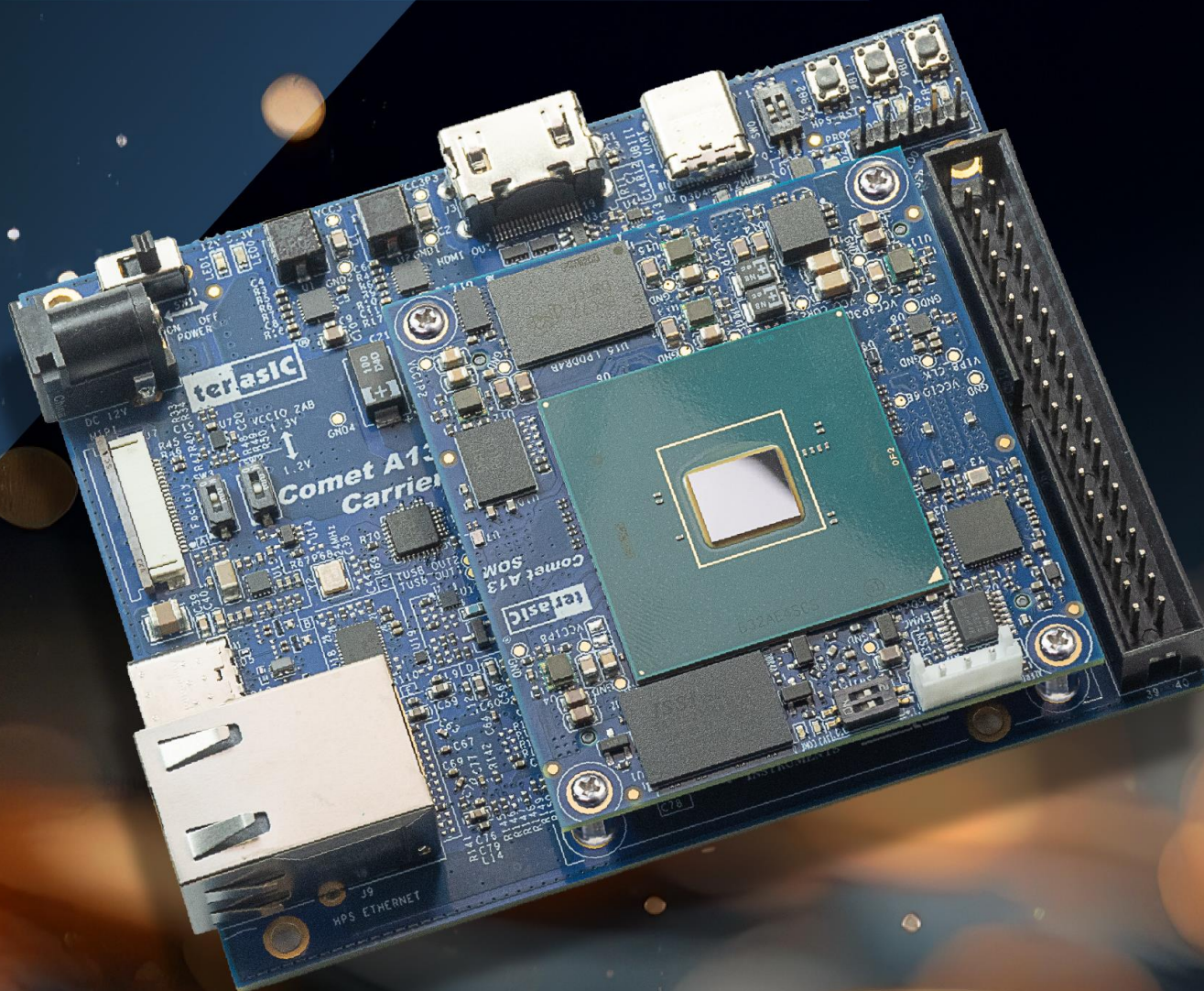


# Comet-A13 SOM Evaluation Kit



## Demonstration Manual

FPGA

## Contents

<b>Chapter 1</b>	<b>Overview .....</b>	<b>3</b>
<b>Chapter 2</b>	<b>Examples For FPGA .....</b>	<b>4</b>
2.1	Comet A13 Factory Configuration .....	4
2.2	Basic Nios V control demo for Temperature/ Power/ Fan .....	6
2.3	Board Information IP .....	9
2.4	HDMI_out_RTL in Verilog .....	13
2.5	MIPI_RTL .....	15
2.6	RTL_LPDDR4_AXI4_Test .....	18
2.7	LPDDR4_Test_NiosV Test .....	20
<b>Chapter 3</b>	<b>Additional Information .....</b>	<b>24</b>
3.1	Getting Help .....	24



# Chapter 1

## *Overview*

---

**T**his Manual will introduce the various application demonstrations on **Comet A13 SoM Evaluation Kit**. These demonstrations cover most of the interfaces on the board. Let users familiarize using these interfaces of the board. Demonstrations according to FPGA fabrics are divided into three categories:

- **Pure use of FPGA fabric resources (Chapter 2)**

Finally, to complete the following demonstration, user needs to install the following software in the computer:

- **[Intel Quartus® Prime Pro Edition Software Version 25.3.1](#) or later.**

**Note:** To run the demo batch file with the Nios V CPU of the demonstration on Windows system, user need to install the Ashling RiscFree IDE together with Quartus software, then you can run the batch file.

# Chapter 2

## *Examples For FPGA*

This chapter provides examples of advanced designs implemented by RTL or Qsys on the **Comet A13 SoM Evaluation Kit**. These reference designs cover the features of peripherals connected to the FPGA, such as LPDDR4, temperature monitor, Power monitor, HDMI TX and MIPI. All the associated files can be found in the directory **\Demonstrations\FPGA** of Comet A13 SoM Evaluation Kit System CD.

### 2.1 Comet A13 Factory Configuration

The Comet A13 evaluation board has a default configuration bitstream pre-programmed, which demonstrates some of the basic features on board. The setup required for this demonstration and the location of its files are shown below.

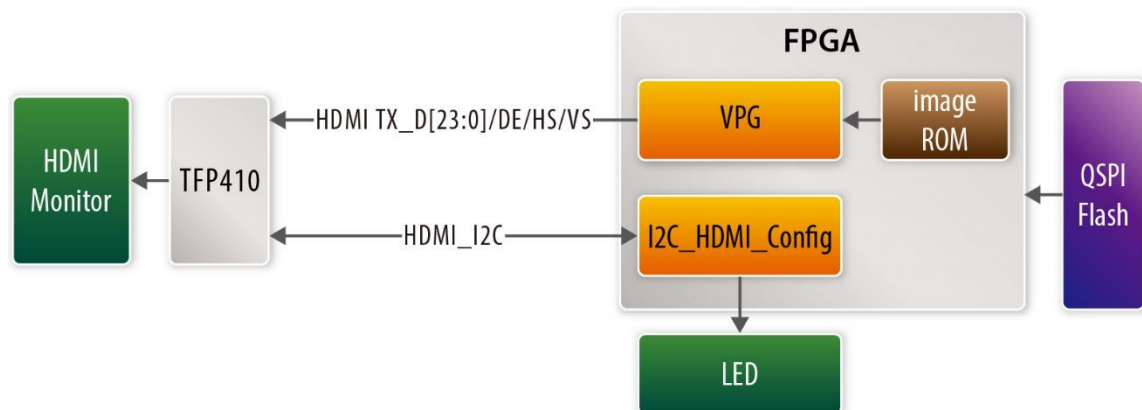


Figure 2-1 Block Diagram of the default demo

#### ■ Design Tools

- Quartus Prime 25.3.1 Pro Edition

#### ■ Demonstration Source Code

- Quartus Project directory: \Demonstrations\HDMI\_ASx4
- Bitstream used: golden\_top.sof or flash.jic

