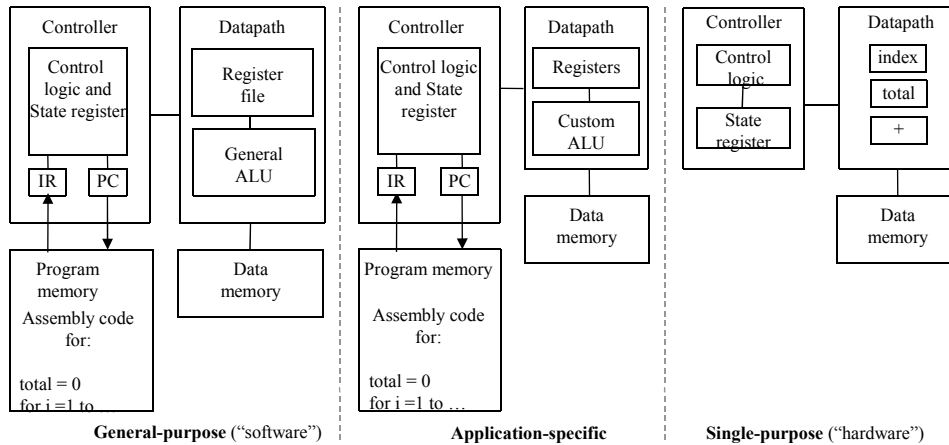

Design technologies

Tre tecnologie chiave nei sistemi embedded

- Tecnologia
 - Un modo di realizzare un task, usando processi, metodi o conoscenza
- Tecnologie per I sistemi embedded
 - Processor technology
 - IC technology
 - Design technology

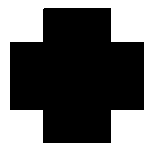
Processor technology

- L'architettura dell'unità di computazione utilizzata per implementare la funzionalità richiesta al sistema
- I processori non devono essere necessariamente programmabili
 - “Processore” non vuol dire “general-purpose processor”



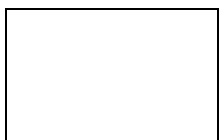
Processor technology

- I processori variano in funzione del problema

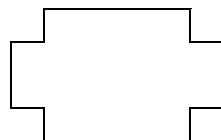


Desired
functionality

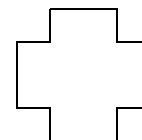
```
total = 0;
for (i = 0; i < N; i++)
    total += M[i];
```



