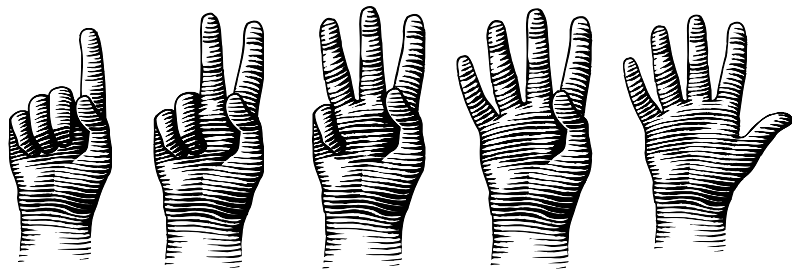
Binary Numbers – The Computer Number System

- Number systems are simply ways to count things. Ours is the base-10 or radix-10 system.
- Note that there is no symbol for "10" – or for the base of any system. We count 1,2,3,4,5,6,7,8,9, and then put a 0 in the first column and add a new left column, starting at 1 again. Then we count 1-9 in the first column again.
- Each column in our system stands for a power of 10 starting at 10^0.
  - Example: ⇒



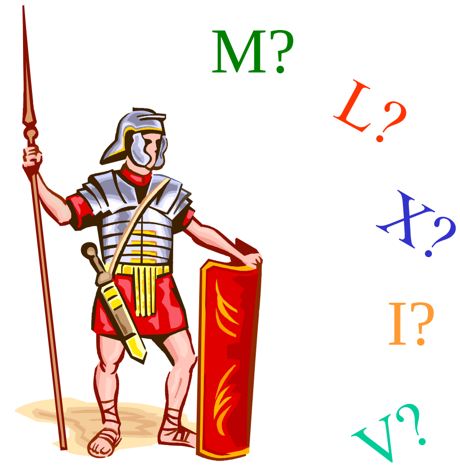
People use the base-10 system because we have 10 fingers!

1	3	5	7	8	9	6
10^6	10^5	10^4	10^3	10^2	10^1	10^0

1,357,896 = 1 x one million + 3 x one hundred thousand + 5 x ten thousand + 7 x one thousand + 8 x one hundred + 9 x ten + 6 x one.

Positional Notation – A History

- Heritage of western culture: The (difficult) Roman representation of numbers:
  - MCMXCVI = 1996, but MM = 2000!
  - (M = 1000, C = 100, X = 10, V = 5, I = 1)
  - VII = 7 (5+1+1), but XC = 90 (100 – 10), and (worst yet!) XLVII = 47 (50 – 10+5+1+1).
  - Want more? X • C = M, L/V=X. Ouch!
- A better idea -- positional notation:
  - Each digit in a column represents a multiplier of the power of the base (10) represented by that column.
  - The first column on the right is the zeroth power of 10. Succeeding columns to the left represent higher powers of 10.



Examples of positional notation:

$1996_{10} = 1 \times 10^3 + 9 \times 10^2 + 9 \times 10^1 + 6 \times 10^0$

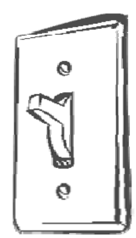
$2000 = 2 \times 10^3$

The Computer Number System

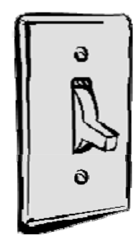
- All computers use the binary system :
  - Binary number system: Base = 2. Thus there are 2 numbers: 0 and 1.
  - A single binary number is called a Binary digIT, or bit.
- Computers perform operations on binary number groups called words.
- Today, most computers use 32-, 64-, or 128-bit words:
  - Words are subdivided into 8-bit groups called bytes.
  - One-half a byte is sometimes referred to as a nibble (a term not often used anymore).

Computer numbers are 1 and 0!

