
Chapter 7

Sub-System Design

Full-Adder

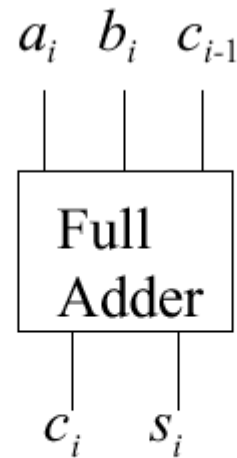


Fig. 7.1 One-bit full-adder.

Ripple Carry Adder

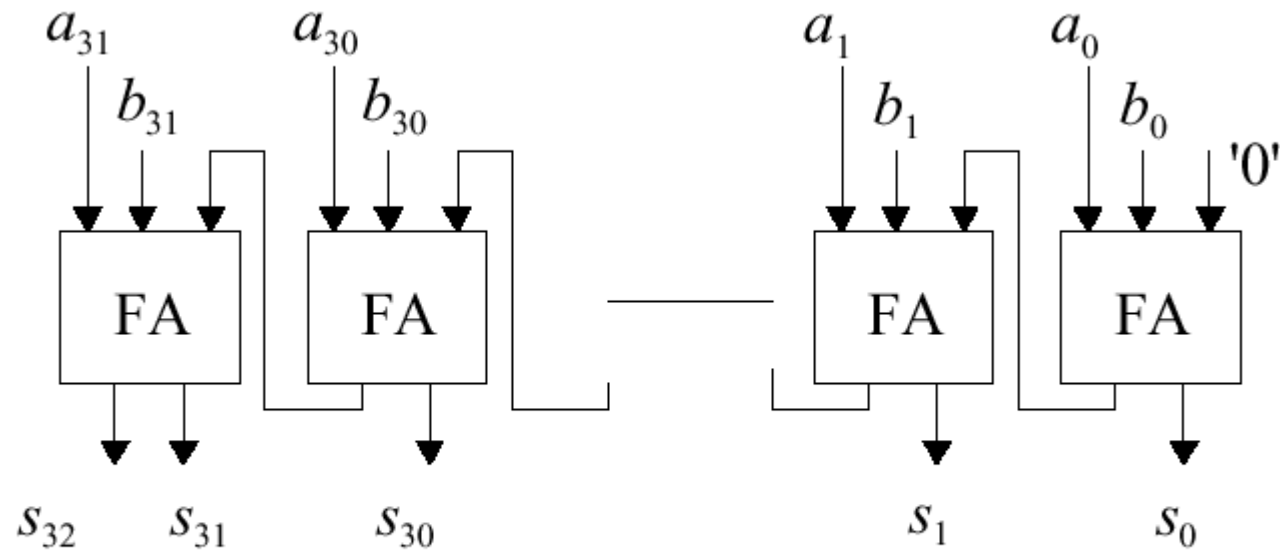


Fig. 7.2 Ripple carry adder.

Full Adder Tessellation

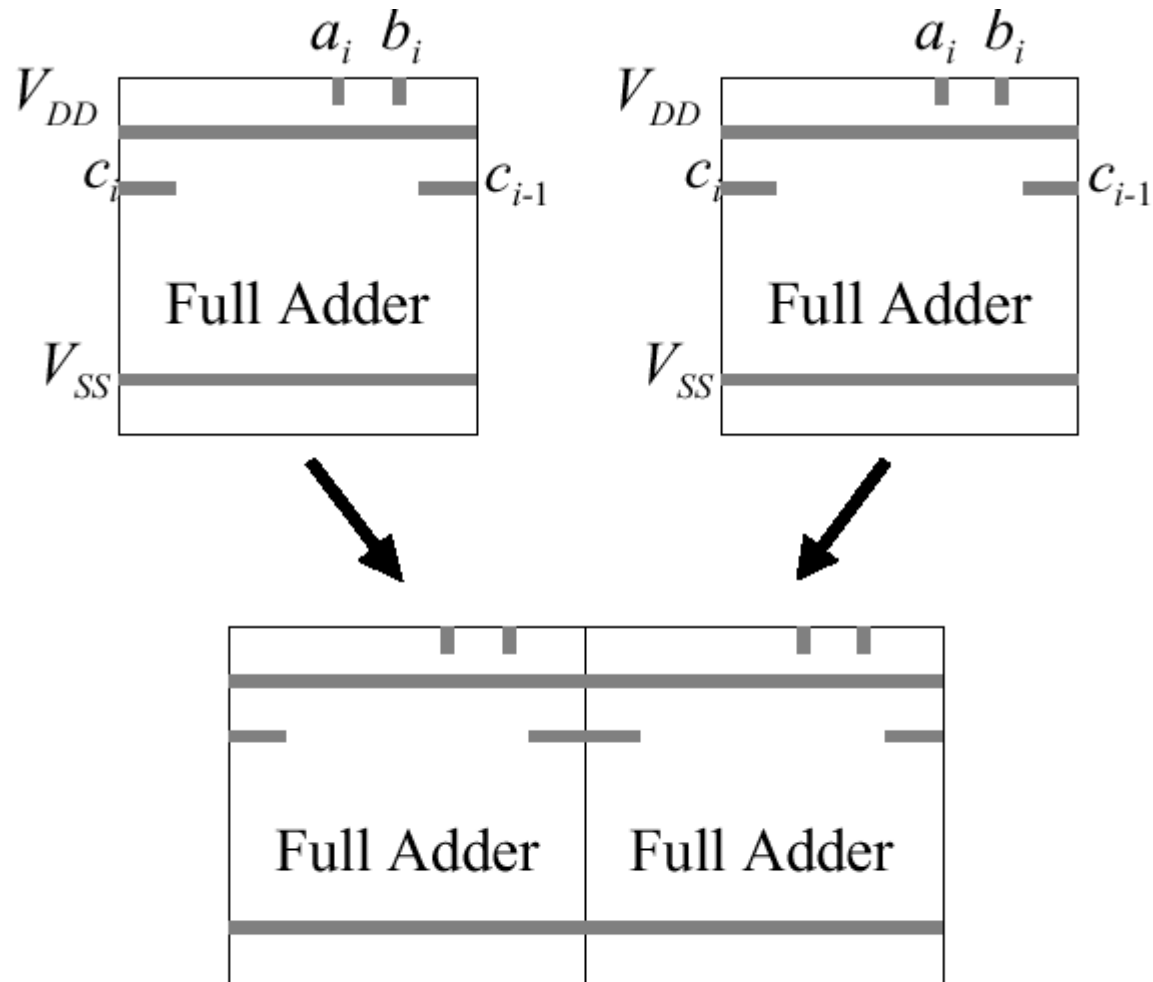


Fig. 7.3 Full adder tessellation.

