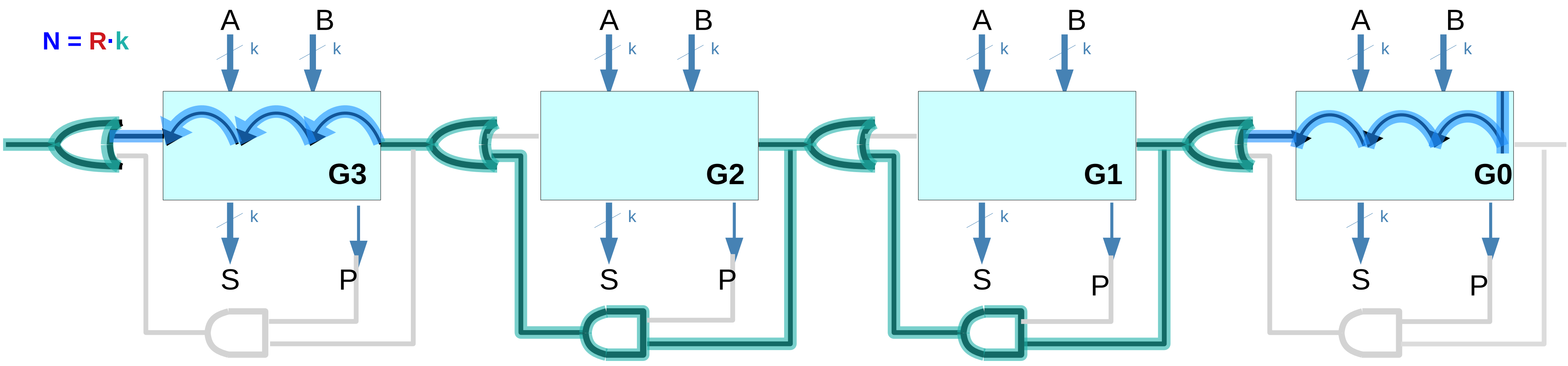
Variable Block Adder (1C)

-
-

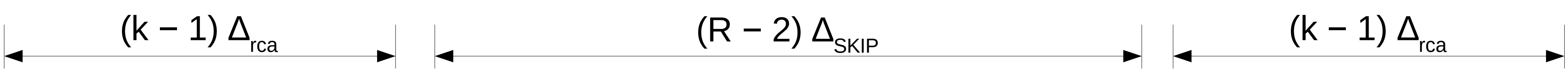
Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using OpenOffice and Octave.

Carry Skip Adder



Fixed block size = k bits



$(k - 1)\ t$ $(m - 2)\ T$ $(k - 1)\ t$

Variable block size = x_i bits for the i-th group

$(x_i - 1)\ t$ $(m - 2)\ T$ $(x_j - 1)\ t$

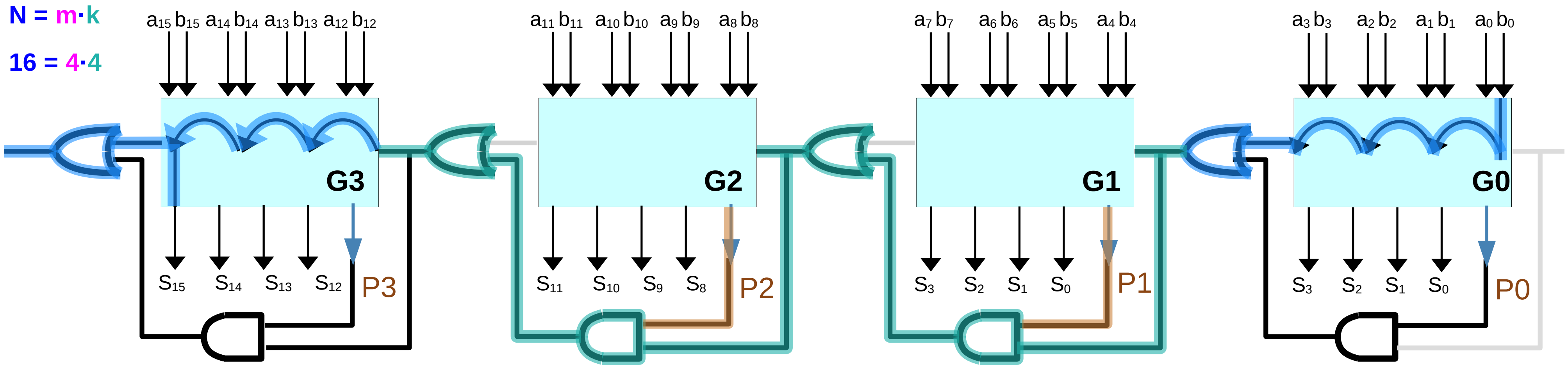
t denote the time required for a carry signal to ripple across a bit
T denote the time required for the signal to skip over a group of bits
m denotes the optimal number of groups for an n-bit carry chain

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Carry Skip Adder – fixed block size

$N = m \cdot k$

$16 = 4 \cdot 4$



$P3 = p_{15} \cdot p_{14} \cdot p_{13} \cdot p_{12}$ $P2 = p_{11} \cdot p_{10} \cdot p_9 \cdot p_8$ $P1 = p_7 \cdot p_6 \cdot p_5 \cdot p_4$ $P0 = p_3 \cdot p_2 \cdot p_1 \cdot p_0$

$(k - 1)\ \Delta_{rca}$ $(m - 2)\ \Delta_{SKIP}$ $(k - 1)\ \Delta_{rca}$

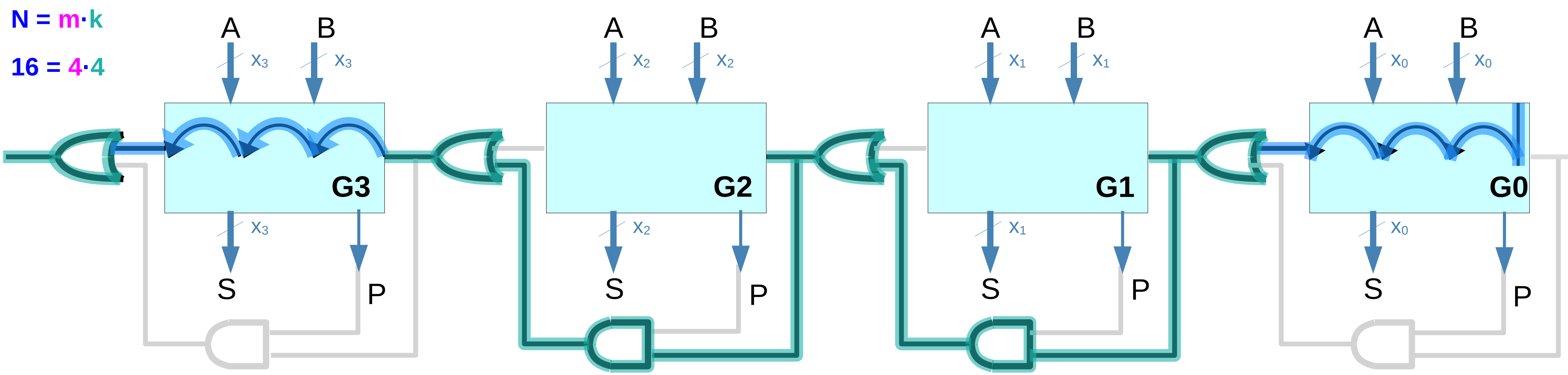
$(k - 1)\ t$ $(m - 2)\ T$ $(k - 1)\ t$

t denote the time required for a carry signal to ripple across a bit
T denote the time required for the signal to skip over a group of bits
m denotes the optimal number of groups for an n-bit carry chain

Fixed Block Size ➡ delay(P3) = delay(P2) = delay(P1) = delay(P0) = Fixed Delay

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Carry Skip Adder – maximum carry delay (3)



x_3 = bit size of G3
x_2 = bit size of G2
x_1 = bit size of G1
x_0 = bit size of G

$(X_3 - 1)\ \Delta_{rca}$ | $(m - 2)\ \Delta_{SKIP}$ | $(x_0 - 1)\ \Delta_{rca}$

$(x_3 - 1)\ t$ | $(m - 2)\ T$ | $(x_0 - 1)\ t$

t denote the time required for a carry signal to ripple across a bit
T denote the time required for the signal to skip over a group of bits
m denotes the optimal number of groups for an n-bit carry chain

$(x_3 - 1)\ 1$ | $(m - 2)\ T'$ | $(x_0 - 1)\ 1$

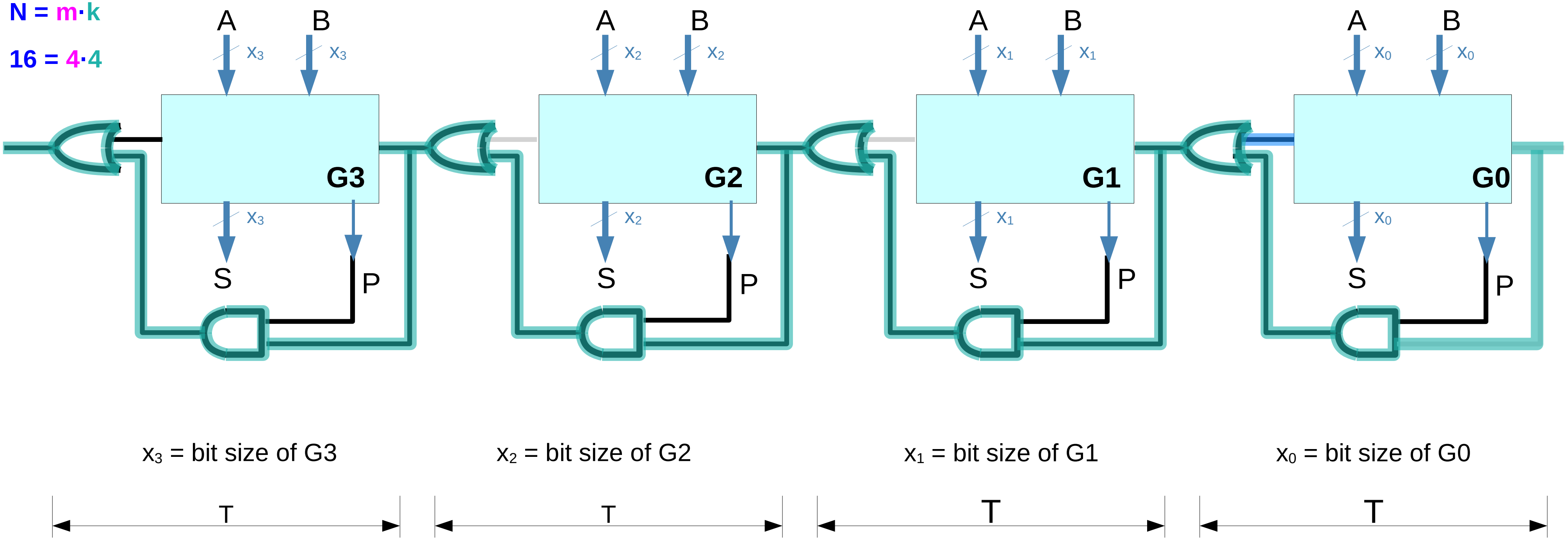
T' normalized delay of *T* over *t*

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Carry Skip Adder – maximum carry delay (3)

