Power pulsing scheme for analog and digital electronics of the vertex detectors at CLIC

G. Blanchot*, C. Fuentes1)*

* CERN, Switzerland

Abstract

The precision requirements of the vertex detector at CLIC impose strong limitations on the mass of such a detector (< 0.2% of a radiation length, X0, per layer). To achieve such a low material budget, ultra-thin hybrid pixel detectors are foreseen, while the mass for cooling and services will be reduced by implementing a power pulsing scheme that takes advantage of the low duty cycle of the accelerator. The principal aim is to achieve significant power reduction without compromising the power integrity supplied to the front-end electronics. This report summarises the study of a power pulsing scheme to power the vertex barrel electronics of the future CLIC experiment. Its main goal is to describe in more detail what has been already presented in TWEPP conferences and other presentations. The report can therefore serve as an operator manual for future use and development of the system described.

This work was carried out in the framework of the CLICdp collaboration

1) cafuente@cern.ch
Contents

1 Introduction to CLIC and the vertex detector ........................................... 4

2 Requirements for the power delivery of the vertex detector readout electronics ........................................... 5

3 Power consumption per ASIC, per half-ladder and power pulsing ........................................... 6

4 Powering schemes for the analog electronics ........................................... 8

5 Analog electronics dummy load ........................................... 10

6 Back-end current source based solution ........................................... 12
   6.1 The basic principle ........................................... 12
   6.2 Principle and waveforms ........................................... 12

7 Current source implementation using an FPGA (analog part) ........................................... 15
   7.1 A) Current source ........................................... 15
   7.2 B) Voltage sensor ........................................... 16
   7.3 C) Front-end control signals ........................................... 17
   7.4 D) Overvoltage protection ........................................... 18
   7.5 E) FPGA ........................................... 18

8 FPGA’s firmware (analog part) ........................................... 22
   8.1 a) FPGA block of the current source ........................................... 22
   8.2 b) FPGA block of the voltage sensor ........................................... 24
   8.3 c) FPGA block of the front-end control signals ........................................... 26
   8.4 d) FPGA block of the overvoltage protection ........................................... 26
   8.5 e) FPGA control ........................................... 29
   8.6 Simulink block diagram ........................................... 35

9 Measurements and results of the implemented current source to power the analog electronics ........................................... 37
   9.1 Setup ........................................... 37
   9.2 Results ........................................... 37
      9.2.1 Regulation and power dissipation ........................................... 37
      9.2.2 Material budget calculation ........................................... 38

10 Proposed powering schemes for the digital electronics ........................................... 40

11 Digital electronics dummy load ........................................... 41

12 Current source implementation using an FPGA (digital part) ........................................... 43
   12.1 A) Current source ........................................... 43
   12.2 B) Voltage sensor ........................................... 43
   12.3 C) Front-end control signals ........................................... 43
   12.4 D) Overvoltage protection ........................................... 43
   12.5 E) FPGA ........................................... 43

13 FPGA’s firmware (digital part) ........................................... 46
   13.1 a) FPGA block of the current source ........................................... 46
13.2 b) FPGA block of the voltage sensor. .................................................... 46
13.3 c) FPGA block of the front-end control signals. ............................. 46
13.4 d) FPGA block of the overvoltage protection. ................................. 46
13.5 e) FPGA control ................................................................. 46

14 Measurements and results of the implemented current source to power the digital electronics .............................. 51
14.1 Setup .................................................................................... 51
14.2 Results .................................................................................. 51
  14.2.1 Regulation and power dissipation ........................................... 51
14.3 Material budget calculation ............................................................. 52

15 Summary and conclusions ................................................................. 53
1 Introduction to CLIC and the vertex detector

The Compact Linear Collider (CLIC) is an electron-positron linear collider under development with a maximum centre-of-mass energy of 3 TeV [1]. It will complement the results of the LHC experiments and measure the properties of new particles with high precision [2]. Figure 1 shows the CLIC beam structure, consisting of bunch trains at a repetition rate of 50 Hz. Each train is formed by 312 bunches separated by 0.5 ns each.

![CLIC Beam Structure](image)

Figure 1: CLIC beam structure.

Two general-purpose detector concepts are currently under study for CLIC: CLIC_ILD with an axial magnetic field of 4 T and CLIC_SiD with a field of 5 T. Both concepts include a pixel vertex detector placed as close as possible to the interaction point to obtain optimal secondary vertex reconstruction and to increase the precision and efficiency of the track reconstruction in particular for low-momentum tracks.

The vertex detector is composed of a barrel and a forward vertex region, as depicted in Figure 2.

![CLIC_ILD Inner Tracking Region](image)

Figure 2: CLIC_ILD inner tracking region: In addition to the vertex barrel and endcaps (VXEC), the two silicon inner tracking layers (SIT) are shown.

The barrel region is placed at the center and is made of several layers, 3 double layers for the CLIC_ILD detector and 5 single layers for the CLIC_SiD detector. Each layer is composed of several ladders, the innermost layer of CLIC_ILD is composed of 18 ladders, while its outermost layer is composed of 34 ladders. Figure 2 shows the inner tracking region of the CLIC_ILD detector, highlighting a 24 cm long vertex ladder which is composed of 24 sensors, each bonded to a 1 cm$^2$ CLICpix hybrid readout chip currently under development [3]. The radiation exposure of the vertex detector is expected to be small, compared to the corresponding regions in high-energy hadron colliders. For the Non-Ionizing Energy Loss (NIEL), a maximum total fluence of less than $10^{11}$ n$_{eq}$/cm$^2$ per year is expected for the inner-barrel and forward-vertex layers. The simulation results for the Total Ionizing Dose (TID) predict approximately 200 Gy/year for the vertex detector region.
2 Requirements for the power delivery of the vertex detector readout electronics

The proposed power delivery solution has to fulfil the following requirements:

- **Material budget**: In order to achieve the precision physics goals, the material budget of the vertex barrel detector is required to be less than 0.2% of a radiation length, $X_o$, per detection layer. To achieve such a low mass, ultra-thin high-resolution pixel detectors are foreseen, consisting of 50 µm thick silicon sensors and 50 µm thick CLICpix readout ASICs (Application-Specific Integrated Circuits). This leaves approximately 0.1% $X_o$ for cooling, mechanical structure and powering. Therefore, the contribution of the power delivery infrastructure should be well below 0.1% $X_o$ per detection layer;

- **Low power consumption**: In order to reduce the mass contributed by cooling liquids and pipes, air cooling is foreseen instead [2]. This limits the average power removal to 50 mW cm$^{-2}$, while the instantaneous power consumption of the front-end (FE) ASICs is foreseen to reach few W cm$^{-2}$;

- **Voltage regulation**: The voltage regulation has to be stable even during high to low and low to high power consumption transitions. This restriction is particularly important for analog electronics, which are much more sensitive than digital. As a design goal, the voltage regulation for the analog electronics of the CLICpix chips should be within 50 mV in order to ensure a stable Time-over-Threshold (ToT) measurement;

- **High magnetic field**: The proposed solution has to work inside a high magnetic field of 4–5 T, preventing the use of ferromagnetic material.  

---

[2] Initial tests of the power pulsing system described here, in a magnetic dipolar field of up to 1.5 T did not reveal any problems, nor with the parameters of the powering, nor with strain due to forces induced by power pulsing in a strong field. Results are reported in a presentation [4].
3 Power consumption per ASIC, per half-ladder and power pulsing

For CLIC_ILD, each ladder contains 24 sensors, each bonded to a 1 cm\(^2\) CLICpix hybrid readout chip currently under development. The number of pixels per CLICpix chip is foreseen to be 400 \(\times\) 400, each pixel having a size of 25 \(\mu\)m \(\times\) 25 \(\mu\)m. This high channel density implies a high power consumption, which is expected to be close to 2 W cm\(^{-2}\). This power consumption is much higher than the 50 mW cm\(^{-2}\) that air-cooling is able to dissipate. Therefore, in order to decrease the average power consumption, the front-end electronics will use full power only when needed, and the rest of the time the electronics will be either idle or off.

The analog electronics of each chip have a peak power consumption of 2 W cm\(^{-2}\) during a time \(t_{on}\) around the bunch train. The time \(t_{on}\) has to be long enough to turn on the analog electronics and to process the data, and it is expected to be close to 20 \(\mu\)s. Contrary to the digital electronics, the analog electronics can be turned off completely after \(t_{on}\). By doing this, the average power consumption can be reduced by a factor corresponding to \(T/t_{on}\). In this particular case, the duty cycle is equal to \(t_{on}/T = 20 \mu s/20 ms = 1/1000\), so the average power consumption is reduced to 2 mW cm\(^{-2}\). This way of providing high power during small intervals and low power during the longest part of every period is referred to as “power pulsing”, and it represents an efficient way to reduce the average power consumption down to the allowed level.

Nevertheless, the transitions from high to low and from low to high power consumption imply load variations which make it more difficult to provide the required regulated voltage. For instance, as there are 24 chips in a ladder and the voltage required by the analog electronics is 1.2 V, the analog current in a ladder changes from 0 to 40 A. In order to facilitate the design of the power distribution system, each chip will be turned on and turned off gradually using a rise \(t_r\) and fall \(t_f\) time (reducing the current bandwidth), as depicted in Figure 3. Nevertheless, there is a compromise in how fast (difficult regulation) and how slow (increase in the average power consumption) the rise and fall time can be. In this paper we assume 1 \(\mu\)s of rise and fall time which we consider to be a good trade-off.

