

UNIDADE REMOTA DE IO'S MODPIC

Nome: Leonardo Shiba

Faculdade de Tecnologia “Adib Moisés Dib” de São Bernardo do Campo

leonardoshib@gmail.com

Orientador: Prof. Dr. Wellington Batista de Sousa

Faculdade de Tecnologia “Adib Moisés Dib” de São Bernardo do Campo

wellington.sousa@cps.sp.gov.br

Orientador: Prof. Me. Pedro Adolfo Gallani

Faculdade de Tecnologia “Adib Moisés Dib” de São Bernardo do Campo

pedro.gallani@cps.sp.gov.br

RESUMO

Este artigo apresenta o desenvolvimento de uma unidade remota de entradas e saídas (E/S) chamada MODPIC, voltada para o contexto da automação industrial e ensino técnico. O problema identificado reside no elevado custo de aquisição e manutenção de Controladores Lógicos Programáveis (CLPs) industriais, o que limita o acesso prático em ambientes acadêmicos com orçamentos restritos. O objetivo principal foi desenvolver uma alternativa economicamente viável utilizando um microcontrolador PIC e o protocolo Modbus. A pesquisa adotou a abordagem metodológica *Design Science Research* (DSR), focada na criação e avaliação de artefatos tecnológicos para solução de problemas práticos. O artefato desenvolvido consiste em uma placa com oito entradas e saídas digitais e analógicas, integrando interfaces RS485 e USB. Os resultados obtidos nos testes de integração com o *software* Codesys demonstraram a funcionalidade do dispositivo na simulação de controle de processos reais, permitindo inferir que o MODPIC é uma ferramenta didática eficaz, alinhada à realidade industrial e capaz de ampliar as estações de trabalho em laboratórios de automação.

Palavras-chave: Automação Industrial. CLP. Modbus. Microcontroladores. SoftPLC.

ABSTRACT

This article presents the development of a remote input and output (I/O) unit called MODPIC, aimed at the context of industrial automation and technical education. The identified problem resides in the high cost of acquisition and maintenance of industrial Programmable Logic Controllers (PLCs), which limits practical access in academic environments with restricted budgets. The main objective was to develop an economically viable alternative using a PIC microcontroller and the Modbus protocol. The research adopted the Design Science Research (DSR) methodological approach, focused on the creation and evaluation of technological artifacts for the solution of practical problems. The developed artifact consists of a board with eight digital and analog inputs and outputs, integrating RS485 and USB interfaces. The results obtained in the integration tests with the Codesys software demonstrated the functionality of the device in the simulation of real process control, allowing to infer that MODPIC is an effective didactic tool, aligned with industrial reality and capable of expanding the workstations in automation laboratories.

Keywords: Industrial Automation. PLC. Modbus. Microcontrollers. SoftPLC.

1. INTRODUÇÃO

O controle e a automação de processos são amplamente utilizados em diversos setores, desde a indústria de grande porte até aplicações prediais e residenciais, sendo valorizado por reduzir a intervenção humana, aumentar a produtividade e garantir a exatidão dos resultados.

Atualmente, a grande maioria das automações utiliza o CLP (Controlador Lógico Programável) como principal responsável pelo processamento e gerenciamento das entradas e saídas (I/O) dos processos. Apesar dos benefícios tecnológicos, a alta complexidade de fabricação dos CLPs industriais resulta em um alto custo de aquisição e manutenção.

Esse fator representa o principal problema de pesquisa deste trabalho, pois cria uma barreira significativa em ambientes acadêmicos e de ensino técnico que possuem orçamentos restritos.

Consequentemente, o alto custo limita a disponibilidade de equipamentos nas instituições e reduz o tempo de prática individual dos estudantes nas disciplinas de Automação Industrial.

Para solucionar essa questão, uma alternativa econômica e tecnicamente viável é a adoção de sistemas baseados em *SoftPLC*, que substituem o hardware dedicado do CLP por um software executado em um computador padrão.

Nesse cenário, o *software* (como o Codesys) assume a lógica de controle e interage com os sinais de campo por meio de uma unidade remota de entradas e saídas (E/S) utilizando um protocolo de comunicação.

Diante desse contexto, o objetivo principal deste trabalho é desenvolver uma unidade remota de E/S, denominada MODPIC, utilizando um microcontrolador PIC. Para atingir esse objetivo, a pesquisa adota a abordagem metodológica *Design Science Research* (DSR), voltada para a criação e avaliação de artefatos tecnológicos visando a solução de problemas práticos no contexto real.

O artefato desenvolvido consistirá em uma placa com oito entradas e saídas digitais, oito entradas e quatro saídas analógicas, comunicando-se com o *SoftPLC* através do protocolo industrial Modbus pelo meio físico USB e RS485.

O projeto justifica-se pela sua relevância acadêmica e prática ao viabilizar e democratizar o acesso à tecnologia de controle. O desenvolvimento desta unidade remota didática cria uma alternativa financeiramente compatível com a realidade das instituições de ensino, permitindo a expansão do número de estações de trabalho nos laboratórios.

Dessa forma, o trabalho contribui ativamente para a melhoria do processo de ensino-aprendizagem, preservando a experiência de programação e a conexão com sinais típicos do ambiente industrial moderno

2. Fundamentação Teórica

Neste capítulo são abordados os tópicos fundamentais para a compreensão da construção e funcionamento do dispositivo desenvolvido.

2.1 Controladores Lógicos Programáveis

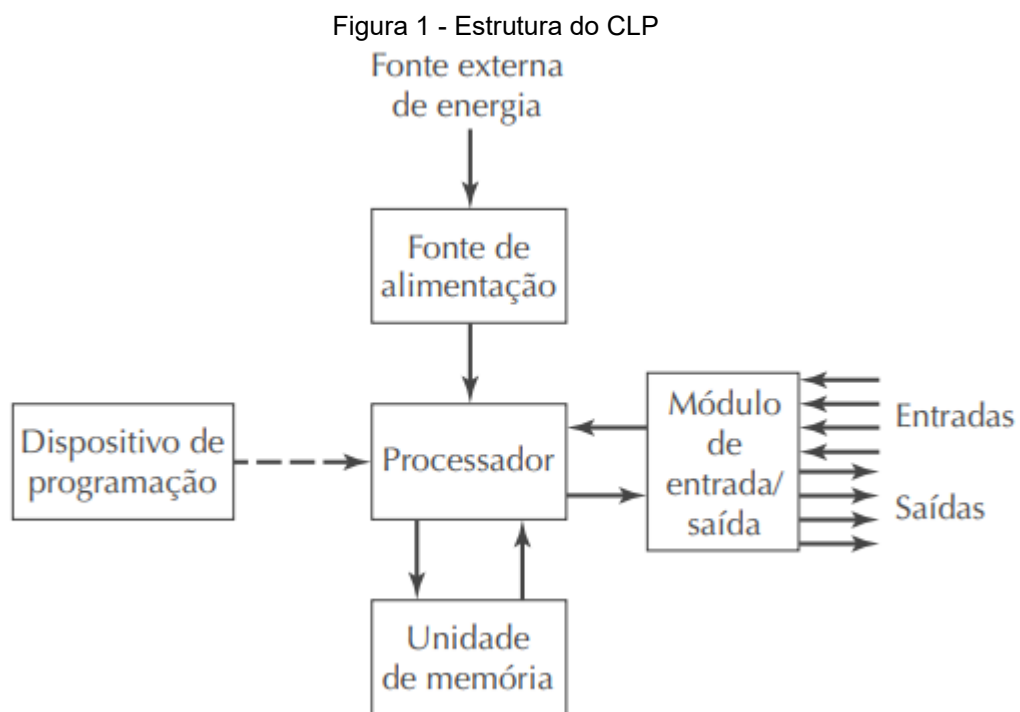
Os Controladores Lógicos Programáveis (CLPs) são dispositivos eletrônicos fundamentais na automação industrial, amplamente empregados devido à sua versatilidade, robustez contra ambientes inóspitos e capacidade de controlar uma vasta gama de processos.

