

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO SUPERIOR DE TECNOLOGIA EM AUTOMAÇÃO INDUSTRIAL**

CLAIRTON CASTILHO BATISTA

**MONITORAMENTO DE PRODUÇÃO DE EMBALAGENS UTILIZANDO
LABVIEW™**

PONTA GROSSA

2019

CLAIRTON CASTILHO BATISTA

**MONITORAMENTO DE PRODUÇÃO DE EMBALAGENS UTILIZANDO
LABVIEW™**

Trabalho de Conclusão de Curso apresentado como requisito parcial para obtenção de grau no Curso Superior de Tecnologia em Automação Industrial da Universidade Tecnológica Federal do Paraná – campus Ponta Grossa.

Orientador: Prof. Dr. Felipe Mezzadri

PONTA GROSSA

2019



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Ponta Grossa



Diretoria de Graduação e Educação Profissional Departamento Acadêmico de Eletrônica
Tecnologia em Automação Industrial

TERMO DE APROVAÇÃO

MONITORAMENTO DE PRODUÇÃO DE EMBALAGENS UTILIZANDO LABVIEW

por

CLAIRTON CASTILHO BATISTA

Este Trabalho de Conclusão de Curso foi apresentado em 04 de julho de 2019 como requisito parcial para a obtenção do título de Bacharel(a) em Tecnologia em Automação Industrial. O(A) candidato(a) foi arguido(a) pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof(a). Dr. Felipe Mezzadri

Orientador(a)

Prof(a). Dr. Frederic Conrad Janzen

Membro Titular

Prof(a). Dr. Josmar Ivanqui

Membro Titular

Prof. Dr. Josmar Ivanqui

Responsável pelos TCC

Prof. Dr. Felipe Mezzadri

Coordenador do Curso

– O Termo de Aprovação assinado encontra-se na Coordenação do Curso –

RESUMO

BATISTA, Clairton Castilho. **Monitoramento de produção de embalagens utilizando LabVIEW™**. 74 p. Trabalho de Conclusão de Curso (Curso Superior de Tecnologia em Automação Industrial). Universidade Tecnológica Federal do Paraná. Ponta Grossa. 2019.

Este trabalho de conclusão de curso apresenta a proposta de desenvolvimento de um sistema de monitoramento de produção através da linguagem gráfica de programação LabVIEW™. Visando maior confiabilidade nos dados de produção coletados em uma fábrica de embalagens, desenvolveu-se um sistema capaz de monitorar as quantidades produzidas, bem como registrá-las em um banco de dados juntamente com os tempos de paradas de máquina de forma automática, a fim de manter um histórico e gerar relatórios de forma mais rápida e prática, direcionando o tempo dos operadores a outras atividades inerentes ao processo de produção. Para tanto, buscou-se integração com um sistema de gestão de banco de dados, juntamente com uma arquitetura de comunicação cliente-servidor, aproveitando a estrutura de contagem de produção já instalada na fábrica através de um controlador lógico programável. Através das telas desenvolvidas, os operadores poderão realizar consultas rápidas sobre a produtividade das linhas de produção, bem como confirmar os registros do sistema a fim de garantir a confiabilidade dos dados. Através dos testes realizados em uma das linhas de produção, o sistema mostrou-se eficiente no que tange sua funcionalidade básica aqui proposta, podendo futuramente admitir melhorias específicas, como geração de relatórios personalizados, comunicação com o ambiente de rede corporativa, bem como o controle remoto das linhas de produção.

ABSTRACT

BATISTA, Clairton Castilho. **Monitoring of packaging production using LabVIEW™**. 74 p. Graduation Course Work (Technology in Industrial Automation). Federal Technological University of Paraná. Ponta Grossa. 2019.

This work presents the proposal for the development of a system of monitoring of production using the graphical programming language LabVIEW™. Aiming for greater reliability in the production data collected in a packaging factory, a system was developed able to monitor the quantities produced, as well as to register them in a database along with the automatic stop times, in order to maintain a history and generate reports more quickly and practically, directing the time of the operators to other activities inherent in the production process. To do this, we are looking for integration with a database management system, along with a client-server communication architecture, taking advantage of the production count structure already installed in the factory through a programmable logic controller. Through the developed screens, operators can conduct quick queries on the productivity of production lines, as well as confirm system records to ensure data reliability. Through the tests performed on one of the production lines, the system proved to be efficient in terms of the basic functionality proposed here and may in the future support specific improvements such as customized reporting, corporate network and remote-control environment communication. of production lines.

LISTA DE FIGURAS

Figura 1: Exemplos de controles e indicadores no Painel Frontal.....	22
Figura 2: Exemplo de Terminais, Nós e Ligações no diagrama de blocos.	23
Figura 3: Exemplo de representação das ligações.....	23
Figura 4: Exemplo de estrutura <i>While Loop</i> e <i>For Loop</i>	24
Figura 5: Exemplo de <i>Shift Register</i> interagindo com uma estrutura <i>For Loop</i> e somando elementos de um arranjo.	25
Figura 6: Exemplo de Arranjo de duas dimensões sendo criado através de iterações das estruturas <i>For Loop</i>	26
Figura 7: Exemplo de junção de diferentes tipos de dados em um <i>Cluster</i> , que depois são separados novamente.....	26
Figura 8: Estruturas <i>Case</i> , <i>Sequence</i> e <i>Event</i>	27
Figura 9: Exemplo de utilização de variáveis compartilhadas.	28
Figura 10: Exemplo de utilização das funções de conectividade a banco de dados.	29
Figura 11: Fluxograma do processo de apontamento de produção.	32
Figura 12: Fluxograma de apontamento de produção desejado a partir do desenvolvimento do sistema de monitoramento de produção.	34
Figura 13: Posição do sensor de contagem boa em uma das linhas de produção.	35
Figura 14: Variáveis utilizadas para o monitoramento de produção.....	36
Figura 15: Configurações <i>NI OPC Server 2016</i> para o sistema de monitoramento de produção.	37
Figura 16: Fluxo dos dados para o sistema de monitoramento de produção.	37
Figura 17: Simulador de máquina.vi – Painel Frontal.....	38
Figura 18: Simulador máquina.vi – Diagrama de Blocos.....	39
Figura 19: Simulador Produção.vi – Painel frontal.	40
Figura 20: Simulador Produção.vi – Diagrama de Blocos.	41
Figura 21: <i>SubVI</i> Gerador de Pulso único.vi e registro de horas totais.	42
Figura 22: <i>SubVI</i> Tempos.vi.....	42
Figura 23: Fluxo de dados do <i>SubVI</i> de simulação de máquina e alimentação dos indicadores do painel frontal através das variáveis compartilhadas.....	43
Figura 24: <i>MySQL Workbench</i> mostrando a lista de tabelas do banco de dados e uma consulta a tabela ‘Paradas Lançadas’, retornando todos os registros desta tabela.	45

Figura 25: Cadastro.vi na página de cadastro dos códigos de paradas.....	46
Figura 26: Diagrama de blocos Cadastro.vi – verificação da existência de tabelas no banco de dados.....	47
Figura 27: Diagrama de blocos Cadastro.vi - estrutura de caso que cria todas as tabelas que serão utilizadas pelo sistema de monitoramento de produção.	47
Figura 28: Diagrama de blocos Cadastro.vi – Estrutura de caso que processa as entradas do usuário e atualiza o banco de dados.....	48
Figura 29: Cadastrar.vi – Diagrama de blocos.	48
Figura 30: Painel frontal Cadastros.vi – Demonstração de exclusão de registro.	49
Figura 31: Segundo Plano.vi – Diagrama de blocos.	50
Figura 32: Segundo Plano.vi – <i>SubVI</i> Gerador de Pulso Único recebendo informação da variável compartilhada Turnos.	50
Figura 33: Inserir Paradas a Lançar.vi – Diagrama de blocos.....	51
Figura 34: Segundo Plano.vi – Representação da estrutura que armazena as informações de paradas das linhas de produção.	52
Figura 35: Segundo Plano.vi – Estrutura de caso para gerar a informação da variável global ‘ <i>String</i> Turno’.	52
Figura 36: Segundo Plano.vi – Estruturas de caso que direcionam as informações de acordo com o índice que representa o turno decorrente.....	53
Figura 37: Inserir Produção Parcial.vi – Diagrama de Blocos.	53
Figura 38: Monitoramento Online.vi – Painel frontal.....	54
Figura 39: Monitoramento Online.vi – Diagrama de blocos.....	55
Figura 40: Dados de produção.vi – Painel Frontal.	56
Figura 41: Dados de Produção.vi – Diagrama de blocos.	58
Figura 42: Diagrama de Estados ‘Apontamento de Paradas’.....	59
Figura 43: Diagrama de Estados ‘Apontamento de Produção’.....	60
Figura 44: Relatórios.vi – Painel frontal mostrando os controles para consulta de dados de paradas por máquina.....	61
Figura 45: Relatórios.vi – Painel frontal mostrando os controles para consulta de dados de apontamento SAP.....	61
Figura 46: Relatórios.vi – Diagrama de blocos.....	62
Figura 47: Diagrama de Estados ‘Relatórios’.....	63
Figura 48: Relatório Diário de Produção.vi – Diagrama de blocos.....	64

Figura 49: Modelo Relatório Produção.vi – Painel frontal utilizado como modelo para gerar documento <i>HTML</i> e <i>PDF</i>	65
Figura 50: Identificação dos endereços das variáveis utilizadas para o teste na linha 11002307.	66
Figura 51: Configuração de <i>TAG's</i> no <i>NI OPC Server</i>	67
Figura 52: <i>Quick client OPC</i> com as <i>TAG's</i> da máquina 11002307 sendo monitoradas.....	67
Figura 53: Teste de Comunicação.vi – Diagrama de Blocos.....	68
Figura 54: Testes de Comunicação.vi – Painel Frontal.	68
Figura 55: Comparação dos valores de produção do Turno 3 no <i>VI Monitoramento Online</i> ' com o <i>VI 'Testes de Comunicação'</i>	69
Figura 56: Dados de Produção.vi – Lista dos dados de paradas e produções gravados no banco de dados durante o teste.	70
Figura 57: Consulta à tabela <i>Produção_Parcial</i>	71
Figura 58: Consulta à tabela <i>Paradas_a_Lançar</i>	71

LISTA DE SIGLAS E ACRÔNIMOS

AMR	<i>Advance Manufacturing Research</i>
CLP	Controlador Lógico Programável
CPF	Cadastro de Pessoas Físicas
DAQ	<i>Data Acquisition</i>
ERP	<i>Enterprise Resource Planning</i>
GPiB	<i>General Purpose Interface Bus</i>
HTML	<i>Hypertext Markup Language</i>
IHM	Interface Homem Máquina
LabVIEW	<i>Laboratory Virtual Instrument Engineering Workbench</i>
MES	<i>Manufacturing Execution Systems</i>
OEE	<i>Overall Equipment Effectiveness</i>
OLE	<i>Object Linking and Embedding</i>
OPC	<i>OLE for Process Control</i>
PDF	<i>Portable Document Format</i>
SGBD	Sistema Gerenciador de Banco de Dados
SQL	<i>Structured Query Language</i>
SubVI	<i>Sub Virtual Instrument</i>
TAG	Etiqueta
TCP/IP	<i>Transmission Control Protocol/Internet Protocol</i>
VI	<i>Virtual Instrument</i>

SUMÁRIO

1	INTRODUÇÃO	12
1.1	PROBLEMA	12
1.2	HIPÓTESE	13
1.3	OBJETIVOS	13
1.3.1	Objetivo Geral	13
1.3.2	Objetivos Específicos	13
1.4	JUSTIFICATIVA	14
1.5	METODOLOGIA	14
1.6	ESTRUTURA DO TRABALHO	14
2	REVISÃO DA LITERATURA	16
2.1	INDICADORES DE DESEMPENHO	16
2.2	SISTEMAS <i>ENTERPRISE RESOURCE PLANNING</i> E <i>MANUFACTURING EXECUTION SYSTEMS</i>	17
2.3	BANCO DE DADOS	18
2.4	O PADRÃO OPC	19
2.5	LABVIEW™	20
2.5.1	Instrumentos Virtuais – VI's e SubVI's	21
2.5.2	Painel Frontal e Diagrama de Blocos	21
2.5.3	Controles, Indicadores, Terminais, Nós e Ligações	22
2.5.4	Tipos de dados e suas representações	23
2.5.5	While Loop e For Loop	23
2.5.6	Shift Registers	24
2.5.7	Arranjos e Clusters	25
2.5.8	Estruturas Case, Sequence e Event	27
2.5.9	Variáveis compartilhadas	27
2.5.10	Funções de conectividade a banco de dados	28
3	DESENVOLVIMENTO	30
3.1	FLUXOGRAMA DOS PROCESSOS DE APONTAMENTO	30
3.2	INTERPRETAÇÃO DAS INFORMAÇÕES DO CLP	35
3.2.1	Aquisição dos dados do CLP via OPC	36

3.3	DESENVOLVIMENTO DA VI DE SIMULAÇÃO DOS DADOS DE PRODUÇÃO.....	38
3.4	ESCOLHA DO SGBD.....	43
3.5	DESENVOLVIMENTO DAS VI'S DO SISTEMA DE MONITORAMENTO	45
3.5.1	VI Cadastro	45
3.5.2	VI Segundo Plano	49
3.5.3	VI Monitoramento Online.....	54
3.5.4	VI Dados de Produção	55
3.5.5	VI Relatórios.....	60
4	RESULTADOS	66
4.1	TESTES DE FUNCIONAMENTO DO SISTEMA COMUNICANDO COM O CLP INSTALADO	66
5	CONCLUSÃO.....	72
	REFERÊNCIAS.....	73

1 INTRODUÇÃO

Com o contínuo crescimento das disputas econômicas globalizadas, os setores produtivos buscam melhorar continuamente seus processos e, diante deste quadro, a confiabilidade e a velocidade de processamento das enormes quantidades de informações se torna vital para decisões estratégicas, a fim de assegurar a competitividade no mercado.

Com o objetivo de garantir a confiabilidade e a velocidade das informações, vários setores da indústria têm buscado e desenvolvido sistemas que transportem informações diretamente do chão de fábrica para o nível de gestão de forma automatizada e em tempo real.

Esses sistemas auxiliam na redução de perdas na produtividade, reduzindo o tempo de parada, perdas de matéria prima, reprocesso e até mesmo problemas na qualidade do produto, pois possibilitam mais agilidade nas tomadas de decisões através da análise de tendências desses indicadores, aumentando assim a eficiência global de seus processos.

Em uma fábrica de embalagens plásticas para derivados lácteos (manufatura), as quantidades de embalagens produzidas são detectadas por um sensor posicionado ao final de cada linha de produção, armazenadas em um Controlador Lógico Programável (CLP) e mostradas em uma Interface Homem Máquina (IHM). O apontamento destas quantidades produzidas bem como das paradas de máquinas e os apontamentos das inspeções de qualidade são feitos pelos operadores de produção de modo manual, ao final do turno de trabalho, em uma planilha eletrônica que diariamente é transferida através de um dispositivo de armazenamento portátil (*pen drive*) para o computador do Encarregado de Produção, para finalmente ter seus dados lançados no sistema *Enterprise Resource Planning (ERP – SAP)* e posteriormente ser arquivada em uma pasta específica do servidor.

1.1 PROBLEMA

A confiabilidade dos dados de produção gerados no chão de fábrica. Por se tratar de um processo essencial para a gestão da produção, passar por várias etapas onde várias pessoas podem interferir, torna a análise dos dados pelo gestor da

produção lenta e nem sempre se pode tomar decisões assertivas, tendo em vista a velocidade e a confiabilidade dos dados.

1.2 HIPÓTESE

Com o desenvolvimento do sistema de monitoramento de produção, apontamento e paradas, aproveitando os recursos de *hardware* (sensores, CLP, IHM's e computadores) já instalados na fábrica, utilizando a linguagem de programação em *LabVIEW™*, pode-se melhorar consideravelmente os indicadores de produtividade, obtendo relatórios automaticamente e conseqüentemente, possibilitar o redirecionamento dos operadores envolvidos para outras tarefas.

1.3 OBJETIVOS

1.3.1 *Objetivo Geral*

Desenvolver um sistema de monitoramento de produção, apontamento e paradas automatizado, utilizando a linguagem de programação em *LabVIEW™* para estudo da possibilidade de implantação em uma fábrica de embalagens.

1.3.2 *Objetivos Específicos*

- Definir um fluxograma dos processos de apontamento para o desenvolvimento do sistema.
- Interpretar as informações do CLP instalado para contagem de produção.
- Testar a aquisição dos dados do CLP via *OLE for Process Control (OPC)*.
- Desenvolver um instrumento virtual (*Virtual Instrument - VI*) para simular os dados de produção.
- Definir banco de dados para o armazenamento das informações.
- Desenvolver os *VI's* necessários para o Sistema.
- Testar o funcionamento do sistema comunicando com o CLP instalado.

1.4 JUSTIFICATIVA

Com os dados de produção acessíveis em tempo real e de forma automatizada, pode-se permitir tomadas de decisão mais assertivas pela gestão da produção, pela confiabilidade e velocidade que os obtém. Ao mesmo tempo, os operadores responsáveis pelo apontamento de suas produções podem direcionar sua concentração e força de trabalho nos processos que realmente agregam valor ao produto, melhorando a produtividade e a eficiência dos processos.

A utilização do *LabVIEW*[™] como linguagem de programação e compilador da aplicação a ser desenvolvida, permite maior flexibilidade na programação, tendo em vista a grande integração para comunicação com diversos *hardwares*, bem como a utilização de ferramentas e objetos que permitem a criação padronizada e flexível de uma interface de usuário amigável, ambas características diferenciais desta linguagem.

1.5 METODOLOGIA

Este trabalho possui natureza exploratória, de abordagem qualitativa, onde o procedimento abrangerá pesquisa bibliográfica e desenvolvimento de conceitos assimilados em sala de aula.

1.6 ESTRUTURA DO TRABALHO

Este trabalho tem início com uma revisão de literatura apresentada no Capítulo 2. Neste capítulo são abordados os conceitos de indicadores de produtividade como o OEE, bem como os sistemas de monitoramento de produtividade usados atualmente nas indústrias, como o Sistema *MES*. Ainda neste capítulo, são apresentados os conceitos sobre Banco de Dados, sobre o padrão de comunicação *OPC (OLE for Process Control)*, e também uma abordagem conceitual sobre o *LabVIEW*[™], apresentando as principais funções utilizadas para o desenvolvimento deste trabalho.

O Capítulo 3 apresenta o desenvolvimento do trabalho, abordando a aplicação dos conhecimentos adquiridos nas disciplinas de sistemas supervisórios, automação e controle discreto, informática aplicada, circuitos lógicos, comunicação de dados,

instrumentação e automação da manufatura, durante o curso superior de Tecnologia em Automação Industrial. Essencialmente, este capítulo abordará o passo a passo do desenvolvimento dos VI's, a utilização do banco de dados, a interpretação dos dados do CLP que controla a contagem das embalagens produzidas e a comunicação entre o CLP e o sistema desenvolvido.

No Capítulo 4 são apresentados os resultados obtidos através do desenvolvimento da aplicação, com os testes de comunicação do sistema com o CLP instalado na fábrica de embalagens.

Por fim, no Capítulo 5 são apresentadas as considerações finais e a conclusão deste trabalho.

2 REVISÃO DA LITERATURA

2.1 INDICADORES DE DESEMPENHO

Segundo Nara (2016), por produzirem um elevado número de dados diariamente, as organizações precisam destacar quais são mais relevantes a fim de facilitar a identificação das áreas que necessitam de atenção e melhorias para aperfeiçoamento do processo.

Atualmente, o desenvolvimento da tecnologia nas áreas de comunicações e informática junto com a globalização, exige dos executivos e empresários a necessidade de preparo com técnicas, métodos e instrumentos gerenciais para que se tomem decisões em tempo hábil e com poucas possibilidades de erro. (FISCHMANN e ZILBER, 2000).

Assim, segundo Fischmann e Zilber (2000), os indicadores de desempenho se destacam como instrumentos para auxiliar na definição do planejamento estratégico, e como consequência, podem determinar as estratégias da empresa, permitindo avaliar a eficácia das decisões tomadas. Da mesma forma, Meier (2013 Apud NARA, 2016, p.141) afirma que os indicadores de produção são fundamentais para fazer a seleção das informações do processo produtivo a fim de destacar as mais importantes e que precisam de maior atenção.

Neste sentido, um Indicador de Desempenho muito importante é o Índice de Eficiência Global (*OEE – Overall Equipment Effectiveness*), utilizado para se conhecer o desempenho dos equipamentos. Através do tratamento adequado dos dados, pode-se verificar a evolução deste índice e como ações implementadas nos equipamentos, bem como retrabalhos, refletem no mesmo, permitindo uma análise crítica e detalhada dos processos de produção (MOELLMANN, 2006).

Para o cálculo deste Indicador, Moellmann (2006) afirma:

O cálculo do indicador *OEE* envolve três fatores: disponibilidade, eficiência e qualidade. A disponibilidade é dada pela porcentagem do tempo em que o equipamento é utilizado efetivamente em atividade produtiva. Essa medida aponta todas as perdas por avarias, troca de ferramentas, manutenção preventiva e corretiva, e quaisquer outras paradas de produção. A eficiência da máquina ou da operação, segundo fator do indicador *OEE*, é dada pela relação percentual entre a produção real e a produção padrão, ou ainda, pela relação percentual entre o tempo padrão da operação e seu tempo real de

execução. E a qualidade é medida pela porcentagem de peças produzidas pela máquina que cumpriu todos os requisitos e exigências de qualidade.

Deste modo, conclui se que “O *OEE* mede a eficácia de um equipamento individual. Pode também servir para compararmos o desempenho de equipamentos iguais a trabalharem em condições semelhantes” (SILVA, 2013, p. 11).

2.2 SISTEMAS *ENTERPRISE RESOURCE PLANNING* E *MANUFACTURING EXECUTION SYSTEMS*

Na gestão da produção, dois grandes problemas podem ser citados no que tange a informação. O primeiro está na sua escassez, onde quem faz a gestão não sabe o que realmente está acontecendo, ataca-se somente os grandes problemas e perde-se tempo tentando resolvê-los muitas vezes sem sucesso ou, quando se consegue, vários outros grandes problemas já surgiram. O segundo, está no atraso que estas poucas informações chegam ao nível de gestão, muitas vezes impossibilitando ações eficientes. Em suma, uma gestão moderna de produção necessita de informação correta e em tempo real. (CARDOSO, 2014)

O desenvolvimento de sistemas de gestão como o *Enterprise Resource Planning (ERP)* e o *Manufacturing Execution Systems (MES)* tem ajudado na resolução destes problemas.

Segundo Sistemas ERP (2018), o sistema *ERP*, cuja sigla do inglês traduz se como ‘Planejamento dos Recursos da Empresa’, é representado por atividades que auxiliam na gestão de processos empresariais, gerenciadas por *softwares* ou por pessoas, portanto, um sistema de gestão empresarial.

Stamford (2000, p.1) melhor define:

“O *ERP* é um sistema integrado, que possibilita um fluxo de informações único, contínuo e consistente por toda a empresa sob uma única base de dados. É um instrumento para a melhoria de processos de negócio, tais como produção, compras ou distribuição, orientado por estes processos e não as funções/departamentos da empresa, com informações *on-line* e em tempo real. Possui uma arquitetura aberta, a qual viabiliza operar com diversos sistemas operacionais, banco de dados e plataformas de *hardware*. Desta forma, o *ERP* permite visualizar por completo as transações efetuadas pela empresa, desenhando um amplo cenário de seus processos de negócios”.

No Brasil, a maioria das indústrias de grande porte atualmente possuem um sistema *ERP* implantado e funcional, porém estes sistemas na maioria das vezes não

possui um monitoramento da produção automatizado, onde as informações muito dinâmicas e em larga escala são inseridas no sistema de forma manual, sob o risco não serem confiáveis e chegarem atrasadas, privando a indústria do benefício do sistema *ERP* em sua principal área que é a produção. (CARDOSO, 2014)

Para Cardoso (2014), a promessa de que os sistemas *ERP* gerenciam a indústria toda ocorre apenas do ponto de vista administrativo, pois para estes sistemas a produção é representada por uma caixa preta que utiliza tempo e recursos humanos e financeiros para processar a matéria prima em produto acabado, onde a informação detalhada do processo de produção (quais os problemas, como resolver, onde investir e onde treinar) é desnecessária.

Para preencher a lacuna entre os sistemas *ERP* e o chão de fábrica, Bruce Richardson da *Advance Manufacturing Research (AMR)*, criou em 1990 o termo *MES*, cuja sigla do inglês se traduz por Sistemas de Execução da Manufatura. (MARDEGAN, 2002).

Segundo Cardoso (2014), um sistema *MES* precisa trabalhar junto com o sistema *ERP* na troca de informações, mas como um sistema híbrido (*Software* e *Hardware*) a parte gerando informação de histórico em tempo real, em operação 24 horas por dia, 7 dias por semana.

2.3 BANCO DE DADOS

Com relação a banco de dados:

“Uma das coisas mais importantes em qualquer sistema de gestão é a base de dados, o que significa informação, e muitas vezes isso é negligenciado. Até se faz de alguma forma o *backup*, mas não estamos falando de perder os dados, mas sim da informação que esses dados representam e que irão representar no futuro” (CARDOSO, 2014, p. 20).

Banco de dados ou base de dados, para Meira (2017), pode ser definido como conjuntos de dados com estrutura regular para organizar uma informação. As informações são normalmente agrupadas para um mesmo propósito representando coleções de informações relacionadas a fim de criar um sentido definindo se como uma coleção de dados relacionados. Assim, segundo Meira (2017), dado pode ser entendido como “toda a informação que pode ser armazenada e que apresenta algum

significado dentro do contexto ao qual ele se aplica”, como o CPF de um cliente em um sistema bancário ou o número de matrícula de um aluno em um sistema escolar.