General-purpose
processor



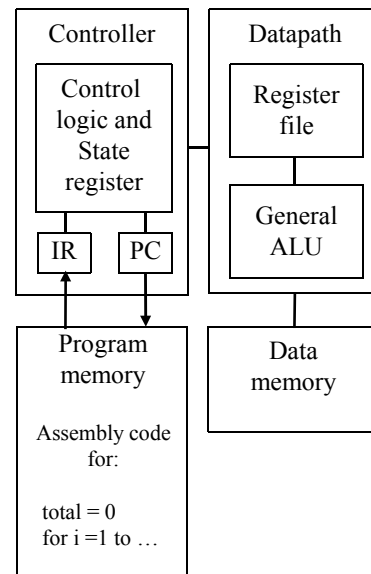
Application-specific
processor



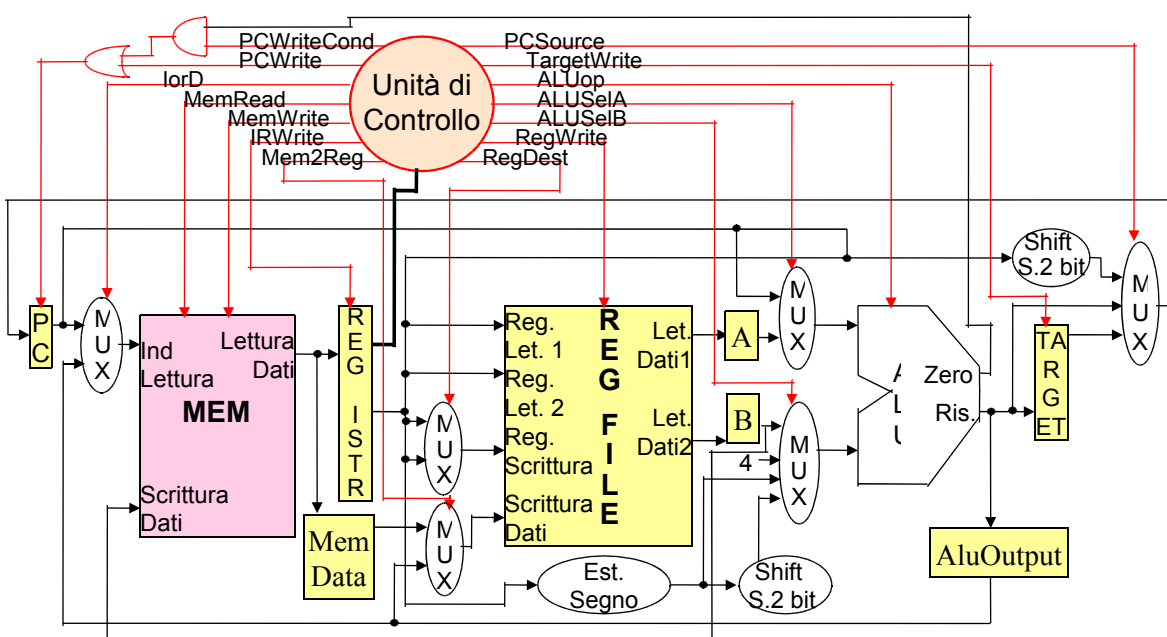
Single-purpose
processor

General-purpose processors

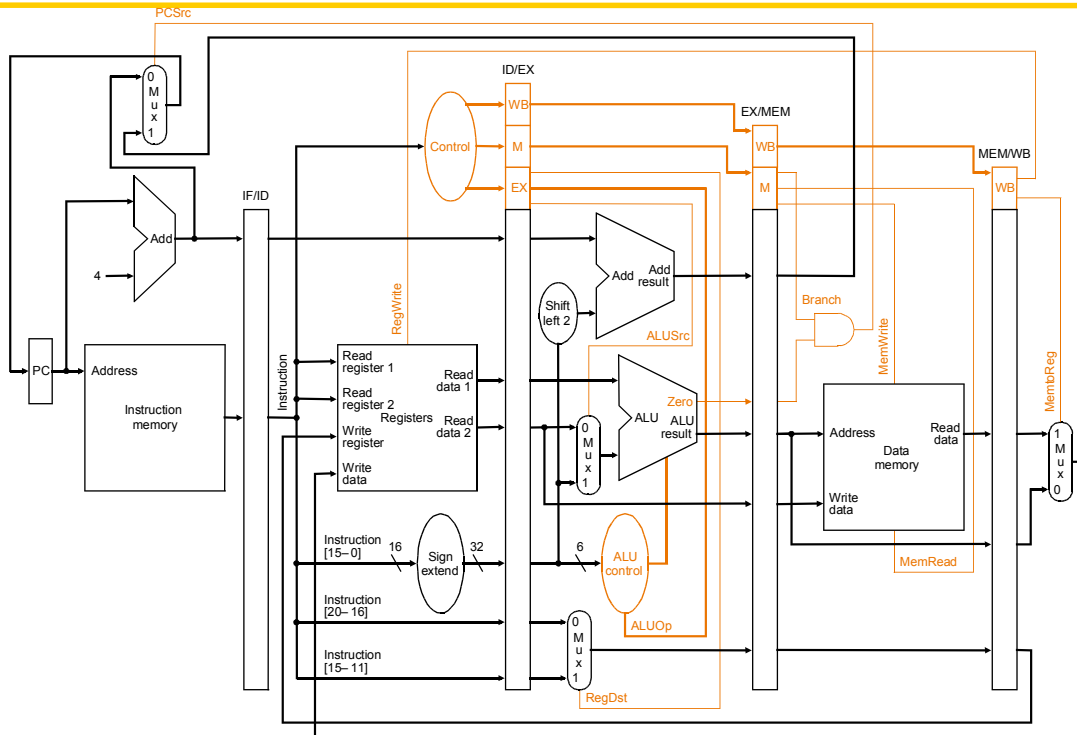
- Dispositivi programmabili usati per un ampio spettro di applicazioni
 - Noti come “microprocessori”
- Caratteristiche
 - Program memory
 - General Datapath con ampio register file e ALU di tipo generale
- Vantaggi
 - Basso time-to-market e NRE costs
 - Alta flessibilità
- Svantaggi
 - Alti costi per unità per grandi volumi
 - Le prestazioni potrebbero essere ridotte



General-purpose processors il DLX sequenziale

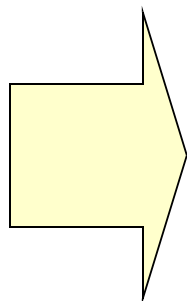


General-purpose processors il DLX pipeline



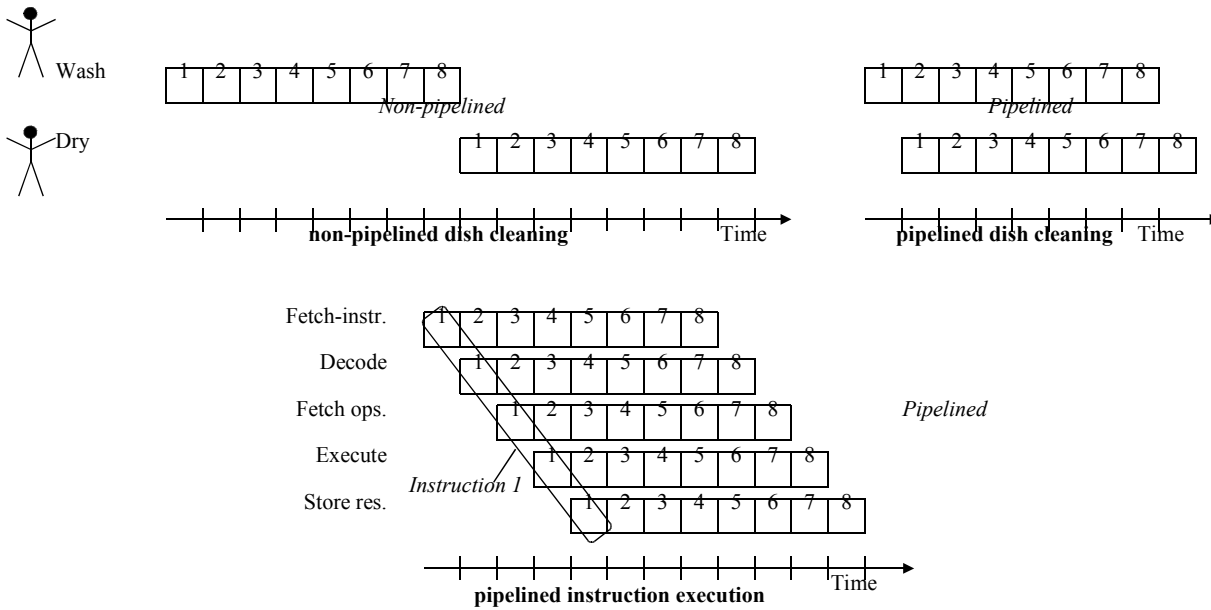
General-purpose processors

```
total = 0;
for (i = 0; i < N; i++)
    total += M[i];
```



```
Loop:  addi r1, r0, 0
      addi r3, r0, 0
      lw r4, M(r1)
      addi r1, r1, 4
      slti r2, r1, 40
      add r3, r3, r4
      bnez r2, loop
```