#### ■ Demonstration Batch File

Demo Batch File Folder: \Demonstrations\HDMI\_ASx4\demo\_batch

The demo batch file includes following files:

- Demo Batch File : test.bat



- FPGA Configure File: golden\_top.sof

## ■ Dual Image Folder

Dual image folder: \Demonstrations\HDMI\_ASx4\Dual\_Image

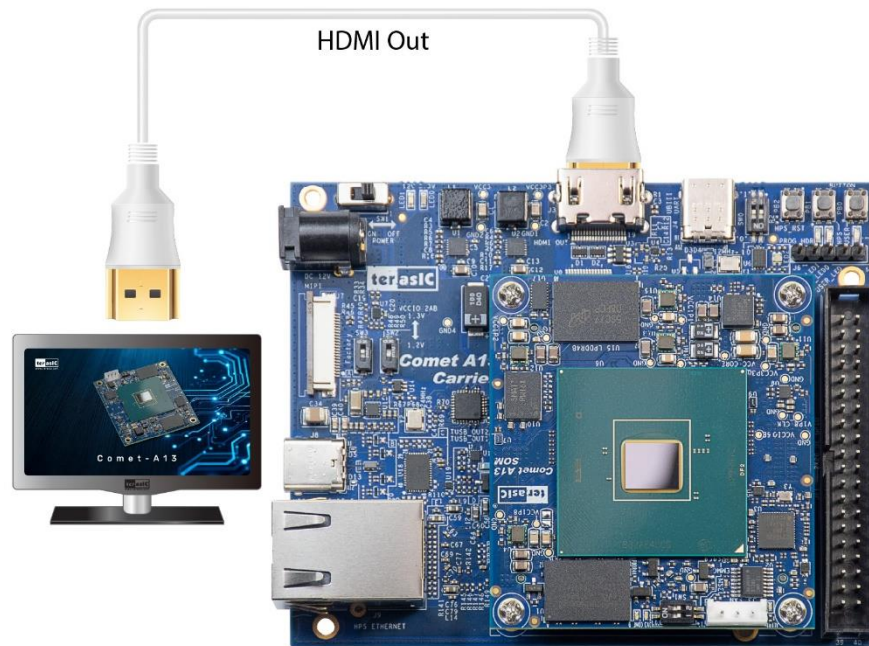
The demo batch file includes following files:

- Convert JIC Batch File: convert\_flash\_image\_qspi.bat
- Flash Program Batch File: flash\_program.bat
- Flash Erase Batch File: flash\_erase.bat
- Flash Address Map File: flash\_jic.map
- Flash Layout File: comet\_a13.pfg
- Factory Image FPGA Configure File: factory.sof (default: \Demonstrations\FPGA\HDMI\_ASx4\demo\_batch\golden\_top.sof)
- User Image FPGA Configure File: user.sof (default: \Demonstrations\SoC\_FPGA\GHRD\output\_files\golden\_top\_hps.sof)
- Flash Image: flash.jic

## ■ Demonstration Setup and Instructions

- Ensure the MSEL switch (SW1) on the SoM module is set to 0 (AS Mode).
- Set the Factory Image Switch (SW3) on the Carrier Board to the **ON** position.
- Connect a USB Type-C cable from the host PC to the USB Blaster III connector (J4) on the Carrier Board.
- Connect an HDMI monitor to the HDMI TX connector (J3) on the Carrier Board.
- Connect the 12V DC adapter to the DC Power Jack (J5) of the Carrier Board and power on it.
- You should observe the user LEDs blinking and the HDMI monitor displaying the Comet A13 SoM board image.
- If necessary (that is, if the default factory configuration is not currently stored in the QSPI device), download the bitstream to the board via the JTAG interface.
- The demo\_batch\test.bat is executed to program .sof to the FPGA. The Dual\_Image\flash\_program.bat is executed to program the .jic to the flash. The Dual\_Image\convert\_flash\_image\_qspi.bat is executed to convert .sof to .jic file.
- If users want to program a new design into the QSPI flash device, copy the new .sof file to the Dual\_Image folder:
  - If SW3 is set to **ON** (Factory Image), rename the .sof file to factory.sof.
  - If SW3 is set to **OFF** (User Image), rename the .sof file to user.sof.
- Then execute convert\_flash\_image\_qspi.bat to generate the .jic file and

flash\_program.bat to program the flash device.



**Figure 2-2 Setup for the default demo**

## **2.2 Basic Nios V control demo for Temperature/ Power/ Fan**

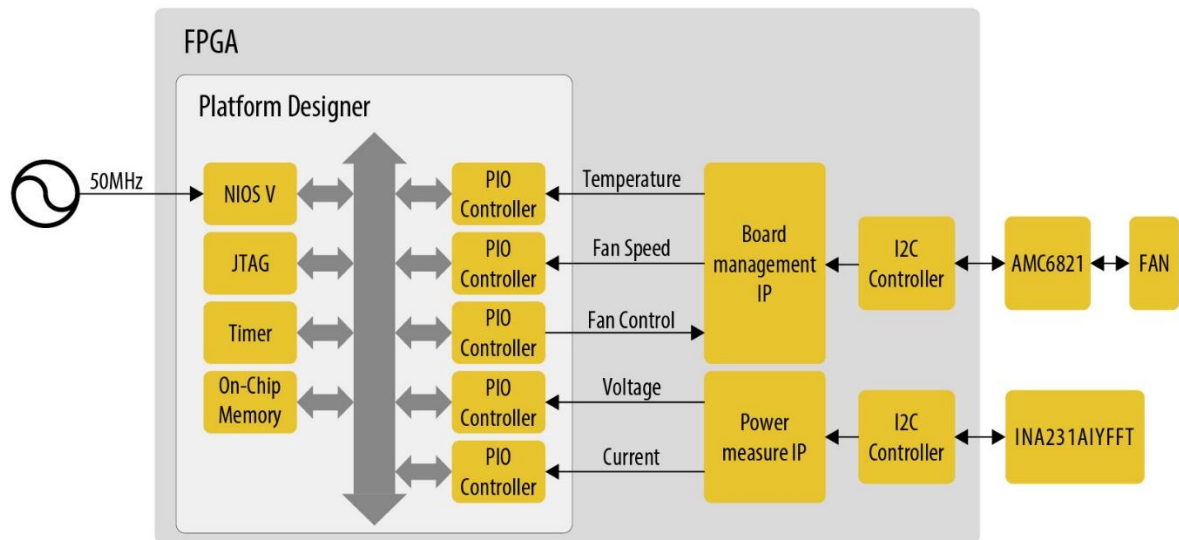
This demonstration shows how to use the Nios V processor to measure the power consumption based on the built-in power measure circuit. The demonstration also includes a function of monitoring system temperature with the on-board temperature sensor and monitoring fan rotation speed.

### **■ System Block Diagram**

**Figure 2-3** shows the system block diagram of this demonstration. The temperature sensor and fan controller are connected to the Agilex 5 FPGA via the FAN\_I2C interface and controlled by the BOARD\_MANAGEMENT IP. The 12V input power monitor is connected via the PM\_I2C interface and managed by the POWER\_MEASURE IP.