Quanto ao uso do banco de dados, detalhes do armazenamento das informações podem ser ocultadas do usuário, uma característica que é fundamental. Um conjunto de conceitos usados para estruturar o banco de dados fornece o significado que permite essa abstração dos dados e são definidos como modelo de dados. (ELMASRI e NAVATHE, 2005).

Para Meira (2017, p.20):

O modelo relacional é uma teoria matemática criada por Edgar Frank Codd em 1970 para descrever como as bases de dados devem funcionar. O Modelo relacional revelou-se ser o mais flexível e adequado ao solucionar os vários problemas que se colocam no nível da concepção e implementação da base de dados. A estrutura fundamental do modelo relacional é a relação (tabela). Uma relação é constituída por um ou mais atributos (campos) que traduzem o tipo de dados a armazenar. Cada instância do esquema (linha) é chamada de tupla (registro). O modelo relacional implementa estruturas de dados organizadas em relações ou tabelas. Porém, para trabalhar com essas tabelas, algumas restrições precisaram ser impostas para evitar aspectos indesejáveis, como: repetição de informação, incapacidade de representar parte da informação e perda de informação. Essas restrições são: integridade referencial, chaves e integridade de junções de relações. (SILVA, 2009)

Quanto a utilização de um banco de dados, Meira (2017) afirma que podem ter muitas aplicações, desde sistemas simples para controle de estoque em uma loja até sistemas avançados como os bancários e de segurança pública. Os aplicativos de banco de dados (*software* de gerenciamento de banco de dados) podem gerenciar as informações, como pequenas ferramentas (uma agenda por exemplo) até os complexos sistemas empresariais como o desempenho de tarefas de contabilidade e possuem uma interface para o banco de dados. São geralmente chamados de sistema gerenciador de banco de dados (SGBD) como o *Microsoft SQL Server*, *Oracle*, *MySQL*, *PostgreSQL*, *Firebird*, etc.

2.4 O PADRÃO OPC

“No final da década de 80 e início da década de 90 os computadores se tornaram o principal meio a partir do qual o operador passaria a se comunicar com as plantas e processos supervisionados” (NASCIMENTO, 2005, p. 47).

Segundo Nascimento (2005), o crescimento do número de protocolos proprietários dos equipamentos de campo trouxe um novo problema para as

empresas de serviço de automação, que gastavam a maior parte do tempo e recursos no desenvolvimento de *drivers* de comunicação.

Assim, “uma força tarefa foi iniciada com o objetivo de definir um padrão de *drivers* para o *Plug&Play* de equipamentos, permitindo um acesso padronizado aos dados de automação sobre plataforma *Windows*” (GONÇALVES, 2012, p.14).

Surgiu então o *OPC (OLE for Process Control)*,

que usa uma arquitetura cliente-servidor para a troca de informações. Um Servidor *OPC* encapsula as informações de processo e disponibiliza na sua interface. Um *OPC* Cliente conecta no Servidor *OPC* e pode acessar e consumir os dados oferecidos. As aplicações consomem e fornecem dados que podem ser tanto do cliente quanto do servidor (GONÇALVES, 2012, p. 16).

Segundo Fonseca (2002), vários produtos no mercado utilizam para comunicação com os dispositivos de chão de fábrica o *OPC*, o tornando o padrão de comunicação adotado pela indústria e pelo mercado da automação industrial.

2.5 LABVIEW™

Originário da *National Instruments* em 1986 para o *Macintosh* e atualmente também para sistemas operacionais *Windows*, *Linux* e *Solaris*, o *LabVIEW™* (*Laboratory Virtual Instrument Engineering Workbench*) é uma linguagem de programação gráfica aplicada a medições e automação, onde a programação é feita de acordo com o fluxo de dados, oferecendo vantagens para a manipulação e aquisição de dados (SOUZA, 2008).

LabVIEW™ é uma linguagem de programação gráfica que utiliza ícones, em vez de linhas de texto, para criar aplicações. Em contraste às linguagens de programação baseadas em texto, em que instruções determinam a execução do programa, o *LabVIEW™* utiliza programação baseada em fluxo de dados, onde o fluxo dos dados determina a execução. (*NATIONAL INSTRUMENTS*, 2001. p. 16).

Utilizando um conjunto de ferramentas e objetos, se constrói uma interface de usuário conhecida como Painel Frontal, onde se adiciona o código de programação através do Diagrama de Blocos, semelhante a um fluxograma, utilizando funções com representações gráficas para controlar os objetos do Painel Frontal. (*NATIONAL INSTRUMENTS*, 2001).

Chamados de *Virtual Instruments (VI's)*, os programas em *LabVIEW™* não são processados por um interpretador e sim compilados, o que torna sua performance comparável às linguagens de alto nível (SOUZA, 2008).

Os blocos de funções são designados por instrumentos virtuais. Isto é assim porque, em princípio, cada programa (Sub-IV) pode ser usado como sub-programa por qualquer outro ou pode, simplesmente, ser executado isoladamente. Devido à utilização do modelo do fluxo de dados, as chamadas recursivas não são possíveis, podendo-se, no entanto, conseguir esse efeito pela aplicação de algum esforço extra (SOUZA, 2008. p. 2).

Além de estar totalmente integrado a comunicação com diversos *hardwares* como o *GPIB, VXI, PXI, RS-232, RS-485* e dispositivos *DAQ plug-in*, o *LabVIEW™* também dispõe de recursos para conexão com a *internet*, através do *LabVIEW Web Server* e aplicativos como *ActiveX* e redes *TCP/IP* (NATIONAL INSTRUMENTS, 2001).

Para Souza (2008, p. 3), “é confortável programar sem código: mas não devemos esquecer que no *LabVIEW™* é importante planejar muito bem o projeto antes de se passar à realização dos IV's”.

2.5.1 Instrumentos Virtuais – VI's e SubVI's

Virtual Instruments ou *VI's*, são os programas desenvolvidos em *LabVIEW™*, Segundo BOTTARO (2012), recebem esse nome por que sua forma de operação e aparência imitam instrumentos como osciloscópios, multímetros, etc. As *VI's* utilizam uma estrutura modular em um sistema hierárquico, permitindo a criação de programas compostos por outros programas, ou seja, *SubVI's* (Sub-rotinas). Com estas características, pode-se dividir uma aplicação complexa em várias subtarefas simples, construir um *VI* que agrupe cada subtarefa e depois os combinar para a execução de uma tarefa complexa.

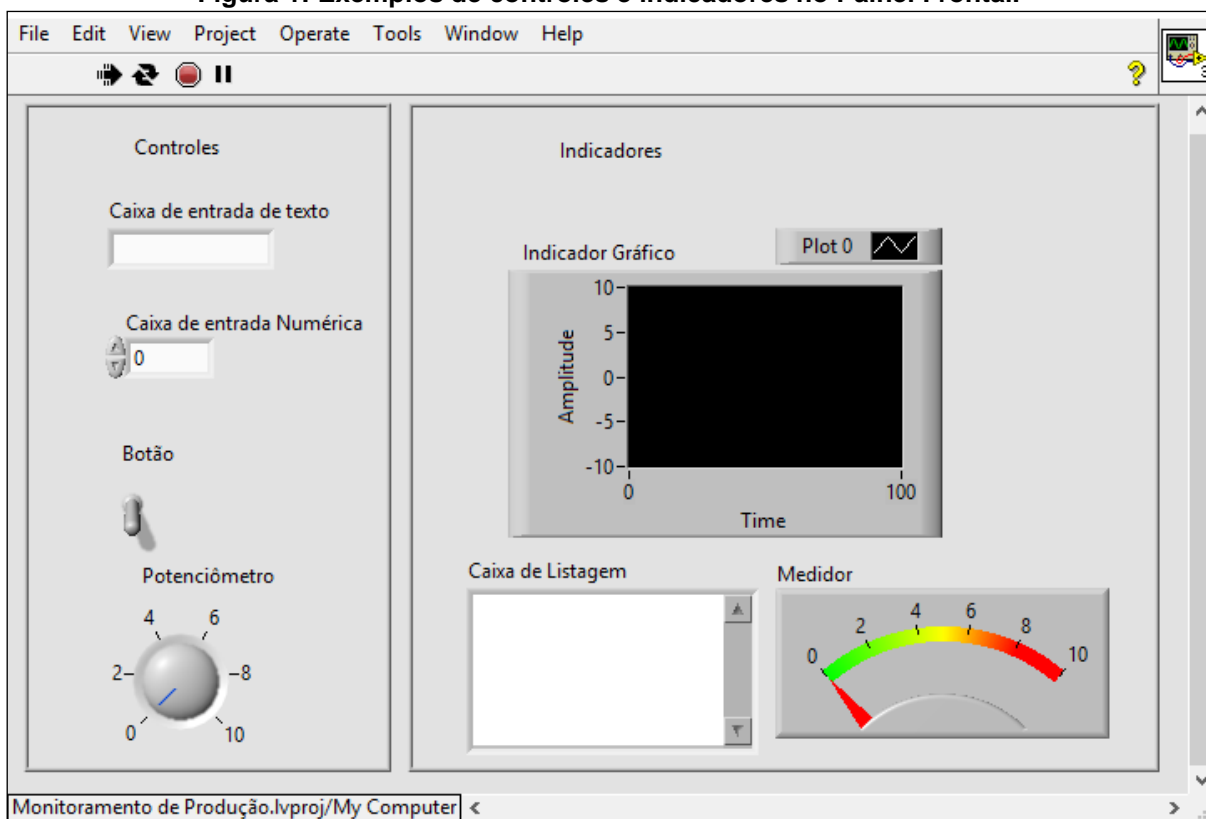
2.5.2 Painel Frontal e Diagrama de Blocos

A interface do *VI* com o usuário é chamada de Painel Frontal. Cada terminal interativo com o usuário (controles e indicadores) é montado neste painel e comunicam com o Diagrama de Blocos, que contém o código fonte gráfico, composto de nós, terminais e ligações. (NATIONAL INSTRUMENTS, 2001)

2.5.3 Controles, Indicadores, Terminais, Nós e Ligações

Controles e indicadores são os terminais interativos de entrada e saída do VI montados no Painel Frontal. Os controles são representados como botões, potenciômetros, caixas de inserção de textos e outros terminais de entrada. Já os indicadores são representados como *displays* numéricos, de texto, gráficos, tabelas, e outros terminais de saída. (NATIONAL INSTRUMENTS, 2001)

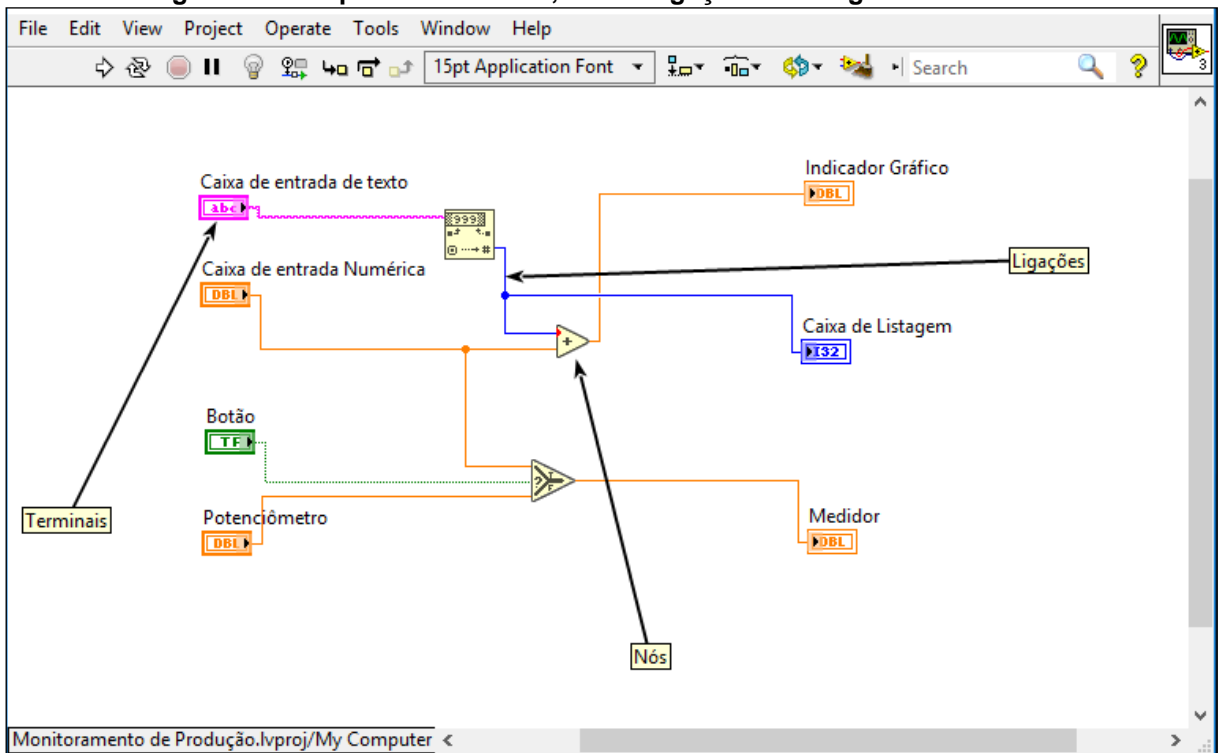
Figura 1: Exemplos de controles e indicadores no Painel Frontal.



Já os Terminais, Nós e Ligações são estruturas montadas no Diagrama de blocos.

Os Terminais são as entradas e saídas, ou seja, a interface entre o painel de controle e o diagrama de blocos, os nós podem ser entendidos como operandos, objetos ou *subVI's* que efetuam as operações com os dados de entrada e estabelecem uma saída. Por fim, as ligações, ou fios, são os caminhos ou transferências de informação de entrada e saída. (BOTTARO, 2012, p21)

Figura 2: Exemplo de Terminais, Nós e Ligações no diagrama de blocos.



2.5.4 Tipos de dados e suas representações

As ligações estabelecem o fluxo de dados entre as variáveis nos VI's e estas variáveis podem ser direcionadas a diversos terminais ou *subVI*'s simultaneamente, conferindo grande versatilidade aos códigos e facilitando a sua elaboração. Estas ligações possuem formas e cores de acordo com o tipo de dados que direcionam (BOTTARO, 2012).

Figura 3: Exemplo de representação das ligações.

Tipo de ligação	Escalar	Arranjo 1D	Arranjo 2D	Cor
Numérico	—————	—————	=====	Laranja (ponto flutuante), Azul (inteiro)
Booleano	Verde
String	~~~~~	~~~~~	~~~~~	Rosa

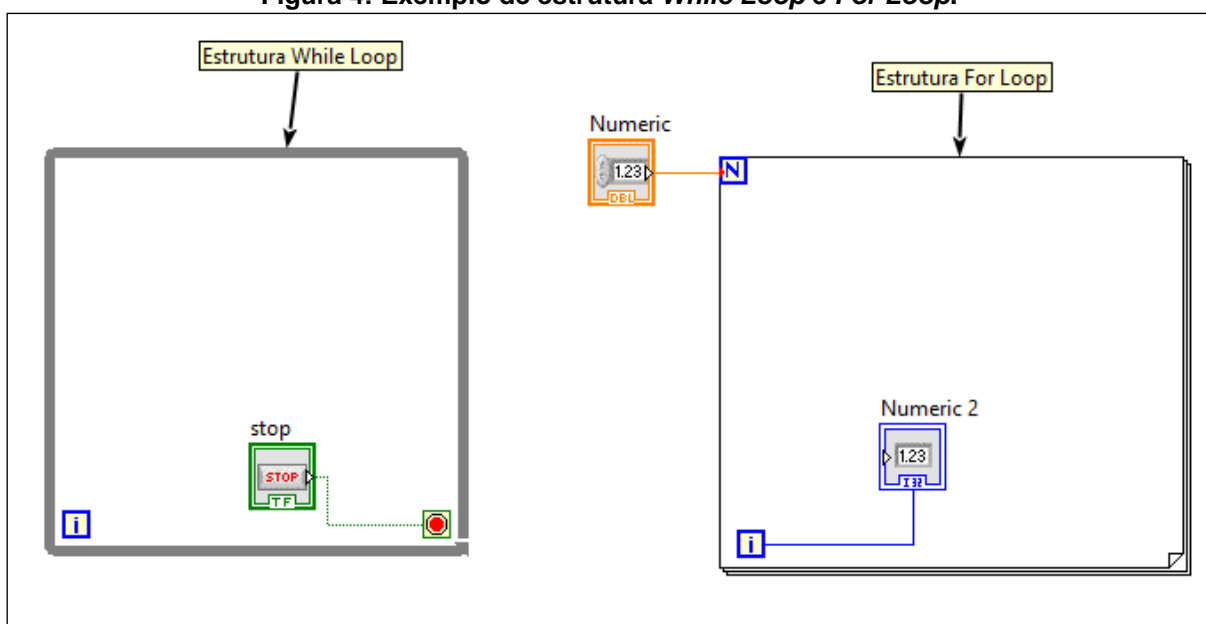
2.5.5 While Loop e For Loop

A estrutura *While Loop* é semelhante a um *Do While* em linguagens de programação baseadas em texto. Esta estrutura executa um sub diagrama até que determinada condição seja alcançada. Um terminal condicional deve receber um valor

booleano da determinada condição para que a execução seja interrompida (NATIONAL INSTRUMENTS, 2001).

Já a estrutura *For Loop* se assemelha a um *For While* em linguagens baseadas em texto. Sua estrutura possui um indicador de número de iterações e um terminal onde é possível programar o número de iterações desejadas através de um controle (BOTTARO, 2012).

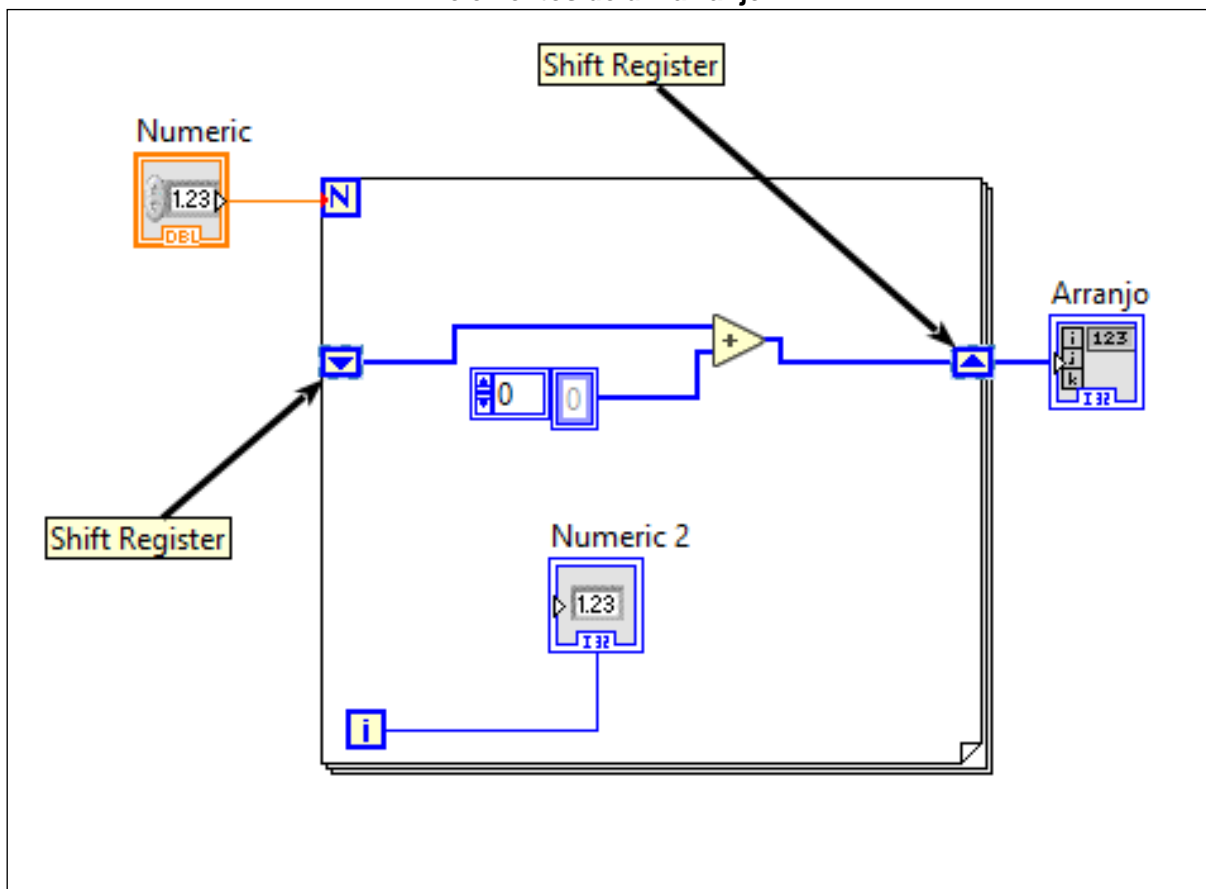
Figura 4: Exemplo de estrutura *While Loop* e *For Loop*.



2.5.6 Shift Registers

Disponível para as estruturas *While* e *For Loop's*, os *Shift Registers* transferem valores de uma iteração do *loop* para a iteração seguinte. Assim, podem ser configurados para registrar os valores de várias iterações, criando-se um terminal adicional para que se possa acessar o valor da iteração anterior. Os *Shift Registers* também podem ser utilizados para executar funções com arranjos e *clusters* (BOTTARO, 2012)

Figura 5: Exemplo de *Shift Register* interagindo com uma estrutura *For Loop* e somando elementos de um arranjo.

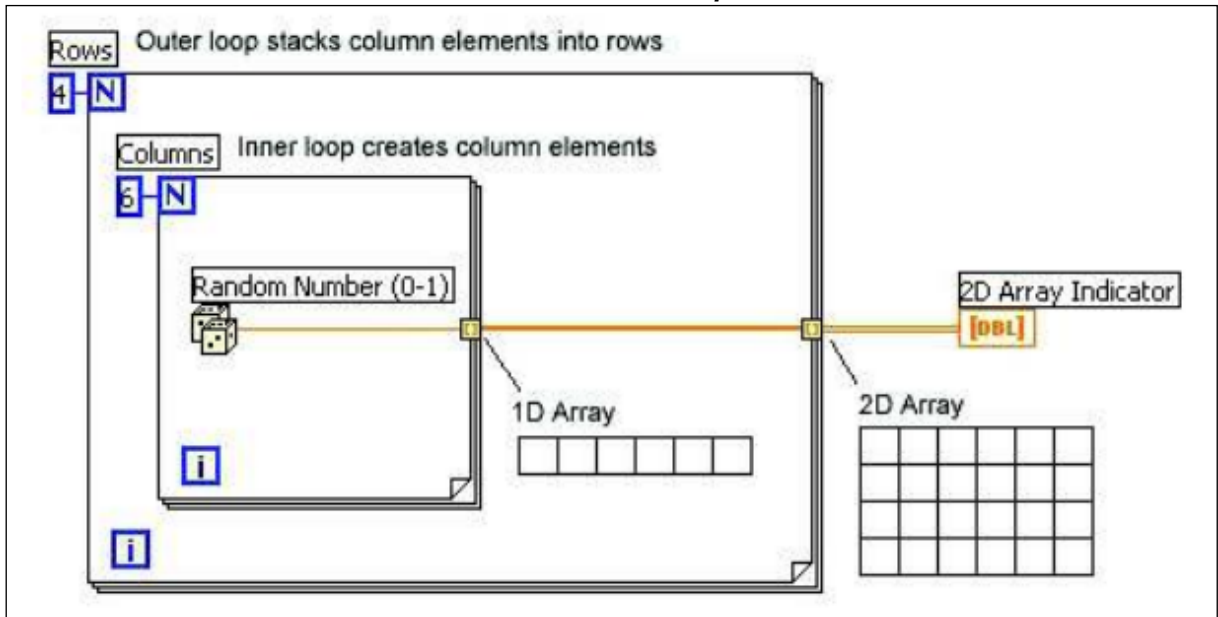


2.5.7 Arranjos e Clusters

Arranjos são agrupamentos de elementos de mesmo tipo. Os elementos de um arranjo são os dados que o constituem, podendo ter uma ou mais dimensões, que são consideradas como o comprimento, a altura ou a profundidade do arranjo (NATIONAL INSTRUMENTS, 2001)

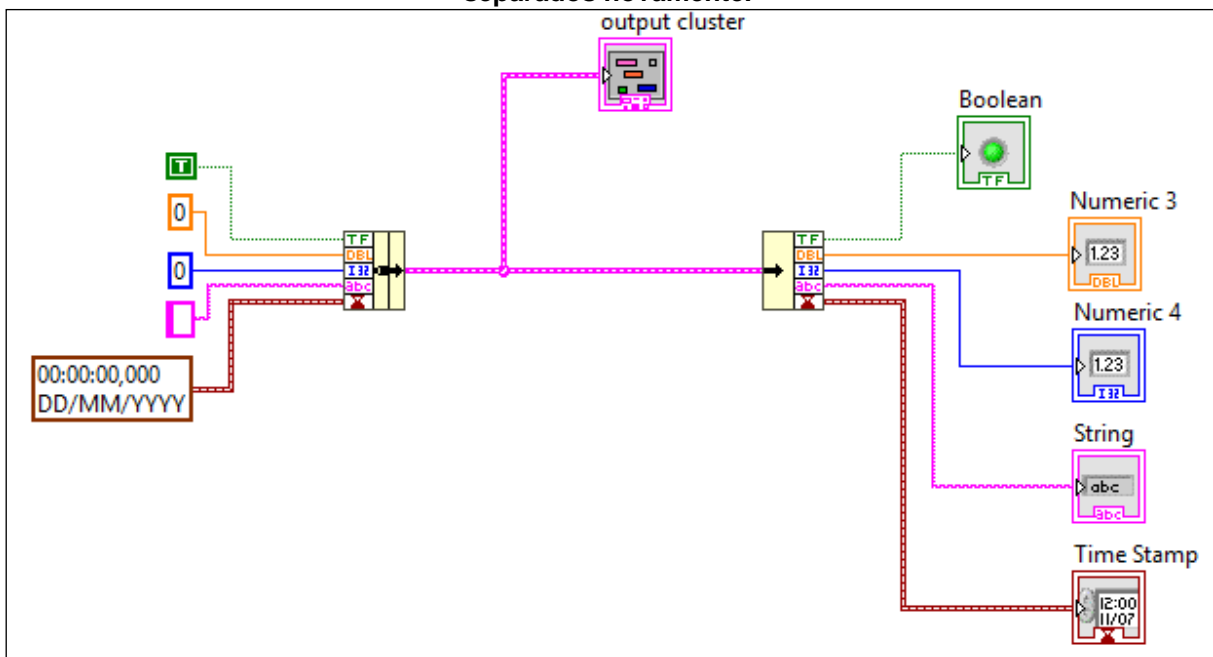
Os elementos do arranjo são ordenados, assim como os nove planetas têm uma ordem estabelecida em relação à distância deles do Sol. Um arranjo utiliza um índice, de forma que você possa acessar rapidamente qualquer elemento específico. O índice é baseado em zero, o que significa que ele está na faixa de 0 até $n - 1$, em que n é o número de elementos no arranjo. Neste exemplo, $n = 9$ para os nove planetas. Dessa forma, o índice varia de 0 a 8. Terra é o terceiro planeta e tem um índice 2. (NATIONAL INSTRUMENTS, 2001, p. 5-2)

Figura 6: Exemplo de Arranjo de duas dimensões sendo criado através de iterações das estruturas *For Loop*.



