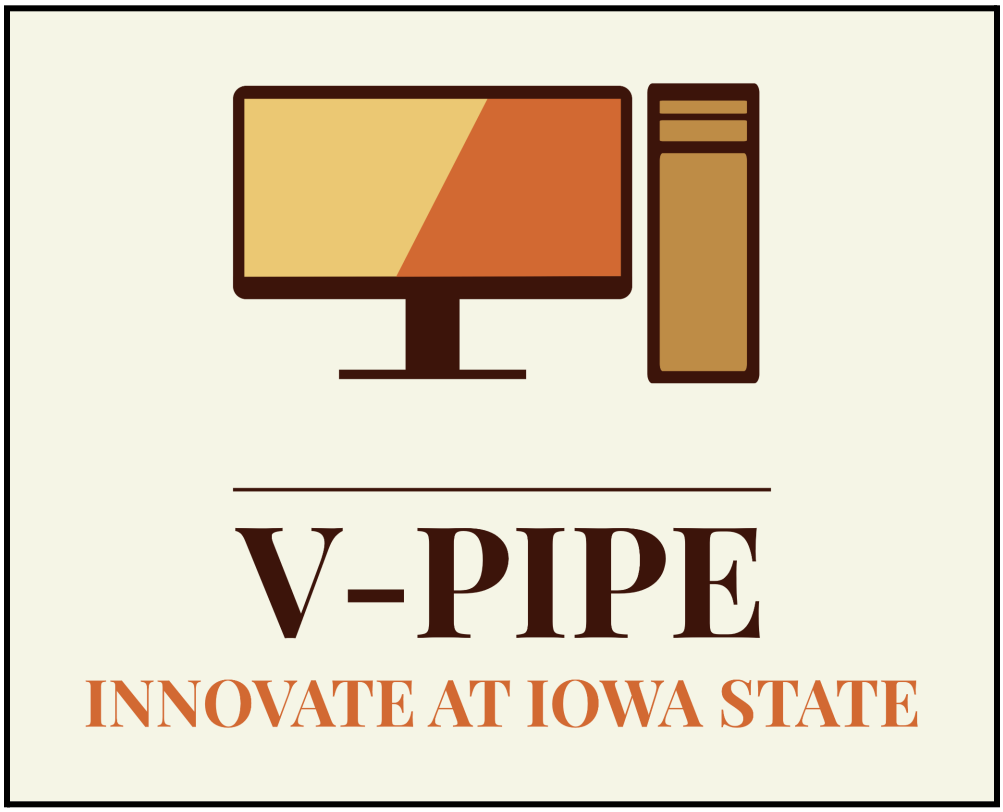
Video Pipeline for Machine Vision



Design Document

Team Number: 6
Adviser: Dr. Phillip Jones

Team Members:
Liam Janda • Taylor Johnson • Ritwesh Kumar • Deniz Tazegul

Team Information

Team Number: 6

Client: JR Spidell

Adviser: Dr. Philip Jones

Team Members/Roles:

- Ritwesh Kumar - Video Stream to FPGA
- Deniz Tazegul - Video Stream to FPGA
- Liam Janda: VDMA to DDRM
- Taylor Johnson: DDRM to Output Display

Team Email: sddec24-06@iastate.edu

Team Website: https://sddec24-06.sd.ece.iastate.edu/

Executive Summary

Background

Eye-tracking seizure detection devices are effective and useful in aiding clinical physicians and healthcare professionals in providing necessary treatment in real time. According to a 2021 Clinical Neurophysiology research study, childhood absence epilepsy (CAE), which is one of the most common forms of childhood epilepsy accounting for 8-15% of all (childhood-onset epilepsy cases), can be utilized for seizure detection. The study used a mobile glasses-type eye tracker to detect sustained upward eye deviation, which takes place in up to 80% of CAE seizure events. This device does not offer performance improvements compared to scalp EEG (electroencephalography) or video EEG technology, but the ease of use over a long time frame (versus scalp EEG) and potential difficulties of capturing video footage of the face in real-time (versus video EEG) makes the wearable eye tracker potentially more practical.

Another eye-tracking device can be used for emotion recognition. Based on a 2020 Sensors survey paper, an eye-tracking device can recognize emotion from a combination of feature data including pupil diameter, EOG (electrooculography), pupil position, fixation duration of the eye, distance between sclera and iris, motion speed of the eye, and pupillary responses. The figure below depicts how emotional recognition can be predicted in real-time utilizing machine learning, eye-tracking, and eye-feature data extraction as discussed in the Sensors survey paper.

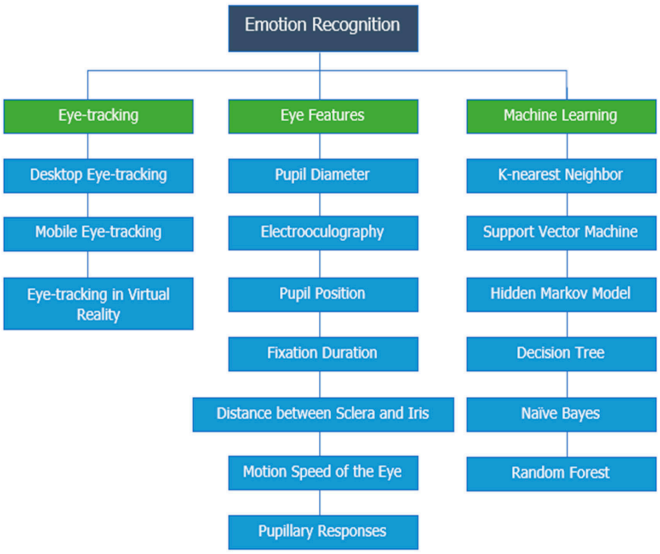


Figure 1: Emotion Recognition Through Eye Movements

Problem Statement

There are numerous applications for a computer vision pipeline that use machine learning algorithms: from medical imaging to self-driving cars to surveillance drones. This project aims to create a proof-of-concept video pipeline using common off-the-shelf materials and software. The inspiration for this design comes from the desire to help disabled or injured individuals increase their quality of life. These individuals can have inhibited fine motor skills that make acute navigation challenging and individuals confined to a wheelchair often require a helper to perform daily living tasks. This project aims to increase independence for these individuals by tracking their eye movements for seizure and emotion detection applications.

Capturing eye movements in real-time for these individuals will be vital in assisting them with their communication and navigational needs. By combining an eye-tracking vision system with other computerized parts, these individuals can interact with their environment in a way that reduces the risk of injury, while gaining a higher sense of independence.

This is just one application for this product which can be used in a broader range. For example, this product could be used as a ready set of components for a programmer to explore an interest in machine learning. After acquiring our product, the programmer can view the existing hardware design components, IP block diagrams, and source code required for running a working video pipeline, taking image sensor (camera) data, and outputting video stream in real-time to a DisplayPort monitor.

After familiarizing themselves with the hardware and software tools and supplemental documentation, the programmer has full control over constructing new hardware IP blocks to introduce new video pipeline pathways for increased functionality of the original product. For instance, they can introduce new and fully customizable, machine-learning IP blocks using hardware-description languages (HDLs) and implement these design overlays into the hardware.

This can be done to add more complexity to the existing product to solve more sophisticated machine-learning tasks. These include but are not limited to image and video analysis (for object classification, detection, and tracking), activity recognition of activities (for people moving, lying down, or standing still), and video content modification (for artificial video generation, object creation, or object removal).

Development Standards & Practices Used

Engineering Standards	Relevance
IEEE Std 2977 2021	Video stream from the image sensor will be sent to a MIPI controller and must adhere to the MIPI-APHY Standard for conversion before being sent to the Ultra96-v2 FPGA board.
I2C Protocol	I2C is the communication system that will be used to transmit incoming data from the camera's image sensor to the Ultra96-v2 FPGA.
AMBA AXI Protocol	The AMBA AXI bus will be used to transport data within the Ulta96-v2 FPGA board. AMBA consists of both the AXI4-Lite bus used to transmit data from the program memory to configure IP registers and the AXI4-Stream bus used to transmit video data between individual IPs within the FPGA including the MIPI controller, VDMA, TPG, and DisplayPort.

Table 1: Used development standards and practices

Summary of Requirements

Functional

- The product will output a live video stream to a DisplayPort monitor.
- Software that routes data through the video pipeline will use a Linux image.
- The product will use an Ultra96-v2 FPGA board and daughter card.
- The product will use a Sony IMX219QH5-C image sensor.

Non-Functional

- The product will operate minimally at a resolution of 640x480p at a frame rate of 15 fps. This is based on the limitations of the image sensor and FPGA hardware.

Assumption

- The user's face is well-lit by a light source.

* Functional requirement is what a product should do

Applicable Courses from Iowa State University Curriculum

CPRE 281: Digital Logic

- Introduction to Verilog, a hardware description language (HDL).

CPRE 288: Embedded Systems I

- Programming with C for embedded system applications
- Move between register and memory space

CPRE 488: Embedded Systems Design

- Introduction to Vivado to build custom hardware overlays for embedded systems including FPGAs

EE 424: Digital Signal Processing

- Signal processing: sampling, windowing, analog-to-digital (ADC) conversion

New Skills/Knowledge acquired that was not taught in courses

Below is a list of new skills/knowledge our team has/will acquire that is not part of our Iowa State curriculum that we will need to complete the project.

Tools

- Integrated Logic Analyzer (ILA)
- PYNQ open source software project for Python embedded systems developers

Skills

- FPGA Design
- Parsing through hundreds of pages of component documentation
- Waterfall-Agile workflow
- Building a systems-level block diagram connecting hardware IPs

Knowledge Gained

- Deeper understanding of image sensors
- Working in a Linux OS environment
- Difference between PYNQ vs. non-PYNQ environments
    - PYNQ accessible via a Jupyter Notebook server
    - Non-PYNQ accessible via a terminal

Table of Contents

List of Figures, Tables, and Definitions

Figures

Tables

Definitions

IMX219 image sensor: camera

MIPI: mobile industry processor interface

CSI-2: high-speed camera serial interface

D-PHY: physical communication layer

VDMA: video direct memory access

DDR: double data rate (memory)

FPGA: field programmable gate array

PYNQ: python productivity for Zynq (python embedded systems developers)

ML: machine learning

OS: operating system

Vivado: custom hardware overlays

ILA (Integrated Logic Analyzer): capture and analyze input and output signals (I/O

1. Introduction

1.1 Problem Statement

There are numerous applications for a computer vision pipeline that makes use of Machine Learning algorithms: from medical imaging to self-driving cars to surveillance drones. This project aims to create a proof-of-concept video pipeline prototype using common off-the-shelf materials and software. The video pipeline will take video data from an image sensor, route it through an FPGA board, and output the video stream to a DisplayPort monitor.

The project is part of a history of current and past senior design teams across several universities, with the inspiration being to help disabled or injured individuals increase their quality of life by tracking their eye movements in real time. These individuals can have inhibited fine motor skills that make acute navigation challenging, and individuals confined to a wheelchair often require a helper to perform daily living tasks. While measuring eye movements in real-time is critical for the broader project, it is not a primary concern for this team. However, the requirements and design are set up to facilitate those capabilities in the future. Thus, having low latency is also not a primary consideration. Furthermore, while it is expected that the final product will need to operate under a variety of lighting conditions, for the scope of this project, it is assumed that the face of the user is well-lit by a light source.

Capturing eye movements in real-time for these individuals will be vital in assisting them with their communication and navigational needs. By combining an eye-tracking vision system with other computerized parts, these individuals can interact with their environment in a way that reduces the risk of injury, while gaining a higher sense of independence. This is just one application for this product which can be used in a broader range. For example, this product could be used as a ready set of components for a programmer to explore an interest in machine learning.

1.2 Intended Users

The primary user for this project is a disabled individual who is confined to a motorized wheelchair and has limited navigation. The video pipeline in combination with machine-learning algorithms will be used to capture a user's eye movements to aid in navigation. Even though this is how our client envisions using a video pipeline for computer vision, our users span a wide range of applications such as personnel in the medical field, surveillance, and people who are interested in learning about cutting-edge technology. Below are a few generalized user personas the team created to understand who the project is designed for, including a disabled individual, a machine learning enthusiast, and a caretaker.

Disabled User

This individual is wheelchair-bound and has a disability. They have difficulty with their fine and gross motor skills. This makes it difficult for the disabled user to perform tasks that require a steady hand and precision. A couple of their favorite pastimes include playing board games and art. However, their motor skills make moving game pieces challenging, and painting within certain lines or regions of a canvas can be difficult.

Unfortunately, the disabled user also experiences seizures and requires the assistance of a caretaker to assist with daily motor tasks such as eating. This individual could benefit from a computer vision camera system by having the camera fixed to their wheelchair to monitor when they have a seizure and quickly alert their caretaker. The camera system could also help to communicate some of the disabled user's other needs, based on monitoring their eyes and passing that video stream through a machine learning algorithm.

Machine Learning Enthusiast

This individual is a middle-aged programmer whose job is to code ML algorithms that assist disabled individuals. They have a special interest in integrating ML into a video camera system, as a way to monitor and learn the specific needs of an individual, predict those needs, and communicate them to another person without using verbal cues.

The ML enthusiast enjoys computer programming and often find themself using their free time to implement their own projects. They are a problem-solver and are always looking to find practical solutions to everyday challenges. They are a busy person who needs a device that can easily process video data and support various ML applications to help them create innovative solutions for disabled individuals. They would benefit from a computer vision video pipeline that is available as a ready set of components because it takes the stress away from needing to understand low-level hardware design and makes it easy for them to develop, implement, and test their ML applications that rely on a video processing system.

