



Universidade Federal de Campina Grande

Centro de Engenharia Elétrica e Informática

Curso de Graduação em Engenharia Elétrica

DÉBORA DINIZ DE MELO

ESTÁGIO INTEGRADO: COTEMINAS S.A

Campina Grande, Paraíba.
Maio de 2013

DÉBORA DINIZ DE MELO

ESTÁGIO INTEGRADO: COTEMINAS S.A

*Relatório de Estágio Integrado submetido à
Unidade Acadêmica de Engenharia Elétrica da
Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção do grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica.*

Área de Concentração: Automação

Orientador:

Professor Alexandre Cunha Oliveira, Dr.

Campina Grande, Paraíba.
Maio de 2013

DÉBORA DINIZ DE MELO

ESTÁGIO INTEGRADO: COTEMINAS S.A

Relatório de Estágio Integrado submetido à Unidade Acadêmica de Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração: Automação

Aprovado em / /

Professor Avaliador
Universidade Federal de Campina Grande
Avaliador

Professor Alexandre Cunha Oliveira, Dr.
Universidade Federal de Campina Grande
Orientador, UFCG

Dedico este trabalho ao meu amado esposo,
Thiago de Freitas.

AGRADECIMENTOS

À Deus, por ter me dado toda a perseverança necessária para que eu concluísse este curso.

Aos meus pais, em especial à minha querida mãe, Cleide Alves, por fazerem da minha existência e da minha felicidade elementos motrizes de suas vidas.

Ao meu esposo e inspiração sempre, Thiago de Freitas, pelo encorajamento, cumplicidade, compreensão, confiança e apoio.

Às minhas irmãs, por comporem cada singelo elemento da minha formação como irmã, filha, tia, esposa e Engenheira.

Ao Engenheiro Sérgio Torres, pela oportunidade fornecida.

À Luana por sua receptividade e acolhimento.

Aos colegas do Laboratório de Eletrônica: Tatiane, Morgana, Cinthya, Reinaldo e demais, pelos momentos de descontração e por toda leveza atribuída ao ambiente de trabalho. Em especial, ao colega Eduardo Geovani, por todas as produtivas discussões e seu suporte técnico oferecido.

À Jesimiel Domingos, pelo suporte logístico.

À Cristiano, Diego e Hallisson pelo auxílio na familiarização com o maquinário.

Ao professor Alexandre Cunha Oliveira, por toda sua presteza e solicitude na orientação deste trabalho.

LISTA DE FIGURAS

Figura 01. Unidade Fabril da Coteminas Campina Grande.	4
Figura 02. II Semana da Inovação: apresentação de projetos de melhorias e inovação.	6
Figura 03. Setor da fabrica destinado à fabricação de tecidos.	7
Figura 04. Método de aquisição de dados da produção.	8
Figura 05. TSUDAKOMA ZAX.	9
Figura 06. Placa SQC: supervisor dos sinais de sensoriamento e outros status de funcionamento.	9
Figura 07. Sistema proposto com transmissão de dados via Wi-Fi.	12
Figura 08. Sistema proposto com transmissão de dados cabeada.	12
Figura 09. Divisor de tensão.	13
Figura 10. Atenuador de tensão, utilizando resistores, a ser interligado entre a máquina e o Arduino.	14
Figura 11. Buffer ou seguidor unitário.	14
Figura 12. Casamento de Impedância.	15
Figura 13. Simulação do circuito do Shield elaborado.	15
Figura 14. Shield desenvolvido de adaptação de impedância.	16
Figura 15. Shield Ethernet R3.	18
Figura 16. Esquema de pinos reservados do Shield Ethernet R3.	19
Figura 17. Instalação no Switch na tecelagem.	22
Figura 18. Topologia de rede em estrela.	23
Figura 19. Access Point utilizado em testes: Motorola AP-5131.	23
Figura 20. Shield WiFi.	24
Figura 21. Níveis hierárquicos de um processo industrial.	28
Figura 22. O diagrama de funcionamento.	29
Figura 23. Interface de Captação de Dados.	29
Figura 24. Log criado pela Interface de Captação dos Dados.	30
Figura 25. Planilha Inteligente.	31
Figura 26. Apresentação das informações em tempo real: SET.	32
Figura 27. Método de Identificação da placa através de Micro SDcard.	34

ÍNDICE

1	Introdução.....	1
1.1	Apresentação do Texto	1
1.2	Objetivos do Estágio	1
1.3	Descrição das Atividades Realizadas.....	2
2	A Empresa: COTEMINAS S.A.....	3
2.1	A História da Empresa	3
2.2	A coteminas Campina Grande	4
2.2.1	O processo de produção do fio ao tecido.....	5
2.2.2	Estrutura Organizacional	6
2.2.3	Estímulo às Melhorias	6
3	Automatização das Informações do Processo Industrial	7
3.1	A Problemática	7
3.2	Descrição de maquinário.....	8
4	Caracterização dos Sistemas Propostos.....	11
4.1	Implementação da Adaptação dos sinais.....	13
4.1.1	A atenuação do sinal.....	13
4.1.2	Casamento de impedância	14
4.2	Sinalização caso ocorra falta de energia	16
4.3	Tecnologia de Transmissão de Dados.....	16
4.3.1	ETHERNET	17
4.3.2	WI-FI.....	22
4.3.3	Ethenet versus Wi-Fi	25
4.4	O Sistema de monitoramento	27
4.4.1	Caracterização	27
4.5	INTERFACE	29
4.5.1	ICD (Interface de Captação dos Dados)	29
4.5.2	EXCEL	30
4.5.3	SET (Supervisório Embratex Tecelagem)	31
4.6	ROTINA DE TESTES.....	32
4.6.1	Veracidade da informação	32
4.6.2	Envio e Recebimento de dados.....	33
4.6.3	Expansão do sistema.....	34
5	Conclusão	35
	Bibliografia.....	36

1 INTRODUÇÃO

O presente relatório refere-se ao estágio curricular desenvolvido pela aluna do curso de graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, Débora Diniz de Melo, realizado na empresa COTEMINAS S.A, unidade de Campina Grande.

O estágio foi realizado no departamento de engenharia da EMBRATEX, uma das duas unidades fabris da COTEMINAS na cidade, tendo como supervisor, por parte da empresa, o engenheiro de operação e manutenção Sérgio Pereira Fernandes Torres, e, na UFCG, como orientador, o professor Alexandre Cunha Oliveira. O estágio se desenvolveu no período de 8 de janeiro de 2013 a 8 de maio de 2013 como objetivo o projeto de um supervisório de monitoramento das máquinas de tecelagem TSUDAKOMA ZAX, bem como sua implantação.

1.1 APRESENTAÇÃO DO TEXTO

Além deste capítulo introdutório, o trabalho é composto por cinco capítulos, descritos a seguir:

No capítulo 2 descreve-se a empresa onde se realizou o estágio, além da sua divisão organizacional.

No capítulo 3 é feita uma descrição da planta a ser automatizada.

No capítulo 4 apresentam-se todas as etapas de implementação do projeto.

No capítulo 5 é apresentada a conclusão.

1.2 OBJETIVOS DO ESTÁGIO

- Apresentar ao aluno o mercado de trabalho oferecendo a oportunidade de uma transição menos impactante da vida estudantil para a vida profissional.

- Oferecer uma oportunidade de conhecimento prático, amadurecendo toda a teoria vista na sala de aula.
- Possibilitar ao profissional recém-formado a oportunidade de ser inserido no corpo de trabalho da empresa.

1.3 DESCRIÇÃO DAS ATIVIDADES REALIZADAS

Durante a atividade do estágio a aluna desenvolveu um sistema supervisor das máquinas de tecelagem TSUDAKOMA ZAX, compreendendo desde as etapas de elaboração dos pré-requisitos até a implantação parcial do sistema, contando-se inclusive com uma interface de captação das informações.

O supervisor encontra-se instalado atualmente em 3 máquinas, esperando a liberação para a expansão para as demais 440.

As etapas que englobaram a elaboração de tal projeto foram:

- Conhecimento do processo produtivo;
- Elaboração da especificação técnica;
- Familiarização com o maquinário;
- Estudo dos sinais da máquina;
- Estudo da viabilidade da automação, confrontando suas vantagens e desvantagens diante a problemática;
- Compra de material;
- Implantação do supervisor;
- Estabelecimento de uma rotina de testes;
- Análise dos resultados obtidos com o sistema.

2 A EMPRESA: COTEMINAS S.A

2.1 A HISTÓRIA DA EMPRESA

Foi em um lar tipicamente mineiro que em 17 de outubro de 1931, em um povoado chamado Itamuri, município de Muriaé, Minas Gerais, nasceu o filho de um pequeno comerciante e uma dona de casa; José de Alencar, fundador do que viria a ser hoje o maior produtor têxtil-lar do mundo, a COTEMINAS.

Em 1949, aos 18 anos de idade, Alencar tomou dinheiro emprestado de seu irmão Geraldo e abriu sua loja de tecidos, "A Queimadeira" em Caratinga, Minas Gerais. Na sequência, começou a fabricar e vender roupas baratas. Contudo, com a morte de seu irmão, assumiu o controle da empresa União dos Cometas, uma empresa têxtil atacadista, que Geraldo havia fundado juntamente com três amigos. Alencar comprou as partes de seus parceiros e em 1965 fundaria a Wembley SA

Somente em 1967, com o apoio da Superintendência de Desenvolvimento do Nordeste (Sudene) e do Banco de Desenvolvimento de Minas Gerais (BDMG), José de Alencar – em parceria com o conceituado empresário e Deputado Luiz de Paula Ferreira, da área de beneficiamento de algodão fundou, em Montes Claros, a Companhia de Tecidos Norte de Minas, COTEMINAS. Todavia, a inauguração da fábrica de fiação e tecidos, a mais moderna em sua época, deu-se apenas em 1975.

Josué Christiano Gomes da Silva, filho de José Alencar, é, na atualidade, o executivo titular da Coteminas. Passou a exercer, em 1996, a Superintendência Geral da Companhia, até então ocupada por José Alencar desde a constituição da empresa em 1967. O crescimento da empresa, sob seu comando, tem sido objeto de admiração por parte de todos que militam no ramo têxtil, no Brasil e até mesmo no exterior.

Hoje a COTEMINAS fabrica e distribui produtos que ostentam conceituadas marcas de sucesso no mercado, como: *Artex, Santista, MMartan, Calfat, Garcia, Íris, Jamm e Atitude*.

A maior aposta de Gomes da Silva é o ataque em várias frentes, atacadista e varejista. As redes de lojas *M.Martan* e *Artex*, crescem em ritmo recorde: 30 unidades foram abertas em 2012, somando 250 em todo o país.

Hoje com 15 fábricas no Brasil, 5 nos EUA, uma na Argentina e uma no México, mais de 15 mil colaboradores, a COTEMINAS conta com a satisfação dos mais exigentes clientes no mercado brasileiro e internacional.

