2.1 Bits & Groups of Bits

C/C++

0x prefix means the # is expressed in hexadecimal

10	11	12	13	14	15
A	B	C	D	E	F

Exercises

1. Binary to hex

$2^3\ 2^2\ 2^1\ 2^0$

8 4 2 1

a) $0100\ 0101\ 0110\ 0111_2 = 4567_{16}$

4 5 6 7

b) $1000\ 1001\ 1010\ 1011_2 = 89AB_{16}$

8 9 A B

c) $1111\ 1110\ 1101\ 1100_2 = FEDC_{16}$

F E D C

d) $0000\ 0010\ 0101\ 0010_2 = 0252_{16}$

0 2 5 2

3. How many bits are represented by the following?

a) $ffff\ ffff_{16}$ → 32 bits

each is 4 bits, 8 * 4 = 32

b) $7fff\ 5867\ def0_{16}$ = 48 bits

each is 4 bits, 12 * 4 = 48

c) 1111_2 = 4 bits

each is 1 bit, 1 * 4 = 4

d) 1111_{16} = 16 bits

each is 4 bits, 4 * 4 = 16

$$\begin{array}{r} 16 \\ 4\overline{)64} \\ \underline{4} \\ 24 \end{array} \qquad \begin{array}{r} 2 \\ 4\overline{)10} \\ \underline{8} \\ 2 \end{array}$$

---

2. Express the following bit patterns in binary

a) $83\overset{10}{a}\overset{15}{f}_{16}$

1000 0011 1010 1111_2

b) 9001_{16}

1001 0000 0000 0001_2

c) $aaaa_{16}$

1010 1010 1010 1010_2

d) 5555_{16}

0101 0101 0101 0101_2

4. How many hexadecimal digits are req'd to rep each of the following?

a) 8 bits → 2

b) 32 → 8

c) 64 bits → 16

d) 10 bits → 3

e) 20 bits → 5

f) 7 bits → 2

2.3 Mathematical Equivalence of Binary & Decimal

base / radix → r
digit → d
number place → n
or position

2nd Ex

1. What are the values of r, n, & each d_i for the following?

a) 29458254_{10}

$r = 10$, $n = 8$
$d_7 = 2$, $d_6 = 9$, $d_5 = 4$, $d_4 = 5$, $d_3 = 8$,

$d_2 = 2$, $d_1 = 5$, $d_0 = 4$

b) 29458254_{16}

$r = 16$, $n = 8$

$d_7 = 2$, $d_6 = 9$, $d_5 = 4$, $d_4 = 5$, $d_3 = 8$,

$d_2 = 2$, $d_1 = 5$, $d_0 = 4$

[illegible]
2. $10100101_2 = 128 + 32 + 4 + 1 = 165_{10}$

3. Convert the following 8-bit binary numbers to decimal by hand:

128 64 32 16 8 4 2 1

a) $1010\ 1010_2 = 128+32+8+2 = 170_{10}$

b) $0101\ 0101_2 = 64+16+4+1 = 85_{10}$

c) $1111\ 0000_2 = 128+64+32+16 = 240_{10}$

OR $255-15 = 240_{10}$:)

d) $0000\ 1111_2 = 15_{10}$

e) $1000\ 0000_2 = 128_{10}$

f) $0110\ 0011_2 = 64+32+2+1 = 99_{10}$

g) $0111\ 1011_2 = 255-128-4 = 123_{10}$!

h) $1111\ 1111_2 = 255_{10}$

4. Convert the following 16-bit binary #s by hand:

a) $1010\ 1011\ 1100\ 1101_2 = 2^{15}+2^{13}+2^{11}+2^{9}+2^{8}+2^{7}+2^{6}+2^{3}+2^{2}+2^{0}$

$= 43981_{10}$

b) $0001\,0010\ 0011\ 0100_2 = 2^{12}+2^{9}+2^{5}+2^{4}+2^{2} = 4660_{10}$

c) $1111\ 111\underline{0}\ 11\underline{0}1\ 11\underline{00}_2 = (2^{16}-1)-2^{8}-2^{5}-2^{1}-2^{0} = 65244_{10}$

d) $0000\ 0111\ 1101\ 1111_2 = (255-2^{5})+2^{8}+2^{9}+2^{10} = 2015_{10}$

e) $1000\ 0000\ 0000\ 0000_2 = 2^{15} = 32768_{10}$

*4 cont'd

f) $0000\ 0100\ 0000\ 0000_2 = 2^{10} = 1024_{10}$

g) $0111\ 1011\ 1010\ 1010_2 = (2^{16}-1) - 2^{15} - 2^{10} - 2^6 - 2^4 - 2^2 - 2^0$

