Release Note

SI / XIO   (SLXOS) for SCO Open Server 5.0

Introduction

This release note details the latest version of the Specialix SLXOS (SI/XIO) driver for SCO Open Server 5.0. The Specialix driver is Release 1.2.0.

If you are upgrading from release 1.00, you will need to remove the old software; see section 'Removing SLXOS from SCO Open Server 5'.

The latest driver supports PCI and EISA host cards. It also supports a new type of ISA host card, as well as continuing to support the existing ISA card. You can place a maximum of four SLXOS host cards in a machine; the cards can be any mix of bus types. Microchannel cards are also supported.

You can attach terminal adapters (TAs or MTAs) in exactly the same way as before. If you are new to the Specialix range of products, details on terminal adapters are contained in earlier documentation accompanying this release note.

If you need to test the host card, TAs or MTAs you must use v4.0 or above of the Hardware Diagnostics program. This program can be found on the DOS Utilities disk supplied to you with the product.

This release note amplifies/changes the information in the SLXOS Manual:

SLXOS Guide to Installation & Operation, 3rd edition.

Contents

The contents of this release note is as follows:

- Known problems

- TTY numbering conventions

- Removing SLXOS from SCO Open Server 5

- Installing SLXOS

- Port Diagnostics

- Port Statistics

- Establishing ISA host card type

- Installation of ISA host cards: Type A and Type B

- Technical Support

- Repairs

Known problems

The following problems are known to exist at the time of writing

- When using the character based CHARM interface for installation/modification of cards you **cannot** set the interrupt mode that cards will run in due to a bug in the way the initialisation program interprets toggle button keystrokes. Using the CHARM interface will result in the cards being installed in polled mode, no matter what position the IRQ mode toggle button is in. It is recommended that cards should be installed via the X/Motif interface if they are required to run in interrupt mode.

TTY numbering conventions   *****important*****

Please note one very important change in this release of the driver. The tty numbering convention used in previous versions of the driver labelled tty devices ttya0 to ttyaN. This release has been altered to label devices ttya1 to ttya(N+1). It is strongly recommended that you remove any previous versions of the driver before installing V1.2.0.

Removing SLXOS from SCO Open Server 5

To remove the previous version of the SLXOS driver from SCO Open Server 5, log in as root and either, in graphical mode:

- From the System Administration folder, go into Software Manager. From the software menu, select your existing version of the SLXOS driver. Then choose 'Remove Software'.

or, in text mode:

- enter the command `custom -p Specialix:sixio -r`

Your machine will remove the software and then re-link the kernel. After this, re-boot your machine. You can also use this method to remove this new version of the SLXOS driver.

Installing SLXOS

From the SCO desktop select the system administration folder (in text mode type 'custom', which will take you to the Software Manager screen).

- Select Software Manager.

- In the Software Manager, select the software menu, and menu item install new.

- Select the name of your machine

- Select the appropriate drive on your machine (floppy, CD-ROM).

- Select full installation

Your machine will install the SLXOS driver onto your hard disk. When complete, you will be presented with the following screen (in graphical mode):
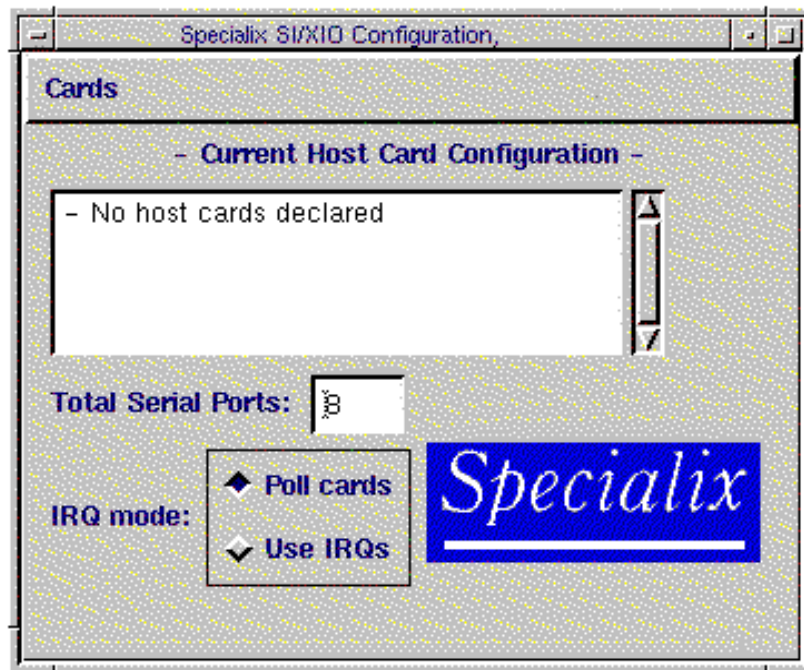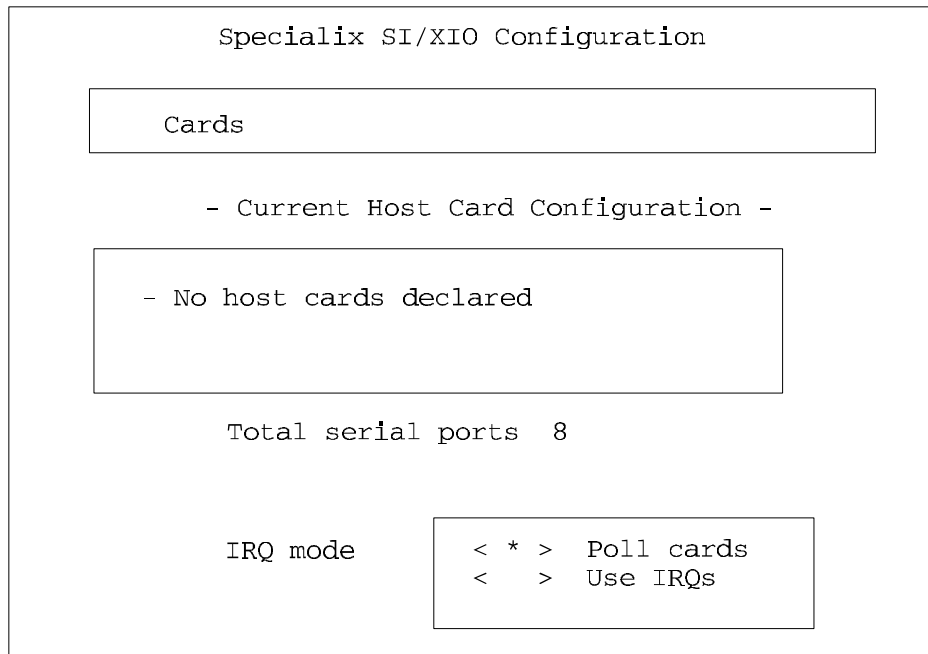
screen presented after driver installation