Pipelining: Increasing Instruction *Throughput*

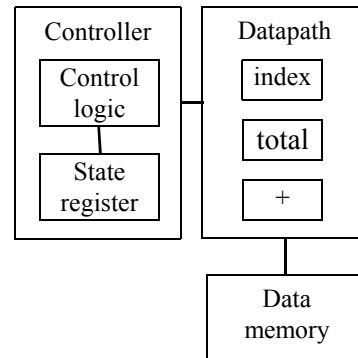


Superscalar and VLIW Architectures

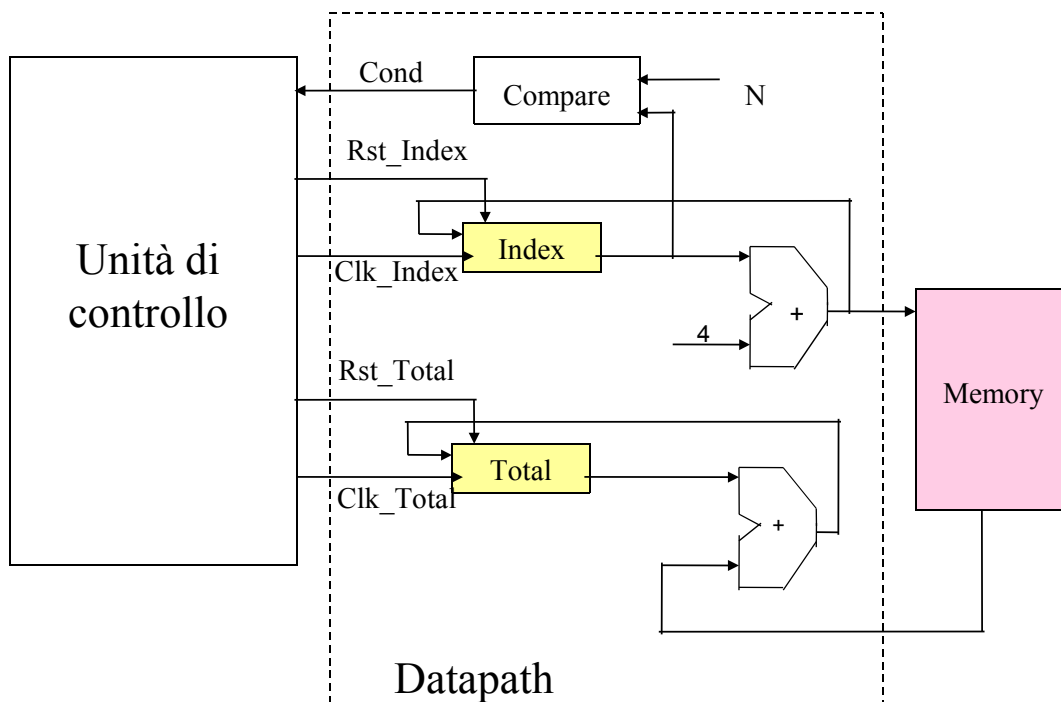
- Performance can be improved by:
 - Faster clock (but there's a limit)
 - Pipelining: slice up instruction into stages, overlap stages
 - *Multiple ALUs* to support more than one instruction stream
 - Superscalar
 - Scalar: non-vector operations
 - Fetches instructions in batches, executes as many as possible
 - May require extensive hardware to detect independent instructions
 - VLIW: each word in memory has multiple independent instructions
 - Relies on the compiler to detect and schedule instructions
 - Currently growing in popularity

Single-purpose processors

- Sistema digitale progettato per eseguire un solo programma
 - Es. coprocessori, acceleratori o periferiche
- Caratteristiche
 - Contengono i soli componenti che servono per eseguire un solo programma
 - Non hanno la program memory
- Vantaggi
 - Veloce, Low power, Piccola dimensione
- Svantaggi
 - Assenza di flessibilità, alto time-to-market, alto costo NRE