Um conjunto de dados de tipos diferentes constituem um Cluster. Os clusters tem o objetivo principal de reduzir o número de ligações e terminais necessários em um *VI* ou *SubVI* (BOTTARO, 2012).

Figura 7: Exemplo de junção de diferentes tipos de dados em um *Cluster*, que depois são separados novamente.



2.5.8 Estruturas Case, Sequence e Event

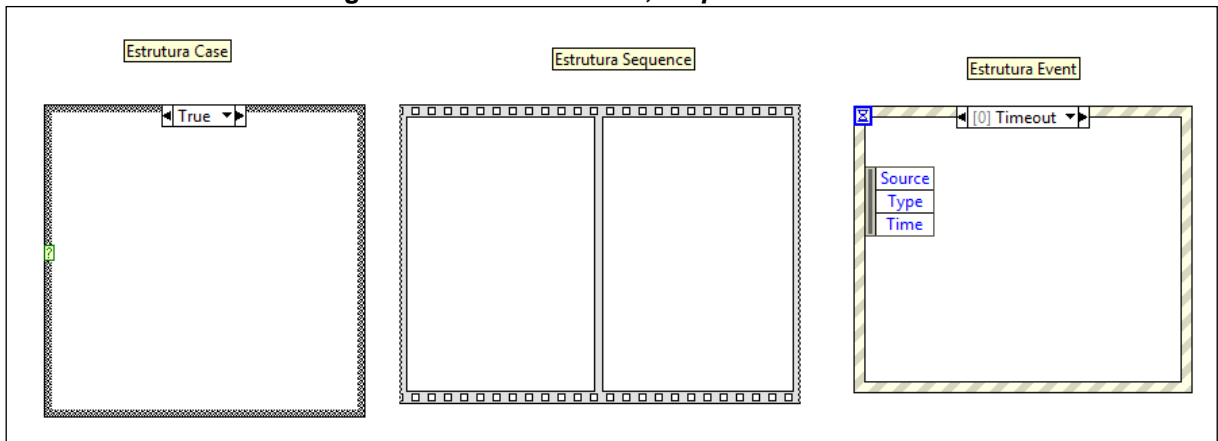
Semelhante às declarações condicionais *If... Then... Else* em linguagens de programação baseadas em texto, a estrutura *Case* executa apenas uma instrução por vez de acordo com o valor do terminal de entrada (*NATIONAL INSTRUMENTS*, 2001).

Uma estrutura *Sequence*, por sua vez,

é baseada em um conjunto de subdiagramas, ou frames, que executam instruções sequenciais, iniciando pelo *frame* de índice "0" e seguido pelos *frames* 1, 2, etc. Este tipo de estrutura não completa uma execução ou retorna dados até que o último *frame* seja executado. Desta forma, esta estrutura deve ser utilizada quando a interdependência de dados não existe durante sua execução. Sua maior aplicação é para a geração de rotinas de inicialização ou preparação anterior a execução da função específica de processamento de dados de um programa. (BOTTARO, 2012, p.63).

Já uma estrutura *Event*, interrompe a execução do VI até que um evento específico ocorra ou quando o tempo programado para esperar que ocorra chegar a zero. Quando o evento ocorre, o sub diagrama executa as funções nele inseridas. Este diagrama é muito útil quando se desenvolve painéis frontais que dependem de ações específicas do usuário para executar suas funções, como uma máquina de estados (*NATIONAL INSTRUMENTS*, 2001).

Figura 8: Estruturas Case, Sequence e Event.

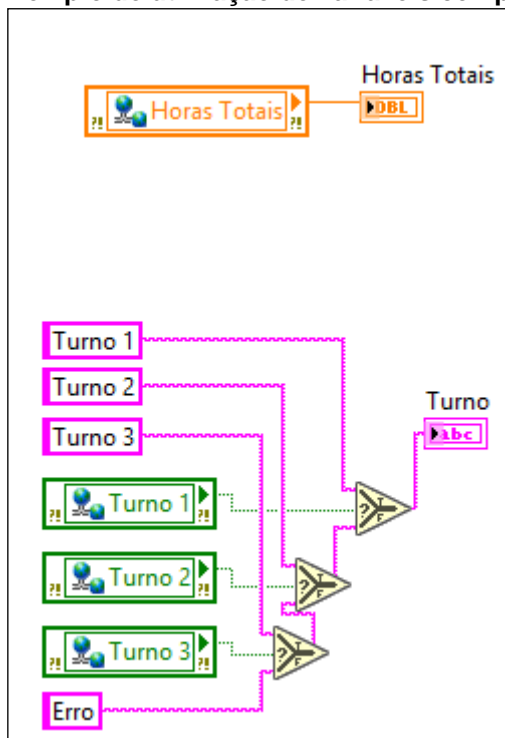


2.5.9 Variáveis compartilhadas

Assim como nas linguagens de programação baseadas em texto, as variáveis compartilhadas permitem o transporte de informações tanto para diversas áreas dentro da aplicação como também permitem interações com outras aplicações. São

de suma importância para o desenvolvimento de aplicações de aquisição de dados via *OPC*, onde podem ser utilizadas como *TAG's*.

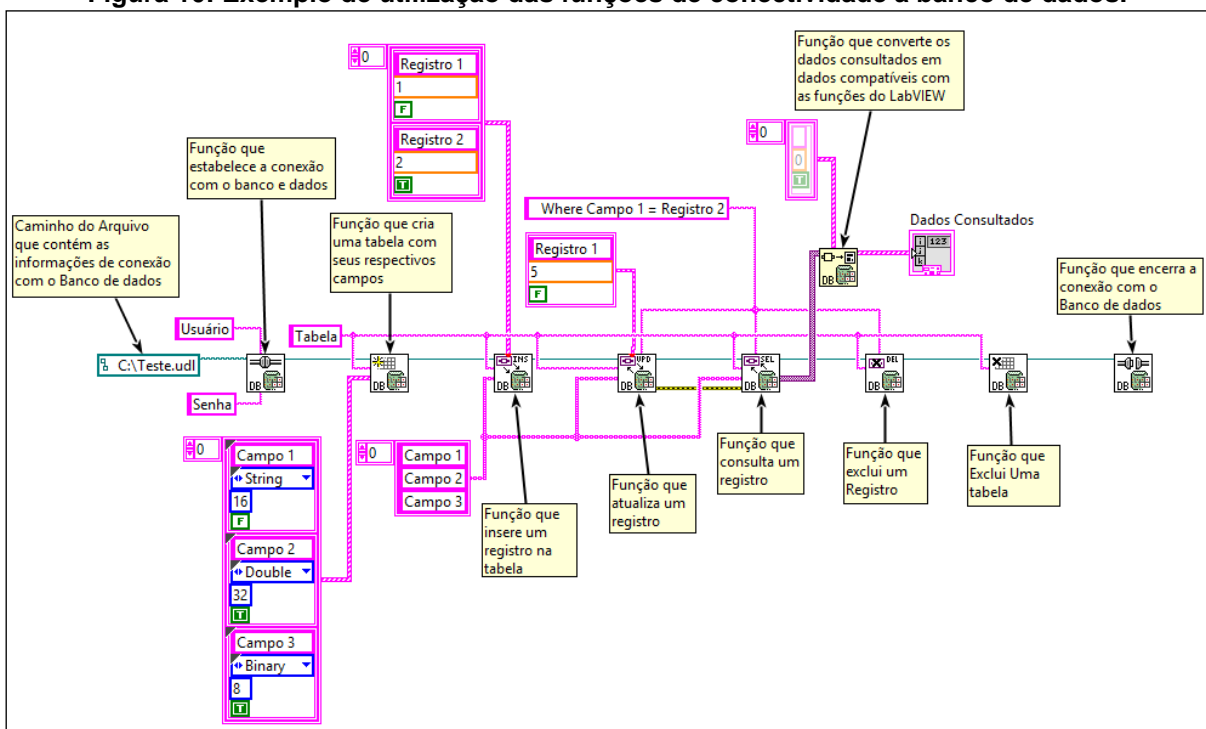
Figura 9: Exemplo de utilização de variáveis compartilhadas.



2.5.10 Funções de conectividade a banco de dados

As funções de conectividade a banco de dados do *LabVIEW™* permitem o desenvolvimento de aplicações que necessitam se comunicar com bases de dados externas, como *Access*, *SQL*, *Oracle*, *MySQL*, etc. Estas funções funcionam como as instruções de um SGBD, que permitem criar ou excluir tabelas, campos e registros, bem como executar consultas e alterar registros de forma prática e automatizada.

Figura 10: Exemplo de utilização das funções de conectividade a banco de dados.



3 DESENVOLVIMENTO

No desenvolvimento deste trabalho, é abordado de forma detalhada o passo a passo do desenvolvimento dos VI's, a utilização do banco de dados e a interpretação dos dados do CLP que controla a contagem das embalagens produzidas via OPC.

3.1 FLUXOGRAMA DOS PROCESSOS DE APONTAMENTO

A forma atual do processo de apontamento na planta de produção se dá de uma maneira lenta e muitas vezes geram dados inconsistentes, o que acaba gerando retrabalho. Para se entender como é esse processo, é preciso compreender como são gerados os dados que serão apontados no sistema *SAP ERP*.

A fábrica funciona de segunda a sábado vinte e quatro horas por dia, o qual é dividido em três turnos de trabalho de oito horas cada. O primeiro turno (Turno 1) inicia as 07:30 e termina as 15:30, seguido do segundo turno (Turno 2) até as 23:30. Das 23:30 até as 07:30, decorre o terceiro turno (Turno 3).

Ao todo são nove linhas de produção que fabricam as embalagens, cada uma com seus respectivos contadores de quantidade total e quantidade boa. A quantidade total é a quantidade de embalagens que a máquina principal produz, em seu respectivo ciclo ou cadência. A quantidade boa é a quantidade de embalagens que passou pelas etapas de acabamento e testes e foi estocada pronta para o envio ao cliente. A subtração destas duas quantidades resulta na quantidade de refugo, ou seja, a perda de embalagens na linha de produção.

Ao final de cada turno, os operadores das linhas de produção precisam anotar os valores de quantidade total, quantidade boa e quantidade de refugo de todas as linhas em uma planilha. Esses valores são registrados em uma IHM.

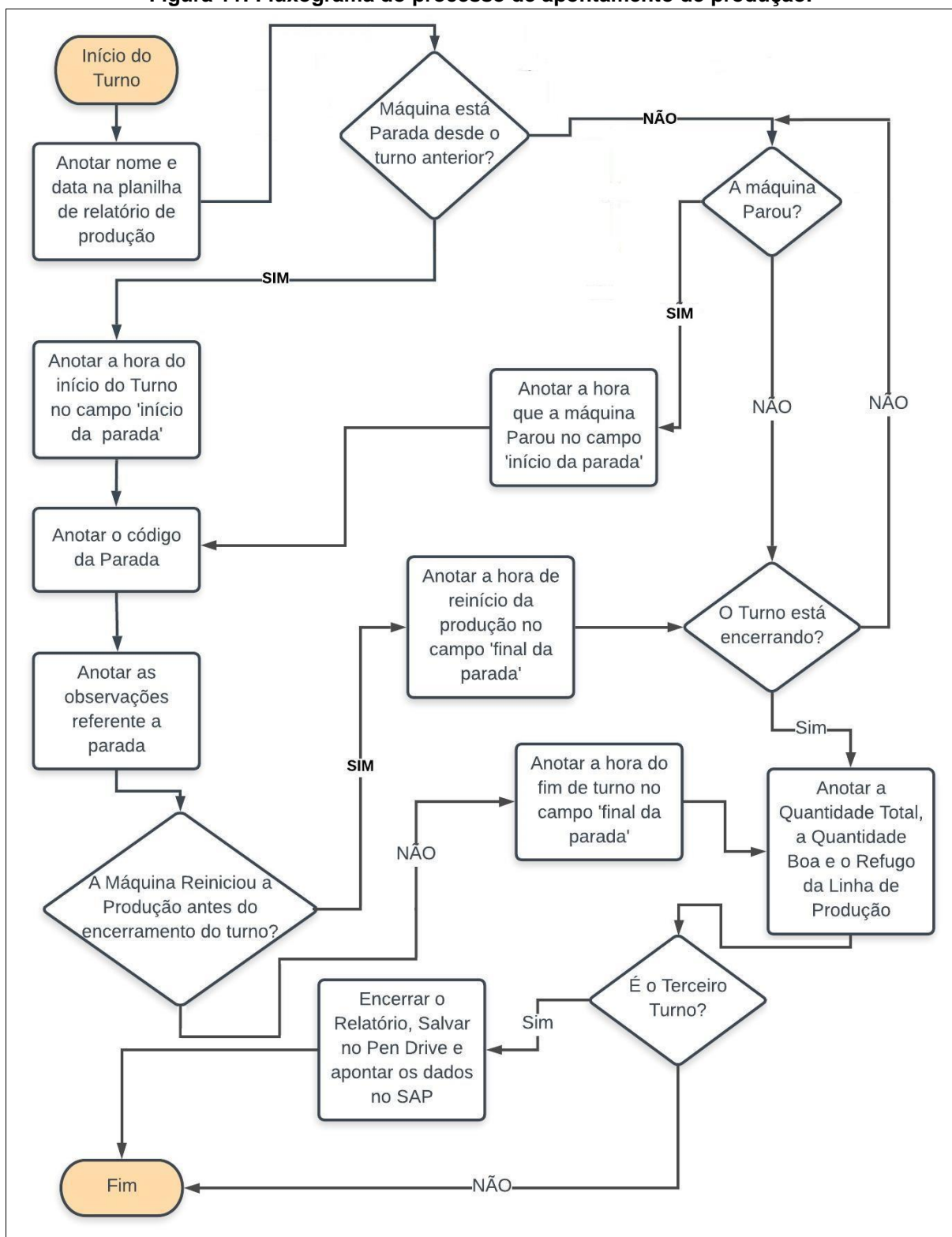
Além das quantidades de embalagens, os operadores precisam registrar na planilha as paradas de máquina que ocorreram durante o turno. Para que se tenha uma informação consistente, é necessário registrar a data e a hora que a máquina parou, data e a hora que a máquina reiniciou a produção, o código do motivo da parada e as observações da parada. Em alguns casos, a parada de máquina se estende para o próximo turno ou para o próximo dia. Nesses casos, o operador deve anotar o horário do início da parada referente a seu turno e anotar o horário final da

parada coincidindo com o horário de encerramento do turno, então o operador do próximo turno deve anotar o horário de início da parada como o horário de início do seu turno e encerrar a parada ao reinício da produção da máquina.

Ao final de cada terceiro turno, as anotações da planilha que recebe os dados de produção são conferidas e a planilha é salva em um *pen drive*, e posteriormente é salva em um diretório específico do computador do encarregado de produção, que irá então analisar e transcrever as informações para o sistema *SAP ERP*.

A Figura 11 mostra o fluxograma do processo de apontamento de produção, com as etapas executadas pelos operadores.

Figura 11: Fluxograma do processo de apontamento de produção.

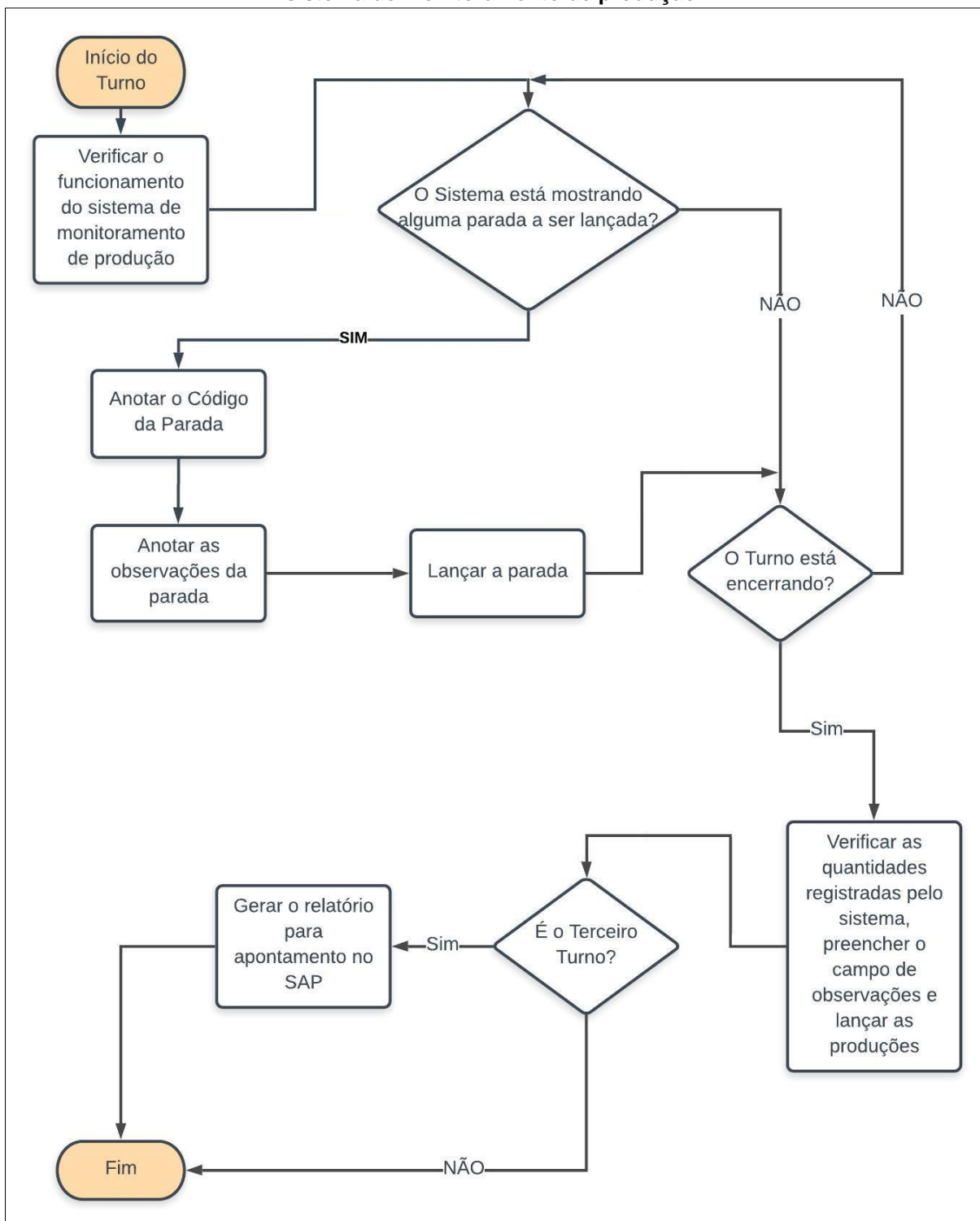


Como todo este processo é executado de forma manual, o risco de inconsistência de dados e até mesmo perda de informações é relativamente alto. Além disso, o tempo que se leva para registrar todas essas informações, de todas as linhas de produção, poderia ser empregado em outras atividades.

O desenvolvimento de um sistema de monitoramento de produção poderia eliminar os riscos de perda de informação bem como aumentar a produtividade dos operadores, tendo em vista o aproveitamento do tempo gasto com o processo de apontamento em outras atividades.

A Figura 12 mostra o fluxograma do processo de apontamento desejado com o desenvolvimento do sistema de monitoramento de produção, onde já se percebe uma simplificação significativa das etapas do processo executadas pelos operadores.

Figura 12: Fluxograma de apontamento de produção desejado a partir do desenvolvimento do sistema de monitoramento de produção.

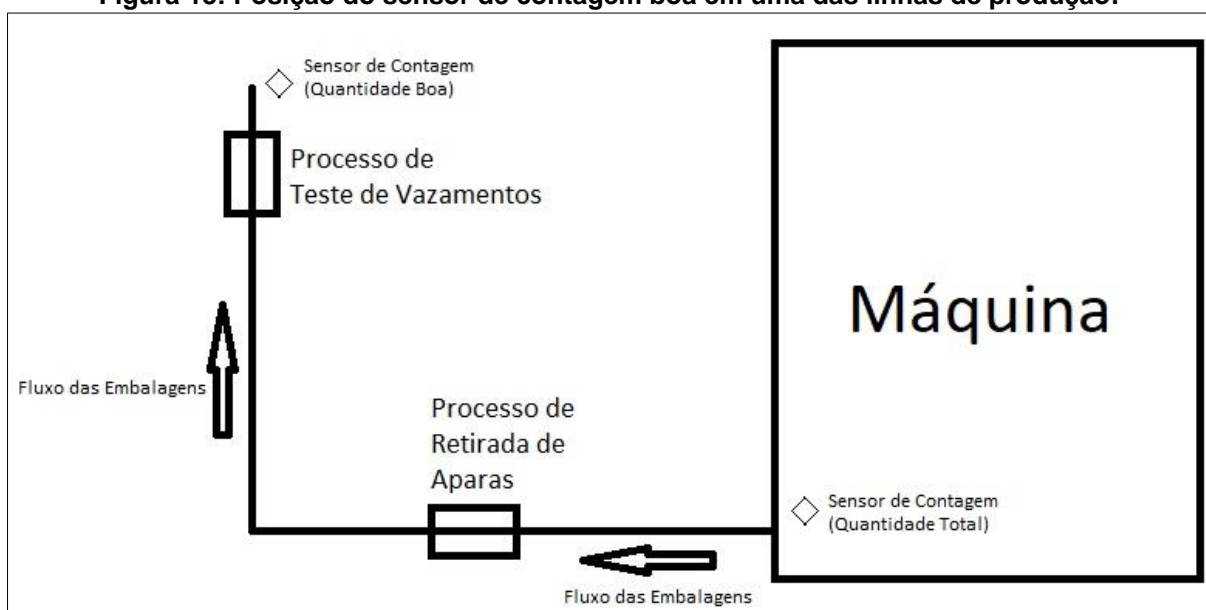


3.2 INTERPRETAÇÃO DAS INFORMAÇÕES DO CLP

O sistema de monitoramento de produção precisa coletar dados do CLP que atualmente controla, além de outros processos, as contagens das linhas de produção. Para isso, é necessário o entendimento da lógica do CLP a fim de selecionar as variáveis corretas que irão comunicar com o sistema.

Através de sensores posicionados logo na saída das máquinas, o CLP armazena a quantidade de embalagens produzidas em contadores instantâneos distintos para cada máquina, que representam a quantidade total de embalagens produzidas. Da mesma maneira, a quantidade boa é contabilizada através de um sensor posicionado no tubo transportador ao final de cada linha de produção, e o valor desta contagem é armazenado em contadores instantâneos distintos para cada linha (Figura 13).

Figura 13: Posição do sensor de contagem boa em uma das linhas de produção.



Estas quantidades armazenadas nas variáveis do CLP, vão se acumulando ao longo do tempo em que cada embalagem passa a frente do respectivo sensor, ao passo que, ao final de um turno de produção um pulso configurado para ser gerado em um horário pré-determinado (horário de troca de turno) ocorre, transferindo esses valores para as variáveis de armazenamento e zerando o contador para iniciar a contagem do próximo turno. A Figura 14 mostra algumas dessas variáveis que serão utilizadas no sistema de monitoramento de produção.