If you have cards other than ISA cards present on your system (eg PCI cards) they will be automatically detected and displayed in the configuration window.

If you have an ISA card :

- From the Cards menu select 'Add Card', you will be presented with an address and interrupt. Accept or change as necessary; as a general rule, you must avoid clashes with other devices in your machine. Note the settings, as you will need to adjust switches on the card's surface.

- Then returning to the Specialix host card screen (shown above), enter the number of serial ports that you want and click on 'Done'. You can accept the default IRQ mode of 'Poll cards'.

```
                    Specialix SI/XIO Configuration

        ┌─────────────────────────────────────────────────────┐
        │  Cards                                               │
        └─────────────────────────────────────────────────────┘


              - Current Host Card Configuration -

        ┌─────────────────────────────────────────────────────┐
        │  - No host cards declared                            │
        │                                                      │
        │                                                      │
        │                                                      │
        └─────────────────────────────────────────────────────┘


              Total serial ports  8



                              ┌───────────────────────────┐
        IRQ mode              │  < * >   Poll cards        │
                              │  <   >   Use IRQs          │
                              └───────────────────────────┘
```

equivalent text screen presented

If you are installing a PCI, EISA or Microchannel card

- Enter the number of serial ports you want and click on 'Done'. You can accept the default IRQ mode of 'Poll cards'. (You do not need to go to the Cards menu).

Your machine will re-link your kernel. After this, you should see a reference to the SLXOS software, e.g. 'Specialix SI/XIO Serial support', in the software manager screen.

- Exit the Software Manager, which returns you to the System Administration window (screen).

- Shutdown your machine.

Once your machine is shutdown, you must install your cards and attach the terminal adapters. Reboot the machine and do the following:

- If you have an EISA card, or an ISA card in an EISA machine, you will need to run your machine's motherboard configuration utility. This will declare your card to the motherboard. Re-boot your machine.

- If you have an ISA card, be aware there are TWO types of ISA card: Type A and Type B. Refer to the Section 'Establishing ISA Host Card Type'. Then refer to the Section 'Installation of an ISA Card' for either Type A or Type B.

- If you have a PCI card you may boot to UNIX immediately, PCI cards need no further configuration, continue with the instructions immediately following this paragraph.

You should now have a folder inside the System Administration Window called 'Specialix Serial'; (if you are in text mode, type 'scoadmin'). See screenshots below.:

The Specialix Serial folder opening window
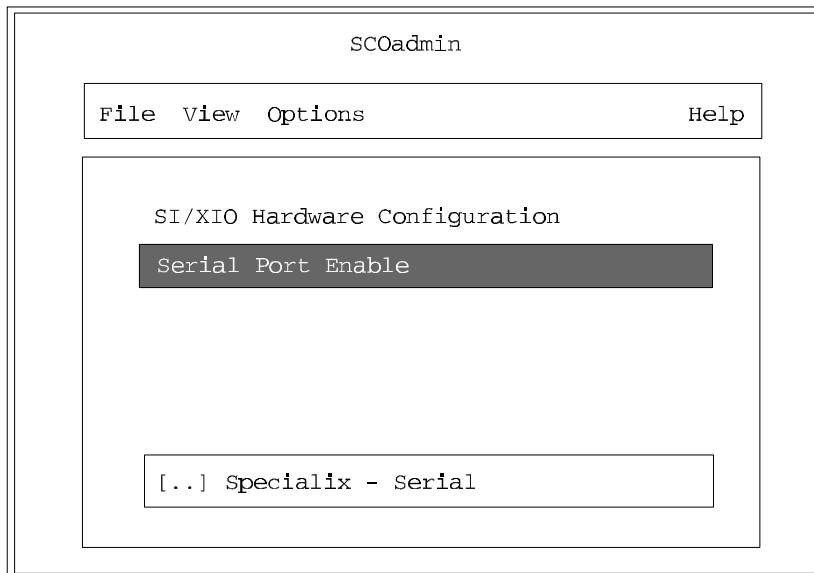
Equivalent text screen presented

Enabling serial ports

From the Specialix serial folder, select Serial Port Enable. You will be presented with the following screen (examples show screens after enabling some ports) :

Show Locks Change Stats Diags				
Lk St Ty Port Runlevel	gettydefs	VT Type	Force	
1 234	9600	unknown		
2 234	9600	unknown		
3 234	9600	unknown		
4 234	9600	unknown		
5 234	19200	unknown		
6 234	9600	unknown		
7 234	9600	unknown		
8 234	9600	unknown		

