

# **Design Specifications**

**ULTRA-FAST CMOS FAILURE TEST UNIT**

**Ivanov Group at UBC SoC Lab**

**Capstone Group 12**

**Jerry Liu**

**Jiaqi Yue**

**Ketan Desai**

**Abdullah Mubushar**

**Keerthana Aparacherla**

## Table of Contents

List of Figures.....	3
List of Abbreviations.....	4
Background and Context.....	6
Design.....	7
1.0 System Architecture.....	7
2.0 Design Choices.....	11
2.1 Controller.....	11
2.2 Analog to Digital Converter.....	13
2.3 Analog Front End.....	15
2.3.1 Signal Path.....	15
2.3.2 Power Supplies.....	17
2.3.3 AFE Board.....	17
2.4 Clock Domain Buffering System.....	20
2.5 Frequency Counter.....	22
2.6 Double Data Rate Buffer.....	23
2.7 Solid State Drive Storage.....	26
2.8 Environment Generation.....	27
2.8.1 Microcontroller.....	27
2.8.2 Heater Driver and Heater Click.....	27
2.8.3 Helmholtz Driver and Coil.....	27
2.8.4 Voltage Stress Circuit.....	27
2.9 Digital Board.....	28
2.9.1 Layer Stackup.....	28

2.9.2 Controlled Impedance.....	29
2.9.3 Length Matching.....	29
2.9.4 Power Supplies.....	30
2.10 State Machine.....	31
2.10.1 Initializing ADCs, DDR3 Controller, FIFO & SD Controller.....	31
2.10.2 Responding to User input to start or stop a test.....	32
2.10.3 Controlling the existing board to start testing the DUT.....	32
2.10.4 Displaying current Status through LEDs.....	32
2.10.5 Directing data flow from ADC to FIFO.....	32
2.10.6 Directing data flow from FIFO to DDR3 Controller.....	32
2.10.7 Storing data into SD Card at the end of a failure testing cycle.....	32
2.10.8 Resetting and preparing for next failure testing cycle.....	33
2.11 Visualization and Data Processing.....	34
3.0 Data Flow.....	35
References.....	37

## List of Figures

Figure 1: High Level System Architecture.....	7
Figure 2: Detailed System Architecture.....	8
Figure 3: AFE Board Architecture.....	9
Figure 4: Digital Board Architecture.....	9
Figure 5: Full System Render.....	10
Figure 6: Controller (Lattice ECP5-85F).....	12
Figure 7: Analog Front End Board.....	18
Figure 8: AFE Board Stackup.....	18
Figure 9: AFE Length Matched Trace Example.....	19
Figure 10: State-Transition Diagram.....	20
Figure 11: Frequency Counter Sample Waveform.....	22
Figure 12: DDR3 Memory (IS43/46TR16512A).....	23
Figure 13: Lattice Diamond DDR3 Controller (JESD79-3).....	24
Figure 14: Digital Board.....	28
Figure 15: Digital Board Layer Stack.....	29
Figure 16: Length Matched Trace Example.....	30
Figure 17: Digital Board Power Supplies.....	30
Figure 18: Visualizer Wave Replay Feature (full triangle sample wave).....	34
Figure 19: Data Flow.....	35

## List of Abbreviations

ADC	- Analog to Digital Converter
AFE	- Analog Front End
DUT	- Device Under Test
DDR	- Double Data Rate
EMI	- Electromagnetic interference
CPU	- Central Processing Unit
FDA	- Fully Differential Amplifier
FIFO	- First-In, First-Out
FPGA	- Field Programmable Gate Array
FTU	- Failure Test Unit
GSPS	- Giga Samples Per Second
IC	- Integrated Circuit
IO	- Input Output
LDO	- Low Drop-Out
LUT	- Look Up Tables (Logic Elements)
LVCMOS	- Low Voltage Complementary Metal Oxide Semiconductor
LVDS	- Low Voltage Differential Signalling
MSPS	- Mega Samples Per Second
MT/s	- Mega Transfers Per Second
PID	- Proportional Integral Derivative
PGA	- Programmable Gain Amplifier
PSRR	- Power Supply Rejection Ratio
SATA	- Serial Advanced Technology Attachment
SoC	- System On a Chip
SPI	- Serial Peripheral Interface
SPICE	- Simulation Program with Integrated Circuit Emphasis
SSD	- Solid State Drive



## **Background and Context**

The Ivanov Group at UBC's SoC lab is performing lifetime testing of transistors and researching ways to improve the lifetime of these transistors. This research involves understanding the fundamental causes of device aging and possible methods to reverse some of the aging effects. Due to the speed at which these processes occur, high performance test equipment is needed.

The current research methods involve the use of expensive oscilloscopes and spectrum analyzers to perform measurements on transistors. While these systems can provide some data for researchers, they are not optimized for these specific tests. These tools require a lot of user interaction to setup, as well as a large amount of resources to use. An Ultra-Fast Failure Test Unit (UF-FTU) is designed to significantly reduce time and resource usage when performing transistor lifetime testing, while remaining more cost effective than commercial test equipment.

The overarching goal is to design a system capable of performing three main tasks:

- Generating a test environment for the DUT
- Performing measurements on the outputs of the DUT
- Displaying the results of this testing

The design document aims to serve as technical documentation to provide insight into the design choices and justifications for these choices. The document is intended for use by the client, Ivanov Group at UBC SoC group. In this document we will discuss the high level system architecture and the sub modules that make up this overall system.

## Design

### 1.0 System Architecture

Based on the existing system, and team experience in designing sampling systems for low cost oscilloscopes, a high level system architecture has been developed.

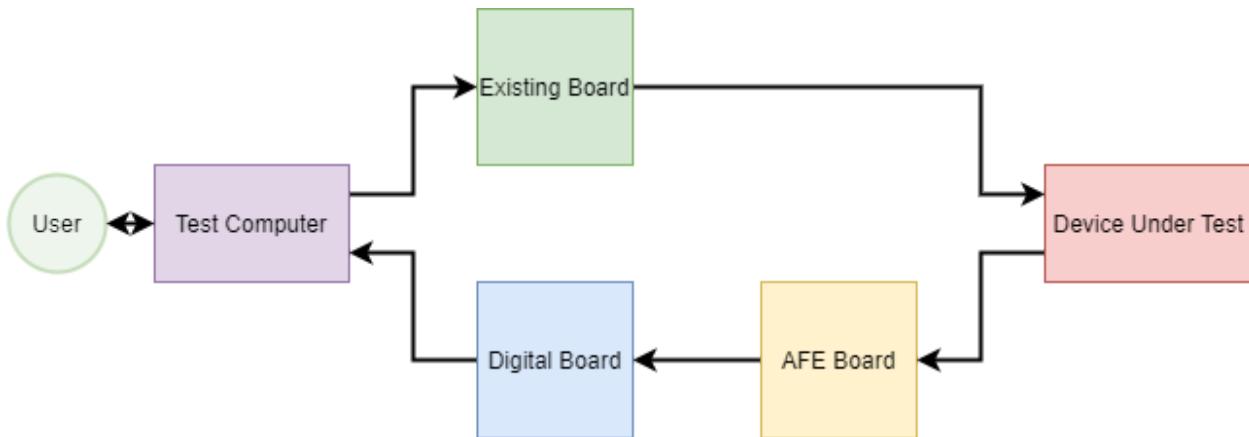


Figure 1: High Level System Architecture

The process starts as the user configures a test to be performed on the device under test. This information is sent to an existing system (Shown by the green box in Figure 1). This existing system generates the desired test environment and drives the Device Under Test (Shown by the red box in Figure 1) with a 50MHz test waveform. The goal is to test the performance of the DUT in a variety of scenarios to characterize performance versus environmental factors. This DUT consists of a CMOS buffer, therefore we can compare the reference waveform to the output of the DUT to determine the performance.

The signal is passed from the DUT to the AFE Board (Shown by the yellow box in Figure 1). The AFE performs scaling and filtering of the signal to remove noise. The signal is then passed to the Digital Board (Shown by the blue box in Figure 1). The digital board samples the signal, buffers it in high speed memory, and finally returns the data to the test computer.

The scope of this project is to develop the Digital Board and the AFE Board, ensuring that they are compatible with the existing system. Based on the high level architecture, a more detailed system visual is shown in Figure 2.

