

SENSORAY CO., INC.

---

# 516/616 MPEG Frame Grabbers

## Programmer's Manual

March 10, 2003

© Sensoray 2001  
7313 SW Tech Center Dr.  
Tigard, OR 97223  
Phone 503.684.8073 • Fax 503.684.8164  
[sales@sensoray.com](mailto:sales@sensoray.com)  
[www.sensoray.com](http://www.sensoray.com)



<b>1. Introduction .....</b>	<b>3</b>
<b>2. Software Components .....</b>	<b>4</b>
<b>2.1. Installation Procedure .....</b>	<b>5</b>
<b>3. Building an Application with SX16 Library .....</b>	<b>6</b>
<b>4. Functions Reference .....</b>	<b>6</b>
<b>4.1. Windows Specific Functions .....</b>	<b>6</b>
X16_DLLOpen .....	6
X16_DLLClose .....	6
<b>4.2. Library Functions .....</b>	<b>7</b>
X16_open.....	7
X16_register .....	7
X16_unregister.....	8
X16_set_input .....	8
X16_set_input_type .....	8
X16_set_input_system .....	9
X16_set_output_system .....	9
X16_set_bitrate .....	9
X16_set_picsize .....	10
X16_set_m.....	10
X16_set_n.....	11
X16_set_vbr.....	11
X16_read_gpio.....	11
X16_write_gpio .....	12
X16_read .....	12
X16_write .....	12
X16_stop.....	13
X16_pause .....	13
X16_resume .....	13
X16_set_brightness .....	14
X16_set_contrast.....	14
X16_set_saturation.....	15
X16_set_colorbars .....	15
X16_set_volume .....	15
X16_set_mute .....	16
X16_prepare_capture .....	16
X16_get_capture_status .....	16
X16_capture_start .....	17
X16_close_capture.....	17
X16_close .....	18

# 1. Introduction

The Sensoray Models 516 and 616 is an MPEG video encoder decoder board. Some of the features include:

## General

- Real time MPEG-2 and MPEG-1 video encoder and decoder
- Support for variable bit rate and constant bit rate
- IPB pictures to 15Mbps for constant bit rate and 10Mbps for variable bit rate
- Supports multiple resolutions (704x480, 640x480, 352x240, etc.)
- Support for NTSC, PAL
- During encoding and standby, video input is fed to output for easy adjustment
- Onboard audio CODEC

## Video encoder

- Generates 13818 (MPEG-2) and 11172 (MPEG-1) compliant elementary streams (ES)
- Operates up to 30 frames per second
- Selectable bit rate

## Video decoder

- Decodes both MPEG-1 and MPEG-2 streams
- Horizontal and vertical scaling

## Video capture (616 board only)

- Supports small (up to 1/2x1/2 of original size) uncompressed image capture.

## 2. Software Components

The software contains the following files:

### Windows

Filename	Function
sm2288.h	Header: hardware-depended constants
sm2288f.h	Header: function prototypes
sm2288.dll	Library functions (DLL)
sxdrv98.sys	The driver
sx16.inf	Windows INF to register the driver
x16demo.c	The demo application
x16app.c	Some common components to connect to the DLL
x16demo.h	Header file for the demo
x16demo.rc	Resource file for the demo
*.ico, *.bmp	Pictures for the demo

**Table 1. Software Components (Windows)**

### Windows CE

Filename	Function
sm2288.h	Header: hardware-depended constants
sm2288f.h	Header: function prototypes
x16.dll	The driver
x16demo.c	The demo application
x16app.c	Some common components to connect to the driver
x16demo.h	Header file for the demo
x16demo.rc	Resource file for the demo
*.ico, *.bmp	Pictures for the demo

**Table 2. Software Components (Windows CE)**

## Linux

Filename	Function
sm2288.h	Header: hardware-depended constants
sm2288f.h	Header: function prototypes
sx16lib.o	Library functions
x16dat1.hex	Hardware data files
x16dat2.hex	
x16dat3.hex	
x16dat4.hex	
x16dat5.hex	
sxdrv.o	The driver
mknode	Script to register the driver
x16demo.c	The demo application
makefile	The rule to build the demo

**Table 2. Software Components (Linux)**

## **2.1. Installation Procedure**

### Windows

Copy the driver file to your Windows System32\Drivers directory. Register the driver with the “Device Manager” utility (Add New Hardware).

