**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA**

**CELSO SUCKOW DA FONSECA - CCGELT**

**BACHARELADO EM ENGENHARIA ELETRÔNICA**

**LUCAS JEAN VIDAL DOS SANTOS SILVA**

**DESENVOLVIMENTO DE PROTÓTIPO DE PCB DE CONTROLE DO EXPERIMENTO CRE@AT**

**TRABALHO DE CONCLUSÃO DE CURSO**

**RIO DE JANEIRO**

**JANEIRO DE 2025**

**LUCAS JEAN VIDAL DOS SANTOS SILVA**

**DESENVOLVIMENTO DE PROTÓTIPO DE PCB DE CONTROLE DO EXPERIMENTO CRE@AT**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Engenharia Eletrônica, do CCGELT, do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca.

Orientador: Prof. Ulisses de Freitas Carneiro da Graça

**RIO DE JANEIRO**

**JANEIRO DE 2025**

Folha destinada à inclusão da **Ficha Catalográfica** (elemento obrigatório somente para teses e dissertações) a ser solicitada à Biblioteca Central do CEFET/RJ e posteriormente impressa no verso da Folha de Rosto (folha anterior).

Espaço destinado a elaboração da ficha catalografica sob responsabilidade exclusiva da Biblioteca Central do CEFET/RJ.

Dedico este trabalho aos meus pais, que sempre me apoiaram tanto nos estudos quanto na vida, e à minha irmã, que sempre esteve ao meu lado em todos os momentos.

agradecimentos

Agradeço ao pesquisador André Massafferri Rodrigues, pela orientação e todo o apoio dado em todas as fases e dificuldades desse projeto.

Ao professor Ulisses de Freitas Carneiro da Graça, pela orientação, e apoio na conclusão deste trabalho.

Ao técnico Fernando Souza, pelo suporte na montagem da *PCB* e em outras atividades no LabHEP.

Ao Centro Brasileiro de Pesquisas Físicas (CBPF) pelos recursos e infraestrutura disponibilizados para a execução deste projeto.

Aos meus colegas Diogo Ayres e Gabriel Fernandes, por me acompanharem em minha jornada no CEFET e no CBPF.

A única coisa que irá redimir a humanidade é a cooperação.

(RUSSELL, Bertrand, 1954).

Resumo

A passagem de raios cósmicos pela atmosfera tem sido proposta como mecanismo de influência na formação de nuvens. Para investigar essa relação com escrutínio científico, é importante coletar dados confiáveis acerca do fluxo de radiação cósmica através de longos períodos. O continente antártico se mostra favorável a esse tipo de experimento devido a suas características físicas e grande isolamento da ação humana. O projeto *CRE@AT* teve sua primeira fase implementada em 2014, medindo o fluxo de RCGs no módulo Criosfera. Este trabalho tem o objetivo de dar continuidade ao projeto, com a confecção de uma nova eletrônica de controle, aquisição e envio de dados, corrigindo problemas encontrados em fases anteriores do projeto. O projeto também abarca o desenvolvimento de novas unidades de *software* adaptadas às novas condições e a integração com os demais módulos do experimento, sejam eles antigos ou novos, assim como o teste do protótipo com respeito às funcionalidades desenvolvidas ou existentes.

**Palavras-chave:** nuvens. raios. cósmicos. Antártida. eletrônica.

abstract

The passage of cosmic rays through the atmosphere has been proposed to be a mechanism of influence over cloud formation. In order to investigate this connection with scientific scrutiny, it is important to collect reliable data on cosmic ray flux over long periods of time. The Antarctic continent is favorable to this type of experiment due to its physical characteristics and isolation of human activity. The CRE@AT project had its first phase implemented in 2014, measuring the flux of GCR at the Criosfera I module. The present work seeks to give continuity to the project, by manufacturing a prototype of an updated control, acquisition and transmission electronic circuit board addressing issues encountered in previous phases of the project. The project also includes the development of new software units adapted to the updated conditions and the integration with other modules of the experiment, both old and new, as well as testing the prototype in terms of developed or existing functionalities.

**Keywords:** clouds. rays. cosmic. Antarctic. electronics.

LISTA DE FIGURAS

[Figura 1 - Representação gráfica de um CAE [2] 17](#_Toc16736)

[Figura 2 - Variação na contagem do número de manchas solares (vermelho) e na intensidade relativa de raios cósmicos na estação Oulu (azul), ambos com uma média de 27 dias [5] 19](#_Toc298)

[Figura 3 - A linha azul mostra variações na nebulosidade global coletada pelo](#_Toc20902) *[International Satellite Cloud Climatology Project](#_Toc20902)*[. A linha vermelha é o registro de variações mensais nas contagens de raios cósmicos na estação de Huancayo, no Peru [8] 20](#_Toc20902)

[Figura 4 - Chegada do CREAT1 no módulo Criosfera I em 2014 [14] 22](#_Toc31212)

[Figura 5 - Diagrama de blocos da](#_Toc31254) *[PCB](#_Toc31254)* [24](#_Toc31254)

[Figura 6 - Montagem mecânica da tira cintilante com fibra](#_Toc30066) *[Wavelength Shifter](#_Toc30066)* [[14] 26](#_Toc30066)

[Figura 7 - Fibras óticas](#_Toc32589) *[WLS](#_Toc32589)* [[14] 27](#_Toc32589)

[Figura 8 - Fotomultiplicadora](#_Toc26355) *[SiPM](#_Toc26355)* [da série S13360 [15] 28](#_Toc26355)

[Figura 9 - Forma de onda do pulso de saída da S13360 [16] 28](#_Toc17629)

[Figura 10 - Circuito da fonte de tensão de polarização da](#_Toc23075) *[SiPM](#_Toc23075)* [29](#_Toc23075)

[Figura 11 - Circuito de amplificação e discriminação de um canal da](#_Toc7339) *[FEE](#_Toc7339)* [30](#_Toc7339)

[Figura 12 - Diagrama de blocos do](#_Toc23985) *[firmware](#_Toc23985)* [do](#_Toc23985) *[FPGA](#_Toc23985)* [31](#_Toc23985)

[Figura 13 - Janela da interface de configuração](#_Toc9621) *[PyControl](#_Toc9621)* [37](#_Toc9621)

[Figura 14 -](#_Toc30484) *[Ratio](#_Toc30484)* [x](#_Toc30484) *[threshold](#_Toc30484)* [do canal 1 39](#_Toc30484)

[Figura 15 -](#_Toc32088) *[Ratio](#_Toc32088)* [x](#_Toc32088) *[threshold](#_Toc32088)* [do canal 2 39](#_Toc32088)

[Figura 16 -](#_Toc903) *[Ratio](#_Toc903)* [x](#_Toc903) *[threshold](#_Toc903)* [do canal 3 40](#_Toc903)

[Figura 17 -](#_Toc18865) *[Ratio](#_Toc18865)* [x](#_Toc18865) *[threshold](#_Toc18865)* [do canal 4 40](#_Toc18865)

[Figura 18 -](#_Toc10834) *[Ratio](#_Toc10834)* [x](#_Toc10834) *[threshold](#_Toc10834)* [do canal 5 41](#_Toc10834)

[Figura 19 -](#_Toc23747) *[Ratio](#_Toc23747)* [x](#_Toc23747) *[threshold](#_Toc23747)* [do canal 6 41](#_Toc23747)

[Figura 20 -](#_Toc17484) *[Ratio](#_Toc17484)* [x](#_Toc17484) *[threshold](#_Toc17484)* [do canal 7 42](#_Toc17484)

[Figura 21 -](#_Toc30905) *[Ratio](#_Toc30905)* [x](#_Toc30905) *[threshold](#_Toc30905)* [do canal 8 42](#_Toc30905)

[Figura 22 -](#_Toc23710) *[Ratio](#_Toc23710)* [x](#_Toc23710) *[threshold](#_Toc23710)* [do canal 9 43](#_Toc23710)

[Figura 23 -](#_Toc30380) *[Ratio](#_Toc30380)* [x](#_Toc30380) *[threshold](#_Toc30380)* [do canal 10 43](#_Toc30380)

[Figura 24 -](#_Toc11977) *[Ratio](#_Toc11977)* [x](#_Toc11977) *[threshold](#_Toc11977)* [do canal 11 44](#_Toc11977)

[Figura 25 -](#_Toc21703) *[Ratio](#_Toc21703)* [x](#_Toc21703) *[threshold](#_Toc21703)* [do canal 12 44](#_Toc21703)

sumário

[1 INTRODUÇÃO 13](#_Toc863)

[1.1 MOTIVAÇÃO 13](#_Toc10614)

[1.2 Objetivos 14](#_Toc23752)

[1.3 Justificativa 14](#_Toc26817)

[1.4 Materiais e Métodos 14](#_Toc20220)

[1.5 Organização do Trabalho 16](#_Toc18955)

[2 Contexto do trabalho 17](#_Toc11020)

[2.1 O Projeto](#_Toc16170) *[CRE@AT](#_Toc16170)* [21](#_Toc16170)

[3 Desenvolvimento do Projeto 24](#_Toc5213)

[3.1 Módulo Detector 26](#_Toc31565)

[3.1.1 Tiras Cintilantes](#_Toc339) *[SciTile](#_Toc339)* [26](#_Toc339)

[3.1.2 Fibra Ótica](#_Toc13170) *[Wavelength Shifter](#_Toc13170)* [27](#_Toc13170)

[3.1.3 Fotomultiplicadora](#_Toc12917) *[SiPM](#_Toc12917)* [27](#_Toc12917)

[3.2 Eletrônica de](#_Toc3962) *[Front-End](#_Toc3962)* [29](#_Toc3962)

