

N. CLASS.	M005-133
CUTTER	F978 _s
ANO/EDIÇÃO	2015

CENTRO UNIVERSITÁRIO DO SUL DE MINAS - UNIS-MG
BACHARELADO EM SISTEMAS DE INFORMAÇÃO
VICENTE DE PAULA SOUZA FILHO

SISTEMAS SUPERVISÓRIOS UTILIZANDO LINGUAGEM DE PROGRAMAÇÃO

C#

Varginha/MG

2015

FEPESMIG

VICENTE DE PAULA SOUZA FILHO

SISTEMAS SUPERVISÓRIOS UTILIZANDO LINGUAGEM DE PROGRAMAÇÃO

C#

Monografia apresentada ao curso de Bacharelado em Sistemas de Informação do Centro Universitário do Sul de Minas – UNIS/MG como pré-requisito para obtenção do grau de bacharel, sob orientação do Prof. Ms. Rafael Rodrigues de Souza.

**Varginha/MG
2015**



VICENTE DE PAULA SOUZA FILHO

**SISTEMAS SUPERVISÓRIOS UTILIZANDO LINGUAGEM DE PROGRAMAÇÃO
C#**

Monografia apresentada ao curso de Bacharelado em Sistemas de Informação do Centro Universitário do Sul de Minas – UNIS/MG, como pré-requisito para obtenção do grau de bacharel pela Banca Examinadora composta pelos membros:

Aprovado em / /

Prof. Ms. Rafael Rodrigues de Souza

Prof. Esp. Ângelo Mesquita

AGRADECIMENTOS

Agradeço a Deus pelo dom da vida e pela inteligência para realizar um trabalho desta complexidade.

Agradeço a todos de minha família que me ajudaram no desenvolvimento desta jornada.

Agradeço a minha esposa por me incentivar nos momentos difíceis.

Ao meu orientador pela sua competência e sabedoria em ensinar-nos.

Aos colegas que lutaram comigo nestes anos em busca do conhecimento. Obrigado pelo companheirismo nas horas difíceis.

RESUMO

Com a evolução dos computadores pessoais evoluíram também as linguagens de programação e a internet. Essa evolução modificou as rotinas e processos na vida das pessoas de um modo geral. Escritórios, automóveis, eletrodomésticos passaram a ser automatizados e isso fez mudar a rotina dos cidadãos. No início, na automação de escritório, a impressora substituiu a máquina de escrever. No entanto, essa evolução alcançou também a indústria e tudo começou a mudar. Evolução de processos, automatização de tarefas antes feitas manualmente, e então o processamento de dados evoluiu e melhorou a qualidade e a quantidade na produção da indústria. Processos antes sujeitos ao erro que pudesse existir na mão de obra humana passaram a ser supervisionados por programas de computador melhorando a performance do sistema. Os sistemas supervisórios surgem então juntamente com os microcontroladores executando serviços com grande precisão de tempo e medida substituindo mecanismos complexos e imprecisos. Tudo isso com precisão, baixo consumo e grande confiabilidade, pois utilizam-se de banco de dados, memórias, processadores com vários núcleos e softwares de alta tecnologia como a linguagem C#. Os sistemas supervisórios revolucionaram a indústria e o mundo de um modo geral. Tudo com uma grande precisão e eficiência.

Palavras-chave: Automação, dados, Processos, Banco de dados.

ABSTRACT

With the evolution of personal computers has also evolved programming languages and the internet. This development changed the routines and processes in people's lives in general. Offices, automobiles, home appliances have become automated and it did change the routine of the citizens. Earlier an office automation, a printer that replaced a typewriter. But these developments also reached the industry and everything began to change. Evolution of processes, automation tasks previously done manually, and then the data processing has evolved and improved quality and quantity of production industry. Processes before subject to error that may exist in the human labor came to be supervised by computer programs improving system performance. The supervisory systems then come with microcontrollers running services with great precision timing and extent replacing complex and unclear mechanisms. All with precision, low power consumption and high reliability are used as database, memories, multicore processors and high-tech software like the C # language. The supervisory systems have revolutionized the industry and the world in general. All with great precision and efficiency.

Keywords: Automation, Data, Processes, Databases.

LISTA DE ILUSTRAÇÕES

	pg
Figura 01 – Pagina Inicial do visual C# 2010.....	23
Figura 02 – Tela Inicial do Microsoft Asp.Net.....	25
Figura 03 – Arquitetura cliente servidor.....	26
Figura 04 – Diagrama eletrônico Arduino Uno.....	30
Figura 05 – Imagem da placa Arduino Uno roteada.....	31
Figura 06 – Tela do <i>Sketch</i> do Arduino Uno.....	32
Figura 07– Tela inicial da ferramenta. Navegador Chrome.....	36
Figura 08 – Sketch do Arduino- Código já carregado.....	37
Figura 09 – Foto das Placas Arduino e Ethernet Shield.....	38
Figura 10 – Foto do Arduino sendo executado comando.....	39
Figura 11 – Tela do Navegador. Comando executado e temperatura.....	40
Figura 12 – Tela do Navegador- Comando desliga e temperatura.....	40

LISTA DE TABELAS

Tabela 01 – Tabela de pinos Atmega 328P28

LISTA DE ABREVIATURAS E SIGLAS

ASP.NET	Active Server Pages da Tecnologia .Net
C#	Linguagem de programação
IBM PC	<i>International Business Machines Personal Computer</i> - IBM Computador Pessoal
DOS	<i>Disk Operating System</i> - Sistema operacional de disco
ENIAC	<i>Electronic Numerical Integrator and Computer</i>
UNIVAC	<i>UNIVersal Automatic Computer</i>
QNX	Sistema Operacional em tempo real
MAN	<i>Wide Area Network</i> - Rede de longo alcance
MAN	<i>Metropolitan Area Network</i> - Rede Metropolitana
ARPA	<i>Advanced research projects agency</i> - Agencia de pesquisa e projetos avançados
BCPL	<i>Basic Combined Programming Language</i> - Linguagem Basic
UNIX	Sistema operacional
POO	Programação Orientada a objetos
BASIC	<i>Beginner's All-purpose Symbolic Instruction Code</i> - Basic
IBM	<i>International Business Machines</i>
COBOL	<i>Commum Business Oriented Language</i> - Linguagem COBOL
XML	<i>eXtensible Markup Language</i> - Linguagem de marcação extensível
HTML	<i>HiperText Markup Language</i> - Linguagem de Marcação de Hipertexto
SGBDR	Sistema Gerenciador de Banco de Dados Relacionais
HTTP	<i>Hipertext Transfer Protocol</i> - Protocolo de transferência de Hipertexto
URL	<i>Uniform Resource Locator</i> - Localizador de recursos Uniforme
DNS	<i>Domain Name Server</i> -Servidor de Nomes
IP	<i>Internet Protocol</i> - Protocolo de internet
VB	<i>Visual Basic</i>
MCU	<i>Microcontroller Unit</i> - Unidade Microcontroladora.
LCD	<i>Liquid Crystal Display</i> - Display de Cristal Líquido
RAM	Memória de acesso Randômico
ROM	Memória de leitura somente.
EEPROM	Memória de escrita e leitura apagável eletricamente
SRAM	Memória de leitura Randômica estática.
RISC	<i>Reduced Instruction Set Computer</i> - Conjunto reduzido de instruções de computador
MIPS	Milhões de instruções por segundo

- Mhz Milhões de hertz por segundo
- PWM Modulação por largura de pulso
- LED Diodo emissor de luz
- FTDI Conversão serial
- USB Barramento Universal Serial
- IDE Interface de desenvolvimento integrado
- SCADA *Supervisory Control and Data Aquisition* - Sistema de controle e aquisição de dados.
- SAAE Serviço Autônomo de Água e Esgoto.
- WIRELESS Rede sem fio

SUMÁRIO

1	INTRODUÇÃO.....	13
1.1	Tema específico.....	13
1.2	Problema de pesquisa.....	13
1.3	Problematização.....	13
1.4	Hipóteses	14
1.5	Objetivo geral	14
1.6	Objetivos Específicos	14
1.7	Justificativa.....	15
2	Revisão da literatura.....	16
2.1	Sistemas de Informação.....	16
2.2	Sistemas Operacionais.....	16
2.3	Redes de computadores	18
2.4	Linguagens de Programação.....	19
2.5	Linguagem de Programação C#	21
2.5.1	Vantagens de Utilização da Linguagem C#.....	22
2.5.2	O Visual C# Express.....	23
2.5.3	O Ambiente de programação Visual C# 2010 Express.	23
2.6	Asp.Net.....	24
2.6.1	Vantagens da Tecnologia Asp.Net	26
2.7	Microcontroladores.....	27
2.7.1	Microcontrolador Atmega 328	28
2.8	Arduino.....	29
2.9	Sistemas supervisórios.....	32
3	MATERIAL E MÉTODOS.....	34
3.1	Coleta de dados.....	34

3.2 Materiais	34
3.3 Metodologia.....	34
4 RESULTADOS	36
4.1 Ferramenta Asp.Net.....	36
4.2 O Sketch e o hardware do Arduino	37
4.3 O Hardware em execução.....	39
5 CONCLUSÃO.....	42
REFERÊNCIAS BIBLIOGRÁFICAS	43

1 INTRODUÇÃO

Este trabalho acadêmico tem o propósito de estudar, analisar, implementar um sistema supervisorio em ASP.NET, conectando-o ao microcontrolador Arduino e apresentando uma solução para supervisão e monitoramento de processos industriais. Tal abordagem justifica-se devido à inexistência de sistemas supervisorios completos abordando o tema e a necessidade de implementação em um *hardware* de baixo custo apresentando uma solução para a indústria. O objetivo deste trabalho é não somente aprender sobre as linguagens de programação, mas também criar um sistema supervisorio para monitoração e controle de processos de produção para a indústria. Este sistema será utilizado no controle de bombas de abastecimento de água em Reservatório do Serviço de água e esgoto de Passos-MG.

