F*TRAN+

V 3.0

操作説明書/解説編

第1版

株式会社 富士通ビー・エス・シー

はじめに

F*TRAN+(エフトラン・プラス)は、汎用機/オフコン/Unixなどのホストのファイル転送データと、パソコンの標準であるWindowsファイルとのデータ交換をする汎用性の高いファイル変換ユーティリティです。Windowsファイル間のデータ変換もできます。

Windows 98/95 または Windows NT /2000上の 32 ビットアプリケーションとして動作します。

ソースプログラム、バイナリファイル、ランダムファイル、プリント形式ファイルなどを変換する基本機能に加え、コンマ区切り(CSV)形式対応など、市販ソフトとのデータ交換に適した強力なデータ加工・編集機能を備えています。また、各種漢字コードに対応し、拡張漢字にも本格対応しています。さらに、COBOLの数値項目等(ゾーン形式、パック形式、2進形式、BCD形式)にも対応しています。

F*TRAN+は、さまざまなホストとパソコンの連携利用を強力に支援します。その高い汎用性、高性能、高機能が有効に活用されることを願ってやみません。

F*TRAN+のマニュアルには、導入編、解説編(本書)、コマンド編、マルチレコード編、 プログラム応用編があります。解説編(本書)の構成はつぎのとおりです。

第1章 概 要

F*TRAN+の概要を説明しています。

第2章 変換の仕様

各種ファイル変換、コード変換、数値変換などの仕様について説明しています。

第3章 操作の基礎

操作方法についての基礎的な知識を説明しています。

第4章 操作

F*TRAN+の操作について説明しています。

目 次

<u> </u>	1	立	+01	-
æ		T	MP4	
77		=	11476	34

1.1	F * T R A N + とは	. 2
1.2	ホストファイルとは	. 4
1.3	Windowsのファイル(DOS形式)とは	. 5
1.4	3 つのファイル変換機能(汎用機・オフコン系)	. 6
1.5	3 つのファイル変換機能(Unix系)	. 8
1.6	3 つのファイル変換機能(Windows系)1	
1.7	補助機能1	2
1.8	セットアップ機能1	4
1.9	特長のまとめ1	6
	第2章 変換の仕様	
2.1	漢字シフトコードKI/KOについて2	0
2.2	ファイル変換機能の使い分け2	2
2.3	ファイル変換機能の使い方のポイント2	4
2.4	ホスト側の扱えるデータ形式2	6
2.5	Windows側の扱えるデータ形式2	. 8
2.6	テキストファイル変換の仕様3	0
2.7	データファイル変換1/プリント形式の仕様3	6
2.8	データファイル変換 2 / デリミタ形式の仕様4	. 4
2.9	ランダムファイル変換の仕様5	2
2.10	A N K 変換5	5 8
2 .11	漢字変換6	
2 .12	漢字変換方式のサポートパターン	
2 .13	ホスト漢字のレパートリー6	5 4
2 .14	漢字コード変換の仕組み	
2 .15	漢字対応表について	
2 . 16	変換できないホストの拡張漢字の扱い6	
2 .17	ホストの一般漢字と拡張漢字の区別	7 0
2 .18	ホストの拡張漢字の領域	
2 .19	変換できないWindowsの拡張漢字の扱い	
2.20	数值变換	
	ゾーン形式(EBCDIC系)	
2.22	パック形式(EBCDIC系)	7

2.23	ゾーン形式(JIS8系)7	8
2.24	パック形式(JIS8系)7	9
2.25	Windows COBOLのゾーン / パック形式8	0
2.26	B C D形式8	2
2.27	2 進形式8	3
2 .28	パック / BCD / 2進形式の桁数とバイト数の換算8	4
2.29	ホスト Winテキストファイル変換のしくみ8	6
2.30	ホスト Winデータファイル変換のしくみ8	8
2.31	ホスト Winランダムファイル変換のしくみ9	0
2.32	Win ホストテキストファイル変換のしくみ9	
2.33	Win ホストデータファイル変換のしくみ9	
2.34	Win ホストランダムファイル変換のしくみ9	6
	第3章 操作の基礎	
3 . 1	ファイルの指定100	Λ
3.1	式の指定107	
3.3	ピクチャの指定104	
3.4	2 進ピクチャの指定100	
3.5	日付データの指定	
3.3	L117	_
	第4章 操 作	
4 . 1	F * T R A N + 起動と終了1 1 2	2
4.2	ホスト Winファイル変換のファイル指定と共通オプション114	4
4.3	ホスト Winテキストファイル変換124	4
4 . 4	ホスト Winデータファイル変換132	2
4 . 5	ホスト Winランダムファイル変換160	0
4.6	Win ホストファイル変換のファイル指定と共通オプション194	4
4 . 7	Win ホストテキストファイル変換204	4
4 . 8	Win ホストデータファイル変換214	4
4 . 9	Win ホストランダムファイル変換248	
4.10		
4 . 11	Winファイラ28	
4 . 12	変換設定29	
4 .13	漢字対応表エディタ3 1	
4 . 14		
4.15	ヘルプ3 3	2

付録

エラーメッセージ (共通)

本書で用いる表記法

本文と画面のパラメータ類の表記法

 $\{A \mid B \mid C\}$ A、B、またはCのうち、どれか1つを選択します。

省略はできません。

A B

同上。

C

(A/B/C) 同上。

[A] Aは省略できます。

[A/B/C] A、B、またはCのうち、どれか1つを選択します。

省略が可能で、その場合、下線を引いたBを選択した

ものとみなします。

A B C

同上。

(A / [B] / C) 同上。ただし、[] でくくった B を選択したものと

みなします。

X ・・・ X類を A B C のように列挙します。

n、nn、<n> 10進数を指定します。

(< > は表記上の記号で、入力はしません)

xxH 16進でxxです。Hを省くこともあります。

改行を意味します。リターンキーのシンボルです。

<u>a</u> 下線部を入力します。

abc 下線部を入力し、リターンキーを押します。

CTRL+A コントロール(CTRL)キーを押しながら、Aキー

を押します。コントロールAと読みます。

^ A 同上。

d: ドライブA:やC:など、任意のドライブ指定を表し

ます。

注意 ---- 実画面と少し差異がある

本書に示す画面と実際の画面には、若干の差異がある場合があります。あらかじめ、ご了承ください。

第1章

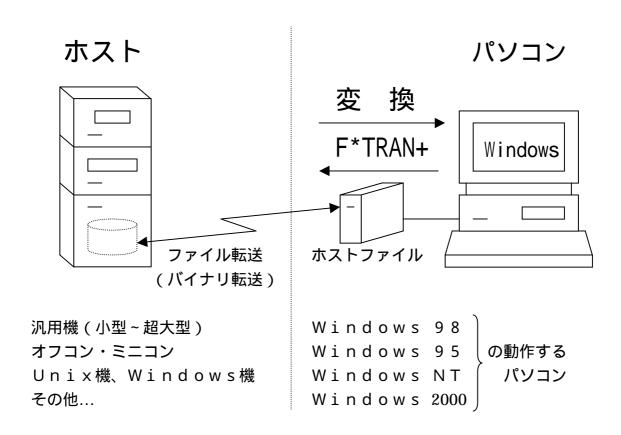
概要

1.1 F*TRAN+とは

F*TRAN+は、汎用機やオフコン、Unix機などのホストのファイル転送データとパソコンのデータ交換を実現するファイルトランスレータです。Windows98/95、WindowsNT/2000上で動作し、オンライン・データとWindowsファイルの相互変換ができます。Windowsファイル間のデータ変換もできます。

さまざまなホストに対応

F * T R A N + の対象とするデータは、基本的にホストのコード体系のままでファイル転送 (バイナリ転送)したデータになります。F * T R A N + を使えば、さまざまなホストとのデータ交換がパソコン上で簡単にできるようになります。



オンラインファイル転送の現状

汎用機中心の時代には「コンピュータどうしのデータ交換は、磁気テープかオンライン接続で行う」のが常識とされていました。現在、それがパソコンとのデータ交換にも当てはまるでしょうか。

磁気テープについては不問とします。パソコンにはつなげませんから(実際には磁気テープを つなげるシステムもありますが、非常に高価です)。

ではオンライン接続なら……。たしかに、遠隔地と即時でデータ交換できる利便は計り知れないものがあります。事実、オンラインネットワークのない社会など、もう考えられません。しかし、ファイル転送に限れば今のところ欠点がめだちます。

オンラインファイル転送の欠点 ホストとのオンライン接続によるファイル転送は、現状では

データ加工能力が不十分

といえます。テキスト転送とバイナリ転送ぐらいしかできない場合が多く、まだ汎用性が低い のです。

F * T R A N + によるファイル変換の利点

そこで、オンラインファイルを扱う F * T R A N + はつぎのような利点を持ちます。まず、最大の特長は、

メーカーの壁を越え、さまざまなホストに対応でき、 データ加工能力が高く、Windowsのさまざまなソフトに対応できる

ことです。汎用性が高いのです。また、 変換速度が速く、自動化も容易

です。このため、運用コストが安くすむのも大きな利点です。このように、オンラインファイル転送の欠点を補う特長を持っています。そして、F*TRAN+は、

ウインドウ上の項目を設定するだけで操作することも コマンド行方式でバッチ処理として実行することも

できます。言い替えれば、

ウインドウ操作による使いやすさ・覚えやすさ コマンド行方式による定形業務の自動化のしやすさ

を両立させているのです。また、 マルチファイル対応

です。ファイル数が多くてもファイル容量が大きくても問題ありません。

パソコンを含め複数メーカーの機種が混在している"マルチベンダ環境"では、F*TRAN + は上手に使いこなしたいソフトウェアです。

1.2 ホストファイルとは

F*TRAN+の「ホストファイル」とは、つぎのファイルを指します。

F*TRAN+では、ホスト端末エミュレータによりファイル転送(バイナリ転送)された、または、するファイルを「ホストファイル」として定義しています。その他、Windows間のデータ変換の際には、Windowsファイルをホストフィルとして扱うこともできます。 つぎの3つのタイプに分けることができます。

汎用機・オフコン系(汎用機またはオフコン):

ホストのコード体系により構成された、 固定長レコードのWindowsファイル(DOS形式) (レコード長情報を持たないため、ユーザーがレコード長を指定する必要あり)

Unix系(Unix機):

Unixのコード体系により構成された、 可変長/固定長のWindowsファイル(DOS形式)

Windows系(Windowsの動作するパソコン):

Windowsのコード体系により構成された、 可変長/固定長のWindowsファイル(DOS形式)

1.3 Windows のファイル (DOS 形式)とは

パソコンの標準 ---- DOS形式

インテル80×86系のCPUを搭載したパソコンの標準OSとなった、マイクロソフト社の DOS。そのDOSのディスク・ファイル管理形式をDOS形式と呼びます。もはや、DOS形 式はパソコンの標準形式といえるでしょう。

DOSの呼び方は、MS-DOS、日本語MS-DOS、英語MS-DOS、日本語DOS、IBMDOS、PC DOS、DOS/Vと、メーカーによりさまざまですが、基本的には同じ DOSです。出元が同じであるため、DOS形式のディスク・ファイルにはメーカーの壁を越えた互換性があります。現在では、コツさえつかんでいればパソコンどうしのデータ交換であまり 不自由することはありません。

DOS形式の特長

DOS形式には、今述べた互換性の高さ以外に、

バイトストリーム指向で、スペース効率がよい 階層的ディレクトリ管理ができる ファイル名に拡張子をつけることができ、整理が容易である 動的な空き領域管理ができる

などの特長があります。

ロングファイル名

DOS形式では、ファイル名(8文字以内). ファイル拡張子(3文字以内)という制約がありましたが、Windows98/95、WindowsNT/2000では、最高255文字までのファイル名(複数のピリオド[.]の使用も可)を使うことができるようになりました。

F*TRAN+では、DOS形式をベースにロングファイル名をサポートしたファイルをWindowsファイルと表現しています。

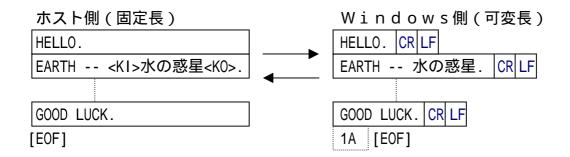
1 . 4 3 つのファイル変換機能(汎用機・オフコン系)

F*TRAN+には、3つの(実質的には4つの)ファイル変換機能があり、

と呼んでいます。これらを目的に応じて使い分けます。どれもホスト Windows双方向サポートです。

テキストファイル変換

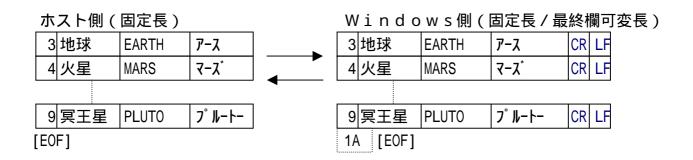
おもに、ソースプログラムの変換に使います。生データの変換やリストファイルの変換にも使えます。



データファイル変換1/プリント形式

おもに、ホストのCOBOLデータと、市販ソフトの入出力用テキストファイルの変換に使います。

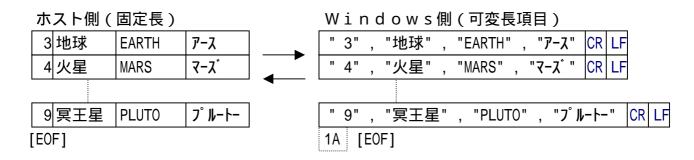
固定長項目、区切り文字なしのテキストファイルを、項目別に分けて変換できます。もちろん レコード全体で1項目とみなす、単純な変換もできます。



データファイル変換2/デリミタ形式

おもに、ホストの C O B O L データと、市販ソフトの入出力用テキストファイルの変換に使います。

可変長項目、区切り文字(デリミタ)つきのテキストファイルを項目別に分けて変換でき、コンマ区切り形式、タブ区切り形式、スペース区切り形式の3タイプが扱えます。



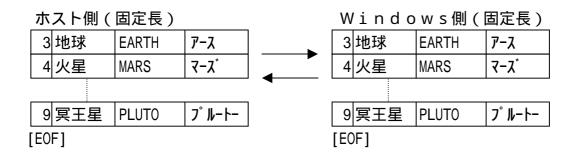
一般に、市販ソフトで「テキストファイルの入出力ができます」といえば、以上のうちコンマ 区切り形式か、プリント形式ファイルを指すのがふつうです。

とくに、コンマ区切り形式は、パソコンのデータベースや表計算ソフトの最も標準的な入出力ファイル形式で、CSV(Comma Separated Value)形式という特別な名前がついています。この形式は、市販ソフトへの入力操作がとても簡単なのが特長です。

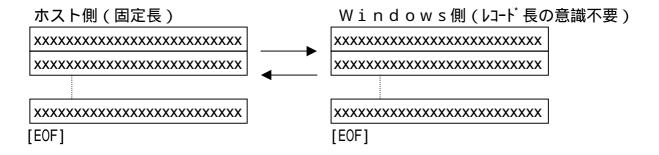
ランダムファイル変換

おもに、Windows COBOLの順ファイルやBASICのランダムファイルの変換に使います。

項目別の変換ができる点では、データファイル変換1/プリント形式とよく似ています。しかし、改行コードも区切り文字もなく、固定長で、レコード長を意識しながら扱う必要があります。



バイナリ変換(レコード全体 / 一部を無変換で通すこと)もできます。機械語ファイル等を扱うときに便利です。



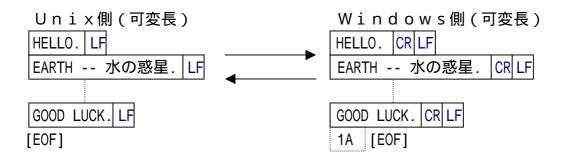
1.5 3 つのファイル変換機能(Unix系)

F*TRAN+には、3つの(実質的には4つの)ファイル変換機能があり、

と呼んでいます。これらを目的に応じて使い分けます。どれもUnix Windows双方向サポートです。

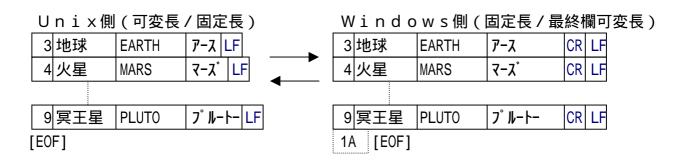
テキストファイル変換

テキストファイル(ソースプログラム等)の変換に使います。生データの変換にも使えます。



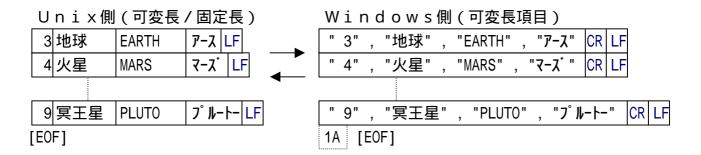
データファイル変換1/プリント形式

おもに、Unixのデータと、市販ソフトの入出力用テキストファイルの変換に使います。 Unixの可変長(プリント/デリミタ形式)または固定長のテキストファイルを、項目別に 分けて変換できます。



データファイル変換2/デリミタ形式

おもに、Unixのデータと、市販ソフトの入出力用テキストファイルの変換に使います。 Unixの可変長(プリント / デリミタ形式)または固定長のテキストファイルを項目別に分けて変換でき、コンマ区切り形式、タブ区切り形式、スペース区切り形式の3タイプが扱えます。



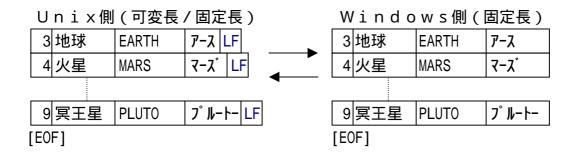
一般に、市販ソフトで「テキストファイルの入出力ができます」といえば、以上のうちコンマ 区切り形式か、プリント形式ファイルを指すのがふつうです。

とくに、コンマ区切り形式は、パソコンのデータベースや表計算ソフトの最も標準的な入出力ファイル形式で、CSV(Comma Separated Value)形式という特別な名前がついています。この形式は、市販ソフトへの入力操作がとても簡単なのが特長です。

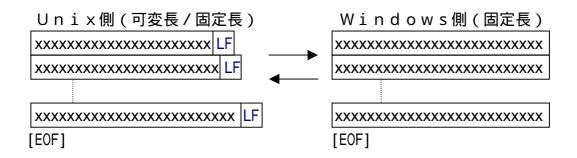
ランダムファイル変換

おもに、Windows COBOLの順ファイルやBASICのランダムファイルの変換に使います。

Unixの可変長(プリント/デリミタ形式)または固定長のテキストファイルを項目別に分けて変換できる点では、データファイル変換1/プリント形式とよく似ています。しかし、改行コードも区切り文字もなく、固定長で、レコード長を意識しながら扱う必要があります。



バイナリ変換(レコード全体/一部を無変換で通すこと)もできます。



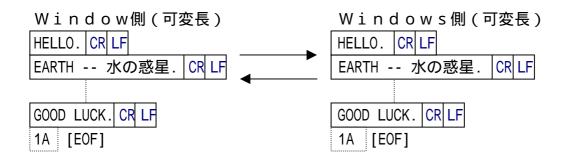
1.6 3つのファイル変換機能(Windows系)

F * T R A N + には、3つの(実質的には4つの)ファイル変換機能があり、

と呼んでいます。これらを目的に応じて使い分けます。どれも、Windowsファイル間の 変換です。

テキストファイル変換

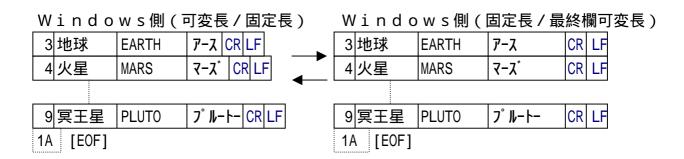
テキストファイルの加工に使います。生データの加工にも使えます。



データファイル変換1/プリント形式

おもに、Windows市販ソフトの入出力用テキストファイルの加工に使います。

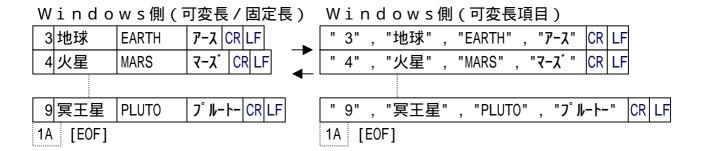
Windowsの可変長(プリント / デリミタ形式)または固定長のテキストファイルを、項目別に分けて変換できます。



データファイル変換2/デリミタ形式

おもに、Windows市販ソフトの入出力用テキストファイルの加工に使います。

Windowsの可変長(プリント / デリミタ形式)または固定長のテキストファイルを項目別に分けて変換でき、コンマ区切り形式、タブ区切り形式、スペース区切り形式の3タイプが扱えます。



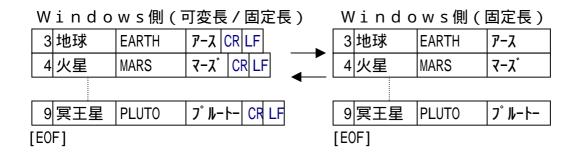
一般に、市販ソフトで「テキストファイルの入出力ができます」といえば、以上のうちコンマ 区切り形式か、プリント形式ファイルを指すのがふつうです。

とくに、コンマ区切り形式は、パソコンのデータベースや表計算ソフトの最も標準的な入出力ファイル形式で、CSV(Comma Separated Value)形式という特別な名前がついています。この形式は、市販ソフトへの入力操作がとても簡単なのが特長です。

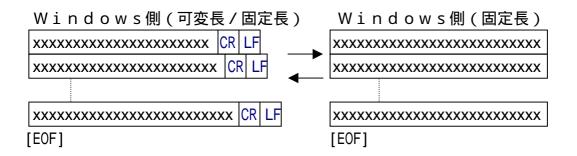
ランダムファイル変換

おもに、Windows COBOLの順ファイルやBASICのランダムファイルの加工に使います。

Windowsの可変長(プリント/デリミタ形式)または固定長のテキストファイルを項目別に分けて変換できる点では、データファイル変換1/プリント形式とよく似ています。しかし、改行コードも区切り文字もなく、固定長で、レコード長を意識しながら扱う必要があります。



バイナリ変換(レコード全体/一部を無変換で通すこと)もできます。



1 . 7 補助機能

F*TRAN+は、補助機能を提供しています。ホストファイルやWindowsファイルを管理します。

Winファイラ

Winファイラはつぎの機能を持っています。

ファイル名一覧

ホストファイルやWindowsファイルの一覧を見ることができます。

ファイルの削除

ホストファイルやWindowsファイルを削除できます。

ファイル編集

ホストファイルやWindowsファイルの中身の表示・修正ができるファイルエディタです。

1.8 セットアップ機能

F*TRAN+は、このセットアップ機能でさまざまなホストへの対応を実現しています。コード変換の方法を設定する機能がその中心です。F*TRAN+を購入しインストールをすませたら、必ずセットアップの作業をしてください。セットアップはホスト選択と変換設定により行います。

ホスト選択

ホスト系を、汎用機・オフコン系(汎用機またはオフコン) Unix系(Unix機) Windows系(Windows機)の3つから選択し、使用するホストに適するコード変換表を選択します。

ANKコードの設定

ホストが汎用機・オフコンの場合、ホストで採用しているANKコード系がEBCDICコードかJIS8/ASCIIコードかを設定します。この設定はANK変換のときだけでなく、バッファのクリア、ゾーン形式やパック形式の数値の変換、その他でも参照される、とても重要なものです。

EBCDICコードのカタカナ版/英小文字版の切り替えもできます。

漢字変換方式の設定

ホストで採用している漢字体系がどの方式かを設定します。ふつうは、F * T R A N + が標準提供している漢字変換方式のなかから、どれか1つを選ぶだけですみます。

各漢字変換方式の設定内容はすべて修正可能です。たとえば、漢字対応表に定義されない拡張 漢字の扱い方の変更などができます。

漢字対応表の設定

ホストの拡張漢字や外字(ユーザー定義文字)などを、利用者の意図した漢字に割り当てるための漢字対応表を設定します。漢字対応表に登録された内容が変換時の漢字コードに反映されます。 ふつうは、標準提供の漢字対応表を利用します。

ホストCOBOLベンダの設定

ホスト上で使用しているCOBOLがどこのベンダかを設定します。この設定は、Unix、WindowsのCOBOLデータと、<math>Windows COBOLデータの変換をする場合に重要になります。

Windows COBOLベンダの設定

Windows上で使用しているCOBOLがどこのベンダかを設定します。この設定は、Windows COBOLで使用するデータを変換する場合に重要になります。

ホストエンディアンの設定

ホストの2進数値項目の格納順序(正順、逆順)を設定します。この設定は、ホストのCOBOLの2進数値項目データ等を変換する場合に重要になります。

コメントの変更

設定を変更したら、適当なコメントをつけておくことができます。

ANK变換表

多くのホストはEBCDICコードを採用しています。一方、パソコンはJIS8/ASCI Iコードです。両者のコード変換の方法には確固とした標準がありません。

そこで、F*TRAN+ではほぼ標準的と思われるANK変換表を提供し、それを利用者が個々の事情に合わせて修正できるようにしています。修正操作はとても簡単です。

< コード変換表 >

以上の設定は、ファイルに保存できます。それを「コード変換表(ファイル)」と呼んでいます。拡張子は.CCT(Code Conversion Table)で、F*TRAN+の動作を決める重要なファイルです。複数のホストとのデータ交換が必要なときは、ホストの種類に応じてこれを何とおりかにセットアップしておき、使い分けることができます。

1.9 特長のまとめ

F*TRAN+の特長を、以下にまとめました。

ファイル変換が速く、パワフルである

ファイルの変換速度が高速です。マルチファイル処理機能と、ハードディスク上での変換のため、大量・大容量のファイルが容易に扱えます。

実質4とおりのファイル変換機能をもつ

テキストファイル変換、データファイル変換、ランダムファイル変換と呼ぶ3とおりの変換ができます。データファイル変換は、さらにプリント形式のファイルの変換とCSV形式に代表されるデリミタ形式のファイルの変換に分けられます。つまり、実質4とおりのファイル変換機能が用意されおり、それらを目的に応じ柔軟に使い分けできます。これにより、標準的なエミュレータのテキスト転送ではできない変換の要求に対応することができます。

マルチファイル対応で、ワイルドカード文字が使える

Windowsファイルの指定にも、ホストファイルの指定にもワイルドカード文字(*、?)が使え、マルチファイル処理ができます。関連ファイルの一括処理が容易です。ワイルドカードは、ファイル名が固定していないとき、短い指定ですませたいときにも役に立ちます。

問い合わせモードがある

ファイルの変換を実行するかどうか問い合わせるモードがあります。ワイルドカード文字を使ったとき、ファイル名を確認しながら安全に処理を進めることができます。

ウインドウ設定方式とコマンド行方式を両立させている

ウインドウ設定方式では、ウインドウ上の項目を選択・入力するだけの簡単な操作での実行が可能です。試行錯誤が容易で、初心者にも向いています。また、コマンド行方式で使えば、定形処理のバッチファイル化が容易に行えます。Visual Basicなどからも呼び出すことができ、定形業務化や組み込み用途にも向いています。終了コード(エラーレベル)で正常・異常の判断ができ、バッチファイルのなかである程度のエラー処理ができます。

OLEオートメーション対応

OLEオートメーションに対応していますので、<math>F*TRAN+をユーザアプリケーションの「部品」として使用することができ、より細かいコントロールをすることができます。

スクリプトファイル機能をサポート

一度、設定/実行した変換作業をスクリプトファイルとして保存しておくことができます。このスクリプトファイルを使って、試行錯誤(変換設定の修正)や自動実行(エクスプローラ等からスクリプトファイルを起動して変換を実行させること)も容易にできます。

パワフルで使いやすいレコードレイアウト設定機能

マップ設定オプションで細かい変換方法を指示できます。マルチレコードレイアウトもサポートしています。設定はウインドウ上の項目を選択・入力する方式ですので、簡単に行うことができます。設定時にホストファイルやWindowsファイルの内容を確認することができます。さらに、設定に従って変換をした場合の結果をシミュレートすることもできます。

コード変換表をカスタマイズできる

ANK変換表や漢字変換方式などを設定するコード変換表を、利用者が自由にカスタマイズできます。自社用にカスタマイズしたコード変換表をいくつか作っておき、それを切り替えながら使うこともできます。

各種漢字変換方式をサポートしている

さまざまな漢字変換方式をサポートしています。KI(漢字イン)/KO(漢字アウト)があるときでも、KI/KOがないときでも、それぞれに適した変換ができます。

漢字変換方式として、各種の方式をコード変換表に登録しておくことができ、きめ細かな調整ができます。

漢字変換のレパートリーは、富士通JEF(12ポイント/9ポイント) 富士通F9450 内部コード、日立KEIS、NEC内部コード(E/J) NEC JIPS(E/J) 東芝漢 字、三菱MELCOM漢字、日本ユニシスLETS-J、カシオ漢字、新JIS(83年版) 旧JIS(78年版) IBM漢字、EUC、シフトJISなどです。

F*TRAN+のサポートする基本パターンに合うなら、さらに新しい漢字変換方式を追加登録することもできます。

拡張漢字変換をサポートしている

ホストの拡張漢字(ユーザー定義文字を含む)にも対応しており、おもなホストベンダ(富士通、日立、NEC、東芝、三菱、日本ユニシス、日本IBM、サンマイクロ、日本HP 他)の漢字対応表を標準提供しています。漢字対応表は漢字対応表エディタ上で編集できるので、利用者の意図した漢字を対応づけることができます。漢字対応表に定義されない範囲外の漢字は、ゲタ変換、クエスチョン変換、ゼロイー変換と呼ぶ3とおりの変換方式のなかから1つを選んで変換できます。

ホストのゾーン形式、パック形式をサポートしている

ホストのCOBOLの内部数値形式であるゾーン形式(符号なし、符号つき)パック形式(符号なし、符号つき)の変換をサポートしています。それらと文字形式数値の変換ができます。小数点もサポートしています。

JIS8系(東芝のシステム)のゾーン形式、パック形式にも対応しています。

POS端末採用のBCD形式数値をサポートしている

POS端末などで採用しているBCD形式数値(符号なし)をサポートしています。それらと 文字形式数値の変換ができます。

Windows COBOLのゾーン形式、パック形式をサポートしている

Windows COBOLのゾーン形式(符号なし、符号つき)パック形式(符号なし、符号つき)の変換をサポートしています。それらとホストのゾーン形式(符号なし、符号つき)パック形式(符号なし、符号つき) BCD形式数値(符号なし)の変換ができます。

日付データ変換をサポートしている

日付データを指定フォーマット(YYYYMMDD、YY-MM-DDなど)で編集して変換する機能をサポートしています。西暦年下2桁、和暦年のデータを指定範囲の西暦年4桁に拡張する機能もあり、西暦2000年対応にも幅広く活用できます。

2 進数値変換をサポートしている

COBOL、Visual Basic、C/C++などで定義できる2進数値をサポートしています。それらと文字形式、ゾーン/パック形式数値の変換ができます。

バイナリ変換をサポートしている

バイナリ変換をサポートしています。機械語ファイル等を扱うときなどに重宝します。 また、ファイルの内容が独特の構成でふつうの変換方法が適用できないときに、最後の手段と してバイナリ変換を使うこともできます。

Windowsファイルの編集/加工ができる

プリント形式ファイル(可変長/固定長)デリミタ形式ファイル(可変長)ランダムファイル(固定長)間のデータ編集/加工が自由にできます。プリント形式からデリミタ形式への形式変換や、デリミタ形式の再編集などは他のアプリケーションとの連携を密にします。

ファイル変換中に処理の中断ができる

ファイル変換中に中止ボタンをクリックすれば、処理を中断することができます。

Winファイラ、Winファイルエディタを内蔵している

Winファイラは、ホストファイルやWindowsファイルの一覧を見ることができ、ホストファイルやWindowsファイルの削除もできます。WinファイラからWinファイルエディタを起動できます。

Winファイルエディタを使えば、ホストファイルやWindowsファイルの内容表示・修正ができます。マップ設定オプションと並列に動作させ、データの確認をしながらレコードレイアウトを設定することもできます。

第2章

変換の仕様

2.1 漢字シフトコードKI/KOについて

以降の説明で、頻繁にKI(漢字イン) KO(漢字アウト)という用語がでてきます。単に KI/KOと表記しますが、F*TRAN+は実際にはホストの3つの漢字シフト方式をサポートしています。ホストが汎用機・オフコン系(汎用機またはオフコン)の時に多種多用に使用されています。

KI/KO方式

狭い意味のKI/KO方式のことで、漢字文字列をKIとKOではさみます。

一番多く使われている方式です。

ANK文字列 <ki></ki>	漢 字	文 字 列	< K O >	ANK文字列
------------------	-----	-------	---------	--------

KI+長さバイト方式

東芝のシステムで使われている方式です。漢字文字列のおわりを示す K O がない代わりに、K I のあとにその長さ(通常 2 ~ 2 5 4)を示すバイトをつけます。

この方式は、少し取り扱いが難しいのが欠点です。



NEC暗黙シフト方式

NECの内部コード(E/J)で使われている方式です。KI/KOを使いません。文字コードを調べて、それがANK文字か漢字の第一バイトかを判定します。

パソコンのシフトJISコードと、少し似ていますが、それほど使いやすくはありません。また、英小文字と数個の特殊文字が使えないという大きな欠点も持っています。

EUCシフトコード

Unix系のオペレーティングシステムでは、漢字コードとして EUC(Extended Unix Code)が使われる場合がほとんどですが、EUCには 1 バイトの半角コードと、 2 バイトの漢字コードの他に、つぎのコードが存在します。

半角カナコード(2バイト) : 8 E H + A 1 H ~ D F H

拡張漢字コード(3バイト) : 8FH + A1A1H~FEFEH

F*TRAN+では、この 8EH、8FH をEUCシフトコードとして認識し、処理しています。

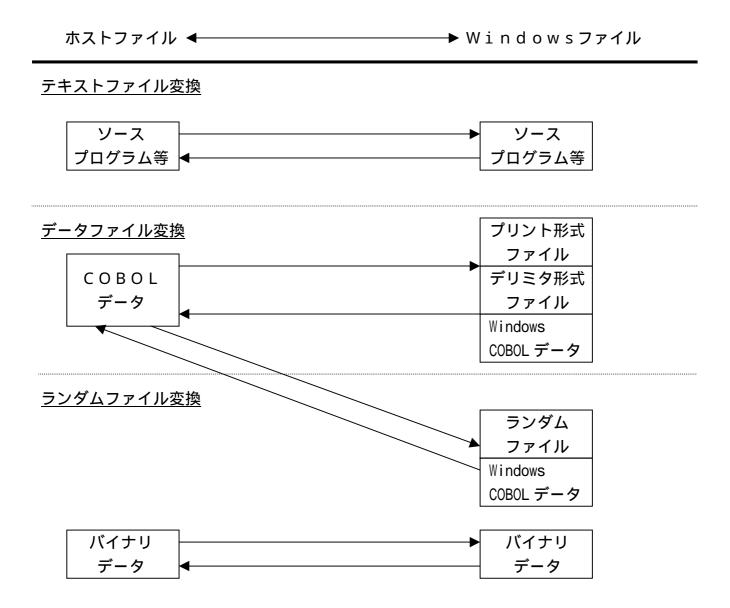
2 . 2 ファイル変換機能の使い分け

F * T R A N + では、3つの(実質4つ)のファイル変換機能、

テキストファイル変換 データファイル変換 1 / プリント形式 データファイル変換 2 / デリミタ形式 ランダムファイル変換

を使い分けることができます。"うまく使い分けなければいけない"といったほうが、正しいかもしれません。

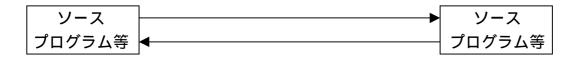
典型的なデータ形式 (ホストが汎用機・オフコンの場合)



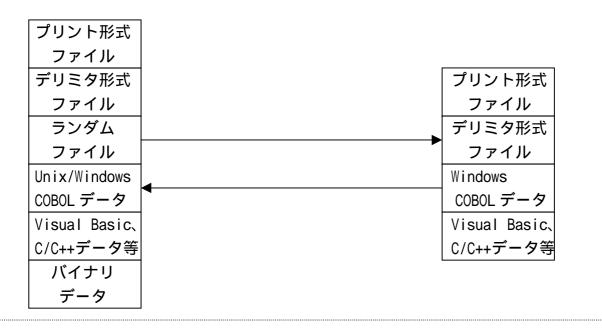
典型的なデータ形式 (ホストがUnix、Windowsの場合)

ホストファイル ◆ Windowsファイル

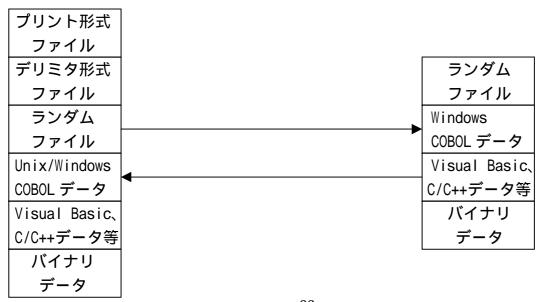
テキストファイル変換



データファイル変換



ランダムファイル変換



2.3 ファイル変換機能の使い方のポイント

テキストファイル変換

テキストファイル変換は、おもに、ソースプログラムの変換に使います。

ホスト Winテキストファイル変換

ホストが汎用機・オフコンの場合、コードがAnkかAnk・漢字(KI/KO付)混在かを 指定するコード設定オプションが重要です。

Win ホストテキストファイル変換

ホストが汎用機・オフコンの場合、コードがAnkかAnk・漢字(KI/KO付)混在かを 指定するコード設定オプションがとくに重要です。

Winファイル側の各レコード末尾に、必ず改行コード(CR/LF=0D0AH)がついていなければいけません(ただし、最終レコードに限り改行コードがなくてもよい)。

データファイル変換1/プリント形式

データファイル変換 1 / プリント形式は、おもに、ホストのCOBOLのデータと市販ソフトの入出力データ形式(プリント形式 固定長テキスト)の変換に使います。

ホスト Winデータファイル変換

項目別に詳細な変換指定をするマップ設定オプションがとくに重要です。

Win ホストデータファイル変換

項目別に詳細な変換指定をするマップ設定オプションがとくに重要です。

Winファイル側の各レコード末尾に、必ず改行コード(CR/LF=0D0AH)がついていなければいけません(ただし、最終レコードに限り改行コードがなくてもよい)。

データファイル変換2/デリミタ形式

データファイル変換 2 / デリミタ形式は、おもに、ホストのCOBOLのデータと市販ソフトの入出力データ形式(デリミタ形式)の変換に使います。コンマ区切り(CSV)形式の変換がとくに重要です。

ホスト Winデータファイル変換

項目別に詳細な変換指定をするマップ設定オプションがとくに重要です。

Win ホストデータファイル変換

項目別に詳細な変換指定をするマップ設定オプションがとくに重要です。

Winファイル側の各レコード末尾に、必ず改行コード(CR/LF=0D0AH)がついていなければいけません(ただし、最終レコードに限り改行コードがなくてもよい)。

ランダムファイル変換

ランダムファイル変換は、おもに、ホストのCOBOLのデータとBASICなどのランダムファイル(ただの固定長ファイル)の変換に使います。バイナリ変換もできます。

ホスト Winランダムファイル変換

項目別に詳細な変換指定をするマップ設定オプション、Winファイル側のレコード長を指定する出力レコード長オプションがとくに重要です。

Win ホストランダムファイル変換

項目別に詳細な変換指定をするマップ設定オプション、Winファイル側のレコード長を指定する入力レコード長オプションがとくに重要です。

2.4 ホスト側の扱えるデータ形式

ホスト側のデータ形式をどう扱えるか示します。の意味はつぎのとおりです。

直接扱える 直接扱えるものと扱えないものがある 直接は扱えない。前処理や後処理が必要

```
ソースプログラム(固定長 *1)
  漢字あり
    サポートしているタイプ
    サポートしていないタイプ
  漢字なし
COBOLの順ファイル(固定長 *1)
  漢字あり
    サポートしているタイプ
    サポートしていないタイプ
    KI/KOあり
    KI/KOなし
  漢字なし
  数値項目(ゾーン形式、パック形式)あり
  数値項目(ゾーン形式、パック形式)なし
  その他の数値項目あり
  その他の数値項目なし
  日付データ項目(数値形式、文字形式)あり
  日付データ項目(数値形式、文字形式)なし
  レコードタイプあり
  レコードタイプなし
COBOLのISAMファイル
データベースからの抽出データ
その他の言語の入出力ファイル
  文字データのみ
  バイナリデータあり
NCデータ(ファナック互換など)
POS端末のBCD形式数値項目
各種バイナリファイル
```

*1) ホストがUnix、Windowsの場合は、可変長/固定長になる

2.5 Windows側の扱えるデータ形式

Windows側のデータ形式をどう扱えるか示します。の意味はつぎのとおりです。

直接扱える 直接扱えるものと扱えないものがある 直接は扱えない。前処理や後処理が必要

ソースプログラム(可変長) テキストファイル形式の文書ファイル 行末に改行コードを入れたもの 行末に改行コードを入れていないもの バイナリ文書ファイル プリント形式ファイル(固定長テキスト形式、SDF形式) デリミタ形式ファイル コンマ区切り(CSV)形式 固定長/可変長項目 全項目引用符くくり 文字項目引用符くくり K3フォーマット(見出しつき) K3フォーマット(データのみ) コンマを含む項目のみ引用符くくり 引用符なし タブ区切り形式(可変長項目) 文字項目引用符くくり

引用符なし

スペース区切り形式(可変長項目)

全項目引用符くくり

文字項目引用符くくり

空白を含む文字項目のみ引用符くくり

引用符なし

SYLK形式ファイル(各種あり)

DIF形式ファイル

ランダムファイル (ただの固定長ファイル)

BASICのランダムファイル

バイナリ数値項目あり(MKI\$/MKS\$/MKD\$)

バイナリ数値項目なし

レコードタイプあり

レコードタイプなし

Windows COBOLの順ファイル/行順ファイル

数値項目(ゾーン形式、パック形式)あり

数値項目 (ゾーン形式、パック形式)なし

その他の数値項目あり

その他の数値項目なし

日付データ項目(数値形式、文字形式)あり

日付データ項目(数値形式、文字形式)なし

レコードタイプあり

レコードタイプなし

Visual Basicのデータ形式ファイル

C / C + + のデータ形式ファイル

各種ISAMファイル

データベースや表計算ソフトの内部形式ファイル

その他の市販ソフトの内部形式ファイル

N C データ (ファナック互換など)

各種バイナリファイル

ダウンロード / アップロードファイル

日付データ(数値形式、文字形式)

•

.

•

2.6 テキストファイル変換の仕様

ホストが汎用機・オフコンの場合

ホストが汎用機・オフコンの場合、テキストファイル変換ではホストファイル(固定長レコード)とWindowsテキストファイル(可変長レコード)間の変換を行います。1レコードが1行に対応する改行コードつきのデータが対象です。ソースプログラムがその典型です。

以下に、テキストファイル変換のコード変換機能、レコード編集機能、EOFコードの扱いを示します。

コード変換機能

テキストファイル変換のコード変換機能は、下表に示すとおり比較的単純です。

コード変換機能

ホスト側コード ◆→ Wir	側コード	変換方法
JIS8/ASCII ↔ JIS	38/ASCII、漢字なし	ANK変換
EBCDIC (カタカナ)◆▶	<i>II</i>	"
EBCDIC(英小文字)◀▶	<i>II</i>	"
ANK・漢字混在 ◆→ JIS	8、シフトJIS漢字混在	ANK・漢字
KI/KOつき		混在変換

注意 ---- KI/KOがついていないと漢字変換はできない

ホスト Winテキストファイル変換のときは、KI/KOがついてないと漢字の変換はできません(NEC内部コード(E/J)を除く)。KI/KOがないときは、代わりにホスト Winデータファイル変換を使って、項目別に分けて変換する必要があります。

レコード編集機能

テキストファイルの代表であるソースプログラムを例に、レコード編集機能を説明します。

ホストのソースプログラムは、レコード長は80バイト(80欄カードイメージ)が中心で固定長である、改行コードに相当するものはない、タブは使わない、などの特徴があります。漢字を使わない限りホスト間での互換性は高いといえます。でも、漢字を使うと、表示イメージと実際の記録形式が違い桁位置がずれていく(KI/KOが入るため) システム間で互換性がなくなる、などの欠点もあります。

一方、パソコン側のソースプログラムは、改行コード(CR/LF=0D0AH)でレコード の区切りとし、可変長である、レコード長にはあまり制限がない、タブ(TAB=09H)が自由に使える、スペース効率がよい、漢字が自由に使えて半角・全角の区別をあまり意識する必要がない、といえます。

ホスト文化とパソコン文化の違い、とでもいったらいいでしょうか。このギャップを埋めるために、かなりのレコード編集機能が必要です。それを右表に示します。

レコード編集機能

機能	説明
改行コード付加	無条件にレコード末尾に改行コードを付加する(ホスト Win)
改行コード削除	無条件にレコード末尾の改行コードを削除する(Win ホスト)
行末圧縮	無条件に行末のスペース類を切り捨てる(ホスト Win)
行末伸長	無条件に行末にスペースを詰める(Win ホスト)
タブ圧縮	連続スペースをタブに置き換える(ホスト Win)
タブ拡張	タブを連続スペースに置き換える(Win ホスト)

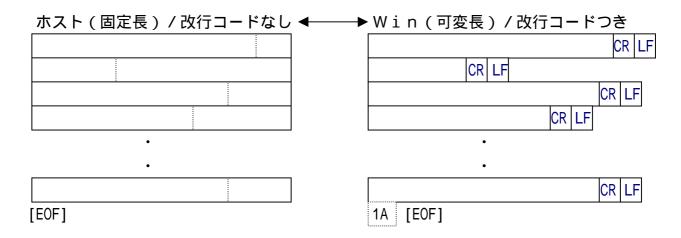
EOFコードの扱い

F*TRAN+のテキストファイル変換では、ホスト Win方向の変換時にはEOFコード (1AH)をつけない、Win ホスト方向の変換時にはEOFコード検出でも物理EOF検出でもEOFにする、というのが省略時の動作です。

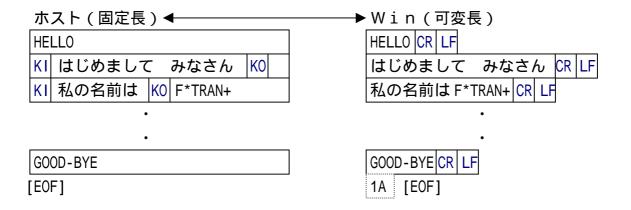
EOFコードの扱い

機能	説明
EOFコード付加	ファイル末尾にEOFコードを付加する・しない(ホスト Win)
EOFコード検出	EOFコード検出でEOFにする(Win ホスト)
EOFコード透過	EOFコードを一般の文字扱いする(Win ホスト)

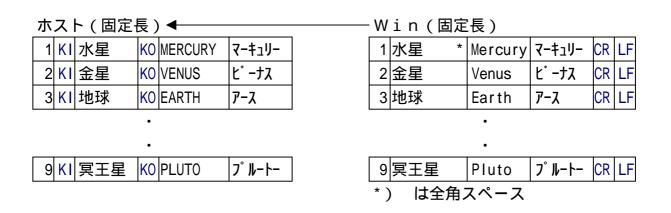
例1)ふつう、固定長 可変長の変換になる



例2)ANKと漢字が混在なら、KI/KOの挿入・削除処理が入る



例3)全角・半角スペースを使い分けていれば、Win ホスト方向の変換にも使える



注意 ---- Win ホストファイル変換時のシフト節約度を弱にすべし

この例3のような変換をするときは、変換設定/漢字変換方式の「Win ホストファイル変換時のシフト節約度」を「弱」に設定してください。そうしないと、KOの位置がレコードごとにずれます。

例4) ANKのみで、Win ホスト方向の変換なら、固定長変換もできる

ホスト (固定長) ◀	Win(固定長)
	CR LF
•	•
•	•
	CR LF

例5) ANKと漢字のみでKI/KOつきのホストファイルは、項目別変換できる

ホスト	(固定長)-			Win(最終項目	可変長)			
Α	В	C		Α	В	C			
	KI	КО						CR	LF
	KI	КО				C	R LF		
	KI	КО				CR L	_F		
	KI	КО						CR	LF
	•				•				
	•				•				
	KI	KO				CR L	_F		

ホストがUnix、Windowsの場合

以下に、テキストファイル変換のコード変換機能、レコード編集機能、EOFコードの扱いを示します。

コード変換機能

テキストファイル変換のコード変換機能は、下表に示すとおり比較的単純です。

コード変換機能

ホスト側コード	◆→ Win側コード	変換方法
JIS8・漢字混在	◆◆ JIS8、シフトJIS漢字混在	ANK・漢字
(EUC、シフト JIS	5)	混在变換固定

レコード編集機能

ソースプログラムを例に、レコード編集機能を説明します。

Unix、Windowsのソースプログラムは、改行コード(UnixはLF=0AH、WindowsはCR/LF=0D0AH)でレコードの区切りとし、可変長である、レコード長にはあまり制限がない、タブ(TAB=09H)が自由に使える、スペース効率がよい、漢字が自由に使えて半角・全角の区別をあまり意識する必要がない、といえます。

レコード編集機能

機能	説明
改行コード正規化	レコード末尾の改行コードを正規化する
行末圧縮	無条件に行末のスペース類を切り捨てる
タブ圧縮	連続スペースをタブに置き換える
タブ拡張	タブを連続スペースに置き換える

EOFコードの扱い

F*TRAN+のテキストファイル変換では、ホスト Win方向の変換時にはEOFコード (1AH)をつけない、Win ホスト方向の変換時にはEOFコード検出でも物理EOF検出でもEOFにする、というのが省略時の動作です。

EOFコードの扱い

機能	説明
EOFコード付加	ファイル末尾にEOFコードを付加する・しない(叔ト Win *)
EOFコード検出	EOFコード検出でEOFにする(Win 杁ト*)
EOFコード透過	EOFコードを一般の文字扱いする(Win 杁ト*)

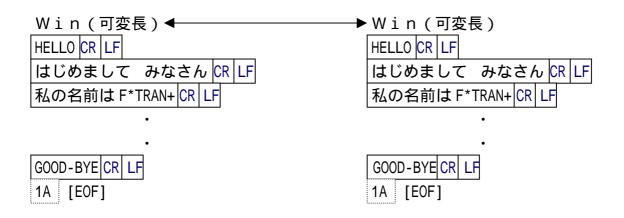
^{*)}ホストがWindowsの場合は、杁ト Win

例)可変長 可変長の変換になる

<ホストがUnixの場合>



<ホストがWindowsの場合>



2.7 データファイル変換1/プリント形式の仕様

ホストが汎用機・オフコンの場合

ホストが汎用機・オフコンの場合、データファイル変換 1 / プリント形式は、おもにホストの COBOL データと、市販ソフトの入出力形式の 1 つであるプリント形式ファイルの変換に使います。

プリント形式ファイルとは、各レコードが改行コードで区切られている固定長のテキストファイルのことです。各項目の桁位置、幅が一定で、固定長・固定欄でなければいけません。なお、 プリント形式ファイルはSDF形式ファイルとも呼ばれています。

例外として、Win ホスト方向の変換のときは、最終項目に限り可変長でもかまいません。

以下に、データファイル変換 1 / プリント形式で可能なデータ変換機能、レコード編集機能、 E O F コードの扱いを示します。

データ変換機能

データファイル変換1/プリント形式のデータ変換機能を下表に示します。

データ変換機能

<u>ノーラを探機能</u>		
ホスト側データ	◆→ Win側データ	変換方法
JIS8/ASCII	◆→ JIS8/ASCII	ANK変換
EBCDIC(カタカナ)	4	"
EBCDIC(英小文字)	4	"
漢字(KI/KOなし)	◆→ シフトJIS漢字のみ	漢字項目変換
ll II	◆ シフトJIS漢字、JIS8混在	"
ANK・漢字混在	◆→ JIS8、シフトJIS漢字混在	ANK・漢字
KI/KOつき		混在变換
文字形式数值	◆▶ 文字形式、Windows COBOL *1	ニューメリック変換 *3
符号なしゾーン形式	+> "	ゾーン変換 *4
符号つきゾーン形式	+> "	"
符号なしパック形式	** "	パック変換 *5
符号つきパック形式	+> "	"
符号なしBCD形式	** "	"
2 進形式数值 *2	** "	2 進数値変換*6
日付データ	◆▶ 日付データ	日付項目変換
特殊データ	◆▶ 特殊データ	特殊データ変換

^{*1)} Windows COBOL の符号なしゾーン形式、符号つきゾーン形式

^{*2)} COBOL の COMP-5 などの形式

^{*3)} Numeric、DispZone、ZoneDisp 変換

^{*4)} ZoneDisp、DispZone、ZoneZone 变換

- *5) PackDisp、DispPack、PackZone、ZonePack 变換
- *6) BinDisp、DispBin、BinZone、ZoneBin 变換

レコード編集機能

プリント形式ファイルは、基本的には固定長ファイルです。その意味では、ホストの標準的なデータ形式と似ています。しかし、項目単位の編集・加工ができないと使いものになりません。 下表に、データファイル変換 1 / プリント形式のレコード編集機能を示します。

レコード編集機能

機能	説明
タブ拡張	タブを連続スペースに置き換える(Win ホスト)
改行コード付加	無条件にレコード末尾に改行コードを付加する(ホスト Win)
改行コード削除	無条件にレコード末尾の改行コードを削除する(Win ホスト)
空行無視	Win側の、改行・改頁コードのみの行を無視する・しない
項目別変換	項目別に分けて変換できる
項目削除	不要な項目を削除できる
項目長変更	項目長を変更できる
項目組み替え	項目の順番の組み替えができる
空項目挿入	空項目を作ることができる
KI/KO挿入	漢字項目の前後にKI/KOを挿入できる(Win ホスト)

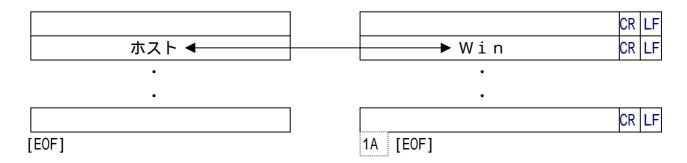
EOFコードの扱い

プリント形式ファイルはテキストファイルの一種ですから、ホスト Win方向の変換時には EOFコード(1AH)をつけない、Win ホスト方向の変換時にはEOFコード検出でも物理EOF検出でもEOFにする、というのが省略時の動作です。

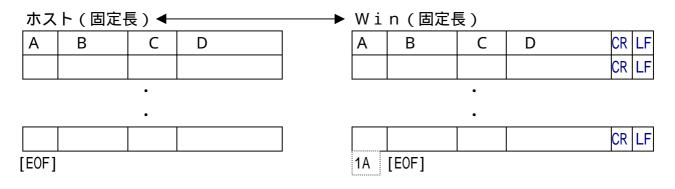
EOFコードの扱い

	機能	説 明
Ī	EOFコード付加	ファイル末尾にEOFコードを付加する・しない(ホスト Win)
	EOFコード検出	EOFコード検出でEOFにする(Win ホスト)
	EOFコード透過	EOFコードを一般の文字扱いする(Win ホスト)

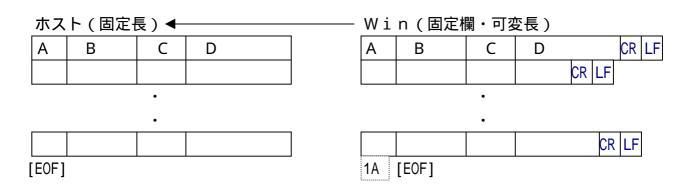
例1)ふつう、固定長 固定長の変換になる



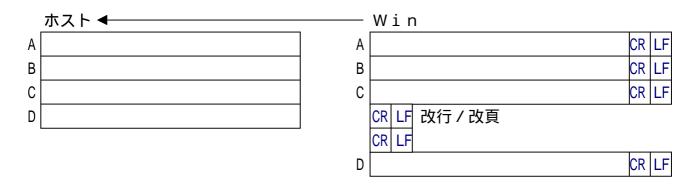
例2)項目別変換ができる



例3)最終項目のみ可変長でもよい(Win ホスト)

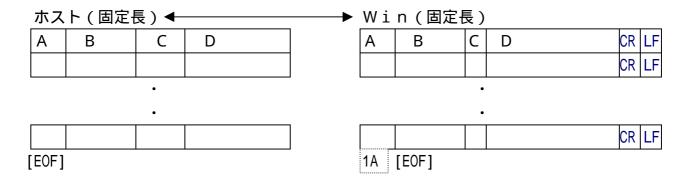


例4) 改行コード、または改頁コード*のみの行は無視できる(Win ホスト)

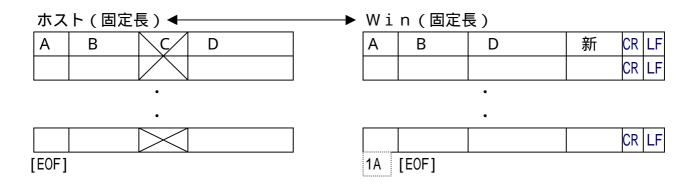


*) 改頁コード = C R / F F (0 D 0 C H) または F F (0 C H)

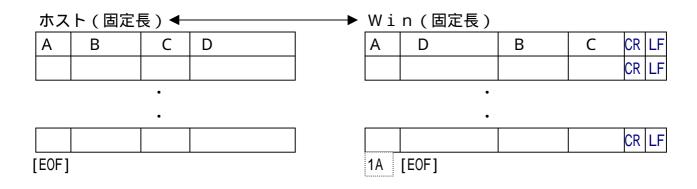
例5)項目長の変更(縮小、拡張)ができる



例6)不要な項目をスキップしたり、逆に空項目を作ったりできる



例7)項目の組み替えができる



ホストがUnix、Windowsの場合

ホストがUnix、Windowsの場合、データファイル変換 1/プリント形式は、ホストのCOBOLデータ等と、市販ソフトの入出力形式の <math>1 つであるプリント形式ファイルの変換に使います。

プリント形式ファイルとは、各レコードが改行コードで区切られているテキストファイルのことです。各項目の桁位置、幅が一定していなければいけません。

以下に、データファイル変換 1 / プリント形式で可能なデータ変換機能、レコード編集機能、 E O F コードの扱いを示します。

データ変換機能

データファイル変換1/プリント形式のデータ変換機能を下表に示します。

データ変換機能

<u> </u>				
ホスト側データ	◆→ Win側ラ	データ	変換方	法
JIS8/ASCII	← ▶ JIS8/	'ASCII	ANK姿	5換
漢字(EUC、シフト JIS)	◆▶ シフトJ∑	IS漢字のみ	漢字項目	目変換
"	← シフトJ]	[S漢字、JIS8混在	"	
JIS8・漢字混在	↔ JIS8、	シフトJIS漢字混在	ANK ·	漢字
			混在	E変換
文字形式数值	◆◆ 文字形式、	Windows COBOL *1	ニューメリック	变換 *3
符号なしゾーン形式	↔	<i>II</i>	ゾーン変	で換 *4
符号つきゾーン形式	↔	<i>II</i>	"	
符号なしパック形式	↔	<i>II</i>	パック変	变换 *5
符号つきパック形式	↔	<i>II</i>	"	
符号なしBCD形式	↔	<i>II</i>	"	
2 進形式数值 *2	↔	<i>II</i>	2 進数値	直変換*6
日付データ	◆→ 日付データ	7	日付項目] 変換
特殊データ	◆▶ 特殊データ	7	特殊データ	9変換

