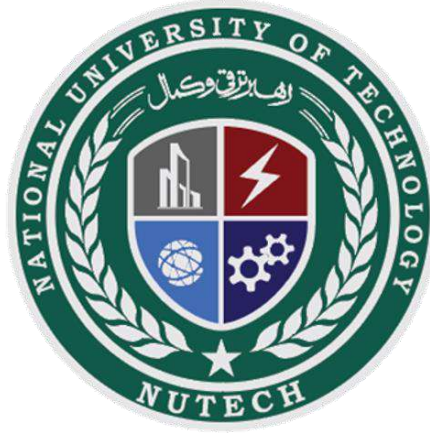


FPGA Based Real-Time Video Processing System



Submitted By
Muhammad Shoaib

Supervised By
ABDUL REHMAN BUZDAR

Department of Computer Engineering
National University of Technology (NUTECH) Islamabad,
Pakistan
2023

FPGA Based Real-Time Video Processing System



By

Muhammad Shoaib

F19604013

**A Project Report Submitted to the Department of
Computer Engineering in partial fulfillment of the
requirements for the degree of Bachelor of Science in
Computer Engineering**

Department of Computer Engineering

National University of Technology (NUTECH) Islamabad,

Pakistan

2023

BS CEN

Muhammad Shoab

2023

CERTIFICATE OF APPROVAL

It is certified that the “FPGA Based Real-Time Video Processing System” project was carried out by Muhammad Shoaib, Reg. No. F19604013, under the supervision of Dr. Abdul Rehman Buzdar, National University of Technology (NUTECH), Islamabad, is fully adequate, in scope and in quality, as a final year project for the degree of BS of Computer Engineering.

Supervisor:

Dr. Abdul Rehman Buzdar

Associate Professor

Dept. of Computer Engineering

National University of Technology (NUTECH), Islamabad

HOD:

Dr. Kamran Javed

Associate Professor

Dept. of Computer Engineering

National University of Technology (NUTECH), Islamabad

ACKNOWLEDGMENT

All the acclamation and appreciation are for Almighty ALLAH who created the universe and bestowed the mankind with knowledge and wisdom to search for its secrets. We feel great pleasure and honor to express our deepest sense of gratitude, sincere feelings and regards to our supervisor Dr. Abdul Rehman Buzdar for their efficient guidance, tremendous help, and special way of advice for the completion of our project. We also like to extend our special thanks to our HoD Dr. Kamran Javed for their encouragement and support.

DEDICATION

We dedicate our project to our parents who supported us during tough situations and miseries to accomplish this project step by step. They tried to get us best suitable knowledge.

ABSTRACT

The real-time video processing system demonstrated in this project makes use of FPGA technology and the Xilinx Vivado software. The system intends to meet the requirement for quick and adaptable video processing across a range of applications. The video processing pipeline collects video data using an OV7670 camera, applies a Gaussian lowpass filter, and then applies a Sobel edge detection filter in hardware. The FPGA development board's buttons and switches allow users to communicate with the system. Users can toggle the video processing on and off as needed to view the raw, unedited footage. They may separately turn on or off the Gaussian and Sobel filters, giving the processing workflow flexibility. Additionally, consumers may modify the Sobel filter thresholding via board buttons, which gives them control over the level of sensitivity of the edge detection.

The FPGA's parallel processing capabilities surpass the limitations of traditional microprocessor based systems, enabling real-time video processing for applications that demand high-speed data handling. This project exemplifies a customizable and efficient video processing solution, adept at meeting specific requirements. The integrated Gaussian and Sobel filters showcase the system's noise reduction and feature extraction abilities, with potential uses in surveillance, medical imaging, and robotics.

This project acts as a crucial stepping stone in advancing video processing technologies. It offers valuable benefits to various industries, including real-time edge detection and optimized video data. The proposed FPGA-based system demonstrates the potential impact of FPGA technology on high-speed video processing. It effectively showcases the advantages of this technology in meeting demanding video processing requirements.

TABLE OF CONTENTS

LIST OF FIGURES	vi
LIST OF TABLES	vii
LIST OF ABBREVIATIONS	viii
ABSTRACT	ix
Chapter 1	1
INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Specifications of Proposed Solution	2
1.4 Purpose of the Project/Research.....	2
The key purposes of the project/research are as follows:	3
1.5 Applications of the Project/ Research	4
1.6 Report Organization.....	4
Chapter 2	5
LITERATURE REVIEW	5
2.1 Related Technologies/Projects/Studies/Research	5
Embedded Real-Time Video Processing System on FPGA	5
2.2 Limitations and Bottlenecks	6
• FPGA Resource Constraints:	6
• Real-Time Processing Speed:	7
• Latency:.....	7
• Power Consumption:.....	7
• Memory Management:.....	7
• Hardware-Software Integration:	7
• Testing and Validation:.....	8
• User Interaction Complexity:.....	8
• Filter Combination:	8
• Scalability:	8
2.3 Summary	8
Chapter 3	11
DESIGN AND IMPLEMENTATION	11
3.1 Design of the Project Hardware/ Software.....	11
3.1.1 Block Diagram.....	11
3.2 Implementation Procedure	12

Camera Interface:.....	12
Camera Configuration:	13
Capture Image Data	13
Protocol Used for Camera Interfacing	13
SSCB (Standard Serial Camera Interface).....	13
RGB To Grey Scale Conversion:.....	13
Kernel Processing:	14
Gaussian Filter Implementation.....	15
Sobel Filter.....	16
3.3 Details About Hardware.....	17
3.3.1 XILINX ZYBO FPGA Board.....	17
3.3.2 Ov7670 Camera Module	17
3.4 Details About Software Tools	17
3.4.1 XILINX VIVADO	17
3.4.2 MATLAB.....	17
3.5 Details of Final Working Prototype	17
3.5.1 Hardware Prototype.....	18
3.5.2 Software Prototype	18
Chapter 4.....	19
RESULTS AND EVALUATION	19
4.1 Project Analysis	19
Chapter 5.....	22
DISCUSSION.....	22
5.1 Discussions on the Findings.....	22
5.1.1 Comparison with Initial Project Specifications	22
5.1.2 Reasoning for short comings	22
The project's sacrifice of video resolution is its only shortcoming. We achieve a pixel resolution of around 640 × 480 at 30 frames per second. Consequently, video processing is a little bit sluggish. We picked this since it is an inexpensive camera (OV7670), costing between \$5 and \$8, and because the data sheet is open source and readily available online. may help in interfacing the camera with the FPGA board.	22
5.2 Limitations	22
5.3 Recommendations	22
Chapter 6.....	23
CONCLUSION.....	23
REFERENCES	24

APPENDICES	25
APPENDIX A	26
TOOLS AND SPECIFICATIONS.....	26
Software Used.....	27
Xilinx, Vivado Design Suite:	27

