

**HARDWARE AND CLOSE SUPPORT ELECTRONICS
ARCHITECTURES FOR ENABLING A PACKETIZED
DISPLAY PROTOCOL ON IRLED SCENE PROJECTORS**

by

Christopher Jackson

A dissertation submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical and Computer Engineering

Summer 2021

© 2021 Christopher Jackson
All Rights Reserved

**HARDWARE AND CLOSE SUPPORT ELECTRONICS
ARCHITECTURES FOR ENABLING A PACKETIZED
DISPLAY PROTOCOL ON IRLED SCENE PROJECTORS**

by

Christopher Jackson

Approved: _____
Jamie D. Phillips, Ph.D.

Chair of the Department of Electrical and Computer Engineering

Approved: _____
Levi T. Thompson, Ph.D.

Dean of the College of Engineering

Approved: _____
Louis F. Rossi, Ph.D.

Vice Provost for Graduate and Professional Education and
Dean of the Graduate College

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed:

Fouad Kiamilev, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed:

Chase Cotton, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed:

Richard Martin, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed:

Furkan Cayci, Ph.D.
Member of dissertation committee

ACKNOWLEDGMENTS

I want to extend my sincerest thanks to my advisor Fouad Kiamilev. I truly admire and appreciate the positive atmosphere he promoted to encourage all of us to succeed. Without his patience and guidance this work would not have been possible.

I would like to thank Dr. Chase Cotton, Dr. Richard Martin, and Dr. Furkan Cayci for agreeing to be members of my committee. I appreciate the immeasurable advice and support each of them has given me during my time at the university.

I also want to thank the past and present members of CVORG, all of whom I have enjoyed working with over the years. I particularly want to thank those of you that I have collaborated closely with: Dr. Tyler Browning, Dr. Aaron Landwehr, Dr. Garrett Ejzak, Jacob Benedict, Daniel May, Andrea Waite, Hamzah Ahmed, Dr. Miguel Hernandez-Raya, Tianne Lassitter, Rebekah Houser, Dr. Jonathan Dickason, Dr. Rodney McGee, Michelle McGee, and Dr. Kassem Nabha. Each and every member of CVORG made it feel more like a family than a research group.

I'd like to thank the faculty and staff in the ECE Department whose knowledge and support were essential to my success, including but not limited to: Dr. Mark Mirotznik, Dr. Dennis Prather, Dr. Ryan Van Antwerp, Dr. Daniel Weile, Dr. Hui Fang, Dr. Vishal Saxena, Debbie Nelson, Gwen Looby, Cyndi McLaughlin, Thomas Lum, Robert Schmidt, and Kjeld Krag-Jensen.

I would also like to extend my appreciation to Dean Michael Vaughan, Veniece Keene, and the Greater Philadelphia Region Louis Stokes Alliance for Minority Participation (LSAMP) for awarding me the Bridge to the Doctorate (BTD)

Fellowship. True to its name, the BTD fellowship inspired me to continue toward completion of a Ph.D.

Thank you to my friends for the support and fun times: Ryan, Jimi, David, Aaron, Alistair, Kristen, Darius, Rich, Lena, Steve, and Caroline.

Most importantly, I would like to thank my parents, siblings, and all of my extended family. Their endless encouragement and love are the reason I was able to pursue this dream. I am eternally grateful for their belief in me and unconditional, unwavering support.

TABLE OF CONTENTS

LIST OF TABLES	viii
LIST OF FIGURES	ix
ABSTRACT	xiv
1 INTRODUCTION TO IRSP	1
1.1 IR Scene Projection	2
1.2 Resistor Arrays	2
1.3 IRLED Arrays.....	3
1.4 Dissertation Outline, Goals and Contributions	6
1.4.1 Outline	6
1.4.2 Goals	6
1.4.3 Contributions	7
2 BACKGROUND AND CURRENT STATE OF SLED PROJECTORS	8
2.1 Superlattice LED System.....	8
2.2 TCSA, NSLED, HDILED	8
2.2.1 RIIC Overview.....	10
2.2.2 CSE Overview	10
2.2.3 Scene Generation and NUCPC	11
2.3 AIREA, HSLEDS, and Future Systems.....	14
3 CONVENTIONAL DISPLAY PROTOCOLS AND PDP	15
3.1 DVI and HDMI.....	15
3.2 Limitations of Conventional Display Protocols.....	15
3.3 Packetized Display Protocol	18
3.3.1 Features	20
3.3.2 PDP vs SNAP	24
3.3.3 Results.....	26
3.3.4 Future PDP Considerations.....	28
4 CLOSE SUPPORT ELECTRONICS AND GIGABIT INTERFACE.....	31
4.1 Components of CSE and Datapath	31
4.2 Development of Gigabit Interface	33

4.2.1	Motivation.....	33
4.2.2	Design	36
4.2.3	Implementation	41
4.3	Gigabit Interface: Results, Testing, and Troubleshooting	61
5	TOWARD DESIGN OF A DISTRIBUTED CSE	79
5.1	Multi-CSE System	79
5.2	Interim System – NESSIE2	80
5.2.1	Synchronization	81
5.2.2	NESSIE2 Design.....	89
5.2.3	NESSIE2 FMC216 Testing and Verification	98
6	PROPOSED DISTRIBUTED CSE SYSTEM	106
6.1	Advantages of Distributed CSE Design.....	106
6.2	NESSIE3 Design Details	107
6.2.1	NESSIE2.5	107
6.2.2	Design Considerations and Requirements	110
6.2.3	NESSIE3 Design Components	113
6.3	Implications for PDP	114
6.3.1	Protocol Write Flexibility	115
6.3.2	Routing to Each Mini-CSE	116
7	CONCLUSION	118
REFERENCES	120
A	ADDITIONAL ZYNQ-FMCL-HDMI SCHEMatics AND LAYOUTS....	130
B	SUPPLEMENTARY ZYNQ-FMCL-HDMI TEST RESULTS.....	137
C	LIST OF TERMS	147

LIST OF TABLES

Table 1	Examples of PDP Packets	18
Table 2	A comparison between Snap firmware and PDP firmware.....	24
Table 3	NESSIE2.5 FPGA Candidates Comparison	109
Table 4	RFSoC FPGA Comparison.....	112
Table B.1	ZYNQ-FMCL-HDMI Testing Checklist, Boards 1-12	137
Table B.2	ZYNQ-FMCL-HDMI Testing Checklist, Boards 13-24	138
Table C.1	List of Terms	147

LIST OF FIGURES

Figure 1	University of Delaware IRLED Scene Projector System.....	5
Figure 2	Superlattice LEDs (SLEDs) Projectors Timeline, Part 1	9
Figure 3	SLEDs Projectors Timeline, Part 2	9
Figure 4	Scene Generation Example.....	12
Figure 5	Scene Projection Overview	13
Figure 6	Traditional video frame with horizontal and vertical blanking.....	16
Figure 7	PDP Backwards Compatibility Mode	19
Figure 8	PDP Stream Mode	19
Figure 9	Simplified First-Generation PDP Architecture	21
Figure 10	Simplified Synchronized Circular Buffer (SCB)	21
Figure 11	Overview of Array Emitter.....	22
Figure 12	PDP State Machine	23
Figure 13	(Top) HDMI frame running at 1080p displaying a static square image of size 32x32 at a frame rate of 60 Hz. (Bottom) An equivalent PDP frame operating with no blanking regions and maximal packing PDP data packets for an effective window of size 32x32 running at a frame rate over 144 KHz	25
Figure 14	Example showing full-frame PDP output of a cross-pattern input image in backwards compatibility mode, running at 100 Hz.....	27
Figure 15	Example dual-window test displaying numbers counting from 1 to 60 at 60 Hz and 1 to 30 at 30 Hz with 32x32 size sub-windows.	28
Figure 16	Minimum Blanking and TMDS Timing Parameters	29
Figure 17	CSE Datapath	33

Figure 18	TB-FMCL-HDMI Indirect Loopback and BRAM Pull	35
Figure 19	1000 Hz BRAM Pull.	36
Figure 20	First-Generation HDMI Board.	37
Figure 21	ZYNQ Architecture	39
Figure 22	Second-Generation HDMI Board.....	39
Figure 23	ZYNQ-FMCL-HDMI Power Layers and Voltage Distribution	44
Figure 24	ZYNQ-FMCL-HDMI Power Regulation Schematic	46
Figure 25	ZYNQ-FMCL-HDMI Power Regulation PCB Layout	49
Figure 26	ZYNQ-FMCL-HDMI DDR Schematic.....	50
Figure 27	ZYNQ-FMCL-HDMI DDR PCB Layout	52
Figure 28	ZYNQ-FMCL-HDMI JTAG, Flash, and SD Card Schematic	54
Figure 29	ZYNQ-FMCL-HDMI JTAG, Flash, and SD Card PCB Layout.....	56
Figure 30	ZYNQ-FMCL-HDMI Board HDMI Connectors Schematic	57
Figure 31	ZYNQ-FMCL-HDMI Board HDMI Connectors PCB Layout	59
Figure 32	Current Generation ZYNQ-FMCL-HDMI Board	60
Figure 33	HDMI Probe Board Schematic.....	67
Figure 34	HDMI Probe Board PCB Layout	68
Figure 35	Standalone test setup, probing HPD (yellow), voltage rails 3.3 V, 1.8 V, and 1.5 V (orange, green, and blue).	70
Figure 36	Spikes on HPD signal after the completion of XorgService restart.	70
Figure 37	Test setup with eight DACs, eight amplifiers, and the master interface board connected. Small ripples on HPD but HDMI cards still detected.	71
Figure 38	ZYNQ-FMCL-HDMI ground plane probe referenced to CSE case GND.	72
Figure 39	Inrevium TB-FMCL-HDMI card ground plane shows similar results....	72

Figure 40	ILA, 240 Hz. Blips seen on vertical data enable (VDE) and data lines during horizontal blanking period.	73
Figure 41	ILA, 240 Hz. All HDMI data and clock signals valid and ready.	74
Figure 42	ILA, 240 Hz. HDMI cables connected directly from NUCPC to HDMI cards, bypassing the internal case HDMI cables. Signals are correct.	74
Figure 43	PRBS detects as active indicating sequence start is detected and data sent. However, data blips still occur during horizontal blanking.	75
Figure 44	As a result of the data corruption during horizontal blanking, PRBS fails after first row of data.	75
Figure 45	Probe of HDMI TMDS signals on internal cable - DATA0P (yellow), DATA0N (green), CLK0P (orange), CLK0N (blue).	76
Figure 46	Probe of HDMI TMDS signals on cable directly from NUCPC - DATA0P (yellow), DATA0N (green), CLK0P (orange), CLK0N (blue). No observable difference from previous test.....	77
Figure 47	Zoomed-in version of this test at 100 Hz in attempt to decode control token to detect any corruption.	78
Figure 48	Prototype Multi-CSE System	80
Figure 49	JESD Subsystem.....	82
Figure 50	JESD Timing Diagram	84
Figure 51	System of Multiple FPGAs and DACs and Clock Generator	85
Figure 52	NESSIE2 Block Design.....	88
Figure 53	NESSIE2 Hardware Design	90
Figure 54	Samtec EQCD-040 Digital Cable.....	92
Figure 55	Samtec TCF-2650F Analog Cable	93
Figure 56	Two Rows of 20 Analog Cables, Samtec HDR-166968-01-EQRF	93
Figure 57	Analog Signal Format.....	94
Figure 58	Custom Kintex Ultrascale FPGA Control Board	96

Figure 59	Firmware Block Diagram	97
Figure 60	Sine Wave Abaco DACs	100
Figure 61	Square Wave Abaco DACs	101
Figure 62	Square wave using Anaconda cables connected directly from FMC216 to the scope. DACs channels 1 (yellow) and 13 (blue) were observed.	103
Figure 63	Fastest square wave observed. DACs channels 1 (yellow) and 13 (blue).....	104
Figure 64	Fastest square sine wave observed using Anaconda cables connected directly from FMC216 to the scope. DACs channels 1 (yellow) and 13 (blue) were observed.	104
Figure 65	NESSIE2 Initial Test Bed.....	105
Figure 66	NESSIE2.5.....	108
Figure 67	NESSIE3 Initial Design.....	114
Figure 68	NESSIE3 PDP Packet Data Routing	117
Figure A.1	FPGA I/O Banks.....	131
Figure A.2	FPGA DDR and MIO Banks	132
Figure A.3	FPGA Power.....	133
Figure A.4	FMC Connector and General User I/O Schematic	134
Figure A.5	FMC Connector PCB Layout	134
Figure A.6	ZYNQ-FMCL-HDMI Full Schematic.....	135
Figure A.7	ZYNQ-FMCL-HDMI Full PCB Layout	136
Figure B.1	First DAC, connected to FMC slot 7.....	138
Figure B.2	Second DAC, connected to FMC slot 2.	139
Figure B.3	Third DAC, connected to FMC slot 8.	139
Figure B.4	Fourth DAC, connected to FMC slot 3.	140

Figure B.5	Fifth DAC, connected to FMC slot 9	140
Figure B.6	Sixth DAC, connected to FMC slot 4.....	141
Figure B.7	Seventh DAC, connected to FMC slot 10.....	141
Figure B.8	Eighth DAC, connected to FMC slot 5.	142
Figure B.9	First amplifier on FMC slot 7, 8 DACs.....	142
Figure B.10	Second amplifier on FMC slot 2, 8 DACs	143
Figure B.11	Third amplifier on FMC slot 8, 8 DACs.	143
Figure B.12	Fourth amplifier on FMC slot 3, 8 DACs.....	144
Figure B.13	Fifth amplifier on FMC slot 9, 8 DACs.	144
Figure B.14	Sixth amplifier on FMC slot 4, 8 DACs.....	145
Figure B.15	Seventh amplifier on FMC slot 10, 8 DACs.	145
Figure B.16	Eighth amplifier on FMC slot 5, 8 DACs.....	146
Figure B.17	Master and Slave Interface Boards with all 8 DACs and amplifiers....	146

ABSTRACT

Infrared (IR) imaging systems can characterize the unique IR signatures of high-temperature objects, making them extremely useful for various military, academic, and industrial applications. Infrared Scene Projectors (IRSP) are devices for simulating real-time IR scenes and allow the user to test and calibrate imaging systems in scenarios that would otherwise be impractical. In 2014, our research group created the first infrared light-emitting diode (IRLED) scene projector called Super-lattice Light Emitting Diode Systems (SLEDS). Since then, these systems have gone through several iterations, each generation increasing the resolution and frame-rates to keep up with consumer demands. IRLED scene projectors are used in hardware in the loop testing (HITL) to test real-time imaging systems with synthetic infrared imagery.

Scene generators produce synthetic infrared imagery, which has historically been transmitted via traditional display protocols to be projected by the IRSP. Traditional display protocols have limitations in terms of high bandwidth requirements, fixed refresh-rates, and control over how frames are displayed. A novel packetized display protocol (PDP) is under development to expand display capabilities for intelligent bandwidth utilization, dynamic inter-frame and intra-frame (sub-window) frame-rates, increased performance, and eased synchronization burden.

This research investigates the hardware necessary to enable PDP on IRLED scene projector systems and to interface with outside scene generator systems. A custom prototype printed circuit board to interface with the IRLED scene projector's close support electronics (CSE) is presented. This board utilizes a field-programmable

gate array (FPGA) and two input/output HDMI links. FPGAs are useful and powerful devices that can be reconfigured to meet the needs of the user. FPGAs perform highly parallel computations, potentially allowing for implementation and hardware acceleration of algorithms (such as non-uniformity correction) directly on the chip.

This first prototype board uses HDMI as the transport medium for PDP to both maintain backward compatibility with prior SLEDs systems and to demonstrate how PDP can be implemented onto any physical data transfer link.

This thesis describes the design and architecture of the next-generation CSE necessary to match future IRLED IRSP resolution and frame-rate demands. This architecture eschews the traditional display protocol hardware links for scene generation in favor of high-speed serial data connection, such as ethernet or fiber. Increased effective bandwidth due to elimination of video blanking periods allows the full potential of PDP to be demonstrated. Synchronization of the new generation CSE DACs through the JESD204B protocol is presented. Design considerations for a more robust third-generation distributed FPGA CSE are discussed.

Chapter 1

INTRODUCTION TO IRSP

Infrared (IR) imaging technology has a variety of applications, including military, academic, automotive, medical, and space industries. IR wavelengths range between 700nm and 1mm, which is below the visible spectrum, and as such require special sensors designed to capture this radiation [1, 2]. The infrared portion of the electromagnetic spectrum can reveal different information about the environment than the visual, such as information about the temperature of an object. Several devices decipher IR sensor information for practical use, including night vision goggles, gas detectors, and infrared cameras.

A particularly useful type of IR technology is known as an infrared scene projector (IRSP). IRSPs are devices that provide simulated heat signatures which can be detected by IR sensors. As a result, IRSPs are tools that can be used to characterize, test, and evaluate complex IR systems to ensure that they are operating as intended. A major advantage of an IRSP is that it is significantly less expensive, more practical, and safer than using IR sensors to directly measure heat signatures in the field. For example, instead of repeatedly firing missiles and measuring the heat signature directly using a sensor, one can synthetically project a virtual missile heat signature using an IRSP in a laboratory setting, thus requiring far less money, space, and safety precautions. This combination of the IRSP and IR sensor system is called a hardware in the loop (HWIL) configuration, which is used to provide the unit under test (UUT) with realistic projected scenery. [3, 4, 5]

1.1 IR Scene Projection

This section provides a small overview of the infrared scene projection process. An IR imaging system, also known as the unit under test (UUT), is used to observe a real-world scene, often at great distances. An IRSP projects an image or scene for the UUT to observe. The main components of an IRSP are the emitter array (also known as a microdisplay), a read-in integrated circuit (RIIC), close support electronic (CSE) to drive the emitter array, a means to produce scene data called a scene generator, a way to provide non-uniformity correction (NUC), and a radiometric mapping of each pixel in the display [3, 5, 32]. In order for the IRSP to reproduce a real-world scene, it must produce accurate and precise radiance values at any location on the sensor at any given time with minimal aberration [3, 5, 6, 7]. Many IRSPs must be capable of updating in real-time based on the unit under test, which is called hardware in the loop (HWIL) testing [5, 8, 9, 10].

1.2 Resistor Arrays

The resistor array technology for IRSP systems is the only one being used commercially, as it was the first technology to successfully produce dynamic scenes in the IR spectrum. In 1994, Barrett E. Cole and Chien J. Han of Honeywell filed the first patent on this technology [11, 12]. The concept for the technology is described below. Electrical current passes through a resistor to cause the resistor to heat up and generate IR signatures. Each pixel is composed of a small serpentine resistor which is suspended over the pixel driver to avoid transmitting the heat to the driver's silicon structure [4, 13]. Furthermore, this structure prevents lowering the resistor temperature and increasing the rise time. Thus, this tuned cavity gets more energy out of the resistor. Resistor pixels are grown individually on top of a silicon read-in integrated

circuit (RIIC) that uses complementary metal-oxide-semiconductor (CMOS) technology. The drive transistor is connected in series with the serpentine resistor, as the transistor controls the current through the resistor. A second transistor acts as the address line and signal voltage for the main drive transistor [13]. The emitter array effectively forms a square grid by connecting many of these devices in parallel [5].

Santa Barbara IR (SBIR) licensed this technology from Honeywell and are now the only commercial resistor array IRSP providers [14]. The most recent commercial model - MIRAGE XL - has resolution 1024x1024 and can reach apparent temperatures of 650 K [15]. Apparent temperature is thermographically equivalent to the radiation that a blackbody emits in the same spectral area of interest [16]. The smallest pixel pitch of commercially available resistor arrays produced by SBIR is currently 48 μm [14, 15, 17]. Contemporary SBIR research resistor arrays attain a maximum apparent temperature as high as 1400K at maximum frame-rate 500 Hz [18]. In order to overcome these limitations, research began in earnest to discover alternative infrared technologies.

1.3 IRLED Arrays

A notable alternative to resistor array IRSPs instead utilized infrared light emitting diodes (IRLED). IRSPs that use LED arrays have been studied since 2008, beginning as a collaboration between the University of Delaware (UDel) and the University of Iowa (UIowa). An IRLED array has two parts to it, the RIIC, a chip that contains all the electrical pixels that control the radiance output of the LEDs and also provides communication to a system, and the Super-Lattice LED (SLED) array, grown on a gallium arsenide (GaSb) semiconductor wafer using Molecular Beam Epitaxy

(MBE) [6]. The RIIC and the SLED array are hybridized together using flip-chip bonding to form a single part known as a SLED hybrid [4].

IRLED scene projectors can potentially resolve several of the limitations of resistor arrays if they can achieve similar accuracy, reliability, and operability levels [5, 6, 19, 20]. *Operability* is an essential metric of IRSP systems defined as the fraction of emitting elements capable of achieving a specified radiance [5]. IRLEDs operate fundamentally very differently than resistors because emission is the result of the recombination of electron-hole pairs in a semiconductor with a band-gap appropriate to produce the desired wavelength [5, 19, 21]. As a result, the emitter does not need to operate at temperatures anywhere near the apparent temperature, and thus need not also need to be made of materials that can withstand those temperatures. Perhaps even more importantly, since the emission is electrical rather than thermal, it is several orders of magnitude faster [5, 8, 22, 23]. Therefore, the frame-rate of the system is effectively limited only by the RIIC and CSE capability to drive the array emitters [5, 23, 24].

Here at UDel, we have developed an LED-based IRSP that has been shown to be stable and reliable through a variety of HWIL testing configurations at several different laboratories. The current components that make up this complete IRLED IRSP system are depicted in Figure 1, which will be described in detail below [4, 25]. The first steps are scene generation and non-uniformity correction (NUC). Scene generation describes the process by which pixel data representing IR scenes is provided to a system for display. The NUC process corrects for physical and thermal non-uniformity in an IRLED array [26, 27, 28]. The close support electronics (CSE) is responsible for converting the digital output imagery of the scene generator to analog

data to be emitted by the array. Thus, the CSE contains all the hardware necessary to process and format the data so that the read-in integrated circuit (RIIC) component of the display can interpret it as an image [29, 30, 31, 32]. The display LED array, or SLED hybrid, is housed in a commercial off-the-shelf (COTS) Dewar so that the system can be run at cryogenic temperatures using liquid nitrogen (LN_2) - typically around 78 K – which increases LED emission by 5x – 10x and enables drawing accurate scenery near room temperature [33, 34, 40]. For internal development purposes, FLIR Systems Inc. camera models SC6800 or SC8200 are used to gather IRSP research data [30].



Figure 1 University of Delaware IRLED Scene Projector System

1.4 Dissertation Outline, Goals and Contributions

1.4.1 Outline

The following details the organization of this dissertation:

Chapter 2 details the superlattice LED (SLED) IRSP technology in its current state, as well as proposed enhancements for the near future.

Chapter 3 describes the challenges of transferring scene generation data to the CSE due to the limitations of traditional display protocols and briefly describes a novel solution known as the packetized display protocol (PDP).

Chapter 4 explores the hardware side of the scene generator to CSE interface, detailing design and testing of a custom gigabit interface printed circuit board.

Chapter 5 describes the concept of combining multiple CSEs for added functionality and details the design of a next generation CSE called NESSIE2 to meet rising customer resolution and frame-rate demands.

Chapter 6 details a custom distributed CSE design building on all of the concepts covered in Chapters 2-5, and notably describes how such a design enhances a protocol such as PDP.

1.4.2 Goals

The primary goal of the work presented in this dissertation is to explore new design architectures for future CSE systems to enable driving emitter arrays of resolution 4K x 4K and beyond. A related goal of this work is to provide a custom hardware design, which more efficiently and effectively utilizes the novel packetized display protocol currently under development for SLEDs IRSP systems. This dissertation provides the reader an understanding of the CSE in current and future systems in order to aid in the development of the next generation of IRLED IRSPs.

1.4.3 Contributions

- Designed and laid out third-generation custom gigabit interface printed circuit board for SLEDs family of IRLED IRSP
- Tested and demonstrated first-generation PDP protocol on SLEDs IRSP
- Proposed and began implementation of second-generation CSE, NESSIE2
- Proposed novel architecture design of third-generation distributed, modular, multi-FPGA CSE, NESSIE3

Chapter 2

BACKGROUND AND CURRENT STATE OF SLED PROJECTORS

The purpose of this chapter is to briefly describe the evolution of the LED IRSP systems developed and implemented by the University of Delaware in conjunction with the University of Iowa. Special attention is given to the close support electronics (CSE) driving these projectors, as the improvement of the CSE is a central goal of this work.

2.1 Superlattice LED System

The Superlattice Light Emitting Diode (SLED) array was constructed in 2008, making it the first IRLED array in the world [35]. This 68-pixel x 68-pixel array was initially tested by hand as a drive system had not yet been developed. This drive system was developed in 2011. In the years that followed, an IRLED wafer of size 512 x 512 pixels with 48 μm pitch was constructed. This wafer in combination with the drive system formed the first complete and functional IRLED IRSP system in the world, known as the Superlattice Light Emitting Diode System (SLEDS) [36, 37].

2.2 TCSA, NSLED, HDILED

The SLEDS system successfully demonstrated the IRLED IRSP proof of concept. From that point on, research and development of this technology advanced rapidly, culminating in the creation of advanced IRLED IRSPs with greater functionality, resolution, frame-rate, and dynamic range. The Two Color SLEDS

Array (TSCA) was completed in 2016 as a 512 x 512 array that was capable of driving LEDs at two separate bands of wavelengths [3, 22, 24, 38, 39]. Around the same time, the N³ Superlattice LED System (NSLEDS) was developed, featuring a single color 24 μm pitch array with four times the resolution of the original SLEDS system, bringing the total resolution to 1024 x 1024 [22, 35, 41, 42]. Further improvements to the technology led to the creation of High-Definition Infrared LED (HDILED) array in 2018, which was the world's first 2048 x 2048 IRLED pixel array [23, 29, 43]. Figures 2 and 3 show the advancement timeline of the SLED technology.

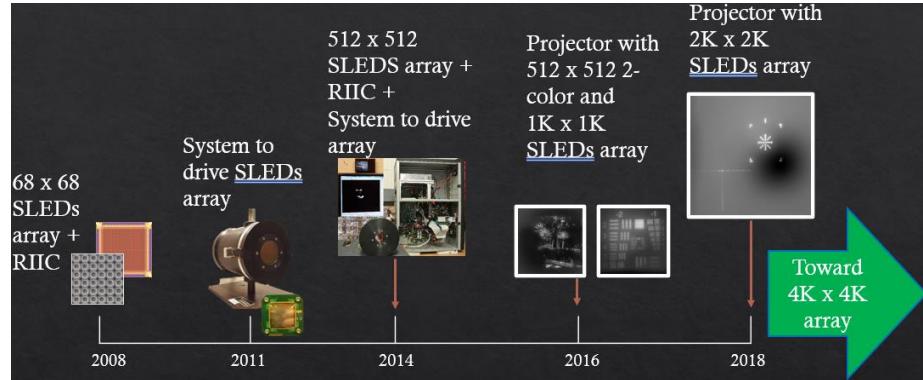


Figure 2 Superlattice LEDs (SLEDs) Projectors Timeline, Part 1

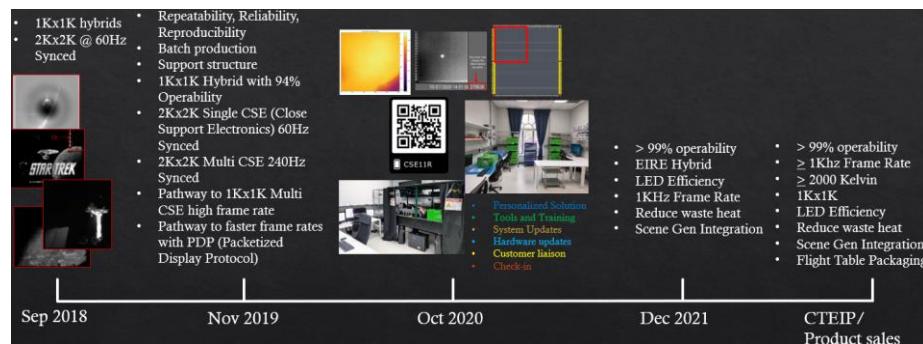


Figure 3 SLEDs Projectors Timeline, Part 2

2.2.1 RIIC Overview

The Read-In Integrated Circuit (RIIC) provided both cathode and anode contacts for every pixel in the LED array. The RIIC is divided into 4 quadrants that can each be controlled separately, with each quadrant requiring between 2 and 32 analog input channels. The number of input channels can be configured by setting the serial peripheral interface (SPI) register.

2.2.2 CSE Overview

The Close Support Electronics of the SLEDs family of IRSP systems consists of a field programmable gate array (FPGA), digital to analog converters (DACs), amplifiers, gigabit I/O printed circuit boards, and a custom printed circuit board to interface all of these components, which we shall refer to as the interface board. The main FPGA board is the Inrevium TB-6V-LX760-LSI, which consists of a Xilinx Virtex 6 FPGA, along with ten low point count (LPC) FPGA mezzanine connectors (FMC) slots to connect to various input/output (I/O) peripherals. Connected to each of eight of these FMC slots are a custom FMC card containing two AD9747 DACs. Each of these DAC cards is connected to an amplifier board containing two TH6012 amplifiers. These signals are then routed to a pair of custom interface boards that provide power to the DACs and amplifiers and send the analog output signals to the SLED hybrid. Additionally, the interface boards convert the 2.5 V digital logic FPGA signals to 5V signals as required by the RIIC [3, 5]. The final two FMC slots contain a pair of gigabit interface cards that are used to collect input imagery from a scene generator to be projected by the IRSP. For the development of TCSA, NSLED, and HDILED these boards were primarily Inrevium TB-FMCL-HDMI boards. Testing and development of gigabit interfaces into the CSE is described in detail in Chapter 4. The

entire CSE system is packaged in a custom 19-inch rack-mountable case called nimble, extensible, and scalable SLEDs infrastructure electronics (NESSIE) [5, 32].

2.2.3 Scene Generation and NUCPC

In order for an infrared scene projector to display IR imagery, it must first be provided the desired scene as an input. This process is known as scene generation. A somewhat imperfect but conceptually useful analogy would be to think of an IR scene projector as similar to a television, whereas a scene projector would be akin to a cable provider or DVD player providing an input into the television. The details of how scene generation is accomplished is beyond the scope of this work and indeed, the exact implementation details of a given scene projector may be proprietary and thus unknown to the IRSP developer [44]. The developers of an IRSP and those of the scene generator instead can work together to decide on the data transfer standard that will be used for communication between the two. Traditionally, this has been accomplished through utilizing well-known existing display protocols such as DVI. An example block diagram of the scene generator can be seen in Figure 4 [51].

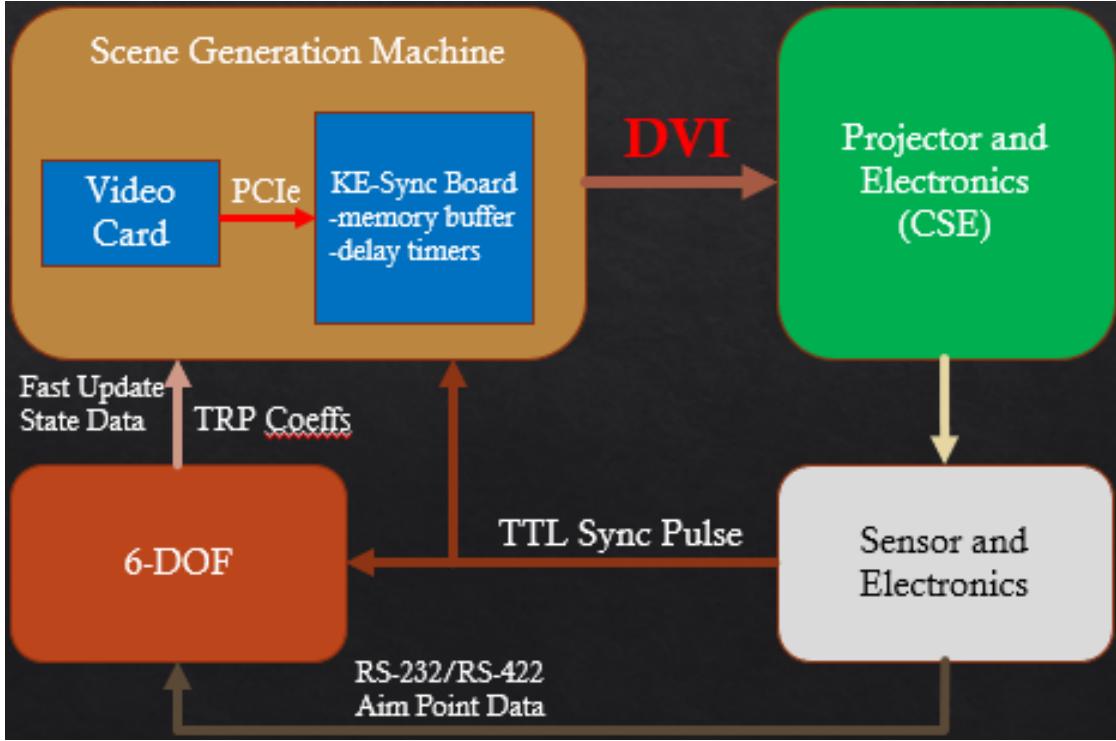


Figure 4 Scene Generation Example

A typical scene generator will provide the IRLED IRSP with a data stream such as video of a given bandwidth as characterized by the desired resolution and frame-rate to be projected to the sensor. The scene generator usually utilizes a GPU to produce the image data, which is sent using a traditional display protocol at a static frame-rate. The next step to display scenes on an IRLED scene projector is called non-uniformity correction (NUC). Non-uniformity correction is done by analyzing the non-uniformity of the IRLED emitter array to produce a NUC table containing a value for each LED pixel in the array. The NUC process is essential to mitigate nonlinear variations in infrared radiation in different LEDs due to physical differences or defects [26, 31, 45, 46, 47]. The NUC process occurs at the same static frame-rate as the

initial scene generation. Finally, after completion of the NUC, each and every pixel of the scene data is converted from digital to analog by the CSE to drive the emitter array pixels at the desired IR intensity [44, 46]. Figure 5 depicts the IRLED scene projection process.

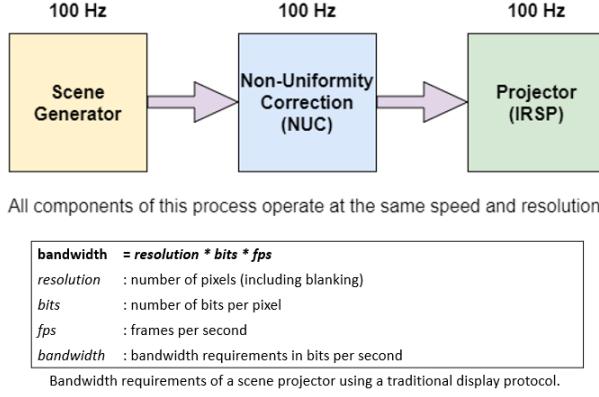


Figure 5 Scene Projection Overview

When developing and testing an IRLED IRSP, researchers often do not have access to a proprietary scene projector. At UDel, we implemented a pseudo-scene generator called a non-uniformity correction personal computer (NUCPC) that sends image data using a GPU and display protocol to the NESSIE CSE. As the name implies, the NUCPC corrects scene data for non-uniformity and sends it to the CSE for display on the array [5]. The current NUCPCs are several years old and use modest technology by today's standards but are still more than sufficient for the needs of programs such as NSLEDS and HDILED. The NUCPC consists of an Intel i7-6700, an Nvidia Quadro K-5000 or M-4000 GPU, an Nvidia Quadro Sync card, an Engineering Design Team (EDT) F-4 Camera Link capture card, and multiple front

mounted solid-state drives (SSDs). The EDT F-4 is used for the collection of NUC data. The front-mounted SSDs allow for easy upgrades to the system simply by replacing the SSD with the upgraded one. The NUCPC sends the scene data from the GPU to the gigabit interface boards in the CSE using two HDMI 1.4 links. In order to synchronize these two HDMI links to each other, as well as potentially to an outside source, the Nvidia Quadro Sync card is utilized. Finally, the NUCPC contains two Datapath VisionLC-HD cards that are used to loop back scene data to visually detect problems in the video output. These cards can also potentially enable connecting the NUCPC to an external scene generator [5, 28].

2.3 AIREA, HSLEDS, and Future Systems

Development of IRLED-based IRSPs continues as researchers tackle the challenges of improving the technology as system complexity increases. One challenge is the fact that RIIC parts larger than square-inch size result in poor yields [48, 49]. As a result, arrays of size 2K x 2K or larger, such as HDILED systems, are operational for testing and proof of concept purposes, but do not yet achieve the operability necessary to qualify for formal Test and Evaluation (T&E) requirements for IR system development. One way to mitigate this issue is to move to a smaller pixel architecture, as described in *Higher Definition Mid-Wave Infrared Scene Projectors via Shrinking Pixel Pitch* by Miguel Hernandez-Raya [4]. This is the basis for the HSLEDS program that has recently completed Small Business Innovation Research (SBIR) Phase I. Another method used in the Advanced Infrared Emitter Array (AIREA) program aims to produce two 24 μm pitch 1K x 2K rectangular arrays and combine them into one 2K x 2K array by carefully aligning them together [14, 41, 50]. Details on this process can be found in the Ph.D. dissertation by Josh Marks [14].

Chapter 3

CONVENTIONAL DISPLAY PROTOCOLS AND PDP

The previous chapters describe the evolution and use cases of infrared scene projectors. Traditionally, scene generators that send the data to be projected by the IRSP have utilized traditional display protocols such as DVI. These protocols have a number of limitations that will be described below. A novel solution to these limitations called the packetized display protocol will be described in this chapter.

3.1 DVI and HDMI

Early scene generators by customers such as Air Force Research Laboratory (AFRL) were designed to utilize the Digital Visual Interface (DVI) interface to send image data to a prospective IRSP [51]. The DVI standard is well understood and ubiquitous, making it a reasonable early choice. DVI connectors are a bit bulky and unwieldy compared to the size of the components in the CSE however, so early on it was decided to instead use High-Definition Multimedia Interface (HDMI) connectors for the CSE's gigabit interface. As far as video data is concerned, the DVI and HDMI interfaces are electrically equivalent, making our systems compatible with the existing scene generator technology [52].