Apoiado na norma IEC 61131-1, Prudente (2020) define o CLP como um equipamento dotado de componentes eletrônicos e memória, cuja função é armazenar

dados e programas para ler e executar instruções, interagindo e controlando sistemas físicos por meio de dispositivos de entrada e saída que operam com sinais digitais ou analógicos.

A introdução dessa tecnologia revolucionou a indústria ao substituir os painéis de relés, trazendo vantagens expressivas destacadas por Groover (2011), como a otimização do espaço físico e da quantidade de fiação, maior confiabilidade operacional, facilidade de manutenção, conectividade com outros dispositivos da planta e grande flexibilidade de programação.

O autor também descreve que para realizar todas essas tarefas, a arquitetura básica de um CLP é estruturada em três componentes fundamentais: os módulos de entrada e saída (I/O), a unidade de memória e a unidade central de processamento. A estrutura funcional do CLP é exemplificada na figura 1.



Fonte: Groover (2011)

Os módulos de I/O são os responsáveis por realizar o interfaceamento direto com os sensores e atuadores no campo, operando com sinais digitais (dois estados, como 0 e 24 V) e analógicos (variação contínua, como 0 a 10 V ou 4 a 20 mA).

As informações transitadas por esses módulos interagem com a unidade de memória, que armazena as configurações, o programa do usuário e a "memória

imagem" das entradas e saídas, enquanto o gerenciamento dinâmico de todo esse conjunto fica a cargo do Processador, responsável por executar o chamado ciclo de *Scan*.

Para complementar essa estrutura de funcionamento contínuo, os CLPs atuais frequentemente contam também com módulos de comunicação, o que permite a troca autônoma de dados em rede com outros sistemas industriais.

2.2 Protocolo ModBus

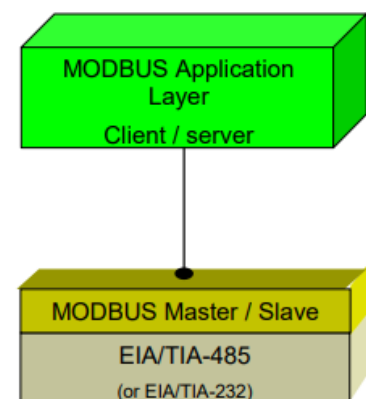
O Modbus, criado pela empresa Modicon, é reconhecido como um dos protocolos de comunicação aberta mais tradicionais e amplamente utilizados no ambiente industrial, criado inicialmente para integrar os CLPs fabricados pela própria empresa.

Operando na camada de aplicação (Nível 7) do modelo OSI, ele viabiliza a comunicação cliente/servidor entre equipamentos distribuídos em diferentes redes por meio de um sistema de requisição e resposta orientado por códigos de função (MODBUS-IDA, 2007).

Para estruturar essa comunicação, a especificação sobre linha serial atua de forma complementar nas camadas Física (Nível 1) e de Enlace de Dados (Nível 2) do modelo OSI. A figura 2 descreve o protocolo Modbus e suas respectivas camadas.

Figura 2 - Camadas do Protocolo Modbus

Layer	ISO/OSI Model	
7	Application	MODBUS Application Protocol
6	Presentation	Empty
5	Session	Empty
4	Transport	Empty
3	Network	Empty
2	Data Link	MODBUS Serial Line Protocol
1	Physical	EIA/TIA-485 (or EIA/TIA-232)



Fonte: Modbus-IDA (2007)

Em sua variação para linhas seriais (*Modbus over serial line*), o protocolo adota o princípio Mestre-Escravo. De acordo com o esse modelo, o Mestre (cliente) é o único dispositivo com permissão para iniciar uma transação enviando requisições, enquanto

os Escravos (servidores) apenas respondem executando a ação exigida ou devolvendo os dados solicitados.

As requisições do Mestre podem ocorrer de forma direcionada a um escravo específico (*Unicast*), que responde ao comando, ou de maneira global (*Broadcast*), na qual todos os escravos recebem a mensagem simultaneamente sem emitir resposta, respeitando uma faixa de endereçamento individual que vai de 1 a 247 (MODBUS-IDA, 2007).

Em termos físicos, a rede serial geralmente exige a implementação do padrão RS485, embora o padrão RS232 seja aceito opcionalmente para ligações ponto-a-ponto. As velocidades de sinalização de dados possuem suporte obrigatório a 9600 bps e 19200 bps, sendo esta última a taxa padrão exigida (MODBUS-IDA, 2007).

Em relação à transmissão serial, o protocolo impõe a implementação obrigatória do modo Modbus RTU (em detrimento do modo ASCII), justificado por oferecer maior densidade de caracteres e, conseqüentemente, melhor eficiência no fluxo de dados para uma mesma velocidade (MODBUS-IDA, 2007).

No modo RTU, os dados são transmitidos em formato binário utilizando 8 bits, compondo uma sequência de 11 bits por caractere (considerando o bit de início, os 8 bits de dados e os bits de parada, além da paridade par, que é um requisito padrão).

A estrutura da mensagem é constituída pelos campos de endereço do alvo (1 byte), código da função (1 byte), dados de mensagem (variável) e a verificação de erro CRC (2 bytes). A Tabela 1 descreve a estrutura de uma mensagem Modbus.

Tabela 1 - Formato do Frame Modbus

Campo	Tamanho (Bytes)	Descrição
Endereço do Escravo	1	Endereço do dispositivo alvo (1–247)
Código de Função	1	Ação a ser executada
Mensagem	Variável	Parâmetros ou dados de resposta
CRC	2	Verificação de erro (<i>Cyclical Redundancy Checking</i>)

Fonte: autoria própria

A mensagem pode possuir um tamanho máximo de 256 bytes de comprimento. O enquadramento dessa transmissão (*framing*) é controlado por intervalos de silêncio na linha serial.

O início e o fim de uma mensagem são delimitados por um silêncio mínimo equivalente a 3,5 tempos de caractere, e o fluxo de dados de uma requisição deve ser contínuo; caso ocorra uma pausa superior a 1,5 tempo de caractere entre os dados, a mensagem é considerada incompleta e acaba sendo descartada.

Por fim, a integridade de todo o pacote RTU é assegurada pelo cálculo de redundância cíclica (CRC) de 16 bits. Se o dispositivo receptor identificar qualquer falha de paridade ou de CRC durante a transmissão, nenhuma resposta é retornada, deixando a cargo do dispositivo mestre a gestão do tempo de esgotamento (*timeout*) da requisição (MODBUS-IDA, 2007).

2.3 Microcontroladores

O microcontrolador é uma peça fundamental da eletrônica moderna para execução de automações. Diferente dos microprocessadores, que exigem componentes externos como memória e periféricos para funcionar, os microcontroladores integram em seus *chips* a CPU, a memória de programa para armazenar o *firmware* e a memória de dados para variáveis, além de diversos periféricos de Entrada/Saída cruciais, como Conversores Analógico-Digitais, Temporizadores e interfaces de comunicação serial (MIYADAIRA, 2009). Essa alta integração garante que o dispositivo seja compacto, de baixo custo e com consumo energético eficiente.

O microcontrolador é programado, usualmente em linguagem C ou *assembly*, para executar uma tarefa específica e repetitiva, sendo fundamental no controle de automação industrial, eletrodomésticos, dispositivos médicos, sistemas automotivos e todo o universo da Internet das Coisas, onde atua como o cérebro que recebe dados de sensores e comanda atuadores em tempo real.

No contexto de automação, a família de microcontroladores PIC (*Peripheral Interface Controller*), fabricada pela Microchip, consolidou-se como uma das mais populares do mercado por focar em soluções de baixo custo e alta confiabilidade.

Os PICs são projetados seguindo uma Arquitetura RISC (*Reduced Instruction Set Computer*), o que significa que eles operam com um conjunto reduzido de instruções que são executadas rapidamente.

