DIGITAL ELECTRONICS

By Prof. Pritosh Mishra

UNIT-1(PART-B)

[Binary Arithmetic]

Binary Addition

- The steps used for a computer to complete addition are usually greater than a human, but their processing speed is far superior.

RULES

- 0 + 0 = 0
- 0 + 1 = 1
- 1 + 0 = 1
- 1 + 1 = 0 (With 1 to carry)
- 1 + 1 + 1 = 1 (With 1 to carry)

Binary Addition

EXAMPLE

$$\begin{array}{r} 1\ 0\ 0\ 1 \\ +\ 1\ 0\ 1\ 1 \\ \hline \end{array}$$

Activity 1

Perform the following additions in binary.

- $101_{10} + 40_{10} =$
- $320_{10} + 18_{10} =$
- $76_{10} + 271_{10} =$

Binary Subtraction

- Computers have trouble performing subtractions so the following rule should be employed:

"X – X is the same as

X + -X"

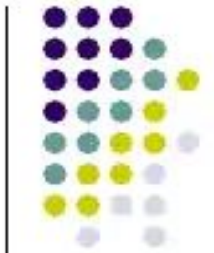
- This is where two's complement is used.

Binary Subtraction

RULES

2. Convert the number to binary.
3. Perform two's complement on the second number.
4. Add both numbers together.

Binary Subtraction



EXAMPLE 1

Convert 12 - 8 using two's complement.

3. Convert to binary

   $12 = 00001100_2$

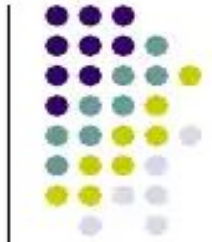
   $8 = 00001000_2$

- Perform one's complement on the 8_{10}

  00001000_2

  11110111_2

Binary Subtraction



EXAMPLE 1

2. Perform two's complement.

$$\begin{array}{r} 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1_2 \\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1_2 \quad + \\ \hline 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0_2 \end{array}$$

6. Add the two numbers together.

   $= \cancel{1}\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0_2$ (Ignore insignificant bits)

Binary Subtraction

EXAMPLE 2

What happens if the first number is larger than the second?

Try 6_{10} - 10_{10}

Binary Subtraction

EXAMPLE 2

2. Convert to binary

   $6 = 00000110_2$

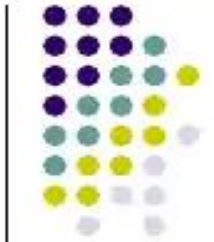
   $10 = 00001010_2$

- Perform one's complement on the 10_{10}

  00001010_2

  11110101_2

Binary Subtraction



EXAMPLE 2

2. Perform two's complement.

$$\begin{array}{r} 1\ 1\ 1\ 1\ 0\ 1\ 0\ 1_2 \\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1_2{}^{+} \\ \hline = 1\ 1\ 1\ 1\ 0\ 1\ 1\ 0_2 \end{array}$$

Binary Subtraction

EXAMPLE 2

2. Add the two numbers together.

$0\ 0\ 0\ 0\ 0\ 1\ 1\ 0_2$

$1\ 1\ 1\ 1\ 0\ 1\ 1\ 0_2$ +

$= 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0_2$ (Ends with a negative bit)

Binary Subtraction

EXAMPLE 2

2. Perform one's complement on the result

   $1\ 1\ 1\ 1\ 1\ 1\ 0\ 0_2$

   $0\ 0\ 0\ 0\ 0\ 0\ 1\ 1_2$

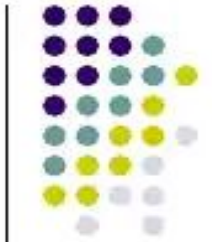
5. Add 1 to the result.

   $0\ 0\ 0\ 0\ 0\ 0\ 1\ 1_2$

   $0\ 0\ 0\ 0\ 0\ 0\ 0\ 1_2$ +

   $= 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0_2$

Binary Subtraction



EXAMPLE 2

2. We then add the sign bit back.

$0\ 0\ 0\ 0\ 0\ 1\ 0\ 0_2$

$= 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0_2$

Activity 2

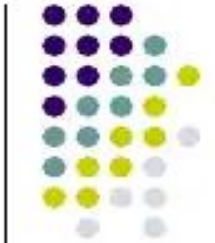
Perform the following subtractions.

