Implementation of a camera system using nios II on the altera DE2-70 board

Chan Boon Cheng, Asral Bahari Jambek
School of Microelectronic Engineering, Universiti Malaysia Perlis, Malaysia

Article Info

Article history:

Received Sep 25, 2018
Revised Nov 26, 2018
Accepted Jan 28, 2019

Keywords:

Cyclon II EP2C70 FPGA
SDRAM controller
SOPC builder
Terasic TRDB-D5M

ABSTRACT

The implementation of a camera system with a field programmable gate array (FPGA) is an important step within research towards constructing a video processing architecture design based on FPGA. This paper presents the design and implementation of a camera system using the Nios II soft-core embedded processor from Altera. The proposed camera system is a flexible platform for the implementation of other systems such as image processing and video processing. The system architecture is designed using the Quartus II SOPC Builder System and implemented on an Altera DE2-70 development platform. The image or video is captured using a Terasic TRDB-D5M camera and stored into two different synchronous dynamic random access memories (SDRAM) using an SDRAM Controller. The specifications of the Terasic TRDB-D5M and SDRAM are examined to confirm that the recorded and stored data match. The results of this experiment show that the system is able to record and store data correctly into SDRAM. The data in the SDRAM correctly displays the recorded image on a VGA monitor.

Corresponding Author:

Chan Boon Cheng,
School of Microelectronic Engineering,
Universiti Malaysia Perlis,
Perlis, Malaysia.
Email: chanbc91@gmail.com

1. INTRODUCTION

The implementation of a camera system is important for FPGA-based processing applications, and especially for video processing. The design of an architecture for a camera system is major stage in the further development of a video processing system on an FPGA platform [1]. An intelligent camera system may be implemented on any real-life video processing-based design. The proposed system is implemented on the Altera DE2-70 development platform [2, 3], and the Altera Cyclone II 2C70 FPGA device is the core of the system. The role of the Cyclone II 2C70 FPGA is as a platform for the architecture design of the camera system. In order to make full use of the Cyclone II 2C70 FPGA, the Quartus II system on programmable chip (SOPC) builder was used as the main software for architecture design [3-5].

The proposed camera system makes use of an external peripheral device, the Terasic 5 Mega Pixel Digital Camera (TRDB-D5M). The output data of the TRDB-D5M is in raw format [6], and needs to be converted to RGB format to reduce the complexity of data storage and processing applications. The RGB conversion of captured video for further processing or storage is well-understood in video processing applications [7]. The full resolution frame rate of the TRDB-D5M Camera is up to 15 frames per second (FPS), and the image capture frame resolution is up to 2592*1944 pixels [6, 8].

SDRAM plays an important role in the design of the camera system [9]. The Altera DE2 board contains an SDRAM chip that can store 8 Mbytes of data, in which the memory is organized into 1M x 16 bits x 4 banks. In order to access the SDRAM Chip, an SDRAM controller circuit is needed while working on the architecture design of the camera system. This SDRAM controller circuit generates signals which can communicate with the SDRAM chip when receiving read or write instructions from the Cyclon II 2C70 processor [2, 10].

Architecture is the most important part of a camera system design [11], and errors in design or simulation will give rise to major or minor errors at the subsequent compilation stage. Figure 1 presents a typical block diagram of the architecture design for the proposed system. The user is able to debug the program in C/C++ language using the Nios II Software Build Tool for Eclipse [1, 3, 5, 12] and to download the instructions into the Nios II processor through the joint test action group-universal asynchronous receiver transmitter (JTAG-UART) core. The video captured by the TRDB-D5M camera is converted into RGB format and stored in the SDRAM Chip through the camera_if controller and the SDRAM controller. Instructions for writing from the Nios II processor allow the SDRAM controller to carry out writing of the recorded data to the SDRAM Chip. The data correctly stored in SDRAM is able to display the recorded image on a VGA monitor [13, 16]. A more detailed explanation of the communication between the TRDB-D5M and the SDRAM chip will be discussed in Section 3.

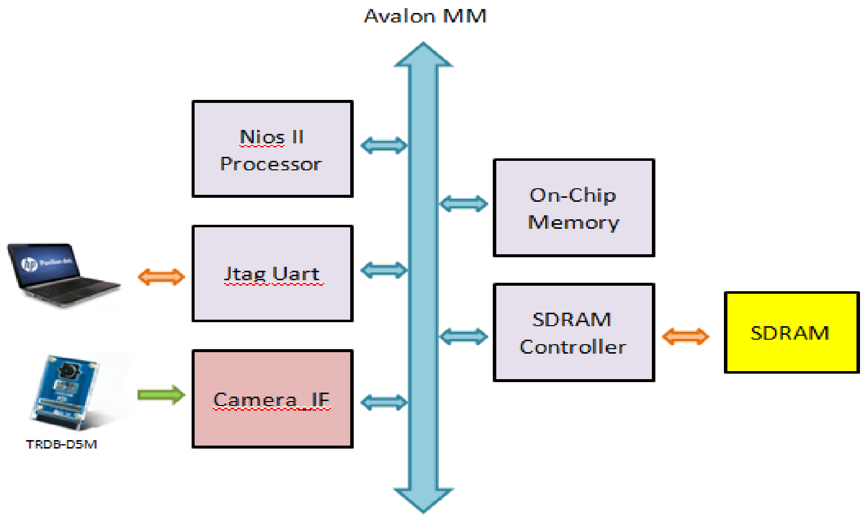

Figure 1. Typical block diagram of the architecture design for the proposed system