A vasta documentação, o extenso ecossistema de suporte e a diversidade de famílias - desde os modelos de 8 bits para tarefas simples e de baixo consumo (PIC16), até os mais robustos de 32 bits (PIC32) para aplicações complexas como

multimídia e IoT (IBM, 2024) - reforçam a posição do PIC como um componente estratégico em projetos que demandam um equilíbrio ideal entre custo-benefício, facilidade de programação e controle preciso.

O microcontrolador PIC utilizado para este projeto é o PIC18F4550. A escolha desse modelo se deu pelos diversos recursos oferecidos, dentre eles: Barramento de comunicação USART, Barramento de comunicação I²C, porta de comunicação USB 2.0 com regulador de tensão interno, frequência interna ajustável para a comunicação USB, até 13 entradas analógicas com 10 bits de resolução, e entradas e saídas digitais com correntes de *Sink/Source* de até 25mA (Microchip Technology, 2009).

3. Metodologia

A definição de uma estruturação metodológica é uma etapa indispensável em pesquisas de teor científico e tecnológico, pois fornece a organização necessária para guiar o pesquisador na busca por soluções frente ao problema investigado (CAUCHICK, 2019).

Para orientar o planejamento e a execução deste projeto, a abordagem metodológica adotada foi a *Design Science Research* (DSR). A DSR caracteriza-se como um método de pesquisa voltado à criação, implementação e avaliação de artefatos inovadores projetados para solucionar problemas práticos e aprimorar as condições do ambiente real no qual serão aplicados.

O grande diferencial dessa abordagem reside em sua capacidade de integrar o embasamento teórico e científico com o desenvolvimento tecnológico prático, assegurando que o artefato construído atenda efetivamente às necessidades da demanda (BROCKE et al., 2020).

Para atingir esses resultados, a metodologia organiza o processo de pesquisa e desenvolvimento em seis fases progressivas e estruturadas. A seguir são apresentadas as etapas da metodologia e suas características.

A primeira fase da metodologia é responsável pela identificação do problema e justificativa do valor de uma solução proposta. Para a execução desta fase, foi realizada uma entrevista com um professor da disciplina de Controladores Lógicos Programáveis na Faculdade de Tecnologia de São Bernardo do Campo (Fatec-SBC), a fim de definir quais limitações foram encontradas no ensino da disciplina.

Na segunda fase, são definidos os objetivos que a solução idealizada deve alcançar, esclarecendo quais resultados o artefato proposto deve atingir. A partir dos

dados coletados na etapa inicial, foram elaborados os requisitos técnicos e funcionais que o protótipo deveria atender, servindo como diretriz para o desenvolvimento subsequente do *hardware* e do *firmware*.

A terceira fase, design e desenvolvimento, foca na construção do artefato para atender aos requisitos estipulados (BROCKE et al., 2020). O desenvolvimento eletrônico foi realizado no *software* KiCad, abrangendo o esquema e o *layout* da placa de circuito impresso. Paralelamente ao desenvolvimento, realizou-se um levantamento de custos a fim de projetar uma solução financeira e tecnicamente viável. O *firmware* foi desenvolvido no ambiente MPLAB X, visando a implementação da lógica de controle e do protocolo de comunicação no microcontrolador.

A quarta fase, de demonstração, tem o objetivo de comprovar a eficácia do artefato diante do problema identificado (BROCKE et al., 2020). Para a execução desta etapa, foram utilizados os *softwares* Docklight, AVEVA Edge e Codesys, cujas validações abrangeram desde a integridade da comunicação de dados até a interação física entre dispositivos de campo e *software*.

Na quinta fase, realiza-se a avaliação do artefato para mensurar o quanto ele satisfaz os objetivos propostos inicialmente. Esta etapa compreendeu uma análise técnica de estabilidade e integridade da comunicação, mas focou, sobretudo, na viabilidade econômica. A avaliação foi baseada em uma análise comparativa de custos entre o MODPIC e os CLPs industriais de entrada, buscando comprovar a redução do investimento necessário para a implementação de laboratórios didáticos sem prejuízo às funcionalidades lógicas essenciais.

Por fim, a sexta fase consiste na comunicação dos resultados. Esta etapa concretizou-se por meio da redação e estruturação deste artigo científico, visando difundir o conhecimento técnico gerado e os resultados da validação do MODPIC para a comunidade acadêmica e profissional da área de automação industrial.

4. Desenvolvimento do Projeto

O projeto consiste no desenvolvimento de uma unidade de interface remota, denominada MODPIC, equipada com 8 entradas e 8 saídas digitais, 8 entradas analógicas e 4 saídas analógicas. O dispositivo utiliza o microcontrolador PIC18F4550 e comunica-se através do protocolo Modbus RTU via interfaces USB e RS485.

Além da atuação local, o hardware possui a funcionalidade de *bridge* entre os barramentos USB e RS485, permitindo a interação de outros dispositivos no barramento RS485 com o *SoftPLC*.

4.1 Identificação do Problema e Justificativa

A motivação para o desenvolvimento deste trabalho nasceu da limitação de dispositivos para aulas práticas de Automação Industrial na Fatec-SBC, uma vez que o custo para aquisição de Controladores Lógicos Programáveis (CLPs) industriais restringe a quantidade de bancadas disponíveis, o que limita o tempo de uso individual dos equipamentos pelos alunos.

Outro fator identificado é que os computadores de laboratório, utilizados para executar softwares de programação, não dispõem de interfaces diretas com o campo, assim o MODPIC justifica-se como uma ferramenta para suprir essa carência de hardware por meio de uma solução que utiliza o processamento de computadores já existentes.

A seguir são destacados os pontos que fundamentam o desenvolvimento do artefato:

1. **Disponibilidade de Hardware:** ampliação do número de estações de trabalho para práticas de automação com um dispositivo de custo inferior ao de um CLP comercial;
2. **Segurança dos Equipamentos:** redução do risco de danos ao patrimônio da instituição, uma vez que o MODPIC utiliza componentes discretos e arquitetura conhecida, facilitando eventuais reparos;
3. **Conectividade:** prover o acesso ao meio físico RS485 para computadores através da porta USB. Essa funcionalidade permite que o dispositivo atue tanto como uma unidade remota escrava quanto como uma interface de comunicação para o intercâmbio de dados com outros equipamentos Modbus conectados ao barramento;
4. **Aplicação Acadêmica:** utilização de conceitos da grade curricular do curso (eletrônica, microcontroladores e redes locais) na construção de um dispositivo funcional, replicável e modificável por outros estudantes.

4.2 Definição de Requisitos

Nesta fase, foram estabelecidos os requisitos técnicos que o MODPIC deveria atender para cumprir os objetivos propostos. Estas especificações foram elaboradas com base nas necessidades identificadas na etapa inicial e serviram como diretriz para a construção do protótipo.

A seguir são apresentados os requisitos estabelecidos para o artefato:

- **Capacidade de I/O:** disponibilização de 28 pontos de sinal, sendo 08 entradas e 08 saídas digitais, 08 entradas analógicas com resolução de 10 bits e 04 saídas analógicas;
- **Viabilidade financeira:** o custo total dos componentes eletrônicos para a montagem do artefato deve ser significativamente inferior ao de uma unidade remota comercial equivalente;
- **Protocolo de Comunicação:** implementação do protocolo Modbus RTU para garantir a compatibilidade com *softwares* de supervisão e controle;
- **Interfaces Físicas:** inclusão de uma porta USB (emulação serial CDC) e uma porta RS485;
- **Configurabilidade:** possibilidade de ajuste do endereço Modbus e da velocidade de transmissão (*baud rate*);
- **Funcionalidade Bridge:** implementação de lógica para que, via conexão USB, o dispositivo identifique o endereço das requisições. Caso o endereço solicitado não seja o local, o *frame* deve ser encaminhado para o barramento RS485;
- **Operação via RS485:** capacidade de atuar como uma unidade remota escrava padrão através da porta RS485 na ausência de conexão USB ativa.