- *1) Windows COBOL の符号なしゾーン形式、符号つきゾーン形式
- *2) COBOL の COMP-5 などの形式
- *3) Numeric、DispZone、ZoneDisp 変換
- *4) ZoneDisp、DispZone、ZoneZone 变換
- *5) PackDisp、DispPack、PackZone、ZonePack 变換
- *6) BinDisp、DispBin、BinZone、ZoneBin 变換

レコード編集機能

プリント形式ファイルは、基本的には固定長ファイルです。しかし、項目単位の編集・加工ができないと使いものになりません。下表に、データファイル変換 1 / プリント形式のレコード編集機能を示します。

レコード編集機能

機能	説明
タブ拡張	タブを連続スペースに置き換える
改行コード正規化	レコード末尾の改行コードを正規化する
空行無視	改行・改頁コードのみの行を無視する・しない
項目別変換	項目別に分けて変換できる
項目削除	不要な項目を削除できる
項目長変更	項目長を変更できる
項目組み替え	項目の順番の組み替えができる
空項目挿入	空項目を作ることができる

EOFコードの扱い

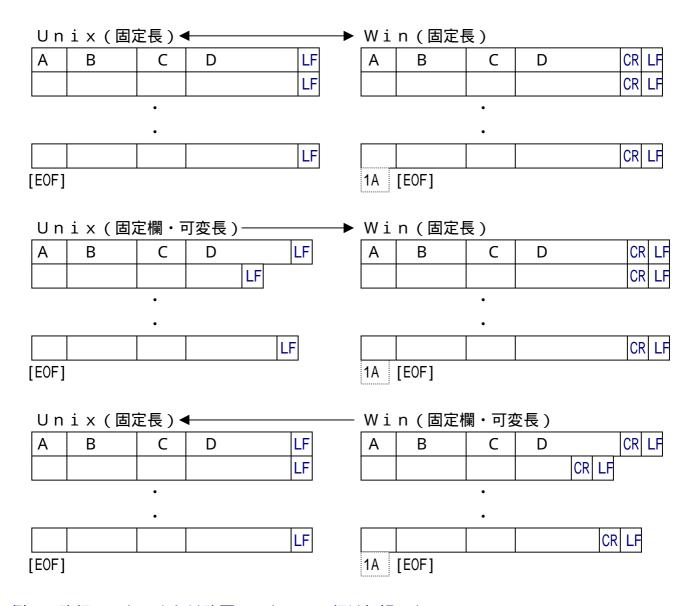
プリント形式ファイルはテキストファイルの一種ですから、ホスト Win方向の変換時には EOFコード(1AH)をつけない、Win ホスト方向の変換時にはEOFコード検出でも物理EOF検出でもEOFにする、というのが省略時の動作です。

EOFコードの扱い

機能	説 明
EOFコード付加	ファイル末尾にEOFコードを付加する・しない(叔ト Win *)
EOFコード検出	EOFコード検出でEOFにする(Win 杁ト*)
EOFコード透過	EOFコードを一般の文字扱いする(Win 杁ト*)

^{*)} ホストがWindowsの場合は、杁ト Win

例1)入力が可変長/固定長、出力が固定長の変換になり、項目別変換ができる

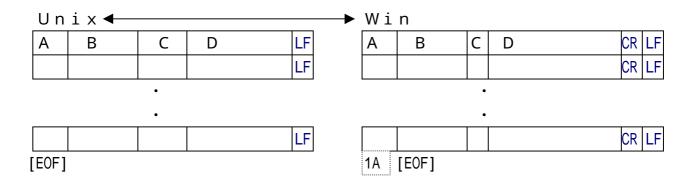


例2) 改行コード、または改頁コードのみの行は無視できる

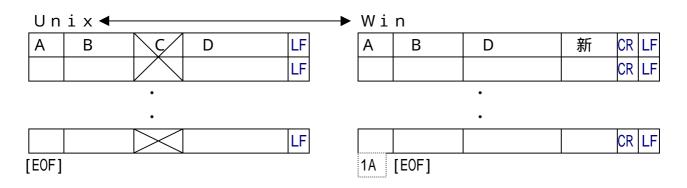
	Unix◀			Win		
Α	L	_F	Α		CR	LF
В	L	_F	В		CR	LF
c	L	_F	С		CR	LF
D	L	_F		CR LF 改行/改頁		
				CR LF		
			D		CR	LF

^{*)} 改頁コード = C R / F F (0 D 0 C H) または F F (0 C H)

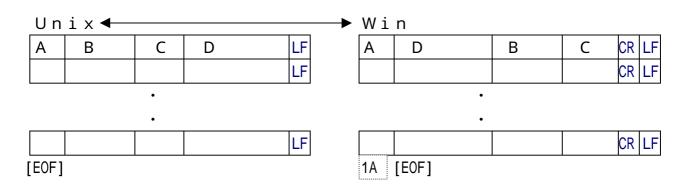
例3)項目長の変更(縮小、拡張)ができる



例4)不要な項目をスキップしたり、逆に空項目を作ったりできる



例5)項目の組み替えができる



2.8 データファイル変換2/デリミタ形式の仕様

ホストが汎用機・オフコンの場合

ホストが汎用機・オフコンの場合、データファイル変換2/デリミタ形式は、おもにホストの COBOLデータと、市販ソフトの入出力形式の1つであるデリミタ形式ファイルの変換に使い ます。

デリミタ形式ファイルとは、各レコードが改行コードで区切られ、さらに各項目がデリミタ(区切り文字)で区切られているファイルのことです。コンマ区切り(CSV)形式[K3フォーマットを含む]がその代表で、つぎの3つのタイプに対応しています。

コンマ区切り(CSV)形式、タブ区切り形式、スペース区切り形式

以下に、データファイル変換2/デリミタ形式で可能なデータ変換機能、レコード編集機能、 EOFコードの扱いを示します。

データ変換機能

データファイル変換2/デリミタ形式のデータ変換機能を、下表に示します。

データ変換機能

<u> </u>		
ホスト側データ	◆→ Win側データ	変換方法
JIS8/ASCII	◆→ JIS8/ASCII	ANK変換
EBCDIC(カタカナ)	4	"
EBCDIC(英小文字)	4	"
漢字(KI/KOなし)	◆▶ シフトJIS漢字のみ	漢字項目变換
ll ll	◆ シフトJIS漢字、JIS8混在	"
ANK・漢字混在	◆◆ JIS8、シフトJIS漢字混在	ANK・漢字
KI/KOつき		混在变換
文字形式数值	◆▶ 文字形式数値、Windows COBOL *1	ニューメリック変換 *3
符号なしゾーン形式	** "	ゾーン変換 *4
符号つきゾーン形式	** "	"
符号なしパック形式	** "	パック変換 *5
符号つきパック形式	** "	"
符号なしBCD形式	** "	"
2 進形式数值 *2	** "	2 進数値変換*6
日付データ	◆▶ 日付データ	日付項目変換
特殊データ	◆▶ 特殊データ	特殊データ変換

^{*1)} Windows COBOL の符号なしゾーン形式、符号つきゾーン形式

^{*2)} COBOL の COMP-5 などの形式

^{*3)} Numeric、DispZone、ZoneDisp 变換

- *4) ZoneDisp、DispZone、ZoneZone 变換
- *5) PackDisp、DispPack、PackZone、ZonePack 变換
- *6) BinDisp、DispBin、BinZone、ZoneBin 变換

レコード編集機能

ホストの標準的なデータ形式は、COBOLのファイル形式です。パソコンの標準的なデータ形式は、プリント形式またはこのデリミタ形式です。両者にはとても大きな隔たりがあり、十分なレコード編集機能が要求されます。F*TRAN+の持つレコード編集機能を下表に示します。

レコード編集機能

レコ 「神来な形	1
機能	説明
タブ拡張	タブを連続スペースに置き換える(Win ホスト)
改行コード付加	無条件にレコード末尾に改行コードを付加する(ホスト Win)
改行コード削除	無条件にレコード末尾の改行コードを削除する(Win ホスト)
空行無視	Win側の、改行・改頁コードのみの行を無視する・しない
項目別変換	項目別に分けて変換できる
可変長項目対応	可変長の項目でも変換できる
デリミタ挿入	項目を区切るデリミタを挿入できる(ホスト Win)
デリミタ検出	項目を区切るデリミタを検出できる(Win ホスト)
引用符くくり	項目を引用符 (") でくくることができる(ホスト Win)
引用符はずし	引用符でくくられた項目の引用符をはずす(Win ホスト)
不要空白削除	デリミタの前後や引用符の前の不要な空白を削除できる
	(ホスト Win)。削除しない指定も可能
項目削除	不要な項目を削除できる
項目長変更	項目長を変更できる
項目組み替え	項目の順番の組み替えができる
空項目挿入	空項目を作ることができる
KI/KO挿入	漢字項目の前後にKI/KOを挿入できる(Win ホスト)
改行コード挿入	レコードの途中にも、改行コードを挿入できる(ホスト Win)

EOFコードの扱い

デリミタ形式ファイルはテキストファイルの一種ですから、ホスト Win方向の変換時には EOFコード(1AH)をつけない、Win ホスト方向の変換時にはEOFコード検出でも物理EOF検出でもEOFにする、というのが省略時の動作です。

EOFコードの扱い

機能	説明
EOFコード付加	ファイル末尾にEOFコードを付加する・しない(ホスト Win)
EOFコード検出	EOFコード検出でEOFにする(Win ホスト)
EOFコード透過	EOFコードを一般の文字扱いする(Win ホスト)

例1)コンマ区切り(CSV)形式の変換ができる

5	木星	JUPITER	ジュピタ-	317832	16
6	土星	SATURN	サターン	095160	17
7	天王星	URANUS	ウラヌス	014540	15
8	海王星	NEPTUNE	ネプ゚チューン	017150	08

ホスト

Win

- " 5" , "木星" , "JUPITER" , "ジュピタ-" , 317.832 , 16
- " 6" , "土星" , "SATURN" , "サターン" , 95.160 , 17
- " 7" ,"天王星" ,"URANUS" ,"ウラヌス" ,14.540 ,15
- " 8" , "海王星" , "NEPTUNE" , "ネプチューン" , 17.150 , 8

例2)タブ区切り形式の変換ができる

5 木星 **゚**シ゛ュピタ-JUPITER 317832 | 16 6 土星 SATURN サターン 095160 17 7 天王星 **URANUS** ウラヌス 014540 15 8 海王星 NEPTUNE ネプ゚チューン 017150 80

ホスト ▲

Win

- 5.....>木星...>JUPITER.>ジュピター.>317.832>16
- 6.....>土星...>SATURN...>サターン....>95.160.>17
- 7.....>天王星.>URANUS...>ウラヌス....>14.540.>15
- 8.....>海王星.>NEPTUNE.>ネプチューン.>17.150.>8

(便宜的に...>でタブを表しています)

例3)スペース区切り形式の変換ができる





\div Win

- 5 木星 JUPITER ジュピタ- 317.832 16
- 6 土星 SATURN サターン 95.160 17
- 7 天王星 URANUS ウラヌス 14.540 15
- 8 海王星 NEPTUNE ネプチューン 17.150 8

例4)不要項目をスキップできる



ホスト

Win "5", "木星", 317.832, 16

" 6" , "土星" , 95.160 , 17

" 7" , "天王星" , 14.540 , 15

"8", "海王星", 17.150, 8

" 5" , "木星" , "JUPITER" , "ジュピタ-" , 317.832 , 16

" 6" , "土星" , "SATURN" , "サターン" , 95.160 , 17

" 7" , "天王星" , "URANUS" , "ウラヌス" , 14.540 , 15

Win "8", "海王星", "NEPTUNE", "ネプチューン", 17.150, 8



5	木星	317832	16
6	土星	095160	17
7	天王星	014540	15
8	海王星	017150	80

例5)項目の組み替えができる





" 5" , "JUPITER" , "ジュピタ-" , "木星" , 317.832 , 16

" 6" , "SATURN" , "サターン" , "土星" , 95.160 , 17

" 7" , "URANUS" , "ウラヌス" , "天王星" , 14.540 , 15

"8", "NEPTUNE", "ネプチューン", "海王星", 17.150, 8

ホストがUnix、Windowsの場合

ホストがU n i x、W i n d o w s の場合、データファイル変換 2 / デリミタ形式は、ホストのC O B O L データ等と、市販ソフトの入出力形式の 1 つであるデリミタ形式ファイルの変換に使います。

デリミタ形式ファイルとは、各レコードが改行コードで区切られ、さらに各項目がデリミタ(区切り文字)で区切られているファイルのことです。コンマ区切り(CSV)形式[K3フォーマットを含む]がその代表で、つぎの3つのタイプに対応しています。

コンマ区切り(CSV)形式、タブ区切り形式、スペース区切り形式

以下に、データファイル変換 2 / デリミタ形式で可能なデータ変換機能、レコード編集機能、 E O F コードの扱いを示します。

データ変換機能

データファイル変換2/デリミタ形式のデータ変換機能を、下表に示します。

データ変換機能

ノーノ攵〕矢1成形			
ホスト側データ	◆→ Win側	データ	変換方法
JIS8/ASCII	←► JIS8,	/ A S C I I	ANK変換
漢字 (EUC、シフト JIS)	◆→ シフトJ:	IS漢字のみ	漢字項目変換
ll ll	← シフトJ∑	[S漢字、JIS8混在	<i>"</i>
JIS8・漢字混在	↔ JIS8,	シフトJIS漢字混在	ANK・漢字
			混在変換
文字形式数值	◆◆ 文字形式数	效値、Windows COBOL *1	ニューメリック変換 *3
符号なしゾーン形式	←→	<i>II</i>	ゾーン変換 *4
符号つきゾーン形式	←→	<i>II</i>	<i>''</i>
符号なしパック形式	←→	<i>II</i>	パック変換 *5
符号つきパック形式	←→	<i>II</i>	<i>''</i>
符号なしBCD形式	←→	<i>II</i>	<i>''</i>
2 進形式数值 *2	←→	<i>II</i>	2 進数値変換*6
日付データ	◆▶ 日付データ	7	日付項目変換
特殊データ	◆▶ 特殊データ	Þ	特殊データ変換

- *1) Windows COBOL の符号なしゾーン形式、符号つきゾーン形式
- *2) COBOL の COMP-5 などの形式
- *3) Numeric、DispZone、ZoneDisp 変換
- *4) ZoneDisp、DispZone、ZoneZone 变換
- *5) PackDisp、DispPack、PackZone、ZonePack 变換
- *6) BinDisp、DispBin、BinZone、ZoneBin 変換

レコード編集機能

Unix、Windowsの標準的なデータ形式は、プリント形式またはこのデリミタ形式です。 F*TRAN+の持つレコード編集機能を下表に示します。

レコード編集機能

機能	説明
タブ拡張	タブを連続スペースに置き換える
改行コード正規化	レコード末尾の改行コードを正規化する
空行無視	改行・改頁コードのみの行を無視する・しない
項目別変換	項目別に分けて変換できる
可変長項目対応	可変長の項目でも変換できる
デリミタ挿入	項目を区切るデリミタを挿入できる
デリミタ検出	項目を区切るデリミタを検出できる
引用符くくり	項目を引用符(")でくくることができる
引用符はずし	引用符でくくられた項目の引用符をはずす
不要空白削除	デリミタの前後や引用符の前の不要な空白を削除できる
	削除しない指定も可能
項目削除	不要な項目を削除できる
項目長変更	項目長を変更できる
項目組み替え	項目の順番の組み替えができる
空項目挿入	空項目を作ることができる
改行コード挿入	レコードの途中にも、改行コードを挿入できる

EOFコードの扱い

デリミタ形式ファイルはテキストファイルの一種ですから、ホスト Win方向の変換時にはEOFコード(1AH)をつけない、Win ホスト方向の変換時にはEOFコード検出でも物理EOF検出でもEOFにする、というのが省略時の動作です。

<u>EOFコードの扱い</u>

機能	説明
EOFコード付加	ファイル末尾にEOFコードを付加する・しない(叔ト Win *)
EOFコード検出	EOFコード検出でEOFにする(Win 杁ト*)
EOFコード透過	EOFコードを一般の文字扱いする(Win 叔ト*)

^{*)} ホストがWindowsの場合は、杁ト Win

例1)コンマ区切り(CSV)形式の変換ができる

5	木星	JUPITER	ジュピタ-	317832	16	LF
6	土星	SATURN	サターン	095160	17	LF
7	天王星	URANUS	ウラヌス	014540	15	LF
8	海王星	NEPTUNE	ネプ゚チューン	017150	08	LF

Unix

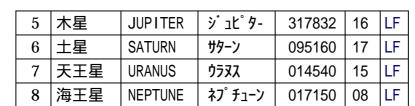
Win "5", "木星", "JUPITER", "ジュピタ-", 317.832, 16

" 6" , "土星" , "SATURN" , "サターン" , 95.160 , 17

" 7" ,"天王星" ,"URANUS" ,"ウラヌス" ,14.540 ,15

" 8" , "海王星" , "NEPTUNE" , "ネプチューン" , 17.150 , 8

例2)タブ区切り形式の変換ができる



Unix

Win

5.....>木星...>JUPITER.>ジュピター.>317.832>16

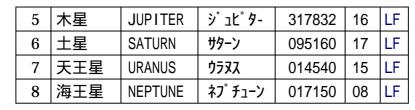
6.....>土星...>SATURN..>サターン....>95.160.>17

7.....>天王星.>URANUS...>ウラヌス....>14.540.>15

8.....>海王星.>NEPTUNE.>ネプチューン.>17.150.>8

(便宜的に...>でタブを表しています)

例3)スペース区切り形式の変換ができる





Win

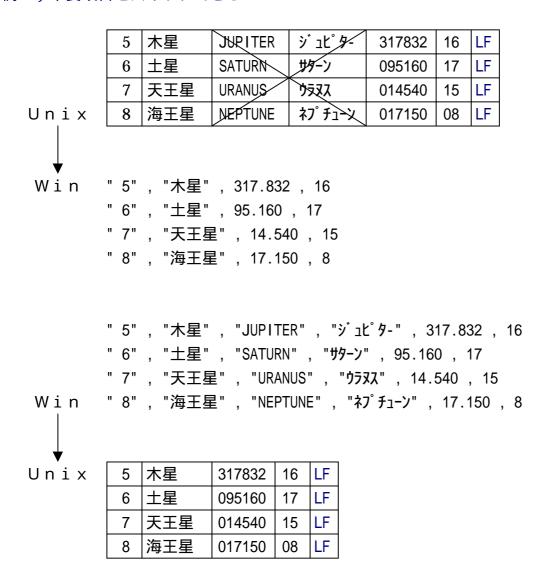
5 木星 JUPITER ジュピタ- 317.832 16

6 土星 SATURN サターン 95.160 17

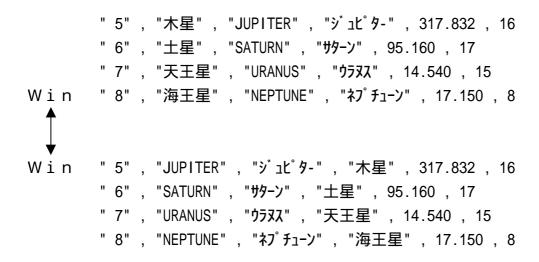
7 天王星 URANUS ウラヌス 14.540 15

8 海王星 NEPTUNE ネプチューン 17.150 8

例4)不要項目をスキップできる



例5)項目の組み替えができる



2.9 ランダムファイル変換の仕様

ホストが汎用機・オフコンの場合

ランダムファイル変換では、アプリケーションで使用する、おもに固定長・固定欄のデータファイルの変換を行います。BASICのランダムファイルやCOBOLの順ファイルなどがおもな対象です。バイナリ変換ができるのも大きな特長です。

以下に、ランダムファイル変換で可能なデータ変換機能、レコード編集機能、EOFコードの 扱いを示します。

データ変換機能

ランダムファイル変換のデータ変換機能を下表に示します。

データ変換機能

ホスト側データ	◆→ Win側データ	変換方法
バイナリ	◆▶ バイナリ	バイナリ変換
JIS8/ASCII	◆→ JIS8/ASCII	ANK変換
EBCDIC(カタカナ)	4	"
EBCDIC(英小文字)	** "	"
漢字(KI/KOなし)	◆→ シフトJIS漢字のみ	漢字項目変換
"	◆ シフトJIS漢字、JIS8混在	"
ANK・漢字混在	◆◆ JIS8、シフトJIS漢字混在	ANK・漢字
K I / K 0 つき		混在变換
文字形式数值	◆→ 文字、WinCOBOL *1、2進 *2	ニューメリック変換 *4
符号なしゾーン形式	** "	ゾーン変換 *5
符号つきゾーン形式	** "	"
符号なしパック形式	** "	パック変換 *6
符号つきパック形式	** "	"
符号なしBCD形式	** "	"
2 進形式数值 *3	** "	2 進数値変換*7
日付データ	◆▶ 日付データ	日付項目変換
特殊データ	◆▶ 特殊データ	特殊データ変換

- *1) Windows COBOL の符号なしゾーン / パック形式、符号つきゾーン / パック形式
- *2) Windows COBOLのCOMP-5、Visual Basic、C/C++などの形式
- *3) COBOL の COMP-5 などの形式
- *4) Numeric、DispZone、ZoneDisp、DispPack、PackDisp、DispBin、BinDisp 变換
- *5) ZoneDisp、DispZone、ZoneZone、ZonePack、PackZone、ZoneBin、BinZone 变換
- *6) PackDisp、DispPack、PackZone、ZonePack、PackPack、PackBin、BinPack 变換
- *7) BinDisp、DispBin、BinZone、ZoneBin、BinPack、PackBin、BinBin 変換

注意 ---- MKI\$/MKS\$/MKD\$には未対応

BASICの内部形式の数値項目である、MKI\$/MKS\$/MKD\$の項目はサポートしていません。

レコード編集機能

ランダムファイル変換では、Win側のレコード長を明示的に指定します(指定を省略するとホストレコード長=Winレコード長になります)。固定長という点では、ホストのデータ形式と似ています。

しかし、データ内容に応じて項目長変更などのレコード編集機能が必要になります。明示的に 指定しない限り改行コードをつけないことなどは、大きな特長です。

下表にランダムファイル変換のレコード編集機能を示します。

レコード編集機能

機能	説明
Winレコード長指定	Win側のレコード長を指定できる
項目別変換	項目別に分けて変換できる
項目削除	不要な項目を削除できる
項目長変更	項目長を変更できる
項目組み替え	項目の順番の組み替えができる
空項目挿入	空項目を作ることができる
KI/KO挿入	漢字項目の前後にKI/KOを挿入できる(Win ホスト)
改行コード挿入	レコードの途中・末尾に、改行コードを挿入できる
	(ホスト Win)

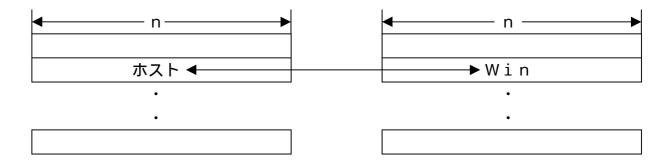
EOFコードの扱い

ランダムファイル変換は、基本的にデリミタや改行コードを使わず、あってもただの一般文字扱いにします。EOFコードについても同様に一般文字扱いし、物理EOFでEOFにします。これは、バイナリ変換ができることを前提にしているためです。

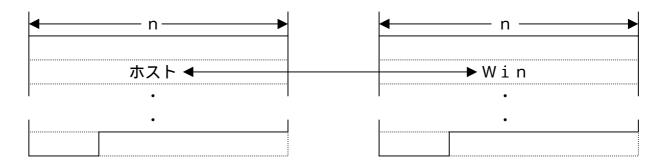
EOFコードの扱い

機能	説明
EOFコード透過	EOFコードを一般の文字扱いする(Win ホスト)

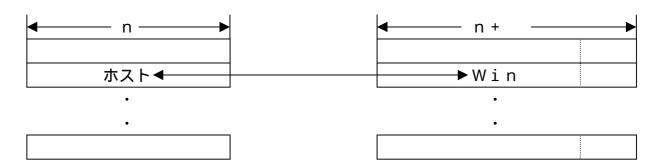
例1)ホスト側とWin側のレコード長を同じにするのがふつう



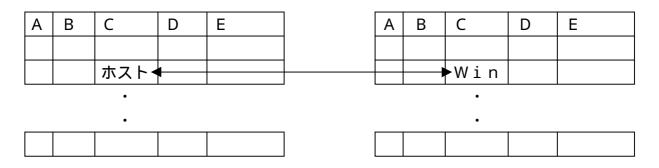
例2)バイナリ変換もできる。レコード長はあまり意味を持たない



例3)レコード長の変更(縮小、拡大)ができる



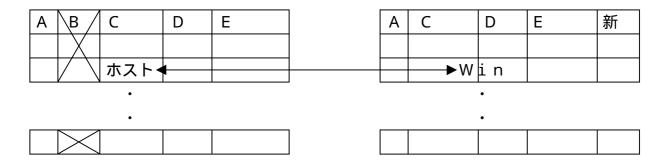
例4)項目別の変換ができる



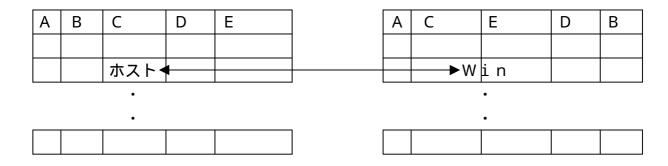
例5)項目長の変更(縮小、拡大)ができる

Α	В	С	D	Е		Α	В	С	D	E
		ホスト◀	1				→ W	in		
		•	•			•				
	•									

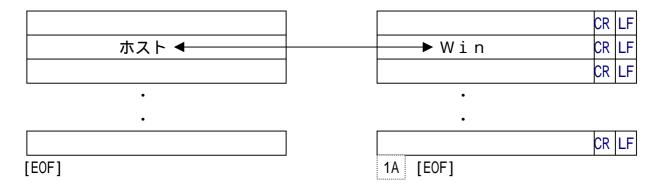
例6)不要な項目をスキップしたり、逆に空項目を作ったりできる



例7)項目の組み替えができる



例8) 改行コードのつく、固定長のテキストファイルも扱える



ホストがUnix、Windowsの場合

ランダムファイル変換では、アプリケーションで使用する、おもに固定長・固定欄のデータファイルの変換を行います。BASICのランダムファイルやCOBOLの順ファイルなどがおもな対象です。バイナリ変換ができるのも大きな特長です。

以下に、ランダムファイル変換で可能なデータ変換機能、レコード編集機能、EOFコードの 扱いを示します。

データ変換機能

ランダムファイル変換のデータ変換機能を下表に示します。

データ変換機能

× × × × 100 100					
ホスト側データ	スト側データ ◆◆ Win側データ			方法	
バイナリ	◆→ バイ	イナリ	バイナ	リ変換	
JIS8/ASCII	↔ J]	IS8/ASCII	ANK	変換	
漢字 (EUC、シフト JIS)	◆→ シフ	フトJIS漢字のみ	漢字項	目変換	
<i>II</i>	← シフ	7トJIS漢字、JIS8混在	"		
ANK・漢字混在	↔ J]	[S8、シフト」 [S漢字混在	ANK	・漢字	
			混	在変換	
文字形式数值	◆◆ 文字	롣、WinCOBOL *1、2数 *2	ニューメリッ	ク変換 ′	*4
符号なしゾーン形式	↔	"	ゾーン	変換 ′	*5
符号つきゾーン形式	↔	"	"		
符号なしパック形式	↔	"	パック	変換 ′	*6
符号つきパック形式	↔	<i>"</i>	"		
符号なしBCD形式	↔	<i>"</i>	"		
2 数形式数值 *3	←→	<i>II</i>	2 進数	值変換'	*7
日付データ	◆◆ 日作	寸データ	日付項	目変換	
特殊データ	◆▶ 特別	未データ	特殊デー	-9変換	

- *1)Windows COBOLの符号なしゾーン/パック形式、符号つきゾーン/パック形式
- *2) Windows COBOLのCOMP-5、Visual Basic、C/C++などの形式
- *3) COBOL の COMP-5 などの形式
- *4) Numeric、DispZone、ZoneDisp、DispPack、PackDisp、DispBin、BinDisp 变換
- *5) ZoneDisp、DispZone、ZoneZone、ZonePack、PackZone、ZoneBin、BinZone 変換
- *6) PackDisp、DispPack、PackZone、ZonePack、PackPack、PackBin、BinPack 变換
- *7) BinDisp、DispBin、BinZone、ZoneBin、BinPack、PackBin、BinBin 变換

注意 ---- MKI\$/MKS\$/MKD\$には未対応

BASICの内部形式の数値項目である、MKI\$/MKS\$/MKD\$の項目はサポートしていません。

レコード編集機能

ランダムファイル変換では、Win側のレコード長を明示的に指定します。

しかし、データ内容に応じて項目長変更などのレコード編集機能が必要になります。明示的に 指定しない限り改行コードをつけないことなどは、大きな特長です。

下表にランダムファイル変換のレコード編集機能を示します。

レコード編集機能

機能	説明				
タブ拡張	タブを連続スペースに置き換える(ホスト Win)				
Winレコード長指定	Win側のレコード長を指定できる				
項目別変換	項目別に分けて変換できる				
項目削除	不要な項目を削除できる				
項目長変更	項目長を変更できる				
項目組み替え	項目の順番の組み替えができる				
空項目挿入	空項目を作ることができる				
改行コード挿入	レコードの途中・末尾に、改行コードを挿入できる				

EOFコードの扱い

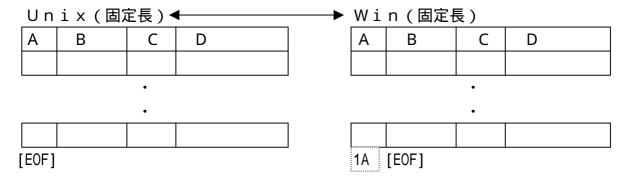
ランダムファイル変換は、基本的にデリミタや改行コードを使わず、あってもただの一般文字扱いにします。EOFコードについても同様に一般文字扱いし、物理EOFでEOFにします。これは、バイナリ変換ができることを前提にしているためです。

EOFコードの扱い

機能	説明
EOFコード透過	EOFコードを一般の文字扱いする(Win 杁ト*)

^{*)} ホストがWindowsの場合は、杁ト Win

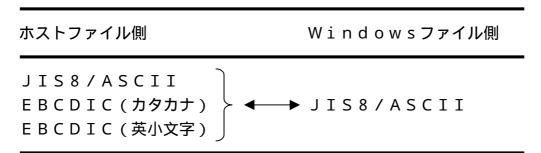
例)ホスト側が可変長/固定長、Win側が固定長の変換になり、項目別変換ができる



2.10 ANK変換

ANK変換(Ank変換)

ANK変換とは、1バイト系の文字、つまり半角文字の変換機能のことです。ホストが汎用機・オフコンの場合のANK変換は、つぎに示すように3とおりの変換ができます。



ホストがUnix、Windowsの場合のANK変換は、ホストファイル側がJIS8/ASCII固定になります。

この切り替えは、変換設定/ANKコードで行います。また、ANK変換という名前にそぐわず、特例としてシフトJIS漢字を素通しにする変換もできます。

JIS8/ASCIII-F

JIS8/ASCIIコードは、パソコンで標準的に使われているコード系です。一部のホストでも使われています。

本書ではJIS8コードとASCIIコードを合わせてJIS8/ASCIIコードと呼んでいますが、本当は違うものです。ASCIIコードは米国標準の7ビットコード系、JIS8コードはそれにカタカナを加えて8ビットに拡張した日本のコード系、と思っておいて大過ありません。なお、国内のパソコン関係で「ASCIIナントカ」というときは、本当はJIS8コードのことです。つぎに示す、

のコードぐらいはソラで覚えておくと、何かと便利です。

E B C D I C コード

EBCDICコードは、汎用機を中心として、ホストで多く用いられているコード系です。事実上の国際標準になっていますが、規格化はされていません。そのため、メーカー・システムごとに少しずつ違いがあります。

また、通常 E B C D I C コードではカタカナと英小文字が同時に使えないことも重要なポイントです。カタカナと英小文字のコードが重複しているためです。たとえば、8 1 H というコードが表す文字は、

カタカナ版 8 1 = " ア " 英小文字版 8 1 = " a "

という具合です。なお、国内ではEBCDIC(英小文字)コードはあまり使われません。しかし、国外との電文の授受がある部署ではこれがごくふつうに使われています。国際的にはカタカナ版はまったく通用しないことを知っておいたほうがよいでしょう。ファイル名にカタカナを使ったりすると、国外のシステムとはファイル転送すらできないことがあります。F*TRAN+では、

EBCDIC (カタカナ) 用のANK変換表 EBCDIC (英小文字) 用のANK変換表

の2つを提供し、どちらか一方に切り替えて使えるようにしています。つぎに示す、

スペース 4 0
" A " ~ " I " C 1 ~ C 9
" J " ~ " R " D 1 ~ D 9
" S " ~ " Z " E 2 ~ E 9
" 0 " ~ " 9 " F 0 ~ F 9

のコードぐらいはソラで覚えておくと、何かと便利です。

JIS8/ASCIIコードとEBCDICコードの対応関係

JIS8/ASCIIコードとEBCDICコードでは、一部の特殊文字の対応がとれていません。一方にあってもう一方にはない特殊文字がいくつかあるのです。そのため、両者のコード変換規則はメーカー・システムによってバラバラなのが現状です。

F * T R A N + では、ほぼ標準的と思われる A N K 変換表を提供し、必要ならそれを利用者が 簡単に変換できるようにして、この問題に対処しています。

2 . 11 漢字変換

F * T R A N + の漢字変換機能には、

Ank・漢字(KI/KO付)混在変換 漢字項目変換

と呼ぶ、2つの変換方法があります。

Ank・漢字(KI/KO付)混在変換

テキストファイル変換では、ホストファイル側で、レコードごとに漢字の位置が固定していないタイプのファイルを変換できます。ソースプログラムがその典型です。

データファイル変換やランダムファイル変換では、ホストファイル側にANK・漢字混在の項目があるタイプのファイルを変換できます。ただし、扱いにくいのであまり使われません。

どちらも、ANKから漢字への切り替えをKIと呼ぶコードで行い、漢字からANKへの切り替えをKOと呼ぶコードで行います。KI/KOがついているからどの桁に漢字が現れてもよいのです。

そして、EUC(Unix)、シフトJIS(Windows)のように、KI/KOの概念がない漢字データ(EUCの場合は、G3等のシフトコード付きの漢字を含む)も扱えます。

このタイプの漢字サポートを F * T R A N + では A N K・漢字(K I / K O 付)混在変換と呼んでいます。

漢字項目変換

データファイル変換とランダムファイル変換の場合、ホストファイルからWindowsファイルへの変換のときは、KI/KOがなくても変換できます。逆のWindowsファイルからホストファイルへの変換のときも、KI/KOをつけないで変換することができます。

どちらも、ANK何バイト、漢字何バイト・・・と指示できるからです。これを漢字項目変換と呼んでいます。

ANK項目	漢字項目	ANK項目	漢字項目	他の項目・・・
-------	------	-------	------	---------

テキストファイル変換の場合は、これに相当する機能はありません。

半角 全角变換機能

Windowsファイルでは、半角文字・全角文字の項目が、ごくふつうに使われています(禁止しているソフトもあります)。

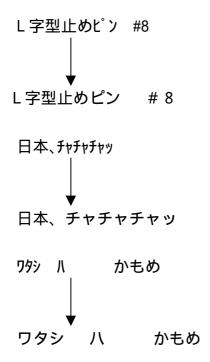
一方、ホストファイル側では(ホスト上では) 半角文字・全角文字混在の項目は非常に扱いにくいものです。

そのため、F*TRAN+では半角 全角変換機能をサポートしています。Win ホスト方向の変換*で、半角文字・全角文字混在の項目に対して漢字項目変換を適用すると、自動的に半角文字を対応する全角文字に変換し、全角文字のみの項目にします。

なお、半角スペース(20H)は特別扱いです。半角スペースは2個1組で1個の漢字スペースとして扱います。半角スペースが奇数個つづくと、最後に1個の半角スペースが残ることになります。それは漢字スペースに変換します。

*)ホストがUnix、Windowsの場合は、ホスト Win方向の変換も対象になる

例をあげます。 は半角スペース、 は漢字スペースを意味しています。



注意 ---- オーバーフローしやすい

この機能を使うときは、項目長やレコード長が増える(最大で2倍になる)ことを計算に入れておかなければいけません。オーバーフローに注意してください。

2.12 漢字変換方式のサポートパターン

利用者が自由に以下の項目を設定し、漢字変換項目をアレンジしたり、新たに登録したりできます。

ホストが汎用機・オフコンの場合

名前 漢字変換方式につける名前(8文字以内) コメント それにつけるコメント(40文字以内) ANKコード JIS8/ASCII EBCDIC(カタカナ/英小文字) 漢字コード JIS漢字 JIS漢字+8080H JIS漢字+8080H(ユニシス) NEC JIPS(E) JIPS(J) 内部コード(E) 内部コード(J) 東芝漢字(漢字コード自体はJIS) IBM漢字 漢字イン *1 0~3バイト 漢字アウト *1 0~3バイト シフト節約度 弱/中/強 (Win ホストファイル変換時) ホスト漢字スペース 全角/半角2個(ユニシスは全角1/全角2あり) Win漢字スペース 全角 / 半角 2 個 ホスト拡張漢字 *2 ゲタ変換 / クエスチョン変換 / ゼロイー変換 " 〓 " の代わり ゲタ変換時の"〓"の代わりの全角文字 半角スペース2個(2020H)にも設定できる W i n 拡張漢字 *3 ゲタ変換に固定 " 〓 " の代わり " ■ "の代わりの全角文字 半角スペース 2 個 (2020H か 4040H) にも設定できる 漢字対応表 漢字コード変換テーブルの設定 *4

- *1) 漢字項目変換のときは使われない
- *2)漢字対応表に定義されないホスト拡張漢字、外字(ユーザー定義文字)も含む
- *3) 漢字対応表に定義されないWin拡張漢字、94区を越える漢字
- *4) 通常は、対応づけができる拡張漢字コードを登録する

ホストがUnixの場合

名前漢字変換方式につける名前(8文字以内)コメントそれにつけるコメント(40文字以内)

ホスト漢字スペース 全角(A1A1H)/半角2個(2020H)

置換する/しない

Win漢字スペース 全角(8140H)/半角2個(2020H)

置換する / しない

半角カタカナ 8 E H + カナコード / 全角化 / ゲタ変換

" = "の代わり ゲタ変換時の " = "の代わりの半角文字

(Win ホストファイル変換時)

未定義カタカナ "="変換固定

(ホスト Win ファイル変換時)

ホスト拡張漢字 *2 ゲタ変換/クエスチョン変換

" **=** " の代わり ゲタ変換時の " **=** " の代わりの全角文字

半角スペース2個(2020H)にも設定できる

Win拡張漢字 *3 ゲタ変換に固定

半角スペース2個(2020H)にも設定できる

漢字対応表 漢字コード変換テーブルの設定 *4

ホストがWindowsの場合

名前 漢字変換方式につける名前(8文字以内)

コメント それにつけるコメント(40文字以内)

ホスト漢字スペース 全角(8140H)/半角2個(2020H)

置換する/しない

Win漢字スペース 全角(8140H)/半角2個(2020H)

置換する / しない

漢字対応表 漢字コード変換テーブルの設定 *4

基本的に、このパターンに合えば漢字変換ができます。

2.13 ホスト漢字のレパートリー

先に述べたサポートパターンに合うものとしては、Ank・漢字(KI/KO付)混在変換の場合、

富士通 JEF(12 ポイント) ---- 一番ふつうのもの

JEF(9ポイント)

F 9 4 5 0 内部コード (F C O N V の N C 指定で変換したもの)

日立 KEIS

三菱 MELCOM漢字

NEC JIPS(E) NEC標準の漢字体系

JIPS(J) (ANKも漢字もJISコードにしたもの)

内部コード(E) オフコン・ミニコンで多く使われる

内部コード(J) (めったに使われない)

東芝東芝標準漢字

日本ユニシス LETS-J(旧日本ユニバック系)

カシオカシオ標準漢字IBMIBM漢字

新JIS 各社ホストのユーティリティで処理したもの

旧JIS " Unix EUC

Windows シフトJIS

などがあります。

漢字項目変換の場合は、KI/KOを合わせる必要がないので、さらに多くのメーカー・システムに適合します。

漢字変換方式は、1 つのコード変換表につき 1 5 種類まで登録できます。よく使用されるものは、はじめから標準提供のコード変換表に登録されています。

注意 ---- IBM漢字(IBM社の内部漢字コード)もサポートしています

F*TRAN+では、IBM漢字コードを内部テーブル変換によって求めています。そのため、 わざわざホスト側のユーティリティを使ってJIS漢字に直す必要はありません。

IBM漢字以外の漢字コードはJIS-C6226/X0208と同一、またはそれに準拠している必要があります(漢字コードを計算で求めるため)。

2 . 14 漢字コード変換の仕組み

F*TRAN+の漢字コード変換の仕組みはつぎのとおりです。

ホストの漢字

ホストの漢字は、大別するとつぎの2つになります。

- 1.一般漢字(JIS第1/第2水準、多くのホストで共通の漢字)
- 2.拡張漢字(ホストによって異なる固有の漢字、ユーザー定義文字を含む)
- (注)ホストによっては、一般漢字エリアの中に特殊な漢字が含まれる場合があります。

Windowsの漢字

Windowsの漢字も、大別するとつぎの2つになります。

- 1.一般漢字(JIS第1/第2水準、多くのホストと共通の漢字)
- 2.拡張漢字(一般漢字以外で追加された漢字、ユーザー定義文字を含む)

(注)一般漢字エリアの一部の漢字は、ホストの一般漢字エリアの同位置に定義されていない場合が多くみられます。

ホストの漢字 Windowsの漢字 処理順序

基本的に、一般漢字と一部の拡張漢字は正当な変換が可能で、多くの拡張漢字は正当な変換が不可能です。ただし、漢字変換表をうまく利用すれば意図した変換ができます。

F*TRAN+では、つぎの順序で漢字コードの変換を行います。

漢字対応表に定義されている漢字の変換

ふつうは、"(株)"のような対応づけができる拡張漢字を対象にしますが、

- 一般漢字エリアの中に含まれる特殊な漢字の変換を含めることもできます。
- 一般漢字エリアの漢字変換
- 一般漢字エリアの漢字は、計算によって求められます。

(IBM漢字は内部テーブル変換)

漢字対応表に定義されない拡張漢字

ホストとWindowsで対応づけができない漢字を、

ユーザーが選択した方式で変換します。

2 . 15 漢字対応表について

漢字対応表は、F * T R A N + の漢字コード変換に反映することができる漢字変換テーブルです。漢字対応表のメンテナンスは、F * T R A N + に標準提供されている漢字対応表エディタによって簡単に行うことができます。ふつうは、メーカー単位で標準提供されている漢字対応表をベースに使用します。

漢字対応表に定義された情報は、データ変換時に読み込まれ、最優先の変換になります。漢字 対応表に定義される基本パターンはつぎのとおりです。

- 2. Windows 漢字(またはコード) → ホストの漢字コード (片方向)
- 3. Windows 漢字(またはコード) **◆** ホストの漢字コード (片方向)

詳細は、4.13 漢字対応表エディタをご覧ください。

2.16 変換できないホストの拡張漢字の扱い

F*TRAN+は、漢字対応表に定義されないホストの拡張漢字の変換をつぎのように扱います。ここでいう拡張漢字とは、ふつうの拡張漢字と外字(ユーザー定義文字)の総称です。つぎの、

ゲタ変換 クエスチョン変換 ゼロイー変換

と呼ぶ、3とおりの変換ができます。

以下、富士通JEFの「㈱」=78D5Hを例にとって、どのように変換できるかを見ていきます。

ゲタ変換

ホストファイルの拡張漢字を、Windowsファイルではゲタ(■) あるいはその代わりの全角文字に変換させる方式です。例外として、半角スペース2個(2020H)も代わりの文字として使えます。

ホストファイル	Windowsファイル
	▶ 81AC = " = "(あるいは、代わりの全角文字)▶ 81AC = " = "(あるいは、代わりの全角文字)

この方式には、 どこに拡張漢字が使われているか、ひと目でわかる "〓"以外の代わりの全角文字に設定することもできる 2 バイト 2 バイト変換なので、桁ずれを起こさない

などの特長があります。

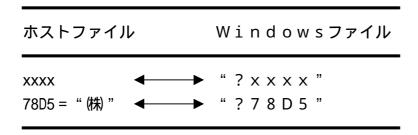
その反面、 パソコンからホストへの変換には使えない ホストの元のコードを知る手掛かりがない

という難点もあります。

なお、出荷時設定はすべてこの「ゲタ変換」になっています。拡張漢字をそれほど使わないな ら、この設定が無難です。

クエスチョン変換

ホストの拡張漢字を、パソコンでは全角"?"+半角16進4桁のパターンで表す方式です。



この方式には、 どこに拡張漢字が使われているか、わかりやすい

ホストのコードがひと目でわかる

文字表現なので、エディタによる置き換えなどがしやすい

パソコンからホストへ変換するときも、元のホストコードに戻る

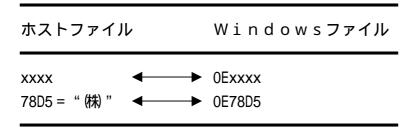
などの特長があります。いろいろと、扱いやすいのです。

その反面、 2 バイト 6 バイト変換なので、桁ずれを起こす 言い替えれば、オーバーフローしやすい

という難点もあります。ふつうは、先のゲタ変換か、このクエスチョン変換を使います。

ゼロイー変換

ホストが汎用機・オフコンの場合、ホストの拡張漢字をパソコンでは0EH+16進ホストコードのパターンで表す方式です。富士通専用の方式で、一般には意味がないと思ってください。



この方式は、富士通のホストとデータ交換するときだけ、意味を持ちます。さらに、JEF拡張漢字ドライバを組み込んでおかなければいけませんし、ソフトがこの方式・ドライバに対応していなければいけません。この条件を満たせば、FMVシリーズでJEF拡張漢字の表示・印刷ができます。

2.17 ホストの一般漢字と拡張漢字の区別

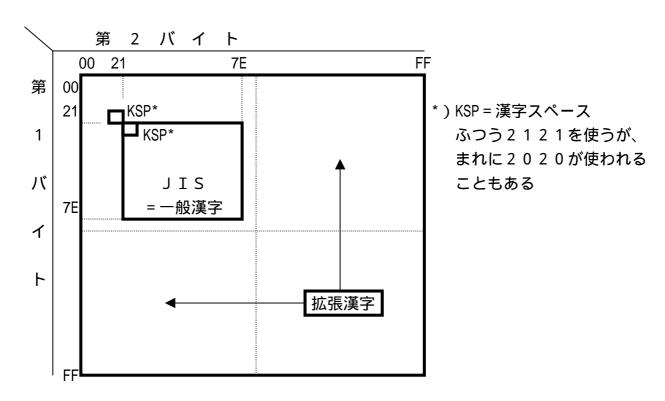
F*TRAN+がホストファイル側の一般漢字と拡張漢字をどのように区別しているかを述べます。

「一般漢字」とは、JIS句点コードで1区1点~94区94点の範囲内の漢字すべてを意味します。「拡張漢字」とは、一般漢字以外のすべてのコード、すなわち、1区1点~94区94点の範囲以外のコードを持つ漢字すべてです。

JISタイプ、NEC漢字、JIS+8080タイプ、EUC、IBM漢字、に分けて説明します。

JISタイプ

JISタイプの1区1点~94区94点内の漢字とは、16進コードでいい替えると第1バイトが21H~7EHかつ第2バイトも21H~7EHの範囲に収まる漢字を指します。そして、この範囲内に収まらないコードの漢字を拡張漢字と呼びます。



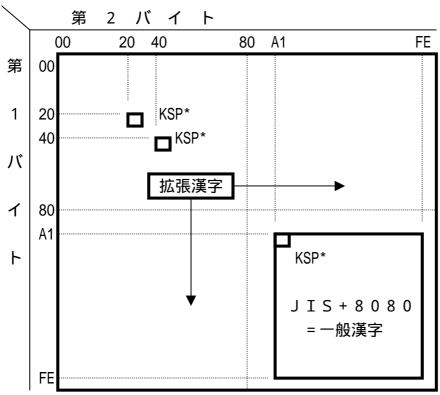
このタイプでは、一般漢字の領域内にそのメーカー・システム固有の特殊漢字を含めていることが多いものです。ですが、図からわかるようにF*TRAN+ではそれは拡張漢字としては扱いません。

NEC漢字

NEC漢字の場合は単純な判定はできないので、いったんJISタイプに変換してから一般漢字・拡張漢字の判定をします。

JIS+8080タイプ

JIS+8080タイプの1区1点~94区94点内の漢字とは、16進コードでいい替えると、第1バイトがA1H~FEHかつ第2バイトもA1H~FEHの範囲に収まる漢字を指します。そして、この範囲内に収まらないコードの漢字を拡張漢字と呼びます。ただし、ホスト側の漢字スペースを半角2個に設定しているときは、それはこの範囲には収まりません。



*) KSP = 漢字スペース。2020、4040、A1A1がある

2020: JIS8/ASCII系のシステムで、まれに使われることがある

4 0 4 0 : E B C D I C 系のシステムで多く使われる A 1 A 1 : J I S + 8 0 8 0 タイプの全角スペース

このタイプの場合は、一般漢字の領域内にそのメーカー・システム固有の特殊漢字を含めることはあまりありません。

EUC

EUCの一般漢字はJIS+8080と同じですが、拡張漢字は8FH+(JIS+8080)の3バイトコードが標準です。

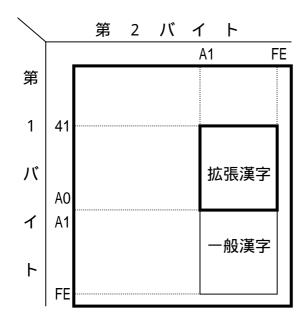
IBM漢字

IBM漢字の場合は、16進コードで第1バイトが41H~7FHかつ第2バイトが41H~FEHの範囲を漢字(ただし、漢字スペースは4040H)として扱い、内部テーブルによってコード変換を行います。漢字エリアの中には、拡張漢字/ユーザー選定文字の領域も含まれています。

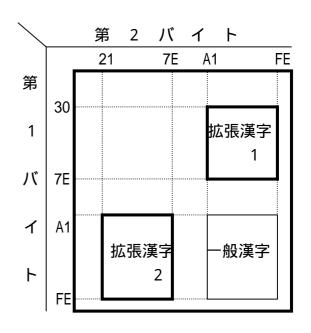
2 . 18 ホストの拡張漢字の領域

F*TRAN+では、拡張漢字の領域をつぎのように認識しています。

JIS+8080系

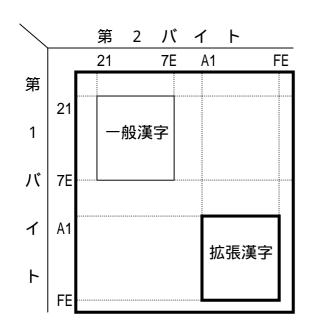


富士通 JEF、日立 KEIS、三菱 MELCOM



日本ユニシス LETS-J

JIS系



東芝漢字、NEC漢字JIPS

EUC

一般漢字は、JIS+8080系と同じ。 拡張漢字は、つぎのタイプがあります。

標準:

8FH + A1A1H~FEFEH DEC漢字:

A 1 2 1 H ~ F E 7 E H

Super DEC漢字:

8FH + A1A1H~FEFEH

8 F H + A 1 2 1 H ~ F E 7 E H

IBM漢字

4040H、4141H~7FFEHの 範囲に一般漢字、拡張漢字が含まれます。

2 . 19 変換できない Windows の拡張漢字の扱い

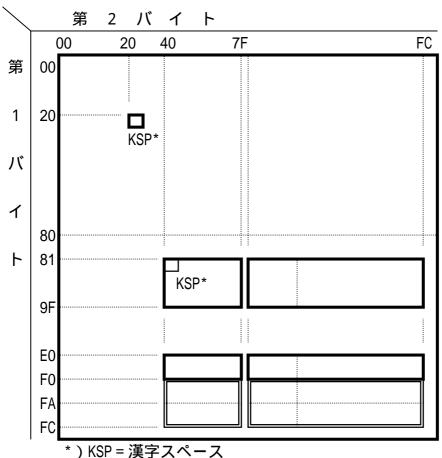
F*TRAN+は、漢字対応表に定義されないWindowsの拡張漢字をつぎのように扱います。文字化けを防ぐためです。ここでいう拡張漢字とは、<math>94区を超える漢字コードのことです。

シフトJISのコード体系では、JISの1区1点~94区94点に収まらない領域があります。JISの区点コードを拡張して表現すれば、95区~120区(シフトJISではF040~FCFC)の領域です。ここが拡張漢字の領域です(下図ののの部分)。Windows環境では、この領域を

 外字(1880字分)
 95区~114区(F040~F9FC)

 IBM社選定拡張漢字
 115区~120区(FA40~FCFC)

に分けて利用しています。IBM社選定拡張漢字は、ホストの拡張漢字の中に割り付けられている漢字が多いので漢字対応表で定義することができます。しかし、それ以外の漢字はJISコードに対応づけることができないので、ホストの漢字コードに変換するときF*TRAN+ではゲタ(〓)、あるいはその代わりの全角文字に置き換えます(ゲタ変換に固定)。



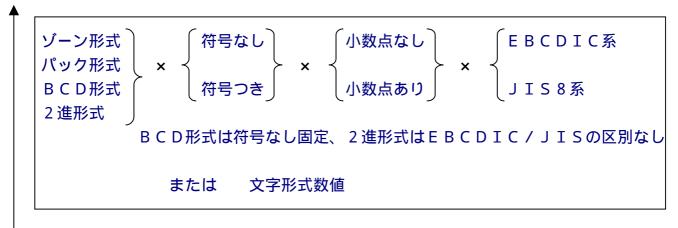
目的に応じ、2020と8140を使い分ける

2 . 20 数値変換

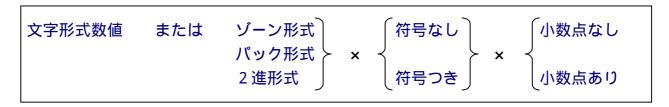
ゾーン形式とパック形式、BCD形式、2進形式、文字形式の変換をサポート

F*TRAN+は、ホストで多く使われるCOBOL数値項目のゾーン形式とパック形式、BCD形式、2進形式、文字形式の数値変換をサポートしています。

ホストファイル側



Windowsファイル側



また、Windows側は、文字形式数値、Windows COBOLのゾーン形式とパック形式、2進形式をサポートしています。ゾーン形式、パック形式、2進形式は、上記のどの組み合わせでもかまいません。

ゾーン形式とは

ゾーン形式は、COBOL数値項目のうち最も多く使われる形式です。1バイトで10進1桁を表します。外部10進数と呼ばれることもあります。文字形式数値に変換するのに都合のよい形式になっているからです。

パック形式とは

パック形式は、COBOL数値項目のうち、ゾーン形式のつぎに多く使われる形式です。1バイトで10進2桁を表す格納効率のよい形式です。内部10進数と呼ばれることもあります。直接、演算の対象にできるからです。

B C D形式とは

BCD形式は、POS端末等で使われているパック形式に似ている形式です。パック形式同様 1バイトで10進2桁を表す格納効率のよい形式ですが、符号情報を持っていません。

2進形式とは

2進形式は、COBOL数値項目、Visual Basic、<math>C/C++などで使われる形式です。 $1\sim8$ バイトで表現可能な最大値を表す最も格納効率のよい形式です。 ただし、F*TRAN+は、アプリケーションが採用しているすべての 2 進形式をサポートしているわけではありません。

F*TRAN+がサポートしている変換

F*TRAN+は、つぎの組み合わせで数値変換を行うことができます。

ホスト側		Windows側
Disp(文字形式)	/	Disp(文字形式)
Zone (ゾーン形式)	\mapsto \leftarrow	Zone(ゾーン形式)
Pack(パック形式〔BCD形式を含む〕)		Pack(パック形式)
Bin (2進形式)	Y	Bin (2進形式)

^{*)} Disp Dispは、Numeric変換

2.21 ゾーン形式(EBCDIC系)

ゾーン形式(EBCDIC系)の記録形式

たとえば、ゾーン形式7桁の数値は、下図のような形式で記録されます。

0	1	2	3	4	5	6
F n	F n	F n	F n	F n	F n	s n

1バイトのうち、上位4ビットをゾーン部と呼びます。図のFとsの部分です(Fは16進のFそのものです)。そして、下位4ビットで10進1桁を表します(図のnの部分、n=0~9)。符号つきの場合、正負の区別はふつう最後の1バイトのゾーン部に記録されます。図のsの部分です。便宜上、sの部分を符号部と呼び、Fの部分をゾーン部と呼び分けることにします。桁数や小数点を表すものは何もありません。この項目をアクセスするプログラムで、それらを指定することになっています。F*TRAN+では「ピクチャ」でそれらを指定します。

符号部(s)の値

下表のように、符号の有無、正負の区別によって符号部(s)のとる値が違います。

ゾーン形式の符号部(s)の値(EBCDIC系)

	符号部(s)
符号なし	常に F
符号あり	CまたはFなら正
	Dなら負

ゾーン形式は、各バイトを"0"~"9"の文字として扱ってもよい記録形式になっています。ただし、符号つきゾーン形式の最後の桁は例外です。EBCDICコードでは、"0"~"9"=F0~F9であることを思い出してください。符号つきゾーン形式の最後の桁だけは文字扱いできない(数字としては読めない)ことがわかります。

例) ゾーン形式(EBCDIC系)

16進表規		<u>又子表現</u>	
F0 F0 F1 F2 F3	=	1 2 3	符号なし、小数部なし
F0 F0 F1 F2 C3	=	12.3	符号つき、正、小数部あり
F0 F0 F1 F2 D3	=	- 12.3	符号つき、負、小数部あり

2.22 パック形式(EBCDIC系)

パック形式(EBCDIC系)の記録形式

たとえば、パック形式7桁の数値は、つぎのような形式で記録されます。

0		1	1 2			3	
n	n	n	n	n	n	n	S

1バイトを上位 4 ビット + 下位 4 ビットに分け、それぞれが 1 0 進 1 桁を表します(図の n の部分、 n = 0 ~ 9)。ただし、最後の 4 ビット(図の s の部分)だけは例外で、数値の記録はしません。ここには、符号なしの場合は詰めものがされ、符号つきの場合は正負の区別が記録されます。ここを、符号部と呼ぶことにします。

パック形式でもやはり、桁数や小数点を表すものは何もありません。この項目をアクセスする プログラムで、それらを指定することになっています。F*TRAN+では「ピクチャ」でそれ らを指定します。

符号部(s)の値

下表のように、符号の有無、正負の区別によって符号部(s)のとる値が違います。

パック形式の符号部(s)の値(EBCDIC系)

	符号部(s)
符号なし	常に F
符号あり	CまたはFなら正
	Dなら負

例)パック形式(EBCDIC系)

<u>16進表現</u>		文字表現	
00 12 3F	=	1 2 3	符号なし、小数部なし
00 12 3C	=	12.3	符号つき、正、小数部あり
00 12 3D	=	- 12.3	符号つき、負、小数部あり