– Run Levels –

☐ 2 ☐ 3 ☐ 4

☐ ixon   ☐ ixany

gettydefs

19200
9600
4800
2400

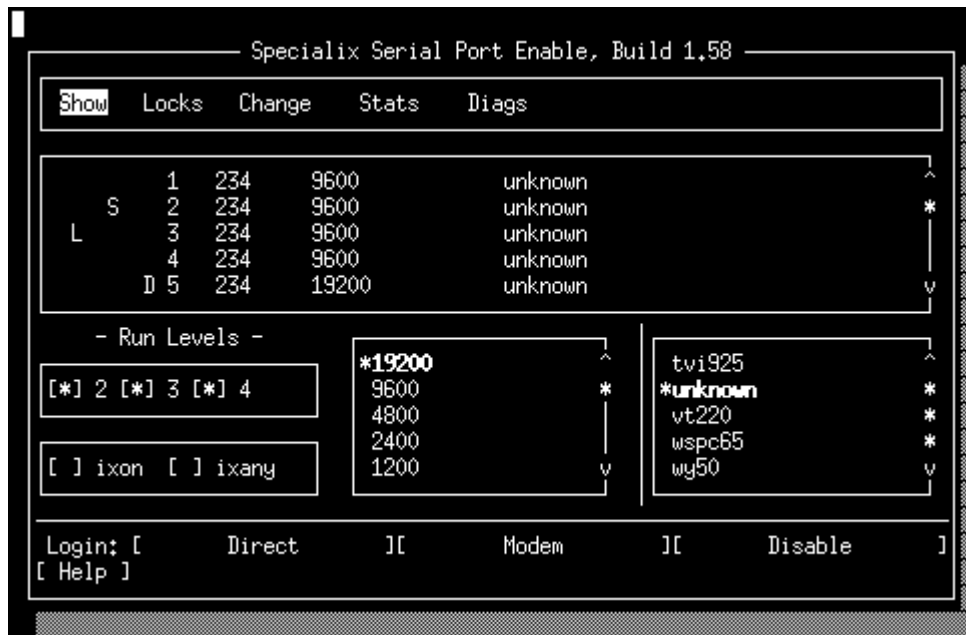**VT type**

tvi925
unknown
vt220
wspc65

Login: Direct   Modem   Disable

Help   *Specialix*

example Serial port Enable dialog box

As you move the mouse over the screen you will see help messages; you also have on-line help. Use this help to configure your card. In the text mode you also have on-line help. To save any changes, select the 'Show' menu and then 'Exit'.
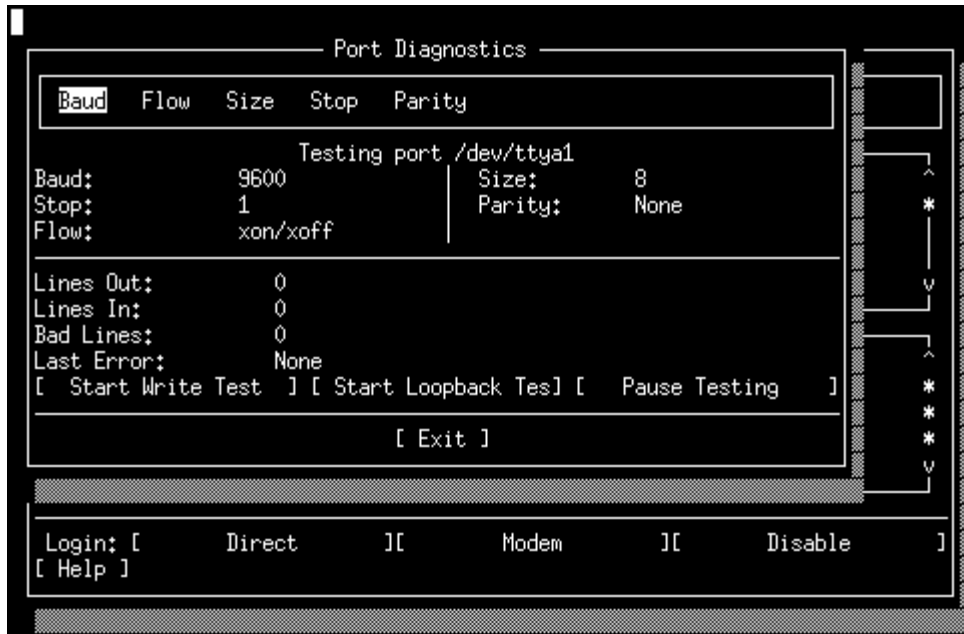
```
┌──────────── Specialix Serial Port Enable, Build 1.58 ────────────┐
│                                                                    │
│  ┌────────────────────────────────────────────────────────────┐   │
│  │ Show   Locks   Change   Stats   Diags                       │   │
│  └────────────────────────────────────────────────────────────┘   │
│  ┌──────────────────────────────────────────────────────────┐┐    │
│  │        1    234     9600          unknown               ^ │     │
│  │    S   2    234     9600          unknown               * │     │
│  │  L     3    234     9600          unknown                 │     │
│  │        4    234     9600          unknown                 │     │
│  │      D 5    234    19200          unknown               v │┘    │
│  └──────────────────────────────────────────────────────────┘     │
│    - Run Levels -     ┌──────────────┐┐  ┌───────────────┐┐        │
│                       │ *19200     ^ │   │ tvi925      ^ │         │
│  ┌──────────────────┐ │  9600      * │   │ *unknown    * │         │
│  │[*] 2 [*] 3 [*] 4 │ │  4800        │   │  vt220      * │         │
│  └──────────────────┘ │  2400        │   │  wspc65     * │         │
│                       │  1200      v │┘  │  wy50       v │┘        │
│  ┌──────────────────┐ └──────────────┘   └───────────────┘        │
│  │ [ ] ixon  [ ] ixany│                                            │
│  └──────────────────┘                                              │
│                                                                    │
│  Login: [    Direct    ][    Modem    ][    Disable    ]           │
│  [ Help ]                                                          │
└────────────────────────────────────────────────────────────────────┘
```

equivalent text screen

When you have enabled a serial port using one of these screens, you should see a login on an attached terminal.

