SENSORAY CO., INC.

Model 526

Software Reference
(Version 1.0)

June, 2002

7313 SW Tech Center Dr.
Tigard, OR 97223
Phone 503.684.8073 • Fax 503.684.8164
sales@sensoray.com
www.sensoray.com

SENSORAY

Introduction

Model 526 is controlled through a set of registers mapped into I/O space of a PC-compatible system. Interrupts can be handled, if necessary, by setting up an interrupt vector which points to the interrupt processing function (handler). Those tasks are relatively easy in the environments that allow I/O operations from the application level (for example, DOS).

Some operating systems (for example, Windows) are more restrictive in terms of hardware access. Under most versions of Windows even a simple I/O access (port read or write) is not allowed directly from the application software and has to be performed by a driver. For this reason Sensoray provides a Windows SDK (Software Development Kit) for Model 526, which includes a driver component, and a DLL which contains necessary board control functions. Sample applications are provided to illustrate the use of the SDK.

Some useful functions are provided in the form of C source code. Those may be used with minor or no changes under various operating systems and development platforms.

Windows SDK

The following components are included in the Windows SDK:

1. Driver component (sxdrv98.sys for use with Windows98/2000/XP, sxdrvnt.sys for use with WindowsNT). The setup program installs the necessary file into \Windows (or \Winnt) \System32\Drivers folder.
2. Dynamic link library (DLL) (s526.dll). The setup program installs the DLL into \Windows (or \Winnt) \System32 folder.
3. Sample applications. Installed into \Program Files\Sensoray\S526\Samples folder. The sample applications are written in Microsoft C, and may require minor modifications if another compiler is used.
4. Source code library. Some useful functions are provided in the form of a C source file (s526lib.c).

Data Types

```
typedef struct {
  int     port;                  //I/O address
  HANDLE  hEvent;                //handle to interrupt event
} BOARD526;
```

The BOARD526 type structure is used for board initialization. The application sets the port member to the value of the base address of the board before a call to S526_Open function. If the function returns successfully, the hEvent member is set to the value of the interrupt event's handle (see Functions section and sample applications for the details).

Functions

S526_Open

```
int S526_Open (
 BOARD526 *pBoard
);
```

PARAMETERS

pBoard

pointer to the variable of BOARD526 type.

Returns

The function returns 0 in case of success, or an error code.

Notes

The application sets the .port member of the BOARD526 structure to the value of the base address of the board before the call to S526_Open. Upon successful return, the .hEvent member contains the handle of the interrupt event. This function has to be called for every Model 526 board present in the system (even if interrupts are not used).

S526_Close

```
void S526_Close (
 BOARD526 *pBoard
);
```

Parameters

pBoard

pointer to the variable of BOARD526 type.

Notes

This function releases the driver resources allocated by S526_Open. It is necessary to call this function for every Model 526 board that was opened. The pointer pBoard has to point to the same variable that was used by S526_Open.

S526_WriteReg

```
void S526_WriteReg (
 unsigned short base,
 unsigned short reg,
 unsigned short data
);
```

Parameters

base

base address of the board;

reg

register address;

data

data to write.

Notes

All write accesses are 2 byte, even address.

S526_ReadReg

```
unsigned short S526_ReadReg (
 unsigned short base,
 unsigned short reg
);
```

Parameters

base

base address of the board;

reg

register address;

Returns

The function returns a value read from the addressed register.

Notes

All read accesses are 2 byte, even address.

Developing a Windows application

Sensoray does not provide a .lib file for the DLL, because thouse are not compatible across various development platforms. Instead, dynamic linking is used. An application has to explicitly open *s526.dll* and get the pointers to DLL fuctions before those functions could be called. To simplify those actions 2 files are provided: *s526f.h* and *s526app.c*. The following steps have to be completed when developing an application with *s526.dll*:

1. File *s526app.c* has to be included in the project. Please, do not try the following: `#include “s526app.c”,` it’s not going to work! Instead use “Add files to project”.
2. Each file of the project that contains calls to DLL functions has to include *s526f.h* (this time with the help of `#include` directive).
3. The first action of the application related to Model 526 board has to be a call to S526_DLLOpen function (declared in *s526f.h*). That is followed by a call (or calls) to S526_Open for each Model 526 board used by the application. When terminating the application first call S526_Close (for all open boards), and then S526_DLLClose (once per application). Errors returned by S526_DLLOpen function are usually related to either DLL not being installed in one of Windows default folders (or an application folder), or a wrong version of the DLL being installed.

Please see sample applications for programming examples.

Interrupts under Windows

The following approach is implemented in the DLL: when a board is initialized (with S526_Open) the driver creates an event that becomes signaled when an interrupt occurs. A handle to this event is passed to the application as part of the BOARD526 structure. The application uses Windows API function WaitForSingleObject to wait until interrupt occurs. If a call to this function is put in a separate thread, other actions could be performed by the application while this thread is waiting for the interrupt (WaitForSingleObject does not consume CPU time). Note that the Interrupt Enable Register (IER) is being reset to 0 by the driver as soon as the interrupt occurs. The application has to restore the value of IER (enable interrupts) after the interrupt processing is complete.

Installation

The following procedure has to be carried out for installation under Windows98/2000/XP. The system response during the installation may vary slightly depending on the specific version of Windows. Under WindowsNT only step 1 is necessary.

1. Run setup.exe from the installation disk. The software is installed under \Program Files\Sensoray\S526.
2. Find the necessary spare resources (I/O space and interrupt). This can be done by going to Device Manager, selecting “View resources by type”, and expanding “Input/output (IO)” and “Interrupt request (IRQ)” items. Configure the jumpers on the board to match the selected I/O base address and IRQ.

3. Open the file s526.inf (it is installed under Program Files\Sensoray\S526\Driver) with a text editor (e.g. Notepad). Locate the line that starts with "IOConfig". If the board's address range is not listed in this line (e.g. 200-23f means that the board's base address is 0x200), append it to the end of the line similar to existing entries. For example, if the board's base address is set to 0x460, the modified line in the s526.inf should look like the following:

   ```
   IOConfig=200-23f(ffff::),280-2bf(ffff::),2c0-2ff(ffff::),460-49f(ffff::)
   ```

   Save s526.inf.

4. Open Control Panel, doubleclick on "Add/remove hardware" icon. Select "Add/troubleshoot a device". Select "Add a new device". Select "No, I want to select the hardware from a list". Select "Sound, video and game controllers". Click on "Have disk". Click on "Browse" and locate s526.inf. When "Sensoray Model 526 I/O Board" is displayed under "Models", click "Next". If "Windows could not detect the settings…" warning is displayed, click "OK". Highlight "Input/Output Range", and click on "Change setting". Select the board's address range from a list-box. Make sure "Conflict information" window shows no conflict, otherwise you may need to select a different I/O range for the board and repeat the procedure starting with step 2. Click "OK". Highlight "Interrupt Request" and click "Change setting". Select the board's IRQ from a list-box. Click "OK". If "Digital signature not found" message is displayed, click "Yes". Click "Finish". Reboot on a prompt.

It is not recommended to remove Model 526 from the device list in Hardware manager (it may cause your Windows to crash). Please contact Sensoray if you absolutely have to do that.