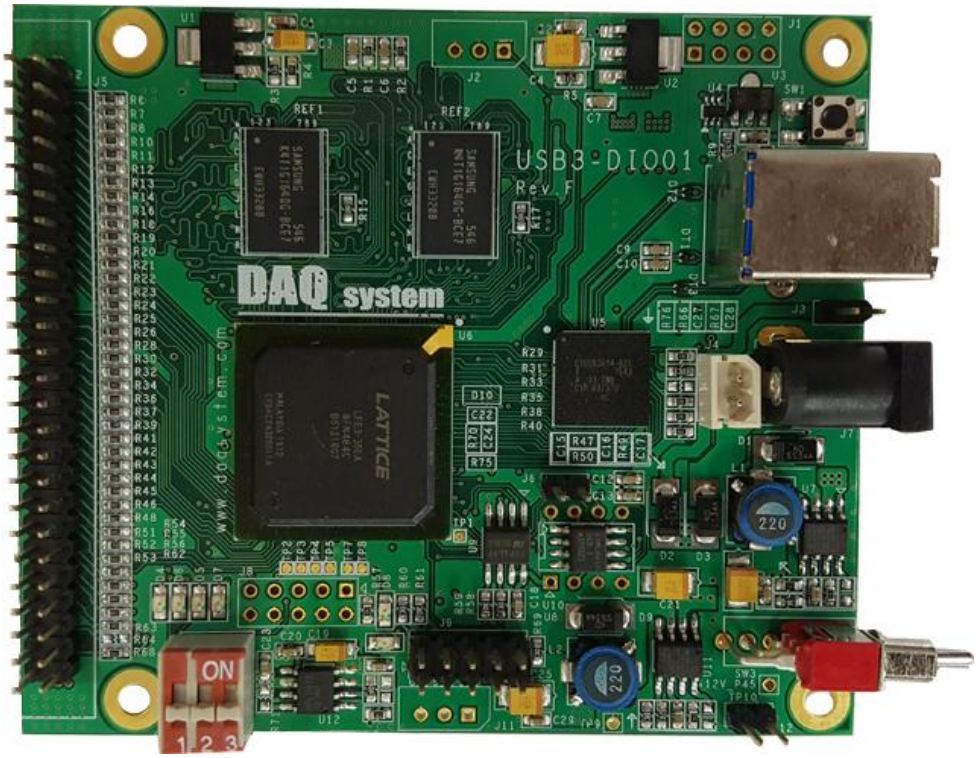
USB3-DIO01

User's Manual



Windows, **Windows2000**, **Windows NT** and **Windows XP** are trademarks of **Microsoft**. We acknowledge that the trademarks or service names of all other organizations mentioned in this document as their own property.

Information furnished by DAQ system is believed to be accurate and reliable. However, no responsibility is assumed by DAQ system for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or copyrights of DAQ system.

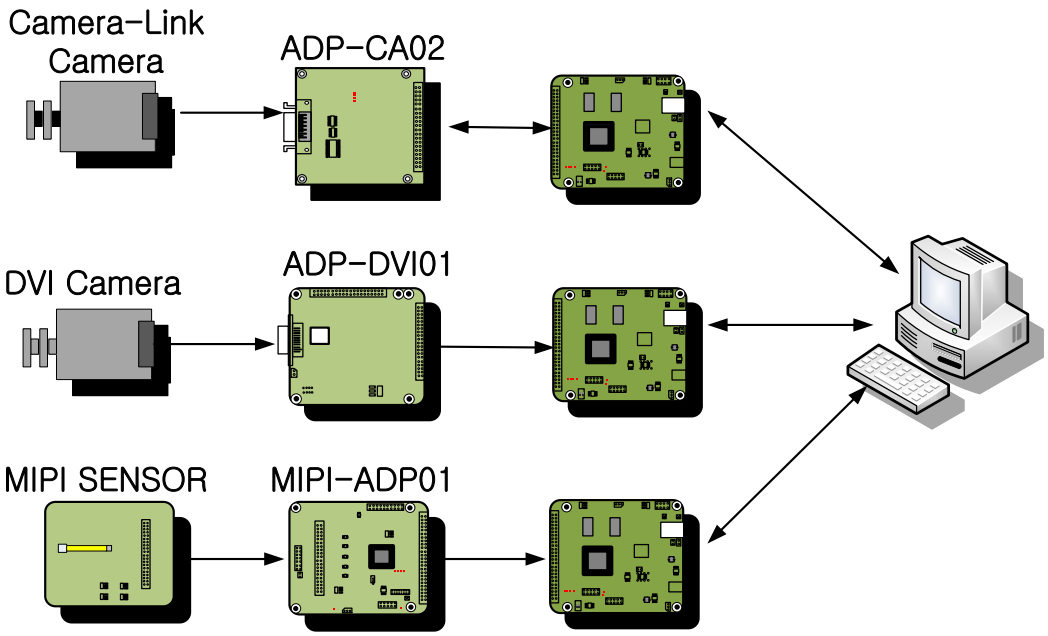
The information in this document is subject to change without notice and no part of this document may be copied or reproduced without the prior written consent.

-- Contents –

1. Introduction

The USB3-DIO01 is a base board type to connect daughter boards. The USB3-DIO01 board receives the image data from USB3 cable and transmits the received data to the ADP-CA01.

A [Figure 1-1] shows Input connection of USB3-DIO01 board. The USB3-DIO01 is received the image data from the Camera Link or DVI camera or MIPI sensor, and transmit the received data to the PC through the USB3 Interface. The ADP-CA01 has a function to exchange the image data for the Camera Link signals. You can be used to connect our DAQ system's daughter board.



[Figure 1-1. USB3-DIO01 Board Usage]

[USB3-DIO01 Specification]

- USB 3.0 Specification (10x faster than USB2.0)
- 128MByte DDR2 x2
- Power by USB-B type connecter or Single 5V operation
- Fully Plug-and-Play
- Use with Daughter Board : USB3-CA01 (Camera-Link Output)
- O.S : Windows 2000/XP/7
- Convenient Windows Application Programming Interface(DLL)

[Application]

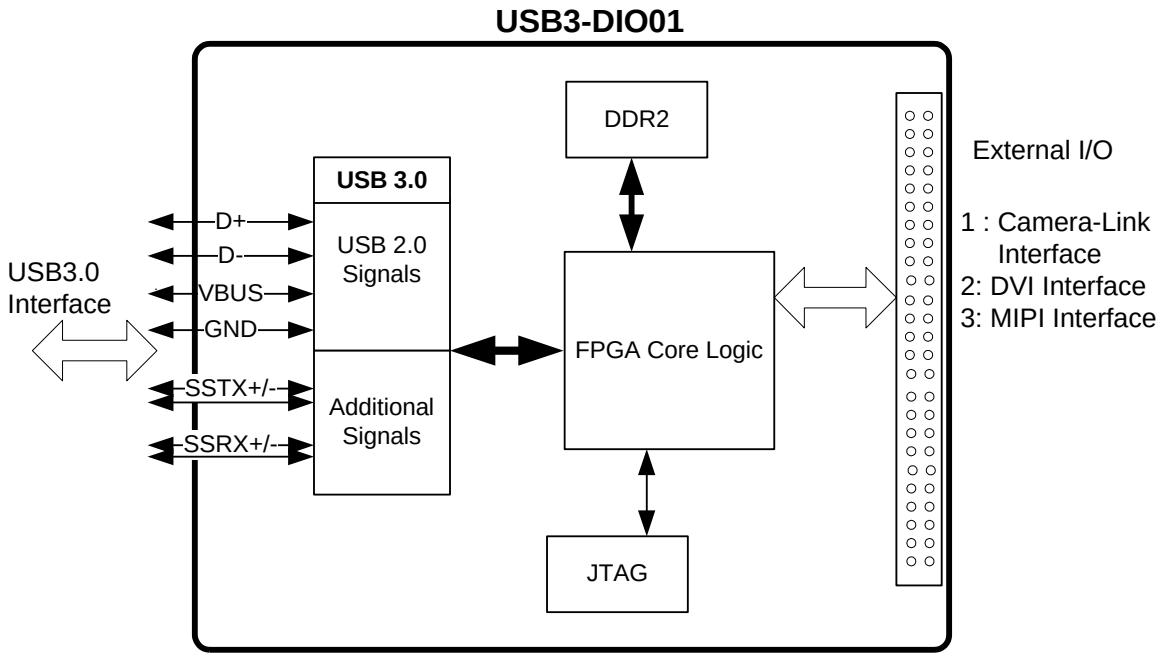
- Frame Grabber

2. USB3-DIO01 Function

2.1 Board Block Diagram

As shown in the following figure, main control of the board is performed FPGA Core Logic. The primary functions are receiving the image frame data through external I/O c onnecter or transmitting it to Camera Link connecter. These functions are performed by u sing API through USB 3.0 Interface at the PC.

