High-performance test bench for the automation of DAPHNE board minimum performance tests

Diego Restrepo (PhD in Physics)

Jaime Osorio (PhD in Physics)

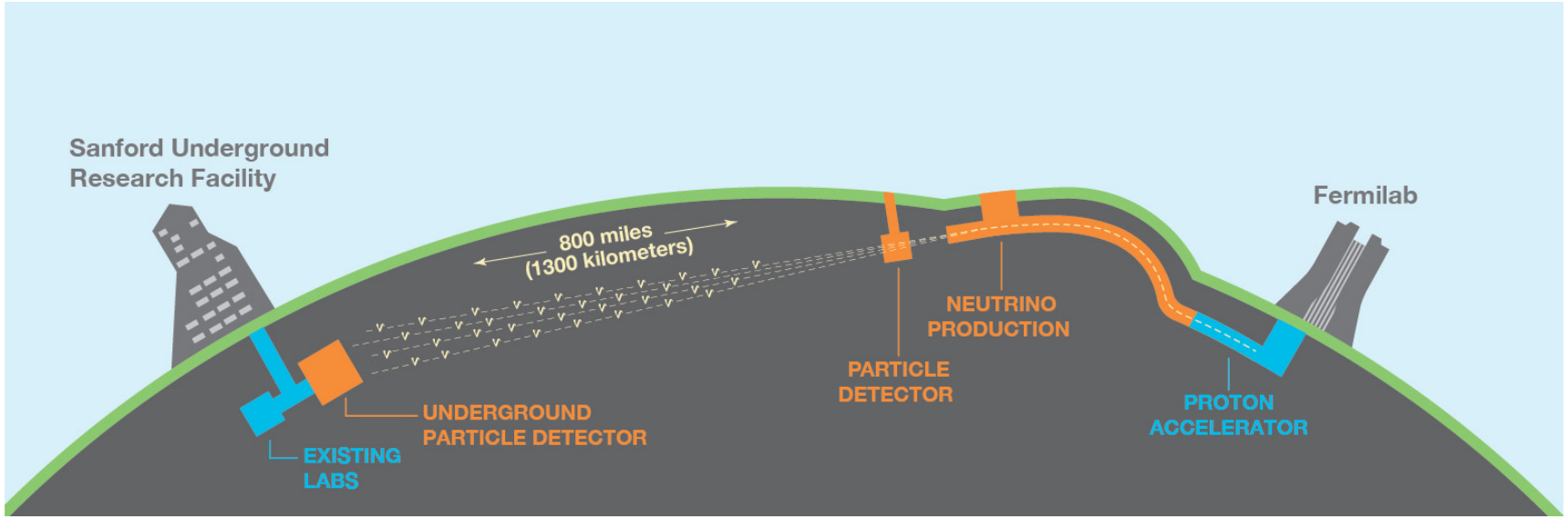
Oscar Zapata (PhD in Physics)

Jorge Lopez (PhD in Physics)

Fabian Castaño (PhD in Electronics Engineering and Computer Science)

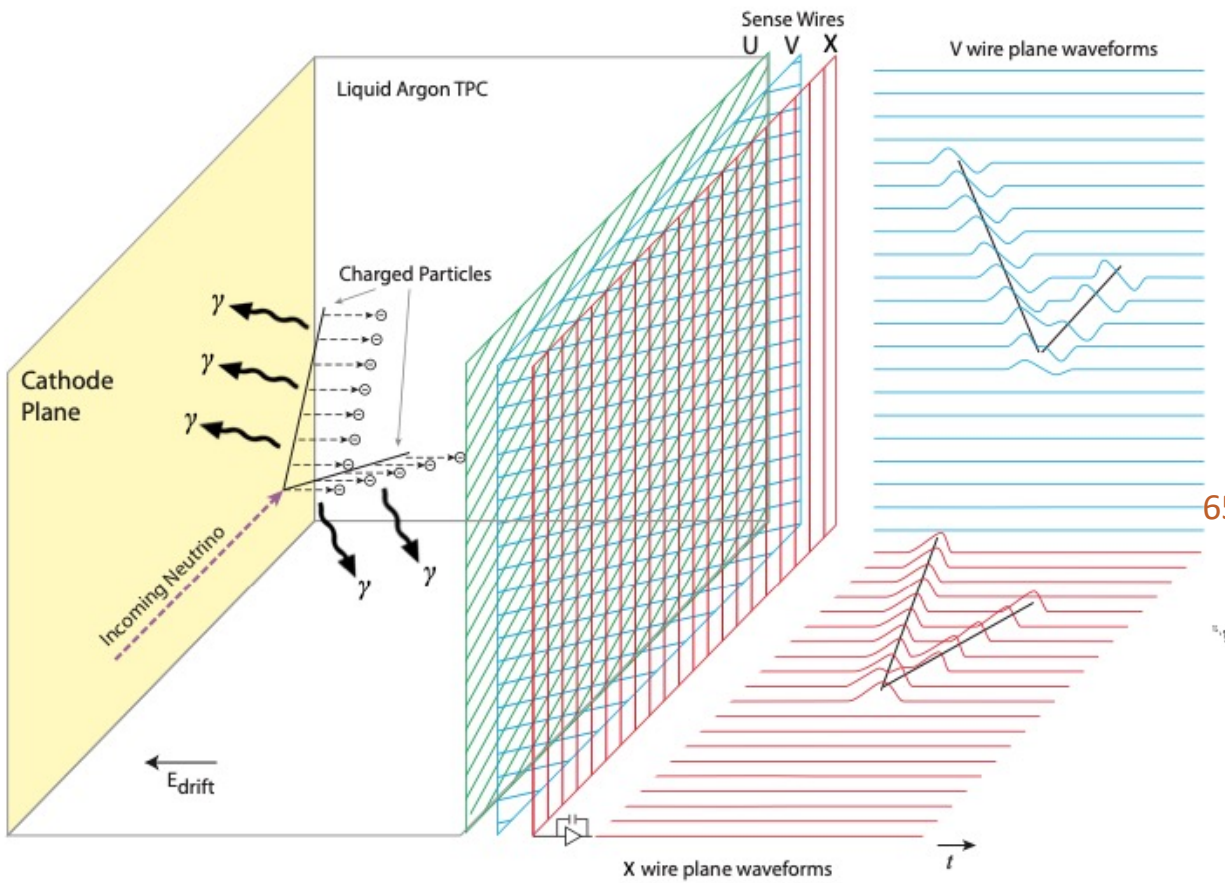
Universidad de Antioquia (DUNE collaboration member since June 2022)

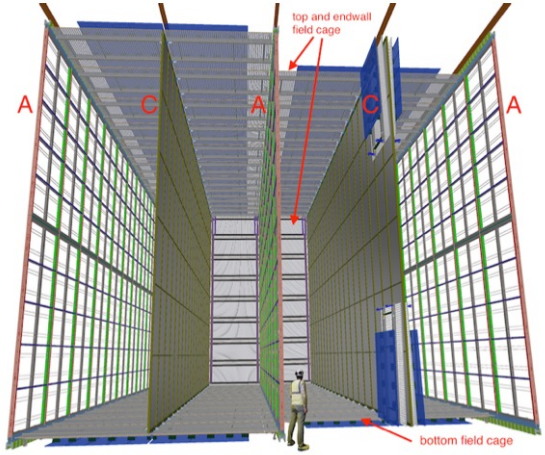
What is DUNE?



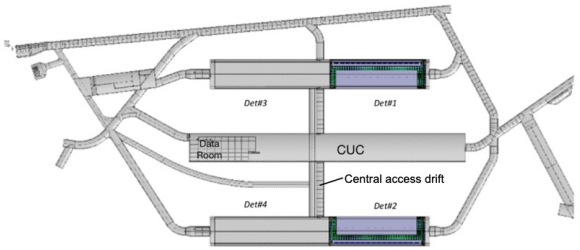
- Investigate Neutrino oscillation to prove charge parity violation (CP)
- Determine the order of the mass of the neutrinos
- Study supernovae, the formation of neutron stars and black holes

Far Detector (FD)





65.8 m (L) by 18.9 m (W) by 17.8 m (H)



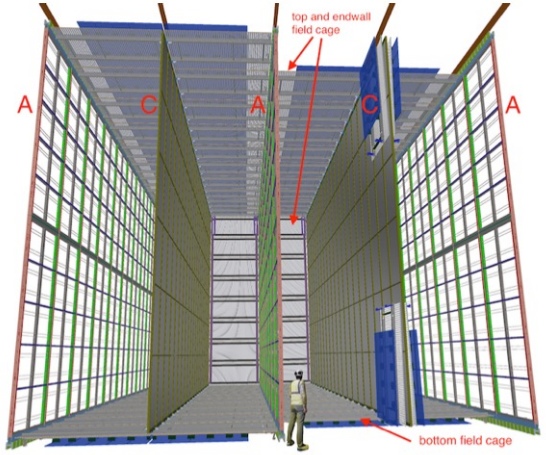
B. Abi, R. Acciarri, M. Acero, G. Adamov, D. Adams, M. Adinolfi et al., Volume IV. the DUNE far detector single-phase technology, Journal of Instrumentation 15 (aug, 2020) T08010–T08010.

Horizontal and Vertical Drift

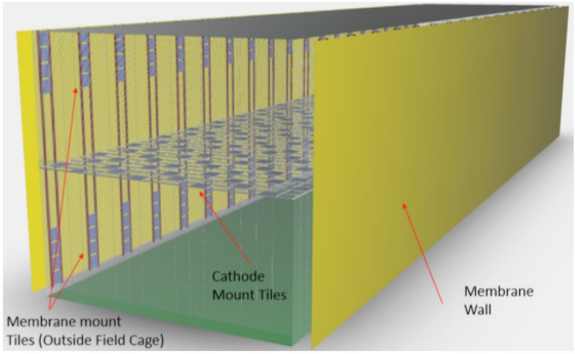
- ProtoDUNE-2

  Prototype Experiment built in CERN (Geneva, Switzerland)