In this article, several methods for the design of camera system architectures and applications are surveyed. The architecture design and applications examined are as follows: the implementation of a smart camera system on Xilinx VSK platform [9], the implementation of an open image processing system on the Altera DE2-70 board [8], the implementation of a smart camera on the Altera Stratix EP1S60F1020C7 device [1], the implementation of a camera system controlled from an LCD touch panel on an Altera DE2 board [14], and a real-time edge detector implementation on FPGA [15]. In Section 2, the design of these architectures and the application of existing methods are discussed. This section ends with a comparison between the pros and cons of existing methods for architecture design. In Section 3, a detailed description is given of the proposed implementation for a camera system. Section 4 describes the outcome of the design in terms of the flow of data conversion and storage.

2. LITERATURE REVIEW

The design of camera system architecture plays an important role in the implementation of a video processing application on FPGA. An intelligent architecture design is able to run perfectly on any processing implementation of the camera system. The research on an FPGA-based smart camera implementation presented by the author in [1] provides another reference for the use of an Altera platform. The Altera Stratix EP1S60F1020C7 plays a major role in this system. The sub-memory is 10 Mb of SRAM, while the major data storage device is 64 Mb of SDRAM. A smart camera LUPA-4000 with an image sensor of 4 Mpixels is the current camera configuration. The communication between smart camera, SSRAM, SDRAM and host computer is shown in Figure 2.

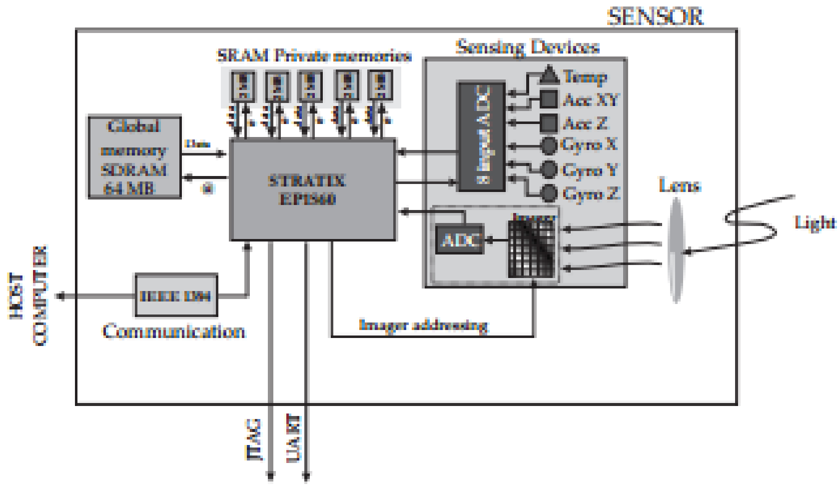

Figure 2. Communication between smart camera, SSRAM, SDRAM and host computer

In [8], the author described the implementation of an open image processing system on the FPGA platform, using the Altera DE2-70 as the chosen development platform. The TRDB-D5M camera and the 4.3" Ultra-high Resolution LCD Touch Panel (TRDB-LTM) are important external peripherals completing the research. The camera sub-system core in this paper provides a good reference for the current camera implementation. The proposed camera sub-system is able to produce a 24-bit RGB image frame with a resolution of 640x480 pixels. SDRAM is used as the major data storage device for the captured images or video for further processing and display. Figure 3 shows the design of the camera sub-system proposed by the author.

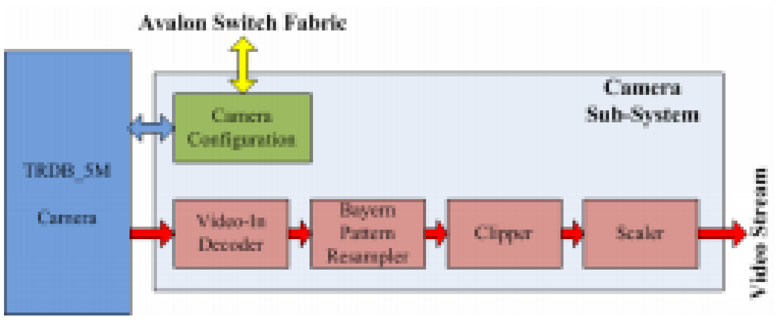

Figure 3. Design of the camera sub-system proposed by the author

The author of [9] has proposed an FPGA-based smart camera system, which involves the two important features of a pan-tilt-zoom (PTZ) Camera and a Spartan-3A DSP-based Xilinx VSK platform. DDR2. The major data storage device for storing and extracting frames is SDRAM. Figure 4 shows the block diagram for the architecture design proposed by this author.