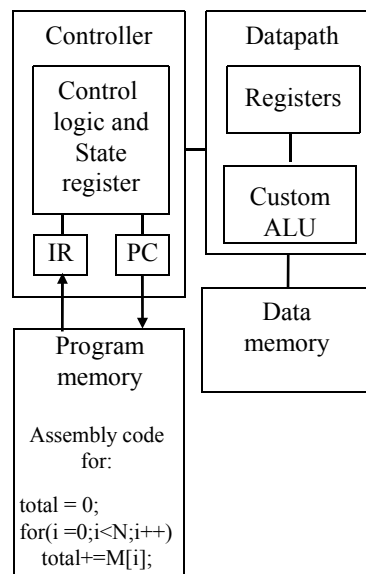


Single-purpose processors



Application-specific processors

- Processori programmabili ottimizzati per una particolare classe di applicazioni
 - Compromesso tra general-purpose e single-purpose processors
- Caratteristiche
 - Program memory, Datapath ottimizzato, Unità funzionali speciali
- Vantaggi
 - Buona flessibilità, buone performance, size e power
- Svantaggi
 - Alto costo NRE (processore e compilatore)



Esempi: Microcontrollori, DSP

A Common ASIP: Microcontroller

- For embedded control applications
 - Reading sensors, setting actuators
 - Mostly dealing with events (bits): data is present, but not in huge amounts
 - e.g., VCR, disk drive, digital camera (assuming SPP for image compression), washing machine, microwave oven
- Microcontroller features
 - On-chip peripherals
 - Timers, analog-digital converters, serial communication, etc.
 - Tightly integrated for programmer, typically part of register space
 - On-chip program and data memory
 - Direct programmer access to many of the chip's pins
 - Specialized instructions for bit-manipulation and other low-level

Digital Signal Processors (DSP)

- For signal processing applications
 - Large amounts of digitized data, often streaming
 - Data transformations must be applied fast
 - e.g., cell-phone voice filter, digital TV, music synthesizer
- DSP features
 - Several instruction execution units
 - Multiple-accumulate single-cycle instruction, other instrs.
 - Efficient vector operations – e.g., add two arrays
 - Vector ALUs, loop buffers, etc.

Microcontrollore: Famiglia ST6

- Microcontrollore a 8 bit



- **Memories**
 - Up to 4 Kbytes of program memory
 - OTP/ROM
 - Up to 64 bytes of RAM
- **I/O Ports**
 - Up to 20 I/O lines
 - Multifunctional, bi-directional I/O pins
 - Up to 4 high current capability I/O line
- **Clock, Reset and Power Supply**
 - Power supply operating range: 3.0V to 6V
 - Maximum external frequency: 8 MHz
 - Oscillator Safeguard (OSG) and Backup oscillator (LFAO)
 - Low Voltage Detector (LVD)
 - 2 power saving modes: WAIT and STOP
- **Interrupts**
 - 4 interrupt vectors plus NMI and RESET
 - Software programmable for each I/O
- **I/O Ports**
 - Up to 20 I/O lines
 - Multifunctional, bi-directional I/O pins
 - Up to 4 high current capability I/O line
- **Peripherals**
 - Watchdog timer
 - 8-bit timer
 - ADC
- **Instruction Set**
 - 8-bit accumulator-based architecture
 - 40 instructions
 - 9 addressing modes

Microcontrollore: STR7(ARM7TDMI® core)

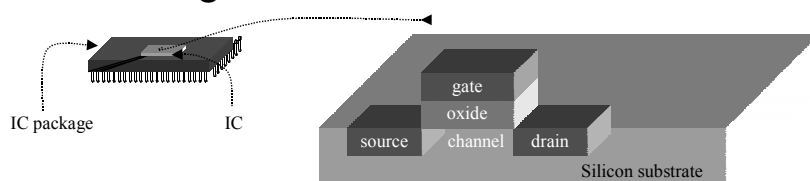
- STR710F Flash Microcontrollers from STMicroelectronics combine the industry standard ARM7TDMI® RISC microprocessor with embedded Flash and powerful peripheral functions including, USB and CAN.

La famiglia ARM7

- **L'ARM7** è un processore RISC a 16/32 bit, prodotto, a partire dal 1993, dalla “Advanced Risc Machines Ltd.”
- È disponibile in una **ampia gamma di versioni**, dedicate principalmente ad applicazioni di tipo “**embedded**”.
- Tra queste, ci sono **numerosi mC prodotti da** almeno 10 diversi costruttori tra i quali STM, Motorola, NEC, Atmel, Philips, Sharp, TI.
- Si tratta quindi di un vero e proprio **standard** industriale.

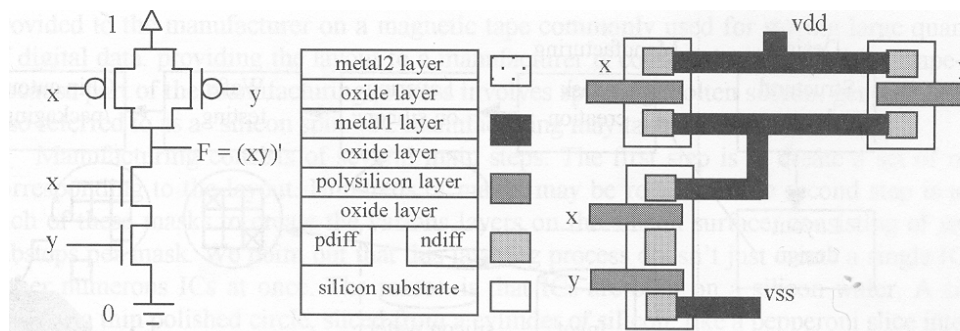
Integrated circuit (IC) technology

- Il modo in cui una implementazione digitale (gate-level) è mappata su un IC
 - IC: Integrated circuit, o “chip”
 - IC sono costituiti da diversi layer
 - I layer più bassi formano i transistor
 - Quelli intermedi formano i componenti logici
 - Quelli più alti collegano i componenti con i wires
 - Le tecnologie differiscono nel livello di specializzazione rispetto a un design



