Digital Electronic Circuits
Section 1 (EE, IE)

Lecture 19

Class Test 2:
29-10-2020 (THU): 8:00 – 8:55 AM
Syllabus: Logic families (not covered in CT1) and primarily post CT1, shall include concept dealt in pre-CT1 part which forms the pre-requisite.

Issue with Binary Code

B_3	B_2	B_1	B_0
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0

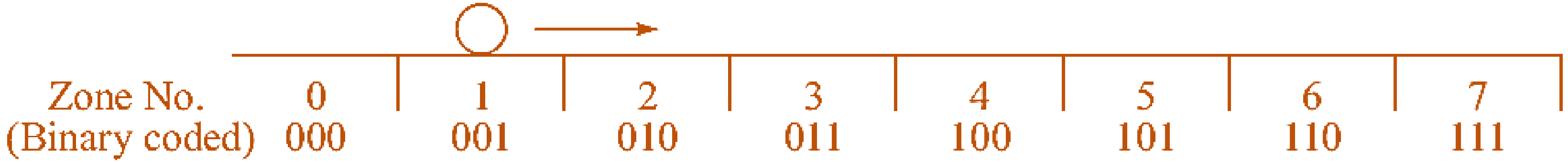
2 digits change (0001 → 0010)

3 digits change (0011 → 0100)

.

.

.



$B_2B_1B_0$:
(If B_1 change is slower than B_0)

001

000 → 010

Possibility of decoding a wrong direction of movement

$$d_n \ldots d_1 d_0 . d_{-1} d_{-2} \ldots d_{-m} = d_n \times r^n + \ldots + d_1 \times r^1 + d_0 \times r^0 + d_{-1} \times r^{-1} + d_{-2} \times r^{-2} + \ldots + d_{-m} \times r^{-m}$$

1010 in 8421 code = 1 x 8 + 0 x 4 + 1 x 2 + 0 x 1

8421 : Weights associated with position

Note: 1010 in 2421 code = 1 x 2 + 0 x 4 + 1 x 2 + 0 x 1

Gray Code

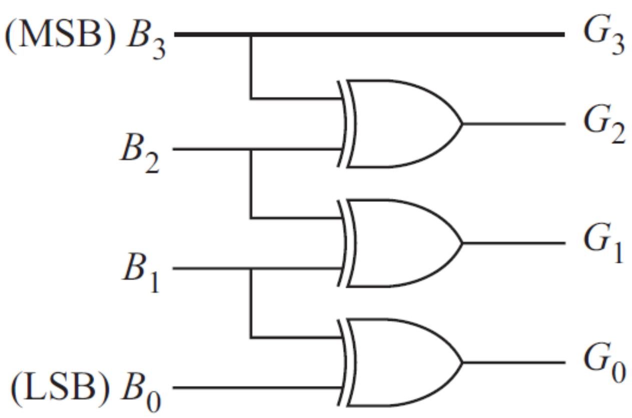
B_3	B_2	B_1	B_0	G_3	G_2	G_1	G_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1

1-bit	2-bit	3-bit	4-bit
0	0 0	0 00	0 000
1	0 1	0 01	0 001
	1 1	0 11	0 011
	1 0	0 10	0 010
		1 10	0 110
		1 11	0 111
		1 01	0 101
		1 00	0 100
			1 100
			1 101
			1 111
			1 110
			....

- Gray Code is an unweighted code with unit distance i.e. only one position changes between two successive positions.
- It is also called reflected code as by reflecting (n – 1)-bit gray code, n-bit gray code can be obtained.

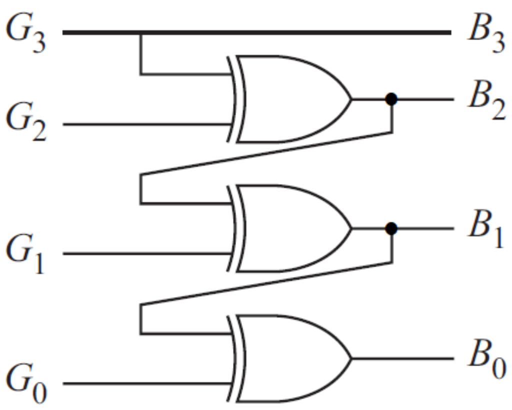
Code Conversion

B_3	B_2	B_1	B_0	G_3	G_2	G_1	G_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1



