



**CMOSIS / AWAIBA**

is now

# Member of the ams Group

The technical content of this CMOSIS / AWAIBA document is still valid.

**Contact information:**

**Headquarters:**

ams AG

Tobelbaderstrasse 30

8141 Premstaetten, Austria

Tel: +43 (0) 3136 500 0

e-Mail: [ams\\_sales@ams.com](mailto:ams_sales@ams.com)

Please visit our website at [www.ams.com](http://www.ams.com)



**CMV12000**

**12 Megapixel machine vision CMOS image sensors**

# **Evaluation System Manual**

**Change record**

<b>Issue</b>	<b>Date</b>	<b>Modification</b>
1	30/08/2012	Origination
2	10/07/2014	Updated for software v2.0
2.1	02/09/2014	Updated for software v2.0.2
2.2	18/03/2015	Updated for software v2.4 Added procedure for removing sensor Removed “confidential” from footer

---

## Table of Contents

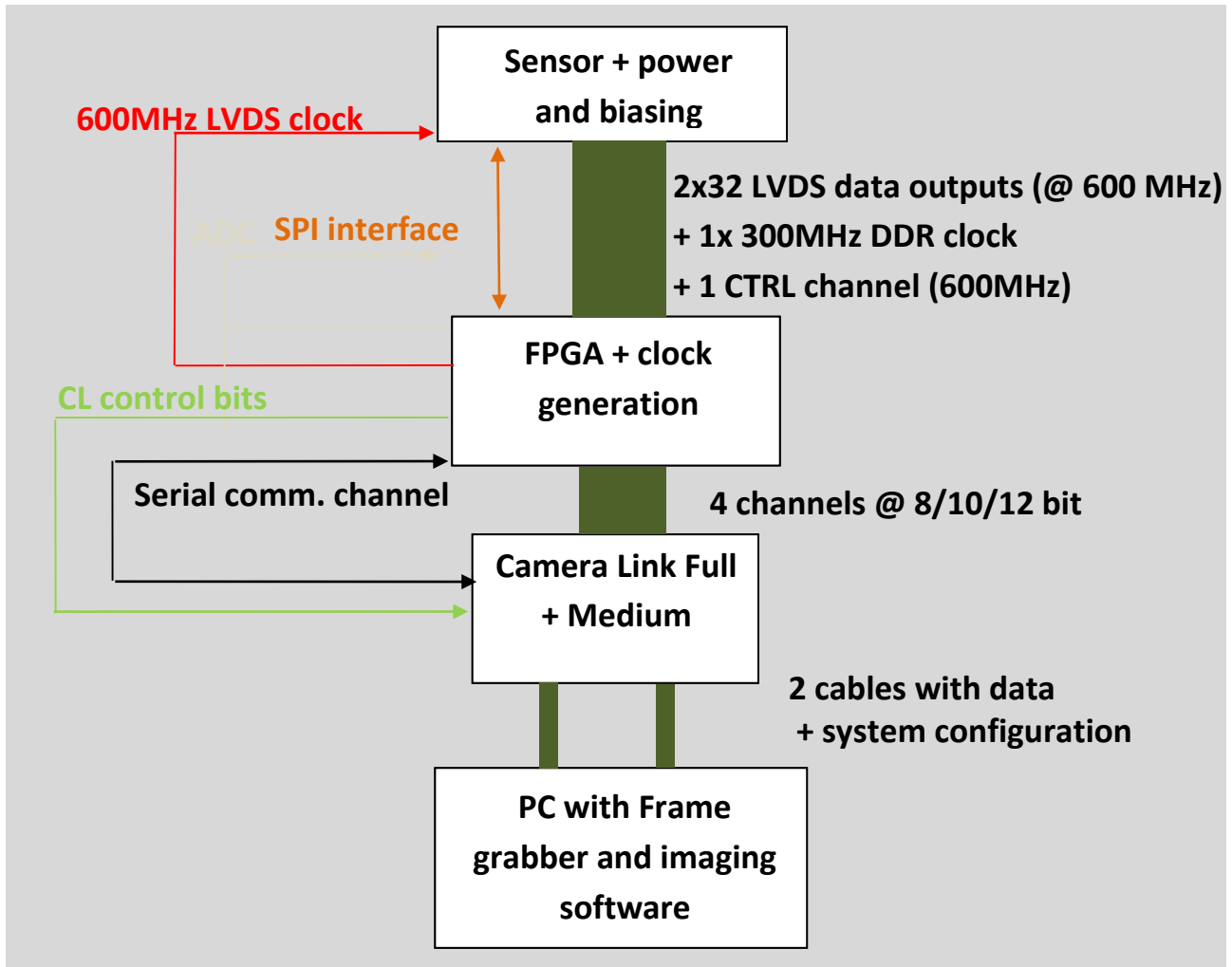
<b>1</b>	<b>System description</b>	<b>4</b>
1.1	Overview	4
1.2	Block diagram	4
1.3	Contents of the test system	4
1.4	CMV12000 board	5
1.4.1	Lens holder	5
1.4.2	Sensor + power and biasing	5
1.5	FPGA	7
1.6	CameraLink Interface	7
1.6.1	Full speed vs. full resolution	8
<b>2</b>	<b>Frame grabber and PC</b>	<b>9</b>
<b>3</b>	<b>Installing the system</b>	<b>10</b>
<b>4</b>	<b>Running the system</b>	<b>11</b>
4.1	Starting the software	11
4.2	Software overview	11
4.3	Software details	12
4.3.1	Control area	12
4.3.2	Register settings area	13
4.3.3	Image area	13
4.3.4	Image statistics area	14
4.3.4.1	Image statistics in ROI mode	16
4.3.5	Clock speed control area	20
4.3.6	Software startup	21
4.3.7	Grabbing multiple images	21
4.3.8	Errors	21

## 1 SYSTEM DESCRIPTION

### 1.1 OVERVIEW

The purpose of this document is to give an overview of the test system which for the 12Mpxl machine vision image sensors produced by CMOSIS. This test system can be used for testing and verifying the operation of the CMV12000 image sensor.

### 1.2 BLOCK DIAGRAM



The block diagram above displays the different parts of the CMV12000 test system. The three upper blocks are implemented on one PCB.

The fourth block is a standard third party frame grabber card which plugs into a PC's PCI Express bus (NI 1433). For this frame grabber and the PCB dedicated software has been developed in a LabVIEW environment.

### 1.3 CONTENTS OF THE TEST SYSTEM

The following components should be present when the CMV12000 test system is delivered.

- CMV12000 board with lens holder and banana-plug power cable
- Nikon F-mount lens AF Nikkor 50mm f/1.8D
- Acrylic bottom and top plate with rubber feet

- 2 CameraLink cables (2m)
- NI PCI1433 frame grabber card (inside PC)
- Lenovo ThinkCentre mini tower model + demo software

## 1.4 CMV12000 BOARD

### 1.4.1 LENS HOLDER

The CMV12000 PCB is equipped with a F-mount lens holder, and extension ring and a Nikon 50mm lens. The extension ring serves to have the correct distance between sensor and lens, so that the focal distance of the lens is correct. Place the lens on the extension ring and twist it to tighten the lens in the ring. The extension ring can be unscrewed from the lens holder by turning counter clock wise. The lens holder itself is tightened on the PSB using four screws.