The architecture of the camera sub-system design presented in [14] provides another reference for the current camera implementation. An LCD touch panel sub-system is a further external peripheral used to display the captured image. JTAG-UART is used to transfer data, and the main FPGA device is the Altera DE2. The camera controller, SDRAM controller and LCD touch panel controller are responsible for communication between the FPGA board, internal devices and external peripheral devices. Figure 5 shows the proposed architecture design of the full system. The camera sub-system in this proposed method captures the image from the CMOS image sensor, which then undergoes some processing before being stored into SDRAM. Figure 6 shows the block diagram of this proposed camera sub-system.

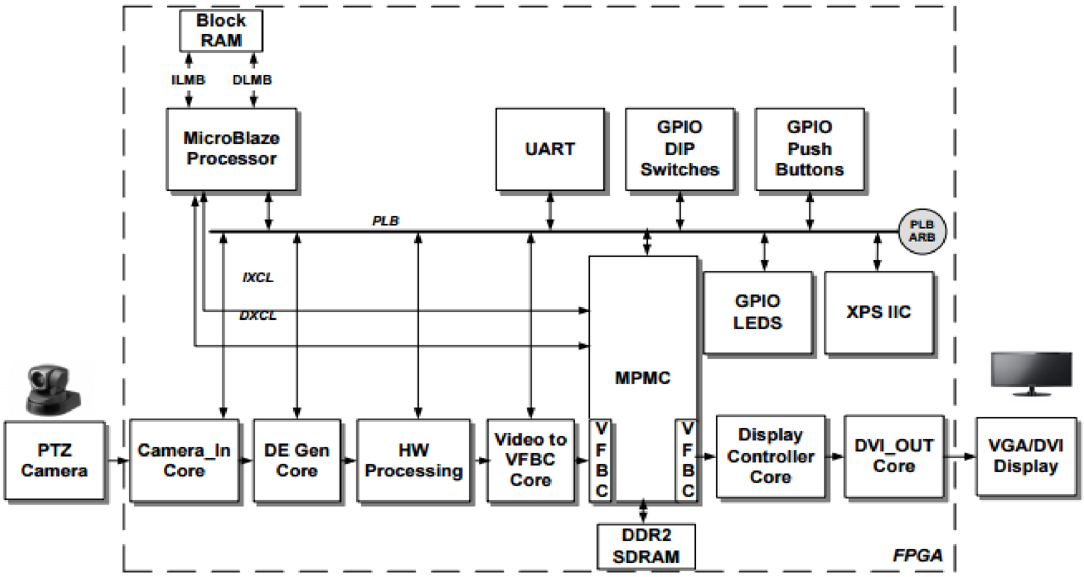

Figure 4. Block diagram for the architecture design proposed by this author

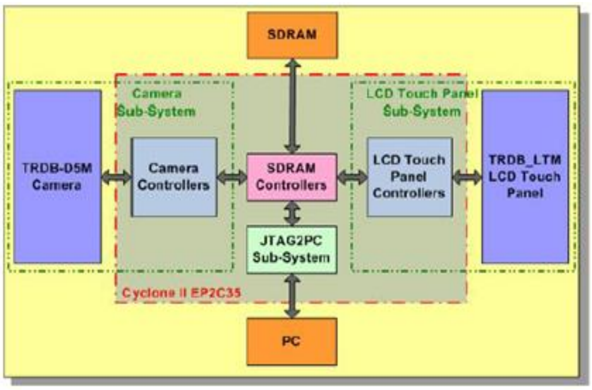

Figure 5. Proposed architecture design of the full system

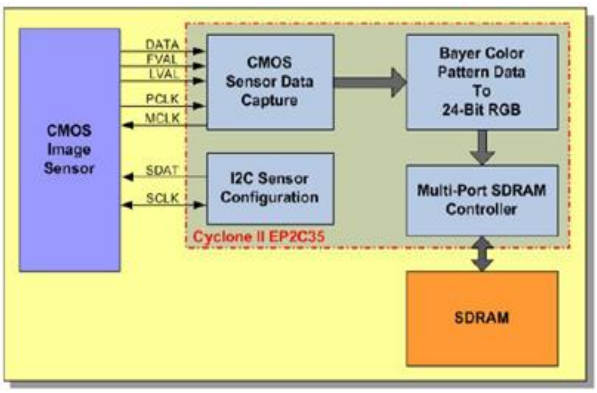

Figure 6. Block diagram of this proposed camera sub-system

In [15], the author proposed a real-time implementation for edge detection using FPGA. A CMOS camera was chosen to capture images in real time and perform further processing steps. The Sobel, Prowitt, Robert and Compass edge detection algorithms were studied and implemented in this design. The Microblase RISC processor was used as the main processing unit, and a DVI display was used as another external peripheral for displaying the outcome of processing. The design of this architecture included LEDR PIO, push-button and switch PIO and other system peripherals. Figure 7 shows the design of the system architecture.

Table 1 shows a comparison between the implementations of camera systems on FPGA. Each of the proposed methods involves an external peripheral camera and FPGA. The different types of output depend on the relevant FPGA. All of these methods have the common feature of collecting video frames using an external peripheral camera and storing these into a memory device for display or further processing. The various development platforms contain different types of memory device for data storage. For example, the Xilinx VSK uses DDR2 SDRAM for video frame storage, while the Altera DE2 and DE2-70 use SDRAM for video frame storage. The Altera Stratix uses both SRAM and SDRAM to store video frames.

