

FORTRAN77プログラムをF-BASICプログラムに変換する方法

小寺 富士*, 杉本 敏司, 石川 俊英

大阪大学大学院工学研究科附属超高温理工学研究施設 〒 565-0871 吹田市山田丘 2-1

*e-mail: koderam@ppl.eng.osaka-u.ac.jp

(Received: June 7, 1999 ; Accepted for publication: September 2, 1999 ; Published on Web: November 1, 1999)

FORTRAN プログラムは、物理化学の広い分野の計算に利用されてきたため膨大なプログラムの蓄積がある。その資源をパソコン上で有効に活用していくため、WINDOWS 上で動作する F-BASIC (富士通(株)社の BASIC) に変換することを試みた。F-BASIC に変換された短い簡単なプログラムは、おおむね手直しせずにそのまま正常動作するのが確認された。

キーワード: File Converter, Fortran77, F-Basic

1 はじめに

パソコンの性能は、年々向上してきているためセンターの大型計算機を使用せずとも通常の数値計算はかなりできるようになってきている。

FORTRAN のプログラム開発の歴史は古く、今までに基本的計算に必要とされるプログラムの膨大な蓄積がある。コンパイラが安く入手できれば、これを個人のパソコンで十分活用することができるが、未だ、比較的高価なためパソコンでこれらのプログラムを自分の計算目的に改変して計算するには、困難な状況にあると思われる。できれば、改めてプログラムを作成せずに少ない労力で、自分のパソコンでかなりの計算ができることを望んでいる人も多いと思う。そこで、WINDOWS 上で動作して、従来からの BASIC 資産をそのまま継承して価格も手頃な、富士通(株)の F-BASIC[1, 2] (以後、FB とする) に我々は着目して、これを使用できるように FORTRAN プログラムを BASIC に変換するファイルの開発を試みた。

ファイル変換の研究は、古くから行われているが [3]、FORTRAN から BASIC への変換は、今まで必要度が少ないためかインターネット上で調べても見つからなかった。しかし、BASIC プログラムも今までかなり多数作られており、変換したプログラムと他の BASIC で書かれたプログラムとを併用すれば、かなりの分野の計算が楽にできるようになる。それには、FB と N88-BASIC との併用の可能性を文献 [1] を参照することによって納得が得られる。

BASIC の一つの弱点と言われている所は、組み込み関数が少なすぎることと、複素数が扱えないことである。FORTRAN にあるすべての機能を全部解決することは、我々の能力だけでは

無理なので、今回は組み込み関数のみを対象にした。それでも FORTRAN の組み込み関数に対応したものすべてを作成するには、かなりの数のものを作らなくてはならない。そこで必要度の高いものだけを作ることにした。BASIC の良い点は、計算結果を直ちに図表化するのに便利なことであろう。もちろん、FORTRAN にも図形化処理可能なソフトウェアもあるが [4]、誰にも簡単に利用できるとは言えない。

2 プログラム作成での留意点

BASIC プログラムは元々 FORTRAN をベースに開発されたものであるから、共通するコードもかなり含まれている。そのため、変換はそれほど複雑とは言えない。しかし、FORTRAN プログラムは初期の頃のものとは較べるとかなり進化してきている。いくつかの本 [5-8] を調べてみて分かるように、必ずしも統一したコードでは書かれていない。同じ機能を持ったものでも、例えば、WRITE 文一つ取り上げても使用形態が少しずつ異なっている。それらの変化したコマンドの使用形態に対応できるようにプログラム化を進めた。

プログラム作成前にコンパイラの仕組みを理解し、その手法を用いてプログラム化を進めることが重要だと思われた [9]。両者の基本的な相違点は、FORMAT 文の書き方にある。この部分の変換にかなりのステップ数を要した。EXTERNAL 文は、FORTRAN 独特の機能でこれを FB に変換するには、やや煩雑な手続きを必要とした。それでも完全なものを作成するのは、かなり困難であった。FB には、三角関数としては SIN(), COS(), TAN(), ATN() の 4 種類しか用意されていない。そこで、COSH(), SINH() や、倍精度データに対応する関数はなかったもので、これらに加えて DSIN(), DCOSH() などを新たに作成した。FB には、プログラムの頭のところでサブルーチンとファンクション文は、DECLARE 文で宣言する必要がある、これをしないとエラーとなる。また、サブルーチン内の引数は、配列を含む変数を用いたときは、A() のようにカッコを挿入する必要がある。さらに、配列変数 A() の型が倍精度の場合は、A#() のようなコードを挿入する必要がある。FORTRAN 側で、これが明示されていない場合もあるから、そのときは単精度扱いと判断するようにした。FB への変換のとき、グローバル変数は、プログラムの頭の部分で、COMMON SHARED 宣言をする必要がある。配列変数でも同様にこれを宣言する。MicroSoft 社の QuickBASIC では、この部分を DIM SHARED と宣言しなければならないのと少し異なる。使用制限事項については、5 項で述べる。