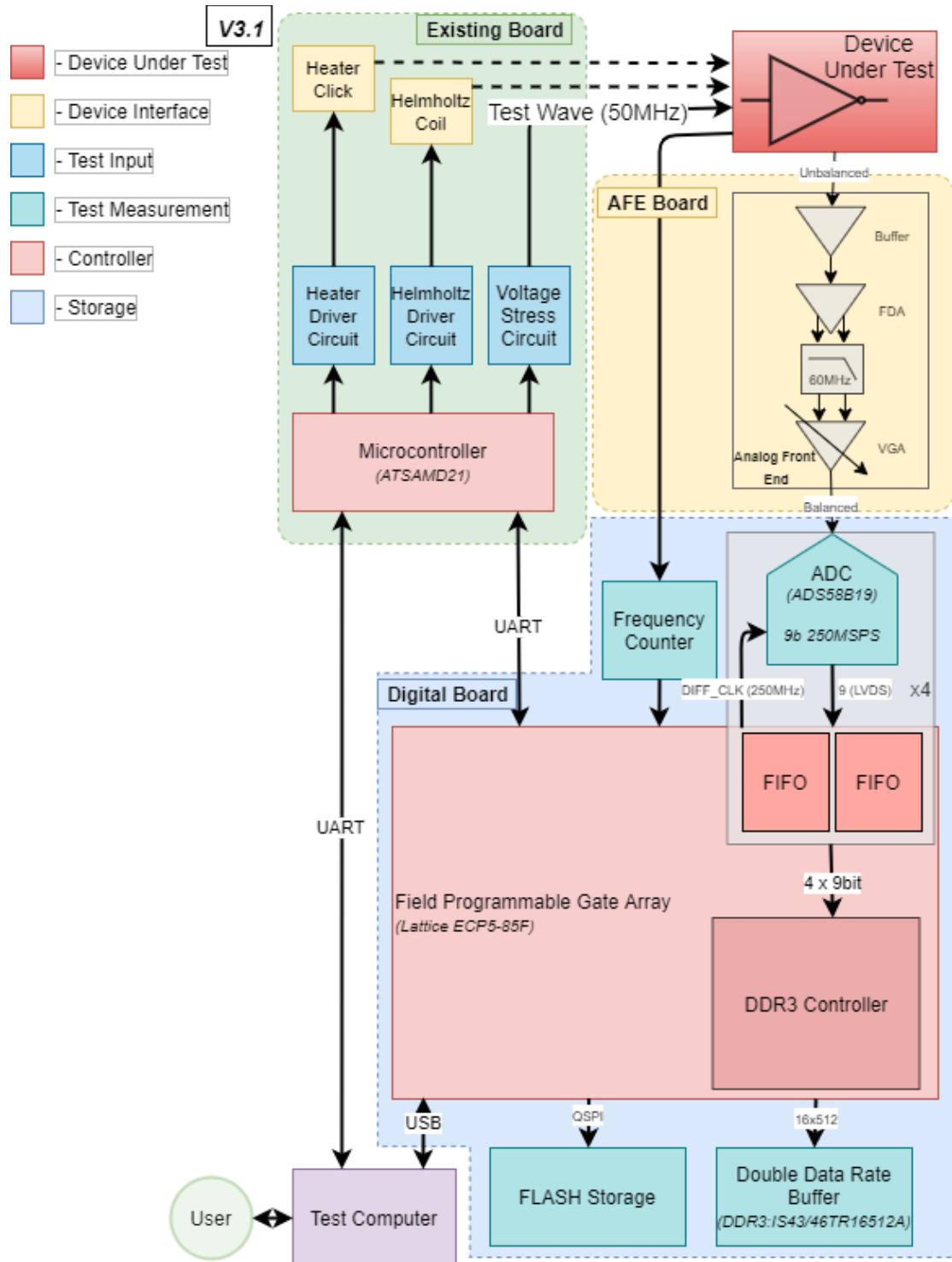


Figure 2: Detailed System Architecture

To enable modularity in the design, the analog front end and digital board have been separated into two separate PCBs. This allows the client to replace the front end to allow the digital board

to be used in multiple scenarios. Figures 3 and 4 show the specific implementations of the AFE Board and the Digital Board.