LIST OF FIGURES

Figure 1.1 Design Time.....	1
Figure 1.2 Block Diagram.....	2
Figure 3.1 Block Diagram.....	10
Figure 3.2 Implementation Procedures	11
Figure 3.3 Camera Interface.....	11
Figure 3.4 RGB to Greyscale.....	13
Figure 3.5 Kernel Processing.....	13
Figure 3.6 Gaussian Filter kernel.....	14
Figure 3.7 Gaussian Filter Implementation.....	14
Figure 3.8 Sobel Filter Kernel.....	15
Figure 3.9 Sobel Filter Implementation.....	15
Figure 3.10 Hardware Prototype.....	16
Figure 3.11 Software Prototype.....	17
Figure 4.1 Video Capture from Camera in RGB.....	14
Figure 4.2 RGB to Grey Scale.....	14
Figure 4.3 Gaussian Low Pass Filter.....	15
Figure 4.4 Human Pose Estimation.....	15
Figure 6.1 FPGA Board	20
Figure 6.2 OV7670 Camera.....	21

LIST OF TABLES

Table 1.1 Literature Review	5
-----------------------------------	---

LIST OF ABBREVIATIONS

FPGA Field Programmable Gate Array.

HDMI: High-Definition Multimedia Interface

HDL: Hardware Description Language

SSCB: Standard Serial Camera Interface

I2C: Inter-Integrated Circuit

HOD: Head of Department

NUTECH: National University of Technology

CEN: Computer Engineering

ABSTRACT

The real-time video processing system demonstrated in this project makes use of FPGA technology and the Xilinx Vivado software. The system intends to meet the requirement for quick and adaptable video processing across a range of applications. The video processing pipeline collects video data using an OV7670 camera, applies a Gaussian lowpass filter, and then applies a Sobel edge detection filter in hardware. The FPGA development board's buttons and switches allow users to communicate with the system. Users can toggle the video processing on and off as needed to view the raw, unedited footage. They may separately turn on or off the Gaussian and Sobel filters, giving the processing workflow flexibility. Additionally, consumers may modify the Sobel filter thresholding via board buttons, which gives them control over the level of sensitivity of the edge detection.

The FPGA's parallel processing capabilities surpass the limitations of traditional microprocessor based systems, enabling real-time video processing for applications that demand high-speed data handling. This project exemplifies a customizable and efficient video processing solution, adept at meeting specific requirements. The integrated Gaussian and Sobel filters showcase the system's noise reduction and feature extraction abilities, with potential uses in surveillance, medical imaging, and robotics.

This project acts as a crucial stepping stone in advancing video processing technologies. It offers valuable benefits to various industries, including real-time edge detection and optimized video data. The proposed FPGA-based system demonstrates the potential impact of FPGA technology on high-speed video processing. It effectively showcases the advantages of this technology in meeting demanding video processing requirements.

Chapter 1

INTRODUCTION

This project shows a hardware-based video processing system that can capture video data from an OV7670 camera and process it in real time while employing a Gaussian lowpass filter and a Sobel edge detection filter. Verilog HDL is utilized for hardware implementation, and the FPGA Board is programmed using the Xilinx Vivado tool.

1.1 Background

In recent years, there has been a significant increase in the demand for real-time video processing. This surge is primarily driven by various applications such as video analytics, surveillance, virtual reality, augmented reality, medical imaging, and more. However, traditional software-based video processing systems that run on general-purpose processors may struggle to meet the high-speed requirements of these demanding applications. Moreover, they often lack optimization for specific video processing tasks, resulting in inefficiencies in computation and resource utilization.

Fortunately, FPGA-based solutions provide a promising alternative. These solutions offer the potential for substantial performance gains and customization to overcome these challenges. By leveraging FPGA technology, organizations can achieve improved efficiency and tailored solutions to meet the demands of real-time video processing applications.

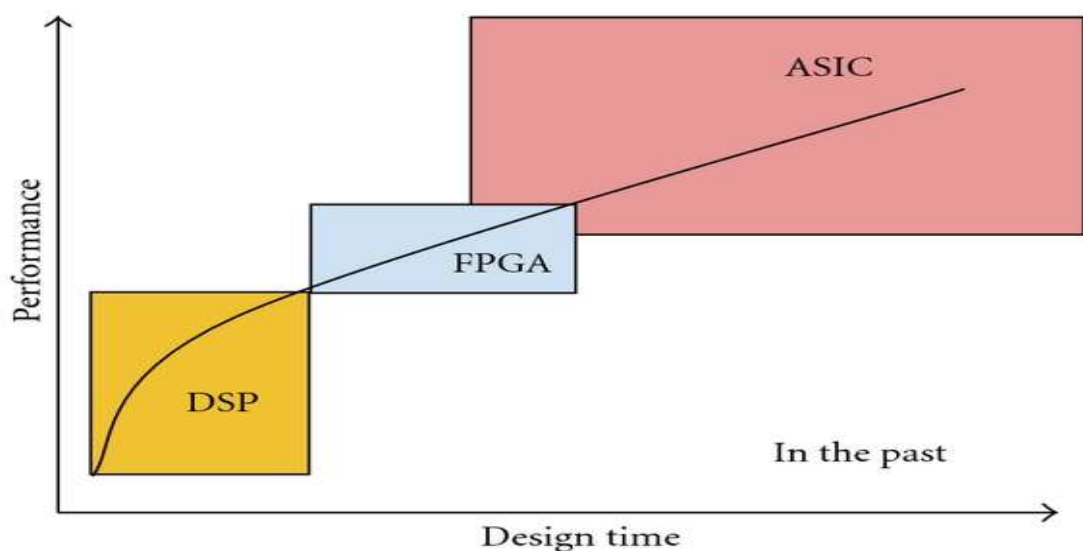


Figure 1.1: Design Time

1.2 Problem Statement

Traditional video processing systems encounter a significant challenge in keeping pace with real-time video data and providing customization options. Their limited speed and inflexibility restrain their ability to optimize performance for specific applications. The project proposes an FPGA-based real-time video processing system. By utilizing the programmability and parallel processing capabilities of FPGAs, the solution aims to overcome these limitations and deliver a high-speed, customizable video processing solution for a variety of applications.

1.3 Specifications of Proposed Solution

The proposed solution aims to achieve real-time video processing by incorporating noise reduction and edge detection capabilities. Additionally, it provides a customizable user interface catering to various video processing scenarios. The effectiveness and efficiency of the implemented system will be assessed through performance metrics, which serve as key indicators.

