



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
PROGRAMA DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

ELANO DIÓGENES GADELHA

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA AQUISIÇÃO DE DADOS DE
PRODUÇÃO INDUSTRIAL UTILIZANDO PROTOCOLO MODBUS RTU EM UMA
REDE RS-485 E INTERFACE GRÁFICA INTEGRADA AO GERENCIADOR DE
BANCO DE DADOS MYSQL**

FORTALEZA

2019

ELANO DIÓGENES GADELHA

DESENVOLVIMENTO DE UM PROTÓTIPO PARA AQUISIÇÃO DE DADOS DE
PRODUÇÃO INDUSTRIAL UTILIZANDO PROTOCOLO MODBUS RTU EM UMA
REDE RS-485 E INTERFACE GRÁFICA INTEGRADA AO GERENCIADOR DE BANCO
DE DADOS MYSQL

Monografia apresentada ao Curso de Engenharia Elétrica do Departamento de Engenharia Elétrica da Universidade Federal do Ceará, como requisito parcial para a obtenção do Título de Engenheiro Eletricista. Área de concentração: Redes Industriais, Automação de processos industriais.

Orientador: Prof. Dr. Paulo Peixoto Praça.

FORTALEZA

2019

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- G12d Gadelha, Elano.
Desenvolvimento de um protótipo para aquisição de dados de produção industrial utilizando protocolo MODBUS RTU em uma rede RS-485 e interface gráfica integrada ao gerenciador de banco de dados MYSQL / Elano Gadelha. – 2019.
50 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Centro de Tecnologia, Curso de Engenharia Elétrica, Fortaleza, 2019.
Orientação: Prof. Dr. Paulo Peixoto Praça.
1. Programa de gerenciamento de dados. 2. Arduino Uno. 3. Rede Rs-485. I. Título.

CDD 621.3

ELANO DIÓGENES GADELHA

DESENVOLVIMENTO DE UM PROTÓTIPO PARA AQUISIÇÃO DE DADOS DE
PRODUÇÃO INDUSTRIAL UTILIZANDO PROTOCOLO MODBUS RTU EM UMA
REDE RS-485 E INTERFACE GRÁFICA INTEGRADA AO GERENCIADOR DE BANCO
DE DADOS MYSQL

Monografia apresentada ao Curso de Engenharia Elétrica do Departamento de Engenharia Elétrica da Universidade Federal do Ceará, como requisito parcial para a obtenção do Título de Engenheiro Eletricista. Área de concentração: Redes Industriais, Automação de processos industriais.

Orientador: Prof. Dr. Paulo Peixoto Praça.

Aprovada em: 24/06/2019.

BANCA EXAMINADORA

Prof. Dr. Paulo Peixoto Praça (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Luiz Henrique Silva Colado Barreto
Universidade Federal do Ceará (UFC)

Prof. MSc. Lucas Silveira Melo
Universidade Federal do Ceará (UFC)

Dedico a Deus.

Aos meus pais, Francisco de Assis e Maria.

A minha esposa Raquel.

Em especial ao meu irmão Francisco Dino.

AGRADECIMENTOS

A Deus, por ter me dado força para a realização deste trabalho.

A minha esposa Raquel, que teve calma e vivenciou toda a trajetória, sempre me incentivando a seguir em frente.

Aos meus pais, Francisco de Assis e Maria, que me ajudaram acreditando em meu potencial e com suas orações.

Aos meus irmãos que muito me ensinaram a superar os desafios encontrados ao longo da vida.

Ao Prof. Dr. Paulo Peixoto Praça, pela excelente orientação e paciência, me direcionando com objetividade nas etapas necessárias de desenvolvimento do trabalho.

Aos professores participantes da banca examinadora Dr. Luiz Henrique Silva Colado Barreto e MSC. Lucas Silveira Melo pelo tempo, pelas valiosas colaborações e sugestões.

RESUMO

O presente trabalho apresenta o desenvolvimento de um protótipo de sistema de aquisição e gerenciamento de dados aplicado aos processos de produção industrial. Para o gerenciamento dos dados foi criado um programa na IDE de desenvolvimento Visual Studio .NET 2017 utilizando a linguagem C#, o programa é baseado no monitoramento de formulários e gráficos que possibilitam ao usuário análises mais rápidas e organizadas. Para suprir a demanda de informações necessitadas pelo software de gerenciamento foi preciso desenvolver placas remotas de aquisição de dados, que tem a placa Arduino Uno responsável por processar as rotinas Modbus RTU e também os eventos ocorridos nas máquinas. A conexão e as trocas de mensagens ocorridas entre o software de gerenciamento e as placas remotas de aquisição de dados foram atribuídas à rede física half-duplex RS-485 e ao protocolo Modbus RTU. Para finalizar foram discutidos alguns aspectos da montagem prática da aplicação e também expostos através do Software de Gerenciamento as análises dos resultados dos gráficos das tags de contagem, estado e analógica.

Palavras-chave: Programa de gerenciamento de dados. Sistema de aquisição de dados. Processo de produção.

ABSTRACT

The present project presents the development of a prototype of data acquisition and management system applied to industrial production processes. For data management a program was created in the IDE of Visual Studio .NET 2017 development using the C # language, the program is based on the monitoring of forms and graphs that allow the user to analyze faster and more organized. In order to supply the information demanded by the management software, it was necessary to develop remote data acquisition boards, which have the Arduino Uno board responsible for processing the Modbus RTU routines and also the events that occurred on the machines. The connection and message exchanges between the management software and the remote data acquisition boards have been assigned to the RS-485 half-duplex physical network and the Modbus RTU protocol. Finally, some aspects of the practical assembly of the application were discussed, as well as the analyzes of the results of the counting, state and analog tags.

Keywords: Data management program. Data Acquisition System. Production process.

LISTA DE FIGURAS

Figura 1 – Foto vista superior Placa Arduino Uno	18
Figura 2 – Processo de gravação do código escrito na IDE Arduino	19
Figura 3 – Periféricos ofertados pelo Hardware Arduino Uno	20
Figura 4 – Comunicação serial utilizando o modelo de transmissão diferencial	21
Figura 5 – Comunicação multiponto estabelecida pela rede RS-485	21
Figura 6 – Fluxograma com o mapa de desenvolvimento do projeto de aquisição e gerenciamento dos dados de produção industrial	26
Figura 7 - Módulo RTC DS3231	29
Figura 8 – Placa de Interfase de Sinais.....	30
Figura 9 – Estrutura da rede RS-485	31
Figura 10 – Módulo conversor TTL/RS-485	32
Figura 11 – Conexões multiponto estabelecida pela circuito integrado MAX485	33
Figura 12 – Módulo Conversor RS-485/USB	34
Figura 13 – Conversões internas realizadas pelo conversor RS-485/USB	34
Figura 14 – Instalação do drive para utilização do conversor RS-485/USB	35
Figura 15 – Dispositivo conversor pronto para ser utilizado	35
Figura 16 – Tela principal do Programa de Gerenciamento de Dados de Produção Industrial	36
Figura 17 – Painel de Comunicação realiza o monitoramento das transferências de mensagens	38
Figura 18 – Exemplo detalhado da comunicação Modbus RTU.....	39
Figura 19 – Tela de Cadastro de Equipamentos e Tags	40
Figura 20 – Gráfico gerado através de uma consulta de produção diária	42
Figura 21 – Gráfico de comparações entre equipamentos em produção diária	43
Figura 22 – Gráfico de análise produção mensal	44

Figura 23 – Gráfico de comparação entre equipamentos, produção mensal	44
Figura 24 – Gráfico de Estado analisa a porcentagem de disponibilidade do: equipamentos	45
Figura 25 – Consulta horária de dados Analógicos	46
Figura 26 – Gráfico montado com os dados recolhidos da tabela de dados analógicos	47
Figura 27 – Montagem prática da estação remota de aquisição de dados.....	48
Figura 28 – Tabela de custos de uma placa de aquisição e estabelecimento da rede RS- 485	49
Figura 29 – Comunicação RS-485 estabelecida, utilizando três escravos em rede.....	50

LISTA DE TABELAS

Tabela 1 – Mensagem enviada pelo dispositivo mestre da rede Modbus RTU	23
Tabela 2 – Mensagem enviada pelo dispositivo escravo da rede Modbus RTU	23
Tabela 3 – Campos de Funções Modbus RTU	24

LISTA DE ABREVIATURAS E SIGLAS

ASCII	American Standard Code for Information Interchange
C#	CSharp
CI	Circuito Integrado
CRC	Cyclic Redundancy Check
F#	FSharp
IDE	Integrated Development Environment
IHM	Interface Homem-Máquina
I ² C	Inter-Integrated Circuit
PLC	Programmable Logic Controller
RTC	Real Time Clock
RTU	Remote Terminal Unit
RS-485	(Recommended Standard) -485
SCL	Serial Clock
SDA	Serial Data
SQL	Structured Query Language
TTL	Transistor-Transistor Logic
USB	Universal Serial Bus
UART	Transmissor Receptor Assíncrono Universal
VB .NET	Visual Basic .NET

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Justificativa	15
1.2	Objetivo	15
1.3	Organização do trabalho	16
2	FUNDAMENTAÇÃO TEÓRICA	17
2.1	IDE Visual Studio .NET 2017	17
2.2	O MySQL	17
2.3	O Arduino Uno	18
2.4	Estrutura de Rede RS-485	20
2.5	Protocolo Modbus RTU	22
2.5.1	<i>Mensagem Protocolo Modbus RTU</i>	22
2.5.1.1	<i>Estrutura da Mensagem Modbus RTU</i>	23
2.5.2	<i>Campos da Mensagem Modbus RTU</i>	24
2.5.2.1	<i>Campo de Endereço da Mensagem</i>	24
2.5.2.2	<i>Campo de Função da Mensagem</i>	24
2.5.2.3	<i>Campo de Dados da Mensagem</i>	25
2.5.2.4	<i>Campo de Validação da Mensagem</i>	25
3	DESENVOLVIMENTO DO PROJETO	26
3.1	Estação Remota de Aquisição de Dados (Escravo Modbus RTU)	27
3.1.1	<i>Configurações Realizadas no Arduino</i>	27
3.1.1.1	<i>Configuração do Modbus RTU no Arduino</i>	28
3.1.2	<i>Módulo RTC DS3231</i>	29
3.1.3	<i>Interface de aquisição de dados</i>	29
3.2	Rede RS-485	31
3.2.1	<i>Módulo Conversor TTL/RS-485</i>	32
3.2.2	<i>Módulo Conversor RS-485/USB</i>	34
3.3	Mestre Rede Modbus RTU	36
3.3.1	<i>Tela Principal do Programa de Gerenciamento de Dados</i>	36
3.3.2	<i>Tela de Comunicação Modbus RTU</i>	38
3.3.3	<i>Tela de manipulação das variáveis dos processos industriais</i>	39
4	RESULTADOS E DISCUSSÕES	41

4.1	Acessando os Históricos de Produção Industrial	41
4.1.1	<i>Tela de Análise de Quantidade de Produtos.....</i>	41
4.1.2	<i>Tela de Análise de Estado.....</i>	45
4.1.3	<i>Tela de Análise Analógica.....</i>	46
4.2	Monitoramentos em tempo real	47
4.3	Montagem do Sistema Remoto de Aquisição de Dados.....	47
4.3	Análise de custos dos equipamentos utilizados.....	48
5	CONCLUSÃO.....	50
5.1	Trabalhos Futuros.....	51
	REFERÊNCIAS	52

1 INTRODUÇÃO

A utilização do computador apenas para aquisição de dados de processos foi a porta de entrada para as técnicas de controles aplicadas atualmente, ainda hoje a captação de dados somente para consulta é a atividade mais presente no meio industrial (RIBEIRO, 2001). O presente trabalho analisa o desenvolvimento de um programa de gerenciamento que possui a função de analisar as informações de produção industrial, os dados de produção das máquinas são enviados ao software através das estações remotas conectadas a uma rede de comunicação situada no chão de fábrica.

A supervisão dos processos industriais eleva a qualidade de análise das informações, possibilitando ao operador prover soluções aos desafios propostos em menor tempo, além de proporcionar a centralização dos dados (ROGGIA, FUENTES, 2016). A possibilidade de coletar as informações em apenas um ponto de análise, nem sempre requer uma tarefa simples, os equipamentos de controle das máquinas apresentam fabricantes diferentes e com isso protocolos de comunicação diferenciados (COELHO, 2010). A consequente falta de comunicação entre os dispositivos presentes no chão de fábrica levou ao desenvolvimento das estações remotas de aquisição de dados trabalharem em conjunto aos dispositivos de controle das máquinas. As estações remotas apresentam apenas funções de coleta de dados e disponibilizam as informações na rede de comunicação quando recebem solicitações do programa supervisor.

As redes industriais de comunicação possuem como característica principal a garantia na interligação de todos os sistemas produtivos, possibilitando maior disposição de informações ao elemento supervisor (BAYER, ECKHARDT, MACHADO, 2011). Os dispositivos de aquisição dos dados desenvolvidos neste trabalho possuem suas comunicações estabelecidas no barramento de rede RS-485, esta apresenta elevada robustez e resistência aos ruídos. O padrão RS-485 foi desenvolvida em 1983 com o objetivo de realizar comunicação multiponto, tentando elevar à confiabilidade na troca de mensagens, as indústrias utilizam juntamente a rede RS-485 o protocolo de comunicação MODBUS em suas aplicações (BARBOSA, 2014).

O protocolo MODBUS estabelecido em rede RS-485 pode operar em modo de transmissão ASCII ou RTU, estes muito utilizados no setor industrial pelos diversos fabricantes de equipamentos (MANUAL DE COMUNICAÇÃO SERIAL RS232/RS485, 2008). O projeto aborda o uso do protocolo MODBUS RTU comunicando todos os dispositivos em uma estrutura de hierarquia mestre/escravo. O computador representa o

mestre da rede MODBUS RTU e tem como tarefa solicitar respostas das estações remotas que se comportam como escravas da rede.

1.1 Justificativa

O projeto foi desenvolvido procurando atender a demanda de indústrias de pequeno e médio porte que almejam com organização na gestão de informações de produção, mas encontram barreiras nos elevados custos para adquirir equipamentos que realizem a aquisição e supervisão das informações ofertadas pelas linhas de produção.

A confiabilidade foi outro fator bastante relevante para a implantação de um sistema automatizado de aquisição de dados, nas indústrias em desenvolvimento, as informações de produção normalmente são inseridas manualmente em tabelas de papéis, sendo depois analisados e finalmente anexados aos sistemas de banco de dados. Este método de gestão apresenta muitas consequências ruins como: a demora na análise das informações, a fraude no preenchimento dos dados, requer tempo para a reunião de todas as informações e entre outros fatores que impactam diretamente na eficiência e tomadas de decisões dos processos industriais.

A aplicação da informatização nos sistemas de aquisição de dados providencia maior flexibilidade do setor de manufatura, rapidez das estratégias executadas e uma motivação maior por parte do grupo que constitui a organização. Com a automação da coleta dos dados de produção fica possível o engajamento não só dos setores de gestão mais elevada, mas também de operadores que estão analisando indiretamente suas participações nos resultados.

