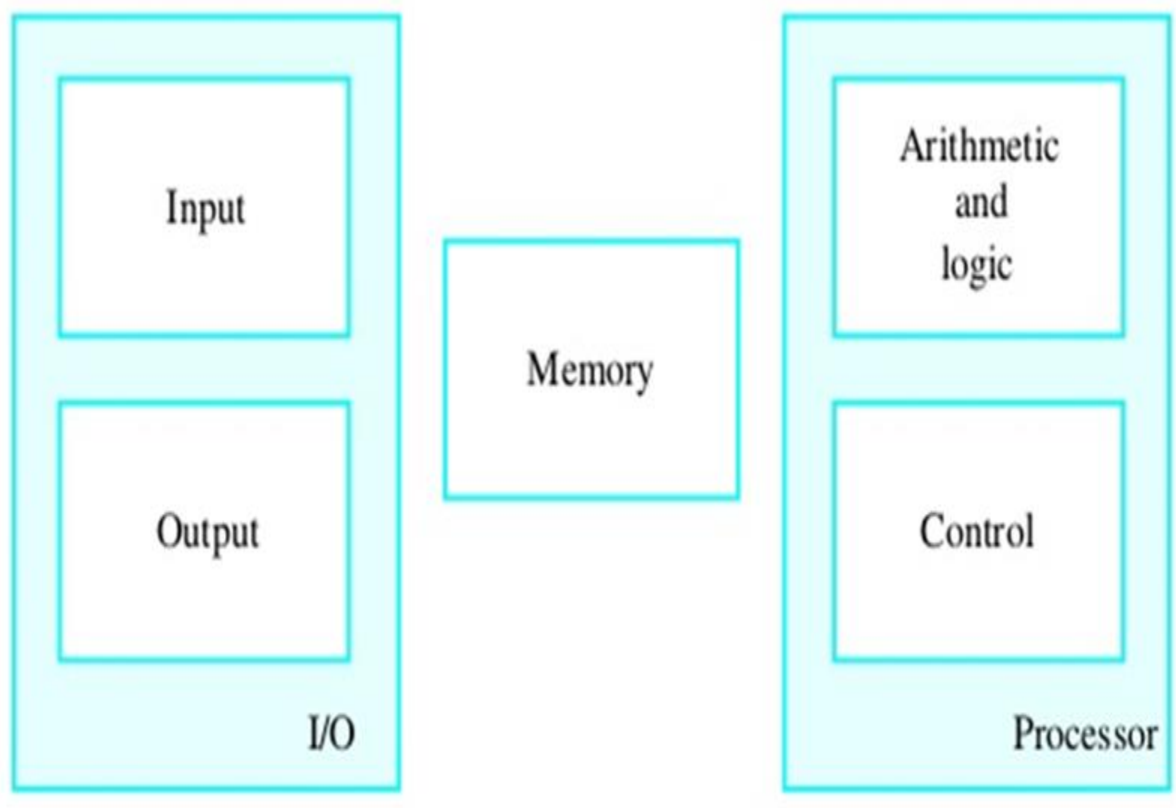
Computer Organization

- Computer organization is knowing, What the functional components of a computer are, how they work and how their performance is measured and optimized.

Introduction

- **Computer organization** is concerned with the way the hardware components are connected together to form a computer system.
- The subject '**Computer Organization**' provides the basic knowledge necessary to understand the hardware operation of digital computers.
- It presents the various digital components used in the organization and design of digital computers.

Introduction



Basic functional units of a computer.

