



Great River Technology

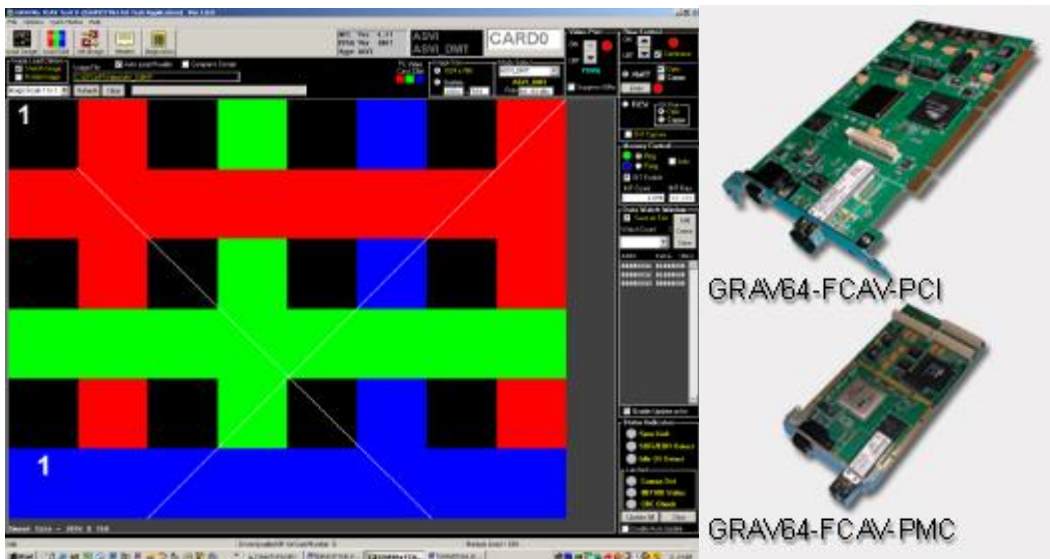
6121 Indian School Rd. NE Ste 141
Albuquerque, NM 87110
www.greatrivertech.com

Gravity64 PCI FCAV Video Card User's Manual

Instruction for using GRAV64_PCI_FCAV_XXX cards

Rev F

July 28, 2006



Contact Information

Telephone	1 (866) 478-4491
Company Fax	(505) 883-1375
Email	grt@greatrivertech.com
Web Address	www.greatrivertech.com

TABLE OF CONTENTS

1	Scope	5
1.1	Products Covered-Gravity 64-bit FCAV	5
1.2	Customer Specific Firmware	5
1.3	Reference Documents.....	5
2	Overview of Gravity FC-AV	6
2.1	Product Description	6
2.2	Fibre Channel Compatibility	6
2.2.1	Compatibility with FC-0, FC-1, FC-2	6
2.2.1.1	Physical.....	6
2.2.1.2	Signaling/Encoding.....	7
2.2.1.3	Signaling Speed.....	7
2.2.1.4	Fibre Channel Frames and Ordered sets	7
2.2.2	Compatibility with the FC-AV Standard.....	7
2.2.3	Compatibility with other FC equipment	7
2.2.4	Non-standard implementations	7
3	Gravity FC-AV Hardware Operation	8
3.1	Recommended System Configuration	11
3.2	Memory Operation	11
3.3	PCI Interface.....	12
3.4	VGA Monitor interface.....	12
3.5	Fiber Interface.....	12
3.5.1	Transmit Operation.....	12
3.5.2	Receive Operation.....	13
3.5.3	Viewing live video on VGA	14
3.5.4	FC-AV Loop though.....	14
3.6	Fibre Channel DVI Cards.....	14
3.7	Supported Video Formats.....	15
4	Using GRAVityFCAVTest with Gravity FC-AV cards	17
4.1	Software Installation.....	17
4.2	System Memory Allocation	18
4.3	Selection of FC-AV transmit and receive modes.....	18
4.3.1	Transmitting using <i>Quick Modes</i>	19
4.3.2	Transmitting using manual controls	20
4.3.3	Transmitting from Ping/Pong image stores	20
4.3.4	Receiving using <i>Quick Modes</i>	21
4.3.5	Receiving using manual controls.....	21
4.3.6	Receiving video and header data to the GUI	22
4.3.6.1	Image update rate in the GUI	22
4.3.6.2	Checking pixel values in the GUI.....	23
4.3.7	Receiving video to the VGA Port.....	23
4.3.8	Image update rate on the VGA Port.....	23

4.3.9	Receiver Status	24
4.4	Transmit with DVI	26
4.4.1	Verifying DVI input	27
4.4.2	Transferring DVI input to FCAV output	27
4.4.3	DVI frame rate	28
4.4.4	Using GRAVityFCAVTest in XGA resolution	28
4.5	Ping and Pong Memory Map	29
4.6	Loading and retrieving images	30
4.7	Loading and retrieving headers	32
4.8	Diagnostics Menu	34
4.8.1	Initialization and Reset	35
4.8.2	HOSTGRAV.INI File settings	35
4.9	Data Watch Window	37
4.10	Compress Screen Mode	37
5	Fibre Channel Frame Format and Ordered Sets	38
5.1	Idle Words	39
5.2	Start of Frame	39
5.3	Data Payloads	39
5.4	Cyclic Redundancy Check (CRC)	39
5.5	End of Frame	39
6	Fibre Channel Containers	40
6.1	Object 0 FC frame	40
6.2	Object 0 FC-AV Container Header	40
6.3	FC-AV Object 0 Ancillary Data	41
6.4	FC-AV Object 2 Video Data	41
7	Header Data and Formats	42
7.1	Header Utility	42
7.2	Header Data Files	43
7.3	Header Formats	44
7.4	Example Header Format Definition (HFD) file	46
8	Standard Firmware - Supported Format Profiles	47
8.1	XGA Color Progressive	47
8.2	SXGA Color Progressive	48

REVISION HISTORY:

DATE	REVISION	SECTIONS	DESCRIPTION	APPROVAL
1/30/2006	REV -	INITIAL RELEASE	INITIAL RELEASE	J. ALEXANDER
2/1/2006	REV A	4.6,6.4	DESCRIPTION OF BYTE ORDERING OF OBJ0 HEADER FILES CORRECTED	J. ALEXANDER
2/17/2006	REV B	VARIOUS CLEANUP	VARIOUS CLEANUP	J. ALEXANDER
2/18/2006	REV C	3.5,4.3.X	DESCRIPTION OF GUI ABD VGA UPDATE RATES, HEADER MENU	J. ALEXANDER
4/6/2006	REV D	4.3, 4.5, 4.9	ADD SECTION FOR SOFTWARE "COMPRESS SCREEN", UPDATE FOR SOFTWARE INT RATE, LAYOUT CHANGES	J. ALEXANDER
4/19/2006	REV E	4.4.1 – 4.4.1	ADDED SECTIONS TO DESCRIBE DVI OPERATION WITH GRAVITYTEST SOFTWARE	J. ALEXANDER
7/27/06	REV F	4.7, 7.3	ADDED DETAIL ABOUT HOW THE HEADER FORMAT FILE NAME IS STORED IN THE INI FILE.	D. COLE

1 Scope

This document describes the following:

- Compatibilities with FC-AV Standards
- Gravity 64-bit FCAV Hardware Operation
- Using Gravity 64-bit FCAV cards with GRAVityFCAVTest Application Software
- Using Gravity 64-bit FCAV cards for transmit and receive of FC-AV video

This document provides a high level description of the hardware, but does not include register definitions and details for controlling the card. For users developing custom software, please consult the **EFC API Users Guide**. For customers using custom firmware, please consult the appropriate addendum for that firmware revision.

1.1 Products Covered-Gravity 64-bit FCAV

This Hardware User's Manual describes hardware functionality of, and low-level interface to, the Great River's FC-AV products.

Unless otherwise indicated, this document describes the functionality of Great River's standard firmware load for Gravity FC-AV. This firmware provides capabilities for several standard image sizes and pixel packing methods.

1.2 Customer Specific Firmware

It is intended that the Gravity FCAV card support customer specific FC-AV implementations. These implementations may cover less common image sizes, pixel packing methods, header definitions, and/or compression methods. For customers using special firmware loads, Great River will assign a unique, customer specific part number. Additional details for a customer specific FC-AV protocol will be covered in a customer specific ICD.

1.3 Reference Documents

ANSI X3.230-1994	Fibre Channel - Physical and Signaling Interface (FC-PH)
ANSI X3.297-1997	Fibre Channel – Physical and Signaling Interface – 2 (FC-PH-2)
ANSI X3.303-1998	Fibre Channel - Physical and Signaling Interface - 3 (FC-PH-3)
ANSI INCITS 356-2002	Fibre Channel – Audio Video (FC-AV), 25 November 2002

2 Overview of Gravity FC-AV

2.1 Product Description

The Gravity FC-AV products are intended to support implementations of ANSI INCITS 356-2002 Fibre Channel – Audio Video (FC-AV) for transporting digital video. Fibre Channel provides a reliable, high bandwidth video link for video networks, point-to-point video connections (up to 500 meters using multi mode fiber and up to 10km using single mode fiber).

Each Gravity FC-AV card is Fibre Channel compliant at the FC-0, FC-1, and FC-2 layers. The cards use Frame Header Control Protocol (FHCP) and described in Clause 7 of the FCAV standard. FHCP is an easy to implement, low overhead, protocol based on the FC-AV container system. FC-AV containers are described in Clause 5 of the FC-AV standard.

The cards can be ordered for 1.0625 Gbps or 2.125Gbps link speed. Customers can special order other link speeds between 1 and 3.1875 Gbps.

The cards have a total of 256Mbits data storage. This allows for 8Mbyte (2Mx32bit) for ping and 8MB (2Mx32bit) for pong image stores. With ping and pong memory storage areas, the PCI bus can access image memory during live transmit and receive. The cards have dedicated SXGA video ports so that incoming video can be viewed live without loading the PCI Bus.

Gravity64 FC-AV is a 64/66 PCI card that can achieve 160Mbytes/sec transfer rates over PCI.

These cards are compatible with Great River's DVI receiver mezzanine card. When ordered with the Mezzanine card and DVI capture firmware, the cards can be used to transfer standard DVI video onto fiber. With the mezzanine, the Gravity FC-AV cards can be directly connected to any graphics card with DVI output, to transfer real-time full motion rendered graphics onto fiber.

The following sections describe the cards compatibility with Fibre Channel Standards when using the standard firmware load.

2.2 Fibre Channel Compatibility

The Gravity FC-AV cards are Fibre Channel compatible at the FC-0, FC-1, and FC-2 as described below.

2.2.1 Compatibility with FC-0, FC-1, FC-2

2.2.1.1 Physical

Gravity FC-AV card can be configured with either optical or copper interfaces (or both).

All cards can be factory loaded with SFF Transceivers that operate at 850 nm or 1310 nm. These transceivers are compatible with standard LC multimode or single mode fiber cable.

Lower baud rate (<1.5Gbps) cards can also be factory loaded with high-speed transformer coupled electrical interfaces. The copper interfaces uses two "differential 1x3" shrouded connectors compatible with 150-Ohm FCN1010 type cable.

See the Product Specification sheet for ordering options.

2.2.1.2 Signaling/Encoding

All cards use 8B/10B Encoding (per ANSI X3.230-1994 -FC-PH) for all signaling transmission.

2.2.1.3 Signaling Speed

Standard signaling speed is 1.0625 Gbps or 2.125 Gbps. Non-standard baud rates can also be achieved through special factory loaded oscillators.

2.2.1.4 Fibre Channel Frames and Ordered sets

All cards use Fibre Channel Frames and Ordered sets as defined in ANSI X3.230-1994 (FC-PH)

2.2.2 Compatibility with the FC-AV Standard

When using Great River's standard firmware along with the GRAVityFCAVTest application software, the Gravity FC-AV cards transport digital video using Frame Header Control Protocol (FHCP) as described in the FC-AV Standard Clause 7.

More specifically, Gravity FC-AV cards use FHCP to transport FC-AV Containers (FC-AV Standard Clause 5), where a single Video frame is loaded into a single simple mode container.

2.2.3 Compatibility with other FC equipment

Gravity FC-AV cards use FC frames as the basic unit of transport for video and header data. FC frames are further discussed in the referenced FC-PH specification. As such, the digital video link will be compatible with commercial off-the-shelf Fibre Channel development tools. FC-AV is not compatible with other data oriented protocols used with Fibre Channel such as SCSI or RDMA.

2.2.4 Non-standard implementations

Gravity FCAV cards supports customer specific FC-AV implementations through special firmware loads. These implementations may cover less common image sizes, pixel packing methods, header definitions, and/or compression methods. These implementations may deviate from FC standards in order to meet the need of proprietary video interfaces.