1.1 Tema específico

Sistema supervisorio para a indústria.

1.2 Problema de pesquisa

Como implementar um sistema supervisorio, para controle e monitoramento de processos, utilizando linguagem de programação C# e Micro controlador Arduino?

1.3 Problematização

O propósito da pesquisa é apresentar uma ferramenta de *hardware* e *software* que possa otimizar processos na indústria. O monitoramento e controle de processos da indústria, como monitoramento de agua de reservatórios, torna-se difícil devido à distância dos reservatórios e das bombas que controlam o abastecimento de água. Assim, são necessários equipamentos que controlem o abastecimento e desliguem as bombas no momento exato. Outras variáveis também necessitam de controle como variáveis do próprio equipamento de monitoração, temperatura e pressão. O monitoramento de variáveis como tempo de trabalho de bombas de água, consumo de água, evolução do consumo, são interessantes para aperfeiçoar o monitoramento e controle.

1.4 Hipóteses

- A codificação em linguagem C# é possível com uma conexão via rede ethernet com o hardware do arduino.
- A linguagem C# é uma plataforma ampla e atenderá perfeitamente ao projeto.
- A conexão ao arduino através de rede ethernet será possível através do módulo *Ethernet Shield*.
- A programação do arduino é feita através de linguagem parecida com a linguagem C e isto facilitará muito o desenvolvimento do projeto.
- O hardware do arduino é adequado para essa proposta e tem desempenho adequado.
- Ao implementar um software supervisorio através de protocolo TCP/IP podemos supervisionar equipamentos onde a rede alcançar.
- O micro controlador arduino é um hardware básico, com código aberto, mas é um hardware limitado e deve ser usado em aplicações mais simples.
- O hardware do arduino tem baixo custo mas tem limitações como número de entradas e velocidade de operação.
- A linguagem C# não é adequada para construir um sistema supervisorio.
- Para construir um sistema supervisorio com alta capacidade, ou seja, com um grande número de entradas (50 a 100) não é recomendado o uso do arduino.

1.5 Objetivo geral

- Construir um sistema supervisorio utilizando linguagem C#, que possa monitorar processos industriais através de uma rede de computadores.

1.6 Objetivos Específicos

- Verificar, analisar, responder sobre dúvidas da linguagem C# e Asp.Net.
- Entender, praticar, aprender sobre o hardware do arduino utilizando um software supervisorio e verificar o desempenho para ajudar processos industriais.

- Verificar e entender os sistemas supervisórios analisando o seu desempenho na indústria, como tecnologia que ajuda os controles melhorando a qualidade.

1.7 Justificativa

Os sistemas supervisórios são muito importantes na indústria e esta pesquisa é um modo de aprender e entender a teoria das linguagens de programação e sua aplicabilidade. O propósito é encontrar uma solução de *hardware* e *software* para otimizar e automatizar processos como o controle de água de reservatórios. Isso é um fator importante, pois a arquitetura do hardware, a sua programação e a ferramenta de software devem respeitar os direitos autorais e propiciar um desempenho adequado e moderno que não existem no mercado nacional. Importante também salientar que nas soluções existentes no mercado nacional, os códigos fontes não estão disponíveis, pois são segredos industriais; por isso a importância do projeto.

Importante também ressaltar que o abastecimento de água urbano e nas indústrias torna-se complexo devido à instabilidade do consumo. Um equipamento de controle irá nos beneficiar com monitoramento automático e controle preciso de níveis. Além disso, é importante o aprendizado teórico de *softwares* e equipamentos de controle como sensores de nível, pressão e temperatura.

2 REVISÃO DA LITERATURA

2.1 Sistemas de Informação

Dados podem ser considerados caracteres, números, letras, dígitos que não foram tratados e, portanto, não podem ser considerados informação. Somente após ser tratado e trabalhado, um dado cria um valor agregado tornando-se útil, ou seja, transforma-se em informação útil. A informação após ser tratada por pessoas, tratada por recursos computacionais, ela pode gerar o conhecimento. Sistemas de informação são sistemas compostos por máquinas, computadores, dispositivos automáticos capazes de processar, analisar, reconhecer, manipular dados tornando-os informações úteis para os negócios, fazendo funcionar máquinas automáticas, melhorando o desempenho do sistema.

Segundo Rezende (2005) os sistemas de informação podem ajudar a organização a resolver inúmeros problemas no nível estratégico e de gestão.

Então os sistemas de informação são uma boa forma de resolver problemas dentro da organização, pois podem ajudar na tomada de decisões melhorando quesitos como qualidade, rentabilidade e segurança dos processos industriais.

2.2 Sistemas Operacionais

A evolução dos sistemas operacionais está ligada à evolução dos computadores. Antes da década de 1940, inúmeros esforços foram feitos para criar uma máquina que pudesse realizar cálculos de forma rápida e precisa. Em 1642, um matemático francês chamado Blaise Pascal inventou uma máquina de somar para ajudar seu pai na arrecadação de impostos.

Em 1944, foi construído o primeiro computador eletromecânico chamado de Mark 1. O ENIAC é considerado o primeiro computador digital eletrônico. Na década de 1950 foi construído o primeiro computador para fins comerciais, o UNIVAC. O IBM PC surgiu na década de 80 juntamente com o sistema operacional DOS. (*Disk operating system*). O sistema DOS foi a base para o surgimento do sistema Windows utilizado atualmente.

Sistemas operacionais, portanto, são camadas de softwares que fazem a interface entre o hardware e os programas aplicativos usados pelos usuários.

Segundo Silberschatz, Galvin e Gagne (2000), um sistema operacional é um componente importante em todo sistema computacional. Um sistema de computação pode ser

dividido em basicamente quatro componentes que são: o *hardware*, o sistema operacional, os programas aplicativos e os usuários.

A principal função de um sistema operacional é servir de interface entre o usuário e o hardware da máquina facilitando o acesso, controlando a diversidade de recursos como *drives*, memória, processador, organizar recursos dando prioridades pré-estabelecidas e gerenciando tempos pré-estabelecidos pelo sistema.

O sistema operacional fornece também alguns aplicativos importantes para os usuários como, por exemplo, o gerenciador Windows Explorer e o Navegador Internet Explorer.

Caso não existisse o sistema operacional, um usuário teria que conhecer a fundo, os diversos detalhes eletrônicos e de funcionamento de um computador para poder realizar algum trabalho útil com ele. O sistema operacional tem como objetivo funcionar como uma interface entre o hardware da máquina facilitando, organizando e distribuindo os recursos computacionais. O sistema operacional permite também que os usuários e programadores não tenham que conhecer a fundo os detalhes do hardware da máquina.

Os sistemas operacionais podem se subdividir em:

- Sistemas monoprogramáveis ou monotarefa.
- Sistemas multiprogramáveis ou multitarefa.
- Sistemas com múltiplos processadores.

Os sistemas monoprogramáveis ou monotarefa são os primeiros sistemas existentes e eram voltados para a execução de um único programa.

Uma outra aplicação para ser executada precisava aguardar o programa atual terminar sua execução.

Os sistemas multiprogramáveis ou multitarefa são uma evolução dos sistemas monotarefa e neste sistema os recursos computacionais são compartilhados entre as diversas aplicações.

A principal vantagem dos sistemas multiprogramáveis é a redução dos custos pois os recursos computacionais existentes são compartilhados pelas aplicações.

Os sistemas multiprogramáveis podem ainda se subdividir em monousuário e multiusuário.

Os sistemas monousuário podem executar várias tarefas ao mesmo tempo. Sistemas multiusuário são ambientes em que diversos usuários se conectam ao sistema para executar suas tarefas ao mesmo tempo.

Os sistemas com múltiplos processadores caracterizam-se por possuir várias unidades processadoras interligadas trabalhando em conjunto em uma ou várias aplicações. Com vários processadores é possível executar aplicações com fins científicos.

Entre os vários sistemas existentes na atualidade, podemos citar: Windows, Linux, QNX etc.

O Linux e o Windows são sistemas multitarefa e multiusuário e estão presentes no nosso dia a dia em estações de trabalho, redes, servidores etc. A seguir uma breve descrição das redes de computadores presentes na maioria dos sistemas seja na indústria ou nas residências.

2.3 Redes de computadores

Quando se fala em informática, nos dias atuais, é impossível não falar em redes de computadores. A internet é hoje, a maior rede de computadores existente. As redes de computadores estão presentes no nosso dia a dia, em farmácias, em supermercados, nos bancos e nas indústrias de modo geral. As redes de computadores estão mudando a maneira como fazemos negócios e o modo como vivemos. Elas surgiram das necessidades de troca de informação entre as pessoas e os negócios.

As decisões no mundo dos negócios “têm de ser tomadas de forma cada vez mais rápida e aqueles que o fazem precisam obter acesso imediato a informações precisas.” (FOROUZAN, 2010, p.3). As redes de computadores nos auxiliam na obtenção de informações relevantes e precisas.

O desenvolvimento do computador pessoal proporcionou grandes mudanças nas empresas, nas ciências e na educação. Uma mudança semelhante também está acontecendo nas comunicações e nas redes de computadores. As pesquisas em comunicações de dados e redes geraram novas tecnologias que possibilitam a troca de dados de forma cada vez mais rápida. Estamos em um mundo em que as informações são de grande valor. Quem tem as informações pode estar à frente no mundo dos negócios.

As informações são transmitidas de diversas formas podendo ser texto, letras, números, imagens, áudio, vídeo etc.

