

8.1.1

Conversational FORTRAN Comparison Table

FORTRAN FEATURES	<u>XDS 940</u> TYMCOM IX TYMSHARE	<u>RDP-10</u> TYMCOM X VENDOR FORTRAN INFORMATION
<b>A. CONSTANTS</b>		
1. Integer	$+2^{23}$	$+2^{35}$
2. Real	$+10^{75}$ , 11 digits	$+10^{38}$ , (8) digits
3. Double precision	$+10^{75}$ , 17 digits	$+10^{38}$ , (16) digits
4. Complex	Two reals	Two reals
* 5. Logical constants	.TRUE.=1, .FALSE=0	T=-1, F=0
6. String constants	YES	YES
7. Hollerith constants	NO	YES
8. Octal	NO	YES
<b>B. VARIABLES</b>		
* 1. Names	31 CHARS.	6 CHARS.
2. Array dimension	ANY NUMBER	SAME
3. Array bounds	A(L:H)	SAME A(L/H)
* 4. Adjustable array dimension	A(*,*)	A(1,1) A(N,M)
5. Adjust to accommodate data in Data Statements	NO	NO
6. Dynamic dimensioning during execution	NO	NO

C. EXPRESSIONS

1. Mixed mode	Mode conversion by subexpression	YES
2. Logical expressions	YES	YES
3. Arithmetic expression hierarchy	for**, UNARY-, *or/, +or-	SAME
4. Typeless expressions	NO	NO

D. CONTROL STATEMENTS

1. GO TO (Normal, computed, and assigned)	YES	YES
2. Arithmetic IF	YES	YES
3. Variable statement #'s in Arithmetic IF	NO	YES
4. Omitted statement #'s in Arithmetic IF	NO	YES
5. Logical IF	YES	YES
X 6. .ELSE. clause	YES	NO
7. Loops	Parameters any type	YES
8. Count modifiable in loop	NO	NO
9. Extended range transfers	YES	YES
X 10. Implied DO loops	YES	NO

E. MULTIPLE STATEMENTS

	YES (WITH;)	YES
1. Logical IF	NO	NO
2. DO end	NO	NO

F. FORMAT

1. FORMAT Specs I, F, E, D, G, L, A, H, P, T, X, J, R, /, (, ', &, S, ", \$	YES	YES
2. M(money), O(octal)	NO	OCTAL

3. Widthless formats	YES	YES
4. Carriage control	NO	YES
5. # of blank records written for n slashes at end of FORMAT	n/	YES
6. Dynamic FORMAT	YES	YES
G. I/O LISTS		
1. Nested implied DO's.	YES	YES
2. Quote strings	YES	YES
3. Constants	YES	YES
4. Expressions	YES	YES
5. More list items than BINARY input record	ERROR	ERROR
H. I/O STATEMENTS		
1. ACCEPT	YES	YES
2. DISPLAY	YES	YES
3. TYPE (format)	NO	YES
4. FREE FORM INPUT	YES	YES
5. FREE FORM OUTPUT	YES	YES
6. END= ERR=	YES	YES
7. Maximum Record Size		640
8. NAMELIST	NO	YES
9. ENCODE/DECODE	USE Strings	YES
I. FILES		
1. OPEN and CLOSE	YES	YES
2. RANDOM files	YES	YES
3. BINARY FILES	YES	YES
4. Maximum size.	3.2 Million Characters	<u>UNLIMITED</u> ?

