FPGA-based:
Surveillance System

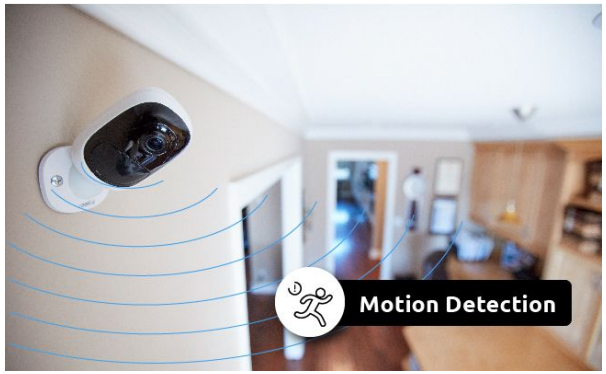
Group 1
Jia Yuan Chen, Han Jie Qiu, Xinran Rui

Background and Motivation

FPGA-based Surveillance System

Features:

- Motion Detection
- Alarm Notification
- Over-the-network Video Streaming
- Camera Angle Adjustment

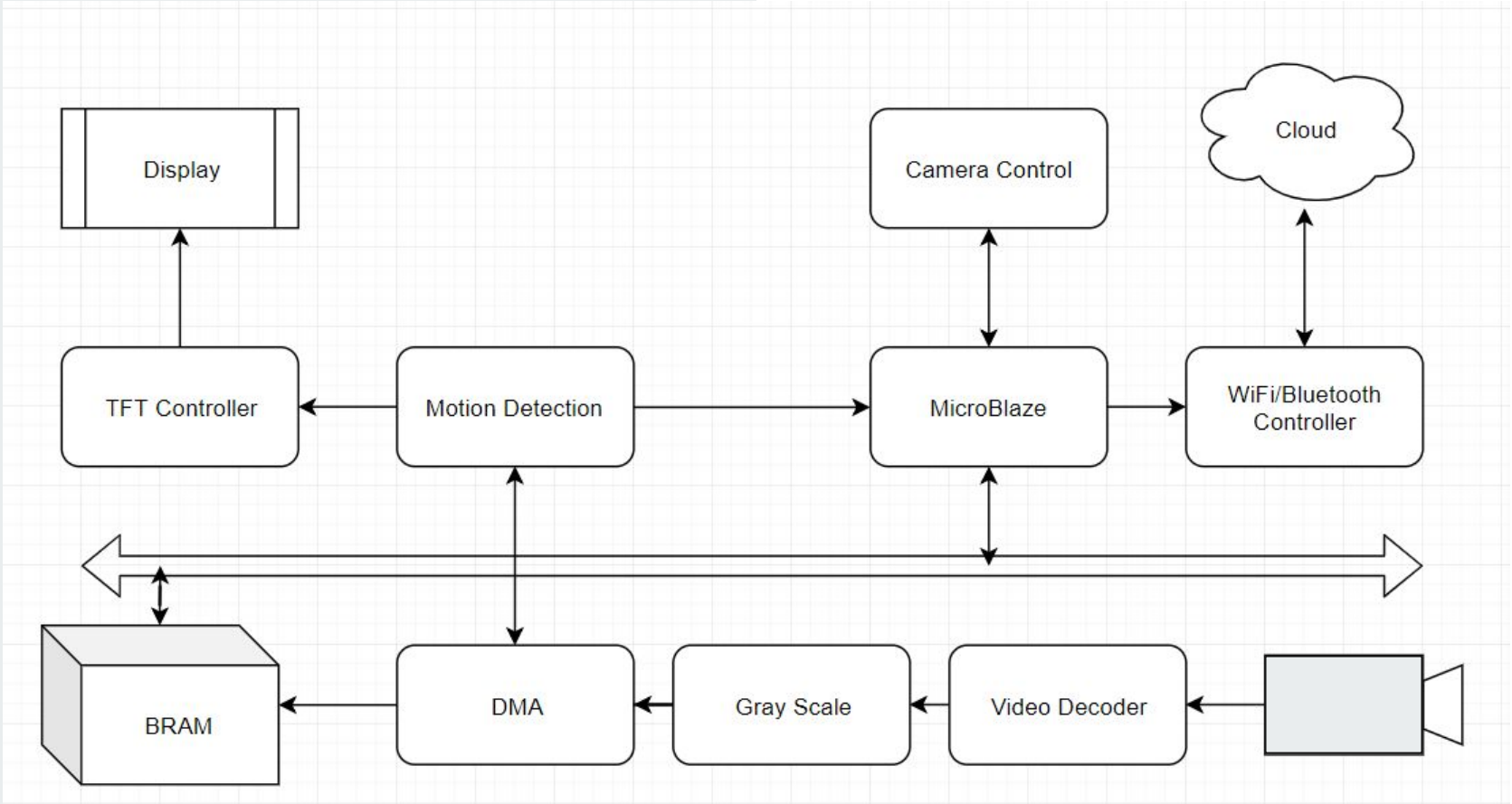


Connection to Internet of Things

Stay Connected:

- **Home Safety**
- **Cameras & Alarms all connected to the cloud**

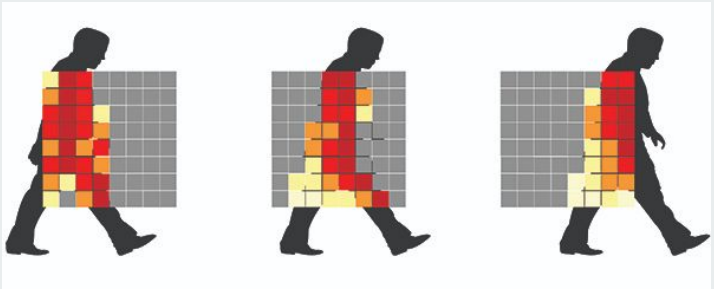
Proposed System Design


Display
Camera Control
Cloud
TFT Controller
Motion Detection
MicroBlaze
WiFi/Bluetooth Controller
BRAM
DMA
Gray Scale
Video Decoder

System Overview

Custom Hardware IPs:

1. Motion Detection
2. Camera Angle Control



Data Processing

Leveraging Existing IPs:

1. Xilinx DMA Controller
2. Xilinx TFT Controller
3. Digilent WiFi Controller

XILINX

DIGILENT®

Data Transferring

Implementation Plan - Milestones

Milestone 1.

DMA Test Bench and Simulation

Milestone 2

TFT Controller for Output to VGA Monitor

Network File Transfer

Camera Feed

Test Bench

DMA test bench allows the video processing hardware cores to be developed and tested independently of the framework

Framework

Parallel developments of individual framework components

Implementation Plan - Milestones

Milestone 3.

Basic Motion Detection Algorithm

Golden Software Algorithm

System Integration

TFT Controller +

Network File Transfer +

Live Camera Feed

Hardware Block

Implement a basic working Motion Detection Algorithm in software, turn it into hardware via ASM. Use the software as a golden solution to verify the hardware.

Integration

Feed the live camera data into the hardware block, which will output alarm signals if motion is detected. Also feed the data into TFT controller for display and into the network file transfer component

Implementation Plan - Milestones

Milestone 4.

- Migration of Motion Detection Algorithm
- Support Video Processing Hardware Cores

Milestone 5.

- Server-Side Application
- Client-Side Application

Enhancing the Motion Detection Algorithm

Explore different algorithms and try them in software. If one works particularly well, migrate it to hardware. Also add support for supplementary video processing cores.

Server & Client

Implement a server that communicates with MicroBlaze.

Implement a client that runs on MicroBlaze and sends image packets to server

Implementation Plan - Milestones

Milestone 6.

Camera Angle Adjustment Support

Milestone 7.