Caretakers

This individual is a nurse who has a passion for helping people. Their job consists of caring for patients with disabilities. Their patients range in their levels of independence. They assist many with accomplishing daily living tasks. Unfortunately, funding is tight and they have an increasing patient load, they need an efficient way to monitor their patients and their needs so they can effectively deliver quality care. The caretaker user is not necessarily the most technologically savvy person, but they are driven to help their patients and willing to learn. They need a way to increase their productivity in assisting their patients without sacrificing their care. A computer vision camera system could help them monitor and quickly identify patient needs, whether that be communication, physical movement, or health, so they can more effectively use their time.

2. Requirements, Constraints, and Standards

2.1 Requirements & Constraints

Below is a table of requirements that the team has identified based on the problem statement and our client's needs that impacted how the team chose to design a proof-of-concept software implementation of a video camera for computer vision. The table also identifies limitations to our design that we are unable to change. These are listed as constraints. Since our project consists of a proof-of-concept design, there are no aesthetic requirements at this point in time.

Description of Requirement	Constraint	Type
The video pipeline will output a live video stream to a DisplayPort monitor.	Yes	Functional
Software-encoded video pipeline will route data using a Linux image.	Yes	Functional
The video pipeline must execute using an Ultra96-v2 FPGA board.	Yes	Functional
The video pipeline will use a Sony IMX219QH5-C image sensor to capture live video feeds.	No	Functional
The video stream should operate at a minimal resolution of 640x480p and frame rate of 15 FPS. This is based on the limitations of the image sensor and FPGA hardware.	No	Non-functional
The user's face is well-let by a light source.	No	Assumption

* Functional requirement is what a product should do

Table 2: Requirements and Constraints

2.2 Engineering Standards

Engineering Standards	Relevance
IEEE Std 2977 2021	The video stream from the image sensor will be sent to a MIPI controller and must adhere to the MIPI-APHY Standard for conversion before being sent to the Ultra96-v2 FPGA board.
I2C Protocol	I2C is the communication system that will be used to transmit incoming data from the camera's image sensor to the Ultra96-v2 FPGA.
AMBA AXI Protocol	The AMBA AXI bus will be used to transport data within the Ulta96-v2 FPGA board. AMBA consists of both the AXI4-Lite bus used to transmit data from the program memory to configure IP registers and the AXI4-Stream bus used to transmit video data between individual IPs within the FPGA including the MIPI controller, VDMA, TPG, and DisplayPort.

Table 3: Engineering Standards Used

3. Project Plan

3.1 Project Management/Tracking Procedures

Our project is best suited for a Waterfall-Agile hybrid project management style. Its combination of procedural and iterative approaches aligns with the project's long-term goals and requirements due to the nature of software development and hardware integration. A hybrid style allows for a structured approach while providing flexibility for changes when issues arise.

The block diagram below depicts the specific methodology the team is using: from an initial Waterfall phase where objectives and requirements, component research, and system design are done sequentially, to an iterative Agile phase where development, integration, testing, and review are done rapidly in design sprints.

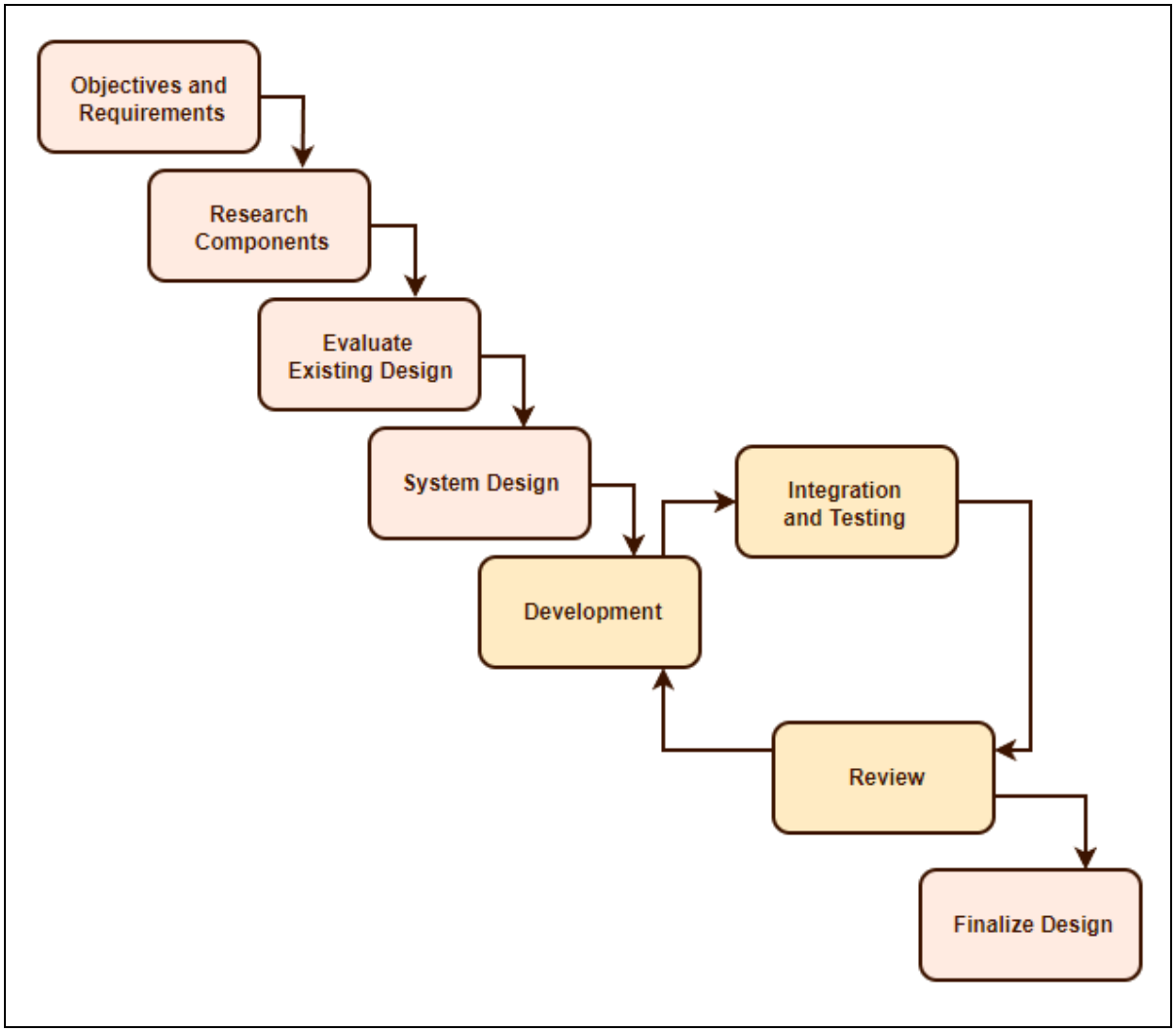


Figure 2: Waterfall-Agile Hybrid Block Diagram

Based on our project management style, here are our project goals and contexts:

- Doing component research first, then rapid design and testing periods.
- One weekly scheduled client, advisor, and team meeting.
- Approximately six or more hours are spent on the project per team member each week.
- Flexibility to add or postpone client or team meetings based on project progress.
- Intermittent check-ins from the client for each member of the team.
- Planning requirements can be large or small based on progress made on component research, code implementation, etc.

Our team will track the progress of the project primarily through our shared GitHub repository.

3.2 Task Decomposition

The team's project task decomposition can be broken down into four subsystems, one for each team member.

Image Sensor to MIPI Controller

Configure the camera properly using the I2C protocol and send video stream data to the MIPI controller on the Ultra96-v2 FPGA, which consists of:

- Writing PYNQ-compatible code that writes to the appropriate I2C port and configures the camera.
- Reading the camera datasheet and documentation for PYNQ libraries, and finding corresponding I2C ports on the Ultra96-v2 board.

MIPI Controller to VDMA

Configure the MIPI controller to receive and transmit camera data video stream correctly. Consisting of:

- Monitoring status registers to confirm data is being handled correctly.
- Understanding data stream formats as received from the IMX image sensor and the corresponding status register information.
- Understanding configuration requirements for input and output streams in order to configure the MIPI controller to accept the correct input stream from the IMX sensor and transmit the correct output stream to VDMA.

VDMA to DDRM

Utilizing AXI VDMA on the Ultra96-v2 to allow direct memory access between memory and AXI4 video stream data. This means:

- Properly connecting the VDMA to DDRM and the CSI-2 module.
- Understanding the requirements and limitations of the AXI4-Stream and the memory module interfaces.
- Creating a system to hand off video data to the DisplayPort through memory.
- Understanding the AXI4-Lite configuration registers.
- Properly configure the VDMA using the AXI4-Lite interface.

DDRM to Output Display

Video data stored in DDR memory on the Ultra96-v2 must be output to a display monitor through:

- Understanding how data is being transferred from DDRM to the display.
- Understanding the requirements for the DisplayPort, TPG, and VDMA.
- Determine the appropriate register configurations.
- Use PYNQ libraries to read from DDR memory and transport data to the DisplayPort.

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

The team has identified certain key milestones and metrics to track the progress of our project, shown below:

- **Milestone 1:** Illustrate and describe connections between components and identify register configurations.
- **Milestone 2:** Evaluate the existing bare metal implementation of the video pipeline to determine register configurations and identify which aspects of the code will require refactoring to execute in a Linux environment.
- **Milestone 3:** Design the video pipeline based on register configurations and block diagram mapping all component inputs and outputs.
- **Milestone 4:** Create software driver and test files to properly interface with the Ultra96-v2 and conduct thorough testing and debugging at each step to ensure the pipeline functions as desired.
- **Milestone 5:** Output the camera video stream data to a display monitor using a Linux environment.
- **Metrics:**
  - Video stream data is output with a predefined resolution and frame rate, such as 640x480p at 15 fps.
  - Video stream data is output in the correct pixel-array format, such as 3280x2464 (as intended).
- **Stretch Goal:** Pass video stream data through a Machine Learning (ML) algorithm, in which the output of the ML algorithm augments the data.
  - **Metrics:**
    - If the ML algorithm is tracking eye movement for both eyes and given a known ground truth bounding box containing a person's eyes, the IOU (intersection over union) of the predicted bounding box should be:
      1. IoU > 0.5 (is "decent")
      2. IoU > 0.7 (is "pretty good")
      3. IoU > 0.9 (is "almost perfect")
    - Given a known ground truth video stream not obstructed by glare and if the ML algorithm reduces glare on an obstructed video stream, computing the SNR (signal-to-noise ratio) of the ML video and ground truth video can help determine how well the algorithm performed.

3.4 Project Timeline/Schedule

The team has created a Gantt chart that contains a realistic schedule for our project objectives. The first section discusses specific project setup requirements, as shown in Figure 3 below.

WBS NUMBER	TASK TITLE	DEPENDENCIES	TASK OWNER	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE
1.00	**Setup Development Environment**						
1.01	Introduction to the project with client. Identify key system level requirements		All	2/11/24	2/11/24	0	100%
1.02	Setup periodic meeting time, telegram project thread		All	2/5/24	2/5/24		100%
1.03	Setup Github repository and get everyone invited to it		JR	2/10/24	2/13/24	2	100%
1.04	Train each team member on the use of github		Liam, Ritwesh, Taylor, Deniz	2/13/24	2/20/24	0	100%
1.05	Setup Github Environment, check in an example file		Liam, Ritwesh, Taylor, Deniz	2/12/24	2/17/24	5	100%
1.06	identify, buy and receive image sensor modules		JR	2/13/24	2/16/24	3	100%
1.07	Send HW to team	1.6	JR	2/21/24	2/24/24	2	100%
1.08	Receive Ultra96 and Learn how to execute code on it	1.7					100%
1.09	Learn to setup Ultra96 wifi and connect to internet	1.7					100%
1.10	Learn to make a backup of the SD card	1.7					100%
1.11	Identify way to make Ultra96 publically available (ask class TA for assistance on this)	1.7					100%

Figure 3: Setup Development Environment

The second section (shown in Figure 4) discusses specific objectives each team member must accomplish to familiarize themselves with the preexisting code, learn about the different components they are responsible for, and share their findings with the rest of the team.

2.00	**Project Acclimation**						
2.01	Read and understand prior team's code. Each team member - give presentation on what code exists from prior team and what are the givers/takers per person	1.3	Liam, Ritwesh, Taylor, Deniz	2/13/24	4/14/24	62	100%
2.02	Run previous team's code on ultra96	1.7	xxxx	2/13/24	4/28/24	76	100%
2.03	Each member creates a slide deck describing their section of the design in as much detail as they know		Liam, Ritwesh, Taylor, Deniz	2/13/24	4/14/24	62	100%
2.04	Learn about the MIPI Rx IP block from Xilinx. Review prior team's code. Create presentation and present to team		Deniz	2/13/24	3/31/24	48	100%
2.05	Learn about the VDMA IP block from Xilinx. Create presentation and present to team		Liam	2/13/24	3/31/24	48	100%
2.06	Learn about the Image sensors. Review prior team's code. Create presentation and present to team		Ritwesh	2/13/24	3/31/24	48	100%
2.07	Learn about TPG. Review prior team's code. Create presentatio and present to team.		Taylor	2/13/24	3/31/24	48	100%
2.08	Learn about the Ultra96. Create presentation and present to team		Liam	2/13/24	3/31/24	48	100%

Figure 4: Project Acclimation

The third section (shown in Figure 5) discusses the relevant planning and project definitions the team will identify to gain a better understanding of the overall scope of project work and how each team member can contribute.

