READ THIS FIRST
Sun FORTRAN

Sun Microsystems, Inc. • 2550 Garcia Avenue • Mountain View, CA 94043 • 415-960-1300

Part No: 800-3140-10
Rev A of 10 March 1989
Release 1.2

Read This First

Software READ THIS FIRST
Sun FORTRAN

Introduction

These notes provide a quick overview of compatibility, getting assistance, and possible problems involved in the installation and use of Sun FORTRAN, including any relevant information obtained *after* the manuals were sent to the printer. The documentation assumes you are familiar with the Sun operating system (SunOS™). If you find any instructions relating to SunOS utilities confusing, please refer to the *Commands Reference Manual.*

Compatibility

This version of Sun FORTRAN runs on a Sun-3™ or Sun-4™ under SunOS Release 4.0 or later.

Getting Help

If you have any problems installing Sun FORTRAN 1.2, in the U.S. call Sun Microsystems at: 1-800-USA-4SUN (1-800-872-4786) or outside the U.S. call your local Sun Microsystems office. Have your system's model number, serial number, Sun FORTRAN 1.2 release number (for software), and Sun operating system (SunOS™) release number ready to give to the dispatcher. You can also send questions by electronic mail to `sun!hotline`. Be sure to include your name, company, and phone number, along with the information above.

If you have questions about Sun's support services or your shipment, call your sales representative.

To see the SunOS release number, type: **`cat /etc/motd`**
To see the FORTRAN release number, type: **`cat /usr/lib/lang_info`**

Installation Background

- Use SunOS 4.0 or later, otherwise you cannot install Sun FORTRAN 1.2 .
- You need about 1.7 megabytes on a Sun-3 (2.6 on a Sun-4) in `/usr` for the Sun FORTRAN 1.2 software.
- If you are upgrading from an older release of Sun FORTRAN you may not have room in the `/usr` directory. If you do not have room there, move the Sun FORTRAN files to another partition (or delete them) before installing this release. To see which files are Sun FORTRAN files, check the file list near the end of this document.
- Installation is done from tape by a general installation script that asks various questions. Some of the terms you will need to know are explained in the next section. The contents of the tape are listed near the end of this document.
- If you have not installed the SunOS `man` pages, then the Sun FORTRAN installation script will not install the Sun FORTRAN man pages, and you will see an error message for each Sun FORTRAN `man` page it tries to install. You can ignore these messages. The non-man page material installs ok.

Installation Script Terms You Need To Know

• *Device Name* — The standard device name for ¼ " cartridge tape is `st0`; and for ½ " reel tape, it is `mt0`. If you have more than one tape drive attached to your workstation, then the last digit might not be zero, but that is uncommon.

• *Server/Client* — If you install Sun FORTRAN onto machine X for use from machines B and C, then X is the *server* and B and C are the *clients*. Note: *install only onto a server or a standalone* (not onto a client).

• *Heterogeneous/Homogeneous* — If machines X and B and C are of the *same* architecture (all Sun-3, or all Sun-4, or all Sun386*i*), then machine X is a *homogeneous* server. Otherwise, machine X is a *heterogeneous* server. Here "homogeneous" includes the special case of installing Sun FORTRAN onto a machine for use by that machine only.

• *Standalone* — If your workstation has its own local (directly connected) hard disk, and it does *not* need a file server to boot, then it is a *standalone* workstation.

• *Local/Remote* tape drive — If your machine has its own tape drive, then that drive is a *local* tape drive. If the tape drive is on some other machine (available through the network), then it is a *remote* tape drive. In the following instructions, the Sun that is remote (and has a tape drive) is named "`sunremote`" and the local one is named "`sunlocal`".

Using the Installation Script* on a Sun-3 or Sun-4

For a *local* tape, go to page 3, Case 1 or Case 2, depending on your configuration.

• **For Remote Tape Only**

- To avoid permission problems, make sure the hostname of your local machine is in the remote machine's `/.rhosts` file. To check this file, login to the remote machine and display the file. For example:

```
sunremote% cat /.rhosts
sunlocal
aphrodite sappho
sunremote%
```

If you need to modify this file, do it as follows:

```
sunremote% su root
Password: root-password
sunremote#
sunremote# invoke your favorite editor, change and save the file
sunremote# exit
sunremote%
```

- If your system is running the Yellow Pages, `yp`, the hostnames of both machines must be in the Yellow Pages' `/etc/hosts` file. If either machine is not running the Yellow Pages, make sure an entry for the other machine exists in the `/etc/hosts` file of the non-Yellow-Pages machine. Now follow the instructions for Case 1 or Case 2, depending on your configuration.