3 Gravity FC-AV Hardware Operation

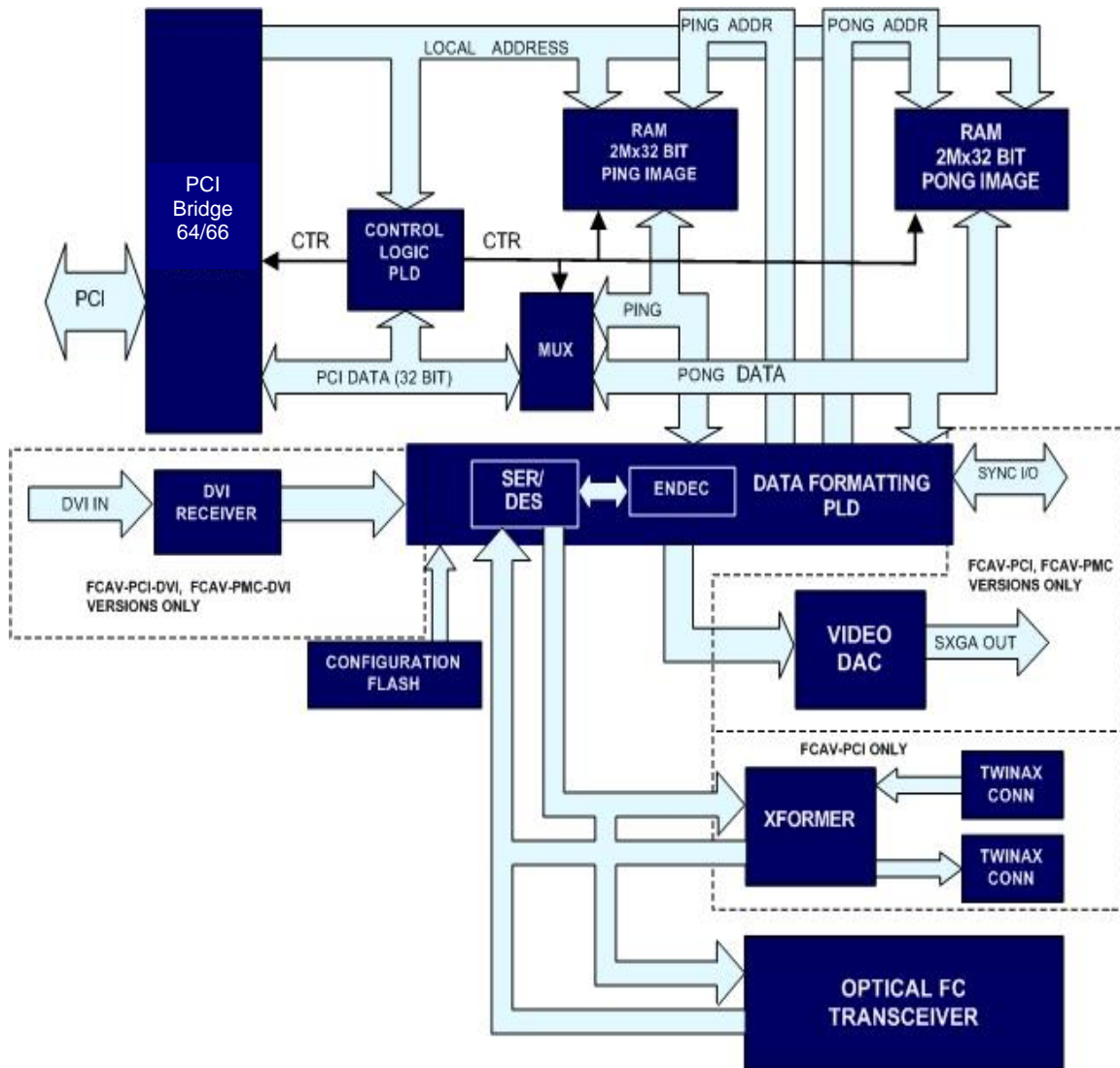


Figure 3.0.1 Hardware Block Diagram

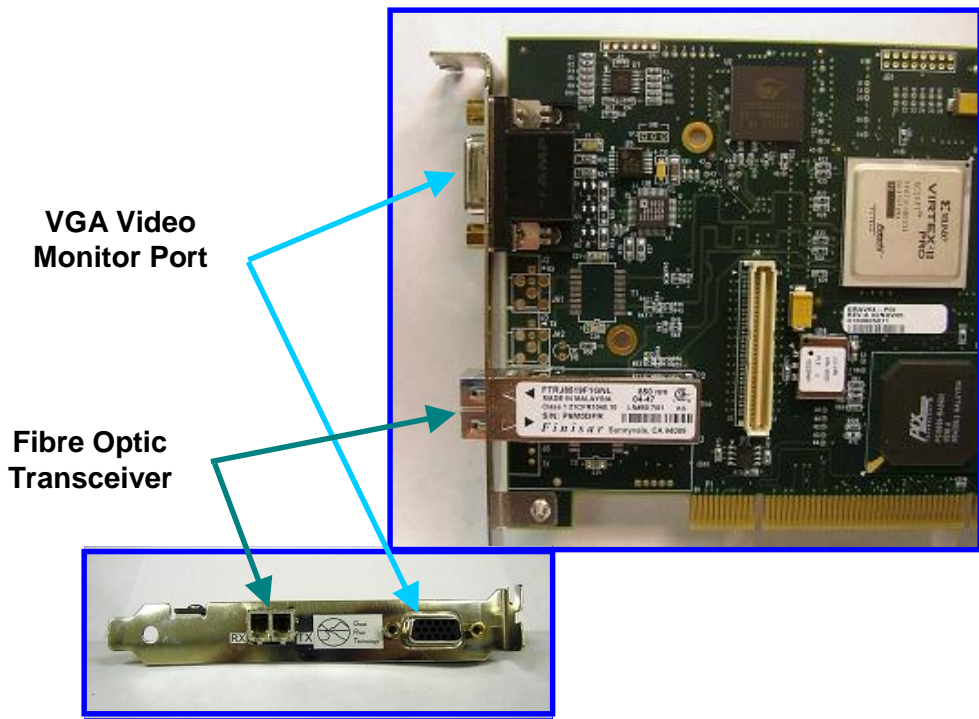


Figure 3.0.3 Circuit Card Front Panel I/O Connectors

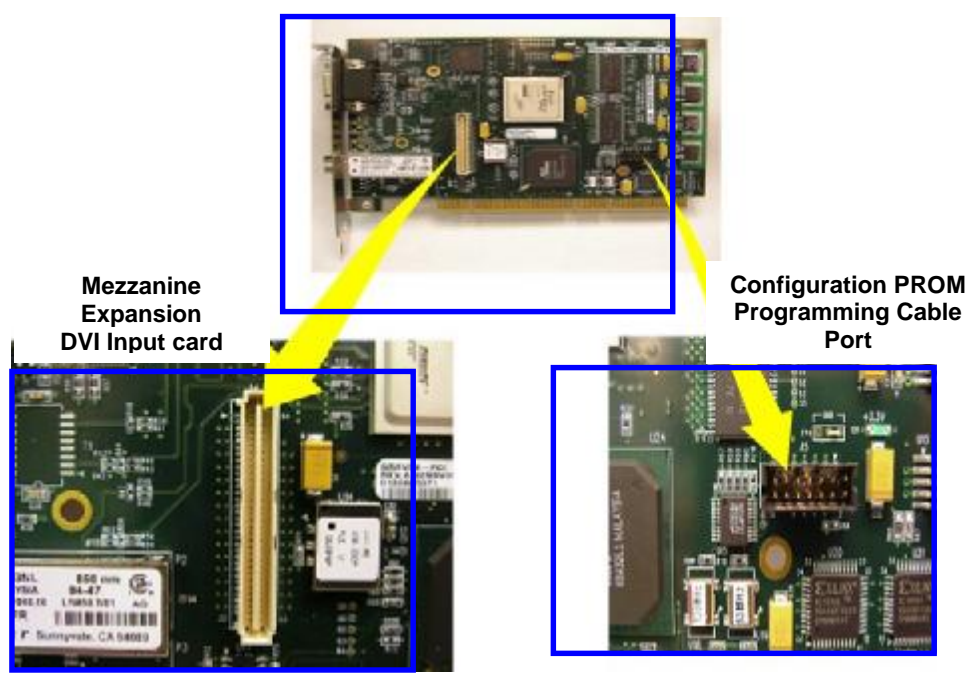


Figure 3.0.4 Circuit Card I/O Connectors

3.1 Recommended System Configuration

The FCAV hardware manages the majority of the tasks needed to receive and display FC video images on VGA display.

Minimum System Requirements:

Processor: PIII 733MHz

Memory: 128MB

BUS: 533MHz FSB w/ (1) PCI slot or (1) PCI-X slot

Video Card: 64 MB (capable of 1280x1024 resolution)

Recommended System Requirements:

Processor: P4 3.0GHz

Memory: 512MB

BUS: 800MHz FSB w/ (1) PCI slot and (1) PCI-X slot

Video Card: ATI or NVIDIA w/ 256 MB

The majority of the image processing is performed in the FCAV hardware. FCAV hardware will process video images from FC link to a VGA display without requiring significant PC system bandwidth. System performance will impact the response time of the application, and time to transfer images from hardware memory to PC memory. The maximum bandwidth for the PCI 64bit/66Mhz bus is 528MBps. The maximum bandwidth for the PCI 32bit/33Mhz bus is 132MBps. The FCAV hardware typically achieves a bandwidth of 160MBps in a PCI 64bit/66Mhz. This is sufficient to support FCAV applications from 1.0625Gbps or 2.125Gbps link speed and displaying VGA images. Operating the FCAV card in a 32-bit slot will impact image loading and retrieval, but not real time capture and display of video images from FC link to VGA display.

3.2 Memory Operation

The hardware has two 2MB by 32-bit memory spaces referred to as Ping and Pong memory. The memory is single port SDRAM. These memory spaces are used for several tasks. The memory is used to store video images received on the FC link, to write stored video images to the VGA output and move video images to and from PC memory via the PCI bus. The Ping and Pong memory spaces can be accessed by one operation at a time. Therefore, two memory spaces are required to facilitate display of live video images to VGA display operating in a Ping, Pong alternating fashion.

Typically the VGA will display currently loaded image from either Ping or Pong while a new image is loaded from FC link to the other memory space either Ping or Pong. In this manner, real time display of video images is accomplished. Proper control of the Ping and Pong memory access is critical for proper acquisition and display of FC images.

The FCAV applications allows for direct control of the Ping and Pong memory spaces. The user must take care to properly select and configure the use of the Ping and Pong memory space. Failure to do so will result in corrupted images and possible application lock up. This is only a concern when the user is trying to override the default settings of the application. In general, there are three modes in which the Ping and Pong memory space is accessed.

1. The first mode is to load a bit map to the Ping or Pong memory space. In this mode real time capture of FC images and display of stored images to the VGA display should be disabled.
2. The second mode is real time capture and display of FC images to the VGA display. Do not attempt to transfer an image from Ping or Pong memory to PC memory during real time capture and display.
3. The third mode, is retrieving a stored FC image to PC memory via the PCI bus. In this mode the real time capture of FC images and display of stored images to the VGA must be disabled.

The application allows the user to load bit maps into Ping and Pong memory for display to a VGA display and output on FC link. The bit maps must be 24-bit true color format. Each pixel in the bit map is represented by three 8-bit bytes for Red, Green, and Blue. When the bit map is loaded into Ping and Pong memory each pixel is represented by four 8-bit bytes or one 32-bit word. The byte order from MSB to LSB is Alpha, Red, Green, and Blue. Where Alpha is always a value of 0x00.

Bits 31-24	Bits 23-16	Bits 15-8	Bits 7-0
Alpha (0x00)	Red	Green	Blue

On the FC link the order of bytes transmitted is Blue, Green, Red. Alpha is not transmitted on the link. When a stored image in Ping or Pong is loaded to PC memory and saved as a bit map the Alpha byte is dropped to support the 24 bit true color bit map format.

See section 4.4 Memory Map for Ping and Pong memory allocation.

3.3 PCI Interface

The Gravity64 FC-AV cards have 64-bit PCI interfaces that run at 66 MHz which drive +3.3 volt logic levels on outputs and have +5 volt tolerant inputs.

3.4 VGA Monitor interface

The Gravity FC-AV cards have standard VGA 15 pin connectors. The connector has a fused +5V output set for 500mA. The SXGA interface has sufficient drive strength for standard computer monitors with RGB 75 Ohm terminations.

Standard computer monitors require precise frame timing. When the SXGA output is turned on, the Gravity FC-AV card is tolerant of incoming video frame rates that are a multiple of 16.67 ms (+/- 0.1%).

3.5 Fiber Interface

3.5.1 Transmit Operation

The GRAVITY FC-AV cards can be used to transmit stored video images. In this mode of operation, images are loaded via PCI and transmitted over the fiber as shown in Figures 3.5.1. Complete images can be loaded or parts of image memory can be modified while the

transmitter is active. Therefore, the cards can be used to transmit static video or video that is updated at the bandwidth limits of PCI.

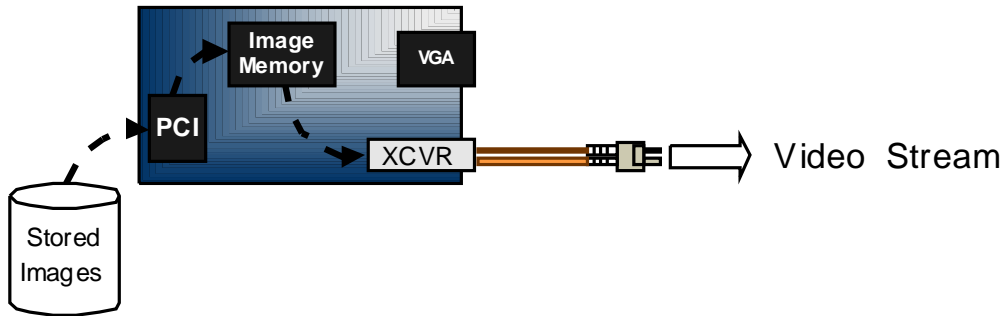


Figure 3.5.1 Stored Image Transmit

3.5.2 Receive Operation

The GRAVITY FC-AV cards can be used to capture images over PCI or display live video on the dedicated VGA output port. Full-motion video (received from fiber) can be viewed on standard computer monitors connected directly to the Gravity card. When displaying video on the VGA port there is no loading of the PCI bus. Images may also be retrieved over PCI at the bandwidth limits of PC.

Figures 3.5.2 indicates the operation of the cards in the receive mode.

