FPGA based System on Chip IPTV Set Top Box

P Kaythry K J Jegadish Kumar
Dept. of Electronics and Communication Engineering SSN College of Engineering, Chennai.

Summary
IPTV is mainly used for Live Streaming, Time Shifting and provide Videos on Demand (VOD). Current Ethernet transmission system on chips (SoC) designs supports 100 Mbps data rate and is transmitted as packets through the ethernet cable. Generally, processor accesses the memory and stores the data in the form of frames. In this paper, an Ethernet with High Definition Multimedia Interface (HDMI) SoC design is proposed to provide high quality picture transmission at the rate of 1Gbps. The encoded video data from the Ethernet is processed by HDMI video decoder in such a way to provide the maximum quality of 1080p. The System is implemented in Zedboard Zynq evaluation kit with FPGA device part number xc7z020clg484-1 using Vivado tool. The implementation cost 7.88% of Look Up Tables(LUTs) and 4.84 % of Flipflop(FF) in the device. The power consumption of the device is estimated to 1.783 watts.

Keywords:
System on Chip, IPTV, HDMI, Set Top Box.

1. Introduction

Television (TV) services have evolved from traditional linear TV to a surplus of new streaming services with the advent of Internet Protocol TV (IPTV). IPTV is delivering television program through different network technologies. IPTV provides some extra advantages such as ability to support video on demand (VOD), ability to integrate TV program with some other IP based services and the ability to respond customer's interactivity. IPTV services are grouped into two types Live TV and VOD. Live TV uses Internet group management protocol (IGMP) to connect a multicast stream whereas VOD uses (Real Time Streaming protocol (RTSP) to transmit media stream. For playback of TV program requires a either a personal computer or a Set top box.
This paper describes the design of an IPTV set top box with high speed data transmission rate of 1Gbps and to support low power HDMI 1.4 video display. The paper is organized as follows. Section II describes the problem statement and related works. The proposed work is described in section III. The design and implementation details are found in section IV. Results and analysis are discussed in section V. Finally in section VI the conclusion and future work are stated.

1.1 IPTV

IPTV (IP Protocol TV) is an evolution from traditional to advanced TV based on digital network. In IPTV the content received through ethernet are shown in user display. The contents of IPTV are usually protected and users access the channel through STB which is connected to the display. Either the users, use STB to access the channel or use the display and STB that are integrated in a single device (Smart TV). IPTV is not similar to WebTV, where users watch video over the Internet. There are some difference between WebTV and IPTV. For instance, WebTV is not limited to specific group of users as in case of IPTV. In WebTV, the video streaming are sent through internet, but in IPTV the services are limited to particular group of users and it is limited by geographic regions where the operator works [1]. In WebTV services, the contents are not protected, users can able to visualize them without any restriction. In IPTV systems, the contents are usually protected from other users and it can be accessed only through STB. IPTV systems support bidirectional communication which allows receiving information from users.

1.2 STB

A set- top box is a device that connects television to broadband network for media processing. Traditional set-top box provides service only for television. But IPTV set-top box serves dual purpose of providing service for Television as well as Personal Computer. The services provided by traditional Set Top box are Video on Demand (VOD), Electronic Program Guide (EPG), Personal Video Recording (PVR) and Linear TV whereas the services provided by IPTV set top box are Web browsing, Email, instant messaging (IM) and advanced multimedia codec[2,3]. The key elements for STBs are the service provider and the requirements of that service provider to define price and the mechanism to protect contents, software updates etc.

2. Proposed Work

In this paper, an Ethernet with HDMI SoC design is proposed for IPTV set-top box with gigabit data rate. Zynq

Manuscript received February 5, 2017
Manuscript revised February 20, 2017

7020 SoC in Xilinx is adapted for the implementation. The SoC has two partitions namely the processing system and the programmable logic. The Zynq 7020 evaluation kit has the ARM Cortex A9 processor and other hard core Intellectual properties namely Accelerated Processing Unit (APU), Universal Serial Bus (USB), Universal Asynchronous Receiver/Transmitter (UART), Secure Digital (SD) card and Ethernet card. The programmable logic consists of configurable logic blocks, Digital Signal Processor (DSP) slices and Block Random Access Memories (BRAMs).