External I/O connector can connect the various daughter board through 2x25 2.54Pitch Male Header. Please refer to section 3.3.4.

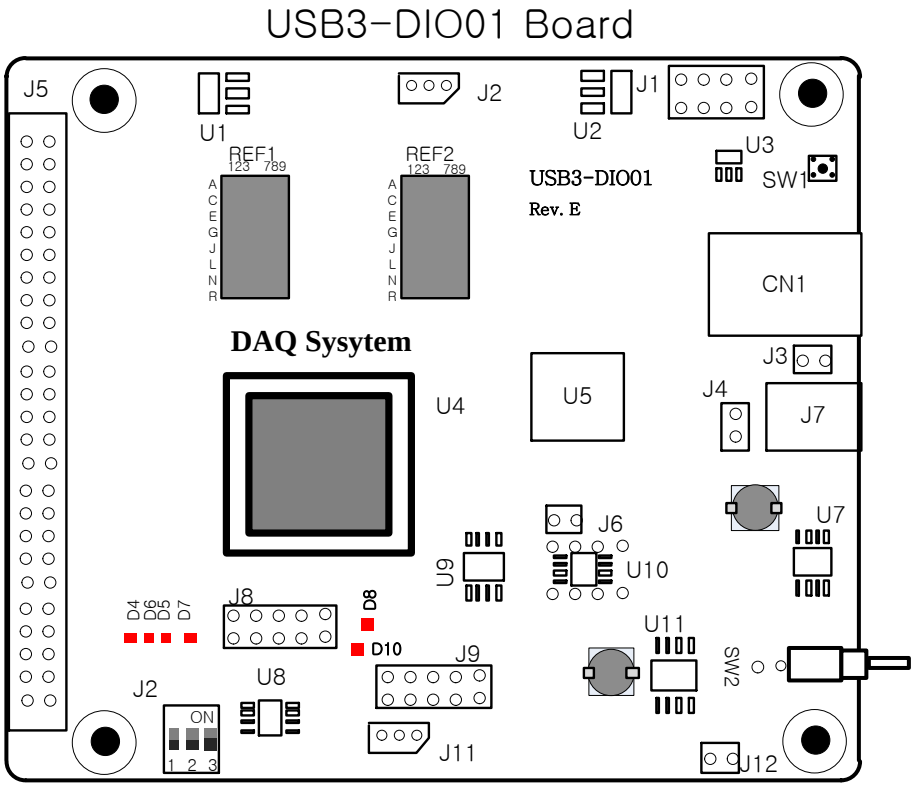


[Figure 2-1. USB3-FRM01 Functional Block Diagram]

Programming FPGA Core Logic is performed via the JTAG interface. The logic program of the FPGA is saved in a flash ROM, it is located on the board and loaded at the power-up time.

3. USB3-DIO01 Description

3.1 USB3-DIO01 Layout



[Figure 3-1. USB3-DIO01 Layout]

The board has 5 LEDs to indicate the operation status.

D8 : turns on when power is applied and the FPGA is ready to run..

D10 : turns on when power is applied to the board.

D4 ~ D7 : For test. Not used.

3.2 Functional Blocks

In this chapter, the primary functions of the board are described briefly. For more information, refer to the device specification.

(1) **FPGA : U4**

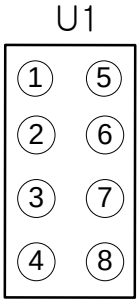
All of the functions are controlled by the logic program of the FPGA.

(2) **USB3.0 SIE : U5**

This block supports USB3.0 Super Speed interface.

(3) **EEPROM : U3**

The 64K Serial I2C EEPROM contains firmware for USB3.0 SIE.



[Figure 3-2. U1 Connecter (Top View)]

[Table 1. U1 Description]

Pin	Signal Name	Description	Remark
1	**A0**	Address 0	+3.3V Power
2	**A1**	Address 1	Ground
3	**A2**	Address 2	Ground
4	**Vss**	Ground	Ground
5	**SDA**	Serial Data	Serial Data
6	**SCL**	Serial Clock	Serial Clock
7	**WP**	Write Protect	3.3V Power
8	**Vcc**	VCC	3.3V Power

(4) **Regulator : U1, U2, U3**

The Regulator is supplying the power (1.8V, 3.3V) to the board.

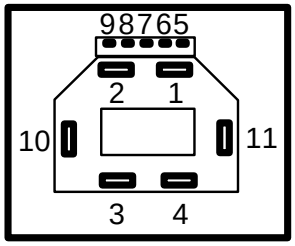
3.3 Connector Pin-out

The board has several connectors and jumpers to set. The USB3-DIO01 board is equipped with two main connecters, One is USB3 B-type connector (CH1) for USB connection and the other is J4 connecter for an external I/O and Image acquisition.

3.3.1 CN1 Connector

The USB3-DIO01 has a USB-B type connector for high speed USB connection.

[Figure 3-3] and [Table 2] shows the CN1 connector and its pin description.



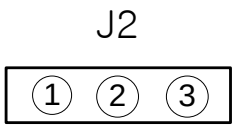
[Figure 3-3. CN1 Connector (USB3.0 standard powered-B type Front View)]

[Table 2. USB3.0 Standard Powered-B Connector]

Pin	Signal Name	Description	Remark
1	**VBus**	+5V Power	+5V Power
2	**USB D-**	USB2.0 data (Negative)	USB2.0 Signal
3	**USB D+**	USB2.0 data (Positive)	USB2.0 Signal
4	**GND**	Ground for Power Return	USB Power GND
5	**StdA_SSTX-**	Super Speed Transmitter (Negative)	USB3.0 Signal
6	**StdA_SSTX+**	Super Speed Transmitter (Positive)	USB3.0 Signal
7	**GND_DRAIN**	Ground for Signal Return	USB Power GND
8	**StdA_SSRX+**	Super Speed Receiver (Positive)	USB3.0 Signal
9	**StdA_SSRX-**	Super Speed Receiver (Negative)	USB3.0 Signal
10	**DPWR**	Power Provided by Device	USB Power GND
11	**DGND**	Ground return for DPWR	USB Power GND

3.3.2 J2 Connector

It is an external RS-232 Debug Port. I2C Idle state of the bus is always High.



[Figure 3-4. J1 Connector (Top View)]

[Table 3. J1 PIN-OUT Description]

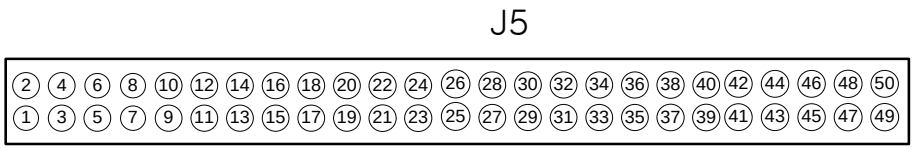
No.	Name	Description
1	**SCL**	Serial Clock
2	**SDA**	Serial DATA
3	**GND**	Ground

3.3.3 J3 Jumper

Caution) This jumper have to be removed when using the power of J7 12V DC-JACK connector.

3.3.4 J5 Connector

This connecter uses to pass through signals that are Red, Green, Blue signals, CC (Camera Control) signals and Image Control (LVAL, FVAL, Clock) signals, that are 32-bit digital input/output signals.



