AdvanSys (Advanced System Products, Inc.) manufactures the following
RISC-based, Bus-Mastering, Fast (10 Mhz) and Ultra (20 Mhz) Narrow
(8-bit transfer) SCSI Host Adapters for the ISA, EISA, VL, and PCI
buses and RISC-based, Bus-Mastering, Ultra (20 Mhz) Wide (16-bit
transfer) SCSI Host Adapters for the PCI bus.

The CDB counts below indicate the number of SCSI CDB (Command
Descriptor Block) requests that can be stored in the RISC chip
cache and board LRAM. A CDB is a single SCSI command. The driver
detect routine will display the number of CDBs available for each
adapter detected. The number of CDBs used by the driver can be
lowered in the BIOS by changing the 'Host Queue Size' adapter setting.

Laptop Products:
    ABP-480 - Bus-Master CardBus (16 CDB)

Connectivity Products:
    ABP510/5150 - Bus-Master ISA (240 CDB)
    ABP5140 - Bus-Master ISA PnP (16 CDB)
    ABP5142 - Bus-Master ISA PnP with floppy (16 CDB)
    ABP902/3902 - Bus-Master PCI (16 CDB)
    ABP3905 - Bus-Master PCI (16 CDB)
    ABP915 - Bus-Master PCI (16 CDB)
    ABP920 - Bus-Master PCI (16 CDB)
    ABP3922 - Bus-Master PCI (16 CDB)
    ABP3925 - Bus-Master PCI (16 CDB)
    ABP930 - Bus-Master PCI (16 CDB)
    ABP930U - Bus-Master PCI Ultra (16 CDB)
    ABP930UA - Bus-Master PCI Ultra (16 CDB)
    ABP960 - Bus-Master PCI MAC/PC (16 CDB)
    ABP960U - Bus-Master PCI MAC/PC Ultra (16 CDB)

Single Channel Products:
    ABP542 - Bus-Master ISA with floppy (240 CDB)
    ABP742 - Bus-Master EISA (240 CDB)
    ABP842 - Bus-Master VL (240 CDB)
    ABP940 - Bus-Master PCI (240 CDB)
    ABP940U - Bus-Master PCI Ultra (240 CDB)
    ABP940UA/3940UA - Bus-Master PCI Ultra (240 CDB)
    ABP970 - Bus-Master PCI MAC/PC (240 CDB)
    ABP970U - Bus-Master PCI MAC/PC Ultra (240 CDB)
    ABP3960UA - Bus-Master PCI MAC/PC Ultra (240 CDB)
    ABP940UW/3940UW - Bus-Master PCI Ultra-Wide (253 CDB)
    ABP970UW - Bus-Master PCI MAC/PC Ultra-Wide (253 CDB)
    ABP3940U2W - Bus-Master PCI LVD/Ultra2-Wide (253 CDB)

Multi-Channel Products:
    ABP752 - Dual Channel Bus-Master EISA (240 CDB Per Channel)
    ABP852 - Dual Channel Bus-Master VL (240 CDB Per Channel)
    ABP950 - Dual Channel Bus-Master PCI (240 CDB Per Channel)
    ABP950UW - Dual Channel Bus-Master PCI Ultra-Wide (253 CDB Per Channel)
    ABP980 - Four Channel Bus-Master PCI (240 CDB Per Channel)
    ABP980U - Four Channel Bus-Master PCI Ultra (240 CDB Per Channel)
    ABP980UA/3980UA - Four Channel Bus-Master PCI Ultra (16 CDB Per Chan.)
    ABP3950U2W - Bus-Master PCI LVD/Ultra2-Wide and Ultra-Wide (253 CDB)
    ABP3950U3W - Bus-Master PCI Dual LVD2/Ultra3-Wide (253 CDB)

Driver Compile Time Options and Debugging

The following constants can be defined in the source file.

1. ADVANSYS_ASSERT - Enable driver assertions (Def: Enabled)

   Enabling this option adds assertion logic statements to the
   driver. If an assertion fails a message will be displayed to
   the console, but the system will continue to operate. Any
   assertions encountered should be reported to the person
   responsible for the driver. Assertion statements may proactively
   detect problems with the driver and facilitate fixing these
   problems. Enabling assertions will add a small overhead to the
   execution of the driver.

2. ADVANSYS_DEBUG - Enable driver debugging (Def: Disabled)

   Enabling this option adds tracing functions to the driver and the
   ability to set a driver tracing level at boot time. This option is
   very useful for debugging the driver, but it will add to the size
   of the driver execution image and add overhead to the execution of
   the driver.

   The amount of debugging output can be controlled with the global
   variable 'asc_dbglvl'. The higher the number the more output. By
   default the debug level is 0.

