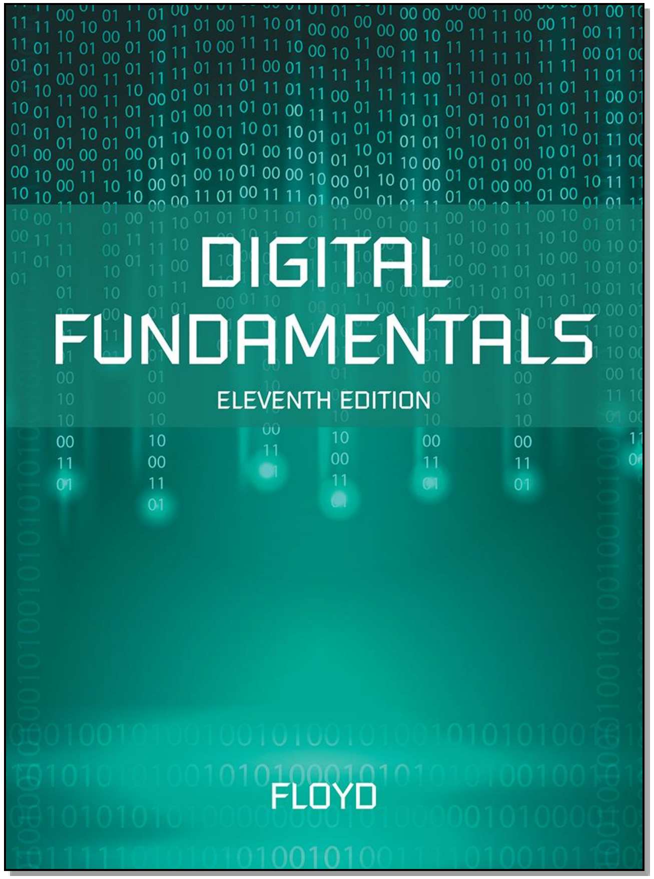
Digital Fundamentals

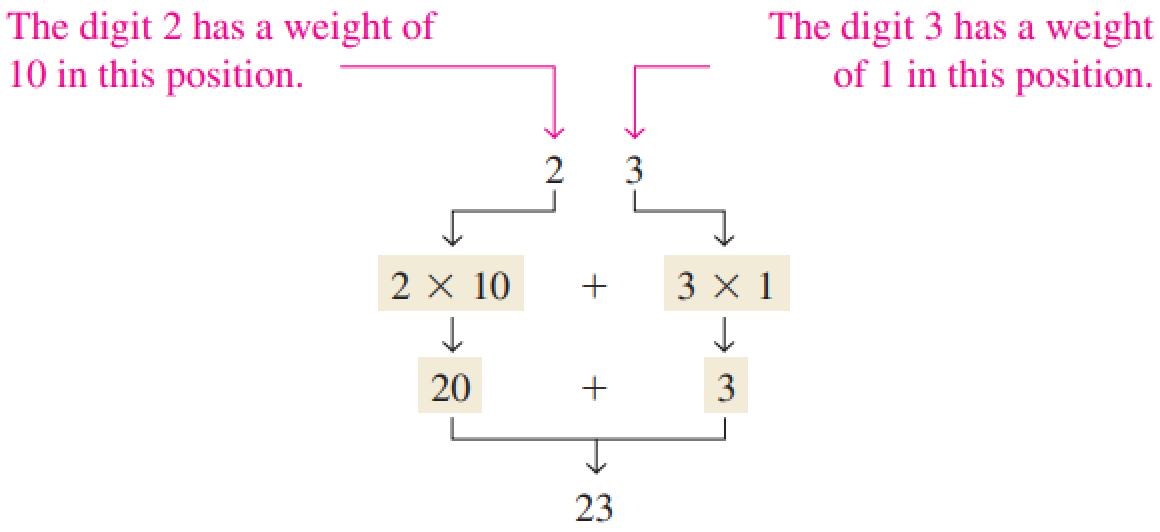
ELEVENTH EDITION



CHAPTER 2

Number Systems, Operations, and Codes

Decimal numbers



Example

Express the decimal number 47 as a sum of the values of each digit.

Solution

The digit 4 has a weight of 10, which is 10^1, as indicated by its position. The digit 7 has a weight of 1, which is 10^0, as indicated by its position.

$$
\begin{aligned}
47 &= (4 \times 10^1) + (7 \times 10^0) \\
&= (4 \times 10) + (7 \times 1) = \mathbf{40 + 7}
\end{aligned}
$$

Example

Express the decimal number 568.23 as a sum of the values of each digit.

Solution

The whole number digit 5 has a weight of 100, which is 10^2, the digit 6 has a weight of 10, which is 10^1, the digit 8 has a weight of 1, which is 10^0, the fractional digit 2 has a weight of 0.1, which is 10^{-1}, and the fractional digit 3 has a weight of 0.01, which is 10^{-2}.

$$\begin{aligned}
568.23 &= (5 \times 10^2) + (6 \times 10^1) + (8 \times 10^0) + (2 \times 10^{-1}) + (3 \times 10^{-2}) \\
&= (5 \times 100) + (6 \times 10) + (8 \times 1) + (2 \times 0.1) + (3 \times 0.01) \\
&= \mathbf{500} + \mathbf{60} + \mathbf{8} + \mathbf{0.2} + \mathbf{0.03}
\end{aligned}$$

Binary Numbers

TABLE 2–1

Decimal Number	Binary Number			
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Illustration of a simple binary counting application

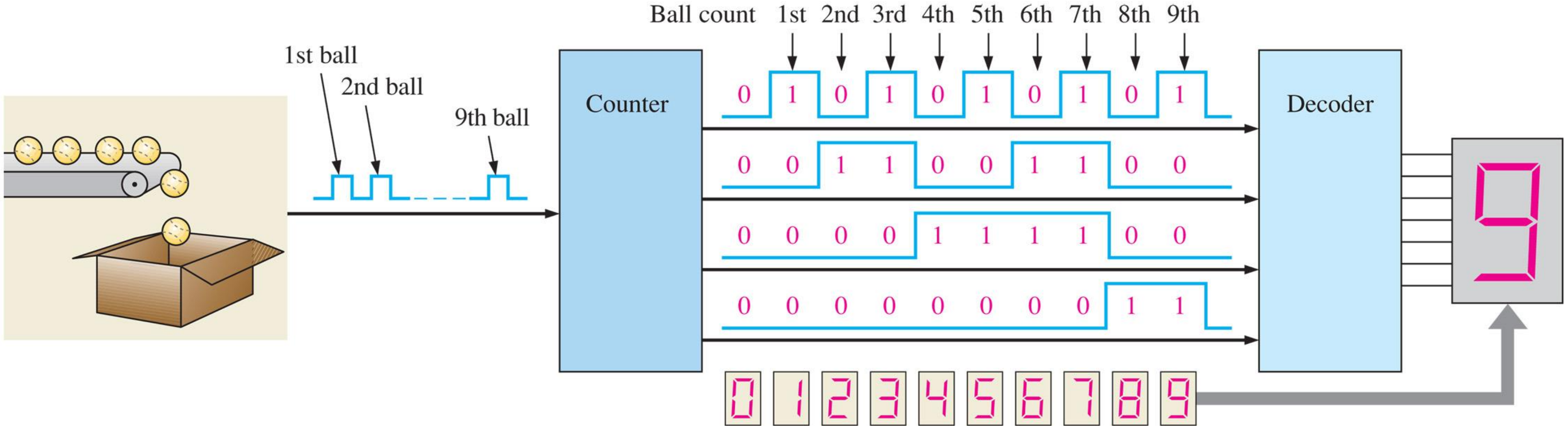


TABLE 2–2

Binary weights.

Positive Powers of Two (Whole Numbers)									Negative Powers of Two (Fractional Number)					
2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}
256	128	64	32	16	8	4	2	1	1/2 0.5	1/4 0.25	1/8 0.125	1/16 0.625	1/32 0.03125	1/64 0.015625

Example

Convert the fractional binary number 0.1011 to decimal.

Solution

Determine the weight of each bit that is a 1, and then sum the weights to get the decimal fraction.

Weight:	2^{-1}	2^{-2}	2^{-3}	2^{-4}
Binary number:	0 . 1	0	1	1

$$\begin{aligned} 0.1011 &= 2^{-1} + 2^{-3} + 2^{-4} \\ &= 0.5 + 0.125 + 0.0625 = \mathbf{0.6875} \end{aligned}$$

Example

Convert the binary whole number 1101101 to decimal.

