

SENSORAY CO., INC.

---

**Crosspoint Video Switch /  
Frame Grabber  
Model 609 (Rev.A)**

June 18, 2001

© Sensoray 2001  
7313 SW Tech Center Dr.  
Tigard, OR 97223  
Phone 503.684.8073 • Fax 503.684.8164  
sales@sensoray.com  
www.sensoray.com



---

# Table of Contents

LIMITED WARRANTY .....	3
SPECIAL HANDLING INSTRUCTIONS .....	4
INTRODUCTION .....	5
Feature Summary .....	5
General Description.....	5
REFERENCE.....	7
Connectors.....	7
Input video connector, J1. ....	7
Output video connectors, J2-J9 .....	7
General purpose inputs 0-7 connector, J12. ....	8
General purpose inputs 8-15 connector, J13. ....	8
General purpose outputs 0-7 connector, J10. ....	8
General purpose outputs 8-15 connector, J11. ....	8
Specifications .....	9
SOFTWARE REFERENCE.....	10
Installation .....	10
Building an application with s609.dll.....	10
DLL exported functions .....	11
S609_InitSystem .....	11
S609_GetHFG .....	11
S609_AllocBuffer .....	12
S609_FreeBuffer .....	12
S609_CloseSystem .....	12
S609_Acquire.....	13
S609_StartAcquire.....	13
S609_StopAcquire .....	13
S609_GetStatus .....	14
S609_GetStatusEx.....	14
S609_SetStatusEx .....	14
S609_SetMode.....	15
S609_GetImageSize .....	15
S609_Switch .....	15
S609_IOWrite .....	16
S609_IORead.....	16
S609_GetSwitchData.....	16

# Limited warranty

Sensoray Company, Incorporated (Sensoray) warrants the hardware to be free from defects in material and workmanship and perform to applicable published Sensoray specifications for two years from the date of shipment to purchaser. Sensoray will, at its option, repair or replace equipment that proves to be defective during the warranty period. This warranty includes parts and labor.

The warranty provided herein does not cover equipment subjected to abuse, misuse, accident, alteration, neglect, or unauthorized repair or installation. Sensoray shall have the right of final determination as to the existence and cause of defect.

As for items repaired or replaced under warranty, the warranty shall continue in effect for the remainder of the original warranty period, or for ninety days following date of shipment by Sensoray of the repaired or replaced part, whichever period is longer.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. Sensoray will pay the shipping costs of returning to the owner parts that are covered by warranty. A restocking charge of 25% of the product purchase price, or \$105, whichever is less, will be charged for returning a product to stock.

Sensoray believes that the information in this manual is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, Sensoray reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult Sensoray if errors are suspected. In no event shall Sensoray be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, SENSORAY MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF SENSORAY SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. SENSORAY WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEROF.

Third party brands, names and trademarks are the property of their respective owners.

# Special handling instructions

The circuit board contains CMOS circuitry that is sensitive to Electrostatic Discharge (ESD).

Special care should be taken in handling, transporting, and installing circuit board to prevent ESD damage to the board. In particular:

- Do not remove the circuit board from its protective anti-static bag until you are ready to install the board into the enclosure.
- Handle the circuit board only at grounded, ESD protected stations.
- Remove power from the equipment before installing or removing the circuit board.

# Introduction

Model 609 combines the functions of a 16x8 crosspoint video switch and a frame grabber. It is based on the Conexant PCI video decoder FUSION878A. The software for the 609 is similar to the SX11 software that supports a line of Sensoray image capture boards.

## Feature Summary

- PCI form factor (32 bit, 33 MHz PCI bus)
- 16 video inputs (flat cable connector), 75 Ohm termination
- 8 video outputs with programmable serial 75 Ohm termination allowing parallel connection of multiple boards
- multichannel frame grabber: allows image capture from 4 out of 8 selected output video channels
- supported video standards: NTSC-M, NTSC-Japan, PAL-B, PAL-D, PAL-G, PAL-H, PAL-I, PAL-M, PAL-N, SECAM, CCIR, RS-170
- PCI bus mastering: low CPU involvement during image capture
- 16-bit input and 16-bit output general purpose I/O ports
- Drivers and a DLL for Windows98/NT/2000