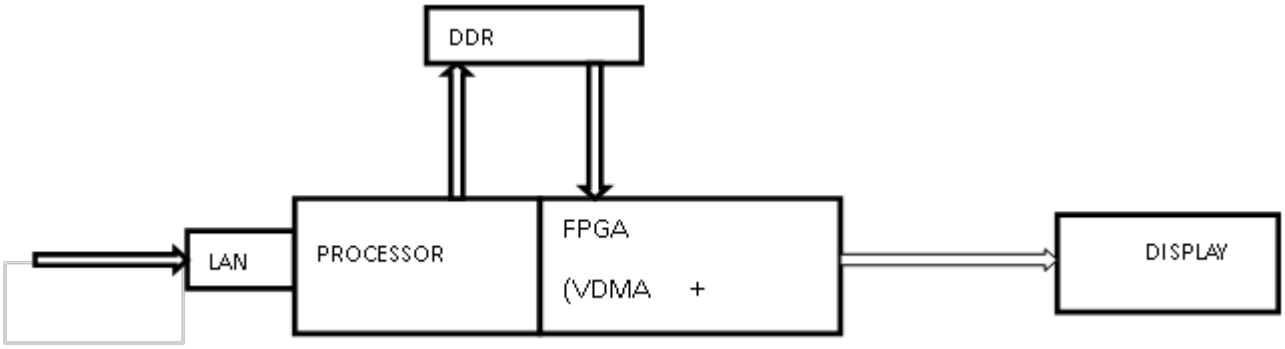

Fig 1: Block diagram of IPTV

Working process of IPTV is shown in figure 1 Processor receives the Internet Protocol (IP) packets via Ethernet and stores the received packets in Double Data Rate (DDR) memory. Then the processor initiates the Direct Memory Access (DMA) engine. The Video Direct Memory Access (VDMA) engine starts reading the stored data in DDR and sends it to the HDMI interface [4].

2.1 Hardware Design of Proposed Work

The video frames are stored in SD card in binary file format. The size of the binary file depends on the number of frames and the frame resolution. For the analysis of the proposed work, a video frame of 1080 pixels and the colour depth of 24 bits are considered. Hence, the size of a single video frame is calculated as 1080*1920*24=6075Kilo bytes (KB). The maximum capacity of DDR is 512Mega Byte (MB) out of which 256MB is utilized for program memory and data memory whereas the remaining 256MB is utilized for holding video frames. This 256 MB can hold about (256*1024)/6075=43.15 frames. The processor copies the video frames from the SD card to the DDR memory using Advanced eXtensible Interface (AXI) [7,8]. The DMA engine has to read the stored video frame from DDR memory. The video frames are temporarily stored in the DDR in order to speed up the entire process. If the DMA engine tries to read directly from SD card, the data retrieval process becomes slow due to long inter-chip communication.

The VDMA engine reads the frame data from the DDR and holds it in a small buffer before transferring it to the HDMI display controller. The pixel clock for the display controller is 82.5MHz, but the processor clock is 660MHz. The processor is much faster compared to the display controller. Due to the high operating frequency of the processor, it can meet the data requirement of the display controller and also monitor other peripherals like ethernet, UART and SD card.

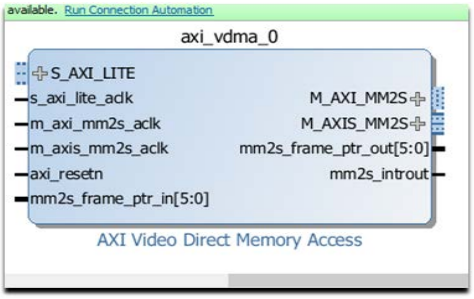

Fig. 1: Motion Estimation Hardware Module

2.2 VDMA

VDMA is preferred, whenever there is a requirement of high speed transfer to the peripheral outside the chip. .Direct memory access is initiated by the processor. Once initiated, the controls of the memory buses are taken over by the peripheral. In this case DMA of DDR3 is used for providing high speed data transfer to the display control unit. VDMA is a special kind of DMA IP, which is designed to support video applications. VDMA once programmed by the processor, continuously reads data from the DDR3 memory and transfer it to the display control unit. The size of the DMA buffer varies with the resolution of display. As the resolution increases, more data buffering is required.