Segundo Forouzan (2010), uma rede é um conjunto de dispositivos que são conhecidos como nós, conectados por links de comunicação.

A maioria das redes utilizam processamento distribuído em que uma tarefa é dividida entre vários computadores para a sua execução. Uma rede deve ter alguns critérios e os mais importantes são desempenho, confiabilidade e segurança. Fisicamente as redes podem ser ponto a ponto ou multiponto. Uma conexão ponto a ponto fornece uma conexão dedicada entre os dois dispositivos. Em uma conexão multiponto o link é dividido por vários dispositivos.

Uma rede tem uma topologia que pode ser: malha, estrela, barramento, Anel e híbrida. As redes também se subdividem em categorias. As redes podem ser:

- Locais
- Redes de ampla abrangência WAN
- Redes de abrangência metropolitana MAN

As redes locais estão em um escritório, prédio ou campus. As redes WAN permitem a troca de dados por longas distâncias como por exemplo um país. As redes MAN estão em distâncias intermediárias como uma cidade.

A internet é uma rede que dá acesso a um amplo conjunto de informações de uma ampla variedade a partir de um computador pessoal.

A Internet surgiu primeiramente na ARPA, *Advanced research projects agency*, do Departamento de defesa dos Estados Unidos que estavam interessados em descobrir como interconectar computadores.

A Internet hoje é composta de várias redes, locais e remotas, ligadas através de diversos dispositivos de conexão e estações comutadoras. A seguir uma descrição das linguagens de programação que originaram a linguagem C#.

2.4 Linguagens de Programação

Para se escrever programas, utilizam-se as linguagens de programação. Existem atualmente centenas de linguagens de programação, mas algumas são mais utilizadas como: *Assembly*, C, C++, C#, Java etc. Essas podem se dividir em: linguagens de máquina, Linguagens *assembly* e as Linguagens de alto nível.

A linguagem de máquina é a linguagem que o computador entende. Qualquer computador pode entender somente a sua linguagem de máquina, ou seja, aquelas instruções que aquele processador instalado irá entender. As linguagens de máquina são um fluxo de

números que instruem o computador a realizar operações diversas pré-determinadas pelo processador. Uma específica linguagem de máquina pode ser usada somente em um determinado computador e é bastante incompreensível para o ser humano.

A programação em linguagem de máquina é bastante complexa, maçante e de baixo rendimento sendo, portanto, inviável sob o aspecto de tempo de desenvolvimento. Em vez de utilizar as *strings* de números que os computadores podiam entender diretamente, os programadores passaram a utilizar abreviações em inglês para representar as operações. Essas abreviações ou mnemônicos são a base das linguagens Assembler. Programas tradutores traduzem programas em *assembler* em linguagens de máquina. Um programa em *assembler* é muito mais fácil de ser entendido pelo ser humano, que um programa em linguagem de máquina.

As linguagens de alto nível surgiram como uma evolução para escrever compiladores, sistemas operacionais e softwares. O C++ evoluiu do C que veio do B e BCPL.

Segundo Deitel (2003), Martin Richards desenvolveu o BCPL em 1967 e a partir disso Ken Thompson desenvolveu a linguagem B, do BCPL e em 1970 usou o B para criar as primeiras versões do UNIX. Após isso, em 1973, Dennis Ritchie desenvolveu a linguagem C a partir da linguagem B, no *Bell Laboratories*. A linguagem C ganhou amplo reconhecimento por ser utilizada para desenvolver o sistema operacional Unix.

Os programadores preferem as linguagens de alto nível que as linguagens de máquina, pois aquelas facilitam o desenvolvimento. A medida em que as linguagens de alto nível são desenvolvidas, novos aspectos são desenvolvidos melhorando recursos das antigas linguagens.

O C++, que é uma extensão da linguagem C, foi desenvolvido por Bjarne Stroustrup no início dos anos 80 no *Bell Laboratories*. A principal característica do C++ é que esta linguagem é orientada a Objetos (POO). Isso aconteceu devido à necessidade de software novo e poderoso e a necessidade de construir softwares mais rápido. Além do C++, muitas outras linguagens foram desenvolvidas como Smalltalk criado pela Xerox, Visual Basic, Java. Segundo Deitel (2003, p.6):

Quando Bill Gates fundou a Microsoft Corporation, ele implementou BASIC em vários computadores pessoais. O BASIC (Beginner's All-Purpose Symbolic Instruction Code _Código de instruções simbólico de propósito Geral para iniciantes) é uma linguagem de programação que foi desenvolvida nos anos 60 pelos professores John Kemeny e Thomas Kurtz, do Dartmouth College, como uma linguagem para escrever programas simples.

O BASIC foi uma das primeiras linguagens de programação de alto nível e foi criada para fins didáticos. Foi muito usada na década de 1970. Embora o Visual Basic seja derivado

da linguagem de programação BASIC, ele é uma linguagem bastante diferente, com uma diversidade de recursos como POO (Programação Orientada a Objetos) e outros.

Embora centenas de linguagens de alto nível tenham sido descobertas poucas linguagens tiveram uma ampla aceitação. A IBM Corporation lançou o Fortran (*Fórmula Translator*) entre 1954 e 1957. O COBOL (*Commum Business Oriented Language*) foi desenvolvido em 1959 por um grupo de fabricantes de computador em conjunto com usuários de computador do Governo e da indústria.

Segundo Deitel (2003), ainda hoje grande parte de softwares são feitos em COBOL. O Pascal foi desenvolvido em 1960 pelo professor *Nicklaus Wirth* para fins acadêmicos e recebeu este nome em homenagem ao matemático e físico *Blaise Pascal*. A seguir, uma descrição da linguagem C# que será usada no aplicativo supervisorio.

2.5 Linguagem de Programação C#

Com o avanço das linguagens de programação como C++ e Java e dos dispositivos como o celular surgiram novos problemas e novas necessidades. Os desenvolvedores descobriram que precisavam de aplicativos baseados na WEB que pudessem ser usados na internet. Com a popularidade dos dispositivos eletrônicos móveis os desenvolvedores descobriram que os clientes estavam mais restritos aos computadores de mesa como Desktop. Então os desenvolvedores reconheceram a necessidade de softwares acessíveis e disponíveis para a variedade de dispositivos. A partir disto a Microsoft anunciou a sua iniciativa .NET em conjunto com a linguagem de programação C#. Lê-se Cê Charpi em português. Segundo Deitel (2003, p.7), a plataforma .NET:

possibilita que os aplicativos baseados na Web possam ser distribuídos para uma grande variedade de dispositivos e para computadores de mesa. A plataforma oferece um novo modelo de desenvolvimento de software que permite que os aplicativos criados em linguagens de programação diferentes se comuniquem.

A plataforma .NET veio para flexibilizar o desenvolvimento permitindo mais facilidade ao desenvolvedor ajudando a resolver problemas como a diversidade de linguagens.

Uma grande característica da estratégia .NET é a sua independência de uma plataforma específica. É possível criar código em qualquer linguagem que seja compatível com essa tecnologia. As linguagens compatíveis são C#, Visual C++.NET, Visual Basic

.NET. Dentro desta tecnologia está o *ASP (Active Server Pages)* da Microsoft que possibilita aos programadores criar aplicativos Web.

A linguagem de programação C# foi desenvolvida especificamente para a Iniciativa .NET como uma linguagem que permite aos programadores migrarem facilmente para esta plataforma. Podemos dizer então que a Iniciativa .Net é uma plataforma em que o programador pode utilizar várias linguagens, ou seja, escolher a linguagem que se adapta. Entre estas linguagens está a linguagem C#. Isso facilita o desenvolvimento de softwares.

A linguagem C# tem raízes no C, C++ e Java e acrescenta novas capacidades em relação a estes. A linguagem de programação C# também permite um novo grau de intercâmbio entre as linguagens, ou seja, diferentes componentes de software podem interagir de modo inédito.

A arquitetura .NET estende o conceito de reutilização de software para a Internet permitindo aos programadores se concentrar em suas especialidades. A interoperabilidade de linguagens permite que programadores de C#, Visual Basic.NET e Visual C++ possam trabalhar lado a lado no mesmo projeto sem precisar aprender outra linguagem.

2.5.1 Vantagens de Utilização da Linguagem C#

Uma das principais características da linguagem C# é a sua interoperabilidade, pois programadores de outras linguagens podem interagir facilmente. Como características podemos destacar:

- Permite a sobrecarga de operadores.
- Pode implementar um modo inseguro para manipulação de ponteiros.
- As exceções não são verificadas em C#.
- Utiliza o XML para gerar documentação automática.
- Suporte a indexadores.
- Objetos não são liberados explicitamente.
- Destrutores não existem.
- Não é permitida herança múltipla.
- Ferramentas para teste de software

A linguagem de programação C# passou por cinco versões sendo a primeira no lançamento da plataforma, a segunda em fevereiro de 2007, depois com o lançamento do

visual Studio em 2010, depois com o lançamento do visual Studio em 2012 e finalmente com o lançamento do Visual Studio em 2013.

2.5.2 O Visual C# Express

A Microsoft separou um modelo de softwares denominado “*Express*” que é uma versão gratuita que difere das versões comerciais por ter algumas limitações, como sendo uma versão para fins didáticos.