In the Agilex FPGA, the BOARD\_MANAGEMENT IP communicates with the AMC6821 temperature/fan controller via I2C to read temperature data and control fan speed. The POWER\_MEASURE IP communicates with the INA231 power monitor IC to read voltage and current information. The Nios V system reads this information through PIO

controllers.



**Figure 2-3 Block Diagram of the Nios V Basic Demonstration**

The system provides a menu in command line window, as shown in **Figure 2-4** to provide an interactive interface. With the menu, users can perform the test for the board info sensor. Note, pressing 'ENTER' should be followed with the choice number.

```

C:\WINDOWS\system32\cmd.exe
Loading section .entry, size 0x20 lma 0x0
Loading section .exceptions, size 0x304 lma 0x20
Loading section .text, size 0x22108 lma 0x324
Loading section .rodata, size 0x1580 lma 0x22430
Loading section .rwddata, size 0x1a00 lma 0x253b0
Start address 0x00000774, load size 152492
Transfer rate: 499 KB/sec, 21784 bytes/write.
[Inferior 1 (Remote target) detached]
juart-terminal: connected to hardware target using JTAG UART on cable
juart-terminal: "Comet A13 [USB-1]", device 1, instance 0
juart-terminal: (Use the IDE stop button or Ctrl-C to terminate)

===== Board_Info_NiosV Demo Program =====
[0] System Info
[1] Fan Control
Input your choice:0
=== BOARD INFO TEST ===
FPGA Temperature: 22 C
Board Temperature: 26 C
Input Voltage: 11928 mV (11.93 V)
Input Current: 302 mA (0.30 A)
Power Consumption: 3602 mW (3.60 W)

[BOARD INFO]
Result=PASS
System Info Test:PASS
===== Board_Info_NiosV Demo Program =====
[0] System Info
[1] Fan Control
Input your choice:

```

**Figure 2-4 Menu of Demo Program**

In board info test, the program will display local temperature, remote temperature, 12V input power monitor and fan rotation speed. The remote temperature is the FPGA temperature, and the local temperature is the board temperature where the temperature

sensor located. The board also provides circuitry to monitor important voltages, currents and power consumption in real time.

### ■ **Demonstration File Location**

- Hardware project directory: Board\_Info\_NiosV
- Bitstream used: golden\_top.sof
- Software project directory: Board\_Info\_NiosV\software
- Demo batch file: Board\_Info\_NiosV\demo\_batch\test.bat

### ■ **Install Ashling RiscFree IDE**

Before executing this demo with RISC-V core, users need to install Ashling RiscFree IDE for Intel® FPGAs to ensure that this batch file can be executed correctly. The file name should be *RiscFreeSetup-<Quartus Version>-windows.exe*. Users can find it under the [Quartus Pro download page](#) (Individual Files tab).

### ■ **Demonstration Setup and Instructions**

1. Make sure Quartus Pro v25.3.1 is installed on the Host PC.
2. Power on the Comet A13 evaluation board.
3. Use the Type-C USB Cable to connect your PC and the FPGA board and install USB Blaster III driver if necessary.
4. Execute the demo batch file “test.bat” under the batch file folder: Board\_Info\_NiosV\demo\_batch.
5. After the Nios V program is downloaded and executed successfully, a prompt message will be displayed in command line window.
6. For temperature, power monitor and fan test, please input key ‘0’ and press ‘Enter’ in the command line window, as shown in **Figure 2-5**.



```
C:\WINDOWS\system32\cmd.exe
juart-terminal: (Use the IDE stop button or Ctrl-C to terminate)

===== Board_Info_NiosV Demo Program =====
[0] System Info
[1] Fan Control
Input your choice:0
=== BOARD INFO TEST ===
FPGA Temperature: 22 C
Board Temperature: 26 C
Input Voltage: 11928 mV (11.93 V)
Input Current: 302 mA (0.30 A)
Power Consumption: 3602 mW (3.60 W)

[BOARD INFO]
Result=PASS
System Info Test:PASS
===== Board_Info_NiosV Demo Program =====
[0] System Info
[1] Fan Control
Input your choice:1
=== FAN CONTROL TEST ===
Current Fan Speed: 4889 RPM

[FAN CONTROL]
Result=PASS
Fan Control Test:PASS
===== Board_Info_NiosV Demo Program =====
[0] System Info
[1] Fan Control
Input your choice:
```

Figure 2-5 Board Info Demo

## 2.3 Board Information IP

This section will introduce two IPs which can be placed in the Agilex FPGA and allows users to obtain board status information such as temperature status and fan speed and VIN's voltage-current on the Comet A13 board.

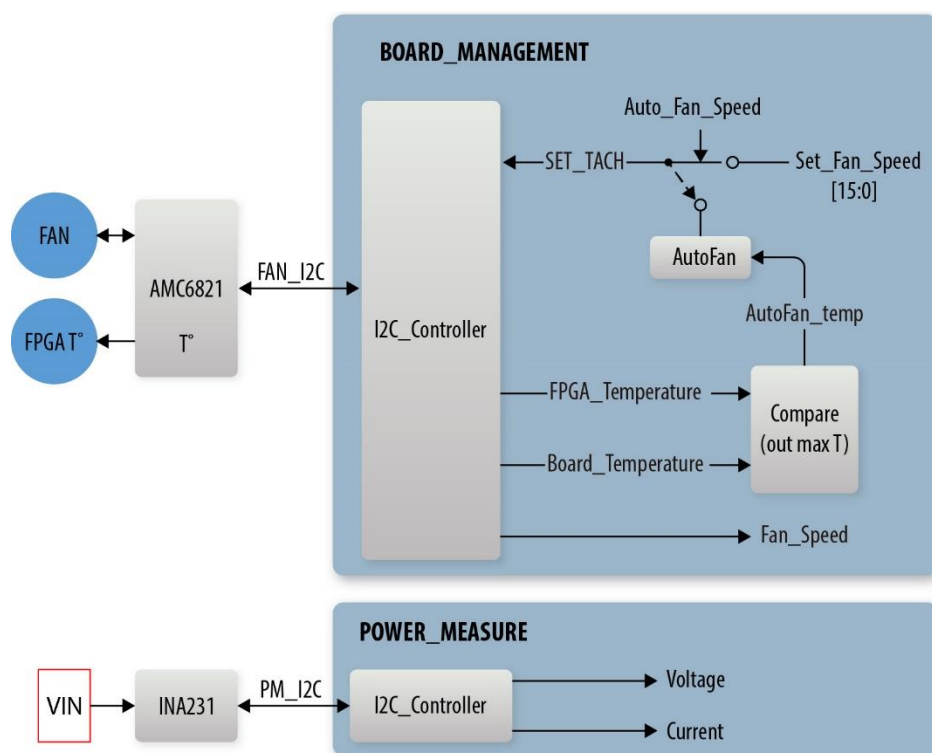
The Comet A13 board provides several sensors to monitor the status of the board, such as FPGA/Board temperature, and fan speed status. The system primarily uses the I<sup>2</sup>C bus (FAN\_I2C) to communicate with on-board devices via the FPGA's I<sup>2</sup>C controller. It reads temperature data from the AMC6821(I<sup>2</sup>C slave address: 0x5C), including FPGA temperature and local-board temperature.

The system also interfaces with the AMC6821 to both configure the fan speed and read the current fan speed.

