



**Universidade de Brasília
Faculdade de Tecnologia**

**Projeto de um sistema embarcado para
controle de exoesqueleto**

Cristóvão Bartholo Gomes

**TRABALHO DE GRADUAÇÃO
ENGENHARIA DE CONTROLE E AUTOMAÇÃO**

Brasília
2022

**Universidade de Brasília
Faculdade de Tecnologia**

**Projeto de um sistema embarcado para
controle de exoesqueleto**

Cristóvão Bartholo Gomes

Trabalho de Graduação submetido como requisito parcial para obtenção do grau de Engenheiro de Controle e Automação.

Orientador: Prof. Dr. Carlos Humberto Llanos Quintero

Brasília
2022

B287p Bartholo Gomes, Cristóvão.
Projeto de um sistema embarcado para controle de exoesqueleto / Cristóvão Bartholo Gomes; orientador Carlos Humberto Llanos Quintero. -- Brasília, 2022.
64 p.

Trabalho de Graduação em Engenharia de Controle e Automação -- Universidade de Brasília, 2022.

1. exoesqueleto. 2. encoder. 3. FPGA. 4. Pynq-Z2. I. Quintero, Carlos Humberto Llanos, orient. II. Título

**Universidade de Brasília
Faculdade de Tecnologia**

**Projeto de um sistema embarcado para controle de
exoesqueleto**

Cristóvão Bartholo Gomes

Trabalho de Graduação submetido como re-
quisito parcial para obtenção do grau de Enge-
nheiro de Controle e Automação.

Trabalho aprovado. Brasília, 26 de Outubro de 2022:

**Prof. Dr. Carlos Humberto Llanos
Quintero**
Orientador

**Prof. Dr. Daniel Mauricio Muñoz
Arboleda**
Examinador interno

Prof. Dr. Renato Coral Sampaio
Examinador interno

Brasília
2022

*“Apesar de tudo, tenho fé nas crianças, e se quisermos
construir um mundo melhor aqui está a base”
(Professor Girafales, **Roberto Gómez Bolaños**)*

Resumo

Os exoesqueletos são dispositivos eletromecânicos destinados a auxiliar movimentos de pessoas. Com o intuito de desenvolver a instrumentação de um exoesqueleto voltado para reabilitação que está em desenvolvimento no Grupo de Automação e Controle (GRACO, UnB), deu-se continuidade ao projeto da estruturação de uma bancada para que possam ser estudados os sensores e os requisitos para instrumentação e controle dos atuadores. Foi determinada a arquitetura do sistema de acoplamento entre as interfaces de sensores, atuadores e controle. Uma interface de acoplamento foi desenvolvida de modo a permitir o controle a ser implementado nas próximas etapas do projeto. A placa teve seus requisitos levantados, foi fabricada em território nacional, foi testada e teve seu funcionamento validado no funcionamento do sistema.

Palavras-chave: exoesqueleto. encoder. FPGA. Pynq-Z2.

Abstract

Exoskeletons are electromechanical devices which purposes are to assist the movement of people. With the intention of developing the instrumentation of an exoskeleton aimed towards rehabilitation that is being developed at UnB's Group of Automation and Control (GRACO), the project of making a stand so that one can study the sensors' nature and the actuators requirements was continued. The architecture of the system's coupling between sensors, actuators and control was specified. A board was designed to allow the control system to be implemented in the future. The board was fabricated in Brazil, was tested and validated.

Keywords: exoskeleton. encoder. FPGA. Pynq-Z2.

Lista de ilustrações

Figura 1 – Categorias de exoesqueletos (TEJERA et al., 2021).	12
Figura 2 – Design CAD das peças projetadas para o exoesqueleto (FREIRE et al., 2020a).	15
Figura 3 – Arquitetura geral do projeto.	17
Figura 4 – Visualização interna de um motor EC da MAXON	19
Figura 5 – Visualização interna de um encoder absoluto	20
Figura 6 – Funcionamento interno de um encoder incremental.	20
Figura 7 – Estrutura interna de uma SoC FPGA Zynq.	21
Figura 8 – Estrutura do exoesqueleto retirado da referência (FREIRE et al., 2020b).	22
Figura 9 – Motor MAXON com encoder MILE embutido.	22
Figura 10 – Caixa de redução Maxon GP6.	23
Figura 11 – Driver Escon 70/10.	24
Figura 12 – Driver Escon 70/10.	25
Figura 13 – Fonte PS-360.	25
Figura 14 – placa SoC da Xilinx, modelo Pynq Z2.	26
Figura 15 – Funcionalidades de interface implementadas pelo <i>base overlay</i>	27
Figura 16 – Bancada desenvolvida pela Ana Paula.	27
Figura 17 – Reconstrução do cursor.	30
Figura 18 – Caracterização gerada no ESCON Stutio.	31
Figura 19 – Circuito da ligação da chave.	32
Figura 20 – Gráfico dos dados do encoder gerado pela IDE do Arduíno.	33
Figura 21 – Placa anterior utilizada como base.	34
Figura 22 – Conexões espelhadas na placa anterior (RODRIGUES, 2017).	35
Figura 23 – Posição angular das juntas do joelho ao longo da marcha (ROBBI, 2018).	36
Figura 24 – Representação 3D do protótipo.	37
Figura 25 – Representação 2D do protótipo.	37
Figura 26 – Conexão compatível com a Raspberry pi na Pynq Z2.	38
Figura 27 – Diagrama representativo do daisy chain.	39
Figura 28 – Segunda iteração do projeto da placa em 2D.	40
Figura 29 – Segunda iteração do projeto da placa em 3D.	40
Figura 30 – Diagrama dos planos de terra. A) Motor 1; B) Pynq Z2; C) Motor 2, D) Encoder 1; E) Encoder 2.	41
Figura 31 – Visualização dos circuitos alimentados pela Pynq Z2	42
Figura 32 – Errata da documentação.	42
Figura 33 – Visualização em perspectiva da placa final.	43
Figura 34 – Visualização por cima da placa final.	43

Figura 35 – Visualização por baixo da placa final.	43
Figura 36 – Placa fabricada e com componentes montados.	44
Figura 37 – Bancada final de testes.	44
Figura 38 – Erro e correção feita em uma das placas.	45
Figura 39 – Montagem do teste para o motor 1.	46
Figura 40 – Montagem do teste para o motor 2.	47
Figura 41 – Montagem do teste para o motor 1 em daisy chain (ou motor 3).	48
Figura 42 – Montagem do teste para o motor 2 em daisy chain (ou motor 4).	49
Figura 43 – Entrada para Driver dada forma de onda de 1kHz.	49
Figura 44 – Entrada para Driver dada forma de onda de 3kHz.	50
Figura 45 – Montagem do teste para o encoder do motor 1.	50
Figura 46 – Montagem do teste para o encoder do motor 2.	51
Figura 47 – Montagem do teste para o encoder do motor 3.	52
Figura 48 – Montagem do teste para o encoder do motor 4.	53
Figura 49 – Entrada do encoder para a Pynq Z2 a uma forma de onda de 1kHz.	53
Figura 50 – Entrada do encoder para a Pynq Z2 a uma forma de onda de 1MHz.	54
Figura 51 – Atraso do sinal do encoder para a Pynq Z2.	54
Figura 52 – Atraso do sinal da Pynq Z2 para o driver do motor.	54
Figura 53 – Conector da placa de interface.	55

Lista de tabelas

Tabela 1 – Tabela resumo de requisitos. (FREIRE et al., 2020b)	15
Tabela 2 – Configuração do ESCON Studio.	24
Tabela 3 – Sinais de input e output por motor	38
Tabela 4 – Conexões Pynq Z2	39

Sumário

1	INTRODUÇÃO	12
1.1	Descrição do problema	12
1.2	Motivação	13
1.3	Trabalhos anteriores	14
1.4	Objetivo Geral	14
1.5	Objetivos Específicos	16
1.6	Hipótese desta parte da pesquisa	16
1.7	Métodos utilizados no trabalho	16
1.8	Estrutura do texto	17
2	MATERIAIS USADOS NO PROJETO	19
2.1	Motor EC	19
2.2	Encoders	19
2.3	SoC FPGA	20
2.4	Estrutura	21
2.5	Atuadores	22
2.6	Caixa redutora	23
2.7	Sensores	23
2.8	Drivers dos motores	24
2.9	Fonte de alimentação	25
2.10	Pynq Z2	26
2.11	Bancada experimental	26
3	MÉTODOS	28
3.1	Projeto	28
3.2	Implementação	28
3.3	Teste e Validação	28
4	PROCEDIMENTOS INICIAIS	30
4.1	Preparativos	30
4.2	Configuração do Driver	30
4.3	Procedimentos gerais	31
5	PLACA DE INTERFACE	34
5.1	Projeto Base	34
5.2	Requisitos Gerais do projeto	35

5.3	Histórico de desenvolvimento	36
6	RESULTADOS	44
7	CONCLUSÃO	56
	REFERÊNCIAS	58
	APÊNDICES	60
	APÊNDICE A – ESQUEMÁTICO	61
	ANEXOS	63

1 Introdução

1.1 Descrição do problema

Exoesqueletos são dispositivos eletromecânicos utilizados para auxiliar no movimento motor de seres humanos, tendo aplicação em diversas áreas, como na área militar, da fisioterapia e na indústria em geral. O equipamento consiste em uma armadura vestível instrumentada com sensores e atuadores que se comportam de maneira realizar o movimento desejado por quem o veste. A vantagem consiste em fazer com que o trabalho seja realizado em parte pelo equipamento.

Os exoesqueletos podem ser categorizados a partir de seis critérios, sendo eles qual o foco da parte do corpo, estrutura, ação, alimentação, propósito e área de aplicação. (TEJERA et al., 2021) As categorias estão apresentadas no diagrama da Figura 1.

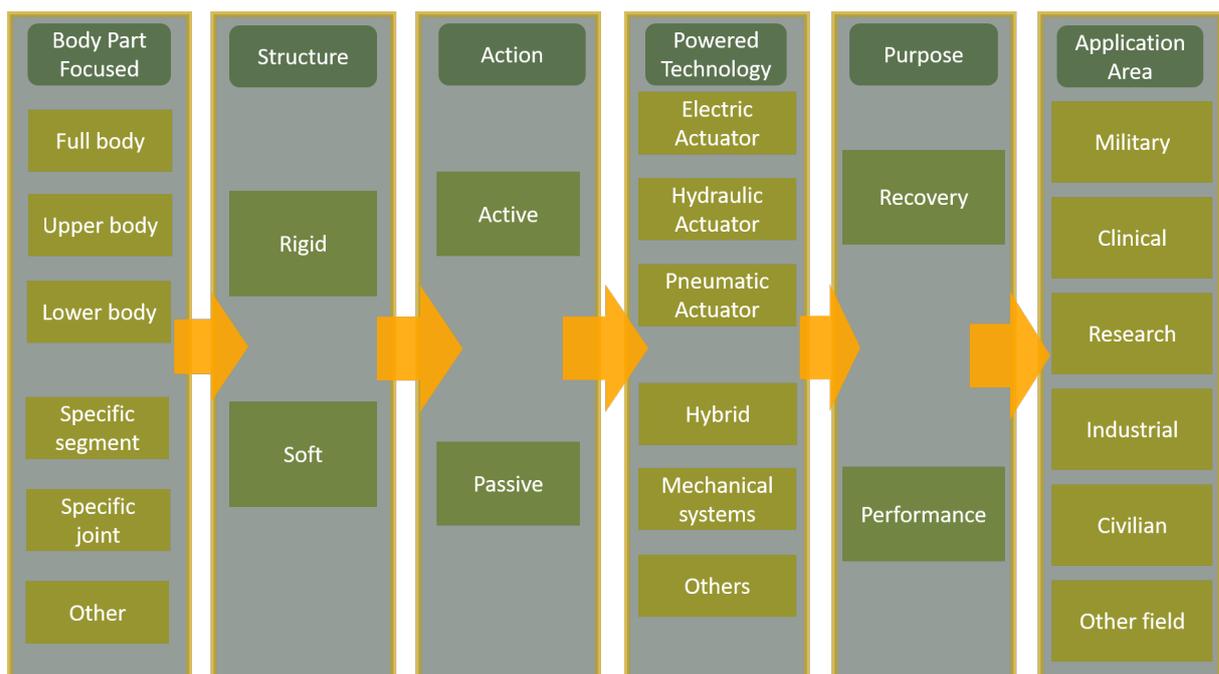


Figura 1 – Categorias de exoesqueletos (TEJERA et al., 2021).

Na área biomédica, indivíduos em que o uso de uma órtese não é capaz de prover a mobilidade desejada um exoesqueleto pode ser usado como alternativa, pois em contraste com uma órtese, um exoesqueleto consegue realizar o trabalho de boa parte da musculatura. Outra aplicação é o uso de exoesqueletos no auxílio da caminhada de pacientes que sofrem de derrame, lesão medular ou envelhecimento.

Há aplicações também área militar, em que os soldados podem utilizar do equipamento para reduzir a fadiga e maximizar a produtividade em uma determinada tarefa, por

exemplo. Exoesqueletos também estão sendo desenvolvidos de modo a auxiliar bombeiros a subirem escadas carregando equipamentos pesados. Além disso, na área da indústria, exoesqueletos passivos são comumente usados para reduzir acidentes de trabalho e erros devido a fadiga.

Vários tipos de exoesqueletos estão disponíveis hoje no mercado, porém existe a necessidade de desenvolvimento de exoesqueletos que são confortáveis, que tenham preço acessível e sejam seguros (WOLFF et al., 2020). Este trabalho consiste em uma contribuição para o projeto de um exoesqueleto de membros inferiores que está em desenvolvimento na Universidade de Brasília. Isso foi realizado através do desenvolvimento de uma interface entre o sistema de controle e os sensores e atuadores. A interface realiza a conversão de tensão bem como o isolamento elétrico entre ambos os lados do sistema. Essa placa comporta uma entrada para a SoC (System on a Chip) *Pynq Z2* e será capaz de controlar dois motores do sistema, além disso se pode ligá-la a outra idêntica através de um daisy chain e controlar simultaneamente quatro motores.

A placa *Pynq Z2* apresenta tensão lógica de trabalho de 3,3 Volts, enquanto a lógica do driver utilizado opera a 5 Volts. A comunicação é feita através de optoacopladores, que por sua natureza realizarão o isolamento elétrico, já que não há conexão entre os lados desse dispositivos e a comunicação ocorre através de um led e um fototransistor. Isso permite também que haja duas diferentes tensões em cada lado do optoacoplador, respeitando os limites do dispositivo.

1.2 Motivação

Existe uma carência por pesquisa na área de exoesqueleto no Brasil, onde não há fabricantes nacionais e as poucas pesquisas na área se encontram principalmente na Universidade de São Paulo e na Universidade de Campinas e o censo de 2000 do IBGE (IBGE, 2000) constatou que haviam cerca de um milhão de brasileiros com deficiência física, crescendo esse valor para em torno de treze milhões em 2010 (IBGE, 2010). Dentre eles estão os hemiplégicos, que apresentam a paralisia em um dos lados do corpo, muitas vezes decorrente de um AVC (Acidente Vascular Cerebral), por exemplo. Essa condição que acomete os indivíduos pode ser tratada com reabilitação, em que o paciente é acompanhado por uma equipe que faz uso de aparatos para auxiliar na recobrança dos movimentos.

Exoesqueletos são equipamentos de grande relevância na sociedade principalmente por sua aplicação na área de reabilitação, porém poucos centros de reabilitação no Brasil estão equipados com esses dispositivos. Portanto, a motivação desse projeto se encontra na intensão de produzir equipamentos nacionais e torná-los mais acessíveis à população em geral, dado que os disponíveis no mercado são de custo proibitivo para grande parte dos consumidores.

O desafio deste trabalho se encontra no fato de que a pesquisa é pioneira e há pouca base para desenvolvimento. Além disso, o evento da pandemia fez com que o desenvolvimento ficasse dificultado devido às restrições sanitárias, que dificultou o acesso aos laboratórios e o acesso aos equipamentos para o desenvolvimento dos trabalhos.

O projeto do exoesqueleto recebeu apoio financeiro do CNPq (Edital Universal, 2016) e da FAPDF (Demanda espontânea, 2017).

1.3 Trabalhos anteriores

Pesquisas na área biomédica vem sendo realizadas por alunos e professores da Universidade de Brasília para o desenvolvimento de equipamentos para auxiliar a reabilitação de indivíduos. Um Estudo de caso utilizando algoritmo de otimização multiobjetivo em conjunto com a técnica de elementos finitos foi empregada para simular os esforços sobre uma peça de exoesqueleto. Isso foi feito para avaliar e propor múltiplas soluções para o design da peça visando segurança (PASSOS et al., 2018).

Também realizou-se uma revisão bibliográfica em que visava encontrar conteúdos relevantes para os estudos referentes ao projeto (MARAFA; LLANOS; TACO, 2020). A partir do conteúdo reunido, iniciou-se os estudos em análise dos componentes biomecatrônicos (mecanismos, atuadores, sensores controle e interação humano-robô) para exoesqueleto de membros inferiores nos contextos de reabilitação e capacidade aumentada (MARAFA; SAMPAIO; LLANOS, 2020).

Com as pesquisas realizadas, iniciou-se o processo de design da estrutura do exoesqueleto (FREIRE et al., 2020a). O processo do desenvolvimento se deu baseado nos requisitos mostrados na Tabela 1, resumidamente (FREIRE et al., 2020b). Paralelamente, desenvolveu-se o projeto em que se instrumentava uma bengala visando coletar dados da marcha de um paciente. Utilizou-se um conjunto de sensores para a coleta e estimou-se os ângulos de Euler através do uso do Filtro de Kalman, além de medição de estimação de força axial aplicada pelo usuário (MEDEIROS GONÇALVES, 2017).

Finalmente, tem-se o trabalho realizada pela ex-aluna Ana Paula Chavier Rodrigues (RODRIGUES, 2017), em que este trabalho tem como base. Nele, foi desenvolvida uma interface que tinha como objetivo o acionamento de atuadores e leitura de encoders para o desenvolvimento de um exoesqueleto de membros inferiores com a plataforma *SoC-FPGA Zybo*.

