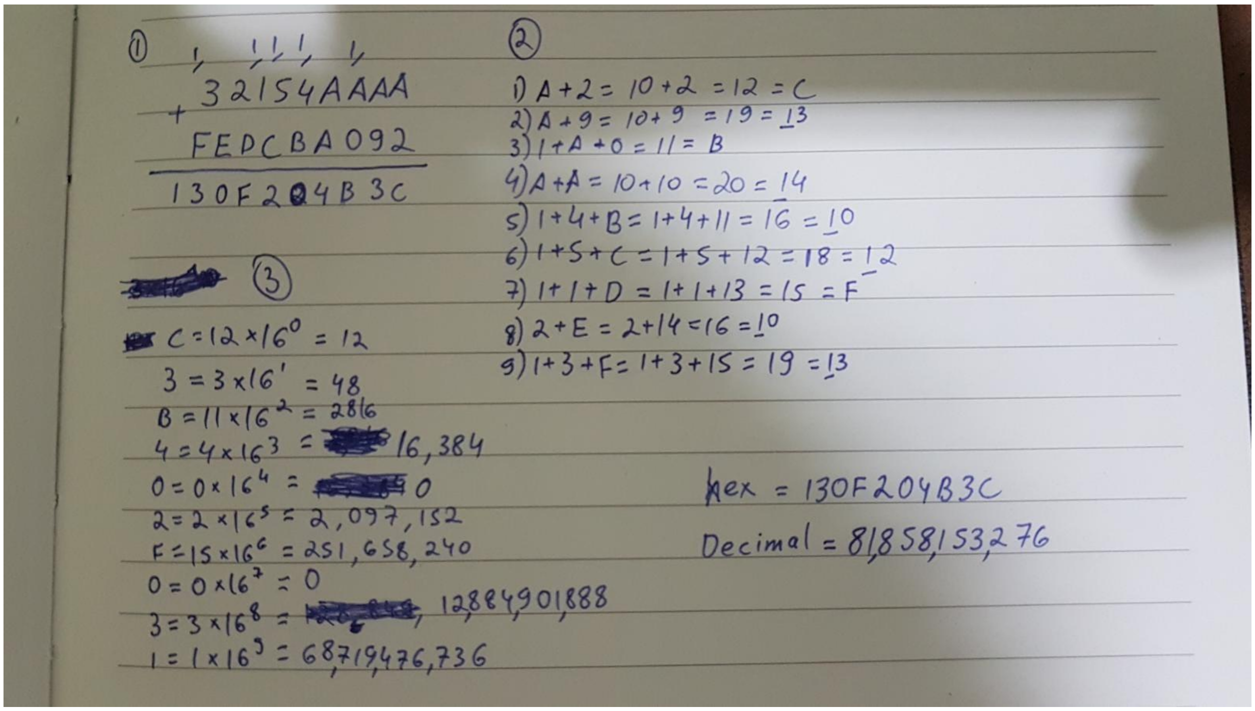
Computer Architecture - Assignment 2

Part 1

1.
Hex: 130F204B3C
Decimal: 81,858,153,276



2.
Use division by 16 repeatedly and note down the remainders
4048891811/16, quotient 253055738, remainder 3.
253055738/16, quotient 15815983, remainder 10 = A.
15815983/16, quotient 988498, remainder is 15 = F.
988498/16, quotient 61781, remainder is 2.
61781/16, quotient 3861, remainder is 5.
3861/16, quotient 241, remainder is 5.
241/16, quotient 15, remainder is 1.
15 = F

Hex: F1552FA3

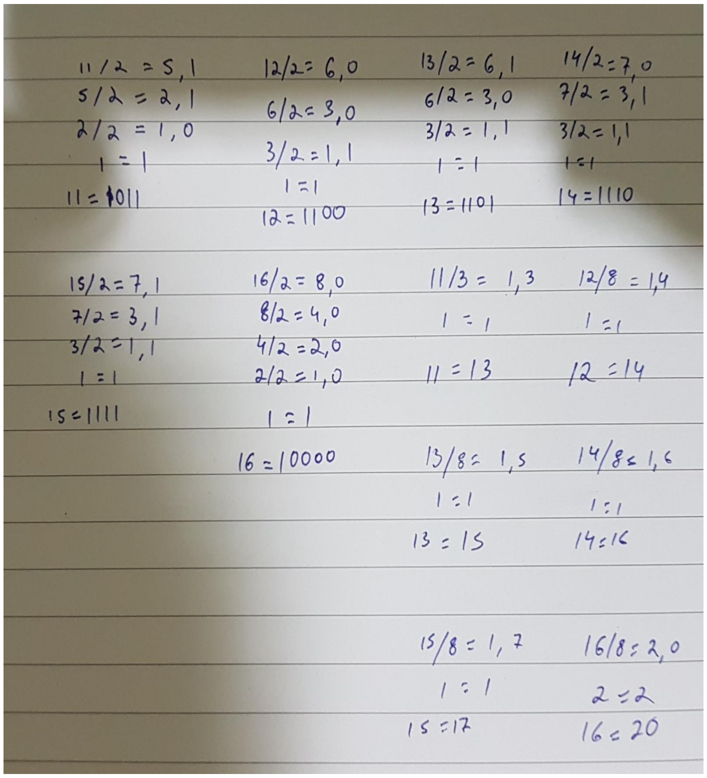
3.

