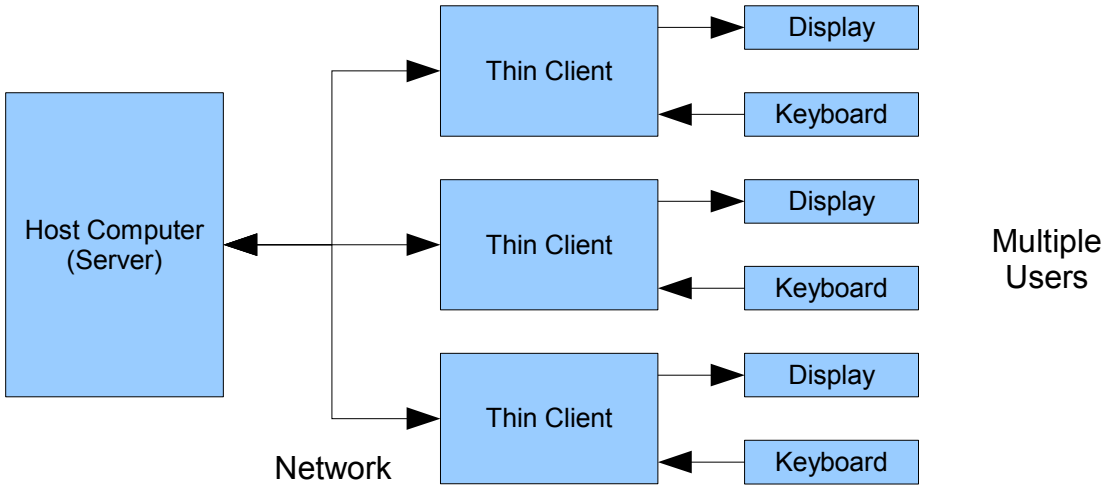
Design and Implementation of an FPGA Based Thin Client

By Liam Davey (13648671)

Supervised by Dr Cesar Ortega-Sanchez

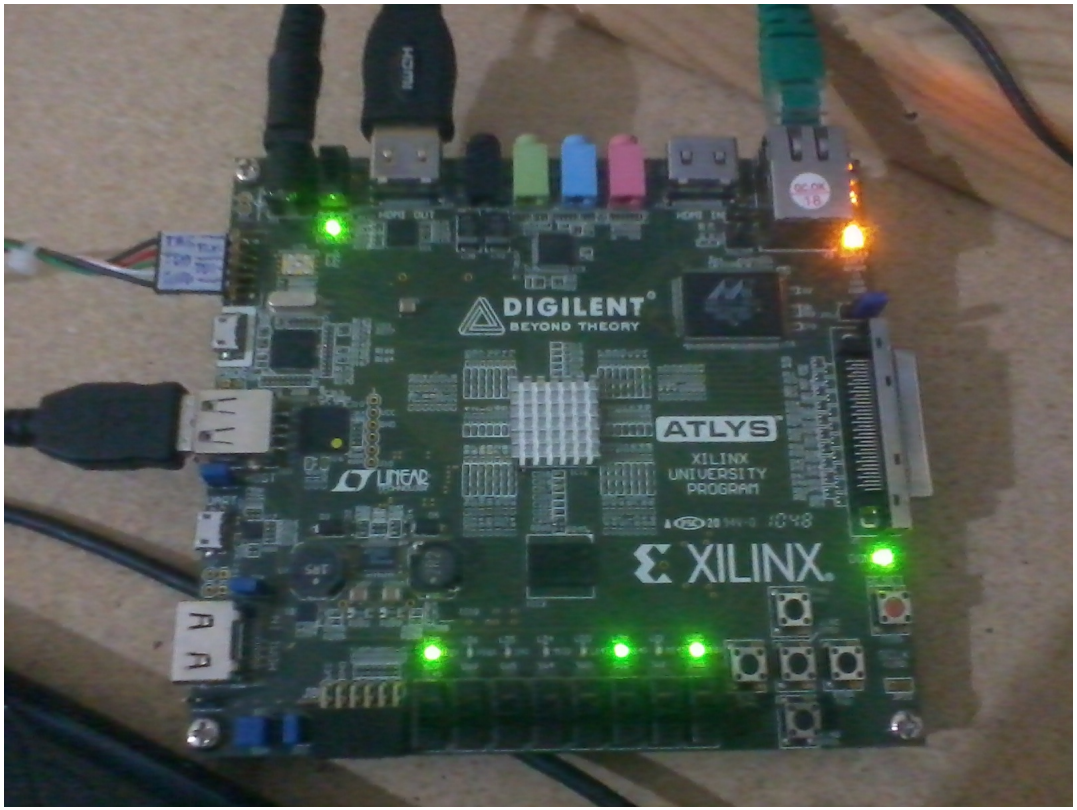
Introduction

- Thin Clients allow server resources to be shared among multiple users
- Traditional Thin Clients compress video data for transmission over low bandwidth networks meaning high processing overhead
- Proposed solution: FPGA Thin Client decoding uncompressed video transmitted over high bandwidth network