![Analog load current per ladder](image)

Figure 3: Analog load current per ladder. It changes from 0 to 40 A with a rise time \(t_r\) and fall time \(t_f\) of 1 \(\mu\)s each.

The barrel is foreseen to be powered symmetrically from both sides. The 24 CLICpix ASICs belonging to a ladder would be read out from both sides in groups of 12. Figure 4 summarizes the expected power consumption per half-ladder of the analog and digital electronics of the CLICpix ASICs, where \(N = 12\) corresponds to the total number of CLICpix chips in the half-ladder and \(T = 20 ms\) to the time between bunch train crossings.

Similar to the case of the analog electronics, the digital electronics have a peak of power consumption around the bunch train where all the chips consume 100 mW cm\(^{-2}\) (Figure 4). After that, the 12 chips change to an idle state of 8 mW cm\(^{-2}\) until the first chip starts to be read out.
Power consumption per ASIC, per half-ladder and power pulsing

Figure 4: Analog and digital power consumption per CLICPix-Sensor module. Further details are given in [5].

Table 1: Summary of the power consumption per half ladder

<table>
<thead>
<tr>
<th>Haf Ladder Power Summary</th>
<th>Analog Circuitry</th>
<th>Digital Circuitry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage</td>
<td>1.2 V</td>
<td>1.0 V</td>
</tr>
<tr>
<td>Regulation</td>
<td>50 mV</td>
<td>50 mV</td>
</tr>
<tr>
<td>Nominal current</td>
<td>20 A</td>
<td>4.3 A</td>
</tr>
<tr>
<td>Nominal power</td>
<td>24 W</td>
<td>4.3 W</td>
</tr>
<tr>
<td>Power pulsed averaged current</td>
<td>20 mA</td>
<td>360 mA (max.)</td>
</tr>
<tr>
<td>Power pulsed averaged power</td>
<td>24 mW</td>
<td>360 mW (max.)</td>
</tr>
</tbody>
</table>

Only one CLICpix chip is read out at a time with a power consumption of 360 mW cm$^{-2}$ while the 11 remaining chips stay in an idle state. The time needed to read out every chip is proportional to the occupancy (number of pixels hit in the sensor divided by total number of pixels per sensor). For instance, it takes around 300 µs to read a chip with an occupancy of 3%, which is the maximum expected occupancy for the barrel region. After this time, the following chip is read out, which is repeated until all the 12 chips are read out.

The average power consumption for the digital electronics, assuming $t_{on} = 20 \mu s$, $T = 20 ms$ and the longest readout time possible (300 µs), corresponds to 13.4 mW cm$^{-2}$. As the voltage required by the digital electronics is 1 V, the current transient during the bunch train will be from 96 mA to 1.2 A. Beside that load transition, there will be one extra transition every time a chip is read. When a chip is read, the current transition for the half-ladder will be from 96 mA to 444 mA. The current transient in the digital electronics case, even if more frequent, is much smaller than for the analog one. This makes it easier to provide good regulation to the digital electronics. In addition, regulation is less critical for digital electronics as it is less sensitive to voltage fluctuations. Analog and digital electronics will be powered separately. In that way, their powering schemes can be optimized independently to achieve their particular requirements.
4 Powering schemes for the analog electronics

Powering the analog electronics is more challenging than the digital, as it has a higher peak consumption, a larger current transient and it requires a better voltage regulation. As a design goal, the voltage ripple for the analog electronics of the CLICpix chips should be within 50 mV in order to ensure a stable Time-over-Threshold (ToT) measurement.

The desired regulation is normally achieved by means of Power Distribution Network (PDN) theory [6]. Knowing the allowed voltage ripple and the maximum current, a target impedance is calculated using Ohm’s law. The goal is to keep the impedance that the chips see looking into the PDN below this target impedance value over a wide frequency range, mainly defined by the bandwidth of the powered signal. The PDN is mainly formed of a voltage regulator module (VRM) that provides low impedance from DC up to a few kHz regime, bulk capacitors (electrolytic, tantalum) at the frequency region up to 100 kHz and multilayer ceramic decoupling capacitors up to the 100 MHz range. Above this frequency, the impedance the chip sees is mainly inductive caused by the package and the only way to decrease it is using on-die capacitors.

The CLICpix electronics have been designed to be turned on by parts, allowing a slower rise time and therefore a smaller bandwidth. This is crucial in allowing good regulation, as the target impedance sets a limit at around 300 kHz (for the case of $t_r = 1 \mu s$).

Several schemes were investigated in view of performance and contribution to the material budget. Local regulation turned out to be necessary in order to accomplish the required regulation while supplying more than 20 A along the 12 cm of half a ladder. Taking that into account, in [7] we proposed a scheme for powering the analog electronics based on local energy storage and local voltage regulation. This power scheme (Figure 5 and 6) consisted of a buck DC-DC converter [8] placed outside the vertex barrel region (where its mass contribution was less critical), connected through aluminium flex cables to a local regulation based on Low-DropOut regulators (LDOs) and silicon capacitors [9]. The latter are fundamental for material mass reduction as the radiation length of silicon is high and as future improvements to the technology are expected to increase their capacitance density by a factor of 4 (from 250 nF mm$^{-2}$ to 1 µF mm$^{-2}$) [9], making the $X_0$ per unit capacitance for silicon significantly smaller than for other commercially available capacitor materials. Electrolytic and tantalum capacitors, on the other hand, were avoided as their contribution to the material budget would be considerably higher.

![Figure 5: View into the inner tracking region of the CLIC_ILD detector with a sketch of the proposed cabling and powering scheme based on DC-DC converters.](image-url)
Even though the solution presented in [7] fulfilled the regulation requirements, we considered that there were a few key points where the design could be improved mainly in its material budget. Therefore in [10] we presented an improved powering scheme where the DC-DC buck converters were replaced by back-end programmable current sources, which allowed us to considerably decrease the material of the cables and to get rid of the mass of the DC-DC converter.
5 Analog electronics dummy load

The front-end readout ASICs are currently under development. In order to test any proposed power pulsing scheme (either using DC-DC converters or back-end current sources), the ASIC behavior has to be emulated using a dummy load. This was achieved by using an array of MOSFETs commuting at the expected values of current and duration.

The schemes considered for powering the analog electronics contain silicon capacitors located on the ladder, to provide energy storage as close as possible to the front-end chips, followed by low dropout regulators (LDOs) to provide the required voltage. Therefore and for convenience, the LDO’s and capacitors were integrated in the same dummy load PCB, as shown in Figure 7. The dimensions were maintained very close to the expected ones in order to get a result as close as possible to the final configuration.

The dummy load consists of 12 identical, 1 cm wide subcircuits, repeated horizontally one after the other emulating the 12 ASICs, the respective power elements and the interconnections. The top of the PCB represents the low-mass flex cable. This low-mass flex cable extension can be easily cut out from the dummy load PCB, and can thus be replaced by other variants of cable. The principal motivation for this was to try different conductive materials (for instance aluminium in order to reduce the material contribution) and thicknesses. Twenty-four SMD resistors connect the flex cable to the middle section of the PCB that contains the capacitors for energy storage and LDOs for local voltage regulation. The value of the resistors can be changed in order to evaluate the performance and dependence of the power pulsing scheme to different types of interconnections.

Similarly, the middle section is connected through resistors to the lower section that emulates the power consumption of the ASICs. Not shown in the picture at the right of the PCB are three LEMO connectors that provide 1) LDO bias voltage (6.2 V), 2) LDO enable signal and 3) MOSFET gate signal for all of the LDOs. The need of a bias voltage is specific to the LDO chip used for this prototype, and is expected not to be needed for the final application.

A simplified schematic of each subcircuit is shown in Figure 8.
Figure 8: Simplified schematic of each subcircuit. It integrates the low-mass flex cable, the array of LDOs and capacitors, the interconnections and the ASICs power emulator.
6 Back-end current source based solution

6.1 The basic principle

This scheme aims at reducing the mass of the cables by eliminating the need of a massive DC-DC converter at the proximity of the vertex barrel. The principle of this idea is very simple, but its implementation requires more effort than that for the scheme based on buck converters.

The scheme is based on storage capacitors (around the LDO input and output ports) that are placed in the ladder. In this way, the capacitors can quickly provide the current needed by the ASICs during the bunch crossing. They discharge (decreasing their voltage) in the process.

As a good estimation, the voltage drop in the capacitors can be calculated as:

\[ \Delta V = \frac{I \Delta t}{C} \]  

where \( C \) is the capacitance at the input of the LDOs and \( I \) is the current consumed by the ASICs during a time \( \Delta t \). Obviously a big capacitor, e.g. 10 mF, will produce a smaller voltage drop than a small capacitor of e.g. 10 µF for the same conditions.

We can not use such a big capacitor due to the mass contribution, so there will be a non-negligible voltage drop in the capacitor’s voltage (\( \Delta V \approx 4 \) V). For this reason we need LDOs to provide a regulated voltage to the ASICs. Nevertheless, the voltage of the capacitor should never be lower than the voltage required for the ASICs (\( V_a = 1.2 \) V, where \( V_a \) is the voltage for the analog electronics (CLICpix)) plus the dropout voltage of the LDO which depends on the particular LDO used (\( V_{do} \approx 100 \) mV in our case, i.e. using the ISL80112 LDO [11]).