HD

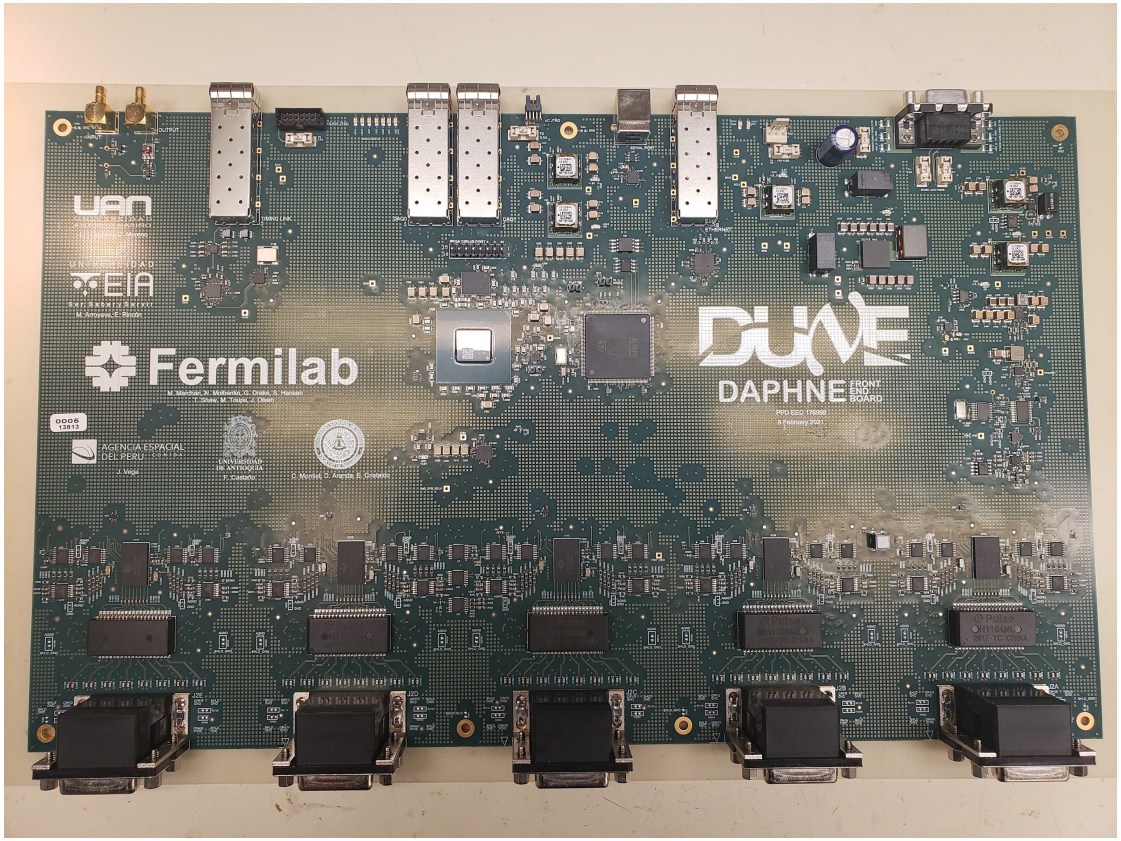


VD

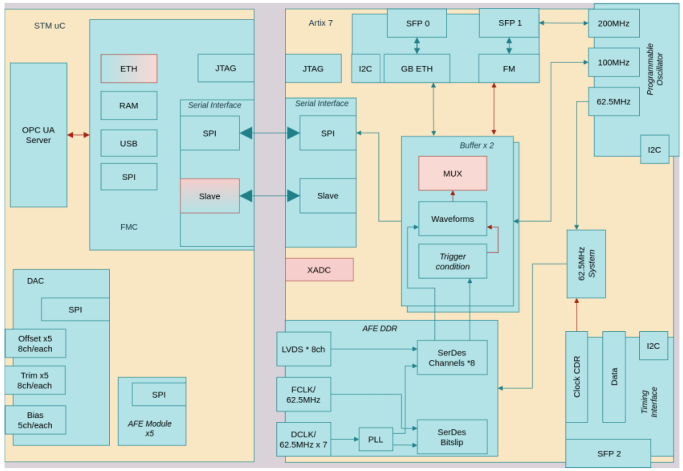


DAPHNE V1 and V2A

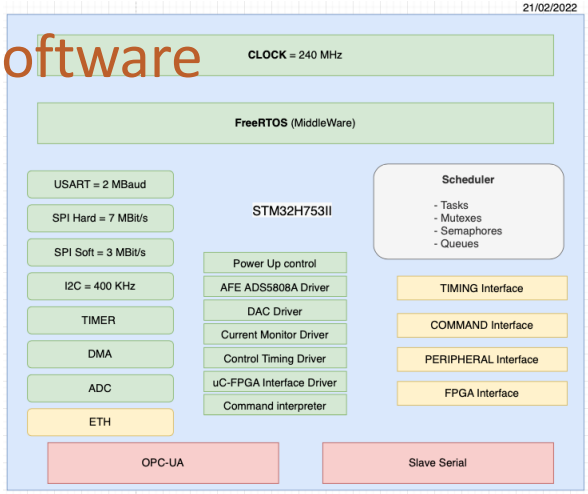
Hardware



Firmware



Software

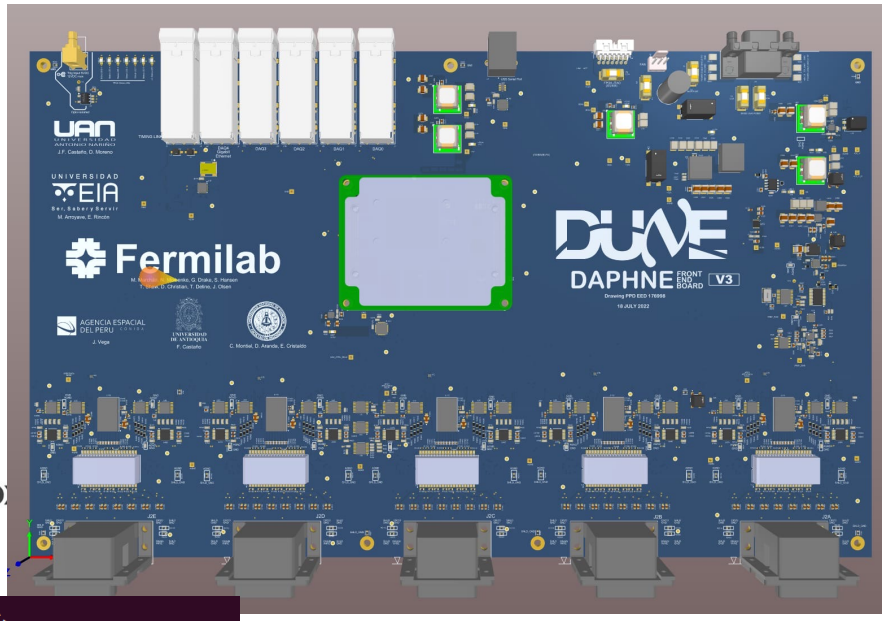


DAPHNE V3

- Kria KR260 – DAPHNE V3

  Design of new architecture for Data Acquisition System

  - Acquisition of development boards for the collaboration



VirtualBox

ubuntu

VIVADO™

XILINX VITIS™

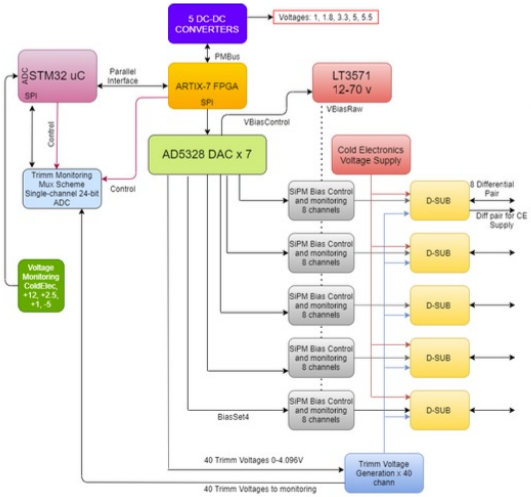
GitHub

https://github.com/fabioc9675/KRIA_Starter_Guide/blob/documentation/Tutorial/T01_Kria_and_Vivado.md

CHALLENGES (Operation and Diagnosis)

- It is necessary to implement an automatic test.
- Manual test:
  - A lot of time
  - Reprocessing
  - Non-standardization
  - Possible failures
- Automatic test:
  - Better performance
  - Multiple evaluation simultaneously
  - Standardization
  - Traceability.

VOLTAGE GENERATION AND MONITORING

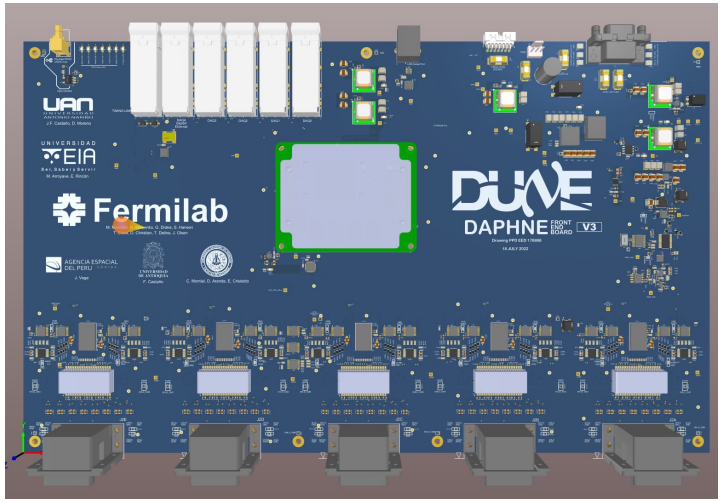


Voltage/Parameter	Bias Raw	SiPM Bias	Trimm
Range (V)	12-70	0 V-BiasRaw	0-4.096
Generated by	DC-DC Boost Converter	Custom Circuit	DAC+custom circuit
Adjusted by	FPGA/12 bits DAC	FPGA/12 bits DAC	FPGA/12 bits DAC
Adjustment Precision	14,16 mV (for BiasRaw=70 V)	14,16 mV	1 mV
Generated for	General	8 channels, 5 groups	Individual channel

Also, can be used to do diagnosis and evaluation of performance in real time

Institutional Goals

- To define of minimum operating requirements and testing protocol.
- To design and manufacturing an electronic test bench board system.
- To evaluate and validating the test bench performance.
- To evaluate the performance of a DAPHNE board using the test bench.



Use the test bench infrastructure to simulate hardware event signals and assess the performance of the DAPHNE board.

Proposed technical tests

- **Operating voltage test:** Verify the board operation under specified voltages.
- **Board impedance test:** Evaluate signal integrity, power distribution and high-frequency performance.
- **Digitizing systems (AFEs) test:** Validate the AFE's performance, linearity, noise levels, dynamic range, and overall fidelity in digitizing signals from various sources.
- **PL operation (Fast DAQ) test:** Timing accuracy, synchronization capabilities, data integrity, and overall system throughput.
- **Signal conditioning test:** Gain, frequency response, linearity, noise levels, distortion, and impedance matching.
- **PS operation (Slow control) test:** Reliability, responsiveness, and accuracy in performing slow control tasks, such as adjusting settings, configuring parameters, or managing system states.

High-performance test bench for the automation of DAPHNE board minimum performance tests

Diego Restrepo (PhD in Physics)

Jaime Osorio (PhD in Physics)

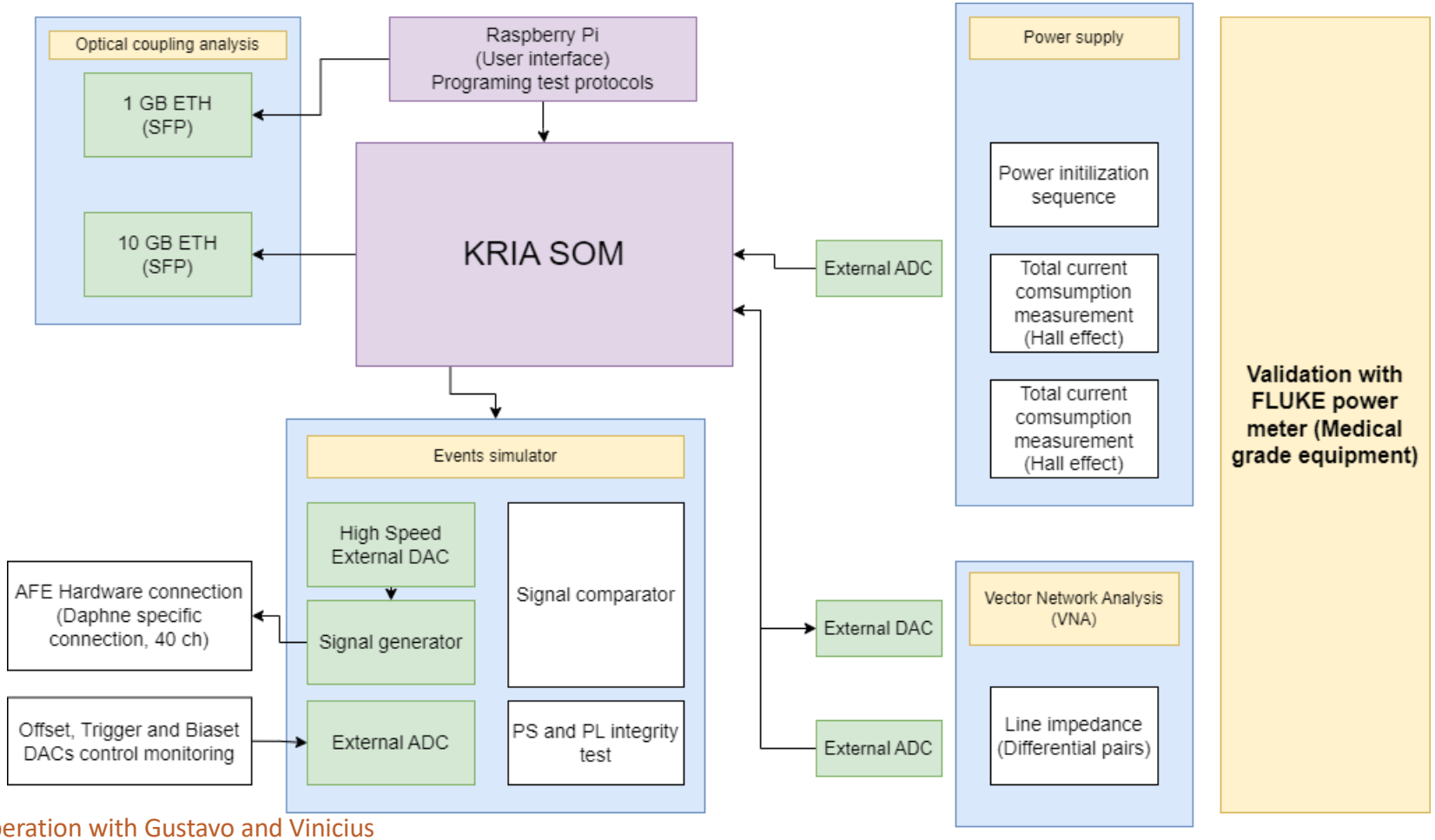
Oscar Zapata (PhD in Physics)

Jorge Lopez (PhD in Physics)

Fabian Castaño (PhD in Electronics Engineering and Computer Science)

Universidad de Antioquia (DUNE collaboration member since June 2022)

Test bench architecture



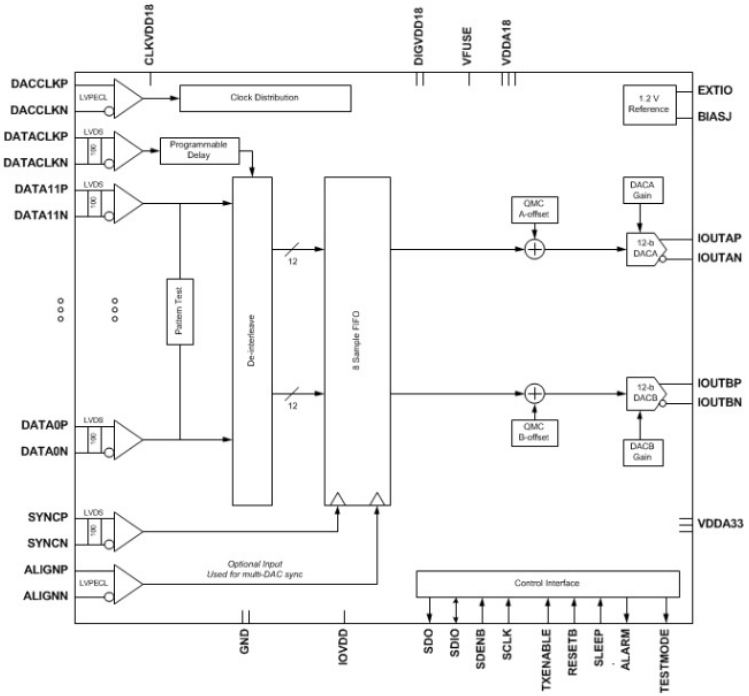
Cooperation with Gustavo and Vinicius

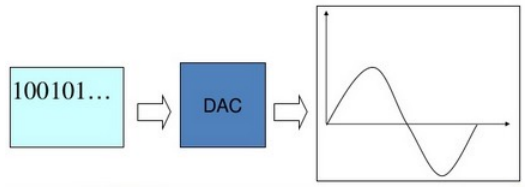
Test bench as event signal simulator

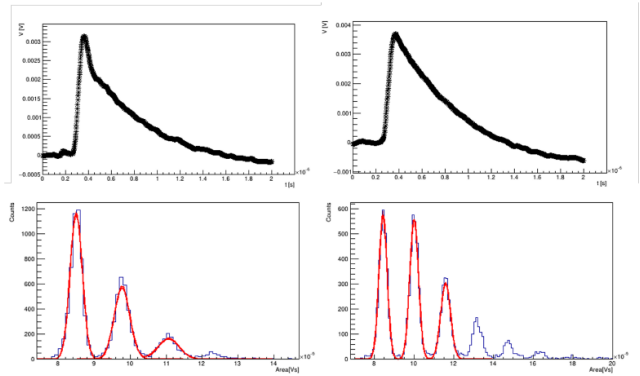
DAC3154 ACTIVE

https://www.ti.com/product/DAC3154

Dual-channel, 10-bit, 500-MSPS digital-to-analog converter (DAC) with input FIFO and current sour