1.2 Objetivo

O desenvolvimento do projeto busca resolver alguns problemas que são muito comuns em chão de fábrica, dentre eles estão a indisponibilidade de dados, a desorganização das informações além das análises e tomadas de decisões demoradas e equivocadas. Para promover as soluções aos problemas citados foi tomada a tarefa de automatizar as coletas de informações dos processos produtivos.

Para atender as aquisições dos dados foi proposto como objetivo a criação de uma plataforma de hardware e software de baixo custo financeiro. A aplicação dessa plataforma

visa monitorar e também registrar as informações de produção e com isso acompanhar remotamente os processos possibilitando tomadas de decisões mais rápidas e eficazes.

1.3 Organização do trabalho

O capítulo 1 exhibe a importância de utilizar programas responsáveis de supervisionar e gerenciar as informações geradas pelos processos industriais. O presente capítulo também identifica os componentes que irão compor as estações remotas e a rede de conexão por onde ocorrerá o tráfego de dados.

O capítulo 2 explica as ferramentas que tornaram possível a criação do projeto, dentre elas estão A IDE Visual Studio .NET 2017 que possibilitou o desenvolvimento do software de gerenciamento, a placa Arduino que processa os dados coletados e os disponibilizam na rede, o protocolo Modbus RTU que é responsável por levar confiabilidade na trocas de informações entre os dispositivos mestre e escravo e por fim a rede RS-485 que conectam fisicamente os dispositivo mestre e os dispositivos escravos.

O capítulo 3 é responsável por detalhar as etapas de desenvolvimento do projeto ou seja como foi organizada cada etapa e quais os equipamentos utilizados em suas constituições. Neste capítulo serão mostradas as características do software de gerenciamento, os módulos conversores que compõem a rede RS-485 e os dispositivos que formam as placas remotas de aquisição de dados.

O capítulo 4 consiste nas discussões dos resultados de análises gráficas baseadas nas tags de contagens, estados e analógica cadastradas no programa de gerenciamento é também detalhada as montagens práticas da estação remota de aquisição de dados assim como o custo aproximado dos equipamentos utilizados no projeto.

O capítulo 5 apresenta a conclusão dos objetivos alcançados pelo projeto e também levanta algumas possibilidades de projetos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 IDE Visual Studio .NET 2017

A IDE Visual Studio .NET, é uma ferramenta de desenvolvimento criada pela Microsoft que garante maior facilidade ao programador na implementação dos elementos gráficos, isso leva o usuário a interagir de modo mais produtivo com os aplicativos desenvolvidos (DEITEL, 2002). A IDE permite a criação de programas em várias linguagens .NET são: C#, VB.NET, F# e Delphi Prism, dentre estas será abordada neste trabalho a linguagem C# que utiliza o conceito de programação orientada a objeto.

A linguagem C# é completamente orientada a objetos, esta apresenta elevada simplicidade em sua estrutura o que facilita posteriores manutenções em seus códigos (LIMA, 2002). A quantidade de recursos oferecidos pela linguagem C# permite ao desenvolvedor apresentar aplicativos com melhores adequações às necessidades solicitadas pelo cliente.

Utilizando as bibliotecas ofertada pela própria IDE VISUAL STUDIO .NET e a biblioteca MySqlConnection é possível a criação de programas supervisórios capazes de realizar o armazenamento e processamento das variáveis trabalhadas no processo abordado, as operação de manipulação de banco de dados, amostragens gráficas bem definidas e interativas, configurações de portas seriais de comunicação e de protocolos de redes, e dentre outras funções que não serão abordadas neste trabalho.

2.2 O MySQL

O MySQL é um sistema de gerenciamento de bancos de dados relacional, este utiliza como linguagem o padrão SQL e através deste são realizadas operações de manipulações e consultas aos dados alocados em bancos relacionais (NIEDERAUER, PRATES, 2006). O Servidor MySQL é uma ferramenta que possibilita visualizações mais intuitivas do banco de dados, além de realizar processamentos mais rápidos e eficazes (AXMARK, 2003).

No desenvolvimento dos códigos foi utilizado a ferramenta de gerenciamento Workbench MySQL que apresenta interfaces visuais bastante amigáveis, permitindo melhor entendimento do desenvolvedor nas elaborações dos códigos SQL e também melhor visualização da estrutura relacional.

Os bancos de dados desenvolvidos em MySQL podem ser acessados por uma aplicação em C# através da instalação de drives que estabelecem as conexões respeitando as condições de confiabilidade necessárias para ocorrer as operações de manipulação dos dados.

2.3 O Arduino Uno

O Arduino é uma placa de sistema embarcado capaz de executar tarefas de processamento de sinais analógicos e digitais através de suas entradas e saídas, possibilitando a melhor interpretação do ambiente analisado (MCROBERTS, 2011). O Arduino apresenta estrutura de Hardware e Software bem intuitivos aos seus usuários, a facilidade de manuseio difunde cada vez mais a sua aplicação nas mais diferentes áreas de abordagens (SANTOS, 2009).

Figura 1 – Foto vista superior Placa Arduino UNO



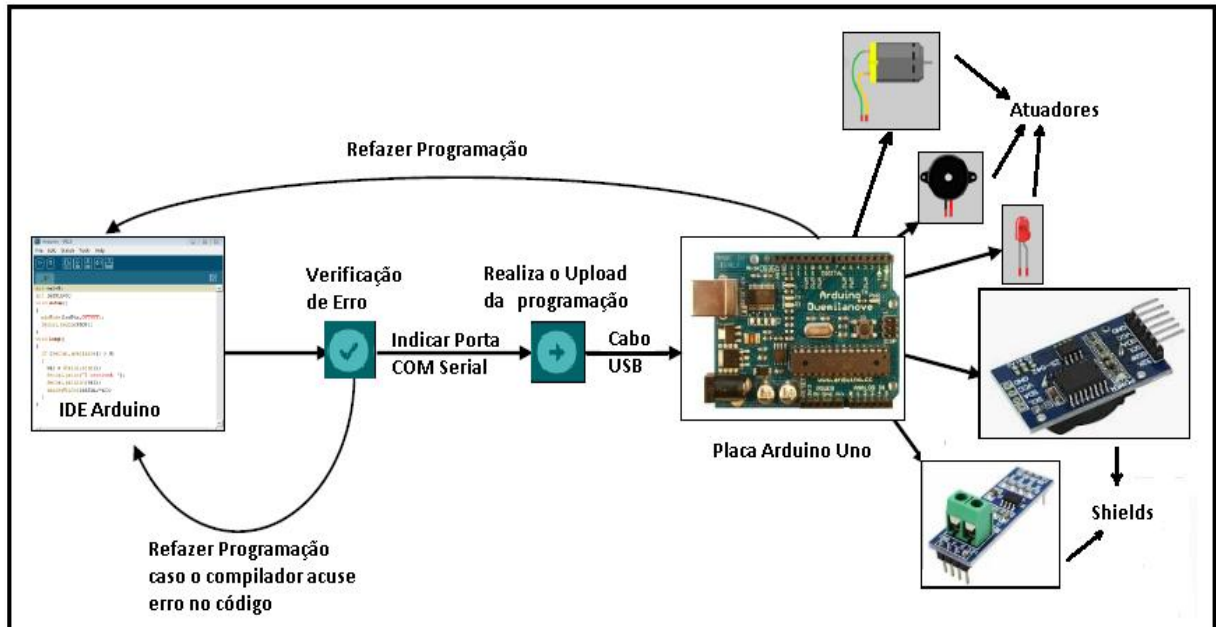
Fonte: elaborada pelo autor

A placa Arduino apresenta diversos modelos para utilização, cabe ao usuário escolher o modelo mais adequado às necessidades imposta pela aplicação. A Figura 1 apresenta a placa Arduino UNO, que possui como principal componente o chip ATMEGA328P responsável pelas execuções das rotinas de código salvas em sua memória Flash de 32 kbytes (Santos, 2009).

As programações de software inseridas no Arduino UNO são produzidas em linguagem C, para isso é utilizada a IDE de desenvolvimento Arduino. O ambiente de

desenvolvimento do software apresenta várias opções de código de exemplo que auxiliam no aprendizado inicial além de disponibilizar várias bibliotecas que possibilita ao usuário trabalhar com os mais diversos atuadores e Shields.

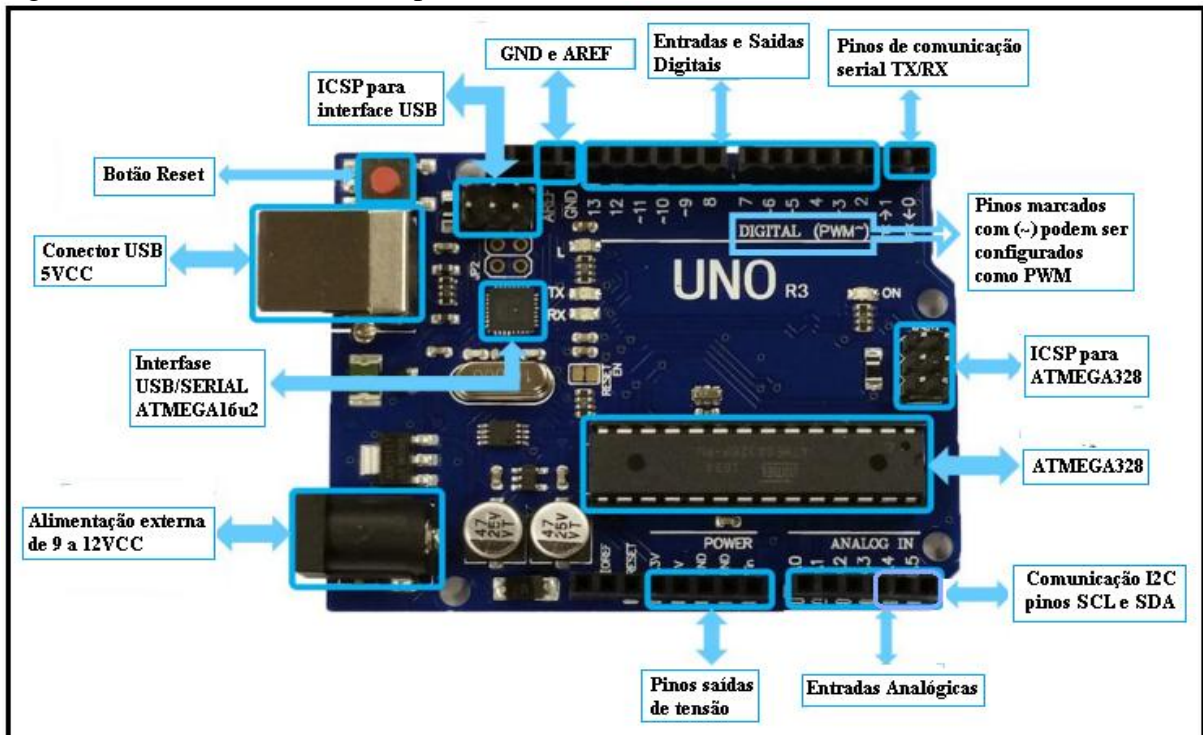
Figura 2 – Processo de gravação do código escrito na IDE Arduino



Fonte: adaptado de Santos (2009).

É possível identificar o passo a passo da gravação de um código na placa Arduino Uno. Com a IDE aberta e código pronto primeiro é acionada a verificação de erro, se tudo estiver correto, será indicada a porta COM serial utilizada pelo Arduino Uno e então realiza a operação de *upload* para a gravação na placa.

Figura 3 – Periféricos ofertados pelo Hardware Arduino Uno



Fonte: elaborada pelo autor.

A grande oferta de periféricos mostrados na Figura 3 foi a principal característica apresentada para a integração ao projeto, a placa também oferece conectores de fácil acesso e manuseio. Outro fator importante é a grande quantidade de equipamentos Shields, totalmente compatíveis com as plataformas tanto de hardware quanto de software existentes no mercado.

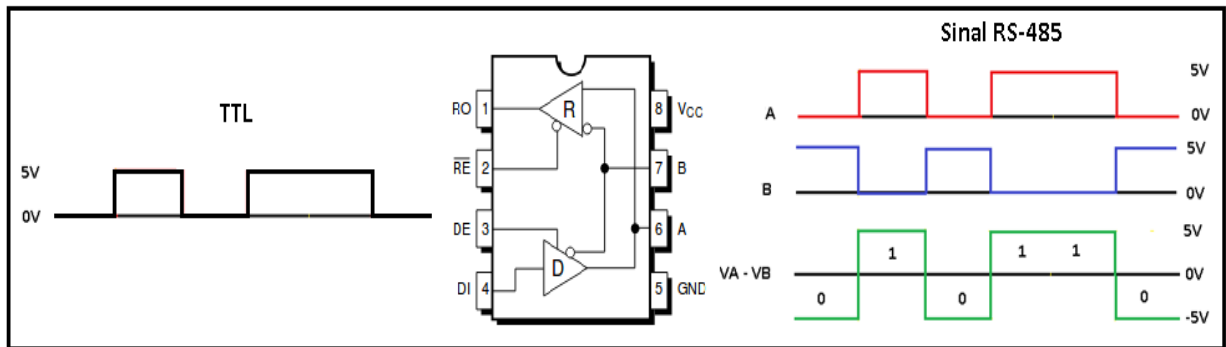
A característica que concede ao Arduino a fama de fácil aplicabilidade, são as quantidades de materiais com exemplos e bibliotecas disponíveis para download em várias páginas da Internet. Os códigos são de fáceis interpretações, solicitando ao usuário apenas algumas modificações caso este esteja utilizando modelos diferenciados de placas Arduinos.

2.4 Estrutura de Rede RS-485

Os meios físicos de transmissão de dados são normalmente escolhidos de acordo com as aplicações e os requisitos de propagação dos sinais (ALBINI, 2019). A necessidade de atender aos desafios de velocidade, resistências a ruídos, quantidade de equipamentos conectados a mesma estrutura de comunicação e dentre outros fatores, possibilitam dia-a-dia a evolução dos dispositivos de comunicação. O desenvolvimento do padrão de rede RS-485 conseguiu englobar os requisitos necessários para ser bastante utilizado nas aplicações do meio industrial.

O ambiente industrial possui elevada incidência de ruídos eletromagnéticos, que são consequência da grande quantidade de máquinas em funcionamento. A Rede RS-485 apresenta a capacidade de atenuação dos ruídos eletromagnéticos, graças ao modelo diferencial de transmissão de dados, esta também consegue conectar vários equipamentos em um único barramento possibilitando a centralização dos dados (PADRÃO RS-485, 2019).

