Library Progress International
Vol.44 No. 3, Jul-Dec 2024: P.13645-13656

Print version ISSN 0970 1052
Online version ISSN 2320 317X

Original Article

Available online at www.bpasjournals.com

Optimized Design of Ethernet to HDMI Accelerator and IP Subsystems for IoT

***Ipseeta Nanda[1], J. Midhunchakkaravarthy[2]**

[1]Department of Computer Science , Lincoln University, Kota Bharu, Malaysia, ipseeta.nanda@gmail.com
[2]Faculty of Computer Science and Multimedia, Lincoln University College, Kota Bharu, Malaysia

How to cite this article: Ipseeta Nanda, J. Midhunchakkaravarthy (2024). Optimized Design of Ethernet to HDMI Accelerator and IP Subsystems for IoT. *Library Progress International,* 44(3), 13645-13656.

Abstract:
The rapid advancement in digital electronics has given rise to System on Chip (SoC) technology, enabling the integration of multiple reusable Intellectual Property (IP) components, processors, memory elements, and bus architectures into a single chip. SoCs are increasingly applied in various sectors due to their ability to integrate multiple functionalities, including Internet of Things (IoT) capabilities, onto a single platform. As the demand for complex, multi-functional devices grows, design complexity, power management, and space constraints become crucial. A hierarchical design approach, emphasizing the reuse of pre-designed and verified IP blocks, reduces development costs and time. Tools like the Xilinx Vivado IP Integrator facilitate this by allowing designers to seamlessly integrate IPs. This paper explores an SoC architecture designed for high-definition multimedia interface (HDMI) and Ethernet signal processing, which utilizes an FPGA system to store and transmit video data over long distances through an Ethernet network. This design supports IoT device connectivity and can display video on multiple screens using HDMI while ensuring signal integrity and high-quality output.

Keywords: System on Chip (SoC), Intellectual Property (IP), Field-Programmable Gate Array (FPGA), Internet of Things (IoT), HDMI

Introduction

The increasing complexity of digital electronics design, combined with shorter development cycles and the demand for cost-efficiency, has transformed the industry and given rise to the era of System on Chip (SoC) technology. SoCs are composed of reusable Intellectual Property (IP) blocks, processors or controllers, memory components like random-access memory (RAM), and a bus architecture designed to connect these elements within the SoC. They may also incorporate block-based processors such as Field-Programmable Gate Arrays (FPGAs), integrated signal blocks, and clock circuits. SoCs offer scalability by allowing more IP components to be added, and multiple SoCs can be integrated to create a more comprehensive system. As more devices are designed with internet connectivity in mind, the need for complex, multi-functional, and application-specific chips has become a driving force in the electronics industry. The increasing demand for smarter, more complex devices with specific design requirements is leading to a rapid increase in gate count on smaller chips. Technological advancements have enabled the integration of heterogeneous technologies, pushing the boundaries of chip design. To meet the challenges of growing design complexity, power efficiency, and space constraints, a hierarchical design approach that focuses on the reuse of pre-designed, pre-verified, and optimized components has emerged as a practical solution. This methodology reduces development time and cost while addressing the complexity of modern SoC designs. In this evolving landscape, SoCs are seen as a predictable and reliable solution, where essential components of a fully functional product can be combined into a single chip. These components are now being developed as IPs, making it easier to integrate them into SoC designs. Adding Internet of Things (IoT) capabilities to existing electronic components introduces additional design complexities, particularly in managing power consumption and minimizing space. SoCs must meet the high performance demands of today's consumers, while also addressing market pressures for competitive pricing and efficiency. By reducing system complexity through the reuse of IP systems, designers can create simple components that meet customer needs. There are various types of IP systems, each offering specific advantages depending on the design requirements. Today, electronic devices are ubiquitous, and consumer demand for them continues to grow exponentially. This demand has driven the need for advancements in connectivity technologies. For instance, connection cables that carry audio and video data, such as those used in televisions and DVD players, must now support higher data transmission rates to ensure flawless communication between devices. Analog and digital signals are transmitted through these cables, and IP systems can be used to design complex systems in FPGAs, which are capable of handling both types of signals. The reuse of validated IP designs further enables the creation of new, larger, and more complex systems. Xilinx, a leader in FPGA technology, offers solutions to address the challenges faced by designers. Their Vivado design suite includes a powerful feature known as the Vivado IP Integrator. This tool allows developers to create complex system designs by integrating IPs from the Vivado IP catalog. Through its interactive IP Canvas graphical user interface (GUI) or Tcl programming interface, designers can create detailed system diagrams. An example of such a system is the HDMI FPGA SoC Ethernet System Architecture (FSEHSA), which facilitates the transmission of both audio and video data. HDMI is widely used for personal computers and home entertainment systems, and connectors can extend the distance between HDMI sources and sinks. The FSEHSA system is designed to handle both HDMI and Ethernet signals. In this architecture, the FPGA is responsible for receiving Ethernet frames and storing them in memory. If necessary, the FPGA retrieves the data from memory, prepares it, and transmits it via HDMI, allowing video data to be sent over long distances. This system can be implemented on an Ethernet network, enabling the transmission of video to multiple displays simultaneously. The objective of this system is to design an FPGA capable of processing high-resolution video and Ethernet data, storing it in memory, and quickly handling HDMI-compatible signals. Additionally, this design allows communication with other network devices, enabling data exchange with IoT-enabled devices, which can then be displayed on connected devices. The HDMI to Ethernet system, built on the Zynq7000 processor, offers a solution for combining audio, video, and data streams into a single HDMI cable. This design ensures superior signal quality, offering a simple yet powerful connection for home entertainment networks. The system also supports the sharing of data over Ethernet with various electronic devices, such as gaming consoles and Blu-ray players. The HDMI format delivers high-quality recording, long-term storage, and flexible playback options, making it a comprehensive solution for modern multimedia and entertainment systems.