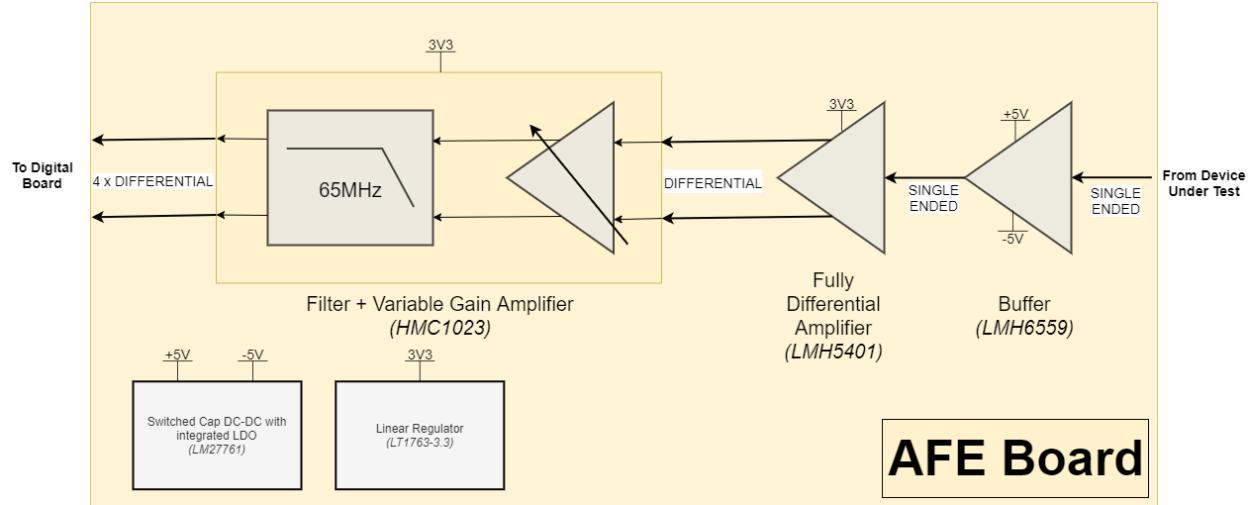


Figure 3: AFE Board Architecture

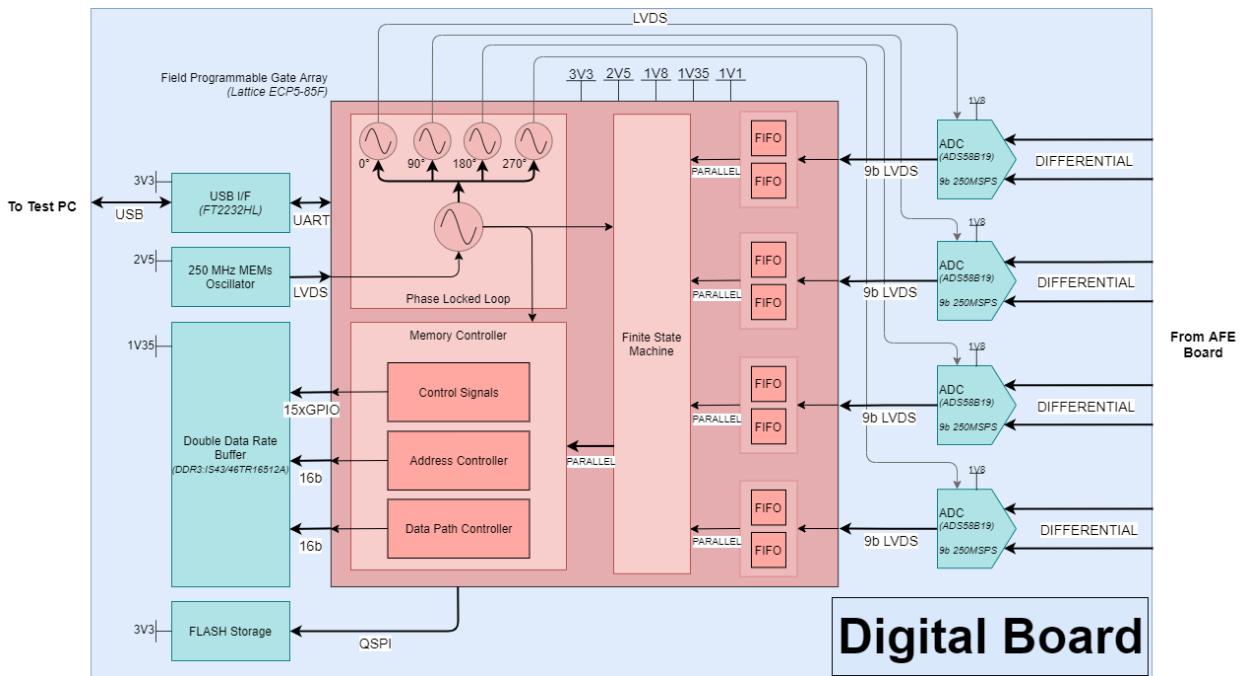
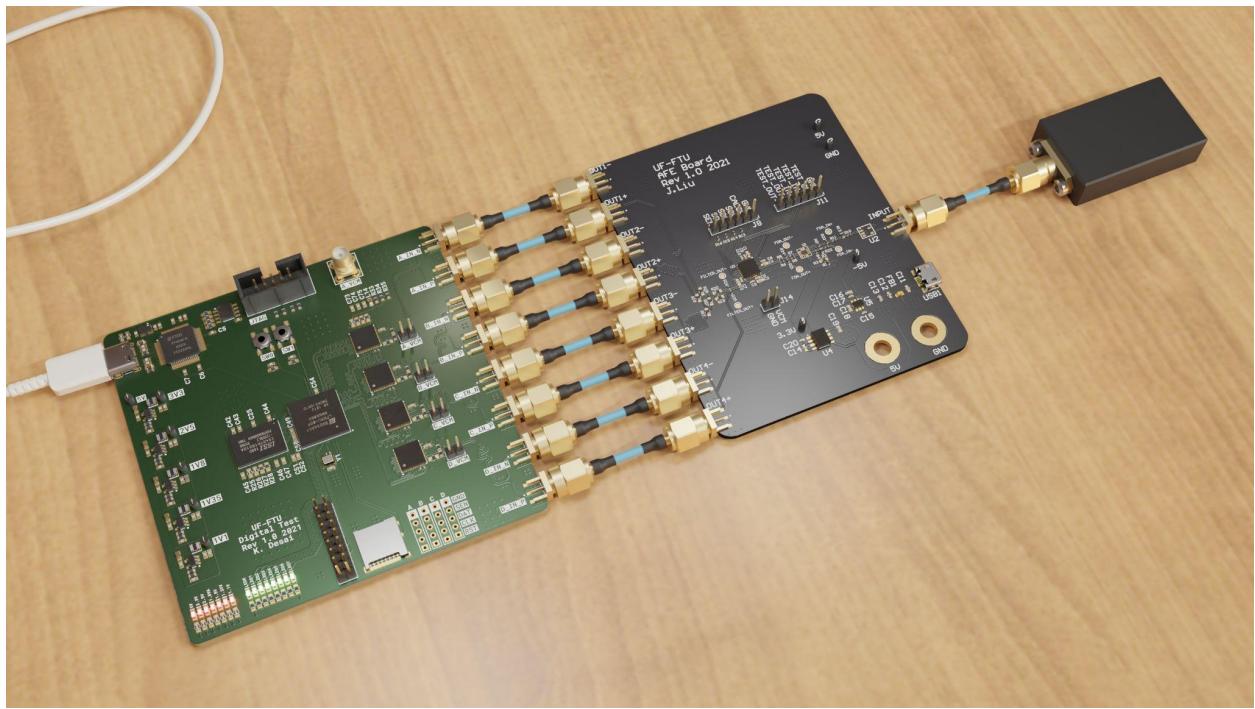


Figure 4: Digital Board Architecture

A render of the system is shown in Figure 5, with the DUT shown on the right of the image, the AFE board in the centre, and the digital board on the left.



*Figure 5: Full System Render*

## **2.0 Design Choices**

Due to the high data rates associated with this project, component selection is extremely important. The design choices are based around the project requirements, key of which is the 1GSPS sampling rate and the 10mV resolution.

Our team chose to split the design into these blocks:

- Analog Front End
- Digital Hardware
- FPGA Gateware and Buffering
- Storage

This design approach allows a high level of parallelism, while also minimizing on integration steps. This parallel design approach also allows for easier remote collaboration and reduces potential delays that could be caused by issues in other sections. Specific design choices are outlined below.

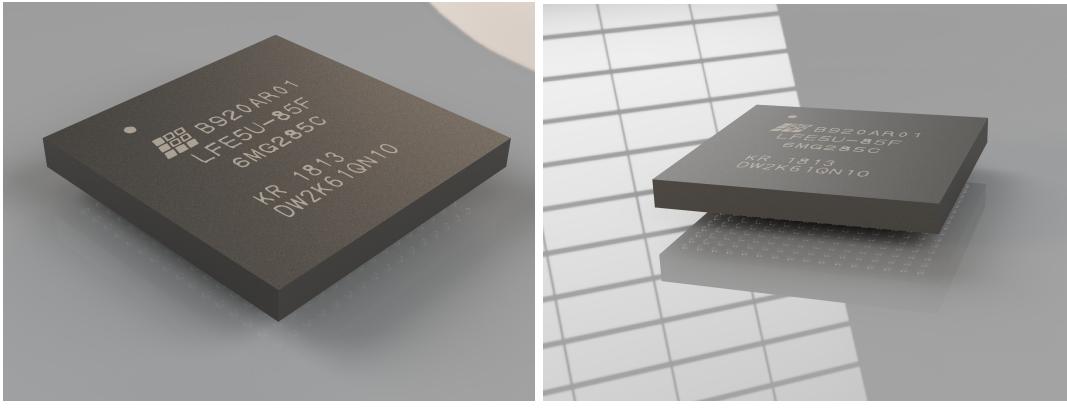
### **2.1 Controller**

To conform to the 1GSPS sample requirement, a controller capable of handling data at effectively 1GHz is required. While CPUs with clock rates above 1GHz exist, there are no publicly available devices capable of receiving parallel data busses at this rate.

A device with configurable logic allows for the data path to be optimized for high speed transfers. An FPGA provides this configurable logic, and allows a host of peripherals to be attached, fitting the other criteria of the project.

To meet the storage requirements of the project, we determined that DDR3 memory will be needed. Therefore the FPGA selected must support this memory.

To reduce costs, we also selected to interleave 4 ADCs. This means that the FPGA must be capable of generating four low skew clock references for the ADC, to minimize interleaving spurs during sampling. Therefore the FPGA select must have an internal PLL capable of introducing 90 degree phase shifts. The FPGA must also be able to interface with ADCs in the 250MSPS class, which take advantage of LVDS to transfer high speed data.



*Figure 6: Controller (Lattice ECP5-85F)*

Based on these criteria, the FPGA selected is the Lattice ECP5-85F [1]. This FPGA has been selected due to its high performance features and low cost. Some of these features include:

- Native DDR3 support with up to 800MT/s and on chip termination
- Four PLLs with advanced clock routing functionality
- 85K LUTs
- LVDS Interface
- Wide support in open source FPGA cores
- 285-pin BGA package

These features allow this FPGA to support the high data rates required for this project and allow for expandability moving forward. While higher cost alternatives from manufacturers like Xilinx were considered, the price to performance of this FPGA is unmatched.

The four PLLs are driven by an external 250MHz oscillator, the Raltron CL3225-250.000 with a 25ppm stability. The biggest factor is that all four clocks must be referenced off the same clock. This ensures that the ADCs are interleaved correctly, minimizing spurs. With a Nyquist frequency of 100MHz, our 250MHz clock provides ample overhead for minor issues with jitter. The clocks are connected to the dedicated clock routing pins of the FPGA, allowing minimal skew, and reduced jitter.

## 2.2 Analog to Digital Converter

The ADC choice is mainly based on the sample rate and resolution requirements. 1GSPS ADCs are available, however the use of these ADCs requires higher cost FPGAs that support SERDES functionality to use the JESD204b serial communication protocol.

To avoid this cost, we have chosen to use an interleaved sampling architecture. This architecture allows multiple lower speed ADCs to work together to achieve higher effective data sampling rates. By using four 250MSPS ADCs interleaved together, we can reach the 1GSPS sampling rate requirement, while also providing a platform for the client to perform varying numbers of tests simultaneously. The overall cost of using 4 low speed ADCs as opposed to one high speed ADC also provides incentive to use the interleaved method. This means that the ADCs used must have a 250MSPS sample rate.

As the measurement signal is a 2.5V LVCMOS signal, we need to scale this down to meet the  $1.5V_{\text{peak-peak}}$  input of the ADC. We can determine that for a 9-bit ADC we have:

$$\frac{2.5V}{2^9} = 4.88mV \text{ resolution}$$

This meets our requirements while also providing some extra headroom for the reduced effective number of bits associated with high frequency ADCs. This means that an ADC with 9 or more bits of resolution is required.

The ADC must also be able to output to a LVDS parallel bus so that low cost FPGAs may be used.

