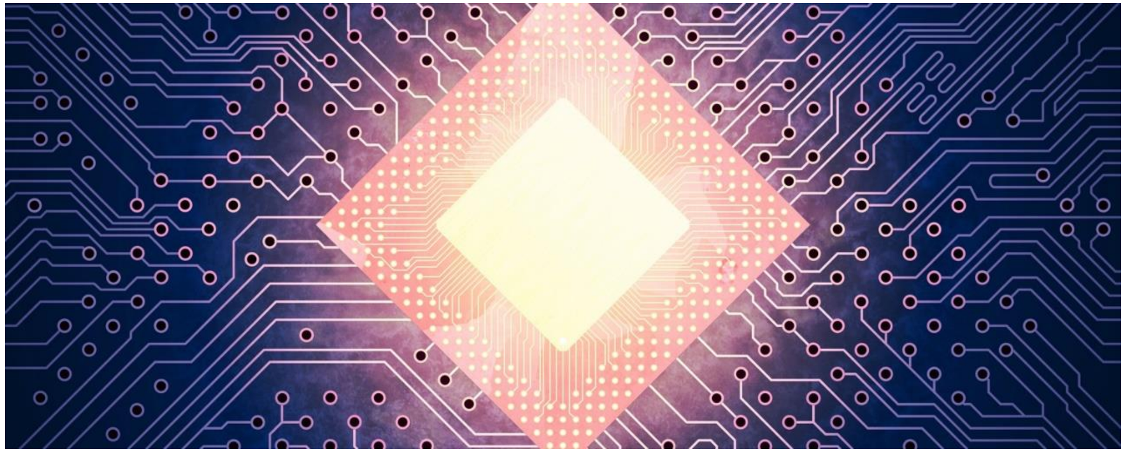
1



ET2223 Microprocessors Microcontrollers & Embedded Systems

Digital Number Systems

Sandali Goonatilleke

Department of Computer Engineering

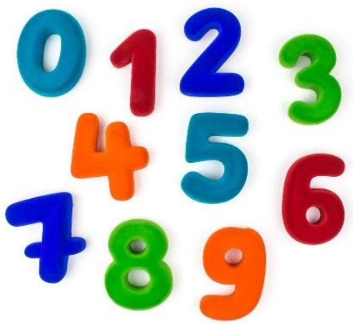
General Sir John Kotelawala Defence University

Numbers – Positional Notation

- A computer can understand the positional number system
- There are only a few symbols called digits and represent different values depending on the position they occupy in the number
- The value of each digit in a number can be determined using
  - The digit
  - The position of the digit in the number
  - The base of the number system

Digital Systems

- Digital systems contains **discrete amounts of data**
- Examples
  - 26 letters in English alphabet
  - 10 decimal digits
- Larger quantities can be built from discrete values:
  - Words made of letters
  - Numbers made of decimal digits (e.g. 239875)



Decimal Number System

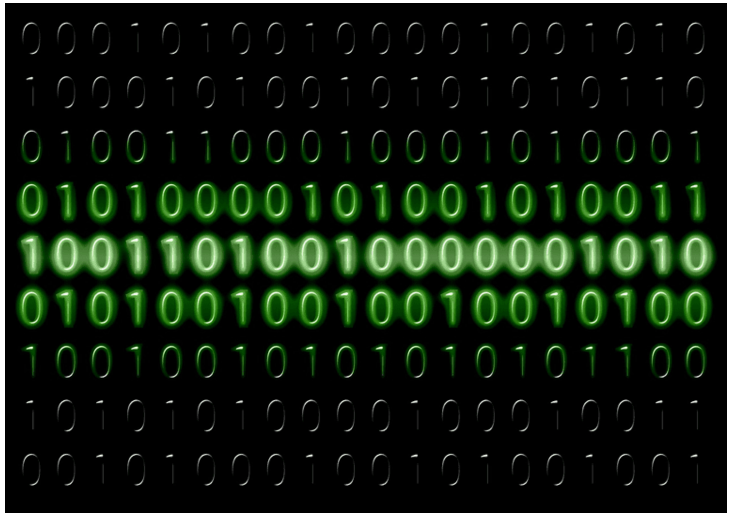
- The number system that we use in our day-to-day life
- Uses 10 digits: **0,1,2,3,4,5,6,7,8,9**
- Has base of 10
- In decimal number system, the successive positions to the left of the decimal point represents units, tens, hundreds, thousands and so on

Example: 98703_{10}

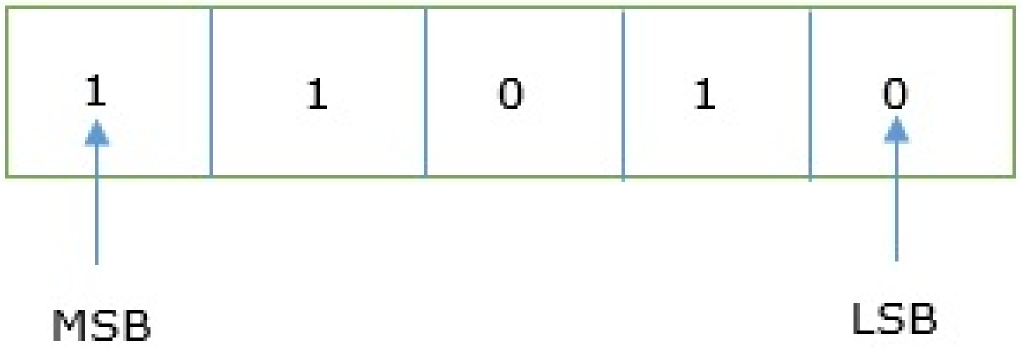
Binary Number System

- Uses **two digits**, **0** and **1**
- Also called as **base 2 number system**

Example: 1101_2



MSB and LSB



- The **leftmost digit** of the number is known as **Most Significant Bit (MSB)**
- The **rightmost digit** of the number is known as **Least Significant Bit (LSB)**