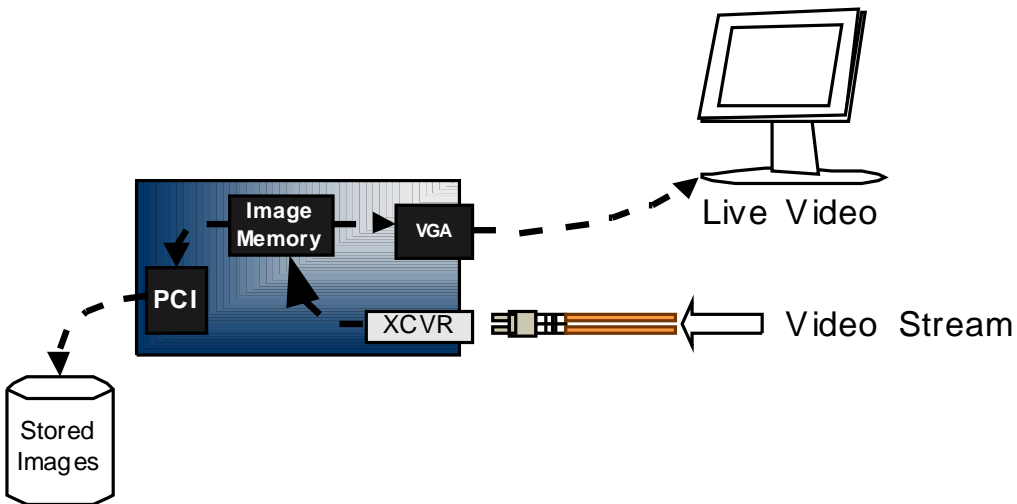


Figure 3.5.2 Image Capture and VGA Display

3.5.3 Viewing live video on VGA

There are two image buffers on the Gravity card. When one of these buffers is receiving image data from the fiber, data from the other buffer is being sent to the VGA output. The vertical and horizontal line timing being sent to the VGA is independent of that being received over the FC-AV link. Therefore, the VGA timing can be tuned independent of what is being received, and consequently, can have precise control over horizontal timing (and jitter) to meet the requirements of standard computer monitors.

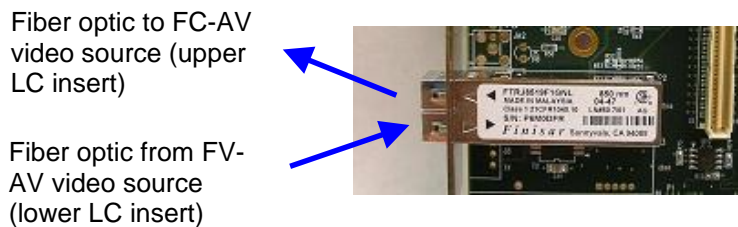
The receive circuit used does not impose any timing requirements for the incoming video on the fiber. The VGA images will always be displayed. There is no requirement that the horizontal line timing (FC frame timing within the container) be uniform on the fiber. There is also no requirement for the timing of containers – the card is tolerant of any incoming vertical rate between 1 Hz and 65 Hz.

In cases where the incoming video is not precisely 60.00Hz, the update of images on VGA will be reduced to 30Hz (every other frame). For very slow vertical rates on the fiber, the update rate on the VGA port may reduce further.

3.5.4 FC-AV Loop through

The Gravity card can be used to monitor FC-AV traffic between an FC-AV source and receiver. When receiving FC-AV video, the Gravity card automatically re-transmits the received video through the optical module transmit port.

To use the card as a loop through monitor, the standard LC connector Tx/Rx plug must be separated, with one fiber coming from the FC-AV transmitter, and the other fiber going to the receiver.



The re-transmitted output is re-clocked by the Gravity card and will have a 21-word latency from the input. The output will preserve all timing characteristics of the input and will maintain the number of ordered sets between fiber channel frames.

Loop through is only active when the card is configured as a receiver. When the card is configured as a transmitter, the content of the on-board memory determines the outgoing container payload.

3.6 Fibre Channel DVI Cards

Gravity FC-AV PCI cards are compatible with Great River's DVI receiver mezzanine card. This card can interface to any standard graphics card with DVI-D output.

With the DVI mezzanine associated DVI firmware, the Gravity FC-AV PCI card can convert DVI video streams into FC-AV video streams. The DVI video streams may be real-time rendered graphics and/or video from stored MPEG Files. In either case, the uncompressed digital video is inserted into FC frames per FC-AV recommendations.

Figures 3.4 indicates the operation of the cards in transmit mode.

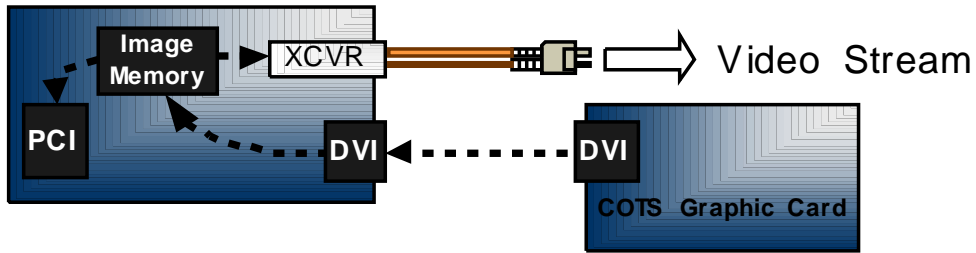


Figure 3.4 DVI play through to fiber



Figure 3.5 Gravity back plate with DVI input connector

*Note: Some Gravity cards using the DVI Mezzanine do not have VGA output ports and, therefore, cannot be used to play fiber through the VGA.

Operating the Gravity FCAV card with a DVI mezzanine is further described in section 4.4

3.7 Supported Video Formats

Specific video parameters, such as frame rate, image size, pixel information, and number of video lines per FC frame, are referred to as a video profile.

Great River's FC-AV cards, with the standard firmware load, support the following video profiles:

Format #	Pixel Format	Resolution	FMV Rate	Pixels/line	Lines/frame	Bits/pixel
1	8:8:8 Color	XGA	Variable	1024	768	24
2	8:8:8 Color	SXGA	Variable	1280	1024	24

Table 3.6 Supported Video Profiles – 32 bit PCI card with 1.0625 Gbps Interface

Contact GRT for additional profiles supported in the 64 bit PCI cards and/or cards using the 2.125Gbps interface.

The full-motion video (FMV) rate is the rate at which a complete new frame can be updated on the VGA monitor port. The VGA output refresh rate is always at 60Hz.

4 Using GRAVityFCAVTest with Gravity FC-AV cards

GRAVityFCAVTest (Figure 4.0) is provided as a useful tool to transmit and receive images in a variety of applications.

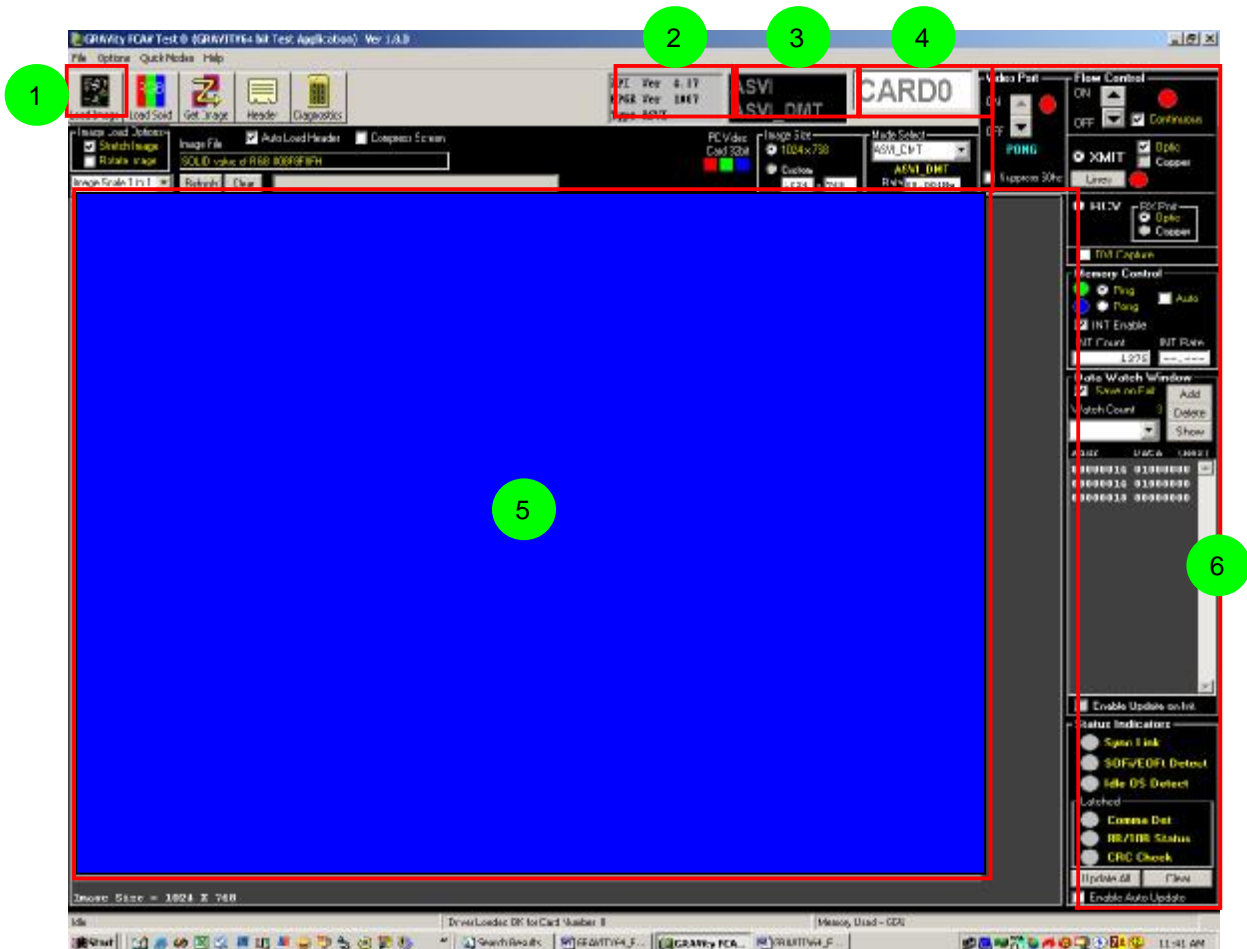


Figure 4.0 GRAVityFCAVTest GUI

1. Area for loading and getting images, header utility, and diagnostics utility.
2. Application information. Shows the Software version, FPGA version and type.
3. Shows Application type and Mode Selection.
4. Shows card number that has been enumerated for the application.
5. Area where image is displayed.
6. Control functions for Flow Transmit and Receive, Memory, Data Watch and Status.

4.1 Software Installation

To Install the GRAVityFCAVTest software, please refer to the following documents:

- EFC API install Guide n ([efc_api_installguide.pdf](#) REV B or later)

4.2 System Memory Allocation

GRAVityFCAVTest allocates large buffers in system memory to store transmit and receive data. The Application uses approx 48 Mbytes for each instance opened.

4.3 Selection of FC-AV transmit and receive modes

The GRAVityFCAVTest software is capable of transmitting and receiving multiple formats (see 3.7 supported video formats and also the FCAV API User Manual). On the right side of the main screen there is a control area in where you can configure GRAVityFCAVTest to transmit, receive and monitor status (*Figure 4.1*).

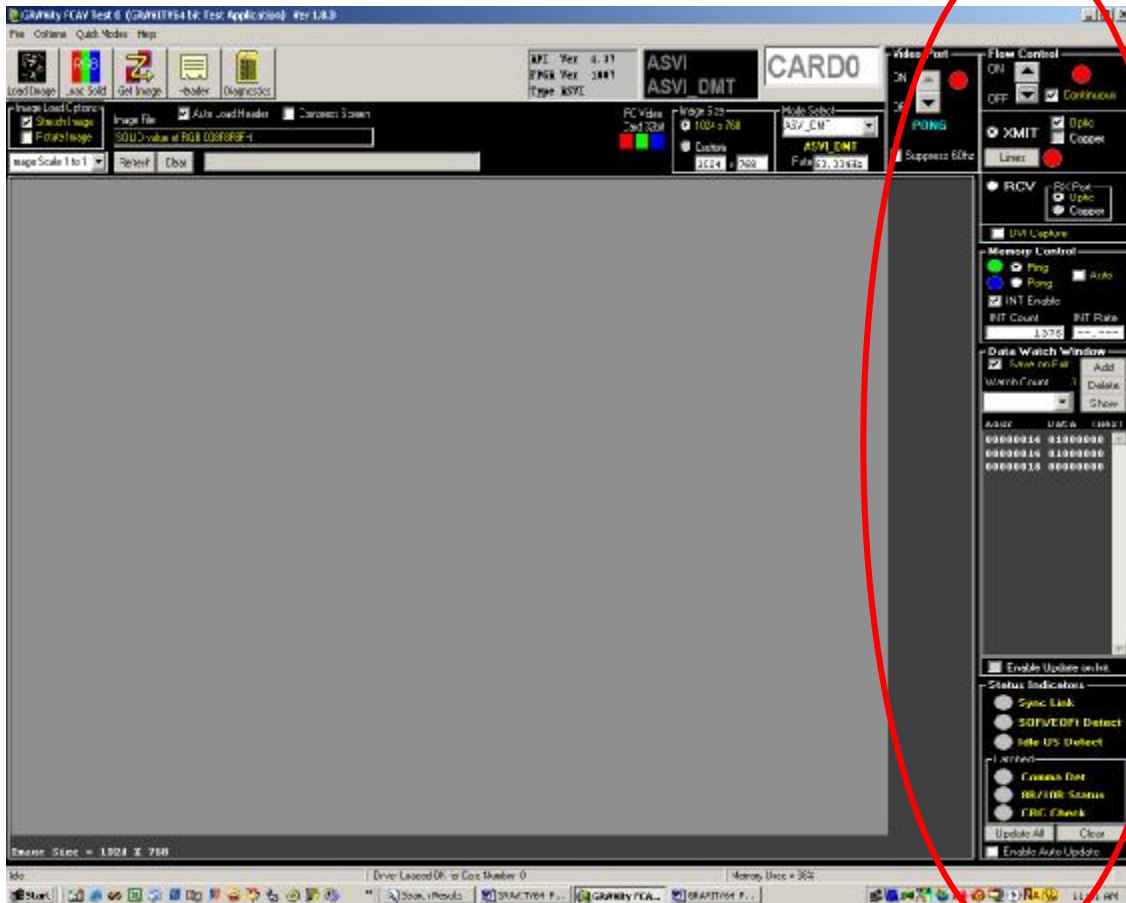
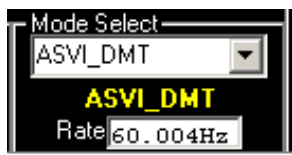


Figure 4.3 GRAVityFCAVTest Controls

4.3.1 Transmitting using *Quick Modes*

The **Mode Select** pull down, located in the control panel to the right, is used to select the FCAV profile for transmit and receive. For the software to correctly transmit the FC-AV, this setting must match the profile of the incoming FCAV. The video rate for the selected mode is displayed below the mode.