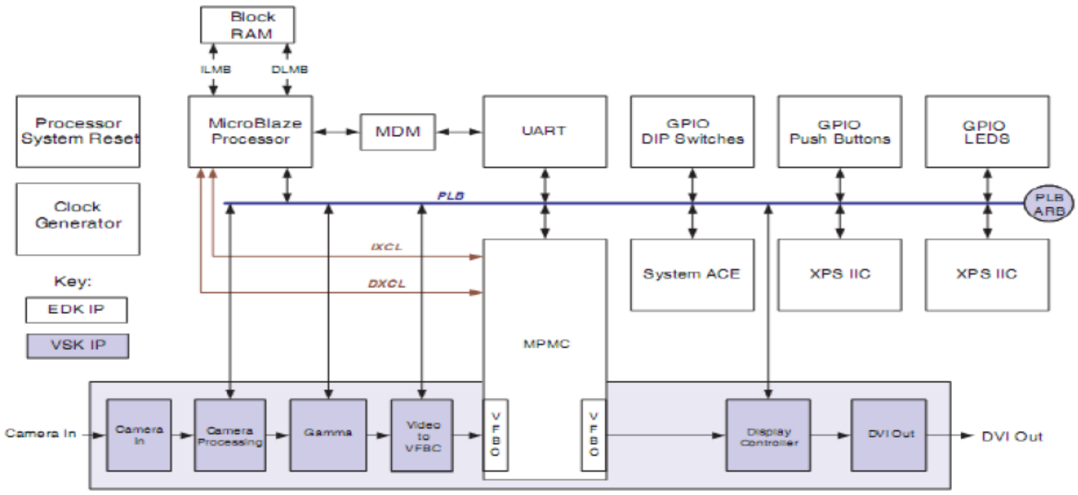

Figure 7. Design of the system architecture

Table 1. Comparison between the Implementations of Camera Systems on FPGA

Features	[1]	[8]	[9]	[14]	[15]
Year	2007	2015	2013	2009	2011
Development Platform	Altera Stratix	Altera DE2-70	Xilinx VSK	Altera DE2	Microblaze RISC Processor
Camera	4Mpixels LUPA-4000	TRDB-D5M Camera	PTZ Camera	TRDB-D5M Camera	CMOS Camera
Display Device	VGA Monitor	TRDB-LTM LCD	VGA Monitor	TRDB LTM LCD	DVI Display
Memory Usage	SRAM & SDRAM	SDRAM	DDR2 SDRAM	SDRAM	Line Buffer
Output Display Size	640*480	680*480	752*582	640*480	720*480
Data Processing	Window-Base Processing	Image Processing	HW Processing	Not Stated	Image Processing
Software	Not Stated	Quartus II, Nios II SBT for Eclipse	ISE, EDK, SDK Xilinx Tools	Not Stated	Not Stated
Language	VHDL, C/C++, Assembler	HDL, VHDL, C/C++	Not Stated	C/C++	Not Stated
Special Features	Not Stated	Selection Mode Display	Not Stated	Adjustable Exposure Time Register	Edge Detection

The selection mode display described in [8] is an extra feature that allows switching between different modes of image processing. The processing mode includes negative colour, edge detection, a median filter and a sharpen convolution filter. The FPGA-based digital camera system proposed in [14] involves the special feature of an adjustable exposure time register. This feature provides adjustable brightness for display image by increasing or decreasing the register value. The edge detection proposed in [15] is another interesting feature that allows the detection of the edges of a captured object. Each of the features listed above can be implemented after the implementation of the camera system.

The implementation of the current camera system is suited to the Altera DE2-70 platform, due to the various types of language chosen to carry out architecture design. The C/C++ algorithm design in the Nios II SBT for Eclipse allows the design of various types of function and run on the Altera DE2-70 board. The Altera DE2-70 board contains both SDRAM and SSRAM devices; SDRAM is suitable for the storage of processed video frames, while SSRAM is suitable for storage of temporary video frames or instructions. The architecture design step is simple, and is carried out using the Qsys function in Quartus II software. The combination of Verilog and Qsys architecture designs simplifies the setting step for the pin planner and camera_if settings. The complete system design is discussed in Section 4.

3. RESEARCH METHOD

In Section 3, the existing methods for implementation of camera systems are discussed. Section 4 explains the architecture design and the important peripherals used in the proposed camera implementation system. The proposed architecture design is implemented on the Altera Cyclone II 2C70 FPGA and is interfaced with an external peripheral, the TRDB-D5M. The important components of the architecture design include a Nios II soft-core processor, one Cypress CY7C1380C SSRAM, two 32_Mb SDRAMs with an SDRAM controller, a JTAG-UART, a RS-232 serial port universal asynchronous receiver transmitter (UART), a timer module, an Avalon bus, a TRDB-D5M with a camera_if and a system ID Peripheral. Figure 8 shows the block diagram for the proposed design for the implementation of the camera system.