1.4 Objetivo Geral

O objetivo geral deste trabalho consiste em fornecer ao projeto do exoesqueleto uma placa que realiza a interface entre a parte de controle e a parte física do sistema, bem como

Tabela 1 – Tabela resumo de requisitos. (FREIRE et al., 2020b)

Componente	Requisito Estabelecido
Altura Máx. (Usuário)	1900 mm
Peso Máx. (Usuário)	100 Kg
Velocidade de Marcha	0.3 m/s
Comprimento do Cintura	435 mm
Comprimento Quadril-Joelho	445 mm
Comprimento Joelho-Calcanhar	435 mm
Graus de Liberdade do Quadril	1 Grau de Liberdade
Graus de Liberdade do Joelho	1 Grau de Liberdade
Range da Junta do Quadril	-20° (Flexão) 120° (Extensão)
Range da Junta do Joelho	0° (Flexão) 135° (Extensão)
Torque da Junta do Quadril	0.07 Nm/Kg (Flexão) -0.28 Nm/Kg (Extensão)
Range da Junta do Joelho	0.14 Nm/Kg (Flexão) -0.25 Nm/Kg (Extensão)
Máx. Torque Absoluto no Quadril	28 Nm
Máx. Torque no Joelho	25 nM

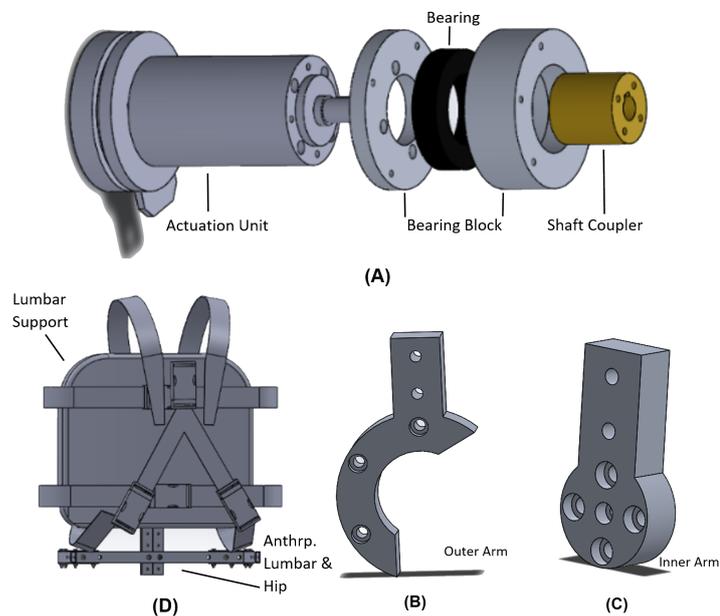


Figura 2 – Design CAD das peças projetadas para o exoesqueleto (FREIRE et al., 2020a).

estruturar uma bancada de testes. Desta forma poderá se desenvolver o projeto de maneira mais adequada, fornecendo a facilidade de uso dos dispositivos necessária para as próximas etapas do projeto.

1.5 Objetivos Específicos

Os objetivos específicos desse trabalho envolvem diversas pequenas etapas que culminam no desenvolvimento e validação da placa de interface. Dentre os diversos objetivos específicos que este trabalho contém, podemos enumerar cinco mais relevantes:

- Revitalização da bancada de testes feita pela ex-aluna Ana Paula Chavier Rodrigues (RODRIGUES, 2017);
- Projetar placa de interface;
- Após a fabricação, montagem dos componentes da placa;
- Testar funcionamento da placa;
- Validação do funcionamento no sistema.

1.6 Hipótese desta parte da pesquisa

Embora não seja um trabalho de pesquisa propriamente dito, podemos colocar como hipótese principal ao fato de que é possível desenvolver uma bancada que permita controlar até quatro motores simultaneamente, a fim de avançar na proposta da construção de um sistema de controle para uma perna do exoesqueleto.

1.7 Métodos utilizados no trabalho

De modo a cumprir com os objetivos do trabalho, estabeleceu-se divisões e marcadores de progresso. O método consistiu em dividir o trabalho em três etapas. Na primeira, a etapa de projeto, o objetivo consistia em estudar o funcionamento da interface anterior, verificar os requisitos de projeto, estabelecer os requisitos da interface e implementar o esquemático e design em software. A segunda etapa, a de implementação, tinha como objetivo fabricar e montar a placa. Na terceira e última etapa, a de teste e validação, o objetivo era testar as placas com uso de osciloscópio e em seguida validar utilizando o SoC.

A arquitetura geral do projeto está representada na Figura 3. A interface consiste nas duas placas, conectadas entre si. Somente uma das placas estará conectada ao SoC, a segunda placa estará conectada à primeira por meio da técnica de *daisy chain*, já que a placa SoC que será responsável pelo controle do sistema (Pynq Z2) não apresenta entradas suficientes para uma conexão direta, além disso ambas as placas serão alimentadas pelo Pynq-Z2. Cada placa será responsável pelo interfaceamento entre dois módulos a serem controlados (Figura 3, em ciano), estes módulos são constituídos pelo driver do motor, motor EC e encoder.

O driver funciona como um controlador de velocidade, em que ele recebe três sinais indicando direção, *enable* e um sinal PWM que define a velocidade conforme o driver foi configurado. O driver então controla o motor para atingir a velocidade correspondente à recebida pelo PWM em *loop* fechado, utilizando os sensores de efeito Hall e encoder contidos no motor. Além do sinal do encoder ser enviado ao driver, o sinal também retorna à interface para ser repassado à Pynq Z2 para que ela possa realizar o controle do movimento do exoesqueleto. O sinal de saída do encoder consiste em quatro bits, sendo dois pares dos sinais do canal A e B, formados pelo sinal e seu respectivo complemento que é utilizado para filtrar picos de ruído.

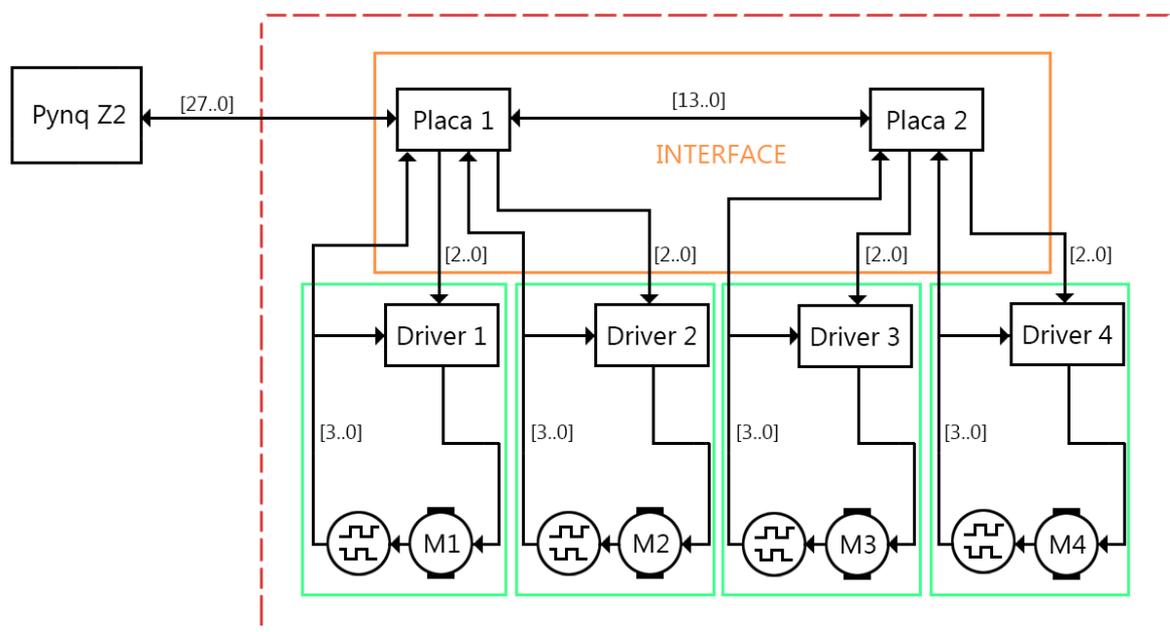


Figura 3 – Arquitetura geral do projeto.

1.8 Estrutura do texto

- **Introdução:** Sumarização autocontida do documento. Descreve sucintamente tudo que é apresentado, além de descrever as motivações e objetivos deste trabalho.
- **Métodos:** Apresenta a metodologia utilizada para definir e rastrear o progresso do trabalho em andamento.
- **Materiais Usados no Projeto:** Descreve os dispositivos e componentes que compõem o projeto.
- **Procedimentos Iniciais:** Relata os procedimentos realizados previamente ao início do trabalho principal.

- **Desenvolvimento:** Relata todo o procedimento de levantamento de requisito, projeto e produção da placa de interface.
- **Resultados:** Apresenta os testes realizados na placa fabricada e montada e seus respectivos resultados.
- **Conclusão:** Sumariza todo o trabalho realizado, desde a motivação até os resultados obtidos.

2 MATERIAIS USADOS NO PROJETO

2.1 Motor EC

Motores EC (Eletronicamente Comutados) são motores que apresentam como uma das vantagens a menor necessidade de manutenção, isso se dá devido a ausência de escovas, que são elementos que fazem o contato elétrico entre a armadura e o eixo, e para isso se faz necessário o contato direto delas com o eixo. Isso faz com que o atrito causado pelo uso do motor desgaste esses elementos fazendo com que se exija uma atenção maior em relação à manutenção do equipamento.

Os motores EC utilizam a estratégia do campo ser controlado na estrutura fixa, e o campo do rotor reage gerando o torque desejado. Motores desse tipo apresentam algumas outras vantagens como alta eficiência e baixo ruído se comparado a outros tipos de motores. Entretanto, para realizar o controle muitas vezes se faz necessário uso de sensores, como sensores de efeito Hall, para que se possa saber a posição angular do rotor e atuar na estrutura fixa para gerar a corrente específica que causa o torque desejado. Então, para que o motor EC funcione, ao contrário de motores DC, se faz necessário o uso de uma aparelhagem eletrônica para comutar o acionamento das bobinas. A estrutura interna de um motor EC pode ser visualizada na Figura 4.

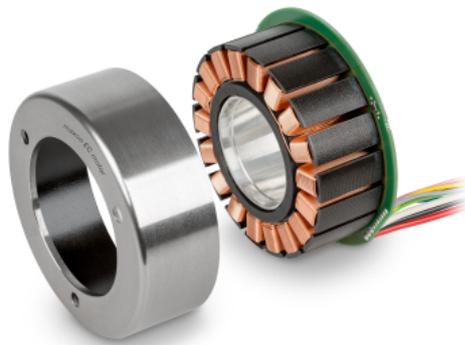


Figura 4 – Visualização interna de um motor EC da MAXON

2.2 Encoders

Encoders são elementos utilizados para a determinação da posição, velocidade e aceleração angular de equipamentos girantes. Eles são divididos em dois grupos: encoders absolutos e incrementais. Os encoders absolutos são utilizados para determinar a posição

angular absoluta de um elemento em qualquer instante de tempo. Ele utiliza de padrões de trilhas sobre uma placa transparente, fazendo com que cada posição tenha uma assinatura digital determinada, como mostrado na Figura 5. Em encoders como esse, para cada trilha é necessário um emissor e um sensor de luz, um em cada lado da placa, dessa forma a combinação dos valores digitais do conjunto dos sensores pode ser decodificado em uma posição angular do disco, e a resolução do sensor é dada pelo número de trilhas e consequentemente de elementos sensores. Pelo fato de possuírem vários sensores e o disco necessitar de boa precisão para um bom funcionamento, esses dispositivos são mais caros e menos usados em aplicações do que encoders de outros tipos.



Figura 5 – Visualização interna de um encoder absoluto

O outro tipo de encoder é o incremental, que utiliza de dois sensores que detectam pulsos gerados por uma trilha com padrão binário defasados em 90° para determinar a posição relativa ao ponto de calibração e a direção de movimentação. Esses sensores costumam ser mais baratos e por esse motivo são mais utilizados em aplicações gerais. Entretanto apresentam algumas desvantagens, dentre elas está a necessidade de calibrar o ponto inicial no início de uma aplicação.

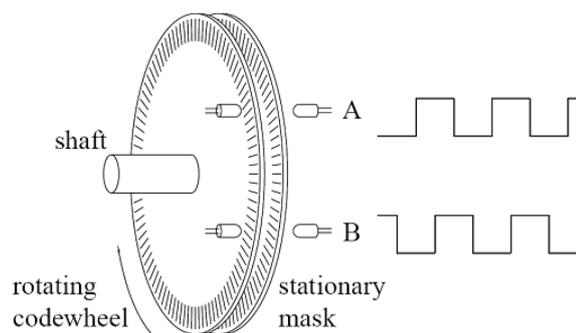


Figura 6 – Funcionamento interno de um encoder incremental.

2.3 SoC FPGA

FPGAs são dispositivos semicondutores baseado em blocos lógicos reconfiguráveis (CLBs) conectados por meio de vias reconfiguráveis. Isso permite que FPGAs tenham sua

função redefinida após a sua fabricação, o que não é possível para Circuitos Integrados de Aplicação Específica (ASICs), que são fabricados para executarem uma única tarefa específica.

O roteamento é realizado por meio de algum software de projeto de circuitos integrados (EDA) que utiliza de uma linguagem de descrição de hardware compilado. Isso faz com que o circuito tenha vantagens características de sistemas implementados por software e por ASICs ao mesmo tempo, pois é reconfigurável, possibilita processamento altamente paralelo de dado, atualização, depuração de código de descrição de hardware, etc.

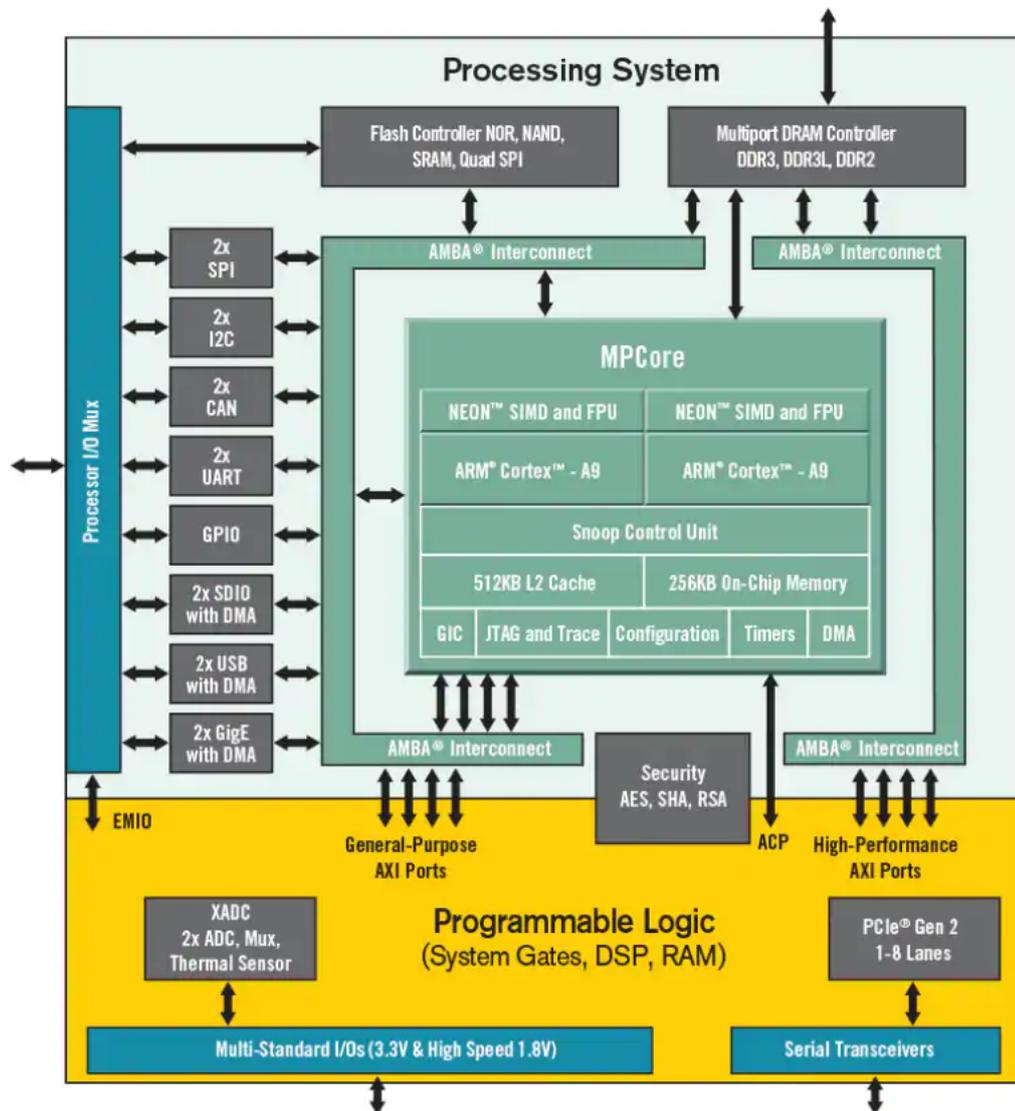


Figura 7 – Estrutura interna de uma SoC FPGA Zynq.

2.4 Estrutura

A estrutura do exoesqueleto teve seus requisitos desenvolvidos (FREIRE et al., 2020b) (FREIRE et al., 2020a) e foi implementada como mostrado na Figura 8. Para o desenvolvi-

mento do projeto foi desenvolvida uma bancada contendo o motor com seus sensores, o driver do motor, a placa de interface e a placa de controle, que representa o sistema desacoplado da parte estrutural do exoesqueleto.



Figura 8 – Estrutura do exoesqueleto retirado da referência (FREIRE et al., 2020b).

2.5 Atuadores

O projeto está sendo desenvolvido com o atuador *EC 90 flat*, que é um motor desenvolvido pela MAXON <WWW.MAXONGROUP.COM/MAXON/VIEW/PRODUCT/323772>. Ele é um motor sem escovas e chato, o que faz com que ocupe pouco espaço na armadura e apresente menor manutenção em comparação ao motor escovado, pois elimina a necessidade do uso de escovas que se desgastam por causa do atrito entre o eixo e a armadura. O modelo selecionado vem equipado com um encoder MILE, que apresenta resolução de 3200 contagens por revolução. (Fig. 9)



Figura 9 – Motor MAXON com encoder MILE embutido.

Entretanto o funcionamento do motor sem escovas traz uma desvantagem, pois para o seu funcionamento é preciso ter um controle mais sofisticado do que motores de corrente contínua. Motores não-escovados contam com sensores de Hall internos para que se possa determinar o posicionamento do eixo, baseado em seu campo magnético, para que o controle possa determinar qual o momento em que cada bobina deve ser acionada. Portanto, para auxiliar o desenvolvimento do projeto, obteve-se também o driver da empresa *MAXON*, modelo *ESCON 70/10* para que o projeto pudesse ser desenvolvido de maneira mais rápida.