Application Extension

Integration and Clean Up

Mechanical Component

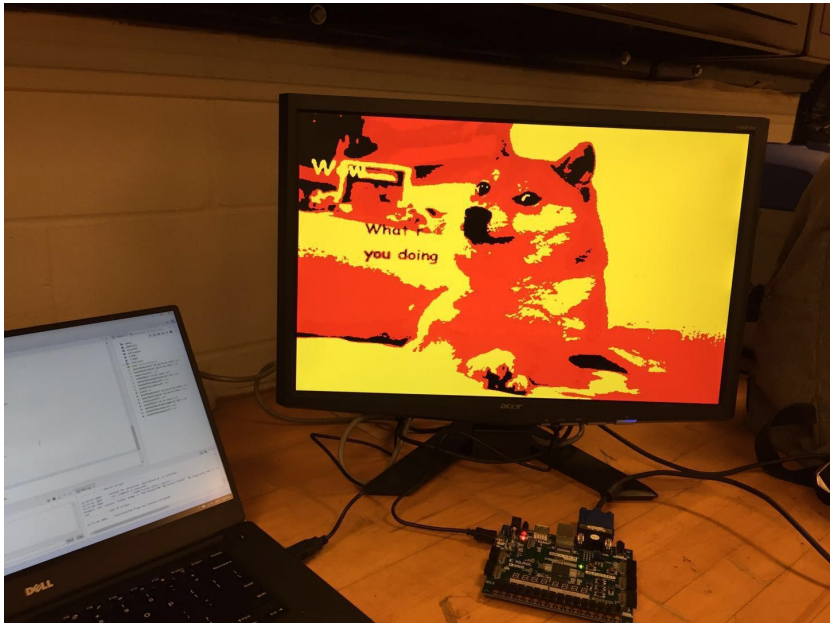
Connect the camera to a mechanical system that is capable of adjusting the camera angle through motors and PWM

More features

Motion sensitivity detection, high frame rate greyscale streaming, etc.

Current Status

```
Payload content:  BEGIN
Payload content:  This is packet # 1
Payload content:  This is packet # 2
Payload content:  This is packet # 3
Payload content:  END
```



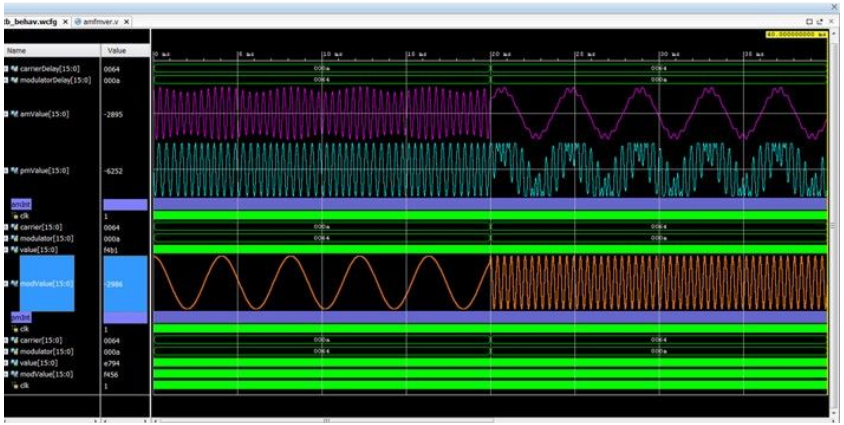
Network File Transfer

TFT Controller Sample Output

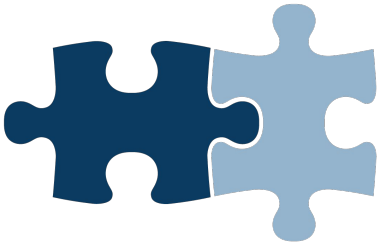
Testing and Integration Plan

- Verify each module functionality individually using
  - Simulation techniques
  - Implementation on FPGA
  - Comparison against golden software implementation

- Integration
  - Decide port protocols in advance (active HIGH/LOW etc.)
  - Integrate related modules first (TFT controller with MicroBlaze)
  - Test each module integration



AMBA



Uncertainties and Risks

- **The entire design might not fit on FPGA**
  - Choose the Nexys Video board
- **Bandwidth issues**
  - Compress images before sending
- **Motion Detection Algorithm**
  - Threshold control & tuning

	Nexys 4	Nexys Video
Logic Slices	15850	33650
BRAM	4860 Kbits	13 Mbits
DSP	240	740
DDR	128 MB	512 MB

Questions?