



Instituto Politécnico  
de Castelo Branco  
Escola Superior  
de Tecnologia

# **Sistema de Monitorização e Controlo para máquinas móveis**

## **Monotorização e controle de um harvester**

Márcia Venerosa Cucubica André

Nº 20210250

Orientador Professor Doutor Pedro Miguel Baptista Torres

Relatório de Projeto apresentado à Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco para cumprimento dos requisitos necessários à obtenção do grau de Licenciado em Engenharia Eletrotécnica e das Telecomunicações, realizada sob a orientação científica do Professor Adjunto Doutor Pedro Miguel Baptista Torres, do Instituto Politécnico de Castelo Branco.

Outubro de 2024



## **Composição do júri**

Presidente do júri

Professor Doutor Paulo Jorge Dias Torres

Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco

Vogais

Professor Especialista Fernando Júlio Marques Miranda

Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco

Professor Doutor Pedro Miguel Baptista Torres (orientador)

Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco



## Dedicatória

Gostaria de dedicar este trabalho a todas as pessoas que tornaram possível a minha jornada acadêmica e que estiveram ao meu lado durante este período desafiador e gratificante.

Primeiramente, quero agradecer a Deus pela vida sem isso não estaria falando com vocês hoje, expressar minha profunda gratidão à minha família. Agradeço aos meus pais pelo amor incondicional, apoio emocional e pelos sacrifícios que fizeram para me proporcionar esta oportunidade de educação. Sem o seu apoio constante, eu não estaria aqui hoje.

Ao meu orientador Pedro Torres, gostaria de expressar minha sincera gratidão pela orientação, paciência e incentivo ao longo deste projeto. Seu conhecimento e orientação foram fundamentais para o desenvolvimento deste trabalho, e sou imensamente grata por sua dedicação na orientação desse trabalho.

Aos meus amigos e colegas de classe, obrigado por compartilharem esta jornada comigo. Suas palavras de encorajamento, colaboração e camaradagem fizeram toda a diferença. Cada conversa, debate e momento compartilhado contribuiu para o meu crescimento pessoal e acadêmico.

Por fim, quero expressar minha admiração e gratidão aos professores e profissionais que me inspiraram ao longo deste percurso. Suas lições e exemplos moldaram meu pensamento e me motivaram a buscar a excelência em cada etapa do caminho.

Este trabalho é dedicado a vocês, que tornaram minha jornada educacional possível e significativa. Espero que meu esforço e dedicação neste trabalho possam retribuir de alguma forma o apoio e a confiança que recebi.

Muito obrigado a todos.

Márcia André



## Agradecimentos

Gostaria de começar expressando minha gratidão a Deus, cuja orientação e bênçãos estiveram sempre presentes em minha jornada acadêmica.

À minha família, especialmente à minha mãe, que tem sido minha fonte de inspiração, apoio incondicional e amor inabalável. Suas palavras de encorajamento e incentivo foram minha força nos momentos mais desafiadores. Às minhas irmãs, que sempre estiveram ao meu lado, compartilhando alegrias e superando obstáculos juntas, meu profundo agradecimento.

Ao meu orientador Pedro Torres, agradeço sinceramente pela orientação excepcional, paciência e apoio dedicados ao longo deste trabalho. Suas habilidades de orientação foram fundamentais para o sucesso deste projeto.

Aos meus amigos e colegas de classe, agradeço pelas memórias compartilhadas, pelo apoio mútuo e pela jornada enriquecedora que vivemos juntos ao longo desses anos.

Também desejo expressar minha gratidão aos professores e funcionários do Instituto Politécnico de Castelo Branco, cujo compromisso com a excelência acadêmica e dedicação ao ensino contribuíram significativamente para minha formação.

Além disso, gostaria de agradecer às organizações e instituições que forneceram suporte financeiro e oportunidades de pesquisa ao longo deste projeto.

Por último, mas não menos importante, agradeço a todos aqueles que, de alguma forma, contribuíram para minha jornada acadêmica. Seus esforços e apoio foram fundamentais para meu crescimento pessoal e profissional.

Muito obrigado a todos.

Márcia André



## Resumo

O presente trabalho propõe o desenvolvimento de uma plataforma modular de monitorização e controlo para máquinas móveis, com foco específico em *harvesters*, uma máquina florestal amplamente utilizada em operações de corte e processamento de árvores. Inserido no conceito de **Indústria 4.0** e **Floresta 4.0**, o sistema tem como objetivo melhorar a eficiência operacional, reduzir falhas e minimizar os custos de manutenção através da monitorização em tempo real, abrindo o caminho para soluções de telemanutenção.

Para tal, foi desenvolvida uma solução baseada em módulos eletrónicos interligados por uma rede **CAN Bus**, utilizando o **protocolo CANopen** para configuração. O sistema integra o módulo **IFM CR1202** (HMI) como interface homem-máquina, o módulo **CR2014** para leitura dos comandos dos joysticks HJ60 e dois módulos **CR2512** para controlo dos acionamentos do *harvester*. A comunicação entre esses módulos ocorre de forma eficiente, garantindo o controlo em tempo real de todos os componentes da máquina.

Além disso, a solução implementada permite a interação com um simulador/emulador para treino de operadores e treino de ações de manutenção, simulador esse que não é objeto deste trabalho, mas é sim parte integrante de um projeto mais vasto em desenvolvimento por outros parceiros.

Os testes realizados confirmaram a eficácia do sistema em ambientes reais e simulados, demonstrando seu potencial para aumentar a produtividade e competitividade no setor florestal, alinhando-se aos princípios da **Floresta 4.0**.

## Palavras-chave

Indústria 4.0; Floresta 4.0; CAN Bus; *Harvester*; Monitorização Remota.



## Abstract

This work proposes the development of a modular monitoring and control platform for mobile machinery, with a specific focus on harvesters, and a widely used forestry machine for tree cutting and processing operations. Aligned with the concepts of Industry 4.0 and Forestry 4.0, the system aims to improve operational efficiency, reduce failures, and minimize maintenance costs through real-time monitoring, paving the way for telemaintenance solutions.

To achieve this, a solution based on electronic modules interconnected via a CAN Bus network was developed, utilizing the CANopen protocol for configuration. The system integrates the IFM CR1202 module (HMI) as the human-machine interface, the CR2014 module for reading commands from the HJ60 joystick, and two CR2512 modules for controlling the harvester's actuators. Communication between these modules occurs efficiently, ensuring real-time control of all machine components.

Additionally, the implemented solution allows interaction with a simulator/emulator for operator training and maintenance actions, which is not the focus of this work but is part of a larger project in development by other partners.

The tests conducted confirmed the system's effectiveness in both real and simulated environments, demonstrating its potential to increase productivity and competitiveness in the forestry sector, aligning with the principles of Forestry 4.0.

## Keywords

Industry 4.0; Florestry 4.0; CAN Bus; *Harvester*; Remote Monitoring.



## Índice geral

1. Introdução .....	1
1.1 Contextualização do Tema .....	1
1.1.1 Diagrama em bloco do sistema de monitorização e controle de um harvester .....	3
1.2 Justificação e Relevância .....	4
1.3 Problema de Investigação .....	5
1.4 Objetivos do Trabalho .....	6
1.4.1 Objetivo Geral .....	6
1.4.2 Objetivos Específicos .....	6
1.5 Metodologia Resumida .....	7
1.6 Estrutura do Trabalho .....	8
2. Fundamentação Teórica .....	9
2.1 Indústria 4.0 e Floresta 4.0 .....	9
2.1.1 Conceito de Indústria 4.0 .....	9
2.1.2 Aplicação no Setor Florestal – Floresta 4.0 .....	9
2.1.3 Contextualização do Projeto SMARTCUT .....	9
2.2 Tecnologias de Monitorização e Controlo de Máquinas Móveis .....	9
2.2.1 Monitorização Remota e Manutenção Preditiva .....	9
2.2.2 Controlo de Máquinas Móveis .....	10
2.3 Protocolo CAN Bus .....	10
2.3.1 Introdução ao Protocolo CAN Bus .....	10
2.3.2 CANopen .....	11
2.4 Interfaces Homem-Máquina (HMI) .....	11
2.4.1 Importância da HMI no Controlo de Máquinas .....	11
2.4.2 Aplicação do IFM CR1202 .....	11
2.5 Joystick HJ60 e Controlo da Lâmina de Corte .....	11
2.5.1 Tecnologia de Joysticks Industriais .....	11
2.5.2 Integração do HJ60 no Sistema .....	11
2.6 Gémeo Digital e Simuladores para Operação e Manutenção .....	12
2.6.1 Conceito de Gémeo Digital .....	12
2.6.2 Simuladores de Máquinas Móveis .....	12
2.7 Resumo da Revisão .....	12

3. Metodologia.....	12
3.1 Descrição Geral do Sistema.....	12
3.1.1 Arquitetura do Sistema.....	13
3.2 Desenvolvimento do Software de Controle .....	48
3.2.1 Ambiente de Desenvolvimento Codesys .....	48
3.2.2 Integração dos Módulos na Rede CAN Bus .....	59
3.2.2.1 Protocolo CAN Bus e CANopen .....	59
3.2.2.2 Implementação do Controle da Lâmina de Corte via Joystick HJ60 .....	59
3.3 Desenvolvimento do Simulador Físico e Gémeo Digital .....	59
3.3.1 Demonstrador Físico .....	59
3.4 Testes e Validação do Sistema.....	60
3.4.1 Testes de Comunicação.....	60
3.5 Processo de Validação.....	60
3.5.1 Implementação de um Filtro para Estabilização dos Sinais.....	60
3.5.2 Resultados da Validação com o Filtro .....	60
3.6 Fluxograma da Metodologia .....	61
4. Desenvolvimento da Plataforma de Controlo e Monitorização.....	61
4.1 Desenho do Sistema.....	61
4.1.1 Diagrama de Blocos Funcional .....	62
4.2 Arquitetura do Sistema de Monitorização e Controlo de Máquinas Móveis.....	62
4.2.1 Ambiente de Simulação, Plataforma de Monitorização e Controlo .....	62
4.2.1 Configuração da Rede CAN e Detalhes das Configurações dos Módulos IFM. 65	
4.2.2 CANopen Manager.....	65
4.3 Controle do Motor via Joystick HJ60 e sua implementação no sistema experimental.....	80
4.4 Desafios e Soluções .....	85
5. Resultados e Discussões.....	96
5.1 Introdução dos Resultados .....	96
5.2 Resultados no Ambiente de Simulação .....	96
5.2.1 Desempenho do Controle via Joystick .....	96
5.2.2 Eficácia da Comunicação CANopen e CANBUS.....	102
5.3 Resultados no Ambiente Físico .....	103
5.3.1 Testes de Funcionamento do Protótipo .....	103

5.4 Discussões dos Resultados.....	108
5.4.1 Análise Comparativa entre o Ambiente de Simulação e o Ambiente Físico ..	108
5.4.2 Contribuições para a Eficiência Operacional .....	108
6 – Conclusão e Trabalhos Futuros .....	109
6.1 Conclusão.....	109
6.1.1 Síntese do Projeto.....	109
6.1.2 Principais Contribuições.....	109
6.2 Trabalhos Futuros .....	109
7. Referências e ANEXOS.....	111



## Índice de figuras

<u>Figura 1- Ilustração de um harvester e logotipo do projeto SMARTCUT.....</u>	2
<u>Figura 2- Ilustração 2D do protótipo desenvolvido.....</u>	3
<u>Figura 3- Diagrama do sistema.....</u>	3
<u>Figura 4 - Joystick Chengong HJ60.....</u>	14
<u>Figura 5 - Ligações elétricas dos joysticks HJ60 .....</u>	15
<u>Figura 6 - cabo de adaptador de joystick.....</u>	17
<u>Figura 7 - Fonte de alimentação EDR-120.....</u>	18
<u>Figura 8 - Ligações elétrica da fonte de alimentação EDR-120.....</u>	18
<u>Figura 9 - Fonte de 5V.....</u>	20
<u>Figura 10 - Ligações elétrica da fonte de alimentação de 5V.....</u>	20
<u>Figura 11 - Driver de Motor BTS7960 (H-Bridge).....</u>	22
<u>Figura 12 - Ligações e funções dos pinos do driver de Motor BTS7960 (H-Bridge).....</u>	23
<u>Figura 13 - Ligações da alimentação do motor (24V).....</u>	24
<u>Figura 14 - Optoacoplador de 8 canais.....</u>	25
<u>Figura 15 - Placa de Potência com 4 canais MOSFET.....</u>	27
<u>Figura 16 - Motor ZGA42RH(A6249) de 24V.....</u>	28
<u>Figura 17 - Sensor Indutivo.....</u>	29
<u>Figura 18 - Esquema de ligações do sensor indutivo através de ficha M8.....</u>	30
<u>Figura 19 - Ligação NPN NA do sensor indutivo.....</u>	31
<u>Figura 20 - Sensor indutivo normalmente fechado e Sensor de velocidade.....</u>	31
<u>Figura 21 - Ligação NPN NA e NPN NF.....</u>	32
<u>Figura 22 - Cabo do sensor de velocidade com ficha M8.....</u>	33
<u>Figura 23 - Ligações elétricas do cabo dos sensores.....</u>	34
<u>Figura 24 - Módulo de E/S remotas - IFM CR2512.....</u>	35
<u>Figura 25 - Cabo de ligação aos módulos de E/S, estanque.....</u>	39
<u>Figura 26 - Módulo de E/S remotas para cabine - IFM CR2014.....</u>	39
<u>Figura 27 - Ligação elétrica do CR2014.....</u>	40
<u>Figura 28 - Cabos de ligação ao módulo IFM CR2014.....</u>	43
<u>Figura 29 - Controlador com HMI integrada - IFM CR1202.....</u>	43
<u>Figura 30 - Conector e cablagem da ligação de 40 pinos do CR1202.....</u>	47
<u>Figura 31 - Botão de pressão.....</u>	48
<u>Figura 32 - Disjuntor.....</u>	48
<u>Figura 33 - Bornes de ligação.....</u>	49
<u>Figura 34 - Sinal PWM.....</u>	50
<u>Figura 35 - Duty cycle (ciclo de trabalho).....</u>	50
<u>Figura 36 - Software de programação para controladores industriais.....</u>	51
<u>Figura 37 - Ilustração da integração dos módulos CR2512 e do CR2014 na rede.....</u>	52
<u>Figura 38 - Ilustração das configurações feitas nos módulos CR2512 e do CR2014.....</u>	53
<u>Figura 39 - Blocos de função desenvolvidos no codesys.....</u>	54
<u>Figura 40 - Exemplificação de barramento CAN.....</u>	56
<u>Figura 41 - Barramento CAN com diferentes nós.....</u>	57
<u>Figura 42 - Acoplamento do barramento com resistências.....</u>	57
<u>Figura 43 - Nível de barramento CAN.....</u>	58

<u>Figura 44 - Modelo OSI e a implementação da rede CANopen e CANbus.</u>	59
<u>Figura 45 - Trama padrão e uma trama alargada.</u>	61
<u>Figura 46 - Arquitectura do sistema.</u>	69
<u>Figura 47 - Ficheiro EDS.</u>	71
<u>Figura 48 - Livraria de dispositivos.</u>	71
<u>Figura 49 - Template ecomatDisplay.</u>	72
<u>Figura 50 - Dispositivos para ligação CAN.</u>	72
<u>Figura 51 - Módulos CAN.</u>	73
<u>Figura 52 - Endereçamento IP.</u>	73
<u>Figura 53 - Verificação da ligação IP e máscara.</u>	73
<u>Figura 54 - Configurações da CAN Manager.</u>	74
<u>Figura 55 - Configurações de sincronismo CAN Manager.</u>	75
<u>Figura 56 - Esquema do módulo CR2014.</u>	75
<u>Figura 57 - Opções do Baud rate e ID do módulo CR2014.</u>	76
<u>Figura 58 - Integração dos módulos na rede can dentro do codesys.</u>	76
<u>Figura 59 - SDOs do módulo CR2014.</u>	77
<u>Figura 60 - Selecionar itens para o objeto do diretório do módulo CR2014.</u>	77
<u>Figura 61 - Configuração do bit rate switch a e b do módulo CR2014.</u>	78
<u>Figura 62 - Configuração dos parâmetros gerais do modulo CR2014.</u>	78
<u>Figura 63 - CR2014, Mapa de entradas se sinal no codesys.</u>	79
<u>Figura 64 - Módulo CR2512 na rede CAN no codesys.</u>	80
<u>Figura 65- Habilitação do módulo CR2512 no codesys.</u>	80
<u>Figura 66 - Definir o Node ID módulo CR2512 1.</u>	80
<u>Figura 67- SDOs do módulo CR2512.</u>	81
<u>Figura 68 - Selecionar o obejeto 20F0 e 20F1 ao módulo CR2512 1.</u>	81
<u>Figura 69 - Alteração da configuração padrão Node ID 32 para Node ID 33.</u>	82
<u>Figura 70 - Configurações dos parâmetros gerais do módulo CR2512.</u>	83
<u>Figura 71- Mapeamento das variáveis no módulo CR2512.</u>	84
<u>Figura 72 - Acessando ao SDOs do módulo CR2512.</u>	84
<u>Figura 73 - Configuração dos canais 1 e 2 como saída PWW e definição do valor da frequência.</u>	85
<u>Figura 74 - Foi configurado os canais 1 e 2 como saída PWW e definido valor da frequência 200Hz.</u>	86
<u>Figura 75 - Sinal PWM no osciloscópio.</u>	86
<u>Figura 76 - Configuração das variáveis no POU.</u>	87
<u>Figura 77 - Bloco de função do PWM e bloco de conversão do inteiro para time.</u>	87
<u>Figura 78 - Sinal PWM na prática.</u>	88
<u>Figura 79 - Blocos de função responsáveis por gerar o sinal PWM no codesys.</u>	88
<u>Figura 80 - Pacote e livrarias ecomatDisplay.</u>	89
<u>Figura 81 - Recovery App e Update to ecomatDisplay.</u>	89
<u>Figura 82 - Instalação do pacote e livrarias do ecomatDisplay.</u>	89
<u>Figura 83 - Programação em LADDER dos botões dos joysticks.</u>	90
<u>Figura 84 - Entradas dos botões dos joysticks alocadas no módulo CR2014.</u>	91
<u>Figura 85 - Entradas dos roletes e eixos X e Y dos joysticks alocados no módulo CR2014.</u>	91

<u>Figura 86 - Barras de visualização dos movimentos do rolete e eixos X e Y do joystick 2.</u>	92
<u>Figura 87 - Blocos de ajuste escala dos roletes e eixos X e Y.</u>	93
<u>Figura 88 - Programação LADDER dos sensores indutivos.</u>	94
<u>Figura 89 - Figura representativa dos sensores na interface.</u>	94
<u>Figura 90 - Interface do ecomatDisplay.</u>	95
<u>Figura 91- Livraria de imagens adicionadas.</u>	95
<u>Figura 92 - Esquema do sistema experimental 1.</u>	96
<u>Figura 93 - Esquema do sistema experimental 2.</u>	97
<u>Figura 94 - Sinal PWM a saída do módulo.</u>	98
<u>Figura 95 - Sinal Enviado pelo joysticker antes do filtro.</u>	102
<u>Figura 96 - Sinal Enviado pelo joysticker depois do filtro ser implementado.</u>	102
<u>Figura 97 - Testando na prática e no codesys o botão 1 do joysticker 1.</u>	106
<u>Figura 98 - Testando na prática e no codesys o botão 2 do joysticker 2.</u>	107
<u>Figura 99 - Blocos de função para ajuste do scroll-j1, scale-Y e bloco de inversão de valor representação no prático.</u>	107
<u>Figura 100 - Blocos de função para ajuste do scroll-j2, scale-Y e scale-X representação no prático.</u>	107
<u>Figura 101 - Testando a scale-Y joytker 2 na prática no máximo.</u>	108
<u>Figura 102 - Testando a scale-Y joytker 2 na prática no mínimo.</u>	108
<u>Figura 103 - Testando a scroll joytker 2 na prática no mínimo.</u>	108
<u>Figura 104 - Testando a scroll joytker 2 na prática no máximo.</u>	109
<u>Figura 105 - Testando a scroll joytker 1 na prática.</u>	109
<u>Figura 106 - Testando o sensor indutivo NF na prática, o led acende quando distante do metal.</u>	110
<u>Figura 107 - Testando o sensor indutivo NF na prática, o led apaga quando próximo do metal.</u>	110
<u>Figura 108 - Testando o sensor indutivo NA, na prática o led apaga quando distante do metal.</u>	111
<u>Figura 109 - Testando o sensor indutivo NA, na prática o led acende quando próximo do metal.</u>	111
<u>Figura 110 - Testando o sensor de velocidade na prática, o led apaga quando distante do metal.</u>	111
<u>Figura 111 - Testando o sensor de velocidade na prática, os leds oscilam entre si quando próximo do metal.</u>	112
<u>Figura 112 - Rede CANbus e CANopen na prática.</u>	112
<u>Figura 113 - Demonstração 1 da variação da tenção entregue ao optoacoplador com o manuseio do joytik 2.</u>	113
<u>Figura 114 - Demonstração 2 da variação da tenção entregue ao optoacoplador com o manuseio do joytik 2.</u>	113
<u>Figura 115 - Demonstração da variação da tenção entregue ao optoacoplador com o manuseio do joytik 2.</u>	114
<u>Figura 116 - Demonstração 4 da variação da tenção entregue ao optoacoplador com o manuseio do joytik 2.</u>	114

<u>Figura 117 - Demonstração 1 da variação da tensão entregue ao driver mosfet com o manuseio do joytik 2.</u> .....	115
<u>Figura 118 - Demonstração 2 da variação da tensão entregue ao driver mosfet com o manuseio do joytik 2.</u> .....	115
<u>Figura 119 - Teste na fonte de 5V.</u> .....	116
<u>Figura 120 - Teste na fonte de 24V.</u> .....	116
<u>Figura 121 - Divice (ECOMATdisplay) no codesys.</u> .....	116
<u>Figura 122 - Scan network no divice ecomatdisplay.</u> .....	117
<u>Figura 123 - Fazendo o login.</u> .....	117
<u>Figura 124 - Entrando no Debug.</u> .....	118

## Lista de tabelas

<u>Tabela 1 - Ligações elétricas do joystick HJ60 – conector de 5 pinos.....</u>	16
<u>Tabela 2 - Ligações elétricas do joystick HJ60 - conector de 7 pinos.....</u>	17
<u>Tabela 3 - Terminal Pin No. Assignment (TB1).....</u>	20
<u>Tabela 4 - Terminal Pin No. Assignment (TB2).....</u>	20
<u>Tabela 5 - Pin-Out (ligações elétricas) da fonte de 5V.....</u>	22
<u>Tabela 6 - Função dos Pinos de Entrada de Controle.....</u>	24
<u>Tabela 7 - Ligações da saída de potência do driver.....</u>	25
<u>Tabela 8 - Ligações elétricas do optocoplador.....</u>	26
<u>Tabela 9 - Mapa de ligações do driver de potência.....</u>	28
<u>Tabela 10 – Infirmarções relevantes do Motor DC.....</u>	30
<u>Tabela 11 - Tipos de ligações.....</u>	35
<u>Tabela 12 – Ligações do CR2512 (Alimentação).....</u>	36
<u>Tabela 13 - Ligações do CR2512 (Comunicação CAN).....</u>	37
<u>Tabela 14 - Ligações completas dos canais.....</u>	37
<u>Tabela 15 - Ligações no conector X1.....</u>	41
<u>Tabela 16 - Ligações no conector X2.....</u>	42
<u>Tabela 17 - Ligações no conector X3.....</u>	44
<u>Tabela 18 - Ligações do conector AMO de 40 pinos do módulo CR1202.....</u>	45
<u>Tabela 19 – Ligações do conector M12 de 5 pinos do CR1202.....</u>	48
<u>Tabela 20 - Ligações do disjuntor.....</u>	50
<u>Tabela 21 - Relação entre ritmo de transmissão e distâncias de comunicação.....</u>	60
<u>Tabela 22 - Arbitragem da Trama de Dados.....</u>	62



## Lista de abreviaturas, siglas e acrónimos

CANBus

*Controller Area Network Bus* (Barramento de rede de área de controlador)

CANopen

*Network communication protocol for embedded systems based on the CAN Bus*  
(Protocolo de comunicação em rede para sistemas embarcados com base no CAN Bus)

CR1202

IFM CR1202 module (ecomatDisplay), used as Human-Machine Interface (HMI)  
(Módulo IFM CR1202 (ecomatDisplay), utilizado como interface homem-máquina)

CR2512

*Control module used for reading sensors and generating PWM signals* (Módulo de controle utilizado para leitura de sensores e geração de sinais PWM)

CR2014

Control module used for reading input signals, such as the HJ60 joystick (Módulo de controle utilizado para a leitura de sinais de entrada, como o joystick HJ60)

HMI

Human-Machine Interface (Interface Homem-Máquina)

HJ60

Joystick used for motor control in the harvester prototype (Joystick utilizado para o controle do motor no protótipo de harvester)

IA

Artificial Intelligence (Inteligência Artificial)

IoT

Internet of Things (Internet das Coisas)

PWM

Pulse Width Modulation (Modulação por Largura de Pulso)

SMARTCUT

Project for Remote Diagnosis, Maintenance, and Simulators for Training in Forest Machine Operation and Maintenance (Projeto de Diagnóstico e Manutenção Remota e Simuladores para Formação de Operação e Manutenção de Máquinas Florestais)

VICORT

Brand owned by Cutplant Solutions S.A. (Marca pertencente à empresa Cutplant Solutions S.A.)

Indústria

4.0

Fourth Industrial Revolution focused on automation, data exchange, and smart technologies (Quarta Revolução Industrial focada em automação, troca de dados e tecnologias inteligentes)

PC

Personal Computer (Computador Pessoal)

Codesys

*Programming and simulation environment for industrial automation systems* (Ambiente de programação e simulação para sistemas de automação industrial)

SIM

*Simulator used to recreate machine operations in a virtual environment* (Simulador utilizado para recriar operações de máquinas em ambiente virtual)

IFM

*Supplier of control and automation systems used in the project* (Fornecedor de sistemas de controle e automação utilizados no projeto)

Gêmeo

Digital

*Virtual representation of a physical system for simulations and optimizations* (Representação virtual de um sistema físico para simulações e otimizações)

VR

*Virtual Reality* (Realidade Virtual)

S.A.

Sociedade Anônima (*type of company*) (Tipo de empresa)

CR

*Controller* (Controlador)

MES

*Manufacturing Execution System* (Sistema de Execução de Manufatura)

PLC

*Programmable Logic Controller* (Controlador Lógico Programável)

OEM

*Original Equipment Manufacturer* (Fabricante Original do Equipamento)

API

*Application Programming Interface* (Interface de Programação de Aplicações)

SCADA

*Supervisory Control and Data Acquisition* (Supervisão e Aquisição de Dados)

IIoT

*Industrial Internet of Things* (Internet das Coisas Industrial)

RTU

*Remote Terminal Unit* (Unidade Terminal Remota)

E/S

*Input/Output, referring to control signals and sensors connected to the system* (Entrada/Saída, referente aos sinais de controle e sensores conectados ao sistema)

IHM

Interface Homem-Máquina, similar term to HMI (*Human-Machine Interface*) (Termo similar a HMI (Interface Homem-Máquina))



# 1. Introdução

## 1.1 Contextualização do Tema

Existe uma lacuna significativa no que diz respeito à digitalização de máquinas, processos e operações no setor florestal. Muitas máquinas ainda operam com sistemas limitados, o que restringe a capacidade de realizar ações de manutenção preditiva e dificulta a monitorização remota das operações. Este atraso na digitalização compromete a competitividade do setor, gerando a necessidade de soluções tecnológicas que integrem controlo e automação de forma eficiente.

Neste contexto, destaca-se o projeto de investigação SMARTCUT (Diagnóstico e Manutenção Remota e Simuladores para Formação de Operação e Manutenção de Máquinas Florestais), liderado pela empresa Cutplant Solutions S.A., detentora da marca VICORT. Este projeto visa combinar tecnologias físicas e digitais, permitindo a ligação remota entre operadores de máquinas florestais e técnicos de manutenção preditiva, facilitando a monitorização, a deteção de anomalias e promovendo a teleassistência e o treino de operadores através de simuladores.

Além disso, o controlo dos acionamentos do harvester é uma questão crucial para a eficiência da operação. O harvester é controlado por meio de um sistema de joystick, como o HJ60, que permite ao operador ajustar os movimentos e a velocidade da lâmina com precisão. A integração deste controlo com o sistema CAN Bus proposto neste projeto facilita a leitura dos comandos do joystick, otimizando a operação da máquina e garantindo maior precisão no corte e no desempenho geral da operação.

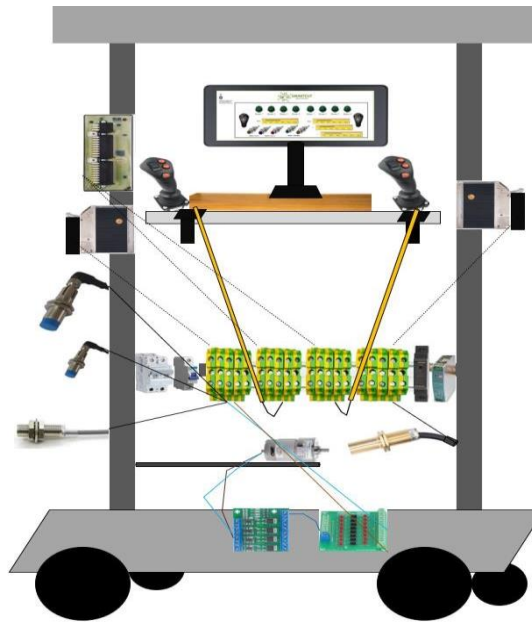
Esta funcionalidade é essencial, pois qualquer erro ou atraso no controlo da lâmina pode causar danos ao equipamento e comprometer a segurança.

Deste modo, o desenvolvimento de uma plataforma modular baseada em CAN Bus para a monitorização e controlo das operações do harvester, incluindo a interação com um conjunto de sensores e o controlo da lâmina através do joystick HJ60, representa uma solução avançada para as ações de modernização do setor florestal. Ao integrar o controlo da máquina com a interface HMI (Human Machine Interface), o operador pode monitorizar e ajustar as operações em tempo real, maximizando a eficiência e a segurança. O projeto contribui, assim, para a implementação de soluções inteligentes alinhadas com o conceito de Floresta 4.0.

A Figura 1 ilustra o que é um harvester e, por baixo, está colocado o logótipo do projeto SMARTCUT, que serviu de base neste trabalho.[10][12]

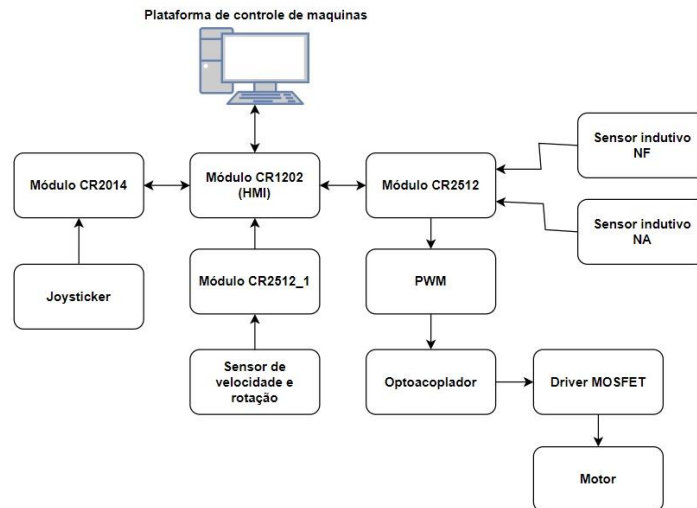


**Figura 1-** Ilustração de um harvester e logotipo do projeto SMARTCUT. [3]



**Figura 2-** Ilustração 2D do protótipo desenvolvido

### 1.1.1 Diagrama em bloco do sistema de monitorização e controlo de um harvester



**Figura 3–** Diagrama do sistema.

Este projeto utiliza o CODESYS como plataforma para programar o controlador central, o módulo CR1202, e associar os módulos CR2512 e CR2014 na rede CAN dentro do controlador, além de os configurar. O objetivo é validar as principais funcionalidades de um harvester moderno.

#### **Plataforma CODESYS**

O CODESYS é uma ferramenta de desenvolvimento de software usada para programar. No caso deste protótipo, o CODESYS é executado no computador e é responsável por integrar todos os módulos e dispositivos do sistema.

#### **Módulo CR1202 e HMI (Interface Homem-Máquina)**

A interface HMI e o controlador, programados no CODESYS, são fundamentais para o controlo e monitorização. Todo o sistema é controlado por esta interface e, através dela, o operador interage em tempo real com o sistema. A HMI exibe informações vindas dos sensores conectados ao sistema, fornecendo ao operador dados como a monitorização da velocidade e rotação do motor (vindo do sensor de Hall) e o estado dos sensores indutivos.

#### **Joystick e Módulo CR2014**

Para o controlo manual, o operador utiliza um joystick, ligado ao sistema através do Módulo CR2014. Este joystick permite que o operador simule as principais funções de manuseio do harvester, como aumentar e diminuir a velocidade do motor. O joystick

envia estes comandos ao controlador, que os envia para o módulo CR2512, responsável pelo manuseio do motor da lâmina de corte.

### **Módulo CR2512 e Optoacoplador**

O Módulo CR2512 é responsável por gerar o sinal de controlo PWM (Modulação por Largura de Pulso) com 24V, utilizado para ajustar a velocidade e o torque do motor. Este sinal PWM, no entanto, precisa ser convertido para 5V para que o driver MOSFET possa controlar efetivamente o motor. Isto é feito através de um optoacoplador, que além de redimensionar a tensão de 24V para 5V, garante o isolamento elétrico entre os circuitos de controlo e potência.

### **Driver MOSFET e Motor**

O driver MOSFET recebe o sinal PWM convertido e ajusta a quantidade de energia enviada ao motor. O motor é o principal atuador do sistema.

### **Monitorização com Sensores**

Além do controlo, o protótipo também oferece monitorização em tempo real das condições do sistema. Os sensores indutivos instalados na máquina são utilizados para detetar a presença de objetos metálicos (por exemplo, para garantir que os braços de corte estão posicionados corretamente ou que não há interferências no caminho). Estes sensores estão conectados ao Módulo CR2512, que envia os dados ao CODESYS para serem exibidos na interface HMI.

O sistema também possui um sensor de velocidade e rotação, que monitoriza a rotação do motor. Embora o protótipo opere em malha aberta (sem ajuste automático com base nas leituras dos sensores), os dados fornecidos pelos sensores são exibidos na HMI para que o operador possa monitorizar o desempenho do motor e fazer ajustes manuais, se necessário.

### **Sistema em Malha Aberta**

Embora o protótipo integre uma série de sensores para monitorizar a operação do harvester, ele não opera em malha fechada. Isto significa que o sistema não utiliza os dados dos sensores para ajustar automaticamente os parâmetros de controlo. No entanto, o operador tem acesso a todos os dados de monitorização e pode ajustar manualmente a operação do motor ou outros atuadores com base nas informações exibidas na interface HMI

## **1.2 Justificação e Relevância**

A relevância deste projeto está diretamente ligada à necessidade de maior eficiência operacional e redução de custos no setor florestal. A digitalização das máquinas móveis, como o harvester, é essencial para garantir maior produtividade e competitividade. A

possibilidade de monitorizar remotamente e em tempo real as operações dessas máquinas não só melhora a resposta a falhas, mas também permite a implementação de manutenção preditiva, reduzindo significativamente o tempo de inatividade e aumentando a vida útil dos equipamentos.

Com a adoção de um sistema baseado em CAN Bus, este projeto visa proporcionar uma solução eficaz para integrar a deteção, controlo e monitorização de máquinas florestais, contribuindo para a modernização do setor e alinhando-se com as diretrizes da Floresta 4.0.

Este trabalho, portanto, visa preencher uma lacuna existente no mercado de máquinas florestais, onde muitas soluções atuais são limitadas em termos de integração de sensores, módulos e sistemas de controlo em rede.

### **1.3 Problema de Investigação**

Um dos problemas de investigação que este trabalho pretende abordar é: “Como desenvolver uma plataforma modular que integre diferentes módulos eletrónicos para monitorizar e controlar eficientemente as operações de um harvester em tempo real, com foco nos sensores e atuadores da máquina?”

