1

I .1– The Gray Code

1- Definition of the Gray Code

Gray code, also known as **Gray code** or **reflected binary code**, is a type of binary coding that allows you to change only one bit at a time when a number is increased by one unit. This property is important for several applications.

The name of the code comes from the American engineer **Frank Gray** who published a patent on this code in 1953, but the code itself is older.

2- Gray Code Principle

The Gray code is a binary coding, that is to say a function that associates with each number a binary representation. This method is different from natural binary coding. The following table partially shows the 4-bit coding (only the first 8 values are presented, the next eight with the first bit at 1 are not).

Decimal Coding	Natural Binary Coding	Gray Coding or Reflected Binary
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100

The main difference between the two is that the Gray coding of two consecutive numbers differs by only one position. For example 5 is coded by 0111, and 6 is coded by 0101: here only the second bit changes.

3- <u>Binary code to Gray code conversion</u>

Let B be a number written in pure natural binary on N bits

$B = B_N \ldots\ldots B_4B_3B_2B_1$; B_N is the most significant bit and B_1 is the least significant bit

G is the Gray code equivalent of the number B also written on N bits

$G_{(Gray)} = G_N \ldots\ldots\ldots G_4G_3G_2G_1$

The transition from pure binary to Gray code is done according to the following two steps:

1- $G_N = B_N$
2- $G_{N-1} = B_N + B_{N-1}$

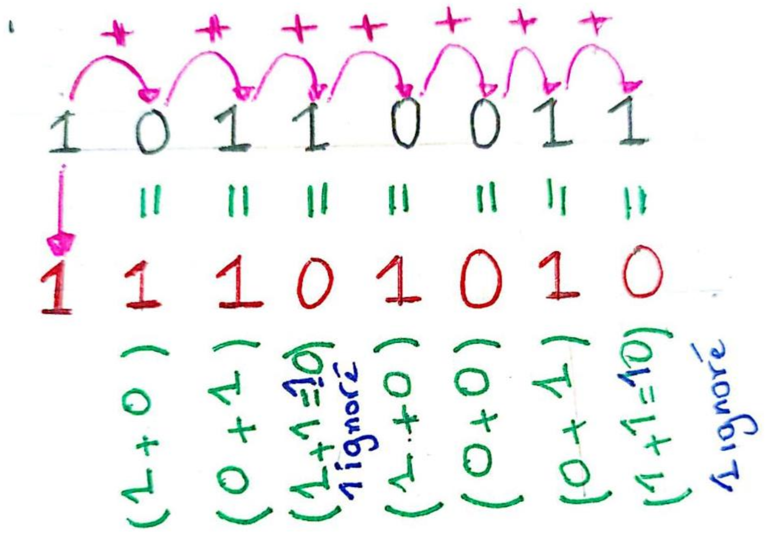

Figure 1: Converting a binary number into Gray code

Note

- ➢ The most significant bit of the binary number is always equal to the most significant bit of its Gray code equivalent.
- ➢ When adding bits, the case of (1 +1 = 0) because the carry will be ignored (See Figure 1 for the cases of the first bit and the 5^{th} bit.

4- <u>Converting Gary code to pure Binary code:</u>

Let G be a number written in Gray code on N bits

$G_{(Gray)} = G_N \ldots\ldots\ldots G_4G_3G_2G_1$ G_N is the most significant bit and G_1 is the least significant bit

B is the equivalent in pure or natural binary code of the number G also written on N bits

$B_{(2)} = B_N \ldots\ldots\ldots B_4B_3B_2B_1$

The transition from the Gray Code to the pure binary code is done according to the following two steps:

1. $B_N = G_N$.
2. $G_{N-1} = G_N + B_{N-1}$

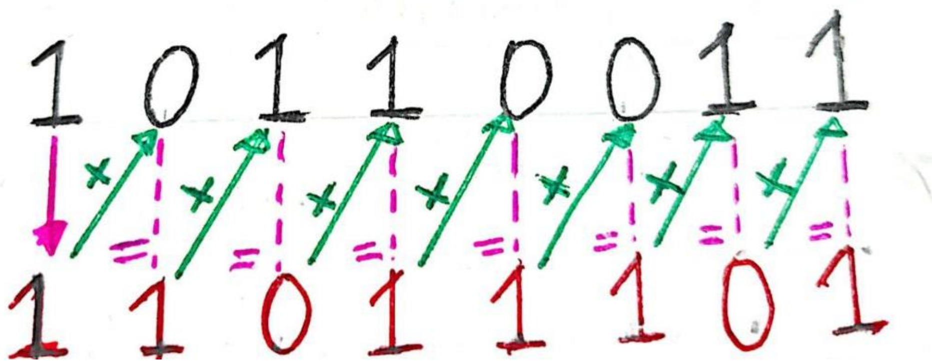

Figure 2: Converting a Gray number into binary

Note

- The most significant bit of the binary number is always equal to the most significant bit of its Gray code equivalent.
- For additions of bits, the case of (1 +1 = 0) because the carry will be ignored (See Figure 2 for the cases of the second-order and seventh-order bits.

I.2 – <u>The DCB Code: Decimal Binary Coded:</u>

1- <u>DCB Code Definition</u>:

The DCB or BCD code, is the acronym for Binary Coded Decimal in English is a numbering system used in digital electronics and computer science to encode numbers by approximating the usual human representation, in base 10. In this format, numbers are represented by one or more digits between **0 and 9**, and each of these digits is encoded on four bits.