Port Diagnostics

The SLXOS online port diagnostics enable you to investigate port failures and while the system is still running. The two variants of the screen are shown below:

```
┌──────────────────────────────────────────────────────────┐
│ Baud Flow Size Stop Parity                                 │
├──────────────────────────────────────────────────────────┤
│                  Testing port /dev/ttya1                   │
│  Baud:        9600            Size:      8                 │
│  Stop:        1               Parity:    None              │
│  Flow:        xon/xoff                                     │
│ ─────────────────────────────────────────────────────────│
│  Lines Out:      0                                         │
│  Lines In:       0                                         │
│  Bad Lines:      0                                         │
│  Last Error:     None                                      │
│   Start Write Test │ Start Loopback Test │ Pause Testing   │
│                     ┌──────┐                               │
│                     │ Exit │                               │
│                     └──────┘                               │
└──────────────────────────────────────────────────────────┘
```

Port Diagnostics Screen

Equivalent text screen

Use Tab and arrows as necessary on a text screen.

Diagnostic tests can only be performed on a closed port. If the port is open, you will be given access to the Port Statistics menu only.

Port Statistics

There are three port statistic utilities:

- View Port Settings—lists the port's current stty attributes under the headings Input, Output, Control, Line Discipline, and Local Driver Flags.

- Port Statistics—Enables you to gather and display statistics for individual ports. This utility counts the number of bytes received, number of bytes transmitted, number of opens, number of closes, and the number of ioctls. Statistics are presented for the period since the last boot, since the last time the statistics were reset, or since the feature was enabled. Statistics gathering is disabled by default, so no statistics will be available the first time you use this feature.

- View Modem Pins—displays the current status (ON/OFF) of the port's modem pins (RTS, CTS, DCD, RI, DTR, and DSR). The display is updated every second.

Port Statistics are accessed from the 'Stats' menu.

Establishing ISA Host Card Type

There are two card types; type A and B. The two types are shown in the figures below:

37-way connector

SI / XIO HOST   COPYRIGHT SPECIALIX INTERNATIONAL   SERIAL NO. nn-nnnnn

CPU

Type Z280

sw1 sw2

rotary switches (only)

Jumper LK1
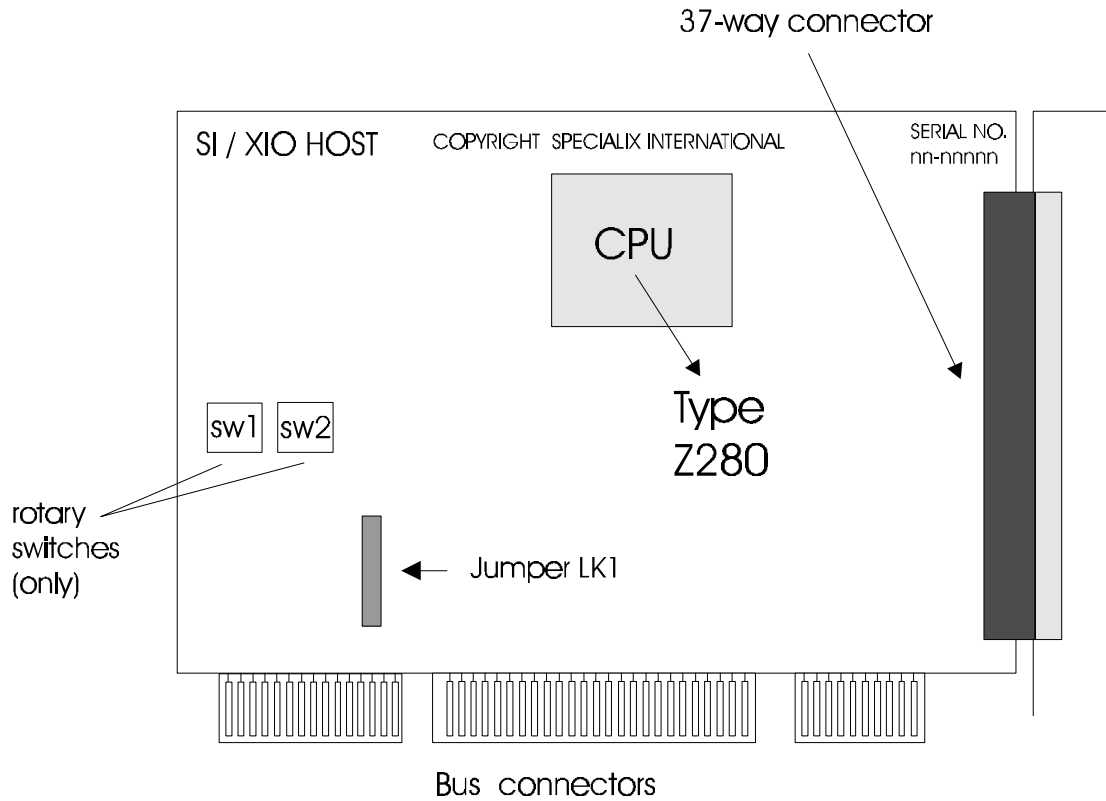
Bus connectors

ISA card Type A

rotary
switches
and jumpers   *or*   10-way
                                             DIL switch