For that reason, we need to ensure that the capacitors are already charged up to the desired level (\( V_m = V_a + V_{do} + \Delta V = 1.2 \) V + 100 mV + 4 V = 6.3 V) before the next bunch crossing arrives. The voltages are defined as follows: \( V_m \) is the minimal voltage on the LDO input capacitor, before each bunch crossing; \( V_a \) is defined above; \( V_{do} \) is the drop-out voltage between the input and output ports of the LDO; \( \Delta V \) is the expected voltage discharge on the LDO input capacitor during the bunch train.

When charging a capacitor using a voltage source (as is the case of a DC-DC converter) the current provided has its maximum at the beginning and decreases exponentially with time. The time needed to charge the capacitor depends on the RC constant of the path. In our application, we have a small interval of time of full consumption in which the capacitors discharge (20 µs) and then a much longer time of low power consumption (20 ms) that can be used to charge the capacitors back to their previous level (\( V_m \)).

In order to minimise the current consumption, a controlled current source can be used instead, using the whole duration of low power consumption to charge the capacitors. This charging current (\( I_{charge} \)) can be easily estimated as follows:

\[ I_{charge} = \frac{I_{discharge} \cdot \Delta t_{discharge}}{\Delta t_{charge}} = \frac{20 A \cdot 20 \mu s}{20 ms} = 20 mA \]  

where \( I_{discharge} \) is the current consumed by the ASICs during the bunch train (20 A) and \( \Delta t_{charge} \) and \( \Delta t_{discharge} \) are the charging (20 ms) and discharging (almost 20 µs) time, respectively.

The small duty cycle (\( \Delta t_{discharge}/\Delta t_{charge} \)) allows for a significantly reduced charging current. For instance, we could deliver the 20 A required by the half-ladder using a 20 mA continuous current from the back-end to the ladder. In this way, using this scheme we will not only get rid of the bulky buck converter but also significantly reduce the mass of the cables from the back-end to the ladder because the current will be smaller than 20 mA.

6.2 Principle and waveforms

Even if in an ideal simulation we could use a constant current source to charge the capacitors, this does not work in reality. For instance, if the consumption of the ASICs changes every cycle and happens
to be less than expected, the capacitors will discharge less during the on-time and after a few seconds, their voltage will exceed the breakdown voltage. On the other hand, if more charge is required, after several cycles the voltage at the capacitors could not reach $V_m$ and therefore the regulation would not be achieved.

Therefore, we need a back-end current source able to sense the voltage of the capacitors just after the on-time, and capable of calculating the current required to arrive to the desired voltage just before the next on-time.

The current needed to charge the storage capacitors depends on the operating conditions of the readout electronics (temperature, occupancy, power consumption) which can vary from one bunch train to the next. A feedback mechanism is therefore implemented adjusting the back-end current based on the voltage at the capacitors at the end of the on-time.

This is illustrated in Figure 9. A field-programmable gate array (FPGA) reads the sensed voltage of the storage capacitors and calculates the next value of the current ($I_{in}$) needed to reach $V_m$, which is programmed to the current source (see section 7 for details on the FPGA).

As the current flowing through the cable is of the order of a few tens of mA, it is possible to measure the voltage of the capacitors at the back-end, reducing the number of cables and therefore the mass. The voltage drop can be compensated knowing the resistance of the cable, as the injected current is known.

To illustrate the most important signals, let us suppose that the analog back-end input current ($I_{in}$) during the previous period was 20 mA. During the time $t_{on}$ the front-end ASICs consume the 20 A. The storage capacitors discharge $\Delta V$ from their initial voltage ($V_m$) and the LDOs provide a regulated voltage close to 1.2 V. After this time, the analog electronics is turned off and the voltage is not needed anymore. For that reason and in order to save power, the LDOs are disabled, dropping their voltage to zero. The voltage at the capacitors is then measured and the new required current is calculated in order to reach the voltage $V_m$ in the available remaining time. If the power consumption during this bunch train is different from the previous one, then the current will be slightly different as depicted in Figure 10. The LDOs are enabled 200 µs before the next bunch train to reach the required voltage before the electronics pass to the next high power consumption state, after which the cycle repeats again. For this first prototype the LDOs are enabled/disabled using a signal from the FPGA (not shown in Figure 9). In the final application that signal could be either provided by the front-end chip or not needed at all in case we use LDOs with smaller quiescent current (in which case the LDOs would be enabled all the time).

![Figure 9: Diagram of a controlled current source powering a half-ladder.](image-url)
Figure 10: Typical waveforms. (left) Back-end current. (middle) Capacitor voltage. (right) Output voltage.
In this section, we explain how the current source was implemented and built. Details are shown later in this paper, see for example Figure 42. In order to implement the powering scheme of Figure 9 we need basically five subcircuits with very well defined functions.

### 7.1 A) Current source.

Its function is to provide the desired current to charge the capacitors. The value of this current will be calculated by the FPGA, so some level of communication between the two has to be foreseen. It needs to fulfil two restrictions:

1. To provide current in the range of few tens of mA with a resolution close to 0.1 mA.
2. To have a communication interface to program the desired current value from the FPGA.

The first restriction was fulfilled using the commercial 2-terminal Programmable Current Source LT3092 from Linear Technology. Figure 11 shows a simple diagram of LT3092, where the value of the current $I_{source}$ can be set using a resistor $R_{set}$ and a resistor $R_{out}$ as follows:

$$I_{source} = 10 \mu A \cdot \frac{R_{set}}{R_{out}}$$  \hspace{1cm} (3)

![Figure 11: Simple diagram of LT3092](image1)

It is straightforward that the current’s value can be controlled by changing the value of the resistance $R_{set}$, which can easily be achieved using a potentiometer. Using a digital potentiometer we can fulfil both restrictions at the same time, as it provides the digital interface to the FPGA that we need.

We chose the 10kΩ AD5160 digital potentiometer from Analog Devices, which features 256 positions (i.e. 8-bits) and SPI-compatible interface. A block diagram is shown in Figure 12. The FPGA will communicate to the AD5160 through the SPI interface, sending a clock (CLK), Chip Select ($\overline{CS}$, negative polarity) and Serial Data (SDI) signals.

The general equation determining the digitally programmed output resistance between $W$ and $B$ is

$$R_{WB}(D) = \frac{D}{256} \times R_{AB} + R_W$$  \hspace{1cm} (4)

where $D$ is the decimal equivalent of the 8-bit word loaded in the digital potentiometer input value, $R_{AB}$ is the end-to-end resistance (in this case 10kΩ) and $R_W$ is the wiper resistance contributed by the on resistance of the internal switch (100Ω).
This SPI interface has to be handled in the FPGA and will be explained in the FPGA section. The resolution in current will be the resistance resolution of the potentiometer \((10\, \text{k}\Omega / 256 = 39\, \Omega)\) times \(10\, \mu\text{A} / R_{\text{out}}\), i.e. close to 0.08 mA for \(R_{\text{out}}\) equal to 5 \(\Omega\). This was considered good enough. If more precision is needed in the future, the digital potentiometer could be upgraded to 10-bits improving the resolution by a factor 4.

The final simplified schematic of this subcircuit, which integrates the LT3092 and AD5160, is shown in Figure 13. At the bottom of the figure we can see the ports assigned for the FPGA. The pin number of each component is shown in blue, e.g. the FPGA is connected to the AD5160 through pins 5 (SDI), 6 (CS), 4 (CLK), 3 (GND) and 2 (\(V_{\text{DD}}\)) while the rest of the pins 7 (B) and 1 (W) are connected to 2 \(\text{k}\Omega\) resistors. The green arrow to pins 4, 5 and 6 illustrate that they are command signals and the direction shows that it goes from the FPGA to the Digital Potentiometer. We decided to put 4 \(\text{k}\Omega\) resistance in series (called \(R_{\text{serie}}\) in the following) to the digital potentiometer mainly to reduce the voltage at its terminals and to set an offset to the \(I_{\text{source}}\) current. The \(R_{\text{set}}\) resistance is, therefore, the sum of the series resistance and the potentiometer’s resistance.

![Figure 13: Schematic of the Current Source subcircuit](image)

7.2 B) Voltage sensor.

In order to calculate the current needed to charge the LDO input storage capacitor, we need to sense its voltage. The main function of this subcircuit is to measure the capacitor’s voltage and to communicate it to the FPGA. In Figure 9 we illustrated that this voltage is measured next to the capacitors. Nevertheless, as the estimated current flowing through the back-end is in the range of few tens of mA only, the voltage
Current source implementation using an FPGA (analog part)

To measure the voltage we can use a difference amplifier like the INA148 from Burr-Brown, which provides a unity-gain and high common-mode rejection. The diagram is shown in Figure 14. The analog differential voltage at the output of the amplifier is digitized by an Analog-to-Digital Converter (ADC) to be later processed by the FPGA. We chose the AD7476A 12-bits ADC from Analog Devices (diagram in Figure 15), which is included in the PmodAD1 board from Digilent (Figures 16 and 17).