A simple electronic switch can represent both computer numbers



= 1



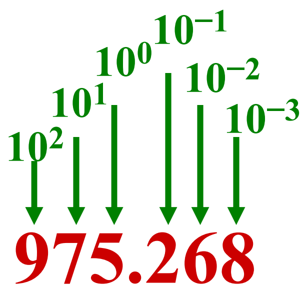
= 0

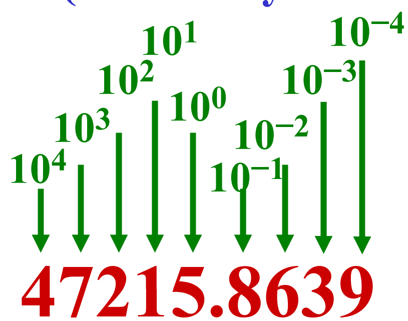
Binary Numeric Representation

- **A 32-bit binary number: 1101 0010 0101 0011 0101 1111 0001 1001**
  - **We will see ways to make this number more comprehensible below.**
- **We mentioned that in decimal notation:**
  - $1996_{10} = 1 \times 10^3 + 9 \times 10^2 + 9 \times 10^1 + 6 \times 10^0$, **and**
  - $2002 = 2 \times 10^3 + 0 \times 10^2 + 0 \times 10^1 + 2 \times 10^0$.
- **Consider the binary number** $255_{10} = 11111111_2$:
  - $255_{10} = 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
    $= 2^8 - 1$.
- **In the decimal system, each position from the right represents a larger power of ten, starting with** 10^0.
- **Likewise, in the binary number system, which is also positional, each position represents a larger power of two, starting with** 2^0.

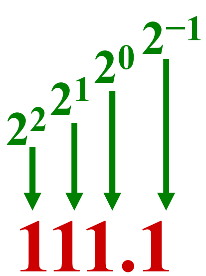
Reading Binary Numbers

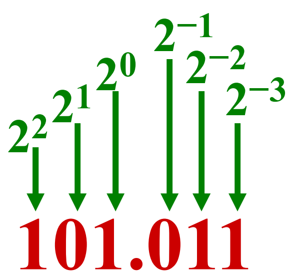
- In a decimal number, a non-0 digit in a column is treated as the multiplier of the power of 10 represented by that column (0's clearly add no value).

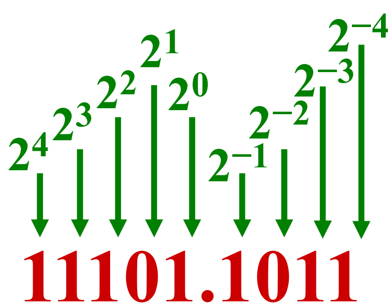




- We read binary numbers the same way; 0's count nothing and a 1 in any column means that the power of 2 represented by that column is part of the magnitude of the number. That is:







Binary Number Examples

- $11 = 1 \times 2^0 + 1 \times 2^1 = 3_{10}$
- $101 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 4 + 1 = 5_{10}$.
- $1001 = 1 \times 2^3 + 1 \times 2^0 = 8 + 1 = 9_{10}$.
- $1100 = 1 \times 2^3 + 1 \times 2^2 = 8 + 4 = 12_{10}$.
- $11101 = 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^0 = 16 + 8 + 4 + 1 = 29_{10}$.
- $0.1 = 1 \text{ X } 2^{-1} = ½ = 0.5_{10}$
- $0.111 = 1 \text{ X } 2^{-1} + 1 \text{ X } 2^{-2} + 1 \text{ X } 2^{-3} = 0.5 + 0.25 + 0.125 = 0.875_{10}$
- $0.10001 = 1 \text{ X } 2^{-1} + 1 \text{ X } 2^{-5} = 0.5 + 0.03125 = 0.53125_{10}$
- $1101.01 = 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^0 + 1 \times 2^{-2} = 8 + 4 + 1 + 0.25 = 13.25_{10}$
- $11.001 = 1 \times 2^1 + 1 \times 2^0 + 1 \text{ X } 2^{-3} = 2 + 1 + 0.125 = 3.125 \text{ 1 X } 2_{10}$
- $10.0011 = 1 \times 2^1 + 1 \text{ X } 2^{-3} + 1 \text{ X } 2^{-4} = 2 + 0.125 + 0.0625 = 2.1875_{10}$