4.3 Desenvolvimento do Artefato

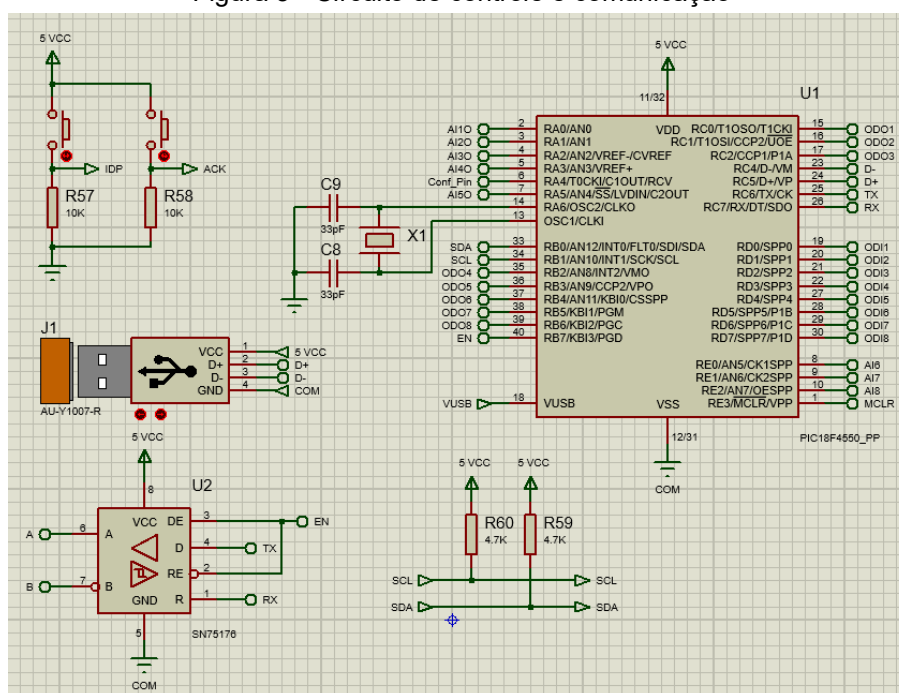
O desenvolvimento do MODPIC foi um processo dividido entre o desenvolvimento de *hardware* e a arquitetura de *software* embarcado (*firmware*). Cada etapa foi realizada de forma separada.

O artefato se iniciou na construção do hardware, seguindo um ciclo de validação para reduzir riscos e perdas financeiras em componentes. A unidade remota foi projetada se baseando na estrutura básica de um CLP, sendo constituída de um módulo de I/Os, módulo de comunicação (RS485 e USB), e o circuito de processamento.

A simulação no Proteus¹ foi o ambiente de decisão para a arquitetura do MODPIC. A facilidade de edição e agilidade no teste dos circuitos tornaram a ferramenta ideal para projetar e reestruturar o arranjo dos circuitos de entradas e saídas, em busca da solução mais simples e eficaz.

O Proteus serviu não apenas para validar os circuitos, mas também para criar o circuito de forma dinâmica, consolidando o design do artefato antes do início da montagem física e dos testes em bancada. A figura 3 apresenta o diagrama inicial do circuito de controle e comunicação da unidade remota.

Figura 3 - Circuito de controle e comunicação



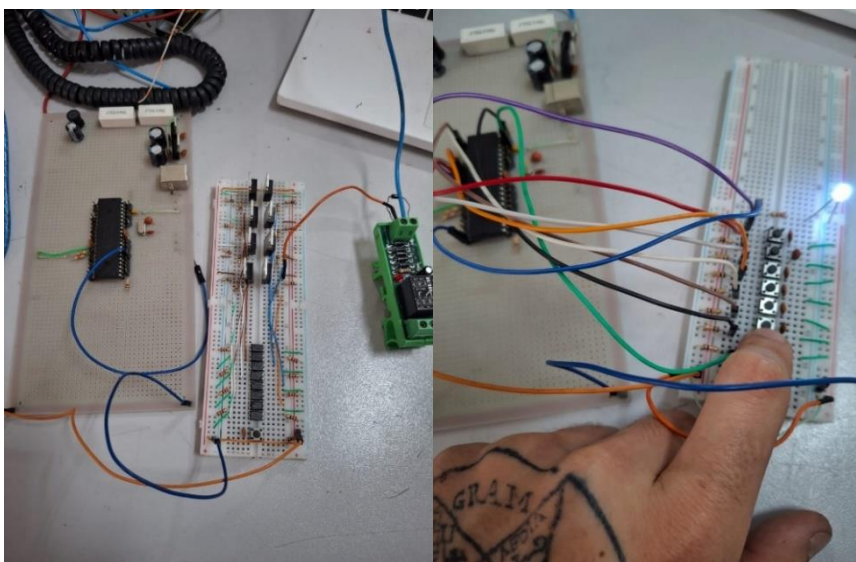
Fonte: Autoria Própria

Após a simulação dos circuitos, iniciaram-se os testes físicos. Foram testados individualmente os circuitos de entrada digital, saída digital, entrada analógica, e saída analógica para validar o funcionamento elétrico de cada periférico de forma separada.

¹ Desenvolvido pela Labcenter Electronics, o Proteus Design Suite é uma plataforma de software voltada ao desenvolvimento eletrônico virtual. É amplamente utilizado no ambiente acadêmico por permitir a validação lógica e funcional de circuitos integrados e microcontroladores antes da montagem física do protótipo, reduzindo custos com componentes e tempo de desenvolvimento.

Os testes de entradas e saídas digitais estão registrados na figura 4.

Figura 4 - Teste de entradas e saídas digitais



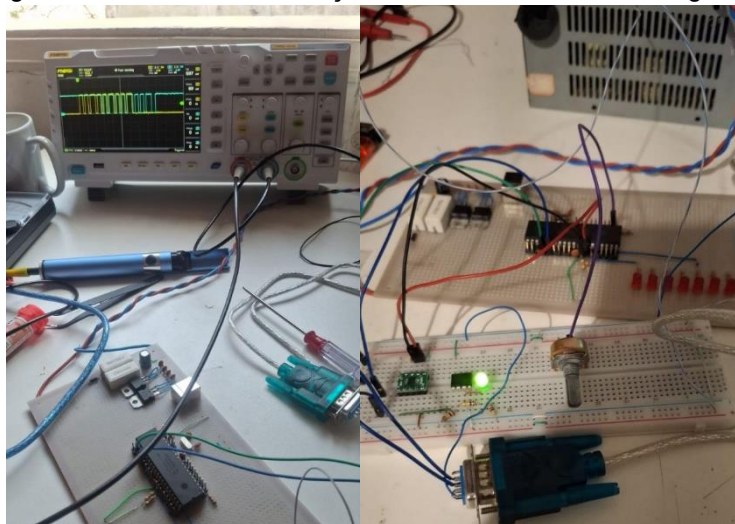
Fonte: Autoria própria

Após essa etapa, a montagem foi transferida para uma placa perfurada para integração com uma fonte de 24 V e testes da interface de comunicação USB.

Durante os experimentos, identificou-se que ruídos provenientes da linha de alimentação causavam falhas intermitentes na transmissão de dados. Para solucionar a instabilidade, foi projetado e implementado um estágio de filtragem no circuito de alimentação, utilizando capacitores de desacoplamento para assegurar a integridade dos sinais e o sincronismo do microcontrolador.

A figura 5 mostra os testes de comunicação no barramento UART e no barramento I²C, que foi utilizado para acionar as saídas analógicas.

