Lightning Talks

Week 6: Contextualization/Design Check-In

Team Information

- **Project ID:** ssddec24-proj006
- [Senior Design Website]
- **Team members:**
    - Deniz Tazegul
    - Liam Janda
    - Taylor Johnson
    - Ritwesh Kumar
- **Client:** JR Spidell
- **Faculty Advisor**
    - Dr. Phillip Jones

V-PIPE
INNOVATE AT IOWA STATE

Project Overview

- Developing a FPGA-based video pipeline
  - MIPI-connected "off the shelf" camera module
  - Video monitor
- Augmented video → active displayport cable → monitor
- Software executes in Linux OS
- STRETCH GOAL: Pass video through a machine learning algorithm

IMX219 Image Sensor

User for Journey Map: Ethyl

Ethyl

Characteristics

- Has a disability & wheelchair-bound
- Poor fine & gross motor skills
- Experiences seizures

Hears

- “There are certain tasks that you can’t do”

Says

- “I want to be able to color inside the lines”

Thinks & Feels

- Desires independence
- Ability to communicate needs

Pain

- Limited ability to perform daily tasks

Artifact #1: Journey Map

Key Experiences	1: User uses a motorized wheelchair, has difficulty completing daily tasks	2: Received our product and begins researching what they can do with it	3: Begin Configuring system to work for that individual set up	4: Start testing the capabilities	5: Can use this product in everyday life
Actions	She works with a family member to see what options might be available to help her	Begins working with the product for the first time and looking online for similar products to learn how to use it	Attach the video camera to her wheelchair with a ML algorithm that monitors her for seizure	Uses the product with multiple ML algorithms to also help monitor her communication needs and navigational needs	Becomes fully independent and confident in using the product to assist her in with her daily navigational, communication, and health needs (e.g. seizure detection)
Feelings					

Market Research #1: Intel

- Goliath 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

Market Research #2: LUCI

- Software/hardware
  - Environment hazard detection
  - For electric wheelchairs
- For people with physical disabilities

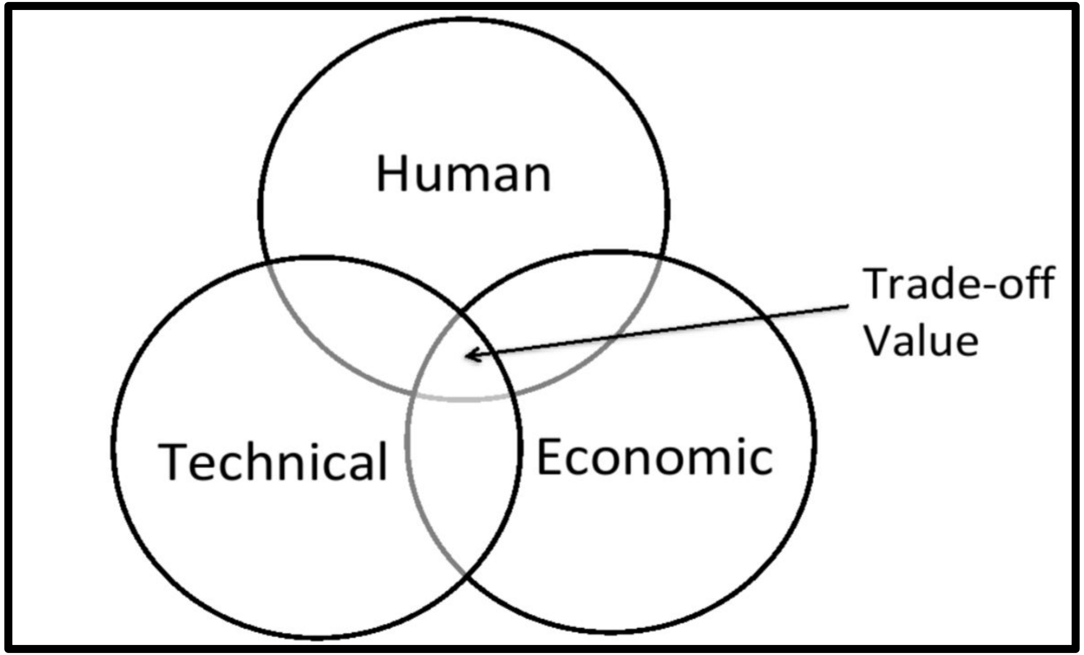
Market Research #3: EyesOnIt

- Custom configuration
  - No programming knowledge needed
- Integratable into surveillance systems