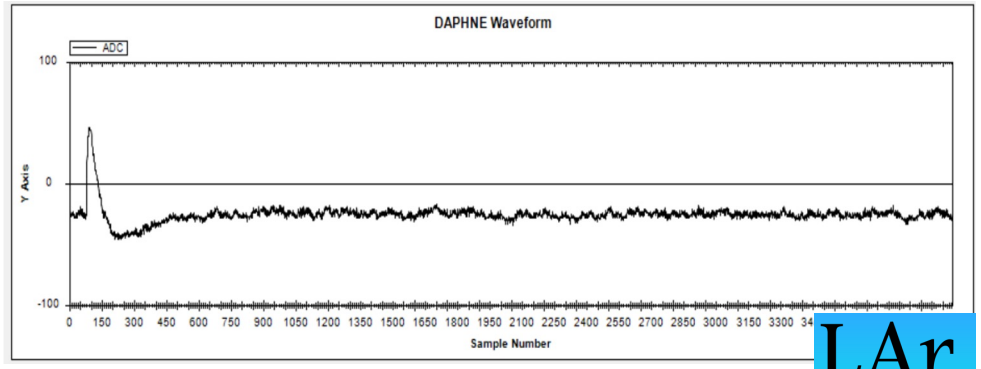


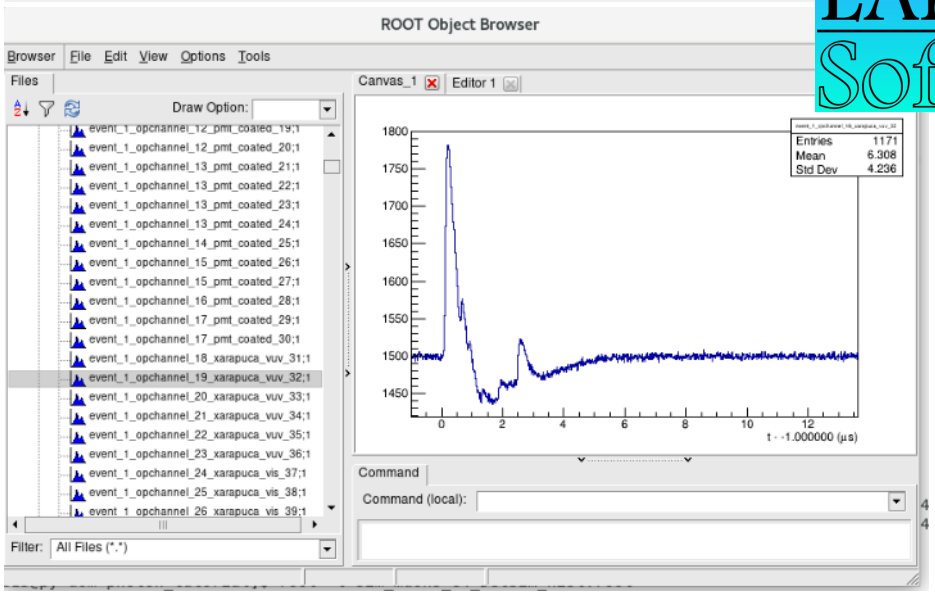
LoopBack with AFE5808A

PoS(NuFact2021)191

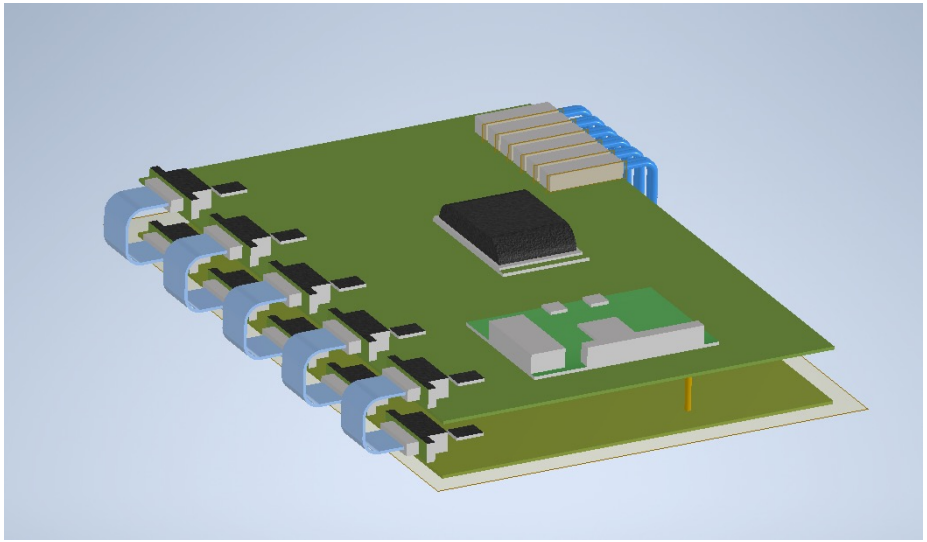
Test bench as event signal simulator

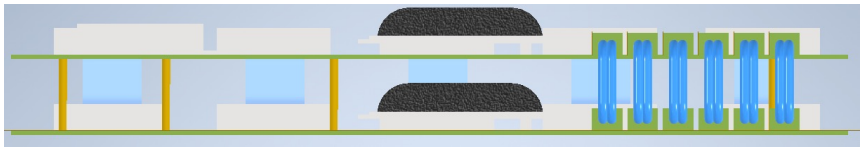
- Generation of simulated events based on physics.
- Measurement of the performance in readout system.
- Diagnosis of electronics' system.

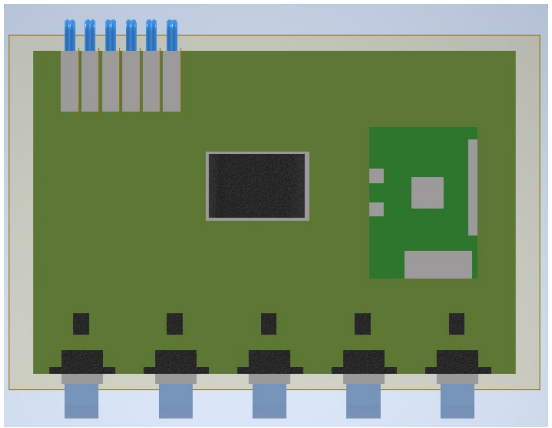


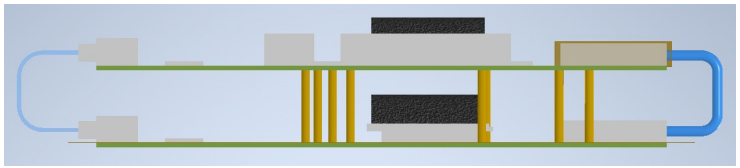


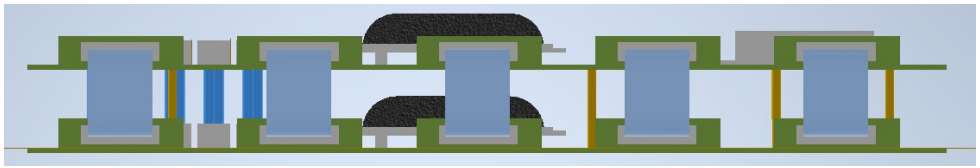
Concept design









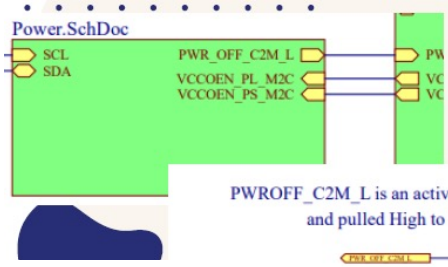


Update March 2024

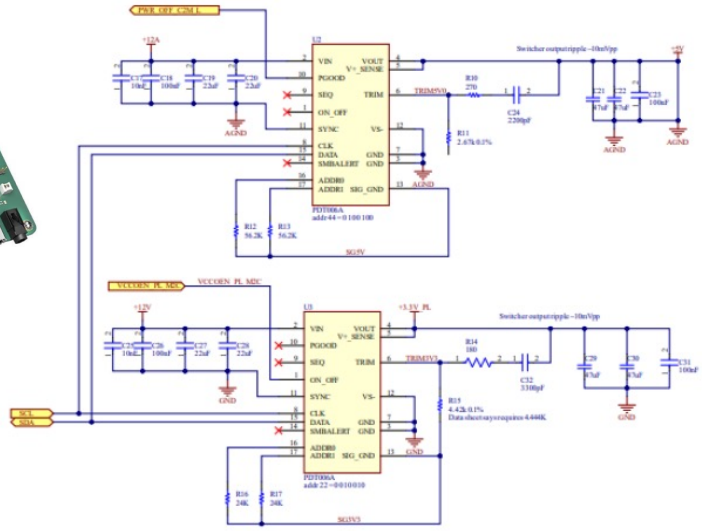
DAC High Speed new analysis

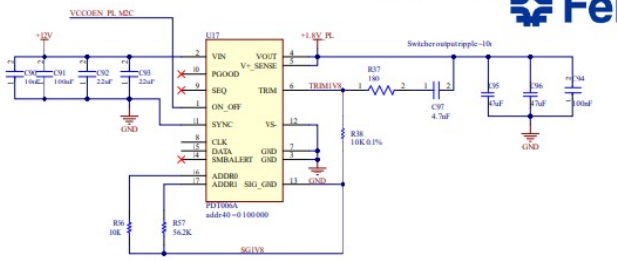
- DAC53508 Octal, 10-bit, buffered-voltage-output DAC with SPI (https://www.ti.com/lit/ds/symlink/dac53508.pdf?ts=1707117365109)
- AFE7950 4-transmit, 6-receive RF-sampling transceiver, 600-MHz to 12-GHz, max 1200-MHz IBW (https://www.ti.com/lit/ds/symlink/afe7950.pdf?ts=1707153096602&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FAFE7950)
- AFE7900 4T6R RF Sampling AFE with 12 GSPS DACs and 3 GSPS ADCs (https://www.ti.com/lit/ds/symlink/afe7900.pdf?ts=1707154656670&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FAFE7900)
- AD9088 Apollo MxFE Octal, 16-Bit, 16 GSPS RF DAC and Octal, 12-Bit, 8 GSPS RF ADC (https://www.analog.com/en/products/ad9088.html)
- MAX5869 16-Bit, 5.9Gsps Interpolating and Modulating RF DAC with JESD204B Interface (https://www.analog.com/media/en/technical-documentation/data-sheets/MAX5869.pdf)
- AD9164 16-Bit, 12 GSPS, RF DAC and Direct Digital Synthesizer (https://www.analog.com/media/en/technical-documentation/data-sheets/AD9164.pdf)

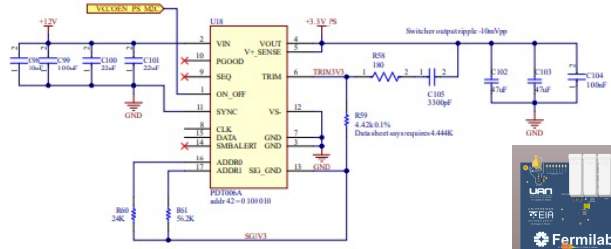
KR260 Carrier Board Explorer

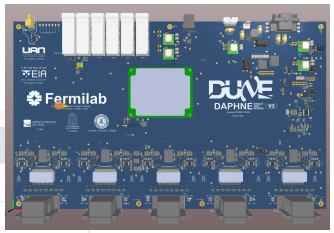


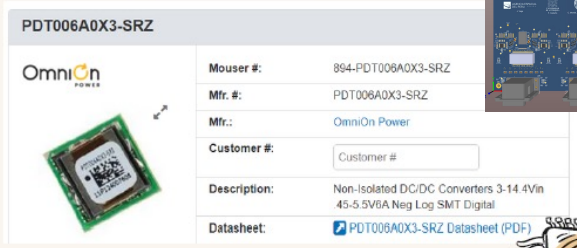
PWROFF_C2M_L is an active-Low signal to power down the SOM and pulled High to the +5V SOM input power rail.

