

**UNIVERSIDADE DO VALE DO RIO DOS SINOS - UNISINOS
UNIDADE ACADÊMICA DE GRADUAÇÃO
CURSO DE ENGENHARIA DE CONTROLE E AUTOMAÇÃO**

GUSTAVO HAUSSMANN NÖR

**INTERNET DAS COISAS INDUSTRIAL:
DISPOSITIVO PARA COMUNICAÇÃO MODBUS ATRAVÉS DE REDES CELULAR**

SÃO LEOPOLDO

2021

GUSTAVO HAUSSMANN NÖR

INTERNET DAS COISAS INDUSTRIAL:
DISPOSITIVO PARA COMUNICAÇÃO MODBUS ATRAVÉS DE REDES CELULAR

Trabalho apresentado como requisito parcial para obtenção do título de Bacharel em Engenharia de Controle e Automação, pela Universidade do Vale do Rio dos Sinos – UNISINOS.

Orientador: Prof. Dr. Rodrigo Marques de Figueiredo

São Leopoldo

2021

RESUMO

O trabalho proposto tem como objetivo estudar a Internet das Coisas Industrial, da sigla IloT, aplicando-a na área de comunicação industrial. O objeto proposto consiste no desenvolvimento de um *gateway* capaz de se comunicar com dispositivos de campo através do protocolo de comunicação industrial Modbus RTU, enviando os dados lidos e escrevendo dados nos registradores dos dispositivos através de redes móveis (2G, 3G, 4G) utilizando o protocolo MQTT (*Message Queuing Telemetry Transport*). No trabalho, é apresentada uma revisão bibliográfica, abordando os principais tópicos envolvidos no trabalho, tais como a Internet das Coisas no ramo industrial, meios físicos de comunicação sem fio e cabeados, principais protocolos de redes de campo (*fieldbus*) e o protocolo MQTT. Além disso, é apresentada uma pesquisa de produtos relacionados ao objeto proposto já disponíveis no mercado, e uma pesquisa de trabalhos científicos correlatos ao assunto do sistema proposto. Para a implementação do protótipo, foram utilizados um modem Cat-M1, um kit de desenvolvimento, responsável pelo gerenciamento e comunicação com o modem e com os dispositivos via Modbus, além de um controlador com suporte a comunicação Modbus RTU para a realização de testes em um cenário real. Para a realização dos testes, foi implementada uma máquina virtual hospedada em um serviço de nuvem rodando um *broker* MQTT, e uma interface de envio de comandos e leitura de mensagens via MQTT. Para a comprovação da eficiência do sistema, foi realizada uma bateria de testes funcionais, onde a taxa de comunicação sem perda de mensagens se manteve acima de 90%. Os resultados obtidos nestes testes comprovaram a eficiência do sistema proposto.

Palavras-chaves: Internet das Coisas Industriais. Modbus. MQTT.

LISTA DE FIGURAS

Figura 1 - Faixas de tensão.....	14
Figura 2 - Configuração do cabo para conectores DB-9 e DB-25.....	15
Figura 3 - Frame Modbus.....	16
Figura 4 - Arquitetura básica de um modelo MQTT broker / client.....	19
Figura 5 - Níveis de QoS.....	20
Figura 6 - dongle desenvolvido pela empresa CTN Tecnologia CNC®.....	21
Figura 7 - Gateway desenvolvido pela empresa Atop Technologies®.....	22
Figura 8 - Roteador VPN industrial para rede celular desenvolvido pela Novus®.....	22
Figura 9 - dongle desenvolvido pela empresa WireFreeCNC®.....	23
Figura 10 - dongle desenvolvido pela empresa MachineMetrics®.....	24
Figura 11 – Diagrama de blocos das etapas do desenvolvimento do protótipo.....	30
Figura 12 - Controlador Universal N1200®.....	31
Figura 13 - Arduino UNO®.....	31
Figura 14 - Módulo adaptador RS485.....	32
Figura 15 - Modem celular CINTERION EXS82-W.....	32
Figura 16 - Infográfico do sistema proposto.....	33
Figura 17 - Fluxograma do algoritmo desenvolvido.....	35
Figura 18 – Protótipo desenvolvido.....	36
Figura 19 - Interface de testes implementada no Node-RED®.....	37
Figura 20 - Fluxograma de testes para validação do sistema.....	38
Figura 21 - Fluxograma de testes do sistema completo.....	39
Figura 22 - Representação dos comandos Incrementa e Ler valores implementados na interface de testes.....	41
Figura 23 - Exemplos de aplicações possíveis.....	43

LISTA DE QUADROS

Quadro 1 - Funções públicas.	17
Quadro 2 - Resumo dos produtos similares disponíveis no mercado.	24
Quadro 3 - Resumo dos trabalhos correlatos.....	28
Quadro 4 - Resultado dos testes funcionais realizados.	40

LISTA DE TABELAS

Tabela 1 - Faixas das categorias de funções.....	17
--	----

LISTA DE ABREVIATURAS E SIGLAS

ASCII	<i>American Standard Code for Information Interchange</i> (Código Padrão Norte-americano para Intercâmbio de Informações)
CAD	<i>Computer Aided Design</i> (Desenho assistido por computador)
CAM	<i>Computer Aided Manufacturing</i> (Manufatura assistida por computador)
CI	<i>Integrated circuit</i> (Circuito integrado)
Gbps	<i>Gigabits per second</i> (Gigabits por segundo)
HTML	<i>HyperText Markup Language</i> (Linguagem de Marcação de Hipertexto)
IIoT	<i>Industrial Internet of Things</i> (Internet das Coisas Industrial)
IoT	<i>Internet of Things</i> (Internet das Coisas)
LR-WPAN	<i>Low-rate wireless personal area networks</i> (Redes de comunicação sem fio com baixa taxa de transmissão de dados)
Mb/s	<i>Megabytes per second</i> (Megabytes por segundo)
MQTT	<i>Message Queuing Telemetry Transport</i> (Enfileiramento de mensagens de transporte de telemetria)
PDU	<i>Protocol Data Unit</i> (Protocolo de unidade de dados)
PHY	<i>Physical Layer</i> (Camada física)
QoS	<i>Quality of Service</i> (Qualidade de Serviço)
RTU	<i>Remote Terminal Unit</i> (Unidade terminal remota)
TLS	<i>Transport Layer Security</i> (Segurança da Camada de Transporte)
TSCH	<i>Time Slotted Channel Hopping</i> (Salto de Canais por Intervalo de Tempo)
USB	<i>Universal Serial Bus</i> (Porta serial universal)
V	Volt
VM	<i>Virtual Machine</i> (máquina virtual)

SUMÁRIO

1. INTRODUÇÃO.....	9
2. REFERENCIAL TEÓRICO.....	11
2.1 Conceitos Básicos.....	11
2.1.1 Internet das Coisas Industrial (IIoT - <i>Industrial Internet of Things</i>).....	11
2.1.2 Meios físicos de comunicação sem fio.....	12
2.1.3 Meios físicos de comunicação cabeados.....	14
2.1.4 Protocolo de comunicação Industrial – Modbus RTU	15
2.1.5 MQTT - <i>Message Queue Telemetry Transport</i>	18
2.2 Pesquisa de Mercado	20
2.2.1 I-MD DUO ETHERNET / USB / RS-232®	21
2.2.2 MB5901B Series®	21
2.2.3 AIRGATE 4G®.....	22
2.2.4 WF3 for Universal Connectivity®	23
2.2.5 Industrial IoT Platform®	23
2.3 Trabalhos Correlatos.....	25
3. METODOLOGIA	30
3.1 Etapas do desenvolvimento do protótipo	30
3.2 Materiais utilizados.....	30
3.3 Sistema proposto	33
3.4 Testes e validação	36
4. ANÁLISE DOS RESULTADOS.....	40
5. CONCLUSÃO	44
REFERÊNCIAS.....	45

1. INTRODUÇÃO

A evolução das tecnologias vem se intensificando cada vez mais ao longo das últimas décadas em diferentes áreas, incluindo os setores industriais. Através dessa evolução e da necessidade de se ter um maior controle sobre os processos industriais, surgiu a chamada Internet das Coisas Industrial (IIoT - *Industrial Internet of Things*). Essa tecnologia utiliza dispositivos e recursos tecnológicos para a criação de processos inteligentes, que através da obtenção de dados dos processos produtivos, permitem tomadas de decisões mais assertivas, otimizando os recursos e melhorando assim a produtividade (Plant Engineering, 2015).

Os ambientes industriais normalmente contam com diferentes tipos de máquinas e equipamentos. O alto custo envolvido nas aquisições desses equipamentos, não permitem que as empresas simplesmente se desfaçam de dispositivos e máquinas antigas, e adquiram equipamentos novos para se adequarem a utilização da Internet das Coisas Industrial (Robotics Tomorrow, 2021).

Por esse motivo, este trabalho propõe e tem como principal objetivo o desenvolvimento de um protótipo, que permita a adaptação de equipamentos com interface de comunicação serial, originalmente sem uma interface de comunicação IIoT, a se comunicarem e trabalharem com essa nova tecnologia. Como objetivos específicos, a comunicação do protótipo com o serviço *cloud* será realizada através de rede de celular, e será bidirecional, permitindo tanto o envio de dados dos equipamentos para a nuvem, assim como receber comandos a partir de um computador ou dispositivo móvel utilizando a internet.

O dispositivo IIoT proposto, será desenvolvido utilizando um kit de desenvolvimento, e um modem Cat-M1, que conta com um sistema de comunicação através de rede celular. A comunicação entre o kit de desenvolvimento e a máquina será feita através da interface de comunicação serial disponível no controlador. Os dados coletados, serão então enviados via rede celular (4G com *fallback* até 2G) para um servidor armazenado em um sistema de armazenamento em nuvem. O acesso a esses dados será feito através de um computador pelo navegador de internet.

O presente trabalho está dividido em 6 capítulos, iniciando pela introdução, que contextualiza e situa o trabalho, apresentando os principais objetivos que se deseja alcançar, assim como a justificativa para o desenvolvimento desse trabalho. Após

isso, é apresentado o referencial teórico, que está dividido em três partes, versando sobre os principais conceitos básicos necessários para o entendimento do trabalho, uma pesquisa sobre trabalhos acadêmicos que se assemelham ao assunto abordado nesse trabalho, e uma pesquisa sobre produtos similares já existentes no mercado. No terceiro capítulo, na seção de metodologia, são apresentados os infográficos exemplificando o sistema proposto, assim como as etapas do desenvolvimento e testes para validação do trabalho. No capítulo de análise de resultados, são apresentados mais detalhes e características do sistema desenvolvido e os resultados obtidos com os testes. No quinto capítulo, são apresentadas e discutidas as conclusões obtidas com o desenvolvimento do trabalho. Finalmente, no último tópico, são apresentadas as referências bibliográficas utilizadas na elaboração do trabalho.

2. REFERENCIAL TEÓRICO

Neste capítulo são explicados os principais conceitos envolvidos no trabalho. Além disso, também é apresentada uma pesquisa de produtos similares disponíveis no mercado, e uma pesquisa sobre trabalhos científicos desenvolvidos na área da Internet das Coisas Industrial (IIoT - *Industrial Internet of Things*), focados na área de meios de comunicação sem fio. Iniciando pela seção de Conceitos Básicos que versam sobre os principais tópicos envolvidos no trabalho, como o IIoT e seus desafios, meios físicos de comunicação sem fio e cabeados, principais protocolos de redes de campo e o protocolo MQTT (*Message Queue Telemetry Transport*).