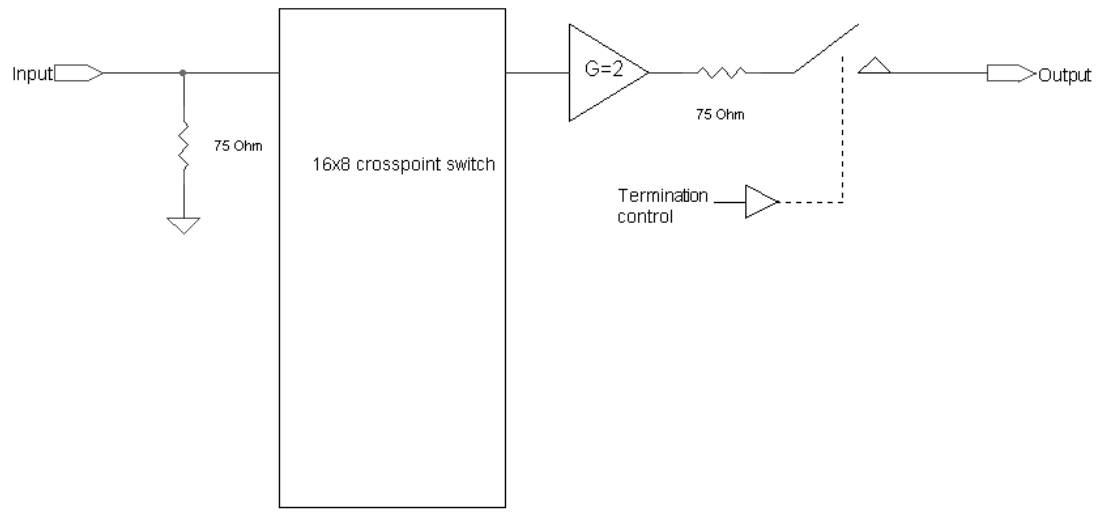
## General Description

Model 609 has 16 video inputs (1 through 16), which accept standard composite video signal. Each input is terminated with a 75 Ohm resistor.

The video outputs (0 through 7) have a series 75 Ohm termination resistor. With a 75 Ohm termination at the receiving end the video switch provides a gain of 1. The programmable switch in the output signal path allows connecting multiple outputs together without disturbing the termination conditions. The termination switch is closed by default after power up.

The outputs 0 through 3 of the crosspoint switch are also connected to 4 multiplexed inputs of the frame grabber. The signal components is digitized with an 8-bit A/D converter. Low-pass filtering and double over-sampling of the input signal provide precise digitization with no aliasing artifacts. The digital signal is then scaled and/or cropped to the desired dimensions, if necessary. The scaled image is transferred to the host RAM using the PCI bus mastering mode, which requires minimum CPU attention. An on-board FIFO provides necessary buffering, minimizing the probability of image loss.

The following simplified diagram shows the path of the video signal.



# Reference

## Connectors

Input video connector, J1.

Pin	Signal	Pin	Signal
1	Ground	2	Input 1
3	Ground	4	Input 2
5	Ground	6	Input 3
7	Ground	8	Input 4
9	Ground	10	Input 5
11	Ground	12	Input 6
13	Ground	14	Input 7
15	Ground	16	Input 8
17	Ground	18	Input 9
19	Ground	20	Input 10
21	Ground	22	Input 11
23	Ground	24	Input 12
25	Ground	26	Input 13
27	Ground	28	Input 14
29	Ground	30	Input 15
31	Ground	32	Input 16
33	-	34	-

Output video connectors, J2-J9

Pin	Signal	Pin	Signal
1	Ground	2	Output (0-7)

Notes:

1. The output video connectors are Molex 22-05-3021. The mating connector is Molex 22-01-3027, crimp terminals 08-50-0114.
2. Output 0 is J2, output 1 is J3, etc.

General purpose inputs 0-7 connector, J12.

Pin	Signal	Pin	Signal
1	+5 V	2	Input 0
3	Input 1	4	Input 2
5	Input 3	6	Input 4
7	Input 5	8	Input 6
9	Input 7	10	Ground

General purpose inputs 8-15 connector, J13.

Pin	Signal	Pin	Signal
1	+5 V	2	Input 8
3	Input 9	4	Input 10
5	Input 11	6	Input 12
7	Input 13	8	Input 15
9	Input 15	10	Ground

Note: General purpose inputs are standard 5V TTL.

General purpose outputs 0-7 connector, J10.

Pin	Signal	Pin	Signal
1	+5 V	2	Output 0
3	Output 1	4	Output 2
5	Output 3	6	Output 4
7	Output 5	8	Output 6
9	Output 7	10	Ground

General purpose outputs 8-15 connector, J11.

Pin	Signal	Pin	Signal
1	+5 V	2	Output 8
3	Output 9	4	Output 10
5	Output 11	6	Output 12
7	Output 13	8	Output 14
9	Output 15	10	Ground

Note: General purpose outputs are 5V TTL outputs. Maximum output current is -32 mA in HIGH state, 64 mA in LOW state.