Solution

Determine the weight of each bit that is a 1, and then find the sum of the weights to get the decimal number.

$$\begin{array}{rl} \text{Weight:} & 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0 \\ \text{Binary number:} & 1\ \ 1\ \ 0\ \ 1\ \ 1\ \ 0\ \ 1 \end{array}$$

$$\begin{aligned} 1101101 &= 2^6 + 2^5 + 2^3 + 2^2 + 2^0 \\ &= 64 + 32 + 8 + 4 + 1 = \mathbf{109} \end{aligned}$$

PEARSON

Decimal-to-Binary Conversion

Sum-of-Weights Method

Convert the following decimal numbers to binary:

(a) 12 **(b)** 25

(c) 58 **(d)** 82

Solution

(a) $12 = 8 + 4 = 2^3 + 2^2 \longrightarrow$ **1100**

(b) $25 = 16 + 8 + 1 = 2^4 + 2^3 + 2^0 \longrightarrow$ **11001**

(c) $58 = 32 + 16 + 8 + 2 = 2^5 + 2^4 + 2^3 + 2^1 \longrightarrow$ **111010**

(d) $82 = 64 + 16 + 2 = 2^6 + 2^4 + 2^1 \longrightarrow$ **1010010**

Repeated Division-by-2 Method

Convert the following decimal numbers to binary:

(a) 19 **(b)** 45

Solution

(a)

	Remainder
$\frac{19}{2} = 9$	1
$\frac{9}{2} = 4$	1
$\frac{4}{2} = 2$	0
$\frac{2}{2} = 1$	0
$\frac{1}{2} = 0$	1

1 0 0 1 1

MSB LSB

(b)

	Remainder
$\frac{45}{2} = 22$	1
$\frac{22}{2} = 11$	0
$\frac{11}{2} = 5$	1
$\frac{5}{2} = 2$	1
$\frac{2}{2} = 1$	0
$\frac{1}{2} = 0$	1

1 0 1 1 0 1

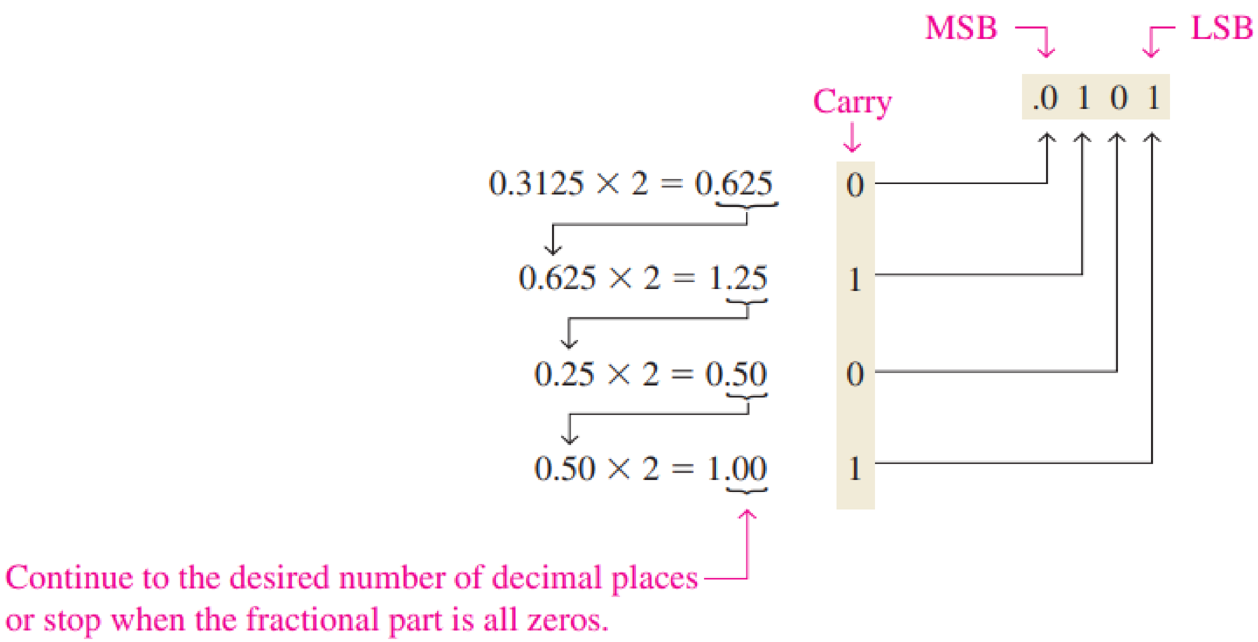
MSB LSB

Converting Decimal Fractions to Binary

Sum-of-Weights

$$0.625 = 0.5 + 0.125 = 2^{-1} + 2^{-3} = 0.101$$

Repeated Multiplication by 2



Binary Arithmetic

Binary Addition

$0 + 0 = 0$	Sum of 0 with a carry of 0
$0 + 1 = 1$	Sum of 1 with a carry of 0
$1 + 0 = 1$	Sum of 1 with a carry of 0
$1 + 1 = 10$	Sum of 0 with a carry of 1

Examples

Add the following binary numbers:

(a) 11 + 11 **(b)** 100 + 10
(c) 111 + 11 **(d)** 110 + 100

Solution

The equivalent decimal addition is also shown for reference.

(a)
$$\begin{array}{r} 11 \\ +\,11 \\ \hline \mathbf{110} \end{array} \qquad \begin{array}{r} 3 \\ +\,3 \\ \hline 6 \end{array}$$

(b)
$$\begin{array}{r} 100 \\ +\,10 \\ \hline \mathbf{110} \end{array} \qquad \begin{array}{r} 4 \\ +\,2 \\ \hline 6 \end{array}$$

(c)
$$\begin{array}{r} 111 \\ +\,11 \\ \hline \mathbf{1010} \end{array} \qquad \begin{array}{r} 7 \\ +\,3 \\ \hline 10 \end{array}$$

(d)
$$\begin{array}{r} 110 \\ +\,100 \\ \hline \mathbf{1010} \end{array} \qquad \begin{array}{r} 6 \\ +\,4 \\ \hline 10 \end{array}$$

Binary Subtraction

$$0 - 0 = 0$$
$$1 - 1 = 0$$
$$1 - 0 = 1$$
$$10 - 1 = 1 \qquad 0 - 1 \text{ with a borrow of } 1$$

Examples

Perform the following binary subtractions:

(a) 11 − 01 **(b)** 11 − 10

Solution

(a)
$$\begin{array}{r} 11 \\ -01 \\ \hline \mathbf{10} \end{array} \qquad \begin{array}{r} 3 \\ -1 \\ \hline 2 \end{array}$$

(b)
$$\begin{array}{r} 11 \\ -10 \\ \hline \mathbf{01} \end{array} \qquad \begin{array}{r} 3 \\ -2 \\ \hline 1 \end{array}$$

No borrows were required in this example. The binary number 01 is the same as 1.

Subtract 011 from 101.

Solution

$$\begin{array}{r} 101 \\ -011 \\ \hline \mathbf{010} \end{array} \qquad \begin{array}{r} 5 \\ -3 \\ \hline 2 \end{array}$$

Binary Multiplication

$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

PEARSON

Examples

Perform the following binary multiplications:

(a) 11 × 11 **(b)** 101 × 111

Solution

(a)

$$\text{Partial products}\left\{\begin{array}{r} 11 \\ \times\ 11 \\ \hline 11 \\ +11 \\ \hline \mathbf{1001} \end{array}\right. \qquad \begin{array}{r} 3 \\ \times\ 3 \\ \hline 9 \end{array}$$

(b)

$$\text{Partial products}\left\{\begin{array}{r} 111 \\ \times\ 101 \\ \hline 111 \\ 000 \\ +111 \\ \hline \mathbf{100011} \end{array}\right. \qquad \begin{array}{r} 7 \\ \times\ 5 \\ \hline 35 \end{array}$$

Binary Division

Perform the following binary divisions:

(a) $110 \div 11$ **(b)** $110 \div 10$

Solution

(a)

$$
\begin{array}{r}
\mathbf{10} \\
11\overline{)110} \\
\underline{11} \\
000
\end{array}
\qquad
\begin{array}{r}
2 \\
3\overline{)6} \\
\underline{6} \\
0
\end{array}
$$