5. File name	Any legal alphanumeric to 45 chars.	Any alphanumeric 6 primary 3 extension
<b>J. SPECIFICATION STATEMENTS COMMON (labeled)</b>	YES	YES
1. EQUIVALENCE	YES	YES
2. DATA	YES	YES
3. IMPLICIT	NO	YES
<b>K. SUBROUTINES &amp; FUNCTIONS</b>		
1. MULTIPLY ENTRY POINTS	NO	YES
2. Hollerith or string arguments in FUNCTION's	YES-no HOLLERITH	YES
3. Intrinsic functions	YES	NO causes diag.
4. EXTERNAL Statement	YES	
5. Assembly language subroutines and functions	NO	YES
6. RETURN K	NO	YES
<b>L. FORTRAN COMPILER DIRECTIVES</b>		
1. PARAMETER Statement	NO	2nd release
2. INCLUDE Statement	NO	2nd release
3. Conditional compilation	NO	YES
4. FORTRAN MACROS	NO	NO
5. DELETE-CONTROLS source lines for compilation	YES	YES
<b>M. PROGRAM DEBUGGING LISTINGS</b>		
1. Definitions & cross-reference of variables	NOT NEEDED	YES
2. Variable Storage Assignments	NOT NEEDED	YES
3. TRACE	NOT NEEDED	YES

SUPER FORTRAN DIFFERENCES  
A. GENERIC VS. INTRINSIC FUNCTIONS

NOTES

PRELIMINARY

TYMCOM IX	TYMCOM-X	METHOD OF MODIFICATION	
EXP (a)	EXP (a) DEXP (a) CEXP (a)	See TYMCOM X SUPERFORTRAN manual for specific argument descriptions. Conversion Aid will flag; user must change manually.	
LOG (a)	ALOG (a) CLOG (a) DLOG (a)		
LOG10 (a)	ALOG10 (a) DLOG10 (a)		
SIN (a)	SIN (a) CSIN (a) DSIN (a)		
COS (a)	COS (a) CCOS (a) DCOS (a)		
ATAN (a)	ATAN (a) DATAN (a)		
ATAN2 (a1, a2)	ATAN2 (a1, a2) DATAN2 (a1, a2)		
SQRT (a)	SQRT (a) CSQRT (a) DSQRT (a)		
MOD (a1, a2)	MOD (a1, a2) AMOD (a1, a2) DMOD (a1, a2)		
ABS (a)	ABS (a) DABS (a) IABS (a) CABS (a)		

TABLE II (continued)

A

TYMCOM IX	TYMCOM-X	METHOD OF MODIFICATION	NOTES
MAX(a,b,c,...)	AMAXO (a,b,c,...) AMAX1 (a,b,c,...) MAXO (a,b,c,...) MAX1 (a,b,c,...) DMAX1 (a,b,c,...)		
MIN(a,b,c,...)	AMINO (a,b,c,...) AMIN1 (a,b,c,...) MINO (a,b,c,...) MIN1 (a,b,c,...) DMIN1 (a,b,c,...)		
SIGN(a1,a2)	SIGN (a1,a2) ISIGN (a1,a2) DSIGN (a1,a2)		
DIM(a1,a2)	DIM(a1,a2) IDIM(a1,a2)		
INT(a)	INT(a), IDINT(a)		
DBLE(a)	DBLE(a), DFLOAT(a)		

TABL I  
FC

SUPER FORTRAN DIFFERENCES  
B. FUNCTIONS HAVING DIFFERENT NAMES

TYMCOM IX	TYMCOM-X	METHOD OF MODIFICATION	NOTES
TRUNC(a)	AINT(a)	Conversion Aid will flag; must be manually changed by user.	
REAL(a)	FLOAT(a) SINGL(a)		
ARSIN(a) ARCOS(a) FIX(a)	ASIN(a) ACOS(a) IFIX(a)		





TABLE II

SUPER FORTRAN DIFFERENCES

D. FUNCTIONS TO BE HANDLED DIFFERENTLY

TYMCOM-IX	TYMCOM-X	METHOD OF MODIFICATION	NOTES
CHAR(a)	Arguments have different table values	Conversion Aid will flag; must be changed by the user. See SUPERFORTRAN X Manual. User must modify.	
String Function does not have to be declared	Must declare String Function		
SUBSTR3 returns string when the parameters exceed declarations	Returns null string when parameters exceed declarations		