A resposta a essa questão envolve o desenvolvimento de um sistema que utilize uma rede CAN Bus, que permita a comunicação entre os diversos módulos eletrónicos da máquina e possibilite o controlo e monitorização das suas operações de forma eficiente.

O setor florestal enfrenta diversos desafios relacionados com a digitalização e modernização das máquinas, principalmente em termos de monitorização remota, manutenção preditiva e controlo eficiente das operações. Entre os principais problemas identificados destacam-se a dificuldade em realizar a monitorização em tempo real das máquinas e a falta de sistemas eficazes para a deteção de anomalias que podem comprometer o desempenho e a durabilidade dos equipamentos.

Este projeto aborda a implementação de um sistema modular para monitorização e controlo de máquinas florestais, como o harvester, inserido no contexto da Floresta 4.0. Dentre os principais desafios está a integração de uma rede CAN Bus que permita a comunicação eficiente entre os módulos eletrónicos e a interface homem-máquina (HMI). O sistema proposto deve garantir a capacidade de interagir com diferentes sensores e atuadores utilizados nas máquinas.

Outro ponto fundamental é o controlo de atuadores acionáveis por PWM, que é realizado através do joystick HJ60. Este dispositivo permite que o operador ajuste os movimentos de um motor e controle a sua velocidade e potência. No entanto, integrar esse controlo de forma eficiente ao sistema CAN Bus, de modo que o joystick HJ60 possa enviar comandos precisos em tempo real, é um dos principais problemas a ser

abordado neste trabalho. A comunicação entre o joystick e os outros módulos da máquina precisa ser ágil e fiável para garantir a segurança e a eficiência das operações.

Portanto, a principal questão de investigação deste trabalho pode ser formulada da seguinte forma:

- Como desenvolver uma solução modular para a monitorização, controlo e manutenção de máquinas florestais, utilizando uma rede CAN Bus, que integre o controlo eficiente de atuadores através do joystick HJ60 e permita a deteção antecipada de anomalias?

## 1.4 Objetivos do Trabalho

O objetivo deste trabalho é desenvolver uma solução modular para monitorização e controlo em tempo real de um harvester, utilizando a rede CAN Bus, com foco na experimentação de sensores e atuadores.

### 1.4.1 Objetivo Geral

Desenvolver uma solução modular de monitorização, controlo e simulação para máquinas florestais, com foco no harvester, inserida no contexto da Floresta 4.0. A solução visa:

- Monitorizar remotamente as operações da máquina, permitindo a visualização em tempo real do seu estado operacional e a deteção de anomalias que possam comprometer o seu funcionamento.
- Controlar a lâmina de corte do harvester de forma eficiente e precisa, utilizando o joystick HJ60 para realizar o comando do motor que regula a operação da lâmina, garantindo o ajuste fino de velocidade e potência durante as operações florestais.
- Reduzir custos de manutenção através da implementação de manutenção preditiva, baseada na análise de dados em tempo real e na deteção precoce de falhas.
- Criar um simulador físico e um gémeo digital, permitindo recriar as operações da máquina em ambiente virtual, tanto para o treino de operadores como para simulações de falhas e diagnósticos remotos.

### 1.4.2 Objetivos Específicos

Os objetivos específicos deste projeto incluem:

- Desenvolver e integrar uma rede CAN Bus para a comunicação entre os módulos eletrónicos do sistema, permitindo a monitorização eficiente das máquinas florestais e a gestão dos dados em tempo real.

- Implementar o controlo da lâmina de corte do harvester a partir do joystick HJ60, permitindo que o operador ajuste de forma precisa a velocidade e a potência da lâmina durante a operação, otimizando a eficiência do corte.
- Desenvolver uma interface homem-máquina (HMI), utilizando o módulo IFM CR1202, para garantir que os operadores possam visualizar o estado da máquina, receber alertas de anomalias e realizar ajustes operacionais em tempo real.
- Implementar um sistema de deteção de objetos metálicos para prevenir danos à lâmina de corte e melhorar a segurança operacional.
- Desenvolver um simulador físico e um gémeo digital que possibilitem a simulação do comportamento do harvester em condições reais e em cenários de falha, promovendo o treino remoto de operadores e técnicos.
- Realizar testes de validação do sistema, garantindo que a monitorização, o controlo da lâmina de corte via joystick HJ60, e as demais funcionalidades operem conforme os requisitos especificados.

## 1.5 Metodologia Resumida

Para o desenvolvimento deste projeto, foi adotada uma metodologia baseada em cinco etapas principais:

1. **Levantamento de Requisitos e Planeamento:** Identificação dos requisitos funcionais e não funcionais do sistema, incluindo a necessidade de monitorização remota e o controlo da lâmina de corte do harvester via joystick HJ60. Definiu-se também a infraestrutura necessária, como a rede CAN Bus e os módulos eletrónicos que integrariam o sistema.
2. **Desenvolvimento da Rede CAN Bus e Integração de Módulos:** Criação da rede CAN Bus para permitir a comunicação eficiente entre os módulos. Os módulos CR2014, CR2512 e CR1202 foram integrados ao sistema, incluindo o joystick HJ60, que comanda o motor da lâmina de corte do harvester.
3. **Implementação do Controlo da Lâmina de Corte:** Configuração do joystick HJ60 para controlar a operação da lâmina de corte, ajustando a sua velocidade e potência. Este controlo foi integrado ao sistema de monitorização em tempo real, permitindo ao operador monitorizar o desempenho e realizar ajustes precisos durante a operação.
4. **Desenvolvimento da Interface Homem-Máquina (HMI):** Utilização da HMI IFM CR1202 para criar uma interface de comunicação entre o operador e o sistema, permitindo visualizar o estado da máquina, controlar a lâmina de corte e receber alertas de anomalias em tempo real.

**5. Desenvolvimento do Simulador Físico e Testes:** Desenvolvimento de um simulador físico para recriar as operações do harvester em ambiente virtual, com testes práticos para validar a funcionalidade do sistema, incluindo o controle da lâmina de corte e o desempenho da rede CAN Bus.

## 1.6 Estrutura do Trabalho

Este relatório de projeto está organizado em seis capítulos que detalham o desenvolvimento da solução de monitorização e controlo de máquinas florestais. A estrutura completa do trabalho é a seguinte:

- **Capítulo 1 - Introdução:** Apresenta o contexto geral da pesquisa, relevância do conceito de Floresta 4.0 e desafios de digitalização do setor florestal. São discutidos problemas de pesquisa, como a monitorização remota e a deteção de anomalias, assim como a justificativa do projeto e sua importância.
- **Capítulo 2 - Revisão da Literatura:** Explora tecnologias e abordagens usadas em monitorização e controlo de máquinas móveis no setor florestal e noutras indústrias, incluindo o protocolo CAN Bus, interfaces HMI, sistemas de monitorização remota e joysticks industriais.
- **Capítulo 3 - Metodologia:** Detalha métodos e processos de desenvolvimento do sistema, incluindo a escolha do CAN Bus, implementação dos módulos eletrónicos e integração do joystick HJ60.
- **Capítulo 4 - Desenvolvimento do Sistema:** Descreve o processo de desenvolvimento, funcionalidades dos módulos CR2014, CR2512 e CR1202, e integração do joystick HJ60.
- **Capítulo 5 - Resultados e Discussões:** Apresenta e discute os resultados da implementação do sistema, incluindo monitorização remota, deteção de anomalias, controlo via joystick HJ60, e resposta a falhas simuladas.
- **Capítulo 6 - Conclusão e Trabalhos Futuros:** Avalia se os objetivos foram atingidos e os benefícios da solução para o setor florestal, discutindo também futuras melhorias e avanços tecnológicos.
- **7 - Referências e Anexos:** Apresentação das tabelas de mapeamento das ligações físicas dos módulos do sistema, esquemas elétricos de cada módulo e esquema elétrico geral do sistema.

## 2. Fundamentação Teórica

### 2.1 Indústria 4.0 e Floresta 4.0

#### 2.1.1 Conceito de Indústria 4.0

A Indústria 4.0 refere-se à quarta revolução industrial, um termo que surgiu na Alemanha por volta de 2011, e que se baseia na interconexão digital de máquinas, sistemas e processos de produção. Esta nova fase da revolução industrial é caracterizada pela automação inteligente, digitalização em massa e o uso de tecnologias emergentes como a Internet das Coisas (IoT), Inteligência Artificial (IA), Big Data, robótica avançada, computação em nuvem e manufatura aditiva. Na Indústria 4.0, todos os componentes de um sistema produtivo são interligados, formando redes inteligentes que podem tomar decisões de forma autónoma, baseadas em dados recolhidos em tempo real. [1]

#### 2.1.2 Aplicação no Setor Florestal - Floresta 4.0

O conceito de Indústria 4.0 foi expandido para diferentes setores da economia, incluindo o setor florestal, dando origem ao termo Floresta 4.0. Esta abordagem tem como objetivo modernizar e automatizar processos no setor florestal e na operação de máquinas, como harvesters, forwarders e skidders, através da utilização de tecnologias avançadas. Máquinas florestais inteligentes, equipadas com sensores e sistemas de comunicação, permitem a monitorização remota, diagnóstico em tempo real e controlo de operações à distância, facilitando a gestão eficiente de florestas e aumentando a produtividade, enquanto se reduzem custos operacionais e impactos ambientais. [2]

#### 2.1.3 Contextualização do Projeto SMARTCUT

O projeto SMARTCUT surge no contexto da Floresta 4.0, desenvolvido pela empresa Cutplant Solutions S.A., com foco na digitalização de máquinas florestais. O projeto procura implementar soluções de monitorização remota, diagnóstico e manutenção assistida, utilizando tecnologias de simulação e formação para operadores de máquinas. Este projeto propõe-se a combinar tecnologias físicas e digitais, de modo a promover a interação remota entre operadores, máquinas e técnicos, permitindo maior controlo e eficiência no campo. [12]

### 2.2 Tecnologias de Monitorização e Controlo de Máquinas Móveis

#### 2.2.1 Monitorização Remota e Manutenção Preditiva

O conceito de monitorização remota envolve a utilização de redes de sensores e sistemas de comunicação para recolher dados operacionais de uma máquina em tempo real, sem a necessidade da presença física de um operador. No setor florestal, isso permite que máquinas como o harvester sejam monitorizadas continuamente,

proporcionando uma visão clara do desempenho do equipamento, identificando anomalias e facilitando ações preditivas de manutenção. A manutenção preditiva baseia-se na análise dos dados gerados pela monitorização, prevenindo possíveis falhas antes que ocorram, reduzindo o tempo de inatividade e custos com reparos inesperados.

As máquinas móveis modernas integram sensores para medir diversos parâmetros, como a temperatura do motor, pressão do sistema hidráulico, velocidade de operação e condição dos principais componentes mecânicos. Estes dados são transmitidos para uma central de monitorização, onde são processados e analisados, ajudando a prever falhas e a otimizar a manutenção.

### 2.2.2 Controlo de Máquinas Móveis

O controlo de máquinas móveis como os harvesters é feito através de interfaces de controlo que comandam diversos sistemas da máquina, como o motor, a lâmina de corte, o braço mecânico e outros componentes. Estes controlos são muitas vezes operados por joysticks industriais, que são interfaces simples e intuitivas para os operadores. O joystick HJ60, por exemplo, é uma peça-chave no controlo da lâmina de corte do harvester, permitindo ajustes finos da velocidade de corte e de seu movimento, o que é fundamental para a eficiência da operação e a preservação da integridade da máquina.

## 2.3 Protocolo CAN Bus

### 2.3.1 Introdução ao Protocolo CAN Bus

O protocolo CAN Bus (Controller Area Network) é um padrão de comunicação amplamente utilizado em sistemas automóveis e de máquinas pesadas, que permite que diversos microcontroladores e dispositivos comuniquem entre si sem a necessidade de um computador central. Este protocolo foi desenvolvido para proporcionar uma comunicação eficiente, robusta e económica entre diferentes módulos de uma máquina, como motores, sensores e atuadores.

As principais características do CAN Bus incluem:

- **Redução de cablagem:** Todos os dispositivos compartilham o mesmo barramento de comunicação, reduzindo drasticamente a quantidade de cabos necessários em comparação com sistemas ponto a ponto.
- **Alta eficiência:** O protocolo é otimizado para transmitir dados com alta velocidade e baixo tempo de resposta, ideal para sistemas críticos de tempo real, como o controlo de máquinas.
- **Resiliência:** O CAN Bus possui mecanismos de correção de erros que aumentam a sua fiabilidade, especialmente em ambientes adversos, como os encontrados em operações florestais. [4]

### 2.3.2 CANopen

O CANopen é um protocolo de alto nível baseado no CAN Bus, amplamente utilizado em aplicações industriais. Foi desenhado para gerir dispositivos distribuídos em sistemas de automação, oferecendo uma estrutura para a configuração, monitorização e controlo de diferentes dispositivos, como sensores e atuadores.

Neste projeto, o CANopen é usado para integrar os módulos CR2014, CR2512 e CR1202 na rede CAN Bus do harvester, permitindo que todos os dispositivos trabalhem de forma coordenada e que os dados sejam processados de maneira eficiente. [4][5]

## 2.4 Interfaces Homem-Máquina (HMI)

### 2.4.1 Importância da HMI no Controlo de Máquinas

As Interfaces Homem-Máquina (HMI) desempenham um papel fundamental na operação e monitorização de máquinas industriais. Elas permitem que os operadores tenham uma visão clara dos parâmetros operacionais e realizem ajustes no sistema de forma intuitiva. A HMI é a ligação entre o operador e o sistema, oferecendo feedback em tempo real sobre o desempenho da máquina e permitindo uma interação direta com os controlos da mesma, como o controlo da lâmina de corte no caso dos harvesters.

### 2.4.2 Aplicação do IFM CR1202

O módulo IFM CR1202 é utilizado como a HMI neste projeto, proporcionando uma interface gráfica para o operador do harvester. Ele exibe dados recolhidos pelos diversos sensores da máquina, como a velocidade de corte, pressão dos sistemas hidráulicos e o desempenho geral da máquina, permitindo ao operador tomar decisões informadas em tempo real. O CR1202 também facilita a deteção precoce de falhas e a configuração de parâmetros do sistema.

## 2.5 Joystick HJ60 e Controlo da Lâmina de Corte

### 2.5.1 Tecnologia de Joysticks Industriais

O joystick HJ60 é uma interface de controlo altamente precisa e durável, amplamente utilizado em máquinas móveis, como os harvesters. Este joystick foi projetado para oferecer controlo fino sobre a lâmina de corte, que é uma parte crítica da operação do harvester. Através do controlo proporcional, o operador pode ajustar a velocidade e a potência da lâmina, garantindo cortes eficientes e evitando sobrecargas no sistema. [8]

### 2.5.2 Integração do HJ60 no Sistema

No contexto deste projeto, o HJ60 está integrado ao sistema de controlo via CAN Bus, onde os seus comandos são interpretados pelo módulo CR2014, que regula o

motor da lâmina de corte. O controlo é realizado utilizando PWM (Modulação por Largura de Pulso), o que permite um controlo suave e preciso da potência da lâmina, melhorando a eficiência da operação e a segurança do operador. [9][6]

## **2.6 Gémeo Digital e Simuladores para Operação e Manutenção**

### **2.6.1 Conceito de Gémeo Digital**

O conceito de gémeo digital refere-se à criação de uma réplica digital de um sistema físico, que permite monitorizar, simular e otimizar o seu funcionamento em tempo real. No setor florestal, o gémeo digital de um harvester possibilita a simulação de diferentes cenários operacionais e a análise do desempenho da máquina em condições reais, permitindo que operadores e técnicos treinem num ambiente virtual seguro. [9]

### **2.6.2 Simuladores de Máquinas Móveis**

A utilização de simuladores de máquinas móveis para treinar operadores e técnicos de manutenção é uma prática cada vez mais comum no setor florestal. Estes simuladores permitem que os operadores pratiquem o controlo da máquina, incluindo a operação da lâmina de corte, sem os riscos associados ao treino em campo. No contexto deste projeto, o simulador físico recria o ambiente de trabalho da máquina, enquanto o gémeo digital oferece dados em tempo real para simular falhas e comportamentos dinâmicos da máquina.

## **2.7 Resumo da Revisão**

Esta revisão da literatura destacou as principais tecnologias e inovações aplicáveis ao desenvolvimento do sistema de monitorização e controlo do harvester, com foco no projeto SMARTCUT. Ao combinar os conceitos da Indústria 4.0, monitorização remota, controlo por joystick e a utilização de gémeos digitais, o projeto proporciona soluções inovadoras que atendem às necessidades actuais do setor florestal, otimizando a operação e manutenção de máquinas.

## **3. Metodologia**

### **3.1 Descrição Geral do Sistema**

O projeto centra-se no desenvolvimento de uma plataforma modular de monitorização e controlo de máquinas florestais, especificamente o harvester, utilizando uma rede CAN Bus que integra módulos eletrónicos de entrada/saída para sensores e atuadores, com uma interface homem-máquina (HMI) para visualização e operação. O controlo da lâmina de corte é realizado através do joystick HJ60, com sinais PWM. A metodologia adoptada foi desenhada para garantir a operação robusta do

sistema, simplificando a manutenção e otimizando o desempenho da máquina no ambiente florestal.

### 3.1.1 Arquitetura do Sistema

O sistema foi concebido para ser modular e expansível, permitindo a adição de mais sensores e atuadores conforme as necessidades. A arquitetura envolve os seguintes componentes:

#### **Joystick Chengong HJ60**

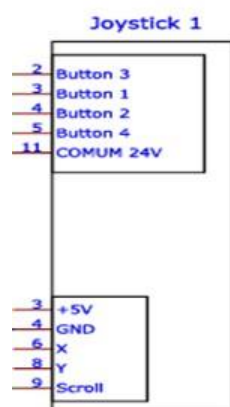


**Figura 4 - Joystick Chengong HJ60[13][14]**

O joystick HJ60 da Chengong, ilustrado na Figura 4, é um dispositivo industrial utilizado para operar máquinas e equipamentos. No projeto, são utilizados dois joysticks HJ60, que oferecem várias configurações e características adaptáveis às necessidades específicas de diferentes aplicações, como o controlo do direcionamento do braço de uma máquina de cortar árvores. O joystick HJ60 possui 3 botões na parte frontal, 1 botão na parte traseira, um scroll (rolete) e permite o controlo dos eixos X e Y, facilitando a operação e a precisão do braço da máquina de corte. [8]

Baseando-se nas suas ligações físicas, temos:

## Ligações Elétricas



**Figura 5** – Ligações elétricas do joystick HJ60

A imagem apresentada na Figura 5 detalha as ligações elétricas do joystick HJ60. A Tabela 1 resume as ligações do joystick.

PINO	IDENTIFICAÇÃO	FUNÇÃO
3	+5V	Alimentação do joystick (5V)
4	GND	Massa ( <i>Ground</i> – 0V)
6	Eixo X	Entrada para acionamento do movimento do joystick no eixo X
8	Eixo Y	Entrada para acionamento do movimento do joystick no eixo Y
9	Roleta	Entrada para o sinal do rolete ( <i>scroll</i> )

**Tabela 1** - Ligações elétricas do joystick HJ60 – conector de 5 pinos.

**Mapa de ligações dos dois joysticks:****Joystick\_1**

1. O pino 3 - 5V foi ligado ao barramento de 5V, e o pino 4 - GND foi ligado ao barramento de GND da fonte de alimentação.
2. O rolete foi ligado no canal 5 (CH5) analógico do módulo CR2014.

Nota: Por falta de canais analógicos disponíveis, não houve mapeamento dos eixos X e Y do Joystick\_1.

**Joystick\_2**

1. O pino 3 - 5V foi ligado ao barramento de 5V, e o pino 4 - GND foi ligado ao barramento de GND da fonte de alimentação.
2. O pino 6 - eixo X foi ligado no canal 13 (CH13) analógico do módulo CR2014.
3. O pino 7 - eixo Y foi ligado no canal 6 (CH6) analógico do módulo CR2014.
4. O rolete foi ligado no canal 6 analógico do módulo CR2014.

<b>PINO</b>	<b>IDENTIFICAÇÃO</b>	<b>FUNÇÃO</b>
<b>2</b>	Botão 3	Entrada do sinal para o Botão 3
<b>3</b>	Botão 1	Entrada do sinal para o Botão 1
<b>4</b>	Botão 2	Entrada do sinal para o Botão 2
<b>5</b>	Botão 4	Entrada do sinal para o Botão 4
<b>11</b>	Comum 24V	Alimentação comum de 24V para os botões

**Tabela 2** – Ligações elétricas do joystick HJ60 - conector de 7 pinos.

**Mapa de ligações dos botões dos dois joysticks:****Joystick\_1**

1. O pino 2 - Botão 3 foi ligado ao canal 3 (CH3) binário do módulo CR2014 (PlugX1).
2. O pino 3 - Botão 1 foi ligado ao canal 1 (CH1) binário do módulo CR2014 (PlugX1).
3. O pino 5 - Botão 4 foi ligado ao canal 4 (CH4) binário do módulo CR2014 (PlugX1).

4. O pino 11 - comum foi ligado ao pino 2 +UB do módulo CR2014 (PlugX1).

### **Joystick\_2**

1. O pino 3 - Botão 1 foi ligado ao canal 7 (CH7) binário do módulo CR2014 (PlugX1).
2. O pino 4 - Botão 2 foi ligado ao canal 8 (CH8) binário do módulo CR2014 (PlugX1).
3. O pino 5 - Botão 4 foi ligado ao canal 10 (CH10) binário do módulo CR2014 (PlugX2).
4. O pino 11 - comum foi ligado ao pino 4 +UB do módulo CR2014 (PlugX1).

Uma vez conhecidas as conexões elétricas do joystick HJ60, a implementação no projeto utilizando o módulo CR2014 tornou-se simples. Abordarei sobre este módulo brevemente a seguir.

### **Cabo de adaptador para o joystick**



**Figura 6** – cabo de adaptador de joystick.[15]

O cabo adaptador ilustrado na imagem possui 5 pinos e mede 35 cm de comprimento. Ele é geralmente utilizado em joysticks do tipo HJ60 para conectar os controles ao sistema principal. O HJ60 é um joystick industrial, aplicado em máquinas pesadas ou sistemas de controle, onde este cabo conecta o joystick à interface de controle, transmitindo sinais de movimento (como direção ou pressão) para controlar os equipamentos. Os diferentes fios coloridos representam funções variadas, como alimentação de energia, sinais de comando ou terra (GND), o que permite que o joystick funcione corretamente e se comunique com o dispositivo controlado.

### **Fontes de Alimentação**

Para o projeto foram utilizadas duas fontes de alimentação, uma de 24V e outra de 5V.

### Fonte de 24V



Figura 7 - Fonte de alimentação EDR-120.[40]

A série EDR-120 é composta por fontes de alimentação com uma única saída, oferecendo modelos com diferentes tensões de saída, como 12V e 24V. Estas fontes são projetadas para funcionar em várias tensões de entrada (90-264VAC ou 127-370VDC), o que as torna versáteis para diferentes aplicações industriais. A fonte usada é de 120 W e alimenta os módulos CR2512, CR2014 e o motor DC ZGA42RH(A6249). Para garantir segurança e durabilidade do equipamento, estas fontes incluem proteções contra sobretensão, sobrecarga, curto-circuito e sobreaquecimento.

### Ligações Elétricas

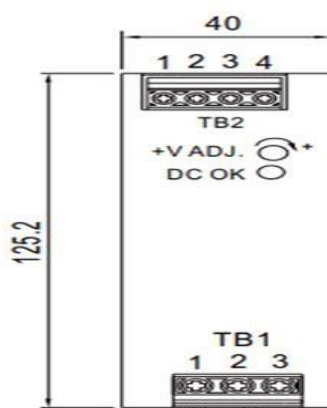


Figura 8 - Ligações elétrica da fonte de alimentação EDR-120

Na Figura 8 são mostradas as ligações da fonte de alimentação. Abaixo, encontram-se tabelas detalhadas sobre as ligações.

### **Pin No. Assignment (TB1)**

<b>Pin No.</b>	<b>Assignment</b>
<b>1</b>	FG (Terra)
<b>2</b>	AC/L (Fase)
<b>3</b>	AC/N (Neutro)

**Tabela 3** - Terminal Pin No. Assignment (TB1)

### **Terminal Pin No. Assignment (TB2)**

<b>Pin. No</b>	<b>Assignment</b>
<b>1, 2</b>	DC OUTPUT -V (GND)
<b>3, 4</b>	DC OUTPUT +V (24V)

**Tabela 4** - Terminal Pin No. Assignment (TB2)

### **Mapa de Ligações do EDR-120 TB2**

#### **EDR-120**

1. O pino 2 foi utilizado como GND para todo o sistema.
2. O pino 4 foi usado para fornecer os 24V a todos os módulos IFM do sistema.

## Fonte de 5V



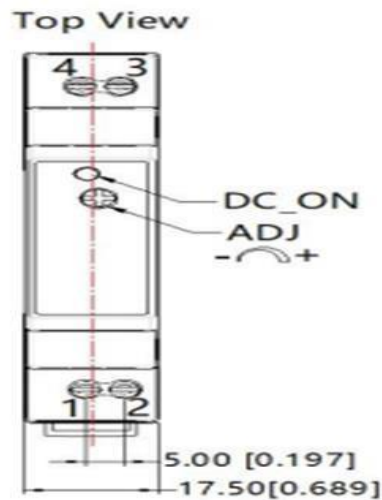
**Figura 9** – Fonte de 5V. [16]

## Características da Fonte de Alimentação de 24V

- Potência de saída: 12W.
- Corrente nominal: 2,4A.
- Faixa de ajuste de tensão: 4,5V a 5,5V.
- Eficiência: 80% a 230VAC.
- Capacitância de carga máxima: 2000 $\mu$ F.
- Proteções: contra curto-circuito, sobrecorrente e sobretensão.

Esta fonte pode ser aplicada numa ampla gama de equipamentos industriais, sendo montada em trilhos DIN para economizar espaço e facilitar a instalação. No projeto, é utilizada para alimentar os joysticks e o driver do motor DC.

## Ligações Elétricas



**Figura 10** - Ligações elétrica da fonte de alimentação de 5

A tabela abaixo contém informações adicionais sobre as ligações elétricas da fonte de 5V, com a descrição dos pinos.

Pin. No	Descrição
1	Entrada AC L (Fase)
2	Entrada AC N (Neutro)
3	Saída +V (Positiva) (5V)
4	Saída -V (Negativa) (GND)

**Tabela 5** - Pin-Out (ligações elétricas) da fonte de 5V.

### Mapa de Ligações da Fonte de 5V

#### Fonte de 5V

1. O pino 3 foi utilizado para alimentar o joystick e o optoacoplador.
2. O pino 4 foi designado como GND comum para todo o sistema.

## Driver de Motor BTS7960 (H-Bridge)



Figura 11 - Driver de Motor BTS7960 (H-Bridge).[17]

O driver de motor BTS7960, ilustrado na Figura 10, é utilizado para controlar motores de corrente contínua (DC) através de uma ponte H integrada. No projeto, ele recebe um sinal PWM proveniente do canal 1 (CH1) do módulo CR2512, o que é essencial para garantir o controlo eficiente e seguro do motor ZGA42RH(A6249).

### Ligações e Funções dos Pinos

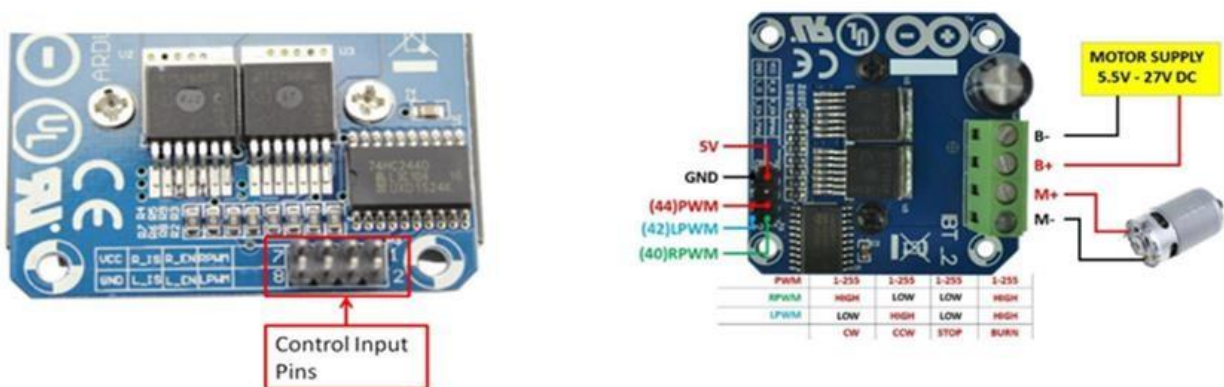


Figura 12 - Ligações e funções dos pinos do driver de Motor BTS7960 (H-Bridge).

Pino	Função	Descrição
1	RPWM	Sinal de nível ou PWM para frente (ativo alto)
2	LPWM	Sinal de nível ou PWM para trás (ativo alto)
3	R_EN	Habilita a condução para frente (ativo alto/baixo desabilita)
4	L_EN	Habilita a condução para trás (ativo alto/baixo desabilita)
5	R_IS	Saída de alarme de corrente lateral para frente
6	L_IS	Saída de alarme de corrente lateral para trás
7	VCC	Fonte de alimentação de +5V para microcontrolador
8	GND	Massa para microcontrolador

**Tabela 6** - Função dos Pinos de Entrada de Controle.

A Tabela 6 detalha as funções dos pinos necessárias para configurar e operar o sistema de controlo do motor, onde se define a direção, habilita ou desabilita os movimentos, monitoriza as condições de corrente e garante que o microcontrolador recebe a alimentação necessária. Este design permite flexibilidade e segurança no acionamento de motores DC.



**Figura 13** - Ligações da alimentação do motor (24V).

Pino	Função	Descrição
1	B+	Fonte de alimentação positiva do motor (6 ~ 27V DC)
2	B-	Fonte de alimentação negativa do motor (GND)
3	M+	Saída “positiva” para motor
4	M-	Saída “negativa” para motor

Tabela 7 – Ligações da saída de potência do driver.

A Tabela 7 descreve os pinos de alimentação de um motor DC. Os pinos B+ e B- ligam a fonte de alimentação ao motor, enquanto os pinos M+ e M- controlam a saída de corrente para o motor, determinando o seu funcionamento e a direção de rotação.

Nota: Devido a problemas de funcionamento do driver de motor BTS7960, este foi substituído pelo driver de MOSFET de 4 canais.

### Optoacoplador BUCC\_817\_8\_V1.0

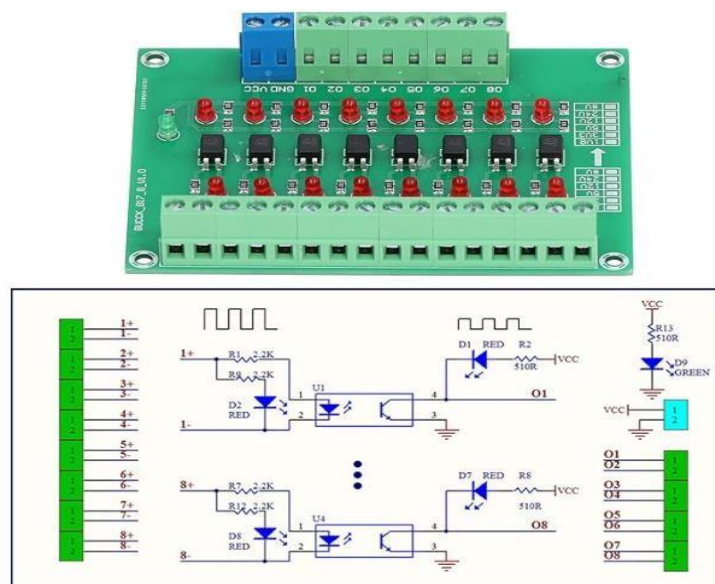


Figura 14 – Optoacoplador de 8 canais.[18]

Este módulo é um isolador ótico (ou optoacoplador) de 8 canais, usado principalmente para isolar sinais com diferentes tensões de alimentação. Esta funcionalidade é importante para evitar interferências ou danos aos circuitos de controlo (como um microcontrolador) resultantes de ruídos ou diferenças de tensão

nos circuitos de carga. O módulo recebe sinais de entrada e replica-os nos canais de saída com outros níveis de tensão, garantindo o isolamento elétrico. Desta forma, sinais de baixa tensão podem acionar dispositivos de maior potência sem que haja uma ligação elétrica direta entre ambos. Usado no sistema para redimensionar a tensão se 24V do PWM para de 5V.

Este módulo permite controlar dispositivos de maior tensão ou corrente (como relés, lâmpadas ou motores) usando sinais de baixa tensão, isolando eletricamente ambos os circuitos para maior segurança e confiabilidade.

<b>Pinos de Entrada</b>	<b>Função</b>	<b>Pinos de Saída</b>	<b>Função</b>
<b>1+ / 1-</b>	Canal 1 de controle	01	Saída isolada do Canal 1
<b>2+ / 2-</b>	Canal 2 de controle	02	Saída isolada do Canal 2
<b>3+ / 3-</b>	Canal 3 de controle	03	Saída isolada do Canal 3
<b>4+ / 4-</b>	Canal 4 de controle	04	Saída isolada do Canal 4
<b>5+ / 5-</b>	Canal 5 de controle	05	Saída isolada do Canal 5
<b>6+ / 6-</b>	Canal 6 de controle	06	Saída isolada do Canal 6
<b>7+ / 7-</b>	Canal 7 de controle	07	Saída isolada do Canal 7
<b>8+ / 8-</b>	Canal 8 de controle	08	Saída isolada do Canal 8
<b>VCC</b>	Alimentação (24V)	GND	GND
<b>0 COM</b>	Comum das saídas	--	--

**Tabela 8** – Ligações elétricas do optocoplador.

Este módulo permite controlar dispositivos de maior tensão ou corrente (como relés, lâmpadas ou motores) usando sinais de baixa tensão, isolando eletricamente ambos os circuitos para maior segurança e confiabilidade.

### **Mapa de Ligações do Optocoplador**

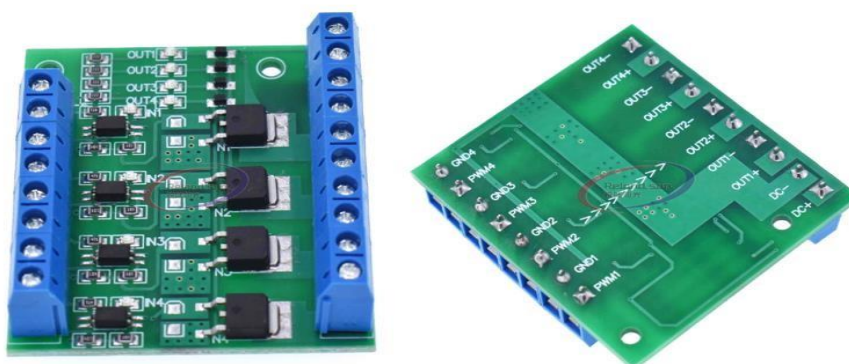
#### **Optocoplador**

1. O pino de entrada do optocoplador “-1” foi ligado ao pino 02 do canal 1 do módulo CR2512.

2. O pino de entrada do optoacoplador “+1” foi ligado ao pino 20 do canal 1 do módulo CR2512.
3. O pino de saída do optoacoplador “-1” foi ligado ao pino PWM2 do driver de MOSFET.
4. VCC foi ligado à fonte de 5V.

Nota: Este modelo DST 1RBP N (não inclui estrutura de montagem em calha DIN).  
Análise do modelo: DST (conversor digital) 1R (transmissão unidirecional) 8P (8 canais) N (saída NPN).

### ***Driver de MOSFET de 4 canais***



**Figura 15** – Placa de Potência com 4 canais MOSFET.[19]

O driver de MOSFET de 4 canais (Figura 14) é utilizado para o controlo de cargas de potência (como motores, aquecedores, LEDs de alta potência, etc.) através de sinais de baixa potência, como PWM (Modulação por Largura de Pulso) de um microcontrolador. Este módulo utiliza transístores MOSFET para comutar cargas de forma eficiente.

