情報処理入門

4章「情報の表現」

中島康彦

§4. 1 情報とデータ

情報

- 次に何がくるかわからない
- 何かがくることにより得られる確かさ＝情報

情報の単位

- 二者択一の事象の1つが起れば，1ビットの情報
- 四者択一の事象の1つが起れば，2ビットの情報
- 生起確率が等しいQ個の事象の1つが起こったことにより得られる情報は，
  $\log_2 Q$ビット
- 事象Eの生起確率がP(E)である場合，Eが起こったことにより得られる情報は，
  $-\log_2 P(E)$ビット

§4. 1 情報とデータ(続き)

データ

- 情報を計算機が処理可能な形にしたもの
- 記号による表現

  実世界の対象物をわかりやすく，少ない量で表現できること.

  現実の計算機の上で，効率良く処理できること.

情報の構造の明確化

- 「記号の性質および記号間の関係」の明確化に置き換えることができる.

§4. 2 計算機における情報の表し方

情報の最小単位:ビット

- 1ビット

  □ 0または1の2通り

- 2ビット

  □□ 00 01 10 11の4通り

- 1バイト(8ビット)

  □□□□□□□□ 2の8乗=256通り

§4. 2 計算機における情報の表し方(続き)

- 1ワード(16ビット)
  □□□□□□□□□□□□□□□□
  2の16乗=6万5536通り

- 1ワード(32ビット)
  □□□□□□□□□□□□□□□□□□□□□□□□…
  2の32乗=42億9496万7296通り

- 1ワード(64ビット)
  □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□…
  2の64乗=1844京6744兆0737億0955万1616通り

§4. 3 基本データ型

符号無し整数/固定小数点数

unsigned integer/fixed-point number

- 8ビット長の場合
  00000000 … 0
  00000001 … 1
  00000010 … 2
  00000011 … 3
  11111111 … 255
- 演算
  00000010(2)+00000011(3)=00000101(5)
- オーバフロー/アンダフロー
  11111111(255)+00000011(3)=00000010(2)
  00000010(2)-00000011(3)=11111111(255)

§4. 3 基本データ型(続き)

符号付き整数/固定小数点数

signed integer/fixed-point number

- 8ビット長の場合

```
10000000    … −128
11111111    … −1
00000000    …  0
00000001    … +1
01111111    … +127
```

- 演算

11111111(−1)+00000011(3)=00000010(2)

- オーバフロー/アンダフロー

01111111(127)+00000011(3)=10000010(−126)

10000010(−126)−00000011(3)=01111111(127)

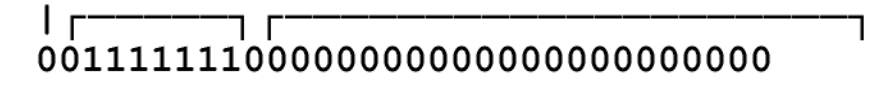
§4. 3 基本データ型(続き)

実数/浮動小数点数(IEEE754形式の場合)

real/floating-point number

- 単精度浮動小数点数 … 32ビット長

符号1 指数8 仮数23



指数=0かつ仮数=0 … 符号付ゼロ

指数=0かつ仮数≠0 … 不正規化数

(−1)^符号×2^(−126)×0.仮数

0<指数<255 … 正規化数

(−1)^符号×2^(指数−127)×1.仮数

指数=255かつ仮数=0 … 符号付無限大(∞)

指数=255かつ仮数≠0 … 非数値

例えば0÷0の結果

§4. 3 基本データ型(続き)

PI = 3.14159265358979323846

▶単精度浮動小数点数は3.141592まで正しい

```
0/1 00000000 00000000000000000000000 … +/-0
0/1 01111100 00000000000000000000000 … +/-0.125
0/1 01111101 00000000000000000000000 … +/-0.25
0/1 01111110 00000000000000000000000 … +/-0.5
0/1 01111111 00000000000000000000000 … +/-1
0/1 10000000 00000000000000000000000 … +/-2
0/1 10000000 10000000000000000000000 … +/-3
0/1 10000000 10010010000111111011010 … +/-3.14159250
0/1 10000000 10010010000111111011011 … +/-3.14159274
0/1 10000000 10010010000111111011100 … +/-3.14159297
0/1 10000000 11111111111111111111111 … +/-3.99999976
0/1 10000001 00000000000000000000000 … +/-4
0/1 10000010 00000000000000000000000 … +/-8
0/1 11111111 00000000000000000000000 … +/-∞
0/1 11111111 XXXXXXXXXXXXXXXXXXXXXXX … 非数値
```

