CAMERA-TO-TOUCHSCREEN DESIGN

Ing. Raissa Likhonina[1], Ing. Lukáš Kohout[2], Ing. Jiří Kadlec, CSc.[3]
[1]The Institute of Information Theory and Automation (ÚTIA)
Pod Vodárenskou věží 4, CZ-182 08, Prague 8, Czech Republic
kadlec@utia.cas.cz
[2]The Institute of Information Theory and Automation (ÚTIA)
Pod Vodárenskou věží 4, CZ-182 08, Prague 8, Czech Republic
kohoutl@utia.cas.cz
[3]The Institute of Information Theory and Automation (ÚTIA)
Pod Vodárenskou věží 4, CZ-182 08, Prague 8, Czech Republic
likhonina@utia.cas.cz

Abstract: The present paper describes an FPGA design of a camera-to-touchscreen demonstrator that has been prepared using Xilinx Vivado 2015.2 and SDK 2015.2 tools. The demonstrator consists of MicorZed 7020 Carrier Board, Avnet 7-inch Zed Touch Display and Avnet Toshiba Industrial 1080P60 Camera Module. The camera transmits a full HD video signal at 60 frames per seconds to MicroZed 7020 board, which processes it and sends to the LCD display with active area of 800x480 pixels. As the display has smaller resolution, only a fragment of the whole video frame can be seen at once on the display, whereas the full image is stored in the memory. By touching the screen one can travel along the stored video frame and look through the whole image. The design can be used, for example, as a car rear view mirror monitor benefiting from touchscreen technologies.

Keywords: Zynq, FPGA, touchscreen display, camera, full HD, SDSoC.

Introduction. Touchscreen technology has steadily come into our life. It is hard to imagine that only a few decades ago it was something unattainable. From the first finger-driven touchscreens invented by E. A. Johnson in 1965 and development of resistive and capacitive touchscreens to the latest optical touchscreen technology based on detection of an object's shadow, it did not passed much time [1], [2]. However, the rapid development of touchscreen technologies due to a large interest in this area is obvious. There is a growing number of devices that benefit from it: smartphones, tablets, computer monitors, eBook readers, game devices, GPS navigators etc. The touchscreens are foreseen to be used also in home appliances. The researchers continue working on the touchscreens that will use so called "microfluid" technology, where buttons rise up due to the fluid pressure on the covered layer when the keypad is in use [2].

The present paper aims at describing a field-programmable gate array (FPGA) design, which demonstrates one of the applications the touchscreen technology can be used for.

The design is based on two demonstrators available at Avnet's webpages. The first one, Toshiba TCM3232PB Frame Buffer Design Tutorial, shows the reference design for using Toshiba camera and getting the HDMI output. It describes step-by-step process how to setup the Toshiba camera module and run a simple design that initializes the image sensor and HDMI output interface and implements a frame buffer in the programmable logic (PL) I/Os [3].

The second design is ALI3 Display Reference Design demonstrating the capabilities of touchscreen display and presenting simple interactive GUIs, which can serve as a start point for more complex applications [4].

However, there is no design available at Avnet, which would send a video stream from camera to a display with touchscreen possibilities, so that an end-user could provide some actions with an image he/she sees on the display.

This work is aimed at combining two example designs from Avnet mentioned above in a way that it would be possible to receive a cropped image from Full HD camera on a WVGA display and observe the whole scene by moving the image, thus, benefiting from touchscreen technology.

Briefly speaking the present design illustrates how to get a video signal from a full HD camera (1920x1080 pixels) to WVGA LCD display with an active area of 800x480 pixels. A video frame is transmitted by the camera at 60 fps with pixel clock 148.5 MHz to MicroZed 7020 board. The board processes the signal and sends it to the LCD display working at pixel clock 33.33 MHz. The output signal is in RGB format. The smaller display resolution results in video cropping. Thus, only a fragment of the whole frame is seen on the display, whereas the full image is stored in the memory. By touching the screen one can move the image and look through the whole picture.