To register a 516 board you must select the IO base address in this dialog:

- Select “No, I want to select the hardware from a list”.
- Select the type of hardware: “Other devices”
- Click “Have Disk”
- Browse the location of sx16.inf file
- Select “Sensoray Model 516 Frame Grabber”
- You will see the list of resources (Input/Output Range), acceptable for the device. Select one and set the same base address on the board (see 516 board manual).

### Windows CE

Copy the driver file to your Windows System32\Drivers directory.

### Linux

Run *mknode* to register the driver. Type: *insmod sxdrv.o* to run the driver.

## 3. Building an Application with SX16 Library

### Windows, Windows CE

Compile and link your application together with x16app.c, including sm2288.h and sm2288f.h.

### Linux

Link your application together with sx16lib.o, including sm2288.h and sm2288f.h.

Please refer to the sample source code for an example of building an application with SX16 Library.

## 4. Functions Reference

The SX16 Library is designed to provide the application developer with full control over the frame grabber. All special data types used by the Library are defined in sm2288.h. The sample application illustrates the use of most of the functions and allows building a custom application within minutes.

### ***4.1. Windows Specific Functions***

#### **X16\_DLLOpen**

```
int X16_DLLOpen(void);
```

##### **Parameters**

None

##### **Return values**

Returns 0 on success. Returns an error code if any errors were detected.

##### **Notes**

Opens the DLL.

#### **X16\_DLLClose**

```
void X16_DLLClose(void);
```

##### **Parameters**

None

#### **Return values**

None

#### **Notes**

Closes the DLL.

## ***4.2. Library Functions***

### **X16\_open**

```
int X16_open(void);
```

#### **Parameters**

None.

#### **Return value**

The function returns number of board in case of successful initialization (positive), or an error code (negative).

#### **Notes**

The function initializes the driver and searches for all boards supported by the Library. Boards enumerated by the OS in order of used PCI slots. The board number 0 located in the slot with the least number on the PCI bus with the least number. You can run some PCI utility such as “scanpci” to check a board presence. The board IDs are: 0x1131 (vendor) and 0x7146 (device). The driver released by a call to X16\_close.

### **X16\_register**

```
int X16_register(int CardId);
```

#### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

#### **Return values**

Returns 0 on success. Returns an error code if any errors were detected.

#### **Notes**

Init and registers the board, allocates system resources.

## **X16\_unregister**

```
void X16_unregister(int CardId);
```

### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

### **Return values**

None.

### **Notes**

Unregisters the board and frees system resources.

## **X16\_set\_input**

```
void X16_set_input(int CardId,int input);
```

### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

input

Video input. Must be between 0 and 3.

### **Return values**

None.

### **Notes**

Selects video input. Default is 0.

*Sensoray 616 board (Revision A) has only one input.*

## **X16\_set\_input\_type**

```
void X16_set_input_type(int CardId,int type);
```

### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

type

Video input type. Must be 0 for Composite Video or 1 for S-Video.

### **Return values**

None.

### **Notes**

Sets video input type. Default is S-Video.



## **X16\_set\_input\_system**

```
void X16_set_input_system(int CardId,int system);
```

### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

system

Video system. Must be PAL (1) or NTSC (2).

### **Return values**

None.

### **Notes**

Sets input video system NTSC or PAL. Default is 2 (NTSC).

## **X16\_set\_output\_system**

```
void X16_set_output_system(int CardId,int system);
```

### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

system

Video system. Must be PAL (1) or NTSC (2).