2.2 A COTEMINAS CAMPINA GRANDE

Inaugurada em 1997, a COTEMINAS de Campina Grande está situada em uma área de 130.000 metros quadrados, localizada no distrito industrial do Ligeiro na BR-230 – Alça Sudoeste (Figura 01).



Figura 01. Unidade Fabril da Coteminas Campina Grande.

Atualmente a COTEMINAS Campina Grande – PB produz fios e tecidos, abastecendo boa parte do mercado têxtil nacional.

A COTEMINAS S.A Campina Grande é composta por duas unidades fabris: Wentex e Embratex.

A unidade Wentex é constituída por:

- Sete linhas de abertura;
- Uma preparação com 105 cardas e 52 passadores;
- Uma fiação com 94 Open-ends.

Já a unidade Embratex é assim constituída:

- Quatro linhas de abertura;
- Uma preparação com 64 cardas e 54 passadores;
- Uma fiação com 66 Open-end's;

- Uma tecelagem com 4 Urdideiras West Pont e 2 Macoy Ellisson, 3 Engomadeiras Ira Griffin e 1 West Point e 235 teares Picanol Omni, 126 teares Sulzer L5300 e 83 teares Tsudakoma.

2.2.1 O PROCESSO DE PRODUÇÃO DO FIO AO TECIDO

O maquinário descrito no tópico anterior é responsável pelo processo de produção desde o tratamento da matéria-prima até o tecido. De modo que podemos descrever os setores:

1) **Abertura**

É composta por um conjunto de máquinas, que através de tubulações, limpam e "flocam" a matéria-prima.

2) **Preparação (Cardas e Passadores)**

Uma vez limpo e flocado, os materiais (algodão e poliéster), na forma de flocos, vão para as cardas, destinadas à paralelização das fibras, transformando-os em uma espécie de manta, que depois de fitada é depositada em latões.

Na sequência, se dá a homogeneização dessas fibras, conduzindo-se ao passador, que as equalizam minimizando as falhas no processo seguinte de fiação.

3) **Fiação**

A fiação é composta por máquinas chamadas "open-end", que através de cilindros abridores, transformam a fibra da fita em fio.

4) **Tecelagem**

Na Tecelagem, o grupo de Urdideiras recebe o fio e o enrola em um grande carretel metálico. Em seguida este conjunto de fios é alinhado e é encaminhado para um grupo de engomadeiras, que através de uma mistura de goma proporciona maior maciez e evita fungos, garantindo uma maior resistência ao fio.

Do carretel metálico da engomadeira, os fios vão até os teares, onde se inicia a fabricação do tecido; um conjunto de 12.700 *liços* tecendo em alta

velocidade, cada milímetro do tecido que é enrolado em um segundo carretel.

2.2.2 ESTRUTURA ORGANIZACIONAL

As duas unidades da COTEMINAS Campina Grande, Embratex e Wentex, estruturam-se organizacionalmente quanto à Manutenção Elétrica e Mecânica, subdividindo-se, também quanto à manutenção corretiva e preventiva.

Conforme apresentado através do organograma, ANEXO A, a aluna teve subordinação direta ao Gerente, Engenheiro Sérgio Torres.

2.2.3 ESTÍMULO ÀS MELHORIAS

O departamento de Engenharia da empresa em Campina Grande estimula o desenvolvimento de projetos de melhorias do processo produtivo por parte do corpo técnico de manutenção elétrica corretiva e preditiva, um exemplo disso é o evento dedicado a apresentação das ideias desenvolvidas por seus colaboradores, com intuito de aperfeiçoar o processo produtivo, e premiando seus funcionários pelas ideias implementadas.

A aluna teve a oportunidade de participar da organização do evento da II Semana de Inovação e Melhorias, estando envolvida com ideias implementadas pelo setor do laboratório de eletrônica da Empresa (Figura 02).



Figura 02. II Semana da Inovação: apresentação de projetos de melhorias e inovação.

3 GERENCIAMENTO DAS INFORMAÇÕES DO PROCESSO INDUSTRIAL

3.1 A PROBLEMÁTICA

O estágio foi desenvolvido no ambiente da tecelagem da Indústria (Figura 03). O desenvolvimento de um sistema de acompanhamento da produção surgiu da necessidade emergencial de acompanhar mais facilmente o processo, aumentando a confiabilidade das informações e adotando-se imediatas medidas corretivas.



Figura 03. Setor da fabrica destinado à fabricação de tecidos.

O método desenvolvido até então, antes do estágio em pauta, para conferência do funcionamento do maquinário, consistia em lógicas de relé que ativava lâmpadas para alertar o operador quanto ocorria alguma anormalidade no processo. Além disso, procedia-se a leitura manual e periódica, diretamente no LCD da máquina, feita por funcionários destinados exclusivamente à obtenção de informações sobre a eficiência; da quantidade de paradas por trama e, totais no intervalo de um turno. Essas eram direcionadas ao setor de controle da produção, e só então enviadas diariamente por e-mail à administração (Figura 04).

No setor industrial, cada interrupção do processo produtivo acarreta em considerável prejuízo.

Há, portanto, demanda pelo desenvolvimento de sistemas que possibilitem o monitoramento remoto da produção, e com isto, a avaliação dos intervalos de correção dos erros e o rendimento do setor.

Logo, mediante a obtenção remota e análise automática das informações, estes sistemas diminuem a mão-de-obra necessária à captação e processamento das características da produção.



Figura 04. Método de aquisição de dados da produção.

3.2 DESCRIÇÃO DE MAQUINÁRIO

A máquina escolhida para prototipagem é a TSUDAKOMA ZAX, que já é dotada de sensoriamento e permite uma saída distinta para cada tipo de erro, sinalizado através da mudança no nível de tensão dos sinais de sensoriamento, quando da ocorrência de um evento (Figura 05).

A sequência de tarefas realizadas pela máquina TSUDAKOMA ZAX representa o processo de formação dos tecidos a partir dos fios. Iniciando-se pelo desenrolar dos fios para formação de mantas: superior e inferior, denominadas urdume, passando pela inserção da trama por alimentadores externos, até a formação do tecido [1].



Figura 05. TSUDAKOMA ZAX.

Existe, na máquina descrita, o painel SQC, supervisor dos sinais de sensoriamento e outros status de funcionamento (Figura 06). Interceptando o sinal antes que o mesmo alimente as saídas luminosas de ocorrência de erro, podem-se obter, com o osciloscópio, os sinais de saída do sensoriamento da máquina.



Figura 06. Placa SQC: supervisor dos sinais de sensoriamento e outros status de funcionamento.

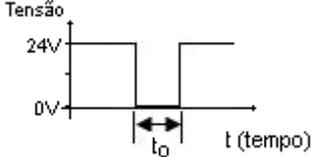
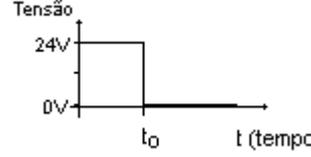
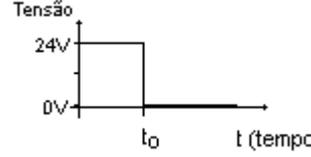
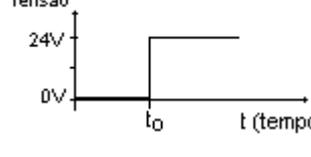
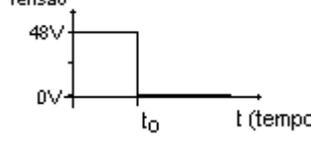
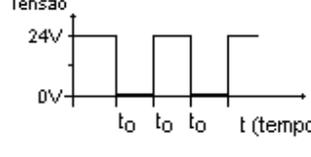
Entradas do Arduino	Descrição do funcionamento	Placa da Máquina	Entrada na placa	Sinalização (LED)	Saída Luminosa	Descrição do sinal elétrico
URDUME	Erro provocado pelo rompimento de um fio de urdume, indicado pela “queda” das lamelas.	SQC	DR	412	- O LED apaga: não ocorrência do erro mencionado; - O LED acende: ocorrência do erro;	 <p>Tensão</p> <p>24V</p> <p>0V</p> <p>t (tempo)</p> <p>t_0 = Ocorrência do evento: 100 ms</p>
TRAMA H1	Sinalização da ocorrência do fio de trama ser menor do que o esperado ou mesmo a ausência desse fio	SQC	H1S	304	- O LED apaga: não ocorrência do erro mencionado; - O LED acende: ocorrência do erro;	 <p>Tensão</p> <p>24V</p> <p>0V</p> <p>t (tempo)</p> <p>t_0 = Ocorrência do evento</p>
TRAMA H2	Sinalização da ocorrência de desperdício do fio de trama, um fio com tamanho além do esperado	SQC	H2S	303	- O LED apaga: não ocorrência do erro mencionado; - O LED acende: ocorrência do erro;	 <p>Tensão</p> <p>24V</p> <p>0V</p> <p>t (tempo)</p> <p>t_0 = Ocorrência do evento</p>
AVARIA	Sinalização da ocorrência de defeito elétrico	SQC	CE	102	- O LED apaga: o erro de avaria ocorre; - O LED acende: não se têm defeito elétrico.	 <p>Tensão</p> <p>24V</p> <p>0V</p> <p>t (tempo)</p> <p>t_0 = Ocorrência do evento</p>
STOP	Sinalização do acionamento da parada	SQC	PS3	003	- O LED apaga: o evento de parada ocorre; - O LED acende: não ocorre evento de parada	 <p>Tensão</p> <p>48V</p> <p>0V</p> <p>t (tempo)</p> <p>t_0 = Ocorrência do evento</p>
BATIDAS	Sinaliza o evento “batida do pente”	SLC	CPK	CPK	- O LED apaga: não está ocorrendo a batida do pente; - O LED acende: a batida do pente ocorre	 <p>Tensão</p> <p>24V</p> <p>0V</p> <p>t (tempo)</p> <p>t_0 = Ocorrência do evento</p>

Tabela 1. Representação dos sinais de erro e localização de ponto de medição.

Ao fim dos experimentos, pôde-se realizar um guia resumo de como obter as informações desejadas, como demonstrado através da Tabela 01.

Na primeira coluna, ilustram-se os tipos de erro analisados. Na segunda coluna, têm-se uma descrição concisa de cada evento. Outra informação imprescindível para intervenções futuras é a documentação da terceira e quarta colunas: a placa e qual o ponto do circuito na mesma onde é possível se obter os sinais desejados. Na sequência,

relata-se à conferência da ocorrência do evento, através das saídas luminosas da placa, seguido pela descrição de seu comportamento uma vez ocorrido o evento. Na última coluna, têm-se a representação dos sinais visualizados no osciloscópio.

