

Craig Shaffer

Dr. Swami Ponpondi

CPSC 300

2-1-2022

Homework 4

1. Convert (D1AF)₁₆ to binary and decimal. Convert (11011101001)₂ to decimal and hexadecimal. [10]

1 convert (D1AF)₁₆ to binary and decimal

D	1	A	F	D	1	A	F
1101	0001	1010	1111	$13 \cdot 16^3 + 1 \cdot 16^2 + 10 \cdot 16^1 + 15 \cdot 16^0 = 53679_{10}$			

$(D1AF)_{16} = (1101000110101111)_2$
 $(D1AF)_{16} = (53679)_{10}$

convert $(11011101001)_2$ to decimal & hexadecimal

11011101001	0110	1110	1001
$2^{10} \cdot 1 + 2^9 \cdot 1 + 2^8 \cdot 1 + 2^7 \cdot 1 + 2^6 \cdot 1 + 2^5 \cdot 1 + 2^4 \cdot 0 + 2^3 \cdot 1 + 2^2 \cdot 1 + 2^1 \cdot 0 + 2^0 \cdot 1 =$ $1024 + 512 + 256 + 128 + 64 + 32 + 8 + 1 = 1769_{10}$	6	E	9

$11011101001 = 1769_{10}$
 $11011101001 = 6E9_{16}$

2. Using 4-bit Twos complement, write the binary number for +5 and -4. [5]

2. Using 4-bit 2's complement write the binary number for +5 and -4.

7	0111	-1	1111
6	0110	-2	1110
5	0101	-3	1101
4	0100	-4	1100
3	0011	-5	1011
2	0010	-6	1010
1	0001	-7	1001
0	0000	-8	1000

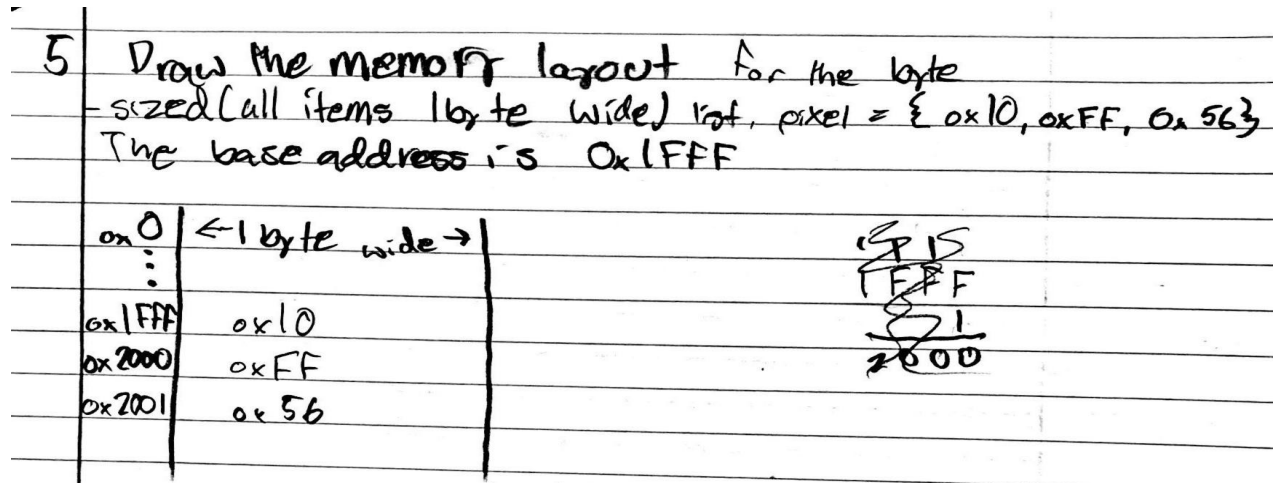
2's complement

$+5 \rightarrow 0101$
 $-4 \rightarrow 1100$

3. Write Python or Java code to convert a binary number to decimal and vice versa. [10]
 4. Write Python or Java code to convert hexa-decimal number to binary and vice versa. [10]

Both 3 and 4 are in the python file submitted

5. Draw the memory layout for the byte-sized (all elements are 1 byte wide) list, $pixel = \{0x10, 0xFF, 0x56\}$. The base address is $0x1FFF$. [10]



6. Draw the memory layout for the half-word-sized (all elements are 2 byte wide) list, $pixel = \{0x1000, 0x03, 0x56FF\}$. The base address is $0x1FF0$. [10]