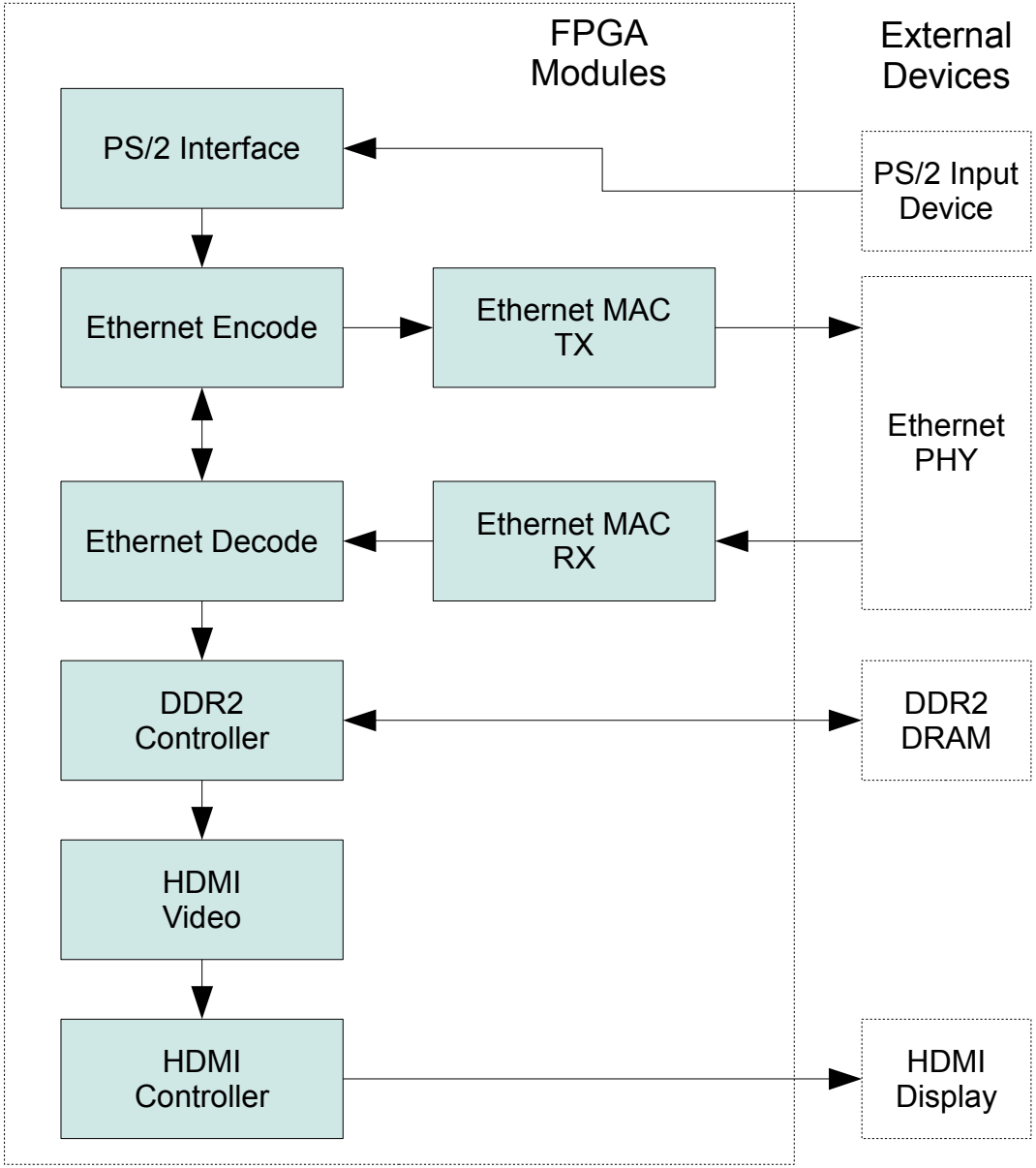
Problem Solution

- Gigabit Ethernet network interface
- HDMI digital display interface
- PS2 for input devices
- FPGAs good for high throughput applications
- Verilog HDL used
- Digilent Atlys development board as prototype hardware platform



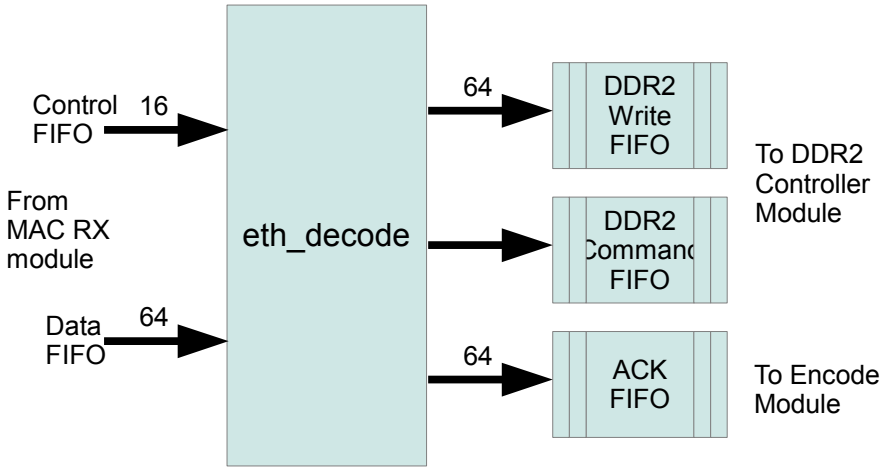
Top Level Design

- Ethernet subsystem and HDMI subsystem communicating via dual port DDR2 controller
- Framebuffer held in external DDR2 DRAM