Como mostrado na última coluna da Tabela 1, para o sinal de Urdume, antes da ocorrência do evento, a tensão de leitura deve ser estável em 24 V, na ocorrência do evento t_o , o valor comuta para zero durante 100 ms, retornando para 24 V uma vez decorrido esse tempo. Já para os sinais da Trama H1 e Trama H2 a tensão inicial de leitura era estável em 24 V, comuta para zero e é mantida em zero até que o erro seja removido da máquina. Para o sinal de Stop, o processo é basicamente o mesmo, mudando apenas que sua comutação é para 48 V, ao invés de 24 V. Quanto ao caso de defeito elétrico, descrito como Avaria, ocorre de forma antagônica, em que o estado natural desse sinal é zero, ocorrendo o evento em t_o , esse valor comuta para 24 V. Para o sinal de Batidas, em que são contabilizadas as batidas do pente da máquina, cada vez que ocorre o evento o sinal vai de zero para 24 V.

4 CARACTERIZAÇÃO DOS SISTEMAS PROPOSTOS

Mediante as especificações fornecidas por parte da equipe técnica e de produção, foram propostas duas arquiteturas para o sistema de aquisição de dados, modificando-se apenas o método de transmissão de dados quanto a estrutura de Rede.

O primeiro sistema proposto, utilizou o conjunto placa Arduino Uno R3, processando os dados da máquina, juntamente com o Shield Wi-Fi, que por sua vez promove o envio de pacotes sem fio, por meio de ondas de rádio. O Shield recebe os sinais, decodifica e os emite, através de sua antena, para um ponto de acesso, que encaminha os dados para um computador remoto, também conectado à internet (Figura 07).

O mesmo procedimento foi sugerido substituindo-se a placa de transmissão de dados Wi-Fi, assim sendo, toda interligação de rede dá-se de maneira cabeada, interligando-se os nós do sistema por Switches (Figura 08).

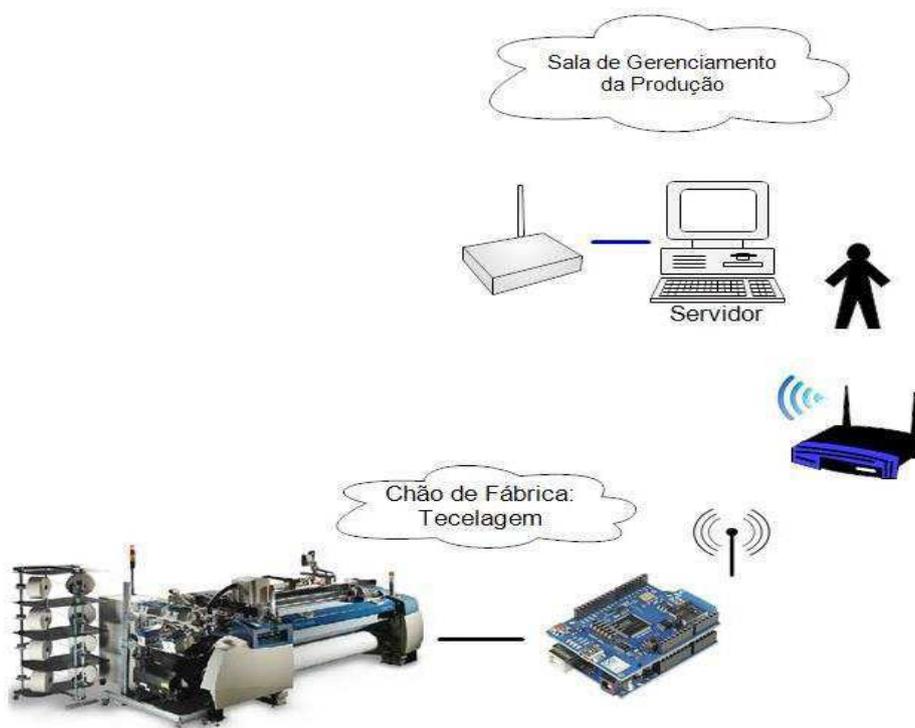


Figura 07. Sistema proposto com transmissão de dados via Wi-Fi.

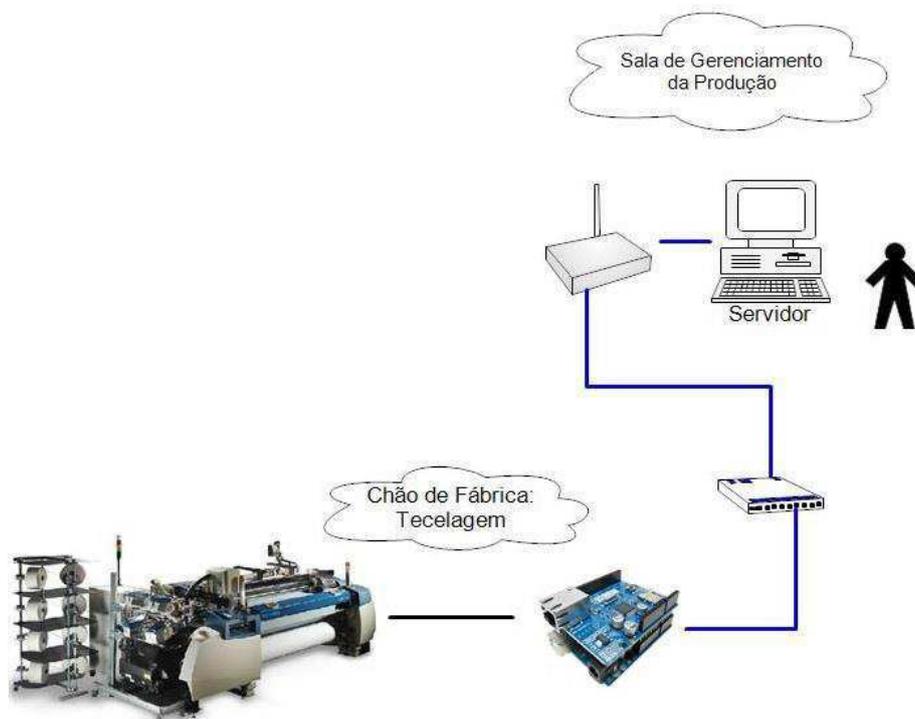


Figura 08. Sistema proposto com transmissão de dados cabeada.

4.1 IMPLEMENTAÇÃO DA ADAPTAÇÃO DOS SINAIS

4.1.1 A ATENUAÇÃO DO SINAL

Pelas características dos sinais obtidos da máquina, foi suficiente fazer a atenuação do sinal de tensão para valores adequados à tolerância dos pinos digitais da placa Arduino.

A tensão que pode ser submetida a entrada de uma placa Arduino é de 5 V, enquanto a corrente máxima permitida é de 40mA. Contudo, os níveis máximos de tensão obtidos da máquina são de 24 V para quase todas as entradas e 48 V, para o caso da ocorrência do evento Stop.

Um método eficiente, prático e barato para essa atenuação é um mero divisor de tensão resistivo (Figura 09), conforme apresentado a seguir.

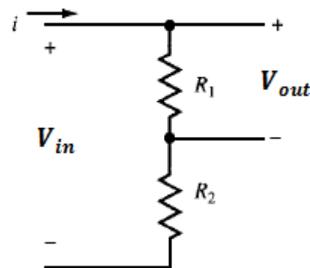


Figura 09. Divisor de tensão.

(Fonte: NAHVI, Mahmood; EDMINISTER, Joseph A., 2003, p.28).

$$V_{out} = \left(\frac{R_1}{R_1 + R_2} \right) \cdot V_{in} , \quad (1)$$

A tensão a ser obtida na saída do redutor, V_{out} , deverá ser um valor menor do que 5 V, em termos de proteção da placa, e maior que 3 V, para que o sinal seja satisfatoriamente interpretado como nível lógico alto na leitura da porta de entrada digital do Arduino.

Desse modo, tem-se o valor de V_{in} como 24 V para a maioria das entradas, e V_{in} como 48 V para o caso de Stop. De modo que, ao atribuímos um valor grande de

resistência para o R_2 , respeitando os limites de corrente, é possível obter o valor de R_1 , e conseqüentemente todos os parâmetros desejados, como demonstrado nas linhas seguintes.

Para $V_{in} = 24V$ e $V_{out} = 4,56V$, adotando-se um valor para $R_2 = 4,7 k\Omega$, substituindo-se tais valores na Equação (1), temos que $R_1 = 18,8 k\Omega$.

Para esses valores, a corrente de saída atende os valores limites satisfatórios, para o caso supracitado que é de $40mA$.

O mesmo procedimento é adotado para $V_{in} = 48V$ e $V_{out} = 4,72V$, adotando-se um valor para $R_2 = 4,7 k\Omega$, em que se obtêm $R_1 = 43 k\Omega$. Para esses valores, o limite de corrente também é atendido.

Desse modo, utilizando apenas resistores, desenvolveu-se um condicionador de atenuação de tensão a ser ligado entre os sinais da máquina e o Arduino (Figura 10).

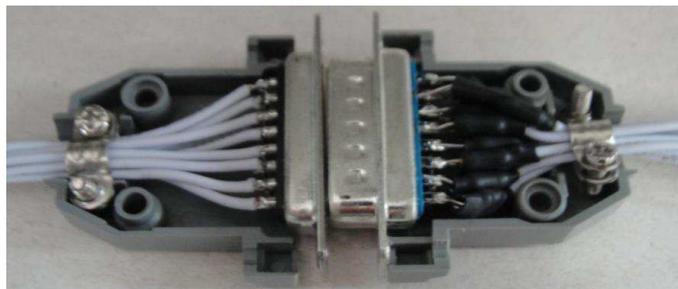


Figura 10. Atenuador de tensão, utilizando resistores, a ser interligado entre a máquina e o Arduino.

4.1.2 CASAMENTO DE IMPEDÂNCIA

O casamento de impedâncias consiste na maximização de potência ativa transferida à carga e também na eliminação de reflexo de sinal da carga.

O circuito da Figura 11, para $R_f = R_s$ tem como característica um ganho de saída unitário, com a saída apresentando mesma amplitude, polaridade e fase que a entrada [2].

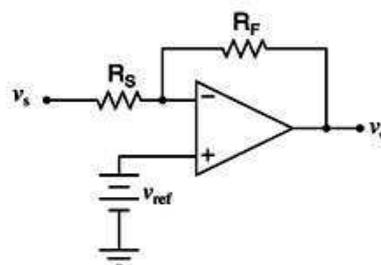


Figura 11. Buffer ou seguidor unitário.
(Fonte: WENDLING, Marcelo, 2010, p.20).

A utilização deste circuito é importante para isolar um circuito de baixa impedância de saída, do Arduino, que comparativamente tem uma baixa impedância de entrada, obtendo-se uma topologia na forma da Figura 12.

Tem-se, portanto, um desacoplamento do sinal aplicado ao Arduino, e garantia da confiabilidade do valor de tensão a ser mensurado no pino de entrada.

