

Sistemi Embedded

Tecnologie hardware

*Luigi Pomante
Università dell'Aquila – DEWS
luigi.pomante@univaq.it*

Sommario

- Tecnologie hardware
 - Tecnologie (AS)IC
 - Full custom
 - Standard cell
 - Gate array
 - Tecnologie programmabili (configurabili)
 - Dalle ROM ai CPLD
 - FPGA

Tecnologie hardware

Tecnologie hardware

- Le tecnologie hardware stanno alla base della realizzazione di gran parte delle componenti di un sistema embedded
 - Con questo termine ci si riferisce a tutte quelle tecnologie che consentono di realizzare fisicamente i circuiti
- Ad un primo livello di analisi vi è un'unica tecnologia hardware e cioè la tecnologia *integrata*
 - Fine degli anni '50: *processo planare*
 - Ha cambiato radicalmente il modo dell'elettronica rendendo possibile la fabbricazione simultanea di milioni/miliardi di componenti

Tecnologie hardware

- Le possibilità cui ci si trova di fronte sono le seguenti
 - COTS
 - Componenti, spesso integrati, che svolgono funzioni specifiche e possono essere acquistati sul mercato e utilizzati direttamente
 - ASIC
 - I circuiti integrati sviluppati specificamente per una sola applicazione, utili quando si richiedono altissime prestazioni oppure quando il mercato richiede volumi molto elevati
 - Logiche programmabili
 - Componenti integrati che mettono a disposizione del progettista una grande quantità di risorse logiche, di memorizzazione e di interconnessione
 - Microprocessori
 - Utili qualora le funzioni che il progettista intende realizzare siano complesse e non disponibili sotto forma di componenti predefiniti

Sistemi embedded
2010/2011

5

Tecnologie hardware

Tecnologie (AS)IC

Tecnologie (AS)IC

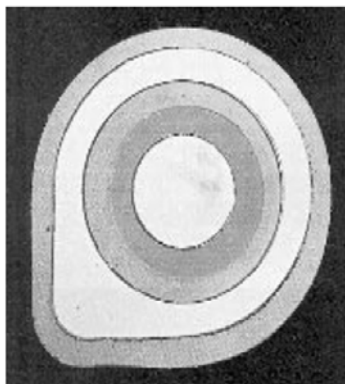
- La tecnologia ASIC nasce nel 1958 grazie all'invenzione, ad opera di Jean Hoerni (e altri) del processo planare
 - Costoro, lasciarono gli Shockley Semiconductor Laboratory nel 1957 per fondare, nello stesso anno la Fairchild Semiconductor, oggi ST Microelectronics
 - Tra questi vale la pena di ricordare Gordon Moore e Robert Noyce, fondatori nel 1968, della Intel Corporation, Victor Grinich, Eugene Kleiner, Sheldon Roberts e Jay Last co-fondatori di quella che nel tempo sarebbe divenuta nota come Silicon Valley
- Questa nuova tecnologia ha aperto nuove frontiere nel campo dell'elettronica e per questa ragione è da molti ritenuta l'evento che ha dato inizio all'era digitale

Sistemi embedded
2010/2011

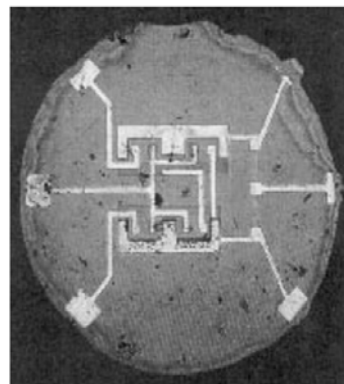
7

Tecnologie (AS)IC

- Fotografie del primo transistor e del primo circuito integrato mai realizzati



Sistemi embedded
2010/2011



8

Tecnologie (AS)IC

- Concetti di base del processo planare
 - In primo luogo è importante ricordare che il processo planare si basa sull'uso di un semiconduttore (Ge, Si, GaAs) le cui caratteristiche fisiche e, di conseguenza, elettromagnetiche, vengono modificate in fasi successive e in aree controllate
 - In secondo luogo è utile notare che tutti i componenti elettronici fondamentali nella realizzazione di circuiti digitali (transistor, diodi, resistenze e condensatori) possono essere realizzati in silicio, opportunamente trattato
 - Le interconnessioni tra i vari componenti sono invece realizzate mediante metalli quali alluminio e, solo recentemente, rame

Sistemi embedded
2010/2011

9

Tecnologie (AS)IC

- Elementi di base del processo planare
 - Silicio n
 - Drogante di tipo n, cioè dotato di un numero di elettroni in banda di conduzione maggiore rispetto al silicio
 - Diffusione o impiantazione ionica
 - Silicio p
 - Drogante di tipo p, cioè un elemento dotato di un numero di elettroni in banda di conduzione minore rispetto al silicio
 - Isolante
 - Ossido di silicio (SiO_2), ottenuto per esposizione all'ossigeno
 - Conduttore
 - Deposizione di vapori di alluminio (Al) o rame (Cu)

Sistemi embedded
2010/2011

10

Tecnologie (AS)IC

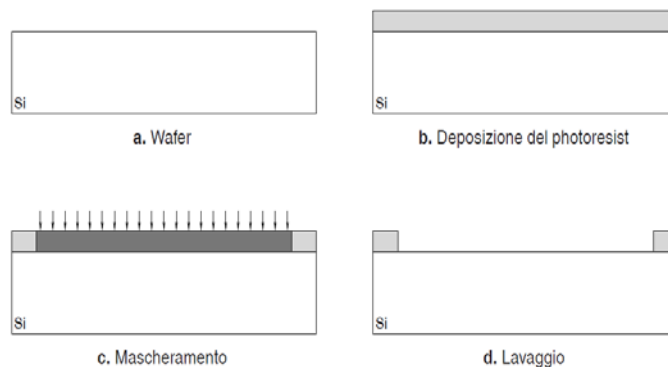
- Ognuno di questi elementi viene fabbricato durante un passo del processo planare
 - Il grande vantaggio di questa tecnologia sta nel fatto che tutte le zone di un certo tipo vengono fabbricate contemporaneamente agendo sull'intera superficie della lamina di silicio
 - Per ogni singolo passo sono necessarie più fasi, più o meno identiche nei diversi casi, che prendono il nome di **maschereature**
 - Lo scopo di tali fasi è quello di preparare il silicio in modo tale che lo specifico processo (diffusione, impiantazione, ossidazione, ecc...) avvenga selettivamente in zone ben precise

Sistemi embedded
2010/2011

11

Tecnologie (AS)IC

- Processo planare: fasi principali (1)

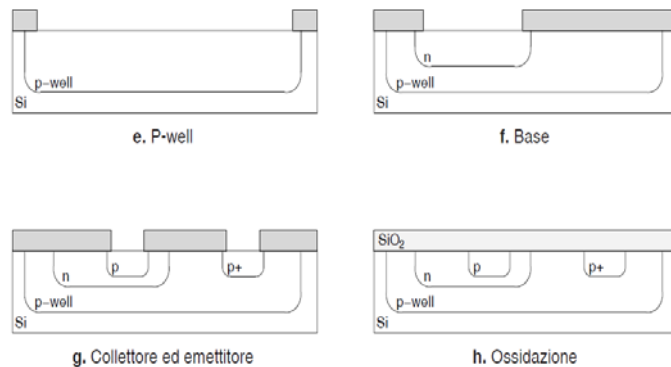


Sistemi embedded
2010/2011

12

Tecnologie (AS)IC

- Processo planare: fasi principali (2)

