
Fujitsu Technology Solutions

FOR1-LZS (BS2000/OSD)
Version 2.2C
November 2010

Release Notice

All rights reserved, including intellectual property rights. Technical data subject to modifications and delivery subject to availability. Any liability that the data and illustrations are complete, actual or correct is excluded. Designations may be trademarks and/or copyrights of the respective manufacturer, the use of which by third parties for their own purposes may infringe the rights of such owner. For further information see http://ts.fujitsu.com/terms_of_use.html

Copyright © Fujitsu Technology Solutions 2010

Release Notice FOR1-LZS 2.2C

1 General	2
1.1 Ordering	4
1.2 Delivery	4
1.3 Documentation	9
1.3.1 Reference literature	10
2 Software Enhancements	11
2.1 SIFGEN description	11
3 Technical information	15
3.1 Memory requirements	15
3.2 Software configurations	15
3.3 Product installation	16
3.3.1 General	16
3.3.2 Installation	17
3.4 Product use	18
3.4.1 Notes	18
3.4.2 Optional REPs in LMS format	22
3.5 Procedure in the event of errors	28
4 Hardware	29

1 General

The FORTRAN compiler FOR1 V2.2C is hereby released for the BS2000/OSD operating system *).

The FOR1 compiler meets the American National Standard ANSI X3.9-1978 (ANS77) requirements for the FORTRAN programming language.

In addition to the FORTRAN 77 standard, the FORTRAN compiler of Fujitsu Technology Solutions has the language elements of FORTRAN IV and its extensions.

FOR1 has considerably extended the possible applications of FORTRAN. FOR1 has been successfully tested for compliance with and fulfilment of the standard by the Gesellschaft fuer Mathematik und Datenverarbeitung (GMD) and the National Centre for Information Technology (NCC) and has the certificate for validation level FULL.

FOR1 offers powerful optimization at five optimization levels.

The compiler, runtime system and generated objects can be loaded shareably.

Special system functions are available from the function pool (FPOOL).

FOR1 allows the extended address space to be used on XS systems for application programs as of BS2000 >= V9.5. By means of dynamic arrays, memory can be requested as late as at object runtime.

*1 This Release Notice is a summary of the major extensions, requirements and operating information with regard to the following technical delivery units

*2 The release level is that of November 2010.

*2 Changes in this correction state are marked with '*2' in the margin.

*2 Changes in correction state December 2003 are marked with '*1' in the margin.

The Release Notice is supplied as a text file. The typographical conventions for uppercase and lowercase are observed in the file.

*) BS2000/OSD (R) is a registered trademark of Fujitsu Technology Solutions

To print this file, use

```
/PRINT-DOCUMENT FROM-FILE=SYSFGM.FOR1-LZS.022.E,-  
/DOC-FORM=*TEXT(LINE-SPACING=*BY-EBCDIC-CONTROL)  
(english)
```

or

```
/PRINT-DOCUMENT FROM-FILE=SYSFGM.FOR1-LZS.022.D,-  
/DOC-FORM=*TEXT(LINE-SPACING=*BY-EBCDIC-CONTROL)  
(german)
```

All BS2000 product Release Notices are available in the Internet. This one is available under the following URL:

<http://manuals.ts.fujitsu.com>

If one or more previous versions are skipped when this version is used, the information from the BS2000/OSD-BC Release Notices (and README files) of the previous versions must be noted.

The use of names, trademarks, etc. in this Release Notice does not entitle readers to assume that these names/designations may be used without restriction by anyone; often the names/designations are protected by law or contract, even if this is not indicated here.

The only difference in V2.2C10 in related to V2.2C is that an addressing error was corrected.

The differences in V2.2C20 in related to V2.2C10 is that an error was corrected, A value is set into IOSTAT-parameter, if a record-splitting occurs, if an include library cannot be opened, the compiler continue with searching other libraries.

*1 The only difference in V2.2C30 in related to V2.2C20 is
*1 that an optimization error was corrected.

FOR1 V2.2C, like its predecessor V2.2B, offers the following functionality:

FOR1 V2.2C supports the symbolic debugger, AID.

FOR1 V2.2C supports the UTM/FOR1 interface.

The standard linkage (ILCS) is supported by this FOR1 version. This interface allows a TIAM call to be implemented from within a FOR1 program.

This version of FOR1 supports the standard linkage (ILCS). This interface allows a TIAM call to be implemented from within a FOR1 program.

The file names of the FOR1 package have been adapted to BS2000 conventions. This delivery includes SYSII files for IMON support.

This version FOR1 V2.2C supports SSCM.

1.1 Ordering

FOR1 V2.2C can be ordered from your local distributors and is subject to the general terms and conditions of the software product use and service agreement.

1.2 Delivery

FOR1 V2.2C files are supplied via SOLIS in BS2000.

