Miron KŁOSOWSKI

Gdańsk University of Technology

Wireless intelligent audio-video surveillance prototyping system

Abstract. The presented system is based on the Virtex6 FPGA and several supporting devices like a fast DDR3 memory, small HD camera, microphone with A/D converter, WiFi radio communication module, etc. The system is controlled by the Linux operating system. The Linux drivers for devices implemented in the system have been prepared. The system has been successfully verified in a H.264 compression accelerator prototype in which the most demanding algorithms like DCT, inter-prediction and intra-prediction have been implemented in the hardware.

*Streszczenie. Przedstawiony system bazuje na układach FPGA Virtex6 i dodatkowych układach jak: szybka pamięć DDR3, mała kamera HD, mikrofon, moduł radiowy WiFi, itp. Opisany system prototypowania został pomyślnie zweryfikowany poprzez realizację prototypu akceleratora kompresji video H.264 w którym najbardziej wymagające algorytmy jak DCT, inter-predykcja i intra-predykcja zostały zaimplementowane sprzętowo. (**System do prototypowania bezprzewodowych inteligentnych urządzeń monitoringu audio-wideo**).*

Keywords: wireless surveillance system, video compression, FPGA prototyping system, H.264.
Słowa kluczowe: bezprzewodowy system monitoringu, kompresja video, system prototypowania FPGA, H.264.

Introduction

The rapid design of embedded systems is possible only when a carefully optimized prototyping platform is available [1]. The presented development system has been especially designed for the applications where audio-video processing is important. The another important feature is a wireless communication capability through a WiFi radio module. Thus the presented prototyping system is suitable for the applications like:
– prototyping and implementation of a hardware accelerated algorithms for HD video and audio compression,
– prototyping and implementation of a machine vision systems,
– prototyping and implementation of a parallel data processing systems.

The presented wireless surveillance system has been used for a development of hardware acceleration for the H.264 video compression standard [2]. The Microblaze processor with the PLB (Processor Local Bus) and the common memory interface for accelerator modules has been implemented in the Xilinx Virtex-6 FPGA.

Linux has been chosen as an operating system for the board. The x.264 encoder software [3] has been installed and modified for a hardware acceleration.

The hardware acceleration has been achieved for an integer DCT transform, inverse integer DCT transform, Hadamard transform, quantization, dequantization, intra-prediction, inter-prediction and CAVLC. An ASIC version of the H.264 accelerator (technology 90nm) has also been designed. It can be easily integrated with the development board using the expansion sockets.

Prototyping system description

Photo of the prototyping system is presented in Fig. 1. Block diagram of the prototyping system is shown in Fig. 2.

Main board:
One of the largest FPGAs offered by Xilinx: XC6VLX365T-1FFG1156C from the Virtex6 family have been chosen to be a heart of the prototyping system. The FPGA can be configured in two ways: by a JTAG socket (slow method) or by a 64 MB flash parallel memory PC28F512P30TF (fast method). Additionally the flash memory may contain a bootloader, operating system, application data and can also be easily modified by the software. A dual FPGA configuration mechanism is also implemented and allows automatic fail-safe boot of the system. The configuration process of the FPGA is controlled by the microcontroller which is also used for power control and measurement of power voltages and currents. The configuration bitstream can be encrypted – a battery socket for the FPGA decryption key memory is supported. An additional reset IC controls reset signal assertion after power on and when the reset push button is pressed. There is also a LED for the configuration process status signalling. A 100 MHz oscillator with differential output is connected to the FPGA to serve as a local clock source.

Fig.1. Photo of the prototyping system

The main 256 MB system memory is composed of two 16-bit DDR3 memory modules: MT41J64M16LA-187E. The memory works with a 400 MHz clock. The memory bus termination power can be disabled and the memory can be switched to a self-refresh mode – this way the system can be easily hibernated and the FPGA power can be switched off to save the power.

The system has got a RS232 RJ-45 YOST standard socket for a console. It can be used for development, diagnostics and configuration of the system. Two input and

two output wires are available (supporting voltage level conversion for the RS232 standard).

A small serial EEPROM memory (16kb) is connected to the FPGA to store the parameters of the application and operating system (for example the IP address). The connection is based on the I^2C bus.

Two bi-directional level converting 8-bit buffers are included for the system extension. The external voltage input can be used for the desired logic level conversion. In the sample system these buffers are used for the external Ethernet-PHY circuit.

A socket for the Micro-SD flash memory has also been implemented. It has been connected to the FPGA through a fast logic level converter. The Micro-SD can be used in the one-bit or four-bit data communication mode.

A/D converters which are integrated with the FPGA are accessible through an additional connector. There are four differential analog channels with simple antialiasing filters.

For the communication RS9110-N-11-22 WiFi module has been utilized. This module is a complete IEEE 802.11bgn based wireless device server that directly provides a wireless interface using the SPI interface for data transfer. It integrates a MAC, baseband processor, RF transceiver, power amplifier, frequency reference, all WLAN protocol and configuration functionality, networking stack in embedded firmware to make a fully self-contained 802.11n WLAN solution for a variety of applications. The external antenna is connected through a small coaxial socket.