Carry Select Adder

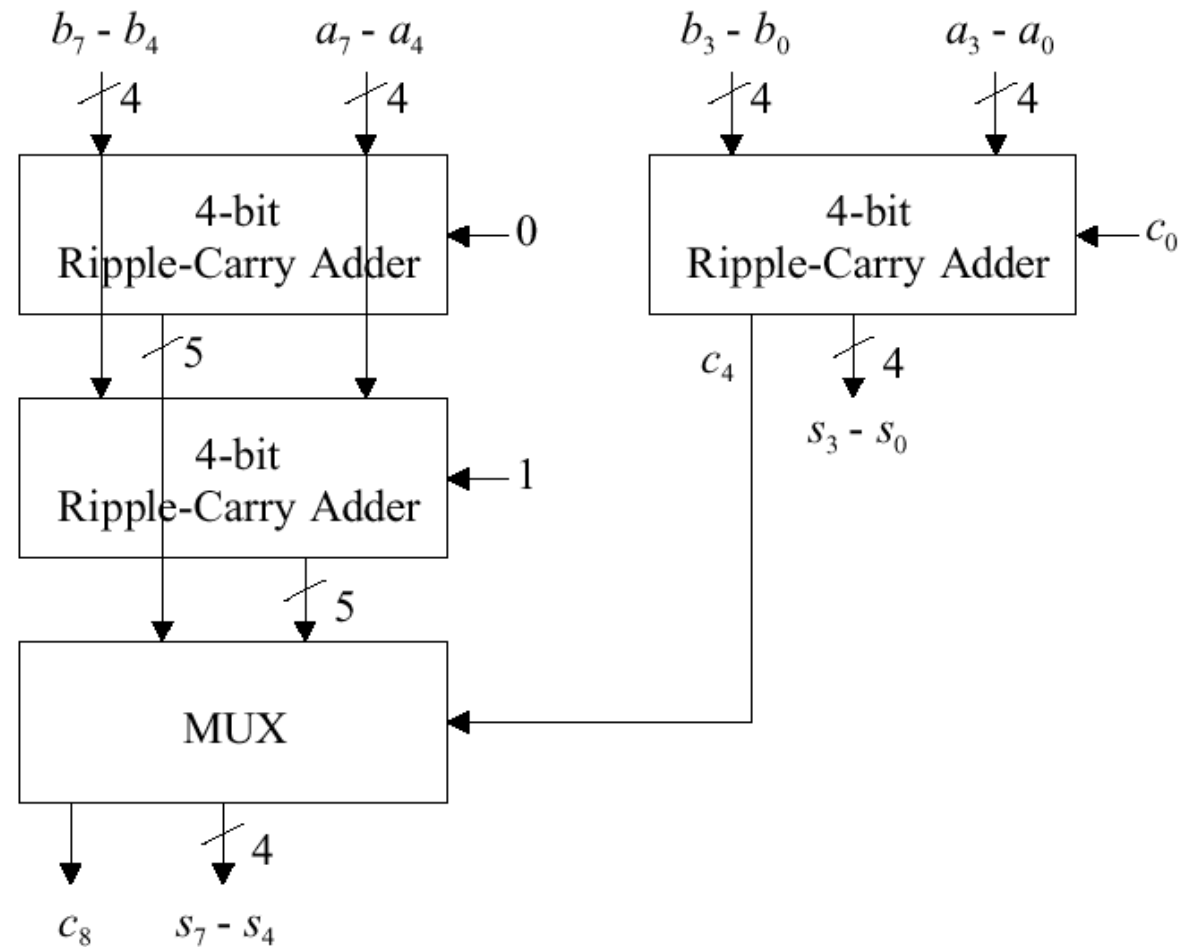


Fig. 7.4 8-bit carry select adder.

Conditional Sum Adder

	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$A =$	1	1	0	1	1	0	1
$B =$	0	1	1	0	1	1	0
S_i^0	1	0	1	1	0	1	1
C_i^0	0	1	0	0	1	0	0
S_i^1	0	1	0	0	1	0	0
C_i^1	1	1	1	1	1	1	1

Fig. 7.5 Conditional sums and conditional carries.

Conditional Sum Adder

	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$A =$	1	1	0	1	1	0	1
$B =$	0	1	1	0	1	1	0
S_i^0	1	0	1	1	0	1	1
C_i^0	0	1	0	0	1	0	0
S_i^1	0	1	0	0	1	0	0
C_i^1	1	1	1	1	1	1	1

$c_{-1} = 0$

Fig. 7.6 Addition with conditional sums and conditional carries.

Conditional Sum Adder

	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$A =$	1	1	0	1	1	0	1
$B =$	0	1	1	0	1	1	0
S_i^0	0	0	1	1	0	1	1
C_i^0	1		0		1		0
S_i^1	0	1	0	0	1	0	0
C_i^1	1		1		1		1

Fig. 7.7 Two-bit conditional sums and conditional carries.

Conditional Sum Adder

	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$A =$	1	1	0	1	1	0	1
$B =$	0	1	1	0	1	1	0
S_i^0	0	0	1	1	0	1	1
C_i^0	1		0		1		0
S_i^1	0	1	0	0	1	0	0
C_i^1	1		1		1		1

$c_{-1} = 0$

Fig. 7.8 Addition with 2-bit conditional sums and conditional carries.

Conditional Sum Adder

	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$A =$	1	1	0	1	1	0	1
$B =$	0	1	1	0	1	1	0
$S_i^0(0)$	1	0	1	1	0	1	1
$C_i^0(0)$	0	1	0	0	1	0	0
$S_i^1(0)$	0	1	0	0	1	0	0
$C_i^1(0)$	1	1	1	1	1	1	1
$S_i^0(1)$	0	0	1	1	0	1	
$C_i^0(1)$	1		0		1		
$S_i^1(1)$	0	1	0	0	1	0	
$C_i^1(1)$	1		1		1		

Fig. 7.9 Parallel determination of 2-bit conditional sums and conditional carries.