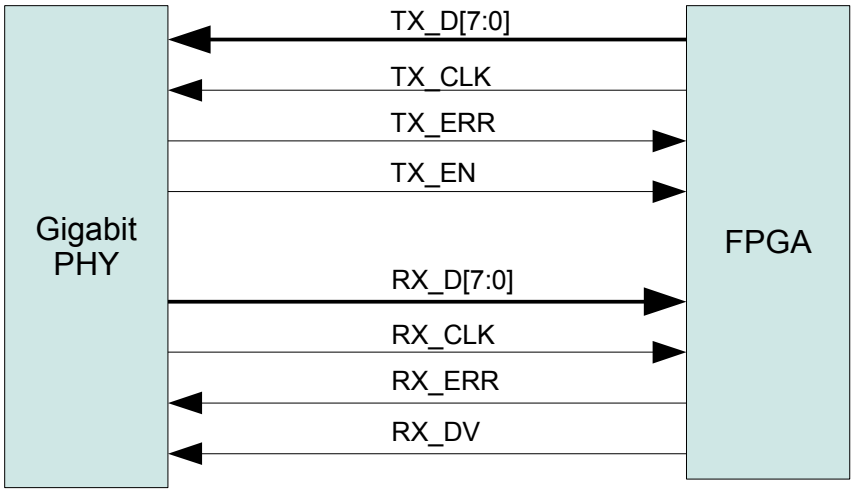


Ethernet Subsystem Design

- Receive side decodes frames, writes RGB data to framebuffer, writes acknowledgement if necessary
- Transmit side encodes outgoing acknowledgement frames or keyboard data frames
- Gigabit Media Independent Interface (GMII) between Ethernet PHY and FPGA



