



Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Engenharia Elétrica



Trabalho de Conclusão de Curso

Desenvolvimento de uma plataforma *Hardware in the loop* para simulação de sistemas em tempo real

Hermes Henrique Silva Melo

João Monlevade, MG
2024

Hermes Henrique Silva Melo

Desenvolvimento de uma plataforma *Hardware in the loop* para simulação de sistemas em tempo real

Trabalho de Conclusão de Curso apresentado à Universidade Federal de Ouro Preto como parte dos requisitos para obtenção do Título de Bacharel em Engenharia Elétrica pelo Instituto de Ciências Exatas e Aplicadas da Universidade Federal de Ouro Preto.

Orientador: Prof. Dr. Marcelo Moreira Tiago

Coorientador: Prof. Dr. Márcio Feliciano Braga

Universidade Federal de Ouro Preto
João Monlevade
2024

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

M528d Melo, Hermes Henrique Silva.

Desenvolvimento de uma plataforma hardware in the loop para simulação de sistemas em tempo real. [manuscrito] / Hermes Henrique Silva Melo. - 2024.

60 f.

Orientador: Prof. Dr. Marcelo Moreira Tiago.

Coorientador: Prof. Dr. Márcio Feliciano Braga.

Monografia (Bacharelado). Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Aplicadas. Graduação em Engenharia Elétrica .

1. Controle de processo. 2. Emuladores (Programas de computador). 3. Microcontroladores. 4. Simulação (Computadores digitais). I. Tiago, Marcelo Moreira. II. Braga, Márcio Feliciano. III. Universidade Federal de Ouro Preto. IV. Título.

CDU 681.5

Bibliotecário(a) Responsável: Flavia Reis - CRB62431



FOLHA DE APROVAÇÃO

Hermes Henrique Silva Melo

Desenvolvimento de uma plataforma *Hardware in the Loop* para simulação de sistemas em tempo real

Monografia apresentada ao Curso de Engenharia Elétrica da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de bacharel em Engenharia Elétrica

Aprovada em 7 de fevereiro de 2024.

Membros da banca

Dr. Marcelo Moreira Tiago — Orientador — Universidade Federal de Ouro Preto
Dr. Márcio Feliciano Braga — Coorientador — Universidade Federal de Ouro Preto
Dr. Igor Dias Neto de Souza — Universidade Federal de Ouro Preto
Dr. Renan Fernandes Bastos — Universidade Federal de Ouro Preto

Marcelo Moreira Tiago, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 21/02/2024.



Documento assinado eletronicamente por **Marcelo Moreira Tiago, PROFESSOR DE MAGISTERIO SUPERIOR**, em 21/02/2024, às 11:10, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0670989** e o código CRC **6E3E8BFF**.

Agradecimentos

Agradeço unicamente a Deus por, primeiramente, ter me proporcionado nascer em um lar com muito incentivo ao meu crescimento como indivíduo, incentivo este dado pelos meus pais, Hermes da Silva Melo e Darlene da Silva Melo, os quais são bênçãos de Deus na minha vida. Consequente, Ele me concedeu ter amigos, colegas e professores que contribuíram para a minha formação profissional, dos quais cito meu orientador Marcelo Moreira Tiago e coorientador Márcio Feliciano Braga, que com muita paciência e empenho, garantiram-me um excelente desenvolvimento do projeto que aqui é apresentado.

*"Ao único Deus sábio, Salvador nosso, seja a glória e a majestade, domínio e poder,
agora e para todo sempre, amém."*

– Judas 1:25

Resumo

Este trabalho descreve as etapas de projeto e implementação de um sistema *Hardware In the Loop* (HIL) de baixo custo, desenvolvido para emular o comportamento de sistemas utilizados em aplicações industriais. Ao longo do trabalho, utilizou-se um kit de desenvolvimento baseado no microcontrolador ESP32. Inicialmente, foram realizados ensaios para caracterizar a plataforma de hardware escolhida. Ao longo desses ensaios, foram analisadas as frequências máximas de amostragem e reconstrução de sinais analógicos, a resolução dos conversores analógico-digitais e digitais-analógicos, além da capacidade de processamento do microcontrolador escolhido. O controle dos períodos de amostragem e reconstrução foi realizado por um *timer*, responsável por gerar interrupções periódicas. Os resultados obtidos mostram que o sistema é capaz de operar a uma frequência máxima de 10 kHz, com uma resolução vertical de 12 bits para conversão AD, e 8 bits para a conversão DA. Na sequência, foram realizados ensaios comparativos entre sistemas físicos reais e sistemas emulados pela plataforma HIL proposta neste trabalho. Para isso, inicialmente utilizou-se o modelo de um tanque industrial, obtido a partir da identificação de um sistema presente na planta didática SMAR PD3-F. Finalmente, foram realizados ensaios utilizando o modelo de um filtro RC passa-baixas, comparando-se os sinais emulados pela plataforma aos medidos num sistema real, por meio de um osciloscópio digital. Os resultados mostram que a plataforma HIL foi capaz de emular, de forma adequada, o comportamento do sistema físico para os casos em que a relação entre a frequência de amostragem e a frequência do sinal de entrada foi superior a 10 vezes, e que a partir desse limite, o sistema passa a apresentar distorções que inviabilizam a utilização da plataforma.

Palavras-chave: *Hardware in the loop*, Simulação em Tempo Real, Microcontroladores, Processamento Digital de Sinais, Controle de Processos Industriais.

Abstract

This work describes the stages of design and implementation of a low-cost Hardware In the Loop (HIL) system, developed to emulate the behavior of systems used in industrial applications. Throughout the work, a development kit based on the ESP32 microcontroller was used. Initially, tests were conducted to characterize the chosen hardware platform. During these tests, the maximum sampling and reconstruction frequencies of analog signals, the resolution of analog-to-digital and digital-to-analog converters, and the processing capacity of the chosen microcontroller were analyzed. The control of sampling and reconstruction periods was carried out by a timer, responsible for generating periodic interruptions. The results obtained show that the system is capable of operating at a maximum frequency of 10 kHz, with a vertical resolution of 12 bits for AD conversion, and 8 bits for DA conversion. Subsequently, comparative tests were conducted between real physical systems and systems emulated by the HIL platform proposed in this work. For this purpose, the model of an industrial tank was initially used, obtained from the identification of a system present in the SMAR PD3-F didactic plant. Finally, tests were conducted using the model of a low-pass RC filter, comparing the signals emulated by the platform to those measured in a real system using a digital oscilloscope. The results show that the HIL platform was able to adequately emulate the behavior of the physical system for cases where the relationship between the sampling frequency and the input signal frequency was greater than 10 times, and beyond this limit, the system begins to present distortions that make the use of the platform unfeasible.

Keywords: Hardware in the loop, Real Time Simulation, Microcontrollers, Digital Signal Processing, Industrial Process Control.

Lista de ilustrações

Figura 1 – Ilustração básica do funcionamento de um sistema HIL comunicando-se com um controlador em malha fechada.	2
Figura 2 – Diagrama de blocos simplificado de uma plataforma de processamento digital de sinais.	8
Figura 3 – Esquema simplificado que relaciona a entrada e a saída do ADC.	9
Figura 4 – Esquema simplificado que relaciona a entrada e a saída do DAC.	9
Figura 5 – Kit de desenvolvimento ESP32-WROOM32 NodeMCU.	11
Figura 6 – Esquema de transferência de dados por DMA.	12
Figura 7 – Esquema da transferência de dados por DMA utilizado pelo microcontrolador ESP32-WROOM32.	13
Figura 8 – Procedimento para determinação de modelo matemático de um sistema.	14
Figura 9 – Caracterização do módulo ADC1 presente no microcontrolador, com o objetivo de validar a utilização da biblioteca de calibração disponível.	18
Figura 10 – Configuração utilizada para determinação da máxima frequência de amostragem considerando o método varredura.	19
Figura 11 – Saídas dos processos de amostragem (antes e após a calibração do ADC1) e reconstrução dos sinais.	20
Figura 12 – Bancada utilizada ao longo da realização dos ensaios de caracterização do sistema.	21
Figura 13 – Caracterização do sistema de amostragem e reconstrução considerando sinal de entrada senoidal, 1 kHz, sem ruído.	22
Figura 14 – Caracterização do sistema de amostragem e reconstrução considerando sinal de entrada senoidal, 1 kHz, com 25% de ruído.	23
Figura 15 – Caracterização do sistema de amostragem e reconstrução considerando sinal de entrada senoidal, 2 kHz, com 25% de ruído.	23
Figura 16 – Sinal interno obtido após amostragem de sinal de entrada senoidal, 3 kHz, com 25% de ruído.	24
Figura 17 – Sinais senoidais, com frequência de 3 kHz, reconstruídos.	24
Figura 18 – Sinais de entrada e saída utilizados nos ensaios realizados com o filtro de média-móvel.	26
Figura 19 – Análise espectral do sinal de saída obtido no ensaio realizado com o filtro de média-móvel.	26
Figura 20 – Esquema genérico de um Filtro RC passa-baixas.	27
Figura 21 – Filtro RC Analógico montado na bancada de ensaios.	27
Figura 22 – Bancada utilizada ao longo da realização dos ensaios do sistema.	28

Figura 23 – Comparação entre sinais medidos por Lage (2019) em seu trabalho, e os sinais obtidos a partir dos modelos discretos considerando dois pontos de operação.	31
Figura 24 – Respostas do sistema HIL e do filtro analógico, para entrada senoidal com frequência de 100 Hz.	34
Figura 25 – Respostas do sistema HIL e do filtro analógico, para entrada senoidal com frequência de 400 Hz.	34
Figura 26 – Respostas do sistema HIL e do filtro analógico, para entrada senoidal com frequência de 700 Hz.	34
Figura 27 – Respostas do HIL e do filtro analógico, para entrada senoidal com frequência de 1 kHz.	35
Figura 28 – Respostas do sistema HIL e do filtro analógico, para entrada em formato de onda quadrada com $2 V_{pp}$, <i>offset</i> de 1,5 V e frequência de 50 Hz.	36
Figura 29 – Respostas do sistema HIL e do filtro analógico, para entrada em formato de onda triangular com $2 V_{pp}$, <i>offset</i> de 1,5 V e frequência de 50 Hz.	36
Figura 30 – Respostas do sistema HIL e do filtro analógico, para entrada em formato de onda dente de serra com $2 V_{pp}$, <i>offset</i> de 1,5 V e frequência de 50 Hzz.	36
Figura 31 – Comparação entre os sinais obtidos pelas equações no domínio z utilizando o Matlab, e os sinais obtidos pela plataforma HIL embarcada com as equações de diferenças, para entrada em degrau.	37
Figura 32 – Comparação entre os sinais obtidos por Lage (2019), e os sinais obtidos pela plataforma HIL, para entrada em degrau.	37
Figura 33 – Comparação entre os sinais obtidos por Lage (2019), e os sinais obtidos pela plataforma HIL, para uma entrada proporcional ao sinal de entrada utilizado por Lage (2019).	38
Figura 34 – Fluxograma da conversão de dados no sistema.	39
Figura 35 – Comparação das respostas do sistema HIL, com as medidas na planta real, utilizando os mesmos sinais de controle de abertura da válvula pneumática medidos por Lage (2019) no sistema real.	40
Figura 36 – Sinal de entrada senoidal e resposta do HIL utilizando DMA por meio da comunicação I2S.	41

Lista de tabelas

Tabela 1	–	Comparação entre os períodos de amostragem dos processos de amostragem (antes e após a calibração do ADC1) e reconstrução dos sinais sem nenhum processamento adicional.	20
Tabela 2	–	RMSE percentual calculado para os dois pontos de operação a partir da diferença entre: valores medidos por Lage (2019) aos valores obtidos pelo HIL antes da reconstrução pelo DAC; e os valores simulados pelo Matlab utilizando o modelo identificado, aos valores obtidos pelo HIL antes da reconstrução pelo DAC.	38
Tabela 3	–	RMSE percentual calculado para os dois pontos de operação a partir das diferenças entre: valores medidos por Lage (2019) aos valores obtidos pelo HIL após a reconstrução pelo DAC; valores simulados pelo Matlab utilizando o modelo identificado, aos valores obtidos pelo HIL após a reconstrução pelo DAC; e os valores internos do microcontrolador antes da reconstrução pelo DAC, aos valores obtidos pelo HIL após a reconstrução pelo DAC.	39

Lista de Siglas

HIL	Hardware in The Loop
ADC	Conversor Analógico Digital (do inglês <i>Analog Digital Converter</i>)
DAC	Digital Analógico (do inglês <i>Digital Analog Converter</i>)
DMA	Acesso Direto à memória (do inglês <i>Direct Memory Access</i>)
CC	Corrente Contínua
IDE	Ambiente de Desenvolvimento Integrado (do inglês <i>Integrated Development Environment</i>)
FFT	Transformada Rápida de Fourier (do inglês <i>Fast Fourier Transform</i>)
RMSE	Raiz do Erro Médio Quadrático (do inglês <i>Root Mean Square Error</i>)
I2S	<i>Inter-IC Sound</i>

Sumário

1	INTRODUÇÃO	1
1.1	Objetivo	3
1.2	Revisão Bibliográfica	4
1.3	Estrutura do Trabalho	5
2	REVISÃO TEÓRICA	7
2.1	Amostragem e reconstrução de sinais	7
2.2	Processamento digital de sinais em tempo real	9
2.3	Modelagem de Sistemas	14
2.4	Considerações Parciais	16
3	METODOLOGIA	17
3.1	Caracterização do ADC	17
3.2	Caracterização do sistema de amostragem e reconstrução	18
3.3	Análise do sistema de amostragem e reconstrução dos sinais	21
3.4	Avaliação do desempenho do sistema em operações de processamento em tempo real	25
3.4.1	Filtro média-móvel	25
3.4.2	Filtro RC Passa-Baixas	26
3.4.3	Modelo de um tanque industrial	29
3.5	Controle dos processos de aquisição e reconstrução por DMA	31
3.6	Considerações Parciais	32
4	RESULTADOS E DISCUSSÕES	33
4.1	Emulação de Filtro Passivo RC Passa-Baixas	33
4.2	Emulação de um tanque industrial	37
4.3	Utilização do DMA no ESP32	40
4.4	Considerações Parciais	41
5	CONSIDERAÇÕES FINAIS	43
5.1	Sugestões Para Trabalhos Futuros	44
	REFERÊNCIAS	45

1 Introdução

Sistemas controlados vêm sendo utilizados cada vez com maior frequência nos dias atuais, especialmente para o desenvolvimento de produtos destinado à automação de processos industriais. A redução do custo de processadores e a popularização de novas tecnologias têm permitido que os controladores também possam ser utilizados em aplicações residenciais, em tarefas que vão desde a automação de lavadoras até o gerenciamento de casas inteligentes.

Para avaliar a robustez e confiabilidade de um controlador, é necessário realizar um conjunto de testes pré-estabelecidos pelos órgãos governamentais responsáveis pelo controle de qualidade dos equipamentos. Essas normas visam garantir não só a qualidade dos produtos, mas também a segurança dos usuários que os utilizam.

Um dos desafios da indústria nesse campo de atuação é que nem sempre é simples realizar testes capazes de identificar possíveis falhas no projeto dos controladores. Esses desafios envolvem, por exemplo, a complexidade para construção dos protótipos que deverão ser controlados, as restrições econômicas para a construção de plantas de teste, ou até mesmo, os altos riscos envolvidos quando testes são realizados em sistema físicos.

Esses fatores motivaram o desenvolvimento de sistemas capazes de reproduzir, por meio de simulações executadas em tempo real, o comportamento de processos que seriam executados por plantas complexas. Esses algoritmos, apesar de complexos, devem ser executados em tempo real, usando plataformas microprocessadas com capacidade de processamento e tempo de resposta suficientemente altas para garantir uma reprodução fidedigna dos processos que estão sendo simulados.

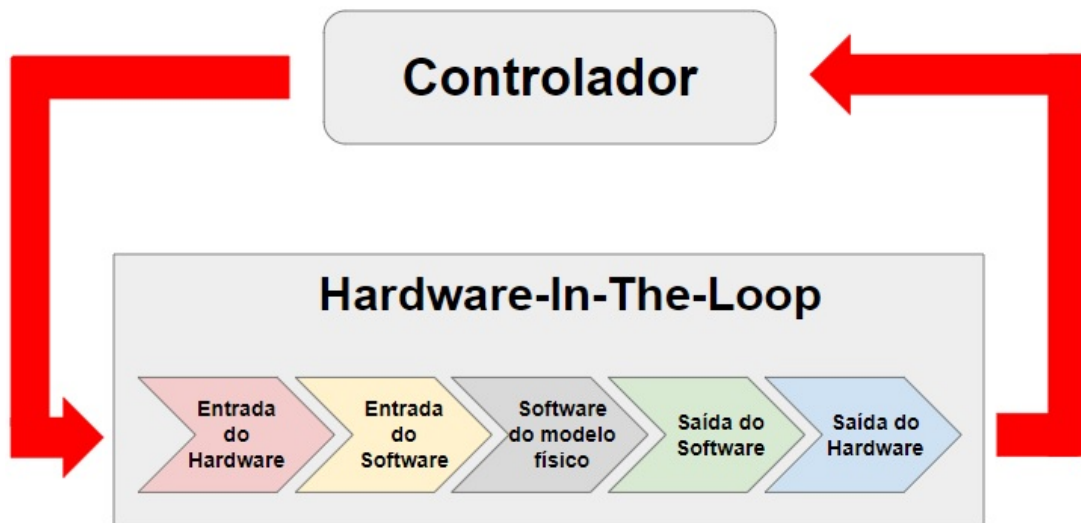
Esse cenário motivou o desenvolvimento dos sistemas HIL (do inglês, *Hardware in the Loop*), compostos por sistemas eletrônicos embarcados capazes de processar sinais em tempo real, executando algoritmos capazes de simular os processos descritos anteriormente.