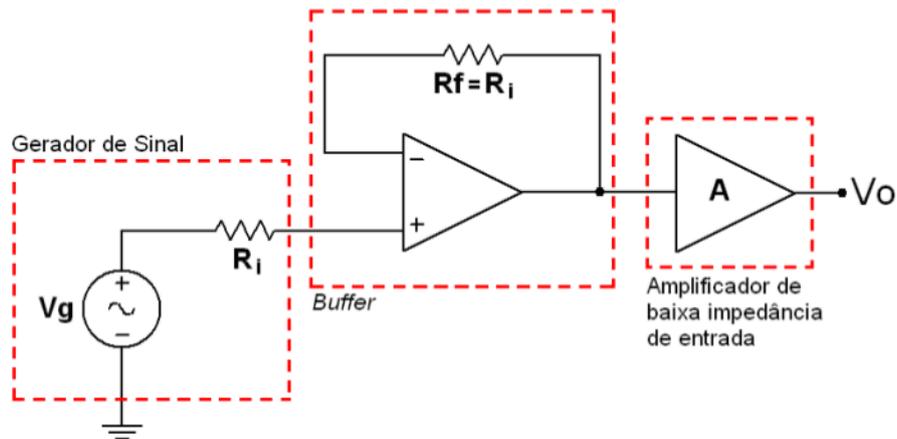


Figura 12. Casamento de Impedância.
(Fonte: WENDLING, Marcelo, 2010, p.20).

Para implementação do conceito, foi utilizado o ULN2003, que contém arrays de pares de transistor Darlington que consistem na interconexão de transistores de forma a obter uma configuração de alta impedância de entrada e alto ganho de corrente.

O circuito final é apresentado na Figura 13, assim como a foto do dispositivo real na Figura 14.

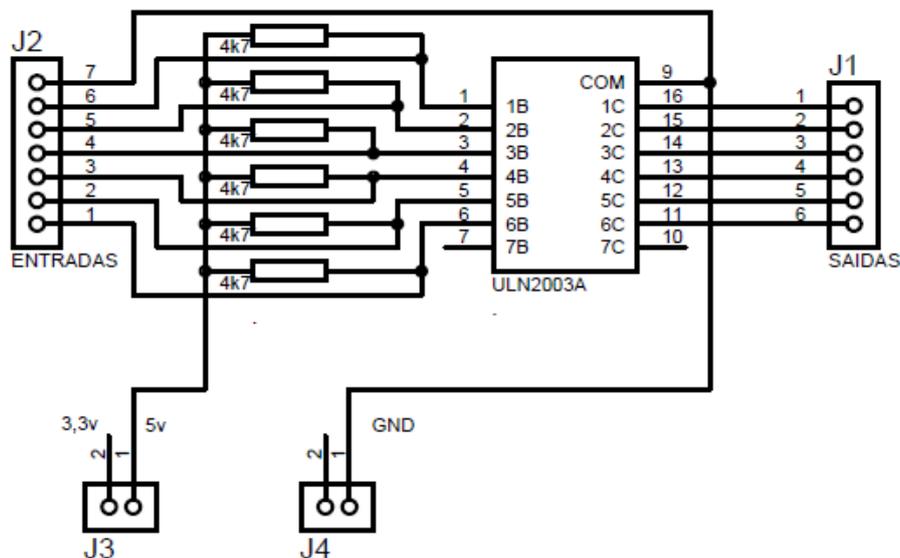


Figura 13. Simulação do circuito do Shield elaborado

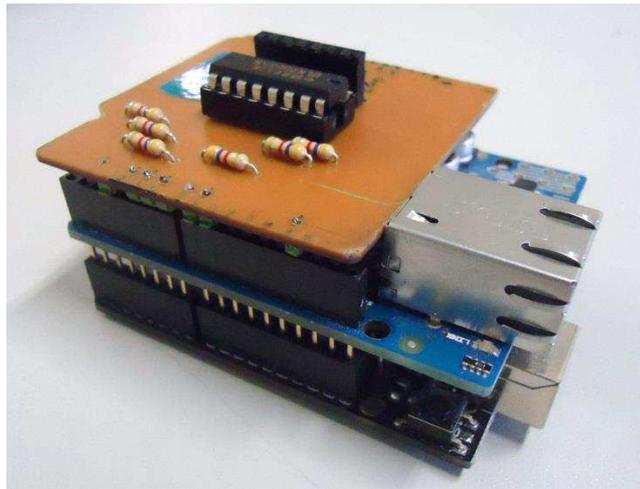


Figura 14. Shield desenvolvido de adaptação de impedância

4.2 SINALIZAÇÃO CASO OCORRA FALTA DE ENERGIA

Uma exigência da empresa foi que fosse implementada uma visualização, quando a placa Arduino fosse desconectada de sua fonte de alimentação. Para tal, utilizou-se uma entrada analógica, avaliando-se quando o valor de tensão é inferior a 3,3V. Assim, dado o evento, deve-se enviar ao servidor, um aviso de falta de energia e assim que a mesma for reestabelecida, envia-se uma mensagem complementar de que a energia foi restaurada.

Uma fase de teste foi executada com sucesso, onde se alterando a tensão na entrada, foi observado que as informações são enviadas corretamente, conforme apresentado.

Atualmente, a entrada analógica encontra-se interligada aos 5V fornecidos pela própria plataforma, não tendo sido ainda interligada à fonte de alimentação. Em uma versão posterior, pretende-se elaborar tal conexão física.

4.3 TECNOLOGIA DE TRANSMISSÃO DE DADOS

No decorrer desse estágio, a aluna foi levada a trabalhar com tecnologias de rede, de maneira a poder efetivamente estabelecer a interconexão de comunicação entre equipamentos industriais. A rede é definida pelo canal físico, através do qual os dados irão ser transmitidos, aliado ao protocolo que organiza as regras que os dispositivos devem seguir para que a comunicação seja realizada com sucesso.

Com o intuito de fornecer opções de implementação, a serem comparadas quanto ao seu desempenho prático, a autora se dispôs a desenvolver os tipos de implementação de redes: Ethernet cabeada e sem fio (Wi-Fi).

4.3.1 ETHERNET

Rede desenvolvida pela Xerox PARC em 1973. É distribuída em pontos de rede capazes de enviar e receber mensagens num determinado formato. Cada ponto tem uma identificação única no mundo, chamado de MAC (Media Access Control) com 48 bits, tendo todo equipamento que utiliza rede Ethernet uma numeração específica que o define na rede [3].

No princípio, a Ethernet não foi considerada ideal para indústria, dado quando uma colisão era detectada, se procedia a uma contagem aleatória de tempo para uma nova transmissão, não havendo assim garantia de que os dados fossem realmente transmitidos. Com esta estrutura a ocorrência de colisões sucessivas potencialmente poderia provocar a perda de informação durante o tempo em que os conflitos ocorrem.

Passou-se a aplicar a tecnologia em ambiente industrial com o surgimento do Switch, uma vez que o mesmo gerencia o controle de colisões, através da utilização de várias portas com buffer. Havendo várias portas, pode-se transmitir o frame de uma porta e aguardar a transmissão da outra.

A informação pode ser transmitida como bit na forma elétrica, luminosa e eletromagnética. Estando os três padrões presentes na indústria, os conectores são outro aspecto relevante, sendo de grande importância, podendo garantir proteção à umidade, poeira e outras situações comuns no chão-de fábrica [4].

O projeto de estágio implementado, por estar em fase de teste, utilizou cabos e conectores simples, de uso não específico para ambiente industrial. Porém, há interesse de futuramente alterar a configuração.

1.1.1.1 SHIELD ETHERNET

O Arduino Ethernet Shield permite que um Arduino seja conectado à internet, utilizando cabos de rede. Pode atuar como um servidor, aceitando conexões de entrada ou cliente, processando saídas (Figura 15). Até quatro conexões simultâneas (de entrada

ou saída ou uma combinação) são suportadas. Ele é baseado no chip Wiznet W5100 que fornece uma biblioteca de network (IP) que suporta tanto TCP como UDP [7].

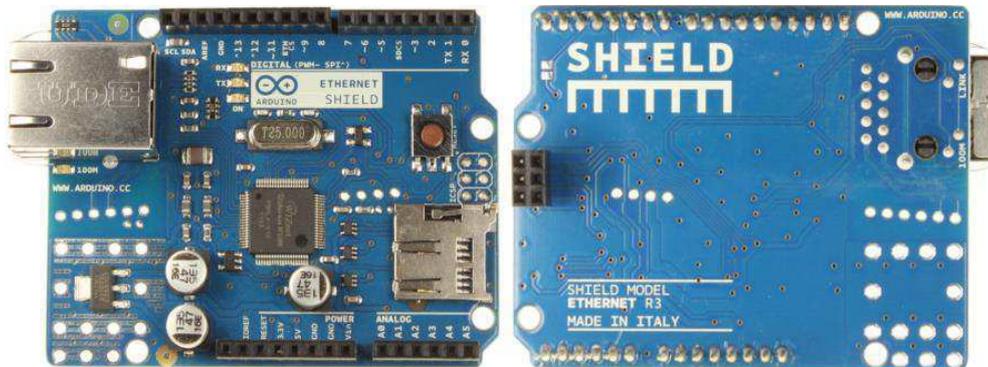


Figura 15. Shield Ethernet R3.

(Fonte: ARDUINO. Disponível em: <<http://arduino.cc/en/Main/ArduinoEthernetShield>>. Acesso em: 01 mai. 2013).

O User Datagram Protocol (UDP) é um protocolo simples da camada de transporte, que permite que a aplicação escreva um datagrama encapsulado num pacote IPv4 ou IPv6, e então o envie ao destino. Não há contudo, qualquer tipo de garantia que o pacote irá chegar ao destino. O UDP é uma escolha adequada para fluxos de dados em tempo real, especialmente aqueles que admitem perda ou deterioração de parte de seu conteúdo, tais como vídeos ou voz [5].

O Transmission Control Protocol (TCP) é um protocolo orientado a fluxos de bytes sem início e sem fim. A diferença básica entre o UDP e o TCP é o fato de que o TCP é um protocolo orientado à conexão e, portanto, inclui vários mecanismos para iniciar, manter e encerrar a comunicação, negociar tamanhos de pacotes, detectar e corrigir erros, evitar congestionamento do fluxo e permitir a retransmissão de pacotes corrompidos, independente da qualidade do meio físico [6].

Por esta razão, com o intuito de garantir confiabilidade máxima na transmissão da informação, elegeu-se como alternativa, implementar a comunicação através do protocolo TCP.

O Shield se conecta ao Arduino por barras de pinos, mantendo o layout e permitindo que outro Shield se encaixe por cima. Há também um slot para cartões micro-SD que pode ser utilizado para armazenar arquivos que estejam disponíveis na rede.

O Arduino se comunica com o Shield através do barramento SPI, que para o Arduino Uno encontra-se nos pinos digitais 11, 12, e 13. O pino 4 é destinado ao cartão

SD e o pino 10 é utilizado como SS. Estes pinos não podem ser utilizados para outras conexões (Figura 16) [7].