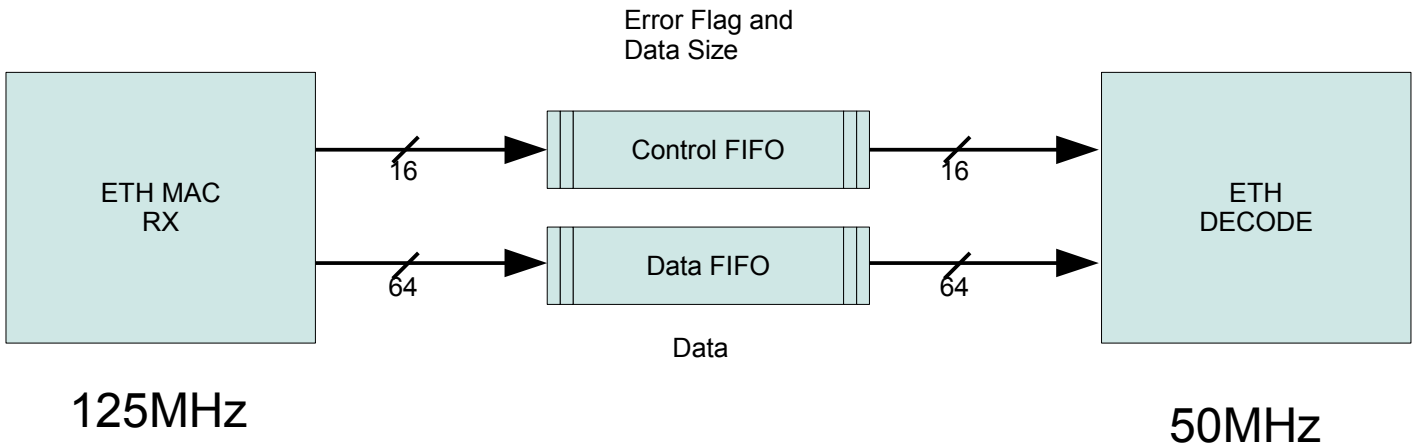
Ethernet Decode Module



GMII

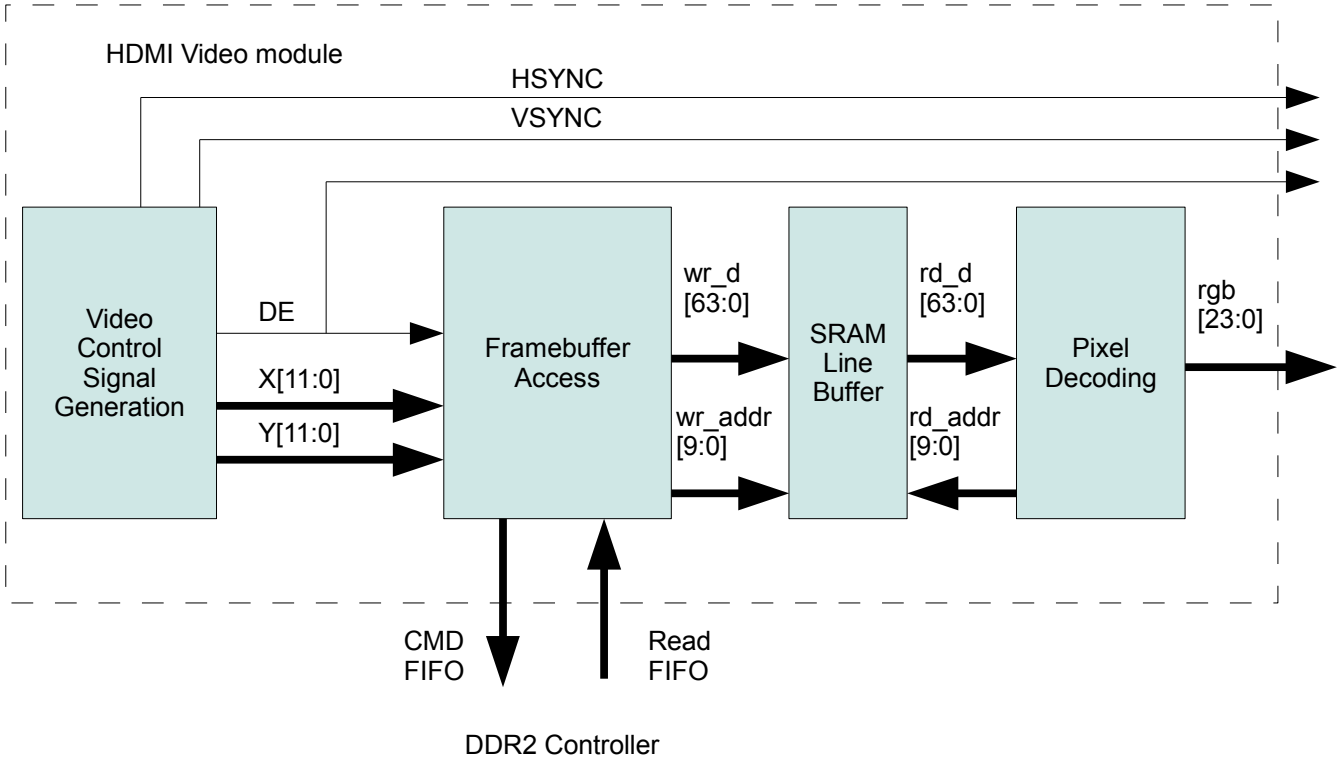
Frame Buffering and Checksum

- Interface between MAC and decode module problematic
- Cross clock domain (125MHz → 50MHz)
- Frames cannot be checked for validity until checksum is received at end of each frame
- Decode module must wait until frame checksum validated before decoding
- Solution: Control and Data Asynchronous FIFOs



HDMI Video Design

- Generates VSYNC, HSYNC, DE video signals
- Reads RGB data from DDR2 framebuffer
- Line buffer to prevent underflow and overflow problems