SUPER FORTRAN DIFFERENCES  
E. SPECIFICATION DIFFERENCES

SPECIFICATION	TYMCOM-IX	TYMCOM-X	METHOD OF MODIFICATION	NOTES
a) REAL NUMBER MAGNITUDE AND PRECISION	±10 <sup>75</sup> 11 digits accuracy	±10 <sup>38</sup> 8 digits accuracy	Program changes may be necessary.	
b) DOUBLE PRECISION NUMBER SIZE	±10 <sup>75</sup> 17 digits accuracy	±10 <sup>38</sup> 16 digits accuracy	"	
c) LOGICAL CONSTANTS	.TRUE. = 1 .FALSE. = 0	.TRUE. = -1 .FALSE. = 0	Program logic may require change.	
d) VARIABLE NAME SIZE	31 characters	6 characters	Flagged by Conversion Aid. User must modify.	
e) FILE NAME SIZE	Up to 45 char.	Up to 6 characters + 3 extension characters with "." separation alphanumeric only.	User must rename files at time of file transfer to conform to TYMCOM X requirements.	
f) ALPHANUMERIC VARIABLES (A FORMAT)	3 characters per word	5 characters per word	If necessary, user must modify. Documented in SFO Conversion Aid document.	



TABLE II  
F. (continued)

OPERATION	TYMCOM-IX	TYMCOM X	COMMENTS	NOTES
Saving symbolics	<p>&gt; SAVE filename TEXT only? <u>Y</u> New File</p>	<p>&gt; SAVE filename NEW file-OK? Y</p>		
Saving binary	<p>&gt; SAVE filename text only? <u>N</u></p>	<p>&gt; LOAD filename &gt; SAVE filename NEW FILE-OK? Y</p>		
Entering	<p>&gt; 10(10)100 &gt; 10:20 &gt; LIST *1,\$</p>	<p>&gt; ENTER 10:20 &gt; LIST #1,\$</p>		
Line addressing				
Making GO files (from EXEC)	<p>-MAKEGO LINK FILE:binary filename NEW FILE</p>	<p>-LOAD filename -SAVE gofilename</p>	<p>The executive LOAD command on the TYMCOM X will use binary or symbolic files. MAKEGO requires a binary file created with the binary SAVE command on TYMCOM IX in SFO.</p>	

TABLE II

## SUPER FORTRAN DIFFERENCES

## G. SYNTAX DIFFERENCES

SYNTAX	TYMCOM-IX	TYMCOM-X	METHOD OF MODIFICATION	NOTES
7) SYNTAX				
a) Square brackets	Allowed	Not allowed	User must modify. In EDITOR, use SUBSTITUTE command.	
b) Line feed as a line continuation	"	"	Modified by @SFAID4	
c) Allows C: to begin comment lines	"	! or * must be first non-blank character on a line.	Modified by @SFAID4	
d) Automatic array declaration in subprograms	A(*), AO(*,*) (* allowed)	A(1), AO(M,N) (* not allowed)	Flagged by @SFAID5 User must correct.	
e) IF ELSE	available	.ELSE. not available	Modified by @SFAID4	
f) Implied DO loops other than in I/O statements	Allowed	Not allowed	Flagged by @SFAID3 User must modify.	
g) Multiple statements after IF statements	"	"	Modified by @SFAID4	
h) COMPLEX/DOUBLE PRECISION mixed mode expressions	"	"	User must modify. See discussion in Conversion Aids.	

SYNTAX	TYMCOM-IX	TYMCOM-X	METHOD OF MODIFICATION	NOTES
i) Location of statement functions and declaration statement position	Anywhere	Before first executable statement	Compilation errors will indicate. User must then correct.	
j) Random access file position indicator in OPEN statements and I/O statements	The position specification is located after format label.	The position specification is located after file number.	@SFAID2 will flag. User must modify.	
k) COMMON blocks	A COMMON block can have different length in separate subroutine.	The COMMON block length in the subroutine must not be greater in size than the calling program.		
l) Exclusive OR	.EOR.	.XOR.	Flagged by @SFAID5. User must modify.	
m) CLOSE	CLOSE (filename)	Not available	User must modify.	
n) CANCEL		CANCEL(n) n=number or integer variable	Will not update file.	

