

平成10年度プロジェクト研究  
「文字コード自動認識機能付き  
ファイル・ビューワの開発」

指導教官 大岩幸太郎 教授

大分大学 教育学部 情報社会文化課程  
情報教育コース 0710300 永岩里恵

# 目次

---

## 第1章 はじめに

- 1.1 研究の動機
- 1.2 研究の目的

## 第2章 日本語の符号化

## 第3章 文字コード

- 3.1 JIS コード
- 3.2 シフト JIS コード
- 3.3 日本語 EUC コード

## 第4章 研究結果

## 参考文献

# 第 1 章 はじめに

## 研究の動機

情報通信において、異なる文字コード間で情報交換した場合、表示されるべき文字が変わってしまう「文字化け」が生じることがあるが、Netscape や IE などのブラウザソフトには文字コードを自動認識し、正しく文字を表示する機能が付いている。

そこで、文字コードに興味を持ち、ファイルの文字コードが何コードかが分かるファイル・ビューワの開発をテーマにしようと考えた。

## 研究の目的

テキストファイルがどの文字コードであるかが明確に分かり、文字コードの操作が出来るように、以下の点を目的としたソフトを開発することにした。

テキストファイルを開いた時に、どの文字コードかを自動認識してユーザに分かるようにする。

## 第2章 日本語の符号化

コンピュータの内部では文字を数値として処理する「符号(コード)化」が行われ、対応する数値によって文字を特定することができる。ASCII コード表を表 1-1 に、符号化の例を図 1-1 に示す。

表 1-1 ASCII コード表

		上位4ビット							
		0	1	2	3	4	5	6	7
下位4ビット	0		D E		0	@	P	'	p
	1	S H	D 1	!	!	A	Q	a	q
	2	S X	D 2	"	2	B	R	b	r
	3	E X	D 3	#	3	C	S	c	s
	4	E T	D 4	\$	4	D	T	d	t
	5	E Q	N K	%	5	E	U	e	u
	6	A K	S N	&	6	F	V	f	v
	7	B L	E B	'	7	G	W	g	w
	8	B S	C N	(	8	H	X	h	x
	9	H T	E M	)	9	I	Y	i	y
	A	L F	S B	*	:	J	Z	j	z
	B	H M	E C	+	:	K	[	k	{
	C	C L	→	,	<	L	¥	l	
	D	C R	←	-	=	M	]	m	
	E	S O	↑	.	>	N	^	n	~
	F	S I	↓	/	?	O	_	o	

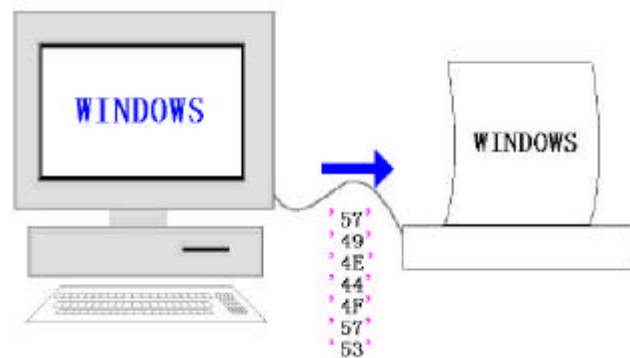


図 1-1 文字の印字

図 1-1 に示したようにコンピュータはプリンタに直接文字を送るのではなく、文字に対応する数値を送り、プリンタ側は数値に対応する文字を印字している。

初期のコンピュータシステムは最初、イギリスとアメリカで開発され、7または8ビットあれば大半の文字が表現可能であった。そのため英語を始めとする言語を書き表す文字を符号化するには1バイトの符号化方式(ASCII)が定着している。しかし、日本語のように4つの表記法(ローマ字、平仮名、片仮名、漢字)を1バイトに収めるには無理があるため、2バイトを用いて日本語を表現する方式が採用された。図 1-2 に ASCII コードと日本語コード(JIS コード)との比較を示す。