## Specifications

Video sources	NTSC, PAL, SECAM, RS-170, CCIR
Video inputs	16, analog composite video
Video outputs	8, analog composite video, 75 Ohm termination on the receiving end required for a gain of 1
Video switch bandwidth, min	6 MHz
Frame grabber output formats	RGB (15, 16, 24, 32 bits/pixel), Y8 (8 bits/pixel), YCrCb (16 bits/pixel)
Output resolution (max), pixels	754x480 (NTSC, RS-170), 922x576 (PAL, SECAM, CCIR)
A/D resolution: luminance channel chrominance channel	8 bit 8 bit
Capture rate	Real time <sup>[1]</sup> : 30 fps (NTSC, RS-170), 25 fps (PAL, SECAM, CCIR)
General purpose I/O port	16 inputs (5V TTL) 16 outputs (5V TTL, I <sub>HIGH</sub> =-32 mA, I <sub>LOW</sub> =64 mA)
Operating temperature	0°C to 70°C
Power	+5 V, 300 mA (max) -12 V, 50 mA (max)

# Software Reference

## Installation

Model 609 is shipped with the software that supports Windows98/NT/2000 platforms. The software includes a driver (windrvr.sys) and a dynamic-link library (s609.dll). The software is installed by running setup.exe from the installation disk.

The following procedures have to be followed to properly install the driver:

- Windows98 : When "Found new hardware..." message appears, point to s609.inf file on the installation disk. Repeat twice, as model 609 is detected by Windows as a multifunctional device. Run setup.exe from the installation disk.
- WindowsNT: Run setup.exe from the installation disk.
- Windows2000: When "Found new hardware..." message appears, select "Search for a suitable driver..." and clear all boxes for search locations. After the message "Windows was unable to locate a driver", chose "Disable the device". Repeat twice, as model 609 is detected by Windows as a multifunctional device. Run setup.exe from the installation disk.

The following components are installed on the target system:

- \Windows\System\s609.dll;
- \Windows\System32\Drivers\Windrvr.sys;
- A copy of the driver and 2 utility programs in \Program Files\Sensoray\S609 SDK\Driver;
- The following 4 files in \Program Files\Sensoray\S609 SDK\Include: s609.h, s609.ico, s609app.c, s609f.h;
- Sample application source files and an executable in \Program Files\Sensoray\S609 SDK\Sample1.

## Building an application with s609.dll

The following files are distributed with s609.dll:

- s609.h – contains data types and constants definitions;
- s609f.h – contains exported functions prototypes;
- s609app.c – contains exported functions and helper functions definitions.

When building an application with s609.dll, it is necessary to include s609app.c in the project. All files containing calls to the s609.dll functions have to also include s609f.h.

## DLL exported functions

All custom types referred to below are defined in s609.h.

### S609\_InitSystem

```
ECODE S609_InitSystem (  
    PCI *pPci          //pointer to a PCI structure  
);
```

#### Return values

Returns 0 in case of success, or an error code (a list of error codes is included in s609.h). A value WNG\_INITIALIZED is returned if the system has already been initialized by another process or thread.

#### Notes

The S609\_InitSystem is called once when the application starts. It initializes all 609 boards found in the system and writes their slot numbers into PCI structure.

### S609\_GetHFG

```
ECODE S609_GetHFG (  
    HFG *pHfg,          //pointer to a handle  
    DWORD slot          //board's PCI slot number  
);
```

#### Return values

Returns 0 in case of success, or an error code.

#### Notes

Gets a handle to the board detected by S609\_InitSystem. From this moment on, the board is addressed by a handle value.

### S609\_AllocBuffer

```
ECODE S609_AllocBuffer (
    MODE *pMode,           //pointer to MODE structure
    BUFFER *pBuffer,      //pointer to BUFFER structure
    DWORD param           //meaning depends on the type of buffer
);
```

#### Return values

Returns 0 in case of success, or an error code.

#### Notes

The MODE structure has to be set according to the desired frame grabber mode before a call to S609\_AllocBuffer is made. A variable (structure) of BUFFER type has to be declared. If S609\_AllocBuffer returns successfully, the buffer handle member of the BUFFER structure is set to the valid buffer handle value. From this moment on this buffer is addressed by its handle.

The *param* variable is reserved for future extensions, and has to be set to 1.

### S609\_FreeBuffer

```
void S609_FreeBuffer (
    HBUF hbuf              //a buffer handle
);
```

#### Return values

None.

#### Notes

Frees the buffer allocated by S609\_AllocBuffer.

### S609\_CloseSystem

```
void S609_CloseSystem (
);
```

#### Return values

None.

#### Notes

This function releases all the resources allocated by S609\_InitSystem. It has to be called only once, when the application terminates.

## S609\_Acquire

```
ECODE S609_Acquire (  
    HFG hfg,                //board handle  
    HBUF hbuf,              //buffer handle  
    float timeout,          //acquisition timeout, seconds  
    DWORD status            //acquisition status  
);
```

### Return values

Returns 0 in case of success, or an error code.