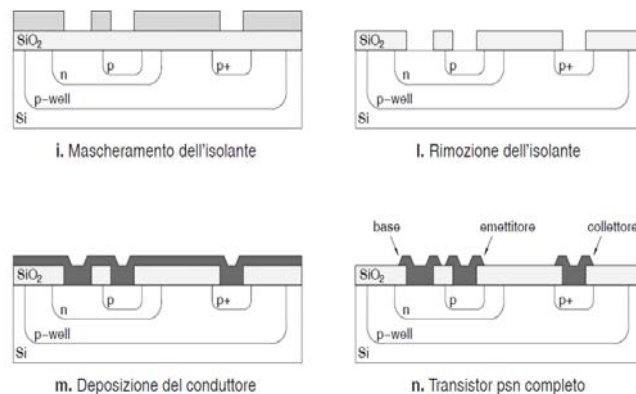


Sistemi embedded
2010/2011

13

Tecnologie (AS)IC

- Processo planare: fasi principali (3)



Sistemi embedded
2010/2011

14

Tecnologie (AS)IC

- Di seguito si realizzano le interconnessioni
 - Non è possibile isolare le singole linee (*net*)
 - Si utilizzano più strati distinti e separati da un isolante
 - Ogni strato prende il nome di *metal layer*
- Il processo appena descritto si riferisce ai transistor bipolari ormai rimpiazzati dai CMOS
 - Passi differenti ma tecniche simili

Grandezza	Descrizione	Valore
Minimum feature size	Dimensione minima della geometria delle maschere	$1 \div 10 \mu\text{m}$
Channel size	Dimensione del canale dei transistor CMOS	$25 \div 65 \text{ nm}$
Maschere	Numero di maschere richieste dal processo	$24 \div 36$
Metal	Numero di livelli di metallizzazione	$6 \div 10$

Sistemi embedded
2010/2011

15

Tecnologie (AS)IC

- Con il processo CMOS è possibile realizzare strutture circuitali generiche rispetto alla dimensione dei transistor, la loro posizione, il loro orientamento, la dimensione e la forma delle interconnessioni e così via
 - Nella progettazione di piccoli circuiti tutta questa flessibilità e libertà nell'organizzazione dei componenti sul silicio è una caratteristica estremamente apprezzabile
 - Si parla in questo caso di design **full custom**

Sistemi embedded
2010/2011

16

Tecnologie (AS)IC

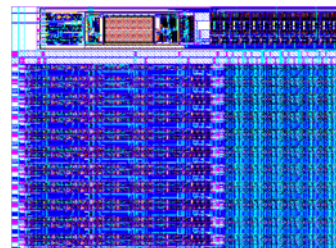
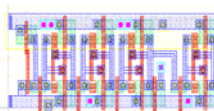
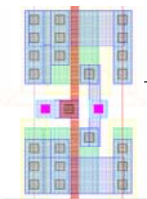
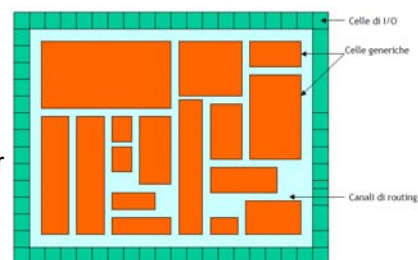
- Quando si considera la progettazione di grandi circuiti digitali, sfruttare appieno la flessibilità offerta dal processo planare diviene troppo complesso
 - Ridurre i gradi di libertà e ricorrere a strutture rigide e regolari
 - Tecnologia **standard cell**
- Avendo constatato che è possibile realizzare circuiti digitali assolutamente generici ricorrendo a strutture molto regolari
 - Tecnologia **gate array** o **sea of gates**
 - Secondo questo approccio, la parte attiva di transistor viene pre-fabbricata, lasciando i terminali disconnessi
 - Al progettista rimane il compito di tracciare le interconnessioni e svolgere le fasi finali del processo planare (metallizzazioni)

Sistemi embedded
2010/2011

17

Tecnologie (AS)IC

- **Full Custom**
 - Geometria realizzata a mano
 - Digitale e analogico
 - Simulazione a livello di transistor
 - Densità elevata
 - Prestazioni elevate
 - Tempo di progetto molto lungo



Sistemi embedded
2010/2011

18

Tecnologie (AS)IC

- *Standard cell*
 - Si alza il livello astrazione
 - Una **cella** è un piccolo circuito digitale completo già ottimizzato che il progettista può utilizzare ignorando molti dei dettagli interni
 - Librerie tecnologiche specifiche dei diversi *silicon vendor*
 - » 500-2000 tipi di celle che realizzano una ampissima gamma di funzioni: logiche di base, combinatorie complesse, elementi sequenziali, memorie
 - Una caratteristica delle celle che semplifica il back-end è che tutte hanno esattamente la stessa altezza
 - Questa tecnologia discretizza i processi di piazzamento e routing, vincolando le celle a posizioni specifiche e prefissate
 - Per quanto riguarda il routing, ci si basa su di una griglia ortogonale di linee lungo le quali le net sono vincolate a correre
 - » *Geometria Manhattan*

Sistemi embedded
2010/2011

19

Tecnologie (AS)IC

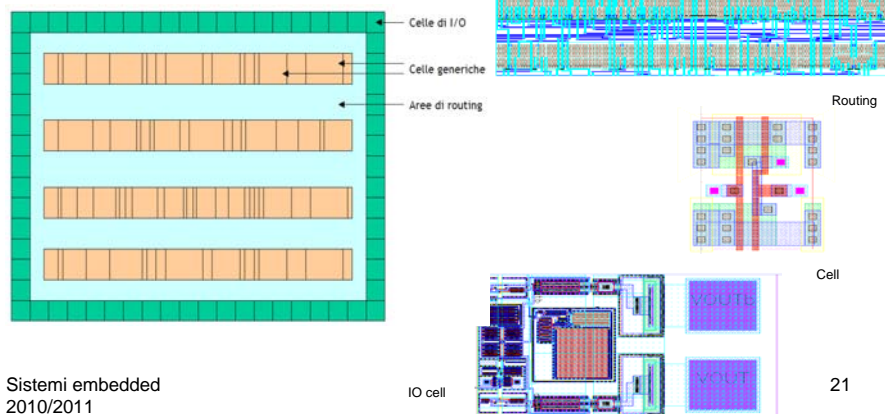
- *Standard cell*
 - Per il progettista una cella è caratterizzata da diverse informazioni
 - Nome logico o reference con il quale indicarla
 - Modello funzionale
 - Proprietà non funzionali
 - Numero, nome e posizione relativa dei pin
 - Assi di simmetria o *flip*
 - Angoli di rotazione o *orientation*
 - Grazie a queste informazioni è possibile, durante il flusso, procedere al mapping tecnologico, all'ottimizzazione locale, alla static timing analysis ed infine al placement e al routing