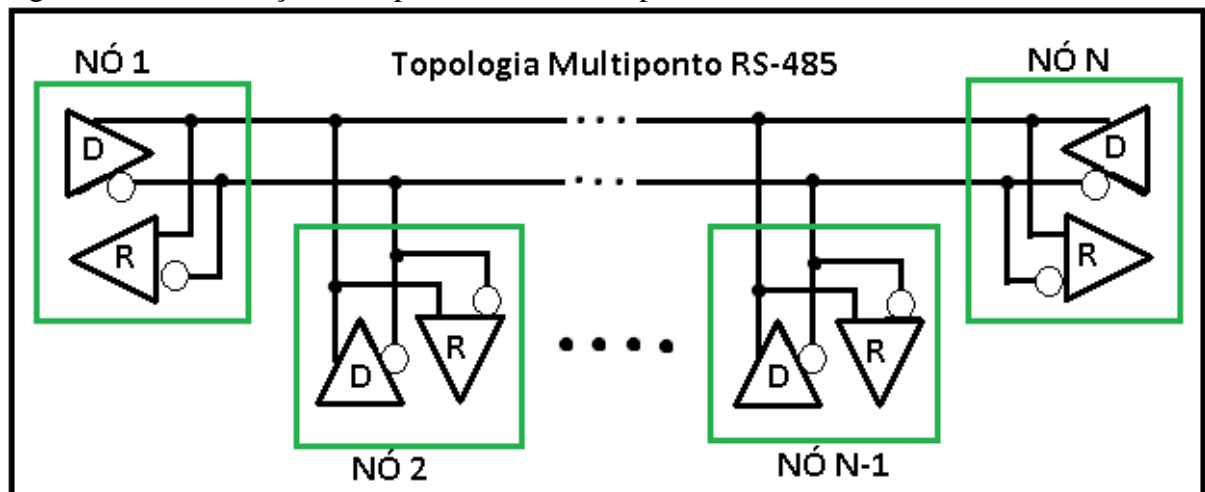
Figura 4 – Comunicação serial utilizando o modelo de transmissão diferencial



Fonte: elaborada pelo autor

O modo de funcionamento diferencial de transmissão mostrado na Figura 4 se dá pela utilização de apenas dois fios que são denominados por A e B, esses determinam os níveis lógicos da rede de acordo com suas polaridades (BARBOSA, 2014). Essa característica de operação da rede permite que as transmissões e recepções sejam resistentes aos ruídos eletromagnéticos (PADRÃO RS-485, 2019).

Figura 5 – Comunicação multiponto estabelecida pela rede RS-485



Fonte: elaborada pelo autor.

Outra capacidade da rede RS-485 é a possibilidade de estabelecer conexões multipontos como mostrada na Figura 5, através desse artifício é possível realizar a comunicação entre vários equipamentos e com isso possibilitar a centralização dos dados em um único dispositivo.

O padrão RS-485 não aconselha utilizar nenhum protocolo desenvolvido em software, mas o setor industrial normalmente utiliza o protocolo Modbus RTU que além de fácil entendimento e aplicação este apresenta plataforma open source.

2.5 Protocolo Modbus RTU

Protocolo de rede de comunicação é o conjunto de regras estruturadas em algoritmos que permite a garantia da ocorrência das trocas de mensagens e a correta execução dos procedimentos entre os equipamentos que constituem a rede (RIOS, 2012).

As indústrias tentando automatizar e gerenciar seus processos, de forma a produzir com melhor eficiência, resolveu investir em equipamentos que conseguisse atuar em rede, então muitos protocolos de redes foram desenvolvidos para atender as mais diversas aplicações industriais. Dentre os protocolos de redes existentes será abordado nesse trabalho o protocolo Modbus RTU que utiliza o arranjo mestre/escravo em suas aplicações.

O Protocolo Modbus RTU utiliza comunicação serial assíncrona e necessita de apenas um dispositivo mestre e pelo menos um dispositivo escravo para que ocorra o estabelecimento das trocas de informações e o estabelecimento da rede. O equipamento determinado a operar como mestre da rede sempre inicia e finaliza as solicitações de informações aos escravos, o mestre também é responsável pela escolha de qual escravo deseja realizar comunicação.

Os ajustes dos parâmetros seriais para o envio dos dados precisam ser realizados de modo que todos os equipamentos pertencentes à rede obedeçam à mesma configuração. Os parâmetros seriais são: quantidade bits de início da mensagem (*start bit*), velocidade de transmissão de dados (*baud rate*), checagem de paridade de bits e bits de parada (*stop bits*).

2.5.1 Mensagem Protocolo Modbus RTU

As trocas de informações entre os dispositivos são definidas por estruturas de mensagens compostas por bytes, no corpo das mensagens estão o endereçamento do

dispositivo, a função desempenhada, os parâmetros manipulados pela função e por fim a verificação de erros (PROTOCOLO DE COMUNICAÇÃO MODBUS RTU/ASCII, 2000).

2.5.1.1 Estrutura da Mensagem Modbus RTU

No modo Modbus RTU as indicações de início e fim da mensagem é realizado por um tempo de não utilização do barramento de comunicação. O tempo é estimado em 3,5 vezes a palavra de dados, este pode ser calculado como mostrado abaixo:

- Se a taxa de transmissão adotada pelos membros da rede for de 9600 bps.
- Obedecendo a configuração de 1 bit de início, 8 bits de dados, sem bits de paridade e 1 bit de parada, tem-se uma palavra com 10 bits.

Então o tempo total para envio de uma palavra será de 1,042 ms ($10 \times (1/9600)$), com isso a identificação de término ou início da próxima mensagem terá 3,65 ms ($3,5 \times 1,042$) sem utilização do barramento. Toda a mensagem necessita deve ser enviada de maneira contínua sem interrupções de tempo entre as palavras.

A mensagem Modbus RTU mestre apresenta as seguintes características apresentadas na Tabela 1:

Tabela 1 – Mensagem enviada pelo dispositivo mestre da rede Modbus RTU

Início da Mensagem	Endereço do escravo	Função a ser desempenhada	Início de endereços manipulados	Total de endereços manipulados	Validação da mensagem		Fim de Mensagem
Tempo Início	1 byte	1 byte	2 bytes	2 bytes	CRC- 1 byte	CRC+ 1 byte	Tempo fim

Fonte: adaptado de Protocolo de Comunicação RTU/ASCII (2000)

O dispositivo Modbus RTU escravo responde ao equipamento mestre através da mensagem apresentada na Tabela 2.

Tabela 2 – Mensagem enviada pelo dispositivo escravo da rede Modbus RTU

Início da Mensagem	Endereço do escravo	Função que foi desempenhada	Quantidade de bytes retornados	Dados que foram solicitados pelo Mestre	Validação da mensagem		Fim da Mensagem
Tempo Início	1 byte	1 byte	2 bytes	N x byte	CRC- 1 byte	CRC+ 1 byte	Tempo fim

Fonte: adaptado de Protocolo de Comunicação RTU/ASCII (2000)

2.5.2 Campos da Mensagem Modbus RTU

A mensagem Modbus RTU está estruturada em campos, estes atuam de modo ordenado com a finalidade de estabelecer os procedimentos necessários para que as trocas de informações consigam ocorrer corretamente entre os dispositivos mestre e escravos. A mensagem é dividida pelos campos de endereço, função, dados e validação de erro. Nas seções seguintes será abordado cada campo de acordo com as suas funções.

2.5.2.1 Campo de Endereço da Mensagem

O campo de endereço é constituído de apenas um byte como indicado nas Tabelas 1 e 2, através desse campo torna-se possível identificar o dispositivo escravo que o mestre quer solicitar informações. Caso o byte de endereço da mensagem enviada pelo mestre seja igual ao byte do endereço escravo, a mensagem é recebida e analisada. Após o processamento efetuado pelo dispositivo escravo confirmar que a mensagem está correta, o endereço do escravo é anexado na mensagem de resposta e enviado para o mestre.

2.5.2.2 Campo da Função da Mensagem

O campo função da mensagem é responsável por indicar ao dispositivo escravo qual ação será executada, após realizar a ação desejada corretamente o número da função será inserida e enviada ao mestre. Este campo possui em sua composição 1 byte, a faixa de valores pode ser tomadas de 0x1H a 0xFFH, mas existem funções que são mais utilizadas e normalmente são as que estão disponíveis para utilização nos equipamentos, estas podem ser visualizadas na Tabela 3:

Tabela 3 – Campo de Funções Modbus RTU

Código da Função em Hexadecimal	Nome	Ação
01	Read Coils	Leitura das saídas digitais
02	Read Discrete Inputs	Leitura das entradas digitais
03	Read Holding Registers	Leitura de Registradores tipo holding
04	Read Input Registers	Leitura de Registrador entrada analógica
05	Write Single Coil	Escrita de único bit na bobina
06	Write Single Register	Escrita em um único Registrador holding
15	Write Multiple Coils	Escrita em múltipla saídas digitais
16	Write Multiple Registers	Escrita em Registradores tipo holding

Fonte: Manual de Comunicação Serial RS232/RS485

2.5.2.3 Campo de Dados da Mensagem

Os campos de dados apresentam duas configurações distintas, caso a mensagem seja enviada pelo mestre, este campo deverá conter respectivamente as informações de início de endereço manipulado e quantos endereços serão manipulados, os endereços podem ser representados por entradas e saídas de bobinas ou registradores, esta característica está ilustrada na Tabela 1.

Quando a mensagem for enviada pelo dispositivo escravo o campo de dados conterá respectivamente as informações da quantidade de dados enviados e os dados enviados, a composição desse campo está ilustrada na Tabela 2.

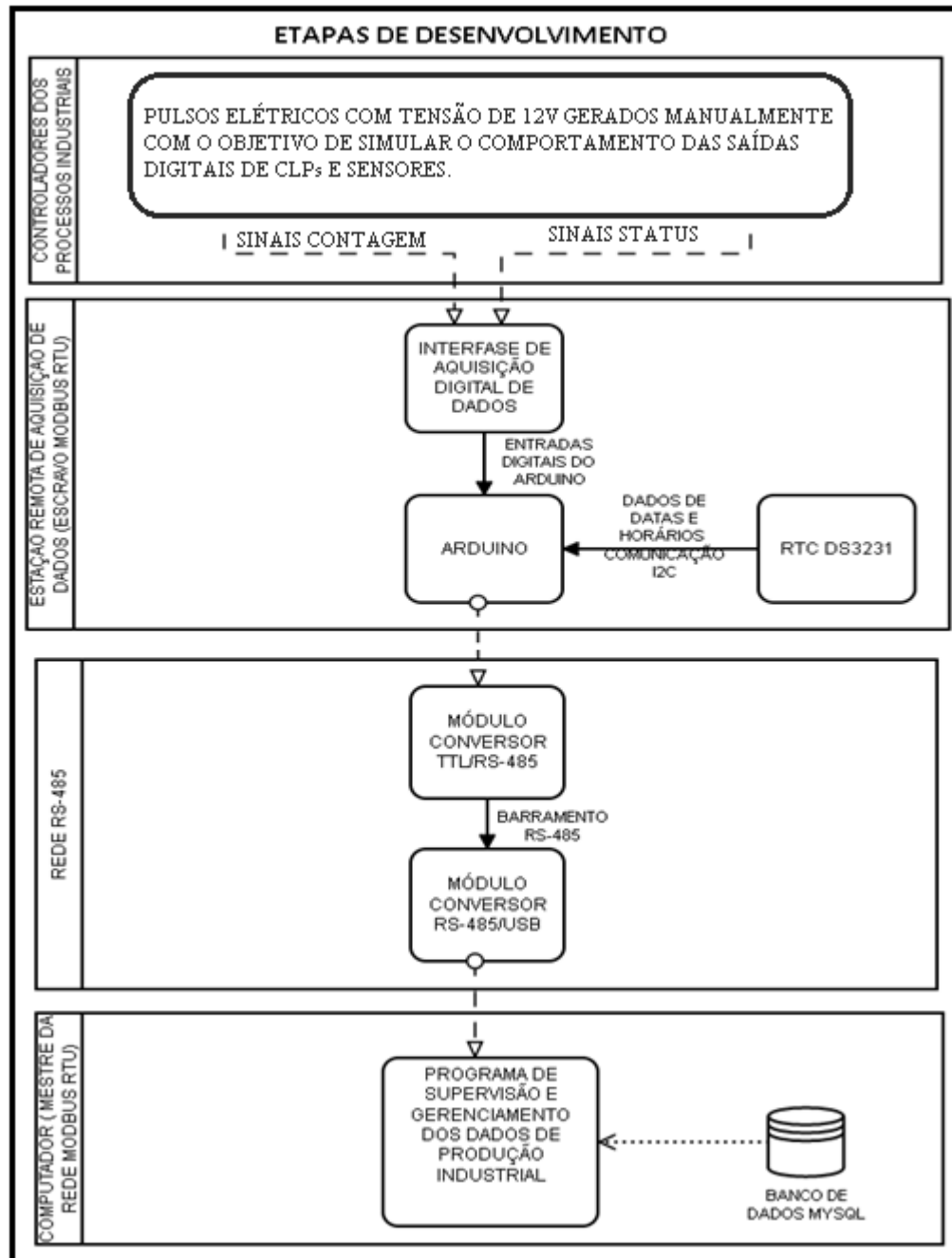
2.5.2.4 Campo de validação da mensagem

Para a mensagem ser recebida pelos dispositivos mestres e escravos, esta precisa primeiramente ser validada, o campo de validação da mensagem Modbus RTU utiliza o método CRC para efetuar as verificações de erro, este método calcula todo o conteúdo da mensagem e apresenta como resultado um valor de dois bytes que são conectados a mensagem a ser transmitida, a ordem de integração segue com byte menos significativo enviado antes do byte mais significativo, como ilustrado nas Tabelas 1 e 2.

3 DESENVOLVIMENTO DO PROJETO

O projeto de aquisição e gerenciamento dos dados de produção industrial foi estruturado de acordo com as etapas de desenvolvimento apresentadas na Figura 6.

Figura 6 – Fluxograma com o mapa de desenvolvimento do projeto de aquisição e gerenciamento dos dados de produção industrial.



Fonte: elaborada pelo autor.

3.1 Estação Remota de Aquisição de Dados (Escravo Modbus RTU)

O Arduino fica responsável pelo processamento das rotinas Modbus RTU, ele trabalha em conjunto com o Módulo RTC DS3231 e a interface de aquisição digital para a obtenção dos dados de data e horários e dos dados de contagem e de estado ofertados pelos dispositivos controladores dos processos industriais.

3.1.1 Configurações Realizadas no Arduino

O *software* foi desenvolvido na própria IDE do Arduino discutida na seção 2.3, a programação procura atender as demandas de processamento, recepção e transmissão de dados solicitada pela comunicação Modbus RTU. A construção do código foi baseada na utilização de bibliotecas oferecidas pela própria IDE do Arduino e de outras bibliotecas obtidas por meio de pesquisas na Internet.

Os pinos 3, 4, 5 e 6 do Arduino UNO são habilitados a trabalharem como entradas digitais, sendo os dois primeiros reservados para os processamentos de dados de contagem e os outros restantes para estado de disponibilidade de funcionamento dos equipamentos. A variável contadora sofre incremento sempre que ocorre um evento de borda de subida no pino analisado, ou seja, quando o estado da entrada digital passa de nível lógico baixo (0V a 0,8V) para nível lógico alto (2,8V a 5V). As entradas 5 e 6 informam quando o equipamento está ligado em nível lógico alto ou desligado em nível baixo permitindo assim a identificação correta da disponibilidade dos equipamentos.

A comunicação serial e gerenciamento da rede RS-485 foram estabelecidas sobre a configuração dos pinos digitais 0, 1 e 2. O pino 0 foi habilitado como entrada digital para servir de receptor (RX) serial, o pino 1 desempenha a função de transmissor (TX) serial, com isso foi ajustado como saída digital e finalmente o pino 2, este foi configurado como saída, ele controla a entrada e saída de dados no módulo de conversão TTL/RS485.