2.6 Caixa redutora

De modo a proporcionar o torque correto, será utilizada uma caixa redutora 100:1, a caixa redutora disponível na bancada é do modelo Maxon GP6. Essa caixa redutora apresenta eficiência de no máximo 70% e peso de 1540 gramas. Ela representa a maior parte do volume do conjunto motor-redução, e futuramente poderá ser utilizada uma caixa de redução mais compacta e mais leve para esta aplicação, de modo a impactar no peso e proporções do produto final. A Figura 10 mostra a imagem da caixa de redução desacoplada do motor.



Figura 10 – Caixa de redução Maxon GP6.

2.7 Sensores

O sistema contará com a integração de diversos tipos de sensores, como sensores *POFF*, que são sensores que se baseiam em diversos efeitos ópticos para realizarem suas medições, além de Unidades de Medição Inercial (IMUs) e encoders. Para que possa ser controlado, o motor conta com um sensor encoder *MILE*, da empresa *MAXON*, produzido para o motor *EC 90 flat*.

2.8 Drivers dos motores

Para realizar o acionamento dos motores, o projeto conta com um driver *ESCON 70/10* da *Maxon* (Figura 11) para cada motor a ser acionado. Este é um pequeno driver controlador de servo por PWM usado para atuar motores de imã permanente DC ou EC. Pode fornecer uma potência de aproximadamente até 700 Watts. Este driver oferece as opções de controle de velocidade em malha aberta, malha fechada e controle de corrente. Para esse projeto será utilizado o controle de malha fechada. O dispositivo é configurável por meio da porta USB através uma máquina Windows utilizando o software *ESCON Studio 2.2*, proprietário da *Maxon*. Utilizando os parâmetros do motor e de seu encoder, o software é configurado como mostra a Tabela 2. A conexão do driver com o sistema de controle, encoder e motor está representada no anexo.



Figura 11 – Driver Escon 70/10.

Tabela 2 – Configuração do ESCON Studio.

Parâmetro	Valor
Constante de velocidade	135 rpm/V
Thermal Time Constant Winding	34.1 s
Number of Pairs	12
Max Permissible Speed	5000 rpm
Nominal Current	5 A
Max Output Current Limit	6.06 A
Type of sensor	Digital Incremental Encoder
resolution	3200 counts/turn
Encoder Direction	Maxon
Mode of Operation	Speed Controller (Closed Loop)

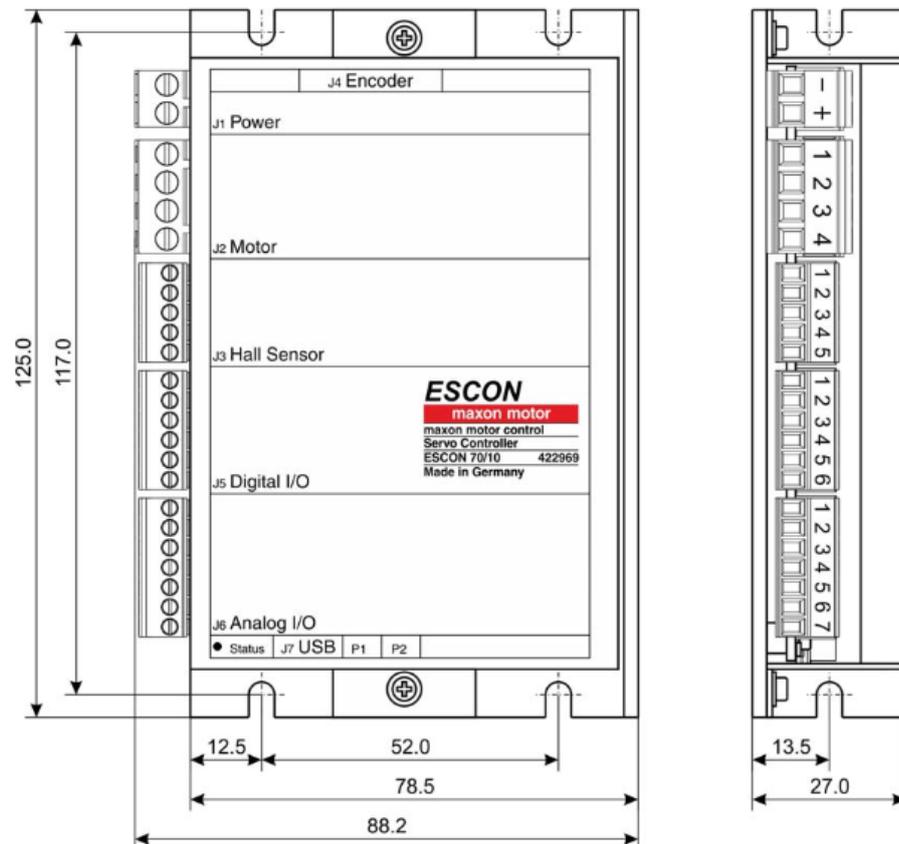


Figura 12 – Driver Escon 70/10.

2.9 Fonte de alimentação

O modelo da fonte de alimentação utilizada na bancada é a PS360. Essa fonte é bivolt e comporta alimentação para dois drivers do motor. Sua saída é de de 24 Volts DC e apresenta corrente máxima de 15 Ampères (Fig 13). Apesar de essa fonte ser provavelmente inadequada para o uso no produto final do exoesqueleto dada sua dimensão e peso, essa fonte se faz útil na fase de projeto e teste.



Figura 13 – Fonte PS-360.

2.10 Pynq Z2

O projeto também conta com uma placa SoC Pynq Z2, que será responsável por realizar controle do sistema. Essa placa SoC se faz útil por apresentar um processador 650MHz dual-core Cortex-A9 juntamente com uma placa FPGA e outros componentes. Isso faz com que seja possível realizar um controle em alto-nível no processador e também realizar tarefas que demandam mais velocidade através da programação da placa FPGA.

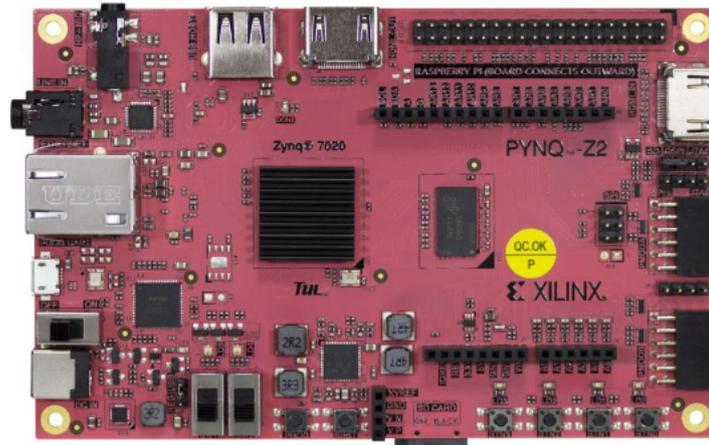


Figura 14 – placa SoC da Xilinx, modelo Pynq Z2.

Essa placa foi desenvolvida pela *Xilinx* para suportar o framework *PYNQ*, que é um projeto *open-source* da mesma empresa usar plataformas de desenvolvimento. Através da linguagem python e do uso de bibliotecas pode-se utilizar da lógica programável advinda do FPGA juntamente com o microprocessador para gerar circuitos das mais variadas aplicações. A placa é ideal para aplicações de alta vazão de dados (como processamento de vídeo, por exemplo) pelo fato de apresentar grande paralelismo devido à execução em hardware. Uma placa PYNQ pode usar o ambiente *Jupiter Notebook* para desenvolver aplicações, fazendo uso de bibliotecas de hardware chamados de *overlays* para acelerar o processamento.

A *Xilinx* criou a arquitetura *Zynq*, que combina uma FPGA com um processador ARM e entradas saídas em um único produto. A parte do processador ARM é chamada de *Processing System* (PS) e a parte da FPGA é chamada de *Programmable Logic* (PL). Um design de hardware da FPGA é chamado de *PYNQ overlay*, ele faz com que a aplicação em hardware (FPGA) fique disponível para ser utilizada em software (ARM). Por padrão a Pynq Z2 tem disponível um overlay de base que permite que o processador utilize os periféricos como mostra a Figura 15.

2.11 Bancada experimental

Foi desenvolvida pela ex-aluna Ana Paula Chavier Rodrigues da Universidade de Brasília uma bancada que comporta os componentes necessários para o projeto. A bancada

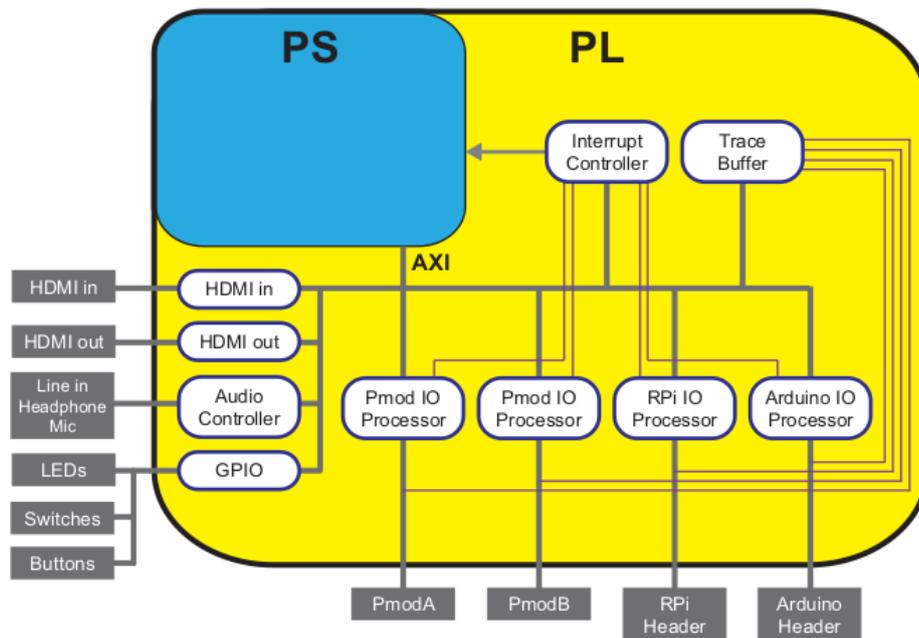


Figura 15 – Funcionalidades de interface implementadas pelo *base overlay*.

consiste em uma plataforma de MDF elevada por suportes de borracha com um suporte para o motor feito também de MDF. Dessa forma pode-se facilmente parafusar os dispositivos que são utilizados, tornando o sistema mais compacto e seguro para os testes a serem realizados.

Os trabalhos anteriores também consistiram em desenvolver um transferidor, juntamente com um cursor fabricados por meio de impressão 3D de modo a auxiliar no desenvolvimento do sistema de controle a ser implementado futuramente (vide Figura 16). Devido à pausa atribuída à pandemia, a bancada desenvolvida teve que passar por uma revitalização devido ao mal estado em que se encontrava.

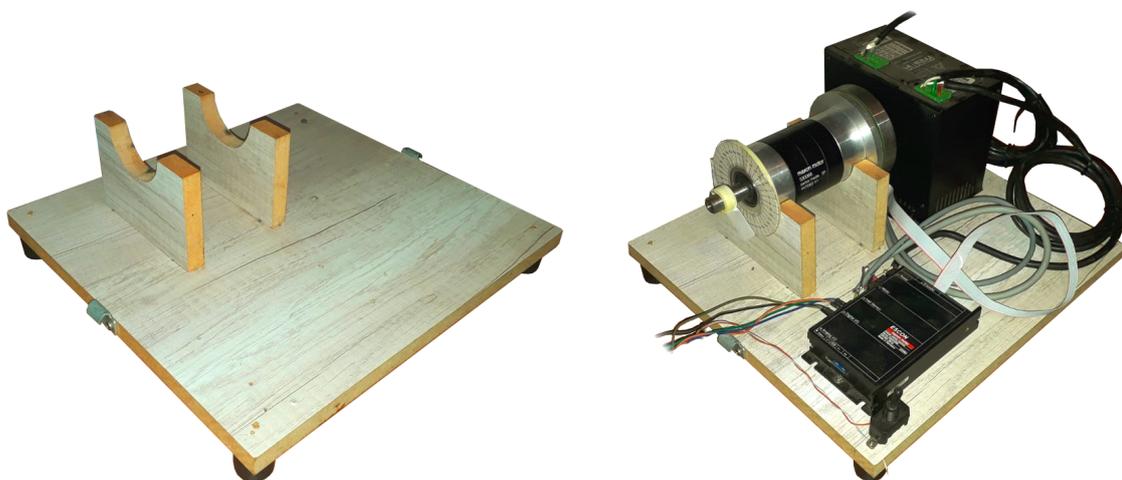


Figura 16 – Bancada desenvolvida pela Ana Paula.

3 MÉTODOS

3.1 Projeto

A etapa de projeto foi ser dividida em cinco partes:

- **Estudo do funcionamento da interface anterior.** Uma placa de interface já havia sido desenvolvida pela ex-aluna Ana Paula Chavier Rodrigues (RODRIGUES, 2017) da Universidade de Brasília e seu funcionamento foi utilizado como base para o desenvolvimento desse projeto.
- **Verificação dos requisitos de projeto.** Se fez necessário observar os requisitos que o sistema exigia para que a comunicação ocorresse de uma forma confiável e eficiente.
- **Estabelecimento dos requisitos da interface.** A partir dos requisitos do projeto, estabeleceu-se os requisitos dos componentes utilizados no projeto da interface e realizou-se a escolha dos produtos.
- **Implementação do esquemático.** A implementação do esquemático consistiu na relação dos componentes do circuito e foi realizada através do software *Altium Designer*.
- **Projeto do design.** O design consistiu na composição espacial dos componentes, em que se escolheu a melhor disposição de modo a otimizar espaço e facilitar o uso da placa. Também foi realizado no software *Altium Designer*.

3.2 Implementação

A etapa de implementação consiste em realizar a fabricação da placa projetada em território nacional, fazer a aquisição dos componentes necessários, montagem dos componentes e do cabeamento necessário para a etapa de teste e validação. Ao final dessa etapa tem-se todo o aparato necessário para iniciar-se a fase de teste e validação.

3.3 Teste e Validação

Nesta etapa, são realizados três passos:

- **Teste da placa com uso de osciloscópio.** Nesta etapa se verifica se os componentes do sistema estão satisfazendo os requisitos elétricos e temporais planejados para a interface.

- **Teste da placa placa secundária.** Aqui é realizado o teste da segunda placa acoplada à placa primária, os testes são similares aos testes com somente uma placa.
- **Validação do sistema utilizando o SoC.** Finalmente, ambas as placas são testadas e validadas com o uso do sistema montado com os sensores, atuadores e a placa controladora.

4 PROCEDIMENTOS INICIAIS

4.1 Preparativos

Para que a bancada pudesse ser estudada, foi necessária uma revitalização do equipamento, colocado de forma em que a disposição dos dispositivos ficasse de maneira mais adequada ao manuseio para testes. Algumas partes da bancada necessitaram de manutenção, como a troca de adesivos de fixação e o cursor que estava quebrado e precisou ser reconstruído, o que foi realizado utilizando materiais domésticos repositados como mostra a Figura 17.



Figura 17 – Reconstrução do cursor.

Em seguida foi necessário realizar um estudo das interfaces de conexões entre os dispositivos. Nesse estudo verificou-se as conexões, tensões e impedâncias que cada componente exige para seu funcionamento para que não houvesse risco de comprometimento de algum dos dispositivos utilizados.

Para realizar testes e estudos utilizou-se uma placa de controle Arduino MEGA, devido a sua compatibilidade de tensão e da facilidade que apresenta na prototipagem dos softwares. O Arduino MEGA também apresenta saídas PWM de 490 Hz, que é compatível com o do driver do motor, que aceita frequências entre 10Hz e 5kHz.

4.2 Configuração do Driver

Em seguida foi realizada a configuração do software *ESCON Studio*, que é o driver do *ESCON 70/10*. A configuração foi feita de acordo com as especificações do motor apresentados no link <WWW.MAXONGROUP.COM/MAXON/VIEW/PRODUCT/323772>. Após a configuração, fez-se a auto-tuning que o software provê, em que é realizada a caracterização

do motor de forma automática a partir de uma bateria de testes realizado pelo driver. Os gráficos de resposta da caracterização estão apresentados na Figura 18.

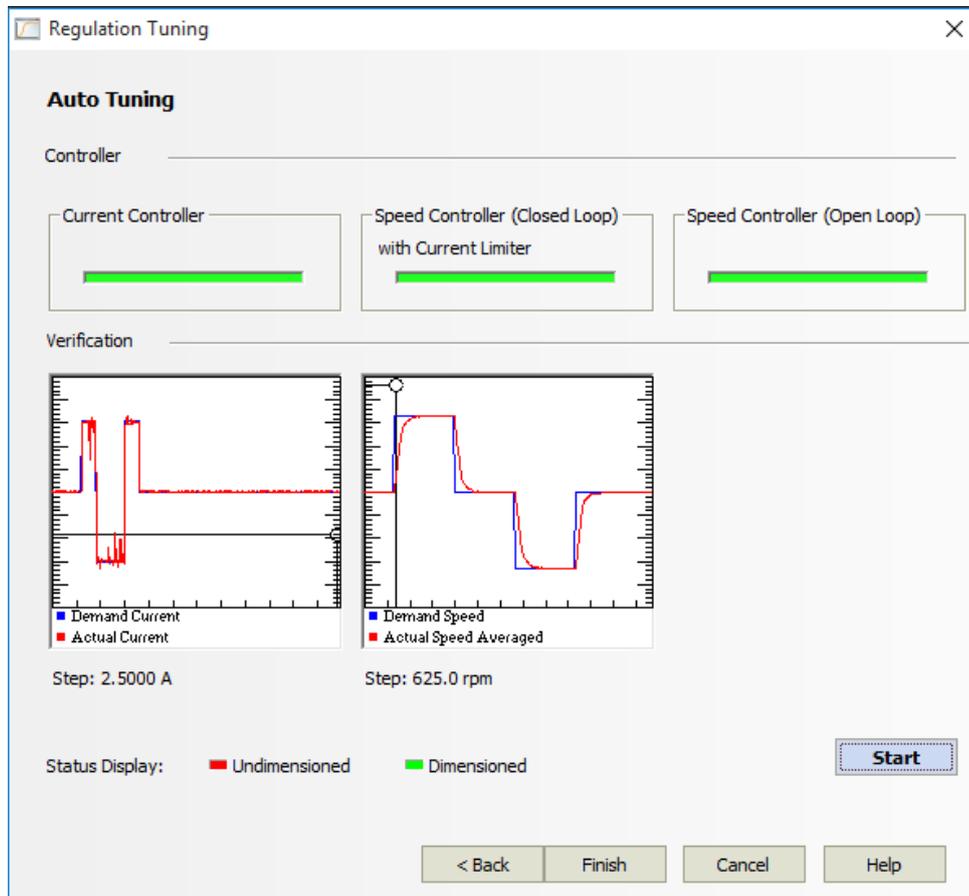


Figura 18 – Caracterização gerada no ESCON Stutio.

