Documentation Report

Implementation of Serial Interface Based Digital to Analog Converter for Electrical Impedance Tomography System

Under the guidance of

Dr. Gary J. Saulnier

Department of Electrical and Computer Engineering (ECE)
University at Albany
College of Engineering and Applied Sciences (CEAS)

Contents

List of Figures

Chapter 1

Introduction

This chapter provides an overview of thez Electrical Impedance Tomography system and about the channel setup implemented in it.

1.1 About ACT 5 EIT System Architecture

The ACT 5 electrical impedance tomography (EIT) system is an imaging tool designed to provide real-time images of conductivity and susceptivity in the human body.

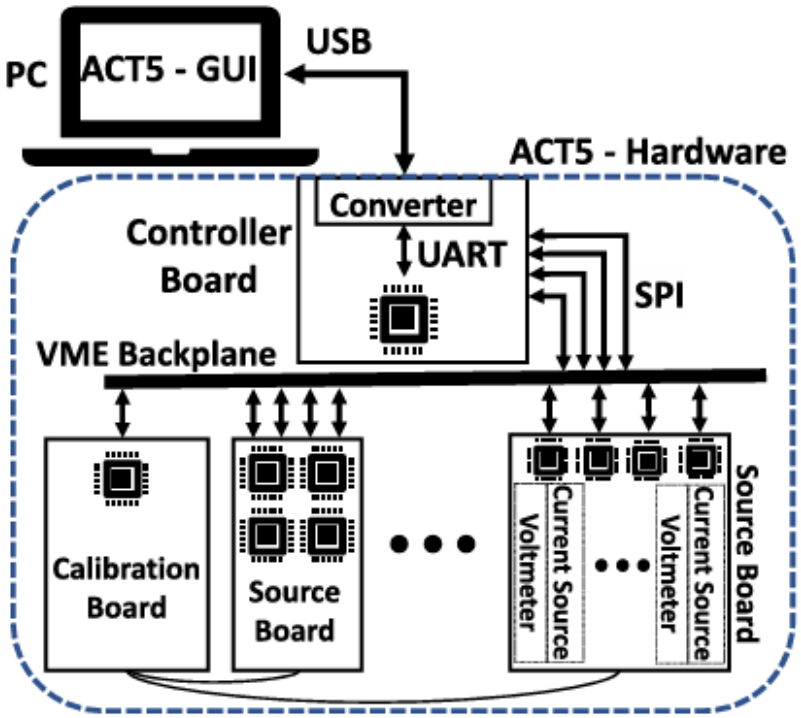

Figure 1.1: Overview of Hardware Architecture of ACT 5 [1].

The ACT 5 system uses 32 electrodes, each linked with a current source and a voltmeter. Varying current signals ranging from 5 kHz to 500 kHz can be programmed and sent over these electrodes.

The ACT 5 EIT system can display real-time images at a rate of 27 frames per second derived by processing voltage data from the electrodes. The frame-rate, which can also be increased to 54 frames per second when using 16 or fewer electrodes, allowing for detailed imaging of dynamic physiological processes such as breathing and blood perfusion.

One of the primary applications of the ACT 5 system is thorax imaging, particularly for monitoring lung performance in cystic fibrosis patients and monitoring respiration of infants under ventilator care. The system can also simultaneously record electrocardiogram (ECG) data, providing comprehensive cardiac cycle imaging.

1.2 ACT 5 Channel Implementation

Each channel uses a 16-bit Digital to Analog Converter (DAC), specifically the Linear Technology LTC1668, which operates at 24 mega samples per second Msps. The digital samples from the DAC are fed into a Howland voltage-to-current converter, transforming the digital signals into the desired current signals.

The FPGA board (Artix 7 Arch) on each channel board is responsible for managing the digital processes. It creates the digital waveform, measures the voltage by reading values from ADC, and compensates for any shunt impedance effects as well. Adaptive current sources update their output by constantly checking voltage levels to make sure the right amount of current reaches the load. This real-time adjustment is key to making up for any current lost through shunt impedance.

The voltmeter samples the signal using an 18-bit Analog to Digital Converter (ADC) at 1.2 MS/s. These samples are processed using quadrature matched filters, which integrate over 1024 samples and the excitation frequencies are chosen so that 1024 samples correspond to an integral number of waveform cycles.

Chapter 2

Goal of the Project

This chapter discusses the goal of this project and the approach being taken to achieve it.

2.1 Goal

In ACT 5, each channel has its own FPGA (4 FPGAs per board). For ACT 6, we aim to use 1 FPGA per board. However, supporting 4 parallel DACs (4 x 17 lines) with a single FPGA is challenging due to limited pins and complex PCB routing. Therefore, we plan to switch from parallel DAC to serial DAC, which requires fewer pins.

To design and implement a serial interface for the existing EIT system, replacing the current parallel interface based DAC. The new system must generate sinusoidal waveform at 12 mega updates per seconds (MUPS) while ensuring compatibility with EIT system's requirements and performance specification.