* If you need to run two versions of Sun FORTRAN, then *before* you install Sun FORTRAN 1.2 go through the preliminary procedure described under *Two Versions of FORTRAN* near the end of this document.

- **For Both Local and Remote Tape**
  - **Case 1:** Installing onto a *homogeneous* server or onto a standalone (Sun-3 or Sun-4)
    - Insert the tape for your machine (Sun-3 or Sun-4) into the tape drive.
    - Login as `root` and run the installation script.

```
demo% su root
Password: root-password
demo# /usr/etc/extract_unbundled
     ... various questions from the installation script ...
demo# ■          Installation for case 1 is now complete.
```

  - **Case 2:** Installing onto a *heterogeneous* server (Sun-3 and Sun-4)

    Run the installation script once for each architecture needed. (You may not need all.)
    - Login to the server as `root`.

```
demo% su root
Password: root-password
demo# ■
```

    - **Sun-4:** Insert the tape for Sun-4 into the tape drive and run the installation script.

```
demo# /usr/etc/extract_unbundled
    ... various questions from the installation script
```

    - When the installation script asks about system type, server type, and Sun-4 client, respond as follows:

```
Enter system type [ standalone | server ]     server
Enter server type [ homo | heter ]            heter
Will this be for a sun4 client?  [ y/n ]      y
  ... other questions ...
  { Sun-4 software installed }
```

    - **Sun-3:** Insert the tape for Sun-3 into the tape drive and run the installation script.

```
demo# /usr/etc/extract_unbundled
    ... various questions from the installation script
```

    - When the installation script asks about system type, server type, and Sun-3 client, respond as follows:

```
Enter system type [ standalone | server ]     server
Enter server type [ homo | heter ]            heter
Will this be for a sun3 client?  [ y/n ]      y
  ... other questions ...
  { Sun-3 software installed }
demo# ■          Installation for case 2 is now complete.
```

- *Remote* install from a ½" tape drive

If you are doing a *remote* install onto a Sun-3, from a ½" tape drive, then after you get the "`Installation Completed`" message, the installation script will idle and you will not get another SunOS prompt until you press a Control C. For example:

```
      ...
Fortran 1.2 ********* Installation Completed **********
      ... here it would idle forever ...
^C
demo# ■
```

Notes

- Converting from SunOS 3.x to SunOS 4.0

  If you recompile part of a program under SunOS 4.0, be sure to recompile all source files for that program under SunOS 4.0, including user-defined libraries.

- Consistent **-f***float_option* Usage

  If you compile or link part of a program with a particular **-f***float_option*, be sure to compile all files for that program with the same option, including user-defined libraries. If you do linking with a separate command, be sure to link with the same option. For more information, see the *Floating Point Programmer's Guide* and the *Floating-Point Programmer's Guide Addendum*, with the SunOS 4.0 *Software Read This First*, included in the *Programmer's Guides Minibox*, Part number 800-1789.

- Performance

  In general if you want to optimize runntime, you specify both a **-O** option and an inline template library. With Sun FORTRAN 1.2, if you specify an inline template library, you *must* use a new version of `libm.il` rather than the one supplied with SunOS 4.0. Both the new versions are installed with FORTRAN 1.2, and a new version is used as follows:

  On a Sun-4, instead of: `/usr/lib/libm.il`, use: `/usr/lib/f77/libm.il`.
  On a Sun-3, instead of: `/usr/lib/f ... libm.il`, *use:* `/usr/lib/f77/f ... /libm.il`.

  A typical Sun-4 compilation for maximum performance, assuming that double precision data is suitably aligned, would be:

```
demo% f77 -O3 /usr/lib/f77/libm.il -dalign prog.f
```

  and for a Sun-3 with a Floating-Point Accelerator:

```
demo% f77 -O3 -ffpa /usr/lib/f77/ffpa/libm.il prog.f
```

  To see what floating-point option you have, type: `/usr/etc/foption`.

  Some of the new templates in the Sun-4 `/usr/lib/f77/libm.il` contain `sqrt`[sd] instructions to optimize performance with matching new hardware that implements these instructions. These templates can cause a significant boost with the new hardware. If they are used with older WEITEK* 1164/5 hardware there may be a significant performance degradation. In that case, `/usr/lib/f77/libm.il`

* WEITEK is a trademark of the WEITEK Corporation.

should be copied to another file, such as `/usr/lib/f77/nosqrt.il`, and the templates containing the `sqrt` should be edited out. Then programs compiled with `/usr/lib/f77/nosqrt.il` will provide satisfactory performance with both new hardware and old hardware.