Sistemi embedded
2010/2011

20

Tecnologie (AS)IC

- *Standard cell*

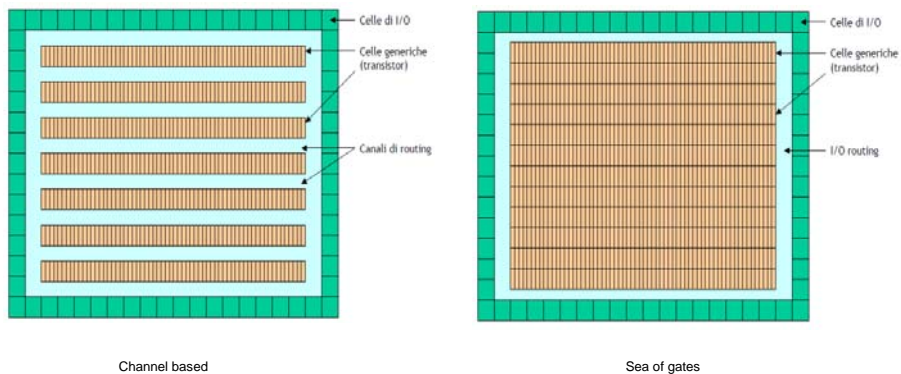


Tecnologie (AS)IC

- *Gate array (o Sea of Gate)*
 - Si tratta ancora di un approccio a celle
 - Le celle sono transistor o porte NAND
 - Sono già presenti sul dispositivo
 - La posizione delle celle è completamente fissata
 - Ogni design richiede solo alcune delle maschere per la realizzazione delle connessioni in quanto le celle sono prefabbricate

Tecnologie (AS)IC

- *Gate array (o Sea of Gate)*



Sistemi embedded
2010/2011

23

Tecnologie hardware

Tecnologie programmabili

Tecnologie programmabili

- Sono dispositivi hardware che mettono a disposizione componenti logici più o meno complessi che possono essere connessi tra loro a seconda delle esigenze
 - Componenti logici
 - Porte logiche, Flip-flop, Buffer...
 - Linee di connessione
- Tipologie di dispositivi programmabili
 - ROM, PLA, PAL
 - GAL, CPLD
 - FPGA

Sistemi embedded
2010/2011

25

Tecnologie programmabili

- I diversi tipi di dispositivi possono essere classificati sulla base di due aspetti
 - *Modalità di Programmazione*
 - Programmabili una sola volta (*One-Time Programmable* - OTP)
 - Fuse, Antifuse
 - Riprogrammabili (*Reprogrammable*)
 - E²PROM, SRAM, Flash
 - Riconfigurabili (*Reprogrammable & Reconfigurable*)
 - SRAM, Flash
 - *Connessioni*
 - Globali
 - Locali e distribuite

Sistemi embedded
2010/2011

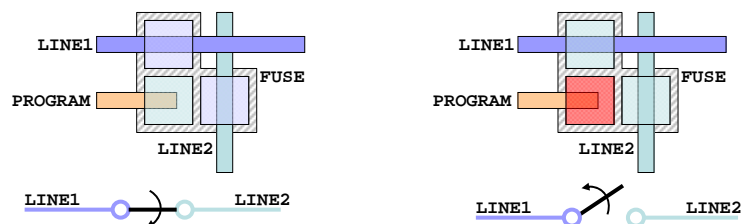
26

Tecnologie programmabili

- Modalità di programmazione

- Fuse (OTP)

- Le linee sono prodotte in modo da essere sempre connesse
 - La programmazione consiste nel “bruciare” (*fuse*) alcune connessioni in modo da mantenere solo quelle necessarie
 - La programmazione avviene mediante una tensione più elevata di quella di normale funzionamento



Sistemi embedded
2010/2011

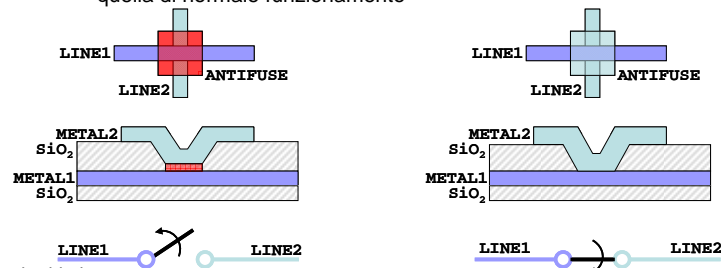
27

Tecnologie programmabili

- Modalità di programmazione

- Antifuse (OTP)

- Le linee sono prodotte in modo da essere sempre disconnesse
 - La programmazione consiste nel “creare” (*antifuse*) le connessioni necessarie
 - La programmazione avviene mediante una tensione più elevata di quella di normale funzionamento



Sistemi embedded
2010/2011

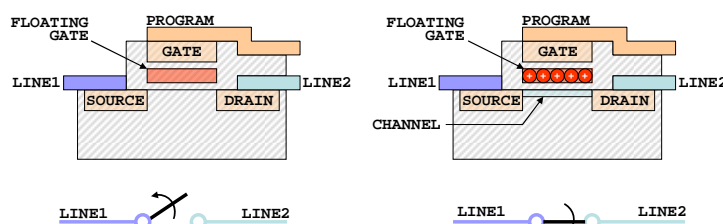
28

Tecnologie programmabili

- Modalità di programmazione

- E²PROM (*Reprogrammable*)

- Le connessioni tra le linee, inizialmente tutte non attive, possono essere attivate e disattivate elettricamente in modo non distruttivo durante la fase **non operativa** del dispositivo
 - La programmazione consiste nel depositare carica sul *floating gate* del transistor in modo da mantenerlo in conduzione



Sistemi embedded
2010/2011

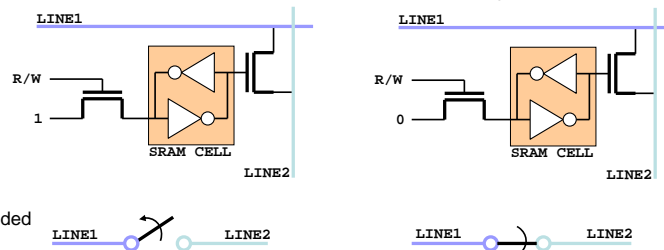
29

Tecnologie programmabili

- Modalità di programmazione

- SRAM (*Reprogrammable e Reconfigurable*)

- Le connessioni tra le linee, inizialmente tutte non attive, possono essere attivate e disattivate elettricamente in modo non distruttivo
 - La programmazione consiste nel memorizzare un valore logico (0 o 1) in una cella di RAM statica
 - In relazione alla tecnologia, la programmazione può avvenire
 - » durante la fase non operativa del dispositivo (*Reprogrammable*)
 - » durante la fase operativa del dispositivo (*Reconfigurable*)

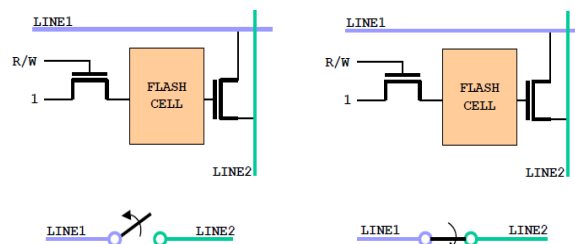


Sistemi embedded
2010/2011

30

Tecnologie programmabili

- Modalità di programmazione
 - Flash (*Reprogrammable* e *Reconfigurable*)
 - Le connessioni tra le linee, inizialmente tutte non attive, possono essere attivate e disattivate elettricamente in modo non distruttivo
 - La programmazione consiste nel memorizzare un valore logico (0 o 1) in una cella di memoria Flash
 - » Non volatile