(b)

$$
\begin{array}{r}
\mathbf{11} \\
10\overline{)110} \\
\underline{10} \\
10 \\
\underline{10} \\
00
\end{array}
\qquad
\begin{array}{r}
3 \\
2\overline{)6} \\
\underline{6} \\
0
\end{array}
$$

Example of inverters used to obtain the 1's complement of a binary number

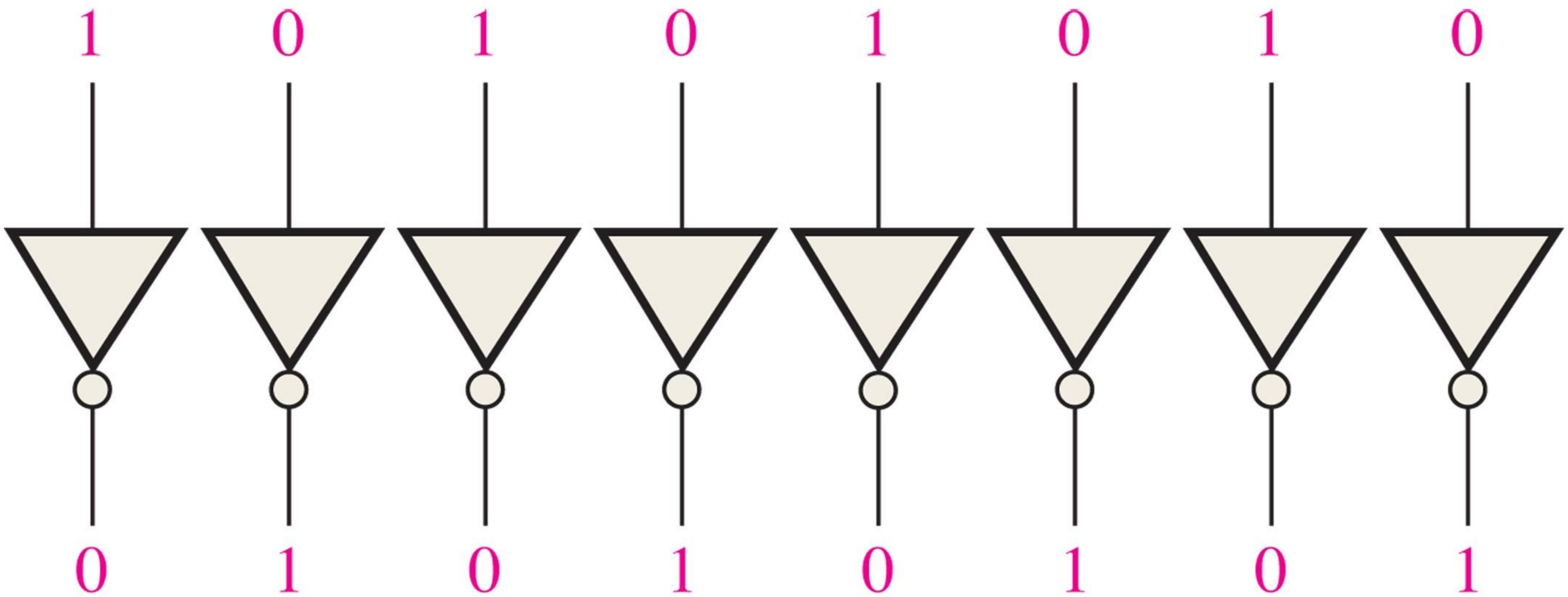
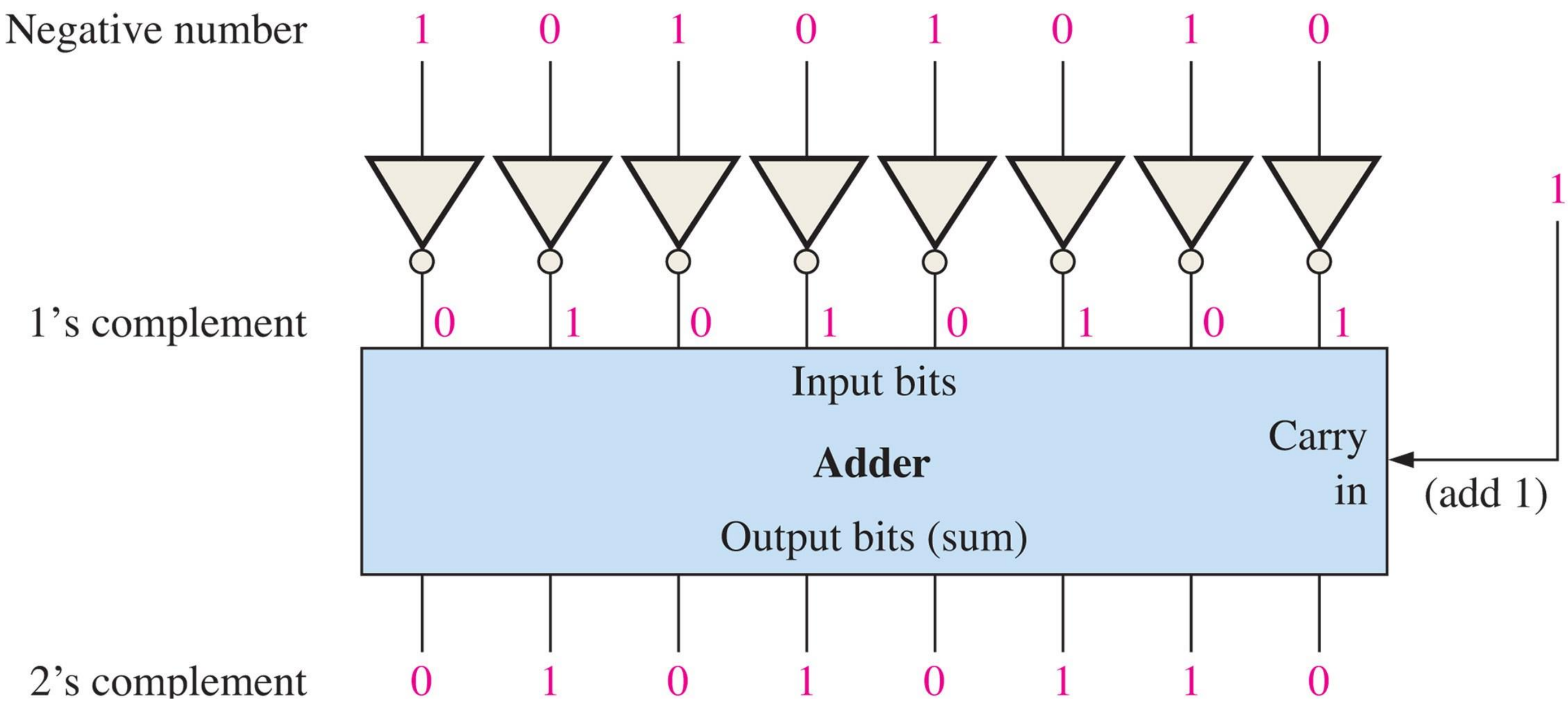


FIGURE 2-3 Example of obtaining the 2's complement of a negative binary number.



example

Find the 2's complement of 10110010.

Solution

10110010	Binary number
01001101	1's complement
+ 1	Add 1
01001110	2's complement

Signed Numbers

Sign-Magnitude Form

$$\underset{\text{Sign bit}}{0}\underbrace{0011001}_{\text{Magnitude bits}}$$

The decimal number -25 is expressed as

$$10011001$$

1’s Complement Form

The 1’s complement of +25
(00011001) as 11100110

In the 1’s complement form, a negative number is the
1’s complement of the corresponding positive number

2’s Complement Form

-25 = 11100111

In the 2’s complement form, a negative number is the 2’s complement of the corresponding positive number.

Example

Express the decimal number −39 as an 8-bit number in the sign-magnitude, 1's complement, and 2's complement forms.

Solution

First, write the 8-bit number for +39.

00100111

In the *sign-magnitude form,* −39 is produced by changing the sign bit to a 1 and leaving the magnitude bits as they are. The number is

10100111

In the *1's complement form,* −39 is produced by taking the 1's complement of +39 (00100111).

11011000

In the *2's complement form,* −39 is produced by taking the 2's complement of +39 (00100111) as follows:

11011000	1's complement
+ 1	
11011001	2's complement

Example

Determine the decimal value of this signed binary number expressed in sign-magnitude: 10010101.

Solution

The seven magnitude bits and their powers-of-two weights are as follows:

2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	0	1	0	1	0	1

Summing the weights where there are 1s,

$$16 + 4 + 1 = 21$$

The sign bit is 1; therefore, the decimal number is **−21**.

