Chapter
2

DATA REPRESENTATION AND CONVERSION

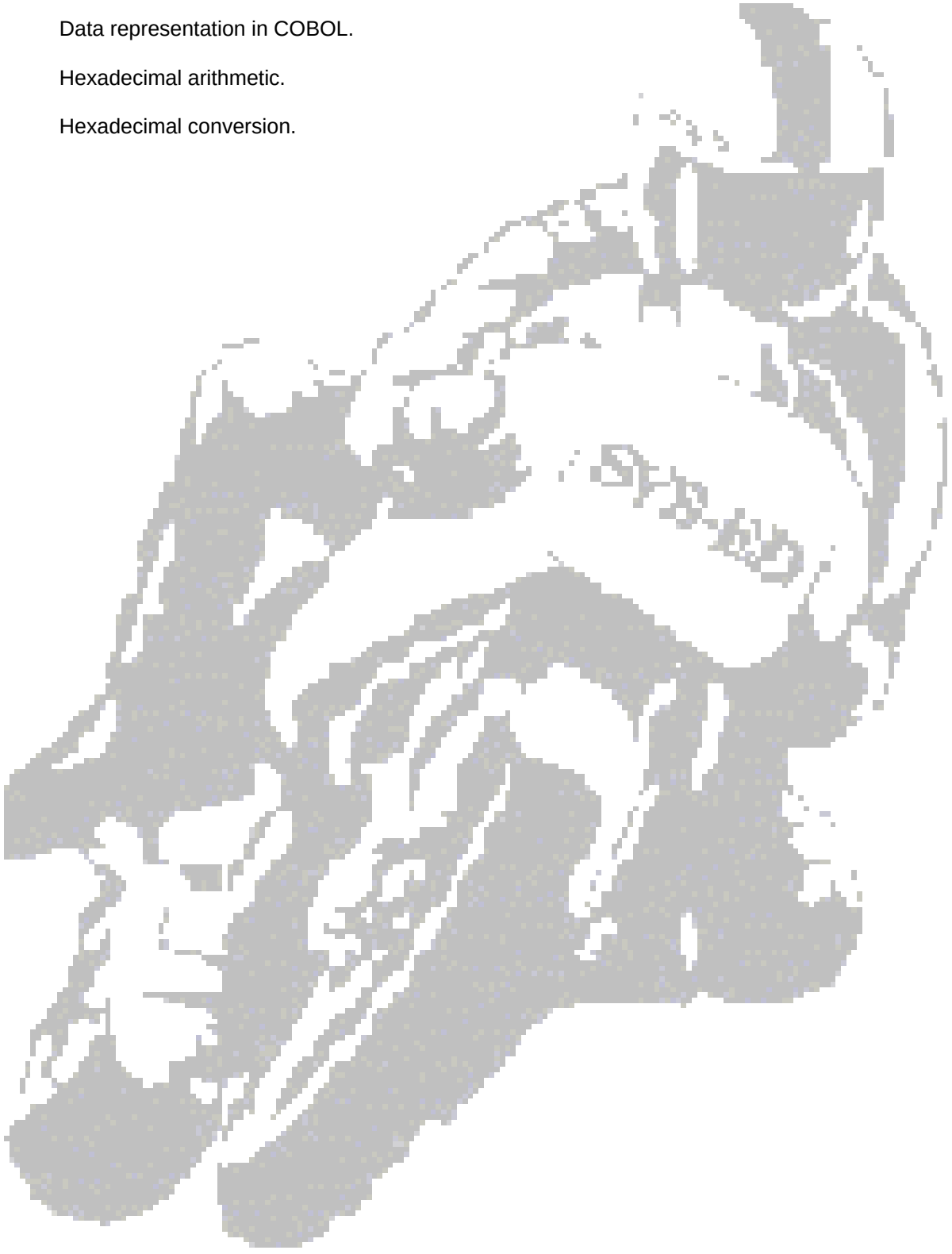
TM

SYS-ED/
COMPUTER
EDUCATION
TECHNIQUES, INC.

