USB3-FRM02

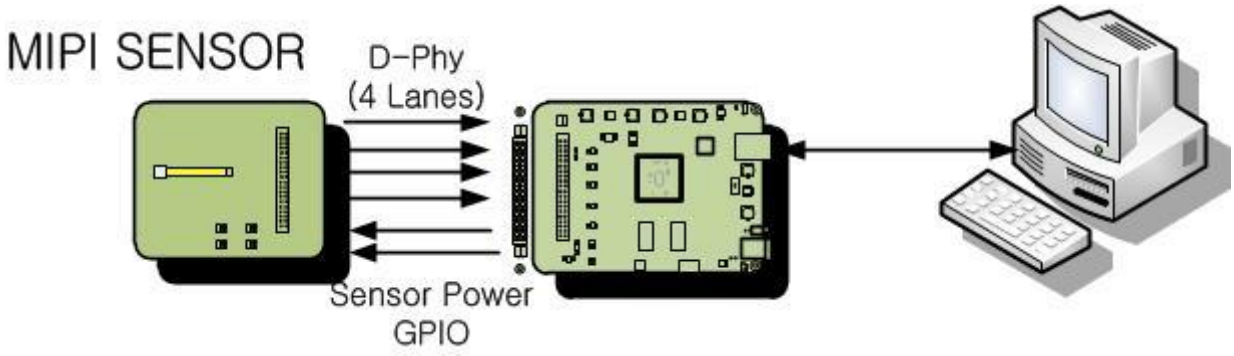
User's Manual

-- Contents –

1. Introduction

The USB3-FRM02 board transmits the MIPI (Mobile Industry Processor Interface) signal of the sensor board and the I/O (Parallel) signal of another image board to the PC in the USB3.0 Super Speed (5Gbps) method. The received signal is processed by software (application) provided by the DQ system on the PC and displayed as an image.

A [Figure 1-1] shows Input connection of USB3-FRM02 board.



[Figure 1-1. USB3-FRM02 Board Usage]

1.1. Product Specification

Item	Description	Remark
Hardware		
PC Interface	USB3.0	B-Type
Power	+12VDC/ 620mA	External 12V DC Power (A6-Type : 5.5x2.1mm)
Video Interface	MIPI CSI DPHY 4 Lane MIPI CSI Virtual Mode	1.2Gbps / 1 Lane
In/Out Terminal	3.3V GPIO 4bit 1.5 ~ 3.3V GPIO 8bit	
On-board memory	256MB (DDR3 x2)	
Communication	I2C/SPI	
Simultaneously Use	Max. 4	
Software		
OS	Windows 2000/XP/7/8/10 (32/64bit)	
API	Windows Client DLL API	
Development	Windows Application by User Custom USB Device Firmware Custom Windows Client DLL	
Support	Sample Program	VC++ (C# please contact us)
Environmental conditions		
Operating temperature range	0 ~ 60℃	
Storage temperature range	-20 ~ 80℃	
Humidity range	5 ~ 95%	Non-condensing
Board Size	80mm X 100mm	

1.2 Product Application

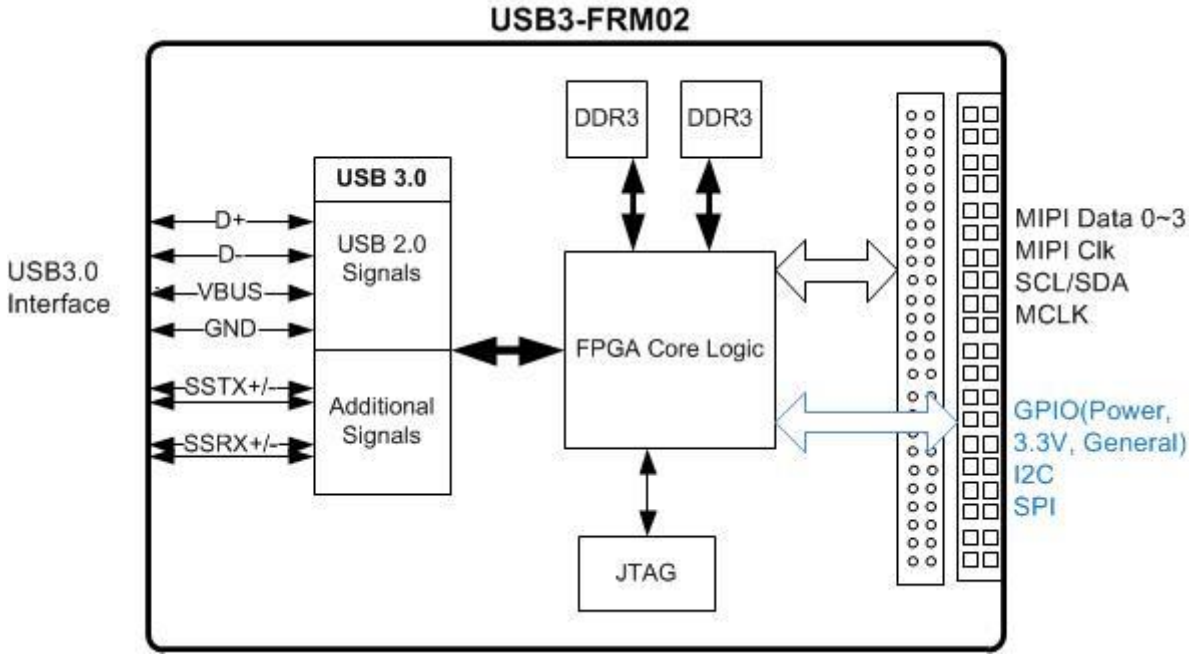
- Frame Grabber
- Test for Variable MIPI Sensor

2. USB3-FRM02 Function

2.1 Board Block Diagram

As shown in the following figure, main control of the board is performed FPGA Core Logic. The main function is to send MIPI Image Frame Data through external I/O conne ctor. These functions are performed by using 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 3.3.3.