Artifact #2: Pros/Cons Table

	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	Duplicating the technology may be relatively easy as it isn't too specialized

Design Trade-Offs

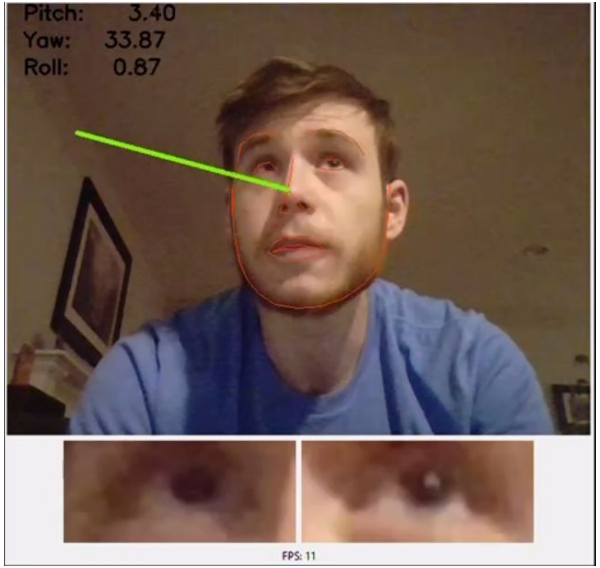


Solution Suitability Perspectives

Human perspective

Addressing user needs

- Design choices have not been tested yet, however they should work
  - Previous teams implementation
  - Updated image sensor to work in a variety of different lighting conditions
  - Open source components



Eye-Tracking Algorithm by a previous SD Team

Human perspective (continued)

Potential project modifications

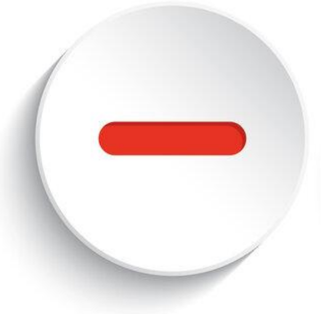
- Once the current design is working add more features
  - Combine with senior design team 5 to test functionality
  - Video pre-processing

Economic perspective

How project improves on existing designs

- Hardware components
  - Commonly available
  - Cost-effective

Drawbacks



- Custom component configuration
- Requires hardware design technical expertise

Economic perspective (continued)

Potential mitigation strategies

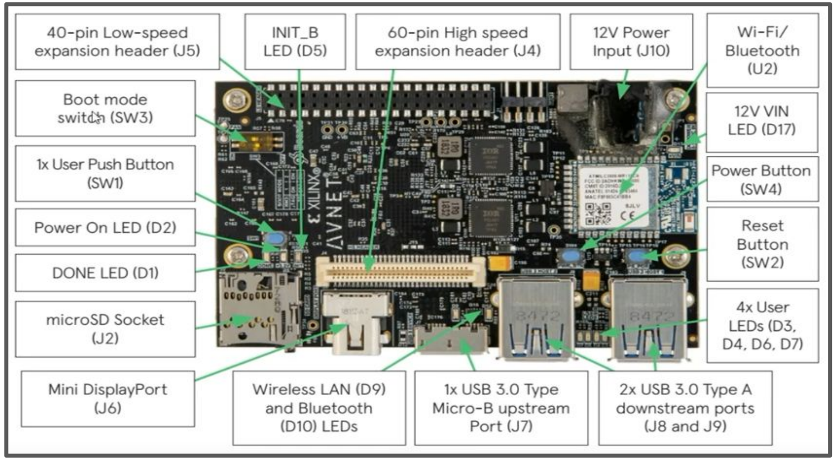
- Integrate PYNQ & non-PYNQ environments
  - PYNQ runs on a Jupyter Notebook server
  - Non-PYNQ runs on a terminal
- **Vivado** custom hardware overlays
- **ILA (Integrated Logic Analyzer)** capture and analyze signals (I/O)