Objectives

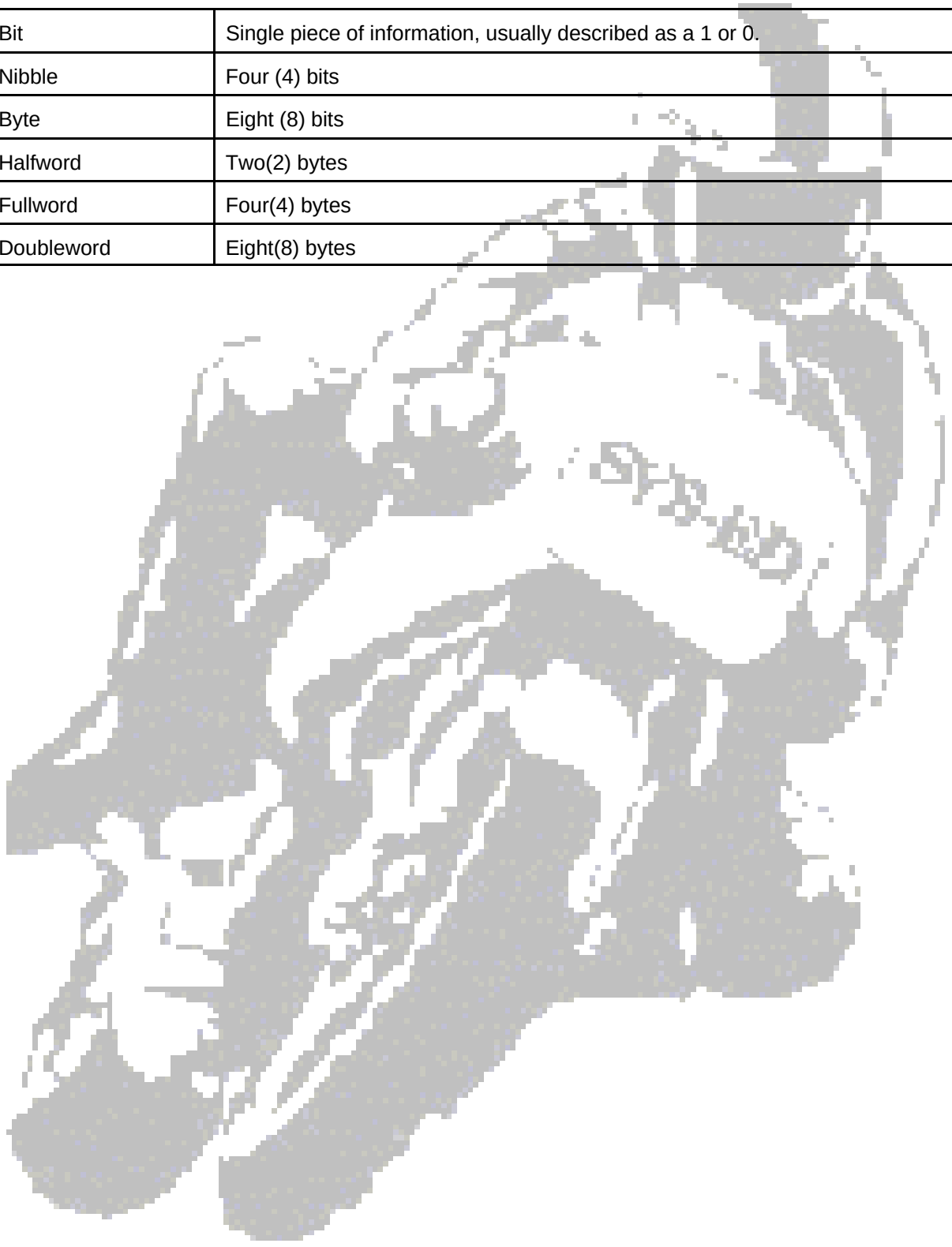
You will learn:

- Data representation in COBOL.
- Hexadecimal arithmetic.
- Hexadecimal conversion.



1 Data Representation

Bit	Single piece of information, usually described as a 1 or 0.
Nibble	Four (4) bits
Byte	Eight (8) bits
Halfword	Two(2) bytes
Fullword	Four(4) bytes
Doubleword	Eight(8) bytes