2.1 Conceitos Básicos

Nesta seção, são contextualizados os principais assuntos envolvidos no trabalho. Iniciando por uma explicação sobre o que é a Internet das Coisas Industrial, alguns exemplos de aplicação e os desafios para sua utilização em larga escala. Na sequência, é explicado sobre as principais tecnologias de comunicação sem fio utilizadas nos dispositivos de Internet das Coisas. Após isso, são apresentados os meios de comunicação cabeados normalmente utilizados na indústria e uma explicação sobre o protocolo MQTT.

2.1.1 Internet das Coisas Industrial (IIoT - *Industrial Internet of Things*)

A Internet das Coisas (IoT – *Internet of Things*), foi um termo criado para transmitir a ideia de que a internet pode estar presente em todas as coisas e equipamentos em nossa volta. A ideia desse conceito consiste em conectar todos os equipamentos à internet, e com isso facilitar o dia a dia das pessoas. A IoT tem diferentes funções, como serviços de localização, rastreamento, reconhecimento inteligente, tudo isso através da troca de informações via internet. A IoT é apontada como um importante tópico em instituições acadêmicas e na indústria, com um grande potencial de melhorias em diferentes áreas, tanto profissionais, como econômicas e sociais (Morais, et al., 2018).

A Internet das Coisas já está presente em nossas vidas a muitos anos. Quando utilizamos o *smartphone* para fazer uma transação bancária, ou quando recebemos um alerta no celular falando que o sistema de segurança das nossas casas foi acionado, isso tudo está relacionada a IoT. Devido aos avanços nos serviços de

comunicação sem fio e serviços de armazenamento em nuvem, o conceito está em alta e sendo aplicado em diferentes produtos (Plant Engineering, 2015).

O termo Internet das Coisas Industrial (IIoT), do inglês *Industrial Internet of Things*, segue o mesmo conceito da IoT. A diferença é que a IIoT é aplicada a comunicação entre máquinas em ambiente industrial. Os sistemas IIoT permitem obter informações sobre o estado das máquinas, problemas em cadeias de suprimentos, entre outros dados. Um exemplo de aplicação desse conceito é um trabalhador que trabalha na manutenção de equipamentos em uma empresa, e ele precisa saber quando a temperatura de um motor está fora dos limites. A IIoT é utilizada para detectar a temperatura e enviar um alerta para o celular do trabalhador, que irá fazer as intervenções necessárias (Plant Engineering, 2015).

Um dos principais desafios enfrentados na implementação dessa tecnologia está relacionado a segurança. Conectando todas as máquinas e equipamentos a sistemas de armazenamento e processamento em nuvem, se abre a possibilidade de ataques por *hackers* ao sistema. Para evitar esses problemas, as empresas precisarão ter times atentos a essas questões de segurança (Machine Design, 2016).

Outro ponto apontado como desafio na IoT e IIoT é em relação ao meio de comunicação utilizado. Redes como Wi-Fi e Bluetooth possuem alcance limitados, o que em muitas empresas seria um problema, pois muitos pontos de acesso seriam necessários. Outro problema nessas redes é em relação a escalabilidade, muitas conexões simultâneas tendem a causar um alto tráfego de dados na rede e lentidão. Uma outra opção são as redes móveis (2G, 3G e 4G), o problema é que elas foram desenvolvidas com foco em aplicações de texto, voz, imagem e vídeo. A próxima geração das redes móveis (5G), é apontada como uma grande solução, pois essa tecnologia possui um baixo consumo de energia, alta velocidade de conexão e baixa latência (Morais, et al., 2018).

2.1.2 Meios físicos de comunicação sem fio

Atualmente, existem no mercado diferentes meios e tecnologias de comunicação sem fio, tais como as redes móveis (2G, 3G, 4G e 4.5G), Wi-Fi, Bluetooth, ZigBee, entre outros. Os dispositivos e sistemas voltados para a Internet das Coisas (IoT – *Internet of Things*), são pensados e desenvolvidos para utilizar essas tecnologias, e muitas vezes, por conta das tecnologias de comunicação

disponíveis, tem seu potencial limitado. Uma das tecnologias mais utilizada por dispositivos IoT é o Wi-Fi. O Wi-Fi tem o seu funcionamento definido pelo padrão IEEE 802.11. Esse padrão possui algumas variações, como 802.11b, 802.11g, e 802.11n, que foram sendo desenvolvidos e implementados ao longo dos anos. A transmissão das redes Wi-Fi é feita através de sinais de radiofrequência. Este padrão, utiliza faixas de radiofrequências livres de licenças para utilização. A utilização dessa tecnologia em dispositivos IoT, e principalmente em dispositivos e sistemas IIoT, tem quatro grandes desvantagens, a primeira é em relação a abrangência do sinal, para cobrir a área de uma grande empresa seria necessário a instalação de vários pontos de acesso. A segunda desvantagem é a escalabilidade, muitos dispositivos conectados simultaneamente a uma rede Wi-Fi tendem a causar lentidão no tráfego de dados na rede. A terceira desvantagem é que o sinal das redes Wi-Fi é afetado pelos ruídos gerados pelas máquinas nas indústrias, e a quarta desvantagem é que essa tecnologia foi projetada para maximizar a vazão de dados sem levar em consideração a minimização do consumo de energia (Comer, 2016).

Outra opção que vem sendo utilizada são as redes móveis, como 3G, 4G (LTE) e 4.5G (LTE Advanced). O problema dessas tecnologias, é que elas foram desenvolvidas com foco na aplicação de texto, imagem, voz e vídeo, não atendendo requisitos muitas vezes fundamentais em dispositivos IIoT, como alta velocidade de conexão e baixa latência (Comer, 2016).

A terceira opção, que é considerada como um grande fator para o avanço da IIoT e indústria 4.0, é a tecnologia 5G. Essa tecnologia é a primeira rede móvel desenvolvida pensando nos casos de uso industrial. As especificações das redes 5G irão permitir a automação de sistemas e o monitoramento de sistemas em tempo real (Vitturi, et al., 2019). De acordo com as especificações técnicas a tecnologia 5G terá velocidade de até 10 Gbps, latência baixa, baixo consumo de energia e alta taxa de disponibilidade (Neto, 2018).

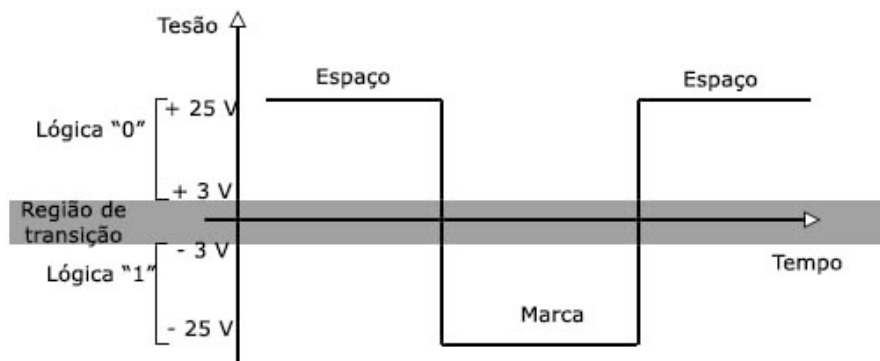
No próximo tópico é explicado sobre as tecnologias envolvidas nos meios físicos de comunicação, dando ênfase a conexão serial RS485, que foi utilizada no desenvolvimento do *gateway*.

2.1.3 Meios físicos de comunicação cabeados

A conexão entre o controlador e os dispositivos externos, é realizada através da interface que o controlador oferece. Geralmente os controladores, que podem ser um CLP, um CNC ou qualquer dispositivo industrial que possua uma interface de comunicação externa, contam com portas serial que utilizam os padrões de comunicação do tipo RS232 ou RS485 (Filho, 2014).

O protocolo RS232 usa transmissão serial. Nesse tipo de transmissão, os bits são enviados um a um por um cabo que interliga o receptor, nesse caso o controlador, e o transmissor, nesse caso o *gateway* IIoT. A Figura 1 apresenta as faixas de tensão do protocolo RS232. Como apresentado na figura, os bits, 0 ou 1, são reconhecidos pelo receptor dependendo da faixa de tensão lida, tensões na faixa de +3 V a +15 V são reconhecidas como níveis lógicos 0, enquanto valores entre -3 V e -15 V são reconhecidos como níveis lógicos 1. O padrão RS232 possui um limite máximo de tensão de 25 V, a diferença de 15 V para 25 V é utilizada como margem de segurança (Zanco, 2008).

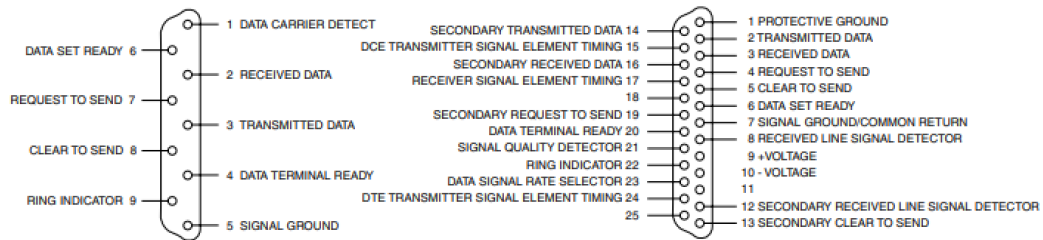
Figura 1 - Faixas de tensão.



Fonte - Instituto Newton C. Braga.

Neste protocolo, o receptor e o transmissor devem ser configurados com as mesmas velocidades. A conexão pode ser *full-duplex*, permitindo uma comunicação bilateral. O padrão também conta com as especificações mecânicas do conector, que podem ser observadas na Figura 2, sendo um conector de 25 pinos conhecido como DB-25. No entanto, o conector de 9 pinos conhecido como DB-9, definido na especificação EIA-574, é o conector mais frequentemente utilizado (Zanco, 2008).

Figura 2 - Configuração do cabo para conectores DB-9 e DB-25.



Fonte - *The RS232 Standard*.

No padrão de comunicação serial RS485, que é muito utilizado na automação industrial, é utilizado um único par de fios. Esse padrão suporta conexões com até 1200 metros de extensão, com uma taxa de transmissão de até 10 Mb/s, e é possível conectar até 32 dispositivos na mesma rede em uma configuração mestre/escravo. Esse padrão tem como característica uma conexão *half-duplex*, em que cada dispositivo aciona o transmissor apenas no momento em que for transmitir ou receber informações, permanecendo desativado no restante do tempo (Filho, 2014).

2.1.4 Protocolo de comunicação Industrial – Modbus RTU

As redes de comunicação industriais utilizadas para a automação industrial já estão presentes nas industriais a muitos anos. A rápida evolução da engenharia de software e da eletrônica, foram importantes fatores para o desenvolvimento dos sistemas de automação distribuídos utilizando as redes de campo, que são muito conhecidas devido ao termo em inglês *Fieldbus* (Moraes, 2019).

