# CSD1100

# Binary Arithmetic Operations

## Vadim Surov

# Binary Arithmetic Operations

- Basic arithmetic operations:
  - Addition ( + )
  - Subtraction ( - )
  - Multiplication ( * )
  - Division ( / )
- Basic mathematical operations with binary numbers works similar to the decimal system.
- However there are a few rules specific to the binary system.
- We'll look at each of them individually.

# Binary Addition

# Binary Addition

- There are 3 basic rules for adding binary numbers:
  1. 0 + 0 = 0
  2. 0 + 1 = 1 and 1 + 0 = 1
  3. 1 + 1 = 10. So we keep 0 in the 1's column and shift (carry over) 1 to the 2's column.
- Other rules are same as the decimal system:
  - we add from right to left and
  - the carry over get's added to the digits in the next column.

# Binary Addition

- Now lets try adding 11 to 13.
- Binary for 11 is 1011 and that for 13 is 1101.

```
   1 0 1 1
 +
   1 1 0 1
 _____
```

# Binary Addition

- 1st col = 1 + 1 = 10. We keep 0 in 1st col and carry over 1 to 2nd col.

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

# Binary Addition

- 2nd col = 1 + 0 + 1 (carry over) = (1 + 0) + 1 = 1 + 1 = 10. Once again we keep 0 in 2nd col and carry over 1 to 3rd col.

$$
\begin{array}{r}
1\phantom{000} \\
1\ 0\ 1\ 1 \\
+\ 1\ 1\ 0\ 1 \\
\hline
0\ 0
\end{array}
$$

# Binary Addition

- 3rd col = 0 + 1 + 1 (carry over) = (0 + 1) + 1 = 1 + 1 = 10. Keep 0 in 3rd col and carry over 1 in 4th col.

$$
\begin{array}{ccccc}
  &   & 1 &   &   \\
  & 1 & 0 & 1 & 1 \\
+ & 1 & 1 & 0 & 1 \\
\hline
  &   & 0 & 0 &   \\
\end{array}
$$

# Binary Addition

- 4th col = 1 + 1 + 1 (carry over) = (1 + 1) + 1 = 10 + 1 = 11. Keep 1 in 4th col and carry over 1 in 5th col.

$$
\begin{array}{r}
1\phantom{000} \\
1\ 0\ 1\ 1 \\
+\ 1\ 1\ 0\ 1 \\
\hline
1\ 0\ 0\ 0 \\
\end{array}
$$

# Binary Addition

- 5th col = 0 + 0 + 1 (carry over) = (0 + 0) + 1 = 1. Keep 1 in 5th col. Done.

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

# Binary Addition

- The sum is 11000. $11000 = 1 * 2^4 + 1 * 2^3 + 0 * 2^2 + 0 * 2^1 + 0 * 2^0 = 16 + 8 + 0 + 0 + 0 = 24 = 11 + 13$.

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

# Binary Multiplication

# Binary Multiplication

- Multiplication in binary is exactly as it is in decimal, i.e. multiply numbers right to left and multiply each digit of one number to every digit of the other number, them sum them up.
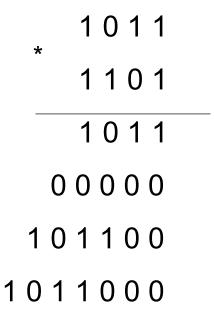- The 3 basic binary multiplication rules are also similar to decimal.
  - 1 * 1 = 1
  - 1 * 0 = 0 * 1 = 0
  - 0 * 0 = 0
- Also, remember that for every left shift of digit of the multiplier, an extra zero needs to be appended to the product. This is similar to the decimal system as well.