3.2 Limitations of Conventional Display Protocols

The traditional display protocols currently used in the IRSP community, such as DVI and HDMI, have a number of drawbacks that are perhaps undesirable for scene generation in future systems. These protocols were designed to accommodate and be

compatible with older connections like video graphics array (VGA), which was based on the raster scan technology such as that found in cathode ray tube (CRT) displays. These devices begin by sending video data to the top left pixel and then writing video data to each pixel in a horizontal row one at a time, then repeating this process for each row until reaching the end of the display frame. In order to account for the time required for the original devices to reset to the start of each row and each frame, blanking periods were used in both horizontal and vertical directions. No video data is displayed during these blanking periods, which contain a sync pulse to signal the end of a row or frame. The time between the last visible pixel in a row or frame and the start of the sync pulse is called the front porch. Similarly, the time from the end of the sync pulse to the start of the next row or frame is known as the back porch. As a result of these blanking periods, the displayed video resolution is smaller than the number of data pixels sent along the physical link. Figure 6 shows a traditional video frame.

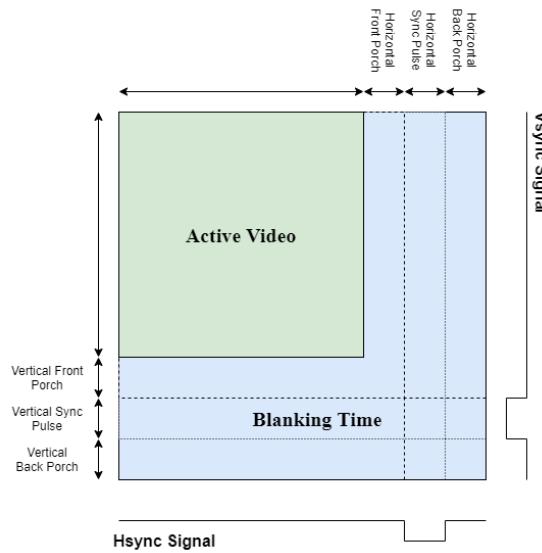


Figure 6 Traditional video frame with horizontal and vertical blanking.

A display protocol like HDMI not only maintains these blanking periods to remain compatible with older technology, but also uses this time to send out-of-band information such as audio. A *pixel clock* is the clock rate at which each pixel is sent along the data stream. The video frame-rate is determined by all of the pixels in the frame – including the blanking periods – as well as the pixel clock such that $\text{frame-rate} = \text{pixel clock} / (\text{total horizontal pixels} * \text{total vertical pixels})$. In other words, the frame-rate is inversely proportional to not just the active video region resolution but the total frame size. Any steps that can be taken to reduce or eliminate the blanking periods would be greatly beneficial to the goal of achieving higher display frame-rates. As an aside, here we shall define the term *modeline* as the set of parameters describing how to drive a display device at a specified video resolution [44, 53, 54]. A modeline consists of the pixel clock and parameters defining the end of the active horizontal region, the start of the hsync pulse, the end of the hsync pulse, and the end of the back porch, and then the same for the parameters of the vertical region.

Another potential drawback of these display technologies is that they assume a fixed frame-rate display. Transfers of frame data occur at predetermined intervals, and if a new frame is unavailable to be transmitted, the previous frame is retransmitted [44]. Moreover, due to the proprietary nature of conventional display protocol hardware and drivers, these frame-drops are effectively silent [55]. Once the modeline is used to initialize the display device, the frame-rate cannot be controlled or changed without re-initialization. As a result, these protocols dictate a static bandwidth for a given frame-rate and resolution [55]. Even in cases where data within a frame is sparse or when only small portions are experiencing change, protocols like these only support transferring the entire frame at a given time interval. IRSP scenarios often use sparse

grid NUCs for correction, so sending entire frames when the region of interest is only a small part of a frame is not the most efficient use of the available bandwidth [56].

3.3 Packetized Display Protocol

A custom protocol named the packetized display protocol (PDP) has been developed for the communication link between a scene generator and the CSE to address the limitations of conventional display protocols. As defined by the original PDP architect, PDP is “a physical layer agnostic method of pixel data transmission capable of providing and addressing latency guarantees while maximizing bandwidth utilization” [55]. A very brief description of how PDP works follows. The first step of the protocol is to split the scene into logical sub-frames, also called *regions*, potentially prioritizing regions with more or faster changing data. A packet header is assigned to each of these regions with the start and end x and y coordinates of the region to be drawn. A PDP packet is formed from the packet header followed by the data to be drawn to the region. Table 1 gives some examples of PDP headers.

Table 1 Examples of PDP Packets

Name	Type ID	Type Specific Fields					
Ignored	0x0						
Draw Region	0x1	X start	X end	Y start	Y end	Data	
Array Reset	0x2	Quad mask					

This scheme provides the ability to draw these sub-frame regions at a faster apparent frame-rate. The PDP packets can be decoded and displayed anywhere on the array in a random-access order [55, 57]. If a particular region or set of regions is changing faster than the others, then the packets for those regions are simply sent more often. The

PDP protocol refers to this as *stream mode* [57]. However, support for sending full frames at the original frame-rate akin to traditional display protocols is maintained. This is accomplished by using a packet header with the x and y start and end coordinates being the same as the start and end of the frame, followed by the pixel data for the frame – this is referred to as *backwards compatibility mode* [58]. Figures 7 and 8 depict the PDP modes of operation.

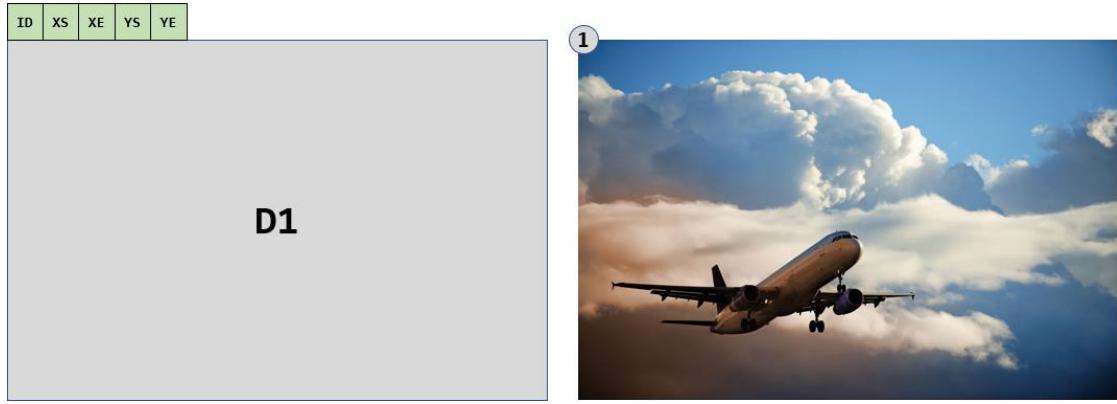


Figure 7 PDP Backwards Compatibility Mode

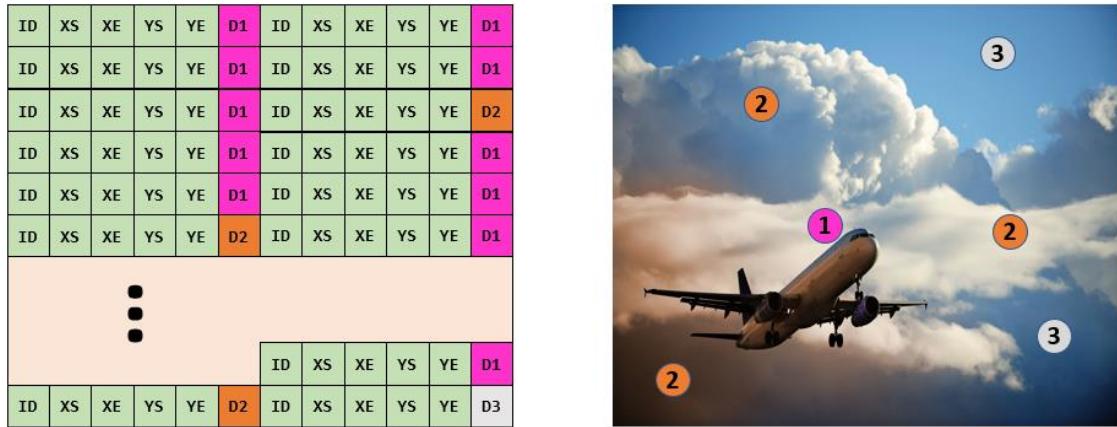


Figure 8 PDP Stream Mode

As specified in the original definition, PDP is “physical layer agnostic”, meaning that it could conceivably be implemented using any data transfer medium such as DVI, ethernet, or fiber optic cables to name a few examples. The first successful version of PDP by Udel researchers was implemented utilizing HDMI data links. This decision was made in order to maintain compatibility with the existing NESSIE CSE and scene generators. For a more in-depth description of the design and architecture of PDP, please refer to Aaron Landwehr’s Ph.D. dissertation [44].

The following sections 3.3.1 – 3.3.3 are a summary of the author’s work in obtaining experimental results of the first generation of the PDP protocol implemented over HDMI using the NSLEDS IRSP array [58, 59]. These sections present updates to the first-generation PDP architecture including scaling the number of inputs, handling bit-errors, a backwards compatibility mode, and modes of recovery. Also, a comparison is made with the previous firmware, and the latest results driving SLEDs arrays are shown.

3.3.1 Features

The first prototype implementation of the PDP decoder was done as firmware on the TB-6V-LX760-LSI board’s Virtex6 FPGA. This firmware decodes the PDP packet headers in order to write data to the array at the designated locations and pixel refresh rates. The first version utilized a single input stream as a proof of concept. Newer revisions include a new two input mode developed within the PDP firmware in order to increase the available hardware bandwidth and allow for the addition of a backwards compatibility mode for the existing SLEDs firmware. This mode utilizes two HDMI hardware links in order to allow two simultaneous PDP streams to be active. The components of the PDP decoder are shown in Figures 9 - 12.

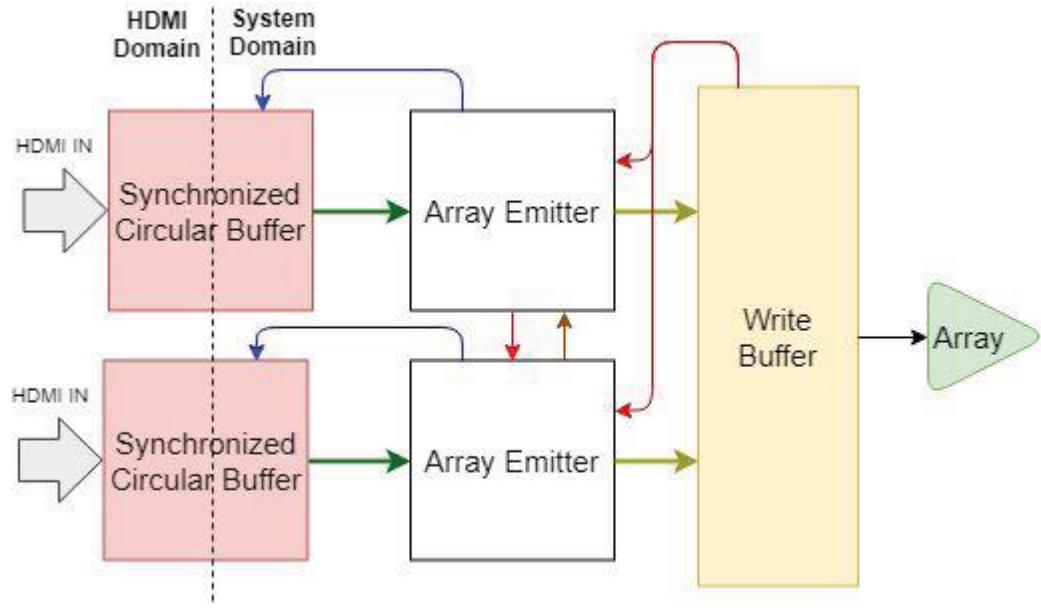


Figure 9 Simplified First-Generation PDP Architecture

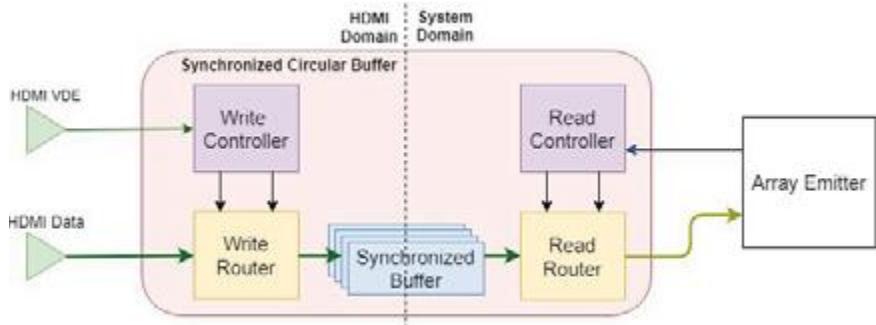


Figure 10 Simplified Synchronized Circular Buffer (SCB)

Bit-errors in data transmission would corrupt the PDP state machine, so a cyclic redundancy check (CRC) pixel was added to both the packet header and packet data to handle this. The word-sized CRC is stored in a register for comparison. As each word of the header data is processed, an internal CRC is computed, and the final value is checked after the full header is received. If the CRC does not match, the

packet is then discarded. The same process occurs for packet data. A CRC for the data is stored in a register. Then as the packet data comes in another internal CRC is computed and the final value is checked after all of the data is received. If the CRC does not match, the packet is discarded, and an error flag is set. The PDP state machine has been changed to reflect the addition of the CRC as shown in Figure 12. For Draw Region packets separate states were added to load the CRCs embedded in the data stream and logic was added to check the CRCs and discard packet data when a mismatch occurs.

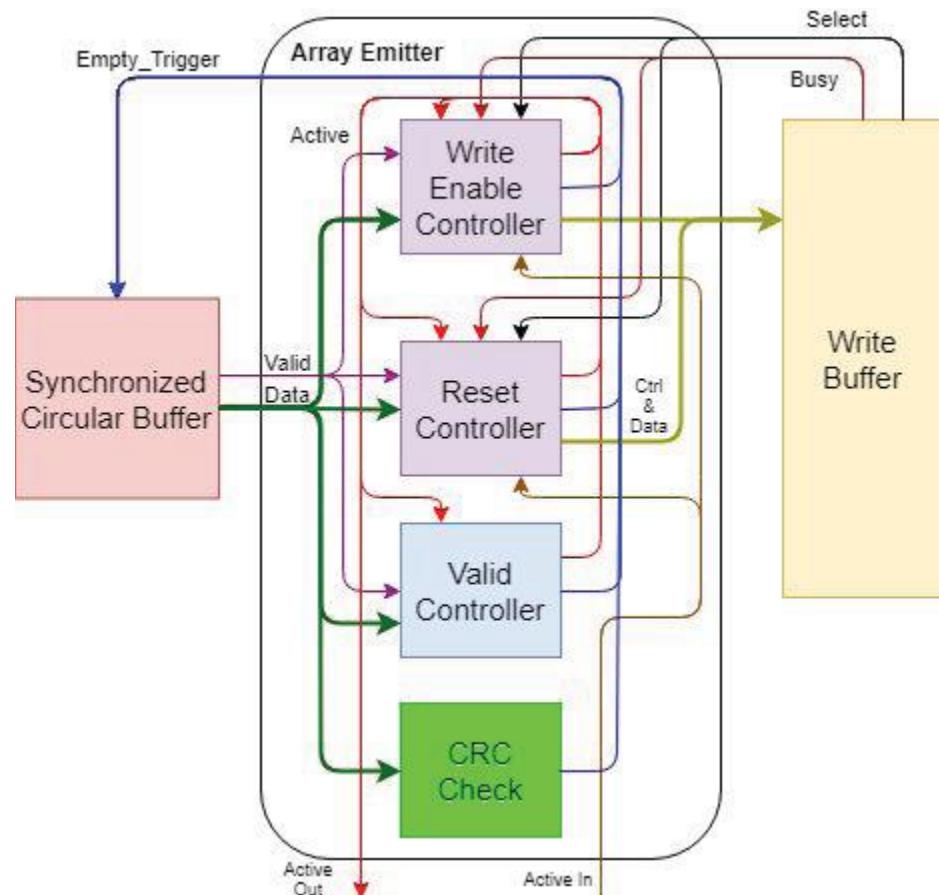


Figure 11 Overview of Array Emitter.

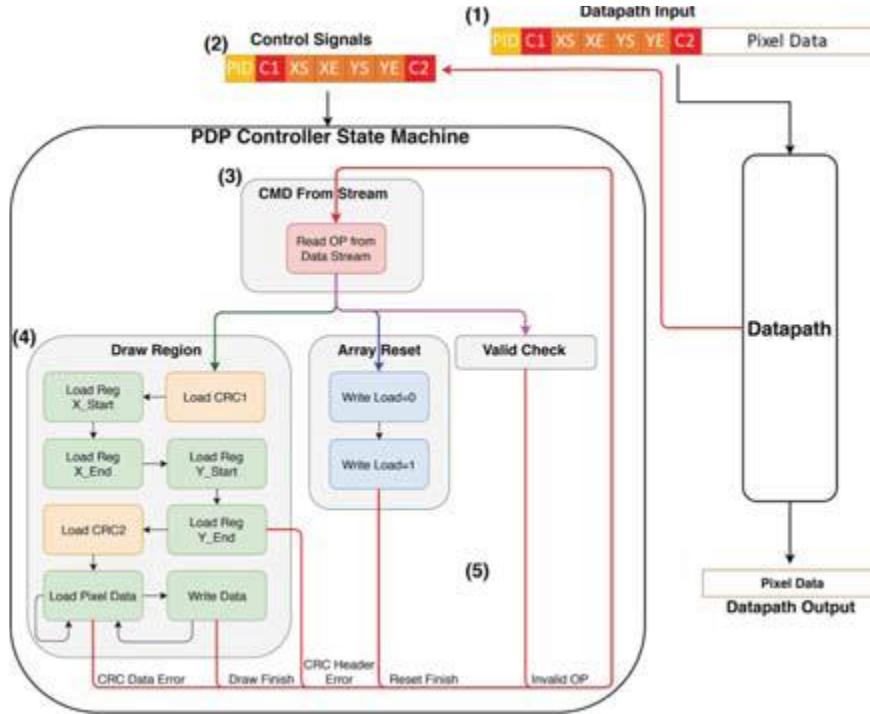


Figure 12 PDP State Machine

As alluded to earlier, one design requirement for PDP is that it be backwards compatible with a traditional DVI/HDMI video stream as in [29]. With this added ability users of PDP could run HDMI video streams to the SLEDs projector with the packet's header set by software registers to be the size of a full video frame (depending on resolution), and eliminate the overhead associated with the packet header allowing for the full link bandwidth to be used just on packet data.

PDP can end up in an undefined state if the data link has a bit-error in various points of the packet decoding process. In order to recover from undefined states, PDP offers four methods of resetting to a known state. The first method allows the user to enter a set time between packets, i.e. 3 seconds, which if exceeded will reset the state machine and clear the circular shift buffer. The second method is with a reserved

command ID which if detected by the circular shift buffer will empty and reset the state. The third is after a detected CRC error either on the packet header or packet data. Lastly, the user also has the ability to set a software register to reset the system.

3.3.2 PDP vs SNAP

Table 2 A comparison between Snap firmware and PDP firmware.

	Snap	PDP
Refresh Rate	Fixed	Dynamic
Bandwidth	Fixed	Dynamic
Scalability	2x	N
Timing	Fails	Passes
Synchronization	Fixed	Controllable
Number of Windows	1	Frame Size

To see the benefits of PDP over the previous firmware (Snap) Table 2 gives a holistic picture.

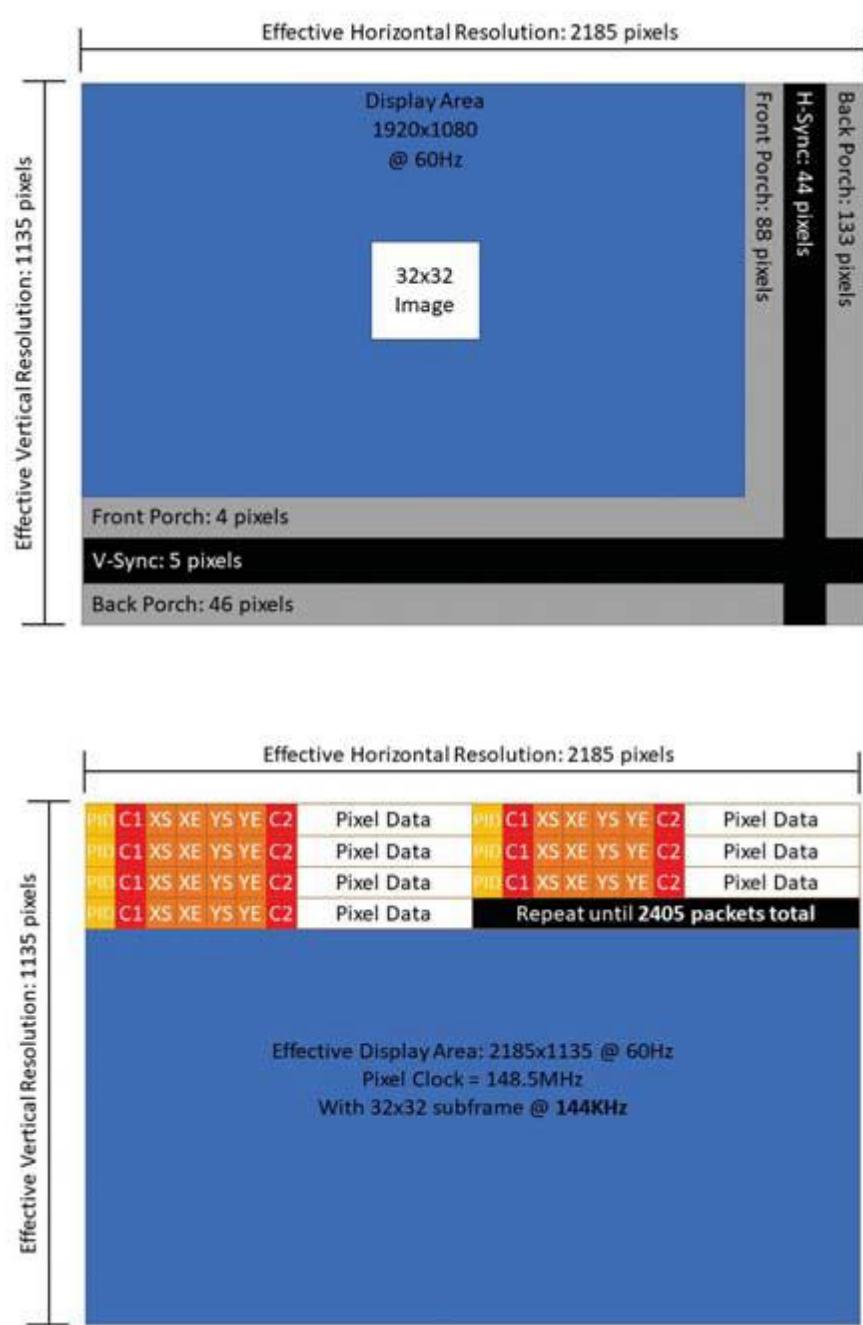


Figure 13 (Top) HDMI frame running at 1080p displaying a static square image of size 32x32 at a frame rate of 60 Hz. (Bottom) An equivalent PDP frame operating with no blanking regions and maximal packing PDP data packets for an effective window of size 32x32 running at a frame rate over **144 KHz**.

Snap is limited to fixed frame-rates and bandwidth per video frame, can only scale to two HDMI inputs per CSE, and has known metastability issues with data transfer from the video clock domain to the FPGA system clock domain. Along with this, Snap has no fine-grained control over when frames are displayed and how they are synchronized. Snap only has the ability to display one window per frame (full-scale or sub-window). Conversely, PDP can have dynamic refresh rates, intelligent bandwidth utilization, and can easily scale beyond the capabilities of the current SLEDs RIIC architecture making it future-proofed for new arrays. PDP has been simulated and verified to ensure no metastability issues exist and all timing in the firmware is met. Sub-windows have the ability to be displayed at any time within a traditional HDMI frame as well as when they are synchronized within a frame. PDP is abstracted away from traditional video modelines and is limited only by the pixel clock driving the video stream. Figure 13 depicts the differences between a traditional HDMI frame and a PDP frame packeted within an HDMI frame.

3.3.3 Results

PDP has been used to drive an NSLEDs array with a single HDMI input. Several test patterns have been used such as the one in Figure 15 which shows the ability PDP has to drive multiple windows at different rates (30 Hz and 60 Hz) while only using a small fraction of the available bandwidth. A software wrapper has been added to the existing control software (PING) to convert traditional HDMI frames to PDP packets which allows for the entire display to be driven by one input. With these features combined, PDP was capable of theoretically reaching rates of 3.2 MHz for a 16x1 pixel sub-window when running a modeline with a pixel clock of 118.4 MHz.

Figure 14 demonstrates a window driven in backwards compatibility mode displaying full-frame PDP output at 100 Hz.

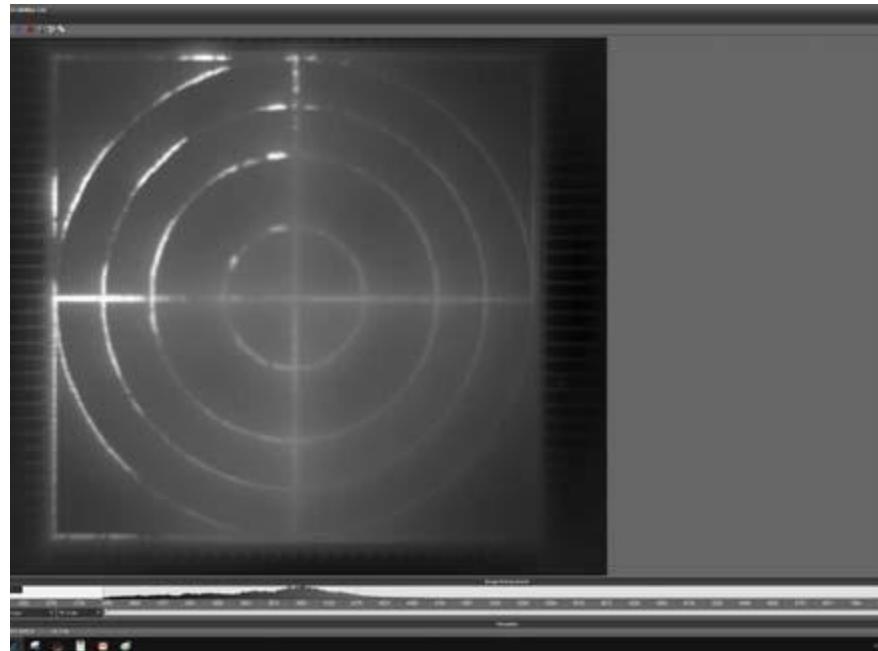


Figure 14 Example showing full-frame PDP output of a cross-pattern input image in backwards compatibility mode, running at 100 Hz.

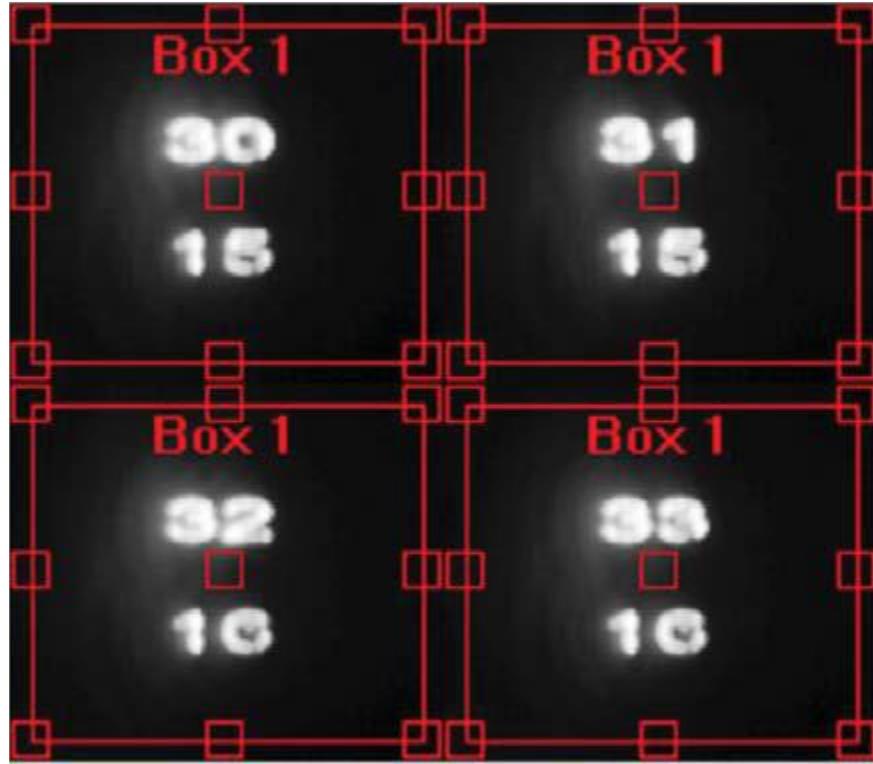


Figure 15 Example dual-window test displaying numbers counting from 1 to 60 at 60 Hz and 1 to 30 at 30 Hz with 32x32 size sub-windows.

3.3.4 Future PDP Considerations

In the months following the presentation of the test results described in the sections above, a second generation of PDP has been developed with several enhancements including improved synchronization methods, benchmarking of the firmware, and superior error detection. PDP version 2 (PDPv2) is beyond the scope of this work, but it can be studied in detail via Tyler Browning's Ph.D. dissertation [60].

The prototype implementation of PDP was done using HDMI as the physical transport layer. The packetized nature of the protocol enables updating sub-frame regions of a scene at differing frame-rates, thus overcoming the static frame-rate limitation of the traditional HDMI protocol. However, by utilizing HDMI for the

physical layer, unfortunately the drawbacks of the blanking periods remain problematic. This is partially due to the hsync and vsync pulses that occur during these periods, which HDMI uses to determine when to advance to the next row or frame. During the testing of PDP, custom modelines were written in an attempt to minimize the size of the front and back porches. Through trial and error using a custom script, it was determined that certain minimum porch sizes were required, or the receiver would misinterpret the display resolution and/or frame-rate, or the display would fail altogether. As it turns out, HDMI sends control tokens during the blanking periods which are used by the receiver to decode the transition-minimized differential signaling (TMDS) HDMI data signals. This in turn determines the minimum required blanking period necessary to recover the data. Figure 16 shows the TMDS link timing and parameters [61].

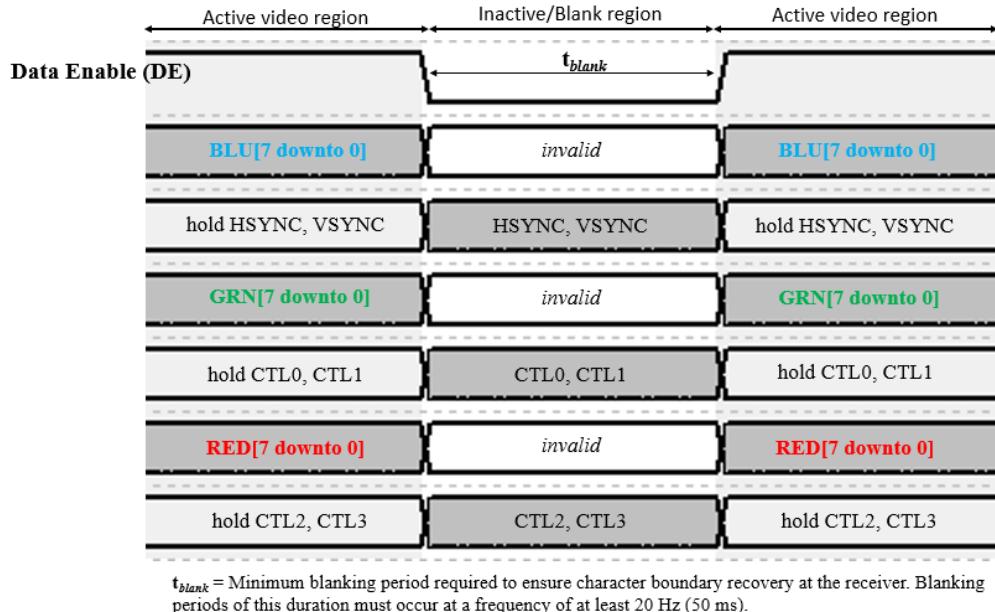


Figure 16 Minimum Blanking and TMDS Timing Parameters

Although the initial PDP version proved that it was possible to implement the protocol over the HDMI physical layer, this is perhaps not the best choice due to limitations such as blanking periods. Using a different transport medium for the data stream between the scene generator and the CSE would be beneficial for PDP but would require significant changes to the current hardware architecture. The remainder of this work dives into the hardware of the CSE. Chapter 4 describes the current NESSIE CSE and pays special attention to the HDMI gigabit I/O interface. Chapters 5 and 6 describe proposed architectures for future CSEs, where a significant driving factor is choosing physical transport layers to leverage the full power of the packetized display protocol.

Chapter 4

CLOSE SUPPORT ELECTRONICS AND GIGABIT INTERFACE

In Chapter 2 the components of the state-of-the-art IRSP systems developed by CDS were introduced and in Chapter 3, a novel packetized display protocol (PDP) developed to overcome traditional display protocol limitations was discussed. In this chapter, a more detailed description of the close support electronics (CSE) of the SLEDs family of IRSPs is given. Special attention is given to how data flows through the CSE from the input (from a scene generator) to the output (to the array). This flow of data will henceforth be referred to as the *datapath*. In order to support and demonstrate the capabilities of the PDP technique, a custom gigabit interface was developed. The remaining sections of this chapter provide an in-depth explanation of the design and implementation of this gigabit interface. An account of the troubleshooting, testing, and results of the revisions of this interface is also provided.

4.1 Components of CSE and Datapath

The CSE consists of the following components: two gigabit interface input cards, a main board containing a field programmable gate array (FPGA), eight digital-to-analog converters (DACs) cards, eight amplifier cards, and one custom interface printed circuit board. The gigabit interface cards are the inputs into the CSE system from an external scene generator and will be discussed in detail in Section 4.2. The CSE uses a Xilinx Virtex-6 FPGA contained on a COTS Inrevium TB-6V-LX760-LSI PCB containing ten low pin count FPGA Mezzanine Card (FMC) slots. The DACs are

AD9747 evaluation boards, each consisting of two channels. Each of these DACs is paired with a TH6012 op-amp, and the combination of the DAC and amp converts the digital signals on the TB-6V-LX760-LSI to the proper analog signals to be interpreted by the RIIC. The interface board then routes the signals from the CSE to the RIIC. The signals are sent along 3M 3302 Series ribbon cables consisting of 28 AWG wires on 0.050-inch centers with propagation delay of 1.47 ns/foot.

A high-level view of the datapath for the CSE is shown in Figure 17. After a video input is provided from a scene generator, the data is decoded by the gigabit interface card and sent to the Inrevium TB-6V-LX760-LSI through two low pin count FMC slots. In the current generation of the CSE, one of the inputs ultimately addresses the top half of the array, while the other input writes to the bottom half. The data is then processed by the TB-6V-LX760-LSI Virtex 6 FPGA, on which a MicroBlaze soft processor has been instantiated. The soft processor allows for control using a traditional software stack, with applications written in C and the ability to define and change several control registers [62]. The FPGA bit packs the digital data and sends it in the proper format to be displayed on the array to the eight remaining FMC slots on the TB-6V-LX760-LSI board. A PCB containing two two-channel AD9747 DACs is connected to each of these eight FMC slots for a total of 32 channels. Paired with each of these DAC boards is a custom card containing two TH6012 amplifiers. All of signals are routed to two custom interface boards, which powers the DAC and amplified boards and sends the digital and analog signals which are routed to the SLEDs hybrid. These interface boards convert the digital logic level from 2.5V from the TB-6V-LX760-LSI to the 5V level that the RIIC requires.

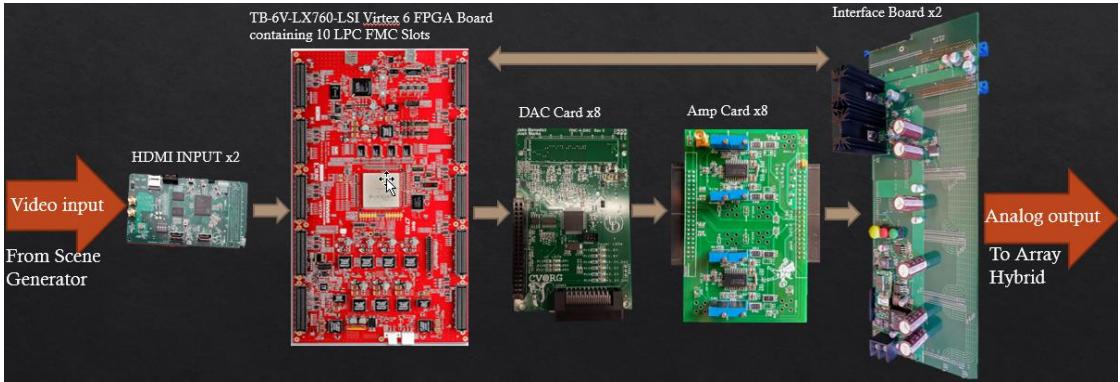


Figure 17 CSE Datapath

4.2 Development of Gigabit Interface

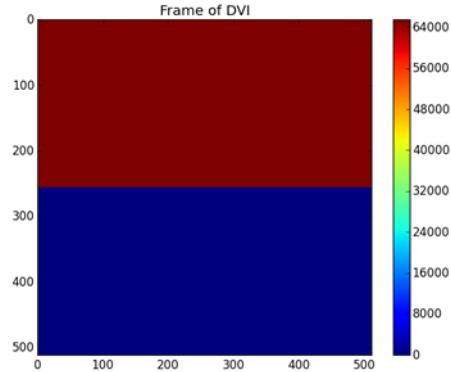
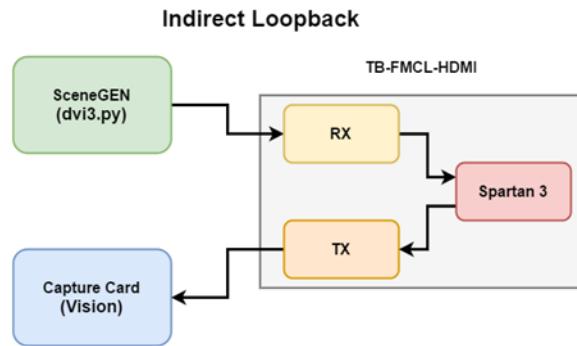
4.2.1 Motivation

Before the concept of PDP was developed and implemented, the initial solution for the CSE gigabit interface with potential scene generators was a COTS Inrevium TB-FMCL-HDMI board. This board consisted of a single dedicated HDMI input port and a single dedicated HDMI output port. HDMI was chosen because it is electrically compatible with DVI, which was a video communication standard used by scene generators at the time. The TB-FMCL-HDMI card could be connected to the rest of the CSE through a low pin count (LPC) FMC connector. This board also contained a Xilinx Spartan 3 XC3S400AN-5FGG400 FPGA, dedicated Analog Devices HDMI decoder and encoder chips, as well as JTAG and Flash configuration options. This PCB was made up of a total of eight layers.