<b>Pinos</b>	<b>Função</b>	<b>Descrição</b>
<b>GND1, GND2, GND3, GND4</b>	GND para cada canal	Referência de GND para os sinais PWM
<b>PWM1, PWM2, PWM3, PWM4</b>	Entrada de controlo PWM	Sinal de controlo para comutação dos MOSFETs
<b>DC+</b>	Alimentação Positiva	Tensão de alimentação do lado da carga (geralmente Vcc)
<b>DC-</b>	Alimentação Negativa (GND)	GND da fonte de alimentação da carga
<b>OUT1+, OUT2+, OUT3+, OUT4+</b>	Saídas Positivas dos canais	Ligação para o terminal positivo da carga (controlado pelos MOSFETs)
<b>OUT1-, OUT2-, OUT3-, OUT4-</b>	Saídas Negativas dos canais	Ligação para o terminal negativo da carga

**Tabela 9** – Mapa de ligações do driver de potência

Este módulo é frequentemente usado em aplicações onde um microcontrolador ou outro sistema de controlo envia sinais de baixa potência para controlar dispositivos de maior potência. O sinal PWM recebido nos pinos de entrada (PWM1 a PWM4) é usado para comutar os MOSFETs correspondentes, que, por sua vez, controlam as cargas ligadas nas saídas (OUT1 a OUT4).

### **Mapa de ligações do MOSFET de 4 canais**

#### **MOSFET de 4 canais**

1. O pino PWM2 foi ligado ao pino 01 da saída do optoacoplador.
2. O pino GND2 foi ligado ao pino 02 do canal 1 do módulo CR2512.
3. O pino “-OUT2” foi ligado ao GND do motor.
4. O pino “+OUT2” foi usado como pino de alimentação do motor.
5. O “-DC” foi ligado ao pino 02 do canal 1 do módulo CR2512.
6. O “+DC” foi ligado ao pino 08 do canal 1 do módulo CR2512.

**Motor ZGA42RH(A6249)****Figura 16** - Motor ZGA42RH(A6249) de 24V. [20]

O motor ZGA42RH (A6249) foi utilizado no projeto para simular o funcionamento da lâmina de corte das árvores num abatedor florestal.

<b>Ligação Elétrica</b>	<b>Velocidade Admissível (RPM)</b>	<b>Corrente (A)</b>
<b>24V DC Direta</b>	200 - 500 RPM (dependendo da carga)	0.5A - 1.2A
<b>12V DC (meia tensão)</b>	100 - 250 RPM	0.25A - 0.6A
<b>Com controle PWM</b>	0 - 500 RPM (variável)	0 - 1.2A

**Tabela 10** – Infirmações relevantes do Motor DC**Direção:**

- A polaridade da conexão de energia determina a direção de rotação (CW: horário, CCW: anti-horário).

**Mapa de ligação do motor ZGA42RH (A6249)****ZGA42RH (A6249)**

1. O cabo de 24V foi ligado ao pino "+OUT2" do driver MOSFET.
2. O cabo de GND foi ligado ao pino "-OUT2" do driver MOSFET.

## Sensores Utilizados



Figura 17 – Sensor Indutivo [21]

Sensores de proximidade indutivos são dispositivos que detectam a presença de elementos metálicos sem a necessidade de contacto físico. Neste projeto, são utilizados para detetar o posicionamento da lâmina de corte. O sensor foi ligado a um dos módulos CR2512.

### Cabo do sensor M8

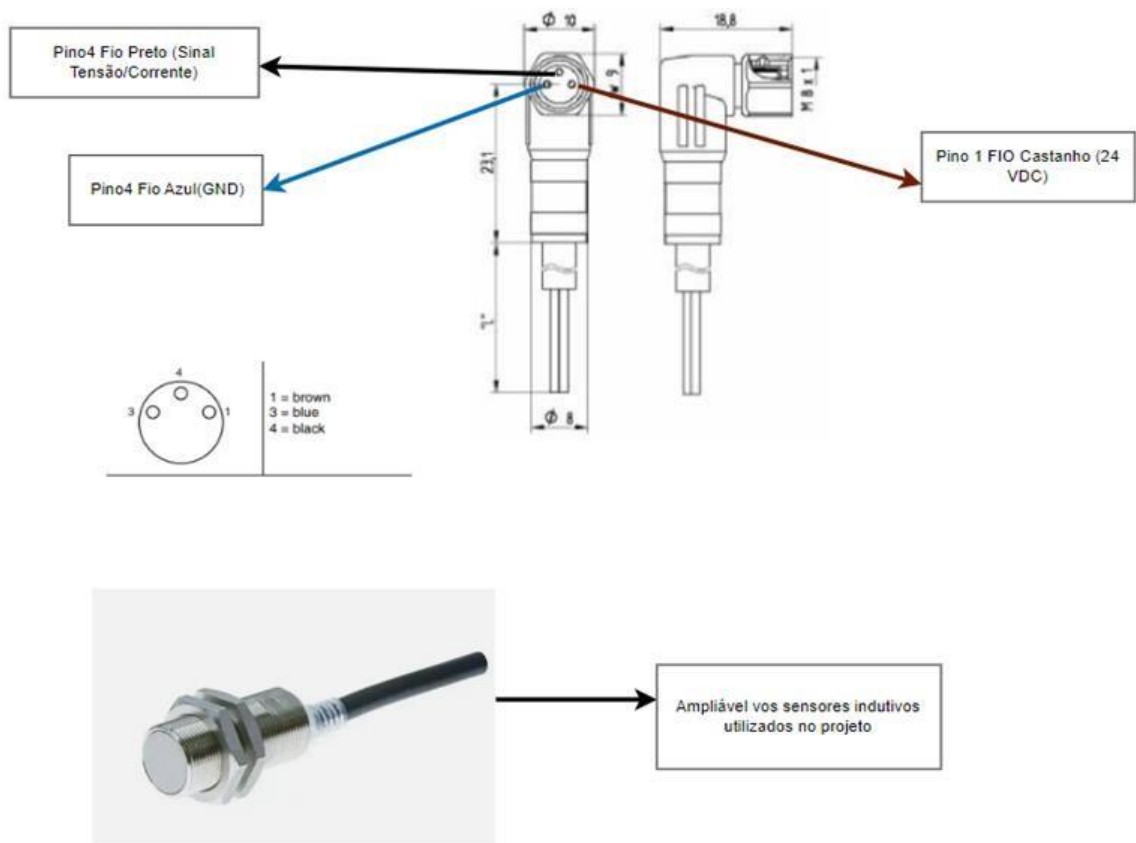


Figura 18 – Esquema de ligações do sensor indutivo através de ficha M8.

Os sensores indutivos deste caso de estudo vêm com uma ficha padrão do tipo M8 para ligação aos barramentos de sinal.

- **M8:** Conector de 3 ou 4 pinos.
- **Ligações de fios:**
  - Marrom (positivo): Ligado à fonte de alimentação +24V.
  - Azul (negativo): Ligado ao GND.
  - Preto (saída): Ligado à carga ou ao controlador.

O tipo de ligação é NPN NA, ou seja, a saída ativa apenas quando o objeto é detetado.

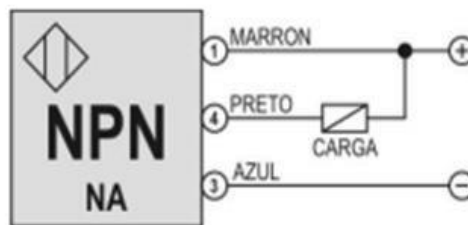


Figura 19 – Ligação NPN NA do sensor indutivo.[24]

### **Mapa de ligação do Sensor Indutivo NA**

#### **Sensor Indutivo NA**

1. O pino 1 – fio castanho foi ligado ao pino 43 no canal 9 (CH9) binário do módulo CR2512.
2. O pino 3 – fio preto foi ligado ao pino 47 no canal 9 (CH9) do módulo CR2512.
3. O pino 4 – fio azul foi ligado ao pino 06 no canal 9 (CH9) do módulo CR2512.

## Sensor indutivo NPN NF e Sensor de Hall



Figura 20 - Sensor indutivo normalmente fechado e Sensor de hall.[22][25]

Na Figura 20, temos um sensor hall utilizado para detetar a rotação ou velocidade de um objeto em movimento. No projeto, monitorizar a rotação da lâmina de corte. O cabo do sensor de velocidade possui a configuração esquematizada na Figura 20. Além disso, são utilizados dois sensores indutivos do tipo NPN NF, ligados a um cabo que permite escolher entre NPN NF e NPN NA. Como a ligação para ambos os tipos é idêntica (como mostrado na Figura 20), a escolha depende do sensor. Neste projeto, foram ligados dois sensores indutivos do tipo NF (normalmente fechado). Estes sensores retornam um valor lógico 0 quando próximos do metal e um valor lógico 1 quando distantes, ou seja, a saída é ativada quando o objeto não é detetado.

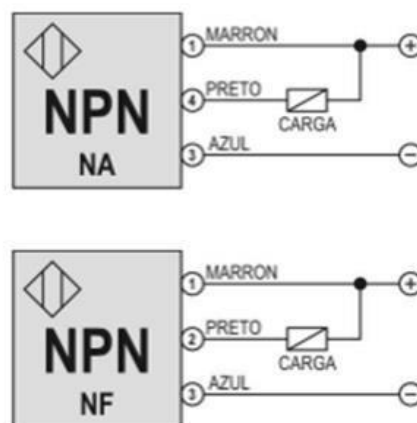


Figura 21 - Ligação NPN NA e NPN NF.[24]



Como ilustrado na Figura 23, o tipo de ligação segue o modelo padrão IEC, com a linha a) N.A. (Normalmente Aberto), como mostrado na tabela a seguir.

<b>Tipo de Ligação</b>	Pino -1 (Castanho)	Pino- 3 (Azul)	Pino-4 (Castanho)	Pino- 4 (Azul)	Pino -2 (Azul)
<b>DC 2-fio padrão</b>	+24VDC	0V	-	-	-
<b>Conexão para modelo padrão</b>	-	0V	+24VDC	0V	-
<b>Conexão IEC modelo padrão a) N.O. (Normalmente aberto) Tipo</b>	+24VDC	-	-	0V	-
<b>Conexão IEC modelo padrão b) N.C. (Normalmente fechado) Tipo</b>	+24VDC	0V	-	-	0V

**Tabela 11** – Tipos de ligações

**Nota:** Apenas os fios castanho e azul foram representados na tabela acima. Para cada tipo de ligação, os pinos de alimentação mudam de posição, e os pinos não representados são os ligados à carga ou ao controlador para sinal de saída.

### **Módulo CR2512**



**Figura 24** – Módulo de E/S remotas – IFM CR2512.[26]

O CR2512 é um módulo de E/S para máquinas móveis, com interface CAN, para ligações descentralizadas de sensores e atuadores, com índice de proteção IP67, indicado para ambientes de operação hostil. Projetado para gerir diversos tipos de sensores e atuadores, utiliza a rede CAN para funções como leitura e escrita de dados, processamento de sinais e controlo de dispositivos. Neste projeto, o CR2512 é utilizado para ligar os sensores ao sistema e ao controlador da máquina, e para acionar o motor DC através de sinal PWM.

Pino	Função
1	Supply GND
23	Supply +DC
05	Supply +DC

**Tabela 12** – Ligações do CR2512 (Alimentação)

Pino	Função
50	CAN_L
51	CAN_H
15	CAN_GND
14	CAN_H
32	CAN_L
33	CAN_GND

**Tabela 13** – Ligações do CR2512 (Comunicação CAN)

**Interface CAN:** A interface CAN possui pinos para as linhas CAN\_H (alta) e CAN\_L (baixa), usados para comunicação CAN bus. A referência da interface CAN é feita através dos pinos CAN\_GND. Como se utilizam dois módulos CR2512, num deles foram usados os pinos 32 e 14 para a rede CANbus, e no outro, os pinos 50 e 51.

**Alimentação (Supply):** A alimentação principal do módulo é fornecida pelos pinos "Supply +DC", enquanto a massa é feita pelos pinos "Supply GND". Estes pinos garantem a alimentação necessária para o funcionamento de todos os canais e da interface CAN, que, neste caso, é de 24V.

<b>Canal</b>	<b>Pino de Entrada (IN)</b>	<b>Pino de Saída (OUT)</b>	<b>Pino de Saída (Corrente)</b>	<b>Pino de Saída (GND)</b>	<b>Pino de Alimentação (+VBB)</b>
<b>Canal 1</b>	47 (IN 1)	20 (OUT 1)	38 (Current 1)	02 (GND 1)	43 (+VBB1)
<b>Canal 2</b>	48 (IN 2)	21 (OUT 2)	39 (Current 2)	03 (GND 2)	44 (+VBB2)
<b>Canal 3</b>	49 (IN 3)	22 (OUT 3)	40 (Current 3)	04 (GND 3)	45 (+VBB3)
<b>Canal 4</b>	31 (IN 4)	42 (OUT 4)	41 (Current 4)	25 (GND 4)	46 (+VBB4)
<b>Canal 5</b>	29 (IN 5)	34 (OUT 5)	52 (Current 5)	16 (GND 5)	10 (+VBB5)
<b>Canal 6</b>	35 (IN 6)	35 (OUT 6)	53 (Current 6)	17 (GND 6)	29 (+VBB6)
<b>Canal 7</b>	36 (IN 7)	36 (OUT 7)	54 (Current 7)	18 (GND 7)	11 (+VBB7)
<b>Canal 8</b>	30 (IN 8)	30 (OUT 8)	55 (Current 8)	19 (GND 8)	08 (+VBB8)
<b>Canal 9</b>	06 (IN 9)	09 (OUT 9)	22 (Current 9)	04 (GND 9)	43 (+VBB9)
<b>Canal 10</b>	07 (IN 10)	28 (OUT 10)	40 (Current 10)	25 (GND 10)	44 (+VBB10)
<b>Canal 11</b>	24 (IN 11)	10 (OUT 11)	52 (Current 11)	16 (GND 11)	45 (+VBB11)
<b>Canal 12</b>	26 (IN 12)	29 (OUT 12)	53 (Current 12)	17 (GND 12)	46 (+VBB12)

**Tabela 14** - Ligações completas dos canais.

### Canais 1 a 12

Assim como os canais de 1 a 8, os canais de 9 a 12 seguem a mesma estrutura de ligação. Cada canal possui uma entrada (IN), uma saída (OUT), uma saída em corrente, a referência (GND) e um pino de alimentação específico (+VBB). O canal 1 foi ligado ao optoacoplador para redimensionar a amplitude do sinal PWM de 24V para 5V.

## **Mapa de ligações do módulo CR2512**

### **CR2512**

1. O pino 01 – Supply GND foi ligado ao GND da fonte de 24V.
2. O pino 23 – Supply +DC foi ligado aos 24V da fonte.
3. O pino 5 – Supply +DC foi ligado aos 24V da fonte.
4. O pino 14 – CAN\_H foi ligado ao CAN HIGH.
5. O pino 32 – CAN\_L foi ligado ao CAN\_LOW.
6. O pino 33 – CAN\_GND foi ligado ao GND da fonte de 24V.
7. O pino 02 foi ligado ao GND4 do driver MOSFET e ao GND do pino “-1” do optoacoplador.
8. O pino 20 foi ligado ao pino de entrada “+1” do optoacoplador.
9. O pino 43 do canal 9 (CH9) binário foi ligado ao pino 1, fio castanho, do sensor indutivo.
10. O pino 06 do canal 9 (CH9) binário foi ligado ao pino 4, fio azul, do sensor indutivo.
11. O pino 47 do canal 9 (CH9) binário foi ligado ao pino 3, fio preto, do sensor indutivo.
12. O pino 44 do canal 10 (CH10) binário foi ligado ao pino 1, fio castanho, do sensor indutivo.
13. O pino 07 do canal 10 (CH10) binário foi ligado ao pino 4, fio azul, do sensor indutivo.
14. O pino 48 do canal 10 (CH10) binário foi ligado ao pino 3, fio preto, do sensor indutivo.
15. O pino 49 do canal 11 (CH11) binário foi ligado ao pino 2, fio branco, do sensor indutivo.

### **CR2512\_1**

1. O pino 01 – Supply GND foi ligado ao GND da fonte de 24V.
2. O pino 23 – Supply +DC foi ligado aos 24V da fonte.
3. O pino 5 – Supply +DC foi ligado aos 24V da fonte.
4. O pino 14 – CAN\_H foi ligado ao CAN HIGH.
5. O pino 32 – CAN\_L foi ligado ao CAN LOW.
6. O pino 33 – CAN\_GND foi ligado ao GND da fonte de 24V.
7. O pino 43 do canal 9 (CH9) binário foi ligado ao fio vermelho do sensor de velocidade.
8. O pino 47 do canal 9 (CH9) binário foi ligado ao fio verde (OUT) do sensor de velocidade.

9. O pino 48 do canal 10 (CH10) binário foi ligado ao fio branco (OUT) do sensor de velocidade
10. O pino 06 do canal 9 (CH9) binário foi ligado ao fio preto (GND) do sensor de velocidade.

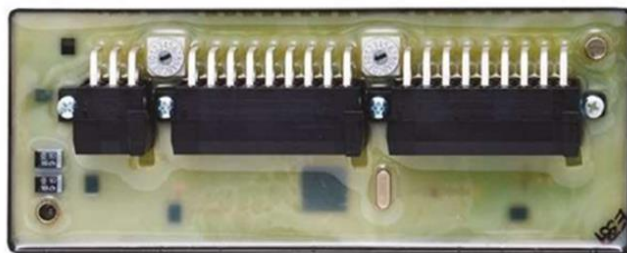
### **Cabo de ligação com conector AMP EC2086**



**Figura 25** – Cabo de ligação aos módulos de E/S, estanque.[29]

O conector AMP EC2086, ilustrado na Figura 24, é usado para ligar e organizar cabos de forma segura e estanque, adequado para ambientes de máquinas móveis e industriais. No projeto, foi utilizado com o módulo CR2512 para interligar componentes eletrônicos e sensores ao controlador.

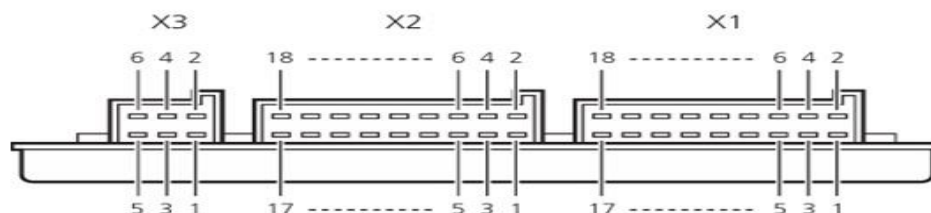
### ***Módulo CR2014***



**Figura 26** – Módulo de E/S remotas para cabine – IFM CR2014.[27]

O módulo CR2014 é um módulo de entradas e saídas digitais e analógicas, específico para ser usado no interior de cabines de máquinas móveis e com capacidade de transmissão de dados sob o protocolo CAN Bus. Neste projeto, foi utilizado para a leitura dos botões do joystick HJ60, do scroll e dos eixos X e Y do joystick HJ60. A seguir, são apresentadas as ligações do módulo CR2014.

## Ligações Elétricas



**Figura 27** - Ligação elétrica do CR2014.

As tabelas a seguir fornecem informações sobre os plugs (conectores) X1, X2 e X3. As tabelas estão separadas para cada um dos plugs do módulo CR2014 com as respectivas explicações:

<b>Pin</b>	<b>Plug X1</b>
<b>1</b>	Canal 1 Binário IN 1
<b>2</b>	+UB
<b>3</b>	Canal 2 Binário IN 2
<b>4</b>	+UB
<b>5</b>	Canal 3 Binário IN 3
<b>6</b>	+UB
<b>7</b>	Canal 4 Binário IN 4
<b>8</b>	+UB
<b>9</b>	+UB
<b>10</b>	+UB
<b>11</b>	Canal 5 Analógico IN 5
<b>12</b>	Canal 6 Analógico IN 6
<b>13</b>	GND
<b>14</b>	GND

<b>15</b>	Canal 7 Binário IN 7 / Binário OUT 7
<b>16</b>	GND
<b>17</b>	Canal 8 Binário IN 8 / Binário OUT 8 PWM 8
<b>18</b>	GND

Tabela 15 - Ligações no conector X1.

<b>Pin</b>	<b>Plug X2</b>
<b>1</b>	Canal 9 Bin IN 9
<b>2</b>	+UB
<b>3</b>	Canal 10 Bin IN 10
<b>4</b>	+UB
<b>5</b>	Canal 11 Bin IN 11
<b>6</b>	+UB
<b>7</b>	Canal 12 Bin IN 12
<b>8</b>	+UB
<b>9</b>	+UB
<b>10</b>	+UB
<b>11</b>	Canal 13 Ana IN 13
<b>12</b>	Canal 14 Ana IN 14
<b>13</b>	GND
<b>14</b>	GND
<b>15</b>	Canal 15 Bin IN 15 / Bin OUT 15
<b>16</b>	GND
<b>17</b>	Canal 16 Bin IN 16 / Bin OUT 16 PWM 16
<b>18</b>	GND

Tabela 16 - Ligações no conector X2

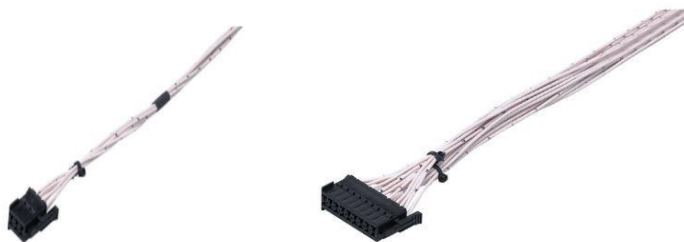
**Explicação:**

- **Plug X1** concentra-se principalmente nas entradas digitais (Bin IN), saídas digitais (Bin OUT) e entradas analógicas (Ana IN) para os canais de 1 a 8. "+UB" representa a tensão de alimentação, e "GND" é a massa.
- **Plug X2** oferece funcionalidades adicionais para entradas e saídas digitais, além de entradas analógicas, expandindo os canais disponíveis de 9 a 16. Este plug continua a fornecer tensão de alimentação (+UB) e massa (GND).

Pin	Plug X3
1	+UB
2	GND
3	+UB
4	GND
5	CAN_L
6	CAN_H

**Tabela 17** - Ligações no conector X3**Explicação:**

- **Plug X3** é utilizado principalmente para a alimentação geral do módulo e para a comunicação via CAN bus.
  - CAN\_L e CAN\_H são as linhas de comunicação do CAN bus, essenciais para a comunicação com outros módulos do sistema.

**Cabos de ligação com conector AMP EC1520 e EC1533****Figura 28** – Cabos de ligação ao módulo IFM CR2014.[30][31]

Os cabos de ligação com conectores AMP EC1520 e EC1533 são usados para interligar módulos eletrônicos com múltiplas entradas e saídas, garantindo ligações seguras e organizadas. O cabo EC2015 é ligado ao Plug X3, e o EC1533 é ligado aos Plugs X1 e X2.

### Módulo IFM CR1202



Figura 29 – Controlador com HMI integrada – IFM CR1202.[26]

O módulo CR1202 é um controlador de máquinas móveis com HMI integrada, que possui a dupla funcionalidade de controlar todo o sistema e apresentar ao operador todas as ações parametrizadas. Possui um display gráfico programável, interfaces de comunicação diversas e um processador ARM Dual Core de 64 bits, que opera a 800 MHz com uma GPU integrada. Este módulo é utilizado principalmente em aplicações industriais devido à sua robustez e capacidade de operar em ambientes severos. Neste projeto, funciona como controlador e interface HMI, fornecendo informações em tempo real dos dispositivos associados.

Pino	Função	Descrição
82	Não utilizado	
83	IFM0-	Ligação para unidade externa
84	CAN0_L	Linha CAN0 Low
85	CAN1_L	Linha CAN1 Low
86	CAN2_L	Linha CAN2 Low

<b>87</b>	CAN3_L	Linha CAN3 Low
<b>88</b>	Não utilizado	
<b>89</b>	SERVICE1	Entrada de Serviço 1
<b>90</b>	Não utilizado	
<b>91</b>	IFM0+	Ligação para unidade externa
<b>92</b>	CAN0_H	Linha CAN0 High
<b>93</b>	CAN1_H	Linha CAN1 High
<b>94</b>	CAN2_H	Linha CAN2 High
<b>95</b>	CAN3_H	Linha CAN3 High
<b>96</b>	Não utilizado	
<b>97</b>	SERVICE0	Entrada de Serviço 0
<b>98</b>	VIDEO0_GND	GND de vídeo 0
<b>99</b>	VIDEO1_GND	GND de vídeo 1
<b>100</b>	Não utilizado	
<b>101</b>	Não utilizado	
<b>102</b>	Não utilizado	
<b>103</b>	Não utilizado	
<b>104</b>	AUDIO_OUT_R-	Saída de áudio direita

<b>105</b>	AUDIO_OUT_L-	Saída de áudio esquerda -
<b>106</b>	VIDEO0_SIGNAL	Sinal de vídeo 0
<b>107</b>	VIDEO1_SIGNAL	Sinal de vídeo 1
<b>108</b>	Não utilizado	
<b>109</b>	Não utilizado	
<b>110</b>	Não utilizado	
<b>111</b>	Não utilizado	
<b>112</b>	AUDIO_OUT_R+	Saída de áudio direita +
<b>113</b>	AUDIO_OUT_L+	Saída de áudio esquerda +
<b>114</b>	VBB0	Alimentação 0
<b>115</b>	IN0	Entrada digital 0
<b>116</b>	IN1	Entrada digital 1
<b>117</b>	OUT0	Saída digital 0
<b>118</b>	OUT1	Saída digital 1
<b>119</b>	VBB15	Alimentação 15
<b>120</b>	GND	GND

121	VBB30	Alimentação 30
-----	-------	-------------------

Tabela 18 - Ligações do conector AMO de 40 pinos do módulo CR1202

**Explicação:**

- **Pinos 83 e 91 (IFM0- e IFM0+):** Utilizados para ligações com uma unidade externa, provavelmente uma interface de comunicação ou módulo de expansão, permitindo a integração de funcionalidades adicionais ao CR1202.
- **Pinos 84, 85, 86 e 87 (CAN0\_L, CAN1\_L, CAN2\_L, CAN3\_L):** Estas são linhas de comunicação CAN (Controller Area Network) de sinal de nível baixo (Low). O barramento CAN é amplamente usado em automação industrial e em automóveis para comunicação rápida e fiável entre diferentes dispositivos.
- **Pinos 92, 93, 94 e 95 (CAN0\_H, CAN1\_H, CAN2\_H, CAN3\_H):** Correspondem às linhas de comunicação CAN de sinal de nível alto (High), que, juntamente com as linhas Low, formam os barramentos CAN para diferentes redes. Cada par (CANx\_L e CANx\_H) deve ser ligado a dispositivos que suportem comunicação CAN.
- **Pinos 97 e 89 (SERVICE0 e SERVICE1):** São entradas de serviço, usadas para tarefas de manutenção ou configuração do módulo.
- **Pinos 98 e 99 (VIDEO0\_GND e VIDEO1\_GND):** São ligações de massa para entradas de vídeo, essenciais para garantir a integridade do sinal e reduzir ruídos.
- **Pinos 106 e 107 (VIDEO0\_SIGNAL e VIDEO1\_SIGNAL):** Utilizados para os sinais de vídeo analógico, recebendo sinais de câmaras compatíveis com os padrões PAL e NTSC.
- **Pinos 104, 105, 112 e 113 (AUDIO\_OUT\_R-, AUDIO\_OUT\_L-, AUDIO\_OUT\_R+, AUDIO\_OUT\_L+):** São pinos para saídas de áudio estéreo, tanto para os canais esquerdo como para os canais direito, podendo ser ligados a alto-falantes ou outros dispositivos de áudio.
- **Pinos 115 e 116 (IN0 e IN1):** São entradas digitais que podem ser usadas para receber sinais de sensores ou outros dispositivos de entrada digital.
- **Pinos 117 e 118 (OUT0 e OUT1):** São saídas digitais, utilizadas para controlar dispositivos como relés, LEDs ou outros atuadores.
- **Pinos 114, 119 e 121 (VBB0, VBB15 e VBB30):** São pinos de alimentação que fornecem diferentes níveis de tensão ao módulo ou aos dispositivos ligados a ele.
- **Pino 120 (GND):** Este é o pino de massa principal do conector, essencial para estabilizar a tensão e evitar ruídos e interferências.

Pino	Função	Descrição
1	TxD+	Transmissão de dados +
2	RxD+	Receção de dados +
3	TxD-	Transmissão de dados -
4	RxD-	Receção de dados -
5	GND	GND

Tabela 19 – Ligações do conector M12 de 5 pinos do CR1202

#### Explicação:

- **Pinos 1 e 3 (TxD+ e TxD-):** Estes pinos são para a transmissão de dados, sendo positivos e negativos respetivamente, usados para enviar dados do módulo para outro dispositivo.
- **Pinos 2 e 4 (RxD+ e RxD-):** Estes são os pinos de receção de dados, positivos e negativos, que recebem dados de outro dispositivo para o módulo.
- **Pino 5 (GND):** Este pino de massa é utilizado para estabilizar as tensões de transmissão e receção, minimizando ruídos e garantindo a integridade da comunicação.

#### *Cabos de conexão com conector AMP EC0705*



Figura 30 – Conector e cablagem da ligação de 40 pinos do CR1202. [32]

Os cabos de ligação com conector AMP EC0705, ilustrado na Figura 29, são projetados para interligar sistemas de forma robusta e organizada, principalmente em aplicações automóveis ou industriais. Quando usados com o módulo CR1202, o conector AMP facilita a transmissão segura de sinais entre diferentes componentes.

## Toggle switch



**Figura 31** - Toggle switch.

A chave do toggle switch de alavanca funciona com um terminal central comum (COM) que se conecta alternadamente a um dos dois terminais laterais, dependendo da posição da alavanca. Quando a alavanca está em uma posição, o terminal central faz contato com um dos terminais laterais, permitindo a passagem de corrente para um circuito. Ao mudar a posição da alavanca, o contato muda para o outro terminal, redirecionando a corrente para outro circuito. Esse funcionamento permite ligar/desligar ou alternar entre dois circuitos.

## *Disjuntor NFN210*



**Figura 32** - Disjuntor. [38]

O dispositivo na imagem é um disjuntor NFN210 da marca Hager, que é um disjuntor bipolar. Ele é utilizado para proteção de circuitos elétricos contra sobrecargas e curto-circuitos, garantindo a segurança dos sistemas elétricos em instalações residenciais e industriais. Neste projeto, o disjuntor é utilizado na proteção do quadro elétrico do demonstrador.

<b>Pinos/Terminais</b>	<b>Função</b>	<b>Descrição</b>
<b>1</b>	Entrada Fase 1	Ligação de entrada da fase para o circuito protegido
<b>2</b>	Entrada Fase 2	Ligação de entrada do Neutro
<b>3</b>	Saída Fase 1	Ligação de saída da fase após a proteção do disjuntor
<b>4</b>	Saída Fase 2	Ligação de saída do Neutro protegido

**Tabela 20** - Ligações do disjuntor.

O NFN210 é um disjuntor bipolar de 10A (corrente nominal), normalmente utilizado em sistemas monofásicos. Ele serve para proteger tanto a fase quanto o neutro num circuito, e pode atuar automaticamente desligando a alimentação em caso de sobrecarga ou curto-circuito.

### Bornes



**Figura 33** - Bornes de ligação

Os bornes ou blocos de terminais são usados principalmente em quadros elétricos para a ligação de cablagem de maneira segura e organizada. Estes blocos de terminais são frequentemente utilizados para ligar diferentes circuitos numa instalação elétrica, permitindo que fios sejam ligados de forma simples, sem a necessidade de soldadura.

## PWM

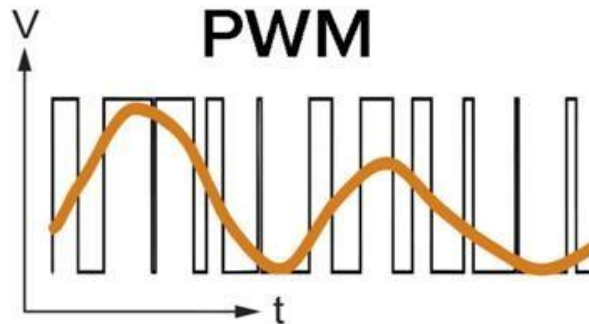


Figura 34 - Sinal PWM. [34]

PWM, ou Modulação por Largura de Pulso, consiste em alternar rapidamente um sinal entre ligado (HIGH) e desligado (LOW) para controlar a quantidade de energia fornecida a um dispositivo. A "largura" do pulso, ou seja, o tempo em que o sinal permanece no estado HIGH durante cada ciclo, determina a quantidade média de potência entregue ao dispositivo.

### Duty Cycle (Ciclo de Trabalho)

O principal parâmetro do PWM é o duty cycle (ciclo de trabalho), que é a proporção do tempo em que o sinal está em nível HIGH durante um ciclo completo de onda.

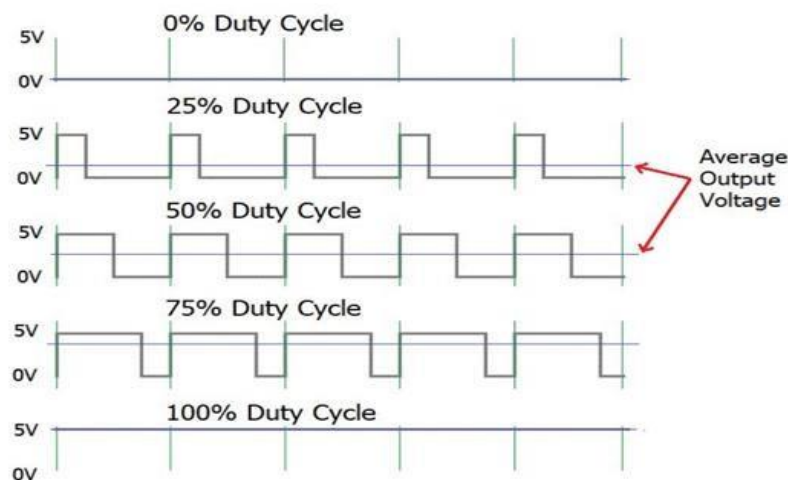


Figura 35 - Duty cycle (ciclo de trabalho)

- **0% Duty Cycle:** O sinal está sempre em LOW, ou seja, não há energia entregue.
- **25% Duty Cycle:** O sinal está 75% em LOW e 25% HIGH.
- **50% Duty Cycle:** O sinal está em HIGH metade do tempo e em LOW na outra metade, resultando em metade da potência total sendo entregue.
- **75% Duty Cycle:** O sinal está 25% em LOW e 75% HIGH.
- **100% Duty Cycle:** O sinal está sempre em HIGH, entregando potência máxima ao dispositivo.

## Frequência do PWM

A frequência do PWM é a taxa na qual o ciclo se repete, medida em Hertz (Hz). Uma frequência alta significa que os pulsos são mais curtos e repetem mais rapidamente. A escolha da frequência depende da aplicação:

- **Baixa Frequência:** Mais comum no controle de potência de motores, onde mudanças mais lentas na potência são aceitáveis.
- **Alta Frequência:** Usada em aplicações como iluminação LED e áudio, onde é importante evitar efeitos perceptíveis, como flicker ou ruído.

## Aplicações do PWM no projeto

O PWM foi usado para controlar o motor DC, permitindo variar a velocidade dos motores DC ao regular a potência fornecida.

## 3.2 Desenvolvimento do Software de Controle

### 3.2.1 Ambiente de Desenvolvimento Codesys

#### Software - Codesys V3.5 SP16 Patch 5



**Figura 36** – Software de programação para controladores industriais.

O desenvolvimento do software do projeto foi realizado no ambiente **Codesys**, uma plataforma amplamente reconhecida na área de automação industrial. Esta ferramenta oferece suporte integral para a implementação de redes CANopen, facilitando a integração de dispositivos e garantindo uma comunicação eficaz entre os módulos de hardware utilizados no projeto.

O Codesys é um ambiente baseado na norma **IEC 61131-3**, que padroniza as linguagens de programação utilizadas na automação industrial, como Ladder Diagram

(LD), Structured Text (ST), Function Block Diagram (FBD), Sequential Function Chart (SFC) e Instruction List (IL). Esta norma permite que os engenheiros desenvolvam aplicações de controlo de maneira estruturada e modular, facilitando a manutenção, a expansão e a reutilização de código. As principais linguagens de programação usadas no projeto foram Ladder Diagram (LD), Structured Text (ST) e Function Block Diagram (FBD).

