5.1 Data Storage and 5.2 Data Representation-Reading

Notebook:	How Computers Work [CM1030]		
Created:	2019-10-09 10:09 AM	**Updated:**	2019-11-10 10:03 AM
Author:	SUKHJIT MANN		
Tags:	AND, ASCII, Bit, Byte, NOT, OR, RGB, UNICODE, XOR		

Cornell Notes	**Topic:** 5.1 Data Storage 5.2 Data Representation-Reading	Course: BSc Computer Science Class: How Computer Work [CM1030]-Reading Date: October 30, 2019

Essential Question:

What are different methods of data storage? Alongside this, what is a bit and/or bytes and how can we use these to represent our data?

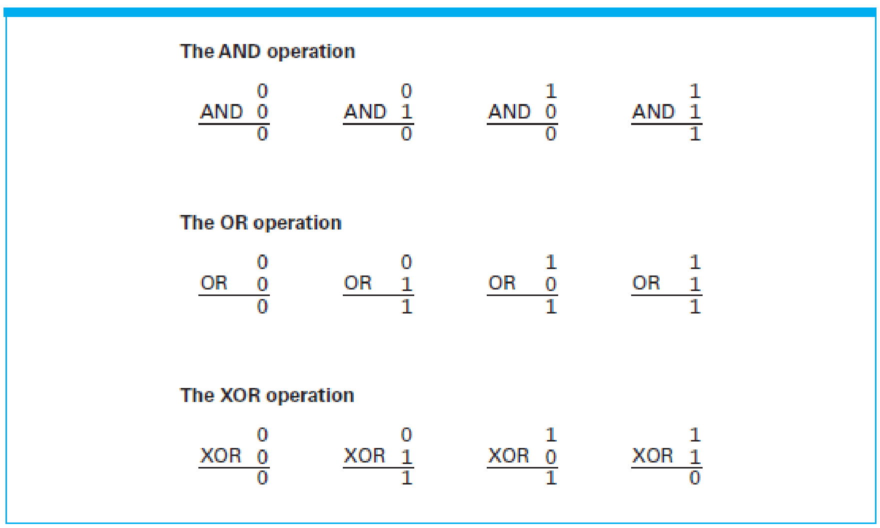
Questions/Cues:

- What are bits and bit patterns
- What are the Boolean operations on bits?
- What is a Gate?
- What is a flipflop circuit?
- What is Very Large-scale Integration?
- What is Hexadecimal Notation?
- What is Main Memory?
- What is an Address?
- What are the Read and Write Operations in Memory?
- What is DRAM and SDRAM?
- What is an Hard disk or Magnetic Disk?
- What are the measurements to evaluate a disk's performance?
- What is an CD?
- What is flash memory?
- What is a Solid State Disk?
- What is ASCII?
- What is Unicode?
- What is a text file?
- What is binary notation?
- What is a pixel and a bit map?
- What is RGB Encoding?