Based on these requirements, the Texas Instruments ADS58B19 [2] has been selected for its low cost and excellent performance. Some of its key specifications are:

- 250MSPS Sampling Rate
- 9-bit resolution
- Max analog bandwidth of 400MHz
- LVDS DDR interface operating at 250MHz

These specifications allow us to reach our required sampling rate of 1GSPS with a 10mV input resolution. All other ADCs conforming to these requirements are significantly more

expensive than the selected device.

The data from the ADC is centre aligned to the data clock, so dedicated clock routing resources are not required for this interface. The data from each ADC is clocked into the FPGA using independent clocks. This is done to ensure that variations in temperature across the ADCs will not introduce timing violations.

One potential issue with interleaved sampling is the introduction of spurious content. These spurs are represented as a time-domain fixed pattern of noise. Potential causes of these are phase offsets in sample clocks and variations in DC offset voltages. The design of the Digital Board makes efforts to minimize these spurs, and due to the fixed pattern of noise, software post processing of the data can remove the remainder of the noise.

## 2.3 Analog Front End

### 2.3.1 Signal Path

The analog front end (AFE) module is required to convert the 50MHz unbalanced single-ended output signal from the DUT to balanced differential signals for our high performance ADCs. The use of differential signaling will also reduce common mode noise that could be coupled on to the sensitive analog signals from environmental EMI or switching noise from circuitry elsewhere on the PCB. The AFE module must also scale the output signal from the DUT in order to match our ADCs' input signal dynamic range. Lastly, the AFE module should filter out unwanted high frequency aliasing noise (125MHz), and present a clean and low-distortion signal to the ADC inputs.

The AFE module consists of an input buffer to provide high impedance for the DUT, a fully differential amplifier (FDA) to perform signal scaling and single-ended to differential signaling conversion, and lastly a filtering stage for anti-aliasing filter.

For the input buffer stage, the Texas Instruments LMH6559 [3] is selected for its simplistic design and high frequency performance. Some of its key specifications are:

- 1050MHz large signal bandwidth
- Dual supply voltage of  $\pm 5V$
- 63dB PSRR
- $200k\Omega$  input resistance

For the FDA stage, the Texas Instruments LMH5401 [4] has been selected for its high bandwidth and low distortion performance. Some of its key specifications are:

- 4.4GHz large signal bandwidth
- 2.8V maximum differential output voltage swing
- Adjustable output common-mode
- 86dBc minimum harmonic distortion at < 100MHz
- 14uV input voltage noise at < 125MHz

These specifications allow us to meet all of the AFE design requirements:

- 4.4GHz bandwidth is more than enough for a flat band response for the input signal fundamental frequency (50MHz) and higher frequency components
- 2.8V maximum differential output voltage swing is enough to cover the 1.5V differential voltage swing range of the ADC, with some extra headroom
- Adjustable output common-mode allows us to set the output common-mode voltage to 1.7V, compatible with the ADC
- 86dBc harmonic distortion at < 100MHz is better than the ADC's distortion performance (55dBc), so the FDA won't degrade overall system distortion performance
- 14uV voltage noise at < 125MHz is significantly less than the signal resolution to not affect overall AFE noise performance

The input buffer and FDA related circuitry have been validated through LTSpice simulations, their performance has been verified and meets our design requirements. Further details can be found in the Validation and Verification document.

Lastly for the filtering and gain stage, we've selected an off-the-shelf programmable gain amplifier (PGA) and filter IC for their superior frequency response and robustness. The chip we selected is HMC1023 [5] by Analog Devices. Some of its key specifications are:

- 6th order butterworth low pass filter
- 5MHz to 72MHz programmable cutoff frequency
- 0 to 10dB programmable gain
- One-time-programmable default settings
- SPI interface

By setting the low pass filter's programmable cutoff frequency to 65MHz, we'll be able to have less than 0.3dB attenuation within our signal pass-band (50MHz), while having around -34dB attenuation above reject band frequencies (>125MHz). This is sufficient to filter out noise due to aliasing effects at

frequencies above 125MHz, while not affecting the lower frequency target signal within the pass-band.

The HMC1023 could also be utilized to provide some additional gain to our signal if needed.

### 2.3.2 Power Supplies

The main power source of the AFE module will be 5V from standard USB port or external power supply. The 5V rail should be regulated externally, but it will still be filtered through a LC filter with a ferrite bead to reduce potential high frequency noise on the rail. Then two on board power supplies provide 3.3V and -5V from the 5V input.

-5V is required by the input buffer (LMH6559) in order to allow rail to rail buffering operation for the 0 to 2.5V input signal. -5V will be generated with the LM27761 [6], switched capacitor voltage inverter with built in regulator. This component was selected for some of its key specifications listed below:

- 250mA max output current
- <1mV output voltage ripple
- inductorless design, only few external components required

This -5V power supply has been simulated and validated in LTSpice and should meet all the power requirements by the input buffer.

3.3V rail is required by the FDA (LMH5401) and it will be generated with the LT1763-3.3 [7] low noise LDO regulator. This component was selected for some of its key specifications listed below:

- 500mA max output current
- 20 $\mu$ V noise
- All-in-one module, only few external components required

This 3.3V LDO regulator has been simulated and validated in LTSpice and should meet all the power requirements by the FDA.

### 2.3.3 AFE Board

The AFE board is designed to contain all circuitry discussed in the previous

sections related to the AFE system. A render of the PCB is shown below in Figure 7.

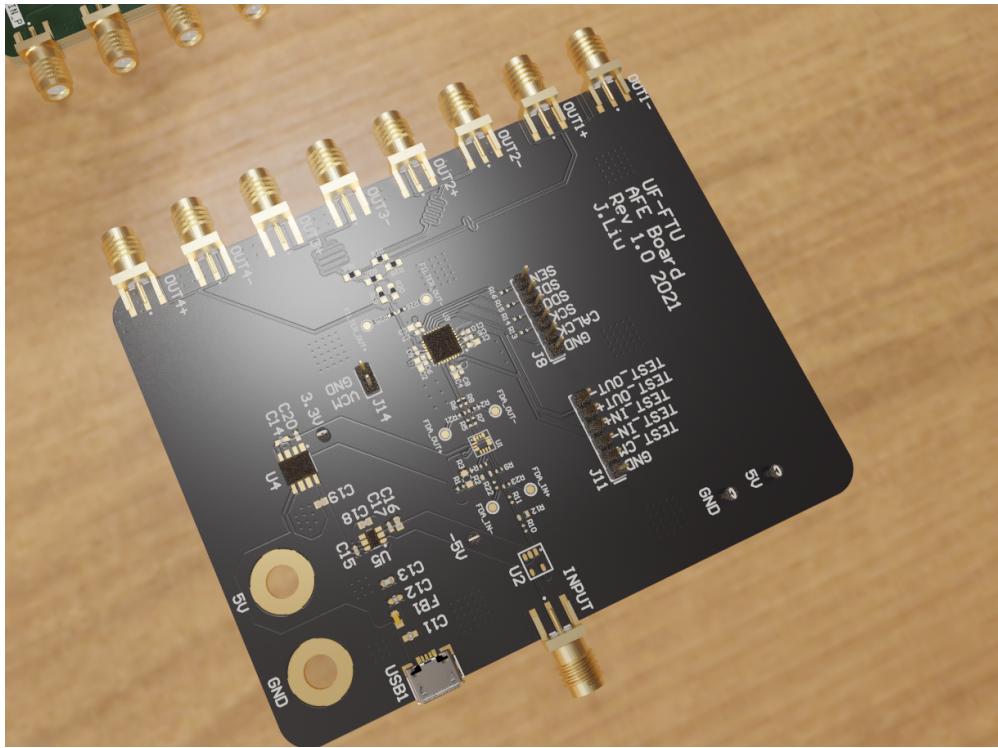


Figure 7: Analog Front End Board

The AFE board has an overall dimension of 80mmx100mm to mate with the digital board. It has a 4 layer stackup as shown below in Figure 8.

This 4 layer stackup design was chosen for better signal integrity for our sensitive analog signals, while still maintaining relatively low cost based on our PCB manufacturer - JLPCB's available processes.

