



BITS Pilani
Pilani Campus

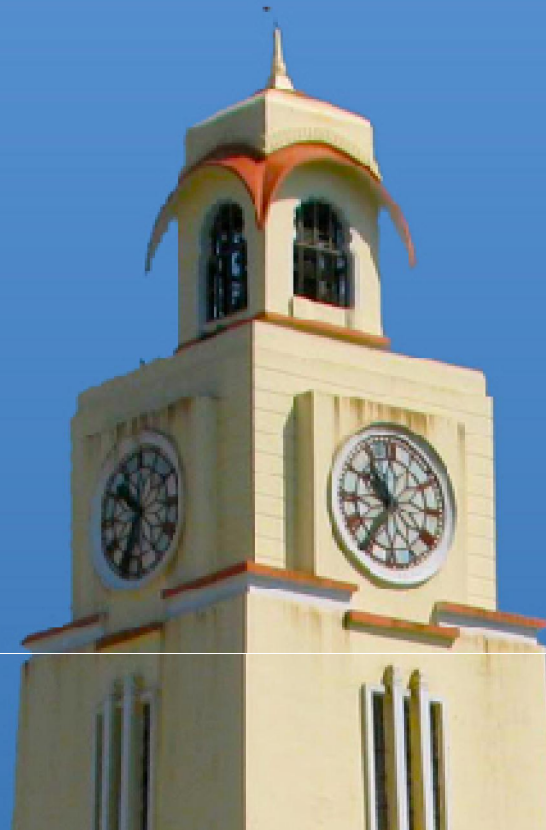
Computer Programming

Module-1 (Lecture-2)

Virendra Singh Shekhawat
Department of Computer Science & Information Systems



BITS Pilani
Pilani Campus



Topics to be Covered:

Numbers & Representations, Number Representation in Computers, Unsigned Binary Representation and Arithmetic Operations

Numbers and Representations-1



- Roman numerals
 - Total seven symbols are used (i.e. I, V, X, L, C, D, M)
 - Numbers are formed by combining symbols
 - Symbols are placed in order of value
 - e.g. MMXII (equals to 2012)

Numbers and Representations-2



- Arabic numerals
 - Total 0 to 9, ten symbols (digits) are used
 - Numbers are represented in a sequence of digits
 - e.g. 578, “read as a Five Hundred and Seventy Eight”



Numbers and Representations-3

- How to perform the arithmetic operations?
 - $V * V = XXV$
 - $VI * VIII = XXXXVIII$
 - $5 * 5 = 25$
 - $6 * 8 = 48$
- Can we define the steps of multiplication?

Numbers and Representations-4



- Looking for a representation that allows operations in efficient manner
- Natural numbers are represented in base 10
- What is good for computers?
 - Answer: Base 2 (Binary). Why?

Binary Numbers

- Two symbols (digits) are used (i.e. 0 and 1)
- Examples: 1001, 0110, 1111, 1010
- The decimal equivalent of a Binary number $d_{n-1} d_{n-2} \dots d_0$ is given as:
$$d_{n-1} * 2^{n-1} + d_{n-2} * 2^{n-2} + \dots + d_0 * 2^0$$
- Decimal equivalent of 10101?



Decimal to Binary Conversion

- How to convert a decimal number N to its equivalent Binary number?
 1. Divide the number N by 2
 - the remainder becomes the Least Significant Bit (LSB)
 2. Divide the quotient by 2 (if it is not 0)
 - the remainder becomes the next bit to the left
 3. Repeat the step 2 until the quotient is 0

Decimal to Binary Conversion



- Exercise:
 - Find the binary equivalent of the decimal number 25



Questions

- Using K bits, what is the range of decimal numbers one can represent?
- Given a decimal number N , how many bits are required to represent it in binary?