$N = m \cdot k$

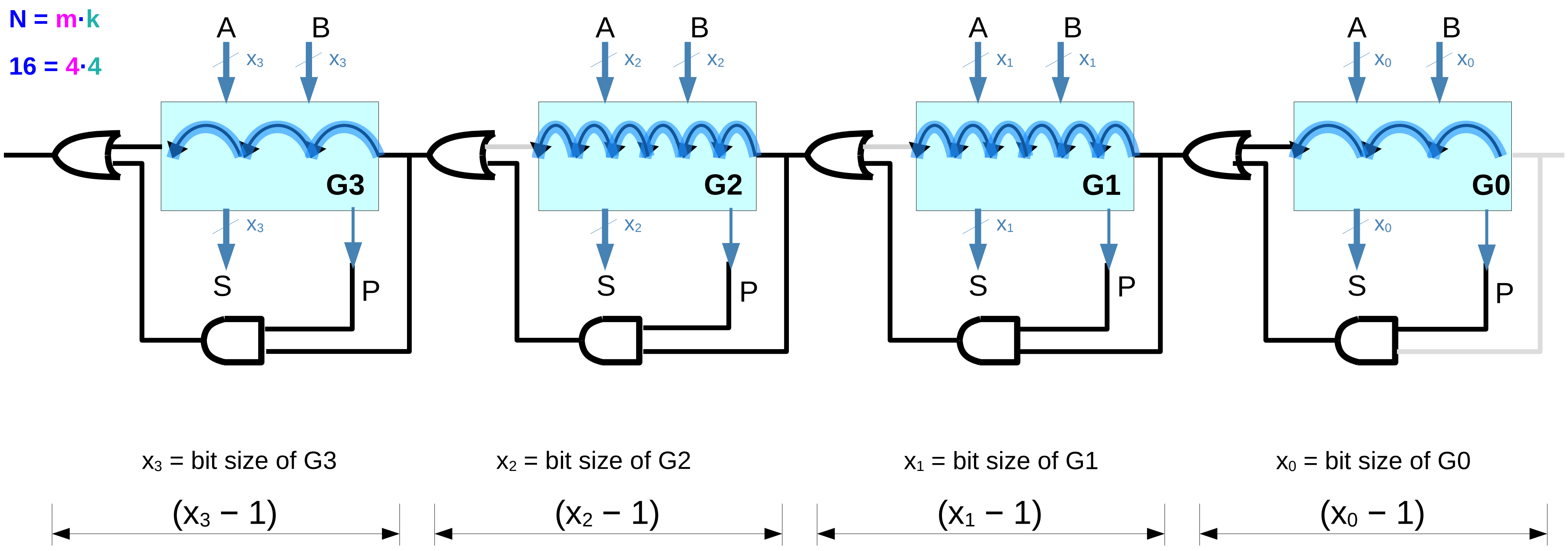
$16 = 4 \cdot 4$



Carry Skip Delays

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

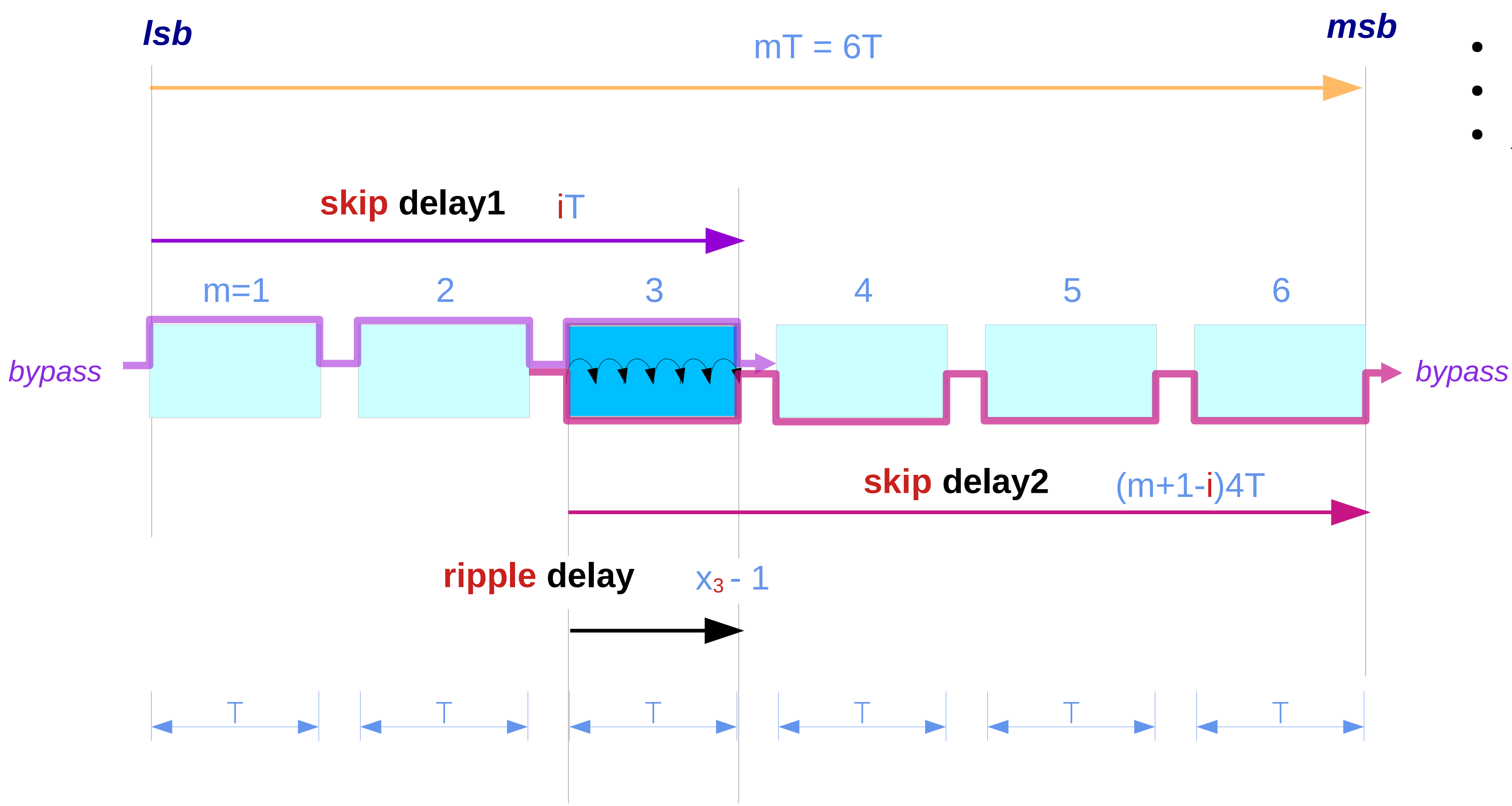
Carry Skip Adder – maximum carry delay (3)



Carry Ripple delays

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Minimum skip path delay y_i of the i^{th} group

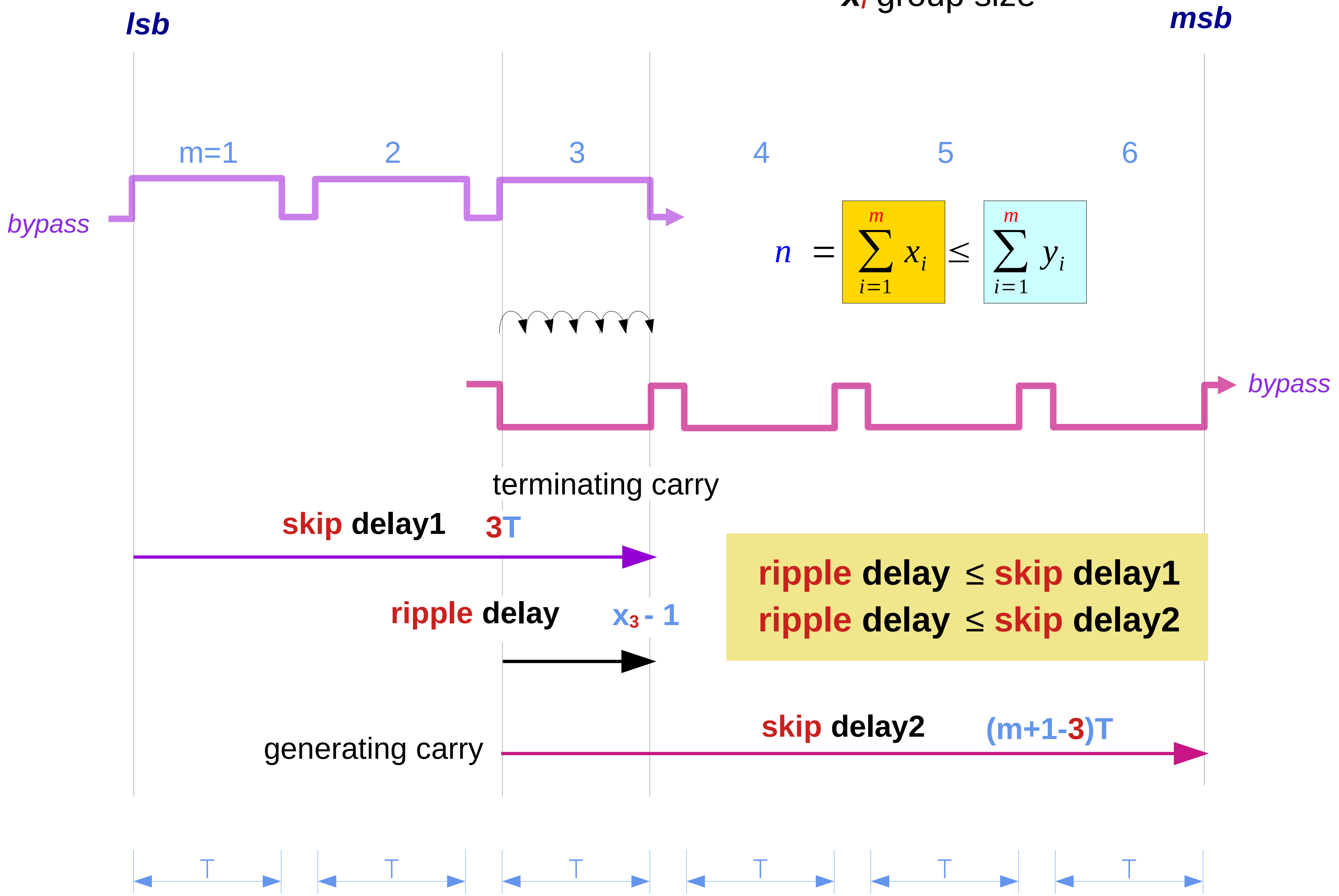


- ***n*** bits
- ***m*** groups
- x_i group size

terminating carry	**ripple delay ≤ skip delay1**
generating carry	**ripple delay ≤ skip delay2**

Parallel Delay Paths

- x_i group size



$$n = \sum_{i=1}^{m} x_i \le \sum_{i=1}^{m} y_i$$

ripple delay ≤ skip delay1
ripple delay ≤ skip delay2

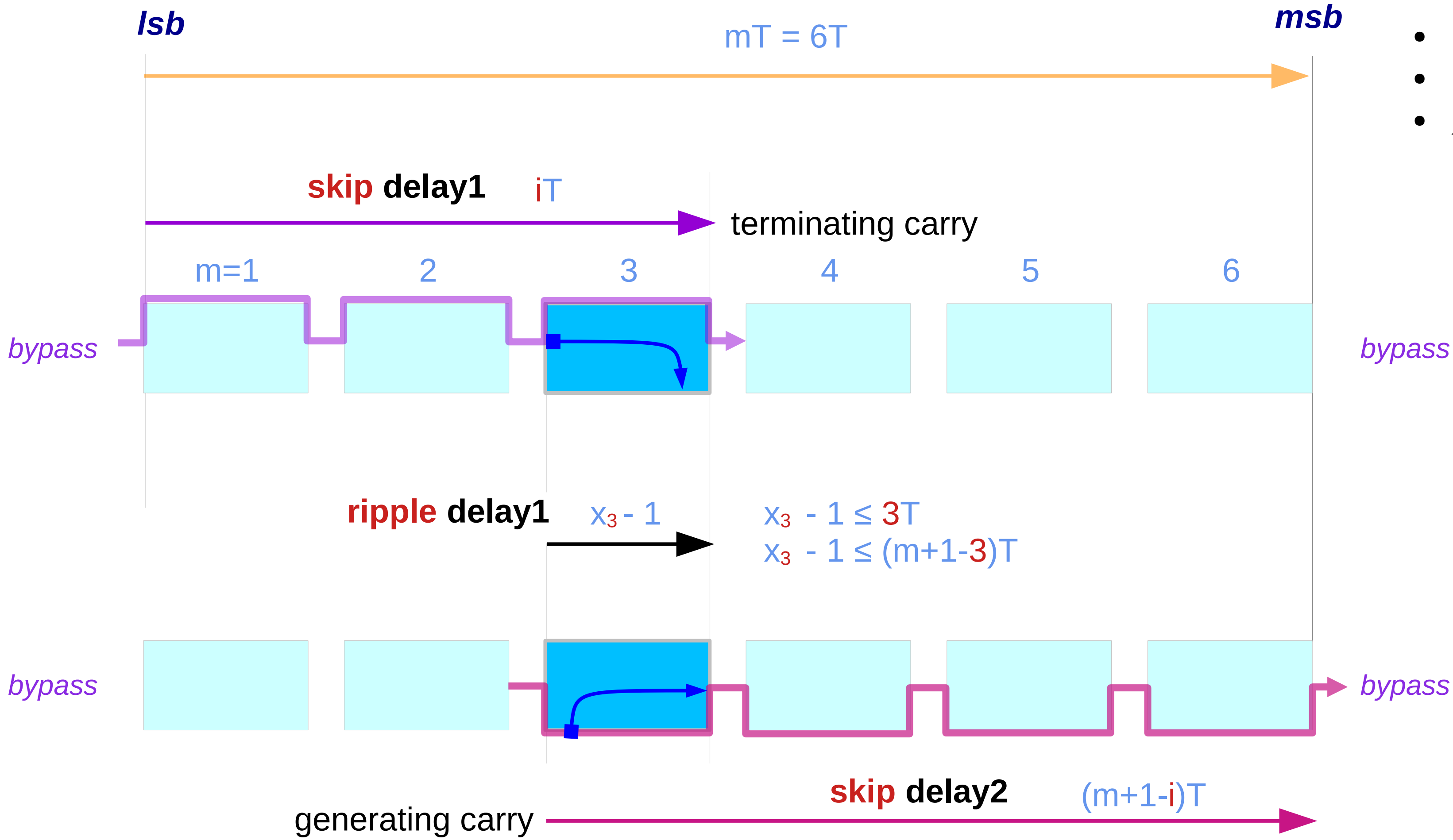
m = 2;
y_1 = 1+T
y_2 = 1+T

m = 3;
y_1 = 1+T
y_2 = 1+2T
y_3 = 1+T

m = 4;
y_1 = 1+T
y_2 = 1+2T
y_3 = 1+2T
y_4 = 1+T

m = 5;
y_1 = 1+T
y_2 = 1+2T
y_3 = 1+3T
y_4 = 1+2T
y_5 = 1+T

Minimum skip path delay y_i of the i^{th} group



- ***n*** bits
- ***m*** groups
- x_i group size

$$x_3 \le y_3 = \min(1 + iT, 1 + (m+1-i)T)$$

Determining *m* group size

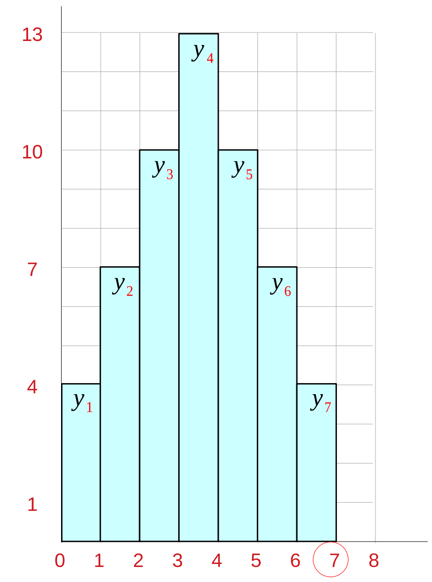
Method 1 – using a histogram

Let m be the <u>smallest</u> positive integer such that

$$n \le \sum_{i=1}^{m} y_i$$

```
m = 2;
while (y1+⋯+ym < n)  m = m+1 ;
```

$$y_i = min\{1+iT, 1+(m+1-i)T\}, \quad i = 1, \dots, m$$



Method 2 – using a *closed formula*

Let m be the <u>smallest</u> positive integer such that

$$n \le m + \frac{1}{2} m T + \frac{1}{4} m^2 T + \left(1 - (-1)^m\right) \frac{1}{8} T$$