3. 22 - 8 =
4. 76 - 11 =
5. 6 - 44 =

Binary Multiplication

- Multiplication follows the general principal of shift and add.
- The rules include:
  - 0 * 0 = 0
  - 0 * 1 = 0
  - 1 * 0 = 0
  - 1 * 1 = 1

Binary Multiplication



EXAMPLE 1

Complete 15 * 5 in binary.

3. Convert to binary

   $15 = 00001111_2$

   $5 = 00000101_2$

6. Ignore any insignificant zeros.

   ~~0000~~1111_2

   ~~00000~~101_2 x

Binary Multiplication

EXAMPLE 1

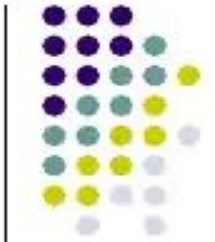
2. Multiply the first number.

$$\begin{array}{r} 1\ 1\ 1\ 1_2 \\ 1\ 0\ \boxed{1}_2 \quad \times \\ \hline 1\ 1\ 1\ 1 \end{array}$$

1111 x 1 = 1111

7. Now this is where the shift and takes place.

Binary Multiplication



EXAMPLE 1

2. Shift one place to the left and multiple the second digit.

$1\ 1\ 1\ 1_2$

$1\ 0\ 1_2$ x

$1\ 1\ 1\ 1$

$0\ 0\ 0\ 0\ 0$ ← Shift One Place

1111 x 0 = 0000

Binary Multiplication

EXAMPLE 1

2. Shift one place to the left and multiple the third digit.

$1\ 1\ 1\ 1_2$ x

$1\ 0\ 1_2$

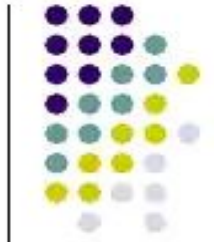
$1\ 1\ 1\ 1$

$0\ 0\ 0\ 0\ 0$

$1\ 1\ 1\ 1\ 0\ 0$ ← Shift One Place

1111 x 1 = 1111

Binary Multiplication



EXAMPLE 1

2. Add the total of all the steps.

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

8. Convert back to decimal to check.

Activity 3

Calculate the following using binary multiplication shift and add.

- $12 * 3 = 1\ 0\ 0\ 1\ 0\ 0_2$
- $13 * 5 = 1\ 0\ 0\ 0\ 0\ 0\ 1_2$
- $97 * 20 = 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0_2$
- $121 * 67 = 1\ 1\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1_2$

Binary Division

- Division in binary is similar to long division in decimal.
- It uses what is called a shift and subtract method.

Binary Division

EXAMPLE 1

Complete 575 / 25 using long division.

2. $25\overline{)575}$ (0)	• Take the first digit of 575 (5) and see if 25 will go into it. • If it can not put a zero above and take the next number.
2. $25\overline{)575}$ (02)	• How many times does 25 go into 57? •TWICE

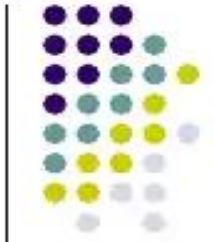
Binary Division

2. $25\overline{)575}$ 02 50 7	• How much is left over? • 57 – (25 * 2) = 7
2. $25\overline{)575}$ 02 $50\downarrow$ 75	• Drop down the next value

Binary Division

2. $25\overline{)575}$ 023 50 75	• Divide 75 by 25 • Result = 3
2. $25\overline{)575}$ 023 50 75 75 0	• Check for remainder • 75 – (3 * 25) = 0 • FINISH!

Binary Division



- Complete the following:
- 25/5

Step 1: Convert both numbers to binary.

25 = 1 1 0 0 1

5 = 1 0 1

Step 2: Place the numbers accordingly:

1 0 1 ⟌ 1 1 0 0 1

Binary Division

Step 3: Determine if 1 0 1 (5) will fit into the first bit of dividend.

1 0 1 ⟌ 1 1 0 0 1

1 0 1(5) will not fit into 1(1)

Step 4: Place a zero above the first bit and try the next bit.