Unit – 1
Part - A

- **Digital Systems and Binary Numbers**
  - Digital Systems
  - Binary Numbers
  - Number Base Conversions
  - Octal and Hexadecimal Numbers
  - Complements
  - Signed Binary Numbers

Digital Systems

- Digital computer is the best-known example of a digital system
- Others are telephone switching exchanges, digital voltmeters, digital calculators, etc.
- A digital system manipulates discrete elements of information
- Discrete elements: electric impulses, decimal digits, letters of an alphabet, any other set of meaningful symbols

Digital Systems

- In a digital system, discrete elements of information are represented by signals
- Electrical signals (voltages & currents) are the most common.
- Present day systems have only two discrete values (binary).
- Alternative, many-valued circuits are less reliable
- A lot of information is already discrete and continuous values can be quantized (sampled)

Digital Systems

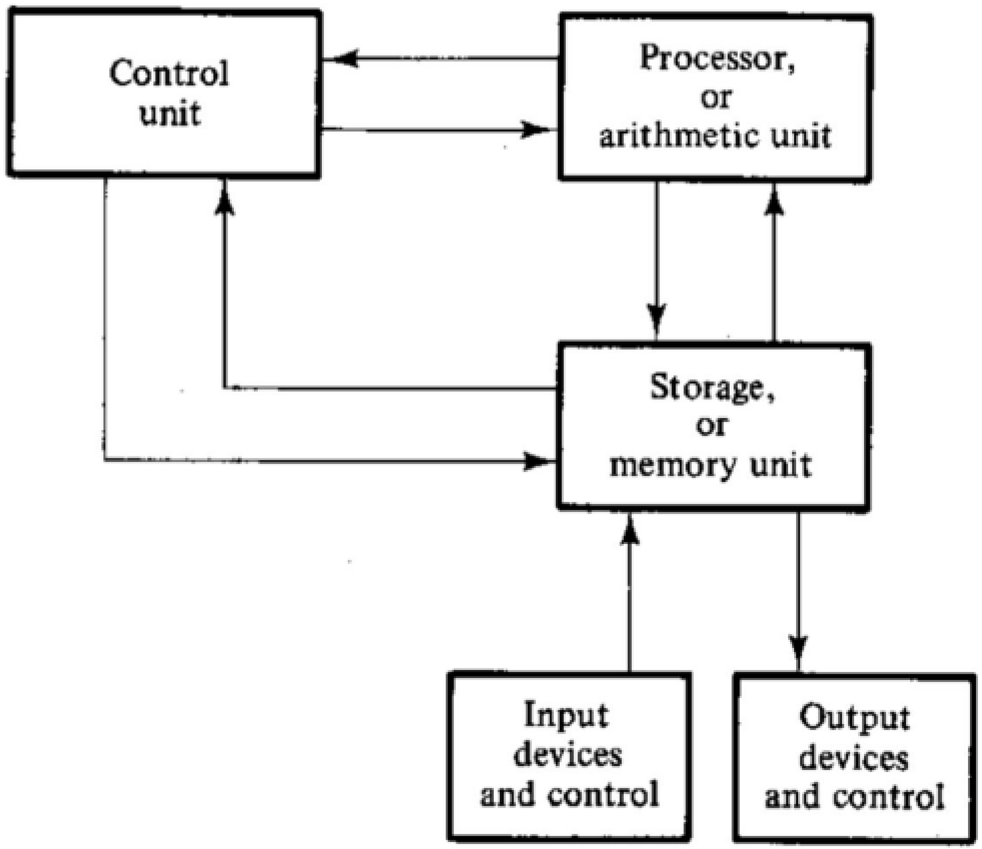


Fig: Block Diagram of a Digital Computer

Digital Systems

- A digital computer is an interconnection of digital modules
- To understand each module, it is necessary to have a basic knowledge of digital systems

BINARY NUMBERS

- $(1010.011)_2$ = 1*2^3+1*2^1+1*2^-2+1*2^-3
- $(630.4)_8$ = 408.5
- $(374.3)_5$ = wrong question