The COTS Inrevium board described in the previous paragraph was serviceable for initial functionality, but a number of problems became apparent as time went by. First and perhaps most importantly, the TB-FMCL-HDMI board was a board that contained older technology in order to be compatible with the TB-6V-LX760-LSI

at the heart of the CSE. This meant that in a few short years, it would become obsolete and Inrevium would no longer produce more board or provide additional support. Secondly, the board contained a single dedicated input and single dedicated output for the HDMI signals. However, the CSE requires only HDMI inputs, so it would be preferable if it instead had two dedicated inputs, for example. Lastly, when testing these boards at very high frame rates (>400 Hz) and nonstandard frame resolutions, some data corruption was observed. These kinds of bit errors were not acceptable to the customers, so some work needed to be done to resolve that, which will be described in some detail below.

A significant effort was spent in troubleshooting the Inrevium TB-FMCL-HDMI boards to overcome the bit errors observed at high frame rates. During this process we worked closely with a very helpful engineer from Inrevium named Kazuhiro Nishimora, hereafter referred to as Kazu for short at his request. At the time, we were testing using a method known as a *BRAM pull*, where we directly read the data stored on the Block Random Access Memory (BRAM) of the FPGAs on the TB-FMCL-HDMI or TB-6V-LX760-LSI boards and compared the data to the expected data that was input from our scene generator. We found that even when performing a simple indirect loopback through the TB-FMCL-HDMI board, the BRAM would show occasional bit errors, even before the data reached the rest of the CSE system. Figure 18 below illustrates the indirect loopback procedure as well as example images of the BRAM pull. The frame rate for this data was 400 Hz.



	A	B	C	D	E	F	G	H
1	65528	65528	65528	65528	65528	65528	65528	65528
2	57336	65528	65528	65528	65528	65528	65528	65528
3	57336	65528	65528	65528	65528	65528	65528	65528
4	57336	65528	65528	65528	65528	65528	65528	65528
5	65528	65528	65528	65528	65528	65528	65528	65528
6	65528	65528	65528	65528	65528	65528	65528	65528
7	65528	65528	65528	65528	65528	65528	65528	65528
8	65528	65528	65528	65528	65528	65528	65528	65528
9	65528	65528	65528	65528	65528	65528	65528	65528
10	65528	65528	65528	65528	65528	65528	65528	65528
11	10616	65528	65528	65528	65528	65528	65528	65528
12	504	65528	65528	65528	65528	65528	65528	65528
13	65528	65528	65528	65528	65528	65528	65528	65528
14	65528	65528	65528	65528	65528	65528	65528	65528
15	65528	65528	65528	65528	65528	65528	65528	65528
16	65528	65528	65528	65528	65528	65528	65528	65528

Figure 18 TB-FMCL-HDMI Indirect Loopback and BRAM Pull

Data errors even before reaching the remainder of our CSE hardware were obviously unacceptable, so we reached out to Inrevium to help troubleshoot the TB-FMCL-HDMI board, because according to their documentation these cards should have been capable of maintaining the bandwidth at 400 Hz [66]. We found that the data problems were even worse at 1000 Hz, as shown in Figure 19 below. We developed an automated method of performing the BRAM pulls and find the optimal HDMI modelines for each non-standard resolution and frame-rate [60]. With the assistance of Kazu, we were able to acquire some proprietary I2C register settings for the TB-FMCL-HDMI board that helped solve the data error issue.

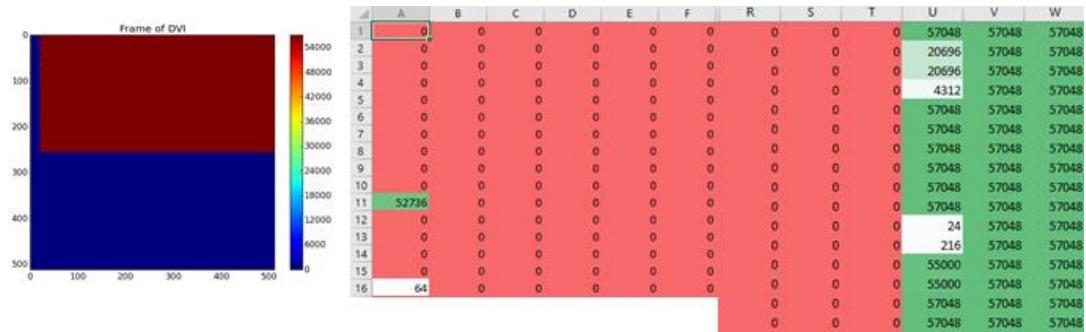


Figure 19 1000 Hz BRAM Pull.

4.2.2 Design

A custom gigabit interface HDMI PCB board to address the various issues discussed in the previous section was developed. This design was based around a relatively inexpensive ZYNQ FPGA similar that found on the Digilent ZYBO evaluation board. One of the most significant design factors was to create a board that could act as a drop-in replacement for the TB-FMCL-HDMI Inrevium board that were used in the existing CSE. This affected decisions on form factor, physical size,

operating voltage levels, connector locations, and so on. Another design principle was to replace the dedicated HDMI input and output on the TB-FMCL-HDMI with two dual-role (Source/Sink) HDMI ports. This would allow for the custom board to match the existing functionality for loopback testing if necessary, while also enabling potentially double the input bandwidth by configuring both of the ports as inputs.



Figure 20 First-Generation HDMI Board.

This custom HDMI interface board was designed around a ZYNQ 7000 series FPGA, rather than the older Spartan 3 FPGA found on the previous TB-FMCL-HDMI card. In addition to being rather inexpensive, the ZYNQ architecture has the special feature of containing both traditional FPGA logic fabric as well as a dedicated portion of the chip set aside as an ARM microprocessor. These sections of the so-called ZYNQ System-on-Chip (SoC) are known as the programmable logic (PL) and processing system (PS) respectively. The ZYNQ SoC has the capability of having a

traditional software stack running on the built-in processor on the PS side, which could seamlessly communicate with peripherals implemented using configurable logic gates on the PL side [63]. Communication between the PS and the PL is accomplished utilizing the Advanced eXtensible Interface (AXI) specification. AXI is a synchronous, parallel multiple master and multiple slave interface that is part of the ARM Advanced Microcontroller Bus Architecture [64]. Figure 21 depicts the ZYNQ SoC architecture. Although the initial design was meant to be a drop-in replacement for the TB-FMCL-HDMI, by choosing the ZYNQ architecture, it opened up the door for more complex calculations and functionality directly on the interface card for future designs. For example, bit-packing, transposition, or even NUC calculations could potentially be off loaded onto the ZYNQ SoC. In a later revision of the HDMI board, a lightweight, custom pseudorandom binary sequence (PRBS) module to check for bit errors is implemented using the ZYNQ SoC [60].

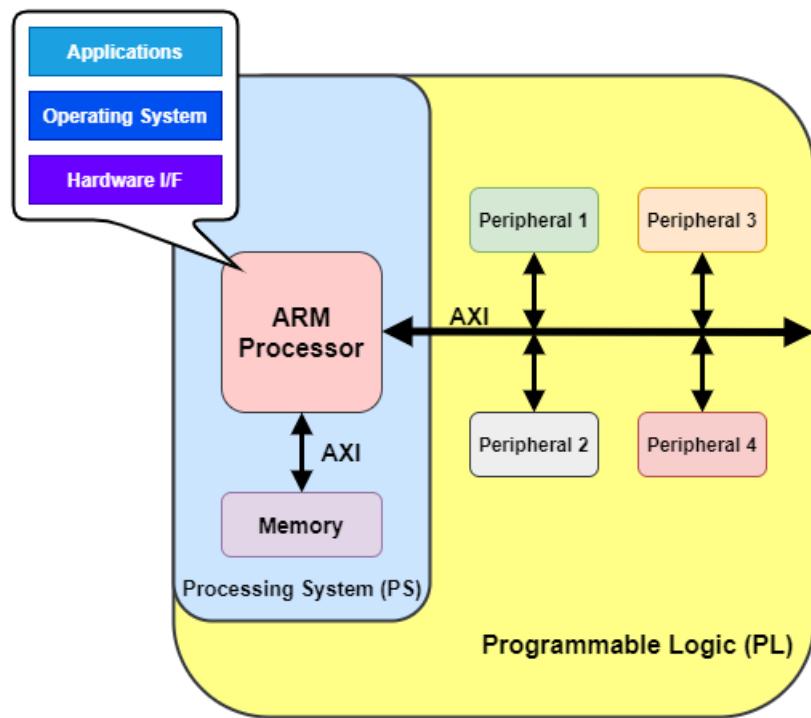


Figure 21 ZYNQ Architecture



Figure 22 Second-Generation HDMI Board

Leveraging the knowledge gained from our initial attempt at an HDMI card design, we took the interim step of designing a second-generation HDMI card as shown in Figure 22 above. This version of the board was primarily intended for pedagogical purposes rather than installation into any existing system. It was used as a testing ground for design ideas and routing techniques, such as a different way to design the power distribution and practicing the tricky task of laying out the DDR RAM chips with limited space. This board was heavily influenced by the Digilent ZYBO board which has been used in several classes at UDel to teach FPGA firmware design and concepts. As such, one of the visions for the second-generation custom HDMI board was to give students a chance to essentially produce a PCB replacement for the ZYBO for their own projects. While this ultimately proved to be an ambitious goal for a single semester course, many lessons were learned both by the students and the instructors during the process.

In the context of the SLEDs systems, the purpose of the second-generation HDMI board was to test ideas for different design paths from the first-generation board, to implement features missing from the first board such as RAM, and to correct some of the problems encountered during the initial design testing. The first major difference between the two designs was the implementation of the power scheme. The first-generation board used a chain of power chips to provide the various digital voltage levels utilized by the card's different logic levels, such 3.3 V and 5 V for HDMI and 2.5 V for ZYNQ FPGA logic, for example. However, we found that startup behavior of these boards was sometimes inconsistent. We found that upon diving into the ZYNQ documentation, the order of powering on each voltage rail was critical to avoid potential race conditions or rail collapse [65]. For the second-generation board,

we opted for a scheme similar to that employed by the ZYBO, where only a single power IC was used to regulate the different voltage levels. Although this design option led to more consistent power up behavior, it had the drawback of introducing a single point of failure so that if the power IC were to fail, the entire board would fail.

The next major goal of the second-generation board was to implement non-volatile memory such as flash or SD card so that the system would maintain its firmware programming after a power cycle. Due to the implementation details of the Xilinx ZYNQ IP core, we discovered it was necessary (or at least much easier) to also implement some form of DDR RAM memory on the board for the flash and SD card to correctly operate. This was because the ZYNQ processing system assumes the existence of DDR memory during its startup process where it initiates the peripherals such as flash, SD cards, and so on. Since we needed non-volatile memory and RAM storage might be useful if we wanted to implement other features, we opted to include both in the second-generation design. The DDR chips used in the design were moderately difficult to lay out on the PCB, and because of this we ended up increasing the number of layers from 8 to 10 to compensate. This is only 2 more layers than the Inrevium TB-FMCL-HDMI board, which does not include any SDRAM options [66].

4.2.3 Implementation

Building on the lessons learned from the previous revisions of the gigabit interface board as described in the previous sections, we began development of the third and final revision, called the ZYNQ-FMCL-HDMI rev3 board. This board was meant to be used in our active CSE systems as a drop-in replacement for the Inrevium TB-FMCL_HDMI cards, as well as allow for future expanded functionality with the

ZYNQ SoC and on-board DDR RAM memory. This section describes the design methodology for this PCB as well as the testing and results of its use in the system.

ZYNQ-FMCL-HDMI rev3 is a ten-layer printed circuit board centering around a Xilinx 7000 series ZYNQ XC7Z020-1CLG484C FPGA. It is a gigabit I/O board meant to take in input data from a scene generator or NUCPC using up to two Molex 0476591002 HDMI connectors. The HDMI data is decoded by a firmware IP on the FPGA, then the scene data is sent to the rest of the CSE in the correct format for the array through a Molex WM9729CT-ND low pin count FMC connector. This PCB also includes a pair of 2 Gb DDR RAM chips, a JTAG header, an SPI Flash chip, and a Micro SD card connector, all of which support volatile and nonvolatile programming options as well as providing headroom for future functionality. The majority of the components are located on the top layer of the board, but some are placed on the bottom due to necessity or convenience, such as the FMC connector and SDIO port expander. The second and ninth layers are dedicated ground planes stapled together by vias so as to be at the same ground potential. The fifth and sixth layers are power planes through which the various voltage rails are supported. The top and bottom layers contain the majority of the signal traces for the components on those layers. Layers three and four were typically used as additional signal layers to support the top layer components, and likewise layers seven and eight primarily for the bottom layer components. Important considerations such as minimum recommended track width, maximum track length, minimum track separation, via sizes, differential signal length matching were determined by consulting Xilinx User Guides on ZYNQ 7000 series PCB design and ball grid array (BGA) device design, programs that calculate different

PCB parameters, and PCB manufacturer minimum design rules [67, 68, 69]. Many of these will be discussed in more detail in the sections below.

One might wonder why the fifth and sixth layers – the layers in the center of the board - were chosen for the power planes. The ground planes were placed on layers two and nine in order to be as close as possible to the components and traces on the top and bottom layers to minimize the length of the return path, essentially treating the traces as microstrip transmission lines [70]. Conventional wisdom would then be to place the associated power planes as close as possible to the ground plane, ideally forming ground-power pairs to reduce parasitic inductance [67, 70]. However, the ZYNQ-FMCL-HDMI board contains several voltage rails, which are split into several sub-planes over the two power layers. Each of these power layers contains approximately half of the total voltage rails utilized by the board, with only the DDR 1.5 V one appearing on both power layers. Figure 23 shows the distribution of the voltage rails on the two power planes. This custom PCB contains important chip components on both the top and bottom layers. If the power plane would have been placed as close as possible to the components and ground plane near the top layer, then conversely it would be as far as possible from the components on the bottom layer requiring that same voltage rail. Therefore, as a compromise, this design instead opts to place the power planes on the center layers five and six in order to minimize the distance between them and *both* the top and bottom layer components and traces.

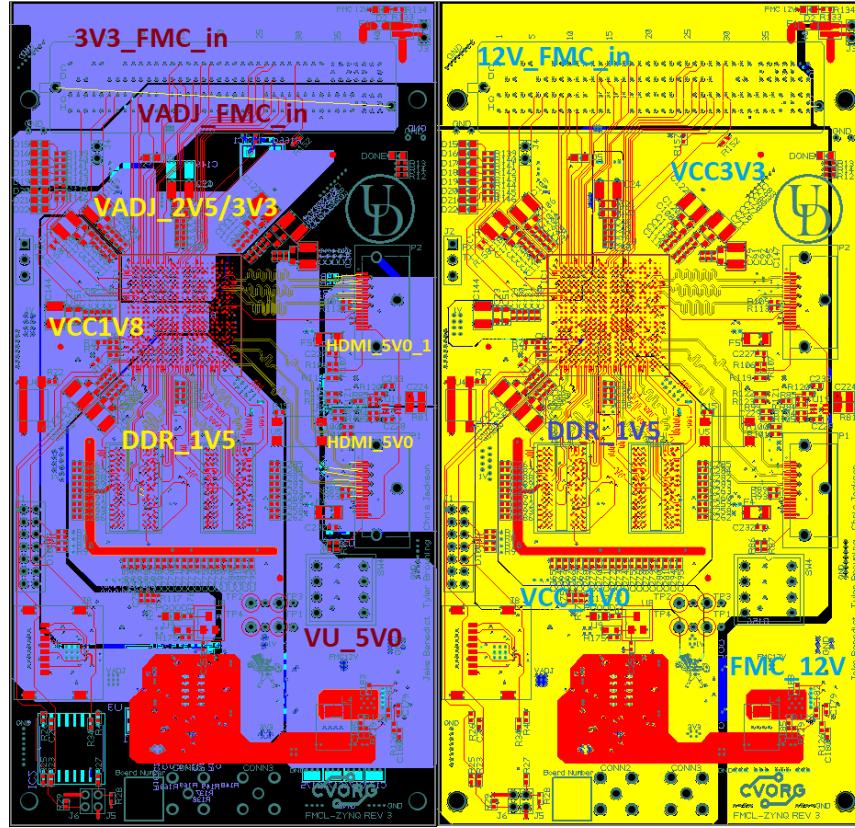


Figure 23 ZYNQ-FMCL-HDMI Power Layers and Voltage Distribution

The ZYNQ-FMCL-HDMI revision 3 board was meant to plug into the LPC FMC connectors of the TB-6V-LX760-LSI board in the NESSIE CSE. The previous revision of the board (aka the class board) was designed to be a standalone device that received power through a standard AC adapter or battery, which would directly provide a 5 V DC source. This revision, on the other hand, receives its power from the CSE through the LPC FMC, which provides it with 12 V DC. In this design, we take that 12 V from the FMC and pass it through a Maxim Integrated MAX8686ETL+ voltage regulator, which then provides a steady 5 V output to the next part of the system. From here, the design is very similar to the second revision of the board where

the 5 V is then used to provide voltage to the other components. This version of the board has several important power rails that are derived from this 5 V input – a 3.3 V, a 1.8 V, a 1.5 V, and a 1.0 V rail. An Analog Devices ADP5052ACPZ-R7 voltage regulator was chosen to provide steady voltages on each of these rails. This voltage regulator can be used to implement the PS and PL power on sequencing as described in the XC7Z020 FPGA data sheet [65, 71]. For example, the 1 V rail, which is tied to VCCPINT and VCCBRAM described in the FPGA data sheet, is activated by the regulator first, and then the “power good” signal for this is tied to the enable signal of next rail to be activated until all of the voltage rails are active in the correct sequence.

The LPC FMC also provides this gigabit I/O board with another voltage source called VADJ, which can be either 2.5 V or 3.3 V. This signal is sent to a Texas Instruments TPS22924CYZPR load switch. The last voltage rail to be enabled in the voltage regulator described in the previous paragraph is the 3.3 V signal rail, which is used for I/O banks 13 and 33 of the FPGA. VADJ is used for I/O banks 34 and 35 of the FPGA. When both 3.3 V and VADJ are asserted, a simple AND gate is used to produce a signal called “power good all” (PG_ALL), indicated that the power up sequence has been successful, and the ZYNQ-FMCL-HDMI board is ready for use. The PG_ALL signal is routed to the PS_POR_B input on bank 500 of the FPGA, taking it out a reset state to activate the PS. This signal is also routed to a small LED circuit so that the user can observe at a glance if there is a problem with the power on sequence. Similarly, LED circuits are also added to the 12 V and VADJ input lines coming from the FMC connector to quickly determine if there is a problem with the power coming into the board. The ZYNQ-FMCL-HDMI board’s power regulation schematic can be seen in Figure 24.

It is perhaps helpful to briefly review some electronics concepts before discussing the ZYNQ-FMCL-HDMI power regulation PCB layout.

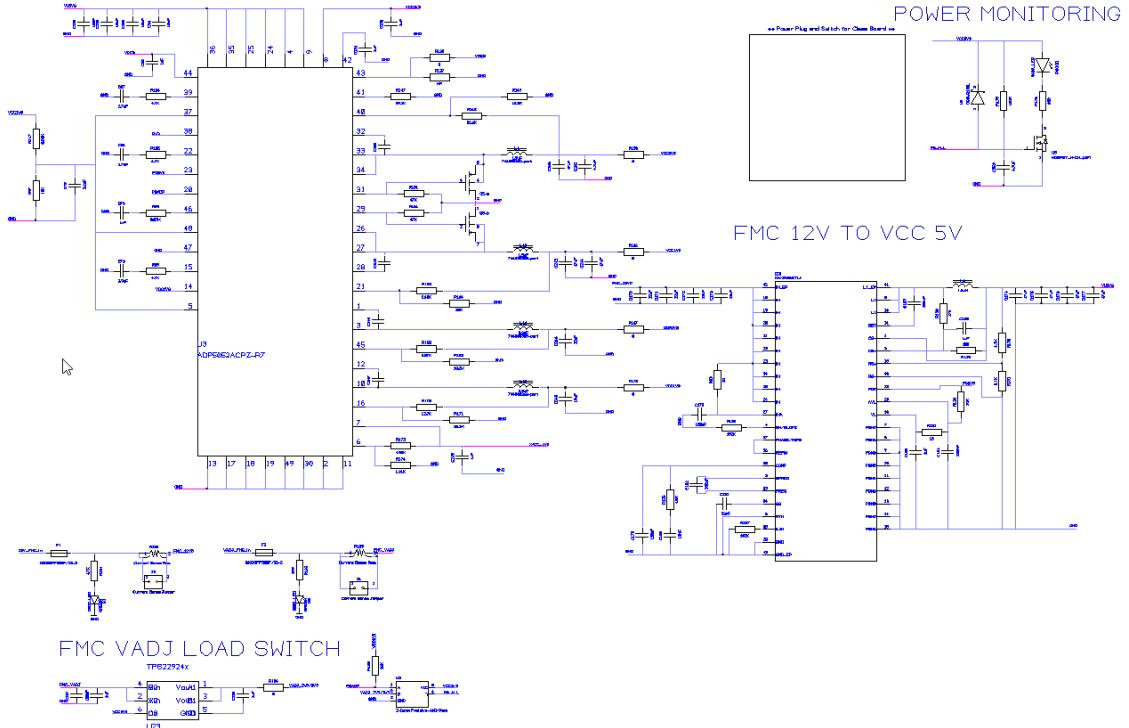


Figure 24 ZYNQ-FMCL-HDMI Power Regulation Schematic

A PCB trace is a strip of a conductor (often copper) with a rectangular cross section that carries current between two circuit elements. The resistance of the trace is dependent entirely on its geometry, which for copper is given by [70]:

$$R = \frac{(0.000487 * l)}{w * t} \quad (4.1)$$

where R is the resistance in Ohms (Ω), w is the trace width, l is the trace length in inches and t is the thickness of the copper, typically given by the industry in ounces.

Clearly and not surprisingly, larger cross sections for the trace decrease resistance, while longer traces increase resistance. Taking this concept to the extreme, if one were to fill an entire PCB layer with copper, then a current passing along that layer would experience the lowest possible resistance per unit length for that board geometry. This is known as a *plane* and is typically used if many circuit elements share the same signal, especially in the case of ground or power connections. When a circuit element passes a signal to another, current flows along the trace then flows along a return path on the ground plane. This entire structure behaves like a transmission line [70, 73, 74]. The return path on the ground plane will follow the path of least impedance, which varies by signal frequency such that at low speeds it follows the path of least resistance (i.e. the shortest path), whereas for high speeds it follows the path of least inductance (i.e. the smallest signal loop, usually directly under the signal trace) [70, 72]. Considering the impedance of the trace, much like the resistance it depends mostly on the geometry but also depends on the separation between the trace and ground plane and the *relative permittivity* (also sometimes known as the *dielectric constant*) of the substrate [70, 73]. The most common substrate material is called FR-4, which is composed of a woven fiberglass cloth with epoxy resin and has a relative permittivity of approximately 4.5. Traces on the top and bottom layers (also called “outer layers”) of the PCB have one side exposed to open air (relative permittivity ~ 1), so that the effective permittivity affecting these traces is lower than traces on inner layers that are instead surrounded by the substrate. As a result, signals on the outer layers typically have a faster propagation speed than those on inner layers [70, 73, 74].

The considerations of the previous paragraph (among many others) drove several of the decisions made during the layout of the ZYNQ-FMCL-HDMI power

regulation circuitry. Unfortunately, due to very high density of signal traces near the FMC connector and the DDR RAM, as well as limited space in the horizontal direction, the voltage regulator integrated circuits were placed on the opposite side from the FMC connector. In other words, if you consider the FMC connector to be on the north side of the board, then the power ICs are located on the south side. A path of this length would increase the resistance, so a portion of one of the power planes are dedicated for the important task of providing the 12 V voltage supply from the FMC connector to the Maxim MAX8686ETL+ chip. This board contains many components operating at different voltage levels, so that it was not feasible to have a separate power plane per layer for each voltage rail. The cost of a PCB increases dramatically with increased number of layers. Instead, two power planes were chosen (layers 5 and 6), which were then split into different “mini-planes” using a method called a *copper pour*. The designer has a lot of leeway on how they shape the copper pour, and it is almost always significantly wider than any of the traces on the PCB. On this board, to carry the 12 V power from the FMC to the power regulator a copper pour was made on layer 6 with a width at its narrowest point of approximately 180 mils (where 1 mil = 1/1000 inch). Lowering the overall resistance using this wide copper pour also helps to curb the potential temperature rise produced by this relatively high voltage.

The Maxim MAX8686ETL+ and Analog Devices ADP5052ACPZ-R7 voltage regulator circuits reside mostly on the bottom side of this gigabit I/O board. This was done to match the fact that the FMC connector is also located on the bottom, but also to avoid some of the DDR RAM circuitry on the top side. Additional smaller copper pours are utilized in these circuits of connect several nodes with the same level, especially ground and power connections. Most notably, the 5 V power between the

two voltage regulators is carried on a copper pour on the top layer, indicated by the orange outline in Figure 25. When traces are used to connect certain nodes, to minimize resistance the minimum width is 8 mils, unlike other parts of the board which have traces as narrow as 5 mils. Passive components like resistors and bypass capacitors are kept as close as possible to ICs. Each voltage rail produced from ADP5052ACPZ-R7 is connected to a small pour, which is then stapled using multiple vias to the appropriate copper pour on one of the power layers on five or six.

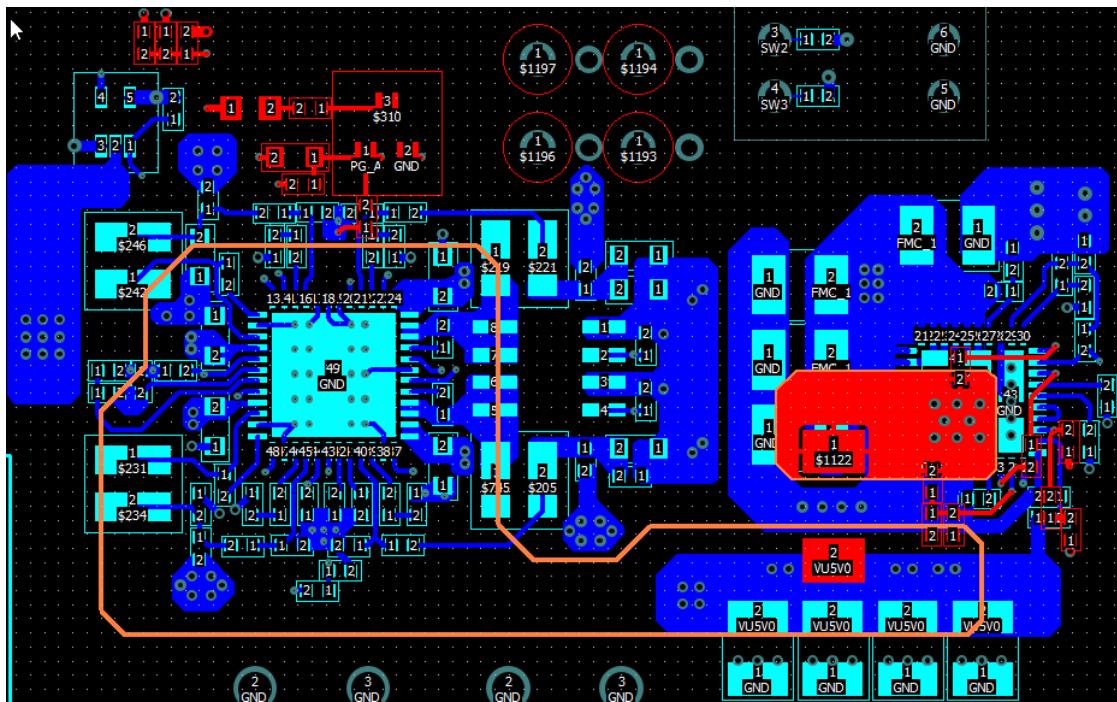


Figure 25 ZYNQ-FMCL-HDMI Power Regulation PCB Layout

The ZYNQ-FMCL-HDMI rev 3 board contains a pair of Micron Technology MT41K128M16JT-125 2 Gb DDR SDRAM chips to support and enable the ZYNQ processing system (PS). This RAM is powered by the 1.5 V voltage rail as described

in the previous section. It has a reference voltage of 0.75 V, which is created by a simple voltage divider using two 10 KΩ resistors and a 10 nF capacitor. It has 15 address bits, 3 bank address bits, and command signals such as write enable (WE) that are pulled up to 0.75 V through 40 Ω resistors. A differential clock, DDR_CLK_P and DDR_CLK_N, is provided to the DDR chips by the ZYNQ FPGA PS. The data, clock, address, bank address, and control input signals are shared between the two individual RAM chips and connected to bank 502 of the XC7Z020 FPGA. The RAM is used by any software stack running on the ZYNQ ARM microprocessor. Figure 26 shows the schematic for this portion of the design.

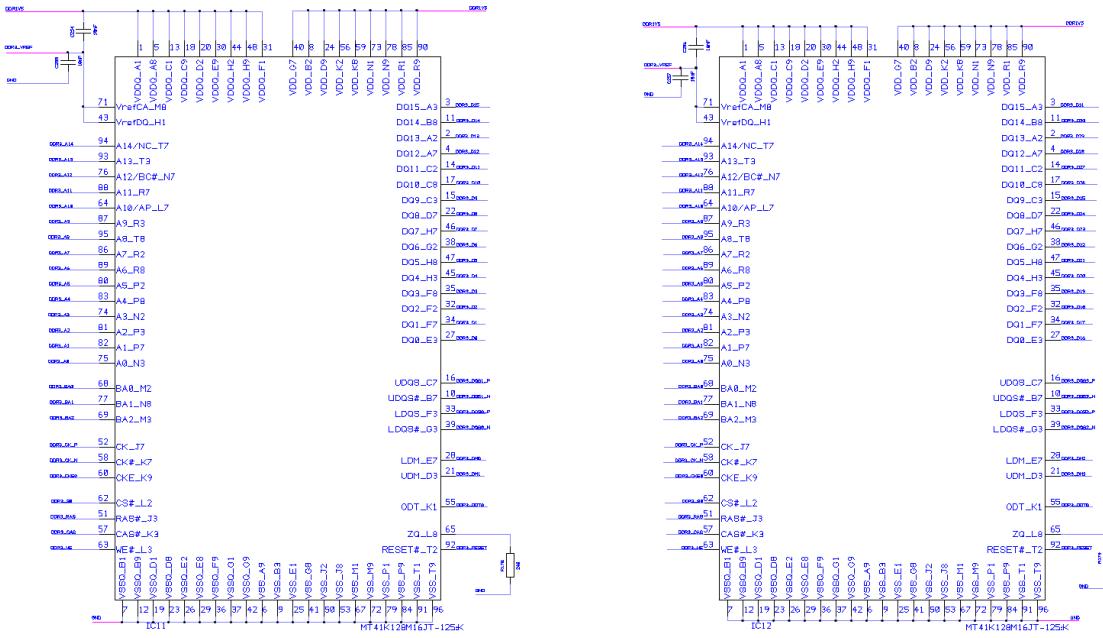


Figure 26 ZYNQ-FMCL-HDMI DDR Schematic

The two MT41K128M16JT-125 SDRAM chips each consist of a 96-pin ball grid array with a pitch of 0.8 mm. The ZYNQ XC7Z020 FPGA was placed near the

center of the board on the top layer and oriented such that bank 502, which contained the control signal logic pins to control the DDR, was located on the side opposite the FMC connector. The RAM chips were placed near the FPGA on the top layer, but with sufficient space to allow routing the many control signals. A copper pour for the reference voltage VREF was placed on the top layer running to the west and south sides of the RAM chips. This connected to another copper por on the bottom layer that connected to the VREF pins on the FPGA. The 0.75 V pullup voltage for the address signals, bank address signals, and control signals was placed on the bottom layer near the first RAM chip, IC11. The pullup resistors for this pour were also placed on the bottom side of the board, under the RAM chips and near this copper pour. The capacitors associated with the RAM circuit were arranged primarily on the top side to the west, south, and east sides of the two DDR chips. Vias were placed equally distant from the BGA pins, usually diagonally from each pin (for example, there is a via 15 mils up and 15 mils to the right of pin 1). Most of these vias connect to the 1.5 V copper pour on power layers 5 and 6, or to the ground planes on layers 2 and 9. Most of the signal routing was done on the top layer. However, the signal routing to the FPGA is quite dense, so the remaining vias help to facilitate any signals that could not be routed on the top layer and instead had to be done on other layers such as layers three, four, or even the bottom layer. The differential clock signals and data signals were treated as high-speed signals, and as such serpentine routing was used to length match these signals to a maximum difference of 50 mils. Trace widths of at least 12 mils were used on most traces connected to power or ground pins to maximize current flow and reduce heating. The signal traces routed to the FPGA were 5 mils wide in order to facilitate routing through the narrow gaps of its BGA as recommended by the

Xilinx BGA device design rules document [69]. Moreover, 5 mils was the minimum trace width capable of being produced by our chosen PCB fabrication manufacturer. Figure 27 depicts the PCB layout of the DDR RAM circuitry.

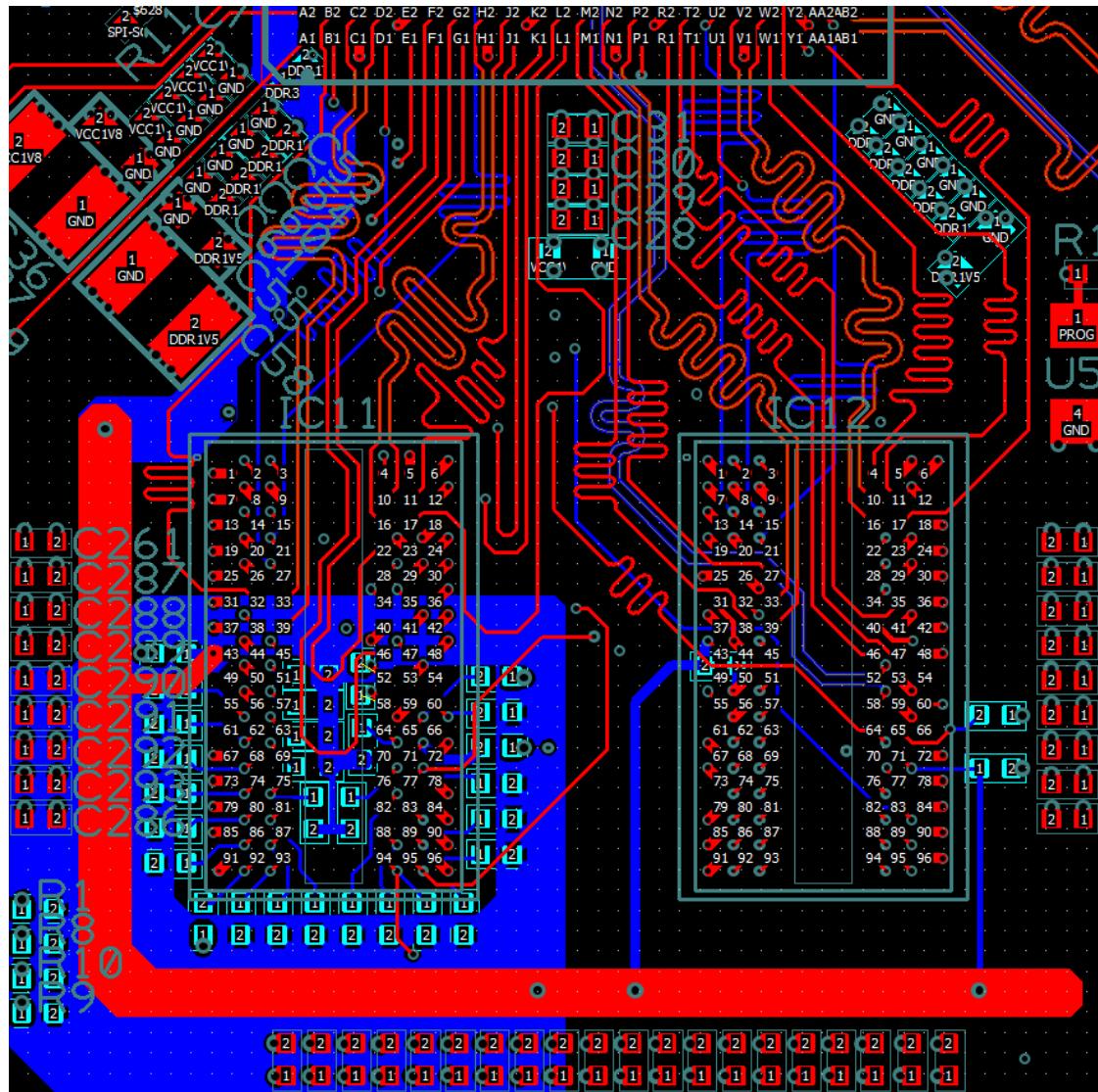


Figure 27 ZYNQ-FMCL-HDMI DDR PCB Layout

Programming the XC7Z020 FPGA involves loading a bit file, which is generated during the process of synthesis and implementation of firmware written in a hardware description language (HDL), to initialize and define the desired functionality of the ZYNQ PS and PL. The primary means of programming the ZYNQ-FMCL-HDMI board is through a JTAG connector, which is routed to bank 0 of the FPGA. The data lines from this connector are also tied to a Littlefuse SESD0802Q4UG diode array to ground in order to help protect against electrostatic discharge (ESD). Also connected to bank 0 is a simple LED circuit for the “DONE” LED, which asserts when the programming of the PS and PL has been successful. The DONE LED is one of the most important early debugging tools as it allows the user to see at a glance whether or not the FPGA has successfully been programmed. The JTAG programmer is used for volatile programming of the FPGA and is mostly used to debugging and testing purposes, since in this case the program is lost whenever the CSE power cycles. Additional nonvolatile means of programming the gigabit I/O board were included in the form of a MicroSD connector and SPI flash chip. A Molex 0472192001 MicroSD connecter is routed to a Texas Instruments TXS02612RTWR SDIO port expander. Only one of the two sets of ports from the port expander is used, with the second being available for future SD expansion if desired. The port expander takes the signals from 3.3 V to 1.8 V, and these SD signals are then sent to FPGA bank 501 to connect to the PS of the ZYNC SoC. Unfortunately, programming using the MicroSD card was not successful in the final product as the PS was never able to properly detect and read from the SD card. If another revision of this board were to be produced, a different MicroSD connector would be chosen to potentially solve this issue, since the most likely cause is a problem with the chip detect (CD) signal.

