

UNIVERSIDADE SÃO FRANCISCO
Engenharia Elétrica

ITALO ITO CUNHA

**CONTROLADOR PARAMETRIZÁVEL VIA REDE
ETHERNET**

Itatiba
2012

Italo Ito Cunha – R.A. 002200800397

Controlador Parametrizável Via Rede Ethernet

Monografia apresentada ao Curso de Engenharia Elétrica da Universidade São Francisco, como requisito parcial para obtenção do título de Bacharel em Engenharia Elétrica.

Orientador: Prof.º André Renato Bakalereskis

Itatiba
2012

Italo Ito Cunha – R.A. 002200800397

Controlador Parametrizável Via Rede Ethernet

Monografia aprovada pelo programa de graduação em Engenharia Elétrica da Universidade São Francisco, como requisito parcial para obtenção do título de Engenheiro Eletricista.

Área de concentração:

Data da aprovação: __/__/__

Banca Examinadora:

Prof. André Renato Bakalereskis (Orientado)
Universidade São Francisco

Prof. João Alex Franciscan Vaz
Universidade São Francisco

Engenheiro de Aplicações: Daniel Brandão Nichele
Dynamic Air LTDA.

Dedico este trabalho a Deus por tudo que tem me dado; a minha família, em especial aos meus pais; e a minha noiva, fonte de inspiração e alegria para as horas mais difíceis.

AGRADECIMENTOS

Agradeço primeiramente a Deus por ter permitido que tudo isso fosse possível, aos meus pais e a minha família por sempre estarem ao meu lado em todos os momentos. A minha noiva, por toda paciência, compreensão e por sempre acreditar em mim.

Agradeço aos grandes amigos que tive a felicidade de conhecer durante o curso de graduação. Apoiamo-nos em muitos momentos importantes e formamos uma grande equipe ao longo dos anos.

A Universidade São Francisco e aos queridos professores, aos quais guardo boas recordações e que me capacitaram para a realização deste trabalho.

*“Se você pensa que pode ou sonha
que pode, comece. Ousadia tem
genialidade poder e mágica. Ouse
fazer e o poder lhe será dado.”*

(Johann Wolfgang Von Goethe)

RESUMO

O presente trabalho busca apresentar e aplicar parte dos conhecimentos adquiridos durante o curso de graduação, através de um conceito para um dispositivo caracterizado como um controlador parametrizável via rede ethernet. Dada a crescente necessidade de haver dispositivos mais inteligentes, flexíveis e de baixo custo, este dispositivo visa propor conceitos para aplicações simples de automação residencial e industrial. Baseado em uma prova de conceito, que utiliza um sistema microcontrolado com interface Ethernet, onde o equipamento poderá ser parametrizado e configurado de forma intuitiva, logrando atender as necessidades de um mercado cada vez mais competitivo. Este trabalho também tratará de conceitos de automação, e de como foi sua evolução pela história até se tornar uma das áreas de maior importância atualmente na indústria. Falará também sobre a estrutura que constitui um CLP, teoria de redes de computadores, utilização de microcontroladores em aplicações com TCP-IP e linguagem de programação em HTML para elaboração de páginas Web. Possui objetivo final de unir todos estes conhecimentos para gerar uma solução eficiente em aplicações simples de automação.

Palavras chave: CLP, microcontrolador.

ABSTRACT

This work presents part of the knowledge obtained during the graduation course, through a concept of new device characterized by configurable controller by Ethernet network interface. There is an increasing need each day for new flexible, low-costs and more intelligent devices, this equipment presents some concepts for simple application of residential and industrial automation. Based in a concept with microcontrolled system with Ethernet interface, the device can be easily configured, setting parameters, achieving attend the needs of a very increasingly competitive market. This work will deal with some concepts of automation, and its history of evolution to become one of the most important area in the industry, PLC structure, computer networks, TCP-IP applications with microcontrollers, HTML program language for Web pages, in order to propose an efficient low-cost solution for simple automations applications.

LISTA DE FIGURAS

Figura 1 – Exemplo de uma rede de computadores.....	17
Figura 2 – Exemplo de um diagrama de controle em malha fechada.....	18
Figura 3 – Diagrama de blocos de um sistema de Automação	18
Figura 4 – Estrutura de um PLC	21
Figura 5 – Exemplo de Módulos de entrada e saída	23
Figura 6 – Exemplo de uma base ou Rack	25
Figura 7 – Exemplo de uma rede de computadores.....	26
Figura 8 – Exemplo de computadores compartilhando periféricos	27
Figura 9 – Exemplo de que não é possível transmitir dados se o barramento está ocupado.....	28
Figura 10 – As sete camadas do modelo OSI.....	29
Figura 11 – Comparação do Modelo OSI com o protocolo TCP/IP	31
Figura 12 – Comparação camadas físicas do Modelo OSI com o padrão Ethernet....	34
Figura 13 – Arquitetura interna Microcontrolador LM3S6965	38
Figura 14 – Interação entre aplicação CGI, Servidor e Usuário	41
Figura 15 – Layout Placa de Desenvolvimento Microcontrolador LM3S6965.....	46
Figura 16 – Arquitetura Placa de Desenvolvimento Microcontrolador LM3S6965	47
Figura 17 – Tela do software IAR Embedded Workbench.....	48
Figura 18 – Interface de operação e parametrização.....	49
Figura 19 – Tela do software IAR Embedded Workbench.....	50
Figura 20 – Placa de expansão e entradas e saídas digitais.....	51
Figura 21 – Circuito de entrada.....	52
Figura 22 – Circuito de saída	52

