Digital Electronics

Binary Arithmetic
Gray code

1st Year of 4 year B.Tech.

Day 2

Arithmetic Operation with Binary numbers:

Addition:

Case	A	+	B	Sum	Carry
1	0	+	0	0	0
2	0	+	1	1	0
3	1	+	0	1	0
4	1	+	1	0	1

0011010 + 001100 = 00100110

$$\begin{array}{rl} \boxed{1\,1} & \text{carry} \\ 0\,0\,1\,1\,0\,1\,0 & = 26_{10} \\ +\,0\,0\,0\,1\,1\,0\,0 & = 12_{10} \\ \hline 0\,1\,0\,0\,1\,1\,0 & = 38_{10} \end{array}$$

Subtraction:

Case	A	-	B	Subtract	Borrow
1	0	-	0	0	0
2	1	-	0	1	0
3	1	-	1	0	0
4	0	-	1	0	1
5	10	-	1	1	0

11010 - 1100 =

1 2

0 1

$$\begin{array}{rccccc l} & \cancel{1} & \cancel{1} & 0 & 1 & 0 & (26)_{10} \\ - & & 1 & 1 & 0 & 0 & (12)_{10} \\ \hline & 0 & 1 & 1 & 1 & 0 & (14)_{10} \end{array}$$

Ex. 1001-110

Multiplication:

Case	A x B	Multiplication
1	0 x 0	0
2	0 x 1	0
3	1 x 0	0
4	1 x 1	1

Example:

0011010 x 001100 = 100111000

$$
\begin{array}{rl}
0\,0\,1\,1\,0\,1\,0 & = 26_{10} \\
\times\, 0\,0\,0\,1\,1\,0\,0 & = 12_{10} \\
\hline
0\,0\,0\,0\,0\,0\,0 & \\
0\,0\,0\,0\,0\,0\,0 & \\
0\,0\,1\,1\,0\,1\,0 & \\
0\,0\,1\,1\,0\,1\,0 & \\
\hline
0\,1\,0\,0\,1\,1\,1\,0\,0\,0 & = 312_{10}
\end{array}
$$

Division:

Case	A/B	Division
1	0 / 1	0
2	1 / 1	1
3	0 / 0	Not allowed
4	1 / 0	Not allowed

101010 / 000110 = 000111

$$
\begin{array}{rrl}
 & 1\,1\,1 & = 7_{10} \\
000110 \,\big) & \cancel{1}\,0\,1\,0\,1\,0 & = 42_{10} \\
 & -\,1\,1\,0 & = 6_{10} \\
\hline
 & \cancel{1}\,0\,0\,1 & \\
 & -\,1\,1\,0 & \\
\hline
 & 1\,1\,0 & \\
 & -\,1\,1\,0 & \\
\hline
 & 0 &
\end{array}
$$

1's complement:

- 1's complement of any binary number can be obtained by changing al 0s and 1s and all 1s to 0s

  For e.g , 1's complement of 1010111= 0101000 and 0101000=1010111

- Subtraction by 1's complement – allows subtraction only by addition

To subtract a smaller number from a larger number

- Determine 1's complement of smaller number
- Add this to the larger number
- Remove the carry and add it to the result

Subtract (1010) from (1111)

```
   1   1   1   1
-  1   0   1   0  ---->  Smaller number
```

So, 1's complement of smaller number = 0 1 0 1

```
       1   1   1
So,  1   1   1   1
   + 0   1   0   1
   ---------------
 1   0   1   0   0
 L-> +           1  ---> carry
   ---------------
     0   1   0   1
```

Answer is = 0 1 0 1

To subtract a larger number from a smaller number

- Determine 1's complement of larger number
- Add this to the smaller number
- Answer is 1's complement of the result and opposite in sign. There is no carry

Subtract (1010) from (1000)

```
   1   0   0   0
-  1   0   1   0  ---->  larger number
```

So, 1's complement of larger number = 0 1 0 1

```
       1   1   1
So,  1   0   0   0
   + 0   1   0   1
   ---------------
     1   1   0   1  ---> no carry
```

Answer is 1's complement of the above results and opposite in sign=

- 0 0 1 0

2's complement

- 2's complement of any binary number can be obtained by obtaining the 1's complement and adding 1 to it

For e.g , 2's complement of 1010111= 0101000 + 1= 0101001 and 0101000=1010111 + 1= 1 0 1 1 0 0 0

1's same

- Subtraction by 1's complement – allows subtraction only by addition

To subtract a smaller number from a larger number

- Determine 2's complement of smaller number
- Add this to the larger number
- Omit the carry

Subtract (1010) from (1111)

```
   1   1   1   1
-  1   0   1   0  ──→  Smaller number
```

So, 2's complement of smaller number = 0 1 0 1+1 = 0 1 1 0

```
        1     1
So,     1     1     1     1
     +  0     1     1     0
   ─────────────────────────
  ~~1~~ 0     1     0     1     (Omit carry)
```

Answer is = 0 1 0 1

To subtract a larger number from a smaller number

- Determine 2's complement of larger number
- Add this to the smaller number
- Answer is 2's complement of the result and negative. There is no carry
- To get answer, find 2's complement of result and change sign

Subtract (1010) from (1000)

```
   1   0   0   0
-  1   0   1   0  ──→  larger number
```

