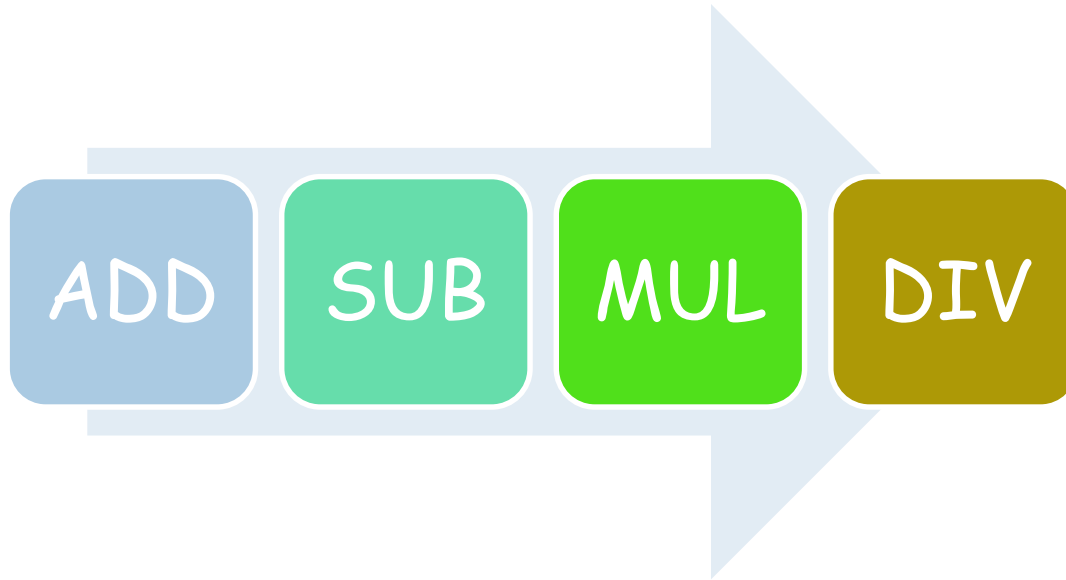


Arithmetic Operations

100010101001010100010010010101010

Decimal and Binary

Four Basic Operations



ALU and the math operations are the heart of the CPU

The computer-CPU (ALU) performs ... mathematical operations ?

Main math operation of ALU ?

Main math operation of ALU: Add

- All the other mathematical operations are performed, using addition, and ...
- HOW ?

Decimal addition


$$\begin{array}{r} 369 \\ + 32 \\ \hline 401 \end{array}$$

Binary addition

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

1 = sum

Binary addition

		
1		0
+0		+1
<hr/>		<hr/>
1 = sum		1 = sum

Binary addition



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

1 = sum

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

1 = sum

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

0 = sum

1 = carry

Example

$$\begin{array}{r} 10110 \\ +10011 \\ \hline 1 \end{array}$$

Example

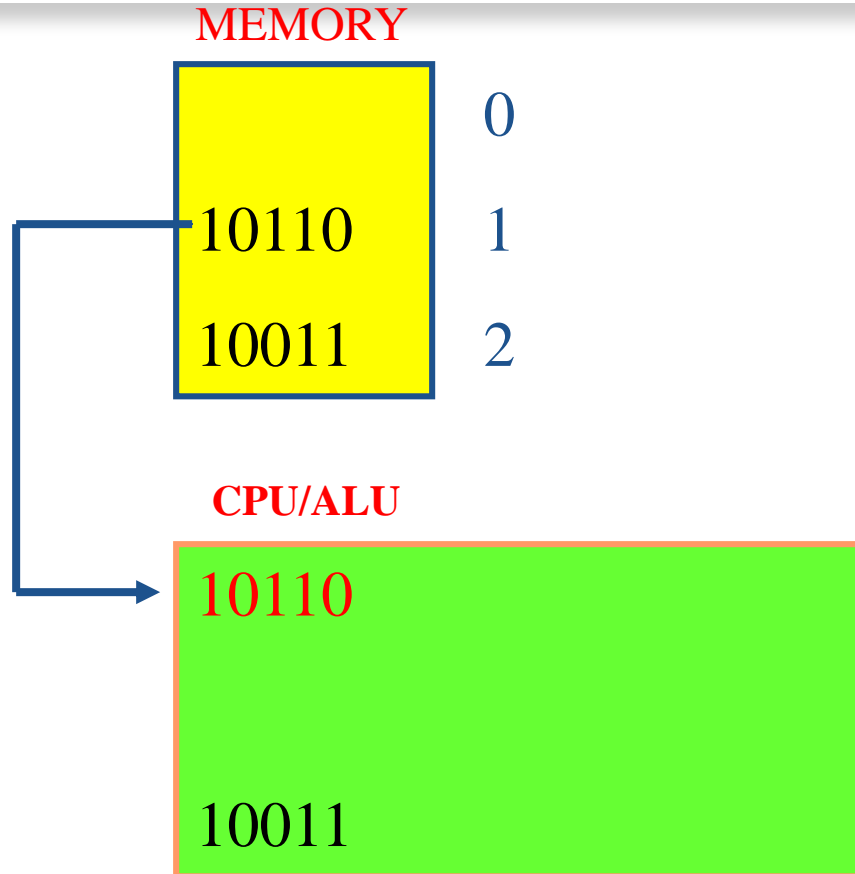
$$\begin{array}{r} 01100 \leftarrow \text{Carry} \\ 10110 \\ +10011 \\ \hline 101001 \leftarrow \text{Result} \\ \begin{array}{cc} \uparrow & \uparrow \\ \text{MSBit} & \text{LSBit} \end{array} \end{array}$$

