1.4 Number Representation

- Representation for Seventy-Five (75):
  - 75
  - 1001011
  - 4B
  - K

Which one is correct ?

1.4 Number Representation

- Representation for (+)Seventy-Five:
  - Integer: 75_{10}
  - Binary: 1001011_2
  - Hexadecimal: $4B_{16}$
  - ASCII: K

All Correct, just different format!!!

Binary Numbers

- **B** is an **n-bit binary number**
  - **B** = $b_{n-1} \ldots b_1\, b_0$ where $b_i = 0$ or 1, for $0 \le i \le n-1$
  - Example: 0110 (n=4)
    - $b_3 = 0$; $b_2 = 1$; $b_1 = 1$; $b_0 = 0$
  - Example: 0110 0111 (n=8)
    - $b_7 = 0$; $b_6 = 1$; $b_5 = 1$; $b_4 = 0$; $b_3 = 0$; $b_2 = 1$; $b_1 = 1$; $b_0 = 1$

Why b_{n-1} and not b_n ?

Computer terms always start from 0

b_0 always at the right end!

Binary to Integer Conversion

- Integer (**B**) = Integer of ($b_{n-1} \ldots b_1\ b_0$)
  - Integer-value = $b_{n-1} \times 2^{n-1} + \ldots + b_1 \times 2^1 + b_0 \times 2^0$
  - Examples: 4- and 8-bit
  - Integer (0110) = $0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 6$
  - Integer (01100111) = $0 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 +$
    $0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$

Integer to Binary Conversion

- Binary (**I**) = Binary of (Integer)
  - Keep dividing the integer by 2 until the remainder is 0 or 1
  - The remainders, in order, represent the binary number
  - Example: 5/2 = 2..Remainder:1

    2/2 = 1.. Remainder: 0

    Binary (5) = 101

    What is 10_{10} in binary ?
  - Example: 10/2 = 5.. Remainder: 0

    5/2 = 2.. Remainder: 1

    2/2 = 1.. Remainder: 0

    Most Significant Bit (MSB)

    Binary (10) = 1010 Least Significant Bit (LSB)

Binary-Integer Range

- An **n-bit** binary number can represent an unsigned integer in the **range** of {0 to 2^n–1}
  - I: {0 to 2^n–1} ; representing = {0,1,2..., 2^n–1}
  - Example: n=4; range = {0 to 2^4–1}

    actual numbers represented = {0,1,2..15}

    n=8; range = {0 to 2^8–1}

    actual numbers represented {0,1,2..255}

Binary Coded Decimal (BCD)

Integer	BCD
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Binary (1110) = ?

Integer (BCD) = Integer of ($b_{n-1} \ldots b_1 b_0$)

$= b_{n-1} X 2^{n-1} + \ldots + b_1 X 2^1 + b_0 X 2^0$

Integer (0011) = Integer of ($b_{n-1} \ldots b_1 b_0$)

$= 0 X 2^3 + 0 X 2^2 + 1 X 2^1 + 1 X 2^0 = 3$

BCD (7) :

= 7/2 = 3..remainder:1

= 3/2 = 1..remainder:1

= 111 or 0111

Hexadecimal Format

- Base 16 representation
- Represent Integer in the range = {0..15}

  ⇔ One hex digit:

  0..9ABCDEF

  or 0..9abcdef
- ⇔ 4-bit binary digits

- 16-bit binary number
- ⇔ Four 4-bit binary
- ⇔ Four hex-bits

Integer	Hex		Binary
0	0	0	0000
1	1	1	0001
2	2	2	0010
3	3	3	.
4	4	4	.
5	5	5	.
6	6	6	.
7	7	7	.
8	8	8	.
9	9	9	1001
10	A	a	1010
11	B	b	1011
12	C	c	1100
13	D	d	1101
14	E	e	1110
15	F	f	1111

Conversion: Hexadecimal

- Example: **0x**15ac to **binary**

What is 0x?

The number following 0x is hex format

- Example: 0001 1110 0000 0111 to **hex**

Why do we use hexadecimal ?

Verbal, Written and Visual Communication Succinctly!

1.5 Character Encoding

- American Standard Code for Information Interchange (ASCII)
  - Use 7-bit binary codes to represent alphabets, numbers, and special characters(/ ; \ $ etc.)
  - Examples:

character	code (7-bit bin)		code (hex)
A	1 0 0	0 0 0 1	41
7	0 1 1	0 1 1 1	37
+	0 1 0	1 0 1 1	2B

ASCII

Bit positions 3210	Bit positions 654: 000	001	010	011	100	101	110	111
0000	NUL	DLE	SPACE	0	@	P	´	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	”	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	’	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	/	l	|
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	ˆ	n	˜
1111	SI	US	/	?	O	—	o	DEL

ASCII (A) = ?

100 1011 = ASCII (?)

ASCII is often represented in 8 bits by inserting a 0 as the MSB

Control Characters

NUL	Null/Idle	SI	Shift in
SOH	Start of header	DLE	Data link escape
STX	Start of text	DC1-DC4	Device control
ETX	End of text	NAK	Negative acknowledgment
EOT	End of transmission	SYN	Synchronous idle
ENQ	Enquiry	ETB	End of transmitted block
ACK	Acknowledgment	CAN	Cancel (error in data)
BEL	Audible signal	EM	End of medium
BS	Back space	SUB	Special sequence
HT	Horizontal tab	ESC	Escape
LF	Line feed	FS	File separator
VT	Vertical tab	GS	Group separator
FF	Form feed	RS	Record separator
CR	Carriage return	US	Unit separator
SO	Shift out	DEL	Delete/Idle

Number Representation

- Exercise: convert among
  - Integer: 75_{10}
  - Binary: 1001011_2
  - Hexadecimal: $4B_{16}$
  - ASCII: K