TABLE II

SUPERFORTRAN DIFFERENCES

H. MISCELLANEOUS DIFFERENCES

OPERATION	TYMCOM IX	TYMCOM X	METHOD OF MODIFICATION	NOTES
a) Link statement	Yes	No	Flagged by @SFAID3. User must modify.	
b) Expanding arrays in subroutines	Yes	No	Program changes may be required.	
c) ESCAPE during run	Files automatically closed.	User will be prompted to type CLOSE or CANCEL files. (Cancel will delete output files.)	Consult manual for details.	
d) .IMP. operator	Available	Not Available	Flagged by @SFAID5 User must modify.	
e) Modification of DO loop index within loop.	Allowed	Not allowed	Flagged by @SFAID3 User must modify.	
f) % variable name	Allowed	Not allowed	User must modify.	

TABLE I  
FUNCTIONS FLAGGED BY AID2  
4-28-76

PRELIMINARY

ABS  
AINT  
ARCOS  
ARSIN  
ATAN  
ATAN2  
CATAN  
CCOSH  
CHAR  
COS  
CSINH  
CTAN  
DARCOS  
DARSIN  
DATE  
DBLE  
DIM  
DTAN  
ENTIER  
ERASE  
ESC  
EXEC  
EXP

FIX  
FRACT  
GOFIELD  
INT  
LOG  
LOG10  
MAX  
MIN  
MOD  
POLAR  
REAL  
ROUND  
SETSIZE  
SIGN  
SIGNUM  
SIN  
SQRT  
TCP  
TEL  
TELL  
TRUNC  
WAIT  
YEAR



SUPER FORTRAN CONVERSION STEPS  
FEB. 27, 1976

PRELIMINARY

The following is a list of steps required to convert a TYMCOM IX Super Fortran program to the TYMCOM X. Depending on the Super Fortran program code, some of the steps listed will not be necessary. An asterisk (\*) indicates that compilation of the program on the TYMCOM X may help locate the particular problem area being discussed.

Table II (attached) shows the Super Fortran differences on the two computers and methods for handling them. Also, a Super Fortran Conversion Aids document, (attached), describes the function and usage of each conversion aid.

PART I

- 1) Log into System 9 or 15, wherever the programs to be converted are located.
- 2) Run the program to be converted through (G2CONVERT)@SFAID1. This aid checks for variable names longer than 6 characters. To run AID1, type  
GO (G2CONVERT)@SFAID1

The program will prompt for input file name and will give a filename for the file generated with line numbers containing invalid variable names.

- 3) Enter the program into EDITOR and rename the variables which were flagged by AID1 as greater than 6 characters. Write these changes to a file.

- 4) Running the program through AID2 is optional; this aid flags functions any commands which may be incompatible between the TYMCOM-IX and the TYMCOM-X. Table I shows which items this aid will flag. To run AID2 type

GO (G2CONVERT)@SFAID2,

The program will prompt for an input filename and will print flagged names on the terminal.

- 5) AID3 is also optional; it flags instances where the value of a DO loop index is changed by an assignment within the DO loop. To run AID3, type

GO (G2CONVERT)@SFAID3,

The program will prompt for an input filename and will output to the terminal.

- 6) Run the modified program through (G2CONVERT)@SFAID4. This step must be done to remove line numbers and to accomplish other necessary syntax modifications. See attached Super Fortran Conversion Aids document for a list of items that will be changed.

To run AID4, type

GO (G2CONVERT)@SFAID4,

The program will prompt for input and output file names. The input filename should be the name of the file which contains corrected variable names.