[Figure 3-5. J5 Connector (Top View)]

[Table 4. J5 Connector Description]

No.	Name	Description	Remark
1	**+3.3V**	+3.3V Power	
2	**+5V**	+5V Power	
3	**DIO_0**	Digital Input 0	

4	**DIO_1**	Digital Input 1	
5	**DIO_2**	Digital Input 2	
6	**DIO_3**	Digital Input 3	
7	**DIO_4**	Digital Input 4	
8	**DIO_5**	Digital Input5	
9	**DIO_6**	Digital Input 6	
10	**DIO_7**	Digital Input 7	
11	**DIO_8**	Digital Input 8	
12	**DIO_9**	Digital Input 9	
13	**DIO_10**	Digital Input10	
14	**DIO_11**	Digital Input 11	
15	**DIO_12**	Digital Input12	
16	**DIO_13**	Digital Input 13	
17	**DIO_14**	Digital Input 14	
18	**DIO_15**	Digital Input 15	
19	**GND**	Ground	
20	**GND**	Ground	
21	**DIO_16**	Digital Input 16	
22	**DIO_17**	Digital Input 17	
23	**DIO_18**	Digital Input 18	
24	**DIO_19**	Digital Input 19	
25	**DIO_20**	Digital Input 20	
26	**DIO_21**	Digital Input 21	
27	**DIO_22**	Digital Input 22	
28	**DIO_23**	Digital Input 23	
29	**DIO_24**	Digital Input 24	
30	**DIO_25**	Digital Input 25	
31	**DIO_26**	Digital Input 26	
32	**DIO_27**	Digital Input 27	
33	**DIO_28**	Digital Input 28	
34	**DIO_29**	Digital Input 29	
35	**DIO_30**	Digital Input 30	
36	**DIO_31**	Digital Input 31	
37	**DIO_32**	Digital Input 32	PCLK(Pixel Clock)
38	**DIO_33**	Digital Input 33	Vsync
39	**DIO_34**	Digital Input 34	Hsync

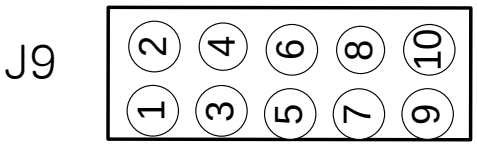
40	**DIO_35**	Digital Input 35	DE(Data Enable)
41	**DIO_36**	Digital Input 36	GPIO1
42	**DIO_37**	Digital Input 37	GPIO2
43	**DIO_38**	Digital Input 38	GPIO3
44	**DIO_39**	Digital Input 39	GPIO4
45	**REV1**	Reserver 1	cReset
46	**U_SDA**	Serial Data	Refer J12
47	**REV0**	Reserved 0	MCLK(Master Clock)
48	**U_SCL**	Serail Clock	SCL
49	**GND**	Ground	
50	**GND**	Ground	

3.3.5 J7 Connector

It's an external 12V DC Jack power connecter.

3.3.6 J9 Connecter

The J9 connector is used for the FPGA program upgrade. Never use it at the normal operation. [Figure 3-6] shows the J9 connector.



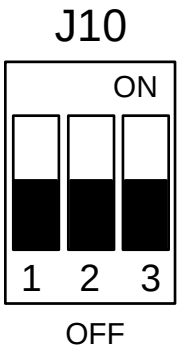
[Figure 3-6. J9 Connecter (Top View)]

[Table 5. J9 Connecter Description]

Bo.	Name	Description
1	**BTCK**	Clock
3	**BTMS**	Mode Select
5	**BTDI**	Data In
7	**BTDO**	Data Out
2,4,8	**GND**	GND
6	**3.3V**	+3.3V Power
9	**N.C**	No Connection
10	**PROG_B**	

3.3.7 J10

The USB3-DIO01 board is designed of four maximum USB3-DIO01 boards at the same time so as usable. Distribution of each board sets it up through the switch in a board.



[Figure 3-7. J10 Switch]

[Table 6. J2 Switch Description]

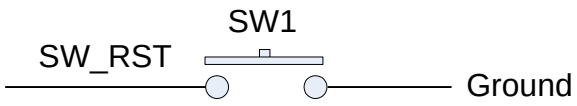
1	2	3	Description
OFF	OFF	OFF	Board No. 0
ON	OFF	OFF	Board No. 1
OFF	ON	OFF	Board No. 2
ON	ON	OFF	Board No. 3
OFF	OFF	On	Board No. 4
ON	OFF	On	Board No. 5
OFF	ON	On	Board No. 6
ON	ON	On	Board No. 7

3.3.8 J12

The 12V input power is applied to the no. 45-pin of J5 connector when jumper connection.

3.3.9 SW1

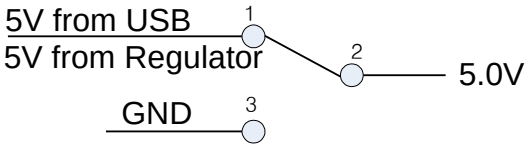
It is a Reset Switch (low active).



[FIgure 3-8. SW1 Switch]

3.3.10 SW3

Board Power (5V from USB or Power Generator) On/Off Switch. When the switch is Up state, power(5V) is On.



[Figure 3-9. SW3 Switch]

4. Installation

After unpacking, inspect the board to make sure there are no damages on the package.

4.1 Package Contents

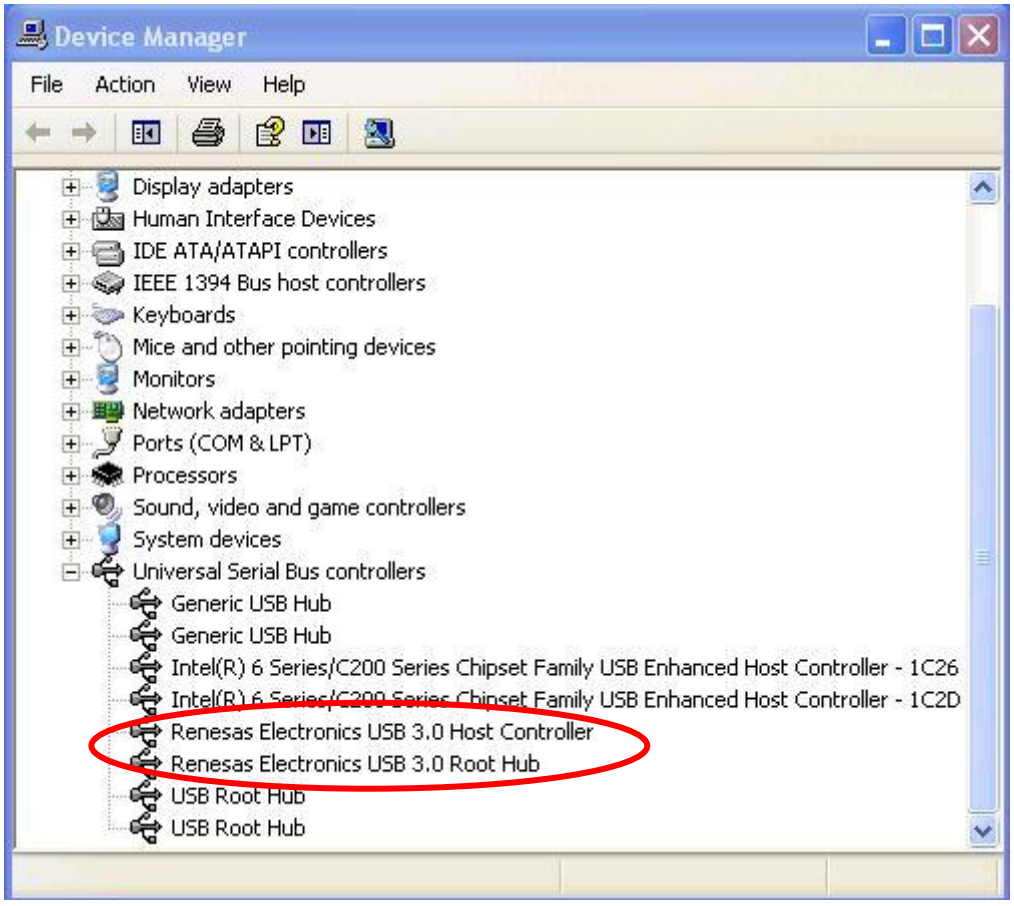
Product Contents

1. USB3-DIO01 Board
2. USB3(A-B) Cable
3. CD (Driver/Manual/API/Samples etc.)

4.2 Installation Sequence

To install USB3-DIO01 board in your environment, do the following steps. The USB3-DIO01 board is completely Hot-Plug and Plug & Play. Therefore, you can install it easily.