Conditional Sum Adder

	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
$A =$	1	1	0	1	1	0	1
$B =$	0	1	1	0	1	1	0
$S_i^0(0)$	1	0	1	1	0	1	1
$C_i^0(0)$	0	1	0	0	1	0	0
$S_i^1(0)$	0	1	0	0	1	0	0
$C_i^1(0)$	1	1	1	1	1	1	1
$S_i^0(1)$	0	0	1	1	0	1	
$C_i^0(1)$	1		0		1		
$S_i^1(1)$	0	1	0	0	1	0	
$C_i^1(1)$	1		1		1		
$S_i^0(2)$	0	0	1	1			
$C_i^0(2)$	1						
$S_i^0(2)$	0	1	0	0			
$C_i^0(2)$	1						

$c_{-1} = 0$

Fig. 7.10 Three levels of conditional sums and conditional carries.

Conditional Sum Adder

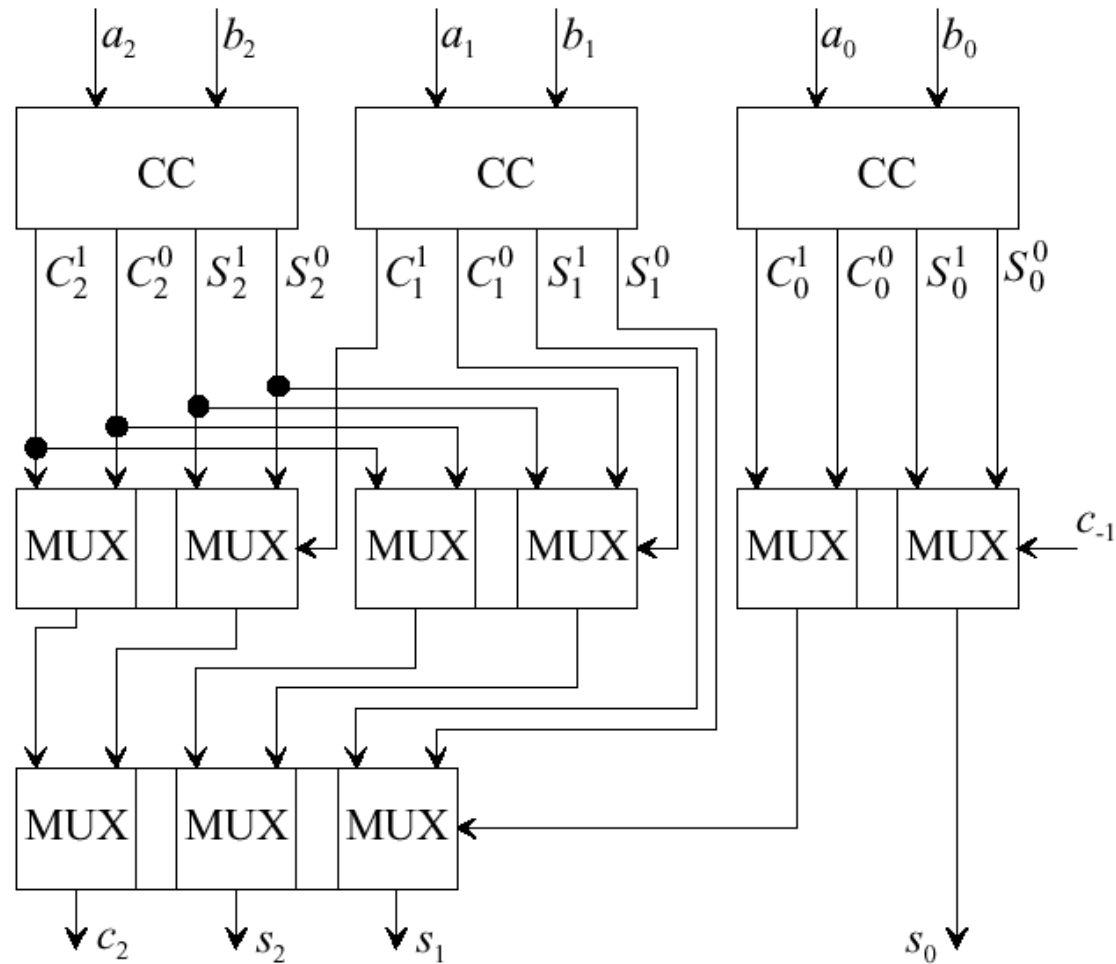


Fig. 7.11 Two-level three-bit conditional sum adder.

Circulating Adder

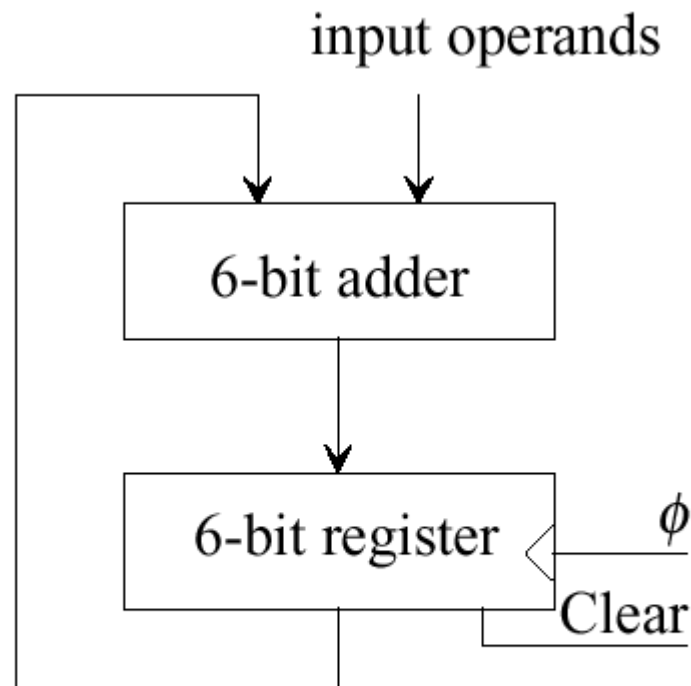


Fig. 7.12 Circulating adder.