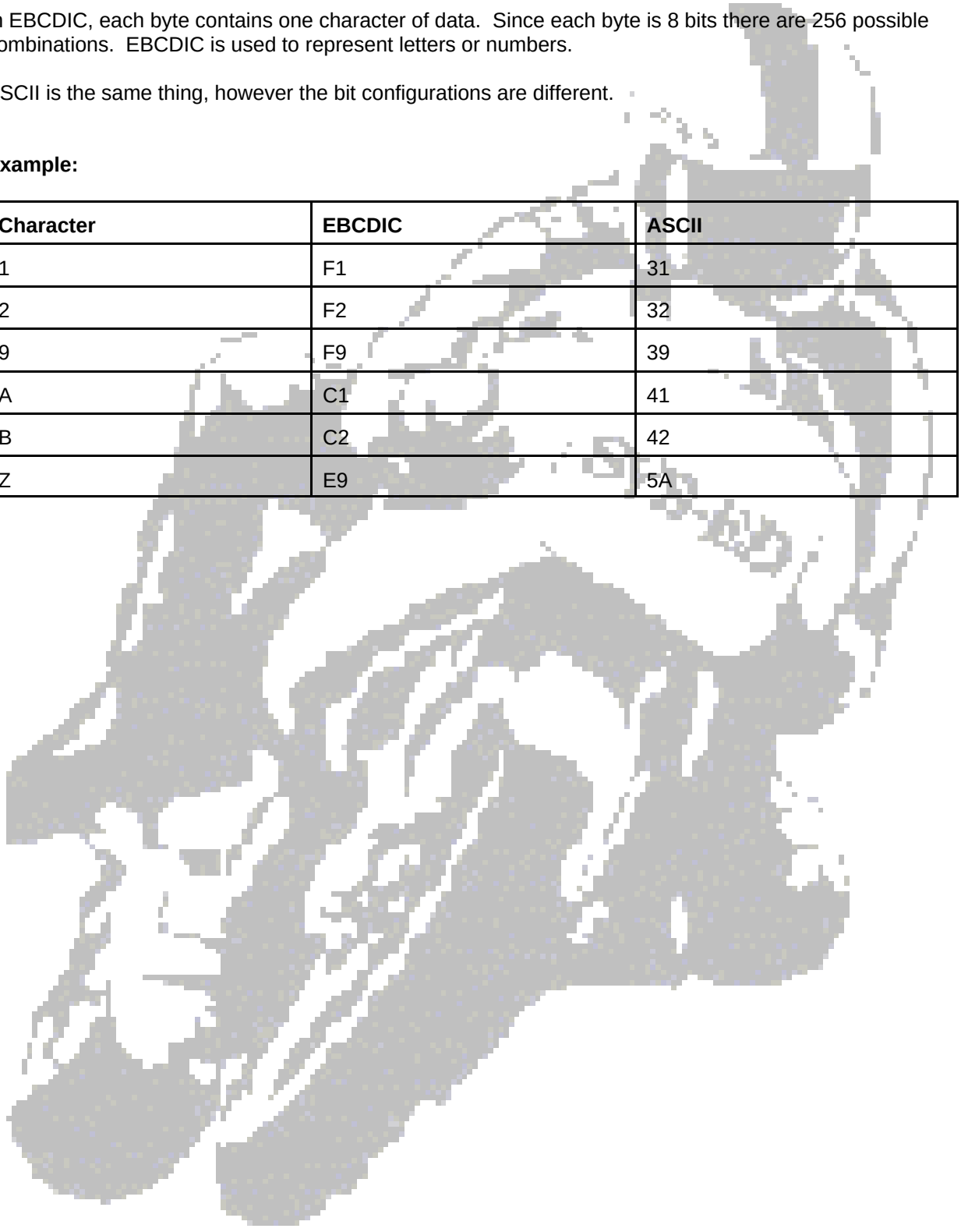
2 EBCDIC and ASCII

In EBCDIC, each byte contains one character of data. Since each byte is 8 bits there are 256 possible combinations. EBCDIC is used to represent letters or numbers.

ASCII is the same thing, however the bit configurations are different.

Example:

Character	EBCDIC	ASCII
1	F1	31
2	F2	32
9	F9	39
A	C1	41
B	C2	42
Z	E9	5A



3 Field Types

Zoned Numeric

Each byte contains a valid EBCDIC digit. This is called display numeric. However, the last byte contains a sign nibble (half byte). The valid signs are F for absolute, C for positive and D for negative.

75	F7F5
+354	F3F5C4
-354	F3F5D4

Fixed Point Binary

This type of field is called a COMP in COBOL. It takes up either 2 or 4 consecutive bytes, depending on the number of digits. The sign is indicated by the first bit in the byte, a 0 is a positive and 1 a negative.

A binary number can be converted to decimal by assigning a place value to each bit. The place value starts at the furthermost right bit position with a value of 1 and double for each position to the left. For example, the place values for a 32 bit number is 1, 2, 4, 8, 16, 32, 64...