3 使用言語と機種性能

使用言語：富士通 F-BASIC Ver.4.1
機種 WINDOWS95/98 対応機種

4 使用方法

1. 我々の作成した組み込み関数のファイル、EXT_FUNC.BAS は、予めライブラリーとして登録しておく。登録の仕方は、FB のマニュアルにしたがって行う。拡張組み込み関数の

DSIN()等を特に使用しない場合は登録しなくてもよい。使用したい関数のみ切り取って自分のプログラムに連結して使う方法もある。

2. FBは、FORTRANのように先に翻訳してから実行させるコンパイラ方式になっているので、BASICのインタープリタ方式に慣れている人には、少し煩わしい所である。また、翻訳後、沢山のファイルが作られてくるためフロッピーに入りきらない事がある。そこで、変換ファイルのFORT_FB.BASをFBの中のディレクトリに¥FBWIN41¥TEMPを作成してその中にインストールしてから翻訳を開始するとよい。
3. ファイルの翻訳が終了したら実行ボタンを押す。最初に計算結果の出力先をスクリーンかプリンタか聞いてくるのでどちらかを選択する。次に、変換用ファイルの入っているフロッピーのドライブ番号を聞いてくるので、例えば、AドライブならAのように入力してリターンキーを押す。
4. 次に、変換するファイル名を入力する。例えば、TEST.FORのように入力してリターンキーを押す。変換されたファイルは、自動的に拡張子にBASが付加されて、TEST.BASになる。但し、変換用ファイルを予めハードデスクの適当なディレクトリに格納しておく方が途中で打ち切りにならず安全である。そのときは、そのディレクトリ名もファイル名の前に付けること。例えば、¥TEMP¥TEST.FORのように入力する。
5. 後は、自動的に変換を行ってくれるから終了するまで待つだけだが、FBの問題点は、ファイルの上書きができない点である。そのため予めエラー処理を行っている。変換に失敗したファイルでもそのまま残しておいて、同じファイルを処理しても大丈夫である。

参考のために、Figure 1に変換前のリスト(左側)と変換後のリスト(右側)図を示した。変換用のサンプルプログラムは、FORTRANは、¥F_SAMPLEの中に、BASICは、¥B_SAMPLEに10例程入れている。正確に変換されていないケースもこの中に含まれていて今後の解決課題として示した。例えば、TEST1.FORは、WRITE文の種々のケースでの変換を試みたものである。しかし、ここに現れなかった特殊な例もあると思うので、それら総てが変換に成功するとは言えない。その他のテストサンプルについても確認してから自分のプログラムを変換してみるのが賢明である。サンプルプログラムの中で未整備な点は、ラベルナンバーがラインナンバー順に付けているため不連続になっているところである。気になる場合は後で変更するとよい。また、元のプログラムでDOループが深くなったとき、ラインに段差を付けているのもあるが、段差を付けるように整備されていないため、変換されたプログラムは少し見にくい。

5 使用制限

プログラム作成の出発時点では、変換は簡単にできると思っていたが、変換プログラムは、試行錯誤を繰り返す間にプログラムの長さが5000行を越えてもバグが取れない状態となった。したがって、どんなFORTRANプログラムにも対応できるほど、このファイル変換のレベルは高いとは言えない。そこで、種々の制限を設けているので注意して使う必要がある。

前述したサンプルプログラムを参照すれば、自分のプログラムの変換が上手くいくかどうかの判断は一応できる。FORTRANプログラムでもメーカーの機種に依存した特殊なコマンドを使っている場合もあり、ケースバイケースでかなり異なってくるため比較表を作成しても簡単なケースのみでは参考にならないためここでは省略する。