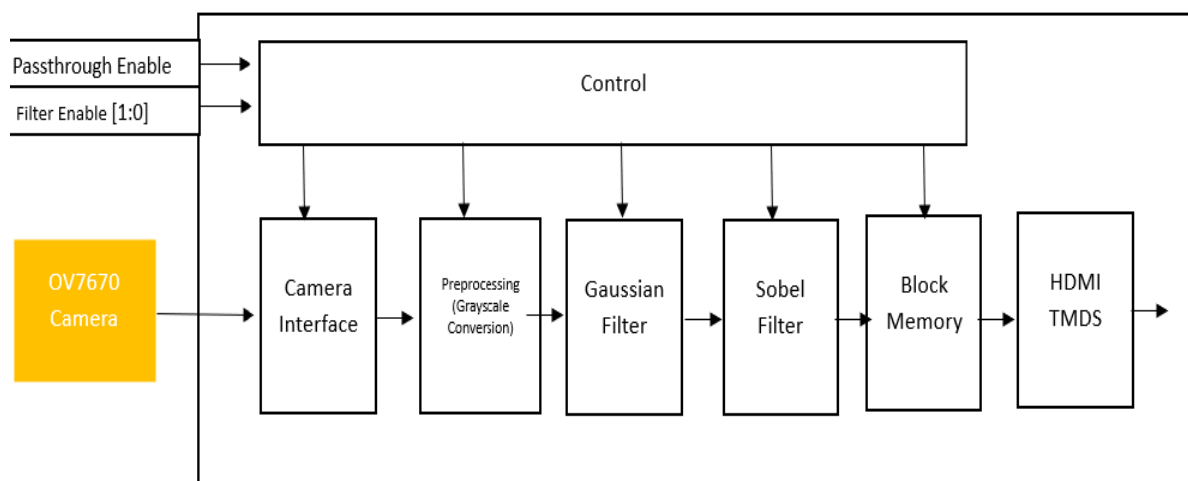


Figure 1.2 Block Diagram of Proposed System

1.4 Purpose of the Project/Research

The project research aims to develop and implement an FPGA-based real-time video processing system. Its primary objective is to address the growing demand for high-speed, customizable video processing in various applications. By harnessing the power of FPGA technology, this project seeks to overcome the limitations commonly found in traditional video

processing systems. These limitations include inefficient handling of real-time video data and limited customization options.

The key purposes of the project/research are as follows:

- **Real-Time Video Processing**

The project's objective is to design and implement a real-time video processing pipeline. This pipeline will play a vital role in applications such as surveillance, robotics, and autonomous systems where prompt data analysis and response are crucial.

- **FPGA-based parallel processing**

FPGA technology showcases remarkable parallel processing capabilities, enabling the seamless handling of extensive amounts of video data simultaneously. This extraordinary potential can be harnessed to accomplish swift video processing without compromising on latency levels.

- **Noise reduction and image enhancement**

The project aims to enhance video quality by incorporating a Gaussian lowpass filter into the video processing pipeline. This integration effectively reduces high-frequency noise and artifacts present in captured video frames.

- **Edge detection and feature extraction**

The video processing system incorporates a Sobel edge detection filter in a pipeline. This powerful feature allows the system to accurately identify edges and contours within the video frames. It plays a crucial role in various applications such as object detection, motion tracking, and feature extraction.

- **Customizability and user interaction**

The project aims to empower users with the capability to interact with the video processing system using buttons and switches on the FPGA board. Through filters users have the flexibility to enable or disable video processing, select specific filters, and adjust parameters like thresholding for edge detection. Such functionalities cater to diverse application scenarios, enhancing versatility in usage.

- **Evaluation of performance metrics**

The project aims to assess the system's performance based on various metrics that include throughput, resource utilization, video quality, and latency. Evaluating these measurements helps gain insights into the efficiency and effectiveness of the implemented video processing solution.

- **Exploration of novelty**

The project aims to explore the possibilities of utilizing FPGA technology for real-time video processing by implementing customizable filters. This research integrates the OV7670 camera module and designs filters using Verilog HDL to showcase the advantages of FPGA-based video processing compared to traditional methods.

- **Impact and Potential Application**

The FPGA-based video processing system has the potential to make a significant impact across various industries. For instance, in domains like surveillance, medical imaging, robotics, and computer vision. This groundbreaking technology allows for real-time video data handling and extraction of valuable insights. As a result, it paves the way for new possibilities in these fields.

1.5 Applications of the Project/ Research

The project can be utilized in different sectors:

- 1) Real-Time Video Analytics
- 2) Surveillance and Security Systems
- 3) Medical Imaging and Diagnosis
- 4) Object Detection and Tracking
- 5) Human-Computer Interaction (HCI)
- 6) Video Enhancement and Restoration.

1.6 Report Organization

Chapter 1: Introduction

Chapter 2: Literature Review

Chapter 3: Design and Implementation

Chapter 4: Tools and Techniques

Chapter 5: Results and Evaluation

Chapter 6: Conclusion

Chapter 2

LITERATURE REVIEW

We conducted a thorough Literature Review before beginning this project to ensure that we are aware of all the related work that has already been done in this field and the topics that are still under research. During our Literature Review, we focused on studying the techniques that are commonly used to overcome the problem of the health, security, and safety system. We also compared the result of existing system and come to conclusion that we should combine these models to get better results.

2.1 Related Technologies/Projects/Studies/Research

Various technologies/Projects/Studies/Research that are related to our thesis/project work are highlighted below in table:

Table 1.1 Literature Review

Sr.	Title	Author
1.	Embedded Real-Time Video Processing System on FPGA 2012	Yahia Said, Taoufik Saidani, Mohamed Atri
2.	Research Real-Time Video Processing with FPGAs 2021	Chris Brown
3.	FPGA implementation of real time video signal processing using Sobel 2017	Emrah Onat

4.	FPGA based Embedded System Design for Video Processing Application. 2013	Sanjay Singh, Ravi Saini, Anil K. Saini, A. S. Mandal
5.	Real Time Image Processing on FPGAs 2018	Shaonan Zhang
6.	A FPGA based efficient camera interfacing architecture for real time video processing. 2021	Sayantam Sarkar, Satish S. Bhairannawar, Raja K.B.
7.	Implementing Gaussian Filter by Using VHDL to Blur Images 2021	Muhammed Kocaoğlu
8.	FPGA implementation of filtered image using 2D Gaussian filter 2016	Leila kabbai, Anissa Sghaier, Ali Douik and Mohsen Machhout

2.2 Limitations and Bottlenecks

Their proposed systems and models are limited to:

- **FPGA Resource Constraints:**

The complexity and size of the video processing pipeline implemented on the FPGA may be limited by its hardware resources. When utilizing large filter kernels or multiple filters,

significant FPGA resources could be consumed, leading to potential constraints on the scope of the project.