Binary Addition

- The rules for binary addition are the following

0+0=0

0+1=1

1+0=1

1+1=10 i.e 0 with a carry of 1

1+1+1=11 i.e 1 with a carry of 1

Binary Addition

- Example

8 4 2 1

1 0 0 1 +

1 0 1 1

\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-

Binary Addition

- Example

8 4 2 1

1 0 0 1

+

1 0 1 1

1

0

Binary Addition

- Example

8 4 2 1

1 0 0 1

\+

1 0 1 1

1 1

----

0 0

Binary Addition

- Example

8 4 2 1

1 0 0 1 +

1 0 1 1

1 1

----

1 0 0

Binary Addition

- Example

8 4 2 1

1 0 0 1 +

1 0 1 1

1 1

----

1 0 1 0 0

Binary Addition

- Example 2

  1 0 1 1 0 1 +

  1 0 0 1 1 1

Binary Addition

- Example 2

1 0 1 1 0 1 +

1 0 0 1 1 1

1 1 1 1

----

1 0 1 0 1 0 0

Binary Subtraction

- The binary subtraction is performed in a manner similar to that in decimal subtraction. The rules for binary subtraction are the following

0-0=0

1-1=0

1-0=1

0-1=1, with a borrow of 1

Binary Subtraction

- Example

  8 4 2 1

  1 0 1 0

     1 1 1 -

Binary Subtraction

- Example

8 4 2 1

0

1 0 ~~1~~ 0

-

1 1 1

----

1 (10-1=1)

Binary Subtraction

- Example

 8 4 2 1

 0 1 0

 ~~1~~ ~~0~~ ~~1~~ 0

   1 1 1 -

----

   1 1   (10-1=1)

Binary Subtraction

- Example

$$
\begin{array}{cccc}
\color{red}{8} & \color{red}{4} & \color{red}{2} & \color{red}{1} \\
0 & 1 & 0 & \\
\cancel{1} & \cancel{0} & \cancel{1} & 0 \\
 & 1 & 1 & 1 \quad -
\end{array}
$$

$$
\begin{array}{cccc}
 & 0 & 1 & 1
\end{array}
\qquad (1\text{-}1=0)
$$

Binary Subtraction

- Example

  8 4 2 1

  0 1 0

  ~~1~~ ~~0~~ ~~1~~ 0

    1 1 1 -

----

  0 0 1 1 (0-0=0)

Binary Subtraction

- Example 2:

```
 1 0 1 1 0 1  -
 1 0 0 1 1 1
---------------
```

Binary Subtraction

- Example 2:

$$
\begin{array}{r}
0\ 0\quad\ \ \\
1\ 0\ \cancel{1}\ \cancel{1}\ 0\ 1 \\
-\ \ 1\ 0\ 0\ 1\ 1\ 1 \\
\hline
\mathbf{0\ 0\ 0\ 1\ 1\ 0}
\end{array}
$$

Binary Multiplication

- The rules for binary multiplication are the following

0x0=0

1x1=1

1x0=0

0x1=0

Binary Multiplication

Example

```
   1 1 1 1
           X
     1 0 1
---------------
```

Binary Multiplication

Example

```
  1 1 1 1
    1 0 1   X
---------------
  1 1 1 1
```

1 1 1 1 X 1 = 1 1 1 1

Binary Multiplication

Example

1 1 1 1 X

1 0 1

----

1 1 1 1

0 0 0 0

1 1 1 1 X 0 = 0 0 0 0

Binary Multiplication

Example

```
    1 1 1 1  X
      1 0 1
  ---------------
    1 1 1 1
  0 0 0 0
1 1 1 1
```

1 1 1 1 X 1 = 1 1 1 1

Binary Multiplication

Example

```
     1 1 1 1
       1 0 1   X
   ---------------
     1 1 1 1
   0 0 0 0
 1 1 1 1
 1 1 1
 -------------
1 0 0 1 0 1 1
```

Binary Multiplication

Example 2

```
          1 0 1 1
            0 1 0   X
        -------------
   0 0 0 0
  1 0 1 1
0 0 0 0
-------------
010110
```

NUMBER-BASE CONVERSIONS

- Convert decimal 41 to binary

Convert decimal 41 to binary

$$\begin{array}{r|l} 2 & 41 \\ \hline & 20 - 1 \end{array}$$

Convert decimal 41 to binary

2 | 41
2 | 20 — 1
  | 10 — 0

Convert decimal 41 to binary

2 | 41
2 | 20 — 1
2 | 10 — 0
   | 5 — 0

Convert decimal 41 to binary

$$
\begin{array}{r|l}
2 & 41 \\
\hline
2 & 20 \;—\; 1 \\
\hline
2 & 10 \;—\; 0 \\
\hline
2 & 5 \;—\; 0 \\
\hline
 & 2 \;—\; 1 \\
\hline
\end{array}
$$