Examples

Determine the decimal values of the signed binary numbers expressed in 1's complement:

(a) 00010111 **(b)** 11101000

Solution

(a) The bits and their powers-of-two weights for the positive number are as follows:

-2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	0	0	1	0	1	1	1

Summing the weights where there are 1s,

$$16 + 4 + 2 + 1 = \mathbf{+23}$$

(b) The bits and their powers-of-two weights for the negative number are as follows. Notice that the negative sign bit has a weight of -2^7 or -128.

-2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	1	1	0	1	0	0	0

Summing the weights where there are 1s,

$$-128 + 64 + 32 + 8 = -24$$

Adding 1 to the result, the final decimal number is

$$-24 + 1 = \mathbf{-23}$$

Examples

Determine the decimal values of the signed binary numbers expressed in 2's complement:

(a) 01010110 **(b)** 10101010

Solution

(a) The bits and their powers-of-two weights for the positive number are as follows:

-2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	1	0	1	0	1	1	0

Summing the weights where there are 1s,

$$64 + 16 + 4 + 2 = \mathbf{+86}$$

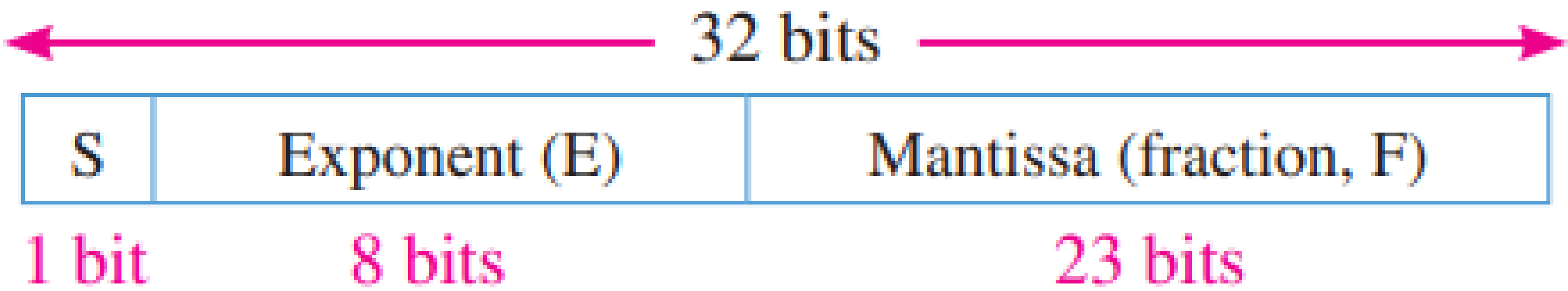
(b) The bits and their powers-of-two weights for the negative number are as follows. Notice that the negative sign bit has a weight of $-2^7 = -128$.

-2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	1	0	1	0	1	0

Summing the weights where there are 1s,

$$-128 + 32 + 8 + 2 = \mathbf{-86}$$

Single-Precision Floating-Point Binary Numbers



The eight bits in the exponent represent a biased exponent, which is obtained by adding 127 to the actual exponent. The purpose of the bias is to allow very large or very small numbers without requiring a separate sign bit for the exponents. The biased exponent allows a range of actual exponent values from -126 to +128. To illustrate how a binary number is expressed in floating-point format, let's use 1011010010001 as an example

First, it can be expressed as 1 plus a fractional binary number by moving the binary point 12 places to the left and then multiplying by the appropriate power of two.

$$101101001 0001 = 1.011010010001 \times 2^{12}$$

Assuming that this is a positive number, the sign bit (S) is 0. The exponent, 12, is expressed as a biased exponent by adding it to 127 (12 + 127 = 139). The biased exponent (E) is expressed as the binary number 10001011.

The mantissa is the fractional part (F) of the binary number, .011010010001.

S	E	F
0	10001011	01101001000100000000000

e a binary number that is already in floating-point format.

$$\text{Number} = (-1)^{S}(1 + F)(2^{E-127})$$

Example

S	E	F
1	10010001	10001110001000000000000

The sign bit is 1. The biased exponent is 10010001 = 145.

Applying the formula, we get Number =

(-1)1 (1.10001110001)(2 to the power $_{145\text{-}127}$) =

(-1)(1.10001110001)(2^18) = -1100011100010000000 This

floating-point binary number is equivalent to -407,688 in decimal.

Example

Convert the decimal number 3.248×10^4 to a single-precision floating-point binary number.

Solution

Convert the decimal number to binary.

$$3.248 \times 10^4 = 32480 = 111111011100000_2 = 1.11111011100000 \times 2^{14}$$

The MSB will not occupy a bit position because it is always a 1. Therefore, the mantissa is the fractional 23-bit binary number 11111011100000000000000 and the biased exponent is

$$14 + 127 = 141 = 10001101_2$$

The complete floating-point number is

0	**10001101**	**11111011100000000000000**

Arithmetic Operations with Signed Numbers

2's complement form for representing signed numbers is the most widely used in computers and microprocessor-based systems

Addition and Subtraction

Both numbers positive:

$$\begin{array}{r} 00000111 \\ +\ 00000100 \\ \hline 00001011 \end{array} \qquad \begin{array}{r} 7 \\ +\ 4 \\ \hline 11 \end{array}$$

Positive number with magnitude larger than negative number:

$$\begin{array}{rr} & 00001111 \\ & +\ 11111010 \\ \hline \text{Discard carry} \longrightarrow \boxed{1} & 00001001 \end{array} \qquad \begin{array}{r} 15 \\ +\ -6 \\ \hline 9 \end{array}$$

PEARSON

Negative number with magnitude larger than positive number:

00010000	16
+ 11101000	+ −24
11111000	−8

The sum is negative and therefore in 2's complement form.

Both numbers negative:

	11111011	−5
	+ 11110111	+ −9
Discard carry → 1	11110010	−14

The final carry bit is discarded. The sum is negative and therefore in 2's complement form.

Overflow Condition

When two numbers are added and the number of bits required to represent the sum exceeds the number of bits in the two numbers, an overflow results as indicated by an incorrect sign bit. An overflow can occur only when both numbers are positive or both numbers are negative. If the sign bit of the result is different than the sign bit of the numbers that are added, overflow is indicated.

$$\begin{array}{r} 01111101 \\ +\ 00111010 \\ \hline 1\underbrace{0110111} \end{array} \qquad \begin{array}{r} 125 \\ +\ \ 58 \\ \hline 183 \end{array}$$

Sign incorrect

Magnitude incorrect

Add the signed numbers: 01000100, 00011011, 00001110, and 00010010.

Solution

The equivalent decimal additions are given for reference.

68	01000100	
+ 27	+ 00011011	Add 1st two numbers
95	01011111	1st sum
+ 14	+ 00001110	Add 3rd number
109	01101101	2nd sum
+ 18	+ 00010010	Add 4th number
127	**01111111**	Final sum

Perform each of the following subtractions of the signed numbers:

(a) 00001000 − 00000011

(b) 00001100 − 11110111

(c) 11100111 − 00010011

(d) 10001000 − 11100010

Like in other examples, the equivalent decimal subtractions are given for reference.

(a) In this case, $8 - 3 = 8 + (-3) = 5$.

	00001000	Minuend (+8)
	+ 11111101	2's complement of subtrahend (−3)
Discard carry →	**1 00000101**	Difference (+5)

(b) In this case, $12 - (-9) = 12 + 9 = 21$.

00001100	Minuend (+12)
+ 00001001	2's complement of subtrahend (+9)
00010101	Difference (+21)

(c) In this case, $-25 - (+19) = -25 + (-19) = -44$.

	11100111	Minuend (−25)
	+ 11101101	2's complement of subtrahend (−19)
Discard carry	**1 11010100**	Difference (−44)

(d) In this case, $-120 - (-30) = -120 + 30 = -90$.

10001000	Minuend (−120)
+ 00011110	2's complement of subtrahend (+30)
10100110	Difference (−90)

Multiplication

Multiply the signed binary numbers: 01001101 (multiplicand) and 00000100 (multiplier) using the direct addition method.

Solution

Since both numbers are positive, they are in true form, and the product will be positive. The decimal value of the multiplier is 4, so the multiplicand is added to itself four times as follows:

01001101	1st time
+ 01001101	2nd time
10011010	Partial sum
+ 01001101	3rd time
11100111	Partial sum
+ 01001101	4th time
100110100	Product

239	Multiplicand
$\times$ 123	Multiplier
717	1st partial product (3×239)
478	2nd partial product (2×239)
+ 239	3rd partial product (1×239)
29,397	Final product

Multiply the signed binary numbers: 01010011 (multiplicand) and 11000101 (multiplier).