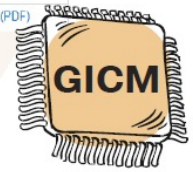












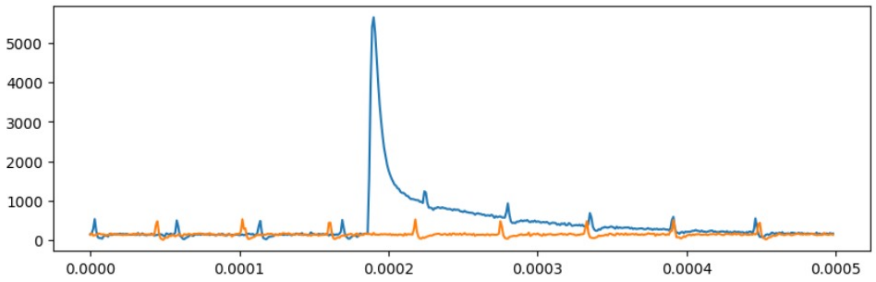
Power sequencing required?
If possible, use regulators with I2C (PMBus) interface and tie it to the Kria
Maybe OK to use LDO to make +1.8V from the +3.3V rail
Do we need any other power rails? +2.5V?

Simulation model building

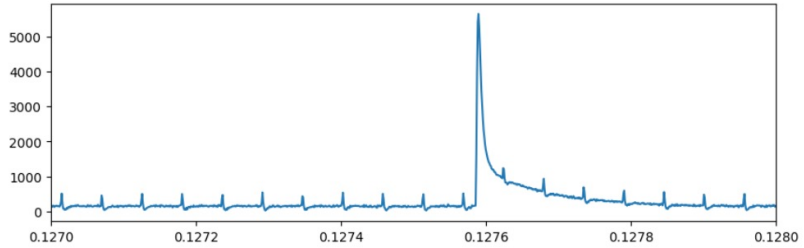
- Biexponential model (Radiation events' model)

$$y(t) = A(e^{(t-t_0)/\tau_D} - e^{(t-t_0)/\tau_R})$$

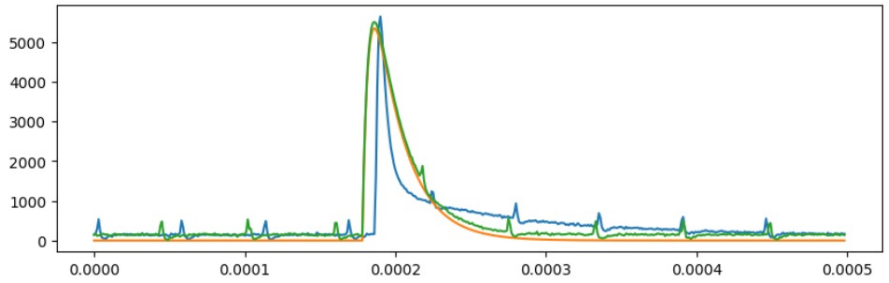
- Baseline noise extraction



- Radiation Event sampled (Geiger Detector)

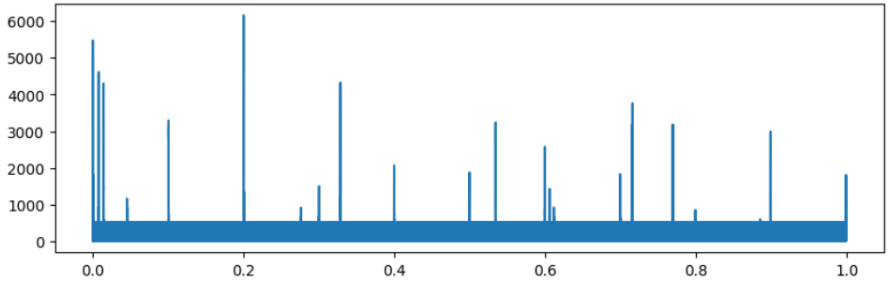


- Event model fitting



Simulation model building

- Multiple event simulation



- Design of Event counter

- Trapezoidal Shaper Filter and DPP implementation

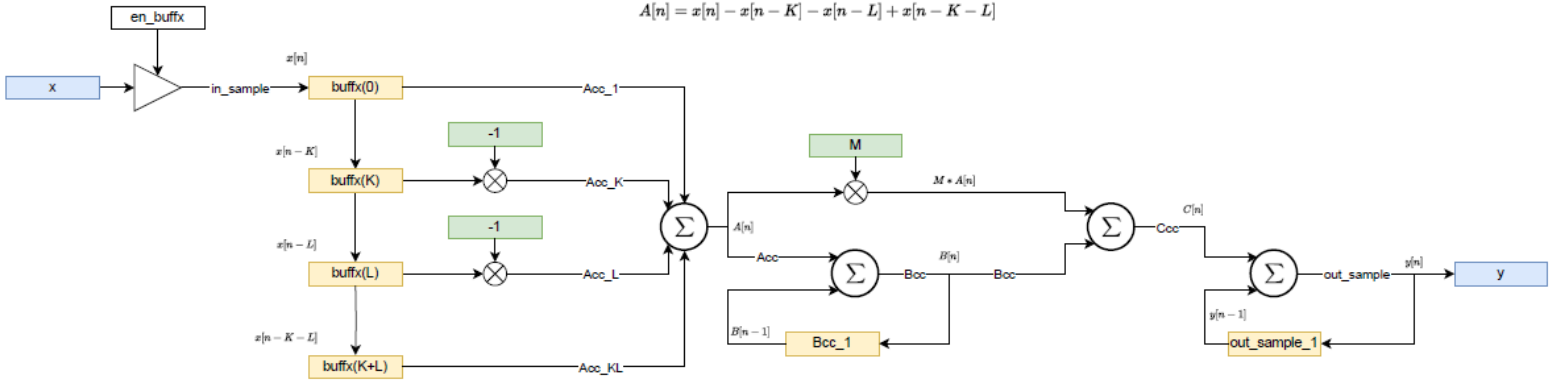
Trapezoidal Shaper

$$y[n] = y[n-1] + C[n]$$

$$C[n] = B[n] + M * A[n]$$

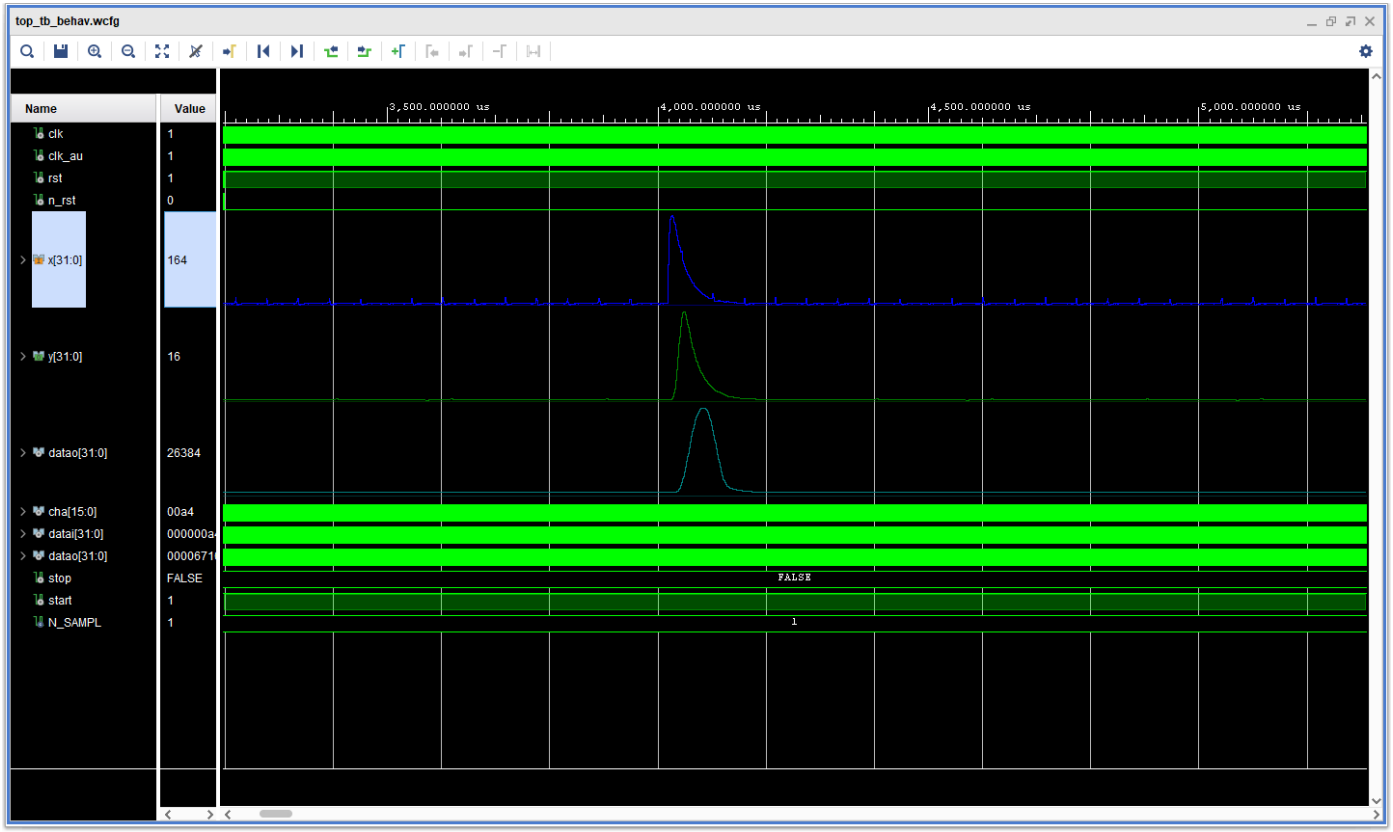
$$B[n] = B[n-1] + A[n]$$

$$A[n] = x[n] - x[n-K] - x[n-L] + x[n-K-L]$$



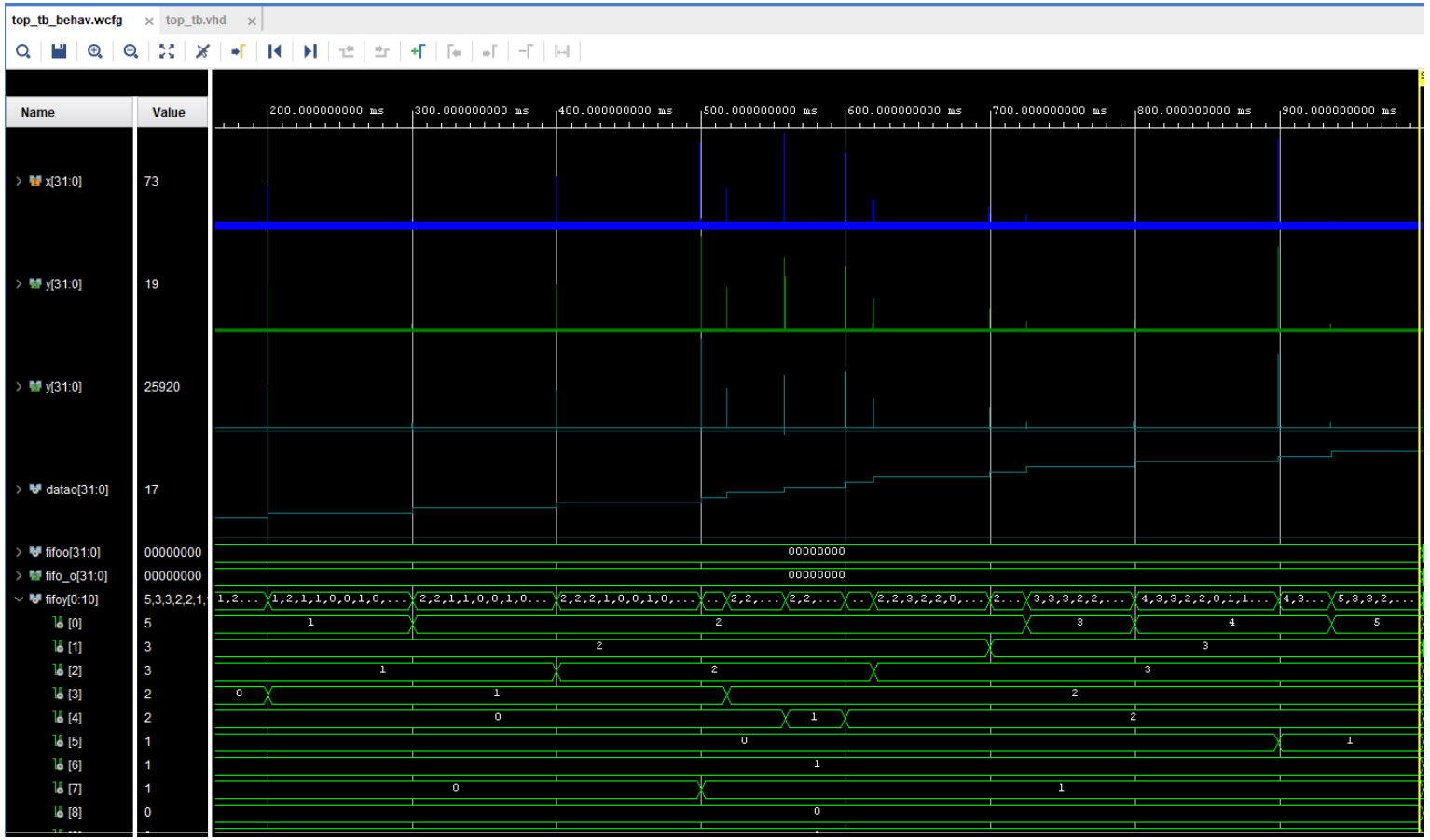
Simulation model building

- Multiple event simulation



Simulation model building

- Multiple event simulation



Next Steps

- Finalization of component evaluation
- Present a first version of the Schematic definition and design
- Generate a difference equation of the simulation model and implementation in Hardware