Convert decimal 41 to binary

2 | 41
2 | 20 — 1
2 | 10 — 0
2 | 5 — 0
2 | 2 — 1
  | 1 — 0

Convert decimal 41 to binary

2	41	
2	20	— 1
2	10	— 0
2	5	— 0
2	2	— 1
2	1	— 0
	0	— 1

Convert decimal 41 to binary

2	41	
2	20	1
2	10	0
2	5	0
2	2	1
2	1	0
	0	1

$$(41)_{10} = (101001)_2$$

- Convert decimal 52 to binary

convert $(41)_{10}$ to an octal number

$$\begin{array}{r|l} 8 & 41 \\ \hline & 5 - 1 \end{array}$$

Convert $(41)_{10}$ to an octal number

$$
\begin{array}{r|l}
8 & 41 \\
\hline
8 & 5 - 1 \\
\hline
 & 0 - 5
\end{array}
$$

convert $(41)_{10}$ to an octal number

$$\begin{array}{r|l}
8 & 41 \\ \hline
8 & 5 - 1 \\ \hline
 & 0 - 5
\end{array}$$

$(41)_{10} = (51)_8$

- Convert decimal 153 to octal

Convert the decimal $(153)_{10}$ to octal

$$
\begin{array}{r|l}
8 & 153 \\ \hline
8 & 19 - 1 \\ \hline
2 & 2 - 3 \\ \hline
 & 0 - 2
\end{array}
$$

read from bottom to up

$(153)_2 = (231)_8$

- Convert decimal 378 to octal

Successive division		Remainders
8	378	
8	47	↑ 2
8	5	7
	0	5

rs from bottom to top. Therefore, $378_{10} = 572_8$.

- Convert decimal 2598 to Hexadecimal

Successive division

		Remainder	
		Decimal	*Hex*
16	2598		
16	162	6	6
16	10	2	2
	0	10	A

ders upwards, $2598_{10} = A26_{16}$.

- Convert decimal 1479 to Hexadecimal

- Convert decimal 1479 to Hexadecimal

Answer – 5C7

- Convert Decimal Fraction 0.6875 to binary

Convert Decimal fraction $(0.6875)_{10}$ to binary

$0.6875 \times 2 = 1.3750 \rightarrow a_{-1} = 1$

$0.3750 \times 2 = 1.75 \rightarrow a_{-2} = 0$

$0.75 \times 2 = 1.5 \rightarrow a_{-3} = 1$

$0.5 \times 2 = 1.0 \rightarrow a_{-4} = 1$

read the integery from top to buttom

$(0.6875)_{10} =$

$\hookrightarrow (0.a_{-1}a_{-2}a_{-3}a_{-4})_2$

$\hookrightarrow (0.1011)_2$

- Convert Decimal Fraction 0.15 to binary

Convert Decimal fraction $(0.15)_{10}$ to binary

$0.15 \times 2 = 0.30 \rightarrow a_{-1} = 0$

$0.30 \times 2 = 0.60 \rightarrow a_{-2} = 0$

$0.60 \times 2 = 1.2 \rightarrow a_{-3} = 1$

$0.2 \times 2 = 0.4 \rightarrow a_{-4} = 0$

$0.4 \times 2 = 0.8 \rightarrow a_{-5} = 0$

$0.8 \times 2 = 1.6 \rightarrow a_{-6} = 1$

$0.6 \times 2 = 1.2 \rightarrow a_{-7} = 1$

repeated, so stop here.

