

> ELETTRONICA PROGRAMMABILE - SSIS Toscana

• Introduzione

- Presentazione del docente
- Presentazione del corso - Obiettivi formativi, contenuti, Prerequisiti

• Premesse

- Sistemi elettronici "cablati"

la funzionalità della rete logica è racchiusa in componenti attivi e interconnessioni.

> Famiglie logiche e board.

- Difficile sfruttare le potenzialità delle tecnologie p.e.l.e.r.
- Se si fanno circuiti più complessi, si RIDUCE il mercato.

> ASIC (circuiti elettronici integrati progettati "ad hoc")

- Convenienti solo per produzioni > di $10^4 \div 10^5$ unità/anno
- Difficili da progettare - Costi costosi -
- Molto tempo (mesi) per andare in produzione.
- Elevatissimi costi non ricorrenti (15K€)

- Nuove strade per la realizzazione di sistemi elettronici

> Caratteristiche ideali

- Separare funzionalità e realizzazione fisica
- Riservare al produttore SOLO e' hardware (grandi numeri)
- Mettere l'utente in grado di

DESCRIVERE, VERIFICARE, IMPLEMENTARE la funzionalità nell'hardware disponibile

PROCESSO di trasferimento VELOCE, economico e senza necessità di particolari apparecchi.

- Riuscire a realizzare grandi e complessi sistemi che sfruttino le potenzialità della Meletronica ($10^6 \div 10^7$ MOS).

- Elementi chiave per realizzare "Sistemi programmi."

> Elementi in grado di conservare informazione in forma elettrica/logica: REGISTRI, MEMORIE

> Connessioni controllate da segnali elettronici

• Possibili soluzioni per l'elettronica programmabile

1- Applicare il concetto di programmabilità alle funzionalità di blocchi elementari, interconnessioni e interfacce con l'esterno.

> Il processo di programmazione "plasma" gli elementi e le connessioni del chip, che diviene praticamente un circuito dedicato.

> PAL, PLD, FPGA

2- Individuare architetture "generali" in grado di eseguire sequenzialmente operazioni diverse, secondo quanto è scritto in una apposita memoria

> Il circuito non viene modificato dal programma. Il programma si identifica con la sequenza delle "istruzioni" collocate in memoria

> Processori: CPU, μ C, PLC, DSP

- confronto tra le due soluzioni

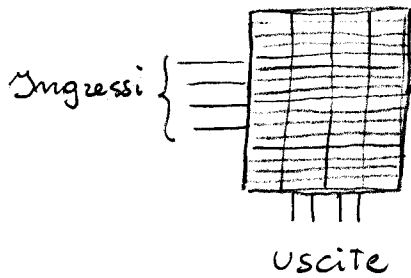
Programmabili	Processori
⊕ Mantengono le caratteristiche di prestazione di un circuito dedicato	sostanzialmente più lenti di un dedicato. Richiedono diverse instr. per risolvere un problema
Complessità dei problemi risolti dipendente dalle dimensioni del chip	Complessità legata alle dimensioni della memoria. Capacità "teorica" di risolvere QUALSIASI problema ⊕
Tempo di progetto complesso con requisiti di CAD stringenti (sintesi)	Ampia disponibilità di strumenti di sviluppo (compilatori/assemblatori) e verifica (debugger) ⊕

• Dispositivi programmabili (cenni)

- Programmabilità degli elementi funzionali

> Esistono realizzazioni "strutturate" di reti combinatorie che permettono di impostare la funzione da realizzare tramite elementi di memoria

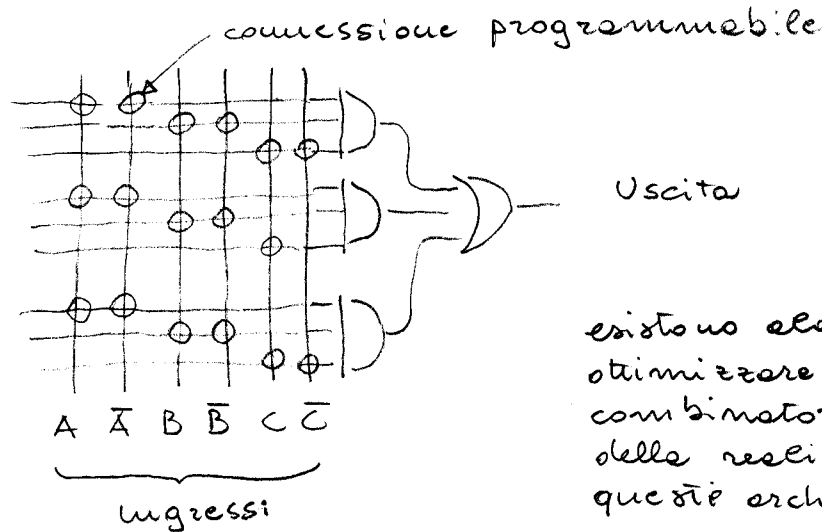
• la memoria stessa (nella fase di lettura)



riga i = deve essere scritta ponendo il valore i-esimo della tabella di verità (rispettivamente delle 4 variabili di uscite)