GRAVityFCAVTest offers a **Quick Modes** option located on the toolbar. By selecting *Load Transmit Patterns*, **GRAVityFCAVTest** will automatically load a default pattern (Figure 4.2) in both Ping and Pong banks ready to be transmitted.

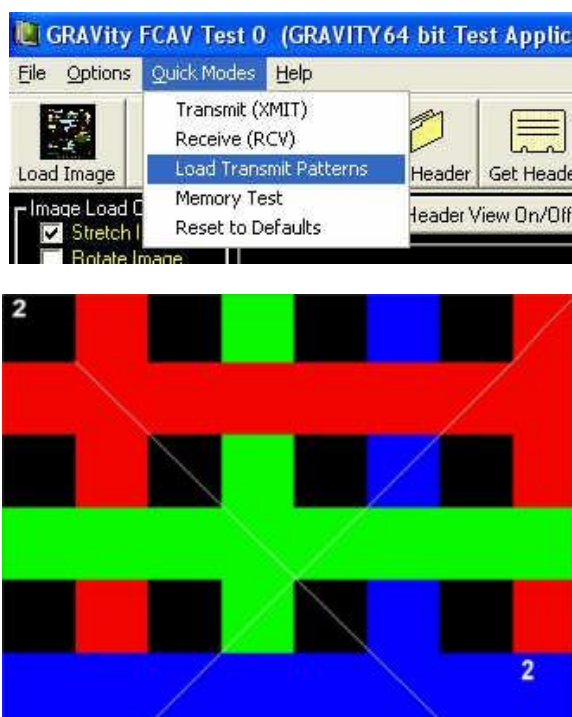
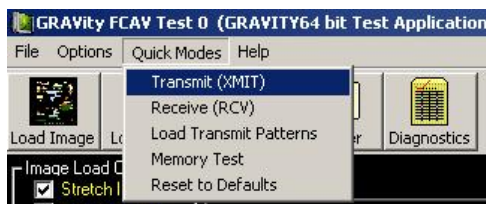


Figure 4.2

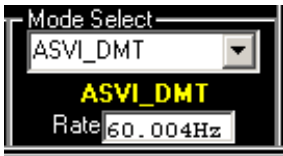
Once the image loads you can then select *Transmit (XMIT)* .



GRAVityFCAVTest will now be transmitting the test pattern across both Fiber and Copper.

4.3.2 Transmitting using manual controls

Use **Mode Select** pull down to select the desired FCAV profile for transmit. For the software to correctly transmit the FC-AV, this setting must match the profile of the incoming FCAV. The video rate for the selected mode is displayed below the mode.

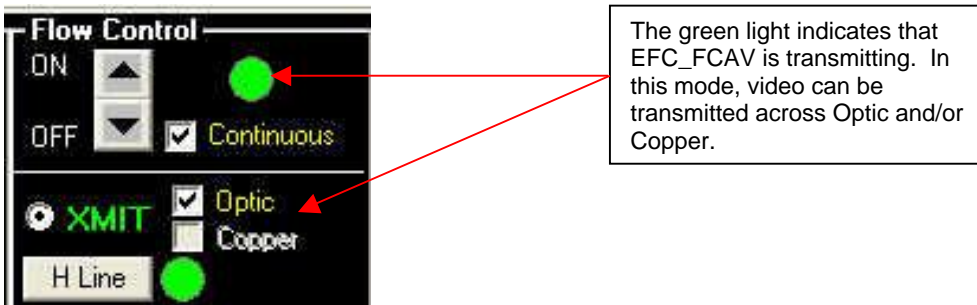


Once the correct mode is selected, you can load bitmap images in to the card using the “Load image” button. These images can be loaded into one of the image store (Ping or Pong) or both image stores. You can also load solid images using the “Load Solid” button.

You can stretch and odd sized bitmap to fit the profile or rotate the image 90 degrees by using these check boxes.



To start transmitting, select “XMIT” in the **Flow Control** box. By selecting either the Optic/Copper checkboxes you can transmit video out Optic, Copper or both.

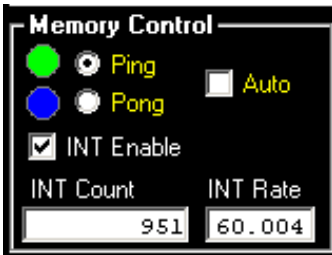


4.3.3 Transmitting from Ping/Pong image stores

The image transmitted on the FC-AV link can be from either the Ping or Pong image store depending on the radio buttons selected on the Memory Control area.

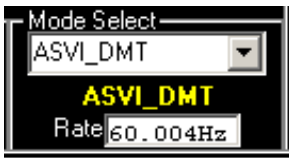
For normal operation the “AUTO” check box should not be selected and the Ping and Pong radio button will determine the transmitted image. With this setting the previously loaded image is transmitted at the designated frame rate on the FC-AV link. (Static)

When the “AUTO” check box is selected, the transmitted video alternates between the Ping and Pong image. This setting is not recommended since it is possible to transmit two distinct images in a alternating sequence on the fiber.

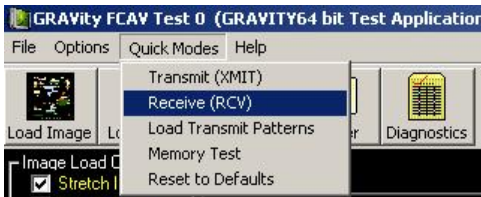


4.3.4 Receiving using *Quick Modes*

Before receiving with GRAVityFCAVTest, make sure that you have selected the correct FPGA load that matches the transceiver.

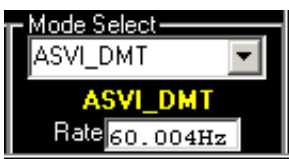


If you want to quickly view your incoming video you can select the *Receive (RCV)* on the Quick Modes toolbar.



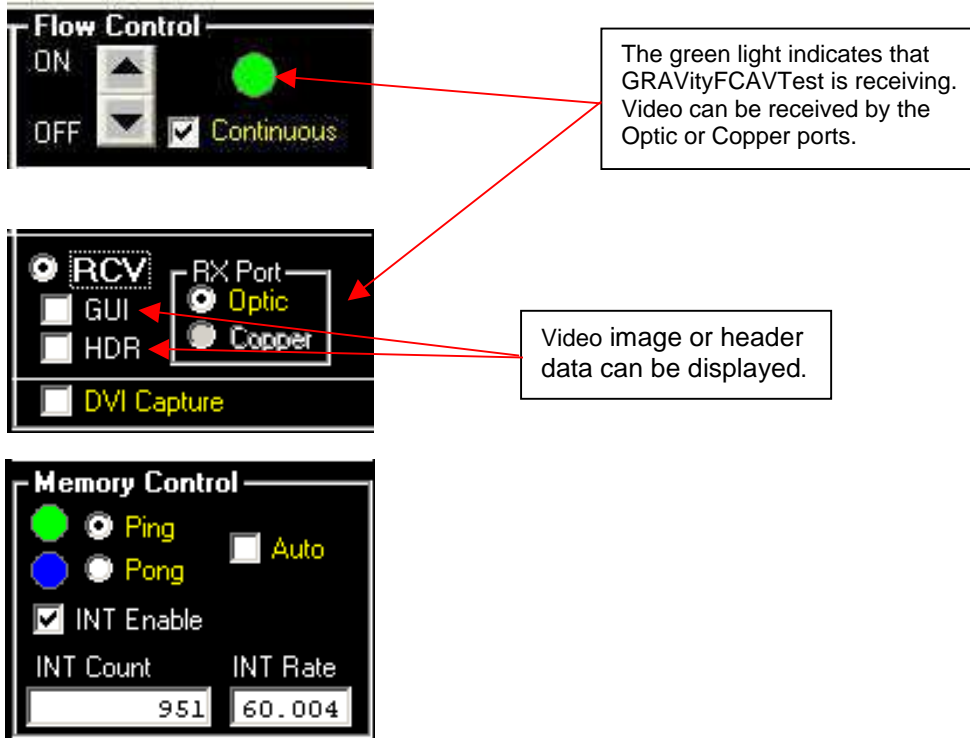
4.3.5 Receiving using manual controls

Use the **Mode Select** pull down to select the desired FCAV profile for receive. For the software to correctly interpret received video, this setting must match the profile of the incoming FCAV.



4.3.6 Receiving video and header data to the GUI

To start receiving, select “RCV” in the **Flow Control** box. By selecting either the Optic/Copper checkboxes you can receive video by the Optic or Copper ports. Flow is started and stopped using the Flow Control ON/OFF control.



The received video image can be displayed within the software GUI by making the following selections.

Select the “AUTO” checkbox in the **Memory Control** group. This allows the incoming video to switch between both image stores.

Select the “INT Enable” checkbox in the **Memory Control** group. This enables interrupt to be sent to the software for each received video frame.

Select the “GUI” checkbox. This indicates to the software that the incoming video frames can be retrieved from the card and displayed in the GUI.

4.3.6.1 Image update rate in the GUI

The update rate of the video within the GUI will depend on several system variables including the following:

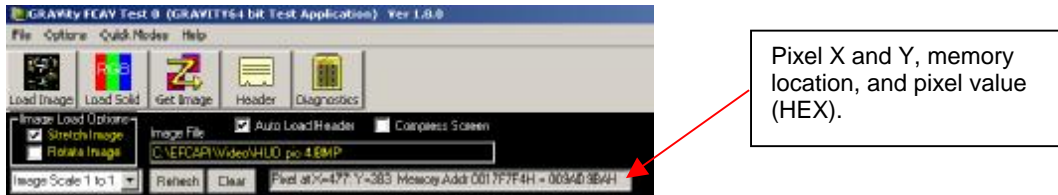
- CPU speed
- PCI type (e.g., 32 v. 64 and clock speed)
- Graphics card (time required to paint the GUI image)
- PCI loading by other adapter cards

The Gravity card and the Application software will update the GUI as fast as possible depending on the system variable indicated above; however, it is likely the update rate will be

reduced from the FCAV container rate. This in effect will lead to a viewing of every Nth frame, and therefore, the full motion quality of the video may be reduced.

4.3.6.2 Checking pixel values in the GUI

The user may check the pixel value of an image in the GUI by right clicking the mouse on the region of interest in the GUI. Upon the right click the X and Y position, the memory location, and the pixel value (in HEX) will be automatically displayed in text box above the image.



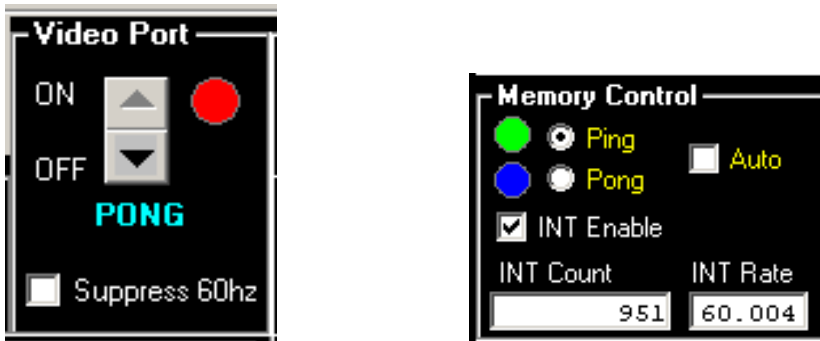
4.3.7 Receiving video to the VGA Port

The **Video Port** control box allows you to view the contents of the Ping and Pong buffers on a standard computer monitor connected to the Gravity card. This can be a static image captured from fiber or can be live video received on fiber.

The **Memory Control** box allows you to control the source for the video. You can use the Ping/Pong radio buttons to switch between the two stores.

When the Flow Control is on, you can check the AUTO box to view live incoming video.

The "Suppress 60Hz" checkbox control will limit the VGA update rate to 30Hz (note that this control only works with specific firmware versions).



4.3.8 Image update rate on the VGA Port

The VGA output of the Gravity card uses 60 Hz refresh for all supported resolutions. However, the update rate of images may be less than 60Hz depending on the incoming frame rate for the FC-AV video.

The Gravity card conforms to precise VESA output timing due to the fact that there are two image buffers on the Gravity card. When one of these buffers is receiving image data from the fiber, data from the other buffer is being sent to the VGA output. The vertical and horizontal line timing being sent to the VGA is independent of that being received over the FC-AV link. Therefore, the VGA timing can be tuned independent of what is being received, and consequently, can have precise control over horizontal timing (and jitter) to meet the requirements of standard computer monitors.

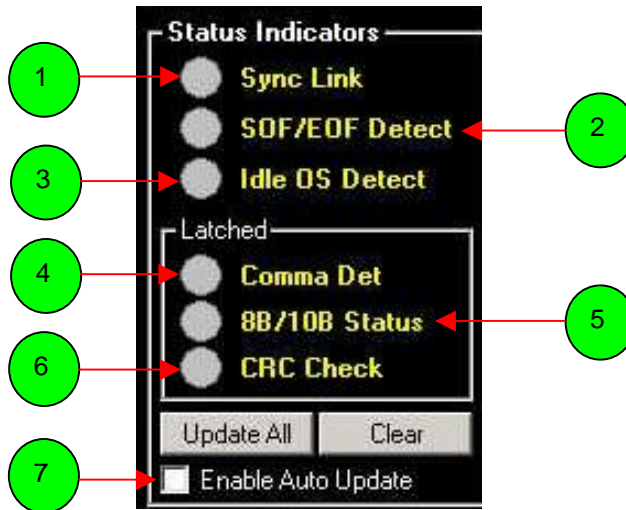
The receive circuit used does not impose any timing requirements for the incoming video on the fiber. The VGA images will always be displayed. There is no requirement that the horizontal line timing (FC frame timing within the container) be uniform on the fiber. There is also no requirement for the timing of containers – the card is tolerant of any incoming vertical rate between 1 Hz and 65 Hz.