Exercise #1

- **Convert the binary numbers to decimal:**
  - **1001001 --**
  - **0.011 --**
  - **10111.101 --**
  - **1111.11 --**

Easier Ways to Express Binary Numbers

- **Unfortunately, we were not born with 4 (or 8!) fingers per hand.**
- **The reason is that <u>it is relatively difficult to convert binary numbers to decimal, and vice-versa</u>.**
- **However, converting hexadecimal (base-16) numbers back and forth to binary is very easy (the octal, or base-8, number system was also used at one time).**
- **Since 16 = 2^4, it is very easy to convert a binary number of any length into hexadecimal form, and vice-versa:**

$0_{16} = 0_{10} = 0000_2$	$4_{16} = 4_{10} = 0100_2$	$8_{16} = 8_{10} = 1000_2$	$C_{16} = 12_{10} = 1100_2$
$1_{16} = 1_{10} = 0001_2$	$5_{16} = 5_{10} = 0101_2$	$9_{16} = 9_{10} = 1001_2$	$D_{16} = 13_{10} = 1101_2$
$2_{16} = 2_{10} = 0010_2$	$6_{16} = 6_{10} = 0110_2$	$A_{16} = 10_{10} = 1010_2$	$E_{16} = 14_{10} = 1110_2$
$3_{16} = 3_{10} = 0011_2$	$7_{16} = 7_{10} = 0111_2$	$B_{16} = 11_{10} = 1011_2$	$F_{16} = 15_{10} = 1111_2$

- **The letters that stand for hexadecimal numbers above 9 can be upper or lower case – both are used. Note that one nibble = one hex digit.**

Binary-Hexadecimal

- **Since $2^4 = 16$, each hex digit effectively represents the same numeric count as four binary digits.**
- **Another way to say this is that one column in a hex number is the same as four columns of a binary number.**

16^2 16^1 16^0 16^{-1} 16^{-2}

$$100101011011.01111010 = 0x\ 95B.7A^*$$

2^{11} 2^{10} 2^9 2^8 2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0 2^{-1} 2^{-2} 2^{-3} 2^{-4} 2^{-5} 2^{-6} 2^{-7} 2^{-8}

***Note: The "0x" prefix before a number signifies "hexadecimal."**

Hexadecimal-Binary Conversion

- **Most computers process 32 or 64 bits at a time.**
  - **In a 32-bit computer such as we will study, each data element in the computer memory (or "word") is 32 bits.**
  - **Example: 01111000101001011010111110111110**
  - **Separate into 4-bit groups, starting at the right:**

    0111 1000 1010 0101 1010 1111 1011 1110

  - **Converting:** $0111_2=7_{16}$, $1000_2=8_{16}$, $1010_2=A_{16}$, $0101_2=5_{16}$, $1010_2=A_{16}$, $1111_2=F_{16}$, $1011_2=B_{16}$, $1110_2=E_{16}$
  - **Or, 01111000101001011010111110111110 = 0x 78A5AFBE**
- **Another example:**
  - **Grouping:** 1001011100.11110011_2 = 10 0101 1100 . 1111 0011

    = (00)10 0101 1100 . 1111 0011

    = 2 5 C . F 3 = 0x 25C.F3

Binary-Hex and Hex-Binary Examples

- More binary-hex conversions*:
  - 101110100010 = 1011 1010 0010 = 0x BA2.
  - 101101110.01010011 = (000)1 0110 1110 . 0101 0011 = 0x 16E.53.
  - 1111111101.10000111 = (00)11 1111 1101 . 1000 0111 = 0x 3FD.87.
- To convert hex-binary, just go the other direction!
  - 0x 2375 = (00)10 0011 0111 0101 = 10001101110101.
  - 0x CD.89 = 1100 1101.1000 1001 = 11001101.10001001.
  - 0x 37AC.6 = (00)11 0111 1010 1100.011(0) = 11011110101100.011.
  - 0x 3.DCAB = (00)11.1101 1100 1010 1011 = 11.1101110010101011.