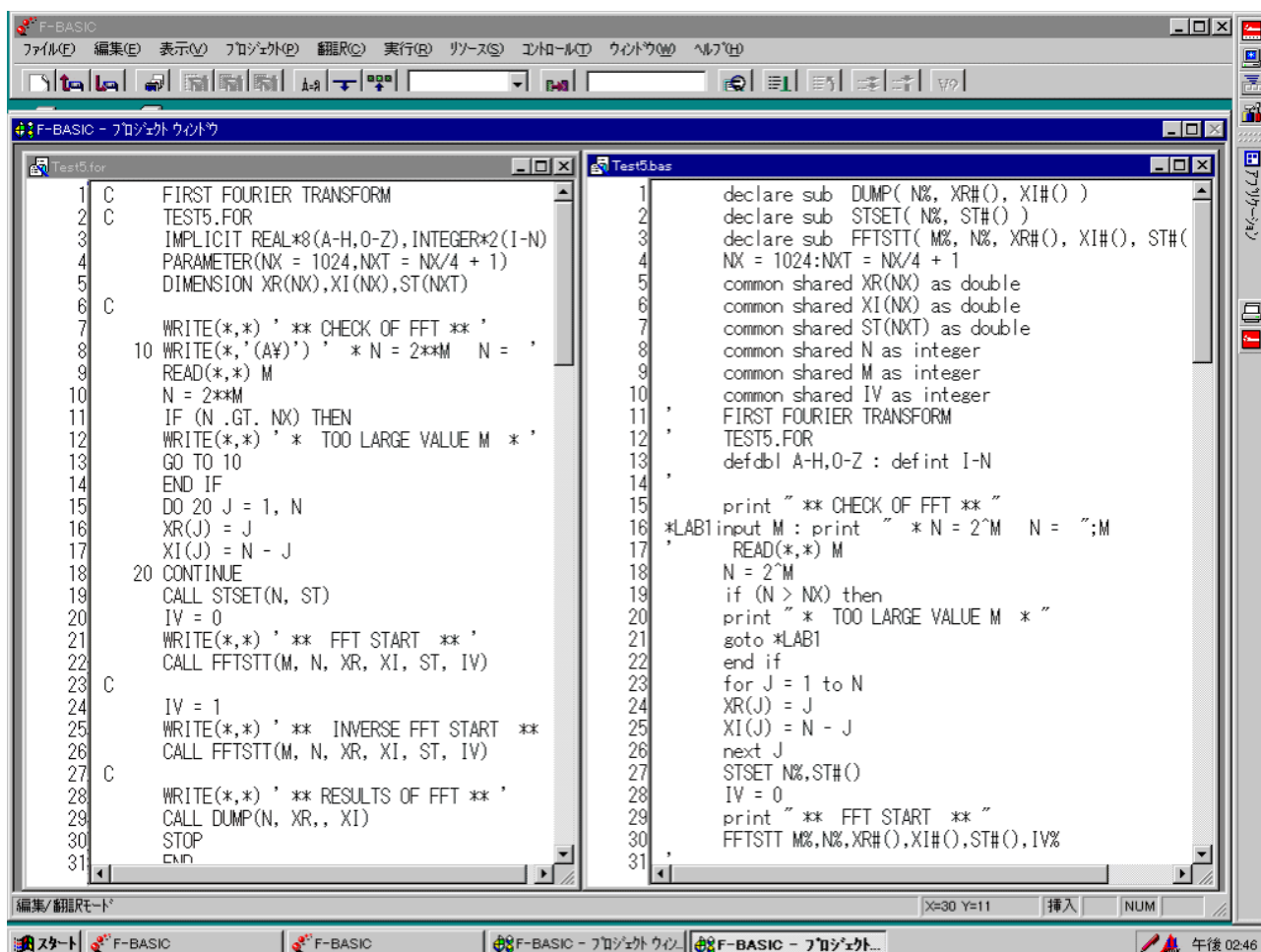


Figure 1. Sample of file conversion; left is a source program, right is a converted one.

1. 複素数を扱うプログラムは現時点では変換できない。
2. IMPLICIT 宣言文で注意すべき点は、次のようなケースである。

IMPLICIT INTEGER (I - N)

IMPLICIT REAL (A - H)

IMPLICIT REAL (O - Z)

のように、三種類に宣言を分けているが、このような場合は変換を途中で中止してしまうようにしている。従って、元の FORTRAN プログラムを纏めて

IMPLICIT INTEGER (I - N)

IMPLICIT REAL (A - H), (O - Z)