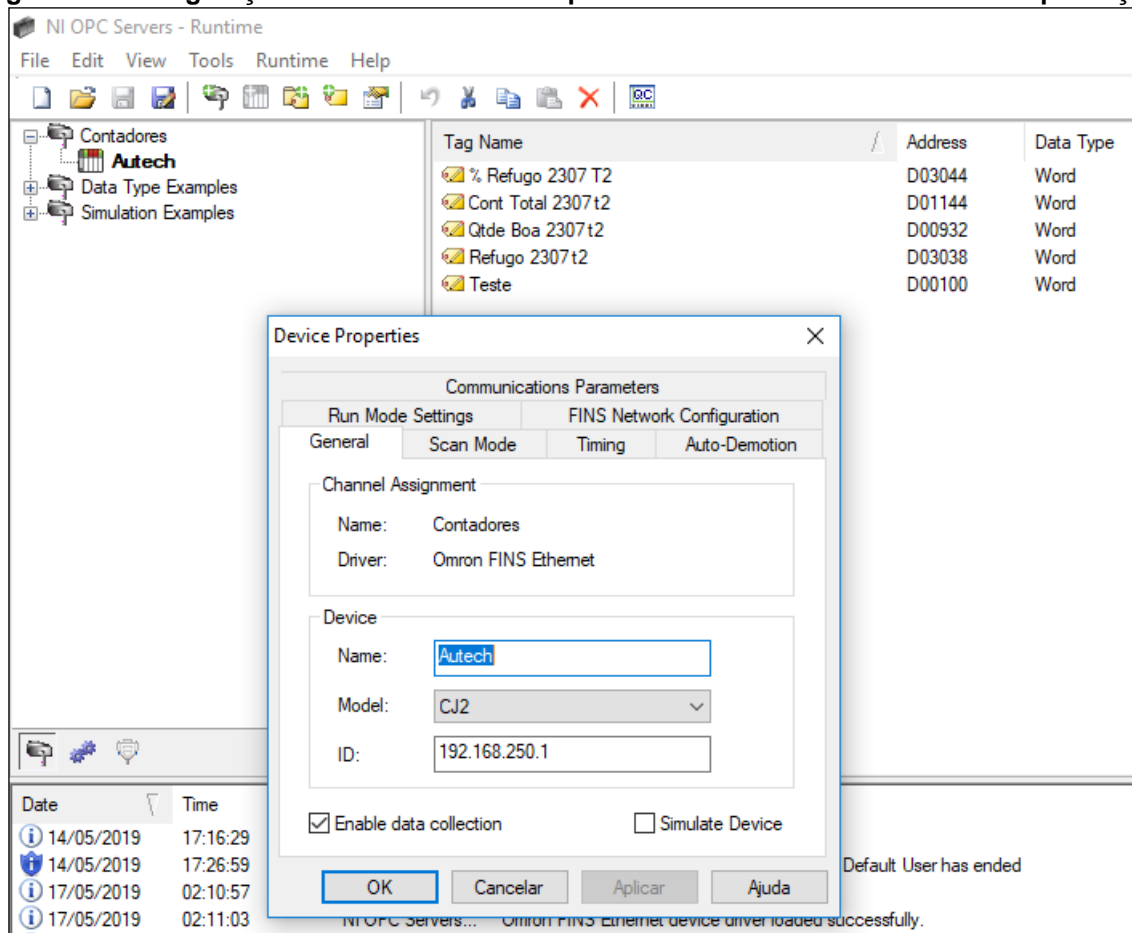
Figura 14: Variáveis utilizadas para o monitoramento de produção.

CONTADOR DE FRASCOS 180G SOPRADORA 11 FRASCODUTO 1			
CONT_180G_SOP11			
CONTADOR_FRASCO			
P_On	(BOOL)	EN	LUIS RAFAEL (BOOL) ENO
Always ON Flag			
37.01	(BOOL)	HABILITA 18...	(DWORD) D312
	HAB_ENVIO_	BRF	CONTADOR_T...
		URNO_1_BRF	
W210.01	(BOOL)	HAB_ENVIO_L	(DWORD) D314
	HAB_ENVIO_L	OGOPLAST	CONTADOR_T...
		URNO_2_BRF	
35.06	(BOOL)	SENSOR CO...	(DWORD) D316
	SENSOR_CON	TADOR	CONTADOR_T...
		URNO_3_BRF	
D50	(WORD)	TROCA DE T...	(DWORD) D318
	TURNO_TRAB		CONTADOR I...
		URNO_1_LO...	
P_On	(BOOL)	Always ON ...	(DWORD) D320
	HAB_CONTAG	EM	CONTADOR I...
		URNO_2_LO...	
W6.05	(BOOL)	RESET VALO...	(DWORD) D322
	RSET_PARCIA	L	CONTADOR I...
		URNO_3_LO...	
D300	(DWORD)	CONTADOR ...	D300
	CONTAGEM_A	CONTAGEM_A	CONTADOR ...
	CUM_FRAS...	CUM_FRAS...	
D302	(DWORD)	CONTADOR ...	D302
	CONTAGEM_A	CONTAGEM_A	CONTADOR ...
	CUM_FRAS...	CUM_FRAS...	
D304	(DWORD)	CONTADOR ...	D304
	CONTAGEM_A	CONTAGEM_A	CONTADOR ...
	CUM_FRAS...	CUM_FRAS...	

3.2.1 Aquisição dos dados do CLP via OPC

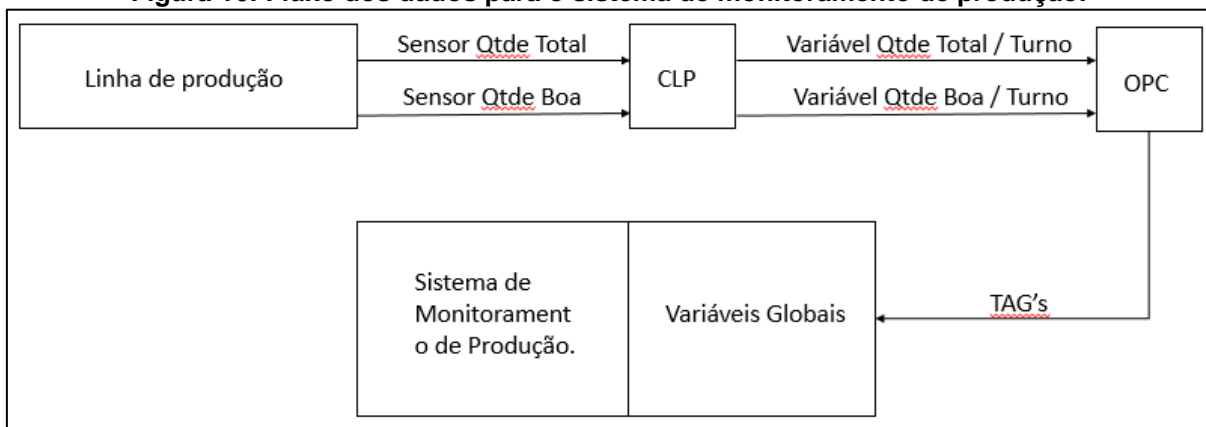
A transferência dos valores das variáveis do CLP para o sistema de monitoramento de produção é realizada através da comunicação entre eles via *OPC*. O *OPC* utilizado para este sistema é o *NI OPC Servers 2016*, da *National Instruments*, a mesma desenvolvedora da plataforma *LabVIEW™*. A Figura 15 mostra alguns detalhes de configuração do *OPC Server*, como as configurações de dispositivo para comunicação com o CLP e algumas *TAG's* criadas para monitorar os valores das variáveis.

Figura 15: Configurações NI OPC Server 2016 para o sistema de monitoramento de produção.



Através destas TAG's o sistema de monitoramento de produção irá armazenar os valores em variáveis compartilhadas específicas para utilização em diversos setores dos blocos de funções dos VI's. A Figura 16 detalha o fluxo dos dados do sistema.

Figura 16: Fluxo dos dados para o sistema de monitoramento de produção.

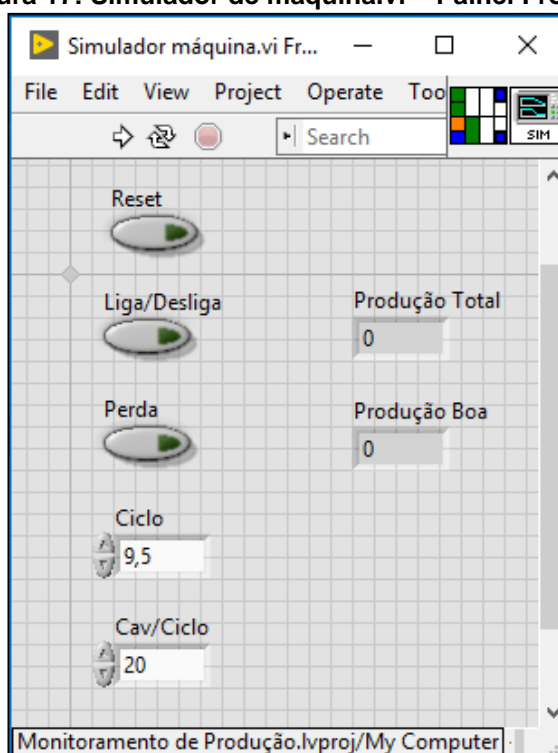


3.3 DESENVOLVIMENTO DA VI DE SIMULAÇÃO DOS DADOS DE PRODUÇÃO

Para o desenvolvimento dos VI's da aplicação, por não haver a possibilidade de conectar com o CLP frequentemente, foi necessário desenvolver um VI que simulasse o comportamento das linhas em produção, gerando quantidades de embalagens produzidas, bem como sinais de paradas e de retomadas de produção. Tendo em vista esta necessidade, foi desenvolvido um VI que simula a geração de dados de uma linha de produção distinta, através de comandos booleanos e numéricos, e outro VI que agrupa o anterior para gerar os dados de todo o parque de máquinas que se pretende monitorar.

A Figura 17 mostra o painel frontal do VI que simula uma linha de produção distinta.

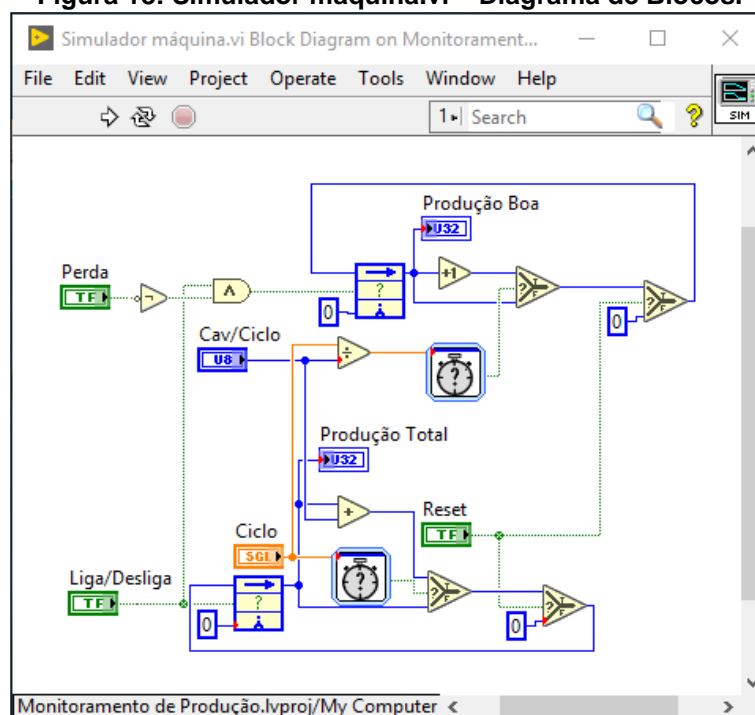
Figura 17: Simulador de máquina.vi – Painel Frontal.



Este VI foi configurado como *SubVI*, onde as entradas e saídas são representadas como controles e indicadores.

Na Figura 18, está representado o diagrama de blocos deste mesmo VI:

Figura 18: Simulador máquina.vi – Diagrama de Blocos.

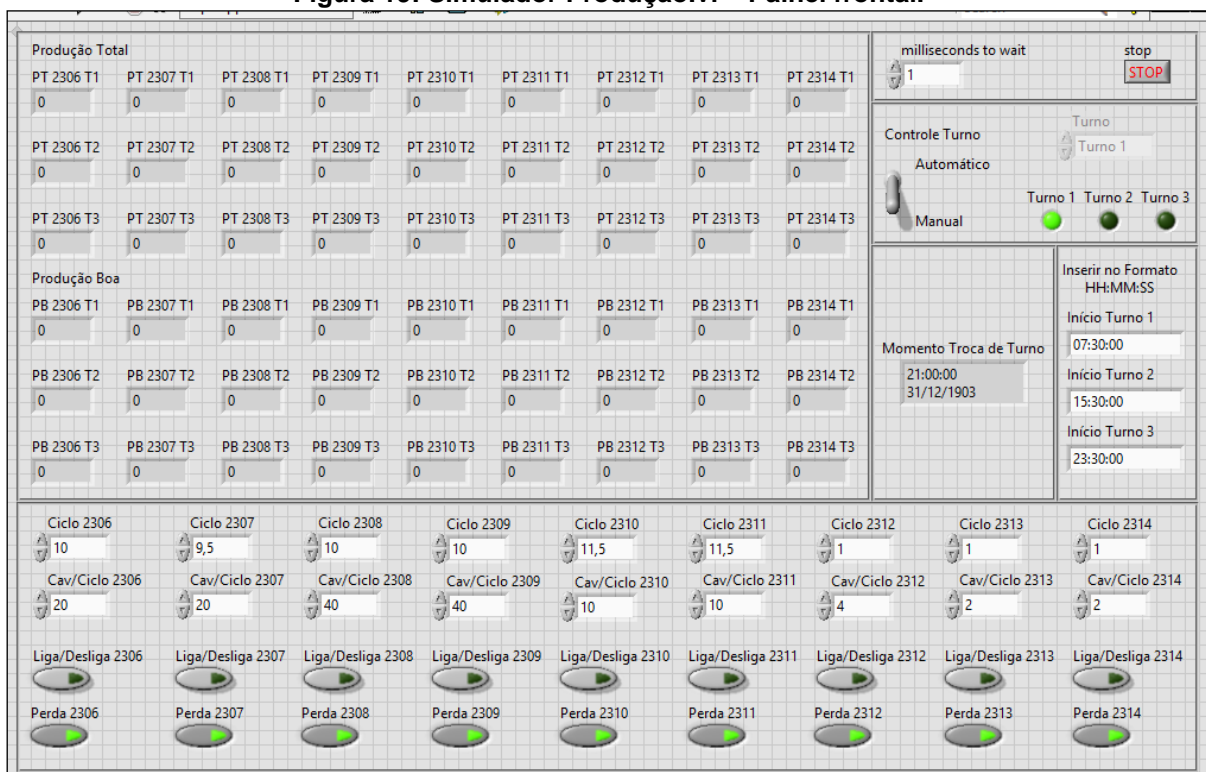


Ao alterar o controle ‘Liga/Desliga’ para verdadeiro, inicia-se o fluxo de dados ciclicamente, com períodos controlados pelos temporizadores. Estes temporizadores alteram as saídas booleanas para verdadeiro quando atingem o tempo programado pelos controladores a eles conectados (‘Ciclo’ e razão entre ‘Ciclo’ e ‘Cav/Ciclo’). Esse processo atualiza os indicadores ‘Produção Total’ e ‘Produção Boa’ de acordo com o tempo de ciclo ajustado e o número de peças por ciclo (Cav/Ciclo). Ao alterar o controle ‘Reset’ para falso, os valores de ambos os indicadores assumem o valor ‘zero’. Ao se alterar o controle ‘Perda’ para verdadeiro, o indicador ‘Produção Boa’ para de atualizar e seu valor se mantém até que o controle se altere novamente para falso. Desta forma, este VI simula o estado de uma máquina, ‘ligada’ ou ‘desligada’, bem como a quantidade total de unidades produzidas de acordo com seu ciclo e a quantidade de perda na linha de produção.

Para simular a geração de dados de toda a planta, foi desenvolvido o um VI que integra nove linhas de produção distintas através da utilização do *SubVI* citado acima. Este VI precisa gerar os dados de quantidade de produção, perdas, estado das máquinas (ligadas ou desligadas) e, também, o momento no tempo (turno de trabalho) que estão gerando estes dados.

A Figura 19 mostra o painel frontal desenvolvido para este VI:

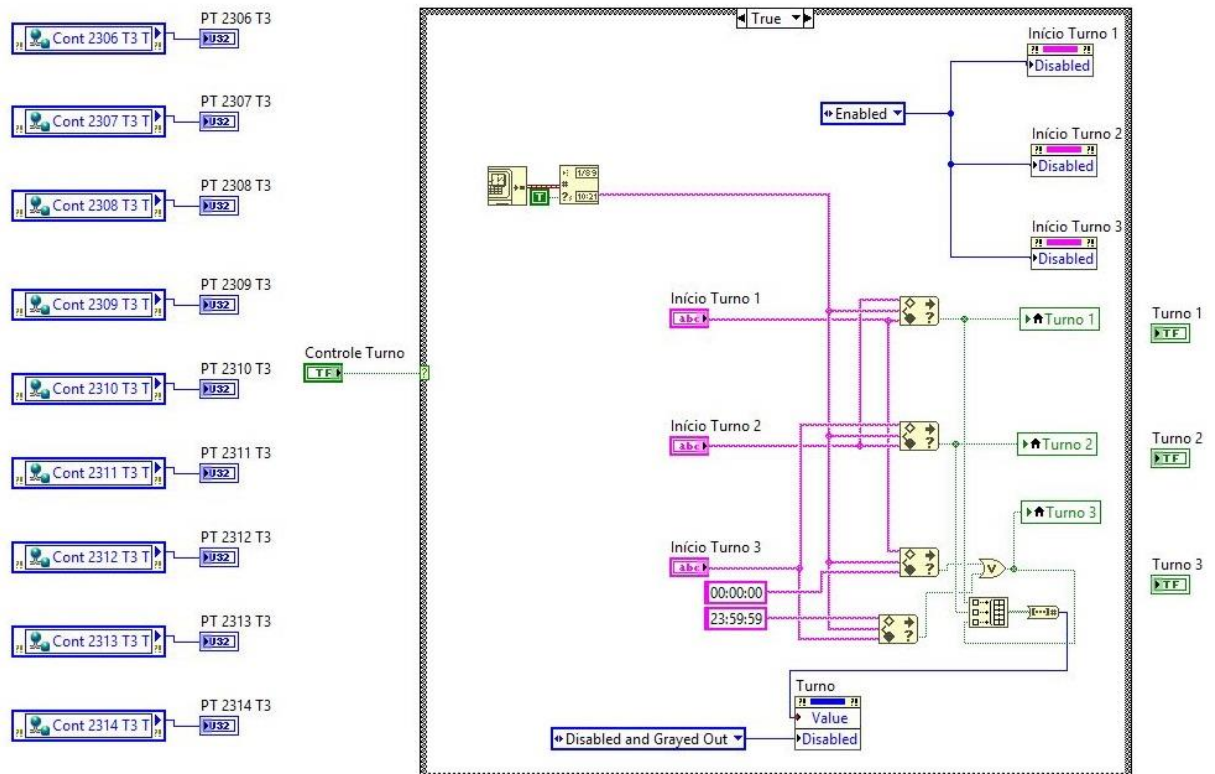
Figura 19: Simulador Produção.vi – Painel frontal.



Neste painel, estão inseridos os indicadores que receberão os valores simulados das quantidades totais de unidades produzidas por cada uma das nove máquinas da planta, em seus respectivos turnos, bem como as suas respectivas quantidades deduzidas das perdas também simuladas ('Quantidade Boa'). Também estão inseridos neste painel os controles que simulam as paradas das máquinas, bem como os controles que simulam as perdas. As características de ciclo e unidades por ciclo, individuais de cada máquina, também são simuladas através dos controles numéricos. Por fim, estão inseridos também controles e indicadores que simulam o momento no tempo em que os dados estão sendo gerados (turnos).

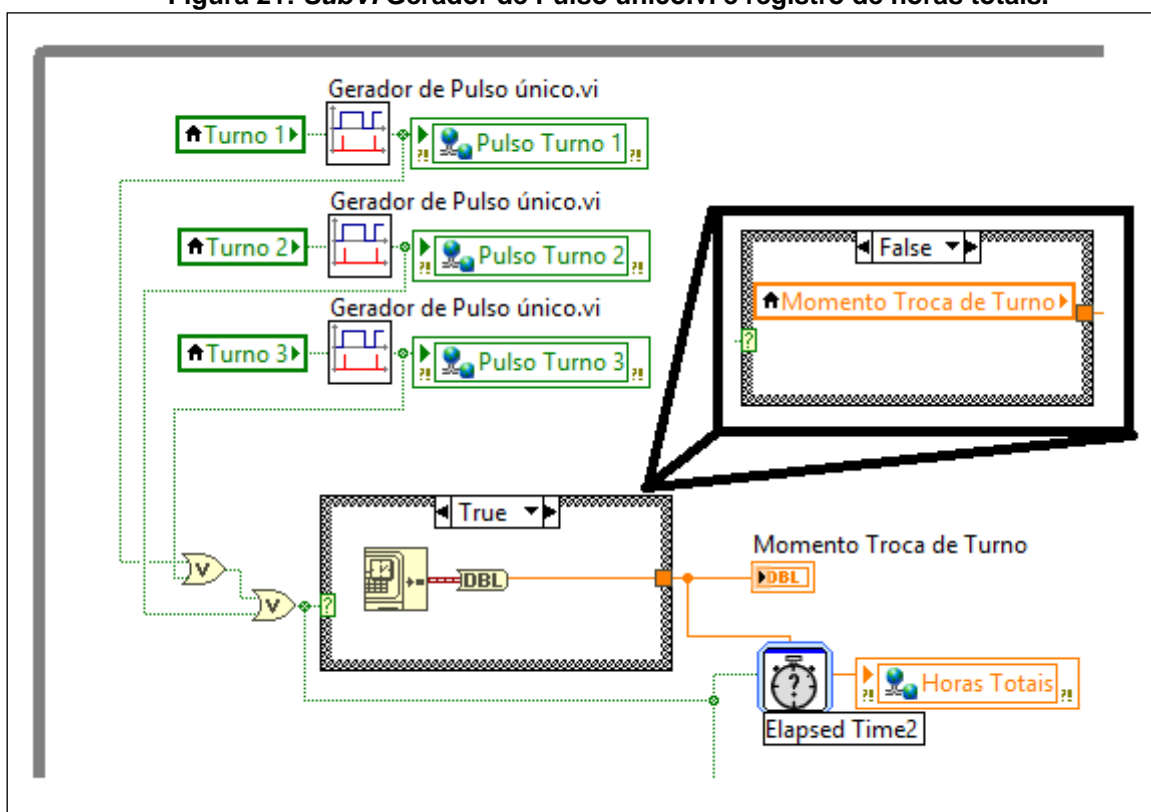
O os componentes principais do diagrama de blocos deste VI estão representados na Figura 20:

Figura 20: Simulador Produção.vi – Diagrama de Blocos.



Além do *SubVI* criado para simular individualmente uma linha de produção, foram criados mais dois *SubVI's* para implementação do simulador. Um deles é o gerador de pulso único (Figura 21), que ao ter sua entrada booleana alterada para verdadeiro, gera em sua saída apenas um pulso. Este *SubVI* é utilizado para gerar um pulso que representa o momento em que ocorre a troca de turno de trabalho.

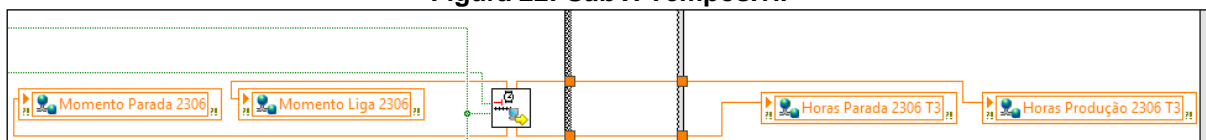
Figura 21: SubVI Gerador de Pulso único.vi e registro de horas totais.



Na Figura 21 também pode-se observar que o pulso gerado alimenta a variável compartilhada correspondente ao turno e, também, gera a condição para que um temporizador registre na variável global o tempo que durou o turno anterior, ou seja, a quantidade de horas transcorridas desde o último pulso gerado. Esta variável compartilhada será utilizada nos VI's de monitoramento de produção.

O outro SubVI citado acima foi desenvolvido para simular os dados de tempo de máquina parada, tempo de máquina ligada, momento em que a máquina parou e momento em que a máquina ligou. A Figura 22 mostra a aplicação deste SubVI no diagrama de blocos:

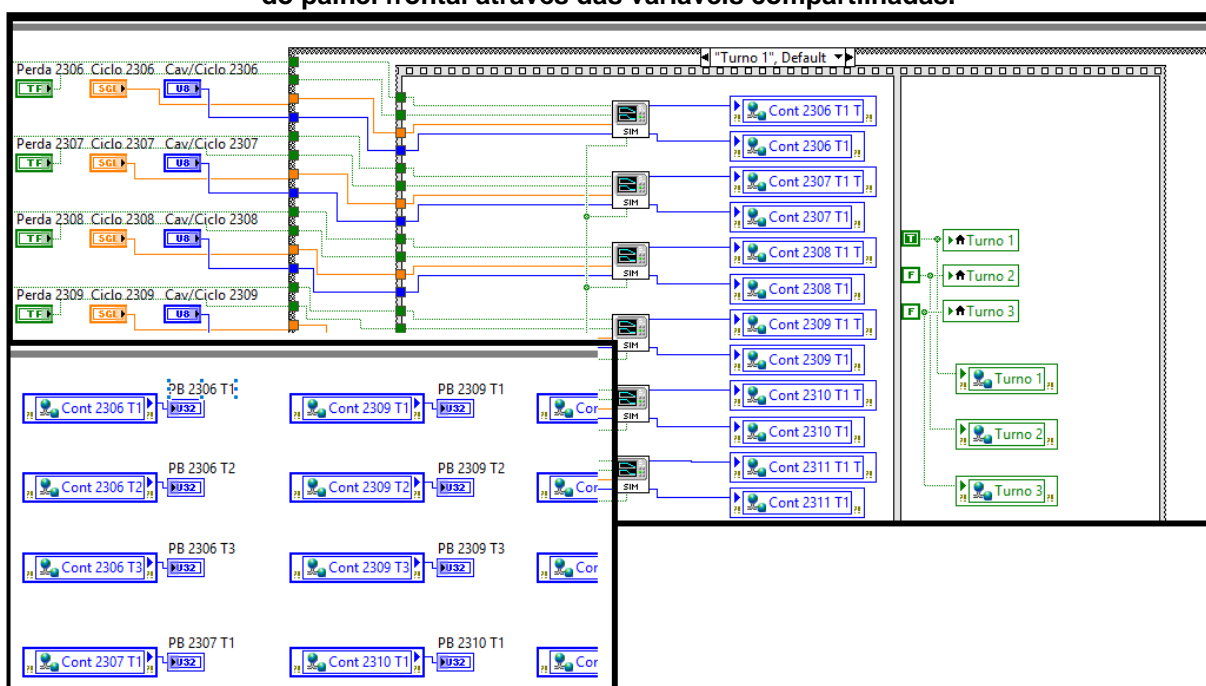
Figura 22: SubVI Tempos.vi.