Octal Number System

- Uses 8 digits: 0,1,2,3,4,5,6,7
- Also called as 8 base number system

Example: 12570_8

Hexadecimal Number System

Called base 16 number system

Digits: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

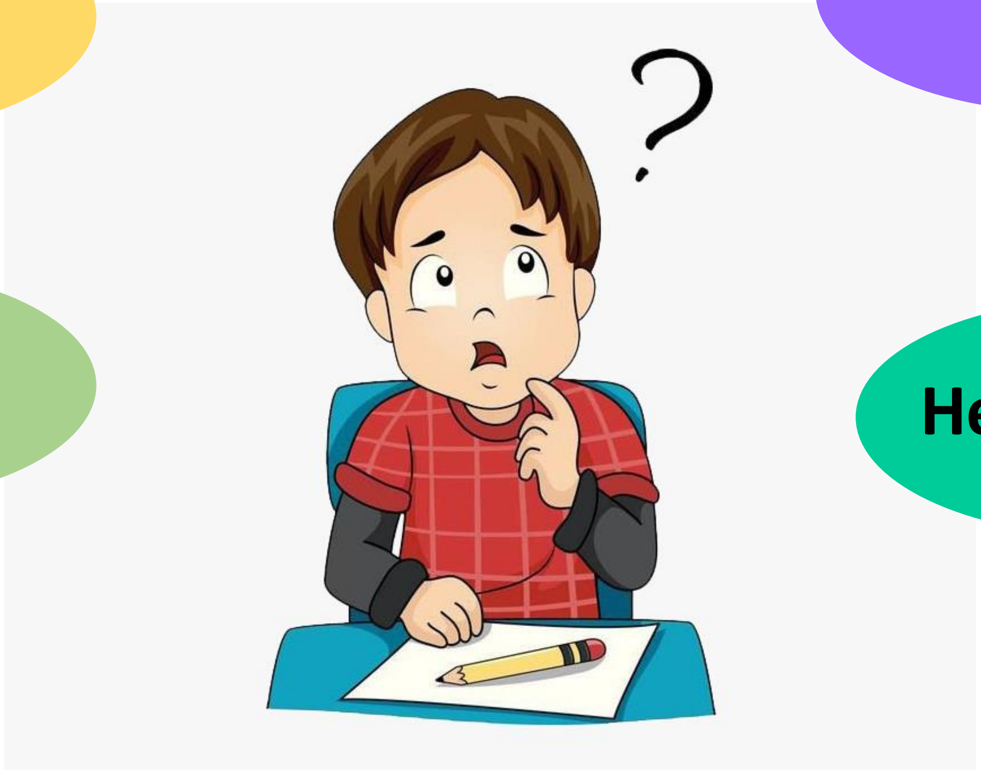
A → 10
B → 11
C → 12
D → 13
E → 14
F → 15

Example: $AB987_{16}$

Which Base Should We Use?

Decimal

Octal



Binary

Hexadecimal

Converting Decimal Numbers into Binary Numbers

1. Divide decimal number by 2
2. Write the integer answer (quotient) under the long division symbol
3. Write the remainder (0 or 1) at the right side of the dividend
4. Continue the process in downwards direction
5. Stop the process when the integer answer (quotient) is 0
6. Starting from bottom, write the sequence of 1's and 0's upwards to the top

Converting Decimal Numbers into Binary Numbers (Cont'd)

Example: Convert decimal number 24_{10} to the binary number

2	24	
2	12	0
2	6	0
2	3	0
2	1	1
	0	1

Quotient

Remainder

LSB

MSB

Answer: $= 11000_2$

Converting Decimal Numbers into Binary Numbers (Cont'd)

Exercise 1: Convert below decimal numbers into binary numbers

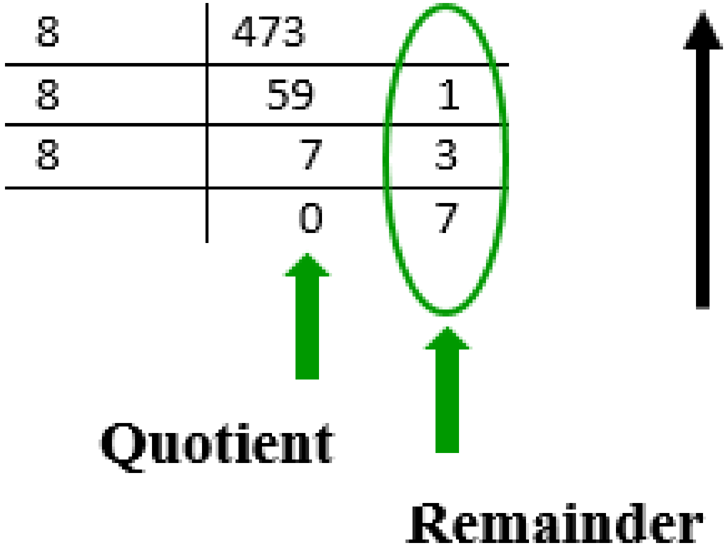
1. 56
2. 385
3. 512

Converting Decimal Numbers into Octal Numbers

1. Divide decimal number by 8
2. Write the integer answer (quotient) under the long division symbol
3. Write the remainder at the right side of the dividend
4. Continue the process in downwards direction
5. Stop the process when the integer answer (quotient) is 0
6. Starting from bottom, write the sequence of reminders upwards to the top

Converting Decimal Numbers into Octal Numbers (Cont'd)

Example: Convert decimal number 473_{10} to the octal number



Answer: 731_8

Converting Decimal Numbers into Octal Numbers (Cont'd)

Exercise 2: Convert below decimal numbers into octal numbers

1. 120

2. 1080

3. 1750

Converting Decimal Numbers into Hexadecimal Numbers

1. Divide decimal number by 16
2. Write the integer answer (quotient) under the long division symbol
3. Write the remainder at the right side of the dividend
4. Continue the process in downwards direction
5. Stop the process when the integer answer (quotient) is 0
6. Starting from bottom, write the sequence of reminders upwards to the top

Converting Decimal Numbers into Hexadecimal Numbers (Cont'd)

Example: Convert decimal number 423_{10} to the hexadecimal number

16	423	
16	26	7
16	1	A
	0	1

Quotient

Remainder

Answer: $1A7_{16}$

Converting Decimal Numbers into Hexadecimal Numbers (Cont’d)

Exercise 3: Convert below decimal numbers into hexadecimal numbers

1. 78

2. 867

3. 1754

Converting Binary Numbers into Decimal Numbers

Example: Convert binary number 110101_2 into the decimal number

Step 1: Write the number as follows

1	1	0	1	0	1

Step 2: Write the weights of positions under each digit of the number

1	1	0	1	0	1
↓	↓	↓	↓	↓	↓
2^5	2^4	2^3	2^2	2^1	2^0

Converting Binary Numbers into Decimal Numbers (Cont'd)

Step 3: Multiply weights of positions with each digit and sum together

$$110101_2 = (1 * 2^5) + (1 * 2^4) + (0 * 2^3) + (1 * 2^2) + (0 * 2^1) + (1 * 2^0)$$

$$110101_2 = (1 * 32) + (1 * 16) + (0 * 8) + (1 * 4) + (0 * 2) + (1 * 1)$$

$$110101_2 = 32 + 16 + 0 + 4 + 0 + 1$$

$$\mathbf{110101_2 = 53_{10}}$$

Converting Octal Numbers into Decimal Numbers

Example: Convert octal number 5127_8 into the decimal number

Step 1: Write the number as follows

5 1 2 7

Step 2: Write the weights of positions under each digit of the number

5	1	2	7
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
8^3	8^2	8^1	8^0

Converting Octal Numbers into Decimal Numbers (Cont'd)