Sistemi embedded
2010/2011

31

Tecnologie programmabili

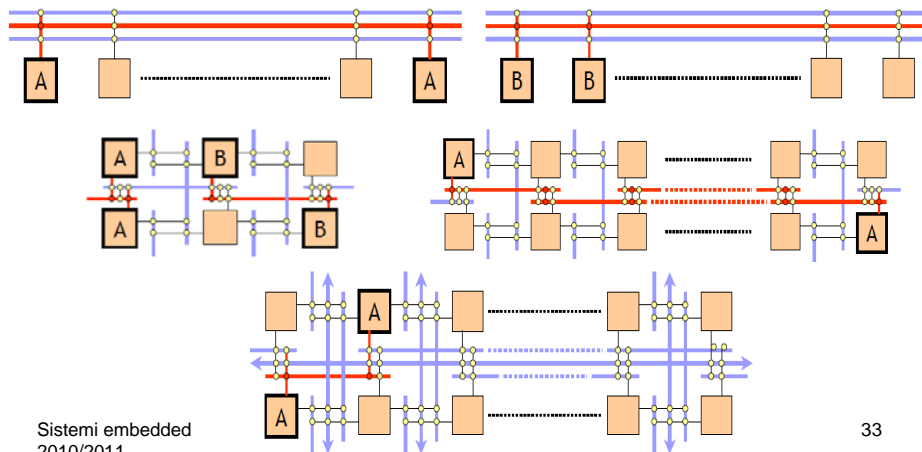
- Connessioni
 - Connessione Globale
 - Linea che attraversa buona parte del dispositivo e che è condivisa da molti elementi logici
 - Elevati ritardi
 - Può essere usata come uscita di **un** solo elemento logico limitandone la flessibilità
 - Connessione locale
 - Linea che attraversa una parte ridotta del dispositivo e che è condivisa da pochi elementi logici
 - Ritardi contenuti
 - All'interno di uno stesso dispositivo possono coesistere differenti linee locali di lunghezza differente
 - » Elevata flessibilità e complessità del routing
 - Connessione gerarchica

Sistemi embedded
2010/2011

32

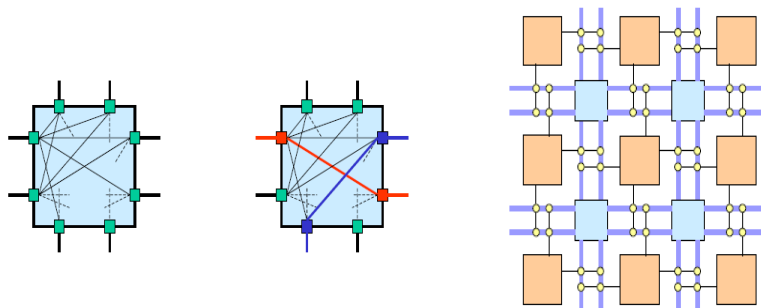
Tecnologie programmabili

- Connessioni



Tecnologie programmabili

- Connessioni
 - Altre tipologie
 - *Programmable Switch Matrix*



Tecnologie programmabili

- Connessioni
 - Connessioni globali
 - Dispositivi logici programmabili a 2 livelli
 - Read-Only Memory (ROM/PROM)
 - Programmable Logic Array (PAL)
 - Programmable Array Logic (PLA)
 - PAL e PLA Avanzate
 - Generic Array Logic (GAL)
 - Complex Programmable Logic Devices (CPLD)
 - Connessioni locali e distribuite (lunghe e corte)
 - Field Programmable Gate Array (FPGA).

Sistemi embedded
2010/2011

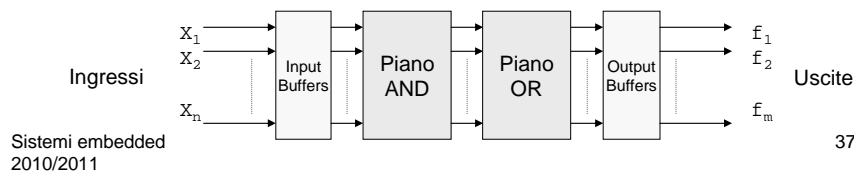
35

Tecnologie programmabili

Logiche programmabili a 2 livelli

Logiche programmabili a 2 livelli

- Sono usate per realizzare generiche funzioni combinatorie ad n ingressi e m uscite
 - $f_i = f_i(x_1, x_2, \dots, x_n)$ con $i=\{1, 2, \dots, m\}$
 - Si possono realizzare funzioni a più livelli o sequenziali retroazionando gli ingressi e aggiungendo elementi di memoria
 - Dispongono di
 - Un numero di ingressi fissato
 - Un numero di uscite fissato
 - Un piano AND, per la costruzione dei mintermini o implicanti
 - Un piano OR, per la somma dei mintermini o implicanti
 - Di Buffer di ingresso e di uscita (per ragioni elettriche e funzionali)

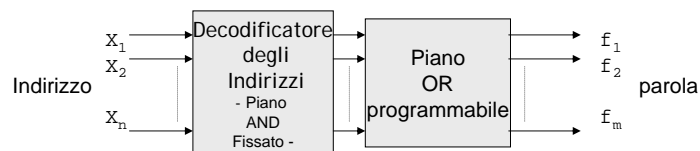


Logiche programmabili a 2 livelli

- Tre tipi principali
 - *Read-Only Memory (ROM/PROM)*
 - Piano AND fissato
 - Implementa tutti i possibili mintermini (decoder)
 - Piano di OR adattabile
 - *Programmable Logic Array (PLA)*
 - Piano AND programmabile
 - Implementa solo i mintermini/implicanti necessari
 - Piano OR programmabile.
 - *Programmable Array Logic (PAL)*
 - Piano AND programmabile.
 - Si costruiscono solo i mintermini/implicanti necessari
 - Piano di OR fissato
 - Impone un vincolo sul numero dei termini prodotto

Logiche programmabili a 2 livelli

- Read-Only Memory (ROM/PROM)
 - Un Memoria a Sola Lettura (ROM) implementa la prima forma canonica di m funzioni a n ingressi
 - Somma di Prodotti (SOP)
 - In una ROM, una configurazione di ingresso, denominata *indirizzo*, è associata una configurazione di uscita, denominata *parola*

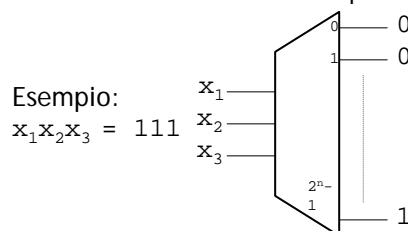


Sistemi embedded
2010/2011

39

Logiche programmabili a 2 livelli

- Read-Only Memory (ROM/PROM)
 - Decodificatore degli indirizzi (*Address decoder*)
 - Il decodificatore degli indirizzi (*address decoder*) nelle memorie ROM realizza tutti i 2^n mintermini
 - n variabili di ingresso x_i
 - Gli ingressi sono le variabili x_i
 - Una ed una sola uscita attiva alla volta
 - Le uscite sono tutti i mintermini costruiti a partire dagli ingressi

