3 minuti) tratto dal web; • un testo di 15-20 righe; • un'infografica; • immagini esplicative. È opportuno alternare momenti di lavoro "in presenza" e momenti di lavoro "da remoto", ad esempio: • organizzazione del gruppo, scelta del tipo di elaborato, suddivisione dei compiti → in presenza; • produzione dell'elaborato e preparazione dell'esposizione orale → da remoto (individualmente o in modo collaborativo); • controllo finale dell'elaborato prodotto e simulazione della sua presentazione alla classe → in presenza.	4 Riflessione conclusiva (tutta la classe)	In che modo lo studio sulla sostenibilità del software incide sulla tua vita personale e professionale? Confrontati con il resto della classe e rifletti anche alla luce della vostra esperienza. Tieni conto di tutti i fattori coinvolti, tra cui i seguenti: • algoritmi conosciuti e utilizzati; • linguaggi di programmazione; • misura e riutilizzo del codice (metodi, funzioni, oggetti, ricorsività...); • tempo di esecuzione della CPU e occupazione di memoria; • tempo di scaricamento di una app e occupazione della memoria; • utilizzo dei servizi in cloud per eseguire applicazioni su server e come repository di dati, immagini, video, musica...
		5 Valutazione	Ogni gruppo valuta il lavoro degli altri esprimendo un giudizio su completezza, chiarezza espositiva, originalità, collaborazione. L'insegnante valuta i lavori usando i criteri di valutazione definiti e tenendo conto delle valutazioni degli studenti. È possibile assegnare un bonus per particolari aspetti emersi dal lavoro presentato.





DEASCUOLA

INSEGNARE SISTEMI E RETI

GRAZIE

SPAZIO ALLE DOMANDE



Scopri "Gateway Terza edizione"

Consulta la [**scheda sul sito**](#)

Contatta il tuo agente di zona:
<https://deascuola.it/rete-commerciale/>



<https://formazione.deascuola.it/insegnare-sistemi-e-reti/>

Insegnare Sistemi e Reti

nella Scuola secondaria di Secondo Grado



CICLO DI APPUNTAMENTI GRATUITI PER I DOCENTI DI **SISTEMI E RETI** DELLA SCUOLA SECONDARIA DI **SECONDO GRADO**

Spunti e suggerimenti di temi da portare in classe e consigli per riuscire a coinvolgere studentesse e studenti.