Figure 18 shows a simplified schematic of the Voltage Sensor subcircuit. The voltage at the output of the difference amplifier INA148 is at times higher than the maximum voltage accepted by the ADC. Therefore a voltage divider is included in the connection to the ADC. Finally, the PmodAD1 is connected to the FPGA that reads the data from the ADC.

7.3 C) Front-end control signals.

Two signals will be generated by the FPGA and distributed to the dummy load. The first signal is needed to enable/disable the LDOs to reduce their power consumption whilst the second is needed to emulate the drop in the back-end cable is negligible and therefore the front-end LDO input capacitor voltage can be directly estimated straight at the back-end power supply output, avoiding long sensing cables. If desired, we could get a more precise measurement by compensating the voltage drop in the cables, which could be closely estimated knowing the resistance of the cables (which can be measured) and the current through them (whose value is known at every moment by the FPGA).

Two signals will be generated by the FPGA and distributed to the dummy load. The first signal is needed to enable/disable the LDOs to reduce their power consumption whilst the second is needed to emulate the drop in the back-end cable is negligible and therefore the front-end LDO input capacitor voltage can be directly estimated straight at the back-end power supply output, avoiding long sensing cables. If desired, we could get a more precise measurement by compensating the voltage drop in the cables, which could be closely estimated knowing the resistance of the cables (which can be measured) and the current through them (whose value is known at every moment by the FPGA).

To measure the voltage we can use a difference amplifier like the INA148 from Burr-Brown, which provides a unity-gain and high common-mode rejection. The diagram is shown in Figure 14. The analog differential voltage at the output of the amplifier is digitized by an Analog-to-Digital Converter (ADC) to be later processed by the FPGA. We chose the AD7476A 12-bits ADC from Analog Devices (diagram in Figure 15), which is included in the PmodAD1 board from Digilent (Figures 16 and 17).

Figure 18 shows a simplified schematic of the Voltage Sensor subcircuit. The voltage at the output of the difference amplifier INA148 is at times higher than the maximum voltage accepted by the ADC. Therefore a voltage divider is included in the connection to the ADC. Finally, the PmodAD1 is connected to the FPGA that reads the data from the ADC.

7.3 C) Front-end control signals.

Two signals will be generated by the FPGA and distributed to the dummy load. The first signal is needed to enable/disable the LDOs to reduce their power consumption whilst the second is needed to emulate the drop in the back-end cable is negligible and therefore the front-end LDO input capacitor voltage can be directly estimated straight at the back-end power supply output, avoiding long sensing cables. If desired, we could get a more precise measurement by compensating the voltage drop in the cables, which could be closely estimated knowing the resistance of the cables (which can be measured) and the current through them (whose value is known at every moment by the FPGA).

To measure the voltage we can use a difference amplifier like the INA148 from Burr-Brown, which provides a unity-gain and high common-mode rejection. The diagram is shown in Figure 14. The analog differential voltage at the output of the amplifier is digitized by an Analog-to-Digital Converter (ADC) to be later processed by the FPGA. We chose the AD7476A 12-bits ADC from Analog Devices (diagram in Figure 15), which is included in the PmodAD1 board from Digilent (Figures 16 and 17).

Figure 18 shows a simplified schematic of the Voltage Sensor subcircuit. The voltage at the output of the difference amplifier INA148 is at times higher than the maximum voltage accepted by the ADC. Therefore a voltage divider is included in the connection to the ADC. Finally, the PmodAD1 is connected to the FPGA that reads the data from the ADC.

7.3 C) Front-end control signals.

Two signals will be generated by the FPGA and distributed to the dummy load. The first signal is needed to enable/disable the LDOs to reduce their power consumption whilst the second is needed to emulate the drop in the back-end cable is negligible and therefore the front-end LDO input capacitor voltage can be directly estimated straight at the back-end power supply output, avoiding long sensing cables. If desired, we could get a more precise measurement by compensating the voltage drop in the cables, which could be closely estimated knowing the resistance of the cables (which can be measured) and the current through them (whose value is known at every moment by the FPGA).

To measure the voltage we can use a difference amplifier like the INA148 from Burr-Brown, which provides a unity-gain and high common-mode rejection. The diagram is shown in Figure 14. The analog differential voltage at the output of the amplifier is digitized by an Analog-to-Digital Converter (ADC) to be later processed by the FPGA. We chose the AD7476A 12-bits ADC from Analog Devices (diagram in Figure 15), which is included in the PmodAD1 board from Digilent (Figures 16 and 17).

Figure 18 shows a simplified schematic of the Voltage Sensor subcircuit. The voltage at the output of the difference amplifier INA148 is at times higher than the maximum voltage accepted by the ADC. Therefore a voltage divider is included in the connection to the ADC. Finally, the PmodAD1 is connected to the FPGA that reads the data from the ADC.
the consumption of the chips, i.e. the signal that activates the MOSFETs of the dummy load. Each signal will firstly pass through a difference amplifier and will then be compared to 1 V using the LT1720 dual comparator from Linear Technology, as shown in Figure 19.

7.4 D) Overvoltage protection.

The main function of this subcircuit is to avoid the capacitors from charging to a higher voltage than the one chosen by design. The FPGA will close a switch to ground when the voltage reaches the limit, therefore diverting the current back to the source as shown in Figure 20.

7.5 E) FPGA

An FPGA creates the signals required for the operation of the setup and calculates the current required for the correct operation of the power pulsing system. We chose the commercial evaluation board Nexys 2 from Digilent (Figure 21) based on the Xilinx Spartan 3 chip. It includes several I/O devices and ports, e.g. four 12-pin Peripheral Module (Pmod) connectors that are used to communicate with the four subcircuits described above (Figure 22). The Nexys 2 board also has 8 slide switches, 8 LEDs and 4 pushbuttons used to select different configuration modes and to display visual outputs.

The FPGA was programmed using Simulink and Sysgen. Alternatively, ISE from Xilinx could have been used. Simulink is based on block diagrams and has little code required, which was suitable for this particular application, making it much faster to develop and test. The load could also easily be modelled in Simulink making the simulation of the program less tedious. The following chapter will briefly present the different parts of the FPGA’s Firmware.
Current source implementation using an FPGA (analog part)

Figure 19: Schematic of the Front-End Control Signals subcircuit

Figure 20: Schematic of the Overvoltage Protection subcircuit
Figure 21: Picture of Digilent Nexys 2 evaluation board.
Current source implementation using an FPGA (analog part)

Figure 22: Schematic of the controlled current source, showing the subsections A,B,C,D connected to the FPGA through four Peripheral Modules (Pmod).
8 FPGA’s firmware (analog part)

The first step was to create the part of the code where the FPGA communicates with the rest of the sections. The capacitor’s voltage is read from the ADC in order to determine the value of the current for the digital potentiometer. In addition the signals to enable the LDOs, the MOSFETs and to activate the over voltage protection’s switch are created. In a second step the control for the operation of the power pulsing system was created.

8.1 a) FPGA block of the current source.

The first block sets the resistance of the digital potentiometer AD5160 (see Figure 13) in order to generate the desired current. It has to generate the SPI signals shown in Figure 23. The I/Os of this block are the following:

- **(Input) EN**: Input that enables the communication with the digital potentiometer.
- **(Input) Address**: The value that will be written in the potentiometer.
- **(Output) clk**: The clock needed by the digital potentiometer. It is generated inside the block and sent together with the other two outputs.
- **(Output) CS_neg**: This signal frames the 8-bit word sent to the digital potentiometer. When CS_neg is low, the clock loads data into the serial register on each positive clock edge.
- **(Output) SDI**: The 8-bit serial data word, which is loaded MSB first.

![Figure 23: SPI interface timing diagram.](image)

Figure 24 shows the implemented simulink block to be programmed in the FPGA. It is composed of several sub-blocks. The upper right block creates the clock, the block in the middle creates the CS frame which is later inverted by the following block to produce CS (CS_neg). Finally the block at the lower right takes the address input (D in Equation 4) of the main block and transforms it into a 8-bit long serial word with the most significant bit first to select the resistance of the digital potentiometer.
Figure 24: Simulink block of the Current Source. It changes the resistance of the digital potentiometer AD5160 using SPI. This block generates the clock and CS to be sent to the digital potentiometer. The position of the potentiometer (which sets the value of the resistance as in Equation 4) is introduced as a number from 0 to 255, and then converted into a 8-bit serial word, which is loaded MSB first, and sent to the ADC together with the clock and CS. (The original high-resolution file of this figure can be found in [12].)
8.2 b) FPGA block of the voltage sensor.

This block has to read the 12-bit word from the ADC AD7476A (see Figure 18), which represents half of the voltage of the capacitors. For this, it has to generate the clock and CS signals shown in Figure 25 and convert the 16-bit serial word read (SDATA) into a decimal number between 0 and 4095. The I/Os of this block are the following:

- **(Input) en_counter**: Input that enables the readout of the ADC. This will be used to control the frequency at which the ADC is read.
- **(Input) Signal_read_ADC**: This is the 16-bit serial word SDATA sent from the ADC to the FPGA. The first four serial bits are zero, while the following 12-bit word corresponds to the sampled analog input (i.e. the voltage read by the ADC) with the MSB first.
- **(Output) clk**: Clock. Sixteen serial clock cycles are required to perform the conversion process and to access data from the ADC.
- **(Output) CS_neg_ADC**: This signal initiates the data transfer and conversion process. The analog input is sampled at the falling edge of CS. Also, the conversion is initiated at this point.
- **(Output) Voltage**: This is the de-serialized equivalent of the SDATA signal sent by the ADC to the FPGA.