$= 31658_{10}$

h) $0011\ 0000\ 0011\ 1001_2 = 2^{13} + 2^{12} + 2^5 + 2^4 + 2^3 + 2^0 = 12345_{10}$

5. Develop an algorithm to convert hexadecimal to decimal, then convert the following 16 bit #s to decimal by hand.

4096 ≠ 16!

a) $a000_{16} = 16^3 * 10 = 4096 * 10 = 40960_{10}$

b) $ffff_{16} = 16^4 - 1 = 65535_{10}$

c) $0400_{16} = 16^2 = 256 * 4 = 1024_{10}$

d) $1111_{16} = 16^3 + 16^2 + 16^1 + 16^0 = 4096 + 256 + 16 + 1 = 4369_{10}$

e) $8888_{16} = (16^3 * 8) + (16^2 * 8) + (16^1 * 8) + (16^0 * 8) =$

$8(16^3 + 16^2 + 16^1 + 16^0) = 34952_{10}$

f) $0190_{16} = 16^2 + (9 * 16^1) = 256 + 144 = 400_{10}$

g) $abcd_{16} = (16^3 * 10) + (16^2 * 11) + (16^1 * 12) + (16^0 * 13) = 43981_{10}$

h) $5555_{16} = 5 * (16^3 + 16^2 + 16^1 + 16^0) = 21845_{10}$

2.5 Unsigned Decimal to Binary Conversion

1. Convert to Binary

2.6 123_{10}

$$\begin{array}{ccccccc} 61 & 30 & 15 & 7 & 3 & 1 & 0 \\ 2\overline{)123} & 2\overline{)61} & 2\overline{)30} & 2\overline{)15} & 2\overline{)7} & 2\overline{)3} & 2\overline{)1} \\ 122 & 60 & 30 & 14 & 6 & 2 & 0 \\ \hline \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{1} & \textcircled{1} \end{array}$$

$\longleftarrow$

$= 1111011_2$

2. Convert the following unsigned decimal integers to 8-bit hexadecimal representation:

a) 100_{10}

$$\begin{array}{cccccccc} 50 & 25 & 12 & 6 & 3 & 1 & 0 & \\ 2\overline{)100} & 2\overline{)50} & 2\overline{)25} & 2\overline{)12} & 2\overline{)6} & 2\overline{)3} & 2\overline{)1} & \\ 100 & 50 & 24 & 12 & 6 & 2 & 0 & \\ \hline \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & 1 & 0 \end{array}$$

$\longleftarrow$

$= 0110\ 0100_2 = 64_{16}$

b) 125_{10}

$$\begin{array}{cccccccc} 62 & 31 & 15 & 7 & 3 & 1 & 0 & \\ 2\overline{)125} & 2\overline{)62} & 2\overline{)31} & 2\overline{)15} & 2\overline{)7} & 2\overline{)3} & 2\overline{)1} & \\ 124 & 62 & 30 & 14 & 6 & 2 & 0 & \\ \hline \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{1} & \textcircled{1} & \textcircled{1} & 0 \end{array}$$

$= 0111\ 1101_2 = 7d_{16}$

c) $10_{10} = 0A_{16}$

d) 88_{10}

$$\begin{array}{cccccccc} 44 & 22 & 11 & 5 & 2 & 1 & 0 & \\ 2\overline{)88} & 2\overline{)44} & 2\overline{)22} & 2\overline{)11} & 2\overline{)5} & 2\overline{)2} & 2\overline{)1} & \\ -88 & -44 & 22 & 10 & 4 & 2 & 0 & \\ \hline \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{1} & 0 \end{array}$$