The output file contains TYMCOM-X Super Fortran syntax (i.e. this output file will probably not execute on TYMCOM-IX Super Fortran). NOTE: This output file should be used for all further processing.

7) AID5 flags the following:

- a) LINK statements
- b) Implied DO loops not in I/O statements
- c) Dummy declaration statements (i.e. declarations using \* or \*, \*)
- d) Use of .EOR. or .IMP. logical operators
- e) Illegal TYMCOM-X file names used in OPEN statements

This aid may be omitted if the program contains none of the above.

To run AID5, type

GO (G2CONVERT)@SFAID5

The program will prompt for an input filename which should be the name of the output file resulting from AID4 processing. Only differences are flagged and printed at the terminal; no output file is generated.

8) The statements flagged by aids may be corrected at this point. See Table II for methods of correction. Items 11 through 24 in Part II of this document also address these needed changes, some of which will become more apparent after compilation on the TYMCOM-X.

9) Copy the modified file to your TYMCOM X account. This can be done by TELECOPY on System 9 or System 15 if the file is small (50,000 characters), by having operations write the file to tape via a Batch Request.

To obtain instructions for running TELECOPY, type

-TELECOPY

:HELP

4

Instructions for magnetic tape transfer of files via Batch Request from the TYMCOM-IX to LMSC's TYMCOM-X (System 88) are found in ~~Timesharing Services Bulletin Vol. II, #2.~~ TIP #1

- 10) If required, run the program through (G2CONVERT)@SFAID6 on System 88. AID6 breaks long lines (up to 256 characters) into shorter lines with the user specifying the length of the shorter line. This is necessary to break up long comment statements which result from the removal of line feeds (a previously used aid, AID4, has removed line feeds).

To run AID6, type

GO (G2CONVERT)@SFAID6

The program will prompt for input and output file names. The input file name should be the name of the file copied from the TYMCOM-IX. The user will also be prompted for desired line length.

- 11) Enter SFO on System 88. Load the program into SFO by the command:

ENTER 1:10000 FROM filename

where 1:10000 gives the line range of your program, and filename is the latest modification of the program being converted.

## PART II

Correct the statements flagged by any of the conversion aids. The methods of handling flagged statements are described in the following items.

NOTE: ALL OF THE FOLLOWING CONVERSION STEPS NEED NOT BE ACCOMPLISHED IN THE ORDER LISTED. HOWEVER, THOSE ITEMS PRECEDED BY AN ASTERISK ARE MORE EASILY DETECTED AFTER THE PROGRAM HAS BEEN COMPILED.

If EDITOR is used to correct any of these items, line numbers are stripped. After EDITOR has been used, the program must be reentered into SFO as shown in item 10.

- 12) Compile the program.
- 13) Make sure the first six characters of filenames are unique.
- 14) Make sure that the main program includes all common blocks used in all subprograms. Also note that if a common block varies in size, the longest reference must be in the main program.

example a)

	<u>illegal</u>		<u>legal</u>
C	MAIN PROGRAM	C	MAIN PROGRAM
	⋮		⋮
	COMMON/A/X(100)		COMMON/A/X(100)
	⋮		COMMON/B/Y(20)
	CALL SUBA		CALL SUBA
	⋮		⋮
	SUBROUTINE SUBA		SUBROUTINE SUBA
	⋮		⋮
	COMMON/A/X(100)		COMMON/A/X(100)
	⋮		⋮
	COMMON/B/Y(20)		COMMON/B/Y(20)
	⋮		⋮
	CALL SUBB		CALL SUBB
	⋮		⋮
	SUBROUTINE SUBB		SUBROUTINE SUBB
	⋮		⋮
	COMMON/B/Y(20)		COMMON/B/Y(20)

example b)

illegal

```
C MAIN PROGRAM
  COMMON/C/A(10),B(10)
  .
  .
  CALL SUBC
  .
  .
  SUBROUTINE SUBC
  COMMON/C/A(10),B(10),X(20)
```

legal

```
C MAIN PROGRAM
  COMMON/C/A(10),B(10),X(20)
  .
  .
  CALL SUBC
  .
  .
  SUBROUTINE SUBC
  COMMON/C/A(10),B(10),X(20)
```

- 15) Check for modification of the index of a DO loop within the loop.