### Integração dos módulos CR2512 E CR2014 da Rede na rede CANopen dentro do codesys

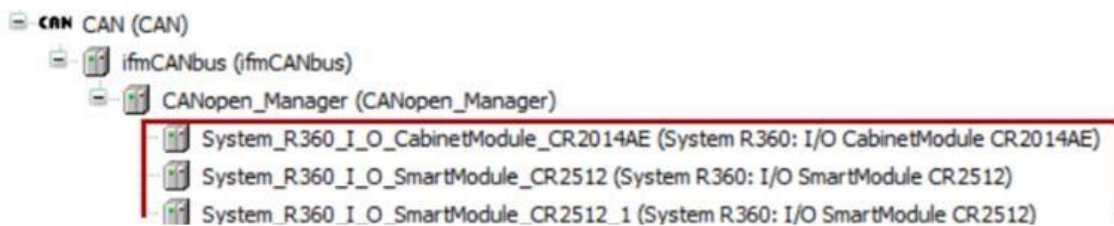


Figura 37 - Ilustração da integração dos módulos CR2512 e do CR2014 na rede

A integração foi feita utilizando arquivos **EDS (Electronic Data Sheet)** dos módulos CR2512 e CR2014AE. A imagem ilustra a estrutura da rede configurada, onde cada dispositivo foi endereçado corretamente e integrado via CANopen.

#### Gestão da Rede com CANopen:

No Codesys, o CANopen atua como o nó mestre da rede, controlando a comunicação entre todos os dispositivos ligados. A estrutura de rede foi configurada para garantir uma troca de dados eficiente e segura.

#### Uso de SDOs, PDOs e CANopen I/O mapping:



Figura 38 - Ilustração das configurações feitas nos módulos CR2512 e do CR2014.

## Uso de SDOs, PDOs e CANopen I/O Mapping:

As configurações dos **SDOs (Service Data Objects)** e **PDOs (Process Data Objects)** foram mantidas conforme especificado nos arquivos EDS dos módulos. Não houve necessidade de ajustes adicionais, pois as configurações padrão eram adequadas para o projeto. O único trabalho realizado foi o mapeamento das variáveis de entrada e saída do programa, utilizando o CANopen I/O Mapping para garantir a correta associação dos sinais de controlo e feedback.

## Desenvolvimento de Blocos de Função no Codesys

No ambiente Codesys, foram desenvolvidos blocos de função específicos para a geração de PWM e controlo do motor, onde a variável de saída do bloco de função é mapeada no canal 1 do módulo CR2512 ligado via CANopen, além de um bloco de função para receber os sinais e valores do joystick HJ60.

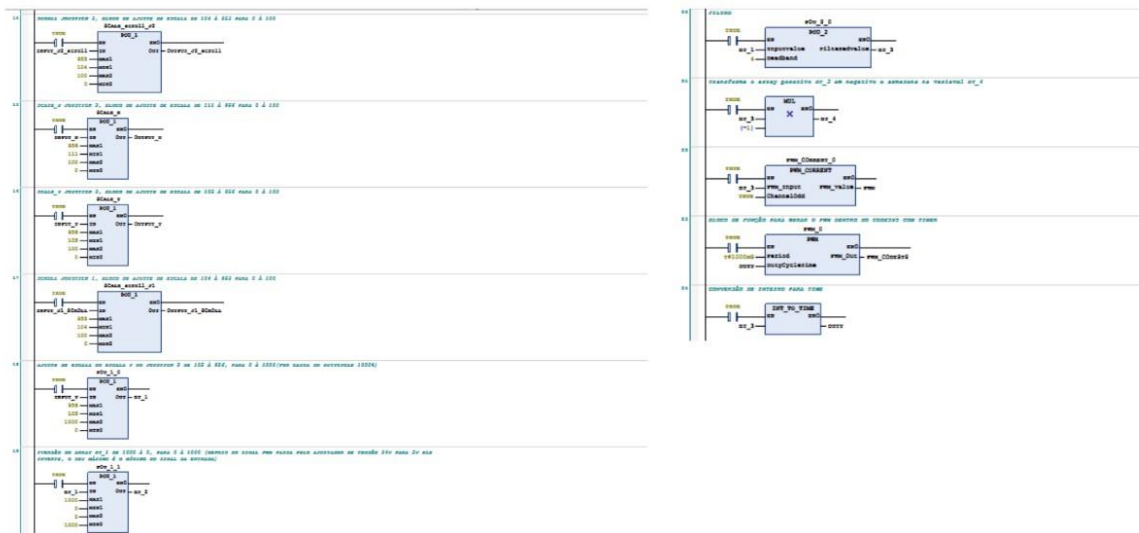


Figura 39 - Blocos de função desenvolvidos no codesys.

**Bloco de Função PWMGenerator:** Este bloco foi programado para gerar sinais de PWM com características configuráveis, como frequência e ciclo de trabalho, que são fundamentais para o controlo da velocidade do motor ZGA42RH(A6249). A parametrização foi realizada para que o bloco de função pudesse ser ajustado dinamicamente, conforme as exigências operacionais do sistema. Posteriormente, foi substituído pelo bloco de função PWM-CORRENT, que envia o valor de duty cycle ao módulo CR2512, o qual por sua vez produz o sinal PWM.

**Bloco de Função POU\_1:** Foi projetado para receber o valor proveniente do joystick HJ60 (dos botões, scroll e dos eixos Y e X) para exibir ao utilizador por meio da HMI. O módulo utilizado para mapear esses valores foi o CR2014.

**Bloco de Função Mul:** Já existente no Codesys, foi utilizado para inverter o sinal dos valores do joystick, tornando-os negativos, de forma a serem aplicados no canal 2 do módulo CR2512 para o PWM.

O Codesys fornece ferramentas de simulação e debug que são essenciais para garantir que as aplicações funcionem conforme o esperado antes de serem implementadas no hardware real. No processo de desenvolvimento, essas ferramentas permitiram a simulação completa da lógica de controlo e da comunicação em rede, identificando possíveis falhas e permitindo a correção proativa de erros.

### **Redes CANbus e CANONPEN e o Modelo OSI CAN bus**

Controller Area Network (CAN), desenvolvido pela empresa alemã Bosch na década de 1980, foi projetado para aplicação na indústria automóvel, proporcionando uma rede de comunicação mais simples e eficiente.

Existem vários protocolos CAN. O CAN Flexible Data-rate (CAN FD) é uma extensão da camada de ligação de dados clássica. Este protocolo oferece benefícios como aumento da carga útil de 8 para 64 bytes, uma taxa de dados mais elevada (até 5 Mbit/s) e maior fiabilidade devido à utilização de uma verificação de redundância cíclica (CRC) aprimorada, que reduz o risco de erros não detetados. Além disso, os controladores CAN FD e CAN clássico podem ser integrados no mesmo sistema em certas condições, o que reduz custos e complexidade, tornando o protocolo aplicável em usos mais exigentes, como em veículos elétricos.

O protocolo SAE-J1939, norma padrão para veículos pesados, permite uma taxa de transmissão de 250 Kbit/s. Este protocolo usa um identificador alargado de 29 bits, com a maioria das mensagens transmitidas no barramento, embora alguns dados apenas sejam disponibilizados após solicitação. As mensagens são identificadas por números de grupo de parâmetros (PGN) de 18 bits e designadas por números de parâmetros suspeitos (SPN). Além disso, contém variáveis multibyte, sendo enviadas pelo byte menos significativo primeiro.

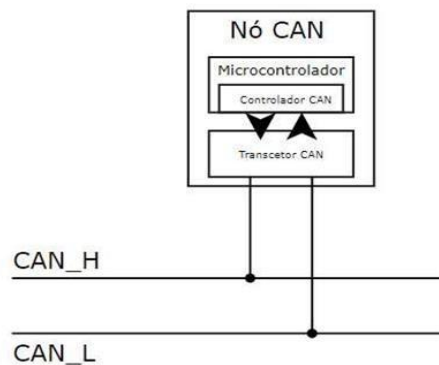
O protocolo On-Board Diagnostic (OBD2) é utilizado, como o nome indica, no diagnóstico on-board. É uma capacidade de autodiagnóstico e comunicação essencial para empresas de manutenção automóvel, pela sua praticidade em detetar problemas em tempo real através dos códigos gerados.

Neste projeto, é utilizado o protocolo CAN, que é uma rede de comunicação em série baseada numa topologia de barramento, comumente utilizada em aplicações de automação industrial. É composta por vários nós ou dispositivos ligados a um único barramento de comunicação, onde cada nó é identificado por um ID (identificador) único, permitindo a comunicação em tempo real.

Existem dois tipos de nós:

- **Nós mestre**, responsáveis por iniciar a comunicação no barramento;
- **Nós escravos**, que respondem às solicitações do nó mestre ou iniciam a comunicação de forma independente, permitindo uma comunicação bidirecional entre os dispositivos ligados.

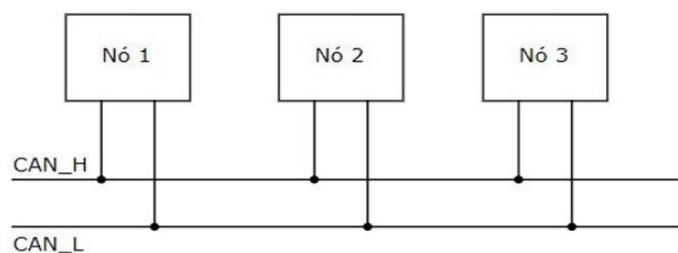
Cada nó consiste numa Unidade Central de Processamento (CPU), controlador CAN e transmissor-recetor. Todos os nós podem enviar e receber dados, mas não em simultâneo. Estes podem enviar dados para a rede onde foram endereçados, mas não para outra rede.



**Figura 40** - Exemplificação de barramento CAN.

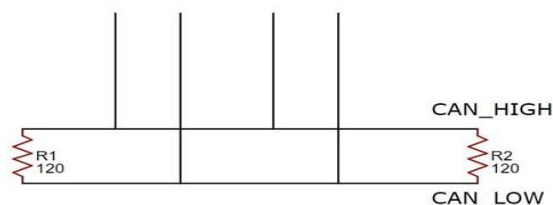
## Endereçamento

Relativamente ao endereçamento dos nós ou dispositivos, o CAN utiliza identificadores únicos (IDs). Cada mensagem transmitida no barramento possui um ID que indica a prioridade e a natureza da mensagem. O ID é composto por bits que determinam a prioridade, com mensagens de ID menor a terem prioridade sobre as de ID maior. Este aspeto é essencial, por exemplo, em sistemas automóveis, onde o sistema CAN interliga todas as unidades de controlo de um veículo que necessitam de comunicar em tempo real, como o motor, sensores, sistemas antiderrapantes, travagem, luzes e vidros elétricos, reduzindo significativamente a quantidade de cablagem necessária.



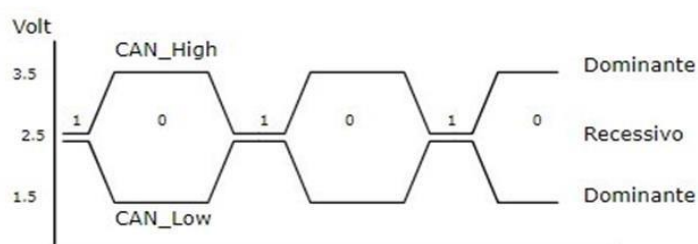
**Figura 41** - Barramento CAN com diferentes nós.

A colocação de duas resistências de  $120\ \Omega$  (ohms) nas extremidades do barramento é necessária para evitar ou eliminar reflexões de sinal, assegurando a integridade das comunicações.



**Figura 42** - Acoplamento do barramento com resistências.

O protocolo define uma forma de transmissão baseada na diferença de tensão entre dois fios, CAN\_High e CAN\_Low. Por convenção, o bit recessivo corresponde ao nível lógico 1, enquanto o bit dominante é o nível lógico 0. Para que o barramento opere, é necessário que a tensão do fio dominante CAN\_High seja de 3,5 V e do CAN\_Low de 1,5 V.



**Figura 43** - Nível de barramento CAN.

### Taxas de Transmissão e Comprimentos de Cabo na Rede CAN Bus

A rede suporta vários ritmos de transmissão, configuráveis até 1 Mbps. Dependendo das necessidades do sistema, é possível ajustar a taxa de bits para otimizar a velocidade de transmissão de dados.

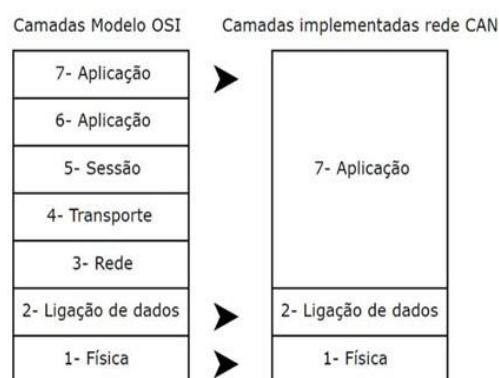
A Tabela 17 mostra as taxas de bits recomendadas e o correspondente comprimento máximo do cabo.

<b>BIT RATE</b>	<b>COMPRIMENTO DO CABO</b>
<b>10 Kbits/s</b>	6,7 km
<b>20 Kbits/s</b>	3,3 km
<b>50 Kbits/s</b>	1,0 km
<b>125 Kbits/s</b>	500 m
<b>250 Kbits/s</b>	250 m
<b>500 Kbits/s</b>	125 m
<b>1 Mbits/s</b>	25 m

**Tabela 21** - Relação entre ritmo de transmissão e distâncias de comunicação

### **CANbus e CANopen no Modelo OSI**

A CANbus opera principalmente nas camadas 1 (Física) e 2 (Ligação) do modelo OSI, enquanto a CANopen opera nas camadas superiores do modelo OSI, especificamente nas camadas 6 (Apresentação) e 7 (Aplicação):



**Figura 44** - Modelo OSI e a implementação da rede CANopen e CANbus.

## **Camada 1: Física**

A camada física define os aspetos elétricos da comunicação. No CANbus, inclui cabos, conectores e características do sinal elétrico, como tensão e tempos de subida/descida dos pulsos. A especificação CAN determina como os bits são transmitidos no barramento, partilhado por todos os dispositivos (nós) na rede. As taxas de transmissão podem variar, sendo comuns valores entre 125 kbps e 1 Mbps.

## **Camada 2: Ligação**

A camada de ligação assegura que os dados transmitidos entre dispositivos na rede CAN estejam livres de erros e que a comunicação seja ordenada. O CANbus utiliza um protocolo de acesso ao meio (CSMA/CR – Carrier Sense Multiple Access with Collision Resolution), que permite a partilha do barramento sem colisões de dados. A prioridade das mensagens é determinada pelos seus identificadores, garantindo que as mensagens mais críticas sejam transmitidas em primeiro lugar.

## **Camada 6: Apresentação<sup>4</sup>**

A camada de apresentação em CANopen é responsável pela interpretação e conversão dos dados em formatos compreensíveis para a aplicação. Em termos práticos, isto pode envolver a definição de tipos de dados, como inteiros, números flutuantes ou sequências de caracteres, usados nas mensagens CANopen. Embora o CANopen opere de maneira abstrata nas camadas superiores do modelo OSI, ele traduz os comandos e dados específicos da aplicação em formatos que podem ser transmitidos pela CANbus.

## **Camada 7: Aplicação**

A camada de aplicação é onde o CANopen define o comportamento da rede e as funcionalidades específicas que cada dispositivo deve desempenhar. Inclui a definição de objetos de dados, como SDOs (Service Data Objects) e PDOs (Process Data Objects), usados para transmitir dados entre dispositivos. O CANopen especifica o mapeamento de dados, a configuração de nós, o gerenciamento de rede e outros serviços necessários para que dispositivos heterogéneos possam interagir eficazmente.

### **Trama de Dados do Protocolo CAN**

O protocolo CAN utiliza dois formatos de trama de dados: o formato padrão e o formato alargado. A principal diferença entre eles está na quantidade de bits no campo identificador, que é responsável pela arbitragem e priorização da transmissão. O formato padrão possui 11 bits no campo identificador, permitindo até 2048 mensagens únicas, enquanto o formato alargado utiliza 18 bits adicionais, totalizando 29 bits, permitindo até 536 milhões de mensagens.

No processo de arbitragem de uma trama de dados no protocolo CAN, diferentes nós competem pela prioridade de transmissão. Cada nó emite um bit dominante (0) para ganhar a arbitragem e continuar a transmissão, enquanto os outros interrompem as suas transmissões ao perderem a arbitragem devido ao bit dominante emitido por outro nó.

	Bits inicial	ID Bits					
		10	8	7	6 5 4 3	2 1 0	
Nó 1	0	0	0	0	0 0 1 0	0 0 1	
Nó 2	0	0	0	0	1 Transmissão interrompida por prioridade		
Nó 3	0	0	0	1	Transmissão interrompida por prioridade		
Dados CAN	0	0	0	0	0 0 1 0	0 0 1	

**Tabela 22** - Arbitragem da Trama de Dados

### Caracterização do Pacote de Bits

As figuras abaixo ilustram a constituição de uma trama padrão e uma trama alargada, destacando os diferentes campos e os seus respetivos tamanhos.

Start	Identifier	SRR	IDE	Identifier	RTR	r1	r0	DLC	DATA	CRC	ACK	EOF+IFS
1 Bit	11 Bit	1 Bit	1 Bit	18 Bit	1 Bit	1 Bit	1 Bit	4 Bit	0...8 Byte	16 Bit	2 Bit	10 Bit

Start	Identifier	RTR	IDE	r0	DLC	DATA	CRC	ACK	EOF+IFS
1 Bit	11 Bit	1 Bit	1 Bit	1 Bit	4 Bit	0...8 Byte	16 Bit	2 Bit	10 Bit

**Figura 45** - Trama padrão e uma trama alargada.

## Detalhes dos Campos das Tramas CAN:

### 1. Start (1 Bit):

- **Função:** Marca o início da trama CAN. Este bit é sempre dominante (0) e é usado para sincronizar todos os nós da rede, indicando que uma nova mensagem está a começar.

### 2. Identifier (11 Bits):

- **Função:** Este campo é o identificador da mensagem, com 11 bits no formato padrão. Define a prioridade da mensagem na rede, onde valores menores indicam maior prioridade.

### 3. SRR (Substitute Remote Request, 1 Bit):

- **Função:** Usado apenas em tramas alargadas, este bit é recessivo (1) e substitui o bit RTR quando o identificador alargado está a ser usado. Serve para diferenciar mensagens que contêm dados reais de mensagens de pedidos de dados (remotos).

### 4. IDE (Identifier Extension, 1 Bit):

- **Função:** Indica se o identificador da trama é do tipo padrão (11 bits) ou estendido (29 bits). Quando IDE é recessivo (1), a trama é alargada e utiliza o campo de identificador adicional.

### 5. Identifier (18 Bits):

- **Função:** Campo adicional de identificador utilizado em tramas alargadas. Juntamente com os 11 bits iniciais, forma um identificador total de 29 bits, permitindo um maior número de mensagens únicas na rede.

### 6. RTR (Remote Transmission Request, 1 Bit):

- **Função:** Este bit distingue entre uma trama que contém dados e uma trama que solicita dados. Se for dominante (0), a trama contém dados; se for recessivo (1), a trama é uma solicitação de dados de outro nó.

### 7. r1 (1 Bit):

- **Função:** Bit reservado, sempre dominante (0). Garante compatibilidade e robustez do protocolo.

#### 8. **r0 (1 Bit):**

- **Função:** Outro bit reservado, também dominante (0), com função semelhante ao r1 para assegurar que o formato da trama seja consistente e compatível com as especificações do protocolo CAN.

#### 9. **DLC (Data Length Code, 4 Bits):**

- **Função:** Especifica o número de bytes de dados na mensagem. Pode variar de 0 a 8 bytes, e o valor deste campo define o tamanho do campo de dados na trama.

#### 10. **DATA (0 a 8 Bytes):**

- **Função:** Campo onde são transmitidos os dados efetivos. O tamanho deste campo é determinado pelo DLC e pode conter de 0 a 8 bytes de dados.

#### 11. **CRC (Cyclic Redundancy Check, 16 Bits):**

- **Função:** Este campo contém um valor calculado a partir dos bits da trama para verificação de integridade. O recetor recalcula o CRC e compara com o valor recebido para identificar possíveis erros na transmissão.

#### 12. **ACK (Acknowledgment, 2 Bits):**

- **Função:** Campo utilizado para confirmar que a mensagem foi recebida corretamente. Um bit dominante (0) neste campo indica que pelo menos um nó na rede recebeu a mensagem sem erros.

#### 13. **EOF + IFS (End of Frame + Interframe Space, 10 Bits):**

- **Função:** Composto por 7 bits de EOF (End of Frame), que marcam o fim da transmissão da trama, e 3 bits de IFS (Interframe Space), um espaço entre tramas que permite libertar o barramento antes de iniciar uma nova trama. Estes bits asseguram a deteção correta do final da trama e evitam colisões entre tramas consecutivas.

### 3.2.2 Integração dos Módulos na Rede CAN Bus

A rede CAN Bus foi selecionada pela sua robustez, confiabilidade e capacidade de operar em ambientes com elevado nível de interferência eletromagnética, comuns em máquinas móveis florestais.

#### 3.2.2.1 Protocolo CAN Bus e CANopen

O protocolo CAN Bus foi implementado para ligar todos os módulos eletrónicos do sistema, enquanto o protocolo CANopen foi usado para facilitar a troca de dados e o controlo dos dispositivos. Cada dispositivo foi configurado com um ID único na rede, garantindo comunicação eficiente entre os módulos e evitando conflitos de endereçamento.

Para assegurar o correto funcionamento da rede CAN Bus:

- **Configuração de IDs:** Cada módulo foi programado com um ID específico para comunicação, de acordo com o protocolo CANopen.
- **Monitorização de dados:** O módulo CR1202, atuando como núcleo da rede, geriu a troca de mensagens e apresentou dados operacionais em tempo real ao operador.

#### 3.2.2.2 Implementação do Controlo da Lâmina de Corte via Joystick HJ60

O controlo do motor da lâmina de corte foi um componente crítico do sistema, sendo o joystick HJ60 integrado para garantir um acionamento preciso e dinâmico.

Passos da implementação:

- **Integração do Joystick HJ60:** O joystick foi ligado ao módulo CR2014, que traduz os movimentos em comandos eletrónicos para ajustar a posição e velocidade da lâmina de corte.
- **Geração de sinais PWM:** O CR2014 gera sinais PWM (Modulação por Largura de Pulso), enviados ao CR2512, que regula a potência fornecida ao motor, controlando velocidade e força de corte.
- **Testes de precisão:** Foram realizados testes para avaliar o controlo da lâmina em várias condições operacionais, como a resistência variável da madeira.

## 3.3 Desenvolvimento do Simulador Físico e Gémeo Digital

### 3.3.1 Demonstrador Físico

Foi desenvolvido um demonstrador físico para simular as condições operacionais do harvester em ambiente controlado. Incluiu:

- **Estrutura de teste:** Atuadores, sensores e réplicas de componentes principais, como a lâmina de corte e motor.
- **Joystick e HMI reais:** Utilização do joystick HJ60 e do módulo CR1202, como no sistema real, garantindo que os testes fossem semelhantes à operação real.

Objetivos principais do demonstrador físico:

- **Validação de hardware e software:** Testar a comunicação entre módulos e a rede CAN, bem como a resposta ao controlo do joystick.

## 3.4 Testes e Validação do Sistema

### 3.4.1 Testes de Comunicação

Embora ferramentas de diagnóstico CAN específicas não tenham sido usadas, o sistema foi extensivamente testado, observando-se o comportamento dos dispositivos e monitorizando dados operacionais na HMI (CR1202). A comunicação entre módulos foi validada através de:

- **Observação direta:** A HMI permitiu ao operador visualizar dados em tempo real e confirmar a execução correta dos comandos.
- **Análise de tempo de resposta:** O sistema foi testado para assegurar comunicação rápida e precisa, evitando atrasos significativos que comprometessem a operação do harvester.

## 3.5 Processo de Validação

Durante os testes, observou-se variações significativas nos valores enviados pelo joystick HJ60 ao programa Codesys, afetando a precisão do controlo da lâmina de corte. As flutuações no sinal do joystick podiam gerar respostas imprecisas ou indesejadas no movimento do motor da lâmina, impactando a operação da máquina.

### 3.5.1 Implementação de um Filtro para Estabilização dos Sinais

Para estabilizar os sinais de PWM no bloco de função PWM\_CORRENT, foi implementado um filtro simples para suavizar variações rápidas e garantir uma resposta mais estável. O filtro média móvel foi a técnica escolhida, suavizando oscilações ao calcular a média de leituras PWM consecutivas. Este filtro ajudou a eliminar ruídos e proporcionou um controlo mais preciso dos atuadores.

### 3.5.2 Resultados da Validação com o Filtro

A validação revelou que o uso do filtro reduziu as flutuações indesejadas no sinal de PWM, proporcionando uma resposta mais estável e confiável. A lógica de desativação

para valores entre -50 e +50 foi eficazmente mantida, assegurando o funcionamento dos canais dentro dos parâmetros definidos sem comprometer o desempenho.

### 3.6 Fluxograma da Metodologia

O fluxograma da metodologia representa o fluxo de implementação e testes do sistema de monitorização e controlo do harvester:

1. Definição da arquitetura do sistema.
2. Configuração da rede CAN Bus e dos módulos CR2014, CR2512 e CR1202.
3. Implementação do controlo da lâmina de corte via joystick HJ60.
4. Desenvolvimento do simulador físico e do gémeo digital.
5. Realização de testes de comunicação e controlo da lâmina.
6. Validação do simulador e do gémeo digital.

## 4. Desenvolvimento da Plataforma de Controlo e Monitorização

Este capítulo descreve o processo de desenvolvimento do sistema de monitorização e controlo, centrado na criação de um protótipo com funcionalidades equivalentes às de um harvester. O protótipo foi projetado para simular as operações principais de um harvester, como o controlo do motor e a monitorização em tempo real, sendo composto por módulos interligados via rede CAN Bus. O objetivo desta fase foi validar o funcionamento do sistema de um harvester moderno.

### 4.1 Desenho do Sistema

O protótipo desenvolvido representa uma versão simplificada de um harvester, integrando os módulos principais através de uma rede CAN Bus. A comunicação entre os módulos e a interface homem-máquina (HMI) permitiu testar as funcionalidades essenciais de controlo e monitorização em tempo real, simulando o ambiente de trabalho de um harvester.

Os componentes principais do sistema incluem:

- **CR1202 (ecomatDisplay):** Interface HMI que exhibe as informações em tempo real para o operador.
- **CR2014:** Módulo que lê os sinais do joystick HJ60, utilizado para enviar comandos de controlo ao motor que simula o motor da lâmina de corte.
- **CR2512:** Dois módulos com funções distintas: um responsável pela leitura de sinais de sensores e pela geração de sinais PWM para o controlo do motor, e outro para leitura de rotação e velocidade simulada.

### 4.1.1 Diagrama de Blocos Funcional

O diagrama de blocos funcional do sistema mostra como os módulos estão conectados via CAN Bus, demonstrando o fluxo de dados entre eles.

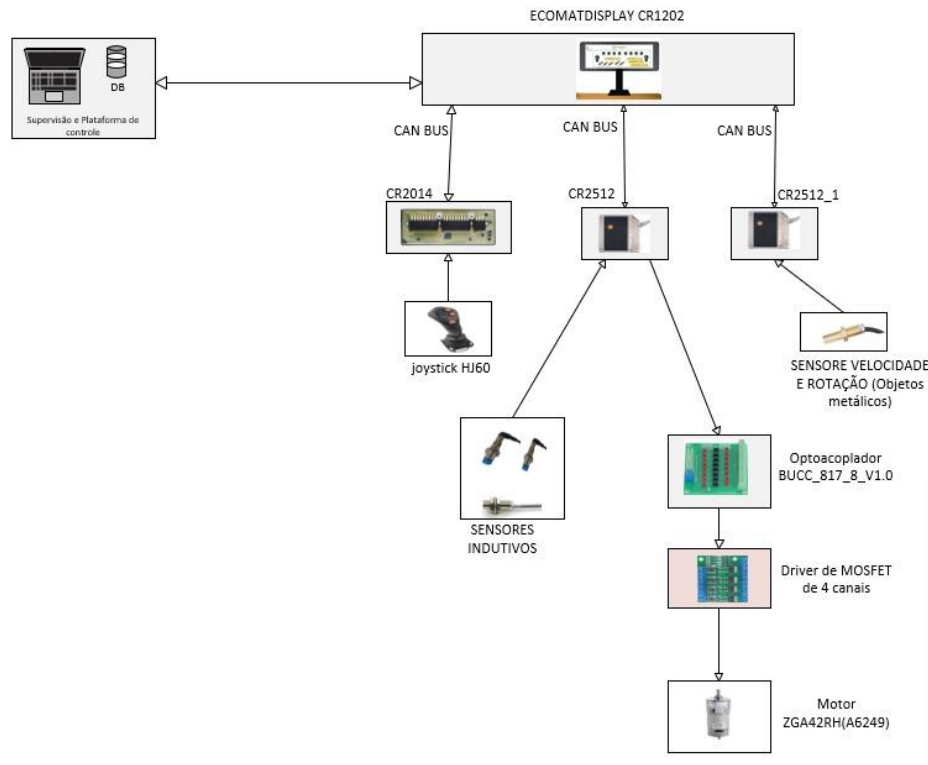


Figura 46 – Arquitetura do sistema.

## 4.2 Arquitetura do Sistema de Monitorização e Controlo de Máquinas Móveis

O desenvolvimento do sistema foi dividido em duas fases principais: a programação no ambiente Codesys, realizada num PC, e a implementação no protótipo físico, que replicou algumas das funcionalidades principais de um harvester. O objetivo destas fases foi validar o comportamento dos módulos e o controlo do motor.

### 4.2.1 Ambiente de Simulação, Plataforma de Monitorização e Controlo

A primeira fase do desenvolvimento foi realizada no ambiente de simulação do Codesys, onde os módulos principais do sistema (CR2014, CR2512 e CR1202) foram configurados e integrados para trabalhar em conjunto via o protocolo CAN Bus. A seguir, são descritos os principais passos da configuração.

## Configuração dos Módulos IFM

Este segmento detalha a configuração dos módulos IFM utilizados no projeto, especificamente os módulos CR2512, CR2014 e CR1202, essenciais para o controlo e monitorização da máquina móvel.

### Configurações Iniciais dos Módulos

Após as ligações físicas dos elementos numa rede CAN Bus, a configuração de comunicação entre módulos foi realizada através do software Codesys, como mencionado anteriormente.

O processo de configuração iniciou-se com a importação dos ficheiros **Electronic Data Sheets (EDS)** de cada módulo, adicionando-os ao software. Como os módulos utilizados no projeto são da marca IFM, os ficheiros EDS podem ser obtidos no site oficial, pesquisando pelo nome do módulo em <https://www.ifm.com/pt/pt>.



Figura 47 - Ficheiro EDS.

Após o download dos ficheiros EDS necessários, procedeu-se à adição destes ao Codesys. Para isso, na aba "Tools", selecionou-se "Device Repository" e depois "Install".

A adição do ficheiro EDS permite integrar os módulos no ecrã principal, na aba "Devices".

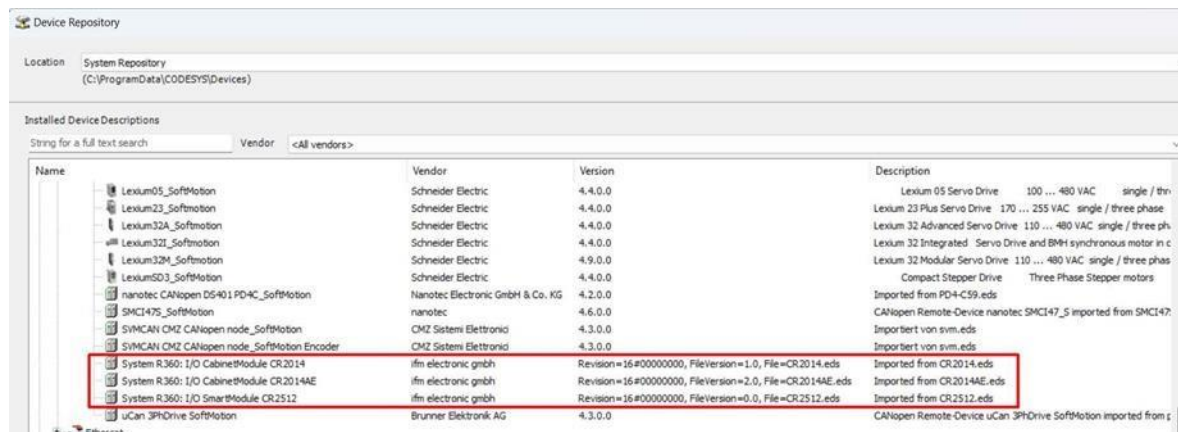


Figura 48 - Livraria de dispositivos.

O módulo CR2512 foi configurado como o controlador principal. Iniciou-se o projeto com "New Project", selecionando a pasta "ecomatDisplay" e escolhendo a template "ifm ecomatDisplay\_64bit, 12 polegada 3:1 Wide".

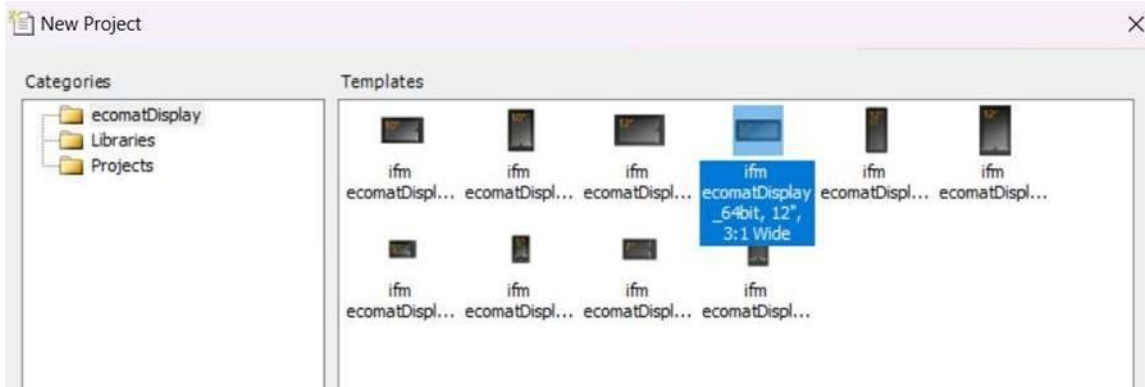


Figura 49 - Template ecomatDisplay.

Após iniciar o projeto, foram adicionados os dispositivos CAN, navegando até ao dispositivo "CAN", "Add Device", "Fieldbuses", "CANbus" e "ifmCANbus".

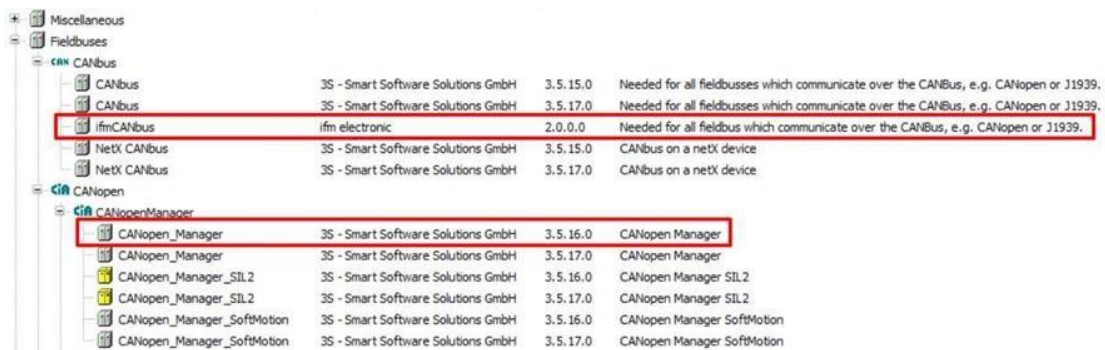


Figura 50 - Dispositivos para ligação CAN.

Posteriormente, adicionou-se o dispositivo "ifmCANbus" e "CANopen\_Manager", que permite realizar o mapeamento das entradas e saídas, configurações e sincronização. A partir do "CANopen\_Manager", foram adicionados os módulos CR2014 e os CR2512.