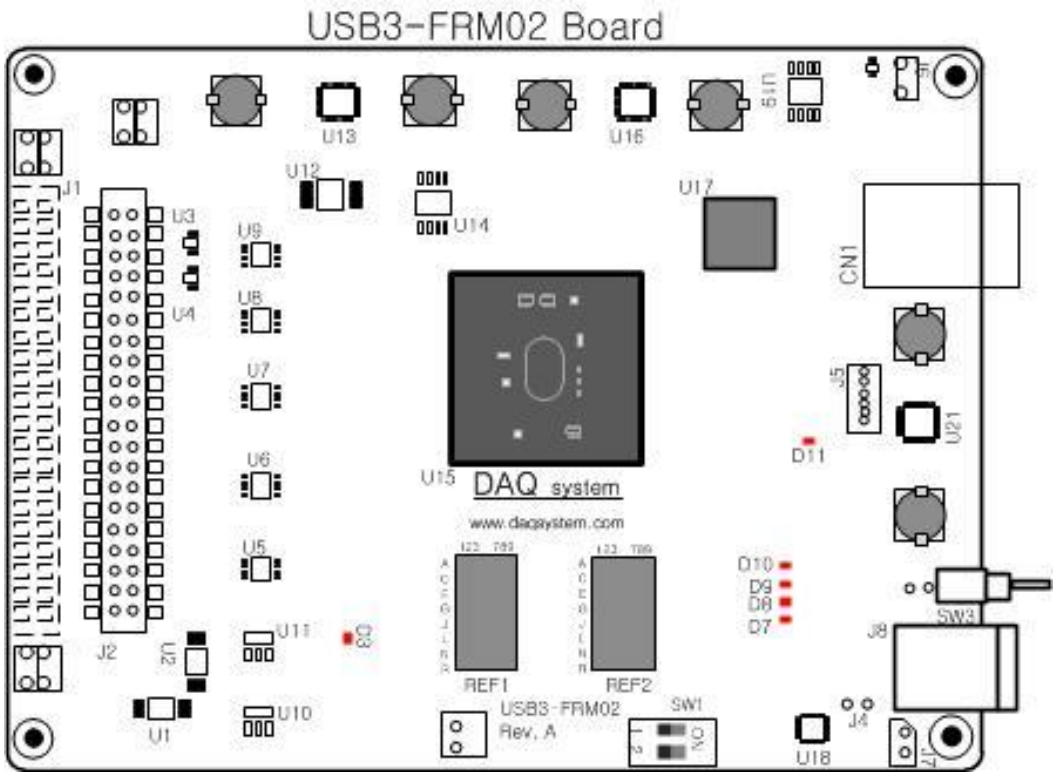


[Figure 2-1. USB3-FRM02 Functional Block Diagram]

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-FRM02 Description

3.1 USB3-FRM02 Layout



[Figure 3-1. USB3-FRM02 Layout]

The board has 6 LEDs to indicate the operation status.

D3 : turns on when power is applied and the FPGA is ready to run..

D8 : turns on when Vertical Synchronization (Vsync) divided by 1/16 is displayed. This is an indication to visually confirm Vsync.

D9 : turns on when Horizontal Synchronization (Hsync) divided by 1/16 is displayed. This is an indication to visually confirm Hsync.

D7 : When blinking, DDR memory is normal.

D10 : turns on when USB3.0 booting is complete.

D11 : turns on when 3.3V power is applied to the board.

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 : U15**

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

(2) **USB3.0 SIE : U17**

This block supports USB3.0 Super Speed interface.

(3) **DDR Memory : REF1, REF2**

DATA is stored frame by frame and transferred to PC through FPGA

(4) **Regulator : U2, U11, U12, U13, U16, U18, U21**

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

(3) **LVDS Buffer : U5 ~ U9**

3.125Gbps LVDS Buffer

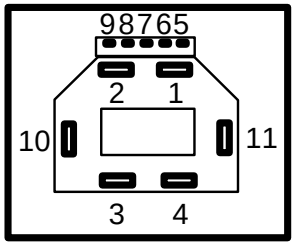
3.3 Connector Pin-out

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 3-2] and [Table 1] shows the CN1 connector and its pin description.



[Figure 3-2. CN1 Connector (USB3.0 standard powered-B type Front View)]

[Table 1. 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 J1 Connector (2x20 Pin Straight Female DIP Type)

It is connected to the GPIO board or the power board, and the signals are as follows.