のように二行に宣言文を予め書き換えてから変換を行う。

次に、IMPLICIT REAL (A - H), (O - Z) のみを宣言しているプログラムでサブルーチンの引数の変数に INTEGER 型のものを使用するときは、IMPLICIT INTEGER (I - N) も宣言しておく。そうしないと誤変換してしまう。

3. 配列変数の次元は二次元までしか扱えない。これは、通常の計算では、三次元以上のプログラムはあまり見かけないため、現在のところそこまで拡張していない。計算そのものは

問題ないが、入出力方法が複雑になってしまうためである。

4. EXTERNAL 文は、簡単なケースしかできない。メインとサブルーチンの両方の中で呼び出して使うようなことは現在のところ考慮していない。
5. 組み込み関数は FB は基本的に 4 種類しか用意されていない。これを 34 種類までに拡張した。FORTRAN のコード名と同じ名称を使用して混乱を避けた。
そのリストを以下に掲げると、
ASIN(X), ACOS(X), ACOT(X), ASINH(X), ACOSH(X), ATANH(X), AMAX1(X),
AMIN1(X), ALOG10(X), SINH(X), COSH(X), TANH(X), COTH(X), DSIN(X), DCOS(X),
DTAN(X), DASIN(X), DACOS(X), DATAN(X), DSINH(X), DCOSH(X), DTANH(X),
DCOTH(X), DABS(X), DEXP(X), DFLOAT(X), DLOG(X), DLOG10(X), DSQRT(X),
DMAX1(X), DMIN1(X), FLOAT(X), MAX0(X), MIN0(X) である。
6. 拡張 FORTRAN で使用している DO WHILE...ENDDO には対応しているが、そのほかの FORTRAN90 や M S -FORTRAN で新たに付加されたコードには、全く対応していない。
7. FORTRAN 文を小文字で書いたプログラムは大文字に変換してから行っている。必要なければ、UCASE\$をはずして使用する。但し、変換エラーの発生の確認は行っていない。
8. WRITE 文の FORMAT の項で制御コードを変数で代用しているものには対応していない。

できない例) FORMA='I3,X2,I2'

```
WRITE(6,FORMA) I,J
```

一部対応したケースとしてサンプルプログラムの TEST5.FOR がある。しかし、FB にないような機能外のコードには、現在のところ全く対応させていない。

9. FORMAT 文の 1H1 等の改頁コードも省資源の立場から対応させていない。
10. READ 文は FORTRAN では、FORMAT 文でデータの入力の規制はできるが、FB では FORMAT 文は無視される。実際の例は、TEST2.FOR の中に種々のケースを載せている。
11. サブルーチン内に DATA 文があるときは、簡単なケースのみ対応させている。
可能な例 : DATA PAI/3.1415/ -> PAI = 3.1415(TEST6.FOR) としている。
12. FB に変換後のサブルーチンの引数は全部グローバル変数になっている。
13. 前に FB に変換したプログラムを他の目的に再使用するときは、サブルーチン内のラベル名が同じになるケースがでてくる恐れがあるので、そのときは手直しして使うこと。
14. EQUIVALENCE 文と ENTRY 文も考慮されていない。メモリーの少なかった時代の遺産であるこのコマンドを使用しているときは、予め、元のプログラムを変更しておく。
15. 拡張 FORTRAN では、FUNCTION 名を = の右側にも書くことができ和算等に使用しているが、BASIC ではこの機能は使えない。そこで、他の変数に変えて後で戻すようにしている。(参考 : TEST4.FOR)
16. 名前付き COMMON 文が使われている場合で注意しなければならない点は、メインとサブルーチン間で同じ名前の COMMON 文が使用されているときは、メインで定義したものと同一変数名を使用するようにプログラムを変える必要がある。

例) メイン : COMMON NAM1/A,B(10)/NAM2/G,H のようなケースでは、
サブルーチン内 : COMMON NAM1/C,D(10) となっているとエラーとなる。そこで、
サブルーチン内 : COMMON NAM1/A,B(10) と書き換える。

その中で使用されている変数名も C は A に変える必要がある。あるいは、メインで、予め変数 A と B() の内容を C と D() にそれぞれ移しておくプログラムを挿入するかの何れかが必要になる。

17. 拡張 FORTRAN では、DATA 文の中で配列変数を用いて、データの初期化を以下の例のようにプログラムを書くことができる。

例) DIMENSION A(10)