2.23 ゾーン形式(JIS8系)

ゾーン形式(JIS8系)の記録形式

たとえば、ゾーン形式7桁の数値は、下図のような形式で記録されます。

0		1		2		3		4		5		6	
3	n	3	n	3	n	3	n	3	n	3	n	s	n

1 バイトのうち、上位 4 ビットをゾーン部と呼びます。図の 3 と s の部分です(3 は 1 6 進の 3 そのものです)。そして、下位 4 ビットで 1 0 進 1 桁を表します(図の n の部分、 n = 0 ~ 9)。 符号つきの場合、正負の区別はふつう最後の 1 バイトのゾーン部に記録されます。図の s の部分です。便宜上、 s の部分を符号部と呼び、 3 の部分をゾーン部と呼び分けることにします。

桁数や小数点を表すものは何もありません。この項目をアクセスするプログラムで、それらを 指定することになっています。 F * T R A N + では「ピクチャ」でそれらを指定します。

符号部(s)の値

下表のように、符号の有無、正負の区別によって符号部(s)のとる値が違います。

ゾーン形式の符号部(s)の値(JIS8系)

	符号部(s)
符号なし	常に3
符号あり	0または3なら正、Bなら負
	(正確には0~7なら正、8~Fなら負)

ゾーン形式は、各バイトを"0"~"9"の文字として扱ってもよい記録形式になっています。ただし、符号つきゾーン形式の最後の桁は例外です。 JIS8Jードでは、"0"~"9"=30~39であることを思い出してください。符号つきゾーン形式の最後の桁だけは文字扱いできない(負なら数字としては読めない)ことがわかります。

例) ゾーン形式(JIS8系)

<u>16進表規</u>		<u>又子表垷</u>	
30 30 31 32 33	=	1 2 3	符号なし、小数部なし
30 30 31 32 33	=	12.3	符号つき、正、小数部あり
30 30 31 32 B3	=	- 12.3	符号つき、負、小数部あり

2.24 パック形式(JIS8系)

パック形式(JIS8系)の記録形式

たとえば、パック形式7桁の数値は、つぎのような形式で記録されます。

0		1		2		3	
n	n	n	n	n	n	n	S

1バイトを上位 4 ビット + 下位 4 ビットに分け、それぞれが 1 0 進 1 桁を表します(図の n の部分、 n = 0 ~ 9)。ただし、最後の 4 ビット(図の s の部分)だけは例外で、数値の記録はしません。ここには、符号なしの場合は詰めものがされ、符号つきの場合は正負の区別が記録されます。ここを、符号部と呼ぶことにします。

パック形式でもやはり、桁数や小数点を表すものは何もありません。この項目をアクセスする プログラムで、それらを指定することになっています。F*TRAN+では「ピクチャ」でそれ らを指定します。

符号部(s)の値

下表のように、符号の有無、正負の区別によって符号部(s)のとる値が違います。

パック形式の符号部(s)の値(JIS8系)

	符号部(s)
符号なし	0 から 3 (正確には 0 ~ F のどれでもよい)
符号あり	0または3なら正、Bなら負
	(正確には0~7なら正、8~Fなら負)

例)パック形式(JIS8系)

<u>16進表現</u>		<u>文字表現</u>	
00 12 30	=	1 2 3	符号なし、小数部なし
00 12 33	=	12.3	符号つき、正、小数部あり
00 12 3B	=	- 12.3	符号つき、負、小数部あり

2 . 25 Windows COBOL のゾーン / パック形式

ゾーン形式(Windows COBOL)の記録形式

たとえば、ゾーン形式7桁の数値は、下図のような形式で記録されます。

	0	1	2	3	4	5	6
符号なし	3 n	3 n	3 n	3 n	3 n	3 n	3 n
	0	1	2	3	4	5	6
符号つき	3 n	3 n	3 n	3 n	3 n	3 n	S S

数値は、基本的に"0"~"9"の文字が使われますが、符号つきの場合、正負の区別は最後の1バイトに数値の情報とともに記録されます。図のssの部分です。

桁数や小数点を表すものは何もありません。この項目をアクセスするプログラムで、それらを 指定することになっています。 F * T R A N + では「ピクチャ」でそれらを指定します。

符号つきバイト(ss)の値

以下の2つの表に示すように、正負の区別、数値、ベンダによって符号つき(ss)のとる値が決まります。

ゾーン形式の符号つきバイト(ss)の値(数値が正の場合)

数值	富士通	日立/メラント/Aa	N E C /Ai	Acucorp -Dcm
+ 0	@(40)	0 (30)	{ (7B)	0 (30)
+ 1	A (41)	1 (31)	A (41)	1 (31)
+ 2	B (42)	2 (3 2)	B (42)	2 (3 2)
+ 3	C (43)	3 (33)	C (43)	3 (33)
+ 4	D(44)	4 (34)	D (44)	4 (3 4)
+ 5	E (45)	5 (35)	E (45)	5 (3 5)
+ 6	F (46)	6 (36)	F (46)	6 (36)
+ 7	G(47)	7 (37)	G (47)	7 (37)
+ 8	H (48)	8 (38)	H (48)	8 (38)
+ 9	I(49)	9 (39)	I(49)	9 (39)

Aa = Acucorp -Dca, Ai = Acucorp -Dci

ゾーン形式の符号つきバイト(ss)の値(数値が負の場合)

数值	富士通	日立/メラント/Aa	N E C /Ai	Acucorp -Dcm
- 0	P(50)	p(70)	}(7D)	}(7D)
- 1	Q(51)	q (71)	J (4A)	J (4A)
- 2	R (52)	r (72)	K (4B)	K (4B)
- 3	S (53)	s (73)	L (4C)	L (4C)
- 4	T (54)	t (74)	M (4D)	M (4 D)
- 5	U (55)	u (75)	N (4E)	N (4 E)
- 6	V (56)	v (76)	O(4F)	O(4F)
- 7	W (57)	w (77)	P (50)	P(50)
- 8	X (58)	x (78)	Q(51)	Q(51)
- 9	Y (59)	y (79)	R (52)	R (52)

Aa = Acucorp -Dca, Ai = Acucorp -Dci

パック形式 (Windows COBOL) の記録形式

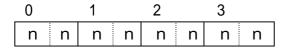
Windows COBOLのパック形式は、EBCDIC系と同じです。

2.22 パック形式(EBCDIC系)を参照してください。

2.26 BCD形式

BCD形式の記録形式

たとえば、BCD形式8桁の数値は、つぎのような形式で記録されます。



1 バイトを上位 4 ビット + 下位 4 ビットに分け、それぞれが 1 0 進 1 桁を表します(図のnの部分、n=0~9)。

BCD形式はパック形式と似ていますが、符号情報を持たない形式です。

BCD形式でもやはり、桁数や小数点を表すものは何もありません。この項目をアクセスするプログラムで、それらを指定することになっています。F*TRAN+では「ピクチャ」でそれらを指定します。

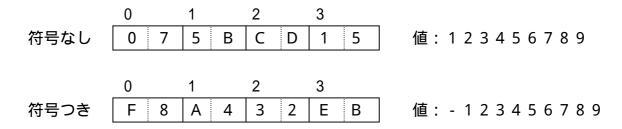
例)BCD形式

16進表現文字表現00 01 23= 123小数部なし00 01 23= 12.3小数部あり

2.27 2 進形式

2 進形式の記録形式

たとえば、2進形式9桁の数値は、下図のような形式で記録されます。



2 進形式は、少ないバイト数で大きな数値を表現可能な最も格納効率のよい形式です。そのため、格納された数値を確認するには専門的な知識が必要となり、判断が難しいという側面があります。一般には、アプリケーションの内部で処理されることが多い形式です。

F * T R A N + でサポートする 2 進形式は、

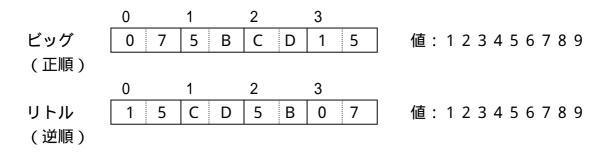
1~8バイトの範囲で表現できる整数/小数(マイナス値を含む)

になり、COBOLのCOMP - 5やVisual Baisc、C/C++などの内部変数の形式がこの対象になります。

2 進形式は、桁数や小数点を表すものは何もありません。この項目をアクセスするプログラムで、それらを指定することになっています。F * T R A N + では「2 進ピクチャ」でそれらを指定します。

エンディアン(格納順)

2 進形式の場合、気をつけなければならないのはエンディアン(格納順)がある事です。エンディアンの違いを図にすると、つぎのようになります。



一般には、ホスト側がビッグ(正順)で、Win側がリトル(逆順)です。

2 . 28 パック/BCD/2 進形式の桁数とバイト数の換算

パック形式の桁数とバイト数の換算

パック形式の数値の桁数(整数部桁数 + 小数部桁数)とバイト数の換算は、下表を参考にしてください。

パック形式の桁数とバイト数の換算表

桁数	バイト数
1	1
2 ~ 3	2
4 ~ 5	3
6 ~ 7	4
8 ~ 9	5
10~11	6
12~13	7
14~15	8
16~17	9
1 8	1 0

BCD形式の桁数とバイト数の換算

BCD形式の数値の桁数(整数部桁数 + 小数部桁数)とバイト数の換算は、下表を参考にしてください。

BCD形式の桁数とバイト数の換算表

桁 数	バイト数
1 ~ 2	1
3 ~ 4	2
5 ~ 6	3
7 ~ 8	4
9 ~ 1 0	5
11~12	6
13~14	7
15~16	8
17~18	9

2 進形式で表現可能な数値の範囲

2 進形式で表現可能な数値の範囲は、つぎのとおりです。

符号なし2進形式の場合

バイト数	値の範囲	桁数
1	0 ~ 255	3
2	0 ~ 65,535	5
3	0~16,777,215	8
4	0~4,294,967,295	1 0
5	0~1,099,511,627,775	1 3
6	0~281,474,976,710,655	1 5
7	0~72,057,594,037,927,935	1 7
8	0~18,446,744,073,709,551,615	1 8 *

符号つき 2 進形式の場合

バイト数	値の範囲	桁数
1	-128 ~ 127	3
2	-32,768 ~ 32,767	5
3	-8,388,608 ~ 8,388,607	7
4	-2,147,483,648 ~ 2,147,483,647	1 0
5	-549,755,813,888 ~ 549,755,813,887	1 2
6	-140,737,488,355,328 ~ 140,737,488,355,327	1 5
7	-36,028,797,018,963,968 ~ 36,028,797,018,963,967	1 7
8	-9,223,372,036,854,775,808~9,223,372,036,854,775,807	1 8 *

^{*) 18}桁ではオーバーフローするが、COBOL規格/F*TRAN+の上限です。

2 . 29 **ホスト** Win テキストファイル変換のしくみ

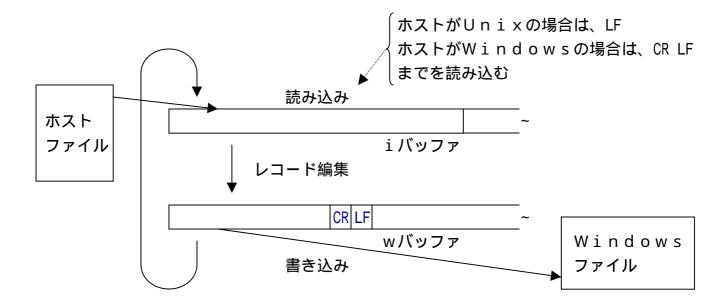
ホスト Winテキストファイル変換は、文字データだけからなるホストファイルをWindowsのテキストファイルに変換します。おもに、ソースプログラムの変換に使用します。ホストが汎用機・オフコンの場合、ホストファイル側に漢字があるときはKI/KOがついていることが前提となり、KI/KOなしのファイルを項目別に変換するときは、ホスト Winデータファイル変換を使います。

解説

テキストファイル変換の考え方

まず、ホストファイルの1レコードを保持するiバッファと、Windowsファイルの1レコードを保持するwバッファを考えます。iバッファ、wバッファは十分な大きさ(約32Kバイト)があると考えてください。

下図のようなサイクルでファイル変換が行われます。



レコード編集

ホストファイルの1レコードがWindowsファイルの1行に対応し、レコードごとに、

ホストがUnixの場合、行末のLFを取る

ホストがWindowsの場合、行末のCR、LFを取る

ホストがUnix、Windowsの場合、

TABをタブ位置の前までのスペースに変換する(オプション指定:タブ拡張)

コード変換する (ホストが汎用機・オフコンの場合、

オプション指定: ANKのみ/ANK・漢字 (KI/KO付) 混在)

レコード末尾の空白類を削除する(行末圧縮)

空白類:スペース(20H、8140H) NUL(00H)

TAB(09H), CR(0DH), LF(0AH)

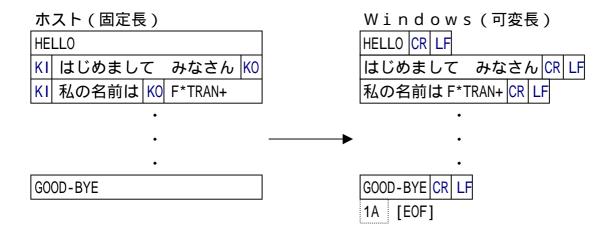
途中のNULを除去する(NULサプレス)

2個以上連続するスペースをTABに置き替える(オプション指定:タブ圧縮)

最後に、改行コード(CR/LF=0D0AH)を付加する

の順で加工・編集されます。

実際には、コード変換が最初に行われ、その後残りの作業が行われるので、ANK変換表を修正して制御コードや未定義コードなどを取り除くことは可能です(それらがNULに変換されるようにします)。



このように、ホスト Winテキストファイル変換では基本的には圧縮がかかるので、レコード長はふぞろい(可変長)になります。固定長のテキストファイルが必要なときは、ホスト Winデータファイル変換を使ってください。

ここではホストが汎用機・オフコンの例をあげていますが、ホストがUnix、Windowsの場合は、ホストファイルが可変長となり、Unixの場合はLF、Windowsの場合は CR LFまでを1レコードとしたKI/KOのないデータが対象になります。

2 . 30 **ホスト** Win データファイル変換のしくみ

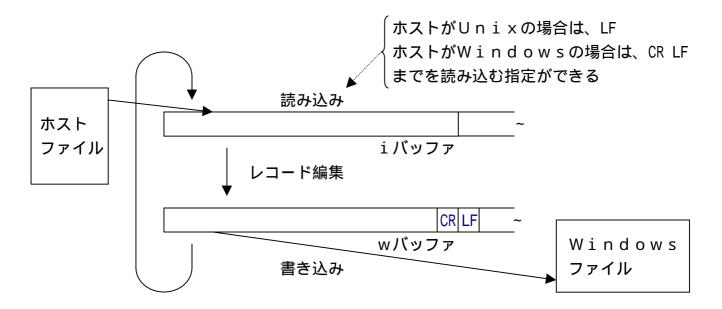
ホスト Winデータファイル変換は、ホストのデータファイル(おもにCOBOLデータ)をWindowsのデータファイル(テキスト形式)に変換します。項目別に分けて変換できます。Windowsファイルは常に改行コード(CR/LF=0D0AH)つきのテキストファイルになります。プリント形式への変換とデリミタ形式への変換の、2つの変換方法があります。

解説

データファイル変換の考え方

まず、ホストファイルの1レコードを保持するiバッファと、Windowsファイルの1レコードを保持するwバッファを考えます。iバッファ、wバッファは十分な大きさ(約32Kバイト)があると考えてください。

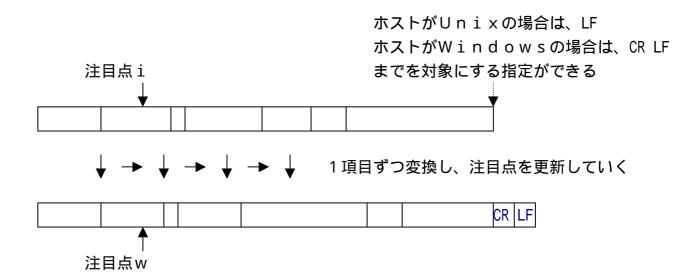
下図のようなサイクルでファイル変換が行われます。



レコード編集の考え方

F*TRAN+は、マップ設定オプションの指定を頭から順番に見ていき、それを中間コードに翻訳して「マップバッファ」と呼ぶ領域に収めます。1件ずつのレコード編集は、この中間コードを見ながら行われるので、高速なレコード編集ができるようになっています。

右図のように、i バッファの上には注目点 i があり、w バッファの上には注目点 w があると考えます。レコード編集とは、1 項目ずつコード変換などを行い、指定に応じて注目点を更新していく作業なのです。



ホスト Winデータファイル変換は、レコード編集の最初の作業として、注目点iと注目点wを、それぞれiバッファ、wバッファの先頭に位置づけます。そして、wバッファをスペースでクリアします。

すべての項目の変換がおわると、wバッファの末尾(最終項目)に改行コード(CR/LF=0D0AH)を付加します。そして、デリミタ形式への変換で圧縮の指定があれば、さらに不要スペースの圧縮を行います。

2 . 31 **ホスト** Win ランダムファイル変換のしくみ

ホスト Winランダムファイル変換は、ホストのデータファイル(おもにCOBOLのデータ)をWindowsのランダムファイル(固定長ファイル)に変換します。おもにWindows COBOLの順ファイルやBASICのランダムファイルに変換するのに使います。項目別変換ができます。作成されるWindowsファイルは固定長で、改行コードもデリミタもないファイルになります。ただし、改行コードの挿入はできます。

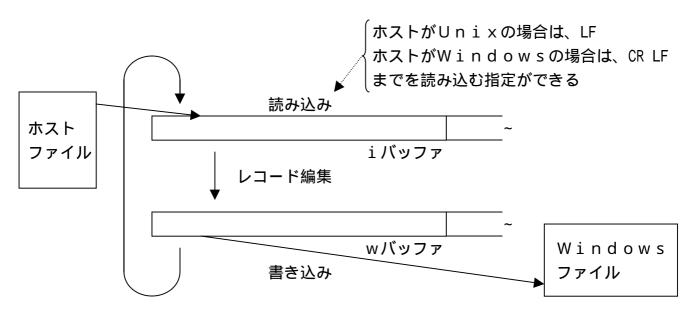
そのほかバイナリ変換もできます。アップロードデータを作成するのにも便利で、オンラインファイル転送と組み合わせて利用できます。

解説

ランダムファイル変換の考え方

まず、ホストファイルの1レコードを保持するiバッファと、Windowsファイルの1レコードを保持するwバッファを考えます。iバッファ、wバッファは十分な大きさ(約32Kバイト)があると考えてください。

下図のようなサイクルでファイル変換が行われます。

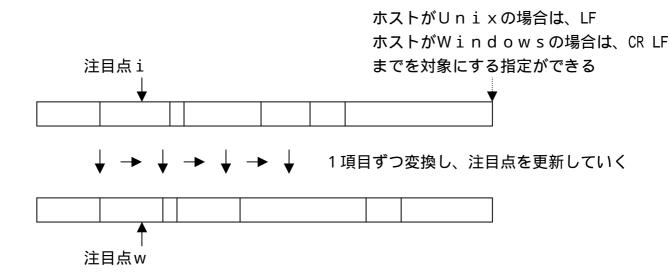


レコード編集の考え方

F*TRAN+は、マップ設定オプションの指定を頭から順番に見ていき、それを中間コードに翻訳して「マップバッファ」と呼ぶ領域に収めます。

1件ずつのレコード編集は、この中間コードを見ながら行われるので、高速なレコード編集ができるようになっています。

右図のように、i バッファの上には注目点 i があり、w バッファの上には注目点 w があると考えます。レコード編集とは、1 項目ずつコード変換などを行い、指定に応じて注目点を更新していく作業なのです。

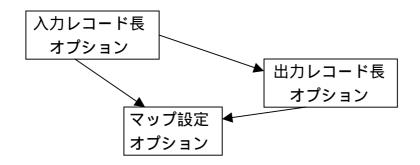


ホスト Winランダムファイル変換は、レコード編集の最初に注目点iと注目点wをそれぞれiバッファ、wバッファの先頭に位置づけます。そして、wバッファをNUL(00H)でクリアします。

すべての項目の変換がおわると、それ以上の加工はせずに、Windowsファイルに出力します。

レコード長の参照関係

ホスト Winランダムファイル変換は、レコード長の参照関係が少し複雑です。それを、下図に示します。



2 . 32 Win **ホストテキストファイル**変換のしくみ

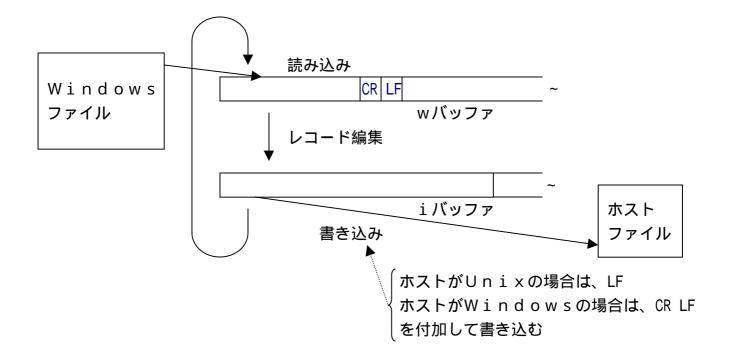
Win ホストテキストファイル変換は、ソースプログラムのようなWindowsファイルをホストファイルに変換します。Windowsファイルはテキストファイル(改行コードつき)でなければなりません。改行コードでレコードのおわりを検出するので、可変長でかまいません。ホストが汎用機・オフコンの場合は、ホストファイルは固定長になり、ホストがUnix、Windowsの場合は、ホストファイルは可変長になります。

解説

テキストファイル変換の考え方

まず、Windowsファイルの1レコードを保持するwバッファと、ホストファイルの1レコードを保持するiバッファを考えます。wバッファ、iバッファは十分な大きさ(約32Kバイト)があると考えてください。

下図のようなサイクルでファイル変換が行われます。



レコード編集

Windowsファイルの1行(CR/LFまで)がホストファイルの1レコードに対応し、レコードごとに、

行末のCR、LFを取る

TABをタブ位置の前までのスペースに変換する(オプション指定:タブ拡張)

コード変換する (ホストが汎用機・オフコンの場合、

オプション指定: ANKのみ/ANK・漢字 (KI/KO付) 混在)

ホストがUnix、Windowsの場合、コード末尾の空白類を削除する(行末圧縮)

空白類:スペース(20H、8140H) NUL(00H)

TAB(09H) CR(0DH) LF(0AH)

NULを除去する(NULサプレス)

ホストがUnix、Windowsの場合、

2個以上連続するスペースをTABに置き替える(オプション指定:タブ圧縮)

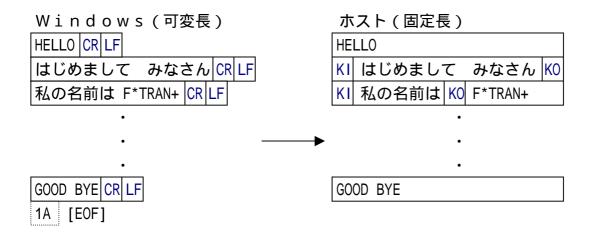
ホストが汎用機・オフコンの場合、レコード末尾にスペースを詰める(行末パディング)

ホストがUnixの場合、レコード末尾にLFを付ける

ホストがWindowsの場合、レコード末尾にCR、LFを付ける

の順で加工・編集されます。

実際には、コード変換が最初に行われ、その後残りの作業が行われるので、ANK変換表を修正して制御文字や未定義文字などを取り除くことは可能です(それらがNULに変換されるようにします)。



このように、Win ホストテキストファイル変換では基本的には可変長のテキストファイル を扱いますが、単純な固定長テキストファイルの変換に使うこともできます(Win ホストデータファイル変換を使うほうが適切ですが)。

ここではホストが汎用機・オフコンの例をあげていますが、ホストがUnix、Windowsの場合は、ホストファイルが可変長となり、Unixの場合はLF、Windowsの場合は CR LFを付加したKI/KOのないデータになります。

2 . 33 Win **ホストデータファイル**変換のしくみ

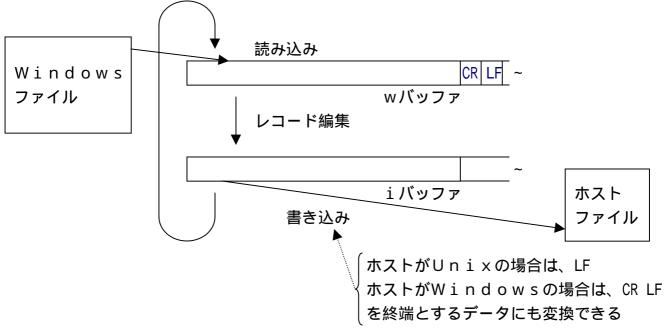
Win ホストデータファイル変換は、Windowsのデータファイルをホストファイルに変換します。Windowsファイルはプリント形式のファイルかデリミタ形式のファイルでなければいけません。ANKデータだけの単純な変換、項目別の変換などが行えます。なお、ホストが汎用機・オフコンの場合、ホストファイルは必ず固定長・固定欄になります。

解説

データファイル変換の考え方

まず、Windowsファイルの1レコードを保持するwバッファと、ホストファイルの1レコードを保持するiバッファを考えます。wバッファ、iバッファは十分な大きさ(約32Kバイト)があると考えてください。

下図のようなサイクルでファイル変換が行われます。

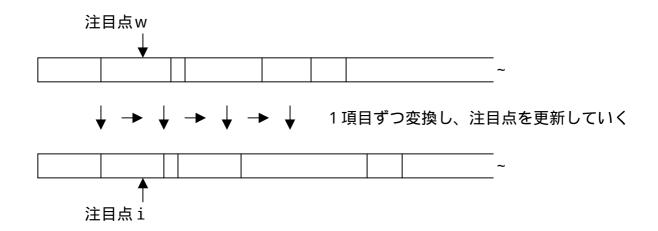


空行(改行/改頁コードだけの行)は、変換の対象にする/しない選択可

レコード編集の考え方

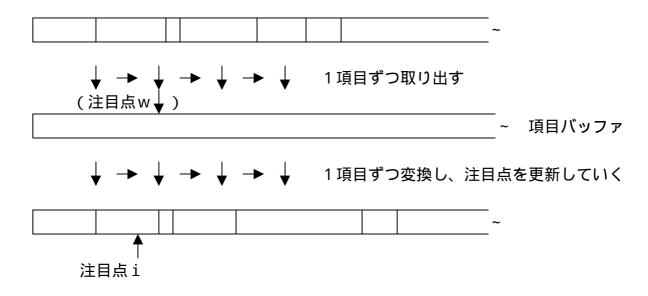
F*TRAN+は、マップ設定オプションの指定を頭から順番に見ていき、それを中間コードに翻訳して「マップバッファ」と呼ぶ領域に収めます。1件ずつのレコード編集は、この中間コードを見ながら行われるので、高速なレコード編集ができるようになっています。

まず、プリント形式からの変換を考えます。右図のように、wバッファの上には注目点wがあり、iバッファの上には注目点iがあると考えます。レコード編集とは、1項目ずつコード変換などを行い、指定に応じて注目点を更新していく作業なのです。



Win ホストデータファイル変換は、レコード編集の最初の作業として、注目点wと注目点iを、それぞれwバッファ、iバッファの先頭に位置づけます。そして、iバッファをスペースでクリアします。すべての項目の変換がおわると、iバッファの内容をホストファイルに書き込みます。

デリミタ形式からの変換の場合は、wバッファとiバッファの間に「項目バッファ」があると考えるとわかりやすくなります。



各項目の変換のたびに項目を項目バッファに取り込み、うしろはスペースでクリアします。そして、それに対して変換をかけます。注目点wは、あまり重要な働きはしません。一方、注目点iのほうは、1項目ずつ変換するたびに更新していきます。

2 . 34 Win **ホストランダムファイル**変換のしくみ

Win ホストランダムファイル変換は、Windowsのランダムファイル(固定長ファイル)をホストファイルに変換します。固定長レコードの単純な変換、固定長・固定欄形式のファイルの項目別変換などが行えます。バイナリ変換もできます。

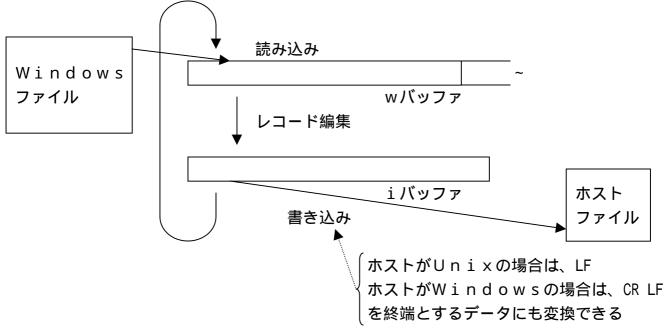
固定長・固定欄なら、改行コードのついているテキストファイルも扱えますが、その場合にはWin ホストデータファイル変換のプリント形式からの変換機能を使ったほうがずっと簡単です。

解説

ランダムファイル変換の考え方

まず、Windowsファイルの1レコードを保持するwバッファと、ホストファイルの1レコードを保持するiバッファを考えます。wバッファ、iバッファは十分な大きさ(約32 Kバイト)があると考えてください。

下図のようなサイクルでファイル変換が行われます。

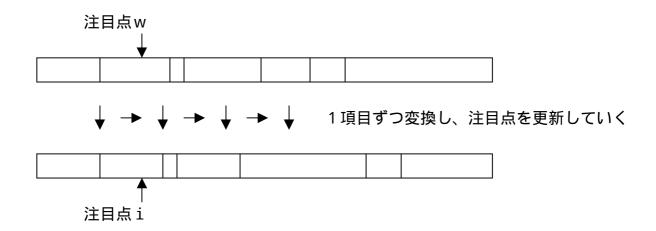


レコード編集の考え方

F*TRAN+は、マップ設定オプションの指定を頭から順番に見ていき、それを中間コードに翻訳して「マップバッファ」と呼ぶ領域に収めます。

1件ずつのレコード編集は、この中間コードを見ながら行われるので、高速なレコード編集ができるようになっています。

右図のように、wバッファの上には注目点wがあり、iバッファの上には注目点iがあると考えます。レコード編集とは、1項目ずつコード変換などを行い、指定に応じて注目点を更新していく作業なのです。

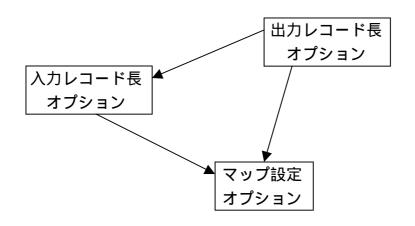


Win ホストランダムファイル変換は、レコード編集の最初の作業として、注目点wと注目点iを、それぞれwバッファ、iバッファの先頭に位置づけます。そして、iバッファをNUL (00H)でクリアします。

すべての項目を変換すると、それ以上は加工しないでホストファイルに出力します。

レコード長の参照関係

Win ホストランダムファイル変換は、レコード長の参照関係が少し複雑です。それを、以下に示します。



第3章

操作の基礎

3.1 ファイルの指定

指定形式

ファイルはつぎの形式で指定します。

[d:][パス名指定][基本ファイル名[.拡張子]]

d:はドライブ名

ドライブ名(d:)

ドライブ名(d:)には、

A: ~ Z: 実ドライブ名を指定します。

また、F*TRAN+内でのみ通用するドライブ名として、

@: カレントドライブのカレントディレクトリを表す

?: インストールディレクトリを表す

も指定できます。

ドライブ名は省略可能です。省略するとカレントドライブを指定したものとみなされます。

パス名指定(¥ディレクトリ名¥サブディレクトリ名¥・・・)

パス名指定(¥やディレクトリ名を使う指定)ができます。指定したディレクトリ配下のファイルを扱うことができます。

基本ファイル名と拡張子

大部分のコマンドでは、入力側のファイル名を指定する場合、基本ファイル名と拡張子にワイルドカード文字(*、?)が使用できます。

出力側のファイル名を指定する場合、基本ファイル名と拡張子にはふつうの基本ファイル名や拡張子以外に、*も指定できます。*は「入力側の基本ファイル名や拡張子を引き継げ」という意味です。ふつうは、

d:*.dat のように、拡張子を指定します。

こうすると、指定ドライブに

元の基本ファイル名.拡張子 というファイルができます。

なお、できる限り適切な拡張子をつけるように心がけてください。

拡張子の省略値

ファイル名の指定ができるところでも、拡張子の省略値が、

. C C T コード変換表ファイルの、省略時の拡張子. P パラメータファイルの、省略時の拡張子

のように決まっているものもあります。

3.2 式の指定

F*TRAN+では、オプションデータなどのパラメータを10進数で指定できるところで、ほとんどの場合、10進数の代わりに値を「式」で指定することができます。計算を省いたり、レコード長のようなその都度変わるデータに対しても同じ指定ですむようにしたり、相対的な指定を可能にしたりするためです。

式とは

式といっても、複雑な数式のようなものを指定できるわけではありません。10進数を、四則 演算子とカッコで組み合わせられるだけの単純なものです。一部のオプションではさらに特殊変 数を使うこともできます。

四則演算子

四則演算子はつぎの5つです。

+ 加算
- 減算
× (小文字の×) または * 乗算
% または // 除算
¥ ¥ 剰余(割算の余り)

演算子間の優先順位はありません。xを+-より先に計算したりしないということです。カッコを使って計算の順番を明示してください。カッコは何重にも入れ子にできます。

特殊変数

特殊変数にはつぎの6つがあります。

\$ 最大値を表す (ふつうはレコード長を意味する) . (ピリオド) 現在値を表す (ふつうはレコードの現在の桁位置を意味する) 残りを表す (ふつうはレコードの残りの長さを意味する) レコード番号 (ふつうはマルチレコードの指定時に使用する) ~SysRecNum ~SysReturn リターン値 (ふつうはマルチレコードの指定時に使用する) ~SysBreak ブレーク値 (ふつうはマルチレコードの指定時に使用する) 式の例を示します。つぎに示すのはいずれも正しい式です。

```
$
7
(15)
                    . - 15
                                         $ x 4
4 + (6 + 8)
                                        $ + 2
                    . + 2
256 x 26
                                        $ % 3
1024×4
                    * - 2
                                        $ x (( $ % 8 0 ) + 1 )
                    *//3
* * 3
                                        * ¥ ¥ 3
~SysRecNum
                    ~SysRecNum¥¥3
```

注意 ---- 式に空白を入れてはならない

演算子やカッコなどの前後に空白を入れてはいけません。空白はパラメータ類の区切りを意味 するからです。

3.3 ピクチャの指定

ピクチャとは

ピクチャとは、СОВО L のゾーン / パック形式や、ВС D 形式の数値項目を変換するとき、

符号の有無

数値の桁数

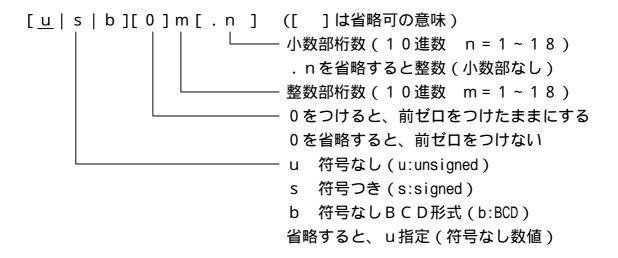
前ゼロの有無

小数部の有無と桁数

を指定するためのものです。これらは、データ自体には記録されていないので、外からこれらの情報を与える必要があります。

ピクチャの指定形式

ピクチャはつぎの形式で指定します。



COBOLをまねた

F*TRAN+のピクチャは、COBOLのピクチャ指定をまねた上で、大幅に簡略化したものです。たとえば、

- 12.3

という数字があって、5バイトのゾーン形式の項目に記録してあるとします。 COBOLのピクチャなら

PIC S9(4)V9(1)

のように指定します。

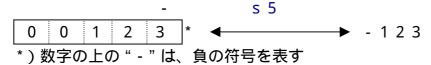
F*TRAN+のピクチャではこれを、 s4.1 と指定します。

ピクチャの指定例を示します。以下の図の左側がゾーン形式の項目、右側が文字形式数値、そ して矢印の上がピクチャです。

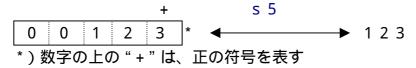
例1)符号なし整数



例2)符号つきの、負の整数



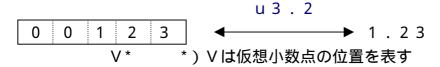
例3)符号つきの、正の整数



例4)符号なし整数、前ゼロをつけたままにする

例5)符号つき整数、前ゼロをつけたままにする

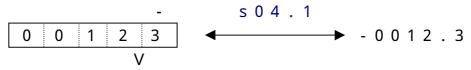
例6)小数部(小数点)がある/その1



例7)小数部(小数点)がある/その2



例8)符号も小数部(小数点)もあり、前ゼロもつけたままにする



3.4 2 進ピクチャの指定

2 進ピクチャとは

2 進ピクチャとは、COBOL、Visual Basic、C/C++などの2 進数値項目を変換するとき、

バイト数

格納順

符号の有無

数値の桁数

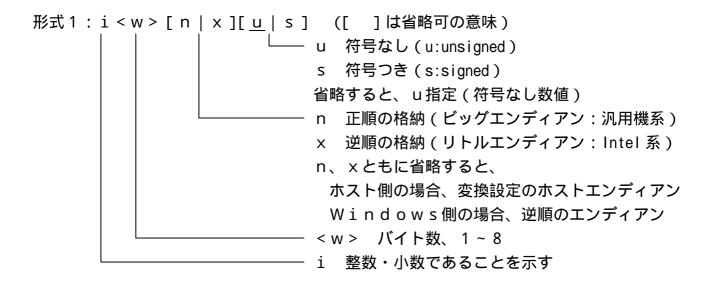
前ゼロの有無

小数部の有無と桁数

を指定するためのものです。これらは、データ自体には記録されていないので、外からこれらの情報を与える必要があります。

2 進ピクチャの指定形式

2 進ピクチャはつぎのどちらかの形式で指定します。



形式 2 : <u>i < w > [n | x] [{ u | s }[0] < m > [. < n >]]</u> ([] は省略可の意味) 2 進キャスト ピクチャ

0 前ゼロ付加

<m> 整数部桁数、1~18

< n > 小数部桁数、1~18、省略すると0

<m>+ < n > が1~18になること

<w>に応じて、<m>+<n>の省略値が定まる 106

各言語の型と2進ピクチャの対応

言語	型	2 進ピクチャ
COBOL	BINARY	i1nu~i8nu
(例)	SつきBINARY	i 1 n s ~ i 8 n s
	C O M P - 5	i1xu~i8xu
	SつきCOMP-5	i1xs~i8xs
Visual	Byte	i 1 u
Basic	Integer	i 2 s
	Long	i 4 s
	Currency	i 8 s 1 4 . 4
C / C + +	signed char	i 1 s
	unsigned char	i 1 u
	signed short	i 2 s
	unsigned short	i 2 u
	signed int	i 4 s
	unsigned int	i 4 u
	signed long	i 4 s
	unsigned long	i 4 u
	signedint64	i 8 s
	unsignedint64	i 8 u

3 . 5 日付データの指定

日付マスク

F*TRAN+で使用できる日付データの編集指定はつぎのとおりです。

日付マスク	データ例	日付マスク	データ例						
	1998/12/31		1998/12						
y y y y - mm - d d	1998-12-31	y y y y - mm	1998-12						
*1	1998.12.31	*1	1998.12						
	98/12/31		98/12						
y y - mm - d d	98-12-31	yy-mm	98-12						
	98.12.31		98.12						
	H10/12/31		H10/12						
nyy-mm-dd	H10-12-31	nyy-mm	H10-12						
	H10.12.31		H10.12						
y y y mm d d	19981231	y y y y m m	199812						
y y mm d d	981231	y y m m	9812						
	12/31/1998		12/1998						
mm - dd - yyyy	12-31-1998	mm - yyyy	12-1998						
*1	12.31.1998	*1	12.1998	n = 年号					
	12/31/98		12/98	M(明治)1868-1911					
mm - dd - yy	12-31-98	mm - y y	12-98	T (大正)1912-1925					
	12.31.98		12.98	S(昭和)1926-1988					
mmddyyyy	12311998	mm y y y y	121998	H(平成)1989-					
mmddyy	123198	mm y y	1298						
	31/12/1998	уууу *1	1998	g = 元号					
dd-mm-yyyy	31-12-1998	уу	98	1(明治)1868-1911					
*1	31.12.1998	n y y	H10	2 (大正)1912-1925					
	31/12/98	gyymmdd	4101231	3(昭和)1926-1988					
d d - mm - y y	31-12-98	gyymm	41012	4(平成)1989-					
	31.12.98	g у у	410						
d d mm y y y y	31121998	注意 出力時、和暦の年号 / 元号の最終年							
d d mm y y	311298	は、次年号 / 元号の元年(01)になる							

実際には、日付マスク分の長さが編集対象になります。たとえば" y y y mmdd"と指定すれば、8 バイトのデータの編集を行います。

入力側に"yy-mm-dd"のような日付区切りのある指定をした場合は、8バイトの内容が"98 12,31"であっても、"98-12-31"と同等のデータとして扱います。つまり、数字(0~9)以外の文字を日付区切り記号とみなします。

ウインドウ方式とシフト方式

日付データの年の2桁(yy)と4桁(yyy)の交換を行う場合、F*TRAN+ではウインドウ方式とシフト方式の概念を採用しています。

ウインドウ方式とは、19××年(基準年)から100年として扱う方式です。ウインドウ方式で"30"と指定すれば、実際のデータはつぎのようになります。

入力側に *1 の日付マスク指定(日付区切りがある4桁の年指定)をし、実際の日付データの年が2桁以下であった場合は、無条件にウインドウ方式による拡張を行います。

日付区切り記号

日付データを出力する際に指定できる日付区切り記号はつぎのとおりです。

日付区切り記号_		<u>デ_</u>			_			タ		例	
/ (スラッシュ)	1	9	9	8	/	1	2	/	3	1	
- (ハイフン)	1	9	9	8	-	1	2	-	3	1	
.(ピリオド)	1	9	9	8		1	2		3	1	

マップ設定オプション指定

日付データを変換する手順はつぎのとおりです。

年設定(日付データ2桁の年の扱い、ウインドウ方式またはシフト方式の指定) 日付区切り設定(日付データ出力時の日付区切り記号の指定) 日付項目変換(日付データ変換時の日付マスク指定)

、 を省略すると、年設定はウインドウ方式で 1 9 3 0 年より (入力、出力とも) "/"で日付区切りとなります。

第4章

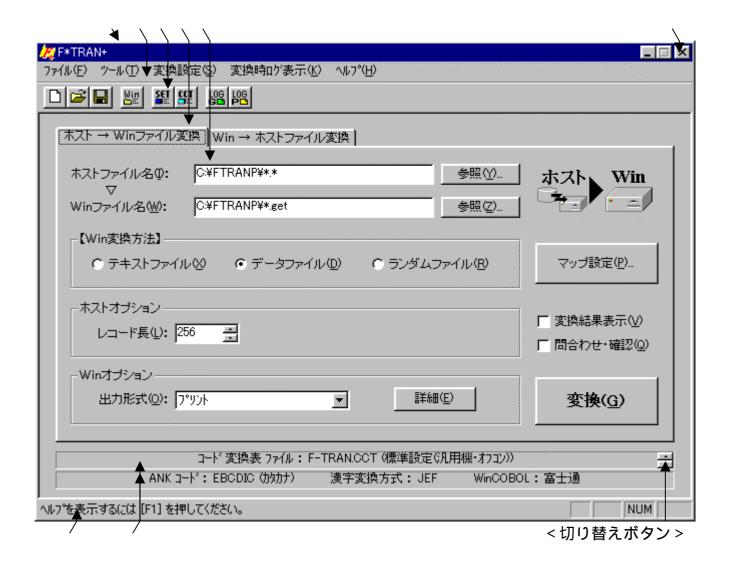
操作

4.1 F*TRAN+ 起動と終了

F*TRAN+の起動

F*TRAN+をウインドウ設定方式で使う場合の起動方法について説明します。

通常、スタートメニューより、プログラム (\underline{P}) F-TRAN+をクリックします。 起動を行うと、つぎのメインウインドウが開きます。 < ホストが汎用機・オフコンの例 >



タイトルバー ------ ファイルメニューで開いたファイル名が表示されます。

メニューバー ------ 目的のメニューを選択して、機能を実行します。

ツールバー ------ の中で、よく使われる機能がボタン化されています。

タブコントロール ---- 2 通りのメイン処理部の切り替えを行います。

メイン処理部 ------ の切り替えによって、メイン処理部の内容が変わります。

ステータスバー ----- 操作のガイダンスやメッセージが表示されます。

設定表示バー ------ おもにコード変換に関する現在の設定が表示されています。

"設定表示バー"は、メインウインドウおよび実行ウインドウ(変換実行時のウインドウ)の下方にあり、コード変換に関する現在の設定等をいつでも確認できるようになっています。表にでていない情報は、切り替えボタンをクリックして見ることができます。内容はつぎのとおりです。

コード変換表ファイル : F-TRAN.CCT (標準設定)

(「ホスト選択」により選択されたコード変換表ファイル名とコメント)

インストールディレクトリ : C:\ftranp

(F*TRAN+mTVX-mFTVD)

ANKコード : EBCDIC (かか) < ホストが汎用機・オフコンの場合 >

(「変換設定」により設定されたホストのANKコード)

漢字变換方式 : JEF

(「変換設定」により設定された漢字変換方式)

WinCOBOL : 富士通

(「変換設定」により設定されたWindows COBOLのベンダ名)

ホストCOBOL : 富士通 <ホストがUnix、Windowsの場合>

(「変換設定」により設定されたホストCOBOLのベンダ名)

F*TRAN+の終了

F*TRAN+をウインドウ設定方式で使う場合の終了の方法について説明します。

ふつう、 メニューバーのファイル (\underline{F}) 終了 (\underline{X}) を実行します。 または の $\boxed{\mathsf{x}}$ をクリックしても終了します。

コード変換表の保存

コード変換表に何らかの修正がかかった状態で、F * T R A N + を終了しようとすると、修正のかかったコード変換表を保存するか否かをたずねるメッセージウインドウが開きます。

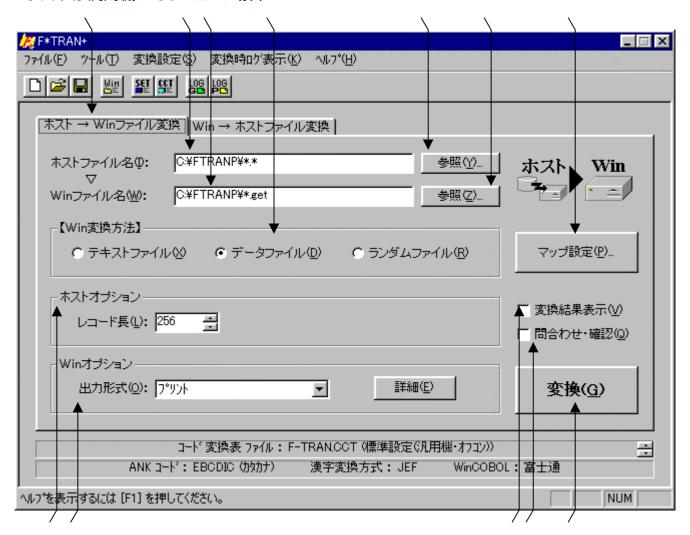


はい(\underline{Y})をクリックすると、修正のかかったコード変換表を保存します。 いいえ(\underline{N})をクリックすると、修正のかかったコード変換表を保存しません。 キャンセルをクリックすると、F*TRAN+の終了をキャンセルします。

4 . 2 ホスト Win ファイル変換のファイル指定と共通オプション

ここでは、ホスト Winファイル変換の共通事項を説明します。

ホストが汎用機・オフコンの場合



ホスト Winのタブをクリックし、メイン処理部をホスト Winファイル変換にします。 変換元のホストファイル名を指定します。

変換元のホストファイル名を指定する参照ウインドウを開くボタンです。

変換先のWindowsファイル名を指定します。

変換先のWindowsファイル名を指定する参照ウインドウを開くボタンです。

変換方法を選択するラジオボタンです。

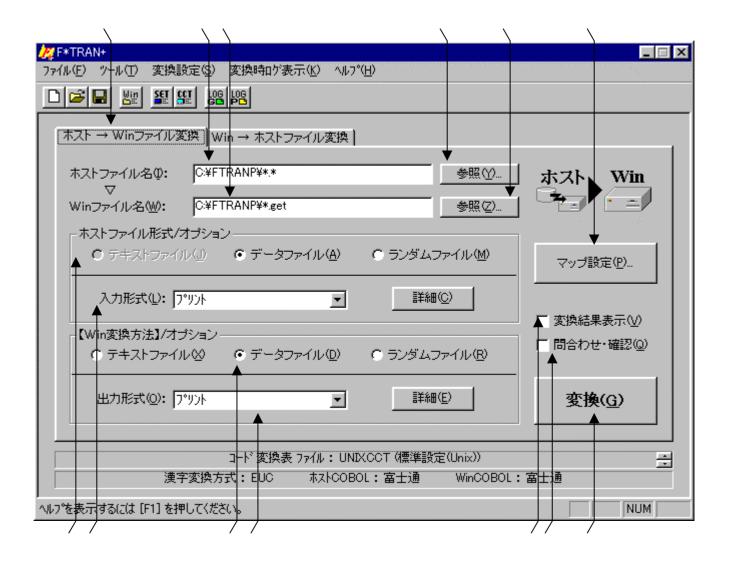
ホストファイルのオプション設定です。 の選択によって、オプションの内容が変わります。 Winファイルのオプション設定です。 の選択によって、オプションの内容が変わります。 マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換実行時の問い合わせ・確認ウインドウ表示の選択をするチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホストがUnix、Windowsの場合



ホスト Winのタブをクリックし、メイン処理部をホスト Winファイル変換にします。 変換元のホストファイル名を指定します。

変換元のホストファイル名を指定する参照ウインドウを開くボタンです。

変換先のWindowsファイル名を指定します。

変換先のWindowsファイル名を指定する参照ウインドウを開くボタンです。

ホストファイル (入力ファイル)の形式を選択するラジオボタンです。

ホストファイルのオプション設定です。

Winファイルの変換方法を選択するラジオボタンです。変換の基準となる項目です。

Winファイルのオプション設定です。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換実行時の問い合わせ・確認ウインドウ表示の選択をするチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホストファイル指定

[d:][パス名指定]基本ファイル名[.拡張子]

d: はドライブ名

入力側のホストファイルを指定します。

ドライブ名は、A:~Z:、@:、?:のどれかで指定します。ドライブ名を指定すると、そのドライブを検索します。

ドライブ名は省略可能です。ドライブ名を省略すると、カレントドライブを検索します。

パス名指定(¥ディレクトリ名¥サブディレクトリ名¥・・・)ができます。指定したディレクトリ配下のファイルを検索します。パス名指定を省略すると、カレントディレクトリを検索します。

基本ファイル名と拡張子には、ワイルドカード文字(*と?)を使うことができます。ワイルドカード文字を使うと、一致するファイルをすべて検索し変換の対象にします。

まとめると、あるディレクトリのファイルをすべて変換したいなら、

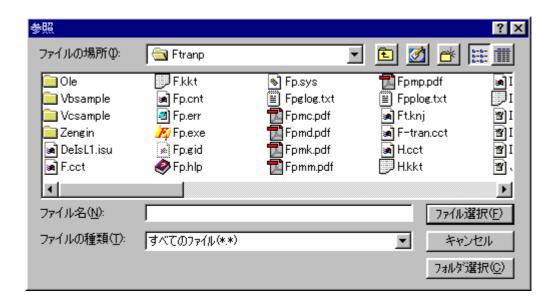
C:*.* のように指定し、

拡張子が. DATのファイルをすべて変換したいなら、

C:*.DAT のような指定になります。

ホストファイル参照ウインドウ

ホストファイルの参照ボタンをクリックすると、つぎのウインドウが開きます。



変換元のファイルを直接指定する場合は、ファイル名を選択して、ファイル選択(<u>F</u>)ボタンをクリックします。

変換元の場所(どこのフォルダのファイルを変換の対象にするか)を指定する場合は、フォルダ選択(<u>C</u>)ボタンをクリックします。ファイル名にフォルダまでのフルパスが入力され、確定します。

Windowsファイル指定

[d:][パス名指定]基本ファイル名[.拡張子]

d:はドライブ名

出力側のWindowsファイルを指定します。

ドライブ名はA:~Z:、@:、?:のどれかで指定します。ドライブ名を指定すると、そのドライブにファイルを作ります。

ドライブ名は省略可能です。ドライブ名を省略すると、カレントドライブにファイルを作ります。

パス名指定(¥ディレクトリ名¥サブディレクトリ名¥・・・)ができます。指定したディレクトリ配下にファイルを作ります。パス名指定を省略すると、カレントディレクトリにファイルを作ります。

基本ファイル名の部分には、ふつうの基本ファイル名、または*を指定します。*を指定するとホストファイルの基本ファイル名がWindowsファイルの基本ファイル名になります。

拡張子を省略すると、拡張子なしのファイルになります。しかし、拡張子をつけたほうがファイルの管理が容易になるので、なるべく適当な拡張子を指定してください。拡張子に*を指定するとホストファイルの拡張子がWindowsファイルの拡張子になります。

以上がディスクファイルの指定方法です。まとめると、基本ファイル名を引き継ぐときは

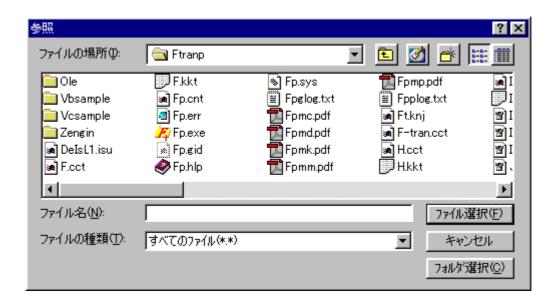
C:*.DAT のような指定になり、

ファイル名をつけ替えるときは

C:NEWNAME.DAT のような指定になります。

Windowsファイル参照ウインドウ

Windowsファイルの参照ボタンをクリックすると、つぎのウインドウが開きます。



変換先のファイルを直接指定する場合は、ファイル名を選択して、ファイル選択(<u>F</u>)ボタンをクリックします。

変換先の場所(どこのフォルダに変換後のファイルを作成するか)を指定する場合は、フォルダ選択(<u>C</u>)ボタンをクリックします。ファイル名にフォルダまでのフルパスが入力され、確定します。

その他の指定

变換結果表示指定

変換後に、変換先のWindowsファイルの中身を表示ウインドウに出力するかどうかを指定します。出力する場合は、変換結果表示のチェックボックスをONにします。

出力の形式は、変換方法によって異なりますので、それぞれの変換方法の節を参照してください。

問合わせ・確認指定

1ファイルごとに処理を問い合わせるか否かを指定します。

問合わせ・確認のチェックボックスを、

ONにすると、変換するか否かを問い合わせる指定

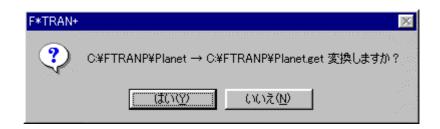
OFFにすると、ファイル名の確認なしで自動変換する指定 となります。

問合わせ・確認指定がONであれば、ファイル名を確認しながら変換できます。

F*TRAN+は、1ファイルごとに変換を実行するか問い合わせてきます。

つぎのどれかで応答してください。この機能は、比較的小さいファイルが多数あって、そのうちいくつかを選んで変換したいときなどに、便利です。

1ファイルの変換



はい(\underline{Y}) いいえ(N) 表示中のファイルを変換する 表示中のファイルは変換しない

2ファイル以上の変換



はい (\underline{Y}) 表示中のファイルを変換する

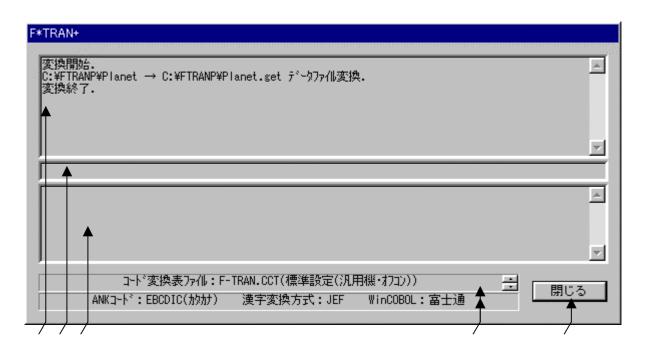
すべて変換 (\underline{A}) 全ファイル変換に切り替え、以降のファイルをすべて変換する

いいえ(<u>N</u>)キャンセル表示中のファイルは変換しないされ以降の変換処理を中断する

問合わせ・確認指定がOFFであれば、ファイル名の確認なしで、自動的に指定のファイルを すべて変換します。こちらがデフォルトです。

変換時の実行ウインドウ

変換ボタンをクリックすると、ホスト Winファイル変換が始まり、つぎの実行ウインドウが開きます。



変換中のメッセージを表示するメッセージフィールドです。

変換が始まると、"変換開始」"というメッセージ表示につづいて、

"ホストファイル名 Windowsファイル名 ~ファイル変換."が表示され、

変換が正常終了すると、"変換終了."と表示されます。

変換中の進行状況(進行度%出力)を表示するフィールドです。

変換中のエラーメッセージを表示するフィールドです。

変換中は「中止(Q)」と表示されています。

変換中にこのボタンをクリックすると、変換を途中で中止します。

変換が終了したら、その変換結果を確認して、"閉じる"ボタンをクリックします。

おもにコード変換に関する現在の設定が表示されています。

注意事項

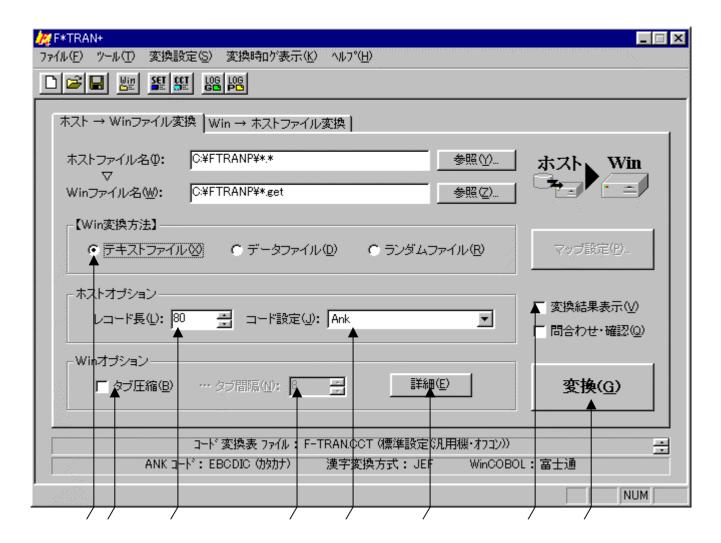
同名ファイルは置換する

すでに同じ名前のWindowsファイルがある場合、自動的に元のファイルを削除し、新たに変換したファイルで置き替えます。このとき警告メッセージは出ないので注意してください。