J1

2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39

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

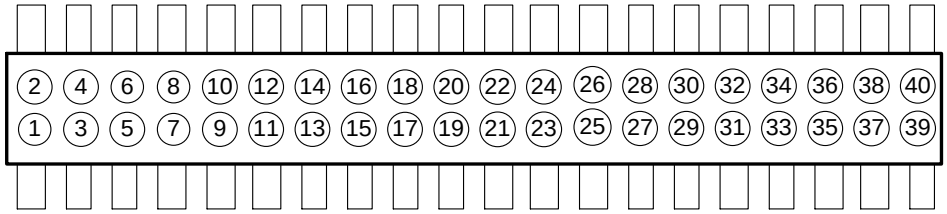
[Table 2. J1 PIN-OUT Description]

No.	Name	Description	Remark
1	**P_GPO_0**	Power Purpose I/O0	For Power Board
2	**P_GPO_1**	Power Purpose I/O1	For Power Board
3	**P_GPO_2**	Power Purpose I/O2	For Power Board
4	**P_GPO_3**	Power Purpose I/O3	For Power Board
5	**P_GPO_4**	Power Purpose I/O4	For Power Board
6	**P_GPO_5**	Power Purpose I/O5	For Power Board
7	**P_GPO_6**	Power Purpose I/O6	For Power Board
8	**P_GPO_7**	Power Purpose I/O7	For Power Board
9	**GND**	Ground	
10	**GND**	Ground	
11	**PWR_SCL**	Power SCL	For Power Board
12	**PWR_SDA**	Power SDA	For Power Board
13	**AD_SCL**	AD SCL	To be updated
14	**AD_SDA**	AD SDA	To be updated
15	**N.C**	No Connection	
16	**N.C**	No Connection	
17	**OS_SCL**	OS SCL	To be updated
18	**OS_SDA**	OS SDA	To be updated
19	**GND**	Ground	
20	**GND**	Ground	
21	**3.3V_GPIO0**	3.3V Purpose I/O0	3.3V
22	**3.3V_GPIO1**	3.3V Purpose I/O1	3.3V
23	**3.3V_GPIO2**	3.3V Purpose I/O2	3.3V
24	**3.3V_GPIO3**	3.3V Purpose I/O3	3.3V

25	**GND**	Ground	
26	**GND**	Ground	
27	**GPIO0**	General Purpose I/O0	1.5 ~ 3.3V
28	**GPIO1**	General Purpose I/O1	1.5 ~ 3.3V
29	**GPIO2**	General Purpose I/O2	1.5 ~ 3.3V
30	**GPIO3**	General Purpose I/O3	1.5 ~ 3.3V
31	**GPIO4**	General Purpose I/O4	1.5 ~ 3.3V
32	**GPIO5**	General Purpose I/O5	1.5 ~ 3.3V
33	**GPIO6**	General Purpose I/O6	1.5 ~ 3.3V
34	**GPIO7**	General Purpose I/O7	1.5 ~ 3.3V
35	**GND**	Ground	
36	**GND**	Ground	
37	**SPI_SCK**	Clock	To be updated
38	**SPI_MISO**	Master In Slave Out	To be updated
39	**SPI_SSN**	Select	To be updated
40	**SPI_MOSI**	Master Out Slave In	To be updated

3.3.3 J2 Connector (2x20 Pin Straight SMD Male Type)

It is connected to MIPI SENSOR board and the signal is as follows.



[Figure 3-4. J2 Connector (Top View)]

[Table 3. J2 Connector Description]

No.	Name	Description	Remark
1	**SP0**	SENSOR Power	0.9 ~ 4.1V
2	**SP1**	SENSOR Power	0.9 ~ 4.1V
3	**SP2**	SENSOR Power	0.9 ~ 4.1V
4	**SP3**	SENSOR Power	0.9 ~ 4.1V
5	**GND**	Ground	
6	**GND**	Ground	
7	**SCL**	Serial Clock	

8	**DATAP_0**	MIPI 1 Lane Positive	
9	**SDA**	Serial Data	
10	**DATAN_0**	MIPI 1 Lane Negative	
11	**GND**	Ground	
12	**GND**	Ground	
13	**ENB**	Enable	
14	**DATAP_1**	MIPI 2 Lane Positive	
15	**S_RESET**	Reset	
16	**DATAN_1**	MIPI 2 Lane Negative	
17	**GND**	Ground	
18	**GND**	Ground	
19	**CNT0**	Control 0	
20	**DATAP_2**	MIPI 3 Lane Positive	
21	**CNT1**	Control 1	
22	**DATAN_2**	MIPI 3 Lane Negative	
23	**GND**	Ground	
24	**GND**	Ground	
25	**CNT2**	Control 2	
26	**DATAP_3**	MIPI 4 Lane Positive	
27	**CNT3**	Control 3	
28	**DATAN_3**	MIPI 4 Lane Negative	
29	**GND**	Ground	
30	**GND**	Ground	
31	**GND**	Ground	
32	**CLKP**	MIPI Clock Positive	
33	**GND**	Ground	
34	**CLKN**	MIPI Clock Negative	
35	**MCLK**	Master Clock	
36	**GND**	Ground	
37	**GND**	Ground	
38	**GND**	Ground	
39	**SP4**	SENSOR Power	
40	**SP5**	SENSOR Power	

Caution) 5W in total from SP0 to SP5

3.3.4 J4 Connector

This connector is used when USB power (5V) is used as the main power for jumper connection. If the USB power is insufficient, the program may crash or there may be no video.