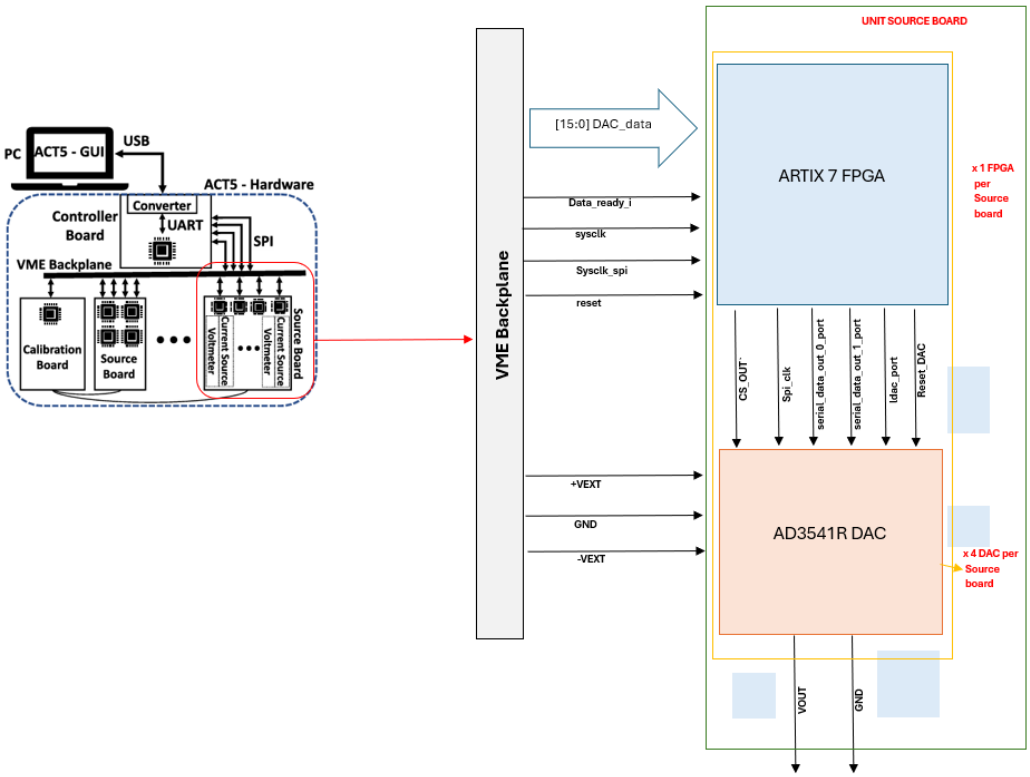

Figure 2.1: FPGA Interface with Serial DAC

2.2 Approach

We will be using Analog Device's AD3541 DAC for the project. This uses serial peripheral interface (SPI) to communicate with the master board. We will use Verilog programming language in AMD Vivado tool to program the FPGA board to communicate with DAC module which also employs IPs to generate clock and waveform data. We will use various capabilities of the DAC, such as dual SPI communication, double data rate, and streaming mode, to achieve 12 MUPS. The design is first implemented and tested on a CMOD module and later verified on the ACT 6 board.

Chapter 3

Modules Overview

This section provides a detailed overview of the components used in this project. It includes descriptions of each component and their functions.

3.1 AD3541 Digital to Analog Converter

The AD3541R is a 12-/16-bit ultra-fast DAC with low drift and this is the DAC used in this project. It operates with a fixed 2.5 V reference and can be configured in multiple voltage span ranges. The device has five pre-configured output voltage ranges: (i) 0 V to 2.5 V, (ii) 0 V to 5 V, (iii) 0 V to 10 V, (iv) −5 V to 5 V, and (v) −2.5 V to 7.5 V.

The AD3541R-16 offers fast mode for maximum speed or precision mode for maximum accuracy, while the AD3541R-12 has a single operation mode.

The serial peripheral interface (SPI) on the DAC supports dual SPI and single SPI (classic SPI) modes with single data rate (SDR) or double data rate (DDR), using logical levels from 1.2 V to 1.8 V.The AD3541R is specified for an extended industrial temperature range (−40 °C to 105 °C).

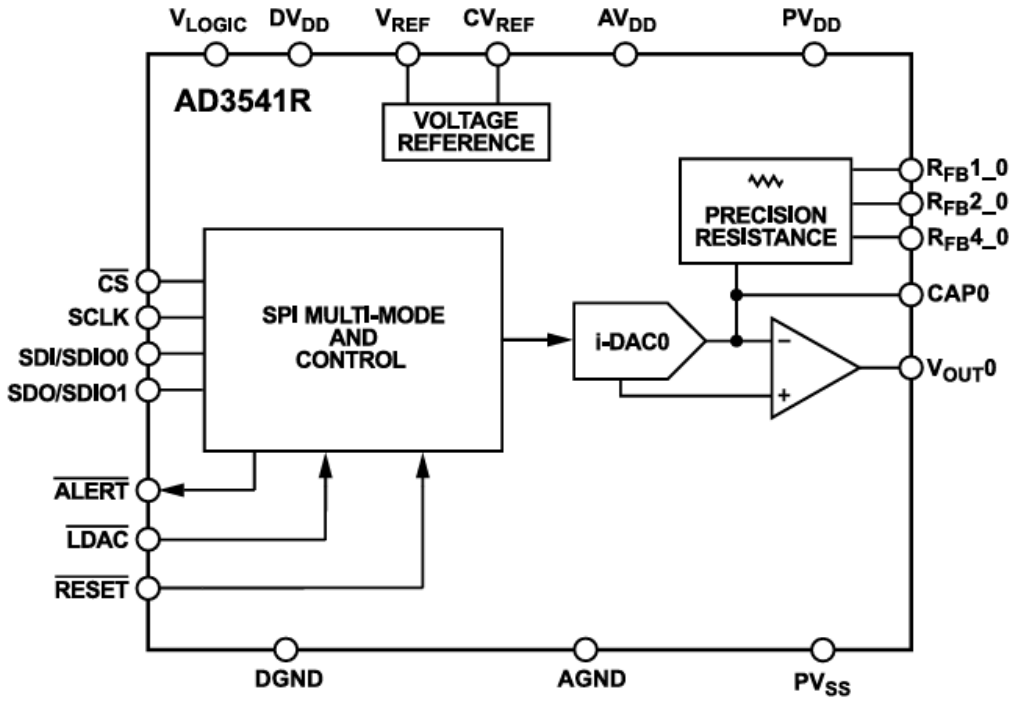

Figure 3.1: AD3541R Functional Diagram [2].

The block diagram shown in Figure 3.1 explains how the DAC functions. The IC receives digital input data via the SPI interface. This data is primary the voltage data that needs to be set. This digital input is converted to an analog voltage by the transimpedance amplfier with the help of feedback resistors RFB1-0, RFB2-0, and RFB4-0. The output is connected to external capacitor for stability while the control signal $\overline{\text{LDAC}}$ and $\overline{\text{RESET}}$ are used to load and reset the module respectively.

AD3541R's applications include instrumentation, hardware in loop, medical devices, optical communications, Data acquisition system, and programmable voltage sources.