For easy diagnostics 8 LEDs and 4 switches are available. There are also 2 outputs for relays and three connectors for extension boards. Two connectors for a digital ASIC with 166 pins total and one connector for an analog ASIC with 40 pins. The main board has been fabricated on a 10-layer PCB with following dimensions: 153 mm x 87 mm.

Front board and camera board:

In the prototyping system a Sony MCB1172 8.08 Mega pixels digital camera module has been used. The 9.5 x 7.1mm camera module delivers 30fps at 720p resolution. The camera's features include movie stabilization, face detection, auto-focus and motion auto-focus. For the applications which need higher resolution, the camera module can provide up to 8 Megapixel still images with related features including a 16x digital zoom, image stabilizer, short interval shooting capabilities, an action capture mode and backlight offset. The 1/3.2-inch type camera module consists of a lens, a feature-rich DSP and a high quality Sony CMOS sensor. The progressive scan CMOS sensor delivers video support with YCbCr video or JPEG still image output.

In the front board a MEMS microphone with an amplifier and an A/D converter has been installed. The microphone type is WM7120A. An audio level measurement circuit for the system wake up has also been included.

The front board has been also equipped with several sensors: a digital PIR sensor PYQ 2898, a digital temperature sensor DS18B20 and a digital color light sensor TCS3414CS. Eight IR LEDs for the observed object illumination and three LEDs (RGB) for the camera status display have been added.

Power source board:

The system works with an external power source (valid voltage range: 6.5 V to 13 V). The power source board generates the power for the FPGA, memory, camera and other devices. It also generates standby power for the microcontroller Atmega1280 and for the sensors on the front board – this allows an automatic wake-up on temperature, PIR, audio or light sensor activity. Additionally the WiFi module can be powered during standby which makes a remote wake-up of the device possible. Software for the microcontroller supports functionality of a wake-up timer. The microcontroller communicates with the FPGA serially using two wires. This channel is used by the FPGA to configure the power source board, setup wake-up events and timers, read sensors data, measure voltages and currents, switch off the power and to send the reset signal to the WiFi module.
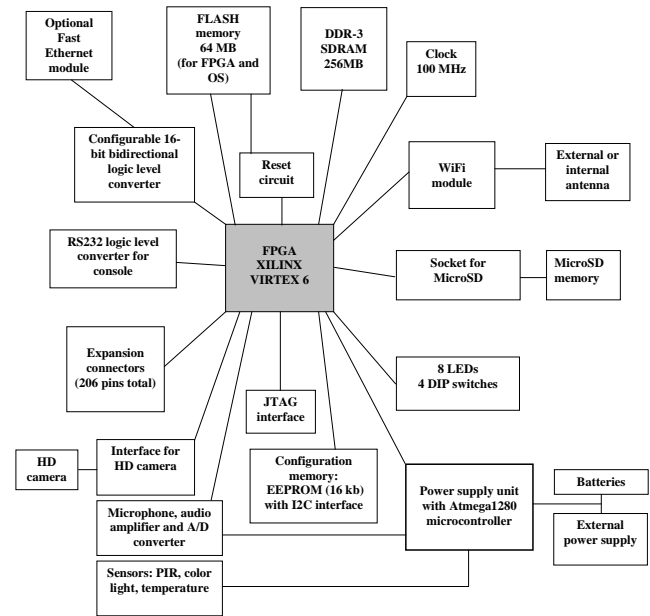

Fig.2. Block diagram of the prototyping system

Prototyping H.264 video compression

Described prototyping system has been used for the prototyping of an accelerator for the H.264 video compression. Fig. 3 depicts the block diagram of the H.264 video compression accelerator. The algorithms for which it is difficult to achieve good performance in the software have been implemented in the hardware. An open source encoder application x.264 has been used for the acceleration. After profiling the functions suitable for the hardware acceleration have been chosen and the accelerator algorithms have been defined. The accelerator can be used with any other encoder application because only the H.264 standard algorithms have been accelerated. The application works on the Microblaze processor under control of the Linux OS. The processor and the accelerator share the same 100 MHz clock.
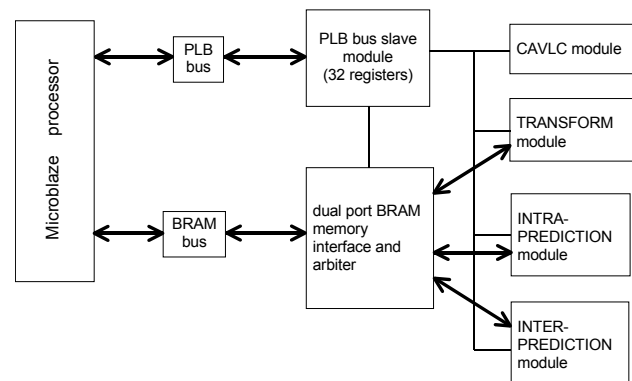

Fig.3. Schematic diagram of the prototyped H.264 hardware accelerator

Transform accelerator:

The transform accelerator calculates a difference between a macroblock and its prediction and performs an integer version of DCT (Discrete Cosine Transform). Then a quantization process on the resulting coefficients is performed and the results in the zig-zag sequence are returned. Additionally a decimation functionality has been implemented to improve the compression with a minimal impact on quality [4]. A dequantizer and inverse DCT are then automatically started. The results of an IDCT are buffered and can be later used by the application software for prediction. The DC part of the transform is processed by the separate hardware path: a Hadamard transform, DC quantization and optional DC dequantization and DC version of IDCT.