In cases where the incoming video is not precisely 60.00Hz, the update of images on VGA will be reduced to 30Hz (every other frame). For very slow vertical rates on the fiber, the update rate on the VGA port may reduce further.

For 60Hz FC-AV inputs that are close to the VGA 60Hz, but are still asynchronous, the synchronization circuit will update the VGA images at 60Hz whenever the SOFi of the FC-AV input is aligned with the VGA's VSYNC output. When these are not aligned, the update rate will be 30Hz. The "Suppress 60Hz" checkbox control will limit the VGA update rate to 30Hz in all cases even if the SOFi and VSYNC are aligned.

4.3.9 Receiver Status

You can check the status of incoming FCAV video by referring to the **Status Indicators box**. Check the "Enable Auto Update" box to sense Sync Link, SOF/EOF Detect, Idle OS Detect, Comma Det, 8B/10B Status and CRC Check.



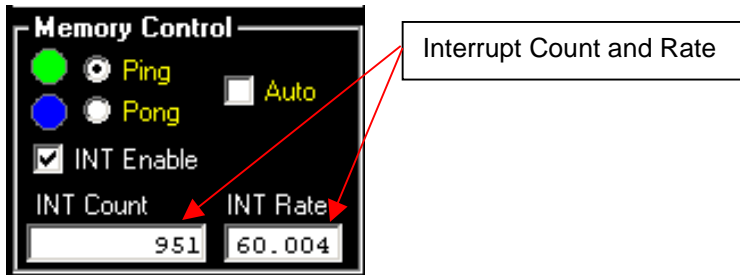
Status Indicators

1. Link Synchronization
 - Green indicates that the serializer/encoder is able to recognize (and synchronize on) 8b/10b codes
2. SOF/EOF Detect
 - This indicator is green when both SOFi and EOFt ordered sets are detected by the receiver.
 - SOFi must occur at a rate faster than 7 Hz.
 - There is no SOF/EOF Detect until flow is on.
3. Idle OS Detect
 - Green indicates the presence of Idle Ordered Sets.
 - This indicator is unlatched – will transition when receiving video.
4. Comma Detect
 - Green indicates that a K28.5 comma character has been detected by the receiver.
 - This indicator is unlatched – will transition when receiving video.
5. 8B/10B Status
 - Latched red when receiver detects any 8b/10b error.
 - Failure is when code is not in 8b/10b table.
 - This signal is latched, and must be reset by Clear button.
6. CRC Check
 - Latched red when receiver detects any CRC error.
 - CRC failure is when received CRC (for any FC frame) fails to match calculated CRC.
 - This signal is latched, and must be reset by Clear button.
7. Enable Auto Update
 - Status Update
 - Check “Enable Auto Update” to monitor link.
 - Updates approximately every second.
 - “Update All” for a one time sample of status.
 - There are both latched and unlatched bits
 - The latched indicators will persist until cleared.
 - The unlatched bits will change in real time.

The horizontal line feature (H Line) can be used to add a scrolling horizontal line to the transmitted video. The scrolling line, which moves one pixel down per frame, provides a convenient method visually identify live video on the FCAV receiver.



The Interrupt enable check box (INT Enable) should be checked when automatic DMA unloading of the image stores is desired.



The INT Count box shows the number of Ping and Pong interrupts and DMA transfers. This value increments for each complete frame transferred. In Receive mode, if the GUI display is On this count may not correspond to the number of frames in the incoming FC-AV, since the DMA transfer on receive may take longer than each incoming frame. The INT Rate box shows the Ping/Pong Interrupt rate for images transferred. This rate may also be less than the incoming frame rate when in receive mode when the GUI display is On.

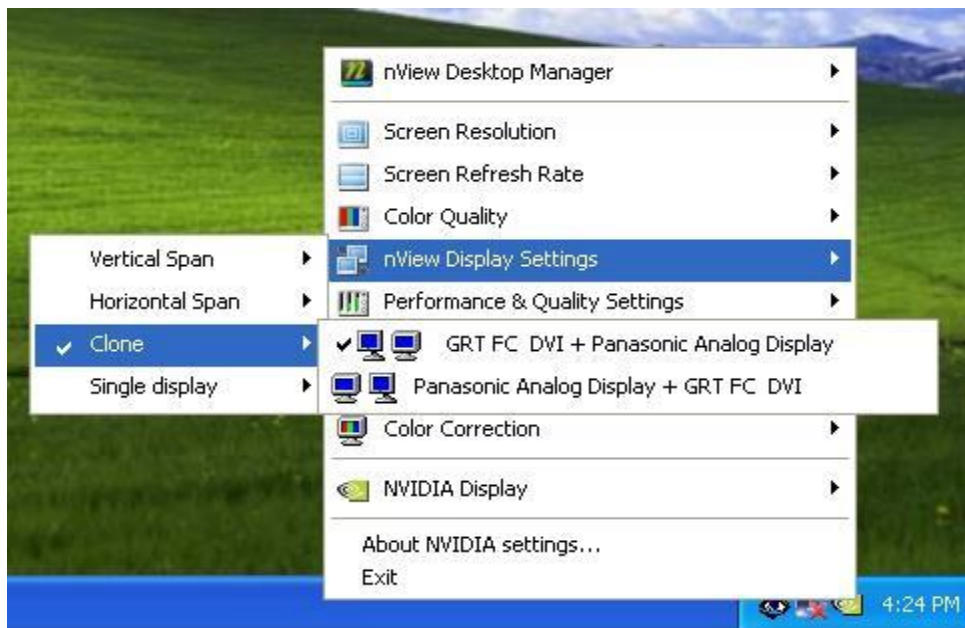
4.4 Transmit with DVI

When Using a Gravity card with a DVI Mezzanine, make sure that the DVI input source is set for the correct resolution. For instance, when working with FCAV interfaces at XGA resolution, make sure the DVI source is set for XGA.

For best results, make sure that the DVI graphics card is set for 32 bit true color, and for 60Hz.

When using dual head cards, the recommended configuration is a “clone” output, where one output can be used as the computers primary monitor and the second output can be used as a DVI input to the Gravity card.

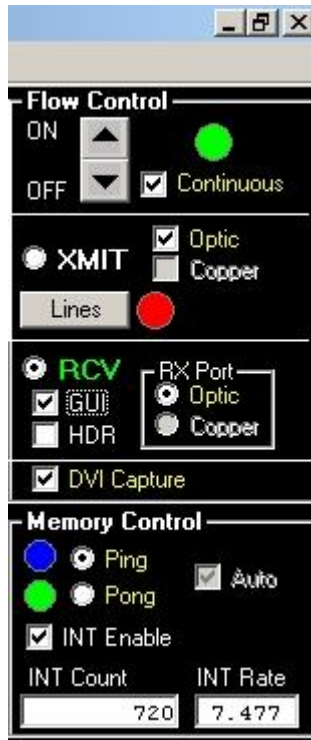
Most dual head graphics cards will have a control panel for this type configuration. For example, nVidia has the following menus for selecting “clone” operation.



4.4.1 Verifying DVI input

Once input connections have been made to the DVI source, **GRAVityFCAVTest** may be used to verify that the DVI input is configured correctly by capturing the DVI video to the GUI.

The following selections should be made to capture DVI to the GUI

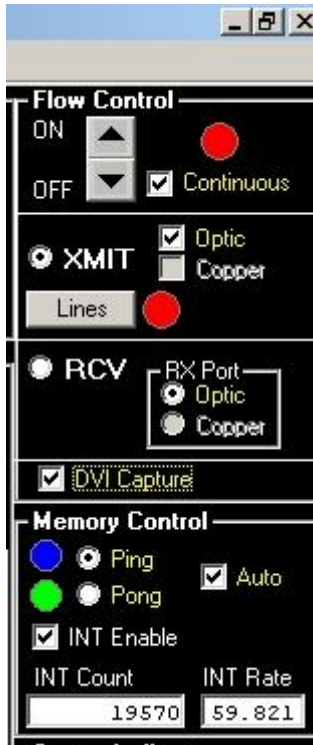


When these settings have been made, the software will capture DVI frames to the GUI view port. The update rate will be limited by available PCI bandwidth and other system performance characteristics.

4.4.2 Transferring DVI input to FCAV output

When transferring the incoming DVI onto the FCAV output it is important to check the DVI Capture check box as well as the AUTO check box.

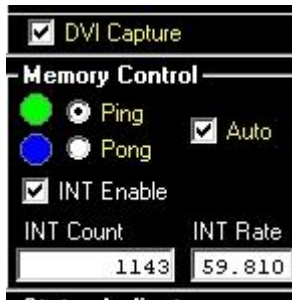
The figure below shows the correct settings:



Once these settings have been made, turning on Flow control will begin the FCAV transmission.

4.4.3 DVI frame rate

When transmitting FCAV that originates from a DVI source, the video frame rate will match the rate of the DVI source. **GRAVityFCAVTest** will report this rate in the INT rate box in the Memory Control group.



4.4.4 Using GRAVityFCAVTest in XGA resolution

When the DVI source for the Gravity card is a Dual Head graphics card and this card supplies video for the primary monitor, the required FCAV resolution may govern the resolution of the primary monitor.

For instance, when setting up a dual head card as an XGA DVI source, it is likely, depending on the graphics card used, that the primary control monitor will be forced into an XGA resolution.

Where this is the case, the GRAVityFCAVTest application can be optimized for XGA. To do this, check the Compress Screen box as shown below:



4.5 Ping and Pong Memory Map

The Ping and Pong image stores are memory mapped on the card as follows:

ADDRESS (BYTE OFFSET)	IMAGE STORE	DESCRIPTION
0x00000000 - 0x0000007C	PING	FC HEADER, CONTAINER HEADER, ANCILLARY DATA FOR PING COLOR FORMATS.
0x00000080 - 0x007FFFFC	PING	PING COLOR IMAGE DATA - STORED AS 32 BIT TRUE COLOR.
0x00800000 - 0x0080007C	PING	FC HEADER, CONTAINER HEADER, ANCILLARY DATA FOR PING MONO FORMATS.
0x00800080 - 0x00FFFFFFC	PING	PING MONO IMAGE DATA - STORED AS 8 BIT MONOCHROME.
0x01000000 - 0x0100007C	PONG	FC HEADER, CONTAINER HEADER, ANCILLARY DATA FOR PONG COLOR FORMATS.
0x01000080 - 0x017FFFFC	PONG	PONG COLOR IMAGE DATA - STORED AS 32 BIT TRUE COLOR.
0x01800000 - 0x0180007C	PONG	FC HEADER, CONTAINER HEADER, ANCILLARY DATA FOR PONG MONO FORMATS.
0x01800080 - 0x01FFFEFC	PONG	PONG MONO IMAGE DATA - STORED AS 8 BIT MONOCHROME.
0x01FFFF00 - 0x01FFFFFF	PLD	CONTROL AND STATUS REGISTERS

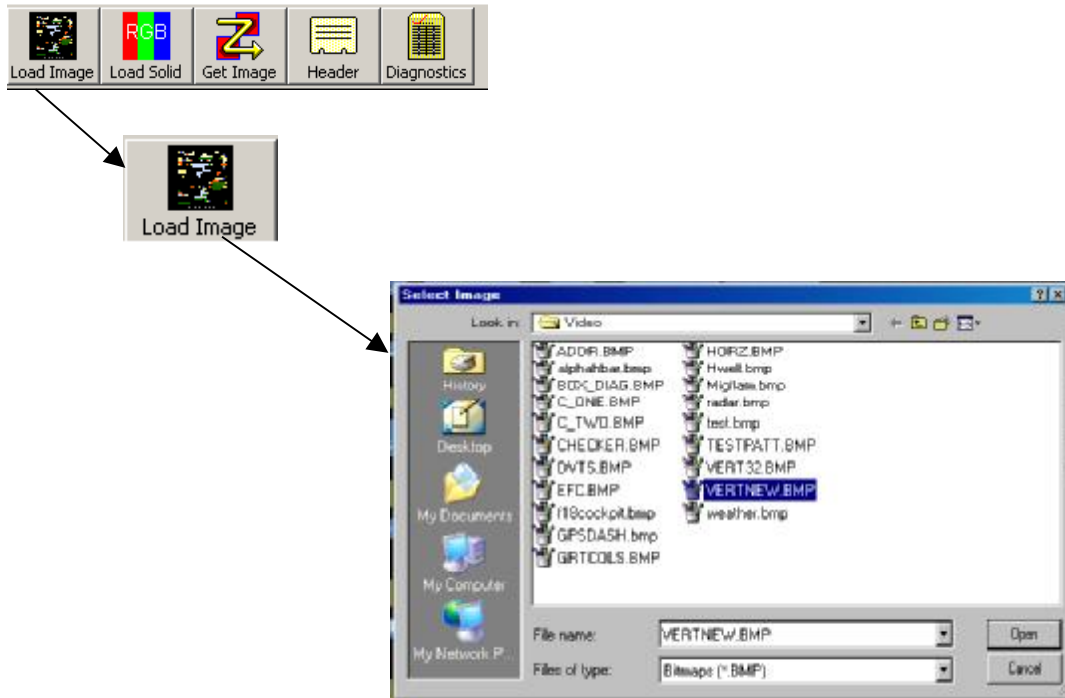
Table 3.1 Local SDRAM Memory Allocation

Color images each have dedicated memory allocated in Ping and Pong. The header data precedes the image in the memory map. The same memory locations are used when transmitting and receiving.