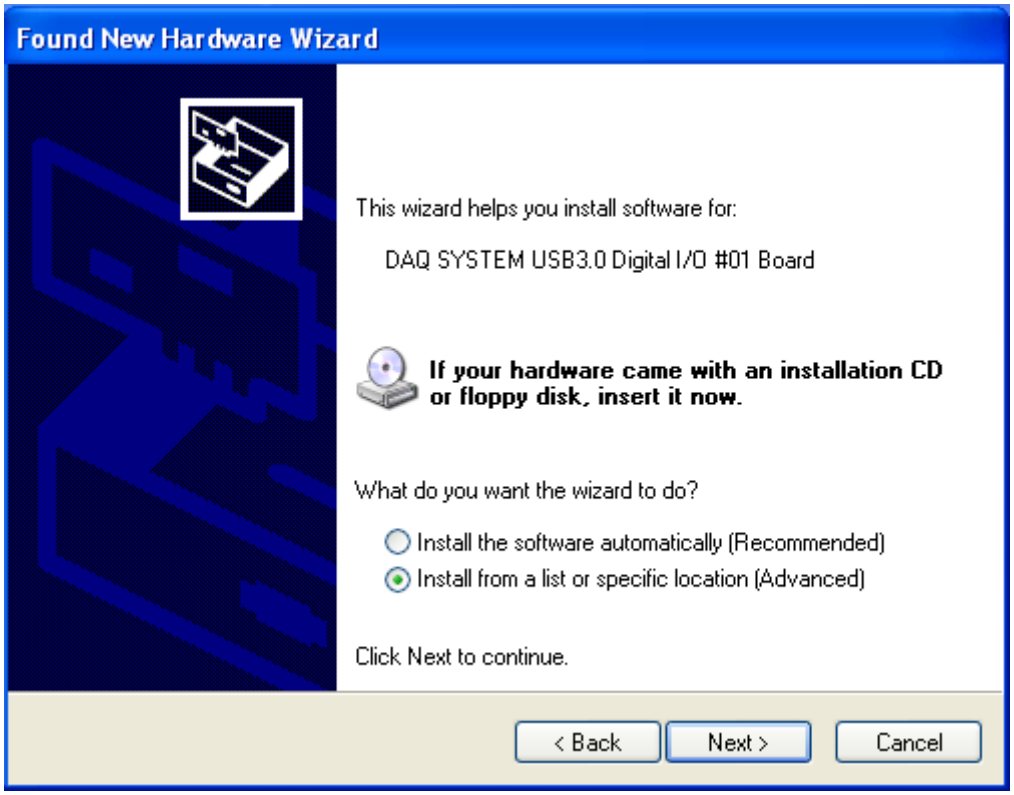
The required PC operating system for the USB3-DIO01 is Windows 2000 SP4 or Windows XP SP1 higher. The USB3-DIO01 uses USB Super Speed interface thus "xxx **USB 3.0 Root Hub**" should be installed in your PC. You can check this condition by doing the following steps.



[Figure 4-1. "Device Manager" window]

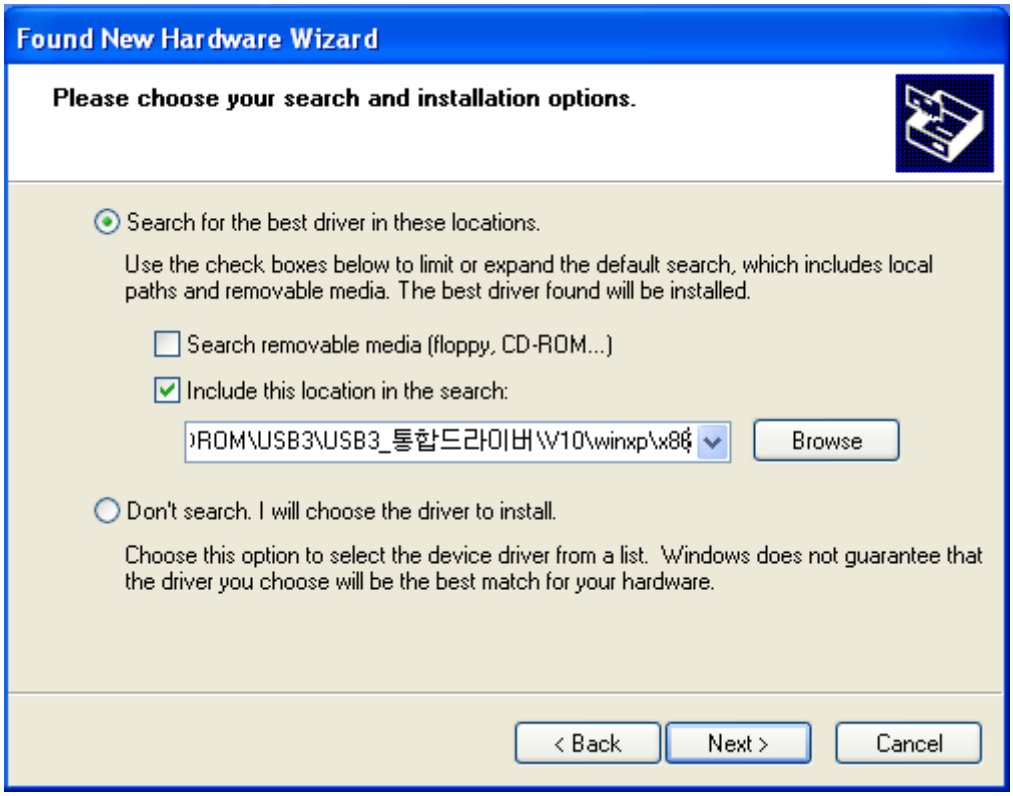
The item “USB 3.0 Root Hub” should be shown in the “Device Manager” window as shown in [Figure 4-1]. After checking the PC environmental conditions for USB3-DIO01, do the following steps to install the board

(1) Install the USB3-DIO01 board into your system.
(2) Power on the frame grabber.
(3) Confirm the LED (D8) and LED (D10) on the USB3-DIO01 board turns on.
(4) After check the power, Connect USB3 A-B cable between the case and your PC.
The Add New Hardware Wizard will appear in order to install the driver for new hardware.
(5) The Add new Hardware Wizard will install the driver in the following process. The following install process is explained based on Windows XP operating system.



[Figure 4-2. “Hardware Wizard” window]

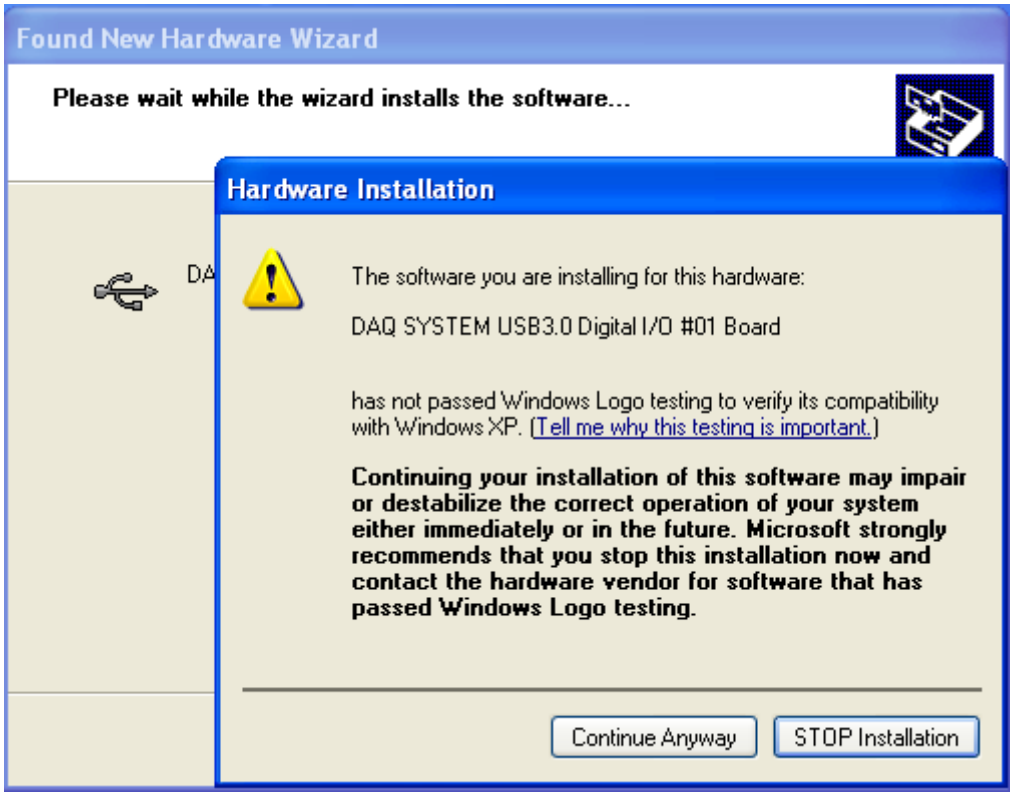
If new hardware is found, Wizard will ask you to install the corresponding driver, For installation of the driver, select item “Install from a list or specific location(Advanced)” and click “Next” as in the [Figure 4-3].



