Design of High speed and dynamic architecture for color conversion with FPGA for real time processing

Taoufik Salem Saidani[1] and Hafedh Mahmoud Zayani[2]

[1]Department of Computer Sciences, Faculty of Computing & Information Technology,
Northern Border University, Rafha, Saudi Arabia
[2]Department of Information System Faculty of Computing & Information Technology
Northern Border University, Rafha, Saudi Arabia

Summary

Due to high rise in demand for video processing on hardware FPGA implementation of various image processing algorithm has become a necessity. Lot of research work is being done in the field of image processing and video processing on FPGA. The paper speaks about two block of implementation mainly the image block generation and the RGB to YCbCr conversion and YCbCr to RGB conversion. It has been proved from the design that it is efficient for high definition (HD) video sequences with frame size 1920 × 1080. The experimental results represent that the performance of the implemented hardware architecture is good when carrying out color space conversion from RGB to YCbCr. It also has the advantage of being high-speed, low complexity, and low-area. Both the bocks are targeted on commercial field programmable gate-array (FPGA) devices.

Key words:

Image Processing, Field Programmable Gate Array(FPGA), , Color Space Conversion,Xilinx system generator..

1. Introduction

Intelligent surveillance system is one of the most in demand system these days, due to rise in demand for better security[1,2]. Color space conversion has come to play a very important role in video processing and transmission technology; generally speaking, it is not practical to try to transmit images in RGB color space because it requires quite high bandwidths. The human eye is more sensitive on brightness change than on color change. Hence, if luminance-chrominance color space is used for color image transmission, data storage amount and bandwidth can be reduced, and the price is on the negligence of very tiny or almost unrecognized color change information. YCbCr is a hardware-oriented color model. When video processing is done in YCbCr color space, we can get very high compression ratio and transmission rate, hence, in most image or video compression applications (for example, JPEG and MPEG, etc.), we will usually need to use the transformation between RGB and YCbCr color space. Color space converter (CSC) is far more efficient in hardware than its implementation in software. Consequently, we can improve the throughput of our system by implementing a hardware accelerator (HA) for color space conversion. However, multiplication in a general-purpose processor or in a custom hardware implementation is generally an expensive operation, in terms of latency or hardware costs. When we realize CSC from RGB to YCbCr, we have two dilemmas in hardware circuit design.

Digital Signals (DSP) or Application Circuits Specific (ASIC). But as the need increases more systems are needed with better performance and shorter execution time The Field-Programmable Gate Arrays, from now on named FPGA, are circuits integrated by arrangements of logical blocks that they allow reconfiguring as many times as necessary to debug their functionality, communicate with each other with terminals of entrance / exit by means of fences called communication channels. The size and speed of FPGAs are comparable to ASICs, but FPGAs are more flexible, its design cycle is shorter and parallelism gives a higher performance [1,2].

To comply with the ITU-R BT.601-5 standard, the card digitizer (handles analog to digital conversion) VDEC-1 of the company DIGILENT delivers the video signal in digital format of the YCbCr color model, these signals are transformed to the RGB color model to be able

be interpreted by a monitor [5,6].

The organization of this article is the following: in the section 2 the model transfer equations are presented from YCbCr to RGB, suggested by the ITU-R standard BT.601-5. In In section 3, a proposed architecture for color system conversion will be presented for the implementation of the digital conversion system between each color model (YCbCr and RGB). In section 4, they are presented the results for the proposed architecture for color system conversion, the results will be bought numbers obtained by the digital design in the FPGA and the shown in section 2. Finally, the conclusions are presented obtained in the development of this work.

Manuscript received January 5, 2018
Manuscript revised January 20, 2018

2. Color System Conversion

The mathematical representation of color set is called as color space. RGB is the most common standard used for the display of image as it is most prevalent choice in display graphics. Any kind of color can be created using the combination of RGB[1,11]. However it cannot be used for processing of video as the frame buffer needs to have pixel depth and display resolution for each RGB component. This can be cleared with an example, assume the intensity of the given input image needs to be modified. In this process we need to develop the unit that needs to work on the entire three components independently and regenerate each RGB value and rewrite the new values on the frame buffer as shown in figure1. This process generally takes more time compared to other color space such as YCbCr where we just need to work on only the intensity instead the color component which makes the work faster and easy with less computational complexities involved.