### Notes

Captures one frame from the board *hfg* into the buffer *hbuf*. Possible acquisition errors are reported in *status*. See *s609.h* for the meaning of status bits.

## S609\_StartAcquire

```
ECODE S609_StartAcquire (  
    HFG hfg,                //board handle  
    HBUF hbuf,              //buffer handle  
    DWORD acqmode          //acquisition mode  
);
```

### Return values

Returns 0 in case of success, or an error code.

### Notes

In case of a single frame capture mode (*acqmode*=*AMODE\_SINGLE*), starts capture of one frame from the board *hfg* into the buffer *hbuf*, and returns. In case of continuous capture (*acqmode*=*AMODE\_CONT*), starts continuous capture into the same buffer. In the latter case the acquisition has to be stopped by calling *S609\_StopAcquire*. The application checks the capture status to determine when the acquisition is complete.

## S609\_StopAcquire

```
ECODE S609_StopAcquire (  
    HFG hfg                //board handle  
);
```

### Return values

Returns 0 in case of success, or an error code.

### Notes

Stops acquisition by the board *hfg*. The acquisition is stopped asynchronously (possibly in the middle of the frame).

#### S609\_GetStatus

```
ECODE S609_GetStatus (
    HFG hfg,                //board handle
    DWORD *pStatus         //pointer to a status variable (DWORD)
);
```

#### Return values

Returns 0 in case of success, or an error code.

#### Notes

Provides status information on acquisition process. See *s609.h* for the meanings of status bits.

#### S609\_GetStatusEx

```
ECODE S609_GetStatusEx (
    HFG hfg,                //board handle
    STATUS *pStatus        //pointer to a status variable (STATUS)
);
```

#### Return values

Returns 0 in case of success, or an error code.

#### Notes

Provides extended status information on acquisition process. See *s609.h* for the meanings of STATUS structure members.

#### S609\_SetStatusEx

```
ECODE S609_SetStatusEx (
    HFG hfg,                //board handle
    STATUS *pStatus,       //pointer to a status variable (STATUS)
    STATUS *pMask         //pointer to a status variable (STATUS)
);
```

#### Return values

Returns 0 in case of success, or an error code.

## Notes

Resets the internal status registers based on the data provided in the first variable of type STATUS and the mask provided in the second variable of type STATUS.

## S609\_SetMode

```
ECODE S609_SetMode (  
    HFG hfg,                //board handle  
    MODE *pMode             //pointer to a MODE structure  
);
```

## Return values

Returns 0 in case of success, or an error code.

## Notes

Sets the operation mode of the frame grabber based on the MODE structure. See s609.h for details.

## S609\_GetImageSize

```
ECODE S609_GetImageSize (  
    MODE *pMode,            //pointer to a MODE structure  
    IMAGESIZE *pImsize     //pointer to an IMAGESIZE structure  
);
```

## Return values

Returns 0 in case of success, or an error code.

## Notes

Sets the members of IMAGESIZE structure to the values corresponding to selected operation mode (MODE).

## S609\_Switch

```
ECODE S609_Switch (  
    HFG hfg,                //board handle  
    WORD output,           //video switch output (0-7)  
    WORD input,            //video switch input (0-16)  
    BOOL term              //termination  
);
```

## Return values

Returns 0 in case of success, or an error code.

### Notes

Controls the video switch: connects input *input* to output *output*, and switches the termination switch (closed, if *term*=TRUE). Active input channels are 1 through 16, selecting 0 for an input disables selected output.

### S609\_IOWrite

```
ECODE S609_IOWrite (  
    HFG hfg,                //board handle  
    WORD data                //output data  
);
```

### Return values

Returns 0 in case of success, or an error code.

### Notes

Writes the 16-bit value *data* to the output I/O bus.

### S609\_IORead

```
ECODE S609_IOWrite (  
    HFG hfg,                //board handle  
    WORD *pData            //variable accepting input data  
);
```

### Return values

Returns 0 in case of success, or an error code.

### Notes

Sets up the variable pointed to by *pData* to the 16-bit value read from the input I/O bus.

### S609\_GetSwitchData

```
ECODE S609_GetSwitchData (  
    HFG hfg,                //board handle  
    VSMODE *pVsmode        //array accepting switch data  
);
```

### Return values

Returns 0 in case of success, or an error code.



## Notes

Reads the full state of the video switch into an 8 element array of VSMODE type:

```
typedef struct {                //video switch data
    WORD input;                 //selected input
    BOOL termination;          //on (TRUE) or off (FALSE)
} VSMODE;
```

Each element of the array receives the data for the corresponding output of the switch. Suggested usage:

```
VSMODE vsmode[8];
if (ecode = S609_GetSwitchData (hfg, vsmode)) {
    return ecode;
} else {
    // get the switch info from vsmode array
}
```