LISTA DE ABREVEATURAS E SIGLAS

CLP – Controlador Lógico Programável

ISO – *International Standards Organization*

OSI – *Open Systems Interconnections*

CPU – Unidade Central de Processamento

TCP – *Transmission Control Protocol*

IP – *Internet Protocol*

HTML – *Hyper Text Markup Language*

I/O – *Input/Output*

PWM – *Pulse Wide Modulation*

LED – *Light Emitting Diode*

SUMÁRIO

1	INTRODUÇÃO	13
2	REVISÃO BIBLIOGRÁFICA	14
2.1	INTRODUÇÃO A AUTOMAÇÃO	14
2.1.1	Breve histórico	14
2.1.2	Industrialização no Brasil	16
2.1.3	Automação Industrial	16
2.2	CONTROLADOR LÓGICO PROGRAMÁVEL.....	20
2.2.1	Arquitetura do PLC	20
2.2.2	CPU - Unidade Central de Processamento	22
2.2.3	Interfaces de Entrada e Saídas.....	22
2.2.4	Fonte de Alimentação	24
2.2.5	Base ou Rack	24
2.3	REDE ETHERNET TCP/IP.....	25
2.3.1	Introdução a Redes.....	25
2.3.2	Protocolos.....	27
2.3.3	Modelo OSI.....	29
2.3.4	Modelo TCP IP	31
2.3.4.1	Camada de Aplicação	32
2.3.4.2	Camada de Transporte.....	32
2.3.4.3	Camada de Internet.....	33
2.3.4.4	Camada de Interface com a Rede	33
2.3.5	Ethernet	33
2.3.5	Placas de Rede	35
2.4	MICROCONTROLADORES.....	35
2.4.1	Arquitetura Harvard.....	35
2.4.2	Processador com núcleo ARM Cortex-M3	36

2.4.3	Microcontrolador Stellaris LM3S6965.....	36
2.4.4	Controlador Ethernet.....	38
2.5	LINGUAGEM HTML	39
2.5.1	Formulários.....	40
2.5.2	CGI (Common Gateway Interface).....	41
2.5.2.1	Método GET	42
2.5.2.2	Linguagem C para CGI.....	42
2.5.3	SSI (Server Side Includes).....	43
3	METODOLOGIA.....	44
3.1	PLACA DE DESENVOLVIMENTO STELLARIS LM3S6965 TI	45
3.2	PÁGINA WEB.....	47
3.3	INTERFACE DE PROGRAMAÇÃO, SIMULAÇÃO E GRAVAÇÃO.....	50
3.4	HARDWARE DE EXPANSÃO DE ENTRADAS E SAÍDAS.....	51
4	RESULTADOS.....	53
5	CONCLUSÃO	54
5.1	DIFICULDADES ENCONTRADAS	54
5.2	PROJETOS FUTUROS.....	54
6	REFERÊNCIAS.....	56

1. INTRODUÇÃO

Tão antiga quanto às próprias necessidades físicas dos seres humanos, a necessidade de controlar sempre esteve presente na vida cotidiana, até mesmo nas épocas mais remotas. Mesmo sem muito avanço tecnológico, o homem era capaz de desenvolver artefatos e dispositivos de qualidade muito eficientes para atender suas mais variadas necessidades.

O ser humano sempre esteve atento ao seu meio na busca de facilitar seu trabalho, encontrando formas, quase sempre através de mecanismos, de torná-lo mais simples ou para diminuir o esforço físico. A cada invenção, seu próprio estilo de vida era mais influenciado, o que foi alterando toda uma sociedade. Dessa forma, cada vez mais o homem percebeu que através de novos equipamentos ou inventos, seria capaz de alcançar o caminho mais rápido para atingir seus objetivos. Neste cenário, torna-se cada vez mais evidente a necessidade da criação de novos dispositivos de baixo custo e de fácil acesso como o conceito do controlador proposto neste trabalho, que pretende facilitar a automatização em aplicações mais simples.

Neste trabalho de conclusão de curso, serão abordados de forma resumida alguns conceitos de automação, microcontroladores, protocolos de redes e controladores lógicos programáveis, com o objetivo de propor alguns conceitos de um dispositivo flexível de baixo custo, destinado a aplicações de automação mais simples industriais ou residenciais. Sua flexibilidade consiste neste dispositivo ser parametrizável através de uma interface Web, acessada pelos navegadores mais comuns, sem a necessidade de uma ferramenta ou software especial. Através desta interface, será possível configurar e parametrizar os sinais de entradas e saídas que serão acionadas pelo controlador proposto neste trabalho, facilitando a implementação deste dispositivo em automatizações simples.

Este trabalho está dividido em duas partes principais, a primeira se constitui basicamente da pesquisa que trata de uma introdução teórica dos principais tópicos e conhecimentos que compõem o desenvolvimento do trabalho. Já a segunda parte, trata da elaboração de um protótipo para aplicação e demonstração do conceito proposto neste trabalho acadêmico.

2. REVISÃO BIBLIOGRÁFICA

2.1. INTRODUÇÃO A AUTOMAÇÃO

Um breve histórico deve ser analisado, com ênfase em seus principais pontos de transformações tecnológicas e econômicas, para entender os processos controlados automaticamente. Desde a antiguidade, estes dispositivos sofreram grandes evoluções até chegarmos ao conceito que hoje conhecemos como “**Automação**”.

Em escala tecnológica, onde se possui qualquer tipo de processo de fabricação ou desenvolvimento simples ou complexo, a existência ou qualquer forma de controle se torna decisiva. O estudo para o mesmo se torna sempre válido, pois assim se tornam visíveis os motivos e as necessidades de sua implantação, definindo-se assim a importância real dos sistemas de controle discretos.

As transformações ocorridas na sociedade foram tão marcantes que influenciaram o estilo de vida da sociedade atual. A cada inovação implantada pelos inúmeros dispositivos e máquinas, maior era o nível de desenvolvimento profissional que evoluía conforme o homem descobria novas funções em uma nova realidade, que possuía e gerava diferentes necessidades e desafios, até então jamais vistos antes. [1]

2.1.1. Breve histórico

A primeira forma de organização da sociedade como conhecemos hoje foram os nômades, que se reuniam em determinado local, exploravam todos os recursos e quando não tinham mais como retirar sua subsistência partiam para outro território. Somente com o desenvolvimento das técnicas agrícolas, os indivíduos passaram a se fixar nas terras que antes eram exploradas e abandonadas. A partir dessa nova forma de organização os territórios começaram a ser definidos e organizados através de feudos, reinos e posteriormente países. Como consequência da definição dos territórios, foi preciso desenvolver técnicas e equipamentos que possibilitassem seu desenvolvimento. Os primeiros equipamentos foram os utensílios agrícolas e posteriormente os destinados a tecelagem e manufaturas.