Figura 51 - Módulos CAN.

## 4.2.1 Configuração da Rede CAN e Detalhes das Configurações dos Módulos IFM

**Configuração da Rede para os Módulos CR2512, CR2014 e CR1202**  
Inicialmente, a ligação entre o HMI e o computador foi estabelecida através de Ethernet. Para isso, foi atribuído manualmente o endereço IP indicado pelo fabricante nas definições de rede do computador.

Atribuição de IP:	Manual
Endereço IPv4:	192.168.82.246
Máscara de IPv4:	255.255.255.0

Figura 52 – Endereçamento IP.

```
Ethernet adapter Ethernet 2:  
  
Connection-specific DNS Suffix . . . :  
Link-local IPv6 Address . . . . . : fe80::fd8e:35d1:5510:14c2%15  
IPv4 Address. . . . . : 192.168.82.246  
Subnet Mask . . . . . : 255.255.255.0  
Default Gateway . . . . . :
```

Figura 53 - Verificação da ligação IP e máscara.

Após configurar o IP, a ligação entre o HMI e o computador ficou estabelecida, permitindo programar, modificar a interface e ajustar as configurações dos módulos.

## 4.2.2 CANopen Manager

No dispositivo CANopen Manager, o **NodeID** foi atribuído com o valor 127, correspondendo ao último ID admissível, já que a configuração varia de 1 a 127. A opção “heartbeat producing” foi ativada, configurando-a para 200 milissegundos. O protocolo **Heartbeat** é utilizado como um comando de confirmação enviado pelo dispositivo mestre, sendo também utilizado para verificar a disponibilidade dos dispositivos na rede CAN.

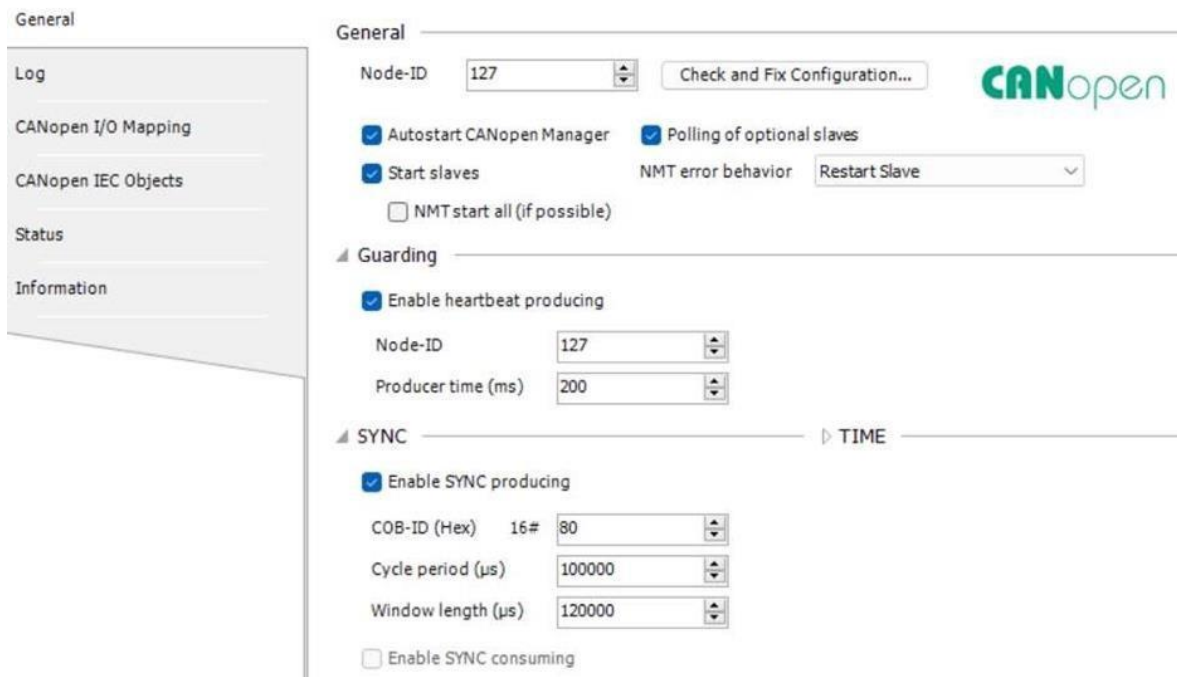


Figura 54 - Configurações da CAN Manager.

A sincronização no CANopen Manager é necessária para que o “Slave CANopen” saiba quando deve enviar o **PDO (Process Data Objects)**, que contém os valores analógicos.

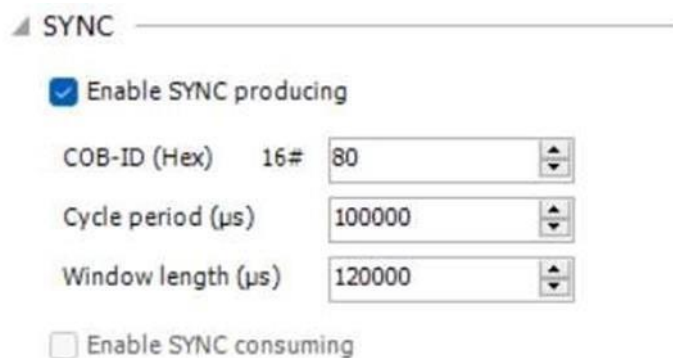


Figura 55 - Configurações de sincronismo CAN Manager.

Os **SDOs (Service Data Objects)** são comunicações mestre/escravo usadas principalmente para a configuração do conversor, enquanto os **PDos** são utilizados para controlo em tempo real e transferência de dados. Um PDO pode ser enviado ponto a ponto ou transmitido para todos os dispositivos.

## Configuração do Módulo CR2014

A configuração do módulo CR2014 iniciou-se definindo fisicamente o baud rate (S1) para 125 Kbits/s e o **Node-ID (S2)**.

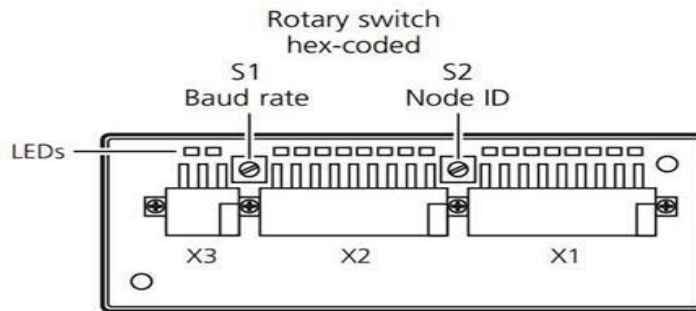


Figura 56 - Esquema do módulo CR2014

O módulo permite a alteração do Baud rate e *NodeID* conforme a aplicação.

Switch	Position	Description
S1 Baud rate	0	1000 Kbits/s
	1	800 kBits/s
	2	500 Kbit/s
	3	250 Kbit/s
	4	125 Kbit/s
	5	100 Kbit/s
	6	50 Kbit/s
	7	20 Kbit/s
	8...E	not defined
	F	adjustment via object directory (default)
S2 Node-ID	0	not defined
	1...E	1...14
	F	adjustment via object directory (default)

Figura 57- Opções do Baud rate e ID do módulo CR2014.

O Node ID deve ser único em cada módulo da rede CAN Bus. Por padrão, o ID dos módulos é 125kbit/s, mas o CR2014 usa 100Kbit/s. Para alterar, basta seguir os passos indicados:

- 1- Na rede CANopen dentro do codesys aceda ao módulo

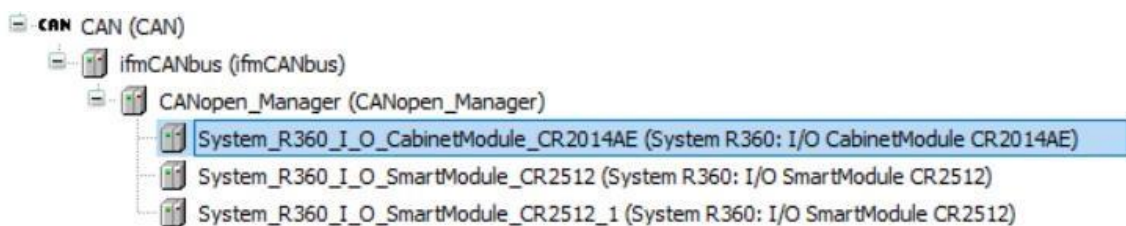


Figura 58 - Integração dos módulos na rede can dentro do codesys.

## 2- - Navega até ao SDOs

Line	Index:Subindex	Name	Value	Bit Length	Abort on Error	Jump to lir
26	16#1A00:16#01	Set Mapping	16#61000110	32	<input type="checkbox"/>	[
27	16#1A00:16#00	Set number of mappings	16#01	8	<input type="checkbox"/>	[
28	16#1800:16#01	Set and enable COB-ID	16#00000181	32	<input type="checkbox"/>	[
29	16#1801:16#01	Disable PDO	16#80000281	32	<input type="checkbox"/>	[
30	16#1801:16#02	Set transmission type	16#01	8	<input type="checkbox"/>	[
31	16#1801:16#03	Set inhibit time	16#0000	16	<input type="checkbox"/>	[
32	16#1801:16#05	Set event time	16#0000	16	<input type="checkbox"/>	[
33	16#1A01:16#00	Clear pdo mapping	16#0	8	<input type="checkbox"/>	[
34	16#1A01:16#01	Set Mapping	16#64010210	32	<input type="checkbox"/>	[
35	16#1A01:16#02	Set Mapping	16#64010210	32	<input type="checkbox"/>	[
36	16#1A01:16#03	Set Mapping	16#64010310	32	<input type="checkbox"/>	[
37	16#1A01:16#04	Set Mapping	16#64010410	32	<input type="checkbox"/>	[
38	16#1A01:16#00	Set number of mappings	16#04	8	<input type="checkbox"/>	[
39	16#1801:16#01	Set and enable COB-ID	16#00000281	32	<input type="checkbox"/>	[

Figura 59 - SDOs do módulo CR2014.

## 3- Clicar em Add SDO e ir até “Bit rate switch a” adicionar e “Bit rate switch b”

The screenshot shows the 'Add SDO' dialog box with the 'Select Item from Object Directory' window open. The 'bit rate switch a' and 'bit rate switch b' items are highlighted in the object directory table.

Index:Subindex	Name	AccessType	Type	Default
* 16#1601	2. receive PDO mapping			
* 16#1800	1. transmit PDO parameter			
* 16#1801	2. transmit PDO parameter			
* 16#1A00	1. transmit PDO mapping			
* 16#1A01	2. transmit PDO mapping			
* 16#2000	io configuration			
16#2001:16#00	PWM frequency	RW	USINT	100
16#20F0:16#00	node id switch a	RW	USINT	16#20
16#20F1:16#00	node id switch b	RW	USINT	16#20
16#20F2:16#00	bit rate switch a	RW	USINT	16#03
16#20F3:16#00	bit rate switch b	RW	USINT	16#03
16#20F4:16#00	Auto Start	RW	UINT	16#00
* 16#2500	8bit user variables			
* 16#2510	16bit user variables			
* 16#2520	32bit user variables			
* 16#6200	binary outputs			
* 16#6414	Analog outputs (ifm specific)			

Below the table, the 'Name' field is set to 'Unknown Object', 'Index' is '16#0', 'Subindex' is '16#0', and 'Bit length' is '8'. The 'Value' field is '0'. The 'OK' button is highlighted.

Figura 60 - Selecionar itens para o objeto do diretório do módulo CR2014.

**Nota:** A adição é um de cada vez, e como orientado no datasheet na página 6 (baixar o datasheet no site oficial ifm) o valor do 20F02e do 20F3 devem sempre ser iguais. E esse 16#4 é posto no value.

#### 4- Trocar para os a posição “4”

Line	Index/Subindex	Name	Value	Bit Length	Abort on Error	Jump to line if error	Next line	Comment
24	16#1800:16#05	Set event time	16#0000	16	<input type="checkbox"/>	<input type="checkbox"/>	0	
25	16#1A00:16#00	Clear pdo mapping	16#0	8	<input type="checkbox"/>	<input type="checkbox"/>	0	
26	16#1A00:16#01	Set Mapping	16#61000110	32	<input type="checkbox"/>	<input type="checkbox"/>	0	
27	16#1A00:16#00	Set number of mappings	16#01	8	<input type="checkbox"/>	<input type="checkbox"/>	0	
28	16#1800:16#01	Set and enable COB-ID	16#00000181	32	<input type="checkbox"/>	<input type="checkbox"/>	0	
29	16#1801:16#01	Disable PDO	16#80000281	32	<input type="checkbox"/>	<input type="checkbox"/>	0	
30	16#1801:16#02	Set transmission type	16#01	8	<input type="checkbox"/>	<input type="checkbox"/>	0	
31	16#1801:16#03	Set inhibit time	16#0000	16	<input type="checkbox"/>	<input type="checkbox"/>	0	
32	16#1801:16#05	Set event time	16#0000	16	<input type="checkbox"/>	<input type="checkbox"/>	0	
33	16#1A01:16#00	Clear pdo mapping	16#0	8	<input type="checkbox"/>	<input type="checkbox"/>	0	
34	16#1A01:16#01	Set Mapping	16#64010110	32	<input type="checkbox"/>	<input type="checkbox"/>	0	
35	16#1A01:16#02	Set Mapping	16#64010210	32	<input type="checkbox"/>	<input type="checkbox"/>	0	
36	16#1A01:16#03	Set Mapping	16#64010310	32	<input type="checkbox"/>	<input type="checkbox"/>	0	
37	16#1A01:16#04	Set Mapping	16#64010410	32	<input type="checkbox"/>	<input type="checkbox"/>	0	
38	16#1A01:16#00	Set number of mappings	16#04	8	<input type="checkbox"/>	<input type="checkbox"/>	0	
39	16#1801:16#01	Set and enable COB-ID	16#00000281	32	<input type="checkbox"/>	<input type="checkbox"/>	0	
40	16#20F2:16#00	bit rate switch a	16#4	8	<input type="checkbox"/>	<input type="checkbox"/>	0	
41	16#20F3:16#00	bit rate switch b	16#4	8	<input type="checkbox"/>	<input type="checkbox"/>	0	

Figura 61 - Configuração do bit rate switch a e b do módulo CR2014.

O valor hexadecimal "16#4" é utilizado para configurar parâmetros específicos, como mostrado na Figura 61.

Ativou-se também, nas definições avançadas, a opção “Enable expert settings”, bem como “Enable heartbeat producing”, configurada para 200 milissegundos.

Figura 62 - Configuração dos parâmetros gerais do módulo CR2014.

No **I/O Mapping**, que é um mapa de entradas e saídas do módulo, alocaram-se os pinos a serem lidos. Estes valores são então transmitidos para a aplicação **POU**, onde se encontra a programação em **LD (Ladder Diagram)**. Na Figura 62, vemos três variáveis inteiras analógicas alocadas aos canais 5, 6 e 13, responsáveis por ler os valores transmitidos pelos roletes e eixos X e Y dos joysticks conforme o utilizador os opera.

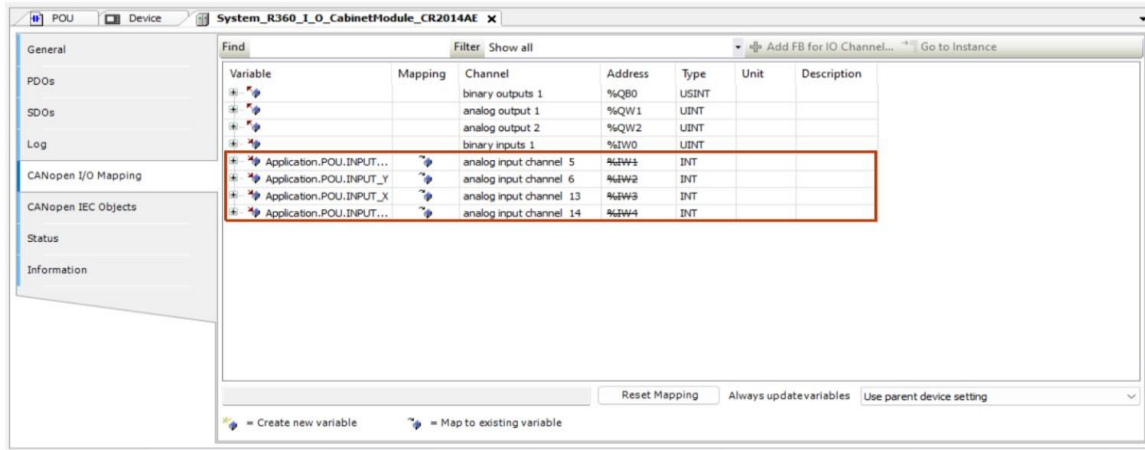


Figura 63 - CR2014, Mapa de entradas se sinal no codesys.

### Configuração do Módulo CR2512\_1

Por padrão, os módulos CR2512 vêm com o **NodeID 32**, e ao conectar mais de um módulo com o mesmo ID na rede CAN Bus, cria-se um conflito. Para resolver este problema, é necessário desligar fisicamente todos os módulos da rede CAN Bus, exceto o que se pretende reconfigurar. No Codesys, reconfigura-se o módulo, atribuindo um **Node ID** diferente, como o 33, conforme demonstrado a seguir:

- 1- A penas o módulo a se reconfigurar pode estar na rede no codesys como na rede CANbus física

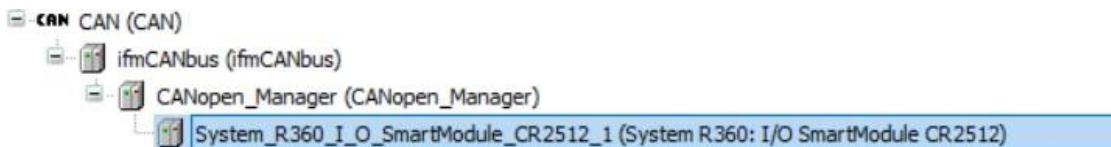


Figura 64 - Módulo CR2512 na rede CAN no codesys.

2- Acede o módulo em questão e navegue até o general e ative o *Enable settings*

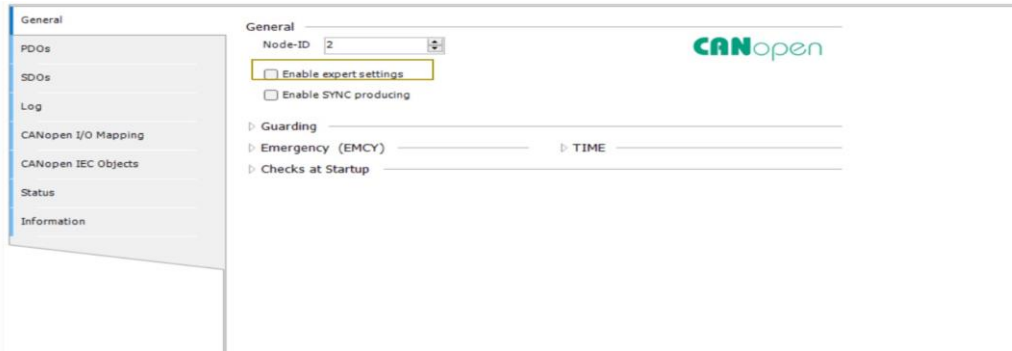


Figura 65- Habilitação do módulo CR2512 no codesys.

3- Troca o Node-ID para 33

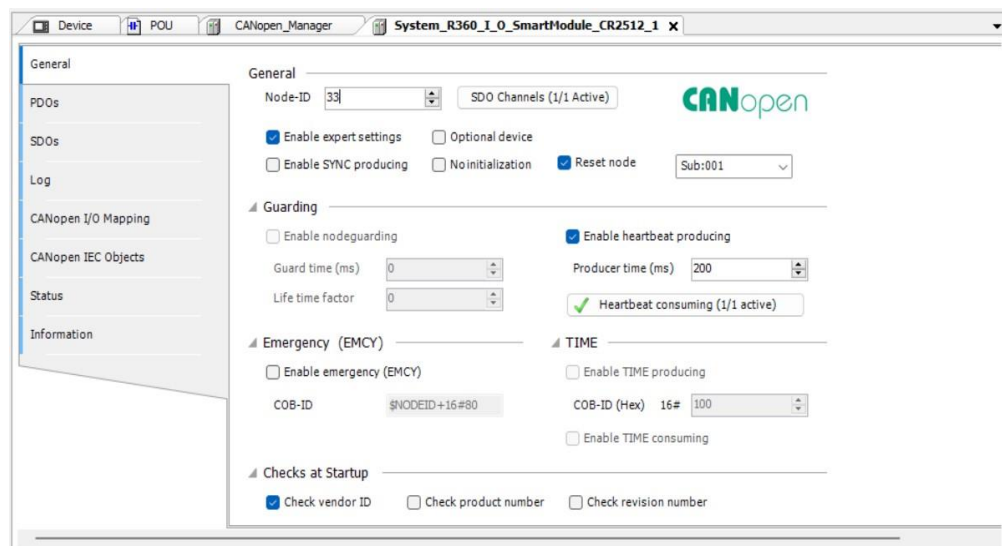


Figura 66 - Definir o Node\_ID módulo CR2512\_1.

4- Acede o módulo em questão e navegue até o SDOs

Line	Index:Subindex	Name	Value	Bit Length	Abort on Error	Jump to line
26	16#1A00:16#01	Set Mapping	16#61000110	32	<input type="checkbox"/>	[
27	16#1A00:16#00	Set number of mappings	16#01	8	<input type="checkbox"/>	[
28	16#1800:16#01	Set and enable COB-ID	16#00000181	32	<input type="checkbox"/>	[
29	16#1801:16#01	Disable PDO	16#80000281	32	<input type="checkbox"/>	[
30	16#1801:16#02	Set transmission type	16#01	8	<input type="checkbox"/>	[
31	16#1801:16#03	Set inhibit time	16#0000	16	<input type="checkbox"/>	[
32	16#1801:16#05	Set event time	16#0000	16	<input type="checkbox"/>	[
33	16#1A01:16#00	Clear pdo mapping	16#0	8	<input type="checkbox"/>	[
34	16#1A01:16#01	Set Mapping	16#64010110	32	<input type="checkbox"/>	[
35	16#1A01:16#02	Set Mapping	16#64010210	32	<input type="checkbox"/>	[
36	16#1A01:16#03	Set Mapping	16#64010310	32	<input type="checkbox"/>	[
37	16#1A01:16#04	Set Mapping	16#64010410	32	<input type="checkbox"/>	[
38	16#1A01:16#00	Set number of mappings	16#04	8	<input type="checkbox"/>	[
39	16#1801:16#01	Set and enable COB-ID	16#00000281	32	<input type="checkbox"/>	[

Figura 67- SDOs do módulo CR2512.

5- Clicar em “Add SDO” ir até “Node ID rate switch a” adicionar e “Node ID rate switch b”

The 'Select Item from Object Directory' dialog displays the following table:

Index:Subindex	Name	AccessType	Type	Default
* 16#1601	2. receive PDO mapping			
* 16#1800	1. transmit PDO parameter			
* 16#1801	2. transmit PDO parameter			
* 16#1A00	1. transmit PDO mapping			
* 16#1A01	2. transmit PDO mapping			
* 16#2000	io configuration			
16#2001:16#00	PWM frequency	RW	USINT	100
16#20F0:16#00	node id switch a	RW	USINT	16#20
16#20F1:16#00	node id switch b	RW	USINT	16#20
16#20F2:16#00	bit rate switch a	RW	USINT	16#03
16#20F3:16#00	bit rate switch b	RW	USINT	16#03
16#20F4:16#00	Auto Start	RW	UINT	16#00
* 16#2500	8bit user variables			
* 16#2510	16bit user variables			
* 16#2520	32bit user variables			
* 16#6200	binary outputs			
* 16#6414	Analog outputs (ifm specific)			

Below the dialog, the configuration fields are:

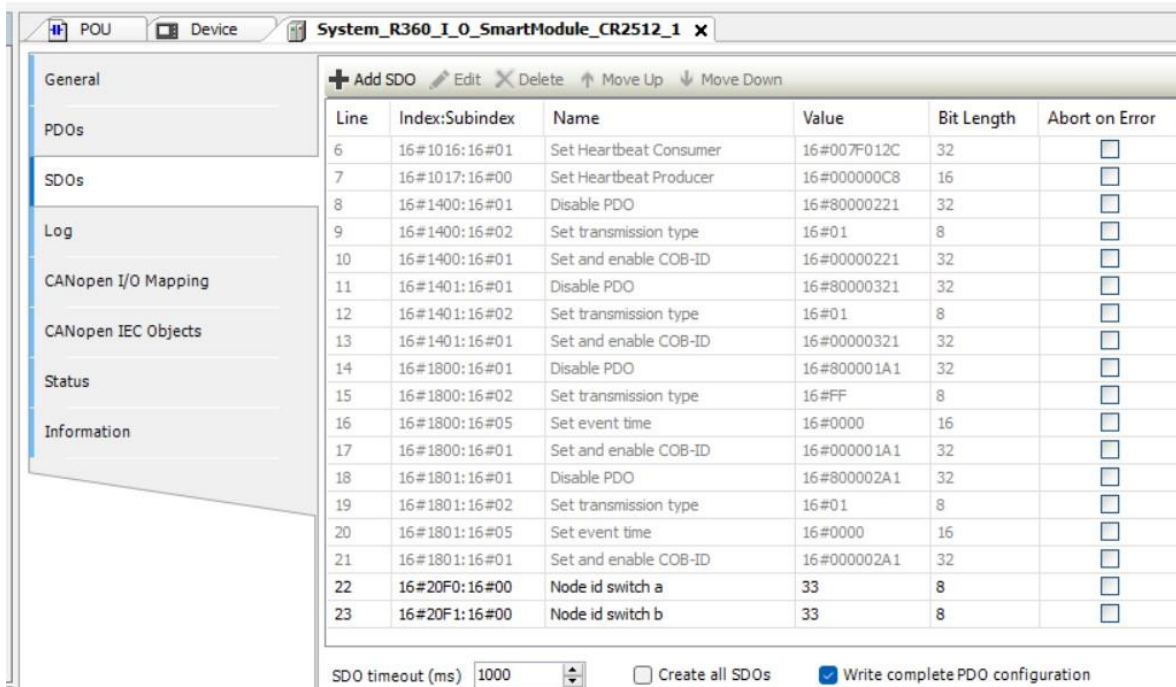
- Name: Unknown Object
- Index: 16#0
- Subindex: 16#0
- Bit length: 8
- Value: 0

Figura 68 - Selecionar o objeto 20F0 e 20F1 ao módulo CR2512\_1.

**Nota:** A adição dos módulos é feita um de cada vez. Conforme orientado no datasheet (página 18), os valores dos registos 20F0 e 20F1 devem ser iguais. Para o

**NodeID 33**, deve-se introduzir o valor hexadecimal no campo “Value”, aceitando-se números hexadecimais ou decimais.

#### 6- Trocar para os o valor 33



**Figura 69** - Alteração da configuração padrão node\_ID 32 para Node\_ID 33.

Assim, depois de desligar e voltar a ligar, a configuração estará alterada. Desta forma, podes reconectar o módulo ao CAN bus e retornar ao programa em CODESYS, com todos os módulos implementados. No projeto, ele recebe dados do sensor de velocidade; a seguir, mostrarei o mapeamento da variável que recebe o sinal do sensor.:

**Nota:** É apenas um sensor que retorna dois valores, velocidade e rotação (Um encoder)

### Configuração do Módulo CR2512

A Configuração do outro módulo CR2512 é definir o NodeID para um 32 que já vem por padrão que é o que vem definido no equipamento desde fábrica, ativar as definições avançadas “*Enable expert settings*” e modificar o “*hearthbeat producing*” para 200 milisegundos.

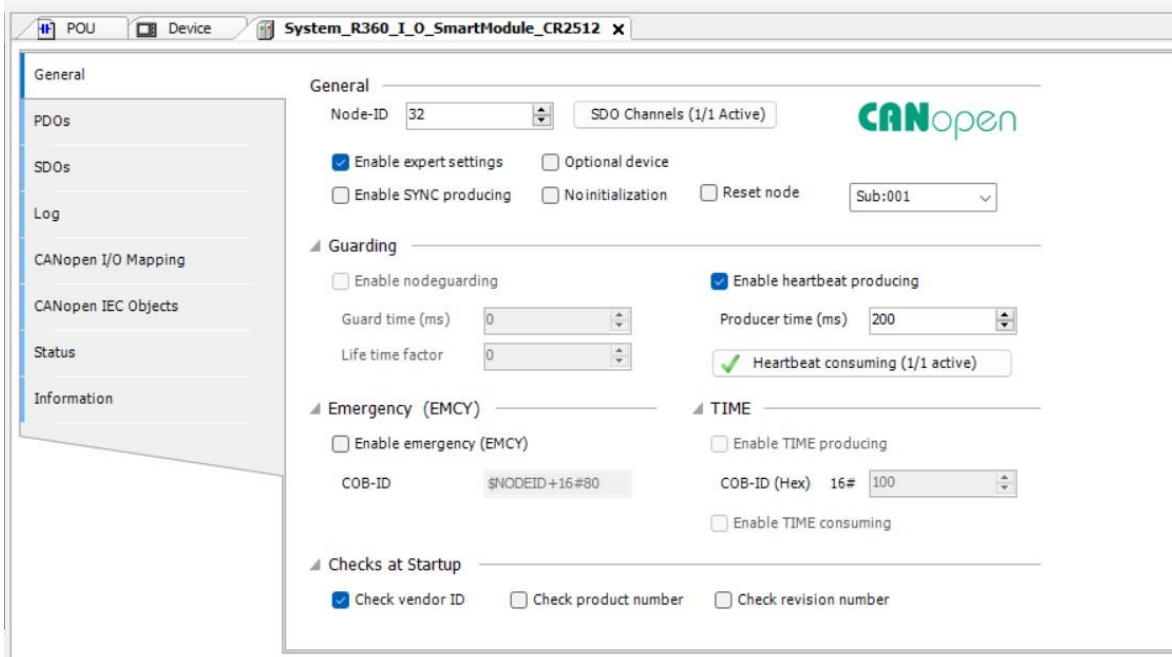


Figura 70 - Configurações dos parâmetros gerais do módulo CR2512.

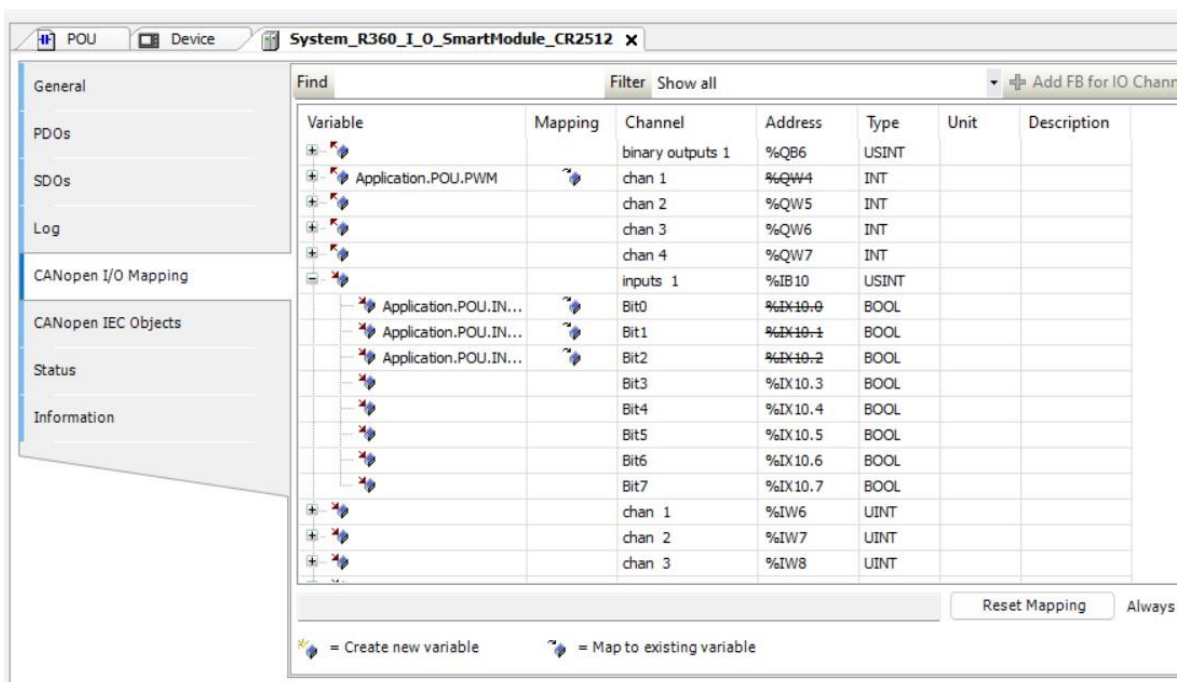


Figura 71- Mapeamento das variáveis no módulo CR2512.

A princípio, o PWM era gerado do zero pelo Codesys e enviado a partir do módulo CR2512. No entanto, devido a algumas complicações, a geração do PWM passou a ser feita pelo próprio módulo CR2512, e o Codesys apenas envia o valor do duty cycle, que recebe através do módulo CR2014 a partir do joystick\_2. O módulo CR2512 encarrega-se então do restante, em termos de temporização, além de receber o sinal dos sensores indutivos. Após a configuração geral do módulo, vou ilustrar como o PWM é ativado:

1. Aceder o módulo CR2512 e ir para SDOs

Line	Index:Subindex	Name	Value	Bit Length	Abort on Error	Jump to line
26	16#1A00:16#01	Set Mapping	16#61000110	32	<input type="checkbox"/>	<input type="checkbox"/>
27	16#1A00:16#00	Set number of mappings	16#01	8	<input type="checkbox"/>	<input type="checkbox"/>
28	16#1800:16#01	Set and enable COB-ID	16#00000181	32	<input type="checkbox"/>	<input type="checkbox"/>
29	16#1801:16#01	Disable PDO	16#80000281	32	<input type="checkbox"/>	<input type="checkbox"/>
30	16#1801:16#02	Set transmission type	16#01	8	<input type="checkbox"/>	<input type="checkbox"/>
31	16#1801:16#03	Set inhibit time	16#0000	16	<input type="checkbox"/>	<input type="checkbox"/>
32	16#1801:16#05	Set event time	16#0000	16	<input type="checkbox"/>	<input type="checkbox"/>
33	16#1A01:16#00	Clear pdo mapping	16#0	8	<input type="checkbox"/>	<input type="checkbox"/>
34	16#1A01:16#01	Set Mapping	16#64010110	32	<input type="checkbox"/>	<input type="checkbox"/>
35	16#1A01:16#02	Set Mapping	16#64010210	32	<input type="checkbox"/>	<input type="checkbox"/>
36	16#1A01:16#03	Set Mapping	16#64010310	32	<input type="checkbox"/>	<input type="checkbox"/>
37	16#1A01:16#04	Set Mapping	16#64010410	32	<input type="checkbox"/>	<input type="checkbox"/>
38	16#1A01:16#00	Set number of mappings	16#04	8	<input type="checkbox"/>	<input type="checkbox"/>
39	16#1801:16#01	Set and enable COB-ID	16#00000281	32	<input type="checkbox"/>	<input type="checkbox"/>

Figura 72 - Aceder ao SDOs do módulo CR2512.