§4. 3 基本データ型(続き)

▶倍精度浮動小数点数 … 64ビット長

符号1 指数11 仮数52

(-1)^符号×2^(指数-1023)×1.仮数

▶拡張倍精度浮動小数点数 … 80／128ビット長

符号1 指数15 仮数64

符号1 指数15 仮数112

(-1)^符号×2^(指数-16383)×1.仮数

実数/浮動小数点数(VAX形式)

実数/浮動小数点数(IBM形式)

§4. 3 基本データ型(続き)

論理 logical number

- 1ビット長の場合
  - 0 … 偽(FALSE)
  - 1 … 真(TRUE)
- 8ビット長の場合
  - 00000000 … 偽(FALSE)
  - その他 … 真(TRUE)
- 各ビットを独立論理値とした場合のビット毎演算
  - 否定: .NOT.00001111 = 11110000
  - 論理和: 00111100.OR. 00001111 = 00111111
  - 論理積: 00111100.AND.00001111 = 00001100
  - 排他論理和:00111100.XOR.00001111 = 00110011

§4. 3 基本データ型(続き)

文字 character

- null文字
  - 00000000 … ''
- 1バイト文字 (半角文字)
  - 00110010 … '2'
- 2バイト文字 (全角文字)
  - 10100011,10110010 … '2'
- 文字の演算(比較)が可能
  - 00110010('2')<00110011('3')
  - 10100100,10100010('あ')<10100100,11110011('ん')

§4. 3 基本データ型(続き)

文字列 string

- 文字の並び+終端(null)文字
  00110010,00110011,00000000 … "23"

- 空の文字列
  00000000 … ""

- 文字列の演算(比較)が可能
  "あい"<"あん"

§4. 4 二種類のエンディアン

データの桁位置の高低

- 下位ビット … 桁位置が低いビット
  最下位ビット=Least Significant Bit

- 上位ビット … 桁位置が高いビット
  最上位ビット=Most Significant Bit

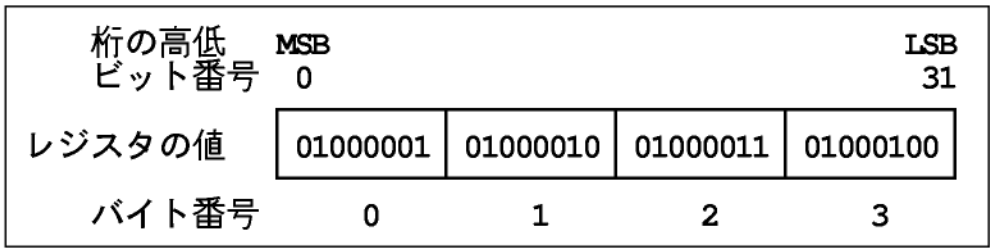
主記憶アドレスの高低

- 低アドレス … 数値として小(0x00010000)
- 高アドレス … 数値として大(0xfffff000)

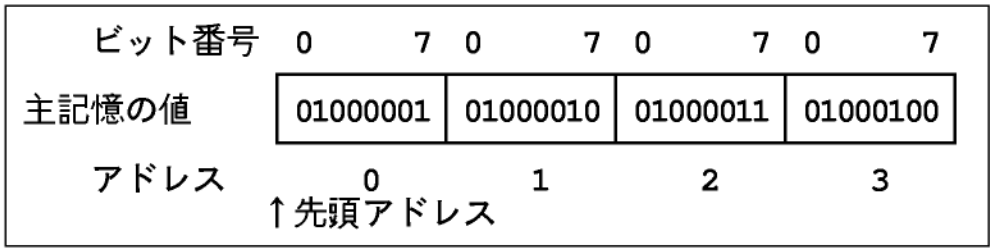
§4. 4 二種類のエンディアン(続き)