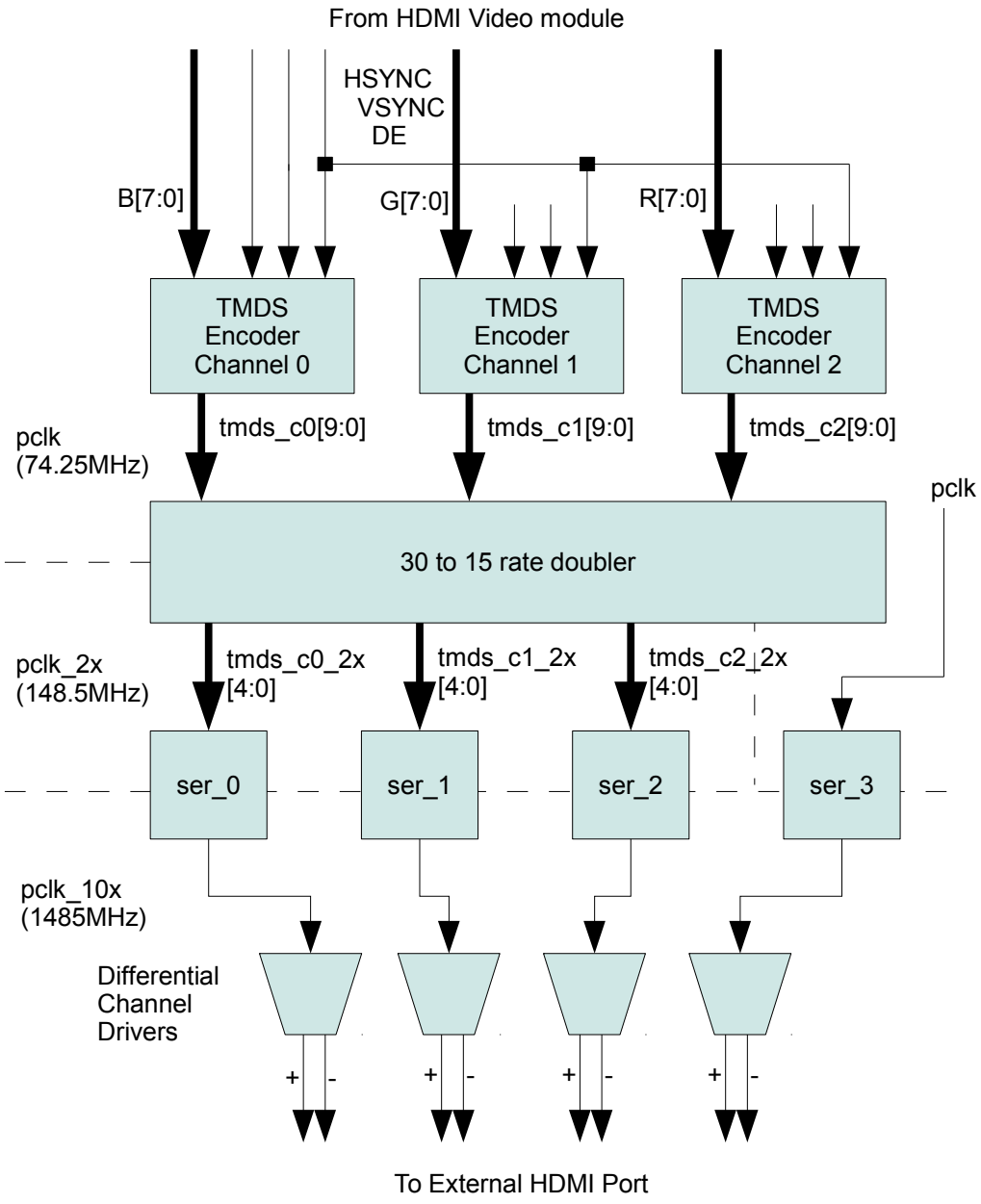


HDMI Video Test Mode

- Test mode generates test pattern and ignores data read from framebuffer
- Used to test video signal generation and HDMI controller separately

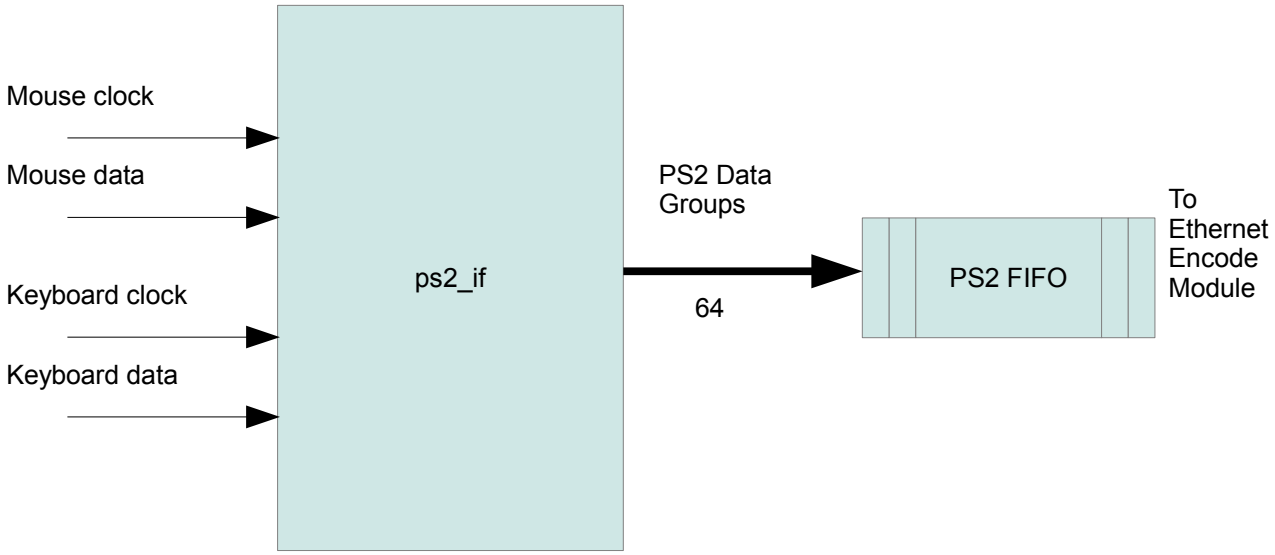
HDMI Controller

- Receives RGB data and video signals from HDMI video module
- Encodes using TMDS algorithm from DVI Specification
- Serialized using FPGA hardware SERializer-DESerializers (SERDES)
- Transmitted on four differential channels for HDMI