### 1.4.2 SENSOR + POWER AND BIASING

This part of the test board includes the sensor, its power sources and biasing (decoupling to ground).

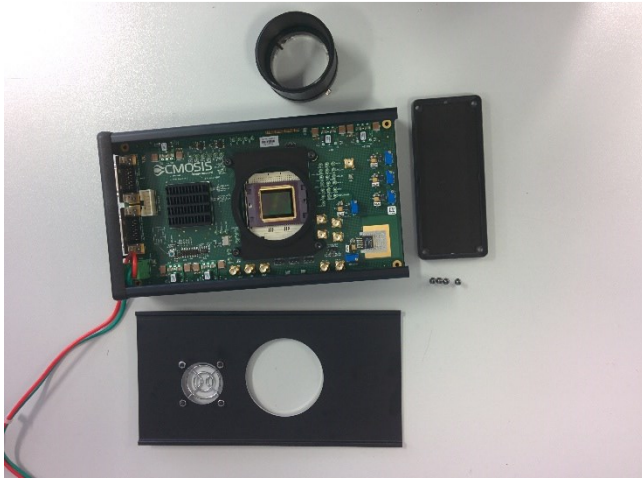
The following procedure should be followed to remove the sensor:



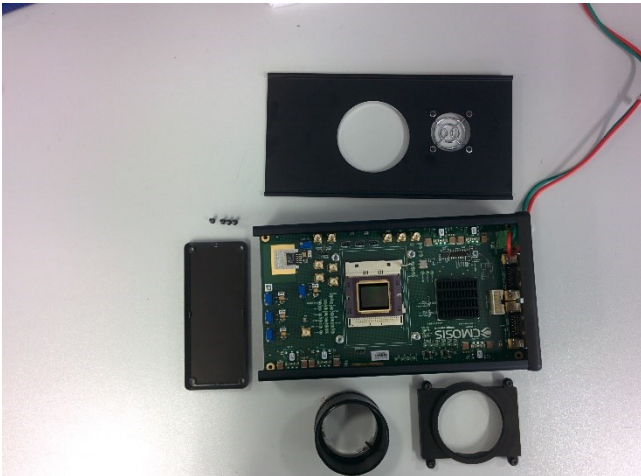
1. Remove the extension from the lens holder



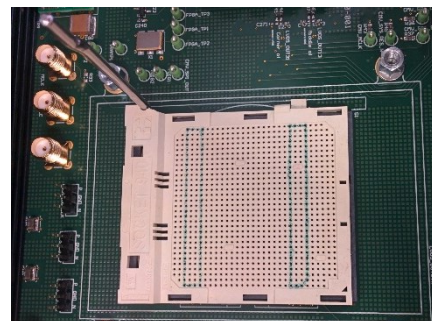
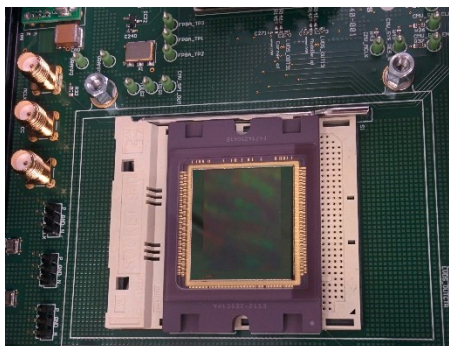
2. Remove the back plate from the demo kit by unscrewing the four screws holding it in place.



3. Remove the top plate by gently pulling it out of the chassis, towards the side of the back plate.



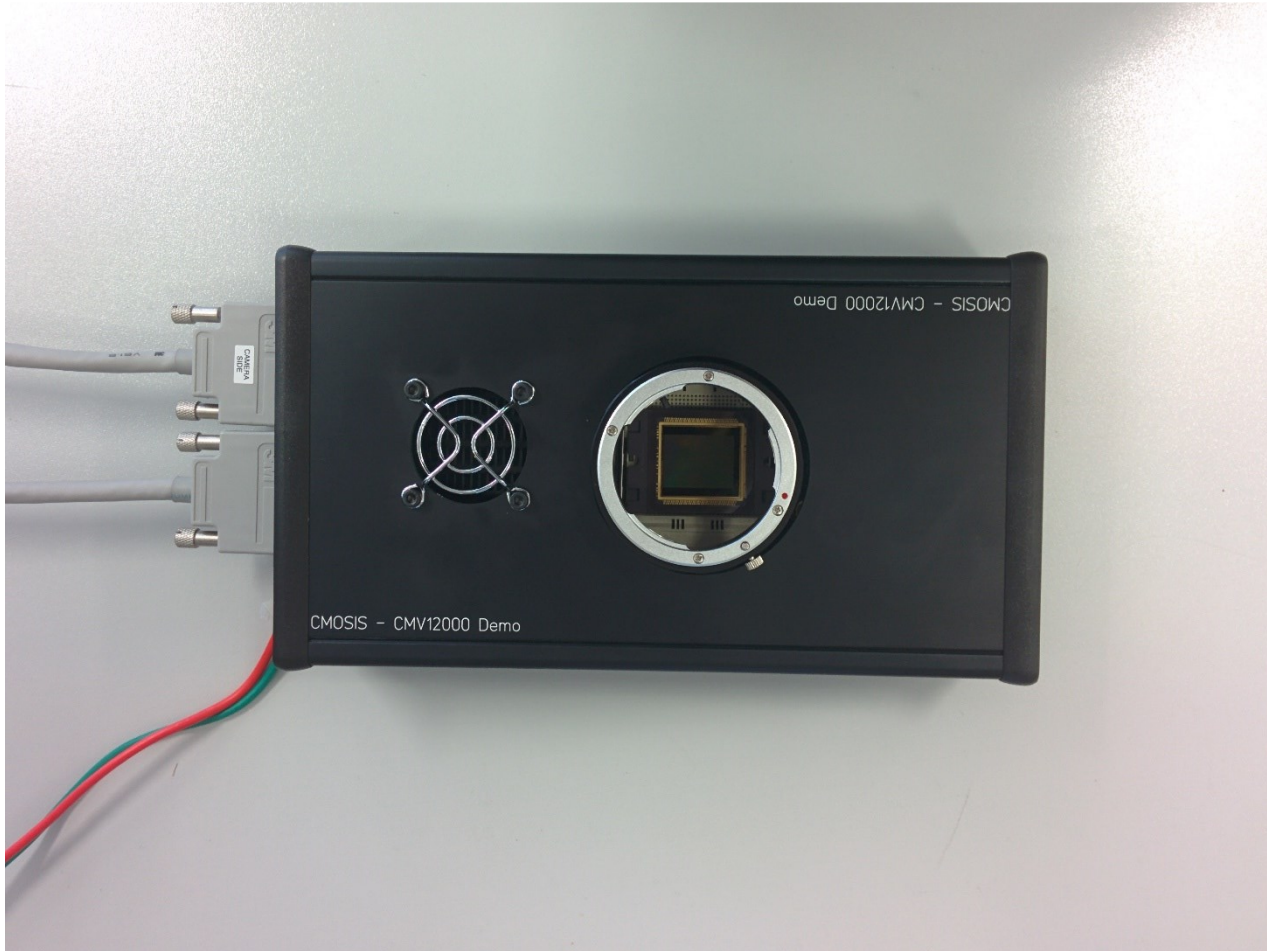
4. Remove the lens holder by removing the four screws holding it into place to access the ZIF socket where the sensor is plugged in. To replace the sensor, simply lift the metal handle upwards 90°; this will open the socket. Remove the sensor and place one back in the correct position. No force is needed to place the sensor in the socket. Now close the socket by pushing the handle down until it is tightened.