Full Adder Tree

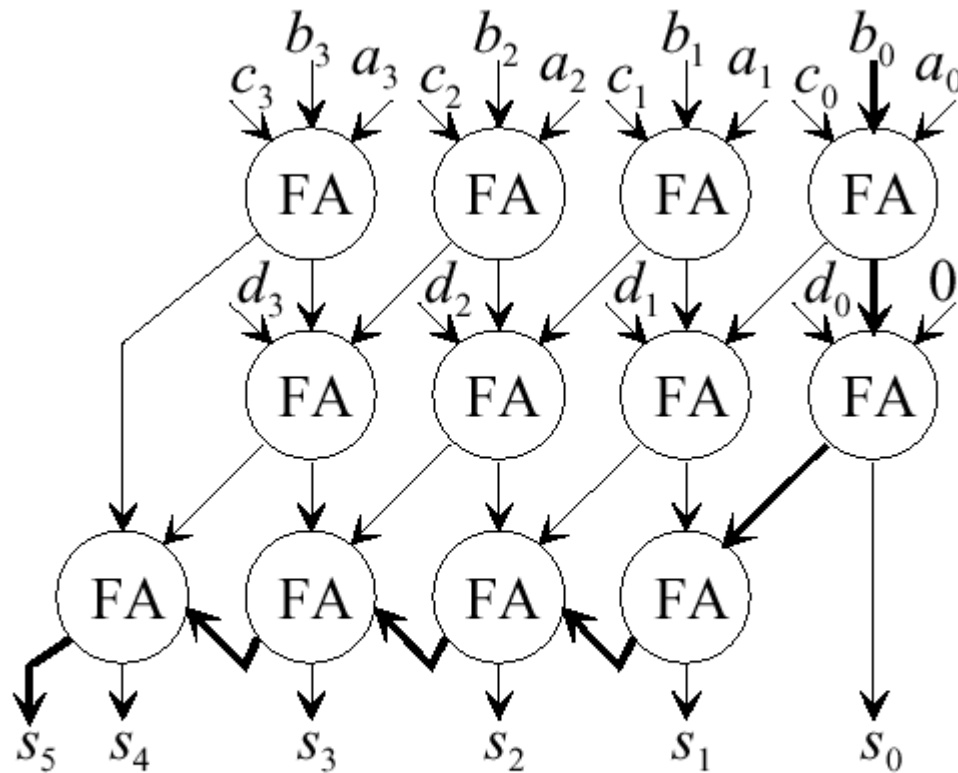


Fig. 7.13 Full adder tree.

Braun Array Multiplier

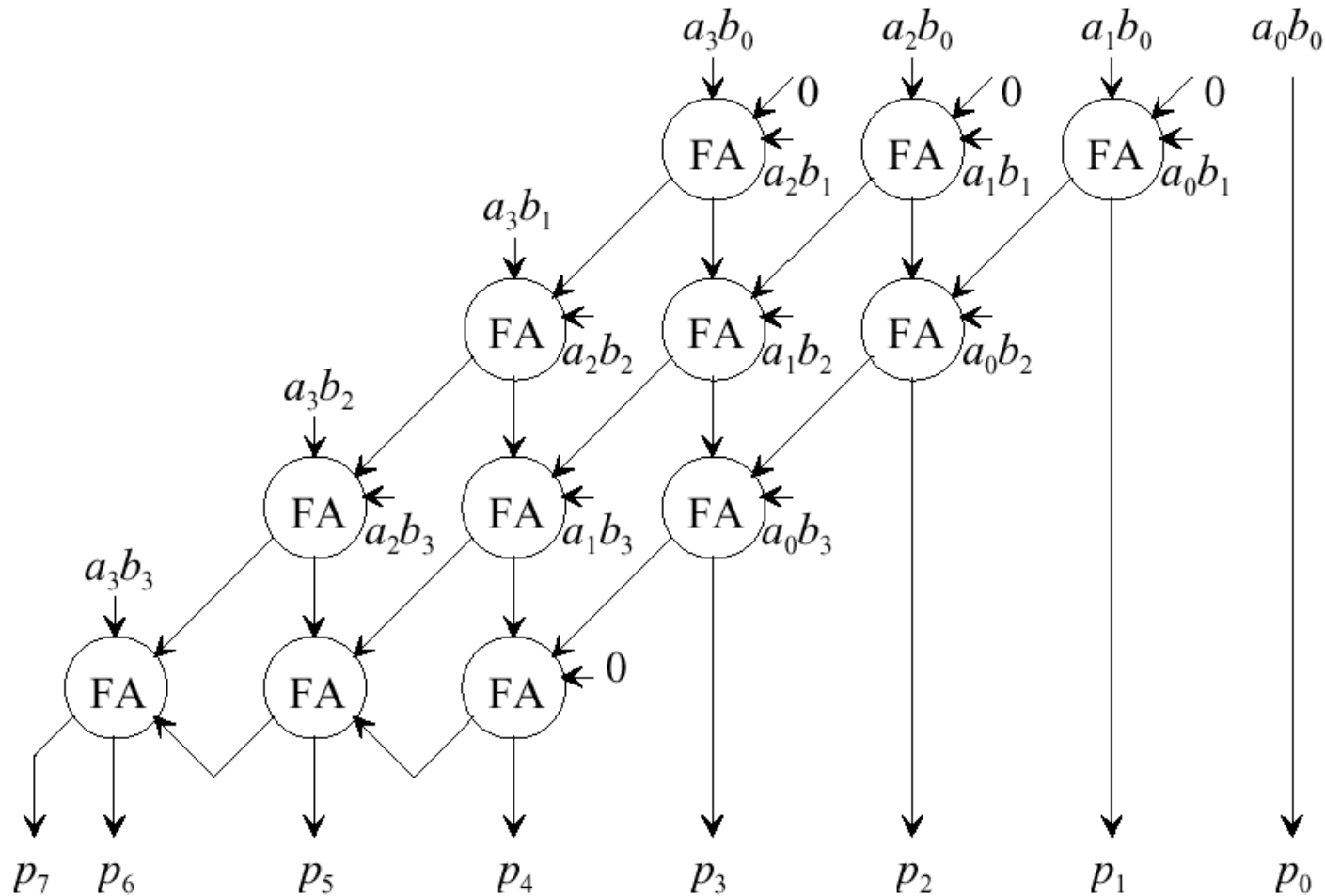


Fig. 7.14 Braun array multiplier.