Este SubVI gera os dados através dos pulsos de troca de turno do SubVI gerador de pulso único e do controle de máquina ligada/desligada, e os armazena nas variáveis compartilhadas respectivas.

Por fim, o VI de simulação através dos controles do painel frontal e do *SubVI* de simulação de máquina, armazena as informações de quantidades produzidas e de perdas nas respectivas variáveis compartilhadas e, através delas, os indicadores do painel frontal, conforme se observa na Figura 23:

Figura 23: Fluxo de dados do *SubVI* de simulação de máquina e alimentação dos indicadores do painel frontal através das variáveis compartilhadas.



Dessa forma, o VI de simulação da planta de produção transmitirá os dados simulados através de variáveis compartilhadas para o desenvolvimento da aplicação de monitoramento de produção.

3.4 ESCOLHA DO SGBD

Para que as informações geradas pelo sistema sejam consistentes e possam ser consultadas por um longo período, é fundamental que sejam armazenadas em um banco de dados externo a aplicação. Desta maneira, o sistema de monitoramento de produção poderá funcionar em segundo plano de forma a não consumir muitos recursos do computador em que está instalado. Além disso, caso haja falha no sistema, os dados estarão protegidos contra perdas. Outra vantagem é a possibilidade

de futuramente desenvolver *VI*'s específicos que possam se comunicar com o banco de dados remotamente de outros computadores ou até mesmo através da internet, tornando a acessibilidade ao sistema mais dinâmica e abrangente.

Dentre os vários SGBD disponíveis no mercado, para esta aplicação foi escolhido o *MySQL*. O *MySQL* foi inicialmente desenvolvido por David Axmark, Allan Larsson e Michael Widenius, na década de 90, pela necessidade de uma interface *SQL* compatível com as rotinas que utilizavam em suas aplicações e tabelas de dados. A partir de então, o *MySQL* começou a ser difundido pelo ótimo resultado e seus criadores fundaram a empresa responsável pela sua manutenção (MILANI, 2006)

O SGBD *MySQL* afirma-se cada vez mais como o principal sistema de base de dados no que toca, pelo menos, ao mundo web. Confirmação disso é a utilização que os monstros empresariais lhe dão, como por exemplo o *Facebook*, *LinkedIn*, *Twitter*, *Cisco*, *Ebay* ou *Amazon*. (PINHO, 2015. p. 40)

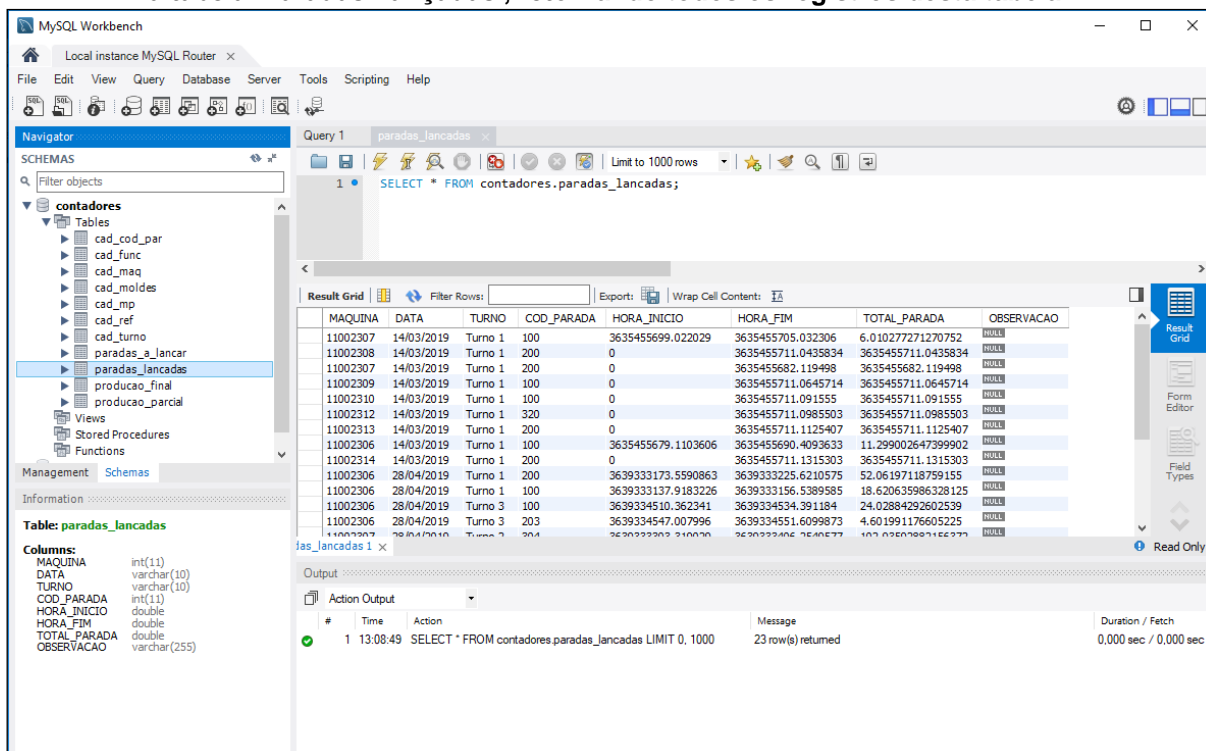
O *MySQL* é um SGBD robusto que atende as necessidades do sistema em desenvolvimento e permite sua expansão, além de possuir licença gratuita.

Inicialmente, para o desenvolvimento do sistema de monitoramento de produção, o *MySQL* será instalado no mesmo computador como servidor local, a fim de facilitar os testes dos *VI*'s em desenvolvimento e tendo em vista que a proposta de instalação da aplicação não abrange conexão com a rede de dados da empresa.

Por se tratar de um tema muito abrangente, este trabalho não abordará detalhes de funcionamento do SGBD, se limitando apenas mencionar as características principais do *MySQL* e apresentar sua ferramenta de edição.

Após a instalação do *MySQL*, através do *MySQL Workbench*, que nada mais é do que o seu editor *SQL*, pode-se configurar o banco de dados para que se comunique com a aplicação. O *MySQL Workbench* é uma ferramenta que facilita a manutenção do banco de dados, permitindo a consulta a todas as suas instâncias de forma dinâmica e intuitiva. A Figura 24 mostra o *MySQL Workbench* com todas as tabelas de dados utilizadas pelo sistema de monitoramento de produção.

Figura 24: MySQL Workbench mostrando a lista de tabelas do banco de dados e uma consulta a tabela 'Paradas Lançadas', retornando todos os registros desta tabela.



Com esta ferramenta, é possível fazer qualquer consulta e alteração no banco de dados de forma direta, o que facilita o desenvolvimento do sistema de monitoramento de produção bem como a manutenção do banco de dados.

3.5 DESENVOLVIMENTO DAS VI'S DO SISTEMA DE MONITORAMENTO

Para que o sistema de monitoramento de produção desempenhe sua funcionalidade básica, é previsto o desenvolvimento de cinco *VI's* com funções específicas, dentro dos quais estão inseridos *SubVI's* que foram desenvolvidos para simplificar o fluxo de dados.

3.5.1 *VI* Cadastro

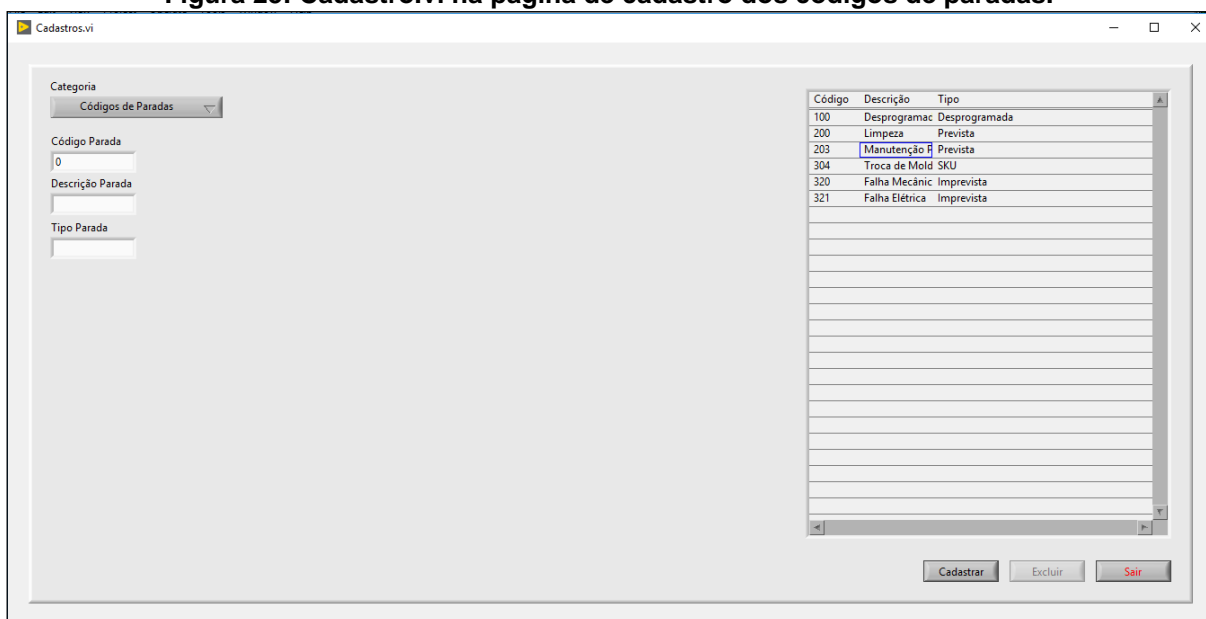
Para que a utilização do sistema seja dinâmica, se faz necessária a criação de algumas tabelas de banco de dados, de onde se possam retirar as informações básicas para a geração de relatórios, bem como cruzar as informações com outras tabelas do mesmo banco. No entanto, para que se possa conectar o *LabVIEW™* com

o banco de dados, são necessários alguns procedimentos externos, como a instalação de *drivers ODBC* e configuração do *driver* no sistema operacional *Windows*. Esses procedimentos não serão abordados por fugir dos objetivos deste trabalho. A execução destes procedimentos irá gerar um arquivo de extensão *.udl* que irá conter todas as configurações que permitirão que o *LabVIEW™* se comunique utilizando as funções de conectividade com banco de dados.

O *VI* Cadastro permite que o usuário cadastre informações como dados de máquina, produtos, funcionários, matéria prima, turnos etc. Estes dados são utilizados em outros *VI*'s e *SubVI*'s para complementar a aquisição dos dados gerados pelas linhas de produção.

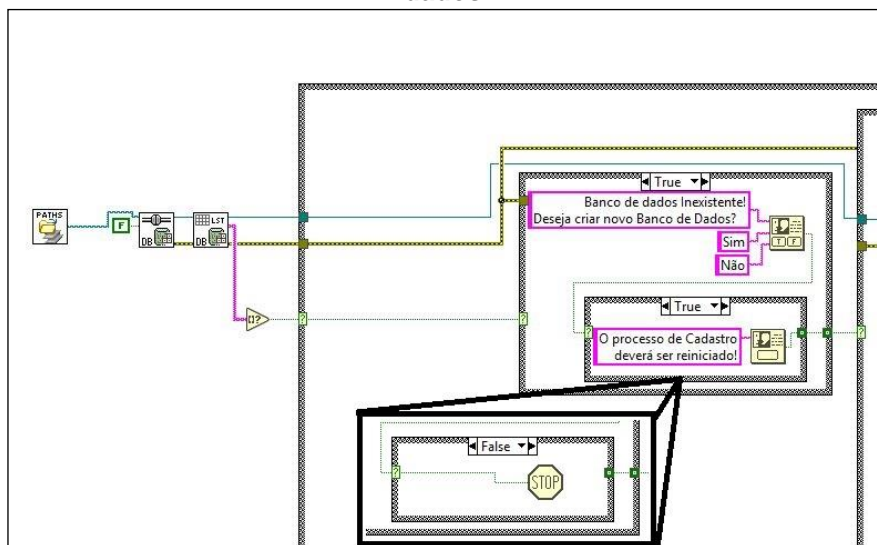
A Figura 25 mostra o painel frontal do *VI* Cadastro.vi na página de cadastro da categoria 'Códigos de Paradas'.

Figura 25: Cadastro.vi na página de cadastro dos códigos de paradas.



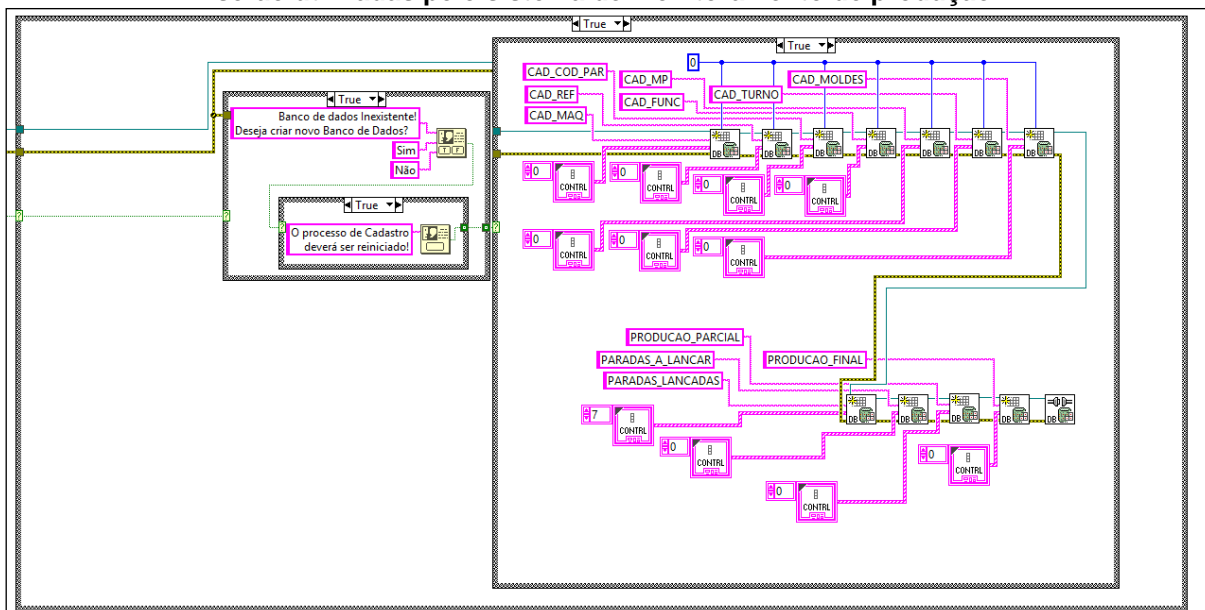
O Diagrama de blocos do *VI* Cadastro foi desenvolvido para desempenhar duas funções. A primeira é a verificação da existência de tabelas no banco de dados instalado. Ao iniciar, o *VI* executa uma consulta ao banco de dados e retorna uma lista das tabelas nele contidas (Figura 26).

Figura 266: Diagrama de blocos Cadastro.vi – verificação da existência de tabelas no banco de dados.



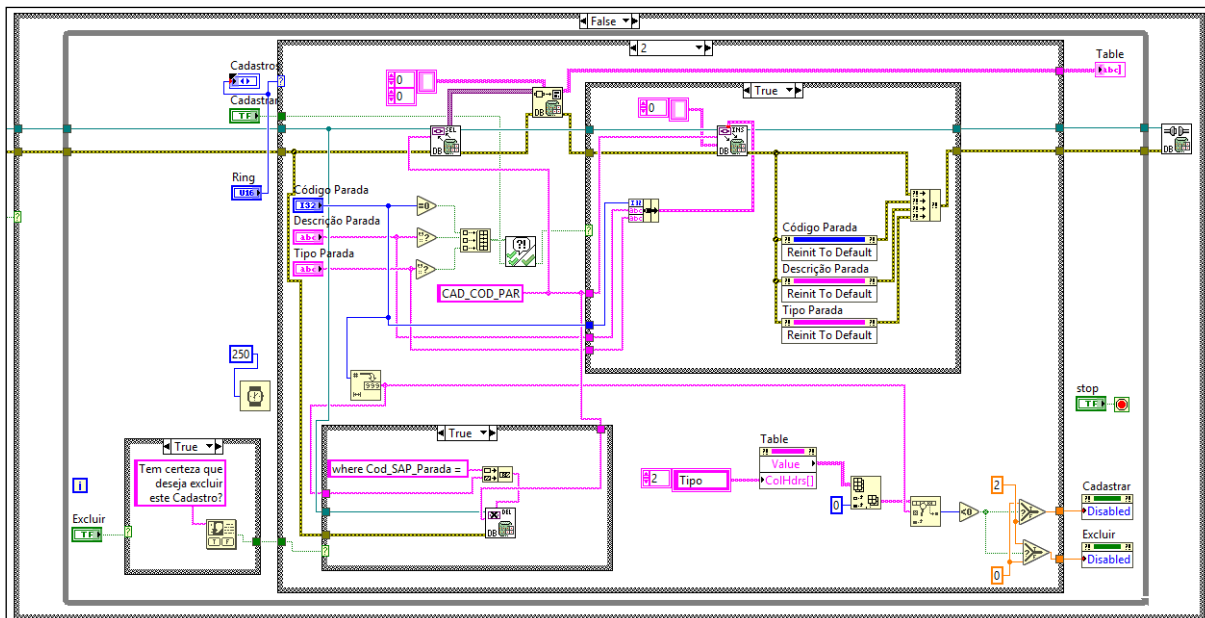
Caso a consulta retorne uma lista vazia, um verificador booleano enviará a condição 'verdadeiro' a estrutura de caso, que por sua vez retornará uma mensagem ao usuário, perguntando se deseja criar as tabelas no banco de dados. Caso a resposta do usuário for 'Não', o VI é encerrado, caso contrário, o VI executa os processos de criação das tabelas no banco de dados, conforme mostra a Figura 27:

Figura 27: Diagrama de blocos Cadastro.vi - estrutura de caso que cria todas as tabelas que serão utilizadas pelo sistema de monitoramento de produção.



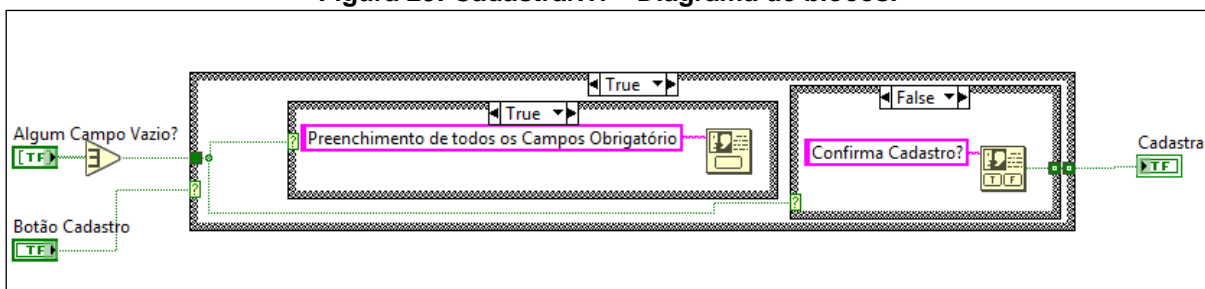
Se a consulta retornar uma lista não vazia, o verificador assume o valor booleano 'Falso', arremetendo a estrutura de caso a mesma condição, conforme mostra a Figura 28:

Figura 28: Diagrama de blocos Cadastro.vi – Estrutura de caso que processa as entradas do usuário e atualiza o banco de dados.



Esta estrutura processa as entradas do usuário através dos controles do painel frontal, enviando as informações à tabela correspondente na categoria que se está cadastrando (neste caso, 'Códigos de Paradas'). Ao acionar o controle 'Cadastrar' no painel frontal, os controles onde foram inseridos os dados passam por uma verificação para conferir se não existem campos vazios. Um *SubVI* foi desenvolvido para receber essa verificação e gerar uma mensagem de orientação ao usuário. A Figura 29 mostra o diagrama de blocos deste *SubVI*:

Figura 29: Cadastrar.vi – Diagrama de blocos.

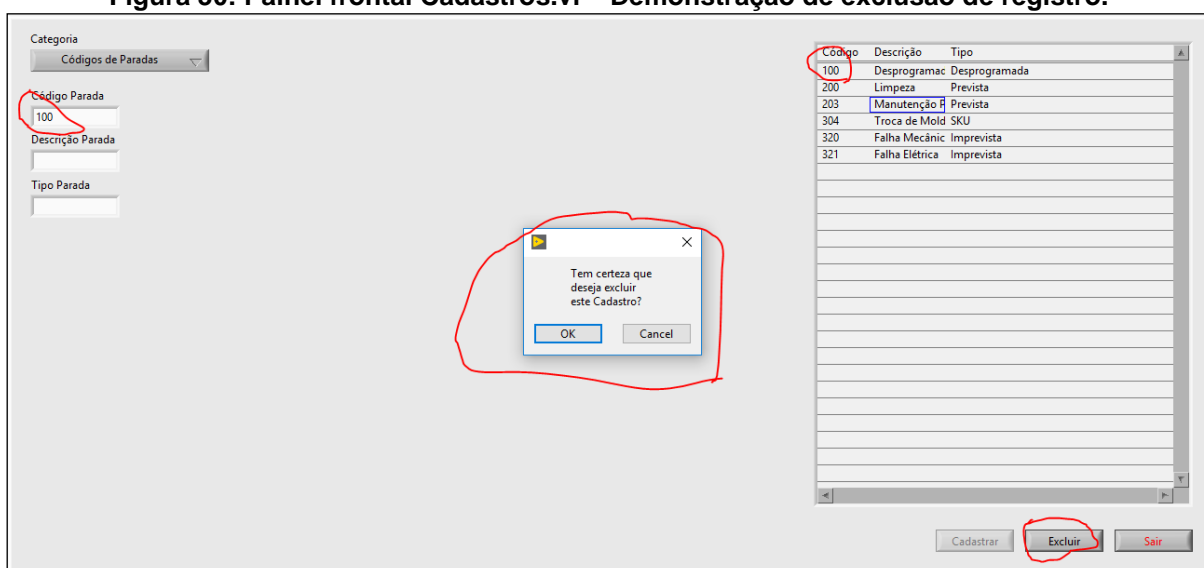


Caso algum campo esteja vazio, o *SubVI* alerta o usuário quanto a obrigatoriedade do preenchimento de todos os campos. Caso todos os campos

estejam preenchidos, uma confirmação de cadastro é exigida ao usuário. Ao confirmar o cadastro, o VI então converte as informações dos controles em um cluster de dados e os envia a uma função que armazena as informações na tabela correspondente do banco de dados.

Além de cadastrar, o VI também possui a função excluir, cujo controle no painel frontal se torna habilitado quando o usuário digita no primeiro campo um valor contido na tabela do banco de dados, que é mostrada de forma resumida no próprio painel frontal (Figura 30).

Figura 30: Painel frontal Cadastros.vi – Demonstração de exclusão de registro.



Ao clicar em excluir, o usuário visualizará uma caixa de texto exigindo uma confirmação de exclusão.

Ao iniciar o sistema de monitoramento de produção pela primeira vez após a instalação, é fundamental que o VI Cadastro seja o primeiro a ser executado, pois este irá configurar o banco de dados e carregar com as informações necessárias para a execução dos outros VI's.

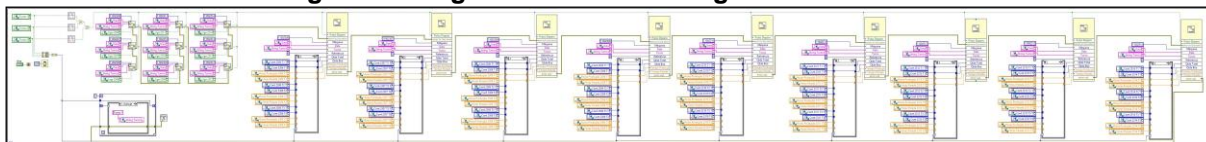
3.5.2 VI Segundo Plano

Para que se possa obter os dados de produção, é necessário um VI que seja executado em segundo plano. Esse VI monitora todas as variáveis geradas pelas linhas de produção e, em eventos específicos de parada ou troca de turno, armazena

estas variáveis no banco de dados, em registros provisórios nas respectivas tabelas. Estes dados permanecem armazenados nestas tabelas até que o usuário os complemente com outras entradas que não podem ser geradas de forma automática, como motivos de paradas e observações durante o turno de trabalho.

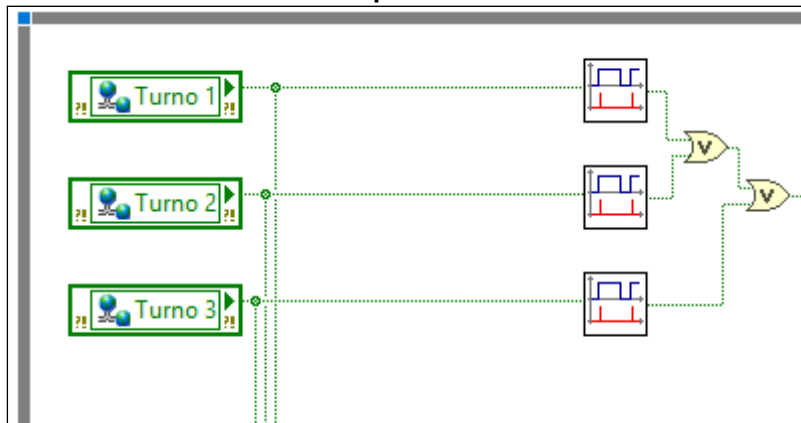
A Figura 31 mostra o Diagrama de blocos do VI Segundo Plano:

Figura 31: Segundo Plano.vi – Diagrama de blocos.