[3.3 Módulo](#_Toc23472) *[DAQ](#_Toc23472)* [30](#_Toc23472)

[3.3.1 Módulo](#_Toc14570) *[FPGA](#_Toc14570)* [31](#_Toc14570)

[3.3.2](#_Toc22684) *[Firmware](#_Toc22684)* [do ESP32 34](#_Toc22684)

[3.3.3](#_Toc7318) *[Software](#_Toc7318)* [de configuração](#_Toc7318) *[PyControl](#_Toc7318)* [36](#_Toc7318)

[4 Resultados 38](#_Toc19436)

[5 Conclusão 46](#_Toc24267)

[REFERÊNCIAS 47](#_Toc19321)

[APÊNDICE A -](#_Toc15802) *[Firmware](#_Toc15802)* [do](#_Toc15802) *[FPGA](#_Toc15802)* [49](#_Toc15802)

[APÊNDICE B -](#_Toc1414) *[Firmware](#_Toc1414)* [do ESP32 50](#_Toc1414)

[APÊNDICE C - Código da interface](#_Toc15400) *[PyControl](#_Toc15400)* [51](#_Toc15400)

[anexo A - Alguma coisa 52](#_Toc28966)

# INTRODUÇÃO

Dados experimentais mostram uma correlação entre o ciclo solar e variações significativas na temperatura terrestre. Estudos vêm indicando a influência da radiação cósmica sobre a formação de nuvens como um forte candidato para a causa desse fenômeno. O estudo dos raios cósmicos torna-se, então, relevante para aprofundar a compreensão dos ciclos climáticos terrestres.

A relação entre a passagem de raios cósmicos e a formação de núcleos de condensação de nuvens já vem sido estudada em câmara de nuvens como, por exemplo, no experimento *CLOUD*. Porém, ainda é limitada a existência de dados experimentais que permitam investigar o fenômeno em escala global. É, portanto, importante a construção de experimentos de monitoramento contínuo da atividade cósmica na atmosfera terrestre.

O continente antártico mostra-se um lugar propício para o estudo da relação entre a incidência de partículas cósmicas e o clima. Isso é devido a suas características físicas únicas, que causam efeitos favoráveis ao aumento no fluxo de radiação cósmica na superfície. Nesse contexto, o projeto *CRE@AT* (*Cosmic Ray Experiment at Antarctica*) foi concebido tendo como objetivo o monitoramento e estudo da incidência de raios cósmicos na Antártida.

Após o desenvolvimento e envio da versão piloto do projeto em 2014, foi percebida a necessidade de melhoramento de diversos aspectos do aparato experimental. O presente trabalho visa a construção de um protótipo da nova *PCB* de controle e aquisição de dados do experimento, e a adaptação do *firmware* responsável pelo seu funcionamento às novas necessidades do projeto.

## MOTIVAÇÃO

Este projeto é motivado pelo estudo da relação entre o fluxo de raios cósmicos na atmosfera terrestre e a formação de nuvens, através da contribuição com o experimento *CRE@AT*. O experimento visa o desenvolvimento de uma plataforma de medida de fluxo de raios cósmicos no continente Antártico, que funciona de maneira inassistida e contínua, permitindo o estudo desse fenômeno em escala macroscópica na atmosfera real.

## Objetivos

O objetivo deste projeto é desenvolver um protótipo da *PCB* que será usada nas próximas fases do projeto *CRE@AT*, realizando melhoramentos considerados necessários após as fases anteriores. Anteriormente, o microcontrolador, sensores, e diversos outros módulos eram interligados através de *jumpers* e fios entre si e à *PCB* da eletrônica de *Front-End*. A nova placa integrará todos esses módulos em uma *PCB* unificada de controle, aquisição e envio de dados (*DAQ*), com o circuito *Front-End* e outros módulos também anexados à placa. Também será desenvolvida a parte de *software* do experimento, incluindo os novos *firmwares* do microcontrolador e do *FPGA*, adaptando-as às novas características do experimento, e um *software* de configuração do experimento através de um computador.

## Justificativa

O desenvolvimento de uma *PCB* unificada para os módulos que compõe o *DAQ* foi considerado necessário para dar maior robustez a eletrônica do experimento, que, em sua configuração anterior, corria o risco de desconexão ou mal contato dos fios conectando os diversos módulos. A substituição das fotomultiplicadoras usadas nos detectores de *MaPMTs* para *SiPMs* também demanda uma série de adaptações na *Front-End*, que devem ser refletidas no circuito do *DAQ* e em todo o *software* associado ao experimento.

## Materiais e Métodos

O projeto foi desenvolvido com a colaboração de diversas pessoas no Laboratório de Física de Partículas de Altas Energias (LabHEP). Ele começou com a discussão das especificações e necessidades do experimento e da *PCB* entre o autor do presente trabalho e o pesquisador do CBPF, o físico André Massafferri Rodrigues. Definido o escopo das atividades que seriam desenvolvidas para o projeto, foi elaborada uma lista de componentes que seriam usados, os quais seriam importados pelo CBPF.

Então, começou-se o desenvolvimento do diagrama esquemático e do *layout* do circuito da *PCB*, utilizando o *software* *Altium Designer*. Uma primeira versão do *DAQ* foi desenvolvida pelo aluno, porém, no meio do projeto, foi decidido integrar o circuito de *Front-End* na mesma *PCB*. Com isso, uma versão atualizada foi desenvolvida pelo aluno do CBPF, Diogo Ayres, integrando o circuito que havia sido desenvolvido para a primeira versão apenas com o *DAQ*. O *layout* foi enviado para uma empresa de impressão de *PCBs*, e a placa foi então impressa. Com a chegada dos componentes comprados, esses foram soldados na placa pelo técnico do laboratório, Fernando Souza.

O *firmware* do *FPGA* foi desenvolvida usando o *software* *Quartus II*, combinando o design em diagrama de blocos e a elaboração de código em *VHDL*. Alguns blocos lógicos já desenvolvidos pelo professor do CEFET e orientador nesse projeto, Ulisses de Freitas Carneiro, puderam ser reutilizados ou adaptados para o novo *firmware*. A lógica de contagem de pulsos passou por uma primeira fase de validação através de simulações realizadas através da integração do *Quartus* com o *ModelSim*.

O *firmware* do microcontrolador ESP32 de controle do *DAQ* foi desenvolvida em C++ utilizando o *software* *Arduino IDE*. Foram usadas bibliotecas de leitura e escrita dos sensores utilizados na placa, disponíveis para importação através do mesmo *software*. Também foi desenvolvido, para funcionamento em conjunto com o microcontrolador, um *software* de configuração da placa pelo *PC*. Esse *software*, denominado *PyControl*, foi escrito em linguagem *Python* por meio do programa *PyCharm Community* e do *framework* *PyQt5*.

Com a placa montada e os *firmwares* desenvolvidas foram realizados diversos testes como de testes de continuidade, teste de alimentação, teste de amplificação de sinal, teste de discriminação de sinal, teste de leitura sensores, teste de contagem de pulsos, e teste do funcionamento da *PCB* como um todo. Esses testes foram realizados com recursos disponíveis no LabHEP, como bancadas, computadores, fontes de tensão, geradores de sinal, multímetro e osciloscópio, avaliando assim o funcionamento de cada módulo integrado na *PCB*.

## Organização do Trabalho

Este trabalho é organizado em quatro capítulos seguindo o capítulo de introdução. O Capítulo 2 (Contexto do Projeto) trata em maior detalhe do fenômeno físico que motiva o experimento que é tema deste projeto, os raios cósmicos e sua influência no clima. Também é dado um breve histórico do projeto *CRE@AT*. O Capítulo 3 (Desenvolvimento do Projeto) descreve os módulos que compõe o projeto como o detector, a eletrônica, e todo o *software* associado, explicando o design e funcionamento de cada um. O Capítulo 4 (Resultados) discute testes de eficiência realizados nos canais de detecção da *PCB* para validar seu funcionamento. O Capítulo 5 (Conclusão) faz uma síntese do projeto e dos resultados alcançados. Por último, seguem-se três apêndices contendo todo o código desenvolvido.

# Contexto do trabalho

Os raios cósmicos são partículas carregadas altamente energéticas, com velocidades próximas à da luz, que viajam pelo espaço. A maior parte deles são núcleos atômicos ionizados de diversos elementos, sendo sua composição, aproximadamente, 90% de hidrogênio (prótons), 9% de hélio (partículas alfa) e 1% de elementos mais pesados. A origem da radiação cósmica ainda não é completamente conhecida para toda sua faixa energética, porém sabe-se que a maior parte deles é proveniente da Via Láctea, os chamados raios cósmicos galácticos (RCG) [1].

A colisão de uma partícula cósmica, denominada partícula primária, com uma molécula da atmosfera causa uma série de interações nucleares e eletromagnéticas, produzindo diversas partículas secundárias. A cascata de colisões subsequentes faz com que o número de partículas produzidas cresça exponencialmente, dando origem a um disco de raio crescente ao se aproximar da superfície terrestre. Essa cascata é conhecida como Chuveiro Atmosférico Extenso (CAE) [2].



Figura 1 - Representação gráfica de um CAE [2]

A composição de um CAE é dada por, aproximadamente, 90% de elétrons, pósitrons e fótons, 9% de múons e 1% de partículas hadrônicas. A primeira interação produz predominantemente píons neutros e carregados. Os píons carregados tem um tempo de vida curto, de aproximadamente 10 nanossegundos, e geram múons ao decair. Grande parte das partículas geradas num chuveiro é absorvida pela atmosfera ou decai antes de atingir a superfície. Os múons, porém, devido a um longo tempo de vida e uma alta energia, atravessam a atmosfera em grande quantidade. O número de partículas de um CAE pode chegar a dezenas de milhares ao nível do mar, e os múons podem representar até 15% do total de partículas carregadas na superfície, sendo assim a partícula carregada mais abundante nessa altitude [2].