IC technology

Porta NAND



IC technology

- Tre tipi di IC technologies
 - Full-custom/VLSI
 - Semi-custom ASIC (gate array and standard cell)
 - PLD (Programmable Logic Device)

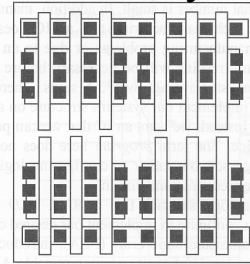
Full-custom/VLSI

- Tutti i layer sono ottimizzati per una particolare implementazione del sistema embedded
 - Placing transistors
 - Sizing transistors
 - Routing wires
- Benefici
 - Eccellenti performance, piccola dimensione, ridotta potenza
- Svantaggi
 - Elevato NRE, lungo time-to-market

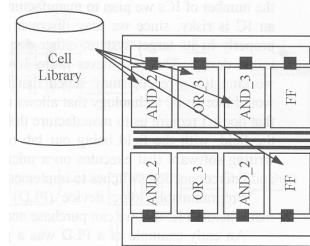
Semi-custom

- I layer più bassi sono in tutto o in parte realizzati
 - I designers devono occuparsi dei livelli più alti
- Benefici
 - Buone performance, buona dimensione, minore NRE rispetto a una implementazione full-custom
- Svantaggi
 - Richiede ancora settimane o mesi per lo sviluppo

Gate array



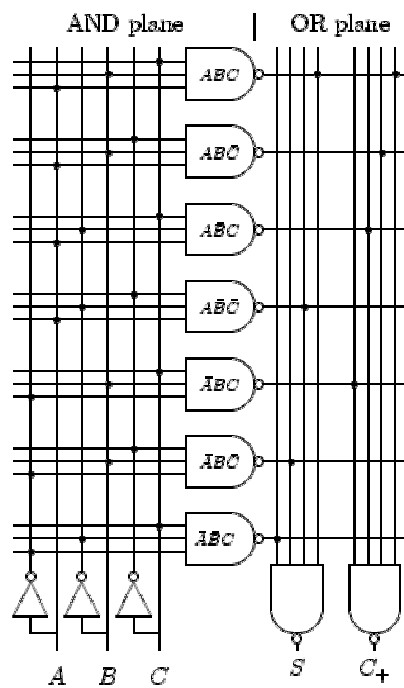
Standard Cell



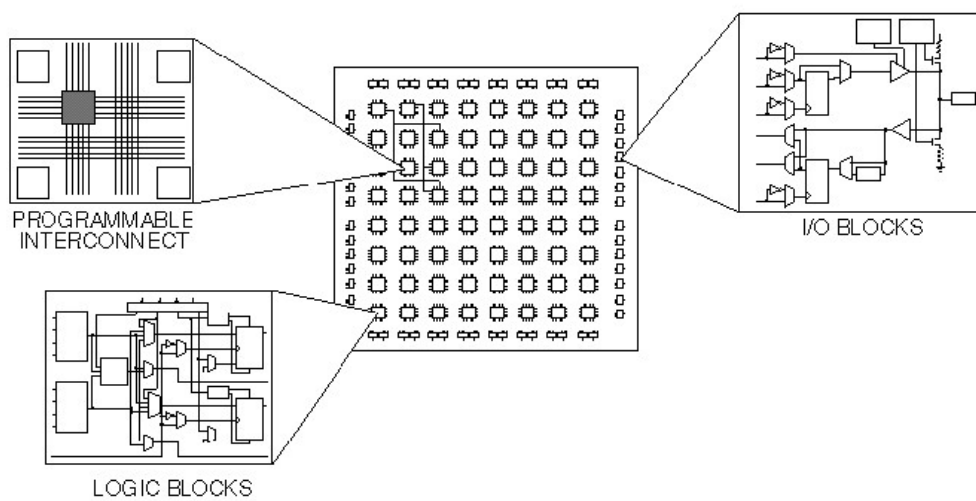
PLD (Programmable Logic Device)

- Tutti i layers già esistono
 - I designers possono comprare un IC
 - Le connessioni sull' IC sono o create o distrutte per implementare la funzionalità richiesta
 - Due tipi:
 - Programmable Logic Array (PLA)
 - Field-Programmable Gate Array (FPGA)
- Benefici
 - Basso NRE, la disponibilità dell'IC è quasi immediata
- Svantaggi
 - Più grande, più costoso, consuma molto, più lento