3.00	Project Definition and Planning					
3.01	Project Proposal - Rough Draft and submit	Deniz	2/13/24	2/13/24	0	100%
3.02	Develop Requirements	All	2/13/24	11/22/24	279	100%
3.03	Identify Block diagram in/outs for each section of the pipeline	All	2/13/24	5/27/24	104	100%
3.04	Identify the transport layer for each to/from segment of the block diagram	All	2/13/24	5/27/24	104	100%
3.05	Identify division of project scope among team members	All	2/11/24	2/11/24	0	100%

Figure 5: Project Definition and Planning

The fourth section (shown in Figure 6) describes the key stages of the project development process that the team must complete to finish the project on schedule.

4.00	Development					
4.01	Identify registers for each block to be configured	All	2/13/24	5/27/24	100	100%
4.02	Identify what we are tramitting (video format) To/from each receiver	All	2/14/24	5/28/24	42	100%
4.03	Modify existing code to monitor status registers at each reciever stage	All	2/15/24	11/22/24	277	100%
4.04	Modify existing OV sensor code to use IMX sensor	All	2/16/24	5/30/24	104	100%
4.05	take image from SD card and display on display port connected monitor	All	2/17/24	5/31/24	104	100%
4.06	Using FPGA image #1, Get TPG generated image into DDR memory	All	2/18/24	6/1/24	103	100%
4.07	Get TPG generated image onto display port connected monitor	All	2/19/24	6/2/24	103	100%
4.08	Use prior team's code to monitor signals coming in from OV camera using FPGA image #2	All	2/20/24	11/22/24	272	100%
4.09	modify MIPI Rx, camera Tx and CSI Rx registers so the video data gets into the CSI block	All	2/21/24	11/22/24	271	100%
4.10	Now that we have video going into CSI block, send it to the DDR memory	All	2/22/24	11/22/24	270	100%
4.11	now that we have camera video in DDR memory - send it to the display port connected monitor	All	2/23/24	11/22/24	269	100%
4.12	Map section inputs/outputs to determine data widths across sections	All	2/24/24	4/1/24	37	100%

Figure 6: Development

The fifth and final section (shown in Figure 7) details the first and second-semester project presentation deadlines.

5.00	Project - Presentations					
5.10	Midterm Presentation	All	2/12/24	4/21/24	69	100%
5.20	Final Presentation	All	2/12/24	12/8/24	296	100%

Figure 7: Project Presentations

3.5 Risks and Risk Management/Mitigation

For each project task described in the task decomposition and Gantt chart sections, the team identified potential risks and risk management strategies (shown below) necessary to limit the significance and severity of these risks. Furthermore, the team estimated a risk factor probability score for each risk between 0 and 1 probability, with a risk factor probability greater

than 0.5 defined as a high-severity risk that must be mitigated through an appropriate risk management strategy.

- There are low safety risks associated with handling the Ultra96-v2 and its software. The maximum voltage supplied to the Ultra96-v2 is 12 V, so electrical hazards are limited and the associated risk factor probability is less than 0.5.

- There are some small chances the hardware is not working and may need replacement (i.e. a risk factor probability greater than 0.5). In this case, the team will reach out to the client and class professors to determine ways of fixing the issues in a timely, cost-effective manner.

- Depending on the team's progress on the first objective of outputting video stream data in a Linux environment, some low to moderate chances may delay our stretch goal of using an ML algorithm. Furthermore, there are some non-zero chances of implementing the algorithm that may prove more difficult than predicted. Both of these risks have a risk factor probability greater than 0.5.

    - If the team is not satisfied with its progress with the project goals, it may contact our client, faculty advisors, and ECpE faculty for technical assistance.

    - Furthermore, the team may alter its original ML algorithm task to ensure that it can realistically accomplish it between now and the end of the Senior Design Project.

3.6 Personnel Effort Requirements

Based on client and advisor feedback, the following task decomposition versus time table was built.

Task	Hours	Reasoning
Set up the hardware and connect to WIFI	2-3	The team must set up the hardware and WIFI to run the code on the FPGA board
Learn how to communicate with the camera by running the prior team's code (be able to send & configure the camera using I2C using the code from before)	4-6	The team will need to understand how the prior team's implementation works in software and hardware to use as a working reference
Modify existing OV sensor code to use IMX sensor	20	The camera sensor code must be modified to run using a different sensor
Run the existing code on the hardware and observe how the registers are updating for MIPI controllers	4-6	The team must understand how the MIPI controller worked for the previous team
Load an image to an SD card and get the image to display on a monitor using the FPGA	4-6	This is essential in seeing the video data output at the end of the video pipeline
Using FPGA image #1, Get the TPG-generated image into DDR memory	20	Another way to output video data is by using a Test Pattern Generator (TPG) Getting the TPG data into DDR memory is the first step in this process
Get TPG generated image onto display port connected monitor	10	This is the final step in outputting the TPG video data to a display monitor
Modify MIPI Rx, camera Tx, and CSI Rx registers so the video data gets into the CSI block	10	From the camera, the video data goes to the CSI block as the first stage of the video pipeline
Now that the team has video data going into the CSI block, send it to the DDR memory	20	The next step is sending video data to DDR memory
Now that the team has camera video in DDR memory - send it to the display port connected monitor	10	Finally, video data can be taken out of DDR memory and output to the display monitor

Table 4: Project Task Decomposition

3.7 Other Resource Requirements

The team has identified resources (other than financial) to help complete the project on schedule as shown below.

- If a technical challenge occurs in the design process, the team will seek technical guidance from our client, faculty advisors, and ECpE faculty.
- The team will need the datasheets for the image sensor, MIPI controller, VDMA, TPG,, etc.
- The team will require the previous team's code in the GitHub repository.

4. Design

4.1 Design Context

Our project design is a video pipeline system designed for computer vision with the intent of helping users with illnesses or disabilities that limit their independence.

4.1.1 Broader Context

Below is a table with relevant considerations for the broader context in which our project exists. We examine the impact our design will have on the communities we're designing for and what societal needs our project addresses.

Area	Description	Examples
Public health, safety, and welfare	How does your project affect the general well-being of various stakeholder groups? These groups may be direct users or may be indirectly affected (e.g., solution is implemented in their communities)	Product design is intended to increase safety for users with illnesses or disabilities by assisting in their navigation and communication needs.
Global, cultural, and social	How well does your project reflect the values, practices, and aims of the cultural groups it affects? Groups may include but are not limited to specific communities, nations, professions, workplaces, and ethnic cultures.	Not applicable to the project design as it would not noticeably change our design if we did.
Environmental	What environmental impact might your project have? This can include indirect effects, such as deforestation or unsustainable practices related to materials manufacture or procurement.	The product will use commercially available components with a consideration on limiting the amount of power consumption.
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader economic effects on communities, markets, nations, and other groups.	The product needs to remain affordable for target users which is why the design uses commonly available components contrary to custom components which require a more expensive design process.

Table 5: Broader Context Analysis

Our project design is a video pipeline system designed for computer vision. The most relevant broader context issues that relate to our project are public health, safety, and well-being. Our product aims to aid primarily in navigation and communication for individuals with illness or disabilities that limit their independence therefore, the design centers around improving their quality of life. There are also considerations being made to limit infractions on the personal data and privacy of users. Our team is not taking economic factors into huge consideration, our primary focus is functionality and performance, however, commonly available component choice and power consumption are areas that are being taken into account and have an effect on the overall cost of the system. This contrasts some of the products already on the market that use custom components that require a more involved and expensive design process. Our team is not considering the global, cultural, and social aspects of the broader context and is not convinced the project design would change noticeably if we did.

4.1.2 Prior Work/Solutions

This discussion relates to the market research the team has done for computer vision products and applications related to our product design. These include work from the following companies Intel, LUCI, and EyesOnIt.

Competitor	Product	Description

Intel	An industry leader in computer vision systems	• An established company in hardware design
		• Top-of-the-line custom products
		  ○ Parts made in-house
		  ○ Processor, memory, graphics processor, etc.
		  ○ Dial in the capabilities of the system

LUCI	Outward-facing wheelchair video camera attachment	• Software/hardware
		  ○ Environment hazard detection
		  ○ For electric wheelchairs
		• For people with physical disabilities

EyesOnIt	Video Surveillance System	• Custom configuration ○ Little to no programming knowledge is needed • Integrable into surveillance systems

*Citations can be found in the appendix.

Table 6: Competitor Analysis

The table below depicts the pros and cons of our product (V-PIPE) along with the work done by Intel, LUCI, and EyesOnIt.

Producer	Pros	Cons
V-PIPE (Ours)	Off-the-shelf hardware	Proper configuration
Intel	Efficient	Specialized components with longer development time
LUCI	Attachable to existing wheelchairs	Niche market
EyesOnIt	Easily configurable	Requires own hardware

Table 7: Market Pros and Cons

4.1.3 Technical Complexity

As an embedded system, our project contains internal and external technical complexities that demonstrate the level of expertise we have gained as undergraduates in the ECPE department at Iowa State. Furthermore, individual team member expertise development as it relates to the ISU curriculum is included.

The project demonstrates a level of external complexity in that it makes use of near-state-of-the-art technology. The hardware that we are using includes an Ultra96-v2 FPGA board and an IMX219 image sensor. Understanding how to interface between hardware and software involves reading numerous datasheets in order to understand the FPGA's hardware capabilities. Figure 8 shows the external complexity of the board itself. The anatomy of the Ultra96-v2 FPGA board is shown below.

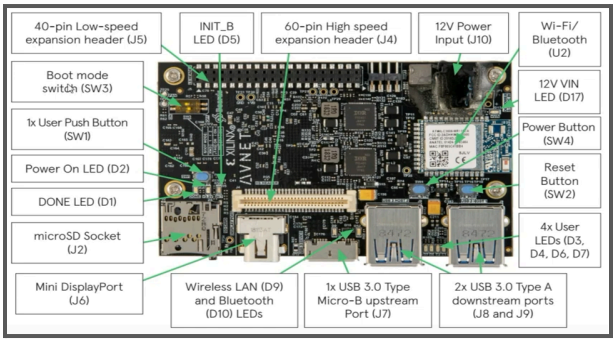


Figure 8: Anatomy of Ultra96-v2 FPGA

In addition, our project has a high level of internal complexity because it involves multiple hardware and software components and subsystems. The major blocks within our design include the IMX219 camera image sensor, the MIPI receiver, VDMA, DDR memory within the processing subsystem, and the DisplayPort. Figure 11 shows the block diagram illustrating the block's connections. These hardware blocks are configured with the help of Vivado IPs (intellectual property). Each with its own datasheet which helps to know how to configure the hardware via internal registers, as well as poll status registers for debugging purposes to watch how data flows through the video pipeline.

Interfacing with hardware involves low-level programming knowledge that few individuals have. Our project builds a camera from scratch showcasing a deep understanding of how cameras work.

Additional Expertise Demonstrated include:

Configuring Image Sensors

- Signal processing
- Sampling, windowing, ADC (analog-to-digital converter)

Programming in Python

- CPRE 288 used C
- Move between register and memory space

Configure MIPI Controller

- Correct data transfer protocol

Code Migration to PYNQ

- Memory access in Python vs C/Shell
- Building off previous teams
- Understanding of Shell scripts, C, Linux OS
- Replace frame buffer with VDMA

VDMA

- Configure registers
- Understand datasheets

Configuring Display

- Vivado overlays (for instantiating TPG & VDMA IPs)
- Configure registers
- Understand datasheets

4.2 Design Exploration

4.2. Design Decisions

Based on client feedback, previous Senior Design Team work, and research, the team made the following design decisions:

- The team decided to use an IMX219 image sensor for the video pipeline instead of the OV5647 image sensor that a previous Senior Design Team utilized.
    - The IMX219 sensor is cost-effective.
    - The IMX219 sensor had more documentation accessible versus the OV5647 image sensor.
- The team agreed on a minimum resolution of 640x480p, RGB color format, and 8-bit component width.
    - Resolution and frame rate are limited by the FPGA and image sensor hardware.
    - The previous Senior Design Team applied RGB and 8-bit component width in their design.
    - These color formats are practical for most DisplayPort monitors as they are supported.
- The team opted to write PYNQ-compatible code that can run in a Linux environment.
    - PYNQ can simplify the FPGA design process using higher-level abstractions and overlays that avoid needing to know the fine details of low-level hardware design.

The team identified a specific design assumption for the product to function properly: the user's face should be well-lit by a light source. This is to account for dim lighting conditions.

Eight potential design approaches were identified and can be seen in the Lotus Diagram below, where each box is assigned a letter in the upper left corner representing different potential design approaches.