As informações de datas (dia/mês/ano) e horários (hora:minuto:segundo) de ocorrência dos eventos tornam-se possíveis graças a inclusão da biblioteca DS3231, esta responsável pelo armazenamento e execução das rotinas de controle efetuadas no dispositivo RTC DS3231 que está conectado via barramento I2C com os pinos SDA (A4) e SCL (A5).

O pino A0 foi ajustado para realizar a operação de entrada analógica, o intervalo de tensão de 0V a 5V onde é transformado internamente ao circuito integrado ATMEGA328P

que possui conversor analógico/digital de 10 bits, o valor analógico é então adequado no intervalo de 0 a 1023, assim fica possível quantificar a grandeza analógica analisada.

3.1.1.1 Configuração do Modbus RTU no Arduino

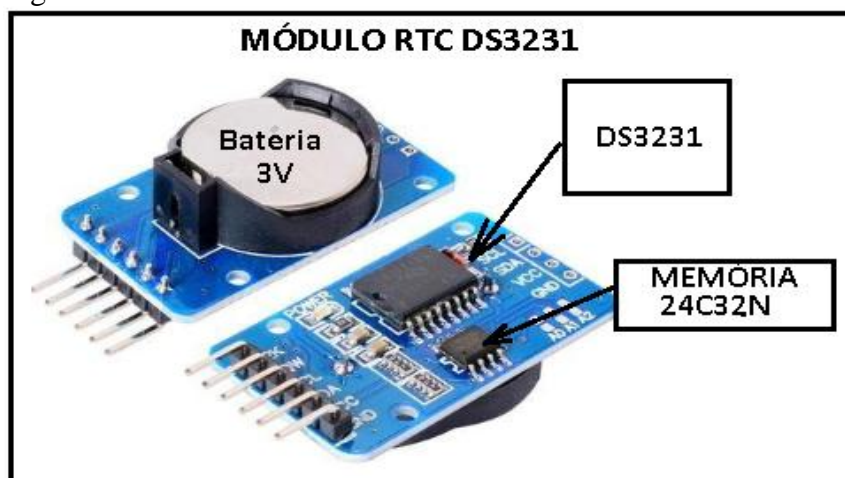
Todas as recepções e transmissões de dados efetuados pelo Arduino são processadas através do protocolo Modbus RTU escravo, as rotinas presentes no protocolo são encarregadas de oferecer maior confiabilidade nas trocas de informações solicitadas pelo dispositivo Mestre da rede.

As configurações utilizadas no Arduino Uno para o desenvolvimento do protocolo Modbus RTU escravo foram realizadas pelas bibliotecas *Modbus* (GitHub, disponível em: github.com/jossef/arduino-modbus-slave. Acesso em: 30 junho 2019). Elas trabalham em conjunto na execução das rotinas, dentre elas a *modbusSlave.h* foi a única que necessitou de alterações no código fonte, a modificação foi necessária para que o Arduino conseguisse controlar o Módulo Conversor TTL/RS-485 e estabelecer a comunicação em rede RS-485.

A mensagem Modbus RTU foi estabelecida de acordo com a estrutura exposta na seção 2.5.1.1 deste trabalho. Foram adotadas algumas condições de operação para o Arduino, procurando tornar a aplicação mais prática e intuitiva. Como o projeto desenvolvido tem a capacidade apenas de leitura de dados, foi utilizada apenas a função 0x03 que realiza a leitura de registradores, assim estarão no campo de dados das mensagens trocadas entre os dispositivos mestre e escravos às informações dos valores de contagem de eventos, dos estados dos dispositivos controladores de processos e dos valores analógicos. As mensagens recebidas com erro pelo dispositivo escravo são totalmente descartadas e não será retornada nenhuma informação de erro ao dispositivo Mestre.

3.1.2 Módulo RTC DS3231

Figura 7 – Módulo RTC DS3231



Fonte: elaborada pelo autor.

O módulo RTC DS3231 mostrado na Figura 7 é um *hardware* que possibilita o fornecimento de informações de datas e horários com elevada precisão. O dispositivo apresenta as conexões GND, VCC, SDA, SCL, SQW e 32k. Os pinos VCC e GND são alimentados por 0V e 5V respectivamente, os pinos SCL e SDA são responsáveis pelo estabelecimento da comunicação I2C já as conexões SQW e 32k não foram utilizadas no desenvolvimento do projeto.

A facilidade de aplicação e segurança dos dados, foram os motivos principais que motivaram o uso do Módulo RTD DS3231, as configurações das rotinas de controle da biblioteca DS3231 utilizada no projeto são bastante intuitivas. O dispositivo mantém as informações de data e horários seguras mesmo desconectado de alimentação da fonte externa graças a bateria de 3V conectada em seu corpo.

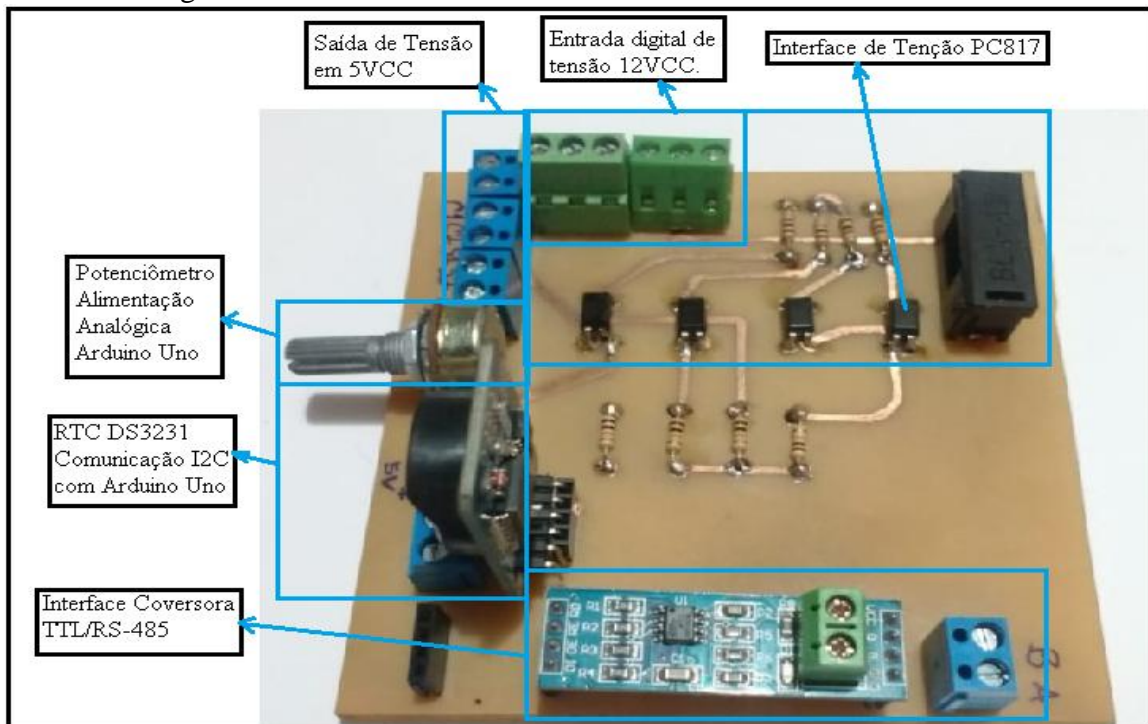
3.1.3 Interface de aquisição de dados

As entradas digitais da placa Arduino UNO obedecem a um range de tensão que identificam os sinais em nível lógico alto e baixo. O nível lógico alto apresenta intervalo de atuação entre 5V e 2,4V, o nível baixo atua entre 0 e 0,8V e as tensões com níveis intermediários podem causar o mau funcionamento do dispositivo.

Na indústria as presenças de equipamentos que apresentam alta taxa de interferência eletromagnética são bastante comuns, esses equipamentos normalmente possuem

diferentes níveis de tensões para o correto funcionamento. Em apenas uma maquina é possível encontrar várias tensões de alimentações que realizam diferentes funções. Tensões alternadas para o acionamento das cargas que necessitam de potências mais elevadas como motores, bobinas vibratórias, transformadores e entre outros, para a realização do controle normalmente são utilizados PLCs, IHMs que normalmente são alimentados com tensões de 24VCC ou 110 e 220VAC.

Figura 8 – Placa de Interface de Sinais



Fonte: elaborada pelo autor

Com toda essa quantidade de tensões incompatíveis com as entradas digitais da placa do Arduino UNO foi idealizada uma interface de tensão que conseguisse trazer mais confiabilidade ao funcionamento da aplicação. A interfase de tensão foi estabelecida com a utilização do circuito integrado PC817 mostrado na Figura 8. O circuito integrado PC817 presente na placa de aquisição de dados digitais atua proporcionando o isolamento galvânico de até 5000Vrms. A entrada do dispositivo optoacoplador é formado por um LED que necessita de um resistor limitador de corrente capaz de suportar uma potência de dissipação de 70mW. Na saída do PC817 existe um fototransistor que também necessita de um resistor para limitar a corrente visto que esse suporta apenas uma potência de 150mW.

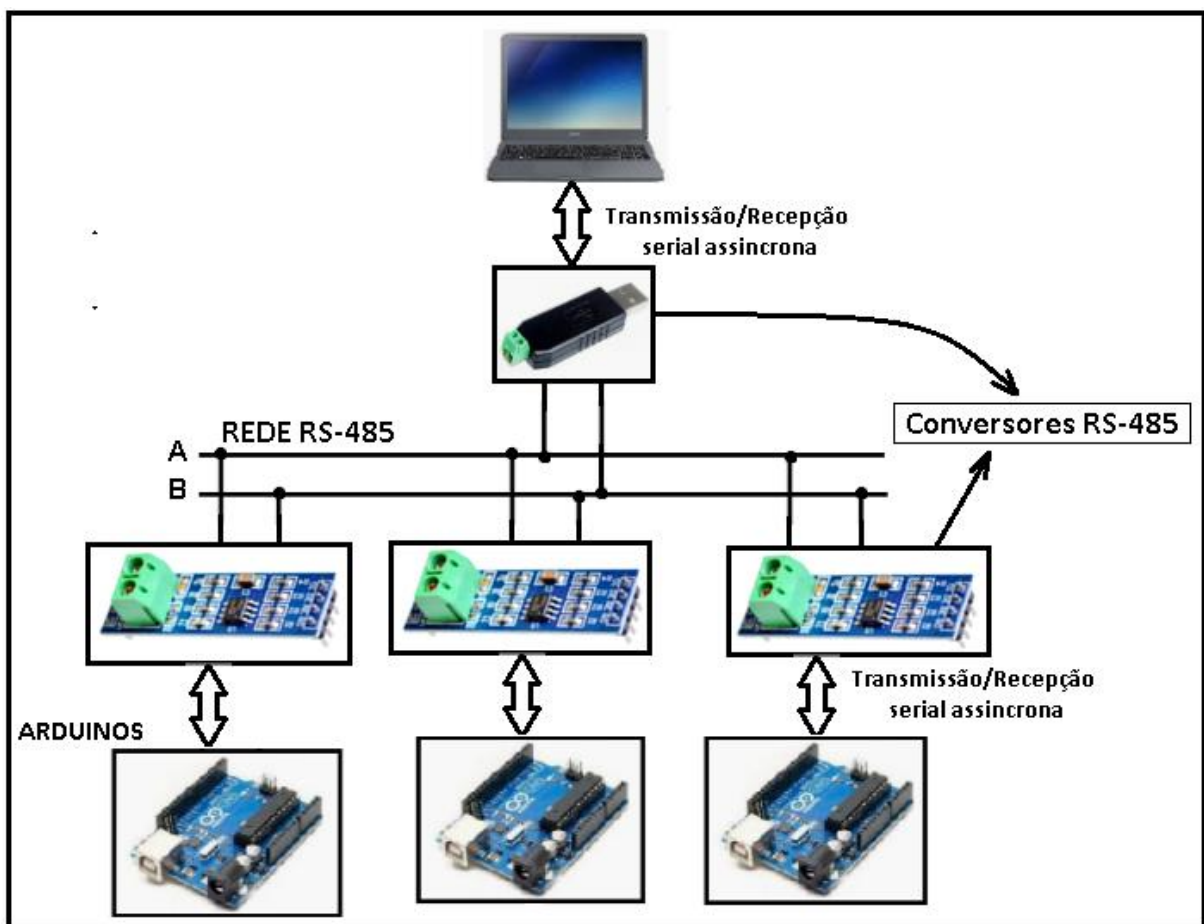
As entradas digitais que recebem os dados dos controladores de processos, foram

dimensionadas para 12VCC enquanto que o lado da saída do PC817 será conectado as entradas digitais de 5VCC do Arduino Uno.

A placa de Interface possui um potenciômetro linear de 10k Ω responsável pela alimentação da entrada analógica A0 do Arduino Uno e ainda estabelece suporte aos *Shields* RTC DS3231 e ao Conversor TTL/RS-485 que serão tratados posteriormente.

3.2 Rede RS-485

Figura 9 – Estrutura da rede RS-485



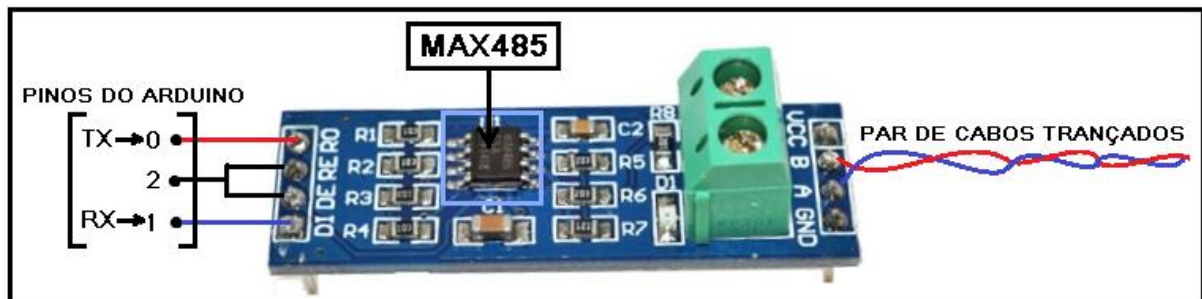
Fonte: elaborada pelo autor

A Figura 9 mostra rede RS-485 utilizada no desenvolvimento do projeto, foi construída utilizando dois dispositivos que realizam as conversões de dados necessárias para o estabelecimento da comunicação entre os equipamentos mestre e escravos. O Módulo TTL/RS-485 é responsável pela conversão e oferta dos dados recebidos da comunicação serial

do Arduino Uno, na outra ponta do barramento encontra-se o Módulo conversor RS-485/USB que permitirá ao computador transmitir e receber informações para a devida alimentação do software de supervisão. A estrutura adotada entre os equipamentos conversores foi à comunicação *half-duplex* onde é utilizado apenas um par de cabos trançados para a interligação de todos os equipamentos que compõem a rede RS-485.