$m = 2k$

$\frac{m}{2} = k$

$$y_i = min\{1+iT, \quad 1+(m+1-i)T\}, \quad i = 1, \dots, m \qquad m \quad \frac{1}{2} \cdot k(k+1)$$

$$
\begin{array}{llll}
y_1 = min\{1+1 \cdot T, & 1+(m-0) \cdot T\} & 0 \le x_1 & \le 1+1 \cdot T \\
y_2 = min\{1+2 \cdot T, & 1+(m-1) \cdot T\} & 0 \le x_2 & \le 1+2 \cdot T \\
y_3 = min\{1+3 \cdot T, & 1+(m-2) \cdot T\} & 0 \le x_3 & \le 1+3 \cdot T \\
y_k = min\{1+k \cdot T, & 1+(k+1) \cdot T\} & 0 \le x_k & \le 1+k \cdot T \\
\hline
y_{k+1} = min\{1+(k+1) \cdot T, & 1+k \cdot T\} & 0 \le x_{k+1} & \le 1+k \cdot T \\
y_{m-2} = min\{1+(m-2) \cdot T, & 1+3 \cdot T\} & 0 \le x_{m-2} & \le 1+3 \cdot T \\
y_{m-1} = min\{1+(m-1) \cdot T, & 1+2 \cdot T\} & 0 \le x_{m-1} & \le 1+2 \cdot T \\
y_{m-0} = min\{1+(m-0) \cdot T, & 1+1 \cdot T\} & 0 \le x_{m-0} & \le 1+1 \cdot T
\end{array}
$$

$\frac{1}{2} \cdot k(k+1)$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

$$0 \le x_i \le y_i, i = 1, \dots, m$$

Determining *m* group size

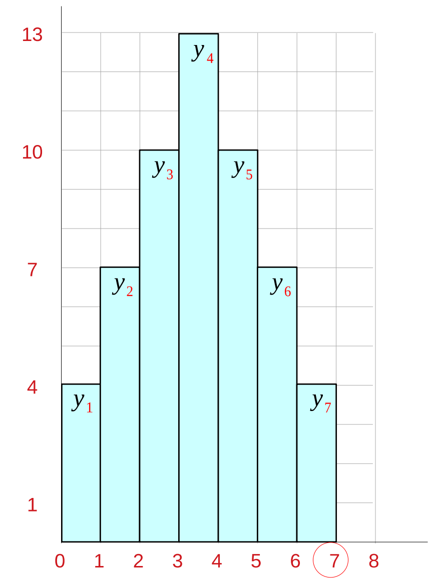
Method 1 – using a histogram

Let m be the smallest positive integer such that

$$n \le \sum_{i=1}^{m} y_i$$

m = 2;
while ($y_1 + \cdots + y_m < n$) m = m+1 ;

$$y_i = min\{1+iT, 1+(m+1-i)T\}, \quad i = 1, \dots, m$$



m = 2; T = 3
y_1 = min {1+T, 1+2T} = 1+T = 4
y_2 = min {1+2T, 1+T} = 1+T = 4

m = 3; T = 3
y_1 = min {1+T, 1+3T} = 1+T = 4
y_2 = min {1+2T, 1+2T} = 1+2T = 7
y_3 = min {1+3T, 1+T} = 1+T = 4

m = 4; T = 3
y_1 = min {1+T, 1+4T} = 1+T = 4
y_2 = min {1+2T, 1+3T} = 1+2T = 7
y_3 = min {1+3T, 1+2T} = 1+2T = 7
y_4 = min {1+4T, 1+T} = 1+T = 4

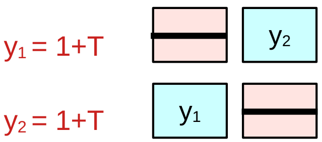
m = 5; T = 3
y_1 = min {1+T, 1+5T} = 1+T = 4
y_2 = min {1+2T, 1+4T} = 1+2T = 7
y_3 = min {1+3T, 1+3T} = 1+3T = 10
y_4 = min {1+4T, 1+2T} = 1+2T = 7
y_5 = min {1+5T, 1+T} = 1+T = 4

m = 6; T = 3
y_1 = min {1+T, 1+6T} = 1+T = 4
y_2 = min {1+2T, 1+5T} = 1+2T = 7
y_3 = min {1+3T, 1+4T} = 1+3T = 10
y_4 = min {1+4T, 1+3T} = 1+3T = 10
y_5 = min {1+5T, 1+2T} = 1+2T = 7
y_6 = min {1+6T, 1+T} = 1+T = 4

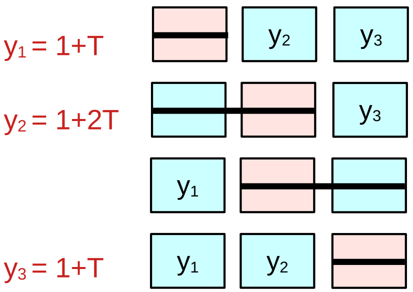
m = 7; T = 3
y_1 = min {1+T, 1+7T} = 1+T = 4
y_2 = min {1+2T, 1+6T} = 1+2T = 7
y_3 = min {1+3T, 1+5T} = 1+3T = 10
y_4 = min {1+4T, 1+4T} = 1+4T = 13
y_5 = min {1+5T, 1+3T} = 1+3T = 10
y_6 = min {1+6T, 1+2T} = 1+2T = 7
y_7 = min {1+7T, 1+1T} = 1+T = 4

Determining *m* group size

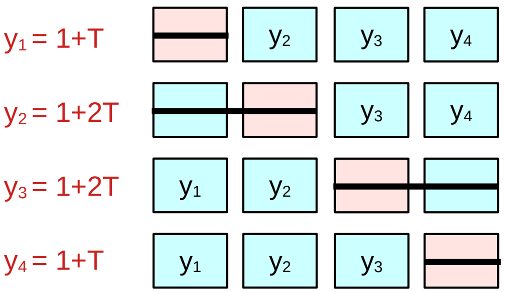
$m = 2;$



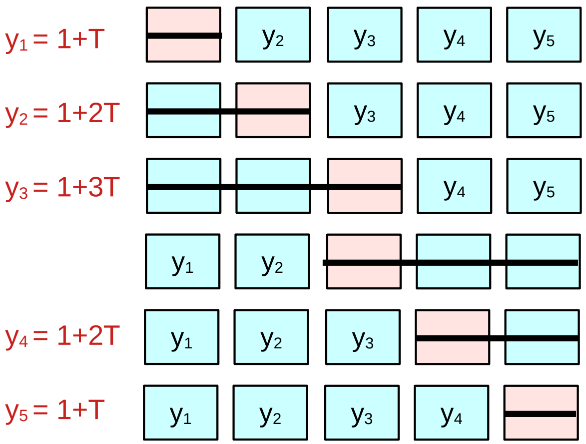
$m = 3;$



$m = 4;$



$m = 5;$



$m = 2;$
$x_1 \le 1+T$
$x_2 \le 1+T$

$m = 3;$
$x_1 \le 1+T$
$x_2 \le 1+2T$
$x_3 \le 1+T$

$m = 4;$
$x_1 \le 1+T$
$x_2 \le 1+2T$
$x_3 \le 1+2T$
$x_4 \le 1+T$

$m = 5;$
$x_1 \le 1+T$
$x_2 \le 1+2T$
$x_3 \le 1+3T$
$x_4 \le 1+2T$
$x_5 \le 1+T$

Determining *m* group size

m = 6;

$y_1 = 1+T$

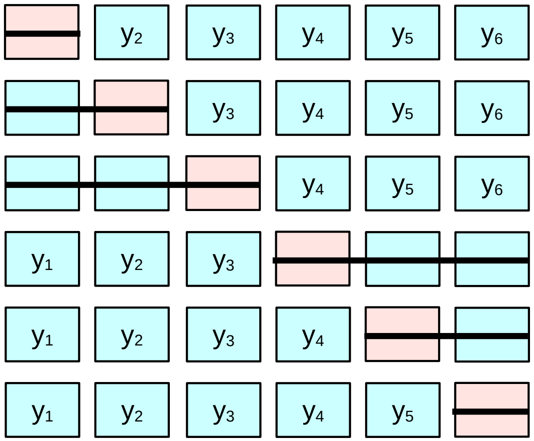
$y_2 = 1+2T$

$y_3 = 1+3T$

$y_4 = 1+3T$

$y_5 = 1+2T$

$y_6 = 1+T$



m = 7;

$y_1 = 1+T$

$y_2 = 1+2T$

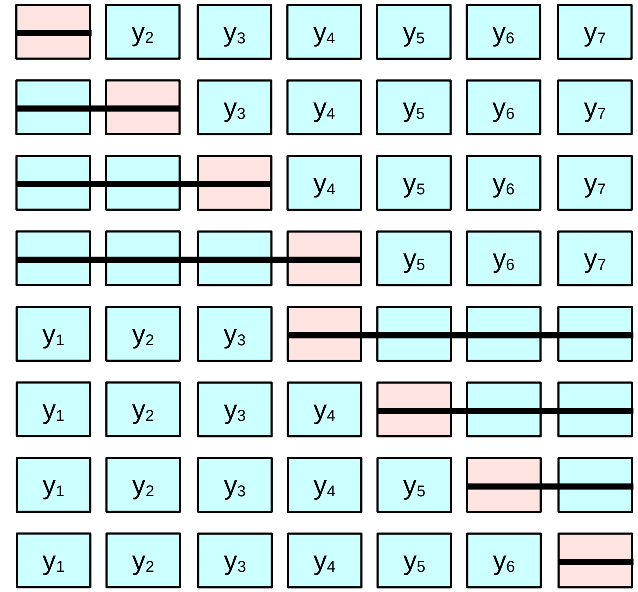
$y_3 = 1+3T$

$y_4 = 1+4T$

$y_5 = 1+3T$

$y_6 = 1+2T$

$y_7 = 1+T$



m = 2;
$x_1 = 1+T$
$x_2 = 1+T$

m = 3;
$x_1 = 1+T$
$x_2 = 1+2T$
$x_3 = 1+T$

m = 4;
$x_1 = 1+T$
$x_2 = 1+2T$
$x_3 = 1+2T$
$x_4 = 1+T$

Determining *m* group size

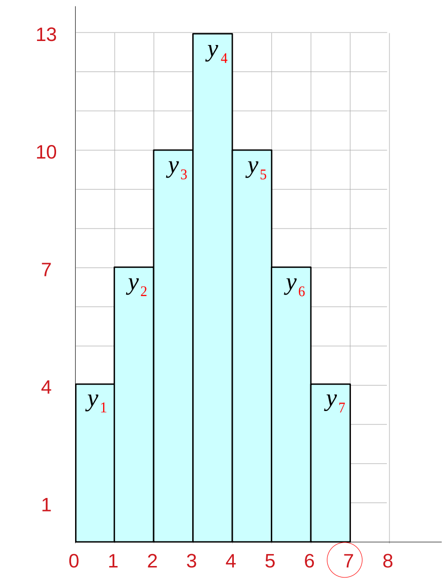
Method 1 – using a histogram

Let *m* be the <u>smallest</u> positive integer such that

$$n \le \sum_{i=1}^{m} y_i$$

m = 2;
while ($y_1+\cdots+y_m$ < n) m = m+1 ;

$$y_i = min\{1+iT, 1+(m+1-i)T\}, \quad i = 1, \ldots, m$$



$m=2$	$m=3$	$m=4$	$m=5$	$m=6$	$m=7$
$\sum_{i=1}^{2} y_i$	$\sum_{i=1}^{3} y_i$	$\sum_{i=1}^{4} y_i$	$\sum_{i=1}^{5} y_i$	$\sum_{i=1}^{6} y_i$	$\sum_{i=1}^{7} y_i$
$2+2\cdot T$	$3+4\cdot T$	$4+6\cdot T$	$5+9\cdot T$	$6+12\cdot T$	$7+16\cdot T$
$2+2\cdot T < n$	$3+4\cdot T < n$	$4+6\cdot T < n$	$5+9\cdot T < n$	$6+12\cdot T < n$	$7+16\cdot T > n$
$2+2\cdot 3 < 48$	$3+4\cdot 3 < 48$	$4+6\cdot 3 < 48$	$5+9\cdot 3 < 48$	$6+12\cdot 3 < 48$	$7+16\cdot 3 > 48$
$8 < 48$	$15 < 48$	$22 < 48$	$32 < 48$	$42 < 48$	$55 > 48$
					$m = 7$