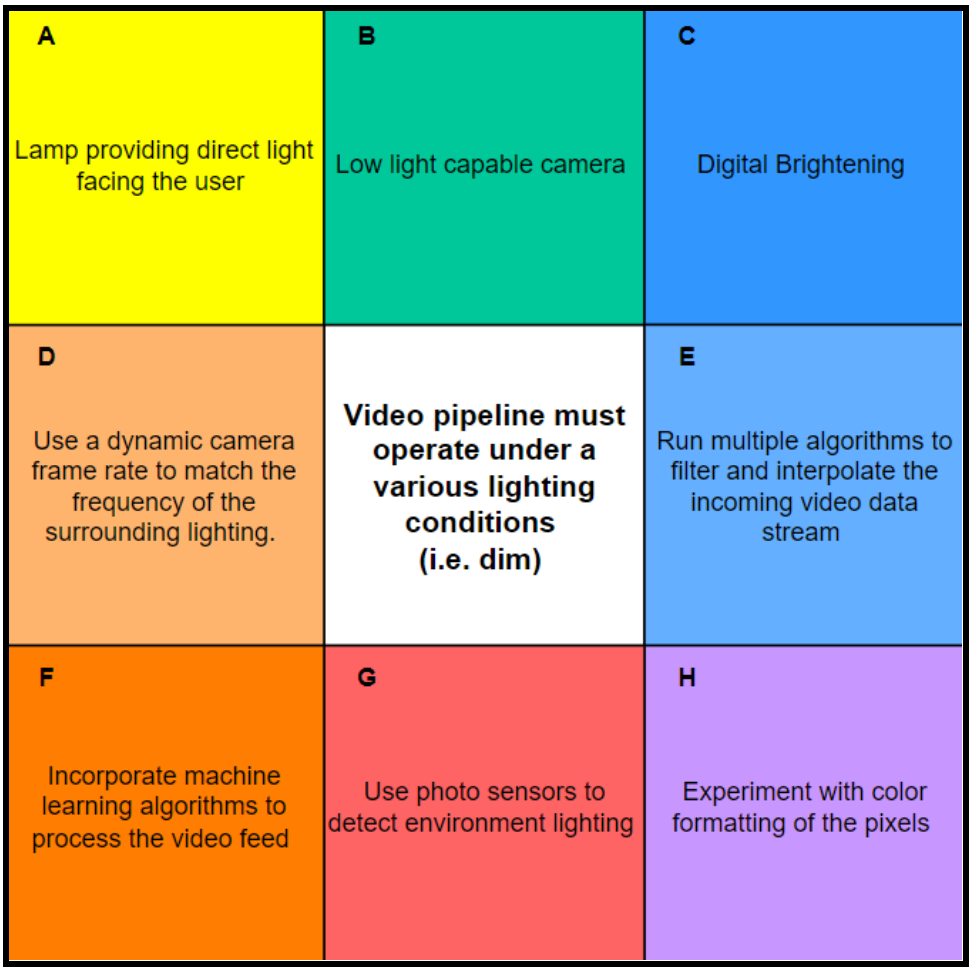


Figure 9: Lotus Blossom Diagram Ideation Design Approaches A-H

4.2.2 Ideation

The team utilized the Lotus Blossom approach to brainstorm potential solutions for the product operating under different lighting conditions. Shown below are the approaches and corresponding Lotus Blossom Diagrams.

Design Approach A

Uses a direct light source pointed towards the user for improved visibility of the output video shown on the monitor. This could include:

- Using a light source directed at the user
- Adding a ring light to the circumference of the camera lens
- Attaching a light configured for auto-flash
- Creating a consistent lighting

Design Approach B

Explores ways to capture video feed under low light conditions including:

- Using a low-light-capable camera
- Adding night vision
- Using a photo sensor that can adjust modes of operation based on lighting

Design Approach C

Looks at ways to enhance video pixel brightness via software to increase the quality of the video display in poorly lit environments. Below are potential methods:

- Using A.I. brightening
- Digital brightening techniques

Design Approach D

Examines the potential for using a synchronized camera frame rate based on the surrounding lightning frequency to reduce flickering. Below are the considerations:

- Use a dynamic camera frame rate that matches the light frequency from the surrounding environment
- The camera set to match a specified number of light flickers
- Use a high frame rate camera
- Use photo sensors to gather flickering data and adjunct the frame rate accordingly

Design Approach E

We could run the video data through multiple algorithms to enhance the output visibility under various lighting.

- Use different algorithms to adjust the output based on common expected lighting conditions.

Design Approach F

Explores the possibility of using machine learning algorithms to enhance video output. Below are the types of ML algorithms we were considering:

- Filter noise from the input data
- Anti-glare
- Prediction algorithm to interpolate pixel data / update light intensity

Design Approach G

Explores ways to use photo sensors to detect the lighting of the environment and then configure the camera accordingly through:

- The use of light intensity readings from optical sensors
- Normalize light intensity data to a specific range that is easy to view
- Photo sensors that trigger flash / extra light

Design Approach H

Potential for selecting an optimal color format for enhanced display visibility.

- Comparing color formats such as RGB and YUV (YCbCr and YCrCb) for the best output
- Adjust luminous intensity

4.2.3 Decision-Making and Trade-Offs

Decision-Making Factors	Weighting (Scale of 1 to 4)	Options			
		Digital Brightening	Low Light Camera	Light Facing the User	Photo Sensors to Detect Lighting Conditions
Cost	2	4	3	2	1
Ease of Implementation	4	1	3	3	1
Debugging Capability	3	2	4	2	2
Available Documentation	1	1	4	1	2
	Total	**19**	**34**	**23**	**14**

Table 8: Weighted Decision Matrix

The team used the weighted matrix (shown above) to identify potential solutions for outputting video data under different lighting conditions. Of the eight Lotus Blossom design approaches from Section 4.2.2, the team narrowed the ideated methods down to digital brightening, low light camera, light-facing user, and photo sensors for light condition detection.

The options are ranked on a scale from 1 to 4, with 1 being the most impractical and 4 being the best design approach. Design approaches were measured using the following metrics: cost, ease of implementation, debugging capabilities, and documentation availability with a weighting being applied to each metric; a metric with higher weighting was more important to the overall decision-making process for the team. In this case, Ease of implementation was the most important factor, and available documentation was the least important.

The team chose the decision matrix approach for assessing the advantages and disadvantages as it provides an accessible and easy-to-interpret graphical representation of the pros and cons of specific design ideas. In addition, the matrix provides a quantitative methodology for comparing design solutions to relate individual and total advantages and disadvantages of different proposed solutions.

4.3 Final Design

4.3.1 Overview

The team created two video pipeline block diagrams. The first one is for routing data from the TPG to the VDMA in Figure 10 and the second is for the full video pipeline shown in Figure 11. The first pipeline was used to verify the VDMA and DisplayPort connection before integrating the IMX219 camera and MIPI controller for the full pipeline.

4.3.2 Detailed Design and Visual

The first video pipeline, shown below in Figure 10, from the TPG to the VDMA was used for testing the backend modules of the full video pipeline, namely the VDMA and the DisplayPort monitor. The TPG sends a static image to the VDMA. Before sending video data to the monitor, the VDMA writes that data to frame buffers in DDR memory. Finally, the data is read from DDR memory and displayed on the monitor.

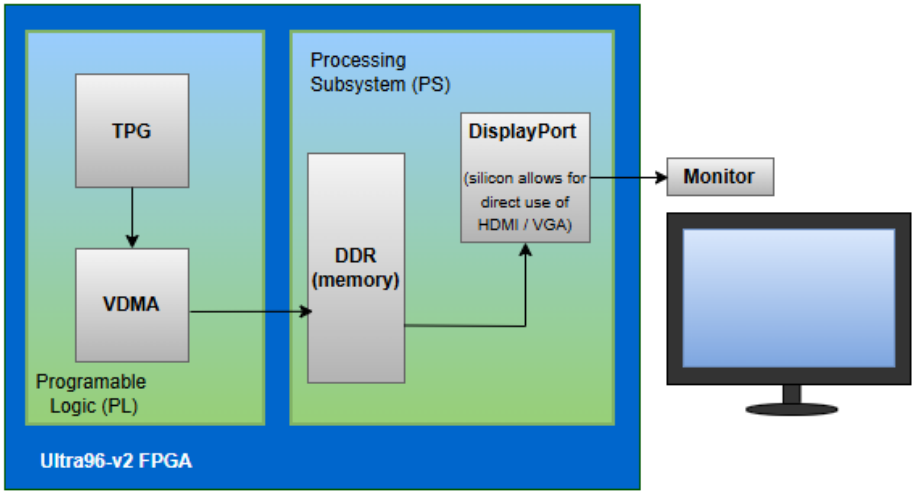


Figure 10: TPG to VDMA Block Diagram

IMX219 to DisplayPort Full Video Pipeline

Once we knew how to properly configure the VDMA and DisplayPort monitor, we worked on designing the full video pipeline, shown below in Figure 11, which includes the IMX219 image sensor and MIPI controller instead of the TPG. Raw video data is captured by the IMX219 and sent to the VDMA via the MIPI controller. The MIPI controller passes the data through the MIPI D-PHY and CSI-2 blocks. The D-PHY acts as the physical communication layer for interfacing between the IMX219 and the Ultra96-v2 FPGA, and the CSI-2 block creates a high-speed camera serial interface link between the IMX219 and VDMA. Once the MIPI controller sends the video data to the VDMA, the VDMA then writes the video data into DDR memory. Finally, the video data is read from DDR memory and output to the DisplayPort monitor.

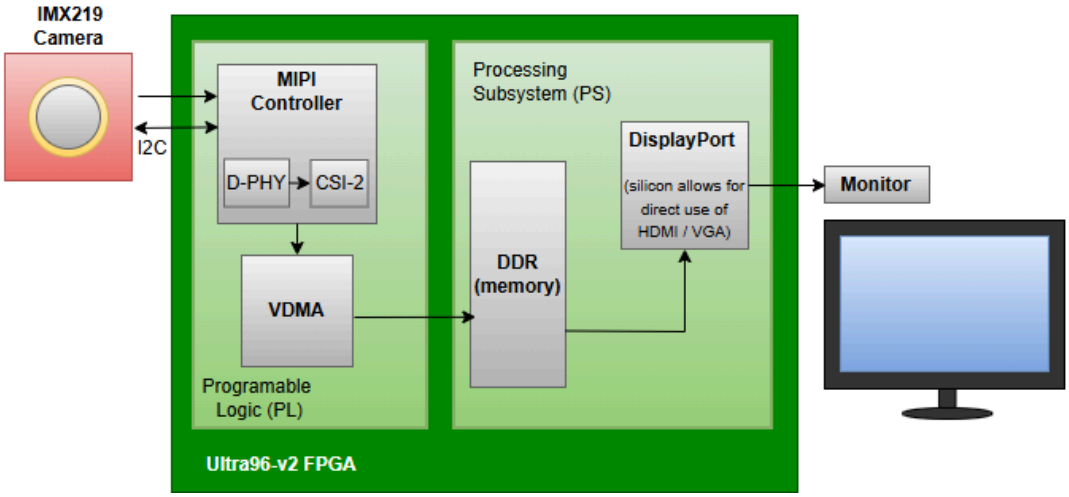


Figure 11: IMX219 Image Sensor to DisplayPort Block Diagram

The image below shows the block diagram of the image sensor (camera) the team will use, called the IMX219 image sensor.

IMX219 Image Sensor

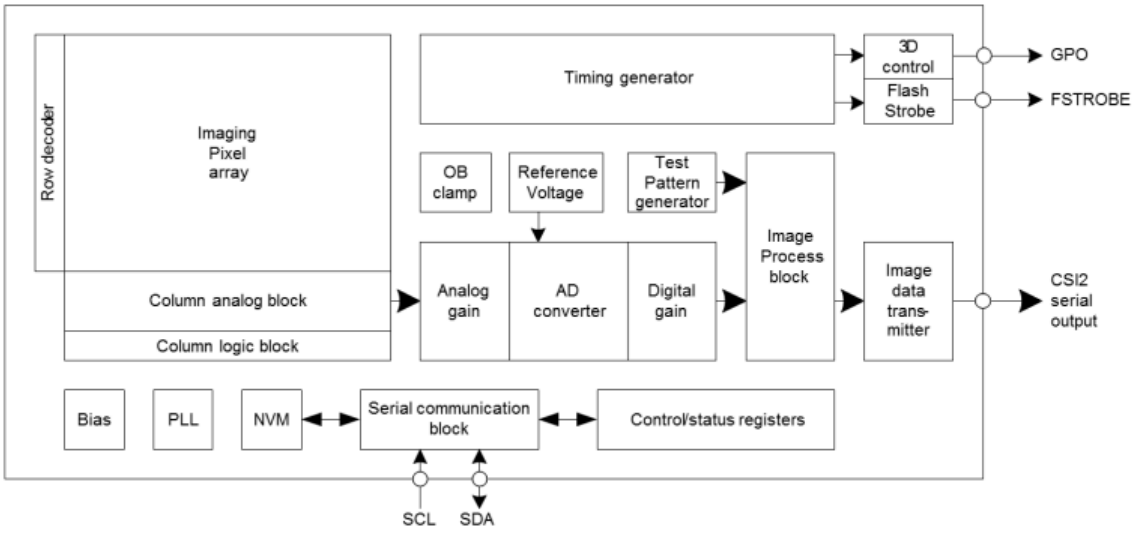


Figure 12: IMX219 Block Diagram

The IMX219 supports digital and analog gain, ADC (analog to digital) conversion, a test pattern generator (TPG), and data transfer. After receiving video stream data as a pixel array, a subset of the pixels are used and output to the MIPI Receive block through MIPI Transmit, shown below.

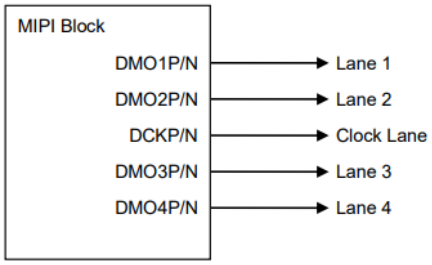


Fig. 4 Relationship between Output pin name and MIPI output Lane

Figure 13: MIPI Transmit block diagram

For the project, the team has discussed using 2 lanes for data transfer in the MIPI Transmit block. This can be set from the following register shown below.

Table 12 Number of CSI lane Setting Registers

Index	Byte	Register Name	RW	Comment	Default (HEX)	Remark
0x0114	[1:0]	CSI_LANE_MODE	RW	03: 4Lane 01: 2Lane	03	Setting before "standby cancel"

Figure 14: Setting Register 0x0114 to 0x01 gives 2-Lane for data output

Finally, shown below is a pixel array frame structure the team has identified for outputting video stream data. The video output, shaded in the yellow rectangle below, shows one example setting at 1920x1080p.