Nonvolatile programming was successfully implemented using a Cypress Semiconductor S25FL128SDSMFV000 SPI Flash IC. The data and clock signals from this chip were sent to FPGA bank 500 for the ZYNQ PS. A bit file loaded from the JTAG and saved onto the Flash would then persist and reprogram the FPGA after a power cycle. A pair of jumpers J5 and J6 are used to select the programming mode. Refer to Figure 28 for the schematics of these circuits.

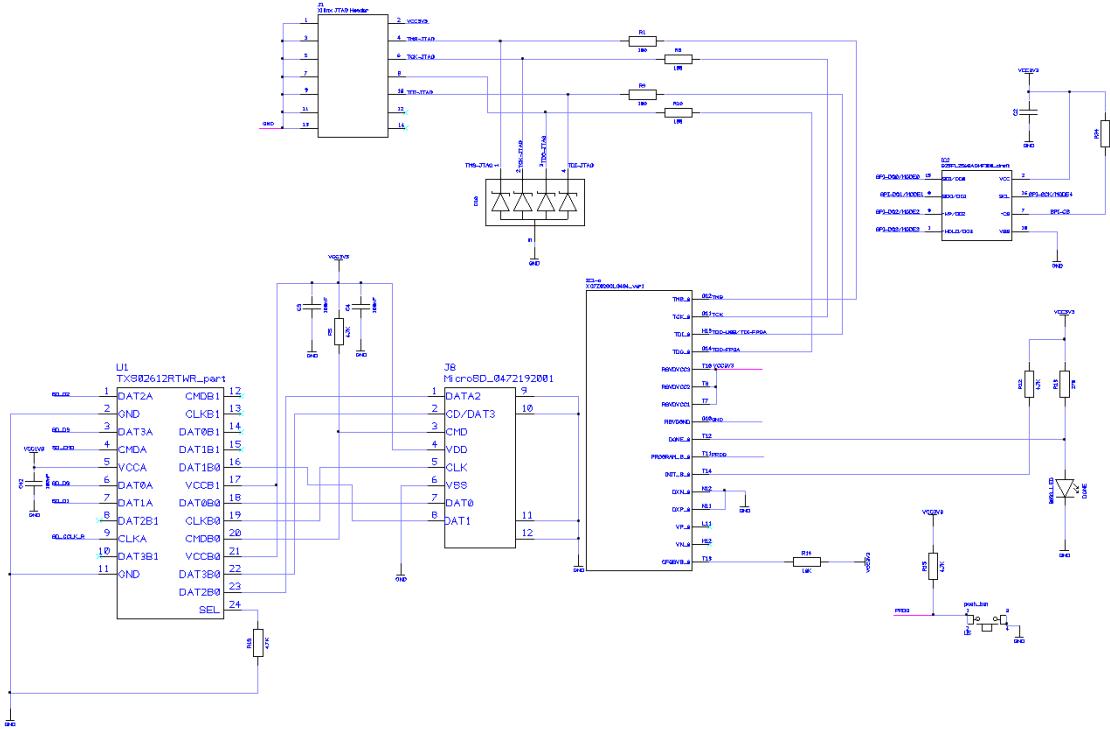


Figure 28 ZYNQ-FMCL-HDMI JTAG, Flash, and SD Card Schematic

The PCB layout considerations for the JTAG, MicroSD connector, and SPI Flash chip were relatively straightforward, especially compared to the densely packed high-speed signals of the SDRAM. All three of these circuits were placed on the

southwest area of the board, on the end opposite the FMC connector on the north side of the board. The JTAG connector is a simple through hole component placed on the top side for east access, with the ground pins tied to the ground planes, the 3.3 V power pin tied to the large 3.3 V copper pour on layer 6. The MicroSD connector was also placed on the top side for ready access, while its associated SDIO port expander was placed directly under it on the bottom side for convenient routing and space efficiency. The power pins for the 1.8 V side of the SDIO port expander were connected to the 1.8 V copper pour located on power layer 5. The SPI Flash chip was placed on the bottom side, with its associated resistors on the top side. Controlling through hole jumpers J5 and J6 were placed directly south of this IC on the board's top side. The SPI Flash power pins used the huge 3.3 V copper pour on power layer 6. These three chips were some of the last components to be added, so many of the signals were routed on inner layers such as three, four, seven, or eight in order to cross the relatively large distance to the FPGA and navigate through existing routing on the top and bottom layers. Figure 29 shows the PCB layout for these three components.

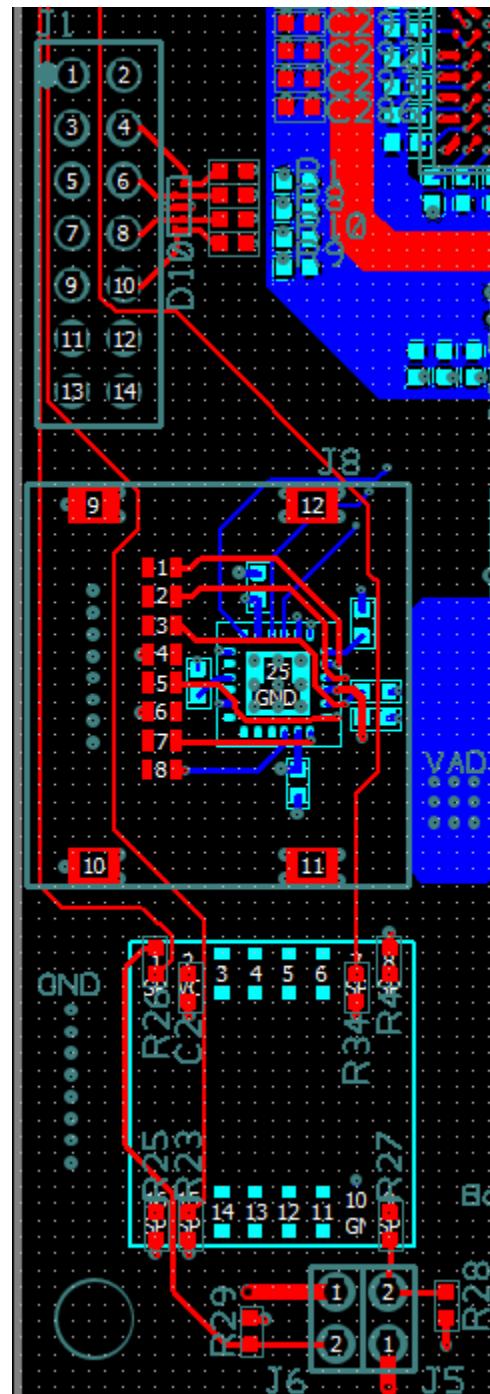


Figure 29 ZYNQ-FMCL-HDMI JTAG, Flash, and SD Card PCB Layout

The last portion of the ZYNQ-FMCL-HDMI rev 3 design that will be discussed in detail is the HDMI connectors. This board uses two vertical mounted Molex 0476591002 HDMI connectors. The high-speed data and clock signals have 49.9Ω pullup resistors to the 3.3 V power rail. The Inrevium TB-FMCL-HDMI board has one dedicated input and one dedicated output connector, but the CSE only requires HDMI inputs (only maybe using the output for loopback testing). Therefore, an advantage of the ZYNQ-FMCL-HDMI design was to have two configurable HDMI ports that could act as inputs or outputs, similar to the HDMI port on the Digilent ZYBO board. The FPGA controls the HDMI enable signals for each port and an Analog Devices ADP198ACPZ load switches on/off the appropriate 5 V HDMI rail.

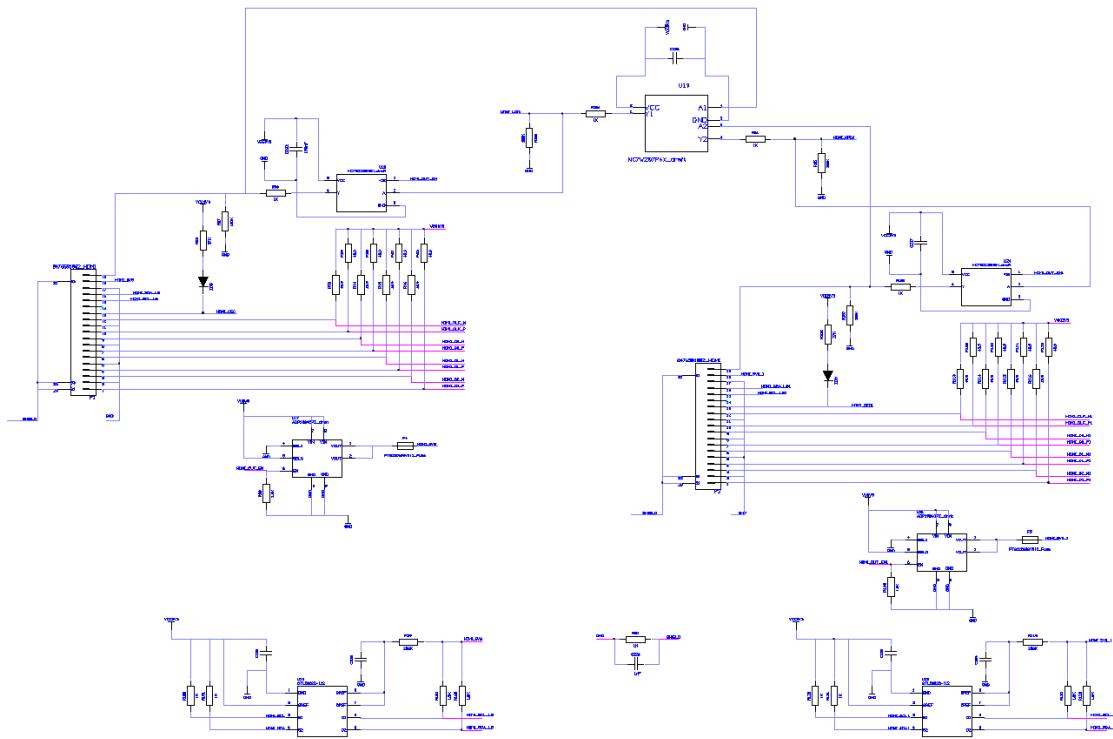


Figure 30 ZYNQ-FMCL-HDMI Board HDMI Connectors Schematic

The HDMI connectors are shielded by a $1\text{ M}\Omega$ resistor in parallel with a 1 nF capacitor connected to ground. The I₂C signals HDMI_SCL and HDMI_SDA for each connector is connected to a GTL2002D 2-bit bidirectional voltage translator, with each side of the translator pulled up to the appropriate voltage depending on whether it is routed to the FPGA I/O bank or the HDMI connector. Figure 30 above shows the schematic for the HDMI connectors.

The HDMI connectors were placed on the right (east) side of this revision of the board, as opposed to the first version where they were placed near the south edge. This was done for two reasons – (1) to provide more space for routing the SDRAM signals on the south side of the board and (2) to minimize the distance between the HDMI connectors and the FPGA for maximum signal integrity [94]. The two HDMI connectors are mounted on the top side of the board, and have large through hole shield pins, but the signals are on top layer surface mount pads. The data and clock signals are treated as high-speed signals since the pixel clock for HDMI can reach frequencies of up to 148.5 MHz [52]. Therefore, these signal traces on each HDMI connector are length matched using serpentine routing to be within a maximum of 50 mils difference. The data signals are routed on the top layer, while the clock signals are routed on the bottom. The data and clock signals are routed as differential signals, with a minimum differential pair gap of 8 mils and at least 80% of the trace treated as paired (to reduce skew). The $49.9\text{ }\Omega$ pullup resistors to 3.3 V on these traces are connected by vias. Each HDMI connector has its own associated 5 V copper pour on layer 5, routed to the ADP198ACPZ-R7 load switch to the board's layer five 5 V copper pour, which is of course tied to the MAX8686ETL+ 12 V to 5 V voltage regulator. The PCB layout for the HDMI circuits is shown in Figure 31.

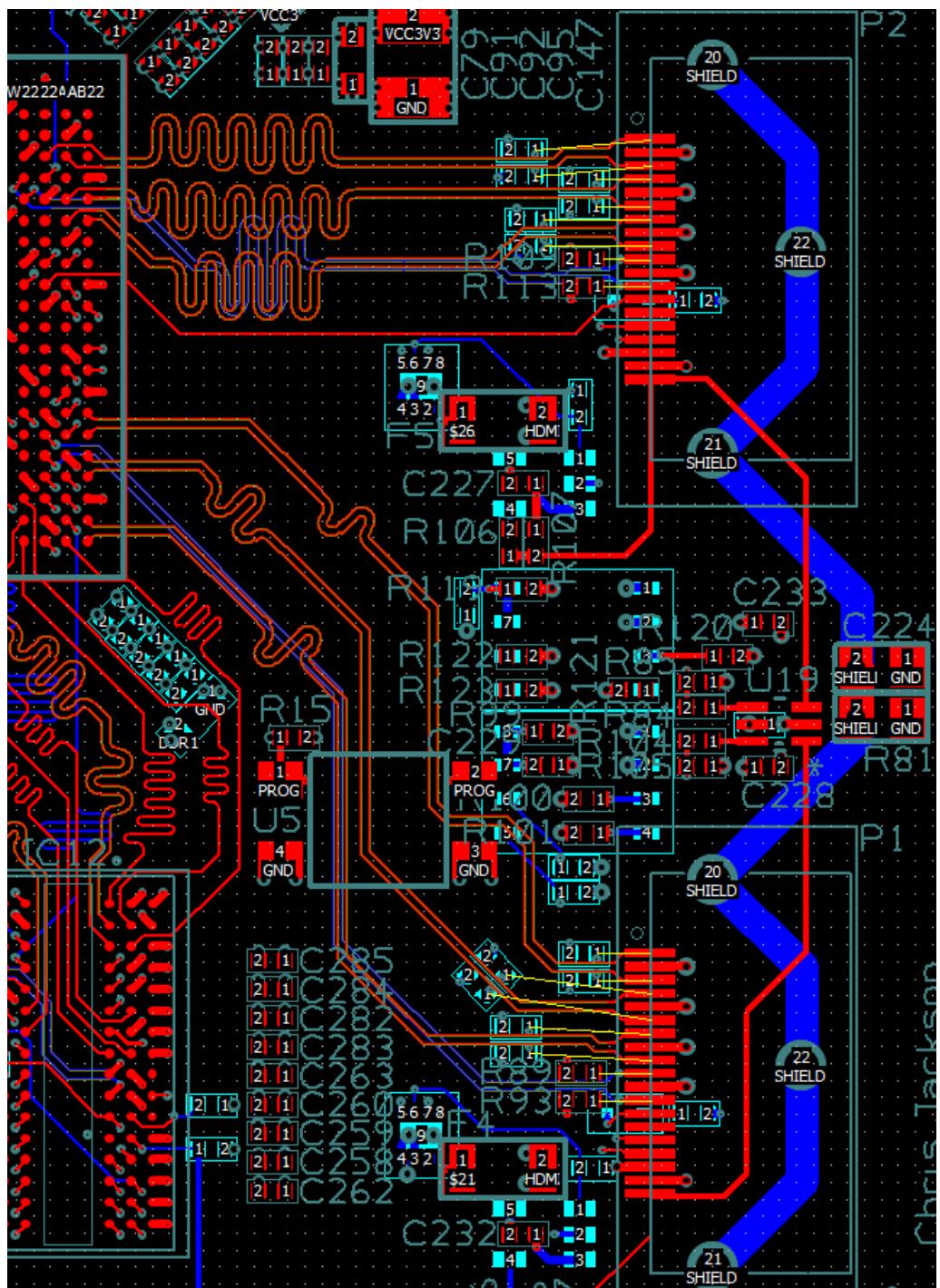


Figure 31 ZYNQ-FMCL-HDMI Board HDMI Connectors PCB Layout



Figure 32 Current Generation ZYNQ-FMCL-HDMI Board

Photographs of the complete ZYNQ-FMCL-HDMI board are shown in Figure 32. Additional information on the schematics and layouts for the parts of the board not discussed in this chapter can be found in Appendix A.

4.3 Gigabit Interface: Results, Testing, and Troubleshooting

Initial results for the completed ZYNQ-FMCL-HDMI rev 3 board were promising. A set of sixteen of these boards were ordered, followed by another set of twenty-four board about one year later. The second batch contained a few minor corrections found due to testing the first batch, along with a few quality-of-life changes such as adding more silkscreen labels. Of these forty total boards, four were not functional due to manufacturing defects, bad parts, or in one case a still-unsolved problem with the FPGA programming. When the ZYNQ-FMCL-HDMI boards first arrived, they were subjected to simple quality tests before being placed in any system. Visual inspection of each board was done to see if any problems could be detected with component soldering, placement, or orientation, followed by connectivity tests to find any short circuits. Next, the boards were connected to one of the TB-6V-LX760-LSI boards in isolation (i.e., not in a full CSE) and the voltages were tested using an oscilloscope and confirmed to be at the expected levels. During this test it could also be determined if the “power good” LED would activate. In the initial batch, some of the boards intermittently had a problem with the “power good” LED indicating a problem on startup, and it was quickly determined that a pullup resistor was missing on one of the power circuits. This problem was corrected through a simple rework in the first batch of boards, and the second batch of board included this correction in the PCB layout. After power testing, next was programming and functionality verification.

Programming and functionality tests were done on the same isolated TB-6V-LX760-LSI board setup described in the previous paragraph. One or two of the revision 3 HDMI cards were tested at a time, connected to the same LPC FMC slots that they would be in a full CSE system. A Xilinx Platform Cable USB programmer module was connected to an HDMI card via JTAG and linked by USB to a control PC

running Windows 10 and the Xilinx Vivado Design Suite. A set of simple firmware projects was developed in VHDL for PL testing or a combination of VHDL and C for PS testing. These projects were designed to each test one or two portions of the revision 3 HDMI board's functionality at a time. For example, the first test was called LEDSSW, written entirely in VHDL, which when completed would confirm that (1) the PL JTAG programming worked as intended at all, and (2) the simple VHDL program could toggle the LEDs using the on GPIO switches, confirming the LED and switch hardware operability and the ability to control them with the PL. This test was run first on all of the boards to confirm that the FPGA could indeed be programmed successfully. Similar tests were used to test the SD card, Flash, UART, HDMI connectors (both input and output), DDR, and FMC connector. The purpose and results of each of these tests is briefly summarized below. Appendix B contains the test results for each of the ZYNQ-FMCL-HDMI revision 3 boards.

The UART test firmware was created in Vivado using the block diagram tool and instantiating a ZYNQ PS block. The hardware design and bit file for this Vivado project were then exported to Xilinx Software Development Kit (SDK), where a simple C program was created to transmit text data over UART. On the hardware side, a USB-to-UART converter was added to the test setup, with the UART side connected to the appropriate pins on the HDMI card and the USB side connected to the test PC. The test PC could listen on the serial port using SDK or a terminal program such as TeraTerm. The serial parameters were configurable in the ZYNQ PS block, but we used the default baud rate of 115200 bps. This test confirmed several important functionality features. First, it showed that it was possible to program the ZYNQ PS on the revision 3 HDMI boards. Secondly, it showed that a C program running on the

ZYNQ processor could be used to control hardware on the board. Lastly, it confirmed that serial communication from the HDMI card operated correctly. This was important because in a full CSE system, serial data could be used to send diagnostic or error data from the HDMI card to a NUCPC.

The SD Card and Flash firmware projects were made for the ZYNQ PS similar to the UART project described in the previous paragraph. Each project was developed to test one of the methods of nonvolatile programming for the HDMI card. The idea behind each project was to use Xilinx SDK to program the SD Card or the Flash with a bit file, then power cycle the system and see if the HDMI card would automatically load the bit file from the nonvolatile media source. If this process were successful, shortly after power up, the “Done” LED would be asserted and the HDMI card would have the functionality of the bit file (for example, if the LEDSW bit file were used, then on power up the switches would control the board’s LEDs).

Unfortunately, the SD Card project never operated correctly. The Xilinx tools could not successfully detect the microSD card in order to write to or load data from it. It is suspected that the problem is due to a difference between the Molex 0472192001 microSD connector chosen for the revision 3 HDMI card versus that used in the ZedBoard that this part of the design was loosely based on. Our design has chip detect (CD) signal that is shared with the DATA3 signal, but on the ZedBoard design, the CD exists as a separate additional signal. If a revision 4 of the HDMI board is ever made, a different microSD connector will be chosen to see if the addition of this signal would correct the problem with detecting the microSD card. Fortunately, the Flash project was successful so that it was possible to load a bit file from flash after a power cycle. Simply using Flash programming was sufficient for our purposes.

The DDR firmware test project was another ZYNQ PS block diagram design that was exported to Xilinx SDK. There is a pre-packaged memory test C program included in SDK meant to test the SDRAM connected to a ZYNQ processor, similar to MemTest86 used to test RAM on a computer [75]. This project simply used the on-board memory tester, which sent the results over UART which were read on the control PC. Of the 40 cards, only 1 failed this test due to a suspected bad RAM chip.

The testing of the HDMI connector consisted of three slightly different firmware projects. These projects were HDMI IN, which tested a single HDMI input, HDMI In-Out (I/O), which was used to test one HDMI connector configured as an input and the other as an output, and HDMI In-In (I/I), where both connectors were configured as inputs. These projects all contained a ZYNQ FPGA firmware IP core to decode the HDMI data, as opposed to the dedicated RX/TX decoder chips found on the Inrevium TB-FMCL-HDMI board. A Xilinx Integrated Logic Analyzer (ILA, not to be confused with the different ILA described in Chapter 5), was added to each of these Vivado projects to observe the digital data on the FPGA side. An ILA is similar to the previous Xilinx tool called ChipScope and is used to monitor internal digital signals in a design [76]. For the HDMI IN project, data was sent from the control PC's GPU, and the control PC could send different HDMI modelines which control the resolution and pixel clock frequency. Thus, in a way analogous to a full IRSP system, the test setup could send video data over HDMI at nonstandard frame rates and resolutions, such as 512x512 pixels at 500 Hz. The input data to the HDMI card would then be observed on the ILA to confirm that it was consistent with the expected output data from the GPU. The HDMI I/O project was similar, except that the input and output data from the two HDMI connectors was observed at the same time, and the

output loopback could be visually observed on a monitor. Finally, the HDMI I/I project confirmed that both HDMI connectors could be configured as inputs and the data from each could be observed on the ILA at the same time.

The last of the initial HDMI card test projects was for the FMC connector. This project was written in VHDL requiring only the FPGA PL, and simple toggled all of the output signals from the FPGA to the FMC slot at different rates. The HDMI card was connected to a standalone TB-6V-LX760-LSI board through this FMC slot. On the TB-6V-LX760-LSI's Virtex6 FPGA, ChipScope was used to confirm the digital inputs from the HDMI card were correct. ChipScope was used because the Vivado ILA was not compatible with technology older than the Xilinx 7-series FPGAs, so the Virtex6 (6-series) was incompatible. Additionally, an oscilloscope was used to monitor some of the signals on the FMC connector because the FMC signals on the HDMI card were densely routed and there was a mild concern of crosstalk or interference between adjacent signal traces. The testing observations from ChipScope showed that the FMC logic levels were correct, and the oscilloscope showed that the voltage levels were correct with no signal crosstalk observed. The results of all of the above tests on the 36 operational HDMI cards provided confidence to use them in the full NESSIE CSE system.

The ZYNQ-FMCL-HDMI revision 3 boards were integrated into several full CSE systems and used extensively to test the University of Delaware IRLED arrays, particularly NSLEDS as well as some initial testing with the multi-CSE HDILED [5, 77, 78]. Additionally, most of the testing, development, and results for the early implementations of the packetized display protocol (PDP) presented in Chapter 3 were done with CSEs with these HDMI gigabit I/O cards installed. For close to a year,

many experiments run using NSLEDS arrays were done at speeds less than 120 Hz, and in this regime the boards acted well as a drop-in replacement for the Inrevium HDMI cards. However, unfortunately as the speeds and resolutions of the scenes were increased in later tests to prepare for increased customer demands in later programs, inconsistencies and failures started to be observed in the CSEs containing these custom boards. At frame-rates in excess of 120 Hz, and especially at rates higher than 240 Hz, it was observed that the HDMI signal from one or both HDMI cards would be lost on the NUCPC side. These lost signals were intermittent, inconsistent, and unpredictable. It was also observed that the initial test setup described previously containing only a TB-6V-LX760-LSI board showed this behavior much less frequently than a full CSE. It was also observed that the type of HDMI cable used to connect to the HDMI seemed to make a difference, with DisplayPort-to-HDMI converter cables being less reliable.

The above observations seemed to indicate a potential problem with HDMI signal integrity that the ZYNQ-FMCL-HDMI revision 3 boards may have been more sensitive to than the Inrevium boards. A small PCB was developed to potentially investigate these issues, called the HDMI Probe board. This board consisted of two HDMI connectors tied together, meant to be an HDMI pass-through. Connected to each of the data and clock differential pairs was a set of SMA connectors configured as AC-coupled terminators. These SMA connectors could be connected directly to an oscilloscope to observe the high-speed HDMI signals for signal integrity problems. Jumpers were connected to the low-speed, non-differential HDMI signals in case they needed to be probed as well. The full schematic for the HDMI Probe Board can be seen in Figure 33.

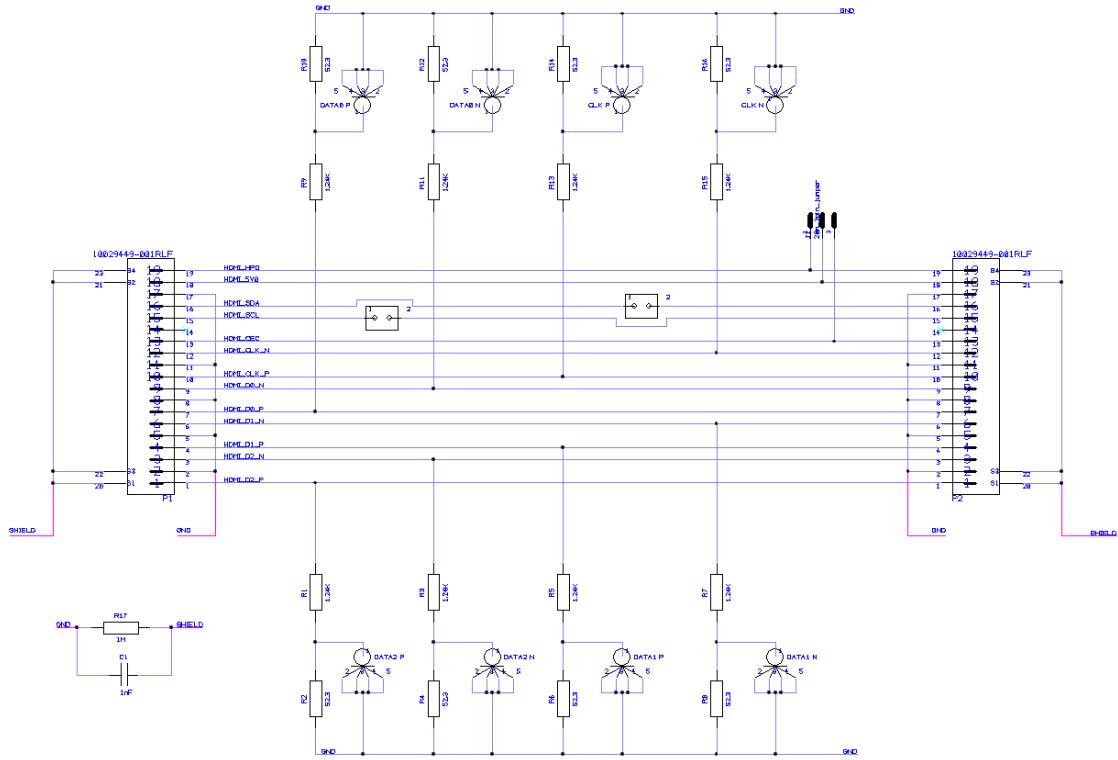


Figure 33 HDMI Probe Board Schematic

The PCB layout for the HDMI Probe Board is depicted in Figure 34 below.

The description of this design will be brief because this board is relatively simple. The Probe Board is a two-layer PCB with the bottom layer mostly consisting of a ground plane. The top layer has the eight through hole SMA connectors on the east side, three through hole jumpers for probing the slower single-ended HDMI signals on the north side, and the two HDMI connectors with their associated resistors on the west side. The differential clock and data high-speed signals between the two HDMI connectors are length matched, and due to the small size of this board serpentine routing was not necessary to achieve this length matching. The SMA connectors and resistors form an AC coupled terminator circuit for a 50:1 probe to an oscilloscope.

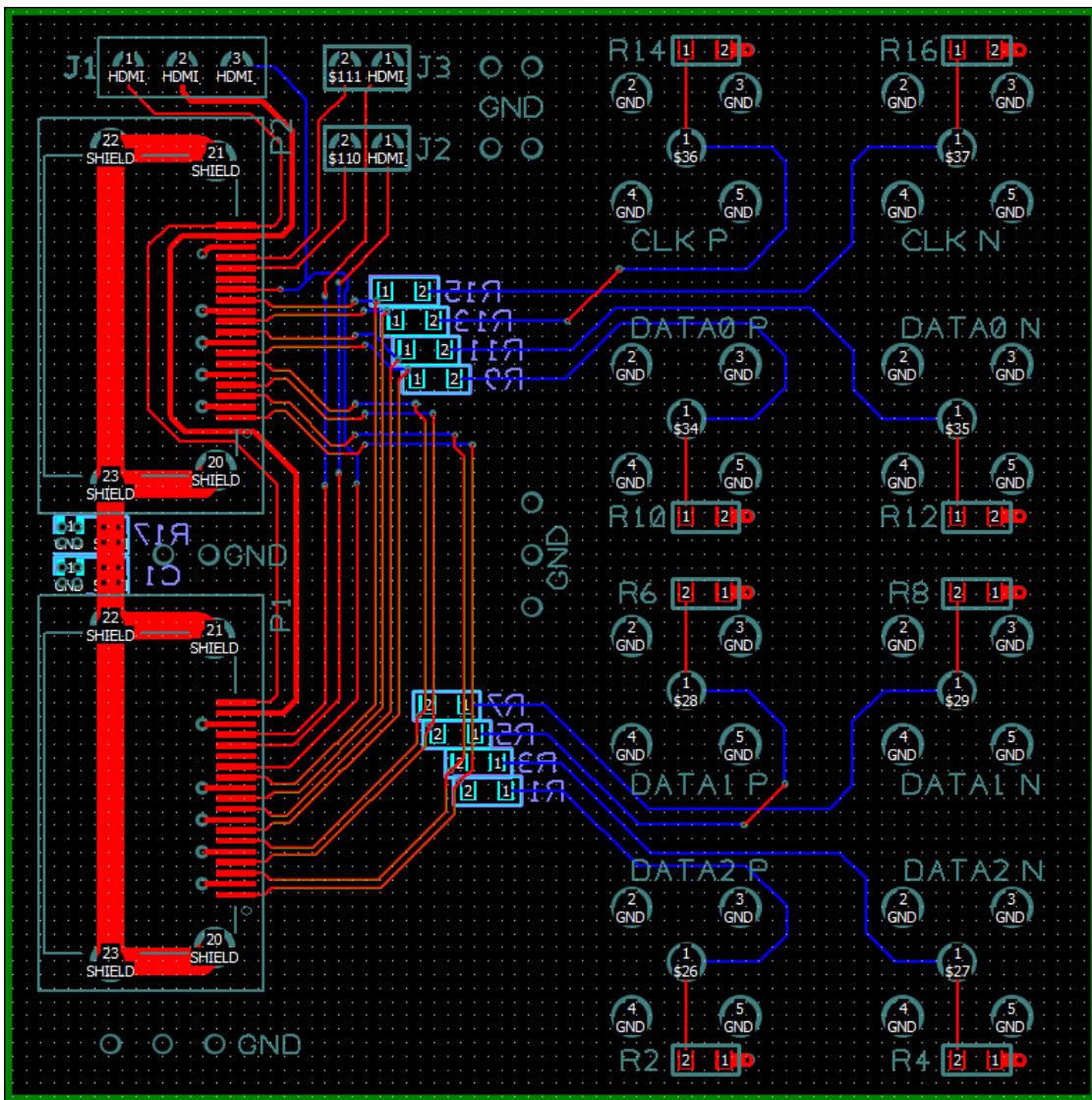


Figure 34 HDMI Probe Board PCB Layout

While waiting for the probe cards to be fabricated, additional tests were done to attempt to determine the cause of the problems with the HDMI I/O cards at high frame-rates. It was observed that on the input to the NUCPC, the XorgService program that interpreted the HDMI data would often fail for one or both of the HDMI

cards when running above 240 Hz. We suspected that maybe there was an intermittent problem with the HDMI hot plug detect (HPD) signal on system startup or during system operation. A problem like this could have been missed during our initial electrical testing, where we simply probed the HPD voltage levels with a multimeter with the HDMI cards linked to a standalone TB-6V-LX760-LSI board. A test setup was built where the HPD signal and several voltage rails were probed with an oscilloscope and observed during startup and normal operation. If HPD was the problem, it was expected that this would be observed as a drop in the HPD signal whenever XorgService reported that one of the HDMI signals was lost. This test setup started as a standalone TB-6V-LX760-LSI board, but then DAC boards, AMP boards, and finally interface boards were added one at a time until it was the equivalent of a full CSE. It was also observed whether or not these signals were affected by restarting the XorgService program.

The results of this testing are summarized below. For more details on the various iterations of this test (different DAC and amplifier configurations, etc.), please refer to Appendix B. The first observation was that on the standalone TB-6V-LX760-LSI test setup with no DACs or amplifiers connected to the other FMC slots, the HDMI cards were always detected correctly by XorgService. In this case, the HPD signal and voltage rails were steady and clean as shown in Figure 35. Upon restarting XorgService, a small amount of noise was observed on HPD and a strange spike in the HDP signal was observed at the finish of the restart – see Figure 36 - but the cards were still detected correctly after the restart process was complete. Adding some DACs and amplifier cards to the setup added some ripples on the HPD signal as seen in Figure 37, but not enough to affect XorgService detection.



Figure 35 Standalone test setup, probing HPD (yellow), voltage rails 3.3 V, 1.8 V, and 1.5 V (orange, green, and blue).



Figure 36 Spikes on HPD signal after the completion of XorgService restart.



Figure 37 Test setup with eight DACs, eight amplifiers, and the master interface board connected. Small ripples on HPD but HDMI cards still detected.

Unfortunately, the results of these tests were inconclusive as the XorgService detection problem was not reproduced on the standalone setup, even with all of the DACs, amplifiers, and interface boards installed. However, when the same HDMI cards were moved to a different, full CSE setup, one of the cards was consistently not detected by XorgService. The only differences between the above test setup and a full CSE were the case housing the entire unit in the full CSE and the internal HDMI cables from the outer case to the HDMI cards.

A problem with the case ground was briefly considered. Another set of tests was done with an oscilloscope probe grounded to earth ground by connecting the ground wire to the CSE power supply case. The ground plane on one of our ZYNQ HDMI revision 3 cards are probed, and then the test was repeated using one of the Inrevium HDMI cards. Both exhibited similar behavior, see Figures 38 and 39.

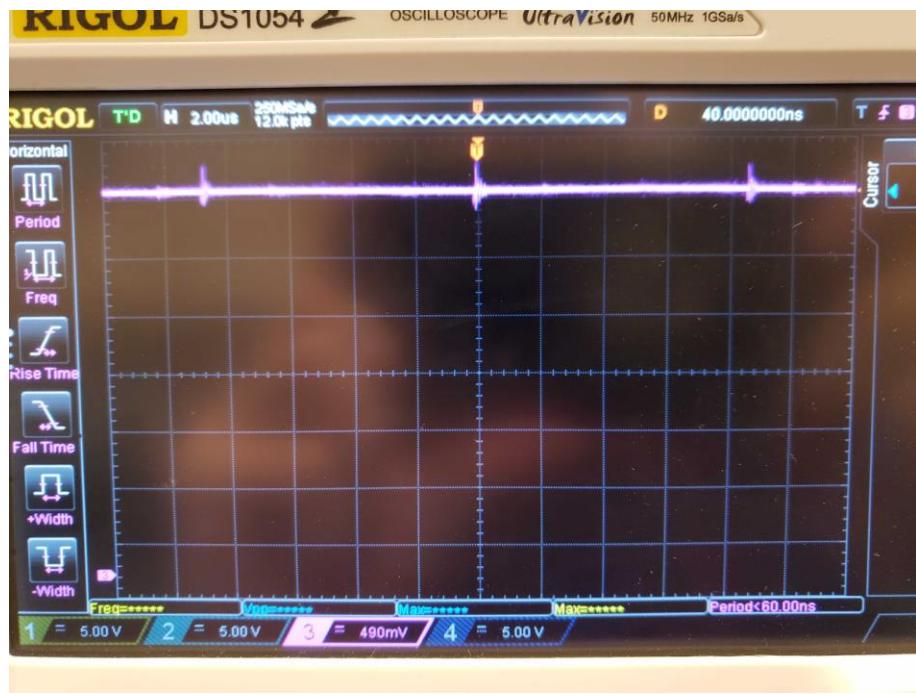


Figure 38 ZYNQ-FMCL-HDMI ground plane probe referenced to CSE case GND.