An automatic fan speed control function (AutoFan) adjusts the fan speed based on the temperature readings. This function can be enabled or disabled via the Auto\_Fan\_Speed (0 or 1):

- When set to 0, the fan speed is controlled manually by the user.
- When set to 1, the fan speed is automatically determined by the AutoFan logic and sent to the I2C controller

The block diagram of the design is as shown in **Figure 2-6**.



**Figure 2-6 Block diagram of the fan speed control demonstration**

User can place a board information IP (BOARD\_MANAGEMENT.v) provided by Terasic in the Agilex FPGA, the board status can be obtained via I2C interface from the FPGA and output to user logic. The **POWER\_MEASURE** module connects to the **INA231** power monitor IC on the Carrier Board via another I2C bus (**PM\_I2C**) to read the voltage and current of **VIN (12V or 5V)**.

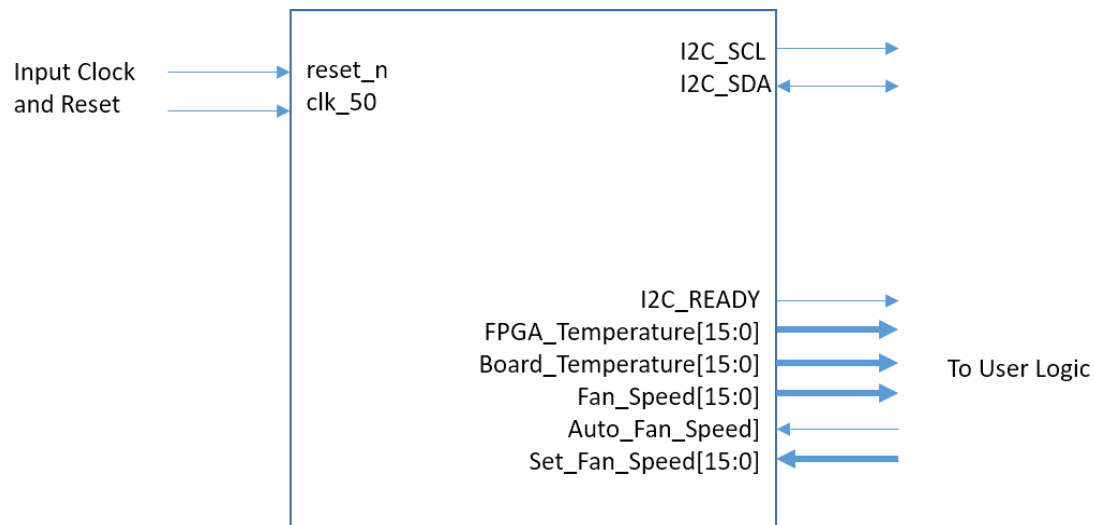
The board information IP can be obtained from the following path in the system CD:  
**Demonstration/FPGA/Board\_info\_RTL/board\_management\_ip/BOARD\_MANAGEMENT.v**

The POWER\_MEASURE IP can be obtained from the following path in the system CD  
**Demonstration/FPGA/Board\_info\_RTL/board\_management\_ip/POWER\_MEASURE.v**

**Figure 2-7** and shows the input and output pins of the board information IP. Detailed pin descriptions and functions can be obtained from **Table 2-1** (Board Information) and **Table 2-2** (POWER\_MEASURE). The user only needs to provide the IP 50Mhz clock and the reset control signal. The IP will automatically communicate with to get the board

status value via the I2C interface. When the logic level of the I2C\_READY signal is from low to high, it means that the board status has been updated and can be used.

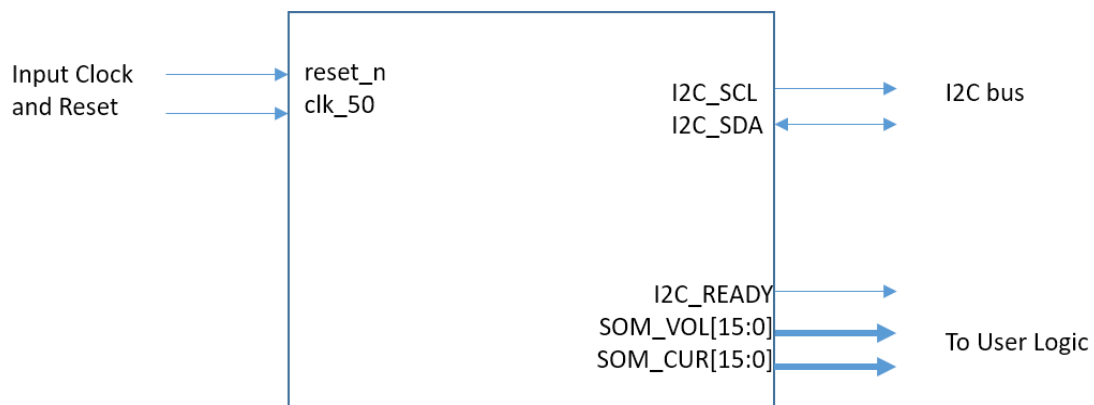
Finally, **Figure 2-9** shows the status of the IP during execution.



**Figure 2-7 Pin out of the board information IP**

**Table 2-1 Board information IP input and output ports**

Port Name	Direction	Width(Bit)	Description
clk_50	Input	1	Clock input for IP, please input 50Mhz clock.
reset_n	Input	1	Reset signal for IP, reset all logic.
I2C_SDA	BIR	1	Master I2C data . Please connect this signal to the <b>FAN_I2C_SDA</b> pin.
I2C_SCL	Output	1	Master I2C clock, I2C master output to salve. Please connect this signal to the <b>FAN_I2C_SCL</b> pin.
I2C_READY	Output	1	Information valid, logic high indicates board status updated ready.
Fan_Speed	Output	16	First fan speed of the board. The unit of the output value is RPM.
Board_Temperature	Output	16	First ambient temperature of the development board. The unit of the output value is Celsius.
FPGA_Temperature	Output	16	Core FPGA temperature of the development board. The unit of the output value is Celsius.
Auto_Fan_Speed	Input	1	1: enable auto speed control.0: user specify fan speed
Set_Fan_Speed	Input	16	Input rotational speed (unit: RPM)



**Figure 2-8 Pin out of the POWER\_MEASURE IP**

**Table 2-2 POWER\_MEASURE IP input and output ports**

Port Name	Direction	Width(Bit)	Description
clk_50	Input	1	Clock input for IP, please input 50Mhz clock.
reset_n	Input	1	Reset signal for IP, reset all logic.
I2C_SDA	BIR	1	Master I2C data . Please connect this signal to the <b>PM_I2C_SDA</b> pin.
I2C_SCL	Output	1	Master I2C clock, I2C master output to salve. Please connect this signal to the <b>PM_I2C_SCL</b> pin.
I2C_READY	Output	1	Information valid, logic high indicates board status updated ready.
SOM_VOL	Output	16	VIN (12V or 5V) voltage of the carrier board. The unit of the output value is mV.
SOM_CUR	Output	16	VIN current of the board. The unit of the output value is mA.

u_BOARD_MANAGEMENT[FPGA_Temperature[15..0]]	28
u_BOARD_MANAGEMENT[Board_Temperature[15..0]]	35
u_BOARD_MANAGEMENT[Fan_Speed[15..0]]	4827
u_BOARD_MANAGEMENT[I2C_READY]	
u_POWER_MEASURE[SOM_VOL[15..0]]	12090
u_POWER_MEASURE[SOM_CUR[15..0]]	287
u_POWER_MEASURE[I2C_READY]	