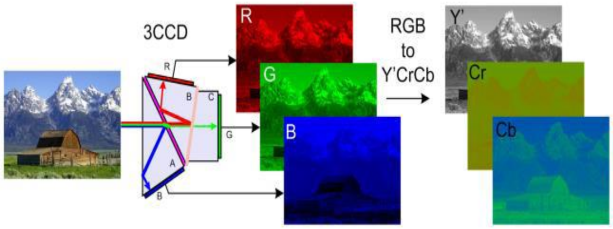

Fig. 1 Representation of RGB to YCbCr format

YCbCr is the color space standard that is used for real time video and image. It is widely used in all the video standard such as PAL, NTSC, SECAM and composite color video standard. Even the Black and white only the Y component is used. Further Cb and Cr component were also introduced. As mentioned earlier RGB is widely used for all the display unit data available to the user is always in this format and it is very much necessary to convert the RGB format to YCbCr format. Below shows equation that are used for the transformation of image form one to the other format.

The change of color space, the RGB color space (red, green, blue) to the YCbCr space (luminance, chrominance) aims to to improve the efficiency of the JPEG compression process. Indeed the eye human is not very sensitive to the alteration of chrominance.

The conversion between the two color spaces is carried out using the mathematical formulas provided below:

$$Y = 0.299xR + 0.587xG + 0.114xB$$

$$C_b = -0.1687xR - 0.3313xG + 0.5xB + 128$$

$$Cr = 0.5xR - 0.4187xG - 0.0813xB + 128$$

The reverse operation that converts information from the YCbCr space to the RGB space is made from the equations provided below:

$$R = Y + 0.402x(C_r - 128)$$

$$G = Y - 0.34414x(C_b - 128) - 0.71414x(C_r - 128)$$

$$B = Y + 1.772x(C_b - 128)$$

For practical reasons, the values of the components {R, G, B} and {Y, Cb, Cr} are represented using whole data and belong to the interval [0, 255].

The luminance channel appears to be very similar to the grayscale version of the original image. Cb is strong where the blue colour is dominant in the image, for instance when the image is taken from the sky. Cr is strong when the image is taken from places where the reddish colours are dominanat and both Cb and Cr factors are weak where the green colour is leading. [7,8]

The following figure 2 shows the colour difference in different channels and also the difference between RGB and YCbCr.

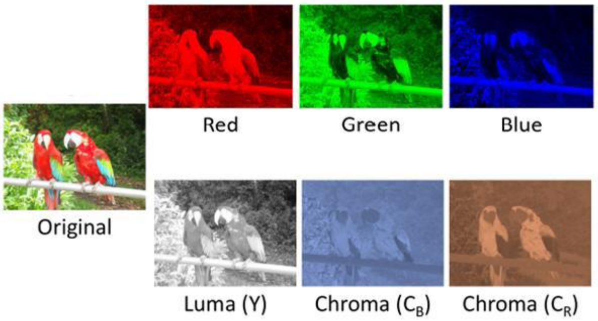

Fig. 2 Difference between RGB and YcbCr

3. Hardware Implementation of the Proposed CSC architecture

In this section, hardware architectures designed for RGB-YCbCr qnd YCbCr-RGBcolor space conversion is described. The architecture has been designed in accordance with the color space conversion algorithm.

The Color space conversion IP block contains two modules RGB to YCbCr and YCbCr to RGB. The RGB to YCbCr Color Space Converter IP module implements the equations to convert 24-bit input RGB color samples to 24-bit YCbCr output samples. The YCbCr to RGB Color

Space Converter IP module converts vice-versa of the RGB to YCbCr. Both the convertors use 4:4:4 sampling format. Both the modules take data enable, horizontal, and vertical sync signals as inputs and pipeline them accordingly to match the conversion video data outputs. To convert the floating point constants into integer multiplication, the floating point constants are scaled by multiplying these constants with 215 = 32768. Then after the computation of the above equations, the output is divided by the scaling factor 215 = 32768.

The top level signal diagram for RGB-YCbCr color space conversion module is shown in Fig. 3. The detailed hardware architecture for RGB to YCbCr and YCbCr to RGB color space conversion is shown in Fig. 4.