O fluxo de RCGs que chegam ao planeta é significativamente afetado pelo ciclo de atividade solar. O ciclo é caracterizado pela variação em número e área de manchas solares, tendo duração de aproximadamente 11 anos, quando ocorre a inversão dos polos magnéticos do Sol. As manchas solares são regiões na superfície do Sol que são mais escuras em relação à área circundante. Essas regiões apresentam temperaturas reduzidas, o que acarreta uma luminosidade menor, e se formam em áreas onde há fortes campos magnéticos, que limitam a transferência de calor por convecção para a superfície [3].

O vento solar é o fluxo de partículas carregadas emitidas da coroa do Sol em direção ao espaço. Esse plasma magnetizado carrega integrado em si o campo magnético heliosférico, que interage com as partículas cósmicas que adentram o sistema solar, alterando significativamente suas energias e trajetórias. Conforme a atividade do Sol se intensifica ao longo do seu ciclo, os campos magnéticos associados ao vento solar bloqueiam os raios cósmicos e, assim, diminuem sua incidência na heliosfera e, consequentemente, na Terra, dando origem a uma anticorrelação entre o ciclo de manchas solares e o fluxo de RCGs. Esse fenômeno é conhecido como Modulação Solar de Raios Cósmicos Galácticos [4].

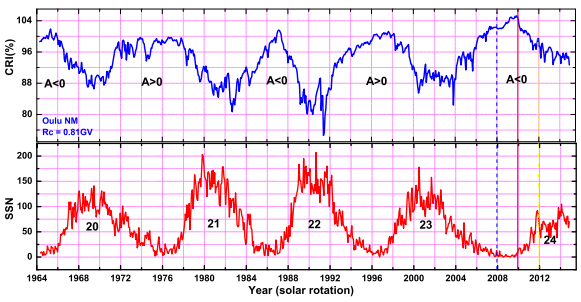


Figura 2 - Variação na contagem do número de manchas solares (vermelho) e na intensidade relativa de raios cósmicos na estação Oulu (azul), ambos com uma média de 27 dias [5]

Em 1991, Friis-Christensen e Lassen observaram uma correspondência entre variações na temperatura global e a duração do ciclo solar. Porém, a mudança na irradiação solar total ao longo de um ciclo, de aproximadamente 1%, seria insuficiente para atribuir à excursão de temperatura observada [6].

Em seguida, os pesquisadores Heinrich Svensmark e Eigil Friis-Christensen, chamaram atenção para uma forte correlação entre a cobertura global de nuvens e o fluxo de radiação cósmica. Em 1997, publicaram “*Variation of cosmic ray ﬂux and global cloud coverage - a missing link in solar-climate relationships*”, onde propuseram a influência dos RCGs na formação de nuvens como um fator de influência da atividade solar sobre o clima terrestre. Ao passarem pela atmosfera, os raios cósmicos podem ionizar átomos no ar, e um aumento na concentração de íons pode, por hipótese, facilitar a formação de núcleos de condensação de nuvens [7].



Figura 3 - A linha azul mostra variações na nebulosidade global coletada pelo *International Satellite Cloud Climatology Project*. A linha vermelha é o registro de variações mensais nas contagens de raios cósmicos na estação de Huancayo, no Peru [8]

O efeito da nebulosidade na temperatura da Terra se dá de duas maneiras: a reflexão da radiação solar de volta para o espaço gera um resfriamento, e a reflexão da radiação infravermelha emitida pela superfície gera um aquecimento. O efeito dominante depende do tipo de nuvem envolvida. Nuvens de baixa altitude geram uma diminuição da temperatura, e nuvens de alta altitude geram um aumento. A passagem de RCGs está correlacionada com um aumento de nuvens de baixa altitude, e, portanto, tem como resultado o resfriamento do planeta. Dessa forma, a influência do ciclo solar no clima acontece da seguinte forma: conforme a intensidade do vento solar diminui, a incidência de raios cósmicos aumenta, ocasionando uma maior cobertura de nuvens, que então gera uma queda na temperatura [7][9][10].

As nuvens são compostas de uma massa visível de gotículas de água ou cristais de gelo microscópicos suspensos no ar. Elas se formam quando o ar se torna saturado de vapor d’água, ao ser resfriado abaixo de seu ponto de orvalho, ou ao receber umidade adicional de uma massa de ar adjacente. Além disso, é necessária a presença de aerossóis atmosféricos, partículas sólidas ou líquidas suspensas no ar, que podem agir como núcleos de condensação de nuvens, superfícies sobre os quais o vapor pode condensar.

O mecanismo físico que explica a ligação entre maior ionização do ar e a formação de núcleos de condensação é tema de debates ainda hoje. Um processo proposto seria que íons atmosféricos podem acumular ligantes orgânicos, formando agrupamentos moleculares carregados, que possuem estabilidade e taxa de crescimento maiores que suas contrapartes neutras, assim gerando aerossóis com maiores chances de crescer até um tamanho crítico para constituir núcleos de condensação [11][12][13].

A implementação de plataformas de monitoramento de raios cósmicos na atmosfera real, visando estudar a possível relação entre os RCG e a cobertura global de nuvens, é um tema de pesquisa de grande interesse. Nesse contexto, foi concebido o projeto *CRE@AT*.

## O Projeto *CRE@AT*

O projeto *CRE@AT* (*Cosmic Ray Experiment at Antarctica*) têm como objetivo a criação de um experimento de monitoração da radiação cósmica no continente antártico, visando estudar a influência dos RCGs no clima, em especial, através de sua possível ação sobre a formação de nuvens. Com esse fim, o experimento visa obter medidas de fluxo de raios cósmicos secundários, principalmente os múons.

O continente antártico possui diversas características que favorecem o estudo da radiação cósmica. Sua atmosfera reduzida e o campo magnético ortogonal à superfície aumentam significativamente o fluxo de RCGs em comparação às demais regiões do globo. A espessa camada de gelo diminui expressivamente a incidência de radiação proveniente de elementos pesados no solo. Além disso, é uma região remota, apresentando um nível mínimo de interferência da atividade humana.

O CREAT1, a versão piloto do projeto, que foi tema de projeto final do professor Ulisses Carneiro, então aluno de graduação do CEFET, foi enviado à Antártida em outubro de 2014, tendo em vista verificar a viabilidade contínua e autônoma do experimento em um ambiente hostil. As medidas de fluxo obtidas no módulo Criosfera foram comparadas com medidas realizadas no CBPF, o que mostrou, como esperado, um fluxo maior na Antártida em comparação ao Brasil.



Figura 4 - Chegada do CREAT1 no módulo Criosfera I em 2014 [14]

Essa versão do experimento coletou dados ao longo de alguns anos, permitindo identificar diversos pontos fracos a serem corrigidos em versões futuras. Em especial, foi considerado necessário um *upgrade* do detector. Foi iniciada a construção do detector do CREAT2, contando com o dobro da área de incidência de partículas do seu antecessor e uma organização que permite um melhor cálculo de eﬁciência de seus planos de detecção. O *firmware* do detector foi reformulado, e foi desenvolvido um novo *hardware* de armazenamento de dados, contendo sensores de temperatura, pressão e campo magnético.

Um dos problemas encontrados no decorrer dos anos foi uma variação na eﬁciência dos conjuntos de detecção no decorrer dos anos, principalmente nas missões que ocorreram em 2016 e 2017. Descobriu-se que isso se devia à dilatação térmica do material, desalinhando sua ótica e comprometendo sua eﬁciência. Para corrigir esse problema, decidiu-se trocar a fotomultiplicadora *MaPMT* (*Multianode Photomultiplier Tube*) por uma *SiPM* (*Silicon Photomultiplier*), que possui dimensões muito menores, e permite um melhor encaixe com os cintiladores, garantindo o alinhamento ótico insensível a mudanças de temperatura.

A mudança de fotomultiplicadora demanda a reformulação da eletrônica de *Front-End* (*FEE*) e também do *firmware* associada. O presente trabalho visa implementar essas mudanças numa *PCB* unificada de *FEE* e *DAQ* (*Data Acquisition*), produzindo um protótipo da placa que permitirá o avanço das fases seguintes do experimento.

# Desenvolvimento do Projeto



Figura 5 - Diagrama de blocos da *PCB*

O sistema de detecção de múons conta com 3 partes principais: o detector, a eletrônica de *Front-End* (*FEE*) e a unidade de aquisição de dados unificada, o *DAQ* (*Data Acquisition*), com a *FEE* e o *DAQ* integrados em uma única *PCB*.

A placa possui dois módulos conversores *DC-DC*, um com saída de 5 V e outro com saída de 3,3 V, que alimentam todos os demais módulos e ICs com suas respectivas tensões de entrada. Os conversores são alimentados por uma entrada de tensão da *PCB*, que pode receber de 12 até 24 V. Foram adicionados sensores de corrente na saída dos conversores para monitoramento da alimentação da placa pelo *DAQ*. A *PCB* conta ainda com módulos *DAC* (*Digital Analog Converter*), responsáveis por controlar o valor diversas tensões do circuito, como tensões de offset dos amplificadores operacionais e tensões de referência dos discriminadores. O controle de tensão dos *DACs* é realizado pelo *firmware* do *FPGA*.