[Figure 4-3. Specify the driver folder]

Select “Search for the best driver in these locations”. Check “Search removable media (floppy, CD-ROM)”. Check “include this location in the search”. Click “Browse” button. Select the folder where the drivers are located. Click “OK”. Click “Next”.

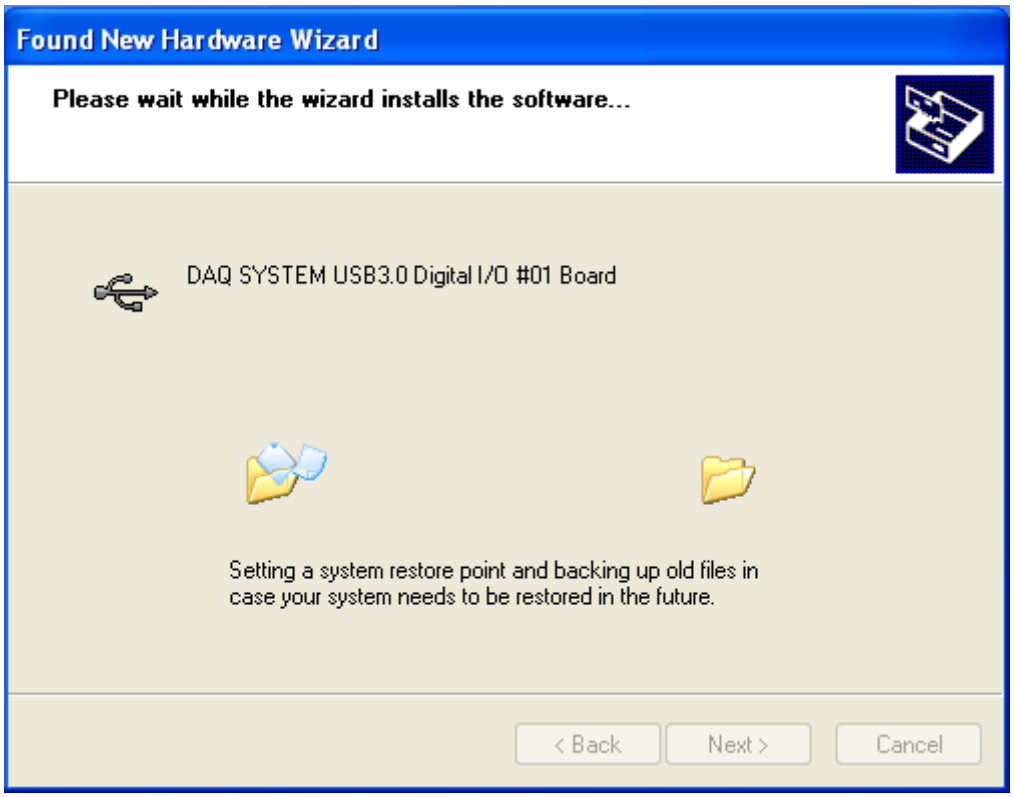
The necessary files are **“cyusb3.inf”** and **“cyusb3.sys”** in the driver polder.



[Figure 4-4. Warning window]

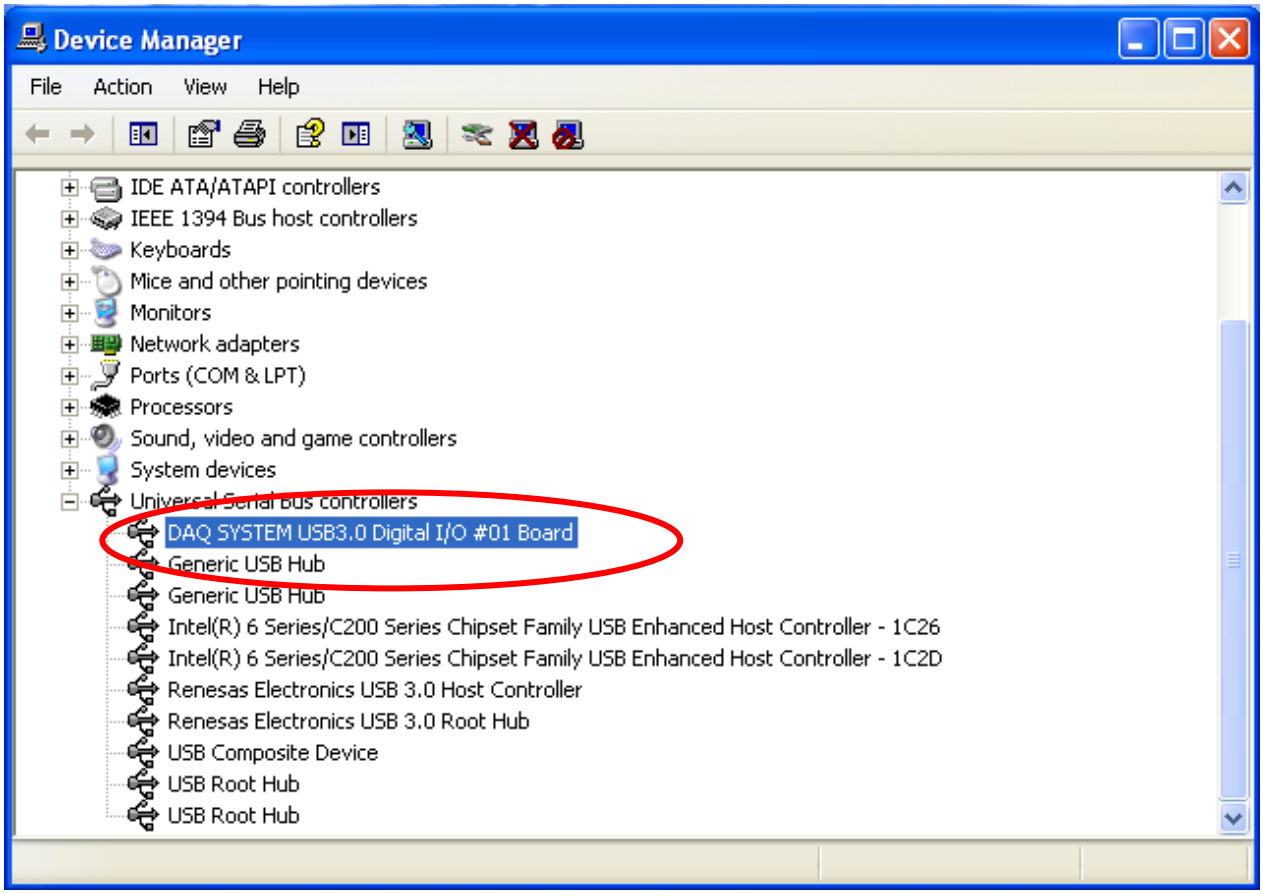
When a window's warning message will be appeared regarding the compatibility problem as shown the [Figure 4-4] during the installation process, just click "Continue" button and go on the installation.

If the installation is completely finished, a completion window message shall be shown as in [Figure 4-5]. Click "Finish".



[Figure 4-5. "Completion" message window]

If you successfully complete the wizard, you can find the item **"DAQ SYSTEM USB3.0 Digital I/O #01 Board"** in the "Device Manager" window as shown in [Figure 4-6].



