MACHINE LEARNING ASL INTERPRETER

Team 8
Grant Day
Josh Johnston
Kendra Francis
Leigha VanderKlock
Luke Metz

INTRODUCTION

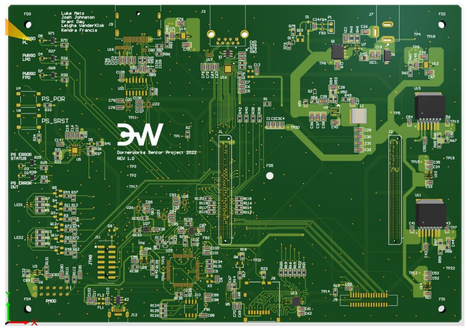
- American Sign Language interpreter that uses a machine learning algorithm accelerated using a field programmable gate array (FPGA).
- This project will act as a demo for the DornerWorks FPGA group, showing the acceleration capabilities of an FPGA on a machine learning application running on custom hardware.
  - This solution specifically makes use of the Xilinx Ultrascale+ Chipset.

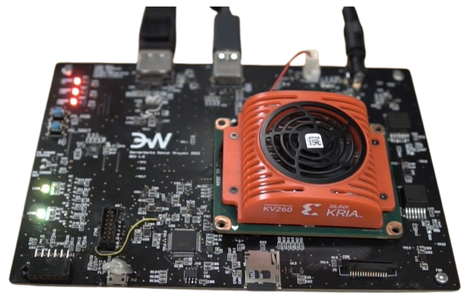
PROPOSED SOLUTION

- Our proposed solution consists of a couple different parts, specifically the ML model and hardware and software.
  - ML Model solution
    - Training: Darknet
    - Model: Yolov4
  - Hardware Solution
    - Kria SOM
  - Carrier Card
    - DisplayPort
    - USB Camera
    - SD card boot
    - UART
  - Software
    - Ubuntu image running the machine learning model with custom scripts that run the application upon boot up

ASL ALPHABET

HARDWARE

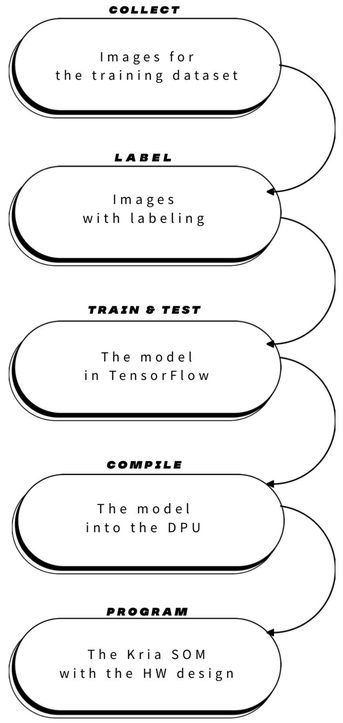




CARRIER CARD

DEMO

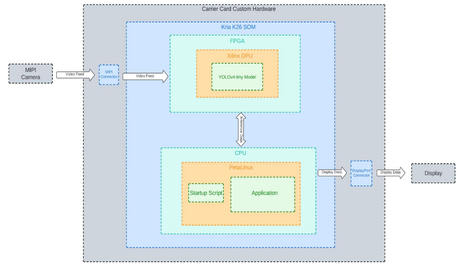
MACHINE LEARNING DESIGN FLOW



ML MODEL PROCESS FLOW

- The process of creating and training the machine learning model requires the following steps:
  - 1. Having a robust training dataset.
    - One of the most important aspects of the project.
    - Needs to be vast, have a large amount of variation in hand position, size, and skin tone.
  - 2. Choose a pre existing machine learning model.
    - Chose Yolov4-tiny.
      - Lightweight model that has weights pre-determined.
      - Layers and nodes already connected.
      - Took this model and trained it with our own ASL dataset.
  - 3. Training the model.
    - Used Darknet to train model, which provides a file of weights.
    - Then used Cafe to convert and quantize the model to run on the Xilinx Ultrascale+ FPGA.
  - 4. Compilation of the model.
    - Model is then compiled so that it can run on the Xilinx deep learning processor unit (DPU).

SYSTEM OVERVIEW



SYSTEM DIAGRAM