2.3 VDMA Hardware

S_AXI_LITE port is connected with M_AXI_LITE port of the processor and it configures the control registers of VDMA. M_AXI_MM2S is connected to DDR and M_AXIS_MM2S is connected to HDMI hardware [5].

2.4 HDMI Hardware interface with VDMA

The HDMI hardware is packaged with slave AXI stream interface (S_AXI_MM2S). The Master (M_AXI_MM2S) port of VDMA is connected to slave (S_AXI_MM2S) port of HDMI hardware. VDMA receives the data from DDR through M_AXI_MM2S and HDMI hardware receives data from VDMA through M_AXIS_MM2S [5].

2.5 VDMA Configuration

The video frame buffer start address is written to the START_ADDRESS register. Then the horizontal size of the frame is written to the HSIZE register and the vertical size of the frame is written to the VSIZE register. The start bit VDMACR.RS is set to 1. All the registers have predefined offset from the base address which defined in VDMA product guide and then processor writes in the register [6].

3. Implementation

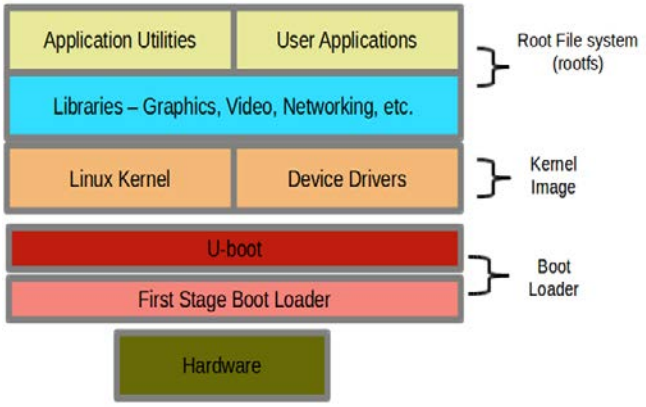

Fig 3: Linux Components

The figure 3 shows the hardware module to estimate the motion of video frames, in which the bit streams are generated using vivado tools. The hardware configuration information is exported to the software development kit (SDK) as hardware definition file. This hardware definition file holds information about device start and end addresses are mapped to DDR memory [9].

The bare metal code or Linux operating system (OS) software runs at the top level of the hardware. Since, the bare metal code has control over a single core of the processor. However, the scheduler in the operating system overcomes the bottleneck by managing the load and distributing over the processor cores. The OS used in the proposed design implementation is Linux based xillybus. The default kernels such as network driver, DDR controller, SD card driver are available in the OS. In addition to this, Kernels for HDMI display hardware are developed separately and then added to the kernel. The files required to run xillybus OS are the kernel image, device tree, first stage boot loader (FSBL) and second stage boot loader (SSBL). The kernel image is the underlying drivers to run the O.S. The device tree provides the information to partition the RAM area and procedures to partition. Then these partitioned sections are mapped to hardware resources. The FSBL loads the bit stream in the Field programmable gate array (FPGA) and transfers the control to the second stage boot loader. Then, the stage two boot loader loads information such as network address, gateway address, netmask, and physical address of the hardware and transfers the control to the kernel.

3.1 Bare Metal To Linux Platform Migration

Bare metal core occupies single core to manage updates of Ethernet and video memory. However, Linux O.S uses dual cores with inbuilt scheduler to have load sharing between the cores. To migrate, First stage boot loader(FSBL), Second stage boot loader (SSBL), FPGA bit stream , device tree, kernel image linux files are required to be stored in the memory. In Xilinx design flow, the FSBL is build and inputs bit stream files and system hardware project file. Then, U-boot loader for the target platform is built successfully depending on the external building device tree compiler (dtc). The input files config.mk (MicroBlaze) and xparameters.h (MicroBlaze, PPC) are required to produce output files U-Boot and mkimage. A Root file system (Rootfs) file describes how to build a RAM disk or init.rd image files. To build RAM disk, the input files required are ramdisk.image.gz (Zynq AP SoC) and initramfs.cpio.gz (MicroBlaze and PowerPC), produces output file uramdisk.image.gz.