Add two binary numbers

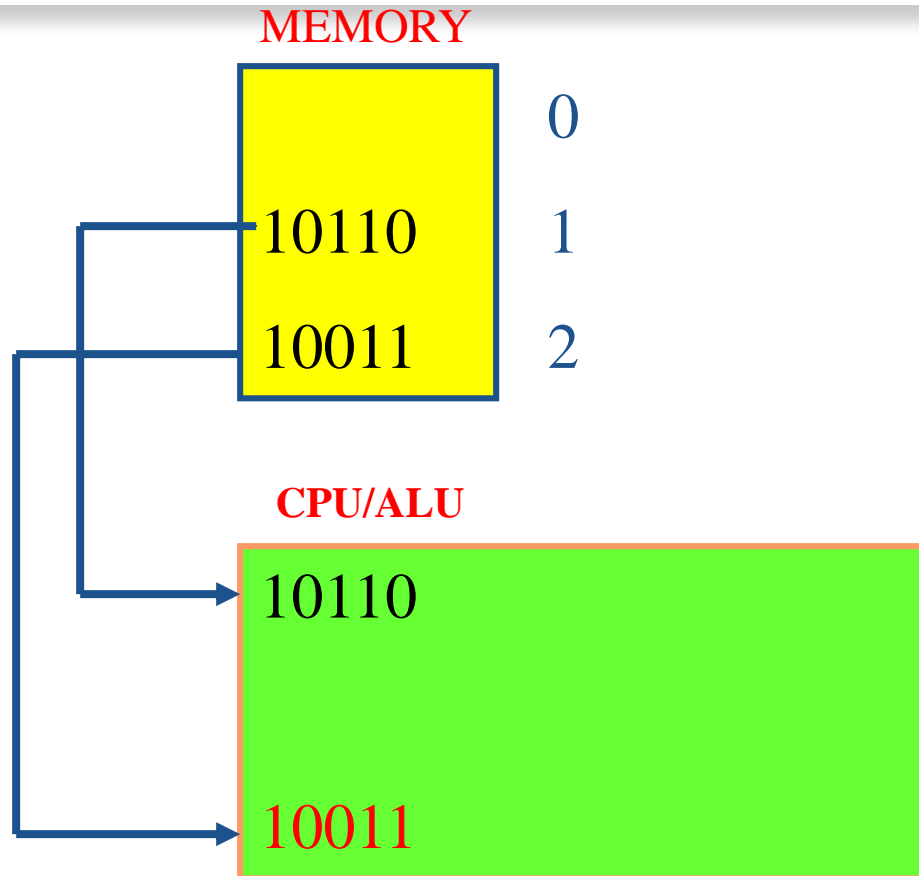
MEMORY

	0
10110	1
10011	2

Add two binary numbers



Add two binary numbers



Add two binary numbers

MEMORY

0

1

2

10110

10011

CPU/ALU

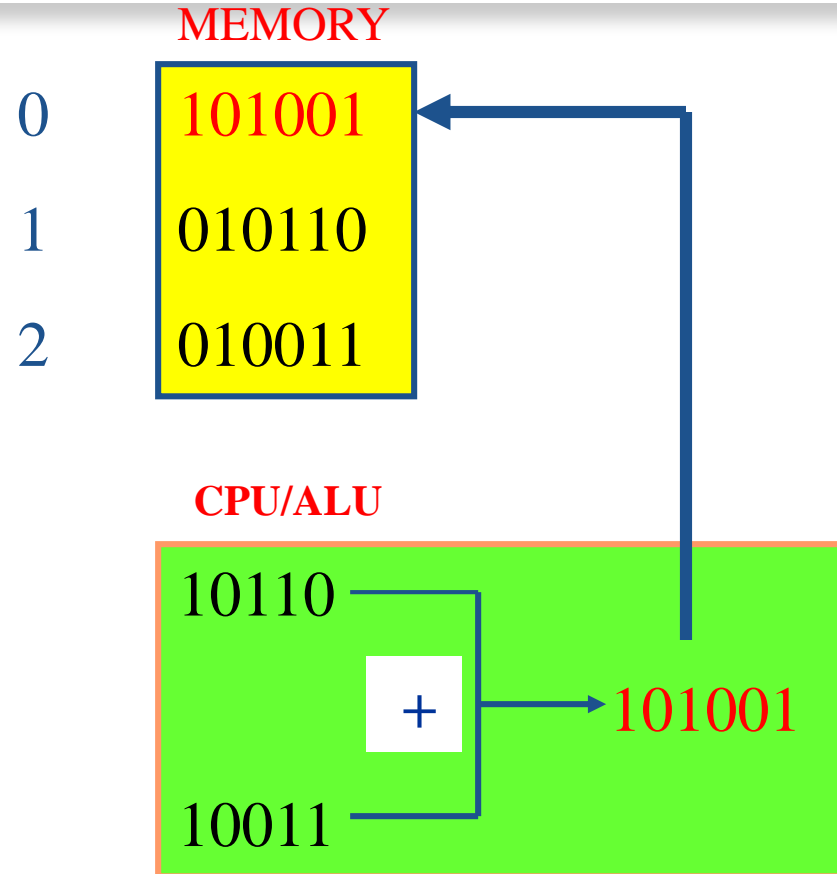
10110

10011

+

101001

Add two binary numbers



10011, 10110 and 101001 are stored in the CPU/ALU in Registers

Addition using Octal (1)

$$\begin{array}{r} 1476 \\ +3554 \\ \hline ? \end{array}$$

Addition using Octal (1)

- 1476
- +3554
- ---

?

$$6+4 = 10 \Rightarrow (1, 2)$$

↑ ↑
carry sum

1

1476

3554

2

Addition using Octal (2)

- 1476
- +3554
- ---

$$1+7+5 = 13 \Rightarrow \begin{matrix} (1, & 5) \\ \uparrow & \uparrow \\ \text{carry} & \text{sum} \end{matrix}$$

$$\begin{array}{r} 11 \\ 1476 \\ 3554 \\ \hline 52 \end{array}$$

Addition using Octal (3)

- 1476
- +3554

$$1+4+5 = 10 \Rightarrow (1, 2)$$

↑ ↑
carry sum

$$\begin{array}{r} 111 \\ 1476 \\ 3554 \\ \hline 252 \end{array}$$

Addition using Octal (4)

- 1476
- +3554

$$1+1+3 = 5 \Rightarrow \begin{matrix} (0, & 5) \\ \uparrow & \uparrow \\ \text{carry} & \text{sum} \end{matrix}$$

111

1476

3554

5252

Addition using Hex (1)

- 59F
 - +E46
-

Addition using Hex (1)

59F
+E46

F+6 = 15 + 6 = 21 => (1, 5)
 ↑ ↑
 carry sum

1
59F
E46

5

Addition using Hex (2)

- 59F
 - +E46
-

$1 + 9 + 4 = 14 \Rightarrow (0, \text{E})$
 ↑ ↑
 carry sum

01

59F

E46

E5

Addition using Hex (3)

- 59F
 - +E46
-

$$0 + 5 + E = 19 \Rightarrow \begin{matrix} (1, & 3) \\ \uparrow & \uparrow \\ \text{carry} & \text{sum} \end{matrix}$$