High-performance test bench for the automation of DAPHNE board minimum performance tests

Diego Restrepo (PhD in Physics)
Jaime Osorio (PhD in Physics)
Oscar Zapata (PhD in Physics)
Jorge Lopez (PhD in Physics)
Fabian Castaño (PhD in Electronics Engineering and Computer Science)

Valentina Restrepo (Electronics Student)
Valentina Rodriguez (Physics Student)
Jeronimo Lopez (Physics Student)
Diego Salazar (Master Student)

Universidad de Antioquia (DUNE collaboration member since June 2022)

Update August 2024

Components support

ELECTRÓNICA I+D S.A.S

FACTURA

27 de mayo de 2024

FP 50054

Forma de pago: Transferencia

Dirección: Sede Principal
Calle 48D N° 65A - 35 / Suramericana
Ciudad: Medellín - Colombia
Nit: 900.034.424-0 Régimen Común
Sitio Web: www.didacticaselectronicas.com
Teléfono: 57(4) + 322 50 71 - 3005006693
Asesor de venta: Juan Camilo Rendón

CLIENTE

Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional
Av. Instituto Politécnico Nacional 2508, Col. San Pedro Zacatenco
Ciudad de México, México, C.P. 07360
RFC. CIE601028IU2
Karlas.tenorio@cinvestav.mx
55 5747 3839

Item	Cant.	Uni.	Descripción	Referencia	Precio Unitario	TOTAL
1	1	Und	ANALOG DISCOVERY 3 PRO BUNDLE	471-060	$ 2.519.400,00	$ 2.519.400,00
2	1	Und	EVAL BOARD FOR DAC3154	DAC3154EVM	$ 3.245.900,00	$ 3.245.900,00
3	1	Und	IC MOD SOM K26C ZYNQ MPSOC	SM-K26-XCL2GC	$ 2.602.600,00	$ 2.602.600,00
4	1	Und	CONN HEADER R/A 14POS 2MM	878331420	$ 11.800,00	$ 11.800,00
5	4	Und	CONN HDR 12POS 0.1 GOLD PCB R/A	PPPC062LJBN-RC	$ 7.800,00	$ 31.200,00
6	1	Und	IC REG BUCK ADJ/6.5V 9A 85LGA	PI3546-00-LGIZ	$ 128.800,00	$ 128.800,00
7	4	Und	DC DC CONVERTER 0.5-5.5V	PDT006A0X3-SRZ	$ 75.400,00	$ 301.600,00
8	1	Und	IC REG LIN POS ADJ 500MA 8SOIC	ADP7105ARDZ-R7	$ 40.100,00	$ 40.100,00
9	1	Und	IC BUS SWITCH 2 X 1:4 16TSSOP	TCA9546APWR	$ 8.300,00	$ 8.300,00
10	1	Und	IC TRANSLATOR BIDIRECTIONAL SM8	PCA9306DCTR	$ 3.900,00	$ 3.900,00
11	1	Und	SENSOR DIGITAL -40C-125C 8MSOP	MCP9808T-E/MS	$ 8.600,00	$ 8.600,00
12	1	Und	IC ADC 12BIT SAR 16WQFN	ADS7138QRTERQ1	$ 52.800,00	$ 52.800,00
13	1	Und	IC CLK GEN I2C QUAD 24QFN	SI5338A-B-GMR	$ 150.500,00	$ 150.500,00
14	1	Und	XTAL OSC XO 25.0000MHZ LVCMOS	CW0813-025.0M	$ 26.700,00	$ 26.700,00
15	1	Und	IC TRANSLTR BIDIRECTIONAL SC70-6	SN74AXC1T45DCKR	$ 3.500,00	$ 3.500,00
16	1	Und	IC USB TO UART BRIDGE QFN24	CP2102N-A02-GQFN24	$ 27.300,00	$ 27.300,00
17	1	Und	FILTER RC(PI) 22 OHM/68PF SMD	STF202-22T1G	$ 3.300,00	$ 3.300,00
18	1	Und	IC DAC 12BIT V-OUT 16TSSOP	AD5327BRUZ	$ 148.000,00	$ 148.000,00
19	1	Und	IC VREF SERIES 0.04% 8SOIC	ADR444BRZ	$ 70.600,00	$ 70.600,00
20	1	Und	IC CMOS 2 CIRCUIT 8SON	OPA2320AIDRGT	$ 23.900,00	$ 23.900,00
21	1	Und	CONN D-SUB RCPT 9POS R/A SLDR	DD9S33E4GV00LF	$ 11.600,00	$ 11.600,00
22	1	Und	IC MOD SOM K26C ZYNQ MPSOC	SM-K26-XCL2GC	$ 2.602.600,00	$ 2.602.600,00
23	2	Und	IC HUB CTLR USB 56VQFN	USB5744T-I/2G	$ 49.000,00	$ 98.000,00
24	2	Und	IC TRANSCEIVER HALF 1/1 32QFN	USB3320C-EZK-TR	$ 13.300,00	$ 26.600,00
25	4	Und	TVS DIODE 5.5VWM 14VC 10USON	TPD4E05U06DQAR	$ 4.300,00	$ 17.200,00
26	5	Und	TVS DIODE 5.5VWM 6SON	TPD4S012DRYR	$ 3.700,00	$ 18.500,00
27	1	Und	IC MEDIA CTRLR USB 2.0 26 QFN63	USB2244-AEZG-06	$ 15.400,00	$ 15.400,00
28	1	Und	TVS DIODE 5.5VWM 8-UQFN	TPD6E004RSER	$ 3.700,00	$ 3.700,00
29	1	Und	CONN MICRO SD CARD PUSH-PUSH R/A	5025700893	$ 27.200,00	$ 27.200,00
30	4	Und	IC ETHERNET PHY 48VQFN	DP83867CSRGZT	$ 47.900,00	$ 191.600,00
31	4	Und	CONN JACK 4PORT 1000 BASE-T PCB	0879-2C2R-54	$ 157.500,00	$ 630.000,00
32	2	Und	CONN RCPT DISPLAYPORT 20P SMD RA	47272000I	$ 37.500,00	$ 75.000,00
33	1	Und	TVS DIODE 5.5VWM 15SOH	TPD8S009DSHR	$ 9.700,00	$ 9.700,00
34	5	Und	IC TRANSLTR BIDIRECTIONAL 10UQFN	SN74AVC2T245RSWR	$ 6.400,00	$ 32.000,00
35	1	Und	IC PWR SWITCH N-CHAN 1:1 SOT23-5	STMPS2151STR	$ 8.400,00	$ 8.400,00
36	1	Und	IC CLK BUFFER 1:4 250MHZ 8TSSOP	NB3V1104CDTR2G	$ 19.700,00	$ 19.700,00
37	1	Und	IC BUS SWITCH 2 X 1:4 16SOIC	TCA9546ADR	$ 10.200,00	$ 10.200,00
38	3	Und	IC EEPROM 64KBIT I2C 1MHZ 8TSSOP	M24C64-DRDW8TP/K	$ 3.200,00	$ 9.600,00
39	1	Und	CONN SFP+ RCPT 20POS SLD R/A SMD	744410001	$ 18.100,00	$ 18.100,00
40	1	Und	CONN RCPT 60POS SMD GOLD	DF40HC(4.0)-60DS-0.4V(51)	$ 11.200,00	$ 11.200,00
41	3	Und	IC PWR SWITCH N-CHAN 1:1 SOT23-5	STMPS2161STR	$ 8.300,00	$ 24.900,00
42	1	Und	IC USB HS QUAD UART/SYNC 64-LQFP	FT4232HL-TRAY	$ 39.100,00	$ 39.100,00
43	1	Und	IC EEPROM 2KBIT MICROWIRE 8DIP	93LC56B-I/P	$ 2.600,00	$ 2.600,00
44	1	Und	CONN RCPT USB2.0 MICRO AB SMD RA	ZX62D-AB-5P8(30)	$ 6.100,00	$ 6.100,00
45	6	Und	IC REG LINEAR POS ADJ 500MA 8QFN	MP8904DD-LF-P	$ 12.300,00	$ 73.800,00
46	1	Und	IC OSC WATCHDOG 10VSON	TPS3430WDRCR	$ 12.300,00	$ 12.300,00
47	8	Und	CONN 0.635MM HDR 60POS SMD	ADM6-60-01.5-L-4-2-A-TR	$ 130.500,00	$ 1.044.000,00
48	4	Und	ROUND SPACER STEEL 5MM	9774050960R	$ 7.300,00	$ 29.200,00
49	6	Und	IC REG LINEAR POS ADJ 500MA 8QFN	MP8904DD-LF-P	$ 12.300,00	$ 73.800,00
50	2	Und	IC CURRENT MONITOR 0.15% 16TSSOP	INA260AIPW	$ 39.300,00	$ 78.600,00
51	1	Und	IC PWR LMT SWITCH N-CH 1:1 10QFN	MP5016HGQH-Z	$ 9.800,00	$ 9.800,00
52	1	Und	Comision por transferencia	N/R	$ 120.000,00	$ 120.000,00
					Subtotal:	$ 14.759.300,00
					Descuento:	$ 0,00
					Subtotal - Desc	$ 14.759.300,00
					Flete Nacional:	$ 0,00
					IVA19%	$ 2.804.267,00
					Total:	$ 17.563.567,00

27 de Mayo de 2024
Se recibe material a conformidad y completo

Recibe: Fabian Andres Castaño Usuga
1038404651
Universidad de Antioquia

Cinvestav

UAN UNIVERSIDAD ANTONIO NARIÑO

UNIVERSIDAD EIA Ser, Saber y Servir

UNIVERSIDAD DE ANTIOQUIA 1803

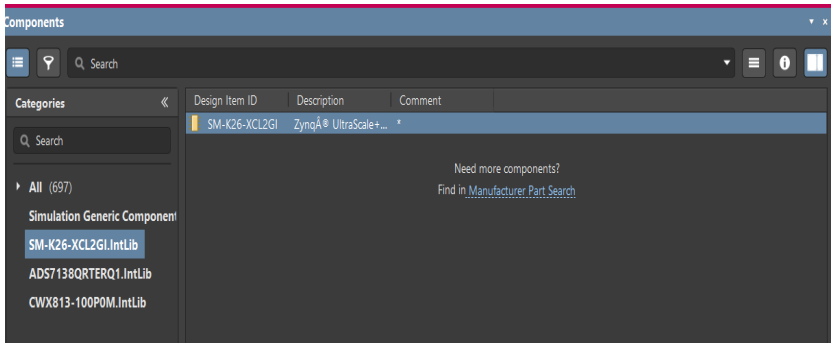
Cinvestav

DUNE

schematics design

- Libraries design

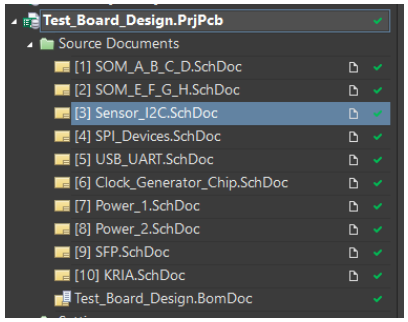
Serching for the libraries of non-generic componentes such as the SOM and converters like DAC and ADC.

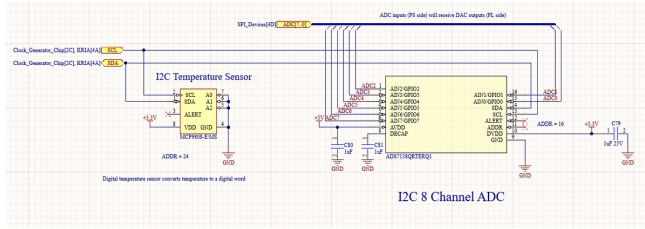


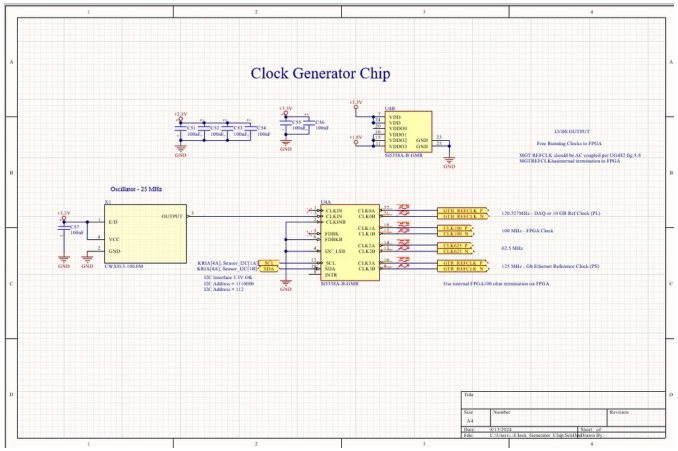
schematics design

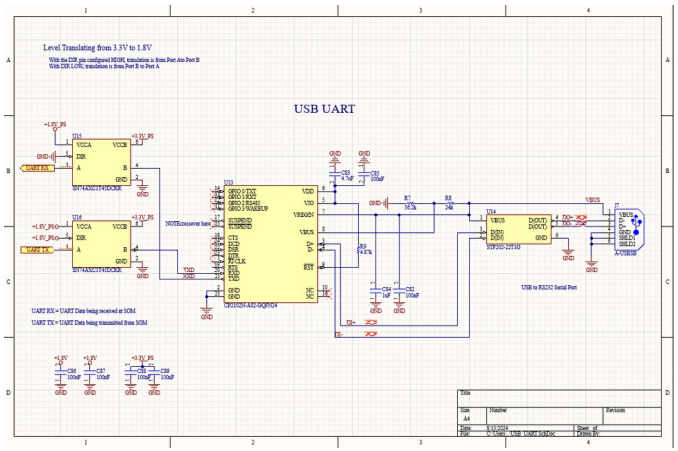
- Component placement

After gathering all the necessary libraries, each schematic sheet was created, following Fermilab's schematic format.