FPGA Types Based on Applications

An FPGA is made up of configurable rationale pieces (CLBs) that incorporate parts such as lookup tables, capacity components (flip-flops or registers), and more. An FPGA is not fair a collection of discrete Boolean entryways. These CLBs have the capacity to store information and carry out math and Boolean operations. In an FPGA, communication is overseen by outside circuits and is subordinate on a expansive number of input/output (I/O) squares and intercontinental. FPGAs are partitioned into "low-end" and "high-end" assortments agreeing to their employments and level of modernity. Low-end FPGAs are culminate for less difficult plans since of their moo control utilization and least rationale needs. Illustrations are the Straightforward arrangement from Xilinx, the Tornado arrangement from Altera, and the Zynq-7000 SoC family (e.g., ZYBO, MicroZed). These contraptions give a cost-performance trade off. Mid-range illustrations are the Artix-7/Kintex-7 arrangement from Xilinx and the Arria arrangement from Altera. High-end FPGAs, such the Virtex family from Xilinx and the Stratix family from Altera, are planned for expanded rationale thickness and tall execution. Mental Property (IP) rationale pieces are fundamental to VLSI plan. Mental property (IP) is reusable rationale or format plans authorized to a few providers so they can be included in different chip plans. Pre-designed IP components, such as built-up conventions for interfacing like Serial Harbour Interface (SPI), USB, Ethernet, Widespread Nonconcurrent Collector Transmitter (UART), and Progressed RISC Machines (ARM) transport conventions, are as often as possible utilized by SoCs (Framework on Chips). Instead of having to begin from scratch, these parts can be built as particular IP squares that can be rented and utilized once more in diverse plans. Equipment portrayal dialects (HDLs) like VHDL, Verilog, and Framework Verilog are cases of plan approaches that have been created to disentangle the plan preparation due to the expanding complexity of cutting-edge SoCs and the reducing estimate of chips. Reusing IP squares is still a common hone despite the progressions in plan apparatuses since it speeds up item improvement, brings down plan blunders, and abbreviates time to advertise. When reusing IP pieces, compatibility, consistency, and arrangement are critical things to consider.

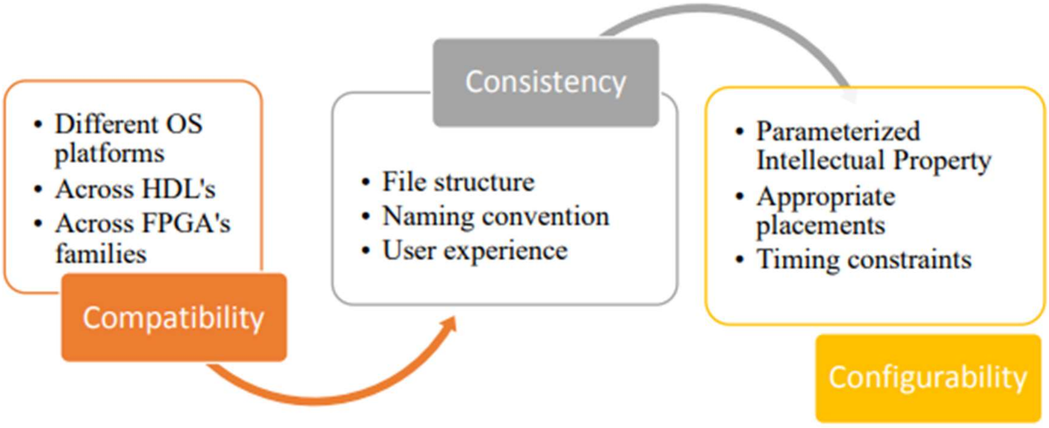

Figure 1: Requirements of an IP User: The Three C's of IP designing

Types of IP Reuse

IP reuse refers to the direct integration of existing design elements into a new system. As shown in Figure 2, IP has many different uses. During configuration, a third-party IP the customer provides can be used to reduce design time. A new design can be created from an existing design by rebuilding it with minor changes and other features.

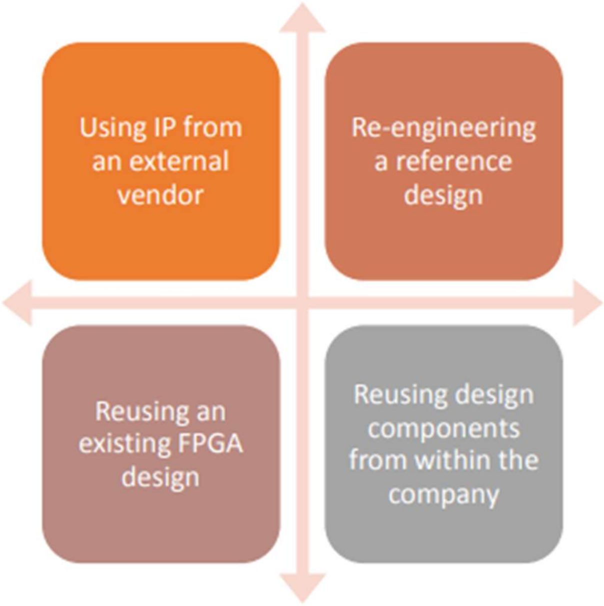

Figure 2: Categories of IP Reuse

Also, some IPs are available as IP lists in the library and can be used immediately in the designs for customization. In some cases, other users' design components can be reused if the required virtual component has been validated and tested. They are also called similar in use because they follow the same standard. Everyone should have a concise and comprehensive document that can be easily reused, thus saving a lot of time.