#	Name	Material	Type	Weight	Thickness	Dk
	Top Overlay		Overlay			
	Top Mask	Solder Resist	Solder Mask		0.01501mm	3.8
1	Top Copper		Signal	1oz	0.035mm	
	Dielectric 1	2313	Prepreg		0.1mm	4.05
2	Signal Layer 1		Signal	1/2oz	0.0175mm	
	Core	FR-4	Core		1.265mm	4.5
3	Signal Layer 2		Signal	1/2oz	0.0175mm	
	Dielectric 2	2313	Prepreg		0.1mm	4.05
4	Bottom Copper		Signal	1oz	0.035mm	
	Bottom Mask	Solder Resist	Solder Mask		0.01501mm	3.8
	Bottom Overlay		Overlay			

Figure 8: AFE Board Stackup

In order to maintain good signal integrity for our sensitive analog signals, we implemented design rules on the AFE board to control the impedance of critical signal transmission lines, this will reduce impedance discontinuities and eliminate signal reflections. Controlled impedance values for our PCB stackup were calculated using JLCPCB's impedance calculator tool and verified using Altium Designer built-in impedance calculator tool as well.

In order to maintain timing consistency for interleaving sampling between the 4 ADCs, length matching techniques are used for critical PCB traces. The output signal from the AFE system is split into 4 differential pairs using resistive splitters, and then the 4 pairs are routed to 8 SMA connectors using the length matching tool in Altium. This will ensure that there will be minimal time delay between the 4 differential pair signals when the ADCs are performing interleaved sampling. A length matched trace example can be seen below in Figure 9.

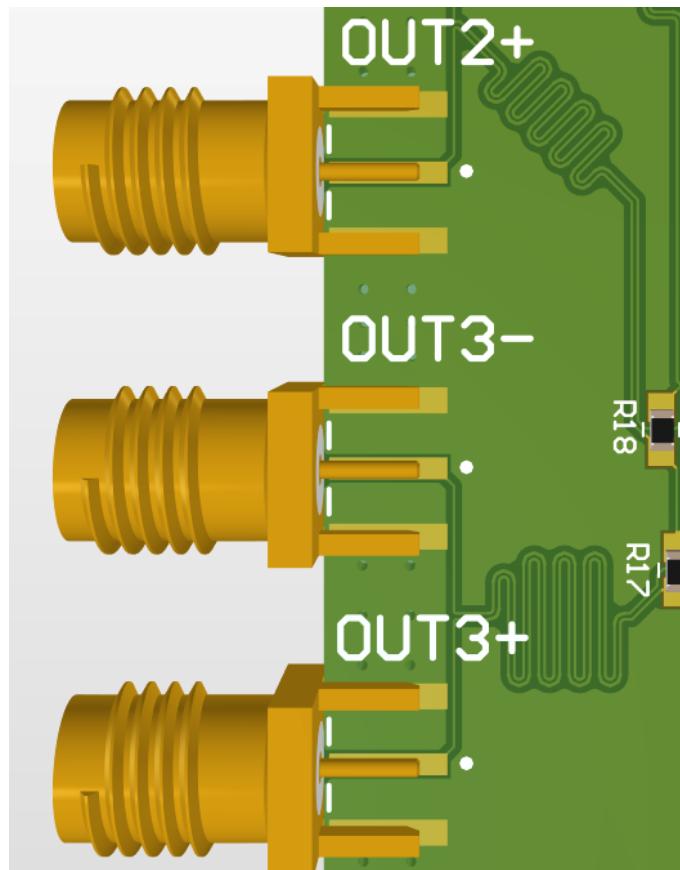


Figure 9: AFE Length Matched Trace Example

## 2.4 Clock Domain Buffering System

The Clock Domain Buffering system is used to transfer the collected data from the clock domain of the ADC to the clock domain of the double data rate buffer. This can be seen as the two FIFO blocks in Figure 4. It contains several embedded RAM based emulated dual clock FIFOs and a control logic implemented by a finite state machine. The purposes of having FIFOs and a state machine are:

- Dual Clock FIFOs allowed reading and writing at different frequencies. By using the feature of Dual Clock FIFOs, passing data through clock domains can be implemented
- FIFOs in Lattice ECP5 are RAM based emulated FIFO. Simultaneous reading and writing of the FIFOs are not supported. Thus, two FIFOs are required to buffer one signal channel under current read frequency (above 1GHz) and write frequency (250 MHz)
- For the two FIFOs of a single channel, there should always have one FIFO available for writing data in. So, a state machine is required to detect/change the status of the FIFOs, and also, change the input/output data flow based on the current FIFOs' status.
- The state machine detects the output flag signals (Full, Empty, Reset) of the FIFOs and also considers the current state to control the next states. State change of the state machine changes the input control signals(WrEn, RdEn) of the FIFOs, thus coordinating the read/write statuses of the FIFOs. The stage-transition diagram is shown below. The stages and the corresponding states are briefly explained below and detailed explanations are included in the in-code comments of statemachine.sv.

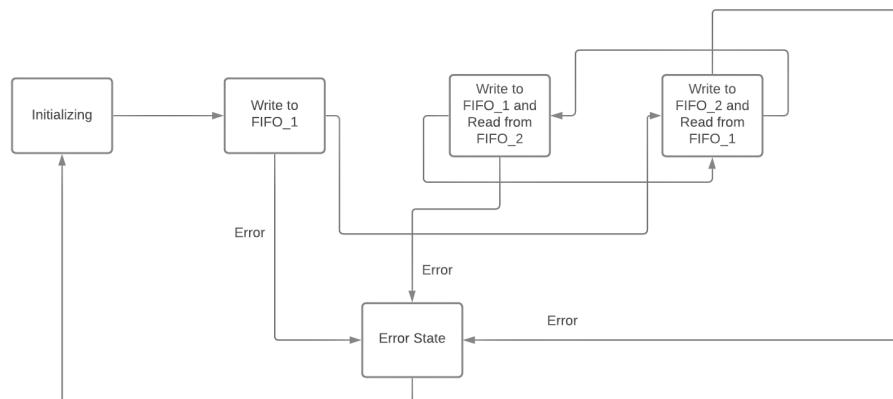


Figure 10: Stage-Transition Diagram

### Reset/Initial Stage

In this state, the on board reset button is triggered and Reset\_1 and Reset\_2 signal is triggered to reset the two FIFOs. FIFO\_1 write function is enabled and read functions of all the FIFOs are disabled. This stage contains three states.

### Emptying FIFO\_1 Stage and write to FIFO\_2

Writing input data into FIFO\_1 until FIFO\_1 is filled. This stage contains four stages.

### Emptying FIFO\_1 Stage and write to FIFO\_2

Writing input data into FIFO\_2 until FIFO\_2 is filled. This stage contains four stages.

### Error/Default state

Triggered by an error detect logic. The logic is not part of the state machine, but can change the state of the state machine into error state once unexpected signal feedback from FIFOs are detected (Eg. empty\_1 = 1 and full\_1 = 1) .

## 2.5 Frequency Counter

During testing the various stresses applied to the design under test will affect the device's output frequency characteristics. In the real world these can be non-linearly varying in accordance to the stressors, as such to measure this phenomena a Frequency Counter is required.

The frequency counter consists of an FPGA input pin, clocked at a high frequency relative to the frequency of the expected measurement signal.

- Frequency determination of asynchronous signals generated during transistor testing
- Expected frequency measurements are at 50MHz, therefore by using one of the high speed IO buffers on the FPGA we will sample the pin at 400MHz. This will allow us to determine the number of 400MHz clock cycles between each measured clock edge
- An abstraction of this could be to think of the measurements taken at the input as a discrete time quantization of the input signal
- This can be seen in Figure 11

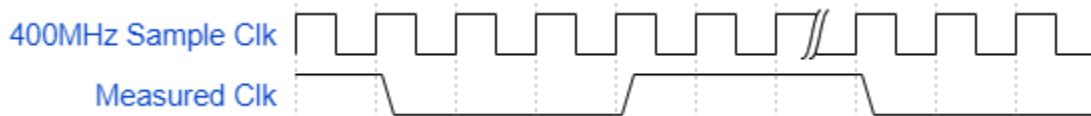


Figure 11: Frequency Counter Sample Waveform

## 2.6 Double Data Rate Buffer

The buffer consists of high speed DDR3 memory. DDR3 memory was selected as it conformed to the 9Gbit/s data storage requirement, while being available in densities of 8GB. Other benefits of DDR3 for this design are:

- Low cost and commercially availability
- High read and write speeds
- High storage capacities
- Ease of interface with the FPGA Controller

