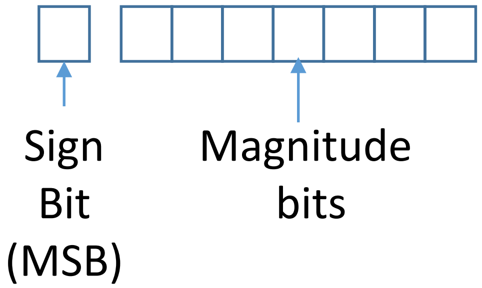
Digital Electronic Circuits
Section 1 (EE, IE)

Lecture 15

Sign-Magnitude Number



0000 0001 : +1
1000 0001 : -1

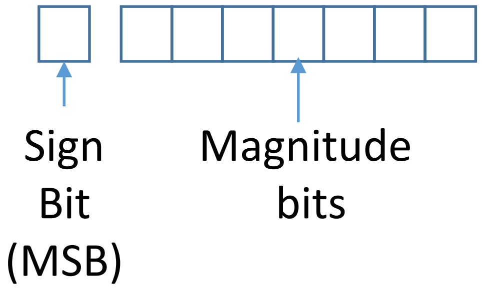
0001 0110 : +22
1001 0110 : -22

0111 1111 : +127
1111 1111 : -127

Range: -127 to +127 i.e. with 8 bits 255 numbers can be represented.
(-127 to -1, 0, +1 to +127)

0 is represented twice: 0000 0000 and 1000 0000

1’s Complement Number



0000 0001 : +1
1111 1110 : -1

0001 0110 : +22
1110 1001 : -22

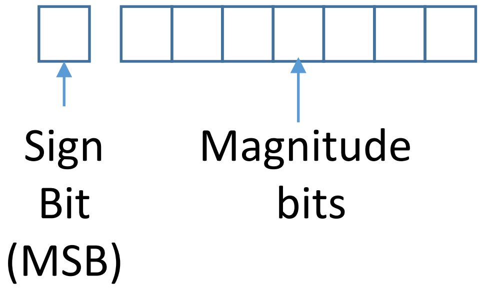
0111 1111 : +127
1000 0000 : -127

Negative number representation is obtained by taking **1’s complement or inversion** of positive counterpart.

Range: -127 to +127 i.e. with 8 bits 255 numbers can be represented.
(-127 to -1, 0, +1 to +127)

0 is represented twice: 0000 0000 and 1111 1111

2's Complement Number



Negative number representation is obtained by adding 1 to **1's complement** of positive counterpart.

Range: -128 to +127 i.e. with 8 bits 256 numbers can be represented. (-128 to -1, 0, +1 to +127)
0 is represented once: 0000 0000

0000 0001 : +1 $\xrightarrow{\text{1's C}}$ 1111 1110
\+ 1

1111 1111 (2's C)

1111 1111 : -1

0001 0110 : +22 $\xrightarrow{\text{1's C}}$ 1110 1001
\+ 1

1110 1010 (2's C)

1110 1010 : -22

0111 1110 : +126
1000 0010 : -126

0111 1111 : +127
1000 0001 : -127

1000 0000 : -128

2’s Complement of 2’s Complement

0000 0001 : +1
1111 1111 : -1

1’s Complement

0000 0000
\+ 1

0000 0001

0001 0110 : +22
1110 1010 : -22

1’s Complement

0001 0101
\+ 1

0001 0110

$$X \xrightarrow{\text{2's C}} -X \xrightarrow{\text{2's C}} -(-X) = X$$

Binary Addition and Subtraction

0 + 0 = 0, 0 + 1 = 1, 1 + 1 = 10

```
            1
  0 001 0101     (21)10
+ 0 010 0001     (33)10
-----------------  ------
  0 011 0110     (54)10
Sign bit

         Carry
          111
  0 001 0111     (23)10
+ 0 110 0101     (101)10
-----------------  -------
  0 111 1100     (124)10
```

0 – 0 = 0, 1 – 0 = 1, 1 – 1 = 0, 0 – 1 = (1)0 – 1 = (1)1

(1): Borrow

```
           Borrow
          1
  0 001 0101     (21)10
- 0 000 0010     ( 2)10
----------------  ------
  0 001 0011     (19)10

     Borrow
      11 11
  0 110 0101     (101)10
- 0 001 0111     ( 23)10
----------------  ---------
  0 100 1110     (78)10
```