![Figure 25: Serial Interface Timing Diagram ADC.](image)

Figure 25: Serial Interface Timing Diagram ADC.

Figure 26 shows the implemented Simulink block to be programmed in the FPGA. The clock (5MHz) and the CS are created in the first sub-block. The rest of the sub blocks convert the 16-bit long SDATA to its equivalent value (from 0 to 4095).
Figure 26: Simulink block of the Voltage Sensor. It sends $C^3$ and clock signals to sample the voltage of the capacitors and converts the SDATA signal sent by the ADC to its equivalent decimal value from 0 to 4095. (The original high-resolution file of this figure can be found in [12].)
8.3 c) FPGA block of the front-end control signals.

This block has to generate a signal to enable/disable the LDOs and a signal to turn on/off the MOSFETs of the dummy load (see Figure 19).

The I/Os of this block are the following:

- **(Input) in_Sw7**: This input takes the value of the switch Sw7 of the FPGA board and controls the on-time of the MOSFETs. When its value is low, the on-time of the MOSFETs is fixed and close to 20 µs. When its value is high, the on-time varies every 20 ms taking one of three different values. Its behaviour is programmed in the function Mosfet_Time_Generator of the green block in Figure 27.

- **(Output) Mosfet_driver**: This signal turns on the MOSFETs at the dummy load to emulate the ASIC’s behaviour. The upper right part of the blue block in Figure 27 generates this signal.

- **(Output) LDOs_driver**: This signal enables the LDOs. It is set high around 200 µs before the bunch train (using internal constant Time_EN_LDO_Before_commutation) and it remains high during 250 µs. This signal is generated in the lower right part of the blue block (Figure 27).

- **(Internal Constant) Time_EN_LDO_Before_commutation**: This internal constant sets the time before the bunch train arrival at which the LDOs will be enabled. By default it is set to 990,000, which means that the LDOs will be enabled 200 µs before \((1000000 - 990000) \times 20\) ns.

8.4 d) FPGA block of the overvoltage protection.

It generates the signal to turn on the MOSFET of Figure 20 when the voltage of the capacitors reaches a value selected by the user. This will avoid the capacitors to be charged to a higher voltage than the one chosen by design.

The main I/Os of this block are the following:

- **(Input) Sw5**: This input takes the value of the switch Sw5 of the FPGA board and controls the voltage at which the over voltage circuit triggers. By default, the value is set to 6.28 V \((3900/4095 \times 2 \times 3.3)\) when Sw5 is low and 4.83 V \((3000/4095 \times 2 \times 3.3)\) otherwise. The default values can be changed in the function Fuse_Value of the blue block in Figure 28.

- **(Input) Voltage**: This is the digitized value of the last sampled voltage at the ADC. It will be used to know if the voltage has reached its limit.

- **(Output) Fuse_Lower**: This signal controls the MOSFET of Figure 20. It is high when the voltage has reached the limit, and stays high until the end of the power pulsing cycle (it is set low when the next bunch train arrives). This signal is generated in the violet block of Figure 28. It also generates the output Fuse_Upper which is the inverse of Fuse_Lower (currently not used).
Figure 27: Simulink block of the Front-End control signals. The Mosfet_Driver output at the upper right corner turns on/off the MOSFETs to emulate the ASICs behaviour while the LDOs_Driver output at the lower right corner enables/disables the LDOs to reduce power consumption. The LDOs have to be enabled before the bunch train in order to arrive at their nominal output voltage. That time can be set using the internal constant Time_EN_LDO_Before_commutation. The on-time of the MOSFETs can be changed in the function Mosfet_Time_Generator inside the green block. (The original high-resolution file of this figure can be found in [12].)
function [z] = Fuse_Value(Sw5)
Kte3900= xfix({xlUnsigned, 12, 0}, 3900);
Kte3000= xfix({xlUnsigned, 12, 0}, 3000);
if Sw5
  z=Kte3000;
else
  z=Kte3900;
end
end

Figure 28: Simulink block of the Overvoltage Protection. It generates the signal to turn on the MOSFET of Figure 20 in case the voltage of the capacitors reaches the limit value. The function Fuse_Value of the blue block sets the limit value. The violet block generates the output Fuse_Lower that controls the MOSFET. It is high when the voltage has reached the limit, and stays high until the end of the power pulsing cycle (it is set low when the next bunch train arrives). (The original high-resolution file of this figure can be found in [12].)
8.5 e) FPGA control

Once all the interfaces to communicate with the external devices have been set up, the control blocks to enable them and to implement the closed-loop control are created.

e.1) Control block to enable ADC readout

The ADC block of Figure 26 reads a sample voltage from the ADC every time that its input en_counter is high. This control block enables the ADC at two different frequencies depending on the mode of operation of the system.

In the first or initial mode of operation the minimum capacitor's voltage is not yet reached. During this mode, the capacitors charge at an initial constant current and the Front-End MOSFETs and LDOs have never been enabled. The ADC is read at a period of 40 µs.

Once the minimum capacitor's voltage is reached, we pass to the second and final mode of operation in which the system will remain. The LDOs are enabled at the very same moment that the value is reached and the MOSFETs will turn on for the first time 200 µs later. This will be repeated every 20 ms to replicate the CLIC beam structure of Figure 1. During this mode, the ADC is read every 150 µs, i.e. 133 times each period.

The block that generates the enable ADC signal corresponds to the grey block in Figure 29. Every time the ADC samples a voltage, the violet block checks if its value is equal to 4095, in which case the measurement is repeated. The block diagram of Figure 29 generates the signal to enable the ADC readout depending on the mode of operation.

e.2) Control block Voltage Over Threshold

The main function of this block is to keep track of the current operation mode. To do so, it constantly samples the capacitor's voltage (voltage_in input of the block in Figure 30) and compares it with a threshold value set in the internal constant named min_voltage_LDOs. Once the threshold is reached, the LDOs are enabled for the first time and a counter will count up to the internal constant Time_voltage_TH (by default 200 µs). After that time, the initial mode of operation ends and the final mode of operation starts. That information is sent to the block in Figure 26 through the outputs EN_Counter_ADC_inicio and EN_Counter_ADC_final, where the first is high whilst in the initial mode and the second is high whilst in the final mode of operation.
Figure 29: Simulink implementation of the control block to enable ADC readout. It will generate the signal to enable the readout of the ADC. The ADC will be read at a period of 40 $\mu$s when working in the initial mode of operation and every 150 $\mu$s when in the final mode of operation. (The original high-resolution file of this figure can be found in [12].)
Figure 30: Simulink implementation of the control block Voltage Over Threshold. Its main function is to keep track of the current operation mode and share that information with neighbouring blocks. The threshold voltage can be set in the internal constant named min_voltage_LDOs. (The original high-resolution file of this figure can be found in [12].)
### e.3) Control block: Principal control

This control block calculates the required position of the digital potentiometer (Address_Digipot output in Figure 31) in order to generate the minimum current to charge up the capacitors to a desired value (equal to the threshold voltage of the previous control block) before the next bunch train arrives.

The current required to charge the capacitor, based on the measurement of the voltage sensor, is described by the equation:

\[
I = C \frac{dU}{dt} = C \frac{V_{bias} - V_{measured}}{\Delta t} = C \frac{(V_{12\text{bits}}(bias) - V_{12\text{bits}}(t)) \cdot K_{ADC}}{\Delta t}. \tag{5}
\]

As:

\[
K_{ADC} = \text{Div} \cdot \frac{V_{ADC\_DD}}{N} = 2 \cdot \frac{V_{ADC\_DD}}{2^M} = 2 \cdot \frac{V_{ADC\_DD}}{4096}
\]

one gets:

\[
I = C \frac{dU}{dt} = C \frac{V_{bias} - V_{measured}}{\Delta t} = C \frac{(V_{12\text{bits}}(bias) - V_{12\text{bits}}(t)) \cdot 2 \cdot V_{ADC\_DD}}{4096} \tag{6}
\]

On the other hand, the current of the source is given by the equation 3. Taking into account the digitally programmed output resistance (equation 4), one obtains:

\[
I = 10[\mu A] \cdot \frac{R_{set}}{R_{out}} = 10[\mu A] \cdot \frac{R_{WB}(D) + R_{serie}}{R_{out}} = 10[\mu A] \cdot \frac{D}{256} \cdot \frac{R_{AB} + R_{W} + R_{serie}}{R_{out}}. \tag{7}
\]

Combining the two formulas (equation 7 and 8), the required position of the digital potentiometer is given by:

\[
D = \frac{C}{\Delta t} \cdot (V_{12\text{bits}}(bias) - V_{12\text{bits}}(t)) \cdot 2 \cdot \frac{V_{ADC\_DD}}{4096} \cdot \frac{R_{out}}{10[\mu A]} \cdot \frac{256}{R_{AB}} - \left( R_{W} + R_{serie} \right) \cdot \frac{256}{R_{AB}}. \tag{9}
\]

