プログラミング言語 16－8
（１９８８．５．１３）

ＦＯＲＴＲＡＮにおける日本語化の現状と今後の方向

黒田　幸明
ＮＴＴソフトウェア研究所

ＦＯＲＴＲＡＮは，日本で最初に日本語機能の標準化が検討された言語であり，現在は，この結果を次期ＦＯＲＴＲＡＮのＩＳＯ規格（ＦＯＲＴＲＡＮ８Ｘ）に提案中である．

ここでは，ＦＯＲＴＲＡＮの日本語機能に関する次の項目について説明する．

（１）ＦＯＲＴＲＡＮで日本語データを処理するには，新しい日本語文字型の導入が必須であること
（２）エスケープシーケンスの隠べい等日本語機能として実現すべき項目
（３）実現方式
（４）ＦＯＲＴＲＡＮ８Ｘに提案している言語仕様の概要
（５）ソースプログラムの日本語化等今後の検討課題

How to implement the national character handling facility on FORTRAN and its proposal to FORTRAN8X

Kohmei KURODA
NTT Software Laboratories

1-9-1 Kohnan Minato-ku Tokyo 108 Japan

Standardization of the national character handling facility was studied on FORTRAN for the first time. This result is now proposed to FORTRAN8X of ISO.

This paper describes as follows.

(1) Necessity of national character data type
(2) Requirements for the national character handling facility
(3) Present implementation
(4) Summary of language specifications proposed to FORTRAN8X
(5) Future tasks

1．はじめに

ＦＯＲＴＲＡＮは，日本で最初に日本語機能の標準化が検討された言語である[1]．現在は，この結果を次期ＦＯＲＴＲＡＮのＩＳＯ規格（ＦＯＲＴＲＡＮ８Ｘ）[2]に提案中である．

ここでは，ＦＯＲＴＲＡＮの日本語機能の現状について紹介するとともに，今後の検討課題について述べる．

2．日本語文字の特徴と問題点

2．1　特徴

（1）字数が１００００字以上あるので，ＡＳＣＩＩ文字のように１バイトでは表現できない．日本では，普通２バイトで表現している．

（2）字数が多いので，文字のコード値による大小比較は，あまり意味がない．電話帳などは，特殊なソートを行っている．

（3）日本語文字のうち，漢字は１文字毎に意味をもっているので，変数名に漢字を使うとプログラムが読みやすくなる．

2．2　問題点

従来の文字型データで日本語文字を処理する場合，次の問題がある．

（1）日本語文字と単バイト文字を区別するために，エスケープシーケンスが必要になる（図１）．

見た目の文字数と計算機の内部表現に差が生じるので，レコードの設計などで，エスケープシーケンスの長さを考慮しなければならず，プログラミングがむずかしくなる．

漢字端末の表示	ISO標準FORTRAN						
（14文字）							
内部表現	49	53	4F	1B 24 42	49 38	30 60	
（20ﾊﾞｲﾄ）	I	S	O	ｴｽｹｰﾌﾟ ｼｰｹﾝｽ	標	準	
	1B 28 4A	46	4F	52	54	52	41	4E
	ｴｽｹｰﾌﾟ ｼｰｹﾝｽ	F	O	R	T	R	A	N

図１　日本語文字と単バイト文字の区別の方法

（2）分離符と同じコード値をもつ日本語文字を定数に書くと翻訳エラーになる（図２）．

ｿｰｽﾌﾟﾛｸﾞﾗﾑ	’規則’								
内部表現	27	1B 24 42	35 2C	42 27	1B 28 4A	27
	’	ｴｽｹｰﾌﾟ ｼｰｹﾝｽ	規	則	ｴｽｹｰﾌﾟ ｼｰｹﾝｽ	’
ｺﾝﾊﾟｲﾗの解釈	’ $B5,B’ (J’								

図２　文字型データには書けない日本語文字の例

3．要求条件

3．1　プログラムの機能

（1）エスケープシーケンスが隠ぺいされること．具体的には，次の内容を実現する必要がある．

①入力時のエスケープシーケンスの自動削除及び出力時のエスケープシーケンスの自動挿入

②部分列処理などでの日本語文字の数え方が、バイト数でなく文字数であること

（2）すべての日本語文字を書ける日本語文字型が導入されること．

3．2　ソースプログラム

（1）日本語文字の変数名が使えること．

（2）日本語文字を含んだ注釈が使えること．

なお，（1）については，ＦＯＲＴＲＡＮ８Ｘの提案に含まれていないので，７．今後の検討課題で説明する．（2）については，既存の仕様のままで実現できるので，説明を省略する．