01

59F

E46

13E5

Subtraction




Decimal subtraction

$$\begin{array}{r} 401 \\ - 32 \\ \hline 369 \end{array}$$


Binary subtraction

$$\begin{array}{r} 0 \\ - 0 \\ \hline 0 \end{array}$$


Binary subtraction

		
0	1	
- 0	- 0	
<hr/>	<hr/>	
0	1	

Binary subtraction

			
0	1	1	
- 0	- 0	- 1	
<hr/>	<hr/>	<hr/>	
0	1	0	

Binary subtraction

			
0	1	1	0
- 0	- 0	- 1	- 1
<hr/>	<hr/>	<hr/>	<hr/>
0	1	0	1 = result
			1 = borrow

Example

$$\begin{array}{r} 1 \quad 0 \quad 1 \quad 1 \quad 0 = 2+4+16 = 22 \\ - 1 \quad 0 \quad 0 \quad 1 \quad 1 = 1+2+16 = 19 \\ \hline \end{array}$$

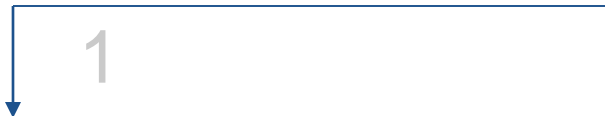
Example (1)

Since 0-1

$$\begin{array}{r} 10110 \\ - 10011 \\ \hline \end{array}$$

Example (1a)

Borrow 1 from next (left) column



1

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

The diagram illustrates a binary subtraction problem. The top row is 1 0 1 1 0 and the bottom row is - 1 0 0 1 1. A dashed line is below the bottom row. A horizontal line with a downward arrow points from the fourth column (the first '1' in the bottom row) to the fifth column (the '0' in the top row). Above this arrow is a '1', indicating a borrow of 1 from the fourth column to the fifth column.

Example (1b)

Borrow 1 from next (left) column

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

1



Therefore $10 - 1 = 1$

Example (1c)

Since we borrowed a 1 a 0 is left


0 1

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

1

Example (1d)

Since $0 - 1$, borrow 1 from next (left) column


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

Example (1e)

Borrow 1 from next (left) column

Diagram illustrating a 1D cellular automaton state. The grid shows the following values:

	1	0	1	1	0
-	1	0	0	1	1
<hr/>					
					1

A blue arrow points from the blue '1' in the second row, fourth column to a grey '1' above it. A green '0' is in the top row, fifth column. A yellow '1' is in the second row, fifth column. A blue diagonal line is over the blue '1' in the second row, fourth column.

Example (1f)

$$\begin{array}{r} 101\cancel{1}0 \\ -100\color{yellow}{1}1 \\ \hline 1\color{red}{1}1 \end{array}$$

Therefore $10\text{-}1 = 1$

Example (1g)

Since we borrowed a 1 a 0 is left

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

Example (1h)

Since we borrowed a 1 a 0 is left

$$\begin{array}{r} 0 \\ 1 0 \cancel{1} 1 0 \\ - 1 0 0 1 1 \\ \hline 0 1 1 \end{array}$$

Therefore $0 - 0 = 0$

Example (1i)

$$\begin{array}{rcccr} 1 & 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & 1 & 1 \\ \hline & 0 & 0 & 0 & 1 \end{array}$$

\uparrow
 $0 - 0 = 0$

Example (1j)

$$\begin{array}{rcccl} & 1 & 0 & 1 & 1 & 0 \\ - & 1 & 0 & 0 & 1 & 1 \\ \hline & 0 & 0 & 0 & 1 & 1 \end{array}$$

↑

$1 - 1 = 0$

Result: $22 - 19 = 3$

$$\begin{array}{r} 10110 \\ - 10011 \\ \hline 00011 \end{array}$$

Multiplication

- Shift + Add operation



Binary multiplication

101
X 100

000

000

+101

10100

100 add zeros

100 shift-add zeros

100 shift-copy number

Add

Binary multiplication

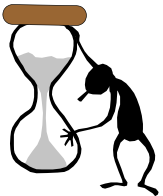
We Need:

- An Adder
- A Shifter

Binary multiplication

We Need:

- An Adder
- A Shifter

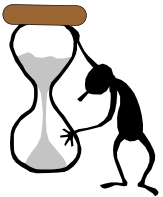


We'll ... design Adders ...

END-1

Please ...

*Read the next lecture,
about complements,
before you come to
class.*



END