4.3 Procedimentos gerais

Esta etapa foi realizada durante o período de isolamento social devido à pandemia, portanto optou-se utilizar de ferramentas que evitassem o acesso aos laboratórios, de forma a maximizar o isolamento sem prejudicar o andamento do projeto. O objetivo desta etapa consistiu em fazer com que o sistema fosse colocado em funcionamento para a familiarização com o equipamento. Primeiramente, fez-se necessário o uso de um interruptor para gerar o sinal de *enable* que é enviado para o driver. Então instalou-se na bancada um interruptor, montado como mostrado na Figura 19.

De modo a testar o funcionamento da bancada, foi utilizada a plataforma Arduino como controladora intermediária para o início da familiarização com o equipamento. O Arduino utilizado foi o Arduino Mega2560, ele apresenta saída PWM a uma frequência de 490Hz, o que é compatível com a entrada do driver do projeto. Dessa forma pôde-se enviar os sinais de controle e observar o comportamento do sistema.

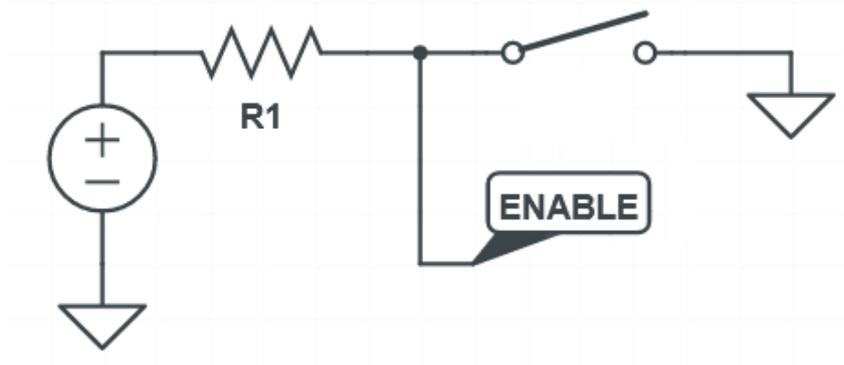


Figura 19 – Circuito da ligação da chave.

Na IDE do Arduino, foi implementado um algoritmo de *debounce* baseado em delays. O funcionamento se dá seguinte forma: uma vez que o sinal de *enable* foi detectado, o software aguarda um período curto de tempo (período suficiente para estabilizar as oscilações do interruptor) e se após esse período o sinal continuar em alto, o sinal *enable* é enviado ao driver. Vale observar na Figura 19 que pela montagem do circuito o sinal de *enable* é alto quando a chave está aberta.

Em seguida, utilizou-se a entrada serial do Arduino para capturar os dados e visualizar em gráficos os sinais do encoder. Como pode ser observado na Figura 20 a resolução do Arduino não é alta o suficiente para captar o sinal corretamente, já que se pode observar o efeito de aliasing na saída, mesmo com uma velocidade baixa no motor. Entretanto, a disposição dos cabos pode ser usada para o projeto da placa de interface que comportará o cabeamento dos dispositivos utilizados.

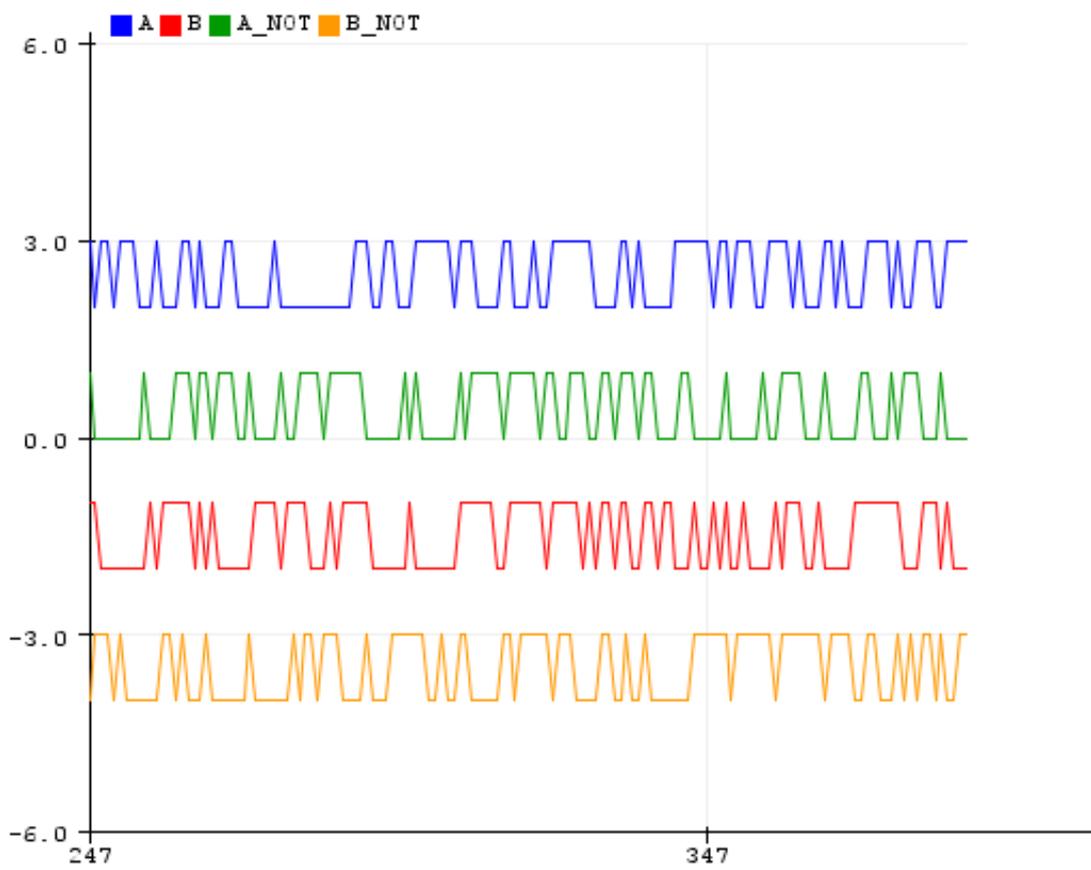


Figura 20 – Gráfico dos dados do encoder gerado pela IDE do Arduino.

5 PLACA DE INTERFACE

5.1 Projeto Base

O projeto do desenvolvimento de uma placa de interface foi iniciado utilizando como base o trabalho realizado pela ex-aluna da UnB Ana Paula Chavier Rodrigues (RODRIGUES, 2017). O projeto anteriores foi implementado com objetivos diferentes do projeto atual, portanto não tinha os requisitos que foram propostos posteriormente.

De modo a manter um bom ritmo no andamento do projeto a última versão do projeto anterior foi projetada para ter uma única camada. Ela teve suas trilhas desenhadas por uma fresadora LPKF ProtoMat S103 pertencente ao Instituto Federal de Brasília (IFB), campus de Taguatinga . A Figura da versão final da placa pode ser visualizada na Figura 21.

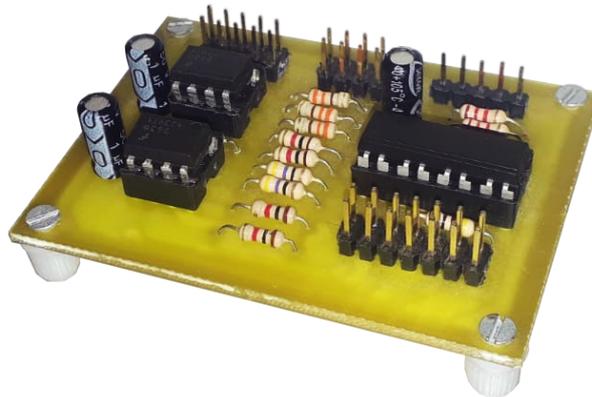


Figura 21 – Placa anterior utilizada como base.

A placa foi fabricada para ser modular, de modo a comportar um motor, ou seja, para controlar os quatro atuadores seriam necessárias quatro dessas placas. Pelo fato de utilizar uma camada, o design exigiu que alguns cabos fossem customizados de maneira espelhada de modo a permitir a extensão do sinal para análise (Figura 22).

A placa foi desenvolvida para ser utilizada com uma SoC Zynq 7000, que apesar de apresentar a arquitetura semelhante (dual-core ARM Cortex-A9 + FPGA) apresenta interfaceamento bastante diferente do que a placa Pynq Z2, escolhida posteriormente para o projeto, por esse motivo, os conectores devem ser selecionados de modo a serem os mais adequados de forma a acoplar com a Pynq Z2.

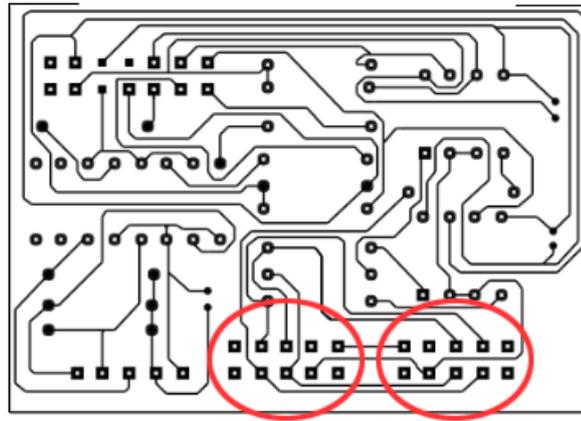


Figura 22 – Conexões espelhadas na placa anterior (RODRIGUES, 2017).

5.2 Requisitos Gerais do projeto

Pelo motivo da placa PYNQ utilizar uma tensão lógica de 3,3 Volts e o driver, juntamente aos encoders do motor utilizam uma tensão de 5V, se faz necessário o uso de alguma estratégia para realizar o acoplamento dos elementos. Para facilitar a manutenção do dispositivo em caso de falha elétrica, optou-se por utilizar elementos optoacopladores, pois estes dispositivos fazem com que a corrente não se propague para a próxima etapa do circuito, impedindo que queime os elementos seguintes, independentemente do lado do circuito em que ocorra tal falha elétrica.

De modo a facilitar a montagem do sistema se faz necessário o uso de conectores específicos e devidamente rotulados para cada dispositivo. Isso é importante para que o interfaceamento fique transparente aos integrantes que venham a trabalhar no projeto futuramente, evitando erros que possam danificar o equipamento e mantendo a fluidez do desenvolvimento. Cada placa deve comportar dois motores, serão utilizadas duas placas idênticas em que cada uma controlará um membro inferior/perna. No total, portanto, o sistema será controlado por uma única placa Pynq Z2, com duas placas de interface que controlam no total quatro motores.

Além disso existem os requisitos de frequência. O Figura 23 retirado da referência (ROBBI, 2018) apresenta uma média da posição angular da junta do joelho. A partir dessa informação podemos estimar a velocidade angular máxima da junta traçando as retas representando a derivada em vermelho e verde, assumindo que o ciclo de uma marcha dura cerca de um segundo (NOVAES; MIRANDA; DOURADO, 2011), temos que a velocidade angular máxima da junta do joelho é de 1,49 revoluções por segundo. Em seguida, tendo que o número de contagens por revolução do motor é de 3200 e o fator da caixa redutora é de 100, obtém-se que a frequência do sinal no pior caso será de 470kHz. Portanto, o sinal advindo do encoder para a Pynq Z2 deve estar preservado para uma frequência nessa faixa.

$$FreqPiorCaso = VelAngJunta \times FatorReducao \times ContagensPorRevolucaoEncoder$$

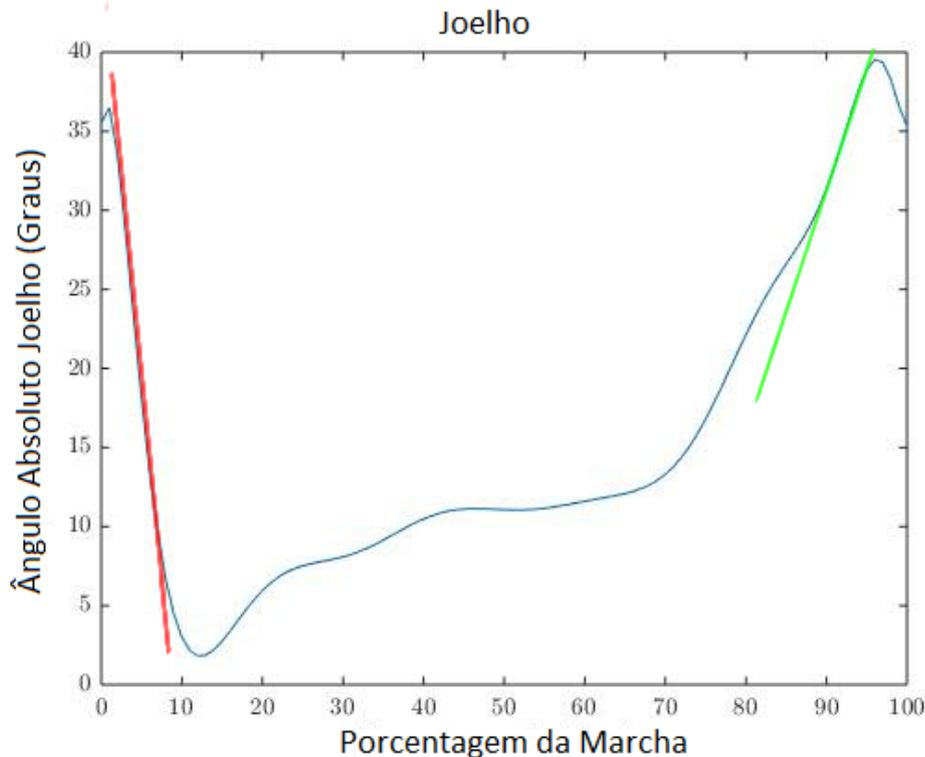


Figura 23 – Posição angular das juntas do joelho ao longo da marcha (ROBBI, 2018).

Já o sinal advindo da Pynq Z2 para o driver é mais permissivo, pois o driver aceita frequências mais baixas, pelo menos 10 Hz. Desta forma, ao selecionarmos uma frequência de 100 Hz com *duty cycle* entre 10% e 90%, fazemos com que o requisito seja que o sinal esteja preservado para um sinal com frequência de 1kHz. Mesmo assim esse valor é um valor arbitrário selecionado, e caso seja necessário pode-se configurar o driver para um intervalo de *duty cycle* menor ou diminuir a frequência do PWM.

Por fim, vale observar que os requisitos não estavam todos especificados durante o desenvolvimento do projeto, boa parte das decisões foram tomadas a partir da análise de cada iteração das placas que eram projetadas em software, para em seguida poder-se realizar a fabricação e verificar o cumprimento dos requisitos.

5.3 Histórico de desenvolvimento

Para o desenvolvimento do projeto da PCB utilizou-se o software *Altium Design*, este software foi escolhido dentre os demais softwares padrões da indústria principalmente por apresentar visualização 3D do produto final, o que permite visualizar a disposição dos

conectores o que permite prever eventuais problemas de montagem e conexão no produto final.

De modo a manter simples e consequentemente mais barata a fabricação da placa, o projeto teve como alvo uma tecnologia de duas camadas. Através de algumas tentativas de roteamento das trilhas no software ficou evidente que seria possível utilizar apenas este número de camadas e manter um tamanho satisfatório do *layout* da placa. Com duas camadas se superariam algumas limitações contidas na placa desenvolvida anteriormente utilizada como base.

A primeira iteração foi desenvolvida de maneira mais ingênua, visando familiarização com o software. Nesta etapa a placa apresenta as mesmas funcionalidades do que a placa do projeto utilizado como base, com a diferença de que agora a placa comporta dois motores. A visualização 3D desta etapa está representado na Figura 24. Nesta representação os resistores e conectores (que seriam do tipo *pin header*, como no projeto antecessor) não estão inclusos. O resultado do projeto pode ser visualizado na Figura 25.

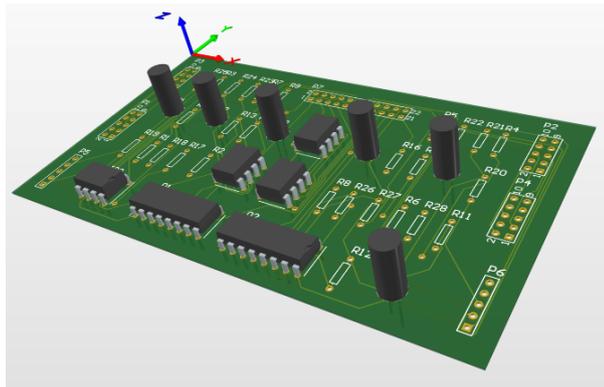


Figura 24 – Representação 3D do protótipo.

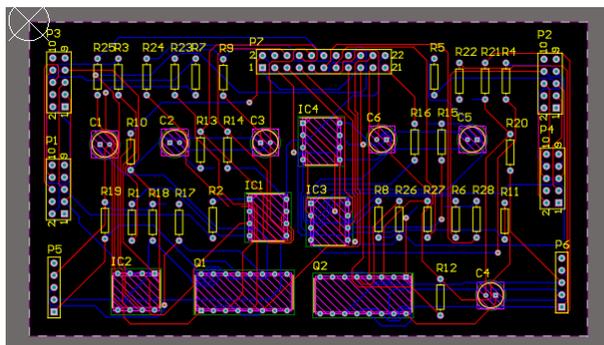


Figura 25 – Representação 2D do protótipo.

Apesar de funcional para dois motores, o projeto agora apresentava algumas questões problemáticas. Uma delas era que a placa Pynq Z2 não apresentava terminais suficientes para fornecer sinais para duas placas idênticas separadamente e assim controlar os quatro motores simultaneamente. Como pode-se observar na Tabela de inputs e outputs 3, seriam necessários 7 sinais GPIO por motor (4 inputs e 3 outputs), totalizando em 14 sinais por

placa, sem levar em conta a alimentação e terra. Portanto não haveria forma de que a Pynq Z2 controlasse placas idênticas com os sinais saindo diretamente dela.

Tabela 3 – Sinais de input e output por motor

Inputs	Outputs
Sinal A	Enable
Sinal \bar{A}	PWM
Sinal B	Direção
Sinal \bar{B}	

A placa Pynq Z2 comporta varias entradas e saídas de propósito geral (GPIO), porém optamos por implementar uma interface compatível com dispositivos de desenvolvimento como Raspberry pi (como mostra a Figura 26) e o problema das conexões foi resolvido graças ao procedimento de Daisy Chain. Dessa forma a montagem fica intuitiva e são utilizados menos cabos e conectores. A estratégia de *daisy chain*, está representada através do diagrama 27 e a estratégia de conexões utilizadas pode ser vista na Tabela 4.