4.3 ホスト Winテキストファイル変換

ホスト Winテキストファイル変換の操作方法について説明します。

ホストが汎用機・オフコンの場合



変換方法のテキストファイル(X)をクリックし、

ホスト Winテキストファイル変換を選択します。

ホストファイルのレコード長を指定します。

ANK変換かAnk・漢字(KI/KO付)混在変換かを指定するオプションです。

タブ圧縮の有無を指定するオプションのチェックボックスです。

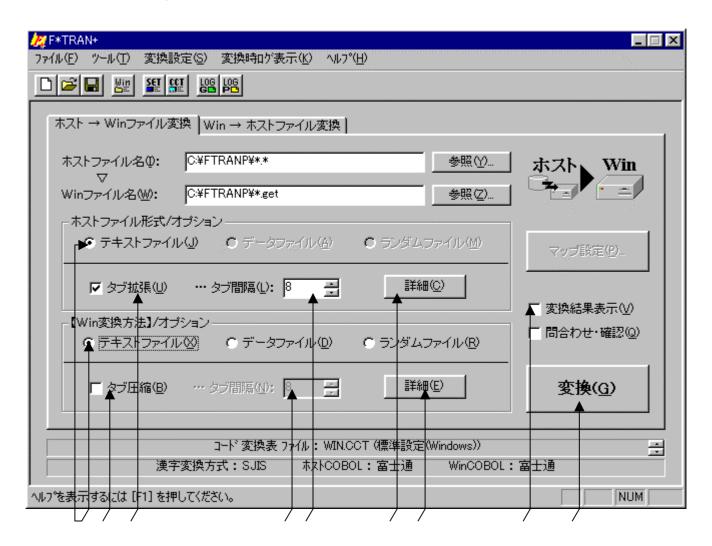
でタブ圧縮のチェックをONにしたとき、タブ間隔を指定できます。

Winファイルの詳細オプションボタンです。EOF付加の選択ができます。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホストがUnix、Windowsの場合



Win変換方法のテキストファイル (X) をクリックし、ホスト Winテキストファイル 変換を選択します。ホストファイル形式は、テキストファイル (J) 固定になります。 タブ拡張の有無を指定するオプションのチェックボックスです。

でタブ拡張のチェックをONにしたとき、タブ間隔を指定できます。

ホストファイルの詳細オプションボタンです。EOF検査の選択ができます。

ホストがWindowsの場合のみ有効です。ホストがUnixの場合はありません。 タブ圧縮の有無を指定するオプションのチェックボックスです。

でタブ圧縮のチェックをONにしたとき、タブ間隔を指定できます。

Winファイルの詳細オプションボタンです。EOF付加の選択ができます。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホスト Winファイル変換の共通指定項目は、「ホスト Winファイル変換」の節ですでに詳しく説明しました。そちらを参照してください。

オプションの指定

ホストファイル・レコード長指定 <ホストが汎用機・オフコンの場合のみ>

ホストファイルのレコード長を1~32767の範囲の10進数で指定します。デフォルトは、80バイトです。レコード長の指定が間違っていると正しいデータ変換が行われませんので、この指定は極めて重要です。

コード設定オプション 〈ホストが汎用機・オフコンの場合のみ〉

コード変換の方法(ANK変換かAnk・漢字(KI/KO付)混在変換か)を指定します。

Ank指定

Ank指定すると、すべてANKデータとして変換します。これがデフォルトです。以下に示す、

の3とおりの変換が可能です。あらかじめ、変換設定のANKコード設定でホストファイル側のコード系を設定しておかなければいけません(ふつう、セットアップ時に1回だけ行います)。 漢字がまじっているときは、つぎのAnk・漢字(KI/KO付)混在指定を使ってください。

Ank・漢字(KI/KO付)混在指定

ANK・漢字混在で、KI/KOもついているとき、この指定をします。あらかじめ、変換設定の漢字変換方式設定で適切な漢字変換方式の割り当てをする必要があります(ふつう、セットアップ時に1回だけ行います)。

タブ拡張のオプション <ホストがUnix、Windowsの場合のみ>

タブ拡張の有無と、タブ拡張するときのタブ間隔を指定します。タブ拡張とは、TAB(09H)をつぎのタブ位置の直前までの連続スペースに展開することです。

タブ拡張のチェックボックスを、

- ONにすると、タブ拡張する指定
- OFFにすると、タブ拡張しない指定 となります。

タブ拡張指定がONであれば、タブ拡張します。タブ間隔は、2~255の範囲で指定し、それがレコードのおわりまで繰り返し適用されます。タブ間隔のデフォルトは標準のタブ間隔(8桁きざみ)になります。これがデフォルトです。

タブ拡張指定がOFFであれば、タブ拡張はしません。

EOF検査指定 <ホストがWindowsの場合のみ>

EOFコード(1AH)を検査するか否かを指定します。

EOF検査のチェックボックスを、

- ONにすると、EOFコードを検査する指定
- OFFにすると、EOFコードを検査しない指定 となります。

EOF検査指定がONであれば、EOFコードを検査し、EOFコードが現れたら変換を終了します。こちらがデフォルトです。

EOF検査指定がOFFであれば、EOFコードを検査しません。単なるデータとして扱います。

タブ圧縮のオプション

タブ圧縮の有無と、タブ圧縮するときのタブ間隔を指定します。タブ圧縮とは、タブ位置の直前までつづく2個以上の連続スペースを、TAB(09H)に置き替えることです。

タブ圧縮のチェックボックスを、

ONにすると、タブ圧縮する指定

OFFにすると、タブ圧縮しない指定 となります。

タブ圧縮指定がONであれば、タブ圧縮します。タブ間隔は、2~255の範囲で指定し、それがレコードのおわりまで繰り返し適用されます。タブ間隔のデフォルトは標準のタブ間隔(8桁きざみ)になります。

タブ圧縮は、ソースプログラムなど空白部分が多いファイルの、変換後のファイル容量を減ら すのに効果的です。

なお、文字列定数中のスペースまでTABに変換してしまうことを避けるため、アポストロフィ(')か引用符(")が見つかると、その行についてはそこでタブ圧縮を打ち切ります。 タブ圧縮指定がOFFであれば、タブ圧縮はしません。これがデフォルトです。

EOF付加指定

EOFコード(1AH)の扱いを指定します。

EOF付加のチェックボックスを、

ONにすると、EOFコードをつける指定

OFFにすると、EOFコードをつけない指定 となります。

EOF付加指定がONであれば、WindowsファイルのおわりにEOFコードをつけます。 現在では少なくなりましたが、テキストファイルのおわりにEOFコードがついていないとエラーにするソフトがあります。その場合にも対処するための機能です。

EOF付加指定がOFFであれば、EOFコードはつけません。これがふつうだと思ってください。こちらがデフォルトです。

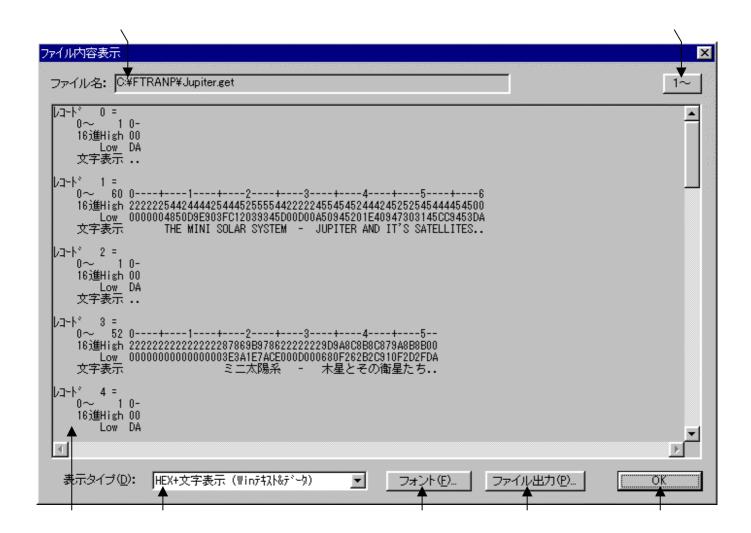
ホスト Winテキストファイル変換の実行

変換ボタンをクリックすると、ホスト Winテキストファイル変換が始まります。

行末の空白類の削除(行末圧縮)と改行コード(CR/LF=0D0AH)の付加は無条件に行われます。

ファイル内容表示ウインドウ

変換結果表示(<u>V</u>)のチェックボックスをONにし、ホスト Winテキストファイル変換を 実行すると、実行ウインドウを閉じた後につぎのファイル内容表示ウインドウが開き、変換先の ファイルの中身を確認することができます。



変換先のファイル名が表示されます。

変換先のファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、改行コード単位のHEX+文字表示(Winテキスト&データ)です。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えることができます。

フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

OKボタンをクリックすると、ファイル内容表示ウインドウを閉じます。

注意事項

<u>漢字があるときはコード設定オプションを A n k・漢字(KI/KO付)混在にすることを忘</u>れずに

漢字が入っているときは、あらかじめ

変換設定の漢字変換方式設定で、適当な漢字変換方式を割り当てておく

のを忘れないでください。また、ホストが汎用機・オフコンの場合、変換のとき、

コード設定オプションをAnk・漢字(KI/KO付)混在にするのを忘れがち

なので注意してください。

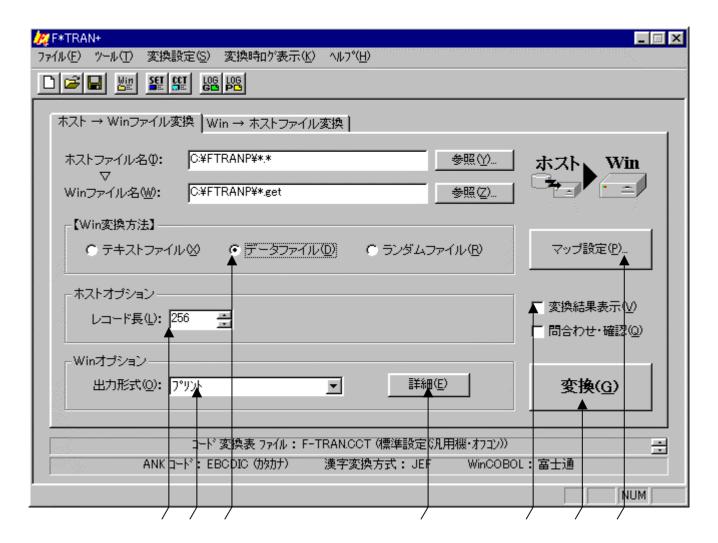
その他の注意事項

「ホスト Winファイル変換」の節を参照してください。

4.4 ホスト Winデータファイル変換

ホスト Winデータファイル変換の操作方法について説明します。

ホストが汎用機・オフコンの場合



変換方法のデータファイル(D)をクリックし、

ホスト Winデータファイル変換を選択します。

ホストファイルのレコード長を指定します。

出力ファイル形式を指定するオプションです。

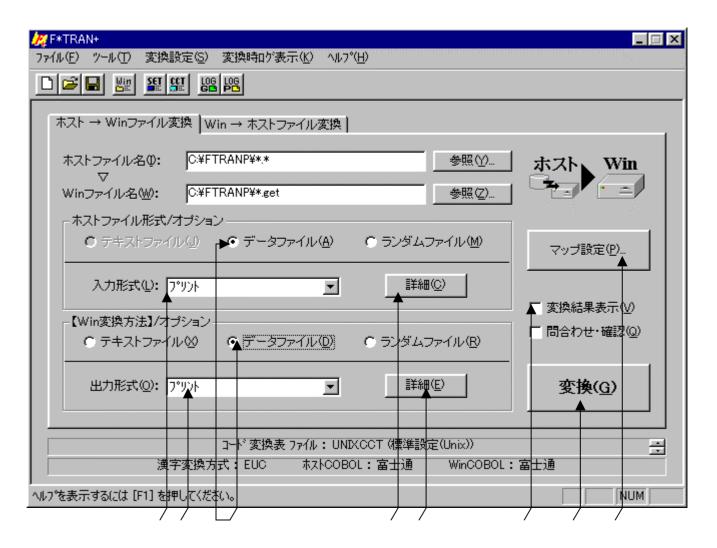
Winファイルの詳細オプションボタンです。EOF付加の選択ができます。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホストがUnix、Windowsの場合



Win変換方法のデータファイル (\underline{D})をクリックし、ホスト Winデータファイル変換を選択します。ホストファイル形式は、データファイル (\underline{A})、ランダムファイル (\underline{M})の選択ができますが、上の例はホストファイル形式がデータファイル (\underline{A}) です。

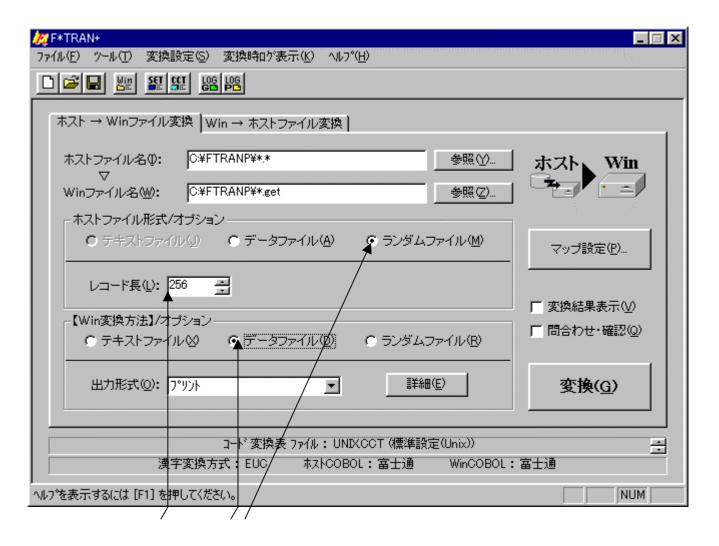
入力ファイル形式を指定するオプションです。

ホストファイルの詳細オプションボタンです。タブ拡張、空行無視等の指定ができます。 出力ファイル形式を指定するオプションです。

Winファイルの詳細オプションボタンです。EOF付加の選択ができます。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。 変換を実行するボタンです。このボタンをクリックすると、変換が始まります。



Win変換方法のデータファイル (\underline{D})をクリックし、ホスト Winデータファイル変換を選択します。ホストファイル形式は、データファイル (\underline{A})、ランダムファイル (\underline{M})の 選択ができますが、上の例はホストファイル形式がランダムファイル (\underline{M})です。ホストファイルのレコード長を指定します。

ホスト Winファイル変換の共通指定項目は、「ホスト Winファイル変換」の節ですでに詳しく説明しました。そちらを参照してください。

オプションの指定

ホストファイル・レコード長指定 <ホストファイルが固定長の場合のみ>

ホストファイルのレコード長を1~32767の範囲の10進数で指定します。デフォルトは、256バイトです。レコード長の指定が間違っていると正しいデータ変換が行われませんので、この指定は極めて重要です。

入力形式オプション < ホストファイルが可変長の場合のみ >

変換するWindowsファイルがプリント形式かデリミタ形式か、デリミタ形式ならコンマ 区切り・タブ区切り・スペース区切りのどれなのかを指定します。

プリント ------- プリント形式を変換する デリミタ(コンマ) ----- デリミタ形式(コンマ)を変換 デリミタ(タブ) ----- デリミタ形式(タブ)を変換 デリミタ(スペース) ---- デリミタ形式(スペース)を変換

プリントを指定すると、デリミタ(区切り文字)なしのプリント形式のファイル(固定長テキストファイル(SDF形式)。ただし、最終項目だけ可変長でもよい)を変換できます。

デリミタを指定すると、デリミタ形式(区切り文字つき)のテキストファイルを変換できます。 区切り文字の種類によって、さらに細かい形式が決まります。デリミタは、

コンマ区切り形式(CSV形式、K3形式) タブ区切り形式(TAB=09H) スペース区切り形式(SP=20H) の3つのなかから指定できます。

後述のマップ設定オプションで、デリミタ検出 = コンマ (,) を指定したところに、上記の区切り文字があるとみなされます。

EOF検査指定 <ホストがWindowsで、可変長ファイルの場合のみ>

- EOFコード(1AH)を検査するか否かを指定します。
- EOF検査のチェックボックスを、
 - ONにすると、EOFコードを検査する指定
 - OFFにすると、EOFコードを検査しない指定 となります。

EOF検査指定がONであれば、EOFコードを検査し、EOFコードが現れたら変換を終了します。こちらがデフォルトです。

EOF検査指定がOFFであれば、EOFコードを検査しません。単なるデータとして扱います。

タブ拡張のオプション 〈ホストファイルが可変長の場合のみ〉

タブ拡張の有無と、タブ拡張するときのタブ間隔を指定します。タブ拡張とは、TAB(09H)をつぎのタブ位置の直前までの連続スペースに展開することです。

タブ拡張のチェックボックスを、

- ONにすると、タブ拡張する指定
- OFFにすると、タブ拡張しない指定 となります。

タブ拡張指定がONであれば、タブ拡張します。タブ間隔は、2~255の範囲で指定し、それがレコードのおわりまで繰り返し適用されます。タブ間隔のデフォルトは標準のタブ間隔(8桁きざみ)になります。これがデフォルトです。

タブ拡張指定がOFFであれば、タブ拡張はしません。

空行無視のオプション 〈ホストファイルが可変長の場合のみ〉

空行を無視するかどうかを指定します。

- ONにすると、空行を無視する指定
- OFFにすると、空行を無視しない指定 となります。

出力形式オプション

データとしてのテキストファイルにはいくつもの形式があるので、どの形式にするかを指定します。 つぎの、 7 種類の中から選択します。

プリントを指定すると、デリミタ(区切り文字)なしで変換され、プリント形式(固定長のテキストファイル [SDF形式])になります。これがデフォルトです。

デリミタを指定すると、デリミタ形式(区切り文字つきのテキストファイル)に変換されます。 デリミタの種類によってさらに細かい形式が決まります。デリミタは、

コンマ区切り形式(CSV形式、K3形式) タブ区切り形式(TAB=09H) スペース区切り形式(SP=20H) の3つのなかから指定できます。

通常は、コンマ区切り(CSV)形式への変換が一般的です。さらに、

変換後に圧縮をかける(可変長になる。デフォルト) 変換後に圧縮しない(固定長のまま) を指定できます。

圧縮をかけると本来不要なスペース、つまり、

レコードの先頭・末尾のスペースデリミタの前後のスペース引用符の前のスペースが削除されます。

圧縮をかける機能があるのは、つぎのような理由があるためです。

変換後のWindowsファイルの容量を、大幅に減らすことができる 一部の市販ソフトでは、不要なスペースがあるとうまくデータを読み込めない

EOF付加指定

EOFコード(1AH)の扱いを指定します。

EOF付加のチェックボックスを、

ONにすると、EOFコードをつける指定 OFFにすると、EOFコードをつけない指定 となります。

EOF付加指定がONであれば、WindowsファイルのおわりにEOFコードをつけます。 現在では少なくなりましたが、テキストファイルのおわりにEOFコードがついていないとエラーにするソフトがあります。その場合にも対処するための機能です。

EOF付加指定がOFFであれば、EOFコードはつけません。これがふつうだと思ってください。こちらがデフォルトです。

マップ設定オプション

このマップ設定オプションで、項目別の細かい変換方法を指示します。本来なら、自動的に項目を認識して変換ができると便利です。しかし、ホストの、とくにCOBOLのデータには、データ自身に桁数や小数点位置の判断に必要な情報が含まれていない、という特性があります。そのため自動変換は原理的に不可能なのです。

マップ設定のボタンをクリックすると、つぎのウインドウが開きます。



マップ設定には、

簡易設定 ---- 設定ウインドウに従って、必要な値を入力するだけで設定ができます。

通常は、こちらの機能だけで間に合う場合がほとんどです。

詳細設定 ---- フリーフォーマットで、マップ設定を記述するエディタモードです。

簡易設定だけでは設定できない特殊な設定をする場合、あるいは、 既存のパラメータファイル等を流用したい場合等に使用します。

マップ設定の文の書き方を覚えなくては設定できません。

の2とおりの方法があります。

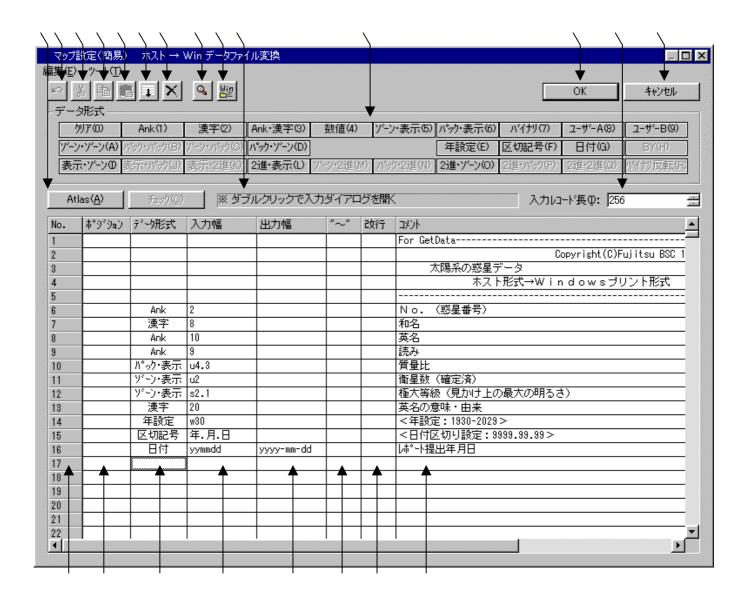
注意 ---- 簡易設定と詳細設定の組み合わせ

簡易設定で設定した値は、詳細設定で読み込むことができるテキスト形式に直されます。よって、簡易設定をした後の項目に、詳細設定で特殊設定をつけ足すこともできます。

注意 ---- 詳細設定 簡易設定の順で設定した場合の制約

詳細設定 簡易設定の順で設定した場合は、原形が変わってしまう場合があります。これは、 詳細設定で設定した項目が簡易設定で読み込まれることにより、スリム化されてしまうからです。 ただし、原形が変わってしまっても、設定内容が変わるわけではありません。

<マップ設定(簡易)ウインドウ>



No.は、1~2000です。ここは非入力項目で、行選択の時にクリックします。項目の桁位置等を入力します。通常、項目を組み替えるとき以外は入力しません。 ダブルクリックすると、入力サブウインドウが開きます。

項目のデータ形式を15種類の中から指定します。

ダブルクリックすると、入力サブウインドウが開きます。

項目の入力幅を入力します。省略すると、残りのバイト数を指定したことになります。

データ形式がゾーン / パック / 2 進指定の場合、ピクチャまたは 2 進ピクチャで指定します。 項目の出力幅を入力します。通常、出力幅 = 入力幅の場合は省略します。

デリミタ形式に変換するときに、項目を引用符でくくる指定ができます。ホストがUnix、Windowsの場合、デリミタ形式の項目の引用符をはずす指定にもなります。

改行コードを挿入する指定です。2種類(通常、強制)の指定があります。

改行コード挿入を指定すると、変換後の文字列の後に改行コードを挿入します。

コメントの入力ができます。

セルポインタの入力ガイダンスメッセージが出力されるフィールドです。

誤入力をした場合のエラーメッセージも出力されます。

~ は編集(E) ツール(T)のメニューの機能がボタン化されています。

直前の編集作業が無効になり、元に戻ります。

1行または複数行を選択(の操作)をした後に、このボタンをクリックすると、

選択した項目の内容が切り取られ、カットバッファに入ります。

1行または複数行を選択(の操作)をした後に、このボタンをクリックすると、

選択した項目の内容がカットバッファに入ります。

カットバッファに入った内容(、 の操作)を、セルポインタがある行へ貼りつけます。

すでに、セルポインタ以降の行に設定項目がある場合は、挿入になります。

セルポインタがある行以降が1行づつ下がります。通常は、行挿入をする場合に使用します。

選択した項目の内容が削除されます。

変換プレビュー(変換前/変換後のデータ確認)ができます。

Winファイルエディタが起動します。

ホストファイルの中身を見る(編集も可能)ことができます。

セルポインタが選択項目にある場合に、選択肢のボタンが表示されるエリアです。

ホストファイルが固定長の場合はレコード長、可変長の場合はデータ形式を指定できます。

変換プレビューの時の重要項目です。

設定が完了したら、OKボタンをクリックします。

キャンセルボタンをクリックすると、すべての入力 / 編集作業が無効になります。

マルチレコードの設定ができるAtlasモードに切り替えるボタンです。

詳細は、マルチレコード編のマニュアルを参照してください。

注意 ---- マップ未設定時のデータ変換動作

マップ未設定の場合は、すべてをANKデータとみなしてデータ変換を行います。

注意 ---- デリミタ挿入について

マップ設定(簡易)で項目を設定し、デリミタ形式に変換する場合、自動的に変換後の項目の後にデリミタ(区切り文字)が挿入されます(ただし、最終項目は除く)。挿入される区切り文字は指定された出力形式により、コンマ、タブ、スペースの3つのうちのいずれかになります。なお、プリント形式への変換のときは、デリミタ挿入は行われません。

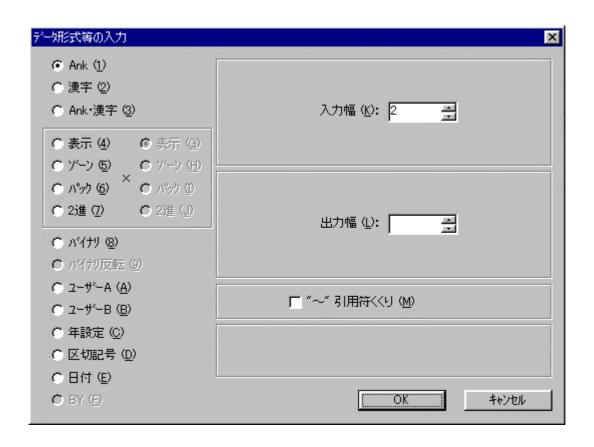
また、デリミタに設定したが、ある項目にはデリミタを付加したくない場合などは、簡易設定 後に詳細設定を行い、修正することができます。

<マップ設定(簡易)サブウインドウ>

ポジションの欄をダブルクリックすると、つぎのサブウインドウが開きます。入出力桁位置や 入出力スキップの設定を簡便に行うことができます。

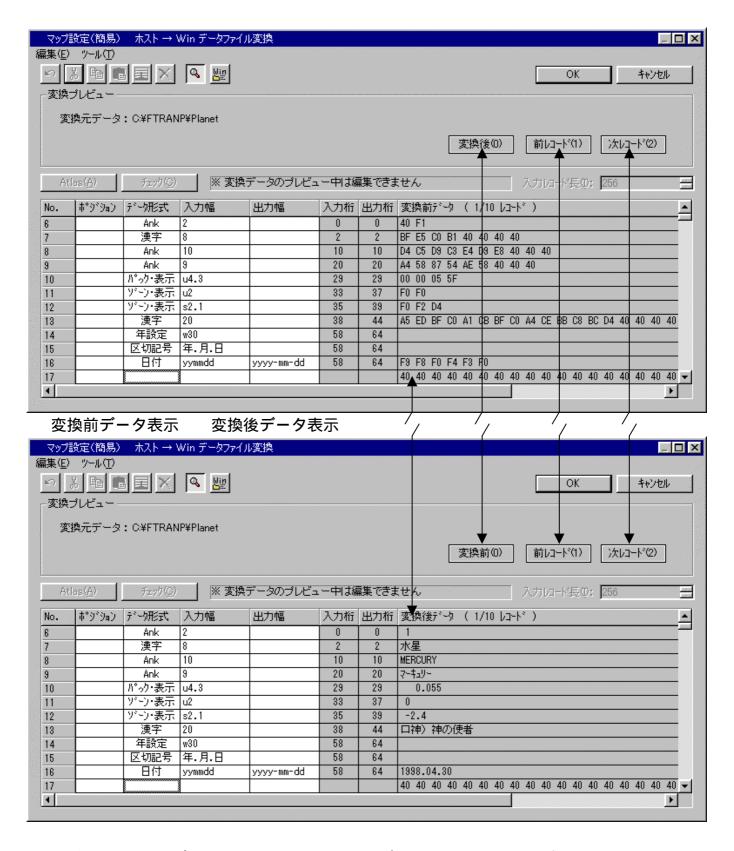


データ形式の欄をダブルクリックすると、つぎのサブウインドウが開きます。データ形式から 引用符くくりまでの設定を簡便に行うことができます。



<マップ設定(簡易)・変換プレビュー機能>

変換プレビューボタンをクリックすると、マップ設定(簡易)のウインドウはつぎのような状態になり、変換前/変換後のデータを確認することができます。



変換前 / 変換後のデータ表示の切り替えを行うボタンです。 の表示が変わります。 前レコード / 次レコード表示のためのボタンです。

ここには、入力レコード/出力レコードのポジション(桁)が表示されます。

データ表示は、変換前が16進、変換後が文字、未定義個所が16進になります。

(1)ポジション(桁位置)の指定

変換対象にするホスト側(入力)の桁位置や、変換結果を書き込むWindows側(出力)の桁位置を、別の任意の位置に移動できます。現在、処理対象にしている桁位置を、この指定で強制的に変更できます。この機能を利用すると、項目の組み替えなどが簡単に実現できます。

入力桁位置は、ふつう10進数で指定します。式による指定もでき、そのなかでは、

- * ホスト側(入力)のレコード長
- . ホスト側(入力)の現在の桁位置 という特殊変数が使えます。

たとえば、

- \$ 8 と指定すれば、ホスト側(入力)レコード末尾の8バイト前に位置する
- . 40 と指定すれば、 現在の入力桁位置から40バイト戻る

という意味になります。

プリント形式への変換の場合、出力桁位置を移動することもできます。ふつう、

- :10進数 で桁位置を指定します。
- 10進数の代わりに式による指定もでき、そのなかでは、
 - Windows側(出力)の現在の桁位置 という特殊変数が使えます。

注意 ---- デリミタ形式の場合、出力桁位置指定はない

デリミタ形式への変換の場合、入力桁位置の移動だけが有効です。出力桁位置の指定は行わないでください。

注意 ---- 先頭を0桁目とする

F*TRAN+では、レコードの先頭を0桁目として数えます。

(2)ポジション(デリミタ位置)の指定

変換対象にするホスト側(入力)データのデリミタ形式の項目位置を、別の任意の項目位置に 移動できます。現在、処理対象にしている項目位置を、この機能で強制的に変更できます。この 機能を利用すると、デリミタ形式の項目の組み替えなどが簡単に実現できます。

デリミタ位置は、 @@入力項目番号 で指定します。式による指定もでき、そのなかでは、

- . ホスト側(入力)の現在の項目位置
- \$ ホスト側(入力)の入力全体の項目数
- * ホスト側(入力)の残りの項目数 という特殊変数が使えます。

たとえば、

@@.-2 と指定すれば、 今の入力項目から2項目戻る

@@\$-2 と指定すれば、 ホスト側(入力)レコード末尾から2項目に位置する

という意味になります。

注意 ---- 先頭を 0 項目目とする

F*TRAN+では、レコードの先頭項目を0項目目として数えます。

(3)ポジション(入力スキップ)の指定

ホスト側(入力)レコードに不要な項目があるとき、それをスキップして変換できます。

スキップする幅は、 ^ バイト数 で指定します。

たとえば、3 バイト分スキップしたいなら、 ^3 と指定します。

バイト数は省略でき、省略すると1バイトとみされるので、

^3 は ^^ と指定したのと同じです。

スキップする幅は式による指定もでき、そのなかでは、

* ホスト側(入力)の残りバイト数 という特殊変数が使えます。

(4)ポジション(出力スキップ)の指定

入力スキップとは逆に、Windows側(出力)に何桁か空きを作ることもできます。プリント形式への変換のとき、空項目を作るのがおもな用途です。

スキップする幅は、 __バイト数 で指定します。

たとえば、3バイト分スキップしたいなら、 __3 と指定します。

バイト数は省略でき、省略すると1バイトとみされるので、

__3 は ____ と指定したのと同じです。

(5)15種類のデータ形式

[Ank(1)]

ANK項目を変換します。ホストが汎用機・オフコンの場合、どのコード変換が行われるかは、 変換設定のANKコードの設定で決まります(ふつう、セットアップ時に一回だけ行います)。 入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- \$ ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、ANK項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、Windows側のANK項目のおわりにスペース(20H)が詰められます。

[漢字(2)]

漢字項目を変換します。どのコード変換が行われるかは、変換設定の漢字変換方式の設定で決まります(ふつう、セットアップ時に一回だけ行います)。

入力幅は、10進のバイト数で指定します(漢字の文字数ではありません)。式による指定もでき、そのなかでは、

- \$ ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を大きくしなければならない場合があります。拡張漢字のクエスチョン変換を使っていて、オーバーフローの危険があるときなどです(最大で入力幅の3倍)。そのときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、漢字項目のおわりのほうが切り捨てられます(漢字の中央で切れることはありません)。逆に、項目長を伸ばすと、Windows側の漢字項目のおわりに漢字変換方式で設定されている漢字スペース(2020H/8140H)が詰められます。

[Ank・漢字(3)]

ANK・漢字まじり項目を変換します。どのコード変換が行われるかは、変換設定の漢字変換方式の設定で決まります(ふつう、セットアップ時に一回だけ行います)。ホストが汎用機・オフコンの場合、漢字の前後にKI/KOがついていないと、この変換方法は適用できません。変換後、KI/KOが取れて、その分、左詰めされます。

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- \$ ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を変更したい場合があります。KI/KOが取れて短くなることや、逆に、拡張漢字のクエスチョン変換を使っていてオーバーフローの危険があるときなどです(最大で入力幅の3倍-KI・KOのバイト数の合計)。そのときは、出力幅を指定します。

項目長を縮めると、ANK・漢字まじり項目のおわりのほうが切り捨てられます(漢字の中央で切れることはありません)。逆に、項目長を伸ばすと、Windows側のANK・漢字まじり項目のおわりにスペース(20H)が詰められます。

「数値(4)]

文字形式の数値項目どうしの変換をします。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコード設定でEBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。数値変換は、Ank項目変換と後述のゾーン・表示変換の中間的なものです。Ank項目変換と比較すると、

文字形式数値しか通さない 入力幅を省略すると15桁(バイト)とみなす 右詰めになる

などの点が異なります。

「文字形式数値しか通さない」というのは、具体的には、

+、-、0~9、ピリオド(.), E、e、D、d

しか変換しないで、これら以外の文字は捨ててしまうということです。たとえば、通貨記号 (¥/\$)や位取りのコンマ(,)などは削除されるので、リストファイルから入力データファイルを作るときなどに役立ちます。

[ゾーン・表示(5)]

ホストのCOBOLのゾーン形式数値項目を、文字形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコード設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)、変換結果は右詰めになります。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、 ピクチャは s 4 . 1 と指定します。なお、ピクチャは省略できません。

出力幅は省略できます。出力幅を省略すると、

符号つきなら1、符号なしなら0とする

- + 整数部桁数
- + 1 + 小数部桁数(小数部があれば)

の要領でピクチャから自動的に計算された値が使われます。たとえば、ピクチャ = s 4 . 1、 出力幅 = 省略という指定は、出力幅に7を指定したのと同じ意味になります。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

[パック・表示(6)]

ホストのCOBOLのパック形式数値項目、BCD形式数値項目を、文字形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコード設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)、変換結果は右詰めになります。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が3バイトの符号つきパック形式の項目に記録されているとすれば、ピクチャは s4.1 と指定します。123.4 という数字が3バイトのBCD形式の項目に記録されているとすれば、ピクチャは b5.1 と指定します。なお、ピクチャは省略できません。

パック形式では、整数部桁数 + 小数部桁数を奇数にしておくのが通例です。

BCD形式では、整数部桁数 + 小数部桁数を偶数にしておくのが通例です。

整数部の最上位桁に意味があるのかないのかは、半々の割合です。

出力幅は省略できます。出力幅を省略すると、

符号つきパック形式なら1、符号なしパック形式・BCD形式なら0とする

- + 整数部桁数
- + 1 + 小数部桁数(小数部があれば)

の要領でピクチャから自動的に計算された値が使われます。たとえば、パック形式でピクチャ = s 4 . 1、出力幅 = 省略という指定は、出力幅に7を指定したのと同じ意味になります。BCD形式でピクチャ = b 5 . 1、出力幅 = 省略という指定は、出力幅に7を指定したのと同じ意味になります。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。 オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

[バイナリ(7)]

バイナリ変換(無変換)は、「コード変換を一切しない」という変換方法です。通常、ホストWinデータファイル変換では使用しません。

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- * ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなされます。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、バイナリ項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、Windows側のバイナリ項目のおわりにスペース(20H)が詰められます。

$[1 - \forall A (8)], [1 - \forall B (9)]$

ユーザーA / B 変換は、利用者独自のバイト単位の変換処理が必要なときに、A N K 変換表ユーザー用A、ユーザー用 B を書き替えて利用します。ユーザーA / B 変換には、A n k 項目変換の説明がほとんどそのまま当てはまります。

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- \$ ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、ユーザーA / B項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、Windows側のユーザーA / B項目のおわりにスペース(20H)が詰められます。

[ゾーン・ゾーン(A)]

ホストのCOBOLのゾーン形式数値項目を、Windows COBOLのゾーン形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、 入力ピクチャは s 4 . 1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[パック・ゾーン(D)]

ホストのCOBOLのパック形式数値項目、BCD形式数値項目を、Windows COBOLのゾーン形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が3バイトの符号つきパック形式の項目に記録されているとすれば、 入力ピクチャは s4.1 と指定します。123.4 という数字が3バイトのBCD形式の 項目に記録されているとすれば、入力ピクチャは b5.1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[年設定(E)]

日付データ項目を変換する際の、年の 2 桁(y y) と 4 桁(y y y y) の交換方式を設定します。W n n、または、S n nの形式で設定します。W n n の " W " はウインドウ方式を、<math>S n nの " S " はシフト方式を意味し、" n n " は 0 0 ~ 9 9 の数字で指定します。

入力幅に W30 と指定すれば、入力データの年を1930~2029年とみなし、 出力幅に S25 と指定すれば、出力データの年下2桁を、-25します。

また、シフト方式("Snn"指定)では、つぎの特殊指定ができます。

S S h o w aは、" S 2 5 " の指定と同じ(昭和通年方式)S H e i s e iは、" S 8 8 " の指定と同じ(平成通年方式)

年設定は日付項目変換が実行された時に適用になり、複数の年設定がなされている場合は、日 付項目変換の直前の年設定が有効になります。

年設定がない場合の日付項目変換のデフォルトは、入力幅/出力幅ともに、 W30 となり、 入出力データの年を1930~2029年とみなします。

[区切記号(F)]

日付データ項目を出力する際の日付区切り記号をつぎの3つの中から設定します。入力幅に、SLASH、または、HYPHEN、または、PERIODと入力します。

指定文字	日付区切り記号	<u>データ例</u>
SLASH	/ (スラッシュ)	1998/12/31
HYPHEN	- (ハイフン)	1998-12-31
PERIOD	.(ピリオド)	1998.12.31

日付区切り設定は日付項目変換が実行された時に適用され、複数の日付区切り設定がなされている場合は、日付項目変換の直前の日付区切り設定が有効になります。

日付区切り設定がない場合の日付項目変換のデフォルトは、 SLASH となり、日付区切り記号を"/"にします。

[日付(G)]

日付データ項目を変換します。コード変換は、変換設定のANKコードの設定で決まります(ふつう、セットアップ時に一回だけ行います)。

入力幅と出力幅は、必ず日付マスクの形式で指定し、省略はできません。日付マスクの指定方法については、「操作の基礎」の章で説明しました。たとえば、

入力幅に、 yymmdd

出力幅に、 yyyy-mm-dd のように指定すると、

コード変換後に、入力側 6 バイトの日付データ項目を、出力側 1 0 バイトの日付データ項目に編集します。その際に、年設定、日付区切り設定が適用になります。

[表示・ゾーン(I)]

ホストの文字形式数値項目を、Windows COBOLのゾーン形式数値項目に変換します。通常は、ホストがUnixまたはWindowsの場合に使います。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。入力幅は、バイト数で指定します。省略すると15バイトとみなされるので、プリント形式からの変換の場合は、ふつうは明示的に桁数を指定します。デリミタ形式からの変換の場合は、数値項目が15バイトを超えることは少ないので、省略するほうがふつうです。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 . 4 という数字を 5 バイトの符号つきゾーン形式の項目に記録するとすれば、ピクチャは s 4 . 1 と指定します。なお、ピクチャは省略できません。

まとめると、プリント形式からの変換の場合は、

入力幅 = 8、ピクチャ = s 4 . 1 のような指定になり、

デリミタ形式からの変換の場合は、

入力幅=省略、ピクチャ=s4.1 のような指定になるのがふつうです。

ただし、入力幅が15バイトを超えるときは、

入力幅=20、ピクチャ=u15.2 のように、ダミーの入力幅を指定します。

[2進・表示(L)]

ホストの2進形式整数・小数項目を、Windowsの文字形式数値項目に変換します。変換結果は右詰めになります。

2 進キャスト / ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字が 4 バイトの符号つき 2 進形式の項目に記録されているとすれば、2 進キャスト / ピクチャは i 4 s 4.1 と指定します。なお、2 進キャストは省略できません。

出力幅は省略できます。出力幅を省略すると、

入力幅	出力幅	
1	3	
2	5	
3	8	
4	1 0	
5	1 3	左記の値を基本として、
6	1 5	符号つきなら + 1、ピクチャ指定の小数部があれば + 1とし、
7	1 7	さらに、ピクチャ指定のほうが大きければ、
8	1 8	整数部桁数に、符号つきなら+1、小数部があれば+1+小数点桁数

の要領で2進キャスト/ピクチャから自動的に計算された値が使われます。例を示すと、入力幅=i4s4.1、出力幅=省略という指定は、出力幅=10という指定と同じです。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

[2進・ゾーン(0)]

ホストの2進形式整数・小数項目を、Windows COBOLのゾーン形式数値項目に変換します。

2 進キャスト / ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字が 4 バイトの符号つき 2 進形式の項目に記録されているとすれば、2 進キャスト / ピクチャは i 4 s 4.1 と指定します。なお、2 進キャストは省略できません。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字を5バイトの項目に記録するとすれば、ピクチャは s4.1 と指定します。

出力ピクチャは省略できます。出力ピクチャを省略すると「2進キャスト/ピクチャと同じ」とみなされます。たとえば、2進キャスト/ピクチャ=i4s4.1、ピクチャ=省略という指定は、ピクチャ=s4.1という指定と同じです。

(6)引用符はずし/くくりの指定

ホストがUnix、Windowsの場合、デリミタ形式からの変換の場合には、項目が引用符(")でくくられていることがあります。その引用符は無視します。また、引用符でくくったなかにデリミタがあっても、ただの文字データとして扱います。

デリミタ形式に変換するときは、変換後の文字列を引用符でくくることができます。ただし、 プリント形式への変換のときは、この引用符くくりの機能は無効になります。

市販ソフトの入力用には、文字項目だけを引用符でくくることが多いのですが、引用符を使わないソフトもあれば、すべての項目を引用符でくくるソフトもあります。

参考 ...

たとえば、デリミタ形式のうちK3フォーマットと呼ばれるものは、

各項目はコンマで区切る 数値はそのまま(位取りのコンマ不可) 文字列は引用符でくくる

というルールになっています。

(7)改行コード挿入の指定

任意のところに改行コードを挿入する(改行コードで項目を区切る)こともできます。改行コード挿入には、つぎの2種類があります。

通常 プリント形式変換時は、指定が無効になる 強制 プリント形式変換時も、指定が有効になる

1 レコード変換がおわったときに、自動的にレコード末尾に改行コードが付加される機能とは 別のものですから、混同しないでください。

<マップ設定(詳細)ウインドウ>



マップ設定の記述をするエディットボックスです。

 \sim はファイル(\underline{F}) 編集(\underline{E})のメニューの機能がボタン化されています。 既存のパラメータファイル等を読みます。

の内容をパラメータファイルへ保存します。

直前の編集作業が無効になり、元に戻ります。

選択した文字列が切り取られ、カットバッファに入ります。

選択した文字列がカットバッファに入ります。

/ の操作でカットバッファに入った内容を、カーソルがある位置へ貼りつけます。 カーソルがある位置以降に文字列があれば、挿入になります。

選択した文字列が削除されます。

の文字列をすべて選択状態にします。

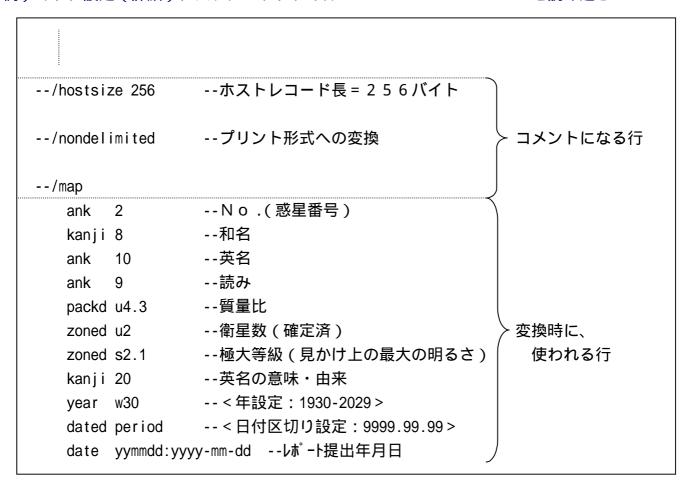
設定が完了したら、OKボタンをクリックします。

キャンセルボタンをクリックすると、すべての入力 / 編集作業が無効になります。

注意 ---- マップ設定(詳細)の記述内容について

マップ設定(詳細)のエディットボックスの中に記述する内容は、レコードレイアウト(/mapを含まない/map以降)のみの記述となります。既存のパラメータファイルを読み込むと、それ以外の記述(/delimited~、/map等)は自動的にコメントになります。記述に誤りがある状態でOKボタンをクリックすると、エラーとなる場合があります。修正してから再度OKボタンをクリックしてください。/map以降の文法については、「コマンド編」の「コマンド型の実行」の章を参照してください。

例)マップ設定(詳細)にパラメータファイル"PNGETPRN.P"を読み込む

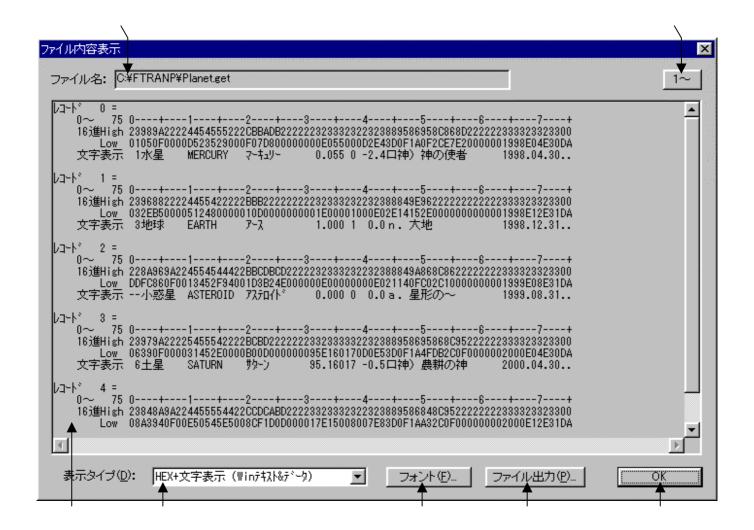


ホスト Winデータファイル変換の実行

変換ボタンをクリックすると、ホスト Winデータファイル変換が始まります。

ファイル内容表示ウインドウ

変換結果表示(<u>V</u>)のチェックボックスをONにし、ホスト Winデータファイル変換を実行すると、実行ウインドウを閉じた後につぎのファイル内容表示ウインドウが開き、変換先のファイルの中身を確認することができます。



変換先のファイル名が表示されます。

変換先のファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、改行コード単位のHEX+文字表示(Winテキスト&データ)です。 の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存します。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えます。

OKボタンをクリックして、ファイル内容表示ウインドウを閉じます。

注意事項

漢字があるときの注意

漢字項目変換やAnk・漢字まじり項目変換を行うときは、あらかじめ

変換設定の漢字変換方式設定で、適当な漢字変換方式を割り当てておく

のを忘れないでください。また、入力幅、出力幅は漢字データについても

バイト単位で指定

します。漢字の文字数ではないことに注意してください。

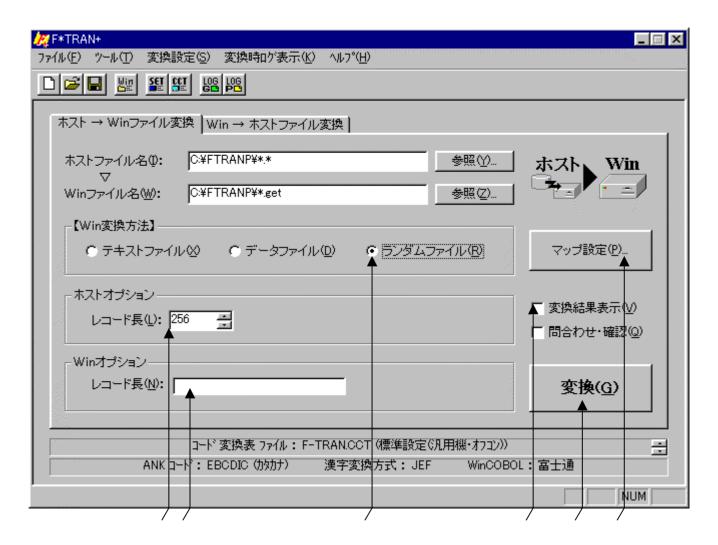
その他の注意事項

「ホスト Winファイル変換」の節を参照してください。

4.5 ホスト Winランダムファイル変換

ホスト Winランダムファイル変換の操作方法について説明します。

ホストが汎用機・オフコンの場合



変換方法のランダムファイル(R)をクリックし、

ホスト Winランダムファイル変換を選択します。

ホストファイルのレコード長を指定します。

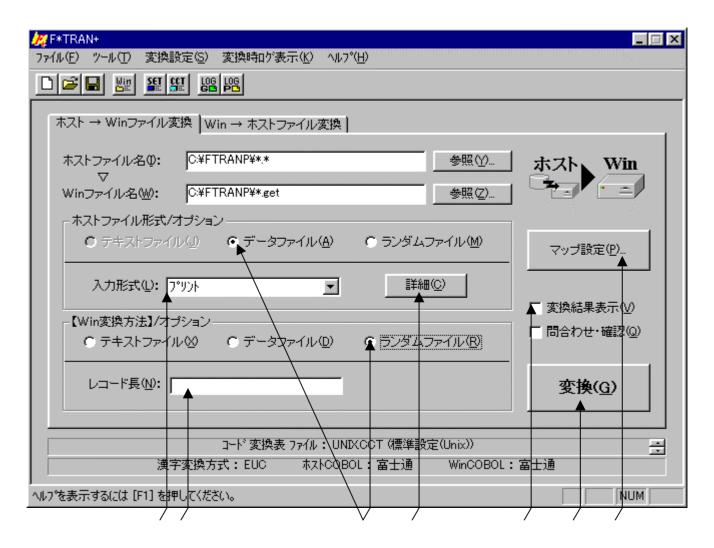
出力レコード長を指定するオプションです。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホストがUnix、Windowsの場合



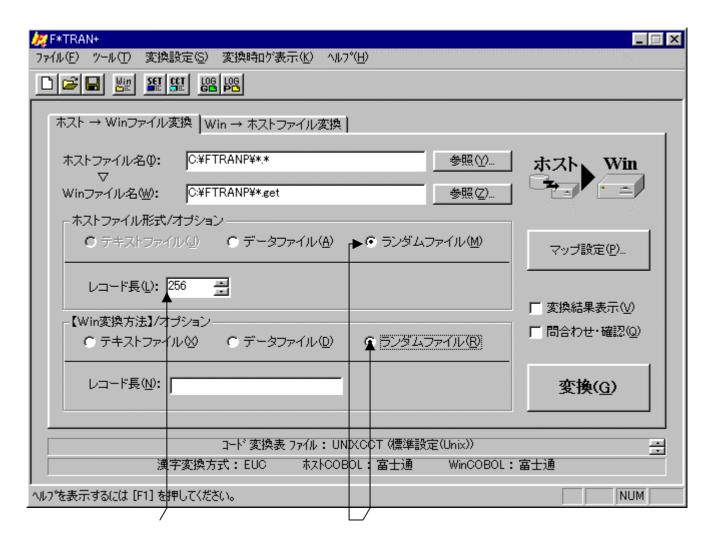
Win変換方法のランダムファイル (\underline{R})をクリックし、ホスト Winランダムファイル 変換を選択します。ホストファイル形式は、データファイル (\underline{A})、ランダムファイル (\underline{M}) の選択ができますが、上の例はホストファイル形式がデータファイル (\underline{A}) です。

入力ファイル形式を指定するオプションです。

ホストファイルの詳細オプションボタンです。タブ拡張、空行無視等の指定ができます。 出力レコード長を指定するオプションです。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。 変換を実行するボタンです。このボタンをクリックすると、変換が始まります。



Win変換方法のランダムファイル (\underline{D})をクリックし、ホスト Winランダムファイル 変換を選択します。ホストファイル形式は、データファイル (\underline{A})、ランダムファイル (\underline{M}) の選択ができますが、上の例はホストファイル形式がランダムファイル (\underline{M}) です。ホストファイルのレコード長を指定します。

ホスト Winファイル変換の共通指定項目は、「ホスト Winファイル変換」の節ですでに詳しく説明しました。そちらを参照してください。

オプションの指定

ホストファイル・レコード長指定 <ホストファイルが固定長の場合のみ>

ホストファイルのレコード長を1~32767の範囲の10進数で指定します。デフォルトは、256バイトです。レコード長の指定が間違っていると正しいデータ変換が行われませんので、この指定は極めて重要です。

入力形式オプション < ホストファイルが可変長の場合のみ >

変換するWindowsファイルがプリント形式かデリミタ形式か、デリミタ形式ならコンマ 区切り・タブ区切り・スペース区切りのどれなのかを指定します。

プリント ------- プリント形式を変換する デリミタ(コンマ) ----- デリミタ形式(コンマ)を変換 デリミタ(タブ) ----- デリミタ形式(タブ)を変換 デリミタ(スペース) ---- デリミタ形式(スペース)を変換

プリントを指定すると、デリミタ(区切り文字)なしのプリント形式のファイル(固定長テキストファイル(SDF形式)。ただし、最終項目だけ可変長でもよい)を変換できます。

デリミタを指定すると、デリミタ形式(区切り文字つき)のテキストファイルを変換できます。 区切り文字の種類によって、さらに細かい形式が決まります。デリミタは、

コンマ区切り形式(CSV形式、K3形式) タブ区切り形式(TAB=09H) スペース区切り形式(SP=20H) の3つのなかから指定できます。

後述のマップ設定オプションで、デリミタ検出 = コンマ (,) を指定したところに、上記の区切り文字があるとみなされます。

EOF検査指定 〈ホストがWindowsで、可変長ファイルの場合のみ〉

- EOFコード(1AH)を検査するか否かを指定します。
- EOF検査のチェックボックスを、
 - ONにすると、EOFコードを検査する指定
 - OFFにすると、EOFコードを検査しない指定 となります。

EOF検査指定がONであれば、EOFコードを検査し、EOFコードが現れたら変換を終了します。こちらがデフォルトです。

EOF検査指定がOFFであれば、EOFコードを検査しません。単なるデータとして扱います。

タブ拡張のオプション 〈ホストファイルが可変長の場合のみ〉

タブ拡張の有無と、タブ拡張するときのタブ間隔を指定します。タブ拡張とは、TAB(09H)をつぎのタブ位置の直前までの連続スペースに展開することです。

タブ拡張のチェックボックスを、

- ONにすると、タブ拡張する指定
- OFFにすると、タブ拡張しない指定 となります。

タブ拡張指定がONであれば、タブ拡張します。タブ間隔は、2~255の範囲で指定し、それがレコードのおわりまで繰り返し適用されます。タブ間隔のデフォルトは標準のタブ間隔(8桁きざみ)になります。これがデフォルトです。

タブ拡張指定がOFFであれば、タブ拡張はしません。

空行無視のオプション 〈ホストファイルが可変長の場合のみ〉

空行を無視するかどうかを指定します。

- ONにすると、空行を無視する指定
- OFFにすると、空行を無視しない指定 となります。

出力レコード長オプション

Windows側(出力)のレコード長を指定します。Windowsレコード長は、通常、10進数で指定します。式による指定もでき、そのなかでは、

\$ ホスト側(入力)のレコード長 という特殊変数が使用できます。

ホストファイルのレコード長を基準にして、Windowsファイルのレコード長を決めることができます。 たとえば、

\$ + 2 と指定すれば、

「元のホストファイルのレコード長に + 2 したものをWindows側のレコード長にせよ」 という意味になります。

出力レコード長オプションでWindowsレコード長を指定できることは、ホスト Winランダムファイル変換の性格をよく反映しています。

ホスト Winランダムファイル変換は、おもにWindows COBOLの順ファイルやBASICのランダムファイルに変換することをねらったものである、ということは冒頭で述べたとおりです。そして、BASIC側の事情を考慮すると、レコード長を任意に指定できないと困ることが多いのです。

とくに困るのは、N88BASICコンパイラ/インタプリタを使うときです。

N88BASICコンパイラ / インタプリタでは、1つのプログラム内で複数のレコード長のランダムファイルを扱うことができません。

もう1つの理由は、ゾーン形式とパック形式の変換、日付データの変換をサポートしているため、レコード長を増減する機能がないと困るからです。また、不要な項目がいくつもあるときや、 予備領域が多すぎるときはレコード長を縮めたいものです。

注意 ---- Windows自身には「レコード長の概念がない」

Windows自体にはファイルごとに登録されたレコード長というものはありません。そのため、かなり自由が利きます。アプリケーションがファイルを扱う論理的な単位をここではWindowsレコード長と呼んでいます。

マップ設定オプション

このマップ設定オプションで、項目別の細かい変換方法を指示します。本来なら、自動的に項目を認識して変換ができると便利です。しかし、ホストの、とくにCOBOLのデータには、データ自身に桁数や小数点位置の判断に必要な情報が含まれていない、という特性があります。そのため自動変換は原理的に不可能なのです。

マップ設定のボタンをクリックすると、つぎのウインドウが開きます。



マップ設定には、

簡易設定 ---- 設定ウインドウに従って、必要な値を入力するだけで設定ができます。

通常は、こちらの機能だけで間に合う場合がほとんどです。

詳細設定 ---- フリーフォーマットで、マップ設定を記述するエディタモードです。

簡易設定だけでは設定できない特殊な設定をする場合、あるいは、 既存のパラメータファイル等を流用したい場合等に使用します。

マップ設定の文の書き方を覚えなくては設定できません。

の2とおりの方法があります。

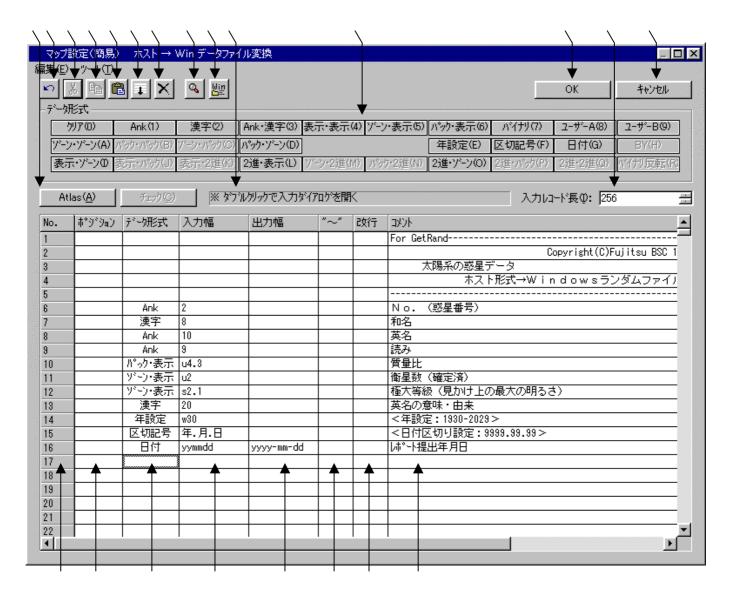
注意 ---- 簡易設定と詳細設定の組み合わせ

簡易設定で設定した値は、詳細設定で読み込むことができるテキスト形式に直されます。よって、簡易設定をした後の項目に、詳細設定で特殊設定をつけ足すこともできます。

注意 ---- 詳細設定 簡易設定の順で設定した場合の制約

詳細設定 簡易設定の順で設定した場合は、原形が変わってしまう場合があります。これは、 詳細設定で設定した項目が簡易設定で読み込まれることにより、スリム化されてしまうからです。 ただし、原形が変わってしまっても、設定内容が変わるわけではありません。