3.3.5 J7 Connector

This connector is an external input 12V power connector.

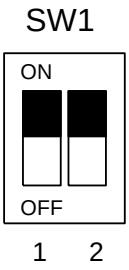
3.3.6 J8 Power jack

This connector is an external 12V DC Jack (A6 Type: 5.5x2.1mm) power connector of DC-005 (2.0) standard. (Recommended basic use)

[Figure 3-5. Power Connection]

3.3.6 SW1 Switch

The USB3-FRM02 board is designed for simultaneous use of up to four USB3-FRM02 boards in one system (PC). Each board's classification can be set via a 4-pin DIP switch in the board.



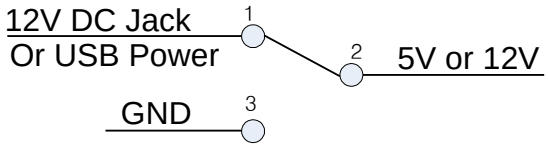
[Figure 3-6. SW1 Switch (Top View)]

[Table 3. J2 Connector Description]

1	2	Description
OFF	OFF	Board No. 0
ON	OFF	Board No 1
OFF	ON	Board No 2
ON	ON	Board No 3

3.3.6 SW3 Switch

Board Power (5V from USB: using J4 or External 12V Power Generator: using J8) 5V or 12V power is On when the terminal is pushed up with On / Off switch.

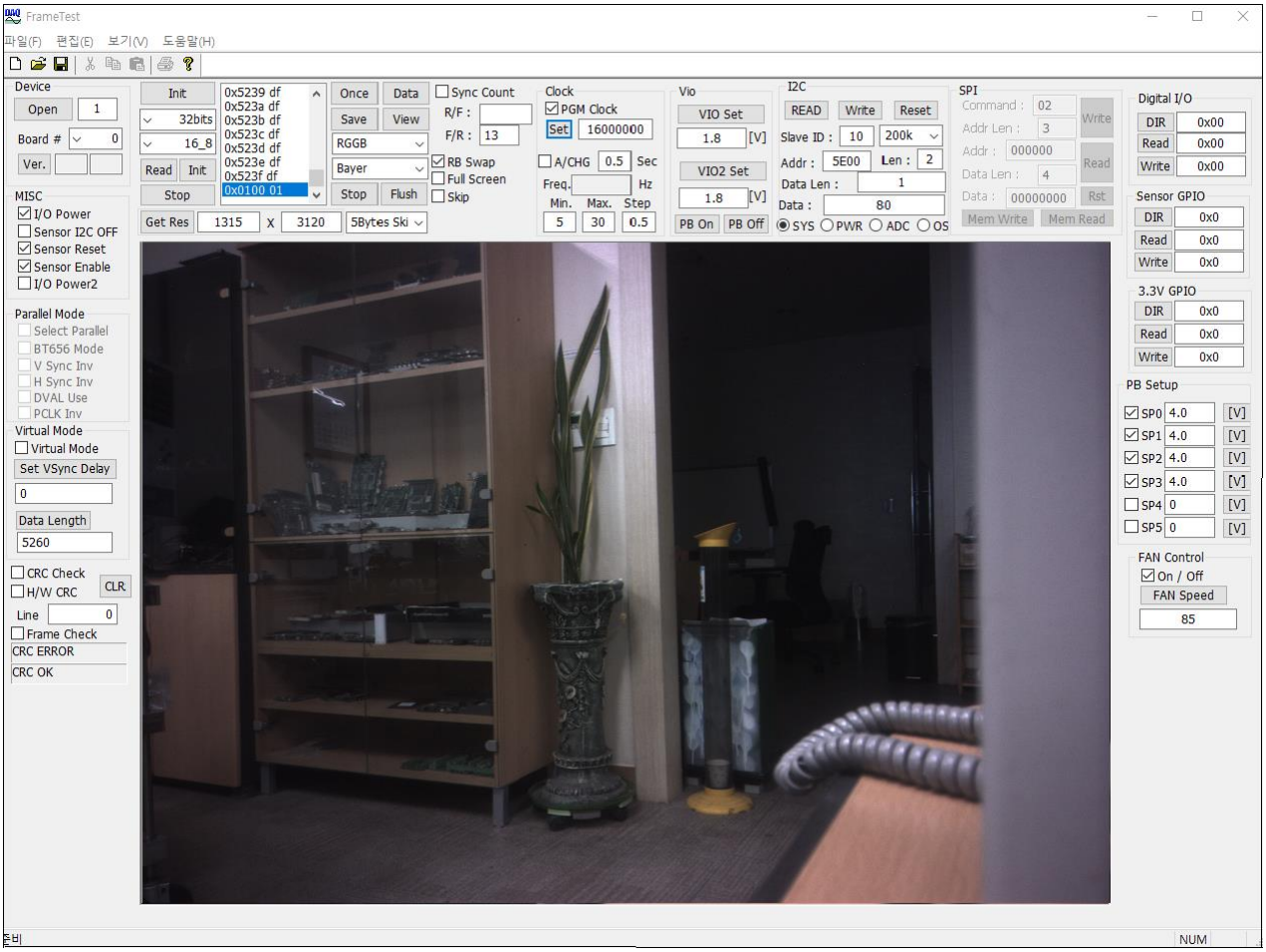


[Figure 3-7. SW3 Switch (Top View)]

4. 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 4-1. 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-FRM02.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.