A maior transformação das sociedades onde se predominava a agricultura foi a Revolução Industrial, que estabeleceu novos desafios e um estilo de vida voltado para o desenvolvimento econômico e o avanço tecnológico. A Revolução Industrial ocorreu no século XVIII e teve origem na Inglaterra, dando assim início a era industrial, que foi caracterizada pela introdução de máquinas para substituir o trabalho braçal, no qual era necessária muita força ou em trabalhos muito repetitivos.

Antes do período de intensa industrialização, o padrão de vida na Inglaterra era muito baixo, a população era dividida em feudos e a principal atividade era a agricultura ou a manufatura de lã para a fabricação de tecidos. Entretanto, este processo era totalmente manual, os tecelões ganhavam muito pouco e trabalhavam praticamente para uma única família durante toda a vida e não possuíam remuneração adequada. Com a industrialização destes processos através de máquinas, os feudos foram crescendo se e tornando vilas, cidades e finalmente chegando aos grandes centros urbanos e metrópoles como conhecemos hoje.

As máquinas foram modernizadas desde a idade média pelos tecelões, principalmente na chamada era industrial, onde os equipamentos eram capazes de realizar o trabalho de 100 homens e poderiam ser operadas até por uma criança. Neste período, o conceito de produção seriada começou a ser implantado, os tecelões passaram a integrar várias tecelagens em um único local, com muitas máquinas de tecer a lã. A partir dessa integração, foram criadas as primeiras indústrias no mundo, mais precisamente por volta de 1811 à 1812. Nesse período foram estabelecidas as cargas horárias de trabalho e o local de trabalho foi separado de onde viviam os trabalhadores.

A invenção da máquina a vapor pelo escocês James Watt, que tinha como base um motor a vapor criado por um ferreiro chamado Thomas Newcomen para bombear água de minas de carvão, foi o marco principal de toda a transformação social desencadeada nestes anos de desenvolvimento tecnológico. Através das máquinas à vapor, aliadas ao aprimoramento das técnicas de beneficiamento dos metais, foi possível que vários anos mais tarde fossem criados os motores a combustão e a aplicação da energia elétrica. Assim, a Revolução Industrial estava consolidada e a Inglaterra era o centro industrial do mundo. Nesta época, houve um aumento demográfico das cidades onde estavam instaladas as indústrias. Porém, apesar de não ser a proposta da industrialização, durante muitos anos, todo esse momento de transição foi sustentado pela exploração de mão de obra, com a maior parte da população na miséria.

Além da exploração da mão de obra, as indústrias passaram a utilizar mão de obra infantil e feminina, pois o trabalho na indústria não era tão pesado, possibilitando o emprego deste tipo de mão de obra. Com isso, houve mobilizações e revoltas da população, que resultaram nas conquistas dos direitos do trabalho, onde muitas destas foram conseguidas depois de muita violência e mortes de trabalhadores.

Mais tarde, no século XIX, foram criadas as primeiras organizações sindicais. Entretanto, antes mesmo destas organizações, através de mobilizações e reformas governamentais ao longo de todo o processo de mecanização, o proletariado já havia conquistado muitos direitos, como a redução de carga horária e a proibição de empregar menores de idade. Neste cenário de produção em série, destacou-se em 1909 Henry Ford,

que foi chamado do pai da indústria automobilística nos Estados Unidos. Ele desenvolveu um método revolucionário de linhas de montagens de automóveis, que já neste mesmo ano era capaz de produzir 2000 automóveis por mês.

Durante todos estes anos, a indústria se modernizou logrando um alto nível de controle. De uma forma geral, em todas as etapas dos processos de fabricação, atendendo as mais altas especificações de qualidade e sustentabilidade. [1]

2.1.2. Industrialização no Brasil

O desenvolvimento industrial durante o período colonial era praticamente nulo, e não passava de alguns processos básicos, como a produção de açúcar, tecidos e fundição de ferro.

Com a chegada D. João VI no século XVIII, houve algumas tentativas para o desenvolvimento industrial, a princípio na área de siderúrgica, que não veio a vingar por falta de mercado. A crise inflacionária de 1850 atingiu diretamente a indústria têxtil, que havia se estabelecido ao norte do país desde 1844. O Brasil continuava tendo a agricultura como base de sua economia, sendo um exportador de matéria prima, e apesar disso, continuava sem muito desenvolvimento em seus processos de cultivo. [1]

Já em 1880, a crise da lavoura ocasionou a queda dos preços e a diminuição das safras. Este cenário veio a se agravar em 1882, quando começaram as primeiras mobilizações a favor da industrialização do país. Outro fator agravante foi que cada vez mais o processo produtivo exigia uma maior qualificação dos trabalhadores, o que não existia no país. No final do reinado do Imperador D. Pedro II, em 1872, havia um índice de analfabetismo de 70% no país que tinha uma população de 10 milhões de habitantes.

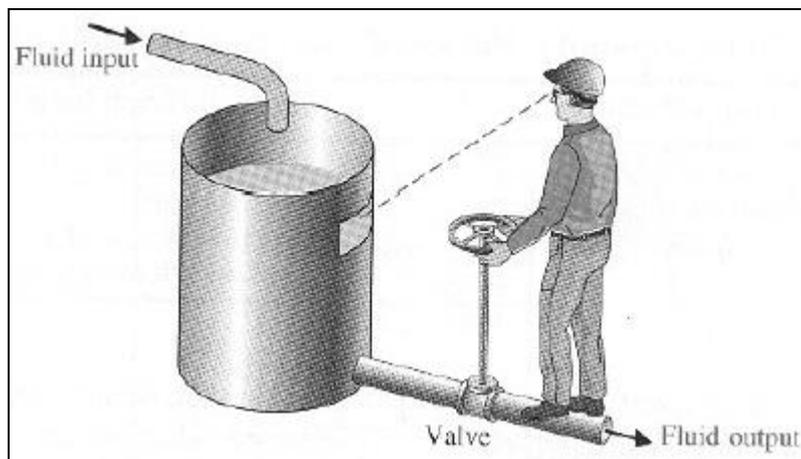
Somente muitos anos mais tarde, no governo de Juscelino Kubitschek (1955-60), foi que a indústria no Brasil se consolidou de verdade, através de medidas políticas protecionistas que controlavam as importações e estimulavam as exportações. Estas medidas ajudaram a estabilizar a economia, o que permitiu finalmente que o Brasil criasse e atraísse novas indústrias para o país. [1]

2.1.3. Automação Industrial

A automação industrial tem como objetivo gerar soluções para os mais diversos desafios presentes na indústria, buscando melhorar a qualidade de vida através de várias técnicas de controle, que aumentam a produtividade e a qualidade do produto. O conceito de automação trata da aplicação de inúmeras técnicas desenvolvidas para os mais diversos

sistemas e setores da indústria, tornando-os capazes de operarem com eficiência em processos que envolvem movimentos automáticos ou repetitivos.