It is not recommended to remove the PCB from the chassis, as doing this can cause wear and tear damage on the sides of the PCB.



The picture below shows the entire kit with all the connections. The power supply cables need an input voltage of 14VDC at 2A.



## 1.5 FPGA

The FPGA has 6 important roles in the test system:

1. Accept the LVDS data from the sensor (64 data channels @ 600Mbit/s, 1 LVDS clock channel @ 300MHz DDR and 1 control channel @ 600Mbit/s)
2. Generate the necessary driving signals for the sensor
3. Generate the clocks needed by the sensor and CameraLink interface
4. Communicate with the sensor over the SPI interface
5. Communicate with the PC over the serial interface of the CameraLink
6. Prepare the data for the CameraLink interface
7. Send the data to the CameraLink interface together with the control bits and clocks

Taking the requirements above into account, we have chosen Xilinx Virtex5 FPGA for our test system. All functionality described in the requirements above is programmed into the FPGA. Communication between the PC and FPGA is possible using the serial channel of the CameraLink interface.

## 1.6 CAMERALINK INTERFACE

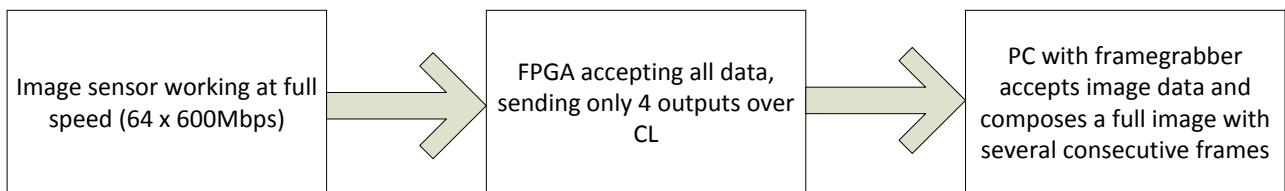
The interface between the test board (camera) and the PC uses the CameraLink standard. This standard is supported by many frame grabber cards on which the raw image data will be dumped. The test board is able to grab images at the 600Mbps. This means that images are grabbed by the system at 600Mbps per LVDS channel.



### 1.6.1 FULL SPEED VS. FULL RESOLUTION

Because of the large image (12Mp) and the CameraLink only supporting 4 channels, a full image is composed of some consecutive frames (#\_of\_frames in the FPGA Registers). For example when a full image is taken when using 2x32 channels output, the fpga will ask the sensor for 16 frames (64/4) shifting the 4 channels each frame. The 16 frames are then stitched together to give a full resolution image. The drawback is that the exposure time for each frame also shifts. So taking images of moving objects or flickering light, will cause image effects. When multiplexing the CMV12000's outputs to 4 channels (2 at each side or 4 at one side), a full image can be made without taking several frames or exposures. Of course the frame rate will not be maximum speed.

If you disable remapping in the software (left bottom in main window) you can see 4 channels at full speed and choose which channels you want to see on the screen but no full image will be built from multiple frames. Also the 4 channels you select are positioned next to each other in the image. So no odd/even top/bottom remapping will be done. The 4 channels can be selected in the FPGA Registers tab by setting Auto mux enable to 0 (off) and use registers "CL Mux 0-4" to select the channels number you want to see.



## 2 FRAME GRABBER AND PC

A frame grabber card which can accept the Medium and Full CameraLink interface was chosen to complete the test system interface. The NI PCIe-1433 allows raw image data to be dumped at 4.8Gbit/s on the PC. This card is fitted in a PCI Express slot of the test system PC and uses two dedicated cables to connect to the CMV12000 board.

The PCIe-1433 frame grabber card comes with an API, which can easily be used in LabVIEW. This API is used to communicate with the frame grabber card and read the image from memory to display it on the screen.

See <http://www.ni.com/pdf/manuals/372015a.pdf> for more details on the PCIe-1433 frame grabber card.

The PC in which the frame grabber card is inserted and on which the software is running has the following minimum requirements:

- Intel Core i3
- 2 Gbyte DDR memory
- 500GB SATA hard disk
- Intel HD Graphics
- 16x DVD-ROM
- PCI-Express slots

On this PC, the LabVIEW RunTime Engine and the Vision Development Module will be installed. A dedicated CMV12000.vi executable is written to grab and save images from the camera board. A detailed description of the software can be found in the next sections.

### 3 INSTALLING THE SYSTEM

When installing the system, the following connections have to be made:

1. Connect the PC to its peripherals (keyboard, mouse and monitor)
2. Connect the PCB to a power supply (hook up +14VDC and GND)
3. Connect the PCB to the PC via the two camera link cables. Use either cable to connect the camera link connector on the PCB labeled BASE to the PC camera link port labeled CAMERA LINK PORT 0. Use the other cable to connect PCB port MEDIUM to the other frame grabber port on the PC. Next two figures show these connections.



Note:

The FPGA configuration is stored in an EEPROM on the PCB. The FPGA is configured automatically when the power is switched on.

## 4 RUNNING THE SYSTEM

### 4.1 STARTING THE SOFTWARE

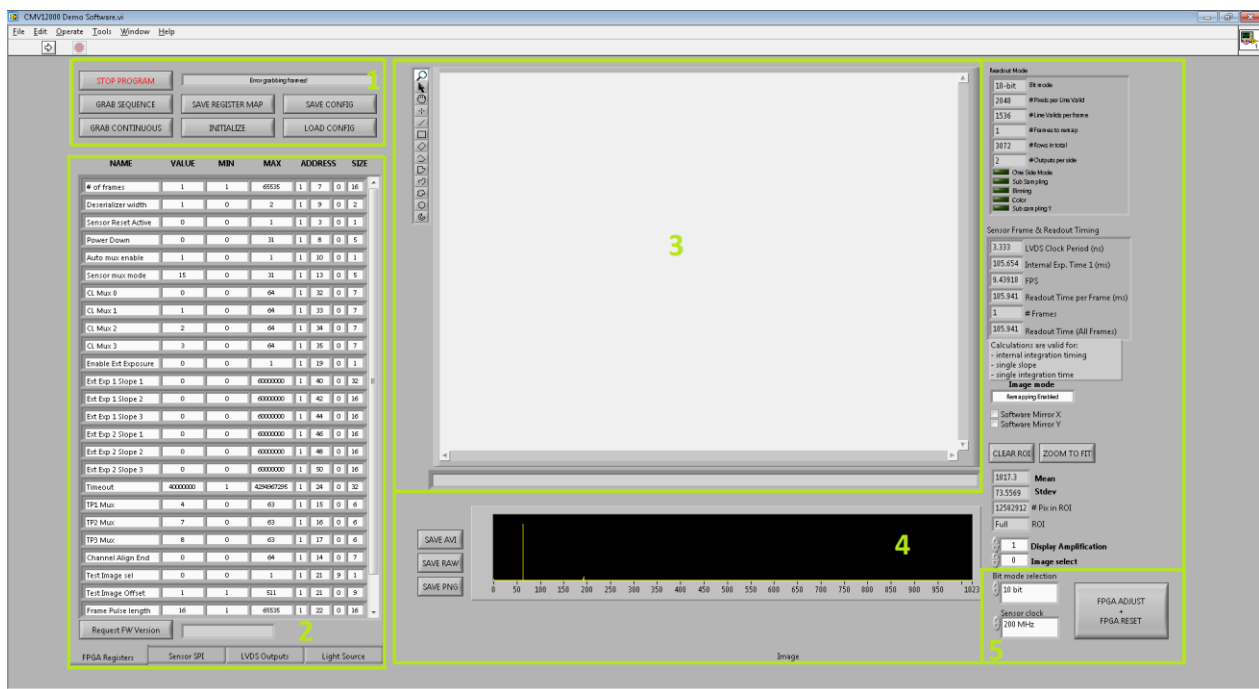
Power on the power supply and wait for the green LED to light up (FPGA is programmed) before starting the software.