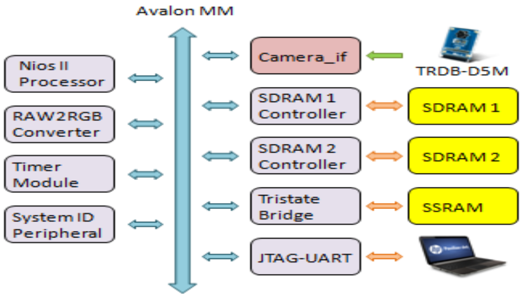

Figure 8. Block diagram for the proposed design for the implementation of the camera system

The TRDB-D5M is the major external peripheral that carries out the image capture progress. The pixel array in the TRDB-D5M consists of a 2752-column by 2004-row matrix of pixels, addressed by column and row. The output of the pixels is in a Bayer pattern format, consisting of four colours, Green1 (G1), Green2 (G2), Red (R) and Blue (B) [6]. The FRAME_VALID and LINE_VALID signals in the TRDB-D5M indicate the boundaries between the frame and the outline of the output image. In order to capture a valid image, FRAME_VALID and LINE_VALID signals are sent, and valid image data is captured and stored into SDRAM. Figure 9 shows the theoretical communication between FRAME_VALID (FVAL), LINE_VALID (LVAL) and SDRAM.

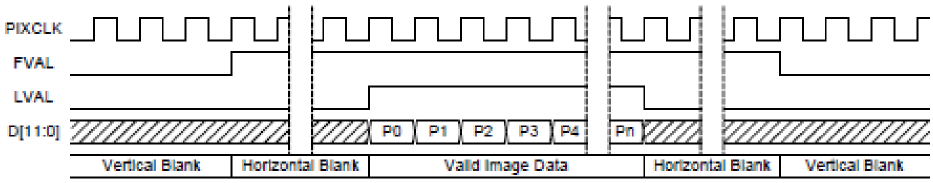

Figure 9. Theoretical communication between FVAL, LVAL and SDRAM

The proposed architecture design was developed using the Qsys tool from the Quartus II Version 10.0 software. Version 10.0 of the Quartus II was chosen rather than Version 13.0, since the external peripheral TRDB-D5M performs better in the older version. Figure 10 shows the interconnection of the proposed camera system in the Qsys tool. The system design is based on the block diagram shown in Figure 8. During the system design, a phase-locked loop (PLL) was used to generate a 100 MHz clock for the system and 100 MHz pulse with a −65 degree phase shift for SDRAM, while 50 MHz clock pulses were used as the supply for the PLL module.

SSRAM was chosen as the reset and exception vector for the Nios II processor design, in order to avoid exceeding block limitations in SignalTap II when generating the waveform of SDRAM data. The camera_if function in the Qsys tool performs as a communication bus between the processor and the TRDB-D5M camera. Verification for the port of the TRDB-D5M camera was carried out using Verilog coding and this was then incorporated into the design unit in Quartus II software. The verification of the port enables the TRDB-D5M to accept and send signals, from and to the development platform, through camera_if. Figure 11 shows the part of the verification code involving port declarations.

The Nios II processor is the primary component in the design of the system. Its relatively high processing speed is able to accelerate the task given to the development platform. The Nios II full version soft-core processor is chosen for this system, since its highest processing speed is approximately 101 Dhrystone million instructions per second (DMIPS) at 100 MHz. A push-button GPIO is added to the Qsys design, which includes start and stop recording functions. The camera begins capturing and saves the image into SDRAM when the start recording push-button is triggered. At the same time, the GPIO of the seven-segment LED display shows the running time of the record function.

