Computer Architecture and operating systems

Theoretical background

Part III

Krystian Wojtkiewicz

Wrocław, Poland

https://i-cu.eu

Table of contents

Signs

Signs

Each sign is represented by a number pointing to an appropriate number in the code table.

Code tables

- ASCII – 128 positions, including small and upper case letters from Latin alphabet.
- extended 256-signs ASCII based codes
  - first 128 positions same as in original ASCII code, following 128 positions include national signs and other symbols
  - **problem** different codes for different countries
- EBCDIC family codes, used mainly by IBM
- UNICODE
  - at first 2^{16}, currently up to 2^{32} possible positions
  - represents all (?) signs used in the world

American Standard Code for Information Interchange

Developed from telegraphic code by the ASA (American Standard Association)

USASCII code chart

Bits b_7 b_6 b_5 →	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4 b_3 b_2 b_1 ↓ Row / Column →	0	1	2	3	4	5	6	7
0 0 0 0	0	NUL	DLE	SP	0	@	P	`	p
0 0 0 1	1	SOH	DC1	!	1	A	Q	a	q
0 0 1 0	2	STX	DC2	"	2	B	R	b	r
0 0 1 1	3	ETX	DC3	#	3	C	S	c	s
0 1 0 0	4	EOT	DC4	$	4	D	T	d	t
0 1 0 1	5	ENQ	NAK	%	5	E	U	e	u
0 1 1 0	6	ACK	SYN	&	6	F	V	f	v
0 1 1 1	7	BEL	ETB	'	7	G	W	g	w
1 0 0 0	8	BS	CAN	(	8	H	X	h	x
1 0 0 1	9	HT	EM	)	9	I	Y	i	y
1 0 1 0	10	LF	SUB	*	:	J	Z	j	z
1 0 1 1	11	VT	ESC	+	;	K	[	k	{
1 1 0 0	12	FF	FS	,	<	L	\	l	|
1 1 0 1	13	CR	GS	-	=	M	]	m	}
1 1 1 0	14	SO	RS	.	>	N	^	n	~
1 1 1 1	15	SI	US	/	?	O	_	o	DEL

Figure: US-ASCII original code

General remarks

- 128 positions, with 95 visible characters and 33 invisible
- visible characters include
  - digits
  - regular and uppercase Latin alphabet letters
  - punctuation signs
  - basic mathematical signs
- invisible signs include
  - space (code 32)
  - tabulators and other formatting items
  - codes for transmissions and machine control

control codes take positions from 0 to 31

27 Introduces an escape sequence.

26 Acts as an end-of-file for the Windows text-mode file I/O

13 CR – return of the carriage (moves the printing position to the start of the line)

12 FF – form feed (cause a printer to eject paper to the top of the next page, or a video terminal to clear the screen)

11 VT – vertical tabulation

10 LF – line feed (which causes a printer to advance its paper)

9 HT – horizontal tab (moves the printing position right to the next tab stop)

8 BS – backspace (may overprint the previous character)

7 BEL – bell (which may cause the device to emit a warning such as a bell or beep sound or the screen flashing)

0 NUL – null (originally intended to be an ignored character, but now used by many programming languages including C to mark the end of a string)[1]

[1] *Control character*. Apr. 2019. URL: `https://en.wikipedia.org/wiki/Control_character`.

Printable characters – codes from 32 to 127

In general uppercase comes before lowercase. Specil characters are placed according to their usage.

32	space, denotes the space between words, as produced by the space bar of a keyboard
33-47	special characters like: !, ", , ..., /
48-57	digits
58-64	special characters like: :, ;, <, =, >, ?, @
65-90	uppercase letters A-Z
91-96	special characters like: [,], ,...
97-122	lowercase letters
123-127	some more special characters

256 code positions

- First 127 codes are exactly the same as in original code
- Next 128 positions represents signs specific to a given national alphabets, e.g.
  - Slavic alphabets
  - Scandinavian alphabets
  - Cyrillic
  - Greek alphabet

Different codes, different problems

There are number of standards for ASCII tables.

- ISO8859 - around 20 tables, including polish one 8859-2
- Microsoft - dozens of tables called **CP - code pages**
- various, local once like Mazovia, Polgaz

Numbers

Words

Computer operate on words, being series of bits, measured as 2^n.
Typically they are 8, 16, 32, 64, 128 bits in a word

Bits operation

Some computers are capable of operating on single bits or bit series of arbitrary any length (bit fields)

Computers word

Data are usually stored using words having length appropriate to this computer type.

Words

To store a logical value only one bit is needed, however in most computers a single native word is used

Value representation

The representation of value depends on operating system and programming language.
Typically **false** is represented as all bits set to 0
The **true** is not that easy, it might be either

- integer equal to 1 (C – operation result)
- integer with no-zero value (C – operation argument)
- word with all bits set to 1

Unsigned integers

Basically uses Natural Binary Code

$$v_{NBC} = \sum_{i=0}^{n-1} b_i 2^i \tag{1}$$

NBC properties

- used for fixed decimal numbers
- 4 bits used for digits coding
- only digits allowed, no other values are considered from 4 bits decoding
- there are 2 forms – condensed (2 digits in a byte), uncondensed (1 digit in a byte)

Signed integers

4 different codes, but the list is just an example

U2

$$v_{U2} = -b_{n-1}2^{n-1} + \sum_{i=0}^{n-2} b_i 2^i \tag{2}$$

U1

$$v_{U1} = -b_{n-1}(2^{n-1} - 1) + \sum_{i=0}^{n-2} b_i 2^i \tag{3}$$

sign-module

$$v_{S-M} = -1^{b_{n-1}} \sum_{i=0}^{n-1} b_i 2^i \tag{4}$$

BIAS

$$v_{BIAS} = -BIAS + \sum_{i=0}^{n-1} b_i 2^i \tag{5}$$

typically $BIAS = 2^{n-1} - 1$

Remarks

- Zero representation
  - number of possible representation
  - easy of 0 detection
- Symmetry for signed values
- Sign representation
- Change in sign
  - U1 - bit negation
  - U2 - negation and increment
  - S-M - negation of sign bit
- Ease of arithmetic operations

Remarks

Zapis	Binary pattern			Byte value	
	min	zero	max	min	max
NBC	00..0	00..0	11..1	0	255
U2	10..0	00..0	01..1	-128	127
Z-M	11..1	00..0 10..0	01..1	-127	127
U1	10..0	00..0 11..1	01..1	-127	127
BIAS	00..0	01..1	11..1	-127	128

Remarks

Zapis	Value of 16 bit word		Value of 32 bit word	
	min	max	min	max
NBC	0	65535	0	4294967295
U2	-32768	32767	-2147483648	2147483647
Z-M	-32767	32767	-2147483647	2147483647
U1	-32767	32767	-2147483647	2147483647
BIAS	-32767	32768	-2147483647	2147483648

References

References I

[1] *Control character*. Apr. 2019. URL: https://en.wikipedia.org/wiki/Control_character.