Como vantagens desses sistemas, pode-se destacar a redução dos tempos para implementação, com possibilidades de readaptações em virtude de novas demandas, além da redução dos custos de produção. Como desafios, destacam-se a complexidade dos modelos matemáticos que deverão ser embarcados no sistema, e as restrições associadas ao tempo de execução dos algoritmos.

A utilização desses sistemas tornou-se uma peça chave na indústria e na pesquisa, principalmente no setor automotivo e nos estudos que envolvem eletrônica de potência. Produtos que dependem de uma unidade controladora podem ser testados em tempo real, sem a construção de um protótipo físico, graças ao sistema HIL.

A Figura 1 apresenta um diagrama de blocos para um sistema HIL, considerando entradas e saídas analógicas e um bloco de processamento digital de sinais em tempo real.

Figura 1 – Ilustração básica do funcionamento de um sistema HIL comunicando-se com um controlador em malha fechada.



Fonte: Do autor.

Inicialmente, os sinais analógicos aplicados na entrada do sistema são adquiridos por um conversor analógico-digital (ADC, do inglês *Analog to Digital Converter*). Os dados adquiridos são processados por um microprocessador, que executa um algoritmo que representa o modelo matemático de um sistema físico que se deseja reproduzir. Finalmente, um conversor digital-analógico (DAC, do inglês *Digital to Analog Converter*) é utilizado para converter os sinais digitais da saída do modelo simulado em sinais de tensão analógicos, representando dessa forma a dinâmica do modelo embarcado.

Pode-se justificar o uso de sistemas HIL para simulação de sistemas controlados devido à redução do custo de projeto, a diminuição dos tempos de implementação e ao aumento da segurança ao longo da realização dos ensaios em casos em que o sistema a ser analisado apresenta riscos potenciais.

Gomez (2001) destaca as vantagens econômicas que o sistema HIL oferecem em projetos de veículos aéreos não tripulados (UAV, do inglês *Unmanned Aerial Vehicle*). O autor apresenta os custos de um projeto HIL (aproximadamente US\$ 25000) e os compara com o custo de projeto de uma unidade UAV, avaliado em milhões de dólares. Dessa forma, caso fosse possível impedir a queda de apenas um UAV a partir de resultados obtidos das simulações com o HIL, a economia gerada para a empresa superaria os custos envolvidos no processo de desenvolvimento da plataforma de simulação.

Na indústria aeroespacial, além dos custos do projeto, há um custo referente a espera para a construção dos sistemas mecânicos utilizados para os testes dos controladores. Além disso, testes mal-sucedidos podem gerar grande prejuízo devido ao elevado investimento necessário para a construção desses equipamentos.

Sistemas HIL também são utilizados com frequência para testes na indústria aeronáutica, quando se deseja reproduzir determinadas condições de voo que podem ser extremamente perigosas, planejadas para que os pilotos sejam treinados para operar os equipamentos em condições extremas.

Como mencionado anteriormente, a implementação de um sistema HIL depende de uma plataforma de processamento capaz de executar algoritmos complexos em tempo real. A escolha da plataforma de processamento deve levar em consideração as características do processo a ser executado e os custos envolvidos ao longo da etapa de implementação.

Neste trabalho, apresentam-se as etapas de desenvolvimento de uma plataforma HIL aplicada a simulação do comportamento de tanques industriais e de filtros passivos. Utilizou-se como base um microcontrolador de 32 bits, operando com um processador dual-core com frequência de 240 MHz. O dispositivo escolhido é capaz de executar cálculos utilizando números representados por meio de ponto flutuante, e possui instruções específicas para executar operações relacionadas a processamento digital de sinais.

Por se tratar de um microcontrolador, há limitações envolvendo a frequência de amostragem e reconstrução de sinais analógicos, a capacidade de armazenamento das memórias volátil e não volátil, além dos tempos de acesso às memórias e periféricos. Apesar das limitações mencionadas, o *hardware* escolhido apresenta características suficientes para reproduzir o sistema proposto neste trabalho.

1.1 Objetivo

O objetivo geral deste trabalho é desenvolver o projeto de um sistema HIL de baixo custo, capaz de simular o modelo de uma planta industrial controlada em malha aberta e em tempo real. O sistema proposto deverá ser capaz de adquirir os sinais de tensão de entrada, processar os sinais adquiridos em tempo real e converter os sinais digitais obtidos após a etapa de processamento em sinais analógicos, garantindo uma frequência de reconstrução equivalente à frequência de amostragem utilizada.

Os objetivos específicos do trabalho são:

- Estudar e compreender as características da simulação em tempo real;
- Caracterizar a plataforma de processamento, definindo limites para frequências de amostragem e reconstrução e janela de tempo para processamento dos sinais;
- Obter o modelo matemático discreto de um tanque utilizado em aplicações industriais;
- Obter o modelo matemático discreto de um filtro RC passa-baixas;

- Implementar no microcontrolador os modelos matemáticos obtidos e realizar testes de validação em tempo real;
- Comparar os resultados obtidos a partir do sistema HIL aos obtidos pelos sistemas originais.

1.2 Revisão Bibliográfica

Laplante e Ovaska (2012) descrevem um sistema como o mapeamento de um conjunto de entradas em um conjunto de saídas. Conforme os autores, sistemas representam uma interconexão de componentes, dispositivos ou subsistemas, em que se estabelece uma relação de causa e efeito. Além disso, um sistema pode ser visto como um processo em que os sinais de entrada são transformados ou induzidos a responder de alguma forma, de modo a gerar os sinais de saída. Oppenheim e Willsky (2010) caracterizam os sistemas com base em suas propriedades de memória, inversão, causalidade, estabilidade, invariância no tempo e linearidade.

Para emular um sistema dinâmico, é necessário utilizar o modelo matemático da resposta dinâmica do sistema em questão. Em geral, é desejável que o modelo desenvolvido seja simples, fidedigno e robusto o suficiente para que atenda a um conjunto de requisitos de desempenho pré-definidos (FRANKLIN; POWELL; EMAMI-NAEINI, 2013).

De acordo com Laplante e Ovaska (2012), uma simulação em tempo real deve satisfazer as restrições de tempo de resposta do sistema analisado. Como exemplo, os autores abordam um problema relacionado a comunicação entre um sensor do tipo acelerômetro e uma aeronave, em que uma sequência de pulsos enviados pelo acelerômetro é utilizada pela aeronave para a determinação de sua posição instantânea. Os autores destacam que o módulo HIL deve ser capaz de processar os dados do acelerômetro em um intervalo de tempo pré-definido, que pode variar em função das especificações da aeronave.

Kopetz (1997) destaca que numa simulação em tempo real, a acurácia dos resultados não depende apenas dos cálculos realizados pela plataforma de processamento de sinais, mas também do momento em que os sinais de entrada aplicados ao sistema HIL foram adquiridos. Um sistema em tempo real deve reagir a estímulos dentro de intervalos de tempo pré-definidos. Falhas críticas podem ocorrer caso os sinais adquiridos não sejam processados dentro do período de amostragem pré-definido. Como exemplo, o autor apresenta o caso de uma ferrovia cruzando uma estrada, indicando que uma falha crítica poderia ocorrer, caso o estado do semáforo utilizado na estrada não fosse alterado a tempo de evitar a passagem do veículo.

A metodologia HIL depende da simulação em tempo real para o modelo do sistema embarcado. Por meio do simulador HIL, o sistema embarcado é forçado para operar em tempo real com entradas e saídas reais (MIHALIČ; TRUNTIČ; HREN, 2022).

De acordo com Gomez (2001), a metodologia HIL visa possibilitar a realização de testes em *hardware* sem a necessidade da prototipagem do sistema físico. O equipamento desenvolvido deve ser capaz de se comunicar com um sistema externo como, por exemplo, um controlador, operando com entradas e saídas físicas em tempo real. Isso ocorre, pois os componentes físicos do sistema emulado são substituídos por modelos matemáticos, de modo que o dispositivo externo não difere sua operação se conectado ao HIL ou ao sistema real.

Isermann et al. (1999) ressalta a importância da utilização de dispositivos HIL em testes de simulação de voo em tempo real, destacando que esta área foi uma das precursoras no uso desse tipo de recurso.

Hanselmann (1993) apresenta o projeto de um sistema HIL utilizado para simular uma unidade eletrônica de controle veicular (ECU, do inglês *Electronic Control Unit*), utilizada para reproduzir o comportamento do sistema de freios antitravamento (ABS, do inglês *Anti lock Braking System*). O autor destaca como vantagem a possibilidade de não se utilizar um veículo para a realização de determinadas fases de testes do sistema de controle, reduzindo riscos, custos de desenvolvimento e tempos de testes.

No cenário atual, algumas estratégias vêm sendo desenvolvidas para reduzir o custo final de implementações do HIL. Bernardes et al. (2019), Bastos et al. (2019), Bastos et al. (2020) apresentam em seus trabalhos o desenvolvimento de sistemas eletrônicos de baixo custo para aplicações diversas, tais como emulação de conversores CC/CC e microrredes, máquinas elétricas, além de sistemas elétricos para acionamentos, por exemplo.

Neste trabalho, foi desenvolvido uma plataforma *Hardware in the loop* para simulação de modelos de sistemas em tempo real, utilizando como contexto o desenvolvimento de um sistema de baixo custo, que opere em tempo real, e que possibilite a emulação de um processo real.

1.3 Estrutura do Trabalho

Este trabalho foi dividido em 5 capítulos.

No Capítulo 1, apresenta-se a contextualização do projeto, com os objetivos e o referencial bibliográfico.

No Capítulo 2, descrevem-se os fundamentos teóricos para a realização deste trabalho.

No Capítulo 3, explica-se o procedimento realizado para a coleta dos resultados, assim como para a validação e caracterização necessárias.

No Capítulo 4, mostram-se os resultados obtidos neste projeto bem como as discussões acerca deles.

Por fim, no Capítulo 5, apresentam-se as considerações finais do projeto e sugestões para trabalhos futuros.

2 Revisão Teórica

Este Capítulo apresenta os conceitos teóricos aplicados ao longo do desenvolvimento deste trabalho. A seção inicial é destinada a uma revisão dos pontos principais referentes ao teorema da amostragem e reconstrução de sinais. São descritos os efeitos causados pelo processo de quantização dos sinais, pelo *aliasing*, quando a frequência de amostragem não é ajustada corretamente, e as principais características dos conversores analógico-digitais de aproximações sucessivas. Além disso, apresentam-se as principais características associadas ao processo de reconstrução de sinais por meio de conversores digital-analógicos, destacando-se os efeitos do ruído de quantização associado ao processo de reconstrução dos sinais e a importância da utilização dos filtros analógicos de reconstrução.

Na sequência, apresenta-se um conjunto de características necessárias para garantir que um algoritmo seja executado em tempo real. Nesse sentido, descreve-se um conjunto de características associadas à plataforma de processamento que devem ser consideradas durante a escolha do *hardware* a ser utilizado em aplicações com essas características.

Finalmente, são apresentadas técnicas para a modelagem de sistemas a partir de expressões com representação discreta no tempo, necessárias para descrever o comportamento dos sistemas físicos analisados a partir de algoritmos executados em tempo real.

2.1 Amostragem e reconstrução de sinais

Os recentes avanços e redução de custos de sistemas microprocessados têm contribuído para que projetos de sistemas de controle, antes desenvolvidos de forma analógica, passassem a ser implementados de forma digital. Por se tratar de uma plataforma digital, é necessário que os sinais analógicos associados ao processo, tais como os provenientes de sensores e transdutores, sejam amostrados por conversores analógico-digitais antes do início do processamento digital de sinais.

O processo de conversão de um sinal analógico em seu equivalente digital pode ser dividida em duas etapas: amostragem e quantização. No processo de amostragem, o sinal analógico de entrada é representado de forma discreta no domínio do tempo. Esse processo deve garantir que a frequência de amostragem do ADC seja, no mínimo, duas vezes maior do que a máxima frequência do sinal aplicado em sua entrada. Boas práticas recomendam a utilização de cinco a dez amostras por período, com o objetivo de se garantir uma boa representação do sinal amostrado no domínio no tempo.

O processo de quantização define uma representação discreta do sinal em relação a sua amplitude de entrada. Esse processo depende do número de bits utilizado pelo ADC e pelo valor das referências positiva e negativa de tensão utilizadas pelo conversor. Em

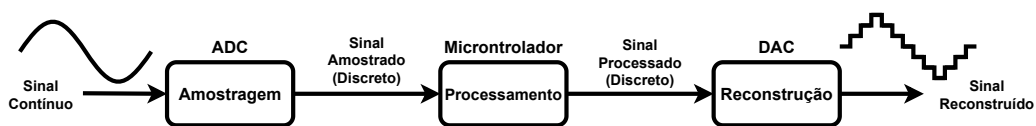
geral, costuma-se encontrar ADC com resolução vertical variando entre 8 bits e 24 bits, mas destaca-se que o aumento do número de bits costuma estar associado ao aumento do custo do conversor e a redução dos valores de frequência de amostragem do sistema (FLOYD, 2007).

Após o processo de aquisição, o sinal digital adquirido costuma ser utilizado como entrada de um algoritmo de processamento de sinais. Especificamente para aplicações envolvendo controladores digitais, esses sinais podem ser usados como entradas para um controlador, que irá relacioná-las com um conjunto de coeficientes de modo a gerar uma saída de controle. O sinal de saída gerado pelo controlador pode ser utilizado, em algumas situações, de forma digital, ou seja, a saída do controlador pode ser enviada de forma direta a um atuador. Entretanto, em algumas situações, é necessário recuperar um sinal analógico a partir do sinal digital obtido na saída do controlador.

Para casos em que se necessita reconstruir sinais analógicos a partir de sinais digitais, pode-se utilizar conversores digital-analógicos. Assim como os ADCs, os DACs também utilizam uma frequência de reconstrução, que deve ser minimamente igual ao dobro da frequência do sinal que se deseja reconstruir. De modo análogo, boas práticas recomendam a utilização de cinco a dez pontos por período, a fim de se garantir uma boa representação temporal do sinal reconstruído. Os DACs também possuem número finito de bits, e assim como os ADCs, apresentam os efeitos causados pelo processo de quantização (FLOYD, 2007).

A Figura 2 apresenta o diagrama esquemático simplificado de um sistema de aquisição, processamento e reconstrução de sinais.

Figura 2 – Diagrama de blocos simplificado de uma plataforma de processamento digital de sinais.



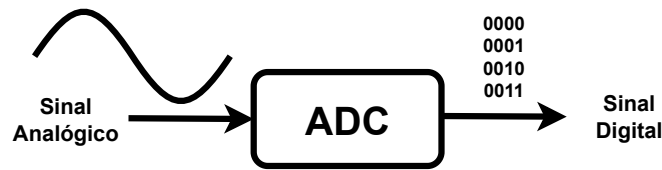
Fonte: Do autor.

Como mencionado, o ADC é um periférico presente em plataformas de processamento de sinais destinado a amostrar o sinal analógico a ser processado. Esse dispositivo possui dois parâmetros importantes que devem ser definidos durante a especificação da plataforma de processamento: a resolução vertical (V) e a frequência de amostragem (Hz). A resolução vertical pode ser calculada a partir da expressão

$$res(V) = \frac{V_{ref(+)} - V_{ref(-)}}{2^n - 1}, \quad (2.1)$$

sendo V_{ref+} e V_{ref-} as referências positiva e negativa de tensão (V), respectivamente, e n o número de bits do conversor. A Figura 3 apresenta um diagrama esquemático de um ADC.

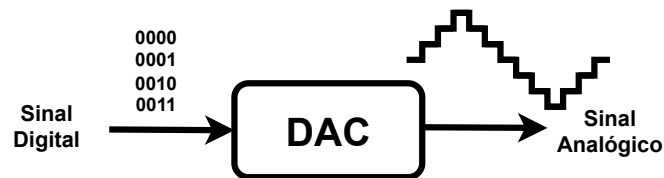
Figura 3 – Esquema simplificado que relaciona a entrada e a saída do ADC.



Fonte: Do autor.

O DAC é o periférico destinado a reconstruir o sinal que foi processado e, assim como para o ADC, a resolução e a frequência de amostragem são dois parâmetros importantes que devem ser considerados durante a etapa de especificação do projeto. A Figura 4 apresenta um diagrama esquemático básico considerando entrada e saída de um DAC.

Figura 4 – Esquema simplificado que relaciona a entrada e a saída do DAC.



Fonte: Do autor.

A resolução e frequência de amostragem de um DAC podem ser calculadas a partir das mesmas expressões definidas para o ADC.

É importante destacar que para simplificar o processo de construção de uma plataforma de processamento digital de sinais, pode-se ajustar os mesmos valores para as frequências de amostragem e reconstrução dos sinais obtidos a partir dos conversores AD e DA. Além disso, pode-se selecionar dispositivos que operem com mesmos valores de tensões de referência e mesmo número de bits. Entretanto, ressalta-se que isso nem sempre é possível, e que mesmo em situações nas quais se opera com dispositivos com características diferentes, pode-se utilizar estratégias de processamento digital de sinais, tais como interpolação e dizimação, para condicionar os sinais de forma adequada.

2.2 Processamento digital de sinais em tempo real

Em aplicações envolvendo processamento digital de sinais em tempo real, é necessário garantir que o algoritmo desenvolvido para executar uma aplicação opere num tempo menor do que o definido pelas frequências de amostragem e reconstrução. Para isso, deve-se minimizar a latência relacionada aos processos de leitura e escrita na memória e periféricos do sistema embarcado, garantindo que o modelo a ser representado apresente tempos de resposta próximos aos do processo que se deseja representar (TEJADA, 2023).