Figura 5 - Teste de comunicação e entradas e saídas analógicas

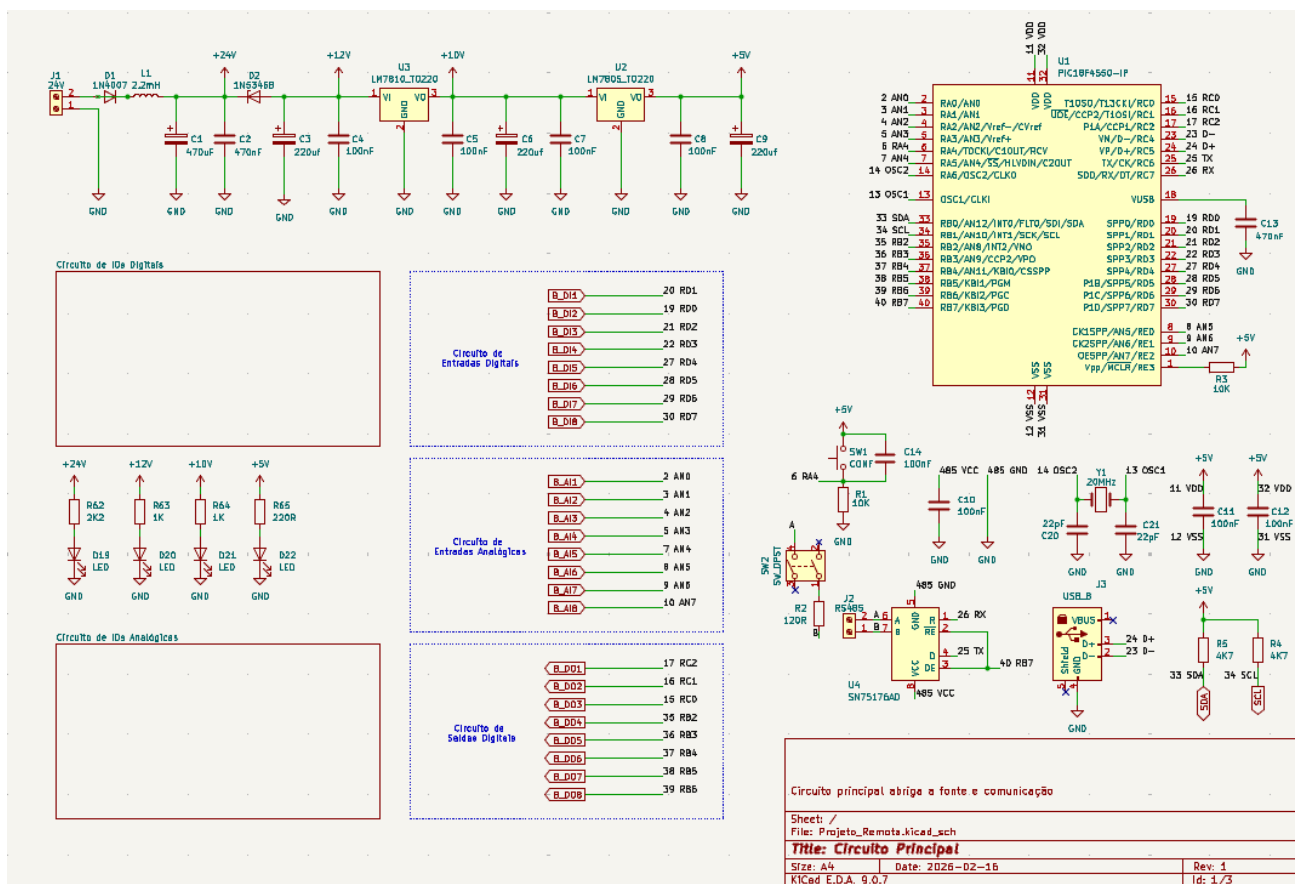


Fonte: Autoria própria

Realizada a validação elétrica e da comunicação concluída em bancada, o projeto foi finalizado no KiCad. Ao partir para o processo de desenvolvimento no KiCad 9.0, foi necessário refazer o projeto no novo *software*, contemplando as novas alterações providas dos testes com a protoboard.

A figura 6 mostra o diagrama elétrico do circuito de controle e comunicação da unidade remota, além de mostrar a fonte projetada para suprimir os ruídos na entrada de alimentação.

Figura 6 - Circuito adaptado ao KiCad 9.0



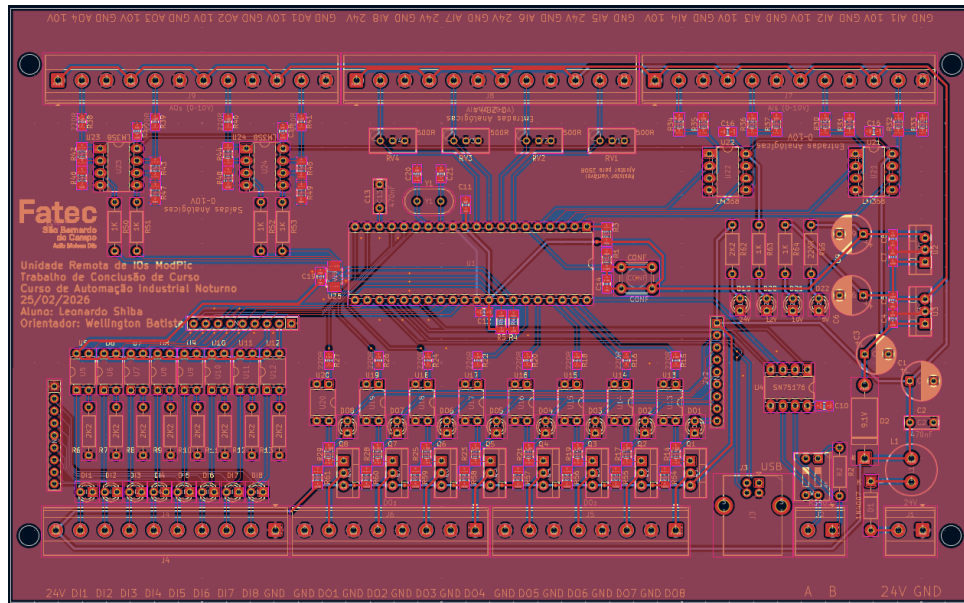
Fonte: Autoria própria

Com o circuito montado no KiCad, a última etapa do desenvolvimento de hardware consistiu em projetar o *layout* da placa de circuito impresso (PCI).

Além do dimensionamento correto das trilhas, um dos pontos imprescindíveis no roteamento da placa foi um plano de terra capaz de mitigar ruídos que pudessem vir a interferir nos sinais de comunicação.

A figura 7 mostra o roteamento final da unidade remota, enquanto a figura 8 apresenta uma visualização 3D prévia da unidade remota com os componentes já soldados.

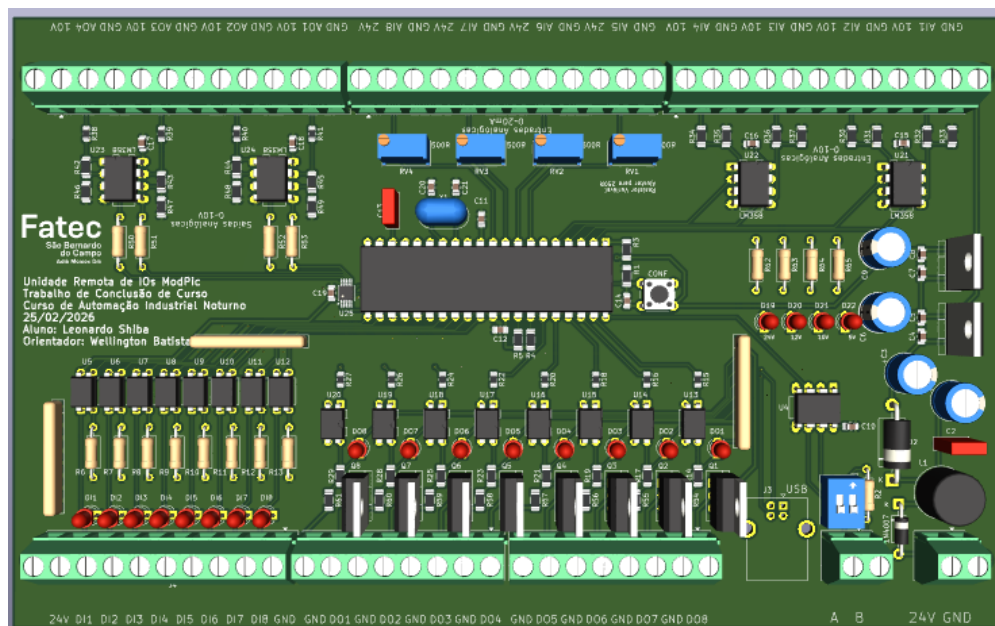
Figura 7 - Layout da Placa de Circuito Impresso



Fonte: Autoria própria

A figura 8 mostra o modelo 3D da placa gerado pelo KiCad.