The order of program execution to view the image is as follows.

(See: located in USB3-FRM02_Documents: \ Video \ USB3-FRM02_Execution.mp4)

① Device "Open" click

② Device "Init" click

③ Select pixel bit Select from "8, 16, 32bits"

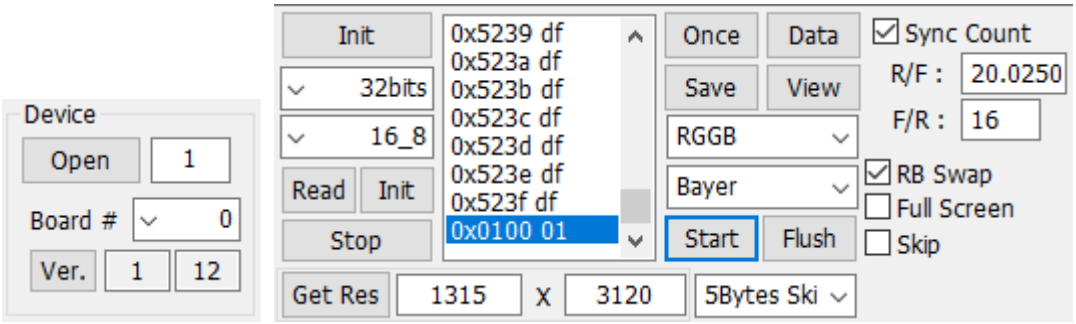
④ Select among Address_Data Bit 8_8, 8_16, 16_8, 16_16 among the contents of sensor initialization file

⑤ After reading sensor initialization file saved as "Read", "Go"

⑥ Image is automatically displayed. Get Res. The resolution

⑦ You can stop the video by clicking "Start / Stop" toggle button.

4.1 Functions related to Image Frame



(1) **" 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) **"Ver." button**

Get current FPGA and Firmware version.

(4) **" 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) **"Read" button**

Read the sensor initialization file. According to the above address_data size (8_8, 8_16, 16_8, 16_16), it is possible to send commands all at once to INI file or use I2C read / write line by line command. The ini file structure and description are as follows. The following example is an address 2-byte data 1-byte structure whose address_data size is 16_8.

.

Ex) sony13M_full.ini File Structure

```
[REGISTER]
    Slave    0x10   //change slave ID as Sensor
    SLEEP   100
    0x0103 01

    0x0300 01
    0x0301 00
    0x0302 38
    0x0303 00
    0x030a 00
    0x300f 11
    0x3010 01
    0x3011 76
    …….

    0x5404 00
    0x5405 80
    0x540c 05
    0x5b00 00
    0x5b01 00
    0x5b02 01
    0x5b03 ff
    0x5b04 02
```

(6) **" 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) **“Get Res.” button**

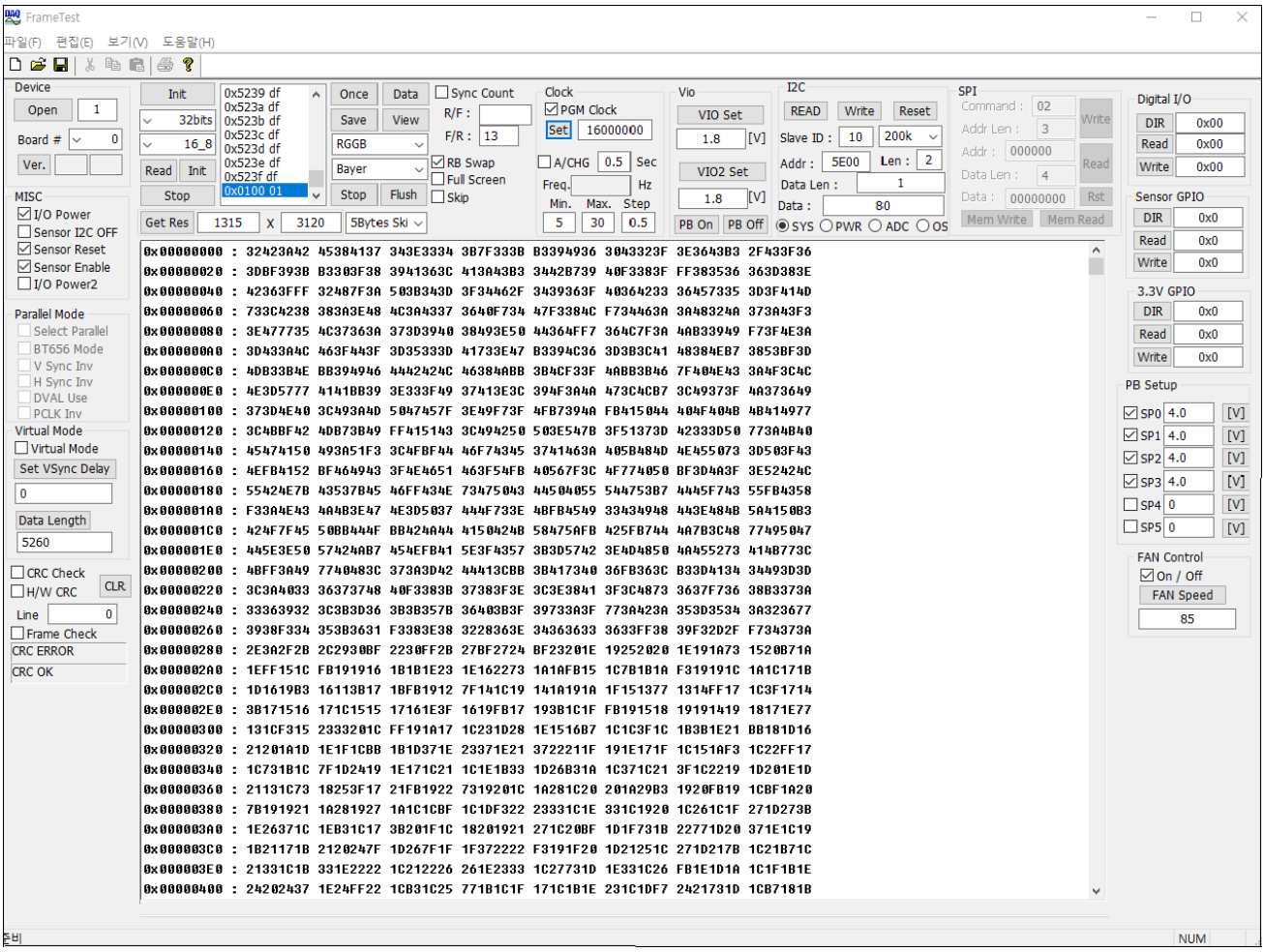
It shows the image resolution.