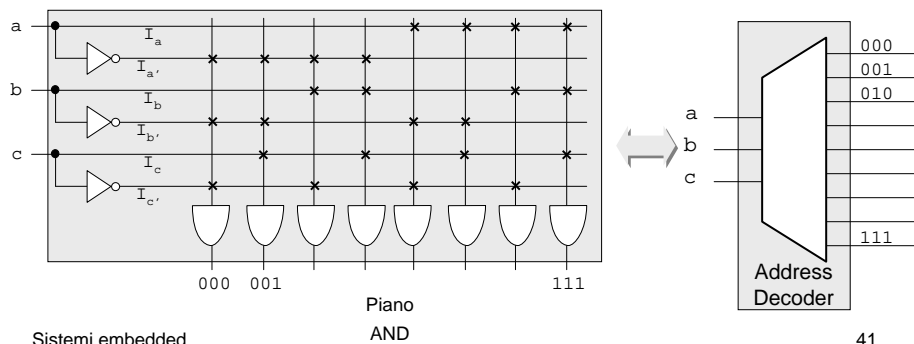


Sistemi embedded
2010/2011

40

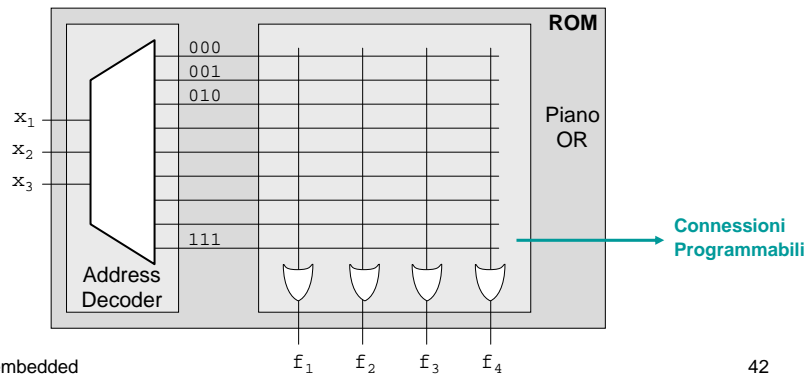
Logiche programmabili a 2 livelli

- Read-Only Memory (ROM/PROM)
 - Schema logico del piano di AND (*Address Decoder*)
 - Per semplicità si utilizza la rappresentazione che si riferisce al *decoder*



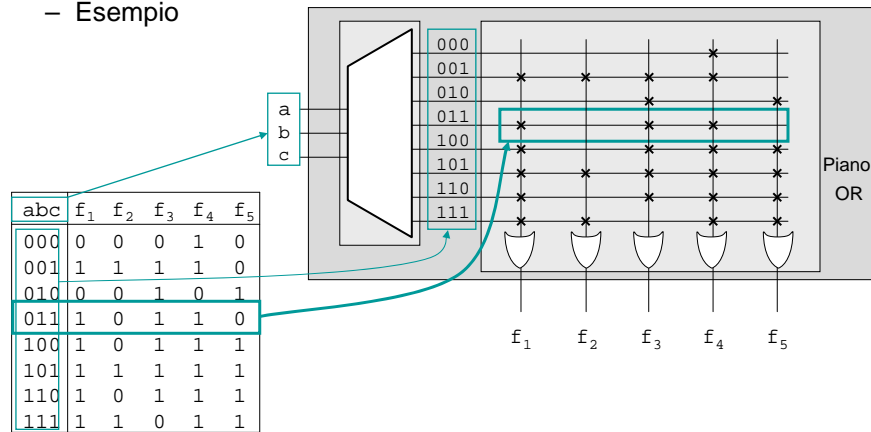
Logiche programmabili a 2 livelli

- Read-Only Memory (ROM/PROM)
 - Schema logico di una ROM
 - Esempio di una ROM a 3 ingressi e 4 uscite (non programmata)



Logiche programmabili a 2 livelli

- Read-Only Memory (ROM/PROM)
 - Esempio

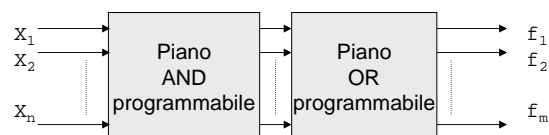


Sistemi embedded
2010/2011

43

Logiche programmabili a 2 livelli

- Programmable Logic Array (PLA)
 - Un Array Logico Programmabile (*Programmable Logic Array - PLA*) consente di implementare una somma di prodotti espressa in forma minima a due livelli (*somma di implicant*)

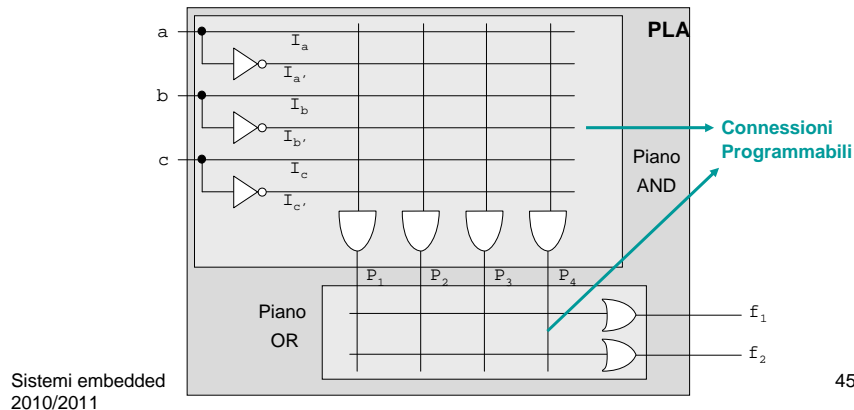


Sistemi embedded
2010/2011

44

Logiche programmabili a 2 livelli

- Programmable Logic Array (PLA)
 - Schema logico di una PLA
 - Esempio di PLA a 3 ingressi 2 uscite (non programmata):



Logiche programmabili a 2 livelli

- Programmable Logic Array (PLA)
 - Esempio (1/2)

- Realizzazione delle funzioni:
 - $f_1 = ab + ac' + a'b'c$
 - $f_2 = ab + ac + a'b'c$

- Prodotti
 - P₁ = ab
 - P₂ = ac
 - P₃ = ac'
 - P₄ = a'b'c



Formato PLA

11-	10
1-0	10
001	10
11-	01
1-1	01
001	01

- Somme
 - $f_1 = P_1 + P_3 + P_4$
 - $f_2 = P_1 + P_2 + P_4$

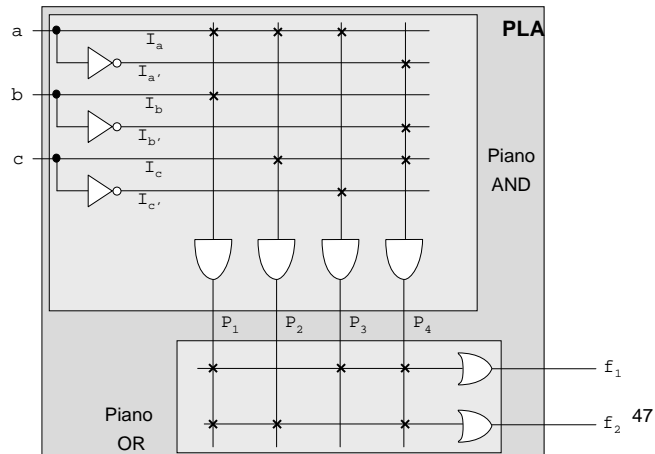
Logiche programmabili a 2 livelli

- Programmable Logic Array (PLA)
- Esempio (2/2)

Formato PLA

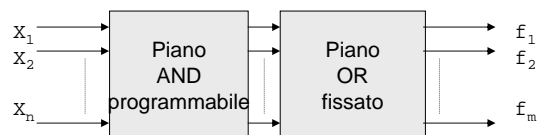
11- 10
1-0 10
001 10
11- 01
1-1 01
001 01

Sistemi embedded
2010/2011



Logiche programmabili a 2 livelli

- Programmable Array Logic (PAL)
- Un Array Programmabile Logico (PAL) consente di implementare una somma di prodotti espressa in forma minima a due livelli (*somma di implicanti*)
 - PLA e PAL coprono lo stesso spazio d'applicazione