Proposed Design for IP System and Sub-system

In this diagram, the FPGASoC architecture is built using the Xilinx Vivado IP system and the subsystem using integration IP. The FPGA SoC is designed based on performance and the number of resources available in the Zynq7000 at different levels. The interface between PS and PL is designed for fast communication between controllers and AXI control. High-level synthesis converts software into functional hardware expressed in hardware description language (HDL). A built-in hardware accelerator based on space, power and other constraints can improve performance.

Block diagram of the proposed system with input and output

The primary objective of this research is to design a device that converts Ethernet frames into HDMI signals and outputs them from the device. The system receives data through Ethernet boards connected via Ethernet cables and outputs the HDMI signal through an HDMI cable. The system's architecture, shown in Figure 3, is divided into three main components: Ethernet, HDMI, and memory. The Ethernet module manages external communication through the Ethernet controller and interfaces with memory. The HDMI module handles external communication via the HDMI port and interacts with memory. The memory module, which includes a built-in DDR2 SRAM chip, facilitates communication between Ethernet and HDMI components. The system's core function is to transmit video data, and it is designed to handle both video and audio transmission through the HDMI interface, enabling the transmission of high-quality video and audio signals across devices.

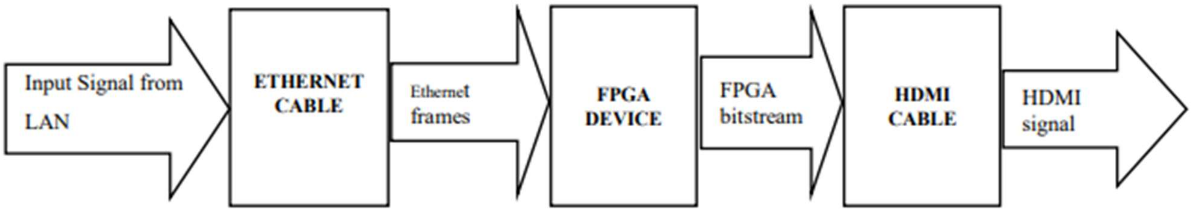

Figure 3: Block Diagram of the Device with Input and Output

Ethernet to HDMI architecture

High-Definition Multimedia Interface (HDMI) technology is the global standard for connecting high-quality devices, capable of transmitting both audio and video. Widely used in digital products, HDMI cables feature 19 pins, with most pins being twisted and shielded by a third pin. High-speed signals like red, green, blue, and orange can transfer data at gigabits per second through these cables. HDMI cables often include Ethernet channels, as introduced in the HDMI 1.4 standard in 2009, to meet the rising demand for Ethernet, particularly with the emergence of 4K resolution. HDMI simplifies setups by replacing up to eleven traditional cables, providing multi-definition audio, surround sound, and digital video. The architecture for Ethernet to HDMI conversion connects an Ethernet controller to an Ethernet core in the FPGA, which then processes data through a memory core (DDR2 RAM) before outputting through the HDMI port. HDMI is commonly used for home and office applications, utilizing TMDS to reduce electromagnetic interference.