Figura 8 - Visualização 3D da PCI

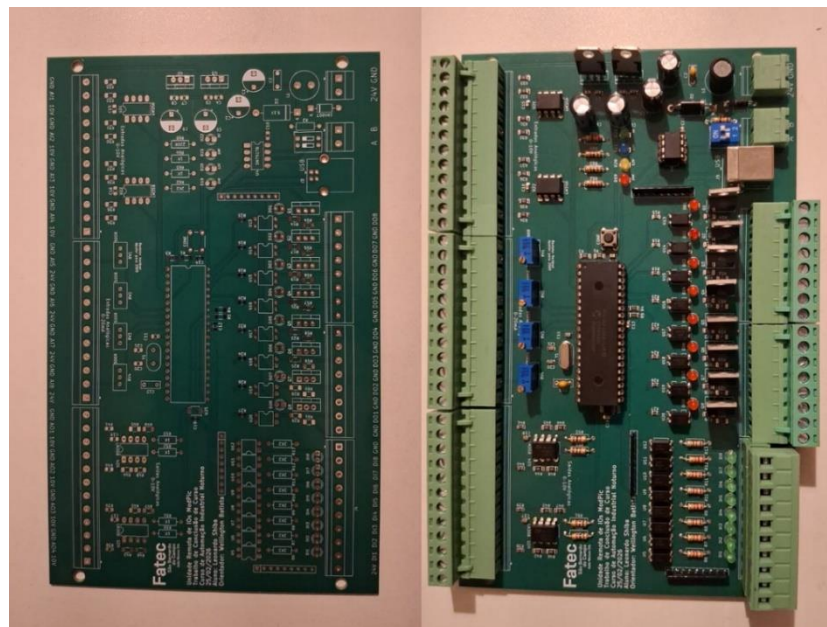


Fonte: Autoria própria

Após a conclusão do projeto de hardware da unidade remota, o projeto de PCI foi encomendado à JLCPCB².

Depois da fabricação da placa os componentes e conectores foram soldados e, após toda a montagem, foi testado mais uma vez o funcionamento de cada circuito individualmente. A figura 9 apresenta a placa de circuito impresso fabricada pela JLCPCB, antes e depois da soldagem dos componentes eletrônicos e conectores.

Figura 9 - PCI com e sem componentes

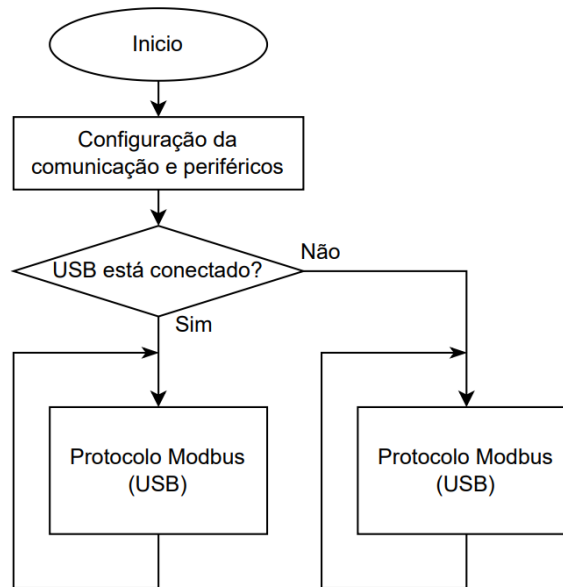


Fonte: Autoria própria

Após a conclusão do desenvolvimento do *hardware*, iniciou-se a elaboração do *firmware*. A programação foi elaborada de forma sequencial com a função principal de distinguir entre o uso do protocolo Modbus com o meio físico USB ou RS485. O funcionamento geral do *firmware* está descrito em forma de fluxograma na figura 10.

² Empresa global de manufatura de alta tecnologia, sediada em Shenzhen, China, especializada na prototipagem rápida e fabricação em larga escala de placas de circuito impresso (PCB) e serviços de montagem de componentes (SMT Assembly).

Figura 10 - Ciclo de funcionamento do *firmware* (Macro)



Fonte: Autoria própria

A primeira etapa da programação foi a configuração de todos os periféricos e portas utilizadas no projeto. Essa etapa inclui a configuração da frequência de *clock*, entradas digitais, saídas digitais, entradas analógicas, canal de comunicação UART, barramento de comunicação I²C, e canal de comunicação USB.

O próximo passo do projeto foi o estabelecimento da comunicação USB entre o microcontrolador e um computador, para que fosse possível validar as leituras e escritas da unidade remota MODPIC. Para isso, foi utilizada a biblioteca *USB CDC Device Basic*, fornecida pela Microchip.

Nesta biblioteca está disposta uma aplicação utilizando o microcontrolador PIC18F4550 como um dispositivo USB CDC (*Communication Device Class*), que permite a comunicação com através de uma porta virtual de comunicação (COM).

Após a inclusão da comunicação USB, foram testados os acionamentos e leituras do microcontrolador através de comandos utilizando a comunicação com o computador. O objetivo desse teste foi validar a comunicação e o funcionamento do circuito simultaneamente. Após os testes utilizando a comunicação USB, a próxima etapa de desenvolvimento foi a criação do protocolo Modbus utilizando o meio físico USB.

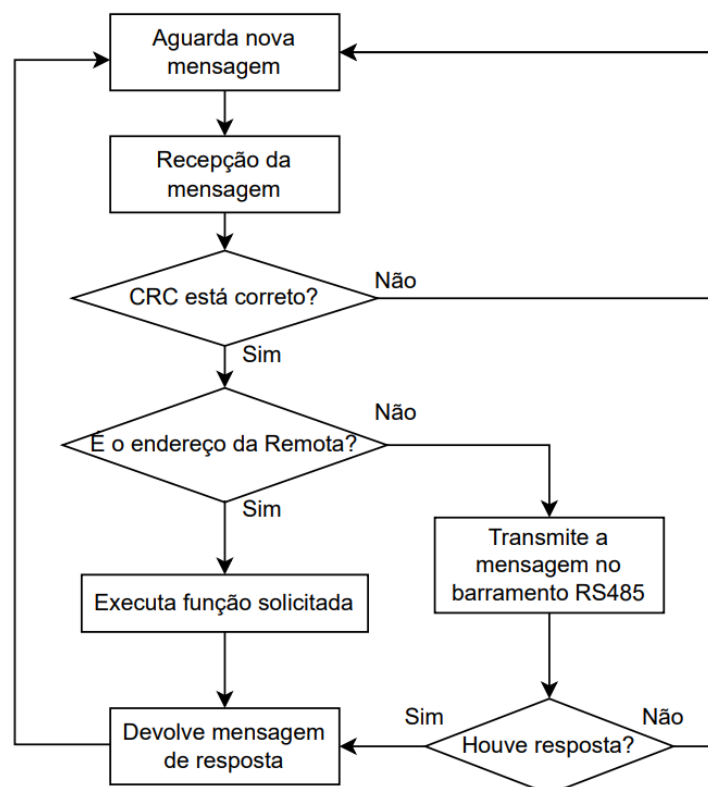
O funcionamento do Modbus através do USB se inicia na identificação de uma mensagem no seu buffer. Em seguida, é verificado se o valor do CRC da mensagem

está correto. Após a confirmação de que a mensagem está correta, é feita a identificação do endereço de escravo que aquela mensagem se destina. Quando a mensagem for destinada para a unidade MODPIC, verifica-se qual o código de função da mensagem e os dados contidos nela.