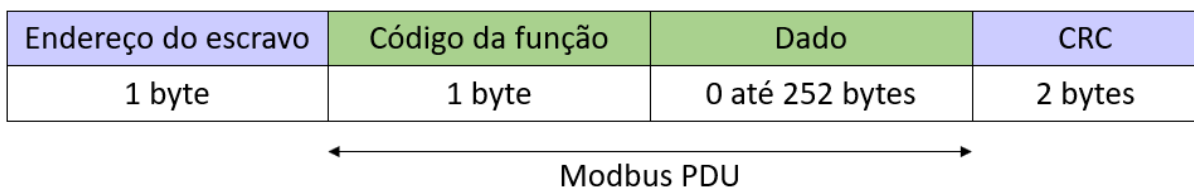
Inicialmente, diversos fabricantes criavam suas próprias soluções de rede de campo, o que acabava fazendo com que os clientes ficassem dependentes dos produtos e dos serviços de manutenção de um único fornecedor. Esse fator foi uma das grandes motivações para os projetos e desenvolvimento dos protocolos de comunicação industrial padronizados. Entre esses protocolos padronizados, podemos destacar o protocolo Profibus, padronizado pela Associação Profibus Internacional, e as redes Modbus, que é um dos protocolos mais utilizados devido ao fato de ser de simples implementação e de baixo custo (Moraes, 2019).

O protocolo de comunicação industrial Modbus, é definido por uma estrutura de envio de mensagens, que através do envio de informações como endereço do dispositivo, código da função e dados, permite a comunicação entre o dispositivo mestre da rede e seus escravos. Esse protocolo suporta a comunicação de

dispositivos conectados a diferentes tipos de redes e barramentos, como comunicação serial através das interfaces físicas RS232 e RS485, e redes do tipo TCP/IP. O Modbus oferece dois modos de transmissão serial, o modo RTU (*Remote terminal unit*) e o ASCII (*American Standard Code For Information Interchange*) (Modbus Organization, Inc, 2006).

No modo de transmissão RTU, cada mensagem contém 8 bits de dados, um *start* bit, um bit de paridade e um *stop* bit. Um ponto importante do Modbus é entender o *frame* que descreve o protocolo, que é apresentado na Figura 3. O *frame* é composto pelo PDU (*Protocol Data Unit*) que engloba a parte dos dados, e pela parte que descreve o endereçamento do dispositivo escravo. O tamanho máximo do *frame* RTU é de 256 bytes (Modbus Organization, Inc, 2006).

Figura 3 - *Frame* Modbus.



Fonte - Adaptado de Modbus Organization (2006).

Um detalhe importante, é que a mensagem enviada pelo *frame* deve ser transmitida de forma contínua. Caso ocorra um atraso de mais de 1,5 vezes o tempo de transmissão de um caractere, a mensagem enviada através do *frame* deve ser considerada incompleta e descartada pelo receptor. Outro detalhe, é que no modo RTU o envio de um *frame* após o outro deve respeitar o tempo sem transmissão que é definido como sendo 3,5 vezes o tempo de transmissão de um caractere (Modbus Organization, Inc, 2006).

No Modbus RTU, o envio de mensagens e requisições sempre são enviadas pelo dispositivo mestre, também chamado de servidor. O protocolo define que um dispositivo escravo só pode enviar informações pela rede após uma requisição do servidor (Modbus Organization, Inc, 2006).

O protocolo possui três categorias de códigos de função, sendo elas a categoria pública, categoria de funções definidas pelo usuário e categoria de códigos

reservados. Esses códigos possuem valores entre 1 e 127, e as categorias são definidas conforme a Tabela 1 (Modbus Organization, 2012).

Tabela 1 - Faixas das categorias de funções.

Faixa de códigos	Categoria
110 – 127	Funções públicas
100 – 109	Funções definidas pelo usuário
72 – 99	Funções públicas
65 – 71	Funções definidas pelo usuário
1 – 64	Funções públicas

Fonte – Adaptado de Modbus Organization (2012).

O protocolo Modbus possui diferentes funções públicas conforme apresentado no Quadro 1, que inclui funções para escrita e leitura de registradores.

Quadro 1 - Funções públicas.

				Código das funções		
				Código	Sub códigos	Hex
Data Access	Bit access	Physical Discrete Inputs	Read Discrete Inputs	02		02
		Internal Bits Or Physical coils	Read Coils	01		01
			Write Single Coil	05		05
			Write Multiple Coils	15		0F
	16 bits access	Physical Input Registers	Read Input Register	04		04
			Read Holding Registers	03		03
		Internal Registers Or Physical Output Registers	Write Single Register	06		06
			Write Multiple Registers	16		10
			Read/Write Multiple Registers	23		17
			Mask Write Register	22		16
			Read FIFO queue	24		18
	File record access		Read File record	20		14
			Write File record	21		15
	Diagnostics		Read Exception status	07		07
		Diagnostic	08	00-18,20	08	
		Get Com event counter	11		0B	
		Get Com Event Log	12		0C	
		Report Server ID	17		11	
		Read device Identification	43	14	2B	
Other		Encapsulated Interface Transport	43	13,14	2B	
		CANopen General Reference	43	13	2B	

Fonte – Adaptado de Modbus Organization (2012).

As funções públicas são todas definidas e documentadas pelo protocolo Modbus. Com elas é possível, por exemplo, escrever em registradores, comandar os dispositivos mesmo a distância e obter dados dos dispositivos e processos. A disponibilidade das funções nos dispositivos depende do que foi implementado em cada dispositivo pelos fabricantes (Modbus Organization, 2012).

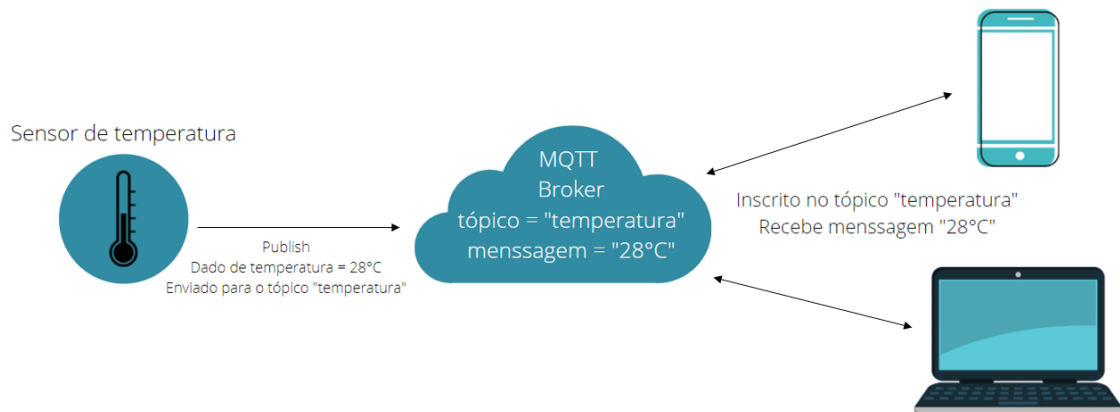
2.1.5 MQTT - *Message Queue Telemetry Transport*

A primeira versão do protocolo MQTT foi criada em 1999 por Andy Stanford-Clark e Arlen Nipper. O desenvolvimento do MQTT foi motivado pela necessidade de se ter um protocolo que consumisse pouca energia e pouca largura de banda para se comunicar com dispositivos em oleodutos via satélite. Os objetivos iniciais continuaram sendo levados em consideração nas últimas versões do protocolo, no entanto, atualmente o foco primário é nos casos de uso relacionado ao IoT (HiveMQ, 2020).

O protocolo MQTT funciona através de um modelo de publicador/assinante, do termo em inglês *publish / subscribe*. Esse modelo funciona de uma forma diferente dos modelos tradicionais cliente-servidor. No MQTT, os publicadores e assinantes em nenhum momento tem contato direto um com o outro, não tendo nenhuma troca de endereços IPs e portas entre eles, a conexão é realizada através de um terceiro componente chamado de *broker*. O *broker* é responsável por filtrar todas as mensagens e distribuir para os corretos assinantes (HiveMQ, 2020).

O MQTT utiliza um sistema de tópicos, onde os assinantes se conectam para receberem as mensagens publicadas pelos publicadores. Cada mensagem enviada contém um determinado tópico, onde o *broker* determina quais assinantes irão receber a mensagem. Esse comportamento pode ser observado na Figura 4 que apresenta a arquitetura do protocolo (HiveMQ, 2020).

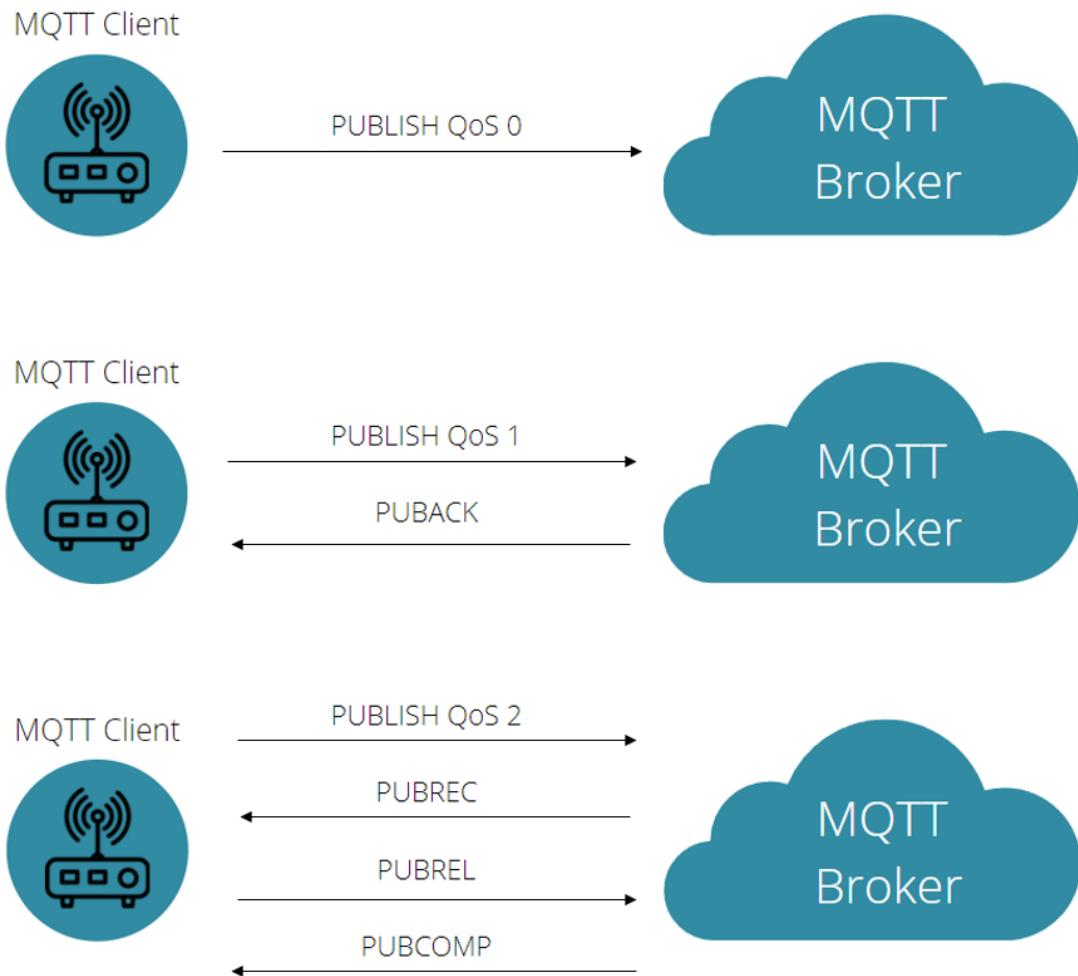
Figura 4 - Arquitetura básica de um modelo MQTT *broker / client*.



Fonte – Elaborado pelo autor.