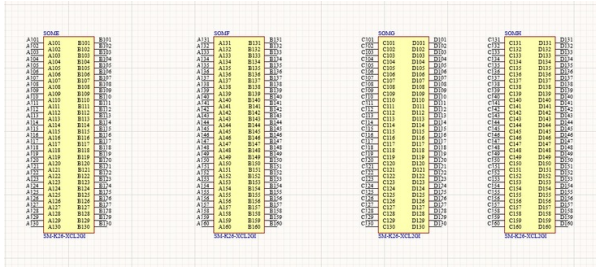


schematics design

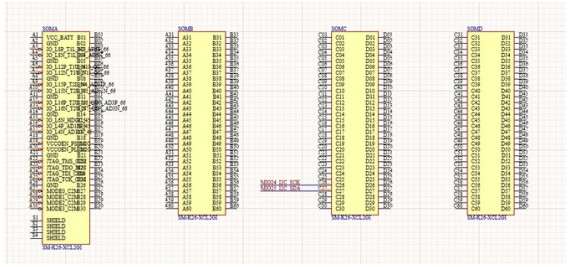
- Component placement

The KRIA, SOM_A_B_C_D, and SOM_E_F_G_H schematics correspond to the sheet 3 of the Fermilab's schematics.

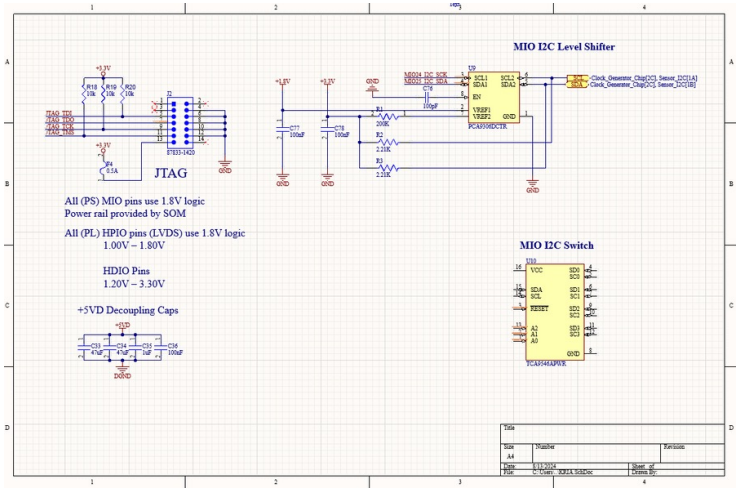
*The libraries needed to be updated



SOM_E_F_G_H schematic*



SOM_A_B_C_D schematic*



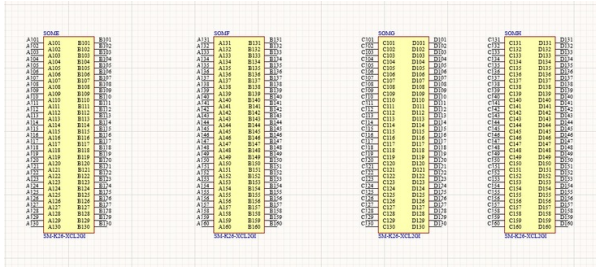
KRIA schemtic

schematics design

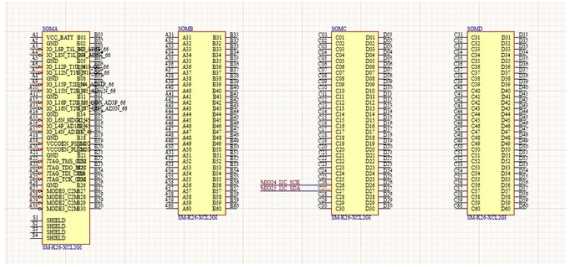
- Component placement

The KRIA, SOM_A_B_C_D, and SOM_E_F_G_H schematics correspond to the sheet 3 of the Fermilab's schematics.