Após processar a mensagem enviada, a unidade remota executa a ação requisitada e devolve um *frame* de resposta ao mestre. Essa mensagem de resposta contém o endereço da remota, a função que ela executou, os dados da resposta, e por fim o CRC da mensagem.

O protótipo desenvolvido possui uma funcionalidade de *bridge*. Esta função faz com que a unidade remota replique a mensagem do mestre no barramento RS485 caso o endereço solicitado seja diferente do endereço configurado na remota. O funcionamento geral do protocolo Modbus USB está descrito em forma de fluxograma na figura 11.

Figura 11 - Ciclo de funcionamento Modbus USB



Fonte: Autoria própria

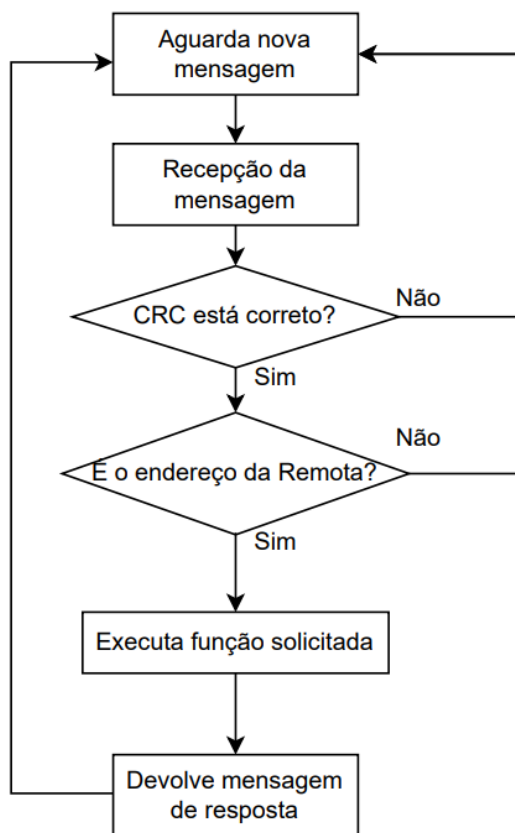
A unidade MODPIC utiliza apenas a faixa de *Holding Registers* para seu funcionamento como dispositivo Modbus. O uso deste tipo de registrador são só

permite a operação de entradas e saídas analógicas, mas também de entradas e saídas digitais utilizando a concatenação de *bits*.

Para garantir a simplicidade e funcionalidade do dispositivo, foram desenvolvidas apenas as funções FC03 (*Read Holding Registers*) e FC06 (*Preset Single Register*) do protocolo Modbus, análogas à leitura e escrita em *Holding Registers* respectivamente.

Após o desenvolvimento do Modbus no meio físico USB, o funcionamento foi replicado no meio físico RS485. A aplicação do protocolo no meio físico RS485 se difere apenas na funcionalidade de bridge e na recepção dos dados, que são transmitidos de forma serial. O funcionamento do Modbus no meio físico RS485 está descrito na figura 12.

Figura 12 - Ciclo de funcionamento Modbus RS485



Fonte: Autoria própria

Com a finalização do protocolo Modbus nos dois meios físicos do dispositivo, a funcionalidade básica da unidade remota foi consolidada. O próximo passo foi a configuração de ID e *baud rate* do protótipo.

Para trazer essas funções ao protótipo, foi necessária a configuração da EEPROM interna do microcontrolador para que esses dados fossem armazenados de forma retentiva.

Para fazer essas configurações, foi utilizado um *holding register* responsável por armazenar os dados de configuração. Para alterar as configurações, é utilizada a função FC06 para alterar o valor do registro conforme a necessidade do usuário.

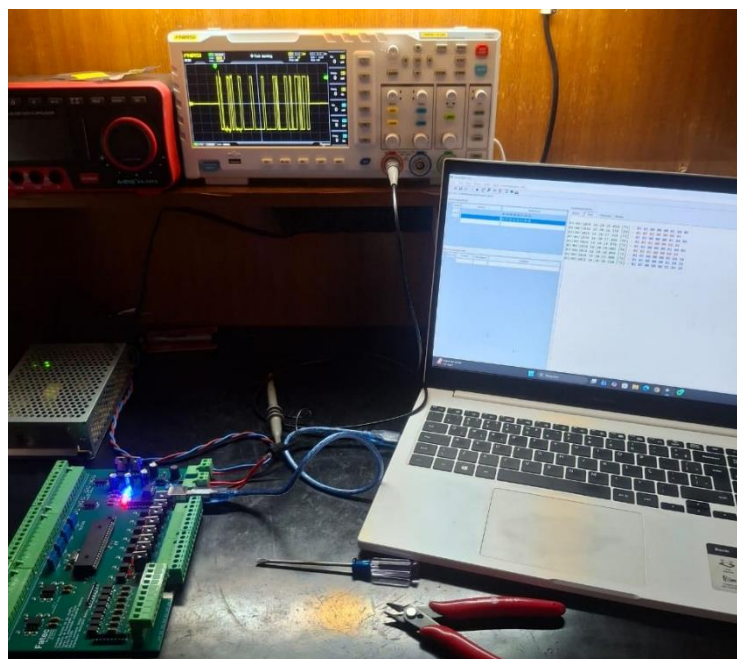
4.4 Integração e validação final

A validação do MODPIC concentrou-se na execução de testes de comunicação para garantir a confiabilidade do firmware e o correto mapeamento dos 28 pontos de I/O.

O processo de testes em bancada foi realizado com o *software* Docklight³. A vantagem desta ferramenta é a construção manual dos *frames* de comunicação, permitindo a manipulação de comandos e visualização do comportamento do MODPIC.

Graças a essa flexibilidade na comunicação, o Docklight foi ideal para realizar os testes iniciais de comunicação Modbus. A figura 13 mostra o teste de escrita no barramento RS485 com funcionalidade *bridge*, através do Docklight.

Figura 13 - Teste da função *Bridge* com o Docklight



Fonte: Autoria própria

³ Ferramenta de desenvolvimento e teste para protocolos de comunicação serial e de rede, utilizada para monitorar o diagnóstico de barramentos e simular o envio de telegramas de dados em tempo real.

Após a confirmação dos comandos básicos e a validação do sincronismo de dados, o dispositivo foi utilizado para se comunicar com o *software* ModScan⁴.

Este programa é um simulador de comunicação Modbus, gerando os *frames* de forma automática conforme as configurações escolhidas pelo usuário.

A escolha deste software se deu pela sua estrutura de comunicação Modbus com uma maior flexibilidade de configuração, permitindo testes menos rigorosos do *firmware* do MODPIC.

Após os testes com o ModScan, a unidade remota foi integrada ao ambiente do supervisor AVEVA Edge⁵, permitindo monitorar e comandar graficamente as entradas e saídas sob condições normais de operação.

Este *software* apresenta um padrão rigoroso de comunicação, sendo menos tolerante com latências de resposta e respostas de exceção.

Por fim, a validação final do protótipo foi concluída através da integração com o software CODESYS⁶, onde testou-se o funcionamento da unidade remota com o *SoftPLC* fornecido pelo *software*.

Para confirmar o teste, a unidade remota foi utilizada como CLP em uma bancada de automação da Fatec-SBC, interagindo com os sinais de campo e trocando dados com *SoftPLC*, que executa sua lógica com base nos sinais fornecidos pela comunicação com o MODPIC.