Delivery components:

SINPRC.FP00LITY.022	Procedure to install FP00LITY in an arbitrary user ID.
SYSDOC.FP00LITY.022.D	Documentation (German)
SYSDOC.FP00LITY.022.E	Documentation (English)
SYSPRG.FP00LITY.022	FP00LITY load module
SYSSII.FP00LITY.022	IMON information file
SINPRC.FOR1.022	Library containing procedure IF@COPY to generate the file names compatible to previous FOR1 versions
SYSBIB.FOR1.022.FP00L	Descriptors for the FP00L-Functions supported by FOR1
SYSENT.FOR1.022.LOAD1	Batch Job to preload the Compiler
SYSENT.FOR1.022.LOAD2	Batch Job to preload the Compiler
SYSLIB.FOR1.022	Library containing Assembler Macros
SYSPRC.FOR1.022.SYSL0D	Procedure to preload Compiler and Runtime system
SYSPRG.FOR1.022	Load Module of the FOR1 Compiler
SYSPRG.FOR1.022.LOAD	Load module of the FOR1 preloader program
SYSPRG.FOR1.022.SIFGEN	Load module for generation of SIF files
SYSSII.FOR1.022	IMON information file
SYSSPR.FOR1.022.095	Starting procedure for the Compiler in BS2000 V9.5
SYSSPR.FOR1.022.100	Starting procedure for the Compiler in BS2000 V10.0 or later versions

SINPRC.FOR1-LZS.022 Library which contains the procedures IF@INSTAL (for the installation of FOR1 under an arbitrary user ID, without IMON), IF@COPY (for the generation of file names compatible to the previous FOR1 versions), IF@OPT.REP (for implementation of optional REP's).

SYSGM.FOR1-LZS.022.D Release Notice in German

SYSGM.FOR1-LZS.022.E Release Notice in English

SYSLNK.FOR1-LZS.022.DYNAMIC Library containing the Shareable Runtime System Modules

SYSLNK.FOR1-LZS.022.FPOOL Module library for FPOOL

SYSLNK.FOR1-LZS.022.STATIC Library containing the Nonsharable Runtime System Modules

SYSPRC.FOR1-LZS.022.SHARE.095 Procedure called by the commands
/START-FOR1-COMPILER and
/START-FOR1-PROGRAM
BS2000 V9.5

SYSPRC.FOR1-LZS.022.SHARE.100 Procedure called by the commands
/START-FOR1-COMPILER and
/START-FOR1-PROGRAM
BS2000 >= V10

SYSSDF.FOR1-LZS.022 SDF system syntax file,
BS2000 V11

SYSSDF.FOR1-LZS.022.SYSTEM SDF system syntax file,
BS2000 V10

SYSSDF.FOR1-LZS.022.SYSTEM.095 SDF system syntax file,
BS2000 V9.5

SYSSDF.FOR1-LZS.022.USER SDF user syntax file,
BS2000 >= V10

SYSSDF.FOR1-LZS.022.USER.095 SDF user syntax file,
BS2000 V9.5

SYSSII.FOR1-LZS.022 IMON information file

SYSSPR.FOR1-LZS.022 Starting procedure for FOR1 programs

SYSSSC.F0R1-LZS.022.CL4	Subsystem declarations class 4 memory
SYSSSC.F0R1-LZS.022.CL5	Subsystem declarations class 5 memory
SYSSSC.F0R1-LZS.022.CL6	Subsystem declarations class 6 memory
SYSSSD.F0R1-LZS.022.CL4	DSSM declarations class 4 memory
SYSSSD.F0R1-LZS.022.CL5	DSSM declarations class 5 memory
SYSSSD.F0R1-LZS.022.CL6	DSSM declarations class 6 memory

names in V2.2C

SINPRC.FP00LITY.022
 SYSDOC.FP00LITY.022.D
 SYSDOC.FP00LITY.022.E
 SYSPRG.FP00LITY.022
 SYSSII.FP00LITY.022
 SINPRC.FOR1.022
 SYSBIB.FOR1.022.FP00L
 SYSENT.FOR1.022.LOAD1
 SYSENT.FOR1.022.LOAD2
 SYSLIB.FOR1.022
 SYSPRC.FOR1.022.SYSL0D
 SYSPRG.FOR1.022
 SYSPRG.FOR1.022.LOAD
 SYSPRG.FOR1.022.SIFGEN
 SYSSII.FOR1.022
 SYSSPR.FOR1.022.095
 SYSSPR.FOR1.022.100

names in V2.2B

FP00LITY.P.INSTA
 FP00LITY.HELPTAB.DEUTSCH
 FP00LITY.HELPTAB
 FP00LITY
 new
 new
 FOR1.FP00L
 SYSENT.FOR1.022.LOAD1
 SYSENT.FOR1.022.LOAD2
 FOR1MACLIB
 SYSPRC.FOR1.022.SYSL0D
 FOR1
 SYSPRG.FOR1.022.LOAD
 SYSPRG.FOR1.022.SIFGEN
 new
 SYSSPR.FOR1.022.COMPILE and
 SYSSPR.FOR1.022.COMPILE.SIF
 SYSSPR.FOR1.022.COMPILE and
 SYSSPR.FOR1.022.COMPILE.SIF