Binary Addition Table

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

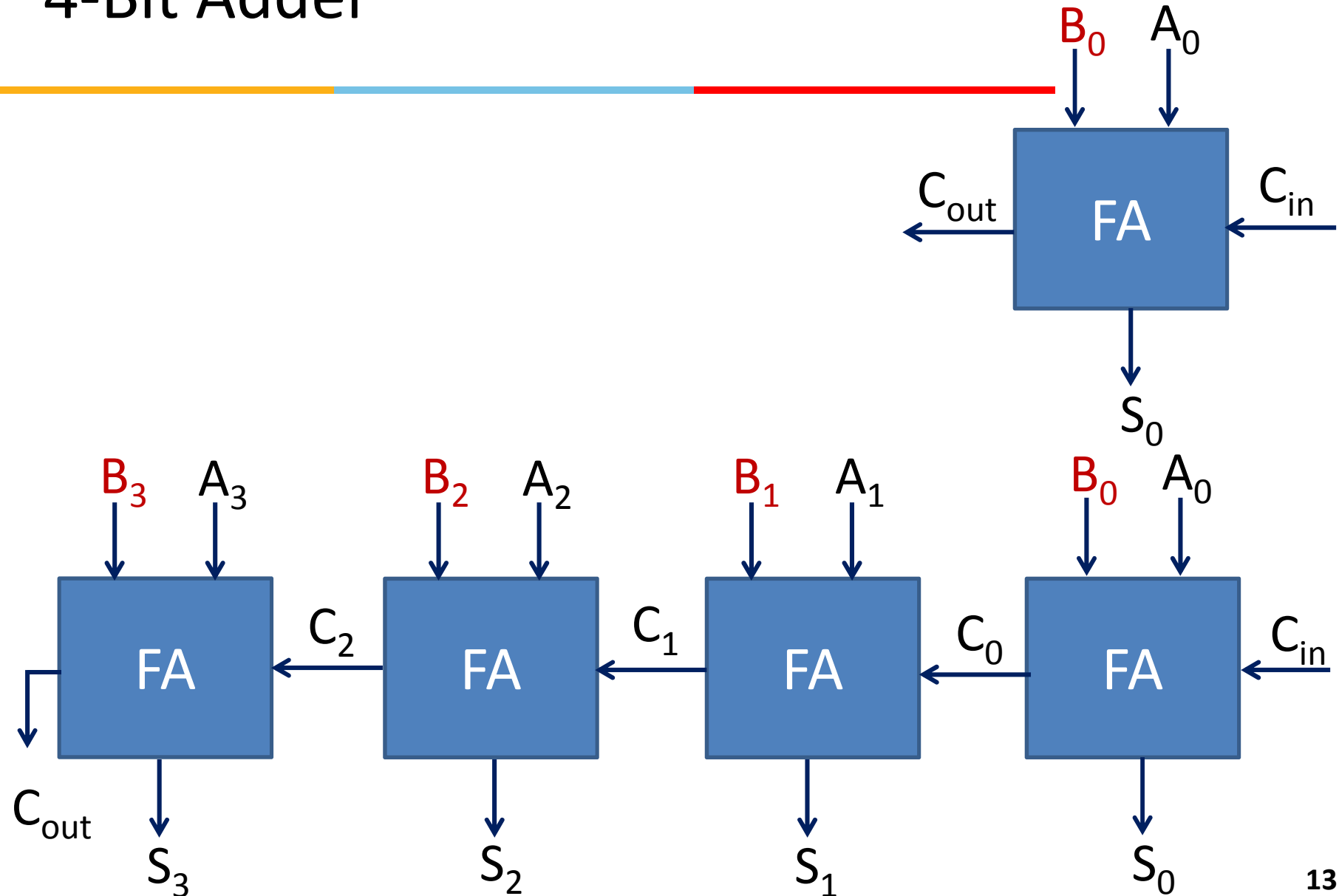


Arithmetic Operations

- How to add two unsigned binary numbers?
- Procedure
 - Right align the numbers
 - Add the two corresponding bit and a **carry-in** bit (*if any*) to get a single sum bit and a **carry-out** bit
- Example:

$$\begin{array}{r} 110011 \\ +10011100 \\ \hline \end{array}$$

4-Bit Adder





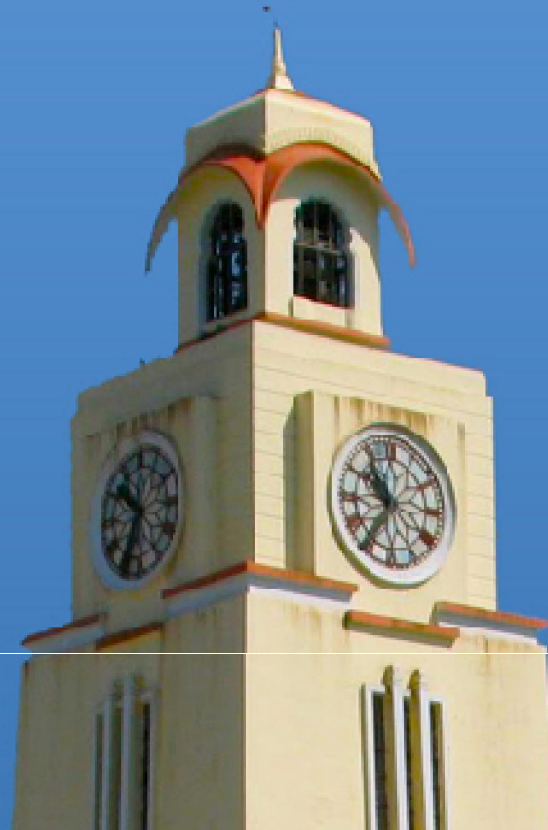
Overflow

- There are finite number of bits available in computers to represent the numbers
 - For a 8 bit computer, the maximum number (*unsigned binary*) which can be represented is $(2^8 - 1)$ or 255
- Let's add the two 8 bit binary numbers:

$$\begin{array}{r} 11110000 \\ + 00010001 \\ \hline \end{array}$$



BITS Pilani
Pilani Campus



Thank You!