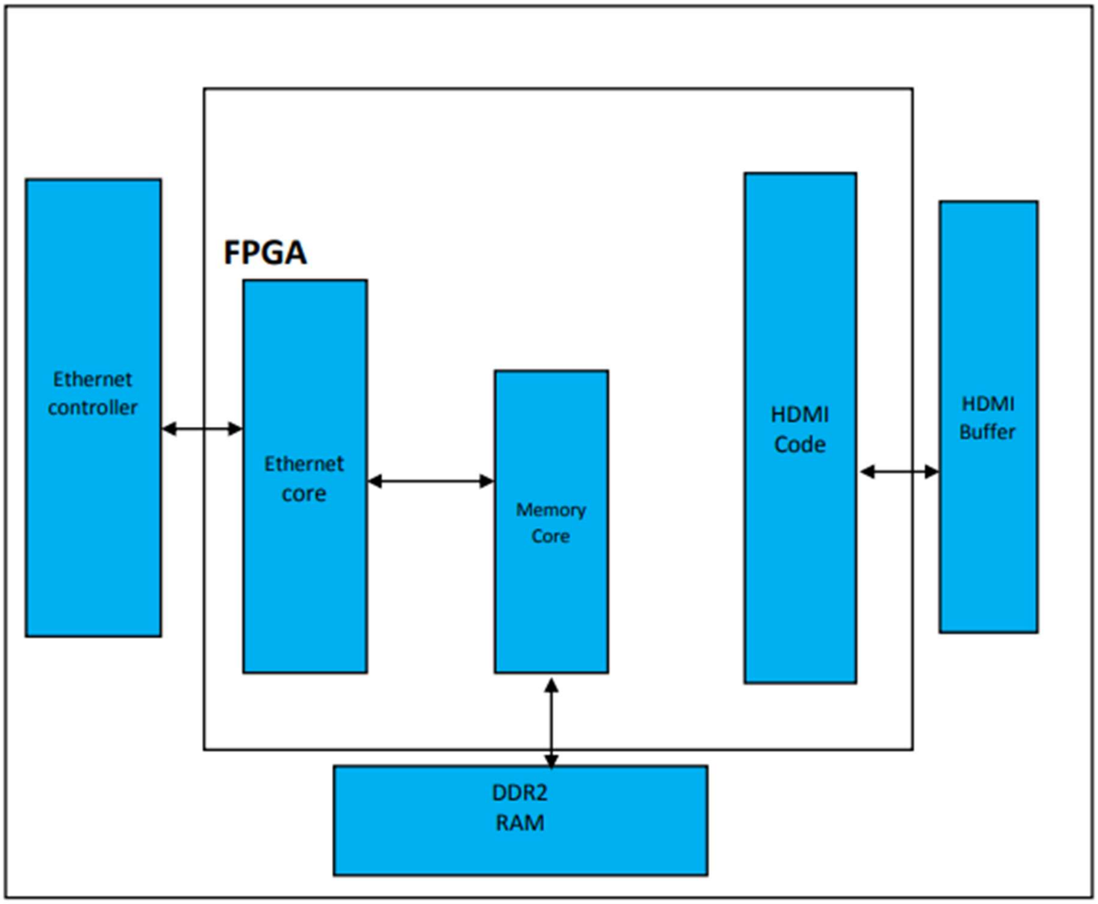

Figure 4: Architecture of Ethernet to HDMI in FPGA

HDMI cable

When transferring data, it is better to reduce the electrical interference in copper cables. To do this, the number of voltage level transitions must be reduced, i.e. the transitions from 0 to 1 and 1 to 0 must be reduced. The block diagram shown in Figure 4 is used. An HDMI source outputs a signal to the left. The video and audio data is sent to the HDMI converter, which processes the signals and converts them to the TMDS. Some HDMI transmitters are also fed with power signals. HDMI transfer signals are encoded and sent over an HDMI cable. The three channels and the clock channel have three wires. One is for the shield and the other two are for data. Both phones exchange data with each other. The same goes for the clock signal. Two wires are used for Digital Data Communication (DDC), which uses an integrated circuit (I2C) bus to communicate with devices. An electronic control cable (CEC) is used for high-level control functions. There are four points. One wire used to determine if the display is connected or not is the Hot Plug Detect (HPD) signal. The remaining three wires are used for +5V, ground and one is reserved. An HDMI receiver decodes the signals from the HDMI cable and separates them into video and audio data and control signals as shown in Figure 5.

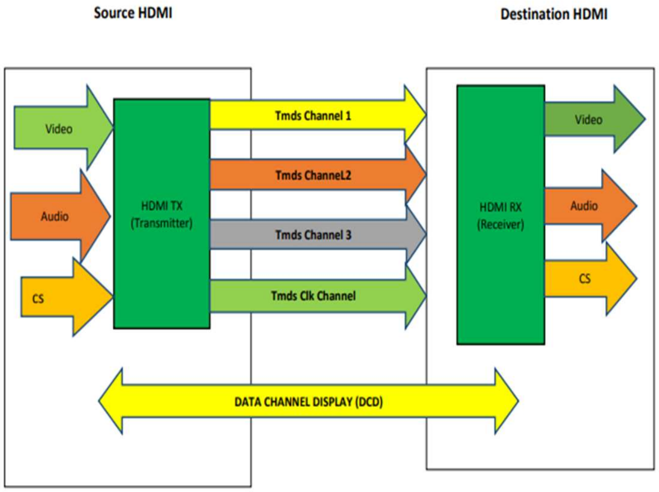

Figure 5: Connection for High Definition Multimedia Interfacing

AXI Interconnect Subsystem

Advanced Extensible Interface (AXI) is an industry-standard bus interface designed for FPGAs, which includes ARM's Advanced Microcontroller Bus Architecture (AMBA) as a protocol for communication between IP blocks. There are three types of AXI protocol: AXI4: ability to perform sequential memory map transfers up to 256 data transfer cycles per address segment, AXI4-Lite: used for single-bit memory map transfers, AXI-Stream : It has no address channel and allows infinite back-and-forth transmission between master and slave. Program Engineering (PL) and Processing System (PS) Communication. The AXI interface is a built-in language between Zynq's PS and PL. FPGA vendors such as Xilinx have made it possible to integrate software and hardware systems on a single chip, and AXI is the main interface between these systems. As an SoC designer, it is important to understand the AXI interface. AXI_GP: These ports are used for general purpose between PL and PS. These are the main ports for PS to PL access and vice versa. High Performance (HP): These memory map connection ports provide a high-speed data path from the cores in the PL to the D-RAM and on-chip memory (OCM) in the PS. Accelerated Integration Port (ACP): This port connects the memory systems from PL to PS. Through this port, the PL can access the cache memory in the PS. It improves overall performance and energy consumption. The system diagram is shown in figure 6.