VDMA device drivers with options/ probe install, device remove/install, device node string, Registers (HSIZE, VSIZE, and START_ADDR) read/write. In the device tree, the list of devices installed to the processor is listed and each device is identified by the device node string and the associated RAM address. The device tree is updated with the VDMA device information. In the Boot loader, the BootROM looks for any bootable devices (SD card or Flash memory). Jumper settings are set to SD card, and the processor starts executing the code from FSBL. This FSBL loads bits streams into FPGA at the time system boot and initializes the processor with processor register initialization codes. Later, FSBL hands-off the control

SSBL. FSBL is configured using Xilinx software development kit. The SSBL configures RAM partition and networks.

4. Results and Analysis

4.1 FPGA Device Floor plan

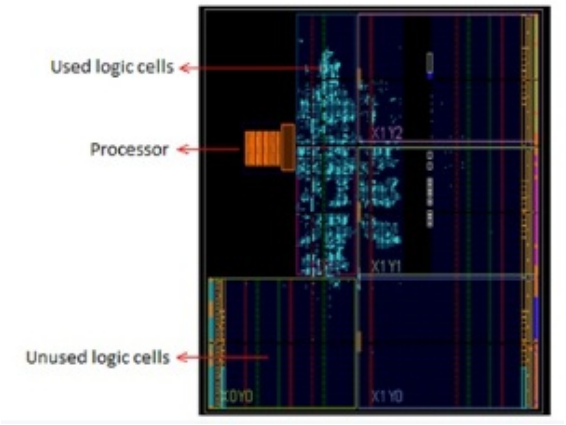

Fig 4: Floor plan of hardware model

Table 1: Utilization report of Hardware resources

Zedboard Zynq evaluation kit / FPGA device part no: xc7z020clg484-1		
Hardware Resources	Utilization measure in quantity	Utilization measure in %
Slice LUTs	4197/53200	7.88%
Slice Registers	5052/106400	4.74%

Power Utilization @ Maximum ambient temperature –: 64.4®C, Junction Temperature –: 45.6®C	
Power Consumption in Watts	
Dynamic Power Consumption	1.626 W
Static power Consumption	0.157 W
Total Power Consumption	1.783 w

4.2 Ethernet Interface

A unique 48 bit IP address and MAC address is configured for the Zedboard. Having configured the IP address and gateway, Dynamic Host Configuration Protocol (DHCP) is initiated. The DHCP timeout is set to be 24 seconds. Now the Zedboard is ready to receive or transmit via Ethernet. An echo server is created in order to check the working status of the configured MAC and IP addresses. The echo server waits for data from the host IP address. At the host end, the transmitted and received data are compared and if found to be the same, confirms the status of IP and MAC configurations.

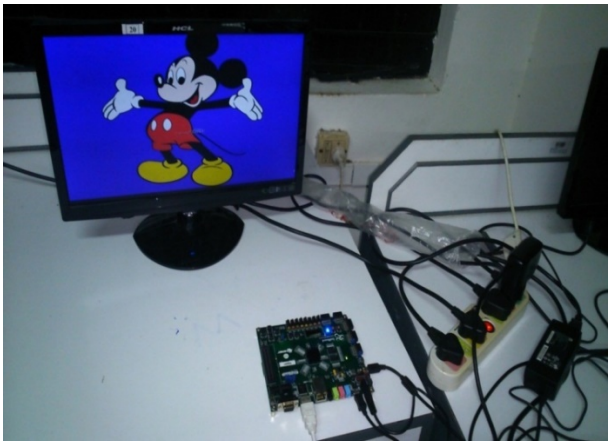

Fig. 5: Experimental Setup for 480 pixels video frames

The figure 5 and figure 6 shown are the experimental setup of SoC designed for IPTV set top box. The setup demonstrates the video frame transmitted at the rate of 100 kbps and 1 Gbps respectively. The data rate achieved by the proposed SoC design set top box is prominently high when compared to the conventional VGA based set top box. Thus, high resolution video frames can be transmitted more effectively using the proposed SoC design with that of still image transmission.