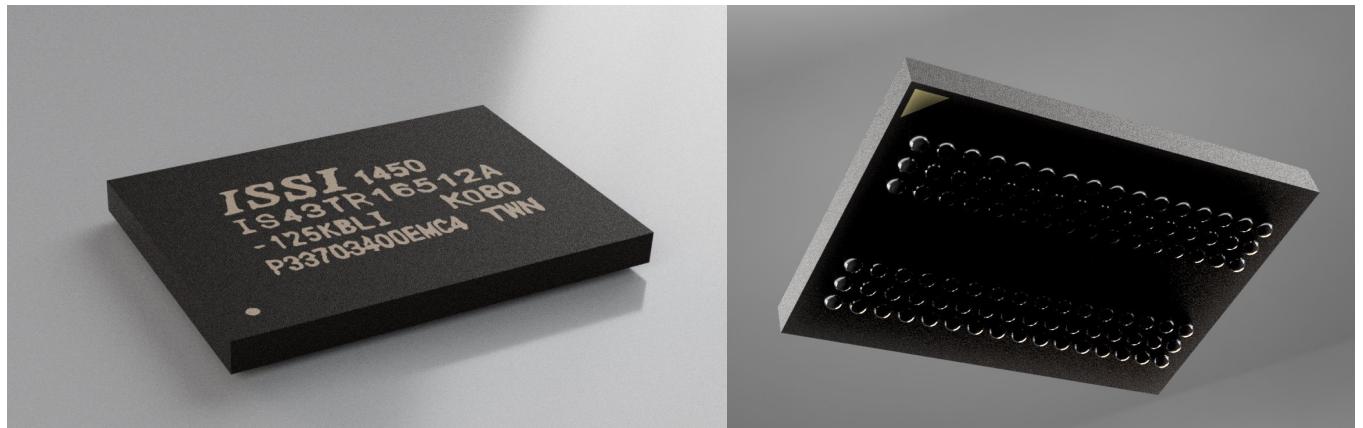


Figure 12: DDR3 Memory (IS43/46TR16512A)

The DDR3 Memory chip selected is IS43/46TR16512A [6] by ISSI, this allows us to transfer all of the incoming data from the 4 ADC's into the SDRAM as an interim buffer before data is finally transferred into non-volatile storage in the SSD bank:

- Total Storage capacity of 8GB DDR3 SDRAM
- Banks are arranged as 512M x 16bit
- Chip clock rate of 200MHz
- Bus clock rate of 800MHz
- Maximum transfer rate of 800 MT/s
- Maximum bandwidth of 12.8 Gb/s

On the board the IS43/46TR16512A [8] chip requires the following I/O with all lines being impedance matched due to very high operating speeds:

- Lines are operated 1.35V
- 16 bit address line (impedance matching is necessary)
- 16 bit data line (impedance matching is necessary)

- 15 bit GPIO control lines for configuring operations
- Clock input from the DDR3 Controller

The bandwidth has been chosen to exceed our needs to give us viable headroom on storing all data acquired from the ADCs while our FTU performs a test. The DDR3 memory is the highest speed component on our board with a max clock rate of 800MHz thus there are tight physical and routing requirements to be met during our board design to ensure signal integrity. Furthermore, extensive gateware and logic is required to allow the FPGA to interact with the DDR3 memory. This constitutes utilizing a full fledged DDR3 memory controller in compliance with the DDR3 JEDEC specification and building a state machine to pull data from the clock domain buffering system. This specification is further highlighted in the section 1.4 in the Verification and Validation document which also further details how it is tested.

These functions will be performed by the DDR3 Controller, which is written in SystemVerilog and targeted at the Lattice ECP5 FPGA architecture to run on our FPGA. This controller is designed to achieve high performance (relative to the clock speed) sequential read/write performance to exceed our design requirements. After rigorous testing as outlined in the Verification and Validation document, we utilize the Lattice Diamond DDR3 Controller for its rapid transfer rate in real world testing. We are able to achieve our desired memory rate with this controller for the memory chip.

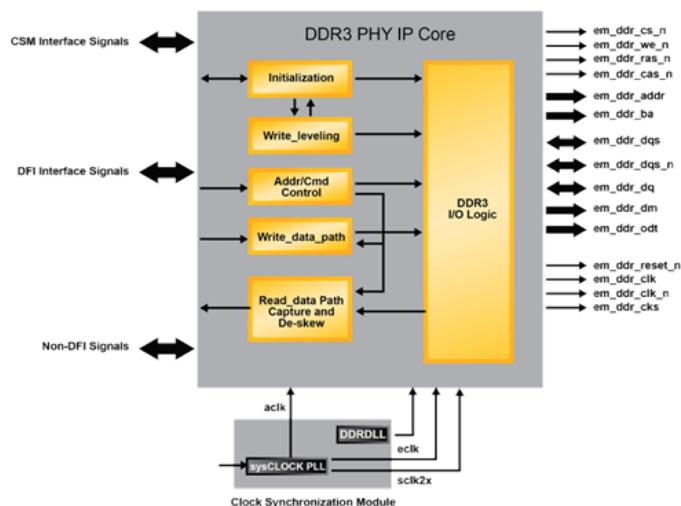


Figure 13: Lattice Diamond DDR3 Controller (JESD79-3)

Lattice Diamond DDR3 Controller (JESD79-3), which is a Lattice FPGA fabric based integrated controller which has support for unbuffered DDR3 DIMM

- Configurable to any standard address or data width
- Natively supported in the FPGA
- Requires interfacing with the FPGA fabric, adds extra cost to gateware development for the global state machine
- Supports: Xilinx 7 series, Lattice ECP5 FPGAs

## **2.7 Solid State Storage**

The non-volatile memory is used to store data for processing and analysis. It can be satisfactorily achieved by using a MicroSD card, this is chosen so that:

- Several data captures can be saved in “mass storage”
- Low cost commercially produced devices
- High capacity and reliability
- Ease of interface with the FPGA Controller using open source cores

Performance is not a key factor for this module as there is no maximum time requirement for emptying the DDR3 memory buffer into the mass storage. Stored data is formatted as comma separated value files, to allow for easy data manipulation with tools such as Matlab and Python. Once samples are transferred to the SD card the card can be inserted into the test computer for immediate processing.

## **2.8 Environment Generation**

While the environment generation is being developed by the co-op students in the Ivanov Group, we are working closely with them to ensure system compatibility. Some detail about this system is given in the following sections.

### **2.8.1 Microcontroller**

An ATSAMD21 microcontroller is used to support the legacy user interface used on the previous FTU design. This microcontroller features a high performance ADC and DAC used in the generation and control of test environments variables.

### **2.8.2 Heater Driver and Heater Click**

The heater for the DUT consists of a printed circuit board with narrow traces running along it. By providing a PWM signal across this resistive load, and measuring the temperature with a K-Type thermocouple, a controlled heater is formed. The supply voltage for this load is controlled by a low side N-Channel MOSFET. The temperature is measured using the on-board ADC and a PID loop is used to maintain a constant temperature.

### **2.8.3 Helmholtz Driver and Coil**

To control the environment of the DUT, a Helmholtz coil has been fabricated. This coil consists of several windings of wire around a 3D printed enclosure. A driver circuit capable of driving a controlled current through these windings allows a controlled magnetic field to be produced. The driver circuit consists of a MOSFET being driven by an LM358 amplifier stage. A varying PWM signal is used to drive these op amps to control the current driven through the windings of the coil.

### **2.8.4 Voltage Stress Circuit**

To provide a bipolar input voltage swing to the DUT, split rail power opamps are used to provide the output voltage. The output voltage is based on a DC voltage generated by the DAC of the microcontroller. This DC voltage is biased using an LM358 op amp before being fed into the non-inverting power buffer.

## 2.9 Digital Board

The digital board contains all of the digital areas associated with the design (See Figure 4). As this board contains many high speed signals, several factors must be carefully considered when designing it.

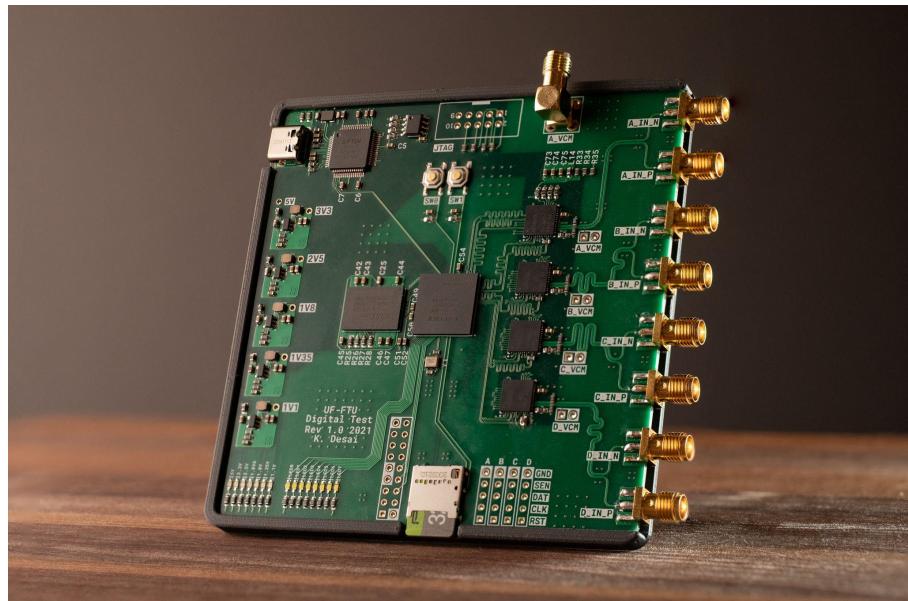


Figure 14: Digital Board

### 2.9.1 Layer Stackup

As the FPGA selected is in a ball grid array package with a 0.8mm pitch, a multilayer PCB is necessary to effectively fan out the design. The design also requires controlled impedances, therefore good ground reference planes are necessary. The high speed, length matched signals associated with the DDR3 memory drove the decision to use a 6-layer process, with the stackup shown below. The stackup is based on JLCPCB's available processes, as this is the most cost effective PCB manufacturing service available.