The scene of interest can be then zoomed if extending the presented design with two finger interaction and zoom capabilities. The design can be used as a car rearview mirror or for home door entry applications.

Moreover, it is worth to note that the camera-to-touchscreen design benefits from FPGA technology. FPGA is an integrated circuit enabling a designer to configure it after its manufacturing. It is widely used in common embedded applications. To create an FPGA design, Xilinx tools – Vivado 2015.2 for a hardware part and SDK 2015.2 for the software - are required.

Methodology. The camera-to-touchscreen design described in the paper is created using the following HW components:

- MicroZed Embedded Vision Carrier Card and MicroZed 7020 board with a processing system with two Cortex A9 cores and a 28 nm programmable logic. It includes two DDR3 memory components totaling 1GB of random access memory [4], [3].
- Avnet Toshiba TCM3232PB full HD color image sensor capable of delivering a video signal at 60 fps. Two technologies are implemented there: High Dynamic Range (HDR) technology and Color Noise Reduction (CNR) technology [3].
- Avnet 7-inch WVGA TFT-LCD display with an industrial projective capacitive touch sensor. The display has an active area of 800x480 pixels, frame rate of 60 Hz, pixel rate of 33.33 MHz and pixel format 24 bits RGB [4].

The hardware design is prepared in Xilinx Vivado 2015.2. This software is purposed for synthesis and analysis of HDL designs and suitable for system on a chip (SoC) development as well as for high-level synthesis, which enables C, C++ and SystemC programs to be directly targeted into Xilinx devices without manually creating RTL.

Using IP Integrator and a list of available IP cores the hardware design has been created. The IP cores used in the design are shown in Figure 1.

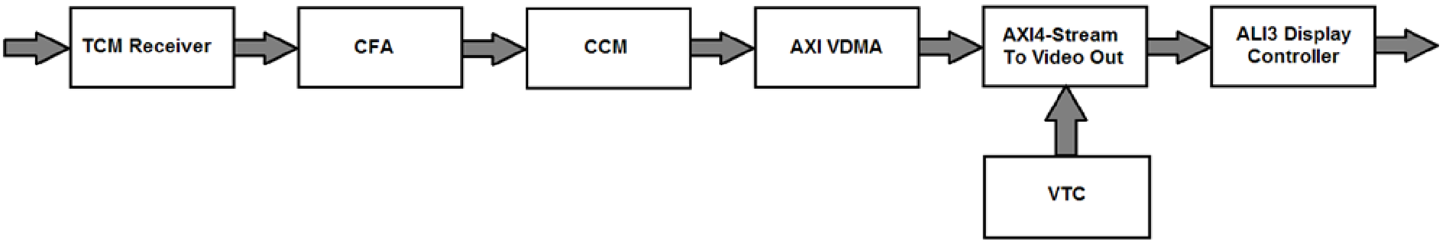

Fig. 1. Hardware block diagram

The camera sends frames coded in Bayer matrix, received by TCM receiver at 24 MHz. These frames pass through Color Filter Array Interpolation (CFA) and Color Correction Matrix (CCM) blocks.

CFA aims at reconstructing the missing color components of an image obtained from an RGB or CMY Bayer filtered sensor by means of interpolations using information from neighboring pixels. This process is called CFA demosaicing [5]. In the present design it also converts a video signal to RGB color space.

CCM is used for adjusting white balance, color, brightness or contrast in an image. It multiplies the pixel values with some coefficients, which strengthen or weaken them. Mathematically it is expressed as a 3x3 matrix multiplication. The weights in equation define a color-correction matrix. The example of color-correction matrix for RGB data can be seen below. For more information refer to [6].

$$\begin{bmatrix} R_C \\ G_C \\ B_C \end{bmatrix} = \begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} O_1 \\ O_2 \\ O_3 \end{bmatrix},$$

where R_C, G_C, B_C are corrected colors for RGB input data, K_n are weights, O_n are offsets used for achieving black levels [6].