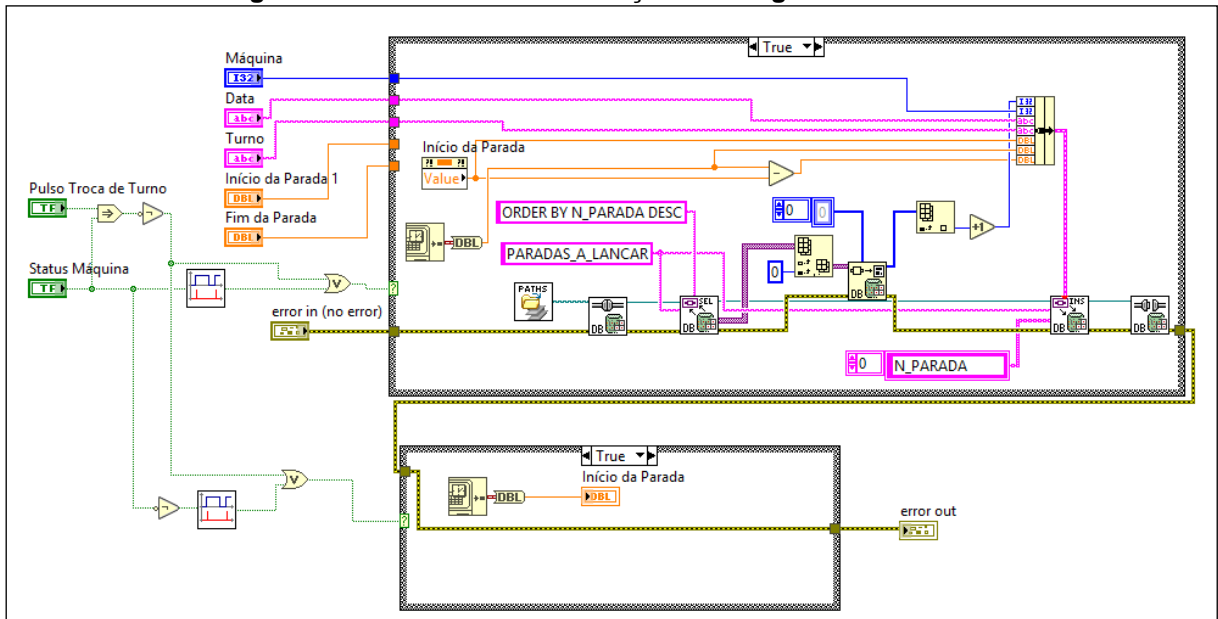
O VI Segundo Plano utiliza três *SubVI's* desenvolvidos para executar a tarefa de armazenar as informações no banco de dados. Um deles é já abordado VI Gerador de Pulso Único, que recebe a informação do turno corrente e envia um pulso a cada troca de turno (Figura 32).

Figura 32: Segundo Plano.vi – *SubVI* Gerador de Pulso Único recebendo informação da variável compartilhada Turnos.



O pulso gerado, irá disparar os outros dois *SubVI's* desenvolvidos para esta aplicação. Um deles é o Inserir Paradas a Lançar, cujo diagrama de blocos é representado na Figura 33:

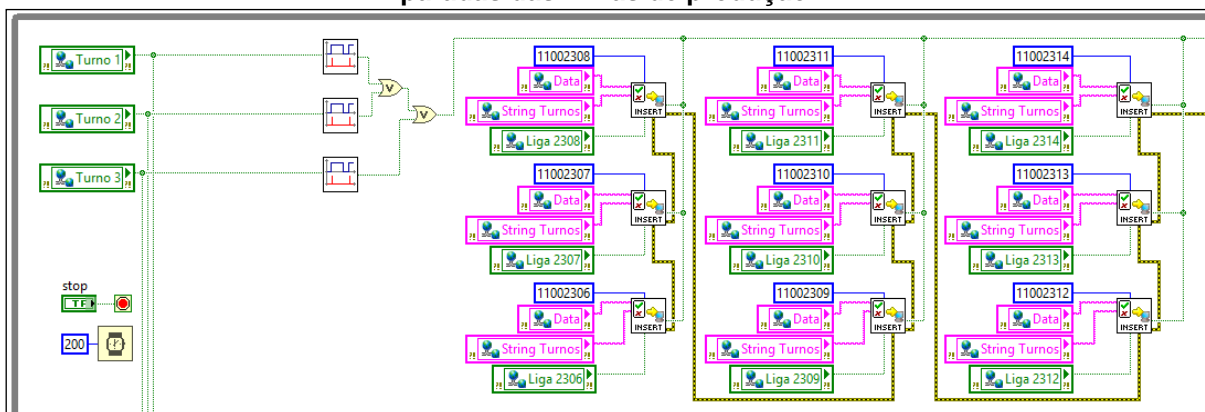
Figura 33: Inserir Paradas a Lançar.vi – Diagrama de blocos.



Este *SubVI* tem a função de armazenar na tabela Paradas a Lançar do banco de dados, as informações das paradas de linha de produção, como a identificação da máquina, a data, o turno, o momento que ocorreu a parada, o momento que a máquina reiniciou e o tempo que durou a parada. O valor booleano que representa o status da máquina (Ligada/Desligada), dispara a estrutura de caso que armazena o momento que foi disparado em um controle que registra o início da parada. Desta forma, outra estrutura de caso processa estas informações e, utilizando as funções de conectividade a banco de dados, armazena as informações na tabela. Mas as existem casos de paradas que podem gerar dados que podem ser armazenados no banco de dados de forma inconsistente, como quando uma parada ocorreu em um turno e o reinício em outro. Para evitar esta inconsistência, as estruturas de caso também podem ser disparadas pelo pulso de troca de turno. Assim, caso o evento exemplificado ocorra, são gerados dois ou mais registros na tabela, um para cada turno decorrido, permitindo que as informações sejam armazenadas de forma correta.

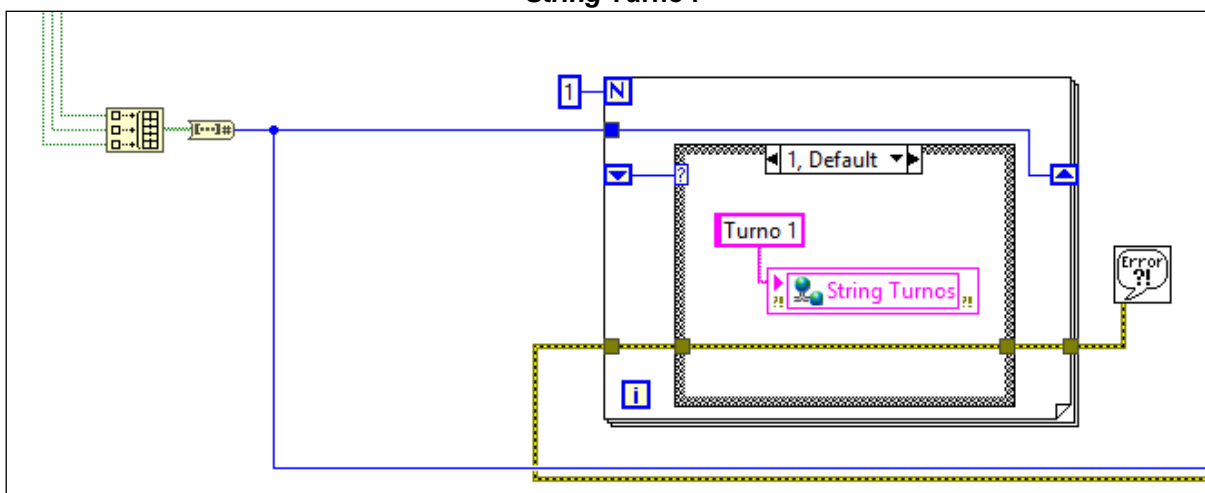
Assim, a estrutura que representa o fluxo de dados para o registro das paradas das linhas de produção que foi desenvolvida para o *VI Segundo Plano* pode ser visualizada na Figura 34:

Figura 34: Segundo Plano.vi – Representação da estrutura que armazena as informações de paradas das linhas de produção.



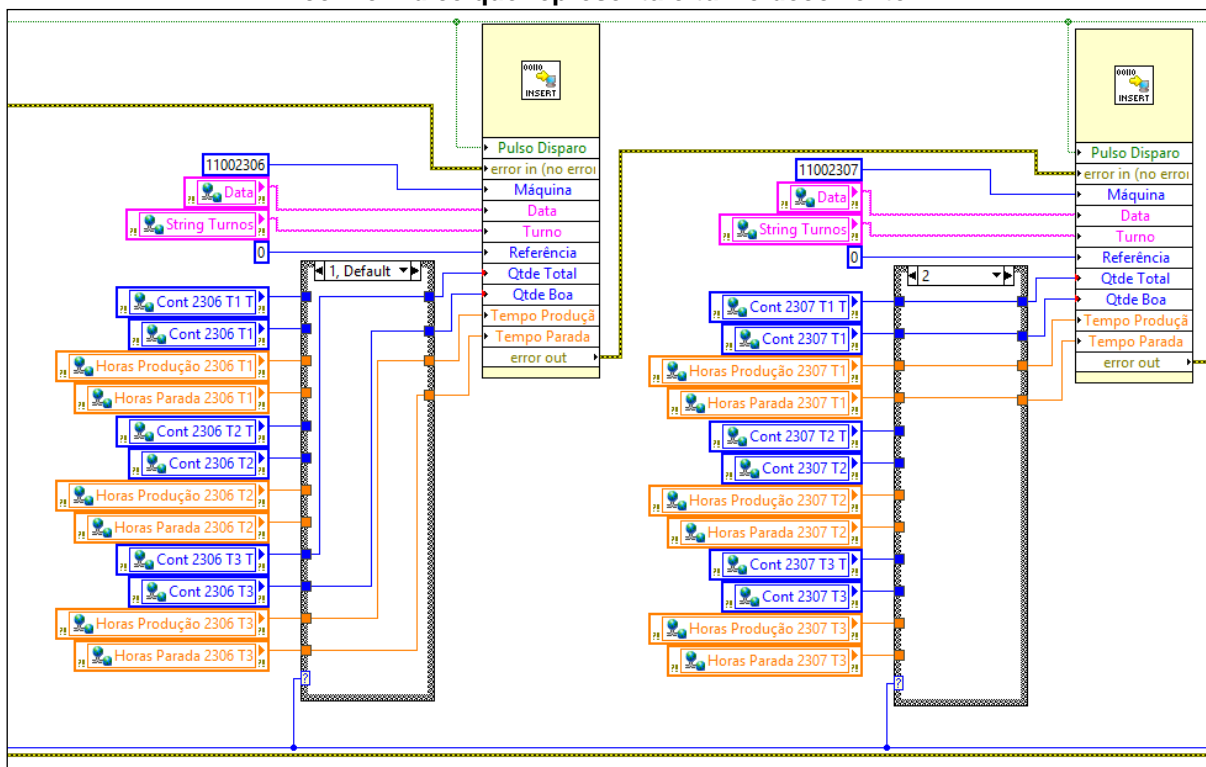
Esta estrutura utiliza cada *SubVI* como a representação de cada uma das linhas de produção da planta, recebendo os valores de suas respectivas variáveis compartilhadas. Porém, a informação de turno é representada por dados booleanos, e o armazenamento deste dado na tabela de paradas a tornaria confusa. Assim a solução foi implementar uma estrutura de caso que transformasse esta informação em dados do tipo *string*. e a armazenasse em outra variável global ('*String Turnos*'), conforme mostra a Figura 35:

Figura 35: Segundo Plano.vi – Estrutura de caso para gerar a informação da variável global '*String Turno*'.



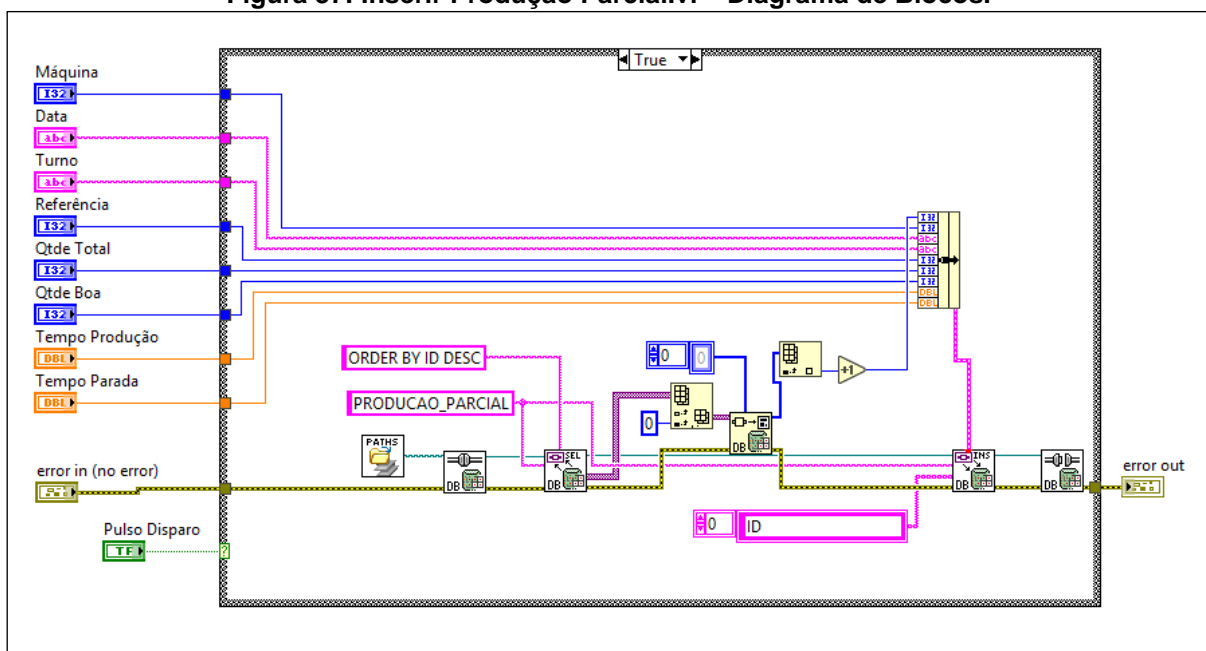
Esta estrutura recebe os dados booleanos da variável global '*Turnos*' e os armazena em um arranjo booleano, que por sua vez é transformado em um número inteiro que irá servir como índice para a estrutura de caso que armazenará os valores na variável compartilhada '*String Turnos*'. A conversão do arranjo booleano também será aproveitada como índice em outras estruturas de caso que irão armazenar valores específicos em outras variáveis compartilhadas (Figura 36).

Figura 36: Segundo Plano.vi – Estruturas de caso que direcionam as informações de acordo com o índice que representa o turno decorrente.



Estas estruturas de caso irão alimentar o terceiro *SubVI*, que foi desenvolvido para esta aplicação com a função de armazenar os dados de produção na tabela 'Produção Parcial' do banco de dados. O diagrama de blocos deste *SubVI* é mostrado na Figura 37.

Figura 37: Inserir Produção Parcial.vi – Diagrama de Blocos.



Ao receber o pulso de troca de turno, este *SubVI* reúne os dados de produção em um *Cluster*, que em sua vez são armazenados na tabela através das funções de conectividade com banco de dados.

Desta forma, o VI 'Segundo Plano' monitora todas as variáveis geradas pelas linhas de produção e as armazena no banco de dados para posterior processamento.

3.5.3 VI Monitoramento Online

A fim de se obter uma consulta rápida quanto ao *status* da linha de produção, foi desenvolvido um *VI* que mostra as informações de produção em tempo real. Com uma interface simples e objetiva, o operador pode verificar as produções individuais de cada máquina por turno de trabalho, bem como o status da máquina, conforme é demonstrado na Figura 38.

Figura 38: Monitoramento Online.vi – Painel frontal.

Turno	
Turno 2	▼

Máquina	
Máquina 2306	▼

Produção Total	60		
Produção Boa	26		
Refugo	34	% Refugo	56,67%
Tempo Real Produção	00:00:37		
Tempo Total Parada	00:00:00		

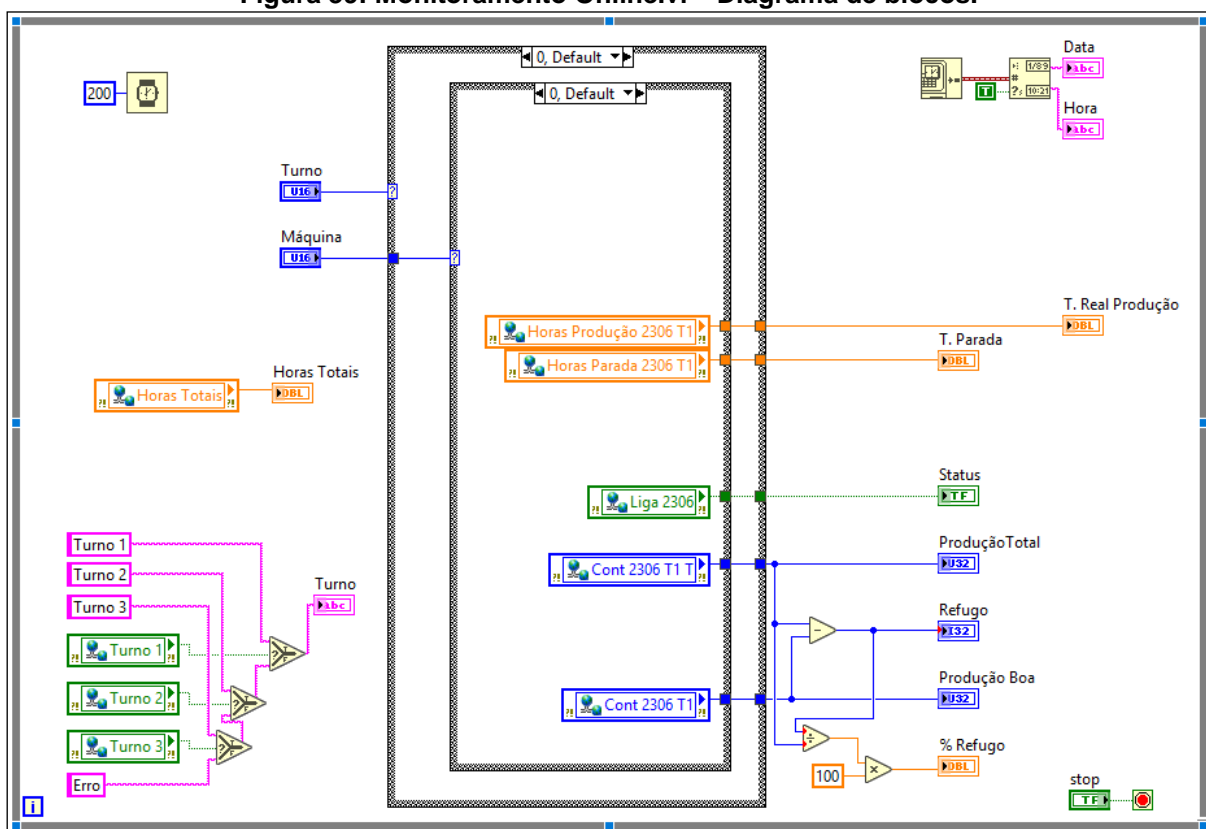
Sair

Data	Hora	Turno Atual
05/05/2019	16:16:24	Turno 1

Horas Totais	00:32:08	Status	●
--------------	----------	--------	---

Este VI recebe os dados de produção através das respectivas variáveis compartilhadas e transmite para os indicadores do painel frontal, de acordo com o turno e a máquina selecionados pelo usuário. A Figura 39 ilustra o diagrama de blocos desenvolvido para este VI.

Figura 39: Monitoramento Online.vi – Diagrama de blocos.



Os controles 'Turno' e 'Máquina' selecionam as estruturas de caso respectivas, que por sua vez transmitem os dados das variáveis compartilhadas para os indicadores no painel frontal. As informações de Turno e Horas Totais não dependem da seleção do usuário. Já os valores de Refugo e Percentual de Refugo são calculados pelo VI fora da estrutura de caso.

3.5.4 VI Dados de Produção

Com sistema de monitoramento de produção já executando as funções de monitoramento, é necessário o desenvolvimento de um VI que permita o processamento e armazenamento dessas informações no banco de dados de forma

que sejam consistentes, e permitam consultas para alimentar os indicadores de produtividade das linhas de produção e da planta. Para tanto, o VI precisa ter uma interface com o usuário intuitiva, que facilite o entendimento das informações e agilize a entrada de dados que por ele é necessária que se realize. Com esse intuito, foi desenvolvido o painel frontal para o VI 'Dados de Produção' conforme mostrado na Figura 40.

Figura 40: Dados de produção.vi – Painel Frontal.

The screenshot displays the 'Dados de produção.vi' interface, which is divided into several sections for data entry:

- Paradas a Lançar:** A table with columns for ID, Máquina, Data, and Turno. It contains two rows: (3, 11002308, 03/05/2019, Turno 1) and (4, 11002308, 03/05/2019, Turno 2).
- Máquina e ID da Parada:** Dropdown menus for selecting the machine (11002308) and the stoppage ID (3).
- Turno:** A dropdown menu set to 'Turno 1'.
- Tempo e Código da Parada:** Fields for 'Hora Inicial' (21:07:29), 'Hora Final' (21:08:00), 'Tempo Total' (00:00:30), and 'Código da Parada' (320).
- Descrição da Parada:** A text field containing 'Falha Mecânica'.
- Operador:** A dropdown menu set to 'Clairton Batista'.
- Máquina e ID da Produção:** Dropdown menus for selecting the machine (11002310) and the production ID (5).
- Observações:** A text field containing 'Máquina perdendo muito na linha'.
- Produção a Confirmar:** A table with columns for ID, Máquina, Data, Turno, Qtde Total, and Qtde Boa. It contains one row: (5, 11002310, 03/05/2019, Turno 1, 110, 75).
- Buttons:** 'Lançar' (green checkmark), 'Atualizar' (refresh), and 'Sair' (red square).

Este VI permite que o usuário visualize os dados de parada e de produção, que foram gerados pelos outros VIs da aplicação, e insira dados complementares para finalizar os registros de cada linha de produção individualmente.

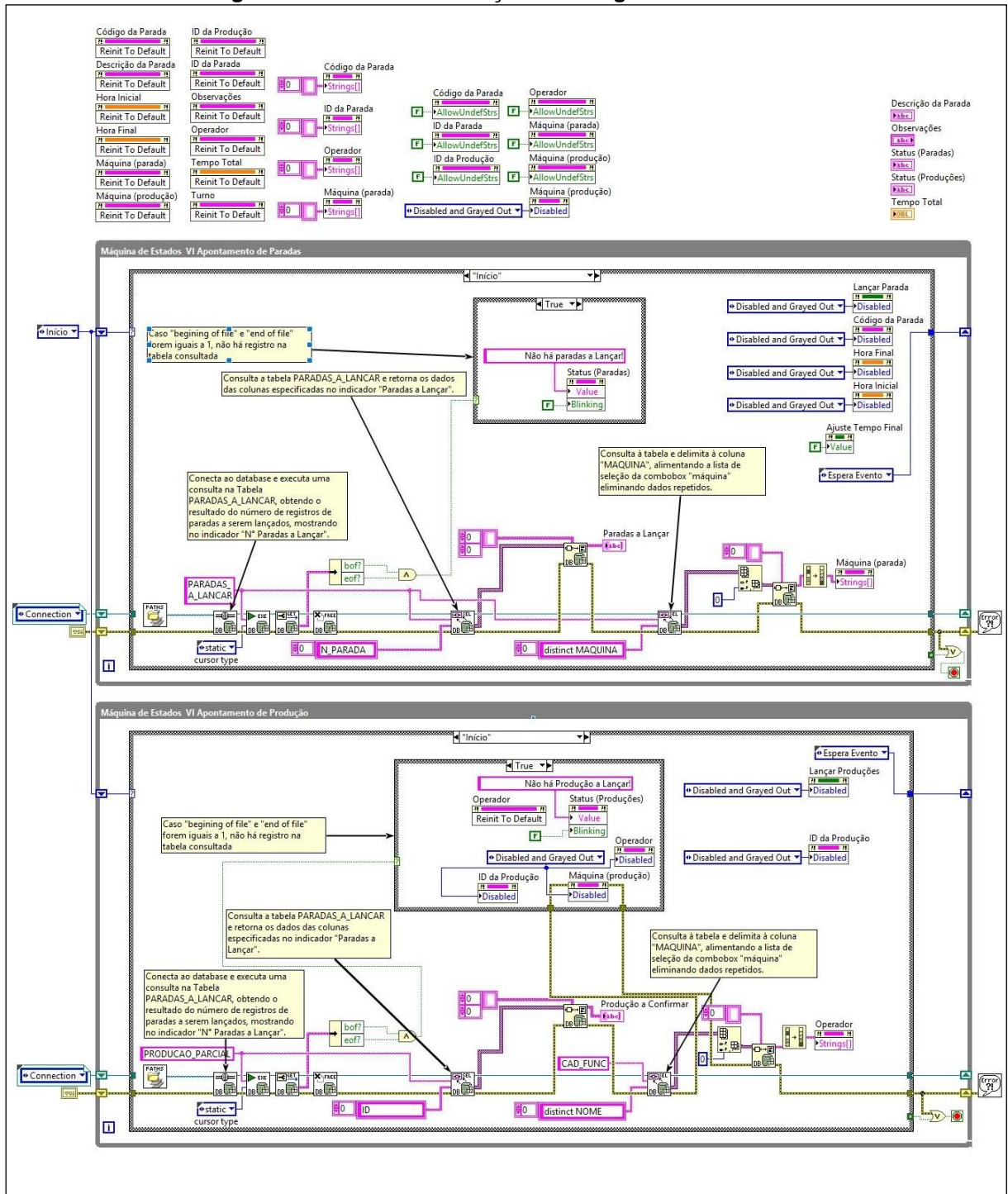
Para lançar as paradas, o usuário primeiro identifica na lista 'Paradas a Lançar' o item que vai registrar. Na sequência seleciona a máquina e o ID da parada. Caso seja necessário apontar dois códigos distintos para esta parada, o usuário pode apontar o momento final para o primeiro código e registrar o código correspondente. Ao clicar em Lançar, o VI armazenará os dados no banco de dados, e removerá o

registro da lista (caso o usuário tenha alterado o momento final da parada, o *VI* utiliza este valor como momento inicial de outra parada para que seja apontado outro código de parada).