An average 16x16 macroblock encoding takes 20 µs when using the hardware accelerator or 430 µs when using the software (Microblaze processor).

Table 1. FPGA resource usage for the transform accelerator

Resource	Use	Available
Slices	2486	56880
Slice REG	3500	455040
LUTs	7705	227520
LUTRAM	192	66080
BRAM	10	416
DSP48E1	4	576

Intra-prediction accelerator:

The intra-prediction accelerator has been developed for the 16x16 and 4x4 macroblocks. An error is calculated using one of the configurable methods: SSD (sum of squared differences) or SAD (sum of absolute differences). It is possible to read the results of all the prediction modes and/or only the best result (with minimum error). The accelerator may be configured to write the best prediction of the macroblock to the memory.

An average 16x16 macroblock intra-prediction takes 2.5 µs when using the hardware accelerator or 55 µs when using the software (Microblaze processor).

An average 4x4 macroblock intra-prediction takes 4 µs when using the hardware accelerator or 27 µs when using the software (Microblaze processor).

Table 2. FPGA resource usage for the intra-prediction accelerator

Resource	Use	Available
Slices	1471	56880
Slice REG	2023	455040
LUTs	4034	227520
LUTRAM	0	66080
BRAM	0	416
DSP48E1	24	576

Inter-prediction accelerator:

This accelerator performs a motion estimation function. The best prediction calculation is performed using a local cache memory to which a source macroblock and a search region have to be downloaded. The search region is a square composed of 64x64 pixels. Data can be downloaded to the search region on demand during a search process. The accelerator usage algorithm is presented below:
– Configure the accelerator (set options, addresses, etc.).
– Load the cache memories (with macroblock and with search region).
– Set the search coordinates list (maximum 16 coordinates for SAD or SSD error calculation).
– Start the search operation (the predictor uses an internal memory during the search so it is possible to use other accelerator modules during the search process).

– The results are written to the FIFO and can be accessed as soon as available.
– After the errors for all coordinates are calculated it is possible to read the position of the best result (minimum error).

An average macroblock inter-prediction takes 51 µs when using the hardware accelerator or 164 µs when using the software (Microblaze processor).

Table 3. FPGA resource usage for the inter-prediction accelerator

Resource	Use	Available
Slices	350	56880
Slice REG	637	455040
LUTs	870	227520
LUTRAM	18	66080
BRAM	2	416
DSP48E1	4	576

CAVLC accelerator:

This accelerator module performs CAVLC (Context-adaptive variable-length coding) algorithm. It uses a PLB interface only for communication with the operating system. Internal CAVLC core is started when eight 32-bit words of data is transferred to the input FIFO of the accelerator. When all partial results are available an interrupt is generated and the encoded data can be read from the output FIFO. This module performs CAVLC operation about 1.5 times faster than software version of CAVLC encoder.

Conclusion

The presented prototyping system has been successfully verified as a platform for the prototype of a H.264 accelerator. This accelerator gives noticeable speedup of the H.264 compression algorithms. It can be used as a prototyping system in any application using a high definition camera.

Possible applications of the prototype system are not limited to an audio-video device prototyping, it can also be used for a High Performance Computing (HPC). 32 samples of the prototype system have been fabricated and they can be easily configured to communicate with computing server unit via the WiFi interface. If the accelerated algorithms do not expose intensive communication demands the WiFi interface may have enough throughput for a data exchange between the computing server and the accelerator modules.

REFERENCES
[1] Chunmiao Y., Zhigang J., Qingyong Y., A Prototype of H.264-Based Remote Video Surveillance System, *Intelligent Networks and Intelligent Systems ICINIS '09 Second Int. Conf. on*, (2009), 334-337.
[2] ITU-T Rec. H.264, ISO/IEC 14496-10 (MPEG4-AVC),, Advanced Video Coding for Generic Audiovisual Services, v1, May, 2003; v2, Jan. 2004; v3 (with FRExt), Sept. 2004; v4, July 2005.
[3] x.264 encoder, http://www.videolan.org/x264.html
[4] JVT-B118 - Joint Video Team of ISO/IEC MPEG and ITU-T VCEG; WD-2, Rev0; 2002-03-13.

Author: *dr inż. Miron Kłosowski, Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki, Katedra Systemów Mikroelektronicznych, ul. Narutowicza 11/12, 80-233 Gdańsk, E-mail: mkl@ue.eti.pg.gda.pl.*