3.2 CMOD A7

The Digilent CMOD A7, shown in Figure 3.2, is a 48 pin Dual-in-line package board built on Artix-7 FPGA (XC7A35T-1CPG236C). It can be programmed via USB / JTAG interface.

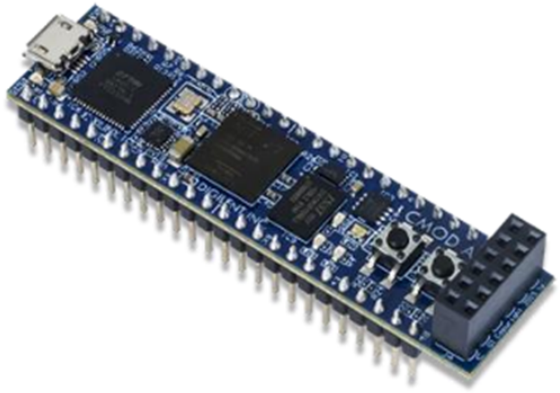

Figure 3.2: CMOD A7 Board [3].

It has a 512 kB RAM with 8-bit bus and 4 MB Quad-SPI flash on which we load the program. Once powered on, the program boots up from this memory. The FPGA module can be interfaced with USB-UART/FIFO (FTDI FT2232HQ) bridge to transfer real-time data to the host computer. This is shown in figure 3.3

Figure 3.3: SPI Flash and UART Interface on CMOD A7 [3].

The board also has interaction devices such as LEDs and push buttons. Of the 48 available pins on the module shown in Figure 3.4, 44 are digital I/Os, 2 are analog inputs with 0 V-3.3 V rating, one is the 5 V input power pin, and the last one is the ground pin. Please note that this board has been used in the development and evaluation phase only.

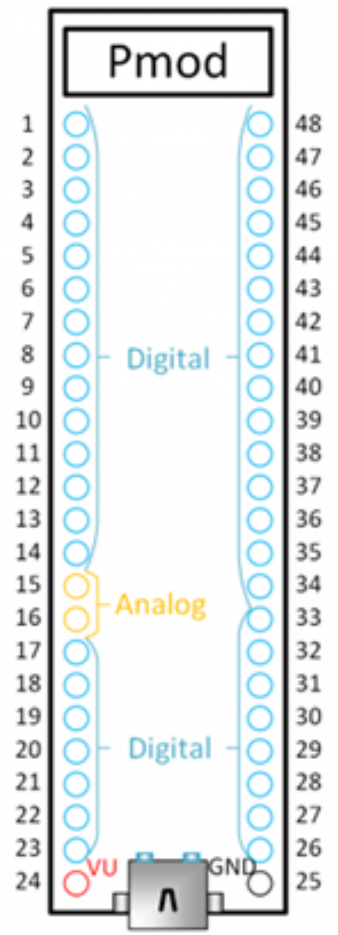

Figure 3.4: DIP Pin Diagram of CMOD A7 [3].

3.3 DAC 13 Click board

The DAC 14 Click board (Figure 3.5) is the evaluation board for AD3541R DAC module. The board provides an option to connect with external transimpedance amplifier marked with VEXT and can be selected using the onboard SMD jumper. The board can operate with either 3.3 V or 5 V voltage levels which can be selected through VCC SEL jumper and the conversion is done with the use of voltage translator. Please note that this board has been used in the development and evaluation phase only.

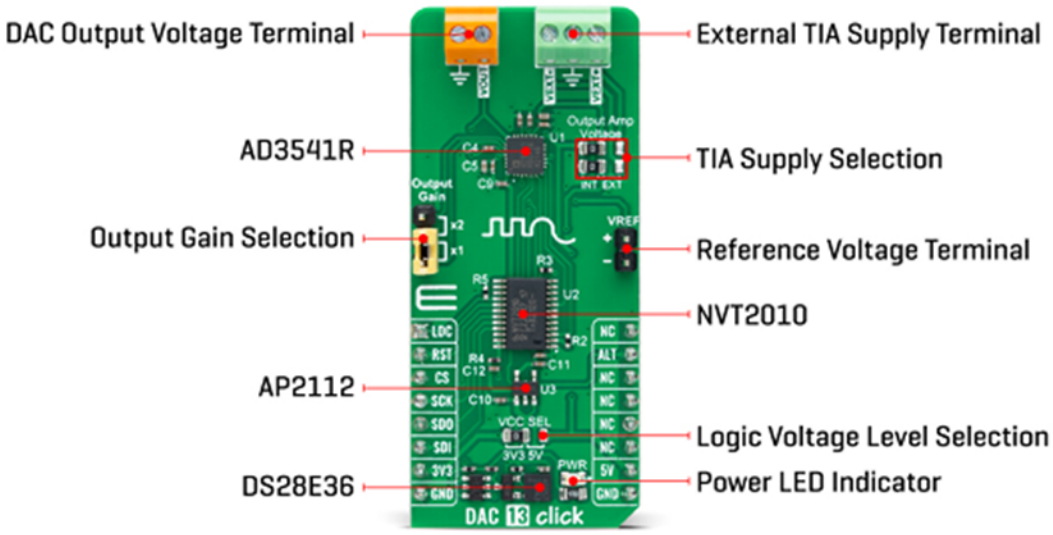

Figure 3.5: DAC 13 Evaluation Board[4].

Chapter 4

Operating Conditions

This chapter gives the details of parameters considered in implementing the DAC system.

- **DAC input data length/resolution:** 16 bits
- **Update rate:** 12 MUPS
- **Update mode:** Fast mode (16 bit)
- **Error detection/correction:** None / CRC Disabled
- **SPI Clock frequency:** 48 MHz
- **System clock:** 192 MHz [48 MHz x 4]
- **Serial Data Lines:** 2 transmission lines operated in Dual SPI Mode
- **Communication Mode:** Configuration done on single SPI SDR. DAC Data loaded through streaming mode.
- **DAC Module signal pins voltage:** 1.8 V
- **Output Analog Voltage range:** 5 V to −5 V

Chapter 5

Methodology

This chapter explains how the DAC system is implemented in detail. It explores the working functionality of the Verilog code, the role of each port involved and ~~the about~~ the setting done on configuration registers.