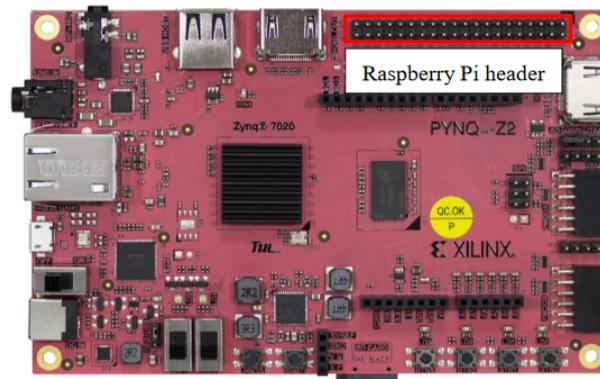


Figura 26 – Conexão compatível com a Raspberry pi na Pynq Z2.

De modo a acomodar melhor os cabos, bem como aprimorar a usabilidade do usuário foi discutido que a interface de encaixe poderia ser aprimorada, já que não ficaria claro de imediato as conexões e a orientação delas no momento da montagem do sistema. Conseqüentemente, as conexões que utilizavam *pin headers* foram todas substituídas por conectores mais adequados. O terminal de output para o motor, contendo VCC, GND e três sinais de controle configurados no driver (*enable*, direção e PWM) foi substituído por dois (um para cada motor) blocos terminal plugável de 6 pinos, que é o mesmo utilizado pelo driver. Os demais conectores utilizados são do tipo GPIO.

Outras alterações também foram feitas ao projeto, como proporções melhores da placa para otimizar espaço, melhor disposição dos componentes e espessura de trilha mais grossa, passando a ter uma espessura mínima de 0,5mm. Os capacitores de $1\mu\text{F}$ foram trazidos para próximo dos seus respectivos optoacopladores, como sugere o *datasheet*. Também foram adicionadas informações extras na camada de rótulos dos encaixes, para agilizar o uso e

Tabela 4 – Conexões Pynq Z2

Conexão	Num	Pinos Zinq		Num	Conexão
N/C	1	3.3V	5V	2	N/C
Daisy Mot1 A	3	W18	5V	4	N/C
Daisy Mot1 B	5	W19	GND	6	N/C
Daisy Mot1 Ā	7	V6	Y18	8	Mot1 A
N/C	9	GND	Y19	10	Mot1 B
Daisy Mot1 B̄	11	U7	C20	12	Mot1 Ā
Daisy Mot1 PWM	13	V7	GND	14	N/C
Daisy Mot1 Dir	15	U8	W6	16	Mot1 B̄
VCC Zybo	17	3.3V	U18	18	Mot1 PWM
Daisy Mot1 En	19	V8	GND	20	N/C
Daisy Mot2 A	21	V10	U19	22	M1 Dir
Daisy Mot2 B	23	W10	F19	24	M1 Enable
N/C	25	GND	F20	26	Mot2 A
Daisy Mot2 Ā	27	Y16	Y17	28	Mot2 B
Daisy Mot2 B̄	29	Y6	GND	30	N/C
Daisy Mot2 PWM	31	Y7	B20	32	Mot2 Ā
Daisy Mot2 Dir	33	W8	GND	34	N/C
Daisy Mot2 En	35	Y8	B19	36	Mot2 B̄
Mot2 Enable	37	W9	A20	38	Mot2 PWM
N/C	39	GND	Y9	40	Mot2 Dir

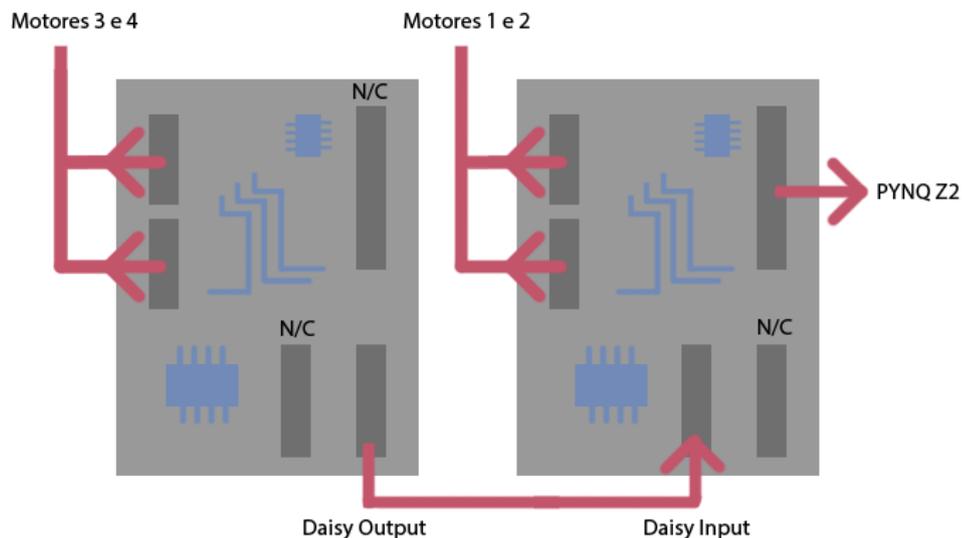


Figura 27 – Diagrama representativo do daisy chain.

evitar erros na montagem do sistema. Os Layouts finais em 2D e em 3D estão representados nas Figuras 28 e 29, respectivamente.

Como próxima etapa na evolução do projeto, observou-se que algumas melhorias poderiam ser realizadas de modo a tornar o projeto mais robusto e prático. Primeiramente se observou que a placa poderia apresentar furos em suas extremidades de modo a permitir a

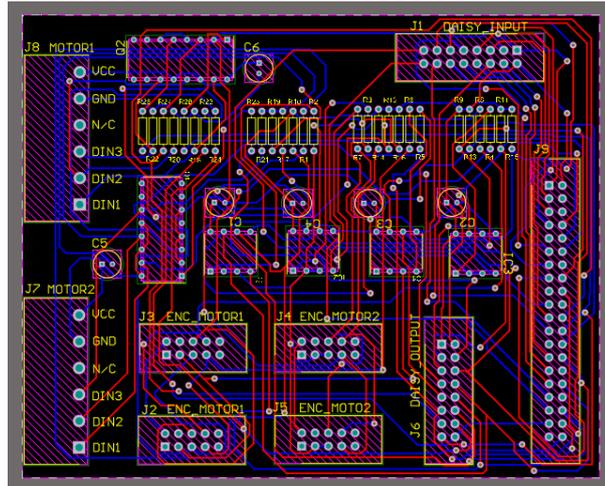


Figura 28 – Segunda iteração do projeto da placa em 2D.

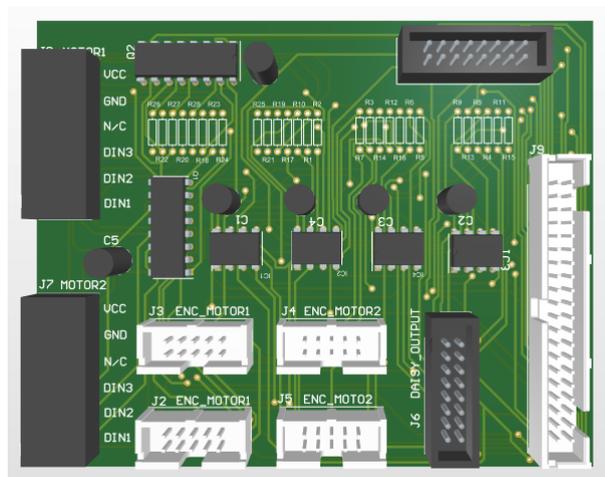


Figura 29 – Segunda iteração do projeto da placa em 3D.

montagem na bancada por meio de parafusos. Um potencial problema desse último design consiste no fato de que há trilhas que passam por regiões muito próximas à borda da placa. Para adereçar estes problemas descritos, expandiu-se a borda da placa e posicionou-se quatro furos para montagem nas extremidades dela.

De modo a reduzir as interferências eletromagnéticas, houve-se a necessidade da implementação de planos de terra. Como a placa comporta vários dispositivos diferentes, foi necessário criar plano de terra para cada dispositivo acoplado. Cada lado de cada optoacoplador advém de uma alimentação específica, é possível separar em cinco regiões as quais se tem as partes em que cada componente acoplado interage com o optoacoplador. De modo a separar espacialmente essas regiões na placa foi necessário reestruturar os componentes para que cada um deles ficasse em sua correspondente região de alimentação e terra. O diagrama da Figura 30 demonstra qual foi a disposição dos componentes optada visando a região de cada componente acoplado.

Nesta etapa também verificou-se que alguns resistores utilizados no projeto base

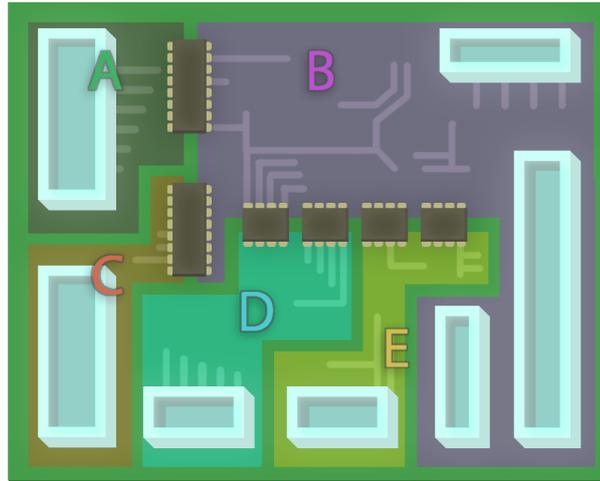


Figura 30 – Diagrama dos planos de terra. A) Motor 1; B) Pynq Z2; C) Motor 2, D) Encoder 1; E) Encoder 2.

apresentavam resistências abaixo do ideal, que poderia comprometer a vida útil dos optoacopladores, bem como fazer com que a corrente exigida da placa Pynq fosse mais alta do que ela pudesse fornecer. Portanto foram selecionados e testados com os componentes optoacopladores os resistores com os novos valores previamente à fabricação da placa. Fez-se portanto o cálculo das correntes do circuito para verificar a necessidade de uma fonte de alimentação externa.

Os circuitos que consomem alimentação da placa Pynq Z2 estão representados na Figura 31, para cada placa. A tensão utilizada pela pynq nesse projeto é de 3.3V, portanto utilizando a lei de Ohm, temos que a corrente do circuito do diodo (à esquerda) é de 7,02mA, enquanto a corrente do circuito do transistor (à direita) é de 2,75mA, assumindo que a resistência de entrada para a Pynq Z2 no modo de alta impedância é ordens de grandeza maior do que as resistências em questão.

No total, sendo o circuito de *output* utilizado seis vezes e o circuito de input utilizado oito vezes temos que a corrente total consumida pela Pynq Z2 no pior caso seria de 64,12mA por placa, totalizando 128,24mA para as duas placas. Esse valor encontra-se abaixo do valor máximo de corrente que a placa Pynq Z2 pode fornecer, fazendo com que não seja necessário o uso de uma fonte externa como *driver* para fornecer mais corrente ao circuito.

Durante a validação do projeto antes da liberação para a produção, observou-se também um erro na documentação oficial da placa Pynq Z2, que faz com que o projeto tivesse uma nova iteração de design devido a isso, portanto vale ser observada de modo a impedir que isso cause algum problema na utilização da placa no projeto por aqueles que estarão futuramente nesse projeto. Foi verificado que o pino 1 dos canais compatíveis com Raspberry PI corresponde ao pino VCC 3.3V, e não como GND, como mostra na documentação oficial da placa.

Por fim, dadas as correções e os aprimoramentos especificados, a versão final da

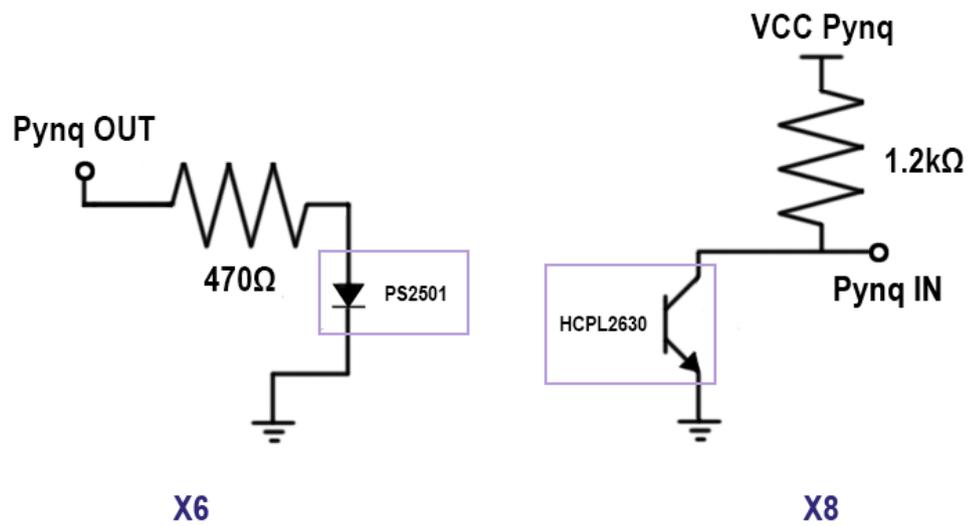


Figura 31 – Visualização dos circuitos alimentados pela Pynq Z2

G	W9	Y8	W8	Y7	Y6	Y16	G	W10	V10	V8	V	U8	V7	U7	G	V6	W19	W18	G
39	37	35	33	31	29	27	25	23	21	19	17	15	13	11	9	7	5	3	1
40	38	36	34	32	30	28	26	24	22	20	18	16	14	12	10	8	6	4	2
Y9	A20	B19	G	B20	G	Y17	F20	F19	U19	G	U18	W6	G	C20	Y19	Y18	G	V	V

É na verdade V 3.3V

G	Ground
V	3.3V
V	5V
...	Raspberry Pi header pin number
...	Zynq Pin

Figura 32 – Errata da documentação.

placa pode ser visualizada através da Figuras 33

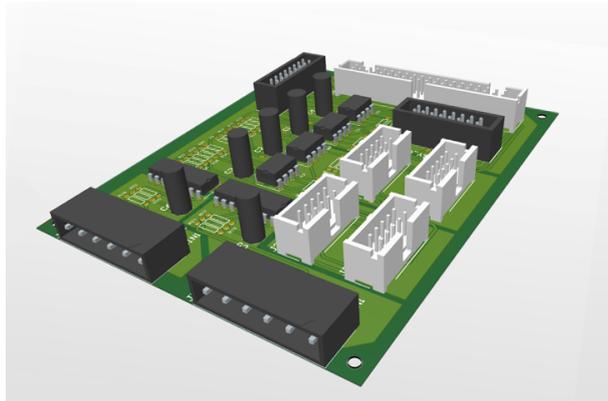


Figura 33 – Visualização em perspectiva da placa final.

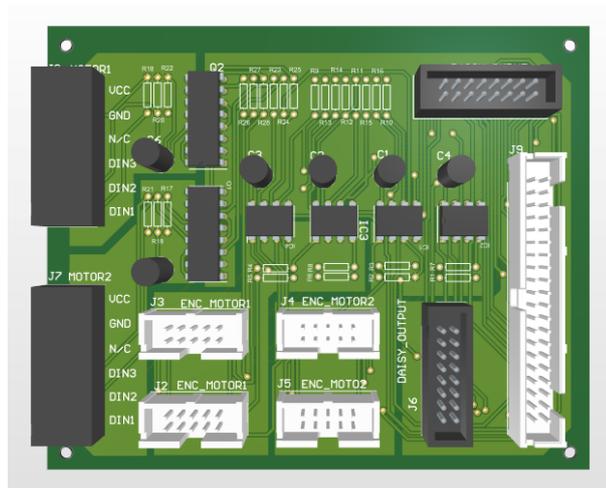


Figura 34 – Visualização por cima da placa final.

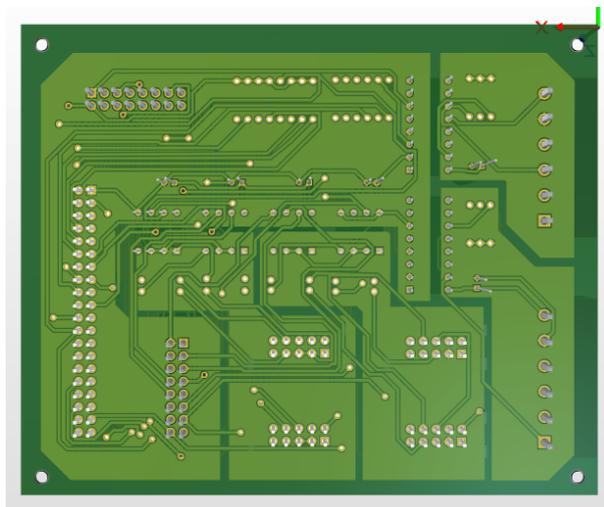


Figura 35 – Visualização por baixo da placa final.

6 Resultados

A Figura 36 apresenta a placa fabricada e a Figura 37 mostra a bancada final de testes com as duas placas montadas. Os testes foram realizados primeiramente com a placa principal e em seguida com a placa em *daisy chain*. Foi observado que houve um erro de projeto durante as testagens, em que os pinos de alimentação e terra foram invertidos na saída do *daisy output*, que compromete o funcionamento da placa secundária. Entretanto esse erro é facilmente corrigível fazendo um ajuste em uma das placas. A imagem 38 mostra o erro contido no projeto juntamente com a correção feita na placa.

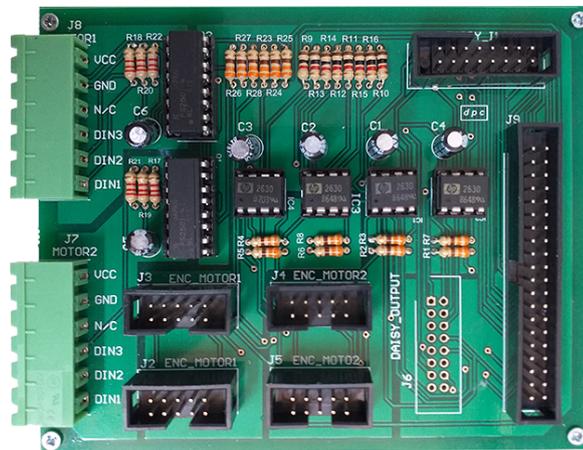


Figura 36 – Placa fabricada e com componentes montados.