Names in V2.2C -----	Names in V2.2B -----
SINPRC.FOR1-LZS.022	includes SYSPRC.FOR1.022.INSTALL1.SDF SYSPRC.FOR1.022.OPT.REP
SYSFGM.FOR1-LZS.022.D	SYSFGM.FOR1.022.D
SYSFGM.FOR1-LZS.022.E	SYSFGM.FOR1.022.E
SYSLNK.FOR1-LZS.022.DYNAMIC	FOR1MODLIBS
SYSLNK.FOR1-LZS.022.FPOOL	FOR1.FPOOLLIB
SYSLNK.FOR1-LZS.022.STATIC	FOR1MODLIB
SYSPRC.FOR1-LZS.022.SHARE.095	SYSPRC.FOR1.022.SHARE.95
SYSPRC.FOR1-LZS.022.SHARE.100	SYSPRC.FOR1.022.SHARE
SYSSDF.FOR1-LZS.022	SYSSDF.FOR1.022 and SYSSDF.FOR1.022.SIF
SYSSDF.FOR1-LZS.022.SYSTEM	SYSSDF.FOR1.022 and SYSSDF.FOR1.022.SIF
SYSSDF.FOR1-LZS.022.SYSTEM.095	SYSSDF.FOR1.022.95 and SYSSDF.FOR1.022.SIF.95
SYSSDF.FOR1-LZS.022.USER	SYSSDF.USER.FOR1.022 and SYSSDF.USER.FOR1.022.SIF
SYSSDF.FOR1-LZS.022.USER.095	SYSSDF.USER.FOR1.022.95 and SYSSDF.USER.FOR1.022.SIF.95
SYSSII.FOR1-LZS.022	new
SYSSPR.FOR1-LZS.022	SYSSPR.FOR1.022.EXECUTE
SYSSSC.FOR1-LZS.022.CL4	new
SYSSSC.FOR1-LZS.022.CL5	new
SYSSSC.FOR1-LZS.022.CL6	new
SYSSSD.FOR1-LZS.022.CL4	SYSSSD.FOR1.022.CL4
SYSSSD.FOR1-LZS.022.CL5	SYSSSD.FOR1.022.CL5
SYSSSD.FOR1-LZS.022.CL6	SYSSSD.FOR1.022.CL6

1.3 Documentation

Manuals for the FOR1 V2.2C product:

Title	! Order no.
FOR1 (BS2000) FORTRAN Compiler Reference Manual (complete V1.6A edition)	! U3298-J-Z55-1-7600 ! ! !
Revision of V2.0A	! U3298-J1-Z55-2-7600
Revision of V2.1A	! U3298-J2-Z55-3-7600 !
FOR1 (BS2000) FORTRAN Compiler User Guide (complete V2.2A edition)	! U577-J-Z125-7-7600 ! ! !
FOR1 (BS2000) FORTRAN Compiler Reference Guide (Revision of V2.1A)	! U489-J1-Z55-5-7600 ! ! !
AID (BS2000) Debugging of FORTRAN Programs	! U4298-J-Z125-3-7600 !
AID (BS2000) Advanced Interactive Debugger	! U2853-J-Z125-4-7600 !
UTM V3.3A/UTM-D V2.0A Supplement for FORTRAN User Guide	! U5460-J-Z135-3-7600 ! !
FHS Supplement for FORTRAN and PLI1	! U6738-J-Z135-1-7600 ! !
C, COBOL, FORTRAN System Interfaces for Applications	! U7318-J-Z145-1-7600 ! !

The documentation is available as online manuals, see
<http://manuals.ts.fujitsu.com>
and can be ordered in paper form for an extra fee through the
manual shop in the internet:

<http://manualshop.ts.fujitsu.com>

The product Adobe Acrobat Reader as of V4.05 is required to
read or print the manuals in PDF format.

There may also be README- files available for the manuals
listed above. These files contain modifications and supplements
for the manual applying to each relevant product. The names of
these files are structured as follows:

SYSRME.<prod>.<vers>.D (file with German text)
SYSRME.<prod>.<vers>.E (file with English text)

1.3.1 Reference literature

- [1] ARITHMOS Product Information Sheet
Brief Description
User Guide
Tables
- [2] FPOOLITY (BS2000) Reference Manual
- [3] AID (BS2000) Reference Manual
Debugging on Machine Code Level, User Guide
- [4] IFG User's Guide
- [5] FHS User Guide
- [6] UTM Programming Applications, Core Manual
- [7] UTM Generation and Administration,
User Guide
- [8] UTM Planning and Design, User Guide
- [9] LMS (BS2000), Reference Manual
- [10] Job Variables (BS2000), Reference Manual
- [11] BS2000/OSD-BC V1.0, User Commands (SDF Format)

2 Software Enhancements

*1 Until now, this chapter was listed as:
*1 'Chapter 3 MODIFIED FUNCTIONS'.

1. SIF

Both user and system syntax files and the procedure SYSSPR.FOR1.022 have been enhanced for support of the SIF facility.

2. ILCS

The module IT0INITG is no more needed and has been replaced by the ILCS modules IT0ENTR and IT0INITS of ILCS version V1.4.