O detector é composto de um conjunto de tiras cintilantes plásticas, os cintiladores. Cada tira conta com um filamento de fibra ótica *WLS* (*Wavelength Shifter*) acoplado em seu interior e uma fotomultiplicadora *SiPM* (*Silicon Photomultiplier*) em sua extremidade. Os cintiladores são feitos de um material fluorescente, cujos átomos emitem fótons ao serem ionizados por uma partícula carregada que o atravessa, nesse caso, os múons a serem detectados. O sinal luminoso é então guiado até a extremidade da tira, chegando à *SiPM*, onde é transformado em um sinal elétrico correspondente. Na placa existe um módulo de alta tensão, responsável por gerar a tensão de polarização reversa das fotomultiplicadoras.

O sinal gerado é então tratado pela eletrônica de *Front-End*. O pulso de saída da *SiPM* é primeiro acoplado e amplificado, chegando então ao discriminador. Esse é basicamente um comparador, com uma tensão de referência previamente escolhida. Quando o sinal amplificado atinge esse limiar, a saída do comparador vai para alto. Isso é feito com o intuito de discernir uma detecção real, ou seja, um sinal com amplitude suficiente para caracterizar uma excitação do cintilador por uma partícula, dos sinais de ruído presentes no circuito. A saída do comparador é então alimentada em uma entrada da unidade *FPGA* no *DAQ*, responsável pela contagem de pulsos.

O *DAQ* abrange sensores de temperatura, pressão, umidade, campo magnético e acelerômetro (para verificar a ortogonalidade do experimento com a superfície) para monitoramento das condições de contorno do experimento, leitura de tensões e correntes relevantes no circuito, relógio digital, *GPS*, um módulo de comunicação *Ethernet*, um módulo para leitura e escrita de cartão *microSD* para armazenamento de dados, um microcontrolador ESP32 e um *FPGA*.

O *FPGA* recebe os pulsos digitais que vêm da *FEE* em suas entradas, sendo responsável pela contagem de detecções individuais em cada canal, assim como detecções simultâneas entre dois ou mais canais. Cada uma dessas contagens é realizada por um tempo predeterminado de aquisição (*TAQ*). Ao fim desse tempo, o bloco de contagens é então enviado para um microcontrolador.

O microcontrolador é responsável pela aquisição, processamento, armazenamento e envio de todos os dados obtidos no experimento, como leituras de sensores e, principalmente, as contagens de pulsos. Ele se conecta a um *PC* via *USB*, recebendo e enviando variáveis de configuração e controle através do *software* desenvolvido *PyControl*.

O *firmware* executado pelo microcontrolador se comunica continuamente com o *FPGA*, esperando confirmação de que o último bloco de contagens foi processado. Quando isso ocorre, essas contagens são lidas e compiladas com os dados dos sensores e demais CIs de interesse a intervalos regulares, sendo elaborado um *dataframe* que é gravado no *microSD* e enviado para um servidor no CBPF em tempo real, via módulo *Ethernet*. O microcontrolador também monitora a cada ciclo a chegada de novas configurações enviadas pelo *PyControl* por meio do *PC*, que pode ser monitorado remotamente através do *software* *TeamViewer*.

## Módulo Detector

### Tiras Cintilantes *SciTile*

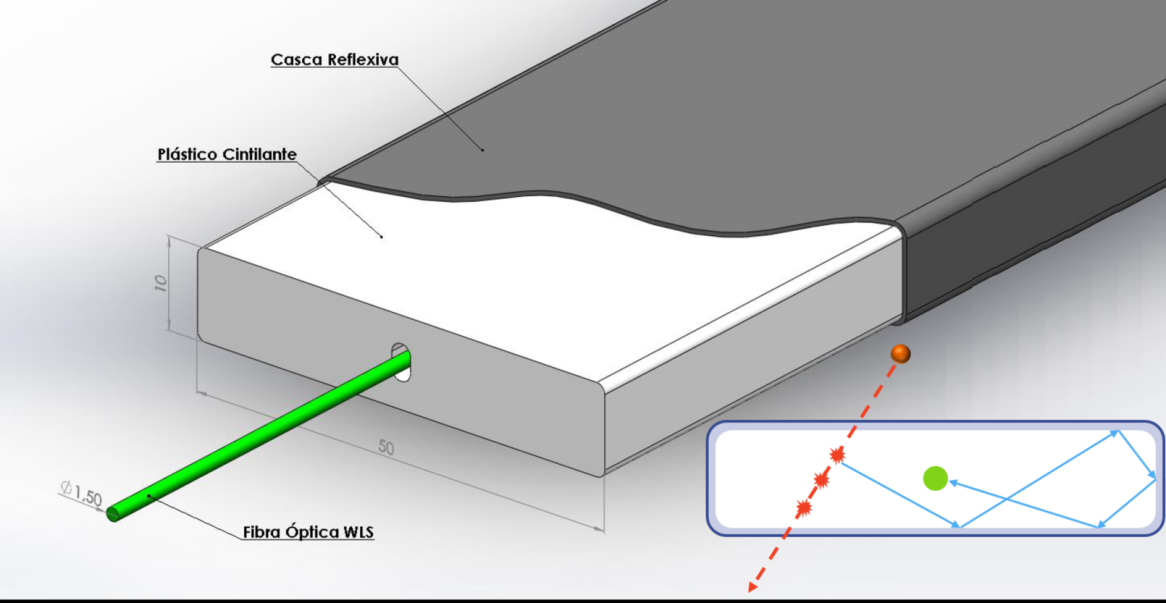


Figura 6 - Montagem mecânica da tira cintilante com fibra *Wavelength Shifter* [14]

Os cintiladores são materiais que exibem o fenômeno de cintilação, a emissão de luz mediante a excitação por radiação ionizante. Quando uma partícula carregada atravessa um cintilador e interage com elétrons de seus átomos, parte de sua energia é depositada nesses elétrons, elevando-os a níveis orbitais de maior energia, porém instáveis. Ao retornarem aos orbitais originais, a energia adicional é liberada através da emissão de um fóton.

As tiras cintilantes plásticas usados no detector do projeto foram desenvolvidas pelo *Fermilab* (*Fermi National Accelerator Laboratory*) e são feitas de um material fluorescente, que emite fótons com comprimento de onda na faixa do azul ao ser excitado por um raio cósmico. As tiras possuem um orifício ao longo de seu comprimento para a acoplação da fibra ótica *WLS*.

Os cintiladores são cobertos por uma casca opaca, que bloqueia a incidência da luz externa. O interior da casca é revestido de um material reflexivo, com o objetivo de refletir os fótons emitidos pelo processo de cintilação, aumentando a probabilidade de que eles atinjam e sejam conduzidos pela fibra *WLS*.

### Fibra Ótica *Wavelength Shifter*



Figura 7 - Fibras óticas *WLS* [14]

A fibra ótica *Wavelength Shifter* (*WLS*) acoplada no cintilador é do modelo Y-11(175)MSJ e é produzida pela KURARAY. Ela absorve os fótons de maior energia, no espectro do azul, emitidos pelas tiras cintilantes, e reemite múltiplos fótons de menor energia, no espectro do verde, conduzindo-os até a fotomultiplicadora *SiPM*. O maior número de fótons aumento a eficiência do detector, já que muitos se perdem no caminho até a *SiPM*. Além disso, a fotomultiplicadora apresenta maior rendimento na frequência do verde.

### Fotomultiplicadora *SiPM*

A fotomultiplicadora *SiPM* (*Silicon Photomultiplier*) é um fotodetector de estado sólido que, em resposta à absorção de um fóton, produz um pulso de corrente com duração na ordem de dezenas de nanossegundos. Ela é composta de uma matriz de *pixels*, sendo cada *pixel* um fotodiodo de avalanche operando em modo Geiger, com a tensão de polarização reversa um pouco acima da tensão de ruptura. Isso torna a *SiPM* sensível suficiente para detectar a passagem de um único fóton.



Figura 8 - Fotomultiplicadora *SiPM* da série S13360 [15]

Quando um fóton acerta um *pixel*, é produzida uma avalanche de portadores de carga, podendo chegar a um número de 105 a 106, gerando um pulso de corrente detectável. Os *pixels* são arranjados num circuito em paralelo uns com os outros. Com isso o sinal na saída da fotomultiplicadora é a soma dos sinais em cada *pixel* e é proporcional ao número de fótons detectados, permitindo uma leitura de alta precisão.



Figura 9 - Forma de onda do pulso de saída da S13360 [16]

O ganho da fotomultiplicadora é o número de portadores de carga gerados em uma descarga, e depende linearmente da diferença entre a tensão de polarização VBIAS e a tensão de ruptura VBR (*breakdown voltage*):

A tensão de ruptura varia com a temperatura, o que pode afetar o ganho. Para garantir um ganho constante é necessário operar a *SiPM* com um VBIAS que garanta um constante apesar das variações de temperatura.

Nesse projeto, são usadas *SiPMs* da série S13360 em conjunto com uma fonte de tensão C11204-02, ambos da Hamamatsu. A fonte fornece a tensão de polarização da fotomultiplicadora, na faixa de 40 V a 90 V, e é programável via protocolo serial *UART*. Ela contém uma função de compensação de temperatura, que ajusta sua tensão de saída de acordo com as variações de temperatura, buscando manter o ganho da *SiPM* constante. Para isso, ela monitora o output analógico do sensor de temperatura LM94021.