* Note that leading zeroes are added or removed as appropriate in the conversion processes.

Some Tricky Conversions

- Converting hex numbers to binary where leading or trailing zeros result is simple: Just drop the extra zeros! Examples:
  - 0x 7A.F8 = (~~0~~111)(1010).(1111)(1~~000~~) = 1111010.11111
  - 0x 29.3C = (~~00~~10)(1001).(0011)(11~~00~~) = 101001.001111
- The binary-hex conversion is a little trickier: Starting at the binary point, create group of 4 bits, then convert to hex (Go → for fractions, ← for integers). Add 0's to either end of the number to complete a group of four if necessary.
  - 1000.00111101 = (1000).(0011)(1101) = 0x 8.3D (No 0's needed)
  - 1111010.010111 = ([0]111)(1010).(0101)(11[00]) = 0x 7A.5C
  - 101110.01011 = ([00]10)(1110).(0101)(1[000]) = 0x 2E.58
- While leading zeroes are only moderately important (you could probably figure out the hex number without completing the group of 4), trailing zeroes are imperative. If a fractional binary number does not have 4 bits in its last group, zeroes must be added to complete the group or the hex number will not be correct! Trailing zeroes are especially important to the value of the fraction.

More Conversions With Fractions

- **Hex/binary conversion rule:**

  On either side of the hexadecimal point, convert each hex digit to an equivalent 4-bit binary number, and drop leading integral and trailing fractional zeroes.

  Examples:
  - **0x 79.EA = (~~0~~111)(1001).(1110)(101~~0~~) = 1111001.1110101**
  - **0x 2D.58 = (~~00~~10)(1101).(0101)(1~~000~~) = 101101.01011**

- **Binary/hex conversion rules:**

  (1) Group binary digits in sets of 4, starting at the binary point (go left for integers, go right for fractions), (2) Add leading zeroes to get a full group of four bits on the left, add trailing zeroes to get a last group on the right, if necessary (The zeroes on the right are very important!), (3) Convert: (4-bit binary number) → (hex digit).

  Examples:
  - **1100111.10101001 = ([0]110)(0111).(1010)(1001) = 0x 67.A9**
  - **101011. 0111101 = ([00]10)(1011).(0111)(101[0]) = 0x 2B.7A**

Exercise #2

- **Convert the following numbers:**
  - **1111000110.01011010 to hex:**
  - **0x 23d.7 to binary:**
  - **1101011.11111 to hex:**
  - **0x ed.8c to binary:**

Integer Binary/Decimal Conversions

- Sadly, we live in a decimal world, so we need to convert from binary to decimal and vice-versa.
- As we saw in earlier slides, for binary → decimal, "brute-force" is easiest: Ignore zeroes; for any 1 in a column i, add the decimal number represented by 2 to the i^{th} power. Thus:
- Number to be converted: 1 0 1 0 1 1 0 1
  - Power of 2 of columns: 7 6 5 4 3 2 1 0
  - Number represented: 128 64 32 16 8 4 2 1
  - Thus the decimal number is: 128 +0 +32 +0 +8 +4 +0 +1 = 173.
- Another example:
  - Number to be converted: 1 1 1 0 0 1 0 0
  - Power of 2: 7 6 5 4 3 2 1 0
  - Number represented: 128 64 32 16 8 4 2 1
  - Thus the decimal number is: 128+64+32 +0 +0 +4 +0 +0 = 228.

Fractional Binary/Decimal Conversions

- For numbers right of the binary point, we use the same approach, remembering that the powers of 2 are negative.