<マップ設定(簡易)ウインドウ>



No.は、1~2000です。ここは非入力項目で、行選択の時にクリックします。項目の桁位置等を入力します。通常、項目を組み替えるとき以外は入力しません。 ダブルクリックすると、入力サブウインドウが開きます。

項目のデータ形式を24種類の中から指定します。

ダブルクリックすると、入力サブウインドウが開きます。

項目の入力幅を入力します。省略すると、残りのバイト数を指定したことになります。

データ形式がゾーン / パック / 2 進指定の場合、ピクチャまたは 2 進ピクチャで指定します。

項目の出力幅を入力します。出力幅=入力幅の場合は、省略するのがふつうです。

ホストがUnix、Windowsの場合、デリミタ形式から変換するときに引用符はずしの指定ができます。

改行コードを挿入する指定です。

改行コード挿入を指定すると、変換後の文字列の後に改行コードを挿入します。通常、

項目長の増減がない単純な変換の最終項目に指定し、テキストファイル化するのに使います。

コメントの入力ができます。

セルポインタの入力ガイダンスメッセージが出力されるフィールドです。 誤入力をした場合のエラーメッセージも出力されます。

 \sim は編集(\underline{E})、ツール(\underline{T})のメニューの機能がボタン化されています。 直前の編集作業が無効になり、元に戻ります。

1 行または複数行を選択(の操作)をした後に、このボタンをクリックすると、 選択した項目の内容が切り取られ、カットバッファに入ります。

1 行または複数行を選択(の操作)をした後に、このボタンをクリックすると、 選択した項目の内容がカットバッファに入ります。

カットバッファに入った内容(、 の操作)を、セルポインタがある行へ貼りつけます。 すでに、セルポインタ以降の行に設定項目がある場合は、挿入になります。

セルポインタがある行以降が1行づつ下がります。通常は、行挿入をする場合に使用します。 選択した項目の内容が削除されます。

変換プレビュー(変換前/変換後のデータ確認)ができます。

Winファイルエディタが起動します。

ホストファイルの中身を見る(編集も可能)ことができます。

セルポインタが選択項目にある場合に、選択肢のボタンが表示されるエリアです。 ホストファイルのレコード長を指定できます。変換プレビューの時の重要項目です。 設定が完了したら、OKボタンをクリックします。

キャンセルボタンをクリックすると、すべての入力 / 編集作業が無効になります。

マルチレコードの設定ができるAtlasモードに切り替えるボタンです。 詳細は、マルチレコード編のマニュアルを参照してください。

注意 ---- マップ未設定時のデータ変換動作

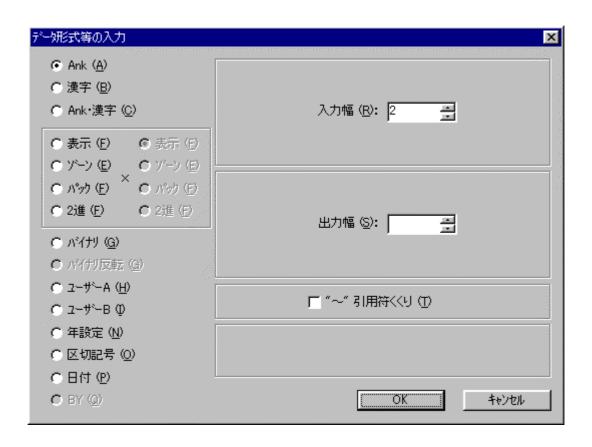
マップ未設定の場合は、すべてをANKデータとみなしてデータ変換を行います。

<マップ設定(簡易)サブウインドウ>

ポジションの欄をダブルクリックすると、つぎのサブウインドウが開きます。入出力桁位置や 入出力スキップの設定を簡便に行うことができます。

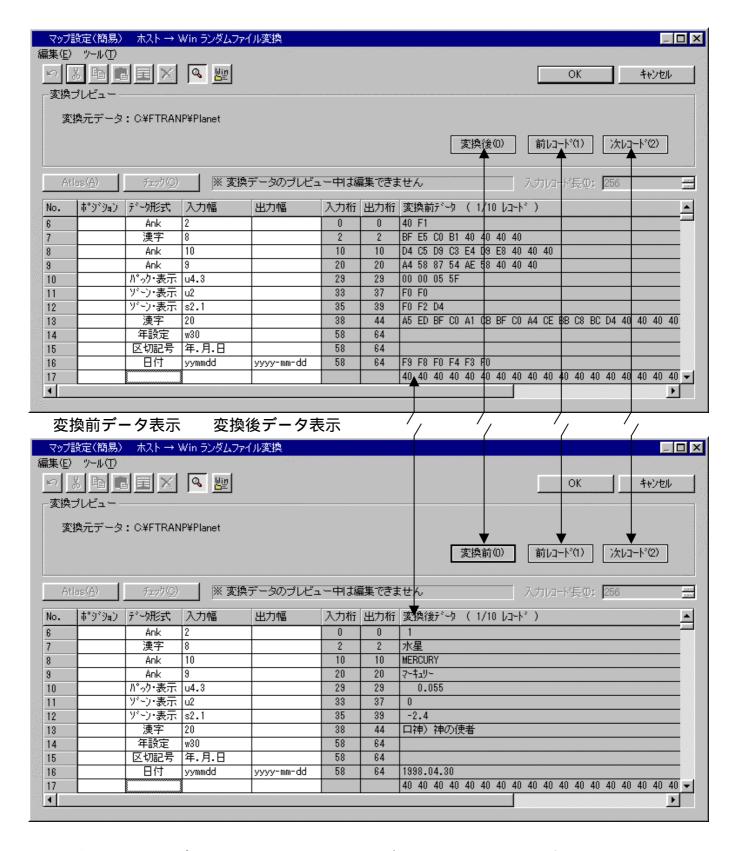


データ形式の欄をダブルクリックすると、つぎのサブウインドウが開きます。データ形式から 引用符くくりまでの設定を簡便に行うことができます。



<マップ設定(簡易)・変換プレビュー機能>

変換プレビューボタンをクリックすると、マップ設定(簡易)のウインドウはつぎのような状態になり、変換前/変換後のデータを確認することができます。



変換前 / 変換後のデータ表示の切り替えを行うボタンです。 の表示が変わります。 前レコード / 次レコード表示のためのボタンです。

ここには、入力レコード/出力レコードのポジション(桁)が表示されます。

データ表示は、変換前が16進、変換後が文字、未定義個所が16進になります。

(1)ポジション(桁位置)の指定

変換対象にするホスト側(入力)の桁位置や、変換結果を書き込むWindows側(出力)の桁位置を、別の任意の位置に移動できます。現在、処理対象にしている桁位置を、この指定で強制的に変更できます。この機能を利用すると、項目の組み替えなどが簡単に実現できます。

入力桁位置は、ふつう10進数で指定します。式による指定もでき、そのなかでは、

- \$ ホスト側(入力)のレコード長
- . ホスト側(入力)の現在の桁位置 という特殊変数が使えます。

たとえば、

- \$ 8 と指定すれば、ホスト側(入力)レコード末尾の8バイト前に位置する
- . 40 と指定すれば、 現在の入力桁位置から40バイト戻る

という意味になります。

出力桁位置を移動することもできます。ふつう、

- :10進数 で桁位置を指定します。
- 10進数の代わりに式による指定もでき、そのなかでは、
 - \$ Windows側(出力)のレコード長
 - . Windows側(出力)の現在の桁位置 という特殊変数が使えます。

注意 ---- 先頭を0桁目とする

F*TRAN+では、レコードの先頭を0桁目として数えます。

(2)ポジション(デリミタ位置)の指定

変換対象にするホスト側(入力)データのデリミタ形式の項目位置を、別の任意の項目位置に 移動できます。現在、処理対象にしている項目位置を、この機能で強制的に変更できます。この 機能を利用すると、デリミタ形式の項目の組み替えなどが簡単に実現できます。

デリミタ位置は、 @@入力項目番号 で指定します。式による指定もでき、そのなかでは、

- . ホスト側(入力)の現在の項目位置
- \$ ホスト側(入力)の入力全体の項目数
- * ホスト側(入力)の残りの項目数 という特殊変数が使えます。

たとえば、

@@.-2 と指定すれば、 今の入力項目から2項目戻る

@@\$-2 と指定すれば、 ホスト側(入力)レコード末尾から2項目に位置する

という意味になります。

注意 ---- 先頭を 0 項目目とする

F*TRAN+では、レコードの先頭項目を0項目目として数えます。

(3)ポジション(入力スキップ)の指定

ホスト側(入力)レコードに不要な項目があるとき、それをスキップして変換できます。

スキップする幅は、 ^ バイト数 で指定します。

たとえば、3 バイト分スキップしたいなら、 ^3 と指定します。

バイト数は省略でき、省略すると1バイトとみされるので、

^3 は ^^ と指定したのと同じです。

スキップする幅は式による指定もでき、そのなかでは、

* ホスト側(入力)の残りバイト数 という特殊変数が使えます。

(4)ポジション(出力スキップ)の指定

入力スキップとは逆に、Windows側(出力)に何桁か空きを作ることもできます。空項目を作るのがおもな用途です。

スキップする幅は、 __バイト数 で指定します。

たとえば、3バイト分スキップしたいなら、 __3 と指定します。

バイト数は省略でき、省略すると1バイトとみされるので、

__3 は ____ と指定したのと同じです。

スキップする幅は式による指定もでき、そのなかでは、

* Windows側(出力)の残りバイト数 という特殊変数が使えます。

(5)24種類のデータ形式

[Ank(1)]

ANK項目を変換します。ホストが汎用機・オフコンの場合、どのコード変換が行われるかは、 変換設定のANKコードの設定で決まります(ふつう、セットアップ時に一回だけ行います)。 入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- \$ ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、ANK項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、Windows側のANK項目のおわりにスペース(20H)が詰められます。

出力幅は式による指定もでき、そのなかでは、

- \$ Windows側(出力)のレコード長
- * Windows側(出力)の残りバイト数 という特殊変数が使えます。

[漢字(2)]

漢字項目を変換します。どのコード変換が行われるかは、変換設定の漢字変換方式の設定で決まります(ふつう、セットアップ時に一回だけ行います)。

入力幅は、10進のバイト数で指定します(漢字の文字数ではありません)。式による指定もでき、そのなかでは、

- \$ ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を大きくしなければならない場合があります。拡張漢字のクエスチョン変換を使っていて、オーバーフローの危険があるときなどです(最大で入力幅の3倍)。そのときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、漢字項目のおわりのほうが切り捨てられます(漢字の中央で切れることはありません)。逆に、項目長を伸ばすと、Windows側の漢字項目のおわりに漢字変換方式で設定されている漢字スペース(2020H/8140H)が詰められます。

出力幅は式による指定もでき、そのなかでは、

- \$ Windows側(出力)のレコード長
- * Windows側(出力)の残りバイト数 という特殊変数が使えます。

[Ank・漢字(3)]

ANK・漢字まじり項目を変換します。どのコード変換が行われるかは、変換設定の漢字変換方式の設定で決まります(ふつう、セットアップ時に一回だけ行います)。ホストが汎用機・オフコンの場合、漢字の前後にKI/KOがついていないと、この変換方法は適用できません。変換後、KI/KOが取れて、その分左詰めにされます。

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- * ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を変更したい場合があります。KI/KOが取れて短くなることや、逆に、拡張漢字のクエスチョン変換を使っていてオーバーフローの危険があるときなどです(最大で入力幅の3倍-KI・KOのバイト数の合計)。そのときは、出力幅を指定します。

項目長を縮めると、ANK・漢字まじり項目のおわりのほうが切り捨てられます(漢字の中央で切れることはありません)。逆に、項目長を伸ばすと、Windows側のANK・漢字まじり項目のおわりにスペース(20H)が詰められます。

出力幅は式による指定もでき、そのなかでは、

- \$ Windows側(出力)のレコード長
- * Windows側(出力)の残りバイト数 という特殊変数が使えます。

[数値(4)]

文字形式の数値項目どうしの変換をします。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。数値変換は、Ank項目変換と後述のゾーン・表示変換の中間的なものです。Ank項目変換と比較すると、

文字形式数値しか通さない 入力幅を省略すると15桁(バイト)とみなす 右詰めになる

などの点が異なります。

「文字形式数値しか通さない」というのは、具体的には、

+、-、0~9、ピリオド(.), E、e、D、d

しか変換しないで、これら以外の文字は捨ててしまうということです。たとえば、通貨記号(¥/\$)や位取りのコンマ(,)などは削除されるので、リストファイルから入力データファイルを作るときなどに利用できます。

[ゾーン・表示(5)]

ホストのCOBOLのゾーン形式数値項目を、文字形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。変換結果は右詰めになります。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が5バイトの符号つきゾーン形式の項目に記録されているとすれば、 ピクチャは s4.1 と指定します。なお、ピクチャは省略できません。

出力幅は省略できます。出力幅を省略すると、

符号つきなら1、符号なしなら0とする

- + 整数部桁数
- + 1 + 小数部桁数(小数部があれば)

の要領でピクチャから自動的に計算された値が使われます。たとえば、ピクチャ = s 4 . 1、 出力幅 = 省略という指定は、出力幅に7を指定したのと同じ意味になります。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。 オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

[パック・表示(6)]

ホストのCOBOLのパック形式数値項目、BCD形式数値項目を、文字形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。変換結果は右詰めになります。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が3バイトの符号つきパック形式の項目に記録されているとすれば、ピクチャは s4.1 と指定します。123.4 という数字が3バイトのBCD形式の項目に記録されているとすれば、ピクチャは b5.1 と指定します。なお、ピクチャは省略できません。

パック形式では、整数部桁数 + 小数部桁数を奇数にしておくのが通例です。

BCD形式では、整数部桁数 + 小数部桁数を偶数にしておくのが通例です。

整数部の最上位桁に意味があるのかないのかは、半々の割合です。

出力幅は省略できます。出力幅を省略すると、

符号つきパック形式なら1、符号なしパック形式・BCD形式なら0とする

- + 整数部桁数
- + 1 + 小数部桁数(小数部があれば)

の要領でピクチャから自動的に計算された値が使われます。たとえば、パック形式でピクチャ = s 4 . 1、出力幅 = 省略という指定は、出力幅に7を指定したのと同じ意味になります。BCD形式でピクチャ = b 5 . 1、出力幅 = 省略という指定は、出力幅に7を指定したのと同じ意味になります。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

[バイナリ(7)]

バイナリ変換 (無変換) は、「コード変換を一切しない」という変換方法です。バイナリ変換には、Ank項目変換の説明がほぼそのまま当てはまります。

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- * ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなされます。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、バイナリ項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、Windows側のバイナリ項目のおわりにNUL(00H)が詰められます。

出力幅は式による指定もでき、そのなかでは、

- \$ Windows側(出力)のレコード長
- * Windows側(出力)の残りバイト数 という特殊変数が使えます。

Ank項目変換の説明が当てはまると書きましたが、実際の運用はかなり違ったものになります。とくに、レコード全体をバイナリ変換することが多いことです。このとき、レコード長が変わらないように出力レコード長は\$を指定します。また、レコードの一部をバイナリ項目として扱うこともあります。

ユーザーA / B変換は、利用者独自のバイト単位の変換処理が必要なときに、A N K変換表ユーザー用A、ユーザー用Bを書き替えて利用します。ユーザーA / B変換には、A n k項目変換の説明がほとんどそのまま当てはまります。

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- * ホスト側(入力)のレコード長
- * ホスト側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、ユーザーA / B項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、Windows側のユーザーA / B項目のおわりにNUL(00H)が詰められます。 出力幅は式による指定もでき、そのなかでは、

- \$ Windows側(出力)のレコード長
- * Windows側(出力)の残りバイト数 という特殊変数が使えます。

[ゾーン・ゾーン(A)]

ホストのCOBOLのゾーン形式数値項目を、Windows COBOLのゾーン形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、 入力ピクチャは s 4 . 1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

$[\mathcal{N} \cup \mathcal{N}$

ホストのCOBOLのパック形式数値項目、BCD形式数値項目を、Windows COBOLのパック形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が3バイトの符号つきパック形式の項目に記録されているとすれば、 入力ピクチャは s4.1 と指定します。123.4 という数字が3バイトのBCD形式の 項目に記録されているとすれば、入力ピクチャは b5.1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ=s4.1という指定は、入力ピクチャ=s4.1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[ゾーン・パック(C)]

ホストのCOBOLのゾーン形式数値項目を、Windows COBOLのパック形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、 入力ピクチャは s 4 . 1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[パック・ゾーン(D)]

ホストのCOBOLのパック形式数値項目、BCD形式数値項目を、Windows COBOLのパック形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が3バイトの符号つきパック形式の項目に記録されているとすれば、 入力ピクチャは s4.1 と指定します。123.4 という数字が3バイトのBCD形式の 項目に記録されているとすれば、入力ピクチャは b5.1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[年設定(E)]

日付データ項目を変換する際の、年の 2 桁(y y) と 4 桁(y y y y) の交換方式を設定します。W n n、または、S n nの形式で設定します。W n n の " W " はウインドウ方式を、<math>S n nの " S " はシフト方式を意味し、" n n " は 0 0 ~ 9 9 の数字で指定します。

入力幅に W30 と指定すれば、入力データの年を1930~2029年とみなし、 出力幅に S25 と指定すれば、出力データの年下2桁を、-25します。

また、シフト方式("Snn"指定)では、つぎの特殊指定ができます。

S S h o w a は、" S 2 5 " の指定と同じ(昭和通年方式) S H e i s e i は、" S 8 8 " の指定と同じ(平成通年方式)

年設定は日付項目変換が実行された時に適用になり、複数の年設定がなされている場合は、日 付項目変換の直前の年設定が有効になります。

年設定がない場合の日付項目変換のデフォルトは、入力幅/出力幅ともに、 W30 となり、 入出力データの年を1930~2029年とみなします。

[区切記号(F)]

日付データ項目を出力する際の日付区切り記号をつぎの3つの中から設定します。入力幅に、SLASH、または、HYPHEN、または、PERIODと入力します。

指定文字	日付区切り記号	_ デ ー タ 例
SLASH	/ (スラッシュ)	1998/12/31
HYPHEN	- (ハイフン)	1998-12-31
PERIOD	.(ピリオド)	1998.12.31

日付区切り設定は日付項目変換が実行された時に適用され、複数の日付区切り設定がなされている場合は、日付項目変換の直前の日付区切り設定が有効になります。

日付区切り設定がない場合の日付項目変換のデフォルトは、 SLASH となり、日付区切り記号を"/"にします。

[日付(G)]

日付データ項目を変換します。コード変換は、変換設定のANKコードの設定で決まります(ふつう、セットアップ時に一回だけ行います)。

入力幅と出力幅は、必ず日付マスクの形式で指定し、省略はできません。日付マスクの指定方法については、「操作の基礎」の章で説明しました。たとえば、

入力幅に、 yymmdd

出力幅に、 yyyy-mm-dd のように指定すると、

コード変換後に、入力側 6 バイトの日付データ項目を、出力側 1 0 バイトの日付データ項目に編集します。その際に、年設定、日付区切り設定が適用になります。

[表示・ゾーン(I)]

ホストの文字形式数値項目を、Windows COBOLのゾーン形式数値項目に変換します。通常は、ホストがUnixまたはWindowsの場合に使います。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。入力幅は、バイト数で指定します。省略すると15バイトとみなされるので、プリント形式からの変換の場合は、ふつうは明示的に桁数を指定します。デリミタ形式からの変換の場合は、数値項目が15バイトを超えることは少ないので、省略するほうがふつうです。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 . 4 という数字を 5 バイトの符号つきゾーン形式の項目に記録するとすれば、ピクチャは s 4 . 1 と指定します。なお、ピクチャは省略できません。

まとめると、プリント形式からの変換の場合は、

入力幅 = 8、ピクチャ = s 4 . 1 のような指定になり、

デリミタ形式からの変換の場合は、

入力幅=省略、ピクチャ=s4.1 のような指定になるのがふつうです。

ただし、入力幅が15バイトを超えるときは、

入力幅=20、ピクチャ=u15.2 のように、ダミーの入力幅を指定します。

[表示・パック(J)]

ホストの文字形式数値項目を、Windows COBOLのパック形式数値項目に変換します。通常は、ホストがUnixまたはWindowsの場合に使います。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

入力幅は、バイト数で指定します。省略すると15バイトとみなされるので、プリント形式からの変換の場合は、ふつうは明示的に桁数を指定します。デリミタ形式からの変換の場合は、数値項目が15バイトを超えることは少ないので、省略するほうがふつうです。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 . 4 という数字を 3 バイトの符号つきパック形式の項目に記録するとすれば、ピクチャは s 4 . 1 と指定します。なお、ピクチャは省略できません。

まとめると、プリント形式からの変換の場合は、

入力幅 = 8、ピクチャ = s 4 . 1 のような指定になり、

デリミタ形式からの変換の場合は、

入力幅=省略、ピクチャ=s4.1 のような指定になるのがふつうです。

ただし、入力幅が15バイトを超えるときは、

入力幅=20、ピクチャ=u15.2 のように、ダミーの入力幅を指定します。

[表示・2進(K)]

ホストの文字形式数値項目を、Windowsの2進形式整数・小数項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

入力幅は、バイト数で指定します。省略すると15バイトとみなされるので、プリント形式からの変換の場合は、ふつうは明示的に桁数を指定します。デリミタ形式からの変換の場合は、数値項目が15バイトを超えることは少ないので、省略するほうがふつうです。

2 進キャスト/ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 1 2 3 . 4 という数字を 4 バイトの符号つき 2 進形式の項目に記録するとすれば、2 進キャスト/ピクチャは i 4 s 4 . 1 と指定します。なお、2 進キャストは省略できません。まとめると、プリント形式からの変換の場合は

入力幅 = 10、 2 進キャスト / ピクチャ = i 4 s 4 . 1 のような指定になり、 デリミタ形式からの変換の場合は

入力幅=省略、2 進キャスト / ピクチャ= i 4 s 4 . 1 のような指定になるのがふつうです。ただし、入力幅が 1 5 バイトを超えるときは、

入力幅 = 20、2 進キャスト / ピクチャ = 18u15.2 のように、ダミーの入力幅を指定します。

[2進・表示(L)]

ホストの2進形式整数・小数項目を、Windowsの文字形式数値項目に変換します。変換結果は右詰めになります。

2 進キャスト / ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字が 4 バイトの符号つき 2 進形式の項目に記録されているとすれば、2 進キャスト / ピクチャは i 4 s 4.1 と指定します。なお、2 進キャストは省略できません。

出力幅は省略できます。出力幅を省略すると、

入力幅	出力幅	
1	3	
2	5	
3	8	
4	1 0	
5	1 3	左記の値を基本として、
6	1 5	符号つきなら + 1、ピクチャ指定の小数部があれば + 1とし、
7	1 7	さらに、ピクチャ指定のほうが大きければ、
8	1 8	整数部桁数に、符号つきなら+1、小数部があれば+1+小数点桁数

の要領で2進キャスト/ピクチャから自動的に計算された値が使われます。例を示すと、入力幅=i4s4.1、出力幅=省略という指定は、出力幅=10という指定と同じです。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

「ゾーン・2進(M)]

ホストのCOBOLのゾーン形式数値項目を、Windowsの2進形式整数・小数項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が5バイトの符号つきゾーン形式の項目に記録されているとすれば、 入力ピクチャは s4.1 と指定します。

2 進キャスト/ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 1 2 3 . 4 という数字を 4 バイトの符号つき 2 進形式の項目に記録するとすれば、 2 進キャスト/ピクチャは i 4 s 4 . 1 と指定します。なお、2 進キャストは省略できません。

入力ピクチャは省略できます。入力ピクチャを省略すると、「2進キャスト/ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、2進キャスト/ピクチャ=i4s4.1という指定は、入力ピクチャ=s4.1を指定したのと同じ意味になります。

[パック・2進(N)]

ホストのCOBOLのパック形式数値項目を、Windowsの2進形式整数・小数項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、 入力ピクチャは s 4 . 1 と指定します。

2 進キャスト / ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 1 2 3 . 4 という数字を 4 バイトの項目に記録するとすれば、2 進キャスト / ピクチャは i 4 s 4 . 1 と指定します。なお、2 進キャストは省略できません。

入力ピクチャは省略できます。入力ピクチャを省略すると、「2進キャスト/ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、2進キャスト/ピクチャ=i4s4.1という指定は、入力ピクチャ=s4.1を指定したのと同じ意味になります。

[2進・ゾーン(0)]

ホストの2進形式整数・小数項目を、Windows COBOLのゾーン形式数値項目に変換します。

2 進キャスト / ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字が 4 バイトの符号つき 2 進形式の項目に記録されているとすれば、2 進キャスト / ピクチャは i4s4.1 と指定します。なお、2 進キャストは省略できません。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字を5バイトの項目に記録するとすれば、ピクチャは s4.1 と指定します。

出力ピクチャは省略できます。出力ピクチャを省略すると、「2進キャスト/ピクチャ」と同じとみなされます。たとえば、2進キャスト/ピクチャ=i4s4.1、出力ピクチャ=省略という指定は、出力ピクチャ=s4.1を指定したのと同じ意味になります。

[2進·パック(P)]

ホストの2進形式整数・小数項目を、Windows COBOLのパック形式数値項目に変換します。

2 進キャスト / ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字が 4 バイトの符号つき 2 進形式の項目に記録されているとすれば、2 進キャスト / ピクチャは i 4 s 4.1 と指定します。なお、2 進キャストは省略できません。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字を3バイトの項目に記録するとすれば、ピクチャは s4.1 と指定します。

出力ピクチャは省略できます。出力ピクチャを省略すると、「2進キャスト/ピクチャ」と同じとみなされます。たとえば、2進キャスト/ピクチャ=i4s4.1、出力ピクチャ=省略という指定は、出力ピクチャ=s4.1を指定したのと同じ意味になります。

[2進·2進(Q)]

ホストの2進形式整数・小数項目を、Windowsの2進形式整数・小数項目項目に変換します。

2 進キャスト/ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 1 2 3 . 4 という数字を 4 バイトの符号つき 2 進形式の項目とする場合、2 進キャスト/ピクチャは i 4 s 4 . 1 と指定します。

2 進キャスト / ピクチャは入力または出力のどちらかを省略できます。入力を省略すると「出力と同じ」とみなされ、出力を省略すると「入力と同じ」とみなされます。なお、入力 2 進キャスト / ピクチャと出力 2 進キャスト / ピクチャを同時に省略することはできません。

[バイナリ反転(R)]

バイナリ反転は、「コード変換を一切せずに、幅分のデータをバイト単位で左右反転する」という変換方法です。2進数値データ(整数、小数、実数)は、ホスト側が正順であるのに対して、Windows側が逆順であることが多いので、2進数値データの内容をそのままバイト単位で左右反転する場合等に使用します。2進・2進変換のような加工機能はありませんが、その分だけ処理が高速です。

幅は、10進のバイト数で指定します。たとえば4バイトの入力データが、16進表現で、01 AB CD EF であれば、出力データは EF CD AB 01 になります。

(6)引用符はずしの指定

ホストがUnix、Windowsの場合、デリミタ形式からの変換の場合には、項目が引用符(")でくくられていることがあります。その引用符は無視します。また、引用符でくくったなかにデリミタがあっても、ただの文字データとして扱います。

(7)改行コード挿入の指定

Windows側(出力)レコードに改行コード(CR/LF=0D0AH)を挿入します。 改行コード挿入には 通常、強制 の2種類がありますが、ホスト Winランダムファイル変 換では違いはありません。

ふつう、項目長の増減がない単純な変換のときマップ項目の最後に指定し、テキストファイル 化するのに使います。そのときは、改行コードの2バイト分Windowsレコード長が増える ので、出力レコード長オプションでその分を\$+2と指定して調整するのを忘れないでください。

変換結果のテキストファイル化には、本来はホスト Winデータファイル変換のプリント形式への変換機能を使うべきです。これなら、変換後のレコード長をほとんど意識する必要がありません。

<マップ設定(詳細)ウインドウ>



マップ設定の記述をするエディットボックスです。

 \sim はファイル(\underline{F}) 編集(\underline{E})のメニューの機能がボタン化されています。 既存のパラメータファイル等を読みます。

の内容をパラメータファイルへ保存します。

直前の編集作業が無効になり、元に戻ります。

選択した文字列が切り取られ、カットバッファに入ります。

選択した文字列がカットバッファに入ります。

/ の操作でカットバッファに入った内容を、カーソルがある位置へ貼りつけます。 カーソルがある位置以降に文字列があれば、挿入になります。

選択した文字列が削除されます。

の文字列をすべて選択状態にします。

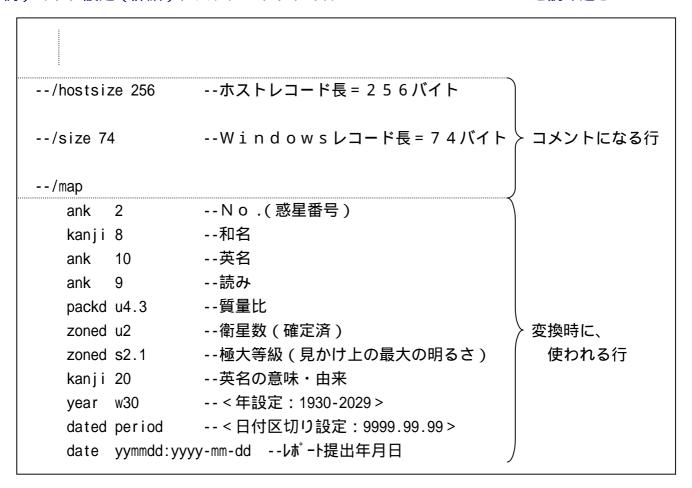
設定が完了したら、OKボタンをクリックします。

キャンセルボタンをクリックすると、すべての入力 / 編集作業が無効になります。

注意 ---- マップ設定(詳細)の記述内容について

マップ設定(詳細)のエディットボックスの中に記述する内容は、レコードレイアウト(/mapを含まない/map以降)のみの記述となります。既存のパラメータファイルを読み込むと、それ以外の記述(/delimited~、/map等)は自動的にコメントになります。記述に誤りがある状態でOKボタンをクリックすると、エラーとなる場合があります。修正してから再度OKボタンをクリックしてください。/map以降の文法については、「コマンド編」の「コマンド型の実行」の章を参照してください。

例)マップ設定(詳細)にパラメータファイル"PNGETRND.P"を読み込む

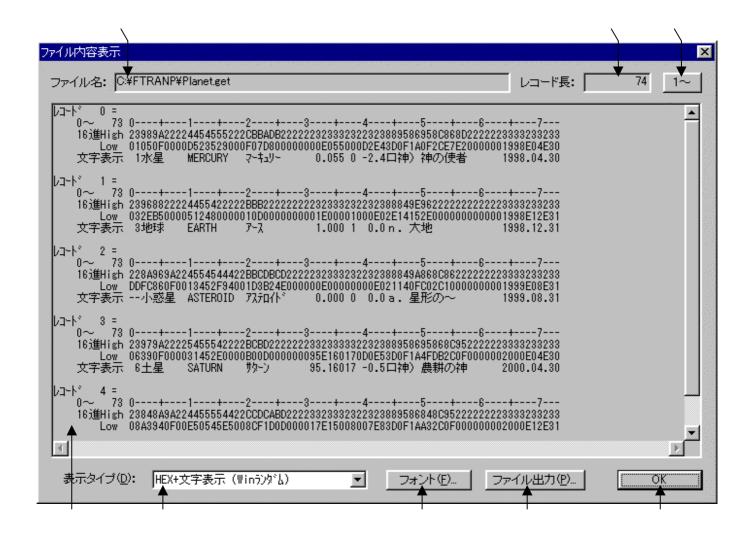


ホスト Winランダムファイル変換の実行

変換ボタンをクリックすると、ホスト Winランダムファイル変換が始まります。

ファイル内容表示ウインドウ

変換結果表示(<u>V</u>)のチェックボックスをONにし、ホスト Winランダムファイル変換を 実行すると、実行ウインドウを閉じた後につぎのファイル内容表示ウインドウが開き、変換先の ファイルの中身を確認することができます。



変換先のファイル名が表示されます。

変換先のファイルのレコード長が表示されます。

変換先のファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、レコード長単位のHEX+文字表示(Winランダム)です。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

フォント (\underline{F}) ボタンをクリックして、 の出力文字フォントを切り替えます。

ファイル出力(P)ボタンをクリックして、の内容をファイルに保存します。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えます。

OKボタンをクリックして、ファイル内容表示ウインドウを閉じます。

注意事項

漢字があるときの注意

漢字項目変換やAnk・漢字まじり項目変換を行うときは、あらかじめ

変換設定の漢字変換方式設定で、適当な漢字変換方式を割り当てておく

のを忘れないでください。また、入力幅、出力幅は漢字データについても

バイト単位で指定

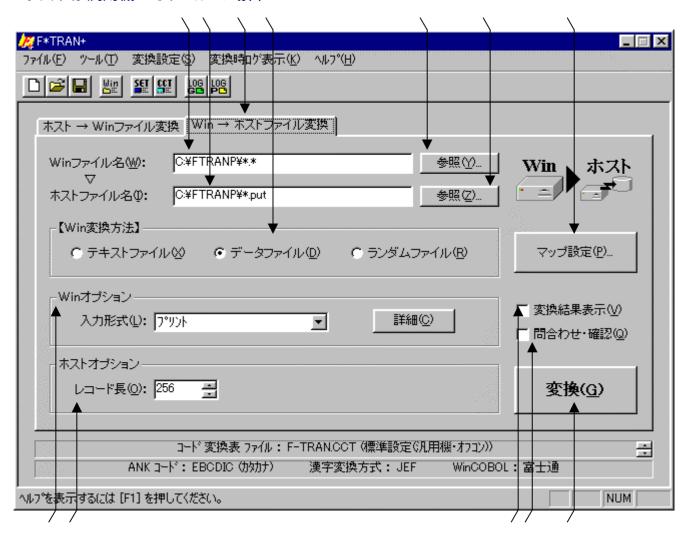
します。漢字の文字数ではないことに注意してください。

その他の注意事項

「ホスト Winファイル変換」の節を参照してください。

ここでは、Win ホストファイル変換の共通事項を説明します。

ホストが汎用機・オフコンの場合



Win ホストのタブをクリックし、メイン処理部をWin ホストファイル変換にします。 変換元のWindowsファイル名を指定します。

変換元のWindowsファイル名を決定する参照ウインドウを開くボタンです。

変換先のホストファイル名を指定します。

変換先のホストファイル名を決定する参照ウインドウを開くボタンです。

変換方法を選択するラジオボタンです。

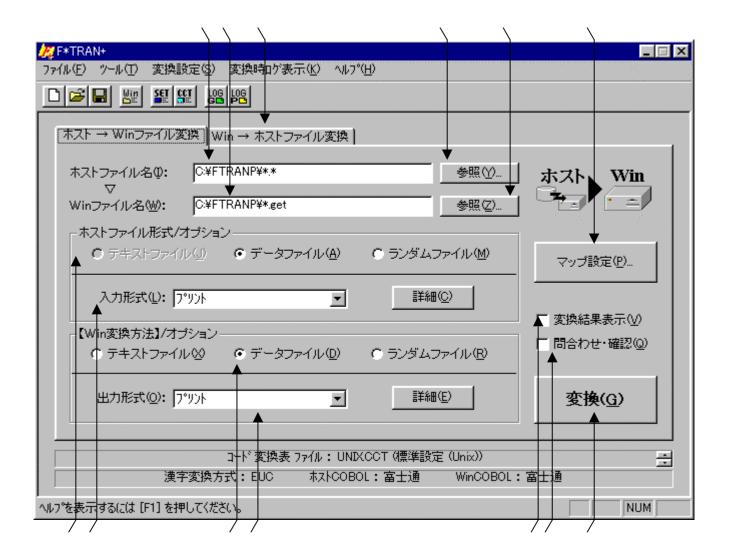
Winファイルのオプション項目です。 の選択によって、オプションの内容が変わります。 ホストファイルのオプション項目です。 の選択によって、オプションの内容が変わります。 マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換実行時の問い合わせ・確認ウインドウ表示の選択をするチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホストがUnix、Windowsの場合



Win のホストタブをクリックし、メイン処理部をWin ホストファイル変換にします。 変換元のホストファイル名を指定します。

変換元のホストファイル名を指定する参照ウインドウを開くボタンです。

変換先のWindowsファイル名を指定します。

変換先のWindowsファイル名を指定する参照ウインドウを開くボタンです。

Winファイルの変換方法を選択するラジオボタンです。変換の基準となる項目です。

Winファイルのオプション設定です。

ホストファイル(出力ファイル)の形式を選択するラジオボタンです。

ホストファイルのオプション設定です。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換実行時の問い合わせ・確認ウインドウ表示の選択をするチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

Windowsファイル指定

[d:][パス名指定]基本ファイル名[.拡張子]

d:はドライブ名

入力側のWindowsファイルを指定します。

ドライブ名は、A:~Z:、@:、?:のどれかで指定します。ドライブ名を指定すると、そのドライブを検索します。

ドライブ名は省略可能です。ドライブ名を省略すると、カレントドライブを検索します。

パス名指定(¥ディレクトリ名¥サブディレクトリ名¥・・・)ができます。指定したディレクトリ配下のファイルを検索します。パス名指定を省略すると、カレントディレクトリを検索します。

基本ファイル名と拡張子には、ワイルドカード文字(*と?)を使うことができます。ワイルドカード文字を使うと、一致するファイルをすべて検索し変換の対象にします。

まとめると、あるディレクトリのファイルをすべて変換したいなら、

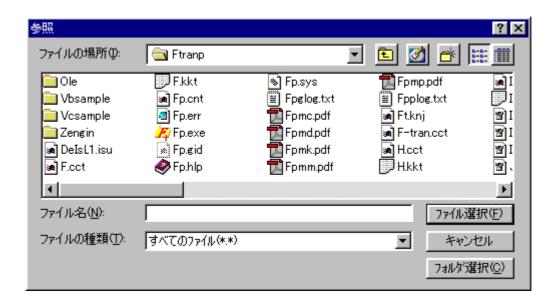
C:*.* のように指定し、

拡張子が. DATのファイルをすべて変換したいなら、

C:*.DAT のような指定になります。

Windowsファイル参照ウインドウ

Windowsファイルの参照ボタンをクリックすると、つぎのウインドウが開きます。



変換元のファイルを直接指定する場合は、ファイル名を選択して、ファイル選択(<u>F</u>)ボタンをクリックします。

変換元の場所(どこのフォルダのファイルを変換の対象にするか)を指定する場合は、フォルダ選択(<u>C</u>)ボタンをクリックします。ファイル名にフォルダまでのフルパスが入力され、確定します。

ホストファイル指定

[d:][パス名指定]基本ファイル名[.拡張子]

d:はドライブ名

出力側のホストファイルを指定します。

ドライブ名は、A: ~Z: ~@: ~?:のどれかで指定します。ドライブ名を指定すると、そのドライブにファイルを作ります。

ドライブ名は省略可能です。ドライブ名を省略すると、カレントドライブにファイルを作ります。

パス名指定(¥ディレクトリ名¥サブディレクトリ名¥・・・)ができます。指定したディレクトリ配下にファイルを作ります。パス名指定を省略すると、カレントディレクトリにファイルを作ります。

基本ファイル名の部分には、ふつうの基本ファイル名、または*を指定します。*を指定するとWindowsファイルの基本ファイル名がホストファイルの基本ファイル名になります。

拡張子を省略すると、拡張子なしのファイルになります。しかし、拡張子をつけたほうがファイルの管理が容易になるので、なるべく適当な拡張子を指定してください。拡張子に*を指定するとWindowsファイルの拡張子がホストファイルの拡張子になります。

以上がディスクファイルの指定方法です。まとめると、基本ファイル名を引き継ぐときは

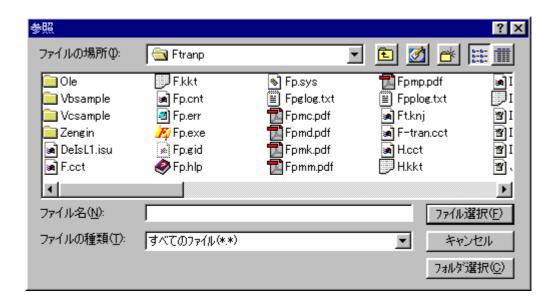
C:*.DAT のような指定になり、

ファイル名をつけ替えるときは

C:NEWNAME.DAT のような指定になります。

ホストファイル参照ウインドウ

ホストファイルの参照ボタンをクリックすると、つぎのウインドウが開きます。



変換先のファイルを直接指定する場合は、ファイル名を選択して、ファイル選択(<u>F</u>)ボタンをクリックします。

変換先の場所(どこのフォルダに変換後のファイルを作成するか)を指定する場合は、フォルダ選択(<u>C</u>)ボタンをクリックします。ファイル名にフォルダまでのフルパスが入力され、確定します。

その他の指定

变換結果表示指定

変換後に、変換先のWindowsファイルの中身を表示ウインドウに出力するかどうかを指定します。出力する場合は、変換結果表示のチェックボックスをONにします。

出力の形式は、変換方法によって異なりますので、それぞれの変換方法の節を参照してください。

問合わせ・確認指定

1ファイルごとに処理を問い合わせるか否かを指定します。

問合わせ・確認のチェックボックスを、

ONにすると、変換するか否かを問い合わせる指定

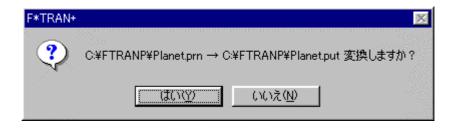
OFFにすると、ファイル名の確認なしで自動変換する指定 となります。

問合わせ・確認指定がONであれば、ファイル名を確認しながら変換できます。

F*TRAN+は、1ファイルごとに変換を実行するか問い合わせてきます。

つぎのどれかで応答してください。この機能は、比較的小さいファイルが多数あって、そのうちいくつかを選んで変換したいときなどに、便利です。

1ファイルの変換



はい(\underline{Y}) いいえ(N)

表示中のファイルを変換する 表示中のファイルは変換しない

2ファイル以上の変換



はい(Y) 表示中のファイルを変換する

すべて変換 (\underline{A}) 全ファイル変換に切り替え、以降のファイルをすべて変換する

いいえ(<u>N</u>)キャンセル表示中のファイルは変換しないこれ以降の変換処理を中断する

問合わせ・確認指定がOFFであれば、ファイル名の確認なしで、自動的に指定のファイルを すべて変換します。こちらがデフォルトです。

変換時の実行ウインドウ

変換ボタンをクリックすると、Win ホストファイル変換が始まり、つぎの実行ウインドウが開きます。



変換中のメッセージを表示するメッセージフィールドです。

変換が始まると、"変換開始」"というメッセージ表示につづいて、

"Windowsファイル名 ホストファイル名 ~ファイル変換."が表示され、

変換が正常終了すると、"変換終了."と表示されます。

変換中の進行状況(進行度%出力)を表示するフィールドです。

変換中のエラーメッセージを表示するフィールドです。

変換中は「中止(Q)」と表示されています。

変換中にこのボタンをクリックすると、変換を途中で中止します。

変換が終了したら、その変換結果を確認して、"閉じる"ボタンをクリックします。

おもにコード変換に関する現在の設定が表示されています。

注意事項

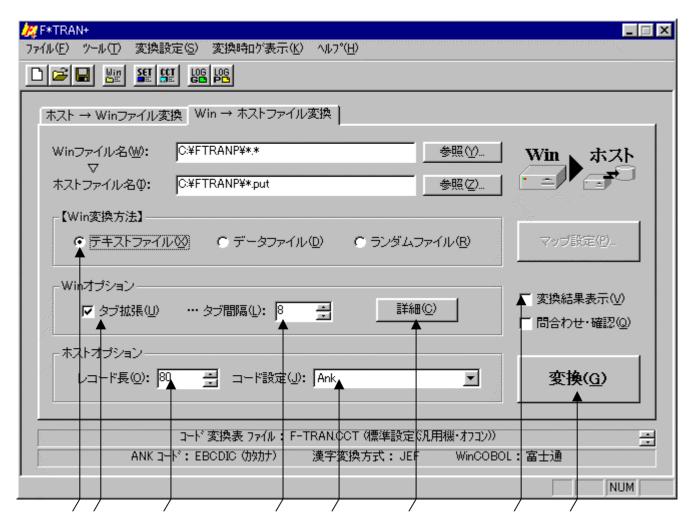
同名ファイルは置換する

すでに同じ名前のホストファイルがある場合、自動的に元のファイルを削除し、新たに変換したファイルで置き替えます。このとき警告メッセージは出ないので注意してください。

4.7 Win ホストテキストファイル変換

Win ホストテキストファイル変換の操作方法について説明します。

ホストが汎用機・オフコンの場合



変換方法のテキストファイル(X)をクリックし、

Win ホストテキストファイル変換を選択します。

タブ拡張の有無を指定するオプションのチェックボックスです。

でタブ拡張のチェックをONにしたとき、タブ間隔を指定できます。

Winファイルの詳細オプションボタンです。EOF検査の選択ができます。

ホストファイルのレコード長を指定します。

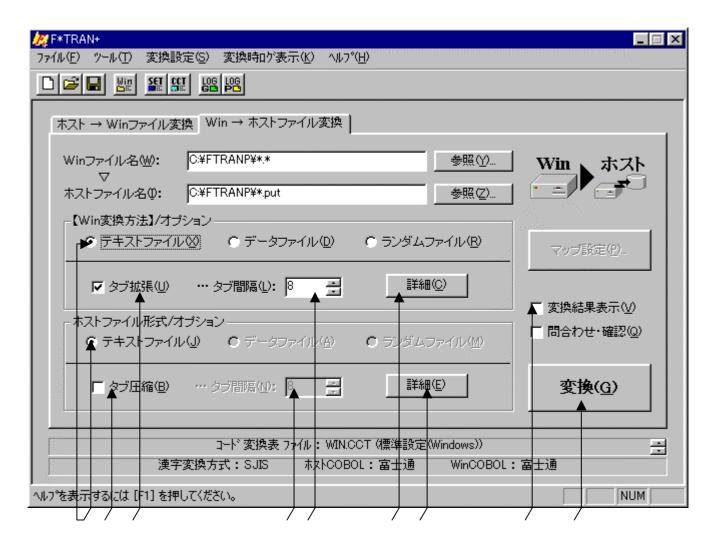
ANK変換かAnk・漢字(KI/KO付)混在変換かを指定するオプションです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

Win ホストファイル変換の共通指定項目は、「Win ホストファイル変換」の節ですでに詳しく説明しました。そちらを参照してください。

ホストがUnix、Windowsの場合



Win変換方法のテキストファイル (X) をクリックし、ホスト Winテキストファイル 変換を選択します。ホストファイル形式は、テキストファイル (\underline{J}) 固定になります。 タブ拡張の有無を指定するオプションのチェックボックスです。

でタブ拡張のチェックをONにしたとき、タブ間隔を指定できます。

Winファイルの詳細オプションボタンです。EOF検査の選択ができます。

タブ圧縮の有無を指定するオプションのチェックボックスです。

でタブ圧縮のチェックをONにしたとき、タブ間隔を指定できます。

ホストファイルの詳細オプションボタンです。EOF付加の選択ができます。

ホストがWindowsの場合のみ有効です。ホストがUnixの場合はありません。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホスト Winファイル変換の共通指定項目は、「ホスト Winファイル変換」の節ですでに詳しく説明しました。そちらを参照してください。

オプションの指定

タブ拡張のオプション

タブ拡張の有無と、タブ拡張するときのタブ間隔を指定します。タブ拡張とは、TAB(09H)をつぎのタブ位置の直前までの連続スペースに展開することです。

タブ拡張のチェックボックスを、

- ONにすると、タブ拡張する指定
- OFFにすると、タブ拡張しない指定 となります。

タブ拡張指定がONであれば、タブ拡張します。タブ間隔は、2~255の範囲で指定し、それがレコードのおわりまで繰り返し適用されます。タブ間隔のデフォルトは標準のタブ間隔(8桁きざみ)になります。これがデフォルトです。

タブ拡張指定がOFFであれば、タブ拡張はしません。

EOF検査指定

EOFコード(1AH)を検査するか否かを指定します。

EOF検査のチェックボックスを、

ONにすると、EOFコードを検査する指定

OFFにすると、EOFコードを検査しない指定 となります。

EOF検査指定がONであれば、EOFコードを検査し、EOFコードが現れたら変換を終了します。こちらがデフォルトです。

EOF検査指定がOFFであれば、EOFコードを検査しません。単なるデータとして扱います。

ホストファイル・レコード長指定 < ホストが汎用機・オフコンの場合のみ >

ホストファイルのレコード長を1~32767の範囲の10進数で指定します。デフォルトは、80バイトです。レコード長の指定が間違っていると正しいデータ変換が行われませんので、この指定は極めて重要です。

コード設定オプション <ホストが汎用機・オフコンの場合のみ>

ホストファイル側のコード系(ANK変換かAnk・漢字(KI/KO付)混在変換か)を指定します。

Ank指定

Ank指定すると、すべてANKデータとして変換します。これがデフォルトです。

 Windows側
 ホスト側

 JIS8/ASCII
 EBCDIC(カタカナ)

 EBCDIC(英小文字)

の3とおりの変換が可能です。あらかじめ、変換設定のANKコード設定でホストファイル側のコード系を設定しておかなければいけません(ふつう、セットアップ時に1回だけ行います)。 漢字がまじっているときは、つぎのAnk・漢字(KI/KO付)混在指定を使ってください。

Ank・漢字(KI/KO付)混在指定

ANK・漢字混在で、KI/KOつきに変換するとき、この指定をします。あらかじめ、変換設定の漢字変換方式設定で適切な漢字変換方式の割り当てをしておかなければいけません(ふつう、セットアップ時に1回だけ行います)。

タブ圧縮のオプション <ホストがUnix、Windowsの場合のみ>

タブ圧縮の有無と、タブ圧縮するときのタブ間隔を指定します。タブ圧縮とは、タブ位置の直前までつづく2個以上の連続スペースを、TAB(09H)に置き替えることです。

タブ圧縮のチェックボックスを、

ONにすると、タブ圧縮する指定

OFFにすると、タブ圧縮しない指定 となります。

タブ圧縮指定がONであれば、タブ圧縮します。タブ間隔は、2~255の範囲で指定し、それがレコードのおわりまで繰り返し適用されます。タブ間隔のデフォルトは標準のタブ間隔(8桁きざみ)になります。

タブ圧縮は、ソースプログラムなど空白部分が多いファイルの、変換後のファイル容量を減ら すのに効果的です。

なお、文字列定数中のスペースまでTABに変換してしまうことを避けるため、アポストロフィ(')か引用符(")が見つかると、その行についてはそこでタブ圧縮を打ち切ります。 タブ圧縮指定がOFFであれば、タブ圧縮はしません。これがデフォルトです。

EOF付加指定 <ホストがWindowsの場合のみ>

EOFコード(1AH)の扱いを指定します。

EOF付加のチェックボックスを、

ONにすると、EOFコードをつける指定

OFFにすると、EOFコードをつけない指定 となります。

EOF付加指定がONであれば、WindowsファイルのおわりにEOFコードをつけます。 現在では少なくなりましたが、テキストファイルのおわりにEOFコードがついていないとエラーにするソフトがあります。その場合にも対処するための機能です。

EOF付加指定がOFFであれば、EOFコードはつけません。これがふつうだと思ってください。こちらがデフォルトです。

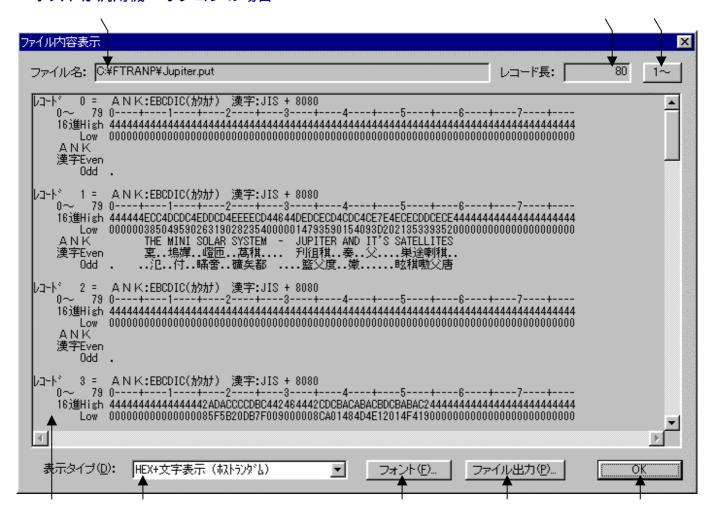
Win ホストテキストファイル変換の実行

変換ボタンをクリックすると、Win ホストテキストファイル変換が始まります。

ファイル内容表示ウインドウ

変換結果表示(<u>V</u>)のチェックボックスをONにし、Win ホストファイル変換を実行すると、実行ウインドウを閉じた後につぎのファイル内容表示ウインドウが開き、変換先のファイルの中身を確認することができます。

<ホストが汎用機・オフコンの場合>



変換先のファイル名が表示されます。

変換先のファイルのレコード長が表示されます。

変換先のファイルの中身が、

レコード長単位のHEX+文字表示(ホスト)形式で出力されます。

の表示タイプは、HEX+文字表示(ホスト)固定です。

フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

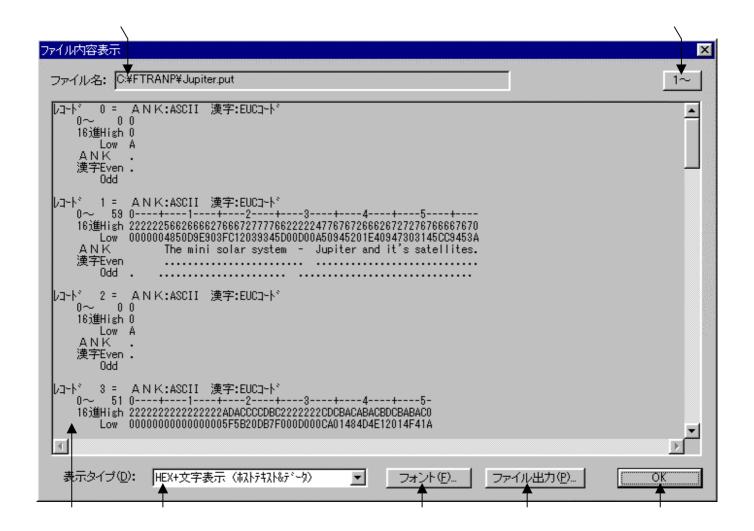
ファイル出力 (\underline{P}) ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

OKボタンをクリックすると、ファイル内容表示ウインドウを閉じます。

<ホストがUnixの場合>



変換先のファイル名が表示されます。

変換先のファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、改行コード単位のHEX+文字表示(ホストテキスト&データ)です。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

フォント(\underline{F})ボタンをクリックして、 の出力文字フォントを切り替えることができます。

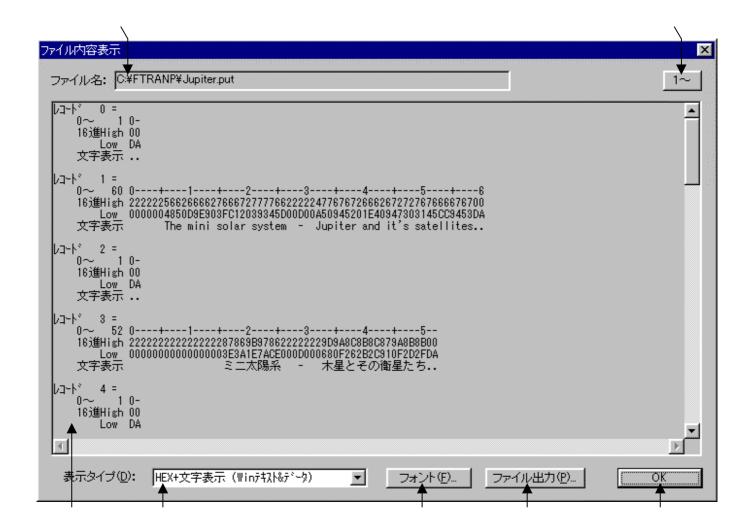
ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

OKボタンをクリックすると、ファイル内容表示ウインドウを閉じます。

<ホストがWindowsの場合>



変換先のファイル名が表示されます。

変換先のファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、改行コード単位のHEX+文字表示(Winテキスト&データ)です。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

フォント(\underline{F})ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

注意事項

<u>漢字があるときはコード設定オプションを A n k・漢字(KI/KO付)混在にすることを忘</u>れずに

漢字が入っているときは、あらかじめデータ交換の相手のシステムに合わせて、

変換設定の漢字変換方式設定で、適当な漢字変換方式を割り当てておく

のを忘れないでください。また、ホストが汎用機・オフコンの場合、変換のとき、

コード設定オプションをAnk・漢字(KI/KO付)混在にするのを忘れがち

なので注意してください。

タブ拡張について

ふつう、タブ拡張は必須です。多くの場合、ホストファイルを受けとるシステムのほうでタブをサポートしていないか、サポートしていてもタブ間隔の設定が異なっています。

あふれた分は捨てられる

Windowsファイルの1行が、変換後ホストファイルの1レコードに入り切れないときは、あふれた分が切り捨てられます。そのときはもっと大きなレコード長を指定して再変換してください。とくに、ホストが汎用機・オフコンの場合、Ank・漢字(KI/KO付)混在変換をしたとき、KI/KOが挿入されて、変換後1行の長さが増えることに注意してください。

改行コードがないと、1件だけの変換になる

Windowsファイルの各レコードの末尾に改行コードがついていないと、先頭の1件だけ 変換されます。この場合は、Win ホストランダムファイル変換を使うべきです。