Substituting the values of the resistors, the control block uses for the calculation the final form of the equation:

\[
D = (V_{12\text{bits}}(bias) - V_{12\text{bits}}(t)) \cdot 2 \cdot \frac{3.3[V]}{4096} \cdot \frac{C}{\Delta t} \cdot \frac{5[\Omega]}{10[\mu A]} \cdot \frac{256}{10[k[\Omega]]} - \left( 100[\Omega] + 4k[\Omega] \right) \cdot \frac{256}{10[k[\Omega]]}. \tag{10}
\]

In the equations above, 
\( D \) is the required position of the digital potentiometer (AD5160) whose value ranges from 0 to 255; 
\( V_{12\text{bits}}(bias) \) is the digital value that would be sampled by the ADC (PmodAD1) for the desired voltage on the capacitor; 
\( V_{12\text{bits}}(t) \) \( K_{ADC} \) is an ADC (PmodAD1) transfer function; 
\( \text{Div} \) is a parameter reflecting voltage divider at the input of the ADC (PmodAD1); 
\( V_{ADC\_DD} \) is a full scale voltage range of the ADC (PmodAD1); 
\( N \) is a number of voltage intervals of the ADC (PmodAD1); 
\( M \) is ADC’s (PmodAD1) resolution in number of bits; 
\( C \) is the value of the storage capacitors in half a ladder i.e. 120 µF in our case; 
\( \Delta t \) is the remaining time to charge up the capacitors (20000µs - 300µs); 
\( R_{WB}(D) \) defines a general equation determining the digitally programmed resistance of the AD5160; 
\( R_{AB} \) is the end-to-end resistance on pins of the digital potentiometer (AD5160); 
\( R_{out} \) is the 5[Ω] resistor of the current source circuit of Figure 11; 
\( R_{serie} \) represents the series resistance connected to the Digipot AD5160 as defined in section 7.1A and in Figure 13;
$R_w$ is the wiper resistance contributed by the on resistance of the internal switch of the digital potentiometer (AD5160).

The code to compute this equation is contained inside the Nb_Finder function shown in the top part of Figure 31. What is shown in the lower part corresponds to the Overvoltage protection presented in Figure 28.
function [Nb,En_DP] = Nb_Finder(vf,v1,cuentas, Naction,Rserie,sw01)
if sw01==0
tke1= xfix({xlUnsigned, 27, 24}, 0.1047 );
end
if sw01==1
tke1= xfix({xlUnsigned, 27, 24}, 0.1152 );
% C=110uF
elseif sw01==2
kte1= xfix({xlUnsigned, 27, 24}, 0.1204 );
% C=115uF
else
kte1= xfix({xlUnsigned, 27, 24}, 0.1257 );
% C=120uF
end
kte2= xfix({xlUnsigned, 27, 24}, 0.0255);
%%%%%%% Body %%%%%%%%
if v1~=0 & v1~=4095 & cuentas==Naction
if v1<vf
Nb=(vf-v1)*kte1-kte2*(Rserie+100);
if Nb>255
Nb=255;
else
Nb=Nb+1;
end
%% I add 1 so the floor function outside will be actually a ceil function
else
Nb=1;
end
En_DP=1;
else
Nb=0;
En_DP=0;
end
end
function [z] = Fuse_Value(Sw5)
Kte3900= xfix({xlUnsigned, 12, 0}, 3900);
Kte3000= xfix({xlUnsigned, 12, 0}, 3000);
if Sw5
z=Kte3000;
else
z=Kte3900;
end
end

Figure 31: Simulink implementation of the Principal control block. Its main function is to calculate the required position of the digital potentiometer ($N_b$) computed using Equation 10 which was programmed in the function Nb_Finder. The lower part of the block corresponds to the Overvoltage protection presented in Figure 28. (The original high-resolution file of this figure can be found in [12].)
8.6 Simulink block diagram

Finally, all the blocks previously presented were integrated in one single Simulink file shown in Figure 32.
Figure 32: Final Simulink block diagram of the FPGA’s based current source to charge the analog part of the ASICs. (The original high-resolution file of this figure can be found in [12].)
9 Measurements and results of the implemented current source to power the analog electronics

9.1 Setup

The CLICpix FE readout ASICs are currently under development. In order to test the proposed scheme, their behavior was emulated using an array of MOSFETs commuting at the expected values of current and duration using signals from the FPGA. For convenience, the low-mass flex cable, LDOs and capacitors were integrated in a single PCB, as shown in Figure 33. The 1 mm wide flex cable has two layers of 20 µm thick aluminium instead of copper to reduce the mass contribution. For the same reason, the input and output capacitors are IPDIA silicon capacitors [9]. The LDOs are commercial off-the-shelf components.

Figure 33: Analog dummy load PCB.

Figure 34 shows the implemented setup. On the left, four voltage supplies power the FPGA and a custom-made PCB serves as the controlled current source. The latter is connected to the back-end cables (represented each by 16 µH inductors with a series resistance of 1 Ω), which are then followed by the aluminium flex cable that is connected to the PCB emulating the FE electronics. The current at the load is measured using a current probe and the voltage regulation is measured using a scope voltage probe on the last chip, which represents the most challenging case. Finally, signals from the FPGA are used to trigger the MOSFET array and to disable/enable the LDOs.

9.2 Results

9.2.1 Regulation and power dissipation

The regulated voltage at the last ASIC is measured for $t_{on}$, $t_r$, and $t_f$ equal to 20 µs, 1 µs and 1 µs, respectively. The analog voltage is constant during the on-time, having less than 20 mV voltage drop during the full on-period (Figure 35). This is achieved using capacitors at the input and output of the LDOs of 10 µF and 1 µF, respectively. The scheme has also been tested for varying readout times to emulate variations from chip to chip and from one bunch train to the next. This resulted in an equivalent regulation of the output voltage.

The power consumption resulting from this scheme has been measured. The total power losses are below the required 50 mW cm$^{-2}$, being less than 10 mW cm$^{-2}$ for analog, therefore leaving around 40 mW cm$^{-2}$ for the digital part. Most of the losses are due to the LDOs and can eventually be reduced at the cost of adding more material.
9 Measurements and results of the implemented current source to power the analog electronics

![Analog test setup](image)

**Figure 34:** Analog test setup.

**Figure 35:** Measurement of main signals (Analog electronics).

### 9.2.2 Material budget calculation

The material budget contribution of the powering infrastructure is calculated taking into account the components placed on the half-ladder: the low-mass flex cable composed of two conductive layers and a dielectric in the middle, the input and output capacitors and the LDOs. The average material thickness of each component was calculated assuming that the material is distributed evenly over the surface of the ladder. Figure 36 (left) shows the results of the calculation using the technologies available today. The total analog material budget contribution is 0.064% $X_o$, which, although small, is close to the material target of 0.1% $X_o$ to share with the digital part and mechanical structures. From the pie chart (Figure 36 (left)) it can be noted that the silicon capacitors are the dominating contribution (75%).

It is expected that within the next few years silicon capacitors with 4 times larger capacitance density
Measurements and results of the implemented current source to power the analog electronics

Figure 36: Material budget calculation for the powering of the ladder area: (left) with today’s technology and (right) with improved silicon capacitors technology expected to be available in a few years time.

will become available [9]. Taking this into account for the calculation, the total analog contribution decreases to 0.028% $X_0$ (Figure 36 (right)). The material can be decreased further using flex cables with thinner/narrower aluminum conductors. The width of the conductor can be decreased further ensuring that its resistance does not produce excessive voltage drop and/or power dissipation in the cable.
10 Proposed powering schemes for the digital electronics

For optimal performances, it is foreseen to have separate analog and digital powering schemes. Following the analog powering scheme, presented above, the option of using a similar scheme for the digital part is assessed here.

For this we need to analyse the power consumption of the digital part that was presented in Figure 4. Similar to the case of the analog electronics, the digital electronics have a peak of power consumption around the bunch train where all the chips consume 100 mW cm\(^{-2}\), which is much smaller than the analog case. After this, the 12 chips change to an idle state of 8 mW cm\(^{-2}\) until the first chip starts to be read out. Each of the 12 chips are each read out, one after the other, with a power consumption of 360 mW cm\(^{-2}\), while the 11 remaining chips stay in an idle state.

To know if the same scheme can be applied in the digital circuits, we need to calculate the size of the storage capacitors to see if the amount of material can be kept sufficiently low. During the bunch train the power consumption is one order of magnitude smaller than that of the analog, which means that the capacitor’s size could actually be reduced. Nevertheless, after the bunch train the chips are read out for a time close to 300 µs each. The capacitor has to be sufficiently large to be ready to provide the current for the worst case (where the time to read the 12 chips is the longest). Therefore, it is not feasible to implement a similar concept as in the analog case.

However, with a small modification one can achieve a digital power pulsing scheme that is similar to the analog case. If one spaces the readout of the 12 ASICs equally in time, one will have the same behaviour as the analog case though with a frequency 13 times higher and with lower power consumptions lasting longer in time. This is shown in Figure 37. One can notice that the 12 chips change to an idle state of 8 mW cm\(^{-2}\) until the first chip starts to be read out \(\frac{20}{13}\) ms after the bunch train.