$G_n = B_n$

$G_i = B_{i+1} \oplus B_i$ for $i < n$



$B_n = G_n$

$B_{n-1} = G_n \oplus G_{n-1}$

$B_i = B_{i+1} \oplus G_i$ for $i < (n-1)$

Excess-3 Code

Binary Code Decimal + 0011 = Excess-3 Code (XS-3)

	BCD	XS-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**

53

1000 0110

$(6.9)_{10}$
$= (0110.1001)_{BCD}$
$= (1001.1100)_{XS\text{-}3}$

Decimal	XS-3
10	0100 0011
11	0100 0100
99	1100 1100
100	0100 0011 0011
101	0100 0011 0100
487	0111 1011 1010
…	…

Code Conversion:

Approach 1:

- BCD to XS-3: Represent each XS-3 bit, $X_i = F_1(B_3,B_2,B_1,B_0)$
- XS-3 to BCD: $B_i = F_2(X_3,X_2,X_1,X_0)$

(Derive *F* by considering 'don't care' for unused input combinations.)

Approach 2:

- BCD to XS-3: Add 0011
- XS-3 to BCD: Subtract 0011 (Consider 4-bit Adder-Subtractor)

Addition of XS-3 Code

- Subtract 0011 from addition result if no carry is produced.

```
   (5)10          1000  (XS-3)        1101  (XS-6)
 + (2)10        + 0101  (XS-3)      - 0011
 ---------       ------              ------
   (7)10          1101  (XS-6)        1010  (XS-3)
```

No carry up to

$(1111)_{XS-6}$

$= (1001)_2$

$= (9)_{10}$

$= (1100)_{XS-3}$

- Add 0011 to addition result if carry is produced.

```
   (5)10          1000                  0010
 + (7)10        + 1010                + 0011
 ---------       ------                ------
   (12)10        1 0010     (0)1(00)   0101  (XS-3)
                               1         2   (Decimal)
```