Outra importante característica do protocolo MQTT são os níveis de qualidade de serviço, do termo em inglês *Quality of Service* (QoS). São definidos pelo protocolo três níveis de QoS, exemplificados na Figura 5. O nível 0 é o mais simples, nele a mensagem é enviada pelo cliente para o *broker*, mas não há nenhuma garantia que a mensagem foi entregue com sucesso, o *broker* não envia nenhuma resposta confirmando o recebimento. No nível 1, o protocolo garante que a mensagem será entregue para o receptor pelo menos uma vez, o cliente armazena a mensagem até receber um PUBACK do receptor confirmando o recebimento da mensagem, caso ocorra algum atraso no recebimento do PUBACK uma nova mensagem é enviada, quando esse QoS é utilizado o receptor deve ser capaz de lidar com mensagens duplicadas. Já no nível QoS 2, que é o mais alto nível do protocolo, é garantido que a mensagem será recebida apenas uma única vez pelo receptor, portanto é o nível mais seguro, porém o mais lento pois o cliente envia e recebe quatro mensagens (HiveMQ, 2020).

Figura 5 - Níveis de QoS.



Fonte – Elaborado pelo autor.

Outro importante fator em protocolos de comunicação é a segurança. Para essa questão, é muito utilizado o protocolo TLS (*Transport Layer Security*), responsável por criptografar o tráfego de dados. O protocolo garante uma conexão segura entre o *broker* e os clientes (HiveMQ, 2020).

A partir dos conceitos básicos apresentados, foi realizada uma pesquisa sobre produtos já disponíveis no mercado similares ao *gateway* proposto neste trabalho. O tópico a seguir apresenta esses produtos e suas principais características.

2.2 Pesquisa de Mercado

Com as perspectivas do avanço da Internet das Coisas (IoT – *Internet of Things*) no ramo industrial, algumas empresas já estão trabalhando no

desenvolvimento de dispositivos para adaptação de máquinas e equipamentos mais antigos, originalmente sem interface de comunicação IIoT, para se comunicar via Internet. A fim de entender o que já existe no mercado, foi feita uma pesquisa, e as características desses produtos estão descritas a seguir.

2.2.1 I-MD DUO ETHERNET / USB / RS-232®

A empresa brasileira CTN Tecnologia CNC®, desenvolve módulos para comunicação com máquinas CNC via Wi-Fi. Esses módulos são conectados na porta RS-232, ethernet ou USB dos controladores CNC. Na Figura 6 é apresentado o dispositivo desenvolvido pela empresa.

Figura 6 - dongle desenvolvido pela empresa CTN Tecnologia CNC®.



Fonte - CTN Tecnologia CNC®

Entre as principais características desses dispositivos, está a funcionalidade de envio de programas para as máquinas, e acesso ao repositório de programas no servidor pelo operador diretamente pela máquina.

2.2.2 MB5901B Series®

A empresa Atop Technologies®, com sede em Taiwan, desenvolve diferentes produtos e soluções voltados a área de comunicação industrial, incluindo *gateways* e roteadores para comunicação entre máquinas e dispositivos. A empresa possui um *gateway* em seu catálogo semelhante ao protótipo proposto neste trabalho, o MB5901B Series®, apresentado na Figura 7.

Figura 7 - Gateway desenvolvido pela empresa Atop Technologies®.



Fonte - Atop Technologies®.

Entre as características do produto estão o suporte a redes celular, que permite o envio de informações e recebimento de comandos via SMS, e o suporte ao protocolo Modbus TCP e RTU via as portas RS232 e RS485.

2.2.3 AIRGATE 4G®

O AIRGATE 4G®, apresentado na Figura 8, é um roteador VPN industrial para rede celular desenvolvido pela empresa brasileira Novus®. A empresa é especializada na fabricação e projeto de instrumentos para mediação, controle, aquisição, registro e supervisão de dados, atuando principalmente na área de automação industrial.

Figura 8 - Roteador VPN industrial para rede celular desenvolvido pela Novus®.



Fonte - Novus®.

O dispositivo possui características muito similares ao protótipo proposto e desenvolvido nesse trabalho. As principais características são o suporte a rede celular, interface de comunicação RS232 e RS485, e o suporte a protocolos de comunicação Modbus RTU e TCP, e MQTT.

2.2.4 WF3 for Universal Connectivity®

A empresa WireFreeCNC®, com sede nos Estados Unidos, possui em seu catálogo um produto similar ao desenvolvido pela brasileira CTN Tecnologia CNC®, que é apresentado na Figura 9. A principal diferença está no suporte a periféricos, então é possível adicionar sensores na máquina e enviar todas essas informações via Wi-Fi.

Figura 9 - dongle desenvolvido pela empresa WireFreeCNC®.



Fonte - WireFreeCNC®.

Outra diferença, é que esse *dongle* também envia dados da máquina. A empresa também conta com um software em seu portfólio, com ferramentas para análise dos dados extraídos das máquinas CNC.

2.2.5 Industrial IoT Platform®

A empresa MachineMetrics®, com sede localizada nos Estados Unidos, desenvolve uma plataforma para Internet das coisas industrial (IIoT), com objetivo de utilizar os dados disponibilizados pelas máquinas, e com isso, reduzir o tempo de ociosidade das máquinas e melhorar a produtividade. A empresa oferece soluções de *software* e *hardware*. O *dongle* desenvolvido pela empresa, apresentado na Figura 10, se comunica com controladores CNC como Fanuc, Siemens, Heidenhain, Haas, entre outros. Ele lê diferentes dados da máquina, como estado da máquina, alarmes, velocidade de rotação, velocidades de avanço e parâmetros. O dispositivo envia os dados através de conexão ethernet, Wi-Fi ou via rede móvel.

Figura 10 - dongle desenvolvido pela empresa MachineMetrics®.



Fonte - MachineMetrics®.

No Quadro 2 é apresentado um resumo sobre os produtos similares disponíveis no mercado.

Quadro 2 - Resumo dos produtos similares disponíveis no mercado.

Empresa	Nome do Produto	Relação com TCC
CTN Tecnologia CNC®	I-MD DUO ETHERNET / USB / RS-232®	A empresa desenvolve um dispositivo que permite enviar programas para as máquinas através da rede Wi-Fi.
Atop Technologies®	MB5901B Series®	O <i>gateway</i> desenvolvido pela empresa permite enviar mensagens via SMS com informações de dispositivos e sensores lidos através do protocolo Modbus, e receber comandos via SMS.
Novus®	AIRGATE 4G®	O roteador possui interface celular, interface serial (RS485 e RS232) e suporte a protocolos como Modbus e MQTT.
WireFreeCNC®	WF3 for Universal Connectivity®	O dispositivo desenvolvido por essa empresa permite enviar programas para a máquina, além disso lê e envia

		dados da máquina e sensores através da rede Wi-Fi.
MachineMetrics®	Industrial IoT Platform®	Essa empresa desenvolve o <i>dongle</i> para comunicação com a máquina, que permite ler dados da máquina e enviar dados para a máquina através de uma rede ethernet, Wi-Fi ou 4G/5G. Além disso, desenvolve o software para gerenciamento destes dados.

Fonte – Elaborado pelo autor.

Como vimos, o mercado já apresenta algumas opções para adaptar dispositivos de campo e máquinas ao IIoT, cada um com suas características particulares. Vistos os produtos semelhantes já existentes no mercado, no próximo capítulo é apresentada uma pesquisa sobre trabalhos científicos correlatos ao trabalho proposto.

2.3 Trabalhos Correlatos

Internet das coisas (IoT), é um assunto que já foi tema de diferentes trabalhos científicos. Com o objetivo de entender os artigos que já foram desenvolvidos nessa área e que tenham correlação com o tema deste trabalho, foi feita uma pesquisa por artigos científicos na plataforma IEEE Xplore e no Portal de Periódicos CAPES/MEC.

Em seu artigo, Batalla (2020), analisou a transmissão de vídeo por rede WiFi e 5G em ambiente industrial. O objetivo da pesquisa foi analisar como o ambiente industrial, que possui diferentes ruídos, impacta na transmissão de vídeo, e comparar os impactos desses ruídos na transmissão via Wi-Fi e 5G. Para fazer essa análise, foi transmitido um vídeo, com duração de três minutos em quatro diferentes situações. Sendo elas via Wi-Fi com o maquinário desligado e depois ligado, e via 5G com as máquinas desligadas e depois ligadas. Após a transmissão do vídeo, ele foi apresentado a 12 trabalhadores que responderam três perguntas, sendo uma delas uma nota de 1 a 5 em relação a qualidade do vídeo. Os resultados mostraram que a média das notas para a transmissão via 5G com as máquinas desligadas e ligadas foram estatisticamente as mesmas, ambas próximas de 5. Já no segundo cenário, que foi a transmissão via Wi-Fi, com o maquinário desligado a média das notas foi de 4,

enquanto que com o maquinário ligado a média caiu para 3, mostrando que os ruídos eletromagnéticos têm um impacto significativo na qualidade da transmissão.

Em seu artigo, Vitturi e Sauter (2019), contextualizam os sistemas de comunicação industrial desenvolvidos e utilizados nas últimas décadas, na atualidade e as perspectivas para o futuro, incluindo os requisitos necessários, configurações e indicadores de performance. Na seção de redes industriais da atualidade, são apontadas três tecnologias fortemente utilizadas na indústria, que é a comunicação *Fieldbus*, redes *Real-Time Ethernet* e redes *Wireless*. Essa última, é apontada como a tecnologia mais recente introduzida no cenário de comunicação industrial, e tem como principal vantagem a eliminação dos cabos, que é uma vantagem do ponto de vista comercial e permite conexões com dispositivos que via cabo seria inviável, e como desvantagem e possíveis problemas foi apontado erros no envio de pacotes e possíveis *delays*. Nas perspectivas para o futuro, os autores apontam os sistemas IIoT, e novas possibilidades na área de comunicação industrial. Uma das tecnologias apontadas são as redes 5G, a primeira rede de celular desenvolvida pensando em aplicações industriais, com baixa latência e alta taxa de confiança e disponibilidade.

Em seu artigo, Sun *et al.* (2019), desenvolveu um *gateway* para aplicações IIoT utilizando um *Raspberry Pi*. O dispositivo desenvolvido, recebe via protocolo *Modbus* informações e dados de sensores, e envia essas informações para serviços de armazenamento em nuvem utilizando protocolo MQTT (*Message Queuing Telemetry Transport*). Os pesquisadores também desenvolveram um aplicativo para configurar o *Raspberry Pi* e acessar o servidor MQTT rodando na nuvem. Como futuras melhorias, os autores pensam em implementar comunicação celular no *gateway* e integrar o sistema com dispositivos PLC e com sistemas SCADA.

Em seu artigo, Wu *et al.* (2019), apresentou o desenvolvimento e os resultados de uma rede *Wireless* destinada a Internet das coisas industrial (IIoT) baseada no padrão IEEE 802.15.4-2015 TSCH (*Time Slotted Channel Hopping*) *Access Mode*, que define as especificações das redes de comunicação sem fio com baixa taxa de transmissão de dados, da sigla LR-WPAN. O objetivo do trabalho foi desenvolver um sistema de comunicação com baixo *delay*, alta confiança e baixo consumo de energia, que são características importantes em dispositivos IIoT. No trabalho, foi desenvolvido um algoritmo para controlar o fluxo de dados recebidos dos sensores. O algoritmo determina quando os sensores devem transmitir os dados, buscando completar o ciclo

de transmissão de todos os sensores no menor tempo possível, com isso, todos os sensores conseguem transmitir sem colisão de dados. Para validação do trabalho, foram realizados testes utilizando oito sensores simultaneamente. A latência média dos pacotes ficou em torno de 10 ms, e a taxa dos pacotes enviados com sucesso ficou em 98%, o que foram resultados considerados bons pelos autores, e que validaram a efetividade do algoritmo.