Economic perspective (continued)

Potential mitigation strategies

- Can’t compete with the efficiency of Intel
    - But someone could build this system themselves
- EyesOnIt uses natural language processing
    - We could technically implement this

Technical perspective



Anatomy of Ultra-96 FPGA

External project complexity

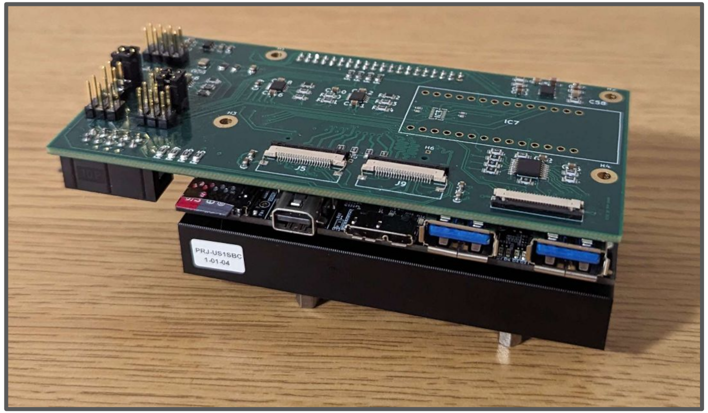
- Using state-of-the-art technology
  - Ultra-96 FPGA created in 2019
  - Image sensors created 10-15 years ago