Step 3: Multiply weights of positions with each digit and sum together

$$5127_8 = (5 * 8^3) + (1 * 8^2) + (2 * 8^1) + (7 * 8^0)$$

$$5127_8 = (5 * 512) + (1 * 64) + (2 * 8) + (7 * 1)$$

$$5127_8 = 2560 + 64 + 16 + 7$$

$$\mathbf{5127_8 = 2647_{10}}$$

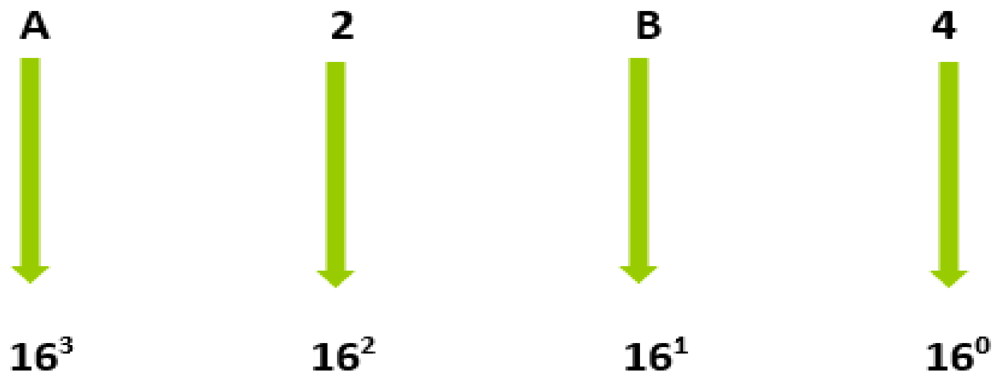
Converting Hexadecimal Numbers into Decimal Numbers (Cont'd)

Example: Convert hexadecimal number $A2B4_{16}$ into the decimal number

Step 1: Write the number as follows

A 2 B 4

Step 2: Write the weights of positions under each digit of the number



Converting Hexadecimal Numbers into Decimal Numbers (Cont'd)

Step 3: Multiply weights of positions with each digit and sum together

$$A2B4_{16} = (A * 16^3) + (2 * 16^2) + (B * 16^1) + (4 * 16^0)$$

$$A2B4_{16} = (A * 4096) + (2 * 256) + (B * 16) + (4 * 1)$$

$$A2B4_{16} = (10 * 4096) + (2 * 256) + (11 * 16) + (4 * 1)$$

$$A2B4_{16} = 40960 + 512 + 176 + 4$$

$$\mathbf{A2B4_{16} = 41652_{10}}$$

Converting Binary, Octal and Hexadecimal into Decimal

Exercise 2: Convert following Binary Numbers into Decimal Numbers

1. 101000111
2. 11100010101
3. 110010101011

Exercise 3: Convert following Octal Numbers into Decimal Numbers

1. 675
2. 1267
3. 76415

Exercise 4: Convert following Hexadecimal Numbers into Decimal Numbers

1. AD34
2. FC879
3. 456EB6

Converting Binary Numbers into Octal Numbers

Example: Let's convert the binary number 110011_2 into the octal number

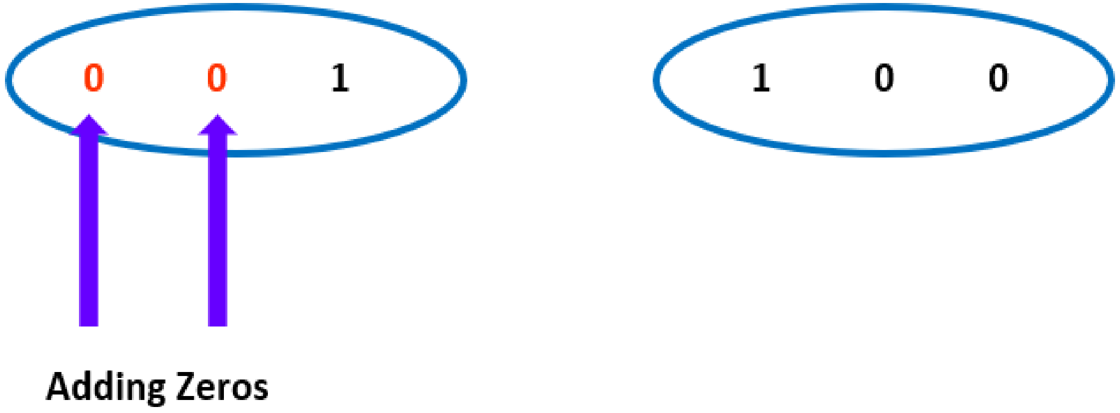
Step 1: Group all the digits such as 1's and 0's in the binary number in sets of three starting from the far right side

1 1 0   0 1 1

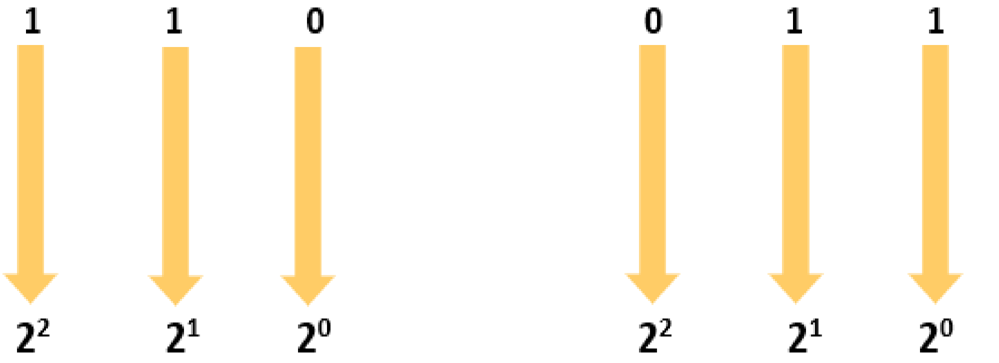
Step 2: Add zeros to the left most group, if there are no enough digits to make a set of three

Example : 1100

Converting Binary Numbers into Octal Numbers (Cont'd)



Step 3: Write the weights of positions under each digit for all the groups separately