Em seu artigo, Kehl *et al.* (2020), apresentou e comparou três possíveis arquiteturas que ilustram como o 5G pode ser utilizado em ambientes industriais. Conforme abordado no artigo, o 5G é apontado como uma tecnologia chave para a transformação e criação das fábricas inteligentes, devido a sua baixa latência e largura de banda, que são pontos fundamentais para aplicações industriais. De acordo com os autores, para entender o potencial dessa tecnologia no ambiente industrial, não se pode analisar apenas a comunicação entre um sensor e uma aplicação *cloud*. Por esse motivo, foram propostas três arquiteturas para análise, a *Closed-Loop*, *Cloud-in-the-Loop* e a *Cloud-on-the-Loop*. A *closed-loop*, descreve o caso clássico, onde o sensor envia os dados diretamente ao controlador situado próximo a máquina, nesse caso a comunicação com a aplicação *cloud* é apenas com o controlador, para salvar os dados. Na arquitetura *cloud-on-the-loop*, o sensor se comunica via 5G com o controlador e com a aplicação *cloud*, onde os dados são armazenados e analisados. Na terceira arquitetura proposta, os dados do sensor são enviados apenas para a aplicação *cloud*, que é responsável pelas análises e comunicação com os equipamentos na fábrica. No artigo também foi especificado o plano de validação das arquiteturas, que envolvem pontos como tempo de reação da comunicação, resiliência, escalabilidade e uma taxa de envio de dados efetiva e constante. Como um trabalho futuro, os autores planejam implementar essas arquiteturas e testar na prática como elas se comportam comparadas a aplicações que utilizam Wi-Fi.

No Quadro 3 é apresentado um resumo sobre os trabalhos correlatos ao assunto do trabalho proposto.

Quadro 3 - Resumo dos trabalhos correlatos.

Autores	Ano de publicação	Título do artigo	Relação com o trabalho
Jordi Batalla	2020	On Analyzing Video Transmission Over Wireless WiFi and 5G C-Band in Harsh IIoT Environments	Neste artigo, foi realizado um estudo e testes comparando a transmissão de vídeo em ambientes industriais via Wi-Fi e 5G. Os resultados demonstraram que quando as máquinas são ligadas, e por consequência mais ruídos são gerados no ambiente, a transmissão via Wi-Fi sofre um impacto significativo, enquanto o comportamento da transmissão via 5G se mantém praticamente inalterado.
Stefano Vitturi Thilo Sauter	2019	Industrial Communication Systems and Their Future Challenges: Next-Generation Ethernet, IIoT, and 5G	Neste artigo, é apresentado um comparativo entre diferentes protocolos de comunicação industrial, 5G, IIoT, <i>wireless networks</i> .
Changqing Sun Kun Guo Zhaoxia Xu Jianhui Ma Dairong Hu	2019	Design and Development of Modbus/MQTT Gateway for Industrial IoT Cloud Applications Using Raspberry Pi	Neste artigo, foi explicado o desenvolvimento de um <i>gateway</i> utilizando um <i>Raspberry Pi</i> para ler informações via protocolo Modbus e enviar essas informações para serviços <i>cloud</i> via protocolo MQTT. Foi utilizada conexão <i>wireless</i> para envio das informações.
Ho-Ting Wu Kai-Wei Ke Song-Feng Po-Hung Chen Guan-De Lee Chen-Yu Tseng Chen-Yu Ho	2019	The Implementation of Wireless Industrial Internet of Things (IIoT) Based upon IEEE 802.15.4-2015 TSCH Access Mode	Neste artigo, foi descrito e apresentado os resultados de uma rede <i>wireless</i> destinada a Internet das Coisas Industrial (IIoT) utilizando o padrão IEEE 802.15.4.
Pierre Kehl Dirk Lange Felix Maurer Gábor Németh Daniel Overbeck Sven Jung Niels König Robert Schmitt	2020	Comparison of 5G Enabled Control Loops for Production	Neste artigo, foi apresentado três arquiteturas IIoT de como a tecnologia 5G pode ser utilizada em ambientes industriais.

Fonte – Elaborado pelo autor.

Através da pesquisa por trabalhos correlatos, foi possível encontrar diferentes artigos na área da Internet das Coisas Industrial, e notar como a tecnologia 5G é considerada um ponto chave para o uso em maior escala de dispositivos e sistemas

IIoT. A partir das pesquisas realizadas para um correto entendimento dos conceitos básicos envolvidos no trabalho e dos artigos correlatos, foi desenvolvida a metodologia para o desenvolvimento do *gateway* IIoT proposto.

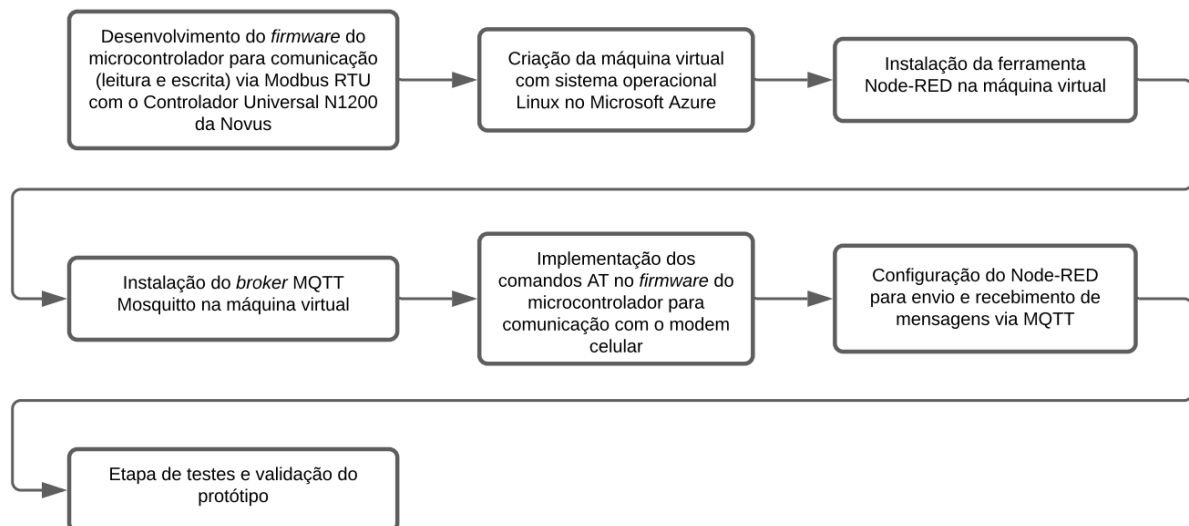
3. METODOLOGIA

Nesta seção, são apresentadas as etapas que foram seguidas para o desenvolvimento do trabalho, um infográfico exemplificando o trabalho proposto, o sistema proposto, os materiais utilizados e os testes para validação da solução.

3.1 Etapas do desenvolvimento do protótipo

A implementação do protótipo foi dividida em 7 etapas como é apresentado na Figura 11. A divisão foi planejada dividindo o protótipo em partes isoladas, de forma que cada parte pudesse ser implementada, testada e validada antes de realizar a integração entre elas.

Figura 11 – Diagrama de blocos das etapas do desenvolvimento do protótipo.



Fonte – Elaborado pelo autor.

3.2 Materiais utilizados

Para implementar o protótipo proposto, foram utilizados um controlador com suporte a comunicação através do protocolo Modbus RTU, um kit de desenvolvimento, dois módulos conversores RS485, um modem para comunicação com rede celular e uma máquina virtual rodando em um serviço de nuvem. A seguir são apresentados mais detalhes sobre os materiais e equipamentos utilizados.

O equipamento escolhido para realizar a comunicação através do protocolo Modbus RTU foi o Controlador Universal N1200®, desenvolvido pela marca Novus®. O modelo é um controlador de processos extremamente versátil, que aceita a maioria

dos sensores e sinais utilizados na indústria. O controlador oferece suporte ao protocolo Modbus RTU através do padrão RS485, suportando os comandos *Read Holding Register* (leitura de bloco de registradores do tipo *holding*), *Preset Single Register* (escrita em um único registrador do tipo *holding*), *Force Single Coil* (escrita em um único bit do tipo *coil*) e *Preset Multiple Register* (escrita em bloco de registradores do tipo *holding*).

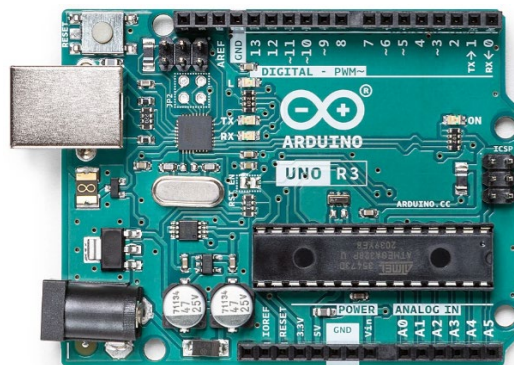
Figura 12 - Controlador Universal N1200®.



Fonte – Retirado de novus.com.br.

O kit de desenvolvimento escolhido foi o Arduino UNO®, apresentado na Figura 13. Optou-se por utilizar esse kit pois ele atendeu os requisitos técnicos necessários como suporte a comunicação serial e entradas e saídas digitais. Além disso, a biblioteca ModbusMaster se mostrou funcional atendendo os requisitos necessários para a comunicação Modbus RTU.

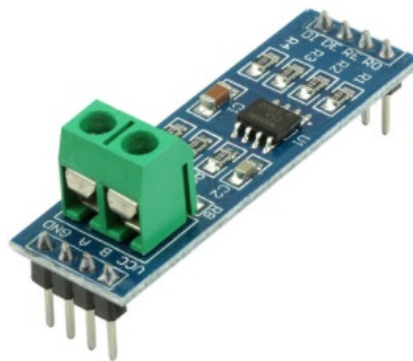
Figura 13 - Arduino UNO®.



Fonte – Retirado de arduino.cc.

Pelo fato do kit de desenvolvimento Arduino UNO® não oferecer nativamente compatibilidade a redes do padrão RS485, foram utilizados dois módulos conversores RS485 da marca ChipSCE®. Esse módulo utiliza o CI (Circuito integrado) MAX485, que faz a conversão do sinal para o padrão RS485. Esses módulos foram utilizados para conectar o controlador ao kit de desenvolvimento, e o kit de desenvolvimento ao modem celular.

Figura 14 - Módulo adaptador RS485.



Fonte – Retirado de chipsce.com.br.

Para a comunicação via rede celular, foi utilizado o modem CINTERION EXS82-W®, da marca Thales Group®. Entre as principais características desse modem estão o suporte a tecnologia Cat.M1, controle via comandos AT e suporte aos protocolos de comunicação TCP/IP e MQTT

Figura 15 - Modem celular CINTERION EXS82-W.



Fonte – Elaborado pelo autor.