Diferentemente do sistema conhecido como malha aberta, onde o processo sempre terá a mesma resposta que possui uma saída independente da sua entrada, e o sistema não realiza a autocorreção de processo conforme a figura abaixo. Na imagem podemos compreender que o sistema não possui controle automático, sendo sempre necessária a intervenção de um operador, por exemplo, para corrigir qualquer variação. Já os sistemas automatizados são capazes de reagir ao seu meio de atuação. [1]

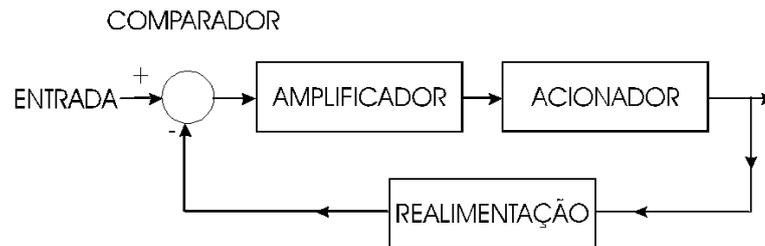


Fonte: < <http://www.ebah.com.br/content/ABAAABM7MAE/aula-02-controle-malha-aberta-fechada>>

Figura 01: Exemplo de uma rede de computadores.

Os sistemas em malha aberta possuem a vantagem de serem mais simples e mais baratos em sua construção, sendo conveniente quando o custo para outros sistema se torna muito elevado.

Os sistemas controlados estão sempre sendo realimentados com novas informações e assim recalculando as variáveis de processo, corrigindo seu trabalho final, dessa forma sua saída está totalmente relacionada à sua entrada. Através dessa relação, o sistema corrige automaticamente sua saída, caso esteja fora dos padrões. A esses tipos de sistema chamamos de sistemas em malha fechada.

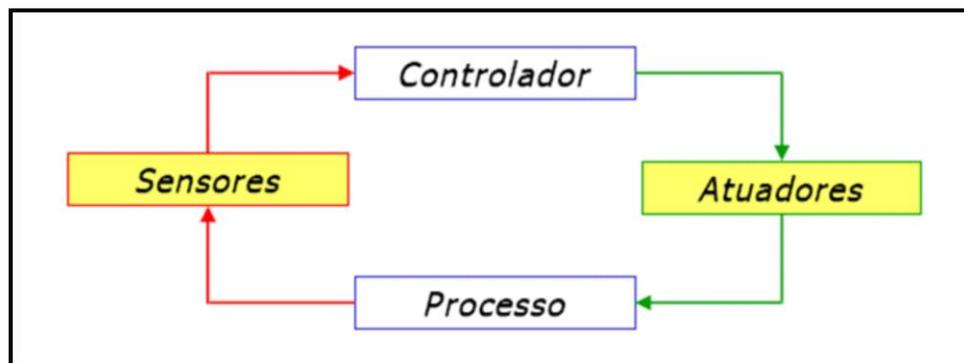


Fonte: < <http://www.dee.feb.unesp.br/~marcelo/robotica/Robot4.htm> >

Figura 02: Exemplo de um diagrama de controle em malha fechada.

A automatização está associada normalmente a algum sistema de malha de controle, onde a saída depende da entrada através de um sistema de realimentação capaz de autocorrigir o sistema caso haja variações no processo. Sendo denominado assim um sistema de controle inteligente.

Nos mais diversos sistemas, independentemente de suas características, todos deverão seguir algumas regras básicas de teoria de controle de sistemas. No caso de um controle de malha fechada, sua principal característica consiste em um sistema de realimentação de informações, para controle no fim do processo. É comum que haja a existência de três componentes básicos: sensores, controladores e atuadores, conforme ilustrados na figura abaixo.



Fonte: <<http://clpredes.wordpress.com/category/tecnologia/>>

Figura 03: Diagrama de blocos de um sistema de Automação.

No sistema de automação representado na figura 03, os sensores são dispositivos capazes de medir fenômenos físicos como luz, pressão, vazão, cores, dentre outros. Estes sensores enviam sinais contínuos, e podem ser simples sinais de abrir e fechar contatos a

medições de grandezas físicas, convertidas em grandezas elétricas como corrente ou voltagem. Os transdutores são exemplos também conhecidos como conversores de sinais, capazes de responder a estímulos físicos através de sinais elétricos, capazes de serem analisados e interpretados.

Já os atuadores, são responsáveis por realizarem a força regulação ou ação física designada pelo controlador, a fim de executar a correção do sistema. Um exemplo seria o controle de velocidade de motores, onde o controlador envia um sinal elétrico para o atuador, esses sinais podem ser magnéticos, pneumáticos, ou até mesmo mistos.

No caso dos controladores, é necessário realizar um modelamento matemático do processo da planta, a fim de conhecê-la integralmente. Dessa forma, pode-se dimensionar um controlador adequado. Neste trabalho, será abordado o tipo de controle discreto, por isso o controlador abordado será o controlador lógico programável.

“O controle do tipo discreto, voltado aos processos digitais teve seu início marcado pela utilização de dispositivos eletromecânicos do tipo a relés. Contactores, temporizadores e dispositivos de proteção constituem a base de projetos de intertravamentos, elaborados em diagrama a relés, capazes de efetuar o controle discreto. Chaves e contatos simulam os níveis lógicos baseados na lógica binária e promovem um controle utilizado na indústria até os dias de hoje.” [1].