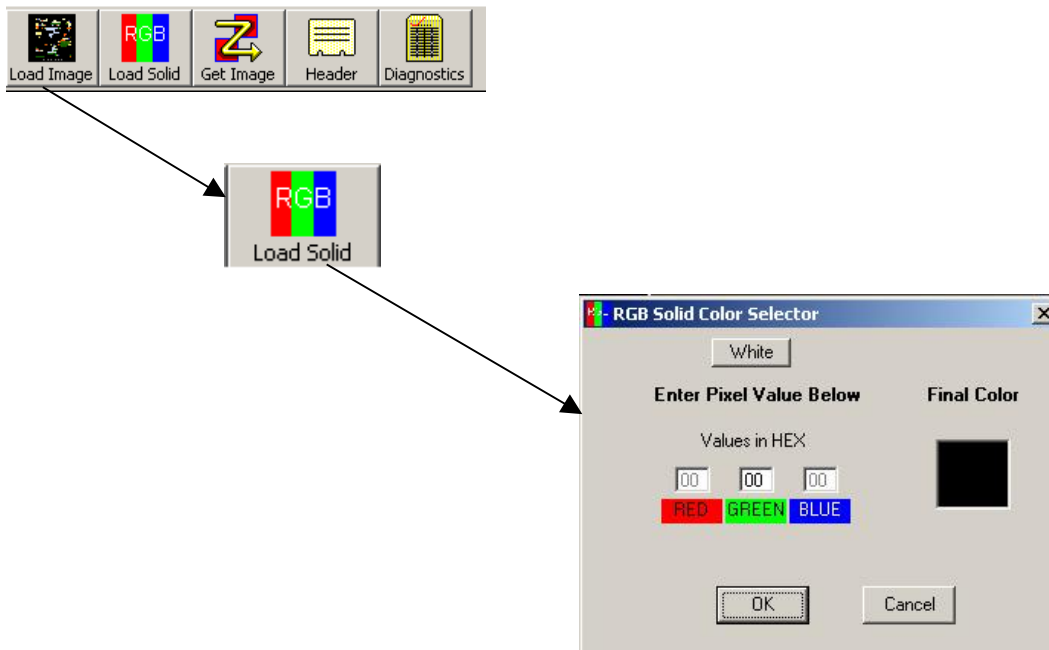
For more information on the Ping and Pong memory map and configuration registers please refer to the FCAV SDK documentation.

4.6 Loading and retrieving images

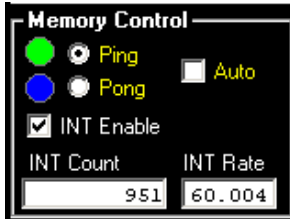
To load an image, select the “Load Image” button and locate the file you wish to load.



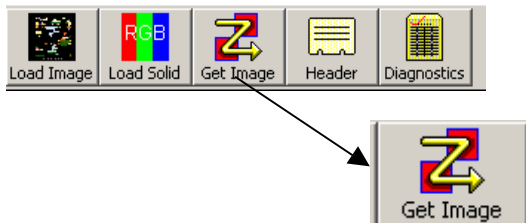
You may also load solid-color images by selecting the “Load Solid” button. Type in the values you want and click ok to load a solid color.



You can load images both in the PING and PONG buffer store areas by making the selection under the **Memory Control** box. To automatically alternate between Ping and Pong buffers while transmitting, you can select the "Auto" option. When Auto is unchecked GRAVityFCAVTest will only transmit out of the selected buffer.



To retrieve an image stored in the Ping or Pong buffer, use the **Get Image** button.



The images will be automatically displayed in the GUI and may then be stored to bitmap file.

Incoming video can be displayed in the GUI automatically by making the following control selections:

Flow Control Continuous box = checked

Flow Control GUI = checked

Memory Control Auto box = checked

Flow Control Flow = ON

4.7 Loading and retrieving headers

To load a header, use the **Header** button. Typically, a header file is loaded at the same time as an image is loaded

The screenshot shows the software interface with the **Header** button highlighted. Two callout boxes point to the **Read** and **Write** buttons in the Header dialog box, with the text "Press to read header data." and "Press to write header" respectively.

The **Header** dialog box displays the following data:

```

Header
File: [Open] [Save] [Open] [Read] [Write] [Ping Pong] [Ping Pong]
PC Frame Header
Word
0 P_CTL 44 Dest_ID 000000
1 CS_CTL 00 Source_ID 000000
2 TYPE 60 F_CTL 000000
3 SEQ_ID 00 DF_CTL 00 SEQ_CNT 0000
4 CC_ID 0 0000 SK_ID 0000
5 Parameter 00000000
Container Header
Word
6 Container Count 00000000
7 Clip ID 00000000
8 (unused) 00000000
9 (unused) 00000000
10 Video Frm Rate 07 Trans. Rate 01 (reserved) 00 (reserved) 00
11 Mode 00 Number of Obj's 04 (reserved) 00 Ext. Header Size 00
12 Type - Ancillary 70 Link Pointer 00 SPDU Index 0000
13 Object 0 Size 00000000
14 Object 0 Offset 00000000
15 (unused) 00000000
16 Type - Audio 40 Link Pointer 00 SPDU Index 0000
17 Object 1 Size 00000000
18 Object 1 Offset 00000000
19 (unused) 00000000
20 Type - Video 10 Link Pointer 00 SPDU Index 0000
21 Object 2 Size 00240000
22 Object 2 Offset 00000000
23 (unused) 00000000
24 Type - Video 10 Link Pointer 00 SPDU Index 0000
25 Object 3 Size 00000000
26 Object 3 Offset 00240000
27 (unused) 00000000
Object 0 Ancillary Data
Word
28 Num of Lines 0000 Num of Cols 0400 Frame 0
29 CI 1 PA 0 PAD 0 PIN 0 SA 7 SE 7 SC 7 SD 0
30 Container CRC 00000000
31 Source Status 00 ICS 00 User Defined 0000
  
```

Pressing the **Header** button brings up the Header utility (shown above). The Header utility automatically opens the default header data for the currently selected Video Mode. It also retrieves the header data format and displays the data using this format. The header data format file name is taken from the INI file, but can be changed by selecting the "Open" button and selecting a different header data format file. Note that the currently selected header format file name is saved in the INI file on program exit.

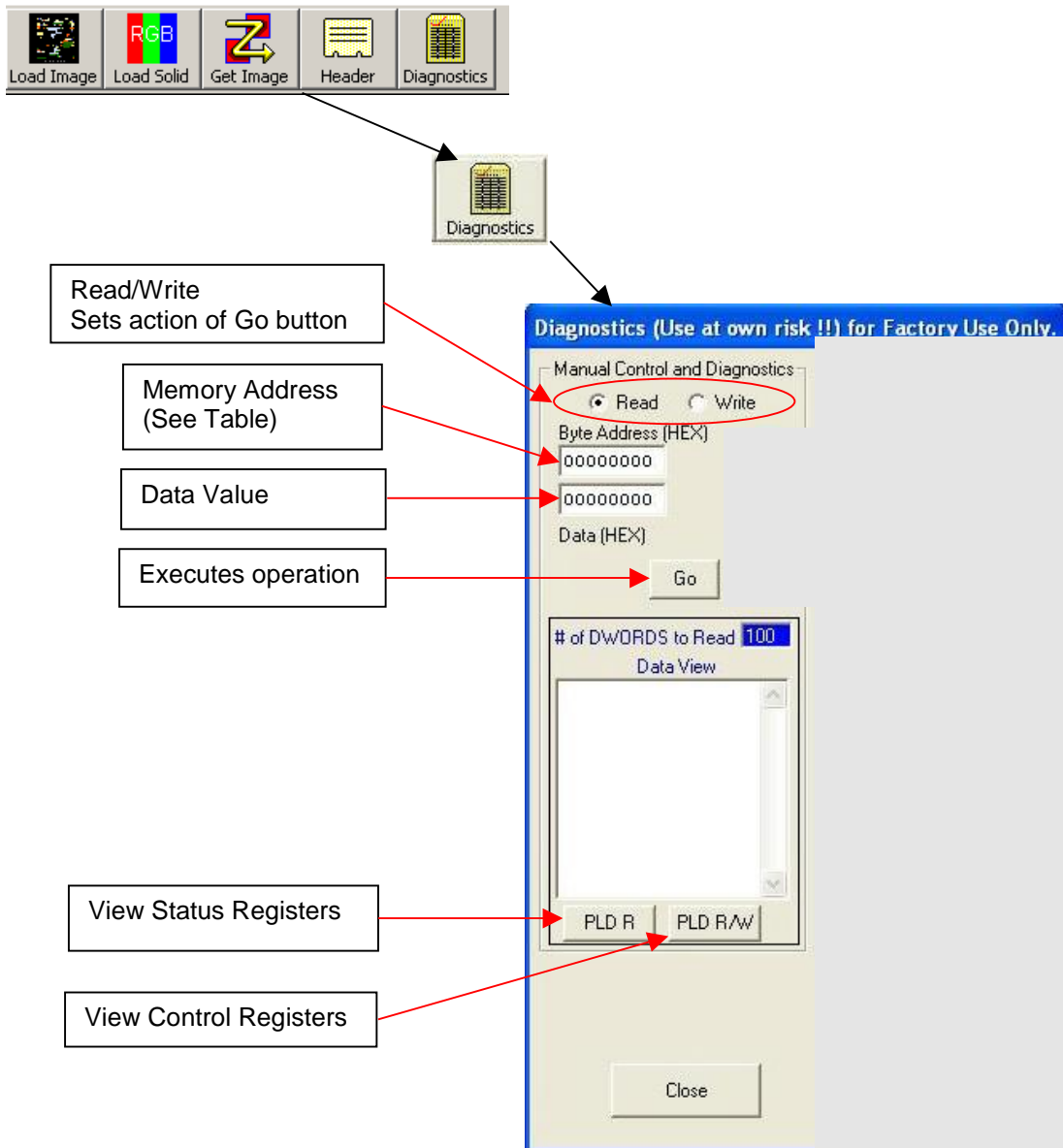
To load the default header data into the card, press the **Write** button on the Header utility display form. Note that the checkboxes to select Ping and Pong buffers will both be selected by default. To write a header to just one of the buffers, select the appropriate checkbox. Normally you will want to load headers into both buffers at the same time. If the default header data is not acceptable you may alter the data before you load it by performing the desired edits before pressing the **Write** button.

To read the header data from the card, select the radio button for the memory area to read, then press the **Read** button to read the header data.

See the section titled "Header Data and Formats" for more information on header files, header formats, and the usage of the Header Utility. This section contains more detail on the usage of the Header Utility.

4.8 Diagnostics Menu

Disclaimer: The diagnostic functions are provided for factory test and verification. Advanced users may find some of the functions helpful for system verification, but it is possible to override safety features of the application that can result in corrupted data, and system lock up. The grayed out section of the menu box should not be accessed unless specifically instructed by a factory representative.



= For Factory Use Only

The diagnostics menu can be used to view the contents of the PLD status registers and to view the content of the Ping and Pong image stores or to change pixel values in memory.

The **PLD R** button is used to retrieve the contents of status registers in the PLD and the **PLD R/W** is used to retrieve content of the PLD control registers.

Ping and Pong memory contents can be observed using the following steps:

1. Enter the base address (In hex) in the top most entry box
2. Check the **Read** radio button
3. Enter the number of Double Words to read
4. Click the **GO** button

The memory contents will be displayed in the view port.

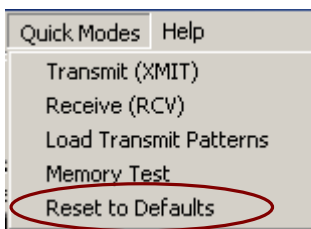
Ping and Pong memory, header and pixel data, can be modified using the following steps:

1. Enter the base address (In hex) in the top most entry box
2. Check the **Write** radio button
3. Enter the new value (In hex) in the bottom most entry box
4. Click the **GO** button

The memory contents will be displayed in the view port. Note that when a write command is executed the number of displayed **DWORDS** is automatically set to 1.

4.8.1 Initialization and Reset

The **Initialize/Reset** Button can be used to perform a PLD reset to bring the control register to its initial conditions. Under Quick Modes you can reset GRAVityFCAVTest by selecting "Reset to Defaults".



4.8.2 HOSTGRAV.INI File settings

The file named HOST.INI will be in the root directory of GRAVityFCAVTest.EXE which is typically C:\EFCAPI.

This file can be edited with a text editor to change entry values for configuration of the EFCTEST software.

The following is a listing of the typical HOST.INI file.

```
AutoEnum=1
DriverNum=0
CustomX=640
CustomX=480
VidDirColor=C:\EFCAPI\Video\
VidDirMono= C:\EFCAPI\Video\
HdrDir= C:\EFCAPI\Headers\
SaveDir= C:\EFCAPI\Video\Saved\
Color=0
Media=
Title=
HdrFmtFile=
```

Only three of the entries should be changed manually. The three entries that can be changed manually are:

```
AutoEnum
DriverNum
Color
. . . . .
```

The **AutoEnum** entry can be either 0 or 1. If it is 1 (preferred), then the driver number opened is automatically incremented for each new FCAV TEST instance. This is the preferred parameter setting and works well for PC's with several Gravity cards installed.

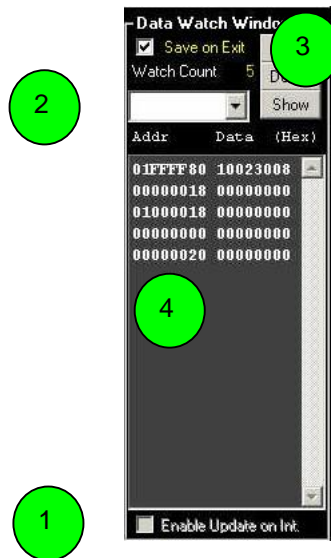
If **AutoEnum** = 0, then the driver number opened will be the value in the **DriverNum** entry. If **DriverNum** = 1, and **AutoEnum** = 0, then the driver opened will be 1. In cases where **AutoEnum** = 0, and there are multiple cards in a PC, then a separate directory is required for each Gravity card and each directory should have a copy of GRAVityFCAVTest.EXE and its own HOST.INI. This allows the HOST.INI file to be associated with the GRAVityFCAVTest.EXE that controls a particular card.