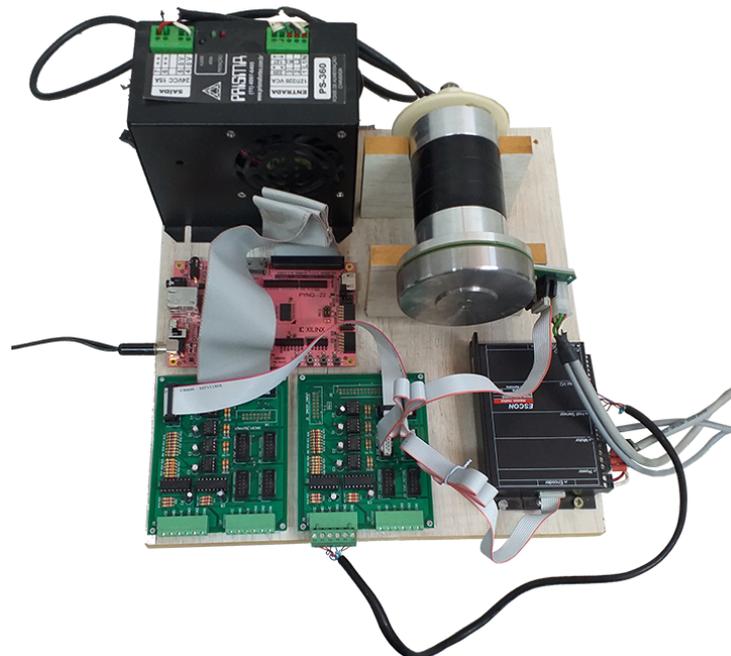


Figura 37 – Bancada final de testes.

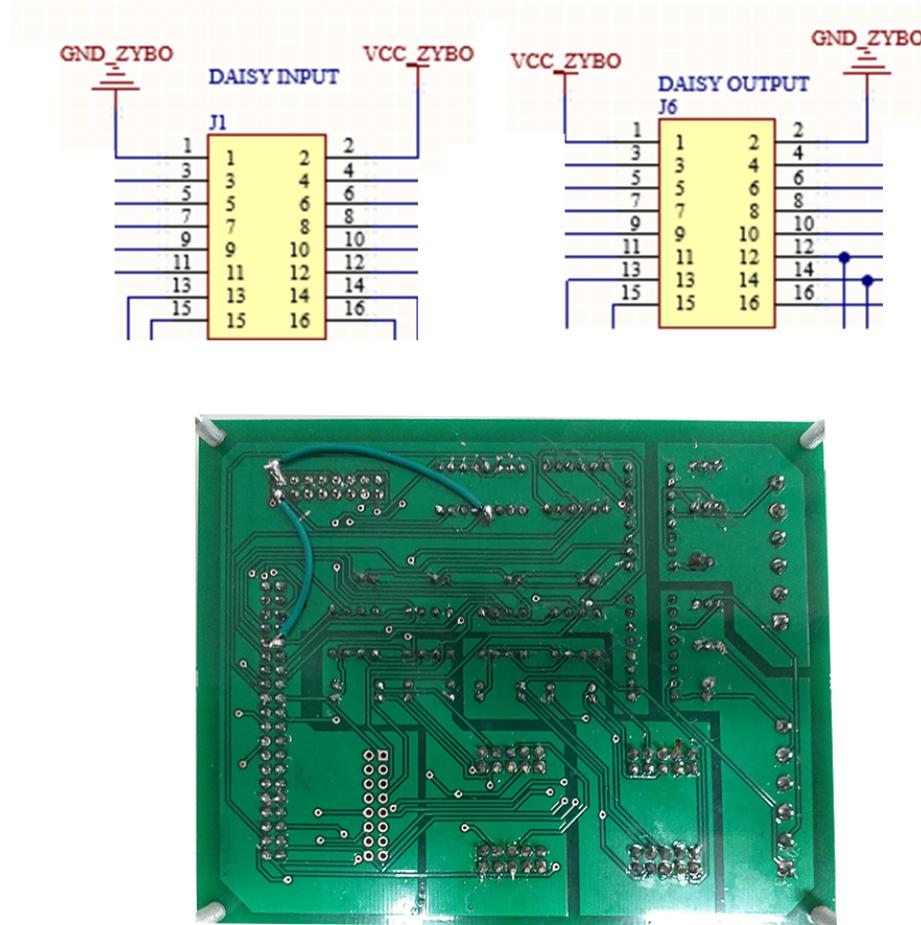


Figura 38 – Erro e correção feita em uma das placas.

Com uso de um osciloscópio e um gerador de função, foram realizados alguns testes no LEIA. Primeiramente, para visualizar a forma de onda obtida ao fornecer uma onda quadrada como entrada pela entrada do pino em que a Pynq Z2 irá transmitir os sinais de PWM, Enable e Direção. As montagens, para cada um dos motores estão apresentadas nas figuras 39, 40, 41 e 42, sendo que o gerador de função gerava uma onda quadrada com amplitude de 3,3 Volts. Os resultados de cada um desses sinais, tanto na placa principal quanto na placa em *daisy chain* não apresentaram diferenças notáveis entre si, portanto alguns gráficos dos resultados foram omitidos.

As Figuras 43 e 44 apresentam resultados para uma entrada de onda quadrada a 1kHz e 3kHz respectivamente. Pode-se observar que para esse sinal o tempo que dura o estado transiente é de em torno de 50 μ S. O valor especificado neste documento para o funcionamento adequado do sistema foi de 1kHz para sinal. A partir da Figura 43 pode-se observar que esse requisito foi cumprido e que o sinal segue íntegro na saída do dispositivo.

Em seguida foram feitos os testes para o sinal gerado pelo encoder que será fornecido à Pynq Z2, montados como mostram as Figuras 45, 46, 47 e 48, tendo o gerador de função uma onda quadrada com amplitude de 5 Volts. Como não apresentaram diferenças significativas

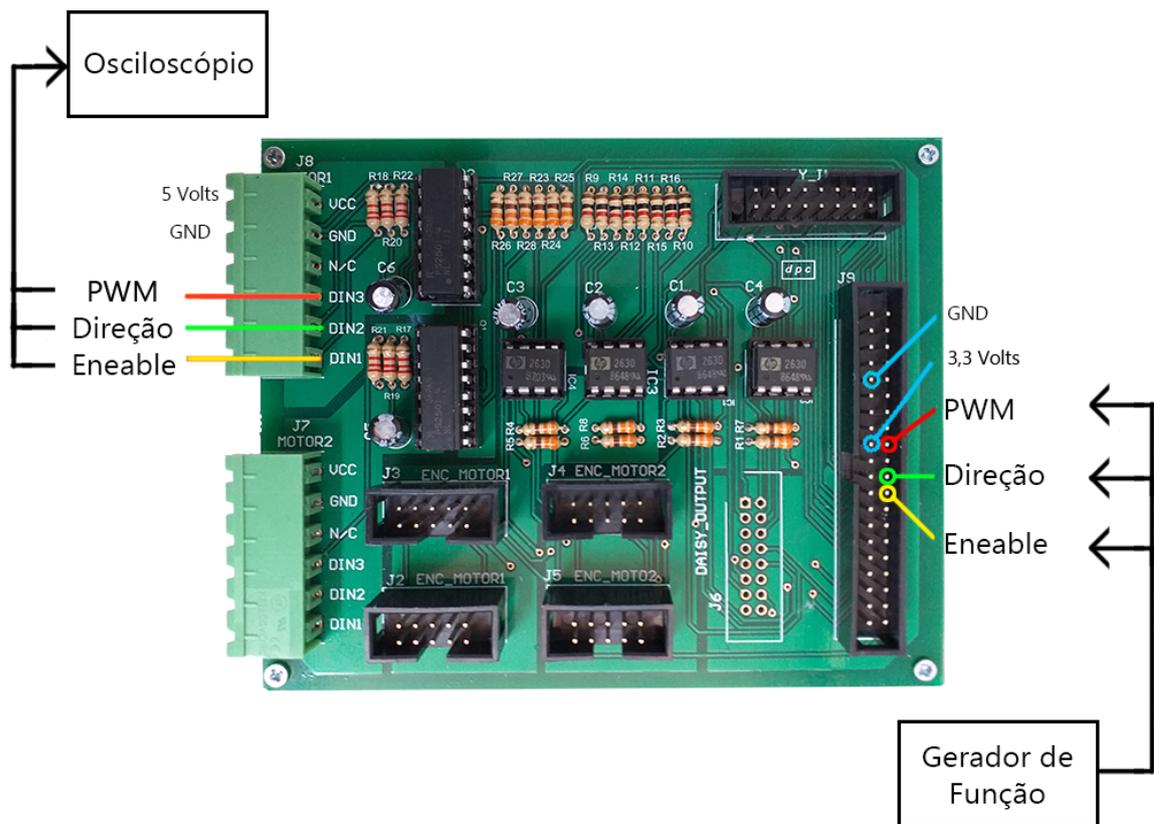


Figura 39 – Montagem do teste para o motor 1.

entre si, alguns gráficos dos resultados foram omitidos.

Os sinais resultantes estão apresentados nas Figuras 49 e 50 para as frequências de 1kHz e 1MHz, respectivamente. Pode-se observar que o transiente para este sinal dura cerca de 100 nanosegundos. A partir da observação da Figura 50 pode-se constatar que o componente se encontra dentro do funcionamento especificado, já que a frequência especificada para o funcionamento adequado foi de 470kHz e o sinal observado segue preservado a uma frequência acima do dobro dela.

Em seguida foram realizados os testes de atraso. As Figuras 51 e 52 apresentam os tempos de resposta para um sinal em degrau. Pode-se observar que o delay é aproximadamente igual ao estado transiente em ambos os casos.

Por fim, realizou-se a validação, em que a partir do conector que se encaixa a controladora Pynq Z2, estimulou-se com um sinal PWM indicando a velocidade de cada entrada de cada motor (tanto direta quanto em daisy chain) e se observou nesse mesmo conector os sinais gerados pelo encoder. As conexões do encaixe estão apresentadas na Figura 53.

Os arquivos de esquemático e design podem ser encontrados no repositório do GitLab <<https://gitlab.com/leiaunb/exoesqueleto>> referente ao projeto do exoesqueleto, juntamente com um vídeo demonstrativo do projeto em funcionamento.

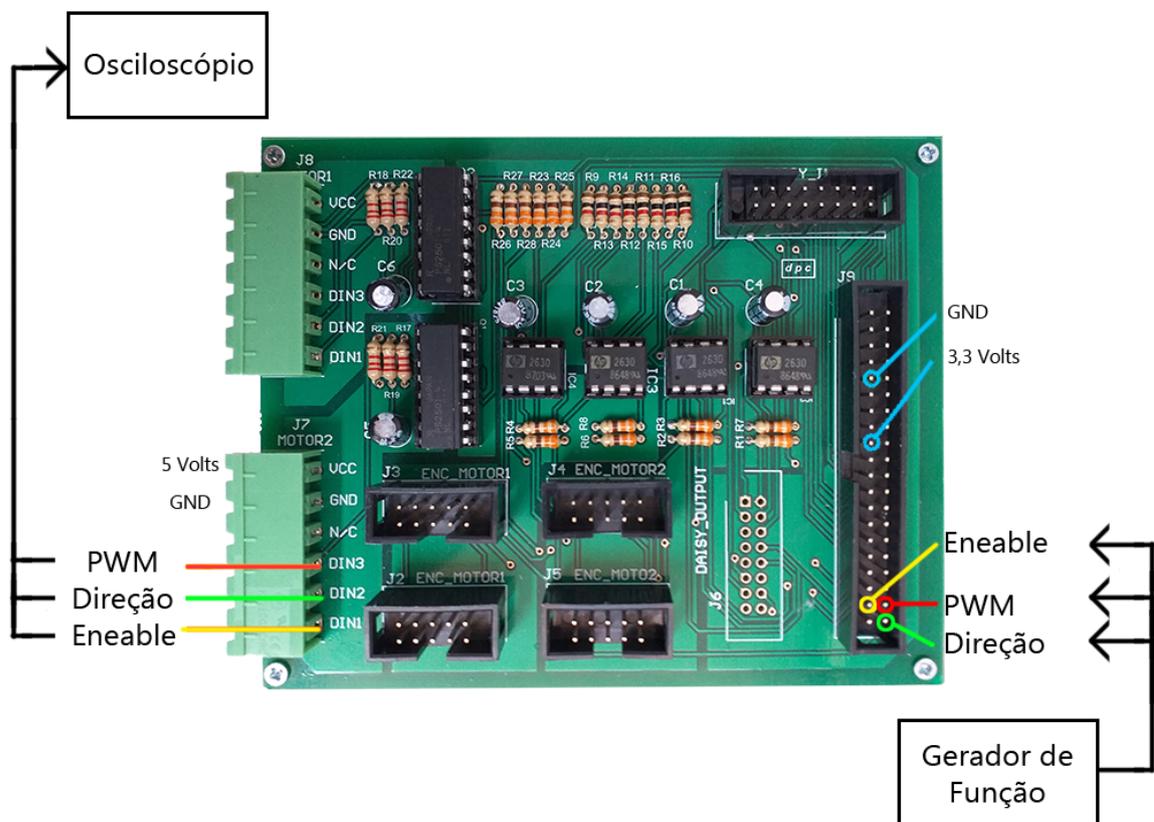


Figura 40 – Montagem do teste para o motor 2.

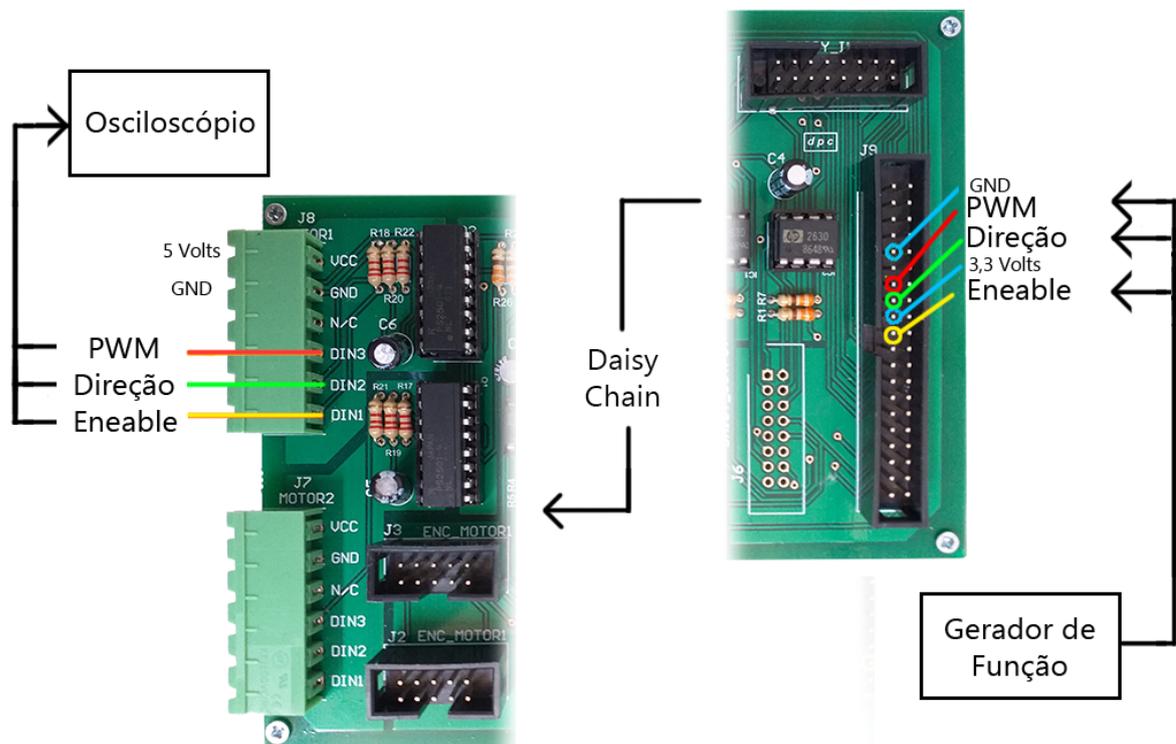


Figura 41 – Montagem do teste para o motor 1 em daisy chain (ou motor 3).

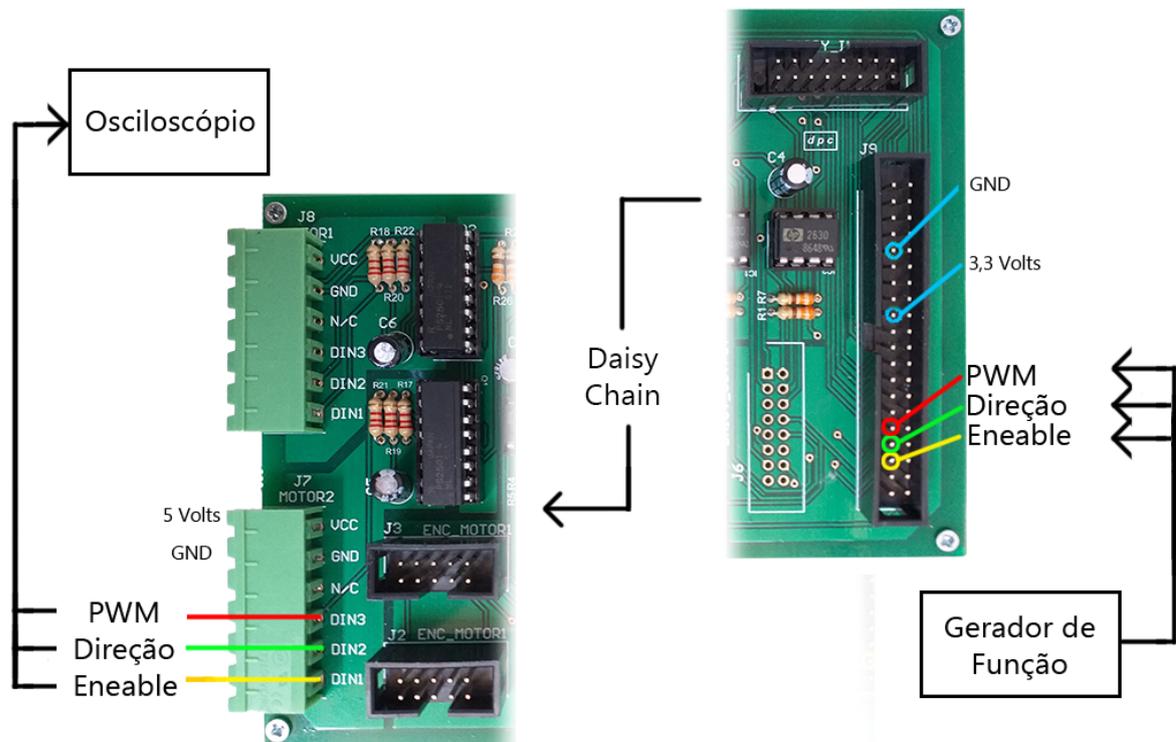


Figura 42 – Montagem do teste para o motor 2 em daisy chain (ou motor 4).