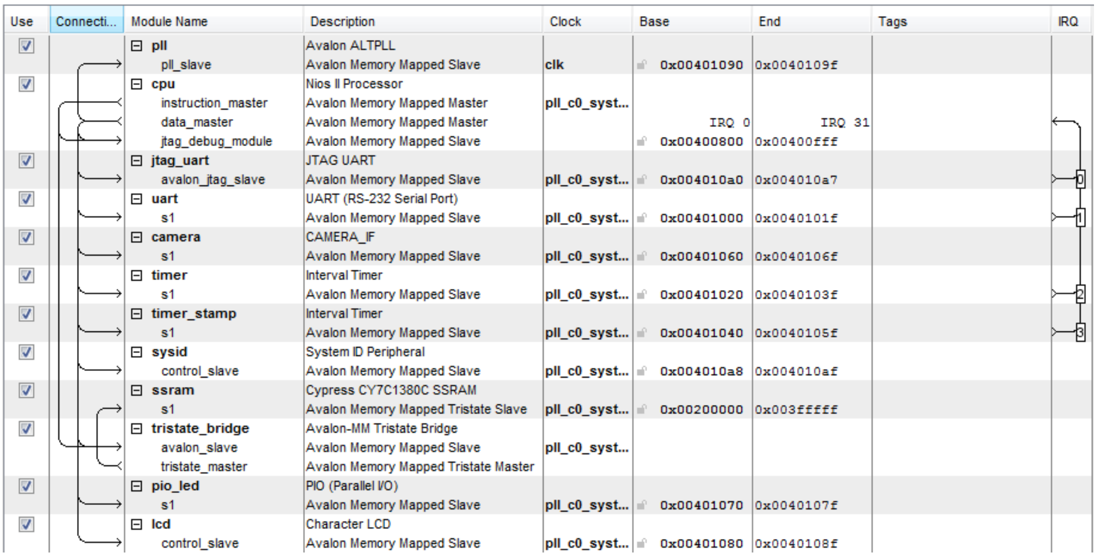

Figure 10. Interconnection of the proposed camera system in the Qsys tool

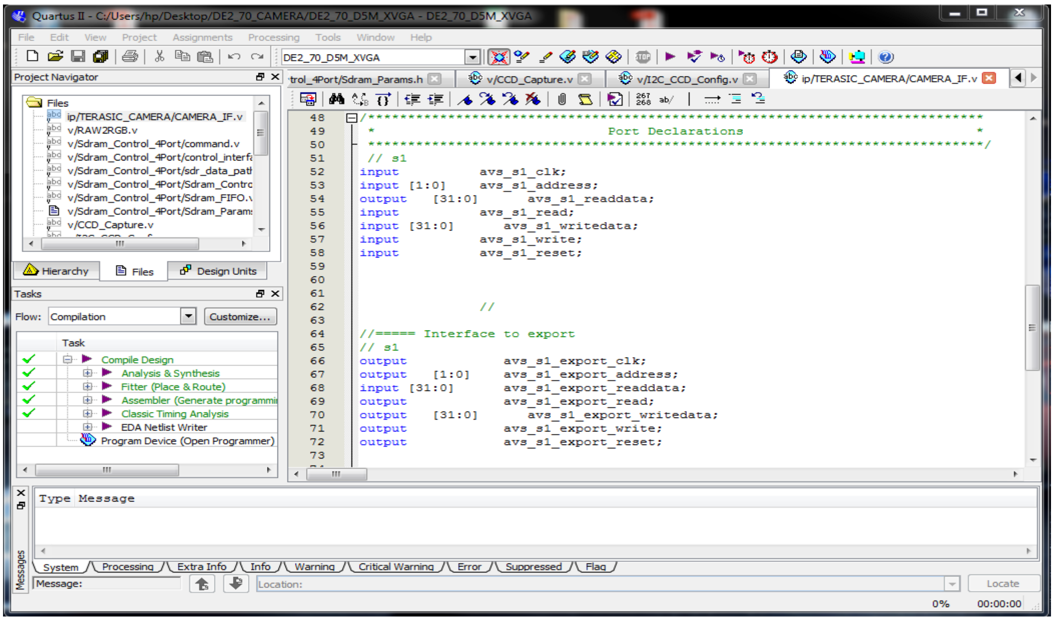

Figure 11. Part of the verification code involving port declarations

The complete design of the camera system was generated using the Qsys tool, and the qip format file was automatically generated in the design folder. The Verilog format of the camera verification and qip format files generated by the Qsys tool were manually added into the design unit of the system. Pin verification of each of the included components was carried out using the pin planner in the Quartus II software. The entire system was compiled following the pin planner verification process. The design of the camera system architecture was then loaded to the FPGA board through the programmer in Quartus II. Figure 12 shows the completed sof format file loaded to the FPGA Board. Figure 13 shows the initialization of FPGA board after uploading of the architecture design.

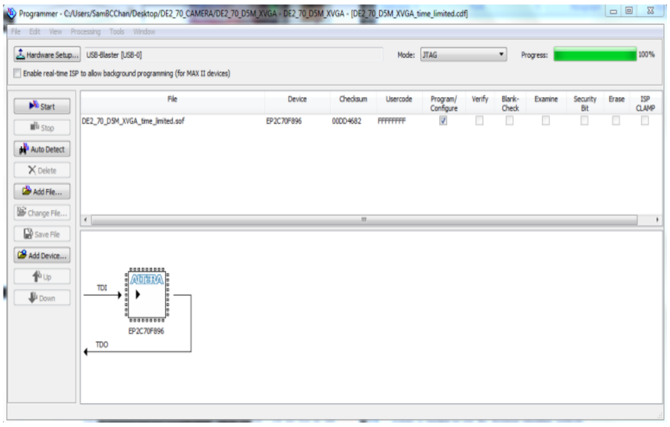

Figure 12. Completed sof format file loaded to the FPGA board

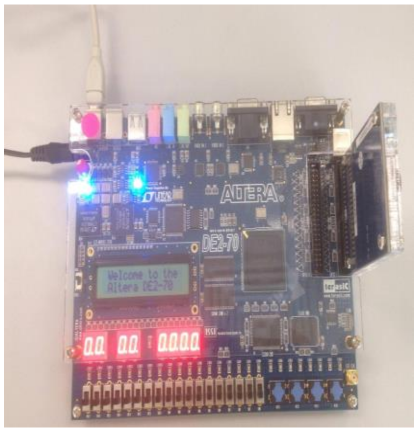

Figure 13. Initialization of FPGA board after uploading of the architecture design

4. RESULTS AND ANALYSIS

In this section, the flow of progress and results are explained in detail. The architecture design uses a process of several stages before user triggers the recording function. The steps of this process are as follows:

a) Reset the seven-segment LED
b) Setup start and stop recording buttons
c) Initialize the TRDB-D5M camera
d) Manage the flow of stored data into SDRAM

When the architecture design is downloaded to the Altera FPGA board, the seven-segment LED on Altera FPGA Board is initialized to zero, as shown in Figure 12. This initialization of the seven-segment LED is in preparation for the frame counter for video recording. The Key 3 push-button is set as the start recording function, while Key 2 is set as the stop recording function. Initialization of the TRDB-D5M camera is carried out using Verilog coding to connect and receive commands from the Nios II processor.