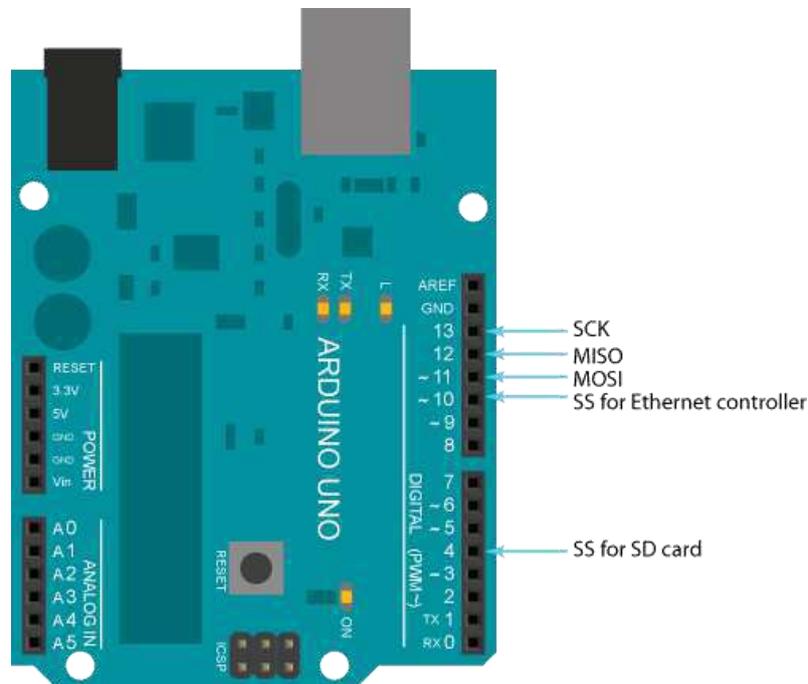


Figura 16. Esquema de pinos reservados do Shield Ethernet R3.

(Fonte: ARDUINO. Disponível em: <<http://arduino.cc/en/Reference/Ethernet>>. Acesso em: 01 mai. 2013).

Há uma série de LED's de informação:

- PWR: indica que a placa está ligada
- LINK: indica a presença de uma rede e pisca quando o Shield transmite ou recebe informações
- FULLD: indica que a conexão de rede é full duplex
- 100M: indica a presença de uma conexão de rede de 100 Mb/s (em oposição a 10 Mb/s)
- RX: pisca quando o Shield recebe informações
- TX: pisca quando o Shield transmite informações
- COLL: pisca quando colisões na rede são detectadas

Para programar a comunicação Ethernet é necessário carregar a biblioteca `Ethernet.h`, definir o endereço MAC, indicado na própria placa Arduino, e definir o IP do servidor e da placa.

```

#include <SPI.h>
#include <Ethernet.h>

byte mac[] = { 0x90, 0xA2, 0xDA, 0x0D, 0xC2, 0x95 };
byte ip[] = { 192, 168, 1, 3 };
byte server[] = { 192, 168, 1, 102 };
byte dns2[] = { 255, 255, 255, 0 };
byte gateway[] = { 192, 168, 1, 1 };

```

Em seguida, consta a instância de um objeto do servidor, acompanhada pelo número da porta do dispositivo. O servidor escuta conexões de entrada na porta especificada. O número da porta decide para onde os dados devem ser enviados.

```
status2 = client.connect(server, 50000);
```

Utiliza-se também, dentro das rotinas, a função `Client.Stop()` com intuito de desconectar o cliente do servidor, não deixando a conexão constantemente aberta, impedindo que informações indesejadas fluam pela rede.

1.1.1.2 SWITCH UTILIZADO

O Comutador ou Switch é um dispositivo utilizado em redes de computadores para reencaminhar pacotes (frames) entre os diversos nós (Figura 17).

O Switch utilizado para implementação desse projeto, apresenta apenas 8 portas, uma vez que nesse primeiro momento o sistema, a ser expandido, encontra-se implementado apenas em uma máquina. As outras especificações do Switch são indicadas a seguir:

Padrão:

IEEE 802.3 10BASE-T

IEEE 802.3u 100BASE-TX

IEEE 802.3x operação full duplex com controle de fluxo

Interface:

Oito portas 10/100 Mbps RJ-45 NWay

Suporte ao Protocolo NWay:

Velocidade da rede: modo duplex de 10Mbps e 100Mbps: full duplex e half duplex

Memória Intermediária:

128 Kbytes para o sistema completo

Taxa de Filtragem/Encaminhamento:

10Mbps: 14,880pps/14,880pps

100Mbps: 148,800pps/148,800pps

Emissão:

CE, FCC e VCCI Classe B

LED Indicadores:

Sistema: Potência, Enlace/Atividade

Umidade Operacional:

10% - 90% (não-condensado)

Temperatura de Funcionamento:

0~50 graus Celsius (32~122 graus Fahrenheit)

Tabela de Endereço MAC:

1K MAC endereço

Dimensão:

183 x 75 x 28 mm (Largura x Altura x Profundidade)

Fonte de Alimentação:

Adaptador de potência externo

O Encore 8 portas, Nway apresenta como características:

- Permite que cada porta funcione em sua velocidade ótima enquanto reduz o congestionamento de rede;
- Apresenta a operação plug-and-play que não requer nenhuma configuração de software;
- Fornece LEDs de diagnóstico no painel frontal para monitorar o status de operação das portas e de todo o sistema;
- Oferece conformidade com muitos protocolos de rede;
- Permite segmentar grandes redes em redes menores para melhor desempenho [8].



Figura 17. Instalação no Switch na tecelagem.

4.3.2 WI-FI

Uma rede sem fio é um arranjo de comunicação, onde os agentes desta comunicação conseguem trocar informações sem a necessidade de serem interligados fisicamente. Estas estruturas de comunicação podem ser utilizadas, como meio de comunicação, do ultrassom, infravermelho, das ondas eletromagnéticas etc., para trafegar a informação. [9]

As redes *wireless* locais, *Wireless Local Area Networks* (WLAN), começaram a aparecer no final dos anos 90 e passaram a ser utilizadas em diversos ambientes, sejam estes industriais, domésticos, governamentais, entre outros. Esta tecnologia surgiu com o propósito de ser uma alternativa às redes convencionais, conhecidas como cabeadas, por serem de fácil implantação e fornecerem as mesmas funcionalidades. [9]

O tipo de rede adotada, respeitando-se a abrangência, de acordo com as especificações do projeto, enumeradas adiante, foi a de uma rede WLAN. Estas redes *wireless* locais utilizam o padrão IEEE 802.11.X para se comunicarem e são frequentemente encontradas onde a mobilidade de usuários é um requisito.

O alcance depende do equipamento utilizado e, da área. Porém, este não constitui uma limitação, uma vez que se pode contornar essa limitação introduzindo sistemas de distribuição interconectados via redes cabeadas nas redes sem fio, alcançando-se uma maior área de abrangência.

- ✓ Área: 20.800 m²
- ✓ Número de pontos de acesso: 444
- ✓ Distância do servidor: 100 metros

A topologia adotada para a rede é do tipo estrela (Figura 18), onde os sinais de envio e recebimento passam sempre por um concentrador. Utilizou-se um Access Point, dispositivo em uma rede sem fio que realiza a interconexão entre todos os dispositivos móveis. Trata-se do modelo da Motorola AP-5131 (Figura 19), que apresenta robustez industrial e suporta mais de 200 pontos de acesso, com taxa de transmissão suficiente para implementação do projeto [3].

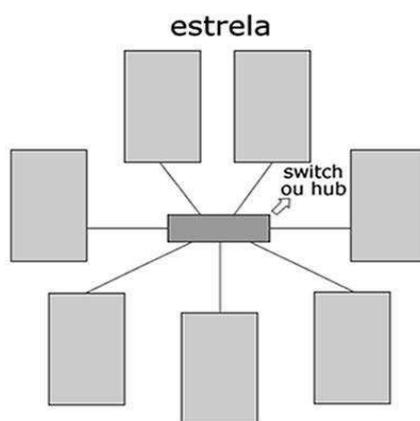


Figura 18. Topologia de rede em estrela.

(Fonte: BRASIL ESCOLA. Disponível em: <<http://monografias.brasilecola.com/computacao/topologias-url.htm>>. Acesso em: 03 mai. 2013).



Figura 19. Access Point utilizado em testes: Motorola AP-5131

(Fonte: MOTOROLA. Disponível em: <<http://motorola.com>>. Acesso em: 03 mai. 2013).

1.1.1.3 SHIELD WI-FI

O Arduino Wi-Fi shield conecta o Arduino à internet por conexão sem fios, utilizando a especificação wireless 802.11 (WiFi). É baseado no sistema wireless LAN 02.11b/g HDG104 (Figura 20). Com o Arduino, cada elemento da plataforma (hardware, software e documentação) está disponível gratuitamente.

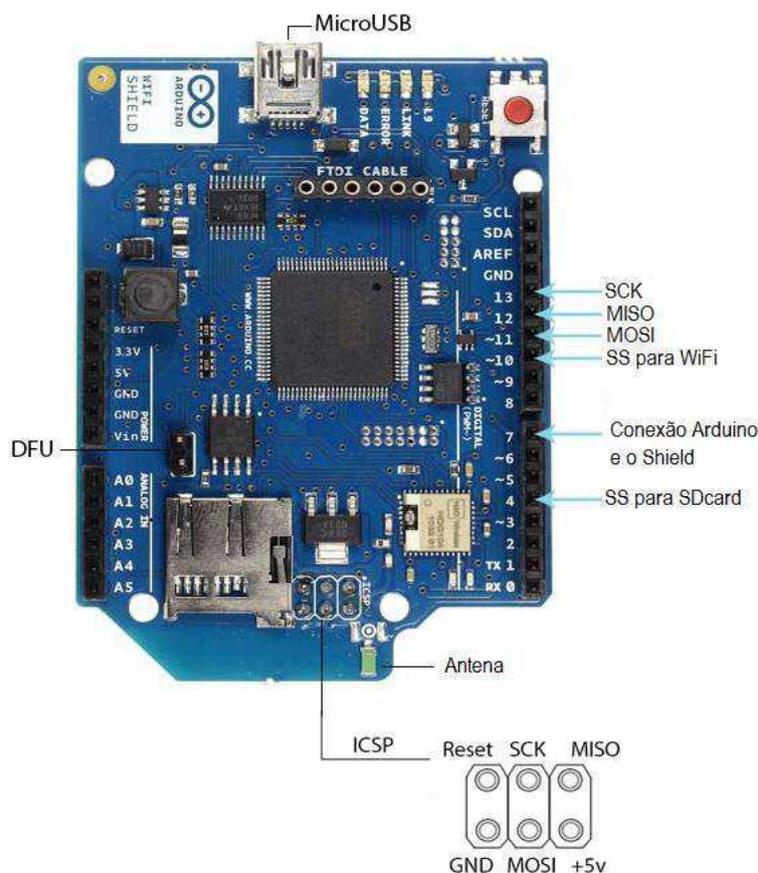


Figura 20. Shield WiFi.