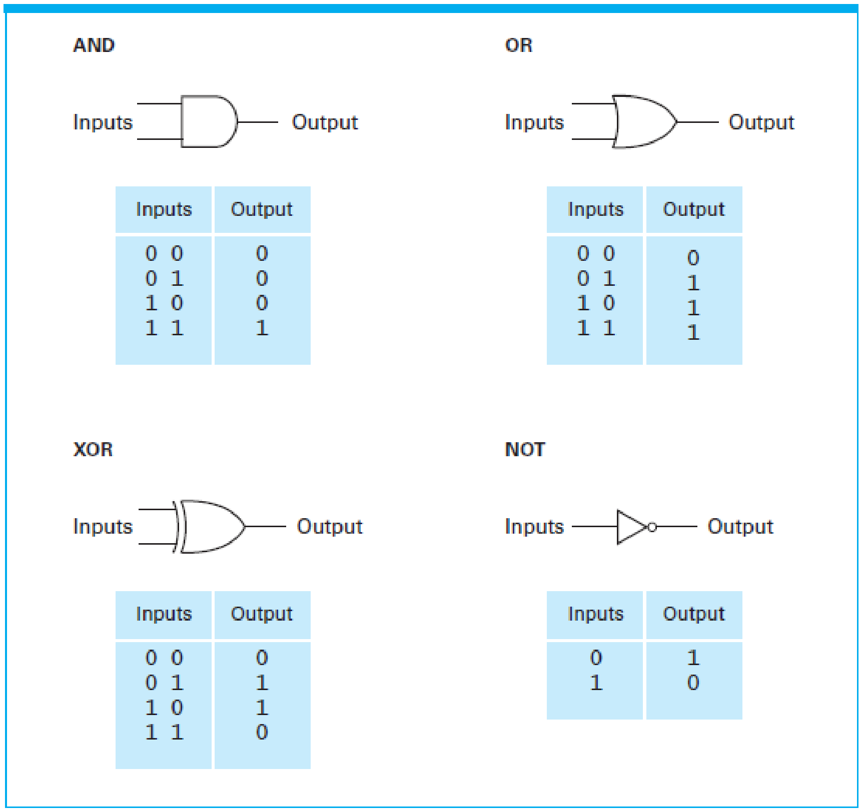
Notes

- Bits = binary digits, encode info inside comp as a patterns of 0's and 1's called bit patterns
    - Sometimes they represent numeric values, characters of alphabet, images or sounds

- Boolean operations = manipulate true/false values
  - bit 0 reps value false and bit 1 reps value true
  - AND reflects truth or falseness of compound statement formed by combining 2 smaller or simpler statement with conjunction and. Both components of the statement must be true for statement to be true ie. 1 and 1 = 1, conversely 1 and 0 = 0
  - OR is also again a compound statement of two smaller or simpler statement
    - OR is true when at least one component is true ie. 1 or 0 = 1
  - XOR is also a compound statement, produces output of 1(true) when one of its inputs is 1 and other is 0 (false)
    - "either P or Q but not both"
    - XOR operation produce output of 1 when inputs are different
  - NOT operation has only one input, output the opposite of the input



- Gate = device that produces output of Boolean operation when given operation's input values
  - constructed from variety of tech such as gears, relays, and optic devices
  - in today's comps usually implemented as small electronic circuits, where 1 and 0 rep'ed as voltage levels