Two's Complement Multiplication

$$\begin{array}{r} 1101 = -3 \\ \text{X) } 0101 = 5 \\ \hline 0101 \\ 1000 \\ 0101 \\ 0111 \\ \hline 01100001 \\ +) 10010000 \quad \text{adjustment} \\ \hline 11110001 = -15 \end{array}$$

Fig. 7.15 Two's complement multiplication with an unsigned multiplier.

Partial Product Generation

$y_{i+1}y_iy_{i-1}$	Encoded Digit	Partial Product Generation
000	0	$0 \times A$
001	+1	$+1 \times A$
010	+1	$+1 \times A$
011	+2	$+2 \times A$
100	-2	$-2 \times A$
101	-1	$-1 \times A$
110	-1	$-1 \times A$
111	0	$0 \times A$

Fig. 7.16 Partial product generation

Modified Booth Multiplication

$$\begin{array}{r}
 10110101 = -75 \\
 01010110 = +86 \\
 \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \end{array} \\
 \hline
 \begin{array}{r}
 0000000010010110 \quad 100:-2A \\
 11111101101010 \quad 011:+2A \\
 111110110101 \quad 010:+1A \\
 1110110101 \quad 010:+1A \\
 \hline
 1110011011001110 = -6450
 \end{array}
 \end{array}$$

Fig. 7.17 Modified Booth multiplication.

Sign Extension

```
          aaaaaaaa
          bbbbbbbb
          -----
SSSSSSSS|S*****
SSSSSS|S*****
SSSS|S*****
SS|S*****
S|S*****
-----
*****
```

Fig. 7.18 Sign extension in modified Booth multiplication.

Correction Vector

$$\begin{array}{r} 11111111 \\ 111111 \\ 1111 \\ 11 \\ \hline 10101011 \end{array}$$

Fig. 7.19 Correction vector production.

Correction Vector

$$\begin{array}{r}
 10110101 = -75 \\
 01010110 = +86 \\
 \begin{array}{cc}
 \boxed{} & \boxed{} \\
 \boxed{} & \boxed{}
 \end{array} \\
 \hline
 \begin{array}{rcl}
 110010110 & 100: & -2A \\
 001101010 & 011: & +2A \\
 010110101 & 010: & +1A \\
 010110101 & 010: & +1A \\
 \hline
 0011101111001110 & & \\
 10101011 & & \\
 \hline
 1110011011001110 & = & -6450
 \end{array}
 \end{array}$$

Fig. 7.20 Adding of correction vector.

Wallace Multiplier

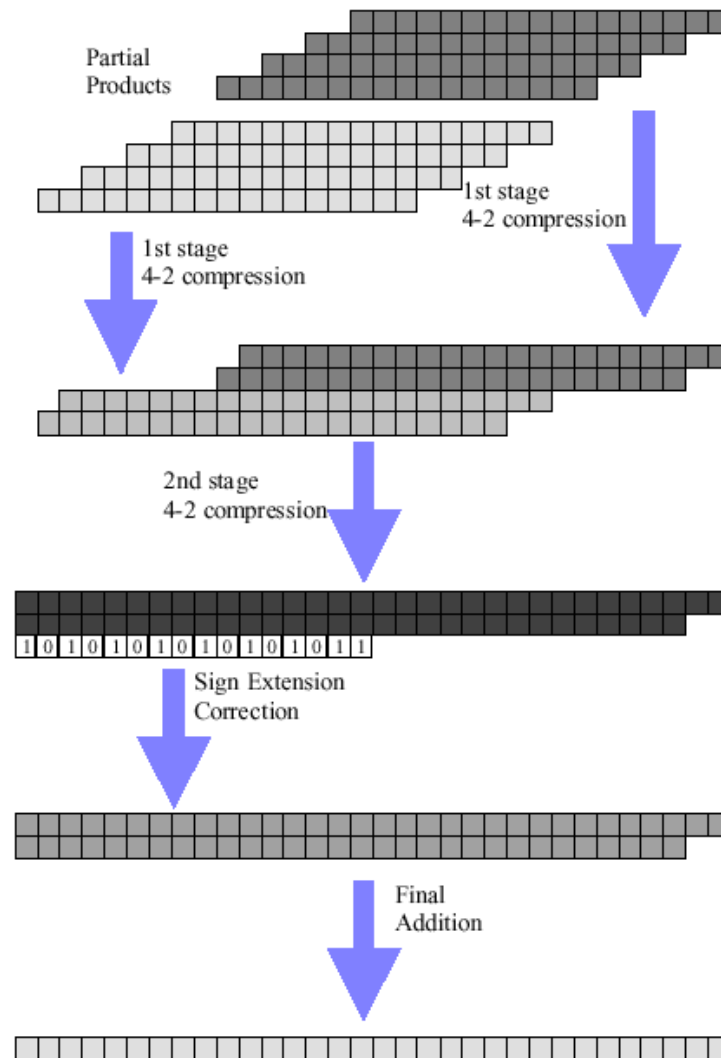


Fig. 7.21 Wallace tree multiplication.