Number to be converted:	0 .	1	0	1	1
Power of 2:		-1	-2	-3	-4
Number represented:		.5	.25	.125	.0625

Thus the decimal number is: 0.5 + 0 + 0.125 + 0.0625 = 0.6875.

- A second example:

Number to be converted:	0 .	0	1	0	1	1
Power of 2:		-1	-2	-3	-4	-5
Number represented:		.5	.25	.125	.0625	.03125

Thus the decimal number is: 0 + 0.25 + 0 + 0.0625 + 0.03125 = 0.34375.

Converting Mixed Binary Numbers

- Binary numbers with binary point fractions are handled by combining the two techniques shown above:

Number to be converted:	1	1	0	.	1	0	1
Power of 2:	2	1	0		-1	-2	-3
Number represented:	4	2	1		.5	.25	.125

Thus the decimal number is: 4 + 2 + 0 + 0.5 + 0 + 0.125 = 6.625.

- Other examples:

1110.0111	=	8+4+2+0+0+0.25+0.125+0.0625 = 14.4375
10.10101	=	2+0+0.5+0+0.125+0+0.03125 = 2.65625
1111.101	=	8+4+2+1+0.5+0+0.125 = 15.625
100.001	=	4+0+0+0+0+0.125 = 4.125

Decimal → Binary Integer Conversions

- The easiest way to convert decimal-to-binary integers is the method of successive division.
- In this method, we simply divide the decimal number by 2:
  - The quotient becomes the new number to divide again.
  - The remainder, which will always be 1 or 0, becomes one bit of the binary number, least significant digit first. The last quotient will always be 0, the last remainder always 1.
  - Division continues until the quotient is 0 (with a last remainder of 1). Thus, converting 26 to binary:

$26 \div 2 = 13$, remainder 0

$13 \div 2 = 6$, remainder 1

$6 \div 2 = 3$, remainder 0

$3 \div 2 = 1$, remainder 1

$1 \div 2 = 0$, remainder 1

$1\,1\,0\,1\,0_2 = 26_{10}$

Another Decimal-to-Binary Example

- **Convert 117_{10} to binary:**

  117÷2=58, remainder 1

  58 ÷2=29, remainder 0

  29 ÷2=14, remainder 1

  14 ÷2=7, remainder 0

  7 ÷2=3, remainder 1

  3 ÷2=1, remainder 1

  1 ÷2=0, remainder 1

  Read this way (↑)

- **Starting from the bottom (MSB), $117_{10} = 1110101_2$.**
- **Check: $1110101_2 = 64+32+16+0+4+0+1 = 117_{10}$.**

Converting Decimal Fractions to Binary Fractions

- The method of successive multiplication: Decimal to binary fractions.
- Example – Convert 0.25 to binary:
  - 0.25 x 2 = 0.5 ; note 0 to left of binary point.
  - 0.5 x 2 = 1.0 ; note 1 to left of binary point.
  - There is no part of the decimal fraction left, thus we are done.
  - Read the binary fraction as the numbers to left of binary point in the two results, the remainder from the first multiplication first.
- Thus, $0.25_{10} = 0.01_2$.
- Decimal-fraction-to-binary-fraction conversion rules:
  - Multiply decimal fraction by 2. Record the number left of the binary point. This will always be a 1 or a 0.
  - Multiply the remaining fraction by 2, repeating above action.
  - Continue until the fraction is eliminated.

Converting Decimal ↔ Binary Fractions (2)

Additional examples:

0.375:

0.375 x 2 = 0.75
0.75 x 2 = 1.5
0.5 x 2 = 1.0

Thus $0.375_{10} = 0.011_2$.

0.427:

0.427 x 2 = 0.854
0.854 x 2 = 1.708
0.708 x 2 = 1.416
0.416 x 2 = 0.832
0.832 x 2 = 1.664
0.664 x 2 = 1.328
0.328 x 2 = 0.656
0.656 x 2 = 1.312

Then, $0.427_{10} \approx 0.01101101_2$. (Finite decimal fractions can result in repeating binary fractions!*)

* Thus we must change the last rule on the previous page to read: "Continue until the decimal fraction is eliminated or the binary fraction has at least twice as many places as the decimal fraction." This is because each column position in a binary fraction is much more significant than each column position in a decimal fraction.