Binary Division

Step 5: Determine if 1 0 1 (5) will fit into the next two bits of dividend.

```
      0
1 0 1 ) 1 1 0 0 1
```

1 0 1(5) will not fit into 1 1(3)

Step 6: Place a zero above the second bit and try the next bit.

Binary Division

Step 7: Determine if 1 0 1 (5) will fit into the next three bits of dividend.

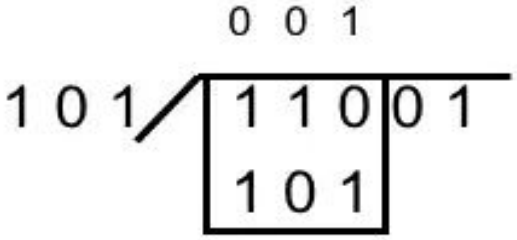
```
        0  0
1 0 1 / 1 1 0 | 0 1
```

1 0 1(5) will fit into 1 1 0(6)

Step 8: Place a one above the third bit and times it by the divisor (1 0 1)

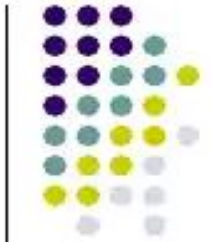
Binary Division

Step 9: The multiplication of the divisor should be placed under the THREE bits you have used.



A subtraction should take place, however you cannot subtract in binary. Therefore, the two's complement of the 2nd number must be found and the two numbers added together to get a result.

Binary Division

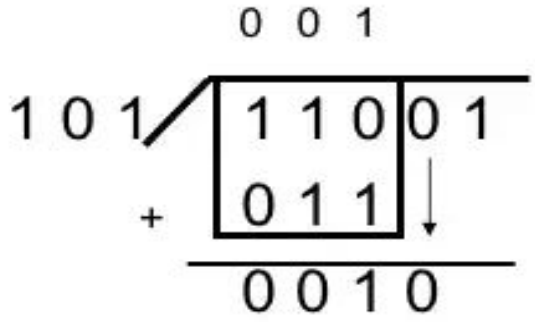


Step 10: The two's complement of 1 0 1 is 0 1 1

```
          0  0  1
      ┌──────────
1 0 1 /  1  1  0 | 0  1
      +  0  1  1 |
         ---------
         0  0  1
      ──────────
```

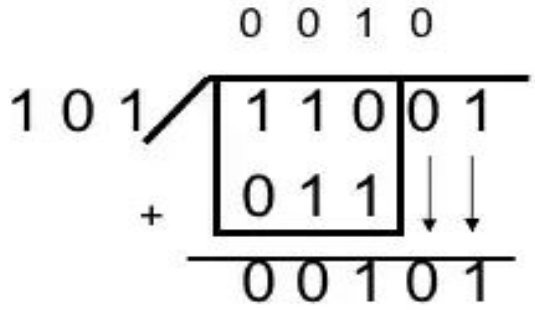
Binary Division

Step 11: Determine if 1 0 1 will fit into the remainder 0 0 1. The answer is no so you must bring down the next number.



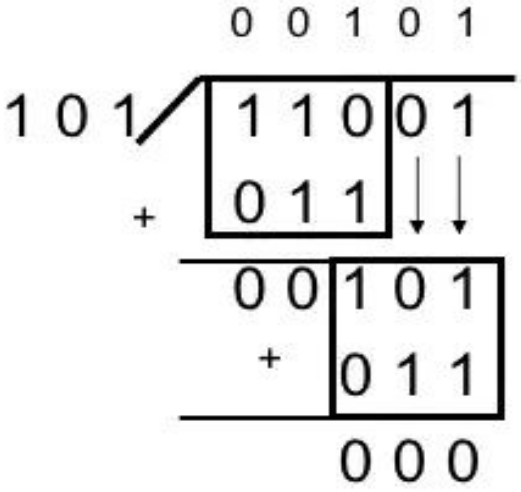
Binary Division

Step 12: 1 01 does not fit into 0 0 1 0. Therefore, a zero is placed above the last bit. And the next number is used.



Binary Division

Step 13: 1 0 1 does fit into 1 0 1 so therefore, a one is placed above the final number and the process of shift and add must be continued.



Activity 4

- Complete the following divisions:
  - 340 / 20
  - 580 / 17

Activity 5

- 40/4
- 36/7

--------THANK YOU--------