- **Real-Time Processing Speed:**

Achieving real-time video processing on a sufficient frame rate can be challenging, especially when applying computationally intensive filters. The processing speed may be constrained by the FPGA's clock frequency and the complexity of the implemented algorithms.

- **Latency:**

The processing time for video frames can cause delays in the system, which might adversely affect time-sensitive applications and impact the system's responsiveness.

- **Image Quality and Accuracy:**

The effectiveness of Gaussian lowpass and Sobel edge detection filters can vary depending on several factors. These factors include filter parameters and the quality of the captured video. It is important to strike a balance between them for optimal results and achieving high-quality video processing results.

- **Power Consumption:**

FPGA-based video processing systems can significantly consume power, particularly when dealing with high-resolution video. It is essential to consider power consumption, especially for applications that require portability or battery operation.

- **Memory Management:**

Efficient buffering and memory management are crucial for ensuring smooth real-time video processing. Factors like the size and type of memory available on the FPGA board can greatly impact the system's performance in handling video data efficiently.

- **Hardware-Software Integration:**

Integrating the OV7670 camera module with the FPGA board and configuring their interface can bring about challenges regarding hardware-software integration and compatibility.

- **Testing and Validation:**

Thoroughly testing and validating the entire video processing system is crucial to ensure its proper functionality and performance. Testing real-time video processing systems can often become intricate and time-consuming. This comprehensive evaluation process plays a vital role in guaranteeing seamless operations and optimal outcomes within the system.

- **User Interaction Complexity:**

Implementing user interaction on the FPGA board can be challenging due to the additional logic requirements and the need for effective user interface design. This involves incorporating buttons and switches to facilitate user engagement. It may also impact responsiveness, requiring thoughtful considerations.

- **Filter Combination:**

Combining the Gaussian lowpass and Sobel edge detection filters in video processing has the potential to introduce interactions and artifacts. To achieve accurate results, it is crucial to ensure proper integration and coordination between these filters.

- **Scalability:**

The project's design might encounter limitations in terms of scalability for handling higher resolution video and accommodating additional processing modules.

- **FPGA Development Learning Curve:**

When working with FPGA technology and Verilog HDL, individuals new to FPGA-based design may experience a learning curve. This learning process can potentially impact the time required for project development.

2.3 Summary

An FPGA-based real-time video processing system is developed to address critical challenges in health, security, and safety applications. By harnessing the capabilities of FPGA technology, this system aims to overcome the limitations of traditional video processing systems. It offers high-speed data processing and enables customization for specific use cases. To lay a strong foundation, the research commences with an extensive Literature Review to explore existing techniques and ongoing research in the field.

Real-time video processing plays a crucial role in various applications, especially in the realms of surveillance, robotics, and autonomous systems. In order to achieve this objective, the project aims to design an efficient video processing pipeline. This pipeline will effectively capture video data from an OV7670 camera module and implement advanced filtering techniques.

One crucial aspect of the system involves utilizing FPGA-based parallel processing. This technology enables the simultaneous handling of large volumes of video data. FPGA's parallel architecture offers significant processing power and impressive low latency capabilities. This makes it an excellent choice for real-time video analytics.

The system features two essential filters: the Gaussian lowpass filter and the Sobel edge detection filter. The Gaussian filter effectively reduces high frequency noise and artifacts found in video frames, ultimately enhancing overall video quality. Conversely, the Sobel filter emphasizes edges and contours, allowing for various applications such as object detection and motion tracking.

Customizability, a key feature of the video processing system, empowers users to effortlessly interact with the FPGA development board using buttons and switches. This allows for seamless toggling of video processing, enabling or disabling specific filters, and fine-tuning parameters such as Sobel filter thresholding. Such flexibility bestows adaptability to various scenarios. This evaluation helps to understand the efficiency and effectiveness of the FPGA-based video processing solution and highlights areas for further optimization.

The project delves into the exploration of utilizing FPGA technology to process real-time videos through customizable filters. By integrating the OV7670 camera module and implementing Verilog HDL filters, it showcases the superiority of FPGA-based solutions over conventional approaches.

The project has a diverse impact and potential applications. In the realm of surveillance and security systems, real-time video analytics can greatly enhance their effectiveness by providing timely data analysis for threat detection and response. Additionally, in the fields of robotics and autonomous vehicles, this advanced system's high-speed processing capabilities enable it to make real-time decisions, improving navigation and obstacle avoidance.

Additionally, the system's noise reduction capabilities offer significant benefits to medical imaging and diagnosis. This leads to enhanced image quality and more accurate diagnoses. Moreover, industries can leverage the power of edge detection in applications such as industrial inspection and quality control. It enables efficient detection of defects, thereby ensuring optimal product quality. Reducing bandwidth requirements while maintaining essential information. Augmented reality and virtual reality experiences can be enriched through real-time video enhancement, offering immersive and lifelike visuals.

Chapter 3

DESIGN AND IMPLEMENTATION

3.1 Design of the Project Hardware/ Software

3.1.1 Block Diagram

Figure 3.1 shows the main block of the system and its implementation. Following are its components.

- Camera Interface
- Processing (RGB to Greyscale Conversion)
- Control Unit
- Gaussian Filter Implementation
- Sobel Filter Implementation
- Block Memory
- HDMI (Display)