(Fonte: ARDUINO. Disponível em: <<http://arduino.cc/en/Main/ArduinoWiFiShield>>. Acesso em: 03 mai. 2013).

Um Atmega 32UC3 fornece recursos de rede (IP) capaz tanto da comunicação TCP quanto UDP. Da mesma maneira que no Shield Ethernet, esta placa se conecta ao Arduino Uno através de barra de conectores.

De maneira análoga ao Shield Ethernet, há um encaixe para cartão microSD e os pinos 4, 10, 11, 12 e 13 não podem ser utilizados como entrada e saída, uma vez que são responsáveis pelo interfaceamento entre a placa Arduino Uno e o Shield.

No caso do Shield WiFi, há uma restrição adicional relativa ao pino digital 7, não podendo ser utilizado como entrada e saída, por ser relacionado à inicialização do protocolo de comunicação entre o WiFi Shield e o Arduino [8].

Este Shield pode funcionar em redes com encriptação do tipo WPA2 pessoal ou WEP. Também se pode conectar às redes abertas.

LED's de informação:

- L9 (amarelo): nível lógico ativo no pino nove;
- LINK (verde): indica conexão a uma rede;
- ERROR (vermelho): indica quando há um erro de comunicação;
- DATA (azul): indica que dados estão sendo enviados ou recebidos.

O WiFi Shield pode se conectar a redes sem fio que operem de acordo com as especificações 802.11b e 802.11g. O padrão IEEE 802.11b utiliza a técnica de espalhamento espectral DSSS para realizar a comunicação entre os dispositivos e opera na faixa de frequência de 2,4 GHz, utiliza quatro velocidades diferentes para a transmissão e recepção de dados, possibilita um número máximo de 32 clientes conectados e tem um alcance de 400 metros em lugares abertos e 500 metros em lugares fechados. Quanto ao 802.22g, utiliza o método de modulação OFDM, operando na banda ISM estreita de 2,4 GHz e pode operar em até 54 Mbps. [9]

4.3.3 ETHERNET X WI-FI

Tanto a tecnologia Ethernet quanto a Wireless permitem a troca de frames entre elementos de rede e são definidas pelo padrão IEEE, 802.3 para Ethernet e 802.11 para Wireless [3]. A diferença está no modo como os dados são transmitidos. Como já mencionado, em redes Ethernet os frames são transmitidos por meio da geração de sinais elétricos em um cabo metálico (ou sinais luminosos em fibra-óptica). Já as redes Wireless, utilizam ondas de rádio para a transmissão desses frames.

A rede Wireless apresenta inúmeras vantagens, tais como:

- Mobilidade: oferecem aos usuários acesso dos mais diversos pontos.
- Flexibilidade: apresentam maior facilidade de organização da rede, adaptando-se a reconfigurações do ambiente. Permitem a expansão.

- Menor Custo de Manutenção: o custo de manutenção é menor em relação às redes cabeadas, dado de que não há troca de cabos danificados e ao se realizar manutenção, o serviço demanda menos tempo, fazendo com que o custo de mão-de-obra seja reduzido.

Porém, em chão de fábrica essa tecnologia ainda apresenta algumas limitações, que vêm sendo tratadas, mas que, juntamente com o custo, levassem à escolha da Ethernet Cabeada para implementação. Esta última apresenta baixo custo e facilidade de configuração, além de já existir uma maior familiarização por parte do corpo técnico da empresa ao trabalhar com essa tecnologia.

Alguns dos pontos críticos quanto à implementação da WiFi em ambiente industrial são:

- Interferência na faixa de frequência utilizada: A faixa de frequência utilizada pelos dispositivos Wi-Fi é a mesma conhecida como ISM (*Industrial, Scientific and Medical*), que apresenta faixas liberadas para uso geral sem necessidade de licenciamento, obrigando a convivência entre esses dispositivos com outras fontes de RF na mesma faixa, como dispositivo Bluetooth, e até mesmo forno de micro-ondas;
- Interferência causada por múltiplos caminhos: Em ambiente fechado, quando emitido, o sinal é refletido em pisos, tetos, paredes, equipamentos e pessoas, fazendo com que esses sinais apresentem diferenças de amplitude, fase e tempo de chegada ao receptor. Desse modo, o receptor deve ser capaz de selecionar o melhor sinal e anular os efeitos dos demais.
- Atenuação do sinal transmitido: O nível de sinal que chega à antena receptora deve ser adequado para uma operação confiável e com a taxa de dados esperada. [10]

Além dos aspectos abordados, em termos de custo, como mostrado através da Tabela 02, a solução Ethernet cabeada ainda aparece com maior viabilidade.

Ethernet			WiFi		
Descrição	Qnt.	Preço	Descrição	Qnt.	Preço
Cabo de Rede Furukawa – 10/100/1000 Mbps – 24 AWG.	3000 m	R\$ 3.480,00	Access Point Motorola Ap-5131 40020 – D - WR	2	R\$ 2.140,00
Switch HP V1905 – 48 (JD994A) – 48 portas.	10	R\$ 1.175,00	Arduino Uno R3	444	R\$ 39.960,00
Arduino Ethernet sem módulo PoE	444	R\$ 87.468,00	Arduino Shield WiFi	444	R\$ 240.648,00
	TOTAL	R\$90.125,00		TOTAL	R\$282.748,00

Tabela 02. Comparativo de custo de implantação do sistema Ethernet versus Wi-Fi no setor da Tecelagem.

4.4 O SISTEMA DE MONITORAMENTO

Com a finalidade de monitorar adequadamente o processo, de modo a obter maior rendimento da produção e aumentar o potencial competitivo da Indústria, a aplicação de sistemas supervisórios vem se tornando cada vez mais comum, ao ponto de inúmeros processos poderem ser acompanhados em tempo real, com todas as variáveis do processo supervisionadas e mostradas remotamente em tempo real [11].

4.4.1 CARACTERIZAÇÃO

Um sistema de monitoramento é definido como uma interface com o objetivo de converter dados do processo de produção em dados acessíveis através de telas “amigáveis”. Diferentemente da maioria dos supervisórios, o sistema desenvolvido não atua no processo, apenas monitora.

Uma planta industrial pode ser estruturada em níveis hierárquicos (Figura 21), com cada nível há associado a um requisito quanto ao tipo de sistema a ser automatizado, conforme ilustrado na Figura 21.



Figura 21. Níveis hierárquicos de um processo industrial.
(Fonte: WENDLING, Marcelo, 2010, p.20).

O sistema proposto tem como finalidade auxiliar no processo coleta e processamento das informações, entre os níveis 2 e 3, inexistentes até então na planta.

Para tal, visando o planejamento eficiente, a estagiária utilizou os passos a seguir [12]:

1. Conhecimento do processo: para o monitoramento de um processo de automação, necessita-se reunir uma variedade de informações. Para tal, entrou-se em contato com os operadores, gerentes e corpo administrativo do sistema a ser automatizado com o objetivo de conhecer a planta e as informações necessárias que precisam ser utilizada pelo sistema e determinar as variáveis a serem monitoradas.
2. Coleta de dados: escolhe-se a apresentação apenas dos dados essenciais, mantendo o supervisório conciso, limitando o número de dados para que o desempenho total da rede não seja prejudicado pelo grande tráfego na comunicação.
3. Visualização de históricos: permite a visualização das variáveis e as ações que foram modificadas em relação ao tempo. Baseado nesses dados, o operador terá condições de avaliar situações críticas do sistema.

4.5 INTERFACE

4.5.1 ICD (INTERFACE DE CAPTAÇÃO DOS DADOS)

A interface de captação dos dados, ICD, foi implementada pela aluna, com intuito de captar os dados enviados pela placa Arduino e introduzir um campo com data/hora obtidos do SO. Os dados de funcionamento da máquina são transmitidos via rede Ethernet para um servidor TCP executando em um computador. O diagrama de funcionamento pode ser visto na Figura 22.

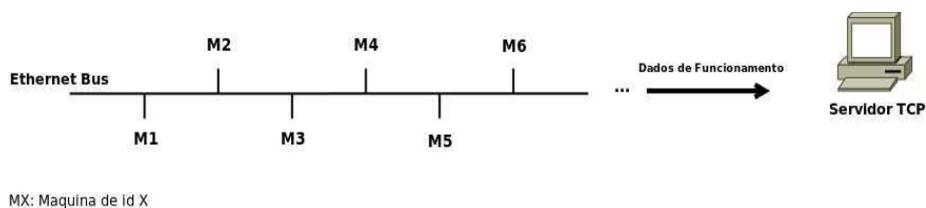


Figura 22. O diagrama de funcionamento.

O funcionamento do programa em tempo real pode ser visualizado na interface da Figura 23. O código foi desenvolvido na linguagem Python e está disponível no ANEXO B.

```

C:\Python27\python.exe
2013-05-08 00:47:04 i,b,s,1,2,a,u
2013-05-08 00:47:23 1,b,193
2013-05-08 00:47:43 1,b,198
2013-05-08 00:48:03 1,b,198
2013-05-08 00:48:23 1,b,198
2013-05-08 00:48:43 1,b,198
2013-05-08 00:49:03 1,b,198
2013-05-08 00:49:23 1,b,198
2013-05-08 00:49:43 1,b,198
2013-05-08 00:50:03 1,b,198
2013-05-08 00:50:23 1,b,198
2013-05-08 00:50:43 1,b,198
2013-05-08 00:51:03 1,b,198
  
```

Figura 23. Interface de Captação de Dados.

As informações são guardadas em um arquivo de log, criado no início da execução do programa e que é salvo a cada pacote recebido para evitar perda de informação. Com o arquivo de log torna-se mais simples o monitoramento do sistema durante períodos de tempo mais prolongados (Figura 24).

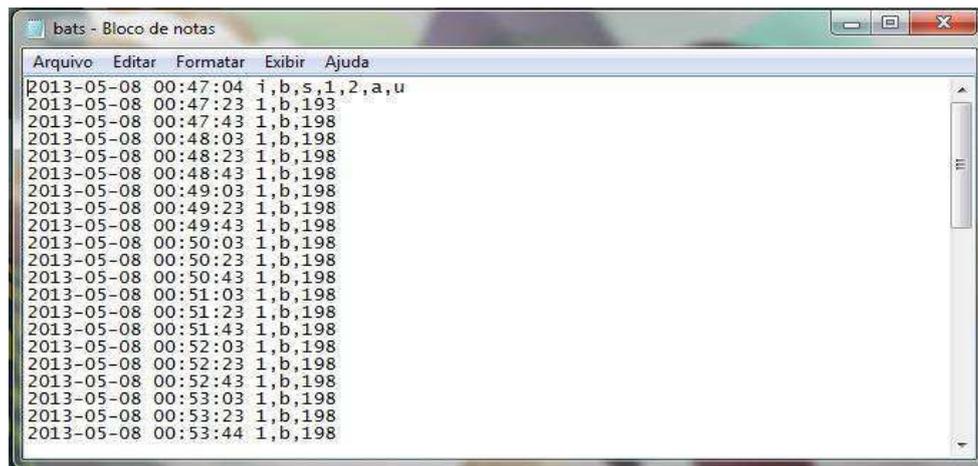


Figura 24. Log criado pela Interface de Captação dos Dados.