AND

Inputs	Output
0 0	0
0 1	0
1 0	0
1 1	1

OR

Inputs	Output
0 0	0
0 1	1
1 0	1
1 1	1

XOR

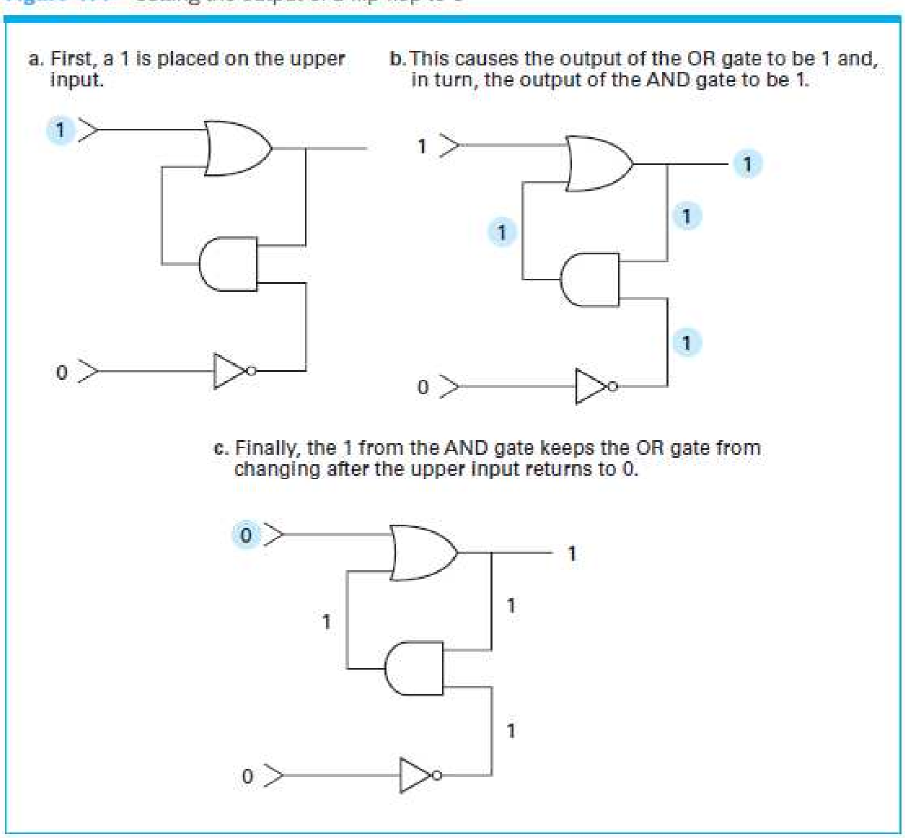
Inputs	Output
0 0	0
0 1	1
1 0	1
1 1	0

NOT

Inputs	Output
0	1
1	0

- Flip flop = fundamental unit of comp memory, circuit that produces output of 0 or 1 which remains constant until pulse (temp change to a 1 that returns to 0) from another circuit causes it to shift to the other value
  - the output can be set to "remember" a 0 or 1 under control of external stimuli

Figure 1.4 Setting the output of a flip-flop to 1



- Very large-scale integration = allows millions of electronic components to be constructed on wafer (chip), used to create mini devices containing millions of flip-

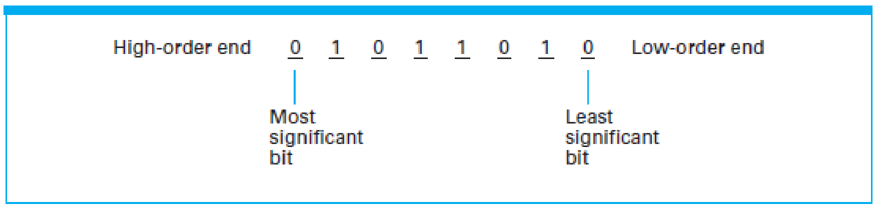
flops along with controlling circuitry
    - VLSI sometimes used to create entire comp system on single chip
- Stream = long string of bits
- Hexadecimal Notation = to simplify bit patterns, because patterns within machine have lengths in multiples of 4. Hexa Note uses single symbol to rep pattern of 4 bits

Figure 1.6 The hexadecimal encoding system

Bit pattern	Hexadecimal representation
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

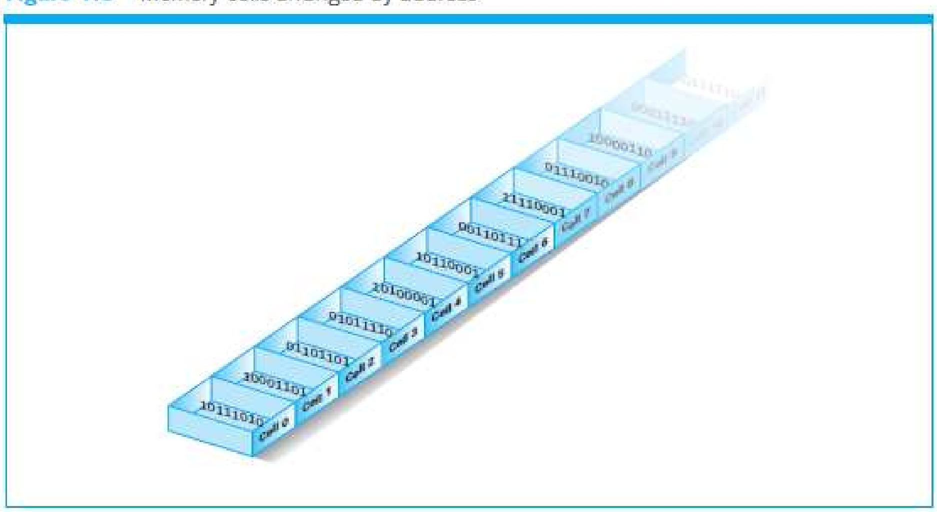
- Main memory = large collection of circuits (such as flip-flops) each capable of storing single bit comes together to form bit reservoir for purpose of storing data
    - organized in manageable units called cells, size of 8 bits or 1 byte
    - most significant or high-order bit determines numercial value, if this were a number