Internal project complexity

- Multiple hardware & software components
  & subsystems

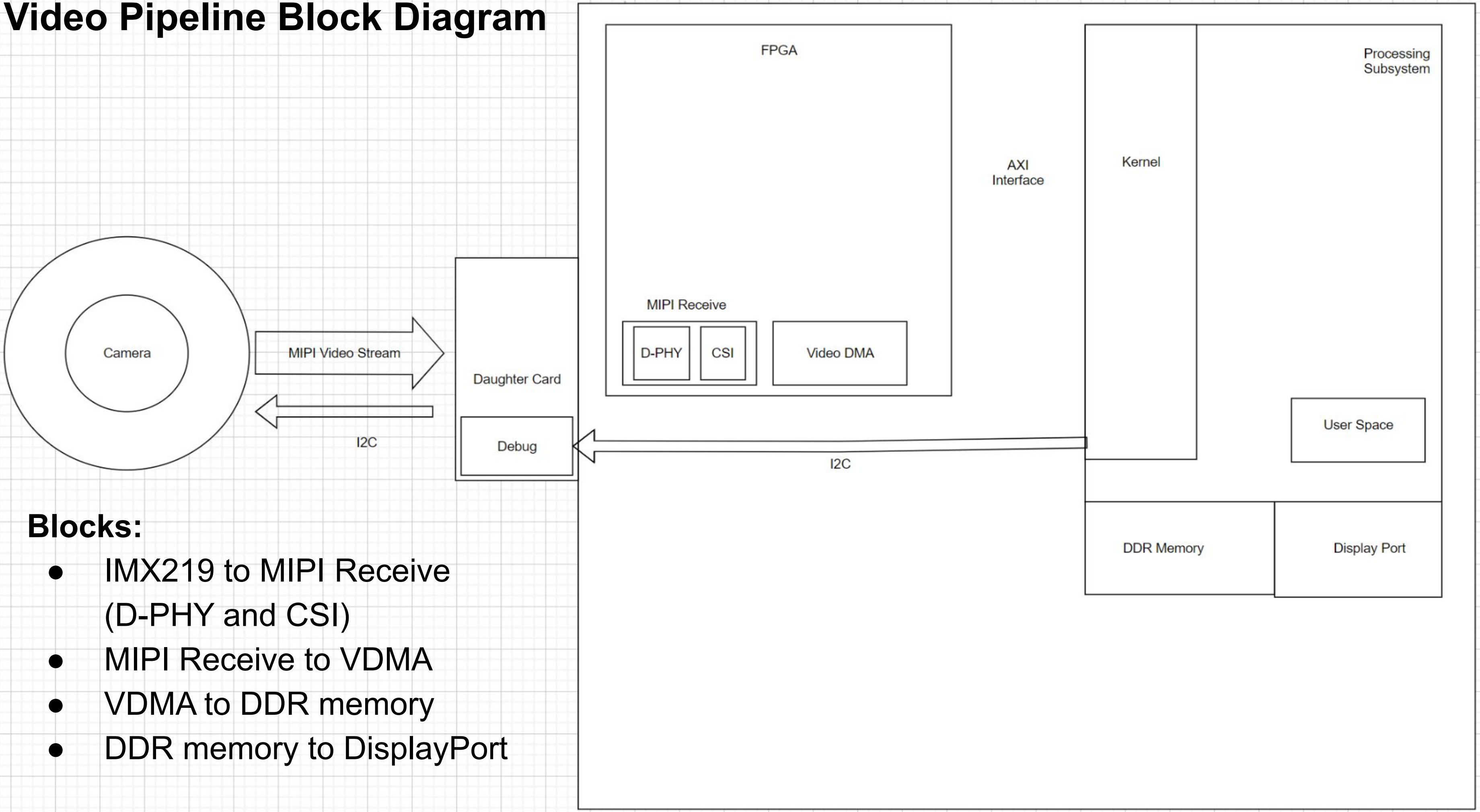
Vocabulary

- **IMX219 image sensor:** camera
- **MIPI:** mobile industry processor interface
- **CSI:** 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)



Ultra-96 FPGA Board

Video Pipeline Block Diagram



Blocks:

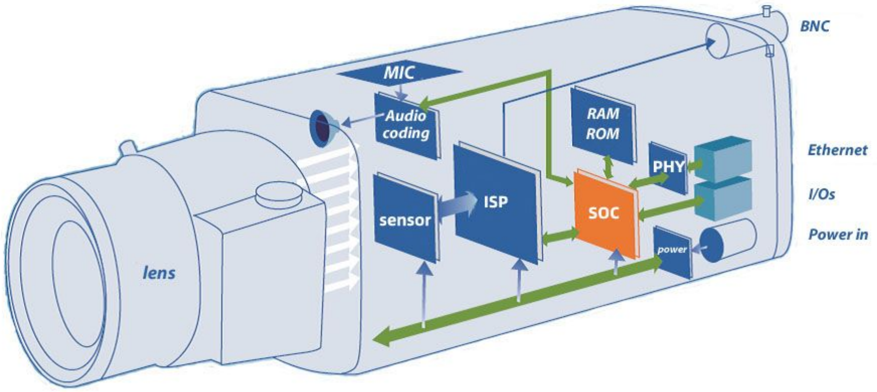
- IMX219 to MIPI Receive (D-PHY and CSI)
- MIPI Receive to VDMA
- VDMA to DDR memory
- DDR memory to DisplayPort

Technical perspective (continued)

Internal project complexity

Project requires extensive hardware design expertise

- Select individuals have:
  - A deep understanding of how cameras work
  - Knowledge on how to build a camera from scratch



Overview of Security Camera Internals

Technical perspective (continued)

- CPRE 488
    - 2-week digital camera lab using image sensors and FPGAs

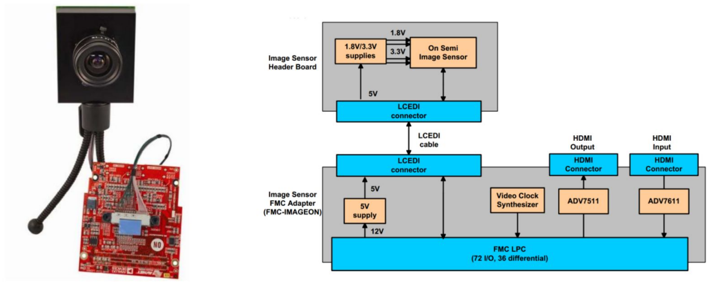


Figure: ON Semiconductor Image Sensor with HDMI Input/Output FMC Bundle – photo and block diagram.