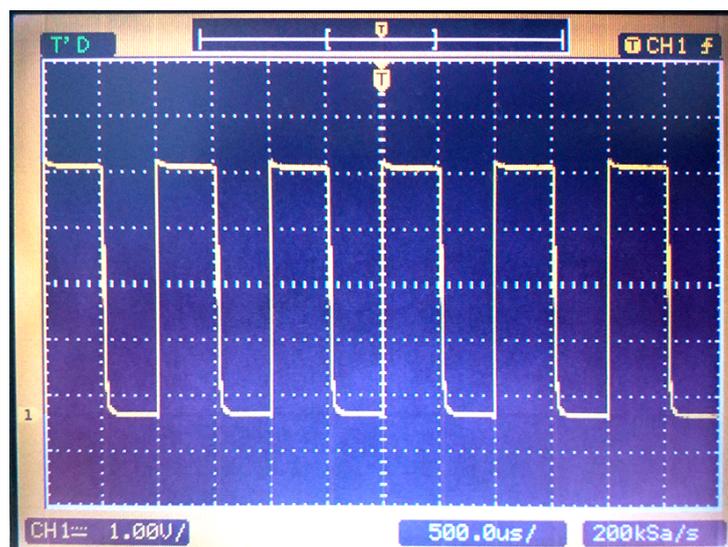


Figura 43 – Entrada para Driver dada forma de onda de 1kHz.

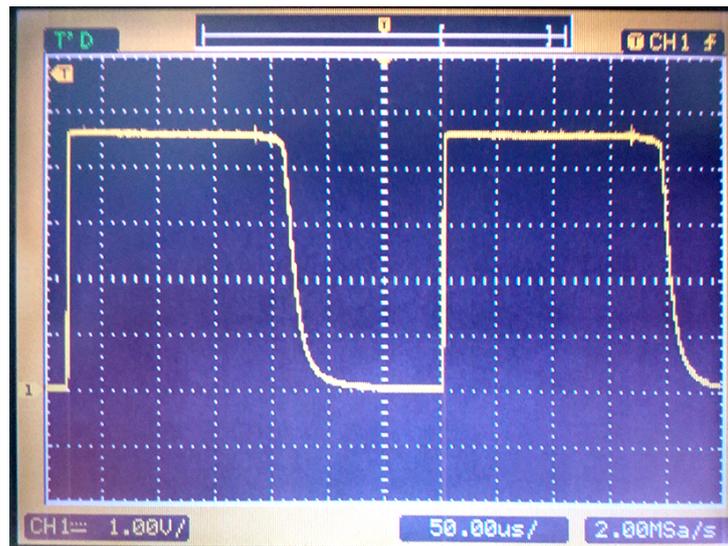


Figura 44 – Entrada para Driver dada forma de onda de 3kHz.

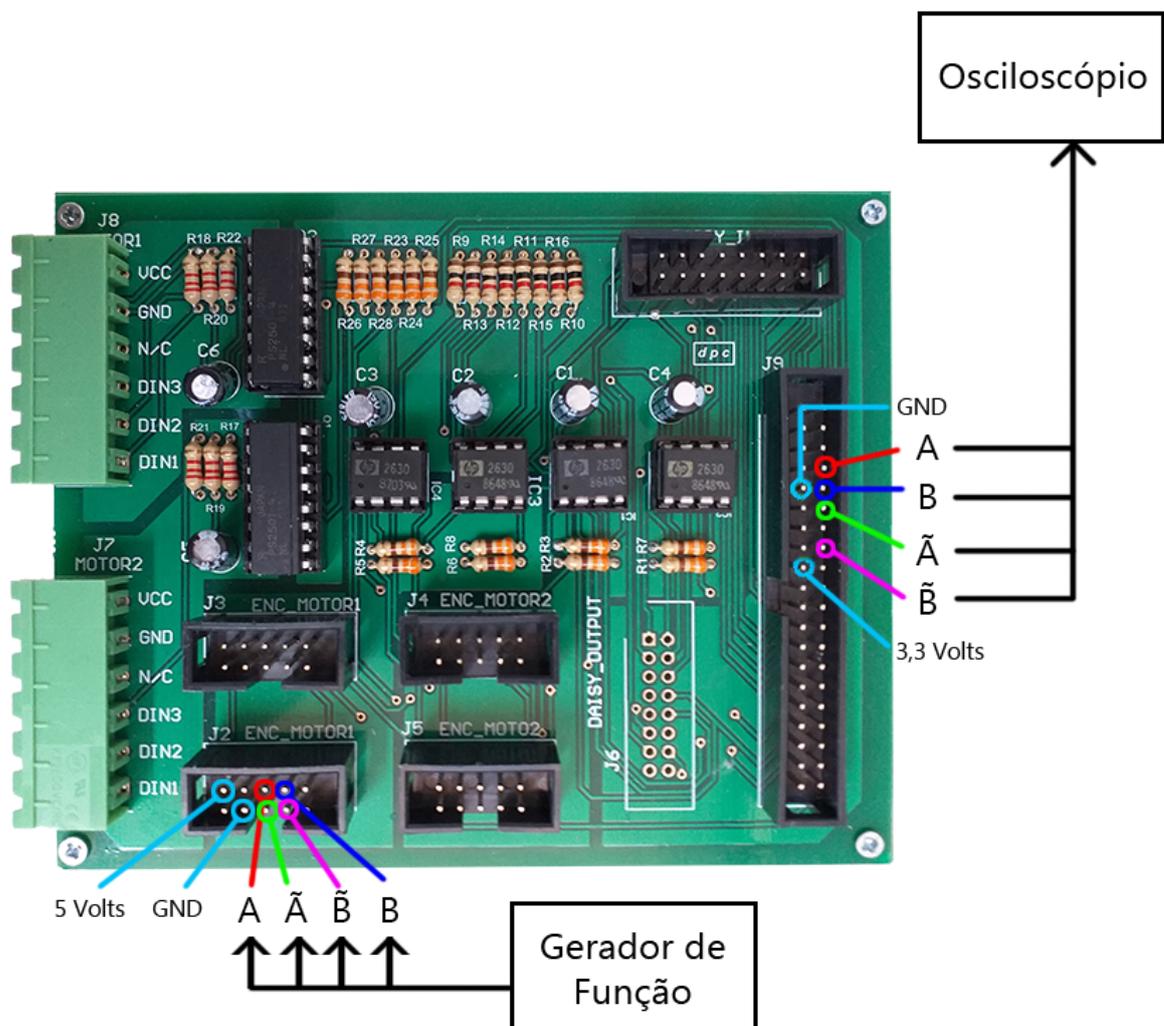


Figura 45 – Montagem do teste para o encoder do motor 1.

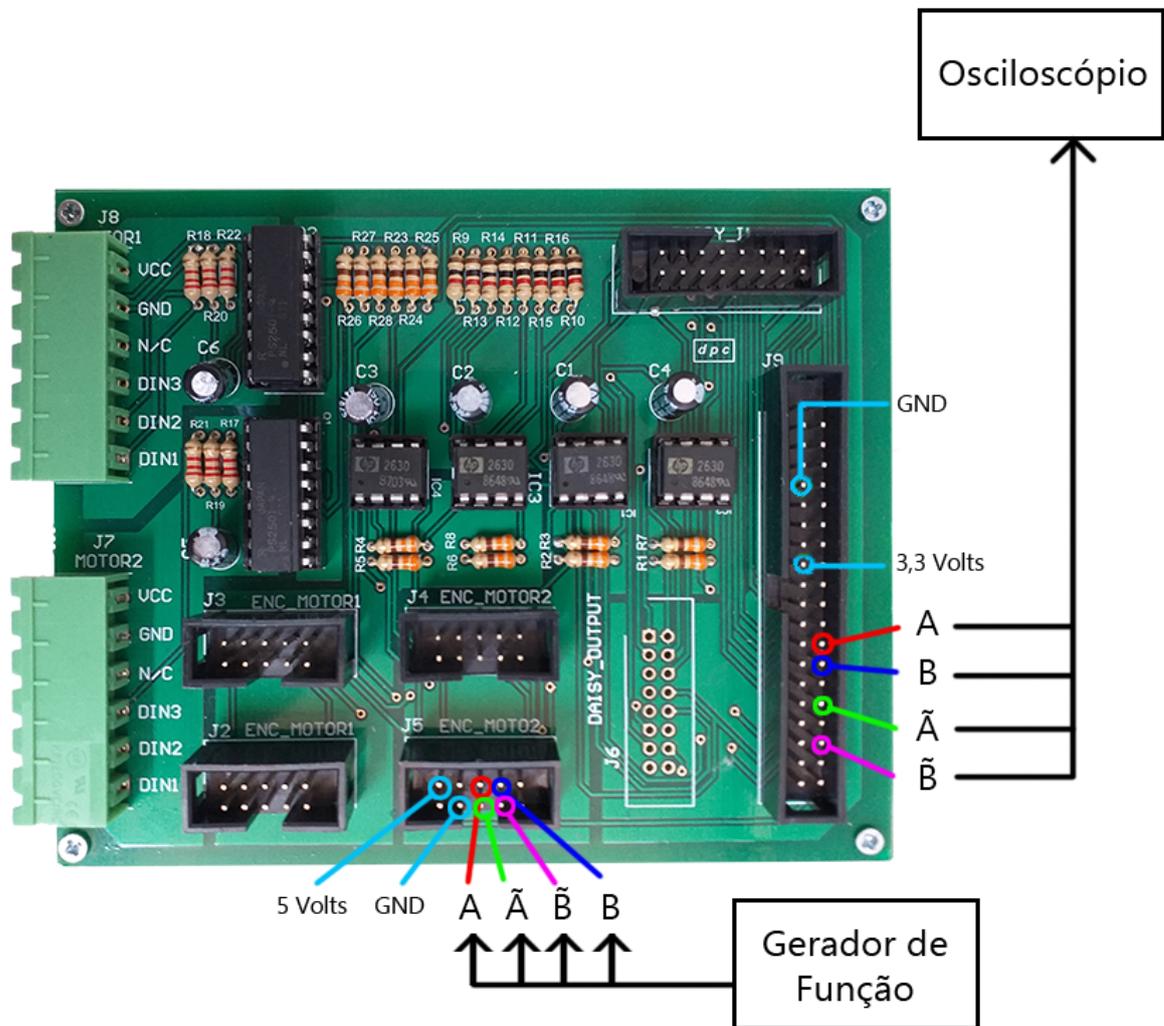


Figura 46 – Montagem do teste para o encoder do motor 2.

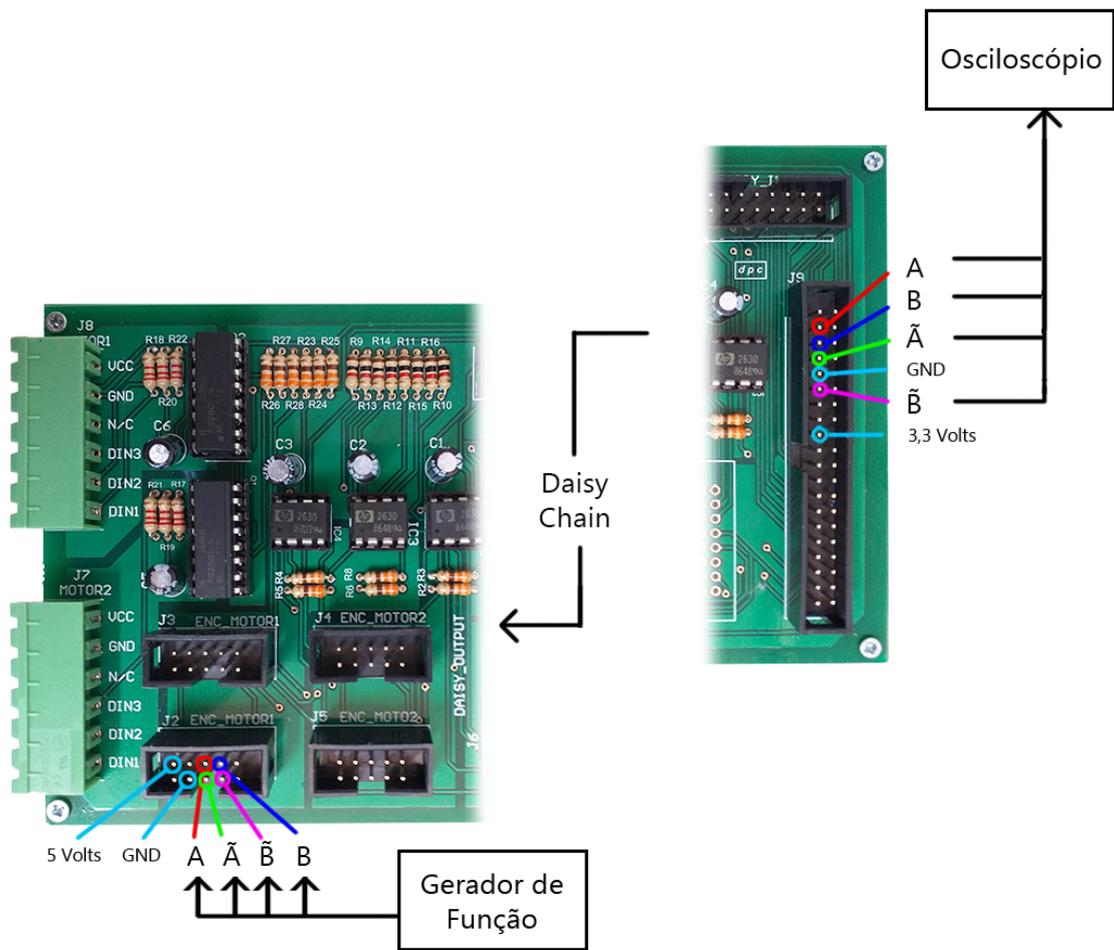


Figura 47 – Montagem do teste para o encoder do motor 3.

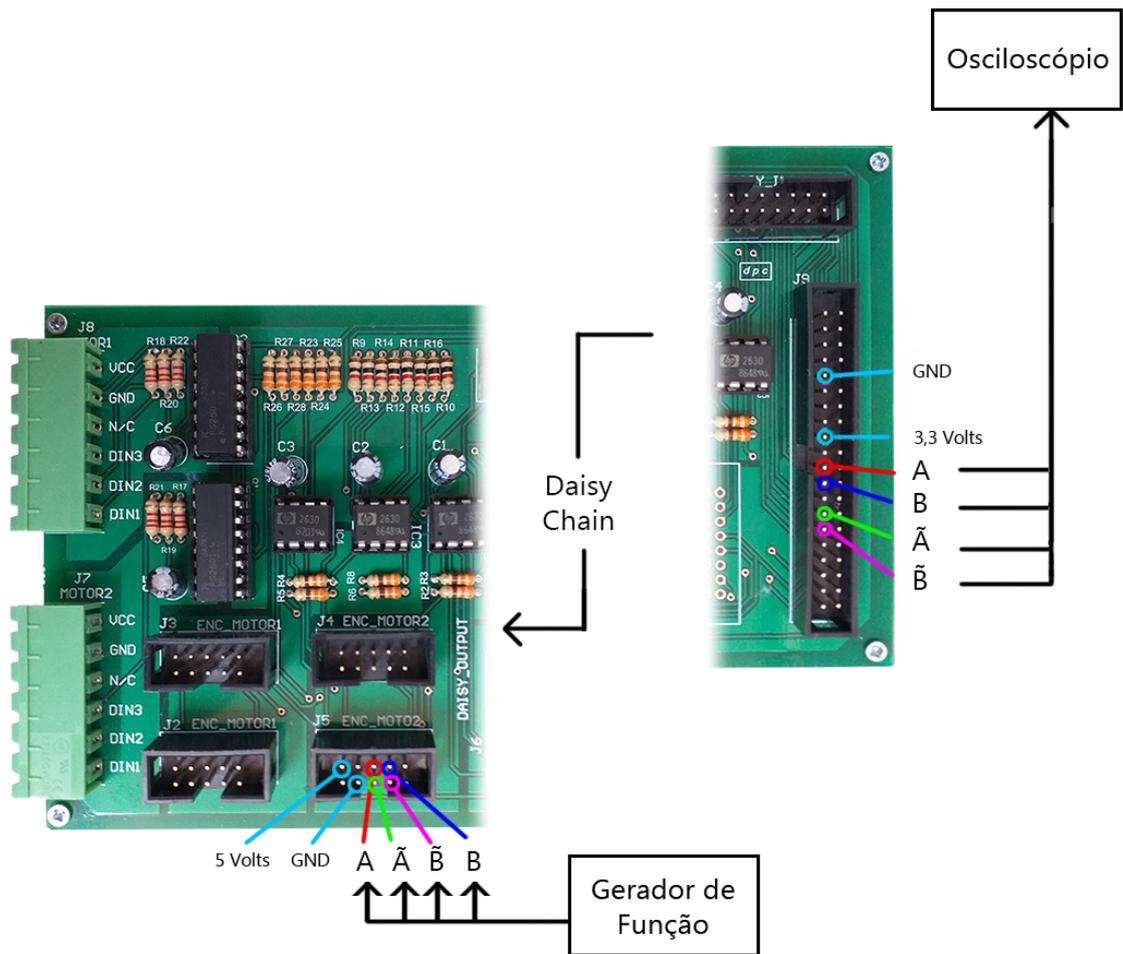


Figura 48 – Montagem do teste para o encoder do motor 4.

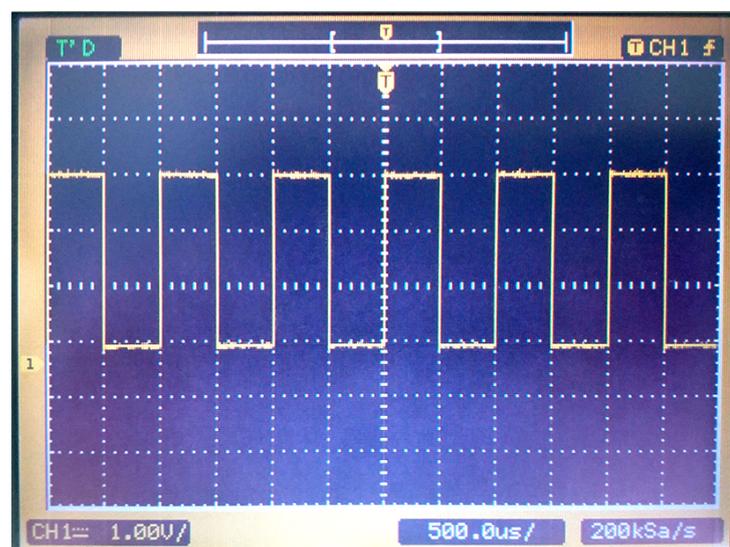


Figura 49 – Entrada do encoder para a Pynq Z2 a uma forma de onda de 1kHz.