Converting Mixed Decimal Numbers

- **Converting mixed decimal numbers means that we must perform two operations.**
- **For the integer part of the number, we do the method of successive division.**
- **For the decimal part, we do the method of successive multiplication.**
- **We recognize that for decimal numbers with a fraction part, we may not be able to convert the number exactly, since we could get a repeating fraction.**
- **In that case, we simply do successive multiplication enough times to get the accuracy of the binary fraction that we desire, at least twice the number of decimal places. Note that if the last of the 2x places is a 0, continue until you get a 1).**

Mixed Decimal Conversions -- Examples

- **36.125: 36/2=18, 0 rem; 18/2=9, 0 rem; 9/2=4, 1 rem; 4/2=2, 0 rem; 2/2=1, 0 rem, 1/2=0, 1 rem.**
  .125x2=0.25; .25x2=0.5; .5x2=1.0
  Then $36.125_{10} = 100100.001_2$.

Read ← for integers,
Read → for fractions.

- **19.375: 19/2=9, 1 rem; 9/2=4, 1 rem; 4/2=2, 0 rem; 2/2=1, 0 rem; 1/2=0, 1 rem.**
  .375x2=0.75; .75x2=1.5; .5x2=1.0.
  Then $19.375_{10} = 10011.011_2$.

- **7.33: 7/2=3, 1 rem; 3/2=1, 1 rem; 1/2=0, 1 rem.**
  .33x2=0.66; .66x2=1.32; .32x2=0.64; .64x2=1.28
  Then $7.33_{10} \approx 111.0101_2$.

- **10.17: 10/2=5, 0 rem; 5/2=2, 1 rem; 2/2=1, 0 rem; 1/2=0. 1 rem.**
  .17x2=0.34; .34x2=0.68; .68x2=1.36; .36x2=0.72; .72x2=1.44
  Then $10.17_{10} \approx 1010.00101_2$. *

***In the last example, we had to go to five places, since the fourth place was a 0.**

Exercise #3

- **Convert as directed:**
  - **237 to binary:**
  - **0.648 to binary:**
  - **48.125 to binary:**
  - **123.45**

Homework

- Anybody in here ever see the movie "50 First Dates?"
- Hopefully, you have listened carefully and perhaps even made a few notes today. Before bedtime tonight, consult your notes (if any), re-read the lecture on-line and make two lists: (1) things that you thought were important today, and (2) things you did not completely understand.
- Making list (1) reinforces your learning; list (2) gives you things to ask about when visiting me during office hours!
- Remember: those office hours are for YOUR benefit.

Hexadecimal-to Decimal Conversion

- Since hex numbers are used in computer displays, it is useful to convert decimal ↔ hex and back.
- For hex → decimal, we use the same brute-force method, as for binary-to-decimal conversion. Consider $3FB7_{16}$:

Number in hex:	3	F	B	7
Position as a power of 16:	3	2	1	0
Decimal value of 16^n:	4096	256	16	1
		"F"	"B"	

The decimal number is then 3(4096)+15(256)+11(16)+7(1) = 16,311.

Hexadecimal-to Decimal Conversion (2)

- **For fractions, the conversion is the same, remembering that hex digits to the right of the hexadecimal point are multipliers of negative powers of 16:**

$$0x\ 0.2A6 = (2/16)+(10/[16]^2)+(6/[16]^3)$$

$$= 0.125 + 0.03906 + 0.00146$$

$$\approx 0.1655$$

- **Mixed numbers are treated similarly:**

$$0x\ B7.CE = 11(16) + 7(1) + 12/16 + 14/256$$

$$= 183.80469$$

Integer Decimal-to-Hex Conversion

- **For converting decimal to hexadecimal integers, we use the method of successive division, as for decimal/binary conversions:**

  Convert 382_{10} to hex: We perform successive divisions by 16.