(9) **“Once” button**

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

(10) **“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 4-2. When Sample program “DATA’ is executed]

(11) **“Save” button**

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

(12) **“View” button**

Image Transfer begins to start.

(13) **"Input Bayer Mode" Selection**

Select from RGGB, BGGR, GRBG, and BGGR.

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

Select from Gray, YUV, Bayer.

(15) **"Start/Stop" toggle button**

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

(16) **"Flush" button**

Clear the LVDS buffer.

(17) **"No Skip" toggle**

"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.

"3'th Skip" toggle

When this button is selected, it goes beyond the 3th byte from the image data.

(18) **"Sync Count"**

"R/F" : Real Frame rates/sec

"F/R" : Frame rates/sec

(19) **"R/B Swap"**

If selected, Red and Blue are exchanged.

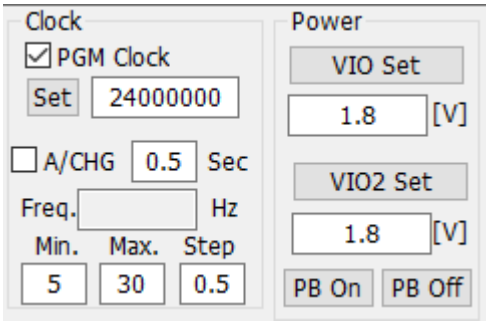
(20) **"Full screen" toggle**

It shows full screen when selected.

(21) **"Skip"**

It skips the screen display when selected.

4.2 Functions related to Clock & Power



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

You can select the frequency range of the master clock supplied to the sensor. The default is 24 MHz.

(2) **"Set"**

It sets the clock to 1039Hz ~ 68MHz.

(3) **"A/CHG" toggle**

If checked, the interval of the frequency set by Min. and Max. can be periodically set and tested according to the Step.

Example) In the above case, the frequency increases between 5 and 30 MHz in units of 0. 5 MHz, and the period increases by 0.5 x 1000 ms = 500 ms as the number next to A / CHG.

(4) **"VIO set"**

You can adjust the I/O level of the MIPI signals (ENB, Reset, MCLK, CNT0 to CNT3 on the J2 connector). (1.5 ~ 3.3V)

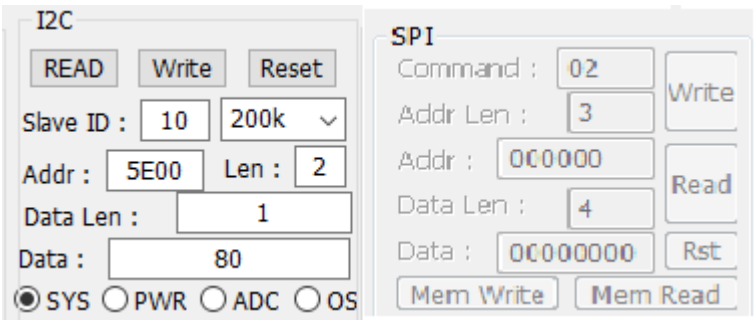
(5) **"VIO2 set"**

Parallel GPIO (Pin 27 ~ 34 of J1) I / O level can be adjusted. (1.5 ~ 3.3V)

(6) **"PB On/Pb Off"**

Turn the setting On/Off according to the SP (Sensor Power) value of PB Setup on the right screen.