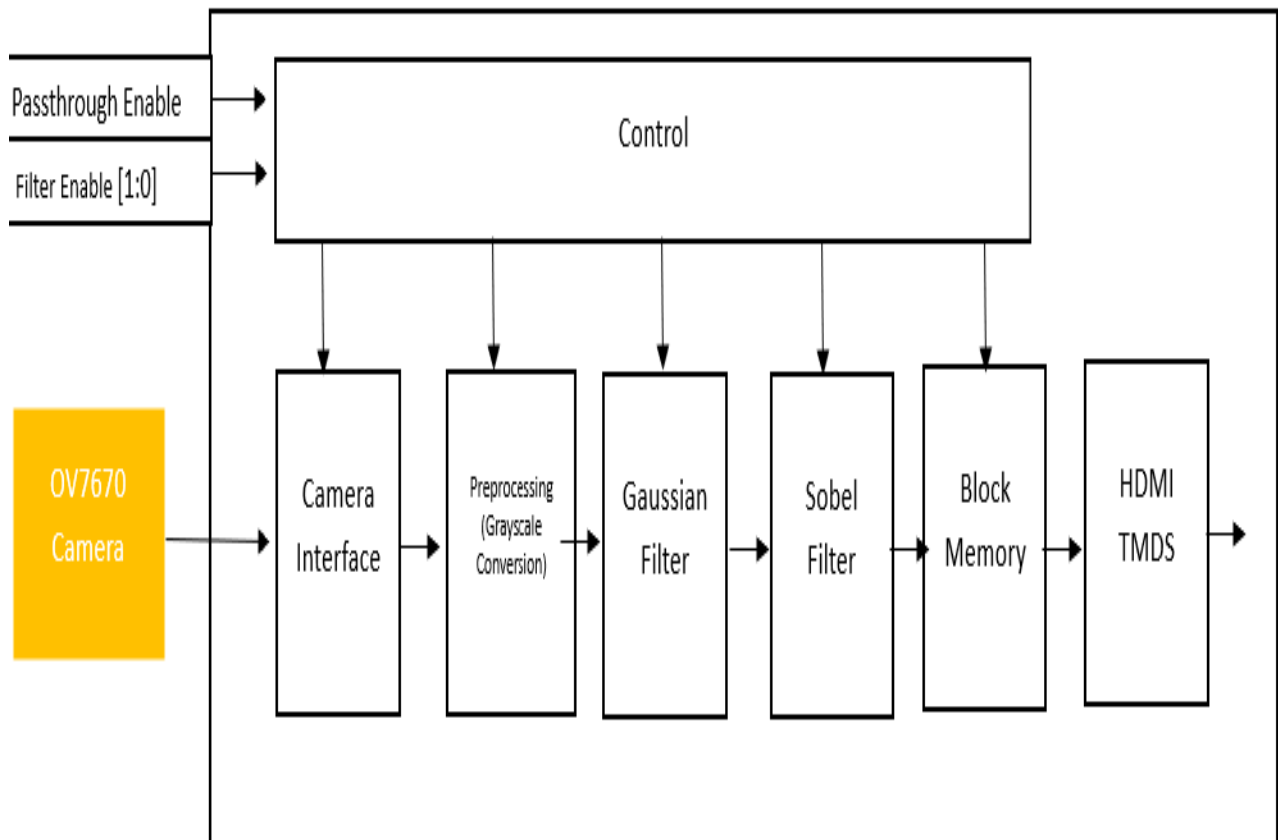


Figure 3.1 Block Diagram

3.2 Implementation Procedure

The detailed implementation procedure of our proposed model is:

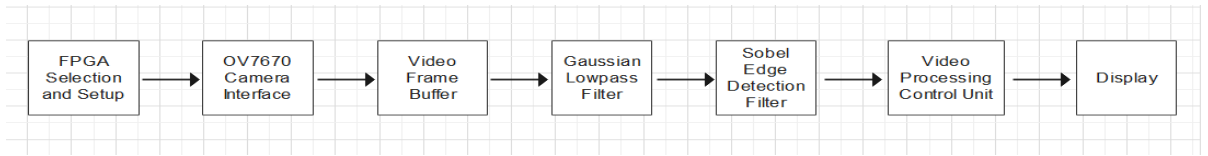


Figure 3.2 Implementation Procedures

1. Implements a video processing system using Xilinx Vivado and Verilog HDL
2. Video data is efficiently captured from an OV7670 camera.
3. The system applies a Gaussian lowpass filter to reduce noise in the video frames.
4. A Sobel edge detection filter is employed to extract features and highlight edges and contours.
5. Real-time video processing is achieved, ensuring minimal latency between video input and output.
6. The processed video is displayed through the HDMI output, facilitating easy visualization of results.

Camera Interface:

OV7670 has an image array size of 656 columns by 488 rows (320,128 pixels).

Steps Involved in Camera Interfacing

- Camera Configuration
- Capture Image Data

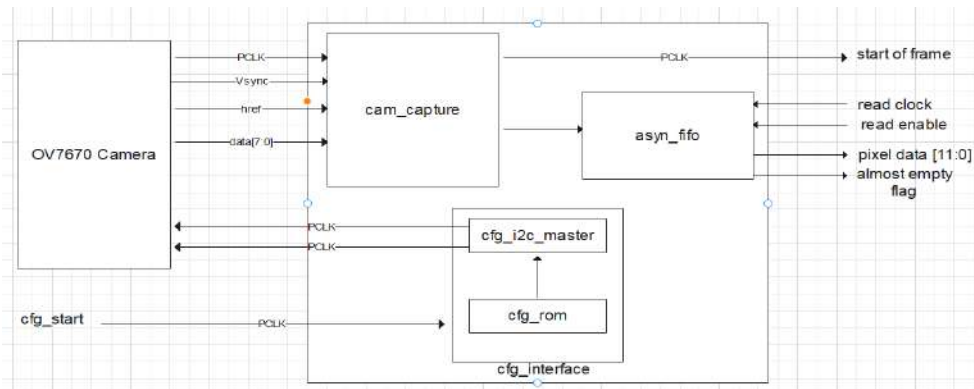


Figure 3.3 Camera Interface Design

Camera Configuration:

The OV7670 camera uses an I2C-like SSCB interface for configuration and data capture. The device control register list contains important configuration parameters such as output format, RGB format, timing signals, and matrix coefficients (MTX1-MTX6) which impact the image quality.

Capture Image Data

- To capture image data, it is necessary to refer to the camera's datasheet and understand the output timing diagram for the selected output format.
- In RGB video passthrough mode, pixels are represented in the RGB444 format and extended to RGB888 for TMDS encoding.
- This format will support display resolutions up to 640x480 at 30FPS.

Protocol Used for Camera Interfacing

SSCB (Standard Serial Camera Interface)

- SSCB is a proprietary serial interface developed by Omni Vision.
- SSCB operates using two lines: a data line (SSCB_SDA) and a clock line(SSCB_SCL).
- Similar to I2C, SSCB interface employs a master-slave communication model.
- Zynq FPGA acts as the master and the camera module acts as the slave.
- SSCB interface allows the master to send commands and read or write data registers within the camera module.

RGB To Grey Scale Conversion:

- The process of converting an RGB image to grayscale involves weighted averaging of the color channels. Typically, the human eye is more sensitive to green, followed by red and blue. As a result, the weighted average method often uses different coefficients for each color channel during conversion.