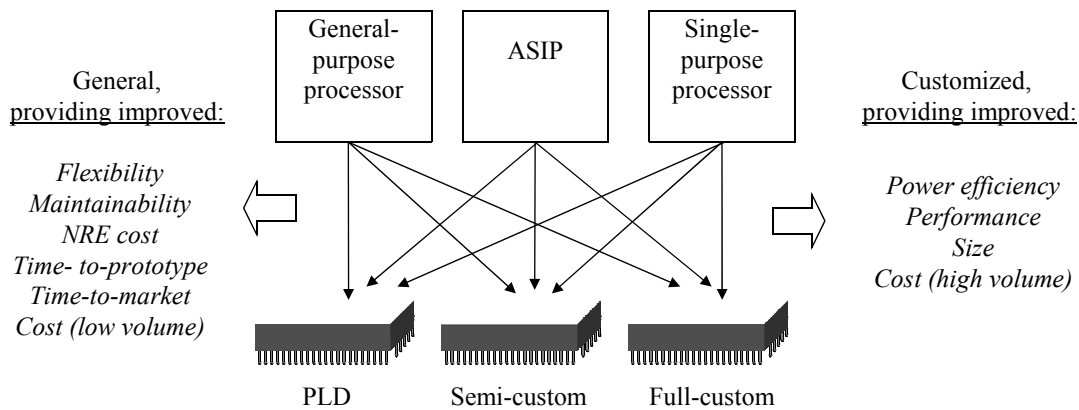
Programmable Logic Array (PLA)



Xilinx FPGA



Indipendenza tra processore e IC technologies



Legge di Moore

- Il più importante trend nei embedded systems
 - Predetto in 1965 dal cofondatore della Intel, Gordon Moore
 - La capacità dei transistor raddoppierà ogni 18 mesi per le prossime decadi**

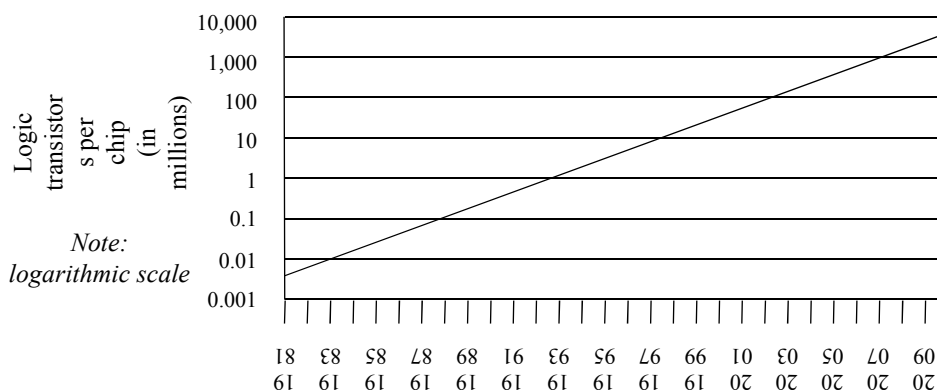
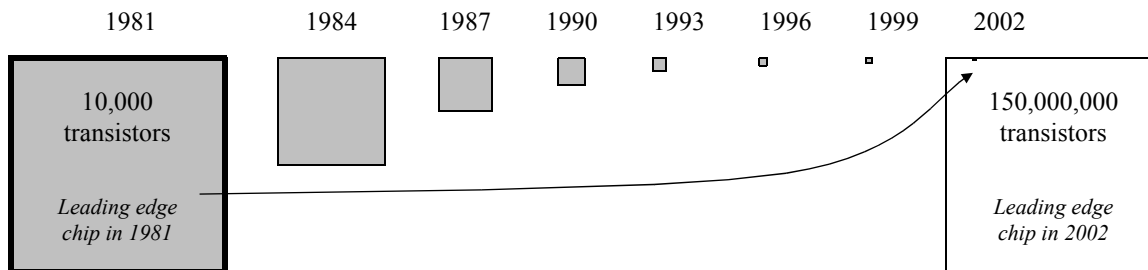
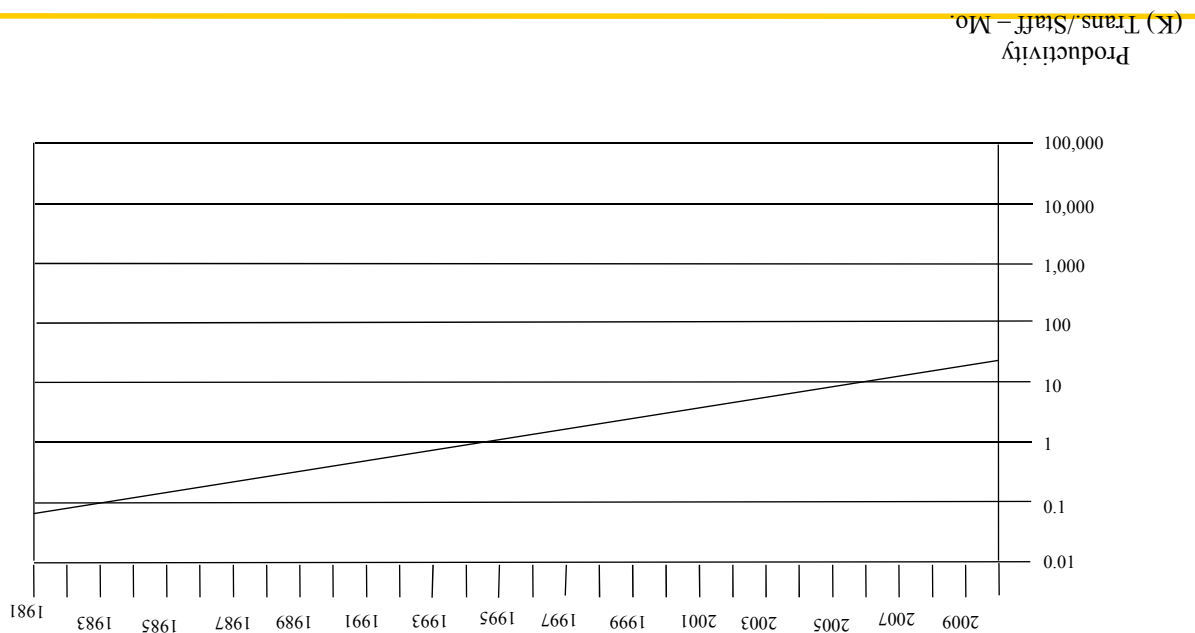