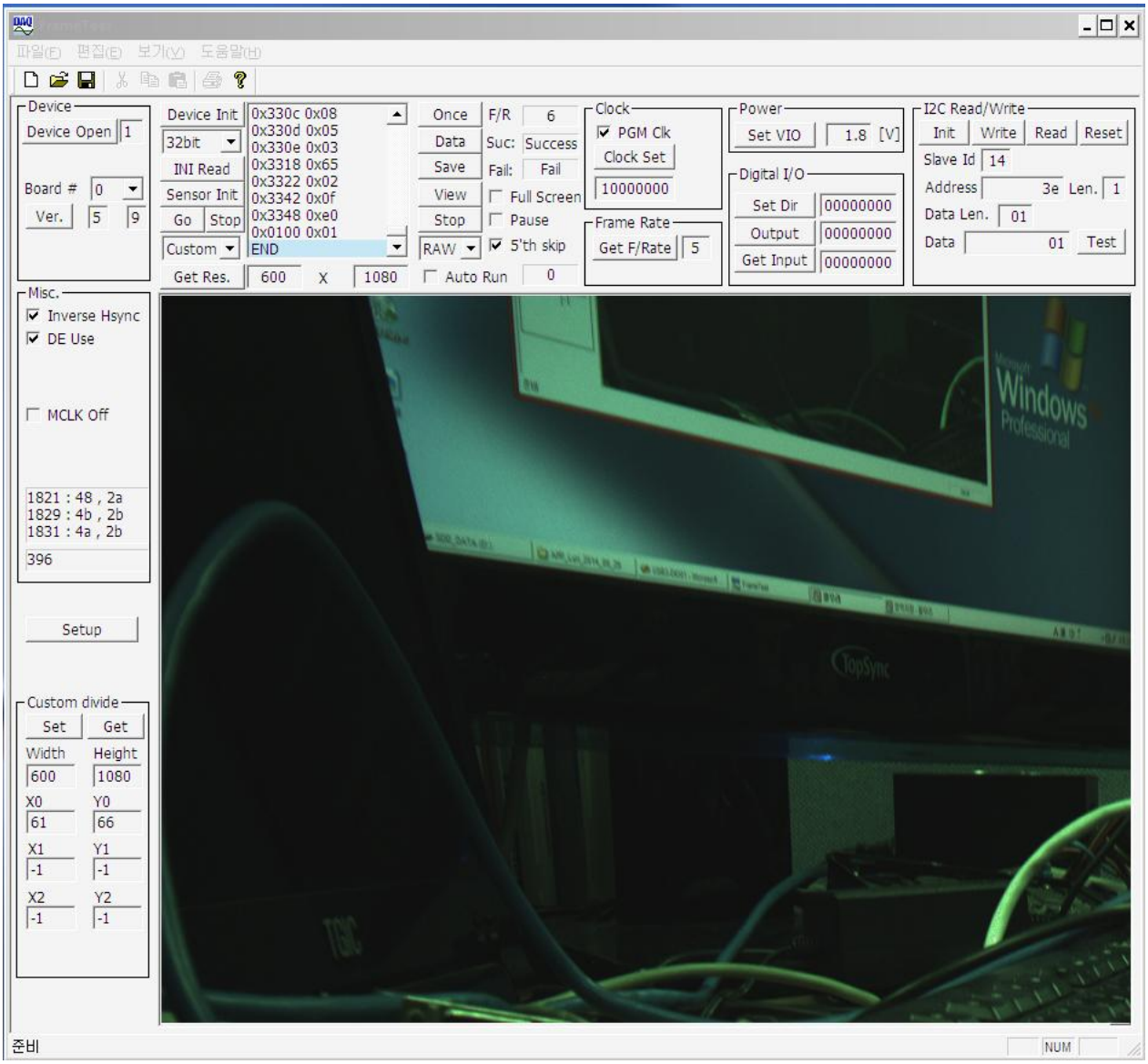
[Figure 4-6. "Device Manager" window]

If you can see the "DAQ SYSTEM USB3.0 Digital I/O #01 Board" at the Universal Serial Bus controllers, the driver installation is to have been over. (Check the red circle)

5. Sample Program

DAQ system provides a sample program to make the user be familiar with the board operation and to make the program development easier. You can find the sample program in the CDROM accompanying with the board.

Sample program is provided in source form in order to show the usage of API (Application Programming Interface) of the board and may be modified for customer's own usage.

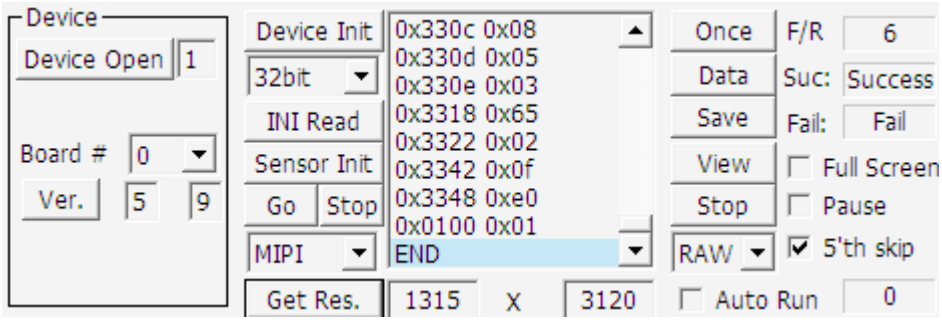


[Figure 5-1. When Sample program "FrmTest.exe' is executed]

To run the sample application program, you need to use API (Application Programming Interface). It is a form of client DLL (Dynamic Link Library). You need the Import Library files and header files for compiling the sample source. You can find them in the CDROM. To run the execution file, the API

DLL file **(USB_DIO01.DLL)** must be located in the same directory with the execution file or Windows system folder. Another method is to add the directory of API DLL file to PATH environmental variable.

5.1 Functions related to Image Frame



(1) **"Device Open" button**

It starts a selected board device. If the value is "0", the device is not connected or no device.

(2) **"Board # :"**

It selects a board number in case of the multi USB3.0 boards. It can select 0 ~ 4 at currently.

(3) **"Get Version" button**

Get current FPGA and Firmware version.

(4) **"Device Init" button**

Video Data Mode to 8bit, 16bit, 24bit, 32bit of image frames, and then initialize the function of receiving image frame data. It is performed only once after power is applied to the board.

(5) **"INI Read" button**

The initialization file will be read all at once at "INI read" button. You can be written an address and data one by one(I2C Read/Write part).ini file structure as described is as follows.

Ex) sony13M_full.ini File Structure

```
[REGISTER]
    Slave    0x10   //change slave ID as Sensor
    SLEEP    100
    0x3087   0x53
```

0x309D 0x94
0x30A1 0x08
0x30AA 0x04
0x30B1 0x00
0x30C7 0x00
0x3115 0x0E
0x3118 0x42
0x311D 0x34
0x3121 0x0D
0x3212 0xF2
0x3213 0x0F

…….

0x3306 0x12
0x3307 0x03
0x3308 0x0D
0x3309 0x05
0x330A 0x09
0x330B 0x04
0x330C 0x08
0x330D 0x05
0x330E 0x03
0x3318 0x65
0x3322 0x02
0x3342 0x0F
0x3348 0xE0

0x0100 0x01//Streaming

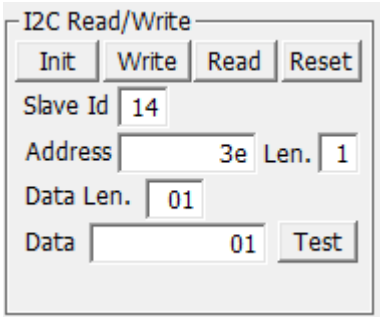
When you want to control the FPGA, Slave Address will be fixed to 0x14. The MIPI control signals that you want have to be written the data to the corresponding address register.

Example) When the FPGA controls, SLAVE ADDRESS(7bit) is fixed to 0x14.

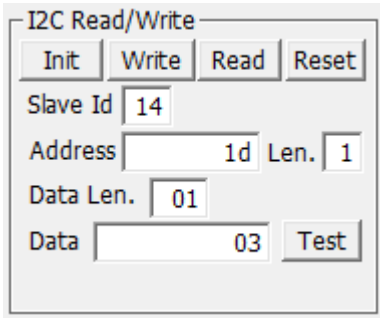
Register Address

0x1D 0x03 : MIPI 4 Lane
0x01 : MIPI 2 Lane
0x00: MIPI 1 Lane

For example, if you are using our own MIPI-SENSOR03



On the first screen, write (reset to sensor)



MIPI 4lane is used when data 0x03 is written to Address 1d.