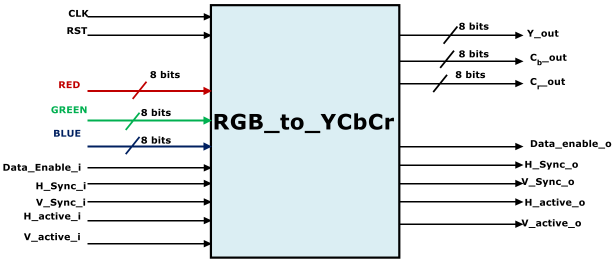

Fig. 3 RGB to YCbCr Block

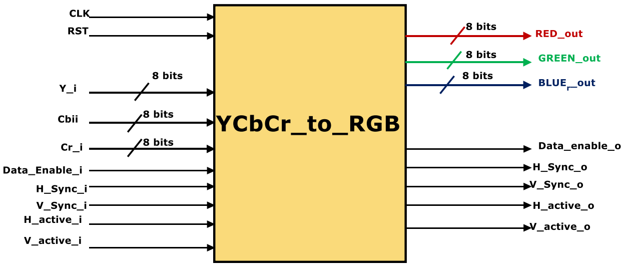

Fig. 4 RGB to YCbCr Block

Figure 5 shows a direct mapping of the above three equations. This conversion is performed as digital multiplication and addition utilizing floating-point multipliers and summation circuits.

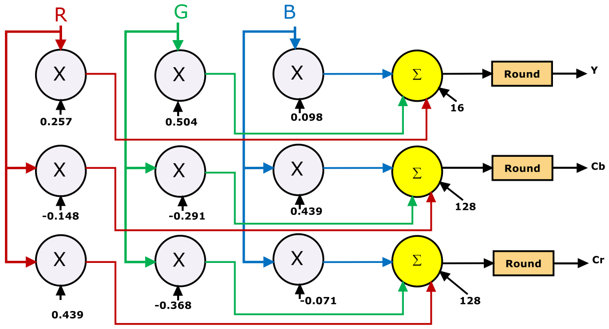

Fig. 5 General block diagram for RGB to YCbCr.

Hardware implementation of color space conversion starts with the Matlab verification followed by hardware architecture development, VHDL coding, logic synthesis, place and route and FPGA implementation. First, we begin with realization RGB to YCbCr converter on FPGA. The RGB to YCbCr converter module design is based on Eqns. (1) to (3). The Hardware components used in RGB to YCbCr conversion are comparator, subtractor, divider, adder and hue selection.

The VHDL codes developed for these modules are pipelined and processed parallel in order to speed up the conversion process [10]. The most complex module in this conversion process is the divider block. However, the divider employed in the conversion process is highly efficient. Finally, the hue selection block selects the individual hue values computed based on the maximum pixel value among R, G, and B.

4. Implementation Results and Evaluation

4.1 Simulation

The proposed FPGA implementation of image color space conversion has been coded in Matlab and tested first in order to ensure the correct working of the color space conversion developed. The tests have provided satisfactory results. Subsequently, the entire design has been coded in VHDL and has been simulated using ISE 14.1 The Synthesis, Place & Route, and bit file generation is done using Xilinx 14.1. The ModelSim waveform results for RGB-YCbCr and RGB-YCbCr conversion is shown in Fig. 6.

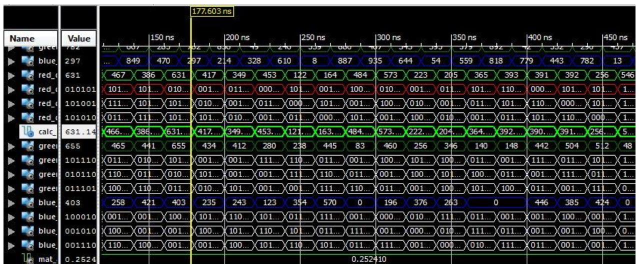

Fig. 6 Waveformsfor RGB to YCbCr and YCbCr to RGB Conversion

4.2 Color System Conversion with Xilinx System Generator

System Generator is part of the ISE® Design Suite and provides Xilinx DSP Block set such as adders, multipliers, registers, filters and memories for application specific design. These blocks leverage the Xilinx IP core generators to deliver optimized results for the selected device. Previous experience with Xilinx FPGAs or RTL design methodologies is not required when using System Generator [10,11,11].