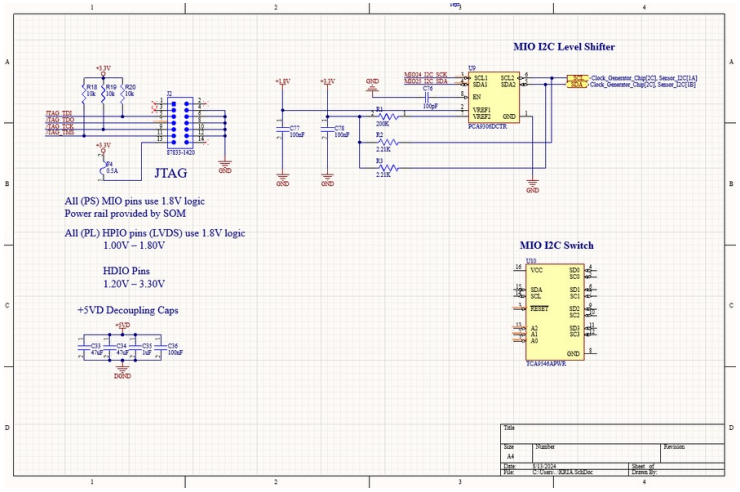
*The libraries needed to be updated



SOM_E_F_G_H schematic*



SOM_A_B_C_D schematic*

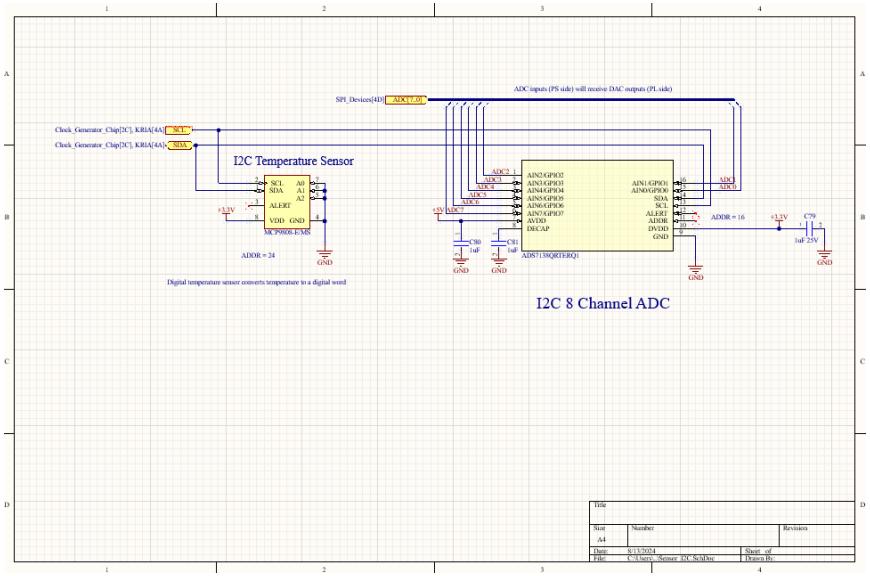


KRIA schemtic

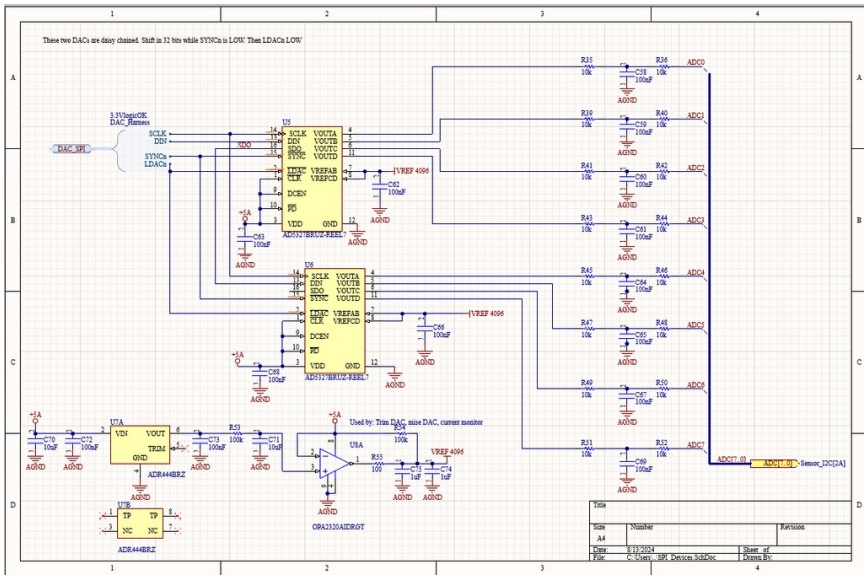
schematics design

- Component placement

Board sensors



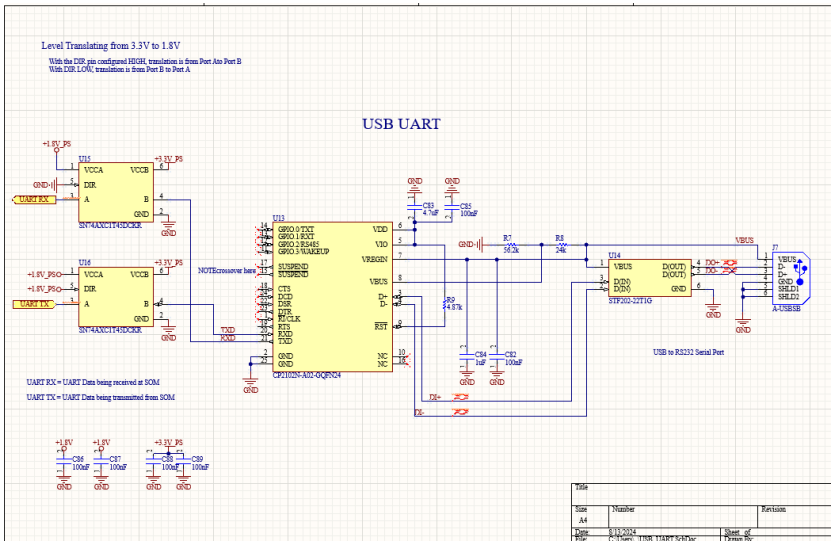
Analog chain test



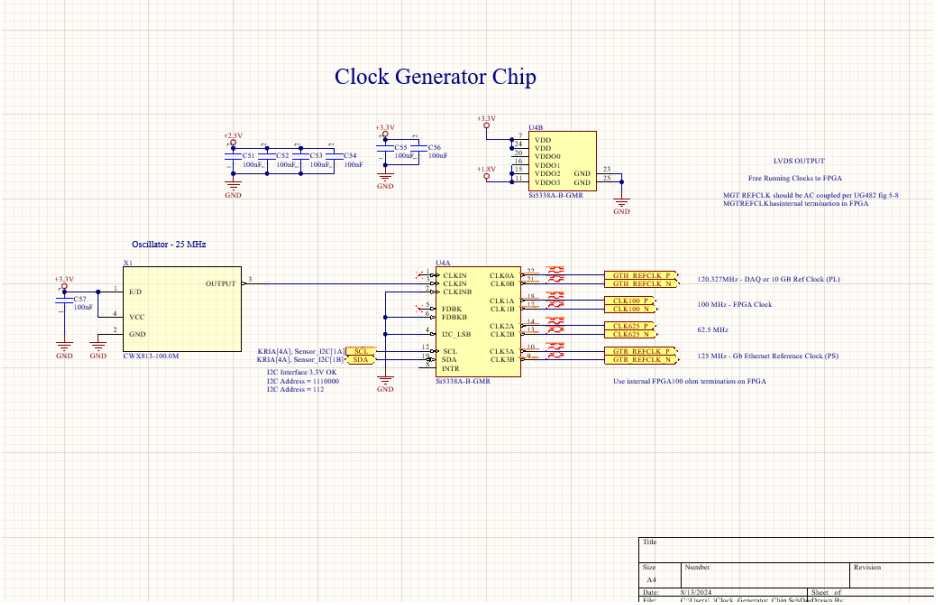
schematics design

- Component placement

JTAG and UART



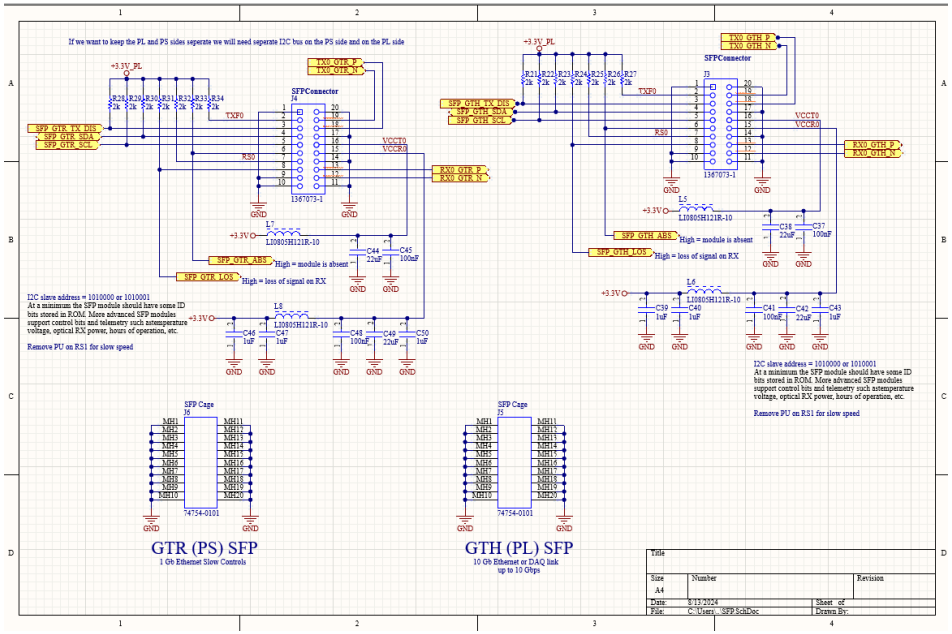
Clock generator



schematics design

- Component placement

SFP connection



schematics design

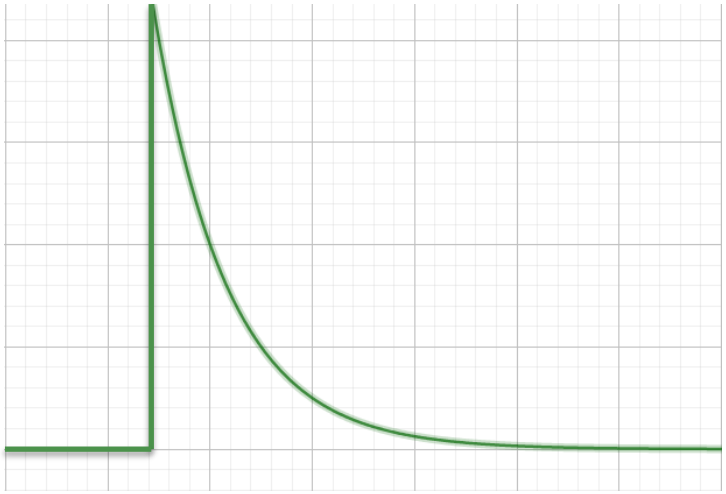
- Summary progress

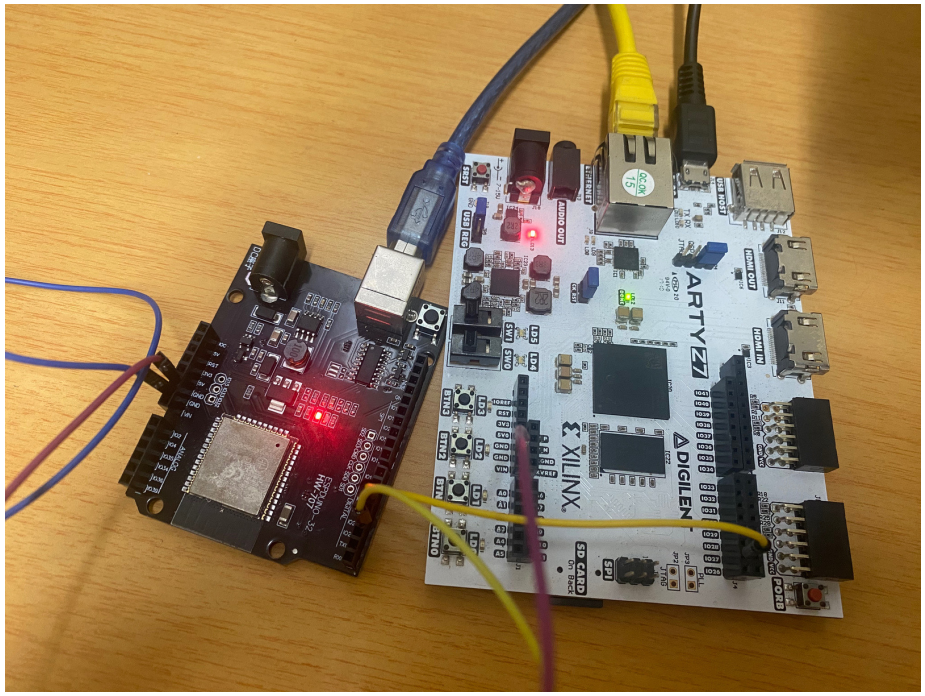
Schematic	Progress
Kria SOM connection	28
Analog sensor chain	17
Dac signal chain	20
JTAG and UART	90
Clock generator	90
SFP connection	52
Test point coupling	3
Voltage sources	80

Generation of exponential pulses

- Pulse simulation

The Generation of Exponential Pulses with Modulation combines the use of the Z-Transform and predefined data arrays to create pulses in digital systems.





Experimental setup

Generation of exponential pulses

- Pulse simulation

Difference equation implementation

```
--import library
library IEEE;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity expo_transf_Z is
    Port ( clk : in STD_LOGIC; --clock divided to 115200
           reset : in STD_LOGIC; --reset
           pulse_trig : in STD_LOGIC; --pulse
           signalout : out STD_LOGIC_VECTOR (7 downto 0); --output signal
           enable_tx : out STD_LOGIC); --enables transmission
end expo_transf_Z;
architecture Behavioral of expo_transf_Z is
    constant Yinicio : integer := 253;
    constant alfa : integer := 968;
    type states_type is (S_rst, S0, S1, S2); --defines 4 states
    signal current_s, next_s : states_type; --signal to store current and next state
    signal counter : integer := 0; --for counting
begin
    transition: process(clk, reset)
           variable Yvector : std_logic_vector(19 downto 0);
           variable Ypaso   : std_logic_vector(7 downto 0);
           variable Yactual: integer;
           variable Ycambio: integer;
    begin
        if reset = '1' then
            current_s <= S_rst;
        --if reset is activated, set the current state to S_rst
        elsif rising_edge(clk) then
            enable_tx <= '1';
            current_s <= next_s;
        --if there is a rising edge in the clock, enable transmission and update the current state with the next state
            if current_s = S_rst then
                Yvector := (others => '0');
                Ycambio := 0;
                Yactual := Yinicio;
                signalout <= std_logic_vector(to_unsigned(Yactual, 8));
             --if the current state is S_rst, initialize variables and set the output signal
             --to the binary representation of 253
            elsif current_s = S0 then
                Yactual := Yinicio;
                signalout <= std_logic_vector(to_unsigned(Yactual, 8));
            --if the current state is S0, show the binary representation of 253
            elsif current_s = S1 then
                Yactual := Yactual * alfa;
                Yvector := std_logic_vector(to_unsigned(Yactual, 20));
                Ypaso   := Yvector(17 downto 10);
                signalout <= Ypaso;
                Yactual := to_integer(unsigned(Ypaso));
            --if the current state is S1, perform calculations with Yactual and alfa, updating Yvector and Ypaso
            --displaying the binary representation of Ypaso
                if Ypaso = "00000000" then
                    current_s <= S2;
                --if Ypaso is 0, then transition to state S2
                end if;
            elsif current_s = S2 then
                signalout <= (others => '0');
            end if;
        end if;
       --if the current state is S2, set the output to 0
    end process;

    states_logic : process(current_s, pulse_trig)
    --this process defines the logic between states
    begin
        case current_s is
            when S_rst =>
                next_s <= S1;
            --S_rst always transitions to state S1
            when S0 =>
                if pulse_trig = '1' then
                    next_s <= S0;
                  --S0 transitions to itself if the pulse is active
                else
                    next_s <= S1;
                end if;
                --otherwise, it transitions to S1
            when S1 =>
                if pulse_trig = '1' then
                    next_s <= S0;
                 --S1 transitions to S0 if the pulse is active
               else
                    next_s <= S1;
                end if;
                --otherwise, it transitions to S1
            when S2 =>
                if pulse_trig = '1' then
                    next_s <= S0;
                 --S2 transitions to S0 if the pulse is active
                else
                    next_s <= S2;
                end if;
                --otherwise, it transitions to S2
            when others =>
                    next_s <= S2;
                    --in other cases, it transitions to S2
        end case;
    end process;
end Behavioral;
```

Generation of exponential pulses

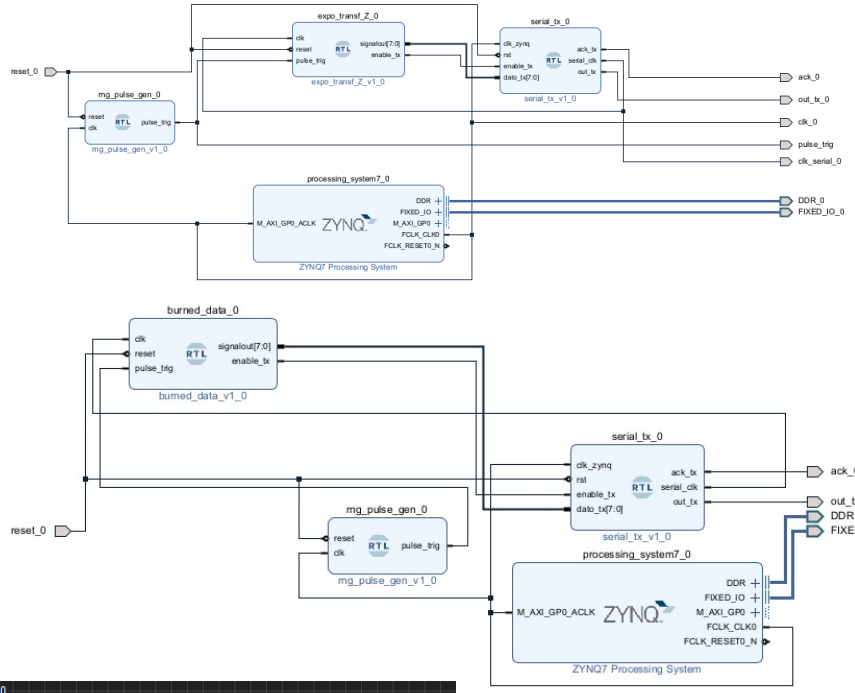
- Pulse simulation

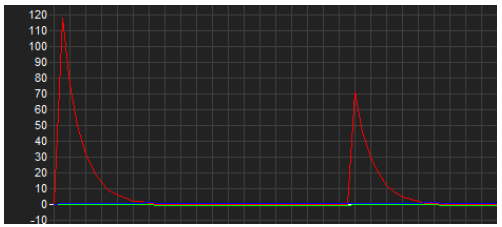
Array approximation implementation

```
--import libraries
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity burned_data is
    Port( clk : in std_logic; --clock divided to 115200
          reset : in std_logic; --reset
          pulse_trig : in std_logic; --pulse
          signalout : out std_logic_vector(7 downto 0); --data output
          enable_tx : out std_logic); --enables transmission
end burned_data;

architecture Behavioral of burned_data is
    type data_arr is array (0 to 127) of integer range 0 to 255; --array of the 128 burned data values
    signal data_array : data_arr := (128, 123, 118, 113, 109, 105, 101, 97, 93, 89, 86, 83, 79, 76, 73, 71,
                                     68, 65, 63, 60, 58, 56, 54, 52, 50, 48, 46, 44, 42, 41, 39, 38, 36,
                                     35, 33, 32, 31, 30, 29, 27, 26, 25, 24, 23, 23, 22, 21, 20, 19, 19,
                                     18, 17, 16, 16, 15, 15, 14, 14, 13, 12, 12, 12, 11, 11, 10, 10, 9,
                                     9, 9, 8, 8, 8, 7, 7, 7, 7, 6, 6, 6, 6, 5, 5, 5, 5, 5, 4, 4, 4, 4,
                                     4, 4, 4, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
                                     2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1);
    signal index : integer := 127; --index traversing the array

begin
    process(clk, reset, pulse_trig) --this block is activated by changes in the divided clock, reset, or the pulse
    begin
        if reset = '1' then
            --index <= 0;
            signalout <= (others => '0');
            --if reset is activated, then the output is 0
        elsif rising_edge(clk) then
            enable_tx <= '1';
            --if there is a rising edge in the clock, then transmission is enabled
            if pulse_trig = '1' then
                index <= 0;
                signalout <= conv_std_logic_vector(data_array(index), 8);
                --if there is a pulse, then the index is reset and signalout is assigned the first
                --element of the array
            else --if there is no pulse
                if index = 127 then
                        signalout <= "00000000";
                --if the index is 127, meaning the end of the array, then the output is 00000000
                else
                    signalout <= conv_std_logic_vector(data_array(index), 8);
                    index <= index + 1;
                --otherwise, signalout is assigned the corresponding value from the array and the index is incremented
                --to move to the next element in the next iteration
                end if;
            end if;
    end if;
    end process;
end Behavioral;
```