Log on to the PC using the 'demo user' account. No password is needed.

On the Desktop, there is a link to the demo software executable, called "CMOSIS CMV12000 Demo Software v2.0.exe". Double clicking this shortcut will launch the demo software.

### 4.2 SOFTWARE OVERVIEW

Next figure shows a screenshot of the demo software.

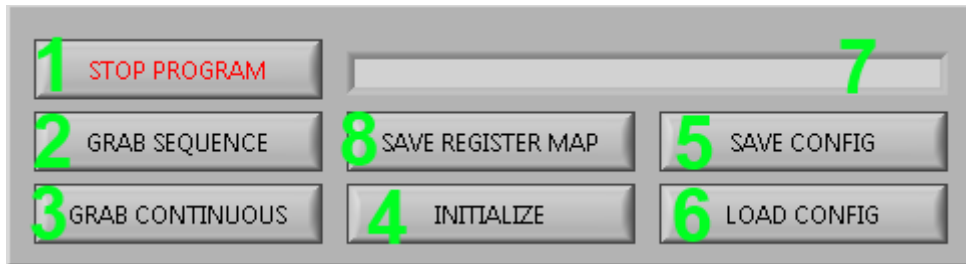


The four main areas of the software are:

1. **Control area:** The user can control the software (grab frames, save and load configuration files ...) via a number of push buttons.
2. **Register settings area:** The user can set the registers of the sensor. The registers are grouped into a number of tabs for better readability.
3. **Image area:** The images acquired from the sensor are displayed in this image viewing plane.
4. **Image statistics area:** A number of image and software statistics are displayed here.
5. **Clock speed control:** Here you can change the LVDS input clock speed.

## 4.3 SOFTWARE DETAILS

### 4.3.1 CONTROL AREA



The 9 controls in the Control Area are:

1. **STOP PROGRAM** button: Click this button to stop the application from running and exit to the labview run-time environment. Then the window can be closed. It is required to use this button to stop the program, instead of closing the window while it is still running.
2. **GRAB SEQUENCE** button: Click this button to grab a single sequence of frames. The number of frames in the sequence is set in the 'register settings area' of both sensor and fpga (needed for possible composing of a full resolution image with multiple frames). When the single grab has finished, the 'image 'area' will be updated.
3. **GRAB CONTINUOUS** button: This button has the same function as the 'grab single' button. The difference is that this button has a different latching action. By clicking it once, it will be enabled. As long as it is enabled, the software will continuously request sequences of frames and update the image area. This mode is also called 'video mode'. To stop the video mode, click the 'grab continuous button' again. This will reset the button to its disabled state.
4. **INITIALIZE** button: Click this after you have loaded a config file or after some strange behavior of the program. This will reload the fpga with the register settings in the register settings area (which are set by a config file), reset the sensor, upload the registers to the sensor and do an LVDS training.
5. **SAVE CONFIG** button: By clicking this button, a configuration file can be saved. A configuration file is a human-readable version of the sensor register map. The saved text file has the same information as the register map that is saved using the 'save regmap button'. The advantage of a configuration file is that it can be reloaded into the software using the 'load config button'. (Tip: save the configuration files with a .txt extension).
6. **LOAD CONFIG** button: By clicking this button, a configuration file can be loaded into the software. When this is done, all settings from the 'register settings area' are overwritten with the information from the configuration file. The new settings are automatically transferred to FPGA and sensor (no need to click the 'reload FPGA button' after loading a configuration file). (Tip: it is allowed to load a configuration file while in video mode. The new settings will reflect in the images immediately.)
7. **MESSAGE**: This text box contains information about what the sensor is currently doing or if an error occurs during image grabbing, it is reported in this text box.
8. **SAVE REGISTER MAP**: This will save the registers as a simple address – value text file.

Note: On the Desktop of the evaluation PC, there is an example of a configuration file. This file has been saved using the 'save config button'. You can load this configuration file into the software with the 'load config button'. This configuration file contains the advised values for optimal sensor operation.

### 4.3.2 REGISTER SETTINGS AREA

The 'register settings area' can be used to change all sensor and fpga registers. All registers in the sensor can be changed or readout from here. Most of these settings can be left unchanged by the user of the evaluation system.

Every register setting in this list refers to a sensor register (see data sheet for complete register map). The user of the evaluation system can only change the 'value' of the registers. The other entries are informational. (Tip: the register addresses (third column from the right) match the address values in the register map in the datasheet).

Upon hitting the enter key after changing a value of a register, it is uploaded to the FPGA; pressing the up/down arrow key will increase/decrease the value and immediately upload it to the sensor/fpga. Note that it is allowed to change register settings while in video mode. The changes will reflect immediately in the displayed images.

NAME	VALUE	MIN	MAX	ADDRESS	SIZE
# of frames	16	1	65535	1 7 0	16
Deserializer width	1	0	2	1 9 0	2
Sensor Reset Active	0	0	1	1 3 0	1
Power Down	0	0	31	1 8 0	5
Auto mux enable	1	0	1	1 10 0	1
Sensor mux mode	0	0	31	1 13 0	5
CL Mux 0	0	0	64	1 32 0	7
CL Mux 1	1	0	64	1 33 0	7
CL Mux 2	2	0	64	1 34 0	7
CL Mux 3	3	0	64	1 35 0	7
Enable Ext Exposure	0	0	1	1 19 0	1
Ext Exp 1 Slope 1	0	0	60000000	1 40 0	32
Ext Exp 1 Slope 2	0	0	60000000	1 42 0	16
Ext Exp 1 Slope 3	0	0	60000000	1 44 0	16
Ext Exp 2 Slope 1	0	0	60000000	1 46 0	16
Ext Exp 2 Slope 2	0	0	60000000	1 48 0	16
Ext Exp 2 Slope 3	0	0	60000000	1 50 0	16
Timeout	40000000	1	4294967295	1 24 0	32
TP1 Mux	0	0	63	1 15 0	6
TP2 Mux	0	0	63	1 16 0	6
TP3 Mux	0	0	63	1 17 0	6
Channel Align End	0	0	64	1 14 0	7