Figure 1.7 The organization of a byte-size memory cell



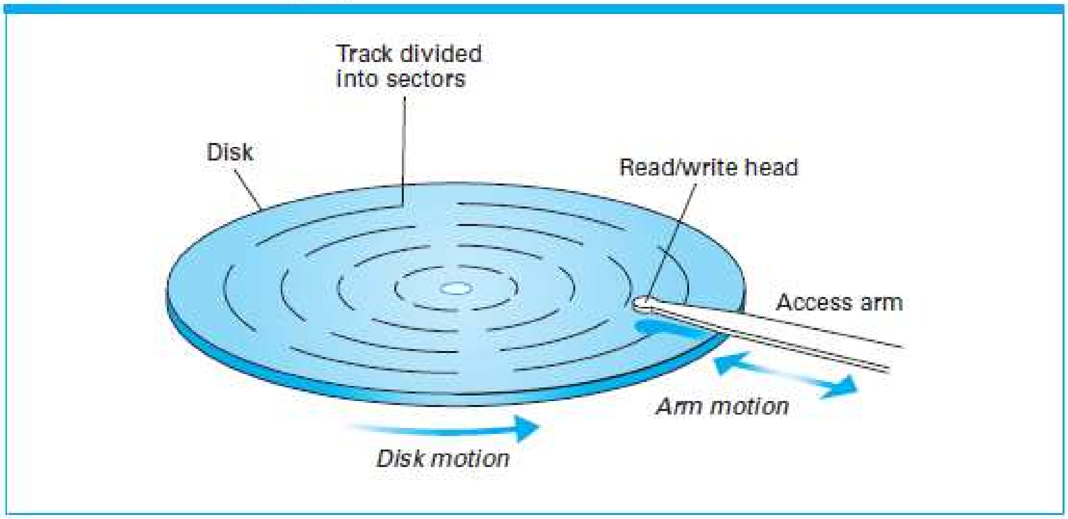
- Address = unique name to identify individual cells in comp's main memory
    - addresses entirely numeric

Figure 1.8 Memory cells arranged by address



- Read operation = circuitry that hold bits combined with other circuitry to allow other circuits to store and retrieve data, other circuits get data from memory by electronically asking for contents of certain address
- Write operation = To record info in memory by requesting that a certain bit pattern be placed in cell at particular address
- Because comp's main memory organized as individual, addressable cells accessed in any order, other name is RAM (Random Access Memory)
- DRAM(Dynamic RAM) = constructed from tech like refresh circuit which replenishes charges many times a sec because bits stored as tiny electric charges which dissipate quickly
- SDRAM(Synchronous DRAM) = applies additional techniques to decrease time need to retrieve contents from its memory cells
- Kilobyte(KB) = 1024 bytes
- International Standards Organized put out specialized terms for use for powers of two, kibi-, mebi-, gibi-, tebi-, but abandon by international community for kilo-, mega-, etc. which refer to powers of two in terms of comp measurements but refer to powers of 1000 when used in other contexts
- Advantage of mass storage over main memory include less volatility, large store capacity, low cost and the ability to remove storage medium from machine for archival purposes
- Magnetic & optical require mech motion so ergo, more time to store & retrieve data. Also more prone to mech failures than solid state systems
- HDD or magnetic disk = thin spinning disk with magnetic coating used to hold data, Read/write heads place above and/or below disk so as disk spins, each head transverses a circle called track
    - by re-positioning read/write heads different tracks accessed
    - disk system consists of several disk mounted on common spindle, 1 on top of other, read/write heads slip between platters and work in unions. Each time heads re-positioned, set of new tracks or cylinder becomes accessible
    - each track divided in small arks called sectors, info is recorded as continuous string of bits, all sectors contain same # of bits (512 bytes to few KB) and in simplest disk store system each track have the same # of sectors