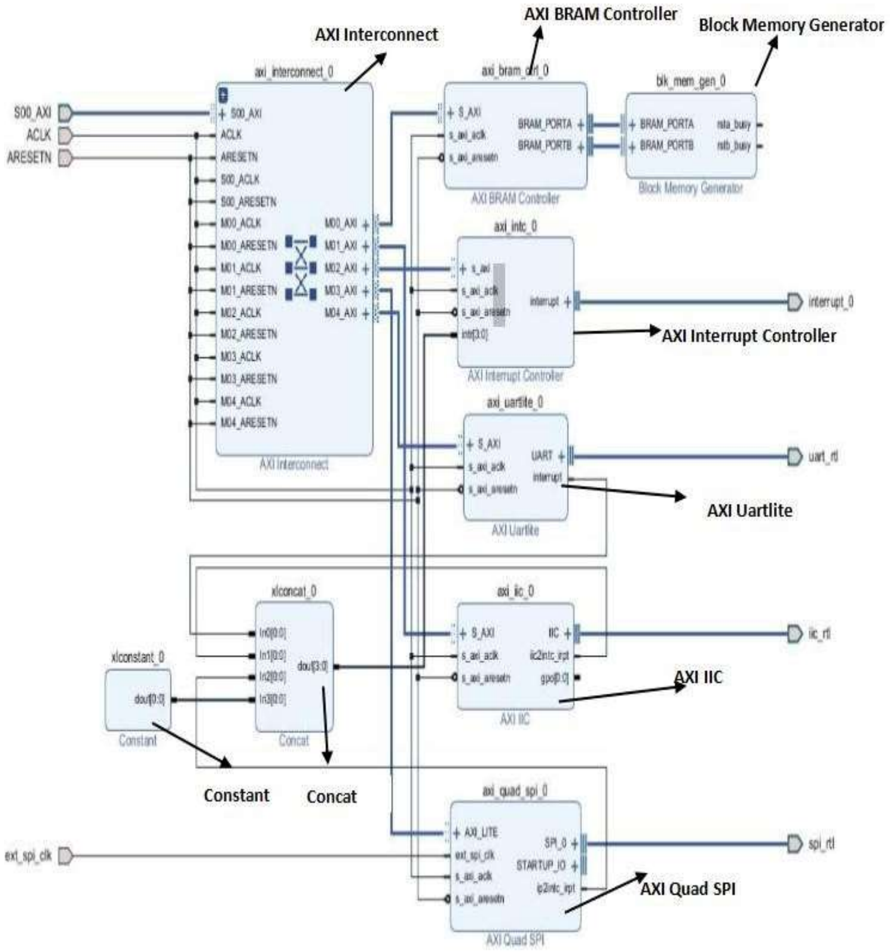

Figure 6: System Diagram for PL with PS

Results and Discussion

The video was created and designed with Xilinx Vivado on a Zynq 7000. The Xilinx Test Pattern Generator IP core is used to generate test patterns to evaluate and debug the proposed video system. . The core provides a variety of test models that allow users to debug and evaluate a proposed video system for color, quality, aspect and motion performance, and/or quality issues. The AXI4-Stream interface signals are converted by the AXI-4 Stream to Video Out IP key to a standard parallel video output interface with timestamps. The output interface is compatible with many external video sinks. Standard video timing signals such as Vsync, Hsync, Vblank, Hblank, DE and Pixel Clock are included. It helps video designers quickly and easily connect video processing blocks to external video sinks such as DVI PHY and AXI4-Stream interface. This core works well with the Xilinx Video Timing Controller (VTC) core and outputs timing signals in video format. Figure 7 shows video production using zynq, created using Xilinx Vivado.

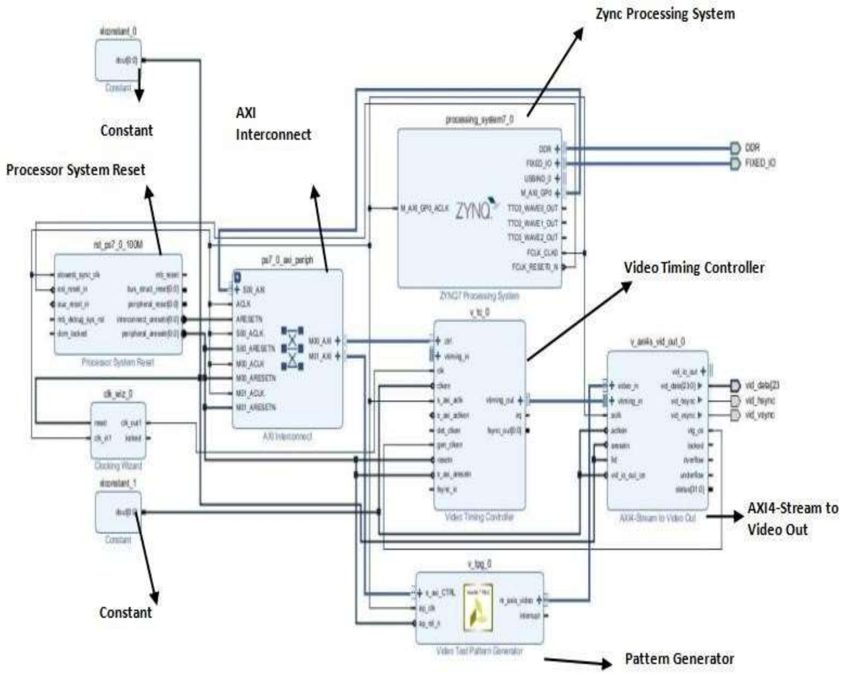

Figure 7: Generating Video using Zynq

The AXI Ethernet subsystem was also designed. The AXI Ethernet subsystem implements a tri-mode (10/100/1000 Mb/s) Ethernet MAC. It supports interfaces such as MII, GMII, SGMII, RGMII and 1000BASE-X to connect the media access control (MAC) to the physical interface chip (PHY). It comes in several variants, including Reduced Gigabit Media Independent Interface (RMII), Serial Gigabit Media Independent Interface (SGMII), and Reduced Gigabit.