The example

```
Do 10 I=1,10
  .
  .
  I = 15
  .
  .
10 CONTINUE
```

is invalid and must be modified.

- 16) Change generic functions to explicit functions. Generic functions on the TYMCOM-IX will accept arguments of various types, while explicit functions on the TYMCOM-X require that the function type be the same as the argument type. For example, the SIN function on the TYMCOM-IX will accept single precision, double precision or complex arguments. On the TYMCOM-X, SIN, DSIN or CSIN must be used to match the type of the argument.

If any of the functions listed below were flagged by AID2, determine the type of the argument(s) and then decide which functions to use from Tables II and ~~III~~. *SFO manual, pages 32-36.*

EXP	SQRT	INT
LOG	MOD	DBLE
LOG10	ABS	
SIN	MAX	
COS	MIN	
ATAN	SIGN	
ATAN2	DIM	

- \*17) Change the names of functions that have different names on the TYMCOM-IX and the TYMCOM-X. These were flagged by AID2. Functions requiring name changes are listed below. See Table II for methods of modifications.

TYMCOM IX

TRUNC

TYMCOM X

AINI

- \*18) The following functions will be flagged by AID2. These capabilities do not currently exist on the TYMCOM-X. Work is in process to provide these functions, so this list may be modified at any time. Consult Timesharing Services for current status of these items and suggested methods or alternatives.

TAN  
 ENTIER  
 SIGNUM  
 POLAR  
 ROUND  
 FRACT  
 ERASE  
 TEL  
 WAIT  
 EXEC

- \*19) Check subscript range statement and change to the proper TYMCOM-X syntax.  
An example is given below.

TYMCOM-IX: DIMENSION A(20:30)

TYMCOM-X: DIMENSION A(20/30)

- \*20) Correct IF or IF-ELSE statements found in error by the compiler.
- \*21) Correct multiple statements after IF statements that were flagged by the compiler.
- \*22) Move function statements and declaration statements above the first executable statements in the main program and all subprograms.
- \*23) Delete blank lines that were not deleted by the conversion program.
- \*24) If random access files are used, change respective READ or WRITE statement syntax when file position is indicated.

example: TYMCOM-IX: Read (6,3000) (I+15) Array  
TYMCOM-X: Read (6#I+15,3000) Array

- \*25) Change square brackets to parentheses. This can be done in EDITOR using the SUBSTITUTE command.
- 26) Save the updated file.
- 27) Run the program and compare results with a TYMCOM-IX run.
- 28) Correct any errors and document TYMCOM-X program.



## QSFAD1

### 1. FUNCTION

QSFAD1 IS A PROGRAM WHICH FLAGS VARIABLE NAMES WITHIN A TYMCOM IX SFD PROGRAM WHICH ARE IN EXCESS OF 6 CHARACTERS IN LENGTH.

QSFAD1 DOES NOT AUTOMATICALLY CONVERT THE FLAGGED VARIABLE NAMES TO NEW NAMES. HOWEVER, QSFAD1 DOES GENERATE A LISTING OF ALL LINES WHICH CONTAIN THE VARIABLE NAMES WHICH ARE IN EXCESS OF 6 CHARACTERS.

PLEASE NOTE THAT THE ONLY VARIABLES WHICH WILL BE AMBIGUOUS TO THE TYMCOM X SFD COMPILER ARE VARIABLE NAMES WHICH ARE IDENTICAL IN THE FIRST 6 CHARACTERS OF THE VARIABLE NAME.