3. Message about ulp-Mathematics when using SYSLNK.FOR1-LZS.022.DYNAMIC

The message output previously in case when a ulp-precision mathematical function is found in the program, will be output only when REP Nr. 13 is installed. (When using SYSLNK.FOR1-LZS.022.STATIC the message is output as before).

2.1 SIFGEN description

*1 Until now, this chapter was listed as 'Chapter 6 Appendix;
*1 Chapter 6.1 SIFGEN description'

If you make symbolic trace with AID, you can use SIF-files (SIF= Source Information File) to get the SOURCE-lines of a FORTRAN-program. You can generate SIF-files by using following SDF-OPTION at FORTRAN-translation:

```
TEST-SUPPORT=PAR(TOOL-SUPPORT=*AID,SOURCE-TRACE=*YES)
```

Prerequisites:

In the parameter SOURCE, the name of a file or a plamspecifica-
tion must be declared.

LISTING=PAR(OUTPUT=full-filename) must be sepcified, a
plamspezifikation, *SYSLST or *STD-FILE are not allowed.

The SIF-files also may be generated by calling the program
SYSPRG.FOR1.022.SIFGEN

The name of the LISTFILE will be asked during execution of this
program.

The listing file must contain the source-listing and the op-
tion-listing. The option listing is only used, if the actual
TSN differs from the TSN during the compilation of the pro-
gram-source.

For every program unit of a FORTRAN-program
 SYSPRG.FOR1.022.SIFGEN generates a separate SIF-file with following name:

AID.SIF.puname .

If a program unit will be translated again, the SIF-file must be generated too.

Example:

```

-----
S-FOR1-C SOURCE=INDEX.S,LIST=PAR(OUT=LF),
                SOURCE-PROP=PAR(LINE=';'),
TEST=PAR(TOOL-SUPPORT=*AID,SOURCE-TRACE=*YES)
% BLS0500 PROGRAM 'FOR1', VERSION '2.2C10' OF
                '1998-10-05' LOADED.
FOR1: V2.2C10 READY, GIVE COMPILER OPTION
FOR1: LIST FILE REPLACED = LF
FOR1: COMPILER NOT PRELOADED (BAD LOAD PERFORMANCE)
FOR1: NO ERRORS DURING COMPILATION OF P.U. INDEX
END OF F O R 1 COMPILATION; CPU TIME USED: 0.435 SEC.
% BLS0500 PROGRAM 'FOR1.SIFGEN.22', VERSION '
                OF '1991-04-12' LOADED
BS2000 F O R 1 : FORTRAN PROGRAM "SIFGEN"
STARTED ON 99-07-07 AT 11:33:00
SIFGEN: GIVE NAME OF LISTING FILE
SIFGEN: LISTING FILE = LF
SIFGEN: COMPILATION UNDER SAME TASK (Y/N)?
SIFGEN: COMPUTED TASK SEQUENCE NUMBER = 9598
SIFGEN: GENERATION OF FILE AID.SIF.INDEX
FOR1 STOP SIFGEN: PROPERLY TERMINATED
BS2000 F O R 1 : FORTRAN PROGRAM "SIFGEN "
                ENDED PROPERLY AT 11:33:01
CPU - TIME USED :      0.0596 SECONDS
ELAPSED TIME   :      0.7640 SECONDS
LOAD *,SYM=ALL
% BLS0517 MODULE 'INDEX' LOADED
%T 20
BS2000 F O R 1 : FORTRAN PROGRAM "INDEX"
STARTED ON 99-07-07 AT 11:33:23
 10      WRITE *, 'PLEASE TYPE FOLLOWING VALUES'
PLEASE TYPE FOLLOWING VALUES
 11      WRITE *, ' NUMBER OF DIMENSIONS:'
NUMBER OF DIMENSIONS:
 12      READ *, NUMBER
2
 13      WRITE *, 'PLEASE TYPE LIMITS ;
                LOWER-, UPPER BOUND, .
PLEASE TYPE LIMITS ; LOWER-, UPPER BOUND, . . .
1,10,1,10
 14      READ *, ((DIM(I,J),I=1,2),J=1,NUMBER)
 14      READ *, ((DIM(I,J),I=1,2),J=1,NUMBER)
 15      WRITE *, 'PLEASE TYPE ACTUAL INDICES'
PLEASE TYPE ACTUAL INDICES
2,12
 16      READ *, (IND(I),I=1,NUMBER)
 16      READ *, (IND(I),I=1,NUMBER)
 17      WRITE *, 'PLEASE TYPE LENGTH OF
                THE ARRAY-ELEMENTS'
PLEASE TYPE LENGTH OF THE ARRAY-ELEMENTS
 18      READ *,L