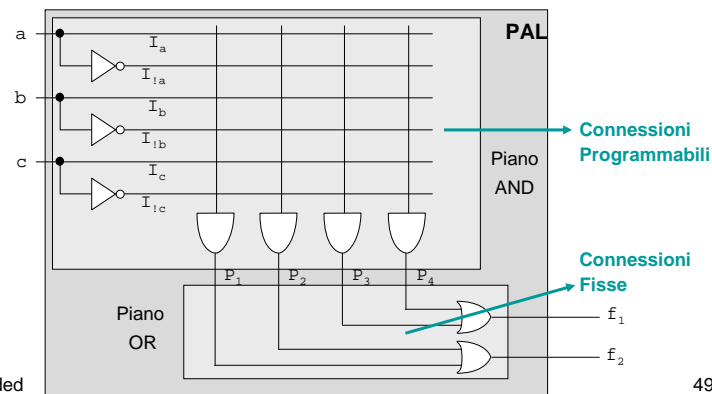


Sistemi embedded
2010/2011

48

Logiche programmabili a 2 livelli

- Programmable Array Logic (PAL)
 - Schema logico di una PAL
 - Esempio di PAL a 3 ingressi 2 uscite (non programmata):



Sistemi embedded
2010/2011

49

Tecnologie programmabili

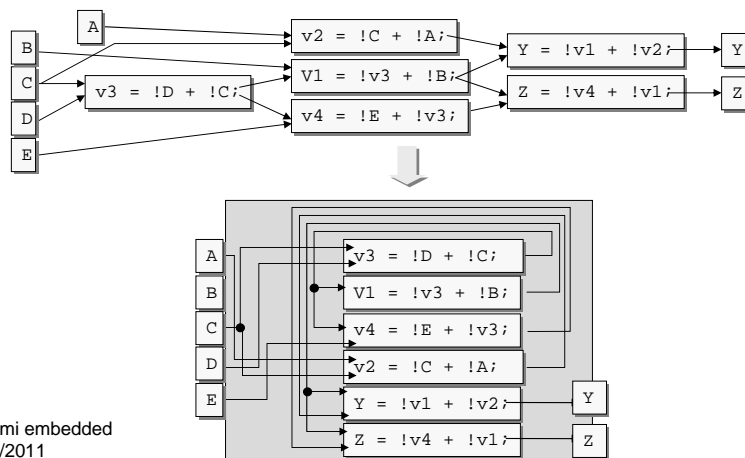
Logiche programmabili a più livelli

Logiche programmabili a più livelli

- PLA e PAL Avanzate
 - PLA e PAL consentono di realizzare solo reti combinatorie a due livelli
 - Questo limite può essere superato
 1. Introducendo delle linee di retroazione
 - Permette di implementare reti combinatorie multi livello a più uscite
 2. Introducendo elementi di memoria (bistabili)
 - Permette di implementare macchine sequenziali sincrone in cui la parte combinatoria è costituita da una rete multi livello a più uscite.

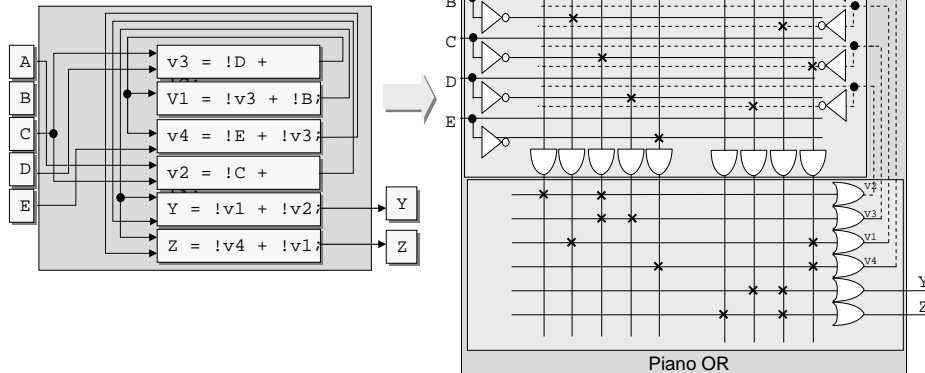
Logiche programmabili a più livelli

- PLA e PAL Avanzate
 - Esempio di una rete multi livello a più uscite (1/2)



Logiche programmabili a più livelli

- PLA e PAL Avanzate
 - Esempio (2/2)

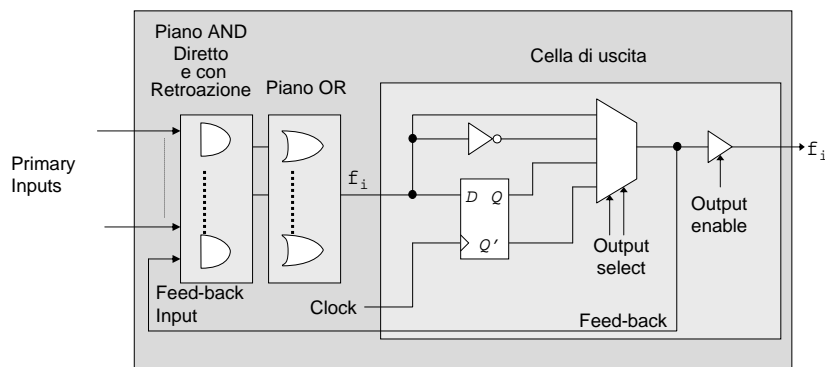


Sistemi embedded
2010/2011

53

Logiche programmabili a più livelli

- PLA e PAL Avanzate
 - Struttura logica dei dispositivi avanzati basati su PLA a PAL

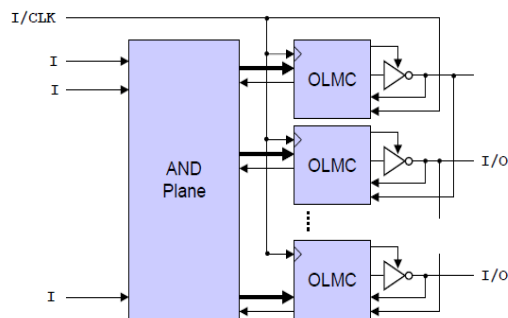


Sistemi embedded
2010/2011

54

Logiche programmabili a più livelli

- Generic Array Logic (GAL)
 - Le GAL sono una generalizzazione delle PAL/PLA
 - Le celle di uscita OLMC o *Output Logic Macro Cell* rendono questa architettura molto flessibile



Sistemi embedded
2010/2011

55

Logiche programmabili a più livelli

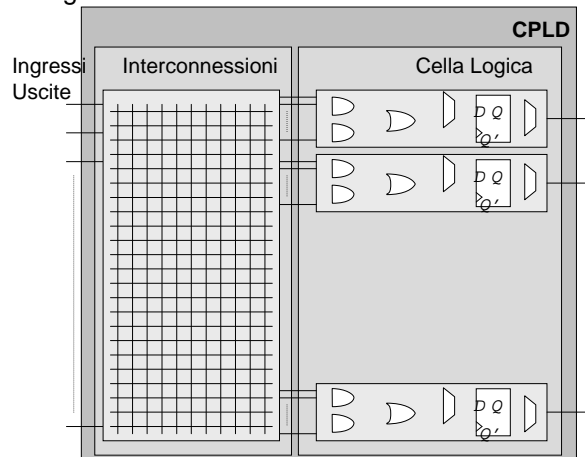
- Complex Programmable Logic Devices
 - I Dispositivi Logici Programmabili Complessi (*Complex Programmable Logic Devices - CPLD*) sono una ulteriore evoluzione di PLA, PAL e GAL
 - Sono caratterizzati da
 - Connessioni globali
 - Logica concentrata
 - Rispetto a PAL, PLA e GAL
 - Offrono una maggiore integrazione
 - Sono costituite da celle più complesse
 - Consentono di ottenere prestazioni più elevate
 - Hanno una struttura più regolare e facilmente programmabile