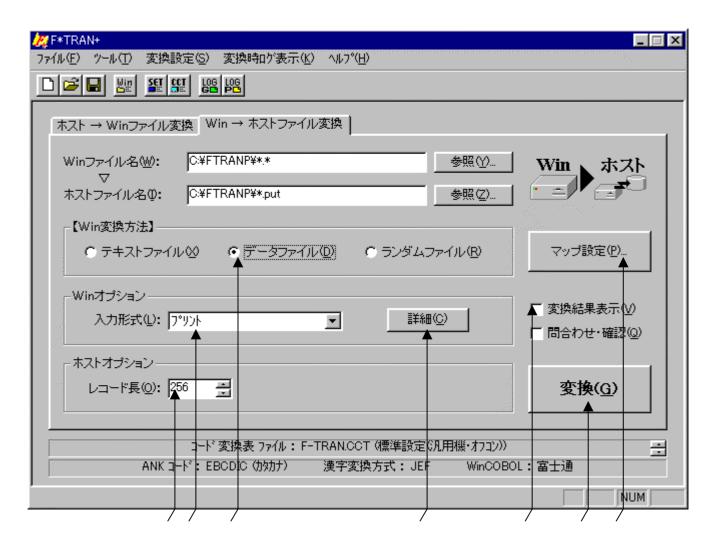
その他の注意事項

「Win ホストファイル変換」の節を参照してください。

4.8 Win ホストデータファイル変換

Win ホストデータファイル変換の操作方法について説明します。

ホストが汎用機・オフコンの場合



変換方法のデータファイル(D)をクリックし、

Win ホストデータファイル変換を選択します。

入力ファイル形式を指定するオプションです。

Winファイルの詳細オプションボタンです。EOF検査、タブ拡張、空行無視の選択ができます。

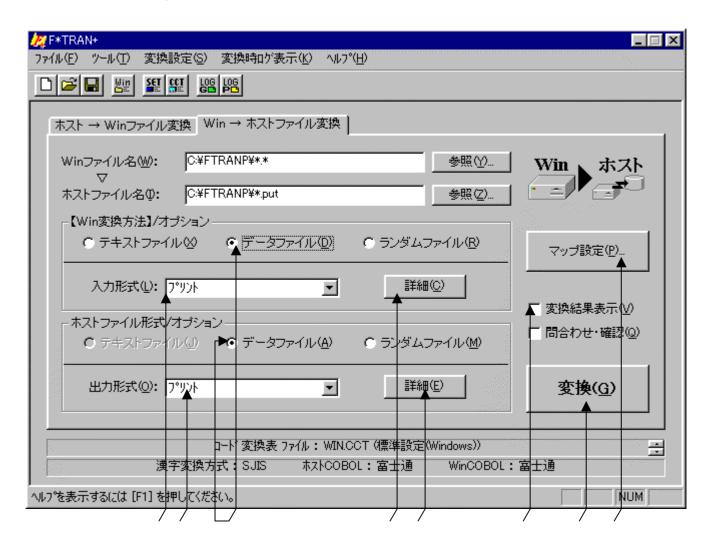
ホストファイルのレコード長を指定します。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホストがUnix、Windowsの場合



Win変換方法のデータファイル (\underline{D})をクリックし、Win ホストデータファイル変換を選択します。ホストファイル形式は、データファイル (\underline{A})、ランダムファイル (\underline{M})の選択ができますが、上の例はホストファイル形式がデータファイル (\underline{A})です。

入力ファイル形式を指定するオプションです。

Winファイルの詳細オプションボタンです。EOF検査、タブ拡張、空行無視の指定ができます。

出力ファイル形式を指定するオプションです。

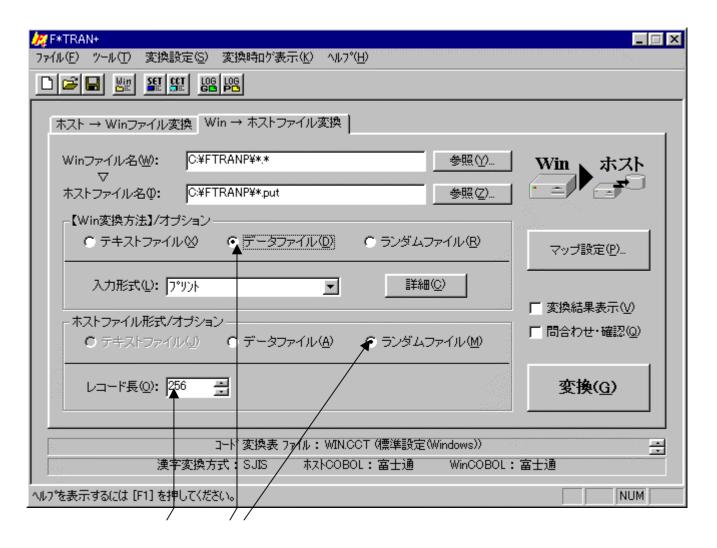
ホストファイルの詳細オプションボタンです。EOF付加の選択ができます。

ホストがWindowsの場合のみ有効です。ホストがUnixの場合はありません。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。



Win変換方法のデータファイル(\underline{D})をクリックし、Win ホストデータファイル変換を選択します。ホストファイル形式は、データファイル(\underline{A})、ランダムファイル(\underline{M})の選択ができますが、上の例はホストファイル形式がランダムファイル(\underline{M})です。ホストファイルのレコード長を指定します。

Win ホストファイル変換の共通指定項目は、「Win ホストファイル変換」の節ですでに詳しく説明しました。そちらを参照してください。

オプションの指定

入力形式オプション

変換するWindowsファイルがプリント形式かデリミタ形式か、デリミタ形式ならコンマ区切り・タブ区切り・スペース区切りのどれなのかを指定します。

プリント ------- プリント形式を変換する デリミタ(コンマ) ----- デリミタ形式(コンマ)を変換 デリミタ(タブ) ------ デリミタ形式(タブ)を変換 デリミタ(スペース) ---- デリミタ形式(スペース)を変換

プリントを指定すると、デリミタ(区切り文字)なしのプリント形式のファイル(固定長テキストファイル(SDF形式)。ただし、最終項目だけ可変長でもよい)を変換できます。

デリミタを指定すると、デリミタ形式(区切り文字つき)のテキストファイルを変換できます。 区切り文字の種類によって、さらに細かい形式が決まります。デリミタは、

コンマ区切り形式(CSV形式、K3形式) タブ区切り形式(TAB=09H) スペース区切り形式(SP=20H) の3つのなかから指定できます。

後述のマップ設定オプションで、デリミタ検出 = コンマ (,) を指定したところに、上記の区切り文字があるとみなされます。

EOF検査指定

EOFコード(1AH)を検査するか否かを指定します。

EOF検査のチェックボックスを、

ONにすると、EOFコードを検査する指定 OFFにすると、EOFコードを検査しない指定 となります。

EOF検査指定がONであれば、EOFコードを検査し、EOFコードが現れたら変換を終了します。こちらがデフォルトです。

EOF検査指定がOFFであれば、EOFコードを検査しません。単なるデータとして扱います。

タブ拡張のオプション

タブ拡張の有無と、タブ拡張するときのタブ間隔を指定します。タブ拡張とは、TAB(09H)をつぎのタブ位置の直前までの連続スペースに展開することです。

タブ拡張のチェックボックスを、

- ONにすると、タブ拡張する指定
- OFFにすると、タブ拡張しない指定 となります。

タブ拡張指定がONであれば、タブ拡張します。タブ間隔は、2~255の範囲で指定し、それがレコードのおわりまで繰り返し適用されます。タブ間隔のデフォルトは標準のタブ間隔(8桁きざみ)になります。これがデフォルトです。

タブ拡張指定がOFFであれば、タブ拡張はしません。

空行無視のオプション

空行を無視するかどうかを指定します。

ONにすると、空行を無視する指定

OFFにすると、空行を無視しない指定 となります。

ホストファイル・レコード長指定 <ホストファイルが固定長の場合のみ>

ホストファイルのレコード長を1~32767の範囲の10進数で指定します。デフォルトは、256バイト です。レコード長の指定が間違っていると正しいデータ変換が行われませんので、この指定は極めて重要です。

出力形式オプション 〈ホストファイルが可変長の場合のみ〉

データとしてのテキストファイルにはいくつもの形式があるので、どの形式にするかを指定します。 つぎの、 7 種類の中から選択します。

プリント -------------------- プリント形式に変換する デリミタ(コンマ ・ 圧縮) ---- デリミタ形式(コンマ)に変換、圧縮あり デリミタ(タブ ・ 圧縮) ---- デリミタ形式(タブ)に変換、圧縮あり デリミタ(スペース・ 圧縮) ---- デリミタ形式(スペース)に変換、圧縮あり デリミタ(コンマ ・非圧縮) ---- デリミタ形式(コンマ)に変換、圧縮なし デリミタ(タブ ・非圧縮) ---- デリミタ形式(タブ)に変換、圧縮なし デリミタ(スペース・非圧縮) ---- デリミタ形式(スペース)に変換、圧縮なし

プリントを指定すると、デリミタ(区切り文字)なしで変換され、プリント形式(固定長のテキストファイル [SDF形式])になります。これがデフォルトです。

デリミタを指定すると、デリミタ形式(区切り文字つきのテキストファイル)に変換されます。 デリミタの種類によってさらに細かい形式が決まります。デリミタは、

コンマ区切り形式(CSV形式、K3形式) タブ区切り形式(TAB=09H) スペース区切り形式(SP=20H) の3つのなかから指定できます。

通常は、コンマ区切り(CSV)形式への変換が一般的です。さらに、

変換後に圧縮をかける(可変長になる。デフォルト) 変換後に圧縮しない(固定長のまま) を指定できます。

圧縮をかけると本来不要なスペース、つまり、

レコードの先頭・末尾のスペースデリミタの前後のスペース引用符の前のスペースが削除されます。

圧縮をかける機能があるのは、つぎのような理由があるためです。

変換後のWindowsファイルの容量を、大幅に減らすことができる 一部の市販ソフトでは、不要なスペースがあるとうまくデータを読み込めない

EOF付加指定 〈ホストがWindowsで、可変長ファイルの場合のみ〉

EOFコード(1AH)の扱いを指定します。

EOF付加のチェックボックスを、

ONにすると、EOFコードをつける指定 OFFにすると、EOFコードをつけない指定 となります。

EOF付加指定がONであれば、WindowsファイルのおわりにEOFコードをつけます。 現在では少なくなりましたが、テキストファイルのおわりにEOFコードがついていないとエラーにするソフトがあります。その場合にも対処するための機能です。

EOF付加指定がOFFであれば、EOFコードはつけません。これがふつうだと思ってください。こちらがデフォルトです。

マップ設定オプション

このマップ設定オプションで、項目別の細かい変換方法を指示します。本来なら、自動的に項目を認識して変換ができると便利です。しかし、ホストの、とくにCOBOLのデータには、データ自身に桁数や小数点位置の判断に必要な情報が含まれていない、という特性があります。そのため自動変換は原理的に不可能なのです。

マップ設定のボタンをクリックすると、つぎのウインドウが開きます。



マップ設定には、

簡易設定 ---- 設定ウインドウに従って、必要な値を入力するだけで設定ができます。

通常は、こちらの機能だけで間に合う場合がほとんどです。

詳細設定 ---- フリーフォーマットで、マップ設定を記述するエディタモードです。

簡易設定だけでは設定できない特殊な設定をする場合、あるいは、 既存のパラメータファイル等を流用したい場合等に使用します。

マップ設定の文の書き方を覚えなくては設定できません。

の2とおりの方法があります。

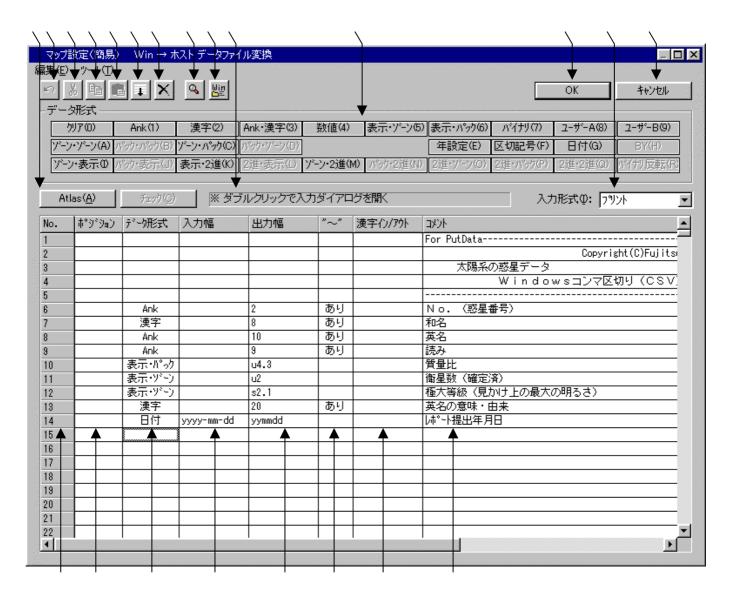
注意 ---- 簡易設定と詳細設定の組み合わせ

簡易設定で設定した値は、詳細設定で読み込むことができるテキスト形式に直されます。よって、簡易設定をした後の項目に、詳細設定で特殊設定をつけ足すこともできます。

注意 ---- 詳細設定 簡易設定の順で設定した場合の制約

詳細設定 簡易設定の順で設定した場合は、原形が変わってしまう場合があります。これは、 詳細設定で設定した項目が簡易設定で読み込まれることにより、スリム化されてしまうからです。 ただし、原形が変わってしまっても、設定内容が変わるわけではありません。

<マップ設定(簡易)ウインドウ>



No.は、1~2000です。ここは非入力項目で、行選択の時にクリックします。項目の桁位置等を入力します。通常、項目を組み替えるとき以外は入力しません。 ダブルクリックすると、入力サブウインドウが開きます。

項目のデータ形式を15種類の中から指定します。

ダブルクリックすると、入力サブウインドウが開きます。

項目の入力幅を入力します。通常、デリミタ形式ファイルを変換する場合は省略します。項目の出力幅を入力します。プリント形式ファイルで、出力幅 = 入力幅の場合は省略します。データ形式がゾーン / パック / 2 進指定の場合、ピクチャまたは 2 進ピクチャで指定します。デリミタ形式から変換するときに、項目の引用符をはずす指定ができます。ホストがUnix、Windowsの場合、デリミタ形式の項目を引用符でくくる指定にもなります。ホストが汎用機・オフコンの場合は、項目の前後に漢字イン / アウトを挿入する指定です。

ホストが川 n i x、W i n d o w s の場合は、改行コードを挿入する指定です。

コメントの入力ができます。

セルポインタの入力ガイダンスメッセージが出力されるフィールドです。 誤入力をした場合のエラーメッセージも出力されます。

 \sim は編集(\underline{E})、ツール(\underline{T})のメニューの機能がボタン化されています。 直前の編集作業が無効になり、元に戻ります。

1 行または複数行を選択(の操作)をした後に、このボタンをクリックすると、 選択した項目の内容が切り取られ、カットバッファに入ります。

1 行または複数行を選択(の操作)をした後に、このボタンをクリックすると、 選択した項目の内容がカットバッファに入ります。

カットバッファに入った内容(、 の操作)を、セルポインタがある行へ貼りつけます。 すでに、セルポインタ以降の行に設定項目がある場合は、挿入になります。

セルポインタがある行以降が1行づつ下がります。通常は、行挿入をする場合に使用します。 選択した項目の内容が削除されます。

変換プレビュー(変換前/変換後のデータ確認)ができます。

Winファイルエディタが起動します。

Windowsファイルの中身を見る(編集も可能)ことができます。

セルポインタが選択項目にある場合に、選択肢のボタンが表示されるエリアです。 入力形式オプションの指定(プリント、デリミタ3種類)を変更できます。 設定が完了したら、OKボタンをクリックします。 キャンセルボタンをクリックすると、すべての入力/編集作業が無効になります。

マルチレコードの設定ができるAtlasモードに切り替えるボタンです。 詳細は、マルチレコード編のマニュアルを参照してください。

注意 ---- マップ未設定時のデータ変換動作

マップ未設定の場合は、すべてをANKデータとみなしてデータ変換を行います。

注意 ---- デリミタ検出について

デリミタ形式のファイルを変換するときは、なんらかの方法で項目分けをしなければいけません。コンマを使うと行頭からデリミタ、デリミタからデリミタ、デリミタから行末までを項目として認識します。なお、引用符でくくられた項目は、そのなかにデリミタがあっても単なる文字データとして扱われます。

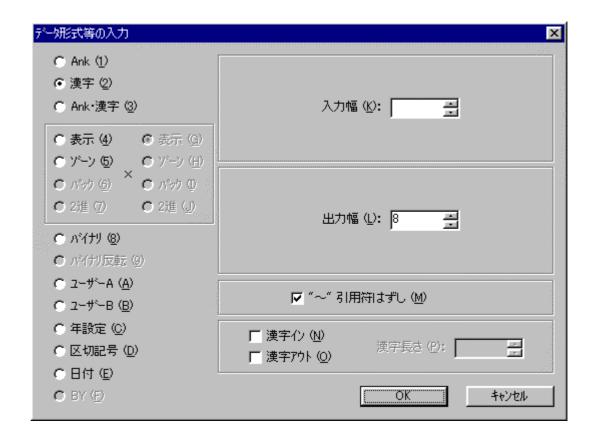
マップ設定(簡易)で項目を設定し、デリミタ形式のファイルを変換する場合、変換元ファイルの項目の後にデリミタ(区切り文字)があることを前提に変換します(最終項目は除く)。検出される区切り文字は指定された入力形式により、コンマ、タブ、スペースの3つのいずれかになります。なお、プリント形式への変換のときは、デリミタ検出の指定は無効になります。また、デリミタ検出に設定したが、ある項目についてはデリミタ検出をしたくない場合は、簡易設定後に、詳細設定を行い修正することができます。

<マップ設定(簡易)サブウインドウ>

ポジションの欄をダブルクリックすると、つぎのサブウインドウが開きます。入出力桁位置や 入出力スキップの設定を簡便に行うことができます。

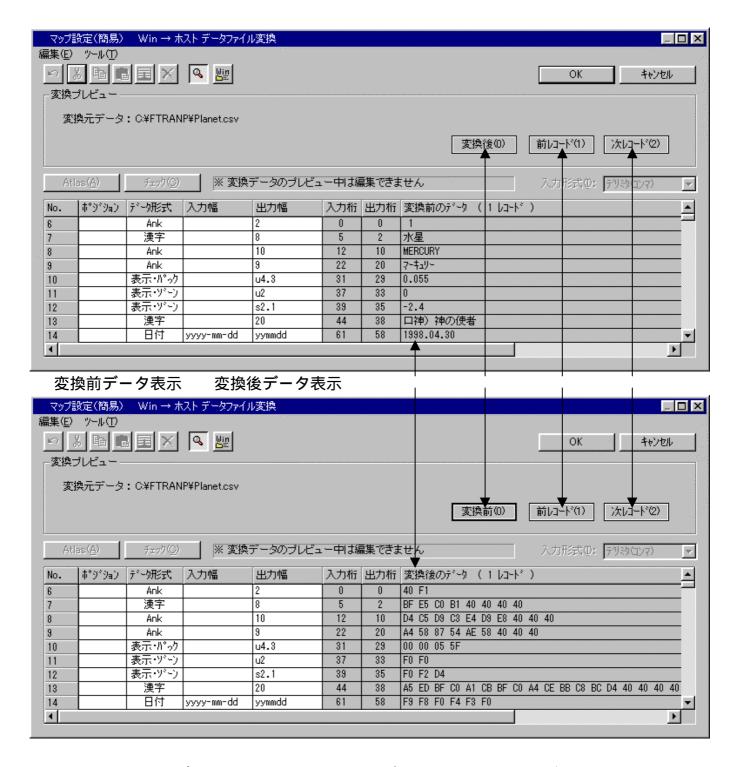


データ形式の欄をダブルクリックすると、つぎのサブウインドウが開きます。データ形式から引用符はずし(ホストが汎用機・オフコンの場合、漢字項目は漢字イン / アウト)までの設定を簡便に行うことができます。



<マップ設定(簡易)・変換プレビュー機能>

変換プレビューボタンをクリックすると、マップ設定(簡易)のウインドウはつぎのような状態になり、変換前/変換後のデータを確認することができます。



変換前 / 変換後のデータ表示の切り替えを行うボタンです。 の表示が変わります。 前レコード / 次レコード表示のためのボタンです。

ここには、入力レコード/出力レコードのポジション(桁)が表示されます。

データ表示は、変換前が文字、変換後が16進、未定義個所が文字になります。

プリント形式とデリミタ形式では指定の方法と考え方がかなり違うので、この 2 つを必要に応じて分けて説明することにします。

デリミタ形式の場合には、Windows側(入力項目)が可変長で、ホスト側(出力項目)が固定長だという点に留意して読み進めてください。

(1)ポジション(桁位置)の指定

プリント形式の場合

変換対象にするWindows側(入力)の桁位置や、変換結果を書き込むホスト側(出力)の桁位置を、別の任意の位置に移動できます。現在、処理対象にしている桁位置を、この指定で強制的に変更できます。この機能を利用すると、項目の組み替えなどが簡単に実現できます。

入力桁位置は、ふつう10進数で指定します。式による指定もでき、そのなかでは、

. Windows(入力)側の現在の桁位置 という特殊変数が使えます。

たとえば、 . - 40 と指定すれば、

今の入力桁位置から40バイト戻る という意味になります。

出力桁位置を移動することもできます。ふつう、 : 10進数 で桁位置を指定します。 10進数の代わりに式による指定もでき、そのなかでは、

- \$ ホスト側(出力)の末尾
- . ホスト側(出力)の現在の桁位置 という特殊変数が使えます。

デリミタ形式の場合

変換結果を書き込むホスト側(出力)の桁位置を、別の任意の位置に移動できます。現在、処理対象にしている桁位置を、この指定で強制的に変更できます。この指定を利用すると、項目の組み替えなどが簡単に実現できます。

デリミタ形式からの変換の場合、出力桁位置の移動だけが有効です。入力桁位置の指定はできません。出力桁位置は、:10進数で指定します。式による指定もでき、そのなかでは、

- * ホスト側(出力)の末尾
- . ホスト側(出力)の現在の桁位置 という特殊変数が使えます。

たとえば、: \$ - 8 と指定すれば、

ホストレコード末尾の8バイト前に出力先を移動する という意味になります。

注意 ---- 先頭を0桁目とする

F*TRAN+では、レコードの先頭を0桁目として数えます。

(2)ポジション(デリミタ位置)の指定

変換対象にするWindows側(入力)データのデリミタ形式の項目位置を、別の任意の項目位置に移動できます。現在、処理対象にしている項目位置を、この機能で強制的に変更できます。この機能を利用すると、デリミタ形式の項目の組み替えなどが簡単に実現できます。

デリミタ位置は、 @@入力項目番号 で指定します。式による指定もでき、そのなかでは、

- . Windows側(入力)の現在の項目位置
- \$ Windows側(入力)の入力全体の項目数
- * Windows側(入力)の残りの項目数 という特殊変数が使えます。

たとえば、

@@.-2 と指定すれば、 今の入力項目から2項目戻る

@@\$ - 2 と指定すれば、 Windows側(入力)レコード末尾から2項目に位置する

という意味になります。

注意 ---- 先頭を 0 項目目とする

F*TRAN+では、レコードの先頭項目を0項目目として数えます。

(3)ポジション(入力スキップ)の指定

プリント形式の場合

Windows側(入力)レコードに不要な項目があるとき、それをスキップして変換できます。

スキップする幅は、 ^バイト数 で指定します。

たとえば、3 バイト分スキップしたいなら、 ^3 と指定します。

バイト数は省略でき、省略すると1バイトとみされるので、

^3 は ^^ と指定したのと同じです。

スキップする幅は式による指定もでき、そのなかでは、

* Windows側(入力)の残りバイト数 という特殊変数が使えます。

デリミタ形式の場合

デリミタ形式の場合には、出力幅に0を指定すると項目スキップできます。

(3)ポジション(出力スキップ)の指定

入力スキップ(プリント形式)とは逆に、ホスト側(出力)に何桁かスペースを作ることもできます。空項目を作るのがおもな用途です。

スキップする幅は、バイト数で指定します。

たとえば、3バイト分スキップしたいなら、 __3 と指定します。

バイト数は省略でき、省略すると1バイトとみされるので、

__3 は ____ と指定したのと同じです。

(4)15種類のデータ形式

[Ank(1)]

ANK項目を変換します。ホストが汎用機・オフコンの場合、どのコード変換が行われるかは、 変換設定のANKコードの設定で決まります(ふつう、セットアップ時に一回だけ行います)。

プリント形式の場合

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

* Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

出力幅は式による指定もでき、式のなかでは、

- \$ ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

項目長を縮めると、ANK項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、ホスト側のANK項目のおわりにスペース(20H/40H)が詰められます。

デリミタ形式の場合

ふつう、入力幅は省略し、出力幅だけを10進のバイト数で指定します。出力幅は式による指定もでき、式のなかでは

* ホスト側(出力)の残りバイト数 という特殊変数が使えます。

[漢字(2)]

漢字項目を変換します。どのコード変換が行われるかは、変換設定の漢字変換方式の設定で決まります(ふつう、セットアップ時に一回だけ行います)。

プリント形式の場合

入力幅は、10進のバイト数で指定します(漢字の文字数ではありません)。式による指定もでき、そのなかでは、

* Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更したいときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、漢字項目のおわりのほうが切り捨てられます(漢字の中央で切れることはありません)。逆に、項目長を伸ばすと、ホスト側の漢字項目のおわりに漢字変換方式に設定されている漢字スペースが詰められます。

出力幅は式による指定もでき、そのなかでは、

- \$ ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

デリミタ形式の場合

ふつう入力幅は省略し、出力幅だけを10進のバイト数で指定します。出力幅は式による指定 もでき、そのなかでは、

* ホスト側(出力)の残りバイト数 という特殊変数が使えます。

[Ank・漢字(3)]

ANK・漢字まじり項目を変換します。どのコード変換が行われるかは、変換設定の漢字変換方式の設定で決まります(ふつう、セットアップ時に一回だけ行います)。ホストが汎用機・オフコンの場合、変換後、漢字文字列の前後にはKI/KOが挿入されます。この、KI/KO挿入のタイミングも漢字変換方式の設定で決まります。そのため、オーバーフローの危険性があるので、出力幅の指定には注意しなければいけません。

プリント形式の場合

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

* Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

ふつうは、KI/KOが挿入されてデータ長が長くなる分を加算した出力幅を指定します。オーバーフローすると、ANK・漢字まじり項目のおわりのほうが切り捨てられます。ただし、漢字モードのままおわることはありません。また、KOが途中で切れることもありません。

出力幅は式による指定もでき、そのなかでは、

- * ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

デリミタ形式の場合

ふつう、入力幅は省略し、出力幅だけを10進のバイト数で指定します。出力幅は式による指定もでき、式のなかでは、

* ホスト側(出力)の残りバイト数 という特殊変数が使えます。

「数値(4)]

文字形式の数値項目どうしの変換をします。ホストが汎用機・オフコンの場合、あらかじめ、 変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしてお かなければいけません(ふつう、セットアップ時に一回だけ行います)。数値変換は、Ank項 目変換と後述の表示・ゾーン変換の中間的なものです。Ank項目変換と比較すると、

文字形式数値しか通さない 入力幅を省略すると15桁(バイト)とみなす 右詰めになる

などの点が異なります。

「文字形式数値しか通さない」というのは、具体的には、

+、-、0~9、ピリオド(.), E、e、D、d

しか変換しないで、これら以外の文字は捨ててしまうということです。たとえば、通貨記号(¥/\$)や位取りのコンマ(,)などは削除されるので、リストファイルから入力データファイルを作るときなどに役立ちます。

[表示・ゾーン(5)]

Windowsの文字形式数値項目を、ホストのCOBOLのゾーン形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

入力幅は、バイト数で指定します。省略すると15バイトとみなされるので、プリント形式からの変換の場合は、ふつうは明示的に桁数を指定します。デリミタ形式からの変換の場合は、数値項目が15バイトを超えることは少ないので、省略するほうがふつうです。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 . 4 という数字を 5 バイトの符号つきゾーン形式の項目に記録するとすれば、ピクチャは s 4 . 1 と指定します。なお、ピクチャは省略できません。

まとめると、プリント形式からの変換の場合は、

入力幅 = 8、ピクチャ = s 4 . 1 のような指定になり、

デリミタ形式からの変換の場合は、

入力幅=省略、ピクチャ=s4.1 のような指定になるのがふつうです。

ただし、入力幅が15バイトを超えるときは、

入力幅=20、ピクチャ=u15.2 のように、ダミーの入力幅を指定してください。

[表示・パック(6)]

Windowsの文字形式数値項目を、ホストのCOBOLのパック形式数値項目、BCD形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

入力幅は、バイト数で指定します。省略すると15バイトとみなされるので、プリント形式からの変換の場合は、ふつうは明示的に桁数を指定します。デリミタ形式からの変換の場合は、数値項目が15バイトを超えることは少ないので、省略するほうがふつうです。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字を3バイトの符号つきパック形式の項目に記録するとすれば、ピクチャは s4.1 と指定します。123.4 という数字を3バイトのBCD形式の項目に記録するとすれば、ピクチャは b5.1 と指定します。なお、ピクチャは省略できません。

まとめると、プリント形式からの変換の場合は、

入力幅 = 8、ピクチャ = s 4 . 1 のような指定になり、

デリミタ形式からの変換の場合は、

入力幅=省略、ピクチャ=s4.1 のような指定になるのがふつうです。

ただし、入力幅が15バイトを超えるときは、

入力幅 = 20、ピクチャ = u 15.2 のように、ダミーの入力幅を指定してください。

[バイナリ(7)]

バイナリ変換(無変換)は、「コード変換を一切しない」という変換方法です。通常、Winホストデータファイル変換では使用しません。

プリント形式の場合

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

* Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

出力幅は式による指定もでき、式のなかでは、

- \$ ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

項目長を縮めると、バイナリ項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、ホスト側のバイナリ項目のおわりに、ホストが汎用機・オフコンの場合はNUL(00H)が、ホストがUnix/Windowsの場合はスペース(20H)が詰められます。

デリミタ形式の場合

ふつう、入力幅は省略し、出力幅だけを10進のバイト数で指定します。出力幅は式による指定もでき、式のなかでは

* ホスト側(出力)の残りバイト数 という特殊変数が使えます。

$[\Delta - \forall - A (8)], [\Delta - \forall - B (9)]$

ユーザーA / B 変換は、利用者独自のバイト単位の変換処理が必要なときに、A N K 変換表ユーザー用A、ユーザー用 B を書き替えて利用します。ユーザーA / B 変換には、A n k 項目変換の説明がほとんどそのまま当てはまります。

プリント形式の場合

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

* Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、ユーザーA/B項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、Windows側のユーザーA/B項目のおわりにスペース(20H/40H)が詰められます。

出力幅は式による指定もでき、式のなかでは、

- \$ ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

デリミタ形式の場合

ふつう、入力幅は省略し、出力幅だけを10進のバイト数で指定します。出力幅は式による指定もでき、式のなかでは、

* ホスト側(出力)の残りバイト数 という特殊変数が使えます。

$[\dot{y}-\dot{y}-\dot{y}-\dot{y}$ (A)]

Windows COBOLのゾーン形式数値項目を、ホストのCOBOLのゾーン形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が5バイトの符号つきゾーン形式の項目に記録されているとすれば、 出力ピクチャは s4.1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[ゾーン・パック(C)]

Windows COBOLのゾーン形式数値項目を、ホストのCOBOLのパック形式数値項目、BCD形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、 出力ピクチャは s 4 . 1 と指定します。B C D 形式なら、 b 4 . 1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ=s4.1という指定は、入力ピクチャ=s4.1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[年設定(E)]

日付データ項目を変換する際の、年の 2 桁(y y) と 4 桁(y y y y) の交換方式を設定します。W n n、または、S n nの形式で設定します。W n n の " W " はウインドウ方式を、<math>S n nの " S " はシフト方式を意味し、" n n " は 0 0 ~ 9 9 の数字で指定します。

入力幅に W30 と指定すれば、入力データの年を1930~2029年とみなし、 出力幅に S25 と指定すれば、出力データの年下2桁を、-25します。

また、シフト方式("Snn"指定)では、つぎの特殊指定ができます。

S S h o w a は、" S 2 5 " の指定と同じ(昭和通年方式) S H e i s e i は、" S 8 8 " の指定と同じ(平成通年方式)

年設定は日付項目変換が実行された時に適用になり、複数の年設定がなされている場合は、日付項目変換の直前の年設定が有効になります。

年設定がない場合の日付項目変換のデフォルトは、入力幅/出力幅ともに、 W30 となり、 入出力データの年を1930~2029年とみなします。

[区切記号(F)]

日付データ項目を出力する際の日付区切り記号をつぎの3つの中から設定します。入力幅に、SLASH、または、HYPHEN、または、PERIODと入力します。

指定文字	日付区切り記号	_ デ ー タ 例
SLASH	/ (スラッシュ)	1998/12/31
HYPHEN	- (ハイフン)	1998-12-31
PERIOD	.(ピリオド)	1998.12.31

日付区切り設定は日付項目変換が実行された時に適用になり、複数の日付区切り設定がなされている場合は、日付項目変換の直前の日付区切り設定が有効になります。

日付区切り設定がない場合の日付項目変換のデフォルトは、 SLASH となり、日付区切り記号を"/"にします。

[日付(G)]

日付データ項目を変換します。コード変換は、変換設定のANKコードの設定で決まります(ふつう、セットアップ時に一回だけ行います)。

入力幅と出力幅は、必ず日付マスクの形式で指定し、省略はできません。日付マスクの指定方法については、「操作の基礎」の章で説明しました。たとえば、

入力幅に、 yyyy-mm-dd

出力幅に、 yymmdd のように指定すると、

コード変換後に、入力側 1 0 バイトの日付データ項目を、出力側 6 バイトの日付データ項目に編集します。その際に、年設定、日付区切り設定が適用になります。

[ゾーン・表示(I)]

Windows COBOLのゾーン形式数値項目を、ホストの文字形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。変換結果は右詰めになります。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、ピクチャは s 4 . 1 と指定します。なお、ピクチャは省略できません。

出力幅は省略できます。出力幅を省略すると、

符号つきなら1、符号なしなら0とする

- + 整数部桁数
- + 1 + 小数部桁数(小数部があれば)

の要領でピクチャから自動的に計算された値が使われます。たとえば、ピクチャ = s 4 . 1、 出力幅 = 省略という指定は、出力幅に7を指定したのと同じ意味になります。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

[表示・2進(K)]

Windowsの文字形式数値項目を、ホストの2進形式整数・小数項目に変換します。

入力幅は、バイト数で指定します。省略すると15バイトとみなされるので、プリント形式からの変換の場合は、ふつうは明示的に桁数を指定します。デリミタ形式からの変換の場合は、数値項目が15バイトを超えることは少ないので、省略するほうがふつうです。

2 進キャスト/ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字を4バイトの符号つき2進形式の項目に記録するとすれば、2 進キャスト/ピクチャは i4s4.1 と指定します。なお、2 進キャストは省略できません。まとめると、プリント形式からの変換の場合は

入力幅 = 10、 2 進キャスト / ピクチャ = i 4 s 4 . 1 のような指定になり、デリミタ形式からの変換の場合は

入力幅=省略、2 進キャスト / ピクチャ= i 4 s 4 . 1 のような指定になるのがふつうです。ただし、入力幅が 1 5 バイトを超えるときは、

入力幅 = 20、2 進キャスト / ピクチャ = 18u15.2 のように、ダミーの入力幅を指定します。

[ゾーン・2進(M)]

WindowsのCOBOLのゾーン形式数値項目を、ホストの2進形式整数・小数項目に変換します。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が5バイトの符号つきゾーン形式の項目に記録されているとすれば、 入力ピクチャは s4.1 と指定します。

2 進キャスト/ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字を4バイトの符号つき2進形式の項目に記録するとすれば、2 進キャスト/ピクチャは i4s4.1 と指定します。なお、2進キャストは省略できません。 入力ピクチャは省略できます。入力ピクチャを省略すると、「2進キャスト/ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、2進キャスト/ピクチャ=i4s4.1という指定は、入力ピクチャ=s4.1を指定したのと同じ意味になります。

(6)引用符はずし/くくりの指定

デリミタ形式からの変換の場合には、項目が引用符 (") でくくられていることがあります。 その引用符は無視します。また、引用符でくくったなかにデリミタがあっても、ただの文字データとして扱います。

ホストがUnix、Widniwsの場合、デリミタ形式に変換するときは、変換後の文字列を引用符でくくることができます。ただし、プリント形式への変換のときは、この引用符くくりの機能は無効になります。

市販ソフトの入力用には、文字項目だけを引用符でくくることが多いのですが、引用符を使わないソフトもあれば、すべての項目を引用符でくくるソフトもあります。

参考 ...

たとえば、デリミタ形式のうちK3フォーマットと呼ばれるものは、

各項目はコンマで区切る 数値はそのまま(位取りのコンマ不可) 文字列は引用符でくくる

というルールになっています。

(7) 漢字イン/アウトの指定

ホストが汎用機・オフコンの場合、項目の前後に、漢字イン(KI)・漢字アウト(KO)を 挿入することができます。

これは、漢字項目に漢字イン(KI)・漢字アウト(KO)を指定するのが、ふつうの使い方です。この指定とAnk・漢字まじり項目変換は、似ているところもありますが別物です。ご注意ください。

1:漢字イン、3:漢字イン+アウトを入力した場合、__10進数数値(0~255の範囲) を入力することもできます。東芝方式のANK・漢字項目化するために「長さバイト」を付加する場合に指定します。

(8)改行コード挿入の指定

ホストがUnix、Windowsの場合、任意のところに改行コードを挿入する(改行コードで項目を区切る)こともできます。改行コード挿入には、つぎの2種類があります。

通常 出力がプリント形式データファイルの時は、指定が無効になる 強制 出力がプリント形式データファイルの時も、指定が有効になる

出力がデータファイルのときに、自動的にレコード末尾に改行コードが付加される機能とは別のものですから、混同しないでください。

<マップ設定(詳細)ウインドウ>



マップ設定の記述をするエディットボックスです。

 \sim はファイル(\underline{F}) 編集(\underline{E})のメニューの機能がボタン化されています。 既存のパラメータファイル等を読みます。

の内容をパラメータファイルへ保存します。

直前の編集作業が無効になり、元に戻ります。

選択した文字列が切り取られ、カットバッファに入ります。

選択した文字列がカットバッファに入ります。

/ の操作でカットバッファに入った内容を、カーソルがある位置へ貼りつけます。 カーソルがある位置以降に文字列があれば、挿入になります。

選択した文字列が削除されます。

の文字列をすべて選択状態にします。

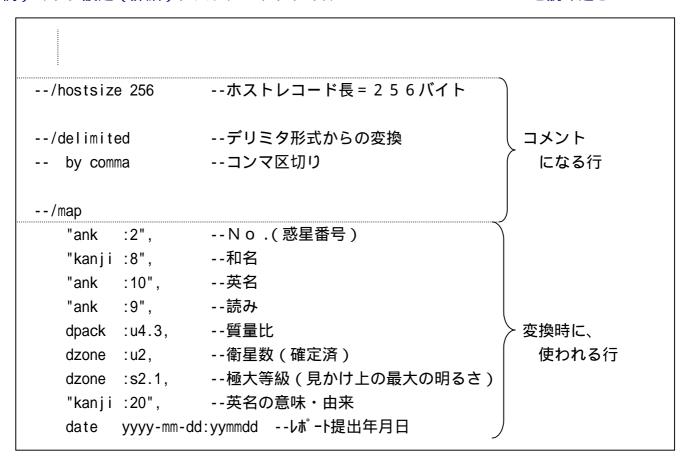
設定が完了したら、OKボタンをクリックします。

キャンセルボタンをクリックすると、すべての入力 / 編集作業が無効になります。

注意 ---- マップ設定(詳細)の記述内容について

マップ設定(詳細)のエディットボックスの中に記述する内容は、レコードレイアウト(/mapを含まない/map以降)のみの記述となります。既存のパラメータファイルを読み込むと、それ以外の記述(/delimited~、/map等)は自動的にコメントになります。記述に誤りがある状態でOKボタンをクリックすると、エラーとなる場合があります。修正してから再度OKボタンをクリックしてください。/map以降の文法については、「コマンド編」の「コマンド型の実行」の章を参照してください。

例)マップ設定(詳細)にパラメータファイル"PNPUTCSV.P"を読み込む



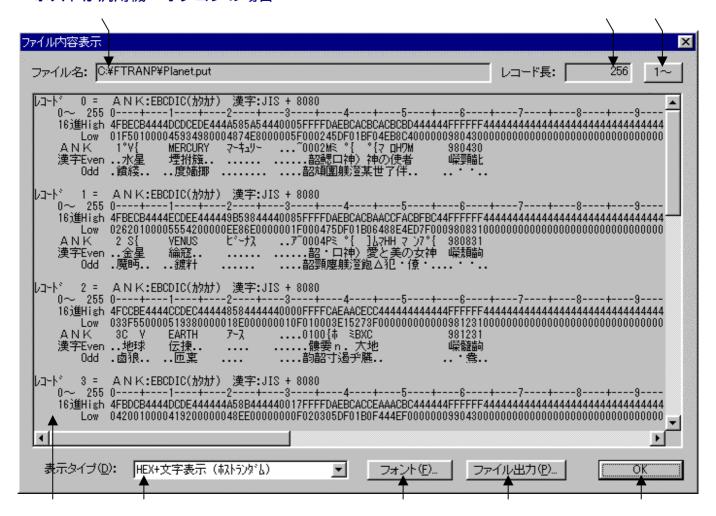
Win ホストデータファイル変換の実行

変換ボタンをクリックすると、Win ホストデータファイル変換が始まります。

ファイル内容表示ウインドウ

変換結果表示(<u>V</u>)のチェックボックスをONにし、Win ホストファイル変換を実行すると、実行ウインドウを閉じた後につぎのファイル内容表示ウインドウが開き、変換先のファイルの中身を確認することができます。

<ホストが汎用機・オフコンの場合>



変換先のファイル名が表示されます。

変換先のファイルのレコード長が表示されます。

変換先のファイルの中身が、

レコード長単位のHEX+文字表示(ホスト)形式で出力されます。

の表示タイプは、HEX+文字表示(ホスト)固定です。

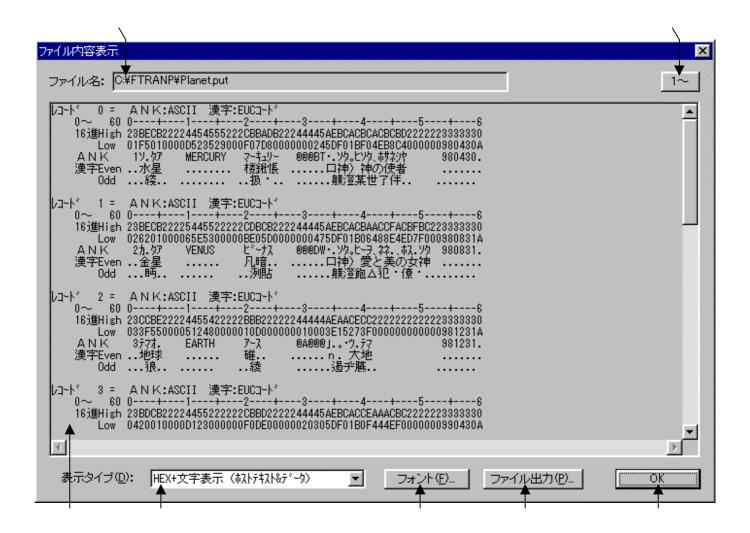
フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力 (P) ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

<ホストがUnixで、出力ファイル形式がデータファイル(可変長)の場合>



変換先のファイル名が表示されます。

変換先のファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、改行コード単位のHEX+文字表示(ホストテキスト&データ)です。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

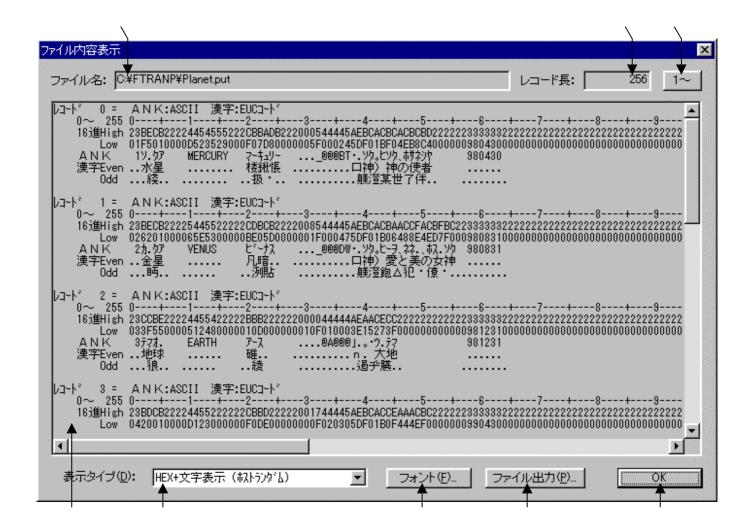
フォント(\underline{F})ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力 (\underline{P}) ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

<ホストがUnixで、出力ファイル形式がランダムファイル(固定長)の場合>



変換先のファイル名が表示されます。

変換先のファイルのレコード長が表示されます。

変換先のファイルの中身が、

レコード長単位のHEX+文字表示(ホストランダム)形式で出力されます。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

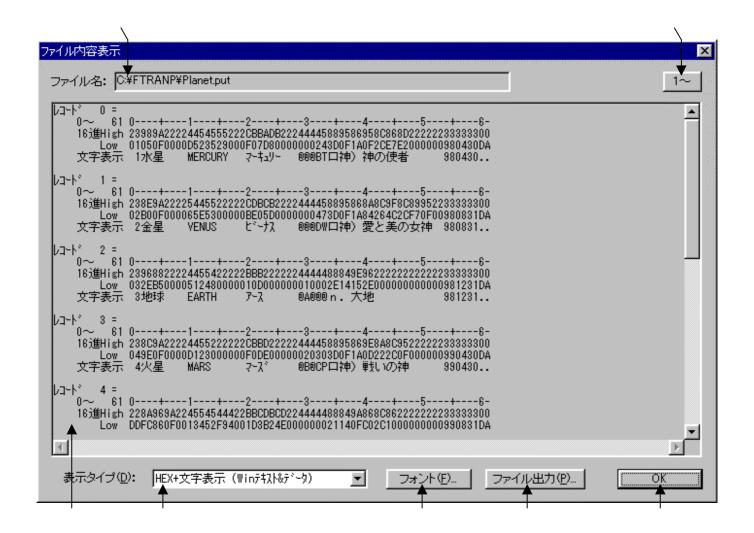
フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

<ホストがWindowsで、出力ファイル形式がデータファイル(可変長)の場合>



変換先のファイル名が表示されます。

変換先のファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、改行コード単位のHEX+文字表示(Winテキスト&データ)です。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

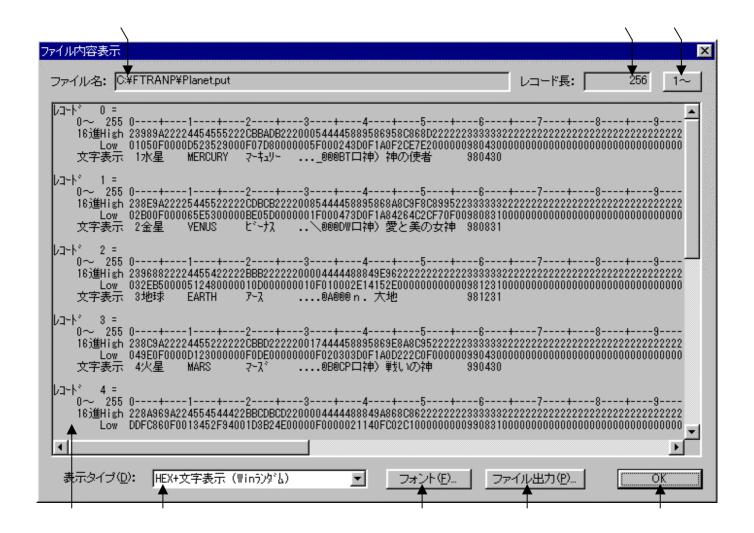
フォント(\underline{F})ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

<ホストがwindowsで、出力ファイル形式がランダムファイル(固定長)の場合>



変換先のファイル名が表示されます。

変換先のファイルのレコード長が表示されます。

変換先のファイルの中身が、

レコード長単位のHEX+文字表示(Winランダム)形式で出力されます。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

注意事項

漢字があるときの注意

漢字項目変換やAnk・漢字まじり項目変換を行うときは、あらかじめ

変換設定の漢字変換方式設定で、適当な漢字変換方式を割り当てておく

のを忘れないでください。また、入力幅、出力幅は漢字データについても

バイト単位で指定

します。漢字の文字数ではないことに注意してください。

改行コードがないと、1件だけの変換になる

Windowsファイルの各レコードの末尾に改行コードがついていないと、先頭の1件だけ 変換されます。この場合は、Win ホストランダムファイル変換を使うべきです。

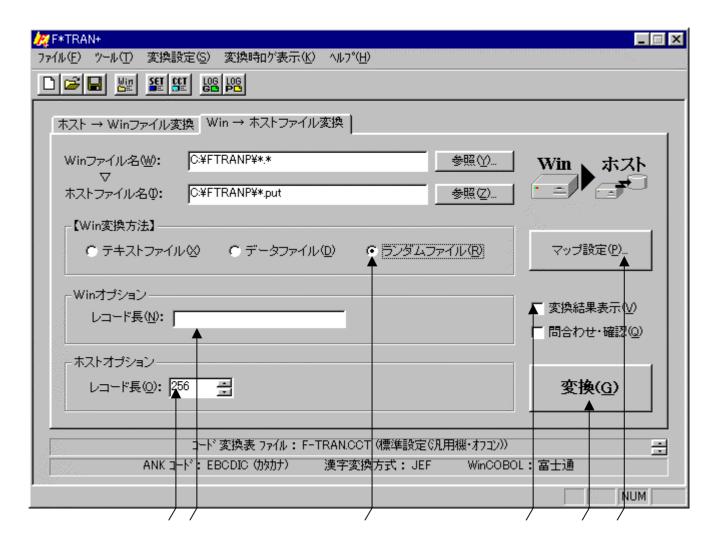
その他の注意事項

「Win ホストファイル変換」の節を参照してください。

4.9 Win ホストランダムファイル変換

Win ホストランダムファイル変換の操作方法について説明します。

ホストが汎用機・オフコンの場合



変換方法のランダムファイル(R)をクリックし、

Win ホストランダムファイル変換を選択します。

入力レコード長を指定するオプションです。

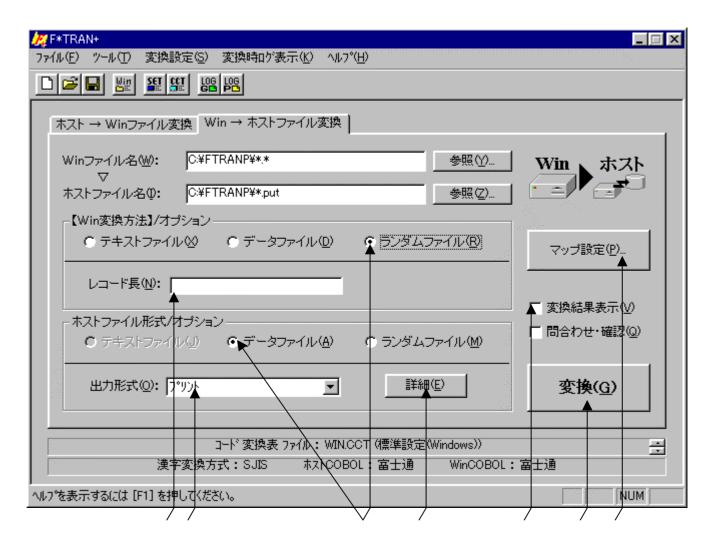
ホストファイルのレコード長を指定します。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。

ホストがUnix、Windowsの場合



Win変換方法のランダムファイル (\underline{R})をクリックし、Win ホストランダムファイル 変換を選択します。ホストファイル形式は、データファイル (\underline{A})、ランダムファイル (\underline{M}) の選択ができますが、上の例はホストファイル形式がデータファイル (\underline{A}) です。

入力レコード長を指定するオプションです。

出力ファイル形式を指定するオプションです。

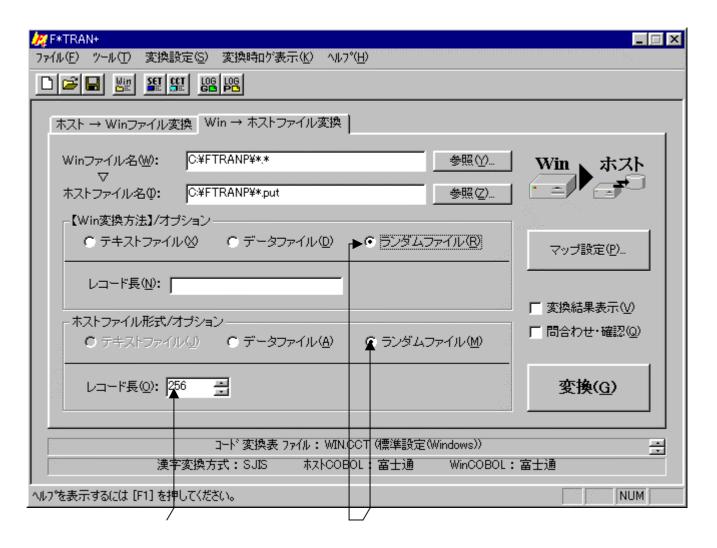
ホストファイルの詳細オプションボタンです。EOF付加の選択ができます。

ホストがWindowsの場合のみ有効です。ホストがUnixの場合はありません。

マップ設定(レコードレイアウト設定)をするオプションボタンです。

変換後に、変換先のファイルの中身を表示ウインドウに出力するチェックボックスです。

変換を実行するボタンです。このボタンをクリックすると、変換が始まります。



Win変換方法のランダムファイル (\underline{R})をクリックし、Win ホストランダムファイル 変換を選択します。ホストファイル形式は、データファイル (\underline{A})、ランダムファイル (\underline{M}) の選択ができますが、上の例はホストファイル形式がランダムファイル (\underline{M}) です。ホストファイルのレコード長を指定します。

Win ホストファイル変換の共通指定項目は、「Win ホストファイル変換」の節ですでに詳しく説明しました。そちらを参照してください。

オプションの指定

入力レコード長オプション

Windows側(入力)のレコード長を指定します。Windowsレコード長は、通常、10進数で指定します。式による指定もでき、そのなかでは、

\$ ホスト側(出力)のレコード長 という特殊変数が使用できます。

ホストファイルのレコード長を基準にして、Windowsファイルのレコード長を決めることができます。 たとえば、

\$ + 2 と指定すれば、

「ホストファイルのレコード長に + 2 したものをWindows 側のレコード長にせよ」という意味になります。

入力レコード長オプションでWindowsレコード長を指定できることは、Win ホストランダムファイル変換の性格をよく反映しています。

Win ホストランダムファイル変換は、おもにWindows COBOLの順ファイルやBASICのランダムファイルを変換することをねらったものである、ということはすでに述べたとおりです。そして、BASIC側の事情を考慮すると、レコード長を任意に指定できないと困ることが多いのです。

とくに困るのは、N88BASICコンパイラ/インタプリタを使うときです。

N88BASICコンパイラ / インタプリタでは、1つのプログラム内で複数のレコード長のランダムファイルを扱うことができません。

もう1つの理由は、ゾーン形式とパック形式の変換、日付データの変換をサポートしているため、レコード長を増減する機能がないと困るからです。また、不要な項目がいくつもあるときや、 予備領域が多すぎるときはレコード長を縮めたいものです。

注意 ---- Windows自身には「レコード長」の概念がない

Windows自体にはファイルごとに登録されたレコード長というものはありません。そのため、かなり自由が利きます。アプリケーションがファイルを扱う論理的な単位を、ここではWindowsレコード長と呼んでいます。

ホストファイル・レコード長指定 <ホストファイルが固定長の場合のみ>

ホストファイルのレコード長を1~32767の範囲の10進数で指定します。デフォルトは、256バイトです。レコード長の指定が間違っていると正しいデータ変換が行われませんので、この指定は極めて重要です。

出力形式オプション 〈ホストファイルが可変長の場合のみ〉

データとしてのテキストファイルにはいくつもの形式があるので、どの形式にするかを指定します。 つぎの、 7 種類の中から選択します。

プリント ------------ プリント形式に変換する デリミタ (コンマ ・ 圧縮) ---- デリミタ形式 (コンマ) に変換、圧縮あり デリミタ (タブ ・ 圧縮) ---- デリミタ形式 (タブ) に変換、圧縮あり デリミタ (スペース・ 圧縮) ---- デリミタ形式 (スペース) に変換、圧縮あり デリミタ (コンマ ・非圧縮) ---- デリミタ形式 (コンマ) に変換、圧縮なし デリミタ (スペース・非圧縮) ---- デリミタ形式 (タブ) に変換、圧縮なし デリミタ (スペース・非圧縮) ---- デリミタ形式 (スペース) に変換、圧縮なし

プリントを指定すると、デリミタ(区切り文字)なしで変換され、プリント形式(固定長のテキストファイル〔SDF形式〕)になります。これがデフォルトです。

デリミタを指定すると、デリミタ形式(区切り文字つきのテキストファイル)に変換されます。 デリミタの種類によってさらに細かい形式が決まります。デリミタは、

コンマ区切り形式(CSV形式、K3形式) タブ区切り形式(TAB=09H) スペース区切り形式(SP=20H) の3つのなかから指定できます。

通常は、コンマ区切り(CSV)形式への変換が一般的です。さらに、

変換後に圧縮をかける(可変長になる。デフォルト) 変換後に圧縮しない(固定長のまま) を指定できます。

を指定できます。圧縮をかけると本来不要なスペース、つまり、

レコードの先頭・末尾のスペース デリミタの前後のスペース 引用符の前のスペース が削除されます。 252 圧縮をかける機能があるのは、つぎのような理由があるためです。

変換後のWindowsファイルの容量を、大幅に減らすことができる 一部の市販ソフトでは、不要なスペースがあるとうまくデータを読み込めない

EOF付加指定 <ホストがWindowsで、可変長ファイルの場合のみ>

EOFコード(1AH)の扱いを指定します。

EOF付加のチェックボックスを、

ONにすると、EOFコードをつける指定 OFFにすると、EOFコードをつけない指定 となります。

EOF付加指定がONであれば、WindowsファイルのおわりにEOFコードをつけます。 現在では少なくなりましたが、テキストファイルのおわりにEOFコードがついていないとエラーにするソフトがあります。その場合にも対処するための機能です。

EOF付加指定がOFFであれば、EOFコードはつけません。これがふつうだと思ってください。こちらがデフォルトです。

マップ設定オプション

このマップ設定オプションで、項目別の細かい変換方法を指示します。本来なら、自動的に項目を認識して変換ができると便利です。しかし、ホストの、とくにCOBOLのデータには、データ自身に桁数や小数点位置の判断に必要な情報が含まれていない、という特性があります。そのため自動変換は原理的に不可能なのです。