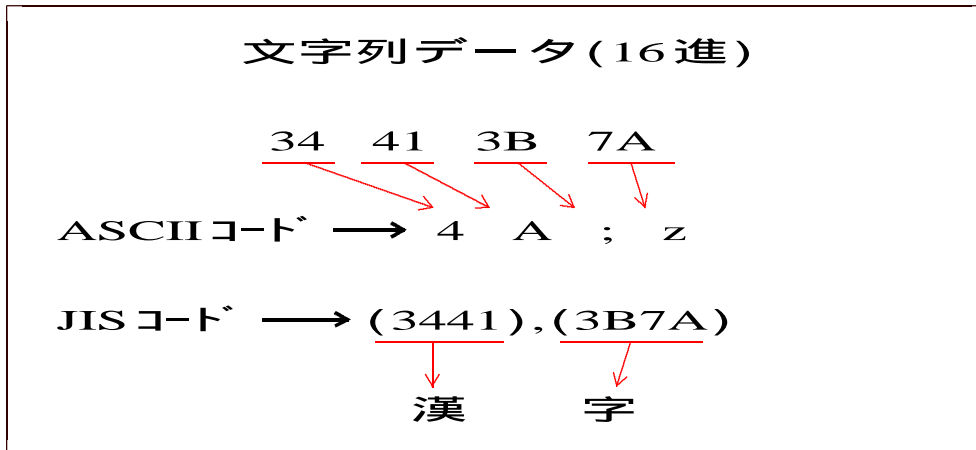


図 1-2 ASCIIコードとJISコードの比較

図 1-2 で示したように ASCII は 1 バイトで文字を処理している。それに対して日本語コード (JIS コード) は文字を 2 バイトで処理している。

このように符号化方式等の環境の異なる間での情報交換では、思わぬ問題に遭遇することがある。こうした問題を考えるためにも日本語の符号化方式を理解する必要があると考えた。一般的に日本語の符号化方式には、

- (1) JIS
- (2) シフト JIS
- (3) 日本語 EUC

の 3 つの方式が用いられている。

## 3章 文字コード

一般的な日本語文字コードに、

- (1) JIS コード
- (2) シフト JIS コード
- (3) 日本語 EUC コード

の3つが挙げられる。

本章では、上記(1)、(2)、(3)を順に説明する。

### 3.1 JIS コード

Japanese Industrial Standard の略。7ビット符号化方式。

JIS コードは日本語を2バイトで表現するが、データを0～127の間で保持するので海外のコンピュータでも問題なく処理できる利点を持ちコンピュータ間での情報を伝達するための情報交換符号として用いられるので、「外部コード」と呼ぶ。

JIS コードの特徴は、エスケープシーケンスを用いることで1バイトモードと2バイトモードを切り替える点である。

ここでエスケープシーケンスについて説明すると、エスケープシーケンスは、エスケープキャラクタ(16進数で1B)とそれ以外の文字の組み合わせによって構成され、このエスケープシーケンスを用いて異なるモード間の切り替えを行う。表3.1にJISコードで用いるエスケープシーケンスを示す。

表 3.1 JIS コードで用いるエスケープシーケンス

[ESC] ( B	ASCII
[ESC] ( J	JIS ローマ字
[ESC] \$ @	旧 JIS
[ESC] \$ B	新 JIS

JIS コードのエスケープシーケンスを実際に用いてモード切り替えを行っているのを図3.1に示す。

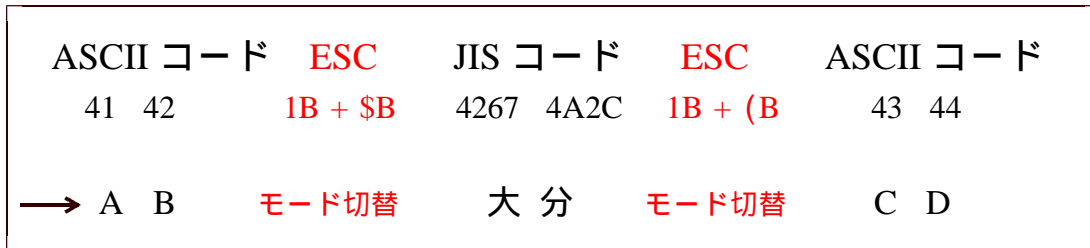


図 3.1 エスケープシーケンスを用いてのモード切り替え

図 3.1 に示したようにエスケープシーケンスを用いたことで ASCII コードと JIS コードを切り替え、2つの異なるコードの混在する文書进行处理することが可能となる。