- Done here as done for standard decimal subtraction; borrowed is 2 (binary) instead of 10 (decimal) from next higher position.
- Sign is changed if smaller number is subtracted from larger number as in decimal e.g. 5 – 8 = - (8 – 5) = -3
  i.e. 0 0101 – 0 1000
  = 1 0011

2's Complement Arithmetic

- Addition follows standard process.
- Subtraction by taking 2's complement of subtrahend (i.e. making it negative) and adding it with minuend

$(83)_{10} + (16)_{10}$ ⇒

Carry
1
$(83)_{10}$: 0101 0011
$+ (16)_{10}$: + 0001 0000

$(99)_{10}$: 0110 0011

Minuend
- Subtraction

Difference

$(83)_{10} – (16)_{10} = (83)_{10} + (-16)_{10}$

Subtrahend $(16)_{10} = (0001\ 0000)_2$

0001 0000 $\xrightarrow{\text{1's C}}$ 1110 1111
$(16)_{10}$

0001 0000 $\xrightarrow{\text{2's C}}$ 1111 0000
$(-16)_{10}$

  0101 0011
+ 1111 0000
----
1 0100 0011

Carry

If carry is generated, discard it and answer is positive.

$(0100\ 0011)_2 = (67)_{10}$

2’s Complement Arithmetic

- Subtraction of larger number from smaller number

Minuend
Subtrahend

Difference

$(16)_{10} - (83)_{10} = (16)_{10} + (-83)_{10}$

Subtrahend $(83)_{10} = (0101\ 0011)_2$

If no carry and the sign bit is 1, the answer is negative and in 2’s Complement form.

0101 0011 $\xrightarrow{\text{1's C}}$ 1010 1100

$(83)_{10}$ $\xrightarrow{\text{2's C}}$ 1010 1101 $(-83)_{10}$

```
  0001 0000
+ 1010 1101
-----------------
  1011 1101
```

1011 1101 $\xrightarrow{\text{1's C}}$ 0100 0010

1011 1101 $\xrightarrow{\text{2's C}}$ 0100 0011

Note:

1010 1101 $\xrightarrow{\text{1's C}}$ 0101 0010

1010 1101 $\xrightarrow{\text{2's C}}$ 0101 0011

2’s C
$(0100\ 0011)_2 = (67)_{10}$
$(1011\ 1101)_2 = (-\ 67)_{10}$

2's Complement Arithmetic

- Subtraction from a negative number

Minuend
Subtrahend

Difference

$(-83)_{10} - (16)_{10} = (-83)_{10} + (-16)_{10}$

If carry then discard it, if the sign bit is 1 then the answer is negative and in 2's Complement form.

$(-83)_{10}$ = 1010 1101

$(-16)_{10}$ = 1111 0000

```
  1010 1101
+ 1111 0000
-----------------
 11001 1101
```

$1001\ 1101 \xrightarrow{1\text{'s C}} 0110\ 0010$

$1001\ 1101 \xrightarrow{2\text{'s C}} 0110\ 0011$

2's C $\left\{ \begin{array}{l} (0110\ 0011)_2 = (99)_{10} \\ (1110\ 1101)_2 = (-99)_{10} \end{array} \right.$

2's Complement Arithmetic

- Subtraction of a negative number

$(83)_{10} - (-16)_{10} = (83)_{10} + (16)_{10}$

$(83)_{10} = 0101\ 0011$

$(-16)_{10} = 1111\ 0000$

$\downarrow$ 2's C

$(16)_{10} = 0001\ 0000$

```
  0101 0011
+ 0001 0000
-----------------
  0110 0011
```

$\downarrow$

$(99)_{10}$

For subtraction,

(i) Take 2's C of the subtrahend

(ii) Add it with Minuend

(iii) Discard carry, if any

(iv) If sign bit is 0, answer positive and can be directly read.

(v) If sign bit is 1, answer is negative and in 2's C form

(Take 2's C to find the magnitude in binary coding i.e. number with position weights ..8421).