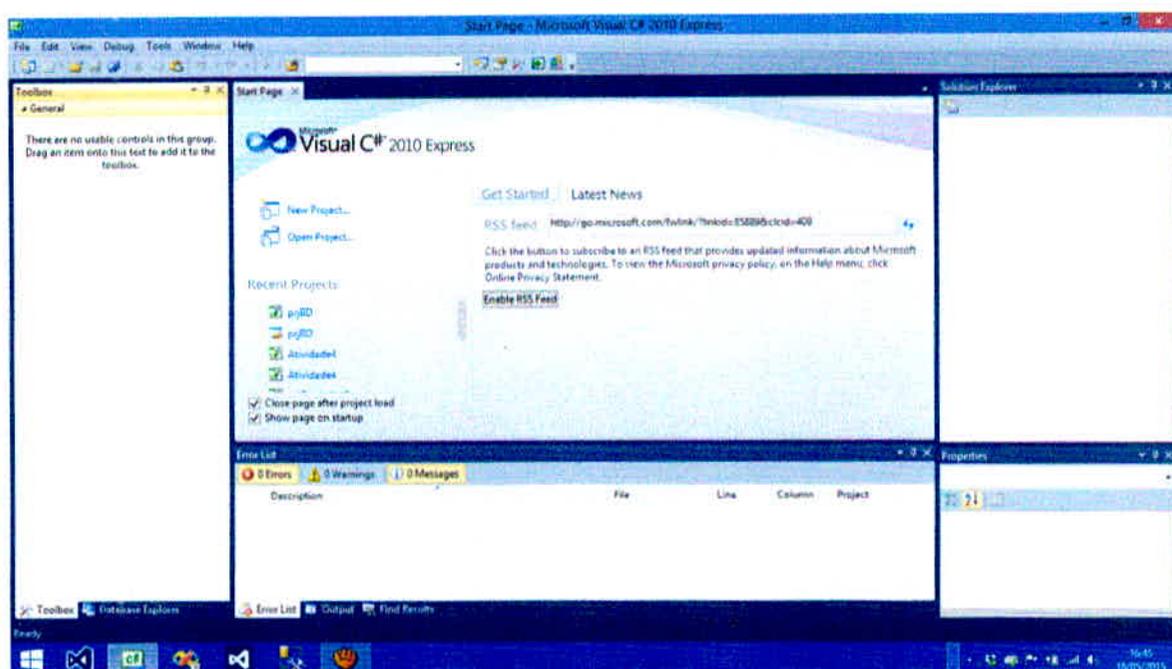
A versão *Express* pode ser baixada no site da Microsoft sem nenhuma despesa bastando apenas estar registrado. O produto do Visual Studio Express possui ferramentas que podem ser usadas para pequenos aplicativos sendo mais indicada para fins didáticos.

O Microsoft Visual C# Express pode ser baixado também em Português Brasileiro. As versões do visual Studio começam na Professional, Premium e vão até a Ultimate. Cada uma das versões oferece recursos sendo a versão Ultimate a mais sofisticada.

2.5.3 O Ambiente de programação Visual C# 2010 Express.

Pode-se ver abaixo o ambiente de programação C# com suas diversas ferramentas:

Figura 1- Pagina Inicial do visual C# 2010 Express.



Autor: Print screen da tela inicial do programa

Inicialmente temos na barra de menus os menus: *File, Edit, view, Debug, Tools, Window e Help*. Temos também a barra de ferramentas e as janelas iniciais como: *Toolbox, Start page, Solution Explorer, Properties, Error List*. Existem muitas outras janelas que podem ser vistas com o devido comando como *Database explorer* em que visualizamos os bancos de dados do sistema.

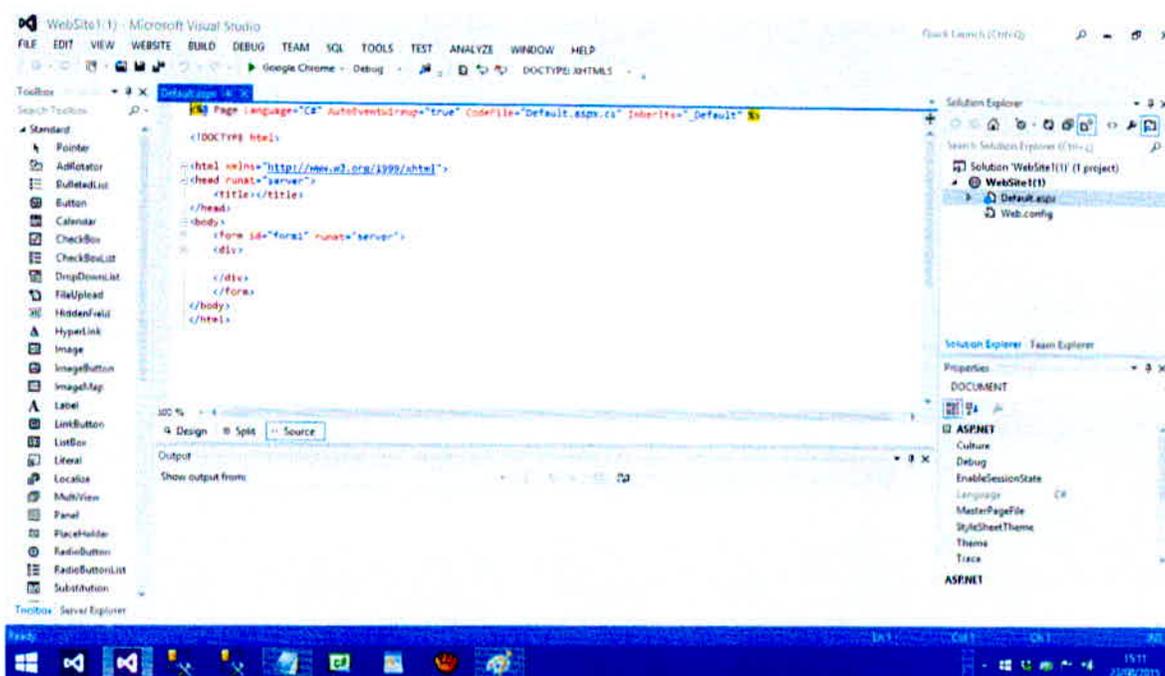
A item a seguir há uma descrição da tecnologia ASP.NET da Microsoft, tecnologia usada para construção de páginas Web.

2.6 Asp.Net

O Asp.Net é um ambiente de programação para web. Trata-se de uma tecnologia de programação para desenvolvimento de aplicativos baseado na Web, feita pela Microsoft. É o sucessor da tecnologia Asp. Os aplicativos baseados na Web criam conteúdo para clientes de navegadores da Web. Estes conteúdos Web podem incluir HTML (HiperText Markup Language).

Trata-se de uma linguagem de marcação de hipertexto, script no lado do cliente, imagens, dados. Trata-se de um framework, pois, através de seus vários componentes facilita enormemente a programação para Web. Abaixo mostra-se a tela de entrada do Asp.Net web Forms:

Figura 2- Tela Inicial do Microsoft Asp.Net- web Form- Código default.aspx.

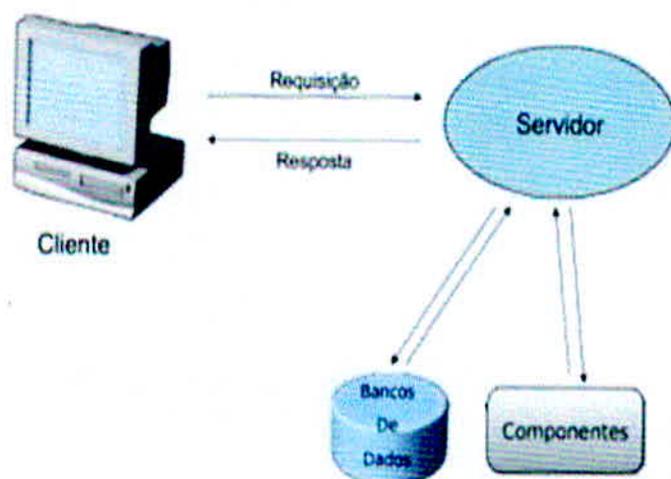


Autor: Print Screen da tela inicial do programa

Segundo Deitel (2003), a maioria dos aplicativos baseados na web são compostos por aplicativos de multicamadas e estes aplicativos dividem a funcionalidade em multicamadas separadas. Normalmente, as camadas dos aplicativos web estão implementadas em computadores diferentes.

A camada de informações armazena os dados relativos aquela aplicação e normalmente utiliza um sistema gerenciador de banco de dados. (SGBDR). É chamada de camada de dados. A camada intermediária controla a lógica do negócio, a lógica controladora e a lógica de apresentação. A lógica do negócio na camada intermediária determina as regras que fazem parte do negócio. Determina as regras daquelas transações e atualiza as informações nos bancos de dados. A camada do cliente é a interface entre o aplicativo web e o usuário do sistema. A camada do cliente mostra os dados recuperados da camada intermediária. Abaixo uma ilustração dessa arquitetura:

Figura 3-Arquitetura cliente servidor. (Figura curso Asp.Net)



Autor: Figura curso Asp.Net- Programando cursos

Em uma transação HTTP simples (Protocolo de Transferência de Hipertexto) que acontece quando de uma requisição simples na web, há uma transferência da página web através da linguagem HTML para o navegador no cliente. HTTP é um protocolo, ou seja, um conjunto de métodos e cabeçalhos que permite aos clientes e servidores trocarem informações diversas. Uma página web simples contém somente HTML.

Os documentos HTML disponíveis na web têm uma URL (*Uniform Resource Locator-Localizador de recurso Uniforme*) que mostra a localização daquele recurso. O URL, por sua vez, contém informações que direcionam para o documento na Web. Quando um URL é fornecido ao navegador, é feito um pedido HTTP ao servidor que através do DNS (Domain Name Server - Servidor de Nomes de domínio) localiza e fornece a página requisitada. DNS é um servidor que mantém um banco de dados de nomes de *host* e os correspondentes endereços IP. A transformação de Nomes DNS em IP chama-se consulta de DNS. A seguir algumas das principais vantagens da tecnologia Asp.Net.