#	Name	Type	Material	Thickness	Dk	Df	Weight
	Top Overlay	Overlay					
	Top Solder	Solder Mask	Solder Resist	0.5mil	3.8		
1	Top Layer	Signal		1.378mil			1oz
	Dielectric 4	Prepreg	2313	3.937mil	4.05	0.02	
2	Ground1	Plane		0.689mil			1oz
	Dielectric 2	Core	2313	22.244mil	4.05	0.02	
3	Layer 2	Signal		0.689mil			1/2oz
	Dielectric 1	Prepreg	2116	41.929mil	4.25		
4	Power + Signal	Signal		0.689mil			1/2oz
	Dielectric 3	Core	2313	22.244mil	4.05	0.02	
5	Ground2	Plane		0.689mil			1oz
	Dielectric 5	Prepreg	2313	3.937mil	4.05	0.02	
6	Bottom Layer	Signal		1.378mil			1oz
	Bottom Solder	Solder Mask	Solder Resist	0.5mil	3.8		
	Bottom Overlay	Overlay					

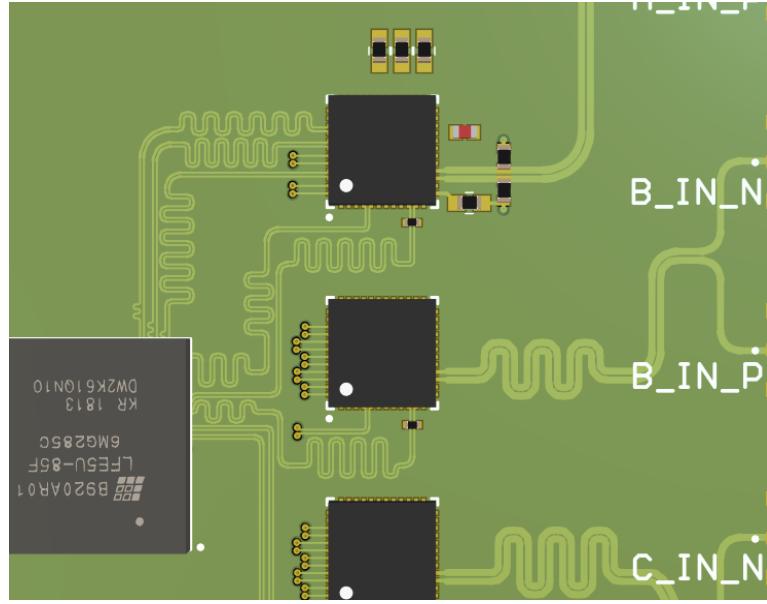
Figure 15: Digital Board Layer Stack

### 2.9.2 Controlled Impedance

As our system deals with high speed analog and digital signals, it is important to control impedance of transmission lines. This is done to prevent impedance discontinuities that could result in signal reflections, and ultimately poor signal integrity. Using the above layer stack and the impedance calculation tools within Altium Designer 2021, we are able to calculate the trace width and space. The two transmission impedances on our PCB are 50 and 100 Ohm.

### 2.9.3 Length Matching

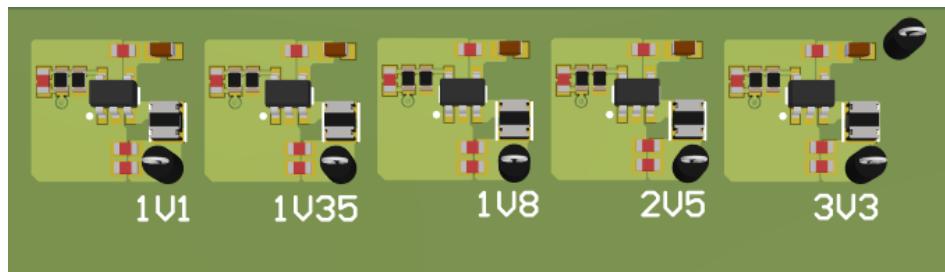
To ensure timing criteria for the digital logic is met, and to minimize on clock skew, length matching PCB traces are required. These traces ensure signals have a similar path length when travelling to a receiver. The DDR3 memory, the ADCs and all clocks require minimized skew to allow maximum performance to be reached. Below is an example of length matching of the data lines on the 4 ADCs. The maximum amount of skew allowed is calculated based on the timing budget of the components, and is based on setup and hold time constraints.



*Figure 16: Length Matched Trace Example*

#### 2.9.4 Power Supplies

To operate several components and functions on the digital board, a group of switch mode converters are used to generate the different voltages. The buck converters used are the TLV62568. This component was selected as it can be used to generate all five voltages used, and also due to prior design heritage in FPGA systems. Using the same component allows for the bill of materials to be reduced, allowing for better unit cost. The layout of these converters is shown below. This converter utilizes an integrated switch; this works together with the shielded inductor to limit noise generated by the switching node from being coupled into the sensitive analog section of the board. These components are also placed as physically far from the analog devices as possible.



*Figure 17: Digital Board Power Supplies*

## 2.10 State Machine

The state machine is responsible for the bulk of the digital boards operations such as running the overall test cycle while managing all initializations, settings, inputs and outputs for the board. The state machine runs on the Lattice ECP5-85F FPGA. The key reason a FPGA is utilized in our design is how the two differ in the methods used to process instructions. Microcontrollers read through each line of the program sequentially; that means the commands are also processed in sequence.

FPGAs can process orders simultaneously and can execute numerous lines of codes at a given moment. They are also wired just like an electric circuit so suitable parallel circuits can be created.

We utilized the Lattice Diamond software by the FPGA chip manufacturer to compile the SystemVerilog code into a usable programming (.sof) file that can be programmed into the FPGA. The FPGA can also be set to save it's program to automatically boot load it once it is turned on.

Functionality is developed as a Mealy machine, which is a finite-state machine whose output values are determined both by its current state and the current inputs. The functionality is divided into 9 major blocks for ease of code development and management, it is structured like this as this project is going to be updated and worked upon in the future for configuration to other use cases for our client.

### 2.10.1 Initializing ADCs, DDR3 Controller, FIFO & SD Controller

The initialization process involves directing power and enabling signals to all components that are required for operations. Different modes are set in the state machine which can modularize the components being enabled or disabled to save power or reduce cross talk by shutting down components that are not necessary.

## 2.10.2 Responding to User input to start or stop a test

The device gets ready to start a testing cycle after boot up and initialization process is complete. The test cycle begins after the user initiates it through physical push buttons on the board. Push button (SW1 on the physical board) is used to start a full testing cycle.

## 2.10.3 Controlling the existing board to start testing the DUT

This stage involves sending the existing client board configuration signals to begin applying the desired stressors to the DUT.

## 2.10.4 Displaying current Status through LEDs

For optimal user feedback the onboard LEDs are utilized to display the stage and status of the state machine. These can be easily updated in the program to suit different testing needs or user requirements.

## 2.10.5 Directing data flow from ADC to FIFO

Once initiated the state machine starts sampling the 4 ADC inputs simultaneously which are fed into the Clock Domain Buffering System (section 2.4).

## 2.10.6 Directing data flow from FIFO to DDR3 Controller

The time series data from the FIFO is then interleaved through an interleaving algorithm which minimizes faulty readings and ensures data validity along with a checksum to be able to verify data integrity once it has been moved into volatile DDR3 Memory (section 2.5)

## 2.10.7 Storing data into SD Card at the end of a failure testing cycle

The data is rapidly compressed using Huffman encoding which is an optimal prefix code widely used for lossless data compression. This helps maximize storage on the SD card to enable as many tests as possible before data has to be transferred out and archived. The data is stored into the SD Card in FAT32 format as a comma separated values file (.csv). These files can be easily accessed on any suitable computer.

