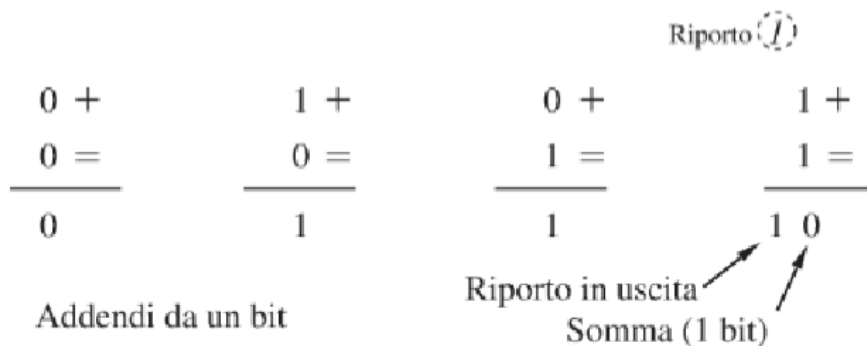


6 Circuiti Aritmetici

Per SOMMARE numeri binari ad 1 bit:



IL RIPORTO IN USCITA della cifra precedente viene assegnato come RIPORTO IN ENTRATA alla successiva

Un addizionatore a 2 bit può essere espresso come 2 funzioni logiche a 3 ingressi (2 bit da sommare e 1 riporto) in cui la prima, somma i 2 bit e la seconda calcola il riporto

abbiamo diversi tipi di addizionatori:

HALF ADDER

addizionatore a 1 bit è un circuito logico che somma 2 bit (A e B) senza considerare un riporto in ingresso, produce 2 uscite (il valore della somma e il bit di riporto).

calcola il riporto ma non lo considera nelle operazioni

E' formato da una porta XOR e una AND

Formule logiche:

- La somma S è calcolata utilizzando una porta XOR: $S = A \oplus B$
- Il riporto C è calcolato utilizzando una porta AND: $C = A \cdot B$

FULL ADDER

è un circuito logico che somma 3 bit, 2 in ingresso e un riporto in ingresso, produce 2 uscite, una somma e un riporto in uscita (che viene passato all'addizionatore successivo)

ADDIZIONATORE A PROPAGAZIONE DI RIPORTO (CLA (CARRY LOOKHAED ADDER))

È un addizionatore avanzato progettato per migliorare la velocità della somma, riducendo i tempi di propagazione del riporto.

Negli addizionatori tradizionali il riporto viene calcolato in modo sequenziale, il che rallenta l'operazione se ci sono molti bit da sommare.

Cosa fa?

Calcola i riporti in modo più rapido rispetto agli addizionatori classici.

Come fa?

Utilizza due concetti chiave:

1. **Generazione del Riporto (G)**: il riporto viene generato quando entrambi gli addendi (A e B) sono 1: $G=A \cdot B$
2. **Propagazione del Riporto (P)**: si verifica quando il riporto deve essere sommato ai due valori di ingresso e almeno uno dei due è 1: $P=A \oplus B$

Grazie a questi segnali, è possibile calcolare i riporti **in modo parallelo anziché sequenziale**, migliorando così le performance dell'addizionatore.

Vantaggio

La velocità: il calcolo parallelo riduce il tempo necessario per ottenere i risultati.

Come si realizza?

Si utilizza una rete di porte logiche (AND, OR, XOR) che calcola rapidamente i segnali P e G e determina il riporto per ogni bit della somma. Questo circuito viene utilizzato in sistemi ad alte prestazioni, come le **ALU dei moderni processori**.

TRABOCCO (Overflow)

Il trabocco si verifica quando il risultato di un'operazione aritmetica supera la capacità di rappresentazione del circuito aritmetico.

Un esempio intuitivo è un bicchiere che si riempie con troppa acqua: l'acqua trabocca. Analogamente, in un circuito digitale, si verifica un trabocco quando il riporto supera il massimo numero di bit gestibili.

Ad esempio:

- Se si sta lavorando con un registro a 4 bit e il riporto supera il bit più significativo (MSB), si ha un trabocco.
 - Questo accade tipicamente con numeri rappresentati in complemento a 2, soprattutto quando si sommano numeri con segno.
-

Come si gestisce?

Addizionatore Algebrico a n bit

Un **addizionatore algebrico a n bit** è un circuito logico che somma due numeri binari a n bit, considerando anche il segno, rappresentato in **complemento a 2**. Questo circuito consente di eseguire l'addizione di numeri interi con segno.

Caratteristiche principali:

1. **Gestione del Riporto:** Calcola il carry out (riporto) per ogni bit.
2. **Gestione del Segno:** Interpreta i numeri secondo la codifica in complemento a 2, distinguendo tra numeri positivi e negativi.
3. **Gestione del Trabocco:**
 - Il riporto finale (Carry) può essere analizzato per determinare se si è verificato un trabocco.
 - Nel complemento a 2, il trabocco si verifica quando il bit del segno cambia in modo inatteso (ad esempio, il risultato di una somma tra due numeri positivi risulta negativo).

GESTIONE DEI RITARDI CON IL RIPPLE CARRY ADDER (RCA)

Il **Ripple Carry Adder (RCA)** è un tipo di addizionatore utilizzato nei circuiti digitali per sommare due numeri binari a più bit. Funziona concatenando più **addizionatori a 1 bit** (half adder o full adder), che sommano i bit uno alla volta procedendo dal meno significativo (LSB) al più significativo (MSB).