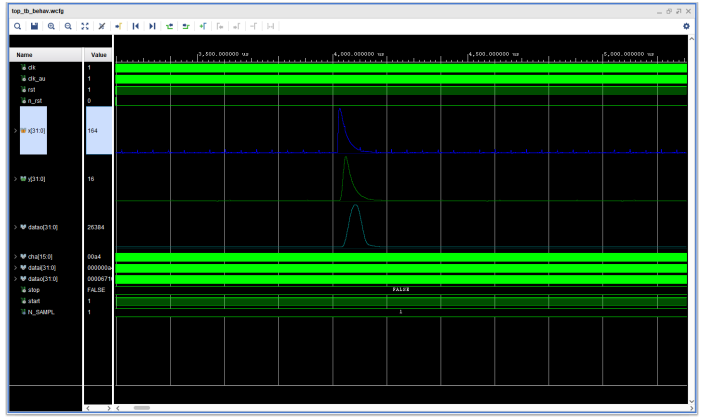


Generation of exponential pulses

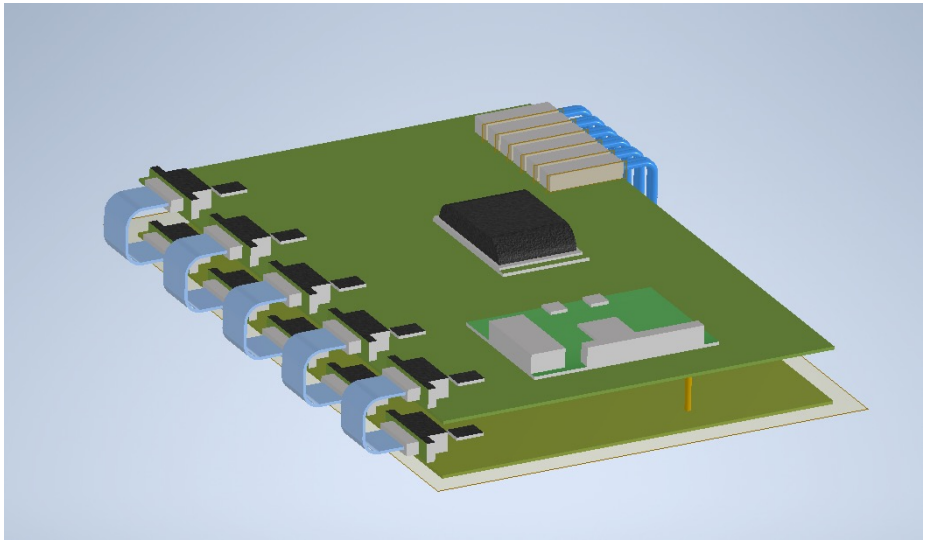
- Pulse simulation

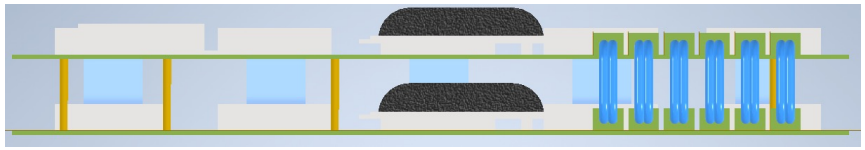
Comparison

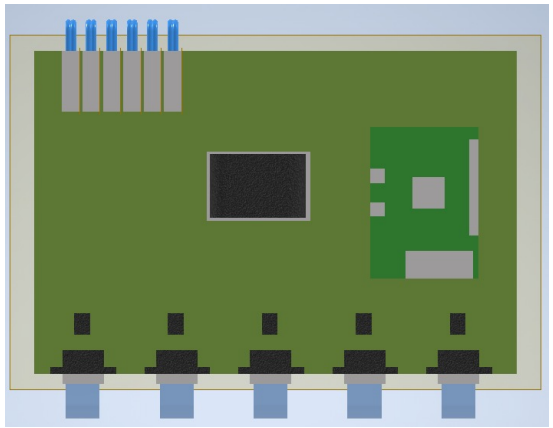
Name	Constraints	Status	WNS	TNS	WHS	THS	...	TPWS	Total Power	Fa...	Methodology	...	QoR Sugg...	LUT	FF	BRAM	URAM	DSP
synth_1 (active)	constrs_1	Synthesis Out-of-date												0	0	0	0	0
impl_1	constrs_1	Implementation Out-of-date	13.534	0.000	0.125	0.000		0.000	1.675	0	113 CW, 2 Warn			223	180	0	0	1
Out-of-Context Module Runs																		
design_2		Submodule Runs Complete																
design_2_processing_system7_0_0_synth_1	design_2_processing_system7_0_0	synth_design Complete!												24	0	0	0	0
design_2_expo_transf_Z_0_0_synth_1	design_2_expo_transf_Z_0_0	synth_design Complete!												9	18	0	0	1
design_2_serial_tx_0_0_synth_1	design_2_serial_tx_0_0	synth_design Complete!												105	77	0	0	0
design_2_rng_pulse_gen_0_0		Using cached IP results																
burned_data_design		Submodule Runs Complete																
burned_data_design_processing_system7_0_0_	burned_data_design_processing_system7	synth_design Complete!												24	0	0	0	0
burned_data_design_burned_data_0_0_synth_1	burned_data_design_burned_data_0_0	synth_design Complete!												33	47	0	0	0
burned_data_design_serial_tx_0_0		Using cached IP results																
burned_data_design_rng_pulse_gen_0_0		Using cached IP results																

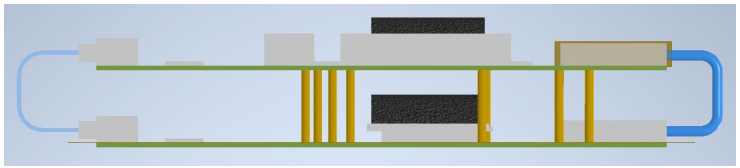


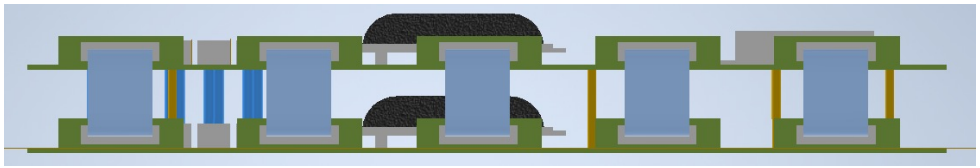
Previous Concept design



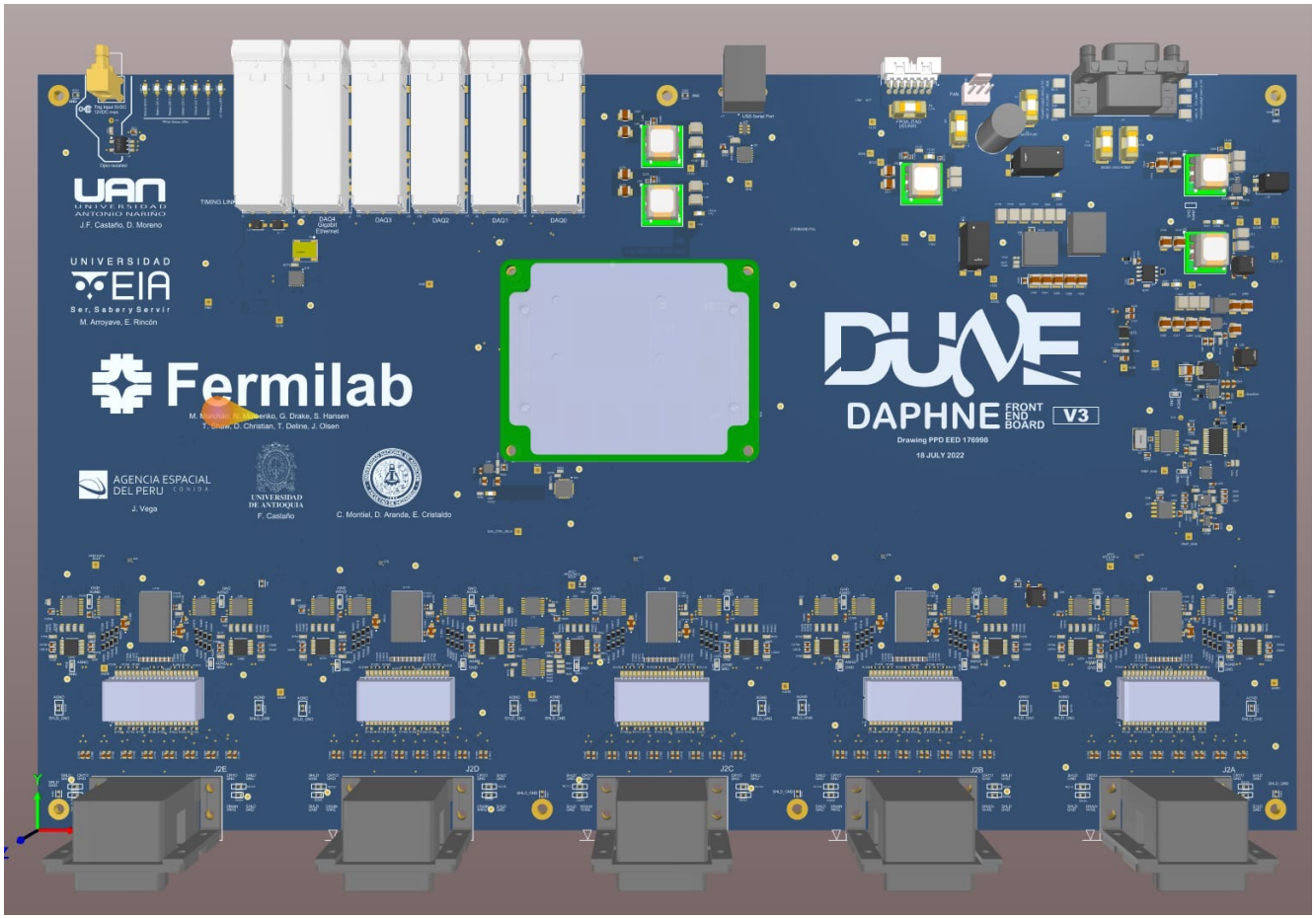




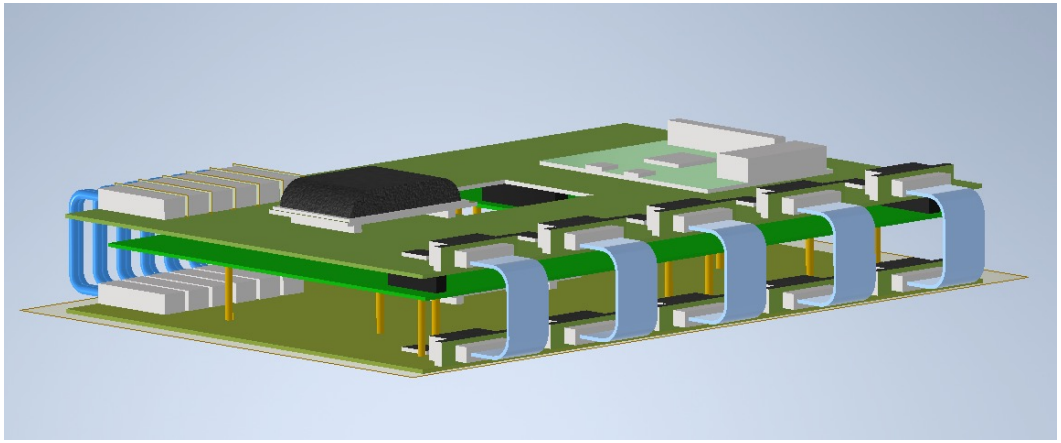


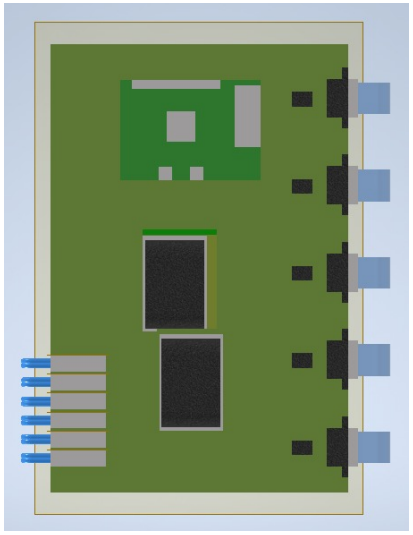


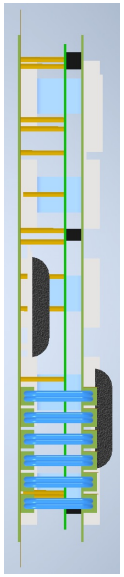
Previous Concept design

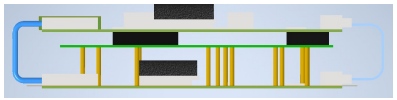


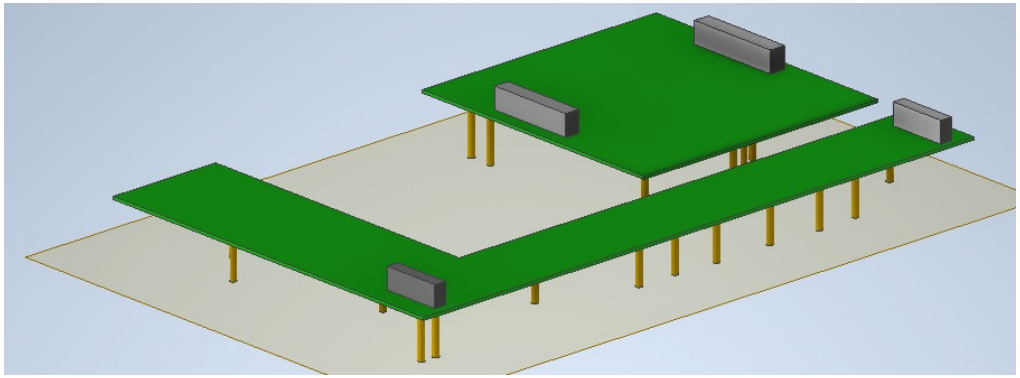
New Concept design











Next Steps

- Completing all remaining schematics with their appropriate libraries.
- Designing boards for DAC testing
- Random Signal Modulation

Thank you!