Sistemi embedded
2010/2011

56

Logiche programmabili a più livelli

- Complex Programmable Logic Devices
 - Architettura generale

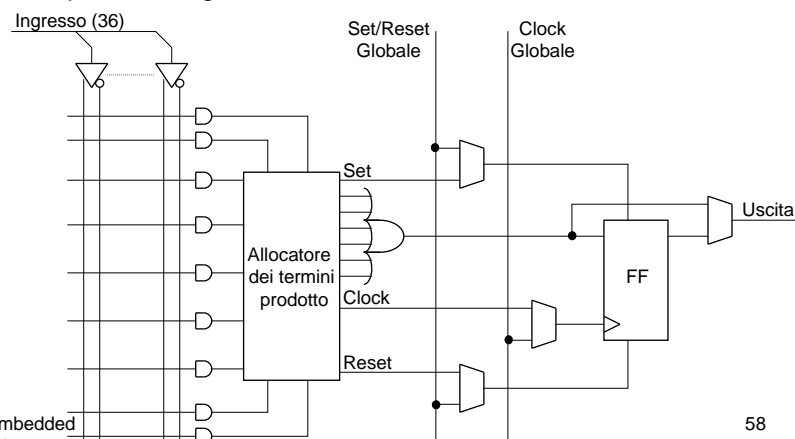


Sistemi embedded
2010/2011

57

Logiche programmabili a più livelli

- Complex Programmable Logic Devices
 - Esempio: cella logica della Xilinx XC9500



Sistemi embedded
2010/2011

58

Tecnologie programmabili

FPGA

(*Field Programmable Gate Arrays*)

FPGA

- Le FPGA sono dispositivi *programmabili* costituiti da un array di *componenti logici* collegabili tra loro
 - Le FPGA mettono a disposizione dell'utente
 - *Componenti logici* (costituiti da porte logiche, Flip-flop, Buffer...) che offrono un insieme di funzionalità che possono essere sfruttate anche solo in parte
 - Molto complessi
 - » Consentono di localizzare funzionalità molto articolate favorendone la velocità ed una ridotta connettività a discapito di un minor sfruttamento
 - Poco complessi
 - » Consentono un maggior grado di sfruttamento (flessibilità) a discapito della elevata connettività

FPGA

- Le FPGA sono dispositivi *programmabili* costituiti da un array di *componenti logici* collegabili tra loro
 - Le FPGA mettono a disposizione dell'utente
 - *Linee di connessione* locali e distribuite (lunghe e corte)
 - Linee che attraversano una parte ridotta del dispositivo e che sono condivise da pochi elementi logici
 - » Ritardi e potenza utilizzata contenuti
 - All'interno di uno stesso dispositivo possono coesistere differenti linee locali di lunghezza differente: elevata flessibilità

FPGA

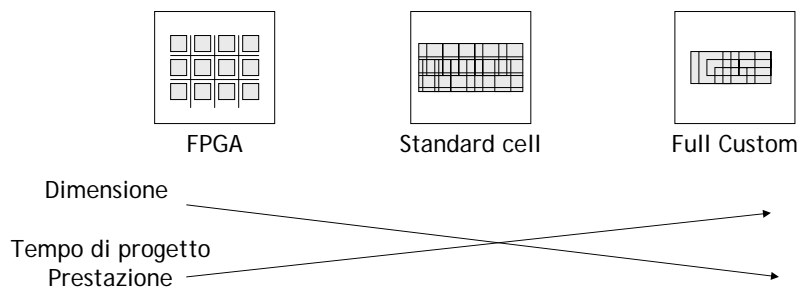
- Aspetti particolarmente interessanti
 - Rappresentano il miglior compromesso nella famiglia dei dispositivi programmabili tra **flessibilità e prestazione**
 - $PLD \Rightarrow FPGA \Leftarrow MPGA$
 - Rappresentano un ottimo compromesso tra **costo e prestazione**
 - FPGA vs. ASIC
 - Rappresentano strumenti per una rapida verifica delle caratteristiche dei dispositivi puramente HW
 - *Fast prototyping*
 - Rappresentano un ottimo compromesso tra **specializzazione e generalità**
 - $SW (GPP, DSP) \Rightarrow FPGA \Leftarrow HW (ASIC)$

FPGA

- Aspetti particolarmente interessanti
 - Consentono di riversare sullo stesso dispositivo, in momenti diversi, configurazioni che descrivono applicazioni differenti o parti di esse
 - Piattaforma Hardware Multi-modale (*off-line*)
 - Le descrizioni sono conservate in ROM e riversate in RAM prima dell'uso
 - Piattaforma Riconfigurabile (*on-line*)
 - Le descrizioni di parte della applicazione sono riversate in RAM, sostituendole ad altre non più in uso, durante il periodo di funzionamento

FPGA

- Tecnologie a confronto



FPGA

- Tecnologie a confronto

	FPGA	Gate array	Standard cell	Full custom
Densità	Basso	Medio	Medio	Alto
Flessibilità	bassa (alta)	Basso	Medio	Alto
Analogico	No	No	No	Si
Prestazioni	Basso	Medio	Alto	Molto alto
Tempo progetto	Basso	Medio	Medio	Alto
Costo progetto	Basso	Medio	Medio	Alto
Tools	Semplice	Complesso	Complesso	Molto complesso
Volume	Basso	Medio	Alto	Alto

Sistemi embedded
2010/2011

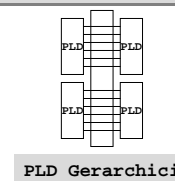
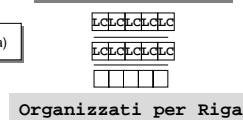
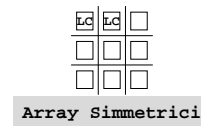
65

FPGA

- Le famiglie di FPGA di contraddistinguono per 3 aspetti

- Architettura generale
 - Array simmetrici (*Xilinx, QuickLogic*)
 - Organizzati per riga (*Actel*)
 - PLD gerarchici (*Altera*)
- Tipi di blocchi logici
 - Basati su Look-up Table (*Xilinx*)
 - Basati su Multiplexer (*Actel*)
 - Blocchi PLD (*Altera*)
- Tecnologia di programmazione
 - RAM statiche (*Xilinx*)
 - EPROM e EEPROM (*Altera*)
 - Anti-Fuse (*Actel, QuickLogic*)

LC: Logic Cell (Cella Logica)



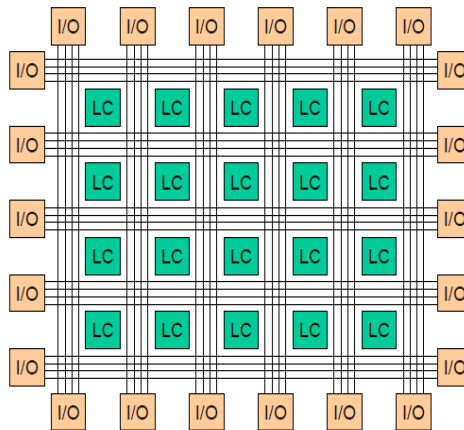
Sistemi embedded
2010/2011

66

FPGA

- Architettura
 - Array simmetrici

LC: Logic Cell (Cella Logica)