2. Clicar em Add SDO e adiciona A frequência tendo em conta que ela vai de 20 a 250, e seleccionar o canal 1 e o canal 2 como PWM (recomendase ler o datasheet nas páginas 16 e 17)

The 'Select Item from Object Directory' dialog shows the following table of objects:

Index:Subindex	Name	Access Type	Type	Default
* 16#1800	1. transmit PDO parameter			
* 16#1801	2. transmit PDO parameter			
16#2000	IO configuration			
:16#01	chan 1	RW	USINT	2
:16#02	chan 2	RW	USINT	2
:16#03	chan 3	RW	USINT	2
:16#04	chan 4	RW	USINT	2
:16#05	chan 5	RW	USINT	2
:16#06	chan 6	RW	USINT	2
:16#07	chan 7	RW	USINT	2
:16#08	chan 8	RW	USINT	2
:16#09	chan 9	RW	USINT	1
:16#0A	chan 10	RW	USINT	1
:16#0B	chan 11	RW	USINT	1
:16#0C	chan 12	RW	USINT	1
* 16#2001:16#00	PWM frequency	RW	USINT	100
* 16#2004	Current control chan 1 2			

The 'Producer heartbeat time' dialog shows the following fields:

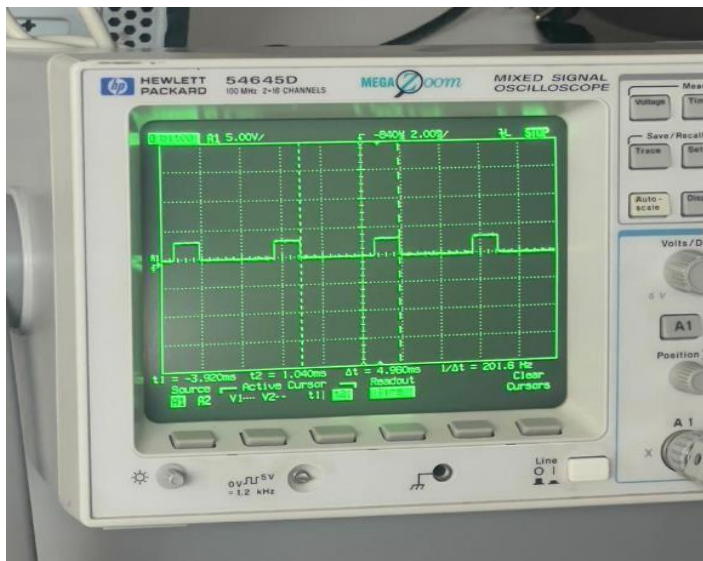
- Name: Producer heartbeat time
- Index: 16#1017
- Subindex: 16#0
- Bit length: 16
- Value: 16#0

Figura 73 - Configuração dos canais 1 e 2 como saída PWW e definição do valor da frequência.

3. Trocar para os o valor 4 nos canais 1 e 2 e definir a frequência desejada dentro da gama

Line	Index:Subindex	Name	Value	Bit Length	Abort on Error
8	16#1400:16#01	Disable PDO	16#80000220	32	<input type="checkbox"/>
9	16#1400:16#02	Set transmission type	16#01	8	<input type="checkbox"/>
10	16#1400:16#01	Set and enable COB-ID	16#00000220	32	<input type="checkbox"/>
11	16#1401:16#01	Disable PDO	16#80000320	32	<input type="checkbox"/>
12	16#1401:16#02	Set transmission type	16#01	8	<input type="checkbox"/>
13	16#1401:16#01	Set and enable COB-ID	16#00000320	32	<input type="checkbox"/>
14	16#1800:16#01	Disable PDO	16#800001A0	32	<input type="checkbox"/>
15	16#1800:16#02	Set transmission type	16#FF	8	<input type="checkbox"/>
16	16#1800:16#05	Set event time	16#0000	16	<input type="checkbox"/>
17	16#1800:16#01	Set and enable COB-ID	16#000001A0	32	<input type="checkbox"/>
18	16#1801:16#01	Disable PDO	16#800002A0	32	<input type="checkbox"/>
19	16#1801:16#02	Set transmission type	16#01	8	<input type="checkbox"/>
20	16#1801:16#05	Set event time	16#0000	16	<input type="checkbox"/>
21	16#1801:16#01	Set and enable COB-ID	16#000002A0	32	<input type="checkbox"/>
22	16#2000:16#01	chan 1	4	8	<input type="checkbox"/>
23	16#2000:16#02	chan 2	4	8	<input type="checkbox"/>
24	16#2001:16#00	PWM frequency	200	8	<input type="checkbox"/>

Figura 74 - Foi configurado os canais 1 e 2 como saída PWW e definido valor da frequência 200Hz.



Escala vertical 5V/div  
 Escala horizontal 2ms  
 Frequência 200Hz

Figura 75 - Sinal PWM no osciloscópio.

O "I/O Mapping", como o nome sugere, é um mapa que define as entradas e saídas do módulo. Nele, são alocados os pinos que serão lidos para que seus valores sejam transmitidos na aplicação "POU", onde está a programação em LD (Ladder). Todas as variáveis que recebem dados externos para o programa, como aquelas associadas aos sensores, assim como as variáveis que transmitem o PWM para o driver do motor, são mapeadas aqui, conforme mostrado nas figuras acima.

Em seguida ilustração das variáveis mapeadas nos módulos CR2512 no programa principal POU



Figura 76 - Configuração das variáveis no POU.

Os sensores indutivos são mapeados nas entradas digitais e associados a um LED. Todas as variáveis começadas com “Input\_sensor” vêm de fora do Codesys e são variáveis do tipo Booleano.

Anteriormente, o PWM era gerado por um bloco de função, como ilustrado na figura abaixo:

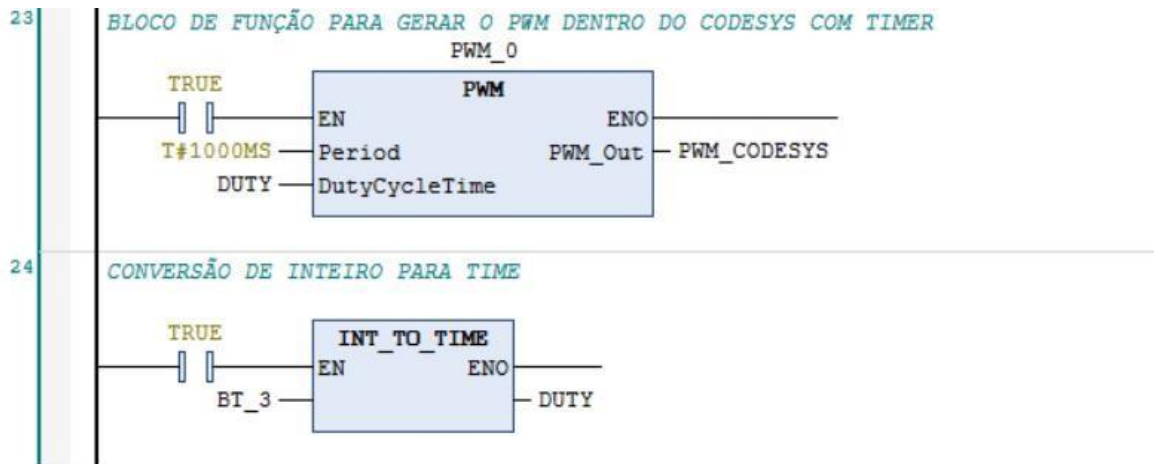
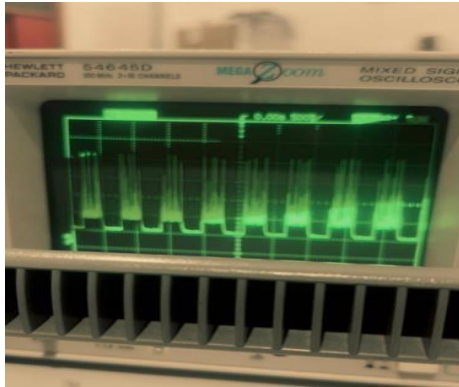


Figura 77 - Bloco de função do PWM e bloco de conversão do inteiro para time.

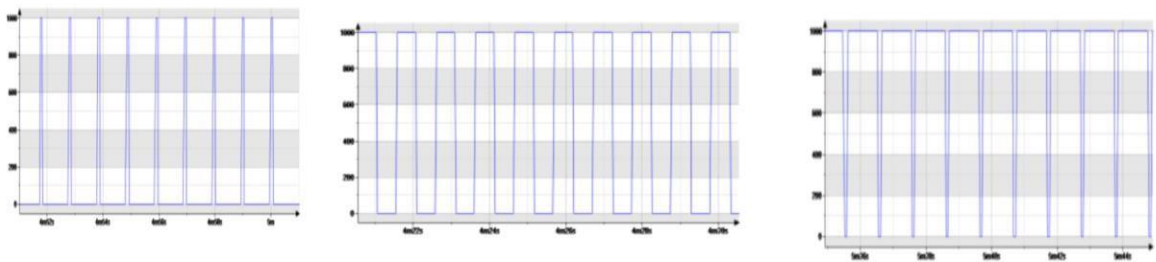
Esse método causava conflito, pois o módulo CR2512 já tinha temporizadores definidos internamente para o PWM e apenas necessitava do valor do duty cycle, considerando que os canais em uso do módulo estivessem configurados no modo PWM. Existem outros estados possíveis, como recomendado na leitura do datasheet. A seguir, apresento uma imagem do osciloscópio com a interferência:



Escala vertical 5V/div  
 Escala horizontal 2ms  
 Frequência 200Hz

**Figura 78** – Sinal PWM na prática.

Agora, utilizo o bloco de função PWM para ter uma noção de como está o sinal PWM dentro do Codesys, sem precisar ligar o osciloscópio, como se pode ver na figura abaixo:



**Figura 79** - Blocos de função responsáveis por gerar o sinal PWM no codesys.

### EcomatDisplay HMI

O processo de configuração do EcomatDisplay envolveu a transferência do pacote e das respectivas bibliotecas a partir do site da ifm. Em seguida, o dispositivo foi atualizado utilizando a aplicação “Recovery App” ao digitar o endereço “<http://192.168.82.247:8080>” no browser e arrastar o pacote anteriormente transferido para o campo destinado.



**Figura 80** - Pacote e livrarias ecomatDisplay.

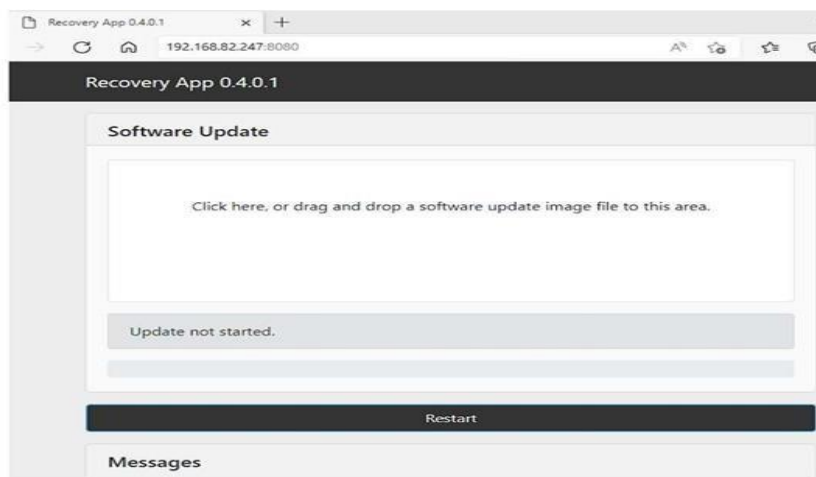


Figura 81 - Recovery App e Update to *ecomatDisplay*.

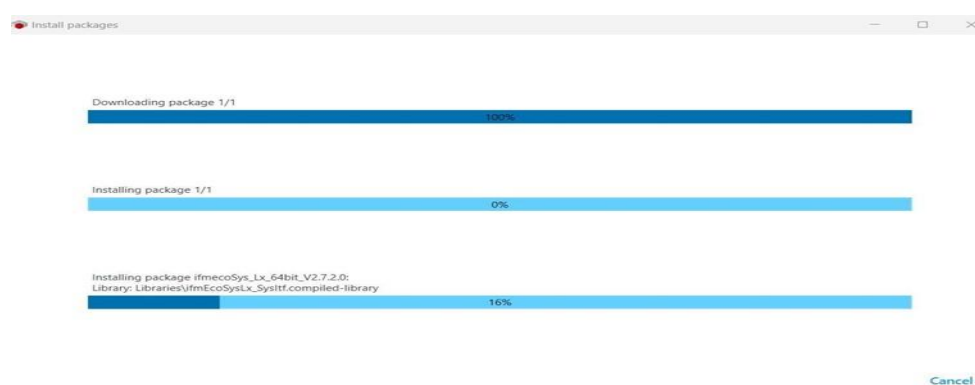


Figura 82 - Instalação do pacote e livrarias do *ecomatDisplay*.

Após todas as atualizações realizadas, é possível estabelecer a ligação entre o computador e o módulo através de um cabo Ethernet, o que permite então a conexão entre todos os dispositivos CAN e o início da programação.

### Solução Desenvolvida

Após as configurações de cada módulo, começou-se a fase de programação em LD (ladder) e FB (function block), a alocação das entradas e saídas nos módulos, e a definição dos nós de comunicação na rede para o correto funcionamento dos joysticks. Todos os valores obtidos pela leitura do joystick e sensores são apresentados num HMI EcomatDisplay ifm CR1202, que está ligado à rede CAN Bus, compartilhando a ligação com todos os módulos.

### 4.3 Controle do Motor via Joystick HJ60 e sua implementação no sistema experimental

Foi feita a ligação dos botões dos joysticks às entradas digitais do módulo CR2014. Quando o utilizador pressiona os botões, uma lâmpada acende, indicando que o botão está ativo.



Figura 83 - Programação em LADDER dos botões dos joysticks.

A captura de tela mostra a interface de usuário do software de configuração de hardware. O menu 'CANopen I/O Mapping' está selecionado. A tabela 'Find' exibe as alocações de variáveis para as entradas dos joysticks no módulo CR2014.

Variable	Mapping	Channel	Address	Type
		binary outputs 1	%QB0	USINT
		analog output 1	%QW1	UINT
		analog output 2	%QW2	UINT
		binary inputs 1	%IW0	UINT
Application.POU.INPUT...		analog input channel 5	%IW1	INT
Application.POU.INPUT_Y		analog input channel 6	%IW2	INT
Application.POU.INPUT_X		analog input channel 13	%IW3	INT
Application.POU.INPUT...		analog input channel 14	%IW4	INT

Figura 84 - Entradas dos botões dos joysticks alocadas no módulo CR2014

Para a leitura dos roletes e dos eixos X e Y, utilizaram-se as entradas analógicas do módulo CR2014. Não foi possível utilizar os movimentos dos eixos X e Y do joystick 1

devido à falta de entradas analógicas nos módulos. O eixo Y do joystick 2 é responsável pelo controlo da velocidade do motor:

+	Application.POU.INPUT...	analog input channel 5	%IW1	INT		
+	Application.POU.INPUT_Y	analog input channel 6	%IW2	INT		
+	Application.POU.INPUT_X	analog input channel 13	%IW3	INT		
+	Application.POU.INPUT...	analog input channel 14	%IW4	INT		

**Figura 85** - Entradas dos roletes e eixos X e Y dos joysticks alocados no módulo CR2014.

Para a configuração das escalas dos roletes e dos movimentos X e Y, criou-se um bloco programável com a fórmula de variáveis inteiras:

$$\text{OUT} := (\text{IN} - \text{MIN1}) * (\text{MAX2} - \text{MIN2}) / (\text{MAX1} - \text{MIN1}) + \text{MIN2}$$

Esta fórmula permite formar a escala de movimentos ao comparar os valores máximo e mínimo da voltagem de entrada e calcular uma percentagem de 0 a 100. Como o joystick\_2 está encarregue do duty cycle, que é da ordem dos 1000%, ele varia de 0 a 1000.



**Figura 86** - Barras de visualização dos movimentos do rolete e eixos X e Y do joystick 2.

Dado que os roletes e os eixos X e Y são alimentados a 5V, existe uma voltagem mínima de operação (0,53V) e uma voltagem máxima de operação (4,67V), atingindo um valor máximo de 953 e um mínimo de 104. Em termos de escala, os valores variam entre aproximadamente 956 e 105 para o movimento Y e entre 956 e 111 para o movimento X. Por esse motivo, foi necessário ajustar as escalas para que a representação corresponda ao real, de modo a que a leitura corresponda visualmente com a barra inserida, pois algumas barras são percentagens e outras em milhar.

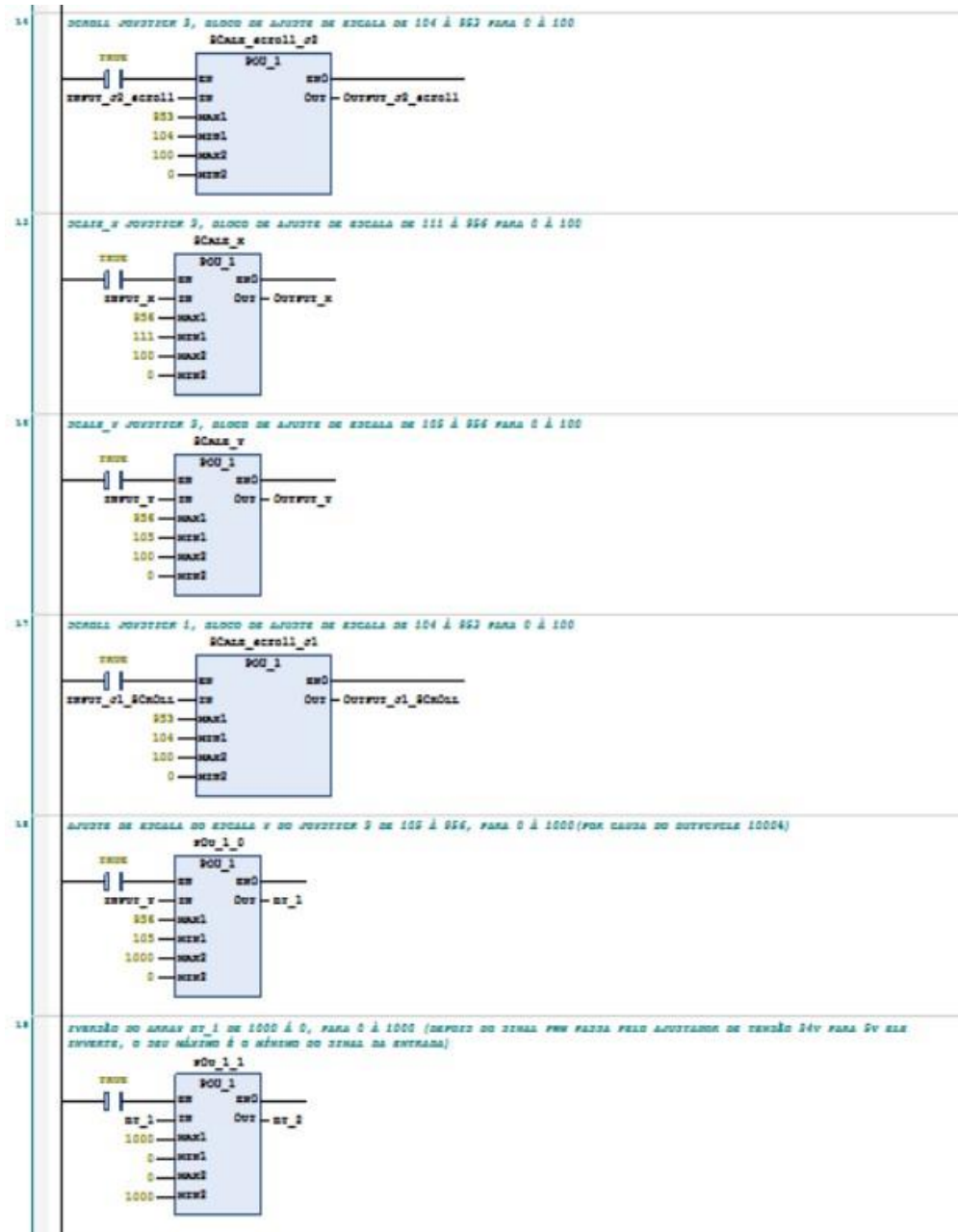


Figura 87 - Blocos de ajuste escala dos roletes e eixos X e Y.

Nota: O último bloco na figura é de inversão, pois o optoacoplador aplica uma inversão antes do sinal chegar ao driver Mosfet Pulso Trigger Controller. Para que o sinal seja bem representado na barra, foi necessário fazer uma inversão de escala no Codesys também.

Como a escala de 0 a 5V corresponde a valores de 0 a 1024, foi feito um ajuste para valores entre 104 e 953, o que equivale a uma resolução de 10 bits.

Por último, foram adicionados sensores indutivos e de velocidade, que permitem a detecção de metais próximos, sinalizando esta proximidade através de uma lâmpada na

interface. No caso do sensor de velocidade, este sinaliza a rotação também através de uma lâmpada na interface.



**Figura 88** - Programação LADDER dos sensores indutivos.



**Figura 89** - Figura representativa dos sensores na interface.

## Interface do HMI

A interface permite a visualização das ações do utilizador em tempo real, nomeadamente a pressão dos botões, os roletes e o movimento do joystick 2 nos eixos X e Y.

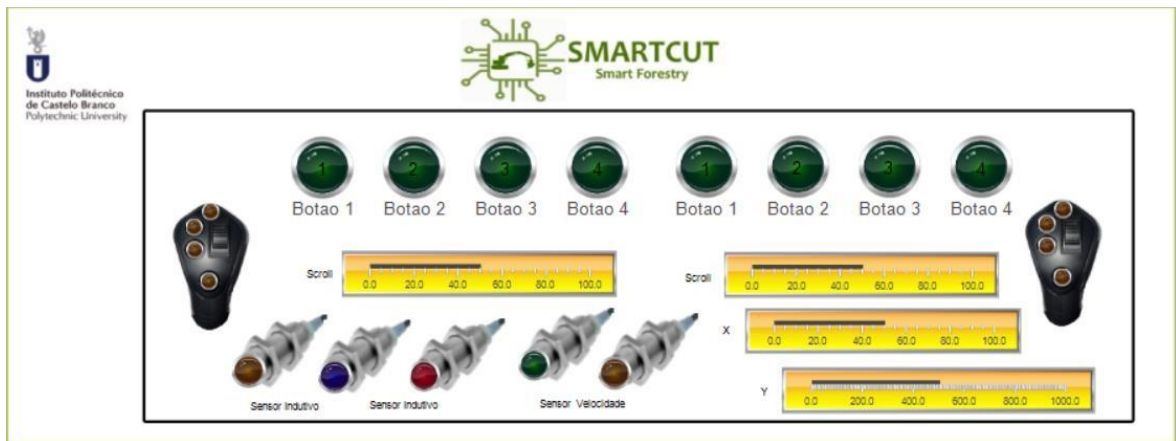


Figura 90 - Interface do ecomatDisplay.

A interface gráfica foi criada utilizando um template com as dimensões do ecrã, no qual foram inseridos os objetos de visualização disponíveis nas bibliotecas do Codesys, incluindo lâmpadas para representar os botões e barras para representar os níveis percentuais de movimento dos roletes e dos eixos X e Y. Foram ainda adicionadas imagens dos joysticks e do sensor para uma visualização secundária.

ID	File name	Image	Link type
1	HMI SMARTCUT WALLPAPER.png		Embedded
2	J1_transparent.png		Embedded
4	J2_transparent.png		Embedded
5	SMARTCUT.png		Embedded
6	sensor.png		Embedded

Figura 91- Livraria de imagens adicionadas.

## 4.4 Desafios e Soluções

### 1. Driver de Motor BTS7960 - Problemas:

- Operação com uma tensão de 5V para o PWM, enquanto a fornecida era de 24V.
- Após a instalação de um optoacoplador, houve problemas no reconhecimento do PWM, ou seja, mesmo com todas as ligações feitas, o motor DC não funcionava.

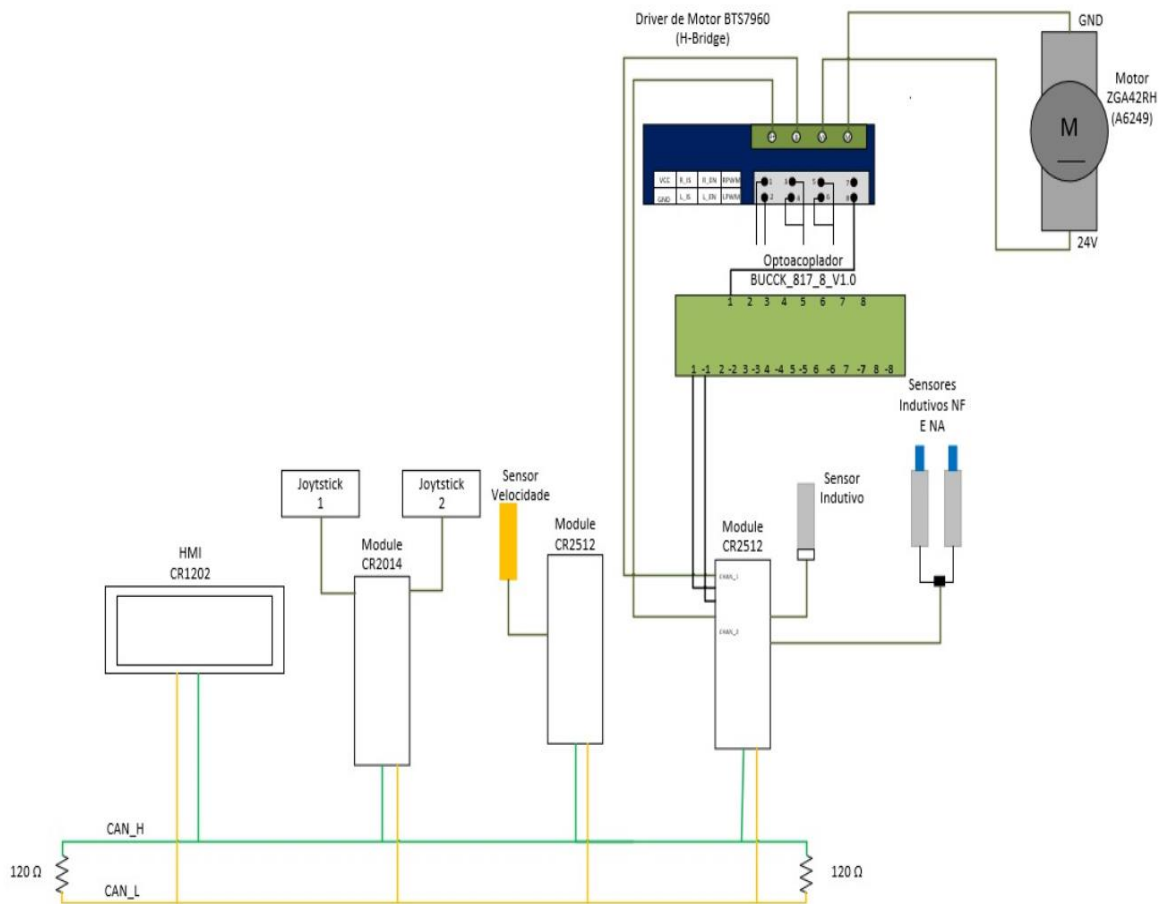


Figura 92 - Esquema do sistema experimental\_1.

**Solução:** Implementou-se posteriormente o Driver MOSFET Pulso Trigger Controller, o que resolveu o problema.

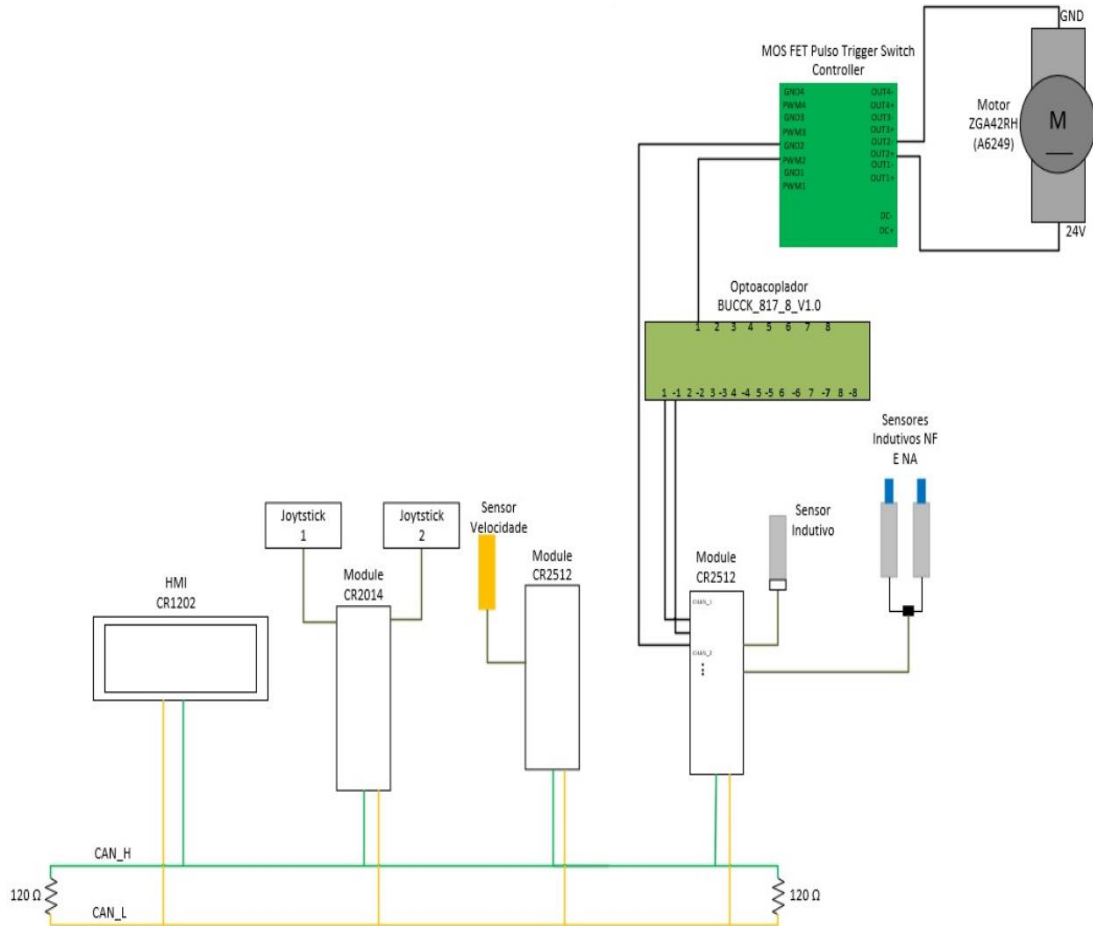
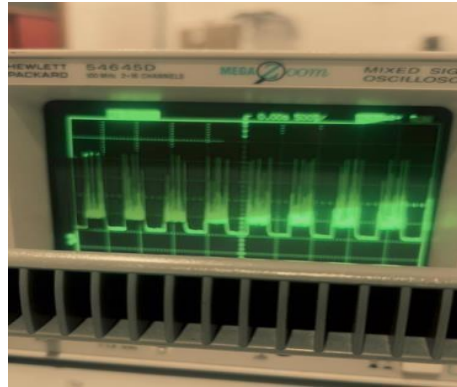


Figura 93 - Esquema do sistema experimental\_2.

### Geração do PWM:

Inicialmente, o PWM era gerado do zero pelo Codesys e transmitido a partir do canal 1 (CHA 1) do módulo CR2512. No entanto, o sinal saía distorcido, pois o módulo já estava configurado para gerar o PWM, ou seja, já tinha um temporizador associado. Assim, o sinal de saída apresentava muito ruído e nunca atingia o pico desejado de 24V.



**Figura 94** - Sinal PWM a saída do módulo.

Desenvolveu-se então um bloco de função utilizando um temporizador TON disponível no Codesys. Este código PWM foi desenvolvido para o Codesys, uma plataforma amplamente usada para programação de Controladores Lógicos Programáveis (CLPs), com estrutura e sintaxe característica da linguagem estruturada (ST) do IEC 61131-3, padrão para CLPs.

**Temporizadores TON:** No Codesys, a função TON é um temporizador do tipo "on delay" (atraso para ligar). Ele conta o tempo desde que a entrada IN está ativa até que o tempo programado em PT seja atingido. Quando o temporizador atinge o tempo programado, o bit Q é definido para TRUE, indicando que o tempo expirou, o que foi útil para definir o tempo de ativação da onda (duty cycle).

Definiram-se as variáveis de entrada, saída e internas:

- VAR\_INPUT: variáveis como Period e DutyCycleTime;
- VAR\_OUTPUT: variável de saída, PWM\_Out;
- VAR: variáveis internas usadas dentro do bloco, como Timer\_ON, Timer\_OFF, OffTime, e CycleState.

Nota: Timer\_ON e Timer\_OFF são blocos de função TON disponíveis no Codesys.

Especificou-se que, nos casos de duty cycle de 0% ou de 100%, o PWM será igual a 0 ou 24V, respetivamente, dado que a tensão de entrada é de 24V.

Para calcular o tempo de desativação da onda (OffTime), aplicou-se a seguinte fórmula:

$$\text{OffTime} := \text{Period} - \text{DutyCycleTime}$$

**CycleState:** O ciclo PWM é implementado controlando o estado de CycleState, que alterna entre FALSE (ciclo desligado) e TRUE (ciclo ligado). Isso reflete o comportamento típico de um PWM, onde o sinal alterna entre ligado e desligado dentro de cada período.

Como utilizei variáveis do tipo temporizador e o duty cycle é definido pelo joystick, há um bloco de função no programa principal (POU) que recebe o valor inteiro e o converte em tempo. Este bloco, no entanto, tem limitações, uma vez que o Codesys não opera com tempos inferiores a milissegundos, o que impede a operação em períodos de tempo menores.

### Código usado:

```

FUNCTION_BLOCK PWM

VAR_INPUT

    Period : TIME := T#10MS;           // Tempo total do ciclo
    (período do PWM)

    DutyCycleTime : TIME := T#50ms;    // Tempo ligado (duty cycle
    em tempo)

END_VAR

VAR_OUTPUT

    PWM_Out : INT;                     // Saída PWM

END_VAR

VAR

    Timer_ON : TON;                    // Temporizador para tempo
    ligado

    Timer_OFF : TON;                   // Temporizador para tempo
    desligado

    OffTime : TIME;                    // Tempo desligado (calculado
    a partir do período -
    DutyCycleTime)

    CycleState : BOOL := FALSE;        // Estado do ciclo (TRUE =
    ligado, FALSE = desligado)

END_VAR

// Calcular o tempo desligado

OffTime := Period - DutyCycleTime;

// Casos especiais: 0% ou 100% de duty cycle

```

```
IF DutyCycleTime <= T#0MS THEN
    PWM_Out := 0; // Duty cycle 0% -> Saída sempre OFF
ELSIF DutyCycleTime >= T#1000MS THEN
    PWM_Out := 24; // Duty cycle 100% -> Saída sempre ON
ELSE
    // Lógica do ciclo PWM
    IF NOT CycleState THEN
        // Temporizador para o tempo ligado (ON)
        Timer_ON(IN := TRUE, PT := DutyCycleTime);
        IF Timer_ON.Q THEN
            PWM_Out := 0; // Saída alta
            CycleState := TRUE; // Muda para o estado ligado
            Timer_ON(IN := FALSE); // Reinicia temporizador ligado
        END_IF
    ELSE
        // Temporizador para o tempo desligado (OFF)
        Timer_OFF(IN := TRUE, PT := OffTime);
        IF Timer_OFF.Q THEN
            PWM_Out := 1000; // Saída baixa
            CycleState := FALSE; // Muda para o estado desligado
            Timer_OFF(IN := FALSE); // Reinicia temporizador
        END_IF
    END_IF
END_IF
```

## Solução para gerar o PWM a partir do módulo CR2512

Em vez de criar o PWM do zero, agora é necessário apenas enviar o valor do duty cycle (explicado acima com mais detalhes), seguindo as orientações do datasheet do módulo CR2512 sobre o PWM. Para tal, deve-se criar o seguinte bloco de função com o código:

```
FUNCTION_BLOCK PWM_CORRENT

VAR_INPUT

    PWM_Input : INT; (* Valor do PWM diretamente em milésimos (-
1000 a 1000) *)

    ChannelOdd : BOOL; (* True se for canal ímpar *)

END_VAR

VAR_OUTPUT

    PWM_Value : INT; (* Valor de saída para o canal correspondente
*)

END_VAR

(* Arredondamento e limitação de valores *)

IF PWM_Input > 1000 THEN

    PWM_Input := 1000; (* Limita o valor a +1000 *)

ELSIF PWM_Input < -1000 THEN

    PWM_Input := -1000; (* Limita o valor a -1000 *)

END_IF;

(* Desativação entre -50 e +50 *)

IF (PWM_Input > -50) AND (PWM_Input < 50) THEN

    PWM_Value := 0; (* Saída desativada *)

ELSE

    IF ChannelOdd THEN

        (* Canais ímpares: apenas valores positivos entre +50 e
+1000 *)
```

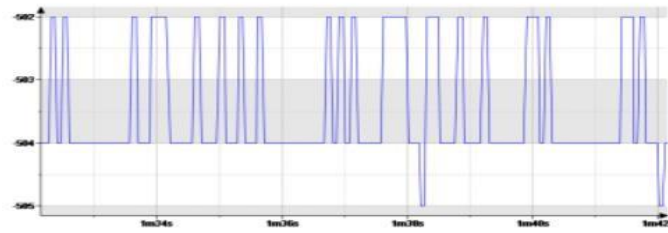
```

IF PWM_Input >= 50 THEN
    PWM_Value := PWM_Input; (* Mantém o valor para canais
ímpares *)
ELSE
    PWM_Value := 0; (* Desativa se o valor for fora do
intervalo *)      END_IF;.
ELSE
    (* Canais pares: apenas valores negativos entre -1000 e -50
*)
    IF PWM_Input <= -50 THEN
        PWM_Value := PWM_Input; (* Mantém o valor para canais
pares *)
    ELSE
        PWM_Value := 0; (* Desativa se o valor for fora do
intervalo *)      END_IF;
    END_IF;
END_IF;

```

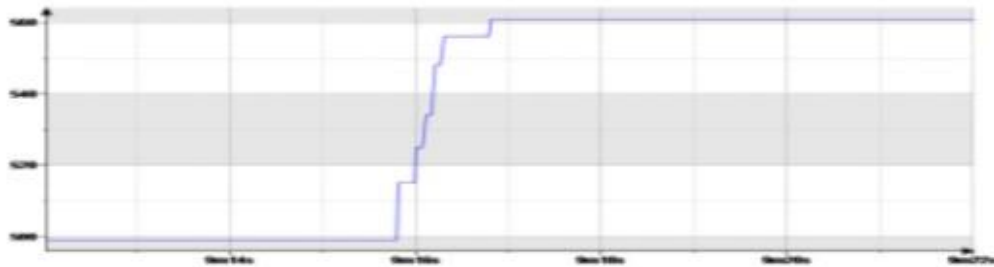
O duty cycle é de 1000%, e temos uma zona morta de -50 a +50, ou seja, o motor só começa a funcionar fora dessa zona. Dispomos de dois canais, 1 e 2, para facilitar o controlo do motor nos sentidos horário e anti-horário. Os canais são definidos com base numa variável de entrada chamada "ChannelOdd": quando esta variável está em TRUE, o canal 1 entra em ação; quando em FALSE, o canal 2 assume o controlo. A saída será um valor inteiro, pois a ideia é identificar qual canal está a ser utilizado. O canal 1 trabalha com números positivos de 0 a 1000, enquanto o canal 2 trabalha com números negativos, de 0 a -1000, respeitando a zona morta de -50 a -1000 no canal 2 e de 50 a 1000 no canal 1.