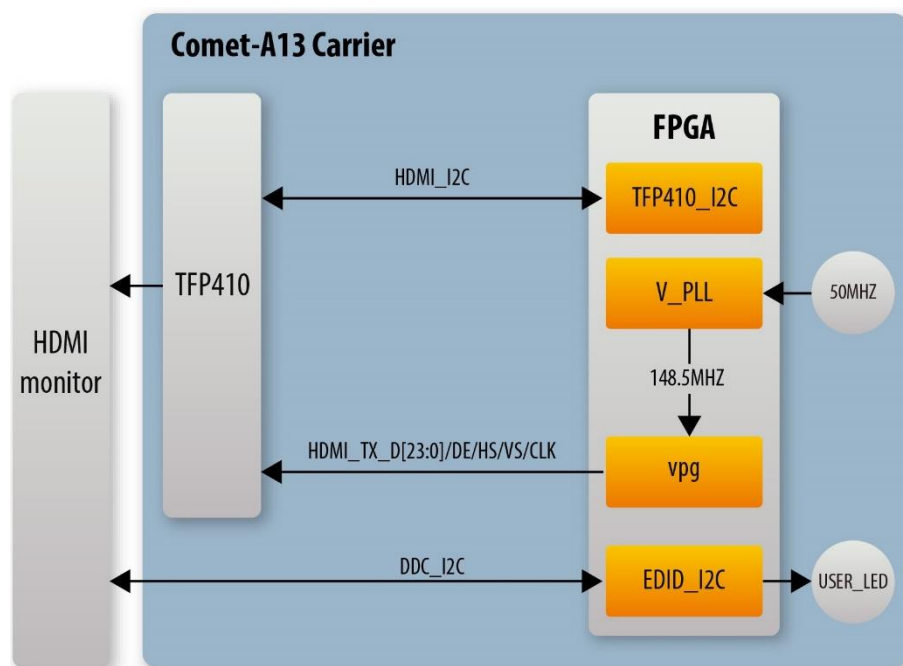
**Figure 2-9 Waveform of the board status output**

## 2.4 HDMI\_out\_RTL in Verilog

Comet A13 evaluation board system CD offers another HDMI (DVI 1.0-compliant) test with its test code written in Verilog HDL. That is, the output test for the TFP410 IC includes I2C configuration and the ability to output 1080p@60 with the specified Color Bar.

### ■ System Block Diagram

**Figure 2-10** shows the function block diagram of this demonstration. The VPG (video pattern generator) uses 50MHz in by PLL to generate 148.5MHz to do input. The VPG generates a 1080p@60 video timing color bar, which is sent to the TFP410 and then displayed on the HDMI monitor.



**Figure 2-10 Block Diagram of the HDMI TX Demonstration**

For the TFP410 to output video, the TFP410\_I2C module must be used to configure the device via I2C by setting bit 0 of register 0x08 (CTL\_1\_MODE) to 1, enabling normal operation.

The EDID\_I2C module reads the EDID RAM addresses 0x00 to 0x07 from the monitor, with the values being 0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, and 0x00 respectively. The USER\_LED lights up.

### ■ Design Tools

- Quartus Prime 25.3.1 Pro Edition



## ■ Demonstration Source Code

- Quartus Project directory: HDMI\_out\_RTL
- Bitstream used: golden\_top.sof

## ■ Demonstration Batch File

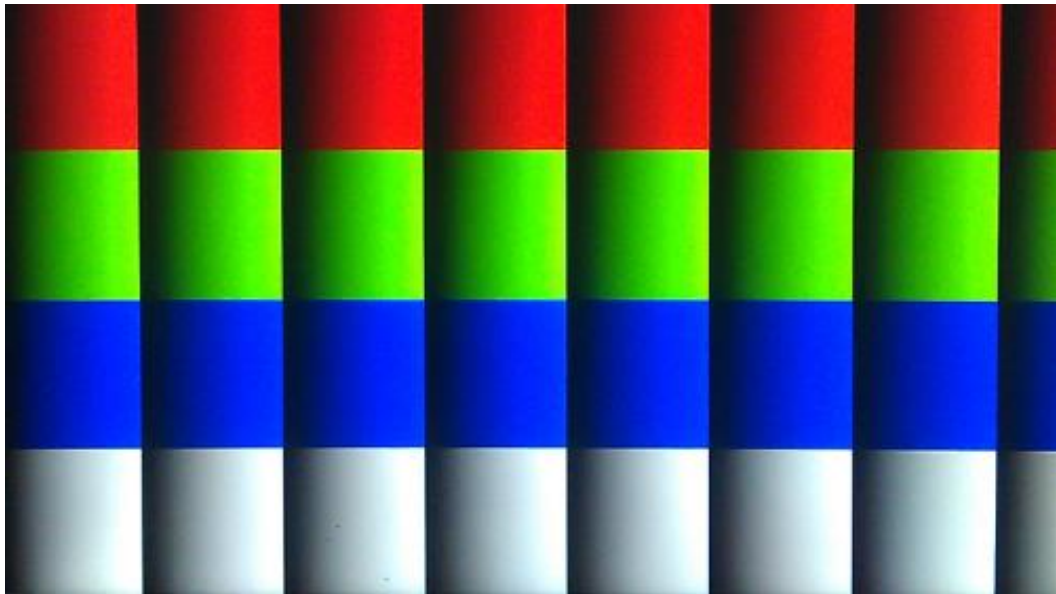
Demo Batch File Folder: HDMI\_out\_RTL\demo\_batch.

The demo batch file includes following files:

- Demo Batch File : test.bat
- FPGA Configure File: golden\_top.sof

## ■ Demonstration Setup and Instructions

- Make sure both Quartus Pro and UBIII driver are installed on the host PC.
- Connect the HDMI cable to J3 (HDMI Out) on the Comet A13 Carrier board.
- Use the Type-C USB Cable to connect your PC and the FPGA board.
- Power on the Comet A13 Evaluation board.
- Execute the demo batch file “ test.bat” from the directory HDMI\_out\_RTL\demo\_batch.
- 1080p@60 color bar should be visible on the monitor, as shown in **Figure 2-11**.