Um dos desafios ao se desenvolver um sistema HIL é justamente encontrar um modelo matemático para representar um determinado processo que seja capaz de ser executado em um sistema embarcado em tempo real. Esse desafio envolve não só as limitações relacionadas ao *hardware*, mas também os efeitos gerados pelas aproximações e simplificações relacionadas ao processo de modelagem e identificação de sistemas complexos.

Devido a essas limitações, em geral, as análises são feitas considerando um intervalo de confiança definido em função do erro entre os tempos de resposta obtidos a partir de sistemas reais em comparação aos obtidos a partir do modelo. Essa análise é de extrema importância para o projeto de sistemas HIL, cujo objetivo é reproduzir o comportamento de sistemas reais a fim de viabilizar análises operacionais.

Vários tipos de plataformas de processamento digital de sinais podem ser utilizadas para embarcar sistemas HIL, sendo escolhidas conforme a complexidade do sistema que se deseja simular. Neste trabalho, optou-se por utilizar uma plataforma de baixo custo, baseada em um microcontrolador com capacidade para realizar operações usando números com representação em ponto flutuante e cálculos usando instruções específicas para processamento digital de sinais.

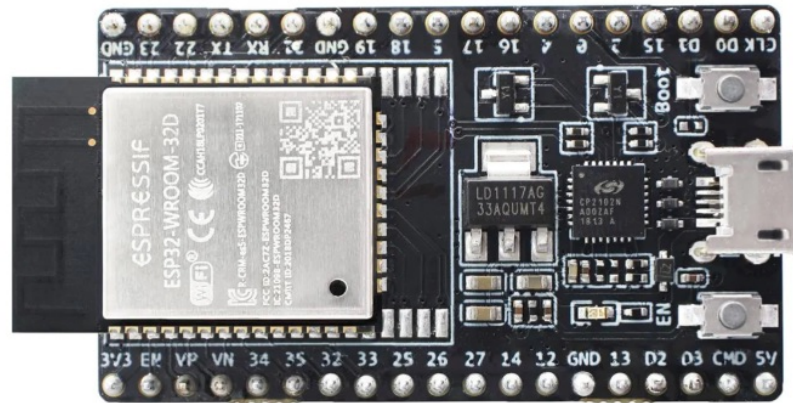
De forma geral, pode-se considerar microcontroladores como circuitos integrados (CIs) compostos por um conjunto de dispositivos embarcados num mesmo encapsulamento. Esses dispositivos possuem memórias volátil em geral, do tipo SRAM (do inglês, *Static Random Access Memory*), não volátil (em geral, do tipo *Flash*), processadores e dispositivos periféricos com características diversas, incluindo conversores AD e DA, por exemplo (PENIDO; TRINDADE, 2013).

Microcontroladores não possuem as mesmas características de processadores embarcados em computadores de uso pessoal, FPGAs (do inglês, *Field-programmable gate arrays*) ou processadores digitais de sinais usados em aplicações de alto desempenho, capazes de processar sinais em tempo reduzido devido a elevada capacidade dos processadores, capazes de operar com frequências na ordem de GHz. Em contrapartida, esses dispositivos apresentam custo consideravelmente menor, podem operar com frequências na ordem de dezenas a centenas de MHz, e são indicados para aplicações envolvendo processos que apresentam tempos de resposta maiores, como o sistema de controle analisado neste trabalho.

Atualmente, pode-se encontrar uma vasta gama de microcontroladores disponível na indústria, proporcionando um ambiente altamente competitivo que favorece uma análise criteriosa para a seleção do dispositivo mais adequado para cada aplicação.

Neste trabalho, optou-se por utilizar uma plataforma de desenvolvimento baseada no microcontrolador ESP32-WROOM32 (ESPRESSIF, 2015-2023). A Figura 5 apresenta uma imagem do dispositivo escolhido.

Figura 5 – Kit de desenvolvimento ESP32-WROOM32 NodeMCU.



Fonte: Retirado de Espressif (2015-2023).

O dispositivo escolhido opera com um processador *dual core* de 240 MHz, possui ADC e DAC integrados, além de diversos periféricos específicos para gerenciamento de tempo e comunicação. Tais características permitem, por exemplo, que algoritmos concorrentes possam ser executados, uma vez que há dois núcleos disponíveis para utilização (ESPRESSIF, 2023).

O processo de gerenciamento do tempo de execução dos algoritmos pode ser feito por meio de temporizadores. Esses dispositivos são responsáveis por gerar uma referência de tempo usada para definir o período de amostragem e reconstrução dos sinais, e essa janela de tempo é utilizada pelo programador como uma referência para limitar os tempos de processamento aceitáveis para cada aplicação (PENIDO; TRINDADE, 2013). Nesse sentido, boas práticas recomendam que o tempo utilizado para executar os algoritmos de processamento de sinais não seja superior a 70 % do tempo definido pelo período de amostragem, garantindo que todo sinal adquirido seja processado antes de uma nova aquisição.

Uma das formas de se sincronizar os processos de amostragem, reconstrução e processamento dos sinais é por meio do uso de interrupções. As interrupções podem ser utilizadas por diferentes tipos de periféricos, mas, neste trabalho, optou-se por utilizar somente as interrupções geradas pelos temporizadores. Interrupções são utilizadas quando se deseja que uma parte da implementação opere com um tempo específico (PENIDO; TRINDADE, 2013).

Uma das formas de se reduzir os tempos envolvidos nos processos de aquisição e reconstrução dos sinais envolve a alteração da maneira como o processador acessa as informações associadas aos periféricos ADC e DAC. Esse acesso poder ser feito de três formas: por varredura, por interrupção ou por acesso direto à memória (DMA, do inglês, *Direct Memory Access*).

No método por varredura, o processador acessa periodicamente os registradores de entrada do ADC e recarrega os registradores de saída do DAC. Esse procedimento é feito

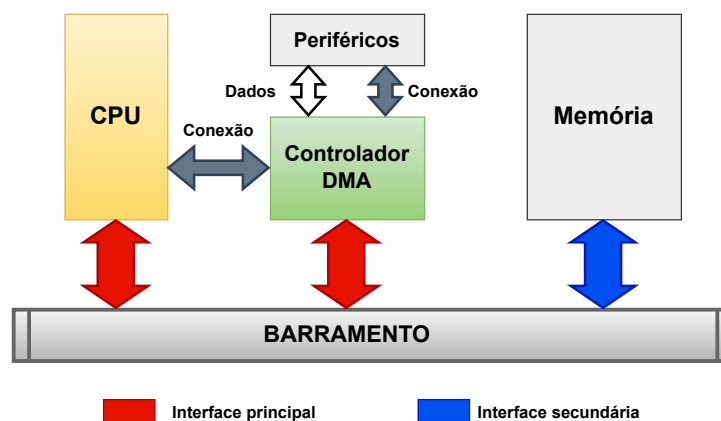
ainda que nenhuma nova informação tenha sido adquirida ou processada, consumindo um tempo de processamento desnecessário.

No método por interrupção, o processador acessa os registradores do ADC e DAC somente se um dos periféricos gera um pedido de interrupção. Esses pedidos de interrupção são gerados em intervalos de tempo definidos conforme a frequência de amostragem dos sinais. Como vantagem, esse método não exige que o processador realize operações de leitura e escrita nos registradores associados aos periféricos sem que haja necessidade. Como desvantagem, o processador deverá interromper suas atividades sempre que uma nova solicitação de interrupção for gerada, utilizando parte de seus recursos para copiar informações de/para registradores de uso específicos associados aos conversores ADC e DAC.

Nos dois casos apresentados anteriormente, a transferência de dados normalmente é feita utilizando a unidade central de processamento (CPU, do inglês *Central Processing Unit*) do microcontrolador, e os processos de leitura, escrita e armazenamento de dados na SRAM consomem recursos da CPU, ocasionando um grande consumo de tempo quando deseja-se transferir uma quantidade significativa de dados (FLOYD, 2007).

Uma forma mais eficiente de se transferir dados entre periféricos e memória é por meio do DMA. Nesse processo, um controlador de DMA, presente no microcontrolador, é programado para receber informações de entrada provenientes do ADC e copiá-las num bloco de memória SRAM pré-definido pelo usuário, sem que seja necessário que o processador participe do processo. De forma análoga, o controlador de DMA pode ser utilizado para realizar operações de escrita nos registradores de uso específico presentes no DAC, copiando informações da memória SRAM para os registradores de uso específico do conversor sem que seja necessária uma intervenção da CPU (FLOYD, 2007). A Figura 6 apresenta um esquema básico de como é realizada a transferência de dados por DMA num dispositivo microcontrolado (FLOYD, 2007).

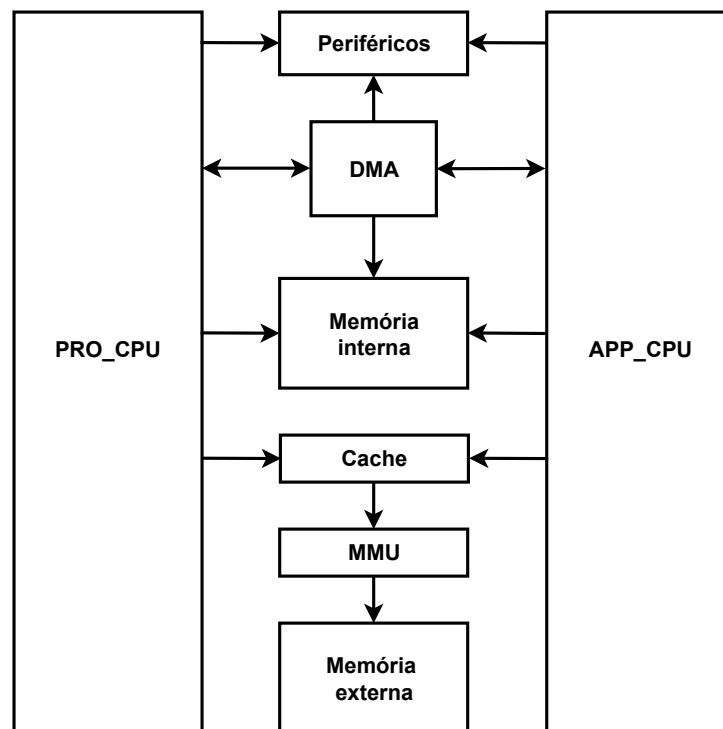
Figura 6 – Esquema de transferência de dados por DMA.



Fonte: Do autor.

A grande vantagem deste método é que o processador irá executar somente o algoritmo de processamento de sinais, reduzindo o tempo gasto para acesso de informações armazenadas nos registradores presentes nos dispositivos periféricos. É importante ressaltar, no entanto, que deve-se garantir que os dados adquiridos e copiados na memória pelo controlador de DMA sejam processados dentro do período de amostragem definido pelo usuário. A Figura 7 apresenta um diagrama esquemático do processo de transferência de dados por DMA utilizado pelo microcontrolador ESP32-WROOM32.

Figura 7 – Esquema da transferência de dados por DMA utilizado pelo microcontrolador ESP32-WROOM32.



Fonte: Do autor.

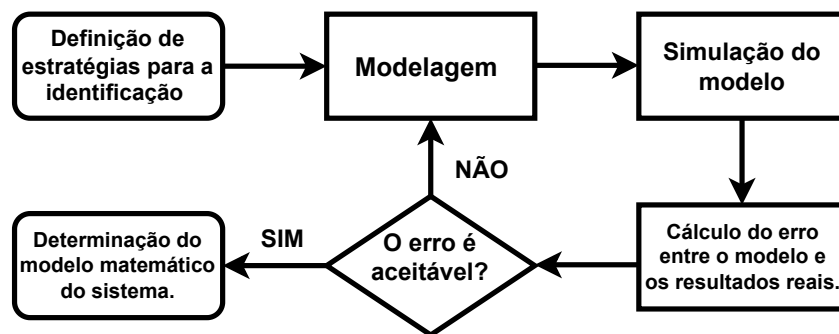
Como já mencionado, o ESP32-WROOM32 possui dois núcleos de processamento, definidos como CPU0 (PRO_CPU) e CPU1 (APP_CPU). A CPU0 é utilizada por padrão, de modo que, caso não seja implementado uma divisão de tarefas entre os dois núcleos, somente a PRO_CPU irá operar (SYSTEMS, 2023).

As duas CPUs presentes no microcontrolador podem se comunicar com periféricos e memórias cache internas. Para acessar a memória SRAM, utiliza-se uma unidade de gerenciamento de memória (MMU, do inglês *Memory Management Unit*) (ESPRESSIF, 2023). Destaca-se, entretanto, que o acesso a memória não é concorrente, ou seja, a memória permite o acesso de apenas uma CPU por vez.

2.3 Modelagem de Sistemas

A modelagem de sistemas conceitua-se como uma abstração matemática capaz de descrever o comportamento dinâmico do sistema, baseando-se nos sinais de entrada, saída e internos. Um modelo matemático considera as principais propriedades do sistema, acrescido de simplificações e constantes de proporcionalidade, de modo a garantir que a representação seja fidedigna ao sistema físico (VOJTESEK; DOSTAL; MASLAN, 2014). O procedimento básico para determinação de um modelo matemático é esquematizado na Figura 8.

Figura 8 – Procedimento para determinação de modelo matemático de um sistema.



Fonte: Do autor.

O objetivo de abstrair um sistema real para um modelo matemático, é permitir trabalhar com uma representação genérica que caracteriza adequadamente o sistema. Além disso, pode-se comparar as respostas do modelo com as obtidas no sistema real, com o intuito de verificar se o modelo representa de forma adequada o processo físico real que se deseja representar.

A relação de entrada e saída de um sistema real é dinâmica e descreve como as entradas aplicadas afetam as saídas. A compreensão dessa relação é essencial para projetar sistemas que respondam de maneira desejada às entradas. A relação pode ser representada por meio de equações diferenciais, para casos em que o sistema representado é contínuo no tempo, ou por equações de diferenças, para sistemas de tempo discreto.

As equações diferenciais são usadas para descrever sistemas cujas variáveis mudam de forma contínua em relação ao tempo ou a outra variável independente (BOYCE; DIPRIMA, 2015). A relação entre sinais de entrada e saída de um sistema pode ser definida por

$$\frac{dy(t)}{dt} = T\{x(t), t\}, \quad (2.2)$$

onde a derivada (variação) da saída $y(t)$ relativa ao tempo é ocasionada pela entrada $x(t)$ e/ou outras variáveis do sistema, e a transformação $T\{\cdot\}$ descreve como a saída muda temporalmente em função do sinal de entrada variante no tempo.

As equações de diferenças são usadas para sistemas discretos, nos quais as variáveis mudam em intervalos discretos de tempo. Nesse caso, a relação entre os sinais de entrada e saída pode ser expressa por uma expressão que relaciona a saída atual com a entrada atual ou, em alguns casos, a saída atual com as entradas e saídas anteriores (OPPENHEIM; SCHAFER, 2013).

Neste trabalho, foram considerados somente sistemas causais, em que a saída atual não depende de valores em instantes de tempo futuros. Nesse caso, a saída pode ser representada por meio da relação

$$y[nT_s] = T\{x[nT_s], x[nT_s - k], y[nT_s - k]\}, \quad k = [1, \infty], \quad (2.3)$$

em que a transformação $T\{\cdot\}$ descreve como a saída no instante nT_s é influenciada pela entrada atual e pelas entradas e saídas nos instantes anteriores $nT_s - k$, sendo $k \in \mathbb{Z}$, e T_s o período de amostragem.

Os modelos matemáticos podem ser definidos em diversos domínios, tais como tempo ou frequência, por exemplo. Nesse sentido, as transformadas são ferramentas utilizadas para converter uma função ou conjunto de dados de um domínio para outro, permitindo que um modelo complexo possa ser representado de maneira mais simples ou compreensível considerando um domínio diferente.

As funções de transferência são ferramentas importantes na teoria de controle e sistemas lineares, definidas a partir de transformadas matemáticas, como a Transformada de Laplace ($\mathcal{L}\{\cdot\}$) utilizada no domínio contínuo, e a Transformada Z ($\mathcal{Z}\{\cdot\}$) utilizada no domínio discreto. Essas funções expressam a razão da transformada da saída pela transformada da entrada (CARRARA, 2012). Como exemplo, pode-se definir a função de transferência $G(s)$ como

$$G(s) = \frac{Y(s)}{X(s)}, \quad (2.4)$$

ou seja, a razão da Transformada de Laplace da saída $Y(s)$ pela Transformada de Laplace da entrada $X(s)$.

Portanto, tendo-se uma representação da relação de entrada-saída de um sistema contínuo ou discreto, por meio de uma equação diferencial ou equação a diferenças, respectivamente, pode-se aplicar a Transformada de Laplace na saída e na entrada da equação diferencial ou, a Transformada Z na saída e na entrada da equação a diferenças, e estabelecer a razão saída-entrada, que é a função de transferência do sistema.

Para que seja possível converter uma expressão definida em tempo contínuo, representada por meio da Transformada de Laplace, para uma versão equivalente em tempo discreto, representada por meio da transformada Z , pode-se utilizar, por exemplo, a Transformada Bilinear. Essa técnica não é exclusiva, porém no desenvolvimento deste projeto optou-se utilizar ela. Essa transformada pode ser definida por

$$\begin{aligned}
 x(t) &\xrightarrow{\mathcal{L}} \mathcal{L}\{x(t)\} = X(s) \\
 X(s) &\rightarrow \text{Transformação Bilinear} \rightarrow X(z) = X(s)\Big|_{s=\frac{2}{T_s}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)} \\
 X(z) &\xrightarrow{\mathcal{Z}^{-1}} \mathcal{Z}^{-1}\{X(z)\} = x[kT_s]
 \end{aligned} \tag{2.5}$$

sendo $x(t)$ o sinal em tempo contínuo, $X(s)$ o sinal no domínio s obtido pela Transformada de Laplace de $x(t)$, T_s o período de amostragem, $X(z)$ o sinal no domínio z obtido pela Transformada Bilinear de $X(s)$, e $x[kT_s]$ o sinal em tempo discreto obtido pela Transformada Z inversa.