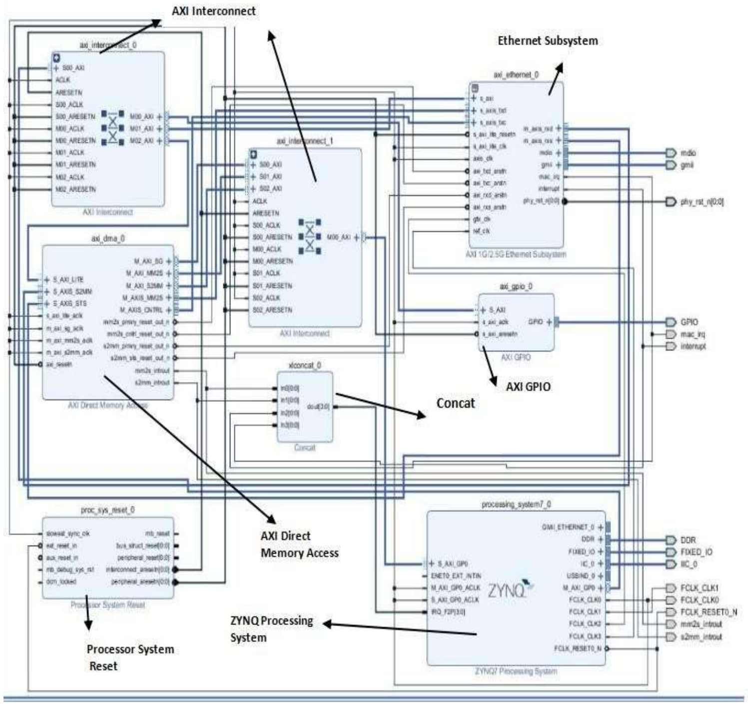

Figure 8: System Diagram of AXI Ethernet Subsystem

Remote Control Interface (RGMII). AXI Direct Memory Access (AXI DMA) IP provides high-speed data transfer between memory and current AXI4 type peripherals. The total chip power analyzed for the AXI Ethernet subsystem is 1.746 W. Figure 11 shows the design of the HDMI subsystem using the Vivado IP integrator. The HDMI subsystem is built using the Xilinx Vivado. RGB/YUV video data is sampled as an AXI4 Stream, pixels are duplicated, timestamped and combined with AUX data in a second AXI4 stream. The output of the RGB or YUV data stream is split into separate RGB/YUV streams for output as HDMI. It has 3 equal data channels for RGB/YUV and separates the data before encoding as TMDS (Reduced Difference Signal). Data that is over-sampled is sent to be serialized before being sent to HDMI output. Figure 9 shows the final design of 4-Gigabit Ethernet implemented using Xilinx Vivado. The diagram in Figure 9 shows three soft layers which are Zynq PS, DMA and AXI Ethernet. Communication is done using the Advanced Advanced Interface (AXI). The AXI Ethernet subsystem provides an interface that controls internal registers. Read and write operations are performed with general interface connections. This is done using the AXI Ethernet IP Subsystem and Gigabit to Gigabit reduced media interface. This design provides Ethernet to the integrated controller using zynq-7000 devices.