3- Definir o *dutycycle* com 0 *joistcker* sem filtro



**Figura 95** - Sinal Enviado pelo joystick antes do filtro.

*O dutycycle* apresentava muita oscilação, não um único número como tinha que ser.



**Figura 96** - Sinal Enviado pelo joystick depois do filtro ser implementado.

Após a implementação de um filtro, o sinal tornou-se estável e passou a ser uma linha reta.

### **Solução filtro**

Aqui está o código usado:

```
FUNCTION_BLOCK POU_2
```

```
VAR_INPUT
```

```
    InputValue : INT;           // Valor de entrada (inteiro)
```

```
    Deadband : INT := 1;       // Largura da banda morta (deadband, inteiro)
```

```
END_VAR
```

```
VAR_OUTPUT
```

```
    FilteredValue : INT;        // Valor de saída filtrado (inteiro)
```

```
END_VAR
```

```
VAR
```

```
LastValue : INT;           // Último valor aceito para a saída
    Initialized : BOOL := FALSE; // Verifica se o filtro foi
inicializado
END_VAR
// Inicialização do valor na primeira execução
IF NOT Initialized THEN
    LastValue := InputValue;
    Initialized := TRUE;
END_IF;
// Ignora valores de entrada negativos
IF InputValue >= 0 THEN
    // Verifica se a variação está fora da banda morta (deadband)
    IF ABS (InputValue - LastValue) > Deadband THEN
        // Atualiza o valor filtrado apenas se a variação for maior
que o deadband
        LastValue := InputValue;
    END_IF;
END_IF;
// Define o valor filtrado como o último valor aceito FilteredValue
:= LastValue;
```

Este código implementa um filtro simples com uma banda morta para suavizar as variações do valor de entrada. Ele verifica se a variação do valor de entrada em relação ao último valor aceito está dentro de um intervalo permitido, que é definido pela banda morta. Caso a variação seja pequena (ou seja, dentro da banda morta), o valor de saída não é atualizado, permanecendo igual ao último valor aceito.

Explicação passo a passo:

#### 1. **Inicialização:**

Na primeira execução do código (IF NOT Initialized THEN), o valor de LastValue é definido como o valor de InputValue e a variável Initialized é

marcada como verdadeira. Isso garante que o filtro seja inicializado apenas uma vez.

## 2. **Filtragem:**

O código ignora qualquer valor de entrada negativo (IF InputValue >= 0). Assim, se InputValue for negativo, não haverá atualização. Em seguida, ele verifica se a diferença entre o valor de entrada e o último valor aceito (LastValue) é maior do que o valor da banda morta (Deadband). Se a diferença for maior que a banda morta, o valor de saída (FilteredValue) é atualizado para o novo valor de entrada (LastValue := InputValue). Caso contrário, ele mantém o valor anterior.

## 3. **Saída:**

O valor de saída (FilteredValue) será o último valor de entrada aceito, ou seja, aquele que passou pela verificação da banda morta.

Exemplo:

### **Parâmetros:**

- InputValue = 10
- Deadband = 2

### **Primeira execução:**

Como Initialized = FALSE, o valor de LastValue será definido como 10, e FilteredValue também será 10.

### **Segunda execução (com InputValue = 11):**

A diferença entre InputValue (11) e LastValue (10) é 1, que é menor que a banda morta (2), logo LastValue não é atualizado, e FilteredValue permanece 10.

### **Terceira execução (com InputValue = 13):**

A diferença agora é 3 (13 - 10), que é maior que a banda morta (2). Portanto, LastValue será atualizado para 13, e FilteredValue também será 13.

4- Leitura do joystick

Solução bloco de função, código usado:

```
FUNCTION_BLOCK POU_1
```

```
VAR_INPUT
```

```
IN: INT;
```

```
MAX1: INT;
```

```

MIN1: INT;
MAX2: INT;
MIN2: INT;
END_VAR
VAR_OUTPUT
OUT: INT;
END_VAR
VAR
END_VAR
OUT := (IN-MIN1) * (MAX2-MIN2) / (MAX1-MIN1) + MIN2;

```

O bloco de função POU\_1 realiza uma interpolação linear para converter um valor de entrada (IN) de uma faixa original para outra faixa desejada. Ele redimensiona IN, mantendo a proporção entre as faixas.

#### **Fórmula:**

$$OUT := (IN - MIN1) * (MAX2 - MIN2) / (MAX1 - MIN1) + MIN2$$

- **IN:** Valor de entrada.
- **MIN1, MAX1:** Definem a faixa original.
- **MIN2, MAX2:** Definem a faixa de destino.
- **OUT:** Resultado convertido.

#### **Como funciona:**

- **Subtração de MIN1:** Ajusta IN para o início da faixa original.
- **Proporção:** A multiplicação e divisão ajustam o valor de IN à proporção da nova faixa.
- **Adição de MIN2:** Realinha o valor ao início da nova faixa.

Substituindo os valores:

$$OUT := (500 - 0) * (10 - 0) / (1000 - 0) + 0;$$

$$OUT := 500 * 10 / 1000 + 0;$$

OUT := 5000 / 1000;

OUT := 5;

- **IN = 500**: Valor medido pelo sensor, ou qualquer outro valor a ser convertido.
- **MAX1 = 1000**: Valor máximo possível na faixa de entrada (por exemplo, a leitura máxima do sensor).
- **MIN1 = 0**: Valor mínimo possível na faixa de entrada (por exemplo, a leitura mínima do sensor).
- **MAX2 = 10**: Valor máximo na nova faixa, onde o valor convertido será aplicado (por exemplo, 10 metros).
- **MIN2 = 0**: Valor mínimo na nova faixa (por exemplo, 0 metros).

## 5. Resultados e Discussões

### 5.1 Introdução dos Resultados

Neste capítulo, são apresentados e analisados os resultados obtidos a partir dos testes realizados com o protótipo do sistema de monitorização e controle do harvester, implementado no ambiente de simulação **Codesys** e no ambiente físico com o protótipo. Serão discutidos os aspectos relacionados ao desempenho do controle via **joystick HJ60**, à comunicação através da rede **CAN**, à precisão dos sinais **PWM** e às funcionalidades da interface **HMI**. Além disso, serão exploradas as limitações e as sugestões para melhorias futuras.

### 5.2 Resultados no Ambiente de Simulação

#### 5.2.1 Desempenho do Controle via Joystick

- **Leitura dos Sinais do Joystick HJ60:**



Figura 97 - Testando na prática e no codesys o botão 1 do joysticker\_1.

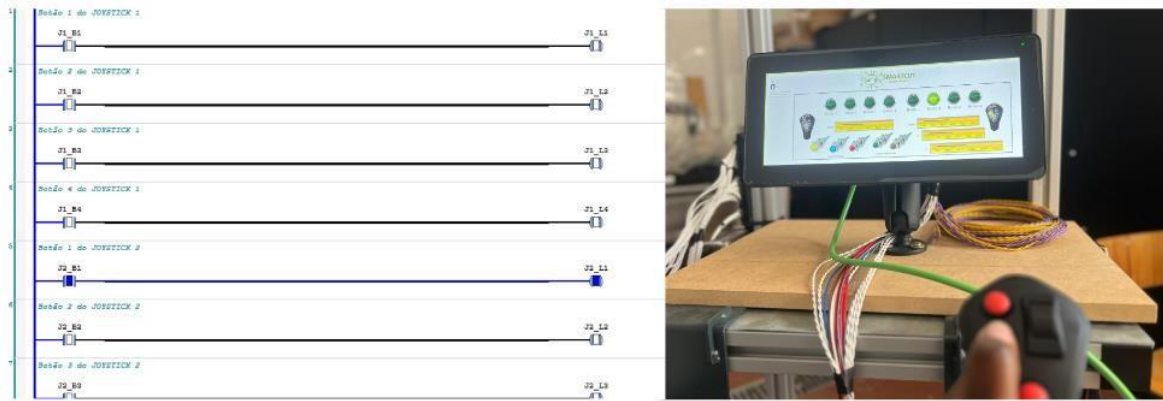


Figura 98 - Testando na prática e no codesys o botão 2 do joysticker\_2.

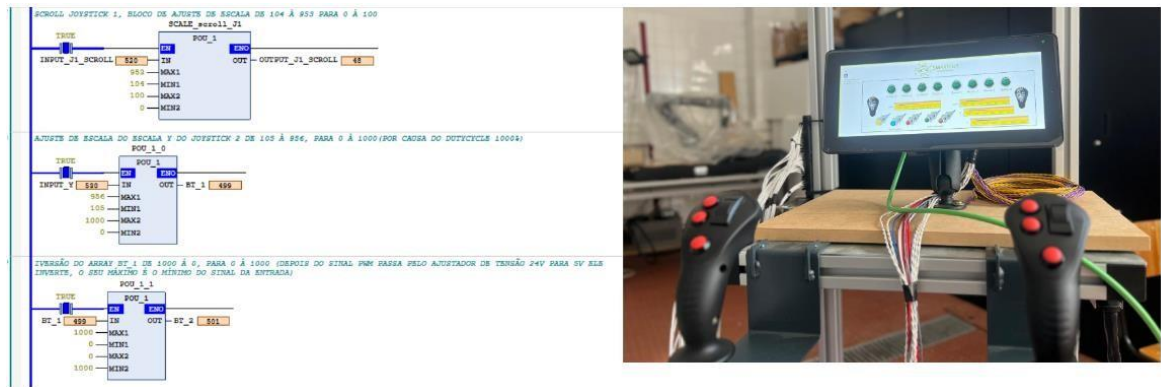


Figura 99 - Blocos de função para ajuste do scroll-j1, scale-Y e bloco de inversão de valor representação no prático.

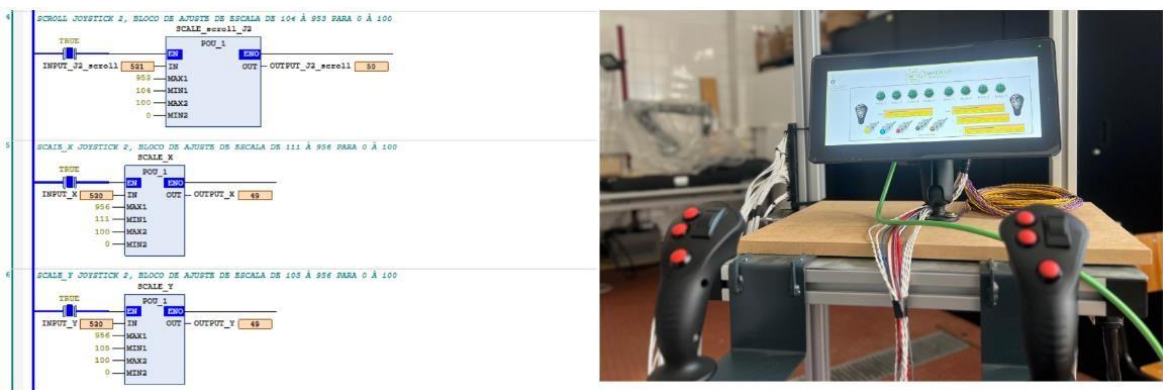


Figura 100 - Blocos de função para ajuste do scroll-j2, scale-Y e scale-X representação no prático.

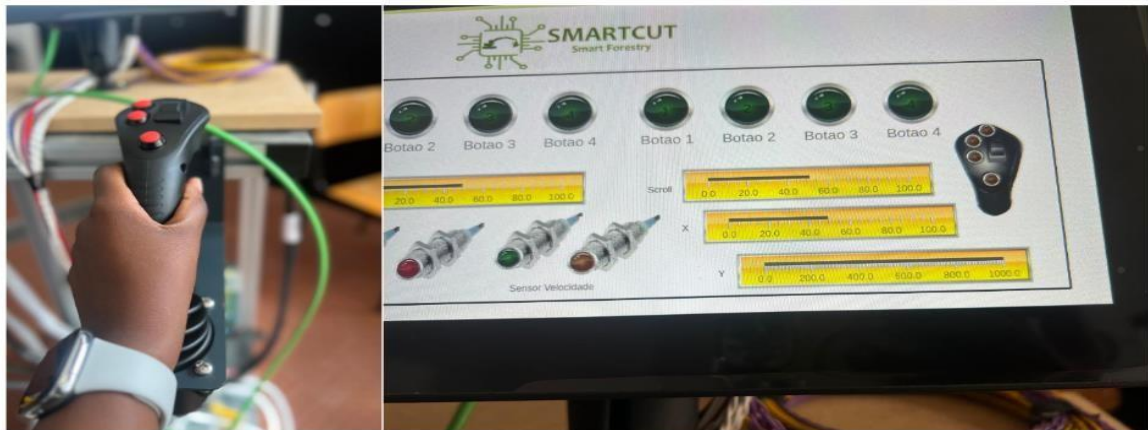


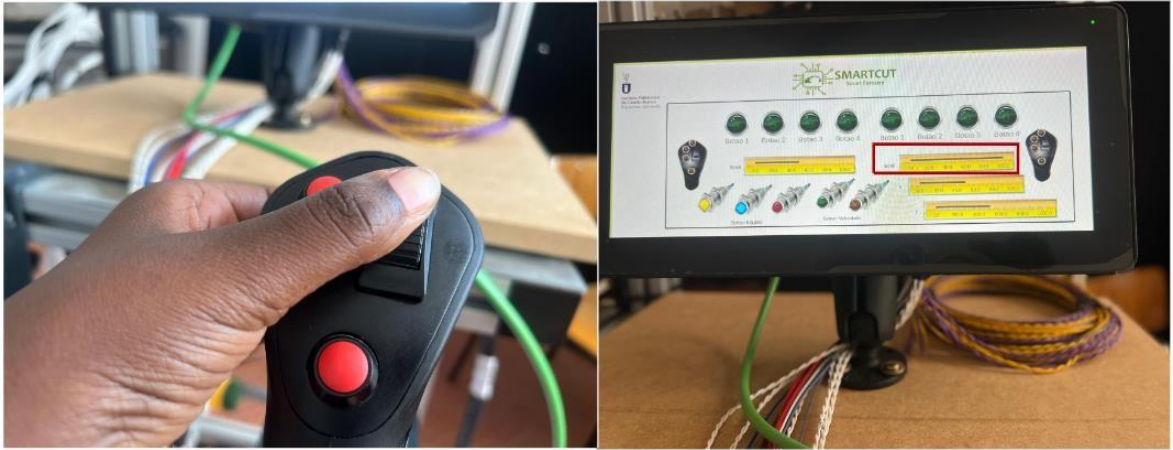
Figura 101 - Testando a scale-Y joytker\_2 na prática no máximo.



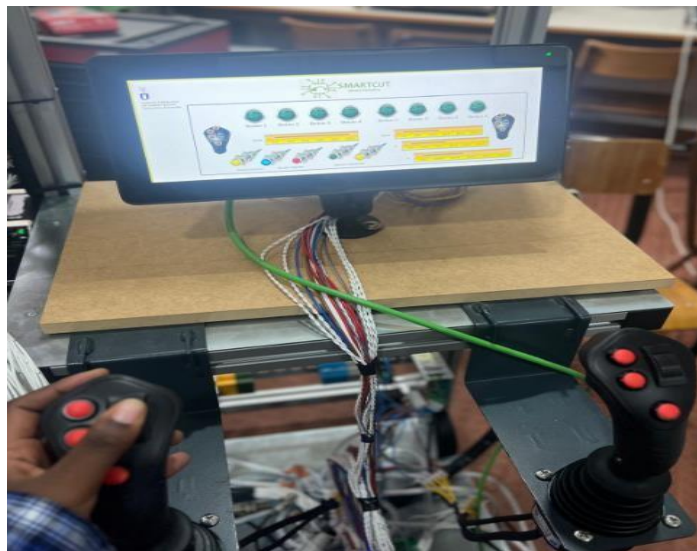
Figura 102 - Testando a scale-Y joytker\_2 na prática no mínimo.



Figura 103 - Testando a scroll joytker\_2 na prática no mínimo.



**Figura 104** - Testando a scroll joytker\_2 na prática no máximo.



**Figura 105** - Testando a scroll joytker\_1 na prática.

- **Leitura dos Sinais dos sensores Indutivos:**



**Figura 106** - Testando o sensor indutivo NF na prática, o led acende quando distante do metal.



**Figura 107** - Testando o sensor indutivo NF na prática, o led apaga quando próximo do metal.



Figura 108 - Testando o sensor indutivo NA, na prática o led apaga quando distante do metal.

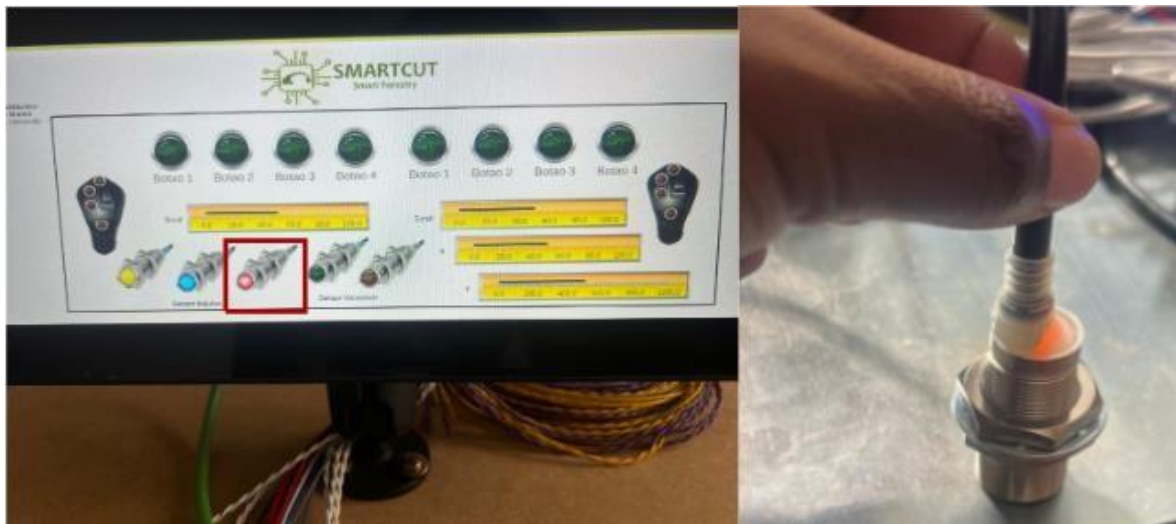


Figura 109 - Testando o sensor indutivo NA, na prática o led acende quando próximo do metal.

- **Leitura dos Sinais do sensor velocidade:**

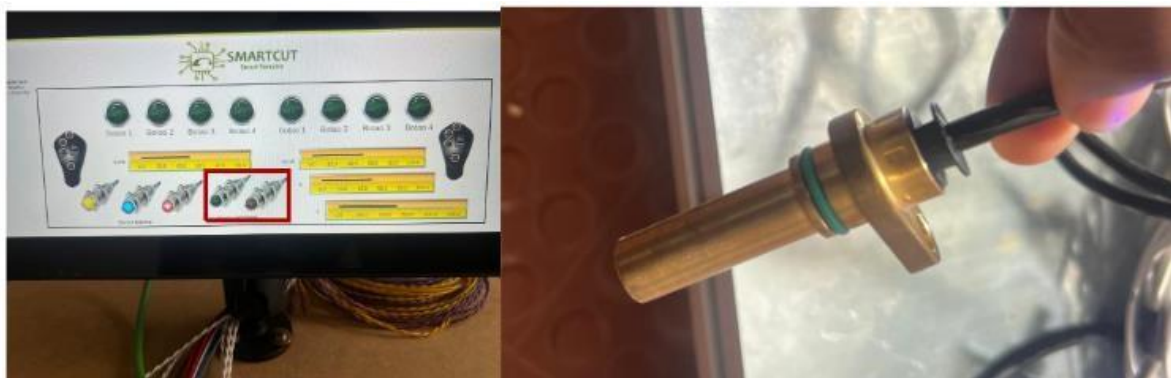
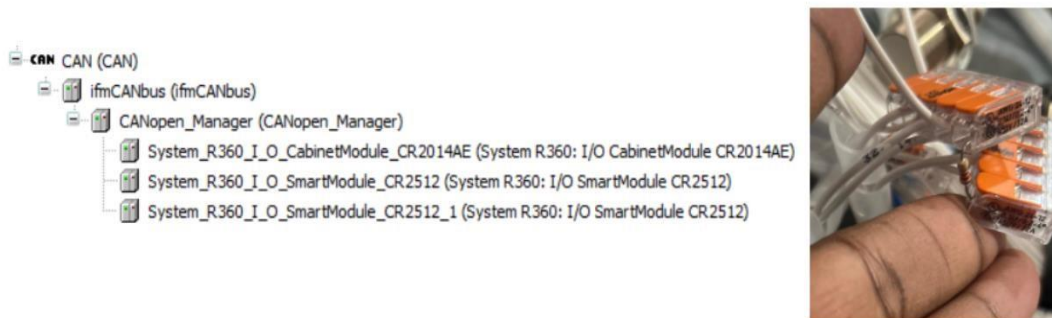


Figura 110 - Testando o sensor de velocidade na prática, o led apaga quando distante do metal.



**Figura 111** - Testando o sensor de velocidade na prática, os leds oscilam entre si quando próximo do metal

## 5.2.2 Eficácia da Comunicação CANopen e CANBUS



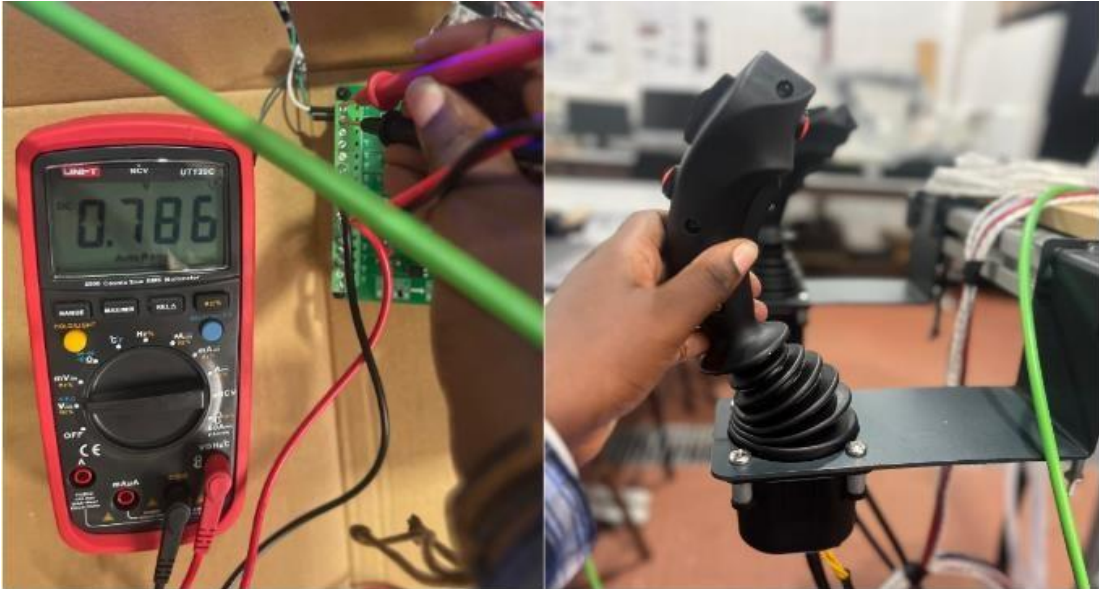
**Figura 112** - Rede CANbus e CANopen na prática

- **Taxa de Transmissão de Dados:**
  - A comunicação entre os módulos via **CANopen** e CANbus funcionou de forma estável e confiável, com uma taxa de transmissão de dados adequada para as operações em tempo real do sistema.
  - Não foram detectados problemas de perda de pacotes ou falhas na sincronização, o que é essencial para garantir a integridade dos dados trocados entre os módulos do sistema.
- **Integração e Sincronização dos Módulos:**
  - A integração dos módulos **CR2014**, **CR2512**, e **CR1202** demonstrou ser robusta, permitindo que o sistema operasse como uma unidade coesa.
  - A sincronização de dados e o feedback em tempo real entre os componentes garantiram um desempenho confiável do protótipo.

## 5.3 Resultados no Ambiente Físico

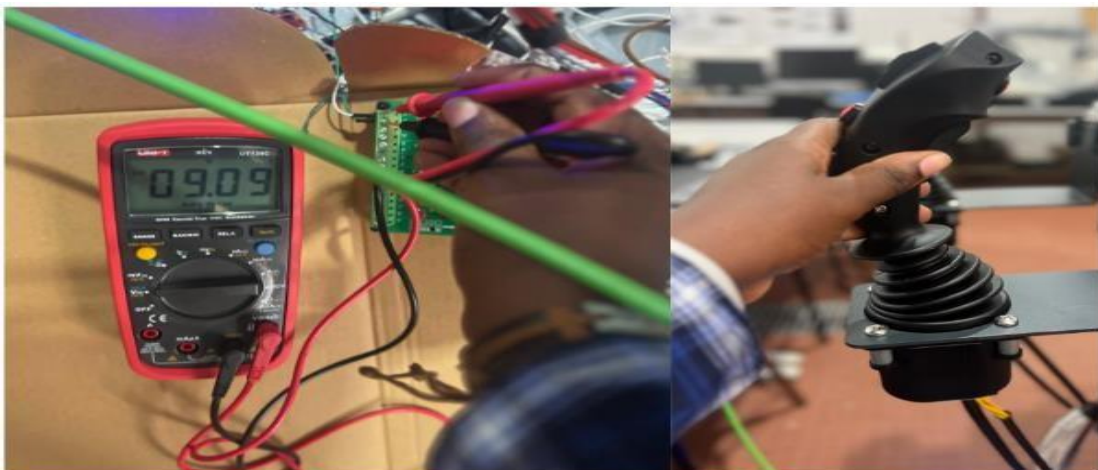
### 5.3.1 Testes de Funcionamento do Protótipo

- **Desempenho do Motor Controlado pelo CR2512:**



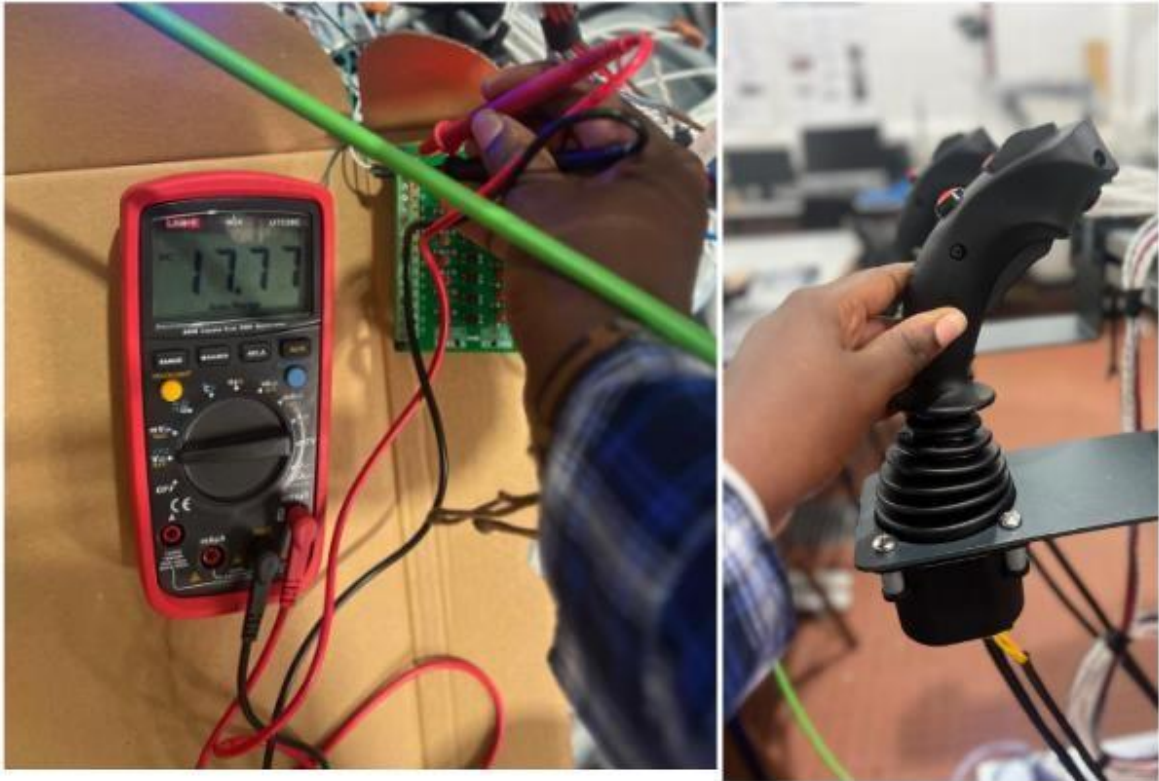
**Figura 113** - Demonstração 1 da variação da tensão entregue ao optoacoplador com o manuseio do joytik\_2.

Joystick no mínimo a tensão entregue é 0 V.



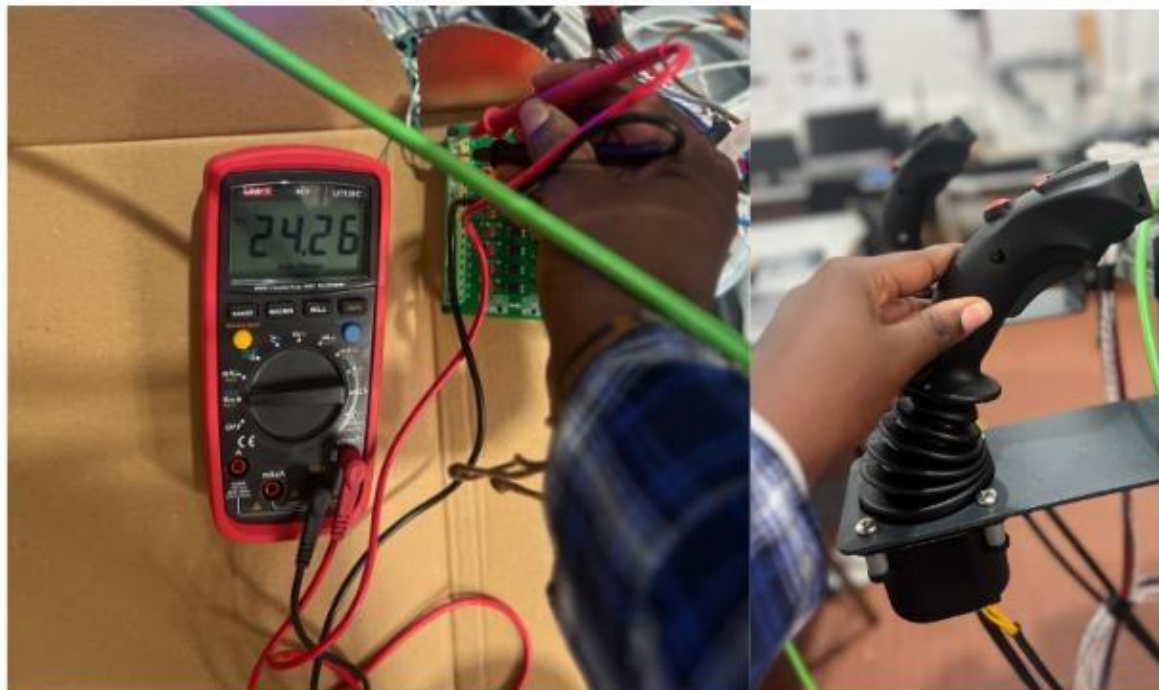
**Figura 114** - Demonstração 2 da variação da tensão entregue ao optoacoplador com o manuseio do joytik\_2.

Tensão entregue de 9 V, o motor continua a girar em uma velocidade mínima.



**Figura 115** - Demonstração da variação da tensão entregue ao optoacoplador com o manuseio do joytik\_2.

Tensão entregue de 17.77V, o motor está quase atingindo a tensão máxima aceitável por ele.



**Figura 116** - Demonstração 4 da variação da tensão entregue ao optoacoplador com o manuseio do joytik\_2.

Tensão entregue de 24V, o motor já atingiu a tensão máxima aceitável por ele.



**Figura 117** - Demonstração 1 da variação da tensão entregue ao driver mosfet com o manuseio do joytik\_2.



**Figura 118** - Demonstração 2 da variação da tensão entregue ao driver mosfet com o manuseio do joytik\_2.

No ambiente físico, o protótipo funcionou conforme esperado, com o CR2512 a gerar sinais PWM que controlaram o motor diretamente. Mesmo sem a lâmina de corte, os testes simularam condições de operação próximas ao cenário real de um "harvester".

O sistema respondeu de forma precisa aos comandos de controlo enviados pelo joystick, com o protótipo a demonstrar um controlo estável e eficiente do motor.

- **Teste das tensões na fonte de 24V e 5V**



**Figura 119** - Teste na fonte de 5V.



**Figura 120** - Teste na fonte de 24V.

O programa executa em loop infinito após seguir os seguintes passos:

- 1- Acesse o dispositivo `ecomatdisplay_64bit`.

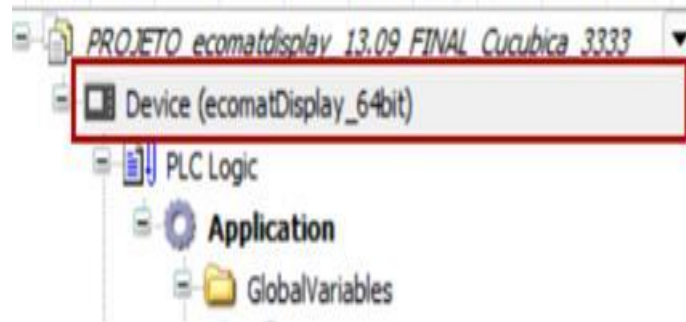


Figura 121 - Device (ECOMATdisplay) no codesys.

2- Acesse a rede SCAN Network e associe o módulo CR2512.