▶ビッグエンディアン … IBM汎用機，SPARCなど

１６進数 0x41424344 … 文字 ＡＢＣＤ



↓バイト番号順に格納



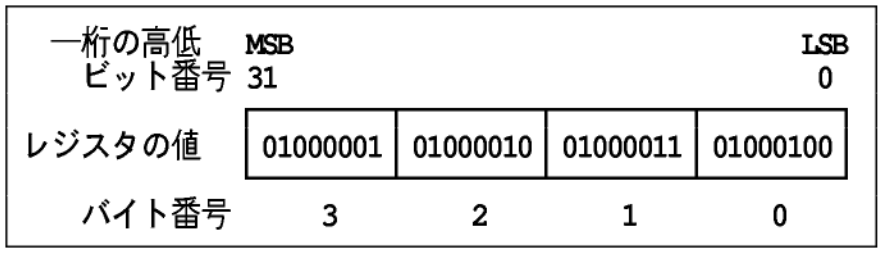
↓アドレス順に出力

ファイル	ＡＢＣＤ

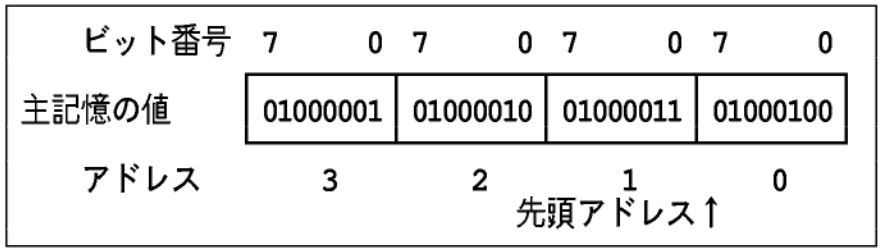
§4. 4 二種類のエンディアン(続き)

▶リトルエンディアン … Pentiumなど

１６進数 0x41424344 … 文字 ＡＢＣＤ



↓バイト番号順に格納



↓アドレス順に出力

ファイル	ＤＣＢＡ

§4. 4 二種類のエンディアン(続き)

エンディアンの異なる計算機間では，主記憶データに互換性がない.

- C言語の`read/write`などにより，主記憶データを直接ファイルに格納したもの.
- 構造体をファイルに格納する場合は，個々の要素のエンディアンを考慮しなければならない.
- ネットワーク間でデータを共有する場合にはエンディアンに関する取り決めが必要.
- 移植性の高いソフトウェアを開発する際も同様.

§4. 5 文字コード

EBCDIC … IBM汎用機

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
														(下位4ビット)		
0	NUL	SOH	STX	ETX	PF	HT	LC	DEL	GE	RLF	SMM	VT	FF	CR	SO	SI
1	DLE	DC1	DC3	TM	RES	NL	BS	IL	CAN	EM	CC	CU1	IFS	IGS	IRS	IUS
2	DS	SOS	FS		BYP	LF	ETB	ESC			SM	CU2		ENQ	ACK	BEL
3			SYN		PN	RS	UC	EOT				CU3	DC4	NAK		SUB
4	SP										¢	.	<	(	+	|
5	&										!	$	*	)	;	
6		/										,	%	_	>	?
7											:	#	@	'	=	"
8		a	b	c	d	e	f	g	h	i						
9		j	k	l	m	n	o	p	q	r						
A			s	t	u	v	w	x	y	z						
B																
C		A	B	C	D	E	F	G	H	I						
D		J	K	L	M	N	O	P	Q	R						
E			S	T	U	V	W	X	Y	Z						
F	0	1	2	3	4	5	6	7	8	9	LVM					EO

§4. 5 文字コード(続き)