3.2.1 Módulo Conversor TTL/RS-485

Figura 10 – Módulo conversor TTL/RS-485



Fonte: elaborada pelo autor

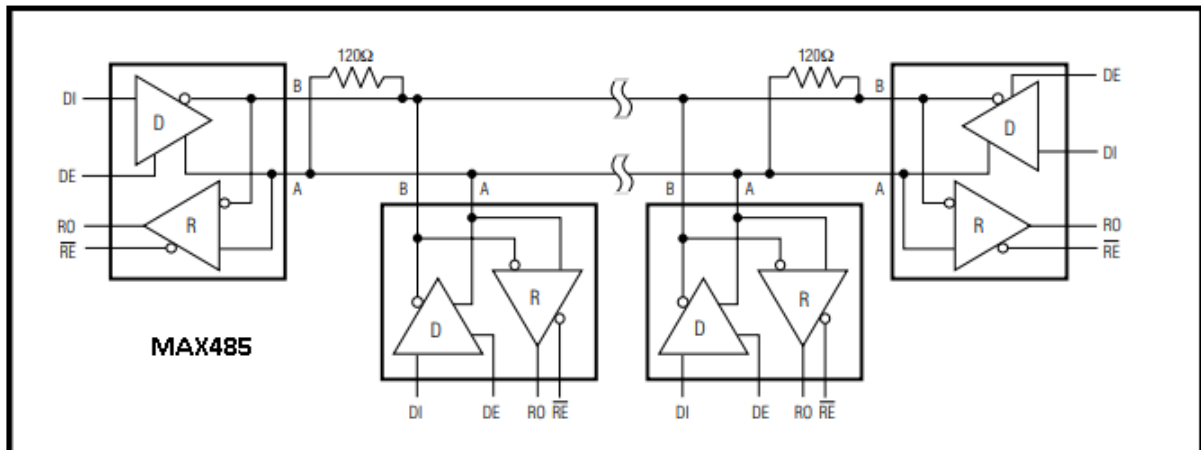
A Figura 10 representa o Módulo de conversão TTL/RS485, através deste é possível realizar as devidas transmissões e recepções de dados solicitadas pelas placas remotas de aquisição de dados.

Os pinos DI, DE, RE e RO presentes na entrada do conversor são dotados de resistores de *pull-ups* com valores de $10k\Omega$, estes são conectados aos bornes 0, 1 e 2 digitais do Arduino. O DI será conectado ao pino 1 que apresenta a configuração de saída para realizar a transmissão serial de dados TX, o conector RO do conversor será conectado ao pino 0 que foi ajustado como entrada para operar na Recepção dos dados RX e os pinos DE e RE são unidos e conectados ao borne 2 que teve o ajuste realizado como saída pra efetuar o controle das recepções e transmissões de dados.

O controle realizado sobre os pinos DE e RE interferem diretamente sobre os elementos *driver* e *receiver* que constituem o chip MAX485, quando a saída 2 do Arduino é colocada em nível lógico alto ocorrerá a habilitação do DI e forçará o RO para ficar em alta impedância possibilitando a transmissão dos dados, caso a saída 2 seja ajustada para nível lógico baixo o RO será habilitada e DI entrará em alta impedância tornando possível a

recepção dos dados.

Figura 11 – Conexões multiponto estabelecida pela circuito integrado MAX485.



Fonte: elaborada por Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers (2003).

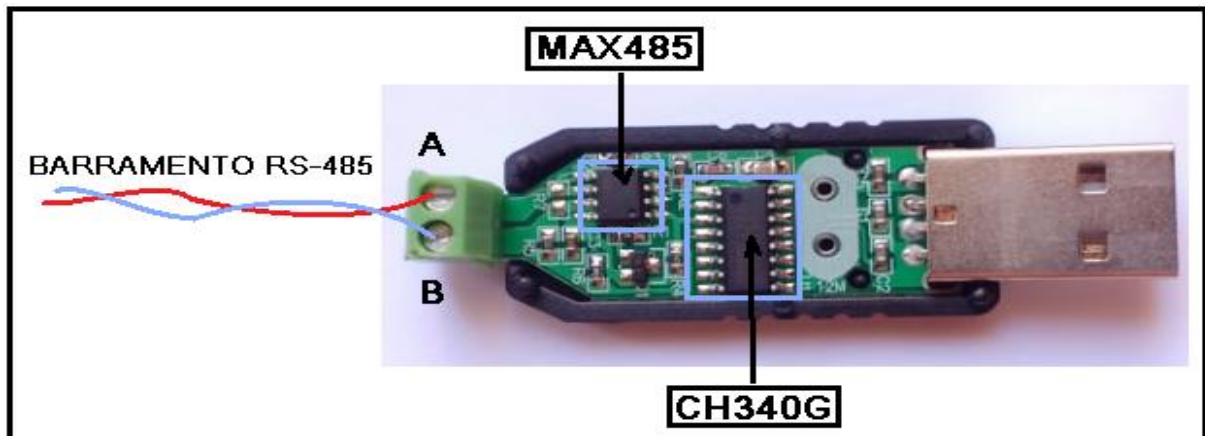
A conversão de sinais RS-485 em TTL são realizadas pelo circuito integrado MAX485 ilustrado na Figura 11, este possui em suas características elétricas os elementos *driver* e *receiver* que transcrevem as informações para o correto entendimento dos equipamentos que possui tecnologia TTL, este consegue organizar-se obedecendo à topologia multiponto, onde mais de um dispositivo pode se conectar a rede de comunicação.

No lado RS-485 do conversor existem resistores com valores de $20\text{K}\Omega$ que servem como resistores de *pull-up* e *pull-down* além do resistor de 120Ω que são utilizados entre o par de cabos trançados da rede para a atenuação do sinal que pode ser refletido pelas pontas do barramento, assim o sinal refletido será dissipado em forma de calor. Os resistores de 120Ω são colocados apenas nos dois dispositivos que ficam nas pontas da rede, ou seja, no equipamento mestre e no ultimo escravo da rede.

Os módulos conversores mesmo alimentados em 5V oferecem confiabilidade para a retirada ou reposição de dispositivos no barramento RS-485 sem sofrer qualquer dano de ruído ou queda de comunicação entre os outros elementos participantes.

3.2.2 Módulo Conversor RS-485/USB

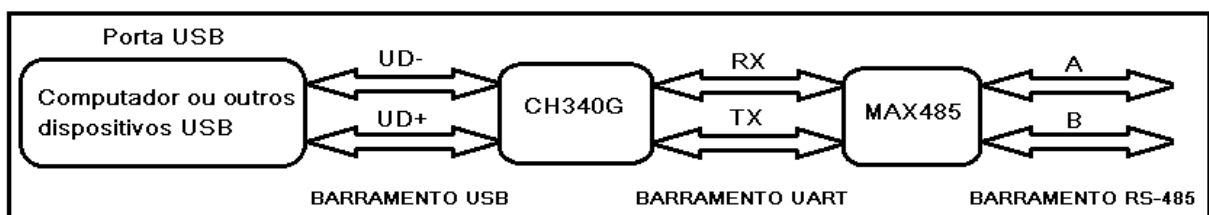
Figura 12 – Módulo Conversor RS-485/USB



Fonte: elaborada pelo autor

As informações trocadas entre a rede RS-485 e o computador ocorrem através das conversões de sinais realizadas pelo módulo de conversão RS-485/USB ilustrado na Figura 12, este dispositivo tem a capacidade de transformar a entrada USB em uma porta serial COM que será acessada pelo programa supervisor.

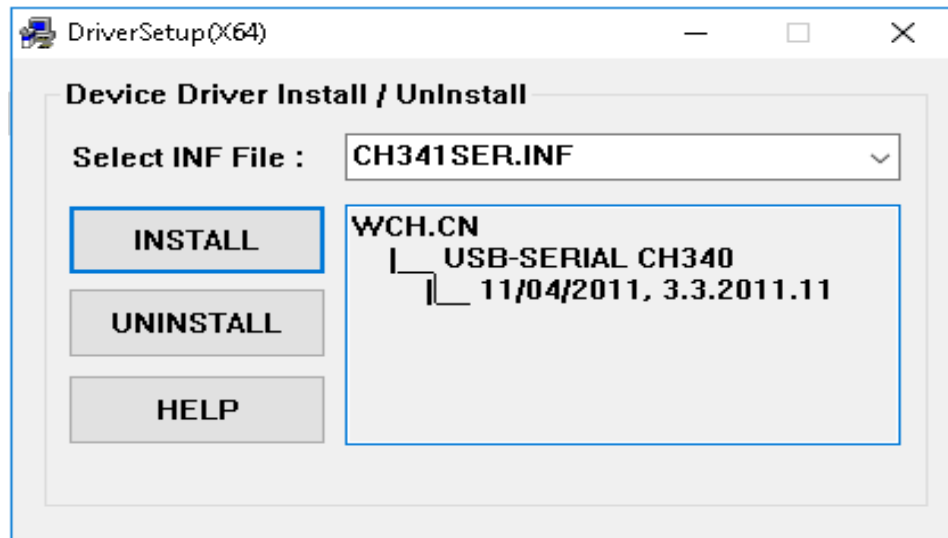
Figura 13 – Conversões internas realizadas pelo conversor RS-485/USB.



Fonte: elaborada pelo autor

O dispositivo de conversão USB Serial RS-485 apresenta em sua composição os CIs MAX485 e o CH340G. O MAX485 apresentado e discutido anteriormente na seção 3.2.1 está presente na composição dos escravos nos módulos de conversão TTL/RS485. O CH340G é responsável pela conversão USB/UART, este dispositivo transcreve os dados vindos da porta USB e disponibiliza no barramento UART onde será acessado pelo MAX485. Na Figura 13 estão indicadas as etapas de conversões de dados realizadas pelo dispositivo USB/RS485.

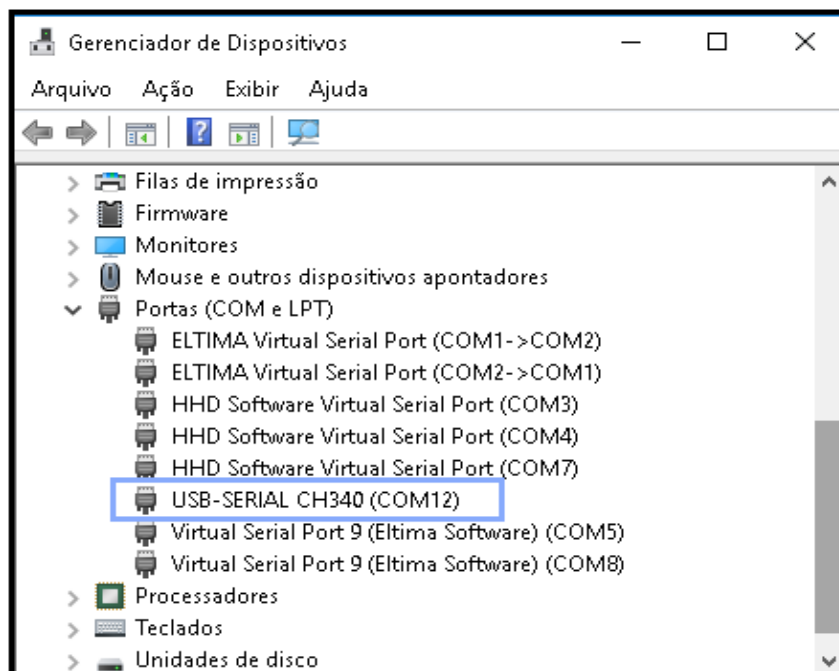
Figura 14 – Instalação do drive para utilização do conversor RS-485/USB



Fonte: elaborada pelo autor

O circuito integrado CH340G realiza a conversão da porta USB para serial COM, mas para efetuar a conversão é necessária a instalação do drive utilizando o programa ilustrado na Figura 14. Após instalação realizada ao conectar o dispositivo no computador ocorre a emulação de uma porta serial COM que pode ser visualizada através da Figura 15, que mostra o gerenciador de dispositivos do Windows 2010.

Figura 15 – Dispositivo conversor proto para utilização.



Fonte: elaborada pelo autor.

3.3 Mestre Rede Modbus RTU

O computador desempenha a função de executar o programa de Supervisão e Gerenciamento de Dados de Produção, esse sistema é responsável pelo controle da rede Modbus RTU, armazenamento de informações produtivas no banco de dados desenvolvido em MySQL, amostragens gráficas dos históricos de produção em dia e mês, realiza comparações produtivas entre os equipamentos presentes nas linhas industriais e dentre outros fatores que podem auxiliar as estratégias de tomadas de decisões pelos gestores.

3.3.1 Tela Principal do Programa de Gerenciamento de Dados

O Programa de Gerenciamento apresenta em sua tela principal vários painéis que permitem que os usuários consigam estabelecer as configurações do sistema e também realizar as devidas análises dos resultados das demandas dos processos industriais.

Figura 16 – Tela principal do Programa de Gerenciamento de Dados de Produção Industrial.



Fonte: elaborada pelo autor.

As configurações do Sistema de Supervisão são dirigidas pelas ações realizadas nos painéis de Configuração Serial, Análise de Conexão e de Gerência de Processos, estes são responsáveis pelos ajustes dos parâmetros da comunicação serial, monitoração das mensagens trocadas entre os equipamentos mestre e escravos e cadastro de variáveis que manipulam os dados dos processos industriais.

As análises de resultados são realizadas através dos históricos armazenados ou em tempo real, através dos painéis de Análise de Processos e Análise em Tempo Real. As verificações são realizadas por amostragens gráficas, possibilitando maior entendimento e rapidez do usuário na pesquisa e organização dos dados. As análises de resultados serão discutidas em maior detalhe no capítulo 4.

Os ajustes dos parâmetros para a realização das trocas de informações entre os equipamentos escravos Modbus RTU e o Sistema de Supervisão e Gerenciamento são realizados através do painel de Configuração Serial. Os parâmetros de *baud rate*, bit de dados, *stop bits*, paridade, Porta Serial e *Timeout* precisam ser preenchidos corretamente de modo que possibilite garantir a oferta de dados pelo protocolo Modbus RTU e a estabilidade da rede física RS-485.

Os ajustes de *baud rate*, bit de dados, *stop bits* e paridade precisam ser preenchidos obrigatoriamente com os mesmos valores nos equipamentos mestre e escravos da rede Modbus RTU, para que as trocas das mensagens possam ocorrer de forma correta. O parâmetro de Porta Serial será completado através da inserção do módulo Conversor RS-485/USB na porta USB, quando o sistema operacional detecta o conversor é criada uma porta serial COM que servirá como via de acesso aos dados que trafegam na rede RS-485. O *Timeout* será ajustado de modo a atender as condições de troca de Mensagens realizadas pelo protocolo Modbus RTU, este parâmetro sinaliza o tempo necessário entre as trocas de mensagens realizadas pelos equipamentos mestre e escravos da rede Modbus RTU.