Essa abordagem foi utilizada neste trabalho para converter um modelo representado no domínio S para uma versão equivalente representada no domínio Z .

2.4 Considerações Parciais

Este capítulo apresentou uma revisão teórica envolvendo os principais conceitos necessários para o desenvolvimento da metodologia proposta neste trabalho. Foram apresentados os conceitos básicos necessários para compreensão dos processos de amostragem e reconstrução de sinais, e descritos alguns pontos principais que devem ser considerados ao se escolher uma plataforma de *hardware* para processamento digital de sinais em tempo real.

Finalmente, foram apresentadas características referentes aos processos de modelagem e identificação de sistemas, com destaque para a técnica da Transformada Bilinear, utilizada para converter funções de transferências de tempo contínuo (domínio S) para tempo discreto (domínio Z).

3 Metodologia

Neste capítulo, apresenta-se uma descrição da metodologia utilizada ao longo deste trabalho. Inicialmente, foram realizados ensaios para caracterização do ADC presente no microcontrolador, com o objetivo de avaliar a linearidade e regiões de operação do mesmo.

Na sequência, foram realizados ensaios para caracterizar o microcontrolador quanto aos limites de operação das frequências de amostragem e reconstrução, considerando-se o método de varredura para acesso aos registradores de uso específico dos periféricos. Os sinais reconstruídos foram adquiridos a partir de um osciloscópio digital e analisados nos domínios do tempo e da frequência.

Após o processo de caracterização da plataforma, foram realizados ensaios de validação do sistema. Para isso, dois modelos com características distintas foram implementados: o modelo de um tanque industrial, que representa o comportamento de um dos tanques presentes na planta didática SMAR-PD3F, e o modelo de um filtro passivo RC passa-baixas. Ao longo dos ensaios, sinais emulados foram comparados aos sinais medidos a partir dos sistemas físicos.

Por fim, é realizada uma análise sobre o suporte para DMA por meio da comunicação I2S que o ESP32 possui.

3.1 Caracterização do ADC

Inicialmente, foram realizados ensaios para caracterizar o ADC do microcontrolador ESP32-WROOM32. Para isso, utilizou-se uma fonte de alimentação CC (Tektronix, modelo PWS2323) como referência de entrada, e um osciloscópio digital (Tektronix, modelo TDS2012C) como referência de saída. Os dados adquiridos pelo microcontrolador foram registrados usando o *software* Arduino IDE, por meio uma plataforma de monitoramento serial RS-232.

O microcontrolador utilizado possui dois ADCs (ADC1 e ADC2) com resolução vertical de 12 bits e tensão de referência positiva ajustável. Nos ensaios realizados, o conversor foi configurado para operar com uma tensão de referência positiva de 3,3 V. Nessa configuração, conforme o fabricante, pode-se medir adequadamente tensões dentro do intervalo compreendido entre 150 mV e 2,45 V.

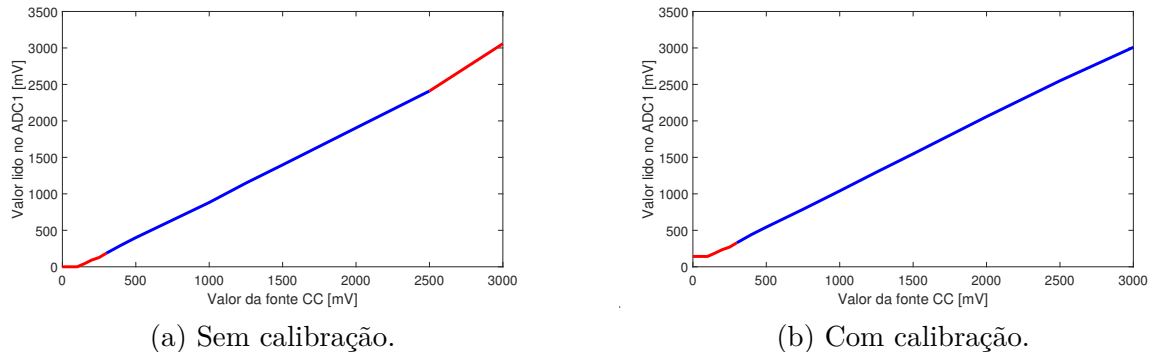
Foram realizados testes usando somente o ADC1 (*inputs* 32 à 36, e 39), uma vez que, segundo o fabricante, o módulo do ADC2 (*inputs* 0, 2, 4, 12 à 15, e 25 à 27) localiza-se fisicamente numa região próxima ao módulo de comunicação Wi-Fi presente no microcontrolador, e pode sofrer interferência significativa caso o módulo Wi-Fi seja ativado.

A fonte CC foi ajustada para fornecer tensões entre 0 e 3 V. Os sinais de tensão aplicados foram adquiridos pelo ADC1, e os valores digitais foram armazenados num arquivo de texto. O osciloscópio foi utilizado como referência, para verificar se os valores ajustados na fonte CC estavam de fato sendo aplicados na entrada do ADC1 do microcontrolador.

De acordo com Espressif (2015-2023), os módulos ADC presentes no microcontrolador possuem alguns problemas relacionados a linearidade e zona morta, que podem ser minimizados utilizando-se uma biblioteca de calibração fornecida pela empresa. Essa biblioteca foi utilizada e os testes descritos anteriormente foram repetidos, a fim de se verificar se o procedimento proposto pela empresa de fato é efetivo para contornar os problemas previamente informados.

A Figura 9 apresenta os resultados obtidos ao analisar os dados adquiridos pelo ADC1 do microcontrolador. Foram comparados os ensaios antes e após a utilização das funções de calibração fornecidas pelo fabricante. Os sinais obtidos de forma direta (sem a utilização da função de calibração) apresentam duas regiões com comportamento não linear, destacadas em vermelho na Figura 9a.

Figura 9 – Caracterização do módulo ADC1 presente no microcontrolador, com o objetivo de validar a utilização da biblioteca de calibração disponível.



Fonte: Do autor.

Após o processo de calibração, verifica-se que o problema de não linearidade do sinal adquirido foi resolvido, conforme mostra a Figura 9b. Entretanto, percebe-se a existência de uma zona morta (destacada com a cor vermelha), definida entre 0 e 0,25 V. Esse comportamento pode ser crítico para casos onde deseja-se monitorar tensões próximas a 0 V, exigindo que um circuito complementar para adição de um *offset* seja adicionado a plataforma de processamento proposta neste trabalho.

3.2 Caracterização do sistema de amostragem e reconstrução

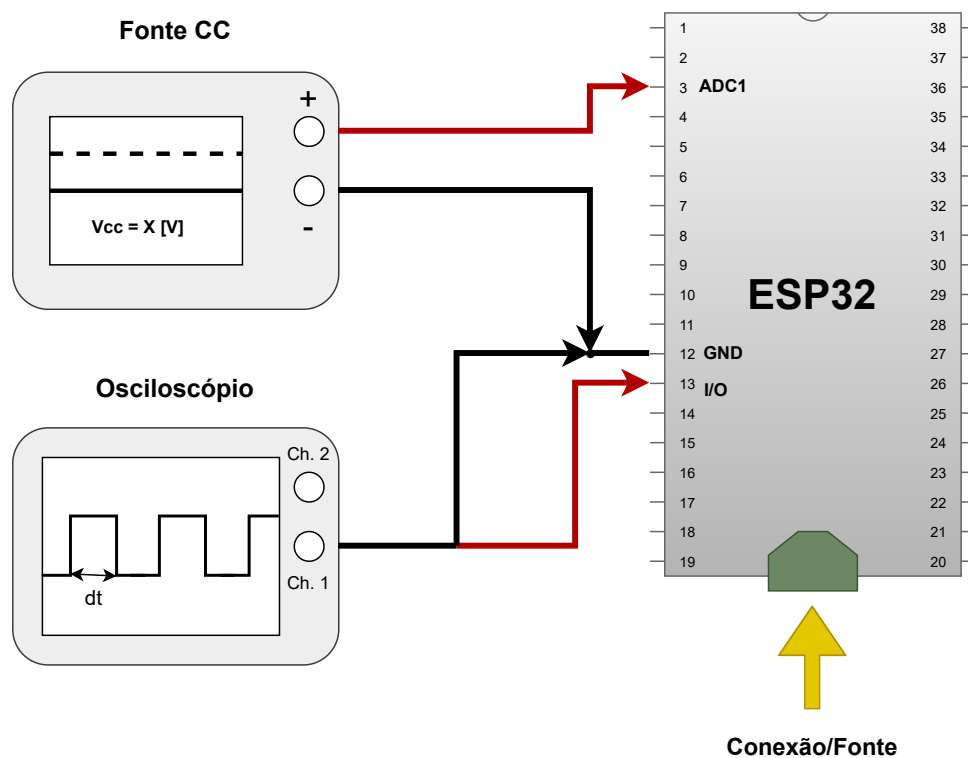
A análise das frequências de amostragem e reconstrução dos sinais foi realizada a fim de se avaliar os limites de operação do microcontrolador quando configurado para

operar em modo varredura. É importante ressaltar que os testes realizados foram sincronizados por um temporizador, ajustado para gerar um pedido de interrupção sempre que um determinado intervalo de tempo fosse alcançado, definindo o período de amostragem T para o processo de conversão analógico-digital.

Para monitorar o início e término do processo de amostragem, utilizou-se um pino de saída digital, configurado em nível alto no início do processo de aquisição e em nível baixo após o término do processo. O período de amostragem T_s foi ajustado de modo a garantir que houvesse um pequeno intervalo entre o término e o início de uma aquisição.

A fim de se determinar os limites para a frequência de amostragem dos sinais, montou-se o circuito apresentado na Figura 10.

Figura 10 – Configuração utilizada para determinação da máxima frequência de amostragem considerando o método varredura.



Fonte: Do autor.

Os ensaios foram realizados considerando a utilização ou não da função de calibração disponibilizada pelo fabricante. Na sequência, adicionou-se ao processo a conversão digital-analógica do sinal adquirido. Para isso, utilizou-se um dos DACs (DAC1) presente no microcontrolador. Esse dispositivo possui resolução vertical de 8 bits, ou seja, menor do que a resolução vertical do ADC do microcontrolador. Dessa forma, foi necessário realizar um ajuste no valor digital adquirido para que o sinal de entrada pudesse ser utilizado como referência para controlar os níveis de tensão gerados pelo conversor digital-analógico. A Tabela 1 mostra os tempos obtidos nos testes.

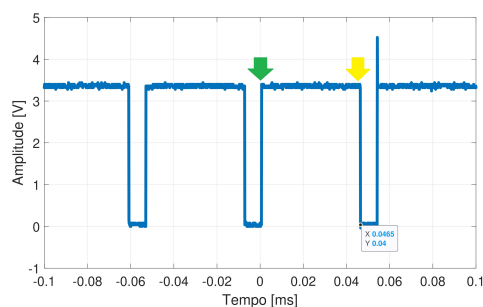
Tabela 1 – Comparação entre os períodos de amostragem dos processos de amostragem (antes e após a calibração do ADC1) e reconstrução dos sinais sem nenhum processamento adicional.

	Período de amostragem	Frequência de amostragem
ADC1 descalibrado	46,50 μs	21,51 kHz
ADC1 calibrado	46,70 μs	21,41 kHz
ADC1 calibrado e DAC	65,20 μs	15,34 kHz

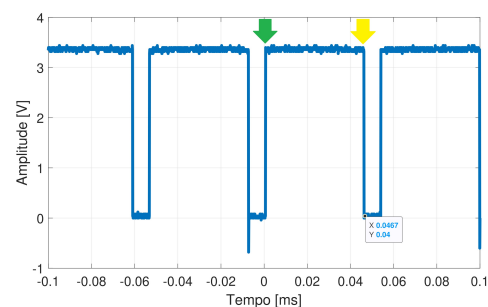
Fonte: Do autor.

Um pino de saída digital foi utilizado como referência para a medição dos tempos apresentados na Tabela 1. Ao longo dos ensaios, o nível lógico do pino escolhido foi alterado de baixo para alto no início do processo de aquisição, indicado pela seta verde, e de alto para baixo no final do processo, indicado pela seta amarela. Um osciloscópio digital foi utilizado para medir os tempos associados aos processos de aquisição, calibração e reconstrução dos sinais. A Figura 11 apresenta exemplos de formas de onda adquiridas nesta etapa, levando-se em consideração os três casos mencionados.

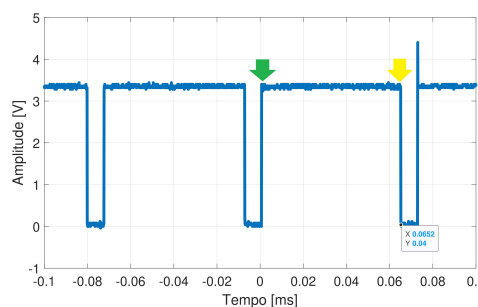
Figura 11 – Saídas dos processos de amostragem (antes e após a calibração do ADC1) e reconstrução dos sinais.



(a) Somente o processo de aquisição pelo ADC sem calibração.



(b) Somente o processo de aquisição pelo ADC com calibração..



(c) Processo de aquisição pelo ADC com calibração e reconstrução pelo DAC.

Fonte: Do autor.

A comparação entre os tempos de amostragem, antes e após a calibração do ADC1, indica que o tempo necessário para executar a função de calibração do ADC1 fornecida pelo fabricante do microcontrolador é de, aproximadamente, 0,2 μs . Dada a importância

do processo e por se tratar de um tempo reduzido, optou-se por utilizar essa função em todas as análises apresentadas na sequência.

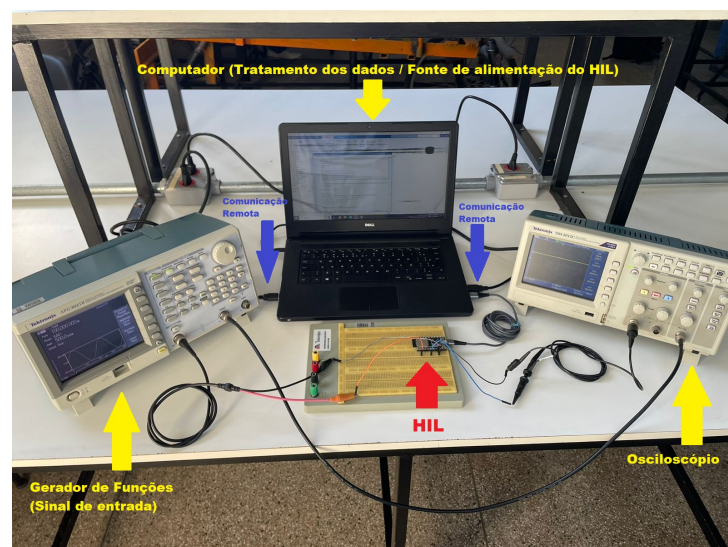
Comparando os tempos do processo de amostragem após a calibração com e sem reconstrução, observa-se que o tempo necessário para a reconstrução dos sinais é de, aproximadamente, $18 \mu\text{s}$. Essa análise indica que o tempo gasto pelo processo de amostragem é maior do que o tempo gasto pelo processo de reconstrução dos sinais, realizado pelo DAC. Essa variação pode estar relacionada a diferença entre o número de bits dos dois conversores (12 bits para o ADC1 e 8 bits para o DAC1). Ressalta-se, entretanto, que os testes foram realizados utilizando funções em modo varredura.

Para os ensaios realizados ao longo deste trabalho, ajustou-se a frequência de amostragem e reconstrução dos sinais para 10 kHz, valor inferior a menor frequência encontrada, de 15,34 kHz, e mostrada na Tabela 1. Esse ajuste foi feito utilizando um temporizador, responsável por gerar um novo pedido de interrupção a cada $100 \mu\text{s}$.

3.3 Análise do sistema de amostragem e reconstrução dos sinais

Após a caracterização do sistema de amostragem e reconstrução, foram realizados testes com sinais de entrada em configurações distintas, a fim de avaliar as condições de operação sistema. A Figura 12 apresenta uma foto da bancada utilizada para realização dos ensaios.

Figura 12 – Bancada utilizada ao longo da realização dos ensaios de caracterização do sistema.



Fonte: Do autor.

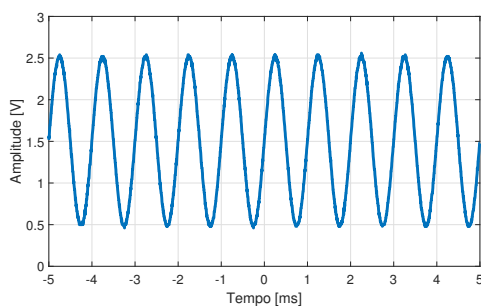
Um gerador de sinais (Tektronix, modelo AFG3021B) foi utilizado para ajustar um sinal senoidal na entrada do ADC1 do microcontrolador. Foram aplicados sinais senoidais, com frequências de 1 kHz, 2 kHz e 3 kHz, com $2 V_{pp}$ e $1,5 V$ de *offset*.

Os sinais adquiridos foram ajustados e enviados ao DAC1 do microcontrolador, e o sinal analógico gerado foi adquirido utilizando-se o osciloscópio digital. Para sincronizar o gerador de sinais e o osciloscópio, utilizou-se o *trigger* de saída do gerador de sinais como entrada de *trigger* externa para o osciloscópio digital. Um algoritmo foi desenvolvido para controlar os instrumentos de bancada de forma automática. Além disso, os sinais adquiridos por meio do osciloscópio foram armazenados para serem utilizados em análises descritas na sequência.