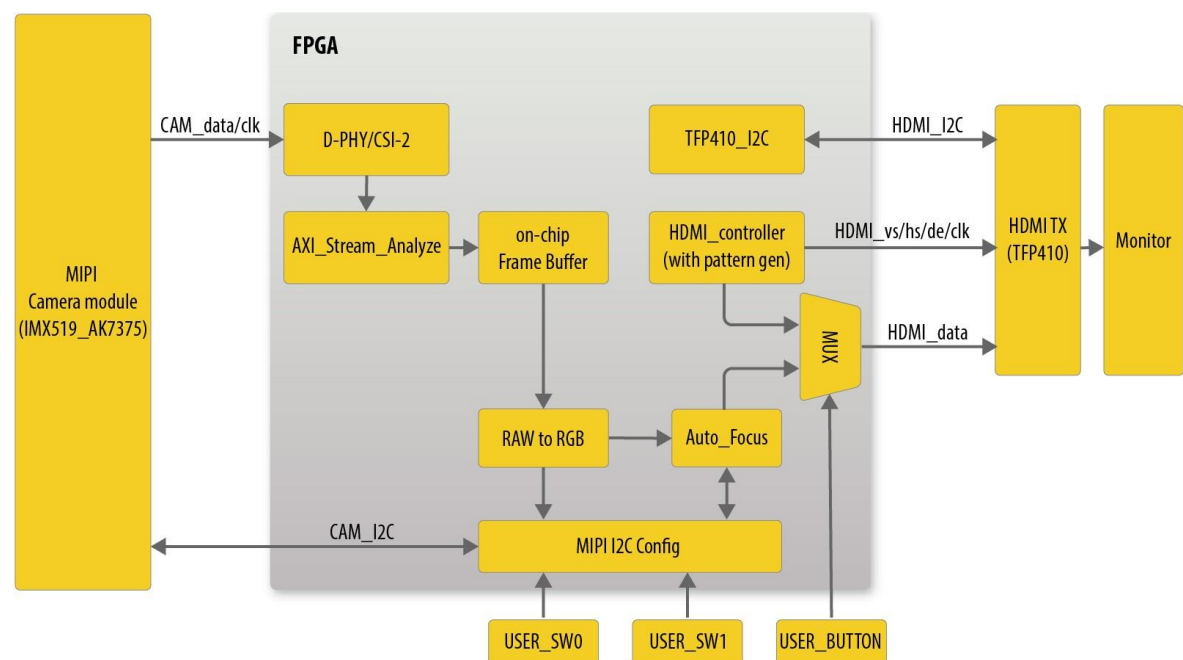
**Figure 2-11 The video pattern in the HDMI TX demonstration**

## 2.5 MIPI\_RTL

This section provides instructions on how to store Camera capturing image (480P@60Hz) in a larger on-chip memory (Frame-Buffer), this Memory is expected to store up to one Frame image, and how to extract Frame-Buffer address data to convert RAW data to RGB data, and output the RGB data to LCD with 480P@60Hz timing.

### ■ System Block Diagram

**Figure 2-12** Shows the Function block diagram of MIPI-Camera demonstration. This design block is one dual-port-ram Control module can control on chip Memory and read/write image data. Camera module uses 2 data lanes and 1 clock lane to input signals into the FPGA. Through Intel D-PHY and CSI-2, the signals are decoded into AXI-Stream formatted video data, then converted into the required raw data format. The raw data is written into on-chip RAM first. After finishing writing a Frame, ON-CHIP\_FRAM module will read out the data from dual-port-ram to RAW to RGB to convert RAW data to RGB data. The RGB data will output along with the signal timing generated by HDMI\_Controller to LCD. In the block, other module (for example, Auto\_Focus, MIPI I2C Config, TFP410\_I2C Config) function instructions and SW/BUTTON operation. All module functions are described below:



**Figure 2-12 Block diagram of the MIPI camera design with on-chip memory**

**D-PHY/CSI-2(csi2\_dphy\_sys.qsys):** This module integrates Intel D-PHY and CSI-2 IPs into a single QSYS system, which converts a 2-data-lane / 1-clock-lane MIPI stream

into an AXI-Stream.

**AXI\_Stream\_Analyze(AXI\_Stream\_Analyze.v):** This module further parses the AXI-Stream, including raw data extraction, overall frame pixel counting, and frame start indication .

**on-chip Frame Buffer(FRAME\_RAM.ip):** This module is one dual-port-ram controller can control 640 x 480 x10bit on-chip ram and read/write image data

**RAW to RGB(RAW2RGB\_J.v):** This module converts RAW data to RGB data and provides real-time RED/BLUE gain values for the entire frame, which are intended to be used as white balance configuration parameters for the MIPI I2C Config block.

**HDMI\_controller(vpg.v):** The LCD signal timing generator can generate 640p@60Hz timing and provide a color bar video pattern output.

**Auto\_Focus(FOCUS\_ADJ.v):** This module is doing the current image high frequency component statistic. When the VCM drives the camera lens' movement, a real-time statistic of image high-frequency sum will be done in every step of the moving. Finally, the lens will move to a position which has the largest number of high frequency to complete the automatic focus operation.

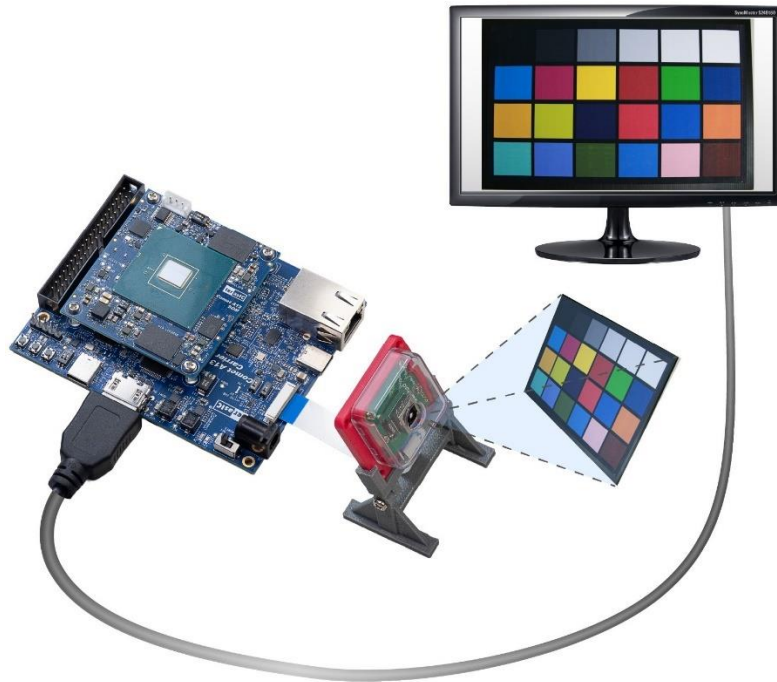
**MIPI I2C Config (IMX519\_AK7375.v/ IMX519\_AK7375\_ctl.v):** This block includes the MIPI bridge I2C and Camera I2C setting controller, such as set to output 720P@60Hz timing. It mainly writes I2C corresponding parameters to MIPI-BRIDGE IC register and Camera Sensor IC register respectively through their own I2C buses. MIPI\_I2C bus is used to write MIPI BRIDGE I2C (I2C Slave Address = 0x34.), CAMERA\_I2C bus is used to write Camera Sensor (IC Slave Address = 0x18).

**TFP410\_I2C Config(TFP410\_I2C.v):** This module will set TFP410(HDMI TX) I2C setting controller, set register respectively through own I2C buses. (I2C Slave Address = 0x78)

We provide the demonstrations on Comet A13. The following are the descriptions of the platforms' set up, as well as the test steps.

## ■ Hardware Setup

The hardware connecting and setup is shown in figure below.



**Figure 2-13 Comet A13 MIP\_RTL demo hardware setting up**

## ■ Design Tools

- Quartus Prime 25.3.1 Pro Edition

## ■ Demonstration Source Code

- Quartus Project directory: /Demonstration/FPGA/MIPI\_RTL
- Bitstream used: golden\_top\_top.sof

## ■ Demonstration Batch File

- Demo Batch File Folder: MIPI\_RTL/demo\_batch

## ■ Demonstration Setup

- Connect J4 on the Carrier board to the host PC using a Type-C cable, and install the UBI1 driver if necessary.
- Connect the MIPI camera to J7 on the Carrier board.
- Connect an HDMI monitor to the J3 (HDMI port) on the Carrier board.
- Plug the 12V adapter into the Comet-A13 carrier board, and Power on the Comet-A13 board.
- Execute the demo batch file test.bat from the directory\MIPI\_RTL\demo\_batch.
- USER\_LED light up, stand the settings of TFP410(HDMI TX) IC and Camera-module I2C are completed.