$n = 48$
$T = 3$

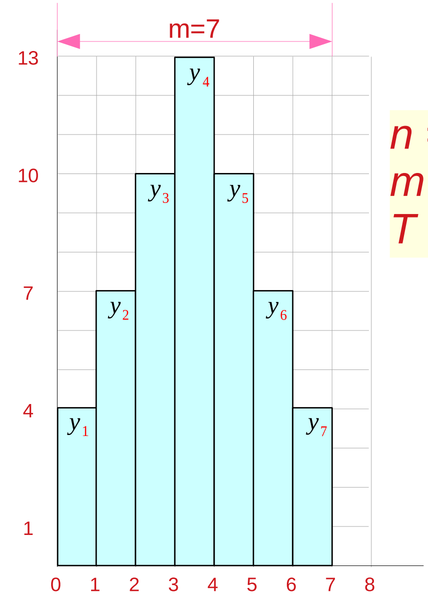
$m = 7$

Determining x_i

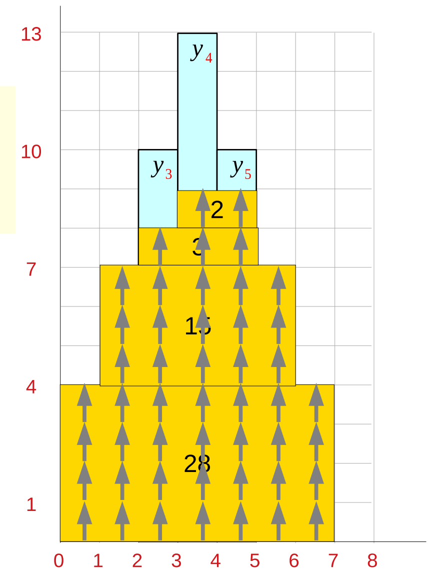
construct a histogram
whose i-*th* column has height y_i

so these y_i's are at least n unit squares
in the histogram, starting with the first row,
shade in n of the squares, row by row

let x_i denote the number of shaded squares
in column i of the histogram,
$i = 1, \ldots, m$



$n = 48$
$m = 7$
$T = 3$

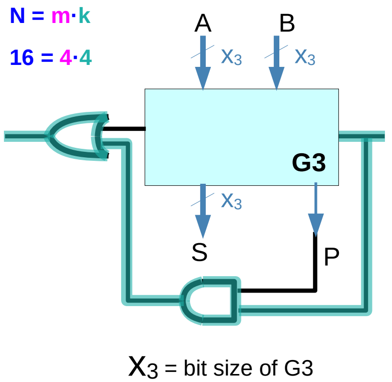


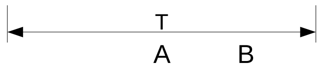
$$0 \le x_i \le y_i, \quad i = 1, \ldots, m$$

$$n = \sum_{i=1}^{m} x_i \le \sum_{i=1}^{m} y_i$$

$$n = \sum_{i=1}^{7} x_i$$

$$n = 4+7+8+9+9+7+4 = 48 \; < \; 7+16\cdot 3 = 55$$





$m = 7$; $T = 3$
$x_1 = 4 \le y_1 = 4$
$x_2 = 7 \le y_2 = 7$
$x_3 = 8 < y_3 = 10$
$x_4 = 9 < y_4 = 13$
$x_5 = 9 < y_5 = 10$
$x_6 = 7 \le y_6 = 7$
$x_7 = 4 \le y_7 = 4$

Procedure

(I) Let m be the smallest positive integer such that

$$n \le m + \frac{1}{2} m T + \frac{1}{4} m^2 T + \left(1 - (-1)^m\right) \frac{1}{8} T = \sum_{i=1}^{m} y_i$$

- total n =48 bits
- m =7 groups
- i-th group has x_i bits (size)
- constant skip delay $T = T(x_i)$ =3

(II) Let

$$y_i = min\{1 + iT, 1 + (m + 1 - i)T\}, \quad i = 1, \ldots, m$$

and construct a histogram whose *i-th* column has height y_i

for example, for T=3, and n=48, we have m=7

(III) It is easily verified that the area of the histogram in (II) is

$$\sum_{i=1}^{m} y_i = m + \frac{1}{2} m T + \frac{1}{4} m^2 T + \left(1 - (-1)^m\right) \frac{1}{8} T \ge n$$

so these are <u>at least</u> n <u>unit squares</u> in the histogram
starting with the first row, shade in n of the squares, <u>row by row</u>
Let x_i denote the number of shaded squares in column i of the histogram,
$i = 1, \ldots, m$

$$n = \sum_{i=1}^{m} x_i \le \sum_{i=1}^{m} y_i$$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum propagation time P

$$y_i = min\{1+iT, 1+(m+1-i)T\}, \quad i = 1,\ldots,m$$

the scheme (i), (ii), (iii)
gives the max prop time *mT*

$$y_1 = min\{1+1\cdot T, 1+(m+1-1)T\} = 1+T \qquad x_1 \le y_1 = 1+T$$

$$y_m = min\{1+m\cdot T, 1+(m+1-m)T\} = 1+T \qquad x_m \le y_m = 1+T$$

$$y_2 = min\{1+2\cdot T, 1+(m+1-2)T\} = 1+2T \qquad x_2 \le y_2 = 1+2T$$

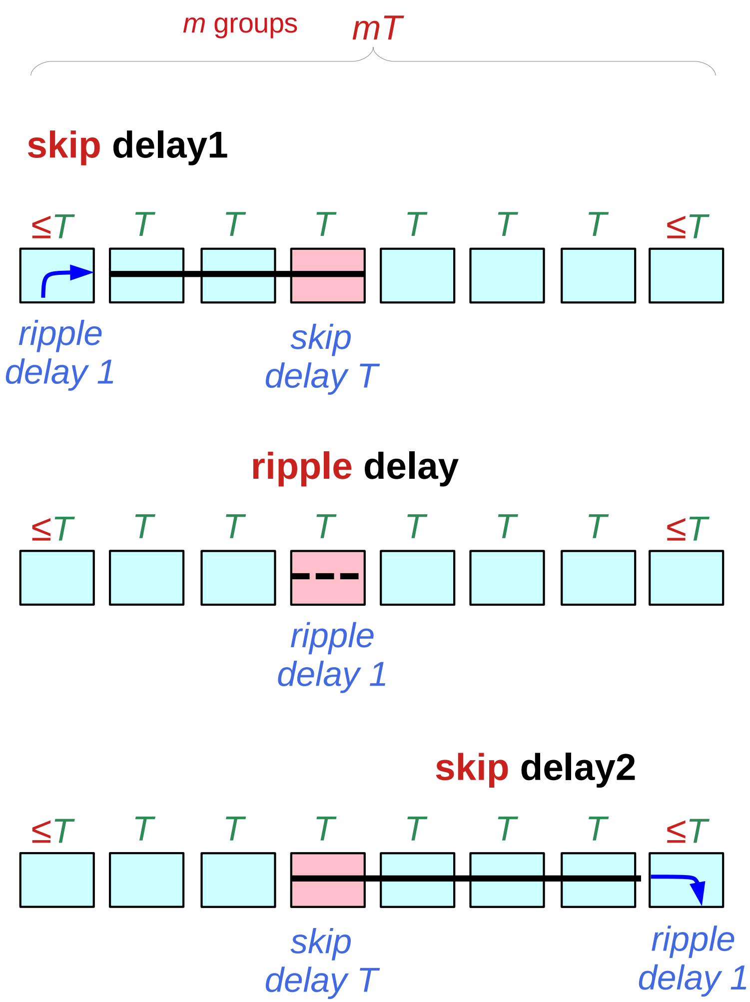
$$y_{m-1} = min\{1+(m-1)\cdot T, 1+(m+1-(m-1))T\} = 1+2T \qquad x_{m-1} \le y_{m-1} = 1+2T$$

$$y_3 = min\{1+3\cdot T, 1+(m+1-3)T\} = 1+3T \qquad x_3 \le y_3 = 1+3T$$

$$y_{m-2} = min\{1+(m-2)\cdot T, 1+(m+1-(m-2))T\} = 1+3T \qquad x_{m-2} \le y_{m-2} = 1+3T$$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum propagation time P



the scheme (i), (ii), (iii)
gives the max prop time mT

skip delay1	iT	generating carry
ripple delay	$x_i - 1$	
skip delay2	$(m+1-i)T$	terminating carry

$x_i - 1 \leq iT$
$x_i - 1 \leq (m+1-i)T$

$x_i \leq 1 + iT$
$x_i \leq 1 + (m+1-i)T$

$x_i \leq \min\{1 + iT, 1 + (m+1-i)T\}$
$x_i \leq y_i$

$y_i = \min\{1 + iT, 1 + (m+1-i)T\}$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum propagation time P

$$y_i = min\{1+iT, 1+(m+1-i)T\}, \quad i = 1, \ldots, m$$

the scheme (i), (ii), (iii)
gives the max prop time mT

$$y_1 = min\{1+1\cdot T, 1+(m+1-1)T\} = 1+T$$

$$y_m = min\{1+m\cdot T, 1+(m+1-m)T\} = 1+T$$

$$x_1 \le y_1 = 1+T$$

$$x_m \le y_m = 1+T$$

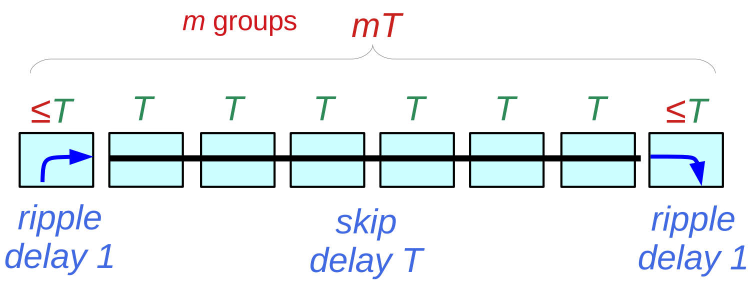
$x_1 - 1 \le 1T$
$x_1 - 1 \le (m+1-1)T$

$x_1 - 1 \le T$
$x_1 - 1 \le mT$

$x_m - 1 \le mT$
$x_m - 1 \le (m+1-m)T$

$x_m - 1 \le mT$
$x_m - 1 \le T$

maximum propagation time



$$P_{max} = P_{1,m} \le mT$$

$$P = P_{i,j} \le mT$$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum propagation time P

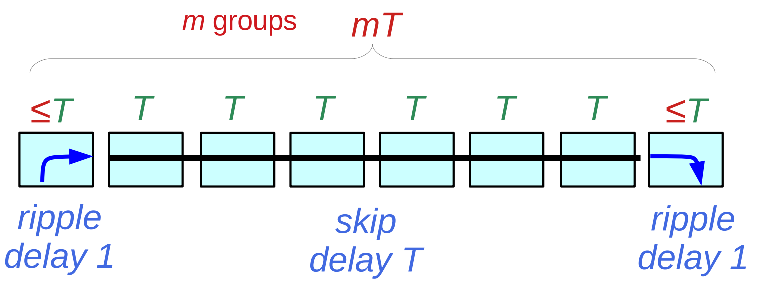
Lemma 1 When the bits of a carry skip adder are grouped according to the scheme (i)-(iii), the maximum propagation time of a carry signal is mT

the scheme (i), (ii), (iii) gives the max prop time mT

The carry generated at the 2nd bit position and terminating at the $(n\text{-}1)^{th}$ bit position clearly has propagation time mT.

We must show that *any other* carry signal has propagation time smaller than or equal to mT

maximum propagation time



$$P_{max} = P_{1,m} \leq mT$$

propagation time of a carry signal $\leq mT$
the maximum propagation time = mT

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Procedure

(I) Let m be the smallest positive integer

$$n \leq \sum_{i=1}^{m} y_i \qquad i = 1, \ldots, m$$

(II) Let

$$y_i = \min\{1 + iT, 1 + (m + 1 - i)T\}$$

(III) Let x_i , $i = 1, \ldots, m$

starting with the first row, <u>row by row</u>

$$n = \sum_{i=1}^{m} x_i \leq \sum_{i=1}^{m} y_i$$

Variable block size = x_i bits for the i-th group

find the <u>smallest</u> m

$$n \leq \sum_{i=1}^{m} y_i = \sum_{i=1}^{m} \min\{1 + iT, 1 + (m + 1 - i)T\}$$

```
m = 2;
while (y1+···+ym < n) m = m+1 ;
```

m = 2;	m = 4;	m = 5;	m = 6;	m = 7;
y_1 = 1+T	y_1 = 1+T	y_1 = 1+T	y_1 = 1+T	y_1 = 1+T
y_2 = 1+T	y_2 = 1+2T	y_2 = 1+2T	y_2 = 1+2T	y_2 = 1+2T
	y_3 = 1+2T	y_3 = 1+3T	y_3 = 1+3T	y_3 = 1+3T
m = 3;	y_4 = 1+T	y_4 = 1+2T	y_4 = 1+3T	y_4 = 1+4T
y_1 = 1+T		y_5 = 1+T	y_5 = 1+2T	y_5 = 1+3T
y_2 = 1+2T			y_6 = 1+T	y_6 = 1+2T
y_3 = 1+T				y_7 = 1+T

the scheme (i), (ii), (iii)
gives the **max propagation time** mT

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Propagation Time P

find the <u>smallest</u> m

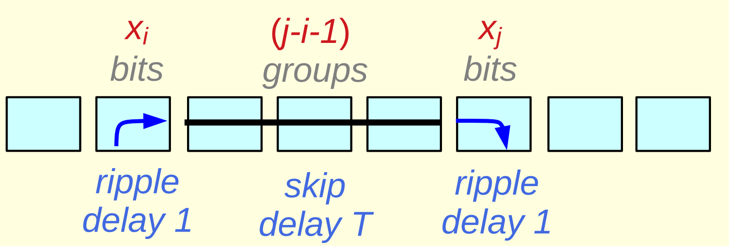
$$n \le \sum_{i=1}^{m} y_i = \sum_{i=1}^{m} min\{1+iT, 1+(m+1-i)T\}$$