2114112 = (2 × 8^6) + (1 × 8^5) + (1 × 8^4) + (4 × 8^3) + (1 × 8^2) + (1 × 8^1) + (2 × 8^0)

524288 + 32768 + 4096 + 2048 + 64 + 8 + 2 = 563,274

4.

Binary	Octal	Decimal	Hexadecimal
000	0	0	0
001	1	1	1
010	2	2	2
011	3	3	3
100	4	4	4
101	5	5	5
110	6	6	6
111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F
10000	20	16	10



Part 2

NAND Gate:
Boolean Expression: X = ~(A & B)
Truth Table:

A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

NOR Gate:
Boolean Expression: X = ~(A | B)
Truth Table:

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

XOR Gate:
Boolean Expression: X = A ⊕ B (XOR is represented by ⊕ symbol)
Truth Table:

A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

NOT Gate:
Boolean Expression: X = ~A
Truth Table:

A	X
0	1
1	0

3-input AND Gate:
Boolean Expression: X = A & B & C
Truth Table:

A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Part 3. The Significance of Studying Fixed-Point Number Manipulation

The study of manipulating fixed-point numbers holds substantial importance in the fields of computer science and engineering. Fixed-point arithmetic enables precise numerical computations in systems with limited resources or lacking floating-point hardware support. This article explores the significance of studying fixed-point number manipulation and provides examples of real-world systems where fixed-point arithmetic plays a vital role.

One key reason to study fixed-point number manipulation is its ability to enhance efficiency and optimization in resource-constrained environments. By utilizing fixed-point representation, complex floating-point operations can be simplified, leading to faster execution and reduced memory requirements. For instance, digital signal processing algorithms extensively utilize fixed-point arithmetic to improve performance. In audio and video processing, telecommunications, and real-time control systems, fixed-point arithmetic offers more efficient computations compared to floating-point alternatives.

Another important domain where the study of fixed-point manipulation is crucial is in embedded systems and low-power devices. These systems often operate under strict limitations of processing power, memory, and energy consumption. By implementing fixed-point arithmetic, efficient mathematical operations can be executed, resulting in cost-effective and energy-efficient solutions. Consider the example of microcontrollers, smartphones, and IoT devices. These devices rely on fixed-point arithmetic to perform calculations within the constraints of limited resources. By optimizing energy usage through fixed-point manipulation, battery-powered devices can operate for extended periods in remote or inaccessible locations.

Fixed-point arithmetic is essential for real-time systems and control applications that require precise and deterministic calculations. In these systems, fixed-point arithmetic offers predictable behavior, enabling timely responses and stability. Take the example of autonomous vehicles, industrial automation, and aerospace systems. Fixed-point arithmetic ensures accurate computations for safe and reliable operations. In control systems, where stability and performance are paramount, fixed-point manipulation plays a critical role. Research has shown that fixed-point arithmetic achieves high performance and stability in control applications, enabling precise control of complex systems.

The study of fixed-point number manipulation is vital for simulating and designing hardware systems. By accurately representing the behavior of digital systems using fixed-point arithmetic, designers can analyze and optimize system performance. This allows for identifying design flaws and enhancing system robustness before fabrication. Fixed-point modeling provides a closer approximation to the behavior of actual hardware systems, enabling efficient simulation and evaluation. For instance, in the design of integrated circuits, fixed-point arithmetic is used to evaluate system performance and ensure reliable operation.

In summery, the study of fixed-point number manipulation is essential due to its ability to enhance efficiency, optimize computations, and enable precise calculations in various real-world systems. From embedded systems to control applications, digital signal processing, and hardware design, fixed-point arithmetic plays a vital role, driving advancements and innovation across diverse domains of computing.