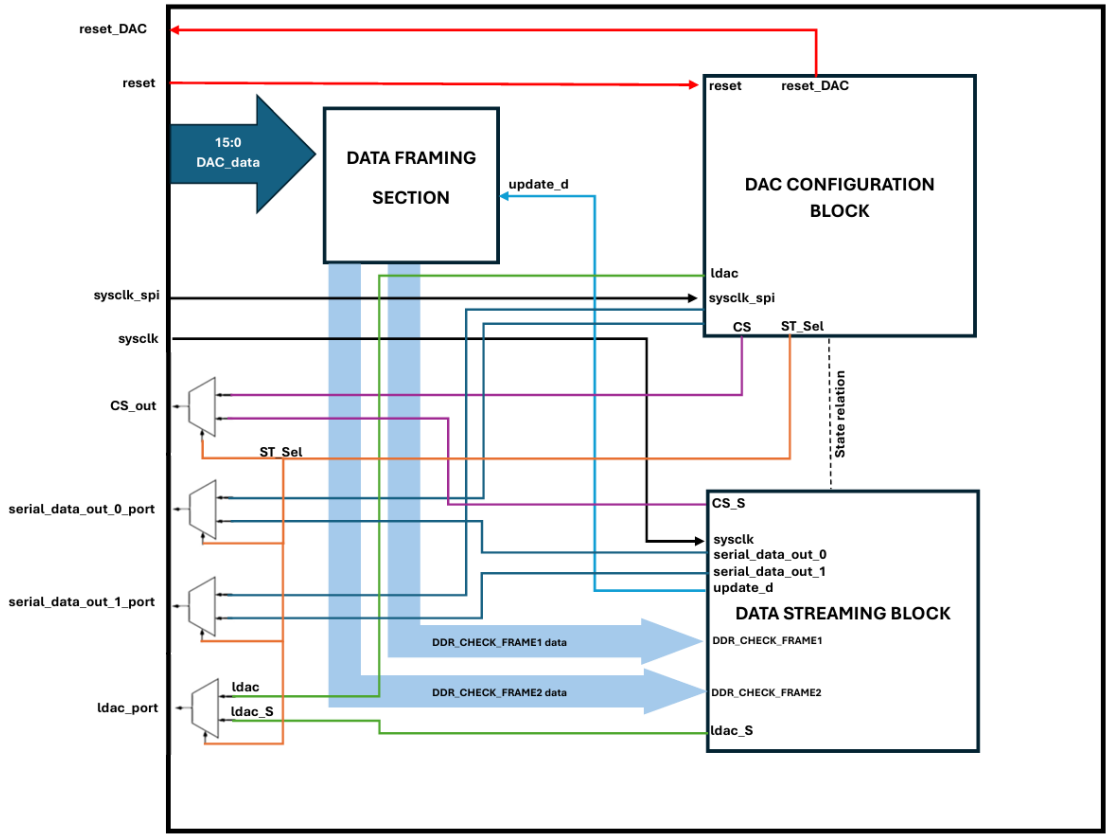

Figure 5.1: Serial DAC Verilog Code Block Diagram

5.1 Code Mechanism

This section describes the Verilog code mechanism, which is consisted of three main sections, data framing section, data configuration block, and data streaming block

5.1.1 Data Framing Section

The first section is on the data framing. A 16-bit parallel data is provided as an input to the FPGA unit. This data needs to be structured as per the expectation of serial ports. To do this, we are configuring two frames. Both frames are 16-bit long. The first frame, `DDR_CHECK_FRAME1`, is the primary serial port which carries the write bit and the address bits along with the data bits. The second frame, `DDR_CHECK_FRAME2`, is meant for transferring the data; therefore, the most significant bit (MSb) bits are appended with 0s. Additionally, the `DDR_CHECK_FRAME1` carries data of even bits (0, 2, 4, ... 14 bits) while the other frame carries data of the odd bits. The data from input is updated into the frame initially when the `data_ready_i` is triggered and in subsequent cases whenever `update_d` is triggered.

5.1.2 Data Configuration Block

~~**Data Configuration Block:**~~ This block is responsible for initializing all the configurations necessary for the expected output and the way data is processed by using a state machine. To update the DAC at 12 MUPS, we will need to transmit data in double data rate, dual channel, and in streaming mode together. Also, the output range of the analog signal must be between +5V to -5V. These settings are configured by accessing the corresponding register and setting the data. These registers and the data written to them are mentioned in Section 5.3. All the SPI communication in this block takes place on the negative edge (neg-edge) of `sysclk_spi` and in classic mode. Refer to the Figure 5.2 to see the sequence of data transmission. When a RESET signal is triggered, all ongoing tasks are terminated, the state machine is set to the IDLE state, and the boot-up begins. The system waits until the `reset_wait` is complete, and then proceeds to reconfigure all the necessary registers. Once the configuration is complete, `ST_Sel` is set and the system moves to streaming mode which will be handled by the data streaming block.

5.1.3 Data Streaming Block

This block is responsible for streaming data. This block uses the neg-edge `sysclk`, which has 4 times the frequency of `sysclk_spi`. This mode has a counter to keep track of bits being propagated. The write and address bits are sent in single SPI mode and the data is sent in dual SPI DDR mode. Once the transmission begins, the address needs to be sent only once while the data can be transmitted continuously. The sequence in which the data is sent can be seen in the Timing Diagrams section. We will only address register 0x2A (first 8 bits from MSB) and since the streaming mode is set to two bytes, the streaming mode will automatically address the rest of the least significant bit (LSb) 8 bits of data to 0x29.

5.2 Port List Information

1. The design has a total of 10 ports, out of which 4 are inputs and 6 are output ports.
2. The clock signals `sysclk` to differentiate! This applies to all such signals and spi_clk are sourced from clocking wizard.
3. The DAC_data port is is disabled when unit testing as the DAC data is generated withing the code itself, usually a square or a ramp signal.
4. When the design is used on the main system, the DAC_data port is enabled and the data is sourced from the direct digital frequency synthesizer (DDS) IP.