• la matrice logica programmabile

Permette di realizzare una rete combinatoria come SOMMA di PRODOTTI



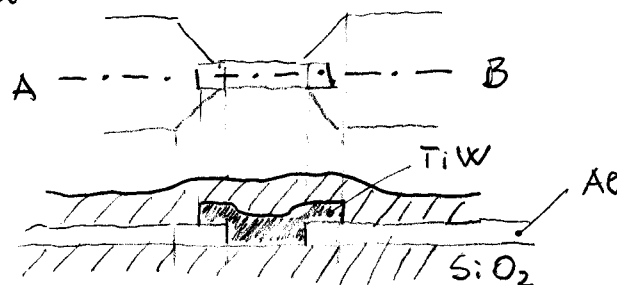
esistono algoritmi per ottimizzare una funzione combinatoria in vista della realizzazione con queste architetture

- Programmabilità delle connessioni

> Agli aspetti di architettura si sovrappongono quelli tecnologici. Occorre la disponibilità di metodi di interconnessione impostabili dall'utente

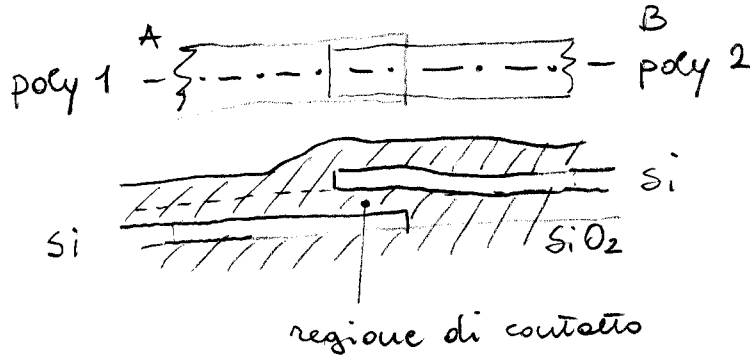
> Permeenti

• Fusibili



Rottura per fusione (effetto Joule)

• Antifusibili

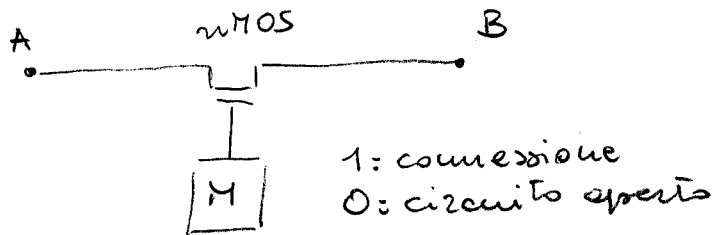


Rottura del dielettrico per superamento del valore limite di campo \vec{E}

> Modificabili, dopo un processo di cancellazione (NON volatili)

> Modificabili "in linea"

• Transistori MOS pilotati da celle di memoria di vario tipo (EPROM, E²PROM, Flash E²PROM, SRAM)



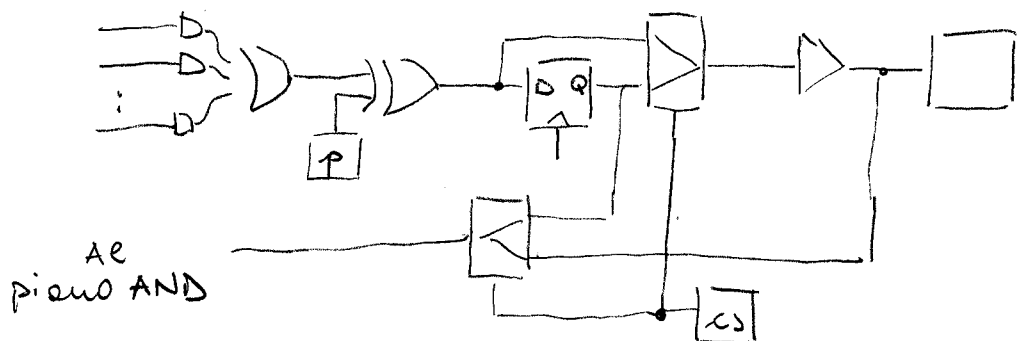
la memoria, tipicamente, prevede in questo caso un accesso seriale.

- Esempio di dispositivi di tipo "GAL".

Analisi dell'architettura e partire dal data-sheet

> Si evidenzia la struttura "somma di prodotti" a cui è aggiunta una particolare macrocella finale.

• Nelle macrocella sono presenti bit di programma "architetturali", nel senso che alterano la funzionalità, riconfigurando l'architettura.