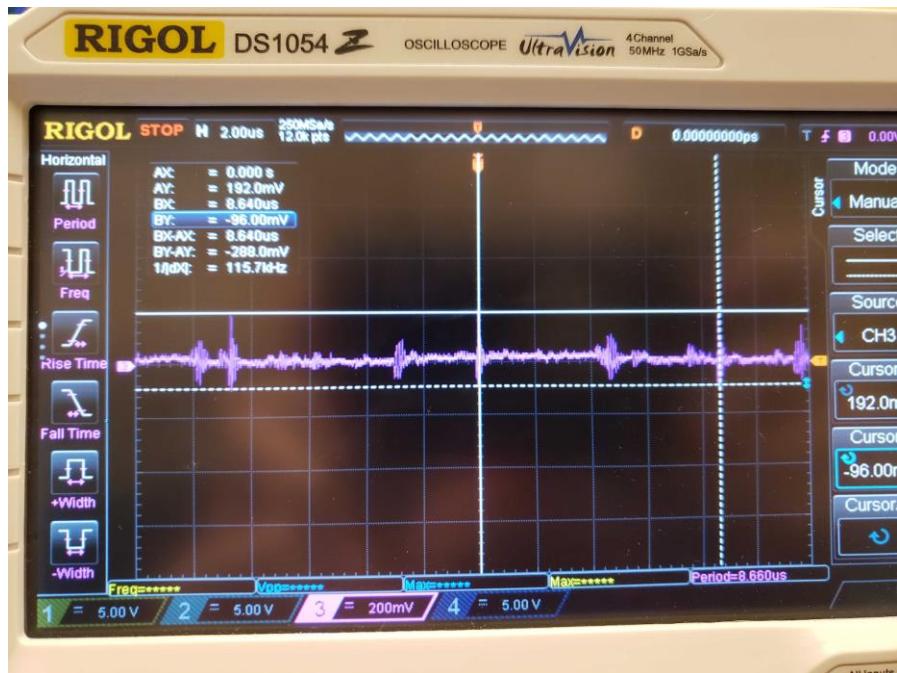


Figure 39 Inrevium TB-FMCL-HDMI card ground plane shows similar results.

Upon the arrival of the assembled HDMI Probe Boards, direct testing of the HDMI signal from various different HDMI cables began. As alluded to earlier, at this point it was apparent that different HDMI cables connected to a NUCPC would exhibit different behavior. However, in all cases the ZYNQ-FMCL-HDMI boards would eventually show failures at frame-rates of 400 Hz or higher. An ILA was added to the HDMI card firmware to observe the digital signals on the board itself. Additionally, colleagues from the CVORG firmware team developed a pseudo random binary sequence (PRBS) test that could be run to detect bit errors on the HDMI signals. The next set of tests was performed at 240 Hz on a full CSE, which is the frame-rate at which the ZYNQ-FMCL-HDMI consistently started to exhibit failures. See Figures 40-42.

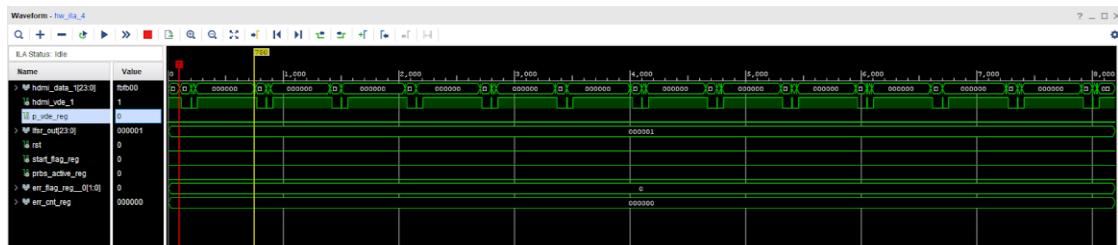


Figure 40 ILA, 240 Hz. Blips seen on vertical data enable (VDE) and data lines during horizontal blanking period.

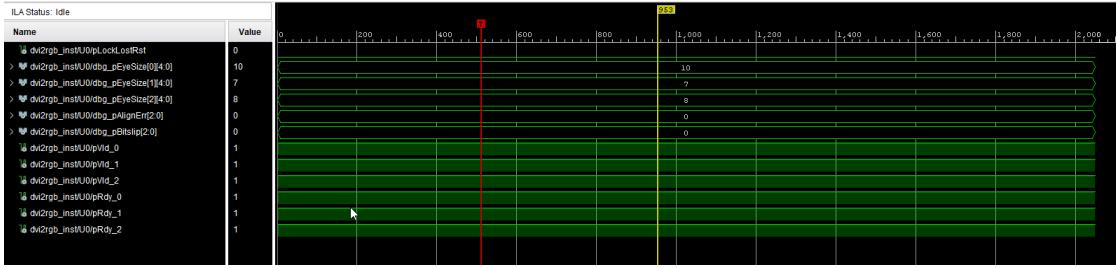


Figure 41 ILA, 240 Hz. All HDMI data and clock signals valid and ready.

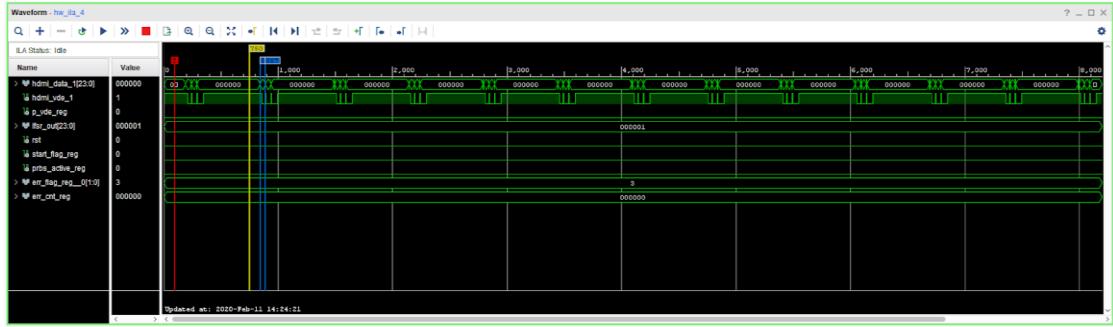


Figure 42 ILA, 240 Hz. HDMI cables connected directly from NUCPC to HDMI cards, bypassing the internal case HDMI cables. Signals are correct.

The tests above seemed to indicate that the issue might lie with the internal HDMI cables on the CSE case, since directly plugging an HDMI cable from a NUCPC to the HDMI card showed clean signals on the ILA even at 240 Hz. This was consistent with the earlier HPD testing where the standalone test setup operated correctly without the CSE case. The HDMI Probe Board was then connected to the CSE's internal HDMI cable and the PRBS test was performed on the HDMI card along with the HDMI TMDS signals probed on an oscilloscope. These tests can be seen in Figures 43-44 below.

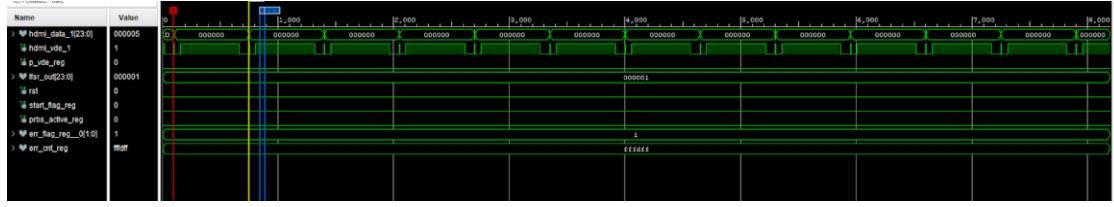


Figure 43 PRBS detects as active indicating sequence start is detected and data sent. However, data blips still occur during horizontal blanking.

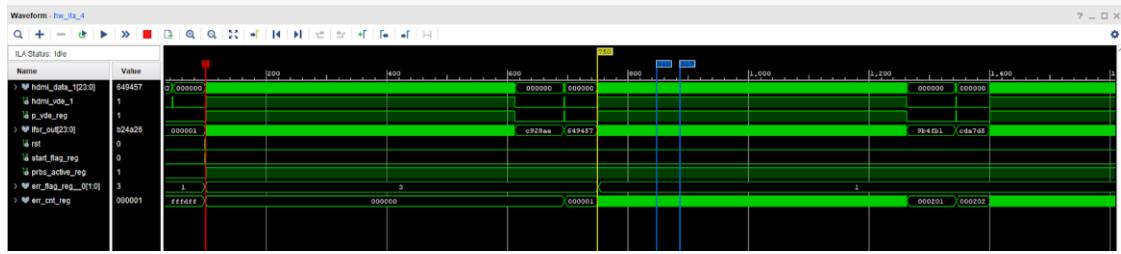


Figure 44 As a result of the data corruption during horizontal blanking, PRBS fails after first row of data.

The working theory at this point was that the internal CSE HDMI cable might be causing an impedance mismatch that might be distorting the TMDS HDMI signals, resulting in incorrect decoding when during the horizontal blanking period when the control tokens are sent [61]. Observation of the TMDS signals using the probe board can be seen in Figures 45–47.



Figure 45 Probe of HDMI TMDS signals on internal cable - DATA0P (yellow), DATA0N (green), CLK0P (orange), CLK0N (blue).

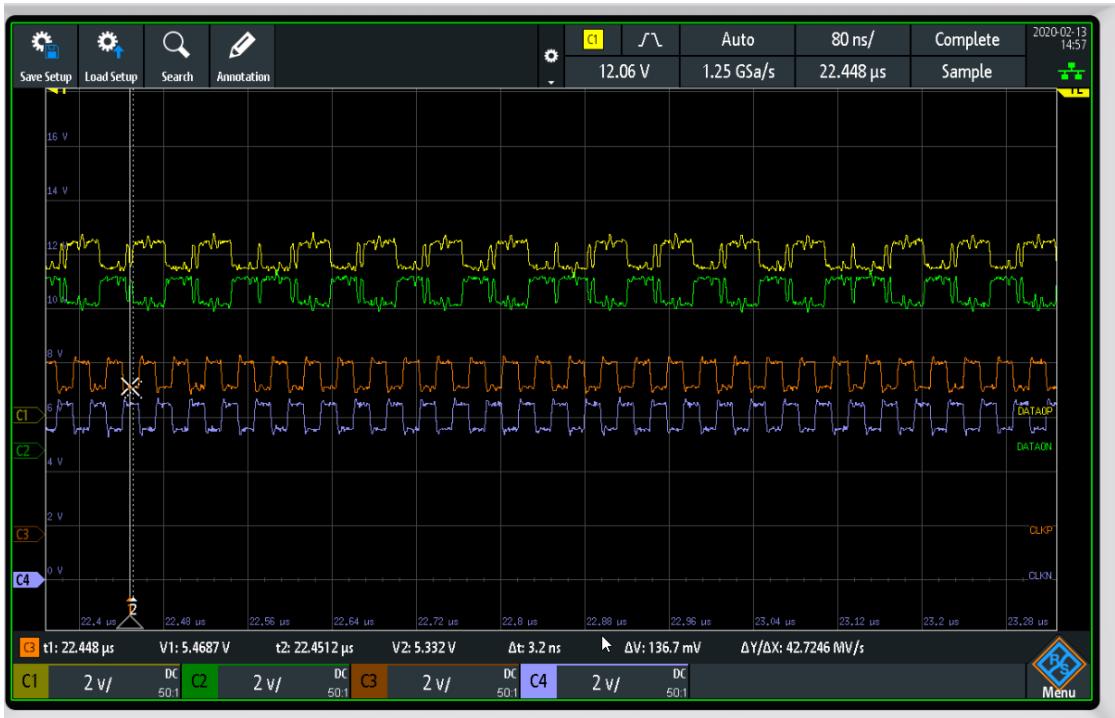


Figure 46 Probe of HDMI TMDS signals on cable directly from NUCPC - DATA0P (yellow), DATA0N (green), CLKP (orange), CLKN (blue). No observable difference from previous test.



Figure 47 Zoomed-in version of this test at 100 Hz in attempt to decode control token to detect any corruption.

Regrettably, even at 100 Hz these tests were inconclusive as it was difficult to reliably decode these high-speed signals and compare to the documentation using the equipment at hand. At this point we determined that perhaps more specialized equipment for observing high-speed differential signals might be helpful, so we began to collaborate with another Udel professor potentially use his equipment. Many of these plans are still pending after behind derailed by limited laboratory access due to the COVID-19 pandemic.

Chapter 5

TOWARD DESIGN OF A DISTRIBUTED CSE

5.1 Multi-CSE System

As described in *Novel Silicon Architecture for Increased Dynamic Range of Infrared Led Scene Projectors* [5], it is possible to drive each of the four quadrants of the SLEDs array with a single CSE. A prototype of this scheme has been developed and proven to work and will briefly be described below. The success of this prototype system is the basis for extending this idea into the distributed CSE concept described in Chapter 6.

The goal of the multi-CSE platform was to create a quad-CSE configuration capable of supporting the HDILED 2Kx2K (4 Megapixel) resolution and achieve a 4x increase in frame-rate [77]. The load on each CSE is divided such that each CSE only need drive one Megapixel in a matter functionally identical to driving a single NSLEDS array. The multi-CSE setup consisted of components such as a NUCPC with 8 display outputs, a custom signal extension board, a custom carrier board PCB, and custom dewar packaging. The carrier board is wire-bonded to the array hybrid such that the outer edge is outside the vacuum chamber. This carrier board directly accepts signals from each of the four CSEs, effectively quadrupling the number of available signals [89]. Figure 48 shows the ribbon cables from each CSE linked to signal extension boards in each quadrant and connected to the carrier board on the dewar.

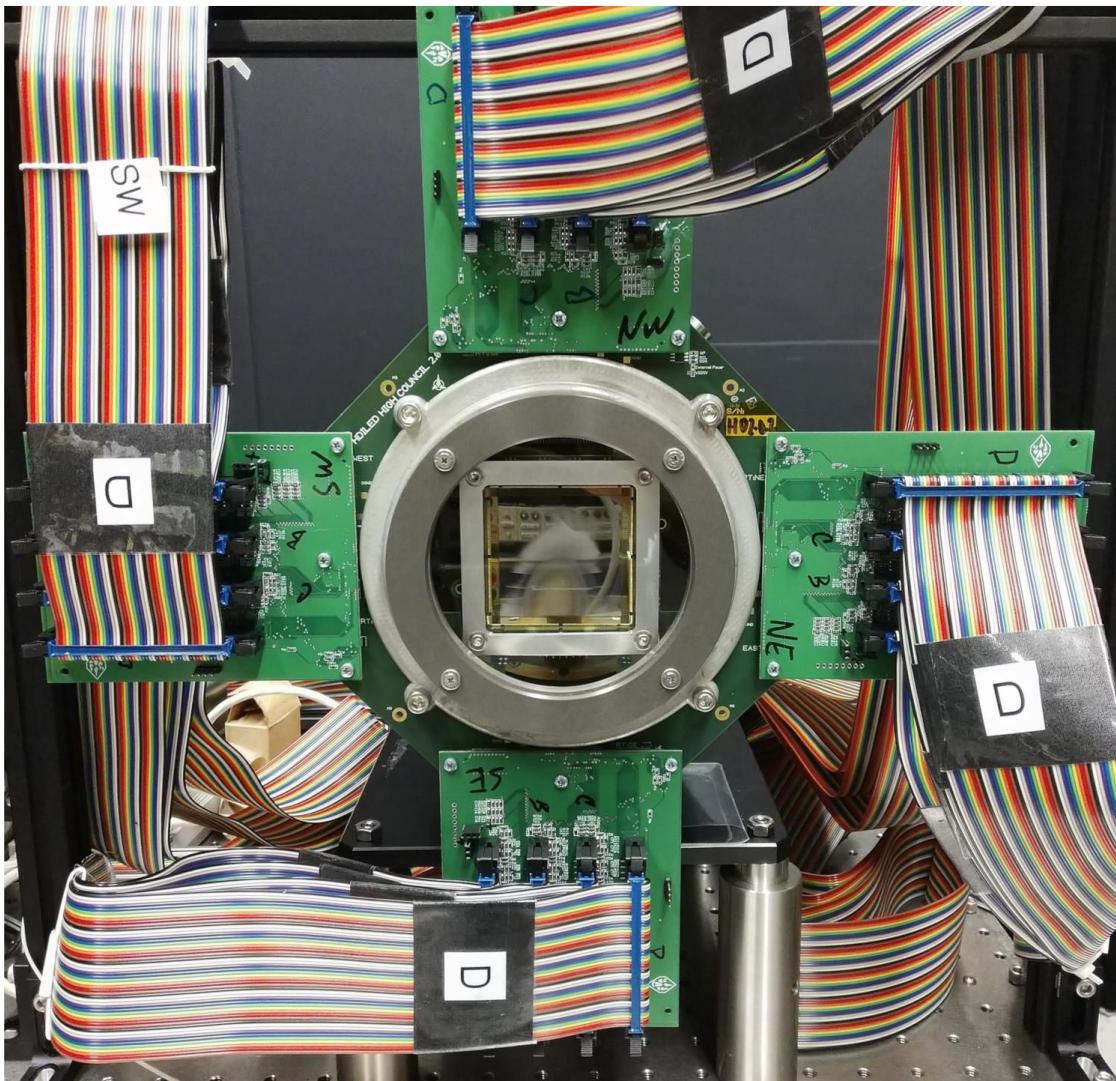


Figure 48 Prototype Multi-CSE System

5.2 Interim System – NESSIE2

Before describing the future distributed CSE design, it is helpful to consider an interim design which will be referred to as NESSIE2. The NESSIE2 design to be

described in this section can be thought of as a functional stepping-stone between the current CSE design and the future direction of the CSE systems going forward.

NESSIE2 is meant to be used immediately and be compatible with the current architecture. A significant goal of this new design is to transition the system to more modern FPGAs and DACs, since the previous design incorporated parts purchased in early 2010s. To help streamline the new design, NESSIE2 also doubles the number of analog channels from 32 to 64. Another improvement for the new design is the migration to high-performance cabling for the digital and analog signals to help improve signal integrity.

5.2.1 Synchronization

A significant challenge to expanding to future multi-CSE systems is the synchronization of the signals from the close support electronics to the array. In the past this was not as much of a concern because all of the data and control signals for the entire array could be provided by a single CSE. As arrays get larger, it becomes necessary to instead control portions of the array - such as halves, quadrants, or even rows - with a single CSE, and then use multiple CSEs to address the full array.

However, even in the case of a single CSE controlling a single array, implementing a proven synchronization scheme can help improve timing closure. In this section, a general examination of an industry standard method of achieving channel-to-channel synchronization in a multiple-input multiple-output (MIMO) system is provided. This scheme is then implemented into the proposed NESSIE2 architecture.

An effective method for synchronization of MIMO FPGA systems is to utilize devices that can support the JEDEC Standard 204B serial interface, hereafter referred to as JESD204B [79, 80]. A brief overview of this JESD204B standard is presented,

providing the necessary background to understand how it can be applied to future CSE architectures such as NESSIE2. At its simplest level, a JESD transmitter or receiver subsystem will typically consist of an FPGA, a clock generator, and one or more devices to be synchronized, such as DACs or ADCs. The clock generator creates a device clock and a reference signal known as SYSREF to each receiver or transmitter in the subsystem. Within the receiver or transmitter, this device clock is used to produce sample clocks, frame clocks, and local multi-frame clocks (LMFC). The SYSREF signal is synchronous to the device clock, and it aligns the frame clocks and LMFCs across all of the receivers and/or transmitters. In order to attain deterministic latency, this alignment of the LMFCs is vital to serve as a common time reference for each device in the subsystem [80]. Figure 49 shows an overview of this subsystem.

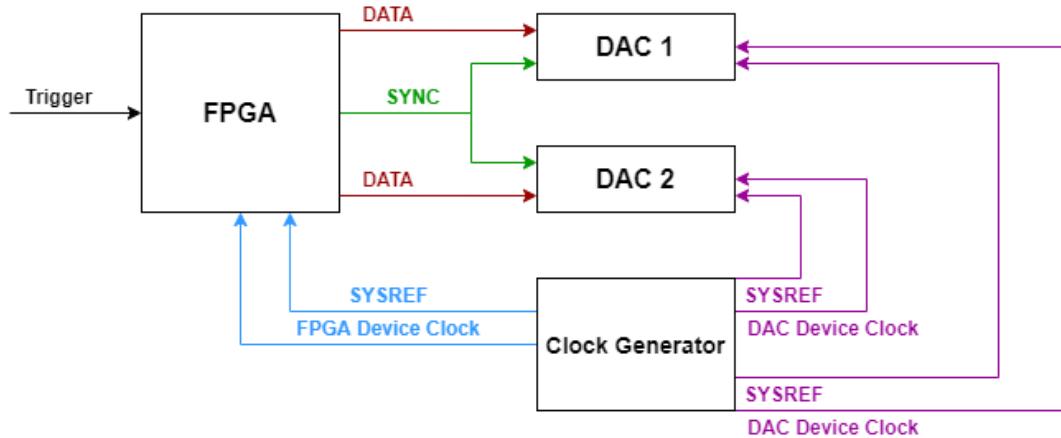


Figure 49 JESD Subsystem

A JESD link makes use of a technique known as 8b/10b encoding when sending data between transmitter and receiver, including synchronization requests.

8b/10b encoding is a standard method that essentially converts 8-bit words into 10-bit symbols to help ensure DC-balance while allowing for clock recovery [80, 82]. One of the features of this encoding scheme is that for any string of at least 20 bits, there are never more than five ones or zeros sent consecutively. For the purposes of this discussion, an important aspect of 8b/10b encoding is the existence of twelve control symbols that are valid 10-bit symbols but do not have a corresponding 8-bit data word. The use of these control signals varies by application. There are three special control signals, depicted by notation K.28.1, K28.5, and K28.7, which are called “comma” symbols which are used for synchronization by determining the alignment of 8b/10b codes within a data stream. These are the only valid symbols that contain bit sequences of five ones or zeros in a row [82]. All other valid data and control symbols have a maximum bit sequence of four or less consecutive ones or zeros. The JESD subsystem described in the previous paragraph makes use of a comma character denoted by /K/ = K.28.5. It also uses two other control characters denoted by /R/ = K.28.0 which denotes the start of a multi-frame, and /A/ = K.28.3 which represents the end of such a frame [80, 82, 86].

A JESD receiver-transmitter link begins with a code group synchronization (CGS) request sent by the receiver by asserting an active-low synchronization signal, SYNC~. On the next LMFC clock, the transmitters repeatedly send the comma symbol /K/. When the receiver successfully receives four consecutive /K/ symbols, it de-asserts SYNC~ on an LMFC edge, and then the transmitters start the initial lane assignment sequence (ILA) of four multi-frames. Each of these multi-frames starts with the aforementioned /R/ symbol and ends with the /A/ symbol, which are used by the receiver to align the multi-frame boundaries. In each serial lane, the received data

is buffered into a FIFO. After a frame buffer delay of some frame clocks after an LMFC edge, these data frames are read from the buffer resulting in deterministic latency between the receiver and transmitter [79, 80, 81, 86]. Four complete multi-frames have been sent when four /A/ symbols are received, which initiates the release of valid transmitter data. Figure 50 illustrates the timing diagram for the process described above. Note that although the serial data arrives misaligned in the timing diagram, since the data is only released from the FIFO once all of the data has arrived, the data departs the buffers with deterministic latency [80, 81].

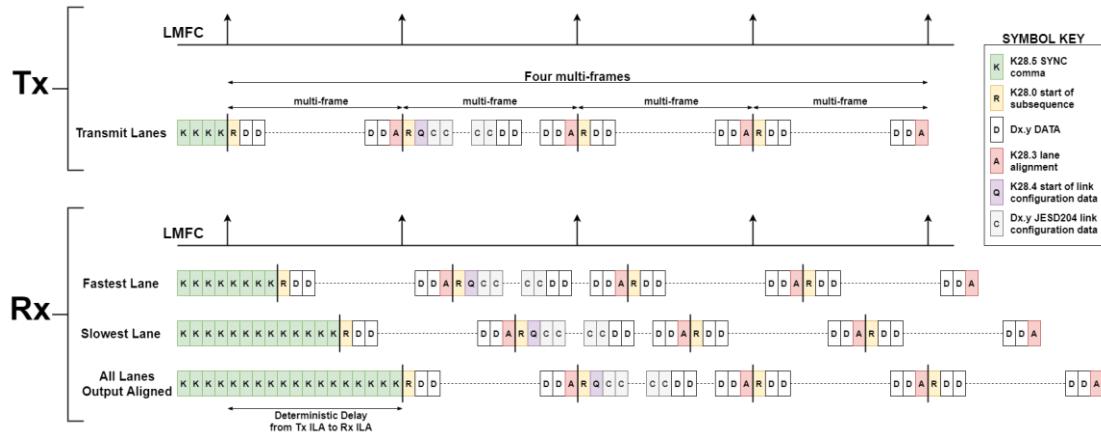


Figure 50 JESD Timing Diagram

Acquiring deterministic latency between receivers and transmitters is an essential step toward achieving system synchronization. Additionally, the JESD204B subsystems described in the previous section can be combined into a full system consisting of several subsystems, several FPGAs, and a common clock source to send a system clock, synchronization pulse, and acquisition trigger to all of the subsystems. Figure 51 shows a diagram of such a system.

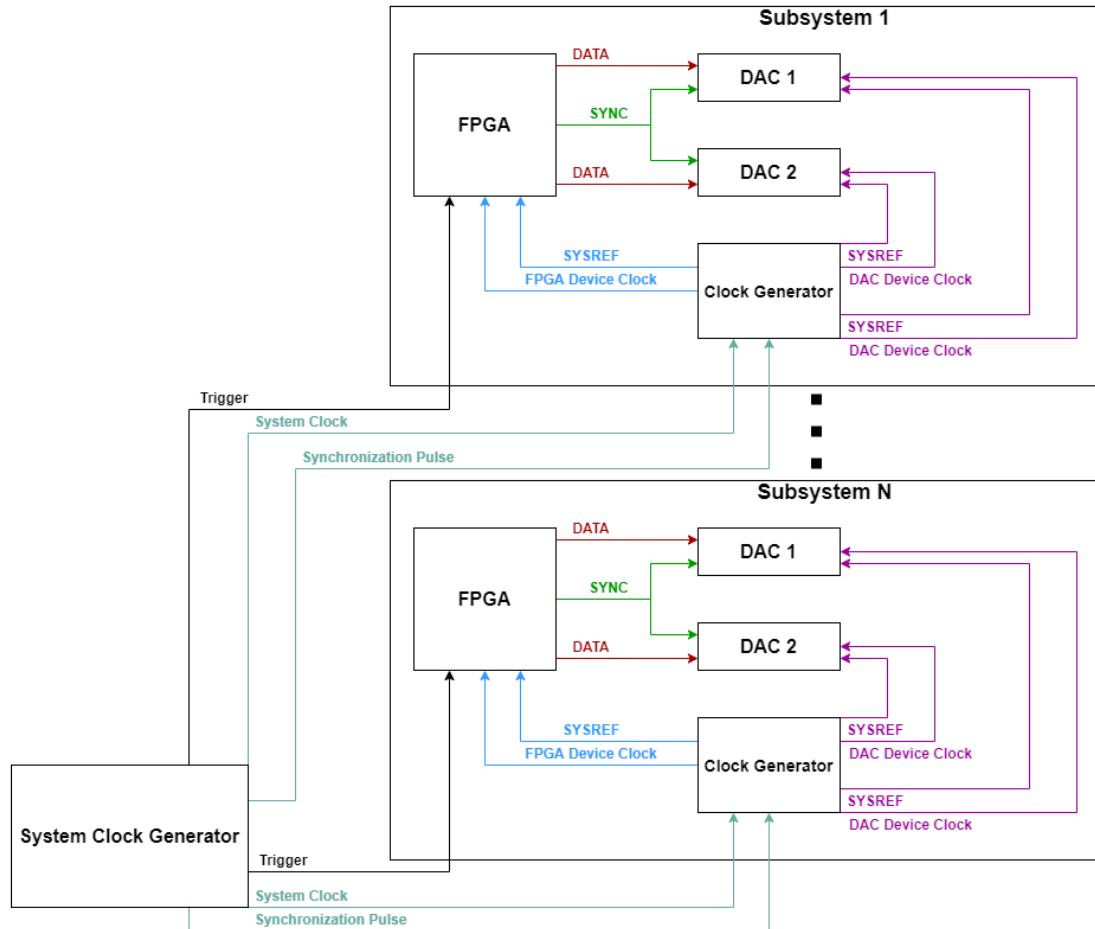


Figure 51 System of Multiple FPGAs and DACs and Clock Generator

A master clock generator can be used to distribute a common system clock and phase coherent synchronization pulse to each subsystem, thus ensuring deterministic latency for all of the FPGA and DAC links. Each subsystem has a clock generator which buffers the common external system clock in order to generate FPGA and DAC

device clocks. The clock generator on each subsystem also receives the synchronization pulse and uses it to reset all of the clock dividers that were used to generate the device clocks and SYSREF signals. As a result, the device clocks and SYSREF signals are phase aligned to each other as well as to the external synchronization pulse. It should be noted that in a real system, each subsystem may have different trace and cable lengths, resulting in skew between the device clocks and synchronization pulses. If this skew is sufficient to cause setup and hold time violations, it could result in a loss of deterministic latency. The higher the clock frequency, the more likely the possibility of timing violations due to each subsystem having less time to register the synchronization pulse. In order to mitigate this potential issue, it must be possible to adjust the skew of the synchronization pulse and SYSREF signals relative to the device clocks.

For the moment, let us assume that there are no timing violations such that the synchronization pulses and SYSREF signals across all of the devices are registered on the same clock period. Hence, JESD204B links can be established after performing code group synchronization and initial lane assignment as described in the previous section. At this point, on each subsystem the data on the FPGA is time-aligned and available. A simultaneous trigger to all FPGAs is then necessary to acquire the data. Similar to the synchronization pulse and SYSREF discussion above, this acquisition trigger needs to be synchronous to the FPGA device clock without failing setup and hold time requirements. As before, if there is a timing violation such that the FPGAs do not register the data on the same clock edge, it would result in skew in the data between the FPGAs in different subsystems. The ability to adjust the skew of the acquisition trigger relative to a device clock is also essential. Fortunately, JESD204B

master clock generator modules can perform the necessary skew to the synchronization pulse, SYSREF, and acquisition trigger signals. This is done using a well-defined system calibration procedure as described in [80, 95].

Figure 52 depicts a block diagram of the proposed NESSIE2 synchronization scheme implementing the JESD204B standard concepts described above. The FMC407 block located on the top right HPC FMC slot is the master clock generator for the system. It sends the Trigger signal as well as the synchronization pulse and system clock to the rest of the design. Each of the FMC216 blocks on the HPC FMC slots on the right side of the diagram consist of local clock generators and the DACs to be synchronized. The central FPGA sends a SYNC pulse to each of the DACs when the data is ready to be released. If multiple NESSIE2 systems are required in a more robust multi-CSE configuration, the master clock generator can also send the trigger, system clock, and synchronization pulse to each additional CSE system. Details on the specific devices used in this design are described in the following sections.

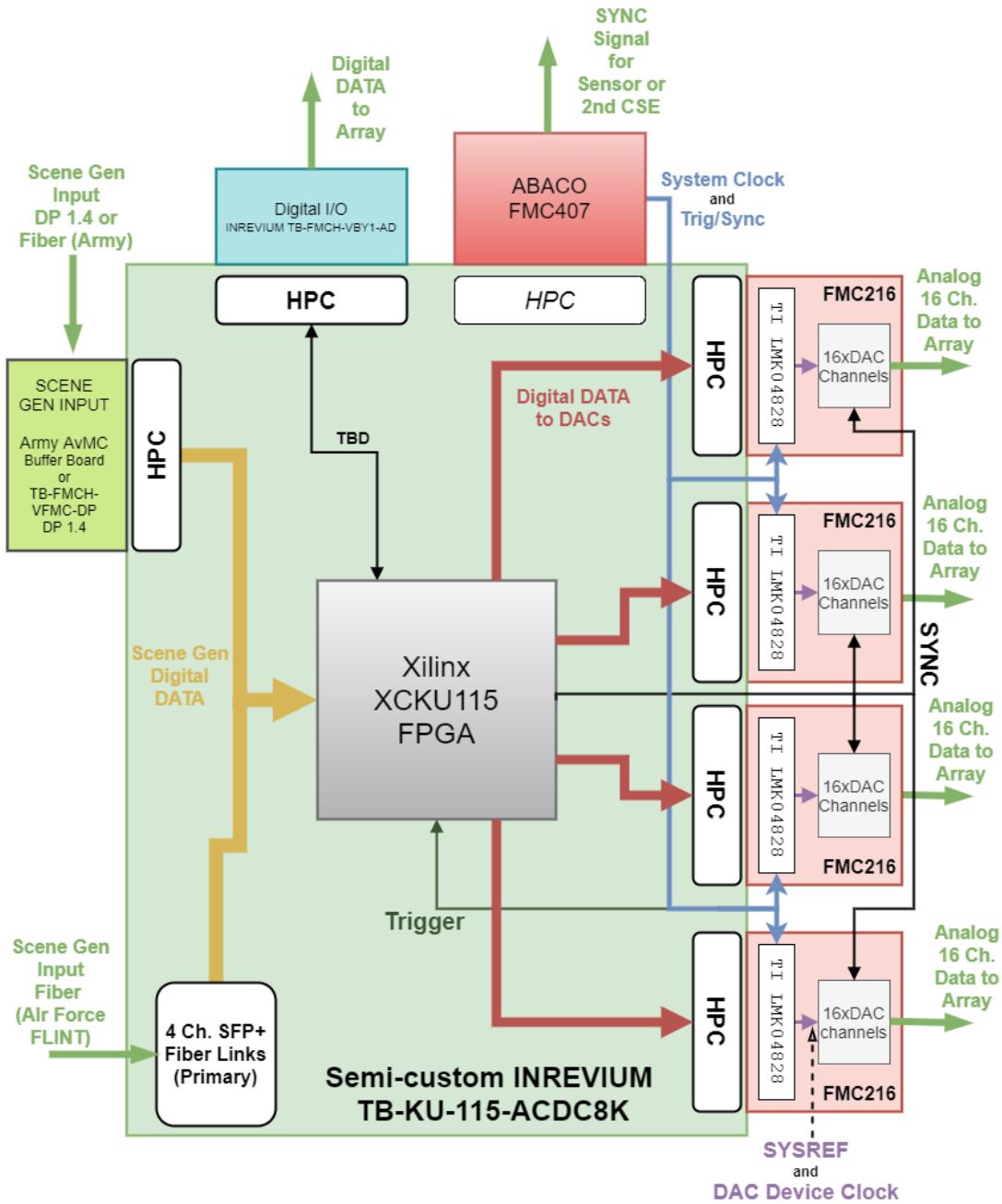


Figure 52 NESSIE2 Block Design

5.2.2 NESSIE2 Design

In the previous section, the general synchronization scheme to be utilized for the second-generation NESSIE CSE was described. In this section, the implementation details for hardware architecture of the so-called NESSIE2 CSE system is presented. Additionally, the motivation for each of the design decisions is briefly described, as well as the expected improvements to be expected compared to the existing CSE currently in use.

The hardware design of NESSIE2 consists of a central Ultrascale Kintex FPGA board with seven HPC FMC slots. This replaces the TB-6V-LX760-LSI board in the current CSE. Four of the FMC slots will house Abaco FMC216 PCB boards, each of which contains four 4-channel 16-bit 2.8 GSPS DAC39J84 chips [83, 84]. An Abaco FMC407 clock generation board is connected to another FMC slot to provide the JESD204B synchronization signals discussed in the previous section. A custom digital I/O card resides on another FMC slot, providing the digital signals for the hybrid currently delivered by the NESSIE interface boards. The last FMC slot is reserved for a custom gigabit I/O board in the case a customer would like to use their own method of delivering scene generation data to the NESSIE2 CSE. Otherwise, the default assumption is that the gigabit I/O take place on up to four on-board small form-factor pluggable plus (SFP+) data interface modules, which typically utilize high-speed serial connections like ethernet or fiber. A rack-mountable metal case will surround the system components, with two fans mounted on the far end for air flow.

Figure 53 shown below depicts the NESSIE2 hardware design.

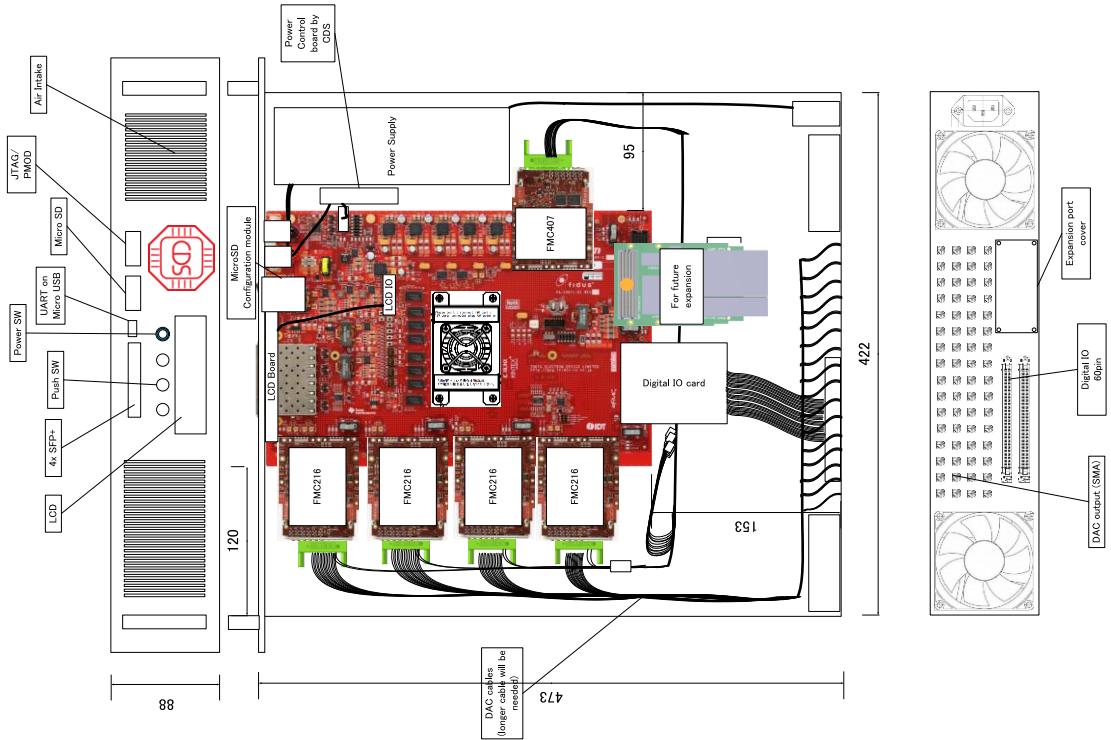


Figure 53 NESSIE2 Hardware Design

The design goals of the NESSIE2 were driven by a number of guiding principles that were devised to mitigate and solve problems encountered in the first-generation design, but also expand the CSE capabilities to support future scene generation resolutions and frame rates. The first principle of this design is that the main NESSIE board and peripherals would consist of commercial hardware. Any custom hardware would instead be migrated to reside on the carrier board to interface with the Dewar and RIIC hybrid. Some of the components of the first-generation NESSIE design were chosen nearly a decade ago, and as such the next design principle for NESSIE2 is to upgrade the control FPGA and serial DACs to more modern technology. In order to clean up signal integrity and improve analog response,

upgrades to the CSE cabling would be required, such as migrating from ribbon cables to serial data links. Finally, to better support emerging technology and support higher bandwidth requirements, the number of analog channels are doubled from 32 to 64. In the paragraphs that follow, the specific design choices for NESSIE2 will be related back to these guiding principles.