4-2 Compressor

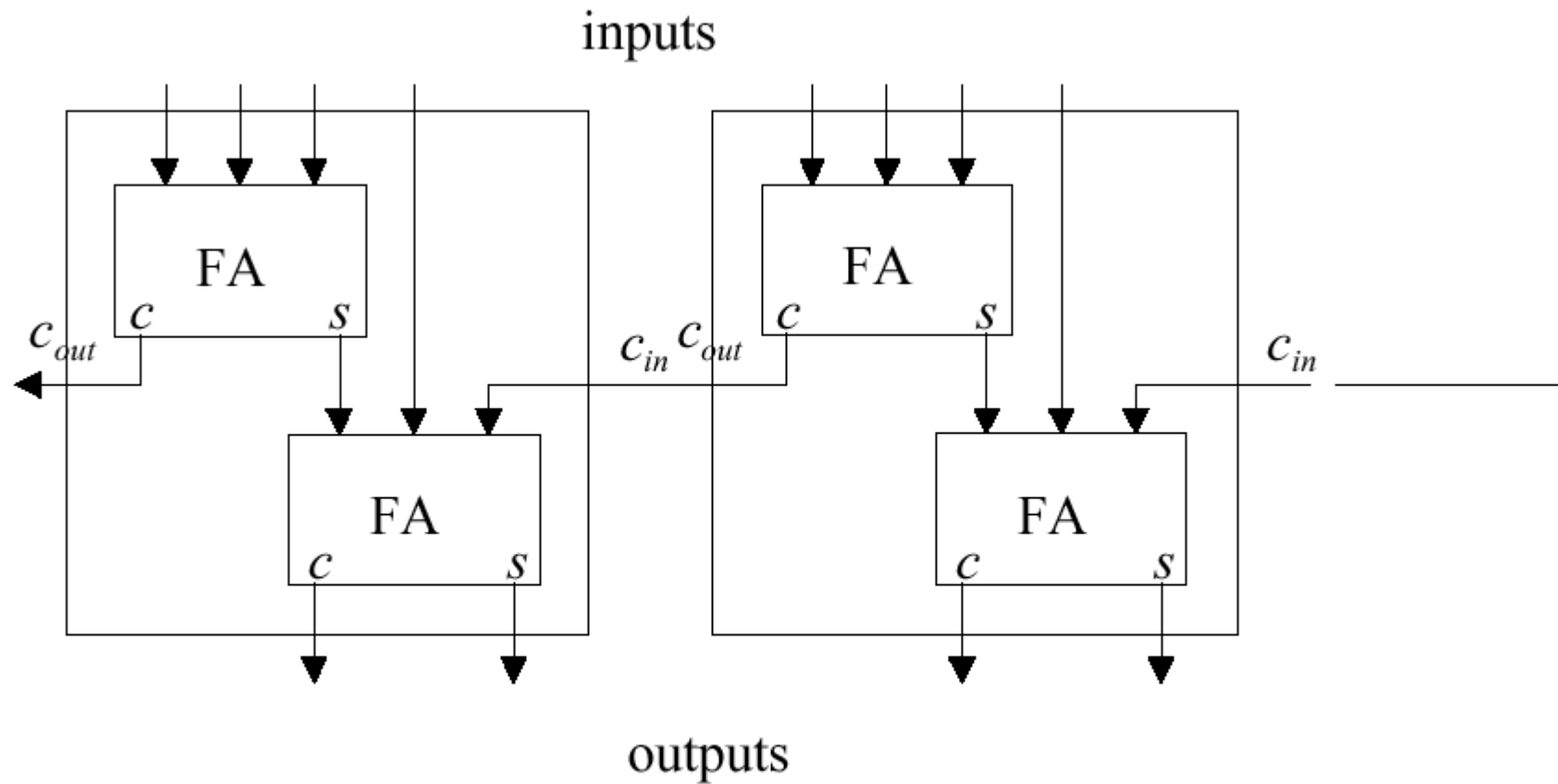


Fig. 7.22 4-2 compressor.

Carry Save Adder

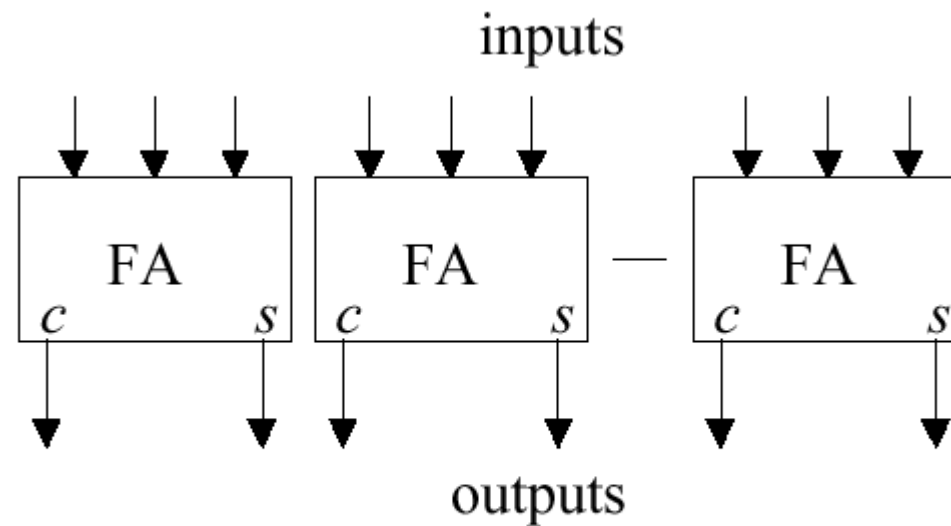


Fig. 7.23 Carry save adder.