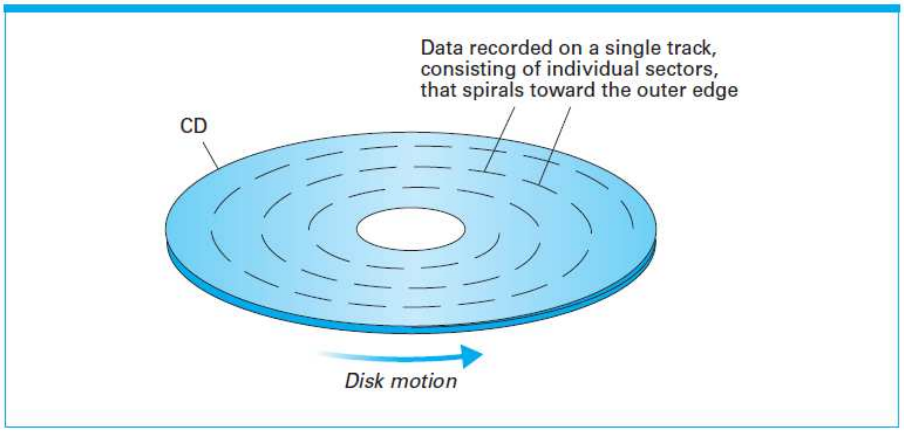
Figure 1.9 A disk storage system



    - zone-bit recording = several adjacent tracks collectively known as zones, with typical disk containing 10 zones. All tracks within zone have same # of sectors, but each zone has more sectors per track than the zone inside of it
    - capacity depends on # of platters used and density in which tracks and sectors are placed
- Measurements to evaluate disk system's performance:

- Seek time = time required to move read/write heads from 1 track to another
    - Rotation delay or Latency time = half the time required for disk to make complete rotation, which is average amount of time required for desired data to rotate around to read/write head once head positioned over desired track
    - Access time = sum of seek time & rotation delay
    - Transfer rate = rate at which data can be transferred to or from disk
        - Note** in case of zone-bit recording, amt of data passing read/write head in single disk rotation > for tracks in outer zone than inner zone; data transfer rate varies depending on portion of disk being used
        - measured in MB per sec
- CD (Compact Disk) = 12cm in diameter, info recorded on them by creating variation in their reflective surfaces, info retrieved by laser that detects irregularities on reflective surface as CD spins
    - originally to audio reccordings using CD-DA (compact disk-digital audio)
    - CD today for comp data store use same format, info is stored on single track that spirals around CD, track spirals from inside out, track divided into sectors each with identifying marks and capacity of 2KB of data

Figure 1.10 CD storage format



- Traditional CDs have capacities of 600-700MB, DVD(Digital Versatile Disks) have several GB, and BDs(Blu-ray Disks) have 5x over the DVD
- Flash memory = bits stored by sending electronic signals directly to storage medium where they cause electrons to be trapped in tiny chambers of silicon dioxide, thus altering characteristics of small electronic circuits. Since chambers are able to hold captive electrons for many yrs without external power; tech perfect for portable, nonvolatile data storage
- SSD(Solid-state disks) = larger flash memory devices, resilient to vibrations & physical shock, quiet operation (no moving parts) and lower access times
    - suffer from limited lifetime out of all flash tech
        - Wear-level techniques to alleviate this by relocating frequently altered data blocks to fresh locations on drive
    - Other flash tech include SD(Secure Digital) memory card, SDHC(High Capacity) memory cards and next-gen SDXC(Extended Capacity) memory cards
- ASCII (American Standard Code for Information Interchange) = uses bit patterns of length seven to rep upper & lowercase letters of the Eng alphabet, punctuation symbols, digits 0-9, and certain info like line feeds, carriage returns and tabs
    - ASCII extended to 8-bit-per-symbol format by adding a 0 at most significant end of each 7-bit pattern
        - this technique produces code in which each pattern into typical byte-sized memory cell. Also provides 128 additional bit patterns (obtained by assigning the extra bit value 1) that can be used to rep symbols beyond Eng Alphabet and associated punctuation