As discussed in Chapter 4, some aspects of the close support electronics were custom designed to accomplish the task at hand. This was very useful in some ways, particularly for learning purposes and the ability to design a component to perform a desired task exclusively. However, this approach also came with some drawbacks. The most notable of these drawbacks was the difficulty in troubleshooting, some of which was detailed in Chapter 4 for the custom Gigabit I/O boards. In a similar vein, customer support was much more difficult to acquire for a device made of several individual parts. Additionally, while some commercial components were difficult to impossible to acquire 5-10 years ago, as time went on these devices became more readily available, mitigating the need for having as many custom parts. Thus, one of the design principles of NESSIE2 was to move to commercial off the shelf (COTS) components wherever it was feasible.

Another goal of the next generation CSE was to improve the cabling in order to improve signal quality and analog response. Chapter 4 briefly describes some of the issues encountered regarding digital signal quality. The current CSE also uses ribbon cables, particularly from the interface boards to the dewar housing the RIIC hybrid. These cables are typically between 1-3 feet long and consist of more than 32 parallel analog signals. Driving the signals along these ribbon cables requires significant amplification of the signals coming from the DACs because these signals arriving at

the dewar are required to be at least 5V. In NESSIE2, the cable losses compensated by using a more modern cabling solution by Samtec. These new cables also better eliminate potential crosstalk with superior shielding between the individual signals. For the NESSIE2 digital signals, Samtec EQCD-040 Qstrip 50 Ω single ended routing, 38 AWG micro coax ribbon cables were chosen. Utilizing a double row system for up to 120 pins, these cables have high-speed performance rated for 14 Gbps and a -3 dB insertion loss at 487 MHz for a 1-meter-long cable. These digital signals connect from the digital I/O card on FMC5 to the RIIC adapter card on the dewar, replacing the connections from the current NESSIE's interface board. The analog cable chosen is a Samtec TCF-2650F 50 Ω 26 AWG micro coax cable with 1.29 ns/foot propagation delay and -3 dB insertion loss at 7 GHz for a 1-meter cable. These analog signals are sent from the FMC216 boards on FMC slots 1-4 and connect via two rows of 20 analog cables to the adapter card on the dewar, replacing the current CSE interface board ribbon cables. Figures 54-56 show the Samtec digital and analog cables.

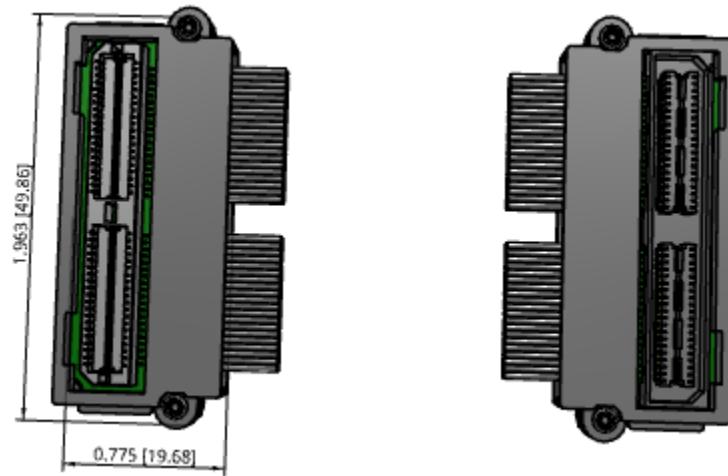


Figure 54 Samtec EQCD-040 Digital Cable

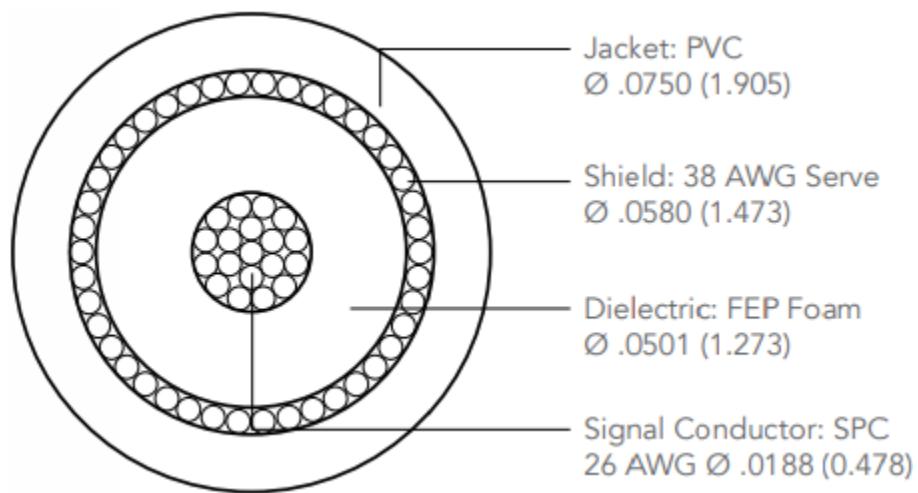


Figure 55 Samtec TCF-2650F Analog Cable

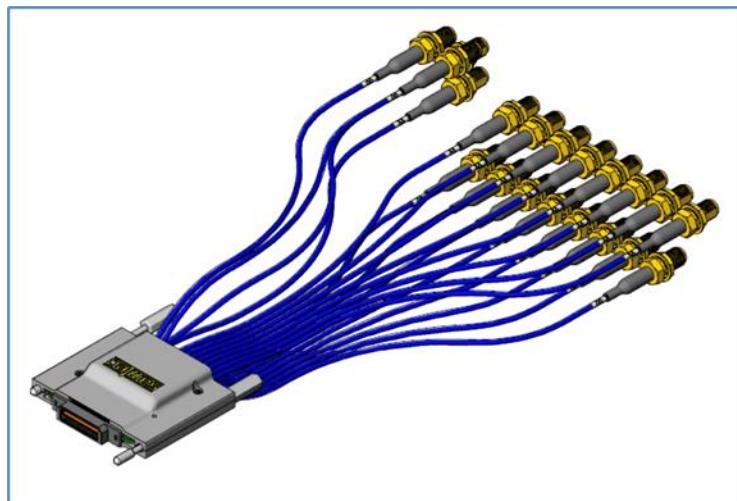


Figure 56 Two Rows of 20 Analog Cables, Samtec HDR-166968-01-EQRF

The digital signal format being sent over the Samtec cables is LVCMOS25, with logic level 2.5 V, drive configurable current of 4 mA, 8 mA, 12mA, or 16 mA, and configurable fast or slow slew rate from the Ultrascale Kintex FPGA [85]. The analog signal format from the FMC216's DACs to the Samtec cables utilizes DC

coupled outputs based on the AD8000 current feedback operation amplifier [83].

Figure 57 depicts the analog signal format.

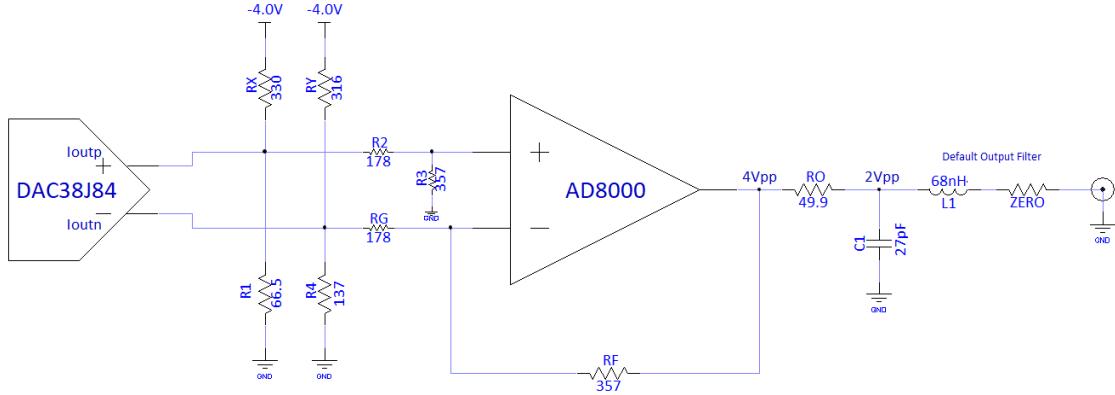


Figure 57 Analog Signal Format

The heart of the NESSIE2 is a custom Kintex Ultrascale FPGA PCB board currently under development by Inrevium. This board is meant to replace the current TB-6V-LX760-LSI and is based on the COTS TB-KU-115-ACDC8K containing a modern Xilinx XCKU115-2FLVA1517E FPGA. It contains seven HPC FMC slots (numbered 0 - 6), onboard DDR RAM, and four SFP+ cages for high-speed serial communication to replace the current scheme using a traditional display protocol like DVI. Customizations to the TB-KU-115-ACDC8K will be made by Inrevium to fit the needs of our IRSPs, such as adding an additional EEPROM and Peripheral Module (Pmod) for data storage. Additionally, an SD card will be added for programming along with an LCD display header. Basic firmware to test functionality of this custom board will be provided to us, which we can then modify for specialized functionality similar to existing CSE firmware. As seen in the hardware design diagram in Figure

53 above, the four FMC216 DAC boards will be connected to HPC FMC slots 1-4, and FMC slot 6 will be populated by the FMC407 board. FMC slot 0 is reserved for a custom Digital I/O board to handle the digital signals currently sent from the interface boards to the hybrid. Finally, FMC slot 5 can be used for a custom gigabit I/O board as desired by the customer, or even the older TB-FMCL-HDMI or ZYNQ-FMCL-HDMI board for backwards compatibility with the original NESSIE. For example, a potential IRSP customer the Army's Aviation and Missile Center (AvMC) is currently developing their own gigabit I/O PCB to interface with our CSE through an FMC slot. Each of these FMC slots have been carefully chosen such that the attached peripheral would have access to the appropriate high-speed transceiver. For example, the FMC216 boards route to FPGA banks that can access GTH transceiver quadrants that can send data at 12.5 Gbps on 8 channels. Another feature of the custom Kintex Ultrascale PCB board is that it is smaller than the TB-6V-LX760-LSI, allowing for the NESSIE2 case to be half the physical size of the current CSE, making for easier rack mounting. See Figure 58 for the control board design.

Custom TB-KU-115-ACDC8K

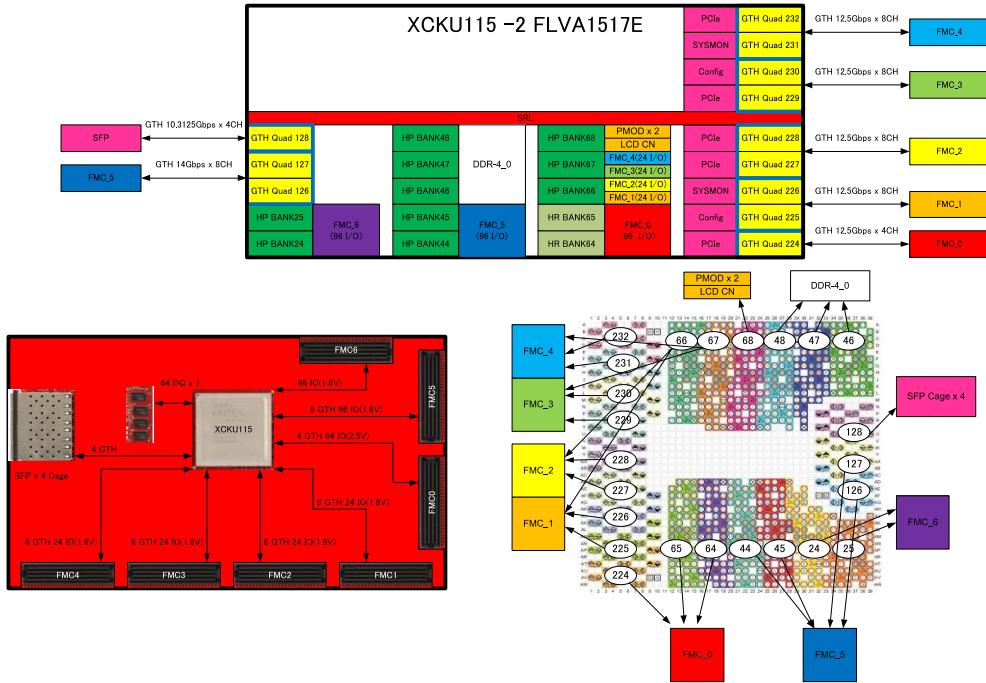


Figure 58 Custom Kintex Ultrascale FPGA Control Board

A basic overview of the NESSIE2 firmware can be seen in Figure 59 below. In the default configuration, scene generator data arrived into the custom Kintex Ultrascale FPGA control board through up to four SFP+ connectors as high-speed serial links. This data is then decoded by a Xilinx Aurora 64b/66b IP core. (64b/66b encoding is similar to 8b/10b encoding described previously.) This data is then sent to a custom PDP decoder block that determines where to send the array data and how fast to update each region of the array based on the information in the PDP packet header. Although the PDP frontend and backend will need to be altered due to the different expected input and output formats in NESSIE2, the PDP decoder firmware module is functionally identical to the current system. The scene data is then sent to up to four

JESD204B firmware interfaces to connect, interface and synchronize with the FMC216 DAC boards. Trigger and clock signals from the FMC407 are tied to each FMC216 to synchronize them to each other and the rest of the NESSIE2.

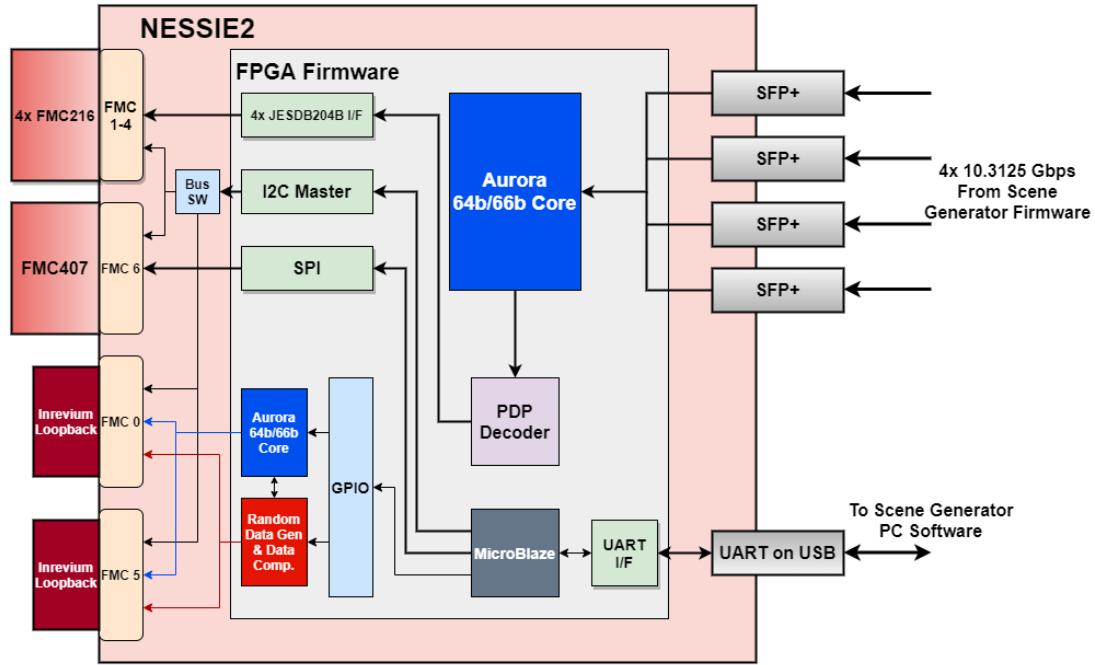


Figure 59 Firmware Block Diagram

A Xilinx MicroBlaze soft processor is instantiated in the FPGA firmware logic, which performs similar functions in this design to the ZYNQ processors described earlier in this work. The MicroBlaze is connected to several other peripherals through an AXI interface and can control these peripherals or write data to registers through a high-level software stack. On the AXI interface, there is a UART interface that connects to a UART-USB convert to send output operational logs to a control PC. An I2C Master, as well as an SPI module on the AXI interface, are used to

send signals to the FMC slots containing the FMC216 and FMC407 boards. Inrevium does not have access to all of the peripheral boards that will make up the NESSIE2 system, so testing of the FMC slots is done by sending control signals over AXI to a GPIO block. This GPIO block, along with random data generators and comparators, is used to send data through an Aurora IP core to the FMC slot under test, which is then looped back through the FMC slot where the comparators confirm the data is correct.

5.2.3 NESSIE2 FMC216 Testing and Verification

The NESSIE2 system described above consists of several modular components, many of which can be independently tested and verified. At the time of this writing, Inrevium has not yet completed the custom Kintex Ultrascale FPGA control board described above. However, testing one of the peripheral components of the proposed NESSIE2 could be accomplished using a different FPGA development kit. In this section, the initial testing of the Abaco FMC216 DAC board is described. Abaco was kind enough to allow us a 1-month evaluation period for the FMC216 product. Furthermore, Xilinx generously donated a VC707 evaluation kit, which was used in place of the custom Kintex Ultrascale board for the FMC216 testing. Along with the FMC216, an Anaconda breakout SMA breakout cable was also acquired from Abaco to ease testing procedures.

A test station was set up in one of our labs consisting of a desktop PC, an oscilloscope, the aforementioned VC707 FPGA Evaluation board, and the Abaco FMC216 DAC board. Initial tests were done without the Anaconda breakout cable, which was scheduled to arrive about 10 days after the arrival of the FMC216. The first step of the procedure was to load the board support package (BSP) provided by Abaco onto the VC707. The BSP essentially consists of drivers and specific IP that allow the

VC707 to readily communicate with the FMC216. Perhaps most importantly, the BSP includes a subset version of the JESD204B IP core to specifically support the DACs on the FMC216. On the lab PC, the Xilinx Vivado Design Suite was used to program the VC707 with a sample bit file to configure the FPGA using the BSP to communicate with the FMC216 DACs through the JESD204B IP core. Additionally, the PC would run a test application in C which could produce various digital signals that the DACs would convert to analog signals, which would then be probed on the oscilloscope. This test application by default produced a simple sine wave as shown in Figure 60 below. The application could then be readily modified to produce various other types of signals such as square waves or saw tooth waves, with different frequencies and amplitudes. Modifying the various parameters and observing the output would allow us to evaluate whether or not the FMC216's performance matched the specifications and if it would be suitable for the signals expected in the CSE.

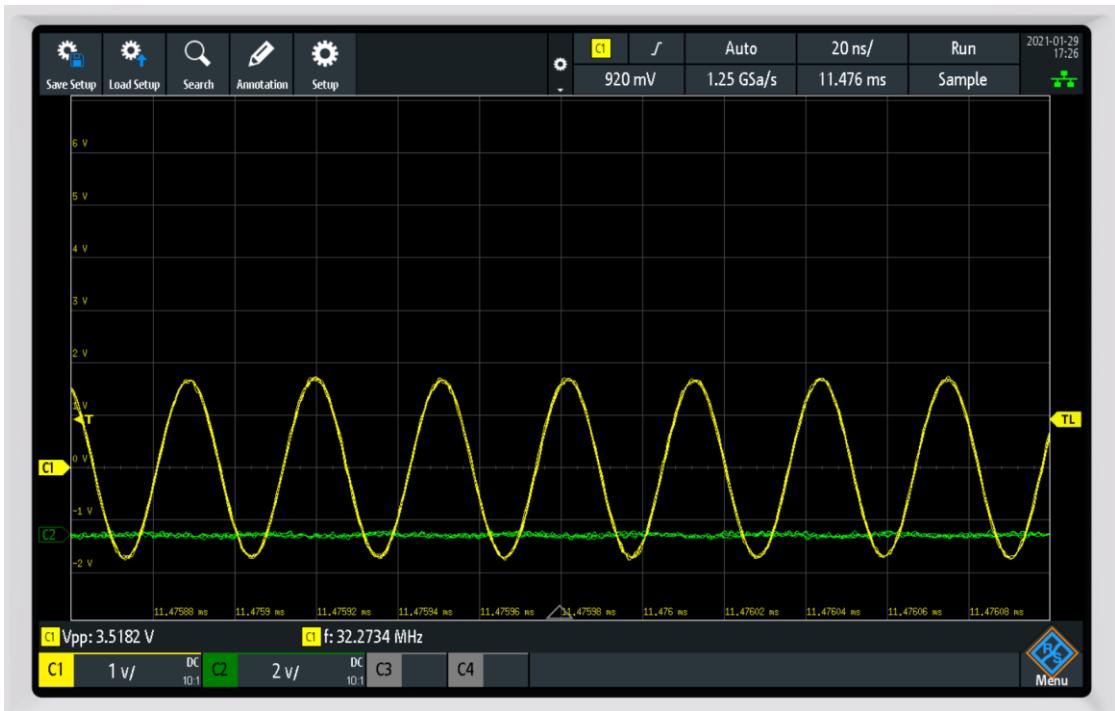


Figure 60 Sine Wave Abaco DACs

The initial test using the default parameters of the test application, VC707, and DAC hardware produced a 32 MHz sine wave with amplitude approximately 3.5 Vpp, as shown in Figure 60 above. The Anaconda breakout cable had not arrived yet, so the probing and grounding was done manually. We set up the remote workstation early in the week and left the VC707 and Abaco FMC216 powered on in case anyone wanted to access the system remotely during the week. However, when we tested the system physically at the end of the week, we observed that the Abaco FMC216 was hot to the touch, even though the VC707 had not been programmed and no data had been sent through the DACs. When running the evaluation program, it reported that the DACs reached temperatures as high as 85° C. After powering down the system for several minutes and trying again, temperatures were more reasonable, reported by the program

around 45° C at start up and increasing to around 65° C over the course of the day of testing. We installed a small fan near the system to help mitigate the heating issue.



Figure 61 Square Wave Abaco DACs

The next step in the testing process was to modify the software application to produce a square wave instead of a sine wave, as this would be more representative of the kinds of signals used in the CSE. Initial results at 64 MHz failed, resulting in a signal that was reminiscent of a noisy sine wave. Lowering the frequency of the signal resulted in signals with more of a square wave shape but with significant ringing. Upon lowering the frequency to 4 MHz, a clean square wave with amplitude 2.2 Vpp was attained, as shown in Figure 61 above.

The next round of testing occurred after the arrival of the Anaconda SMA breakout cables. This made the measuring easier, allowing us to readily connect the FMC216 directly to the oscilloscope. This also allowed for remote testing without someone needing to be physically in the lab to probe the board. This also made the measurements more consistent because the grounding of the signals was taken care of through the cables rather than through manual probes. As a result of these improvements, we endeavored to push the limits of the FMC216 to see how high of a frequency we could attain while maintaining a usable signal. We attained a relatively clean square wave with amplitude 2.6 Vpp when we pushed the frequency up to 16 MHz, as seen the Figure 62 below. We were also able to push the square wave up to 32 MHz and this signal may also be usable, but the signal integrity is noticeably degraded (see Figure 63). We are able to push the sine wave with amplitude 3.1 Vpp up to frequencies as high as 128 MHz, as seen in Figure 64. This is consistent with the FMC216 documentation listing analog output bandwidth as -3.0 dB at 120 MHz. [83]

At this point, it may be helpful to recall the bandwidth targets that will be required for the initiatives that NESSIE2 will be used for. For example, the goal of the HSLEDS program is to produce scenery with resolution 1K x 1K at 1 KHz. A rough calculation of the minimum required bandwidth to attain this is shown below.

$$\frac{(1024 * 1024) \text{ pixels} * 1000 \text{ Hz}}{64 \text{ channels}} = 16.384 \text{ MHz} \quad (5.1)$$

Since the minimum bandwidth is on the order of 16 MHz, the testing of the FMC216 shows that even using the square wave results, the DACs should have sufficient bandwidth while maintaining acceptable signal integrity. Finally, one concerning aspect was the high operating temperatures observed at times. We noticed

temperatures as high as 111° C, possibly necessitating the addition of a heat sink or similar temperature mitigation measures.



Figure 62 Square wave using Anaconda cables connected directly from FMC216 to the scope. DACs channels 1 (yellow) and 13 (blue) were observed.



Figure 63 Fastest square wave observed. DACs channels 1 (yellow) and 13 (blue).

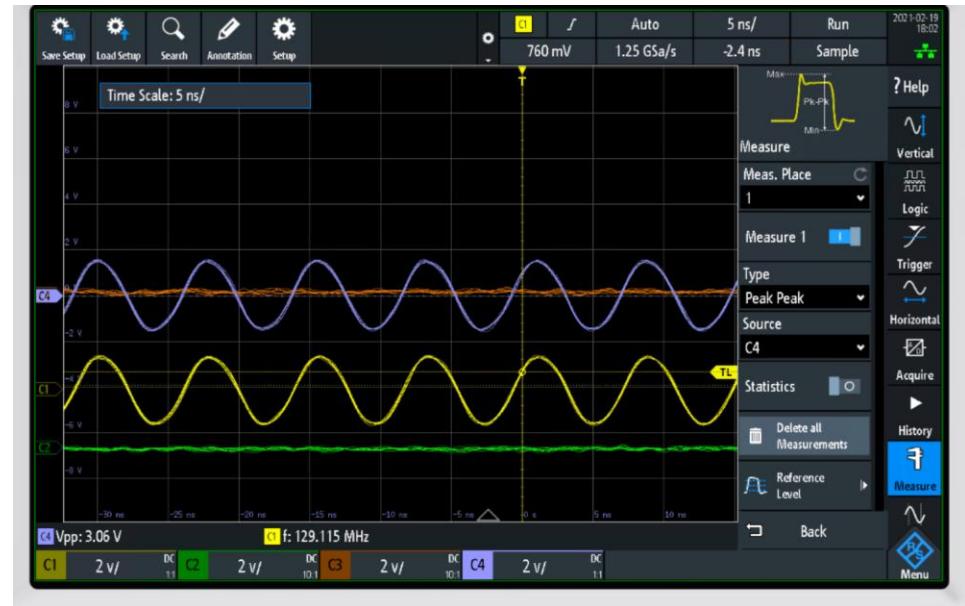


Figure 64 Fastest square sine wave observed using Anaconda cables connected directly from FMC216 to the scope. DACs channels 1 (yellow) and 13 (blue) were observed.

The custom Kintex Ultrascale board currently under development by Inrevium is scheduled to arrive by the end of 2021, assuming no additional delays due to the current worldwide semiconductor shortage. Upon arrival of this board, there is a plan to test it in its default configuration where the scene generator data is expected to arrive over the onboard SFP+ connectors by utilizing a Xilinx Ultrascale KCU116 evaluation board that we have on hand. Scene data will be sent over four SFP optical cable channels at up to 10.3125 Gbps each for a total of over 40 Gbps bandwidth. This setup can be used to test the modified PDP frontend to leverage the lack of blanking periods on the high-speed serial links as opposed to the previous implementation using HDMI. Figure 65 shows the initial Kintex test setup.

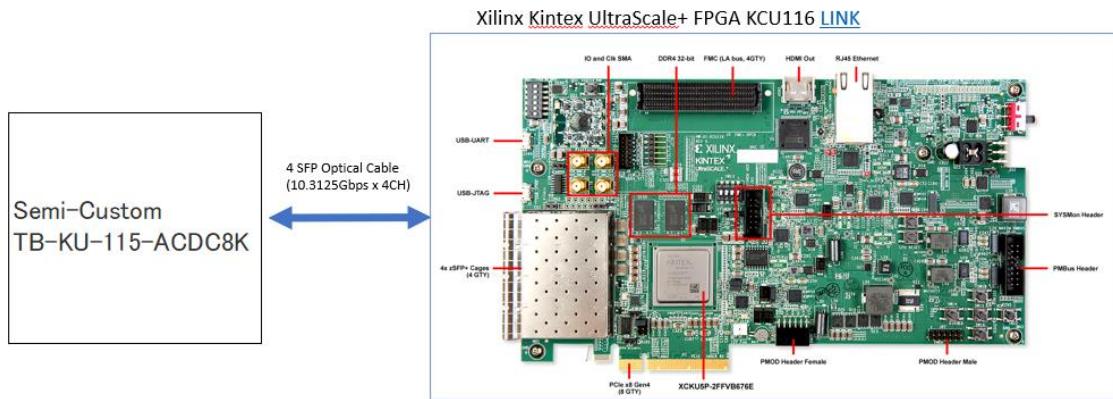


Figure 65 NESSIE2 Initial Test Bed

Chapter 6

PROPOSED DISTRIBUTED CSE SYSTEM

6.1 Advantages of Distributed CSE Design

As IRLED IRSP technology advances to support the large resolutions of projects such as HSLEDS, AIREA, and beyond, it becomes desirable to reconsider the CSE control scheme for more efficient operation. Heretofore, the control scheme described was to use one or more CSE systems to drive the RIIC, which was separated into four quadrants. Recall that for a program like NSLEDS, a single CSE controls all four quadrants of an array hybrid of up to 1K x 1K resolution, whereas for HDILED, each quadrant of a 2K x 2K array hybrid is assigned to one of four CSE systems. The disadvantage of this approach is that each CSE requires a cumbersome number of analog channels, as there are currently 32 channels per CSE and the NESSIE2 system described in the previous chapter contains a whopping 64 channels. However, future RIIC hybrid architectures need not be constrained to a quadrant-based control scheme. Indeed, one could logically split the array into a much larger number of smaller pieces, each of which could be controlled by a CSE with a smaller number of analog channels. One such scheme is described in [5], but any number of similar schemes could also be considered. In order to support these new control schemes, we expand on the lessons learned in the multi-CSE approach of HDILED to move away from a single monolithic CSE to a larger number of smaller autonomous CSEs, which shall

be referred to as a set of *distributed* CSEs [5, 93]. Advantages of this approach include (1) the aforementioned reduction in analog signals per CSE thus greatly reducing complexity, (2) moving from a large FPGA to several smaller FPGAs reduces both cost and complexity, and (3) more fine-grained control of the array by addressing only the necessary CSEs helps enable the packetized display protocol.

6.2 NESSIE3 Design Details

This section describes the design considerations and details for the next generation NESSIE3 CSE architecture. First, an intermediate design is presented which is meant to bridge the gap between the current and next CSE generations while maintaining compatibility with existing scene generation and IR sensor technology. Next the proposed NESSIE3 architecture is presented, including requirements, components, PDP implications, and PCB considerations.

6.2.1 NESSIE2.5

Reimplementing the NESSIE architecture into a distributed CSE system could potentially be a large and expensive undertaking prone to errors and growing pains. One difficulty in developing new IRLED systems is maintaining compatibility with the existing systems that are currently under active research and development. A proposed first step in the development of the distributed CSE scheme shall be dubbed NESSIE2.5, which builds on the system described in the previous chapter. The idea of NESSIE2.5 is very simple – instead of using the large KCU FPGA on the custom board, a small set of DAC boards similar to the FMC216 are developed. However, each of these DAC board will also include a small control FPGA responsible for its own set of DAC signals. As such, each of these small custom boards would essentially

be its own so-called mini-CSE and part of a rudimentary distributed architecture. The rest of the system would remain the same, so that the RIIC control scheme would still be assumed to be quadrant-based, and each of these rather large mini-CSEs would have up to 16 analog channels. The functionality of the KCU FPGA would be largely off loaded to the smaller FPGAs instead. See Figure 66 for NESSIE2.5 concept.

NESSIE 2.5

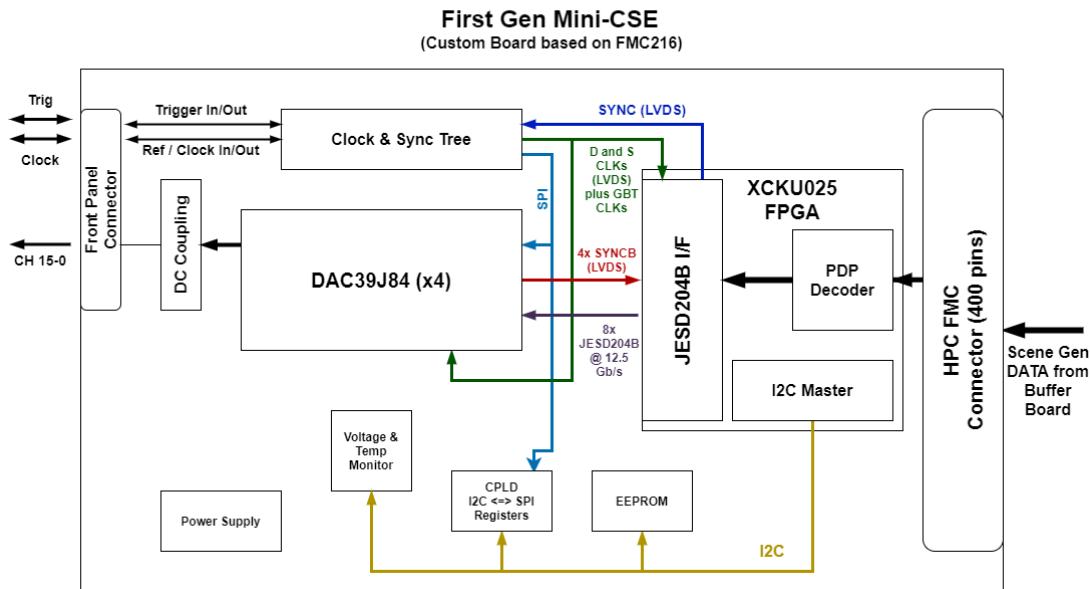


Figure 66 NESSIE2.5

The NESSIE2 mini-CSE PCB connects to the custom control board through a high pin count FMC connect in a manner nearly identical to the FMC216 in the NESSIE2 design described in the previous chapter. This mini-CSE has all of the functionality of the FMC216 while also adding additional capabilities through a small

on-board FPGA. For the purposes of this discussion, a Xilinx XCKU025 FPGA was chosen, but any number of similar FPGA devices could potentially be substituted. While the XCKU025 is less powerful than the beastly XCKU115 FPGA contained on the control board of the NESSIE2 design, it is also significantly less expensive. To put this in perspective, each XCKU025 is about 20-25% as powerful as the XCKU115, but the XCKU115 costs about nine times as much. A comparison of some FPGA candidates to the XCKU115 can be found in Table 3 below.

Table 3 NESSIE2.5 FPGA Candidates Comparison

	XC7K70T	XCKU025	XCKU035	XCKU115
Logic Cells	65.6K	318K	444K	1451K
Memory	4.86 Mb	12.7 Mb	19.0 Mb	75.9 Mb
DSP Slices	240	1152	1700	5520
Max transceivers	8 (GTX)	12 (GTH)	16 (GTH)	64 (GTH)
Transceiver speed(Gb/s)	12.6	16.3	16.3	16.3
I/O Pins	300	312	520	832
Cost (USD)	~200	~1000	~1200	~9000

The primary functionality of the firmware on NESSIE2's XCKU115 FPGA would instead be off-loaded to the individual XCKU025 FPGAs on each mini-CSE in the NESSIE2.5 design. As shown in Figure 66 above, these firmware blocks include – but are not necessarily limited to - an I2C Master, a JESD204B interface, and a custom PDP decoder. The I2C master module has functionality similar to its role in NESSIE2, such that it is interfaces with the EEPROM on the PCB for non-volatile programming, and the SPI and I2C registers. Another potential role of this module would be to connect to an on-board temperature and voltage monitor. This is desirable since the

testing of the FMC216 described in the previous chapter revealed that the DAC temperatures can reach high levels which could adversely affect performance. The JESB204B interface module controls the synchronization between the DACs on the different mini-CSE boards. The JESD204B standard for synchronization was described in the previous chapter and is essentially unchanged in NESSIE2.5. The I2C master and JESD204B interface are industry standard IPs and as such most likely do not need to be fully developed in-house. The last module however, called the PDP decoder, is a custom firmware IP block specific to our systems. The purpose of this block is to decode the headers of the PDP packets (see Chapter 3) to determine where to draw pixels on the array and how fast to update them. Unlike the original NESSIE, NESSIE2 and beyond are designed to be able to accept high-speed serial data from a scene generator that is not necessarily tied to a display protocol, such as the fiber optic connections through the on-board SFP+ cages on the main control board. In NESSIE2.5, this data is sent to each mini-CSE through a HPC FMC. By utilizing our packetized display protocol on a data stream without the drawbacks of a traditional display protocol, we can not only minimize or eliminate frame blanking periods, but also leverage techniques from computer networking to deliver data most efficiently to the desired portions of the array. Each mini-CSE is responsible for writing to a specific logical sub-section of the array, and the PDP decoder deciphers the packet data to determine whether to forward the data to the array or drop the packet.

6.2.2 Design Considerations and Requirements

The primary design consideration for the future NESSIE3 system is to break away from the current approach of having a large, monolithic “super FPGA” approach to control the CSE to move to a distributed system of smaller, largely independent

systems. Larger and more powerful FPGAs are significantly more expensive than smaller ones, to the point where it is often less expensive to purchase many smaller FPGAs instead. An example of this was seen in the NESSIE2.5 description above. Additionally, as the monolithic system becomes more complex, the firmware to control it becomes more unwieldy and prone to errors. The current firmware for NESSIE contains thousands of lines of code. Not only is it difficult to maintain this code, but it is also more difficult to teach the intricacies to new researchers. NESSIE3 will instead have smaller FPGAs, each with less complex functionality for their firmware in order to achieve a specialized task (such as PDP decoding).