The **Color** entry can be used to change the background color of the EFCTEST application. The default is 0 which is black. The decimal RGB value defines the color.

4.9 Data Watch Window

The Data Watch Window can be used to view the contents at different memory locations.

1. Make sure the checkbox for Enable Update on Int is checked.
2. Enter in a memory location.
3. Click on Add.
4. View the Address and Data in the window. Some data will be static and other dynamic.

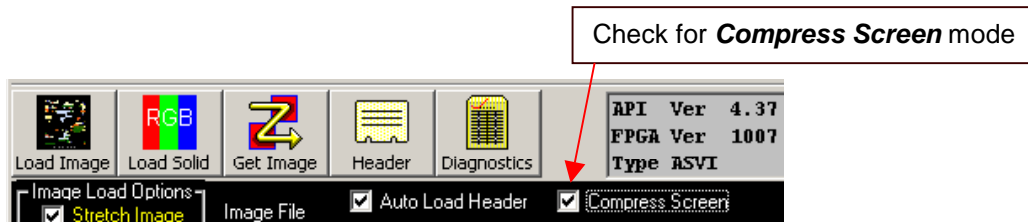


To remove a memory location, click on the down arrow in the memory location entry box(#2). Select the memory location from the drop down list then click on the Delete button.

Refer to the Memory Map Table 3.1.

4.10 Compress Screen Mode

The GRAVityFCAVTest application has been designed to work best with a computer display resolution setting of 1280x1024 or greater. However, in some cases (such as when using the DVI capture module) it is necessary to use the 1024x768 setting for the computer display, The **Compress Screen** mode can be used if you require the computer display to be set to the 1024x768 setting. Not that in this mode the FCAV image will always be displayed with 2-to-1 compression. To use this mode, check the **Compress Screen** checkbox shown below:



5 Fibre Channel Frame Format and Ordered Sets

The Fibre Channel transport layer is comprised of frames as defined below.

The basic unit of transport for video and header data will be FC frame. Each FC frame has the structure shown below (and as described in the referenced FC-PH specification).

Word	Identifier	Byte 0	Byte 1	Byte 2	Byte 3
XX	Idle word	K28.5	D21.4	D21.5	D21.5
XX	Idle word	K28.5	D21.4	D21.5	D21.5
0	SOF_{in}	K28.5	D21.5	D23.x	D23.x
1	Frame Header R_CTL D_ID	FC Frame Header byte	FC Frame Header byte	FC Frame Header byte	FC Frame Header byte
2	Frame Header CS_CTL S_ID	FC Frame Header byte	FC Frame Header byte	FC Frame Header byte	FC Frame Header byte
3	Frame Header Type F_CTL	FC Frame Header byte	0000 x000 (Bit 19)	FC Frame Header byte	FC Frame Header byte
4	Frame Header	FC Frame Header byte	FC Frame Header byte	0000 0000 (SEQ_CNT)	0000 0000 (SEQ_CNT)
5	Frame Header	FC Frame Header byte	FC Frame Header byte	FC Frame Header byte	FC Frame Header byte
6	Frame Header	FC Frame Header byte	FC Frame Header byte	FC Frame Header byte	FC Frame Header byte
7 to N*	Payload	Data	Data	Data	Data
N+1	CRC	Data	Data	Data	Data
N+2	EOF_{n/t}	K28.5	D21.x	D21.x	D21.x
XX	Idle word	K28.5	D21.4	D21.5	D21.5

Table 5.0a Fibre Channel Frame Definition

The 24 FC Frame header bytes originate from header files loaded via GRAV_FCAV. All fields can be set by the user and may be defined differently than described in FC-PH. However, Gravity FC-AV will automatically update the sequence count and the end sequence bit as described below.

SEQ_CNT	The <i>Sequence Count</i> is used as defined in FC-PH and is incremented by one LSB for each consecutive FC frame of the FC-AV data sequence.
END_SEQ	The <i>End Sequence</i> bit (Word 3, bit 19) is set to logic 1 for the last FC frame of a data sequence (FC-AV container), and set to logic 0 for all other frames of the sequence.

Table 5.0b Fields updated by hardware

5.1 Idle Words

Idle ordered sets, K28.5, D21.4, D21.5, D21.5, are transmitted between FC frames and their number can be adjusted to change the overall timing of the video field rate (or FC-AV container rate).

When receiving, the Gravity FC-AV requires a minimum of six Idle ordered sets between each Fibre Channel frame.

When transmitting, the number of Idle ordered sets between FC frames can be varied to adjust the video frame time.

The Gravity FC-AV automatically pads the Idle OS after the last FC frame in a sequence, such that the required video frame rate (required by the profile) is achieved. Users may set the minimum number of Idle OS between all other frames by writing to a PLD control register. (See EFC API User's Manual)

5.2 Start of Frame

The first frame of an FC-AV container transfer sequence uses the Start of Frame Initiate Class 1 SOF_{i1} ordered set (K28.5, D21.5, D23.2, D23.2). Subsequent frames of the FC-AV container sequence use the Start of Frame Normal Class 1 SOF_{n1} ordered set (K28.5, D21.5, D23.1, D23.1).

5.3 Data Payloads

The first frame of an FC-AV container data sequence contains the container header and Object 0 ancillary data as its data payload.

Subsequent frames of the FC-AV container sequence contain the Object 2 video pixel data.

5.4 Cyclic Redundancy Check (CRC)

All data frames include a 4 byte CRC using the following 32-bit polynomial:

$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

5.5 End of Frame

All frames except the last frame of an FC-AV container data transfer sequence use the End of Frame Normal EOF_n ordered set (beginning RD Negative - K28.5, D21.4, D21.6, D21.6 or beginning RD Positive - K28.5, D21.5, D21.6, D21.6).

The last frame of an FC-AV container data transfer sequence uses the End of Frame Terminate EOF_t ordered set (beginning RD Negative - K28.5, D21.4, D21.3, D21.3 or beginning RD Positive - K28.5, D21.5, D21.3, D21.3).

6 Fibre Channel Containers

Gravity FC-AV cards adhere to the FC-AV container system of transport as described in Clause 5 of the standard.

For Gravity FC-AV cards, a container is simply the set of Fibre Channel frames used to transport a single video frame. For uncompressed progressive scanned video, only two container objects are required: Object 0, which contains header information, and Object 2 which carries the video payload. Object 1, by definition, is restricted to Audio and is not used by Gravity FC-AV.

6.1 Object 0 FC frame

In all video profiles, container Object 0 is a single Fibre Channel frame. This frame is the first frame of the sequence and begins with an SOFi (rather than SOFn). All subsequent frames (Object 2) carry video payload.

The Object 0 Fibre Channel frame is comprised of 32 long words (128 bytes) as follows:

FC Frame header	6 Lwords	(24 bytes)
FC-AV Container Header	22 Lwords	(88 Bytes)
Ancillary Data	4 Lwords	(16 bytes)
Total	32 Lwords	(128 bytes)

The FC frame header is that same as that defined in section 5. The container header and ancillary data are described below.

6.2 Object 0 FC-AV Container Header

Gravity FC-AV cards allow users to adhere to the standard definition of the Container Header (as shown in table 6.2) or, since Gravity cards use text based header files, part or all of these fields may be user defined.

Word	Identifier	Byte 0	Byte 1	Byte 2	Byte 3
0	Container Count	(MSB)			(LSB)
1	Clip ID	(MSB)			(LSB)
2	Container Time Stamp	(MSB)			
3	Container Time Stamp				(LSB)
4	Transmission Type	(Video Frame Rate)	(Transmission Rate)	(Reserved)	(Reserved)
5	Container Type	(Mode)	(Number of Objects)	(Reserved)	(Size of Ext. Header)
6	Object 0 Class	(Type)	(Link Pointer)	(SPDV Index)	(SPDV Index)
7	Object 0 Size	(MSB)			(LSB)
8	Object 0 Offset	(MSB)			(LSB)
9	Object 0 Object	Type Defined	Type Defined	Type Defined	Type Defined

	Type Defined				
10	Object 1 Class	(Type)	(Link Pointer)	(SPDV Index)	(SPDV Index)
11	Object 1 Size	(MSB)			(LSB)
12	Object 1 Offset	(MSB)			(LSB)
13	Object 1 Object Type Defined	(Type)	(Link Pointer)	(SPDV Index)	(SPDV Index)
14	Object 2 Class	(Type)	(Link Pointer)	(SPDV Index)	(SPDV Index)
15	Object 2 Size	(MSB)			(LSB)
16	Object 2 Offset	(MSB)			(LSB)
17	Object 2 Object Type Defined	Type Defined	Type Defined	Type Defined	Type Defined
18	Object 3 Class	(Type)	(Link Pointer)	(SPDV Index)	(SPDV Index)
19	Object 3 Size	(MSB)			(LSB)
20	Object 3 Offset	(MSB)			(LSB)
21	Object 3 Object Type Defined	Type Defined	Type Defined	Type Defined	Type Defined

Table 6.2 Object 0 Container Header

6.3 FC-AV Object 0 Ancillary Data

Also included in the Object 0 FC frame are 24 bytes of Ancillary Data (as shown in table 6.3). Since the Gravity card uses text based header files, part or all of these fields may be user defined.

Word	Identifier
0	Ancillary Data
1	Ancillary Data
2	User Defined Data
3	User Defined Data

Table 6.3 Object 0 Ancillary data

6.4 FC-AV Object 2 Video Data

In all video profiles, container Object 2 is comprised of a set of Fibre Channel frames. These frames begin with SOF_n and can carry up to 2112 bytes video of video payload per FC frame. The exact number of bytes per FC frame and the total number of FC frames, depends on the video profiles.

Gravity FC-AV cards, using the standard firmware, support video profiles as described in Section 7. Supported video profiles insert one half of a video line Using three bytes per pixel. in each FC payload frame.

7 Header Data and Formats

7.1 Header Utility

The GRAVityFCAVTest program has a Header utility that replaces the "Load Header" and "Get Header" functions in previous versions of the program. This new utility allows the user to open a header data file, edit header data, save header data, load the header data into a card, or read header data from the card. The Header Utility simplifies the typical operations performed with header data. The Header Utility makes use of the new HFD (Header Format Definition) files described in the section below.

The screenshot shows the Header Utility interface with several callouts explaining its functions:

- Opens (reads) a header format file.** Points to the 'Header' icon in the top toolbar.
- Opens (reads) or saves (writes) a header data file.** Points to the 'Open' button in the 'File' menu.
- Writes header data to the card. The Ping/Pong checkboxes select the buffer(s) to be written.** Points to the 'Write' button and the 'Ping' and 'Pong' checkboxes in the 'Card' menu.
- Reads a header from the card. The Ping/Pong radio buttons select the buffer to be read.** Points to the 'Read' button and the 'Ping' and 'Pong' radio buttons in the 'Card' menu.

The main window displays the following data:

```

FC Frame Header
Word
0 F_CTL 44 Test_ID 000000
1 CS_CTL 00 Source_ID 000000
2 TYPE 40 F_CTL 000000
3 SEQ_ID 00 DF_CTL 00 SEQ_CMT 0000
4 OK_ID 0 0000 RC_ID 0000
5 Parameter 00000000

Container Header
Word
6 Container Count 00000000
7 Clip ID 00000000
8 (unused) 00000000
9 (unused) 00000000
10 Video Frm Date 07 Trans. Rate 01 (reserved) 00 (reserved) 00
11 Mode 00 Number of Obj 04 (reserved) 00 Ext. Header Size 00
12 Type - Ancillary 50 link Pointer 00 SPOU Index 0000
13 Object 0 Size 00000010
14 Object 0 Offset 00000050
15 (unused) 00000000
16 Type - Audio 40 link Pointer 00 SPOU Index 0000
17 Object 1 Size 00000000
18 Object 1 Offset 00000050
19 (unused) 00000000
20 Type - Video 10 link Pointer 00 SPOU Index 0000
21 Object 2 Size 00200000
22 Object 2 Offset 00000050
23 (unused) 00000000
24 Type - Video 10 link Pointer 00 SPOU Index 0000
25 Object 3 Size 00000000
26 Object 3 Offset 00200050
27 (unused) 00000000

Object 0 Ancillary Data
Word
28 Num of Lines 0300 Num of Col 0400 Frame 0
29 CI 1 PA 0 PA0 0 P70 0 SA 0 SB 0 SC 0 SD 0
30 Container CRC 00000000
31 Source Status 00 IGS 00 User Defined 0000
    
```

7.2 Header Data Files

The header data is stored in text files located in the EFCAPI Headers directory. These header data files specify the data (usually 128 bytes) that is loaded into the header area of the memory buffers. For FCAV, these bytes are:

FC Frame header	6 Lwords	(24 bytes)
FC-AV Container Header	22 Lwords	(88 Bytes)
Ancillary Data	4 Lwords	(16 bytes)

The text files can be created and edited using a text editor. The format of the files is as follows:

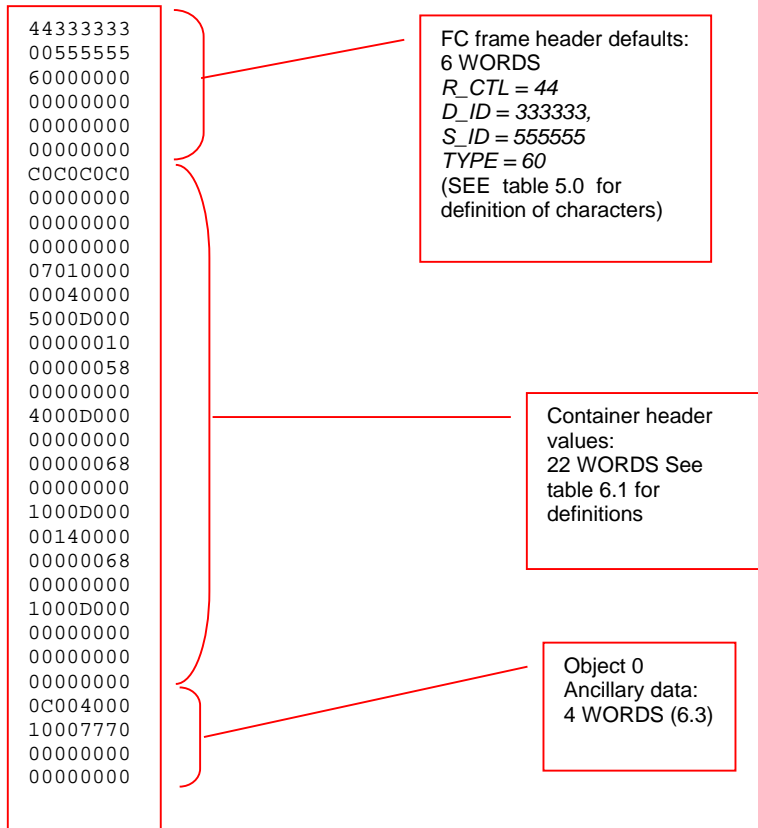


Figure 6.4 Example header data file

In header data files, ASCII characters represent Hex values to be used for the indicated fields. Some header data fields are updated by the Gravity card when transmitting. These fields include the SEQ_CNT and the F_CTL bit 19. In some firmware modes, the container count in the container header section is also incremented by hardware for each container transmitted.