Memory Structure

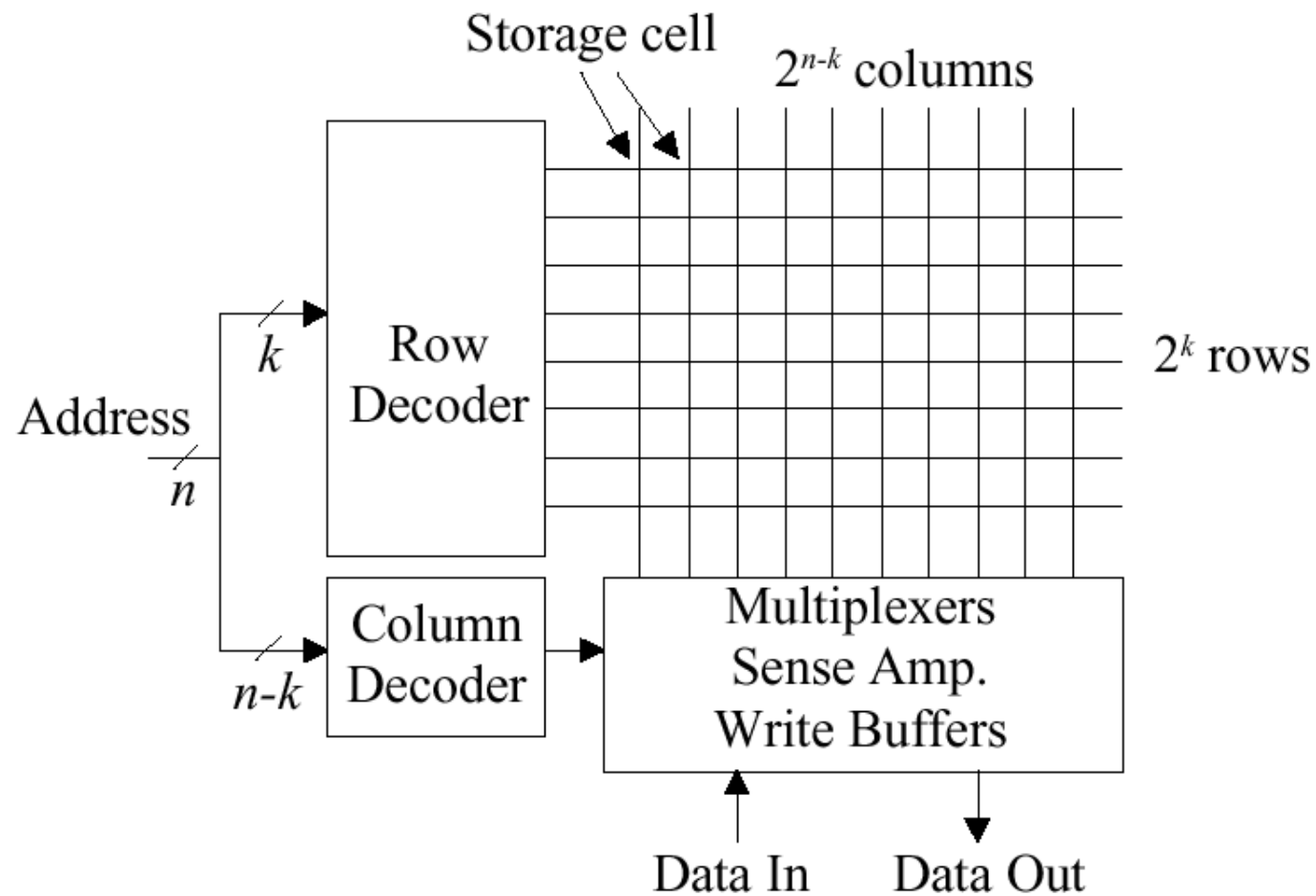


Fig. 7.24 Memory structure.

ROM

Address ($a_2a_1a_0$)	Data (d_1d_0)
000	00
001	01
010	01
011	10
100	01
101	10
110	10
111	11

Fig. 7.25 Data values to be stored in a ROM.

ROM

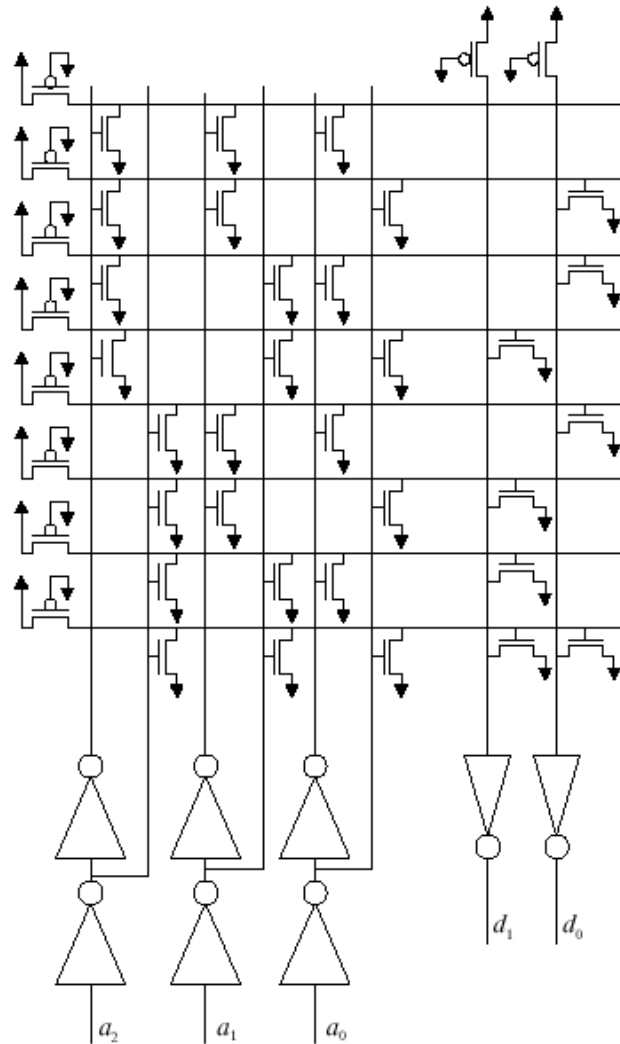


Fig. 7.26 Implementation of a ROM.