Table **??** provides information about individual port being used in design.[.]

Table 5.1: **[I have redone the table for comparison. A lot of it is personal taste, but usually tables without "boxes" are more appealing. There is also a "tabular" package that has "toprule", "midrule", and "bottomrule" commands instead of "hline" which are apparently better in tables!]**

Port Signals	Direction	Description
data_ready_i	Input	Signal that tells the FPGA that the input data has been generated and is ready to be loaded into transmitting buffer register. This signal is HIGH only once after the startup.
DAC_data	Input	16-bit parallel input data received from the Master controller board via the VM backplane. The sinusoidal waveform generated on the output depends on the value being set on this.
sysclk	Input	192 MHz clock signal that is used to stream the data to DAC module. The frequency of this clock should always/at least be four times the spi_clk.
reset	Input	Active LOW signal used to reset the FPGA module and all the initialization sequence and configurations will be redone. Additionally, once triggered, the DAC will also be reset by triggering signal on reset_DAC pin.
CS_out	Output	Chip Select signal is an active low signal. This signal should be low when data is being transmitted to the DAC.
spi_clk	Output	Clock signal used for single SPI write operations and as the clock signal for DAC module. For achieving 12 MUPS, this clock is operated at 48 MHz.
serial_data_out_0	Output	Primary port that is used to transmit data to DAC. All write operations are done through this port in Single SPI write operation while the address and even bits of the DAC_data are transmitted in Dual SPI write operation.
serial_data_out_1	Output	Secondary data transmission port. Only used in Dual SPI write and carries only the odd bits of DAC_data. This pin is not used to transmit address of register.
dac_port	Output	Used when loading data to DAC_INPUT register (0x34). Currently DAC data is loaded into address 0x2A and this pin needs to be high throughout.
reset_DAC	Output	Port used to send reset signal to DAC.

Port Signals	Direction	Description

5.3 AD3541 Register Configuration

INTERFACE_CONFIG_A [0x00]			
Bit	**Value Set**	**Bit Name**	**Description**
7	0	SW_RESET_MSB	Do nothing
6	0	RESERVED	
5	0	SDO_ACTIVE	SDO disabled to enable SDIO1. This will enable the FPGA to transmit data over serial_data_out_1_port
3:1	000	Reserved	
0	0	SW_RESET_LSB	Do nothing

INTERFACE_CONFIG_B [0x01]			
Bit	**Value Set**	**Bit Name**	**Description**
7	1	SINGLE_INSTRUCTION	Set to single instruction to configure other registers. This register will be reconfigured later
6:4	000	RESERVED	
2	1	SHORT_INSTRUCTION	Set to 7–bit addressing
2:0	000	Reserved	
0	0	SW_RESET_LSB	Do nothing

INTERFACE_CONFIG_TX [0x7F]			
Bit	**Value Set**	**Bit Name**	**Description**
7:6	00	MULTI_IO_MODE	Set to single SPI for configuration purpose. Will be later changed.
5:3	000	RESERVED	
2	1	STREAM_LENGTH_KEEP_VALUE	Stream length to remain same when streaming is completed. Therefore, we need not reconfigure the length for next streaming cycle.
1:0	00	Reserved	
0	0	SW_RESET_LSB	Do nothing

INTERFACE_CONFIG_C [0x10]			
Bit	**Value Set**	**Bit Name**	**Description**
7:6	00	CRC_ENABLE	CEC disabled.
5	1	STRICT_REGISTER_ACCESS	Read-only bit, not applicable.
4:2	000	RESERVED	
1:0	11	CRC_ENABLE_B	CRC Disabled. Must be complimentary to bits [7:6] / CRC_ENABLE
0	0	SW_RESET_LSB	Do nothing

INTERFACE_CONFIG_D [0x14]			
Bit	**Value Set**	**Bit Name**	**Description**
7	0	RESERVED	
6	0	ALERT_ENABLE_PULLUP	Internal pull up disabled
5	0	RESERVED	
4	0	MEM_CRC_EN	Memory CRC disabled as CRC is not being used
3:2	11	SDIO_DRIVE_STRENGTH	Set to High SDIO drive strength
1	0	Reserved	
0	0	SPI_CONFIG_DDR	DDR mode disabled. Will be changed later

INTERFACE_STREAM_MODE [0x0E]			
Bit	**Value Set**	**Bit Name**	**Description**
7:0	00000010	LENGTH	Set to address two bytes. Maximum of 255 bytes can be addressed

INTERFACE_CONFIG_TX_DSPI[0x0F]			
Bit	**Value Set**	**Bit Name**	**Description**
7:6	01	MULTI_IO_MODE	Set to DUAL SPI for enabling dual SPI write mode. Will be used during data streaming.
5:3	000	RESERVED	
2	1	STREAM_LENGTH_KEEP_VALUE	Stream length to remain same when streaming is completed. Therefore, we need not reconfigure the length for next streaming cycle.
1:0	00	Reserved	

INTERFACE_CONFIG_B_STREAM [0x01]			
Bit	**Value Set**	**Bit Name**	**Description**
7	0	SINGLE_INSTRUCTION	Set to streaming mode.
6:4	000	RESERVED	
2	1	SHORT_INSTRUCTION	Set to 7–bit addressing
2:0	000	Reserved	

INTERFACE_CONFIG_D_DDR [0x14]			
Bit	**Value Set**	**Bit Name**	**Description**
7	0	RESERVED	
6	0	ALERT_ENABLE_PULLUP	Internal pull up disabled
5	0	RESERVED	
4	0	MEM_CRC_EN	Memory CRC disabled as CRC is not being used
3:2	11	SDIO_DRIVE_STRENGTH	Set to High SDIO drive strength
1	0	Reserved	
0	1	SPI_CONFIG_DDR	DDR mode enabled

VREF_INIT_COMMAND [0x15]			
Bit	**Value Set**	**Bit Name**	**Description**
7	0	RESERVED	
6	0	IDUMP_FASDTMODE	Current gain setting meets requirements, so set to 0
5:2	0000	Reserved	
1:0	01	Reserved	Reference voltage generated internally and output on the VREF pin