Header data can be loaded manually using the Header utility accessed by the **Header** button on the top bar of the FCAV Test Application, or they can be loaded automatically, every time a bitmap file is loaded into memory, by checking the **Auto Load Header** button in the GUI. When the **Auto Load Header** button is checked, the application will look for a file with the same name as the selected FCAV mode. For instance, if the selected mode is FCAV10, then the Application will automatically load a header named FCAV10.txt (if this file exists in the

EFCAPI\Headers directory) whenever an image is loaded into the card. Some header data files may have more than the 128 bytes in the example above, as the GRAVityFCAVTest application supports several different header formats. See the section below for detail on Header Format files that support these header data files.

It is important that a header file be loaded into any image store that is being transmitted from. Therefore, when transmitting from the Ping image store, make sure that a header has been loaded into the Ping image area (see table 3.1), When transmitting from both Ping and Pong (in auto mode) a header must be loaded into both Ping and Pong image areas.

New header files may be created for different FC-AV profiles. Header files may also be created with intentional errors to verify error checking in receiving hardware.

7.3 Header Formats

Header formats are defined in Header Format Definition (HFD) files that define format of the header data. HFD files for all common Video Modes are included in the software distribution.

The header format is used to display the data for ease of editing by the program user. The header format is totally independent from the raw header data itself, except that the header data file loaded into the card should contain at the same number of defined data words in the format used. If the header contains less than the number of words defined in the format, the data will be padded to the length defined in the format. If the header data file contains more words than what is defined in the HFD file, the excess words are ignored.

All HFD files are named using the "hfd" extension (i.e. "FCAV14.hfd"). When the Header Utility is executed by pressing the "Header" button on the main screen, the program will load a HFD file and header data file. The program will first look for the HFD file name currently saved in the INI file. If that file is not specified or is not found, the program will look for a HFD file with the same name as the Video Mode (much like the "Auto Load Header" feature of the program looks for a data file). If that fails, it will look for a file named with the "base" portion of the Video Mode. For example, if no HFD file name is saved in the INI file and the mode is ASVI_GTF3, the Header Utility will look for a HFD file named "ASVI_GTF3.hfd", then look for a file named "ASVI.hfd". If neither of those are found, the Header Utility will use the HFD file named "default.hfd" if it exists, or load default header format values if not. The Header Utility follows the same logic when looking for a header data file to load. In the case of the ASVI_GFT3 mode, it will first look for "ASVI_GFT3.txt", then "ASVI.txt", then "default.txt". If none of these are found, the header data will be all zeros (0).

It is important to remember than the Video Mode and Header Format Definition file determine the size of the header, not the number of words in the header data file. Also, the HFD file for each mode can be tailored for the specific video mode if desired. If it is preferable to have video modes "share" a common HFD file, all that is required is to create a single HFD file using the "base" name of these video modes. For example, all of the ASVI modes (i.e. ASVI_DMT, ASVI_GFT1, etc.) can share a the "ASVI.hfd" file.

To edit an existing header data file:

1. Press the "Header button. The program will load header data for the current Video Mode, and display the data using the correct format for this mode.
2. If you wish to edit a different data file, press File - Open and select the desired file.
3. Make the desired changes, then press the File - Save button. The program will prompt you for confirmation because you are overwriting an existing data file.

To create a new header data file:

1. Press the "Header button. The program will load header data for the current Video Mode, and display the data using the correct format for this mode.
2. If you wish to create a new data file using an existing data file, press File - Open and select the desired file.
3. Make the desired changes, then press the File - Save button. Enter the file name for the new file. The program will prompt you for confirmation if you use an existing file name. Take care not to unintentionally overwrite the standard data files for the video modes, unless that is your intention.

You may display header data using an alternate format. The Header Utility will automatically load the format saved in the INI file, or the format matching the current video mode. If you wish to display the data using another format, press the Format – Open button and select the format file you wish to use to display the data. Note that the currently selected header format will be saved in the INI file on program exit.

You may display alternate data using the same format. The Header Utility will automatically load the data file for the current mode. If you wish to display alternate data, press the File - Open button and select the data file you wish to display.

Comments in the HFD file begin with the "" char and are ignored by the program. All other lines are assumed to be format definitions. The type of definition is identified by first character in the line, as follows:

F header format identifier
N number of words in header
S section definition
W Word definition
U user area definition
X extended data definition

The F definition contains the format identifier string in this format:

F, <format identifier>

The N definition contains the number of words in the header in this format:

N, <number of words in header>

The S definition contains a section identifier and number of words in section in this format:

S, <section identifier>, <number of words in section>

The U definition contains the parameters for an "undefined" area of the format. An undefined area is specified as part of the header but has no word definitions. This definition contains the word offset, the number of words, and the identifier of the undefined area in the following format:

U, <word offset>, <number of words>, <undefined area identifier>

The W definition contains the word index and word name in the following format:

W, <word offset>, <word identifier>, ...

followed by any number of bit field definitions in this format:

<field offset>, <field length>, <field name>, ...

Note that field offset and length of the bit field definition are in bits. There may be up to 8 bit field definitions on a word definition. The 3 field parameters repeat for the number of bit fields in the word. The Word Identifier and Field Identifier may be left blank. The order of appearance of the bit field definitions on the Header Utility screen is determined by the order they are listed in the word definition.

The X definition contains information about the extended header data. This record is included for future use and its contents of this have not yet been determined. It is intended to be used for extended header data which may include color palette or other information.

7.4 Example Header Format Definition (HFD) file

The contents of an example HFD (Header Format Definition) file are shown below:

```
' Header format definition file
'
' This file defines the first 32 words (128 bytes) of header data, and includes
' an "undefined section of an additional 500 words.
'
' format identifier:
f, Default Header Format
'
' number of words in header:
n, 532
'
' FC Frame Header section (6 words, 0 through 5)
s, FC Frame Header
w, 0, FC Frame Header (word 0), 8, 24, D_ID, 0, 8, R_CTL,
w, 1, FC Frame Header (word 1), 8, 24, S_ID, 0, 8, CS_CTL,
w, 2, FC Frame Header (word 2), 8, 24, F_CTL, 0, 8, TYPE,
w, 3, FC Frame Header (word 3), 16, 16, SEQ_CNT, 8, 8, DF_CTL, 0, 8, SEQ_ID,
w, 4, FC Frame Header (word 4), 16, 16, RX_ID, 0, 16, OX_ID,
w, 5, FC Frame Header (word 5), 0, 32, Parameter
'
' Object 0 Header section (22 words, 6 through 27)
s, Object 0 Container Header
w, 6, Object 0 Container Header (word 0), 0, 32,
w, 7, Object 0 Container Header (word 1), 0, 32,
w, 8, Object 0 Container Header (word 2), 0, 32,
w, 9, Object 0 Container Header (word 3), 0, 32,
w, 10, Object 0 Container Header (word 4), 0, 32,
w, 11, Object 0 Container Header (word 5), 0, 32,
w, 12, Object 0 Container Header (word 6), 0, 32,
w, 13, Object 0 Container Header (word 7), 0, 32,
w, 14, Object 0 Container Header (word 8), 0, 32,
w, 15, Object 0 Container Header (word 9), 0, 32,
w, 16, Object 0 Container Header (word 10), 0, 32,
w, 17, Object 0 Container Header (word 11), 0, 32,
w, 18, Object 0 Container Header (word 12), 0, 32,
w, 19, Object 0 Container Header (word 13), 0, 32,
w, 20, Object 0 Container Header (word 14), 0, 32,
w, 21, Object 0 Container Header (word 15), 0, 32,
w, 22, Object 0 Container Header (word 16), 0, 32,
w, 23, Object 0 Container Header (word 17), 0, 32,
w, 24, Object 0 Container Header (word 18), 0, 32,
w, 25, Object 0 Container Header (word 19), 0, 32,
w, 26, Object 0 Container Header (word 20), 0, 32,
w, 27, Object 0 Container Header (word 21), 0, 32,
'
' Object 0 Ancillary Data section (4 words, 28 through 31)
' Ancillary data
s, Object 0 Ancillary Data
w, 28, Object 0 Ancillary Data (word 1), 0, 32,
w, 29, Object 0 Ancillary Data (word 1), 0, 32,
w, 30, Object 0 Ancillary Data (word 2), 0, 32,
w, 31, Object 0 Ancillary Data (word 3), 0, 32,
'
' Undefined area
u, 32, 500, Undefined Area
'
' Extended data (TBD) section (??? words, 32 through ???)
x, Extended Data section
```

8 Standard Firmware - Supported Format Profiles

8.1 XGA Color Progressive

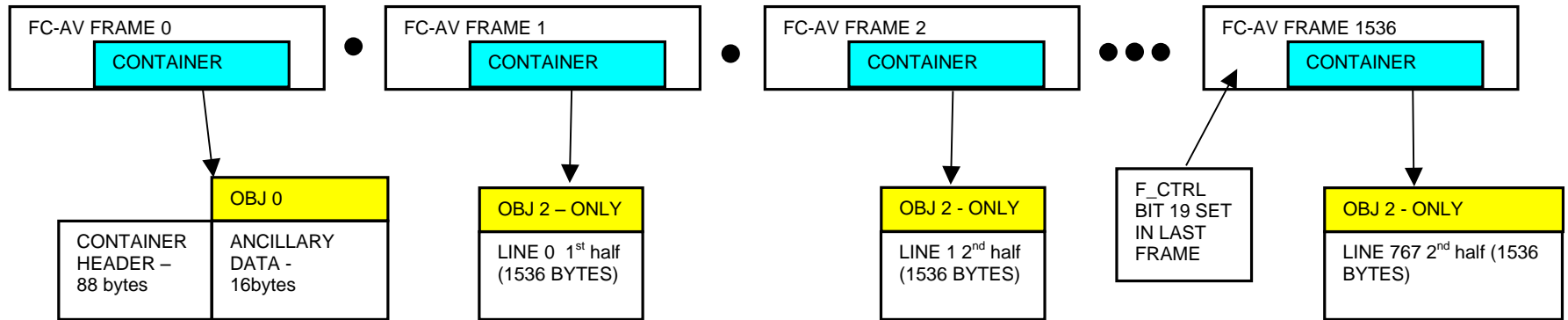


Figure 7.1 - FC Frame Sequence XGA Format

Parameter	Value	Ancillary data value (hex)
Video frame rate	15 to 60Hz	14
Number of rows	768	300
Number of columns	1024	400
Frame or field based	Frame	0
Color Information	RGB	1
Packing Table Number	32-bit	0
Bits Per Sub-pixel	8:8:8:0	8880
Video lines per FC frame	0.5	na
FC frames per Video frame	1536	na

Table 7.1 - Video Format XGA

8.2 SXGA Color Progressive

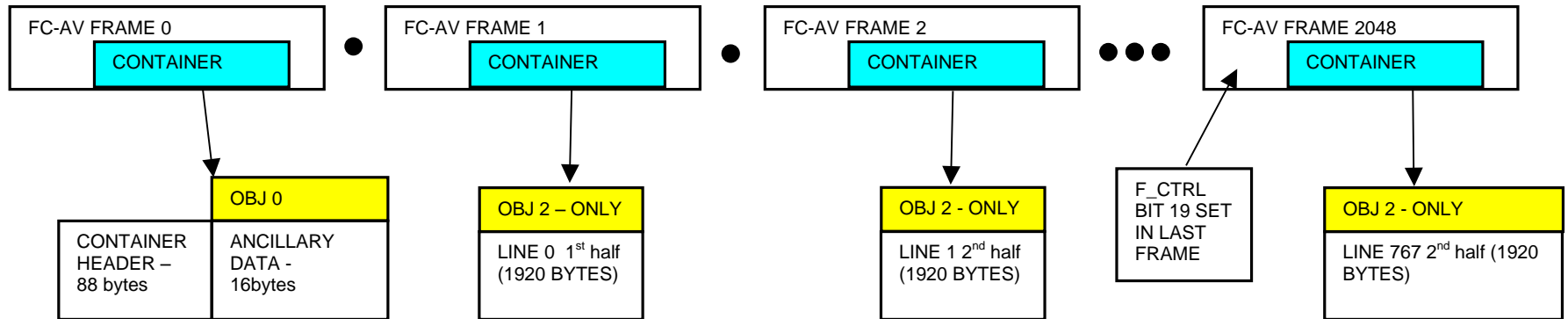


Figure 7.1 - FC Frame Sequence XGA Format

Parameter	Value	Ancillary data value (hex)
Video frame rate	15 to 60Hz	14
Number of rows	1024	300
Number of columns	1280	400
Frame or field based	Frame	0
Color Information	RGB	1
Packing Table Number	32-bit	0
Bits Per Sub-pixel	8:8:8:0	8880
Video lines per FC frame	0.5	na
FC frames per Video frame	2048	na

Table 7.2 - Video Format SXGA

END OF DOCUMENT