De forma semelhante, para lançar os dados de produção o usuário identifica na lista 'Produção a Confirmar' o registro que vai apontar, seleciona o seu nome, a máquina e o *ID* da produção e registra as observações que julgue necessário ao relatório. Por fim, ao clicar em 'Lançar', o *VI* armazena os dados no banco de dados e remove o registro da lista.

Para que o *VI* execute esta funcionalidade, o diagrama de blocos foi desenvolvido no modelo de máquina de estados finita. Desta forma, o *VI* só irá executar as tarefas de acordo com o momento em que o usuário insira as informações e os comandos que necessitar. A Figura 41 mostra o diagrama de blocos.

Figura 41: Dados de Produção.vi – Diagrama de blocos.

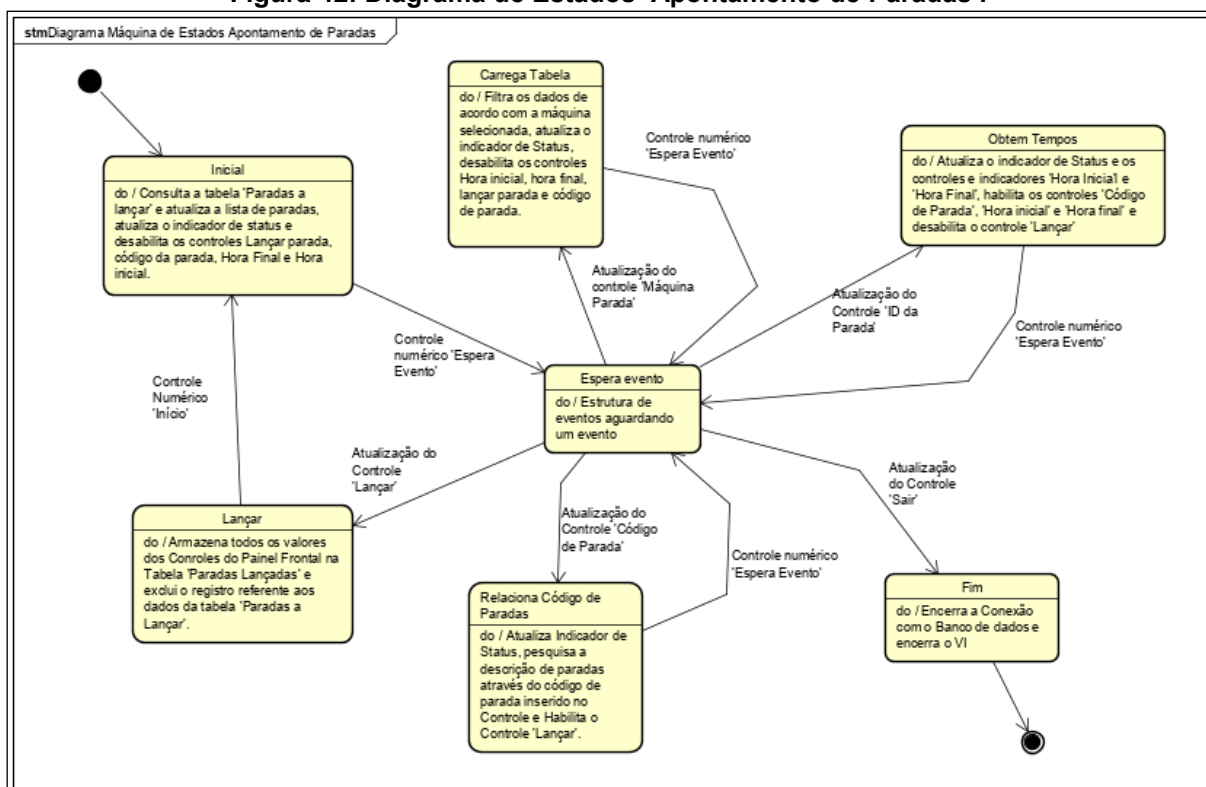


Ao iniciar o VI, os controles e indicadores do painel frontal são atualizados para os valores padrões e alguns se tornam desabilitados através dos nós de propriedade. Na sequência, através do controle numérico de índice, as estruturas de caso principais, no estado 'Início' executam uma varredura no banco de dados através das funções de conectividade a banco de dados e atualizam os indicadores do painel frontal. Após a atualização, o controle numérico dentro de cada estrutura de caso

armazena seu valor ('Espera Evento') nos respectivos *Shift Registers*, alterando a estrutura para o próximo estado. Neste estado, o VI não executa nenhuma função até que algum evento ocorra no painel frontal, como a seleção ou clique de algum controle pelo usuário. Dependendo do controle acionado pelo usuário, o VI executa as funções inseridas dentro da respectiva estrutura de caso, como o preenchimento de listas suspensas, atualização das tabelas de descrição de paradas e produções, geração de alertas nos indicadores de status e leitura e armazenamento nas tabelas do banco de dados.

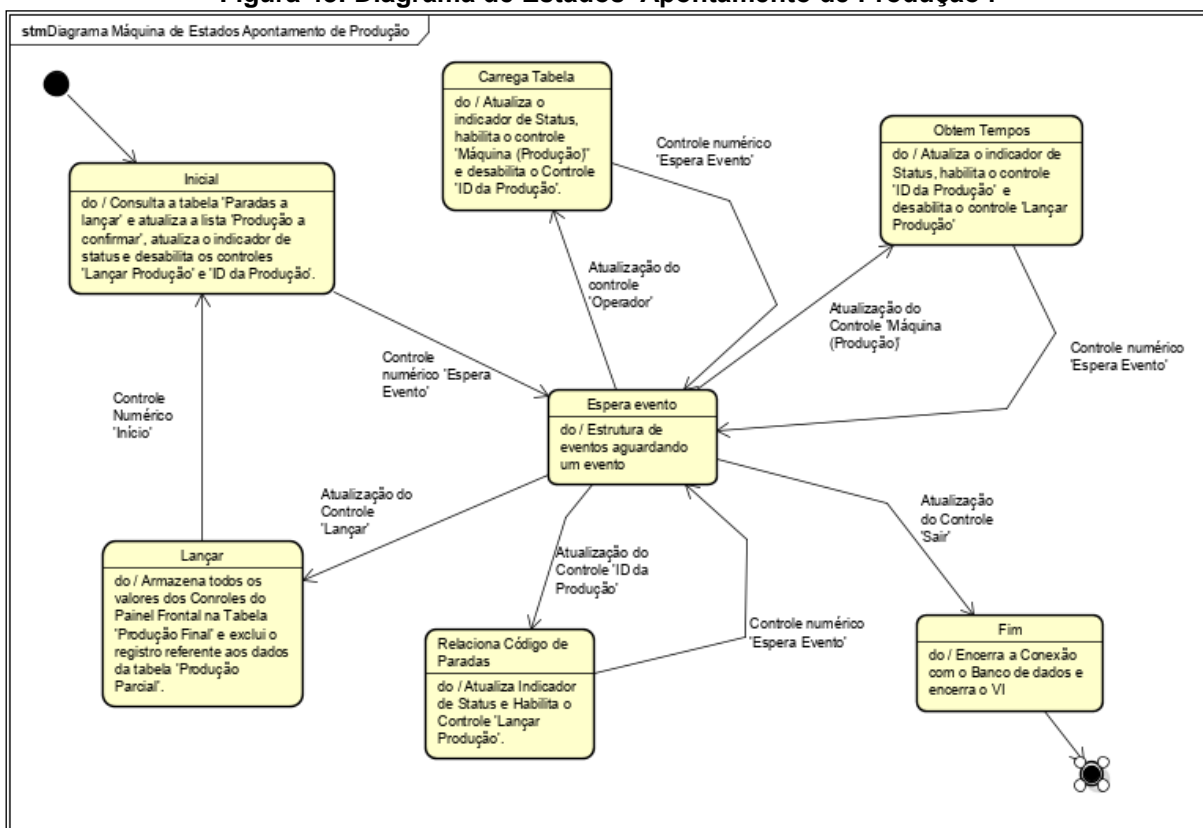
Para um melhor entendimento das etapas da máquina de estados, a Figura 42 ilustra o diagrama de estados 'Apontamento de Paradas'.

Figura 42: Diagrama de Estados 'Apontamento de Paradas'.



Do mesmo modo, a Figura 43 mostra o diagrama de estados 'Apontamento de Produção'.

Figura 43: Diagrama de Estados 'Apontamento de Produção'.

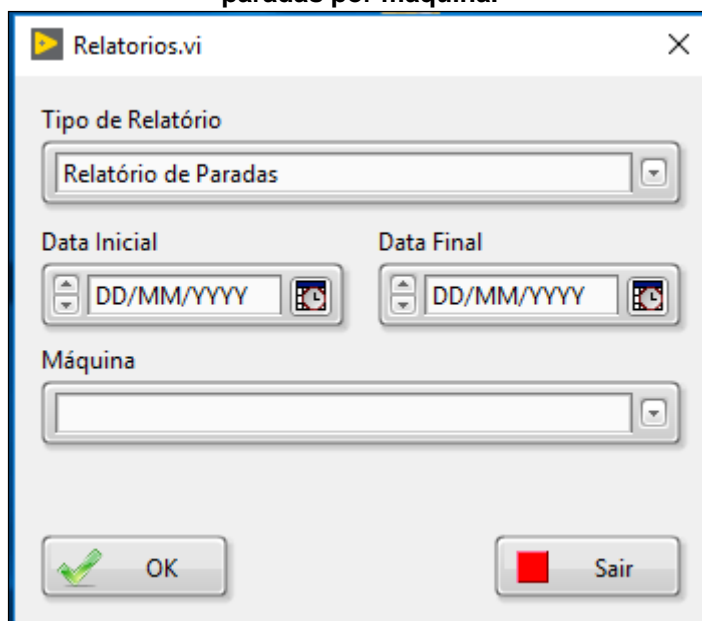


O VI 'Dados de Produção' é o único que permite armazenamento no banco de dados mesclando dados gerados automaticamente com dados inseridos pelo usuário. Desta forma, o banco de dados se torna mais consistente para que se possa analisar as informações que alimentarão os indicadores de produtividade da planta.

3.5.5 VI Relatórios

Para que o sistema de monitoramento de produção seja implementado com suas funcionalidades básicas, é necessário o desenvolvimento de um último VI que faça consultas ao banco de dados e mostre as informações em forma de relatórios. Assim, foi desenvolvido um VI com um painel frontal simples e dinâmico, com modelos de relatórios pré-definidos, conforme mostrado na Figura 44.

Figura 44: Relatórios.vi – Painel frontal mostrando os controles para consulta de dados de paradas por máquina.

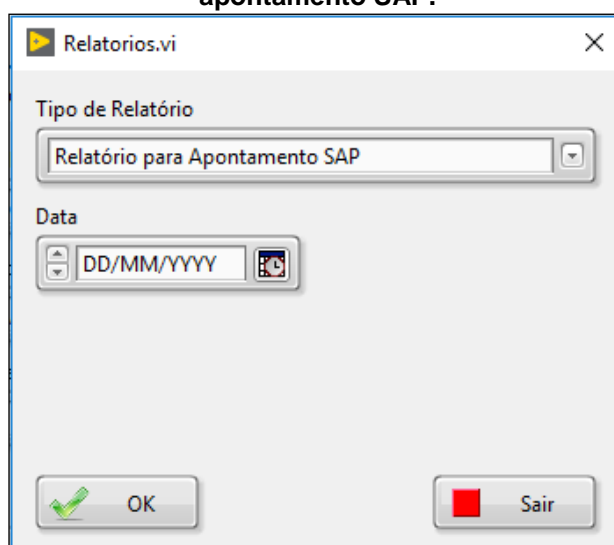


The screenshot shows a dialog box titled 'Relatorios.vi'. It contains the following controls:

- Tipo de Relatório:** A dropdown menu with 'Relatório de Paradas' selected.
- Data Inicial:** A date picker with the format 'DD/MM/YYYY' and a calendar icon.
- Data Final:** A date picker with the format 'DD/MM/YYYY' and a calendar icon.
- Máquina:** An empty text input field.
- Buttons:** 'OK' (with a green checkmark icon) and 'Sair' (with a red square icon).

Ao selecionar o tipo de relatório, o VI dinamicamente habilita os controles que são necessários para cada tipo de relatório. Na Figura 44, o painel frontal está mostrando os controles necessários para que se efetue uma busca no banco de dados que retorne as informações de paradas por linha de produção. Já na Figura 45, o painel está mostrando apenas os controles necessários para a busca que retorna as informações para o relatório utilizado para o apontamento de produção e paradas no sistema *ERP (SAP)* da empresa.

Figura 45: Relatórios.vi – Painel frontal mostrando os controles para consulta de dados de apontamento SAP.

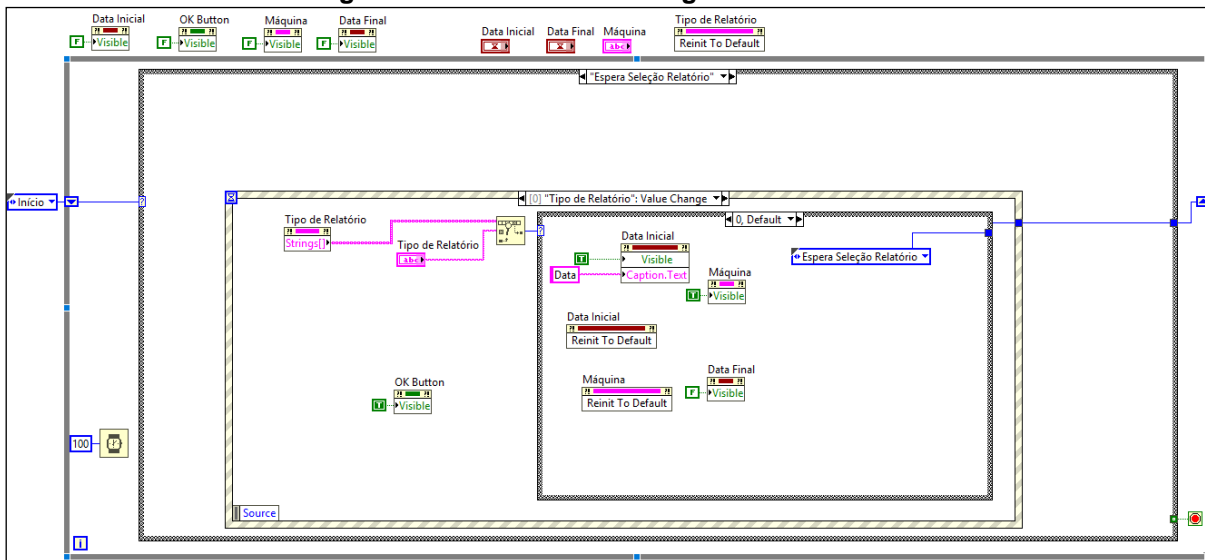


The screenshot shows the same dialog box 'Relatorios.vi' but with different settings:

- Tipo de Relatório:** A dropdown menu with 'Relatório para Apontamento SAP' selected.
- Data:** A single date picker with the format 'DD/MM/YYYY' and a calendar icon.
- Buttons:** 'OK' (with a green checkmark icon) and 'Sair' (with a red square icon).

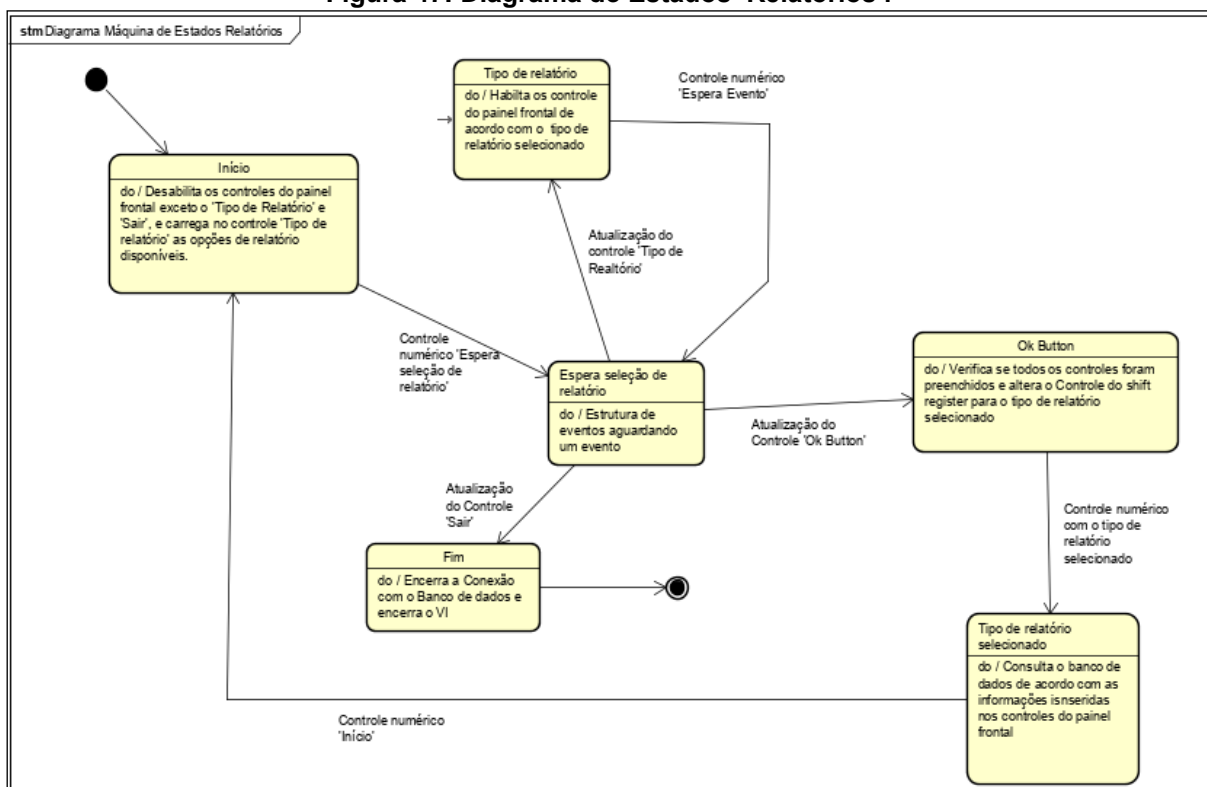
O Diagrama de blocos deste VI também foi desenvolvido no modelo de máquina de estados finita (Figura 46).

Figura 46: Relatórios.vi – Diagrama de blocos.



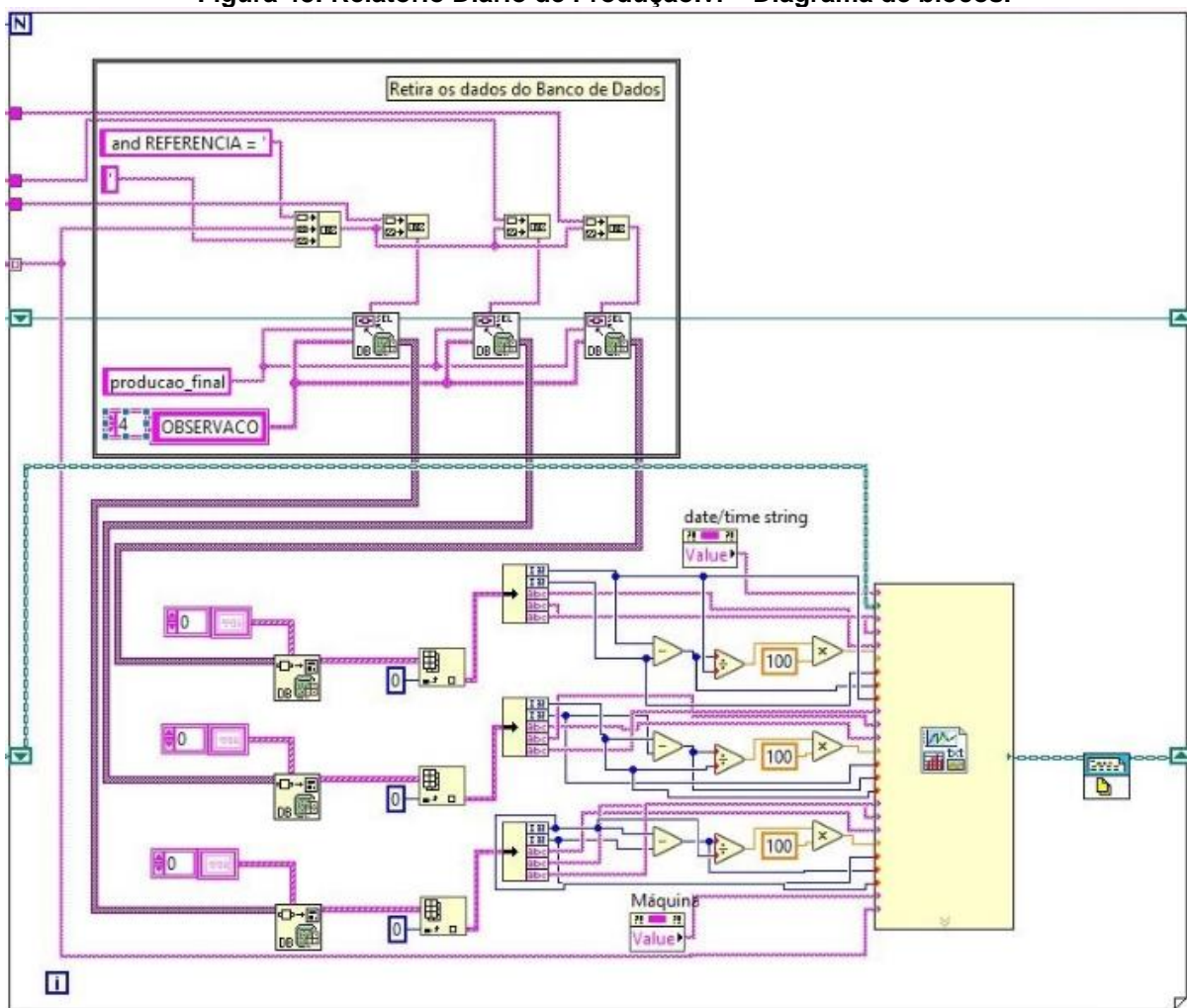
Ao iniciar, o VI desabilita a visibilidade de todos os controles do painel frontal, exceto o controle 'Tipo de Relatório' e 'Sair'. Na sequência, o valor 'Início' do controle numérico altera a estrutura de caso para que o controle 'Tipo de Relatório' carregue as informações de seleção. Após, o controle numérico altera o *shift register* para o valor 'Espera Seleção Relatório', alterando para a estrutura de caso que contém a estrutura de evento, que por sua vez irá aguardar que o usuário altere o controle selecionando o tipo de relatório desejado. Quando isso ocorre, o *shift register* recebe a informação do respectivo controle numérico que altera a estrutura de caso para processar a consulta através das funções de conectividade a banco de dados, de acordo com o relatório escolhido. A Figura 47 ilustra o diagrama de estados desenvolvido para este VI.

Figura 47: Diagrama de Estados 'Relatórios'.



Para que as consultas retornem informações consistentes, foi necessário desenvolver *SubVI's* que agrupassem os dados de forma ordenada e enviassem esses dados para outros *SubVI's* (Figura 48) que possuem seus painéis frontais organizados como modelos de relatório.

Figura 48: Relatório Diário de Produção.vi – Diagrama de blocos.



Por sua vez, esses *SubVI*'s atualizam seus indicadores com as respectivas consultas, e utilizando as funções de impressão de relatórios, geram os documentos em *HTML* e *PDF*. A Figura 49 mostra o painel frontal do *SubVI* 'Modelo Relatório Produção'.

Figura 49: Modelo Relatório Produção.vi – Painel frontal utilizado como modelo para gerar documento *HTML* e *PDF*.

LOG PRO 01 - RELATÓRIO DE PRODUÇÃO		
Data	Referência	Máquina
<input type="text"/>	<input type="text"/>	<input type="text"/>
Turno 1		
Qtde Total 1	Refugo 1	Observações 1
<input type="text" value="0"/>	<input type="text" value="0"/>	
Qtde Boa 1	% Refugo 1	
<input type="text" value="0"/>	<input type="text" value="0"/>	
Operador 1	<input type="text"/>	
Encarregado 1	<input type="text"/>	
Turno 2		
Qtde Total 2	Refugo 2	Observações 2
<input type="text" value="0"/>	<input type="text" value="0"/>	
Qtde Boa 2	% Refugo 2	
<input type="text" value="0"/>	<input type="text" value="0"/>	
Operador 2	<input type="text"/>	
Encarregado 2	<input type="text"/>	
Turno 3		
Qtde Total 3	Refugo 3	Observações 3
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text"/>

Para este trabalho, foram considerados apenas dois modelos de relatórios, por se tratar de apenas um estudo para uma possível implementação na unidade de produção de embalagens.

4 RESULTADOS

Após o desenvolvimento de todos os VI's necessários para a funcionalidade básica do sistema de monitoramento de produção, foi necessário realizar alguns testes de comunicação entre o CLP e o sistema.