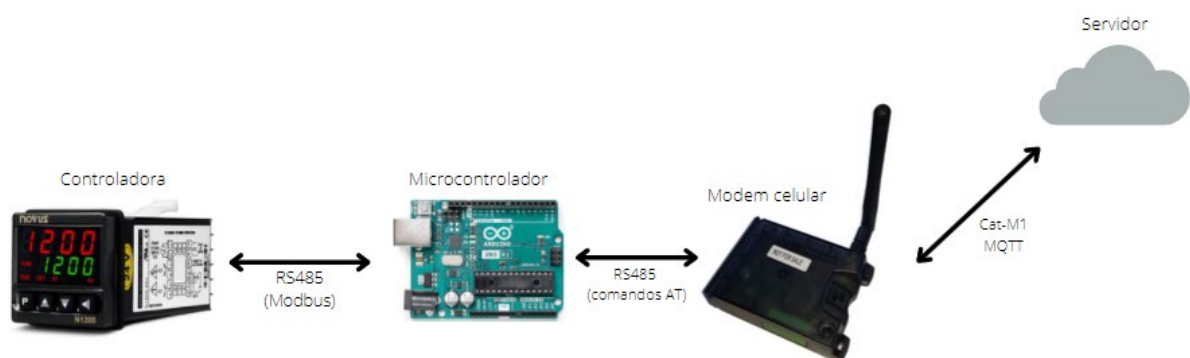
A máquina virtual utilizada para receber e enviar as mensagens através do protocolo MQTT foi criada no serviço de nuvem Microsoft Azure®. Na VM (máquina virtual), foi instalado o sistema operacional Ubuntu®, da distribuição Linux, além da ferramenta Node-RED® e do *broker* MQTT Mosquitto®. O *broker* MQTT foi instalado diretamente na máquina virtual e utilizado especificamente para os testes do *gateway*.

Visto a explicação e características dos materiais e equipamentos utilizados no desenvolvimento do protótipo, no próximo capítulo é explicado mais a fundo o sistema proposto e sua arquitetura.

3.3 Sistema proposto

É proposto neste trabalho um *gateway* capaz de se comunicar com redes de campo (*fieldbus*) que utilizam o protocolo de comunicação industrial Modbus RTU, e que utilize rede celular 2G/3G/4G para enviar dados lidos pela rede de campo para um servidor localizado em qualquer lugar do mundo, e receber comandos deste servidor e encaminhá-los pela rede de campo ao dispositivo. A comunicação entre o *gateway* e o servidor é realizada utilizando o protocolo MQTT. O sistema proposto é apresentado na Figura 16.

Figura 16 - Infográfico do sistema proposto.



Fonte – Elaborado pelo autor.

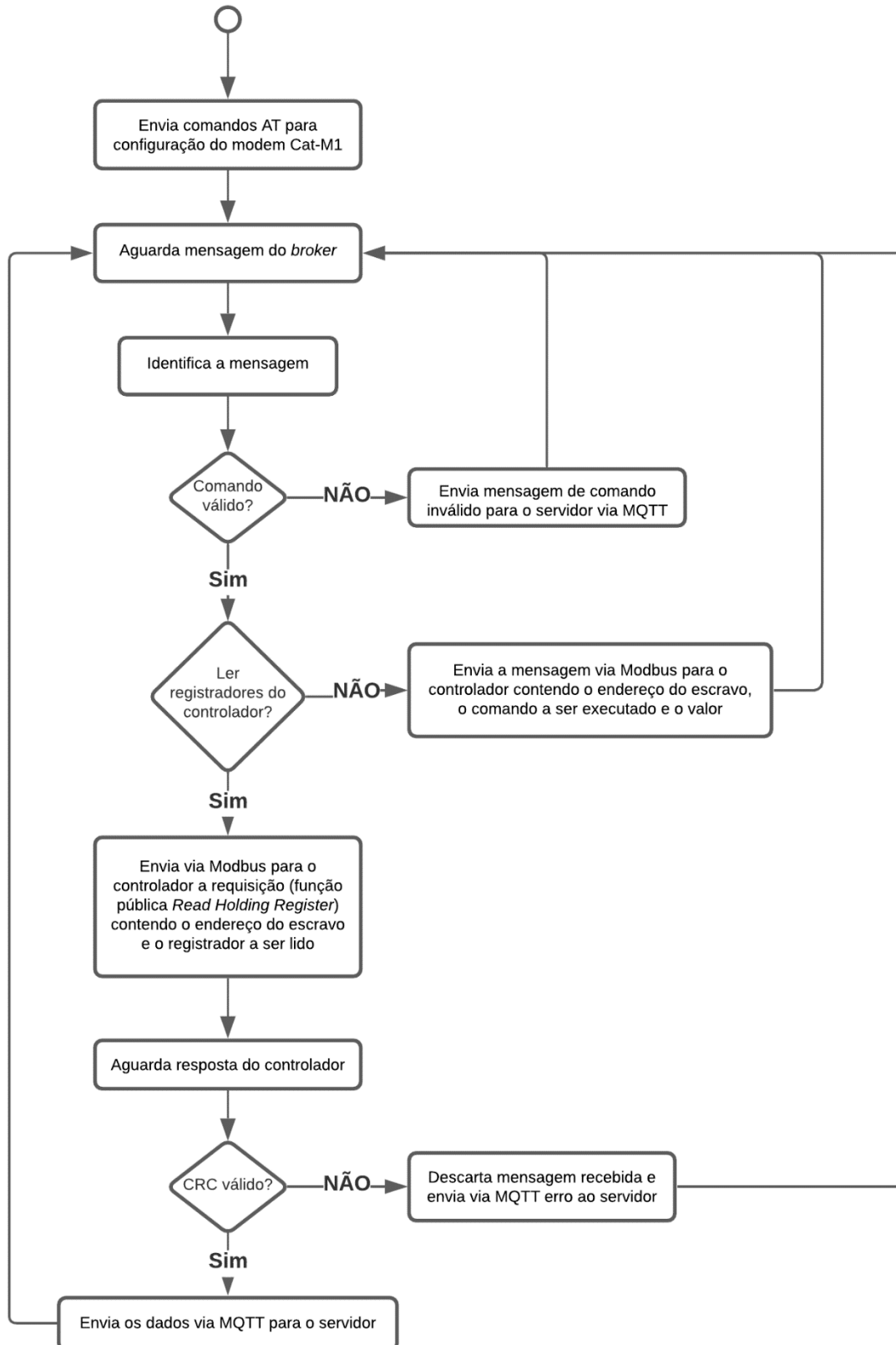
Na implementação do protótipo foi utilizado o kit de desenvolvimento Arduino UNO®, junto com a biblioteca ModbusMaster. Na interface de comunicação Modbus RTU, o kit de desenvolvimento foi configurado como o dispositivo mestre da rede, e o controlador universal como escravo, o que permite adicionar mais dispositivos de campo e sensores ao kit de desenvolvimento. A comunicação entre o controlador e o kit de desenvolvimento foi realizada através da interface RS485. A rede suporta mais

de um dispositivo, sendo a identificação de cada dispositivo realizada através do ID dos dispositivos escravos.

A estrutura de tópicos do protocolo MQTT foi utilizada para enviar os dados do kit de desenvolvimento para o servidor, e receber os dados do servidor através do modem Cat-M1. Com o objetivo de garantir o recebimento das mensagens uma única vez, foi utilizado o QoS 2.

A comunicação entre o kit de desenvolvimento e o modem da Thales Group® também foi realizada através da interface RS485. A comunicação com o modem ocorre através de comandos AT. Na inicialização do kit de desenvolvimento os comandos para configuração do modem, como comandos para ativar serviços de internet e dados para conexão no *broker* MQTT, são enviados. Após isso, o *gateway* tem seu funcionamento conforme exemplificado no algoritmo em forma de fluxograma apresentado na Figura 17.

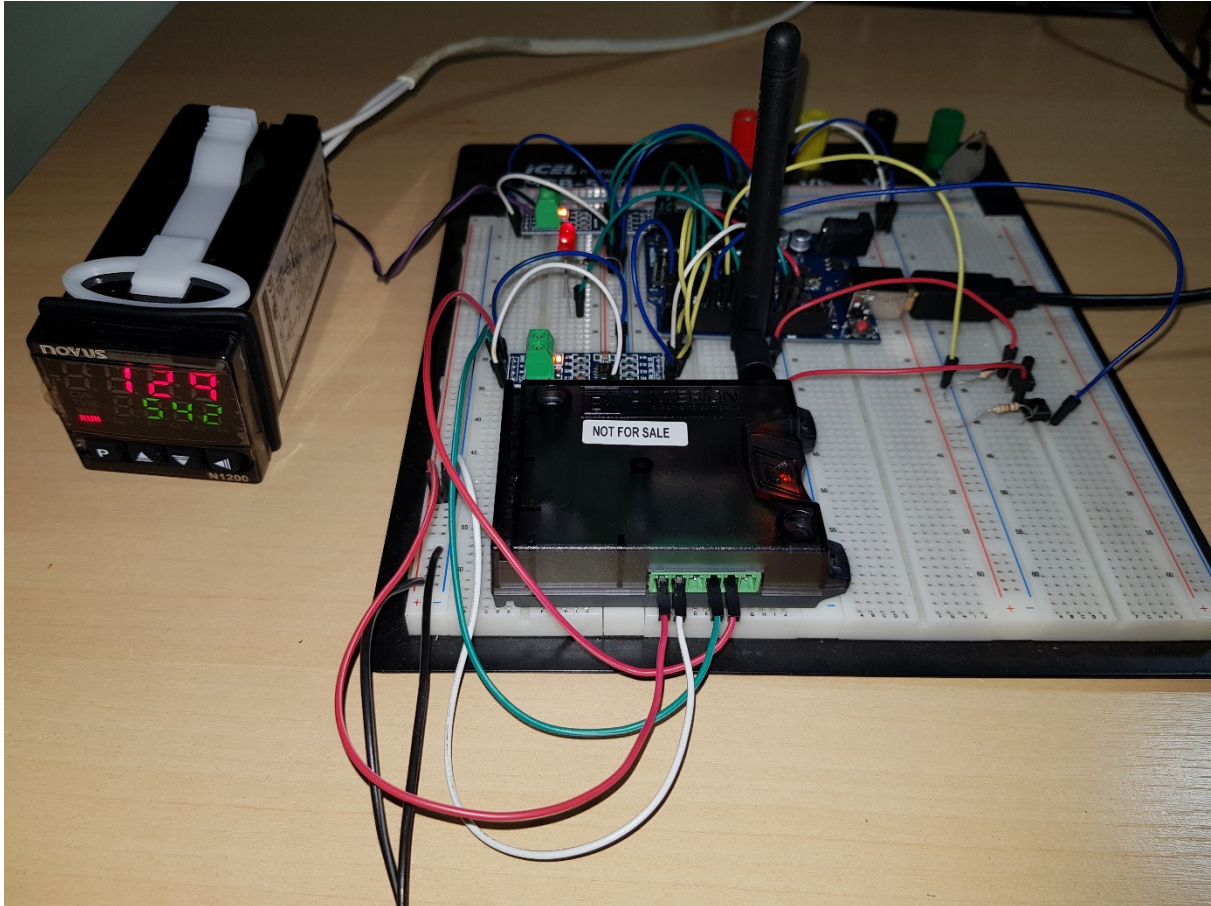
Figura 17 - Fluxograma do algoritmo desenvolvido.



Fonte – Elaborado pelo autor.

O protótipo desenvolvido pode ser observado na Figura 18.

Figura 18 – Protótipo desenvolvido.



Fonte – Elaborado pelo autor.