4.5 Avaliação do Artefato

A conclusão das etapas de testes e a integração final em bancada permitiram avaliar o MODPIC com base nos requisitos técnicos e operacionais estabelecidos no início do projeto. Essa análise verifica o grau de eficiência do protótipo e mede o impacto das escolhas de projeto frente à realidade prática de desenvolvimento.

O primeiro critério avaliado foi a capacidade de I/O e a integridade dos sinais. O hardware cumpriu o requisito de disponibilizar os 28 pontos de sinal mistos, garantindo o acionamento e a leitura das 08 entradas e 08 saídas digitais, bem como

⁴ Aplicativo Windows desenvolvido pela Win-Tech, projetado para atuar como um dispositivo Mestre na arquitetura Modbus (RTU e TCP/IP). É utilizado para testar e validar redes industriais, permitindo a leitura e escrita direta em registradores (*Holding Registers*, *Input Registers*, *Coils* e *Discrete Inputs*) de dispositivos escravos conectados.

⁵ Plataforma de software para desenvolvimento de sistemas IHM e SCADA, desenvolvida originalmente pela InduSoft sob o nome *InduSoft Web Studio* e atualmente integrada ao portfólio da AVEVA. É utilizada na automação industrial para criar telas de monitoramento gráfico, gerenciar alarmes e estabelecer comunicação com CLPs e remotas via drivers nativos.

⁶ Plataforma de software independente de hardware, desenvolvida pela empresa alemã CODESYS Group (3S-Smart Software Solutions), utilizada para a programação de Controladores Lógicos Programáveis (CLPs) segundo a norma IEC 61131-3. O ambiente integra o desenvolvimento de lógica de controle, configuração de redes industriais e criação de interfaces de supervisão (IHM).

o processamento das 08 entradas analógicas com resolução de 10 bits e das 04 saídas analógicas.

Os testes de realizados no ModScan e no AVEVA Edge confirmaram que o mapa de registradores internos do firmware manteve-se estável, atualizando as variáveis em tempo satisfatório sem perdas de sincronismo ou travamentos do microcontrolador PIC18F4550.

O segundo aspecto crucial foi a validação das interfaces de comunicação e da lógica de *bridge*. O firmware demonstrou conformidade com o padrão Modbus RTU, respondendo corretamente aos frames do mestre.

Durante os experimentos com o Docklight e o CODESYS, a porta USB emulada via classe CDC manteve a conexão ativa e estável.

A comunicação se mostrou eficaz na bancada da unidade de ensino da Fatec-SBC permitindo, a interação entre o *SoftPLC* e os sinais da bancada de automação, realizando o controle de um processo discreto.

Por outro lado, o requisito de viabilidade econômica não foi atendido devido ao impacto alfandegário na aquisição da placa de circuito impresso. A taxa de importação elevou o custo isolado da PCI de R\$300 para R\$700, fazendo com que o custo total do protótipo finalizasse em mais de 1000 reais. A tabela 2 a seguir apresenta uma estimativa de custos envolvidos no projeto.

Tabela 2 - Tabela de custos envolvidos na fabricação da MODPIC

Item / Grupo de Componentes	Qtd	Valor Unitário Médio	Subtotal Calculado
Placa de Circuito Impresso (PCI) Fabricada + Taxação	1	R\$ 700,00	R\$ 700,00
Microcontrolador PIC18F4550-IP (U1)	1	R\$ 100,00	R\$ 100,00
MOSFETs Canal P IRF9540N (Q1 a Q8)	8	R\$ 6,00	R\$ 48,00
Conversor DAC I2C MCP4728 (U25)	1	R\$ 25,00	R\$ 25,00
Optoacopladores PC817 (U5 a U20)	16	R\$ 1,50	R\$ 24,00
Transceptor RS485 SN75176AD (U4)	1	R\$ 20,00	R\$ 20,00
Amplificadores Operacionais LM358 (U21 a U24)	4	R\$ 2,50	R\$ 10,00
Reguladores de Tensão LM7805 / LM7810 (U2, U3)	2	R\$ 3,00	R\$ 6,00
Bornes KRE / Fênix (J1, J2, J4 a J9)	9	R\$ 8,00 (média por bloco)	R\$ 72,00
Componentes Discretos (Resistores, Capacitores, LEDs, Diodos, Cristal)	1	Lote Geral	R\$ 48,00
Custo Total Calculado	-	-	R\$ 1.053,00

Fonte: Autoria própria

Diante deste cenário, o MODPIC falhou no objetivo financeiro inicial, uma vez que o valor final superou o preço de mercado de módulos de aquisição integrados comerciais equivalentes.

Conclui-se que, embora o artefato tenha obtido sucesso pleno em todas as suas capacidades técnicas e de engenharia de software, sua replicação em larga escala nos laboratórios torna-se economicamente inviável sob as condições atuais de importação de hardware, demandando uma readequação futura nos métodos de fabricação da placa para reduzir os custos fixos.

5 Considerações Finais

Analisando-se os objetivos iniciais do projeto, pode-se inferir diante dos resultados que o MODPIC obteve êxito em suas capacidades técnicas, especialmente no desenvolvimento do código.

Contudo, por mais que as medidas tomadas para reduzir os custos de desenvolvimento de *hardware* (como o uso de um GND comum entre todas as fontes) tenham diminuído drasticamente o preço do protótipo, o requisito de viabilidade econômica não foi atendido devido ao impacto direto da taxa alfandegária sobre a placa de circuito impresso.

O custo final de R\$ 1.053,00 superou o preço de mercado de soluções comerciais integradas equivalentes, inviabilizando a replicação imediata do artefato em larga escala para os laboratórios da instituição sob as condições econômicas atuais.

Como trabalho futuro, é sugerido o desenvolvimento de uma unidade remota modular. Desta forma é possível o desenvolvimento de um *hardware* com isolamento galvânico dos sinais de campo. Além disso, a criação de placas separadas simplifica o processo de desenvolvimento dos circuitos, permitindo que a fabricação de PCI seja feita de forma artesanal, diminuindo drasticamente o custo de fabricação.

REFERÊNCIAS

BROCKE et. al. **Design Science Research. Cases**. [S.l.]: Springer, Cham, 2020.

CAUCHICK, P. P. **Metodologia Científica para Engenharia**. 1a edição ed. [s.l.] GEN LTC, 2019.

GROOVER, Mikell P. **Automação Industrial e Sistemas de Manufatura**. 3. ed. São Paulo: Pearson, 2011.

IBM. **O que é um microcontrolador?** *IBM Cloud Education*, [S.l.], 2024. Disponível em: <<https://www.ibm.com/br-pt/think/topics/microcontroller>>. Acesso em: 02 nov. 2025.

MICROCHIP TECHNOLOGY. **PIC18F2455/2550/4455/4550 Data Sheet: 28/40/44-Pin High-Performance, Enhanced Flash USB Microcontrollers with nanoWatt Technology.** Edição: DS39632C. Chandler: Microchip Technology, 2009. Disponível em: <<https://ww1.microchip.com/downloads/en/devicedoc/39632c.pdf>>. Acesso em: 2 nov. 2025.

MIYADAIRA, Alberto Noboru. **Microcontroladores PIC18: aprenda e programe em linguagem C. 1 Ed.** São Paulo: Érica, 2009.

MORAES, Cícero. **Engenharia de Automação Industrial.** 2. Ed. Rio de Janeiro: LTC, 2007.

MODBUS-IDA. **Modbus over serial line specification and implementation guide V1.02.** Massachusetts: Modbus-IDA, 2007. Disponível em: <http://www.modbus.org/docs/Modbus_over_serial_line_V1_02.pdf>. Acesso em: 22 out. 2025.

PRODANOV, Cleber Cristiano; FREITAS, Ernani Cesar de. **Metodologia do trabalho científico: métodos e técnicas da pesquisa e do trabalho acadêmico.** 2. ed. Novo Hamburgo: Feevale, 2013.

PRUDENTE, Francesco. **Automação Industrial: PLC: Teoria e Aplicações.** 2. Ed. Rio de Janeiro: LTC, 2020.