A utilização de relés eletromecânicos foi a única opção possível até a década de 1960, porém com o desenvolvimento da eletrônica e de circuitos integrados, surgiram os primeiros controladores lógicos-programáveis. A princípio, os mesmos se limitavam apenas a controlar sinais discretos digitais, que substituíam perfeitamente os antigos relés. Já os controles analógicos, se desenvolveram através do aperfeiçoamento dos amplificadores operacionais. Os controladores se desenvolviam à medida que a microeletrônica evoluía e passava a se utilizar de circuitos eletrônicos, cada vez mais complexos e capazes de efetuar muitos recursos poderosos de controle, como os utilizados atualmente, por exemplo, o controle Proporcional-Integral-Derivativo (PID).

Dentre as vantagens de se automatizar, é relevante ressaltar, que esta transformação trata-se de uma tendência da evolução tecnológica necessária para ser competitivo e se manter no mercado atual. Além de buscar a valorização do ser humano, em um novo contexto de trabalho, no qual o trabalhador não é mais apenas um mero executor de tarefas repetitivas, mas um sujeito dotado de conhecimento em uma nova relação de produção, criado pelo avanço tecnológico e técnico do mercado de trabalho. [1]

2.2. CONTROLADOR LÓGICO PROGRAMÁVEL

O PLC (*Programmable Logic Controller*) ou Controlador lógico programável pode ser considerado um computador de porte industrial, dedicado a executar várias tarefas de controle, como sequências lógicas, operações aritméticas, temporização ou contagem, e, além disso, pode ser conectado a vários tipos de redes. [2]

Ainda nos anos 50, os relés eram largamente utilizados na indústria para controle de acionamentos e intertravamentos nas linhas de produção, porém, apesar de funcionarem bem, apresentavam muitos inconvenientes. Como, por exemplo, na utilização de controle complexo. Nessa situação, era necessário utilizar centenas de relés, que normalmente possuem dimensões físicas elevadas, sendo necessário armazená-los em grandes painéis ou cabines, onde ficariam protegidos das adversidades do ambiente.

Outro grande inconveniente era o fato de a manutenção se tornar mais complicada, pois para que fosse realizada qualquer alteração era preciso realizar interrupções no processo produtivo, o que assim como nos dias atuais causava muitos transtornos. [1]

Já no fim dos anos 60, a empresa General Motors Corporation iniciava os primeiros testes com controle de lógica através de softwares. Estes testes se utilizavam de um computador e periféricos, que realizavam acionamentos discretos. Estes testes foram muito bem sucedidos e apresentaram inúmeras vantagens sobre os sistemas existentes. Razão pela qual justificavam os custos de implantação destes tipos de sistemas de controle, com isso surgiram os primeiros controladores lógicos programáveis.

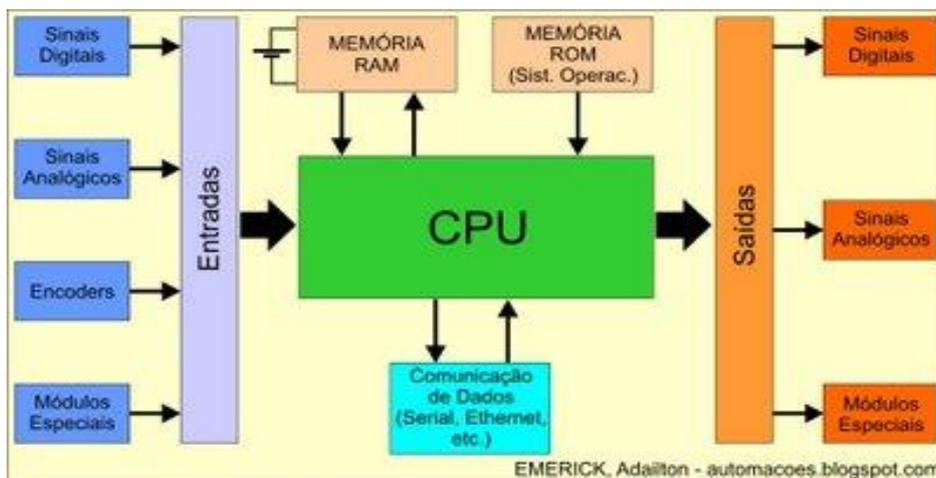
Durante este mesmo momento, a indústria da eletrônica teve um avanço tecnológico muito notável e com o advento dos microprocessadores, o controlador lógico programável ou por denominação PLC pôde receber muitas melhorias e recursos, tais como: interfaces de comunicação, recursos de comunicação por vários tipos de redes, manipulação de dados, inserção de módulos intercambiáveis para ampliações, dentre outras. [1]

2.2.1. Arquitetura do PLC

Basicamente, um PLC é um dispositivo de estado sólido como um computador, e é constituído por alguns blocos principais a seguir:

- **CPU** (Unidade Central de Processamento): Onde está presente o processador e as unidades de memória RAM e ROM e demais circuitos lógicos.

- **Módulos de entradas e saídas I/O's:** Estes são responsáveis por realizarem os acionamento e receberem informações do processo, estes podem ser módulos de sinais digitais, analógicos ou até mesmo para redes de expansão.
- **Fontes de Energia:** Responsáveis pela alimentação da CPU e dos módulos de entradas e saídas.
- **Base ou Rack:** Pode ser um barramento que conecta mecanicamente e eletricamente todos os componentes descritos acima. [2]



Fonte: <<http://www.automacoes.com/2008/12/o-plc-parte-1.html/>>

Figura 04: Estrutura de um PLC.

O PLC, através da CPU, realiza uma leitura constante dos sinais de entrada analógicos ou digitais, que chegam até os módulos de entrada. Estas informações que chegam são armazenadas na memória RAM, para serem posteriormente analisadas pelo programa lógico desenvolvido especificamente para aplicação. Com os sinais analisados pelo programa de aplicação, a CPU executa os acionamentos através dos módulos de saída, este ciclo será contínuo enquanto o PLC estiver executando seu programa lógico.