EBCDIK … 片仮名に対応

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
														(下位4ビット)		
0	NUL	SOH	STX	ETX	PF	HT	LC	DEL	GE	RLF	SMM	VT	FF	CR	SO	SI
1	DLE	DC1	DC3	TM	RES	NL	BS	IL	CAN	EM	CC	CU1	IFS	IGS	IRS	IUS
2	DS	SOS	FS		BYP	LF	ETB	ESC			SM	CU2		ENQ	ACK	BEL
3			SYN		PN	RS	UC	EOT				CU3	DC4	NAK		SUB
4	SP	。	「	」	、	・	ヲ	ァ	ィ	ゥ	[	.	<	(	+	|
5	&	ェ	ォ	ャ	ュ	ョ	ッ		ー		]	¥	*	)	;	
6		/									^	,	%	_	>	?
7											:	#	@	'	=	"
8		ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ		サ	シ	ス	セ
9	ソ	タ	チ	ツ	テ	ト	ナ	ニ	ヌ	ネ	ノ			ハ	ヒ	フ
A			ヘ	ホ	マ	ミ	ム	メ	モ	ヤ	ユ		ヨ	ラ	リ	ル
B											レ	ロ	ワ	ン	゛	゜
C	{	A	B	C	D	E	F	G	H	I						
D	}	J	K	L	M	N	O	P	Q	R						
E	$		S	T	U	V	W	X	Y	Z						
F	0	1	2	3	4	5	6	7	8	9	LVM					EO

§4. 5 文字コード(続き)

ASCII

(American national Standard Code for Information Interchange)

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

§4. 5 文字コード(続き)

JIS X0201 ローマ字

(下位4ビット)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	NL	VT	NP	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP	!	”	#	$	%	&	’	(	)	*	+	,		.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	¥	]	^	_
6	‘	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	‾	DEL

§4. 5 文字コード(続き)

制御コード

(下位4ビット)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	NL	VT	NP	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP															
7																DEL

- NUL : NULl
- SOH : Start Of Heading
- STX : Start of TeXt
- ETX : End of TeXt
- EOT : End Of Transmission
- ENQ : ENQuiry
- ACK : ACKnowledge
- BEL : BELl
- BS : BackSpace
- HT : Horizontal Tab
- NL : New Line
- VT : Vertical Tab
- NP : New Page
- CR : Carriage Return
- SO : Shift Out
- SI : Shift In
- DLE : Data Link Escape
- DCn : Device Control n
- NAK : Negative AcKnowledge
- SYN : SYNchronous idle
- ETB : End of Transmission Block
- CAN : CANcel
- EM : End of Medium
- SUB : SUBstitute
- ESC : ESCape
- FS : File Separator
- GS : Group Separator
- RS : Record Separator
- US : Unit Separator
- SP : SPace
- DEL : DELete

§4. 5 文字コード(続き)

JIS X0201 片仮名

►いわゆる半角片仮名，使用は避けるべき

(下位4ビット)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		。	「	」	、	・	ヲ	ァ	ィ	ゥ	ェ	ォ	ャ	ュ	ョ	ッ
B	ー	ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ	サ	シ	ス	セ	ソ
C	タ	チ	ツ	テ	ト	ナ	ニ	ヌ	ネ	ノ	ハ	ヒ	フ	ヘ	ホ	マ
D	ミ	ム	メ	モ	ヤ	ユ	ヨ	ラ	リ	ル	レ	ロ	ワ	ン	゙	゚

§4. 5 文字コード(続き)