2- <u>DCB Code Principle</u>

To encode a decimal number in BCD, we will separately encode each digit of the base ten number in Binary according to the table below, remember that we work on **4 bits (4 positions),** so the numbers from 10 to 15 (1010 to 1111) are not supported by the DCB code.

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

Example

We will encode the number $(785)_{10}$ in DCB, for this, each digit of the number will be encoded separately in binary on 4 bits.

Decimal	7	8	5
BCD	0111	1000	0101

So $(785)_{10}$ = (0111 1000 0101) $_{DCB}$

3- <u>Addition in DCB code</u>:

As already mentioned, the DCB code allows the coding of only 10 numbers, namely, the digits from 0 to 9 , therefore, the numbers from 10 to 15, are not allowed, that is, the DCB code does not know them.

When we add 2 DCB-coded numbers, we add each 4-bit block of the first number, with its equivalent of the second number.

Now, if the result of one of the blocks exceeds 9, that is to say its value is between 10 and 15 (1010 to 1111), the number 6 (0110) is added to this block.

Examples

- Either to add $(352)_{10}$ + $(34)_{10}$ in DCB

	Decimal		BCD
	352		0011 0101 0010
+		+	
	34		0011 0100
=		=	
	386		0011 1000 0110

There, the result is equivalent, and correct.

- Either to add $(153)_{10}$ +$(151)_{10}$ in DCB

	Decimal		BCD		
	153		0001	0101	0011
+		+			
	151		0001	0101	0001
=		=			
	304		0010	1010	0010

There, the result obtained in DCB is not equivalent to that of the Decimal, moreover, we obtained the value 1010 (10 in Decimal which is >9) which is not recognized by the DCB code, there, we must add to this block which exceeds 9, the value 6 (0110).

Let's see, what happens when we add 6 (0110) to this block.

	Decimal		BCD		
	153		0001	0101	0011
+		+			
	151		0001	0101	0001
=		=			
	304		0010	1010	0100
		+	0110		
		=	0011	0000	1000

There, the result is correct!.

One question remains, why add 6 (0110) specifically?.

Well, because as in DCB, we code on 4 bits, so 2^4=16 numbers are possible in binary coderen, but only 10 numbers are allowed (from 0 to 9), the remaining 6 numbers (from 10 to 15) are prohibited, when the sum exceeds 9, we add 6 to make a loop and return to the value 0.

Notes

- The number encoded in BCD does not correspond to the decimal number converted to natural binary.
- Combinations greater than 9 (from 10 to 15) are prohibited. For example, the combination 1010 does not belong to the BCD code.
- BCD decimal coding is simple, but it is not possible to do mathematical operations directly on it.
- This code is mostly used for the display of decimal data. (In calculators for example)

I.3 – The DCB Code plus three: DCB +3

1- DCB+3 Code Definition

The **DCB code plus 3** or **excess 3 code** also called **Stibiz code** of the name of its inventor, is an unweighted code from the **DCB** code to which **3** is systematically added to each digit. This code is often used on arithmetic units that calculate in a decimal number system rather than a binary system.

The code plus 3 makes it possible to perform the arithmetic operations of addition and subtraction with a minimum of logical functions.

2- DCB+3 Code Principle

To code a decimal number in BCD+3, before coding it in DCB, we add 3 to it, then we code it in DCB, we remember that we work on **4 bits (4 positions),** so we have 16 numbers (from 0 to 15) that we can code in DCB +3, but combinations of 13 to 15 (1101 to 1111) are prohibited, we can just code from 0 to 12.

The table below illustrates the coding of decimal numbers in DCB+3

Decimal	Decimal +3	DCB+3
0	3	0011
1	4	0100
2	5	0101
3	6	0100
4	7	0111
5	8	1000
6	9	1001
7	10	1010
8	11	1011
9	12	1100

Coding of Decimal Numbers by DCB+3

Example

We will encode the number $(785)_{10}$ in DCB+3, for this, we add to each digit 3, then it will be encoded separately in binary on 4 bits, then we

Decimal	7	8	5
Decimal +3	10	11	8
DCB+3	1010	1011	1000

So $(785)_{10}$ = (1010 1011 1000) $_{DCB+3}$.

Exercise 1

1.Count from 0 to 11 in Gray's code

2.Find the Gray code for the following numbers: 1111101 , 1011110 , 1100100

3.Find the numbers with the following Gray codes: 1011001, 1101100 , 1000011

Exercise 2

1. Convert the following decimal numbers into their DCB and DCB +3 equivalents

74 165 9201

2. Perform the following additions in DCB:

38 + 72 51 + 19

SOLUTION EXERCICE 1

Décimal	Code de Gray
0	0000
1	0001
2	0011
3	0010
4	0110
5	0111
6	0101
7	0100
8	1100
9	1101
10	1111
11	1110

2.

Binaires	1111101	1011110	1100100
Codes de Gray	**1000011**	**1110001**	**1010110**

3.

Codes de Gray	1011001	1101100	1000011
Binaires	**1101110**	**1001000**	**1111101**

SOLUTION EXERCICE 2

1.

Décimal	**74**	**165**	**9201**
Code DCB	0111 0100	0001 0110 0101	1001 0010 0000 0001
Code DCB_{+3}	1010 0111	0100 1010 1000	1100 0101 0011 0100

2.

Décimal			**DCB**		
	3 8			0011	1000
+		+			
	7 2			0111	0010
=		=			
	110			1010	1010
		+		0110	0110
		=	0001	0001	0000

Décimal			**DCB**	
	5 1		0101	0001
+		+		
	1 9		0001	1001
=		=		
	7 0		0110	1010
		+		0110
		=	0111	0000