Figura 10 - Circuito da fonte de tensão de polarização da *SiPM*

## Eletrônica de *Front-End*

A amplitude e largura dos pulsos advindos da *SiPM* são muito pequenos para detecção pelas portas de entrada do *FPGA*. Para isso, o circuito de *Front-End* (*FEE*) é responsável por amplificar e alongar esse sinal. Além disso, para diferenciar o pulso proveniente da detecção de uma partícula de sinais de ruído, que podem ser de origem térmica, da fonte de alimentação ou de interferência externa, a saída do circuito amplificador passa por um discriminador, cuja saída é então alimentada no módulo contador, a unidade *FPGA*. Segue abaixo o esquemático do circuito:



Figura 11 - Circuito de amplificação e discriminação de um canal da *FEE*

O circuito possui dois estágio de amplificação, sendo o ganho total o produto do ganho de cada estágio. O discriminador se trata basicamente de um comparador ADCMP600. Quando o sinal de entrada ultrapassa a tensão de referência (*threshold*) na porta Vn, a saída vai para alto, transformando o pulso analógico em um pulso digital.

A tensão de referência pode ser gerada por um circuito manual de *threshold*, que usa um trimpot como divisor de tensão, ou a partir de uma saída de um dos módulos *DAC*, representada pela tensão DACthDisc no esquemático. A escolha da tensão é feita através da ligação via jumper entre o terminal do header conectado ao discriminador e o terminal com a tensão escolhida.

## Módulo *DAQ*

A unidade de aquisição de dados possui dois componentes principais: a unidade *FPGA* e o microcontrolador ESP32. Ela também abrange sensores de temperatura, pressão, umidade, campo magnético e acelerômetro (para verificar a ortogonalidade do experimento com a superfície) para monitoramento das condições de contorno do experimento, conversores analógico-digitais (*ADC*) para a leitura de tensões relevantes no circuito, relógio digital, *GPS*, um módulo de comunicação *Ethernet* e um módulo para leitura e escrita de cartão *microSD*. Os sensores de corrente na saída dos módulos *DC-DC* têm como output uma tensão proporcional à corrente lida, e essa tensão é alimentada em uma entrada de um *ADC*, e pode então ser lida e convertida para o valor de corrente pelo código do microcontrolador.

O *FPGA* desempenha o papel de módulo contador de eventos. Ele recebe os pulsos digitais dos 12 canais da *Front-End*, oriundos da amplificação, alongamento e discriminação dos pulsos analógicos gerados pelas *SiPM* de cada canal. Após cada período de contagem, esses dados são enviados para o microcontrolador.

O ESP32 age como um módulo de controle central do experimento. Ele é responsável pela aquisição, processamento, armazenamento e envio de todos os dados obtidos no experimento, como leituras de sensores e, principalmente, as contagens de pulsos. Ele se conecta a um *PC* via *USB* e se comunica com o programa *PyControl*, recebendo e enviando leituras de dados, status do experimento e variáveis de configuração e controle.

### Módulo *FPGA*



Figura 12 - Diagrama de blocos do *firmware* do *FPGA*

O módulo contador é implementado como um circuito programável através do *firmware* da unidade *FPGA*. Ele é responsável pela contagem do número de pulsos digitais vindos da eletrônica de *Front-End*, da saída do discriminador. Cada um desses pulsos corresponde a um sinal analógico na saída de uma *SiPM* que é processado por um dos doze canais da *FEE*.

Os cintiladores são organizados em três grupos de quatro tiras cintilantes. Os canais de 1 a 4 formam o grupo A, os canais de 5 a 8 formam o grupo B, e os canais de 9 a 12 formam o grupo C. O *firmware* realiza a contagem de pulsos individuais em cada um dos canais e também realiza contagens de todas as possíveis coincidências duplas, triplas e quádrupla entre os canais de um mesmo grupo.

O *FPGA* deriva seu sinal de clock de um cristal oscilador de 50 MHz, representado pelo nome LOCALCLK. No *firmware*, é utilizado um módulo *PLL*, que a partir do LOCALCLK deriva como output um sinal de 100 kHz, usado como clock do bloco de controle dos módulos *DAC* do circuito, e um sinal repetido de 50 MHz, que tem o papel de sinal de clock de todos os demais blocos lógicos.

Devido a instabilidades no sinal da fotomultiplicadora e ruídos de chaveamento no discriminador é possível que um único pulso vindo do detector gere a detecção de diversas bordas de subida no pino de entrada correspondente do *FPGA*. Para evitar que isso aconteça, cada um dos doze canais digitais de entrada é alimentado no bloco denominado *Gate Timer*. Quando ocorre uma borda de subida em um desses canais, a saída correspondente nesse bloco é mantida em alto por um tempo configurável chamado nT\_GATE. A duração do gate tem o valor de um número inteiro de ciclos de clock que vai de 1 a 255.

Os doze sinais de saída do *Gate Timer* então chegam no bloco de coincidências. Esse bloco é responsável por determinar as coincidências de detecção em canais de um mesmo grupo. Isso é implementado alimentando cada sinal de um grupo em portas AND organizadas de acordo com cada uma das combinações possíveis de dois, três ou quatro canais. Assim, por exemplo, para o grupo A, temos os canais A1, A2, A3 e A4. Com isso, as combinações possíveis são:

* Combinações duplas: A1A2, A1A3, A1A4, A2A3, A2A4 e A3A4;
* Combinações triplas: A1A2A3, A1A2A4, A1A3A4 e A2A3A4;
* Combinação quádrupla: A1A2A3A4;

Os grupos B e C apresentam combinações análogas entre seus canais. Assim temos 6 combinações duplas, 4 combinações triplas e uma quádrupla para cada grupo de detectores.

Finalmente, os pulsos individuais e os pulsos de coincidência são direcionados para a entrada do bloco de contagem. Com 11 combinações mais 4 pulsos individuais, são 15 contagens por grupo, num total de 45 contagens. Cada contagem é armazenada em 3 bytes de memória, com todas as contagens ocupando em todo 135 bytes.

Quando é detectada uma borda de subida em uma das 45 entradas, a contagem correspondente é incrementa em uma unidade. As contagens vão sendo atualizadas pela duração do ciclo de contagem atual. Uma entrada de RELEASE é responsável por sinalizar o fim do ciclo em curso. Quando RELEASE sofre uma borda de descida, cada valor de cada contagem é descarregado em saídas correspondentes do bloco, para depois serem armazenadas no bloco de memória do *firmware*. As contagens então voltam para zero, e inicia-se um novo ciclo.

O bloco de memória é formado pelos bancos A e B. Ele é acessível através de um bloco de comunicação serial via protocolo *I2C* para o usuário que, no caso deste experimento, é o microcontrolador ESP32. O banco A é acessível apenas para leitura, sendo responsável por armazenar as 45 contagens de pulsos vindas do bloco de contagem. Já o banco B é acessível para leitura e escrita, e armazena variáveis de configuração e de status, que são usadas para mediar e sincronizar a comunicação entre o usuário e o *FPGA*.

Cada banco possui 255 bytes de memória, podendo ser endereçados por um único byte. Assim os endereços de memória vão de 0x00 a 0xFF. Os cinco endereços mais altos, de 0xFF a 0xFB, são comuns aos dois bancos, representando, nessa ordem, a versão do *firmware*, o reset de *software*, a seleção de banco, e os endereços de FLAG e ACK. O banco A é completamente reservado para contagens de pulsos. O banco B armazena variáveis para configuração do período de um ciclo de contagem de pulsos, o *TAQ* (time of acquisition), o número de ciclos de clock em um período do *Gate Timer*, o nT\_GATE, o reset do bloco de controle dos módulos *DAC*, e o controle dos *LED*s do módulo *FPGA* para testes e cada uma das tensões a serem gravadas nos *DAC*s.

Os endereços de FLAG e ACK servem para sincronizar a comunicação entre o *FPGA* e o microcontrolador. Quando um ciclo de aquisição, ou de contagem, chega ao fim, o valor do FLAG se torna 0x01. O ESP32, que monitora continuamente o FLAG, interpreta isso como sinal de que as últimas contagens estão prontas para serem lidas. O ACK é um endereço reservado para escrita pelo microcontrolador, e é setado para 0x01 quando todas as contagens já foram lidas. Com isso, o *firmware* seta o FLAG e, em seguida, o ACK, ambos de volta para 0x00.

A arbitragem da duração do ciclo de aquisição, da atualização dos valores de nT\_GATE e de *TAQ*, e da comunicação com o microcontrolador é feita pelo bloco de controle. Ao começo de um novo ciclo, ele lê da memória o valor mais recente configurado para o *TAQ* e o nT\_GATE, sendo esse enviado para o *Gate Timer*. Então, um bloco interno de cronometragem marca a passagem do número de segundos desde o início do ciclo. Quando esse número se iguala ao valor do *TAQ*, o sinal de saída de RELEASE vai para alto, fazendo o bloco de contagem descarregar as contagens no banco A da memória. Por fim, o endereço de FLAG é setado, o RELEASE volta para zero, e um novo ciclo se inicia.

O bloco de controle dos *DAC*s grava cada uma das 8 tensões de saída de cada um dos 5 ICs de *DAC* através do envio de mensagens seriais via protocolo *SPI*. Cada uma das tensões é gravada na ordem que se encontram na memória, e são representadas cada uma por um byte, tendo um valor inteiro vDIG de 0 a 255. Sendo vREF a tensão de referência alimentada no pino REF dos módulos *DAC*. O valor de tensão correspondente é dado pela fórmula:

No circuito, a tensão de referência para todos os *DAC*s é de 5 V, portanto, eles proveem saídas de tensão de 0 a 5 V.

### *Firmware* do ESP32

O microcontrolador age como controlador mestre de todos os demais módulos no *DAQ*. Assim que o experimento é ligado, o *firmware* do ESP32 realiza a inicialização de seus módulos de comunicação serial, os quais são usados para comunicação com os sensores, módulos *ADC*, módulo de cartão *SD*, módulo *Ethernet* e com a unidade *FPGA*.