```

```

1
19          DO 10,I=1,NUMBER
20 10      D(I) = DIM(2,I) - DIM(1,I) + 1
20 10      D(I) = DIM(2,I) - DIM(1,I) + 1
21          M(0) = 1
22          M(1) = D(1)
23          DO 20,I=2,NUMBER
24 20      M(I) = D(I)*M(I-1)
25          SUM = 0
26          DO 30, I=1,NUMBER
STOPPED AT SRC_REF: 26
SOURCE: INDEX ,
PROC: INDEX
%T
27 30      SUM = SUM + (DIM(1,I)*M(I-1))
27 30      SUM = SUM + (DIM(1,I)*M(I-1))
28          A0 = -L * SUM
29          SUM = 0
30          DO 40,I=1,NUMBER
31 40      SUM = SUM + (IND(I)*M(I-1))
31 40      SUM = SUM + (IND(I)*M(I-1))
32          AS = A0 + L * SUM
33          WRITE *, 'POSITION OF ARRAY ELEMENTS: ',AS
POSITION OF ARRAY ELEMENTS: 111
34          WRITE *, 'COMPLETE SIZE OF ARRAYS      :
                                     ',M(NUMBER)*L
COMPLETE SIZE OF ARRAYS      : 100
35          WRITE *, ' 0 <= ',AS,' < ',M(NUMBER)*L,' ?'
0 <= 111 < 100 ?
BS2000 F O R 1 : FORTRAN PROGRAM
                "INDEX " ENDED PROPERLY AT 11
CPU - TIME USED :          0.2430 SECONDS
ELAPSED TIME   :          40.7600 SECONDS

```

Original program:

```
-----  
PROGRAM INDEX  
INTEGER NUMBER      ; NUMBER OF DIMENSIONS  
INTEGER DIM(2,20)   ; UPPER AND LOWER BOUNDARIES OF  
*                   ; SINGLE DIMENSIONS  
INTEGER IND(20)     ; AKTUEL INDICES OF COMPUTED  
*                   ; ARRAY-ELEMENT  
INTEGER D(20)       ; SIZE OF A SINGLE DIMENSIONEN  
INTEGER M(0:20)     ; SIZE OF I-DIMENSIONAL SUBMATRIX  
INTEGER L           ; LENGTH OF ONE ARRAY-ELEMENT  
INTEGER SUM         ; TEMPORARY VARIABLE  
INTEGER AS,A0       ; ARRAY-ELEMENT; VIRTUEL ZERO-ELEMENT  
  
WRITE *, 'PLEASE TYPE FOLLOWING VALUES'  
WRITE *, ' NUMBER OF DIMENSIONS:'  
READ *, NUMBER  
WRITE *, 'PLEASE TYPE LIMITS ; LOWER-, UPPER BOUND,...'  
READ *, ((DIM(I,J),I=1,2),J=1,NUMBER)  
WRITE *, 'PLEASE TYPE ACTUAL INDICES'  
READ *, (IND(I),I=1,NUMBER)  
WRITE *, 'PLEASE TYPE LENGTH OF THE ARRAY-ELEMENTS'  
READ *, L  
*  
DO 10, I=1, NUMBER  
  10      D(I) = DIM(2,I) - DIM(1,I) + 1  
*  
M(0) = 1  
M(1) = D(1)  
DO 20, I=2, NUMBER  
  20      M(I) = D(I)*M(I-1)  
*  
SUM = 0  
DO 30, I=1, NUMBER  
  30      SUM = SUM + (DIM(1,I)*M(I-1))  
*  
A0 = -L * SUM  
*  
SUM = 0  
DO 40, I=1, NUMBER  
  40      SUM = SUM + (IND(I)*M(I-1))  
*  
AS = A0 + L * SUM  
*  
WRITE *, 'POSITION OF ARRAY ELEMENTS: ', AS  
WRITE *, 'COMPLETE SIZE OF ARRAYS      : ', M(NUMBER)*L  
WRITE *, ' 0 <= ', AS, ' < ', M(NUMBER)*L, ' ?'  
END
```

3 Technical information

3.1 Memory requirements

The FOR1 Version 2.2C compiler can be used on systems whose real memory size is ≥ 2 Mbytes. The recommended memory size, however, is ≥ 3 Mbytes.

Paging area space required by the FOR1 compiler:

non-sharable	- approx. 1.6 Mbytes per user (without work areas)
sharable	- approx. 2.5 Mbytes (once)

Runtime system:

sharable	- approx. 0.34 Mbytes
----------	-----------------------

A virtual address space of 2 Mbytes is theoretically sufficient. As this may limit the program size, it is good policy to generate a user address space of ≥ 3 Mbytes.

3.2 Software configurations

*2 FOR1 Version 2.2C requires BS2000 as of V9.5
*2 (BS2000-GA as of V9.5 or OSD-BC as of 1.0).

Compilers from versions $< V1.6$ can no longer run under BS2000 systems $\geq V9.0$.

The use of the FOR1 SDF commands is only possible as of SDF (System Dialog Facility) V1.0D. As of B2000 V10.0 SDF version $\geq V2.0A$ is required.

Old FORTRAN objects from versions < 1.5 cannot be linked to the FOR1 V2.2C runtime system. The FORTRAN programs concerned must be recompiled.

For debugging by means of AID, you need AID $\geq V1.0C$, the earliest AID version to support FOR1.