FOR EXAMPLE:

THE FOLLOWING VARIABLE NAMES COULD CAUSE UNFORSEEN PROBLEMS IN TYMCOM X SFD:

PROJECT1  
PROJECT2

HOWEVER, THE FOLLOWING VARIABLE NAMES WOULD NOT CAUSE ANY PROBLEMS OF AMBIGUITY IN TYMCOM X SFD:

PROJ1FLAG  
PROJ2FLAG

### 2. RULES

THE INPUT FILE TO QSFAD1 MUST BE A TYMCOM IX SFD PROGRAM WHICH HAS BEEN SAVED IN SYMBOLIC FORM ONLY AND WHICH CONTAINS THE SFD NUMBERS. (I.E. A FILE WHICH DOES NOT CONTAIN SFD NUMBERS WILL NOT BE PROPERLY ANALYZED).

### 3. LIMITATIONS

VARIABLE NAMES BEGINNING WITH THE FOLLOWING SETS OF CHARACTERS WILL NOT BE PROPERLY DETECTED:

CALL  
ACCEPT  
DISPLAY  
RANDOUT  
SUBSTR3  
POSITION

SYMBOLIC

GOTO (FOLLOWED BY A DIGIT. E.G.: GOTO1FLAG)  
DO (FOLLOWED BY A DIGIT. E.G.: DO1FLAG)

FOR EXAMPLE:

THE FOLLOWING VARIABLE NAMES WILL BE DETECTED:

AAABBECCC  
GOTOXXXX  
DOYZABCDE

THE FOLLOWING VARIABLE NAMES WILL NOT BE PROPERLY DETECTED:

CALLFLAG  
DISPLAYARG  
GOTO1XYZ  
DO7FLAGVAR

4. SAMPLE RUN

## @SFAID2

### 1. FUNCTION

@SFAID2 IS A PROGRAM WHICH FLAGS CERTAIN SFD FUNCTIONS AND COMMANDS WHICH MAY BE INCOMPATIBLE BETWEEN THE TYMCOM IX AND THE TYMCOM X.

THE SPECIFIC FUNCTIONS AND COMMANDS WHICH ARE FLAGGED ARE CONTAINED IN THE FILE NAMED (G2CONVERT)@SFAID2TABLE. A LISTING OF THIS TABLE AS OF 2-1-76 IS ATTACHED.

@SFAID2 DOES NOT ATTEMPT TO GENERATE NEW SOURCE CODE; IT MERELY FLAGS SUSPECTED INCOMPATIBILITIES.

IF THE AID ENCOUNTERS AN OPEN STATEMENT FOR A RANDOM FILE A SECOND PASS IS MADE THRU THE FILE AND ALL READ OR WRITE STATEMENTS REQUIRING A SYNTAX CHANGE ARE FLAGGED.

### 2. RULES

THE INPUT FILE TO @SFAID2 MUST BE A TYMCOM IX SFD PROGRAM WHICH HAS BEEN SAVED IN SYMBOLIC FORM ONLY AND WHICH CONTAINS THE SFD LINE NUMBERS. (I.E. A FILE WHICH DOES NOT CONTAIN SFD LINE NUMBERS WILL NOT BE PROPERLY ANALYZED).

### 3. LIMITATIONS

NO LIMITATIONS ARE KNOWN AT THIS TIME.

### 4. SAMPLE RUN

## QSFID3

### 1. FUNCTION

QSFID3 DO LOOP INDEX MODIFICATION VIA AN ASSIGNMENT STATEMENT CONTAINED WITHIN THE LOOP.

QSFID3 DOES NOT GENERATE NEW SOURCE CODE; IT MERELY FLAGS THE OCCURRENCES.

### 2. RULES

THE INPUT FILE TO QSFID3 MUST BE A TYMCOM IX SFD PROGRAM WHICH HAS BEEN SAVED IN SYMBOLIC FORM ONLY AND WHICH CONTAINS THE SFD LINE NUMBERS. (I.E. A FILE WHICH DOES NOT CONTAINS SFD LINE NUMBERS WILL NOT BE PROPERLY ANALYZED).