Video Direct Memory Access (VDMA) writes to and results from DDR3 24-bit RGB frames. This core provides handling three frame buffers with internal lock between them to avoid image tearing. VDMA output is sent to Video Out block to map video data on output timing coming from Video Timing Controller (VTC). ALI3 Controller passes the video signal through physical interface to display with 33 MHz pixel clock. Communication between the blocks runs at clock rate of 150 MHz, while peripheral configurations use 100 MHz clock speed. The Zynq processing system performs the whole video chain initialization. The image processing operations are fulfilled outside the processor. It contributes to fast functioning of the system. To detect touches on display the system uses interrupts. The interrupt routine analyzes it and performs appropriate actions (a video frame shift in certain direction on the display).

The next step after finishing the design is to validate it. The validation process helps to find the errors in the design that could prevent the hardware from working properly. The most frequent errors can appear in connections between blocks or in parameter settings for individual blocks.

If validation is successful, then so called HDL (Hardware Description Language) wrapper can be generated. It is basically a top-level description of the system. The synthesis process in its turn will generate all source files for the IP blocks as well as any relevant constraints files.

After design implementation, i.e. placing and routing the netlist onto the FPGA device resources, and generation of a bitstream file with configuration data for implementation in the PL, the building of the hardware image is complete and the hardware platform can be exported to SDK (Software Development Kit) environment, where different software applications can be created.

As a basis for software development of camera-to-touchscreen application the previously mentioned designs from Avnet are used [4], [3].

Results. The complete camera-to-touchscreen demo kit can be seen in Figure 2 [7]. Its components are the same as it was described in previous chapters.

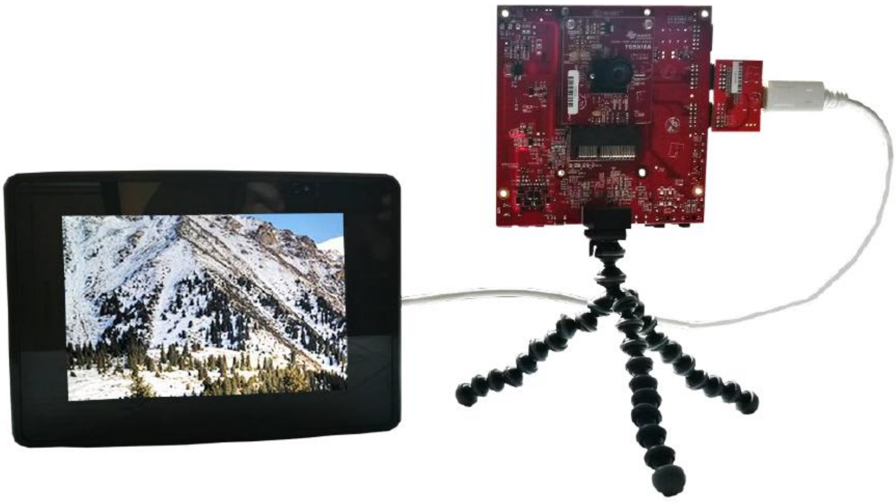

Fig. 2. Complete camera-to-touchscreen demo kit

The resultant software is programmed in such a way, that it enables to transfer a full HD image from TCM receiver to touchscreen LCD display with resolution 800x480 pixels at frame rate 60 fps. By touching the screen we can see the whole picture part by part. After interrupt occurs, the processor knows that a touch event has taken place and evaluates the concurrent touch events in a way, that it saves X and Y coordinates of two concurrent events, and make subtraction: X=X2-X1, resp. Y=Y2-Y1. If there is 0 for some axis, then there will be no movement in this direction. If X is a non-zero variable, the image will move for 40 pixels left or right depending on a variable sign. The same is valid for Y, but the shift is down/up. The directions of movement are similar one has in his touchscreen mobile phone [7]. The number of pixels for moving in each direction has been taken optionally after testing image incremental movement. The chosen number is optimal for demonstrating the possibilities of the design, but it can be changed if necessary. The example of the camera-to-touchscreen demo functioning is shown in Figure 3.