JIS コードは「JIS7」と「JIS8」と呼ばれる 2 種類の方式で、半角片仮名もサポートしている。JIS7 コードでは第 8 ビットを全く使用せず JIS コードそのものに半角片仮名モードに切り替えるエスケープシーケンスを加えたもので、テキスト内に 2 バイトの日本語、1 バイトの ASCII 文字、半角片仮名が混在している時は、少なくとも 3 つのエスケープシーケンスが必要となる。

これに対し JIS8 コードは第 8 ビットを使用し、シフト JIS コードの半角片仮名のバイト範囲と同じである。

ここで、JIS コードの仕様を表 3.2 に、また JIS コードのコード空間を図 3.2 に示す。

表 3.2 JIS コードの仕様

バイト範囲	16 進数
2 バイト文字 第 1 バイト範囲 第 2 バイト範囲	21 - 7E 21 - 7E
JIS7 半角片仮名 バイト範囲	21 - 5E
JIS8 半角片仮名 バイト範囲	A1 - DF
ASCII/ローマ字 バイト範囲	21 - 7E



図 : 3.2 JIS コードのコード空間

## 3.2 シフト JIS コード

Shifted JIS Code の略。8 ビット符号化方式。

名称の由来は、2 バイト文字のコード位置が半角片仮名とぶつからないようにシフトしたことに由来する。

マイクロソフト社により開発され、日本のパソコン等の内部コードとして使用されている。シフト JIS は「MS 漢字」(MS はマイクロソフトの略称)、「SJIS」(Shift-JIS の略称)とも呼ばれる。

図 3.2 に JIS コードとシフト JIS コードの比較を示す。

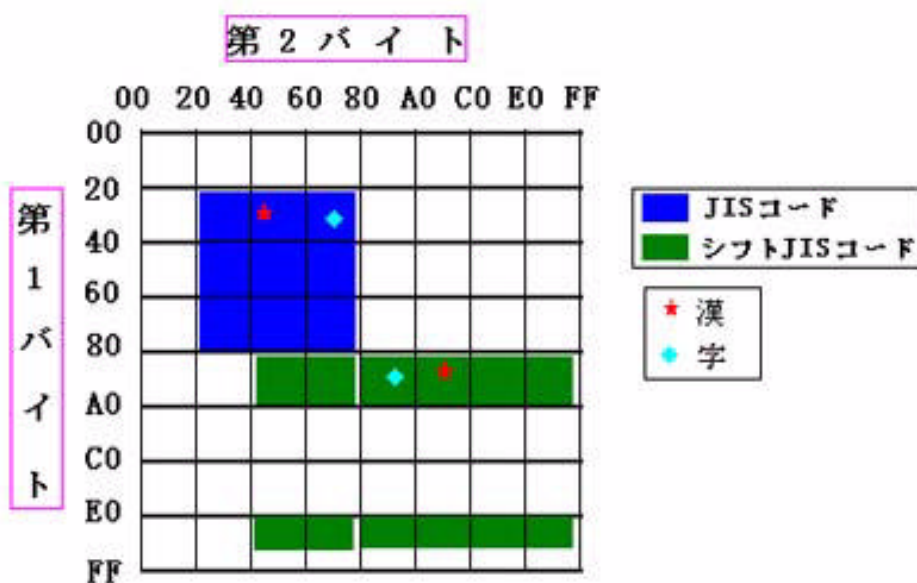


図 3.2 JIS コードとシフト JIS コード

図 3.2 に示したように JIS コードとシフト JIS コードでは、同じ「漢」と「字」でもコード値が違っている。

また、シフト JIS はエスケープシーケンスを用いず 81 ~ 9F、E0 ~ EF の範囲のバイトが現れると 2 バイトモードが開始され、このバイトは 2 バイト文字の第 1 バイトとして処理される。