- Camera-module capturing image displays on HDMI monitor.
- **Table 2-3** Summarizes the functional keys and details of each LED status.

**Table 2-3 The functional keys of the Comet A13 MIPI\_RTL demonstration**

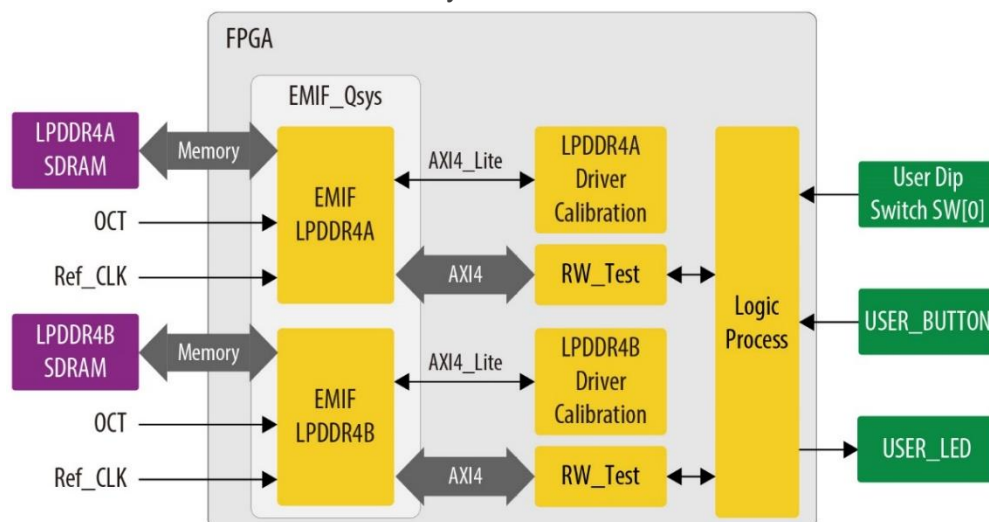
Name	Description
USER_LED	Lights up when the I2C configuration of TFP410 (HDMI TX) and the MIPI decoder is successful
USER_BUTTON	Pressing it displays the color bar, releasing it shows the camera image
USER_SW0	Toggling from 0 to 1 triggers the white balance operation
USER_SW1	Toggling from 0 to 1 triggers the autofocus operation

## 2.6 RTL\_LPDDR4\_AXI4\_Test

This demonstration performs a memory test function using RTL code on the two LPDDR4 modules of Comet A13 SoM board. The LPDDR4-A module is controlled by FPGA fabric in this demo. The memory size of LPDDR4 A and B used in this test are 2GB and 4GB respectively.

### ■ Function Block Diagram

**Figure 2-14** shows the function block diagram of this demonstration. There are two LPDDR4 controllers IP (LPDDR4A and LPDDR4B) in this project. Both controllers use 166.666MHz as a reference clock. The test program will calibrate the two LPDDR4 before writing the data; after writing the memory capacity, it will read the values from LPDDR4 and check whether there is any error.



**Figure 2-14 Block diagram of the LPDDR4 RTL demonstration**



- **EMIF\_Qsys:** It includes two LPDDR4 external memory interface (EMIF) IP which will handle all the LPDDR4 protocols and timings. It provides standard AXI4 for high-speed data transmission and AXI4-Lite for register configuration interfaces externally.
- **LPDDR4 Driver Calibration:** LPDDR4 has an extremely high signal rate and therefore must be calibrated at startup to compensate for signal delays and impedance variations on the board. This module acts as an AXI-Lite Master, after the system starts, it automatically gives instructions to the control registers of EMIF\_Qsys to trigger and monitor the built-in hardware calibration program. After the calibration is successful, the reset signal will be released to the subsequent test modules.
- **AXI4 Master:** This is the core logic to verify LPDDR4. It acts as an AXI4 Master. After the calibration is completed, it will sequentially execute the test process of write, read and verify. It will generate a large amount of pseudo-random data, write it to LPDDR4 at high speed through the AXI4 bus, then read the data from the same address, and perform real-time comparison internally to ensure the accuracy of the data. After the test is completed, it will output the test\_pass and test\_complete signals.
- **Reset & Debouncer:** The debouncer module ensures that when users press User\_Button, a clean and noise-free pulse can be generated. The hyper\_pipe module is responsible for safely synchronizing this reset signal to the high-speed ack frequency domain, preventing metastability issues and ensuring the reliability of system reset.

## ■ Design Tools

- Quartus Prime 25.3.1 Pro Edition

## ■ Demonstration Source Code

- Project Directory: Demonstration\FPGA\RTL\_LPDDR4\_AXI4\_Test
- Bit Stream: golden\_top.sof
- Demonstration Batch File : RTL\_LPDDR4\_AXI4\_Test\demo\_batch

The demo batch file includes following files:

- ◆ Batch File: test.bat
- ◆ FPGA Configuration File: golden\_top.sof

## ■ Demonstration Setup

1. Make sure Quartus Prime Pro Edition v25.3.1 is installed on the Host PC.
2. Connect the Comet A13 EVK board to Host PC via the Type-C USB cable. Install

the USB-Blaster III driver if necessary.

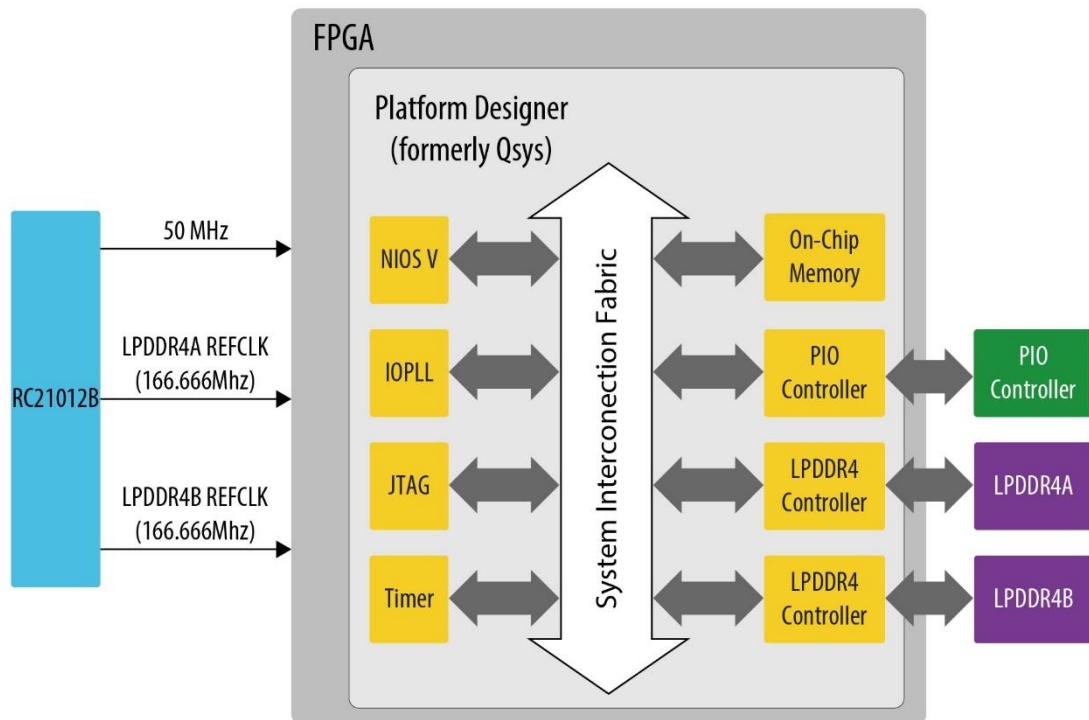
3. Power on the Comet A13 EVK board.
4. Execute the demo batch file “test.bat” under the batch file folder \RTL\_LPDDR4\_AXI4\_Test\demo\_batch.
5. Press **USER\_BUTTON** on the A13 carrier board to start LPDDR4 write & loopback verify process. It will take about 1 second to perform the test.
6. The test result will show on **USER\_LED** on the carrier board. If USER\_LED is light on, it means the two LPDDR4 are tested pass. If there is one LPDDR4 tested fail, USER\_LED will blink.
7. User can press **USER\_BUTTON** again to regenerate the test control signals for a repeat test.
8. During the test process, user can set the **User DIP SW[0]** to OFF, it will insert error manually, it can verify whether the code is correct or there is difference between the write and read data.