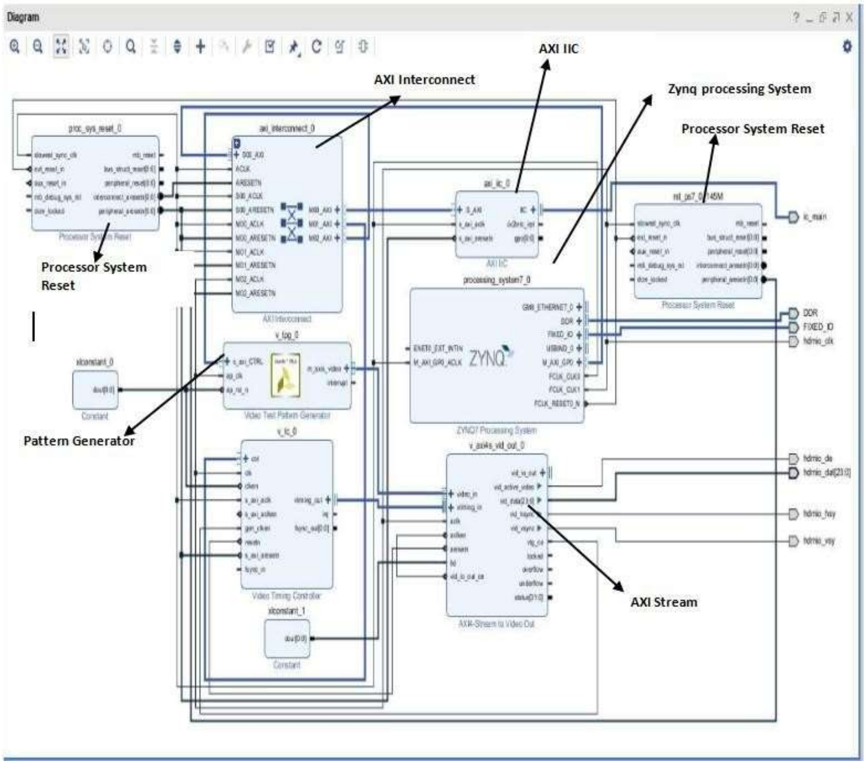

Figure 9: Design of HDMI Subsystem using Vivado IP Integrator

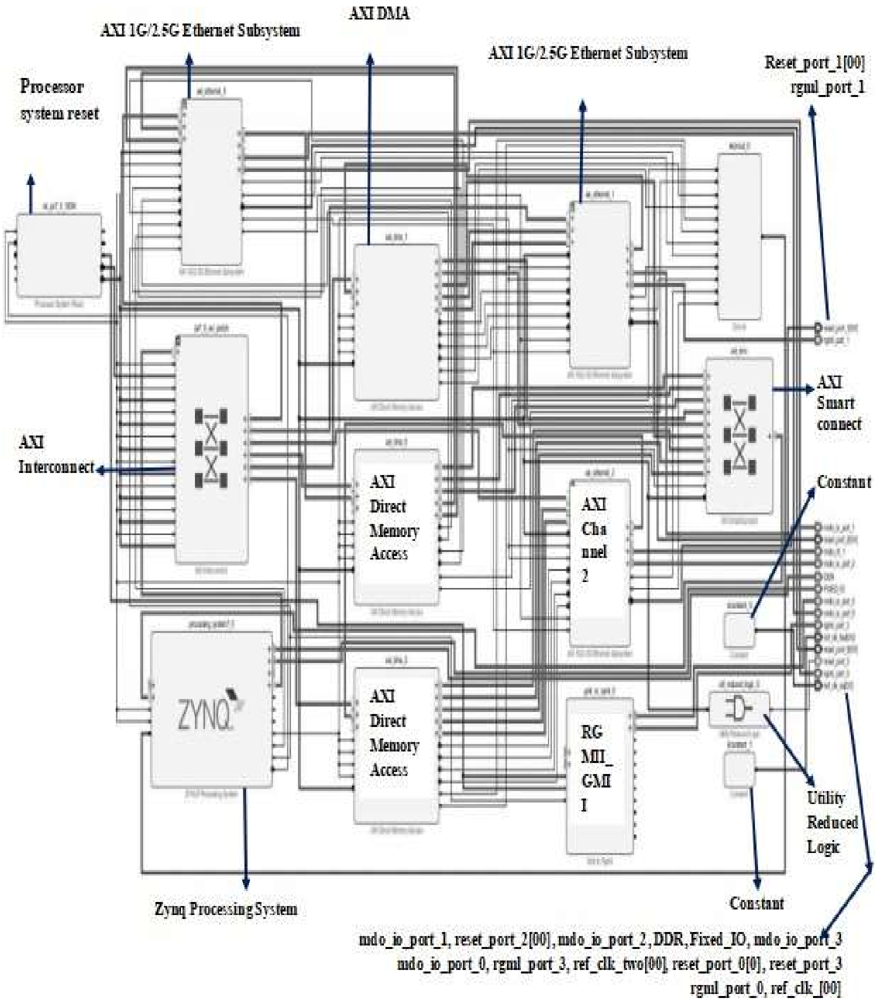

Figure 10: Design of Quad Gigabit Ethernet Implemented with the help of XilinxVivado

Result and Discussion

Local analysis of resource utilization for system design with and without DPR Zedboard using Xilinx Vivado is shown in Table 6.1. Table 6.1 details the materials used for design purposes. LUT available is 53200 and resource usage is 23755, LUTRAM not available 17400 and FF used 3055, flip flop (FF) available 106400 and FF not used is 36605, from BRAM 140 and used 19.50 56 and no. There are 32 BUFGs available and 7 in use. The use of space is shown the graph shown in figure 11 that less resources are used to reduce energy consumption such as bar design.

Table: 6.1 Resources Utilized in the Proposed System Design

Resource	Available	Utilization	Usage%
LUT	53200	23755	44.65
LUTRAM	17400	3053	17.55
FF	106400	36605	34.40
BRAM	140	19.50	13.93
IO	200	56	28.00