2.6.1 Vantagens da Tecnologia Asp.Net

Podem ser consideradas as vantagens da Tecnologia Asp.Net:

- *(What you see is what you get)*. Aquilo que você vê é aquilo que você encontra.
- *Garbage Collector*. Coletor de lixo.
- Ambiente de desenvolvimento Visual Studio.
- Avançados recursos para fazer a depuração.
- Riqueza de detalhes nos controles ASP.NET, ou seja, nos *web controls*.
- Grande quantidade de classes prontas facilitando a programação e melhorando o desempenho.
- Orientada a objetos
- Pode-se usar diversas linguagens como VB.Net, C++, C#.
- Separação entre código e interface.
- Mantem automaticamente o estado das páginas.
- Segurança na navegação.

Com Asp.Net podemos fazer sites, aplicações Web, Portais, Lojas virtuais. Como exemplo de controles *Web* podemos citar *Label, Button, TextBox, RadioButtonList, AdRotator, ListBox, Calendar, Image, ImageButton* etc. A seguir uma introdução aos microcontroladores que é utilizado no Arduino.

2.7 Microcontroladores

As primeiras máquinas programáveis foram os teares que eram usados para produção de tecidos. Após os teares surgiram as máquinas de calcular mecânicas e posteriormente as eletromecânicas. Mais tarde, com o desenvolvimento da eletrônica, surgiram os primeiros computadores como o ENIAC. Ele possuía 18000 válvulas, pesava 30 toneladas e consumia 150 KW.

Com o avanço da eletrônica digital, os fabricantes de semicondutores passaram a fabricar dispositivos integrados que pudessem ser programados através de uma sequência de instruções para executar uma determinada operação.

Em 1969 a Intel lançou o primeiro microprocessador do mercado, o Intel 4004. Após o 4004 surgiu o 4040. Mais algum tempo e surgiu o primeiro microprocessador de 8 bits. Foi o 8008. O 8008 foi o precursor do 8080, uma família de microprocessadores com barramento de 16 bits, 64K de memória e 78 instruções. Com base na arquitetura do 8080 surgiram o 8085,

8086, Z80 etc. Em meados de 1980 surgiram os primeiros MCU's (Micro controller Unit) Unidade Micro controladora.

Segundo Pereira (2007), Microcontroladores são unidades mais simples que os processadores, possuem memória RAM e ROM interna, oscilador interno, I/O interno, e outros, sendo então chamados de computadores em um único Chip. Os microcontroladores facilitam tarefas como controle de periféricos como Leds, Displays LCD etc.

Segundo Souza (2007), um Microcontrolador é um pequeno componente eletrônico, dotado de uma "inteligência" programável utilizado no controle de processos lógicos. O termo pequeno se refere ao fato de em um único Chip termos a unidade de controle, a unidade lógica e aritmética, memória Ram e Rom entre outros. A seguir uma descrição do Microcontrolador Arduino Uno.

2.7.1 Microcontrolador Atmega 328

O Microcontrolador Atmega 328 P é um microcontrolador fabricado pela Atmel, tem 8 bits, 32K de memória flash, 1K de memória EEPROM e 2K de memória SRAM e arquitetura RISC.

Esse microcontrolador contém um rico conjunto de instruções e um conjunto de 32 registradores de propósito geral. Tem um desempenho de acima de 20 MIPS em um *clock* de 20 Mhz. Possui uma retenção de dados de 20 anos a 85°C e 100 anos a 25°C. Possui dois temporizadores de 8 bits com *Prescaler* separado e modo de comparação. Possui um temporizador de 16 bits com *Prescaler* separado, modo de comparação e captura. Abaixo temos o diagrama dos pinos do chip Atmega 328P:

Tabela 1- Diagrama de pinos Atmega 328P

Número do Pino	Descrição	Função
1	PC6	Reset
2	PD0	Pino Digital (RX)
3	PD1	Pino Digital (TX)
4	PD2	Pino Digital
5	PD3	Pino Digital (PWM)
6	PD4	Pino Digital
7	VCC	Voltagem Positiva (Power)
8	GND	Terra
9	XTAL 1	Cristal Oscilador
10	XTAL 2	Cristal Oscilador

11	PD5	Pino Digital (PWM)
12	PD6	Pino Digital (PWM)
13	PD7	Pino Digital
14	PB0	Pino Digital
15	PB1	Pino Digital (PWM)
16	PB2	Pino Digital (PWM)
17	PB3	Pino Digital (PWM)
18	PB4	Pino Digital
19	PB5	Pino Digital
20	AVCC	Voltagem Positiva para ADC
21	AREF	Referencia
22	GND	Terra
23	PC0	Entrada Analógica
24	PC1	Entrada Analógica
25	PC2	Entrada Analógica
26	PC3	Entrada Analógica
27	PC4	Entrada Analógica
28	PC5	Entrada Analógica

Autor: < <http://www.learningaboutelectronics.com/Articles/Atmega328-pinout.php>>

Esse chip microcontrolador é o principal circuito integrado da plataforma Arduino.

2.8 Arduino

Um microcontrolador arduino é uma plataforma de código fonte aberto de fácil uso de software e hardware e tem:

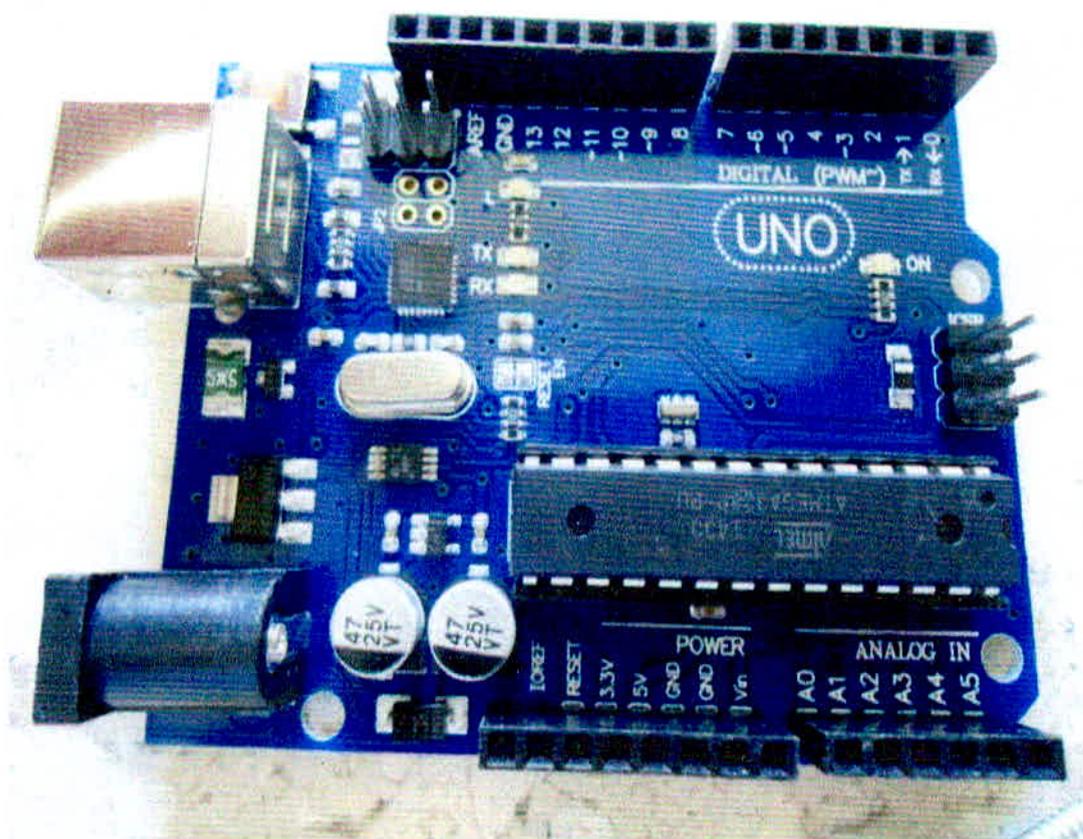
placa única e um conjunto de software para programá-lo. O hardware consiste em um projeto simples de hardware livre para o controlador, com um processador Atmel AVR e suporte embutido de entrada e saída. (MC ROBERTS, 2011,p.22)

O software consiste de uma linguagem de programação padrão e de um *bootloader* que roda na placa. Pode ser utilizado para realizar projetos como ser interligado a um computador, uma rede ou à internet. Pode, ainda, ser conectado a leds, sensores, reles de comando, sensores de distância, sensores de temperatura etc.

A mais recente placa do Arduino, a UNO difere da versão anterior, a *Duemilanove* por não ter o chip FTDI para a conversão serial. Ao contrário, ela tem um chip Atmega 8U2 programado como um conversor USB serial.

Abaixo temos o diagrama eletrônico da placa Arduino Uno:

Figura 4- Diagrama eletrônico Arduino Uno.



Autor: Foto do hardware adquirido.

Na imagem (Figura 5) vemos na parte de baixo, o circuito integrado principal, ou seja, o microcontrolador Atmega que executa as principais funções. Temos também capacitores, transistores e cristais.