DATA A/10*1.0/

変換ファイルで同じ処理を行うには、2 行目を

DATA (A(I), I = 1,10)/10*1.0/

と古い FORTRAN プログラム形式に書き直さないとデータの初期化はできない。

以上で述べたように、現在のところ数多くの変換に不適切な箇所が散見される状況にあるが、今後、さらに改良を進めていかねばならないと考えている。

6 F-BASIC と QuickBasic との相違点

上記のソフトウェアを QuickBasic 上でも動作させるようにするために、両者の違いについて若干述べたい。

FB と Microsoft 社の QuickBASIC(QB) との相違点は、両社が提携して開発を行っているため使用コードの違いはそれ程大きいとは言えない。前者は、コードをさらに進化させた Visual BASIC に近いものを用いている。

FB は、コマンド類は小文字を使用している。従って、ここで作成した FORT_FB.BAS を UCASE\$ コマンドで全部大文字に変えてやる。次に、サブルーチンを呼ぶときは、QB では、CALL 文を使用しているから CALL を挿入する。また、引数があるときは、括弧で両端を括ってやる必要がある。FB の LOCATE 文は、(X,Y) 座標表示であるが、QB では、(Y,X) 座標表示になっているのでこれを交換する必要がある。また、FB では、GOTO 文の行き先アドレスの先頭に従来の習慣で * が付いているので QB では、これを取ってやり、更に、後尾にコロンを付ける。GOTO 文の行き先アドレスも同様に変える。FB の TRIM\$ コマンドは使用できない。QB では、RTRIM\$ と LTRIM\$ の 2 回の操作を必要とする。

ファイルの上書きは、QB では可能であるが、FB ではできないためファイルをオープンする前に KILL コマンドで不要なファイルを消去しているが、この箇所は不要なので消してもよい。

その他、グローバル配列変数定義に FB では COMMON 文を使用しているが、QB は、これを DIM 文に置き換えてやる。FB で多数使用している記号で _ は使用できない。QB では、ピリオドと交換する等の処置をする。サブルーチンの中で括弧を使用している箇所や、コロンを使用している箇所が、何カ所かあるのでそこを修正すれば正常に動作するはずである。これ以上の詳細は、ここでは省略する。

7 おわりに

どんなプログラムであれ、未発表のプログラムを開発するときは、一度で完全なものを作ることはなかなか難しい。このことは、一度経験したことのある人には理解して頂けると思う。今回、あえて発表するのも LINUX の例でも明らかなように、多くの人の批判に曝されて、早く改良が進められればよいと考えたからである。このプログラムの欠点や改良点をどんどんご指摘いただけることを期待している。

参考文献

- [1] カンノコウキ, *F-BASIC97* プログラミングのすべて, 電波新聞社 (1998).
- [2] 高城重道, 緒方正幸, *WINDOWS BASIC* プログラミング (1), パワー社 (1998).
- [3] John R. Wolberg, 岸 和孝訳, ソフトウェア変換ハンドブック, 啓学出版 (1988).
- [4] 黒瀬能幸, 松島勇雄, 松尾俊彦, *FORTRAN77* による科学技術計算サブルーチンライブラリ, 啓学出版 (1988).
- [5] 徳田 昭, *FORTRAN77*, 森北出版 (1989).
- [6] 町田東一, 小島紀男, *FORTRAN* 応用数値計算, 東海大学出版 (1989).
- [7] 小暮 仁, *FORTRAN77* の応用, 産業図書 (1992).
- [8] 橋本英美, *FORTRAN77* プログラム入門, 技術評論社 (1992).
- [9] 渡邊 坦, コンパイラーの仕組み, 朝倉書店 (1998).

Development of a Program Converter from FORTRAN 77 to F-BASIC

Fuji KODERA*, Satoshi SUGIMOTO and Toshihide ISHIKAWA

Plasma Physics Laboratory, Graduate School of Engineering, Osaka University
Yamada-oka 2-1, Suita City, Osaka, 565-0871 Japan

**e-mail: koder@ppl.eng.osaka-u.ac.jp*

It has been recognized that most physical chemistry calculations are conveniently performed on desk top computers with a program code written in BASIC languages. The compilers for this purpose are considerably cheaper than FORTRAN ones. We have developed a program converter from FORTRAN 77 to F-BASIC(Fuji-tsuLtd.) on WINDOWS95/98 (Microsoft Corp.). It was confirmed that some short simple FORTRAN programs are converted with our method to give satisfactory results without any manual modification. These converted programs were able to run with some other BASIC environments such as Quick BASIC and N88-BASIC following some modifications.

Keywords: File Converter, Fortran77, F-Basic