sw1 sw2   1 2 3 4 5 6 7 8 9 10
                     ON
                     OFF

                                                        37-way connector

SI / XIO HOST

COPYRIGHT SPECIALIX

SERIAL NO. _____

                            CPU
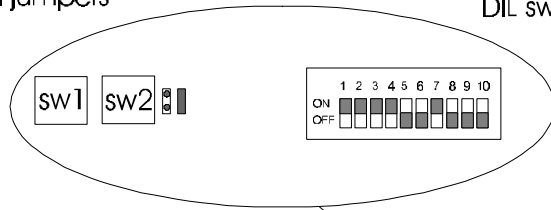
                     Type
                     T225

                              Bus connectors
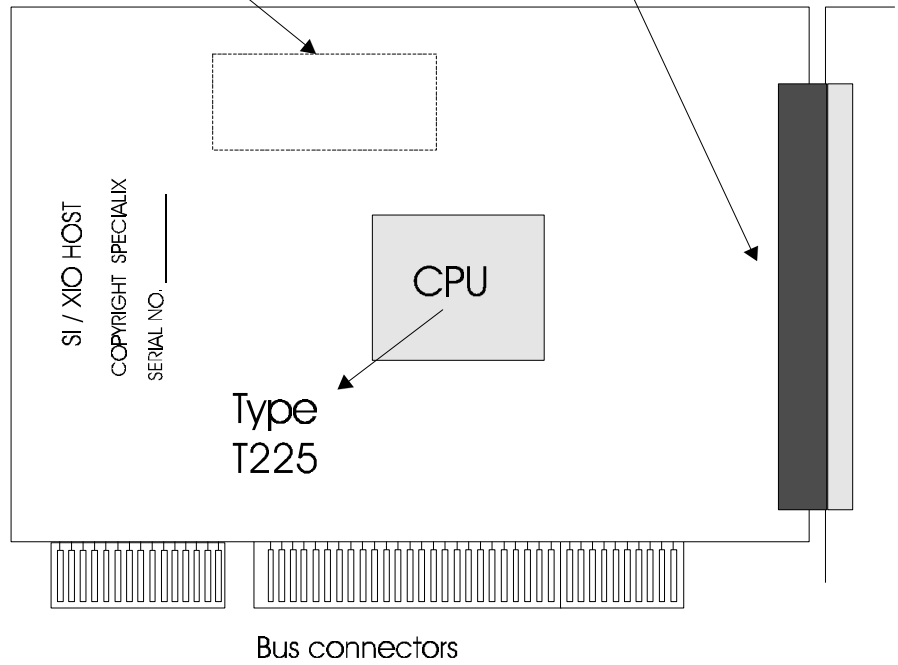
ISA card Type B

Installation of ISA Host Card Type-A

Setting the Memory Address

Set the memory address of the card using the two rotary switches (SW1 and SW2) and jumper (LK1) on the card. The rotary switches select a 64Kb segment.

In detail, set the address setting as follows:

Each switch is a dial displaying the hex values 0-F. Set the two switches to the first two digits of the address (in hexadecimal). For example, if you were using the default address 0D0000 (i.e. below 1 megabyte), you would set the first switch to 0 and the second switch to D. If you were using the address F00000 (i.e. in the 16th Megabyte), you would set them to F and 0 respectively. If you are unsure of the memory address, use the default address 0D0000.

Make a note of the address you have used: you will need to enter it into SMIT later.

LK1 consists of three pins: the '0' pin, the middle pin and the '8' pin (these annotations are marked on the card). Ensure the jumper is fitted on the 0 and middle pins.

If installation fails due to a clashing memory address, you can change the switch settings and re-run the installation program.

Installation of an ISA card Type B

You have two ways of setting the base address of an ISA Host Card Type B; 10-way DIL switch, or a pair of rotary hex switches and a single header.
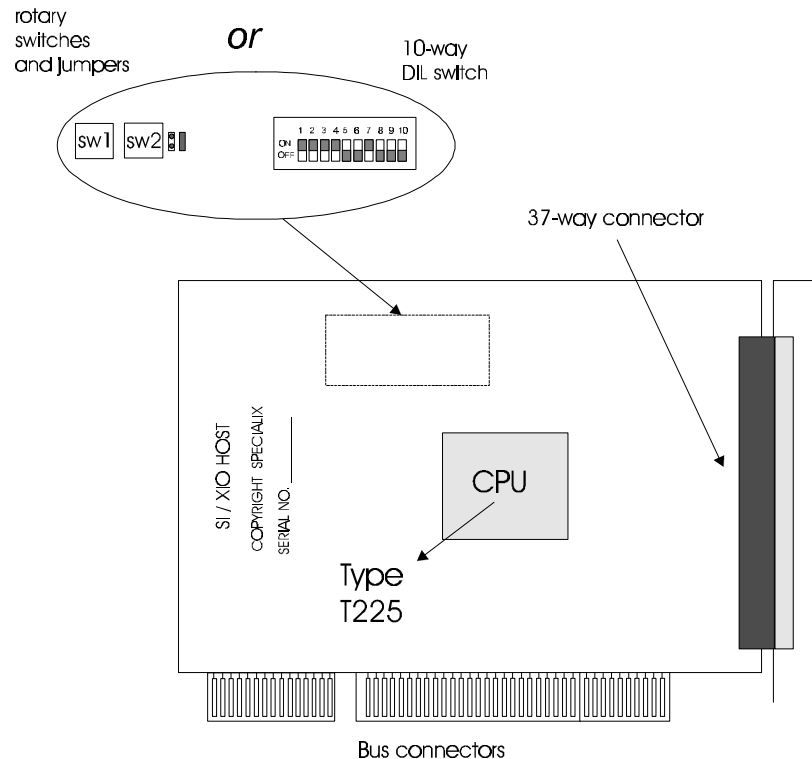
10-way DIL switch