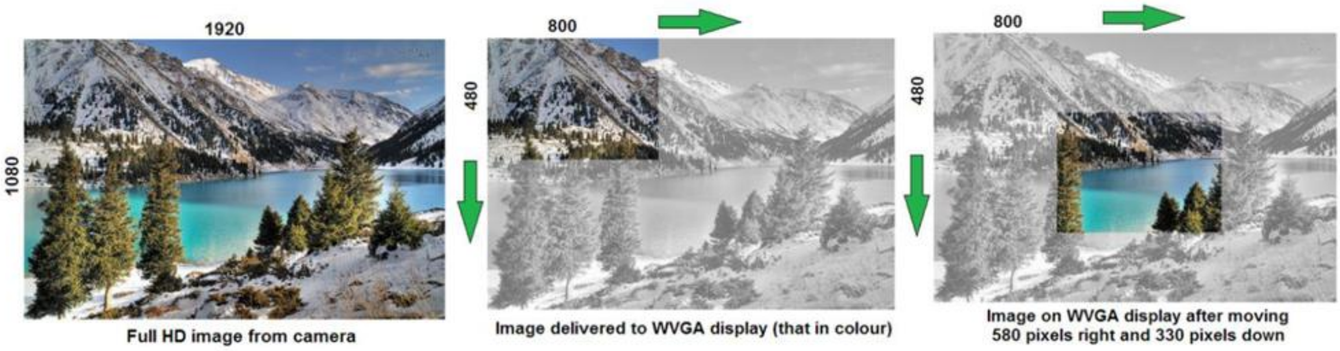
1920
1080
Full HD image from camera
800
480
Image delivered to WVGA display (that in colour)
800
480
Image on WVGA display after moving
580 pixels right and 330 pixels down

Fig. 3. Hardware block diagram

Discussion. The present design demonstrates a camera-to-touchscreen technology, which can be used for different applications.

It can be complemented with more sophisticated designs such as face detection, vehicle recognition, edge detection, optical flow designs and many others. For these purposes the SDSoC (Software-Defined System On Chip) environment is a good tool to use.

The SDSoC is a system compiler, which targets a base platform and is capable to compile C/C++ functions into programmable logic. The system compiler works one level above the Vivado HLS compiler. After analyzing a program and determining the data flow between software and hardware functions, it generates an application specific SoC including a complete boot image with firmware, operating system, and application executable. Compilation itself is performed by the Xilinx HLS compiler. HLS compiles the transformed C/C++ to the HDL code. The HDL code and the corresponding cores are automatically packed into the IP-XACT format and serve as input for Vivado IP integrator. The Xilinx SDSoC environment also automatically generates compatible data-mover IP-cores and the interface IP-cores for the programmable logic part of the ZYNQ device. This can result in automated generation of new SoC system with new HW accelerators, which replace the original SW based system. It can reduce the energy per pixel in case of video processing algorithms [8].

It is also worth to note that the SDSoC system includes OpenCV libraries, which comprise different mathematical functions such as Gaussian, Median, Bilateral, Canny edge detection, SVM, LK Optical Flow and etc. [9]

Moreover, there is still a space for improvements and optimization for the presented design. One of the directions for improvement is to solve the problem of a false touch event registration, which was not fully managed in the present demonstrator. Though the design functions satisfactory for demonstration purposes, in future it would be desirable to get rid of these errors by making more precise calibration and, perhaps, by implementing some additional filtering.

Besides, the display supports two finger touches and gesture recognition. It means that there is a possibility to use the image scaling in a way we are used to making it in our mobile phones. This function could be a useful extension of the presented demonstrator bringing further benefits for some applications including, e.g., home entry systems or in-car systems.

There are also other possibilities, which can be profitable for some applications. For example, it is possible to connect one more block, on Screen Display block, in the hardware design, that enables alpha blending and composition of external video inputs. It supports several layers, so the user can configure multiple input video sources. Each video source layer can be displayed at different cropped sizes, positions, and transparency [10].