O desenvolvimento do programa de aplicação dos PLC's pode ser realizado por programas específicos instalados em microcomputadores comuns, como é o caso de alguns fabricantes. Atualmente, a linguagem mais utilizada é a linguagem Ladder (*RLL – Relay Ladder Logic*), que se tornou muito popular por se parecer muito com os antigos sistemas de controle a relés. [2]

2.2.2. CPU - Unidade Central de Processamento

Assim como em um computador, esta pode ser considerada a parte principal do PLC. Esta parte é o “cérebro” que controla o funcionamento do dispositivo e se constitui de processador, várias memórias e um barramento. Porém, o desempenho e capacidades dos PLC's estão diretamente ligados à qualidade dos processadores e seu poder para executar operações matemáticas e de manipulação de dados, estes podem ainda ser microprocessadores ou microcontroladores. [1]

O processador atua continuamente no sistema de memória conforme o programa de execução, que é desenvolvido por cada fabricante e realiza as funções desejadas pelo programa de aplicação desenvolvido pelo usuário, gerenciando dessa maneira todo o controle lógico. [2]

Contudo, o processador tem por principal função o gerenciamento de todo o sistema que constitui o PLC. Para tal, o programa de execução que é semelhante a um sistema operacional, realiza constantemente um escaneamento de todo o sistema, analisando assim todas as ações a serem tomadas.

Os fatores mais importantes em um processador são primeiramente sua velocidade de operação (ou *clock*), o tamanho de informações que podem manipular, que são normalmente 16 ou 32 bits, os conjuntos de instruções de programação que podem ser executadas e o número de memórias e sinais de entrada e saída que podem ser utilizados. Estas características tornam o PLC mais ou menos poderoso.

2.2.3. Interfaces de Entrada e Saídas

As interfaces de entradas e saídas possibilitam ao CPU do PLC interagir com o meio físico, as mesmas também podem ser chamadas de módulos de I/O, que podem ser constituídos de diversos circuitos que protegem e isolam a CPU. Existem várias categorias de módulos de I/O's que executam as mais diversas funções, dentre todos estes, os mais comuns são os módulos de sinais discretos e os de natureza numérica, como os módulos de entradas ou saídas analógicas. [1]



Fonte: <<http://www.intereng.com.br/produtos/rockwell-automation/io-distribuidos/compact-logix-io/>>

Figura 05: Exemplo de Módulos de entrada e saída.

Basicamente, estes módulos são de entrada e saída, mas também existem módulos combinados com ambos os recursos. Os módulos de entrada recebem informação do ambiente ou do processo a ser controlado, estas informações vem na forma de sinais de vários tipos de dispositivos, tais como botões, chaves, sensores, transdutores, dentre outros. Estes sinais vem em valores adequados que são processados pelo PLC.

Já os módulos de saída, realizam os acionamentos de outros dispositivos, como motores, válvulas, sinalizações, atuadores ou até mesmo outros PLC's, a fim de automatizar, controlar ou corrigir algum processo. Estes acionamentos são realizados segundo o programa de aplicação desenvolvido pelo o usuário. Os módulos de saída possuem algumas características construtivas relevantes, tais como:

- Para proteção da CPU, estes módulos são isolados através de foto acopladores. No qual não há conexão elétrica entre os dispositivos externos, sensores ou motores.
- Normalmente os módulos possuem indicações como LED (*Light Emitting Diodes* – Diodo Emissor de Luz), localizados na parte frontal.
- Os módulos de sinais discretos informam quais entradas estão recebendo sinais ou quais saídas estão sendo acionadas. Alguns blocos ainda podem trazer algumas informações como falhas no módulo ou falta de energia.
- Na maioria dos módulos, os conectores são removíveis. Reduzindo assim o tempo de manutenção no caso da necessidade de troca do módulo.
- O módulos de sinais discretos utilizam sinais em 0 ou 1, e são utilizados na maioria dos PLC's.

Cada sinal nos módulos discretos, de entrada ou saída, é tratado como um bit de um endereço da tabela de dados. Esta tabela de dados é acessada a cada execução do programa de aplicação e o número de entradas ou saídas determina a densidade do módulo. No caso dos módulos de saída, quanto maior a densidade do módulo, menor a corrente que cada saída pode fornecer. Já os módulos analógicos, são utilizados em sistemas contínuos, tratando de sinais como temperatura, pressão, corrente, dentre outros.

Através de conversores de sinais analógico/digital, estes módulos transformam os sinais dos mais diversos sensores, como transdutores ou termopares, em sinais digitais, fornecendo informações adequadas para a CPU. Nos módulos analógicos de saída, os sinais digitais fornecidos pela CPU são convertidos por conversores digital/analógico e geram assim sinais de tensão ou corrente para acionar os dispositivos de saída como atuadores ou reguladores.

Diferentemente dos módulos discretos, cada entrada ou saída analógica nos módulos analógicos são denominados canais. Os valores recebidos ou enviados nesses canais também são armazenados em endereços específicos da tabela de dados, e a quantidade de bits que formam esses valores determina a resolução do sinal analógico. [2]

2.2.4. Fonte de Alimentação

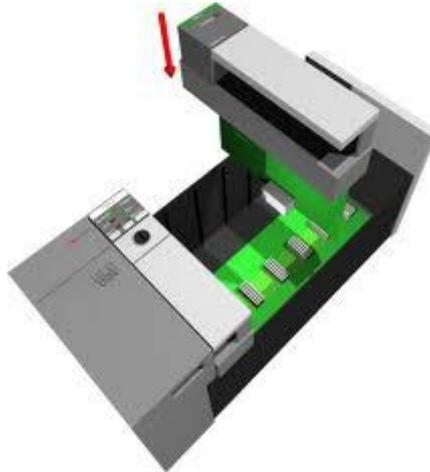
Além de fornecer a tensão necessária para a CPU e nos módulos de entrada e saídas, também desempenha um papel de proteção. Garante os níveis adequados de tensão para a alimentação do sistema e monitora constantemente o fornecimento de energia da rede.