Il termine "ripple" indica che il **riporto** si propaga sequenzialmente da un bit al successivo, come un'onda.

Ritardo

Il ritardo in un RCA è causato dalla **propagazione sequenziale del riporto** attraverso tutti i bit successivi. Questo introduce una **dipendenza dal dato**, poiché ogni bit deve aspettare il calcolo del riporto precedente prima di completare la sua somma. Il ritardo complessivo cresce in modo **lineare** con il numero di bit da sommare.

Vantaggi e Svantaggi

- **Vantaggi:**
 - **Semplicità di realizzazione:** il design del circuito è facile da implementare e richiede poche risorse hardware.
 - **Svantaggi:**
 - **Ritardo:** il tempo di propagazione del riporto limita la velocità di calcolo, rendendo l'RCA inadatto a operazioni ad alte prestazioni su numeri con molti bit.
-

In sintesi

Il Ripple Carry Adder è un tipo di addizionatore a più bit in cui i riporti si propagano sequenzialmente attraverso tutti i bit successivi, causando ritardi proporzionali alla lunghezza dei numeri da sommare. È semplice da costruire, ma il ritardo crescente lo rende meno efficiente rispetto ad addizionatori più avanzati.

G e P

Generazione del riporto determina la creazione del riporto quando entrambi i bit (x_i e y_i) sono uguali a 1. In altre parole, il riporto è generato se $x_i=1$ e $y_i=1$

- **Funzione di generazione:** $G_i = x_i \cdot y_i$
- **Funzione di propagazione:** $P_i = x_i + y_i$

Propagazione determina se il riporto in ingresso (c_i) verrà propagato al bit successivo. Se il riporto di un bit dipende da quello precedente, la propagazione garantisce che il

riporto venga considerato nei bit successivi.

Il calcolo del riporto può essere **parallelizzato**, il che permette di svincolarsi dalla sequenzialità del riporto e migliorare la velocità dell'operazione.

Come si fa? Per parallelizzare il calcolo del riporto, si utilizzano tecniche che riducono il ritardo associato alla propagazione sequenziale del riporto, come gli **addizionatori a propagazione di riporto (Carry Lookahead Adders - CLA)**. Queste tecniche permettono di calcolare il riporto in parallelo per tutti i bit, riducendo significativamente il tempo di esecuzione.

CELLA DI STADIO DA UN BIT (FULL ADDER BIT CELL)

È un componente fondamentale in un addizionatore completo a n bit (è quello che esegue la somma).

Come funziona?

Ogni stadio dell'addizionatore a n bit si occupa di sommare 2 bit e 1 riporto in ingresso, fornendo un eventuale riporto in uscita.

Una cella di stadio da un bit riceve 3 input:

- **X_i** : il primo bit
- **Y_i** : il secondo bit
- **C_i** : il carry-in (riporto della somma precedente).

ADDIZIONATORE CON ANTICIPO A 2 LIVELLI (versione ottimizzata di CLA)

Per ridurre ulteriormente il tempo di calcolo, si utilizza una struttura a più livelli logici. In questo schema, i riporti vengono calcolati in parallelo su due livelli distinti, diminuendo la profondità della rete di calcolo, ovvero il numero di stadi che il calcolo deve attraversare.

La struttura suddivide i bit in gruppi, calcolando i riporti a un livello superiore. In pratica, si prende la struttura di un CLA e la si estende su due livelli per ottimizzare ulteriormente la velocità.

ADDIZIONATORE CON ANTICIPO DI RIPORTO A 16 BIT

È un tipo di addizionatore che utilizza un meccanismo per calcolare i riporti in modo più efficiente rispetto al Ripple Carry Adder (RCA), al fine di ridurre i tempi necessari per il calcolo del riporto. In questo schema, vengono calcolati in anticipo tutti i riporti necessari per i 16 bit disponibili. Anche questo addizionatore fa uso dei segnali di *Generazione del Riporto* (G) e *Propagazione del Riporto* (P).

MOLTIPLICATORE SEQUENZIALE

È un tipo di moltiplicatore che esegue la moltiplicazione in più fasi utilizzando un algoritmo iterativo (un approccio top-down). Invece di calcolare il risultato della moltiplicazione in un unico ciclo, sfrutta registri per memorizzare i risultati parziali.

Rispetto al moltiplicatore parallelo, presenta il vantaggio di una ridotta complessità hardware (è meno costoso). Tuttavia, le performance del moltiplicatore parallelo sono superiori.

Questi moltiplicatori sequenziali si trovano prevalentemente nei vecchi processori o nelle CPU a basso costo.

Come è fatto?

Circuito moltiplicatore sequenziale



Algoritmo:

$A = 0$

$Q = \text{Moltiplicatore}$

$M = \text{Moltiplicando}$

Per n cicli:

se $q_0 = 1$:

$A = A + M$

altrimenti:

$A = A + 0$

$c = \text{riporto}$

$c \parallel A \parallel Q$ scorrono verso destra di una posizione

Dopo n cicli i due registri $A \parallel Q$ concatenati conterranno il **Prodotto finale**