$m = 2$;
while ($y_1 + \cdots + y_m < n$) $m = m+1$;

the scheme (i), (ii), (iii)
gives the max propagation time mT

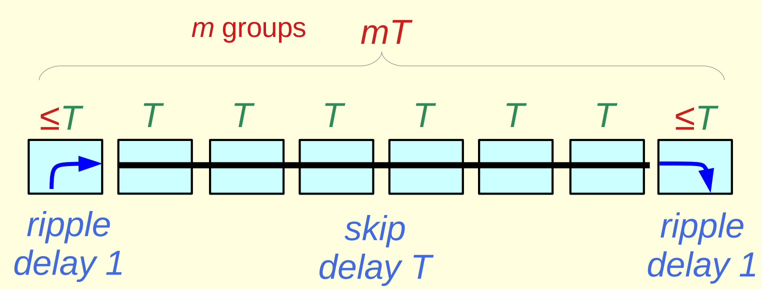
propagation time of a carry signal $\le mT$
the maximum propagation time = mT

propagation time



$$P = P_{i,j} \qquad \forall i, \ \forall j \quad 1 \le i, \ j \le m$$

maximum propagation time



$$P_{max} = P_{1,m} \le mT$$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum delay and optimal group size

the maximum propagation time
$\propto$ the number of groups

$$D \propto m$$

- *not an optimal optimal division*
  - *larger number of groups →*
  - *larger delays →*

- when group size m is not *optimal*

  then there is an optimal group size = r

  - the maximum delay with the group size m $D_m = mT$
  - the maximum delay with the group size r $D_r = rT$
  - *r must be smaller than m* $r \le m$

$$D_r < D_m$$

$$\Rightarrow rT < mT$$

$$\Rightarrow r < m$$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum delay of a carry signal

Lemma 2 Let D denote the maximum delay of a carry signal in a n bit carry skip adder with group sizes chosen optimally. Then

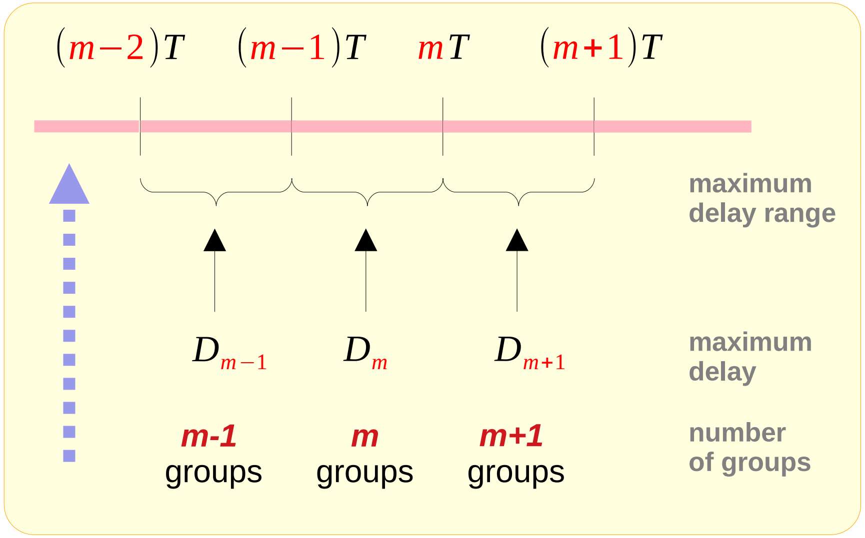
$$(m-1)T \le D \le mT$$

- ***n*** bits
- ***r*** groups

Since we have exhibited a division of the carry chain into groups
In such a way that the maximum delay of a carry signal is mT
We clearly have $D \le mT$

the **maximum delay** = D
the **optimal group size** = m

$$(m-1)T \le D \le mT$$



Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum delay of a carry signal

$$(m-1)T \le D \le mT$$

Assume there are ***r*** *groups*

then 2 cases : ***even r****,* ***odd r***

for each of these 2 cases

prove $mT - D < T + 1$

$\Rightarrow$ $\boxed{mT - D \le T}$

$\Rightarrow$ $(m-1)T \le D$

P: ***the propagation delay of any carry signal path ≤ mT***

upper bound

D: ***the max of P***

$\boxed{\textbf{diff}\,(mT, D) \le T}$

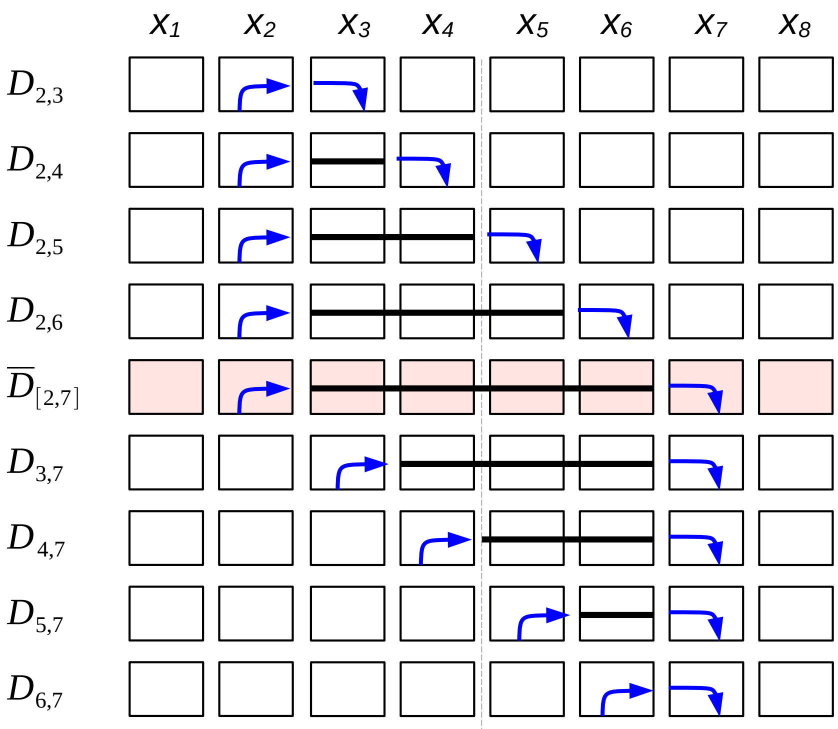
$\Rightarrow$ ***diff (*** mT***,*** *max* ***P) ≤ T***

lower bound

$(m-1)T \le D$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum delays of carry signals (r = 2k)



$\overline{D}_{[2,7]}$ = *the maximum delay of carry signals generated in the i-th group and terminated in the j-th group such that 2 ≤ i, j ≤ 7* $\leq$ D

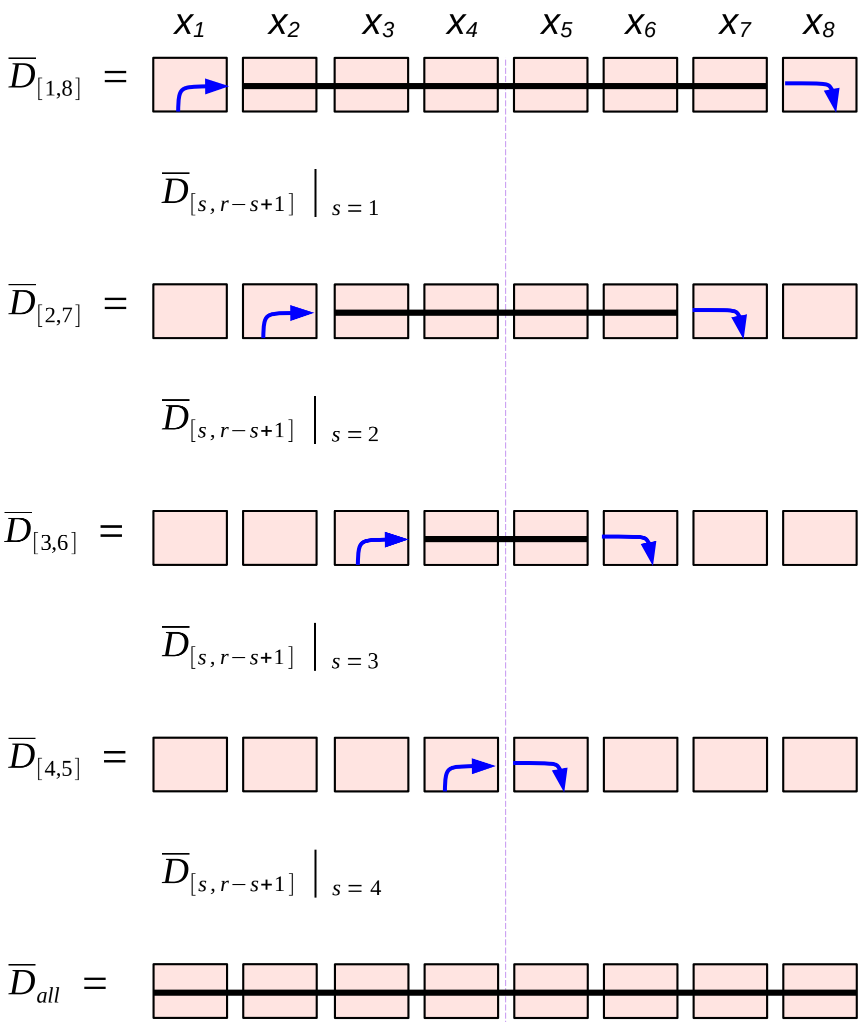
$$\overline{D}_{[2,7]} = max \left\{ \begin{matrix} D_{2,3}, & D_{2,4}, & D_{2,5}, & D_{2,6}, \\ & D_{2,7}, & & \\ D_{3,7}, & D_{4,7}, & D_{5,7}, & D_{6,7} \end{matrix} \right\}$$

$$\overline{D}_{[2,7]} = \overline{D}_{[2,8-2+1]} = \overline{D}_{[s,8-s+1]}, \quad s = 2$$

$\overline{D}_{[s,r-s+1]}$ = *the maximum delay of carry signals generated in the i-th group and terminated in the j-th group such that s ≤ i, j ≤ r-s+1*

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum delays of carry signals (**r = 2k**)



$\overline{D}_{[1,8]}$ = *The maximum delay of carry signals generated in the i-th group or terminated in the j-th group such that 1 ≤ i, j ≤ 8* $\leq D$

$\overline{D}_{[2,7]}$ = *The maximum delay of carry signals generated in the i-th group or terminated in the j-th group such that 2 ≤ i, j ≤ 7* $\leq D$

$\overline{D}_{[3,6]}$ = *The maximum delay of carry signals generated in the i-th group or terminated in the j-th group such that 3 ≤ i, j ≤ 6* $\leq D$

$\overline{D}_{[4,5]}$ = *The maximum delay of carry signals generated in the i-th group or terminated in the j-th group such that 4 ≤ i, j ≤ 5* $\leq D$

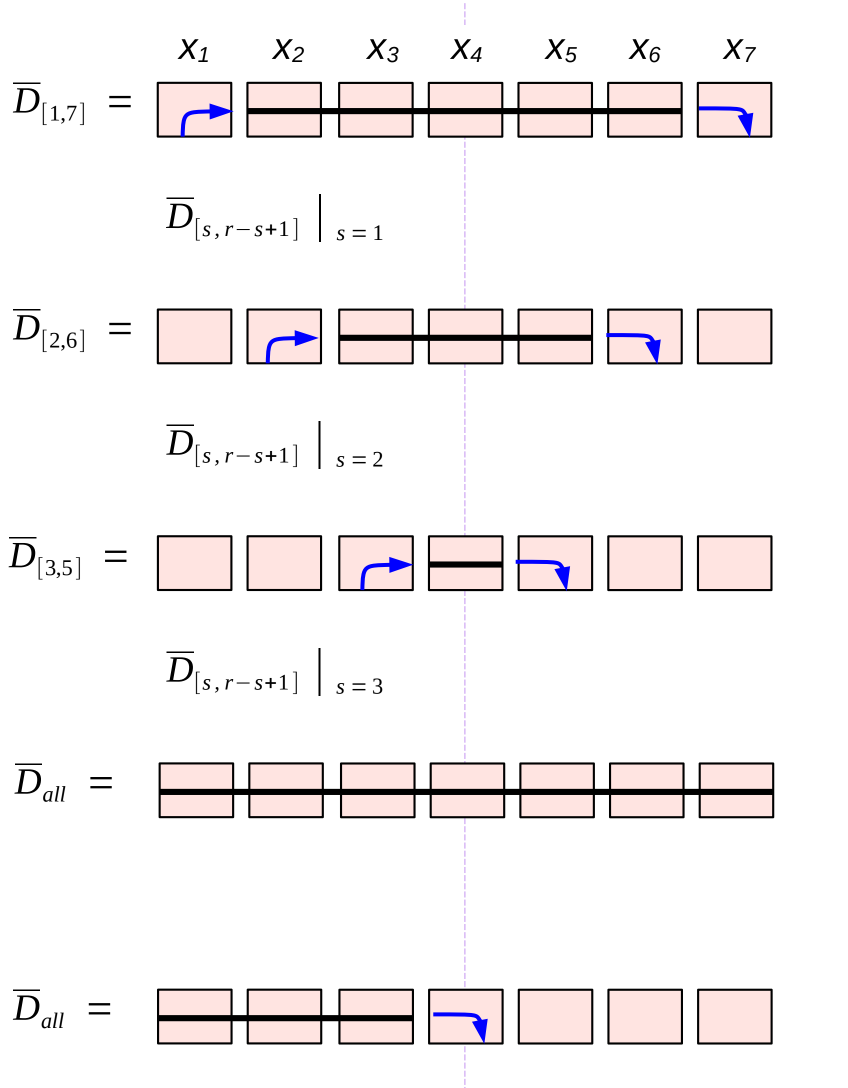
$\overline{D}_{all}$ = *All skip delay* $\leq D$

$$D = max\{\overline{D}_{[1,8]},\ \overline{D}_{[2,7]},\ \overline{D}_{[3,6]},\ \overline{D}_{[4,5]}\}$$