ERR_ALARM_MASK_CONFIG [0x16]			
Bit	**Value Set**	**Bit Name**	**Description**
7	0	RESERVED	
6	1	REF_RANGE_ALARM_MASK	Reference alarm mask disabled
5	1	CLOCK_COUNT_ERR_ALARM_MASK	Clock Count Error Alarm Mask disabled
4	1	MEM_CRC_ERR_ALARM_MASK	Memory CRC Error Alarm Mask disabled
3	1	SPI_CRC_ERR_ALARM_MASK	SPI CRC Error Alarm Mask disabled
2	1	WRITE_TO_READ_ONLY_ALARM_MASK	Write to Read-Only Alarm Mask disabled
1	1	PARTIAL_REGISTER_ACCESS_ALARM_MASK	Partial Register Access Alarm Mask disabled
0	1	REGISTER_ADDRESS_INVALID_ALARM_MASK	Register Address Invalid Alarm Mask disabled

OUTPUT_RANGE_CONFIG [0x19]			
Bit	**Value Set**	**Bit Name**	**Description**
7:4	0000	RESERVED	
3:0	011	CH0_OUTPUT_RANGE_SEL	Range set between -5V to +5V

HW_LDAC_SETTING_FST [0x28]			
Bit	**Value Set**	**Bit Name**	**Description**
7:1	0000000	RESERVED	
0	0	HW_LDAC_MASK_CH0	LDAC not masked

5.4 Timing Diagrams

5.4.1 Single SPI Write Operation in Classic Mode

This mode is used for setting configuration registers, especially those below 0x1E.

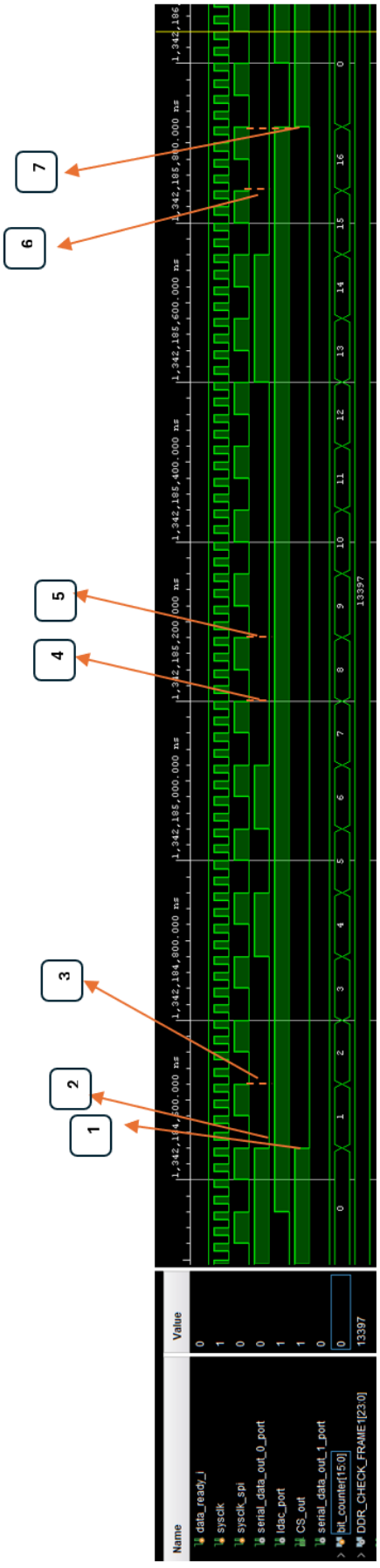

Figure 5.2: Timing Diagram of Single SPI Write in Classic Mode

T. No	Description
1	CS is made low, initiating instruction phase.
2	The first MSB bit is also set. This case is a write operation, so 0 is being sent through serial_data_out_0_port. The bit is set at the negedge of spi_clk giving ample time for data to settle and this is sampled by DAC module at the subsequent posedge. This consideration applies to all bits.
3	First bit of 7-bit Address is being set
4	Last bit of address (LSB)
5	First bit of Data is being set
6	Last bit of Data is being set
7	CS is pulled high. Transaction is complete

5.4.2 Dual SPI Write operation in DDR - Streaming Mode

This mode is used to stream DAC Data. The Write bit and the address is sent only over serial_data_out_0_port while the data is sent over both the ports synchronously. Additionally, the ldac pin should remain high throughout the streaming transaction.