マップ設定のボタンをクリックすると、つぎのウインドウが開きます。



マップ設定には、

簡易設定 ---- 設定ウインドウに従って、必要な値を入力するだけで設定ができます。

通常は、こちらの機能だけで間に合う場合がほとんどです。

詳細設定 ---- フリーフォーマットで、マップ設定を記述するエディタモードです。

簡易設定だけでは設定できない特殊な設定をする場合、あるいは、 既存のパラメータファイル等を流用したい場合等に使用します。

マップ設定の文の書き方を覚えなくては設定できません。

の2とおりの方法があります。

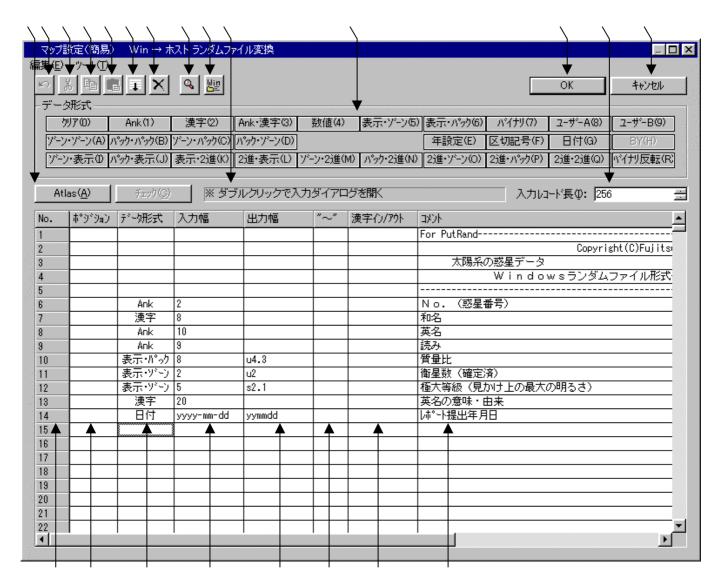
注意 ---- 簡易設定と詳細設定の組み合わせ

簡易設定で設定した値は、詳細設定で読み込むことができるテキスト形式に直されます。よって、簡易設定をした後の項目に、詳細設定で特殊設定をつけ足すこともできます。

注意 ---- 詳細設定 簡易設定の順で設定した場合の制約

詳細設定 簡易設定の順で設定した場合は、原形が変わってしまう場合があります。これは、 詳細設定で設定した項目が簡易設定で読み込まれることにより、スリム化されてしまうからです。 ただし、原形が変わってしまっても、設定内容が変わるわけではありません。

<マップ設定(簡易)ウインドウ>



No.は、1~2000です。ここは非入力項目で、行選択の時にクリックします。項目の桁位置等を入力します。通常、項目を組み替えるとき以外は入力しません。 ダブルクリックすると、入力サブウインドウが開きます。

項目のデータ形式を24種類の中から指定します。

ダブルクリックすると、入力サブウインドウが開きます。

項目の入力幅を入力します。

項目の出力幅を入力します。出力幅=入力幅の場合は省略します。

データ形式がゾーン / パック / 2 進指定の場合、ピクチャまたは 2 進ピクチャで指定します。ホストがUnix、Windowsの場合、デリミタ形式に変換するときに項目を引用符でくくる指定ができます。

ホストが汎用機・オフコンの場合は、項目の前後に漢字イン / アウトを挿入する指定です。ホストがUnix、Windowsの場合は、改行コードを挿入する指定です。コメントの入力ができます。

セルポインタの入力ガイダンスメッセージが出力されるフィールドです。 誤入力をした場合のエラーメッセージも出力されます。

 \sim は編集(\underline{E})、ツール(\underline{T})のメニューの機能がボタン化されています。 直前の編集作業が無効になり、元に戻ります。

1 行または複数行を選択(の操作)をした後に、このボタンをクリックすると、 選択した項目の内容が切り取られ、カットバッファに入ります。

1 行または複数行を選択(の操作)をした後に、このボタンをクリックすると、 選択した項目の内容がカットバッファに入ります。

カットバッファに入った内容(、 の操作)を、セルポインタがある行へ貼りつけます。 すでに、セルポインタ以降の行に設定項目がある場合は、挿入になります。

セルポインタがある行以降が1行づつ下がります。通常は、行挿入をする場合に使用します。 選択した項目の内容が削除されます。

変換プレビュー(変換前/変換後のデータ確認)ができます。

Winファイルエディタが起動します。

Windowsファイルの中身を見る(編集も可能)ことができます。

セルポインタが選択項目にある場合に、選択肢のボタンが表示されるエリアです。

入力レコード長オプションの指定を変更できます。

設定が完了したら、OKボタンをクリックします。

キャンセルボタンをクリックすると、すべての入力 / 編集作業が無効になります。

マルチレコードの設定ができるAtlasモードに切り替えるボタンです。 詳細は、マルチレコード編のマニュアルを参照してください。

注意 ---- マップ未設定時のデータ変換動作

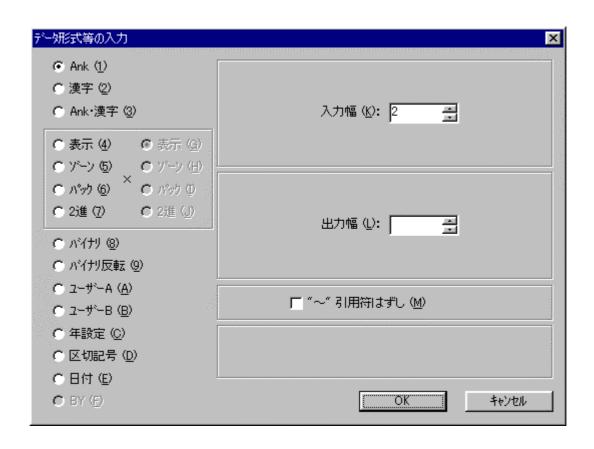
マップ未設定の場合は、すべてをANKデータとみなしてデータ変換を行います。

<マップ設定(簡易)サブウインドウ>

ポジションの欄をダブルクリックすると、つぎのサブウインドウが開きます。入出力桁位置や 入出力スキップの設定を簡便に行うことができます。

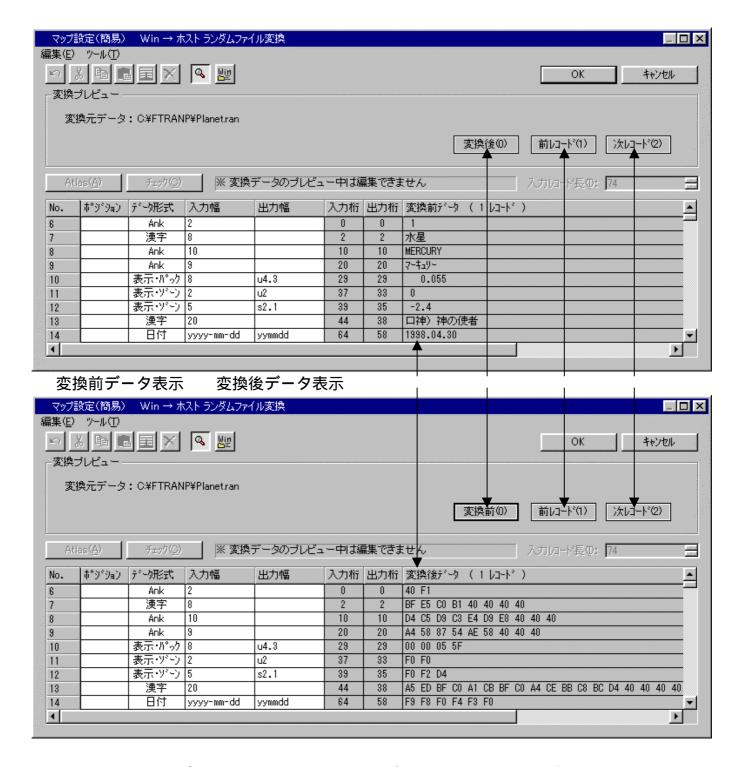


データ形式の欄をダブルクリックすると、つぎのサブウインドウが開きます。データ形式から 引用符はずし(漢字項目は漢字イン/アウト)までの設定を簡便に行うことができます。



<マップ設定(簡易)・変換プレビュー機能>

変換プレビューボタンをクリックすると、マップ設定(簡易)のウインドウはつぎのような状態になり、変換前/変換後のデータを確認することができます。



変換前 / 変換後のデータ表示の切り替えを行うボタンです。 の表示が変わります。 前レコード / 次レコード表示のためのボタンです。

ここには、入力レコード/出力レコードのポジション(桁)が表示されます。

データ表示は、変換前が文字、変換後が16進、未定義個所が文字になります。

(1)ポジション(桁位置)の指定

変換対象にするWindows側(入力)の桁位置や、変換結果を書き込むホスト側(出力)の桁位置を、別の任意の位置に移動できます。現在、処理対象にしている桁位置を、この指定で強制的に変更できます。この機能を利用すると、項目の組み替えなどが簡単に実現できます。

入力桁位置は、ふつう10進数で指定します。式による指定もでき、そのなかでは、

- \$ Windows側(入力)のレコード長
- . Windows側(入力)の現在の桁位置 という特殊変数が使えます。

たとえば、

. - 40と指定すれば、今の入力桁位置から40バイト戻る

という意味になります。

出力桁位置を移動することもできます。ふつう、

:10進数

で桁位置を指定します。10進数の代わりに式による指定もでき、そのなかでは、

- \$ ホスト側(出力)のレコード長
- . ホスト側(出力)の現在の桁位置 という特殊変数が使えます。

たとえば、

: . - 40 と指定すれば、 現在の出力桁位置から40バイトバックせよ

という意味になります。

注意 ---- 先頭を0桁目とする

F * T R A N + では、レコードの先頭を 0 桁目として数えます。

(2)ポジション(入力スキップ)の指定

Windows側(入力)レコードに不要な項目があるとき、それをスキップして変換できます。

スキップする幅は、 ^ バイト数 で指定します。

たとえば、3 バイト分スキップしたいなら、 ^3 と指定します。

バイト数は省略でき、省略すると1バイトとみされるので、

^3 は ^^ と指定したのと同じです。

スキップする幅は式による指定もでき、そのなかでは、

* Windows側(入力)の残りバイト数 という特殊変数が使えます。

(3)ポジション(出力スキップ)の指定

入力スキップとは逆に、ホスト側(出力)に何桁か空きを作ることもできます。

スキップする幅は、 __バイト数 で指定します。

たとえば、3バイト分スキップしたいなら、 __3 と指定します。

バイト数は省略でき、省略すると1バイトとみされるので、

__3 は ____ と指定したのと同じです。

スキップする幅は式による指定もでき、そのなかでは、

* ホスト側(出力)の残りバイト数 という特殊変数が使えます。

(4)24種類のデータ形式

[Ank(1)]

ANK項目を変換します。ホストが汎用機・オフコンの場合、どのコード変換が行われるかは、 変換設定のANKコードの設定で決まります(ふつう、セットアップ時に一回だけ行います)。 入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- \$ Windows側(入力)のレコード長
- * Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、ANK項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、ホスト側のANK項目のおわりにスペース(20H/40H)が詰められます。

出力幅は式による指定もでき、そのなかでは、

- \$ ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

「漢字(2)]

漢字項目を変換します。どのコード変換が行われるかは、変換設定の漢字変換方式の設定で決まります(ふつう、セットアップ時に一回だけ行います)。

入力幅は、10進のバイト数で指定します(漢字の文字数ではありません)。式による指定もでき、そのなかでは、

- \$ Windows側(入力)のレコード長
- * Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、漢字項目のおわりのほうが切り捨てられます(漢字の中央で切れることはありません)。逆に、項目長を伸ばすと、ホスト側の漢字項目のおわりに漢字変換方式に設定されている漢字スペースが詰められます。

出力幅は式による指定もでき、そのなかでは、

- \$ ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

[Ank・漢字(3)]

ANK・漢字まじり項目を変換します。どのコード変換が行われるかは、変換設定の漢字変換方式の設定で決まります(ふつう、セットアップ時に一回だけ行います)。ホストが汎用機・オフコンの場合、変換後、漢字文字列の前後にはKI/KOが挿入されます。この、KI/KO挿入のタイミングも漢字変換方式の設定できまります。そのため、オーバーフローの危険性があるので、出力幅の指定には注意しなければいけません。

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- \$ Windows側(入力)のレコード長
- * Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

ふつうは、KI/KOが挿入されてデータ長が長くなる分を加算した出力幅を指定します。オーバーフローすると、ANK・漢字まじり項目のおわりのほうが切り捨てられます。ただし、漢字モードのままおわることはありません。また、KOが途中で切れることもありません。

出力幅は式による指定もでき、そのなかでは、

- \$ ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

「数値(4)]

文字形式の数値項目どうしの変換をします。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。数値変換は、Ank項目変換と後述の表示・ゾーン変換の中間的なものです。Ank項目変換と比較すると、

文字形式数値しか通さない 入力幅を省略すると15桁(バイト)とみなす 右詰めになる

などの点が異なります。

「文字形式数値しか通さない」というのは、具体的には、

+、-、0~9、ピリオド(.), E、e、D、d

しか変換しないで、これら以外の文字は捨ててしまうということです。たとえば、通貨記号 (¥/\$)や位取りのコンマ(,)などは削除されるので、リストファイルから入力データファイルを作るときなどに役立ちます。

[表示・ゾーン(5)]

Windowsの文字形式数値項目を、ホストのCOBOLのゾーン形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

入力幅は、バイト数で指定します。省略すると15バイトとみなすので、ふつうは明示的にバイト数を指定します。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 . 4 という数字が 5 バイトの符号つきゾーン形式の項目に記録するとすれば、ピクチャは s 4 . 1 と指定します。なお、ピクチャは省略できません。

[表示・パック(6)]

Windowsの文字形式数値項目を、ホストのCOBOLのパック形式数値項目、BCD形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

入力幅は、バイト数で指定します。省略すると15バイトとみなすので、ふつうは明示的にバイト数を指定します。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字を3バイトの符号つきパック形式の項目に記録するとすれば、ピクチャは s4.1 と指定します。123.4 という数字を3バイトのBCD形式の項目に記録するとすれば、ピクチャは b5.1 と指定します。なお、ピクチャは省略できません。

[バイナリ(7)]

バイナリ変換(無変換)は、「コード変換を一切しない」という変換方法です。 入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- \$ Windows側(入力)のレコード長
- * Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなされます。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、バイナリ項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、ホスト側のバイナリ項目のおわりに、ホストが汎用機・オフコンの場合はNUL(00H)が、ホストがUnix/Windowsの場合はスペース(20H)が詰められます。

出力幅は式による指定もでき、そのなかでは、

- \$ ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

ユーザーA / B 変換は、利用者独自のバイト単位の変換処理が必要なときに、A N K 変換表ユーザー用A、ユーザー用 B を書き替えて利用します。ユーザーA / B 変換には、A n k 項目変換の説明がほとんどそのまま当てはまります。

入力幅は、10進のバイト数で指定します。式による指定もでき、そのなかでは、

- \$ Windows側(入力)のレコード長
- * Windows側(入力)の残りバイト数 という特殊変数が使えます。

入力幅を省略すると*を指定したとみなします。

出力幅も省略できます。出力幅を省略すると「入力幅と同じ」とみなします。

項目長を変更するときは、出力幅を10進のバイト数で指定します。

項目長を縮めると、ユーザーA/B項目のおわりのほうが切り捨てられます。逆に、項目長を伸ばすと、ホスト側のユーザーA/B項目のおわりにNUL(00H)が詰められます。

出力幅は式による指定もでき、そのなかでは、

- \$ ホスト側(出力)のレコード長
- * ホスト側(出力)の残りバイト数 という特殊変数が使えます。

$[\dot{y}-\dot{y}-\dot{y}-\dot{y}$ (A)]

Windows COBOLのゾーン形式数値項目を、ホストのCOBOLのゾーン形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が5バイトの符号つきゾーン形式の項目に記録されているとすれば、 出力ピクチャは s4.1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

$[\mathcal{N} \cup \mathcal{N}$

Windows COBOLのパック形式数値項目を、ホストのCOBOLのパック形式数値項目、BCD形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が3バイトの符号つきパック形式の項目に記録されているとすれば、 出力ピクチャは s4.1 と指定します。BCD形式なら b5.1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[ゾーン・パック(C)]

Windows COBOLのゾーン形式数値項目を、ホストのCOBOLのパック形式数値項目、BCD形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、 出力ピクチャは s 4 . 1 と指定します。B C D 形式なら b 5 . 1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[パック・ゾーン(D)]

Windows COBOLのパック形式数値項目を、ホストのCOBOLのパック形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字が3バイトの符号つきパック形式の項目に記録されているとすれば、 出力ピクチャは s4.1 と指定します。

入力ピクチャは省略できます。入力ピクチャを省略すると、「出力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、出力ピクチャ= s 4 . 1 という指定は、入力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

出力ピクチャは省略できます。出力ピクチャを省略すると、「入力ピクチャ」と同じとみなされます。たとえば、入力ピクチャ = s 4 . 1、出力ピクチャ = 省略という指定は、出力ピクチャ = s 4 . 1を指定したのと同じ意味になります。

なお、入力ピクチャと出力ピクチャを同時に省略することはできません。

[年設定(E)]

日付データ項目を変換する際の、年の 2 桁(y y) と 4 桁(y y y y) の交換方式を設定します。W n n、または、S n nの形式で設定します。W n n の " W " はウインドウ方式を、<math>S n nの " S " はシフト方式を意味し、" n n " は 0 0 ~ 9 9 の数字で指定します。

入力幅に W30 と指定すれば、入力データの年を1930~2029年とみなし、 出力幅に S25 と指定すれば、出力データの年下2桁を、-25します。

また、シフト方式("Snn"指定)では、つぎの特殊指定ができます。

S S h o w a は、" S 2 5 " の指定と同じ(昭和通年方式) S H e i s e i は、" S 8 8 " の指定と同じ(平成通年方式)

年設定は日付項目変換が実行された時に適用になり、複数の年設定がなされている場合は、日 付項目変換の直前の年設定が有効になります。

年設定がない場合の日付項目変換のデフォルトは、入力幅/出力幅ともに、 W30 となり、 入出力データの年を1930~2029年とみなします。

[区切記号(F)]

日付データ項目を出力する際の日付区切り記号をつぎの3つの中から設定します。入力幅に、SLASH、または、HYPHEN、または、PERIODと入力します。

指定文字	日付区切り記号	_ デ ー タ 例
SLASH	/ (スラッシュ)	1998/12/31
HYPHEN	- (ハイフン)	1998-12-31
PERIOD	.(ピリオド)	1998.12.31

日付区切り設定は日付項目変換が実行された時に適用になり、複数の日付区切り設定がなされている場合は、日付項目変換の直前の日付区切り設定が有効になります。

日付区切り設定がない場合の日付項目変換のデフォルトは、 SLASH となり、日付区切り記号を"/"にします。

[日付(G)]

日付データ項目を変換します。コード変換は、変換設定のANKコード設定で決まります(ふつう、セットアップ時に一回だけ行います)。

入力幅と出力幅は、必ず日付マスクの形式で指定し、省略はできません。日付マスクの指定方法については、「操作の基礎」の章で説明しました。たとえば、

入力幅に、 yyyy-mm-dd

出力幅に、 yymmdd のように指定すると、

コード変換後に、入力側 1 0 バイトの日付データ項目を、出力側 6 バイトの日付データ項目に編集します。その際に、年設定、日付区切り設定が適用になります。

[ゾーン・表示(I)]

Windows COBOLのゾーン形式数値項目を、ホストの文字形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。変換結果は右詰めになります。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、ピクチャは s 4 . 1 と指定します。なお、ピクチャは省略できません。

出力幅は省略できます。出力幅を省略すると、

符号つきなら1、符号なしなら0とする

- +整数部桁数
- + 1 + 小数部桁数(小数部があれば)

の要領でピクチャから自動的に計算された値が使われます。たとえば、ピクチャ = s 4 . 1、 出力幅 = 省略という指定は、出力幅に7を指定したのと同じ意味になります。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

[パック・表示(J)]

ホストのCOBOLのパック形式数値項目を、文字形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。変換結果は右詰めになります。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 3 バイトの符号つきパック形式の項目に記録されているとすれば、 ピクチャは s 4 . 1 と指定します。なお、ピクチャは省略できません。

パック形式では、整数部桁数 + 小数部桁数を奇数にしておくのが通例です。整数部の最上位桁に意味があるのかないのかは、半々の割合です。出力幅は省略できます。出力幅を省略すると、

符号つきなら1、符号なしなら0とする

- + 整数部桁数
- + 1 + 小数部桁数(小数部があれば)

の要領でピクチャから自動的に計算された値が使われます。例を示すと、ピクチャ = s 4 . 1、 出力幅 = 省略という指定は、出力幅に7を指定したのと同じ意味になります。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。 オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

[表示・2進(K)]

Windowsの文字形式数値項目を、ホストの2進形式整数・小数項目に変換します。

入力幅は、バイト数で指定します。省略すると15バイトとみなされるので、プリント形式からの変換の場合は、ふつうは明示的に桁数を指定します。デリミタ形式からの変換の場合は、数値項目が15バイトを超えることは少ないので、省略するほうがふつうです。

2 進キャスト/ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 1 2 3 . 4 という数字を 4 バイトの符号つき 2 進形式の項目に記録するとすれば、 2 進キャスト/ピクチャは i 4 s 4 . 1 と指定します。なお、2 進キャストは省略できません。

[2進・表示(L)]

Windowsの2進形式整数・小数項目を、ホストの文字形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。変換結果は右詰めになります。

2 進キャスト / ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字が 4 バイトの符号つき 2 進形式の項目に記録されているとすれば、2 進キャスト / ピクチャは i 4 s 4.1 と指定します。なお、2 進キャストは省略できません。

出力幅は省略できます。出力幅を省略すると、

入力幅	出力幅	
1	3	
2	5	
3	8	
4	1 0	
5	1 3	左記の値を基本として、
6	1 5	符号つきなら + 1、ピクチャ指定の小数部があれば + 1とし、
7	1 7	さらに、ピクチャ指定のほうが大きければ、
8	1 8	整数部桁数に、符号つきなら + 1 、小数部があれば + 1 + 小数点桁数

の要領で2進キャスト/ピクチャから自動的に計算された値が使われます。例を示すと、入力幅=i4s4.1、出力幅=省略という指定は、出力幅=10という指定と同じです。

出力幅を明示的に指定するときは、オーバーフローに注意しながら10進のバイト数で指定します。 オーバーフローすると、符号や上位桁が切り捨てられるので、注意してください。

[ゾーン・2進(M)]

WindowsのCOBOLのゾーン形式数値項目を、ホストの2進形式整数・小数項目に変換します。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 5 バイトの符号つきゾーン形式の項目に記録されているとすれば、 入力ピクチャは s 4 . 1 と指定します。

2進キャスト/ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字を4バイトの符号つき2進形式の項目に記録するとすれば、2進キャスト/ピクチャは i4s4.1 と指定します。なお、2進キャストは省略できません。入力ピクチャは省略できます。入力ピクチャを省略すると、「2進キャスト/ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、2進キャスト/ピクチャ=i4s4.1という指定は、入力ピクチャ=s4.1を指定したのと同じ意味になります。

[パック・2進(N)]

Windows COBOLのパック形式数値項目を、ホストの2進形式整数・小数項目に変換します。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 .4 という数字が 3 バイトの符号つきパック形式の項目に記録されているとすれば、 入力ピクチャは s 4 . 1 と指定します。

2進キャスト/ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字を4バイトの符号つき2進形式の項目に記録するとすれば、2進キャスト/ピクチャは i4s4.1 と指定します。なお、2進キャストは省略できません。入力ピクチャは省略できます。入力ピクチャを省略すると、「2進キャスト/ピクチャ」と同じとみなされます。たとえば、入力ピクチャ=省略、2進キャスト/ピクチャ=i4s4.1という指定は、入力ピクチャ=s4.1を指定したのと同じ意味になります。

[2進・ゾーン(0)]

Windowsの2進形式整数・小数項目を、ホストのCOBOLのゾーン形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

2 進キャスト / ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字が 4 バイトの符号つき 2 進形式の項目に記録されているとすれば、2 進キャスト / ピクチャは i 4 s 4.1 と指定します。なお、2 進キャストは省略できません。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 1 2 3 . 4 という数字を 5 バイトの符号つきゾーン形式の項目に記録するとすれば、ピクチャは s 4 . 1 と指定します。

出力ピクチャは省略できます。出力ピクチャを省略すると、「2進キャスト/ピクチャ」と同じとみなされます。たとえば、2進キャスト/ピクチャ=i4s4.1、出力ピクチャ=省略という指定は、出力ピクチャ=s4.1を指定したのと同じ意味になります。

[2進・パック(P)]

Windowsの2進形式整数・小数項目を、ホストのCOBOLのパック形式数値項目に変換します。ホストが汎用機・オフコンの場合、あらかじめ、変換設定のANKコードの設定で、EBCDIC系かJIS8/ASCII系かの設定をしておかなければいけません(ふつう、セットアップ時に一回だけ行います)。

2 進キャスト/ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 123.4 という数字が4バイトの符号つき2進形式の項目に記録されているとすれば、2 進キャスト/ピクチャは i4s4.1 と指定します。なお、2 進キャストは省略できません。

ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、

- 123.4 という数字を3バイトの符号つきパック形式の項目に記録するとすれば、ピクチャは s4.1 と指定します。

出力ピクチャは省略できます。出力ピクチャを省略すると、「2進キャスト/ピクチャ」と同じとみなされます。たとえば、2進キャスト/ピクチャ=i4s4.1、出力ピクチャ=省略という指定は、出力ピクチャ=s4.1を指定したのと同じ意味になります。

[2進·2進(Q)]

Windowsの2進形式整数・小数項目を、ホストの2進形式整数・小数項目項目に変換します。

2 進キャスト / ピクチャの指定方法については、「操作の基礎」の章で説明しました。たとえば、 - 1 2 3 . 4 という数字を 4 バイトの符号つき 2 進形式の項目とする場合、2 進キャスト / ピクチャは i 4 s 4 . 1 と指定します。

2 進キャスト / ピクチャは入力または出力のどちらかを省略できます。入力を省略すると「出力と同じ」とみなされ、出力を省略すると「入力と同じ」とみなされます。なお、入力 2 進キャスト / ピクチャと出力 2 進キャスト / ピクチャを同時に省略することはできません。

[<u>バイナリ反転(R)</u>]

バイナリ反転は、「コード変換を一切せずに、幅分のデータをバイト単位で左右反転する」という変換方法です。2進数値データ(整数、小数、実数)は、ホスト側が正順であるのに対して、Windows側が逆順であることが多いので、2進数値データの内容をそのままバイト単位で左右反転する場合等に使用します。2進・2進変換のような加工機能はありませんが、その分だけ処理が高速です。

幅は、10進のバイト数で指定します。たとえば4バイトの入力データが、16進表現で、01 AB CD EF であれば、出力データは EF CD AB 01 になります。

(5)引用符くくりの指定

ホストがUnix、Widniwsの場合、デリミタ形式に変換するときは、変換後の文字列を引用符でくくることができます。ただし、プリント形式への変換のときは、この引用符くくりの機能は無効になります。

市販ソフトの入力用には、文字項目だけを引用符でくくることが多いのですが、引用符を使わないソフトもあれば、すべての項目を引用符でくくるソフトもあります。

参考 ...

たとえば、デリミタ形式のうちK3フォーマットと呼ばれるものは、

各項目はコンマで区切る 数値はそのまま(位取りのコンマ不可) 文字列は引用符でくくる

というルールになっています。

(6) 漢字イン/アウトの指定

ホストが汎用機・オフコンの場合、項目の前後に、漢字イン(KI)・漢字アウト(KO)を 挿入することができます。

これは、漢字項目に漢字イン(KI)・漢字アウト(KO)を指定するのが、ふつうの使い方です。この指定とAnk・漢字まじり項目変換は、似ているところもありますが別物です。ご注意ください。

1:漢字イン、3:漢字イン+アウトを入力した場合、__10進数数値(0~255の範囲)を入力することもできます。東芝方式のANK・漢字項目化するために「長さバイト」を付加する場合に指定します。

(7)改行コード挿入の指定

ホストがU n i x、W i n d o w s の場合、任意のところに改行コードを挿入する(改行コードで項目を区切る)こともできます。改行コード挿入には、つぎの 2 種類があります。

通常 出力がプリント形式データファイルの時は、指定が無効になる 強制 出力がプリント形式データファイルの時も、指定が有効になる

出力がデータファイルのときに、自動的にレコード末尾に改行コードが付加される機能とは別のものですから、混同しないでください。

<マップ設定(詳細)ウインドウ>



マップ設定の記述をするエディットボックスです。

 \sim はファイル(\underline{F}) 編集(\underline{E})のメニューの機能がボタン化されています。 既存のパラメータファイル等を読みます。

の内容をパラメータファイルへ保存します。

直前の編集作業が無効になり、元に戻ります。

選択した文字列が切り取られ、カットバッファに入ります。

選択した文字列がカットバッファに入ります。

/ の操作でカットバッファに入った内容を、カーソルがある位置へ貼りつけます。 カーソルがある位置以降に文字列があれば、挿入になります。

選択した文字列が削除されます。

の文字列をすべて選択状態にします。

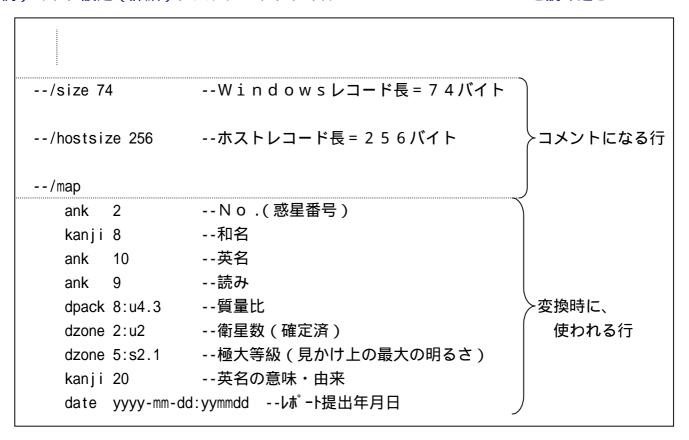
設定が完了したら、OKボタンをクリックします。

キャンセルボタンをクリックすると、すべての入力 / 編集作業が無効になります。

注意 ---- マップ設定(詳細)の記述内容について

マップ設定(詳細)のエディットボックスの中に記述する内容は、レコードレイアウト(/mapを含まない/map以降)のみの記述となります。既存のパラメータファイルを読み込むと、それ以外の記述(/delimited~、/map等)は自動的にコメントになります。記述に誤りがある状態でOKボタンをクリックすると、エラーとなる場合があります。修正してから再度OKボタンをクリックしてください。/map以降の文法については、「コマンド編」の「コマンド型の実行」の章を参照してください。

例)マップ設定(詳細)にパラメータファイル"PNPUTRND.P"を読み込む



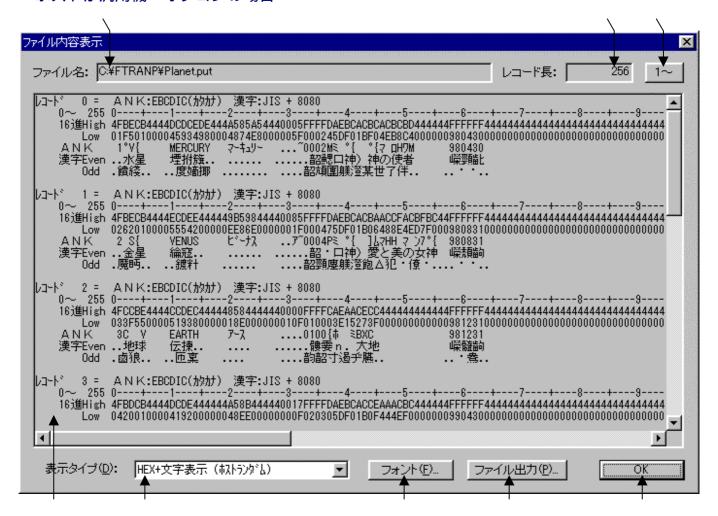
Win ホストランダムファイル変換の実行

変換ボタンをクリックすると、Winホストランダムファイル変換が始まります。

ファイル内容表示ウインドウ

変換結果表示(<u>V</u>)のチェックボックスをONにし、Win ホストファイル変換を実行すると、実行ウインドウを閉じた後につぎのファイル内容表示ウインドウが開き、変換先のファイルの中身を確認することができます。

<ホストが汎用機・オフコンの場合>



変換先のファイル名が表示されます。

変換先のファイルのレコード長が表示されます。

変換先のファイルの中身が、

レコード長単位のHEX+文字表示(ホストランダム)形式で出力されます。

の表示タイプは、HEX+文字表示(ホストランダム)固定です。

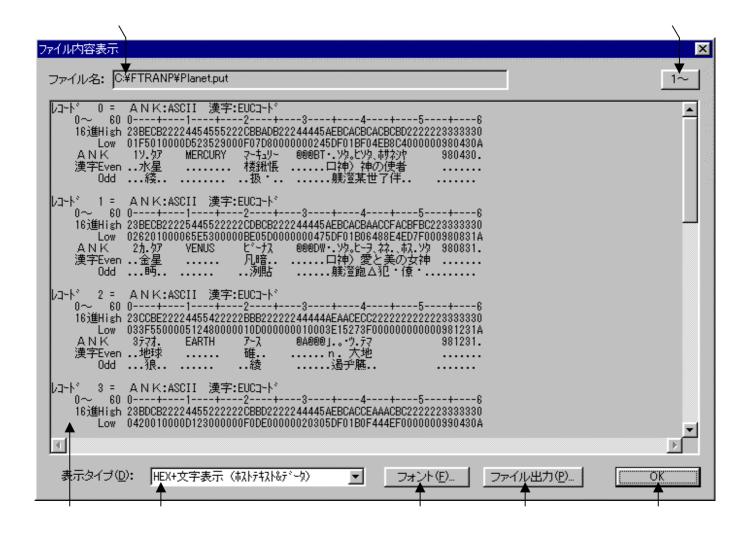
フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(<u>P</u>)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

<ホストがUnixで、出力ファイル形式がデータファイル(可変長)の場合>



変換先のファイル名が表示されます。

変換先のファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、改行コード単位のHEX+文字表示(ホストテキスト&データ)です。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

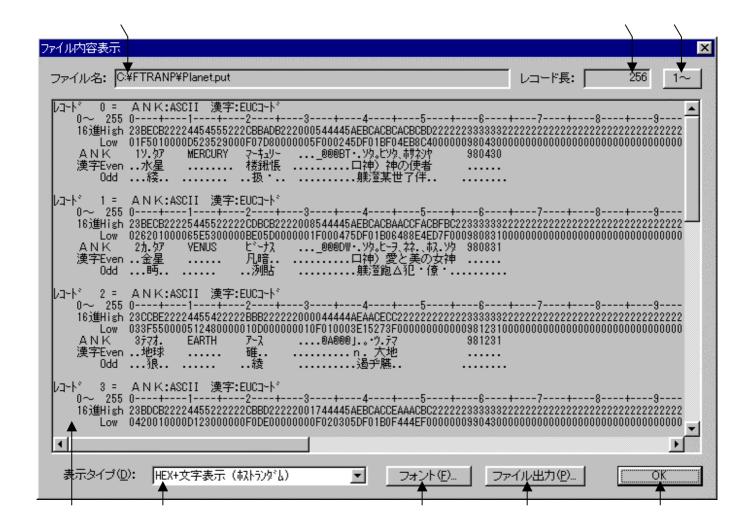
フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(\underline{P})ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

<ホストがUnixで、出力ファイル形式がランダムファイル(固定長)の場合>



変換先のファイル名が表示されます。

変換先のファイルのレコード長が表示されます。

変換先のファイルの中身が、

レコード長単位のHEX+文字表示(ホストランダム)形式で出力されます。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

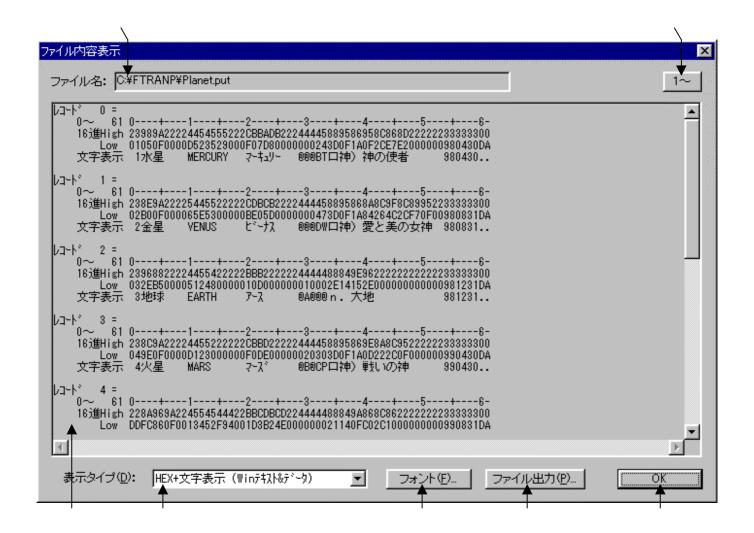
フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

<ホストがWindowsで、出力ファイル形式がデータファイル(可変長)の場合>



変換先のファイル名が表示されます。

変換先のファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、改行コード単位のHEX+文字表示(Winテキスト&データ)です。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

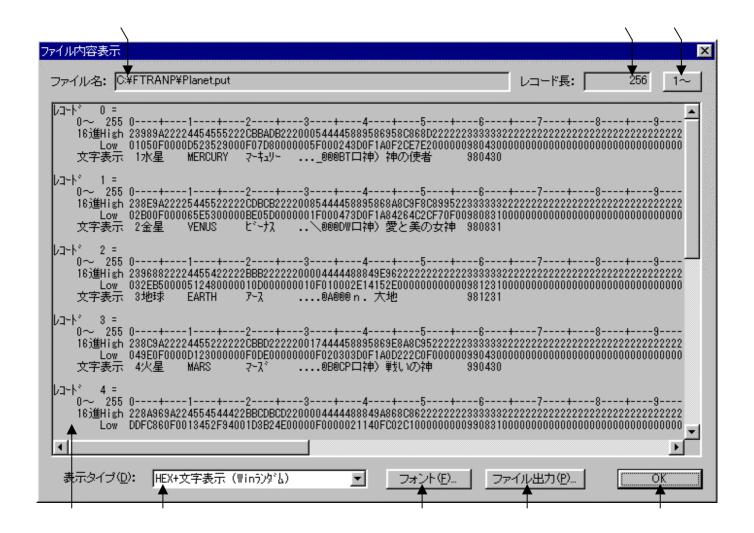
フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

<ホストがwindowsで、出力ファイル形式がランダムファイル(固定長)の場合>



変換先のファイル名が表示されます。

変換先のファイルのレコード長が表示されます。

変換先のファイルの中身が、

レコード長単位のHEX+文字表示(Winランダム)形式で出力されます。

の表示タイプをHEX+文字表示、文字表示のどちらかに切り替えます。

フォント(F)ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える:デフォルト)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

注意事項

漢字があるときの注意

漢字項目変換やAnk・漢字まじり項目変換を行うときは、あらかじめ

変換設定の漢字変換方式設定で、適当な漢字変換方式を割り当てておく

のを忘れないでください。また、入力幅、出力幅は漢字データについても

バイト単位で指定

します。漢字の文字数ではないことに注意してください。

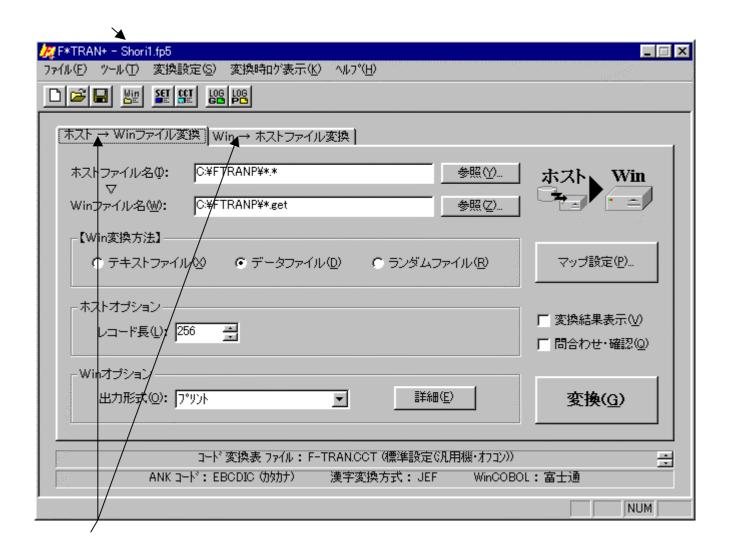
その他の注意事項

「Win ホストファイル変換」の節を参照してください。

4.10 スクリプトファイル

ホスト Winファイル変換、ホスト ホストファイル変換の2つのメイン処理部で設定した 項目は、F*TRAN+のスクリプトファイルとして保存することができます。

作成されたスクリプトファイルは、エクスプローラからクリックするだけで実行できます。スクリプトファイルの実行については、解説で説明します。



読み込まれたスクリプトファイル名。 選択されているメイン処理部が、保存の対象となる。

「新規にスクリプトファイルに保存する場合]

のタブコントロールをクリックし、処理(ホスト Winファイル変換、Win ホストファイル変換のいずれか)を選択します。必要な項目を設定し、実行が可能なら、実際に処理を行い結果を確認します。

この状態の設定を保存するなら、メニューバーのファイル(\underline{F}) 名前を付けて保存(\underline{A})を実行します。ここで、つぎのウインドウが開きます。



これは、スクリプトファイルを実行する時の設定をするウインドウです。 スクリプトファイルを実行する必要がない場合は、どの設定でもかまいません。

実行ウインドウ表示をON、実行ウインドウ自動クローズをOFFで、 実行ウインドウを表示し、処理後、実行ウインドウ開いたまま待つ

実行ウインドウ表示をON、実行ウインドウ自動クローズをONで、 実行ウインドウを表示し、処理後、自動的に実行ウインドウ閉じて終了する

実行ウインドウ表示をOFFで、

実行ウインドウを表示しないで処理を実行し、終了する

の3とおりの設定ができます。

実行終了時の待ち時間を秒数で設定することもできます。通常は0秒のままで設定しますが、 データ変換の対象となるファイルのサイズが小さすぎてアプリケーションからスクリプトファ イルを実行した後の戻り値が取得できない場合は、数秒の値を設定します。実行ウインドウ等の タイトルを変更する場合は、タイトルのエディットボックスに出力する文字列を入力します。

設定が完了したら、OKボタンをクリックします。

つぎに、"名前を付けて保存"のウインドウが開いたら、ファイル名を入力し、保存(<u>S</u>)ボタンをクリックして、スクリプトファイルへの保存がなされます。

「保存したスクリプトファイルを読み込む場合 1

メニューバーのファイル (\underline{F}) 開く (\underline{O}) を実行します。"ファイルを開く"のウインドウが開いたら、ファイル名を選択し、開く (\underline{O}) ボタンをクリックします。

スクリプトファイルが読み込まれると、 のタブコントロールが切り替わり、メイン処理部の内容が保存した時の状態になります。 のタイトルバーにスクリプトファイル名が表示されます。

[変更した設定をスクリプトファイルに書き込む場合]

スクリプトファイルを読み込みます。変更する設定があれば、修正します。

メニューバーのファイル(F) 上書き保存(S)を実行します。

"名前を付けて保存"のところで説明した保存設定のウインドウが開きます。必要があれば変更してOKボタンをクリックすると、スクリプトファイルへの上書き保存がなされます。

[今までの設定を無効にし、新規に設定をはじめる場合]

メニューバーのファイル (\underline{F}) 新規作成 (\underline{N}) を実行します。実行すると、"新規に作成しますか?"と確認ウインドウが開きます。はい (\underline{Y}) ボタンをクリックすれば、メイン処理部の設定が起動時の状態になります。いいえ (\underline{N}) ボタンをクリックすれば、新規作成を中止できます。

以上が、ファイル・メニューの基本的な操作です。

解説 - スクリプトファイルの実行

F*TRAN+のファイルメニューの機能を使って作成(保存)されたファイル(拡張子が". FP5"のファイル)は、つぎの方法で実行することができます。

コマンドプロンプトから下記の書式を入力し、実行する。 (または、バッチファイルの中に記述し、実行する)

[START /W[AIT]] FP FP5 ファイル名.FP5 [-H ホストファイル名] [-W Win ファイル名]

通常は、

C: ¥ > fp shori1.fp5

のように入力し、実行します。

バッチファイルの中で組み合わせて使用する場合は、START/W を頭に付けます。

エクスプローラ等からFP5ファイルをダブルクリックし、実行する。

F * T R A N + を正常にインストールしてある環境では、エクスプローラ等から F P 5 ファイルをダブルクリックするだけで、実行することができます。

注意 ---- FP5ファイルが実行できる範囲

FP5ファイルが実行できる範囲はつぎの通りです。

ホスト Winファイル変換 Win ホストファイル変換

のいずれか1処理です(組み合わせ等はできません)。

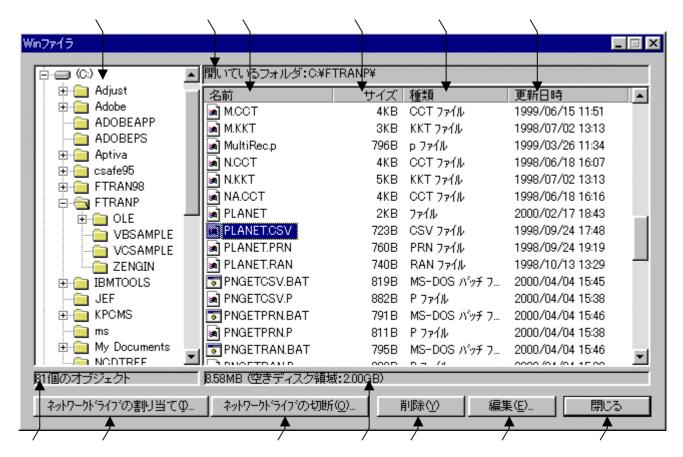
また、下記の条件の FP5 ファイルを実行しようとすると、正常な実行ができません。 F*TRAN+のメインウインドウが起動し、 <math>FP5 ファイルを開いた状態になります。 不足している項目を確定させてから実行してください。

- ・ホストファイル名が指定されていない場合
- ・Winファイル名が指定されていない場合(Win ホストファイル変換のみ)

- Hや - Wを使ってファイル名を指定する場合は、F P 5 ファイル作成時にファイル名の欄を 省略してください。F P 5 ファイルにファイル名が指定されている場合は、そちらが優先されま す。

4.11 Winファイラ

Win ファイラは、Windowsのファイルを管理する機能を持っています。メニューバーのツール(T) Winファイラ(W)を実行すると、つぎのウインドウが開きます。



全ドライブのフォルダ情報が表示されるエリアです。

開いているフォルダ名が表示されます。

開いているフォルダ内のフォルダ名とファイル名が表示されます。

開いているフォルダ内のフォルダとファイルのサイズが表示されます。

開いているフォルダ内のフォルダとファイルの種類が表示されます。

開いているフォルダ内のフォルダとファイルの更新日時が表示されます。

開いているフォルダ内のオブジェクトの個数が表示されます。

開いているフォルダ内のオブジェクトの総容量と空きディスク容量が表示されます。

ネットワークドライブの割り当てをするウインドウを開くボタンです。

ネットワークドライブの切断をするウインドウを開くボタンです。

のフォルダまたはファイルを選択し、削除(Y)ボタンをクリックすると、

選択したフォルダまたはファイルを削除することができます。

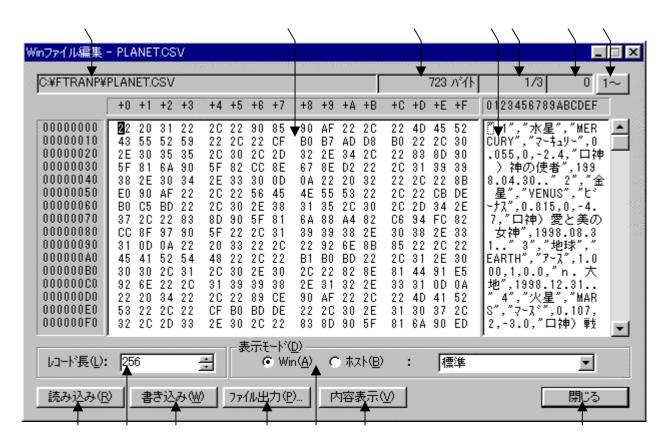
のファイルを選択し、編集(E)ボタンをクリックすると、

Winファイルエディタが開きます。

閉じるボタンをクリックすると、Winファイラを終了し、ウインドウを閉じます。

Winファイル編集(Winファイルエディタ)

Windowsファイル(ホストファイルを含む)の中身を16進で表示し、編集できます。



ファイル名が表示されます。

ファイルサイズが表示されます。

データのレコード番号 / 全体のレコード件数が表示されます。

レコード内の桁位置(先頭は0または1)が表示されます。

桁位置を数える起点を、0起点、1起点のどちらかに切り替えることができます。

ファイルの中身を16進表現で表示するエリアです。0~Fを入力できます。

ファイルの中身を文字表現で表示するエリアです。文字を入力できます。

レコード長を1~32767で指定できます。デフォルトは256です。

文字表示エリアの表示モードを、Win(標準、ASCII(漢字表現なし)) ホストが 汎用機・オフコンの場合は、ホスト(JIS8/ASCII、EBCDIC (かか)、EBCDIC (英小文字)) ホストがUnixの場合は、ホスト(標準、ASCII(漢字表現なし))の中から選択します。

読み込み(R)ボタンをクリックすると、データをディスクから読み直します。

書き込み(W)ボタンをクリックすると、データをディスクに書き戻します。

ファイル出力(P)ボタンをクリックすると、表示されている画面からのデータ内容を、

ファイルに出力するウインドウが開きます。出力するレコード数とファイル名を指定して、

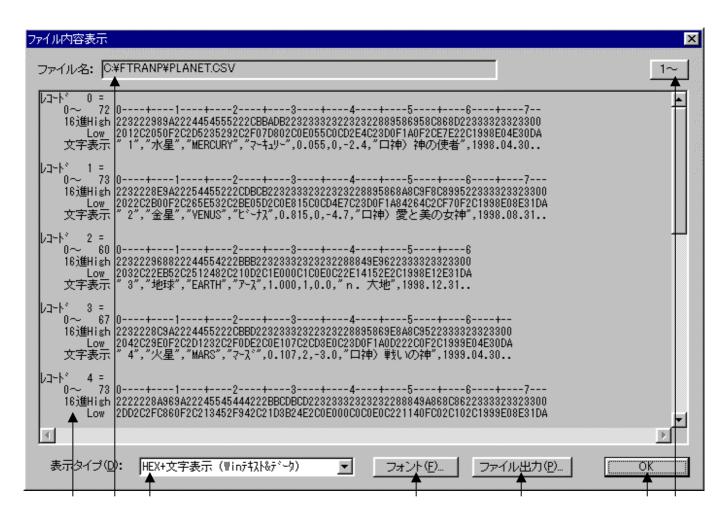
保存(S)ボタンをクリックすると、ファイルに出力されます。

内容表示(V)ボタンをクリックすると、ファイル内容表示ウインドウが開きます。

閉じるボタンをクリックすると、Winファイルエディタを終了します。

ファイル内容表示ウインドウ

Winファイルエディタの表示モード(\underline{D})がWin(\underline{A})の状態で、内容表示(\underline{V})ボタンをクリックすると、つぎのファイル内容表示ウインドウが開きます。



Windowsファイル名が表示されます。

Windowsファイルの中身が、で選択されている表示タイプで出力されます。

デフォルトは、改行コード単位のHEX+文字表示(Winテキスト&データ)です。

の表示タイプを4~6種類の中から切り替えることができます。

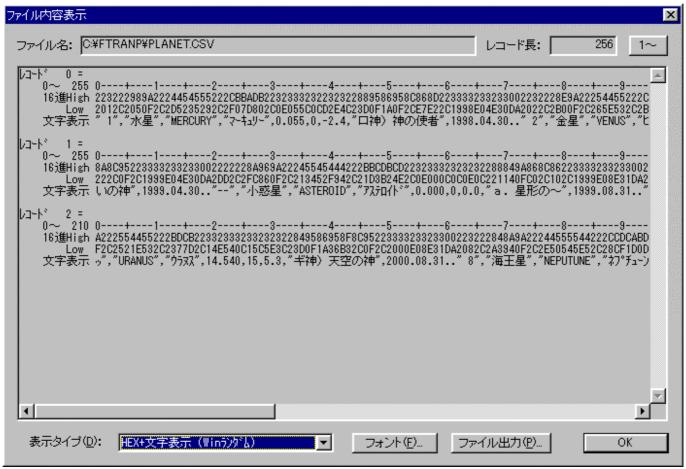
フォント(\underline{F})ボタンをクリックして、 の出力文字フォントを切り替えることができます。

ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える)

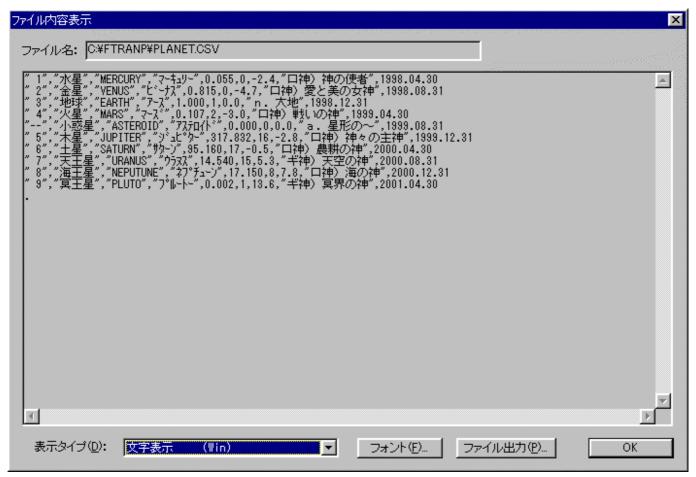
1起点(先頭を1として数える)のどちらかに切り替えることができます。

OKボタンをクリックすると、ファイル内容表示ウインドウを閉じます。



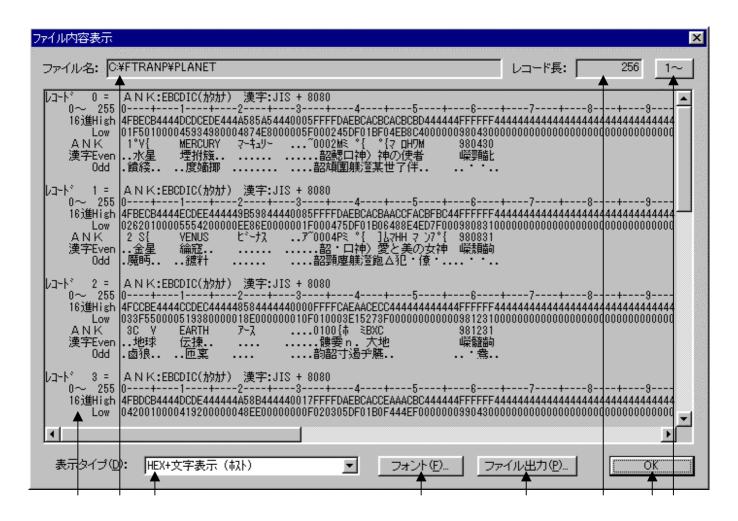
< H E X + 文字表示(Winランダム)>

<文字表示(Win)>



Winファイルエディタの表示モード(\underline{D})がホスト(\underline{B})の状態で、内容表示(\underline{V})ボタンをクリックすると、つぎのファイル内容表示ウインドウが開きます。

<ホストが汎用機・オフコンの場合>



ホストファイル名が表示されます。

ホストファイルのレコード長が表示されます。

ホストファイルの中身が、レコード長単位のHEX+文字表示(ホスト)形式で出力されます。 の表示タイプを4種類の中から切り替えることができます。

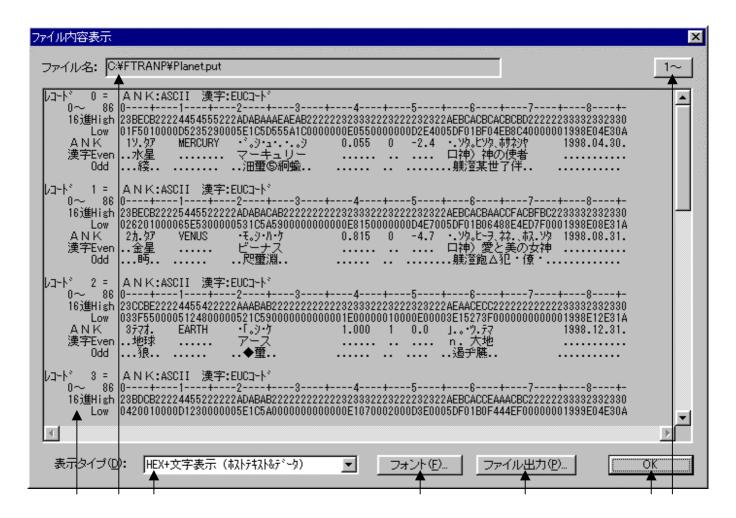
フォント(\underline{F})ボタンをクリックして、 の出力文字フォントを切り替えることができます。 ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

OKボタンをクリックすると、ファイル内容表示ウインドウを閉じます。

<ホストがUnixの場合>



ホストファイル名が表示されます。

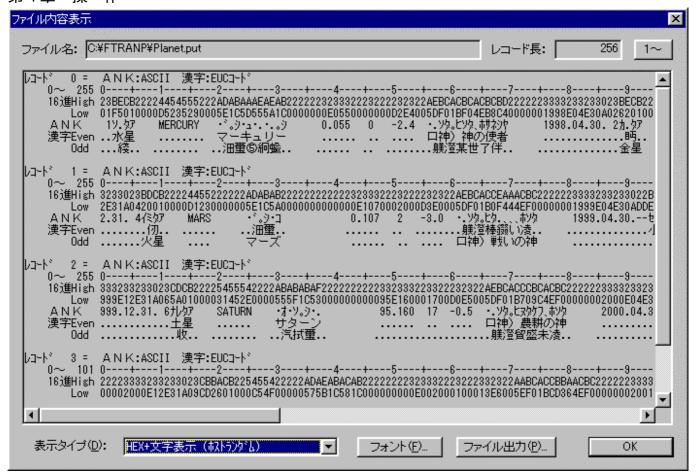
ホストファイルの中身が、HEX+文字表示(ホストテキスト&データ)形式で出力されます。 の表示タイプを6種類の中から切り替えることができます。

フォント(\underline{F})ボタンをクリックして、 の出力文字フォントを切り替えることができます。 ファイル出力(P)ボタンをクリックして、 の内容をファイルに保存できます。

桁位置のガイド表示を、0起点(先頭を0として数える)

1起点(先頭を1として数える)のどちらかに切り替えることができます。

OKボタンをクリックすると、ファイル内容表示ウインドウを閉じます。



< H E X + 文字表示(ホストランダム) >

<文字表示(ホスト)>

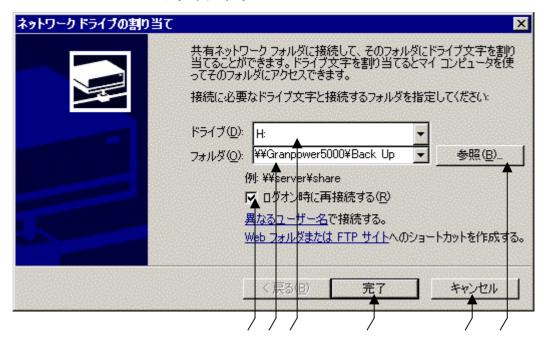


ネットワークドライブの割り当て

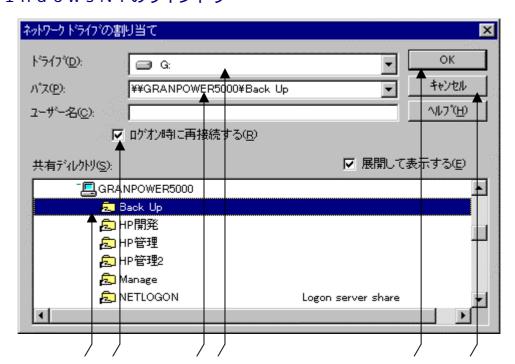
Winファイラの中でネットワーク上のファイルを扱うには、対象とするファイルが存在する場所をネットワークドライブとして割り当てる必要があります。