CPRE 488 Digital Camera Lab

Questions?

Background Information

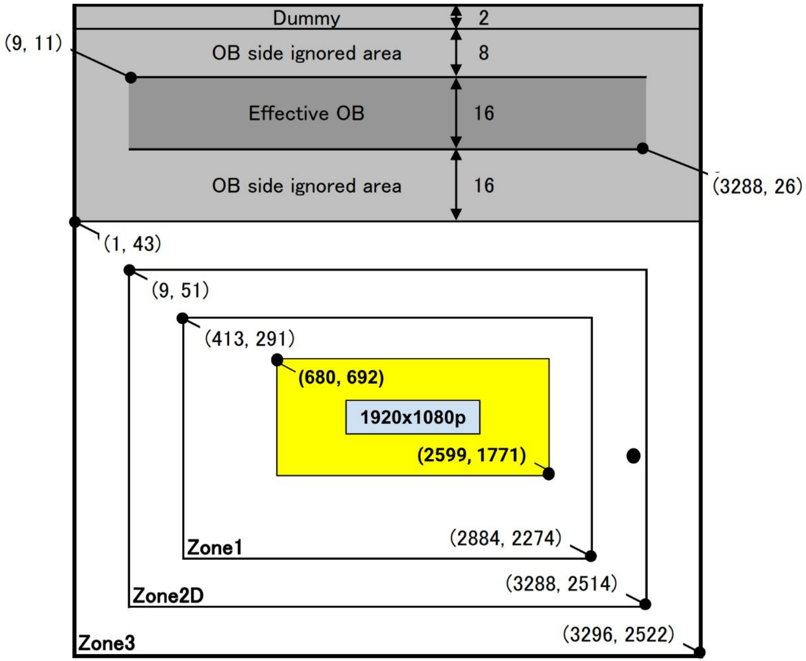
Expertise Development: Ritwesh

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



1920x1080p Frame Format

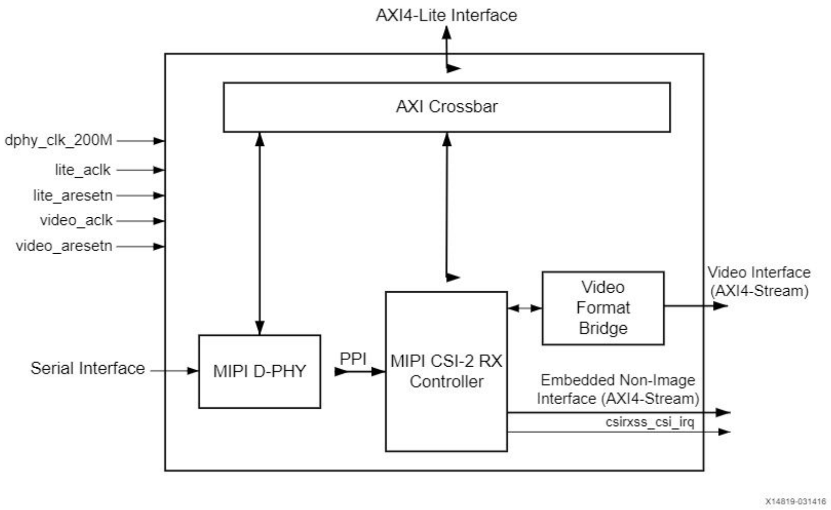
Expertise Development: Deniz

Configure MIPI Controller

- Correct data transfer protocol

Code Migration to PYNQ

- Memory access in python vs C/Shell



Expertise Development: Liam

VDMA