$(0.15)_{10} = (0.001001)_2$

- Convert Decimal Fraction 0.513 to octal

Convert Decimal fraction $(0.513)_{10}$ to octal

$0.513 \times 8 = 4.104 \rightarrow a_{-1} = 4$

$0.104 \times 8 = 0.832 \rightarrow a_{-2} = 0$

$0.832 \times 8 = 6.656 \rightarrow a_{-3} = 6$

$0.656 \times 8 = 5.248 \rightarrow a_{-4} = 5$

$0.248 \times 8 = 1.984 \rightarrow a_{-5} = 1$

$0.984 \times 8 = 7.872 \rightarrow a_{-6} = 7$

$(0.513)_{10} = (0.406517\ldots)_8$

- Convert Decimal Fraction 0.93 to octal

Convert Decimal fraction $(0.93)_{10}$ to Octal

$0.93 \times 8 = 7.44 \rightarrow a_{-1} = 7$

$0.44 \times 8 = 3.52 \rightarrow a_{-2} = 3$

$0.52 \times 8 = 4.16 \rightarrow a_{-3} = 4$

$0.16 \times 8 = 1.28 \rightarrow a_{-4} = 1$

$0.28 \times 8 = 2.24 \rightarrow a_{-5} = 2$

$0.24 \times 8 = 1.92 \rightarrow a_{-6} = 1$

$0.92 \times 8 = 7.36 \rightarrow a_{-7} = 7$

$(0.93)_{10} = (0.7341217)_8$

- Convert Decimal Fraction 0.675 to Hexadecimal

Convert Decimal fraction $(0.675)_{10}$ to Hexadecimal

~~0.675~~ $\times$ 16

$0.675 \times 16 = 10.8 \rightarrow a_{-1} = 10 = A$

$0.8 \times 16 = 12.8 \rightarrow a_{-2} = 12 = C$

$0.8 \times 16 = 12.8 \rightarrow a_{-3} = 12 = C$

repeat

Stop up to here

$(0.675)_{10} = (AC)_{16}.$

Problems

- Convert decimal 41.6875 to binary
- Convert decimal 153.513 to octal
- Convert decimal 2598.675 to Hexadecimal

Octal and Hexadecimal Numbers

- The conversion from and to binary, octal and hexadecimal plays an important part in digital computers.
- Since 2^3 = 8 and 2^4 = 16, each octal digit corresponds to three binary digits and each hexadecimal digit corresponds to four binary digits.

Discuss

- Binary to Octal
- Octal to Binary
- Binary to Hexadecimal
- Hexadecimal to Binary

Decimal (base 10)	Binary (base 2)	Octal (base 8)	Hexadecimal (base 16)
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Binary to Octal

- To convert a binary number to an octal number, starting from the binary point make groups of 3 bits each, on either side of the binary point, and replace each 3-bit binary group by the equivalent octal digit.

Binary to Octal

- Covert the following binary number to octal

  10110001101011

10 110 001 101 011

2 6 1 5 3

Binary to Octal

- Covert the following binary number to octal

  0.111100000110

0.111 100 000 110

7 4 0 6

Binary to Octal

- Covert the following binary number to octal

1. 110101.101010
2. 10101111001.0111

Binary to Octal

- Covert the following binary number to octal

1. 110101.101010

   Ans 65.52

2. 10101111001.0111

   Ans 2571.34

Octal to Binary

To convert a given octal number to a binary, just replace each octal digit by its 3-bit binary equivalent

Octal to Binary

- Convert the octal number 367 to binary

Answer

Check whether the given number is octal or not?

Given octal number is 367

Convert each octal digit to binary

3	6	7
011	110	111

Final answer is 011110111

Octal to Binary

- Convert the octal number 0.52 to binary

Answer

Check whether the given number is octal or not?

Given octal number is 0.52

Convert each octal digit to binary

5	2
101	010

Final answer is 0.101010

Octal to Binary

- Convert the octal number 673.124 to binary

Answer-

Octal to Binary

- Convert the octal number 673.124 to binary

Answer-

6 7 3 . 1 2 4

110 111 011 . 001 010 100

Binary to Hexadecimal

- To convert a binary number to a hexadecimal number, starting from the binary point, make groups of 4 bits each, on either side of the binary point, and replace each 4-bit binary group by the equivalent hexadecimal digit.

Binary to Hexadecimal

- Covert the following binary number to hexadecimal

1011011011

Answer

Make groups of 4 bits and replace each 4-bit group by hexadecimal digit.