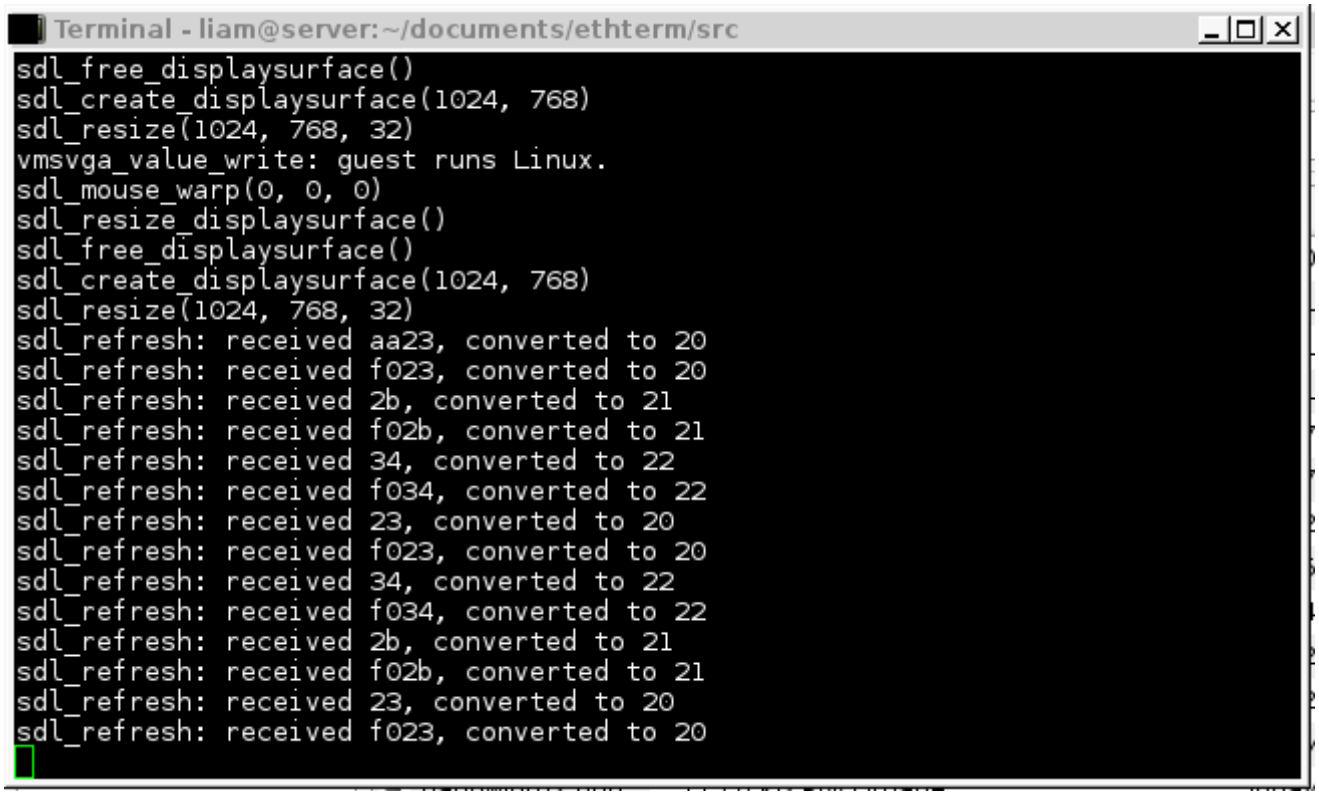
PS2 Controller

- Reads scan codes from keyboard
- Sends scan codes to Ethernet Encode module



Software

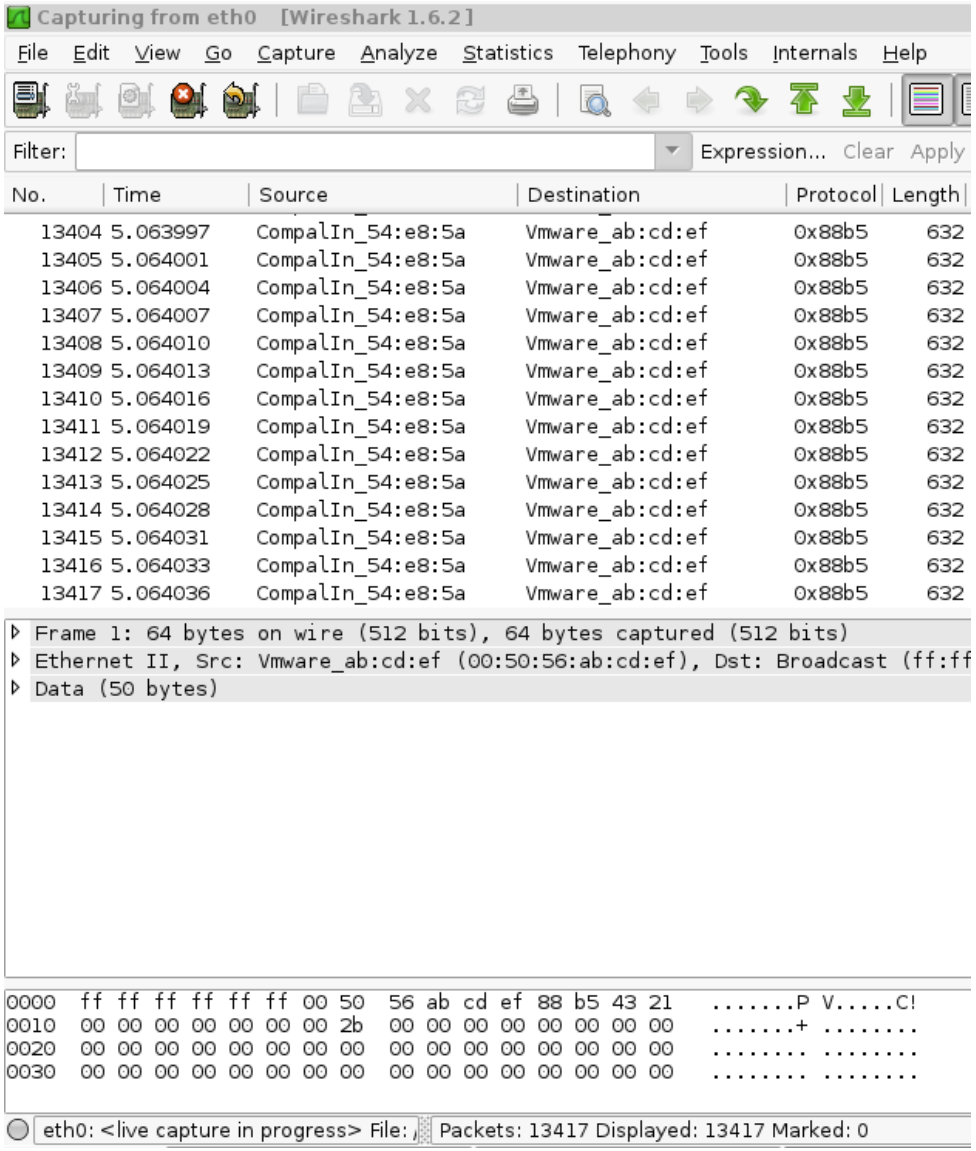
- QEMU, an open source virtualization solution
- Modified default video output driver to send Raw Ethernet frames
- Listened for incoming frames for keyboard data