   If the driver is loaded at boot time and the LILO Driver Option
   is included in the system, the debug level can be changed by
   specifying a 5th (ASC_NUM_IOPORT_PROBE + 1) I/O Port. The
   first three hex digits of the pseudo I/O Port must be set to
   'deb' and the fourth hex digit specifies the debug level: 0 - F.
   The following command line will look for an adapter at 0x330
   and set the debug level to 2.

       linux advansys=0x330,0,0,0,0xdeb2

   If the driver is built as a loadable module this variable can be
   defined when the driver is loaded. The following insmod command
   will set the debug level to one.

       insmod advansys.o asc_dbglvl=1

   Debugging Message Levels:
       0: Errors Only
       1: High-Level Tracing
       2-N: Verbose Tracing

   To enable debug output to console, please make sure that:

   a. System and kernel logging is enabled (syslogd, klogd running).
   b. Kernel messages are routed to console output. Check
      /etc/syslog.conf for an entry similar to this:

          kern.*   /dev/console

   c. klogd is started with the appropriate -c parameter
      (e.g. klogd -c 8)

   This will cause printk() messages to be be displayed on the
   current console. Refer to the klogd(8) and syslogd(8) man pages
   for details.

   Alternatively you can enable printk() to console with this
   program. However, this is not the 'official' way to do this.
   Debug output is logged in /var/log/messages.

      main()
      {
              syscall(103, 7, 0, 0);
      }

   Increasing LOG_BUF_LEN in kernel/printk.c to something like
   40960 allows more debug messages to be buffered in the kernel
   and written to the console or log file.

3. ADVANSYS_STATS - Enable statistics (Def: Enabled)

   Enabling this option adds statistics collection and display
   through /proc to the driver. The information is useful for
   monitoring driver and device performance. It will add to the
   size of the driver execution image and add minor overhead to
   the execution of the driver.

   Statistics are maintained on a per adapter basis. Driver entry
   point call counts and transfer size counts are maintained.
   Statistics are only available for kernels greater than or equal
   to v1.3.0 with the CONFIG_PROC_FS (/proc) file system configured.

   AdvanSys SCSI adapter files have the following path name format:

      /proc/scsi/advansys/{0, 1, 2, 3, ... }

   This information can be displayed with cat. For example:

      cat /proc/scsi/advansys/0

   When ADVANSYS_STATS is not defined the AdvanSys /proc files only
   contain adapter and device configuration information.

Driver LILO Option

If init/main.c is modified as described in the 'Directions for Adding
the AdvanSys Driver to Linux' section (B.4.) above, the driver will
recognize the 'advansys' LILO command line and /etc/lilo.conf option.
This option can be used to either disable I/O port scanning or to limit
scanning to 1 - 4 I/O ports. Regardless of the option setting EISA and
PCI boards will still be searched for and detected. This option only
affects searching for ISA and VL boards.

Examples:

```
  1. Eliminate I/O port scanning:
         boot: linux advansys=
            or
         boot: linux advansys=0x0
  2. Limit I/O port scanning to one I/O port:
         boot: linux advansys=0x110
  3. Limit I/O port scanning to four I/O ports:
         boot: linux advansys=0x110,0x210,0x230,0x330
```

For a loadable module the same effect can be achieved by setting
the 'asc_iopflag' variable and 'asc_ioport' array when loading
the driver, e.g.

        insmod advansys.o asc_iopflag=1 asc_ioport=0x110,0x330

If ADVANSYS_DEBUG is defined a 5th (ASC_NUM_IOPORT_PROBE + 1)
I/O Port may be added to specify the driver debug level. Refer to
the 'Driver Compile Time Options and Debugging' section above for
more information.

Credits (Chronological Order)

Bob Frey <bfrey@turbolinux.com.cn> wrote the AdvanSys SCSI driver
and maintained it up to 3.3F. He continues to answer questions
and help maintain the driver.

Nathan Hartwell <mage@cdc3.cdc.net> provided the directions and
basis for the Linux v1.3.X changes which were included in the
1.2 release.

Thomas E Zerucha <zerucha@shell.portal.com> pointed out a bug
in advansys_biosparam() which was fixed in the 1.3 release.

Erik Ratcliffe <erik@caldera.com> has done testing of the
AdvanSys driver in the Caldera releases.

Rik van Riel <H.H.vanRiel@fys.ruu.nl> provided a patch to
AscWaitTixISRDone() which he found necessary to make the
driver work with a SCSI-1 disk.

Mark Moran <mmoran@mmoran.com> has helped test Ultra-Wide
support in the 3.1A driver.

Doug Gilbert <dgilbert@interlog.com> has made changes and
suggestions to improve the driver and done a lot of testing.

Ken Mort <ken@mort.net> reported a DEBUG compile bug fixed
in 3.2K.

Tom Rini <trini@kernel.crashing.org> provided the CONFIG_ISA
patch and helped with PowerPC wide and narrow board support.

Philip Blundell <philb@gnu.org> provided an
advansys_interrupts_enabled patch.

Dave Jones <dave@denial.force9.co.uk> reported the compiler

warnings generated when CONFIG_PROC_FS was not defined in
the 3.2M driver.

Jerry Quinn <jlquinn@us.ibm.com> fixed PowerPC support (endian
problems) for wide cards.

Bryan Henderson <bryanh@giraffe-data.com> helped debug narrow
card error handling.

Manuel Veloso <veloso@pobox.com> worked hard on PowerPC narrow
board support and fixed a bug in AscGetEEPConfig().

Arnaldo Carvalho de Melo <acme@conectiva.com.br> made
save_flags/restore_flags changes.

Andy Kellner <AKellner@connectcom.net> continued the Advansys SCSI
driver development for ConnectCom (Version > 3.3F).

Ken Witherow for extensive testing during the development of version 3.4.