JIS C6226-1978(旧JIS)…日本規格協会

「第1水準」2965字,「第2水準」3384字,「非漢字」453字

、。，．・：；？！゛゜´｀¨＾￣＿ヽヾゝゞ〃仝々〆〇ー―‐／＼～∥｜…‥‘’“”（）〔〕［
◆□■△▲▽▼※〒→←↑↓〓

0123456789 ABCDEFGHIJKLMN

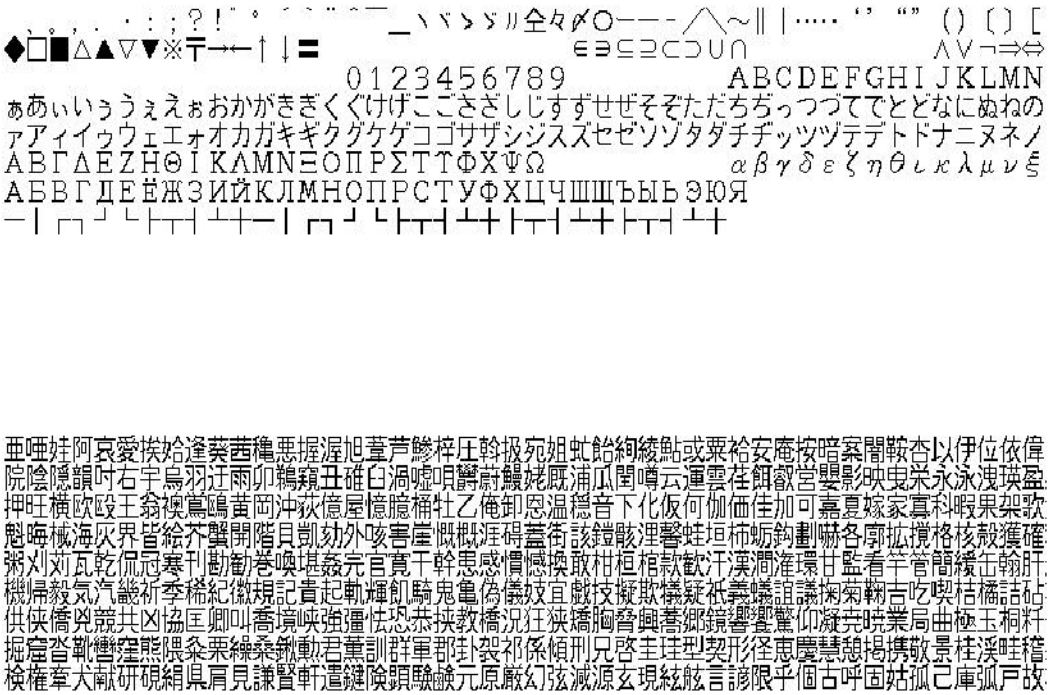
ぁあぃいぅうぇえぉおかがきぎくぐけげこごさざしじすずせぜそぞただちぢっつづてでとどなにぬねの
ァアィイゥウェエォオカガキギクグケゲコゴサザシジスズセゼソゾタダチヂッツヅテデトドナニヌネノ
ΑΒΓΔΕΖΗΘΙΚΛΜΝΞΟΠΡΣΤΤΦΧΨΩ αβγδεζηθικλμνξ
АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

亜唖娃阿哀愛挨姶逢葵茜穐悪握渥旭葦芦鯵梓圧斡扱宛姐虻飴絢綾鮎或粟袷安庵按暗案闇鞍杏以伊位依偉
院陰隠韻吋右宇烏羽迂雨卯鵜窺丑碓臼渦嘘唄鬱蔚鰻姥厩浦瓜閏噂云運雲荏餌叡営嬰影映曳栄永泳洩瑛盈
押旺横欧殴王翁襖鴬鴎黄岡沖荻億屋憶臆桶牡乙俺卸恩温穏音下化仮何伽価佳加可嘉夏嫁家寡科暇果架歌
魁晦械海灰界皆絵芥蟹開階貝凱劾外咳害崖慨概涯碍蓋街該鎧骸浬馨蛙垣柿蝸鈎劃嚇各廓拡撹格核殻獲確
潟刈苅瓦乾侃冠寒刊勘勧巻喚堪姦完官寛干幹患感慣憾換敢柑桓棺款歓汗漢澗灌環甘監看竿管簡緩缶翰肝
機帰毅気汽畿祈季稀紀徽規記貴起軌輝飢騎鬼亀偽儀妓宜戯技擬欺犠疑祇義蟻誼議掬菊鞠吉吃喫桔橘詰砧
供侠僑兇競共凶協匡卿叫喬境峡強彊怯恐恭挟教橋況狂狭矯胸脅興蕎郷鏡響饗驚仰凝尭暁業局曲極玉桐粁
掘窟沓靴轡窪熊隈粂栗繰桑鍬勲君薫訓群軍郡卦袈祁係傾刑兄啓圭珪型契形径恵慶慧憩掲携敬景桂渓畦稽
検権牽犬献研硯絹県肩見謙賢軒遣鍵険顕験鹸元原厳幻弦減源玄現絃舷言諺限乎個古呼固姑孤己庫弧戸故