第 1 バイトの範囲は 8 ビット文字セット (ASCII 文字セット) の範囲に収まっており、またシフト JIS コードは半角片仮名、ASCII/JIS ローマ字をサポートしている。

表 3.3 にシフト JIS のコード仕様を示す。

表 3.3 シフト JIS コードの仕様

バイト範囲	16 進数
2 バイト文字 第 1 バイト範囲 第 2 バイト範囲	81 - 9F, E0 - EF 40 - 7E, 80 - FC
半角片仮名 バイト範囲	A1 - DF
ASCII/ローマ字 バイト範囲	21 - 7E

### 3.3 日本語 EUC コード

Extended Unix Code の略。8 ビット符号化方式。

EUC は UNIX ワークステーションの内部コードとして使用され、「UJIS」(UNIXized JIS の略)、「AT&T JIS」とも呼ばれる。

マルチバイトコードをサポート。日本語の符号化専用ではなく日本語や、それ以外の言語を処理するため開発された。

日本語システムの実装に用いられるのは一般的に「EUC 圧縮フォーマット」である。日本語 EUC コードもエスケープシーケンスは用いずに、ASCII 文字はそのまま、JIS 漢字は 2 バイトのそれぞれ最上位ビットを 1 に変えたのもで表し、1 バイト片仮名(半角片仮名)は最上位ビットを 1 にした上で前に 8E をつけて表す。

日本語 EUC コードは 4 つのコードセットで構成されている。

コードセット 0 は常に ASCII 文字セット、または ASCII 文字セットのその国独自の版が定められ、その他のコードセットは各種の版が定義され各国ごとに選択可能である。