- Evoluzione dei dispositivi programmabili
Esempio di dispositivi di tipo "Aere" "

> Si può incrementare la capacità di elaborazione "quantitativamente": aumentare i termini prodotto e il numero di macrocelle.

- Limitazione: la complessità dei problemi non cresce solo per quantità. Occorrono approcci articolati, con numerosi sottosistemi organizzati gerarchicamente

- Per questo motivo non si è andati oltre le 10 macrocelle (serie 22V10) e si è preferito assemblare nello stesso chip più architetture tipo GAL, dando loro la possibilità di comunicare reciprocamente (con matrici di interconnessione programmabili).

> Dispositivi di questo tipo (array di array progr., con matrice di interconnessione) si definiscono PLD.

- Esempio della macrocella e degli espansi
- cenno al problema della stima delle prestazioni

la configurabilità comporta prestazioni differenziate in funzione dei bit di programmazione

importanza dei microparametri e dello scheme funzionale per la loro combinazione

- Riferimento a scheme e parametri di una PLD altera.

- Programmabili tipo FPGA (field programmable GA) 1.6

> Un ulteriore passo avanti verso prestazioni più elevate si ha con architetture di tipo matriciale, con blocchi elementari persino più semplici di una PAL, ma con disposizione bidimensionale e ampie risorse di connettività

- la capacità di programmare connessioni viene resa "gerarchica". Si prevedono connettività a breve, medio, largo raggio e globali.

- la cella, che può avere dimensioni diverse, è arricchita di registri e altre funzionalità (fine grain, large grain)

- I terminali di I/O acquisiscono ampie possibilità di configurazione (in, out, memoria, slew rate ...)

- Programmabili dell'ultima generazione (SoPC)

> Oggi, gli stessi processi tecnologici permettono di realizzare FPGA da diversi MGate.

- Oltre alle normali architetture, sono disponibili risorse aggiuntive per realizzare sistemi più complessi:

- MEMORIA RAM
- ARCHITETTURE TIPO "PROCESSORE"
- MOLTIPLICATORE DI FREQUENZA PER CLOCK, PLL
- BLOCCHI ANALOGICI E DI CONVERSIONE

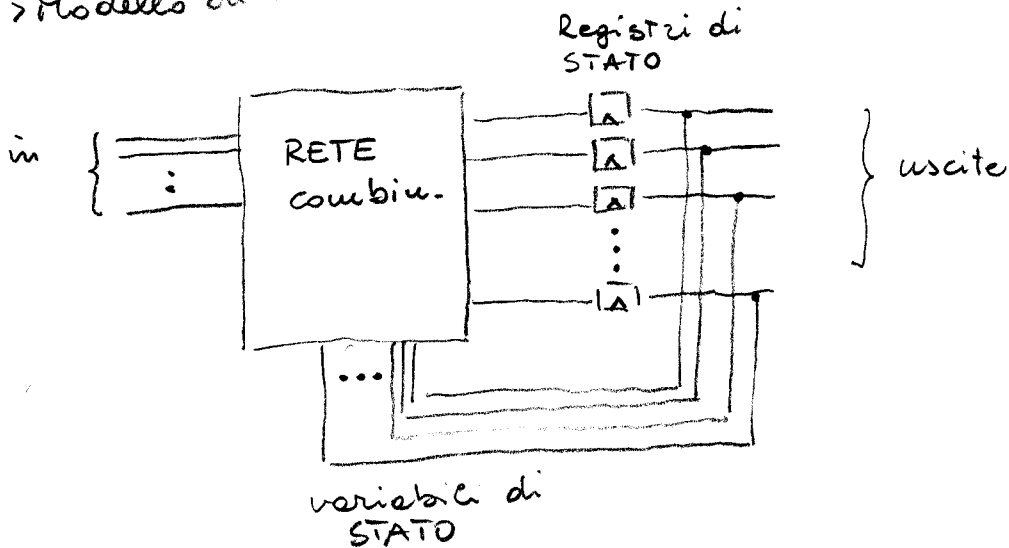
• Processori: sistemi a controllo di programma

1.7

- Circuiti in grado di eseguire sequenze di istruzioni conservate in un dispositivo specifico (RAM, ROM)

- Macchine sequenziali complesse sincrone
Partiamo dal modello base di rete logica.

> Modello di Moore



E' evidente che la funzionalità della rete è contenuta nel cablaggio delle rete combinatoria.

> Idea: cerchiamo di separare capacità "OPERATIVE" della capacità di gestire il flusso delle operazioni

• Alcuni registri si "specializzano" e supportano dell'esecuzione delle operazioni (appoggio ai dati)

• Altri servono per gestire il "flusso" di esecuzione del programma

- Modello di macchina divisa in PARTE OPERATIVA e PARTE CONTROLLO