Symbol	ASCII	Hex	Symbol	ASCII	Hex	Symbol	ASCII	Hex
line feed	00001010	0A	>	00111110	3E	^	01011110	5E
carriage return	00001011	0B	?	00111111	3F	_	01011111	5F
space	00100000	20	@	01000000	40	`	01100000	60
!	00100001	21	A	01000001	41	a	01100001	61
"	00100010	22	B	01000010	42	b	01100010	62
#	00100011	23	C	01000011	43	c	01100011	63
$	00100100	24	D	01000100	44	d	01100100	64
%	00100101	25	E	01000101	45	e	01100101	65
&	00100110	26	F	01000110	46	f	01100110	66
'	00100111	27	G	01000111	47	g	01100111	67
(	00101000	28	H	01001000	48	h	01101000	68
)	00101001	29	I	01001001	49	i	01101001	69
*	00101010	2A	J	01001010	4A	j	01101010	6A
+	00101011	2B	K	01001011	4B	k	01101011	6B
,	00101100	2C	L	01001100	4C	l	01101100	6C
.	00101101	2D	M	01001101	4D	m	01101101	6D
/	00101110	2E	N	01001110	4E	n	01101110	6E
0	00101111	2F	O	01001111	4F	o	01101111	6F
1	00110000	30	P	01010000	50	p	01110000	70
2	00110001	31	Q	01010001	51	q	01110001	71
3	00110010	32	R	01010010	52	r	01110010	72
4	00110011	33	S	01010011	53	s	01110011	73
5	00110100	34	T	01010100	54	t	01110100	74
6	00110101	35	U	01010101	55	u	01110101	75
7	00110110	36	V	01010110	56	v	01110110	76
8	00110111	37	W	01010111	57	w	01110111	77
9	00111000	38	X	01011000	58	x	01111000	78
:	00111001	39	Y	01011001	59	y	01111001	79
;	00111010	3A	Z	01011010	5A	z	01111010	7A
<	00111011	3B	[	01011011	5B	{	01111011	7B
=	00111100	3C	\	01011100	5C	|	01111100	7C
	00111101	3D	]	01011101	5D	}	01111101	7D

- UNICODE = uses a unique pattern of up to 21 bits to rep each symbol
  - combined with Unicode Transformation Format 8-bit (UTF-8), traditional ASCII chars still rep'ed with 8 bits while thousands of additional chars form langs like Chinese, Japanese and Hebrew can be rep'ed by 16 bits
    - UTF-8 uses 24- or 32-bit patterns to rep more obscure Unicode symbols
- Text file = consisting of long sequence of symbols encoded using ASCII or Unicode
  - text file and file produced by word processor both consist of textual material, but txt file contains only char-by-char encoding of text, whereas file produced by word processor contains numerous proprietary codes rep'ing changes in fonts, align info, and other parameters
- Binary notation = way of rep'ing numeric values using only digits 0 & 1