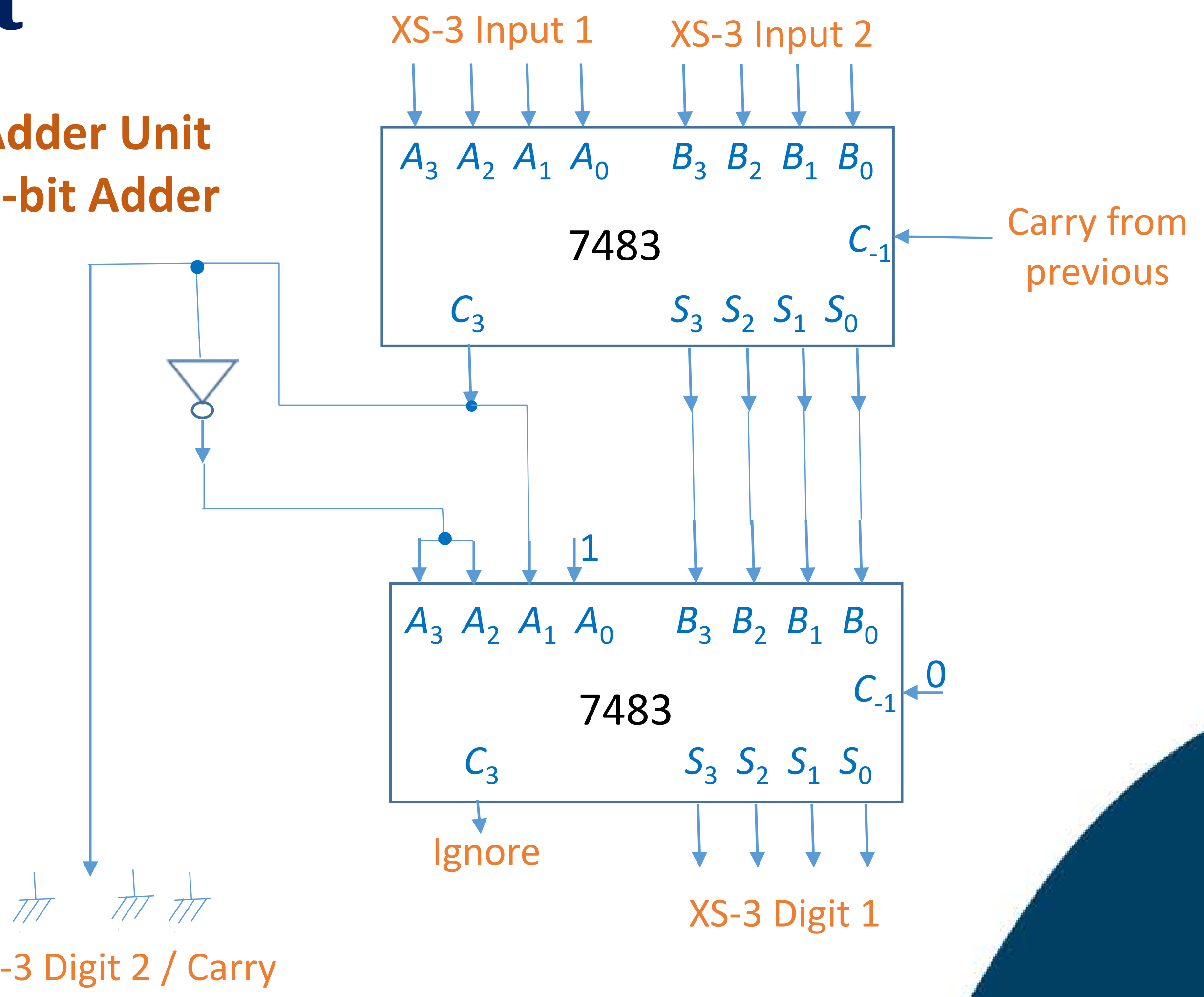
```
        1
   0101   1000          (25)10
 + 1000   1010        + (57)10
 ---------------      ---------
   1110   0010          (82)10
 - 0011 +0011
 ---------------
   1011   0101  (XS-3)
    8       2   (Dec.)
```

XS-3 Adder Unit

1
0101 1000 $(25)_{10}$
+ 1000 1010 + $(57)_{10}$
---------------- ---------
1110 0010 $(82)_{10}$
- 0011 +0011

1011 0101 (XS-3)
8 2 (Dec.)