As fontes mais utilizadas nos últimos anos são as fontes chaveadas, que possuem ajuste automático e melhor qualidade no sinal. Essas fontes possuem proteções externas específicas, conforme determinação de cada fabricante. Entretanto, existem algumas proteções padrões, como aterramento, transformadores isoladores, dispositivos de proteção contra surtos na rede, dentre outros, conforme normas técnicas locais.

Nos módulos de entradas e saídas, as fontes do PLC são responsáveis pela alimentação do circuito lógico dos módulos, sendo necessária uma fonte de alimentação externa para o circuito de potência ou para circuitos de alguns dispositivos externos presentes no processo a ser controlado.

2.2.5. Base ou Rack

A base, também é chamada de Rack, é a estrutura física onde são acomodados fisicamente todos os elementos do PLC. Interconectando eletricamente os componentes como CPU e módulos de entradas e saídas. Nesta conexão, a base fornece em seu barramento os sinais para transferência de dados, endereçamento e controle além da alimentação para os componentes do PLC. Vale destacar que algumas famílias de PLC's, de alguns fabricantes, não utilizam bases. Nestes casos, os módulos de entradas e saídas e a CPU são conectados diretamente por encaixes, alojamentos e acopladores ou até mesmo por cabos.



Fonte: <http://www.plcdev.com/how_plcs_work/>

Figura 06: Exemplo de uma base ou Rack.

As posições onde são instalados os módulos ou a CPU são chamados de slots ou ranhuras e a quantidade de slots disponíveis varia conforme a capacidade ou família do PLC a qual foi projetado para atender as necessidades da aplicação, e também de fabricante para fabricante. Na maioria dos casos, os módulos podem ser alojados livremente em qualquer posição na base.

2.3. REDE ETHERNET TCP/IP

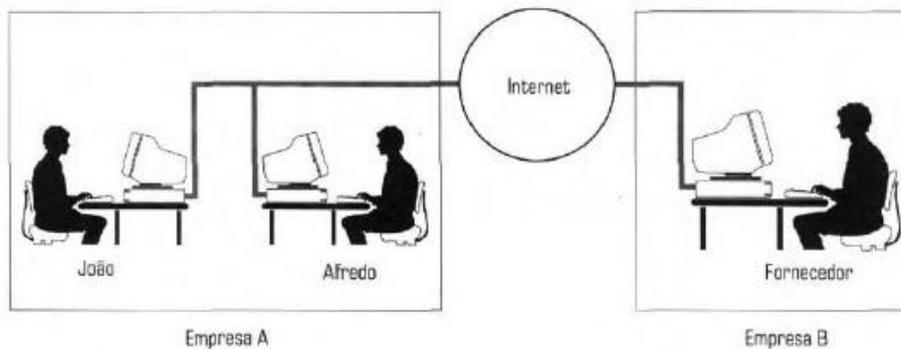
2.3.1. Introdução a Redes

Atualmente, é difícil algum tipo de rede de computadores não fazer parte da vida dos indivíduos. Muitas vezes, mesmo sem perceber, elas estão presentes em inúmeros estabelecimentos comerciais, supermercados, farmácias e agências bancárias. Estas redes interligam vários computadores em tempo real, facilitando o gerenciamento de informações e utilizando um ou mais bancos de dados. Permitindo assim um controle mais eficaz e mais preciso das informações nos mais diversos processos.

Devido às necessidades de receber e enviar informações para lugares distantes, foram desenvolvidas as primeiras redes de computadores, que não são novas, posto que existem desde a época dos primeiros computadores. As novidades ficam por conta dos

avanços tecnológicos e das padronizações que possibilitaram o barateamento e o melhor desempenho na comunicação dos computadores.

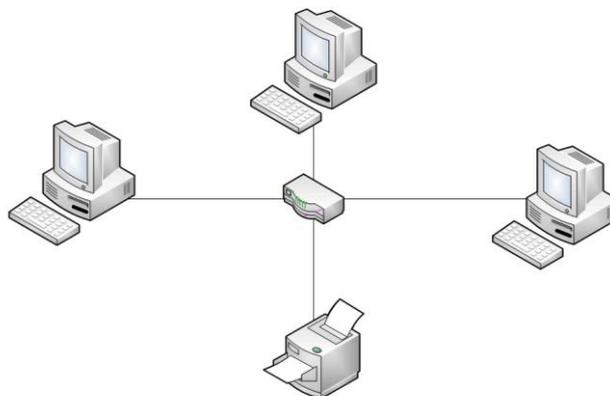
Com maior facilidade para instalação de redes, este recurso se tornou essencial no desempenho. Principalmente em empresas e escritórios de todos os seguimentos, sendo altamente utilizado, mesmo em estabelecimentos onde há poucos computadores. A necessidade de troca de informação contínua tornaria inviável levar estas informações manualmente de um computador a outro, através de discos removíveis ou mídias por exemplo. [3]



Fonte: [3]

Figura 07: Exemplo de uma rede de computadores.

“Além da facilidade de troca de informações – como arquivos -, há a vantagem de compartilhar periféricos como uma impressora ou um modem (para acesso a Internet, por exemplo), podendo significar uma redução nos custos de equipamentos.” [3]



Fonte: <<http://thiagofrodrigues.blogspot.com.br/>>

Figura 08: Exemplo de computadores compartilhando periféricos.

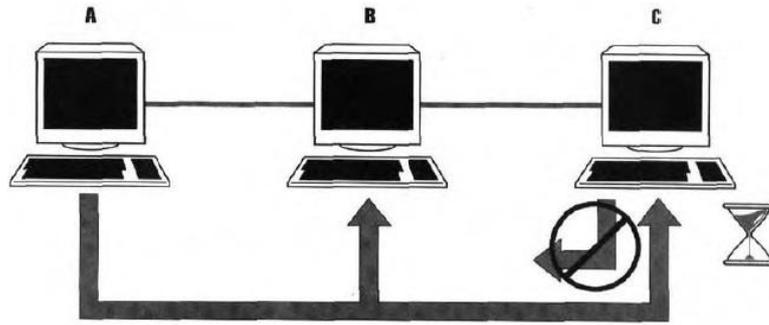
Cada vez mais as empresas estão adotando os mais variados tipos de redes, que vão desde redes de escritórios a redes industriais, buscando sempre o aumento da produtividade através da interação e troca de informações com maior praticidade. Razão pela qual, nos últimos anos, vários desenvolvedores vem trabalhando e investindo no aperfeiçoamento tecnológico destas redes. [3]

2.3.2. Protocolos

Todos os dispositivos de uma rede se comunicam e se entendem através uma mesma linguagem, independentemente do fabricante. Esta linguagem também é conhecida como protocolo.