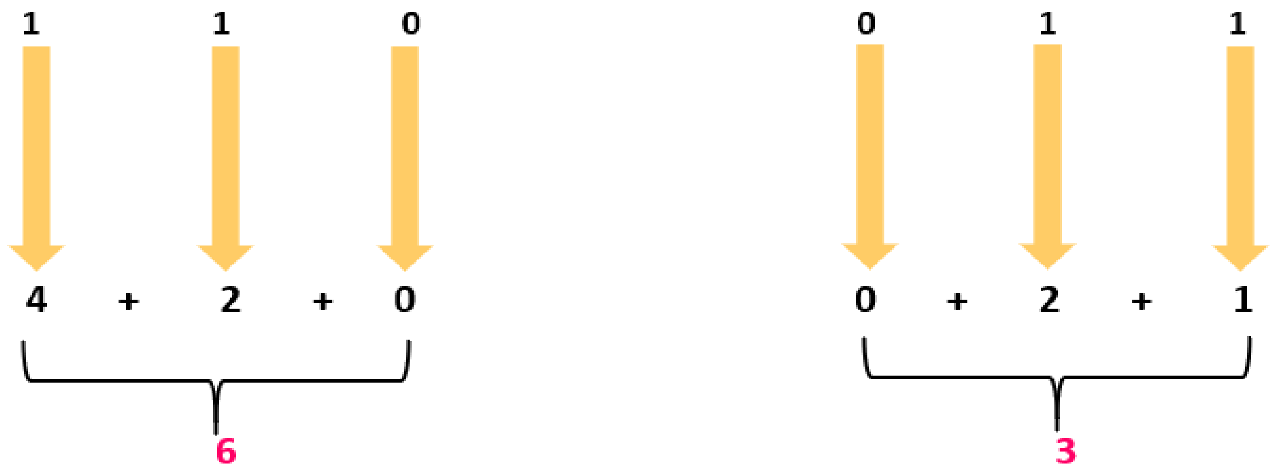


Converting Binary Numbers into Octal Numbers (Cont’d)

Step 4: Multiply positional weights with it appropriate digit

Step 5: Sum the values of the groups separately

Converting Binary Numbers into Octal Numbers (Cont'd)



$$110011_2 = 63_8$$

Converting Binary Numbers into Hexadecimal Numbers (Cont’d)

Example: Convert the binary number 11110101_2 into the hexadecimal number

Step 1: Group all the digits such as 1’s and 0’s in the binary number in sets of four starting from the far right side

1 1 1 1   0 1 0 1

Converting Binary Numbers into Hexadecimal Numbers (Cont'd)

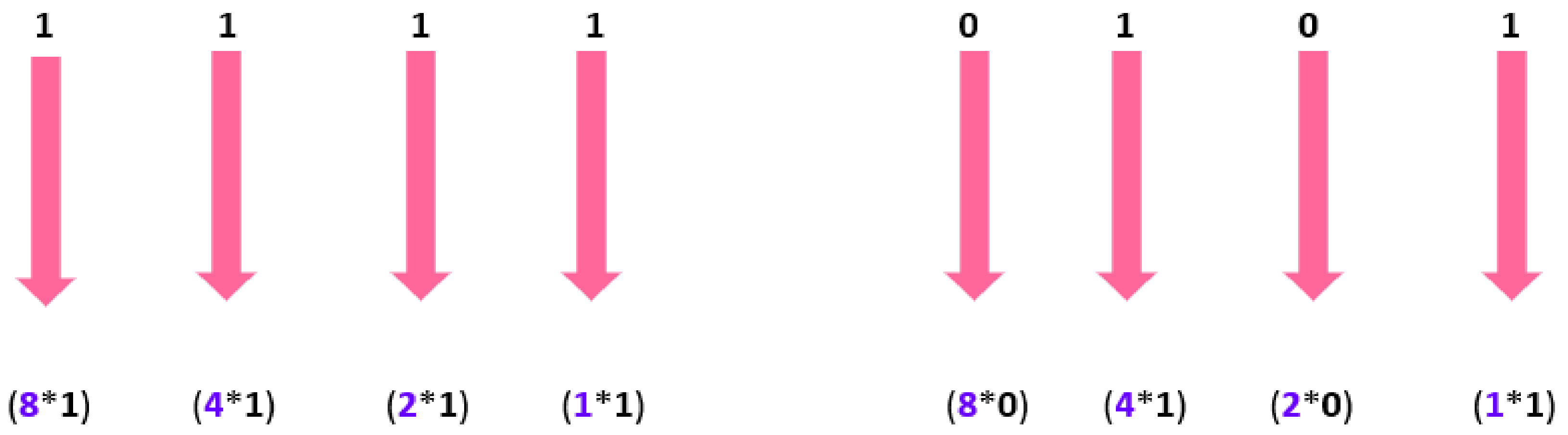
Step 2: Add zeros to the left most group, if there are no enough digits to make a set of four

Step 3: Write the weights of positions under each digit for all the groups separately

1	1	1	1	0	1	0	1
↓	↓	↓	↓	↓	↓	↓	↓
2^3	2^2	2^1	2^0	2^3	2^2	2^1	2^0

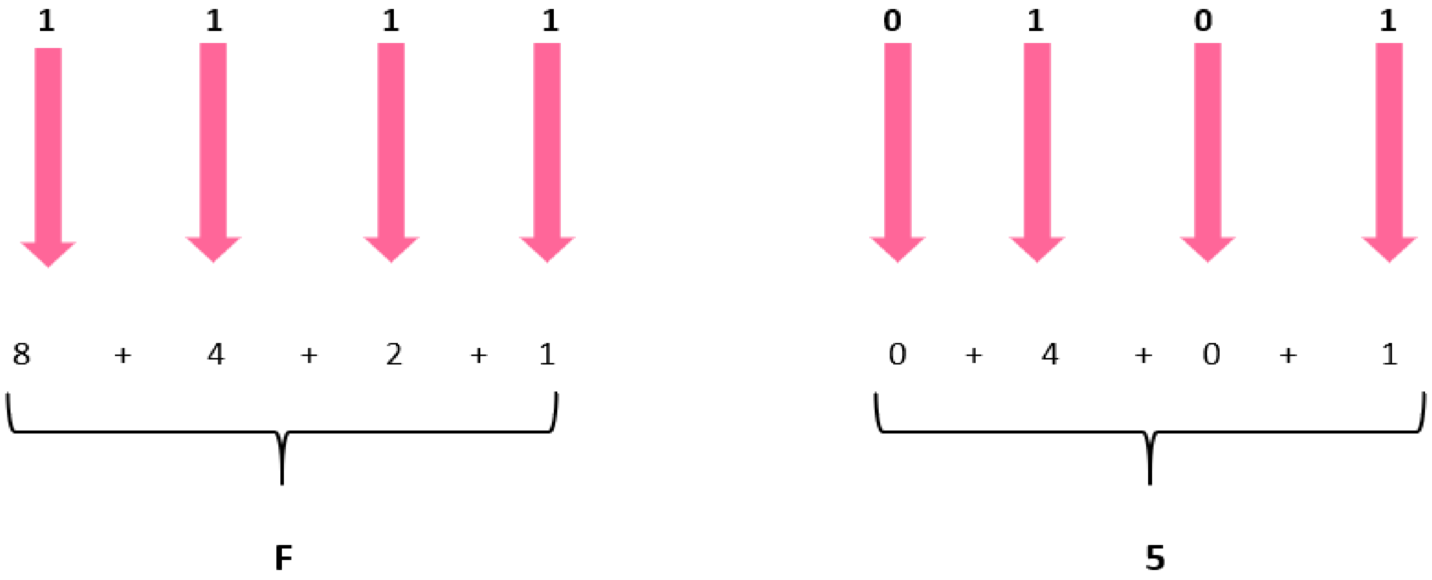
Converting Binary Numbers into Hexadecimal Numbers (Cont'd)

