

Addition

$$(+5) + (-3)$$

$$\text{Augend} \Rightarrow -5 \Rightarrow 10101$$

$$\text{Addend} \Rightarrow -3 \Rightarrow 10011$$

$$(-A) + (-B) \Rightarrow -(A+B)$$

$$\Rightarrow \begin{array}{r} 011 \\ 011 \\ \hline 1000 \end{array}$$

Since the sign is -ve (1 1000)

Booth Algorithm

$$(-5) \times (-3)$$

$$-5 \Rightarrow 1101 \text{ (BR)}$$

$$-3 \Rightarrow 1011 \text{ (QR)}$$

$$BR \Rightarrow 1101$$

$$\overline{BR} \Rightarrow 0010$$

$$\begin{array}{r} 1 \\ 0011 \end{array}$$

SC

4

AC	QR	Q _{n+1}	
0000	1011	0	
0011			
0011	1011	0	
0011			3
0110			
0001	1101	1	
0000	1110	1	3
1101			2
1101	1110	0	
1110	1111	0	
0011			1
0001	1111	0	
0000	1111	1	0

42

$$\frac{-5}{-3}$$

$$-5 \Rightarrow \begin{array}{c} -2 \quad 5 \\ | \quad | \quad 0 \quad | \\ - \end{array}$$

$$-3 \rightarrow 10(11)_3$$

$M \Rightarrow \underline{01011} \rightarrow \text{Form 16}$

$N_A \approx 10^{23}$

10101 accum
stop

	Accumulator	Divident (Q)	Divisor (M)
AXO	00000	1101	ϕ 0011 Count $\Rightarrow 4$
Shift left	00001	10 ϕ	Count $\Rightarrow 4-1 \Rightarrow 3$
	ϕ 0101		
A-M	ϕ 0110	101 ϕ	
ALO			
Shift left	01101	010 ϕ	Count $\Rightarrow 3-1 \Rightarrow 2$
	01011		
A+M	11000	010 ϕ	
ALO			
Shift left	10000	100 ϕ	Count $\Rightarrow 2-1 \Rightarrow 1$
	0 ϕ 0 ϕ 1		
A-X	ϕ ϕ 0 ϕ 1	100 ϕ	
Shift left	ϕ 0 ϕ 1 1	000 ϕ	Count $\Rightarrow 1-1 \Rightarrow 0$
	0 ϕ 0 ϕ 1		
ALO	000 ϕ 0	000 ϕ	

Count $\Rightarrow 2-1 \Rightarrow 1$

count $\Rightarrow 1-1 \Rightarrow 0$

Quotient $\Rightarrow 0001 \Rightarrow 1$

Remainder $\Rightarrow 00010 \Rightarrow 2$

$$\begin{array}{r} 1 \\ + 3 \overline{) 5} \\ \underline{3} \\ 2 \end{array}$$

3) Restoring ^{div} Algorithm

$$-5/-3$$

$$-5 \Rightarrow \underline{1}101$$

$$-3 \Rightarrow \underline{1}011$$

Accumulator	Divident (Q)	Divisor (H)
00000	1101	1011
00001	101	
10101		
<u>10110</u>	101	0
Restoring A 01011		
00001	101	0
00011	010	
10101		
<u>11000</u>	010	0
Restoring A 01011		
00110	100	
10101		
<u>01011</u>	100	0
Restoring A 00101		
01011	100	0
Restoring A 00100		

$$H = 01011$$

$$V_b = 10100$$

$$10101$$

$$\text{count} \Rightarrow 4$$

$$\text{count} \Rightarrow 4-1 \Rightarrow 3$$

$$\text{count} \Rightarrow 3-1 \Rightarrow 2$$

$$\text{count} \Rightarrow 2-1 \Rightarrow 1$$

My PC specification:

Model - HP 14ce1001tx
Processor - Intel® core(tm) i5-8265u cpu @ 2.30ghz (14 nm Processor)
Ram - 16 gb (LPDDR 4 2133)
internal - 500 gb → ssd
1 tb → hdd
Type: - 64-bit processor
OS - Windows 10 Home Single Language