The 10-way DIL switch is marked on the card as either Switch 3 or Switch 4.The switch has ten actuators, which can be in either the ON or OFF positions (see diagrams over the page). ON = 0 OFF = 1 (in binary). The memory address is expressed in hexadecimal e.g. 0x0D0000 but entered into the card in a binary form, i.e.OFF or ON (for 1s and 0s respectively).

The functions of the actuators on Switch 3 (4) are as follows:

Actuator Number(s)	Function
1-4	set the value of the most significant nibble of the address of the card in memory. (e.g ON, ON, ON, ON (equates to binary 0000) for the nibble 0 in 0D)
5-8	sets the value of the second most significant nibble of the address of the card in memory. (e.g OFF, OFF, ON, OFF (equates to binary 1101) for the nibble D in 0D)
9	addresses the card on the 0K boundary (OFF position, equating to the value of 0 in the memory address) or 32K boundary (ON position, equating to the value of 8 in the memory address). In the memory address see the number which follows the letter, e.g. 8 in

0x0E8000; the value can only be either 0 or 8
10   leave in the OFF position (factory default setting)

For example, to set address 0x0D0000 for a card addressed on the 0K boundary, set the actuators numbered 1 thro' 9 as shown below.

SW3 (SW4)

memory address: 0x0D0000

leave actuator 10 in the OFF position

card is addressed on the 0K boundary
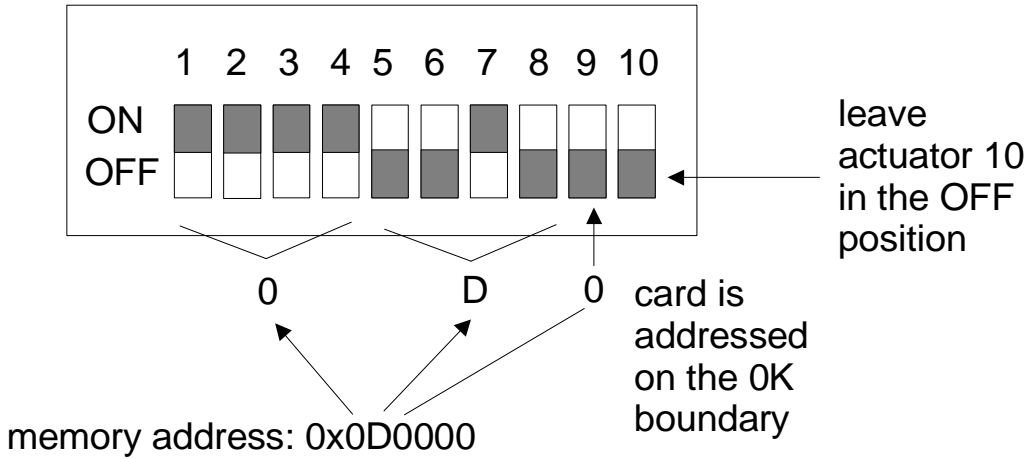
To set address 0x0E8000 for a card addressed on the 32K boundary, set the actuators numbered 1 thro' 9 as shown below.

SW3 (SW4)

memory address: 0x0E8000

leave actuator 10 in the OFF position

card is addressed on the 32K boundary
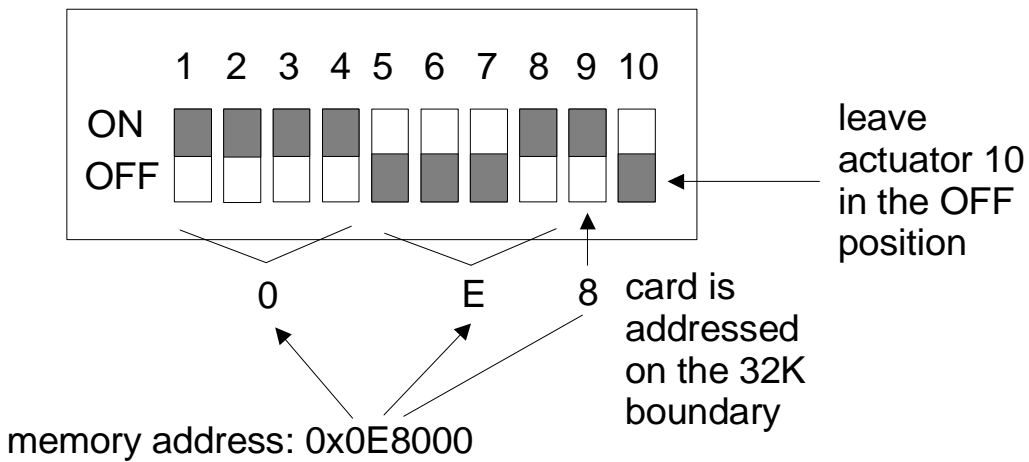
Hex Rotary Switches and Header

If your ISA card Type B is fitted with a pair of hexadecimal rotary switches and header, the function of these components is as follows:

Component	Function
SW1	sets the value of the most significant nibble (e.g. 0 in 0D) of the address of the card in memory.
SW2	sets the value of the second most significant nibble (e.g. D in 0D) of the address of the card in memory.
J2	addresses the card on the 0K boundary (header not fitted, value of 0 in the memory address) or 32K boundary (header fitted, value of 8 in the memory address). In the memory address see the number which follows the letter, e.g. 8 in 0x0E8000 and can only be either 0 or 8
J3	leave with header not fitted (factory default setting)

For example, to set address 0x0C0000 for a card addressed on the 0K boundary, adjust the rotary switches as shown below.