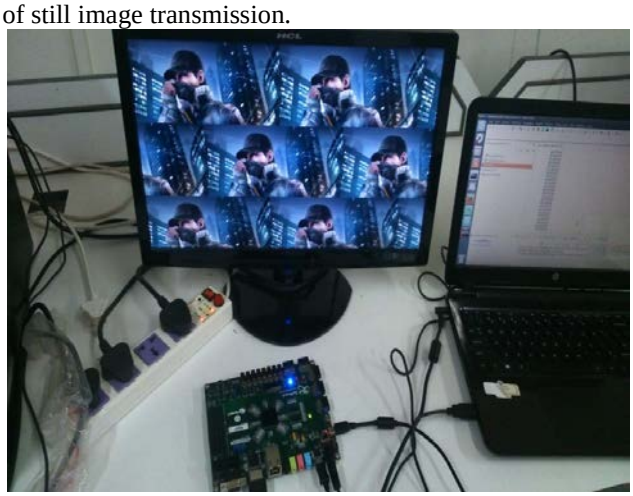

Fig. 6: Experimental Setup for 1080 pixels video frames

4.3 Performance of Dual Core and Single Core

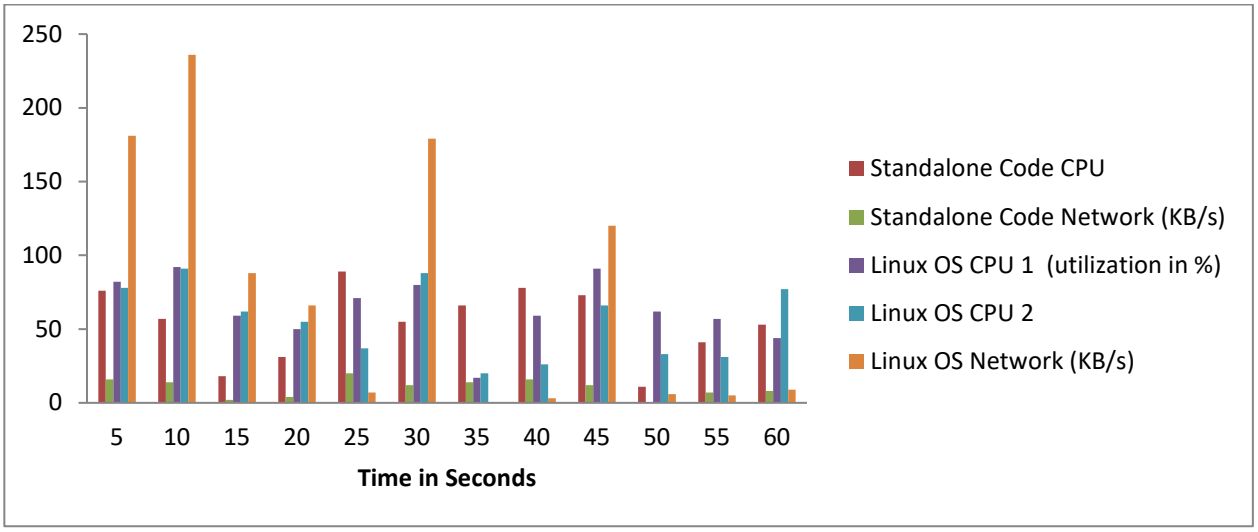

Fig. 7 Performance Evaluation of Core Processors

Table 2: Performance Analysis for Standalone Code and Linux

Time in seconds	Standalone Code		Linux OS		
	CPU (Utilization in %)	Network (KB/s)	CPU 1 (utilization in %)	CPU 2 (utilization in %)	Network (KB/s)
5	76	16	82	78	181
10	57	14	92	91	236
15	18	2	59	62	88
20	31	4	50	55	66
25	89	20	71	37	7
30	55	12	80	88	179
35	66	14	17	20	0
40	78	16	59	26	3
45	73	12	91	66	120
50	11	0	62	33	6
55	41	7	57	31	5
60	53	8	44	77	9

The system performance of single core and dual core processor are plotted in the above figure 5 and figure 6. Compared dual core processor of Linux OS, the standalone code uses only single core processor and hence, the performances of the processor speed and network capability are efficient. The VDMA is implemented in Zedboard Zynq evaluation kit with FPGA device part number xc7z020clg484-1 using Vivado tool. The performance analysis of standalone code and Linux describes that the CPU utilization increases as the number of cores increases. In the dual core processor the work performed by a single core can be shared such that the processing speed increases and the network capability also increases which is shown in table 2.