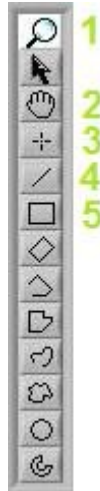
  

NAME	VALUE	MIN	MAX	ADDRESS	SIZE
y_size_28	0	0	3072	2 62 0	16
y_size_29	0	0	3072	2 63 0	16
y_size_30	0	0	3072	2 64 0	16
y_size_31	0	0	3072	2 65 0	16
yoffset	0	0	65535	2 66 0	16
ystep	1	1	3072	2 67 0	16
mono	1	0	1	2 68 0	1
sub	0	0	1	2 68 1	1
bin	0	0	1	2 68 2	1
mono_inte	1	0	1	2 68 3	1
mirror_x	0	0	1	2 69 0	1
mirror_y	0	0	1	2 69 1	1
inte_ext	0	0	1	2 70 0	1
inte_dual	0	0	1	2 70 1	1
inte_short	0	0	1	2 70 2	1
inte_1	1536	1	16777215	2 71 0	24
inte_2	1536	1	16777215	2 73 0	24
inte_s2	0	0	16777215	2 75 0	24

### 4.3.3 IMAGE AREA

The images that are acquired from the sensor are displayed in the 'image area'. Only one image is displayed. If a sequence of frames is acquired, the entire burst is in the memory. A selection switch in the 'image statistics area' can be used to select which frame from the sequence is displayed. With right-clicking on the image you can save the image as is shown on the screen. This will not save the image as it comes out of the sensor (see next section).

Next to the image plane is a small toolbar with image controls. These can be used to zoom in the image and to select region of interest (ROI) contours for the image statistics. Next figure shows this toolbar.



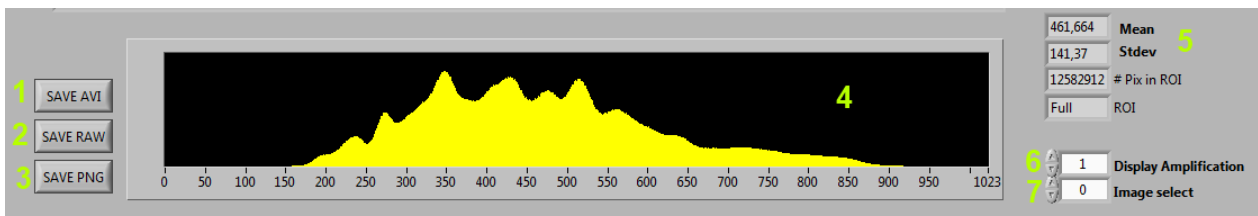
Of these tools, the following 5 are most commonly used:

1. **ZOOM:** When selected, the mouse pointer turns into a magnifying glass when moved over the image. Every click will zoom into the image by 1X. When holding the SHIFT key on the keyboard, the mouse pointer will turn into a magnifying glass with a small 'minus sign' in it. Clicking in the image with the SHIFT key held will zoom the image out by 1X.
2. **PAN:** When the image is zoomed, this function allows the user to pan the image.
3. **POINT ROI:** Allows the user to set an ROI in the image of exactly 1 pixel. Click on the image to place the point ROI.
4. **LINE ROI:** Allows the user to set a line ROI in the image. Click on the image to set the first point of the line. Hold the left mouse key and drag to where the end point of the line should be. Release the left mouse button to place the line.
5. **RECTANGLE ROI:** Allows the user to set a rectangular ROI in the image. Click on the image to set the first corner of the rectangle. Hold the left mouse key and drag to the point where the opposite corner of the rectangle should come. Release the left mouse button to place the rectangle.

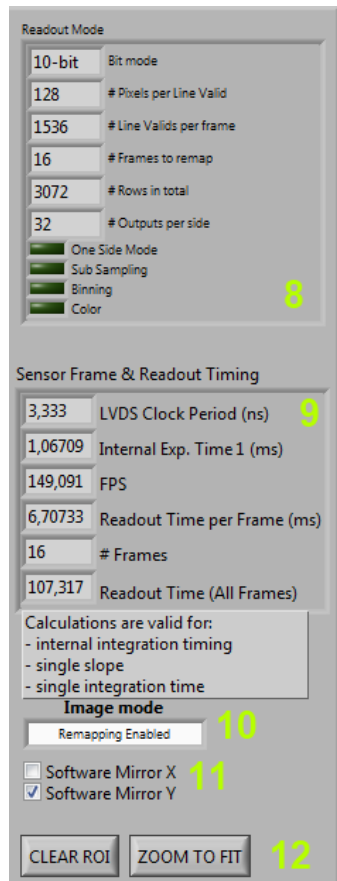
More information on the ROI statistics can be found in the 'image statistics area' section.

#### 4.3.4 IMAGE STATISTICS AREA

Next figure shows the image statistics area.







The 'image statistics area' consists of:

1. **SAVE AVI** button: If a sequence of images is acquired, the entire sequence can be saved as a movie with the avi format. Click the button and select the path and file name for the movie (including the .avi extension).
2. **SAVE RAW** button: Clicking this button will save the requested images in a raw file format. Set the path and name and the software will automatically add the frame number per image.
3. **SAVE PNG** button: Clicking it will save the images in a lossless PNG file format.
4. **HISTOGRAM**: Shows the 10-bit histogram of the image that is displayed in the 'image area'. (Note: see next section for explanation of ROI image statistics)
5. **IMAGE INFORMATION**: Show the average and standard deviation of the selected image in 10-bit values. (Note: see next section for explanation of ROI image statistics). Also the size of the ROI is shown.
6. **IMAGE AMPLIFICATION**: This will digitally amplify the image on screen. This will only be showed in the image on screen and not in the images when clicking the SAVE buttons. Right-clicking on the image will give you the option to save the image as it is shown on screen.
7. **IMAGE SELECTION** control: If a sequence of more than one image is grabbed from the sensor, this control is used to select which of the images is displayed in the 'image area'. The index number '0' identifies the first image of the sequence. If the selection index is larger than the number of frames-1, the 'image area' shows no image.
8. **READOUT MODE**: Shows information about the mode at which the sensor is set.
9. **SENSOR FRAME & READOUT TIMING** shows numerous timings of the sensor. These values are only valid for internal integration mode, single slope and single integration time.
10. **IMAGE MODE**: When Remapping Enabled is selected, the software will remap the data coming from the sensor outputs to get an full resolution correct image, depending on the mode which is selected. The fpga register "# of frames" will have to bet set to the correct value to compose this full resolution image. This value should at least be the numbers of sensor outputs used divided by 4. When you select Remapping disabled. You will only get image data from four sensor output channels on screen

(but at full sensor speed). In the FPGA register settings you can choose which channels you want to view: set the “Auto mux enable” fpga register to 0 and change the values of the “CL MUX 0/1/2/3” to the channel value you want.