Only one CLICpix chip is read out at a time with a power consumption of 360 mW cm\(^{-2}\), while the 11 remaining chips stay in an idle state. The following chip is read out \(\frac{20}{13}\) ms after the first one started to be read out in order to keep them equally spaced in time. This is repeated until the 12 chips are read out.

![Diagram of Analog and Digital Power Consumption](image)

Figure 37: Analog and digital power consumption per half-ladder (N=12 ASICs), with the readout of the 12 ASICs every \(\frac{20}{13}\) ms.

It should be feasible to use the same principle and even reduce the size of the capacitors for each ASIC from 10 µF to 6.6 µF. The downside is that the digital electronics cannot be turned off, so the constant current of the 11 idle ASICs has to be provided from the back-end. This will imply that the back-end cables will be thicker for the digital part than for the analog part.

---

For optimal performances, it is foreseen to have separate analog and digital powering schemes. Following the analog powering scheme, presented above, the option of using a similar scheme for the digital part is assessed here.

For this we need to analyse the power consumption of the digital part that was presented in Figure 4. Similar to the case of the analog electronics, the digital electronics have a peak of power consumption around the bunch train where all the chips consume 100 mW cm\(^{-2}\), which is much smaller than the analog case. After this, the 12 chips change to an idle state of 8 mW cm\(^{-2}\) until the first chip starts to be read out. Each of the 12 chips are each read out, one after the other, with a power consumption of 360 mW cm\(^{-2}\), while the 11 remaining chips stay in an idle state.

To know if the same scheme can be applied in the digital circuits, we need to calculate the size of the storage capacitors to see if the amount of material can be kept sufficiently low. During the bunch train the power consumption is one order of magnitude smaller than that of the analog, which means that the capacitor’s size could actually be reduced. Nevertheless, after the bunch train the chips are read out for a time close to 300 µs each. The capacitor has to be sufficiently large to be ready to provide the current for the worst case (where the time to read the 12 chips is the longest). Therefore, it is not feasible to implement a similar concept as in the analog case.

However, with a small modification one can achieve a digital power pulsing scheme that is similar to the analog case. If one spaces the readout of the 12 ASICs equally in time, one will have the same behaviour as the analog case though with a frequency 13 times higher and with lower power consumptions lasting longer in time. This is shown in Figure 37. One can notice that the 12 chips change to an idle state of 8 mW cm\(^{-2}\) until the first chip starts to be read out \(\frac{20}{13}\) ms after the bunch train.

Only one CLICpix chip is read out at a time with a power consumption of 360 mW cm\(^{-2}\), while the 11 remaining chips stay in an idle state. The following chip is read out \(\frac{20}{13}\) ms after the first one started to be read out in order to keep them equally spaced in time. This is repeated until the 12 chips are read out.

![Diagram of Analog and Digital Power Consumption](image)

Figure 37: Analog and digital power consumption per half-ladder (N=12 ASICs), with the readout of the 12 ASICs every \(\frac{20}{13}\) ms.

It should be feasible to use the same principle and even reduce the size of the capacitors for each ASIC from 10 µF to 6.6 µF. The downside is that the digital electronics cannot be turned off, so the constant current of the 11 idle ASICs has to be provided from the back-end. This will imply that the back-end cables will be thicker for the digital part than for the analog part.
11 Digital electronics dummy load

As was done for the analog electronics, we need to replicate the digital power consumption in a dummy load. For the analog case, all the MOSFETs are switched at the same time. For the digital part selective switching of each MOSFET is required. This can easily be implemented using a DEMUX, so we can control the 12 MOSFETs using 5 signals from the FPGA (gate signal + 4 address signals) instead of 12 (as shown in Figure 38).

Figure 38: Simplified schematic of each subcircuit of the Digital Dummy Load. It integrates the low-mass flex cable, the array of LDOs and capacitors, and the interconnections. There is just one DEMUX for the whole dummy load, and it is used to control the MOSFETs individually from the FPGA using 5 signal cables.

Similarly to what is described above for the analog part, the dummy load consists of 12 vertical and identical 1 cm wide subcircuits, repeated horizontally one after the other emulating the 12 ASICs, the respective power elements and the interconnections, as shown in Figure 39.

The top of the PCB represents the low-mass flex cable made of aluminium that is soldered to a transition board with the silicon capacitors. Similar to the analog part, the middle part is connected through resistors to the lower part that emulates the power consumption of the ASICs. The LEMO at the right corner brings the $T_{on}$ signal that turns on and off all the MOSFETs in series with the 10.87Ω resistor at the same time. Those located at the bottom of the Figure correspond to the signals of the FPGA to control individual MOSFETs (those in series with the 2.84Ω resistors), where $A_0$ to $A_3$ control the address, while Read is the gate signal. The chosen DEMUX is the Philips HEF4514B and it is located just above the LEMO connectors.
Figure 39: PCB that emulates the digital ASICs power consumption. It integrates the low-mass flex cable, the array of LDOs and capacitors, and their interconnections. A DEMUX allows the selection of individual MOSFETs to reproduce the readout of the ASICs.
12 Current source implementation using an FPGA (digital part)

For the current source of the digital part several subcircuits of the analog circuit are reused.

12.1 A) Current source

The current source is similar to the one presented in section 7.1, but a second stage is added in parallel, providing two independent digital potentiometers. One produces the current needed to charge up the capacitors after a current transient and the other one provides the constant current due to the idle state of the ASICs (Figure 40). To cover a current range from 0 to 200 mA, the digital potentiometer has a value of 100kΩ.

12.2 B) Voltage sensor

The Voltage Sensor is the same as the one described in section 7.2.

12.3 C) Front-end control signals

The FPGA needs to provide 6 signals to control the digital circuitry power pulsing dummy load. In addition to the MOSFET enable and gate signals as required in the analog power pulsing scheme, the FPGA requires also four additional lines to control the gate signal demultiplexer. The LDOs are however here permanently turned on. The front-end control is schematically shown in Figure 40.

12.4 D) Overvoltage protection

This Overvoltage Protection is the same as the one described in section 7.4.

12.5 E) FPGA

The final schematic of the current source is shown in Figure 42. The modifications in the firmware of the FPGA are presented in the next section.
Figure 40: Final simplified schematic of the Current Source subcircuit for the Digital Part. A second stage is added in parallel to provide a constant current for the idle state of the ASICs. This allows to have two independent digital potentiometers.

Figure 41: Final simplified schematic of the Front-End Control Signals subcircuit for the Digital Part. We triplicate the analog schematic of this subsection to provide the 6 signals required.
Figure 42: Simplified schematic of the controlled current source, showing the subcircuits A,B,C,D connected to the FPGA through four Peripheral Module (Pmod). (The original high-resolution file of this figure can be found in [12].)
13 FPGA’s firmware (digital part)

For this setup a few of the blocks made for the analog current source are reused, whilst others are modified accordingly.

13.1 a) FPGA block of the current source.

The current source changes its value every $20 \frac{1}{10}$ ms to compensate the current transients while another current source predicts the idle current and provides it (its value is set once at the beginning). In the firmware, the interface with the digital potentiometer is replicated and a different control is added to detect the constant current consumption during idle mode (which is added in the control section).

13.2 b) FPGA block of the voltage sensor.

The Voltage Sensor is the same as the one described in section 8.2.

13.3 c) FPGA block of the front-end control signals.

In this part the signal to the LDOs is removed and 5 new signals are added to control the MOSFETs to replicate the readout of the ASICs. This will be done using one gate signal to the input of the DEMUX and 4 address signals to select which of the 12 MOSFETs will get the input signal. The address is generated using the code at the right of Figure 43. The code at the left of the Figure allows to put different readout times to each of the ASICs. It can be activated using switch sw6 of the board. By default the readout time is set to the maximum expected of 300 µs.

13.4 d) FPGA block of the overvoltage protection.

The Overvoltage Protection is the same as the one described in section 8.4.

13.5 e) FPGA control

e.1) Control block to enable ADC readout

The only change in this part is that the ADCs are read out every 50 µs during the final state of operation instead of the 150 µs mentioned earlier for the analog part.

e.2) Control block Voltage Over Threshold

The LDOs time is removed as the LDOs remain turned on continuously.

e.3) Control block: Principal control.

This part is modified to include the control for the new digital potentiometer that detects and provides the constant current due to the idle time of the ASICs. The other digital potentiometer that provides the current after each of the 13 current transitions (1 bunch train + 12 readout) has a very similar control as that for the analog part. The latter is described first.