  382÷16=23, remainder 14 (= E)

  23 ÷16=1, remainder 7

  1 ÷16=0, remainder 1

  Read in reverse order as before.

- **In reverse order, the hexadecimal number is 17E, or $382_{10} = 17E_{16}$.**
- **Similarly, converting 651_{10}:**

  651 ÷16=40, remainder 11 (=B)

  40÷16=2, remainder 8

  2÷16=0, remainder 2

- **Thus, 651_{10}=$28B_{16}$.**

Note that as in binary conversion, the last quotient will always be 0. However, the last remainder may be anything from 1 to F.

Integer Decimal-to-Hex Conversion (2)

- Additional examples of integer conversion (read ← for the answer):
- 100 =? 100/16=6, 4 rem; 6/16=0, 6 rem; thus 100 = 0x 64.
- 4096 =? 4096/16=256, 0 rem; 256/16=16, 0 rem; 16/16=1, 0 rem; 1/16=0, 1 rem; thus 4096 = 0x 1000.
- 335 =? 335/16=20, 15 (= 0x F) rem; 20/16=1, 4 rem; 1/16=0, 1 rem; thus 335 = 0x 14F.
- 23795 =? 23795/16=1487, 3 rem; 1487/16=92, 15 (= 0x F) rem; 92/16=5, 12 (= 0x C) rem; 5/16=0, 5 rem; thus 23795 = 0x 5CF3.
- 1024 =? 1024/16=64, 0 rem; 64/16=4, 0 rem; 4/16=0, 4 rem; thus 1024 = 0x 400.
- Note that remainders that are > 9 must be converted to the hex digits A-F to get the correct hexadecimal number.

Fraction Decimal-to-Hex Conversion

- **For decimal to hexadecimal fraction conversions, we use the method of successive multiplication, as we did for decimal to binary conversions: (we also read → for the answer)**
- **Convert 0.125 to base 16: 0.125x16=2.0; thus 0.125 = 0x 0.2.**
- **Convert 0.335: 0.335x16=5.36; .36x16=5.76; .76x16=12.16,**

  thus 0.335 ≈ 0x 0.55C.
- **As in the binary case, exact decimal fractions can give hexadecimal repeating fractions. However, since the base 16 is > 10, we do not have to carry out the fraction to so many places in this case – the same number of places as the decimal fraction is sufficient.**
- **Final example: 0.95 =? .95x16=15.2; .2x16=3.2;**

  thus 0.95 ≈ 0x 0.F3 (note that 0x 0.F3 = 0.9492).
- **Once again, fractional numbers shown as > 9 must be converted to the hex digits A-F (or a-f) to get the correct hexadecimal fraction.**

Mixed Number Decimal/Hex Conversion

- **For decimal numbers with integers and fractions, we use both techniques just as for decimal-binary conversion:**
- **Convert 254.76 to hexadecimal:**

  254/16=15, 14 remainder (= E); 15/16=0, 15 remainder (= F).

  .76x16=12.16; .16x16=2.56 (note that 12 = C).

  Thus, 254.76 ≈ 0x FE.C2.
- **Convert 58.642 to hexadecimal:**

  58/16=3, 10 remainder (= A); 3/16=0, 3 remainder.

  .642x16=10.272; .272x16=4.352; .352x16=5.632 (again, 10 = A).

  Thus, 58.642 ≈ 0x 3A.A45.

Exercise

Do the following conversions and check answers with those shown.

Convert to decimal :

0x 12.c	**--**	**18.75**
0x 3e.78	**--**	**62.46875**
0x bcd.4f	**--**	**3021.3086 (approx.)**

- **Convert to hex:**

87.5	**--**	**0x 57.8**
693.875	**--**	**0x 2b5.e**