A temporização do experimento é realizada em conjunto pelo relógio interno do ESP32, pelo módulo *Real-Time Clock* (*RTC*) e pelo módulo *GPS*. Quando a *PCB* é ligada, o microcontrolador registra o horário *UTC* obtido via satélite pelo *GPS*. Esse horário é então usado como referência para medida do tempo a cada instante do experimento pelo relógio interno. Além disso, esse tempo é gravado no *RTC*, que é alimentado por uma bateria própria, e serve para garantir que o sincronismo temporal não seja perdido mesmo que a placa venha a ser desligada, por exemplo, por uma possível falha na alimentação energética do experimento. Isso é importante, já que a fixação do sinal de satélite pelo *GPS* pode levar até dezenas de minutos até ser obtida.

Para garantir que não haja drift na marcação do tempo depois de um período extenso desde a última fixação por satélite, considerando que o experimento deverá permanecer ativo sem interrupção por meses ou anos, o horário é corrigido periodicamente pela leitura do *GPS*. Esse período é programável pelo usuário, podendo ter um valor na ordem de dias.

Os dados adquiridos, processados, salvos e enviados pelo microcontrolador consistem nos valores de contagens vindos do *FPGA*, mais os valores de leituras dos sensores, referidos como dados de *slow control*. Esses dados são organizados em um *dataframe* que é posteriormente salvo no cartão o *microSD* e enviado para um servidor no CBPF via *Ethernet*.

O *dataframe* é dividido em *frames* e *blocks*. Cada *block* consiste na coleção de todas as contagens de pulsos feitas dentro da duração de um período de aquisição (*TAQ*), um tempo de início e um índice de bloco. O *frame* é composto por um conjunto de *blocks*, tendo a duração de um *TAQ* vezes o número de blocos que possui. Ele também agrega o conjunto de medidas de *slow control* feitas ao longo de sua duração. Cada medida, por exemplo, de temperatura, é feita uma vez no início de cada bloco, e ao final é armazenada a média dessas medidas. Isso é feito devido às medidas de *slow control* serem quantidades de baixa variabilidade ao longo do tempo de medição característico de um *frame*, da ordem de minutos. Por fim, a cada *frame* são associados um tempo de início, um índice de *frame*, e um conjunto de configurações, de forma que, quando novas configurações são escritas no microcontrolador, essas são aplicadas somente ao início de um novo *frame*.

Assim que o experimento é ligado, o microcontrolador entra em uma fase de setup seguida do ciclo normal de execução. No setup, o *firmware* inicializa variáveis de código, estabelece a comunicação com os sensores, carrega as configurações mais recentes armazenadas no cartão *SD*, escreve as configurações do *FPGA*, escreve as configurações do módulo de alta tensão, e ajusta a temporização do experimento conforme descrito acima.

Ao entrar no ciclo normal de execução um novo *frame* é construído, com as últimas configurações sendo associadas a ele, e seu cabeçalho é escrito no cartão *SD*. O endereço de FLAG do *FPGA* é lido continuamente, esperando a sinalização de término do ciclo de contagem atual. Quando a FLAG vai para alto, é enviado um ACK. São lidas todas as contagens armazenadas no banco A do *FPGA*, é registrado o tempo associado à recepção desse block de contagens, e são feitas as leituras de *slow control* correspondentes. O bloco recebido é armazenado no *frame* atual e escrito no cartão *SD*. Esses passos se repetem até que o *frame* atual seja povoado com o número de blocos por *frame* configurado. Quando o último bloco é recebido, é feita média das leituras de *slow control*, e essas são escritas no *SD*. O *frame* inteiro é então enviado via *Ethernet*, um novo *frame* é construído, e repete-se o ciclo.

No ciclo, há ainda uma segunda fase, quando a FLAG está em baixo. É então aproveitado o tempo para verificar a existência se mensagens seriais vindas do *PC*. Essas mensagens podem incluir novas configurações, ou pedidos de envio de dados, e serão descritas em melhor detalhe na próxima seção.

### *Software* de configuração *PyControl*

A interface de configuração *PyControl* foi desenvolvida com o intuito de modificar em tempo real variáveis utilizadas pelo *firmware* do microcontrolador. Essas podem incluir valores como o de nT\_GATE, de tempo de aquisição (*TAQ*), de período de update do tempo a partir do *GPS* e de cada uma das tensões de output dos módulos *DAC*. O programa em si é configurável, com suas variáveis em display, que podem ter seu valor modificado e gravado no ESP32, podendo ser escolhidas através de um arquivo de texto usado pelo programa, que contem uma lista das variáveis utilizadas, com seus nomes e valores padrão.

Foram implementados os comandos *Request Config* e *Send Config*, que podem ser ativados pelos botões de mesmo nome. O *Request* solicita ao microcontrolador o último valor gravado para cada uma das variáveis, que após recebidos são mostrados na coluna *Last Recorded*. O *Send*, por sua vez, envia em ordem cada uma das variáveis em *display*, com seus valores na coluna *Last Configured*. O microcontrolador identifica cada variável pelo nome em *display*, portanto é indispensável que os nomes usados reflitam aqueles gravados no *firmware* do ESP32.

Para modificar o valor *Last Configured* de uma variável, basta escrevê-lo na caixa de diálogo na mesma linha, e então pressionar *Enter*. O novo valor será mostrado na coluna, e também será enviado individualmente para o microcontrolador. Há ainda uma caixa de diálogo abaixo da tabela de variáveis, servindo para enviar *strings* de comando personalizadas, seja para testes, ou para a implementação de novas *features*. Na figura abaixo é mostrada a interface do *PyControl* configurada para mostrar algumas variáveis descritas no texto.



Figura 13 - Janela da interface de configuração *PyControl*

# Resultados

A caracterização dos canais da *Front-End* e do algoritmo de contagem de pulsos foi feita através da injeção de pulsos nos canais por um gerador de função, e a medição da razão entre o número de contagens e o número de pulsos injetados por ciclo de aquisição, caracterizando uma figura de mérito do funcionamento do canal. A razão definida acima, nomeada apenas por *ratio* daqui emdiante*,* foi mensurada para diversos valores configurados de tensão de *threshold* do discriminador, gerando assim um gráfico de “*ratio* x *threshold”* para cada canal.

A *ratio* por *threshold* foi medida para um período de aquisição (*TAQ*) de um segundo, injetando pulsos numa frequência de 1 MHz, resultando num total de um milhão de pulsos injetados por período. Foram escolhidos pulsos com amplitude de 100 mV e largura de 10 ns, valores próximos dos típicos para os pulsos gerados pelas *SiPM* usadas nos detectores, como visto na Figura 9. Foram feitas medidas para valores de *threshold* de 0 a 2,5 V, com um passo de 0,1 V. A *ratio* para cada valor de *threshold* foi medida para 10 períodos de aquisição, sendo então calculada a média e o desvio padrão. Com isso, foram elaborados gráficos onde cada ponto representa a *ratio* para dada tensão de *threshold*, com uma barra de erro de 3 sigma para cada ponto. Seguem os gráficos resultantes para cada canal:

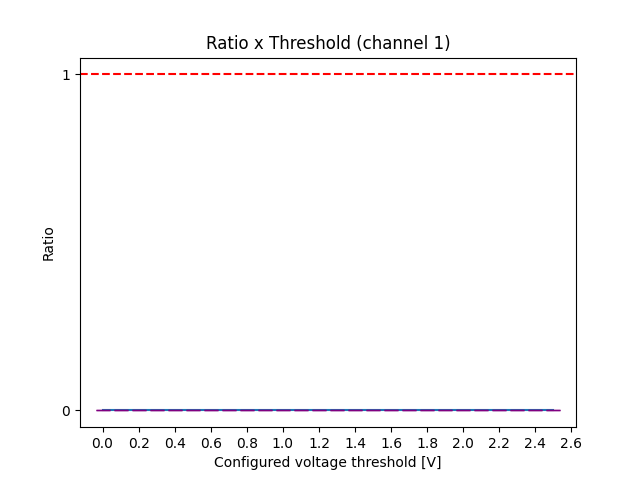


Figura 14 - *Ratio* x *threshold* do canal 1

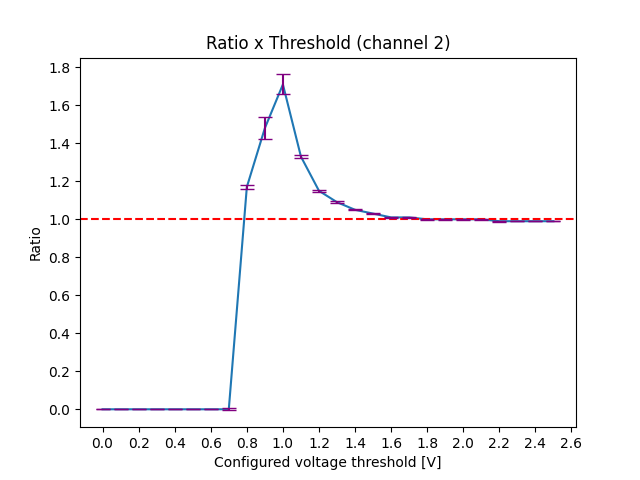


Figura 15 - *Ratio* x *threshold* do canal 2

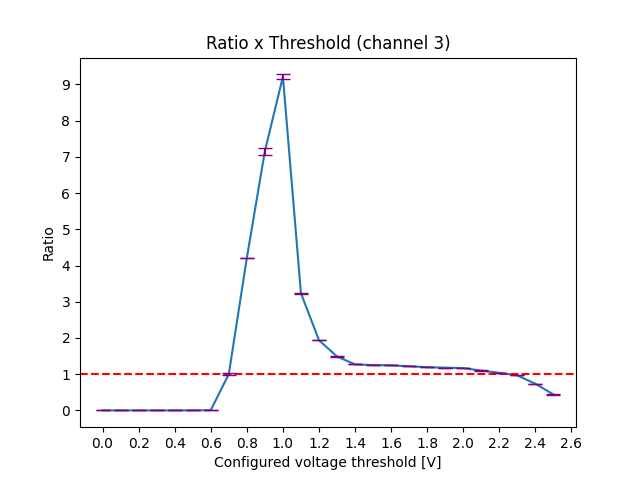


Figura 16 - *Ratio* x *threshold* do canal 3

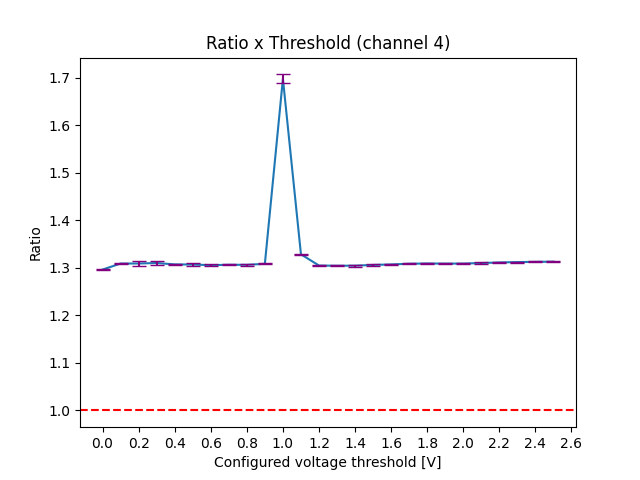


Figura 17 - *Ratio* x *threshold* do canal 4

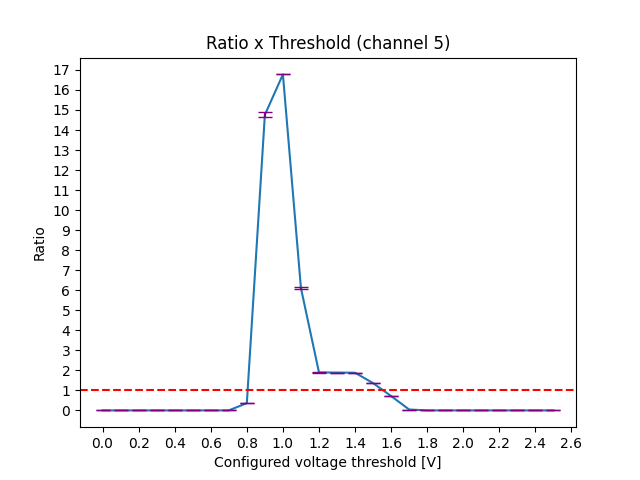


Figura 18 - *Ratio* x *threshold* do canal 5

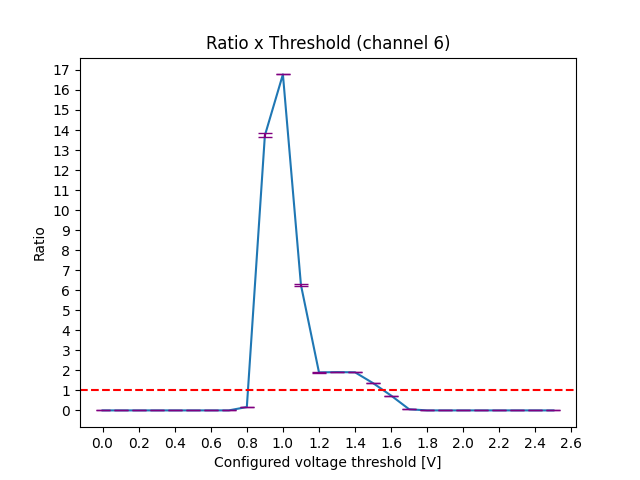


Figura 19 - *Ratio* x *threshold* do canal 6

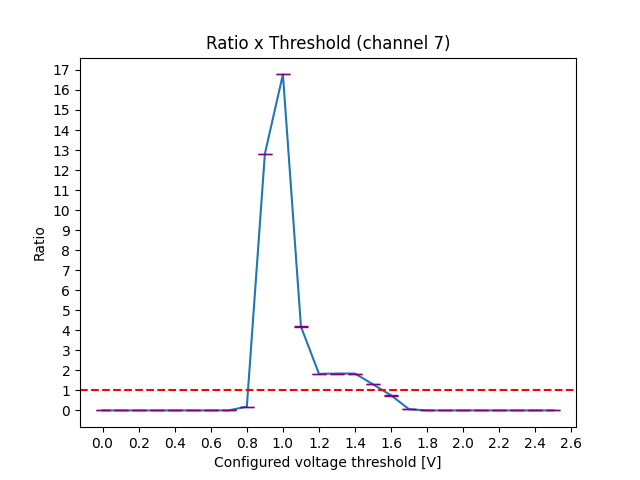


Figura 20 - *Ratio* x *threshold* do canal 7

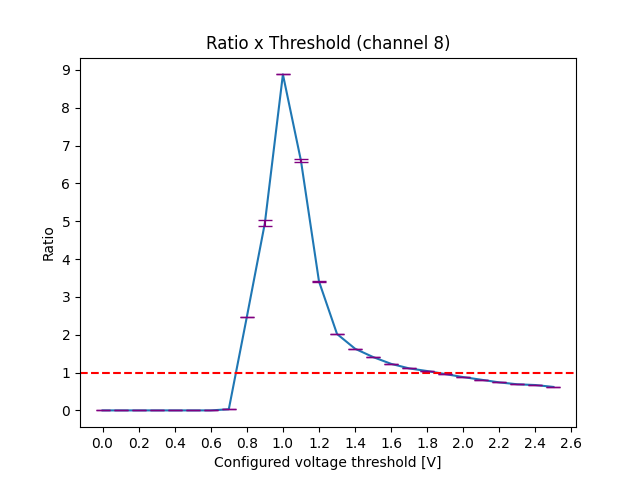


Figura 21 - *Ratio* x *threshold* do canal 8

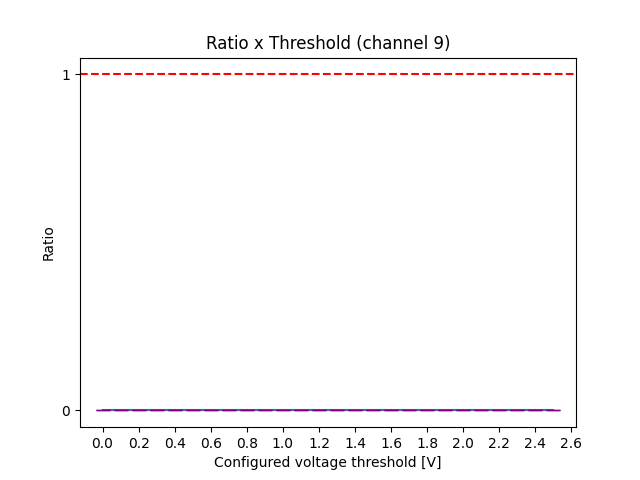


Figura 22 - *Ratio* x *threshold* do canal 9

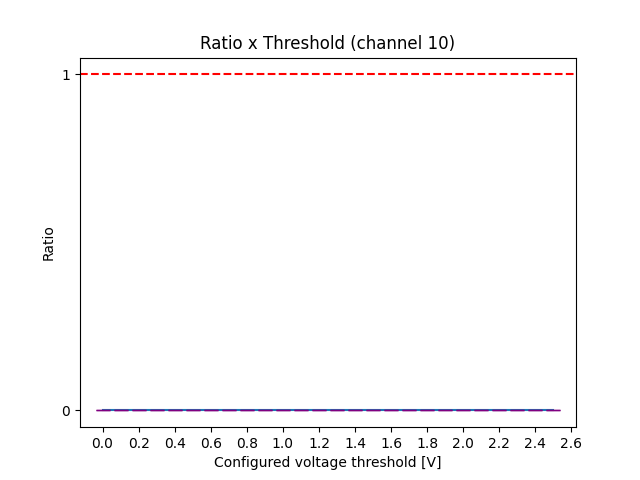


Figura 23 - *Ratio* x *threshold* do canal 10

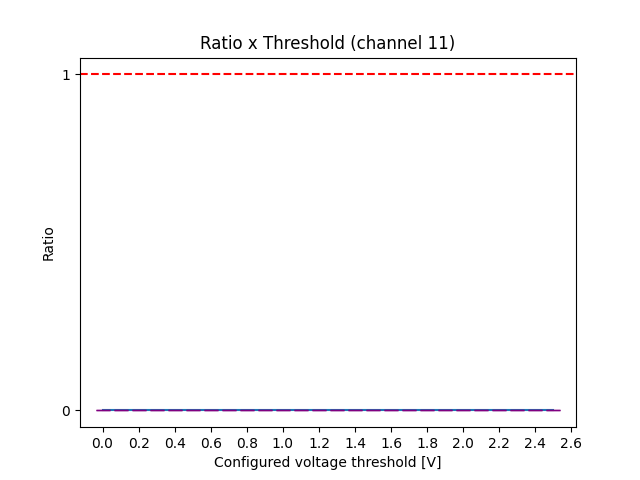


Figura 24 - *Ratio* x *threshold* do canal 11

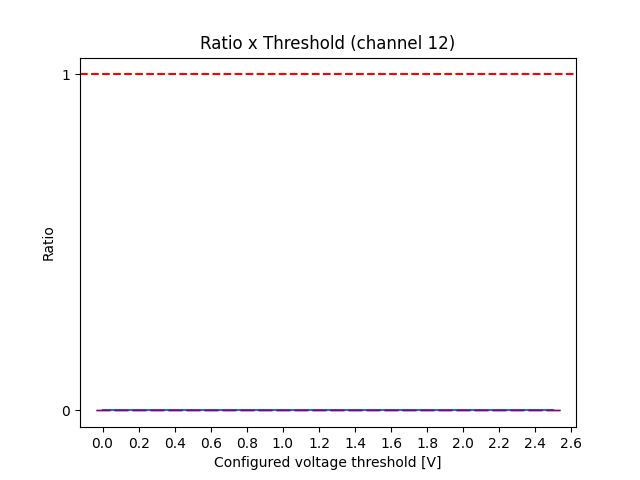


Figura 25 - *Ratio* x *threshold* do canal 12

Os canais 1, 9, 10, 11 e 12 apresentaram contagens nulas para todos os valores de *threshold*. É possível que existam falhas na soldagem de componentes por onde os pulsos se propagam no caminho da entrada dos canais até a entrada correspondente no FPGA, impedindo a chegada do sinal. Também é possível que componentes não estejam funcionando, ocasionando o mesmo problema. O canal 4, apesar de apresentar contagens, manteve sua resposta constante em mais ou menos 1,3 milhão de contagens, com um pico por volta do *threshold* de 1 V. Sua resposta no geral insensível às mudanças de *threshold* evidencia um provável sinal de ruído aproximadamente constante na entrada correspondente do FPGA, gerando contagens constantes sem relação com os pulsos injetados, com o pico relacionado a sinais de ruído anteriores ao discriminador sendo propagados para aquela faixa de *threshold*. Assim, esses canais não apresentaram funcionamento e resultados adequados.

Para os demais canais (2, 3, 5, 6, 7 e 8) pode-se notar que a *ratio* começa com valor zero, e, a partir de certo valor de *threshold*, começa a crescer rapidamente até atingir um pico. Então ela decai novamente até estabilizar próximo de um determinado valor. Os canais 5, 6 e 7 apresentam ainda um decaimento de volta para zero após a primeira estabilização. O formato dos gráficos tem a seguinte explicação: quando o *threshold* está próximo de zero, a saída do segundo estágio de amplificação tem um valor *DC* que é sempre maior que o *threshold*, deixando a saída do discriminador sempre em alto, e nenhuma contagem é feita (a contagem é feita mediante uma borda de subida). Quando o *threshold* se aproxima desse nível *DC*, o número de contagens sobe rapidamente, devido ao ruído do circuito gerando um número muito alto de bordas de subida, resultando em falsas contagens (por isso a *ratio* se torna bem maior do que 1 próximo do pico). Conforme o *threshold* continua crescendo, as bordas de subida do ruído vão sendo eliminadas, resultando num número de contagens cada vez mais próximo do número de pulsos reais injetados. Foi adicionada uma linha horizontal vermelha aos gráficos acima, correspondendo ao valor de *ratio* 1. O ponto de interseção dessa linha com a linha azul, a curva de *ratio*, corresponde ao valor de *threshold* para o qual o número de contagens mais se aproxima no número real. O decaimento após a estabilização, quando ocorre, é devido ao *threshold* se tornando maior que a amplitude dos pulsos amplificados, eliminando-os da saída do discriminador.

Os canais 2, 3 e 8 apresentaram a melhor resposta dentre todos, com a razão entre pulsos contados e injetados se estabilizando após o pico em valores bem próximos de 1, especialmente o canal 2, que se estabilizou quase exatamente em 1. Isso mostra que, para esses canais, foi possível cortar o sinal de ruído do sinal dos pulsos, a partir de determinados valores de *threshold*.

# Conclusão

O objetivo inicial desse projeto foi a criação de uma *PCB* unificada de aquisição de dados (*DAQ*), interligando sensores, microcontrolador, *FPGA* e demais ICs numa única placa de forma robusta. Esse objetivo, assim como o desenvolvimento dos firmwares do microcontrolador principal e do *FPGA*, foram cumpridos satisfatoriamente, com testes mostrando que a aquisição, o armazenamento e o envio de todos os dados do experimento ininterruptamente foram bem sucedidos.