When the Key 3 push-button is triggered, a signal of 1 bit is sent to the NIOS II processor. The Nios II processor then sends a signal to the seven-segment LED to begin incrementing the value, while the TRDB-D5M camera starts the recording process. Recorded video frames are converted into 12-bit signal data as output data (oDATA) using common-core data wire capture (CCD capture). Since pixels are generated in raw format by the TRDB-D5M camera, a conversion step from raw to RGB format is required. The oDATA from the CCD capture therefore becomes the input data (iDATA) for the RAW2RGB converter.

When the data conversion is complete, the RAW2RGB converter generates three types of 12-bit signal data output. These are red output data (oRED), green output data (oGREEN) and blue output data (oBLUE). The output of the RAW2RGB converter is stored in the SDRAM devices on the Altera DE2-70 FPGA board for future processing. Since an SDRAM device on the Altera DE2-70 FPGA board is able to store 16 bits of data, two SDRAM devices are needed to store the three types of output data of RAW2RGB Converter. The data stored into the two different SDRAM devices is shown as a WR1_DATA waveform for SDRAM 1 (u8) and SDRAM 2 (u9). The waveforms of the input and output data of each component described above are generated using SignalTap II Logic Analyzer in Quartus II. Figure 14 shows the waveforms of the input and output data for each component.

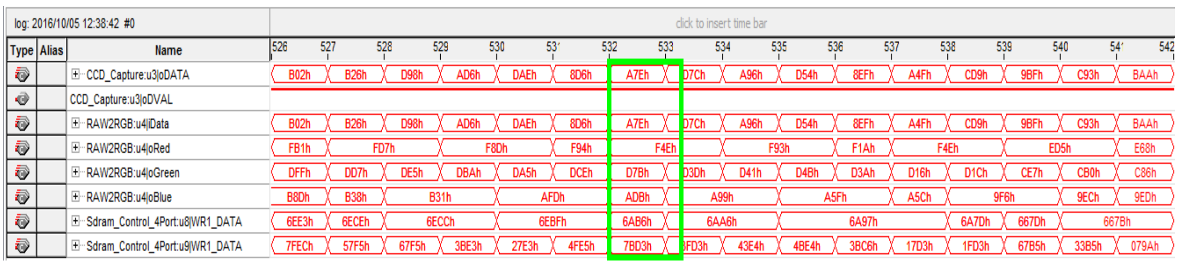

Figure 14. Waveforms of the input and output data for each component

A column of the waveform in Figure 14 is used in order to prove that the flow of data from the TRDB-D5M camera to SDRAM is correct. The chosen column is highlighted using a green rectangular shape with 12-bit CCD capture output data of A7Eh (oDATA). The waveform generated using SignalTap II logic Analyzer shows that the output data from the camera was identical to the input for the RAW2RGB converter. The input data (iDATA) for RAW2RGB was the same 12-bit data received from oDATA. The received data, A7Eh, was converted into 12-bit output data in three parts: F4Eh for oRED, D7Bh for oGREEN and ADBh for oBLUE.

In order to store the three outputs of RAW2RGB into the two SDRAM devices, the data for oGREEN (D7Bh) was split into two parts, that is, data between [11:7] and [6:2]. Data between [11:7] of the oGREEN output and [11:2] of the oBLUE output was saved into SDRAM 1 (u8), while data between [6:2] of the oGREEN output and [11:2] of the oRED output was saved into SDRAM 2 (u9). The splitting and recombining of the RAW2RGB output data is shown below in binary format.

Data stored in SDRAM 1 (u8):
The binary format of D7Bh for oGREEN is 110101111011
[11:7] of oGREEN is 11010

The binary format of ADBh for oBLUE is 101011011011
[11:2] of oBLUE is 1010110110

The data stored in SDRAM 1 (u8) is 6AB6h, which in binary format is 110101010110110

The binary format of SDRAM 1 (u8) is proven to be a combination of [11:7] of oGREEN and [11:2] of oBLUE.

Data stored in SDRAM 2 (u9):
The binary format of D7Bh for oGREEN is 110101111011
[6:2] of oGREEN is 11110

The binary format of F4Eh for oRED is 111101001110
[11:2] of oRED is 1111010011

The data stored in SDRAM 2 (u9) is 7BD3h, which in binary format is 111101111010011

The binary format of SDRAM 2 (u9) is proven to be a combination of [6:2] of oGREEN and [11:2] of oRED.

Based on the SignalTap II Logic Analyzer compilation waveform, the output data of RAW2RGB is the same data that is stored into SDRAM 1 and SDRAM 2. The waveform shows that the data capture from TRDB-D5M camera is the same as the input data for RAW2RGB. RAW2RGB generates output data in three parts, which is successfully saved into two different SDRAM devices.

Figure 15 shows the results of compiling the full camera system, including the design of the VGA display. The data in SDRAM in RGB format is converted into video frames with resolution 640 x 480. The display result shows that the data stored in the SDRAM in RGB format is correctly converted from the raw format camera capture. The logic elements used in the compilation is 10,639 / 68,416 which is 16 % of the total logic elements while pins used is 530 / 622 which is 85 % of total pin in DE2-70 Board. Total thermal power dissipation of full compilation is 1420.09 mW.