Leave the jumper J2 with header not fitted.
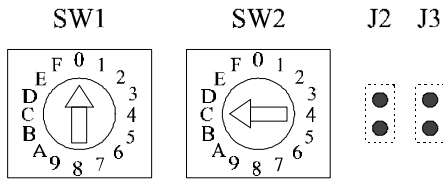
Address: 0x0C0000   Boundary : 0K

For example, to set the address for the same card - but addressed on the 32K boundary - , adjust the rotary switches and place a header on jumper J2 as shown below. The address would become 0x0C8000

The jumper required is 0.1" (inch) pitch (not supplied).
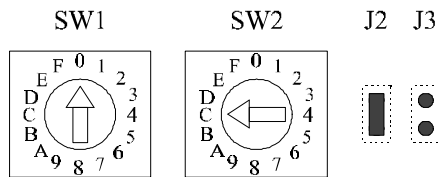
Address : 0x0C8000   Boundary : 32K

Speed Mapping Ioctls

The faster speeds supported by the SLXOS driver are not recognised by UNIX applications. To enable you to use these speeds, they are mapped to slower, infrequently used speeds which are recognised by UNIX. For example, 57.6 Kb and 115.2 Kb are mapped to 50 baud and 110 baud respectively. However, some applications (like cu) do not recognise 50 or 110 baud either, so the fast SLXOS speed(s) must be mapped to a system speed(s) which is recognised by the application. This can be done with the three speed mapping ioctls provided with this release of the SLXOS driver.

When writing a program, follow the instructions below. The following declarations must be made at the start of the program:

```
#include <sys/types.h>

#include <termio.h>

.

.

#define GETBAUD     (TIOC + 130)

#define SETBAUD     (TIOC + 131)

#define RESETBAUD   (TIOC + 132)

struct bdmap {

     dev_t bd_dev;

     int   bd_map[16];

};

struct bdmap bd;

.

.
```

The structure definition above is used to pass baud rate information into and out of the driver. The bd_dev field should be loaded with the device number of the port whose baud rates are to be mapped.

To get the current baud mapping on a port, add the following:

```
.

.

.

if ((fd=open("/dev/slx", O_RDWR))<0) {

    perror("Could not open /dev/slx");

    exit(1);

}

bd.bd_dev = dev_no;

if (ioctl(fd, SLX_GETBAUD, &bd)<0) {

    perror("SLX_GETBAUD failed");

    exit(1);

}

.

.

.
```

To perform the speed mapping (50 and 9600 baud in this case), the following code should be added:

```
.

.

.

int            baud_tmp = 0;


bd.bd_dev = dev_no;


if (ioctl(fd, SLX_GETBAUD, &bd)<0) {

    perror("SLX_GETBAUD failed");

    exit(1);

}


baud_tmp = bd.bd_map[B50];

bd.bd_map[B50] = bd.bd_map[B9600];

bd.bd_map[B9600] = baud_tmp;
```

```
if (ioctl(fd, SLX_SETBAUD, &bd)<0) {

    perror("SLX_SETBAUD failed");

    exit(1);

}

.

.

.
```

To reset the baud rate definitions to their original settings, add the following:

```
.

.

bd.bd_dev = dev_no;


if (ioctl(fd, SLX_RESETBAUD, &bd)<0) {

    perror("Could not reset baud rates");

    exit(1);

}

.

.

.
```

IOCTL Support

This release of the driver supports 3 ioctls: TCSIMGET, TCSIMBIC and TCSIMBIS. These allow applications to get, clear and set, respectively, the values of the modem control bits for the SI/XIO. The ioctl definitions are as follows (N.B. These values are not defined in standard header files):

```
TIOC ('T'<<8)

TCSIMBIS                (TIOC+113)

TCSIMBIC                (TIOC+114)

TCSIMGET                (TIOC+115)
```

To get the value of the modem bits the following call should be made:

```
unsigned short mbits = 0;
.
.
.
ioctl(fd, TCSIMGET, &mbits);
```

44-000347 Rev 5   Page 19 of 26

(fd represents the file descriptor of a tty device.) The bit values returned from the TCSIMGET call are as follows:

```
TCSIM_DCD               0001
TCSIM_RI                0002
TCSIM_DSR               0004
TCSIM_CTS               0010
TCSIM_DTR               0020
TCSIM_RTS               0040
```

The bit values TCSIM_RTS and TCSIM_DTR control the SLXOS lines CTS and DSR respectively. To set/clear the values of the modem bits the following operations should be performed (N.B. The only values that can be effectively set/cleared are the CTS and DSR lines):

```
unsigned short mbits = 0;
.
.
.
/*
 * The first operation causes the RTS line to assert
 * (assuming it was previously deasserted).
 */
mbits |= TCSIM_RTS;
ioctl(fd, TCSIMBIS, &mbits);
/*
 * The next operation causes the DTR line to assert
 * (assuming it was previously deasserted).
 */
mbits |= TCSIM_DTR;
ioctl(fd, TCSIMBIS, &mbits);
```