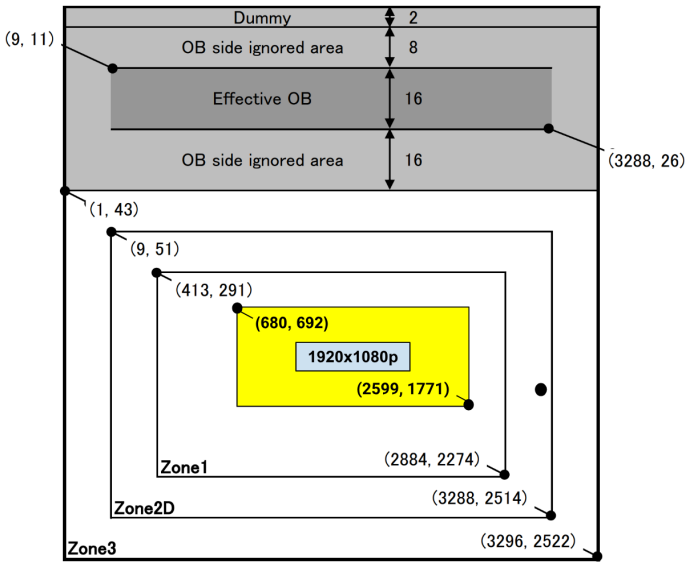


Figure 15: The Frame Structure for 1920x1080p

The figure below shows the path for I2C communication from the ZU3EG ARM processor to the TCA9548A MUX to the IMX219 image sensor. These components are found on the Ultra96-v2 FPGA board or can be set up on the board.

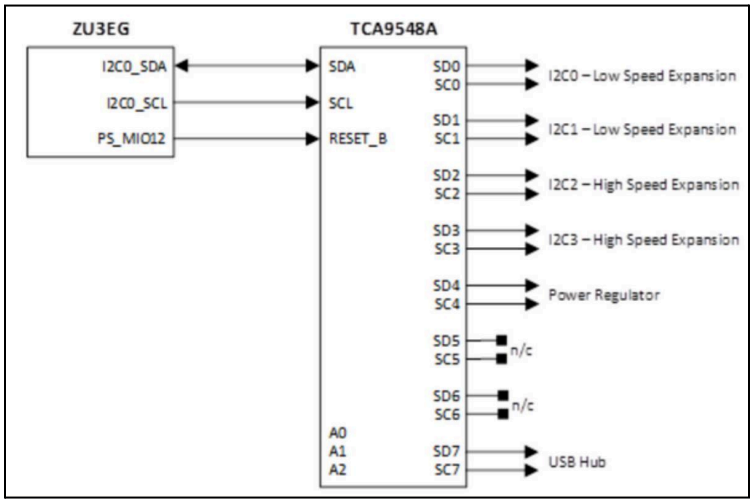


Figure 16: I2C Block Diagram

Left to Right: ZU3EG ARM Processor → TCA9548A I2C MUX on Ultra96-v2 → Select 1 of 8 I2C ports to the image sensor.

The Ultra-96 uses the I2C1 interface within the ARM processing system to connect to an I2C expander, shown below.

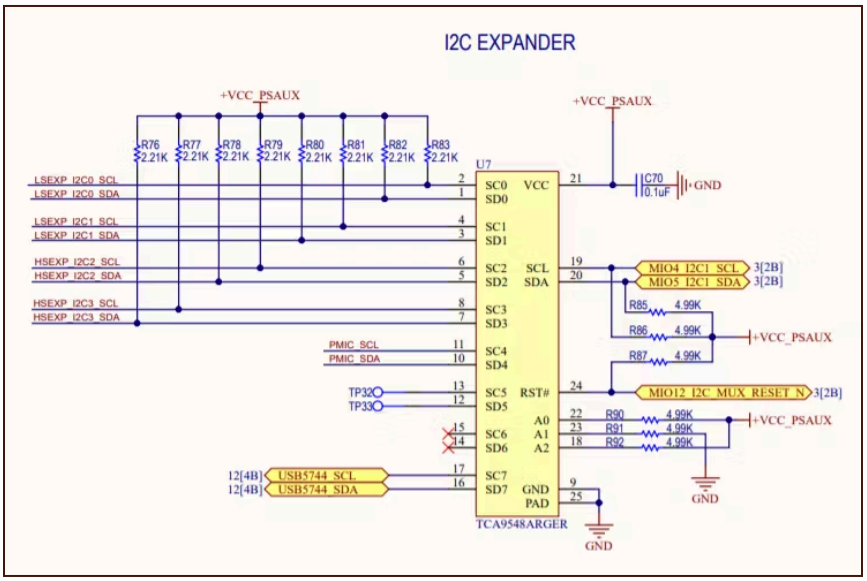


Figure 17: I2C Expander

Channels 0 and 1 connect to the low-speed connector which can be used for I2C. Channels 2 and 3 connect to the high-speed connector that can be used for high-speed serial interface communication (such as CSI-2).

MIPI Controller

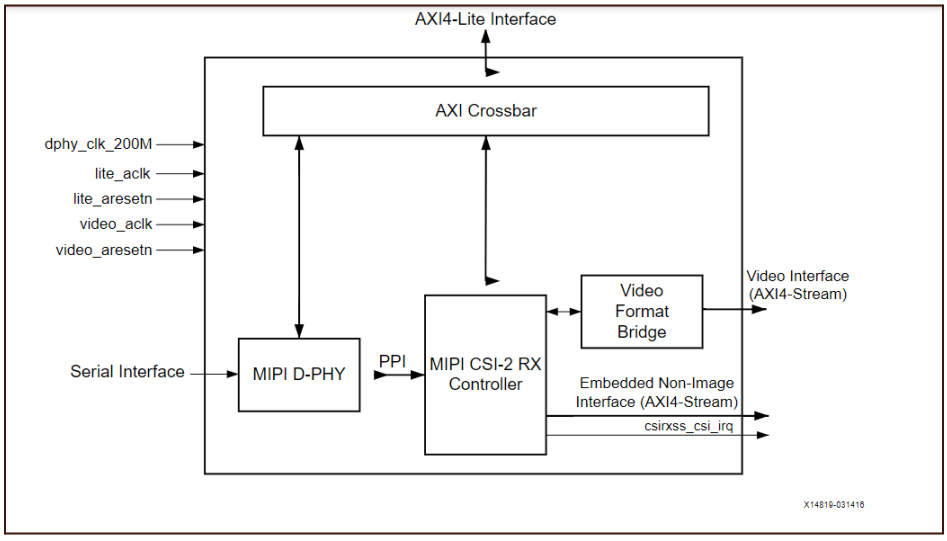


Figure 18: MIPI Subsystem Block Diagram

The MIPI subsystem diagram above interfaces with MIPI-based image sensors and MIPI-based pipelines. The subsystem includes four sub-cores:

- **MIPI D-PHY**
    - The MIPI D-PHY IP core implements a D-PHY RX interface and provides PHY protocol layer support for compatibility with the CSI-2 RX interface. This core is included to provide direct connection support to a MIPI image sensor via the Serial Interface Input. A diagram for the MIPI D-PHY RX core is provided below.

- **MIPI CSI-2 RX Controller**
    - The MIPI CSI-2 RX Controller core includes multiple layers, including lane management, low-level protocol, and byte-to-pixel conversion. The MIPI CSI-2 controller receives 8-bit data per lane for up to four lanes from the D-PHY controller through the PHY Protocol Interface (PPY). A diagram for the MIPI CSI-2 Controller is provided below.

- **Video Format Bridge**
    - The Video Format Bridge core uses the user-selected configuration to filter only the required AXI4-Stream data beats. AXI4-Stream Video is output from the MIPI Controller to the VDMA subsystem.

- **AXI Crossbar**
    - The AXI Crossbar core routes AXI4-Lite requests to the corresponding sub-cores. AXI4-Lite interface is the communication format used to write to and read from registers required for configuration and monitoring of the subsystem.

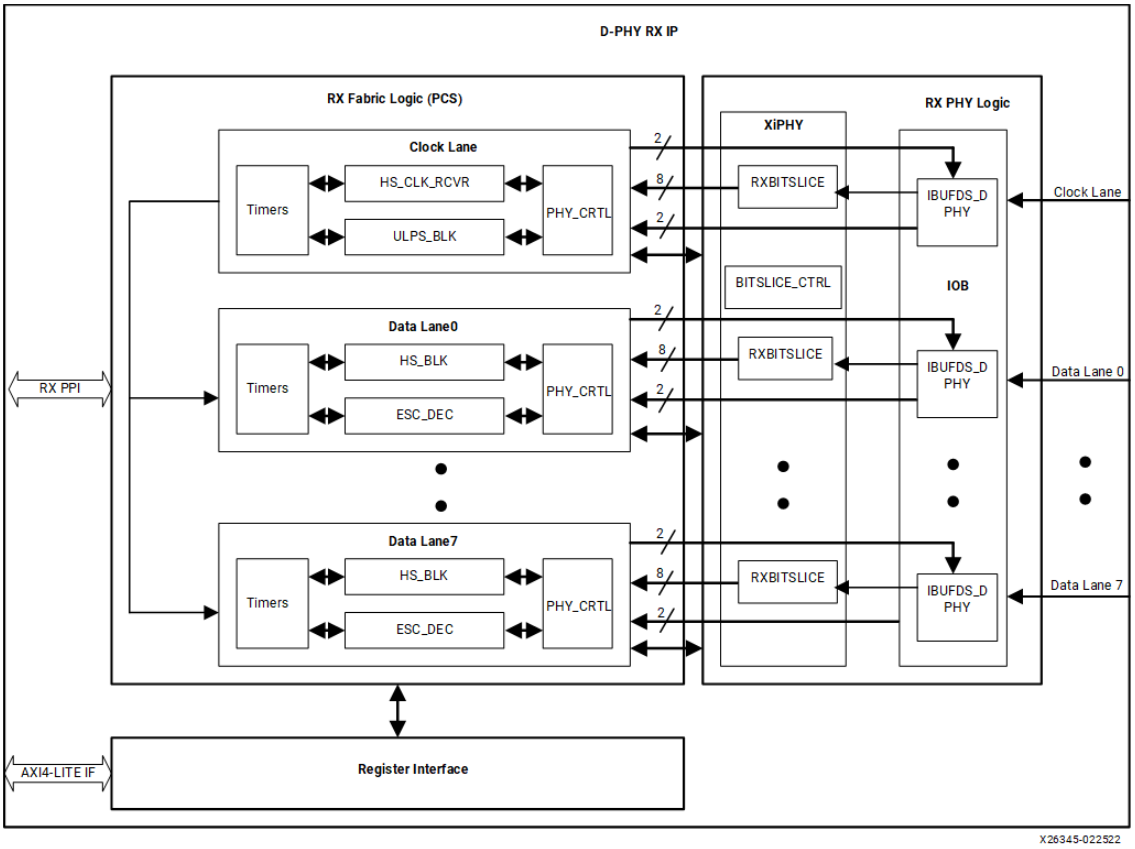


Figure 19: MIPI D-PHY RX Architecture

The MIPI D-PHY RX architecture contains three main components:

- **RX PCS Logic**
  - The RX PCS Logic block interfaces with PHY and delivers PPI-compliant transactions, like High-Speed and Escape mode Low-Power Data Transmission Packets, and is responsible for lane initialization, start-of-transmission detection, and clock recovery in escape mode.
- **RX PHY Logic**
  - The RX PHY Logic block performs clock recovery in high-speed mode and de-serialization.

- **Register Interface**
  - AXI4-Lite interface is used to control and monitor timers and registers.

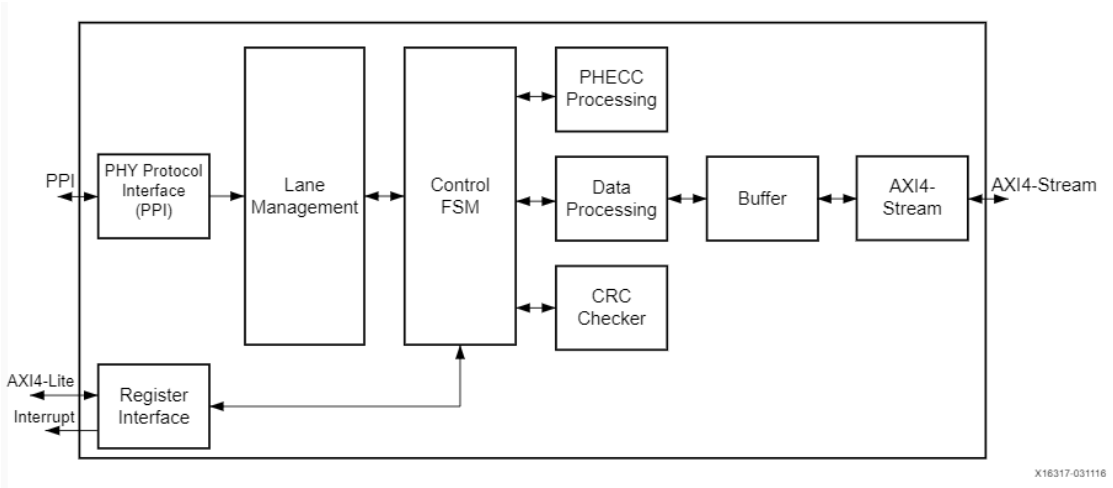


Figure 20: MIPI CSI-2 RX Architecture

The MIPI CSI-2 controller contains components to translate the low-level PPI protocol into a high-level AXI-4 stream available to the user. These components include Lane Management, Control FSM, Data Processing, a buffer, and a Register Interface accessible by AXI4-Lite protocol.