MIPI 2lane is used when data 0x01 is written to Address 1d.

MIPI 1lane is used when data 0x00 is written to Address 1d.

(6) **"Sensor Init" button**

The initialization file will be read all at once at "INI read" button.

(7) **"GO" button**

Open the device, initialize the device, open the ini file, bring the resolution, these operations execute at a time.

(8) **"STOP" button**

Stop using the device.

(9) **"Input Video Mode" Selection**

Select from YUV, BT656, MIPI, Custom Mode among.

(10) **"Get Res." button**

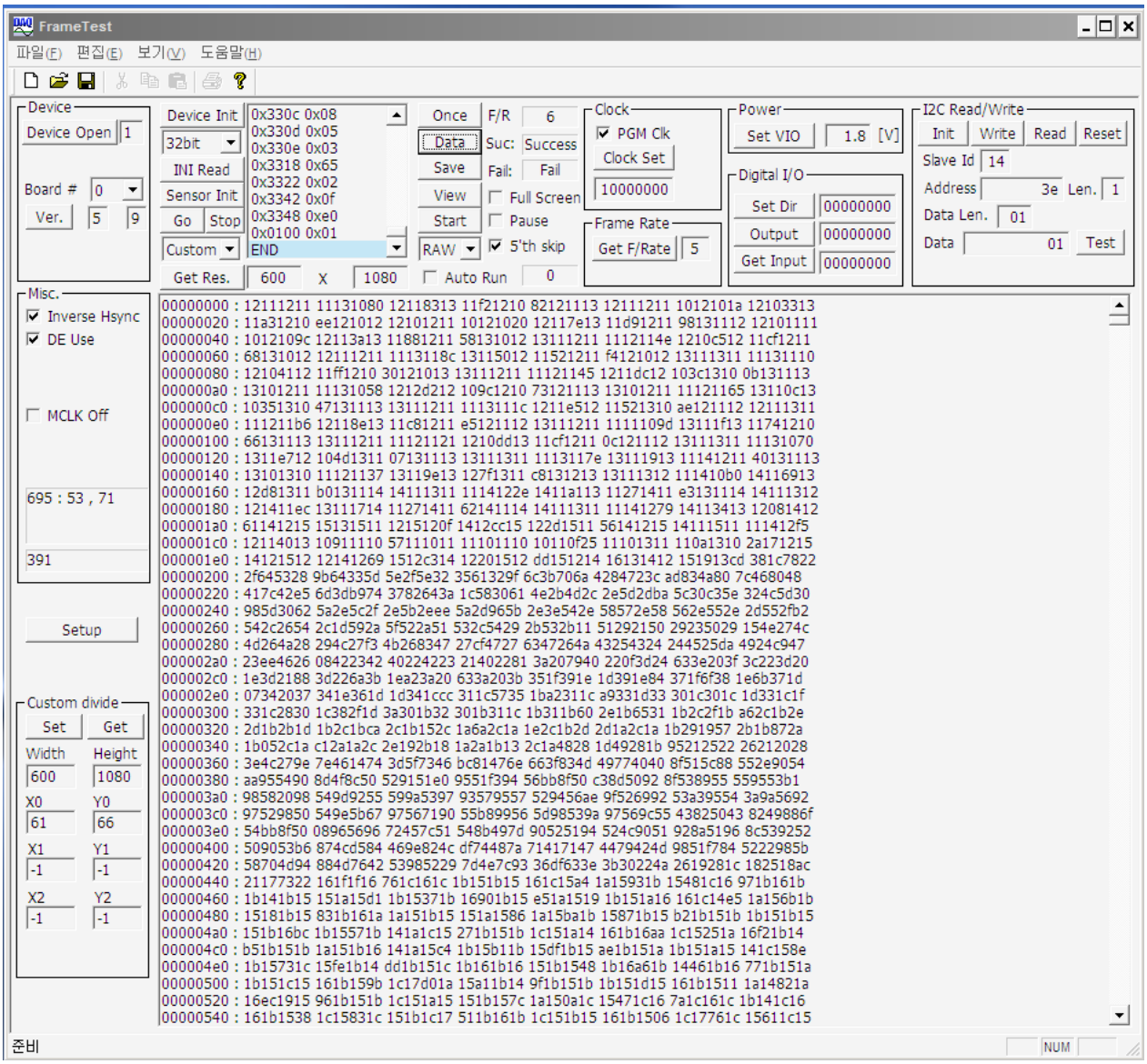
It shows the image resolution.

(11) **“Once” button**

It is a Toggle button. When the button presses, it displays a stop image at that time.

(12) **“Data” button**

Press this button to read the image frame data of the board to your PC. If image frame data is not saved on the board, you must wait until the end of data collection.



[Figure 5-1. When Sample program “DATA’ is executed]

(13) **“Save” button**

Save the frame on the board. Save by binary format type to the default folder.

(14) **"View" button**

Image Transfer begins to start.

(14) **"Start" button**

It starts the image transfer. It is a Toggle button, press again stop the image transferring.

(15) **"Frame Data Format" Selection**

Select the Frame Data type from YUV, RGB, RAW among.

(17) **"Auto Run" toggle**

Features are not implemented.

(18) **"F/R : "**

Frame rates/sec

(19) "**Suc : Success" and "Fail : Fail"**

"Suce: Success" : It shows good image transfer state.

"Fail" : It shows bad image transfer state.

(20) **"Full screen" toggle**

It shows full screen.

(21) **"Pause" toggle**

When check this box, it stops the displayed image.

(22) **"5'th Skip" toggle**

When this button is selected, it goes beyond the 5th byte from the image data.

For 10-bit Bayer, it is stored in the 5 th byte of each remaining one bit. The 5 th byte is not required , it is used by removing it.

(23) **"inverse Hsync" toggle**

If the box clicks, you use the inverse HSYNC signal.

If the box doesn't click, you use the HSYNC signal.

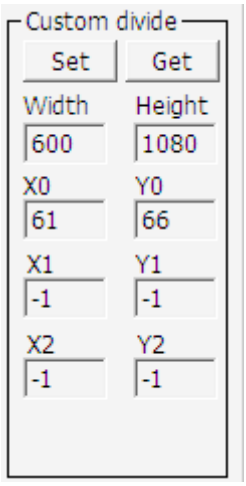
(24) **"DE use" toggle**

If the box clicks, you use the Data Enable signal instead of Hsync.

If the box don't click, you use the HSYNC(Horizontal Synchronization) signal.

(25) **"Custom Divide"**

You can divide the screen into nine equal parts.



Set : Set the results in the coordinates x and y of dividing a frame into 9 equal parts by the user..

Get: Get the results in the coordinates x and y of dividing a frame into 9 equal parts by the user.

Width : Each divided screen Width.

Height : Each divided screen height.

X0 : First X coordinate values you want to set

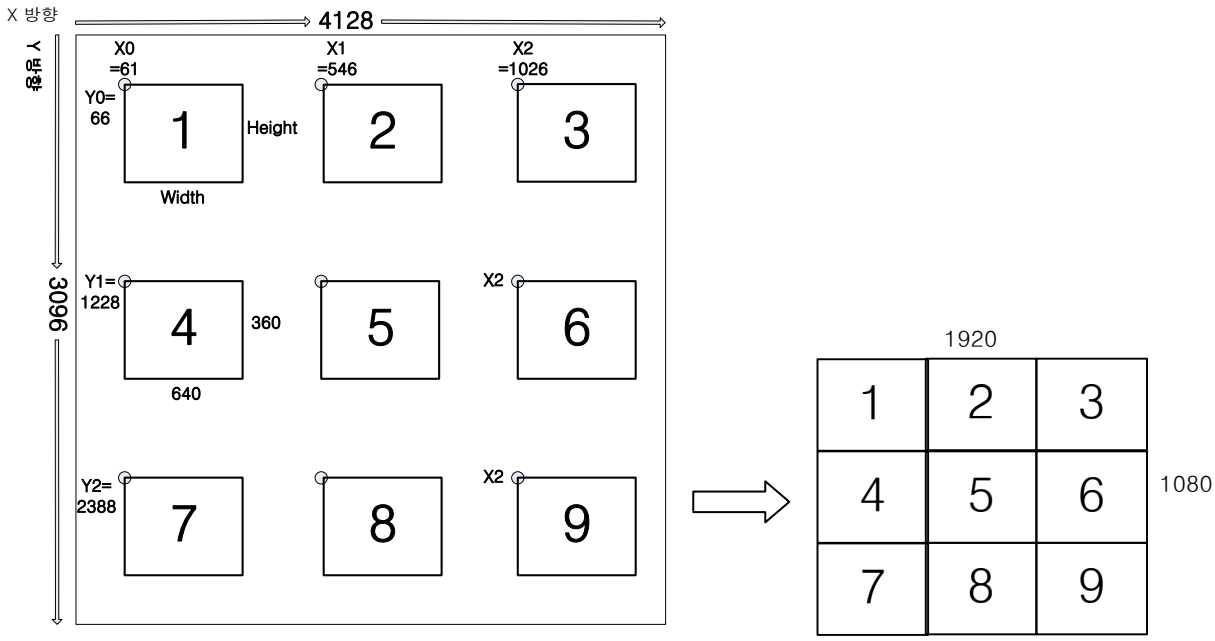
X1 : Second X coordinate values you want to set