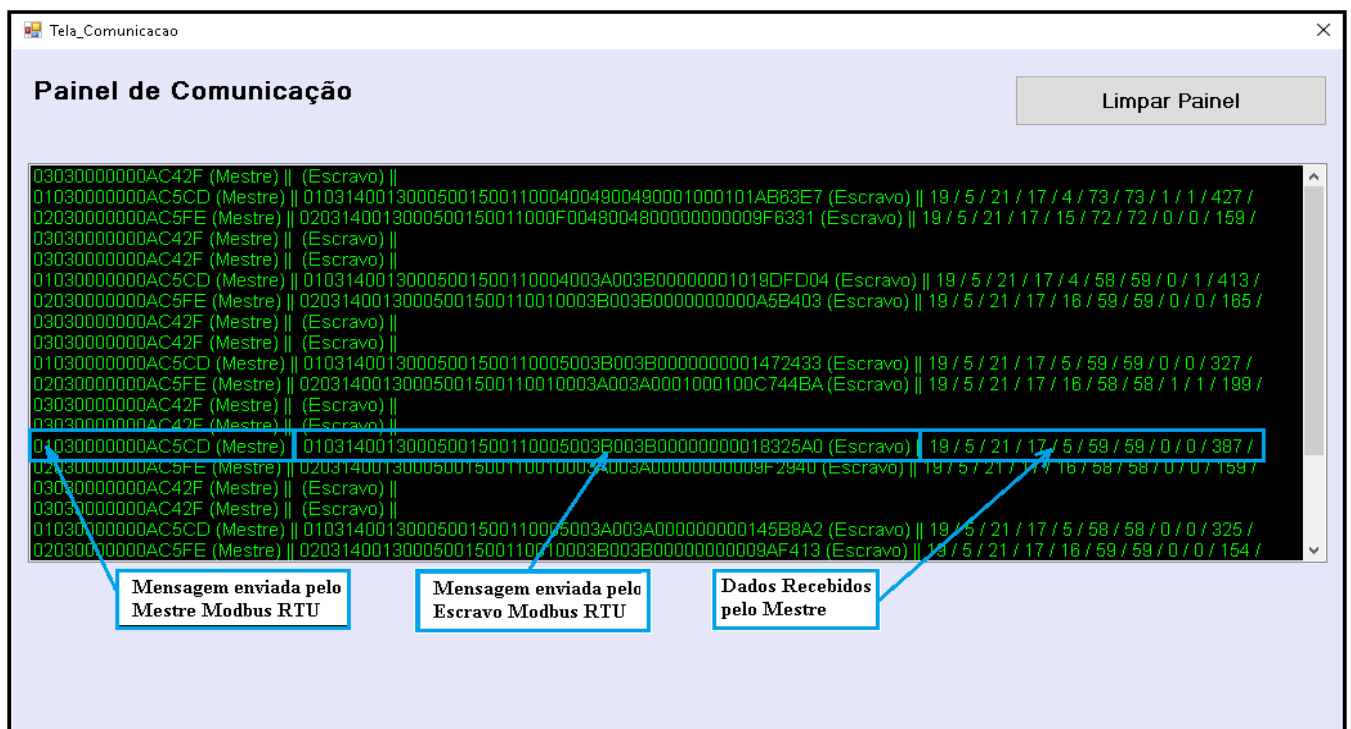
Após o preenchimento dos ajustes do painel de Configuração Serial serem realizados, a comunicação pode ser iniciada após apertar o botão de Iniciar Conexão ou encerrada através do botão de Parar Conexão o estado da comunicação poderá ser visualizada pelo sinalizador ao lado dos botões com as cores vermelho e verde para as ocorrências de paradas e execuções da comunicação serial respectivamente.

3.3.2 Tela de Comunicação Modbus RTU

A comunicação Modbus RTU poderá ser visualizada através do botão de Ativar Painel de Conexão localizado no Painel de Análise de Conexão, através da Tela de Comunicação é possível visualizar os bytes enviados pelo dispositivo mestre e os bytes retornados pelos dispositivos escravos, é possível notar as trocas de informações ocorrendo durante o timeout estabelecido no painel de Configuração Serial.

Através da exposição do Painel de Comunicação Modbus RTU, mostrado na Figura 17, é possível notar as composições das mensagens enviadas pelo mestre, às mensagens retornadas pelos dispositivos escravos e as amostragens dos dados de data, horários e *tags* separados por barras. O *software* consegue detectar também a não ocorrência de resposta dos dispositivos escravos, que neste caso ocorreu pela falta de alimentação elétrica de um dos Arduinos Uno. Caso não seja detectado nenhuma resposta do equipamento escravo o Sistema de Gerenciamento passará a solicitação ao próximo escravo da rede possibilitando que a comunicação continue estável e realizando as devidas coletas de dados.

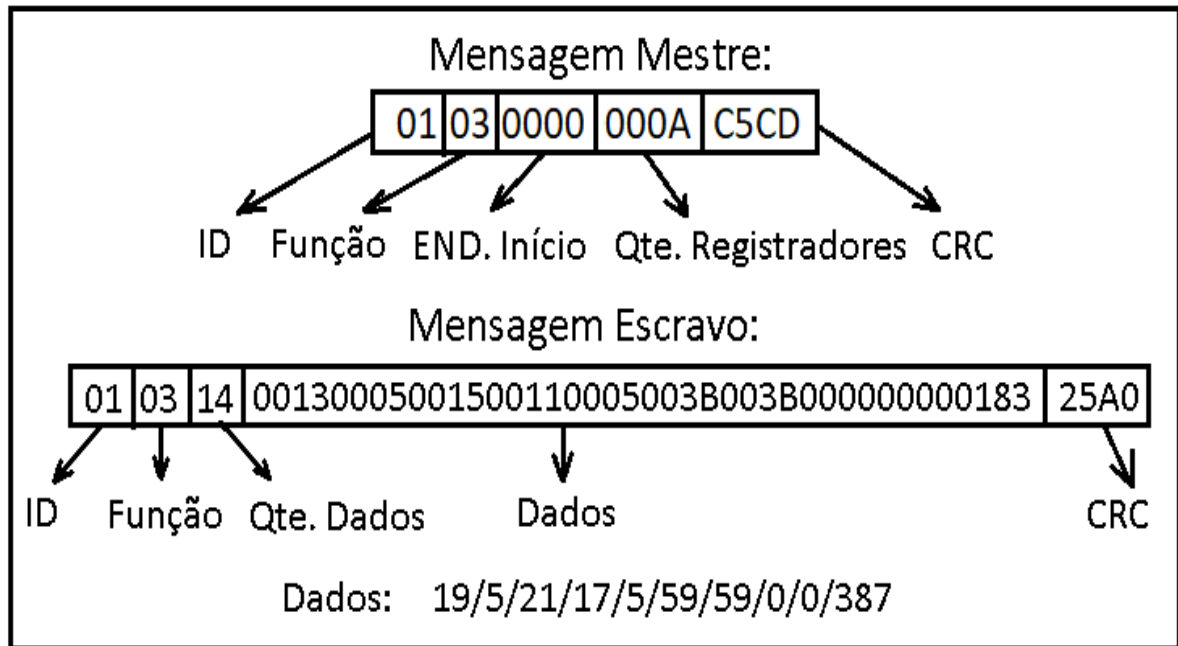
Figura 17 – Painel de Comunicação realiza o monitoramento das transferências de mensagens.



Fonte: elaborada pelo autor.

Foi escolhido como exemplo um trecho de comunicação da rede Modbus RTU selecionado na Figura 17, com a análise da Figura 18 tem-se o conteúdo da mensagem sendo desmembrado com o intuito de abordar os detalhes da comunicação entre os dispositivos mestre e escravos da rede.

Figura 18 – Exemplo detalhado da comunicação Modbus RTU.



Fonte: elaborada pelo autor

3.3.3 Tela de manipulação das variáveis dos processos industriais

Os recursos para os cadastros das *tags* e equipamentos que serão responsáveis pelo controle e alocação das informações das demandas dos processos industriais estão localizadas no painel de Gerência de Processo na Figura 16. Cada Equipamento e *tags* cadastrados são armazenados no banco de dados, com isso cada elemento terá seus históricos de eventos, podendo ser consultado posteriormente pelos painéis de Resultados.

O painel de Gerência de Processo fica responsável pelas operações de manipulações de equipamentos e *tags*. Ao pressionar o botão de Cadastro de Equipamentos e Tags abrirá uma tela de Cadastro ilustrada na Figura 19. Através desta será possível desenvolver cadastros, exclusão, atualização e consultas de registros. Os registros são compostos das colunas ID, nome_do Equipamento, tag_de_contagem1, tag_de_contagem2,

tag_de_status1, tag_de_status2 e tag_analogica, através desses fica possível mapear e denominar os Equipamentos e também as Tags conectadas a eles.

Figura 19 – Tela de Cadastro de Equipamentos e Tags.

GERENCIAMENTO DE EQUIPAMENTOS

Nome de Equipamento: ID do Equipamento:

GERENCIAMENTO DE TAGS

Nome da TAG: Funcao_Tag:

Nome Equipamento: ID do Equipamento:

LISTA DE EQUIPAMENTOS TAGS CADASTRADAS

id	nome_equipamento	tag_contadora1	tag_status1	tag_status2	tag_analogica
1	maquina1	pesagem	estado1	-	temperatura
2	maquina2	batidas	sta1	stad2	-
3	maquina3	bateladas	-	m2	quantidade

Total de tags cadastrados: 3

Fonte: elaborada pelo autor.

Para efetuar operações de cadastro, exclusão e atualização de equipamentos é necessário acessar o painel de gerenciamento de equipamento e preencher os parâmetros de Nome de Equipamento e ID do Equipamento. O parâmetro de ID do equipamento serve para identificar o acesso no banco de dados e também a localização do equipamento escravo da rede Modbus RTU. A denominação do equipamento acontece através do preenchimento do parâmetro de Nome de equipamento.

No painel de Gerenciamento de *tags* também são possíveis executar as operações de cadastro, exclusão e atualização, estas tarefas para serem realizadas com sucesso precisam indicar qual equipamento a *tag* está conectada, com isso todas as manipulações desenvolvidas nas *tags* necessitam das identificações de ID e do nome do equipamento.

4 RESULTADOS E DISCUSSÕES

As análises dos processos de produção são realizadas utilizando os painéis de Análise de Processo e Análise em Tempo Real mostrados na Figura 16, através destes as informações digitais armazenadas no banco de dados MySQL são interpretadas de forma gráfica. O usuário terá de efetuar o preenchimento de alguns parâmetros bem intuitivos para as verificações gráficas serem realizadas corretamente.

4.1 Acessando os Históricos de Produção Industrial

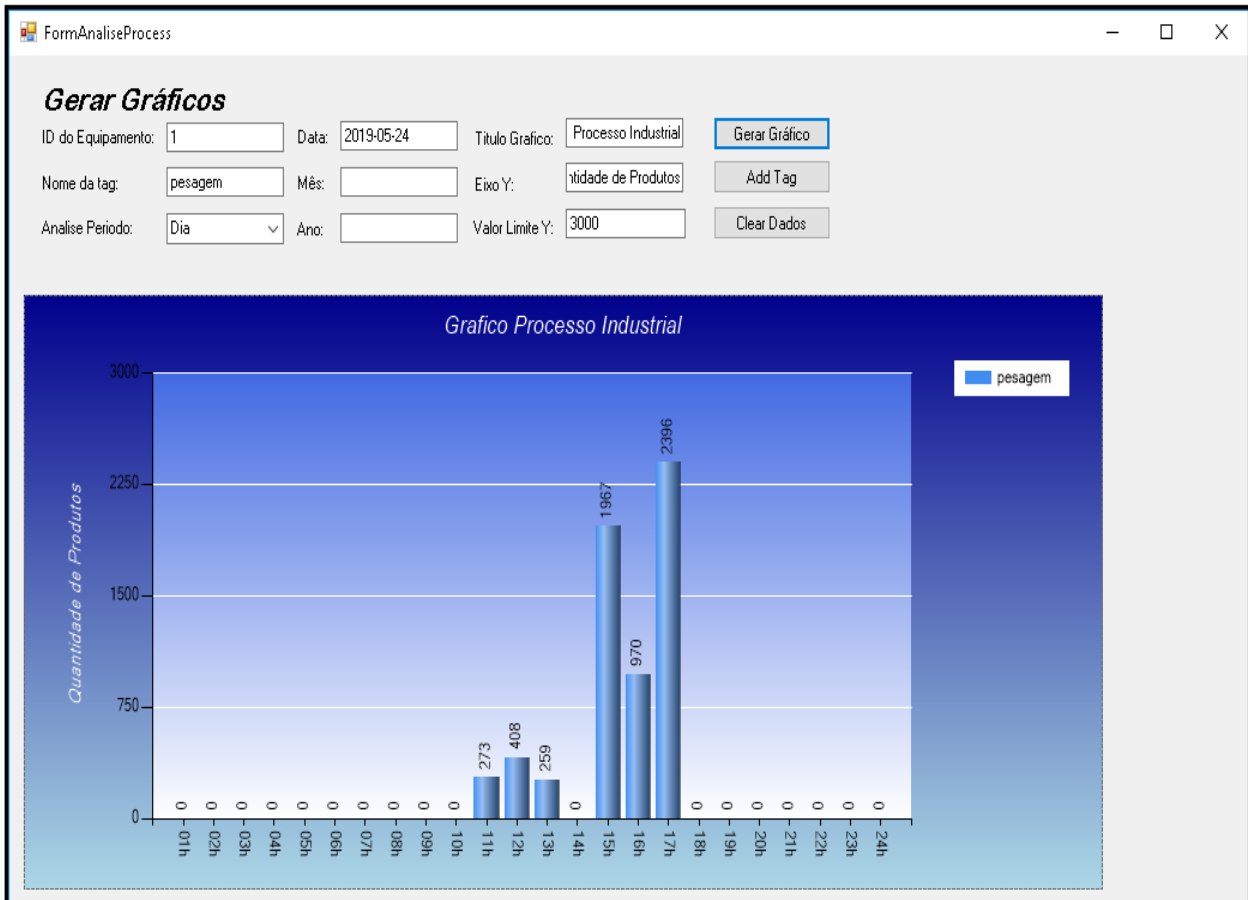
O painel de Análise de Processo possui três botões que acessam respectivamente os resultados referentes às *tags* de contagem, de estado e analógica. As consultas são realizadas baseadas nos históricos produtivos que serão verificados de acordo com os valores desejados pelos usuários.

4.1.1 Tela de Análise de Quantidade de Produtos

As verificações de quantidade de produtos produzidos são realizadas através das *tags* de contagem, estas são selecionadas de acordo com as indicações de Análise do Período, id e nome da *tag*. O parâmetro de Análise do Período é dividido entre dia e mês, quando a seleção ficar sobre o dia a solicitação requerida é a data no formato ano-mês-dia, caso seja selecionado o mês será necessário à indicação de mês e ano que se deseja pesquisar.

A resposta gráfica quando solicitado o dia será estruturada em divisões horárias com intervalos de uma hora e em quantidades que possui os limites estabelecidos pelo usuário nos parâmetros a serem preenchidos, o gráfico é representado na forma de colunas com os valores produtivos registrados nos picos. As ilustrações gráficas são mostradas na Figura 20.