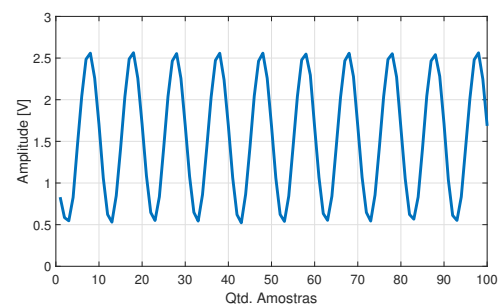
Em seguida, a fim de se avaliar a influência do ruído no processo de aquisição e reconstrução, repetiu-se o procedimento descrito anteriormente, adicionando-se ruído branco gaussiano ao sinal, com amplitude máxima de 25% em relação ao sinal aplicado na entrada do ADC1.

A Figura 13 apresenta os resultados obtidos após o processo de amostragem e reconstrução de um sinal senoidal com frequência conhecida aplicado na entrada ADC1 do microcontrolador.

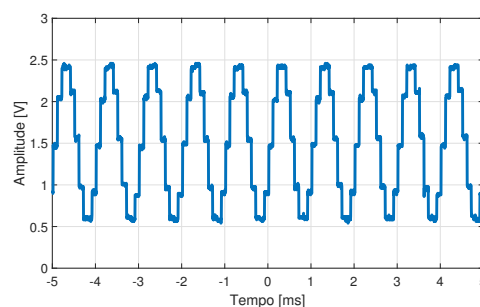
Figura 13 – Caracterização do sistema de amostragem e reconstrução considerando sinal de entrada senoidal, 1 kHz, sem ruído.



(a) Sinal de entrada.



(b) Sinal interno do ESP32.



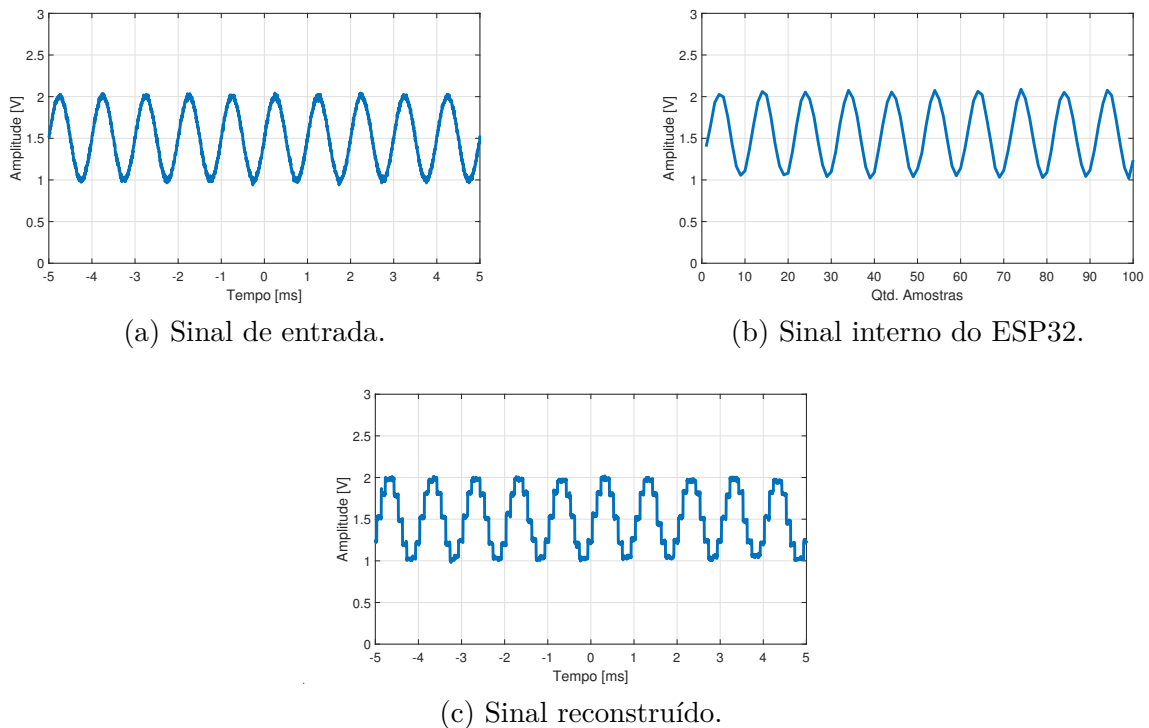
(c) Sinal reconstruído.

Fonte: Do autor.

O sinal apresentado na Figura 13a, obtido a partir do osciloscópio digital, foi utilizado como referência para comparação. A forma de onda apresentada na Figura 13b apresenta a forma de onda interna do microcontrolador. A Figura 13c apresenta a forma de onda reconstruída a partir do sinal apresentado na Figura 13b. Observam-se claramente os efeitos da quantização, associados à redução do número de bits do DAC1.

O ensaio foi repetido, adicionando-se 25% de ruído branco gaussiano ao sinal senoidal de entrada. O resultado obtido é apresentado na Figura 14, em que se percebe que a inserção do ruído ocasiona a redução de, aproximadamente, 50% da tensão pico a pico medida para os três sinais adquiridos.

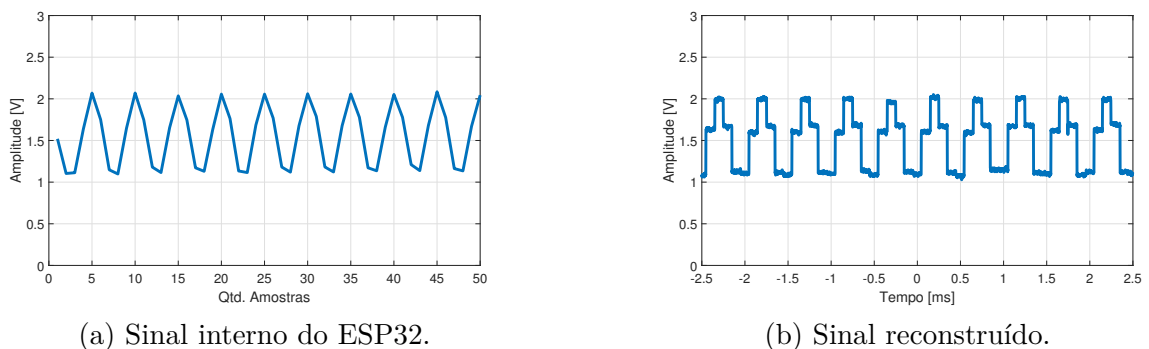
Figura 14 – Caracterização do sistema de amostragem e reconstrução considerando sinal de entrada senoidal, 1 kHz, com 25% de ruído.



Fonte: Do autor.

A Figura 15a apresenta os resultados considerando um sinal de entrada senoidal com frequência de 2 kHz. Nessas condições, verifica-se que a saída reconstruída apresenta um sinal com formato comprometido em comparação com o observado no caso anterior.

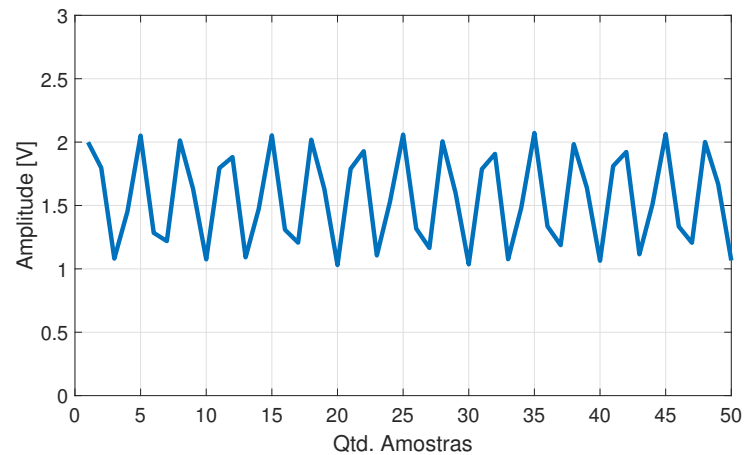
Figura 15 – Caracterização do sistema de amostragem e reconstrução considerando sinal de entrada senoidal, 2 kHz, com 25% de ruído.



Fonte: Do autor.

O teste final considerou um sinal senoidal com frequência de 3 kHz, com adição de ruído. O objetivo desse ensaio é avaliar os efeitos da redução do número de pontos por período somado ao acréscimo de ruído ao sinal de entrada. A Figura 16 apresenta o resultado obtido.

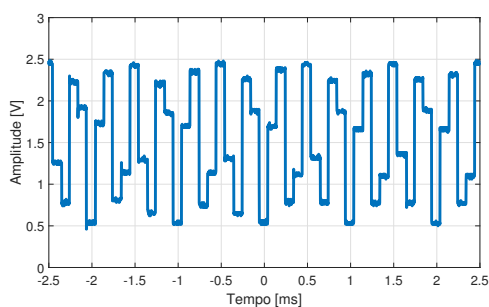
Figura 16 – Sinal interno obtido após amostragem de sinal de entrada senoidal, 3 kHz, com 25% de ruído.



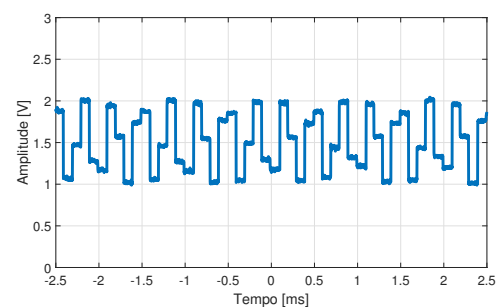
Fonte: Do autor.

Nota-se que o aumento do ruído, somado à redução de número de pontos por período, afeta de forma significativa a representação do sinal digital utilizado como referência pelo microcontrolador. A Figura 17 apresenta a forma de onda dos sinais senoidais de 3 kHz, com e sem ruído, reconstruídos a partir do DAC1. Observa-se que, nessas condições, não se pode garantir uma boa reconstrução dos sinais no domínio temporal.

Figura 17 – Sinais senoidais, com frequência de 3 kHz, reconstruídos.



(a) Sinal sem ruído.



(b) Sinal com ruído.

Fonte: Do autor.

3.4 Avaliação do desempenho do sistema em operações de processamento em tempo real

Para testar a capacidade de processamento do sistema, foram realizados testes utilizando um filtro de média-móvel, um modelo utilizado para relacionar vazão de entrada e nível em tanques industriais, e um filtro RC passivo passa-baixas.

3.4.1 Filtro média-móvel

A fim de se avaliar os tempos necessários para o processamento de algoritmos, realizou-se um teste inicial considerando um filtro de média-móvel. Matematicamente, pode-se definir o comportamento de um filtro média-móvel a partir da expressão

$$y[i] = \frac{1}{M} \sum_{j=0}^{M-1} x[i + j], \quad (3.1)$$

sendo M o número de pontos da janela, x o vetor de entrada e y o vetor de saída.

Esse tipo de filtro foi escolhido, pois pode ser implementado por meio de operações básicas, como soma e divisão, e seu grau de complexidade pode ser alterado com facilidade por meio da alteração do número de pontos utilizados para definir o tamanho da janela. Além disso, uma vez que são necessárias informações de entradas anteriores para o cálculo da saída atual, a estrutura desse tipo de filtro exige que sejam realizadas operações de leitura e escrita na memória SRAM do micro a cada iteração. Esse conjunto de fatores motivou a utilização desse tipo de filtro nos testes iniciais.

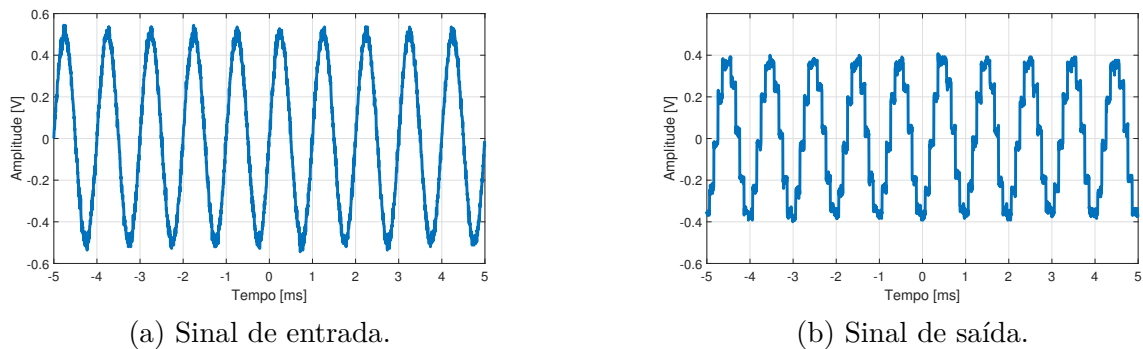
O filtro de média-móvel implementado foi ajustado inicialmente com uma janela de 10 amostras. Ajustou-se na entrada um sinal senoidal, com frequência de 1 kHz, amplitude de $2 V_{pp}$ e *offset* de 1,5 V, com um ruído branco gaussiano com amplitude de 25% em relação ao sinal de entrada.

O sistema de aquisição foi ajustado com frequência de amostragem de 10kHz, o ADC1 foi utilizado como canal de entrada e o DAC1 foi utilizado como canal de saída. Assim como nos casos anteriores, os sinais de saída foram adquiridos por meio do osciloscópio digital e processados num computador.

Um pino de saída digital foi utilizado para monitorar os tempos envolvidos nos processos de amostragem, processamento e reconstrução dos sinais, assim como descrito nas seções iniciais deste Capítulo.

A Figura 18 apresenta os sinais amostrado e reconstruído obtidos ao longo dos ensaios utilizando o filtro de média-móvel. Durante os ensaios, a frequência de amostragem do sistema foi ajustada em 10 kHz. Destaca-se que o objetivo deste ensaio é avaliar o desempenho da plataforma de processamento ao longo da execução de algoritmos executados em tempo real. Dessa forma, não se levou em consideração a qualidade da resposta do filtro implementado, mas sim o tempo de execução do mesmo.

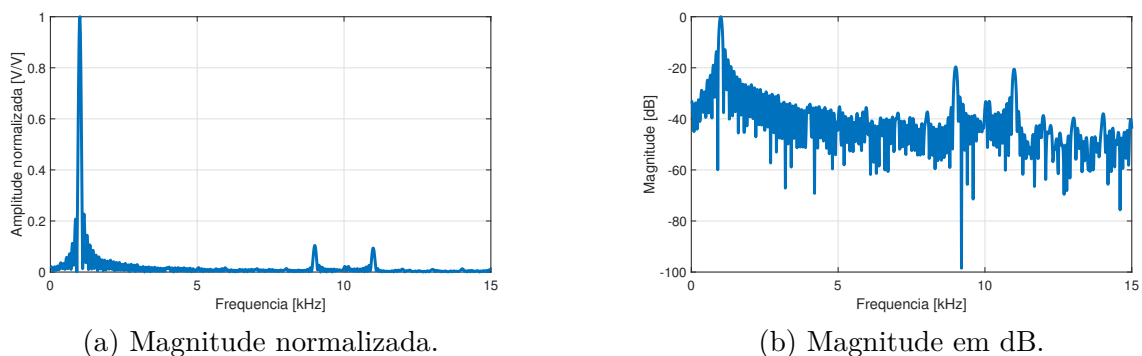
Figura 18 – Sinais de entrada e saída utilizados nos ensaios realizados com o filtro de média-móvel.



Fonte: Do autor.

A Figura 19 apresenta a análise espectral dos sinais de entrada e saída apresentados na Figura 18. Os resultados mostraram que o tempo de processamento para essa configuração foi de, aproximadamente, $4 \mu\text{s}$. Dessa forma, somando-se os tempos de aquisição, reconstrução e processamento, tem-se um período total de, aproximadamente, $70 \mu\text{s}$, o que corresponde a 70% do período de amostragem definido. O tempo utilizado efetivamente para as tarefas de processamento é equivalente a 4% do período de amostragem. Esse efeito destaca as limitações do método de aquisição e reconstrução por varredura, cujos tempos superam 60% do período de amostragem.

Figura 19 – Análise espectral do sinal de saída obtido no ensaio realizado com o filtro de média-móvel.



Fonte: Do autor.

3.4.2 Filtro RC Passa-Baixas

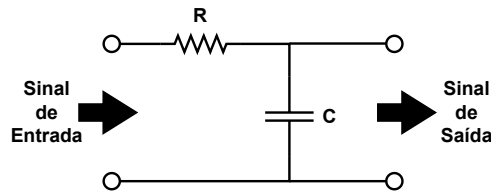
Para avaliar o comportamento do sistema HIL quando submetido a condições críticas de operação, decidiu-se emular um Filtro RC passa-baixas, pois com este sistema pode-se arbitrar a frequência de operação, que para os testes, será igual à frequência de corte.

O Filtro RC passa-baixas é modelado genericamente por meio de

$$G(s) = \frac{V_{entrada}}{V_{saída}} = \frac{1}{s + \frac{1}{RC}}, \quad (3.2)$$

em que $G(s)$ é a função de transferência do sistema, R o valor da resistência e C o valor da capacitância. A Figura 20 mostra o esquema de um Filtro RC passa-baixas genérico.

Figura 20 – Esquema genérico de um Filtro RC passa-baixas.



Fonte: Do autor.

Assim como para o modelo, a frequência de corte do sistema acima é definida pelos valores de resistência e capacitância, isto é

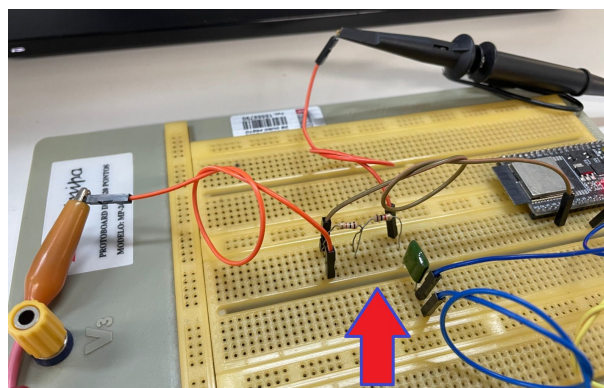
$$f_c = \frac{1}{2\pi RC}, \quad (3.3)$$

em que f_c é a frequência de corte.