1010011	Multiplicand
× 0111011	Multiplier
1010011	1st partial product
+ 1010011	2nd partial product
11111001	Sum of 1st and 2nd
+ 0000000	3rd partial product
011111001	Sum
+ 1010011	4th partial product
1110010001	Sum
+ 1010011	5th partial product
100011000001	Sum
+ 1010011	6th partial product
1001100100001	Sum
+ 0000000	7th partial product
1001100100001	Final product

Since the sign of the product is a 1 as determined in step 1, take the 2's complement of the product.

1001100100001 $\longrightarrow$ 0110011011111

Attach the sign bit

1 0110011011111

Division

$$\frac{\text{dividend}}{\text{divisor}} = \text{quotient}$$

- **If the signs are the same, the quotient is positive.**
- **If the signs are different, the quotient is negative.**

21	Dividend
− 7	1st subtraction of divisor
14	1st partial remainder
− 7	2nd subtraction of divisor
7	2nd partial remainder
− 7	3rd subtraction of divisor
0	Zero remainder

Divide 01100100 by 00011001.

Step 1: The signs of both numbers are positive, so the quotient will be positive. The quotient is initially zero: 00000000.

Step 2: Subtract the divisor from the dividend using 2's complement addition (remember that final carries are discarded).

01100100	Dividend
+ 11100111	2's complement of divisor
01001011	Positive 1st partial remainder

Add 1 to quotient: 00000000 + 00000001 = 00000001.

Step 3: Subtract the divisor from the 1st partial remainder using 2's complement addition.

01001011	1st partial remainder
+ 11100111	2's complement of divisor
00110010	Positive 2nd partial remainder

Add 1 to quotient: 00000001 + 00000001 = 00000010.

Step 4: Subtract the divisor from the 2nd partial remainder using 2's complement addition.

00110010	2nd partial remainder
+ 11100111	2's complement of divisor
00011001	Positive 3rd partial remainder

Add 1 to quotient: 00000010 + 00000001 = 00000011.

Step 5: Subtract the divisor from the 3rd partial remainder using 2's complement addition.

00011001	3rd partial remainder
+ 11100111	2's complement of divisor
00000000	Zero remainder

Add 1 to quotient: 00000011 + 00000001 = **00000100** (final quotient). The process is complete.

Hexadecimal Numbers

TABLE 2–3

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Convert the following binary numbers to hexadecimal:

(a) 1100101001010111 **(b)** 111111000101101001

Solution

(a) $$\underbrace{1100}_{\displaystyle C}\,\underbrace{1010}_{\displaystyle A}\,\underbrace{0101}_{\displaystyle 5}\,\underbrace{0111}_{\displaystyle 7} = \mathbf{CA57}_{16}$$

(b) $$\underbrace{0011}_{\displaystyle 3}\,\underbrace{1111}_{\displaystyle F}\,\underbrace{0001}_{\displaystyle 1}\,\underbrace{0110}_{\displaystyle 6}\,\underbrace{1001}_{\displaystyle 9} = \mathbf{3F169}_{16}$$

Two zeros have been added in part (b) to complete a 4-bit group at the left.

Hexadecimal-to-Binary Conversion

Determine the binary numbers for the following hexadecimal numbers:

(a) $10A4_{16}$ **(b)** $CF8E_{16}$ **(c)** 9742_{16}

Solution

(a) $\overset{1}{\mathbf{1}}\overbrace{\mathbf{0000}}^{0}\overbrace{\mathbf{1010}}^{A}\overbrace{\mathbf{0100}}^{4}$

(b) $\overbrace{\mathbf{1100}}^{C}\overbrace{\mathbf{1111}}^{F}\overbrace{\mathbf{1000}}^{8}\overbrace{\mathbf{1110}}^{E}$

(c) $\overbrace{\mathbf{1001}}^{9}\overbrace{\mathbf{0111}}^{7}\overbrace{\mathbf{0100}}^{4}\overbrace{\mathbf{0010}}^{2}$

In part (a), the MSB is understood to have three zeros preceding it, thus forming a 4-bit group.

Hexadecimal-to-Decimal Conversion

Convert the following hexadecimal numbers to decimal:

(a) $1C_{16}$ **(b)** $A85_{16}$

Solution

Remember, convert the hexadecimal number to binary first, then to decimal.

(a)

$$\overbrace{0001}^{1}\overbrace{1100}^{C} = 2^4 + 2^3 + 2^2 = 16 + 8 + 4 = \mathbf{28}_{10}$$

(b)

$$\overbrace{1010}^{A}\overbrace{1000}^{8}\overbrace{0101}^{5} = 2^{11} + 2^9 + 2^7 + 2^2 + 2^0 = 2048 + 512 + 128 + 4 + 1 = \mathbf{2693}_{10}$$

Convert the following hexadecimal numbers to decimal:

(a) $E5_{16}$ **(b)** $B2F8_{16}$

Solution

Recall from Table 2–3 that letters A through F represent decimal numbers 10 through 15, respectively.

(a) $E5_{16} = (E \times 16) + (5 \times 1) = (14 \times 16) + (5 \times 1) = 224 + 5 = \mathbf{229_{10}}$

(b)
$$\begin{aligned} B2F8_{16} &= (B \times 4096) + (2 \times 256) + (F \times 16) + (8 \times 1) \\ &= (11 \times 4096) + (2 \times 256) + (15 \times 16) + (8 \times 1) \\ &= 45{,}056 + 512 + 240 + 8 = \mathbf{45{,}816_{10}} \end{aligned}$$

Decimal-to-Hexadecimal Conversion

Convert the decimal number 650 to hexadecimal by repeated division by 16.

Solution

Hexadecimal remainder

$$\frac{650}{16} = 40.625 \rightarrow 0.625 \times 16 = 10 = \text{A}$$

$$\frac{40}{16} = 2.5 \rightarrow 0.5 \times 16 = 8 = 8$$

$$\frac{2}{16} = 0.125 \rightarrow 0.125 \times 16 = 2 = 2$$

Stop when whole number quotient is zero.

2 8 A Hexadecimal number

MSD LSD

Hexadecimal Addition

Add the following hexadecimal numbers:

(a) $23_{16} + 16_{16}$ **(b)** $58_{16} + 22_{16}$ **(c)** $2B_{16} + 84_{16}$ **(d)** $DF_{16} + AC_{16}$

(a) $\begin{array}{r} 23_{16} \\ +\ 16_{16} \\ \hline \mathbf{39_{16}} \end{array}$

right column: $3_{16} + 6_{16} = 3_{10} + 6_{10} = 9_{10} = 9_{16}$
left column: $2_{16} + 1_{16} = 2_{10} + 1_{10} = 3_{10} = 3_{16}$

(b) $\begin{array}{r} 58_{16} \\ +\ 22_{16} \\ \hline \mathbf{7A_{16}} \end{array}$

right column: $8_{16} + 2_{16} = 8_{10} + 2_{10} = 10_{10} = A_{16}$
left column: $5_{16} + 2_{16} = 5_{10} + 2_{10} = 7_{10} = 7_{16}$

(c) $\begin{array}{r} 2B_{16} \\ +\ 84_{16} \\ \hline \mathbf{AF_{16}} \end{array}$

right column: $B_{16} + 4_{16} = 11_{10} + 4_{10} = 15_{10} = F_{16}$
left column: $2_{16} + 8_{16} = 2_{10} + 8_{10} = 10_{10} = A_{16}$

