

## Lecture 6

### Multioperand Addition

---

---

---

---

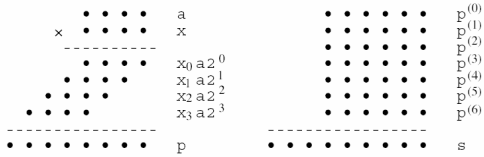
---

---

---

---

#### Applications of multioperand addition



**Multiplication**

$$p = a \cdot x$$

**Inner product**

$$s = \sum_{i=0}^{n-1} x^{(i)} y^{(i)} = \sum_{i=0}^{n-1} p^{(i)}$$

---

---

---

---

---

---

---

---

#### Number of bits of the result

$$S = \sum_{i=0}^{n-1} x^{(i)} \quad x^{(i)} \in [0..2^k-1]$$

$$S_{\min} = 0 \quad S_{\max} = n(2^k-1)$$

$$\begin{aligned} \# \text{ of bits of } S &= \lceil \log_2 (S_{\max} + 1) \rceil = \\ &= \lceil \log_2 (n(2^k-1) + 1) \rceil \leq \lceil \log_2 n 2^k \rceil = \\ &= k + \lceil \log_2 n \rceil \end{aligned}$$

---

---

---

---

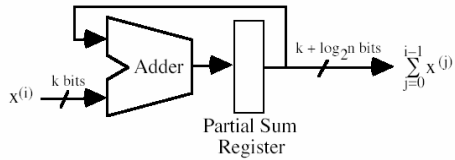
---

---

---

---

**Serial implementation of multioperand addition**




---

---

---

---

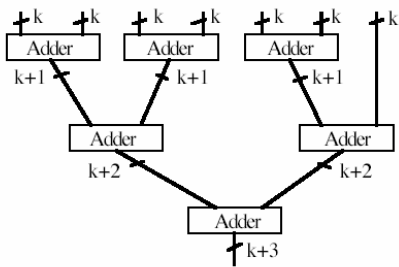
---

---

---

---

**Adding 7 numbers in the binary tree of adders**




---

---

---

---

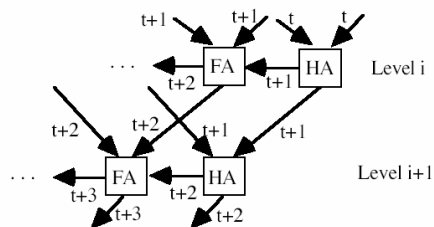
---

---

---

---

**Ripple-carry adders at levels i and i+1**




---

---

---

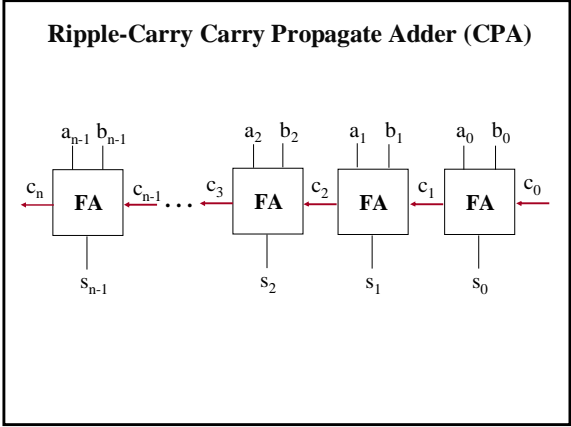
---

---

---

---

---




---

---

---

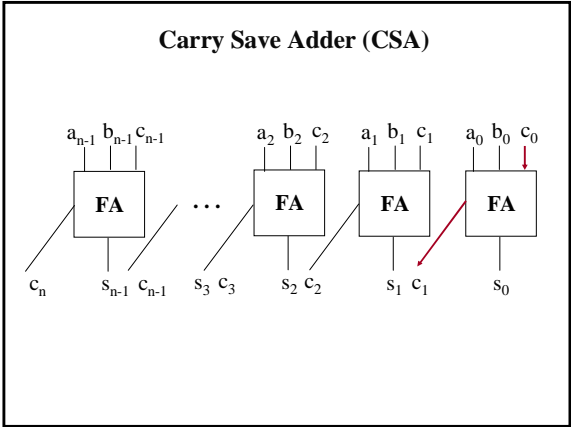
---

---

---

---

---




---

---

---

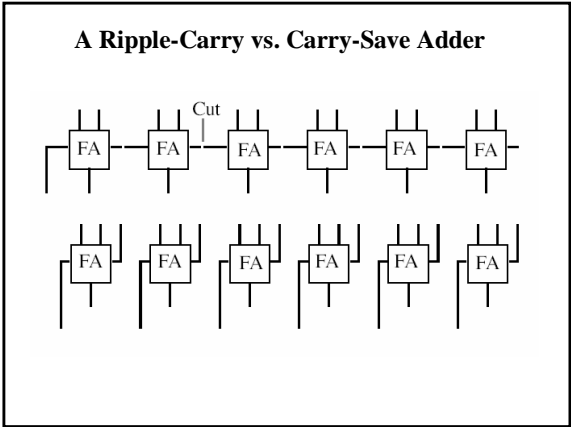
---

---

---

---

---




---

---

---

---

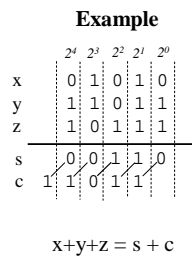
---

---

---

---

### Operation of a Carry Save Adder (CSA)




---

---

---

---

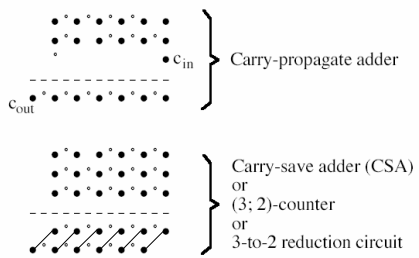
---

---

---