One early consideration for NESSIE3 was to make use of the recent advances in Xilinx FPGAs known as radio frequency system on a chip (RFSoC) [87]. These devices are quite convenient because they contain an FPGA along with on-board DACs, similar to how the ZYNQ SoC contains an on-board microprocessor. It would be incredibly convenient to control the FPGA and DACs with the same firmware. Synchronization would be much easier due to not needing to worry about propagation delays along long PCB traces. However, at the current time, there are a number of reasons why the RFSoC FPGAs may not be desirable for NESSIE3. At the time of this writing, RFSoC systems contain 14-bit RF-DACs, but IRSPs customers have always expressed interest in 16-bit pixel resolution. The DACs on NESSIE2's FMC216 boards are 16-bit, and even the current NESSIE system contains 16-bit DACs. RFSoC was also considered for NESSIE2, but currently each FPGA only has eight or at most sixteen DAC channels per FPGA. This limitation would not be as much of a problem for NESSIE3, if not for the next factor which is cost. At the moment, RFSoC FPGA devices are very expensive, such that it is cheaper to use a standard FPGA and a PCB

with 16-bit DACs of our choosing than to migrate to RFSoC at this time. See Table 4 for RFSoC comparison and evaluation board costs. At the time of this writing, individual RFSoC FPGAs are difficult to find for sale, so Table 4 lists the prices of Xilinx RFSoC evaluation boards. In an evaluation board, the FPGA itself takes up the lion's share of the price. For example, the XCKU115 individually costs around 6000 USD, whereas the TB-KU-115 board costs approximately 12000 USD [88]. For the purposes of the above discussion, it was assumed that whenever they become available, the RFSoC FPGAs would cost approximately 25-50% of the associated evaluation board. Even using this approximation, the expected RFSoC FPGA prices far exceed non-RFSoC FPGAs, such as those listed in Table 3. One interesting note is that the most recent RFSoC evaluation board for the XCZU49DR FPGA is substantially less expensive than expected, as seen in the last column of Table 4. This is a promising sign that these devices may become more cost effective in the near future. Since NESSIE3 is a future project, the prices of RFSoC devices could decrease, the bit resolution on RF-DACs could increase, and the DACs per FPGA device could increase. In the event of such improvements to RFSoC in the next few years, it may well be worthwhile to revisit this technology as an option for NESSIE3.

Table 4 RFSoC FPGA Comparison

	ZU28DR	ZU29DR	ZU39DR	ZU49DR
Logic Cells	930 K	930 K	930 K	930 K
Block Memory	38 Mb	38 Mb	38 Mb	38 Mb
DSP Slices	4272	4272	4272	4272
14-bit RF-DACs	8	16	16	16
DAC Rate (GSPS)	6.554	6.554	6.554	10.0
Eval Board Cost (USD)	8995	19995	24995	11995

6.2.3 NESSIE3 Design Components

The NESSIE3 system primarily consists of a host printed circuit board similar to the motherboard in a PC, which in turn houses a host of smaller PCBs which will be called “mini-CSEs”. Each mini-CSE plugs into the motherboard in a modular fashion, and the user would then choose to use as many mini-CSEs as necessary for their use case application. The NESSIE3 will contain an on-board clock generation module to replace the NESSIE2’s FMC407 board. This board provides the trigger signals to synchronize the DACs on each mini-CSE and sync signal for synchronization to additional motherboards if necessary. It also provides the system clock for all of the mini-CSEs. Scene generator data would enter the motherboard either through a custom gigabit I/O card attached to an FMC, similar to the current NESSIE, or through a number of high-speed serial connections such as SFP connectors as is done in NESSIE2. A diagram of the basic NESSIE3 block diagram is shown in Figure 67.

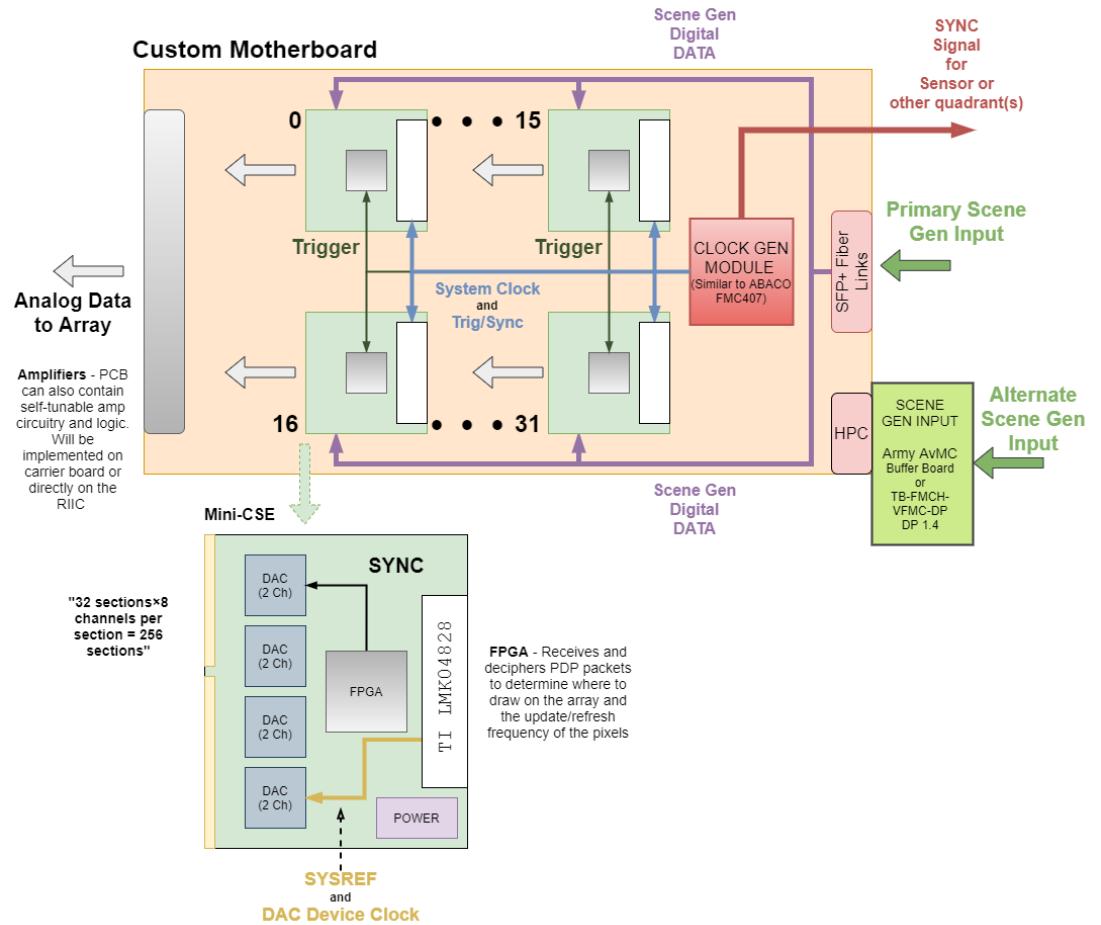


Figure 67 NESSIE3 Initial Design

6.3 Implications for PDP

As described in Chapter 3, the packetized display protocol (PDP) has a distinct advantage over more traditional display protocols because it is capable of agilely writing data to any portion of the frame on demand. This allows for potentially dynamic refresh rates even at the sub-frame level, such that fast imagery and slow imagery in the same scene can be updated at different rates. While PDP is transport layer agnostic in its theoretical design, for the purposes of infrared scene projectors it

would be helpful to have a system that can maximize the strengths of the protocol. Most notably, NESSIE3 distributed CSE system enables a much wider variety of writing schemes beyond writing in a raster fashion. As described in Chapter 5, a multi-CSE prototype was developed which could write to each quadrant of a SLEDs array independently. NESSIE3 expands on this functionality, providing the means to independently control a number of regions based on the number of mini-CSEs utilized. These regions could be quadrants, or smaller square or rectangular regions (such as splitting the frame into 16ths for example), or rows, columns, or other shapes as desired by the user. This agility in the hardware configurations in turn eases the implementation of PDP on a NESSIE3 system.

6.3.1 Protocol Write Flexibility

The current generation of SLEDs family IRLED scene projectors has limited bit resolution and a maximum of thirty-two analog channels [7, 89]. The current NESSIE CSE is capable of 16-bit resolution, however at present the analog circuitry on the array side does not support resolutions in excess of 11-12 bits [44]. Improvements to the analog chain are currently underway through upgrades to the RIIC and its supporting carrier board designs, but these are beyond the scope of this work [89, 90]. Suffice it to say that NESSIE2, NESSIE3 and beyond should continue to support 16-bit or higher resolution to support future IRSP systems.

The current NESSIE contains 32 of these 16-bit analog output channels, whereas NESSIE2 upgrades this to 64 channels. Each of these channels writes an intensity value to a single pixel, producing a dynamic range of values between 0 and 65535. The total number of analog channels represents the maximum number of pixels that can be updated in parallel in a single write operation. As far as PDP is concerned,

a larger number of analog channels means the ability to potentially update sub-frames of the array faster (i.e. with a fewer number of write operations). This in turn results in support for larger sub-frame regions and faster apparent frame-rates for the regions.

The distributed CSE nature of NESSIE3 provides the benefits of a large number of analog channels while avoiding some of the pitfalls of forcing more channels onto a single system. Each mini-CSE contains a more reasonable number of analog channels, such as 8, 16, or maybe 32, and operates quasi-independently of the others. The mini-CSEs each control their own small logical section of the array. This provides another very important benefit for PDP in the ability to write to different portions of the IRLED array simultaneously.

6.3.2 Routing to Each Mini-CSE

The distributed CSE NESSIE3 platform is extremely well-suited for the packetized display protocol. Recall that in PDP, the data for a region of pixels is encapsulated into a packet, which starts with a header with the locations of the start and end x and y coordinates of the region. The segmentation of a scene into sub-regions of potentially different apparent frame-rates is handled by the *compositor*, which is part of the PDP frontend [60]. Faster updating regions have packets that are sent on the data stream more frequently, and these apparent frame-rate values can change dynamically as the scene changes by adjusting the packet transfer rates. The NESSIE3 motherboard can then act in a manner similar to a network switch. A network switch is a hardware device that receives and forwards packet data to the appropriate destination device [91, 92]. Network switches are most commonly used for Ethernet and are a well-established means of managing data flow across a network [91]. The NESSIE3 CSE takes in scene generation data along high-speed serial links

after being parsed and packetized by the compositor, then can forward the packet to the correct mini-CSE based on the array location given in the packet header. The network switch could be implemented in firmware on a small FPGA, or perhaps more likely using dedicated network hardware repurposed for forwarding PDP packets using less intensive network protocols [5]. Figure 68 shows this scheme.

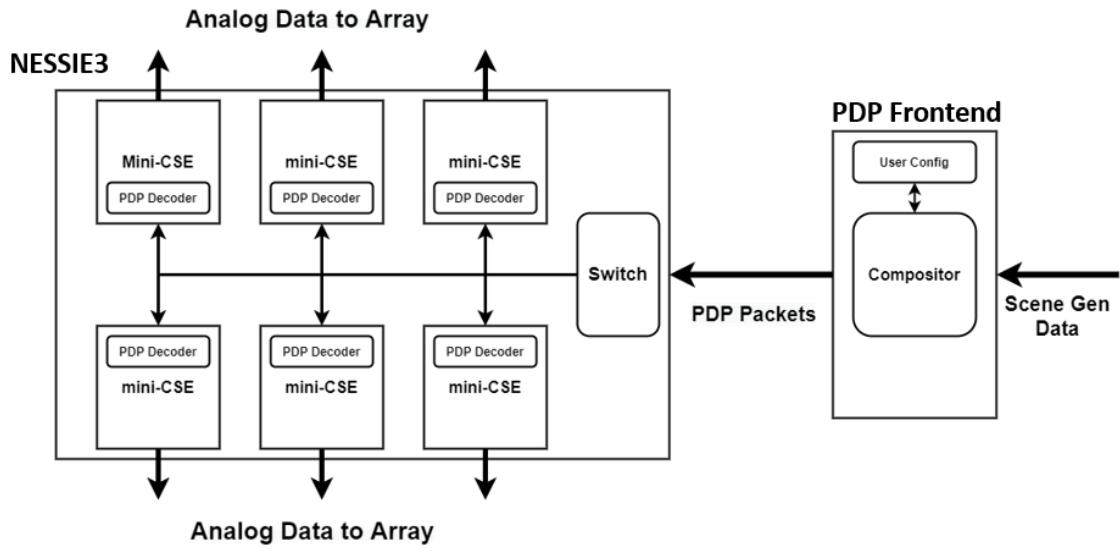


Figure 68 NESSIE3 PDP Packet Data Routing

Chapter 7

CONCLUSION

Infrared scene projectors are an invaluable tool for efficient and cost-effective hardware-in-the-loop testing of infrared sensors. Infrared LED scene projectors have been in development by Dr. Kiamilev's research group at the University of Delaware, in partnership with researchers from the University of Iowa, since 2008. IRLED-based IRSPs represent a key step forward in this field of research, providing the means to continue to support testing of sensors that demand faster, brighter, and higher resolution scenery. The close support electronics of these IRSPs are the vital components that ingest the scene data to be tested and convert it to a form suitable for display on the infrared projector array.

This thesis presented an overview of several key components of the current CSE as well as two architectures for supporting continued imagery demands for the immediate and foreseeable future. A novel packetized display protocol was briefly summarized as a means to overcome the bandwidth and control limitations of the traditional display protocols customarily used for scene generator data transfer. An extensive look at the current CSE design was presented, with particular attention given to the gigabit I/O interfaces used to receive input data to the projector from a scene generator. Next, a complete overhaul of the current CSE hardware architecture was introduced, designed for analog and digital channel improvements of 2x, bandwidth improvements in excess of 2.4x, and addressing synchronization concerns for larger and more complex systems moving forward. Finally, a custom CSE architecture based

on distributed FPGAs was presented to provide a roadmap for support electronics to continue to keep pace with future scene resolution and frame-rate demands. This design allows for PDP to realize its full potential for dynamic bandwidth allocation as needed to efficiently project changing scenery to different portions of the IRSP array.

Chapters 5 and 6 of this work in particular laid the groundwork for both short-term and long-term future work for SLEDs IRSP systems. In the immediate future, the next step will be to design and implement a new amplifier circuit to add to either the carrier board or the RIIC to support the DAC outputs from NESSIE2. Next, upon arrival of the NESSIE2 prototype system, the JESD204B synchronization scheme will need to be tested and confirmed. Another important task for NESSIE2 and NESSIE3 is to adjust the PDP frontend and backend to further take advantage of the improved CSE capabilities. These enhancements to the CSE hardware and improvements to the PDP firmware will allow our IRLED IRSPs to continue their trend of supporting higher resolution and faster frame-rate systems for years to come.

REFERENCES

- [1] International Commission on Non-Ionizing Radiation Protection, “ICNIRP Guidelines on Limits of Exposure to Incoherent Visible and Infrared Radiation,” *Health Physics* 105(1):74-96; 2013.
- [2] International Commission on Illumination, “ILV: International lighting vocabulary,” Vienna: CIE; 2011.
- [3] G. A. Ejzak, J. Dickason, J. A. Marks, K. Nabha, R. T. McGee, N. A. Waite, J. T. Benedict, M. A. Hernandez, S. R. Provence, D. T. Norton, J. P. Prineas, K.W. Goossen, F. E. Kiamilev, and T. F. Boggess, “ 512×512 , 100 Hz mid-wave infrared LED microdisplay system,” *J. Display Technol.*, vol. 12, pp. 1139–1144, Oct 2016.
- [4] M. A. Hernandez-Raya, *Higher Definition Mid-Wave Infrared Scene Projectors via Shrinking Pixel Pitch*. PhD thesis, University of Delaware, 2020.
- [5] G. A. Ejzak, *Novel Silicon Architecture for Increased Dynamic Range of Infrared LED Scene Projectors*. PhD thesis, University of Delaware, 2020.
- [6] D. T. Norton, *Type-II InAs/GaSb superlattice LEDs: applications for infrared scene projector systems*. PhD thesis, University of Iowa, 2013.
- [7] G. A. Ejzak, M. Hernandez, A. Landwehr, and F. E. Kiamilev, “Hybrid DAC approach to increasing dynamic range and signal to noise in IRSP systems,” *IEEE Research and Applications of Photonics in Defense*, Aug 2019.
- [8] N. C. Das, P. Shen, G. Simonis, J. Gomes, and K. Olver, “Light emitting diode arrays for HWIL sensor testing,” in *Technologies for Synthetic Environments: Hardware-in-the-Loop Testing X* (R. L. M. Jr., ed.), vol. 5785, pp. 14–23, International Society for Optics and Photonics, SPIE, 2005.

- [9] H. S. Lowry, III, D. H. Crider, W. H. Goethert, W. T. Bertrand, and S. L. Steely, "Scene projection developments in the AEDC space simulation chambers," in *Technologies for Synthetic Environments: Hardware-in-the-Loop Testing X* (R. L. M. Jr., ed.), vol. 5785, pp. 140–151, International Society for Optics and Photonics, SPIE, 2005
- [10] T. M. Canney, D. B. Beasley, M. Bender, T. Messer, D. A. Saylor, and J. Buford, Jr., "Progress in the development of a cold background flight motion simulator mounted infrared scene projector for use in the AMRDEC hardware-in-the-loop facilities," in *Technologies for Synthetic Environments: Hardware-in-the-Loop Testing X* (R. L. M. Jr., ed.), vol. 5785, pp. 152–162, International Society for Optics and Photonics, SPIE, 2005.
- [11] O. M. Williams, G. C. Goldsmith, II, and R. G. Stockbridge, "History of resistor array infrared projectors: hindsight is always 100% operability," in *Technologies for Synthetic Environments: Hardware-in-the-Loop Testing X* (R. L. M. Jr., ed.), vol. 5785, pp. 208–224, International Society for Optics and Photonics, SPIE, 2005
- [12] P. Bryant, S. Solomon, and J. James, "Bolometers running backward: the synergy between uncooled IR sensors and dynamic IR scene projectors," in *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XVII* (G. C. Holst, ed.), vol. 6207, pp. 207–218, International Society for Optics and Photonics, SPIE, 2006
- [13] B. E. Cole and C. J. Han, "Low power infrared scene projector array and method of manufacture," Feb. 4 1997. US Patent 5,600,148.
- [14] J. Marks, *Abuted IRLED infrared scene projection design and their characterization*. PhD thesis, University of Delaware, 2019.
- [15] S. B. Infrared, "Mirage dynamic IR scene projectors." <https://sbir.com/dynamic-ir-scene-projectors/>, 2019. Accessed on 2021-2-6.
- [16] V. Malyutenko, "Mid-IR LEDs project dynamic scenes," SPIE, 15 December 2016.
- [17] C. Wood, "Materials for thermoelectric energy conversion," *Reports on progress in physics*, vol. 51, no. 4, p. 459, 1988.

- [18] T. Danielson, G. Franks, N. Holmes, J. LaVeigne, G. Matis, S. McHugh, D. Norton, T. Vengel, J. Lannon, and S. Goodwin, “Achieving ultra-high temperatures with a resistive emitter array,” in *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXVII* (G. C. Holst and K. A. Krapels, eds.), vol. 9820, pp. 240–248, International Society for Optics and Photonics, SPIE, 2016.
- [19] E. J. Koerperick, *High power mid-wave and long-wave infrared light emitting diodes: device growth and applications*. PhD thesis, University of Iowa, 2009.
- [20] E. J. Koerperick, J. T. Olesberg, J. L. Hicks, J. P. Prineas, and T. F. Boggess, “Active region cascading for improved performance in InAs–GaSb superlattice LEDs,” *IEEE Journal of Quantum Electronics*, vol. 44, pp. 1242–1247, Dec 2008.
- [21] S. M. Sze and K. K. Ng, *Physics of Semiconductor Devices*. John Wiley & Sons, Ltd, 2006.
- [22] J. Benedict, R. McGee, J. Marks, K. Nabha, N. Waite, G. Ejzak, J. Dickason, H. Ahmed, M. Hernandez, P. Barakhshan, T. Browning, J. Volz, R. J. Ricker, F. Kiamilev, J. Prineas, and T. Boggess, “1kx1k resolution infrared LED scene projector at 24 μ m pixel pitch,” Government Microcircuit Applications and Critical Technology, Mar 2017.
- [23] D. T. Norton, Jr., J. LaVeigne, G. Franks, S. McHugh, T. Vengel, J. Oleson, M. Mac-Dougal, and D. Westerfeld, “Development of a high-definition IR LED scene projector,” in *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXVII* (G. C. Holst and K. A. Krapels, eds.), vol. 9820, pp. 222–228, International Society for Optics and Photonics, SPIE, 2016.
- [24] G. Ejzak, J. Dickason, J. Marks, J. Benedict, R. McGee, K. Nabha, A. Waite, H. Ahmed, M. Hernandez, P. Barakhshan, T. Browning, J. Volz, F. Kiamilev, and T. Boggess, “512x512, two-color infrared scene projector,” Government Microcircuit Applications and Critical Technology, Mar 2017.
- [25] R. Houser, H. Ahmed, K. Nabha, and F. Kiamilev. Modular system architecture as a foundation for rapid IRSP development. In *2018 IEEE Research and Applications of Photonics In Defense Conference (RAPID)*, pages 1-4, 2018

- [26] A. Landwehr, N. Waite, P. Barakhshan, J. Volz, and F. Kiamilev, "Non-uniformity correction (NUC) and 1KHz frame rate for IRLED scene projectors," Government Microcircuit Applications and Critical Technology Reno NV, 2017.
- [27] P. Barakhshan, G. Ejzak, K. Nabha, M. Hernandez, H. Ahmed, A. Landwehr, J. Lawler, and F. Kiamilev, "Test plan for IRLED scene projectors," Government Microcircuit Applications and Critical Technology Reno NV, 2017.
- [28] P. Barakhshan, J. Volz, R. J. Ricker, M. Hernandez, A. Landwehr, S. Provence, K. Nabha, R. Houser, J. P. Prineas, C. Campbell, F. Kiamilev, and T. F. Boggess. "End to end testing of IRLED projectors," In *2018 IEEE Research and Applications of Photonics In Defense Conference (RAPID)*, pages 1-4, Aug 2018.
- [29] T. Browning, C. Jackson, R. Houser, A. Landwehr, H. Ahmed, and F. Kiamilev, "A modular platform for rapid IRSP development," in *2019 IEEE Photonics Journal* Volume 11, Number 3, pp. 1-1, IEEE, 2019.
- [30] H. Ahmed, K. Nabha, J. Benedict, G. Ejzak, N. Waite, M. Hernandez, F. Kiamilev, C. Jackson, T. Browning, and B. Kathryn, "Modular and scalable firmware for infrared LED scene projectors," Government Microcircuit Applications and Critical Technology Orlando FL, 2016.
- [31] T. Browning, J. Volz, N. Waite, R. McGee, and F. Kiamilev, "Hardware acceleration of non-uniformity correction for high-performance real-time infrared LED scene projectors," Government Microcircuit Applications and Critical Technology Orlando FL, 2016.
- [32] G. A. Ejzak, J. Dickason, J. Benedict, H. Ahmed, R. McGee, K. Nabha, J. Marks, N. Waite, M. Hernandez, S. Cockerill, and F. Kiamilev. Scalable and modular architecture for close support electronics of an infrared scene projector. Government Microcircuit Applications and Critical Technology GOMAC Tech, Mar 2015.
- [33] C. Lange, R. McGee, N. Waite, R. Haislip, and F. E. Kiamilev. "System for driving 2D infrared emitter arrays at cryogenic temperatures," *Proc. SPIE*, 8015:801507, 2011.
- [34] Z. Marks, A. Waite, J. Marks, J. Dickason, R. McGee, and F. Kiamilev. "Advanced packaging for infrared scene projectors," Government Microcircuit Applications and Critical Technology GOMAC Tech, Mar 2017.

- [35] H. Ahmed, R. McGee, J. Marks, A. Waite, A. Landwehr, C. Jackson, G. Ejzak, T. Browning, P. Barakhshan, M. Hernandez, A. Deputy, T. Lassitter, C. Campbell, F. Kiamilev, J. Prineas, and E. Koerperick. “Fabrication, evaluation, and improvements of 1Kx1K and 2Kx2K infrared LED scene projector systems,” In *2019 IEEE Research and Applications of Photonics in Defense Conference (RAPID)*, pages 1-1, Aug 2019.
- [36] N. C. Das, M. Taysing-Lara, K. A. Olver, F. Kiamilev, J. P. Prineas, J. T. Olesberg, E. J. Koerperick, L. M. Murray, and T. F. Boggess. “Flip chip bonding of 68x68 MWIR LED arrays,” *IEEE Transactions on Electronics Packaging Manufacturing*, 32(1):9-13, Jan 2009.
- [37] D. T. Norton, J. T. Olesberg, R. T. McGee, N. A. Waite, J. Dickason, K. W. Goossen, J. Lawler, G. Sullivan, A. Ikhlassi, F. Kiamilev, E. J. Koerperick, L. M. Murray, J. P. Prineas, and T. F. Boggess. “512x512 individually addressable MWIR LED arrays based on type-II InAs/GaSb superlattices,” *IEEE Journal of Quantum Electronics*, 49(9):753-759, Sep 2013.
- [38] R. McGee, F. Kiamilev, N. Waite, J. Marks, K. Nabha, G. Ejzak, J. Dickason, J. Benedict, and M. Hernandez. “512x512, 100Hz mid-wave infrared LED scene projector,” Government Microcircuit Applications and Critical Technology GOMAC Tech, Mar 2015.
- [39] R. J. Ricker, S. Provence, L. M. Murray, D. T. Norton, J. T. Olesberg, J. P. Prineas, and T. F. Boggess. “512x512 array of dual-color InAs/GaSb superlattice light-emitting diodes,” In Jong Kyu Kim, Michael R. Krames, Li-Wei Tu, and Martin Strassburg, editors, *Light-Emitting Diodes: Materials, Devices, and Applications for Solid State Lighting XXI*, volume 10124, pages 201-206. International Society for Optics and Photonics, SPIE, 2017.
- [40] I. Plotog and M. Vladescu, “Power LED efficiency in relation to operating temperature,” in *Advanced Topics in Optoelectronics, Microelectronics, and Nanotechnologies VII* (I. Cristea, M. Vladescu, and R. Tamas, eds.), vol. 9258, pp. 652–657, International Society for Optics and Photonics, SPIE, 2015.
- [41] J. Marks, F. Kiamilev, R. McGee, T. Boggess, and N. Waite, “SLEDS technology: Present status and future,” Government Microcircuit Applications and Critical Technology, Mar 2016.

- [42] H. Ahmed, A. Landwehr, C. Jackson, T. Browning, P. Barakhshan, M. Hernandez, A. Deputy, T. Lassitter, C. Campbell, J. Singh, B. Steenkamer, R. McGee, A. Waite, F. Kiamilev, J. Prineas, M. Bellus, L. E. Koerperick, and L. Nichols. “Testing, instrumentation, and results to make the world’s first usable 1Kx1K infrared LED scene projector systems,” In *2020 IEEE Research and Applications of Photonics in Defense Conference (RAPID)*, pages 1-2, 2020.
- [43] J. Benedict, R. McGee, H. Ahmed, M. Hernandez, P. Barakhshan, R. Houser, J. Marks, K. Nabha, G. Ejzak, N. Waite, J. Dickason, T. Browning, C. Jackson, R. J. Ricker, A. Muhowski, R. Heise, F. Kiamilev, and J. Prineas. “4-megapixel infrared scene projector based on superlattice light emitting diodes,” Government Microcircuit Applications and Critical Technology GOMAC Tech, Mar 2018.
- [44] A. Landwehr, *A Packetized Display Protocol Architecture for Infrared Scene Projection Systems*. PhD thesis, University of Delaware, 2020.
- [45] P. Barakhshan, G. Ejzak, M. Hernandez, A. Landwehr, N. Waite, K. Nabha, F. Kiamilev, R. J., S. Provence, J. Prineas, and T. F. Boggess. “Gamma correction and operability of IRLED scene projectors,” Government Microcircuit Applications and Critical Technology GOMAC Tech, Mar 2018.
- [46] P. Barakhshan, M. Hernandez, J. Marks, N. Waite, and F. Kiamilev. “Non-uniformity detection and correction of IRLED scene projectors,” Government Microcircuit Applications and Critical Technology GOMAC Tech, Mar 2019.
- [47] P. Barakhshan, C. Campbell, M. Hernandez, and F. Kiamilev. “Evaluation of performance, non-uniformity and thermal limits for infrared LED scene projectors,” In *2019 IEEE Research and Applications of Photonics in Defense Conference (RAPID)*, pages 1-1, 2019.
- [48] M. Hernandez, J. Marks, E. Koerperick, P. Barakhshan, G. A. Ejzak, K. Nabha, J. Prineas, and F. Kiamilev, “Improving density and efficiency of infrared projectors,” *2019 IEEE Photonics Journal* Volume 11, Number 3, 2019.
- [49] M. Hernandez, J. Dickason, P. Barakhshan, J. Marks, G. Ejzak, A. Deputy, A. Waite, R. McGee, and F. Kiamilev, “Results of testing read-in integrated circuit (RIIC) wafers and hybrids,” Government Microcircuit Applications and Critical Technology Reno NV, 2017.

- [50] J. Marks, R. McGee, F. Kiamilev, R. Ricker, T. Boggess, J. Prineas, and G. Sullivan, "Design of an abutted hybrid system for infrared scene projection," *Government Microcircuit Applications and Critical Technology* Reno NV, 2017.
- [51] J. R. Grimes, W. Herald, R. A. Thompson and Eglin AFB. "Solving synchronization issues in scene simulation based on GPU-accelerated computing," *AIAA Modeling and Simulation Technologies Conference*, Kissimmee, FL, 2015.
- [52] "High-Definition Multimedia Interface Specification 1.3a," HDMI Licensing, LLC., <https://web.archive.org/web/20160305072940/http://www.microprocessor.org/HDMISpecification13a.pdf>, November 10, 2006. Accessed on 2020-12-4.
- [53] Video Electronics Standards Association. VESA coordinated video timings CTV standard 1.2. <https://vesa.org/vesa-standards/>, February 2013.
- [54] The Open Group. The X Window System™. <http://www.opengroup.org/tech/desktop/x-window-system/>, October 2020.
- [55] A. Landwehr, A. Waite, T. Browning, C. Jackson, R. Houser, H. Ahmed, and F. Kiamilev. "Toward a packetized display protocol architecture for IRLED projector systems," In *2018 IEEE Research and Applications of Photonics In Defense Conference (RAPID)*, pages 1-4, Aug 2018.
- [56] J. LaVeigne, M. Prewarski, G. Franks, S. McHugh. "A Hybrid Approach to Non-Uniformity Correction of Large Format Emitter Arrays," *Proc. SPIE 8356, Technologies for Synthetic Environments: Hardware-in-the-Loop XVII*, 83560D (10 May 2012)
- [57] T. Browning, C. Jackson, A. Landwehr, A. Waite, D. May and F. Kiamilev, "Architectural Enhancements of a Packetized Display Protocol for High-Speed IRSP Operation," *2020 IEEE Research and Applications of Photonics in Defense Conference (RAPID)*, pp. 1-2, 2020
- [58] C. Jackson, T. Browning, A. Landwehr, D. May, H. Ahmed, A. Waite, F. Kiamilev, "Demonstration of Packetized Display Protocol (PDP) to Overcome Speed and Resolution Limitations of Conventional Display Protocols," *2019 IEEE Research and Applications of Photonics in Defense Conference (RAPID)*, pp. 1-1, 2019

- [59] A. Landwehr, T. Browning, C. Jackson, D. May, A. Waite, H. Ahmed, F. Kiamilev, “An Implementation of a Packetized Display Protocol Architecture for IRLED Projector Systems,” *IEEE Photonics Journal*, vol. 11, no. 2, pp. 1-10, April 2019, Art no. 7000510. (2019)
- [60] T. Browning, *Enhancing and Characterizing a Packetized Display Protocol for Infrared Scene Projectors (IRSP)*. PhD thesis, University of Delaware, 2021.
- [61] "Digital Video Interface DVI Specification 1.0," Digital Display Working Group, http://www.cs.unc.edu/Research/stc/FAQs/Video/dvi_spec-V1_0.pdf, April 2, 1999. Accessed on 2020-12-6.
- [62] “MicroBlaze Processor Reference Guide UG984,” Xilinx, Inc., https://www.xilinx.com/support/documentation/sw_manuals/xilinx2020_2/ug984-vivado-microblaze-ref.pdf, November 18, 2020, Accessed on 2021-01-27
- [63] L. H. Crockett, R. A. Elliot, M. A. Enderwitz, R. W. Stawart, *The Zynq Book: Embedded Processing with the Arm Cortex-A9 on the Xilinx Zynq-7000 All Programmable SoC*. Strathclyde Academic Media, 2014.
- [64] Wikipedia contributors. Advanced eXtensible Interface - Wikipedia, the free encyclopedia, 2021. [Online; accessed 4-April-2021].
- [65] “Zynq-7000 All Programmable SoC Technical Reference Manual UG585,” Xilinx, Inc., December 6, 2017.
- [66] “TB-FMCL-HDMI Hardware User Manual Rev.3.00,” Inrevium Tokyo Electron Device Ltd., March 3, 2011.
- [67] “Zynq-7000 All Programmable SoC PCB Design Guide UG933,” Xilinx, Inc., September 27, 2016.
- [68] “Zynq-7000 All Programmable SoC Packaging and Pinout Product Specification UG865,” Xilinx, Inc., June 14, 2017.
- [69] “Recommended Design Rules and Strategies for BGA Devices User Guide UG1099,” Xilinx, Inc., March 1, 2016.
- [70] H. Johnson and M. Graham, *High Speed Digital Design: A Handbook of Black Magic*. Prentice Hall, Inc., 1993.

- [71] “Zynq-7000 SoC (Z-7007S, Z-7012S, Z-7014S, Z-7010, Z-7015, and Z-7020): DC and AC Switching Characteristics Product Specification DS187,” Xilinx, Inc., December 1, 2020
- [72] M. Fortunato, “Successful PCB Grounding with Mixed-Signal Chips - Follow the Path of Least Impedance,” Maxim Integrated Products, Inc., <https://www.maximintegrated.com/en/design/technical-documents/tutorials/5/5450.html>, October 15, 2012, Accessed on 2020-3-9.
- [73] M. I. Montrose, *Printed Circuit Board Design Techniques for EMC Compliance*. John Wiley & Sons, Inc., 2000.
- [74] S. Dabral and T. J. Maloney, *Basic ESD and I/O Design*. John Wiley & Sons, Inc., 1998.
- [75] Wikipedia contributors. Memtest86 - Wikipedia, the free encyclopedia, 2021. [Online; accessed 23-February-2021].
- [76] “ChipScope Pro Software and Cores User Guide UG029,” Xilinx, Inc., October 16, 2012.
- [77] T. L. Lassitter, J. Marks, J. Dickason, G. A. Ejzak, Z. Marks, A. Waite, and F. E. Kiamilev, “Modular carrier board and package for infrared LED arrays,” *IEEE Photonics Journal*, vol. 11, pp. 1–6, Aug 2019.
- [78] T. Lassitter, G. A. Ejzak, A. Waite, M. Hernandez, and F. E. Kiamilev, “Electronic & mechanical development of a multi-platform infrared LED scene projector system,” *IEEE Research and Applications of Photonics in Defense*, Aug 2020.
- [79] “JEDEC Standard No.204B Serial Interface for Data Converters,” JEDEC Solid State Technology Association, www.jedec.org, July 2011.
- [80] “A Guide to Multi-Channel Synchronization for MIMO Systems,” Abaco Systems Ltd., www.abaco.com, September 2018.
- [81] “Understanding JESD204B Subclasses and Deterministic Latency,” Texas Instruments Inc., www.ti.com, October 2012.
- [82] Wikipedia contributors. 8b/10b encoding - Wikipedia, the free encyclopedia, 2021. [Online; accessed 15-October-2020].
- [83] “FMC216 User Manual r1.1 UM061,” Abaco Systems Ltd., www.abaco.com, April 2016.

- [84] “DAC39J84 Quad-Channel, 16-Bit, 2.8 GSPS, Digital-to-Analog Converter with 12.5 Gbps JESD204B Interface,” Texas Instruments Inc., www.ti.com, January 2015.
- [85] “7 Series FPGAs SelectIO Resources User Guide UG471,” Xilinx, Inc., May 8, 2018.
- [86] J. Harris, “Understanding Layers in the JESD204B Specification—A High Speed ADC Perspective,” Analog Devices, Inc., www.analog.com, 2017.
- [87] “An Adaptable Direct RF-Sampling Solution WP489,” Xilinx, Inc., February 20, 2019.
- [88] “TB-KU-115-QUATTRO (ACDC Quattro Kintex UltraScale Development Platform),” KAGA FEI EUROPE Webshop, <https://shop.feeu.com/ACDC-Quattro-Kintex-UltraScale-Development-Platform>, Accessed on 2021-3-5.
- [89] T. Lassitter, G. A. Ejzak, A. Landwehr, C. Campbell, T. Browning, D. May, M. Hernandez, R. McGee, A. Waite, and F. E. Kiamilev, “Multiple Computer Support Electronics (CSE) Test Bed for Infrared Scene Projector Systems,” *IEEE Research and Applications of Photonics in Defense*, 2021, submitted.
- [90] J. Singh, F. E. Kiamilev, M. Hernandez, A. Landwehr, T. Browning, “Development of a Single, Scalable Testing Platform for Read-In Integrated Circuit LED Wafers,” *IEEE Research and Applications of Photonics in Defense*, 2021, submitted.
- [91] Wikipedia contributors. Network switch - Wikipedia, the free encyclopedia, 2021. [Online; accessed 27-March-2021].
- [92] “IEEE Standard for Local and metropolitan area networks: Media Access Control (MAC) Bridges,” in *IEEE Std 802.1D-2004* (Revision of IEEE Std 802.1D-1998) , vol., no., pp.1-281, 9 June 2004
- [93] M. Dreschmann, J. Heisswolf, M. Geiger, J. Becker and M. HauBecker, "A framework for multi-FPGA interconnection using multi gigabit transceivers," *2015 28th Symposium on Integrated Circuits and Systems Design (SBCCI)*, 2015, pp. 1-6.
- [94] “HDMI Design Guide,” Texas Instruments Inc., www.ti.com, June 2007.
- [95] “JESD204B Deterministic Latency,” Texas Instruments Inc., www.ti.com, 2016.