### 3. LIMITATIONS

QSFID3 ONLY FLAGS CASES WHERE THE INDEX VARIABLE IS CHANGED VIA AN ASSIGNMENT STATEMENT. IT DOES NOT FLAG CASES WHERE THE INDEX VARIABLE IS CHANGED VIA AN ACCEPT OR READ STATEMENT.

### 4. SAMPLE RUN

1. FUNCTION

- A. REMOVES LINE NUMBERS.
- B. STATEMENT LABELS ARE ALIGNED AND STATEMENTS ARE INDENTED IN A FIXED FORMAT (SEE EXAMPLE).
- C. REMOVES LINE-FEEDS (LINE CONTINUATIONS) THIS PRODUCES LONG LINES.
- D. CHANGES "C:" COMMENTS TO "♦" COMMENTS.
- E. MODIFIES SIMPLE MULTIPLE STATEMENTS IN IF STATEMENTS ENCLOSED IN PARENTHESES (SEE EXAMPLE).
- F. MODIFIES SIMPLE IF-THEN-ELSE STATEMENTS (SEE EXAMPLE).
- G. MODIFIES IF-THEN-ELSE STATEMENTS NESTED TO A LEVEL OF THREE DEEP.
- H. MODIFIES MULTIPLE STATEMENTS ENCLOSED IN PARENTHESES IN SINGLE NON-NESTED IF-THEN-ELSE STATEMENTS.

2. RULES

- A. THE INPUT FILE MUST BE A STANDARD TYMCOM IX SFORTRAN PROGRAM WITH CONVENTIONAL TYMCOM IX SFORTRAN LINE NUMBERS.

3. LIMITATIONS

- A. THE CONVERSION AID MAY GENERATE STATEMENTS WITH LABELS BEGINNING WITH 9000 IN INCREMENTS OF ONE (1). THESE LABELS MAY CONFLICT WITH EXISTING STATEMENT LABELS.
- B. ALL MODIFIED STATEMENTS ARE DISPLAYED ALONG WITH THE NEW CODE GENERATED BY THE AID. THE USER SHOULD EXAMINE THE TRANSLATED CODE LOGIC TO DETERMINE THAT THE NEW CODE IS EQUIVALENT TO THE OLD.
- C. THIS AID WILL NOT PROPERLY RECOGNIZE IF OR IF-ELSE TYPE STATEMENTS WHICH HAVE EMBEDDED BLANKS EXAMPLE:  
 10 I F(A.GT.B)GO TO 10 . E L S E . DISPLAY 'NOT MODIFIED'

4. SAMPLE

## QSFIDS

### 1. FUNCTION

FLAGS LINK STATEMENTS.

FLAG IMPLIED DO STATEMENTS IN NON-I/O TYPE STATEMENTS.

FLAGS DIMENSIONING OR DECLARATIONS USING  
♦♦ OR ♦ OR :

FLAGS THE USE OF .EOR. AND .IMP. OPERATORS.

FLAGS THE USE OF IMPROPER TYMCOM X FILE NAMES IN OPEN STATEMENTS.

### 2. RULES

THE FILE SHOULD HAVE BEEN PROCESSED BY QSFIDS4 PRIOR TO EXAMINATION BY THIS AID.

OUTPUT CONSISTS OF SEQUENTIAL LINE NUMBERS AND THE ERROR FLAGS. THESE LINE NUMBERS CORRESPOND TO EDITOR LINE NUMBERS.

### 3. LIMITATIONS

IT IS RECOMMENDED THAT THIS PROGRAM BE RERUN AFTER CORRECTIONS ARE MADE.

CERTAIN VARIABLE NAMES CONTAINING LINK MAY CAUSE A FALSE FLAGGING OF A LINK STATEMENT.

SOME STRING CONSTANTS MAY TRIGGER FALSE FLAGGING OF IMPLIED DO MODIFIERS.

### 4. EXAMPLE