Auto mux enable	1	0	1	1	10	0	1
Sensor mux mode	0	0	31	1	13	0	5
CL Mux 0	0	0	64	1	32	0	7
CL Mux 1	1	0	64	1	33	0	7
CL Mux 2	2	0	64	1	34	0	7
CL Mux 3	3	0	64	1	35	0	7

NAME	VALUE	MIN	MAX	ADDRESS	SIZE
# of frames	16	1	65535	1 7 0	16

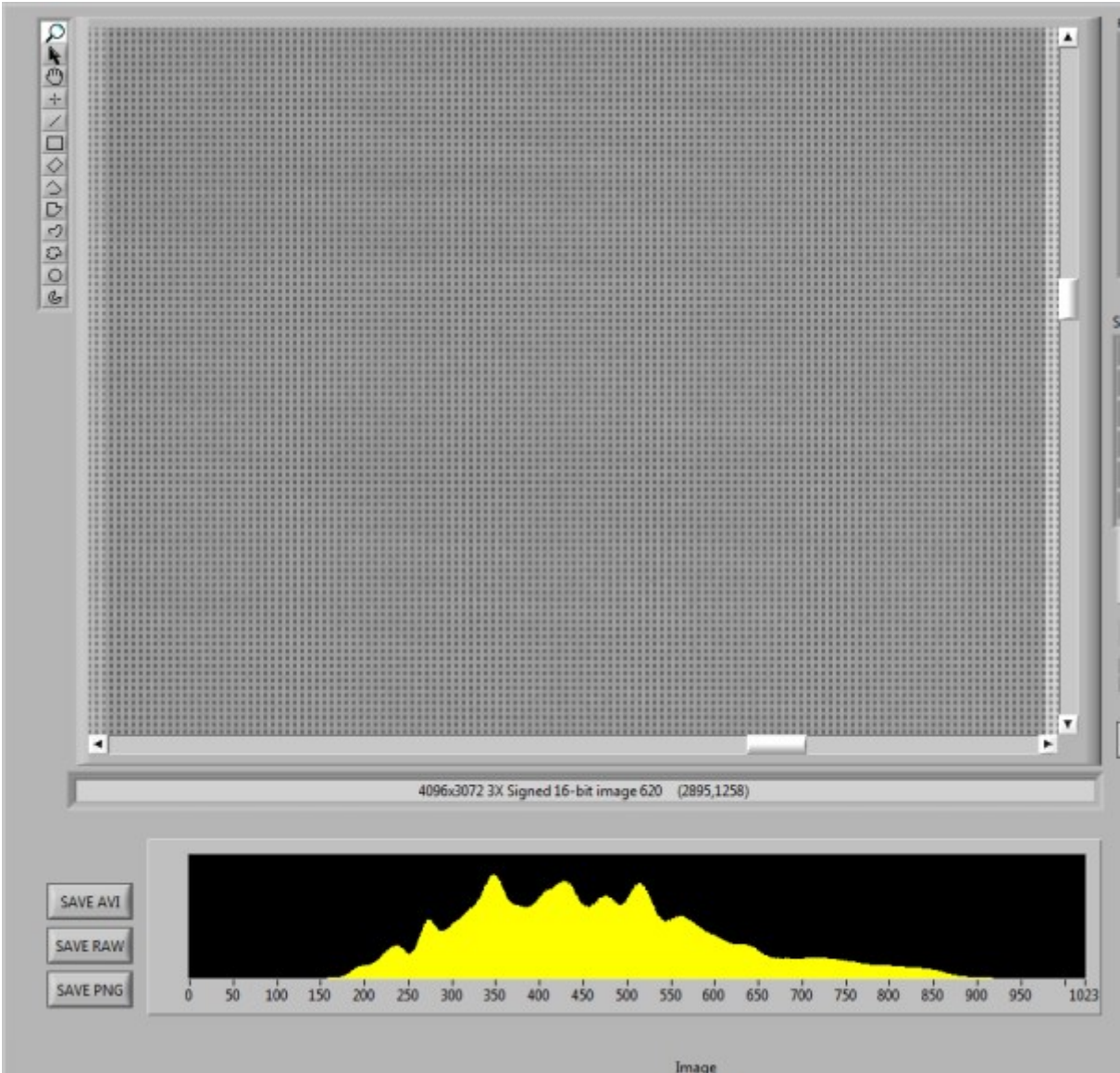
11. **MIRROR:** This will mirror the image on screen in the X- or Y-axis. This is not saved when clicking the SAVE buttons.
12. **ZOOM TO FIT & CLEAR ROI:** Clicking these buttons will zoom so the image on screen fits the window or clear the selected ROI.

#### 4.3.4.1 IMAGE STATISTICS IN ROI MODE

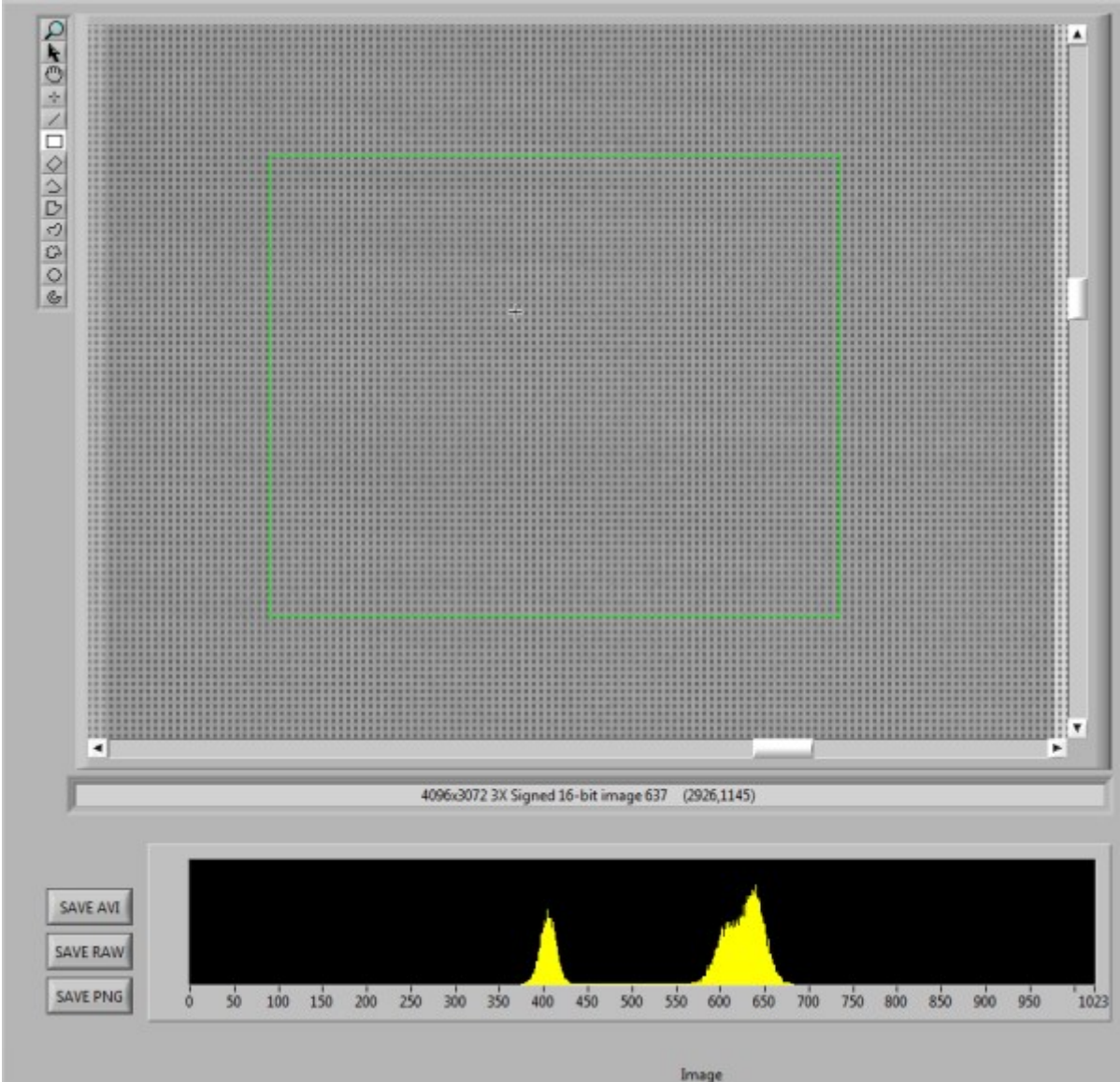
By default, the image statistics apply to the entire image. If statistics of certain areas in the image are desired, the ROI functionality can be used. If an ROI is placed on the image, all statistics that are displayed (histogram, average and standard deviation) are of all pixels that are defined by the ROI.