Step 4: Multiply positional weights with it appropriate digit



Converting Binary Numbers into Hexadecimal Numbers (Cont'd)

Step 5: Sum the values of the groups separately



$$11110101_2 = F5_{16}$$

Converting Binary Numbers into Octal and Hexadecimal (Cont'd)

Exercise 5: Convert following Binary Numbers into Octal Numbers

1. 101000
2. 1100101
3. 1110101010

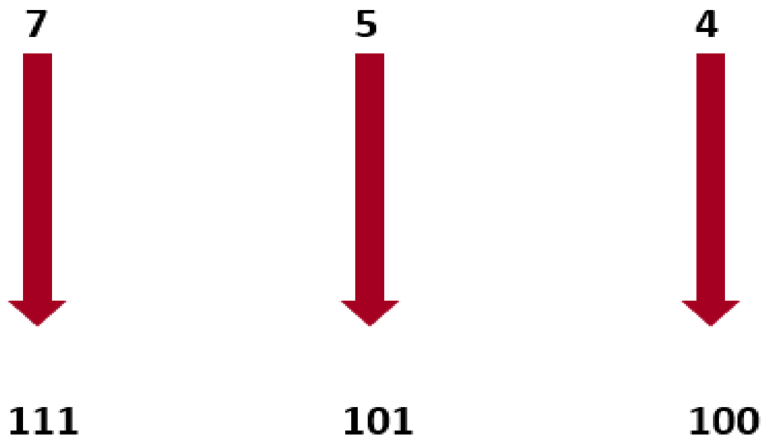
Exercise 6: Convert following Binary Numbers into Hexadecimal Numbers

1. 10110011
2. 111010100
3. 1011100000

Converting Octal Numbers into Binary Numbers

Example: Convert the octal number 754_8 into the binary number

Step 1: Write the digits of the given number separately and represent each digit of the octal number in group of **3 bits** in binary number



$$754_8 = 111101100_2$$

Converting Hexadecimal Numbers into Binary Numbers(Cont'd)

Example: Let's convert the hexadecimal number $BC76_8$ into the binary number

Step 1: Write the digits of the given number separately and represent each digit of the hexadecimal number in group of **4 bits** in binary number

B	C	7	6
1011	1100	0111	0110

$$BC76_{16} = 1011110001110110_2$$

Converting Octal and Hexadecimal Numbers into Binary (Cont'd)

Exercise 7: Convert following Octal Numbers into Binary Numbers

1. 237
2. 1456
3. 35643

Exercise 8: Convert following Hexadecimal Numbers into Binary Numbers

1. ABCDE
2. 9876F2
3. 56AC65

Binary Addition

- Binary addition is very simple

Rule 1: 0 + 0 = 0
Rule 2: 0 + 1 = 1
Rule 3: 1 + 0 = 1
Rule 4: 1 + 1 = 10 (which is 0 carry 1)

```
  1  1  1  1  1  1 <------- carries

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

Binary Subtraction

Rule 1: 0 – 0 = 0
Rule 2: 0 – 1 = 1 (with a borrow of 1)
Rule 3: 1 – 0 = 1
Rule 4: 1 – 1 = 0

Exercise: 1010 – 101