4．実現方式

国内のほとんどのメーカでは，ＦＯＲＴＲＡＮの日本語機能の標準化検討が開始される以前に，日本語機能を実現済みであった．その実現方式は，細かい仕様の差異はあるものの，大筋は同一の実現方式である．ここでは，その実現方式について述べる．

4．1　処理方式

次の処理を行うことにより，エスケープシーケンスをプログラムに入れない方式とし，３．１の内容を実現している．

（1）日本語文字のみからなる日本語文字型を導入する．この情報を用いて，

（2）定数中のエスケープシーケンスは，翻訳時に自動削除する．

（3）エスケープシーケンスは，入出力時に自動削除挿

入する．

4．2　日本語文字型の制約

4．1で示した方式は，3．1の要求条件を実現するために定めたものであり，従来の文字型データの機能を包含したものではない．

日本語文字型データは，制御コードを含まない日本語文字だけの文字列処理に用いる場合に有効である．

エディタなどのように制御コードを含んだ文字列処理を行う場合は，従来の文字型データを使う必要がある．

4．3　プログラム例

図3に、日本語機能がある場合とない場合のプログラム例を示す。図3（b）の例は，日本がＩＳＯに提案中の言語仕様案に基づいて書いたものである．

```
      PROGRAM  EXAM1
      CHARACTER A*14,B*8
      A='国内標準'
C           (Aの内容) 1B 24 42 国内標準 1B 28 4A
      WRITE(2,100) A
      B='際'
C           (Bの内容) 1B 24 42 際 1B 28 4A
      A(6:7)=B(4:5)
      WRITE(2,100) A
C           「国際標準」が出力される
  100 FORMAT(1X,A14)
      STOP
      END
```

（a）日本語機能がない場合

```
      PROGRAM  EXAM2
      NCHARACTER A*4
      A=NC'国内標準'
C           (Aの内容)  国内標準
      WRITE(2,100) A
      A(2:2)=NC'際'
      WRITE(2,100) A
C           「国際標準」が出力される
  100 FORMAT(1X,N4)
      STOP
      END
```

（b）日本語機能がある場合

図3　プログラム例

5．言語仕様

日本がＩＳＯに提案中の言語仕様案の概要を次に示す．

5．1　ソースプログラム

（1）日本語文字を含むときのソースプログラム1行に書ける文字数は，処理系依存である．

5．2　文字の大小順序と記憶単位

（1）文字の大小順序は規定しない．

（2）文字データと日本語文字データ間の記憶単位の関係は，規定しない．

（3）日本語文字型データの記憶単位の共有は，同じ型のときのみ可能である．

5．3　定数と型定義

（1）日本語文字の定数は，次のように書く．

NC'日本語'　又は　NC"日本語"

（2）例えば3文字の日本語文字型の変数Ａは，次のように書く．

NCHARACTER A*3

5．4　部分列

（1）日本語文字の数え方は，文字数で行う．

5．5　演算

（1）代入，結合（／／）及び比較（．ＥＱ．と．ＮＥ．）ができる．

5．6　入出力

（1）書式付き記録に，単バイト文字と日本語文字が混在してもよいが，書式付き記録の長さは，単バイト文字に換算して計算する．換算方法は処理系依存である．

（2）内部ファイルでは，日本語文字を扱えない．

（3）日本語文字データの編集用に，Ｎ［w］形編集記述子を追加する．

（4）エスケープシーケンスの削除挿入の指定方法は規定していない．

5．7　組込関数

（1）単バイト文字用と同等の組込関数を用意する．

6．ＦＯＲＴＲＡＮ日本語機能の検討体制及び経緯

検討体制及び経緯を図4に示す．

現在は，ＦＯＲＴＲＡＮ８Ｘに対するＩＳＯ／ＴＣ９７／ＳＣ２２でのＤＰ投票及びＡＮＳＩ／Ｘ３でのパブリックレビューの中で，日本の提案が検討されている．

7．今後の検討課題

7．1　型定義と定数の書き方

日本語文字型の定義方法として，次に示す2案があり，

ＩＳＯ及びＡＮＳＩでは，案１の支持者の方が多い．

（案１）複数の多バイト文字集合を扱えるようにするために，文字集合の番号を指定する方法[3)]