Intel® Core™ i5-8265U Processor

6M Cache, up to 3.90 GHz

Performance Specifications

Number of Cores	4
Number of Threads	8
Processor Base Frequency	1.60 GHz
Max Turbo Frequency	3.90 GHz
Cache	6 MB Intel® Smart Cache
Bus Speed	4 GT/s
Intel® Turbo Boost Technology 2.0 Frequency‡	3.90 GHz
TDP	15 W
Configurable TDP-up Frequency	1.80 GHz
Configurable TDP-up	25 W
Configurable TDP-down Frequency	800 MHz
Configurable TDP-down	10 W

Memory Specifications

Max Memory Size (dependent on memory type)	64 GB
Memory Types	DDR4-2400, LPDDR3-2133
Max # of Memory Channels	2
Max Memory Bandwidth	37.5 GB/s
ECC Memory Supported ‡	No

CPU Utilization

Utilization	Speed	Base speed:	1.80 GHz
8%	1.78 GHz	Sockets:	1
		Cores:	4
Processes	Threads	Handles	Logical processors: 8
259	3229	117118	Virtualization: Enabled
Up time		L1 cache:	256 KB
1:09:08:58		L2 cache:	1.0 MB
		L3 cache:	6.0 MB

Memory Specification

In use (Compressed)	Available	Speed:	2133 MHz
7.2 GB (192 MB)	8.5 GB	Slots used:	1 of 1
		Form factor:	SODIMM
Committed	Cached	Hardware reserved:	113 MB
10.0/18.3 GB	6.1 GB		
Paged pool	Non-paged pool		
599 MB	847 MB		

SSD Specification

Active time	Average response time	Capacity:	466 GB
1%	3.4 ms	Formatted:	466 GB
Read speed	Write speed	System disk:	Yes
0 KB/s	104 KB/s	Page file:	No
		Type:	SSD

HDD Specification

Active time	Average response time	Capacity:	932 GB
4%	3.0 ms	Formatted:	932 GB
Read speed	Write speed	System disk:	No
115 KB/s	16.4 KB/s	Page file:	No
		Type:	HDD

b) Instruction

- 1) My Processor is compatible with 64 bit instruction set.
 - 2) An instruction set refers to the basic set of commands and instructions that a microprocessor understands and can carry out.
 - 3) My processor uses x86_64-bit architecture.
- Basically my processor supports following types of instructions (x86_instruction):
- 4) Data Transfer Instructions
 - 5) Arithmetic Instructions
 - 6) Bit Manipulation Instructions
 - 7) Program Execution Transfer Instructions (Branch & Loop Instructions)
 - 8) Processor Control Instructions
 - 9) Iteration Control Instructions
 - 10) Interrupt Instructions
 - 11) Instruction Set Extensions are additional instructions which can increase performance when the same operations are performed on multiple data objects. These can include SSE (Streaming SIMD Extensions) and AVX (Advanced Vector Extensions).
 - 12) String Instructions

c).Arithmetic Instructions (of x86 instruction set)

These instructions are used to perform arithmetic operations like addition, subtraction, multiplication, division, etc.

Following is the list of instructions under this group –

Instructions to perform addition

- 1)ADD – Used to add the provided byte to byte/word to word.
- 2)ADC – Used to add with carry.
- 3)INC – Used to increment the provided byte/word by 1.
- 4)AAA – Used to adjust ASCII after addition.
- 5)DAA – Used to adjust the decimal after the addition/subtraction operation.

Instructions to perform subtraction

- 6) SUB – Used to subtract the byte from byte/word from word.
- 7) SBB – Used to perform subtraction with borrow.
- 8) DEC – Used to decrement the provided byte/word by 1.
- 9) NPG – Used to negate each bit of the provided byte/word and add 1/2's complement.
- 10) CMP – Used to compare 2 provided byte/word.
- 11) AAS – Used to adjust ASCII codes after subtraction.
- 12) DAS – Used to adjust decimal after subtraction.

Instruction to perform multiplication

- 13) MUL – Used to multiply unsigned byte by byte/word by word.
- 14) IMUL – Used to multiply signed byte by byte/word by word.
- 15) AAM – Used to adjust ASCII codes after multiplication.

Instructions to perform division

- 16) DIV – Used to divide the unsigned word by byte or unsigned double word by word.
- 17) IDIV – Used to divide the signed word by byte or signed double word by word.
- 18) AAD – Used to adjust ASCII codes after division.
- 19) CBW – Used to fill the upper byte of the word with the copies of sign bit of the lower byte.
- 20) CWD – Used to fill the upper word of the double word with the sign bit of the lower