## 2.10.8 Resetting and preparing for next failure testing cycle

After a successful testing cycle all necessary modules are reset and FIFOs are cleared to ensure a clean data pipeline. The clock is also resynchronized to ensure calibrated performance directly preventing possible time skew in the FPGA interconnects.

## 2.11 Visualization and Data Processing Software

To display the collected data off of the SD Card we have developed a Python program which visualizes the wave with accurate time characteristics, with the ability to speed up or slow down the replay time. It utilizes a web application front end to display the graph output to any web browser. This allows it to be set up on any computer on the network and be viewed on any other computer if necessary, this feature could be especially useful if a small single board computer such as Raspberry Pi may be utilized in the future to control our Digital Board to further automate the post testing cycle analysis.

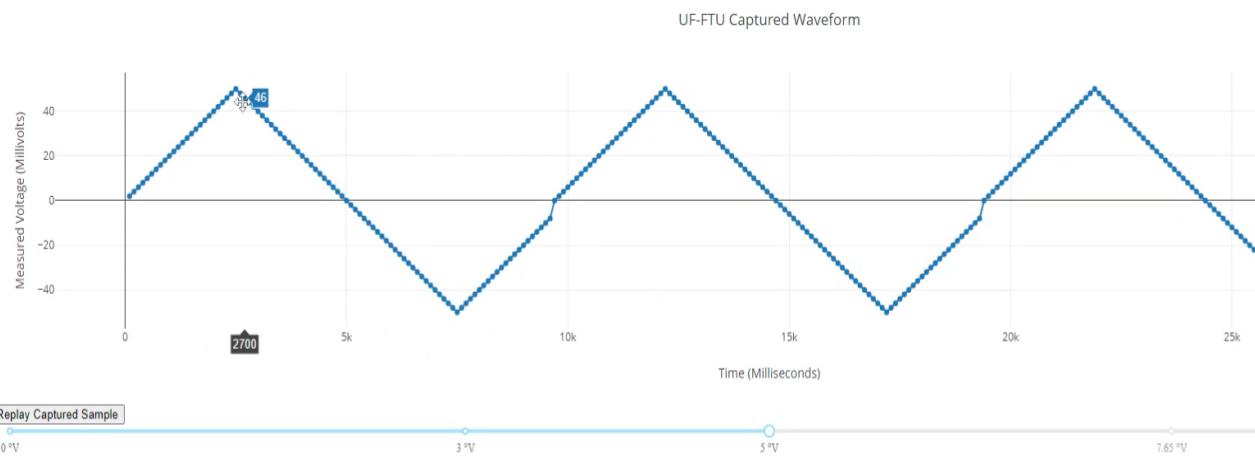


Figure 18: Visualizer Wave Replay Feature (full triangle sample wave)

The GUI is designed to be highly configurable and just requires basic Hypertext Markup Language (HTML) to incorporate additional elements and develop further functionality and can be stylized by Cascading Style Sheets (CSS).

The data processing back-end interface allows for implementing various manipulations of the wave such as choosing time segments, or formulaically processing the data e.g computing and displaying the FFT output of a selected segment or wave. This tool can be further expanded upon in Python to support additional use cases by editing the documented code.

### 3.0 Data Flow

To further explain the high-level operation of the system we have developed a data flow. For the scope of this project it is focused on the measurement of the DUT and does not include the environment generation. The flow is also shown by the high level system architecture, shown below.

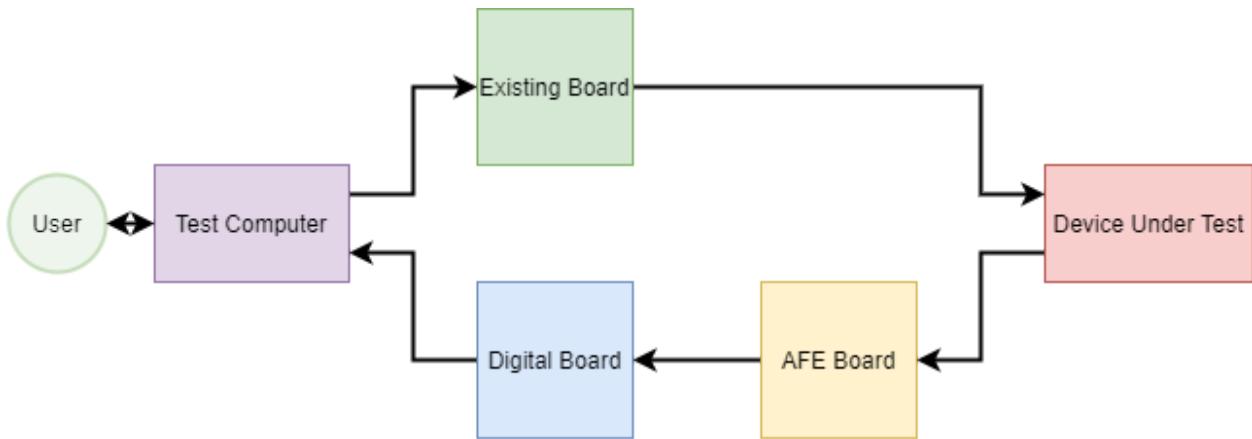


Figure 19: Data Flow

This flow begins when a user configures a test on the test computer. The parameters of the test are transferred to the existing board (Green Box) via a serial connection. Here, environment variables are set such as temperature and magnetic field. A test waveform is then generated and passed to the Device Under Test (Red Box) through a coaxial input. The device under test is a CMOS transistor device, configured to act as a buffer. The output of the DUT is then ready to be collected and compared to the test waveform. This signal is of the LVCMOS type, meaning that it is a single ended 0-2.5V signal.

This signal enters the Analog Front End of our design, which first buffers the signal, presenting a high impedance to the DUT. The signal is then scaled down and converted to a balanced differential signal through a Fully Differential Amplifier. This is then fed into a 6th order Butterworth Low Pass filter with a cutoff frequency of 65MHz. This filter is used to filter unwanted noise and acts as an antialiasing filter. After the filter stage, the signal is passed through a Variable Gain Amplifier in which it is scaled for the ADC input voltage range of  $1.5V_{pp}$ .

The signal is then passed (via coaxial cable) into 4 identical 250MSPS ADCs which are clocked

with a 90 degree phase shift between them, also known as quadrature clocking. This means that an effective sampling rate of 1GSPS is achieved. The 4 ADCs pass the collected 9 bit samples over an LVDS interface to the FPGA.

Within the FPGA, two FIFOs are used per ADC to buffer the input signal and synchronize the data with the system clock of the FPGA. The data is then sent to the DDR3 controller contained within the FPGA. The controller organizes the data and it is sent via a 16 bit DDR interface to be stored in DDR3 memory.

Finally, after the DDR3 memory has been filled, the controller reads back the contents and stores it within the mass storage drive. From here the data can be transferred to the user's personal computer as a comma separated value file. The file is then ready for any processing, after which it can be displayed to the user.

## References

- [1] Lattice Semiconductors, "ECP5 and ECP5-5G Family," FPGA-DS-02012 datasheet, Mar. 2014 [Revised Mar. 2018].
- [2] Texas Instruments, "11-Bit, 200MSPS/9-Bit, 250MSPS, Ultralow-Power ADCs with Analog Buffer," ADS58B18 datasheet, Nov. 2009 [Revised Jan. 2011].
- [3] Texas Instruments, "LMH6559 High-Speed, Closed-Loop Buffer," LMH6559 datasheet, Apr. 2003 [Revised Mar. 2013].
- [4] Texas Instruments, "LMH5401 8-GHz, Low-Noise, Low-Power, Fully-Differential Amplifier," LMH5401 datasheet, Oct. 2014 [Revised Feb. 2018].
- [5] Analog Devices, "72 MHz Dual Programmable low pass filter with driver," HMC1023LP5E datasheet, v01.0113
- [6] Texas Instruments, "LM27761 Low-Noise, Regulated, Switched-Capacitor Voltage Inverter," LM27761 datasheet, Oct. 2015 [Revised Jan. 2017].
- [7] Linear Technology, "500mA, Low Noise, LDO Micropower Regulators," LT1763 Series datasheet, May. 2010 [Revised Nov. 2014].
- [8] ISSI, "512Mx16 8Gb DDR3 SDRAM," IS43/46TR16512A, IS43/46TR16512AL datasheet, Feb. 2018