For UTM V3.3A you need the FOR1 runtime system V2.2. Refer to chapter 5, section 9, UTM-ILCS. To run programs in an ILCS environment, you need ILCS $\geq V1.4$. The required ILCS files are delivered together with FOR1 and the FOR1 runtime system.

For FOR1-TIAM calls, you need an ILCS-capable TIAM version. This interface is described in the FHS Supplement for FORTRAN and PLI1.

3.3 Product installation

3.3.1 General

Standard installation is via SOLIS.

The files are created with the SHARE=YES and ACCESS= READ parameters.

The macros from SYSLIB.FOR1.022 can be transferred to MACROLIB under TSOS.

The modules from SYSLNK.FOR1-LZS.022.DYNAMIC or SYSLNK.FOR1-LZS.022.STATIC can be transferred to the system TASKLIB. In any case, the IF@ERR2 and IF@DBG2 runtime modules must be transferred to the system TASKLIB since they are dynamically loaded at runtime, if SYSLNK.FOR1-LZS.022.DYNAMIC is not used, when an error occurs or the DEBUG subroutine is called up.

*1 In case the FOR-LZS library cannot be found via the TASKLIB,
*1 the runtime library for FOR1 assigned by IMON is used.

DSSM and SSCM declarations are available for loading the sharable runtime system in class 4, class 5 or class 6 memory.

The system administrator can also load the sharable runtime system in class 4 memory by using the SHARE command.

The preload program (SYSPRG.FOR1.022.LOAD or FOR1.L.LOD) from the previous version must be replaced by the current SYSPRG.FOR1.022.LOAD program from FOR1 V2.2C as the preload program is related to a certain compiler version. The preload job is started with /ENTER-JOB SYSENT.FOR1.022.LOAD1.

The SYSPRC.FOR1.022.SYSL0D procedure contains the &SHARE parameter which can be used to select whether the sharable runtime system should be loaded with the /SHARE command (class 4) or via DSSM or SSCM (class 4, 5 or 6).

Use of USER syntax files for V2.2C may sometimes create problems when the system syntax file contains entries for an earlier version of the compiler. Should you meet problems when running FOR1 V2.2C as a parallel version, you should use the SDF USER file SYSSDF.FOR1-LZS.022.USER.095 even for BS2000 >= V10.0.

3.3.2 Installation

The Install procedures are now contained in the LMS libraries SINPRC.FOR1.022 and SINPRC.FOR1-LZS.022. These procedures must be executed only when the installation by SOLIS has been completed.

Procedure SINPRC.FOR1-LZS.022(IF@COPY) serves to generate file-names compatible to the names in previous FOR1 versions. Thus user procedures can be executed without changes, in many cases.

Procedure IF@COPY in library SINPRC.FOR1.022 copies

from SYSPRG.FOR1.022 to FOR1 ,
from SYSBIB.FOR1.022.FP00L to FOR1.FP00L and
from SYSLIB.FOR1.022 to FOR1MACLIB

Procedure IF@COPY in library SINPRC.FOR1-LZS.022 copies

from SYSLNK.FOR1-LZS.022.FP00L to FOR1.FP00LLIB ,
from SYSLNK.FOR1-LZS.022.STATIC to FOR1MODLIB and
from SYSLNK.FOR1-LZS.022.DYNAMIC to FOR1MODLIBS

*1 As of OSD V2.0, the file attributes of the old files may be
*1 unchanged, if 'IF@COPY.IGNORE' is used instead of 'IF@COPY'.

Optional REPS are installed in the runtime library by use of procedure SINPRC.FOR1-LZS.022(IF@OPT.REP).

IF@INSTAL in library SINPRC.FOR1-LZS.022 should be used to make any further adaptations necessary during installation (see below).

IF@INSTAL procedure with LMS element version 095 must be used for installation in BS2000 V9.5,
IF@INSTAL version 100 must be used for BS2000 V10, and
IF@INSTAL version 110 must be used for BS2000 V11 or later versions.

The install procedures only modify the file attributes SHARE and ACCESS if MOD=' ' is specified as argument when the procedure is called. When MOD=' ' is specified, the procedure behaves as in FOR1 V2.2B :
before any modification to the file the file attributes are set to ACCESS=WRITE, when the modification is done, the attributes are changed to ACCESS=READ and SHARE=YES.

When the default value of the MOD parameter is in effect, no file attributes are changed.

If the product SDF-A is installed, procedure SINPRC.FOR1-LZS.022(IF@INSTAL) can be used to change the names of the SDF procedure files as well as the name of the file which contains the compiler. Furthermore, the default user id's '\$' for the SDF procedures can be adopted in both user and system syntax files.

In this case, the system administrator must merge in the FOR1 V2.2C system syntax files manually with the SDF-I utility. The following parameter must be specified when the MERGE command is executed:

REPLACE-PRODUCT=YES

The SYSSDF.FOR1.022.USER user syntax file does not contain a USERID for the compiler call nor for the SYSSPR.FOR1.022.COMPILE and SYSSPR.FOR1.022.EXECUTE procedure files.