The preprocessing block converts RGB444 to greyscale for filtering. It uses the algorithm.

$$y = 0.299*R + 0.587*G + 0.114*B$$

Preprocessing (RGB to Greyscale):

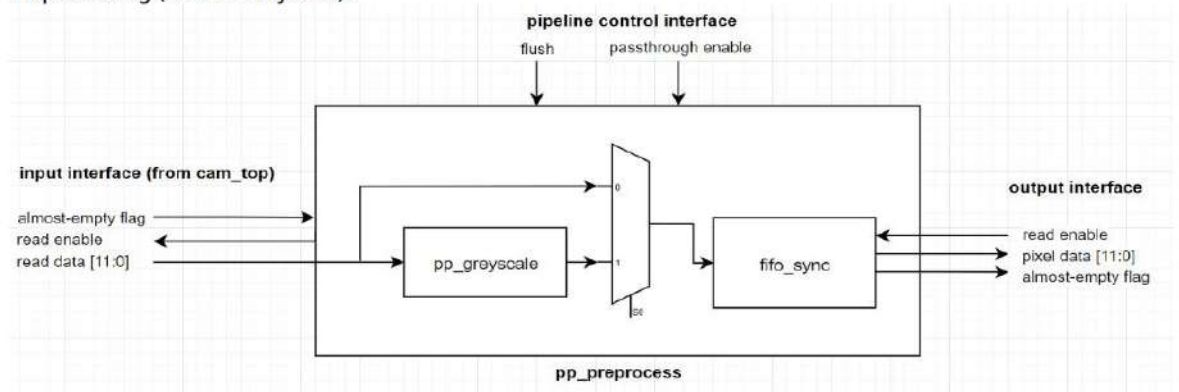


Figure 3.4 RGB to Greyscale

Kernel Processing:

- In the system pipeline, the ps_kernel_control module retrieves data from the preceding stage's FIFO and sequentially populates the line buffers.
- Once the three-line buffers reach maximum capacity, the system begins feeding pixel data nine pixels at a time. Specifically, it retrieves three pixels from each line and processes them through the kernel processing pipeline.
- The output from the kernel processing pipeline is written to a synchronous FIFO.

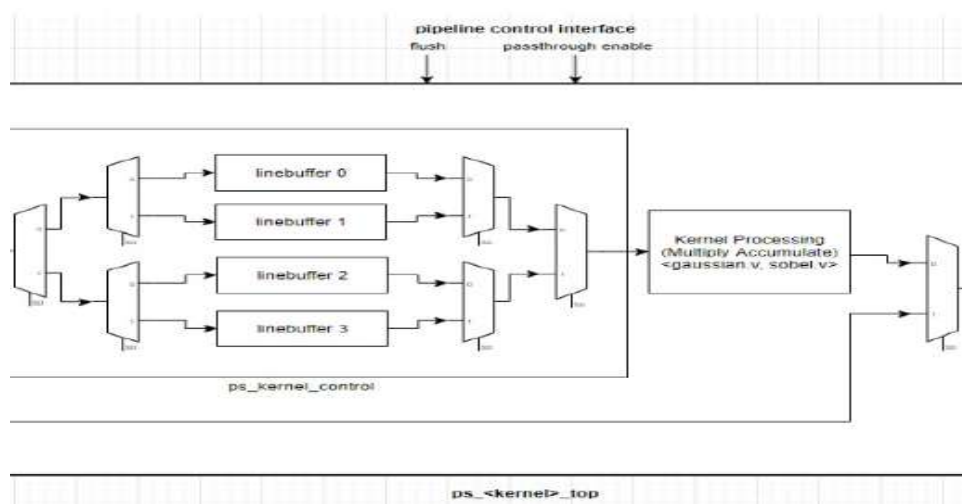


Figure 3.5 Kernel Processing

Gaussian Filter Implementation

- A Gaussian filter is a linear filter used in image processing to smooth an image and reduce noise by convolving the image with a Gaussian function.

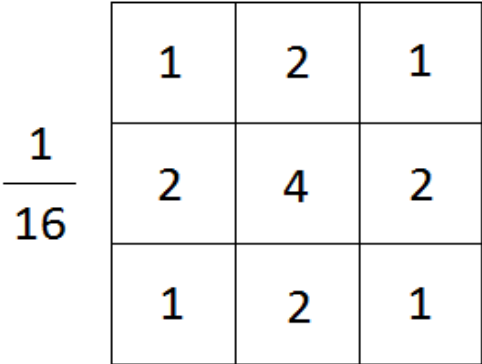


Figure 3.6 Gaussian Kernel

Steps Involved:

- Image acquisition
- Image preprocessing
- Gaussian filter kernel
- Convolution operation
- Post-processing

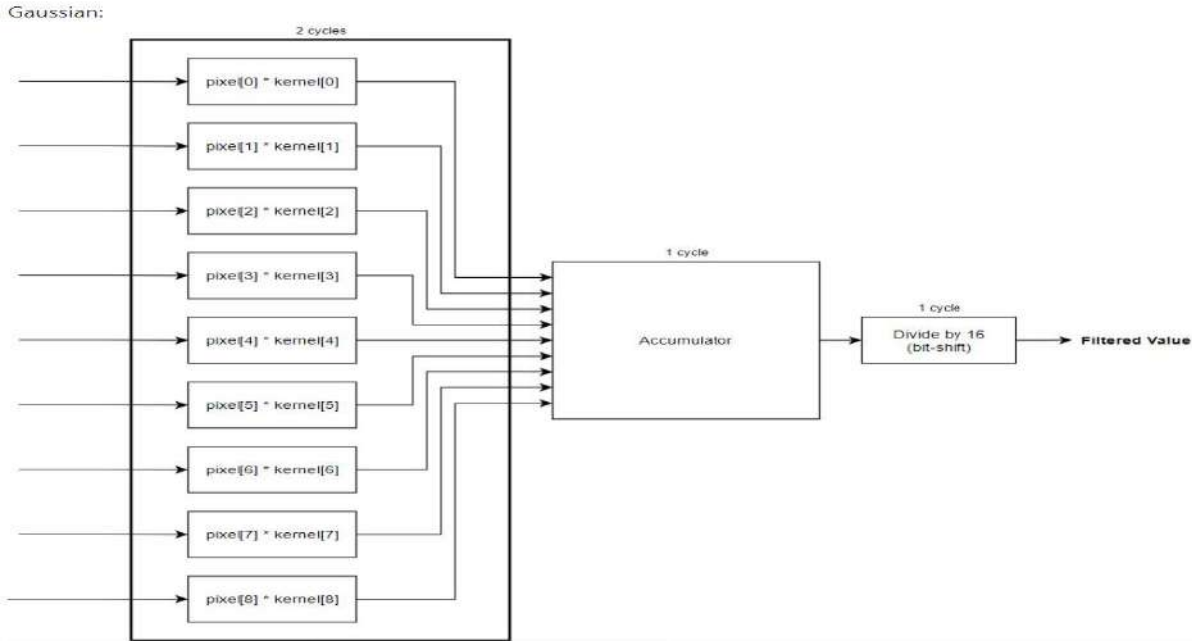


Figure 3.7 Gaussian Filter Implementation

Sobel Filter

The Sobel filter is a type of digital image processing filter that is used for edge detection. It works by convolving an image with a small kernel in the x and y directions, which highlights the edges in the image.