次に、一般的な EUC 圧縮フォーマットのコードの仕様を表 3.4 に示す。

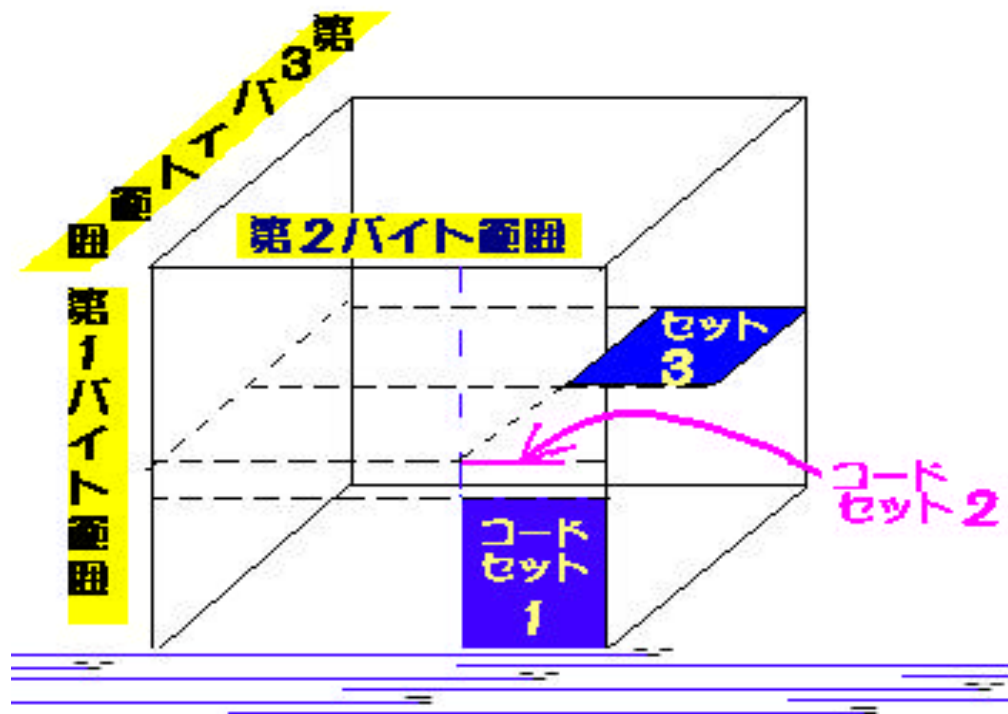
表 3.4 EUC 圧縮フォーマットの仕様

バイト範囲	16 進数
コードセット 0 (ASCII/JIS ローマ字) バイト範囲	21-7E
コードセット 1 (JIS コード) 第 1 バイト範囲 第 2 バイト範囲	A1-FE A1-FE
コードセット 2 (半角片仮名) 第 1 バイト範囲 第 2 バイト範囲	8E A1-DF
コードセット 3 (補助漢字) 第 1 バイト範囲 第 2 バイト範囲 第 3 バイト範囲	8F A1-FE A1-FE

8E は「SS2」(Single Shift 2 の略)で、コードセット 2 に収められた文字 1 つ 1 つの先頭に付けられ、同様に 8F は「SS3」(Single Shift 3 の略)で、コードセット 3 に収められた 1 つ 1 つの先頭に付けられる。