Modified QEMU console output showing received keyboard data

Ethernet Verification

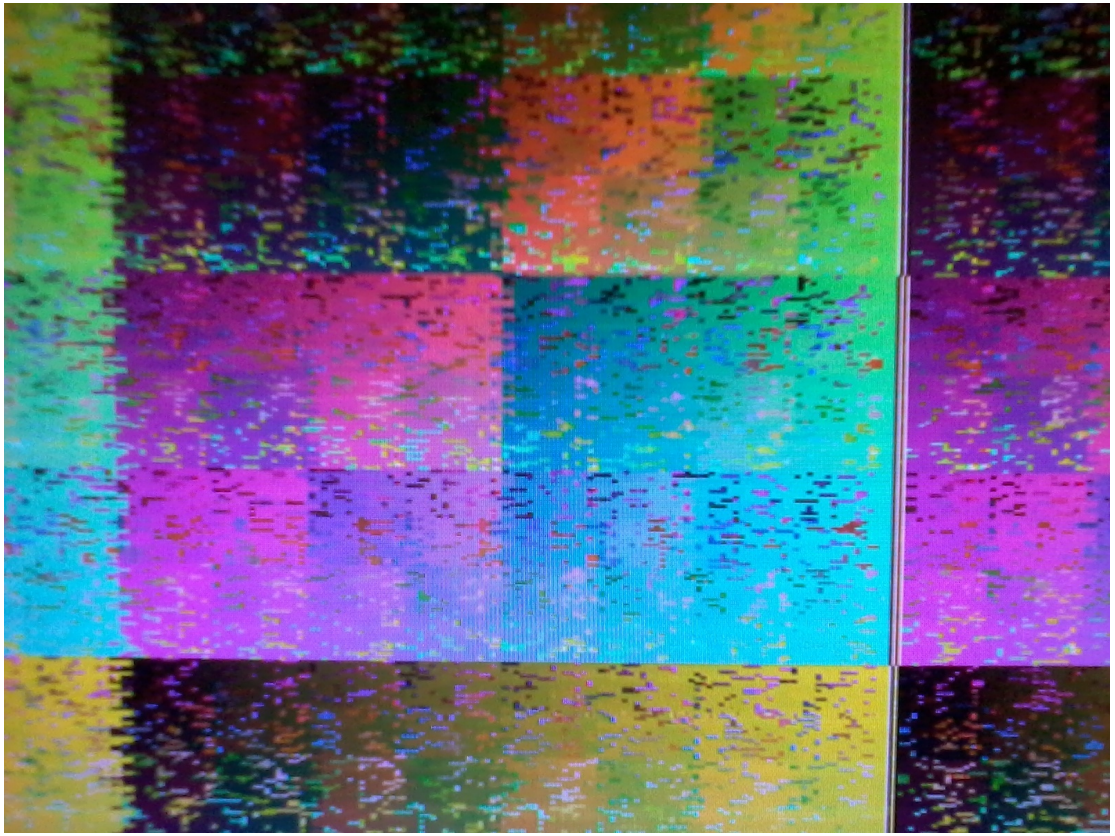
- Simulated with Verilog test benches then tested on FPGA
- Raw frames dumped and analyzed using Wireshark
- Problems with bit ordering and frame check sum



Example frame capture in Wireshark

HDMI Verification

- Test mode used to discover problem with incorrect clock
- DDR2 controller underflow and overflows caused image distortion
- Distortion fixed by adding line buffer