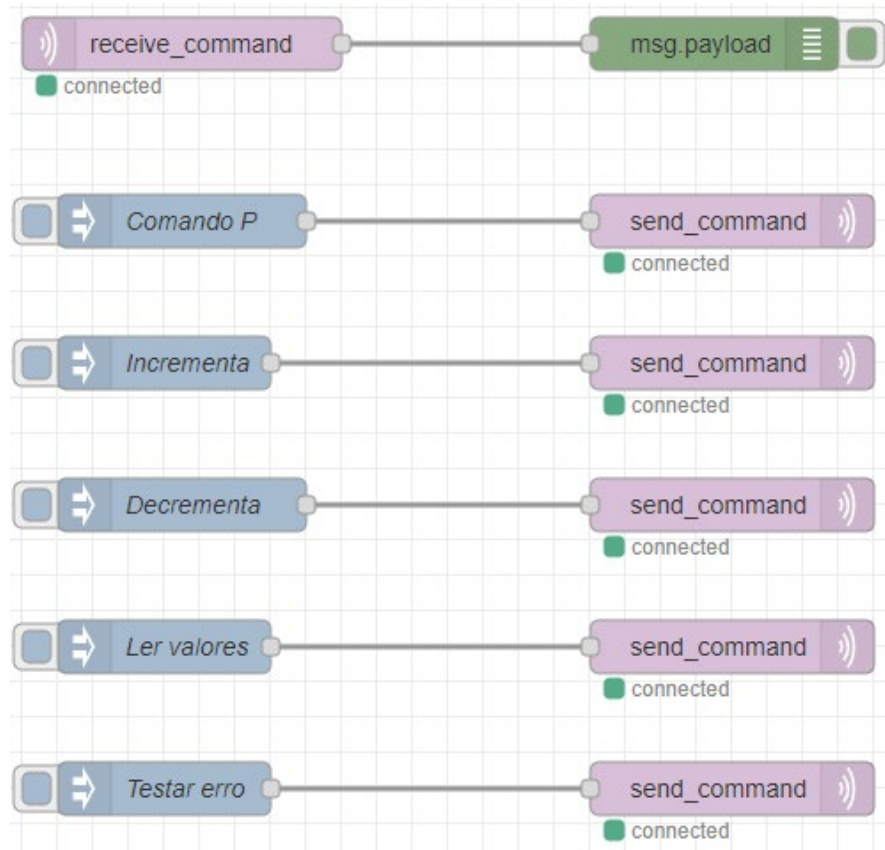
Com o protótipo do sistema implementado foi iniciada a etapa de testes para validação do sistema. No próximo capítulo é explicado em mais detalhes como foram realizados os testes.

3.4 Testes e validação

Para a realização dos testes e validação do sistema, foi criada uma máquina virtual com sistema operacional Ubuntu[®]-Linux na plataforma Microsoft Azure[®]. Nessa máquina foram instaladas a ferramenta de programação visual Node-RED[®] e o *broker* MQTT Mosquitto[®]. Na plataforma Node-RED[®] foi implementada a interface para testes com o *gateway* conforme apresentado na Figura 19. Na interface foram adicionados botões que enviam mensagens para escrever nos registros do controlador, como botões para incrementar e decrementar o valor da tela principal do controlador, um botão que realiza a troca de tela, um botão que lê alguns registradores predefinidos

no *firmware* do kit de desenvolvimento e um botão para teste de erro em caso de falha na comunicação.

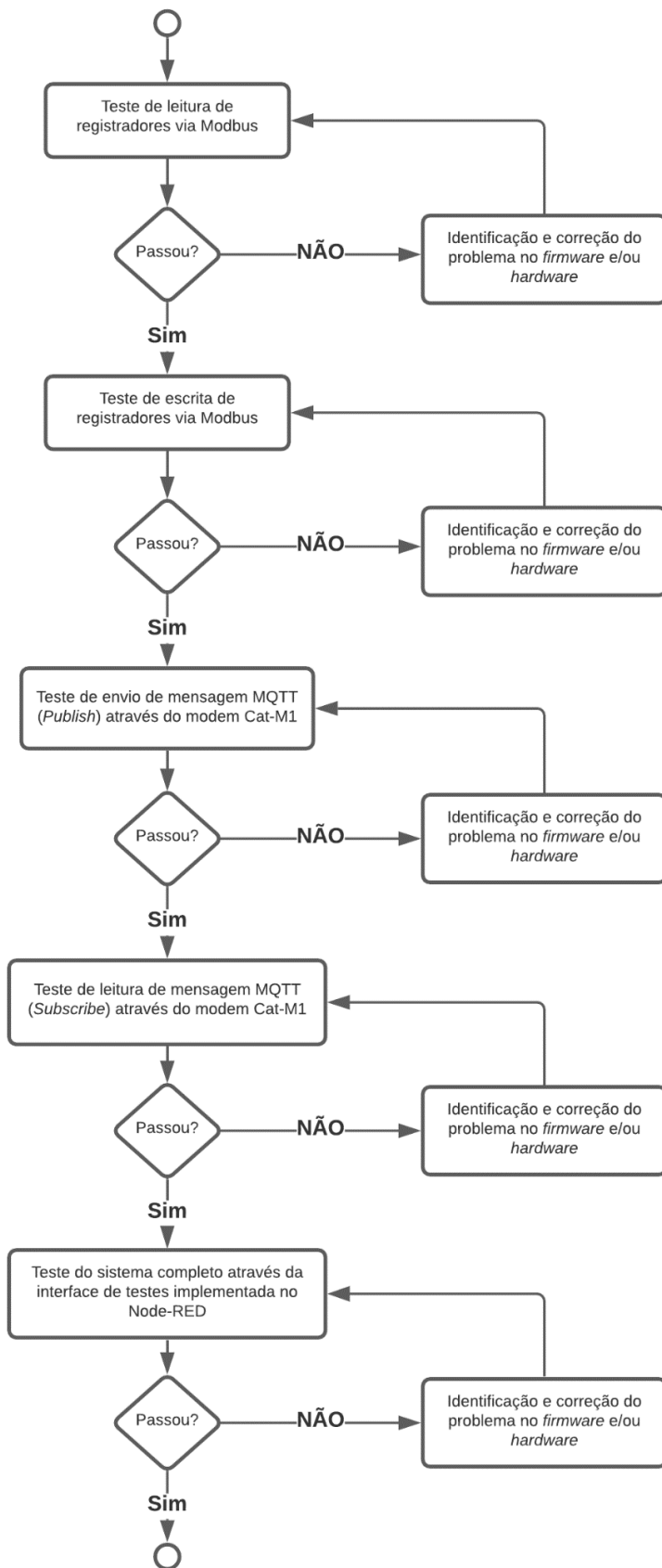
Figura 19 - Interface de testes implementada no Node-RED®.



Fonte – Elaborado pelo autor.

Os testes realizados foram divididos em etapas conforme o fluxograma apresentado na Figura 20. As etapas foram divididas com o objetivo de testar a comunicação entre o controlador e o kit de desenvolvimento, e o kit de desenvolvimento e o servidor através do modem, inicialmente de forma isolada, com o objetivo de identificar problemas que pudessem ocorrer na comunicação. Após todos os testes individuais validados, foram realizados os testes da solução completa utilizando a interface implementada na máquina virtual.

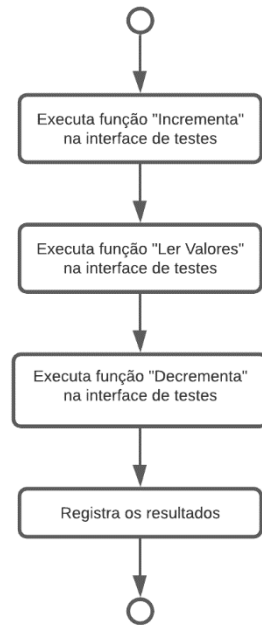
Figura 20 - Fluxograma de testes para validação do sistema.



Fonte – Elaborado pelo autor.

Os testes individuais foram realizados ainda na etapa de desenvolvimento do protótipo. Já os testes do sistema completo foram divididos em quatro etapas, conforme apresentado no fluxograma da Figura 21. A estrutura de testes funcionais foi repetida 30 vezes.

Figura 21 - Fluxograma de testes do sistema completo.



Fonte – Elaborado pelo autor.

Essas foram as etapas utilizadas para testar e validar o sistema proposto. No próximo capítulo são apresentados os resultados e uma análise em relação ao que foi obtido com a solução.

4. ANÁLISE DOS RESULTADOS

Nos testes, que foram executados em diferentes momentos do dia, foram analisados o êxito no envio e recebimento dos comandos no kit de desenvolvimento e no servidor, além da correta comunicação via Modbus com o controlador. No Quadro 2 são apresentados os resultados obtidos.

Quadro 4 - Resultado dos testes funcionais realizados.

Teste	Incrementa	Ler Valores Primeira mensagem	Ler Valores Segunda mensagem	Ler Valores Terceira mensagem	Decrementa
1	OK	OK	OK	OK	OK
2	OK	OK	OK	OK	OK
3	OK	OK	OK	OK	OK
4	OK	OK	OK	OK	OK
5	OK	OK	Erro - perda do pacote	Erro - perda do pacote	OK
6	OK	OK	OK	OK	OK
7	OK	OK	OK	OK	OK
8	OK	OK	OK	OK	OK
9	OK	OK	OK	OK	OK
10	OK	OK	OK	OK	OK
11	OK	OK	OK	OK	OK
12	OK	OK	OK	OK	OK
13	OK	OK	OK	OK	OK
14	OK	OK	OK	OK	OK
15	OK	OK	OK	OK	OK
16	OK	OK	OK	OK	OK
17	OK	OK	OK	OK	OK
18	OK	OK	OK	OK	OK
19	OK	OK	OK	OK	OK
20	OK	Erro - comando não recebido no modem	Erro - comando não recebido no modem	Erro - comando não recebido no modem	OK
21	OK	OK	OK	OK	OK
22	OK	OK	OK	OK	OK
23	OK	OK	OK	OK	OK
24	OK	OK	OK	OK	OK
25	OK	OK	OK	OK	OK
26	OK	OK	OK	OK	OK
27	OK	OK	OK	OK	OK

28	OK	OK	OK	OK	OK
29	OK	OK	OK	OK	OK
30	OK	OK	OK	OK	OK

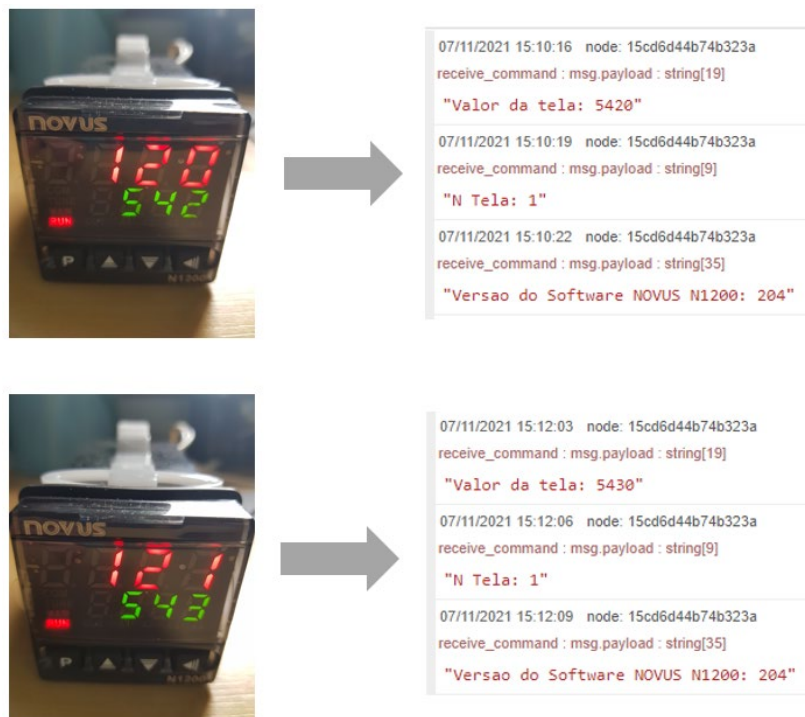
Fonte – Elaborado pelo autor.