また、EUC コードでは半角片仮名は 2 バイトで表現される。

図 3.3 に EUC 圧縮フォーマットのコード空間を示す。



図：3.3 EUC 圧縮フォーマットのコード空間

## 第 4章 研究結果

各コードの認識結果をそれぞれ図 4.1、図 4.2、図 4.3 に示す。

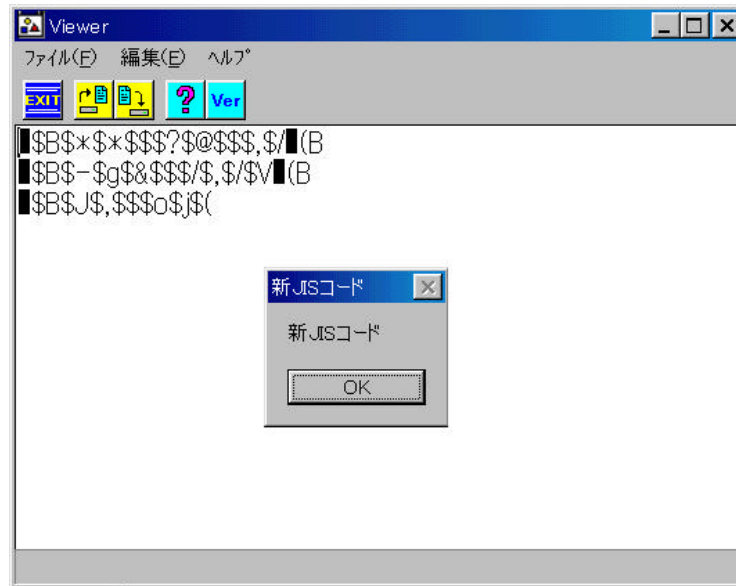


図 : 4.1 JIS コードと認識

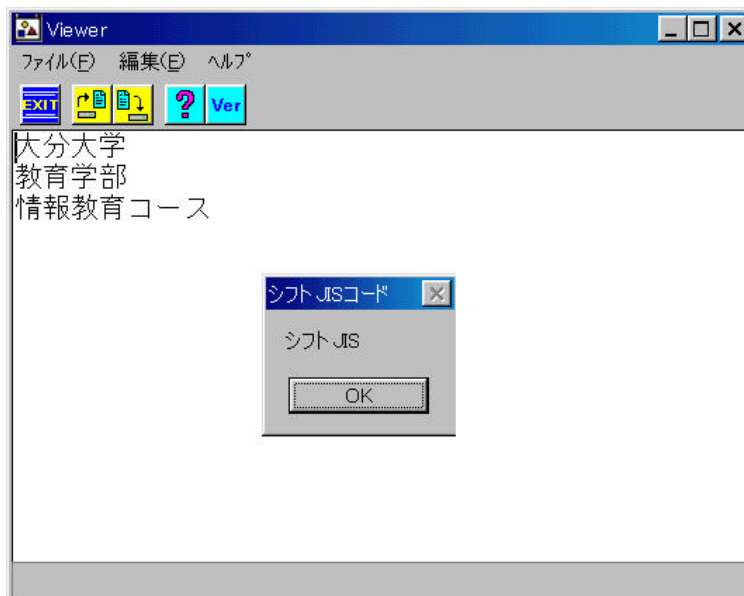


図 : 4.2 シフトJIS コードと認識

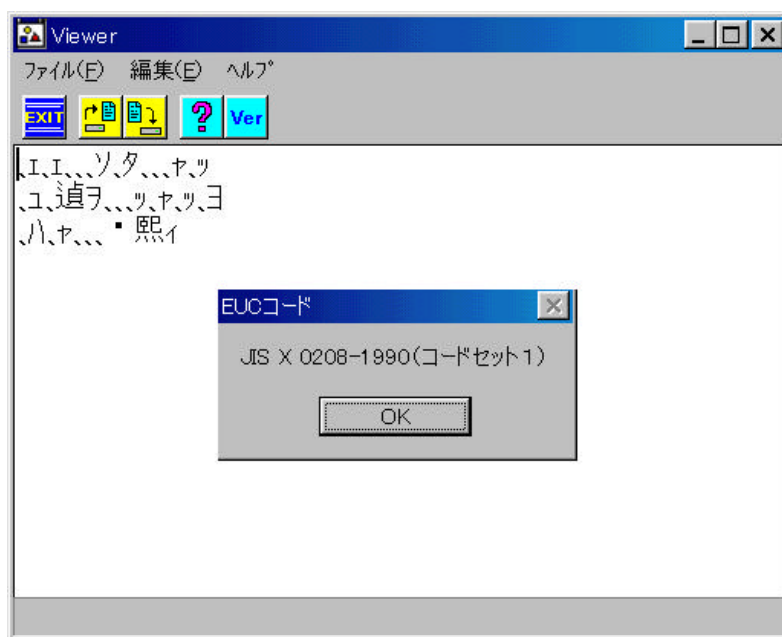


図 :4.3 日本語 EUC コードと認識

本ソフトでは file.ide のプロジェクトファイルでファイルを4つに分けて開発を行った。  
そのステップ数、実行ファイルのサイズをそれぞれ表 4.1、表 4.2 に示す。

表 4.1 ファイル名と結果

ファイル名	ステップ数
edit.cpp	362 steps
Kanji.h	38 steps
edit.rc	418 steps
edit.rh	24 steps
合計	842 steps

表 4.2 実行ファイルサイズ

実行ファイル名	実行ファイルサイズ
fileviewer.exe	419KB

また、開発環境は以下の通りである。

表 4.3 開発環境

機種	COMPAQ DESKPRO
OS	Windows98
言語	Borland C++ Ver5.0

## 参考文献

---

Ken Lunde 著

「日本語情報処理」 1998  
ソフトバンク株式会社

Duke 著

「Borland C++ 達人テクニック」(上・下巻) 1995  
株式会社ディー・アート

Clayton Walnum 著

「BORLAND C++ 4.X プログラマーズハンドブック」 1996  
株式会社ビー・エヌ・エヌ

## 参考サイトURL

---

### 1.日本語と文字コード

<http://www.kanzaki.com/docs/jcode.html>

### 2.文字コードの国際規格について

<http://turbine.kuee.kyoto-u.ac.jp/FAQ/kanji-code.html>

### 3.インターネットメールの注意点

<http://www02.so-net.ne.jp/~hat/imap/cover.html>



# 日本語コード変換表

## JIS/EUC シフトJISの変換

- 1) 第1バイトは変換表を用いて変換。JIS/EUCコードは第1バイトの値を見つけ「シフトJIS第1バイト」の欄でそれに対応する値を見る。
- 2) 第2バイトはJIS/EUCコードの第1バイトが奇数であれば、シフトJISの第2バイトの欄にある左の値を選ぶ。偶数の場合は同欄の右の値を選ぶ。