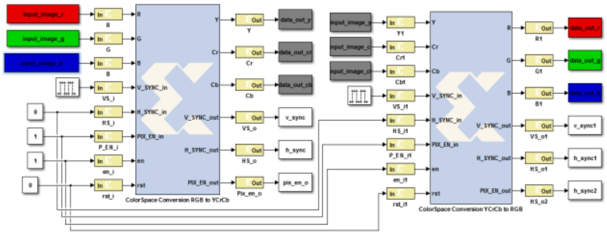

Fig. 7 Xilinx System generator model for CSC

The Xilinx CORE Generator system generates and delivers parameterizable cores optimized for Xilinx FPGAs. It is used to design high-density Xilinx FPGA devices and achieves high performance results, while at the same time, reducing the design time. The CORE Generator is included in ISE Xilinx Foundation, with a variety of cores memories and storage elements, math functions , DSP functions, and a variety of basic elements. Elements. The RGB to YCbCr and the YCbCr to RGB cores are generated by the Xilinx Core Generator14.2 and configured for 8-bit input data, 8-bit output data. FPGA Implementation of the color-space conversion IP-cores presented in Fig. 7 is composed of two IP-cores: RGB2YCrCb IP-core converts the RGB inputs to YCrCb, and YCrCb2RGB IP-core converts YCbCr signals back to RGB. The original and converted images and resulting conversion errors are plotted based on MATLAB/Simulink tools.

4.3Synthesis Results

Implementation of the proposed design was made on Xilinx Zynq and Virtex Family Platforms: XC7Z020 and XC7VX330T devices. We have used the Xilinx ISE tools version 14.1 .The synthesis results of the architecture is shown in Table 1 and Table 2.

Table 1: RGB to YCbCr Color Space Conversion Implementation using Xilinx FPGA Device

	xc7z020-2clg484		**xc7vx330t-2ffg1157**	
Logic Utilization	***Used***	***Available***	***Used***	***Available***
Number of Slice Registers	174	106400	174	408000
Number of Slice LUTs	170	53200	170	204000
Number of fully used LUT-FF pairs	146	120	146	120
Number of bonded IOBs	63	200	63	600
Number of BUFG/BUFGCTRLs	1	32	1	32
Number of DSP48E1s	4	220	4	1120
Frequency (MHz)	438.702		492.017	

Table 2: YCbCr to RGB Color Space Conversion Implementation using Xilinx FPGA Device

	xc7z020-2clg484		**xc7vx330t-2ffg1157**	
Logic Utilization	**Used**	**Available**	**Used**	**Available**
Number of Slice Registers	120	106400	120	408000
Number of Slice LUTs	99	53200	99	204000
Number of fully used LUT-FF pairs	99	120	99	120
Number of bonded IOBs	63	200	63	600
Number of BUFG/BUFGCTRLs	1	32	1	32
Number of DSP48E1s	4	220	4	1120
Frequency (MHz)	441.562		494.148	

The RGB to YCbCr and YCbCr to RGB conversions are quite simple and support very high frequencies. but they require a large number of resources. This is because each output channel is a linear combination of the inputs, so the forward conversion requires 9 multipliers and the backward conversion requires 5. The difference in the number of multipliers accounts for why the RGB2YcbCr conversion is able to operate a nearly 438 MHz, while the YCbCrtoRGB conversion can only support around 441 MHz. The utilizations are shown in Table 3 and Table 4.

Table 3 and table 4 present frame rate to decode a different FHD frames.

Table 3: Frame Rate control for RGB to YCbCr conversion

Max frequency : 438**MHz**		
Resolution	***Pixel per frames***	***Maximum Frame Rate***
1920x1080	2073600	232,92 FPS
1440x900	1296000	372,71 FPS
1024x1024	1048576	460,58 FPS
1280x720	921600	524,04 FPS
1024x768	786432	614,21 FPS
640x480	307200	1572,30FPS
512x512	262144	1842,48FPS

Table 4: Frame Rate control for YCbCr to RGB conversion

Max frequency : 441 **252 MHz**		
Resolution	***Pixel per frames***	***Maximum Frame Rate***
1920x1080	2073600	212,62 FPS
1440x900	1296000	340,2 FPS
1024x1024	1048576	420,52 FPS
1280x720	921600	478,45 FPS
1024x768	786432	560,7 FPS
640x480	307200	1435,52 FPS
512x512	262144	1682,27 FPS

5. Conclusion

In this paper, efficient architectures suitable for FPGA/ASIC implementation of RGB-HSV and RGB-YCbCr color space conversion was presented. The potential applications of the proposed techniques includes image compression, image enhancement, and segmentation etc. Pipelining and parallel processing techniques are adopted in order to speed up the conversion process. The Verilog code developed for the complete system is RTL compliant and works for ASIC design. The implementation presented in this paper has been realized on an Xilinx XC7Z020-2clg484 FPGA device. The experimental results show that proposed color space conversion approaches exhibit better performances when compared with the other existing implementations.