### **Return values**

None.

### **Notes**

Sets output video system NTSC or PAL. Default is 2 (NTSC).

## **X16\_set\_bitrate**

```
void X16_set_bitrate(int CardId,int rate);
```

### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

rate

Video bit rate in bits per second. Must be between 700,000 and 15,000,000.

### Return values

None.

### Notes

Sets desired bit rate.

## X16\_set\_picsize

```
void X16_set_picsize(int CardId,int size);
```

### Parameters

CardId

Card identifier. Must be between 0 and number of cards minus 1.

size

Picture size index.

### Return values

None.

### Notes

Sets picture size and compression system:

- **SIZE176\_1** - MPEG-1, NTSC-176x112 PAL-176x144
- **SIZE352\_1** - MPEG-1, NTSC-352x240 PAL-352x288
- **SIZE352\_2** - MPEG-2, NTSC-352x480 PAL-352x576
- **SIZE480\_2** - MPEG-2, NTSC-480x480 PAL-480x576
- **SIZE640\_2** - MPEG-2, NTSC-640x480 PAL-640x576
- **SIZE704\_2** - MPEG-2, NTSC-704x480 PAL-704x576
- **SIZE720\_2** - MPEG-2, NTSC-720x480 PAL-720x576

## X16\_set\_m

```
void X16_set_m(int CardId,int m);
```

### Parameters

CardId

Card identifier. Must be between 0 and number of cards minus 1.

m

Number of frames in Group of Pictures.

### Return values

None.

### Notes

Sets number of frames in Group of Pictures. Default is 3.

## **X16\_set\_n**

```
void X16_set_n(int CardId,int n);
```

### **Parameters**

CardId  
Card identifier. Must be between 0 and number of cards minus 1.

n  
Distance between I/P frames.

### **Return values**

None.

### **Notes**

Sets distance between I/P frames. Default is 15.

## **X16\_set\_vbr**

```
void X16_set_vbr(int CardId,int vbr);
```

### **Parameters**

CardId  
Card identifier. Must be between 0 and number of cards minus 1.

vbr  
Enables (1) or disables (0) variable bit rate compression

### **Return values**

None.

### **Notes**

Enables/disables variable bit rate compression. Default is 0.

## **X16\_read\_gpio**

```
int X16_read_gpio(int CardId);
```

### **Parameters**

CardId  
Card identifier. Must be between 0 and number of cards minus 1.

### **Return values**

State of GPIO

### **Notes**

Reads state of GPIO pins: active input is low (when input is pulled down bit is set to 1).

## **X16\_write\_gpio**

```
void X16_write_gpio(int CardId,int state);
```

### **Parameters**

CardId  
Card identifier. Must be between 0 and number of cards minus 1.  
state  
State of GPIO

### **Return values**

None.

### **Notes**

Sets state of GPIO pins: active output is low (output is pulled down when bit is set to 1).

## **X16\_read**

```
int X16_read(int CardId,char *buffer,int nbytes);
```

### **Parameters**

CardId  
Card identifier. Must be between 0 and number of cards minus 1.  
buffer  
Pointer to input video buffer.  
nbytes  
Number of bytes to input or -1 if any errors were detected.

### **Return values**

Number of bytes actually read.

### **Notes**

Attempts to read "nbytes" bytes of MPEG data from the SM2288 output FIFO. The actual amount read is returned. If no data is available "0" is returned. If this is the first read the encoder is initialized and started.

## **X16\_write**

```
int X16_write(int CardId,char *buffer, int nbytes);
```

### **Parameters**

CardId  
Card identifier. Must be between 0 and number of cards minus 1.  
buffer  
Pointer to output video buffer.  
nbytes

Number of bytes to output or -1 if any errors were detected.

#### **Return values**

None.

#### **Notes**

Attempts to write "nbytes" bytes of MPEG data to the SM2288 input FIFO. The actual amount written is returned. If no room in the board buffer is available "0" is returned. If this is the first write the decoder is initialized and started.