Winファイラの " ネットワークドライブの割り当て (\underline{I})..." ボタンをクリックして、つぎの操作を行います。

(1)Windows2000のウインドウ



(2)WindowsNTのウインドウ



(3)Windows98/95のウインドウ



割り当てをするドライブを選択します。

割り当てをするパス名を入力します。Windows2000またはNTの場合は、 をクリックしてパスの選択をすることができます。

電源再投入時にネットワークドライブの割り当てを有効にする場合は、

このチェックボックスをONにします。

設定が完了したら、OKボタンをクリックします。

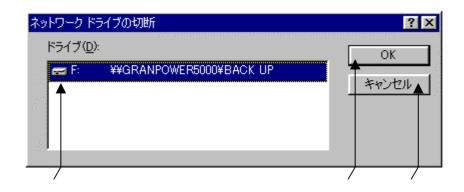
割り当てを中止する場合は、キャンセルボタンをクリックします。

注意・・・

割り当てをしたドライブはWinファイラの中だけではなく、Windows上において、通常のドライブと同様に扱われます。

ネットワークドライブの切断

Winファイラの中で不要になったネットワークドライブを切断するには、Winファイラの "ネットワークドライブの切断 (\underline{O})..." ボタンをクリックして、つぎの操作を行います。



切断するネットワークドライブをクリックして、選択します。 選択が完了したら、OKボタンをクリックします。

切断を中止する場合は、キャンセルボタンをクリックします。

4 . 12 変換設定

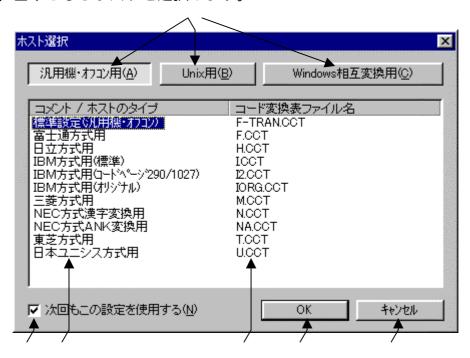
変換設定には、コード変換等に関するさまざまな設定項目があります。この設定項目は、メモリ上のコード変換表の中身です。変更した設定は即座に変換処理の動作に反映されます。

通常、ホスト選択で指定したコード変換表に対して、その中身を設定し、自分の運用に適した 設定に変更していきます。その作業は、ふつうインストールの直後に1回だけ行います。しかし、 運用に入ってからでも随時、設定の変更ができます。

データ交換の相手システムが複数あるときは、コード変換表の名前と設定を変えて、相手システムごとに切り替えて使うと便利です。

ホスト選択

メニューバーの変換設定(\underline{S}) ホスト選択(\underline{H})を実行すると、つぎのウインドウが開きます。ここでは、基準となるホストを選択します。



ホスト系を汎用機・オフコン用(\underline{A}) \underline{U} \underline{N} $\underline{N$

選択できるホストのタイプが表示されるエリアです。

使用しているホストのタイプをクリックして選択します。

実際に設定されているコード変換表ファイル名が表示されるエリアです。

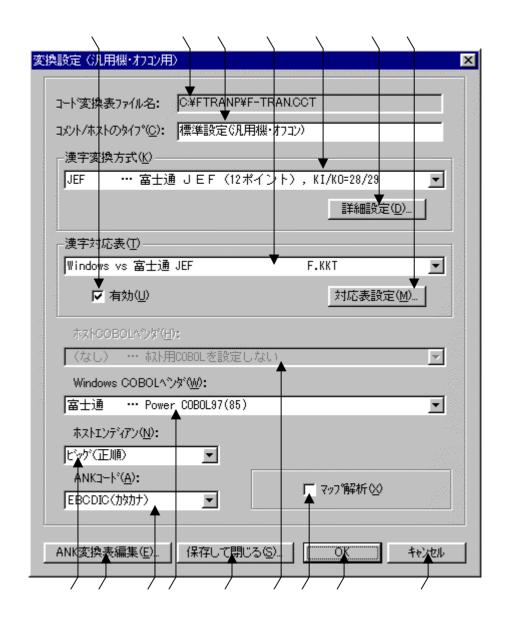
選択したホストの情報を保存するためのチェックボックスです。

次回にF*TRAN+を起動した時に、選択したホストを基準にする場合は、ONにします。 一時的にホストを変更する場合は、OFFにします。デフォルトは、ONです。

の選択が完了した状態でOKボタンをクリックすると、基準となるホストが確定します。 キャンセルボタンをクリックすると、ホスト選択を中止します。

变換設定

メニューバーの変換設定(S) 変換設定(S)を実行すると、つぎのウインドウが開きます。



読み込まれているコード変換表のファイル名が表示されています。

コメントの変更ができます。

コード変換表は、漢字20文字(半角40文字)以内のコメントをつけることができます。

漢字変換方式の選択項目です。どの漢字変換方式を使うかを割り当てます。

漢字変換方式の設定変更ボタンです。漢字変換方式の設定内容を細かに変更できます。

漢字変換時に使われる漢字対応表を選択することができます。

選択されている漢字対応表を有効にするチェックボックスです。

漢字対応表エディタを起動するボタンです。漢字対応表の修正ができます。

ホストがUnix、Windowsの場合、ホストCOBOLのベンダを選択します。

ホストCOBOLのゾーン形式データを変換する時に重要になります。

Windows COBOLのベンダを選択します。

Windows COBOLのゾーン形式データを変換する時に重要になります。

ホストエンディアン(ホストの2進項目のデータ格納順)を選択します。

ANKコードの選択項目です。ホスト系が、JIS8/ASCII系か、EBCDIC系かを設定します。EBCDIC系の場合、カタカナ版か英小文字版の選択をします。

ANK変換表の編集ボタンです。

6つのANK変換表(1バイト単位のコード変換表)の表示・修正ができます。

コード変換表の書き込み(保存)を行い、変換設定のウインドウを閉じます。

修正したコード変換表を元のファイルに書き戻すだけでなく、

別の名前のコード変換表としてディスクに書き込むこともできます。

OKボタンをクリックすると、修正した変換設定が有効になります。

ただし、あくまでメモリ上の変更であり、

コード変換表ファイルに書き込まれるわけではありません。

キャンセルボタンをクリックすると、修正した変換設定を無効にし、

変換設定のウインドウを閉じます。

マップ解析の機能をON/OFFするチェックボックスです。デフォルトは、OFFです。 詳細は、マルチレコード編のマニュアルを参照してください。

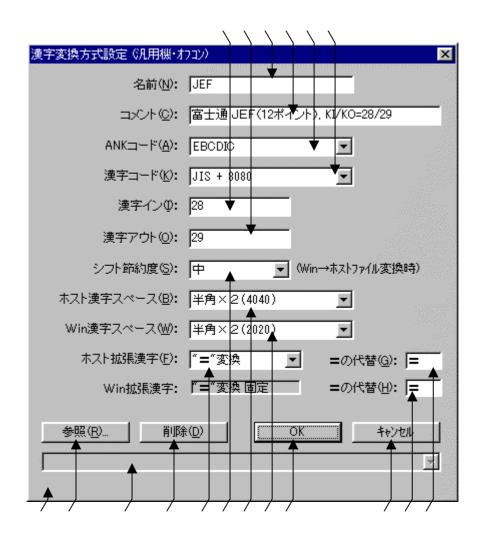
コメント / ホストのタイプの設定

コード変換表ごとにつけるコメントを設定できます。漢字20文字(半角40文字)以内で、 通常、ホストのタイプを設定します。

漢字変換方式の設定

変換処理時に使う漢字変換方式を選択します。選択後に、詳細設定(<u>S</u>)のボタンをクリックすると、漢字変換方式設定のウインドウが開き、細かな設定内容を変更することができます。新規に作成する場合は、「新規作成」を選んでから詳細設定のボタンをクリックします。ただし、登録できるのは合計で15種類までです。

< ホストが汎用機・オフコンの場合 >



この漢字変換方式につける名前です。英数字8文字以内で設定できます。

設定内容をわかりやすくするためのコメントです。

漢字20文字(半角40文字)以内で設定できます。

ANKコードがJIS8/ASCII系か、EBCDIC系かを設定します。

この設定は、Ank・漢字(KI/KO付)混在変換のANK部の変換や、

漢字スペースのコードを決めるために使用されます。EBCDIC系の時に、

カタカナ版、英小文字版のどちらになるかは、変換設定のANKコードの設定で決まります。

漢字コードのタイプを設定します。以下の中から選択します。

JIS、JIS+8080、JIS+8080(ユニシス) 東芝(漢字コード自体は JIS漢字) NEC-JIPS、NEC-内部コード、IBM

- JIS漢字の定義域は2121H~7E7EHです。JIS+8080は、
- JISに8080Hを加えて定義域をA1A1H~FEFEHにずらしています。
- 多くのホストシステムがJIS+8080を採用しています。その理由の1つは、
- JIS漢字との相互変換が極めて簡単であることです。2つ目の理由は、
- ANKコードとしてEBCDICコードを採用していることが多いため、
- JIS漢字をそのまま採用すると、EBCDICコードの制御文字のうち、
- 21日~3FHがJIS漢字の一部とぶつかってしまうからです。
- 以上の2タイプと比較すると、NEC漢字はかなり特殊です。NEC漢字には、

JIPS(E) JIPS(J) 内部コード(E) 内部コード(J)

- の4とおりのコード系があります。JIS8/ASCII系かEBCDIC系かは、 のANKコードの設定で間接的に決められます。
- ()内のEは、EBCDIC系、Jは、JIS8系を意味しています。
- IBM漢字は、IBM漢字 シフトJIS漢字用の内部テーブルを持って対応しています。 漢字インのシーケンスを16進で6桁(3バイト)以内で指定します。
- 漢字アウトのシーケンスを16進で6桁(3バイト)以内で指定します。
- Win ホストファイル変換時に、Ank・漢字(KI/KO付)混在変換するときの、
- シフト節約度を設定します。弱、中、強の3とおりの設定ができます。
- [弱] 半角スペース(20H)はあくまでANK、全角スペース(8140H)はあくまで漢字、として完全に区別して扱います。この設定にすると、頻繁にKI/KOが入る傾向があり、レコード長や項目長のオーバーフローを起こすことが多くなります。ただし、Windowsファイル側で半角スペースと全角スペースを完全に使い分けているときは、この設定が一番よいこともあります。
- [中] 半角スペース2個(2020H)と全角スペース(8140H)を同一視し、なるべくANKモード、漢字モードが長く続くようにします。いい替えれば、ANK文字に続く半角スペースはANKとして、漢字に続く半角スペースは2個1組で漢字スペースとして扱うということです。ただし、漢字モード中で奇数個の半角スペースが続くと、最後の1個だけはANKとして扱います。この設定が、いろいろな意味で無難です。
- [強] 半角スペース2個(2020H)と全角スペース(8140H)を同一視し、なるべくANKモード、漢字モードが長く続くようにします。また、ホストファイル側の漢字モード中に奇数個の半角スペースが現れてもよしとします。この設定のデータを受け付けることのできるシステムやソフトは少ないので、注意してください。なお、ANKがEBCDIC、漢字がJIS漢字の組み合わせのときは、この設定にしてはいけません。スペースが40××Hというコードの漢字に化けてしまいます。

例をあげます。 はANKスペース、 は漢字スペースです。弱 中 強の順にKI/KO の節約になり、オーバーフローしにくくなるのがわかると思います。

Win ホストファイル変換時に、漢字スペースを全角のコードにするか、 半角×2個のコードにするかを設定します。具体的なコードは、 ANKコードと漢字コードの組み合わせによって、つぎの表のように決まります。

漢字スペースのコード(Win ホスト変換時)

	漢字スペース	全 角	半角 × 2
ANKコード&漢字コード			
JIS8/ASCII	JIS	2 1 2 1 H	2020H
	JIS+8080	A 1 A 1 H *1	2020H
	NEC JIPS(J)	2 1 2 1 H	2020H
	NEC内部コード(J)	6021H	2020H
	IBM漢字	4040H	2020H
EBCDIC	JIS	2 1 2 1 H	禁止 *2
	JIS+8080	A 1 A 1 H *1	4040H
	NEC JIPS(E)	4 F 4 F H	4040H
	NEC内部コード(E)	7 9 4 F H	4040H
	IBM漢字	4040H	4040H

^{*1)} JIS+8080(ユニシス) の場合は、2020 Hにも設定可能

ホスト Winファイル変換時に、Windowsファイル側の漢字スペースを、 全角 = 8 1 4 0 Hのコードにするか、半角 \times 2 = 2 0 2 0 Hのコードにするかを設定します。 ふつうは、半角 \times 2 の設定のほうがよいのですが、パソコンのソフトによっては、 全角の漢字スペースしか受けつけないものがあります。

ホストファイル側の漢字対応表に定義されないホスト拡張漢字 (ユーザー定義文字を含む) の扱い方を設定します。

ゲタ変換、クエスチョン変換、ゼロイー変換と呼ぶ3とおりの設定ができます。

^{*2)} もし半角×2の設定にすると、漢字スペースが「誓」の字に化ける

[" ■ " 変換 (ゲタ変換)]

ホスト漢字を"〓(ゲタ)"に置き換える。

" ■ "以外の全角文字にも変換可能。

ホスト Win方向の変換時に有効。元のコードには戻せない。 桁ずれは起こさない。

["?xxxx"変換(クエスチョン変換)]

?は全角。xxxx は半角 4 桁の 1 6 進数でホストコードを表す。

ホスト Win双方向で有効。

エディタで処理するのに都合がよく、扱いやすい。

ただし、2バイト 6バイトの変換なので、桁ずれを起こすことに注意。

[OExxxx 変換 (ゼロイー変換)]

OE(SO)で、つぎにホストコード xxxx が続くことを表す。xxxx はバイナリ 2 バイト。ホスト Win 双方向で有効。

この変換方法では、ホストが富士通機でパソコンが F M V シリーズ、かつパソコン側の ソフトが J E F 拡張漢字ドライバに対応しているときに限って、 J E F 拡張漢字の画面 表示やプリントができる。 2 バイト 3 バイトの変換なので、桁ずれを起こすことに 注意。

変換の対象になるのは、句点コードでいえば1区1点~94区94点以外の漢字コードです。 JIS漢字なら2121H~7E7EH以外の漢字コード、

JIS+8080タイプならA1A1H~FEFEH以外の漢字コードです。

ゲタ変換のとき、"■"に代わる別の全角文字を設定できます。

また、半角スペース2個の設定にもできます。

代わりの文字の設定を解除して"〓"に戻したいときは、半角の"≡"を入力します。

Windowsの拡張漢字(95区以上の漢字)に対しては、"〓"変換固定です。

" ■ " に代わる別の全角文字を設定できます。また、半角スペース 2 個の設定にもできます。 代わりの文字の設定を解除して " ■ " に戻したいときは、半角の " = " を入力します。

入力ガイダンス・メッセージが表示されるフィールドです。

参照(R)ボタンは、既存の漢字変換方式設定の中から1つを選択して、

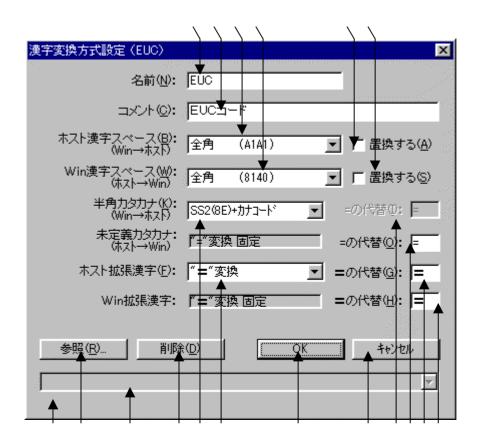
~ の項目を基準値にする機能です。

おもに、既存値をベースに「新規作成」を行う場合に重宝します。

の参照ボタンをクリックすると、既存の漢字変換方式設定の参照ダイアログが開きます。 削除(<u>D</u>)ボタンをクリックすると、表示されている漢字変換方式設定を削除し、 設定ウインドウを閉じます

OKボタンをクリックすると、ウインドウ上の漢字変換方式設定が有効になります。 キャンセルボタンをクリックすると、ウインドウ上の漢字変換方式設定を無効にし、 設定のウインドウを閉じます。

<ホストがUnixの場合>



この漢字変換方式につける名前です。英数字8文字以内で設定できます。

設定内容をわかりやすくするためのコメントです。

漢字20文字(半角40文字)以内で設定できます。

Win ホストファイル変換時に、Unixファイル側の漢字スペースを、

全角=A1A1Hのコードにするか、半角×2=2020Hのコードにするかを設定します。

Win ホストファイル変換時に、漢字スペースの置換をする/しないを選択します。

ホスト Winファイル変換時に、Windowsファイル側の漢字スペースを、

全角 = 8 1 4 0 Hのコードにするか、半角×2 = 2 0 2 0 Hのコードにするかを設定します。

ふつうは、半角×2の設定のほうがよいのですが、パソコンのソフトによっては、

全角の漢字スペースしか受けつけないものがあります。

ホスト Winファイル変換時に、漢字スペースの置換をする/しないを選択します。

Win ホストファイル変換時の半角カナ文字を、8 E H + カナコード、全角化、"="変換の3つの変換方法から選択できます。

で"="変換を選択すると、"="に代わる別の半角文字を設定できます。

ホスト Win ファイル変換時の未定義半角カナ文字に対しては、"="変換固定です。

"="に代わる別の半角文字を設定できます。

Unixファイル側の漢字対応表に定義されないホスト拡張漢字(ユーザー定義文字を含む)の扱い方を設定します。ゲタ変換、クエスチョン変換と呼ぶ2とおりの設定ができます。

["〓"変換(ゲタ変換)]

ホスト漢字を"〓(ゲタ)"に置き換える。

" ■ "以外の全角文字にも変換可能。

ホスト Win方向の変換時に有効。元のコードには戻せない。 桁ずれは起こさない。

["?xxxx"変換(クエスチョン変換)]

?は全角。xxxx は半角 4 桁の 1 6 進数でホストコードを表す。

ホスト Win 双方向で有効。

エディタで処理するのに都合がよく、扱いやすい。

ただし、2バイト 6バイトの変換なので、桁ずれを起こすことに注意。

変換の対象になるのは、句点コードでいえば $1 \boxtimes 1$ 点 $\sim 94 \boxtimes 94$ 点以外の漢字コードです。 EUCならA1A1H~FEFEH以外の漢字コードです。

ゲタ変換のとき、"■"に代わる別の全角文字を設定できます。

また、半角スペース2個の設定にもできます。

代わりの文字の設定を解除して"〓"に戻したいときは、半角の"≡"を入力します。

Windowsの拡張漢字(95区以上の漢字)に対しては、"〓"変換固定です。

"〓"に代わる別の全角文字を設定できます。また、半角スペース2個の設定にもできます。 代わりの文字の設定を解除して"〓"に戻したいときは、半角の"≡"を入力します。

入力ガイダンス・メッセージが表示されるフィールドです。

参照(R)ボタンは、既存の漢字変換方式設定の中から1つを選択して、

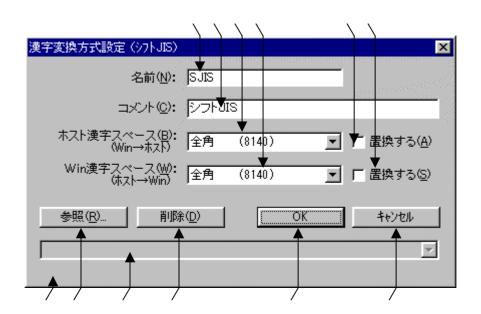
~ の項目を基準値にする機能です。

おもに、既存値をベースに「新規作成」を行う場合に重宝します。

の参照ボタンをクリックすると、既存の漢字変換方式設定の参照ダイアログが開きます。 削除(<u>D</u>)ボタンをクリックすると、表示されている漢字変換方式設定を削除し、 設定ウインドウを閉じます

OKボタンをクリックすると、ウインドウ上の漢字変換方式設定が有効になります。 キャンセルボタンをクリックすると、ウインドウ上の漢字変換方式設定を無効にし、 設定のウインドウを閉じます。

<ホストがWindowsの場合>



この漢字変換方式につける名前です。英数字8文字以内で設定できます。

設定内容をわかりやすくするためのコメントです。

漢字20文字(半角40文字)以内で設定できます。

Win ホストファイル変換時に、ホスト(出力)側の漢字スペースを、

全角=8140Hのコードにするか、半角×2=2020Hのコードにするかを設定します。

Win ホストファイル変換時に、漢字スペースの置換をする / しないを選択します。

ホスト Winファイル変換時に、Win(出力)側の漢字スペースを、

全角 = 8 1 4 0 Hのコードにするか、半角×2 = 2 0 2 0 Hのコードにするかを設定します。

ホスト Winファイル変換時に、漢字スペースの置換をする/しないを選択します。

入力ガイダンス・メッセージが表示されるフィールドです。

参照(R)ボタンは、既存の漢字変換方式設定の中から1つを選択して、

~ の項目を基準値にする機能です。

おもに、既存値をベースに「新規作成」を行う場合に重宝します。

の参照ボタンをクリックすると、既存の漢字変換方式設定の参照ダイアログが開きます。

削除(D)ボタンをクリックすると、表示されている漢字変換方式設定を削除し、

設定ウインドウを閉じます

OKボタンをクリックすると、ウインドウ上の漢字変換方式設定が有効になります。

キャンセルボタンをクリックすると、ウインドウ上の漢字変換方式設定を無効にし、

設定のウインドウを閉じます。

漢字対応表の設定

漢字変換時に使われる漢字対応表を設定します。漢字体系に合った漢字対応表を選択し、有効のチェックボックスをONにします。対応表設定(\underline{S})のボタンをクリックすると、漢字対応表エディタが起動し、設定内容の変更ができます。漢字対応表(\underline{T})が"(新規作成)"の場合は、ファイル名の入力をした後に、漢字対応表エディタが起動します。



詳細は、4.13 漢字対応表エディタをご覧ください。

ホストCOBOLベンダの設定

ホストCOBOLのゾーン形式データを変換する場合は、この設定が重要になります。つぎの 7種類から選択できます。選択しないこともできます。

富士通	Power COBOL97(85)
日立	C O B O L 8 5
NEC	C O B O L 8 5
メラント	MICRO FOCUS COBOL *
Acucorp	AcuCOBOL (コンパイルオプション : -Dca)
Acucorp	AcuCOBOL (コンパイルオプション : -Dci)
Acucorp	AcuCOBOL (コンパイルオプション : -Dcm)

*)ホストがWindowsの場合は、

メラント Net Express (MICRO FOCUS COBOL) です。

Windows COBOLベンダの設定

Windows COBOLのゾーン形式データを変換する場合は、この設定が重要になります。つぎの7種類から選択できます。選択しないこともできます。

Power COBOL97(85)
C O B O L 8 5
C O B O L 8 5
Net Express (MICRO FOCUS COBOL)
AcuCOBOL (コンパイルオプション : - Dca)
AcuCOBOL (コンパイルオプション : - Dci)
AcuCOBOL (コンパイルオプション : - Dcm)

ホストエンディアンの設定

2 進形式データを変換する場合は、この設定が重要になります。ホスト2進形式のエンディアン(格納順)をつぎのどちらかに設定します。

ビッグ(正順) リトル(逆順)

ANKコードの設定

ホストシステムには、JIS8/ASCII系のシステムと、EBCDIC系のシステムがあります。さらに、EBCDICコードには、カタカナ版と英小文字版があります。それらのどれに合わせるかを設定します。

この設定はとても重要です。というのは、JIS8/ASCII系にするかEBCDIC系にするかで、単純なANK変換にとどまらず、バッファをクリアするコードや、ゾーン形式、パック形式の変換などにも影響を及ぼすからです。

第4章 操作 ——

ANK变換表

6種類のANK変換表があります。

EBCDIC(カタカナ)JIS8変換表JIS8 EBCDIC(カタカナ)変換表EBCDIC(英小文字)ASCII変換表ASCII EBCDIC(英小文字)変換表ユーザー用(A)変換表ユーザー用(B)変換表

このうち、ユーザー用(A)変換表、ユーザー用(B)変換表は、ふつうは使わない特別なものです。

< ANK変換表について>

ANK変換表はつぎの3組に分けられます。

EBCDIC(カタカナ版) JIS8EBCDIC(英小文字版) ASCIIユーザー用(A/B)

前の2組が基本的なEBCDIC系 JIS8/ASCII系ANK変換表です。ふつう、

EBCDICコードではカタカナと英小文字のコードが重複していて、 両方同時には使えない

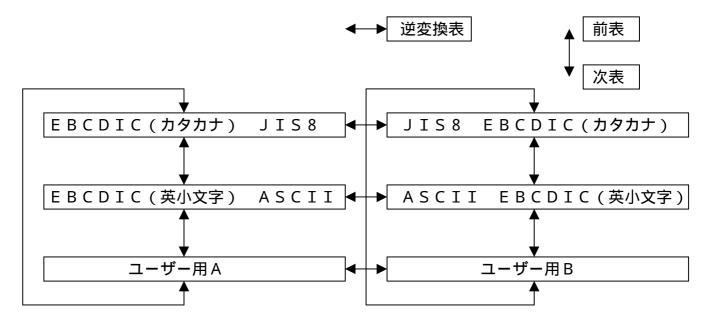
ため、カタカナ版と英小文字版の2組に分けてあります。

この2組のANK変換表は、あらかじめかなり一般的だと思われる設定にしてあります。しかし、

EBCDIC系とJIS8/ASCII系では、 いくつかの特殊文字が対応していない EBCDICコード自体もシステムによって若干の差異がある

などの理由で、メーカー・システムによって、バラバラなコード変換規則を採用しているのが 現状です。そのため、使用目的によっては不都合な変換が行われることがあります。この問題に 対処するため、ANK変換表を簡単に修正できるようにしています。 ユーザー用(A/B)変換表は、利用者独自の特別な1バイト単位の変換処理が必要なときに使用します。ただし、この2枚はふつうのANK変換に使うものではありません。

ANK変換表はそれぞれがつぎのようにリンクし合っていて、相互に呼び出せます。



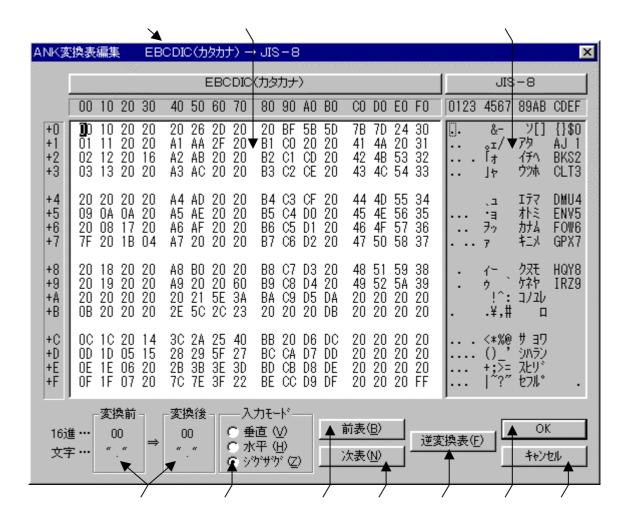
この仕組みによって、ANK変換表どうしの比較や、正しく逆変換表になっているかの確認の ための突き合わせなどが簡単にできます。

なお、ユーザー用Aとユーザー用Bは、本当は互いに逆変換表になっているわけではありません。それぞれ独立した変換表です。

ANK変換表編集 (\underline{E}) のボタンをクリックすると、下のウインドウが開きます。編集する ANK変換表を選択して、OKボタンをクリックします。



OKボタンがクリックされると、下のウインドウが開き、編集モードに入ります。



編集するANK変換表名が表示されます。

変換表の内容が16進表示されるエリアです。変更入力することができます。 変換表の内容が文字表示されるエリアです。

変換元の16進コードと文字、変換後の16進コードと文字が表示されます。 カーソル移動にとともに表示内容が変わります。

16進表示側のC0の列、+1の行のコードは、"41"と表示されています。 図形文字表示側のCの列、+1の行の文字は、"A"と表示されています。

これは、EBCDICコードのC1H("A")が、

JIS8コードの41H("A")に変換されることを表しています。

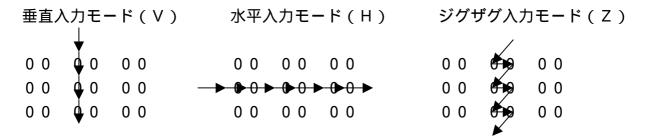
ここにカーソルを合わせると、00 . 00 .と表示されていたところが、

C 1 A 4 1 A という表示に変わったはずです。

変換表を変更入力する場合の入力モード(カーソルの動き)を指定するラジオボタンです。

入力モードとカーソルの動き

入力モードによるカーソルの動きの違いを示します。



前表に移るためのボタンです。カーソル位置は変わりません。

次表に移るためのボタンです。カーソル位置は変わりません。

逆変換表に移るためのボタンです。

カーソル位置の内容が逆変換表のカーソルの位置になります。

OKボタンをクリックすると、修正したANK変換表が有効になります。ただし、

あくまでメモリ上の変更であり、コード変換表ファイルに書き込まれるわけではありません。

キャンセルボタンをクリックすると、修正したANK変換表を無効にし、

ANK変換表のウインドウを閉じます。

<ユーザー用(A/B)変換表について>

ユーザー用(A/B)は、特別な変換処理を行う変換表です。バイト単位の変換ができます。 利用者独自の処理が必要になったときに書き替えて使います。

使用目的 通常のEBCDIC JIS8/ASCII変換を使うべきでない、特殊な1 バイト単位の変換が必要になるときがあります。たとえば、 特殊漢字 JIS漢字変換 などが挙げられます。

利用できる所 ユーザー用(A/B)変換表は、マップ設定オプションのユーザーA変換、 ユーザーB変換で利用できます。

出荷時設定 ユーザー用(A/B)変換表の出荷時の設定は、何もしない変換表、つまりバイナリ変換と同様になっています。

コード変換表の書き込み(保存)

コード変換表の書き込み(保存)は、現在のメモリ上のコード変換表を元のコード変換表ファイルに保存、または別のコード変換表ファイルとしてディスクに書き込みます。

保存して閉じる(\underline{S}) ボタンをクリックすると、"ファイル名を付けて保存"ウインドウが開きます。保存するファイルを選択して、保存(\underline{S}) ボタンをクリックします。

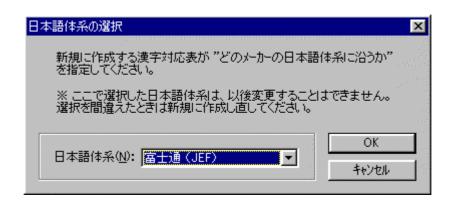
4.13 漢字対応表エディタ

漢字対応表エディタは、F * T R A N + の漢字コード変換で使用する漢字変換テーブルのメンテナンスを行う機能を持っています。

通常は、変換設定の対応表設定(S)のボタンをクリックして起動します。

新規作成の場合

漢字対応表を新規に作成する場合は、つぎのウインドウが開き、どのメーカーの日本語体系で作成するのかを選択します。



選択肢は、つぎのとおりです。

```
各社(JIS)

富士通(JEF)

日立(KEIS)

三菱(MELCOM)

三菱(MELCOM)

日本ユニシス(LETS-J)

NEC(JIPS)

日本IBM(IBM)

東芝(JIS)

シフトJIS

EUC、G3あり(補助漢字)

EUC、G3あり(JEF拡張)

Super DEC漢字
```

日本語体系(N)を選択し、OKボタンをクリックすると、編集ウインドウが開きます。

既存の漢字対応表を修正する場合

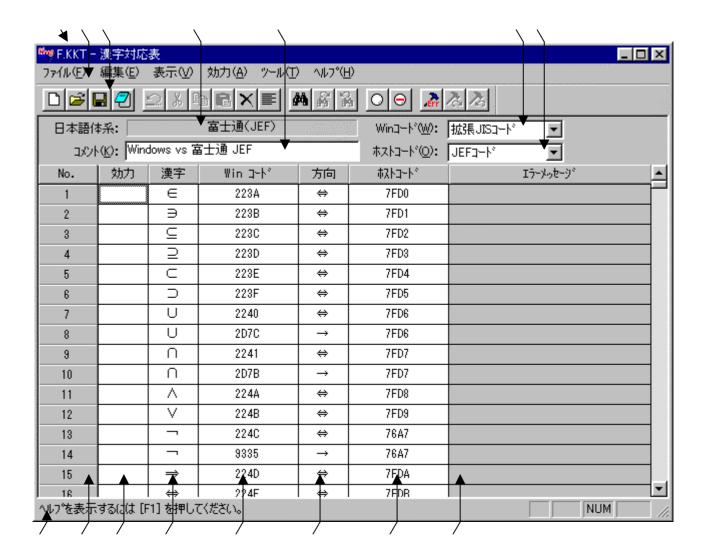
F*TRAN+には、各メーカーの日本語体系に沿った漢字対応表が標準提供されています。 通常は、提供されている漢字対応表をベースに変更して使用します。

変換設定で漢字対応表を選択して対応表設定(<u>S</u>)のボタンをクリックすると、編集ウインドウが開き、選択された漢字対応表の内容が読み込まれます。



F*TRAN+には、つぎの12種類の漢字対応表が標準添付されています。

編集ウインドウ



編集する漢字対応表のファイル名が表示されています。

目的のメニューを選択して、機能を実行します。

の中で、よく使われる機能がボタン化されています。

編集する漢字対応表の日本語体系が表示されています。

編集する漢字対応表のコメントが表示されています。

新規に作成する場合は、漢字20文字(半角40文字)以内で設定します。変更もできます。

Winコードのコード表記法を3種類の中から選択します。

ホストコードのコード表記法を選択肢の中から選択します。

No.1からNo.4000まであります。ここをクリックして、行選択を行います。

行の内容を消去せずに無効にする場合に、無効"-"の印を付ける欄です。

Win側の漢字を入力します。ダブルクリックで、漢字入力サブウインドウが開きます。

Winコードを入力します。とは連動し、どちらかを入力すれば表示が変わります。

変換する方向を3種類(、、、、)の中から選択し、入力します。

ホストコードを入力します。

ファイルの読み / 書き、" 検査 " を実行した時のエラーメッセージが表示されます。 操作のガイダンスやメッセージが表示されます。

コメント(K)

漢字対応表のコメント欄です。通常は、漢字対応表の内容を意味するタイトルを入力します。 漢字20文字(半角40文字)以内で設定します。

$Win \exists -F(W)$

Winコード(Windowsの漢字コード)の表示・入力をどのコード表記法に従って行うかを指定します。つぎの3種類の中から選択します。

拡張 J I S コード (基準値) 拡張句点コード シフト J I S コード

立ち上げ時/ファイル読み込み時には、常に拡張JISコード(基準値)になります。

ホストコード (H)

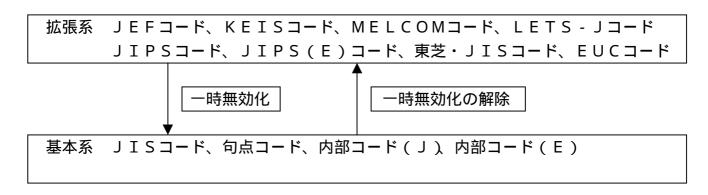
選択されている日本語体系ごとにホストコードの表記法が異なり、つぎの中から選択します。

日本語体系	ホストコード	基準値	系列
各社(JIS)	JISコード		基本
	句点コード		基本
富士通(JEF)	JISI-F		基本
	句点コード		基本
	JEFコード		拡張
日立(KEIS)	JISI-F		基本
— ()	句点コード		基本
	K E I Sコード		拡張
三菱(MELCOM)	JISコード		基本
	句点コード		基本
	MELCOMコード		拡張
日本ユニシス(LETS-J)	JISコード		基本
	句点コード		基本
	LETS - Jコード		拡張
NEC(JIPS)	JISコード		基本
	句点コ ー ド		基本
	JIPSコード		拡張
	JIPS(E)コード		拡張
	内部コード(J)		基本
	内部コ ー ド(E)		基本
日本IBM(IBM)	IBM漢字コード		基本
東芝(JIS)	JISコード		拡張
	句点コード		基本
シフトJIS	拡張JISコード		拡張
	拡張句点コード		拡張
	シフトJISコード		拡張
EUC、G3なし	JISコード		基本
DEC漢字	句点コード		基本
EUC、G3あり(補助漢字)	EUCコード		拡張
EUC、G3あり(JEF拡張)			
Super DEC漢字			

立ち上げ時/ファイル読み込み時には、常に基準値になります。

系列が"拡張"の場合のみ、拡張漢字領域のコード入力ができます。

ホストコードは基本系と拡張系に大別できます。ホストコードの選択を拡張系から基本系に変更すると、拡張コードが表現できなくなり、拡張コードを持つ入力データの一時無効化が起こります。効力欄が"(-)"になり、ホストコード欄が"****"になります。逆に、基本系から拡張系に変更すると、一時無効になっていた入力データが解除されて通常の表示に戻ります。実際には、つぎの図のようになります。



例)JEFコード	効力	漢字	Winコード	方向	ホストコード
			2 2 3 A		7 F D 0

JISコード	効力	漢字	Winコード	方向	ホストコード
	(-)		2 2 3 A		* * * *

入力基本パターン

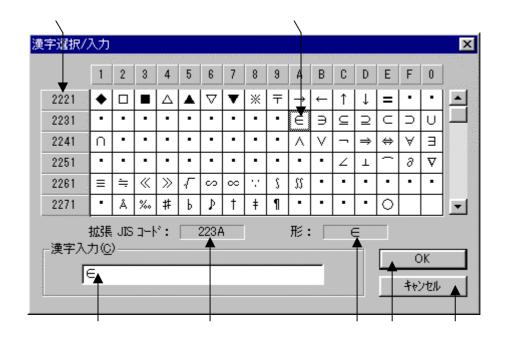
漢字対応表に定義される基本パターンはつぎのとおりです。

- 1. Windows 漢字 (またはコード) → → → ホストの漢字コード (双方向) (双方向)
- 2. Windows漢字(またはコード) ホストの漢字コード (片方向)
- 3. Windows 漢字 (またはコード) **◆** ホストの漢字コード (片方向)

Windows漢字またはWindowsの漢字コード、変換する方向、ホストの漢字コードの3要素を入力します。Windowsの漢字コードおよびホストの漢字コードは、正しいコードの入力のみ受け付けます。句点コードを入力する場合は、12-34のように"-"で区切って入力します。EUCのG3コードを入力する場合は、G3:A1A1のように入力します。

漢字入力サブウインドウ

漢字欄をダブルクリックすると、つぎのウインドウが開きます。



選択されているWinコードの表記法(拡張JISコード、拡張句点コード、シフトJISコード)に従って、コードが表示されます。

漢字の一覧が表示されます。入力する場合は、漢字をダブルクリックします。

で選択されている漢字のコードが表示されます。

で選択されている漢字が表示されます。

このエディットボックスの中で文字列を入力し、目的の漢字を左端に置くことで、 選択することもできます。

漢字選択 / 入力後に、 O K ボタンをクリックして確定させます。

キャンセルボタンをクリックすると、漢字入力サブウインドウが閉じます。

ファイル (\underline{F}) メニュー

ファイルメニューの各機能について説明します。

新規作成(N) ------ "新規に作成しますか?"と確認ウインドウが開きます。

はい(Y)ボタンをクリックし、日本語体系を選択して、

OKボタンをクリックすると、

新規に漢字対応表を作成する状態になります。

開く(0) ----- "ファイルを開く"のウインドウが開きます。

ファイル名を選択し、開く(0)ボタンをクリックすると、

選択された漢字対応表が読み込まれます。

上書き保存(<u>S</u>) ------ タイトルバーに表示されている漢字対応表ファイルに、

現在の内容を保存します。

保存する前に内容を検査し、エラーがあれば、

"エラーデータがありますが、このまま保存を続行しますか?"

と確認ウインドウで聞いてきます。

名前を付けて保存(A) -- 実行すると、内容を検査します。エラーがあれば、

"エラーデータがありますが、このまま保存を続行しますか?"

と確認ウインドウで聞いてきます。

"名前を付けて保存"のウインドウが開いたら、

ファイル名を入力し、

保存(S)ボタンをクリックして保存します。

テキストファイル出力(T) ----- "テキストファイル出力"のウインドウが開きます。

ファイル名を入力し、保存(S)ボタンをクリックすると、

設定内容をテキスト形式ファイルへ出力します。

< テキストファイル出力例 >

コメント : Windows vs 富士通 JEF

Winコート : 拡張 JISコート

日本語体系 : 富士通 (JEF)

ホストコート゛ : JEF コート゛

No. 効力 漢字 Win コード 方向 ホストコード エラーメッセージ

1 223A 7FD0

漢字対応表の終了(X) -- 漢字対応表エディタを終了します。

編集(E)メニュー

編集メニューの各機能について説明します。

元に戻す(U) ---- 直前の編集作業を無効にし、元に戻します。

切り取り(T) ---- 選択されている行の内容が切り取られ、カットバッファに入ります。

コピー(C) ----- 選択されている行の内容がカットバッファに入ります。

貼り付け(\underline{P}) ---- カットバッファの内容を、セルポインタがある行へ貼り付けます。

貼り付けた行以降に既存の設定項目があれば、挿入になります。

削除(D) ------ 選択されている項目の内容が削除されます。

すべて選択(L) -- すべての設定項目欄が選択された状態になります。

一行下げる(I) -- セルポインタがある行以降が1行づつ下がります。

通常は、行挿入をする場合に使用します。

整列 (<u>S</u>) ------ Winコード、ホストコードを基準として、

設定値の並べ替えができます。

行選択をしてから実行すると、その範囲が対象になります。

つぎの4つのパターンがあります。

Winコード順(W) 昇順(U)

Winコード順(W) 降順(D)

ホストコード順(H) 昇順(U)

ホストコード順(H) 降順(D)

検索 (\underline{F}) ------ 漢字やコードで設定項目を検索することができます。

漢字(または、コード)と、検索する方向を指定して実行します。

漢字(Winコード) ホストコードの両方が指定された場合は、

どちらかが一致する項目を検索します。

漢字(C)の入力欄の横にあるボタンをクリックすると、

漢字入力サブウインドウが開きます。



ツールバーには、上方向検索ボタン / 下方向検索ボタンがあり、 指定値で再検索することができます。

表示 (<u>V</u>)メニュー

表示メニューの各機能について説明します。

ツールバー(T) ----- ツールバー表示のON/OFFができます。

ステータスバー(\underline{S}) -- ステータスバー表示のON/OFFができます。

漢字部のフォント(F) ---- 漢字欄の表示フォントを、つぎのウインドウで設定できます。



効力(A)メニュー

効力メニューの各機能について説明します。

有効化(\underline{E}) -- 選択されている行の内容を有効にし、効力欄の印" - "を消します。

ただし、一時無効の印"(-)"は消えません。

無効化(\underline{D}) -- 選択されている行の内容を消去せずに無効にし、

効力欄に印"-"を表示します。

ツール(T)メニュー

ツールメニューの各機能について説明します。

検査(<u>R</u>) ------- 設定されている内容に誤りがあるかどうかを検査します。 行選択をしてから実行すると、その範囲が対象になります。 エラーの種類はつぎのとおりです。

> 不完全 ---- Winコード、方向、ホストコードの3要素が そろわない

迷子 ------ Winコード ホストコード 別のWinコード のように、コード変換が収束しない

1対N ---- Winコード ホストコード、
Winコード 別のホストコードのように、
1つのコードが2つ以上のコードに設定されている

2重定義 -- 同一の内容が重複して設定されている

前のエラー(\underline{P}) -- 現在のセルポインタから 1 つ前のエラーがある行へ セルポインタを移動します。

次のエラー(N) -- 現在のセルポインタから次のエラーがある行へ セルポインタを移動します。

[別表A:ホスト漢字コード表記]

11:各社(JIS)・JISコード12:各社(JIS)・句点コード

21:富士通(JEF)・JISコード 22:富士通(JEF)・句点コード 23:富士通(JEF)・JEFコード 31:日立(KEIS)・JISコード 32:日立(KEIS)・句点コード 33:日立(KEIS)・KEISコード 41:三菱 (MELCOM)・JISコード 42: 三菱 (MELCOM)・ 句点コード 43:三菱(MELCOM)・MELCOMコード 51:日本ユニシス(LETS-J)・JISコード 52:日本ユニシス(LETS-J)・句点コード 53:日本ユニシス(LETS-J)・LETS-Jコード 61:NEC(JIPS)・JISコード 62: NEC (JIPS)・ 句点コード 63:NEC(JIPS)·JIPSコード 64: NEC (JIPS) · JISS (E) コード 65:NEC(JIPS)・内部コード(J) 66:NEC(JIPS)・内部コード(E) 71:日本IBM (IBM)・IBM漢字コード 81:東芝(JIS)・JISコード 82:東芝(JIS)・句点コード 91:シフトJIS・拡張JISコード 92:シフトJIS・拡張句点コード 93:シフトJIS・シフトJISコード 101: EUC、G3なし・JISコード 102:EUC、G3なし・句点コード 103:EUC、G3なし・EUCコード 111:DEC漢字・JISコード 1 1 2 : D E C 漢字・句点コード 113:DEC漢字・EUCコード 121:EUC、G3あり(補助漢字)・JISコード 122:EUC、G3あり(補助漢字)・句点コード 123:EUC、G3あり(補助漢字)・EUCコード 131:EUC、G3あり(JEF拡張)・JISコード 132:EUC、G3あり(JEF拡張)・句点コード 133:EUC、G3あり(JEF拡張)・EUCコード 141:Super DEC漢字・JISコード 142:Super DEC漢字・句点コード 143:Super DEC漢字・EUCコード

[別表 B:エラーコード番号]

0:正常

1: 不完全

2:迷子

4:1対N

8:2重定義

2つ以上のエラーがある場合は、組み合わせになります。 たとえば、エラー番号 = 12は、1対N/2重定義エラーになります。

(読み込みデータの例) 拡張 JISコード:223A 富士通JEF:7FD0

0, ,1;223A,3,23;7FD0

ヘルプ(H)メニュー

ヘルプメニューの各機能について説明します。

トピックの検索(<u>H</u>) -- トピックの検索ウインドウを開き、 漢字対応表エディタに関する操作の概要を見ることができます。

バージョン情報(<u>A</u>) -- バージョン情報のウインドウが開きます。 漢字対応表エディタのバージョン情報を知ることができます。



漢字対応表エディタの単独起動

漢字対応表エディタをF*TRAN+からではなく、エクスプローラ等から単独で起動することもできます。起動するとつぎのウインドウが開きますので、新規または既存の漢字対応表の選択から始めます。既存の漢字対応表を選択した場合は、ファイル選択を行います。以下の手順は、いままで説明したとおりです。



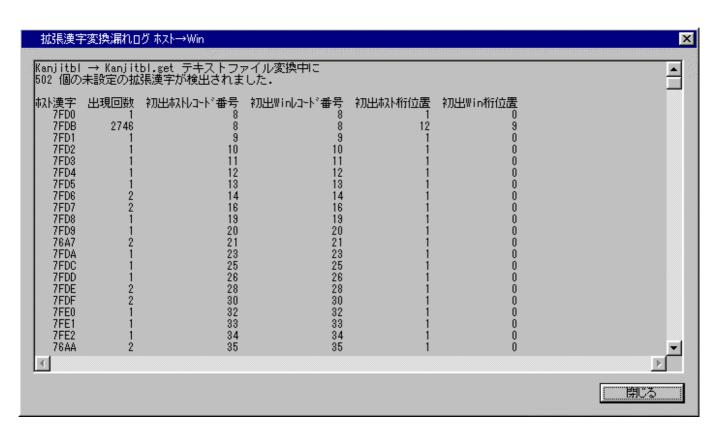
また、エクスプローラ等から漢字対応表ファイル(拡張子が".KKT"のファイル)をダブルクリックし、直接、漢字対応表エディタを立ち上げ、メンテナンスすることもできます。

4.14 変換時ログ表示

F*TRAN+は、ホスト Winファイル変換/Win ホストファイル変換実行時に、変換できない漢字をログファイルに出力しています。その変換時のログをウインドウ上に表示させて見ることができます。

変換時ログ表示(ホスト Winファイル変換)

メニューバーの変換時ログ表示 (\underline{U})のホスト Win(\underline{G})を実行すると、ホスト Winファイル変換のログ (Fpglog.txt)を見ることができます。通常は、ホスト Winファイル変換を実行した直後に確認します。



ここに出力されるホスト漢字は、変換設定/漢字変換方式の設定/ホスト拡張漢字の指定 ("〓"変換、"?xxxx"変換、0Exxxx変換)の対象になった漢字コードです。

ログの中に"〓"等の変換対象にしたくないホスト漢字があった場合は、ログに出力されたホスト漢字コードを漢字対応表に登録してから、再度、変換をしてください。

変換時ログ表示(Win ホストファイル変換)

メニューバーの変換時ログ表示(\underline{U})のWin ホスト(\underline{P})を実行すると、Win ホストファイル変換のログ(Fpplog.txt)を見ることができます。通常は、Win ホストファイル変換を実行した直後に確認します。



ここに出力されるWin漢字は、変換設定/漢字変換方式の設定/Win拡張漢字の指定("〓"変換固定)の対象になった漢字コードです。

ログの中に"〓"等の変換対象にしたくないWin漢字があった場合は、ログに出力されたWin漢字コードを漢字対応表に登録してから、再度、変換をしてください。

4.15 ヘルプ

ヘルプ・メニューには、トピックの検索、バージョン情報の2つの機能があります。

トピックの検索

メニューバーのヘルプ (\underline{H}) トピックの検索 (\underline{I}) を実行すると、トピックの検索のウインドウが開きます。

このウインドウ上で、F * T R A N + に関する用語の説明や、操作の概要を見ることができます。実際にウインドウを開いて、見ることをおすすめします。

バージョン情報

メニューバーのヘルプ (\underline{H}) バージョン情報 (\underline{A}) を実行すると、バージョン情報のウインドウが開きます。

現在、立ち上がっているF*TRAN+のバージョン情報を知ることができます。



ここには、サポートサービスを受ける場合に必要となる情報が入っています。

付 録

エラーメッセージ (共通)

起動・終了時のエラーメッセージ

オープンできません.

F*TRAN+の起動時に、コード変換表ファイルがオープンできませんでした。大抵はコード変換表のファイル名のつづりミスです。

/ Codeオプションを省略したときは?:デフォルト.CCTを探します。?:はインストールディレクトリを表します。

必要なら / Codeオプションを指定してください。 / Codeオプションを指定したのにこのメッセージが出るときは、 / Codeオプションの指定の誤りです。 修正して再起動してください。

読み込みでエラーになりました.

何らかの入出力エラーが発生し、コード変換表ファイルを読み込めませんでした。このエラーは、めったに発生しません。

コード変換表ではありません.

F*TRAN+の起動時に、/Codeオプションで指定したコード変換表ファイルは実際にはコード変換表ではありませんでした。正しいコード変換表ファイルを指定してください。

書き込みが失敗しました..

何らかの入出力エラーが発生し、コード変換表ファイルの書き込み(保存)ができませんでした。

共通のエラーメッセージ

メモリが足りません.

作業に必要なメモリの空き容量が足りません。他のアプリケーションを終了させるなどメモリ 不足を解消してください。

コマンドがありません.

指定されたコマンドは、ありません。正しいコマンドを指定してください。

オプションはありません..

該当するオプションがありません。たぶん他のオプションのつづり誤りでしょう。間違って別のコマンドを呼び出していたということもあります。

オプションに誤りがあります..

オプションの指定に何らかの誤りがあります。修正して再実行してください。

パラメータが余分です..

パラメータの指定に誤りがあるため、うしろの部分が余分だとみなされました。修正して再実行してください。

<u>エラーが発生しました.内部エラーの可能性があります.</u>

F*TRAN+自身のエラーが発生しました。購入元または発売元に通報してください。

パラメータファイルに関するエラーメッセージ

<u>+ + の後にパラメータファイルを指定し</u>てください.

パラメータファイル参照の開始マーク + + があるのに、そのあとにファイルが指定されていませんでした。 + + のあとに空白を置いた場合も同様です。空白を置いてはいけません。

オープンできません.

指定されたパラメータファイルがオープンできませんでした。大抵はファイル名のつづり誤りです。

<u>読み込みでエラーに</u>なりました.

指定されたパラメータファイルの読み込み時に入出力エラーが発生しました。このエラーは、 めったに発生しません。

パラメータバッファがいっぱいです...

パラメータファイルを展開した結果、約64Kバイトあるパラメータバッファがいっぱいになってしまいました。パラメータファイル内の記述を単純化する必要があります。

ホストファイルに関するエラーメッセージ

ホストファイルの指定に誤りがあります...

ホストファイルの指定に構文的な誤りがあります。指定し直してください。

ホストファイル名が必要です.

ホスト側のファイル名が必要なところです。指定してください。

ホストファイルがありません.

指定されたホストファイルはありません。

ホストファイルを作成できません.

ホストファイルを作成できません。つぎのような原因が考えられます。

ファイル名や拡張子に使えない文字を含んでいた ホストファイルが書き込み禁止になっていた 目的のディレクトリがなかった

ホストファイルをオープンできません..

指定されたホストファイルをオープンできません。そのホストファイルがない場合が大部分で す。

ホストファイルをクローズできません..

ホストファイルをクローズできません。ディスクの容量不足などが考えられます。 ディスクへ出力時にこのメッセージが出たときは、そのディスクが壊れた可能性があります。 すぐに、ユーティリティ・プログラム等で検査・修復してください。

ホストファイルの書き込みでエラーになりました..

ホストファイルの書き込みでエラーになりました。ほとんどの場合、ディスクがいっぱいになってそれ以上書き込めなくなったことを意味しています。

ホストファイルの読み込みでエラーになりました...

ホストファイルの読み込みでエラーになりました。通常は発生しないエラーです。

Windows ファイルに関するエラーメッセージ

Windowsファイルの指定に誤りがあります.

Windowsファイルの指定に構文的な誤りがあります。指定し直してください。

Windowsファイル名が必要です.

Windows側のファイル名が必要なところです。指定してください。

Windowsファイルがありません.

指定されたWindowsファイルはありません。

Windowsファイルを作成できません.

Windowsファイルを作成できません。つぎのような原因が考えられます。

ファイル名や拡張子に使えない文字を含んでいた Windowsファイルが書き込み禁止になっていた 目的のディレクトリがなかった

Windowsファイルをオープンできません.

指定されたWindowsファイルをオープンできません。そのWindowsファイルがない場合が大部分です。

Windowsファイルをクローズできません.

Windowsファイルをクローズできません。ディスクの容量不足などが考えられます。 ディスクへ出力時にこのメッセージが出たときは、そのディスクが壊れた可能性があります。 すぐに、ユーティリティ・プログラム等で検査・修復してください。

Windowsファイルの書き込みでエラーになりました.

Windowsファイルの書き込みでエラーになりました。ほとんどの場合、ディスクがいっぱいになってそれ以上書き込めなくなったことを意味しています。

Windowsファイルの読み込みでエラーになりました.

Windowsファイルの読み込みでエラーになりました。通常は発生しないエラーです。

GetData(gd)コマンドのエラーメッセージ

MAPバッファがいっぱいです...

変換する項目数が多すぎて、MAPバッファがいっぱいになりました。項目数を減らせないかどうか検討してください。

GetRand(gr)コマンドのエラーメッセージ

MAPバッファがいっぱいです.

変換する項目数が多すぎて、MAPバッファがいっぱいになりました。項目数を減らせないかどうか検討してください。

PutData(pd)コマンドのエラーメッセージ

/MAPオプションの各項目をコンマ(,)で区切ってください.

/ Delimitedオプションを指定したのに、/MAPオプションでコンマ(,)を1つも指定していませんでした。

MAPバッファがいっぱいです.

変換する項目数が多すぎて、MAPバッファがいっぱいになりました。項目数を減らせないかどうか検討してください。

PutRand(pr)コマンドのエラーメッセージ

MAPバッファがいっぱいです.

変換する項目数が多すぎて、MAPバッファがいっぱいになりました。項目数を減らせないかどうか検討してください。

VirDrive(vd)コマンドのエラーメッセージ

ドライブ名が正しくありません.

第 1 パラメータとして指定したドライブ名が正しくありません。使えるのは、 A : ~ Z :、@:、?:だけです。

第1パラメータはドライブ名だけです...

第1パラメータとして、仮想ドライブ名でないものを指定しようとしました。誤りです。

パス名が長すぎます.

第2パラメータとして指定した実ドライブ・ディレクトリを表すパス名が63文字を超えました。長すぎます。

Ank(an)コマンドのエラーメッセージ

指定に誤りがあります...

指定したキーワードが、

Jis Ascii Ebcdic EbcdicKana EbcdicSmall

の、どれでもありませんでした。正しく指定し直してください。

Kanji(kan)コマンドのエラーメッセージ

指定された漢字変換方式名はありません。

Kanji(kan)コマンドのパラメータとして指定した漢字変換方式が、コード変換表に登録されていません。パラメータをつけずにKanji(kan)コマンドを呼び出して漢字変換方式の一覧を表示し、そのなかに目的の漢字変換方式があるかどうか調べてください。単純なつづりミスでないかどうかも調べてください。

Comment(com)コマンドのエラーメッセージ

コメントが長すぎます..

コード変換表につけるコメント文字列が40文字(漢字20文字)を超えました。もう少し短いコメントにしてください。

cLoad(cl)コマンドのエラーメッセージ

コード変換表ではありません..

c L o a d (c l) コマンドで指定したコード変換表ファイルは、実際にはコード変換表ファイルではありませんでした。正しいコード変換表ファイルを指定してください。

オープンできません.

c L o a d (c l) コマンドで読み込もうとしたコード変換表ファイルが、オープンできませんでした。コード変換表のファイル名のつづりを間違えた場合が大部分です。ドライブ名、ファイル名、拡張子をチェックしてください。

読み込みでエラーになりました。

何らかの入出力エラーが発生し、コード変換表ファイルを読み込めませんでした。このエラーは、めったに発生しません。

c S a v e (c s)コマンドのエラーメッセージ

書き込みが失敗しました.

コード変換表の書き込み(保存)に失敗しました。つぎのような原因が考えられます。

目的のディレクトリが見つからなかった ファイルが書き込み禁止だった ディスクの残り容量不足だった

ディスクを差し替えるなり、別のドライブや名前を指定するなりして、再実行してください。

F*TRAN+ V3.0 操作説明書・解説編

2000年 10月 第1版発行

編集・著作 株式会社 富士通ビー・エス・シー 所 在 地 〒108-8531 東京都港区芝浦 4 - 15 - 33 芝浦清水ビル プロダ ケツ&サービ ス事業部 TEL 03 - 5445 - 2101 FAX 03 - 5445 - 2109

- ・Windows、MS-DOS、Visual Basic は米国 Microsoft Corporation の米国および その他の国における登録商標または商標です。
- ・会社名および製品名はそれぞれ各社の商標または登録商標です。
- ・本書およびシステムは、改善のため事前連絡なしに変更することがあります。
- ・無断複製、および転載を禁じます。
- ・落丁、乱丁はお取り替えいたします。