§4. 5 文字コード(続き)

JIS X0208-1983(新JIS)…日本規格協会

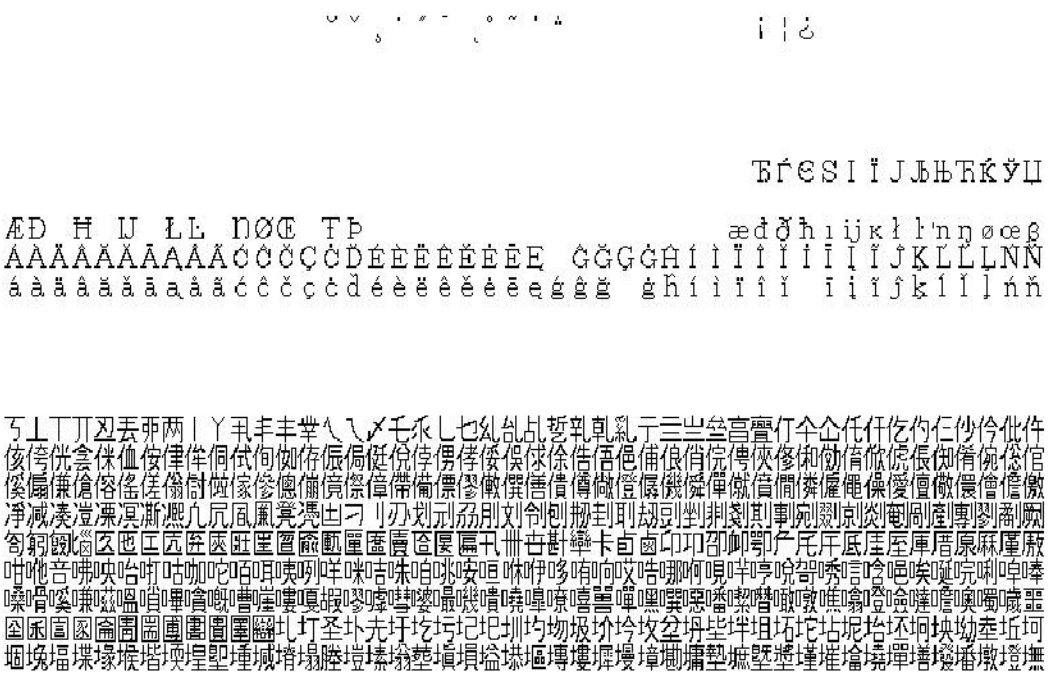
「第1水準」2965字,「第2水準」3388字,「非漢字」524字



§4. 5 文字コード(続き)

JIS X0212-1990(補助漢字)

「漢字」5801字,「非漢字」266字



§4. 5 文字コード(続き)

ISO 10646
JIS X0221-1995(UCS)…日本規格協会

- UCS-4 ： 1文字を4バイトで表現
  上位2バイトが0の範囲を
  BMP(Basic Multilingual Plane)と呼ぶ.