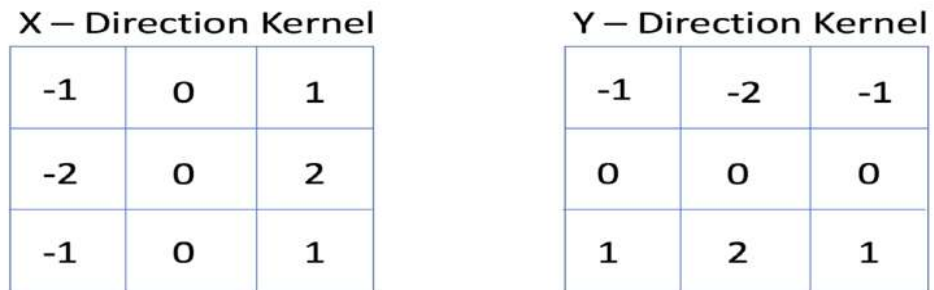


Figure 3.8 Sobel Kernels

Steps Involved:

- Image acquisition
- Image preprocessing
- Edge detection
- Thresholding

Sobel:

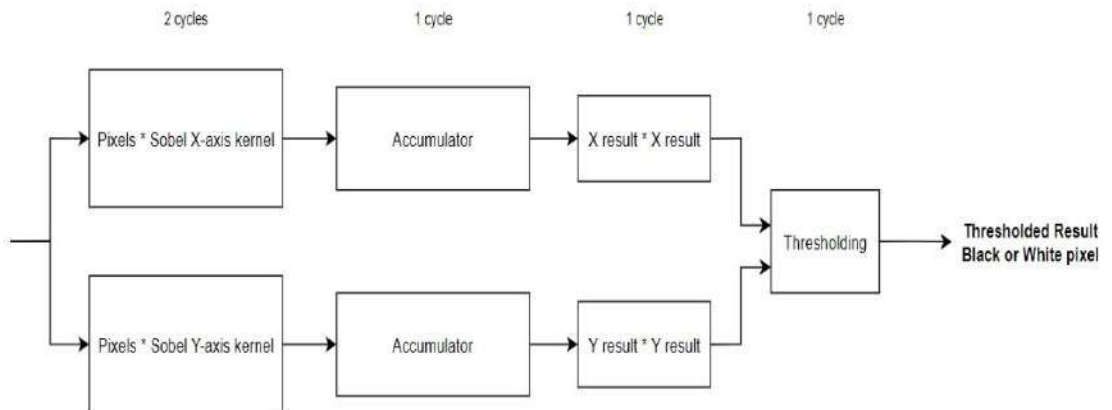


Figure 3.9 Sobel Filter Implementation

3.3 Details About Hardware

The project has following main hardware and software modules listed below with their specifications:

3.3.1 XILINX ZYBO FPGA Board

APPENDIX A

3.3.2 Ov7670 Camera Module

APPENDIX A

3.4 Details About Software Tools

The project has following main software modules with their features listed below:

3.4.1 XILINX VIVADO

APPENDIX A

3.4.2 MATLAB

APPENDIX A

3.5 Details of Final Working Prototype

Our project has the following types of prototypes i.e., hardware prototype and software prototype. They are following:

3.5.1 Hardware Prototype



Figure 3.10 Hardware Prototype

The hardware prototype includes components listed below:

- Xilinx ZYBO FPGA Board
- OV7670 Camera Module

3.5.2 Software Prototype



Figure 3.11 Software Prototype

Chapter 4

RESULTS AND EVALUATION

In this chapter, we will explain all the results what we achieved after completing all what we explained in previous chapters in our work.

4.1 Project Analysis

1. Strengths:
 - a) Real-Time Processing: The project showcases a remarkable strength in its ability to conduct real-time video processing. By utilizing FPGA-based implementation and harnessing parallel processing capabilities, the system efficiently handles video data with minimal delay, making it perfectly suited for time-critical applications.
 - b) Customizability: The system allows users to interact and customize their experience. They can easily toggle video processing on or off and enable or disable specific filters. Moreover, the system offers flexibility by allowing users to adjust the Sobel filter thresholding. This feature empowers them to adapt the system according to different scenarios.
 - c) Noise Reduction and Edge Detection: The video quality is effectively enhanced by the Gaussian lowpass filter, which efficiently reduces noise. Simultaneously, the Sobel filter accurately identifies edges and contours in real-time, contributing to improved video analytics.
 - d) Potential Impact: The project's findings demonstrate the potential impact of the system in various applications. These include surveillance, robotics, medical imaging, and industrial inspection. Its efficiency and accuracy in real-time video processing make it valuable in multiple domains.
2. Performance Evaluation:
 - a) Throughput: The system demonstrates an exceptional throughput rate, effortlessly processing video frames in real-time with no noticeable delays or frame drops.

- b) **Resource Utilization:** The optimized design ensures efficient utilization of FPGA resources, making the system resource friendly. It effectively utilizes the available resources while maintaining optimal performance.
- c) **Video Quality:** The application of the Gaussian filter plays a crucial role in reducing noise and improving the overall quality of videos. It ensures that video frames are clearer.
- d) **Edge Detection Accuracy:** The accuracy of edge detection achieved by the Sobel filter is highly reliable. This makes it a suitable choice for applications that necessitate precise object detection and motion tracking.



Figure 4.1 Video Capture from Camera in RGB

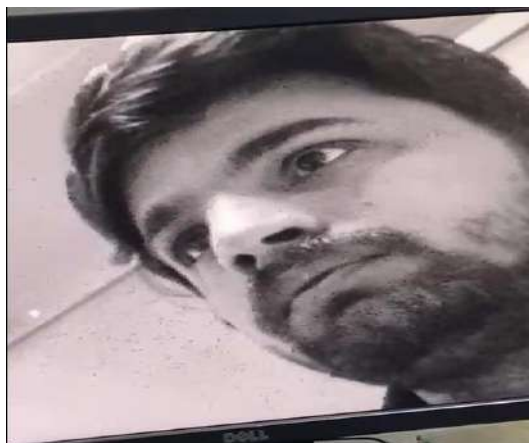


Figure 4.2 RGB to Grey Scale

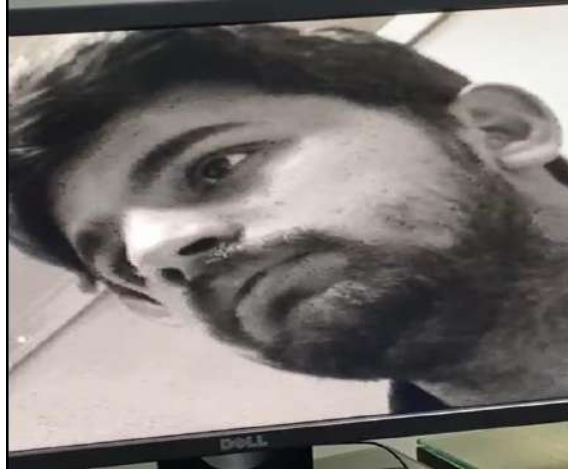


Figure 4.3 Gaussian Low pass Filter



Figure 4.4 Human Pose Estimation

Chapter 5

DISCUSSION

5.1 Discussions on the Findings

We have obtained results and provide a critical analysis of our work explained in previous chapters. We have also made comparisons with some standards or other authentic work done.