For further information about inline expansion, consult the *Floating-Point Programmers Guide* accompanying SunOS 4.0 and its separate Addendum, and `inline(1)`.

- The FORTRAN Version Number

  Some users want to know the FORTRAN version number, although most do not need it. You can usually display this number as indicated in either of these examples:

```
demo% what /usr/lib/f77pass1 | grep RELEASE
        RELEASE f77pass1                release 1.2 FCS
demo%
```

or

```
demo% cat /usr/lib/lang_info | grep fortran
fortran 1.2     4.0
demo%
```

- The Binding Mode Options **`-Bstatic`** and **`-Bdynamic`**

  If you specify the *binding mode* on the `f77` or `ld` command line, the selected mode applies to all *subsequent* object files until the next `-Bdynamic` or `-Bstatic` option. Therefore, if you want `-Bstatic` to apply to *all* object files, put the `-Bstatic` option first. Similarly for `-Bdynamic`. For more information, see `ld`(1), online or in the *SunOS Reference Manual.*

Known Bugs for the Loader

- Position-Independent Code for a Large Model on a Sun-4

  For a Sun-4 only, if you compile with the large-model position-independent code option (**`-PIC`**), then the loader sometimes produces an executable file that will core dump when executed. For example:

```
demo% f77 test.f -PIC
demo% a.out
Segmentation fault (core dumped)
demo%
```

Known Bugs for `f77cvt`

- `f77cvt` does not recognize VMS constants of the form `O`*nnn and* `Z`*nnn* . For example:

```
DATA  MARK / o17 /,  FORMFEED / z0c /
```

The character before the "17" is a letter, not a numeral. These VMS octal and hexadecimal forms are treated as variables by `f77cvt` and by `f77`. They usually result in an ill-formed program and a diagnostic message, such as:

```
        impossible tag error
```

Workaround: Use one of the documented VMS forms:

```
DATA MARK / "17 /
DATA MARK / '17'o /   FORMFEED / '0c'x /
```

- `f77cvt` does not convert VMS typeless constants in a `PARAMETER` statement. For example:

```
demo% cat par1.for
PARAMETER X = '00000008'X
END
demo%  f77cvt par1.for
par1.for:
 MAIN:
Translator error line 1 of par1.for: Impossible type bitstr in routine print_type
demo%
```

Known Bugs for `dbx`

- `dbx` loses track of actual parameters that are not referenced.

  Workaround: Be sure to reference all actual parameters.

- `dbx` can get confused by subprograms with entry points.

  Workaround: Avoid subprograms with entry points.

- You Cannot Use the `dbx` Debugger on `f77cvt` Output

  The `f77cvt` output file is valid Sun FORTRAN source, but `dbx` cannot cope with it. There is an obscure diagnostic. For example:

```
demo% f77cvt test.for
demo% f77 -o test -g test.f
demo% dbx test
Reading symbolic information...
Read 252 symbols
(dbx) stop in MAIN
(1) stop in MAIN
(dbx) run
Running: a.out
can't write to process (address 0x0)
(perhaps you are already debugging this file?)
(dbx) quit
demo%
```

Known Bugs for `f77`

- The `ioinit` function requires redundant loading of `lI77`

  If you use `ioinit` (3F), you must explicitly specify `-lI77` in the compile or load command. For example:

```
demo% f77 file.f -lI77
```

  or

```
demo% ld file.o -lI77
```

- Trigonometric functions for multiples of 360 degrees are incorrect and slow.

  The functions `sind`, `cosd`, and `tand` reduce arguments incorrectly and slowly. For example, if `x` is a very large multiple of 360 degrees, where you should get zero for `sind(x)`, you can get small numbers such as `-6.57D-08` or even `-0.5`.

- For a Sun-4 only, the entry point and its function must have the same type.

  The wrong value will be returned if the function is real and the entry is integer.

  Workaround: Do not write a function of one type with an entry point of a different type.

- A COMMON block can't be initialized in more than one program.

  VMS FORTRAN allows one part of a COMMON block to be initialized in one subprogram and another part initialized in another subprogram. The parts must be non-overlapping. The equivalent code provokes a linker diagnostic in Sun FORTRAN. For example:

```
demo% cat ini1.f
C ini1.f
      COMMON / MATH / E, PI
      DATA PI / 3.14159 /
      CALL INIT
      PRINT *, E, PI
      END
demo% cat ini2.f
C ini2.f
      SUBROUTINE INIT
      COMMON / MATH / E, PI
      DATA E / 2.71828 /
      RETURN
      END
demo% f77 ini1.f ini2.f
      ...
Linking:
_math_: ld: ini2.o: multiply defined
demo%
```

Workaround: Initialize COMMON in a BLOCKDATA subprogram, or put both sources into the same file and compile that single combined file.