- UCS-2 ： BMPの上位2バイトを省いた2バイトコード
  これがいわゆるUnicode

§4. 6 日本語コード

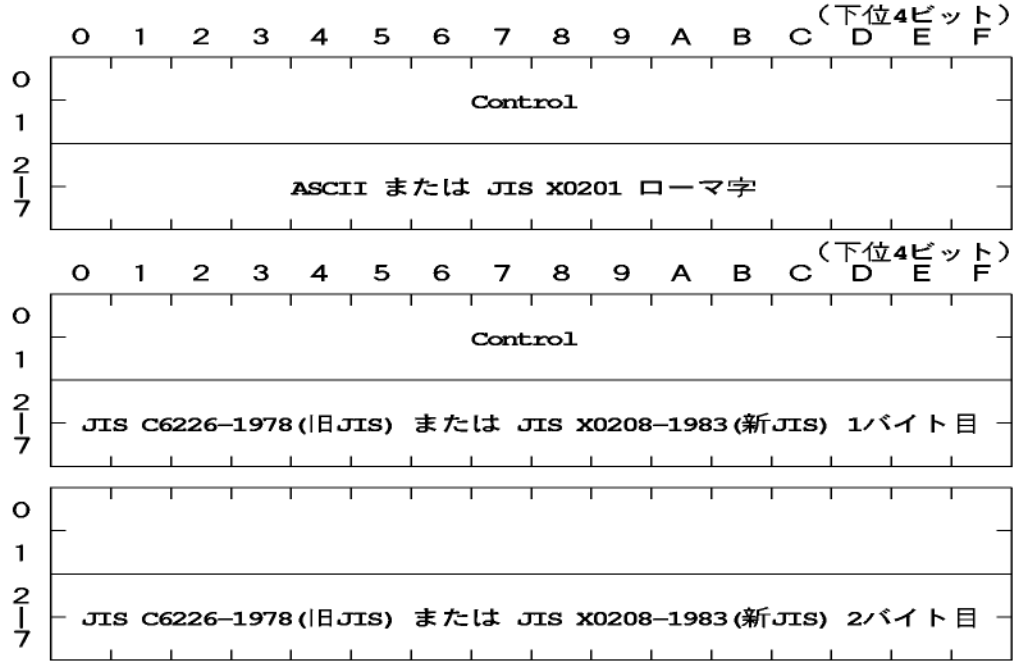
シフトJISコード … Microsoft, アスキー

- 特別な遷移コードがない代わりに字数が少ない

§4. 6 日本語コード(続き)

ISO-2022-JP(いわゆるJISコード)

- 7ビットの文字列で漢字を表現できる



§4. 6 日本語コード(続き)

- 2バイト文字への遷移コード

  JIS C6226-1978(旧JIS)へは, 1B2440 ESC$@

  JIS X0208-1983(新JIS)へは, 1B2442 ESC$B

- 1バイト文字への遷移コード

  JIS X0201-1976 ローマ字へは, 1B284A ESC(J

  ASCIIへは, 1B2842 ESC(B

ISO-2022-JPにはJIS X0201 片仮名が無い.

- JIS7方式 … 20～5Fに割り当てる.