Com isso, realizou-se o dimensionamento do Filtro RC passa-baixa, em que foram arbitrados os valores teóricos de $2 \text{ k}\Omega$ para a resistência, e de 110 nF para o capacitância, definindo a frequência de corte em $723,43 \text{ Hz}$. A utilização desses valores está considerando uma aplicação onde $f_s > 10 \cdot f$, sendo o limite para f de 1 kHz .

Contudo, os componentes utilizados apresentam tolerância de 10% , por isso optou-se por medir os valores deles a partir de um multímetro de bancada com 6,5 dígitos de precisão (Tektronix, modelo DMM4040), em que o valor da resistência foi de $1,96 \text{ k}\Omega$, e da capacitância de 109 nF , dimensionando a frequência de corte com $744,96 \text{ Hz}$. A Figura 21 mostra o Filtro RC passa-baixas analógico montado.

Figura 21 – Filtro RC Analógico montado na bancada de ensaios.



Fonte: Do autor.

Com esses valores foi gerado o modelo

$$G(s) = \frac{4680,771}{s + 4680,771}, \quad (3.4)$$

que foi discretizado utilizando a transformação bilinear, empregando o período de amostragem $T_s = 0.0001$ s, e obteve-se a seguinte expressão

$$G(z) = \frac{0,189z + 0,189}{z - 0,621} = \frac{Y(z)}{X(z)} \quad (3.5)$$

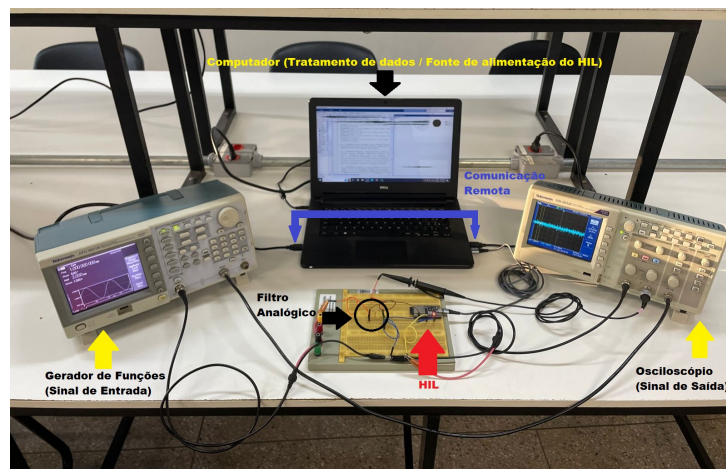
em que $Y(z)$ é a saída do sistema, e $X(z)$ a entrada do sistema.

Por fim, (3.9) foi organizada para equacionar a relação de entrada e saída. Ao aplicar a transformada Z inversa, tem-se a seguinte equação de diferenças

$$y[n] = 0,189x[n] + 0,189x[n - 1] + 0,621y[n - 1] \quad (3.6)$$

A Figura 22 mostra a bancada de testes montada, em que o gerador de funções e osciloscópio são controlados remotamente por meio do Matlab, e a saída de trigger do gerador de sinais foi utilizada como entrada de trigger para o osciloscópio digital.

Figura 22 – Bancada utilizada ao longo da realização dos ensaios do sistema.



Fonte: Do autor.

Para todos os testes realizados, ajusta-se um valor de frequência inicial no gerador de sinais, considerando um sinal de entrada com *offset* de 1,5 V e $2 V_{pp}$. Após, realiza-se a aquisição simultânea dos sinais de saída do filtro analógico e HIL, canais 1 e 2 do osciloscópio, respectivamente. Na sequência, compara-se as respostas no domínio do tempo, e por fim, altera-se a frequência de saída do gerador e repete o procedimento. O ajuste do valor de frequência do sinal no gerador é feito por meio da comunicação com o Matlab, em que foram definidas as frequências limites de 100 Hz à 1 kHz.

Após a realização dos testes com sinal de entrada senoidal, fez-se três ensaios utilizando como sinais de entrada: (i) uma onda quadrada; (ii) uma onda triangular: e

(iii) uma onda dente de serra. O objetivo dos testes é verificar se o microcontrolador emula corretamente o comportamento do filtro quando submetido a sinais não senoidais. A configuração para todos os sinais de entrada foi de $2 V_{pp}$ com *offset* de 1,5 V e frequência de 50 Hz. Escolheu-se esse valor para que relação entre a frequência de amostragem e a frequência do sinal fosse maior que cem vezes.

Para o teste, foi definido um novo modelo que representasse o filtro analógico. Inicialmente, dimensionou-se os valores de 15 k Ω para a resistência, e de 110 nF para a capacitância, definindo a frequência de corte em 96,46 Hz. Contudo, os componentes utilizados apresentam tolerância de 10 %, por isso optou-se por medir os valores deles a partir de um multímetro portátil (Minipa, modelo ET-2082E), em que o valor da resistência foi de 14,60 k Ω , e da capacitância de 108 nF, dimensionando a frequência de corte com 100,94 Hz.

Com esses valores foi gerado o modelo

$$G(s) = \frac{634,200}{s + 634,200}, \quad (3.7)$$

que foi discretizado utilizando a transformação bilinear, empregando o período de amostragem $T_s = 0.0001$ s, e obteve-se a seguinte expressão

$$G(z) = \frac{0,031z + 0,031}{z - 0,939} = \frac{Y(z)}{X(z)} \quad (3.8)$$

em que $Y(z)$ é a saída do sistema, e $X(z)$ a entrada do sistema.

Por fim, (3.9) foi organizada para equacionar a relação de entrada e saída. Ao aplicar a transformada Z inversa, tem-se a seguinte equação de diferenças

$$y[n] = 0,031x[n] + 0,031x[n - 1] + 0,939y[n - 1] \quad (3.9)$$

Para os testes realizados, ajusta-se a forma de onda com suas características no gerador de sinais. Após, realiza-se a aquisição simultânea dos sinais de saída do filtro analógico e HIL, canais 1 e 2 do osciloscópio, respectivamente. Por fim, compara-se as respostas no domínio do tempo.

3.4.3 Modelo de um tanque industrial

Para avaliar o desempenho do sistema HIL, optou-se por utilizar um modelo obtido a partir da análise de um tanque industrial presente na planta piloto SMAR-PD3-F. O modelo utilizado neste trabalho foi obtido por Lage (2019), que em seu trabalho, utilizou um sistema composto por bomba monofásica *on/off*, válvula pneumática com abertura controlável por um sinal de referência, tanque, sensor de vazão e sensor de nível.

O autor realizou uma série de ensaios com o sistema operando em malha aberta, a fim de obter curvas características que pudessem ser utilizadas para identificar o sistema

por meio do método caixa-preta, relacionando o valor percentual de abertura da válvula pneumática ao nível de água medido no tanque.

Inicialmente, foram comparados com os dados adquiridos aos modelos apresentados por Lage (2019), que em seu trabalho, utilizou quatro pontos de operação distintos para controlar a abertura da válvula pneumática: 25%, 50%, 75% e 100%.

Conforme os resultados obtidos pelo autor, os pontos de operação referentes a 50% e 75% de abertura da válvula pneumática apresentam ganhos e constantes de tempo semelhantes. Esses resultados foram obtidos, comparando-se a resposta dinâmica temporal dos modelos aos valores medidos. Dessa forma, optou-se por utilizar esses dois pontos de operação ao longo deste trabalho.

Os sinais de entrada e saída para cada ponto de operação, adquiridos em função do tempo, foram utilizados para determinação das funções de transferência discretas do sistema. Esse processo foi realizado por meio da ferramenta de identificação de sistemas do *software* Matlab.

Definindo-se inicialmente uma frequência de amostragem de 1 Hz, valor informado Lage (2019), e utilizando-se a ferramenta *System Identification* para identificação de sistemas do *software* Matlab em modo discreto, foram obtidas as expressões

$$G_{50}(z) = \frac{3,015 \cdot 10^{-3} z^{-1}}{1 - 1,605 z^{-1} + 6,081 \cdot 10^{-1} z^{-2}}, \quad (3.10)$$

e

$$G_{75}(z) = \frac{4,131 \cdot 10^{-3} z^{-1}}{1 - 1,435 z^{-1} + 4,394 \cdot 10^{-1} z^{-2}}. \quad (3.11)$$

Essas expressões foram representadas por meio de equações de diferenças, definidas como

$$y_{50}[n] = 3,015 \cdot 10^{-3} x[n-1] + 1,605 y[n-1] - 6,081 \cdot 10^{-1} y[n-2] \quad (3.12)$$

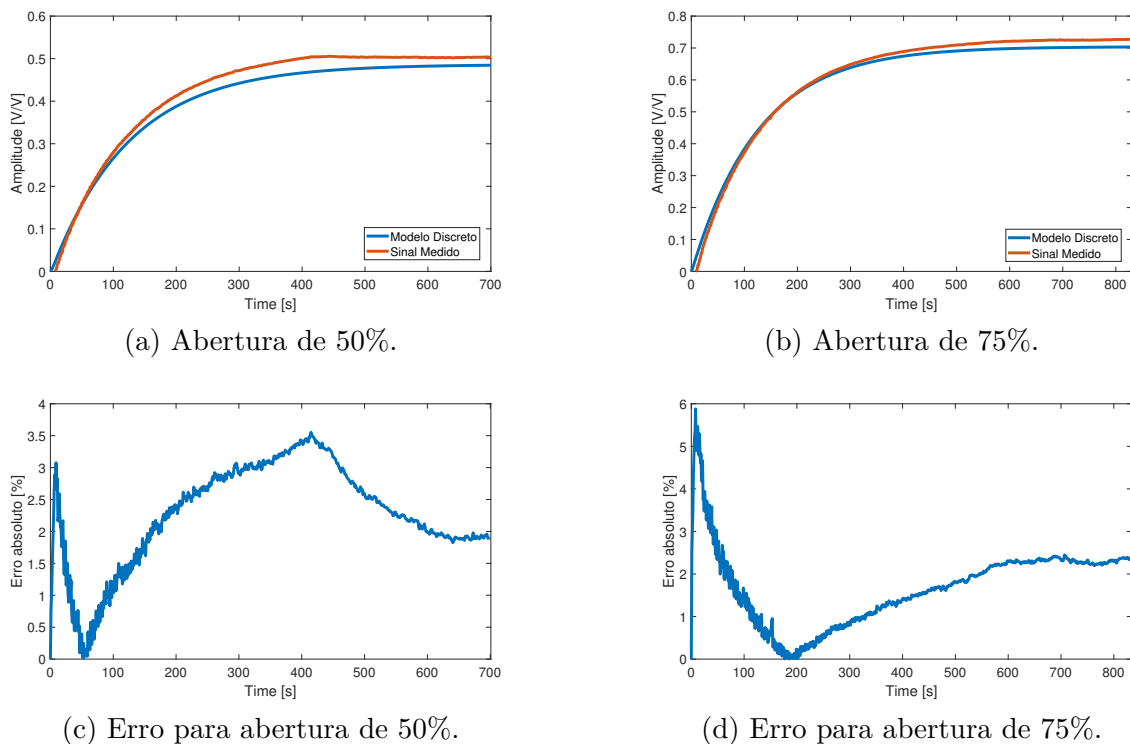
e

$$y_{75}[n] = 4,131 \cdot 10^{-3} x[n-1] + 1,435 y[n-1] - 4,394 \cdot 10^{-1} y[n-2], \quad (3.13)$$

que foram embarcadas na plataforma de processamento proposta neste trabalho.

A Figura 23 apresenta os resultados obtidos ao simular (3.4) e (3.5) no Matlab, considerando os pontos de operação referentes as aberturas de 50% e 75%. Nas Figuras 23a e 23b, as curvas vermelhas são as respostas obtidas pela simulação de (3.10) e (3.11), e as curvas azuis são as respostas medidas por Lage (2019). Os erros entre o sinal medido e modelo discreto são mostrados nas Figuras 23c e 23d.

Figura 23 – Comparação entre sinais medidos por Lage (2019) em seu trabalho, e os sinais obtidos a partir dos modelos discretos considerando dois pontos de operação.



Fonte: Do autor.

Observa-se que o erro em regime permanente estabiliza-se em, aproximadamente, 2% e 2,2% para os pontos de operação em questão. Esses erros, possivelmente, estão relacionados com a imprecisão das medições realizadas pelo sensor de nível, e por possíveis falhas no processo de modelagem do sistema.

3.5 Controle dos processos de aquisição e reconstrução por DMA

Determinadas aplicações necessitam de uma frequência de amostragem maior do que a definida neste trabalho, e uma forma de aumentar a velocidade de processamento de microcontroladores é a utilização do Acesso Direto à Memória (DMA, do inglês *Direct Memory Access*), caso o *hardware* possua este recurso.

O DMA é uma técnica que permite a transferência direta de dados entre periféricos sem a intervenção direta da CPU. No caso do ESP32, a transferência de dados via DMA é feita por meio do periférico de comunicação I2S (do inglês, *Inter-IC Sound*). A comunicação I2S é um protocolo utilizado para transmitir dados de áudio digital entre dispositivos, como microfones, alto-falantes e processadores de sinal digital.

Existe uma biblioteca para a Arduino IDE que possibilita o controle dos processos de aquisição e reconstrução por DMA no ESP32, porém existe um número mínimo de amostras a serem coletadas, e não foi possível processar os dados adquiridos enquanto

uma nova janela de dados era amostrada. Com isso, os resultados obtidos mostram que essa configuração não atende aos requisitos de um sistema HIL, haja vista que as restrições associadas ao modo de operação do sistema impedem que uma amostra seja processada por vez.

3.6 Considerações Parciais

Neste capítulo, foram descritos os procedimentos utilizados para caracterizar a plataforma HIL proposta neste trabalho. Inicialmente, foram realizados ensaios para caracterização dos conversores ADC e DAC, além de testes para verificação dos limites das frequências de amostragem e reconstrução do sistema proposto.

Na sequência, foram realizadas análises dos sinais reconstruídos no domínio da frequência, a fim de se avaliar os efeitos causados pelo processo de quantização no processo de reconstrução dos sinais.

Após, foram realizados testes para validação do sistema, avaliando-se o desempenho do mesmo ao executar o algoritmo de um filtro média-móvel, a emulação do modelo de um tanque industrial em tempo real, e a emulação de um Filtro RC.

Finalmente, avaliou-se a possibilidade de se utilizar o método de aquisição por DMA a partir do periférico I2S, presente no microcontrolador.

4 Resultados e discussões

Neste capítulo, são apresentados os principais resultados obtidos ao longo do desenvolvimento deste trabalho.

Para os testes iniciais, foram realizadas comparações entre um filtro analógico passa-baixas RC e seu equivalente digital, implementado na plataforma. Essa comparação teve como objetivo validar o sistema em condições adversas, em que a relação entre frequências de amostragem e reconstrução se aproxima do limite projetado.

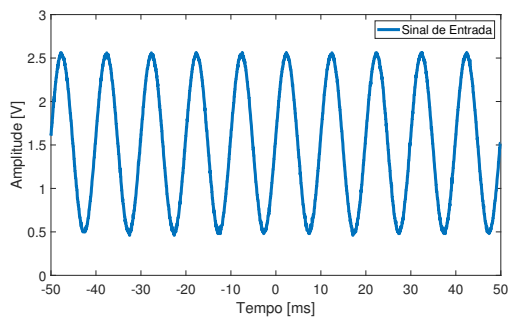
Ao final, fez-se os testes de validação do HIL, em que utilizou-se como sistema a ser emulado a planta didática SMAR PD3-F presente na instituição de ensino. O modelo empregado, bem como os sinais de referência aplicados ao longo dos testes, foram adquiridos por Lage (2019). Inicialmente, realiza-se a comparação entre as respostas obtidas pelos modelos em equações de diferenças encontrados no Capítulo 3, e os sinais gerados internamente no microcontrolador. Na sequência, compara-se os sinais analógicos gerados na saída do DAC aos valores de referência adquiridos por Lage (2019). Após, realizou-se uma comparação entre o desempenho dos dois sistemas operando com sinal de controle, aplicando-se na entrada do sistema HIL o mesmo sinal de controle utilizado por Lage (2019) para controlar o nível do tanque da planta real.

4.1 Emulação de Filtro Passivo RC Passa-Baixas

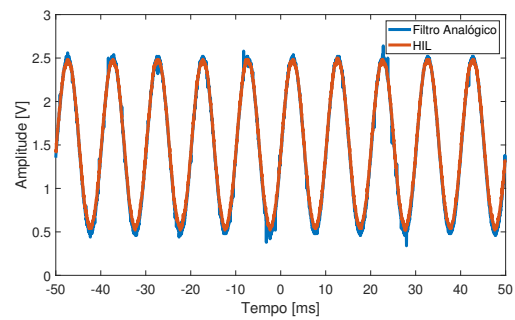
A fim de validar a capacidade de processamento da plataforma, utilizou-se um modelo discreto de um filtro RC passa-baixas. Como mencionado, a saída do filtro analógico real foi comparada à saída do sistema HIL embarcado com o modelo. Os resultados obtidos foram analisados no domínio do tempo. As Figuras 24, 25, e 26, mostram os resultados obtidos para sinais de entrada senoidais com frequências de 100 Hz, 400 Hz e 700 Hz, respectivamente.

Nota-se pelas figuras, que o aumento da frequência de entrada, e conseqüentemente, a proximidade com a frequência de corte de 744,96 Hz, ocasiona um atraso da resposta do HIL. Este erro está relacionado com a capacidade de processamento do *hardware*.

Figura 24 – Respostas do sistema HIL e do filtro analógico, para entrada senoidal com frequência de 100 Hz.