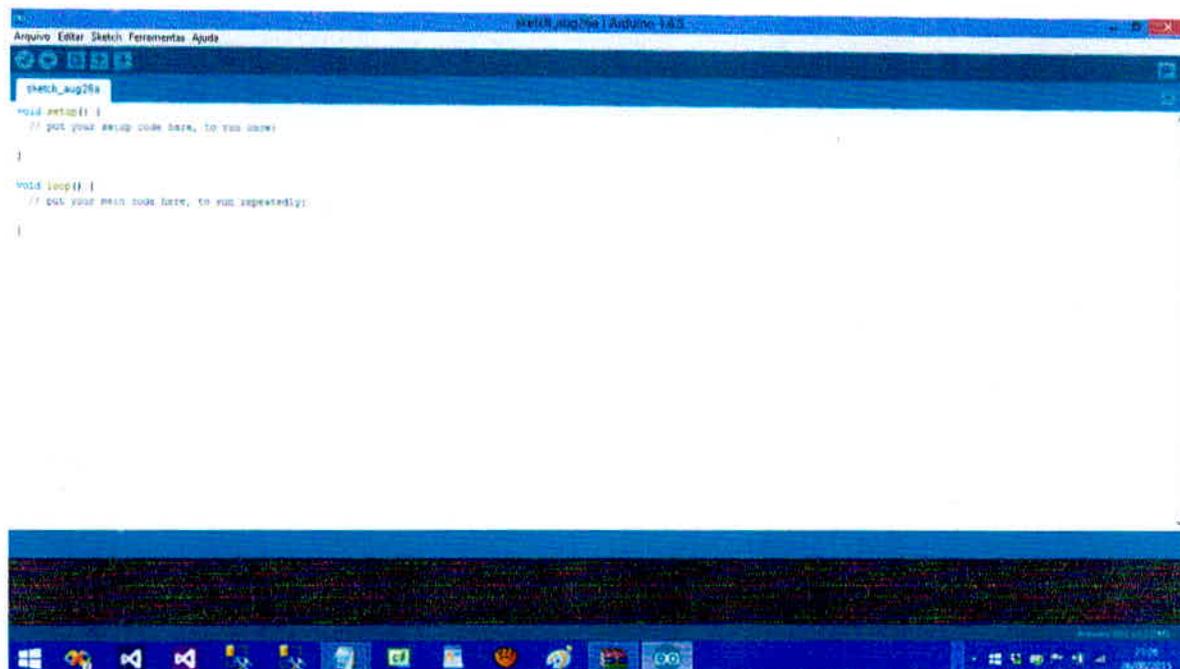
A imagem da placa nos mostra fisicamente as posições dos componentes eletrônicos suas interligações através de circuito impresso após ter sido roteada e confeccionada.

A imagem da placa nos mostra fisicamente os componentes ligados através de circuito impresso em placa de microfibras formando a placa que executa as ligações do diagrama. O roteamento é usado para posicionar as trilhas de cobre e os componentes eletrônicos soldados.

O microcontrolador Atmega 328P é carregado com o firmware Optiboot Atmega 328 que roda na memória do Microcontrolador fazendo funcionar o *Bootloader* para o funcionamento do Arduino e do *Sketch*. O *Bootloader* funciona como um sistema operacional para prover todos os recursos da placa. O *Sketch* é a IDE (Ambiente de desenvolvimento

Integrado) para desenvolvimento dos softwares que após carga executam no Arduino. Abaixo a tela do *Sketch* aberta:

Figura 6 -Tela do Sketch do Arduino Uno



Autor: *Print screen* do programa

A seguir uma descrição dos sistemas supervisórios que normalmente utilizam-se de sistemas microcontroladores.

2.9 Sistemas supervisórios

Em sistemas de supervisão é possível encontrar de sistemas simples com poucos dispositivos locais até complexos conjuntos de equipamentos, computadores e sistemas *SCADA* (*Supervisory Control and Data Acquisition*).

Segundo Viana (2008) o termo *SCADA* refere-se a um sistema de controle e aquisição de dados. No mercado de desenvolvimento existe ainda uma área pouco explorada. São os sistemas supervisórios. Os processos industriais necessitam de um controle mais apurado e isso pode ser feito com sistemas computacionais ou com sistemas *SCADA*.

A maioria dos desenvolvedores desconhece essa área de desenvolvimento. Esse tipo de necessidade é encontrado em quase todas as empresas onde se faz necessário o controle preciso de alguns processos industriais. Hoje já é possível a automação de processos industriais principalmente devido ao custo dos equipamentos de informática que diminuíram. Hoje já existem sistemas supervisórios em vários setores da indústria como hospitais, fabricas, escolas, distribuição de água, distribuição de energia elétrica etc.

O software supervisório é uma aplicação capaz de controlar e monitorar um processo em uma planta industrial trazendo como vantagem um melhor desempenho para o sistema. No software são construídas telas gráficas que auxiliam o monitoramento e dispositivos como microcontroladores que auxiliam no controle e na transmissão dos dados. As telas gráficas devem estar de acordo com o processo monitorado fornecendo informações sobre os processos.

Segundo Jurizato e Pereira (2002), existem diversos softwares supervisórios nacionais e internacionais. Com eles, vários drivers já foram construídos facilitando as interligações.

O sistema supervisório, além de poder automatizar funções em tempos exatos, permite também a supervisão de alarmes diversos como porta aberta, GMG (Grupo Motor Gerador) ligado, falta de energia elétrica, algum equipamento inoperante, temperatura alta e umidade alta ou baixa.

O controle da temperatura de ambiente alta é possível ser realizado através de um telecomando, a ligação do ar condicionado ou a sua automatização através de programas em microcontroladores.

No passado, os sistemas supervisórios eram bastante caros e a sua eletrônica complexa. Os circuitos integrados complexos utilizando milhares de portas lógicas permitiam a supervisão de equipamentos, mas eram susceptíveis a falhas e isso gerava muita manutenção. Hoje com a eletrônica avançada e o advento dos microcontroladores, tornou-se possível a supervisão de equipamentos diversos com eficiência e segurança.

Acrescentou-se ainda devido aos recursos dos microcontroladores programar funções diversas que nos ajudam devido a sua precisão. Permite, por exemplo, ajustar a entrada em funcionamento do ar condicionado na temperatura exata ou em um tempo exato. Então, conclui-se que os sistemas supervisórios são de extrema importância nos processos industriais e atualmente sem eles a indústria não estaria no padrão dos dias atuais.

3 MATERIAL E MÉTODOS

Esta pesquisa é considerada como uma pesquisa aplicada porque tem como objetivo investigar o conhecimento existente e propor uma ferramenta de *software* e *hardware* capaz de controlar e monitorar processos na indústria como o armazenamento de água para distribuição aos consumidores.

3.1 Coleta de dados

Feito levantamento técnico das necessidades do comando de bombas para o reservatório da Vila São José em Passos-MG. Neste local necessita-se do comando de bombas do *Booster* e dos alarmes de bomba ligada e temperatura. Em cada local será implementado um sistema com configurações específicas. Atualmente é feito controle manual pois, um funcionário vai ao local para executar os comandos e medidas.

3.2 Materiais

Nesta pesquisa foram utilizados materiais eletrônicos como resistores, Leds, chaves e as placas Arduino Uno e Ethernet Shield além da fiação para ligação dos componentes. Foi utilizado também um roteador para rede Wireless.

3.3 Metodologia

Segundo Mascarenhas (2012), a pesquisa aplicada é usada para estudar um problema em um certo contexto buscando soluções para problemas específicos, mas utiliza um embasamento teórico para suportá-lo.

Ainda segundo Mascarenhas (2012) sempre que iniciamos um estudo devemos verificar o que já existe sobre aquele determinado tema. Os estudos bibliográficos desta pesquisa ofereceram informações extras para complementar o trabalho.

Inicialmente foi feito um levantamento bibliográfico sobre sistemas supervisórios. Posteriormente passamos ao estudo do Arduino e aos testes de hardware entre o Arduino e a

ferramenta em Asp.Net. Feito também um estudo básico sobre o Arduino Ethernet shield e a tecnologia Asp.net.

Através de um desenvolvimento dedutivo sobre as linguagens de programação e sobre o Arduino e também experimentos no hardware do Arduino ligado ao PC (Computador Pessoal) foi implementada uma ferramenta de sistema supervisorio.

Os testes foram realizados em *hardware* adquirido sendo eles a placa do *Arduino Uno*, a placa *Arduino ethernet Shield e protoboards* (placa de experimentos) para testes de implementação.

A pesquisa bibliográfica do tipo documentação indireta, foi realizada em livros e sites na internet para entendimento da teoria envolvida.

Foi utilizado esta técnica para absorção da teoria envolvida e entendimento do código. Trata-se de conteúdo não estudado e, portanto, necessitou aprofundamento. A ferramenta é uma inovação.

Foi utilizado o motor de busca Google para encontrar conteúdo na internet para orientar a pesquisa e posteriormente foi realizado testes técnicos no hardware do *Arduino*. Utilizou-se este método como única fonte de pesquisa pois as fontes do assunto são raras.

Os dados foram analisados através de testes no *hardware* adquirido, pois isto comprova o funcionamento da ferramenta e dos equipamentos envolvidos. Foi feito estudos em sites no Google UK.

4 RESULTADOS

Nesta pesquisa pode-se verificar o funcionamento da ferramenta de programação Visual Studio Express através das tecnologias Asp.Net e também a plataforma Open Source Arduino, todas disponíveis na internet.