Max delay of all carry signals

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum delays of carry signals (**r = 2k+1**)



$\overline{D}_{[1,7]}$ = *The maximum delay of carry signals generated in the i-th group or terminated in the j-th group such that 1 ≤ i, j ≤ 8* $\leq$ D

$\overline{D}_{[2,6]}$ = *The maximum delay of carry signals generated in the i-th group or terminated in the j-th group such that 2 ≤ i, j ≤ 7* $\leq$ D

$\overline{D}_{[3,65]}$ = *The maximum delay of carry signals generated in the i-th group or terminated in the j-th group such that 3 ≤ i, j ≤ 6* $\leq$ D

$\overline{D}_{all}$ = *All skip delay*

$\tilde{D}_{all}$ = *Comparable to all skip delay* $\leq$ D

$$D = max\{\overline{D}_{[1,8]},\ \overline{D}_{[2,7]},\ \overline{D}_{[3,6]},\ \overline{D}_{[4,5]}\}$$

Max delay of all carry signals

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum delays of carry signals (**r = 2k**)

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
$\overline{D}_{[1,8]} =$								

$$D = \max_{s=1}^{r/2} \overline{D}_{[s,r-s+1]}$$

$$= \max_{s=1}^{k} \overline{D}_{[s,2k+1-s]}$$

$$= \max_{s=1}^{4} \overline{D}_{[s,9-s]}$$

Max delay of all carry signals

$$\begin{aligned} \overline{D}_{[1,r]} &\le D \\ \overline{D}_{[2,r-1]} &\le D \\ \vdots &\quad \vdots \\ \overline{D}_{[k,k+1]} &\le D \end{aligned}$$

$$\overline{D}_{all} \le D$$

$$(m-1)T \le D$$

Lower bound of D

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Maximum delays of carry signals (**r = 2k+1**)

x_1 x_2 x_3 x_4 x_5 x_6 x_7

$\overline{D}_{[1,8]} = \boxed{}\,\boxed{}\,\boxed{}\,\boxed{}\,\boxed{}\,\boxed{}\,\boxed{}$

$$D = \max_{s=1}^{floor(r/2)} \overline{D}_{[s,r-s+1]}$$

$$= \max_{s=1}^{k} \overline{D}_{[s,2k+2-s]}$$

$$= \max_{s=1}^{3} \overline{D}_{[s,8-s]}$$

Max delay of all carry signals

$$\begin{aligned} \overline{D}_{[1,r]} &\le D \\ \overline{D}_{[2,r-1]} &\le D \\ &\vdots \\ \overline{D}_{[k,k+1]} &\le D \end{aligned}$$

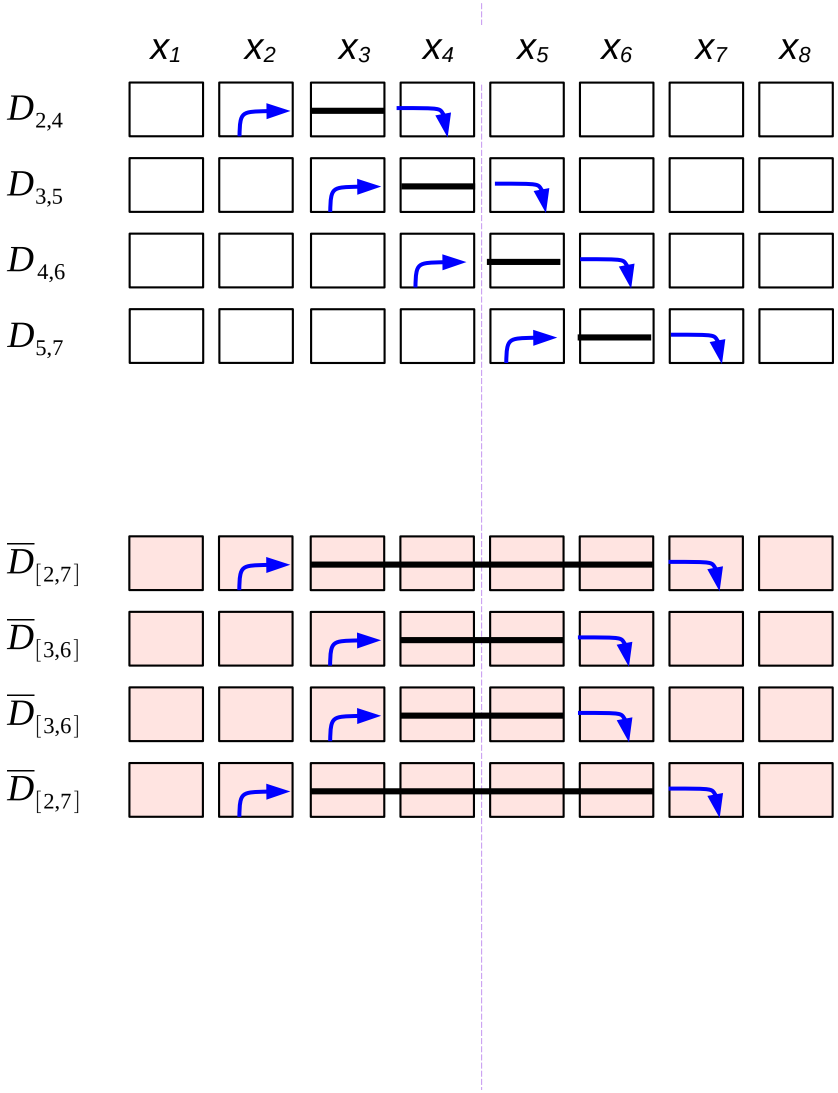
$$\widetilde{D}_{all} \le D$$

$$(m-1)T \le D$$

Lower bound of D

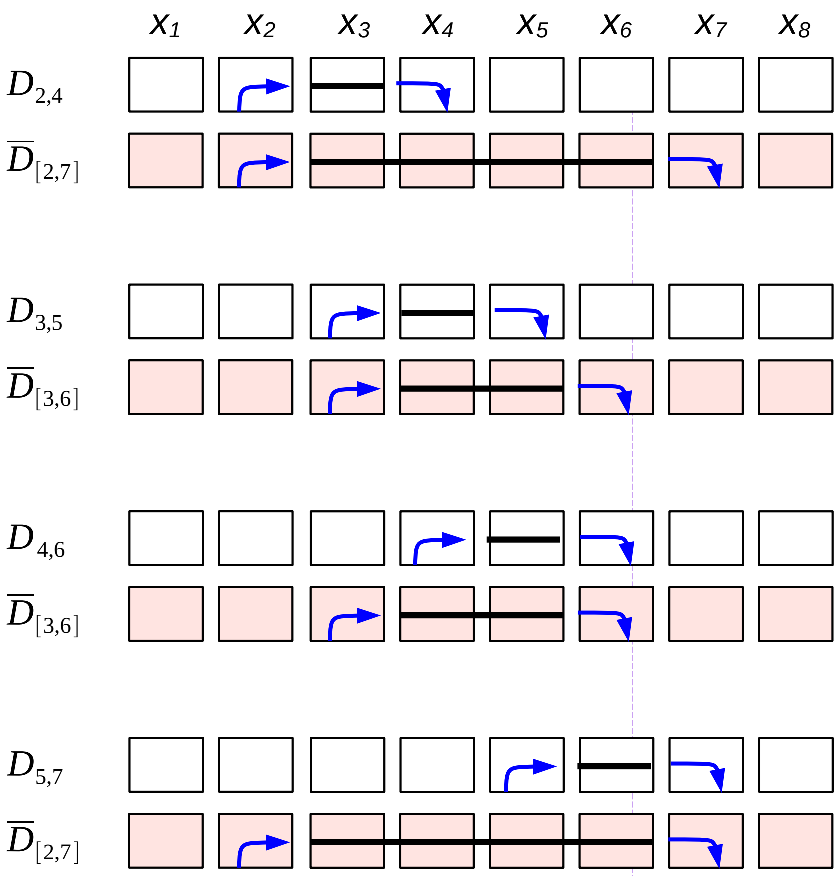
Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Example delays of carry signals (**r = 2k**) (1)



Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Example delays of carry signals (**r = 2k**) (2)



$D_{2,4} \leq \overline{D}_{[2,7]}$

$D_{3,5} \leq \overline{D}_{[3,6]}$

$D_{4,6} \leq \overline{D}_{[3,6]}$

$D_{5,7} \leq \overline{D}_{[2,7]}$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (1-1)

Theorem 1

The scheme 2(i) – 2(iii) given above
for dividing the bits of a carry skip adder into groups
is optimal for $2 \le T \le 7$

dividing the bits into groups by the scheme 2(i) – 2(iii)
gives m groups

propagation time of a carry signal $\le mT$
the maximum propagation time = mT

the maximum delay = D
the optimal group size = m

$$(m-1)T \le D \le mT$$

(I) Let m be the smallest positive integer such that

$$n \le m + \frac{1}{2}mT + \frac{1}{4}m^2T + (1-(-1)^m)\frac{1}{8}T$$

(II) Let $y_i = min\{1+iT, 1+(m+1-i)T\}$, $i = 1, \dots, m$

and construct a histogram
whose i-th column has height y_i

(III) the area of the histogram in (II) is

$$m + \frac{1}{2}mT + \frac{1}{4}m^2T + (1-(-1)^m)\frac{1}{8}T \ge n$$

so these are at least n unit squares
in the histogram starting with the first row,
shade in n of the squares, row by row
Let x_i denote the number of shaded squares
in column i of the histogram,
$i = 1, \dots, m$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (1-2)

Assume

- the scheme by 2(i) – 2(iii) (m groups) is <u>not</u> optimal
- let ***D*** be the maximum delay
  corresponding to an optimal division of the bits into groups
- there are ***r*** groups in the optimal division.

Since a carry in signal to the least significant bit group
can skip over each group

we have $rT \le D \le mT$ so $r \le m$

if m is <u>not</u> optimal, <u>but</u> r is *then $mT \ge rT$* (smaller delay rT) *thus $m \ge r$* (smaller r exists)

m groups
- *<u>not</u> optimal division*
- ***D*** = maximum delay
- ***m****T* skip delay

r groups
- optimal division
- ***r****T* skip delay

skip delay
$rT \le D \le mT$

$r \le m$

D *= max delay is assumed*
To be greater than all skip
delay ***rT*** *of the optimal division*

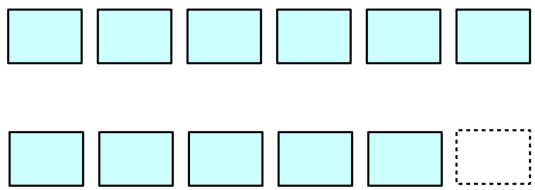
Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (1-2)

If the optimal division gives *m* groups

m groups — *mT*

(*m*-1) groups — (*m*-1)*T*



$$D \le mT$$

$$(m-1)T \le D$$

- when optimal group size = m
  the maximum delay $D_m \le mT$
- when optimal group size = (*m*-1)
  the maximum delay $D_{m-1} \le (m-1)T$

Normally, by 2(i) – 2(iii) (*m* groups) is optimal
and its maximum delay *D* is less than all skip delay *mT*

$D \le mT$

D = maximum delay

To prove this, first, negate that

- *m is <u>not</u> by the optimal division, but r is*
- *D is <u>greater</u> than all skip delay of the optimal division*

$rT \le D \le mT$

$r \le m$ ➡ $r < (m-1)$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (1-2)

$rT \le D \le mT$ so $r \le m$

Optimal division : r groups

D' ≤ all skip delay rT (r groups)

Non-optimal division : m groups

D ≤ all skip delay mT (m groups)

too many partitions m $r \le m$

Assume max delay D is greater than all skip delay rT of the optimal division

if m is <u>not</u> optimal, <u>but</u> r is
then $mT \ge rT$ (smaller delay rT)
thus $m \ge r$ (smaller r exists)

D is max delay for m groups
D' is max delay for r groups
then $D' \le rT \le D \le mT$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

D = maximum delay

$rT \le D \le mT$

$r \le m$ ➡ $r < (m-1)$

Optimal division into groups (2)

we have $rT \le D \le mT$ so $r \le m$

If $r = m$

then $D = mT$ $\Rightarrow$ $D = rT$ $\qquad$ $rT = D$

If $r = m\text{-}1$, ($r < m$)

$D \ge (m\text{-}1)T$ $\Rightarrow$ $D \ge (m\text{-}1)T = rT$ $\qquad$ $rT \le D$

if $r < m\text{-}1$, ($r < m$) $\qquad$ $r = m\text{-}2,\ m\text{-}3,\ m\text{-}3,\ \ldots$

$D \ge (m\text{-}1)T$ $\Rightarrow$ $D \ge (m\text{-}1)T > rT$ $\qquad$ $rT < D$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (3)

we have $rT \le D \le mT$ so $r \le m$

If $r = m$ then $D = mT$
and the **theorem** holds by **lemma** 1

When the bits of a carry skip adder are grouped according to the scheme (i)-(iii), the maximum propagation time of a carry signal is mT

$(m-1)T \le D \le mT$

Lemma 1 *When the bits of a carry skip adder are* *<u>grouped</u>* *according to the scheme (i)-(iii), the maximum propagation time of a carry signal is mT*

Theorem 1 *The scheme 2(i) – 2(iii) given above for dividing the bits of a carry skip adder into groups is optimal for* $2 \le T \le 7$

(5) $r=2k$ $\quad X = 4 - T^2$

$$mT - D \le T + \frac{-8(T/n)+4}{\sqrt{4(T/n)+8(T/n^2)} + \sqrt{4(T/n)+4/n^2}}$$

$r=2k+1$ $\quad X = 4$

$$mT - D \le T + \frac{(T-2)^2/n}{\sqrt{4(T/n)+4(T/n^2)} + \sqrt{4(T/n)+(T/n)^2+4/n^2}}$$

m groups – not optimal division
r groups – optimal division

D = maximum delay

$rT \le D \le mT$

$r \le m$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (3)

If $r = m-1$, ($r < m$)
m and r have different parities and
it follows from (5)
that $mT - D \leq T$ for $2 \leq T \leq 7$

so that $D \geq (m-1)T$
since $r = m-1$,
$D \geq (m-1)T = rT$ $rT \leq D$

This means that a signal which
skips over each of the r groups (rT)
has delay less than the maximum D.