Appendix A

ADDITIONAL ZYNQ-FMCL-HDMI SCHEMATIC AND LAYOUTS

This supporting material for Section 4.2.3 contains additional schematic and PCB layout information for the ZYNQ-FMCL-HDMI revision 3 board.

The schematic for the Input/Output (I/O) Banks for the ZYNQ XC7Z020-1CLG484C Field Programmable Gate Array (FPGA) is shown in Figure A.1. There are four of these I/O banks that operate at either 2.5 V or 3.3 V logic levels. The FMC signals, HDMI signals, LEDs, switches, and global clock all connect to these banks. Additionally, the global clock signal sub-circuit is shown, based on an Integrated Device Technology Inc. XLH736100.000000I HCMOS crystal oscillator.

Figure A.2 depicts the DDR and Multiplexed I/O (MIO) Banks of the FPGA. The typical logic levels for each of these banks is 1.5 V and 1.8 V respectively, although the FPGA block RAM (BRAM) is limited to 1.0 V, and bank 500 can accept 3.3 V levels. The two DDR SDRAM integrated circuits, SD Card, and SPI Flash circuits described in Chapter 4 are all connected here. Additionally, a 3-pin UART jumper is tied to MIO bank 501 for outside communication. The Processing System (PS) clock is derived from a ECS Inc. ECS-2520S33-500-FN-TR crystal oscillator. The PS reset subcircuit is also shown, simply tying this to a push button and 10 K Ω resistor to the 1.8 V power rail.

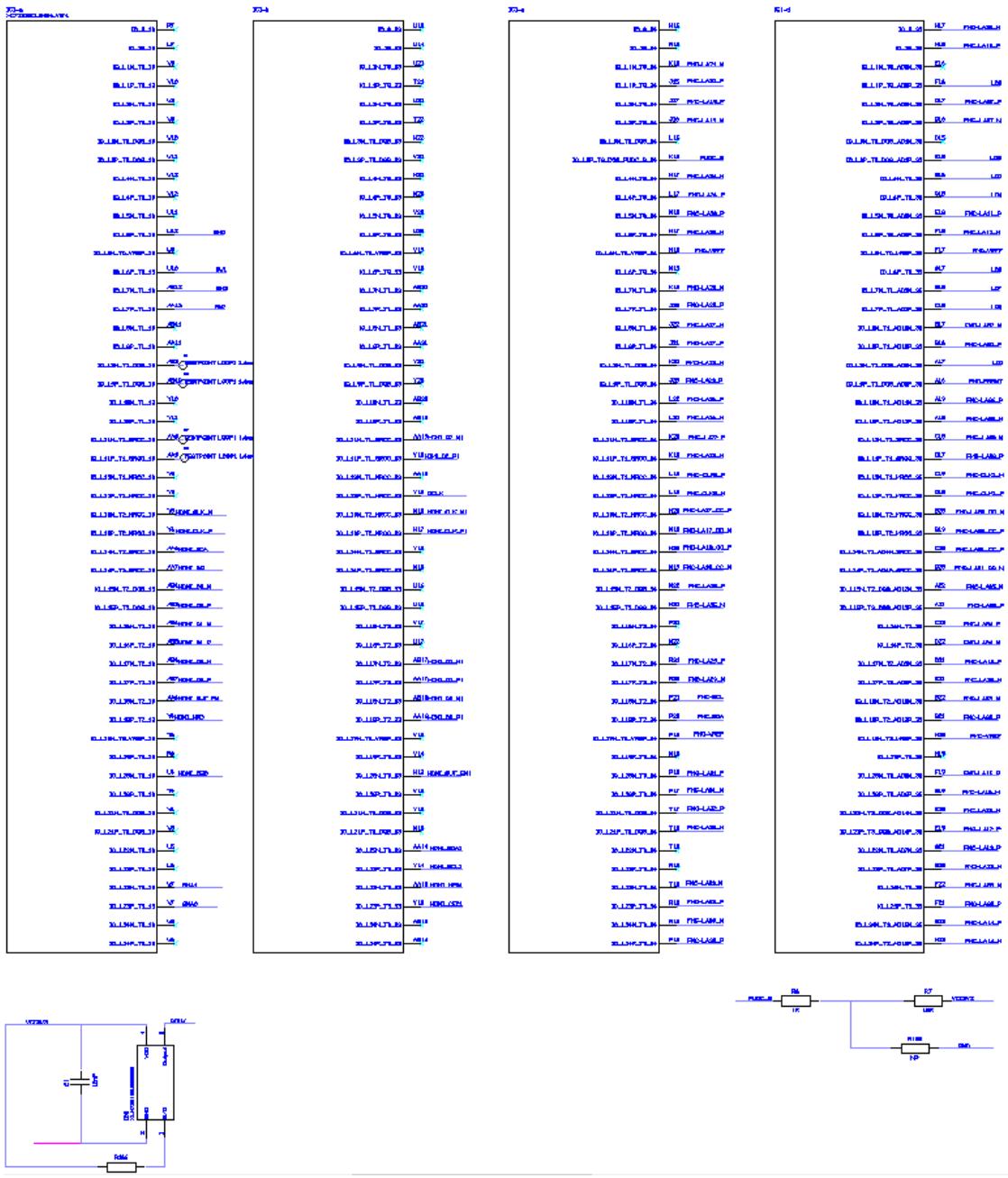


Figure A.1 FPGA I/O Banks

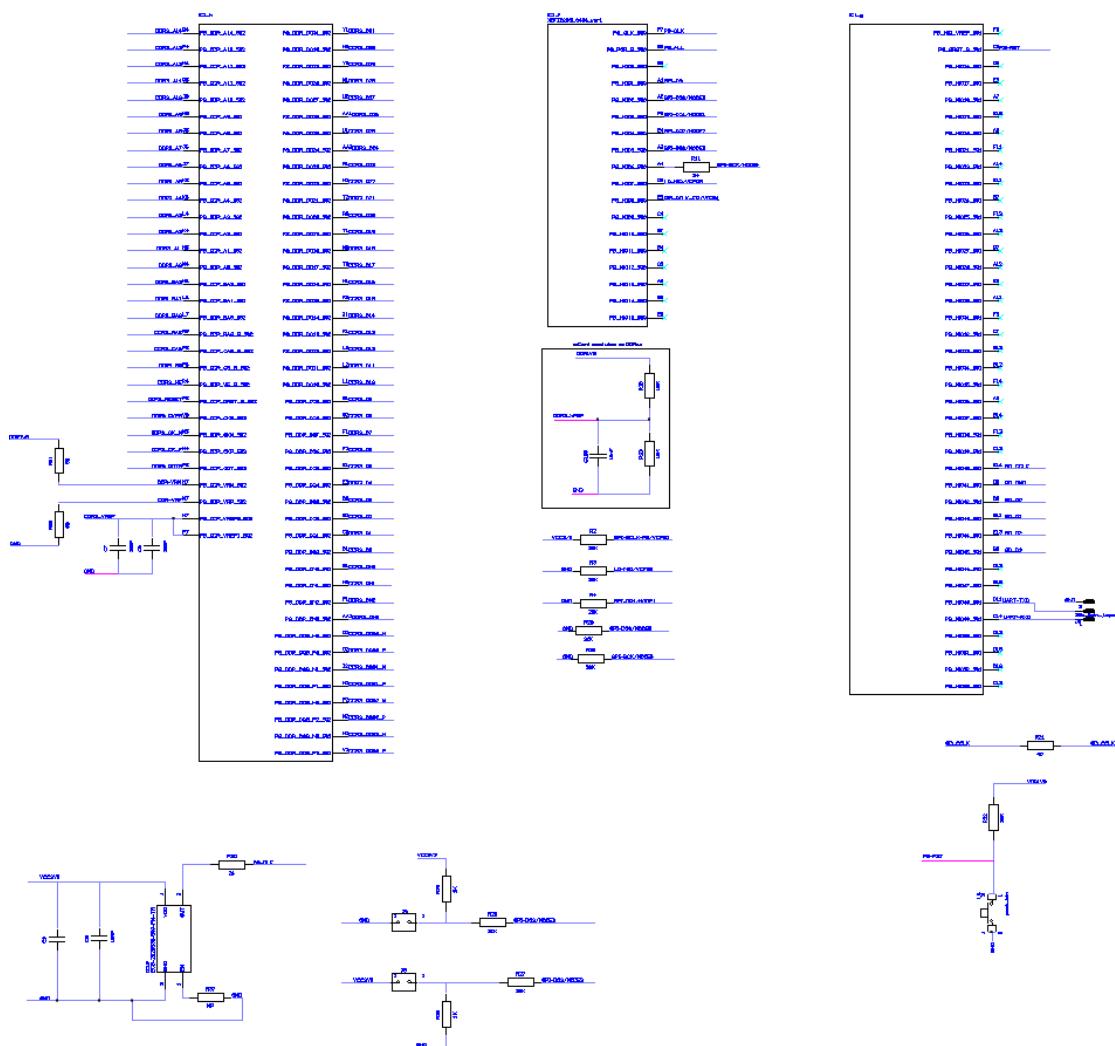


Figure A.2 FPGA DDR and MIO Banks

The FPGA power schematic is shown in Figure A.3. Each of the voltage rails connected to one of the FPGA banks is shown in this schematic. The power nets are named VCC3V3 (3.3V rail), VADJ_2.5/3.3 (adjustable 2.5 V or 3.3 V), VCC1V8 (1.8V), DDR1V5 (1.5 V for SDRAM), VCC1V0 (1.0 V). The associated bypass capacitors are also shown, and the ground connections are seen on the bottom left.

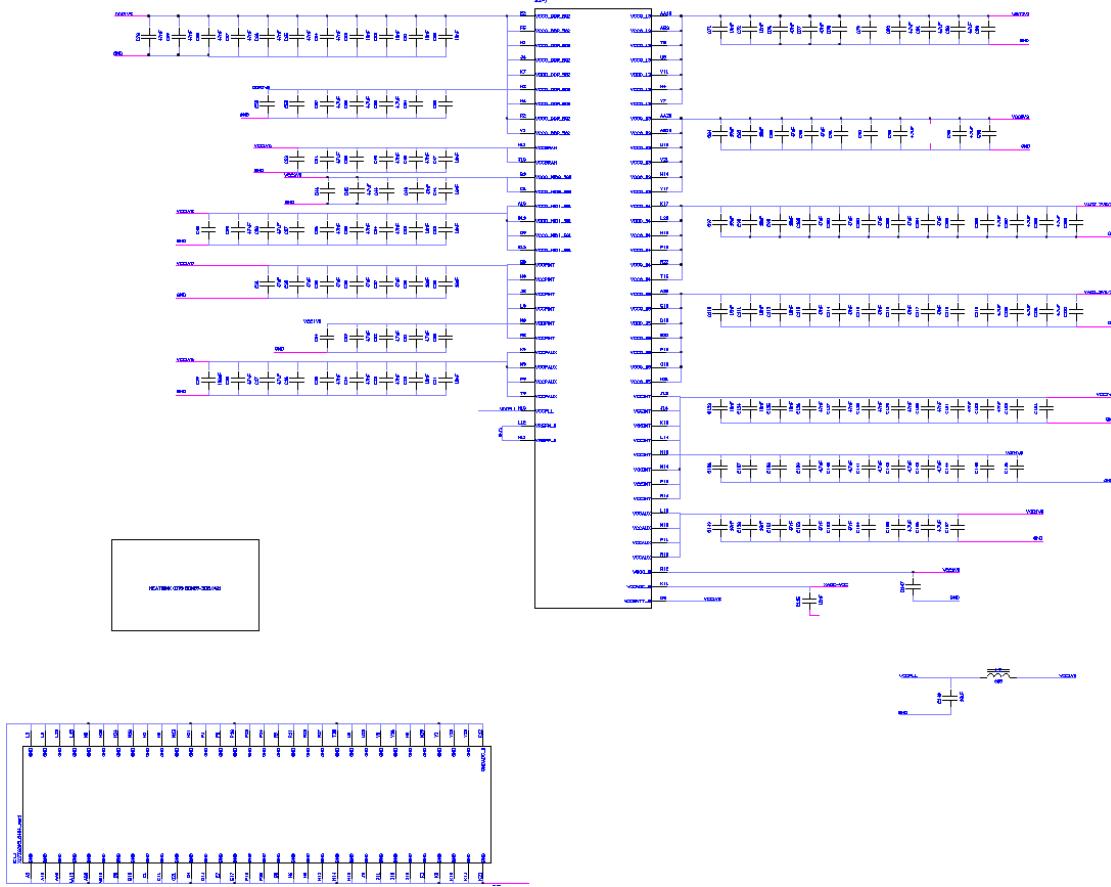


Figure A.3 FPGA Power

The low pin count (LPC) FPGA Mezzanine Card (FMC) connector schematic can be seen in Figure A.4. The majority of these signals are outputs from the FPGA I/O banks to the FMC connector, which sends these signals to the TB-6V-LS760-LXI board in the NESSIE CSE. Power is delivered from the CSE to the ZYNQ-FMCL-HDMI board through the 12V_FMC_in, 3V3_FMC_in, and VADJ_FMC_in lines. Also depicted on this schematic page are the general purpose I/O (GPIO) LED and switch circuits.

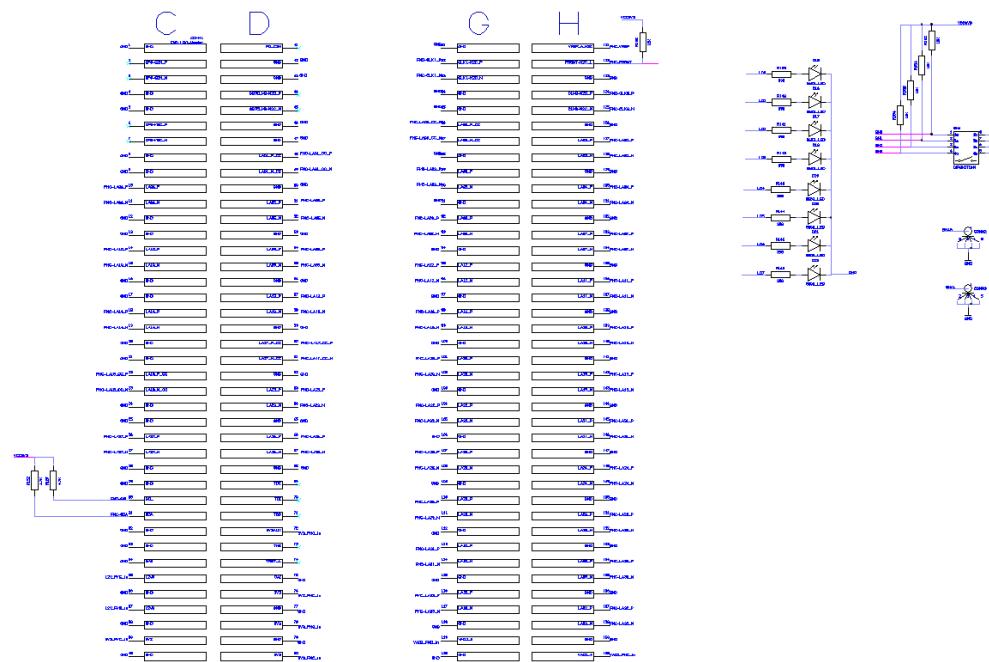


Figure A.4 FMC Connector and General User I/O Schematic

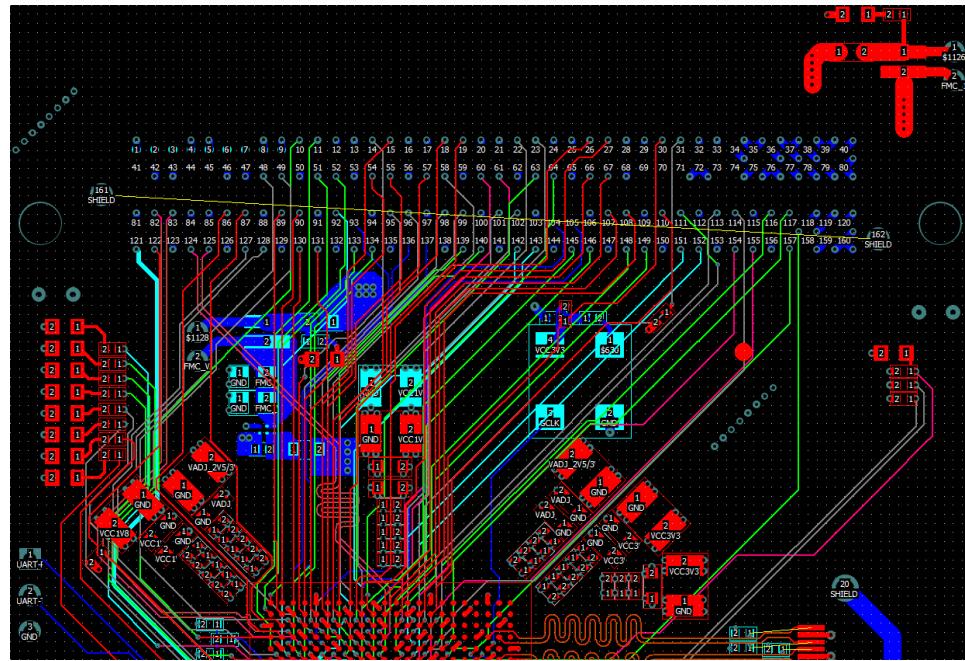


Figure A.5 FMC Connector PCB Layout

Figure A.5 above portrays the PCB layout for the LPC FMC connector. It should be noted that although FMC data signals are typically differential, for this board these signals are treated as single-ended. This was done to maximize the number of signals that could be delivered and because the differential functionality was not necessary for sending data to the CSE. The Inrevium TB-FMCL-HDMI board uses a similar signaling scheme to its FMC connector. [66]

The full schematic and PCB layout are shown in Figures A.6 and A.7.

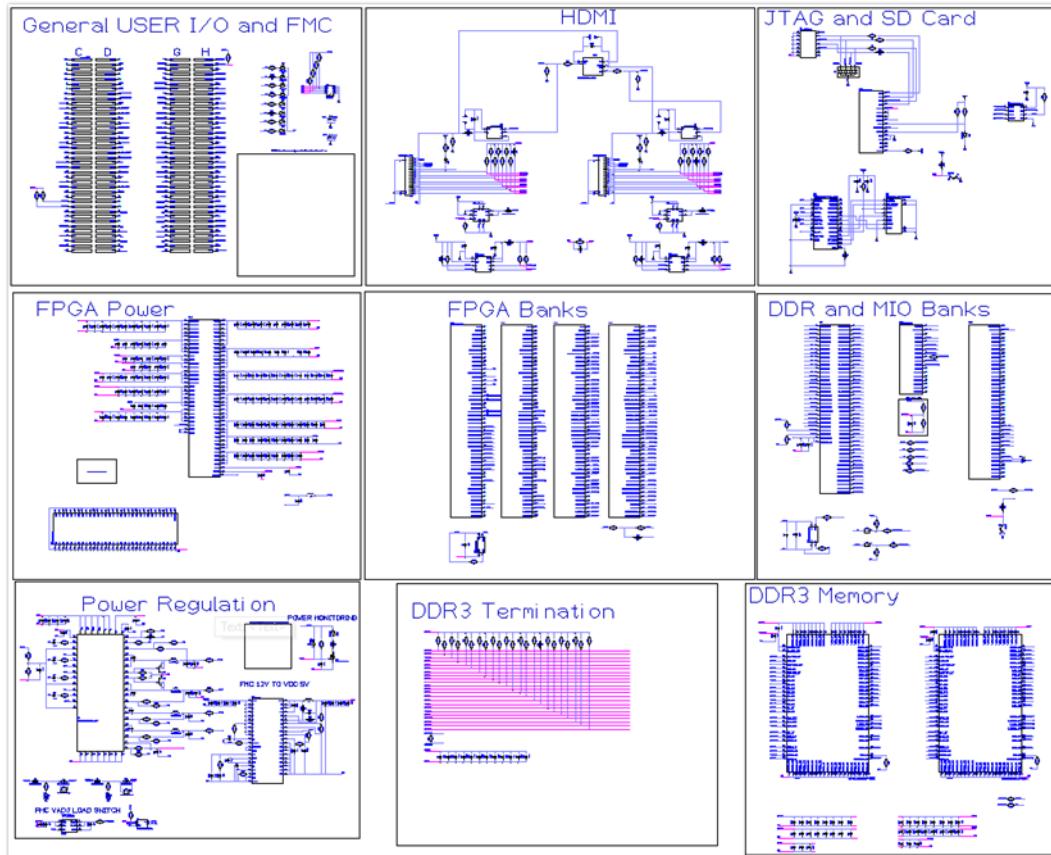


Figure A.6 ZYNQ-FMCL-HDMI Full Schematic

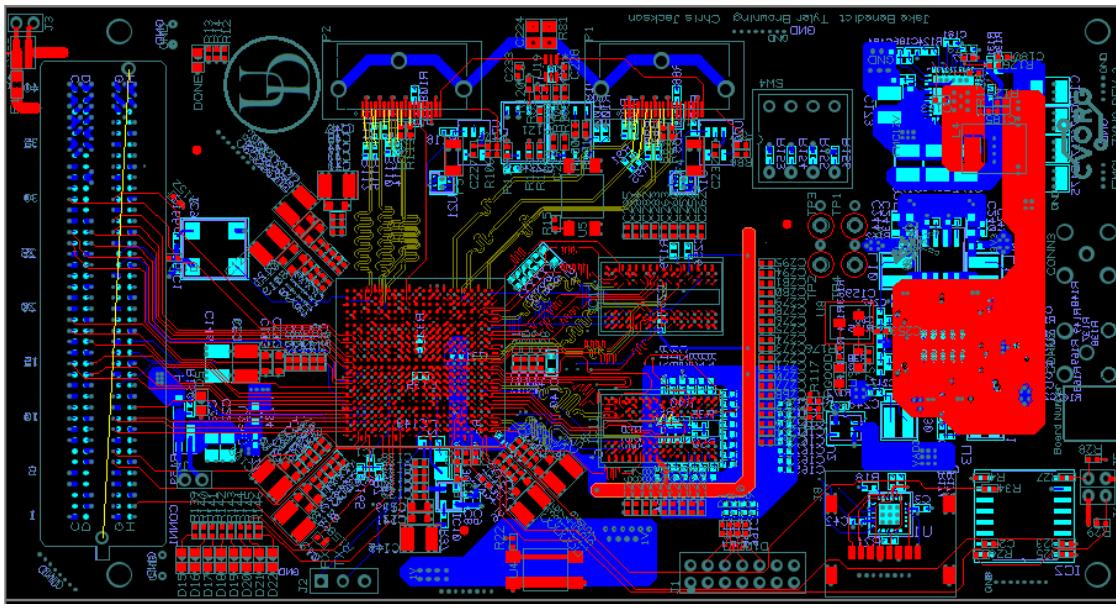


Figure A.7 ZYNQ-FMCL-HDMI Full PCB Layout

Appendix B

SUPPLEMENTARY ZYNQ-FMCL-HDMI TEST RESULTS

Supplementary material for Section 4.3 is presented below.

In Chapter 4, a series of firmware test programs was introduced to initially evaluate the ZYNQ-FMCL-HDMI boards. Variations of these firmware projects were used to test all three revisions of this board, but the results for the most recent batch of revision 3 boards is presented below in Table B.1 and B.2. The left column (starting from the second row) gives the name of the test or firmware test project. For example, “Short test” is the result of electrical connectivity tests to ensure no nets were unexpectedly shorted together. On the other hand, “Flash” is a test of the firmware project to confirm proper operation and programming using the on-board SPI flash chip. The top row is the board number, from 1 to 24.

Table B.1 ZYNQ-FMCL-HDMI Testing Checklist, Boards 1-12

Board #	1	2	3	4	5	6	7	8	9	10	11	12
Short Check	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Power Good	✓	✗	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓
LEDSW	✓		✓	✓	✓	✓		✓	✓	✓	✓	✓
DDR	✓		✓	✓	✓	✓		✓	✓	✓	✓	✓
PS	✓		✓	✓	✓	✓		✓	✓	✓	✓	✓
UART	✓		✓	✓	✓	✓		✓	✓	✓	✓	✓
HDMI I/O	✓		✓	✓	✓	✓		✓	✓	✓	✓	✓
Flash	✓		✓	✓	✓	✓		✓	✓	✓	✓	✓
FMC	✓		✓	✓	✓	✓		✓	✓	✓	✓	✓

All of the boards passed the connectivity tests, indicating that there were no short circuits and there was proper connectivity on each of the power rails and ground planes. Boards 2 and 7 failed the Power Good test, indicating the power on sequence was unsuccessful. The remaining boards passed all of the firmware test projects. Boards 10 and 17 – indicated by the yellow column – developed issues during

operation after this initial round of testing, showing intermittent drops in vsync and hsync data. These were removed from the rotation pending further testing.

Table B.2 ZYNQ-FMCL-HDMI Testing Checklist, Boards 13-24

Board #	13	14	15	16	17	18	19	20	21	22	23	24
Short Check	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Power Good	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
LEDSW	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
DDR	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
PS	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
UART	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
HDMI I/O	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Flash	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
FMC	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Section 4.3 described a test process in which the standalone TB-6V-LX760-LSI test platform was incrementally modified by adding DACs, amplifiers, and interface boards until it matched the configuration of a full CSE. Figures B.1–B.17 are oscilloscope captures of each of these configurations. In each figure, the yellow signal is the signal of interest Hot Plug Detect (HPD), and the others are voltage rails 3.3 V, 1.8 V, and 1.5 V (orange, green, and blue).

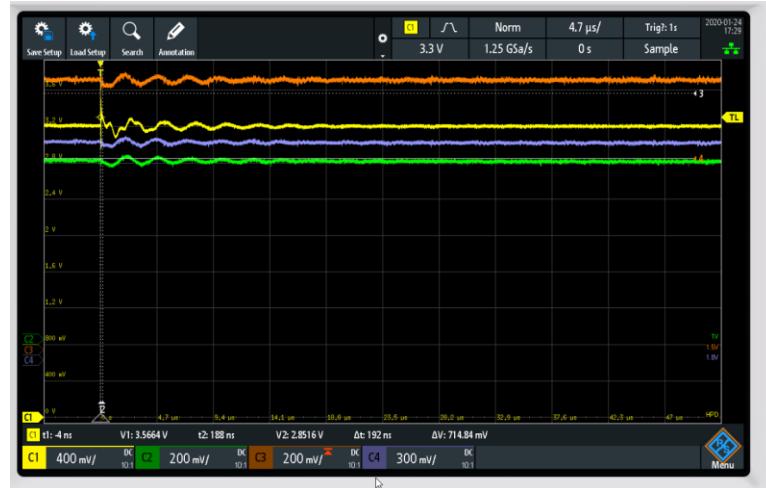


Figure B.1 First DAC, connected to FMC slot 7.

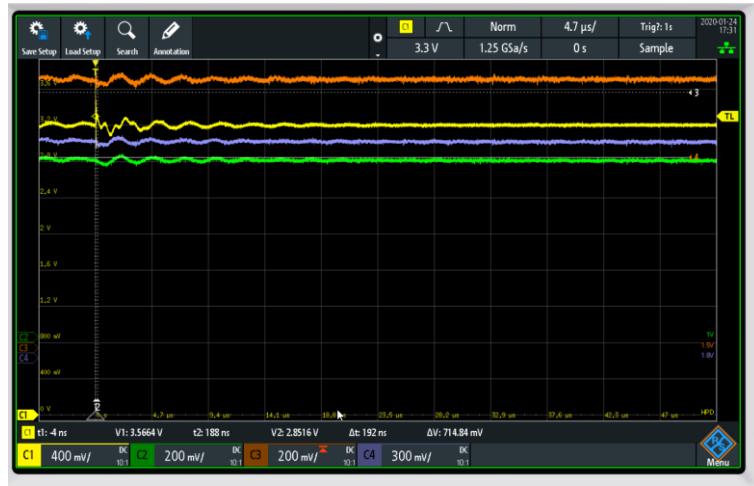


Figure B.2 Second DAC, connected to FMC slot 2.



Figure B.3 Third DAC, connected to FMC slot 8.

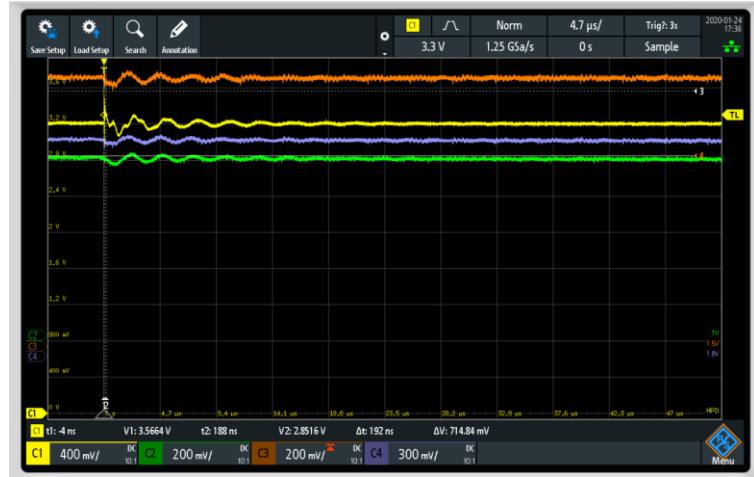


Figure B.4 Fourth DAC, connected to FMC slot 3.

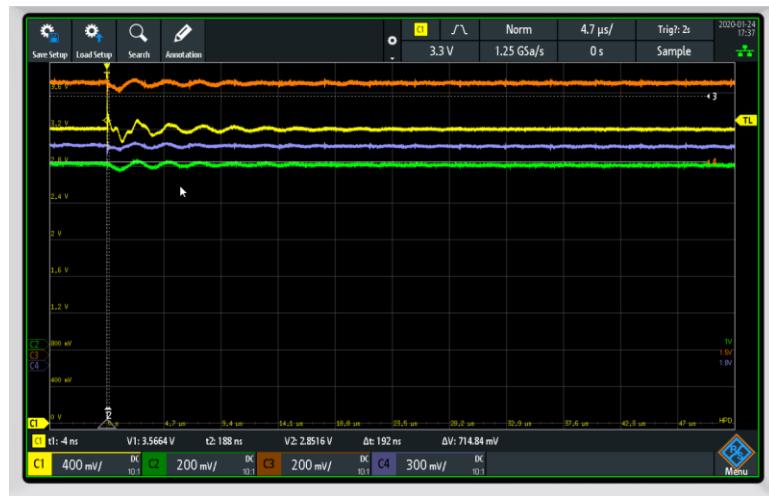


Figure B.5 Fifth DAC, connected to FMC slot 9.



Figure B.6 Sixth DAC, connected to FMC slot 4.



Figure B.7 Seventh DAC, connected to FMC slot 10.



Figure B.8 Eighth DAC, connected to FMC slot 5.



Figure B.9 First amplifier on FMC slot 7, 8 DACs.



Figure B.10 Second amplifier on FMC slot 2, 8 DACs.



Figure B.11 Third amplifier on FMC slot 8, 8 DACs.



Figure B.12 Fourth amplifier on FMC slot 3, 8 DACs.



Figure B.13 Fifth amplifier on FMC slot 9, 8 DACs.



Figure B.14 Sixth amplifier on FMC slot 4, 8 DACs.



Figure B.15 Seventh amplifier on FMC slot 10, 8 DACs.

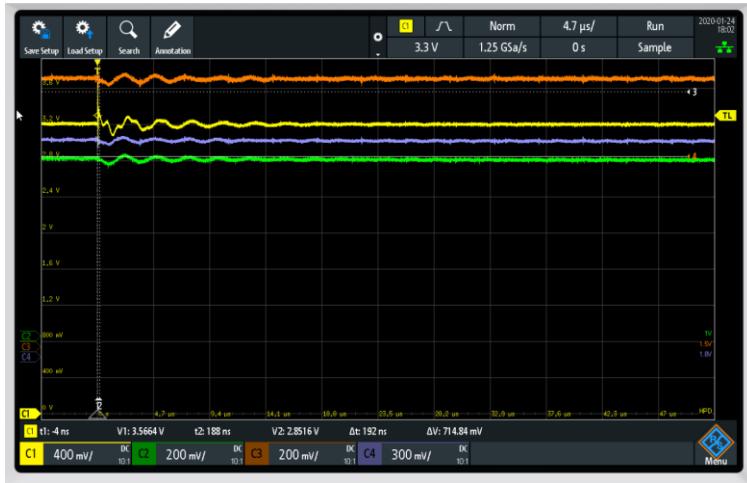


Figure B.16 Eighth amplifier on FMC slot 5, 8 DACs.



Figure B.17 Master and Slave Interface Boards with all 8 DACs and amplifiers.

There is signal disruption, but the boards still demonstrated correct behavior.

Appendix C

LIST OF TERMS

Table C.1 List of Terms

Term	Explanation
AC	Alternating Current
ADC	Analog-to-Digital Converter
AFB	Air Force Base
AFRL	Air Force Research Laboratory
AIREA	Advanced-Infra-Red Emitter Array
AMBA	Advanced Microcontroller Bus Architecture
AMD	Advanced Micro Devices
APG	Aberdeen Proving Ground
API	Application Programming Interface
ARL	Army Research Laboratory
ART-IDEA	Advanced RIIC Technologies for Increasing Density of Emitter Arrays
ASIC	Application-Specific Integrated Circuit
AvMC	Aviation and Missile Center
AWG	American Wire Gauge
AXI	Advanced eXtensible Interface
BERT	Bit Error Rate Tester
BGA	Ball Grid Array
BRAM	Block Random Access Memory
BSP	Board Support Package
CDS	Chip Design Systems
CMOS	Complementary Metal-Oxide Semiconductor
COTS	Commercial off-the-Shelf
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CRT	Cathode-Ray Tube
CSE	Close Support Electronics
CTEIP	Central Test and Evaluation Investment Program
CuW	Copper-Tungsten (Cuprium Wolfram)
DAC	Digital-to-Analog Converter

DC	Direct Current
DDR	Double Data Rate
DMA	Direct Memory Access
DRC	Design Rule Check
DVI	Digital Visual Interface
EAGLE	Efficient Arrays for Generating Light Emission
EDT	Engineering Design Team
EEPROM	Electrically Erasable Programmable Read-Only Memory
EIRE	Efficient InfraRed Emitters
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
ESD	Electrostatic Discharge
FMC	FPGA Mezzanine Connector
FPGA	Field-Programmable Gate Array
FPS	Frames per Second
GaSb	Gallium-Antimonide (Stibium)
GND	Ground
GPIO	General Purpose Input/Output
GPU	Graphics Processing Unit
GUI	Graphical User Interface
GWEF	Guided Weapons Evaluation Facility
HDILED	High-Definition Infrared Light-Emitting Diode
HDL	Hardware Description Language
HDMI	High-Definition Multimedia Interface
HPC	High Pin Count
HPD	Hot Plug Detect
HSLEDS	High Fidelity Superlattice Light Emitting Diode System
Hsync	Horizontal Sync
HWIL	Hardware-In-The-Loop
I2C	Inter-Integrated Circuit, also written as I ² C or IIC
IC	Integrated Circuit
IDE	Integrated Drive Electronics
I/F	Interface
ILA	Initial Lane Assignment (JESD204B)
ILA	Integrated Logic Analyzer (Xilinx)
InSb	Indium Antimonide
I/O	Input/Output
IP	Intellectual Property
IR	Infrared
IRLED	Infrared Light Emitting Diode
IRSP	Infrared Scene Projector

ISLEDS	Innovative Superlattice Light Emitting Diode System
JEDEC	Joint Electron Device Engineering Council
JESD204B	JEDEC Standard 204B
JTAG	Joint Test Action Group
LCD	Liquid Crystal Display
LED	Light Emitting Diode
LMFC	Local Multi-Frame Clocks
LPC	Low Pin Count
LSB	Least Significant Bit
LVCMOS	Low-Voltage Complementary Metal-Oxide Semiconductor
LVDS	Low-Voltage Differential Signaling
LRF	Lucky Region Fusion
MIMO	Multiple-Input Multiple-Output
MOUT	Monitor-out
MSB	Most Significant Bit
MWIR	Mid-Wave Infrared
NESSIE	Nimble, Extensible, and Scalable SLEDs Infrastructure Electronics
NMOS	N-channel Metal Oxide Semiconductor
NSLEDS	N^3 Superlattice Light Emitting Diode System
NUC	Non-Uniformity Correction
NUCPC	Non-Uniformity Correction Personal Computer
NUD	Non-Uniformity Detection
OnSemi	OnSemiconductor
PCB	Printed Circuit Board
PCI-E	Peripheral Component Interconnect Express
PDP	Packetized Display Protocol
PING	Projector IRLED Native GUI
PL	Programable Logic
PMOD	Peripheral MODule
PMOS	P-channel Metal Oxide Semiconductor
PRBS	Pseudorandom Binary Sequence
PS	Processing System
RAM	Random Access Memory
RF	Radio Frequency
RFSoC	Radio Frequency System-on-Chip
RIIC	Read-In Integrated Circuit
ROI	Region of Interest
ROIC	Read-Out Integrated Circuit
RS-232	Recommended Standard 232
SBIR	Santa Barbara Infrared
SBIR	Small Business Innovation Research

SCB	Synchronized Circular Buffer
SD	Secure Digital
SDK	Software Development Kit
SDR	Single Data Rate
SDRAM	Synchronous Dynamic Random Access Memory
SFP	Small Form-factor Pluggable
SLED	Superlattice Light Emitting Diode
SLEDS	Superlattice Light Emitting Diode System
SMA	SubMiniature version A
SoC	System-on-Chip
SO-DIMM	Small Outline Dual In-Line Memory Module
SPI	Serial Peripheral Interface
SSD	Solid-State Drive
SWAP	Size, Weight and Power
SWIR	Short Wave Infrared
TCSA	Two Color SLEDs Array
TMDS	Transition-Minimized Differential Signaling
UART	Universal Asynchronous Receiver-Transmitter
UDel	University of Delaware
UIowa	University of Iowa
USB	Universal Serial Bus
UUT	Unit Under Test
VGA	Video Graphics Array
VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuit
VLSI	Very Large Scale Integration
Vsync	Vertical Sync
ZnSe	Zync Selenide
ZYBO	ZYnq BOard