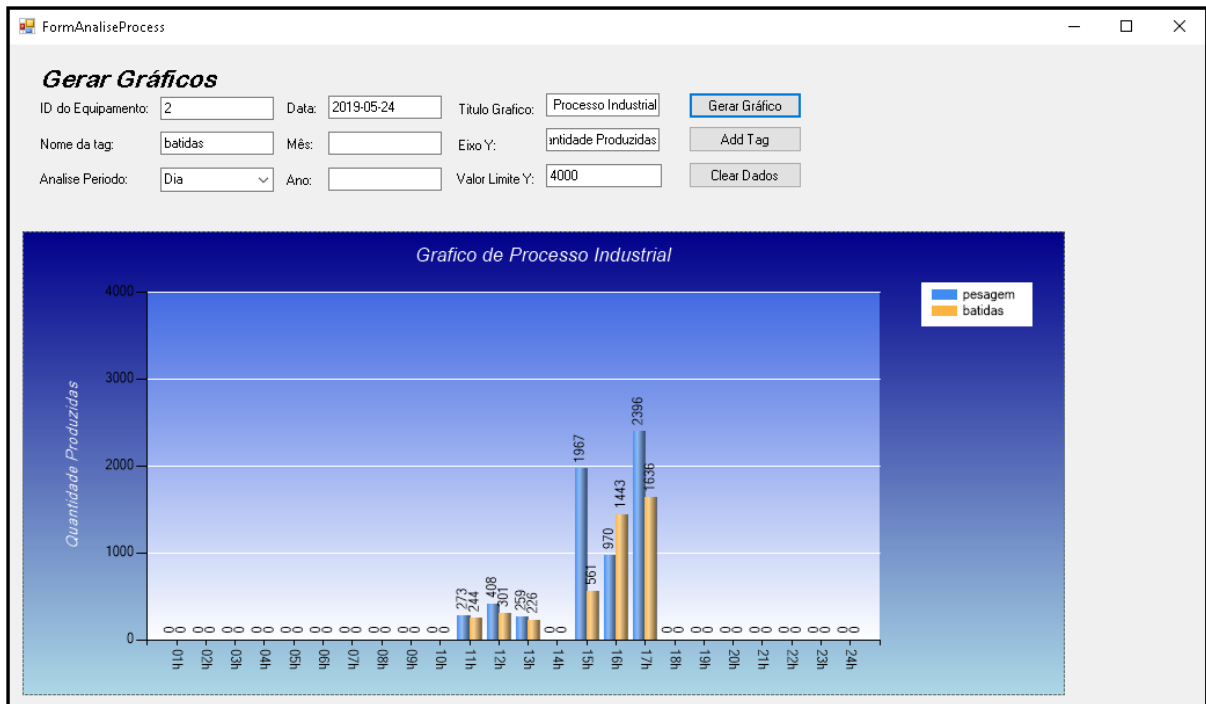
Figura 20 – Gráfico gerado através de uma consulta de produção diária.



Fonte: elaborada pelo autor.

As Tags de contagem também podem ser comparadas entre elas, permitindo análises de desempenho dos equipamentos verificados ou estabelecendo condições de gargalo de produção. As comparações são habilitadas através do botão Add Tag, estas são ilustradas na Figura 21.

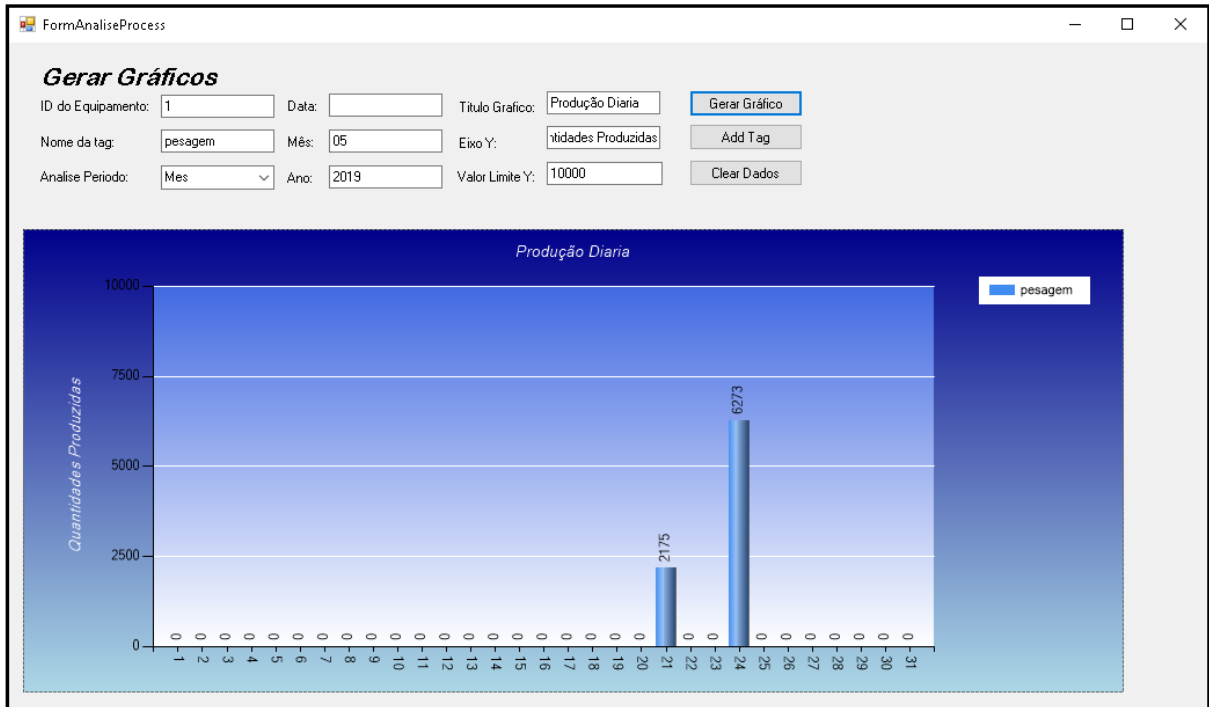
Figura 21 – Gráfico de comparações entre equipamentos em produção diária.



Fonte: elaborada pelo autor.

Quando for selecionada a condição de Mês no parâmetro de Análise de período o resultado obtido será um gráfico de coluna com os valores retornados de acordo com as quantidades produzidas diariamente durante todos os dias do mês. A Figura 22 representa a ilustração das quantidades de mensais produzidas.

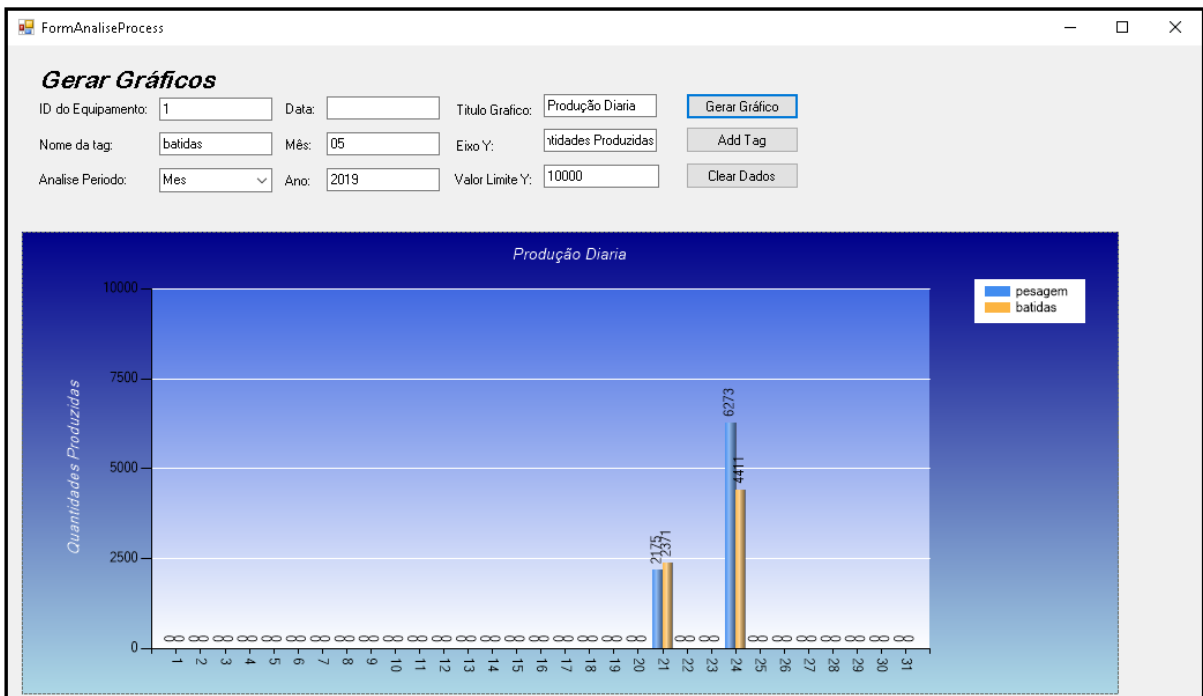
Figura 22 – Gráfico de análise produção mensal



Fonte: elaborado pelo autor.

As produções mensais também aceitam comparações entre tags, como apresentada na Figura 23.

Figura 23 – Gráfico de comparação entre equipamentos, produção mensal.

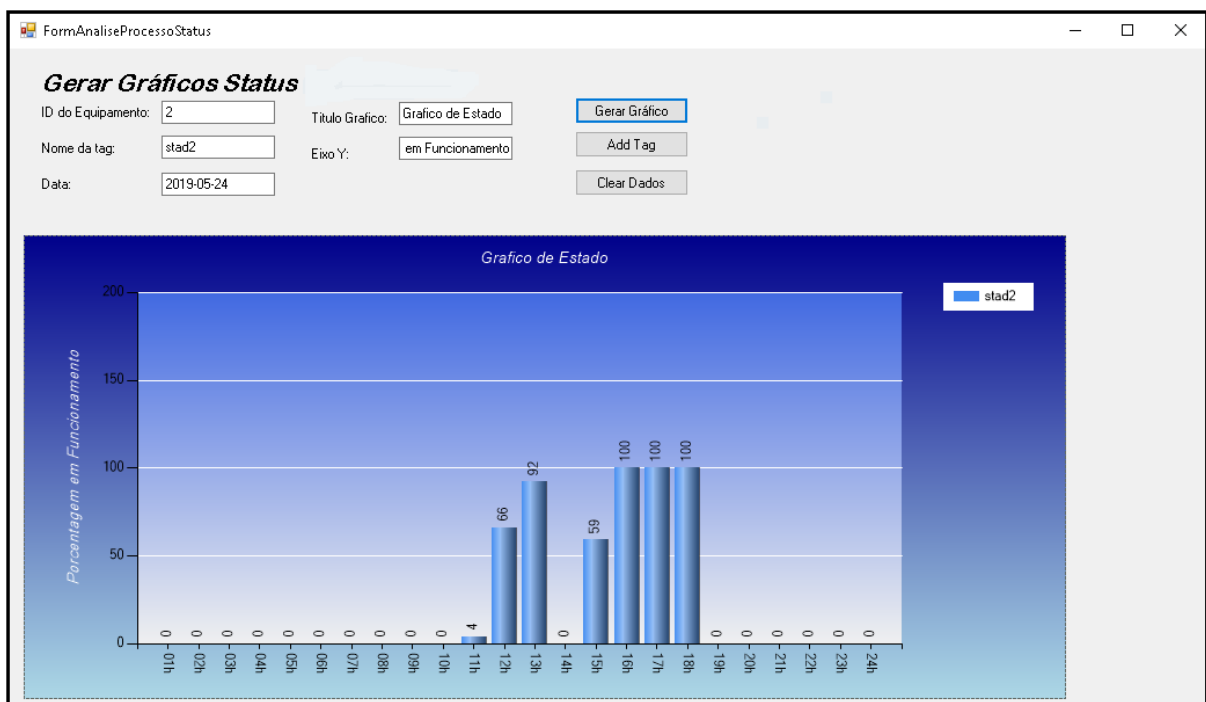


Fonte: elaborado pelo autor.

4.1.2 Tela de Análise de Estado

A análise da tag de estado é necessária para confirmar que o equipamento verificado está em operação ou disponível para funcionamento, através deste recurso é possível diagnosticar perdas de eficiência no processo produtivo como o consumo de energia sem que a máquina esteja produzindo, como a subutilização de equipamentos e entre outras consequências. A Tela de Análise de Estado apresenta apenas os parâmetros ID do Equipamento, Nome da tag, Data no formato ano-mês-dia, Título do Gráfico e Título do Eixo Y. Após o preenchimento desses parâmetros o usuário pressiona o botão Gerar Gráfico, então surgirá uma estrutura gráfica dividida em colunas organizadas em horas como apresentada na Figura 24.

Figura 24 – Gráfico de Estado analisa a porcentagem de disponibilidade dos equipamentos



Fonte: elaborada pelo autor.

4.1.3 Tela de Análise Analógica

As informações analógicas são selecionadas através da consulta da Tabela de Dados Analógicos, os parâmetros de entrada são ID do Equipamento, Nome da tag, Data no formato ano-mês-dia e a Hora. As seleções dos dados analógicos estão ligadas diretamente ao monitoramento dos vários processos industriais que normalmente suportam baixos intervalos de variação de suas grandezas físicas. A Figura 25 mostra o retorno dos dados analógicos tabelados e prontos para consulta.

Figura 25 – Consulta horária de dados Analógicos

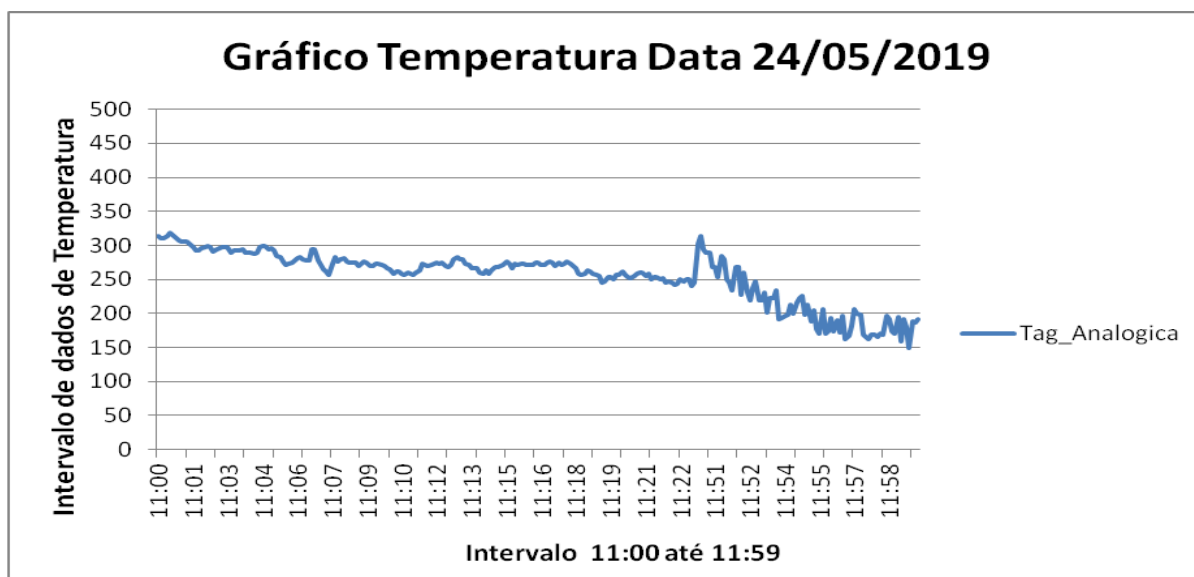
The screenshot shows a software window titled "FormTabelaAnalogica" with a title bar containing standard Windows window controls. The main area is titled "Gerar Tabela Dados Analógicos". It contains a form with the following fields: "ID do Equipamento:" with the value "1", "Data:" with the value "2019-05-24", "Nome da tag:" with the value "temperatura", and "Hora:" with the value "11". There is a "Gerar Tabela" button. Below the form is a table titled "Tabela Dados Analógicos" with the following data:

id	nome Equipamento	nome_tag_analogica	valor_tag_analogica	valor_hora
1	maquina1	temperatura	313	24/05/2019 11:00:00
1	maquina1	temperatura	310	24/05/2019 11:00:00
1	maquina1	temperatura	310	24/05/2019 11:00:00
1	maquina1	temperatura	313	24/05/2019 11:00:00
1	maquina1	temperatura	318	24/05/2019 11:00:00
1	maquina1	temperatura	316	24/05/2019 11:00:00
1	maquina1	temperatura	312	24/05/2019 11:00:00
1	maquina1	temperatura	308	24/05/2019 11:00:00
1	maquina1	temperatura	306	24/05/2019 11:01:00
1	maquina1	temperatura	306	24/05/2019 11:01:00
1	maquina1	temperatura	305	24/05/2019 11:01:00
1	maquina1	temperatura	302	24/05/2019 11:01:00

Fonte: elaborada pelo autor.