Thus, the designed CSC architecture is capable of encoding 232 video frames/s of high-definition TV of 1,920 pixels. Requirement of high-speed real-time image compression systems like satellite imagery, medical imaging, cartography and others is satisfied by our architecture.

Acknowledgments

The authors wish to acknowledge the approval and the support of this research study by the grant N° CIT-2017-1-7-F-6888 from the Deanship of the Scientific Research in Northern Border University, Arar, KSA.

References

[1] Shih-An Li • Ching-Yi Chen • Ching-Han Chen: Design of a shift-and-add based hardware accelerator for color space conversion : Journal of real time image processing Springer (2013)

[2] Korrapati Rajitha, Usha Rani.Nelakuditi, Venkata Naresh Mandhala, and Tai-hoon Kim, "FPGA Implementation of Watermarking Scheme Using XSG", International Journal of Security and Its Applications Vol.9, No.1 (2015), pp.89-96.

[3] Fons, F., Fons, M., Canto´, E.: Real-time embedded systems powered by FPGA dynamic partial self-reconfiguration: a case study oriented to biometric recognition applications. J. Real-Time Image Proc.—Special Issue on Real-time biometrics and Secure Media (2011).

[4] Li, S.A., Hsu, J., Wong, C.C., Yu, C.J.: Hardware/software co-design for particle swarm optimization algorithm. Inf. Sci. 181(20), 4582–4596 (2011).

[5] ITU-R, "ITU-R Recommendation BT.601-5: Studio encoding parameters of digital television for standard 4:3 and wide, screen 16:9 aspect ratios", ITU-R.

[6] P. Yuhua Y. Yang and L. Zhoanguang, "A fast algorithm for ycbcr to rgb conversion", in IEEE Transactions on Consumer Electronics, 53(4): 1490 - 1493, noviembre, 2007.

[7] Alareqi Mohammed, Elgouri Rachid, Hlou Laamari, "High Level FPGA Modeling for Image Processing Algorithms Using Xilinx System Generator", International Journal of Computer Science and Telecommunications ,Volume (5), Issue 6, June 2014.

[8] R. Naresh , R. Shivaji , Prakash J. patil ,"Authentic Time Hardware Co-simulation of Edge Discovery for Video Processing System", International Journal of Research in Modern Engineering and Emerging Technology, Vol. 1, Issue: 7, August: 2013.

[9] Sol Perdre, Tomas Krajnik, Elia Todorovich, Patricia Boensztejn, "Accelerating embedded image processing for real time: a case study", J Real Time Image DOI 10.1007/s11554-013-0353-2, May 2013.

[10] Adhyana Gupta, "Hardware Software Co-Simulation For Traffic Load Computation Using Matlab Simulink Model Blockset", International Journal of Computational Science and Information Technology (IJCSITY) Vol.1, No.2, May 2013.

[11] M. Bilal and Sh. Masud ," Efficient Color Space Conversion using Custom Instruction in a RISC Processor", IEEE International Symposium on Circuits and Systems, 2007,pp. 1109- 1112.

[12] T. Saidani , D. Dia, W. Elhamzi, M. Atri and R. Tourki, "Hardware Co-simulation For Video Processing Using Xilinx System Generator", Proceedings of the World Congress on Engineering 2009 Vol I WCE 2009, July 1 - 3, 2009, London, U.K.

Taoufik Salem Saidani received the M.S. and Ph.D degrees in Electronics from Faculty of Science of Monastir, Tunisia in 2007 and 2014, respectively. His major research interests include VLSI and embedded system in video and image compression. His current research interests include digital signal processing and hardware-software co-design for rapid prototyping in telecommunications. ; hi is currently an Assistant Professor in computer science at the Department of Computer Science as a Head of Department at the Faculty of Computing and Information Technology – Northern Border University (NBU).

Hafedh Mahmoud Zayani Said received his Ph. D in computer science and telecommunication from CNAM (CEDRIC) - France & ENIT (SysCom)-Tunisia in 2009 and my Engineer Diploma in Computer Science from National Engineering School of Computer sciences ENSI - Tunisia in 2001; hi is currently an Assistant Professor in computer science at the Department of Information System as a Head of Department at the Faculty of Computing and Information Technology – Northern Border University (NBU). My research is focused on wireless sensors networks performance optimization based on the communication protocols.