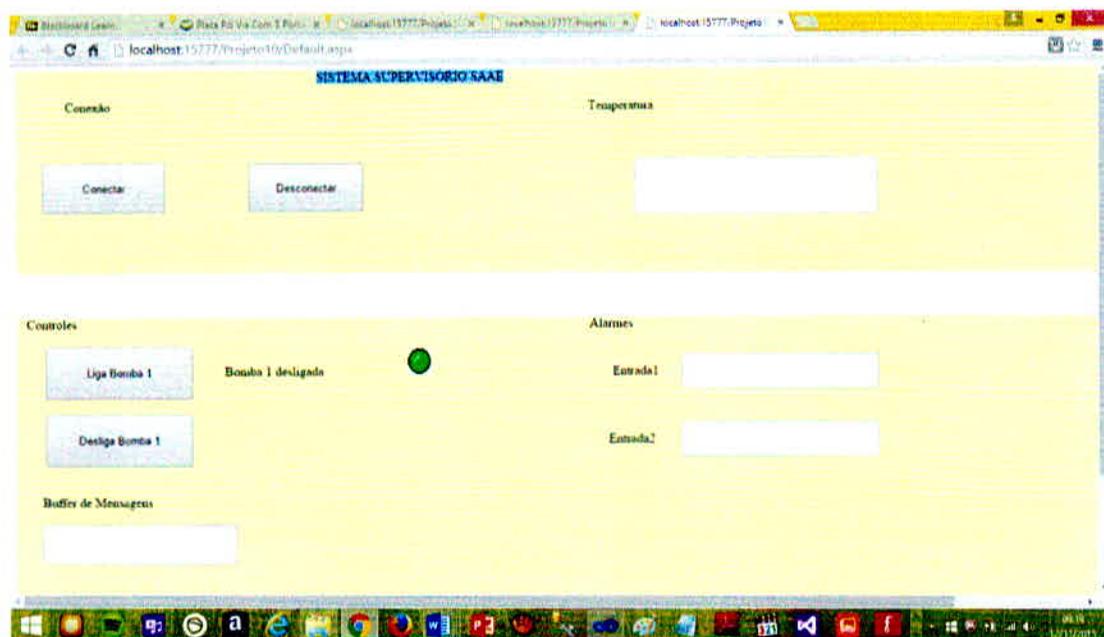
O objetivo da ferramenta é servir de sistema supervisorio que pode ser utilizado na indústria de modo geral, mais especificamente para ser utilizado em um comando de bombas em um *booster* ou alimentador de reservatório de água do Serviço Autônomo de água e esgoto de Passos.

4.1 Ferramenta Asp.Net

Na ferramenta utilizou-se conceitos de ASP.Net, Linguagem C#, TCP/IP para comunicação em rede ethernet e a linguagem C usada no Sketch do Arduino.

A ferramenta feita em linguagem Asp.Net apresentou bom resultado e é mostrada abaixo:

Figura 07- Tela inicial da ferramenta. Navegador Chrome



Autor: Print screen da tela do Navegador

A tela acima apresenta a tela do navegador que está no endereço do projeto. Isso mostra a ferramenta sendo executada no navegador. Nesta tela inicial, ainda não há conexão e por isso não há indicação de alarmes e temperatura.

Poderíamos ter utilizado a linguagem C# em um aplicativo para rede mas optou-se por um aplicativo web em Asp.Net, pois, as indústrias de modo geral tem utilizado esta tecnologia. Usa-se então um navegador para visualizar e executar comandos. A entrada de comandos utilizou-se *web Controls* do tipo Button e as saídas de dados utilizou-se os *TextBox*. Isso foi feito para melhorar o layout e a comunicação visual. Utilizou-se também um Led para sinalizar Bomba ligada (Vermelho) e bomba desligada (Verde)

4.2 O Sketch e o hardware do Arduino

O Sketch que representa a IDE do Arduino, permite gravar no Arduino o código em linguagem C. Isto é transferido através de uma porta USB. Abaixo vê-se a tela do sketch com o código que roda no Arduino:

Figura 08- Sketch do Arduino

```

Arduino_contra7 | Arduino 1.6.5
Arquivo Editar Sketch Ferramentas Ajuda
Arduino_contra7
#include <SPI.h>
#include <Ethernet.h>
#include <CS2.h>

//Configurações do Ethernet Shield
byte mac[] = { 0x2E, 0x28, 0x2E, 0x2E, 0x2E, 0x2E };
byte ip[] = { 192,168,2,20 }; // ip que o arduino assumir
byte gateway[] = { 192,168,2, 1 }; // ip do roteador
byte subnet[] = { 255, 255, 255, 0 }; //Mascara de subrede

//Definições
//Pinos digitais:
const int inputPin2 = 2; // Para uma bobina com o número do pino conectado a terra
const int inputPin3 = 3; // Para uma bobina com o número do pino conectado a terra
//Entrada analógica

EthernetServer server(80); //Inicializa a biblioteca EthernetServer com os valores de IP acima citados e configura a porta de acesso(80)

void setup()
{
  Ethernet.begin(mac, ip, gateway, subnet);
  pinMode(inputPin2);
  pinMode(inputPin3);
  digitalWrite(inputPin2, LOW); // Define o pino de terra como entrada
  digitalWrite(inputPin3, LOW); // Define o pino de terra como entrada
}

void loop()
{
  if (server.hasClient())
  {
    EthernetClient client = server.available(); // Um cliente novo se conecta ao servidor
    if (!client)
    {
      return;
    }
    // Quando o cliente se conecta ao servidor, o servidor envia um pequeno "Bom dia" ao cliente
    client.println("HTTP/1.1 200 OK");
    client.println("Content-Type: text/html");
    client.println("Content-Length: 150");
    client.println();
    // Envia o conteúdo HTML para o cliente
    client.println("<html>");
    client.println("<head>");
    client.println("<title>Controle de Bomba</title>");
    client.println("</head>");
    client.println("<body>");
    client.println("<p>Bomba ligada</p>");
    client.println("</body>");
    client.println("</html>");
  }
}

```

Autor: Print screen

A ferramenta ASP.NET que executa no navegador, comunica através de rede ethernet com o hardware do Arduino que executa comandos, recebe alarmes e dados daquele local que pode estar em qualquer lugar da rede. Abaixo foto do hardware Arduino e placa Ethernet Shield que executam o código em linguagem C:

Figura 09- Foto das Placas Arduino e Ethernet Shield



Autor: Foto do Hardware adquirido.

Nesta foto pode-se ver a placa do *Arduino* na parte de baixo e a placa *Ethernet Shield* logo acima. Há uma conexão padrão entre as placas.

A ferramenta foi feita em ASP.Net, utiliza web controls como Buttons, Labels, TextBox, Leds. A ferramenta gera o código HTML que roda no navegador. Os códigos que não são HTML, ou seja, os códigos em C# rodam no servidor Web não sendo acessível ao usuário a partir do navegador. É chamado de code behind.

Ao executar os comandos no navegador, esses comandos são transferidos ao servidor que executa o *code behind* e transfere os comandos ao *arduino* que os executa utilizando o código em linguagem de máquina hospedado naquele hardware.

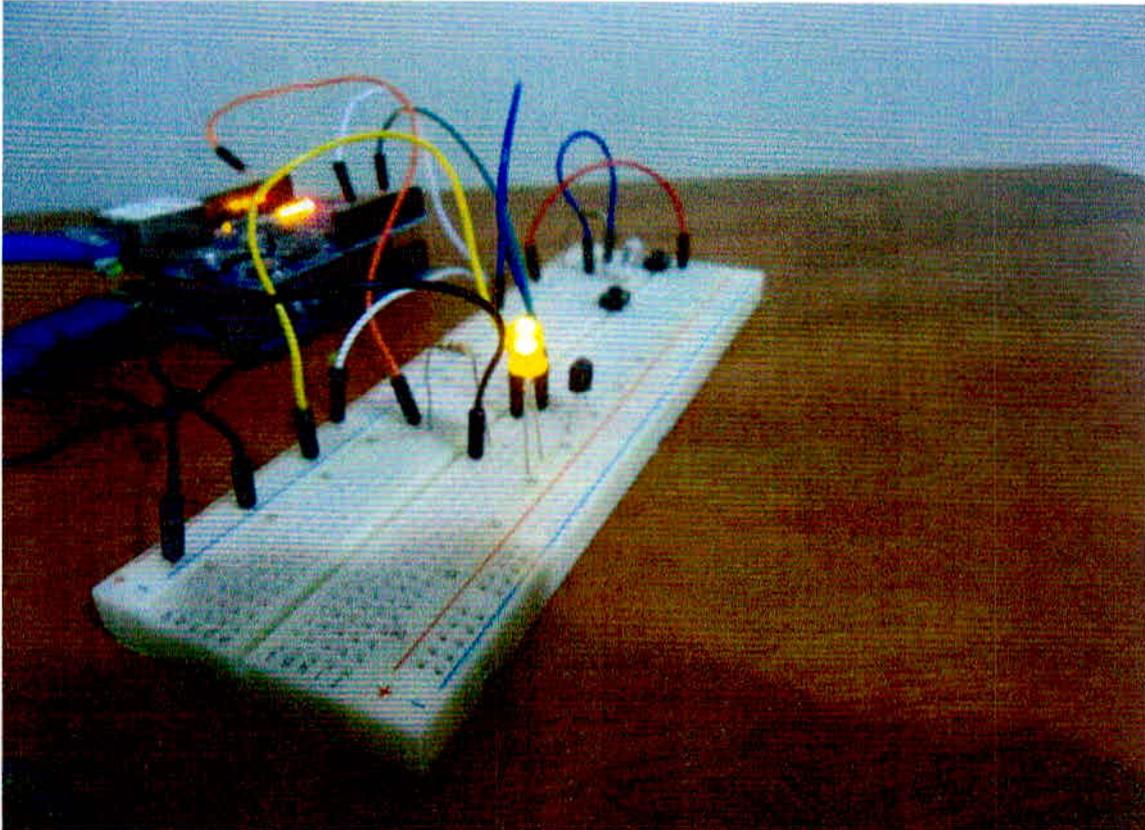
Era função deste trabalho também implementar um código de comando no Arduino para transmitir a outros Arduinos na rede, mas devido à complexidade da operação implementaremos em uma nova versão.

4.3 O Hardware em execução

Foi montado um hardware que representa a bomba juntamente com os sensores do sistema supervisorio conectado ao *Arduino*.

Abaixo pode-se ver o hardware montado executando os comandos:

Figura 10- Foto do *Arduino* sendo executado comando.