  1B2849 ESC(Iを片仮名開始とする方法

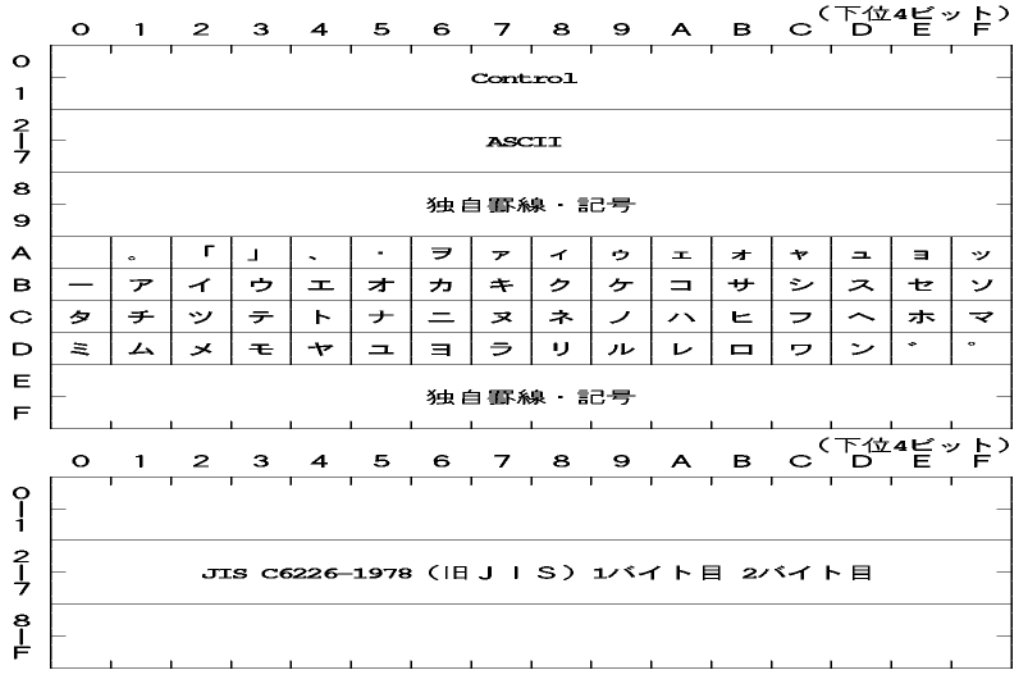
  0E/0F(ShiftOut/In)を片仮名開始/終了とする方法

- JIS8方式 … A1～DFに割り当てる.

§4. 6 日本語コード(続き)

NEC漢字コード

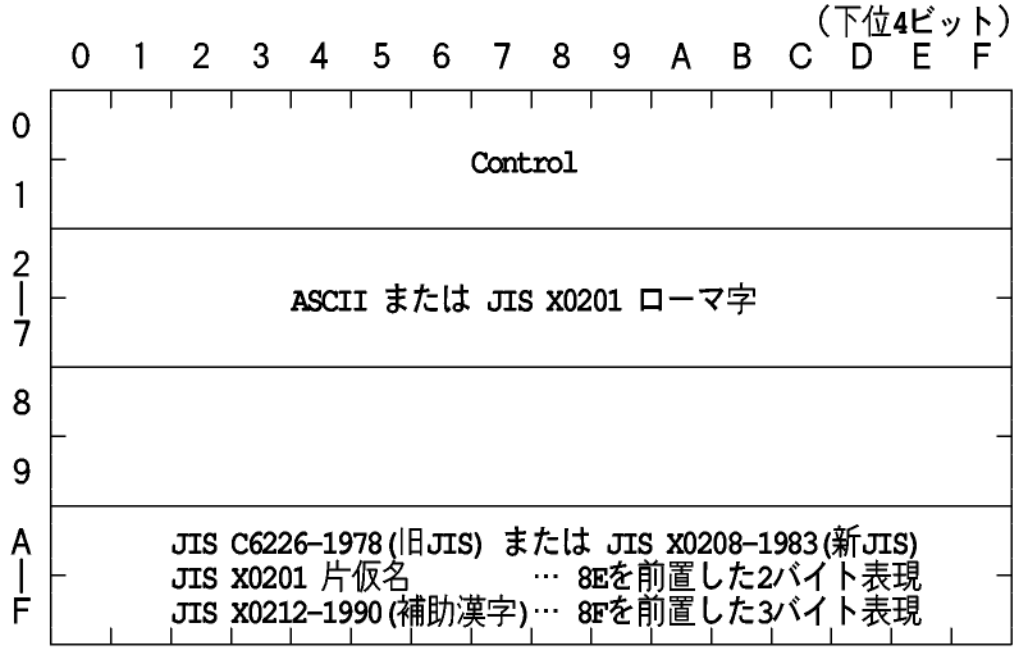
- 独自の遷移コード 1B4B/1B48(KanjiIn/Out)



§4. 6 日本語コード(続き)

EUC(Extended Unix Code)

- 重複する場合, シフトJISと区別できない



§4. 6 日本語コード(続き)

UTF-8

- UCS-2に基づく1～3バイトの文字コード

- 0000～007Fの場合
  下1バイトを使用
- 0080～07FFの場合(00000xxxxxyyyyyy)
  110xxxxxを1バイト目, 10yyyyyyを2バイト目とする
- 0800～FFFFの場合(xxxxyyyyyyzzzzzz)
  1110xxxxを1バイト目, 10yyyyyyを2バイト目, 10zzzzzzを3バイト目とする

今日はここまで