Figure 1.13 The base 10 and binary systems

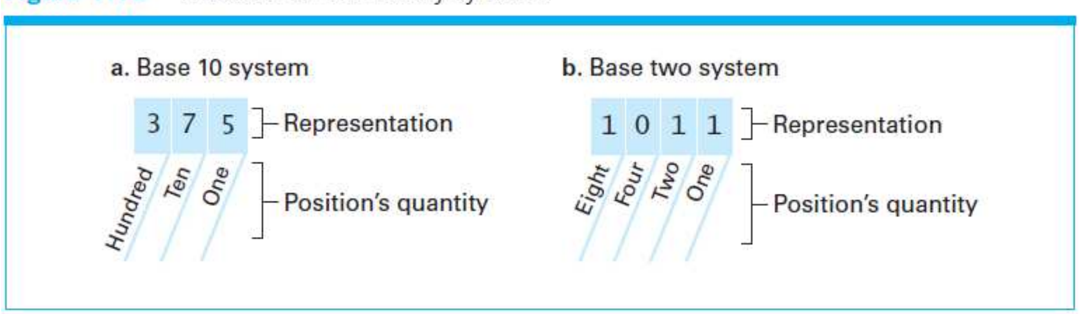


Figure 1.14 Decoding the binary representation 100101

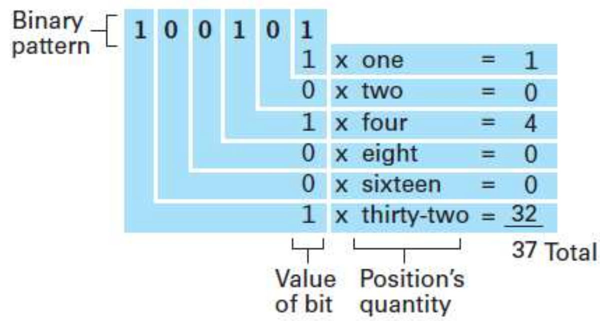


Figure 1.15 An algorithm for finding the binary representation of a positive integer

Step 1. Divide the value by two and record the remainder.

Step 2. As long as the quotient obtained is not zero, continue to divide the newest quotient by two and record the remainder.

Step 3. Now that a quotient of zero has been obtained, the binary representation of the original value consists of the remainders listed from right to left in the order they were recorded.

Figure 1.16 Applying the algorithm in Figure 1.15 to obtain the binary representation of thirteen

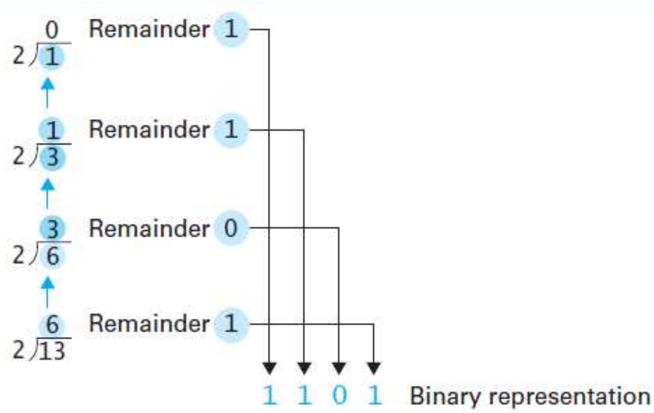
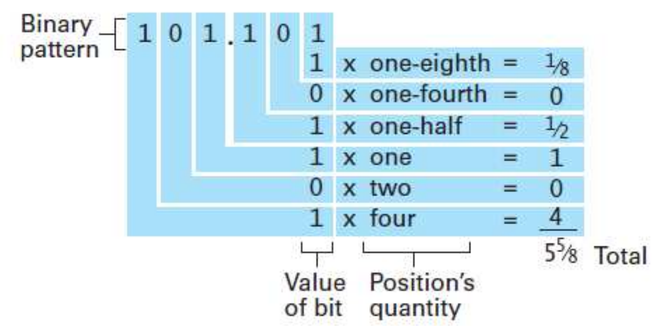


Figure 1.17 The binary addition facts

$$\begin{array}{r}0\\+0\\\hline 0\end{array}\qquad\begin{array}{r}1\\+0\\\hline 1\end{array}\qquad\begin{array}{r}0\\+1\\\hline 1\end{array}\qquad\begin{array}{r}1\\+1\\\hline 10\end{array}$$

Figure 1.18 Decoding the binary representation 101.101



- Pixel (picture element) = dot
- Bit map = appearance of each pixel is encoded and an entire image is rep as collection of the encoded pixels
    - image cannot be easily to any arbitrary size
- For eleborate B & W pics, each pixel can be rep'ed by collection of bits (usually 8) which allows variety of shades of grayness to be rep'ed
- RGB encoding = each pixel is rep'ed as 3 color components- red, green, and blue, 1 byte is normally used to rep intensity of each color component. So 3 bytes of storage required to rep single pixel in original image

Summary

In this week, we learned what is a bit is and bytes are. Also we looked at the different operations performed on bits. Alongside this, we looked at the binary representation of decimal numbers, letters and etc. Further, we touched on the ASCII and UNICODE standards in computing. Finally, we explored the various types of storage media