Illustrazione grafica della legge di Moore

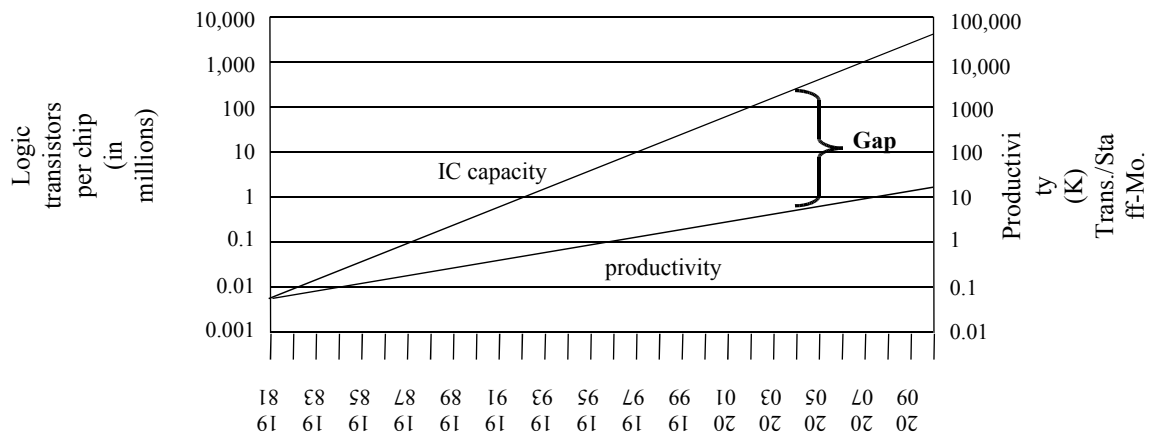


La produttività aumenta in modo esponenziale



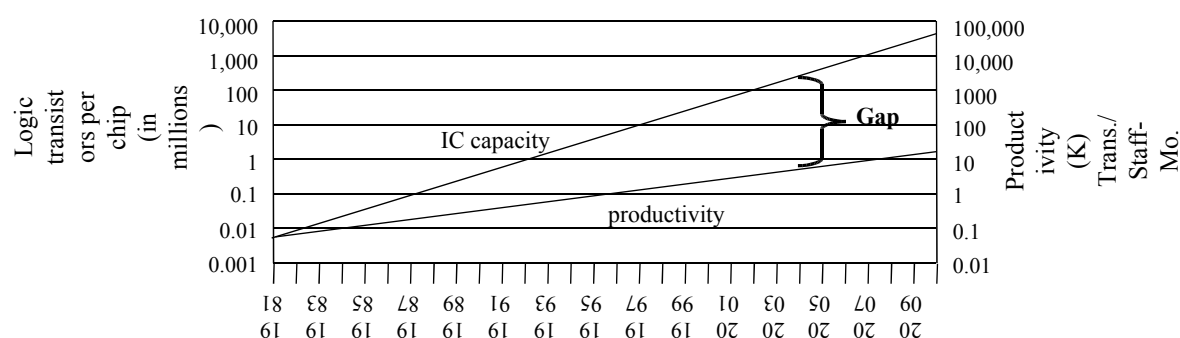
Design productivity gap

- Sebbene la capacità produttiva dei progettisti sia aumentata in modo impressionante, non ha tenuto il passo dell'incremento della capacità dei chip



Design productivity gap

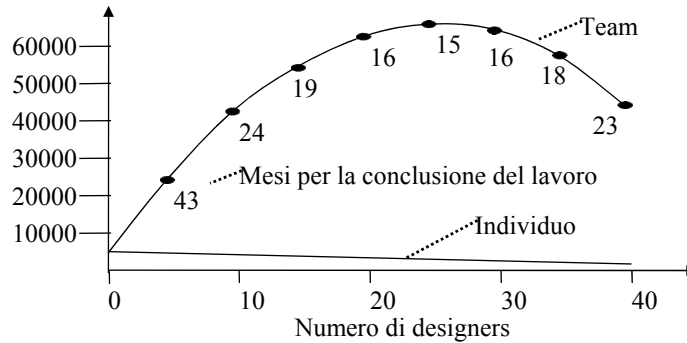
- 1981 Il chip più avanzato richiedeva 100 mesi uomo
 - 10,000 transistors / 100 transistors/mese
- 2002 Il chip più avanzato richiedeva 30,000 mesi uomo
 - 150,000,000 / 5000 transistors/mese



Il mese-uomo

- La situazione è peggiore di ciò che mostra il productivity gap
- In teoria l'aumento del numero di progettisti dovrebbe ridurre i tempi di completamento di un progetto. In realtà, la produttività del singolo progettista diminuisce a causa della complessità nella gestione e comunicazione nel gruppo di lavoro.
- In alcune casi, un incremento di persone potrebbe allungare il tempo di realizzazione di un progetto.