Os testes com injeção de pulsos por gerador de função, porém, mostraram que a integração do *DAQ* com a *FEE* não foi satisfatória. Boa parte dos canais não funcionou, apresentando resposta nula ou constante. Os canais que obtiveram contagens de pulsos apresentaram nível bastante elevado de ruído, e apenas os canais 2, 3 e 8 conseguiram um valor estável e correto de contagens a partir de um determinado valor de threshold.

Os problemas de não funcionamento de canais e de elevado nível de ruído tem sua origem provável na montagem e também no layout da placa. Para as próximas fases do projeto, será necessário realizar mais testes na placa para detectar a origem de fontes de ruído e também de possível ocorrência de *crosstalk* entre diferentes linhas de sinal. Também é recomendado a confecção de *PCBs* de teste para validar o design do circuito de *Front-End,* que foi a parte que apresentou mais problemas no projeto.

Apesar dos problemas encontrados, o projeto representou um avanço para o experimento *CRE@AT*, obtendo um design funcional para o circuito de *DAQ*. Os firmwares do ESP32 e do *FPGA* foram validados e estão prontos para serem reutilizados ou adaptados para etapas subsequentes do experimento.

**REFERÊNCIAS**

1. GAISSER, T. K.; ENGEL, R.; RESCONI, E. **Cosmic rays and particle physics**. 2. ed. Cambridge: Cambridge University Press, 2016.
2. FAUTH, A. C. *et al*. Demonstração experimental da dilatação do tempo e da contração do espaço dos múons da radiação cósmica. **Revista Brasileira de Ensino de Física**, [São Paulo], v. 29, n. 4, p. 585–591, 2007.
3. HATHAWAY, D. H. The solar cycle. **Living reviews in solar physics**, Suíça, v. 12, n. 1, 2015.
4. HEBER, B.; FICHTNER, H.; SCHERER, K. Solar and heliospheric modulation of galactic cosmic rays. **Space science reviews**, [Alemanha], v. 125, n. 1–4, p. 81–93, 2007.
5. ASLAM, O. P. M.; BADRUDDIN. Study of cosmic-ray modulation during the recent unusual minimum and mini-maximum of solar cycle 24. **Solar physics**, Dordrecht, v. 290, n. 8, p. 2333–2353, 2015.
6. FRIIS-CHRISTENSEN, E.; LASSEN, K. Length of the solar cycle: an indicator of solar activity closely associated with climate. **Science**, Washington, D.C., v. 254, n. 5032, p. 698–700, 1991.
7. SVENSMARK, H.; FRIIS-CHRISTENSEN, E. Variation of cosmic ray flux and global cloud coverage—a missing link in solar-climate relationships. **Journal of atmospheric and solar-terrestrial physics**, Grã-Bretanha, v. 59, n. 11, p. 1225–1232, 1997.
8. SVENSMARK, H. Cosmoclimatology: a new theory emerges. **Astronomy & geophysics**, Oxford, v. 48, n. 1, p. 1.18-1.24, 2007.
9. SVENSMARK, H. Influence of cosmic rays on earth’s climate. **Physical review letters**, [EUA], v. 81, n. 22, p. 5027–5030, 1998.
10. MARSH, N.; SVENSMARK, H. Cosmic Rays, Clouds, and Climate. **Space science reviews**, [Alemanha], v. 94, n. 1/2, p. 215–230, 2000.
11. MARSH, N.; SVENSMARK, H. Solar influence on Earth’s climate. **Space science reviews**, [Alemanha], v. 107, n. 1–2, p. 317–325, 2003.
12. TURCO, R. P.; ZHAO, J.-X.; YU, F. A new source of tropospheric aerosols: Ion-ion recombination. **Geophysical research letters**, [EUA], v. 25, n. 5, p. 635–638, 1998.
13. YU, F.; TURCO, R. P. Ultrafine aerosol formation via ion‐mediated nucleation. **Geophysical research letters**, [EUA], v. 27, n. 6, p. 883–886, 2000.
14. DA GRAÇA, U. DE F. C. **Desenvolvimento do experimento Antártico de monitoração de raios cósmicos para o módulo Criosfera I**. 2015. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Eletrônica) – Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, Rio de Janeiro, 2015.
15. **Fotografia de MPPC da série S13360**. [entre 2016 e 2024]. 1 fotografia. Disponível em: https://www.digchip.com/datasheets/photos/190/S13360-1325CS-1.jpg. Acesso em: 15 dez. 2024.
16. HAMAMATSU PHOTONICS K.K. **[MPPC (Multi-Pixel Photon Counter) S13360 series](https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwift9WalquKAxUwppUCHRPKMFMQFnoECB0QAQ&url=https://www.hamamatsu.com/content/dam/hamamatsu-photonics/sites/documents/99_SALES_LIBRARY/ssd/s13360_series_kapd1052e.pdf&usg=AOvVaw0bmzzQUFmo0J7Hlb3UAlnz&opi=89978449)**[. Japão: Hamamatsu Photonics K.K., 2024. 12 p. Disponível em: https://www.hamamatsu.com/content/dam/hamamatsu-photonics/sites/documents/99\_SALES\_LIBRARY/ssd/s13360\_series\_kapd1052e.pdf. Acesso em: 15 dez. 2024.](https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwift9WalquKAxUwppUCHRPKMFMQFnoECB0QAQ&url=https://www.hamamatsu.com/content/dam/hamamatsu-photonics/sites/documents/99_SALES_LIBRARY/ssd/s13360_series_kapd1052e.pdf&usg=AOvVaw0bmzzQUFmo0J7Hlb3UAlnz&opi=89978449)
17. *Firmware* do *FPGA*

1. *Firmware* do ESP32

1. Código da interface *PyControl*

* 1. Alguma coisa