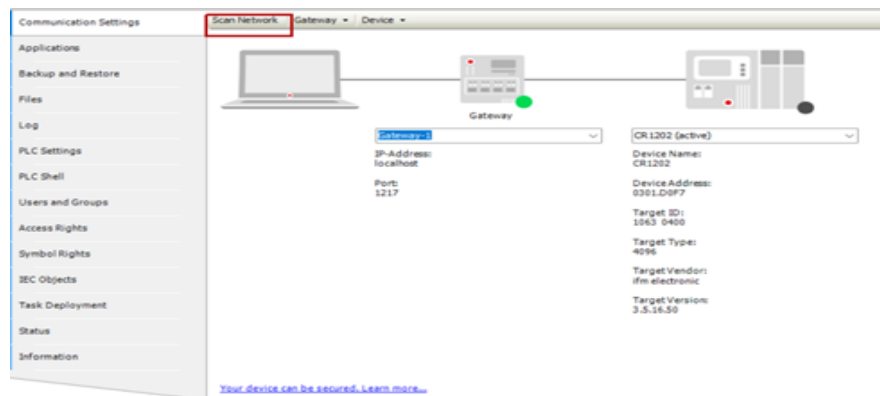


Figura 122 - Scan network no device ecomatdisplay.

3- Clique na opção "Online" e faça o login.

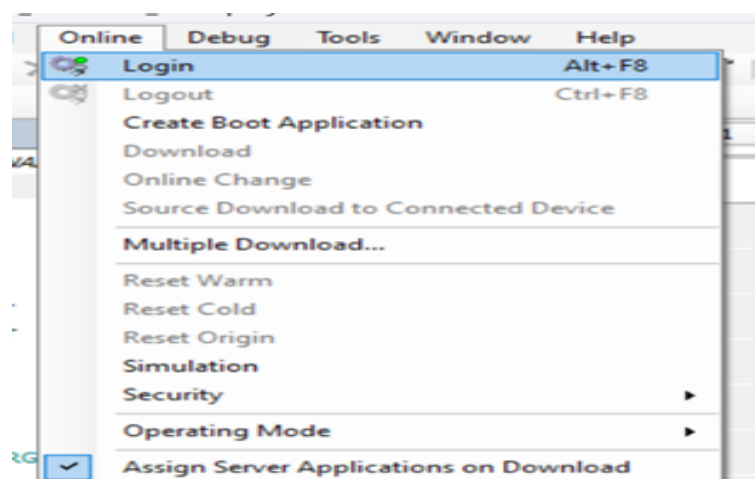


Figura 123 - Fazendo o login.

4- Acesse "Debug" e clique em "Start".

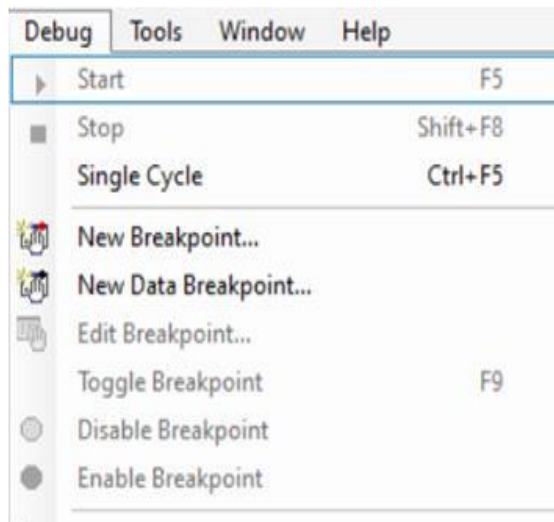


Figura 124 - Entrando no Debug.

Nota: Isso é relevante caso esteja a utilizar o ecomatDisplay HMI. Se estiver a simular dentro do CODESYS, que possui uma interface interna, o procedimento é diferente.

Após seguir esse procedimento, o programa começará a correr em um ciclo infinito. Com as variáveis já definidas e a lógica de programação previamente descrita implementada, pode-se dar início à fase de testes do programa.

## 5.4 Discussões dos Resultados

### 5.4.1 Análise Comparativa entre o Ambiente de Simulação e o Ambiente Físico

A análise comparativa entre os dois ambientes revelou que o desempenho geral do sistema foi consistente, embora os testes físicos tenham apresentado desafios adicionais relacionados à variabilidade dos sinais.

O ambiente de simulação permitiu uma validação inicial eficiente, mas alguns ajustes específicos só puderam ser identificados e implementados quando o sistema foi testado em condições físicas reais no caso dos joysticks.

### 5.4.2 Contribuições para a Eficiência Operacional

- **Treinamentos e Capacitação:**
  - protótipo desenvolvido e os resultados obtidos indicam uma clara vantagem para a utilização em treinamentos e capacitações de operadores e técnicos, simulando com precisão as operações de controlo do harvester.

- A utilização do CODESYS como plataforma de desenvolvimento demonstrou ser eficaz para configuração do controlador.
- A capacidade de monitorização em tempo real a condição operacional do sistema contribui significativamente para a manutenção preditiva permitindo a identificação precoce de possíveis falhas e a redução de tempos de inatividade.

## 6 - Conclusão e Trabalhos Futuros

### 6.1 Conclusão

#### 6.1.1 Síntese do Projeto

Neste projeto, foi desenvolvido e implementado um protótipo de sistema de monitorização e controle para máquinas móveis, utilizando uma abordagem integrada com o uso de módulos como **CR1202**, **CR2512**, e **CR2014**, todos interligados através de uma rede **CANbus**. A utilização do **Codesys** como ambiente de simulação permitiu a programação do controlador principal.

O desenvolvimento do protótipo demonstrou ser eficaz para o controle de um motor da lâmina simulado, utilizando o **joystick HJ60**. A implementação do filtro para ajustar a variação dos sinais do joystick foi essencial para alcançar um controlo mais estável e confiável do sistema. Esses resultados indicam que a solução desenvolvida contribui significativamente para os conceitos de **Indústria 4.0** aplicados ao setor florestal, promovendo maior eficiência e redução de custos de manutenção.

#### 6.1.2 Principais Contribuições

- **Monitorização em Tempo Real:** O sistema mostrou-se eficiente na monitorização em tempo real dos componentes do protótipo, com feedback preciso e responsivo por meio da interface HMI.
- **Diagnóstico e Manutenção Preditiva:** A integração das funcionalidades de diagnóstico remoto e monitorização contribui para a identificação precoce de falhas e otimiza os processos de manutenção preditiva, reduzindo significativamente o tempo de inatividade das máquinas.

### 6.2 Trabalhos Futuros

#### 6.2.1 Implementação de uma gateway LTE(4G)

Permitirá que o operador aceda a máquina (haverster) em tempo real a partir de um dispositivo móvel, e obtenha diagnóstico das condições da máquina.

### 6.2.2 Expansão do Sistema de Monitorização

Outra sugestão para trabalhos futuros é a expansão do sistema para incluir mais sensores e módulos que possam monitorizar variáveis adicionais da máquina, como temperatura, pressão e vibração. Essa expansão aumentaria a capacidade do sistema de prever falhas e executar manutenção predintiva de forma mais eficaz.

### 6.2.3 Implementação de Inteligência Artificial para Diagnóstico

A aplicação de técnicas de **Inteligência Artificial (IA)** e **Machine Learning** para análise dos dados coletados pelo sistema pode aprimorar significativamente a detecção de anomalias e a previsão de falhas, oferecendo uma abordagem proativa para manutenção e operação de máquinas florestais.

### 6.2.4 Desenvolvimento de um Gêmeo Digital Completo

Investir no desenvolvimento de um gêmeo digital completo do harvester pode oferecer benefícios consideráveis em termos de simulação e otimização do desempenho da máquina em ambientes virtuais, antes mesmo da implementação em campo. Isso ajudaria a prever cenários de falha e a testar diferentes estratégias de operação.

### 6.2.5 Integração com Tecnologias de IoT

A integração do sistema com tecnologias de Internet das Coisas (IoT) pode proporcionar maior conectividade e automação, permitindo monitorização remota em tempo real e coleta de dados contínua, que podem ser analisados para gerar insights sobre o desempenho do harvester.

## 6.3 Considerações Finais

O desenvolvimento deste protótipo representa um avanço significativo na aplicação de tecnologias da Indústria 4.0 para o setor florestal, oferecendo uma solução eficiente para monitorização, controle e manutenção de máquinas móveis. A implementação de tecnologias como a rede CANbus, a interface HMI, e os módulos CR1202, CR2512, e CR2014 reforça a importância de sistemas integrados para aumentar a produtividade e a sustentabilidade das operações.

O sucesso dessa parte do projeto, juntamente com as sugestões para trabalhos futuros, abre caminho para novas pesquisas e inovações que poderão continuar a transformar a forma como as máquinas móveis são operadas e mantidas, contribuindo para um setor florestal mais avançado e eficiente.

## 7. Referências e ANEXOS

### Referências

- [1] Wikipédia. "Indústria 4.0." Consultado em 8 de maio de 2024 Disponível em: [https://en.wikipedia.org/wiki/Industry\\_4.0](https://en.wikipedia.org/wiki/Industry_4.0).
- [2] Springer Link. "Forest 4.0." Consultado em 15 de maio de 2024. Disponível em: <https://link.springer.com/search?query=forest+4.0>.
- [3]"Harvester de Esteira SY185FL." Consultado em 15 de maio de 2024. <https://grupotimber.com.br/produto/harvester-de-esteira-sy185fl/>.
- [4] Bosch. "Protocolo CAN Bus e CANopen." Consultado em 20 de maio de 2024. Disponível em: <https://www.bosch-semiconductors.com/products/can-ip-modules/>.
- [5] CAN in Automation (CiA). "Comunicação CAN - Documentação e Normas." Consultado em 2 de maio de 2024Disponível em: <https://www.can-cia.org/>.
- [6] IFM Electronic. "Módulos CR1202, CR2512 e CR2014." Consultado em 22 de maio de 2024. Disponível em: <https://www.ifm.com/>.
- [7] Codesys. "Plataformas de Programação para Automação." Consultado em 23 de maio de 2024. Disponível em: <https://www.codesys.com/>.
- [8] Wikipédia. "Joystick e Controle PWM." Consultado em 4 de junho de 2024 Disponível em: <https://en.wikipedia.org/wiki/Joystick>.
- [9] IBM. "Gêmeo Digital e Simulação Física." Consultado em 10 de junho de 2024 Disponível em: <https://www.ibm.com/topics/digital-twin>.
- [10] VICORT. *Tecnologia de ponta em máquinas agrícolas e florestais*. Disponível em: <https://www.vicort.com/>. Acesso em: 4 de junho de 2024.
- [11] Siemens. "Simulação e Modelagem para Engenharia." Consultado em 14 de junho de 2024Disponível em: <https://new.siemens.com/global/en/products/software/simulation.html>.
- [12] Cutplant Solutions. "Projeto SMARTCUT e VICORT." Consultado em 19 de junho de 2024Disponível em: <https://www.cutplantsolutions.com/projects/smartcut>.
- [13] *Heavy machine joystick*. Alibaba. Disponível em: <https://portuguese.alibaba.com/product-detail/heavy-machine-joystick-60695259111.html>. Acesso em 16 de junho de 2024

- [14] *Ergonomics design HJ60 hand grip multiaxis*. Alibaba. Disponível em: [https://www.alibaba.com/product-detail/Ergonomics-Design-HJ60-Hand-Grip-Multiaxis\\_1600868593448.html](https://www.alibaba.com/product-detail/Ergonomics-Design-HJ60-Hand-Grip-Multiaxis_1600868593448.html). Acesso em: 1 de julho de 2024
- [15] *Cabo do joystick*. AliExpress. Disponível em: <https://pt.aliexpress.com/i/1005006147975035.html>. Acesso em: 29 de junho de 2024
- [16] *Fonte de alimentação calha DIN 5V DC*. Botnroll. Disponível em: <https://www.botnroll.com/pt/acdc-5v/877-fonte-de-alimentacao-calha-din-5v-dc-24a.html>. Acesso em: 12 de julho de 2024.
- [17] *BTS7960 dupla ponte H para controle de motores DC*. Botnroll. Disponível em: <https://www.botnroll.com/pt/controladores/3589-bts7960-dupla-ponte-h-p-controlo-de-motores-dc-at-43a-27v.html>. Acesso em: 16 de julho de 2024
- [18] *Optokoppler*. Funduino. Disponível em: <https://funduinoshop.com/media/pdf/1c/f2/c8/optokoppler.pdf>. Acesso em: 16 out. 2024.
- [19] *Driver MOSFET*. AliExpress. Disponível em: <https://pt.aliexpress.com/item/33015020793.html>. 15 de julho de 2024
- [20] *Motor redutor controle de velocidade 3,2 KG.cm*. Amazon. Disponível em: <https://www.amazon.com.br/Aexit-redutor-controle-velocidade-3-2KG-cm/dp/B07KTSKD78>. Acesso em: 17 de julho de 2024.
- [21] *Sensor indutivo NA E2EH7052R*. Electgo. Disponível em: <https://electgo.com/om-e2eh7052r>. Acesso em: 20 de julho de 2024
- [22] *Esquema elétrico de sensores indutivos*. Harde. Disponível em: <https://harde.com.br/sensor-indutivo.php>. Acesso em: 21 de julho de 2024
- [23] *Sensor indutivo PRT18-5DO tabela técnica*. Eletruscomp. Disponível em: <https://www.eletruscomp.com.br/produto/sensor-indutivo-prt18-5do-autonics/>. Acesso em: 28 de julho de 2024
- [24] *Sensor indutivo NF*. AliExpress. Disponível em: <https://pt.aliexpress.com/item/1005006463527998.html>. Acesso em: 29 de julho de 2024
- [25] *Sensor de velocidade*. AliExpress. Disponível em: <https://pt.aliexpress.com/i/1005005467713062.html>. Acesso em: 1 de agosto de 2024.
- [26] IFM. *CR2512*. Disponível em: <https://www.ifm.com/ie/en/product/CR2512>. Acesso em: 1 de agosto de 2024.

- [27] IFM. *CR2014*. Disponível em: <https://www.ifm.com/ie/en/product/CR2014>. Acesso em: 4 de agosto de 2024.
- [28] IFM. *CR1202*. Disponível em: <https://www.ifm.com/ie/en/product/CR1202>. Acesso em: 8 de agosto de 2024.
- [29] IFM. *EC2086*. Disponível em: <https://www.ifm.com/ie/en/product/EC2086>. Acesso em: 12 de agosto de 2024.
- [30] IFM. *EC1520*. Disponível em: <https://www.ifm.com/ie/en/product/EC1520>. Acesso em: 18 de agosto de 2024.
- [31] IFM. *EC1533*. Disponível em: <https://www.ifm.com/ie/en/product/EC1533>. Acesso em: 20 de agosto de 2024.
- [32] IFM. *EC0705*. Disponível em: <https://www.ifm.com/ie/en/product/EC0705>. Acesso em: 21 de agosto de 2024.
- [33] Rockwell Automation. *Botão*. Disponível em: <https://literature.rockwellautomation.com/idc/groups/literature/documents/br/800f-br001-en-p.pdf>. Acesso em: 24 de agosto de 2024.
- [34] Embarcados. *Controle PWM e aplicação no Franzininho C0*. Disponível em: <https://embarcados.com.br/entendendo-o-controle-pwm-e-aplicando-no-franzininho-c0/>. Acesso em: 25 de agosto de 2024.
- [35] Webfleet. *Glossário CAN Bus*. Disponível em: <https://www.webfleet.com/pt-pt/webfleet/fleet-management/glossary/can-bus/>. Acesso em: 14 de agosto de 2024.
- [36] Wikipedia. *CAN Bus*. Disponível em: [https://en.wikipedia.org/wiki/CAN\\_bus](https://en.wikipedia.org/wiki/CAN_bus). Acesso em: 16 de agosto de 2024.
- [37] Kalatec. *Codesys*. Disponível em: <https://blog.kalatec.com.br/codesys/>. Acesso em: 28 fev 2024.
- [38] Hager. *Disjuntor 2P 25A NRN225*. Disponível em: <https://hager.com/pt/catalogo/produto/nrn225-disjuntor-2p-25a-c-25ka-2m>. Acesso em: 20 fev 2024.
- [39] Amazon. *Conversor com isolamento*. Disponível em: <https://www.amazon.com.br/Conversor-Isolamento-Conversora-Automa%C3%A7%C3%A3o-Industrial/dp/B0DHLDNBM5>. Acesso em: : 2 Maio 2024.
- [40]LevelOne. *Fonte de alimentação industrial 120W 48VDC DIN Rail PoE Ready POW-4841*. Kuantokusta. Disponível em: <https://www.kuantokusta.pt/p/194173/level->

[one-fonte-de-alimentacao-industrial-120w-48vdc-din-rail-poe-ready-pow-4841.](#)

Acesso em: 5 fev. 2024.

## ANEXOS

Tabela de conexões físicas do módulo CR2512 LR (lado direito)

<b>Lado Direito</b>	<b>Canais</b>	<b>Pinos</b>	<b>Ligações</b>
<b>LR1</b>	Canal-9	43	24V - Sensor indutivo (NA)
<b>LR2</b>	Canal-9	47	IN - Sensor indutivo (NA)
<b>LR3</b>	Canal-9	06	GND - Sensor indutivo (NA)
<b>LR4</b>	Canal-10	44	24V- Sensor indutivo (NF)
<b>LR5</b>	Canal-10	48	IN - Sensor indutivo (NF)
<b>LR6</b>	Canal-10	07	GND - Sensor indutivo (NF)
<b>LR7</b>	Canal-11	45	
<b>LR8</b>	Canal-11	49	IN - Sensor indutivo (NF)
<b>LR9</b>	Canal-11	24	
<b>LR10</b>	Canal-12	46	
<b>LR11</b>	Canal-12	31	
<b>LR12</b>	Canal-12	26	
<b>LR13</b>	Canal-1	08	24V – Mosfest de 4 canais “+DC”
<b>LR14</b>	Canal-1	20	OUT-PWM – Optoacoplador “+1”
<b>LR15</b>	Canal-1	38	
<b>LR16</b>	Canal-1	02	OUT-PWM – Optoacoplador “-1”
<b>LR17</b>	Canal-2	27	
<b>LR18</b>	Canal-2	21	
<b>LR19</b>	Canal-2	39	
<b>LR20</b>	Canal-2	03	
<b>LR21</b>	Canal-3	09	
<b>LR22</b>	Canal-3	22	
<b>LR23</b>	Canal-3	40	
<b>LR24</b>	Canal-3	04	
<b>LR25</b>	Canal-4	28	

<b>LR26</b>	Canal-4	42	
<b>LR27</b>	Canal-4	41	
<b>LR28</b>	Canal-4	25	
<b>LR29</b>	Canal-5	10	
<b>LR30</b>	Canal-5	34	
<b>LR31</b>	Canal-5	52	
<b>LR32</b>	Canal-5	16	
<b>LR33</b>	Canal-5	39	
<b>LR34</b>	Canal-6	29	
<b>LR35</b>	Canal-6	35	
<b>LR36</b>	Canal-6	53	
<b>LR37</b>	Canal-6	17	
<b>LR38</b>	Canal-7	11	
<b>LR39</b>	Canal-7	36	
<b>LR40</b>	Canal-7	54	
<b>LR41</b>	Canal-7	18	
<b>LR42</b>	Canal-8	30	
<b>LR43</b>	Canal-8	37	
<b>LR44</b>	Canal-8	55	
<b>LR45</b>	Canal-8	19	
<b>LR46</b>			
<b>LR47</b>			
<b>LR48</b>			
<b>LR49</b>			
<b>LR50</b>			

Tabela de conexões físicas do módulo CR2512 LF (lado Esquerdo)

<b>Lado Direito</b>	<b>Canais</b>	<b>Pinos</b>	<b>Ligações</b>
<b>LF1</b>	Canal-9	43	24V - Sensor Hall
<b>LF2</b>	Canal-9	47	IN - Sensor Hall
<b>LF3</b>	Canal-9	06	GND - Sensor Hall
<b>LF4</b>	Canal-10	44	
<b>LF5</b>	Canal-10	48	IN - Sensor Hall
<b>LF6</b>	Canal-10	07	
<b>LF7</b>	Canal-11	45	
<b>LF8</b>	Canal-11	49	
<b>LF9</b>	Canal-11	24	
<b>LF10</b>	Canal-12	46	
<b>LF11</b>	Canal-12	31	
<b>LF12</b>	Canal-12	26	
<b>LF13</b>	Canal-1	08	
<b>LF14</b>	Canal-1	20	
<b>LF15</b>	Canal-1	38	
<b>LF16</b>	Canal-1	02	
<b>LF17</b>	Canal-2	27	
<b>LF18</b>	Canal-2	21	
<b>LF19</b>	Canal-2	39	
<b>LF20</b>	Canal-2	03	
<b>LF21</b>	Canal-3	09	
<b>LF22</b>	Canal-3	22	
<b>LF23</b>	Canal-3	40	
<b>LF24</b>	Canal-3	04	
<b>LF25</b>	Canal-4	28	
<b>LF26</b>	Canal-4	42	
<b>LF27</b>	Canal-4	41	
<b>LF28</b>	Canal-4	25	
<b>LF29</b>	Canal-5	10	

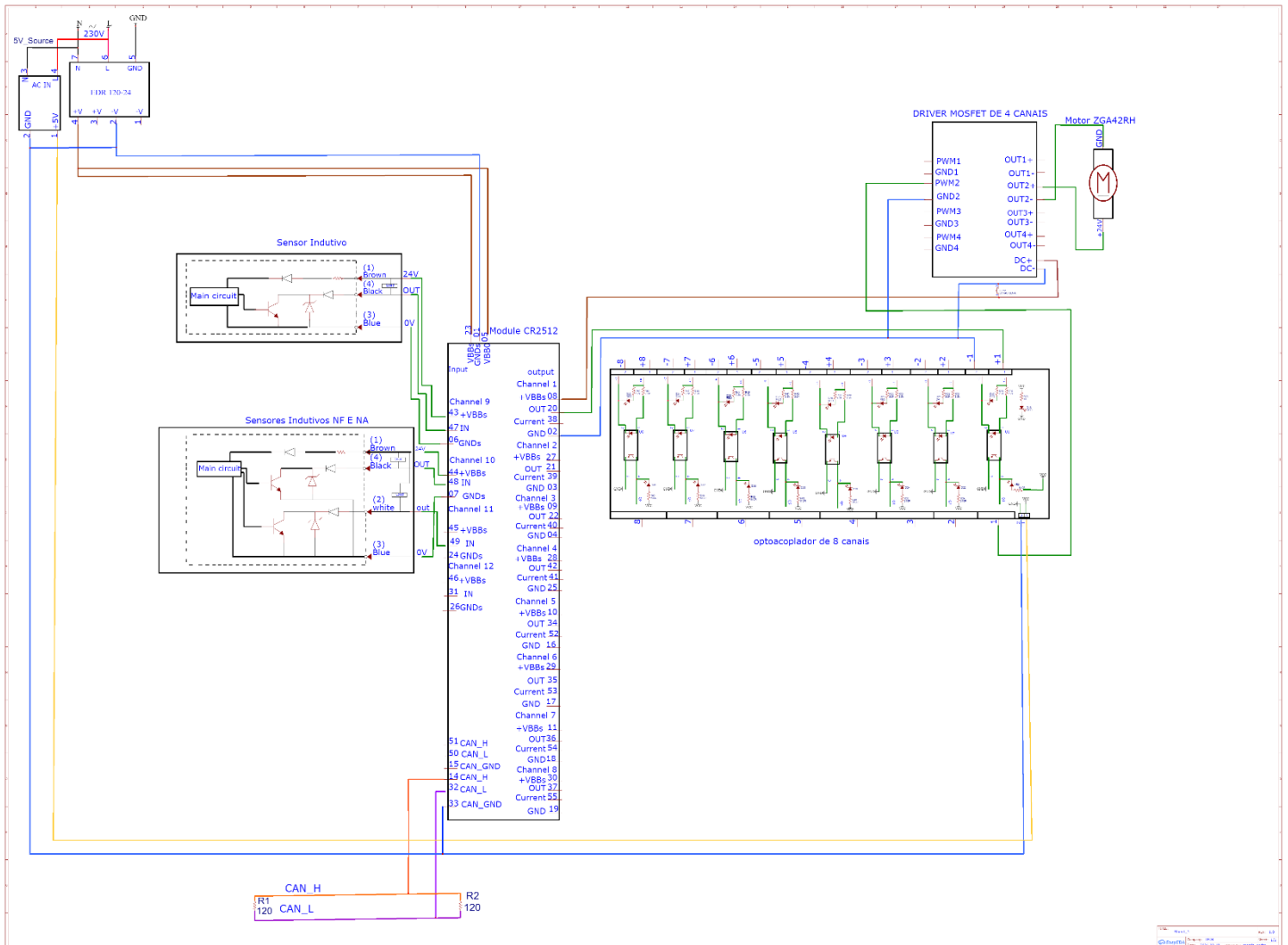
<b>LF30</b>	Canal-5	34	
<b>LF31</b>	Canal-5	52	
<b>LF32</b>	Canal-5	16	
<b>LF33</b>	Canal-5	39	
<b>LF34</b>	Canal-6	29	
<b>LF35</b>	Canal-6	35	
<b>LF36</b>	Canal-6	53	
<b>LF37</b>	Canal-6	17	
<b>LF38</b>	Canal-7	11	
<b>LF39</b>	Canal-7	36	
<b>LF40</b>	Canal-7	54	
<b>LF41</b>	Canal-7	18	
<b>LF42</b>	Canal-8	30	
<b>LF43</b>	Canal-8	37	
<b>LF44</b>	Canal-8	55	
<b>LF45</b>	Canal-8	19	
<b>LF46</b>			
<b>LF47</b>			
<b>LF48</b>			
<b>LF49</b>			
<b>LF50</b>			

Tabela de conexões físicas do módulo CR2014 Cabine (C)

<b>C</b>	<b>Plug</b>	<b>Pino</b>	<b>Canal</b>	<b>Tipo</b>	<b>Ligação</b>
<b>C1</b>	X1	1	Canal 1	Bin IN 1	Joystck1-Pino3-botão 1
<b>C2</b>	X1	2	+UB		COMUM Joystck1-Pino11-24V
<b>C3</b>	X1	3	Canal 2	Bin IN 2	Joystck1-Pino4-botão 2
<b>C4</b>	X1	4	+UB		COMUM Joystck2-Pino11-24V
<b>C5</b>	X1	5	Canal 3	Bin IN 3	Joystck1-Pino2-botão 3
<b>C6</b>	X1	6	+UB		
<b>C7</b>	X1	7	Canal 4	Bin IN 4	Joystck1-Pino5-botão 4
<b>C8</b>	X1	8	+UB		
<b>C9</b>	X1	9	+UB		
<b>C10</b>	X1	10	+UB		
<b>C11</b>	X1	11	Canal 5	Ana IN 5	Joystc1-Pino9-SCROLL
<b>C12</b>	X1	12	Canal 6	Ana IN 6	Joystck2-Pino8- ESCALA Y
<b>C13</b>	X1	14	+UB		
<b>C14</b>	X1	15	Canal 7	Bin IN 7	Joystck2-Pino3-botão 1
<b>C15</b>	X1	16	+UB		
<b>C16</b>	X1	17	Canal 8	Bin IN 8	Joystck2-Pino4-botão 2
<b>C17</b>	X1	15	Canal 7	Bin OUT 7	
<b>C18</b>	X1	17	Canal 8	Bin OUT 8	
<b>C19</b>	X2	1	Canal 9	Bin IN 9	
<b>C20</b>	X2	2	+UB		
<b>C21</b>	X2	3	Canal 10	Bin IN 10	Joystck2-Pino2-botão 3
<b>C22</b>	X2	4	+UB		
<b>C23</b>	X2	5	Canal 11	Bin IN 11	Joystck2-Pino5-botão 4
<b>C24</b>	X2	6	+UB		
<b>C25</b>	X2	7	Canal 12	Bin IN 12	
<b>C26</b>	X2	8	+UB		
<b>C27</b>	X2	9	+UB		
<b>C28</b>	X2	10	+UB		

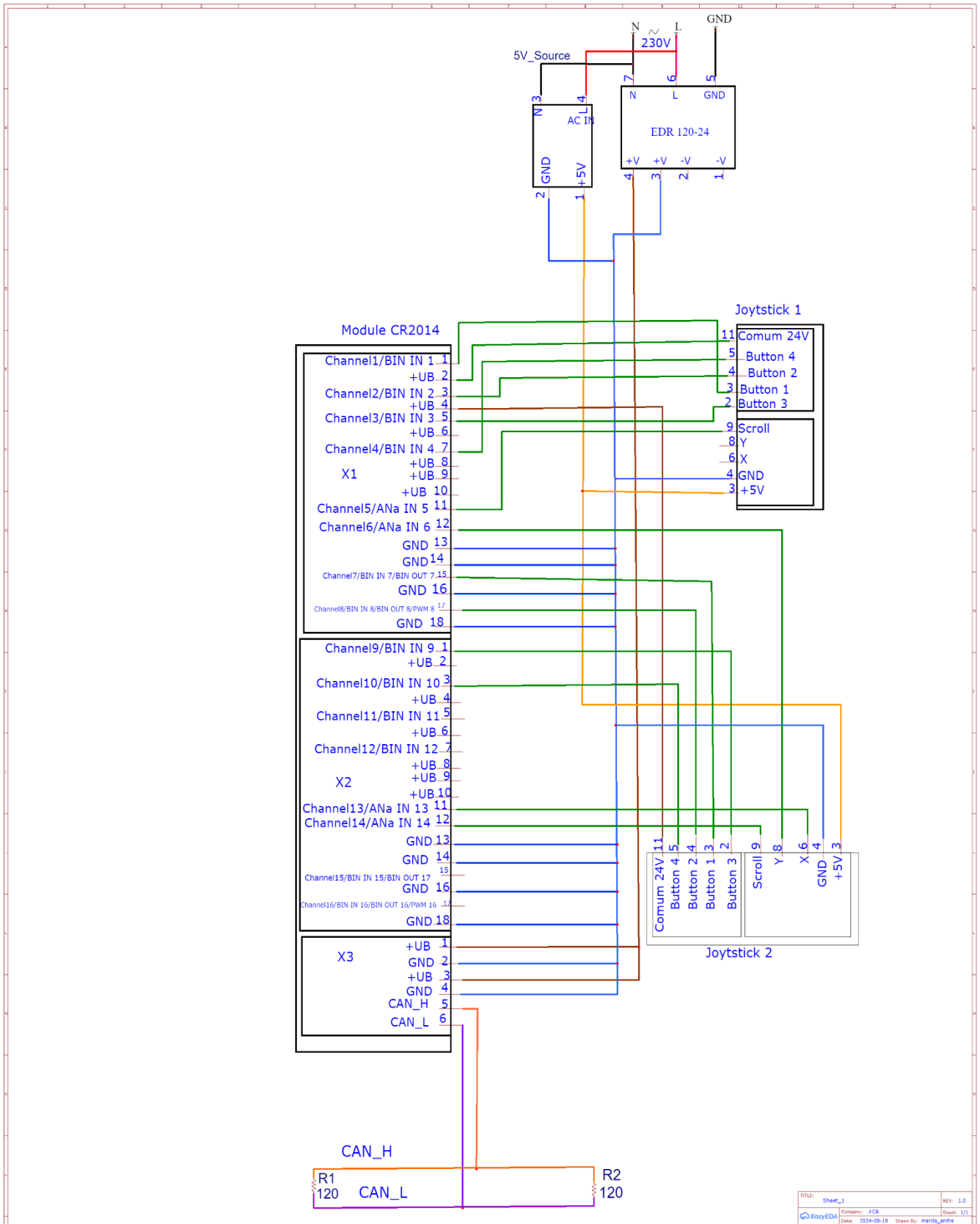
<b>C29</b>	X2	11	Canal 13	Ana IN 13	Joystck2-Pino6- ESCALA X
<b>C30</b>	X2	12	Canal 14	Ana IN 14	Joystck2-Pino9- SCROLL
<b>C31</b>	X2	14	+UB		
<b>C32</b>	X2	15	Canal 15	Bin IN 15	
<b>C33</b>	X2	16	+UB		
<b>C34</b>	X2	17	Canal 16	Bin IN 16	
<b>C35</b>	X2	15	Canal 15	Bin OUT 15	
<b>C36</b>	X2	17	Canal 16	Bin OUT 16	
<b>C37</b>					
<b>C38</b>					
<b>C39</b>					
<b>C40</b>					

# 1- Esquema elétrico do CR2512

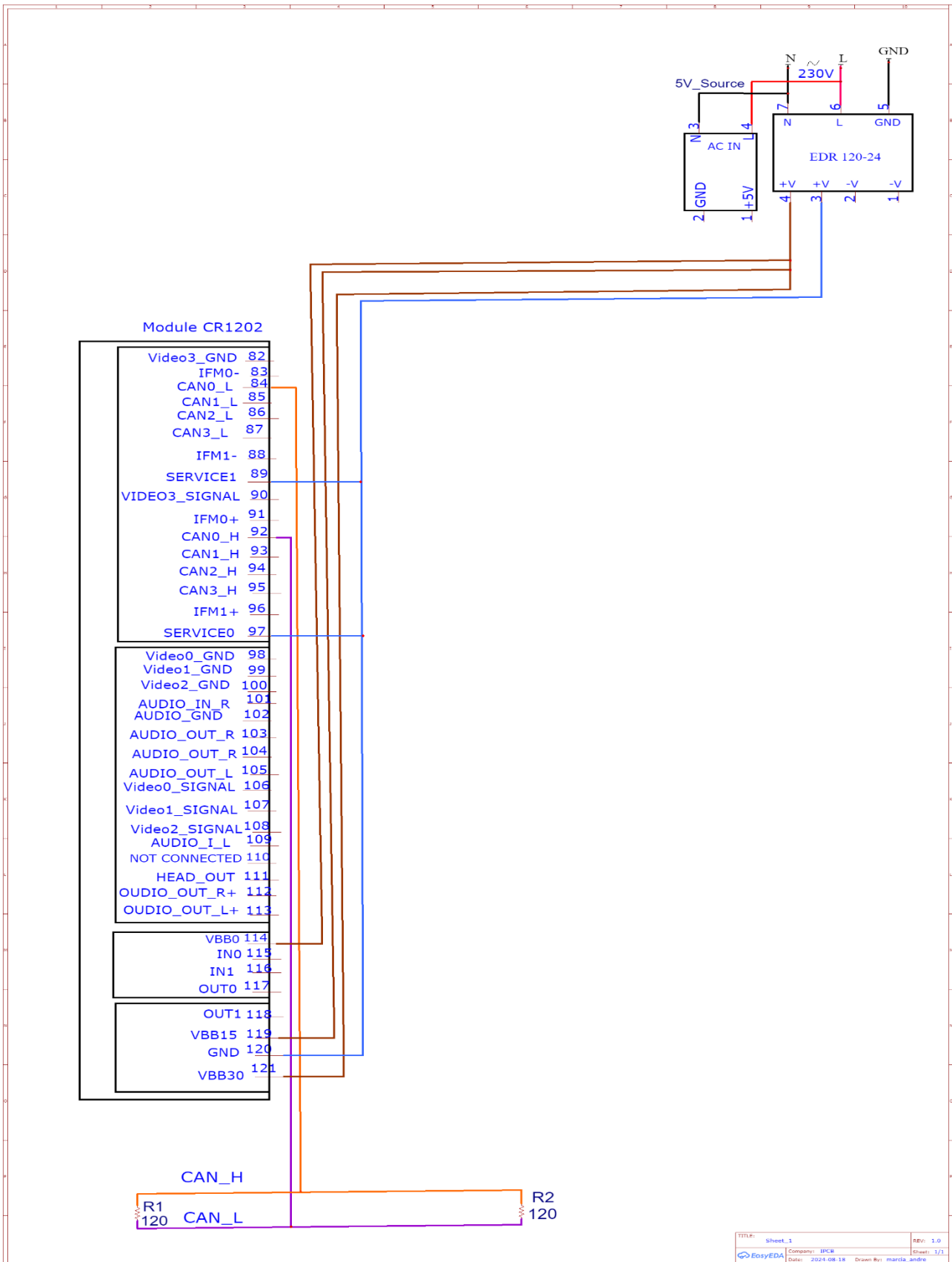




### 3- Esquema elétrico do CR2014



#### 4- Esquema elétrico do CR1202





### 5- Esquema elétrico geral