---

### Carry propagate and carry-save adders in dot notation




---

---

---

---

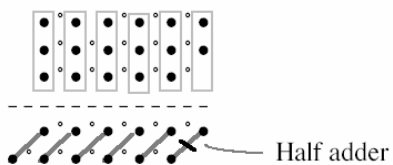
---

---

---

---

### Specifying full- and half-adder blocks in dot notation




---

---

---

---

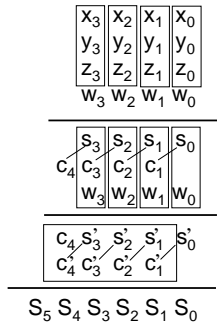
---

---

---

---

**Carry-save adder for four operands**




---

---

---

---

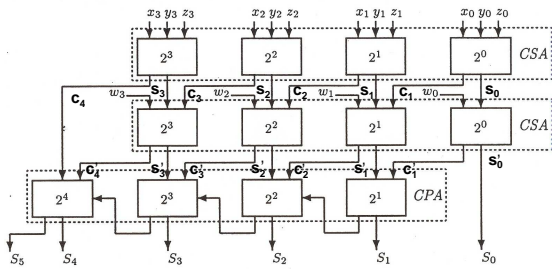
---

---

---

---

**Carry-save adder for four operands**




---

---

---

---

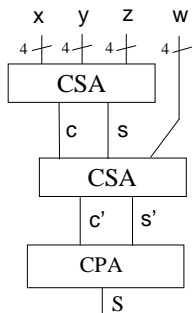
---

---

---

---

**Carry-save adder for four operands**




---

---

---

---

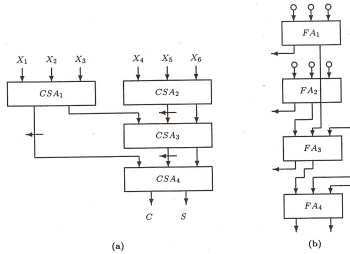
---

---

---

---

### Carry-save adder for six operands



(a) CSA tree

(b) Implementation of one-bit slice

---

---

---

---

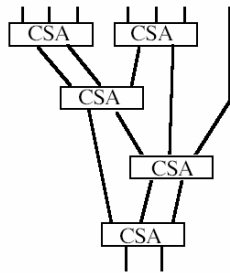
---

---

---

---

### Tree of carry save adders reducing seven numbers to two




---

---

---

---

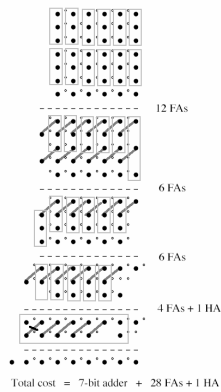
---

---

---

---

### Addition of seven six-bit numbers in dot notation




---

---

---

---

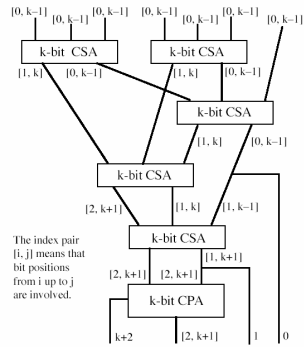
---

---

---

---

**Adding seven  
k-bit numbers:  
block diagram**




---

---

---

---

---

---

---

---

**Parameters of tree carry-save adders (1)**

**Latency**

$$\text{Latency}_{\text{CSA}} = h(n) \cdot T_{\text{FA}} + \text{Latency}_{\text{CPA}}(k, n)$$

Tree height  
for  $n$  operands

Component Adders	Widths
CSA	$k \dots k + \lceil \log_2 n \rceil$ typically close to $k$ bits
CPA	$\leq k + \lceil \log_2 n \rceil$

---