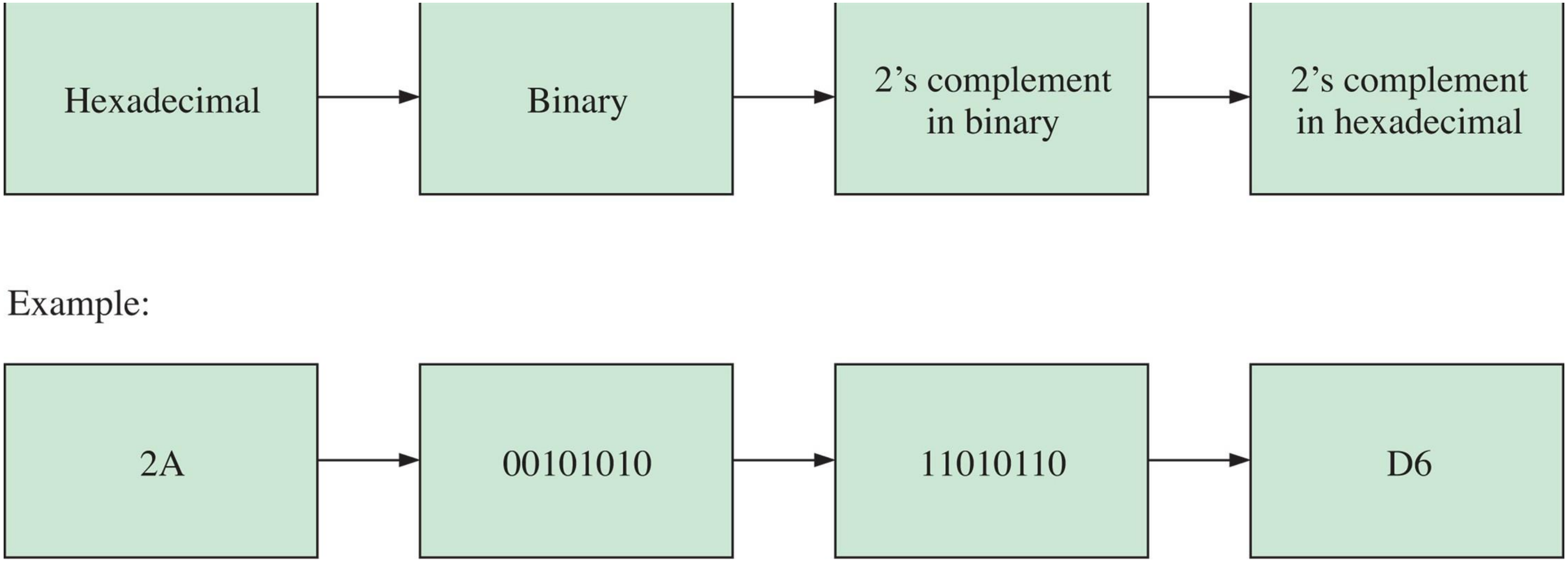
(d) $\begin{array}{r} DF_{16} \\ +\ AC_{16} \\ \hline \mathbf{18B_{16}} \end{array}$

right column: $F_{16} + C_{16} = 15_{10} + 12_{10} = 27_{10}$
$27_{10} - 16_{10} = 11_{10} = B_{16}$ with a 1 carry

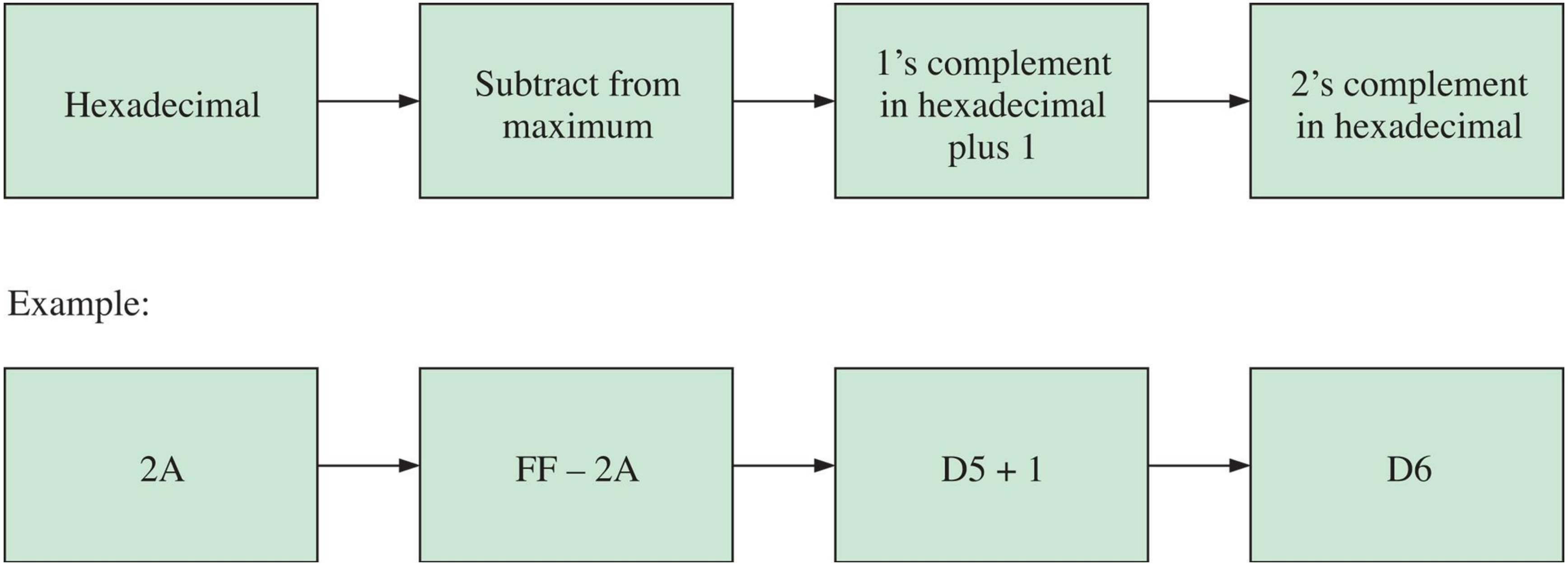
left column: $D_{16} + A_{16} + 1_{16} = 13_{10} + 10_{10} + 1_{10} = 24_{10}$
$24_{10} - 16_{10} = 8_{10} = 8_{16}$ with a 1 carry

Hexadecimal Subtraction

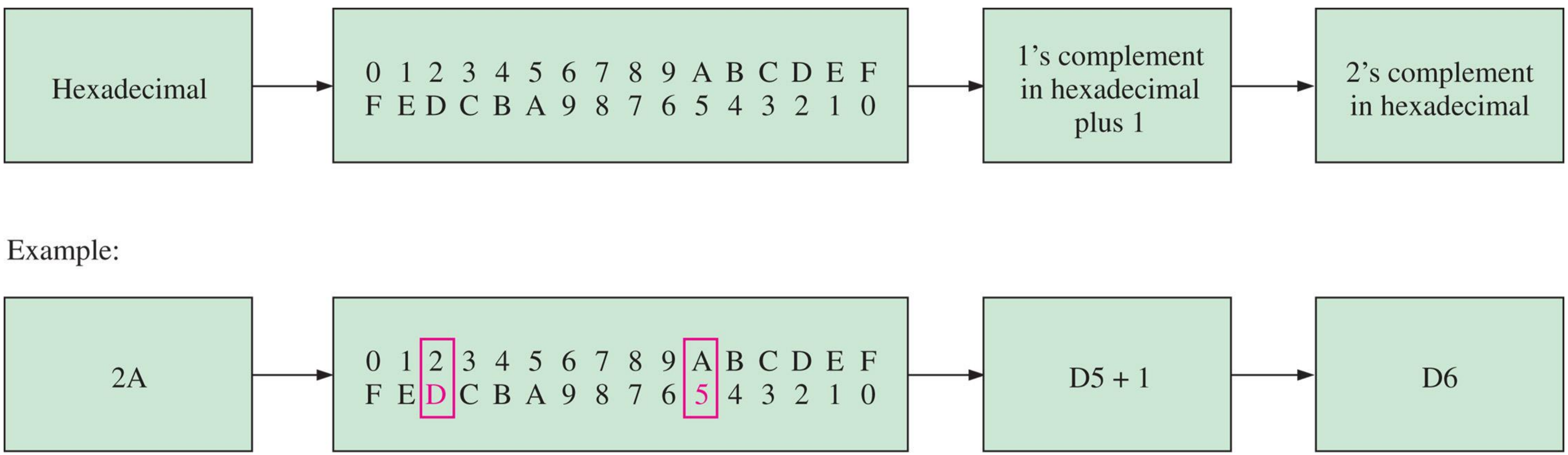
Getting the 2’s complement of a hexadecimal number, Method 1.



Getting the 2's complement of a hexadecimal number, Method 2.



Getting the 2’s complement of a hexadecimal number, Method 3.