4.3 Functions related to I2C/SPI



I2C of the system module is used.

(1) **"Read" Button**

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

(2) **"Write" Button**

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

(3) **"Reset" Button**

There is Initialization resource of I2C system.

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

100k to 1000k: I2C speed

" Addr " : Slave Register Address

"Len :" : Address Length

"Data Len :" : Data Length

"Data :" : Data that will transmit

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

(5) "SYS" : J2 connector SCL #7, SDA #9

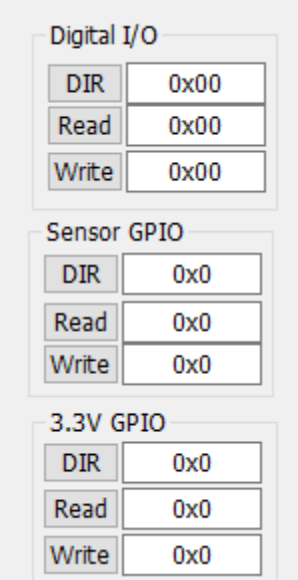
"PWR" : J1 connector SCL #11, SDA #12

"ADC" : J1 connector SCL #13, SDA #14 (feature pending)

"OS" : J1 connector SCL #17, SDA #18

P.S) SPI communication will be updated later.

4.4 Functions related to Digital IO/Sensor/3.3V GPIO



(1) Digital I/O **"DIR" button**

It sets whether to use the signal line of the General GPIO port of the J1 connector (27..34) as input or output. During operation, I/O Power2 should be checked. If the end bit is "0", it is an input, and if it is "1", it is an output.

(2) Digital I/O **"Read" button**

It reads the input value of General GPIO port of J1 connector (27..34). During operation, I/O Power2 should be checked.

(3) Digital I/O **"Write" button**

It outputs the desired value to the General GPIO port of the J1 connector (27..34). During operation, I/O Power2 should be checked.

(4) Sensor GPIO **"DIR" button**

It sets whether to use the Sensor GPIO (J2 connector CNT0..3) port as an input or output. If the end bit is "0", it is an input, and if it is "1", it is an output.

(5) Sensor GPIO **"Read" button**

It reads the input value of Sensor GPIO (J2 connector CNT0..3) port.

(6) Sensor GPIO **"Write" button**

It outputs desired value to Sensor GPIO (J2 connector CNT0..3) port.

(7) 3.3V GPIO **"DIR" button**

It sets whether to use the 3.3V GPIO (J1 connector 21..24) port as an input or output. If the end bit is "0", it is an input, and if it is "1", it is an output.

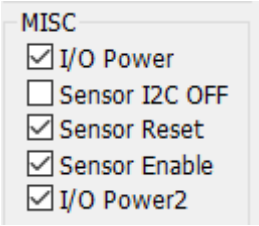
(5) 3.3V GPIO **"Read" button**

It reads the input value of 3.3V GPIO (J1 connector 21..24) port.

(6) 3.3V GPIO **"Write" button**

It outputs desired value to 3.3 GPIO (J1 connector 21..24) port.

4.5 Functions related to MISC



You can set and use different kinds of states.

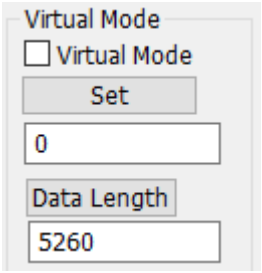
"I / O Power": Turns all signal operations On/Off.

"Sensor I2C Off": Sensor I2C operation On/Off.

"Sensor Reset": Set the reset output of the sensor from High to Low.

"Sensor Enable": Set the sensor's ENB(Enable) output from High to Low.

"I / O Power2": Use the General GPIO voltage of the J1 connector (27..34). You can set the voltage value with VIO2 Set.



"Virtual Mode" toggke

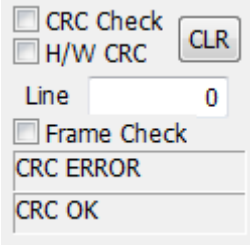
It selects MIPI Virtual Mode. You can give a delay of Vsync.

4.6 Functions related to CRC

Cyclic Redundancy Check (Cyclic Redundancy Check) refers to a method of determining a check value for checking whether there is an error in transmitted data when transmitting data.

CRC value is calculated and transmitted according to the given data value before data is transmitted, and the CRC value is calculated again using the received data value after the data transmission is completed. Subsequently, the two values are compared, and if these two values are different, it can be seen that an error was additionally transmitted due to noise in the data transmission process.

The process of calculating the CRC can be considered a hardware method and a software method. When it comes to hardware, it tends to use serial data. In this case, the CRC gets the output for the bitwise input. Creating a logic circuit simplifies it. However, parallel data (octet unit, 8 bits) is used as it goes to a higher layer. In this case, it is mainly calculated in bytes by software access. Below are the test items related to functions.



"CRC Check": CRC Check On / Off.

When checking, check sum 2 bytes data is added at the end of each frame or line according to the selection of "Frame Check".

"H / W CRC": H/W ⇔ S/W CRC On / Off.