RAM Cell

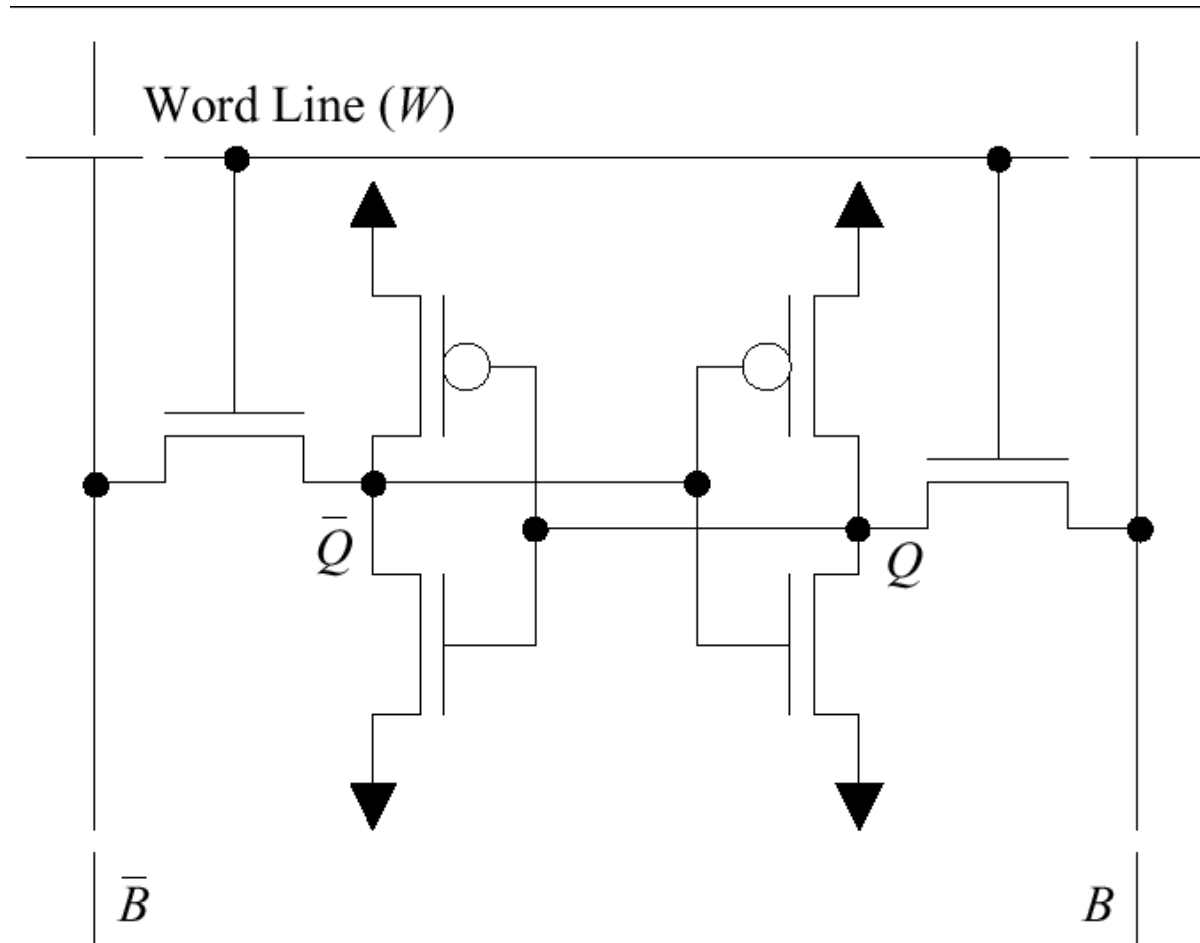


Fig. 7.27 RAM cell.

Sense Amplifier

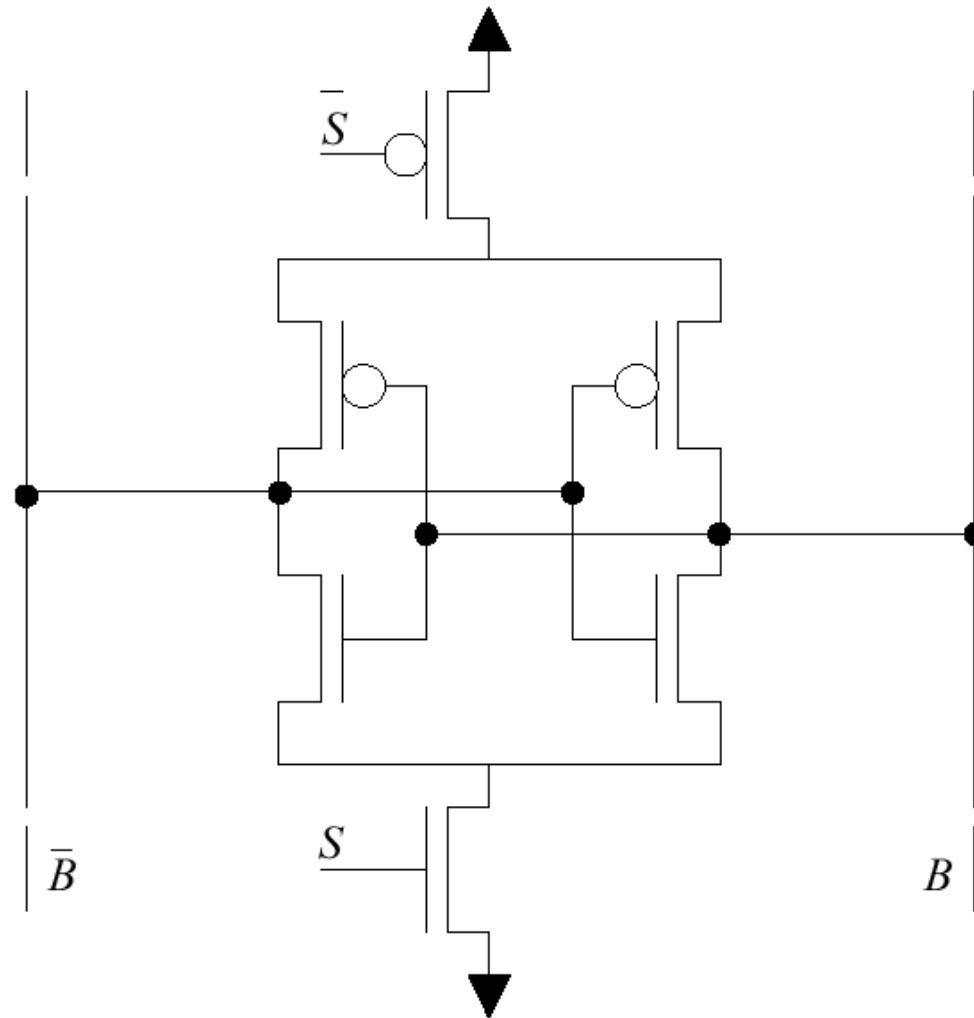


Fig. 7.28 Sense amplifier.

DRAM Cell

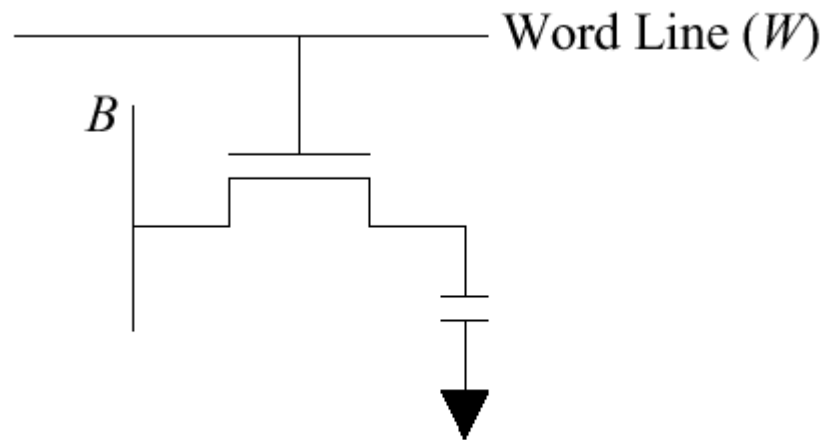


Fig. 7.29 One-transistor DRAM cell