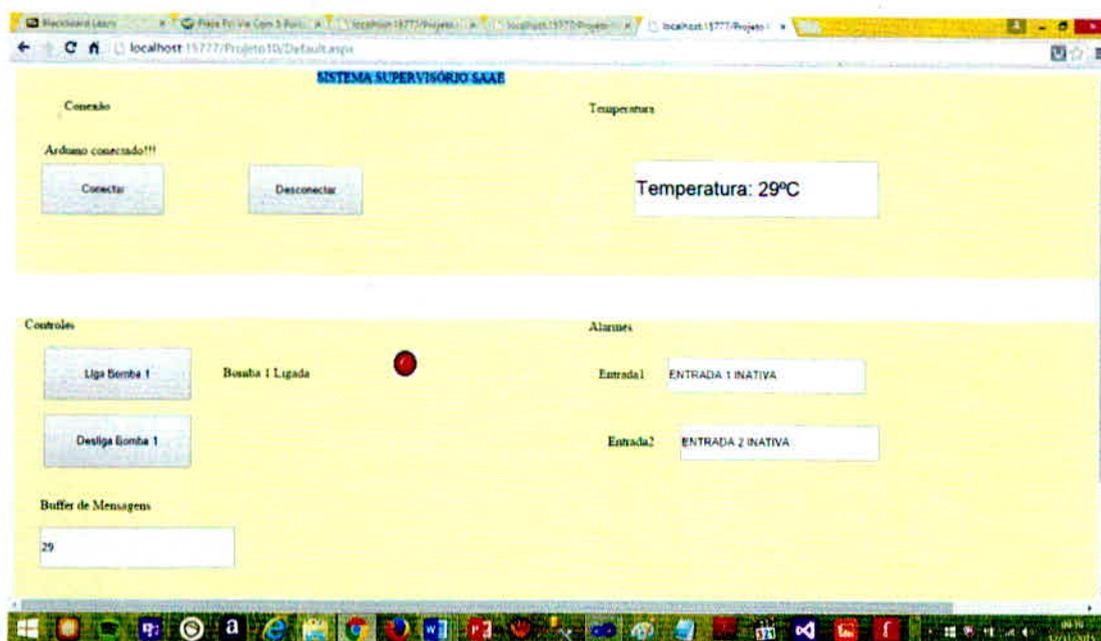
Autor: Foto do hardware.

Na figura acima, a placa do *Arduino* ao fundo, a placa *ethernet Shield* e o hardware montado que simboliza os comandos de bomba, sensor de temperatura LM35. O LM35 é um sensor de temperatura semiconductor fabricado pela *Texas Instruments* entre outras.

Acima pode-se ver também o Led aceso, que indica comando de bomba ligado, ou seja, comando executado pelo *Arduino* que recebeu comando externo.

Na tela abaixo, a tela da ferramenta executa o comando de bomba, as medições de temperatura, as entradas de alarme:

Figura 11- Tela do Navegador. Comando executado e temperatura.

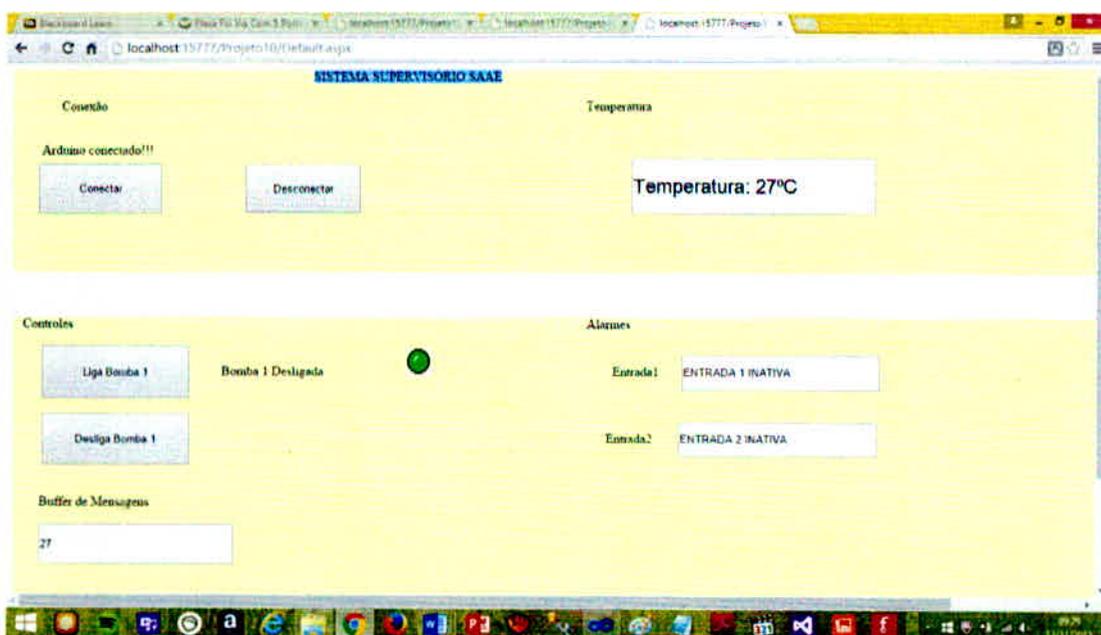


Autor: Print screen da tela do Navegador

Assim, a tela no navegador web indica conexão com o Arduino, o comando que foi executado, as entradas de alarme monitoradas conforme proposto.

Abaixo pode-se observar a tela da ferramenta em ASP.net rodando no navegador. Nesta tela o comando “Desliga Bomba” foi executado.

Figura 12- Tela do Navegador- Comando desliga e temperatura



Autor: Print screen da tela do Navegador

Viu-se através da pesquisa que os resultados foram alcançados, porém com limitações, pois os comandos foram executados a distância, via rede ethernet, mas pretendia-se também executar comandos em outros módulos *arduinios* na rede.

Porém, isso torna o projeto mais complexo e será feito em uma nova etapa. Será preciso uma pesquisa mais aprofundada, pois é necessário verificar os aspectos de segurança de rede e da própria ferramenta.

5 CONCLUSÃO

Através desta pesquisa podemos verificar um sistema supervisorio em execução. Os sistemas supervisorios são de grande importância na indústria de um modo geral. Pode-se concluir que são importantes ferramentas para utilizar-se nos processos da indústria tornando os processos mais confiáveis, com maior precisão e facilitando tarefas feitas a distância.

A comunicação a distância através da rede permite melhorar os processos, os controles tornando o sistema mais confiável e menos susceptível a falhas. Utilizou-se Asp.Net e linguagem C# para codificar a ferramenta com bons resultados para comunicação com o Arduino.

A programação do arduino é simples, utiliza-se linguagem C, é uma plataforma aberta o que permite uma grande variedade de projetos e neste caso, um sistema supervisorio. Para verificar os limites de uso do Arduino é necessário um conhecimento mais profundo e o site nos fornece muita informação.

O arduino é um hardware eficiente, mas básico, com limitação no número de portas e velocidade de operação. Então, para construção de um sistema com alta capacidade, não se recomenda o uso do Arduino. Mas a grande vantagem de projetos com o Arduino é seu baixo custo e facilidade de implementação já que a plataforma é livre.

A linguagem C# tem suporte ao TCP/IP permitindo comunicação na rede com segurança necessária. Isso foi comprovado na pesquisa, mas verificou-se também a grande dificuldade de referências sobre o TCP/IP, pois ainda falta conhecimento profundo sobre esse protocolo.

Então, para expandir a pesquisa precisa-se de mais conhecimento sobre a linguagem C#, TCP/IP e melhorar o hardware, pois o sistema apresentou falhas devido aos problemas eletrônicos como baixa tensão da porta USB do computador. Isso acarreta falha na precisão das medidas. Em uma futura versão outras funcionalidades estão previstas como a comunicação entre os Arduinos para que um Arduino possa comandar um equipamento em um outro Arduino na rede como por exemplo uma bomba de água.

REFERÊNCIAS BIBLIOGRÁFICAS

DEITEL, H.M at al. **C # Como Programar**. Primeira Edição. São Paulo. Makron Books. 2003. 1152p.

FOROUSAN, Behrouz A. **Comunicação de Dados e Redes de Computadores**. Quarta Edição. São Paulo. Bookman. 2008. 1134p.

FUENTES.Ivan.**Diagrama Arduino Uno**.02/03/2013. Disponibilidade < <http://blog.goshield.es/2013/03/como-instalar-en-windows-8-los-drivers.html>> Acesso em 30/08/2015.

LEARNINGAboutelectronics.**Pinologia Atmega 328**. Disponibilidade < <http://www.learningaboutelectronics.com/Articles/Atmega328-pinout.php>> Acesso em 26/10/2015- Revista eletrônica

MASCARENHAS, Sidnei Augusto. **Metodologia científica**. São Paulo. Pearson Education. 2012

MASSIMO,DAVID,TOM. **Introdução ao Arduino**. Disponibilidade <<https://www.arduino.cc/en/Guide/Introduction>> Acesso em: 27 agosto 2015.

MCROBERTS, Michael. **Arduino Básico**. Primeira Edição. São Paulo. Novatec. 2011.215p.

PEREIRA, Fabio. **Microcontroladores PIC**. Sexta Edição. São Paulo. Érica. 2007. 366p.

REZENDE, Denis Alcides. **Engenharia de Software e Sistemas de Informação**: São Paulo, Brasport , 2005

SILBERSCHATZ, Abraham.GALVIN ,Peter.GAGNE,Greg. **Sistemas Operacionais**. Primeira Edição. Rio de Janeiro. Campus. 2000. 585p.

SOUZA, David . **Desbravando o PIC**. 12º Edição. São Paulo, Erica,2008,268p.