Minuend
Subtrahend

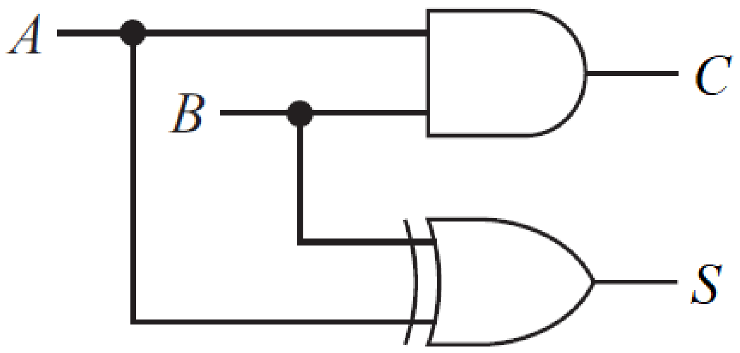
Difference

Half Adder

Input		Output	
A	***B***	***C***	***S***
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

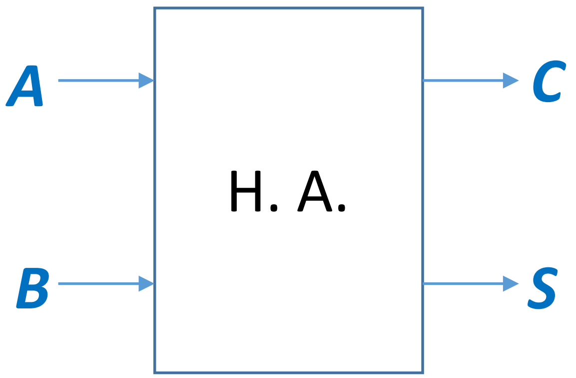
Augend → *A*
Addend → *B*

C ← Carry Sum → *S*



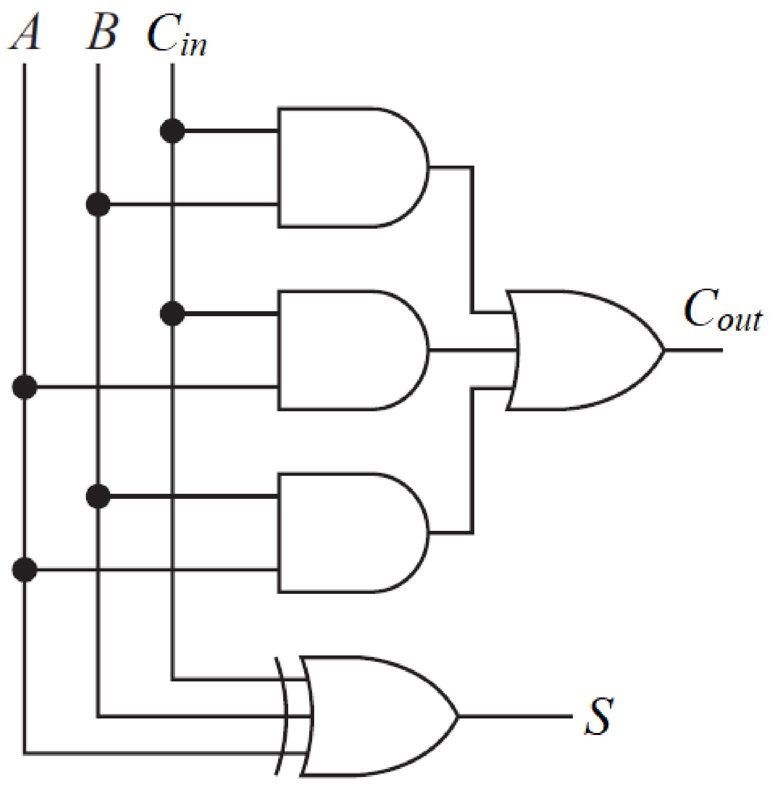
$C = A.B$

$S = A'.B + A.B' = A \oplus B$



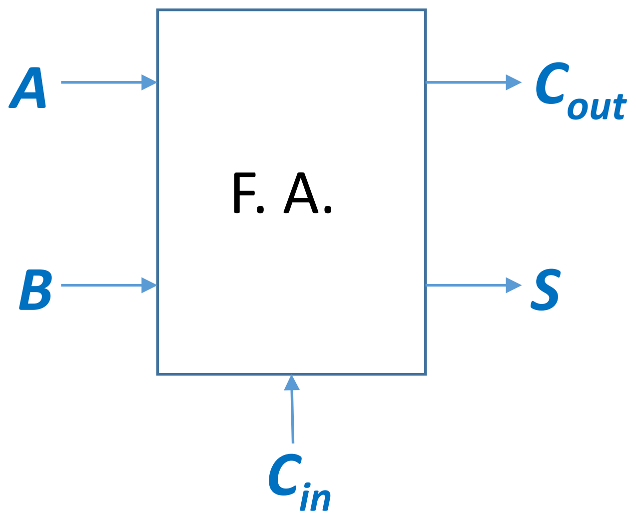
Full Adder

Input			Output	
C_{in}	A	B	C_{out}	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

