**RIPASSO ARCHITETTURA DEGLI ELABORATORI**

* 1. **EQUIVALENZE**

KB = Kbps =

MB = Mbps =

GB = Gbps =

TB = Tbps =

* 1. **LEGGE DI MOORE**

#Transistor quadruplica in una memoria (a parità di capacità) ogni 3 anni

Densità =

* 1. **NUMERO TRANSISTOR**

NOR = 2 AND = NAND + NOT = 3

NAND = 2 (se a 2 ingressi, altrimenti 3 o +) OR = NOR + NOT = 3

NOT = 1

* 1. **COMPOSIZIONE FUNZIONI**

NOT con : - NAND:

- NOR:

AND con : - NAND :

- NOR :

NOR con : - NOR :

- NAND :

* 1. **ALGEBRA BOOLEANA**

Distributività: *A + BC = (A + B)\*(A + C)*

Assorbimento (1): *A + AB = A 🡪 A(B+1) = A*

Assorbimento (2) *: A + (NOT A)B = A + B 🡪 A + B(A + (NOT A)) = A + AB + (NOT A) B = A(B + 1) + (NOT A) B*

* 1. **RAM**

SRAM : usate per gli strati di memoria CACHE, 6 transistor per bit;

DRAM : usate per gli strati di memoria RAM e sono più capienti e lente

* 1. **SOMMA**

In C1 la somma se da OVERFLOW, si risommano i bit in eccesso ai bit meno significativi

In C2 la somma da OVERFLOW sse i due addendi hanno lo STESSO SEGNO, altrimenti si scarta l’eccesso

* 1. **CODIFICA DEI CARATTERI**

ASCII: 7-8 bit;

UNICODE : 2 o più byte;

UTF-8 : estensione, usa fino 4 byte a carattere

* 1. **CODICI CORRETTORI**

CODICI DI PARITA’: 1 bit per inserire un bit di CONTROLLO e posso avere parità pari/dispari🡪Dist.Hamm.= 2;

CODICI DI HAMM: #bit = 🡪 #bit controllo =

COSTO DEL CODICE =

Con DISTANZA DI HAMMING: RILEVA bit e CORREGGE bit

**2.0 REGISTRI**

Program Counter : **PC** **(R15)**

Instruction Register : **IR**

Link Register : **LR (R 14)**

Stack Pointer : **SP (R13)**

**2.1 DIPENDENZE TRA REGISTRI**

RAW(READ after WRITE)

WAW (WRITE after WRITE)

WAR (WRITE after READ)

**2.3 SHIFT (RANGE da 0 a 31, con 5 bit)**

LOGICAL SHIFT : LEFT/RIGHT e inserisce 0 come nuovo bit;

ARITHMETIC SHIFT RIGHT: inserisce a sx un uguale al bit di SEGNO;

ROTATE RIGHT : i bit eliminati a dx rientrano a sx.

**2.4 BRANCH**

Il BRANCH, nel caso di istruzioni da 32 bit con OPCODE da 8 bit, ha un totale di (32 – 8 = 24 bit) e

considerando che i salti sono in avanti/indietro (1 bit di segno) 🡪 RANGE =

**2.5 MECCANISMI DI SINCRONIZZAZIONE**

I/O PROGRAMMATO: La CPU controlla periodicamente lo stato di ogni periferica;

I/O CONTROLLATO: La CPU riceve un interrupt da un controllore e decide se occuparsene;

DMA

**2.6 BANDA, SKEW**

BANDA = ]

SKEW è il periodo limite entro cui puoi risincronizzare due linee adiacenti 🡪 T SKEW

**2.7 CACHE**

Rapporto CACHE/HIT 🡪 Tempo di ACCESSO =

**2.8 LEGGE DI AHMDAL (PARALLELISMO)**

INCREMENTO DI PRESTAZIONI (i): dove (1 – f) è la parte sequenziale

Se f = 0 🡪 Situazione ottimale MASSIMO GUADAGNO

Se f = 1 🡪 Situazione peggiore NESSUN GUADAGNO

🡪 se i = n il guadagno è massimo