Given binary number is 1011011011

0010 1101 1011

2 D B

Final answer is 2DB

Binary to Hexadecimal

- Covert the following binary number to hexadecimal

0.011111

Answer

0 . 0111 1100

0 . 7 C

Final answer is 0.7C

Binary to Hexadecimal

- Covert the following binary number to hexadecimal

10111001101011.1111011

Binary to Hexadecimal

- Covert the following binary number to hexadecimal

10111001101011.1111011

Answer

0010 1110 0110 1011 . 1111 0110

2 E 6 B F 6

Final answer is 2E6B.F6

Hexadecimal to Binary

- To convert a given hexadecimal number to a binary, just replace each hexa digit by its 4-bit binary equivalent.

Hexadecimal to Binary

- Convert the following hexadecimal value to binary.

4BAC

Answer

4	B	A	C
0100	1011	1010	1100

Final answer is 0100101110101100

Hexadecimal to Binary

- Convert the following hexadecimal value to binary.

0.B0D

Answer

0	.	B	0	D
0	.	1011	0000	1101

Final answer is 0.101100001101

Hexadecimal to Binary

- Convert the following hexadecimal value to binary.

306.D

Answer ?

Problems

- Convert binary to Octal:
  10110001101011.111100000110
- Convert binary to Hexadecimal:
  10110001101011.11110000011
- Convert Octal to binary:
  673.124
- Convert Hexadecimal to binary:
  306.D
- Convert from Hexadecimal to Decimal:
  37B

Complements

- Complements are used in digital computers to simplify the subtraction operation and for logical manipulation.
- Ex- a-b=a+(-b) for changing “b” to “-b” we are using complements.
- Simplifying operations leads to simpler, less expensive circuits to implement the operations.

Complements

- There are two types of complements for each base-r system:

1. The radix complement
2. The diminished radix complement.

- The first is referred to as the r's complement and the second as the (r - 1)'s complement.
- When the value of the base r is substituted in the name, the two types are referred to as the 2's complement and 1's complement for binary numbers and the 10's complement and 9's complement for decimal numbers.

Diminished radix complement

- Given a number N in base r having n digits, the (r – 1)’s complement of N is defined as (r – 1) – N.
- For decimal numbers, r = 10 and r – 1 = 9, so the ninth complement of N is (10^n – 1) – N.
- Now, 10^n represents a number that consists of a single 1 followed by n 0’s.
- 10^n – 1 is a number represented by n 9’s.
- Ex- Decimal 45, here N=45 r=10 n=2

Diminished radix complement

- If n = 4 -> 10^4 = 10,000 and 10^4 – 1= 9999
- Find the 9's complement for the following

546700

Answer:- 9 9 9 9 9 9

- 5 4 6 7 0 0

----

4 5 3 2 9 9

Diminished radix complement

- Find the 9’s complement for the following

012398

Answer:-?

Diminished radix complement

- For binary numbers, $r = 2$ and $r – 1 = 1$, so the 1's complement of N is $(2^n – 1) – N$.
- 2^n is represented by a binary number that consists of a 1 followed by n 0's.
- If $n = 4$ -> $2^4 = (10000)$ base 2 and $2^4 – 1 = (1111)$ base 2.

Diminished radix complement

- Find the 1's complement for the following

1011000

Ans-0100111

Ex-2

0101101

Ans-?

Radix complement

The *r's* complement of *n*-digit number *N* in base *r* is defined as $r^n - N$ for $N \neq 0$ and 0 for $N = 0$. Comparing with the $(r - 1)$'s complement, the r's complement is obtained by adding 1 to the $(r - 1)$'s complement since $r^n - N = [(r^n - 1) - N] + 1$.

Radix complement

- Find the 10’s complement of 012398

Answer-

9’s complement of 012398 is 987601

10’s complement is 9’s complement + 1

Final answer is 987602

Radix complement

- Find the 10's complement of 246700.