The SYSSDF.FOR1.022 system syntax file contains \$FOR1 for the compiler call and the \$TSOS USERID for the SYSSPR.FOR1.022.COMPILE and SYSSPR.FOR1.022.EXECUTE procedure files.

FOR1 programs run under ILCS >= V1.3. The associated ILCS files are delivered with FOR1 and FOR1 runtime system. To install ILCS please use the supplied SYSPRC.ILCS.XXXXXXXX.INSTAL procedure (XXXXXXX = version, e.g. 014).

The file names of the FPOOLITY files can be modified for FPOOLITY with the FPOOLITY.P.INSTA installation procedure.

3.4 Product use

3.4.1 Notes

*1 Until now, this chapter was named 'Chapter 5 NOTES'.

1. Linking with SYSLNK.FOR1-LZS.022.DYNAMIC

When linking with TSOSLNK, the error handling modules, IF@ERR2 and IF@DBG2, do not have to be explicitly linked in; these modules are in the main module, IF@RTS1.

Important:

If you want to run a program in the upper address space, the IF@ERR2 and IF@DBG2 modules must not be explicitly linked in as they cannot run in the upper address space.

2. Module names

As the FOR1 runtime system also contains some modules of the SPL runtime system, you should not use any module names beginning with ITS.

3. Error message

If an error occurs in a statement containing more than 1638 lines, the error message always appears after the 1638th line.

4. Compilation time

If a program contains a large number of EQUIVALENCE statements, the compilation time with COMOPT OPTIMIZE=3 worsens many times over (up to factor 40). Such programs should be compiled with OPTIMIZE=1. This applies in particular if you are using IFG or UTM whose includes contain a large number of EQUIVALENCE statements.

5. ASCII and BTAM

Under BS2000 \geq V10, conversion to EBCDIC takes place by default when tape files in ASCII format are accessed via the BTAM access method.

If you want to load ASCII files (via BTAM) under BS2000 \geq V10 without them being automatically converted to EBCDIC, you must set OPEN=INPUT and TRANS=NO in the FILE command. A file opened in this way cannot be written to.

6. FPOOLITY and NK4

FPOOLITY does not support NK4 pubsets.

7. Loading above 16 Mbytes

Programs that are to run above 16 Mbytes cannot be linked in by SYSLNK.FOR1-LZS.022.STATIC. In this case, you must use SYSLNK.FOR1-LZS.022.DYNAMIC.

8. Language mixture

To use a mixture of languages, e.g. C and Fortran 90, you must use the shareable runtime system SYSLNK.FOR1-LZS.022.DYNAMIC. The nonshareable runtime system is to be used only for statically linking in pure FOR1 programs.

9. UTM-ILCS

In a UTM-ILCS application, even the FOR1 program units must have been generated with *COMOPT LINKAGE=STD.

A UTM application may not consist of a mixture of NXS programs created with V2.0 or V2.1 and programs created with V2.2.

When a program unit that was generated with COMP=FOR1 and compiled with LINKAGE=FOR1-SPECIFIC (old FOR1 linkage) is started up, the "KDCCFOR1" FOR1 linkage module is called up. This module eliminates the first two parameters in the parameter list, KB and SPAB, which are not required by FOR1.

If a FOR1 program unit is compiled with LINKAGE=STD (ILCS linkage), it must be generated with COMP=ILCS. In this case, when the program unit is started up, the "KDCCILC" ILCS linkage module is called up. This module does not convert the parameter list.

If a FOR1 program unit has a parameter list, KB and SPAB, the first two parameters, must be added to it when converting to ILCS. These parameters must be specified as dummy parameters, i.e. they must not be used. KB and SPAB are transferred via COMMON as before.

In the FORTRAN Supplement manual, the second example given in the section on additional data areas that can be used simultaneously must be corrected as follows:

Example:

FOR1 linkage:

```
SUBROUTINE TPR1 (BER1,BER2,BER3)
CHARACTER *2000 BER1, BER2, BER3
```

ILCS linkage:

```
SUBROUTINE TPR1 (DUMMY1,DUMMY2,BER1,BER2,BER3)
CHARACTER *2000 BER1, BER2, BER3
CHARACTER *1 DUMMY1, DUMMY2
```

10. CRTE

When the CRTE library has to be accessed, e.g. in case of a language mixture, care must be taken that the CRTE library be scanned by the linker before the library containing library containing the FOR1 runtime system.

Any version of ILCS V1.3 or later may be used to link programs compiled by FOR1.
For programs involving language mixture, however, ILCS version V1.4 or later must be used.

Examples of correct sequences of linker directives are :

```
/EXEC $TSOSLNK
..
RESOLVE ,$.SYSLNK.FOR1-LZS.022
RESOLVE ,$.SYSLNK.CRTE.022.SHARE
EXCLUDE ,TASKLIB
END
```

```
/START-BINDER
//..
//RESOLVE-BY-AUTOLINK LIBRARY=$.SYSLNK.CRTE.022.SHARE
//RESOLVE-BY-AUTOLINK LIBRARY=$.SYSLNK.FOR1-LZS.022
//..
```

3.4.2 Optional REPs in LMS format

With the SINPRC.FOR1-LZS.022(IF@OPT.REP) procedure, the following REPs can be integrated in the SYSLNK.FOR1-LZS.022.DYNAMIC, SYSLNK.FOR1-LZS.022.STATIC, FOR1MODLIB and/or FOR1MODLIBS runtime system either individually or all together.