Os dados podem ser copiados da tabela para ser tratados no software Excel 2007, como a quantidade de dados é bastante elevada foi escolhida como amostra o intervalo de 11:00 até 11:59, durante esse período é possível observar com clareza na Figura 26 o comportamento da tag Temperatura.

Figura 26 – Gráfico montado com os dados recolhidos da tabela de dados analógicos.



Fonte: elaborada pelo autor

4.2 Monitoramentos em tempo real

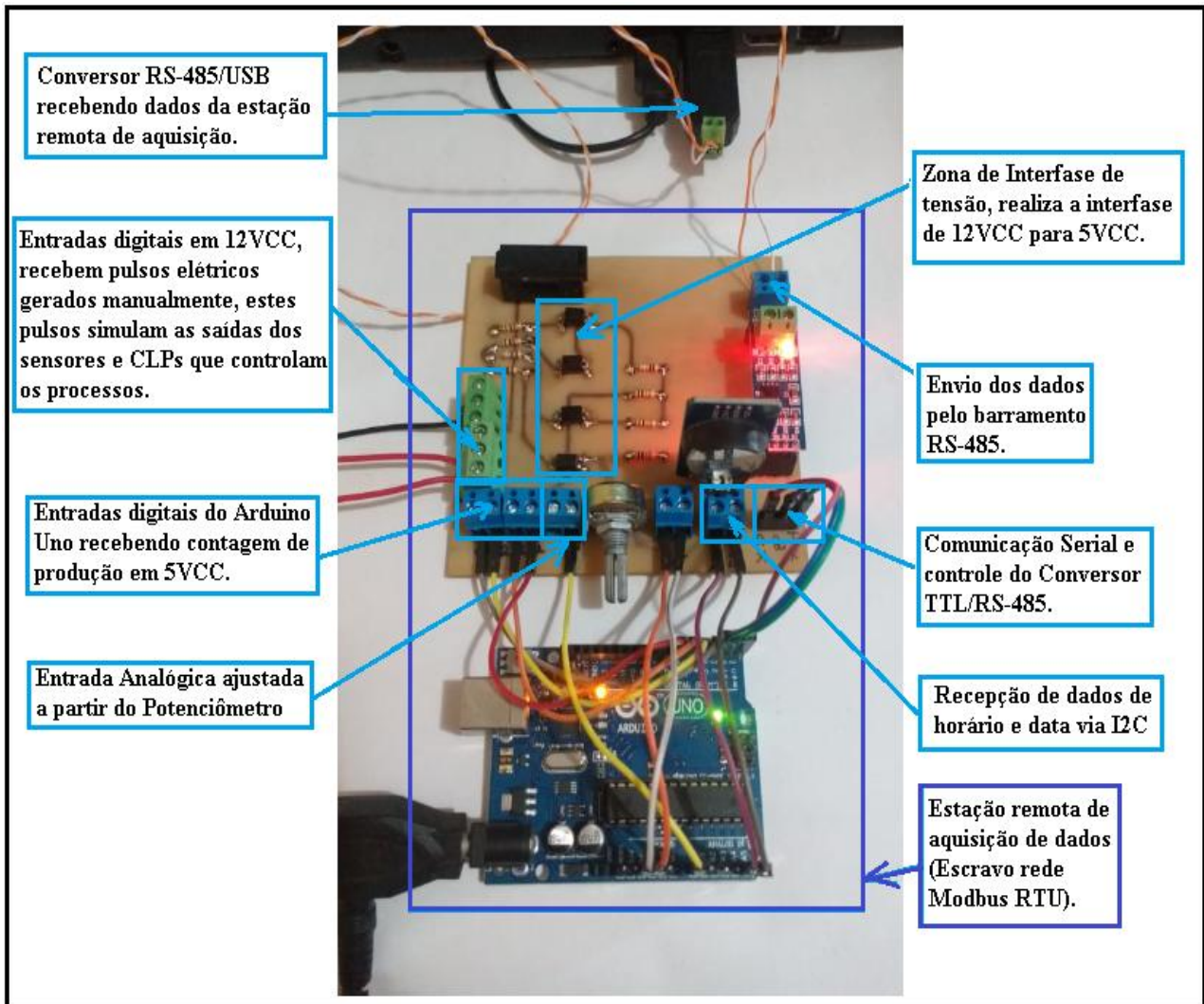
O painel de Análise em Tempo Real contém um botão de Processo em Tempo Real, através deste é acessada a Tela de Análise em Tempo Real que permite que a variável de contagem seja monitorada de forma gráfica de acordo com atualizações realizadas pelo próprio programa.

As análises são atualizadas a cada 10s, o gráfico é organizado em colunas separadas de acordo com os horários das ocorrências dos eventos, o formato de representação gráfica é a mesma da Figura 20 mostrada na seção 4.1.1.

4.3 Montagem do Sistema Remoto de Aquisição de Dados

Os resultados gráficos abordados nas seções 4.1 e 4.2 foram obtidos com êxito graças ao bom desempenho dos instrumentos que constituem a Estação Remota de Aquisição de Dados. A montagem prática foi estabelecida utilizando as placas de aquisição remota de dados e Arduino Uno, através da Figura 26 é possível identificar as conexões realizadas entre as placas e as funções executadas por cada elemento.

Figura 27 – Montagem prática da estação remota de aquisição de dados



Fonte: Elaborado pelo autor.

4.4 Análise de custos dos equipamentos utilizados

O custo do projeto foi estabelecido através da tabela de custos apresentada na Figura 27. Os valores foram adquiridos tomando como base a utilização de uma estação remota de aquisição de dados e os conversores responsáveis pelo estabelecimento da rede RS-485.

Figura 28 – Tabela de custos de uma placa de aquisição e estabelecimento da rede RS-485.

Tabela de Custos do Projeto			
Descrição	Valor Unidade	Qte.	Valor Acumulado
Arduino Uno	54,90	1	54,90
Cabos	0,36	40	14,40
Barra de Pinos	1,00	1	1,00
Barra p/ pinos Fêmea	1,00	1	1,00
Conector KRE 2 vias	0,95	6	5,70
Conector KRE 3 vias	1,30	2	2,60
RTC DS3231	12,80	1	12,80
Optoacoplador PC817	0,90	4	3,60
Potenciômetro Linear 10k Ω	1,90	1	1,90
Porta Fusível 5x20mm	2,00	1	2,00
Fisível 0.5A	0,25	1	0,25
Resistores 10k Ω 1/4W	0,15	4	0,60
Resistores 1k Ω 1/4W	0,15	4	0,60
Conversor RS-485/TTL	11,00	1	11,00
Conversor RS-485/USB	15,89	1	15,89
Placa fenolite 10x15cm	7,50	1	7,50
Total			R\$ 135,74

Fonte: elaborada pelo autor

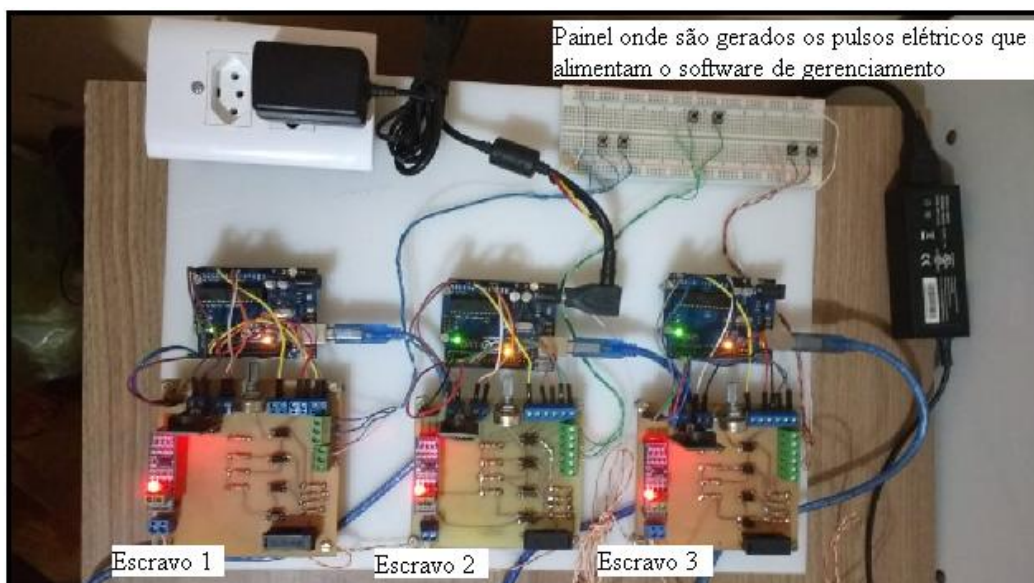
5 CONCLUSÃO

O programa de gerenciamento dos dados de produção conseguiu atender os objetivos propostos com elevada precisão possibilitando maior organização dos dados através de análises e comparações gráficas. O programa apresenta várias telas que tornam possível as consultas de históricos de produção ou consultar a situação atual de produção baseadas em atualizações propostas pelo próprio sistema.

Os equipamentos escravos mostraram-se bastantes eficazes nas aquisições e transmissão dos dados, o Arduino Uno responsável pelo processamento dos dados recebidos da placa de interface e do módulo RTC DS3231 apresentou-se como equipamento de fácil utilização tanto da plataforma do software quanto do hardware, a placa oferta as condições necessárias para a execução da aplicação.

Os gráficos de produção exibidos em todo o capítulo 4 conseguiu exibir os comportamentos de funcionamento dos equipamentos durante os períodos diário e mensal com êxito, permitindo acelerar as tomadas de decisão e o estabelecimento de estratégias em períodos reduzidos. Todas as operações só foram possíveis graças ao suporte de armazenamento do banco de dados MySQL.

Figura 29 – Comunicação RS-485 estabelecida, utilizando três escravos em rede.



Fonte: elaborada pelo autor.

O estabelecimento da rede RS-485 utilizando os dispositivos conversores TTL/RS-485 conectados aos dispositivos escravos e o conversor RS-485/USB conectado ao dispositivo mestre mostraram-se bastantes estáveis. A rede RS-485 foi estabelecida na quantidade de três escravos como mostrado na Figura 28, foram realizados testes de saídas e reposições dos dispositivos escravos durante as transmissões e mesmo assim as condições de estabilidade da rede RS-485 conseguiram ser suportadas.

O protocolo Modbus RTU utilizado em conjunto com a rede RS-485 mostrou elevada segurança nas transmissões dos dados. As mensagens Modbus RTU foram transmitidas com sucesso, podendo ser monitoradas pelo painel de comunicação em tempo real, permitindo a realização de diagnósticos de problemas em tempo bastante reduzido.

A elaboração do projeto utilizando elementos acessíveis e de custos bastante reduzidos era o principal objetivo a ser alcançado, mas para isso ocorrer a mão de obra foi elevada, visto a quantidade de elementos que necessitavam funcionar interligados trocando informações entre si. Mas a dedicação e os resultados obtidos ao término do trabalho foram muito recompensadores permitindo ao projetista acreditar em desenvolvimentos de melhoras futuras para o projeto.

5.1 Trabalhos Futuros

Várias ideias de aplicações podem ser citadas para fortalecer a utilização do trabalho apresentado, dentre elas pode-se indicar:

- A criação de uma placa remota de aquisição de dados mais sólida e robusta fisicamente.
- Realizar comunicação via WIFI com o objetivo de reduzir a mão de obra de instalação e economizar na compra de cabos elétricos.
- Disponibilizar as informações coletadas na WEB.

REFERÊNCIAS

ALBINI, Luiz Carlos Pessoa; **Rede de Computadores I**. 2019.

ALFA INSTRUMENTOS; **Protocolo de Comunicação Modbus RTU/ASCII**. 2000

AXMARK, David et al. **MySQL Reference Manual: MySQL até versão 5.0.0-alfa** – Suécia, 2003.

BAYER, Fernando Mariano; ECKHARDT, Moacir; Machado, Renato. **Automação de Sistemas**. 4. ed. Santa Maria: Universidade Federal de Santa Maria, Colégio Técnico Industrial de Santa Maria; Escola Técnica Aberta do Brasil, 2011.

BARBOSA, Andre Sarmento; **Interfaces e Periféricos**. 2014.

COELHO, Marcelo Saraiva; **Sistemas Supervisórios**. - São Paulo: Instituto Federal Campus Cubatão, 2010.

DEITEL, Harvey et al. **C# Como Programar**. São Paulo: 2002.

LIMA, Edwin; **C# e .NET – Guia do desenvolvedor**. Rio de Janeiro: Campus, 2002.

MAXIM INTEGRATED. **Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers**. 2003.

MCROBERTS, Michael; **Arduino Basico**. São Paulo: Novatec Editora, 2011.

NIEDERAUER, Juliano; PRATES, Rubens; **Guia de consulta rápida MySQL 5**. São Paulo: Novatec Editora, 2006.

PADRÃO RS-485; Disponível:

http://olaria.ucpel.tche.br/autubi/lib/exe/fetch.php?media=padrao_rs485.pdf. Acesso em: 30 de maio de 2019.

RIOS, Renan Osório; **Protocolos e Serviços de Redes**. Colatina: Instituto Federal Espírito Santo, Rede e-Tec Brasil, 2012.

WEG S. A.. **Manual da Comunicação Serial RS232/RS485**. 2010.

ROGGIA, Leandro; FUENTES, Rodrigo Cardozo. **Automação Industrial**. Santa Maria: Universidade Federal de Santa Maria, Colégio Técnico Industrial de Santa Maria, Rede e-Tec Brasil, 2016.

RIBEIRO, Marco Antônio; **Automação Industrial**. 4. ed. Salvador: 2001.

SANTOS, Nuno Pessanha; **Arduino Introdução e Recursos Avançados**. Escola Naval, 2009.