Subtract the following hexadecimal numbers:

(a) $84_{16} - 2A_{16}$ **(b)** $C3_{16} - 0B_{16}$

Solution

(a) $2A_{16} = 00101010$

2's complement of $2A_{16} = 11010110 = D6_{16}$ (using Method 1)

$$\begin{array}{r l} 84_{16} & \\ +\ D6_{16} & \text{Add} \\ \hline \cancel{1}5A_{16} & \text{Drop carry, as in 2's complement addition} \end{array}$$

The difference is $\mathbf{5A_{16}}$.

(b) $0B_{16} = 00001011$

2's complement of $0B_{16} = 11110101 = F5_{16}$ (using Method 1)

$$\begin{array}{r l} C3_{16} & \\ +\ F5_{16} & \text{Add} \\ \hline \cancel{1}B8_{16} & \text{Drop carry} \end{array}$$

The difference is $\mathbf{B8}_{16}$.

PEARSON

Octal Numbers

TABLE 2–4

Octal/binary conversion.

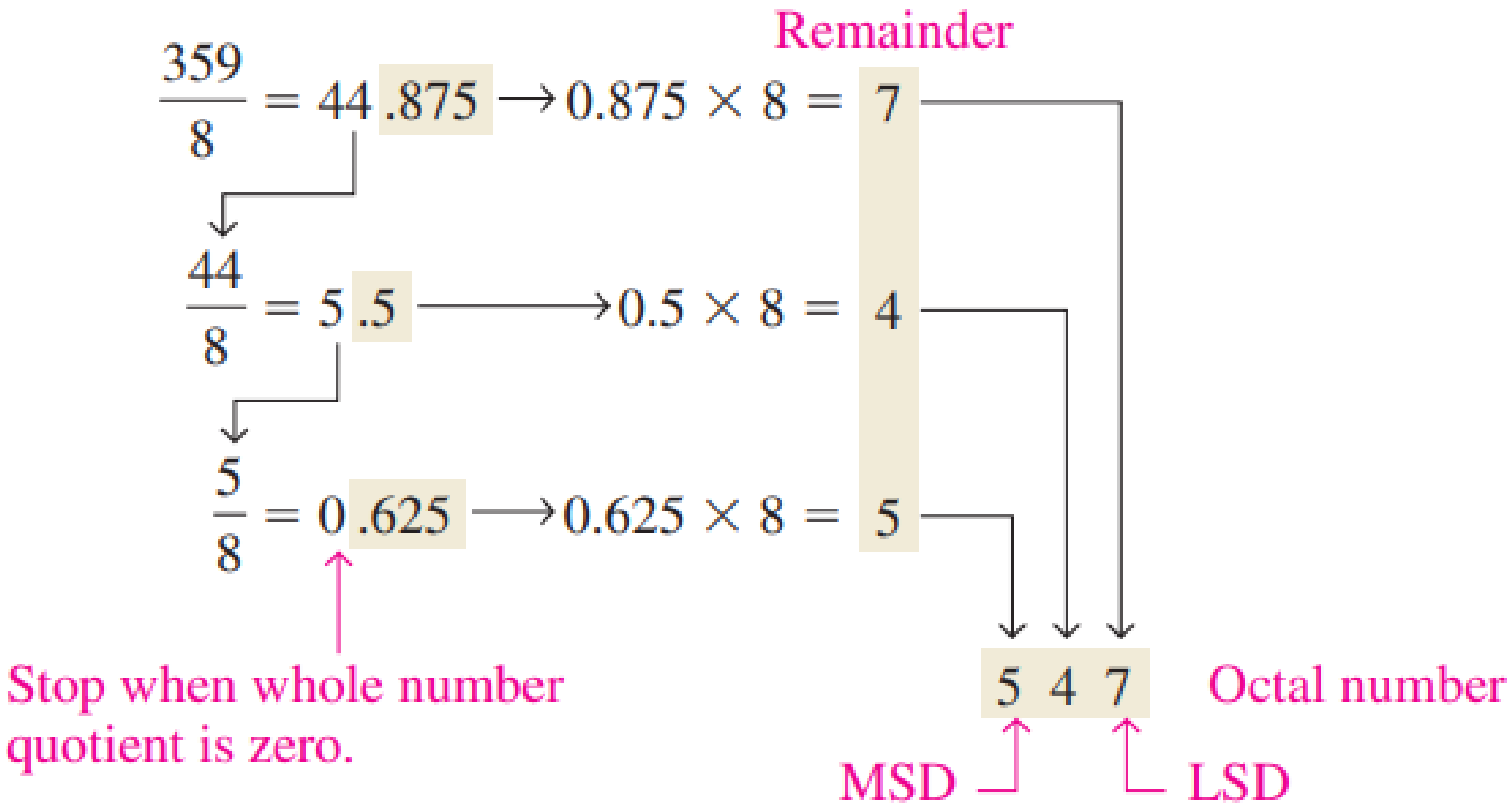
Octal Digit	0	1	2	3	4	5	6	7
Binary	000	001	010	011	100	101	110	111

Octal-to-Decimal Conversion

$$
\begin{array}{rl}
\text{Weight:} & 8^3\ 8^2\ 8^1\ 8^0 \\
\text{Octal number:} & 2\ \ 3\ \ 7\ \ 4
\end{array}
$$

$$
\begin{aligned}
2374_8 &= (2 \times 8^3) + (3 \times 8^2) + (7 \times 8^1) + (4 \times 8^0) \\
&= (2 \times 512) + (3 \times 64) + (7 \times 8) + (4 \times 1) \\
&= 1024 + 192 + 56 + 4 = 1276_{10}
\end{aligned}
$$

Decimal-to-Octal Conversion



Octal-to-Binary Conversion

Convert each of the following octal numbers to binary:

(a) 13_8 **(b)** 25_8 **(c)** 140_8 **(d)** 7526_8

Solution

(a) $\overset{1}{\overset{\downarrow}{\overbrace{\mathbf{001}}}}\overset{3}{\overset{\downarrow}{\overbrace{\mathbf{011}}}}$

(b) $\overset{2}{\overset{\downarrow}{\overbrace{\mathbf{010}}}}\overset{5}{\overset{\downarrow}{\overbrace{\mathbf{101}}}}$

(c) $\overset{1}{\overset{\downarrow}{\overbrace{\mathbf{001}}}}\overset{4}{\overset{\downarrow}{\overbrace{\mathbf{100}}}}\overset{0}{\overset{\downarrow}{\overbrace{\mathbf{000}}}}$

(d) $\overset{7}{\overset{\downarrow}{\overbrace{\mathbf{111}}}}\overset{5}{\overset{\downarrow}{\overbrace{\mathbf{101}}}}\overset{2}{\overset{\downarrow}{\overbrace{\mathbf{010}}}}\overset{6}{\overset{\downarrow}{\overbrace{\mathbf{110}}}}$

Binary-to-Octal Conversion

Convert each of the following binary numbers to octal:

(a) 110101 **(b)** 101111001 **(c)** 100110011010 **(d)** 11010000100

Solution

(a) $\underbrace{110}_{6}\underbrace{101}_{5} = \mathbf{65_8}$

(b) $\underbrace{101}_{5}\underbrace{111}_{7}\underbrace{001}_{1} = \mathbf{571_8}$

(c) $\underbrace{100}_{4}\underbrace{110}_{6}\underbrace{011}_{3}\underbrace{010}_{2} = \mathbf{4632_8}$

(d) $\underbrace{011}_{3}\underbrace{010}_{2}\underbrace{000}_{0}\underbrace{100}_{4} = \mathbf{3204_8}$

Binary Coded Decimal (BCD)

TABLE 2–5

Decimal/BCD conversion.

Decimal Digit	0	1	2	3	4	5	6	7	8	9
BCD	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

Convert each of the following decimal numbers to BCD:

(a) 35 **(b)** 98 **(c)** 170 **(d)** 2469

Solution

(a) $\overbrace{0011}^{3}\overbrace{0101}^{5}$

(b) $\overbrace{1001}^{9}\overbrace{1000}^{8}$

(c) $\overbrace{0001}^{1}\overbrace{0111}^{7}\overbrace{0000}^{0}$

(d) $\overbrace{0010}^{2}\overbrace{0100}^{4}\overbrace{0110}^{6}\overbrace{1001}^{9}$

Convert each of the following BCD codes to decimal:

(a) 10000110 **(b)** 001101010001 **(c)** 1001010001110000

Solution

(a) $\underbrace{1000}_{\mathbf{8}}\underbrace{0110}_{\mathbf{6}}$

(b) $\underbrace{0011}_{\mathbf{3}}\underbrace{0101}_{\mathbf{5}}\underbrace{0001}_{\mathbf{1}}$

(c) $\underbrace{1001}_{\mathbf{9}}\underbrace{0100}_{\mathbf{4}}\underbrace{0111}_{\mathbf{7}}\underbrace{0000}_{\mathbf{0}}$

Add the following BCD numbers:

(a) 0011 + 0100 **(b)** 00100011 + 00010101

(c) 10000110 + 00010011 **(d)** 010001010000 + 010000010111

Solution

The decimal number additions are shown for comparison.

