# Digital Design

## 1s and 2s complement subtraction and Sign-Magnitude Representation

# Complements

- **Why Complements ….?**
  - To simplify the subtraction operation and for logical manipulation
  - Computer does not directly subtract, it adds…
- **Two Types of Complements in binary**
  - 1's complement
  - 2's complement

r is the radix of a given number

# Binary 1's Complement Example

**For $r = 2$, $N = 01110011_2$, $n = 8$ (8 digits), we have:**

$$(r^n - 1) = 256 - 1 = 255_{10} \text{ or } 11111111_2$$

**The 1's complement of $01110011_2$ is then:**

$$\begin{array}{r} 11111111_2 \\ -\ 01110011_2 \\ \hline 10001100_2 \end{array}$$

**NOTE: Since the $2^n$-1 factor consists of all 1's and since 1 - 0= 1 and 1 - 1 = 0, forming the one's complement consists of complementing each individual bit.**

# Binary 2's Complement Example

**For $r = 2$, $N = 01110011_2$, $n = 8$ (8 digits), we have:**

$$(r^n) = 256_{10} \text{ or } 100000000_2$$

**The 2's complement of $01110011_2$ is then:**

$$
\begin{array}{r}
100000000_2 \\
-\ 01110011_2 \\
\hline
10001101_2
\end{array}
$$

**Note that this is the 1's complement plus 1.**

# Using (r-1)s Complement

- **Case 1: M-N, when M>=N**
  - Take (r-1)s complement of N
  - Add M + N
  - Add carry propagated to the result to obtain the answer
- **Case 2: M-N, when M<N**
  - Take (r-1)s complement of N
  - Add M + N
  - Take (r-1)s complement of the result to obtain the final answer

# Using (r)s Complement

- **Case 1: M-N, when M>=N**
  - Take (r)s complement of N
  - Add M + N
  - Ignore the carry to get the final answer

- **Case 2: M-N, when M<N**
  - Take (r-1)s complement of N
  - Add M + N
  - Take (r)s complement of the result to obtain the final answer

# Example (10)s Complement

Example: Find $543_{10} - 123_{10}$

1). Form 10's complement of 123:

```
 1000
-  123
------------
   877
```

2). Add the two:

```
      543
(+)   877
--------------
     1420
```

3). Since M $\geq$ N, we discard the carry. Ans: 420

# Using (1)s Complement

- **Case 1: M-N** 1010 - 1000
  - Take (1)s complement of N 0111
  - Add M + N 1010 + 0111 = 10001
  - Add carry propagated to the result to obtain the answer

0001 + 0001 = 0010

- **Case 2: M-N, when M<N** 1000-1010
  - Take (1)s complement of N 0101
  - Add M + N 1000+0101 = 1101
  - Take (1)s complement of the result to obtain the final answer

1s complement of 1101 is 0010

# Using (2)s Complement

- Case 1: M-N

  1010 - 1000

  - Take (2)s complement of N

    1000

  - Add M + N

    1010 + 1000 = 10010

  - Ignore the carry to get the final answer

    10010= 0010

- Case 2: M-N, when M<N

  1000-1010

  - Take (2)s complement of N

    0110

  - Add M + N

    1000+0110 = 1110

  - Take 2s complement of the result to obtain the final answer

    2s complement of 1110 is 0010

# Exercise (attempt with 1s and 2s both)

- 11111 - 10101
- 11110 – 1001
- 10101-11111
- 1001-11110

# Sign Magnitude Representation

- Positive numbers and zero can be represented by unsigned n-digit, radix r numbers.
- Need a representation for negative numbers
- To represent a sign (+ or -) we need exactly one more bit of information (1 binary digit gives $2^1 = 2$ elements which is exactly what is needed)
- Since most computers use binary numbers, the most significant bit is interpreted as a sign bit as shown below:

**S Bn-1 Bn-2 … B2 B1 B0**

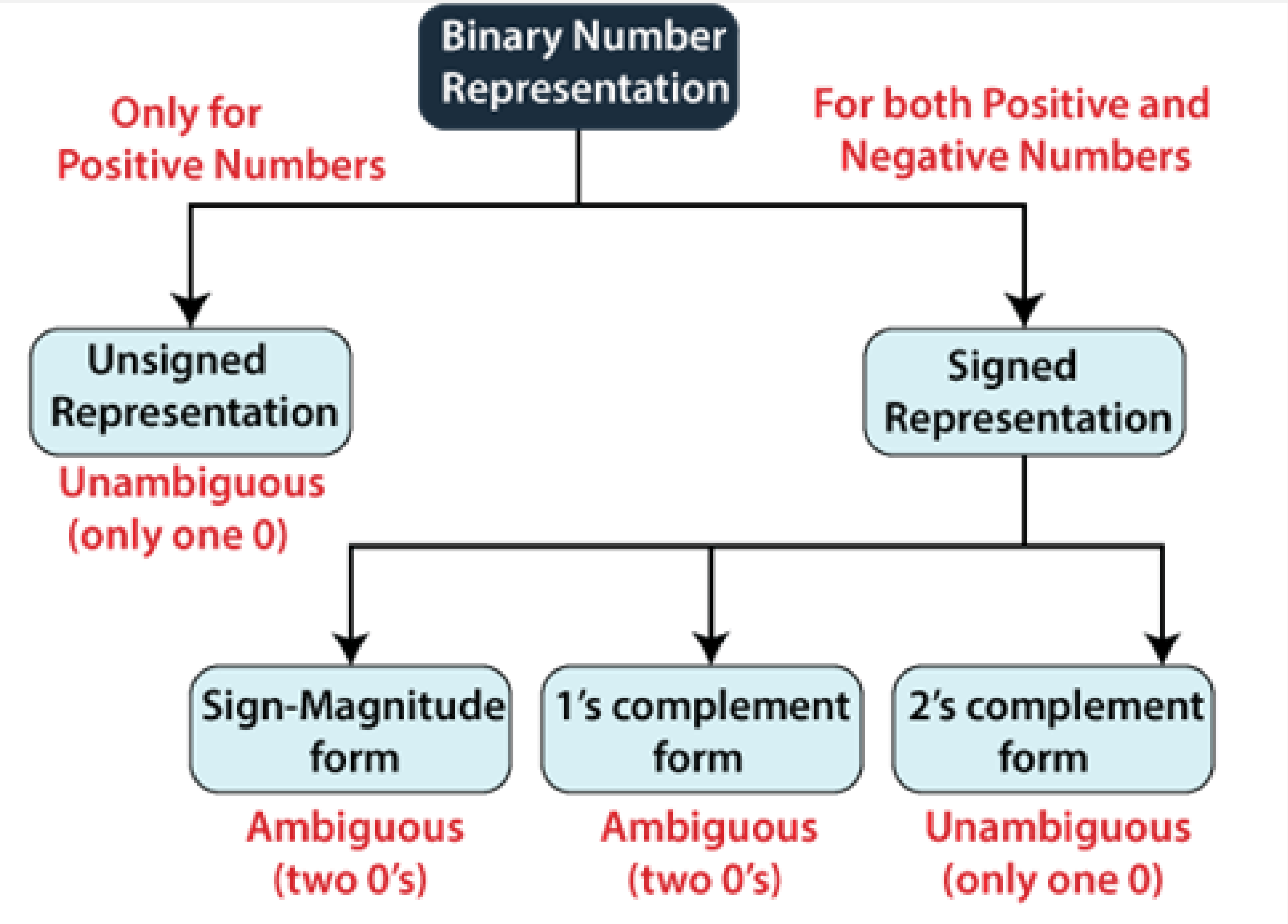
Where: S = 0 for Positive numbers

S = 1 for Negative numbers

and Bn can be 0 or 1 (data)

# Sign Magnitude Representation

- Sign Magnitude Representation of Binary Numbers



# Sign Magnitude Representation

- We have the following interpretations for signed integer representation of three bits in binary

| Number | Sign-Mag. | 1's Comp. | 2's Comp. |
|---|---|---|---|
| +3 | 011 | 011 | 011 |
| +2 | 010 | 010 | 010 |
| +1 | 001 | 001 | 001 |
| **+0** | **000** | **000** | **000** |
| **-0** | **100** | **111** | **---** |
| -1 | 101 | 110 | 111 |
| -2 | 110 | 101 | 110 |
| -3 | 111 | 100 | 101 |
| -4 | --- | --- | 100 |

# Random Examples

# Thank You