## 2.7 LPDDR4\_Test\_NiosV Test

Many applications use a high-performance RAM, such as a LPDDR4 SDRAM, to provide temporary storage. In this demonstration hardware and software designs are provided to illustrate how to perform LPDDR4 memory access in the Platform Designer (formerly Qsys). We describe how the memory controller Agilex External Memory Interfaces is used to access the two LPDDR4 SDRAM on the FPGA board, and how the Nios V processor is used to read and write the LPDDR4 for hardware verification. The LPDDR4 SDRAM controller handles the complex aspects of using the LPDDR4 by initializing the memory devices, managing the LPDDR4 banks, and keeping the devices refreshed at the appropriate intervals.

### ■ System Block Diagram

**Figure 2-15** shows the system block diagram of this demonstration. In the Platform Designer (formerly Qsys), one PLL clock generator (RC21012B) are used. The RC21012B will provide 166.666MHz reference clock to the LPDDR4A and LPDDR4B. There are two LPDDR4 controllers which are used in the demonstrations. Each controller is responsible for one LPDDR4 (LPDDR4A and LPDDR4B) and configured as 2GB and 4GB LPDDR4 controller. The Nios V processor is used to perform the memory test. The Nios V software program is running in the On-Chip Memory. A PIO Controller is used to monitor buttons status which is used to trigger starting memory testing.



**Figure 2-15 Block diagram of the LPDDR4 Nios V Basic Demonstration**

The system flow is controlled by a Nios V program. First, the Nios V program writes test patterns into the whole capacity of LPDDR4 A and B. Then, it calls Nios V system function, `alt_dache_flush_all()`, to make sure all data has been written to LPDDR4. Finally, it reads data from LPDDR4 for data verification. Maybe the process takes a long time, and there is a quick test. The Nios V program writes a constant pattern into the address line and data line and reads it back for verification. The program will show progress in Nios V terminal when writing/reading data to/from the LPDDR4. When verification process is completed, the result is displayed in the Nios V terminal.

## ■ Design Tools

- Quartus Prime 25.3.1 Pro Edition

## ■ Demonstration Source Code

- Quartus Project directory: LPDDR4\_Test\_NiosV
- Ashling RiscFree IDE: LPDDR4\_Test\_NiosV/software

## ■ Nios V Project Compilation

Before you attempt to compile the reference design under Ashling RiscFree IDE, make sure the project is cleaned first by clicking 'Clean' from the 'Project' menu of Ashling RiscFree IDE.

## ■ Demonstration Batch File

Demo Batch File Folder: LPDDR4\_Test\_Nios\demo\_batch

The demo batch file includes following files:

- Batch File for USB-Blaster III: test.bat
- FPGA Configure File: golden\_top.sof
- Nios V Program: MEM\_TEST.elf

## ■ Demonstration Setup

Please follow below procedures to set up the demonstrations.

1. Make sure Quartus Pro 25.3.1 and Ashling RiscFree IDE are installed on your PC.
2. Power on the FPGA board.
3. Use a Type-C USB cable to connect the PC and the Comet A13 carrier board, install USB Blaster III driver if necessary.
4. Execute the demo batch file “test.bat” under the folder “LPDDR4\_Test\_Nios\demo\_batch”.
5. After the Nios V program is downloaded and executed successfully, a prompt message will be displayed in the Nios V Command Shell.
6. For LPDDR4 test (Full), please input key ‘0’ and press ‘Enter’ in the Nios V Command Shell terminal as shown in **Figure 2-16**. The program will display progressing and result information.
7. For LPDDR4 Quick test, please input key ‘1’ and press ‘Enter’ in the Nios V Command Shell terminal as shown in **Figure 2-17**. The program will display progressing and result information. Press User Button of the Comet A13 carrier board to start LPDDR4 verify process.

```

C:\WINDOWS\system32\cmd. x + v
100%
LPDDR4B address bank: 1GB ~ 2GB:
write...
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
read/verify...
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
LPDDR4B test:Pass, 331 seconds
====> LPDDR4x2 Testing, Iteration: 2
== LPDDR4-A Testing...

```

Figure 2-16 Progress option [0] LPDDR4x2 Test

```

C:\WINDOWS\system32\cmd. x + v
Start address 0x60001b94, load size 161260
Transfer rate: 512 KB/sec, 23037 bytes/write.
[Inferior 1 (Remote target) detached]

=== LPDDR4 Calibration Status Check ===
LPDDR4-A Calibration: PASS
LPDDR4-B Calibration: PASS

===== Agilex 5 NIOS LPDDR4x2 Program =====
[0] LPDDR4x2 Test (Full)
[1] LPDDR4x2 Quick Test
Input your choice:1
===== LPDDR4x2 Quick Test! Size=A: 4GB, B: 2GB =====

=====
Press any BUTTON on the board to start test [BUTTON-0 for continued test]
====> LPDDR4x2 Quick Testing, Iteration: 1
== LPDDR4-A Quick Testing...
LPDDR4A address bank: 0GB ~ 1GB: PASS
LPDDR4A address bank: 1GB ~ 2GB: PASS
LPDDR4A address bank: 2GB ~ 3GB: PASS
LPDDR4A address bank: 3GB ~ 4GB: PASS
LPDDR4A test:Pass, 115 seconds
== LPDDR4-B Quick Testing...
LPDDR4B address bank: 0GB ~ 1GB: PASS
LPDDR4B address bank: 1GB ~ 2GB: PASS
LPDDR4B test:Pass, 57 seconds
====> LPDDR4x2 Quick Testing, Iteration: 2
== LPDDR4-A Quick Testing...
LPDDR4A address bank: 0GB ~ 1GB: |

```

Figure 2-17 Progress and Result Information for “LPDDR4x2 Quick Test”



# Chapter 3

## ***Additional Information***

### **3.1 Getting Help**

Here are the addresses where you can get help if you encounter problems:

#### ■ Terasic Technologies

No.80, Fenggong Rd., Hukou Township, Hsinchu County 303035. Taiwan

Email: [support@terasic.com](mailto:support@terasic.com)

Web: [www.terasic.com](http://www.terasic.com)

Comet A13 Evaluation Kit Web: <https://comet-a13.terasic.com/>

#### ■ Revision History

Date	Version	Changes
2026.01	First publication	