- For a Sun-4/110 only, the optimizer generates code for floating-point hardware that is an option.

  The Sun-4/110 does not include floating point hardware in the base product: the Floating-Point Unit is optional. This can make the optimization of large programs with many basic blocks be about 50% slower on a Sun-4/110 without FPU than on the same machine type with an FPU.

  Workaround: Purchase the optional FPU, or do not optimize.

- The optimizer gets confused on function calls with equivalenced arrays .

  If you use function calls to change or reference elements of equivalenced arrays, and if this is in a loop, then the optimizer incorrectly moves code to outside of the loop.

  Workaround: Make the equivalenced variables scalars, or do not equivalence them, or use a subroutine instead of a function, or do not optimize.

Documentation Errata

- Size Error in Concatenate Example

  On page 31 of the *Sun FORTRAN Programmer's Guide*, the first line must be:

```
CHARACTER a*4, b*2, c*12
```

- The **-dalign** option

  On page 255 of the *Sun FORTRAN Programmer's Guide*, under `-dalign`, insert this statement: "This option applies to uninitialized data only."

Two Versions of FORTRAN

This section shows how to have two versions of FORTRAN at the same time. Most users do not need this. Sun does *NOT* support the use of both versions of FORTRAN on one system. Test this before you install.

- **Save the Old Version**

  Usually the partition for `/usr` fills up first. Save the old version wherever there is enough room. The following examples use `/home`, which is usually in a different partition.
  Login as superuser, change directory to `root`, and create a directory for the old version.

```
demo% su
Password:  .. your own root password ...
demo# cd /
demo# mkdir /home/oldf77
demo# mkdir /home/oldf77/lib
demo# mkdir /home/oldf77/misalign
demo#
```

  Move the files that will be replaced into the directory you just created:

```
demo# mv /usr/bin/{f77*,ratfor}              /home/oldf77
demo# mv /usr/ucb/{fsplit,fpr}               /home/oldf77
demo# mv /usr/lib/{lib?77.a,lib?77_p.a,libpfc*}   /home/oldf77
demo# ranlib -t /home/oldf77/lib*.a

demo# mv /usr/lib/misalign/lib?77*  /home/oldf77/misalign {Sun-4 only}
demo# ranlib -t /home/oldf77/misalign/lib*.a              {Sun-4 only}

demo# cd /usr/lib
demo# tar cf - f77* | (cd /home/oldf77/lib ; tar xf - )
demo# mv /home/oldf77/lib/f77pass1    /home/oldf77/lib/f77
```

  If you installed man pages onto your workstation, rather than mounting from a server, then do:

```
demo# mv /usr/man/man1/f77.1 /home/oldf77  ("man1" and "f77.1" end in the numeral one.)
```

- **Run Two Versions**

  Here is a way to make special commands to run another `f77`. For example: If the old compiler components are in the `/home/oldf77`, directory, then insert the lines:

```
alias of77 "/home/oldf77/f77 -Qpath /home/oldf77/lib/f77 -L/home/oldf77"
alias mf77 "/home/oldf77/f77 -Qpath /home/oldf77/lib/f77 \
        -misalign -L/home/oldf77/misalign"
```

  into the `.cshrc` file, then `source` that file. `of77` runs the old FORTRAN compiler, and `mf77` runs it with `-misalign`. The `-Qpath` option of `f77` specifies the directory which will be searched first for the compiler components. The `-L` option specifies the directory which will be searched for the libraries.

Sun FORTRAN 1.2 Installation Tape Contents

The tape contains the following files:

File 1: Copyright notice
File 2: Installation scripts
File 3: A tar file of the Sun FORTRAN 1.2 software
File 4: Another copyright notice

A simple list of the Sun FORTRAN 1.2 files is provided below. For a more detailed list, insert the tape into the tape drive and list the contents of the above `tar` file as follows:

```
demo% mt -f /dev/rst0 rew
demo% mt -f /dev/nrst0 fsf 2     The 'st0' is for ¼ " cartridge tape
demo% tar tvf /dev/nrst0         For ½ " reel tape, use 'mt0'
```