$rT \leq D \leq mT$

m is not optimal division
r is optimal division

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Lemma 1 *When the bits of a carry skip adder are grouped according to the scheme (i)-(iii), the maximum propagation time of a carry signal is mT*

Theorem 1 *The scheme 2(i) – 2(iii) given above for dividing the bits of a carry skip adder into groups is optimal for* $2 \leq T \leq 7$

(5) $r = 2k$ $X = 4 - T^2$

$$mT - D \leq T + \frac{-8(T/n)+4}{\sqrt{4(T/n)+8(T/n^2)} + \sqrt{4(T/n)+4/n^2}}$$

$r = 2k+1$ $X = 4$

$$mT - D \leq T + \frac{(T-2)^2/n}{\sqrt{4(T/n)+4(T/n^2)} + \sqrt{4(T/n)+(T/n)^2+4/n^2}}$$

m groups – not optimal division
r groups – optimal division

D = maximum delay

$rT \leq D \leq mT$

$r \leq m$

Optimal division into groups (4)

Similarly,
if $r < m\text{-}1$, ($r < m$)

$(m\text{-}1)T \le D$

since $r < m\text{-}1$,
$rT < (m\text{-}1)T \le D$

so that a signal which skips over each group
has delay $rT < D$.

$rT < D \le mT$

m is <u>not</u> optimal division
r is optimal division

m groups – not optimal division
r groups – optimal division

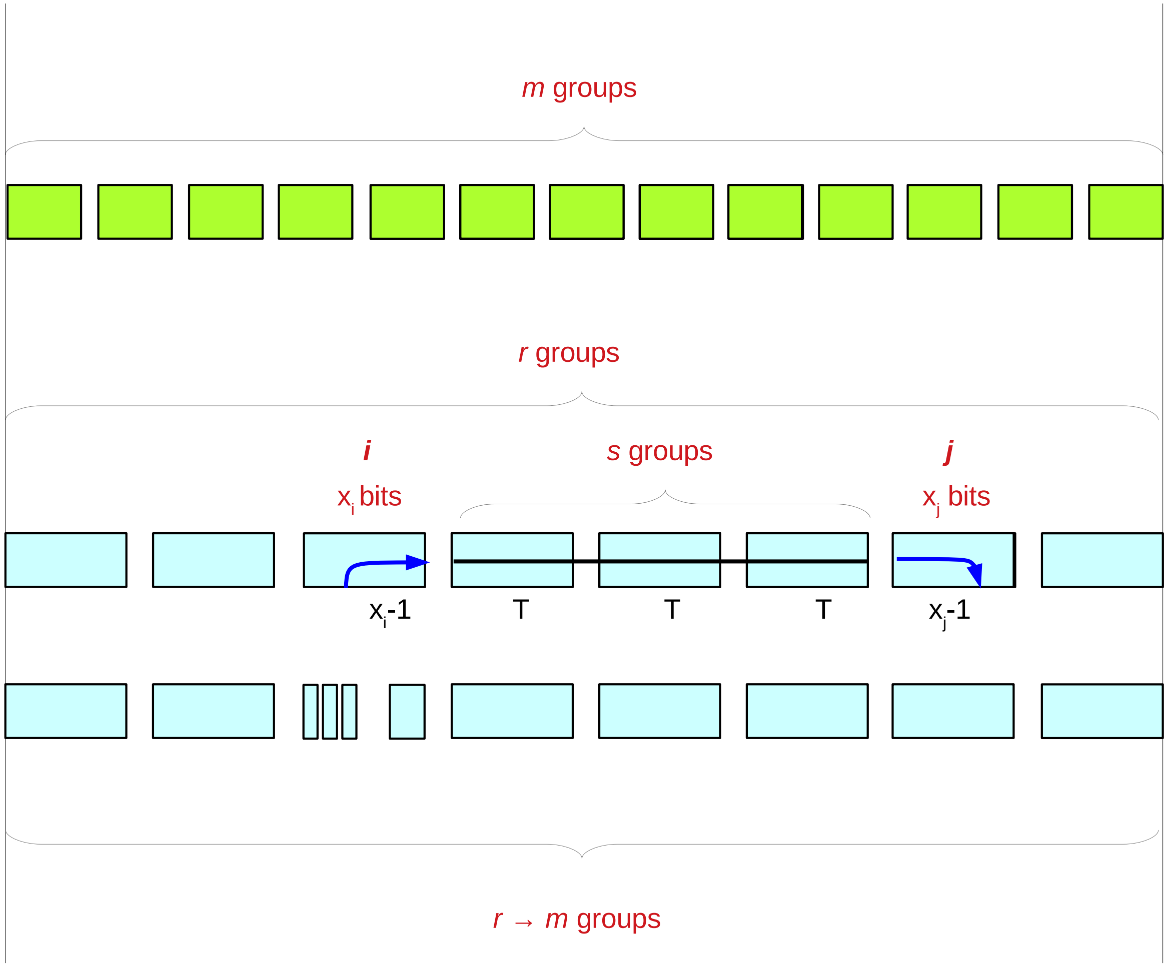
D = maximum delay

$rT \le D \le mT$

$r \le m$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (5)



m groups – not optimal division
r groups – optimal division

D = maximum delay

$rT \le D \le mT$

$r \le m$

if m is not optimal, but r is

$(((r\ \ +1)\ \ +1)\ \ +1)\ \ \ldots \longrightarrow m$

contradiction! *r must be m*

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (5)

It follows that a signal with delay *D*

must <u>start</u> in a group *i*,
<u>ripple</u> to the <u>end</u> of group *i*,

then <u>skip</u> over *s* < *r* groups and

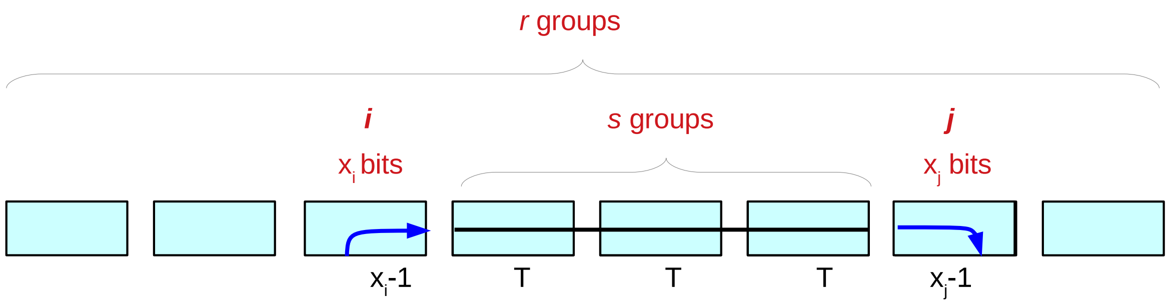
either <u>terminate</u>, or <u>ripple</u> through
the first few bits of a group *j* > *i*.

m groups – not optimal division
r groups – optimal division

D = maximum delay

$rT \le D \le mT$

$r \le m$

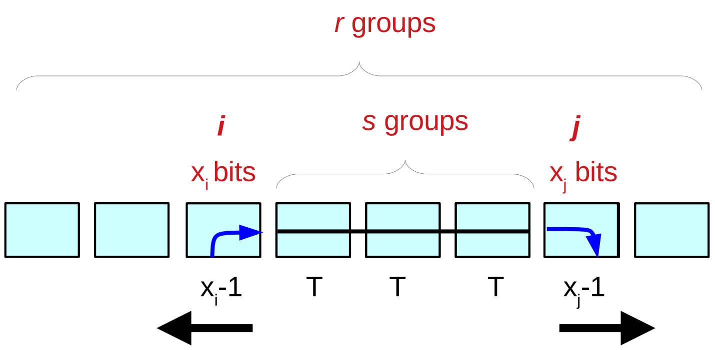


Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (6-1)

Let x_i and x_j denote
the lengths of the *i-th* and *j-th* groups respectively.

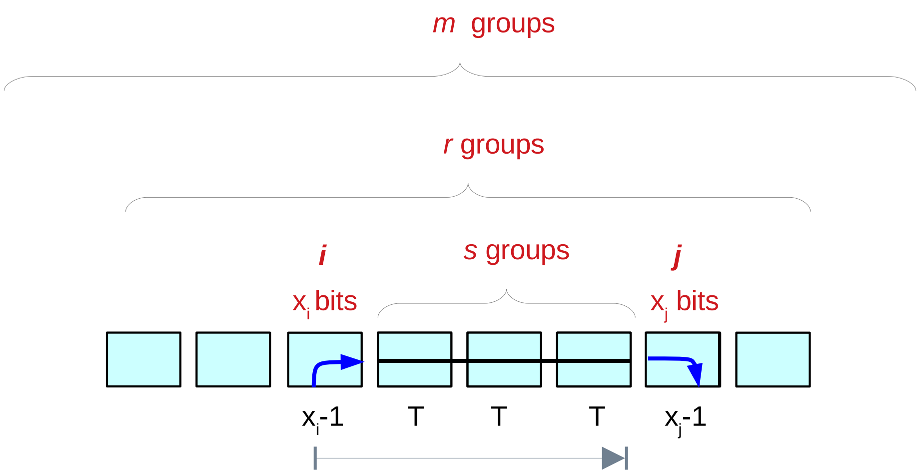
Assume that *i* is chosen as small as possible
and *j* as large as possible. (longer path)



Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (6-2)

A signal originating in group i, rippling to the end of this group i and then skipping over the next s group has delay $(x_i - 1) + sT$



$$D \le (x_i - 1) + sT$$
$$\le (x_i - 1) + (r-1)T$$
$$\le (x_i - 1) + (m-2)T.$$

$s < r$ groups ⇒ $s \le (r-1)$

$r < m$ groups ⇒ $r \le (m-1)$

$$D \le (x_i - 1) + sT$$
$$< (x_i - 1) + rT$$
$$< (x_i - 1) + mT.$$

if m is not optimal, but r is

$$s < r < m$$
$$s \le (r-1) < (m-1)$$
$$s \le (r-1) \le (m-2)$$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (6-3)

$$D \le (x_i - 1) + sT$$
$$\le (x_i - 1) + (r - 1)T$$
$$\le (x_i - 1) + (m - 2)T$$

$$(m-1)T \le D$$
$$D \le (x_i - 1) + (m - 2)T$$

$$(m-1)T \le D \le (x_i - 1) + (m - 2)T$$

$$(m-1)T \le (x_i - 1) + (m - 2)T$$

Since $D \geq (m\text{-}1)T$
this implies that $x_i \geq T+1$

$$T \le (x_i - 1)$$

$$T + 1 \le x_i$$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (7)

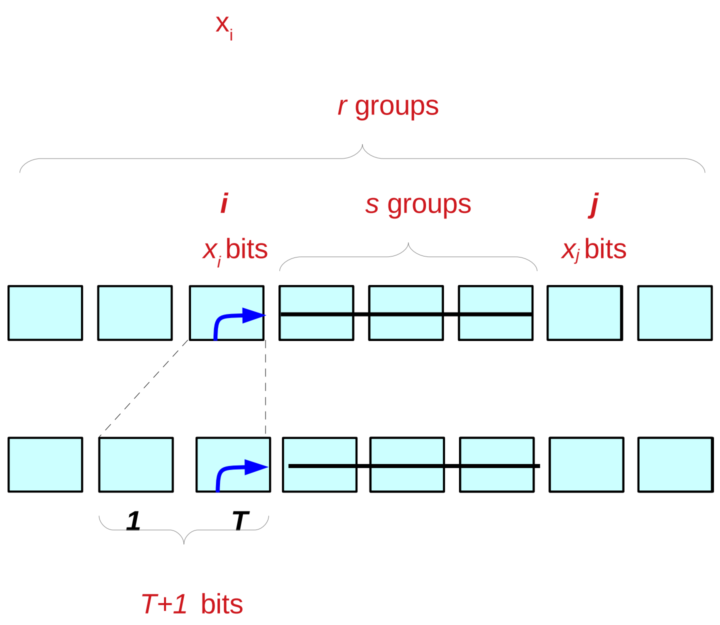
Divide group *i* into two groups
such that the group containing the msb has size *T*.

Since the *i*-th group is the first group
in which a signal having maximum delay can originate,

this subdivision does not increase
the delay of any carry signal of maximum delay

However, it increases the number of groups by **1**

$x_i \geq T+1$ *bits size*



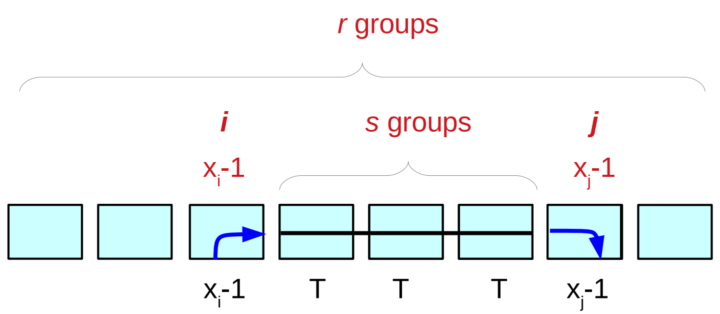
$$D \leq (x_i - 1) + sT$$
$$\leq (x_i - 1) + (r-1)T$$
$$\leq (x_i - 1) + (m-2)T.$$

$$(m-1)T < D$$
$$D < (x_i - 1) + (m-2)T$$
$$x_i \geq (T+1)$$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (8-1)

Suppose now that a carry signal originates in a group *i*, ripples to its end, skips over $s \le r-2$ groups and finally ripples through the first few bits of a group *j* and terminates.