XS-3 Adder Unit from 4-bit Adder

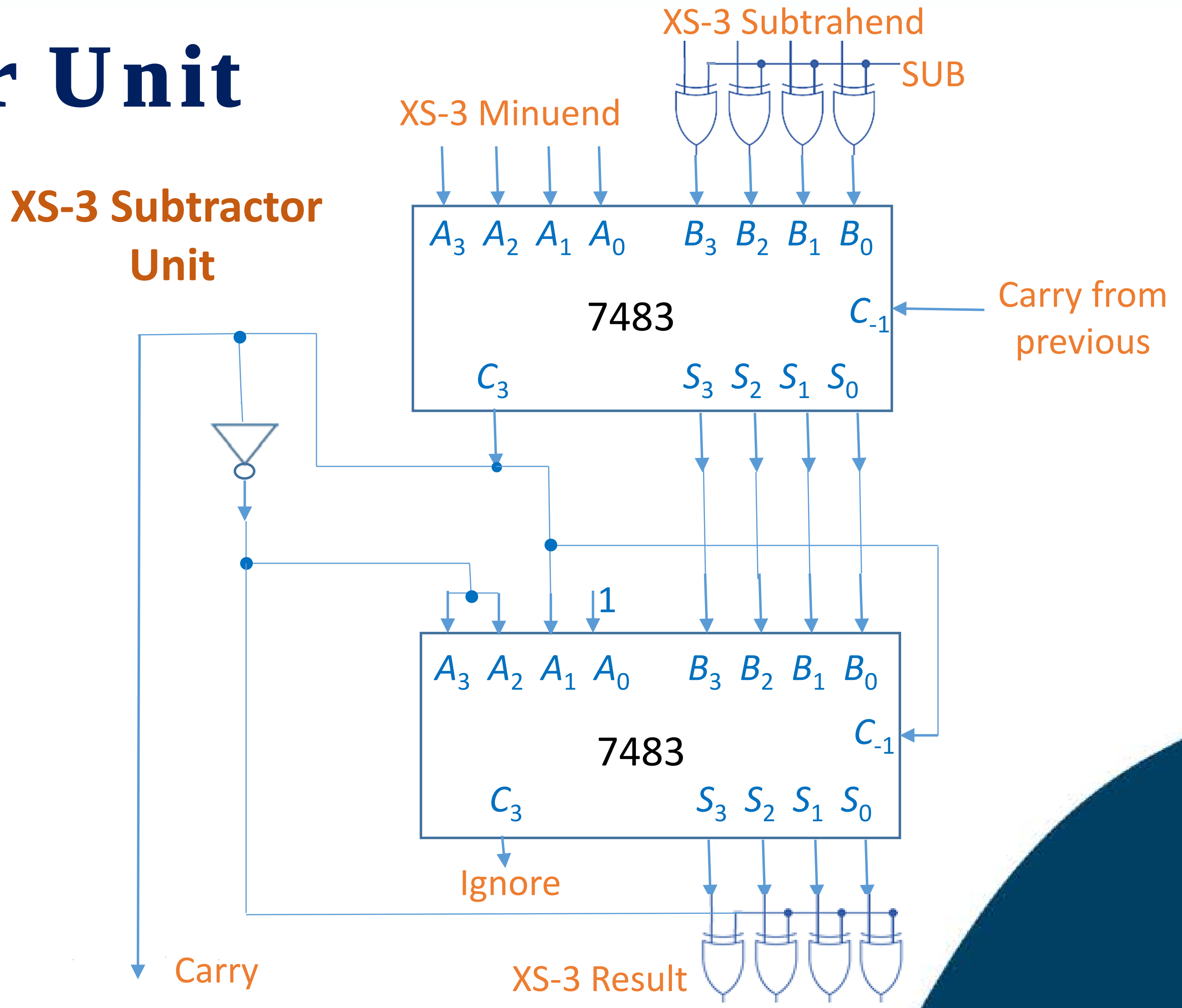


XS-3 Subtractor Unit

Subtraction using 9's C:
Inverting bits of XS-3 code 9's C in XS-3 directly obtained.

$(2)_{10}$: $(0101)_{XS\text{-}3}$
9's C of 2 = 9 – 2 = 7
$(7)_{10}$: $(1010)_{XS\text{-}3}$

- Add 9's C of subtrahend with minuend
- If carry, result +ve, add carry, also add 0011 for XS-3
- If no carry, result –ve, subtract 3 for XS-3, invert its bits



ASCII Code

ASCII: American Standard Code for Information Interchange; standardized for computer hardware

	$X_6X_5X_4$					
$X_3X_2X_1X_0$	010	011	100	101	110	111
0000	SP	0	@	P		p
0001	!	1	A	Q	a	q
0010	“	2	B	R	b	r
0011	#	3	C	S	c	s
0100	$	4	D	T	d	t
0101	%	5	E	U	e	u
0110	&	6	F	V	f	v
0111	’	7	G	W	g	w
1000	(	8	H	X	h	x
1001	)	9	I	Y	i	y
1010	*	:	J	Z	j	z
1011	+	;	K		k	
1100	,	<	L		l	
1101	–	=	M		m	
1110	•	>	N		n	
1111	/	?	O		o	

7 bits: 128 Codes
Includes control codes for peripherals and printable characters

A: 1000001 a: 1100001
B: 1000010 b: 1100010
C: 1000011 c: 1100011

0: 0110000 =: 0111101
1: 0110001 : : 0111110
2: 0110010 . : 0101110

0000000: Null Character
0000001: Start of Heading
0001101: Carriage Return

EBCDIC (Extended Binary Coded Decimal Interchange Code) was introduced for IBM devices.

References

References:

- Donald P. Leach, Albert P. Malvino, and Goutam Saha, Digital Principles & Applications 8e, McGraw Hill
- https://www.ascii-code.com/ accessed on 15-12-2018

FREE ONLINE EDUCATION swayam