BUFG	32	7	21.88

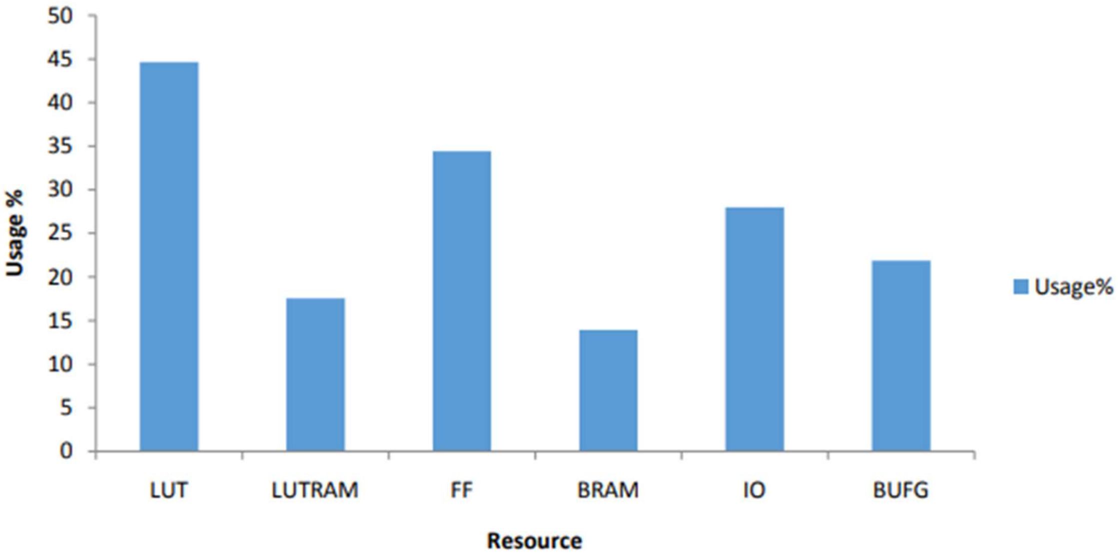

Figure 11 : **Resource Utilization Representation in Designing and Implementation**

Conclusion

The accelerator is implemented using Xilinx Vivado 18.X. HDMI and Ethernet are built in. It uses less resources than what is available in the current system. Less than fifty percent and more than fifty percent of the resources are available for use. The comparison shows that the utilization rate of the available resources is below 50%. Therefore, the design will be less powerful and take up less space. This work discusses the design of the HDMI and Ethernet design using the AXI connector, as well as the construction of the IP system and subsystem. In its broadest sense, the term Internet of Things includes connected "things" (devices) from cell phones to smartphones and wearables that are all connected to the Internet. For efficient communication devices and computers should be connected with low power. For this, the HDMI-Ethernet AXI connection system is designed and implemented using Xilinx Vivado IP embedded and Zynq processor. The HDMI system is considered to have very few resources and less power out there. Without frame loss, the quality of audio and video files will be poor during transmission. The proposed project will help design configurable smart devices. FPGAs can solve the problems of current IOT devices: power consumption and integration.

References:

[1] Ipseeta Nanda and Nibedita Adhikari, “Analysis and Design of Ethernet toHDMI Gateway using Xilinx Vivado”, Advances in Data ScienceandManagement, Springer Nature, pp. 463-477, 2020. ,

[2] Ipseeta Nanda and Nibedita Adhikari, “Application and PerformanceofFPGA using Partial Reconfiguration with Xilinx PlanAhead”, Transportation Electrification Conference (ITEC-India),IEEE, Pune, India, 13-15 Dec. 2017, ISBN: 978-1-5386-2668-9.

[3] Ipseeta Nanda, Dr. J. Midhun Chakkaravarthy Digital Reconfigurable Design COOKBOOK Using XILINX VIVADO Paperback – 5 August 2024

[4] J. P. Delahaye, G. Gogniat, C. Roland, and P. Bomel, "SoftwareRadio and Dynamic Reconfiguration on a DSP/FPGAPlatform", 2004

[5] Richard Anthony, Achim Rettberg, Dejiu Chen, Isabell Jahnich, Gerrit de Boer, and Cecilia Ekelin, "Towards a DynamicallyReconfigurable Automotive Control SystemArchitecture", IFIP, Advances in Information and Communication Technology, VolumeNo. 231, pp. 71–84. Springer

[6] S. Corbetta, F. Ferrandi, M. Morandi, M. Novati, M. D. SantamBrogio, and D. Sciuto, "Two Novel Approaches to OnlinePartialBitstream Relocation in a Dynamically Reconfigurable System",

InProceedings of the IEEE Computer Society Annual SymposiumonVLSI, Washington, DC, USA, IEEE Computer Society, pp. 457–458, 2007.

[7] S. Tripathi, R. Mathur, and J. Arya, "Unified 3GPPand3GPP2Turbo Encoder FPGA Implementation using RuntimePartialReconfiguration", In Wireless Telecommunications Symposium(WTS), 2010, pp. 1 –8, April 2010.

[8] Sheetal U.Bhandari, Shila Subbaraman, Shashank Pujari, RashmiMahajan, “Internal Dynamic Partial Reconfiguration for Real TimeSignal Processing on FPGA”, Indian Journal of ScienceandTechnology, Volume No. 3, Page(s):365-368, 2010.

[9] Tobias Becker, Wayne Luk, and Peter Y. K. Cheung, "EnhancingRelocatability of Partial Bitstreams for Run Time Reconfiguration", In Proceedings of the 15th Annual IEEE SymposiumonFieldProgrammable Custom Computing Machines, WashingtonDC, USA, IEEE Computer Society, pp. 35–44, 2007.

[10] Tyrone TaiOn Kwok and Yu Kwong Kwok, "On the DesignofaSelf Reconfigurable SoPC Based Cryptographic Engine", InProceedings of the 24 th International Conference on DistributedComputing Systems Workshops W7: EC (ICDCSW’04) VolumeNo. 7, ICDCSW ’04, Washington DC, USA, IEEEComputerSociety, pp. 876–881, 2004.

[11] Xilinx Inc. "MicroBlaze Processor Reference Guide", January2008.

[12] Xilinx Inc. "Processor IP Reference Guide", 2005.

[13] Xilinx Inc. "Two Flows for Partial Reconfiguration: ModuleBasedor Small Bit Manipulations XAPP 290 (v1.0)", May, 2002.