Pack Numeric

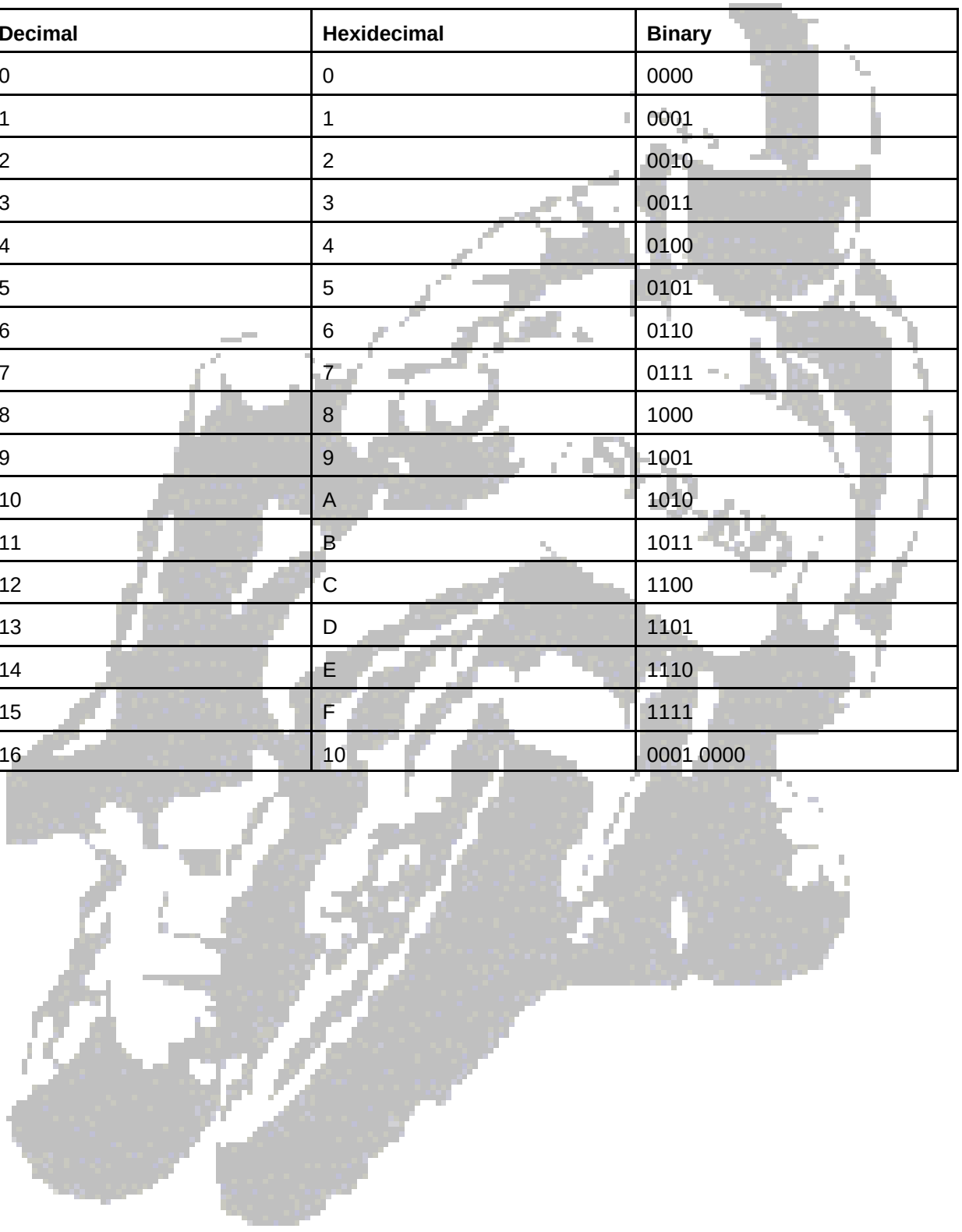
This type of field is called a COMP-3 in COBOL. It stores 2 digits in each byte, one nibble per digit. The exception is the sign nibble which is stored in the rightmost position. The valid sign nibbles are C, D and F.

Float

This type of field is called a COMP-1 or COMP-2 in COBOL. It stores the position of the decimal in the number itself, giving it the name floating point data item.

4 Hex Representation Arithmetic

Decimal	Hexidecimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111
16	10	0001 0000



4.1 Adding Hex Numbers - Desk Exercise

8 + 4 = _____
5 + 6 = _____
F + 1 = _____

1A02B8 6E88
+ 534 +1D3

4.2 Subtracting Hex Numbers - Desk Exercise

9 - 4 = _____
F - 1 = _____
B - A = _____
F - 8 = _____

1A0245 5C3C
- 34 -4D88

4.3 Converting Hex to Binary

Determine the bit configuration for each hex digit (nibble).

Example:

Hex A = 1010

Hex A1 = 1010 0001

Hex 1234 = 0001 0010 0011 0100