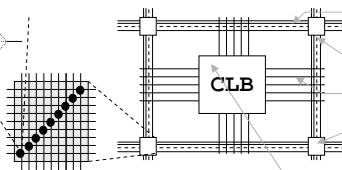
Sistemi embedded
2010/2011

67

FPGA

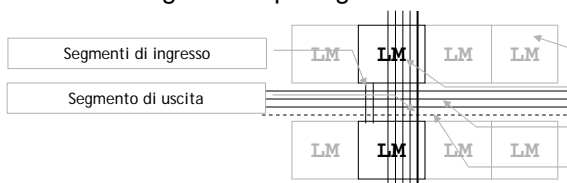
- Architettura
 - Array simmetrici

Ogni punto della
matrice di
interruttori
è costituito da sei
interruttori
indipendenti per
l'instradamento



- Linee di connessione orizzontali
- Linee di connessione verticali
- Linee di connessione
- Matrice di interruttori
- CLB: blocco logico configurabile (configurable logic block)

- Organizzati per riga



- LM: modulo logico (logic module)
- Linee di connessione verticali
- Linee di connessione orizzontali
- Linea di clock

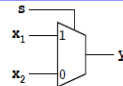
Sistemi embedded
2010/2011

68

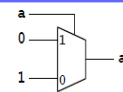
FPGA

- Blocchi logici
 - Elementi base
 - Multiplexer

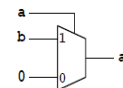
$$y = x_1s + x_2s'$$



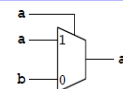
$$\begin{aligned} s &= a \\ x_1 &= 0 \\ x_2 &= 1 \\ y &= (0)a + (1)a' = a' \end{aligned}$$



$$\begin{aligned} s &= a \\ x_1 &= b \\ x_2 &= 0 \\ y &= ba + (0)a' = ab \end{aligned}$$



$$\begin{aligned} s &= a \\ x_1 &= a \\ x_2 &= b \\ y &= aa + ba' = a + a'b \\ &= a + b \end{aligned}$$

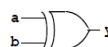
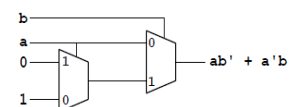
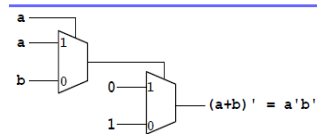
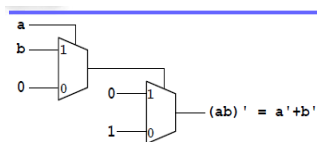


Sistemi embedded
2010/2011

69

FPGA

- Blocchi logici
 - Elementi base
 - Multiplexer

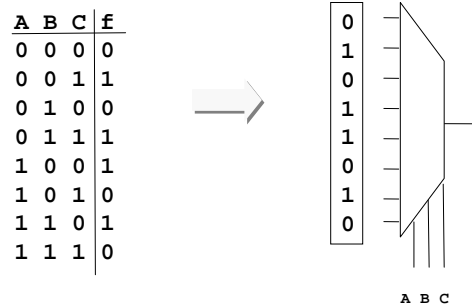


Sistemi embedded
2010/2011

70

FPGA

- Blocchi logici
 - Elementi base
 - Multiplexer a 2^n ingressi: funzione combinatoria di n variabili
 - *Look-up table*

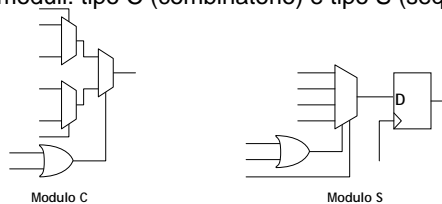


Sistemi embedded
2010/2011

71

FPGA

- Blocchi logici
 - Schema semplificato di un CLB (Xilinx)
 - Schema semplificato di un LM (Altera)
 - 2 tipi di moduli: tipo C (combinatorio) e tipo S (sequenziale)

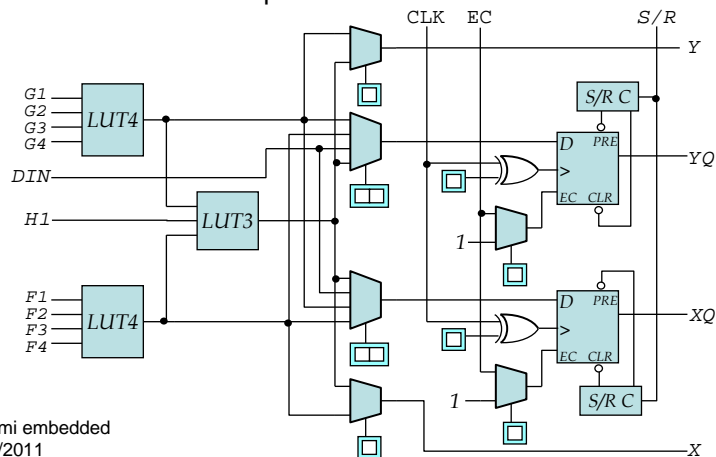


Sistemi embedded
2010/2011

72

FPGA

- Blocchi logici
 - Schema meno semplificato di un CLB

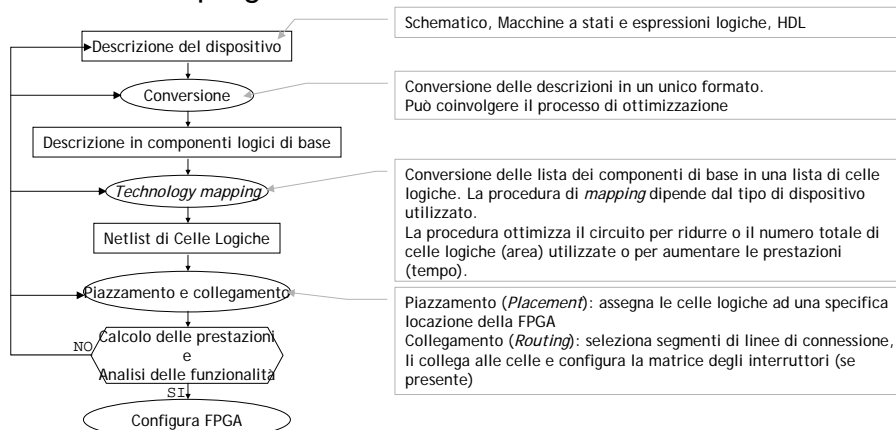


Sistemi embedded
2010/2011

73

FPGA

- Flusso di progetto

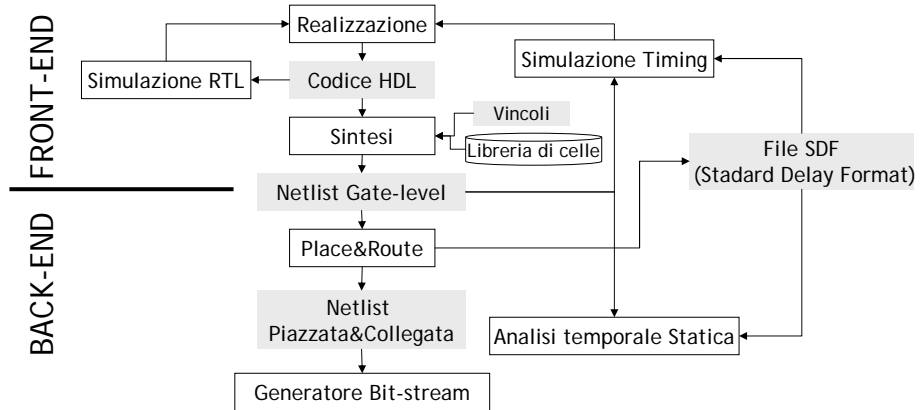


Sistemi embedded
2010/2011

74

FPGA

- Flusso di progetto (bis)



Sistemi embedded
2010/2011

75