Charging current:

This control block calculates the required position of the digital potentiometer in order to generate the minimum current to charge up the capacitors to a desired value (equal to the threshold voltage of the previous control block) after current transients due to train bunches or readout ASICs.
The setting of the digital potentiometer satisfies the following equation:

\[ D = \left( V_{12\text{bits}}(\text{bias}) - V_{12\text{bits}}(t) \right) \cdot 2 \cdot \frac{V_{\text{ADC,DD}}}{4096} \cdot \frac{C}{\Delta t} \cdot \frac{R_{\text{out}}}{10[\mu\text{A}]} \cdot \frac{256}{R_{AB}} \left( -\left( R_{W} + R_{\text{serie}} \right) \cdot \frac{256}{R_{AB}} \right) \cdot kte_1 \]  

(11)

where:
- \( D \) is the required position of the digital potentiometer whose value ranges from 0 to 255;
- \( V_{12\text{bits}}(\text{bias}) \) is the digital value that would be sampled by the ADC (PmodAD1) for the desired voltage on the capacitor;
- \( V_{12\text{bits}}(t) \) is the digital value measured and sampled by the ADC (PmodAD1) i.e. 50\,\mu\text{s} after train bunches and 350\,\mu\text{s} after readout;
- \( C \) is the value of the storage capacitors in half a ladder i.e. \( 12 \times 6.6 \, \mu\text{F} \) in our case;
- \( \Delta t \) is the remaining time to charge up the capacitors (20000\,\mu\text{s} - 350\,\mu\text{s}) or (20000\,\mu\text{s} - 50\,\mu\text{s});
- \( R_{\text{out}} \) is the 5\,\Omega resistor of the current source circuit of Figure 11;
- \( R_{AB} \) is the end-to-end resistance on pins of the digital potentiometer (AD5160);
- \( R_{\text{serie}} \) represents the series resistance connected to the Digipot AD5160 as defined in section 7.1A and in Figure 13;
- \( R_{W} \) is the wiper resistance contributed by the on resistance of the internal switch of the digital potentiometer (AD5160).

The code to compute this equation is shown at the top right part of Figure 44. During the initial mode of operation when the capacitors charge up, the initial charging current is close to 200 mA.

**Constant current:**

Knowing the voltage at two different times during the initial mode of operation, we can deduce the constant current required during the idle time of the ASICs. The equation to calculate the value to write in the potentiometer is the following:

\[ D = \left[ \frac{2 \cdot 3.3}{4095} \cdot \frac{C}{\Delta t} \cdot kte_1 \cdot 1400\mu\text{s} \right] \cdot \frac{255}{R_{\text{out}}} - \frac{200\mu\text{A}}{100\,\text{K}} \cdot \frac{1275}{R_{AB}} \]  

(12)

where:
- \( D \) is the required position of the digital potentiometer whose value ranges from 0 to 255.
- \( V_{12\text{bits}}(\text{now}) \) is the last digitized value of the sampled voltage.
- \( V_{12\text{bits}}(\text{past}) \) is the sampled voltage 1400\,\mu\text{s} earlier.
- \( C \) is the value of the storage capacitors in half a ladder i.e. \( 12 \times 6.6 \, \mu\text{F} \) in our case.
- \( \Delta t \) is the time between the two samples (1400\,\mu\text{s}).
- \( R_{\text{out}} \) is the 5\,\Omega resistor of the current source circuit of Figure 11.

The code to compute this equation is shown at the bottom right part of Figure 44.

**Final Simulink block diagram**

The final Simulink block diagram of the FPGA’s based current source to charge the digital part of the ASICs is shown in Figure 45.
function \[A_0,A_1,A_2,A_3\] = Multiplexer(Nper)

switch Nper
    case 2
        A_3=1;
        A_2=0;
        A_1=0;
        A_0=0;
    case 3
        A_3=1;
        A_2=0;
        A_1=0;
        A_0=1;
    case 4
        A_3=1;
        A_2=1;
        A_1=1;
        A_0=0;
    case 5
        A_3=1;
        A_2=1;
        A_1=1;
        A_0=1;
    case 6
        A_3=1;
        A_2=1;
        A_1=0;
        A_0=0;
    case 7
        A_3=1;
        A_2=1;
        A_1=0;
        A_0=1;
    case 8
        A_3=0;
        A_2=0;
        A_1=0;
        A_0=0;
    case 9
        A_3=0;
        A_2=0;
        A_1=1;
        A_0=0;
    case 10
        A_3=0;
        A_2=0;
        A_1=0;
        A_0=1;
    case 11
        A_3=0;
        A_2=0;
        A_1=1;
        A_0=1;
    case 12
        A_3=0;
        A_2=1;
        A_1=0;
        A_0=0;
    case 13
        A_3=0;
        A_2=1;
        A_1=0;
        A_0=1;
    otherwise
        A_3=0;
        A_2=1;
        A_1=1;
        A_0=1;
end
end

function \[R\text{Time}\] = Read\_Time(Nper,sw6)
if sw6
    switch Nper
        case 2
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 10000); 
        case 3
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 12000); 
        case 4
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 10000); 
        case 5
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 14000); 
        case 6
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 13000); 
        case 7
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 11000); 
        case 8
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 13000); 
        case 9
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 8000); 
        case 10
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 12000); 
        case 11
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 15000); 
        case 12
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 15000); 
        case 13
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 15000); 
        otherwise
            R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 15000); 
end
else
    R\text{Time}=xfix({xl\text{Unsigned}, 17, 0}, 15000); 
end

Figure 43: Simulink block of the Front-End control signals for the Digital current source. The Mosfet\_Driver output at the upper right corner turns on/off the MOSFETs in series with the 10.87Ω resistor to emulate the ASICs behaviour. The Mosfet\_Read\_neg output at the bottom right corner connects to the DEMUX which controls the MOSFETs in series with the 2.84Ω resistors. The 4 address signals are generated using the code at the block Multiplexer and are shown on the right. (The original high-resolution file of this figure can be found in [12].)
Figure 44: Simulink implementation of the Principal control block. Its main function is to calculate the required position of the digital potentiometers computed using Equations 11 and 12 which were programmed in the two Mcode blocks. The lower part of the block corresponds to the Overvoltage protection presented in Figure 28. (The original high-resolution file of this figure can be found in [12].)
Figure 45: Final Simulink block diagram of the FPGA’s based current source to charge the digital part of the ASICs. (The original high-resolution file of this figure can be found in [12].)
14 Measurements and results of the implemented current source to power the digital electronics

14.1 Setup

The digital dummy load previously designed (Figure 39) was powered by the current source to test its behaviour. The setup is illustrated in Figure 46. On the left, four voltage supplies power the FPGA and a custom-made PCB serves as the controlled current source. The latter is connected to the back-end cables (represented each by 16 µH inductors with a series resistance of 1 Ω), which are then followed by the aluminium flex cable that is connected to the PCB emulating the FE electronics. The current at the load is measured using a current probe and the voltage regulation is measured using a scope voltage probe on the last chip, which represents the most challenging case.

Figure 46: Digital test setup.

14.2 Results

14.2.1 Regulation and power dissipation

The regulated voltage at the last ASIC was measured for $t_{on}$, $t_r$, and $t_f$ equal to 20 µs, 1 µs and 1 µs, respectively. The digital voltage had a good regulation with a spike of 70 mV during the transitions as shown in Figure 47. This was achieved using capacitors at the input and output of the LDOs of 6.6 µF and 1 µF, respectively. The presumed occupancy for all the chips was 3%, so the readout lasted for 300 µs per chip. The scheme was also tested for varying readout times to emulate variations from chip to chip and from one bunch train to the next.

The power consumption resulting from this scheme was measured. The digital power losses are 35 mW/cm², which result in a total power consumption density below the required 50 mW/cm² when the 10 mW/cm² of the analog part is added. As for the analog part, most of the losses are due to the LDOs, which could be reduced at the cost of adding more material.
14 Measurements and results of the implemented current source to power the digital electronics

I

V

load

1.0 V

360 mA

ΔV = 70 mV

I

BE

100 mA

8 mA

180 mV

3 V

V

Cap

20 ms

3 V

1.8 V

Figure 47: Measurement of main signals (digital electronics).

14.3 Material budget calculation

The material budget for the implemented solution was calculated. Figure 48 (left) shows the results of the calculation using the technologies available today (0.04% \(X_0\)) and those expected in a few years’ time (0.015% \(X_0\)).

Figure 48: Material budget calculation for the ladder digital electronics powering circuitry: (left) with today’s technology and (right) with improved silicon capacitors technology expected to be available in few years’ time.

If we add today’s digital and analog material contribution, we get a total contribution of 0.104% \(X_0\) (Figure 49, left) which still exceeds the material target of 0.1% \(X_0\) to share with the mechanical structures. From the pie chart it can be noted that the silicon capacitors are the dominating contribution (almost 80%). If silicon capacitors increase their capacitance density as expected, the total material contribution of this power scheme will decrease to 0.043% \(X_0\) (Figure 48, right) which is in agreement
with the imposed restrictions.

Figure 49: Total material budget calculation for the ladder powering circuitry: (left) with today’s technology and (right) with improved silicon capacitors technology expected to be available in few years’ time.

15 Summary and conclusions

A low-mass pulsed-powering scheme is proposed for the future CLICpix ASIC of the CLIC vertex detectors. It has been specifically designed for the CLIC beam structure and to fulfill CLIC experiment requirements. It is comprised of regulation and silicon capacitors in the front-end, which are charged up using a back-end current supply of around 20 mA for the analog part and less than 200 mA for the digital part. The scheme fulfills the regulation requirements, having a voltage drop during the acquisition time of less than 20 mV for the analog electronics and 70 mV for the digital electronics. The average power consumption of 45 mW cm$^{-2}$ (10 mW cm$^{-2}$ for analog and 35 mW cm$^{-2}$ for digital) is below the target of 50 mW cm$^{-2}$ in the sensor area.

The material contribution in the ladder area of 0.104% $X_0$ per detection layer is already close to the target value. Future improvements of silicon capacitor technology are expected to decrease the material budget even further, to levels close to 0.04% $X_0$.

This solution is matching very well the CLIC requirements for which it has been designed.

References