Answer:753300

Radix complement

- Find the 2's complement of 1101100
- Answer-

1's complement of 1101100 is 0010011

2's complement is 1's complement + 1

```
        0  0  1  0  0  1  1
                    1  1  1
------------------------------
        0  0  1  0  1  0  0
------------------------------
```

Final answer is 0010100

Radix complement

- Find the 2’s complement of 0110111

Answer-?

1 0 0 1 0 0 0

1

----

1 0 0 1 0 0 1

Subtraction with Complements

The subtraction of two n-digit unsigned numbers $M - N$ in base r can be done as follows:

1. Add the minuend M to the r's complement of the subtrahend N. Mathematically, $M + (r^n - N) = M - N + r^n$.
2. If $M \geq N$, the sum will produce an end carry r^n, which can be discarded; what is left is the result $M - N$.
3. If $M < N$, the sum does not produce an end carry and is equal to $r^n - (N - M)$, which is the r's complement of $(N - M)$. To obtain the answer in a familiar form, take the r's complement of the sum and place a negative sign in front.

Subtraction with Complements

- Using 10's complement,
  subtract 72532 – 3250
  M - N

Ans:- let M=72532, N=3250 here M>N

10's complement of N=99999-3250+1=96750

Sum=72532+96750=169282

- If M>=N, the sum will produce an end carry r^n, which can be discarded; what is left is the result M - N.
- Final Answer is 69282

Subtraction with Complements

- Using 10’s complement,

  subtract 3250 – 72532

  M - N

Ans:- let M=3250, N=72532 here $M<N$

10’s complement of N=99999-72532+1=27468

Sum=03250+27468=30718

- If $M<N$, the sum does not produce an end carry and is equal to $r^n-(N-M)$, which is the r’s complement of (N-M). To obtain the answer in a familiar form, take the r’s complement of the sum and place a negative sign in front.
- There is no end carry. Therefore, the answer is –(10’s complement of 30718) = -(99999-30718+1)
- Final Answer is -69282

Subtraction with Complements

- Using 2’s complement,
  subtract 1010100 – 1000011
  X - Y

Ans:-let X=1010100, Y=1000011

2’s complement of Y is 1’s Complement of Y + 1.

So, 2’s complement of Y is 0111100+1=0111101

Sum=1010100+0111101

1 0 1 0 1 0 0

0 1 1 1 1 0 1

1 1 1 1

----

1 0 0 1 0 0 0 1

Discard carry end carry

Final Answer X-Y=0010001

Subtraction with Complements

- Using 2's complement,

  subtract 1000011 - 1010100

  X - Y

Ans:-let X=1000011, Y=1010100

2's complement of Y is 1's Complement of Y + 1.

So, 2's complement of Y is 0101011+1=0101100

Sum=1000011+0101100

```
1 0 0 0 0 1 1
0 1 0 1 1 0 0
---------------
1 1 0 1 1 1 1
```

There is no end carry. Therefore, the answer is X-Y = -(2's complement of 1101111)

Final Answer X-Y=-0010001

Subtraction with Complements

- Using 1's complement,
  subtract 1010100 – 1000011
  X - Y

Ans:-let X=1010100, Y=1000011

1's Complement of Y is 0111100

Sum=1010100+0111100

1 0 1 0 1 0 0
0 1 1 1 1 0 0
1 1 1 1

----

1 0 0 1 0 0 0 0

If you get carry, remove the end carry and add 1 to the sum.

Final Answer X-Y=0010000+1=0010001

Subtraction with Complements

- Using 1's complement,

  subtract 1000011 - 1010100

  X - Y

Ans:-let X=1000011, Y=1010100

1's Complement of Y is 0101011.

Sum=1000011+0101011

```
 1 0 0 0 0 1 1
 0 1 0 1 0 1 1
         1 1
---------------
 1 1 0 1 1 1 0
```

There is no end carry. Therefore, the answer is X-Y = -(1's complement of 1101110)

Final Answer X-Y=-0010001

Signed Binary Numbers

- It is customary to represent the sign with a bit placed in the leftmost position of the number and to make it 0 for positive and 1 for negative.
- Consider the number 9 represented in binary with 8 bits. +9 is represented with sign bit 0 in the leftmost position followed by the binary equivalent of 9 to give 00001001.