---

---

---

---

---

---

---

**Parameters of tree carry-save adders (2)**

Maximum number of inputs that can be reduced to two by an  $h$ -level tree,  $n(h)$

$$n(0) = 2$$

$$n(h) = \left\lfloor \frac{3}{2} n(h-1) \right\rfloor$$

- $n(1) = 3$
- $n(2) = 4$
- $n(3) = 6$
- $n(4) = 9$
- $n(5) = 13$
- $n(6) = 19$

$$2 \left( \frac{3}{2} \right)^{h-1} < n(h) \leq 2 \left( \frac{3}{2} \right)^h$$

---

---

---

---

---

---

---

---

**Parameters of tree carry-save adders (3)**

**Smallest height of the tree carry save adder for n operands, h(n)**

$$h(n) = 1 + h\left(\left\lceil \frac{2}{3}n \right\rceil\right)$$

$$h(2) = 0$$

$$h(n) \geq \log_{\frac{3}{2}}\left(\frac{n}{2}\right)$$

---

---

---

---

---

---

---

---

**Table 8.1 The maximum number n(h) of inputs for an h-level carry-save-adder tree**

h	n(h)	h	n(h)	h	n(h)
0	2	7	28	14	474
1	3	8	42	15	711
2	4	9	63	16	1066
3	6	10	94	17	1599
4	9	11	141	18	2398
5	13	12	211	19	3597
6	19	13	316	20	5395

---

---

---

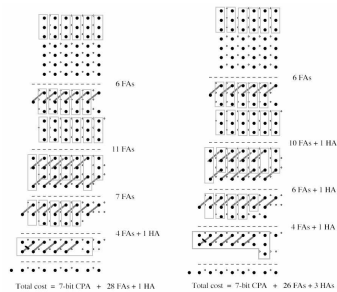
---

---

---

---

---



- 3. 8.14 Adding seven 6-bit numbers using Dadda's strategy (left).
- 3. 8.15 Adding seven 6-bit numbers by taking advantage of the final adder's carry-in (right).

---

---

---

---

---

---

---

---

### Wallace vs. Dadda Trees

#### Wallace trees

- Reduce the size of the final Carry Propagate Adder (CPA)
- Optimum from the point of view of **speed**

#### Dadda trees

- Reduce the cost of the carry save tree
- Optimum (among the CSA trees) from the point of view of **area**