例：　CHARACTER（KIND=n）A*3　　（n=1,2,3･･･）

（案２）５．３－（２）で示した方法[4)]

例：　NCHARACTER　　A*3

また，定数の書き方は，５．３－（１）で示したNC'～'に対して，全言語N'～'に統一しようというのが国内の動向である．

実際のニーズ及びＣＯＢＯＬ等他の言語の動向を考慮すると，型定義は案２，定数はN'～'とするのが適切であると思われる．

７．２　ソースプログラムの日本語化

ソースプログラムの日本語化については，変数名に日本語を使えるようにすることに加えて，プログラムを入力しやすくするために，プログラム全体を２バイト文字で入力したいというニーズがある．

前者については，単バイト文字と日本語文字で同形の文字の扱いを規定すれば実現できる．

後者については，定数の分離符だけは単バイト文字でないと従来プログラムとの互換がとれなくなるという問題がある．

例えば，定数の中に日本語文字の ' を書いた従来のプログラムを新しい処理系に入力すると，定数内の ' が定数の分離符とみなされて翻訳エラーとなる．

これを解決するには，コード系を２バイトに統一するより方法がない．

７．３　処理系依存の項目の扱い

５．で示した言語仕様案には，処理系依存の項目が多い．これは，コード系の影響が言語仕様ににじみ出るからである．

これを解決するにも，コード系を２バイトに統一するより方法がない．

８．おわりに

４．２及び７．で示したように，現在提案中の日本語機能は，従来の文字型データの機能を完全に包含したものではない．根本的な解決をするには，計算機のコード系を２バイトに統一するより方法がないことを認識しておく必要がある．しかし，現実にはこのような対処は極めて困難であり，利用者のニーズ及び計算機のメモリとファイル効率を考えると，現在の提案が最も妥当な内容であるといえる．

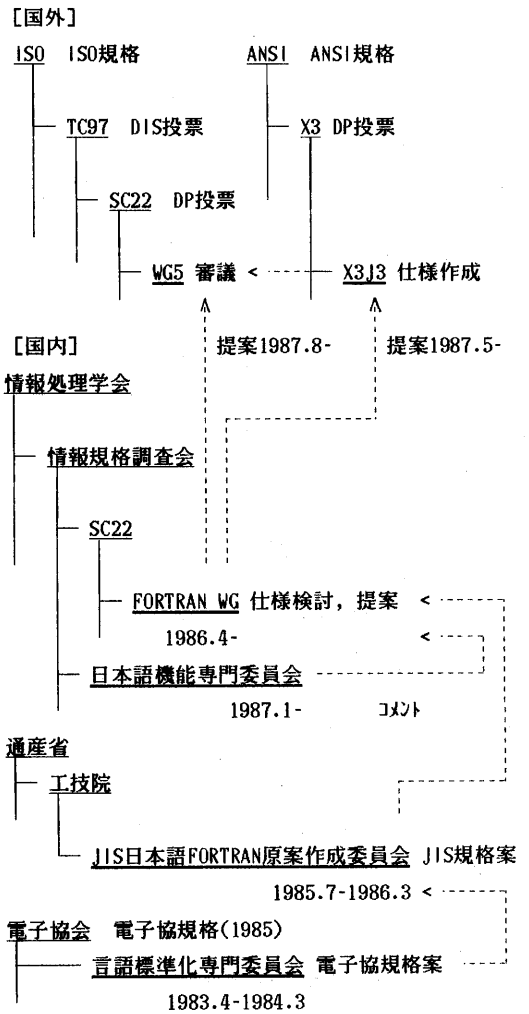

図４　検討体制及び経緯

謝辞

本資料作成に当り，貴重なご意見をいただいた国内ＳＣ２２／ＦＯＲＴＲＡＮ　ＷＧの皆様に深謝いたします．

参考文献

1)電子協：日本語FORTRAN JEIDA-42-1985，(1985).

2)ANSI/X3J3：FORTRAN8X S8.104，(1987).

3)国内SC22/FORTRAN WG：Japan's Proposal to FORTRAN 8X，SC22/WG5リバプール会議資料，(1987).

4)国内SC22/FORTRAN WG：Japanese Proposal to FORTRAN8X--NCHARACTER type for National Character Handling--，ANSI/X3J3 106会議資料，(1987).