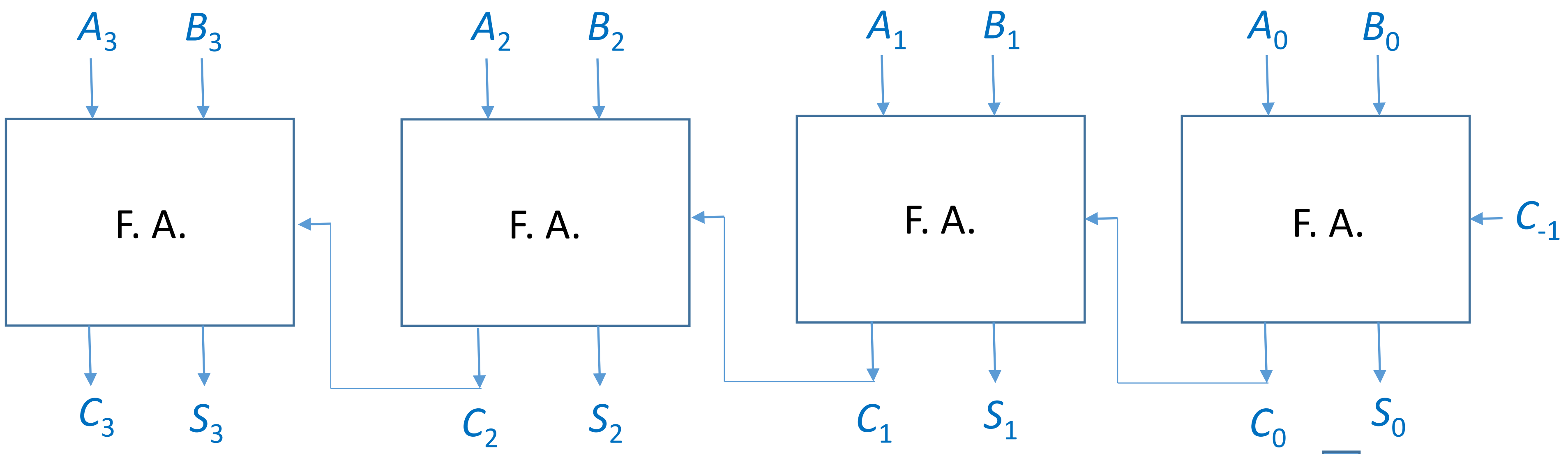


$C_{out} = A.B + A.C_{in} + B.C_{in}$

$S = A \oplus B \oplus C_{in}$



Ripple Carry Adder



Intermediate Carry

$C_2\ C_1\ C_0$

$A_3\ A_2\ A_1\ A_0$

$B_3\ B_2\ B_1\ B_0$

$C_3\ \ S_3\ S_2\ S_1\ S_0$

Final Carry and Sum

$(15)_{10} + (1)_{10}$
$= (1111)_2 + (0001)_2$

Carry 111

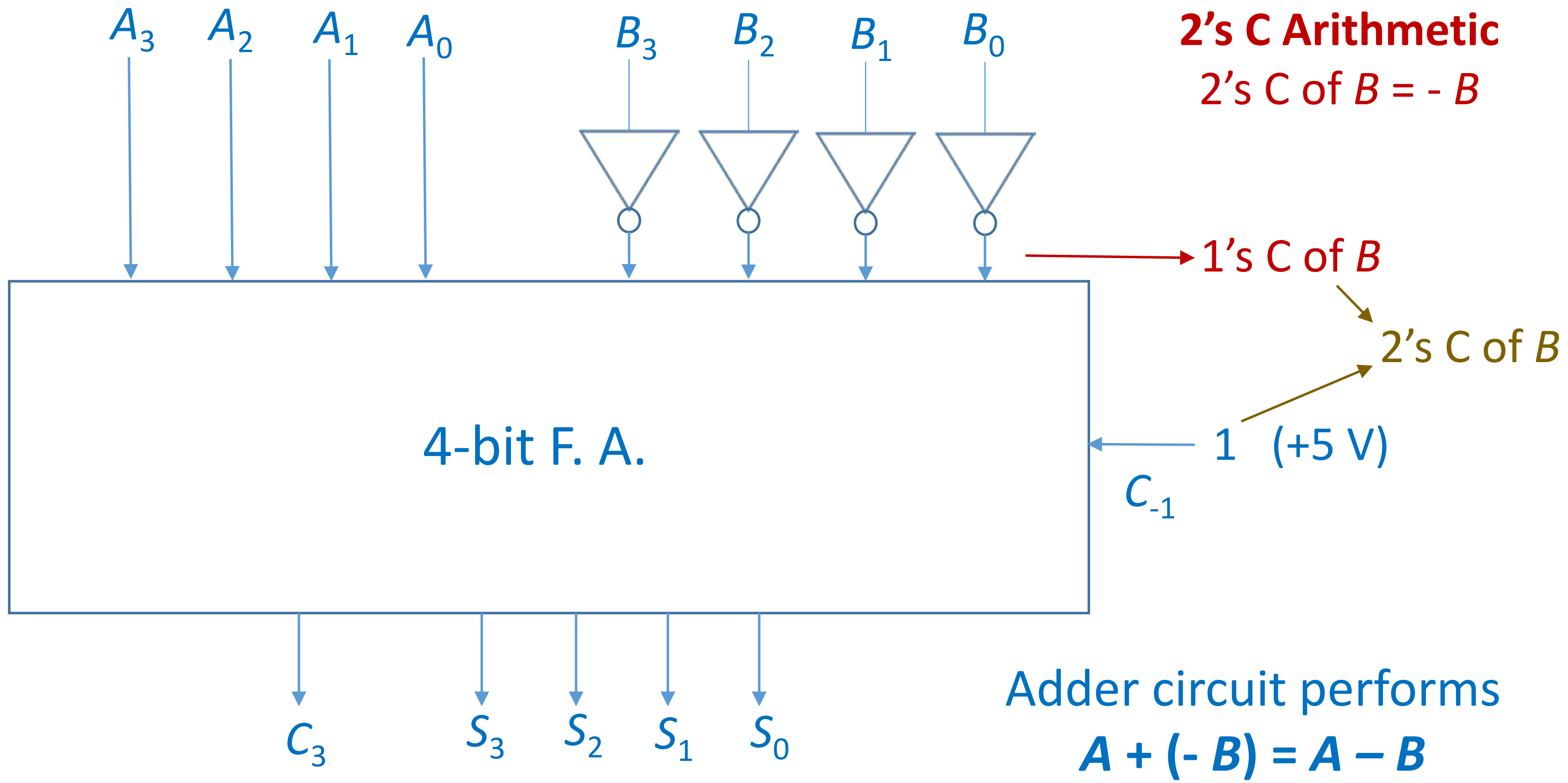
```
   1111
 + 0001
---------
 1 0000
```

4-bit Ripple Carry Addition

$C_{-1} = 0$ (GND)

First stage can be H.A.

Subtraction using Adder Circuit



2's C Arithmetic

2's C of B = - B

Adder circuit performs

A* + (- *B*) = *A* – *B

$(15)_{10} - (1)_{10}$

$= (1111)_2 - (0001)_2$

2's C

```
    1111
    1110
+      1
---------
  1 1110
```

Discard

$(1110)_2 = (14)_{10}$

References:

- Donald P. Leach, Albert P. Malvino, and Goutam Saha, Digital Principles & Applications 8e, McGraw Hill