## シフトJIS JIS/EUCの変換

- 1) シフトJIS第2バイトの値が「シフトJIS第2バイト」欄の左側の列の場合「シフトJIS第1バイト」欄を始めから見ていき、最初に出現したJIS第1バイトの行を用いて、求めるコードの第1バイトを決める。シフトJISの第2バイトが右列にある場合は「シフトJIS第1バイト」欄の中で2回目に出現した行を用いる。
- 2) シフトJISの第2バイトは変換表を用いて一対一の変換が可能。

JIS	EUC	シフトJIS第 1 バイト	シフトJIS第 2 バイト
21	A1	81	40 9F
22	A2	81	41 A0
23	A3	82	42 A1
24	A4	82	43 A2
25	A5	83	44 A3
26	A6	83	45 A4
27	A7	84	46 A5
28	A8	84	47 A6
29	A9	85	48 A7
2A	AA	85	49 A8
2B	AB	86	4A A9
2C	AC	86	4B AA
2D	AD	87	4C AB
2E	AE	87	4D AC
2F	AF	88	4E AD
30	B0	88	4F AE
31	B1	89	50 AF
32	B2	89	51 B0
33	B3	8A	52 B1
34	B4	8A	53 B2
35	B5	8B	54 B3
36	B6	8B	55 B4
37	B7	8C	56 B5
38	B8	8C	57 B6
39	B9	8D	58 B7
3A	BA	8D	59 B8

JIS	EUC	シフトJIS第 1 バイト	シフトJIS第 2 バイト
3B	BB	8E	5A B9
3C	BC	8E	5B BA
3D	BD	8F	5C BB
3E	BE	8F	5D BC
3F	BF	90	5E BD
40	C0	90	5F BE
41	C1	91	60 BF
42	C2	91	61 C0
43	C3	92	62 C1
44	C4	92	63 C2
45	C5	93	64 C3
46	C6	93	65 C4
47	C7	94	66 C5
48	C8	94	67 C6
49	C9	95	68 C7
4A	CA	95	69 C8
4B	CB	96	6A C9
4C	CC	96	6B CA
4D	CD	97	6C CB
4E	CE	97	6D CC
4F	CF	98	6E CD
50	D0	98	6F CE
51	D1	99	70 CF
52	D2	99	71 D0
53	D3	9A	72 D1
54	D4	9A	73 D2

JIS	EUC	シフトJIS第 1 バイト	シフトJIS第 2 バイト
55	D5	9B	74 D3
56	D6	9B	75 D4
57	D7	9C	76 D5
58	D8	9C	77 D6
59	D9	9D	78 D7
5A	DA	9D	79 D8
5B	DB	9E	7A D9
5C	DC	9E	7B DA
5D	DD	9F	7C DB
5E	DE	9F	7D DC
5F	DF	E0	7E DD
60	E0	E0	80 DE
61	E1	E1	81 DF
62	E2	E1	82 E0
63	E3	E2	83 E1
64	E4	E2	84 E2
65	E5	E3	85 E3
66	E6	E3	86 E4
67	E7	E4	87 E5
68	E8	E4	88 E6
69	E9	E5	89 E7
6A	EA	E5	8A E8
6B	EB	E6	8B E9
6C	EC	E6	8C EA
6D	ED	E7	8D EB
6E	EE	E7	8E EC
6F	EF	E8	8F ED

JIS	EUC	シフトJIS第 1 バイト	シフトJIS第 2 バイト
70	F0	E8	90 EE
71	F1	E9	91 EF
72	F2	E9	92 F0
73	F3	EA	93 F1
74	F4	EA	94 F2
75	F5	EB	95 F3
76	F6	EB	96 F4
77	F7	EC	97 F5
78	F8	EC	98 F6
79	F9	ED	99 F7
7A	FA	ED	9A F8
7B	FB	EE	9B F9
7C	FC	EE	9C FA
7D	FD	EF	9D FB
7E	FE	EF	9E FC