4.1 TESTES DE FUNCIONAMENTO DO SISTEMA COMUNICANDO COM O CLP INSTALADO.

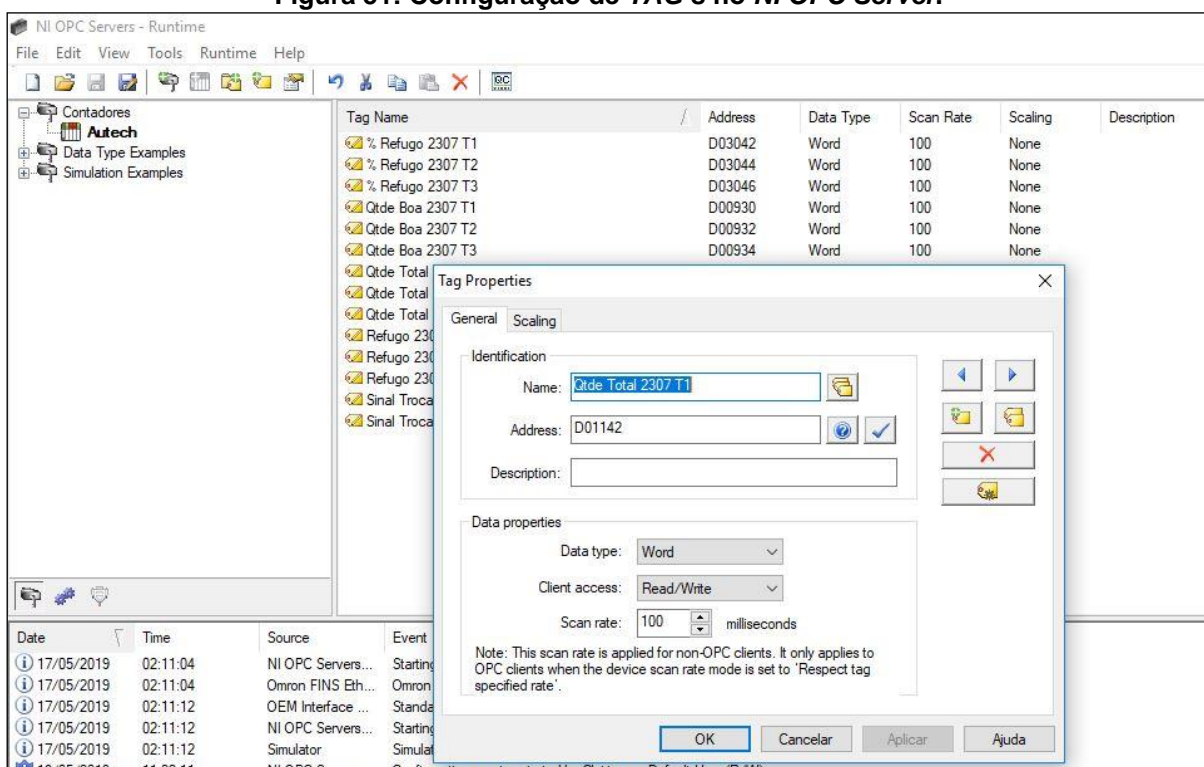
Para execução dos testes, foi necessário localizar no programa do CLP as variáveis que continham as informações da linha de produção 11002307 (linha escolhida para os testes). A Figura 50 mostra a tabela com os endereços das variáveis que contém as informações desta linha de produção.

Figura 50: Identificação dos endereços das variáveis utilizadas para o teste na linha 11002307.

Name	Data Type	Address /...	Net. Variable	Rack Locati...	Usage	Comment
CHANNEL	CHANNEL	D928			Work	INST_2306_T3_IHM
CHANNEL	CHANNEL	D930			Work	INST_2307_T1_IHM
CHANNEL	CHANNEL	D932			Work	INST_2307_T2_IHM
CHANNEL	CHANNEL	D934			Work	INST_2307_T3_IHM
CHANNEL	CHANNEL	D936			Work	INST_2308_T1_IHM
CHANNEL	CHANNEL				Work	Total 2307 T1
CHANNEL	CHANNEL				Work	Total 2307 T2
CHANNEL	CHANNEL				Work	Total 2307 T3
CHANNEL	CHANNEL	D3026			Work	Refrigo 2306 Turno 2
CHANNEL	CHANNEL	D3034			Work	% Refugo 2306 Turno 3
CHANNEL	CHANNEL	D3036			Work	Refrigo 2307 Turno 1
CHANNEL	CHANNEL	D3038			Work	Refrigo 2307 Turno 2
CHANNEL	CHANNEL	D3040			Work	Refrigo 2307 Turno 3
CHANNEL	CHANNEL	D3042			Work	% Refugo 2307 Turno 1
CHANNEL	CHANNEL	D3044			Work	% Refugo 2307 Turno 2
CHANNEL	CHANNEL	D3046			Work	% Refugo 2307 Turno 3
CHANNEL	CHANNEL	D3048			Work	Refrigo 2308 Turno 1
					Work	TOTALizador LOGOPLASTE 23/24
					Work	TROCA DE TURNO QTD Boa
					Work	Troca de Turno QTD Total
					Work	TURNO+RETRADA TURNO 1 SOPRO 11

Com os endereços localizados, o próximo passo foi configurar o NI OPC Servers para comunicar com o CLP através das TAG's (Figura 51).

Figura 51: Configuração de TAG's no NI OPC Server.



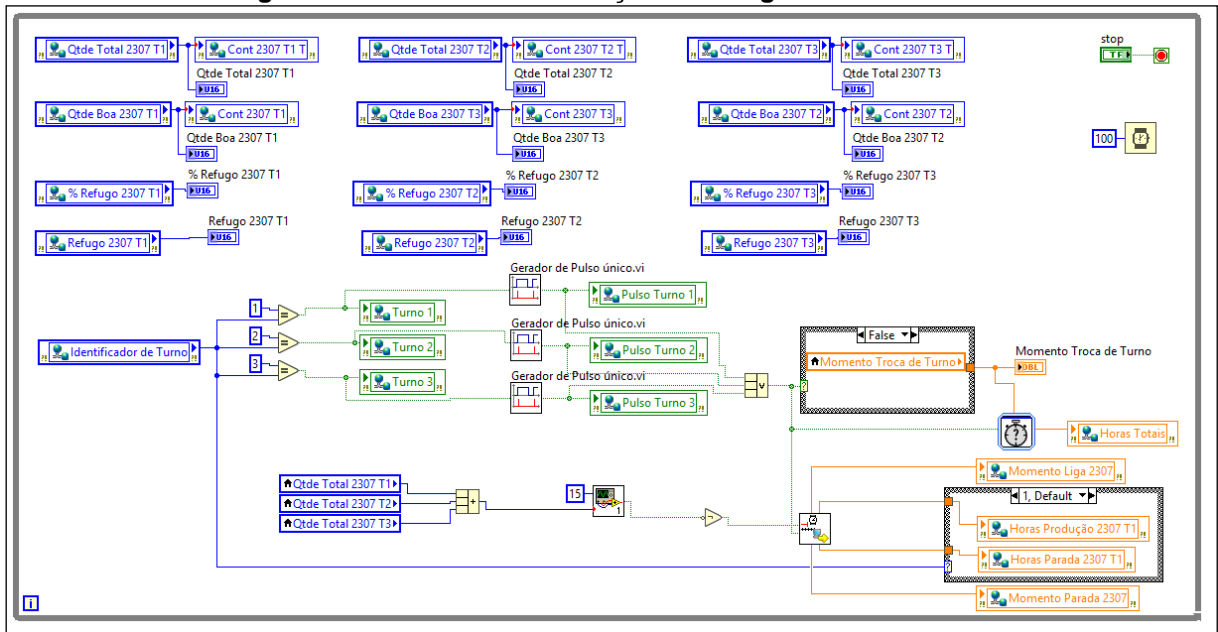
Após a configuração das TAG's, com o computador conectado à porta ethernet do CLP, foi possível monitorar os estados das variáveis e seus respectivos valores, conforme visto na Figura 52.

Figura 52: Quick client OPC com as TAG's da máquina 11002307 sendo monitoradas.

Item ID	Data Type	Value	Timestamp	Quality	Update Count
Contadores.Autech.% Refugo 2307 T1	Word	9964	16:49:58.154	Good	1
Contadores.Autech.% Refugo 2307 T2	Word	16189	16:50:46.178	Good	11
Contadores.Autech.% Refugo 2307 T3	Word	47155	16:49:58.154	Good	1
Contadores.Autech.Qtde Boa 2307 T1	Word	49348	16:49:58.196	Good	1
Contadores.Autech.Qtde Boa 2307 T2	Word	11019	16:50:51.194	Good	52
Contadores.Autech.Qtde Boa 2307 T3	Word	33113	16:49:58.196	Good	1
Contadores.Autech.Qtde Total 2307 T1	Word	51356	16:49:58.201	Good	1
Contadores.Autech.Qtde Total 2307 T2	Word	11716	16:50:51.207	Good	26
Contadores.Autech.Qtde Total 2307 T3	Word	35971	16:49:58.201	Good	1
Contadores.Autech.Refugo 2307 T1	Word	2008	16:49:58.210	Good	1
Contadores.Autech.Refugo 2307 T2	Word	697	16:50:51.184	Good	51
Contadores.Autech.Refugo 2307 T3	Word	47155	16:49:58.210	Good	1
Contadores.Autech.Sinal Troca Turno Qtde Boa	Word	2	16:49:58.215	Good	1
Contadores.Autech.Sinal Troca Turno Qtde Total	Word	2	16:49:58.215	Good	1

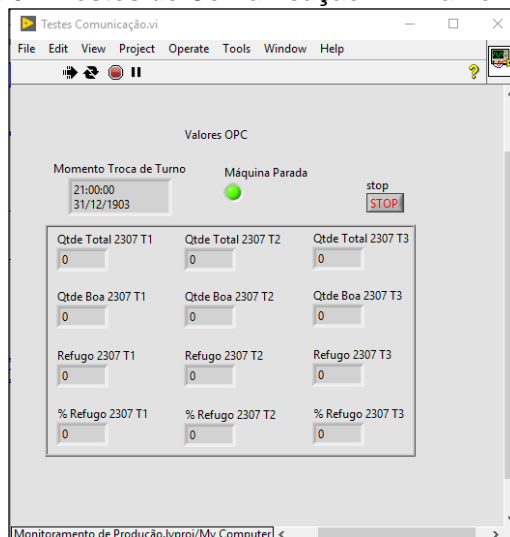
Para que o sistema de monitoramento de produção desenvolvido por fim comunicasse com o OPC, foi necessário desenvolver um VI que transferisse os valores obtidos das TAG's para as respectivas variáveis compartilhadas inseridas nos diagramas de blocos dos VI's do sistema. O Diagrama de Blocos deste VI pode ser observado na Figura 53.

Figura 53: Teste de Comunicação.vi – Diagrama de Blocos.



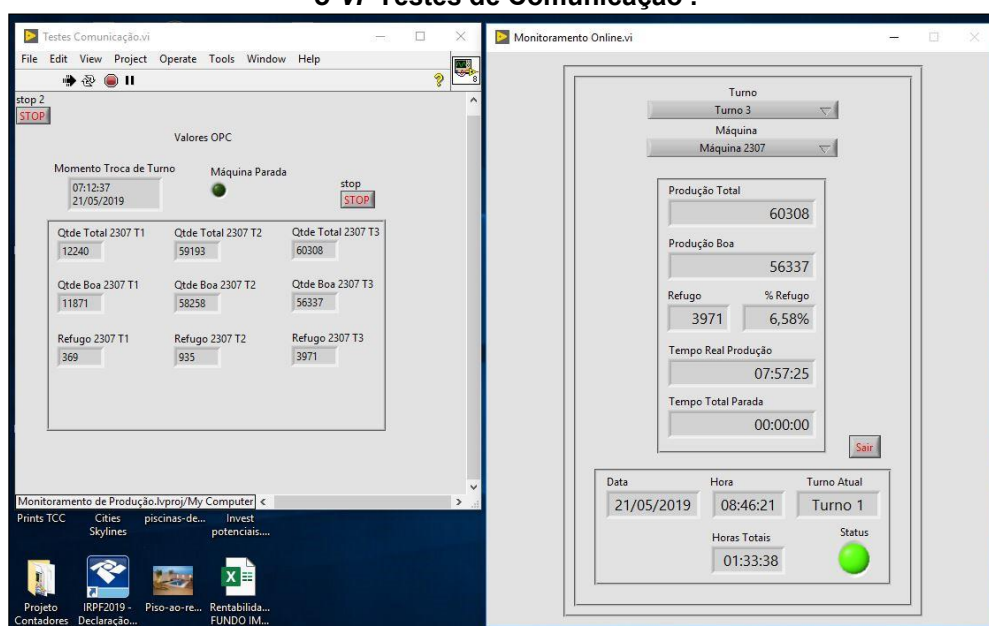
Além de transferir os dados de produção, este VI também carrega as variáveis compartilhadas com informações de turno e tempos de máquina parada. As informações de tempo são baseadas no contador de quantidade total de frascos, onde o VI monitora o incremento dos contadores baseado em tempo. Caso os contadores não incrementem no intervalo de tempo predeterminado, o VI interpreta que a máquina está parada e carrega esta informação na respectiva variável compartilhada. A Figura 54 mostra o painel frontal deste VI, que será utilizado para comparar com os dados do sistema de monitoramento de produção na execução do teste.

Figura 54: Testes de Comunicação.vi – Painel Frontal.



Por fim, o sistema de monitoramento de produção foi conectado com o CLP e permaneceu assim durante 48 horas, armazenando as informações da linha 11002307 no banco de dados. Para o teste de funcionamento do sistema, além do VI 'Testes de Comunicação', foram executados os VIs 'Segundo Plano', 'Monitoramento Online' e 'Dados de Produção'. Conforme explicado no capítulo 3, o VI 'Segundo Plano' transfere os valores das TAG's para as variáveis compartilhadas usadas nos outros VIs e também registra os eventos no banco de dados. Já o VI 'Monitoramento Online' permite acompanhar em tempo real os valores de produção constantemente atualizados, bem como os dados de produção dos turnos anteriores ao decorrente. A Figura 55 mostra a comparação dos dados deste VI com os dados reais adquiridos pelo VI 'Testes de Comunicação'.

Figura 55: Comparação dos valores de produção do Turno 3 no VI Monitoramento Online' com o VI 'Testes de Comunicação'.



Para verificar os dados de produção e paradas registrados em todo o período de teste foi executado o VI 'Dados de Produção', que executa uma consulta ao banco de dados e prepara esses dados para o apontamento de motivos de paradas e justificativas, a fim de gerar dados consistentes para um relatório completo, conforme mostrado na Figura 56.

Figura 56: Dados de Produção.vi – Lista dos dados de paradas e produções gravados no banco de dados durante o teste.

Paradas a Lançar

ID	Máquina	Data	Turno
11	11002307	14/05/2019	Turno 1
12	11002307	14/05/2019	Turno 1
13	11002307	14/05/2019	Turno 1
14	11002307	14/05/2019	Turno 1
15	11002307	14/05/2019	Turno 1

Máquina: ID da Parada: Turno:

Hora Inicial: 21:00:00 Hora Final: 21:00:00 Tempo Total: 00:00:00 Código da Parada:

Descrição da Parada:

Há paradas a Lançar. Selecione a Máquina!

Produção a Confirmar

ID	Máquina	Data	Turno	Qtde Total	Qtde Boa
2	11002307	14/05/2019	Turno 1	58991	58141
11	11002307	14/05/2019	Turno 3	60110	56232
13	11002307	14/05/2019	Turno 1	59800	59409
22	11002307	14/05/2019	Turno 2	57023	56437
31	11002307	14/05/2019	Turno 3	54018	47940
33	11002307	14/05/2019	Turno 1	43757	43217

Operador:

Máquina: ID da Produção:

Observações:

Há Produção a Lançar!

Ao executar o VI 'Dados de Produção', percebeu-se que a data registrada em cada evento estava diferente da data da execução do teste. Este erro se deu devido a não se prever, para a execução do teste, a inserção de uma função que atualizasse a variável compartilhada 'Data' no VI 'Teste de Comunicação'.

Para comprovação do registro dos dados no banco de dados, foi executado uma consulta ao banco de dados utilizando o *MySQL Workbench*. As Figuras 57 e 58 mostram as consultas às tabelas de produções e paradas, respectivamente, revelando os registros que foram gerados automaticamente.

Figura 57: Consulta à tabela Produção_Parcial.

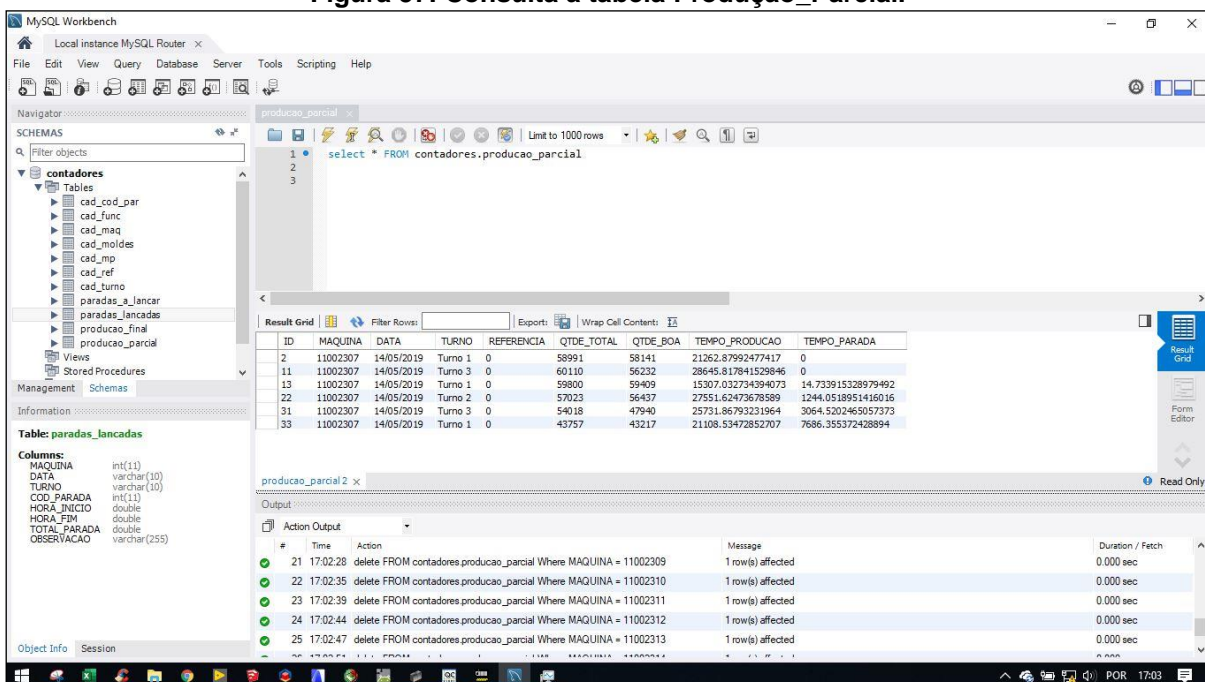
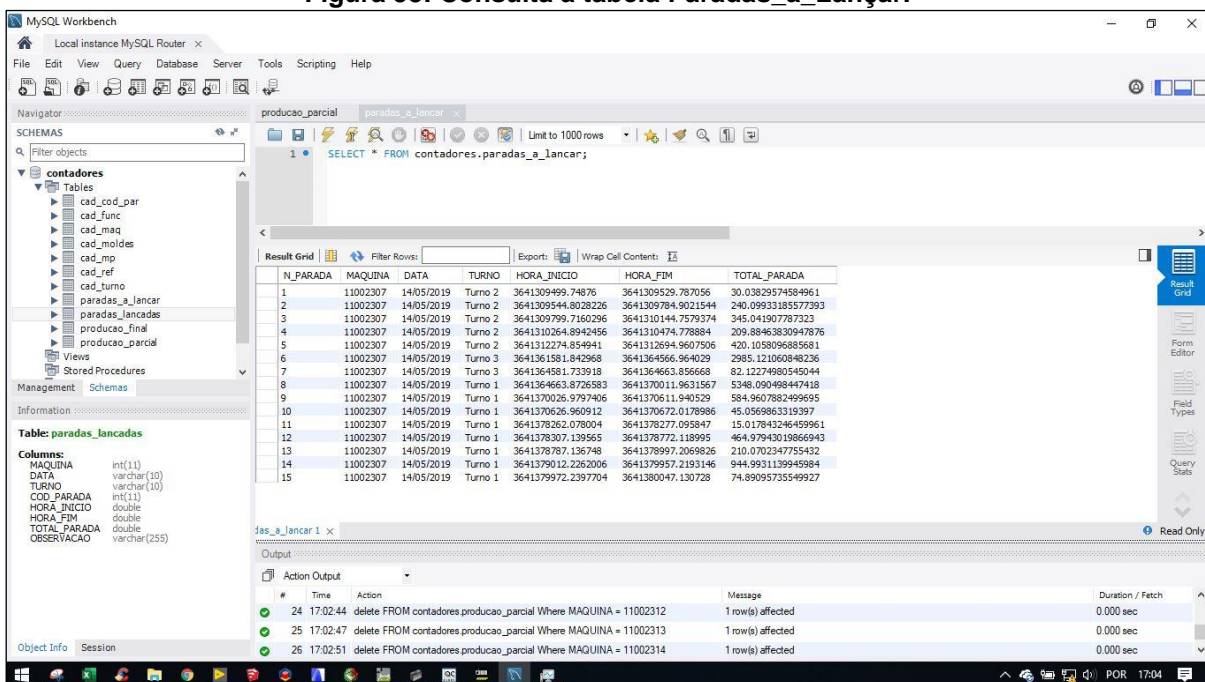


Figura 58: Consulta à tabela Paradas_a_Lançar.



5 CONCLUSÃO

A versatilidade do *LabVIEW™* como ferramenta de programação tornou possível o desenvolvimento de um sistema de monitoramento de produção em tempo real com registro de eventos em um banco de dados de forma mais prática, do ponto de vista de que não há a necessidade de conhecimento avançado em linguagens de programação convencionais. Desta forma, foi possível obter um resultado satisfatório nos testes da aplicação, com poucas alterações para correções. Porém, isto não eliminou a necessidade do planejamento minucioso das etapas de desenvolvimento, essencial ao desenvolvimento de qualquer projeto.

As maiores dificuldades no desenvolvimento desta aplicação se deram na busca de informações e aprendizado das funções básicas dos sistemas de gerenciamento de banco de dados, onde foram necessárias várias horas de vídeo aulas e exercícios. Porém, isso tornou o desenvolvimento da aplicação utilizando as ferramentas de comunicação com banco de dados muito mais fácil, devido a aplicação do conhecimento adquirido. Outro fator que demandou mais tempo de desenvolvimento, foi a inviabilidade de se desenvolver o sistema conectado diretamente ao PLC instalado, sendo necessário a criação de um *VI* de simulação. Porém esta dificuldade é encontrada pela maioria dos desenvolvedores de aplicações deste tipo, o que torna a solução encontrada na simulação a mais eficaz.

Ao que tange a funcionalidade do sistema desenvolvido, a integração com um SGBD torna a aplicação, com as funcionalidades básicas atualmente desenvolvidas, passível de grandes melhorias. Um exemplo é a possibilidade do desenvolvimento futuro de *VI's* para geração de relatórios mais específicos, que atendam a maioria das necessidades dos usuários. Além disso, é possível desenvolver *VI's* que se comuniquem em diferentes computadores da rede, o que ampliaria as funcionalidades do sistema ao nível corporativo, com possibilidades de integração com sistemas *ERP* como o *SAP*, tornando a aplicação cada vez mais semelhante a um sistema *MES*.

REFERÊNCIAS

BOTTARO, Marcio. **Instrumentação virtual em laboratório: introdução ao LabVIEW**. São Paulo: IEE-USP, 2012

CARDOSO, C. **MES na Prática**. 1. ed. 33 p. 2014.

ELMASRI, R; NAVATHE, S. B. **Sistemas de Banco de Dados**. São Paulo: Pearson Addison Wesley, 2005.

FISCHMANN, A. A.; ZILBER, M. A. Utilização de indicadores de desempenho para a tomada de decisões estratégicas: Um sistema de controle. **Revista de Administração Mackenzie**. São Paulo, v. 1, n. 1, p. 10-25. 2000.

FONSECA, Marcos de Oliveira. VI Seminário de Automação de Processos, Associação Brasileira de Metalurgia e Materiais, 2002, Vitória – ES. **Comunicação OPC -Uma abordagem prática**. Vitória, 2002. 12p.

GONÇALVES, Robson Neves. **Desenvolvimento de servidores OPC DA, OPC UA e wrappers para aplicação em automação**. Dissertação de Mestrado, UFI. Itajubá – MG, 2012.

MARDEGAN, R; AZEVEDO, R. C; OLIVEIRA, J. F. G. Encontro Nacional de Engenharia de Produção, 22, 2002, Curitiba – PR. **Os benefícios da coleta automática de dados do chão-de-fábrica para o processo de negócio gestão de demanda**. Curitiba: ENEGEP, 2002. 8 p.

MEIRA, R. **Banco de Dados**. Ilhéus, 2017. (Apostila IFBA).

MILANI, André, **MySQL: Guia do Programador**. 1ed. São Paulo: Editora Novatec, 2006.

MOELLMANN, A. H. et al. Aplicação da teoria das restrições e do indicador de eficiência global do equipamento para melhoria de produtividade em uma linha de fabricação. **Revista Gestão Industrial**. Ponta Grossa, v. 2, n. 1, p. 89-105. 2006.

NASCIMENTO FILHO, Osmar Assis. **Desenvolvimento de servidores OPC DA e OPC XML DA para sistemas de aquisição de dados via telefone celular**. Dissertação de mestrado, UFES. Vitória – ES, 2005.

NARA, E. O. B. et al. Gestão visual e a sua influência nos indicadores de produção. **Revista SODEBRAS**. v. 11, n.126, p. 141-145, jun.2016.

NATIONAL INSTRUMENTS. **Manual de treinamento do LabVIEW Básico 1**. Austin, 2001. (Manual)

PINHO, Fábio. MySQL. **Revista PROGRAMAR**. Edição 51, p. 40, dez. 2015.

SILVA, J. P. A. R. **OEE – A forma de medir a eficácia dos equipamentos**. Lean em Portugal, mai. 2009. Disponível em: <<http://www.freewebs.com/leanemportugal/artigoswhitepapers.htm>>. Acesso em: 6 abr. 2018.

SISTEMAS ERP. **O que é ERP? Para que serve esse software?** Disponível em: <<http://sistemaserp.org/o-que-e-erp/>>. Acesso em: 11 mar. 2018.

SOUZA, V. A. **Artigo sobre LabVIEW**. 2008. (Apostila).

STAMFORD, P. P. **ERP: prepare-se para esta mudança**. Recife: Ed. da UFPE, jun. 2000.