### **X16\_stop**

```
int X16_stop(int CardId);
```

#### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

#### **Return values**

Returns 0 on success. Returns an error code if any errors were detected.

#### **Notes**

Stops read/write. Sends "STOP" command to SM2288.

### **X16\_pause**

```
int X16_pause(int CardId);
```

#### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

#### **Return values**

Returns 0 on success. Returns an error code if any errors were detected.

#### **Notes**

Pauses read/write. Sends "PAUSE" command to SM2288.

### **X16\_resume**

```
int X16_resume(int CardId);
```

#### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

#### **Return values**

Returns 0 on success. Returns an error code if any errors were detected.

#### **Notes**

Resume read/write after pause. Sends "RESUME" command to SM2288.

### **X16\_set\_brightness**

```
void X16_set_brightness(int CardId,int brightness);
```

#### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

brightness

Brightness. Must be between 0 and 255.

#### **Return values**

None.

#### **Notes**

Sets brightness on video input.

### **X16\_set\_contrast**

```
void X16_set_contrast(int CardId,int contrast);
```

#### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

contrast

Contrast. Must be between 0 and 255.

#### **Return values**

None.

#### **Notes**

Sets contrast on video input.

## **X16\_set\_saturation**

```
void X16_set_saturation(int CardId,int saturation);
```

### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

saturation

Saturation. Must be between 0 and 255.

### **Return values**

None.

### **Notes**

Sets color saturation on video input.

## **X16\_set\_colorbars**

```
void X16_set_colorbars (int CardId,int bars);
```

### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

bars

Saturation. Must be 0 or 1.

### **Return values**

None.

### **Notes**

Video output internal color bar generator on/off.

## **X16\_set\_volume**

```
void X16_set_volume(int CardId,int volume);
```

### **Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

volume

Volume. Must be between 0 and 63.

### **Return values**

None.

## Notes

Sets audio volume in dB (0 - maximum, 63 – minimum).

## X16\_set\_mute

```
void X16_set_mute(int CardId,int mute);
```

### Parameters

CardId

Card identifier. Must be between 0 and number of cards minus 1.

mute

Mute. Must be 0 or 1.

### Return values

None.

## Notes

Sets audio mute (0 – sound is on, 1 – sound is off).

## X16\_prepare\_capture

```
int X16_prepare_capture (int CardId, BUFFER *buffer);
```

### Parameters

CardId

Card identifier. Must be between 0 and number of cards minus 1.

buffer

Pointer to a BUFFER structure. Fields:

unsigned char *capbuf;	//	pointer to capture (uncompressed) buffer
int capsize;	//	size of buffer in bytes
int width,height;	//	picture size in pixels

### Return values

Returns 0 on success. Returns an error code if any errors were detected.

## Notes

Prepares a buffer to capture uncompressed frames. User must specify desirable *width* and *height*. Function calculates compatible *width* and *height* (maybe different) and allocates *capbuf* with *capsize* (in bytes).

## X16\_get\_capture\_status

```
void X16_get_capture_status(int CardId);
```



**Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

**Return values**

Capture status is active (1) during data transferring to the buffer.

**Notes**

Status set to 1 by X16\_capture\_start and cleared by hardware.

**X16\_capture\_start**

```
int X16_capture_start (int CardId);
```

**Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

**Return values**

Returns 0 on success. Returns an error code if any errors were detected.

**Notes**

Starts capture.

**X16\_close\_capture**

```
void X16_close_capture (int CardId);
```

**Parameters**

CardId

Card identifier. Must be between 0 and number of cards minus 1.

**Return values**

Returns 0 on success. Returns an error code if any errors were detected.

**Notes**

Closes and clears caption buffer.

## **X16\_close**

```
void X16_close(void);
```

### **Parameters**

None.

### **Return values**

None.

### **Notes**

Closes the driver.