---

---

---

---

---

---

---

---

### 5-to-3 Parallel Counter

	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
a	0	1	0	1	0
b	1	1	0	1	1
c	1	0	1	1	1
d	1	0	1	1	1
e	1	1	1	1	1
s0	0	1	1	1	0
s1	0	1	1	0	0
s2	1	0	0	1	1

$$a+b+c+d+e = s_0+s_1+s_2$$

---

---

---

---

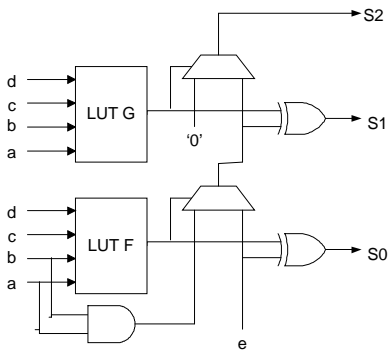
---

---

---

---

### Implementation of 1-bit of 5-to-3 parallel counter using single CLB slice of a Virtex FPGA




---

---

---

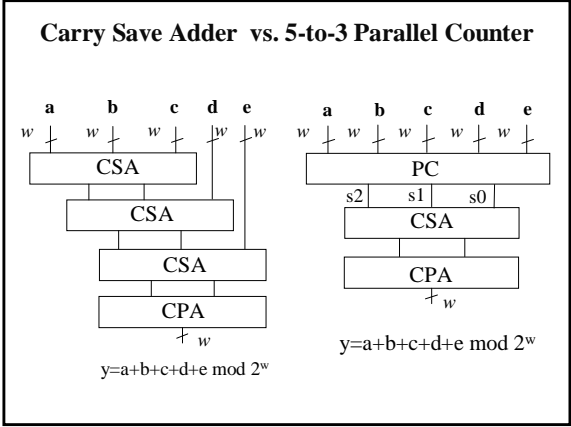
---

---

---

---

---




---

---

---

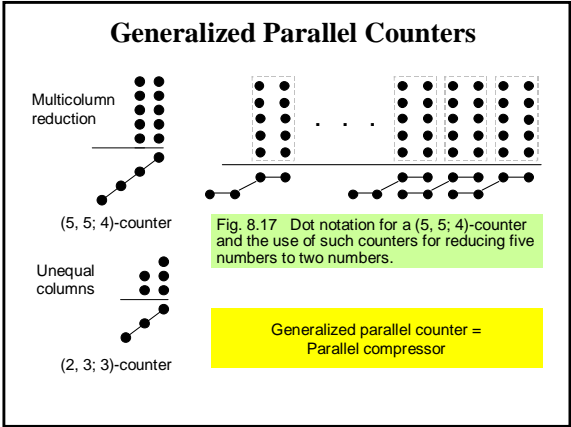
---

---

---

---

---




---

---

---

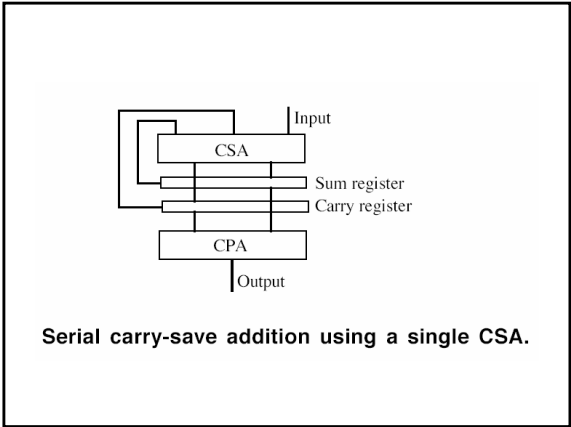
---

---

---

---

---




---

---

---

---

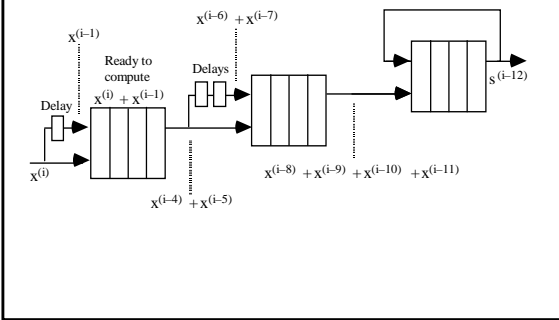
---

---

---

---

### Serial carry propagate adder with pipelining




---



---



---



---



---



---