Analisando o quadro com os resultados, podemos constatar que os comandos Incrementa e Decrementa da interface de testes foram executados corretamente em 100% dos testes. Já o comando de leitura de registros falhou em dois dos 30 testes, o que representa um percentual de 93,33% de leituras corretas.

Os erros são provavelmente relacionados ao *timeout* do cliente MQTT, que no modem da Thales® é de até 60 segundos. Uma maneira de solucionar esse problema seria reiniciar o cliente a cada 60 segundos, de forma que a conexão esteja sempre ativa para ler as mensagens no tópico.

Na Figura 22 é exemplificado os comandos Incrementa e Ler valores.

Figura 22 - Representação dos comandos Incrementa e Ler valores implementados na interface de testes.



Fonte – Elaborado pelo autor.

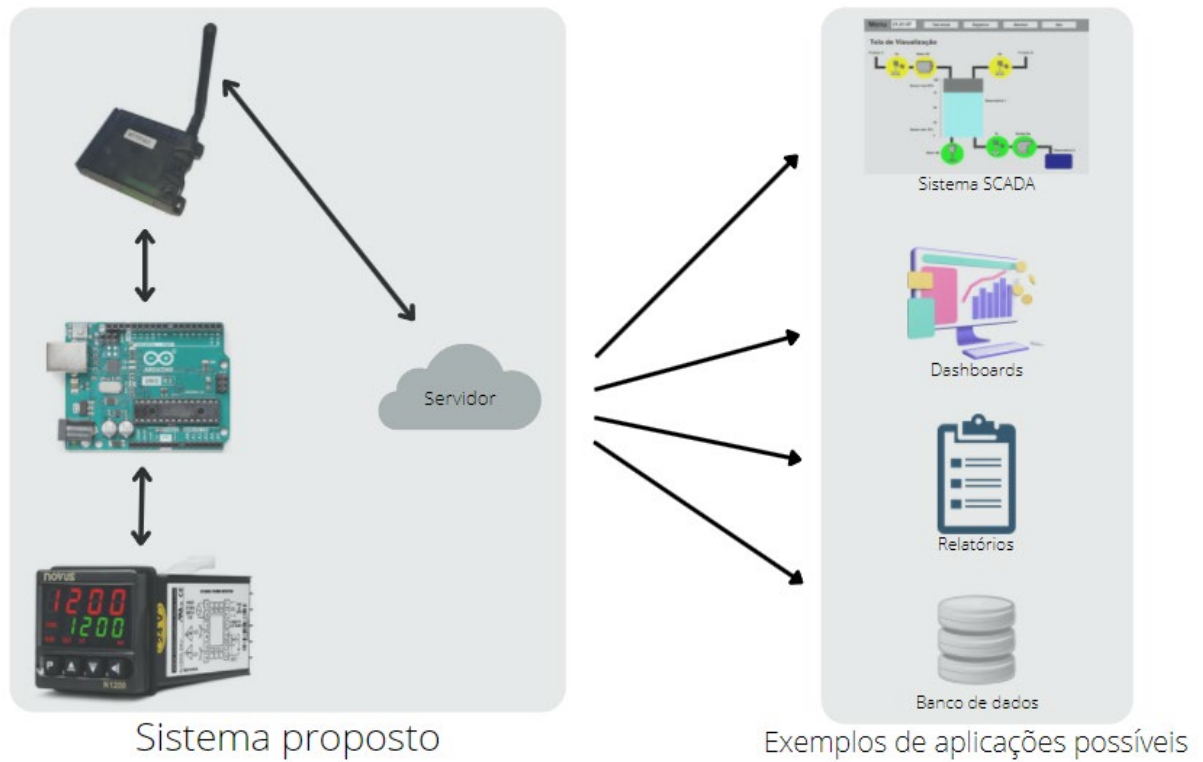
Como já explicado na metodologia do trabalho, o comando Incrementa aumenta um número na tela principal do controlador, enquanto o Ler valores retorna

o valor dos registrados que indicam o valor na tela principal, o número da tela atual e a versão do software do controlador.

Outro importante fator em sistemas IIoT é a questão da segurança. Por se tratar de um sistema destinado a dispositivos industriais, o acesso indevido de usuários a dados e a possibilidade de executar comandos, pode causar grandes riscos aos processos. Por esse fator, na comunicação MQTT foi utilizada a segurança por autenticação, desta forma é possível controlar e restringir o acesso para que cada mensagem seja acessada apenas por usuários autenticados.

Comparando o protótipo desenvolvido e os resultados obtidos com os produtos já disponíveis no mercado, podemos destacar o potencial que a solução possui de customização para ser adaptada a diferentes casos de uso. No infográfico da Figura 23 são apresentados alguns casos em que, através de implementações no *firmware* do kit de desenvolvimento, seria possível utilizar o sistema proposto. O sistema proposto pode ser aplicado em diferentes aplicações através de adaptações no *firmware*. O sistema pode ser utilizado para enviar dados de máquinas, sensores e controladores para sistemas SCADA, para dashboards, para relatórios de processos, para banco de dados, entre muitas outras aplicações, tudo isso utilizando redes móveis, sem as limitações impostas pelas redes físicas.

Figura 23 - Exemplos de aplicações possíveis.



Fonte – Elaborado pelo autor.

No capítulo a seguir, é apresentada a conclusão do trabalho, onde é feita uma recapitulação do objeto proposto e dos resultados obtidos. Além disso, são apresentadas algumas sugestões para trabalhos futuros envolvendo melhorias no sistema.

5. CONCLUSÃO

Com o avanço da internet das coisas no ramo industrial, a implementação nos próximos anos das redes de celular com tecnologia 5G, e as limitações que as redes cabeadas impõem nas redes industriais, fica clara a necessidade por equipamentos e soluções que permitam que dispositivos de campo se comuniquem com sistemas supervisórios e interliguem sistemas sem as restrições de distâncias e localidades existentes na atualidade.

Por mais que já existam soluções para comunicação destes dispositivos através de redes Wi-Fi (padrão IEEE 802.11), essa tecnologia de transmissão possui algumas características que podem complicar seu uso em ambientes industriais, como o custo para abranger uma grande área de cobertura, além das questões de interferência causadas pelas máquinas e equipamentos. Por este motivo, foi proposto neste trabalho o desenvolvimento de um dispositivo que atendesse essas necessidades, através do uso de redes móveis, que possuem grandes áreas de cobertura.

A implementação do sistema proposto, através de pesquisas e da elaboração da metodologia, se mostrou viável. Os resultados obtidos, atenderam de forma satisfatória os objetivos propostos. Se a implementação das redes 5G atenderem os requisitos de determinismo propostos nas especificações, o *gateway* terá possibilidades de aplicações de controle na indústria, controlando e operando processos de forma remota na nuvem.

O sistema proposto possui um grande potencial para ser explorado, onde através de modificações no *firmware* do kit de desenvolvimento, é possível adaptar a solução para diferentes aplicações em industriais de diferentes segmentos.

Como sugestão para trabalhos futuros, fica a sugestão de desenvolver a solução em um sistema embarcado, que possa ser inserido no mercado como um produto para comercialização. Outra sugestão é modificar o *firmware* para tornar o *gateway* transparente na comunicação, de forma que as requisições sejam enviadas do servidor para o dispositivo de campo sem codificações e identificação de comandos no *gateway*. Além disso, pode-se fazer melhorias no *firmware* para suportar outros protocolos além do Modbus RTU, e adicionar suporte a diferentes equipamentos, como por exemplo suporte a máquinas CNC.

REFERÊNCIAS

- Batalla, Jordi Mongay. 2020.** On Analyzing Video Transmission Over Wireless WiFi and 5G C-Band in Harsh IIoT Environments. 07 08, 2020, pp. 118534-118541.
- Comer, Douglas. 2016.** *Redes de computadores e internet*. Porto Alegre : Bookman, 2016.
- Filho, Guilherme Filippo. 2014.** *Automação de Processos e de Sistemas*. São Paulo : Érica, 2014.
- Fracaro, Janaina. 2017.** *Fabricação pelo processo de usinagem e meios de controle*. Curitiba : InterSaberes, 2017.
- HiveMQ. 2020.** *MQTT & MQTT 5 Essentials - A comprehensive overview of MQTT facts and features for beginners and experts alike*. Landshut : s.n., 2020.
- INDÚSTRIAS ROMI S.A.** *Manual de programação e operação - Linha ROMI D - CNC FANUC Oi-Mc*. São Paulo : s.n.
- . **2021.** ROMI. *ROMI*. [Online] 23 de 05 de 2021. https://www.romi.com/wp-content/uploads/2018/04/cat_romi_d_ng_po_ae_112019_baixa-1.pdf.
- Kehl, Pierre, et al. 2020.** Comparison of 5G Enabled Control Loops for Production. *IEEE 31st PIMRC'20 - Workshop on 5G Mobile Communication System for Smart Factories*. 2020.
- Liu, Shan, et al. 2020.** Research on 5G technology based on Internet of things. *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC 2020)*. 2020, pp. 1821-1823.
- Machine Design. 2016.** IIoT challenges and promises. *Gale Academic OneFile*. 7 2016, p. 20+.
- Modbus Organization. 2012.** MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1b3. 4 26, 2012.
- Modbus Organization, Inc. 2006.** MODBUS over Serial Line - Specification and Implementation Guide - V1.02. s.l. : Modbus.org, 12 20, 2006.
- Moraes, Alexandre Fernandes de. 2019.** *Redes industriais para automação industrial : AS-I, PROFIBUS e PROFINET*. São Paulo : Érica, 2019.

Morais, Izabelly Soares de, et al. 2018. *INTRODUÇÃO A BIG DATA E INTERNET DAS COISAS (IOT)*. Porto Alegre : SAGAH EDUCAÇÃO S.A, 2018.

Neto, Vicente Soares. 2018. *Telecomunicações avançadas e as tecnologias aplicadas*. São Paulo : Érica, 2018.

Plant Engineering. 2015. IIoT: Defining the terms. *Gale Academic OneFile*. Agosto 2015, p. 28.

Robotics Tomorrow. 2021. Retrofitting Older Machines Provides Valuable IIoT Update. *Robotics Tomorrow*. Disponível em <https://www.roboticstomorrow.com/story/2021/05/retrofitting-older-machines-provides-valuable-iiot-update/16870/>. Acesso em: 22 dec. 2021.

Silva, Sidnei Domingues da. 2008. *CNC: Programação de Comandos Numéricos Computadorizados: Torneamento*. São Paulo : Érica, 2008.

Sun, Changqing, et al. 2019. Design and Development of Modbus/MQTT Gateway for Industrial IoT Cloud Applications Using Raspberry Pi. 2019, pp. 2267-2271.

Vitturi, Stefano and Sauter, Thilo. 2019. Industrial Communication Systems and Their Future Challenges: Next-Generation Ethernet, IIoT, and 5G. *Proceedings of the IEEE*. 06 06, 2019, pp. 944-961.

Wu, Ho-Ting, et al. 2019. The Implementation of Wireless Industrial Internet of Things (IIoT) Based upon IEEE 802.15.4-2015 TSCH Access Mode . *IEEE Computer Society*. 2019, pp. 367-369.

Zanco, Wagner da Silva. 2008. *Microcontroladores PIC: técnicas de software e hardware para projetos de circuitos eletrônicos com base no PIC16F877A*. São Paulo : Érica, 2008.