4.5.2 EXCEL

Além de obter as informações desejadas, foi necessário elaborar um sistema de gerência da informação industrial, responsável pela transformação desses dados em informação de negócio, atuando como “intermediário” entre o chão-de-fábrica e os sistemas corporativos de gestão da planta.

O software Microsoft Excel auxilia na organização das informações obtidas. Uma vez capturadas as informações através do SET, essas informações são carregadas em uma planilha, onde as separações por vírgulas são interpretadas como informações a serem carregadas em células distintas, através de uma planilha inteligente, também desenvolvida pela estagiária (Figura 25). O processamento, desse modo, é realizado turno a turno, por máquina, reunindo os dados relativos ao processo, tais como o rendimento.

O desempenho pode ser medido em tempo real com relatórios resumidos que permitem que ações instantâneas ou planejadas sejam tomadas de forma que a qualidade seja mantida ou melhorada, de modo a favorecer:

- Redução do tempo de ciclo de produção: identificam-se quais erros levam à parada da máquina, contornando os nós problemáticos;
- Aumentar a qualidade do produto: evitando paradas sucessivas, que aumentam a probabilidade de defeitos nos tecidos;
- Otimização dos recursos empregados na produção: Caso a incidência de erros mecânicos ou elétricos esteja acima da margem de aceitabilidade,

esses problemas podem ser contornados mais celeremente, poupando recursos;

- Diminuir custos de produção: quanto mais otimizado o processo, for em termos de mão-de-obra ou maquinário, menores serão os custos de produção;

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Data	Hora	Tipo	Bat	Stop	H1	H2	Avar.	Urd.	Temp.Par.		RPM		
2	4/1/13	14:25:00	bat	248						0:00:00	0	Batidas	497.56	
3	4/1/13	14:25:30	bat	249						0:00:00	0	Batidas Cor.	225044	
4	4/1/13	14:26:00	bat	249						0:00:00	0	Tempo Parado	225016	
5	4/1/13	14:26:30	bat	248						0:00:00	0	Stop	0:40:09	
6	4/1/13	14:27:00	bat	249						0:00:00	0	H1	0	
7	4/1/13	14:27:30	bat	248						0:00:00	0	H2	15	
8	4/1/13	14:28:00	bat	249						0:00:00	0	Avaria	0	
9	4/1/13	14:28:30	bat	248						0:00:00	0	Urdume	3	
10	4/1/13	14:29:00	bat	249						0:00:00	0	Falha	28	
11	4/1/13	14:29:30	bat	249						0:00:00	0	Paradas	18	
12	4/1/13	14:30:00	bat	248						0:00:00	0	Eficiencia	92.294%	
13	4/1/13	14:30:30	bat	249						0:00:00	0			
14	4/1/13	14:31:00	bat	248						0:00:00	0	Bat. ótima	243804.4	
15	4/1/13	14:31:30	bat	249						0:00:00	0			
16	4/1/13	14:32:00	bat	249						0:00:00	0		0:00:10	
17	4/1/13	14:32:30	bat	248						0:00:00	0		0:03:00	
18	4/1/13	14:33:00	bat	249						0:00:00	0			
19	4/1/13	14:33:30	bat	249						0:00:00	0			
20	4/1/13	14:34:00	bat	249						0:00:00	0	Leitura: Máq. PO3		
21	4/1/13	14:34:30	bat	248						0:00:00	0	Eficiência	92.2%	
22	4/1/13	14:35:00	bat	249						0:00:00	0	Par. Por Trama	15	
23	4/1/13	14:35:30	bat	249						0:00:00	0	Par. Por Urd.	3	
24	4/1/13	14:36:00	bat	249						0:00:00	0			
25	4/1/13	14:36:30	bat	248						0:00:00	0			
26	4/1/13	14:36:56	bat	186	0	1	0	0	0	0:01:36	0			
27	4/1/13	14:38:32	err	cor						0:00:00	0			
28	4/1/13	14:38:33	1	1	0	1	0	0	0	0:00:01	1			
29	4/1/13	14:38:34	err	cor						0:00:00	0			
30	4/1/13	14:39:01	bat	226						0:00:00	0			
31	4/1/13	14:39:31	bat	249						0:00:00	0			
32	4/1/13	14:40:01	bat	249						0:00:00	0			
33	4/1/13	14:40:31	bat	248						0:00:00	0			
34	4/1/13	14:41:01	bat	249						0:00:00	0			
35	4/1/13	14:41:31	bat	249						0:00:00	0			
36	4/1/13	14:42:01	bat	249						0:00:00	0			

Figura 25. Planilha Inteligente.

4.5.3 SET (SUPERVISÓRIO EMBRATEX TECELAGEM)

Com o intuito de aperfeiçoar a visualização quanto das principais informações captadas em tempo real, foi desenvolvido um sistema de processamento automático dessas informações integrado diretamente ao ICD, denominado SET, Supervisório Embratex Tecelagem. O Código também foi desenvolvido em Python, ANEXO C.

```

$python map_bat.py bats.txt 1
-----
DADOS OBTIDOS PELO SET
TEMPO PARADO: 0:39:35
BATIDAS: 225044
FALHAS: 28
BATIDAS CORRIGIDAS: 225016
EFICIENCIA: 92.2936583589%
PARADAS: 18
STOP: 0
H1: 15
H2: 0
AVARIA: 0
URDUME: 3
-----
DADOS OBTIDOS NA MAQUINA
EFICIENCIA: 92.2%
Par.Por Trama: 15
Par.Por Urd: 3
-----

```

Figura 26. Apresentação das informações em tempo real: SET.

4.6 ROTINA DE TESTES

Vários testes foram implementados com o intuito de verificar a confiabilidade do sistema, sua robustez. A verificação é realizada a partir do:

1. Confronto entre os dados obtidos no Arduino e na máquina;
2. Robustez quanto ao envio e recebimento dos dados;
3. Expansão do sistema para um maior número de máquinas;

4.6.1 VERACIDADE DA INFORMAÇÃO

A verossimilhança da informação foi confirmada confrontando-se os dados coletados na máquina com os dados captados pelo sistema no intervalo de um turno. Esse processo se deu reiteradas vezes, como apresentado na documentação.

A eficiência do sistema implementado é, portanto, confirmada através da equivalência dos dados apresentados.

4.6.2 ENVIO E RECEBIMENTO DE DADOS

O sistema apresentou inicialmente inúmeras deficiências quanto ao envio e recebimento de dados, enviando a informação corretamente durante um intervalo de tempo razoável, mas deixando de enviar essas informações e não voltando sem que antes disso a placa fosse reinicializada.

Para identificar em qual dos nós estaria o problema, a aluna executou uma rotina de teste, iniciando sua verificação quanto à implementação do programa, em que se executou:

1. “*Resetar*” a placa e aguardar, analisando-se quanto tempo demoraria para que o envio de dados fosse interrompido novamente.
2. Retirar o cabo Ethernet, deixando desconectado durante alguns segundos e conectando-o novamente. Aguarda-se um intervalo de tempo de 10 minutos e reinicia-se o processo, durante 1 hora, analisando-se se em algum momento o envio dos dados foi reestabelecido;
3. Desligar o Arduino e ligar novamente, verificando-se ao limpar totalmente os registradores do Arduino, se o problema persistia.

Em todos os casos, observou-se que a conexão dos dados não era reestabelecida.

Em um segundo plano, os testes foram executados com o sinal sendo gerado diretamente pela placa Arduino, em uma Rede individualizada. A partir de então, notou-se que uma vez corrigidas às deficiências quanto à implementação do software quanto aos aspectos descritos acima, o programa enviava e recebia dados permanentemente, sem interrupção.

Verificou-se assim, que o mesmo não ocorria quando os sinais da máquina eram gerados pelo PIC. Utilizando-se um osciloscópio, notava-se que o sinal claramente era distorcido. Para contornar esse problema, foi implementado o casamento de impedância, como já descrito.

Superado este problema, implementou-se uma rede particular, reservada apenas para a captura do status de funcionamento da máquina. Nesse último contexto, o Arduino sustentou o envio de dados de maneira eficiente, durante um longo intervalo de tempo.

O sistema foi colocado então nas máquinas, onde até o presente momento apresenta resultados satisfatórios.

4.6.3 EXPANSÃO DO SISTEMA

O sistema foi aplicado inicialmente em uma máquina, sendo expandido em um segundo momento para três, com o intuito de verificar a robustez do sistema quanto ao tratamento de pacotes na rede. Desse modo, placa a placa implementou-se a identificação das mesmas, com a sua numeração. Programou-se também a identificação de todas as mensagens a serem enviadas. De modo a testar o envio simultâneo de informações.

O tratamento dessas informações, separando-se as informações da máquina foi feito no Excel, turno a turno.

Para evitar redefinir a identificação das placas no programa sempre que houver uma placa a ser programada, coloca-se a identificação da máquina no micro SDcard, suportado na placa Ethernet.



Figura 27. Método de Identificação da placa através de Micro SDcard.

5 CONCLUSÃO

O desenvolvimento do estágio foi de extrema valia para a formação profissional da aluna, tendo a mesma uma excelente oportunidade de elaborar e implementar um projeto de importante e positiva repercussão no âmbito em que foi inserido.

A Empresa por sua vez, deu início a um projeto que provou ao longo do seu desenvolvimento sua viabilidade, tendo introduzido novas tecnologias, desconhecidas até então no meio.