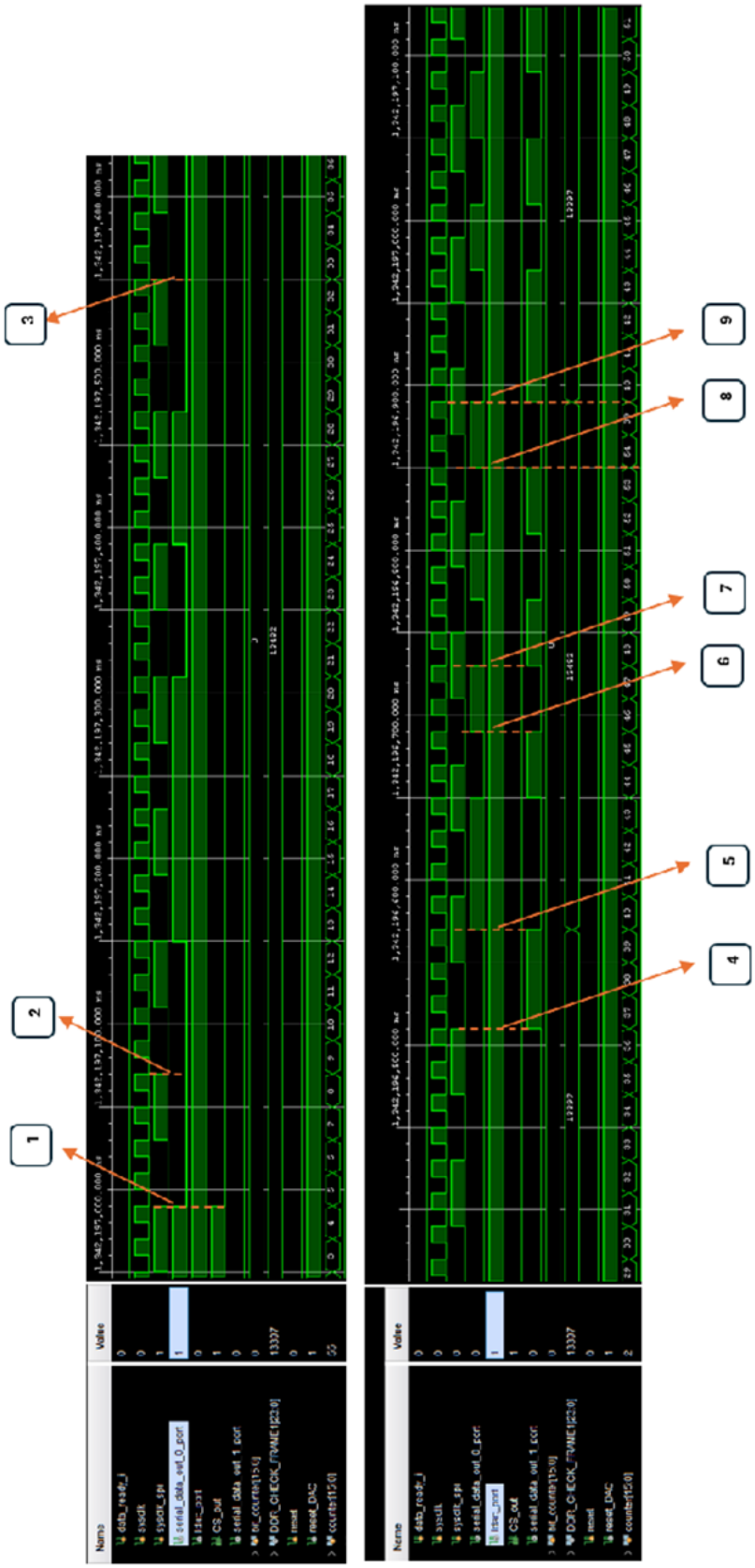

Figure 5.3: Timing Diagram of DUAL DDR SPI Write in Streaming Mode

T. No	Description
1	Start of transaction. CS is pulled down. Write bit is sent over serial_data_out_0_port
2	First bit of 7-bit Address is being set. All address bit are sent serially that aligns with neg-edge of sysclk_spi
3	Last bit of Address
4	First bit of Data is set here
5	Once the fist bit is sampled on pos-edge of sysclk_spi, the next bit(second bit in this case) is set in mid of the clock so that the second bit is sampled at neg-edge of sysclclk_spi.
6-7	All data bits are set in the mid of the sysclclk_spi as shown in these points
8	The last bit is sampled on neg-edge, the next bit is the first bit of data which is set here. Then the counter is looped to 39. The state machines transmit the second MSB bit here and the cycle continues. We do not need to transmit address again, only the data is sent in loop until the transaction is complete or reset.

Chapter 6

Results

This section provides an overview of outcomes post implementation of the DAC system.

6.1 Waveform

We were able to successfully generate waveform such as square and Ramp waveform at both low (10 MHz) and at high (48 MHz) clock frequency.

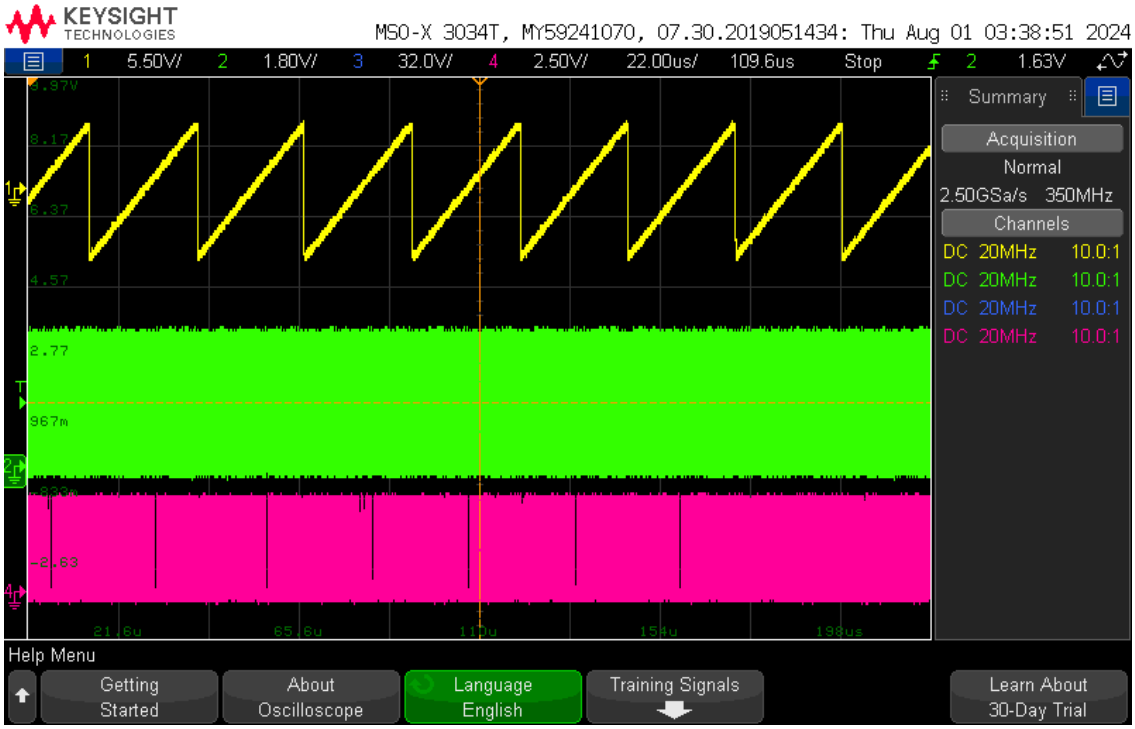

Figure 6.1: Ramp Wave Form Generated using AD3541R DAC at 48 MHz, Yellow is Analog output from DAC, green is the SCLK and pink is the SDIO0 output

6.2 Advantages Gained

1. The serial interface of the DAC significantly reduces the number of pins required. By utilizing the XC7A100T FPGA board, we can consolidate our setup to just one FPGA board instead of four.