- Configure registers
- Understand through datasheets

Code Migration to PYNQ

- Replace frame buffer with VDMA

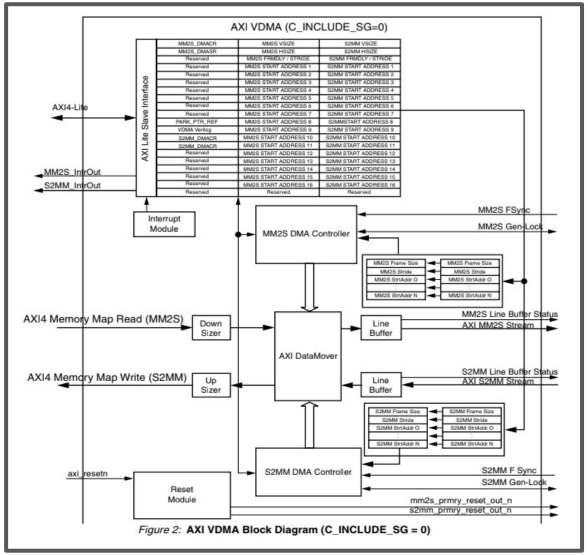


Figure 2: **AXI VDMA Block Diagram (C_INCLUDE_SG = 0)**

VDMA Block Diagram

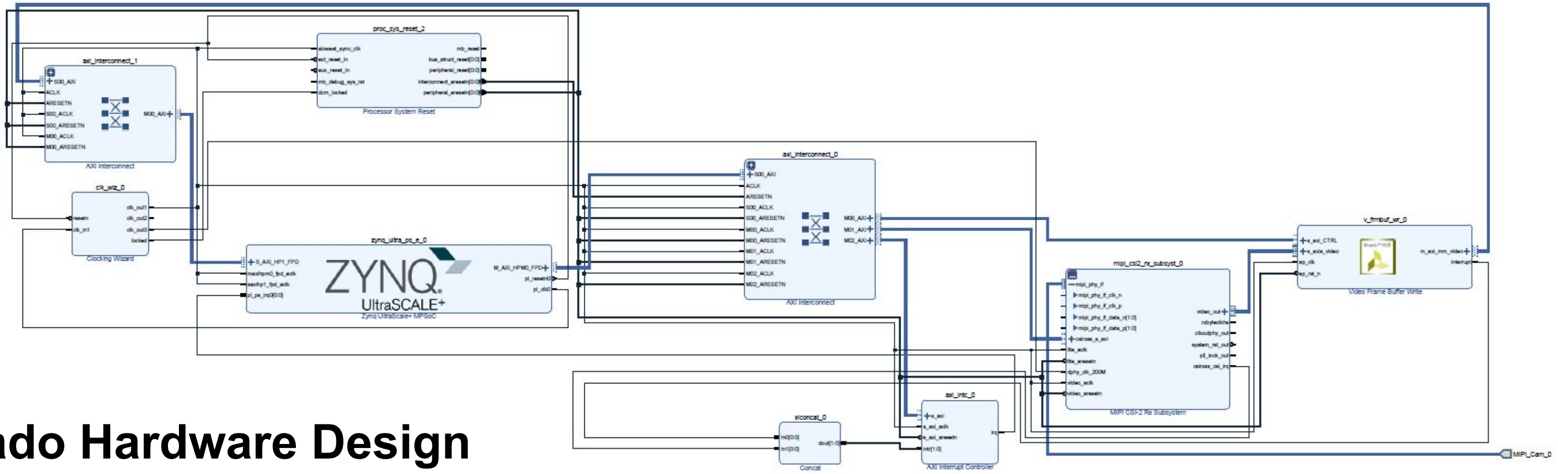
Expertise Development: Taylor

Hardware Components

- Vivado overlays
- Datasheets

Code Integration/Migration to PYNQ

- Building off previous teams
  - Shell scripts & C



Vivado Hardware Design