VDMA

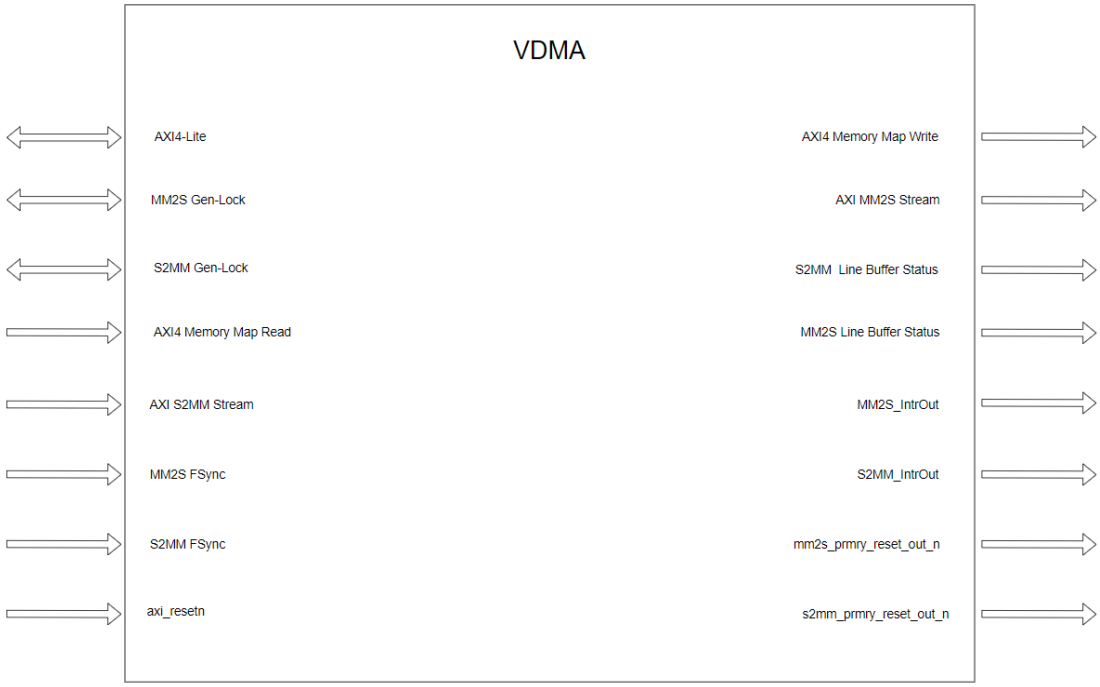


Figure 21: VDMA IP Block Diagram

The block diagrams above show the general I/O of the VDMA component.

DisplayPort

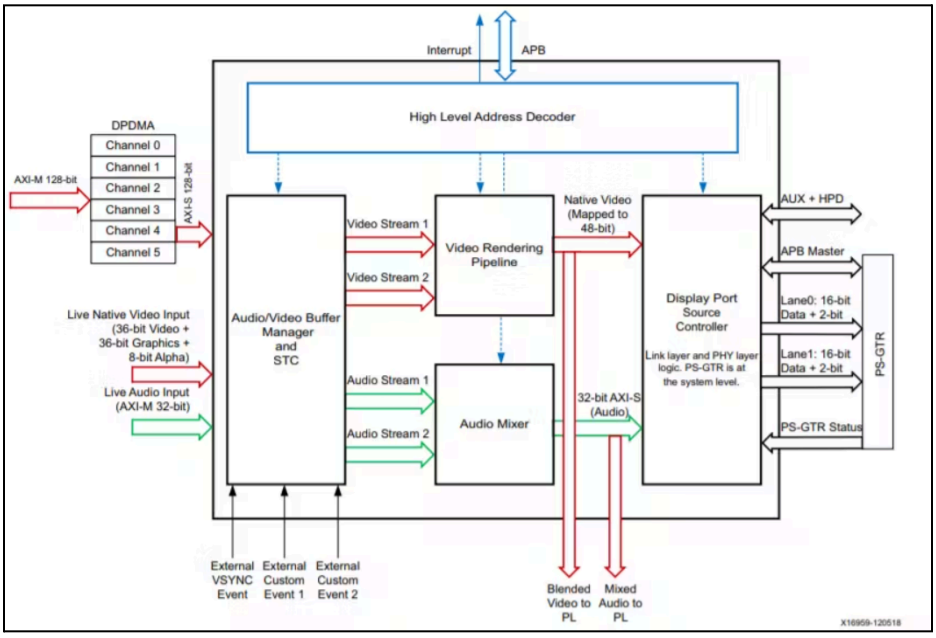


Figure 22: Ultra96-v2 Displayport Controller Architecture

Figure 22 shows the block diagram for the DisplayPort found on the Ultra96-v2 FPGA. The DisplayPort uses silicon, which allows for direct use of HDMI/VGA.

The Ultra96-v2 FPGA board contains a silicon display inside the processing system, which allows for direct use of HDMI. The VDMA is responsible for storing video data frames in the DDR memory within the processing system from either the live feed from the video camera or from a pre-set pattern from the TPG. The VDMA will use a ping-pong method to store at least two frames at a time to allow for both the processing and reading of frames. Then, a software-encoded buffer retrieves frames from the DDR memory and transfers them to the DisplayPort located within the processing system. The video will go directly to the mini display port cable which is connected to a monitor to output the video stream.

Note: that the AXI Interconnect block signifies a connection either to AXI4-Lite which connects the program memory to configuring registers or AXI-Stream which is for video data transfer between IP blocks.

4.3.3 Functionality

The product will be utilized by the following users and in the following real-world contexts:

> **Disabled users** can utilize the product to track their eye movements in real-time for seizure detection and communication and navigation assistance.

Programmers can utilize this product to implement Machine Learning algorithms for different tasks.

- Improving the navigational abilities of individuals restricted to motorized wheelchairs.
- Facial recognition.
- Object detection and avoidance for self-driving vehicles.
- Surveillance technology for use in traffic control or intelligence agency applications.

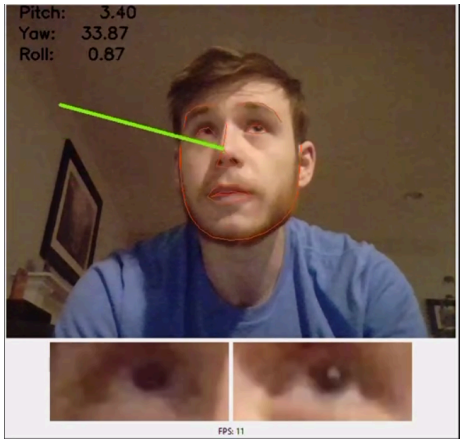


Figure 23: Eye-Tracking Algorithm by a previous Senior Design Team

4.3.4 Areas of Challenge

We faced the following challenges in completing our design with respect to technical progress, satisfying requirements, and meeting user needs:

- Learning how to program in a PYNQ versus a non-PYNQ environment was a new challenge.
    - We overcame learning how to code in PYNQ by reading and understanding the source code for different IP driver libraries, such as for the VDMA.
- Debugging incorrect configurations in the Vivado hardware overlays were not a trivial problem to overcome.
    - We overcame the incorrect outputs due to incorrect Vivado hardware overlay configurations by making observations of the unexpected results, proposing theories to fix what we were noticing, and testing our theories by making a change to the hardware overlay while keeping all other parameters constant.
- The team met with the client and advisor regularly to share progress reports, ask for clarification of potential solutions we were facing at the time, and verify completion of project milestones over time.

4.4 Technology Considerations

The team is using the following technologies. The associated strengths, weaknesses, and tradeoffs are listed below:

- The team will utilize Sony IMX219 mage sensor.
- A programmable Ultra96-v2 FPGA will be used.

Advantages

- Commonly available off-the-shelf hardware components and image sensors.
- The hardware components are cost-effective.
- Documentation is available from previous Senior Design teams and datasheets online.
- The image sensors have programmable exposure time, gain control, black level adjustment, defect correction, etc.

Disadvantages

- FPGA hardware design and software development technical expertise is needed.

5. Testing

5.1 Unit Testing

Individual components were tested as per the project description and requirements design. A unit in this project is one component, for example, the IMX219 image sensor would be one component/unit. Testing the configuration for each component primarily consists of writing to and reading from important register values. This was completed using terminal commands and shell scripts using TeraTerm. For the IMX219, for example, we successfully read from read-only registers with known values and verified that the values matched with the datasheet values for the IMX219.

5.2 Interface Testing

There are no interfaces being developed for this project. The primary way the team is interfacing with the hardware is through terminal commands using TeraTerm and Jupyter Notebook, both of which are developed externally for this project and are thus not tested.

5.3 Integration Testing

Integration testing was completed between connecting components of the system, such as the image sensor connecting to the MIPI controller. This testing was completed by reading appropriate register values using TeraTerm terminal commands. For the IMX219 and MIPI controller interface, for example, we monitored the MIPI D-PHY and CSI-2 registers and verified that the datatype of RAW8 we sent from the IMX219 was observed, the number of packets continuously increased up to 0xFFFF then reset back to 0x0000, and the frame received bit was observed.

5.4 System Testing

System testing was done after all unit tests and integration tests were completed and passed. System testing entailed reading register values to ensure video data was being transmitted effectively throughout the system. This was done by successfully observing the frame received bit at the MIPI controller from the IMX219 and the frame count interrupt being asserted by the VDMA as expected.

In addition, we did visual testing by comparing the video data from the IMX219 on the DisplayPort monitor with a known image that the IMX219 was pointed at to verify the video data outputted as expected. For example, Figure 24 below shows the rainbow image that was used to compare specific colors and image orientation, showing the video pipeline output on the right when the IMX219 is pointed at the original rainbow image on the left.

Figure 24: 640x480p at 20 fps: test image (left), frame from IMX219 (right)

Figure 24 shows that the orientation of the video data and colors match the expected orientation of the letters and colors from left to right and top to bottom, along with the expected colors of the rainbow from red to violet.

5.5 Regression Testing

Regression testing was handled by the unit, integration, and system testing. This was done, for instance, by creating a custom configuration code for the IMX219 to send a desired rectangular bounding box, framerate, width, and height while maintaining integral configuration constant across different resolutions and framerates. The code for configuring the DisplayPort monitor, VDMA, MIPI D-PHY, and MIPI CSI-2, along with most of the IMX219 registers was kept the same regardless of custom configuration. This was necessary to ensure that the video pipeline functions regardless of changes to the IMX219 registers relevant to configuring a desired resolution and framerate.

5.6 Acceptance Testing

Acceptance testing was handled by regular discussions and progress reports with the client and advisor. The progress of the unit, integration, system, and regression testing was shared with the client and advisor over time for feedback and verification of the completion of the project's functional and non-functional requirements.

5.7 User Testing

This project is part of a larger project which is still not fully developed. As such, the larger project is not ready for user testing at this time.

5.8 Results

In addition to the results shown in Figure 24 for 640x480p video data at 20 frames per second from the IMX219, the following patterns from the TPG are given below in Figure 25 at 640x480p. These tests were useful for verifying and debugging the configuration of the VDMA and DisplayPort monitor using the video pipeline from Figure 10.

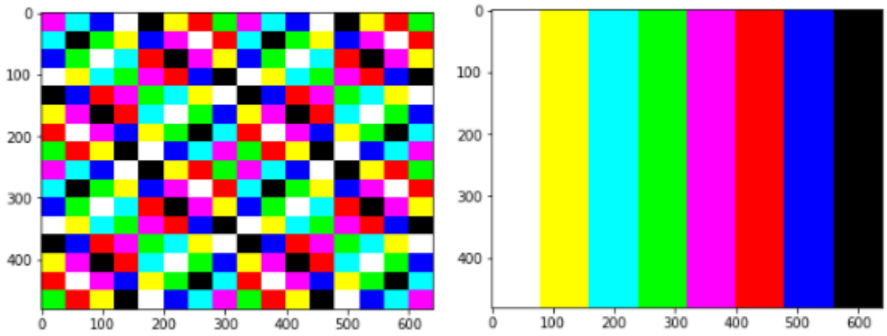


Figure 25: Test Patterns from TPG IP block at 640x480p

The majority of the work in debugging the video pipeline was done on the Vivado hardware overlays, which are used to configure the hardware. Common issues encountered during testing included incorrect video synchronization, timing, and a noticeable shift in colors. Figure 26 below gives an example of a noticeable shift in colors on the left for the TPG pattern on the right.

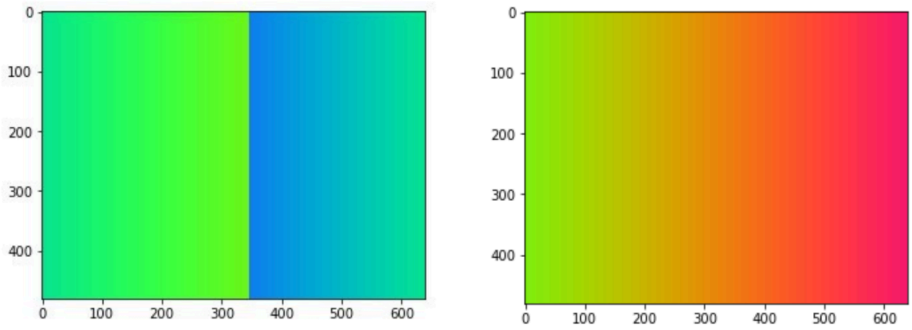


Figure 26: Incorrect noticeable shift TPG (left) and correct TPG 640x480p patterns

Figure 27, below, shows an incorrect TPG video synchronization and timing output on the left and the correct output on the right at 640x480p.