$= 0101\ 0100_2 = 58_{16}$

$\nearrow 1111\ 1111_2$

---

e) $255_{10} = FF_{16}$

$\nearrow 0001\ 0000_2$

f) $16_{10} = 10_{16}$

g) $32_{10} = 0010\ 0000_2 = 20_{16}$

h) $128_{10} = 1000\ 0000_2 = 80_{16}$

3. Convert the following unsigned decimal integers to 16bit hexadecimal representation:

a) $1024_{10} = 0100\ 0000\ 0000_2 = 0400_{16}$

c.) $32768_{10} = 1000\ 0000\ 0000\ 0000_2 = 8000_{16}$

b.) 1000_{10}

$$\frac{500}{2\overline{)1000}} \quad \frac{250}{2\overline{)500}} \quad \frac{125}{2\overline{)250}} \quad \frac{62}{2\overline{)125}} \quad \frac{31}{2\overline{)62}} \quad \frac{15}{2\overline{)31}} \quad \frac{7}{2\overline{)15}} \quad \frac{3}{2\overline{)7}} \quad \frac{1}{2\overline{)3}} \quad \frac{0}{2\overline{)1}}$$

$$\frac{-1000}{0} \quad \frac{-500}{0} \quad \frac{-250}{0} \quad \frac{-124}{1} \quad \frac{-62}{0} \quad \frac{-30}{1} \quad \frac{-14}{1} \quad \frac{-6}{1} \quad \frac{-2}{1} \quad \frac{-0}{1}$$

$= 0000\ 0011\ 1110\ 1000_2$

$= 03E8_{16}$

d) $32767_{10} = 0111\ 1111\ 1111_2 = 0.7FF_{16}$

e) $256 = 0001\ 0000\ 0000_2 = 0100_{16}$

f) $65535_{10} = 1111\ 1111\ 1111\ 1111_2 = FFFF_{16}$

g) 4660_{10}

2330	1165	582	291	145	72	36
$2\overline{)4660}$	$2\overline{)2330}$	$2\overline{)1165}$	$2\overline{)582}$	$2\overline{)291}$	$2\overline{)145}$	$2\overline{)72}$
4660	2330	1164	582	290	144	72
⓪	⓪	①	⓪	①	⓪	⓪

18	9	4	2	1	0
$2\overline{)36}$	$2\overline{)18}$	$2\overline{)9}$	$2\overline{)4}$	$2\overline{)2}$	$2\overline{)1}$
36	18	8	4	2	0
①	⓪	①	⓪	⓪	①

$0001\ 0010\ 0011\ 0100_2 = 1234_{12}$

h) $43981_{10} = 1010\ 1011\ 1100\ 1101_2 = ABCD_{16}$

21990	10995	5497	2748	1374	687
$2\overline{)43981}$	$2\overline{)21990}$	$2\overline{)10995}$	$2\overline{)5497}$	$2\overline{)2748}$	$2\overline{)1374}$
43980	21990	10994	5496	2748	1374
①	⓪	①	①	⓪	⓪

343	171	85	42	21	10	5
$2\overline{)687}$	$2\overline{)343}$	$2\overline{)171}$	$2\overline{)85}$	$2\overline{)42}$	$2\overline{)21}$	$2\overline{)10}$
686	342	170	84	42	20	10
①	①	①	①	⓪	①	⓪

2	1	0
$2\overline{)5}$	$2\overline{)2}$	$2\overline{)1}$
4	2	0
①	⓪	①

4. To represent letter grades with +/-, you'd need 3 bits for letters A-F (6 value) and another bit for sign. 4 bits

2.8 Exercises

1. Say you want to allocate an area in memory for storing any # in the range 0- 4,000,000,000. This memory area will start at 2fffeb96. Give the addresses of each byte of memory that will ber required: ↘(8 bits)

We need 32 bits ($2^{32} = 4,294,967,296$) in order to store 4,000,000,001 distinct values. This is equivalent to 4 bytes. If the first address begins at 2fffeb96, the other three addresses are:

2fffeb97,
2fffeb98,
2fffeb99

2. You need to allocate an area in memory for storing an array of 30 bytes. The first byte will have the value 0 ... etc. The memory are will start at location 00100e. Show what the memory area looks like

Address	: value	address	: value	address	: value
00100e	00	001013	05	001019	0b
00100f	01	001014	06	00101a	0c
001010	02	001015	07	00101b	0d
001011	03	001016	08	00101c	0e
001012	04	001017	09	00101d	0f
		001018	0a	00101e	10

→ over

SARA KAZEMI

2. cont'd

address : value
00101f : 11
001020 : 12
001021 : 13
001022 : 14
001023 : 15
001024 : 16
001025 : 17
001026 : 18
001027 : 19
001028 : 1A
001029 : 1B
00102A : 1C
00102B : 1D.

3.

If the 16th address in hexadecimal is 0000 000f, the 17th is 0000 0010.