Next figures show examples of the ROI results for different ROIs on the same image. In the examples, note histogram, the average and standard deviation values and the ‘ROI # pix’ value.

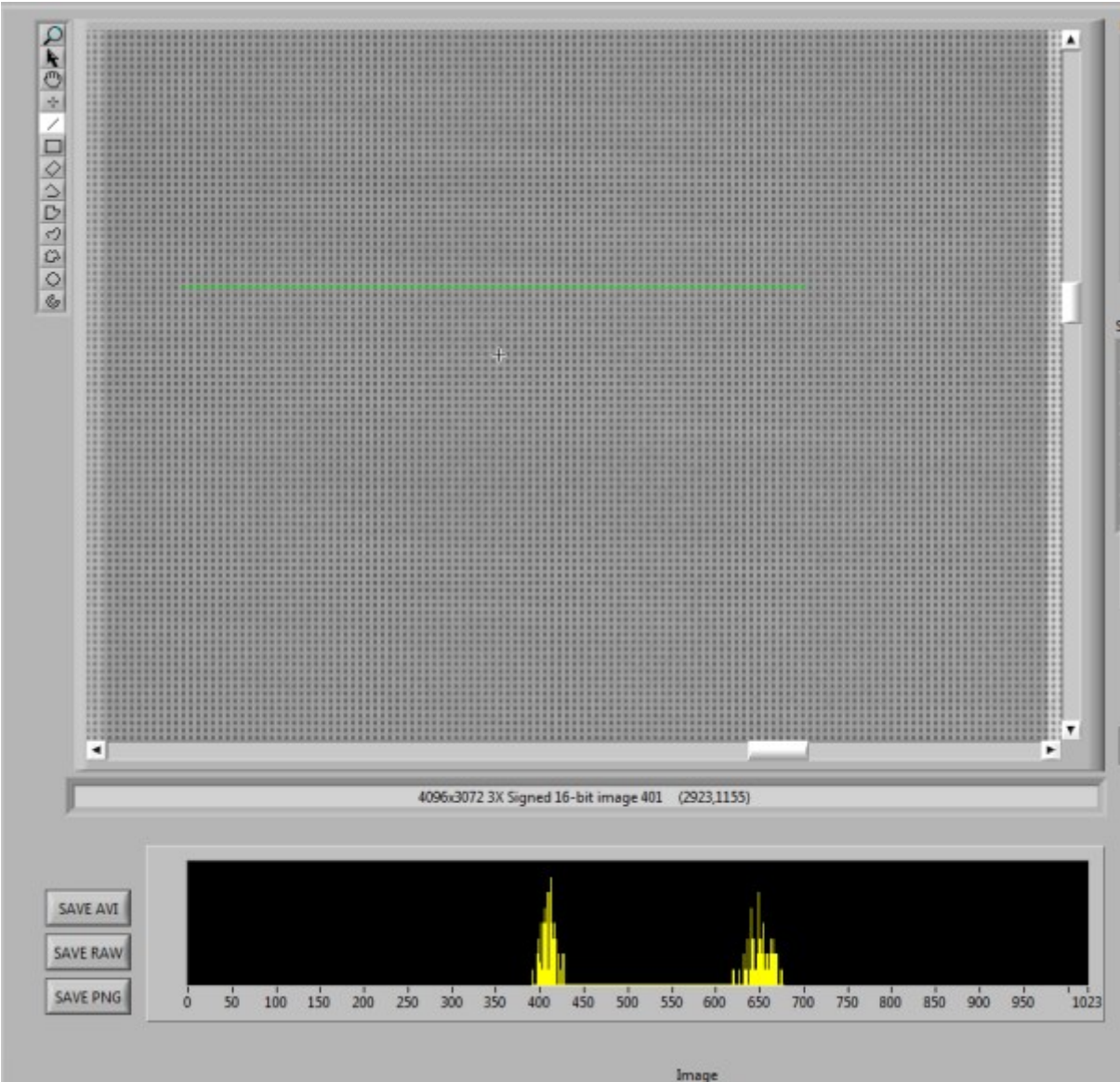
**No ROI:**



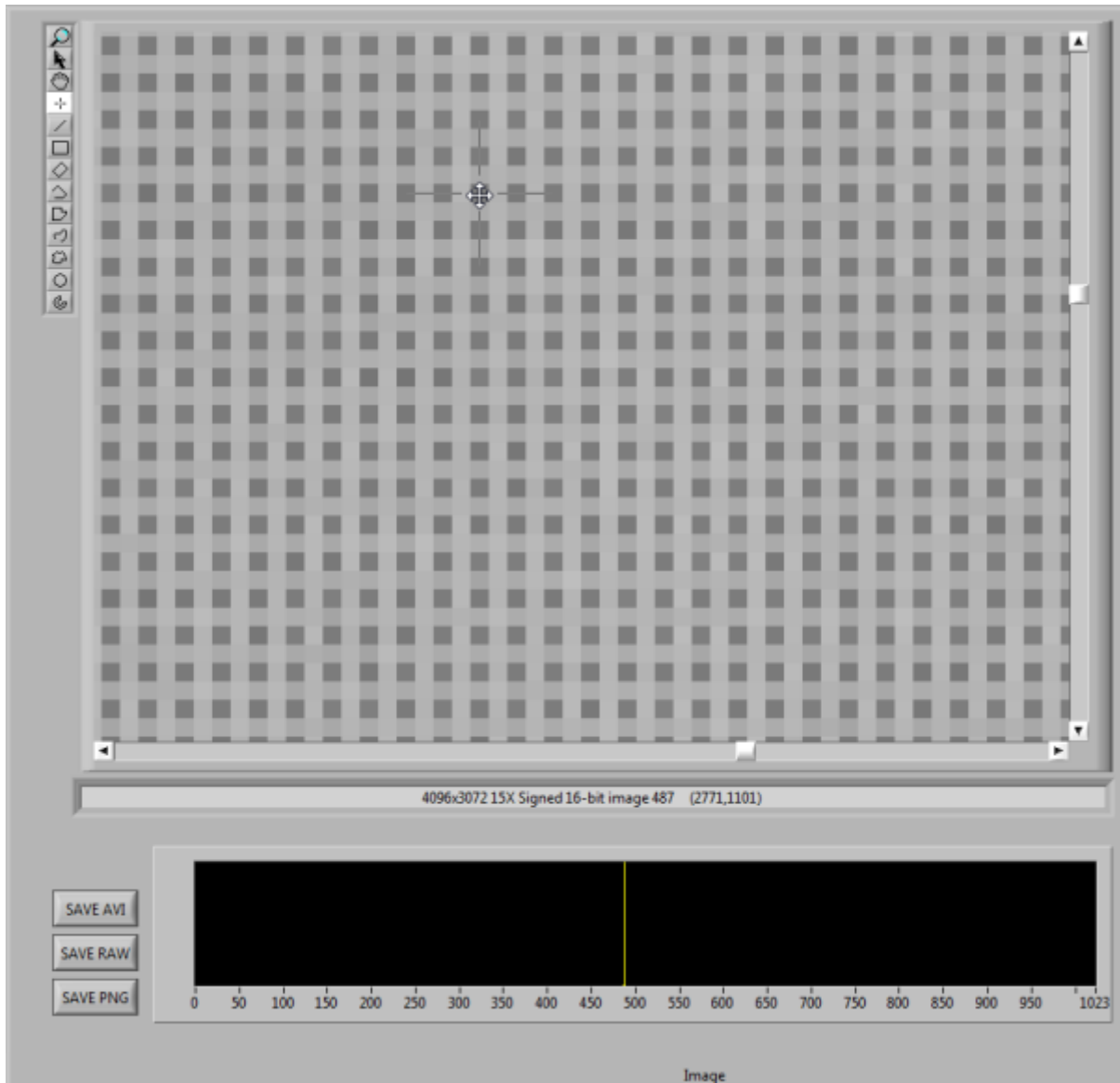
**Rectangle ROI:**



Line ROI:

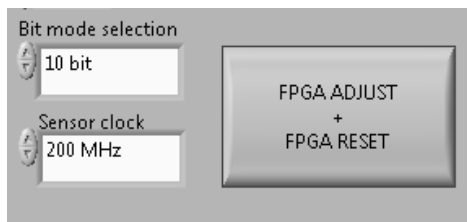


**Point ROI:**



**4.3.5 CLOCK SPEED CONTROL AREA**

Here you can change the LVDS input clock speed. Don't change this while grabbing frames. Select a clock speed from the list. Then click the FPGA ADJUST + FPGA RESET button and afterwards the INITIALIZE button in the control area for the changes to take effect. Remember that some settings will have to be changed for the sensor to work optimally with a certain clock speed (especially ADC\_range)!





#### 4.3.6 SOFTWARE STARTUP

After the software is loaded, load a configuration file and click the INITIALIZE button, to initialize the sensor. This does a sensor reset, uploads the registers currently filled in the register list and does LVDS training. If that is OK, you can start grabbing a frame by clicking GRAB SEQUENCE or GRAB CONTINUOUS. If you see an image, the system is working fine.

#### 4.3.7 GRABBING MULTIPLE IMAGES

If you want to take multiple sequential frames, you will have to change the FPGA #\_of\_frames register. For example, if you are running at 2x32channels, you need 16 frames for one full resolution image. So if you want 2 sequential frames, you will need to change this register to 32.

If you disable the remapping, the number of frames you fill in in this register is the actual frames it will take.