To set both TCSIM_RTS and TCSIM_DTR the values can be bitwise OR-ed together. Clearing modem bits is carried out in the same manner by bitwise OR-ing TCSIM_RTS and TCSIM_DTR in different combinations and using the TCSIMBIC ioctl to deassert the CTS and DSR lines. For example:

```
unsigned short mbits = 0;
.
.
.
/*
 * To deassert RTS (assuming it was previously asserted).
 */
mbits |= TCSIM_RTS;
ioctl(fd, TCSIMBIS, &mbits);
/*
 * Similarly for DTR (assuming that it was previously
 * asserted).
 */
mbits |= TCSIM_DTR;
ioctl(fd, TCSIMBIS, &mbits);
```

Xprint Ioctl

When data is transmitted to the transparent print device, the data stream is divided into packets for transmission to the port at discrete time intervals. Under normal circumstances, data consists of printable ASCII characters or short escape sequences that require no special processing during output. However, in some cases an application may transmit an escape sequence that exceeds the maximum packet size allowed by the driver. For example, an application may need to download a bitmap to a printer. Should this occur, the driver will transmit the first portion of the escape string but transmission of the remaining bytes will be suspended until the next scheduled write to the device. This may in turn cause the receiving device to timeout prematurely causing data corruption.

To prevent this an ioctl has been implemented which enables the user to instruct the driver to search the data stream for escape characters. If an escape character is found after a pre-determined number of bytes have been written to the port, the transmission of the remaining bytes will be suspended until the next time interval. By doing this, long escape sequences can be transmitted intact.

The maximum number of characters that can be transmitted at any one time is determined as follows:

```
if ((cps/5) < 223)
        Then transmit (cps/5) characters
else
        Transmit 223 characters
```

where cps is the number of characters to be transmitted per second. The tunable escape search value is set to 5 by default. In other words, the last five bytes of any packet will be searched for escape characters. To send particularly long escape strings, cps should be set to a large value, as should the tunable search parameter. To increase the value of the tunable search parameter, use the following ioctl call:

```
#include <sys/types.h>

#include <termio.h>


#define XPSEARCH        (TIOC + 129)


int     tune_Xprint;


if ((fd=open("/dev/slx", O_RDWR))<0) {

        perror("Could not open /dev/slx");

        exit(1);

}


/*

** NB This doesnt have to be 100 it can be anything a user

** wants to set it to within reason

*/

tune_Xprint = 100;


if (ioctl(fd, XPSEARCH, &tune_Xprint)<0) {

                perror("XPSEARCH ioctl failed");

        exit(1);

}
```

Technical Support

If you bought your product from a registered Specialix Supplier, you must contact their Technical Support department; they are qualified to deal with your problem.

If you are a registered Specialix Supplier, and bought your product from Specialix, contact Specialix Technical Support at the offices listed below.

In addition to describing your problem, please have ready or send, as a minimum, your name, company, country, phone and fax numbers and the name of the Specialix product.

For information on repairs, see the section 'Repairs' at the end of this release note.

Europe

United Kingdom

Specialix Europe Ltd.
3 Wintersells Road
Byfleet Surrey KT14 7LF
UK

Telephone: +44 (0) 1932 792555 (Support)
+44 (0) 1932 792592(Main SwitchBoard)

Facsimile: +44 (0) 1932 792593

e-mail: support@specialix.co.uk

ftp site: ftp@specialix.co.uk. Use your e-mail address for your password.

web site: www@specialix.co.uk

if you have an internet connection please send details of your problem to Technical Support using the form shown below. You will find the form on the Specialix web site, in the 'Tech Support' area.

France

Specialix Ltd.
13 Avenue Morane - Saulnier
78140 Velizy
FRANCE

Telephone: +33 (1) 34 63 0670

Facsimile: +33 (1) 34 65 38 60

e-mail, ftp and www contact details, as for UK.

USA

Specialix Inc.
745 Camden Avenue #129
Campbell
CA95008-4146
USA

Telephone: +1 (408) 378-7919

(Freephone): +1 (800) 423-5364 (USA only)

Facsimile: +1 (408) 378-0786

US support is open from 7am Pacific (10am Eastern) time until 5pmPacific(8Eastern) time.

E-mail: support@specialix.com

World Wide Web: www@specialix.com

If you have an internet connection please send details of your problem to Technical Support using the form shown previously. You will find the form on the Specialix web site, in the 'Tech Support' area.

Bulletin Board Service (BBS). The US BBS can be accessed using the following numbers:

+1-408-378-4766 Hayes 2400 Baud
+1-408-378-4767 MultiTech 28800 Baud
+1-408-378-4768 MultiTech 28800 Baud
+1-408-378-4769 US Robotics 14400 Baud

To log into the BBS, the user id is guest and the password support. The BBS can also be accessed from our FTP site slxbbs.specialix.com. Use your e-mail address for your password.

Asia

Singapore
Specialix (Asia) Pte Ltd.
Yu Li Industrial Building
No 37 Lor 23 Sims Avenue #04-01/05
Singapore 388371

Telephone: +65 (0) 749 1700

Facsimile: +65 (0) 749 2003

Email: support@specialix.co.uk

Australia

Specialix Pty Ltd.
Unit 15a, 390 Eastern Valley Way
Roseville
NSW 2069
AUSTRALIA

Telephone: +61 (0) 2 417 3666

Facsimile: +61 (0) 2 417 3791

Repairs

Before sending an item for repair, you must contact your Specialix supplier. If, however, you bought your product directly from Specialix, contact Specialix at the numbers listed below.

In Europe,   phone/fax: +44 1932 350990
             or e-mail: RMA@specialix.co.uk

In USA, Asia and Australia, contact the office shown in the Technical Support section.