```
 1 0 1 0
   1 0 1 –
__________
 0 1 0 1
==========
```

Representing Negative Numbers

- **Signed Numbers Vs Unsigned Numbers**
- Historically, there are three approaches

  Sign-and- magnitude (Sign magnitude)

  One’s Complement

  Two’s Complement

- Two’s complement is the most important approach

  - Simplifies arithmetic

  - Used most universally

Sign Magnitude

- The sign-magnitude binary format is commonly applied in 8 bit systems
- A simple conceptual format that is used for represent signed numbers
- This notation deals with most significant bit (MSD) to indicate a positive or a negative value
- If the MSB is a **0**, this indicates that the number as a **positive** one
- If the MSB is a **1**, this indicates that the number is **negative**

Example

+36 → **0**0100100

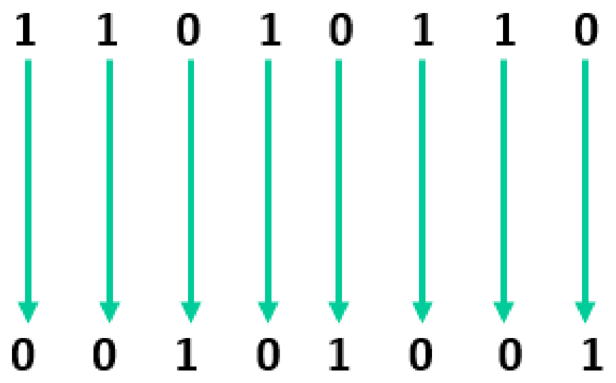
-36 → **1**0100100

One's Complement

- In one's complement, the given binary number will be inverted
- All the zero (0) bits will be replaced with ones (1)
- All the ones (1) bits will be replaced with zeros (0)

One’s Complement

Example : Find 1’s complement of binary number 11010110



The 1’s complement of the binary number 11010110 is 00101001