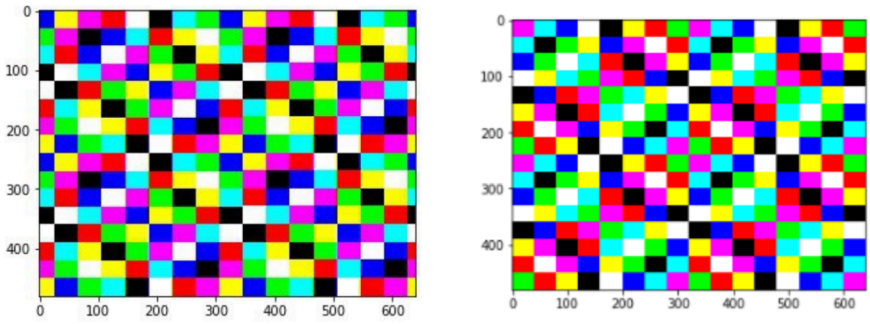


Figure 27: Incorrect video synchronization/timing TPG (left); correct TPG 640x480p patterns

After identifying the above errors from Figures 26 and 27 and similar video synchronization, timing, and noticeable color shifts, we made corrections to the hardware design in Vivado for both the TPG to VDMA block diagram in Figure 10 and the full video pipeline in Figure 11. These changes resulted in the correct expected outputs shown in Figures 24 and 25.

6. Implementation

We successfully built a video pipeline that takes image data from the IMX219 image sensor and outputs it to a DisplayPort monitor. This matches with our final design given in Figure 11. Video data is passed from the IMX219 to the MIPI controller, VDMA, and finally to the monitor; all components have been implemented.

6.1 Design Analysis

The design works as intended. It can output data at 640x480p at 20 fps, thus achieving the objective of 640x480p at 15 fps set by the team. In addition, the team has created a custom

configuration code for the IMX219 to output a desired resolution and frame rate. This code can be used by future teams in conjunction with machine learning algorithms for capturing a user's eye-movements and using this information to help aid the user in navigation.

7. Professional Responsibility

This discussion is with respect to the paper titled "Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment", *International Journal of Engineering Education* Vol. 28, No. 2, pp. 416–424, 2012.

7.1 Areas of Responsibility

A table with the different areas of professional responsibility as defined by NSPE and ACM Canons are given in table 9.

Area of Responsibility	Definition	NSPE Canon	ACM Canon
Work Competence	Perform work of high quality, integrity, timeliness, and professional competence	Perform services only in areas of their competence; Avoid deceptive acts	Strive to achieve high quality in both the processes and products of professional work; Perform work only in areas of competence
Financial Responsibility	Deliver products and services of realizable value and at reasonable costs	Act for each employer or client as faithful agents or trustees	Professionals should be cognizant of any serious negative consequences affecting any stakeholder that may result from poor quality work and should resist inducements to neglect this responsibility.
Communication Honesty	Report work truthfully, without deception, and understandable to stakeholders	Issue public statements only in an objective and truthful manner; Avoid deceptive acts	Be honest and trustworthy; Foster public awareness and understanding of computing, related technologies, and their consequences
Health, Safety, Wellbeing	Minimize risks to safety, health, and well-being of stakeholders	Hold paramount the safety, health, and welfare of the public	Avoid harm; Give comprehensive and thorough evaluations of computer systems and their impacts, including analysis of possible risks
Property Ownership	Respect property, ideas, and information of clients and others	Act for each employer or client as faithful agents or trustees	Respect privacy; Honor confidentiality; Access computing and communication resources only when authorized or when compelled by the public good
Sustainability	Protect environment and natural resources locally and globally	NSPE does not mention sustainability	Contribute to society and to human well-being, acknowledging that all people are stakeholders in computing
Social Responsibility	Produce products and services that benefit society and communities	Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession	Ensure that the public good is the central concern during all professional computing work

Table 9: Area of responsibility analysis

Listed below are explanations for each table entry describing how the ACM code of ethics differs from the NSPE code.

Work Competence

- The ACM code of ethics states that computing professionals should spend the extra time to produce high-quality products by using processes that they are competent with.

Financial Responsibility

- The ACM code of ethics states that computing professionals should be aware of any negative impacts on stakeholders resulting from quality of work.
- The ACM code of ethics states that computing professionals should keep client financial information confidential.

Communication Honesty

- The ACM code of ethics states that computing professionals should avoid dishonesty and information misdirection by promoting public understanding of the potential negative consequences of computing technologies.

Health, Safety, Wellbeing

- The ACM code of ethics states that computing professionals should minimize, justify, and/or report any risk that may lead to harm or injury.

Property Ownership

- The ACM code of ethics states that computing professionals should only use relevant information from a client.
- The ACM code of ethics states that computing professionals should only access client / others property (including ideas and information) when authorized or in the interest of the public good.

Sustainability

- The ACM code of ethics states that computing professionals should avoid harm to the environment and carefully consider how certain choices could lead to damages.

Social Responsibility

- The ACM code of ethics states that computing professionals should balance multiple group interests with a focus on the needs of less advantaged groups.
- The ACM code of ethics states that computing professionals should aim to meet a broad range of social needs.

7.2 Project-Specific Professional Responsibility Areas

Below is a table examining our team's performance thus far as it relates to the seven areas of professional responsibility.

Area of Responsibility	Professional Context	Team Performance
Work Competence	Our team has worked well together with each team member taking responsibility for their roles, keeping open lines of communication, and asking for assistance as needed to ensure the team delivers quality work.	High
Financial Responsibility	The project aims to use commercially available components as a means to keep costs down.	High
Communication Honesty	Transparency between team members is important in understanding project expectations as well as helping to ensure the team can meet deadlines.	High
Health, Safety, Wellbeing	This is the team's primary focus, as the design intends to reduce the overall health and safety risks for individuals with illnesses and/or disabilities.	High
Property Ownership	Our design requires a fair amount of research to understand how to configure the hardware and software. We will cite the information that we use.	Medium
Sustainability	The team is cognizant of the impact our design may have on the environment based on what materials are used and the amount of power required to run the design.	Medium
Social Responsibility	The team is aiming to increase the independence of users with physical disabilities.	High

Table 10: Area of responsibility team reflection

7.3 Most Applicable Professional Responsibility Area

The most relevant broader context issue that relates to our project is public health, safety, and well-being. Our video pipeline aims to improve the quality of life of individuals with physical disabilities by utilizing various machine learning programs, which can act as a navigational aid,

emotion monitor, or seizure detector. By focusing on meeting the hardware requirements of these programs, we can better serve the user.

8. Conclusions

8.1 Summary of Progress

Our project successfully completed the creation of a prototype video pipeline that has a base configuration of 640x480p at 20fps. The code for this was done using Python in a Jupyter Notebook. The current code also contains a custom IMX219 image sensor configuration that allows future design teams to use the video pipeline to focus on a specific region of a frame, such as the eyes for eye-tracking.

8.2 Value Provided

Overall, the design broke significant ground in the broader context of this project. The pipeline is successfully able to send video data through for a custom size and frame rate, which can be updated as the user needs. The testing framework implemented through Jupyter Notebook and PYNQ commands allows future teams to easily update and test the design as needed. Throughout the course of this project, the team has created a large amount of documentation on the individual components as well as code and testing processes that will greatly help for the further development of this project in future steps. For this purpose, we met with the next team and shared our code with them to help them understand and potentially build on our work and implement machine learning algorithms such as for eye-tracking.

8.3 Next Steps

This project is part of a larger project to create a navigational tool for people confined to motorized wheelchairs. There are a variety of future steps needed before that vision can be realized. The first of those steps will be to configure a machine-learning eye-tracking algorithm to run on the designed pipeline. The eye-tracking algorithm is being developed by a different team, and the pipeline and algorithm will need to be updated to work together. A type of application will be developed to control the pipeline and customize the video data depending on variables such as user orientation, eye location, processing speed, etc. Before this process, the initialization and configuration scripts for the pipeline will need to be written in a faster, more efficient programming language, like C. This is because the application will be able to request a new configuration multiple times a second, and the current PYNQ implementation will not be fast enough to handle that. Another team will also need to construct a container to house this pipeline, as well as a mount that will attach this to a motorized wheelchair.

9. References

[1] T. Mitsuhashi, M. Sonoda, H. Iwaki, K. Sakakura, and E. Asano, “Detection of absence seizures using a glasses-type eye tracker,” *Clinical Neurophysiology*, vol. 132, no. 3, pp. 720–722, Mar. 2021, doi: https://doi.org/10.1016/j.clinph.2020.12.015.

[2] J. Z. Lim, J. Mountstephens, and J. Teo, “Emotion Recognition Using Eye-Tracking: Taxonomy, Review and Current Challenges,” *Sensors*, vol. 20, no. 8, p. 2384, Apr. 2020, doi: https://doi.org/10.3390/s20082384.

[3] “Low code computer vision,” EyesOnIt, https://www.eyesonit.us/ (accessed Apr. 16, 2024).

[4] “Meet Luci: Luci Mobility,” LUCI, https://luci.com/meet-luci/ (accessed Apr. 16, 2024).

[5] “MIPI CSI-2 Receiver Subsystem LogiCORE IP Product Guide (PG232),” AMD Technical Information Portal, https://docs.amd.com/r/en-US/pg232-mipi-csi2-rx/MIPI-CSI-2-RX-Controller (accessed Apr. 16, 2024).

[6] “Machine Vision FPGA Computer vision - intel® FPGA,” Intel, https://www.intel.com/content/www/us/en/industrial-automation/products/programmable/applications/machine-vision.html (accessed Apr. 16, 2024).

[7] Sony Corporation, "IMX219PQ Exmor R™ CMOS Image Sensor," Sony Corporation, April 2016. Available: https://www.opensourceinstruments.com/Electronics/Data/IMX219PQ.pdf (accessed: April 16, 2024).

[8] SparkFun Electronics, "OV5647 Datasheet," SparkFun Electronics, April 2013. Available: https://cdn.sparkfun.com/datasheets/Dev/RaspberryPi/ov5647_full.pdf (accessed: April 16, 2024).

[9] OmniVision Technologies, "OmniVision Technologies Serial Camera Control Bus (SCCB) Specification," OmniVision Technologies, March 2006. Available: https://www.waveshare.com/w/upload/1/14/OmniVision_Technologies_Seril_Camera_Control_Bus%28SCCB%29_Specification.pdf (accessed: April 16, 2024).

[10] “MicroZed Chronicles: Ultra96, PYNQ, Click Mezzanine, SPI and I2C,” *Hackster.io*. https://www.hackster.io/news/microzed-chronicles-ultra96-pynq-click-mezzanine-spi-and-i2c-ec6186496e00 (accessed Apr. 16, 2024).

[11] “PYNQ Introduction — Python productivity for Zynq (Pynq),” *pynq.readthedocs.io*. https://pynq.readthedocs.io/en/latest/ (accessed Apr. 16, 2024).

10. Appendices

10.1 Appendix 1 - Operation Manual

To send video data in the correct orientation from top to bottom and left to right, the data must first be flipped vertically and horizontally. After this, the top left and right and bottom left and right corners of the IMX219 are labeled in Figure 29 below, where TL stands for top left, TR stands for top right, BL stands for bottom left, and BR stands for bottom right.

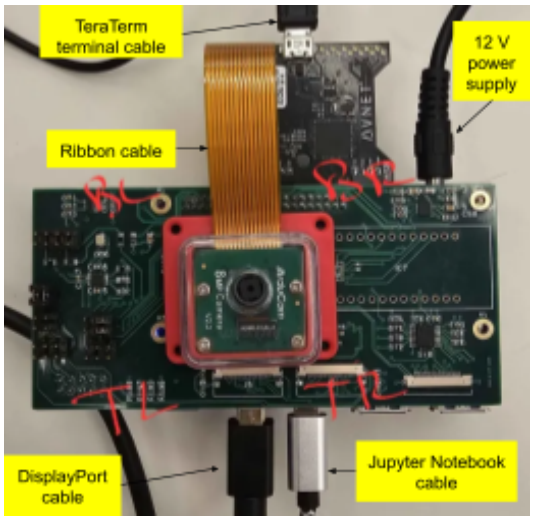


Figure 28: Orientation of the IMX219 image sensor after vertical and horizontal flipping

There are four cables used to set up the Ultra96-v2 FPGA and IMX219 image sensor, all shown in Figure 8. These include the 12 V power supply in the top right of the image, a micro-USB cable to connect to the TeraTerm terminal to the left of the power supply cable, a mini-DisplayPort cable on the bottom of the image to the left, and a micro-USB cable to connect to the Jupyter Notebook environment to the right of the mini-DisplayPort cable. In addition to these cables, the IMX219 must be connected to the MIPI connectors via its gold-colored ribbon cable. It is important to note that the IMX219 ribbon cable must be connected to the gold side facing up.