The names, location, and number of files may vary slightly with architecture and release.

```
/usr/ucb/
/usr/ucb/fpr
/usr/ucb/fsplit

/usr/bin/
/usr/bin/f77
/usr/bin/f77cvt
/usr/bin/ratfor

/usr/lib/
/usr/lib/f77/
/usr/lib/f77/cg
/usr/lib/f77/iropt
/usr/lib/f77/as
/usr/lib/f77/c2                  (Sun-3 only)

/usr/lib/f77/libm.il             (Sun-4 only)
/usr/lib/f77/fsoft/libm.il       (Sun-3 only)
/usr/lib/f77/fswitch/libm.il     (Sun-3 only)
/usr/lib/f77/f68881/libm.il      (Sun-3 only)
/usr/lib/f77/ffpa/libm.il        (Sun-3 only)

/usr/lib/f77pass1
/usr/lib/libF77.a
/usr/lib/libF77_p.a
/usr/lib/libI77.a
/usr/lib/libI77_p.a
/usr/lib/libU77.a
/usr/lib/libU77_p.a
/usr/lib/libV77.a
/usr/lib/libV77_p.a
/usr/lib/libpfc_p.a
/usr/lib/libpfc.a
```

... *continued on next page* ...

```
/usr/lib/misalign/libF77.a      (Sun-4 only)
/usr/lib/misalign/libF77_p.a    (Sun-4 only)
/usr/lib/misalign/libI77.a      (Sun-4 only)
/usr/lib/misalign/libI77_p.a    (Sun-4 only)
/usr/lib/misalign/libU77.a      (Sun-4 only)
/usr/lib/misalign/libU77_p.a    (Sun-4 only)
/usr/lib/misalign/libV77.a      (Sun-4 only)
/usr/lib/misalign/libV77_p.a    (Sun-4 only)

/usr/include/
/usr/include/f77/
/usr/include/f77/f77_floatingpoint.h

/usr/share/
/usr/share/man/
/usr/share/man/man1/
/usr/share/man/man1/f77.1
/usr/share/man/man1/f77cvt.1
/usr/share/man/man1/fpr.1
/usr/share/man/man1/fsplit.1
/usr/share/man/man1/ratfor.1

/usr/share/man/man3/
/usr/share/man/man3/abort.3f
/usr/share/man/man3/access.3f
/usr/share/man/man3/alarm.3f
/usr/share/man/man3/bessel.3f
/usr/share/man/man3/bit.3f
/usr/share/man/man3/chdir.3f
/usr/share/man/man3/chmod.3f
/usr/share/man/man3/etime.3f
/usr/share/man/man3/exit.3f
/usr/share/man/man3/f77_floatingpoint.3f
/usr/share/man/man3/f77_ieee_environment.3f
/usr/share/man/man3/fdate.3f
/usr/share/man/man3/flush.3f
/usr/share/man/man3/fork.3f
/usr/share/man/man3/fortran.3f
/usr/share/man/man3/free.3f
/usr/share/man/man3/fseek.3f
/usr/share/man/man3/getarg.3f
/usr/share/man/man3/getc.3f
/usr/share/man/man3/getcwd.3f
```

... *continued on next page* ...

```
/usr/share/man/man3/getenv.3f
/usr/share/man/man3/getfd.3f
/usr/share/man/man3/getlog.3f
/usr/share/man/man3/getpid.3f
/usr/share/man/man3/getuid.3f
/usr/share/man/man3/hostnm.3f
/usr/share/man/man3/idate.3f
/usr/share/man/man3/index.3f
/usr/share/man/man3/intro.3f
/usr/share/man/man3/ioinit.3f
/usr/share/man/man3/kill.3f
/usr/share/man/man3/len.3f
/usr/share/man/man3/libm_double.3f
/usr/share/man/man3/libm_single.3f
/usr/share/man/man3/link.3f
/usr/share/man/man3/loc.3f
/usr/share/man/man3/long.3f
/usr/share/man/man3/malloc.3f
/usr/share/man/man3/perror.3f
/usr/share/man/man3/putc.3f
/usr/share/man/man3/qsort.3f
/usr/share/man/man3/rand.3f
/usr/share/man/man3/range.3f
/usr/share/man/man3/rename.3f
/usr/share/man/man3/signal.3f
/usr/share/man/man3/sleep.3f
/usr/share/man/man3/stat.3f
/usr/share/man/man3/system.3f
/usr/share/man/man3/time.3f
/usr/share/man/man3/topen.3f
/usr/share/man/man3/ttynam.3f
/usr/share/man/man3/unlink.3f
/usr/share/man/man3/wait.3f
```