Two's Complement Representation

The 2's complement of a binary number can be calculated by simply inverting the given number and adding 1 to the least significant bit (LSB) of given result

Example : Find 2's complement of binary number 11010110

Step1: Get the 1's complement of the given number

1 1 0 1 0 1 1 0

0 0 1 0 1 0 0 1

Two's Complement Representation (Cont'd)

Step 2: Add 1 to the least significant bit (LSB)

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

The 2's complement of the binary number 11010110 is 00101010

One's Complement and Two's Complement

Exercise 9: Find one's complement of given binary numbers

1. 1100110
2. 1110010101

Exercise 10: Find two's complement of given binary numbers

1. 110101
2. 101010
3. 1100110

Half Adder

Add two binary numbers

A_0 , B_0 → Single Bit Inputs

S_0 → Single Bit Sum

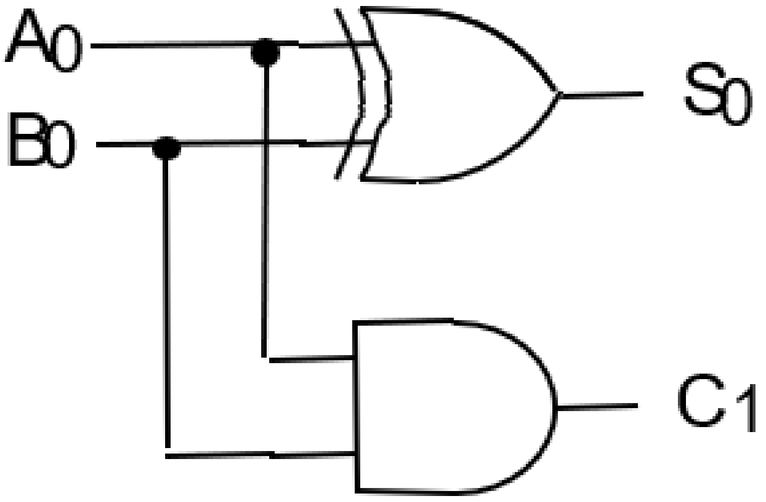
C_1 → Carry Out

Half Adder

Truth Table

A_0	B_0	S_0	C_1
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Logic Circuit



Full Adder

Truth Table

Inputs			Outputs	
A	B	C – IN	Sum	C - Out
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Full Adder