5.1.1 Comparison with Initial Project Specifications

- We have successfully achieved real time video, which will support display resolutions up to 640x480 at 30FPS.

5.1.2 Reasoning for short comings

The project's sacrifice of video resolution is its only shortcoming. We achieve a pixel resolution of around 640×480 at 30 frames per second. Consequently, video processing is a little bit sluggish. We picked this since it is an inexpensive camera (OV7670), costing between \$5 and \$8, and because the data sheet is open source and readily available online. may help in interfacing the camera with the FPGA board.

5.2 Limitations

Integrating new hardware components or camera modules can pose a challenge due to potential issues with hardware compatibility and the need for additional interfaces. Such limitations may restrict the system's flexibility in utilizing different camera modules or external devices.

5.3 Recommendations

To optimize performance in the FPGA-based real-time video processing system, it is important to focus on memory design. By minimizing data transfer bottlenecks through optimized memory design, one can greatly enhance system performance. Efficient usage of on-chip memory and careful data buffering can effectively reduce memory access latency, resulting in improved overall performance of the system.

Chapter 6

CONCLUSION

The FPGA-based real-time video processing system successfully achieved its main goals by utilizing the Xilinx Vivado tool and Verilog HDL. This efficient system effectively interfaces with the OV7670 camera module and incorporates Gaussian lowpass and Sobel edge detection filters. These features allow for real-time noise reduction, image enhancement, and accurate edge detection in video frames. With its parallel processing capabilities, the FPGA enables seamless handling of video data, making it ideal for time-sensitive applications in surveillance, robotics, and medical imaging. Additionally, the user-friendly interface allows for convenient control of video processing, with the ability to toggle filters and adjust settings on-the-fly. Despite facing challenges related to timing, resource utilization, and hardware integration, the system demonstrates commendable performance and functionality.

REFERENCES

1. Embedded Real-Time Video Processing System on FPGA. https://www.researchgate.net/publication/232252240_Embedded_Real-Time_Video_Processing_System_on_FPGA
2. Research Real-Time Video Processing with FPGAs. <https://blog.tempus-ex.com/real-time-video-processing-with-fpgas/>
3. FPGA implementation of real time video signal processing using Sobel. <https://www.semanticscholar.org/paper/FPGA-implementation-of-real-time-video-signal-using-Onat/3466d847c5ba166ea126ee36d2dd3f3286b784b2>
4. FPGA based Embedded System Design for Video Processing Application. <https://www.semanticscholar.org/paper/Embedded-Real-Time-Video-Processing-System-on-FPGA-Said-Saidani/948ec611dcc32d9cc421690d764698ea10b50d60>
5. Real Time Image Processing on FPGAs. https://livrepository.liverpool.ac.uk/3028893/1/200832211_Jan2018.pdf
6. A FPGA based efficient camera interfacing architecture for real time video processing. <https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/cds2.12074>
7. Implementing Gaussian Filter by Using VHDL to Blur Images. <https://medium.com/@kocamuha/implementing-gaussian-filter-by-using-vhdl-to-blur-images-8bc40d95fff3>
8. FPGA implementation of filtered image using 2D Gaussian filter. https://thesai.org/Downloads/Volume7No7/Paper_71-FPGA_implementation_of_filtered_image_using%202D.pdf

APPENDICES

APPENDIX A

TOOLS AND SPECIFICATIONS

XILINX ZYBO FPGA Board:

The Xilinx Zybo, developed by Digilent Inc., is a versatile FPGA (Field-Programmable Gate Array) board. It serves as an excellent tool for both embedded processing and the development of digital circuits. Digilent Inc., renowned for their educational and developmental resources in FPGA and microcontroller applications, proudly manufactures this board.

The Zybo board typically incorporates a Xilinx Zynq-7000 series FPGA chip. This family of chips combines a dual core- ARM Cortex-A9 processor with FPGA fabric, enabling both software and hardware development on a single chip.

Memory: The system incorporates various types of memory to support processing tasks. It utilizes DDR3 memory for the processor subsystem and additional memory resources for FPGA implementations.

Peripherals: The board incorporates a variety of peripherals, including USB ports, an Ethernet port, HDMI output, audio input/output, and more. This extensive range of functionalities makes it highly versatile for numerous applications.

Expansion and Connectivity: Zybo provides a range of connectors, headers, and expansion ports that allow seamless interfacing with other devices and expansion modules for customization in application development.

In the realm of Programming and Development, one can utilize Xilinx's Vivado Design Suite to program the FPGA. This comprehensive software suite is specifically designed for FPGA development purposes. In addition, standard ARM development tools can be employed to program the ARM Cortex-A9 cores.



Figure 6.1 FPGA Board

Ov7670 Camera Module:

The OV7670 camera module utilizes a CMOS image sensor, ensuring remarkable image quality while consuming minimal power. It supports various image formats, including JPEG, BMP, and RAW, boasting a maximum resolution of 640x480 pixels. Additionally, this camera module offers several features to enhance user experience such as auto-exposure control, color saturation adjustments, and brightness control.

The OV7670 camera module finds widespread use in numerous applications. These include robotics, drones, home automation, and even educational settings for teaching computer vision and image processing.

To utilize the OV7670 camera module, it is commonly linked to a microcontroller or development board that possesses a camera interface. The communication between the camera module and the microcontroller or development board takes place via either a parallel data bus or a serial interface. Once connected, programmers can create software to manage the camera module's functions and capture images or record videos.



Figure 6.2 OV7670 Camera

Software Used

Xilinx, Vivado Design Suite:

The Vivado Design Suite is utilized to expand the capabilities of system-on-chip development and high-level synthesis. Developed by Xilinx Inc., Vivado stands as a robust and comprehensive design suite catering to FPGA, SoC (System-on-Chips), and CPLD (Complex Programmable Logic Devices) design, programming, and configuration. With significant

advancements over its predecessor ISE (Integrated Software Environment), Vivado offers diverse features and optimizations aimed at enhancing the FPGA design process.

MATLAB:

MATLAB, developed by MathWorks, is a widely used high-level programming language and interactive environment. It finds extensive application in academia, research, and industry across various domains. With its capabilities for data analysis, scientific simulations, image and signal processing, control systems, and machine learning tasks, MATLAB has become an indispensable tool.

CONTACT INFORMATION

1) Muhammad Shoaib

Email ID: muhammadshoaibf19@nutech.edu.pk

Mobile No: +92 321 8833394