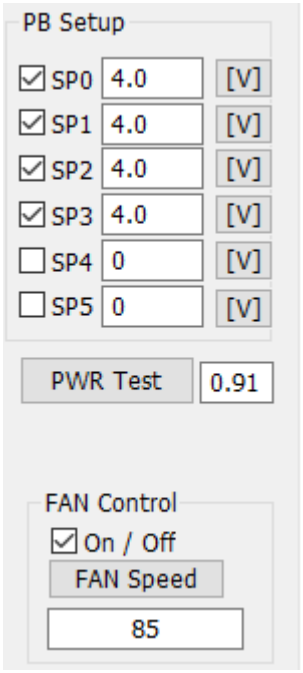
CRC processing method, check hardware processing

"Frame Check": Line CRC ⇔ FRAME CRC Check conversion.

(The line can be selected in the Edit Box at the top of the Frame Check)

"CLR": Initialize the CRC ERROR and CRC OK values below.

4.7 Functions related to PB Setup/FAN Control



(3) **"PB Setup"**

SP (Sensor Power) 0..3 voltage (0.9 ~ 4.1V) can be controlled. (SP4..5 Reserve).

(4) FAN Control **"On/Off"**

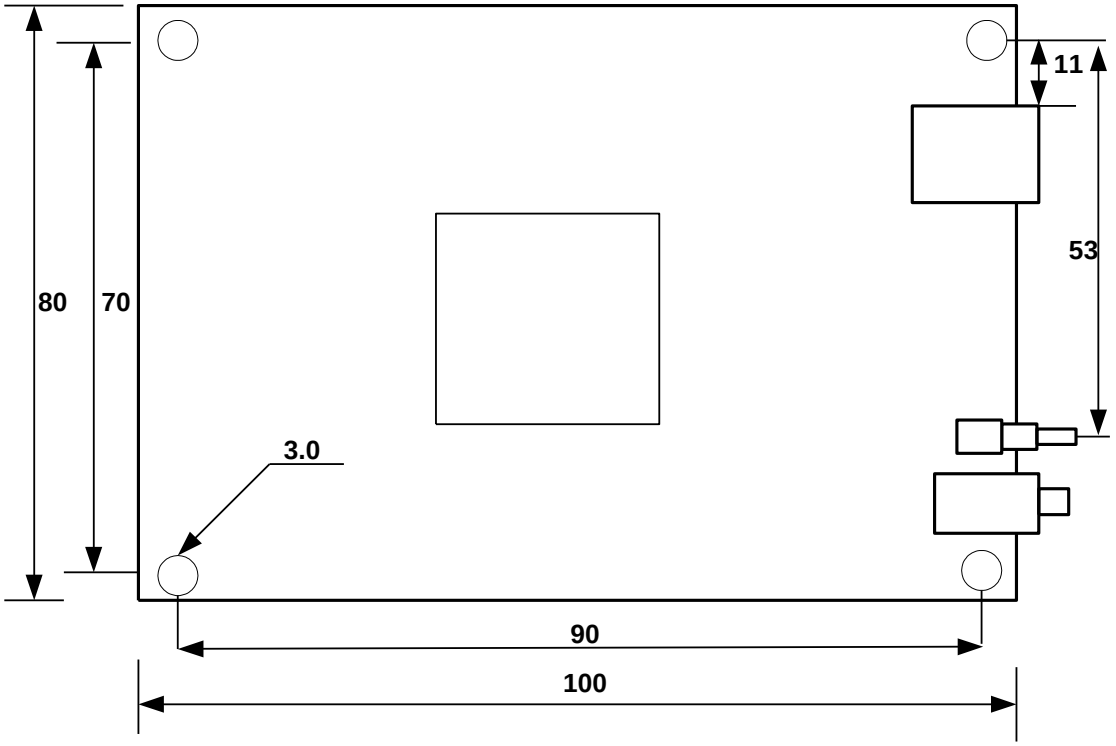
Functions as a FAN On/Off switch. The speed (0 ~ 255) can be controlled.

Appendix

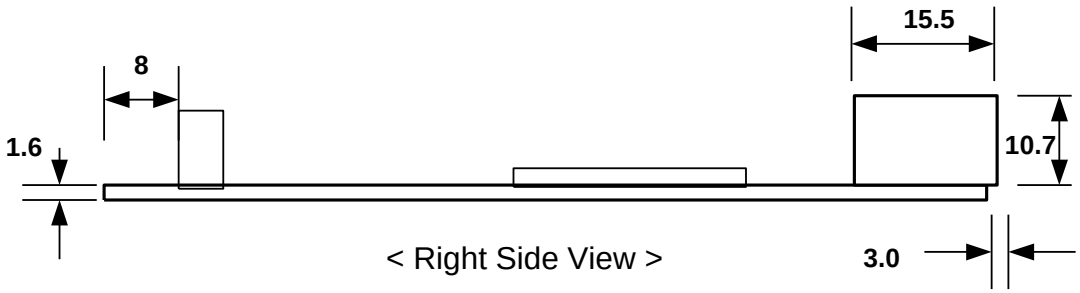
A.1 Physical Dimension.

Approximate dimensions of the board are shown below.

(Please contact the person in charge for detailed dimensions.)



< Top View >



< Right Side View >

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. USB3_Framegrabber_Installation-ver1.1

-- DAQ system

4. AN342 USB3-FRM02 API ver1.2

-- DAQ system