Logical Expression for SUM:

= A’ B’ C-IN + A’ B C-IN’ + A B’ C-IN’ + A B C-IN

= C-IN (A’ B’ + A B) + C-IN’ (A’ B + A B’)

= **C-IN XOR (A XOR B)**

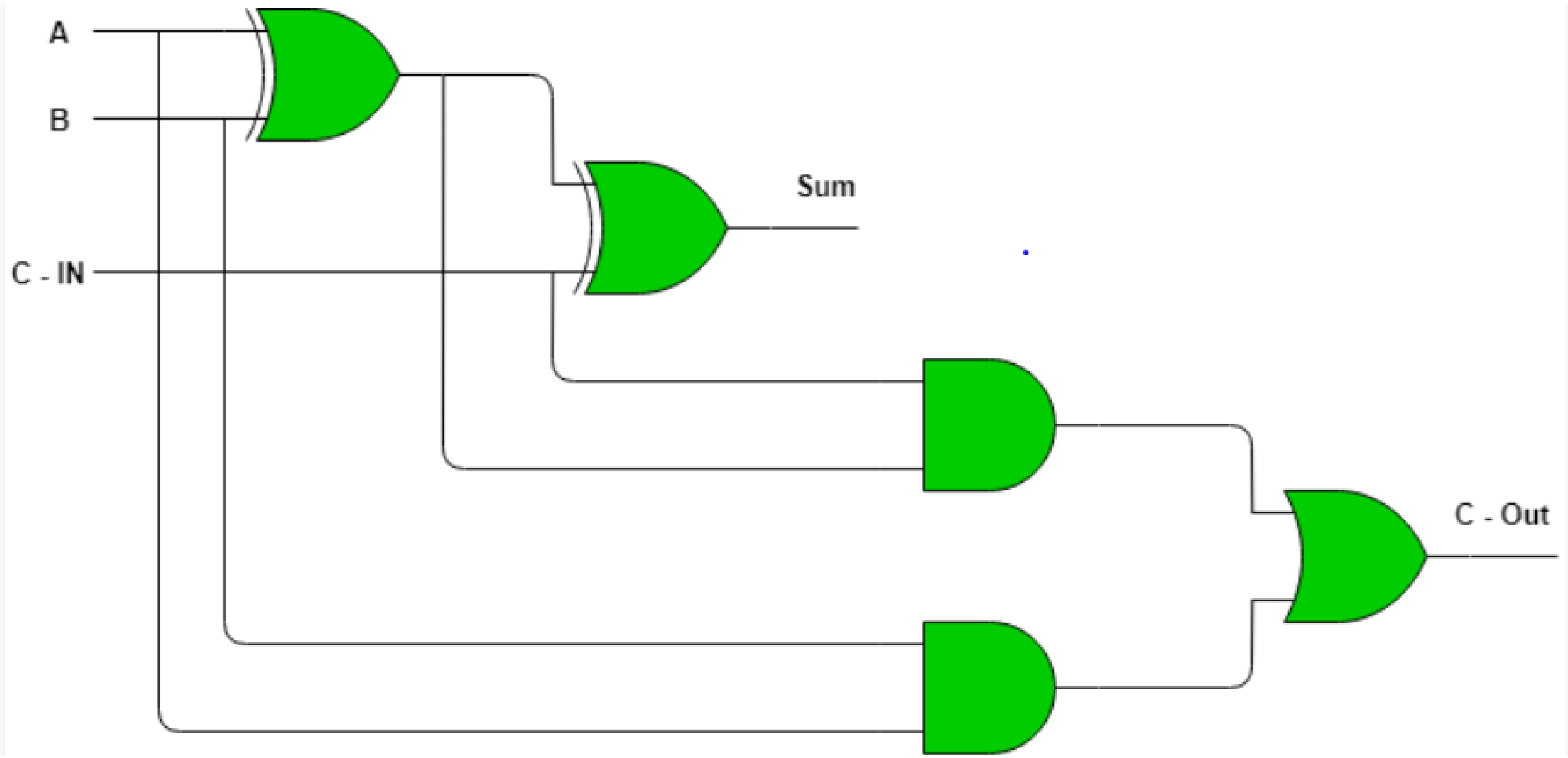
Logical Expression for C-OUT:

= A’ B C-IN + A B’ C-IN + A B C-IN’ + A B C-IN

= **A B + B C-IN + A C-IN**

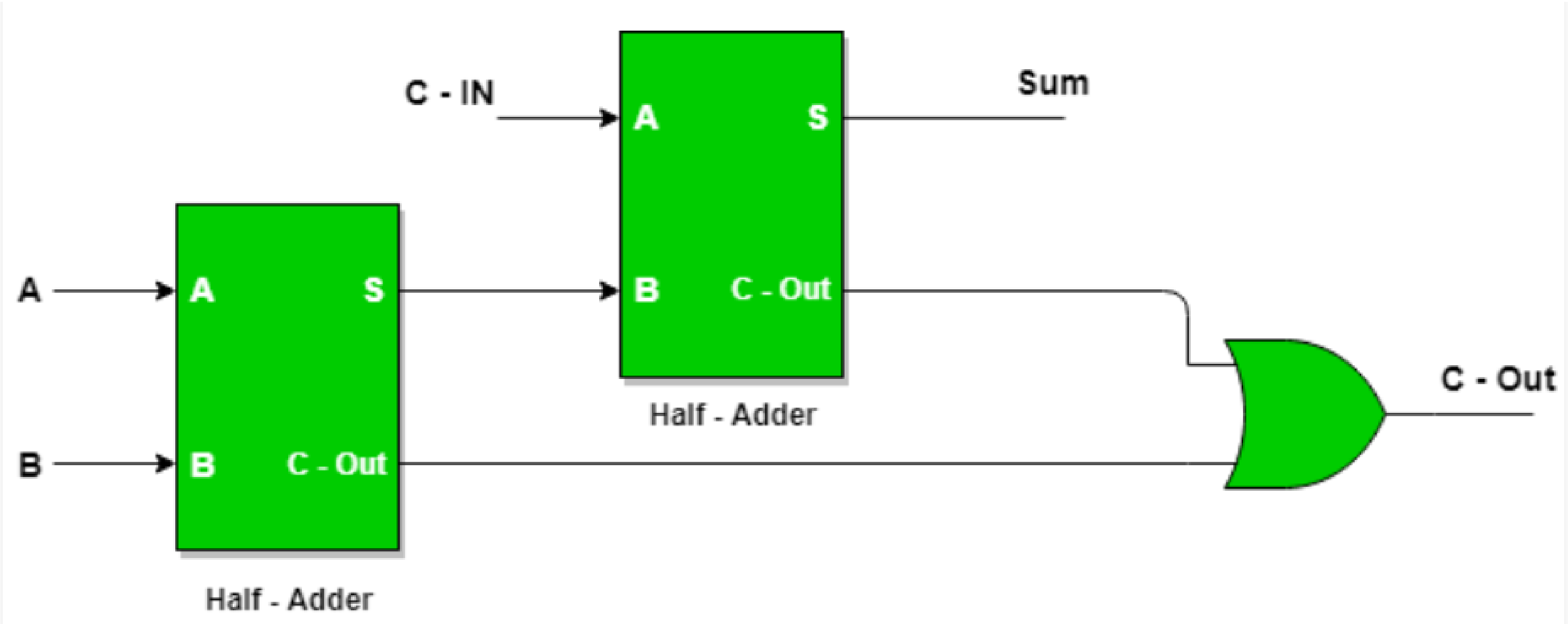
Full Adder

Logic Circuit



Full Adder

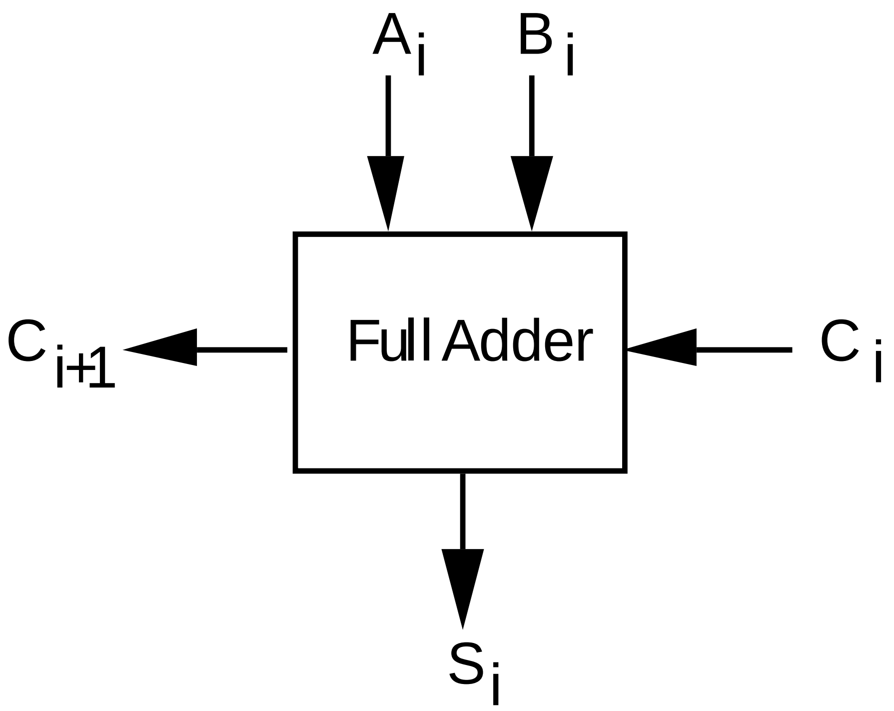
- **Hardware repetitions simplifies hardware design**
- A full adder can be made from two half adders (plus an OR gate)



Full Adder

Putting it all together

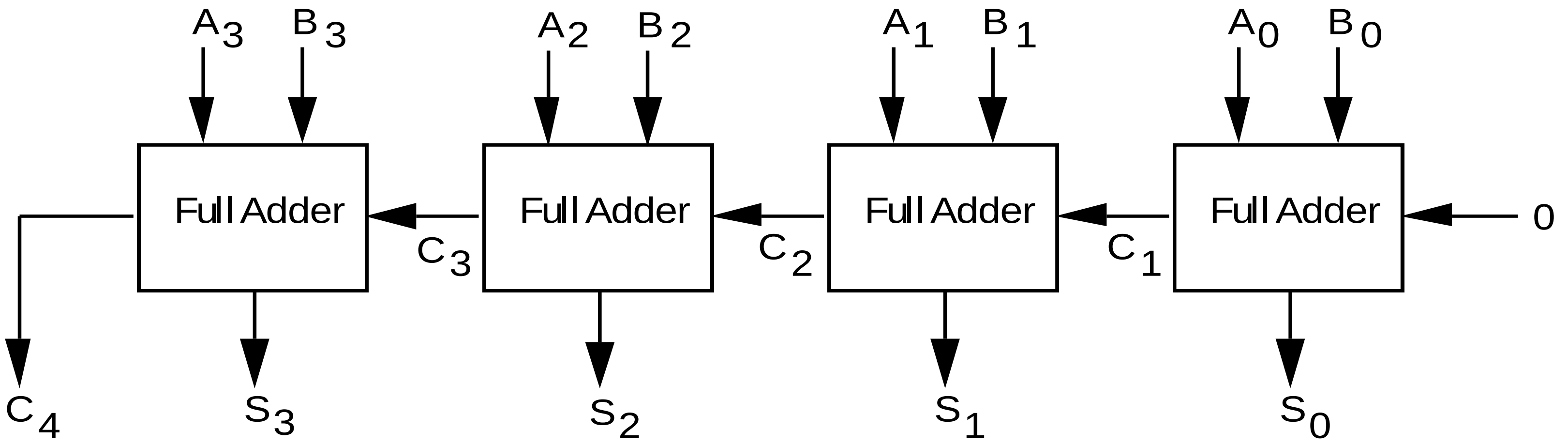
- Single-bit full adder
- Common piece of computer hardware



Block Diagram

4 Bit Adder

Chain single-bit adders together



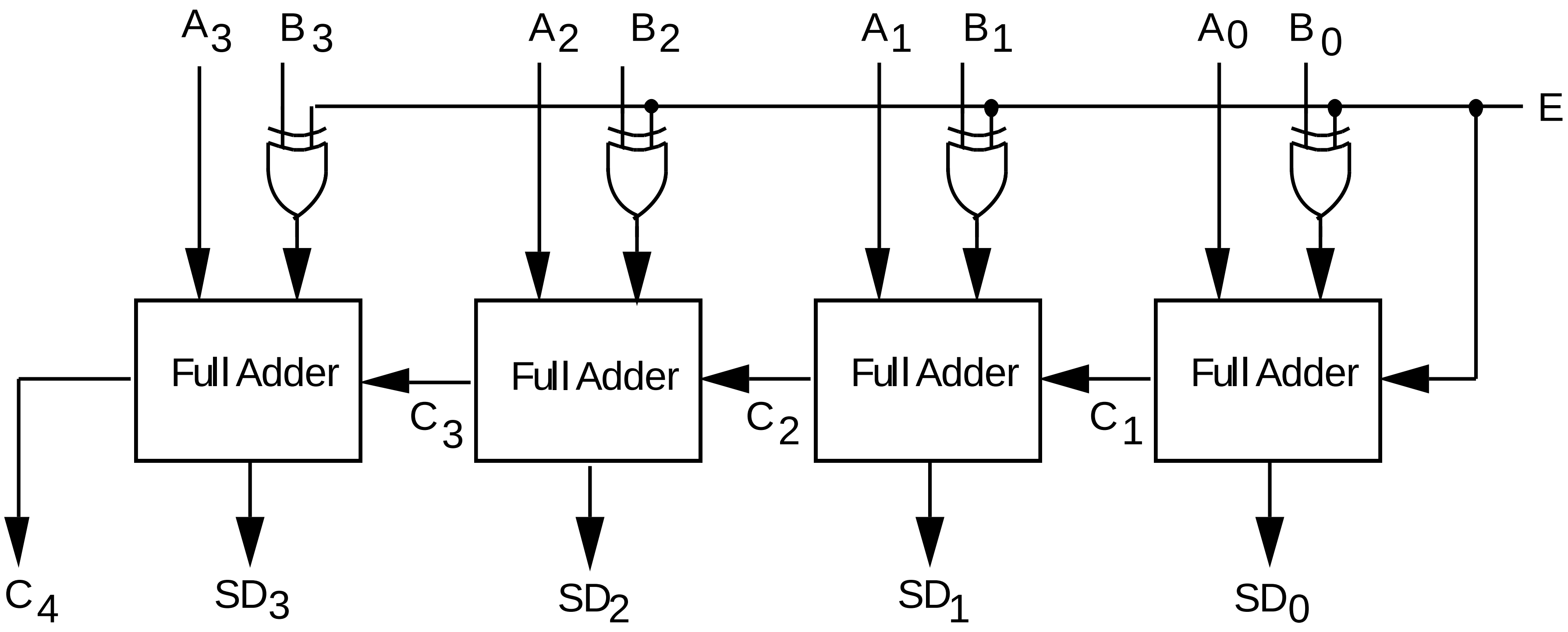
C	1	1	1	0
A	0	1	0	1
B	0	1	1	1
S	1	1	0	0

Negative Numbers – 2's Complement

Subtracting a number is the same as:

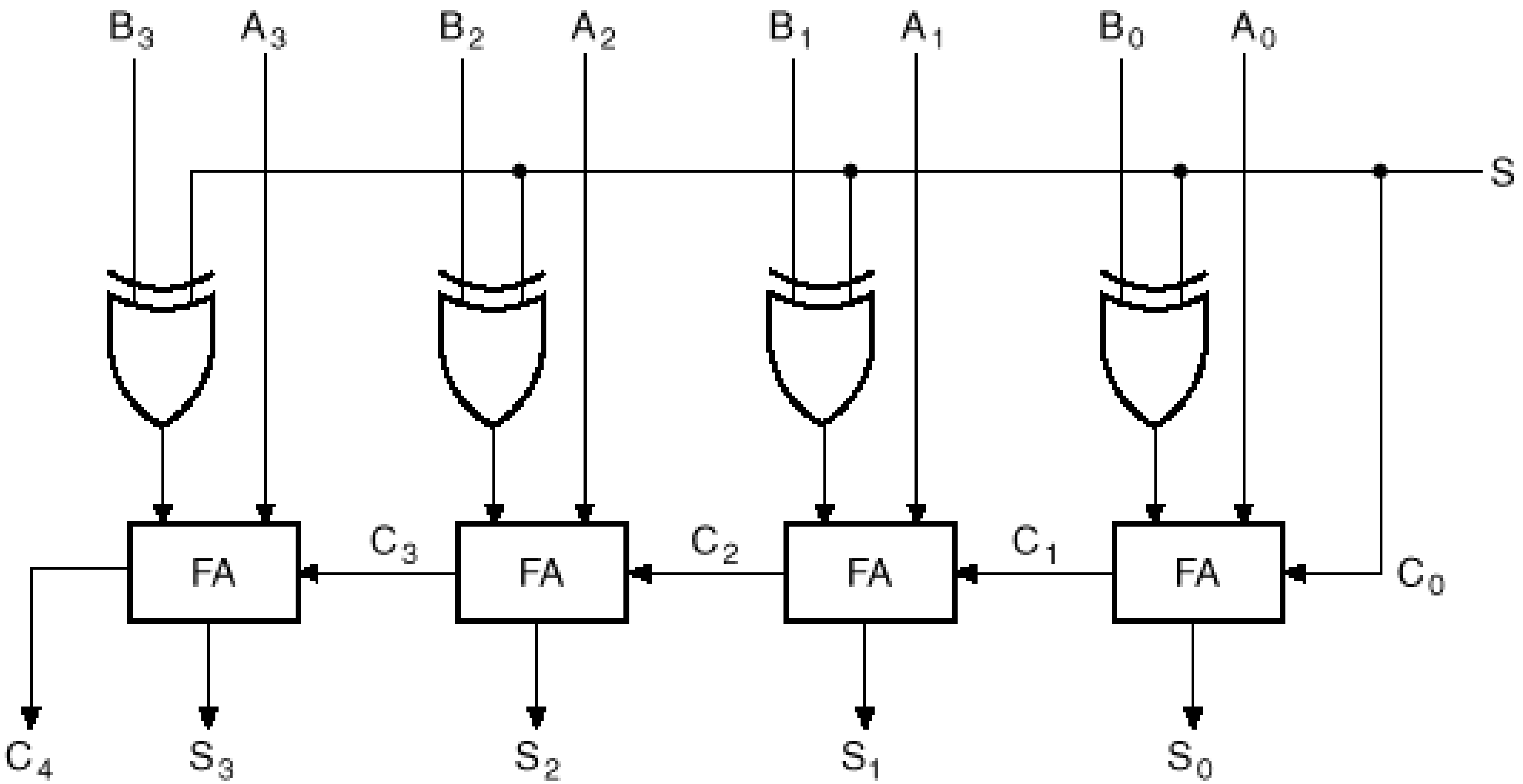
1. Perform 2's complement
2. Perform addition

4-Bit Subtractor



Add **A** to **B'** (one's complement) plus
That is, add **A** to two's complement of **B**
D = **A** - **B**

Adder-Subtractor Circuit



Thank You!