5. Conclusion

In this paper, video display controller with 1080p resolution and HDMI support is implemented in FPGA. Conventional display controllers are implemented in DSP processors which are more power consuming and low processing speed comparing to FPGA based display controllers. Ethernet controller present in Zynq SOC can support gigabit Ethernet capability. This Ethernet controller is tested using a loopback application present in Xilinx SDK. The Ethernet controller and display controller has been integrated in Zynq SOC to get a continuous streaming of video data at gigabit rate. Further, this work can be extended to process the audio and video data parallel. It could be done by parsing the IP packets to segregate the audio and the video data. Raw video data are used for analysis which occupies more bandwidth. Hence, Video codecs can be implemented in addition to the display controllers in FPGA to process the compressed data.

References

[1] Taeseok Kim and HyonKyung Bahn, "Implementation of Storage Manager for an IPTV Set -Top Box" IEEE Transaction on Consumer Electronics, Vol.54, No.4, pp 1770-1775, November 2008.

[2] Shiguo Lian and Zhongxuan Liu, "Secure Media Content Distribution Based on the Improved Set-top Box in IPTV" IEEE Transactions on Consumer Electronics, Vol. 54, No.2, pp 560-566,May 2008.

[3] Christopher A. Chambers, "Designing a Set-Top Box Operating System", IEEE, pp 368-369,1995.

[4] Ivan Bernabe Sanchez, Daniel Diaz-Sanchez, Mario Munoz Organero, "A Flexible Software Architecture for IPTV Set-top Boxes", IEEE Second International Conference on Consumer Electronics,2012.

[5] ADV7511 Low-Power HDMI 1.4 Compatible Transmitter with Audio Return Channel Programming Guide.

[6] AXI VDMA v6.2 LogiCORE IP Product Guide.

[7] Armando Mora´Campos , Francisco J. Ballester Merelo, Marcos A. Marti´nez Peiro, Jose A. Canals Esteve, "Integer-pixel motion estimation H.264/AVC accelerator architecture with optimal memory management", journal on Microprocessors and Microsystems vol. 32 pp. 68–78.

[8] Minyoung Sung, "A Rich Media Framework for Communication-Broadcasting Converged IPTV", IEEE Transactions on Consumer Electronics, Vol. 56, No.2, May 2010, pp. 463-470.

[9] Abdelrahman Abdelazima, Martin Varleya and Djamel Ait-Boudaoudb, "Effect Of The Hadamard Transform On Motion Estimation Of Different Layers In Video Coding", International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 38, Part 5 Commission V Symposium, Newcastle upon Tyne, UK. 2010.

Ms.P.Kaythry, Assistant Professor in the Department of Electronics and Communication has 12 years of teaching and research experience. She received her B.E (ECE) degree first class with distinction from University of Madras, M.Tech Electronics and Communication degree first class with distinction from Pondicherry University. She has published over 10 research papers in proceedings of International and National Conferences and 2 research papers in acclaimed journals. She is a member of IEEE and Life member of ISTE and IETE. Her areas of interest include Vehicular Adhoc Networks and Error Control schemes in Wireless Networks.

K. J. Jegadish Kumar has 14 years of teaching and research experience, including 6 years of research experience in the field of Cryptographic techniques for various wireless communication applications. He received his B.E (ECE) degree with first class from Manonmaniam Sundaranar University, M.E. Communication systems first class with distinction from Madurai Kamaraj University. He completed Ph.D in JNTU, Hyderabad. He has published over 30 research publications in refereed international journals and in proceedings of international and National conferences. His current research work includes Low power and area efficient cryptographic techniques for wireless communication applications, RF energy scavenging systems, Antenna designs and Wireless Embedded systems etc.