You may select the libraries where the REPs are to be integrated either by specifying procedure arguments or by prompting. When prompting is used, you will be asked for every of the four libraries whether a specific REP or all REP's are to be installed.

For SYSLNK.FOR1-LZS.022.DYNAMIC linkage REPs 13 and 14 can also be used as linkage REPs while REPs 1-12 have no effect during linkage as the IF@RTS1 module is reloaded. With REP 14, you can use a runtime system that is provided with REPs suitable for your purposes under your own ID, even if the FOR1 runtime system is loaded in class 4 memory.

1. Lowercase characters read in from the terminal are automatically converted to uppercase. This can be suppressed by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS 7F0
*REP 324,X'00'=X'10'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@FCB
*REP 324,X'00'=X'10'
*END
```

2. Terminal control characters are automatically suppressed in screen display. This can be changed by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 56,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 56,X'00'=X'01'
*END
```

3. An executing FOR1 program automatically displays start and end messages. These messages can be suppressed by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 57,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 57,X'00'=X'01'
*END
```

4. Z-format output automatically generates leading zeros. These leading zeros can be replaced with blanks by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 58,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 58,X'00'=X'01'
*END
```

5. FOR1, when generating a file, automatically display a message. This message can be suppressed by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 59,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 59,X'00'=X'01'
*END
```


6. FOR1, when closing a previously unopened file, automatically displays a message. This message can be suppressed by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 5A,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 5A,X'00'=X'01'
*END
```

7. When a FOR1 file is created, a secondary storage of 30 PAM pages is automatically declared. This can be changed to the BS2000 installation presetting by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 5B,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 5B,X'00'=X'01'
*END
```

8. FOR1 automatically uses the presetting

```
*RUNOPT OVERPRINT=LASER
```

RUNOPT OVERPRINT=YES can be set by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 5C,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 5C,X'00'=X'01'
*END
```

9. FOR1 >= V2.0A, for a repeated OPEN on the same file, does not automatically position to the beginning of the file. Positioning at the beginning of the file for a repeated OPEN according to the previous version, can be set by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 5D,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 5D,X'00'=X'01'
*END
```

10. When a file is written to without format via ACCESS='DIRECT', the Fortran record is automatically split if it does not fit in the system record and a warning is displayed.

This warning can be suppressed by the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 5E,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 5E,X'00'=X'01'
*END
```

11. The EXIST parameter of the INQUIRE statement automatically checks whether a unit number is within the valid range (0...99).

This REP checks whether a unit is or was linked to a file.

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 5F,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 5F,X'00'=X'01'
*END
```

12. FOR1 automatically makes the current position the end of the file in the ENDFILE statement, and there is no positioning to the beginning of the file. Previous behaviour (ENDFILE same as CLOSE) can be restored with the following REP:

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 60,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 60,X'00'=X'01'
*END
```

13. As of FOR1 V2.2A, a message was displayed once at program startup if a new ulp-precision mathematical routine exists in the FOR1 program. This message is suppressed for FOR1 runtime versions \geq V2.2C.

By using REP 13, this message is displayed at every program startup independently of whether a ulp-precision mathematical function is contained in the program or not.

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@SHR0
*REP 1E,X'00'=X'01'
*END
```

14. Usually, the IF@RTS1 main module is reloaded with TASKLIB=YES. By using REP 14, IF@RTS1 is reloaded from a private FOR1 runtime library.

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@SHR0
*REP 87,X'B0'=X'00'
*REP 100,X'03'=X'83'
*END
UPDR IF@XSHR0
*REP CB,X'18'='58'
*END
```

15. An ISAM-record is readed without Setting of LOCK-BITS
at OPEN=INPUT.

```
LIB SYSLNK.FOR1-LZS.022.DYNAMIC,BOTH
UPDR IF@RTS1
*BAS DB8
*REP 63,X'00'=X'01'
*END
```

```
LIB SYSLNK.FOR1-LZS.022.STATIC,BOTH
UPDR IF@INIT
*REP 63,X'00'=X'01'
*END
```

3.5 Procedure in the event of errors

In the event of an error, the following error documentation will be required for diagnostic purposes:

Error documentation to be sent:	! Compilation ! error	! Runtime ! error
-----+-----+-----		
Source (including INCLUDEs)	! X	! X
Source, diag./option listings	! X	! X
Runtime log (MSG=FH)	! X	! X
Object listing	!	! X
Linkage editor listing	!	! X
Input/output files	!	! X
Expected result	!	! X
Brief description of the run	!	! X

4 Hardware

- *2 FOR1 Version 2.2C will run on all CPUs supported by
- *2 - BS2000/BS2000-GA as of V9.5
- *2 - OSD/BC as of V1.0.