(a)
$$\begin{array}{r} 0011 \\ +\,0100 \\ \hline \mathbf{0111} \end{array} \qquad \begin{array}{r} 3 \\ +\,4 \\ \hline 7 \end{array}$$

(b)
$$\begin{array}{rr} 0010 & 0011 \\ +\,0001 & 0101 \\ \hline \mathbf{0011} & \mathbf{1000} \end{array} \qquad \begin{array}{r} 23 \\ +\,15 \\ \hline 38 \end{array}$$

(c)
$$\begin{array}{rr} 1000 & 0110 \\ +\,0001 & 0011 \\ \hline \mathbf{1001} & \mathbf{1001} \end{array} \qquad \begin{array}{r} 86 \\ +\,13 \\ \hline \mathbf{99} \end{array}$$

(d)
$$\begin{array}{rrr} 0100 & 0101 & 0000 \\ +\,0100 & 0001 & 0111 \\ \hline \mathbf{1000} & \mathbf{0110} & \mathbf{0111} \end{array} \qquad \begin{array}{r} 450 \\ +\,417 \\ \hline \mathbf{867} \end{array}$$

Note that in each case the sum in any 4-bit column does not exceed 9, and the results are valid BCD numbers.

Add the following BCD numbers:

(a) 1001 + 0100

(b) 1001 + 1001

(c) 00010110 + 00010101

(d) 01100111 + 01010011

(a)		1001		9
		+ 0100		+4
		1101	Invalid BCD number (>9)	13
		+ 0110	Add 6	
	0001	**0011**	Valid BCD number	
	$\downarrow$	$\downarrow$		
	1	3		

(b)		1001		9
		+ 1001		+ 9
	1	0010	Invalid because of carry	18
		+ 0110	Add 6	
	0001	**1000**	Valid BCD number	
	$\downarrow$	$\downarrow$		
	1	8		

(c)		0001	0110		16
		+ 0001	0101		+ 15
		0010	1011	Right group is invalid (>9), left group is valid.	31
			+ 0110	Add 6 to invalid code. Add carry, 0001, to next group.	
		0011	**0001**	Valid BCD number	
		3	1		

(d)		0110	0111		67
		+ 0101	0011		+ 53
		1011	1010	Both groups are invalid (>9)	120
		+ 0110	+ 0110	Add 6 to both groups	
	0001	**0010**	**0000**	Valid BCD number	
	1	2	0		

TABLE 2–6

Four-bit Gray code.

Decimal	Binary	Gray Code	Decimal	Binary	Gray Code
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

(a) Convert the binary number 11000110 to Gray code.

(b) Convert the Gray code 10101111 to binary.

Solution

(a) Binary to Gray code:

$$\begin{array}{ccccccccccccccc}
1 & -+\rightarrow & 1 & -+\rightarrow & 0 & -+\rightarrow & 0 & -+\rightarrow & 0 & -+\rightarrow & 1 & -+\rightarrow & 1 & -+\rightarrow & 0 \\
\downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\
\mathbf{1} & & \mathbf{0} & & \mathbf{1} & & \mathbf{0} & & \mathbf{0} & & \mathbf{1} & & \mathbf{0} & & \mathbf{1}
\end{array}$$

(b) Gray code to binary:

$$\begin{array}{ccccccccccccccc}
1 & & 0 & & 1 & & 0 & & 1 & & 1 & & 1 & & 1 \\
\downarrow & +\nearrow & \downarrow & +\nearrow & \downarrow & +\nearrow & \downarrow & +\nearrow & \downarrow & +\nearrow & \downarrow & +\nearrow & \downarrow & +\nearrow & \downarrow \\
\mathbf{1} & \diagup & \mathbf{1} & \diagup & \mathbf{0} & \diagup & \mathbf{0} & \diagup & \mathbf{1} & \diagup & \mathbf{0} & \diagup & \mathbf{1} & \diagup & \mathbf{0}
\end{array}$$

TABLE 2–7

American Standard Code for Information Interchange (ASCII).

Control Characters				Graphic Symbols											
Name	**Dec**	**Binary**	**Hex**	**Symbol**	**Dec**	**Binary**	**Hex**	**Symbol**	**Dec**	**Binary**	**Hex**	**Symbol**	**Dec**	**Binary**	**Hex**
NUL	0	0000000	00	space	32	0100000	20	@	64	1000000	40	′	96	1100000	60
SOH	1	0000001	01	!	33	0100001	21	A	65	1000001	41	a	97	1100001	61
STX	2	0000010	02	”	34	0100010	22	B	66	1000010	42	b	98	1100010	62
ETX	3	0000011	03	#	35	0100011	23	C	67	1000011	43	c	99	1100011	63
EOT	4	0000100	04	$	36	0100100	24	D	68	1000100	44	d	100	1100100	64
ENQ	5	0000101	05	%	37	0100101	25	E	69	1000101	45	e	101	1100101	65
ACK	6	0000110	06	&	38	0100110	26	F	70	1000110	46	f	102	1100110	66
BEL	7	0000111	07	’	39	0100111	27	G	71	1000111	47	g	103	1100111	67
BS	8	0001000	08	(	40	0101000	28	H	72	1001000	48	h	104	1101000	68
HT	9	0001001	09	)	41	0101001	29	I	73	1001001	49	i	105	1101001	69
LF	10	0001010	0A	*	42	0101010	2A	J	74	1001010	4A	j	106	1101010	6A
VT	11	0001011	0B	+	43	0101011	2B	K	75	1001011	4B	k	107	1101011	6B
FF	12	0001100	0C	,	44	0101100	2C	L	76	1001100	4C	l	108	1101100	6C
CR	13	0001101	0D	–	45	0101101	2D	M	77	1001101	4D	m	109	1101101	6D
SO	14	0001110	0E	.	46	0101110	2E	N	78	1001110	4E	n	110	1101110	6E
SI	15	0001111	0F	/	47	0101111	2F	O	79	1001111	4F	o	111	1101111	6F
DLE	16	0010000	10	0	48	0110000	30	P	80	1010000	50	p	112	1110000	70
DC1	17	0010001	11	1	49	0110001	31	Q	81	1010001	51	q	113	1110001	71
DC2	18	0010010	12	2	50	0110010	32	R	82	1010010	52	r	114	1110010	72
DC3	19	0010011	13	3	51	0110011	33	S	83	1010011	53	s	115	1110011	73
DC4	20	0010100	14	4	52	0110100	34	T	84	1010100	54	t	116	1110100	74
NAK	21	0010101	15	5	53	0110101	35	U	85	1010101	55	u	117	1110101	75
SYN	22	0010110	16	6	54	0110110	36	V	86	1010110	56	v	118	1110110	76
ETB	23	0010111	17	7	55	0110111	37	W	87	1010111	57	w	119	1110111	77
CAN	24	0011000	18	8	56	0111000	38	X	88	1011000	58	x	120	1111000	78
EM	25	0011001	19	9	57	0111001	39	Y	89	1011001	59	y	121	1111001	79
SUB	26	0011010	1A	:	58	0111010	3A	Z	90	1011010	5A	z	122	1111010	7A
ESC	27	0011011	1B	;	59	0111011	3B	[	91	1011011	5B	{	123	1111011	7B
FS	28	0011100	1C	<	60	0111100	3C	\	92	1011100	5C	|	124	1111100	7C
GS	29	0011101	1D	=	61	0111101	3D	]	93	1011101	5D	}	125	1111101	7D
RS	30	0011110	1E	>	62	0111110	3E	^	94	1011110	5E	~	126	1111110	7E
US	31	0011111	1F	?	63	0111111	3F	_	95	1011111	5F	Del	127	1111111	7F

Error Codes

TABLE 2–8

The BCD code with parity bits.

Even Parity		Odd Parity	
P	BCD	*P*	BCD
0	0000	1	0000
1	0001	0	0001
1	0010	0	0010
0	0011	1	0011
1	0100	0	0100
0	0101	1	0101
0	0110	1	0110
1	0111	0	0111
1	1000	0	1000
0	1001	1	1001