- 1M transistors, 1 designer=5000 trans/mese
- Ogni designer in più riduce di 100 trans/mese
- 2 designer producono 4900 trans/mese ciascuno



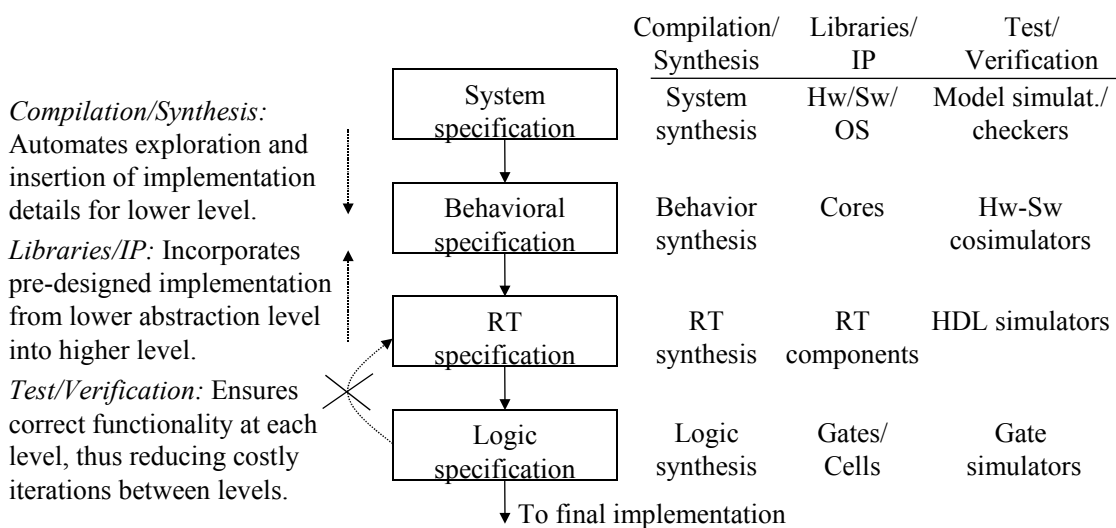
Gestione della crisi della produttività

- IP (Intellectual Property) Reuse
 - Assemblaggio di componenti pre-progettati
 - Componenti proprietari o provenienti dall'esterno
 - Soft e Hard IPs
- System-Level Design
 - Progetto e verifica a livello di sistema anziché a livello RTL o di gate

Design Technology

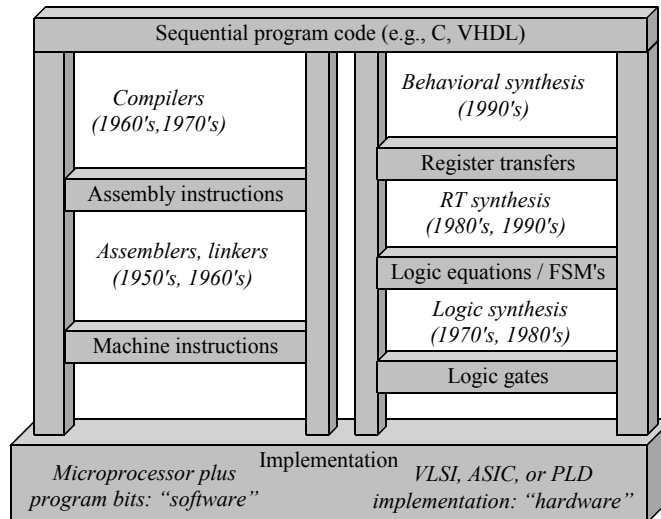
- E' il modo in cui si traduce la funzionalità richiesta del sistema in una sua implementazione
- Una procedura per progettare un sistema
- Il flusso di progettazione può essere parzialmente o totalmente automatizzato mediante un insieme di tools.
 - Software engineering tools,
 - Compilers,
 - Computer-Aided Design tools,
- Un'adeguata metodologia di progettazione aiuta la gestione del processo di design e migliora la qualità, le performance e i costi del progetto.

Design Technology



Il co-design

- In passato:
 - Hardware e software design technologies erano molto differenti
 - La recente maturazione della sintesi permette di avere una visione unificata di hardware e software
- Hardware/software “codesign”



La scelta tra hardware e software per una particolare funzione è semplicemente un tradeoff tra varie design metrics, come performance, power, size, NRE cost, e flessibilità; non c'è una differenza sostanziale tra cosa l'hardware o il software può implementare.

Riferimenti

- “*Embedded System Design: A Unified Hardware/Software Introduction*”, Frank Vahid, Tony Givargis, John Wiley & Sons Inc., ISBN:0-471-38678-2, 2002.
- “*Computers as Components: Principles of Embedded Computer Systems Design (With CD-ROM)*”, Wayne Wolf, Morgan Kaufmann Publishers, ISBN: 1-55860-541-X, 2001
- “*Embedded System Design*” by Peter Marwedel, Kluwer Academic Publishers, ISBN: 1-4020-7690-8, October 2003