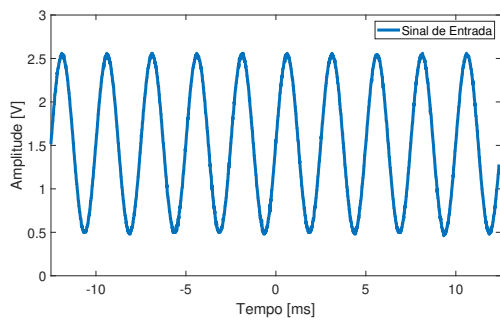
(a) Sinal de entrada com 100 Hz.



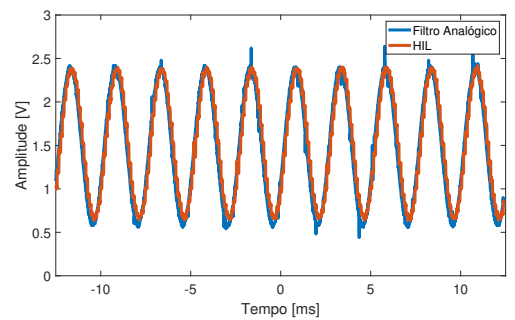
(b) Respostas do HIL e do filtro analógico.

Fonte: Do autor.

Figura 25 – Respostas do sistema HIL e do filtro analógico, para entrada senoidal com frequência de 400 Hz.



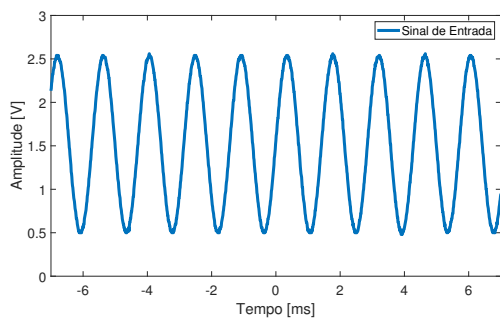
(a) Sinal de entrada com 400 Hz.



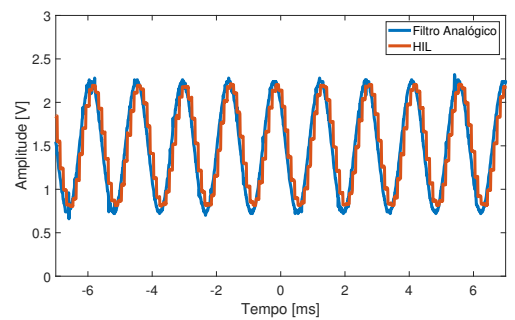
(b) Respostas do HIL e do filtro analógico.

Fonte: Do autor.

Figura 26 – Respostas do sistema HIL e do filtro analógico, para entrada senoidal com frequência de 700 Hz.



(a) Sinal de entrada com 700 Hz.

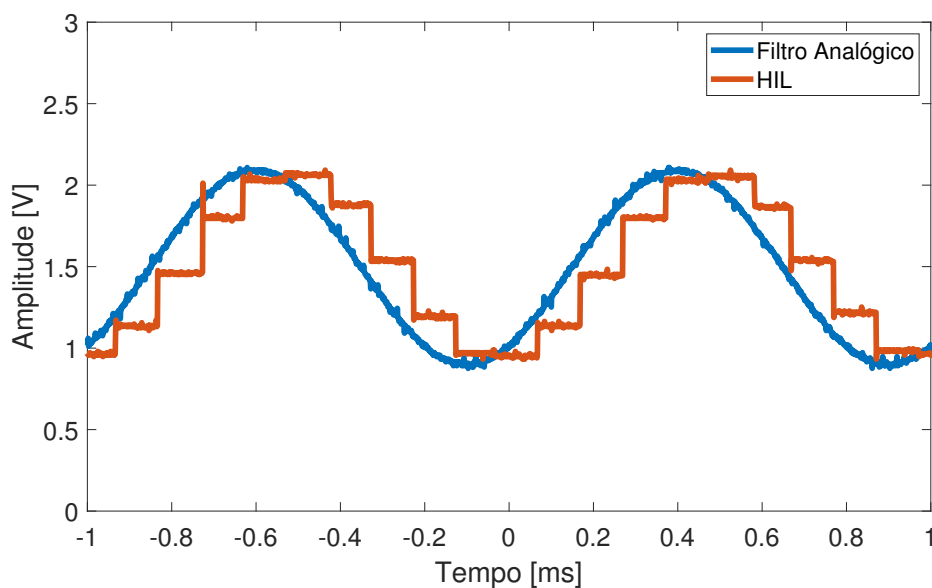


(b) Respostas do HIL e do filtro analógico.

Fonte: Do autor.

A Figura 27 mostra o teste realizado para um sinal de entrada senoidal de 1 kHz. Evidencia-se o efeito da quantização do sinal analógico reconstruído pelo DAC, a tendência é que esse efeito se destaque a medida em que a relação entre a frequência de amostragem e a frequência do sinal diminui. Com base nesses testes, esse erro inviabiliza o ESP32 para o uso como HIL caso a frequência de entrada seja próxima à 10% da frequência de amostragem, que para os teste foi estabelecida em 10 kHz.

Figura 27 – Respostas do HIL e do filtro analógico, para entrada senoidal com frequência de 1 kHz.

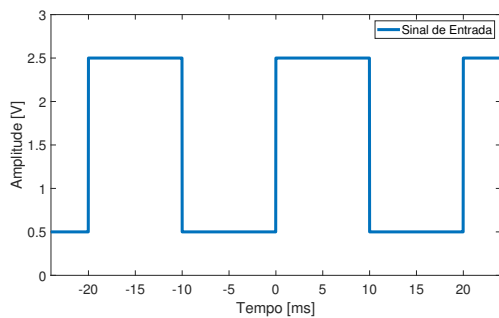


Fonte: Do autor.

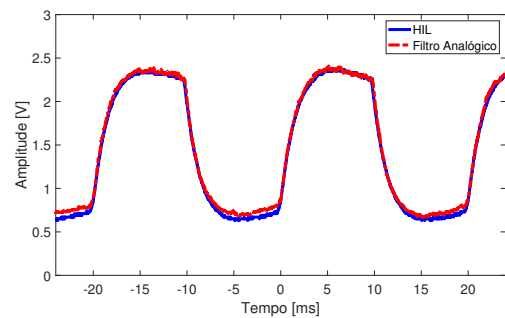
A fim de verificar o comportamento da plataforma para sinais de entrada não senoidais, utilizou-se um modelo discreto de um filtro RC passa-baixas considerando a relação de cem vezes entre a frequência de amostragem e a frequência do sinal. Como mencionado, a saída do filtro analógico real foi comparada à saída do sistema HIL embarcado com o modelo. Os resultados obtidos foram analisados no domínio do tempo. As Figuras 28, 29, e 30, mostram os resultados obtidos para os sinais de entrada em onda quadrada, onda triangular e onda dente de serra, respectivamente.

Nota-se pelas figuras, que as regiões de discontinuidades são representadas suavemente. Isso ocorre, porque essas regiões apresentam harmônicos de alta frequência, que são filtrados por estarem com frequência acima da frequência de corte. Apesar disso, o HIL conseguiu emular o comportamento do filtro analógico para os três testes realizados.

Figura 28 – Respostas do sistema HIL e do filtro analógico, para entrada em formato de onda quadrada com $2 V_{pp}$, *offset* de 1,5 V e frequência de 50 Hz.



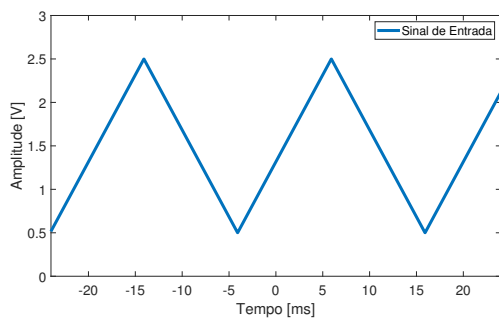
(a) Sinal de entrada onda quadrada.



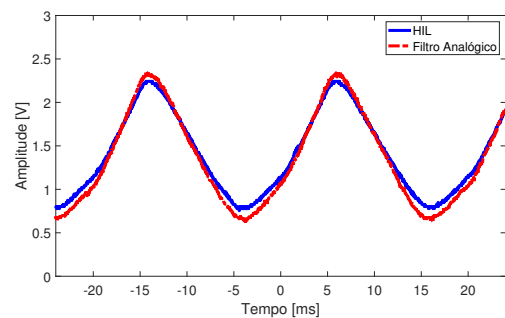
(b) Respostas do HIL e do filtro analógico.

Fonte: Do autor.

Figura 29 – Respostas do sistema HIL e do filtro analógico, para entrada em formato de onda triangular com $2 V_{pp}$, *offset* de 1,5 V e frequência de 50 Hz.



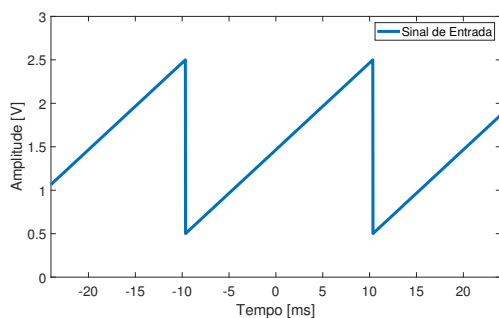
(a) Sinal de entrada onda triangular.



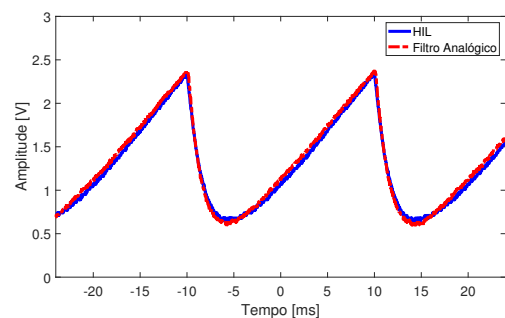
(b) Respostas do HIL e do filtro analógico.

Fonte: Do autor.

Figura 30 – Respostas do sistema HIL e do filtro analógico, para entrada em formato de onda dente de serra com $2 V_{pp}$, *offset* de 1,5 V e frequência de 50 Hzz.



(a) Sinal de entrada onda dente de serra.



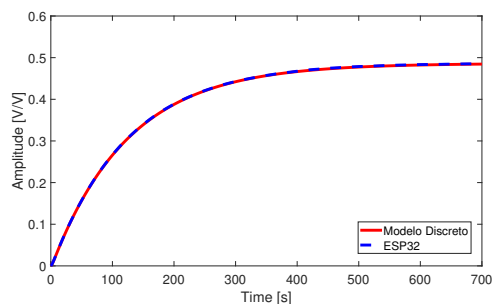
(b) Respostas do HIL e do filtro analógico.

Fonte: Do autor.

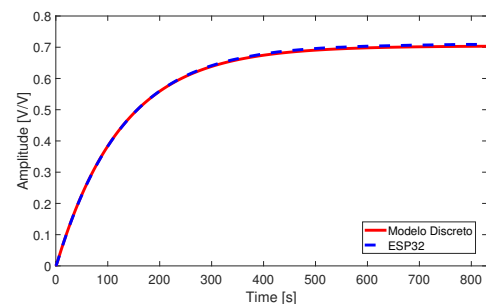
4.2 Emulação de um tanque industrial

Inicialmente, foram comparados os resultados obtidos a partir das expressões matemáticas representadas no domínio z utilizando o *software* Matlab, aos resultados obtidos pelo ESP32 embarcado com as equações de diferenças antes da reconstrução dos dados pelo DAC. Os resultados dessa comparação podem ser observado na Figura 31.

Figura 31 – Comparação entre os sinais obtidos pelas equações no domínio z utilizando o Matlab, e os sinais obtidos pela plataforma HIL embarcada com as equações de diferenças, para entrada em degrau.



(a) Abertura de 50%.



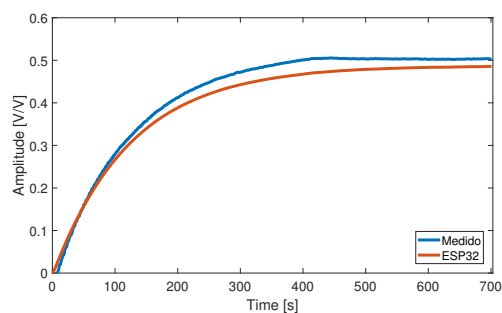
(b) Abertura de 75%.

Fonte: Do autor.

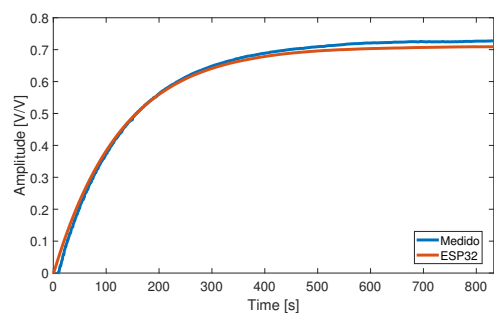
Nota-se uma pequena diferença entre as respostas. Esse erro pode estar relacionado a redução da precisão dos cálculos, uma vez que o Matlab trabalha com variáveis do tipo *double* (64 bits), ao passo que o ESP32-WROOM32 utiliza variáveis do tipo *float* (32 bits).

Na sequência, repetiu-se o ensaio ao degrau comparando as respostas obtidas pelo HIL antes da reconstrução dos dados pelo DAC, às respostas obtidas por Lage (2019) na planta real. A Figura 32 mostra os resultados da comparação.

Figura 32 – Comparação entre os sinais obtidos por Lage (2019), e os sinais obtidos pela plataforma HIL, para entrada em degrau.



(a) Abertura de 50%.



(b) Abertura de 75%.

Fonte: Do autor.

Os erros apresentados nas respostas mostradas pelas figuras acima, podem ser justificados em função das simplificações e manipulações algébricas realizadas para obtenção

do modelo caixa-preta que, possivelmente, não contempla todos os efeitos associados ao processo físico que se deseja representar. A Tabela 2 apresenta a raiz do erro médio quadrático (RMSE, do inglês *Root Mean Square Error*) das respostas obtidas.

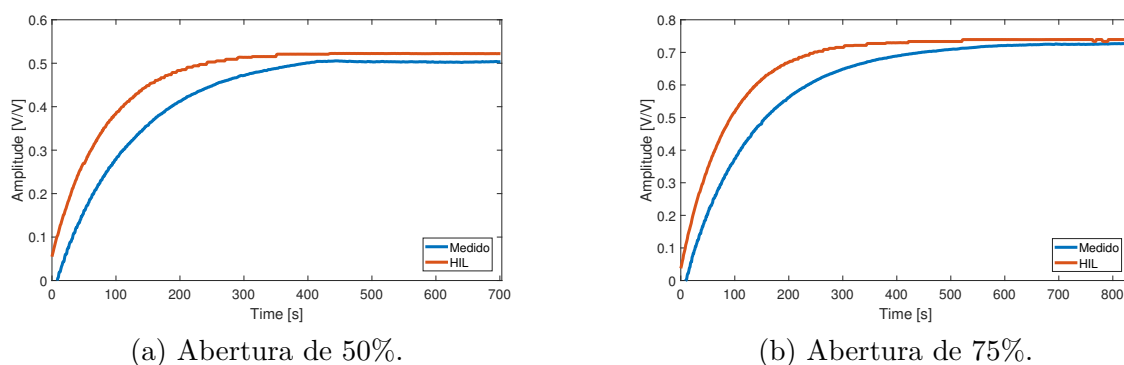
Tabela 2 – RMSE percentual calculado para os dois pontos de operação a partir da diferença entre: valores medidos por Lage (2019) aos valores obtidos pelo HIL antes da reconstrução pelo DAC; e os valores simulados pelo Matlab utilizando o modelo identificado, aos valores obtidos pelo HIL antes da reconstrução pelo DAC.

	Ponto de operação	Valor interno
Valores Medidos	50%	2,26 %
Modelo Identificado		0,06 %
Valores Medidos	75%	1,63 %
Modelo Identificado		0,51 %

Fonte: Do autor.

Na sequência, foram comparadas as respostas obtidas a partir do sistema HIL após a reconstrução dos dados pelo DAC, aos valores medidos por Lage (2019), aplicando-se na entrada do ADC1 valores de tensão proporcionais aos valor utilizados pela referência para ajustar a porcentagem de abertura das válvulas pneumáticas. Nesta etapa, os valores percentuais de referência foram convertidos em valores de tensão entre 0 V e 1 V. Esses valores de tensão foram utilizados para configurar o gerador de sinais de forma remota, por meio de uma aplicação desenvolvida em Matlab. Os resultados obtidos são apresentados na Figura 33.

Figura 33 – Comparação entre os sinais obtidos por Lage (2019), e os sinais obtidos pela plataforma HIL, para uma entrada proporcional ao sinal de entrada utilizado por Lage (2019).

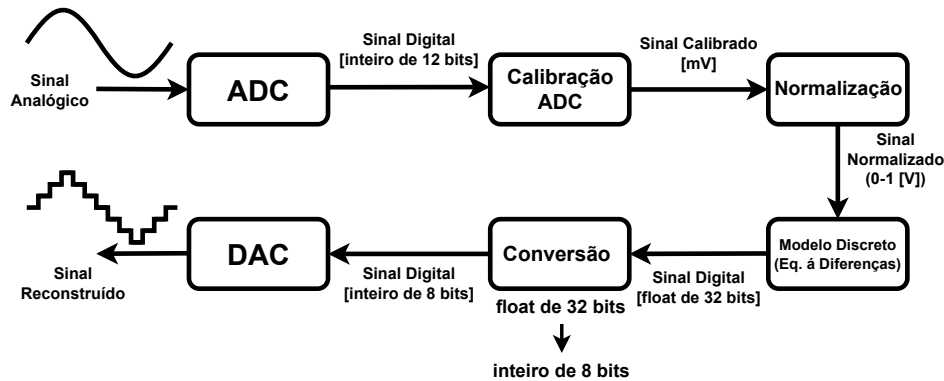


Fonte: Do autor.