We then have

$$D \le (x_i - 1) + sT + (x_j - 1)$$
$$\le x_i + x_j - 2 + (m-3)T$$

So that either $x_i \ge T+1$ or $x_j \ge T+1$

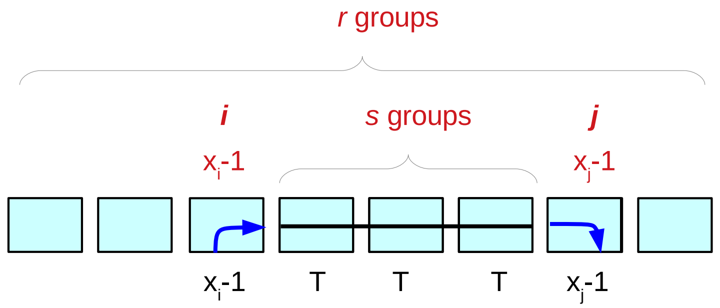
$s < r$ groups	$s < r$ groups
$s \le (r-1)$ groups	$s \le (r-2)$ groups
$r < m$ groups	$r < m$ groups
$r \le (m-1)$ groups	$r \le (m-2)$ groups

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (8-2)

$$s < r < m$$
$$s \le (r-1) < (m-1)$$
$$s \le (r-1) \le (m-2)$$

$$s \le (r-2) \le (m-3)$$



$$s \le (r-2)$$

$$\begin{aligned} D &\le (x_i - 1) + sT + (x_j - 1) \\ &\le x_i + x_j - 2 + (r-2)T \\ &\le x_i + x_j - 2 + (m-3)T \end{aligned}$$

$$(m-1)T < D$$
$$D < (x_i - 1) + (m-2)T \iff x_i \ge (T+1)$$
$$D < (x_j - 1) + (m-2)T \iff x_j \ge (T+1)$$

So that either $x_i \ge T+1$ or $x_j \ge T+1$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

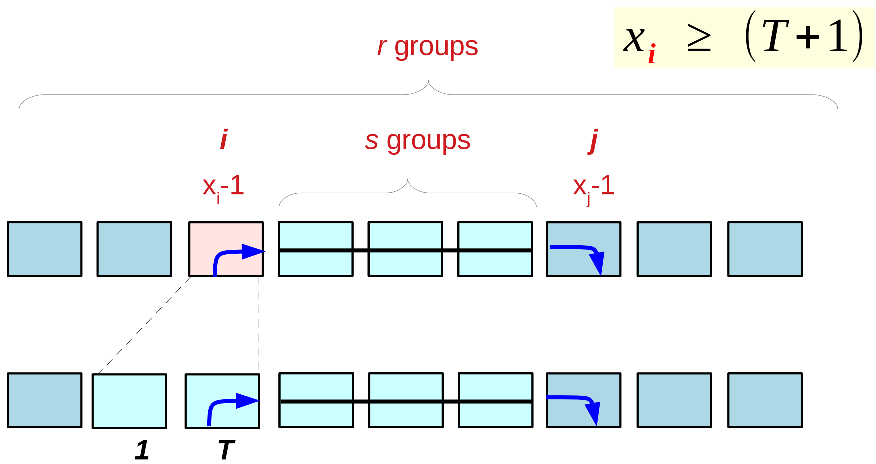
Optimal division into groups (9)

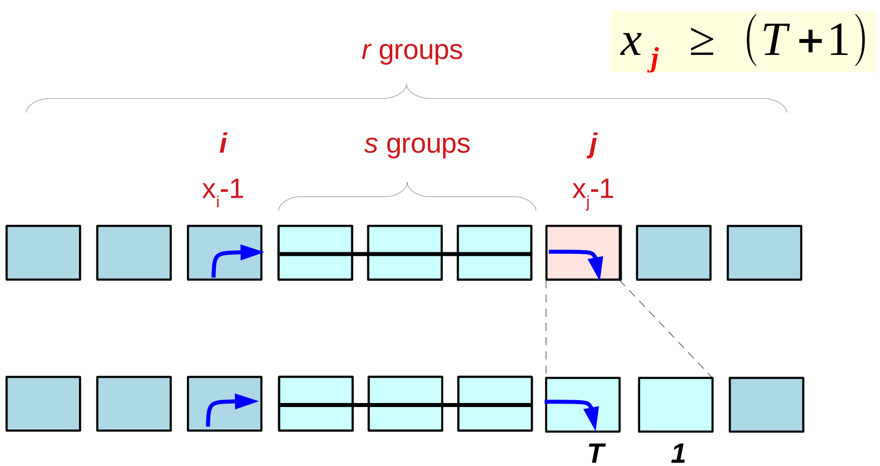
So that either $x_i \geq T+1$ or $x_j \geq T+1$

This means that we can subdivide
one of the groups i, j
without increasing D
not both of them

Continuing in this way, we can always increase
the number r of group in an optimal division
of a carry chain by 1
without increasing D if $r < m$

This means that we can arrive
at an optimal division of the carry chain
into m groups.





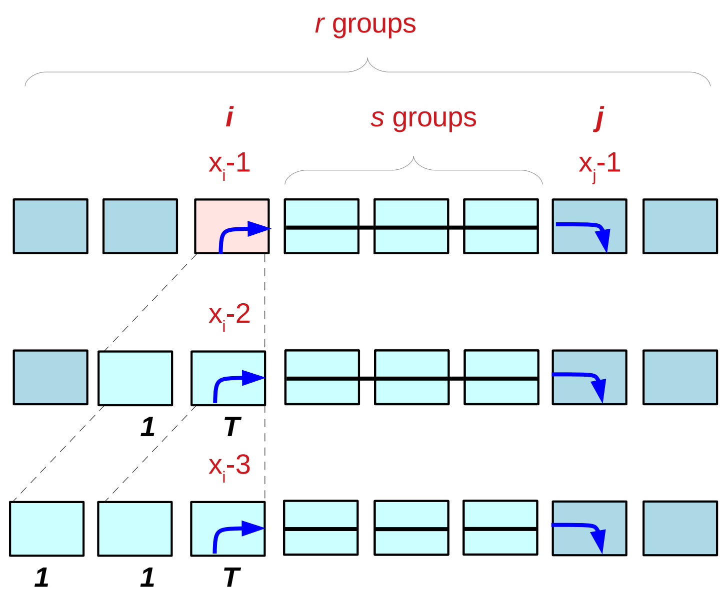
Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (9)

$x_{i} \geq (T+1) > (T+2) \cdots$

$x_{i} \geq (T+1)$

$x_{i} \geq (T+2)$



$$D \leq (x_i - 1) + sT$$
$$\leq (x_i - 1) + (r-1)T$$
$$\leq (x_i - 1) + (m-2)T$$

$$(m-1)T \leq D \leq (x_i - 1) + (m-2)T$$

if m is not optimal, but r is

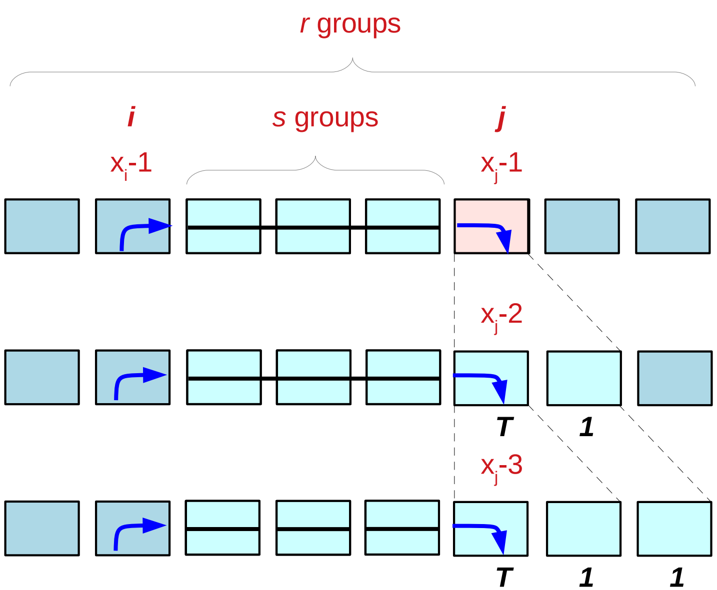
(((r +1) +1) +1) … → m

contradiction! m must be r

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (9)

$$x_j \geq (T+1) > (T+2) \cdots$$



$$x_j \geq (T+1)$$

$$x_j \geq (T+2)$$

if m *is* ***not*** *optimal,* ***but*** r *is*

(((r +1) +1) +1) … → m

contradiction! m *must be* r

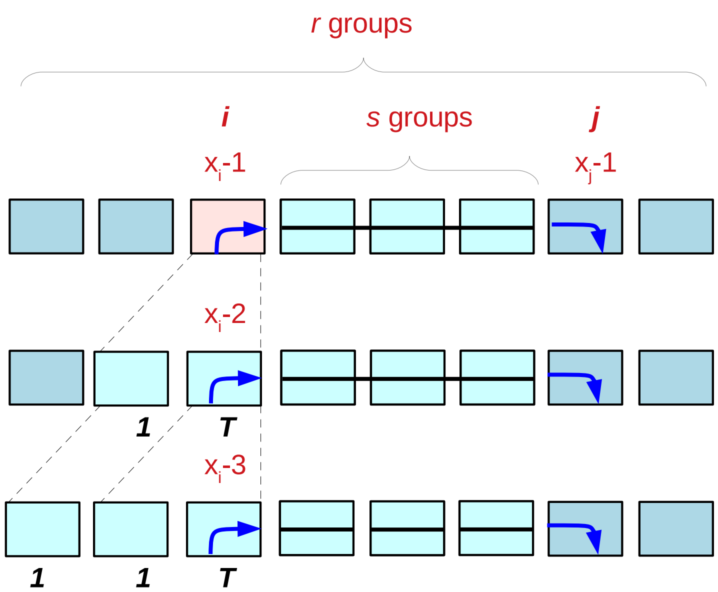
$$D \leq (x_i - 1) + sT$$
$$\leq (x_i - 1) + (r-1)T$$
$$\leq (x_i - 1) + (m-2)T$$

$$(m-1)T \leq D \leq (x_i - 1) + (m-2)T$$

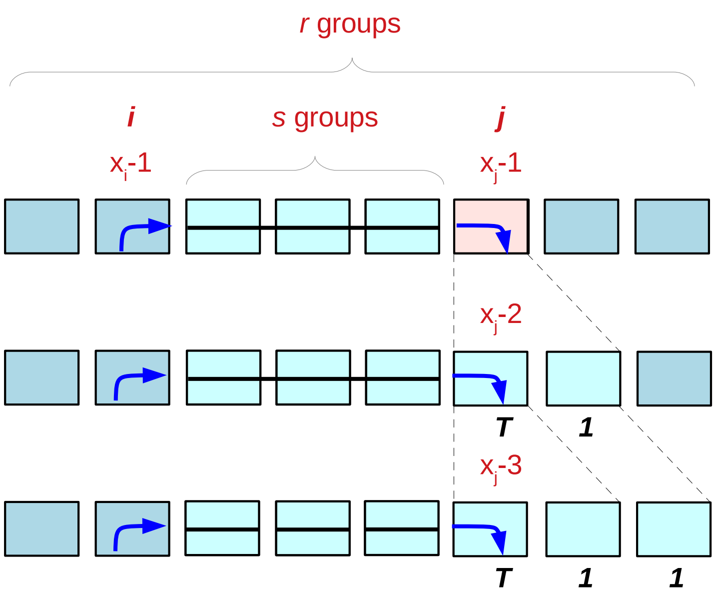
Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (9)

$$x_i \geq (T+1) > (T+2) \cdots$$



$$x_j \geq (T+1) > (T+2) \cdots$$



if m is not optimal, but r is

(((*r* +1) +1) +1) … → *m*

contradiction! m must be r

if m is not optimal, but r is

(((*r* +1) +1) +1) … → *m*

contradiction! m must be r

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (9)

Normally, by 2(i) – 2(iii) (*m* groups) is optimal
and its maximum delay *D* is less than all skip delay *mT*

$D \le mT$

To prove this, first, negate that

- *m is not by the optimal division, but r is*
- *D is greater than all skip delay of the optimal division*

Assume

- the scheme by 2(i) – 2(iii) (*m* groups) is not optimal
- let ***D*** be the maximum delay corresponding to an optimal division
- there are ***r*** groups in the optimal division.

(...(((r+1)+1)+1) … +1) → m : optimal

if m is not optimal, but r is

$(((r\ +1)\ +1)\ +1)\ \ldots \longrightarrow m$

contradiction! m must be r

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers

Optimal division into groups (11)

We must then have $D \geq mT$
which, together with **Lemma 2**,
Implies $D = mT$

This completes the proof of the theorem

m groups – not optimal division
r groups – optimal division

D = maximum delay

$rT \leq D \leq mT$

$r \leq m$

Lemma 2
Let D denote the maximum delay
of a carry signal in a n bit carry skip adder
with group sizes chosen optimally.

$$(m-1)T \leq D \leq mT$$

Theorem 1

The scheme 2(i) – 2(iii) given above
for dividing the bits of a carry skip adder into groups
is optimal for $2 \leq T \leq 7$

Oklobdzija: High-Speed VLSI arithmetic units : adders and multipliers