Conclusion. The present paper describes camera-to-touchscreen-display design, which allows getting a full HD image from camera module and sends it to 800x480 display, so that on display there is only a part of the whole image, but the full image is stored in memory. We can see the whole image part by part by touching display and so moving the image right/left or/and up/down. The mathematical models are the part of system generation process and are hidden in the separate blocks of the design. It can be also complemented with complex mathematical functions for edge detection, optical flow or image recognition applications using SDSoC environment, which compiles C/C++ functions for the target platform.

The present design is based on two original designs for the camera and for the display from Avnet, Inc. However, the Avnet's designs do not allow getting the image from camera to display and move along it as it was described in the present article.

In this paper the principle of design was presented in more details. Besides, this paper summarizes the strengths and drawbacks of the design, and names some other possibilities and space for improvements as, for example, to get out of false detection of touch events, to add an image scaling or even to use On Screen Display and benefit from several layers for multiple input video sources, to use complex algorithms for edge detection and image recognition capabilities and etc.

Such kind of displays can be used as a car rear viewer with more functionality or, for example, in house door entry systems. The application field of such displays can be quite broad and is not limited by above mentioned examples. There is a space for engineers' imagination.

Acknowledgement. This work has been supported from project ALMARVI, project number ARTEMIS JU 621439 and MSMT 7H14004.

References

1. Baanto, "Types of Touch Screen Technologies" [online]. Available at http://baanto.com/types-of-touch-screen-technologies
2. Ion F., "The past, present, and future of touch: From touch displays to the Surface: A brief history of touchscreen technology", ARSTechnica, pp. 1-3, April 2013 [online]. Available at http://arstechnica.com/gadgets/2013/04/from-touch-displays-to-thesurface-a-brief-history-of-touchscreen-technology/1/
3. Avnet Electronics Marketing, "Toshiba Industrial 1080P60 Camera Module: Getting Started Guide", Version 1.2, pp. 4-12, April 2015 [online]. Available at https://zedboard.org/content/getting-started-guide-3 - 2
4. Avnet Electronics Marketing, "MicroZed Embedded Vision Carrier (EMBV): ALI3 Display Reference Design and Tutorial", Version 2014.4, pp. 1-4, June 2015 [online]. Available at http://zedboard.org/product/microzed-embedded-vision-kits
5. Xilinx, "Color Filter Array Interpolation v7.0. LogiCORE IP Product Guide", PG002, pp. 4-7, November 2015 [online]. Available at http://www.xilinx.com/support/documentation/ip documentation/v cfa/v7 0/pg002 v cfa.pdf
6. Xilinx, "Color Correction Matrix v6.0. LogiCORE IP Product Guide", PG001, pp. 4-6, November 2015 [online]. Available at http://www.xilinx.com/support/documentation/ip documentation/v ccm/v6 0/pg001 v ccm.pdf
7. Likhonina R., Kohout L., Kadlec J., "Camera to Touchscreen Demonstration for MicroZed 7020 carrier board, Avnet 7-inch Zed Touch Display and Avnet Toshiba Industrial 1080P60 Camera Module", ÚTIA, 2016
8. Kadlec J., Pohl Z., Steven van der Vlugt, Jääskeläinen P., Koskinen L., „Algorithms, Design Methods, and Many-Core Execution Platform for Low-Power Massive Data-Rate Video and Image Processing", Almarvi, 2016.
9. Xilinx, "SDSoC Environment User Guide", UG1027 (v2017.2), pp. 6-9, August 2017 [online]. Available at https://www.xilinx.com/cgi-bin/docs/rdoc?v=2017.2;d=ug1027-sdsoc-user-guide.pdf
10. Xilinx, "Video On-Screen Display v6.0. LogiCORE IP Product Guide", PG010, pp. 4-7, November 2015 [online]. Available at http://www.xilinx.com/support/documentation/ip documentation/v osd/ v6 0/pg010 v osd.pdf