	00001001(+9)
Signed magnitude	-> 10001001(-9)
Signed-1's complement	-> 11110110(-9)
Signed-2's complement	-> 11110111(-9)

Arithmetic Addition

- The addition of 2 numbers in the signed-magnitude system follows the rules of ordinary arithmetic.
- If the signs are the same, we add the two magnitudes and give the sum the common sign.
- If the signs are different, we subtract the smaller magnitudes from the larger and give the result the sign of the larger magnitude.

Arithmetic Addition

- For example, (+25) + (-37) = -(37 - 25) = -12 is done by subtracting the smaller magnitude, 25, from the larger magnitude, 37, and appending the sign of 37 to the result.
- This is a process that requires a comparison of the signs and magnitudes and then performing either addition or subtraction.

Arithmetic Addition

- The same procedure applies to binary numbers in signed-magnitude representation.
- In contrast, the rule for adding numbers in the signed-complement system does not require a comparison or subtraction, but only addition.
- The procedure is very simple and can be stated as follows for binary numbers:
- **The addition of two signed binary numbers with negative numbers represented in signed-2's-complement form is obtained from the addition of the two numbers, including their sign bits. A carry out of the sign-bit position is discarded.**

Arithmetic Addition

+ 6	0 0 0 0 0 1 1 0 (+6 in 8 bits)
+13	0 0 0 0 1 1 0 1 (+13 in 8 bits)
------	1 1
+19	------------------
------	0 0 0 1 0 0 1 1

Arithmetic Addition

```
-  6          1 1 1 1 1 0 1 0 (-6 in 2's complement)
+13           0 0 0 0 1 1 0 1 (+13 in 8 bits)
------         1 1 1 1
+ 7           ----------------
------      1 0 0 0 0 0 1 1 1
               ----------------
```

Arithmetic Addition

```
+  6        0 0 0 0 0 1 1 0 (+6 in 8 bits)
-13          1 1 1 1 0 0 1 1 (-13 in 2's complement)
------                   1 1
- 7         -----------------
------       1 1 1 1 1 0 0 1
            -----------------
```

Arithmetic Addition

```
-  6        1 1 1 1 1 0 1 0 (-6 in 2's complement)
-13         1 1 1 1 0 0 1 1 (-13 in 2's complement)
------      1 1 1     1
- 19       --------------------
------    1 1 1 1 0 1 1 0 1
           --------------------
```

- Note - Negative numbers must be in 2's complement and that the sum obtained after the addition if negative is in 2's-complement form.

Arithmetic Subtraction

- Subtraction of two signed binary numbers when negative numbers are in 2’s-complement form is simple and can be stated as follows:
- Take the 2’s complement of the subtrahend (including the sign bit) and add it to the minuend (including the sign bit). A carry out of the sign-bit position is discarded.

Arithmetic Subtraction

- This procedure is adopted because a subtraction operation can be changed to an addition operation if the sign of the subtrahend is changed, as is demonstrated by the following relationship:

$$(\pm A) - (+B) = (\pm A) + (-B)$$
$$(\pm A) - (-B) = (\pm A) + (+B)$$

Arithmetic Subtraction

$(+A)-(+B)=(+A)+(-B)$

$(-A)-(+B)=(-A)+(-B)$

$(+A)-(-B)=(+A)+(+B)$

$(-A)-(-B)=(-A)+(+B)$

Arithmetic Subtraction

- Changing a positive number to a negative number is easily done by taking the 2’s complement of the positive number. The reverse is also true, because the complement of a negative number in complement form produces the equivalent positive number.

Arithmetic Subtraction

```
+13          0 0 0 0 1 1 0 1 (+13 in 8 bits)
-  6         1 1 1 1 1 0 1 0 (-6 in 2's complement)
------        1 1 1 1
+ 7          ------------------
------     1 0 0 0 0 0 1 1 1
              ------------------
```

- Therefore, **computers need only one common hardware circuit to handle both types of arithmetic.**