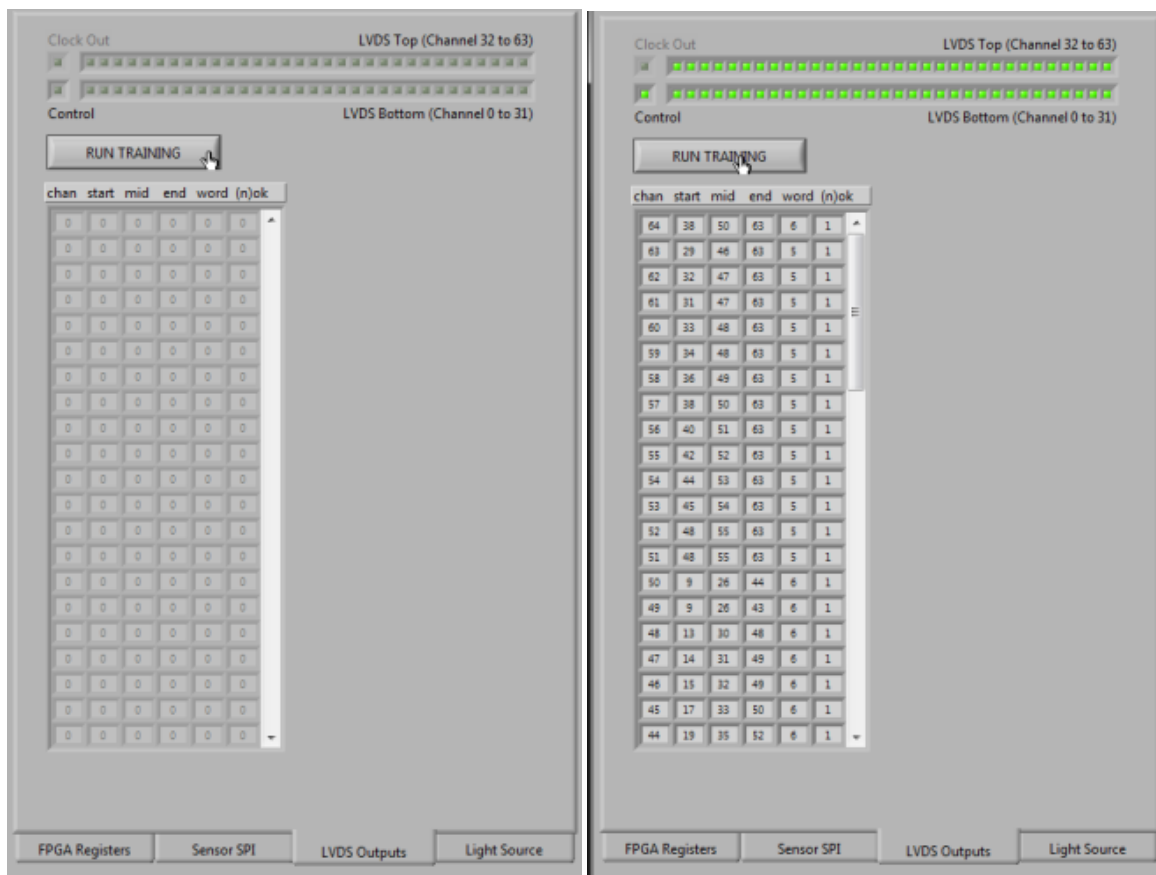
If you set the Sensor SPI frames register to 1 (default), sequential frames will be taken in non-pipeline mode (so no exposure overlap during read-out). If you set this register to 2, the frames will be read-out in pipeline mode. The number of frames in the Sensor SPI frames register does not have to be equal to the #\_of\_frames register of the FPGA.

#### 4.3.8 ERRORS

When something goes wrong during frame grabbing you will get an Error message. The LABView program stops its running state. You will have to click on the black arrow on the left top side of the window to start the program again and enter its running state. Also closing the application and restarting the executable will work.

Try to fix the error by resetting the sensor (to the default register values as stated in the datasheet or given config file), initialize and then read-out the registers to see if there are any conflicts (=red light) in the registers.

Then look at the LVDS training on the FPGA Registers tab, to see if the FPGA can train OK on the sensor data (all lights on the tab should be green).





If everything fails, close the application, power off the power supply, check the CameraLink cables. Power on the supply, wait for the green PCB LED to come on and start the executable again.

In normal operation, when idle, the board will draw about the following currents. If you see a big difference, there is probably hardware fault or the sensor isn't socketed correctly.

<b>Supply Voltage</b>	<b>Supply Current at 300MHz</b>	<b>Supply current at 600MHz</b>
14V	570mA	620mA