X2 : Third X coordinate values you want to set

Y0 : First Y coordinate values you want to set

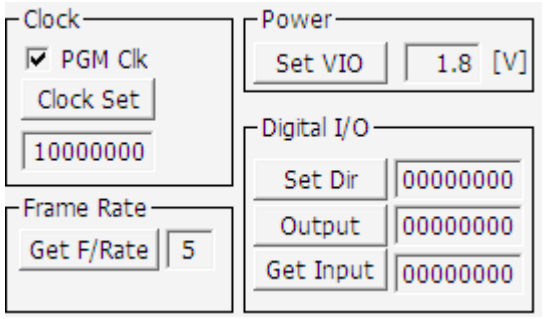
Y1 : Second Y coordinate values you want to set

Y2 : Third Y coordinate values you want to set



< X, Y coordinates and the width, height Relationships>

5.2 Functions related to Clock & Power



(1) **"PGM Clk" toggle**

You can set the Clock to 1039Hz ~ 68MHz.

(2) **"Get F/Rate**

It shows the number of frames in the display.

(3) **"Set VIO"**

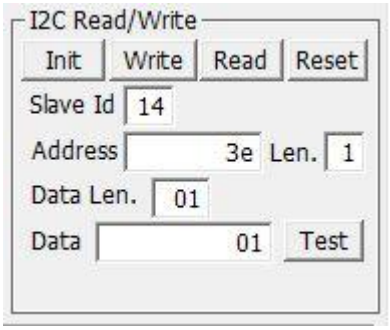
You may adjust the power to the board. (Function is not implemented)

(4) **"Digital IO"**

It will be used later when you add Digital Input / Output functions.

(Function is not implemented)

5.3 Functions related to I2C



(1) **"Init" Button**

It will initiate the resources for the I2C system.

(2) **"Write" Button**

Transmit the data through I2C for control to MIPI or CMOS camera.

(3) **"Read" Button**

Receive the data through I2C for control to MIPI or CMOS camera.

(4) **"Reset" Button**

There is Initialization resource of I2C system.

(5) **"Slave ID " :** Slave Address

"Reg. Addr " : Slave Register Address

"Addr Len :" : Address Lenght

"Data Len :" : Data Lenght

"Data :" : Data that will transmit

The above price is variable when using **I2C Read or I2C Write**.

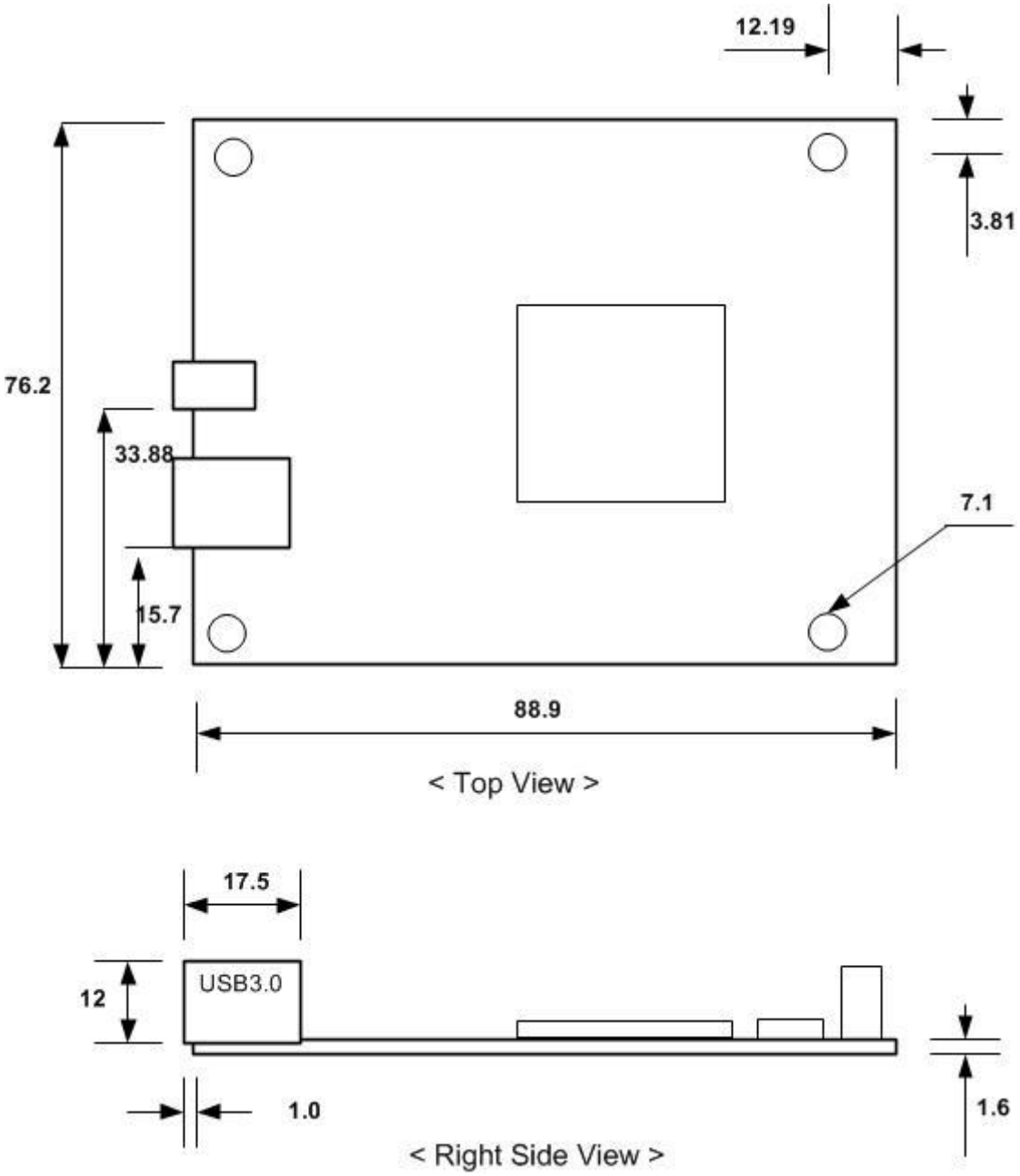
Appendix

A.1 General Specification

Specification	
General	♦ USB 3.0 Specification with type-B connecter ♦ Super Speed Device 5Gbps
Interface	♦ High-resolution, High-Speed image acquisition Base Board
Functions	♦ Supports fully Plug-and-Play ♦ 256MByte DDR2 Memory
Software	
Supported OS	Windows XP/7/8/10 32/64bit
API	Interface with Application through client DLL
Sample Software	Test Sample software for evaluation

A.2 Physical Dimension.

The approximate external dimensions of the board are as follows. For more detailed measurements, ask the person in charge.



References

1. USB 3.0 System Architecture

-- Don Anderson, USB SIG (www.usb.org)

2. Universal Serial Bus Specification

-- Compaq/Intel/Microsoft/NEC, MindShare Inc. (Addison Wesley)

3. AN201 How to build application using API

-- DAQ system

4. AN342 USB3-DIO01 API ver1.2

-- DAQ system