# Binary Multiplication

```
          1 0 1 1
    *
          1 1 0 1
        _____
          1 0 1 1
```

- 1011 * 1 (multiplier 1's col) = 1011

# Binary Multiplication

$$1 \; 0 \; 1 \; 1$$
$$* \qquad \qquad$$
$$1 \; 1 \; 0 \; 1$$

---

$$1 \; 0 \; 1 \; 1$$

$$0 \; 0 \; 0 \; 0 \; 0$$

- 1011 * 0 (multiplier 2's col) = 00000 (one zero appended at the end)

# Binary Multiplication

```
        1 0 1 1
  *
        1 1 0 1
      _____
        1 0 1 1

      0 0 0 0 0

    1 0 1 1 0 0
```

- 1011 * 1 (multiplier 4's col) = 101100 (two zero's appended at the end)

# Binary Multiplication

$$
\begin{array}{r}
1\ 0\ 1\ 1 \\
*\quad 1\ 1\ 0\ 1 \\
\hline
1\ 0\ 1\ 1 \\
0\ 0\ 0\ 0\ 0 \\
1\ 0\ 1\ 1\ 0\ 0 \\
1\ 0\ 1\ 1\ 0\ 0\ 0 \\
\end{array}
$$

- 1011 * 1 (multiplier 8's col) = 1011000 (three zero's appended at the end)

# Binary Multiplication

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

- Sum up.

# Binary Multiplication

```
          1 0 1 1
    *
          1 1 0 1
        _____

          1 0 1 1
      +
         0 0 0 0 0
      +
        1 0 1 1 0 0
    +
       1 0 1 1 0 0 0
        _____
      1 0 0 0 1 1 1 1
```

- So the product is 10001111 which is = $1 * 2^7 + 1 * 2^3 + 1 * 2^2 + 1 * 2^1 + 1 * 2^0 = 128 + 8 + 4 + 2 + 1 = 143 = 11 * 13$.

# Binary Subtraction

# Binary Subtraction

- Subtraction can be done as addition when second operand is changed to it's negative value:

$$10 - 5 = 10 + -5$$

- Negative binaries will be defined soon.
- Another way is to use idea of **borrowing** from decimal arithmetic.

# Binary Subtraction

- When subtracting binary numbers, there are four points or steps to remember before proceeding through the operation. These are:

$$0-0 = 0$$
$$0-1 = 1, \; borrow \; 1 \; from \; the \; next \; more \; significant \; bit$$
$$1-0 = 1$$
$$1-1 = 0$$

# Binary Subtraction

- Example 10101100 - 1010100.
- The first step is 0 – 0 = 0 and that's what is written in the place for result.

$$\begin{array}{r} 10101100 \\ 1010100 \\ \hline 0 \end{array}$$

# Binary Subtraction

- Similarly again the last step is repeated as here the numbers are both 0 and from the table we know 0 – 0 = 0.

```
10101100
 1010100
_____
      00
```
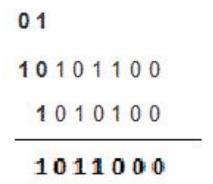
# Binary Subtraction

- From the table, we can find out that 1 – 1 = 0 and it is written

```
1 0 1 0 1 1 0 0

  1 0 1 0 1 0 0
_____

      0 0 0
```

# Binary Subtraction

- The table shows that 1 – 0 = 1 and we have written exactly that in result

```
1 0 1 0 1 1 0 0
  1 0 1 0 1 0 0
_____
        1 0 0 0
```

# Binary Subtraction

● Here 0 – 1 = 1 with the borrowing of 1 from the next significant bit and that's what has been done. We will treat the next 1 as 0 in the next step as shown on next slide.

```
        0 1
10 101 100
  1 010 100
_____
   11 000
```

# Binary Subtraction

- As the 1 was borrowed in the previous step we are treating the 1 as 0 and the result is 0 – 0 = 0 and that is written.
- Again the last 1 has been borrowed because the operation done was 0 – 1 = 1 with borrow 1 from the next most significant bit and the final result of binary subtraction, we got is written in the place of result in the final step.

```
0 1

1 0 1 0 1 1 0 0

  1 0 1 0 1 0 0
_____

  1 0 1 1 0 0 0
```

# Binary Division

# Binary Division

- Binary division is similar to decimal division.
- Binary division is opposite to multiplication that uses shift to left and addition, so division is made of shifts to right and subtractions (where subtraction is addition of a made negative value).

# References

- https://medium.com/@malaybiswas/binary-addition-multiplication-subtraction-and-division-55ad8d27ff02
- https://www.electrical4u.com/binary-subtraction/