So, 2's complement of larger number = 0 1 0 1 +1 = 0 1 1 0

```
        1
So,     1     0     0     0
     +  0     1     1     0
   ─────────────────────────
        1     1     1     0
```

no carry

Answer is 2's complement of the above results and opposite in sign=

\- 0 0 1 0

9’ complement and 10’s complement:

- 9’s complement of a number can be found out by subtracting each digit of the number from 9
- 10’ complement is equal to its 9’s complement + 1

- 9’s complement of 146

$$\begin{array}{cccc} & 9 & 9 & 9 \\ - & 1 & 4 & 6 \\ \hline & 8 & 5 & 3 \end{array}$$

- 10’s complement = 853+1 = 854

9’s complement of 4397

$$\begin{array}{ccccc} & 9 & 9 & 9 & 9 \\ - & 4 & 3 & 9 & 7 \\ \hline & 5 & 6 & 0 & 2 \end{array}$$

10’s complement = 5602 + 1 = 5603

BCD or Binary Coded Decimal:

- In binary form $(12)_{10} = (1100)_2$
- In BCD, $(12)_{10}$ = [0001 0010]$_{BCD}$

- Therefore, BCD code for $(842)_{10}$ = [1000 0100 0010]$_{BCD}$ BCD code for $(96.42)_{10}$ = [1001 0110 . 0100 0010]$_{BCD}$

- BCD ADDITION:
  - Add the two numbers according to binary addition rule
  - If a four bit sum is equal to or less than $(9)_{10}$ i.e., $(1001)_2$, it is a valid BCD number
  - If the four bit sum is greater than 9, or a carry is generated, it is not a valid BCD number and we add $(6)_{10}$ i.e., $(0110)_2$, to skip the six invalid states in BCD (10-15) and return to BCD.
  - If a carry is generated, we add the carry to the next four bit group

Add the following BCD numbers:

- 1001 and 0100

$$
\begin{array}{cccccc}
 & 1 & 0 & 0 & 1 & \quad 9 \\
+ & 0 & 1 & 0 & 0 & \quad 4 \\
\hline
 & 1 & 1 & 0 & 1 & \longrightarrow \text{invalid BCD number} \\
+ & 0 & 1 & 1 & 0 & \longrightarrow \text{add 6} \\
\hline
\end{array}
$$

$$
\begin{array}{cccc|cccc}
0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\
\hline
 & & 1 & & & & 3 &
\end{array}
$$

Add the following BCD numbers:

	1	1	
478	0100	0111	1000
+137	0001	0011	0111
	0110 → 5	1011 → 3	1111 → 1st
	0000	0110	0110 +6
615	0110 → 6	0001 → 4	0101 → 2nd

Excess-3 code:

Decimal digit	BCD	Excess-3
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

Gray Code:

- Only one bit in the code group changes when moving from one step to the next
- Conversion of Binary number to Gray code:
  - The MSB of the Gray code is the same as the first bit of the binary number
  - The second bit of Gray code equals the exclusive-OR of the first and the second bits of the binary number, it will be 1 if code bits are different and 0 if the code bits are same
  - The third bit of Gray code bit equals the exclusive-OR of the second and third bits of the binary number and so o

Convert $(10110)_2$ to Gray code

Binary 1 0 1 1 0

Step1: 1 $\oplus$ 0 1 1 0

1 1

Step 2: 1 0 $\oplus$ 1 1 0

1 1 1

Step 3: 1 0 1 $\oplus$ 1 0

XOR

1 1 1 0

Step 4: 1 0 1 1 $\oplus$ 0

| 1 | 1 | 1 | 0 | 1 | Gray code |

Convert $(10110)_2$ to Gray code

Binary 1 0 1 0 1 1

Step1: 1 $\oplus$ 0 1 0 1 1

1 1

Step 2: 1 0 $\oplus$ 1 0 1 1

1 1 1

Step 3: 1 0 1 $\oplus$ 0 1 1

1 1 1 1

Step 4: 1 0 1 0 $\oplus$ 1 1

1 1 1 1 1

Step 5: 1 0 1 0 1 $\oplus$ 1

1 1 1 1 1 0

Gray code to Binary

Imp.

- Only one bit in the code group changes when moving from one step to the next
- Conversion of Gray code to Binary
  - The MSB is the same as the first bit of the Gray code
  - If the second bit of Gray code is 0, the second binary bit is the same as the first binary bit. If the second bit of Gray code is 1, the second binary bit is inverse of the first binary bit
  - Step 2 is repeated for each bit

0 - Same
1 - Reverse

<u>Convert the Gray code (110101) to Binary code</u>

Gray 1 1 0 1 0 1

Step1: 1 1 0 1 0 1

Reverse

1 0

Step 2: 1 1 0 1 0 1

Same

1 0 0

Step 3: 1 1 0 1 0 1

1 0 0 1

Step 4: 1 1 0 1 0 1

1 0 0 1 1

Step 5: 1 1 0 1 0 1

$\boxed{1\ 0\ 0\ 1\ 1\ 0}$

Binary code

Convert $(1010111)_G$ to Binary code

Gray 1 0 1 0 1 1

Step1: 1 0 1 0 1 1

1 1

Step 2: 1 0 1 0 1 1

1 1 0

Step 3: 1 0 1 0 1 1

1 1 0 0

Step 1 0 1 0 1 1

1 1 0 0 1

Step 5: 1 0 1 0 1 1

$\boxed{1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0}$

Binary