2. By transitioning to a serial interface for the DAC, the number of required pins is significantly reduced. This reduction allows for a more compact and efficient design. With fewer components and connections, the overall footprint of the board is minimized.

3. The decrease in the number of components, particularly the FPGAs, leads to a significant reduction in overall costs

6.3 Design Utilization Details

Design Type	**Slice LUTs**	**Slice Registers**	**Dynamic Power (W)**	**Static Power (W)**
With Timing Wizard	117	92	0.103 W	0.072 W
Without Timing Wizard	115	92	1.374 W	0.074 W

Chapter 7

Defnitions

- **ADC:** Analog-to-Digital Converter (ADC) transforms continuous analog signals into discrete digital values by sampling the analog signal at regular intervals.
- **CRC:** Cyclic Redundancy Check (CRC) is an error-detection method used in digital networks and storage devices to identify unwanted changes to raw data or transmitted data. It works by attaching a short check value to data blocks, which is verified upon retrieval to detect any errors.
- **DAC:** Digital-to-Analog Converter (DAC) transforms digital data into continuous analog signals, enabling digital devices to produce electrical signals, RF signals, sound and other analog outputs.
- **DDR:** DDR stands for Double Data Rate. It refers to a type of transmission method widely used in computer memory technology that transfers data on both the rising and falling edges of the clock signal, effectively doubling the data transfer rate compared to traditional single data rate which transmits data only on single edge of the clock.
- **EIT:** Electrical Impedance Tomography (EIT) is a non-invasive imaging technique used to create tomographic images of the body's internal conductivity and impedance. By applying small alternating currents through surface electrodes and measuring the resulting voltages, EIT can infer the electrical properties of tissues, which vary among different biological tissues. This method is particularly useful in medical applications, such as monitoring lung function and detecting cardiac abnormalities.
- **FPGA:** Field-Programmable Gate Array (FPGA) is a semiconductor device that can be programmed and reprogrammed after manufacturing to perform various digital logic functions. It consists of programmable logic blocks and interconnects, allowing users to configure it for specific tasks like signal processing or data encryption.
- **IP:** Intellectual Property (IP) refers to pre-designed and pre-verified blocks of logic that can be integrated into FPGA designs to perform specific functions. Vivado provides an IP catalog that includes a wide range of IP cores, such as memory controllers, communication interfaces, and signal processing blocks. Users can also create custom IP blocks using the IP Packager and integrate them into their designs.
- **LSB:** LSB stands for Least Significant Bit. In digital systems, it refers to the bit in a binary number that has the smallest value or weight. For example, in the binary number 1010, the LSB is the rightmost bit, which is 0.

- **LUT:** Look-Up Tables (LUTs) are tiny memory units with FPGA device that store truth tables for specific logic functions. When an input signal is applied to the LUT, it uses the stored truth table to determine the output signal. This allows FPGAs to implement complex combinational logic efficiently by performing table lookups instead of traditional gate-based logic.

- **MSB:** MSB stands for Most Significant Bit. In digital systems, it refers to the bit in a binary number that has the highest value or weight. For example, in the binary number 1010, the MSB is the leftmost bit, which is 1.

- **MUPS:** Mega Updates per Second (MUPS) is a term used to describe the rate at which updates or changes are processed in a system, typically in the context of updating data of a wave's digital form . It measures how many million updates are performed each second.

- **Negedge:** Negedge refers to the negative edge or falling edge of a clock signal in digital circuits. It occurs when the clock signal transitions from a high state (1) to a low state (0).

- **Posedge:** Posedge refers to the positive edge or rising edge of a clock signal in digital circuits. It occurs when the clock signal transitions from a low state (0) to a high state (1).

- **SPI:** Serial Peripheral Interface (SPI) is a synchronous serial communication protocol used to transfer data between microcontrollers and peripheral devices like sensors, displays, and memory chips. It operates in full-duplex mode, meaning data can be sent and received simultaneously, using four main lines: SDI/SDIO0 (SPI Data Input/ SPI Data Input Output 0), SDO/SDIO1 (SPI Data Output/ SPI Data Input Output 1), SCK (Serial Clock), and CS (Chip Select).

- **Dual SPI Write:** Dual SPI write refers to a mode in which data is written to a device using two data lines simultaneously, effectively doubling the data transfer rate compared to standard SPI. This mode is particularly useful for high-speed data transfer applications.

- **Streaming Mode:** Data is streamed continuously without the need to re-transmit the address per transaction. This requires fewer clock cycles to write the data making the write operation more efficient.

Acknowledgments

Sincere gratitude to **Dr. Omid Rajabi Shishvan** for his invaluable guidance and support throughout this project. His expertise was instrumental in overcoming challenges and ensuring the success of this project. His willingness to share knowledge and provide assistance at every step is greatly appreciated

References

[1] Omid Rajabi Shishvan, Ahmed Abdelwahab, Nilton Barbosa da Rosa, Gary J Saulnier, Jennifer L Mueller, Jonathan C Newell, and David Isaacson. ACT5 electrical impedance tomography system. IEEE Transactions on Biomedical Engineering, 2023.

[2] Analog Devices. Analog Devices AD3541R, Single Channel, 12-/16-Bit, 16 MUPS, Multispan, Multi-IO SPI DAC. Available at `https://www.analog.com/media/en/technical-documentation/data-sheets/ad3541r.pdf`.

[3] Digilent. Digilent CMOD A7 Reference Manual. Available at `https://digilent.com/reference/_media/cmod_a7/cmod_a7_rm.pdf`.

[4] Mikroe. DAC 13 Eval Board Reference manual. Available at `https://www.mikroe.com/brochure?ref=MIKROE-5412`.