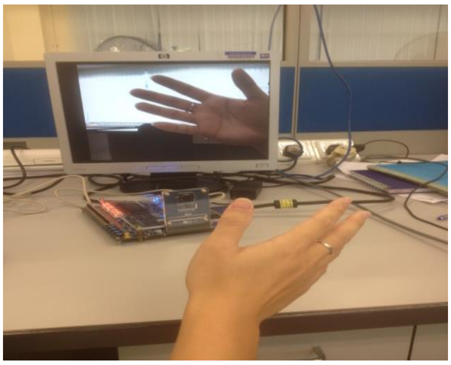

Figure 15. Results of compiling the full camera system, including the design of the VGA display

5. CONCLUSION

In this paper, a design for the architecture of a camera system is implemented using the Altera DE2-70 FPGA board. A video frame is captured using a TRDB-D5M camera attached to the FPGA Board. Pixels captured in raw format are converted into RGB format and stored into SDRAM. An analysis is carried out using the SignalTap II Logic Analyzer to ensure that the data stored into SDRAM is correct. This correct data storage into SDRAM 1 and SDRAM 2 forms a valuable basis to continue future work, in which the data in SDRAM will be processed to detect and track moving objects.

REFERENCES

[1] Dias, F., Berry, F., Serot, J., and Marmoiton, F.: Hardware, "*Design and implementation issues on a FPGA-based smart camera*", First ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC '07); Vienna, Austria, pp 20-26 (2007).

[2] Chu, P.P.: "*Embedded SoPC Design with Nios II Processor and VHDL Examples*", J. Wiley, Aug 2011.

[3] ALTERA: "*DE2-70 Development and Education Board: User Manual*" v1.08, 2009.

[4] ALTERA: "*SOPC Builder: User Guide*", v1.0, Dec., 2010.

[5] ALTERA: "*Quartus II Handbook*" v14.1, Dec. 2014.

[6] Terasic, "*TRDB-D5M: Terasic D5M Hardware Specification*", April 2008.

[7] Tchahou Tchendjeu, A.E., Tchitnga, R. and Fotsin, H.B.: "*FPHA-based seven-language electronic calendar for the revival of the Cameroon culture*", Sciences, Technologies et Développement, Edition spéciale, pp197-202, July 2016.

[8] Pyrgas, L., Kalantzopoulos, A., and Zigouris. E.: "*Design and implementation of an open image processing system based on NIOS II and Altera DE2-70 Board*", 2015.

[9] Said, Y., Saidani, T., Smach, F., Atri, M. and Snoussi, H.: "*An FPGA-based smart camera system*", Third International IEEE Conference (IPWIS13), 2013.

[10] Lakis, E. and Schoeberl, M.: "*An SDRAM controller for real-time systems*", Proceedings of the 9th Workshop on Software Technologies for Embedded and Ubiquitous Systems, 2013.

[11] Oetken, A., Wildermann, S,. Teich, J. and Koch, D.: "*A bus-based SoC architecture for flexible module placement on reconfigurable FPGAs*", International Conference on Field Programmable Logic and Applications, 2010.

[12] ALTERA: "*Nios II Software Developer's Handbook*", v13.1, Jan., 2014.

[13] Lam, C.Y., Soon, V.S. and Ehkan, P.: "*Reconfigurable tri-colour RGB SD card bitmap image file writer*", Jan 2016.

[14] Petouris M. et al.: "*An FPGA-based digital camera system controlled from an LCD touch panel*", International Symposium on Signals, Circuits and Systems (ISSCS), 2009.

[15] Sudeep, K.C.and Majumdar, J.: "*A novel architecture for real time implementation of edge detectors on FPGA*", IJCSI International Journal of Computer Science, Volume 8, Issue 1, pp. 193-202, January 2011.

[16] Lee, S.M. et al. "*Design of a DMA controller for lossless image processing*", IDEC Journal of Integrated Circuits and Systems, Vol 3, No.1, July 2016.

BIOGRAPHIES OF AUTHORS

	Chan Boon Cheng is currently an M.Sc. by research student at School of Microelectronic Engineering, University Malaysia Perlis. In 2015, he completed his B.Eng. degree in Electronic Engineering at University Malaysia Perlis, Malaysia. He has served Sensmaster SDN.BHD as graduate trainee and failure analysis assistance in 2014. His research interest includes object tracking algorithm and implementation using SOC devices.
	Associate Professor Dr. Asral Bahari Jambek is a member of the School of Microelectronics Engineering, Universiti Malaysia Perlis (UniMAP), and was a Programme Chairperson for the Electronics Engineering Degree Programme, UniMAP. He has more than 15 years experience in integrated circuit and system design in both the industry and academic sectors, and has been involved at various levels of VLSI design such as transistor modelling, digital circuit design, analogue circuit design, logic synthesis and physical place and route, architecture design and algorithm development.Currently, he is actively researching new techniques to minimize power consumption in portable system-on-chip design. His research interests include integrated circuits and systems design, digital signal processing (DSP), low power algorithms and architectures design, and image and video processing.