10.2 Appendix 2 - Alternate/Initial Version of Design

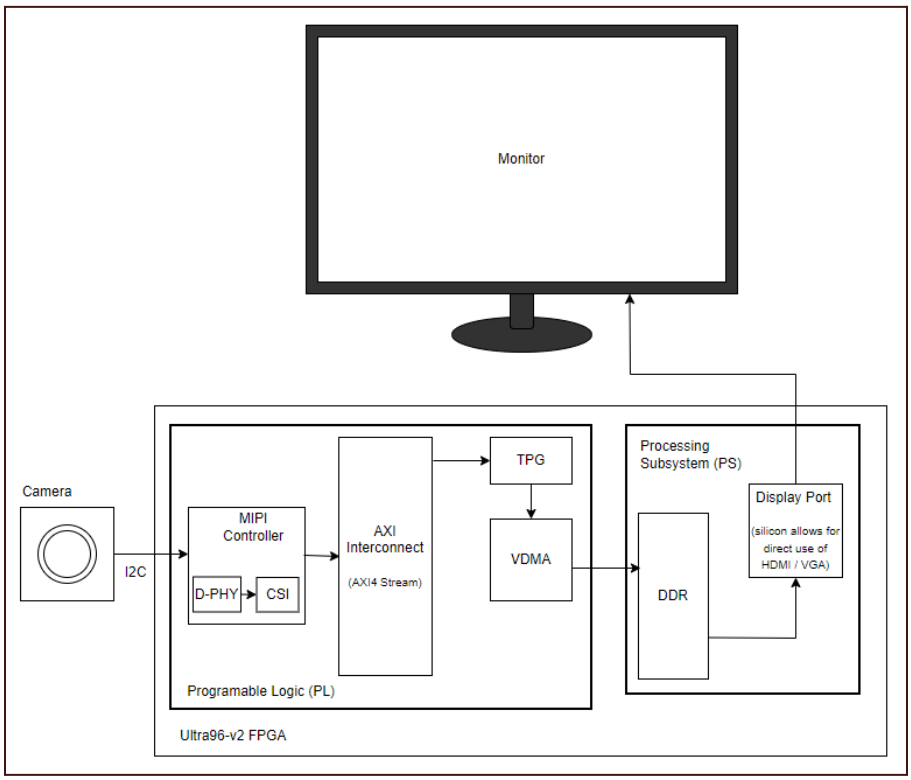


Figure 29: Original Video Pipeline Design

The figure above shows the original design and datapath for the video data. This design includes the TPG IP block as that block can be configured to directly pass video data. The idea was to be able to use the TPG to debug the VDMA and DisplayPort. After confirming that the TPG to DisplayPort was working, the TPG block could be kept and configured to pass data directly while focusing on the proper configuration of the camera and MIPI blocks. This approach was so that the video pipeline could be built modularly. However, based on the feedback we received for our faculty presentation, the inclusion of the TPG in the full video pipeline is unnecessary as the video data could be passed directly from the MIPI block to the VDMA for a more efficient data handoff. Therefore, the design was broken into two separate pipelines (see Figures 10 and 11).

10.3 Appendix 3 - Other Considerations

Throughout the design process, we have learned how to break down a relatively complicated engineering problem and handle it in smaller, more manageable chunks. We have learned how to think of potential issues in our design and come up with high-probability theories to test and either prove or disprove our initial hypotheses to overcome bottlenecks in our design. We used this method of thinking extensively, for example, during our testing process for sending a TPG

image to the DisplayPort monitor. During this challenge, we learned to record any unexpected results we were observing during our debug process, such as those seen in Figures 6 and 7, and then come up with theories of ways to change either our software code or the hardware design in Vivado to accomplish our project objectives and send video data to the monitor.

10.5 Appendix 4 - Team

Team Members

- Liam Janda
- Taylor Johnson
- Ritwesh Kumar
- Deniz Tazegul

Required Skill sets for your project

The following are the skills necessary to complete our project, skills required include:

- Programming knowledge is needed to write the software to interface with the hardware on the Ultra96-v2 FPGA (Shell, C, and Python will be used).
- Ability to operate an FPGA board as our project will execute on an Ultra96-v2 FPGA board.
- An understanding of digital logic circuits to be able to map the data flow through the video pipeline.
- Experience reading datasheets and polling registers through software for configuration and debugging of the MIPI controller, TPG, and VDMA IPs, etc.
- Having knowledge of or the ability to learn how to execute code in a Linux OS; the project will run on a Linux image.
- Hardware/Software debugging and troubleshooting capabilities.

Skill Sets covered by the Team

- All team members have programming knowledge.
- All team members have experience operating an FPGA.
- All team members have at least a basic understanding of digital logic circuits.
- All team members have experience reading datasheets and configuring registers through software
- Liam, Deniz, and Taylor have knowledge of code execution in a Linux OS.
- All team members have experience debugging and troubleshooting hardware and software.

Project Management Style Adopted by the team

Our team has adopted a Waterfall-Agile Hybrid project management style.

Individual Project Management Roles

Team Members/Roles:

- Ritwesh Kumar - Image sensor to MIPI Controller
- Deniz Tazegul - MIPI Controller to VDMA
- Liam Janda - VDMA to DDRM
- Taylor Johnson - DDRM to Output Display

Team Contract

Team Name: V-PIPE

Team Members:

1) Liam Janda 2) Taylor Johnson 3) Ritwesh Kumar 4) Deniz Tazegul

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings:
    - Mondays from noon to 1, face-to-face in a study room. (Library or Student Innovation Center)
2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):
    - Discord and Telegram
3. Decision-making policy (e.g., consensus, majority vote):
    - Discussion and then consensus, including the client.
4. Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):
    - Meeting minutes will include who was in attendance, notes from the meeting, and action items for the coming week. Meeting minutes will be archived in a shared Google Drive.

Participation Expectations

1. Expected individual attendance, punctuality, and participation at all team meetings:
    - Individuals are expected to show up to weekly meetings with the advisor, client, and team members. If someone is unable to attend, please communicate ahead of time.

2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
    - Team members are expected to complete their work as well as they are able and should reach out to the team/client/advisor if they are unable to complete their work on time. This is a learning experience for all of us, so no one is expected to be perfect.

3. Expected level of communication with other team members:
    - Team members are expected to respond promptly; it can be concerning if 48 hours have passed and have yet to receive a response. Team members are expected to update the team regarding work completion or attendance if they can't finish work, request assistance, or attend meetings.

4. Expected level of commitment to team decisions and tasks:
    - Everyone is expected to take part in team decisions, if someone cannot attend a meeting in which a decision is to be made, their input will be taken remotely, and they will be included in the decision-making.

Leadership

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):
    - These are the roles that are expected from our client, it is too early to finalize which team member will be assigned to which role at this time.
    - There is also a **Meeting Leader** for the meeting with our client each week; they are responsible for updating team milestones and team status, arranging the meeting invitation, and choosing the next meeting leader.

2. Strategies for supporting and guiding the work of all team members:
    - Discussions will be held with the team and client to collaborate on the development of the project and make sure all our work is still compatible with each other. To ensure a general knowledge of each aspect of the project, we will have "secondary experts" in each of the leadership roles.

3. Strategies for recognizing the contributions of all team members:
    - With the separation of leadership roles, we will be able to recognize the work done by each team member. For example, the schematic owner will be responsible for schematic changes, thus it will be easy to attribute a schematic change to the schematic owner.

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.

Liam

I have a decent amount of experience with VHDL and FPGA boards and designing schematics for them. I have some experience in circuit building and image processing from previous EE classes. I have used microcontrollers to control various hardware components by reading datasheets and interpreting data.

Taylor

I have experience with some aspects of digital logic circuits and some hardware design. I have exposure to the Xilinx FPGA hardware and software through CprE 488, which I am currently taking. I have general knowledge of circuit design from EE230. I have coding experience with C, VHDL, and Python.

Ritwesh

I have experience having taken or am currently in numerous EE signals & systems classes (224, 324, 424) and a few CPRE digital logic and embedded systems courses (281 and 288). I have completed a few final projects working in teams between 2-4 in EE 230, CPRE 281, and CPRE 288 that involved building circuits on a circuit board, using Verilog to design and implement a stopwatch counter, and learning C to create a medical robot delivery system to meet our group's design requirements. I am currently taking a Machine Learning class (EE 428X) that may help me understand this project more.

Deniz

I have decent experience with hardware design using VHDL and Verilog, more focused on the CPRE logic than electrical circuit design. I have some experience with signals and image processing from classes (EE 224 and CPRE 288). I have more experience with software design and development processes than hardware design due to my job and past classes.

2. Strategies for encouraging and supporting contributions and ideas from all team members:

- There will be plenty of opportunities in team meetings and meetings with the client for team members to bring up questions and ask for input from the team. Component choice will require discussion from the whole team and client before a final decision.

3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)

    - The team member is responsible for bringing up issues they have with their ability to contribute, and the team is responsible for maintaining a healthy environment where everyone is comfortable sharing their concerns and opinions.

Goal-Setting, Planning, and Execution

1. Team goals for this semester (This is the initial roadmap from the client for this project):

    - Identify Requirements, Create a block diagram
    - Review existing schematic from prior team
    - Identify Components and alternative parts
    - Learn how to use the CAD tool for schematic entry and footprint creation
    - Learn the technologies involved - create deep-dive technical slide decks
    - Develop Schematic
    - Give presentation
    - Stretch goal: some PCB layout, if time and interest exists

2. Strategies for planning and assigning individual and teamwork:

    - Tasks will be assigned through the lens of our leadership roles with the help of our client.
3. Strategies for keeping on task:

    - Weekly meetings with the advisor, client, and team members to update progress for the week.

Consequences for Not Adhering to Team Contract

1. How will you handle infractions of any of the obligations of this team contract?

    - There will be a discussion with the team about that specific infraction and what to do to avoid that in the future.

2. What will your team do if the infractions continue?

- If infractions continue, we will bring in the course instructors and client to see how to move forward, depending on the severity of the infraction.

****

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*
b) *I understand that I am obligated to abide by these terms and conditions.*
c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*

1) Liam Janda DATE 1/29/2024
2) Taylor Johnson DATE 1/29/2024
3) Ritwesh Kumar DATE 1/29/2024
4) Deniz Tazegul DATE 1/29/2024

10.6 Appendix 5 - Empathy Map And User Personas

Empathy Map

Hears	Sees
• "There are certain tasks that you can't do" (Ethyl) • Heard about how a new device can save time by automating tasks (Nurse Jackie & Elon) • Favorable reviews of ChatGPT, AI, and machine learning (Elon) • AI is the future (Nurse Jackie & Elon) • Machine learning for medical imaging and identification is currently in development (Nurse Jackie & Elon)	• They often need assistance with accomplishing tasks requiring precise movement (Ethyl & Nurse Jackie) • Sees lots of inefficiencies that machine learning can fix (Nurse Jackie & Elon) • Maybe they have difficulty seeing and need a computer to help out (Nurse Jackie) • Sees a way to help people in need with emerging technologies (Nurse Jackie & Elon)
Says & Does	**Thinks & Feels**
• "I want to be able to color inside the lines" (Ethyl) • Likes to use cutting-edge medical technology (Elon & Nurse Jackie) • "Why spend 5 minutes doing something manually when you can spend 8 hours automating it" (Elon) • Look for ways that new technology can help people in need (Elon & Nurse Jackie) • Likes to experiment with brand-new technologies (Elon & Nurse Jackie) • Enjoys talking about ML with others (Elon)	• Wishes they didn't always need as much help (Ethyl) • Wishes they could better communicate their needs (Ethyl) • Tired of wasting time when computers can do the job instead (Elon) • Computers may be able to better monitor for small patterns that could be overlooked (Nurse Jackie & Elon) • Believes that software may be more interesting than hardware (Elon) • Machine learning and AI are the future (Nurse Jackie & Elon)
Pain	**Gain**
• Medical problems limit the ability to perform daily tasks (Ethyl) • Seizures can occur and medical staff need to be notified ASAP (Ethyl & Nurse Jackie) • The camera needs proper lighting to produce meaningful data (Elon, Nurse Jackie, Ethyl) • Requires a lot of research and fine-tuning (Elon)	• The camera allows for tracking of an individual's eye movements that can be used to communicate medical needs to caretakers (Ethyl & Nurse Jackie) • A sense of satisfaction from using the product and learning something new (Nurse Jackie & Elon) • Automation of tasks frees up time (Nurse Jackie & Elon) • The assistance of a camera better monitors specific patient needs (Ethyl & Nurse Jackie)

Table 11: Empathy Map

User Personas

Ethyl (a disabled user)

- Has a disability and is wheelchair bound
- Describe the user: has poor fine and gross motor skills, is confined to a wheelchair, has difficulty performing daily tasks by themself, occasionally gets seizures
- Characteristics: exhibits interest in playing simple board games, learning new concepts, enjoys art
- Needs: assistance with accomplishing fine motor tasks, to be monitored for seizures
- How would they benefit: able to monitor for certain health-related issues such as seizures, could also learn specific patterns for an individual's communication needs
- How benefits relate to the problem statement: The camera tracks her eye movements in real-time for quick assistance with the onset of a seizure, and the camera tracks eye movements to relay her communication needs to her caretaker.

Elon (an ML enthusiast)

- ML programmer coding for disabled users
- Describe user: is interested in implementing ML algorithms for video pipelines to solve tasks that can benefit disabled people and perform their day-to-day tasks more easily.
- Characteristics: enjoys computer programming and finding practical solutions to everyday challenges.
- Needs: a device that easily processes video data for machine learning applications because he wants to create an innovative product.
- How would they benefit: this user could derive value by exploring their interests in machine learning and seeing their ML design implementations positively affect disabled users with everyday communication and navigation tasks.
- How benefits relate to the problem statement: ML enthusiast can explore their interest in machine learning and help disabled users at the same time.

Nurse Jackie (a caretaker for disabled individuals)

- Describe user: someone who is passionate about taking care of disabled individuals and is looking for more effective methods to complete their work that can improve their productivity and satisfy the needs of their patients.

- Characteristics: may not be very tech-savvy or up to date with recent technological trends (like ML) and is open-minded to new ways of helping their patients.
- Needs: a more effective and efficient way of taking care of their patients that is easy to learn so that they can have more time to take care of multiple patients at the same time.
- How would they benefit: if an ML-enabled eye-tracking device were readily available and easy to learn to use, this user could take care of many patients at the same time and get notified in the case of emergencies such as the onset of a seizure episode to take the right actions as quickly as possible to remedy the situation.
- How benefits relate to the problem statement: this user could benefit from the ML-enabled eye-tracking technology to potentially work with many more patients simultaneously and be able to take care of emergencies in real-time. Taken together, these benefits may boost their productivity and allow them to more effectively take care of their patients.