Observa-se que há uma diferença entre os resultados obtidos pelo HIL e os resultados obtidos experimentalmente, possivelmente relacionada ao modelo aproximado utilizado para os testes e perdas de dados nos processos de transferência. O fluxograma

apresentado na Figura 34 mostra as conversões dos dados a partir do processo de amostragem.

Figura 34 – Fluxograma da conversão de dados no sistema.



Fonte: Do autor.

No sistema utilizado, são realizadas diversas etapas de conversão de dados, ocasionando erros na transferência dos dados. Primeiramente, pode-se considerar o erro associado ao processo de quantização e de linearidade do ADC. Esses efeitos podem ter sido agravados pelos erros causados a partir da conversão dos valores percentuais de referência (utilizados no trabalho de Lage (2019)) em tensão, aplicada pelo gerador de sinais. Finalmente, há ainda erros causados pelos efeitos da quantização dos sinais reconstruídos pelo DAC, que apresenta uma redução de precisão causada pela redução do número de bits deste dispositivo (8 bits) em relação ADC (12 bits). A Tabela 3 apresenta os valores de RMSE para cada comparação realizada descrita nesta seção.

Tabela 3 – RMSE percentual calculado para os dois pontos de operação a partir das diferenças entre: valores medidos por Lage (2019) aos valores obtidos pelo HIL após a reconstrução pelo DAC; valores simulados pelo Matlab utilizando o modelo identificado, aos valores obtidos pelo HIL após a reconstrução pelo DAC; e os valores internos do microcontrolador antes da reconstrução pelo DAC, aos valores obtidos pelo HIL após a reconstrução pelo DAC.

	Ponto de operação	Valores de saída do HIL
Valores Medidos	50%	6,02 %
Modelo Identificado		7,51 %
Valores internos ESP32		7,46 %
Valores Medidos	75%	7,89 %
Modelo Identificado		7,87 %
Valores internos ESP32		7,48 %

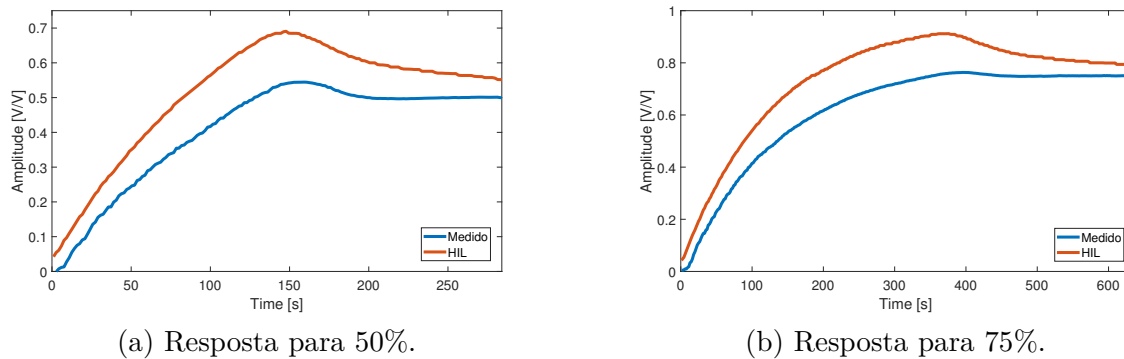
Fonte: Do autor.

O teste final realizado na plataforma HIL simula o controle do nível do tanque por meio do ajuste do percentual de abertura da válvula pneumática. Esse ensaio foi realizado utilizando os mesmos sinais de controle apresentados por Lage (2019) no sistema real. No

trabalho, o autor utiliza um controlador PI, com constante proporcional $K_P = 75$, e constante integral $K_I = 5$.

Utilizando-se os mesmos valores de abertura da válvula de vazão medidos por Lage (2019) como entrada do HIL, obteve-se a resposta do sistema apresentada na Figura 35.

Figura 35 – Comparação das respostas do sistema HIL, com as medidas na planta real, utilizando os mesmos sinais de controle de abertura da válvula pneumática medidos por Lage (2019) no sistema real.



Fonte: Do autor.

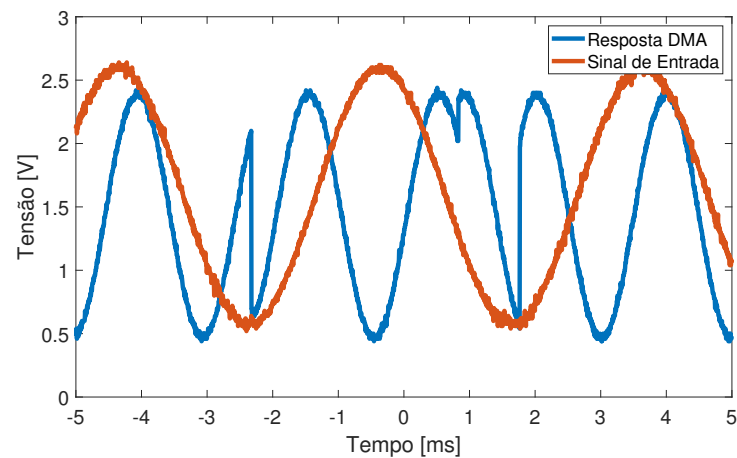
As diferenças obtidas nos dois casos são significativas, e estão possivelmente relacionadas a diferenças entre os modelos obtidos e o comportamento do sistema real. Esses resultados reforçam a importância da qualidade do modelo embarcado em sistemas HIL, uma vez que o resultado emulado depende diretamente dessa representação.

4.3 Utilização do DMA no ESP32

A fim de aumentar a frequência de amostragem do ESP32, utilizou-se o DMA (do inglês, *Direct Memory Access*) por meio da comunicação I2S, em que realizou-se o teste de verificação do recurso, em que um sinal de entrada senoidal foi amostrado pelo ADC do microcontrolador, e o intuito foi o de reconstruir o sinal por meio do DAC utilizando o DMA. A Figura 36 mostra o resultado.

Nota-se pela figura que existem lacunas entre o sinal reconstruído e o sinal de entrada. Essas lacunas são devidas a configuração do *buffer* de dados de 1024 posições. Este é utilizado para o processo de aquisição e reconstrução dos dados, em que, os dados adquiridos não são processados enquanto uma nova aquisição está sendo realizada. Além disso, não foi possível reduzir o tamanho do *buffer*, pois o número mínimo de posições a ser configurado é de 1024 posições.

Figura 36 – Sinal de entrada senoidal e resposta do HIL utilizando DMA por meio da comunicação I2S.



Fonte: Do autor.

Os resultados obtidos mostram que essa configuração não atende aos requisitos de um sistema HIL, haja vista que as restrições associadas ao modo de operação do sistema impedem que uma amostra seja processada por vez.

4.4 Considerações Parciais

Neste capítulo, foram apresentados os principais resultados obtidos ao longo do desenvolvimento deste trabalho. Para os testes iniciais de validação do HIL, utilizou-se a planta didática SMAR PD3-F como o sistema a ser emulado no ESP32. Os dados coletados por Lage (2019), nível do tanque e abertura da válvula de vazão, foram os sinais de entrada e saída, respectivamente. Primeiramente, foi realizada a análise dos parâmetros de interesse medidos e os modelos matemáticos desenvolvidos pelo autor, de modo que se definiram dois pontos de operação a serem usados, sendo a abertura da válvula em 50% e 75%.

Após, foi realizada a identificação do sistema a partir dos sinais de entrada e saída medidos. As respostas obtidas pelo modelo identificado, os valores medidos por Lage (2019), os sinais internos e a saída do HIL foram analisadas e comparadas. Com isso, validou-se o ESP32 como sistema HIL para emular a planta didática SMAR PD3-F.

Na sequência, realizou-se o teste de controle em malha aberta no HIL, comparando os resultados obtidos pelo microcontrolador com os obtidos na planta real. As respostas obtidas pelo *hardware* apresentaram erros significativos causados pela modelagem realizada.

A fim de validar a capacidade de processamento do ESP32, embarcou-se o modelo de um Filtro RC passa-baixas, e comparou-se a resposta do HIL com a do filtro analógico em tempo real. Os resultados obtidos mostraram que o microcontrolador foi capaz de

emular o sistema real dentro das especificações definidas, validando a plataforma quanto a capacidade de processamento. Entretanto, a medida que a frequência do sinal de entrada se aproxima de 10% da frequência de amostragem, a quantidade de amostras reconstruídas se torna menor, destacando os efeitos da quantização e piorando a qualidade do sinal representado no domínio do tempo.

Para verificar se o microcontrolador consegue emular sistemas com sinais de entrada não senoidais, embarcou-se o modelo de um Filtro RC passa-baixas com frequência de corte cem vezes menor que a frequência de amostragem, e comparou-se a resposta do HIL com a do filtro analógico em tempo real, utilizando para os sinais de entrada as ondas quadrada, triangular e dente de serra. Os resultados obtidos mostraram que o microcontrolador foi capaz de emular o sistema real dentro das especificações definidas, validando a plataforma quanto a capacidade de operar com entradas não senoidais.

Por fim, foi realizado o teste utilizando DMA por meio da comunicação I2S, em que foi possível observar a inviabilidade da utilização do recurso. O resultado mostrou que existem lacunas entre o sinal reconstruído e o sinal de entrada. Essas lacunas são devidas a configuração do *buffer* de dados, utilizado para o processo de aquisição e reconstrução dos dados, em que, os dados adquiridos não são processados enquanto uma nova aquisição está sendo realizada. Além disso, não foi possível reduzir o tamanho do *buffer*, pois o número mínimo de posições a ser configurado é de 1024 posições.

5 Considerações finais

Este trabalho apresentou as etapas de desenvolvimento e validação de um sistema HIL para emulação de processos industriais. Inicialmente, foi realizada a descrição do contexto do projeto e a fundamentação teórica. Consequente, realizou-se a caracterização do ESP32, dispositivo que foi utilizado como HIL.

Quanto á frequência de amostragem, definiu-se o valor máximo de 10kHz, e avaliou-se as respostas para amostragem e reconstrução de sinais arbitrários com frequências distintas e tempo de processamento de um filtro média móvel.

Após a caracterização, o HIL foi validado utilizando como sistema a ser emulado, a planta didática SMAR P3D-F presente na instituição de ensino. Com base nos resultados obtidos, percebeu-se que o tratamento de dados pelo MatLab e as conversões no HIL, foram fatores significativos para os valores de erro encontrados. O primeiro fator é proporcionado devido o MatLab trabalhar com dados em 64 *bits*, enquanto que os dados coletados estão no formato de 32 *bits*. As diversas conversões de dados realizadas no HIL impactaram em um somatório de erros, evidenciado na resposta obtida. A emulação do sistema real sem o controle validou o ESP32 como HIL, porém, os erros computados nos testes com controle em malha aberta foram significativos, devidos ao modelo matemático desenvolvido.

Consequente, fez-se os testes para avaliar a capacidade de processamento do HIL por meio da emulação de um Filtro RC passa-baixa. Os resultados obtidos mostraram que o *hardware* é capaz de trabalhar com frequências próximas ao limite estipulado (1 kHz). Com base nos testes, validou-se a plataforma para emulação do sistema com frequências próximas à 7% da frequência de amostragem (10 kHz).

Para verificar se o microcontrolador consegue emular sistemas com sinais de entrada não senoidais, embarcou-se o modelo de um Filtro RC passa-baixas com frequência de corte cem vezes menor que a frequência de amostragem, e comparou-se a resposta do HIL com a do filtro analógico em tempo real, utilizando para os sinais de entrada as ondas quadrada, triangular e dente de serra. Os resultados obtidos mostraram que o microcontrolador foi capaz de emular o sistema real dentro das especificações definidas, validando a plataforma quanto a capacidade de operar com entradas não senoidais, para o sistema emulado.

Por fim, analisou-se o controle dos processos de aquisição e reconstrução de dados por DMA. Os resultados obtidos mostraram que essa configuração não atende aos requisitos de um sistema HIL, devido às restrições associadas ao modo de operação do sistema impedem que uma amostra seja processada por vez, ocasionando lacunas entre o sinal reconstruído e o sinal de entrada.

Conclui-se, portanto, que o ESP32 pode ser utilizado como *Hardware In The Loop*

para emulação de um processo industrial de baixa frequência, e para emulação de sistemas mais rápidos, próximos a 10% da frequência de amostragem definida (10 kHz). Contudo, evidencia-se que o modelo matemático desenvolvido para a planta diática P3D-F ocasionou erros significativos nos testes de controle em malha aberta.

5.1 Sugestões Para Trabalhos Futuros

Como sugestões para desenvolvimento de próximos trabalhos, pode-se citar:

- Utilização de sistemas RTOS para gerenciamento de tarefas e periféricos;
- Projeto e validação de emuladores para outros tipos de aplicações de controle, tais como sistemas de controle de velocidade e posição;
- Avaliação dos sistemas emulados em malha fechada, considerando sinais de controle gerados em tempo real por outras plataformas de processamento;
- Desenvolver aplicações que possam ser executadas de forma paralela, permitindo que os dois núcleos de processamento do microcontrolador possam ser utilizados;
- Utilização de outras plataformas de processamento de sinais, capazes de controlar ADC e DAC por meio de interrupções e DMA;

Referências

- BASTOS, R. F. et al. Model, design and implementation of a low-cost hil for power converter and microgrid emulation using dsp. *IET Electric Power Applications*, Institution of Engineering and Technology, v. 12, p. 9, 9 2019. ISSN 1755-4535.
- BASTOS, R. F. et al. Low-cost hardware-in-the-loop for real-time simulation of electric machines and electric drive. *IET Electric Power Applications*, Institution of Engineering and Technology, v. 14, p. 9, 9 2020. ISSN 17518679.
- BERNARDES, A. et al. Low-cost hardware-in-the-loop platform for embedded control strategies simulation. *IEEE Access*, Institute of Electrical and Electronics Engineers Inc., v. 7, p. 14, 2019. ISSN 21693536.
- BOYCE, W. E.; DIPRIMA, R. C. *Equações Diferenciais Elementares e Problemas de Valores de Contorno*. 10. ed.: Grupo GEN, 2015. 1040 p. ISBN 978-85-216-2832-3.
- CARRARA, V. *ANALISE E CONTROLE DE SISTEMAS LINEARES*. São José dos Campos, SP/BR, 2012.
- ESPRESSIF. *Datasheet ESP32 Series*. Shanghai, CN, 2023.
- ESPRESSIF, S. *ESP32-DevKitC*. 2015–2023. Disponível em: <<https://www.espressif.com/en/products/devkits/esp32-devkitc>>.
- FLOYD, T. L. *Sistemas Digitais, Fundamentos e Aplicações*. 9. ed.: Bookman, 2007. 888 p. ISBN 978-85-7780-107-7.
- FRANKLIN, G. F.; POWELL, J. D.; EMAMI-NAEINI, A. *Sistemas de Controle para Engenharia*. 6. ed.: Bookman, 2013. 721 p. ISBN 9780136019695.
- GOMEZ, M. *Hardware-in-the-Loop Simulation*. 2001. Disponível em: <http://jmargolin.com/uavs/jm_rpv2_npl_14.pdf>.
- HANSELMANN, H. *Hardware-in-the-Loop Simulation as a Standard Approach for Development, Customization, and Production Test*: SAE Technical Paper 930207, 1993. 13 p.
- ISERMANN, R. et al. *Hardware-in-the-loop simulation for the design and testing of engine-control systems*: Elsevier Science, 1999.
- KOPETZ, H. *Real-Time Systems, Design Principles for Distributed Embedded Applications*: Kluwer Academic Publishers, 1997. 353 p. ISBN 0-792-39894-7.
- LAGE, R. L. *Projeto de um sistema de controle de nível para tanques industriais*. 57 f. Monografia (Graduação) — Departamento de Engenharia Elétrica, ICEA, Universidade Federal de Ouro Preto, João Monlevade, 2019.
- LAPLANTE, P. A.; OVASKA, S. J. *Real-Time Systems Design and Analysis*. 4. ed.: John Wiley Sons, Inc., Hoboken, 2012. 584 p. ISBN 9781118136577.

MIHALIČ, F.; TRUNTIČ, M.; HREN, A. Hardware-in-the-loop simulations: A historical overview of engineering challenges. *Electronics (Switzerland)*, MDPI, v. 11, 8 2022. ISSN 20799292.

OPPENHEIM, A. V.; SCHAFER, R. W. *Processamento em Tempo Discreto de Sinais*. 3. ed.: Grupo Pearson, 2013. 888 p. ISBN 978-85-8143-102-4.

OPPENHEIM, A. V.; WILLSKY, A. S. *Sinais e Sistemas*. 2. ed.: Pearson Education, 2010. 592 p. ISBN 978-85-4301-380-0.

PENIDO Édilus C. C.; TRINDADE, R. S. *Microcontroladores*. Ouro Preto, MG/BR, 2013.

SYSTEMS, E. *ESP32 Technical Reference Manual*. Shanghai, CN, 2023.

TEJADA, Z. *Processamento em tempo real*. 2023. Disponível em: <<https://learn.microsoft.com/pt-br/azure/architecture/data-guide/big-data/real-time-processing>>.

VOJTESEK, J.; DOSTAL, P.; MASLAN, M. Modelling and simulation of water tank. In: . Zlin, Czech Republic: European Conference on Modelling and Simulation, 2014. p. 7.



TERMO DE RESPONSABILIDADE

O texto do trabalho de conclusão de curso intitulado “Desenvolvimento de uma plataforma *Hardware in the loop* para simulação de sistemas em tempo real” é de minha inteira responsabilidade. Declaro que não há utilização indevida de texto, material fotográfico ou qualquer outro material pertencente a terceiros sem a devida citação ou consentimento dos referidos autores.

João Monlevade, 21 de Fevereiro de 2024 .

Hermes Henrique Silva Melo