Distorted test pattern

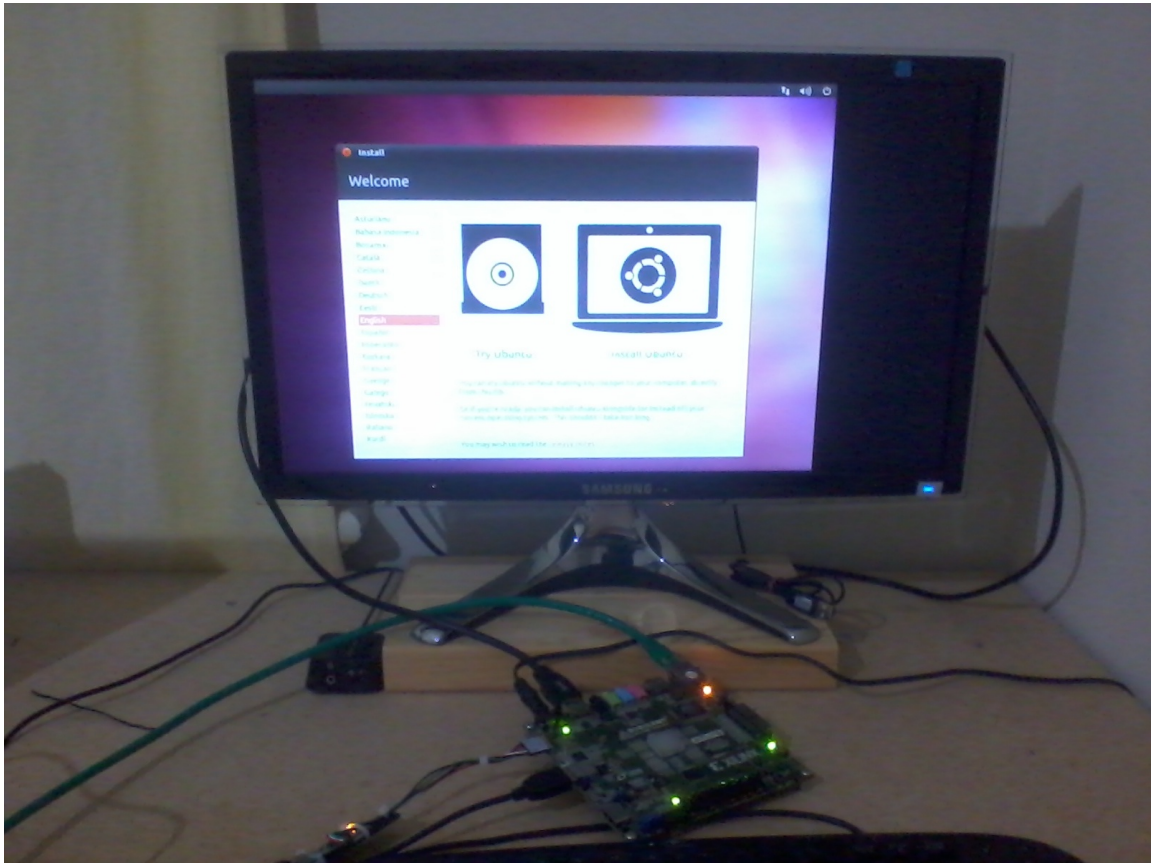
System Testing

- Testing with High Definition video
- Program written to send uncompressed video data to Thin Client
- 20 – 30 Frames per second displayed (1280 x 720 resolution)

A HD video being displayed on the prototype Thin Client

System Testing

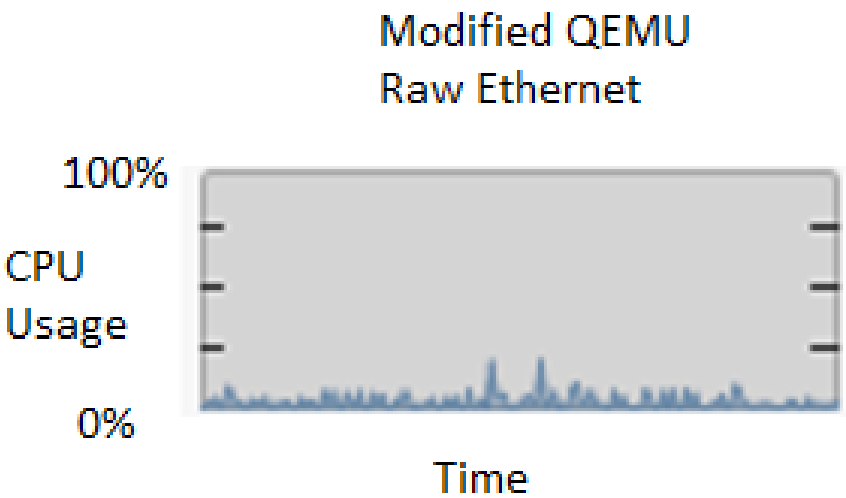
- Testing with modified QEMU
- Live CD used as virtualized guest under QEMU

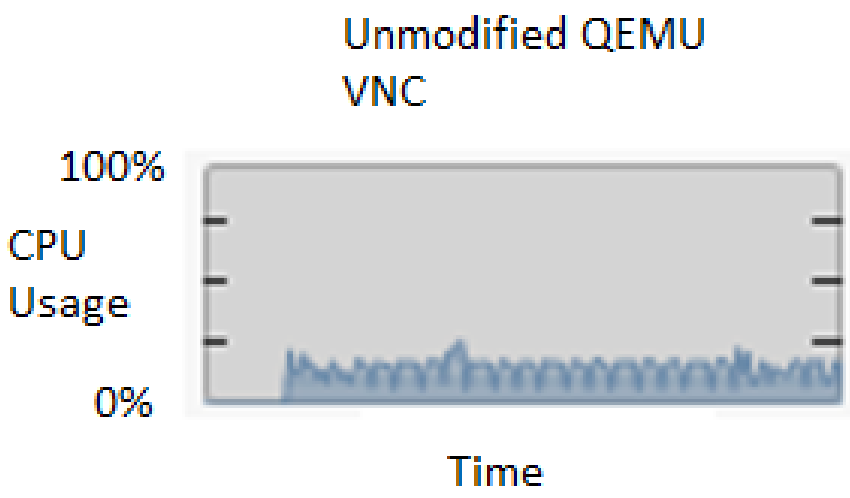


Prototype Thin Client being used to access virtualized guest on server

Test Results

- 57MB/s throughput using integrated Realtek NIC on Host PC
- 98MB/s throughput using Intel NIC on Host PC
- Test performed comparing VNC and modified QEMU CPU usage





Note that 25% CPU usage in thes graphs represents 100% usage of a single core of the quad core processor used in the test.

Conclusions

- Prototype FPGA based Thin Client successfully designed and implemented
- Server CPU overhead reduced compared to when using video compression
- Future Improvements: USB support, Audio support, more HDMI resolutions, Ethernet reliability improvement

End