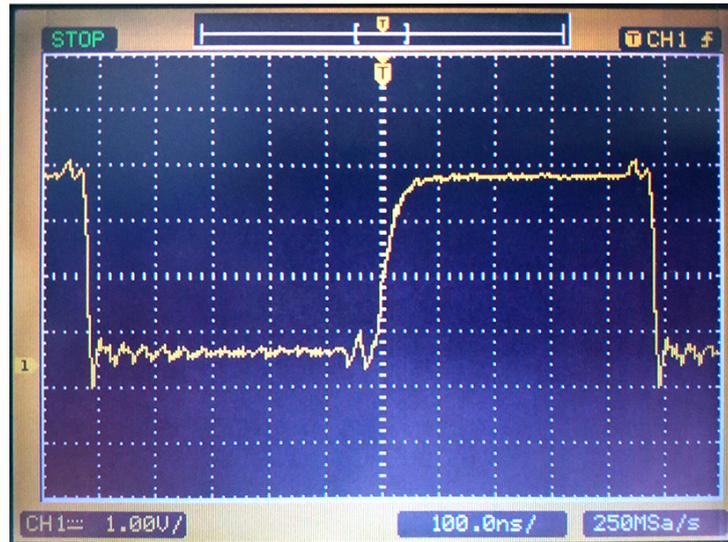


Figura 50 – Entrada do encoder para a Pynq Z2 a uma forma de onda de 1MHz.

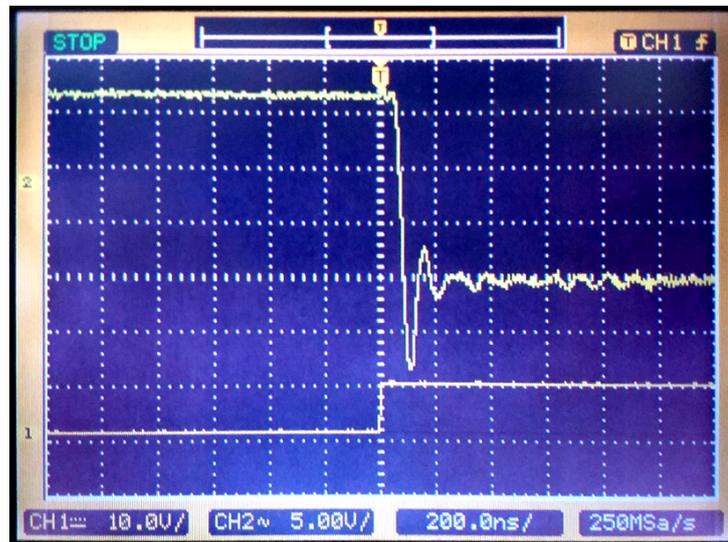


Figura 51 – Atraso do sinal do encoder para a Pynq Z2.

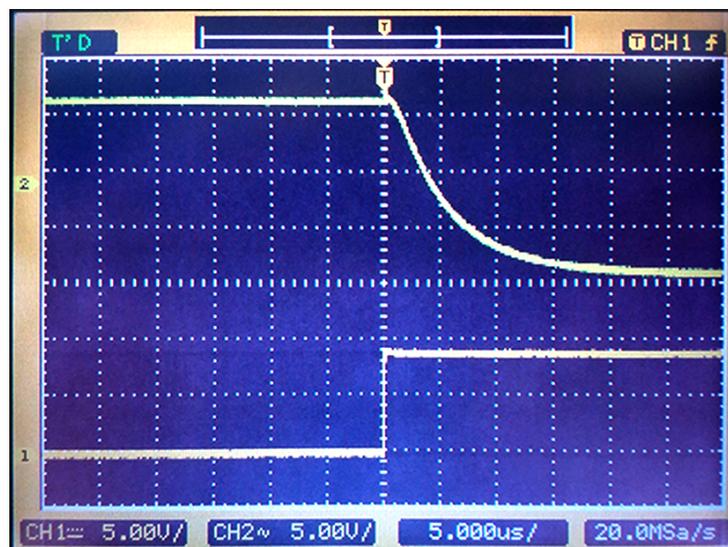


Figura 52 – Atraso do sinal da Pynq Z2 para o driver do motor.

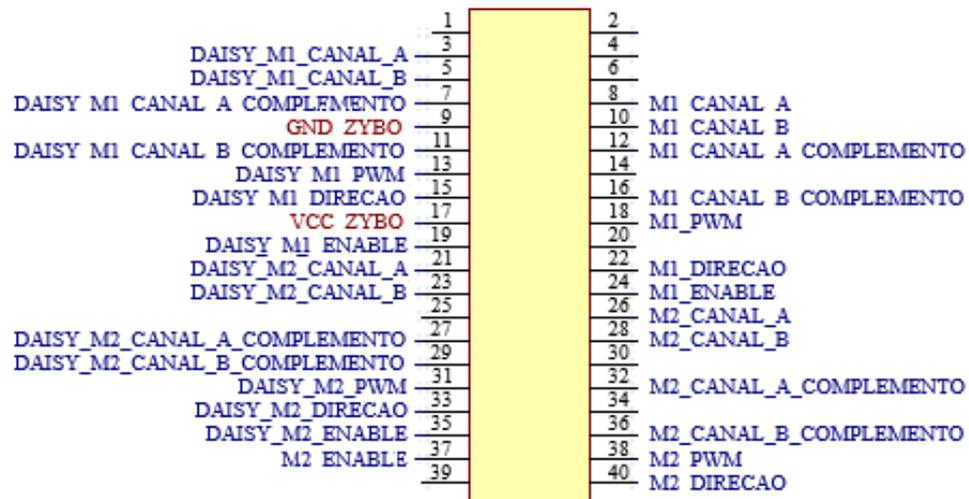


Figura 53 – Conector da placa de interface.

7 Conclusão

Existe uma demanda crescente com o desenvolvimento da tecnologia em exoesqueletos nas mais diversas áreas em que eles se fazem úteis. Com isso em mente e com o foco na área da fisioterapia a UnB está desenvolvendo no GRACO o projeto de um exoesqueleto para os membros inferiores destinado a indivíduos hemiplégicos. Este projeto não só se mostra atrativo para a indústria como também apresenta grande importância social, pois o desenvolvimento de tecnologias como essa pode propiciar uma qualidade de vida aumentada para indivíduos deficientes, além de ter sido mostrada útil na recuperação por meio de fisioterapia assistida.

Este documento relata o procedimento do desenvolvimento e instrumentação de uma bancada que simula o sistema, de modo que se possa com ela desenvolver o sistema de controle do dispositivo. Para isso, foi realizada a revisão bibliográfica, estudado os dispositivos que compõe o sistema, revitalizado a bancada e desenvolvido o projeto de uma placa de interface do sistema para acionamento dos motores e leitura de encoders que o exoesqueleto comportará. O projeto da placa envolveu certo retrabalho à medida que as demandas e os requisitos ficavam mais claros. Isso faz parte da metodologia de projeto e mostra a importância do entendimento de tal ciência tanto na área acadêmica quanto na indústria.

O desenvolvimento da interface consistiu no estudo do funcionamento da interface anterior, verificar e estabelecer os requisitos de projeto seguido da implementação do esquemático e projeto do design em software. Após isso se realizou a fabricação da placa, e após a compra e montagem dos componentes deu-se a fase de testes e validação, que consistiu em verificar o funcionamento correto das trilhas bem como avaliar o desempenho dos componentes com uso de um osciloscópio, em que se verificou se o funcionamento se enquadra nos requisitos especificados.

Teve-se que os resultados obtidos correspondem aos exigidos na especificação. Os testes realizados para cada um dos componentes (motores e encoders) mostraram que o funcionamento foi equivalente ao esperado. Além disso o teste de validação mostrou o funcionamento correto no controle do motor e a leitura correta dos encoders nos conectores em que se encaixam a placa controladora Pynq Z2. Nas próximas etapas do projeto espera-se realizar o desenvolvimento do software e o projeto do hardware (em FPGA) para que se possa dar início ao controle das juntas do exoesqueleto.

Fica evidente que este é um projeto de longo prazo dada a ambição e a força tarefa necessária para desenvolvê-lo, os conhecimentos envolvidos no projeto e implementação percorrem diversas áreas da tecnologia e ciência, e sua conclusão pode acarretar em grande

impacto ao proporcionar inclusão e acelerar recuperação de acometidos, bem como nas aplicação das diversas outras áreas que estes dispositivos atuam.

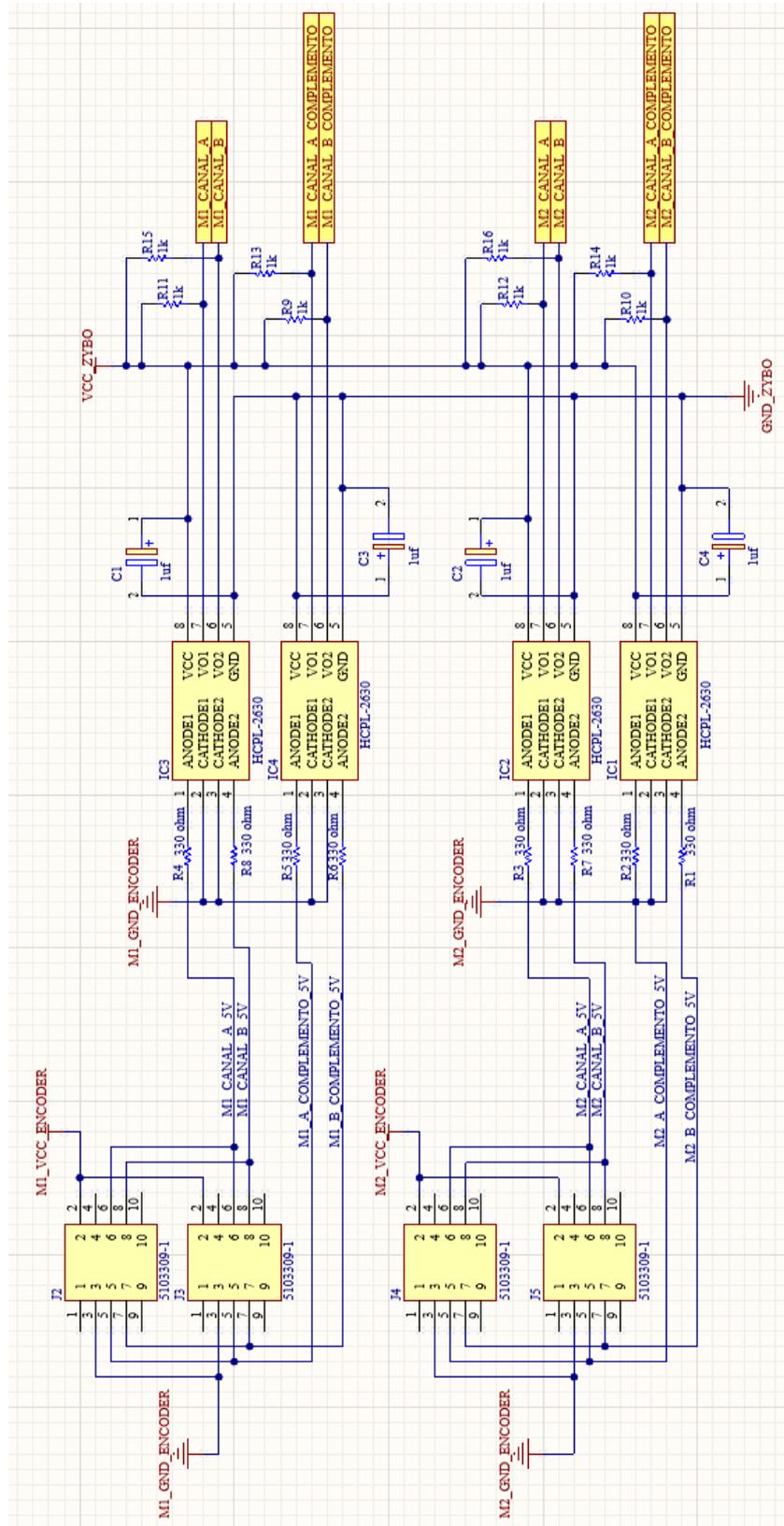
Referências

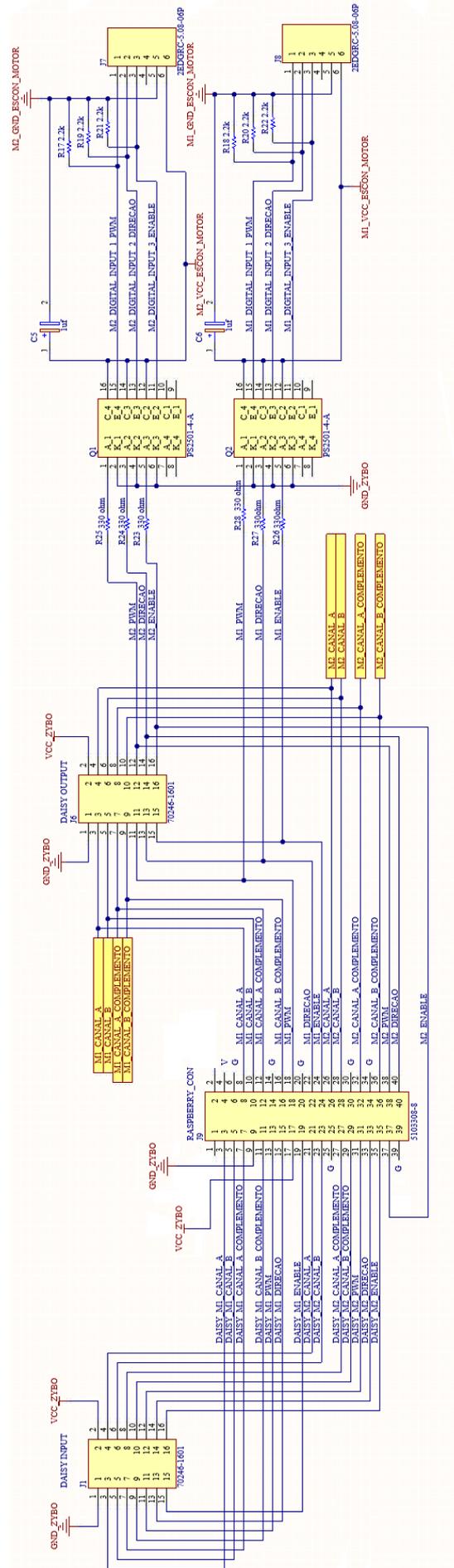
- FREIRE, J. P.; MARAFA, N.; SAMPAIO, R.; SUMIHARA, Y.; BARROS, J.; VIDAL FILHO, W.; LLANOS, C. MECHANICAL DESIGN OF A LOWER LIMB EXOSKELETON FOR REHABILITATION OF PARAPLEGIC PATIENTS. **XXVII Congresso Brasileiro de Engenharia Biomédica**, 2020a. Vitória. Citado nas pp. 14, 15, 21.
- FREIRE, J. P.; MARAFA, N.; SAMPAIO, R.; SUMIHARA, Y.; BARROS, J.; VIDAL FILHO, W.; LLANOS, C. Mechanical Design of an Active Hip and Knee Orthosis for Rehabilitation Applications. **XXVII Congresso Brasileiro de Engenharia Biomédica**, 2020b. Vitória. Citado nas pp. 14, 15, 21, 22.
- IBGE. Censo Demográfico 2000: características gerais da população, 2000. Citado na p. 13.
- IBGE. Censo Demográfico 2010: características gerais da população, 2010. Citado na p. 13.
- MARAFA, N. A.; LLANOS, C. H.; TACO, P. W. G. A Bibliometric Analysis of Lower Limb Exoskeletons for Rehabilitation Applications. **XXVII Congresso Brasileiro de Engenharia Biomédica**, 2020. Vitória. Citado na p. 14.
- MARAFA, N. A.; SAMPAIO, R. C.; LLANOS, C. H. Biomechatronic Analysis of Lower Limb Exoskeletons for Augmentation and Rehabilitation Applications. **XXVII Congresso Brasileiro de Engenharia Biomédica**, 2020. Vitória. Citado na p. 14.
- MEDEIROS GONÇALVES, H. de. **Instrumentação Eletrônica de uma Bengala para Auxiliar no Monitoramento de Marcha de Usuários de Exoesqueleto Inferior**. UnB, 2017. Trabalho de Conclusão de Curso. Citado na p. 14.
- NOVAES, R. D.; MIRANDA, A. S.; DOURADO, V. Z. Velocidade usual da marcha em brasileiros de meia idade e idosos. **Revista Brasileira de Fisioterapia**, 2011. São Carlos. Citado na p. 35.
- PASSOS, J. C. P.; SANTOS, C. E. D. S.; SAMPAIO, R. C.; COELHO, L. D. S. Sizing Optimization of a Exoskeleton Structure Utilizing Finite Element Analysis and Multi-Objective Search. **XXII Congresso Brasileiro de Automática**, 2018. João Pessoa. Citado na p. 14.
- ROBBI, D. B. **ANÁLISE DINÂMICA DE UM EXOESQUELETO DE MEMBROS INFERIORES UTILIZADO NO CONTEXTO DE REABILITAÇÃO DE INDIVÍDUOS COM LESÃO MEDULAR**. 2018. Monografia (Graduação) – Universidade de Brasília, Brasília. Citado nas pp. 35, 36.

- RODRIGUES, A. P. C. **Desenvolvimento de uma interface para acionamento de atuadores e leitura de encoders para um exoesqueleto de membro inferior com a plataforma SoC-FPGA Zybo**. UnB, 2017. Trabalho de Conclusão de Curso. Citado nas pp. 14, 16, 28, 34, 35.
- TEJERA, J. A. de la; BUSTAMANTE-BELLO, R.; RAMIREZ-MENDOZA, R. A.; IZQUIERDO-REYES, J. Systematic Review of Exoskeletons towards a General Categorization Model Proposal. **Applied Sciences**, v. 11, n. 1, 2021. ISSN 2076-3417. DOI: [10.3390/app11010076](https://doi.org/10.3390/app11010076). Disponível em: <<https://www.mdpi.com/2076-3417/11/1/76>>. Citado na p. 12.
- WOLFF, J.; PARKER, C.; BORISOFF, J.; MORTENSON, W. B.; MATTIE, J. A survey of stakeholder perspectives on exoskeleton technology. **Journal of NeuroEngineering and Rehabilitation**, v. 11, 2020. Citado na p. 13.

Apêndices

APÊNDICE A – Esquemático





Anexos