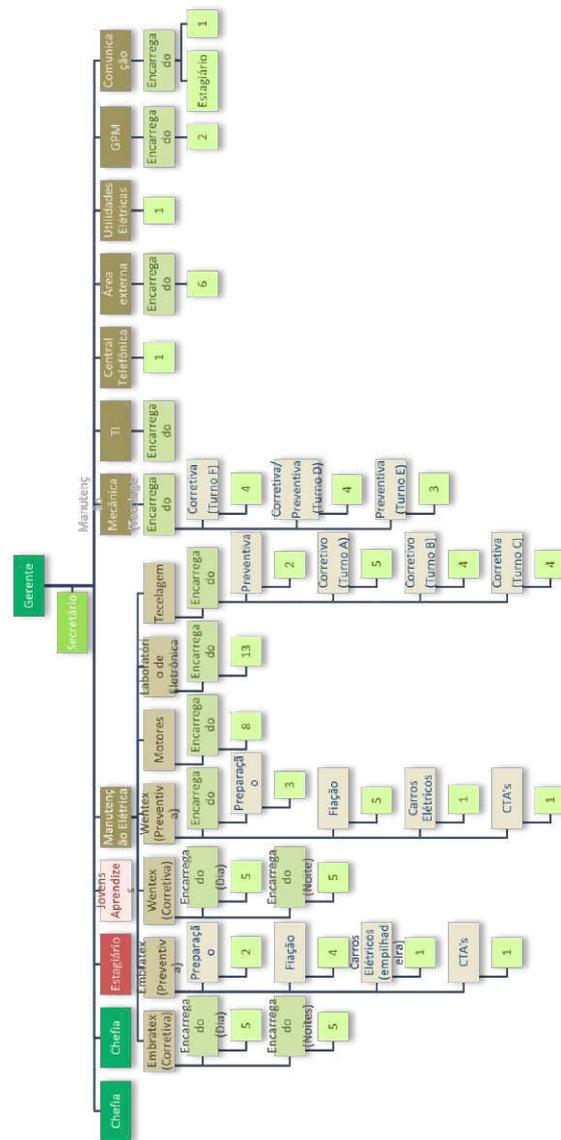
Disciplinas como Eletrônica, Arquitetura de Sistemas Digitais, Redes de Computadores, Informática Industrial e Automação Industrial foram de suma importância para o desenvolvimento exitoso das atividades realizadas durante o estágio.

A aluna praticou ainda a utilização de métodos no relacionamento interpessoal: respeito à hierarquização e administração sadia de conflitos de ideias.

BIBLIOGRAFIA

- [1] SANTOS, Felipe; GOMES, Winston. **Tear Jato de Ar**. 2010. 34p. Seminário (Curso de Operador Polivalente de Tecelagem Plana). SENAI de Americana, São Paulo, 2010.
- [2] WENDLING, Marcelo. **Amplificadores Operacionais**. 2010. 44p. Apostila sobre Amplificadores Operacionais. Universidade Estadual Paulista, Guaratinguetá, 2010.
- [3] LOPES, Marco A.M. **A importância de Sistemas Supervisórios no Controle de Processos Industriais**. 2009. 49p. Monografia (Curso de Engenharia de Controle e Automação).Universidade Federal de Ouro Preto, Minas Gerais, 2009.
- [4] LUGLI, Alexandre; SANTOS, Max; RODRIGUES, Regina. **Redes Ethernet Industriais: Visão Geral**. 2005. Disponível em: <<<http://pt.scribd.com/doc/57418630/Artigo-2009-Redes-Ethernet-Industrais-visao-Geral-1>>. Acesso em: 15 fev. 2013.
- [5] BERNAL, Volnys B. **Protocolo UDP**. Slides da Disciplina de Tecnologia de Redes. Universidade de São Paulo, 2000. Disponível em: <<http://www.lsi.usp.br/~volnys/courses/tecredes/pdf/11udp-pb.pdf>>. Acesso em: 24 abr. 2013.
- [6] VENTURA, Cristina; SOUSA, Hugo; GOMES, Jorge. **Protocolo TCP**. Curso de Competências Básicas da Internet, Faculdade de Engenharia da Universidade do Porto. Disponível em:<<http://paginas.fe.up.pt/~goii2000/>>. Acesso em: 24 abr.. 2013.
- [7] ARDUINO. Plataforma de Prototipagem Eletrônica Open-source. **Arduino**. 2005. Disponível em: <<http://www.arduino.cc/en/Main/Products>>. Acesso em: 26 abr. 2013.
- [8] ENCORE. Comutador Ethernet Rápida.**Encore**. Disponível em: <<http://www.encoreusa.com/br/product/ENH908-NWY>>. Acesso em: 28 abr. 2013.
- [9] GUASTI, Rainer M. **Solução Wireless para redes de Automação Industrial**. 2012. 78p. Monografia (Curso de Engenharia de Controle e Automação).Universidade Federal de Ouro Preto, Minas Gerais, 2012.
- [10] CUNHA, Roberto L. R. **Revista Mecatrônica Atual: Redes Wireless em chão de fábrica**.2008. Disponível em: <<http://www.mecatronicaatual.com.br/secoes/leitura/114>>. Acesso em: 20 abr. 2013.
- [11] GOHN, Maurício. SCADA: Sistema Supervisório. A Importância da Informação na logística Integrada, 2006, 22 p.
- [12] MORAES, C. M.& CASTRUCCI, P, L. Engenharia de Automação industrial. Editora LTC, Rio de Janeiro, 2001.ABIMAQ. **A História das Máquinas**. 1.ed. São Paulo: Magma, 2006, 168 p.

ANEXO A – ESTRUTURA ORGANIZACIONAL



ANEXO B – CÓDIGO ICD (INTERFACE DE CAPTAÇÃO DOS DADOS)

```
#!/usr/bin/env python

"""

teste.py: Esta interface capta as informacoes da maquina
apresenta na tela e gera um arquivo de log com as informacoes

"""

__author__ = "Debora Diniz de Melo"
__copyright__ = "Copyright 2013, Universidade Federal de Campina Grande"
__credits__ = ["Alexandre Cunha"]
__license__ = "GPL"
__version__ = "1.0"
__maintainer__ = "Debora Diniz de Melo"
__email__ = "debora.melo@ee.ufcg.edu.br"
__status__ = "Prototype"

"""

Inclusao das bibliotecas necessarias.

"""

from twisted.internet import reactor, protocol
from twisted.protocols import basic
from time import gmtime, strftime, localtime

""" Geracao do arquivo para gravacao do log """

myFile = open('bats.txt', 'w')

""" Determinacao do protocolo para tratamento dos dados recebidos """

class PubProtocol(basic.LineReceiver):
    def __init__(self, factory):
        self.factory = factory

    def connectionMade(self):
        self.factory.clients.add(self)
```

```
defconnectionLost(self, reason):
    self.factory.clients.remove(self)

deflineReceived(self, line):
    lineP = strftime("%Y-%m-%d %H:%M:%S ", localtime()) + line + '\n'
print lineP
    myFile.write(lineP)
    myFile.flush()

""" Tratamento dos clientes conectados """

classPubFactory(protocol.Factory):
def__init__(self):
    self.clients =set()

defbuildProtocol(self, addr):
return PubProtocol(self)

""" Inicializacao do servidor TCP """

reactor.listenTCP(50000, PubFactory())
reactor.run()
myFile.close()
```

ANEXO C – CÓDIGO DO SET (SUPERVISÓRIO EMBRATEX TECELAGEM)

```
#!/usr/bin/env python

"""

map_bat.py: Este arquivo serve para mapear as batidas

"""

__author__ = "Debora Diniz de Melo"
__copyright__ = "Copyright 2013, Universidade Federal de Campina Grande"
__credits__ = ["Alexandre Cunha"]
__license__ = "GPL"
__version__ = "1.0"
__maintainer__ = "Debora Diniz de Melo"
__email__ = "debora.melo@ee.ufcg.edu.br"
__status__ = "Prototype"

import sys
from datetime import datetime
from datetime import timedelta

FMT = '%H:%M:%S'
bs = []
bat_unit = 0
bat = 0
RPM = 497.56
stop = 0
h1 = 0
h2 = 0
avaria = 0
urdume = 0

stop_p = 0
h1_p = 0
h2_p = 0
avaria_p = 0
urdume_p = 0
```

```

falha = 0
paradas = 0
bat_otima = 490*RPM
tempo = 0

```

```

class batidas_lines:

```

```

    def __init__(self, date, time_h, info):

```

```

        self.date = date
        self.time_h = time_h
        self.info = info
        self.bat = None
        self.stop = None
        self.h1 = None
        self.h2 = None
        self.avaria = None
        self.urdume = None
        self.falha = None
        self.paradas = None
        self.valid = None

```

```

    def __repr__(self):

```

```

        return repr((self.date, self.time_h, self.info))

```

```

lines = file(sys.argv[1]).readlines()

```

```

wanted_id = (int)(sys.argv[2])

```

```

for i in range(len(lines)):

```

```

    lines_s = lines[i].split()
    b_line = batidas_lines(lines_s[0], lines_s[1], lines_s[2])

```

```

    b_line_temp = b_line.info.split(',')

```

```

    b_line.id = (int)(b_line_temp[0])

```

```

    if "err" in b_line_temp:

```

```

        if (b_line.id == wanted_id):

```

```

            tdelta = datetime.strptime(b_line.time_h, FMT) - datetime.strptime(bs[i-1].time_h, FMT)

```

```

            tdelta = tdelta.seconds

```

```

            tempo += tdelta

```

```

    elif "id" in b_line_temp:

```

```

        pass

```

```

    elif "bat" in b_line_temp:

```

```

        if (b_line.id == wanted_id):

```

```

            b_line.bat = b_line_temp[2]

```

```

            bat += (int)(b_line.bat)

```

```

        if ((int)(b_line.bat) <= 1):

```

```

            bat_unit += (int)(b_line.bat)

```

```

            b_line.valid = 0

```

else:

```
if(b_line.id == wanted_id):
    b_line.bat = (int)(b_line_temp[1])
    b_line.stop = (int)(b_line_temp[2])
    b_line.h1 = (int)(b_line_temp[3])
    b_line.h2 = (int)(b_line_temp[4])
    b_line.avaria = (int)(b_line_temp[5])
    b_line.urdume = (int)(b_line_temp[6])
```

```
    bat += b_line.bat
```

```
if (b_line.bat <= 1):
```

```
    bat_unit += b_line.bat
```

else:

```
if(not b_line.stop==stop_p):
```

```
    stop += 1
```

```
elif (not b_line.h1==h1_p):
```

```
    h1 += 1
```

```
elif(not b_line.h2==h2_p):
```

```
    h2 += 1
```

```
elif(not b_line.avaria==avaria_p):
```

```
    avaria += 1
```

```
elif(not b_line.urdume==urdume_p):
```

```
    urdume += 1
```

```
    h1_p = b_line.h1
```

```
    h2_p = b_line.h2
```

```
    avaria_p = b_line.avaria
```

```
    urdume_p = b_line.urdume
```

```
    stop_p = b_line.stop
```

```
bs.append(b_line)
```

```
paradas = h1 + h2 + urdume + stop + avaria
```

```
tempo = tempo + paradas*10
```

```
print"-----"
```

```
print"DADOS OBTIDOS PELO SET"
print"TEMPO PARADO:", timedelta(seconds=tempo)
print"BATIDAS:", bat
print"FALHAS:", bat_unit
diff = bat - bat_unit
print"BATIDAS CORRIGIDAS:" , diff
print"EFICIENCIA:", (str)(diff*100/bat_otima)+"%"
print"PARADAS:", paradas
print"STOP:", stop
print"H1:", h1
print"H2:", h2
print"AVARIA:" , avaria
print"URDUME:" , urdume
print"-----"
print"DADOS OBTIDOS NA MAQUINA"
print"EFICIENCIA:", "92.2%"
print"Par.Por Trama:", 15
print"Par.Por Urd:", 3
print"-----"
```