Os protocolos são parte fundamental em todos os tipos de redes, pois os mesmos determinam qual será o funcionamento de transferência dos dados pela rede. Para que os equipamentos consigam interpretar os dados enviados pela rede, devem utilizar o mesmo protocolo.

Os protocolos viabilizam a comunicação de rede, solucionando muitos problemas de transmissão de dados. Como os dispositivos estão conectados apenas por um único cabo, a transmissão de dados não pode ser realizada simultaneamente para vários equipamentos. Se fosse necessário enviar um arquivo muito grande de uma só vez, levaria muito tempo, e conseqüentemente os outros dispositivos teriam que esperar. Conforme a figura a seguir, na qual enquanto o computador A envia uma informação para o computador B e por sua vez o computador C não pode enviar seus dados. [3]



Fonte: [3]

Figura 09: Não é possível transmitir dados se o barramento está ocupado.

Os protocolos solucionam vários problemas nas redes, como colisão de dados e velocidade de transmissão, por isso, são fundamentais em redes de computadores e redes industriais. Para transmitir um determinado dado, o protocolo divide a informação em vários blocos menores chamados pacotes, isso significa que o dado não é transmitido todo de uma vez, mas sim aos poucos. Cada pacote possui informações de endereçamento que identificam a origem e o destino do pacote.

Os computadores na rede reconhecem que determinado pacote foi enviado para ele, pois estes pacotes possuem o endereço da placa de rede do computador de destino em seu cabeçalho de destino.

A transmissão de dados divididos em vários pacotes aperfeiçoa muito a utilização da rede por vários dispositivos, pois ao invés de um arquivo ocupar a rede em um único pacote, congestionando a rede, ele é dividido para que outros dispositivos encontrem espaço para transmitir também outros dados. Dessa maneira, vários dispositivos podem se comunicar teoricamente ao mesmo tempo, mas na realidade o que acontece é que os pacotes são todos intercalados.

A velocidade da rede está diretamente relacionada ao número de transmissões que estão sendo executadas, pois quanto maior o número de pacotes enviados ao mesmo tempo, mais lenta estará a rede. Quando a placa de rede de um computador for colocar um pacote de dados para ser transmitido na rede, ele realiza uma conta conhecida como checksum ou CRC (*Cyclical Redundancy Check*).

Nesta conta, a placa de rede soma todos os bytes do pacote de dados e envia esse valor junto com o pacote. Essa mesma conta será executada pela placa de rede do dispositivo receptor e depois a placa irá comparar o resultado com o valor enviado pelo outro dispositivo, se os valores forem iguais significa que o pacote enviado não foi corrompido. Se houver uma diferença nos valores, quer dizer que os dados foram

corrompidos em algum momento da transmissão. Quando ocorre uma situação dessas, o dispositivo receptor solicita uma nova transmissão do pacote que foi corrompido. [3]

2.3.3. Modelo OSI

Com o intuito de facilitar a interconexão de redes de computadores de fabricantes diferentes, a ISO (*International Standards Organization*) criou um modelo chamado OSI (*Open Systems Interconnections*), que serve de referência para os protocolos dos vários fabricantes. Embora este seja um modelo internacional para qualquer fabricante, a maioria dos protocolos existentes não segue exatamente este modelo, os principais protocolos correspondem apenas a uma parte do modelo OSI. [3]

O modelo OSI é constituído de sete camadas, para transmissão de um dado que vem sempre da camada superior. A camada seguinte recebe o dado da camada superior e acrescenta mais informações correspondentes a sua função e posteriormente transmite para a próxima camada.



Fonte: <<http://www.diegomacedo.com.br/entendendo-o-modelo-osi/>>

Figura 10: As sete camadas do modelo OSI.

Muitos protocolos existentes utilizam este conceito de camadas, entretanto nem sempre estas camadas possuem o mesmo nome ou aplicações das camadas do modelo OSI. Alguns protocolos podem ser também um conjunto de protocolos, pois em cada camada pode haver um protocolo envolvido.

O modelo OSI é basicamente dividido em três partes, uma parte das camadas é destinada a aplicação, outra ao transporte e outra para a rede. As camadas de rede são

camadas de baixo nível e são responsáveis pela transmissão e recepção de dados na rede. Já as camadas de transporte, recebem os dados da rede e se encarregam de tratá-los e repassá-los para as camadas de aplicação. Por fim, as camadas de aplicação, que são camadas de alto nível, atuam no tratamento e codificação de todos os dados para uma forma compreensível para a aplicação no programa do receptor final.

Na figura acima, a camada sete é a camada de aplicação, pois ocorre a interação entre protocolo de rede e o aplicativo que irá transmitir ou receber dados. [3]

A camada seis é a camada de apresentação. Recebe os dados da camada de apresentação e os converte para um formato utilizado para transmissão de sinais. Ainda pode ter o recurso de compressão de dados, tornando a transmissão de dados mais rápida, pois serão menos dados para trafegar.

A camada cinco, ou sessão, cria uma sessão de comunicação entre dois dispositivos, estabelece como serão transmitidos os dados e realiza marcações nos dados. Assim, se por algum motivo a transmissão for interrompida, quando a rede se estabelecer novamente a transmissão de dados continua do mesmo ponto onde estava.

Por sua vez, a camada quatro é a camada de transporte. Tem por função receber os dados da camada de sessão e dividi-los em pacotes, que serão transmitidos pela rede. Quando esses pacotes chegam ao receptor, a camada de transporte do mesmo remonta os pacotes para formar o dado original e repassa para sua camada de sessão. A camada de transporte também separa as camadas de alto nível das camadas físicas, que são as camadas 1,2 e 3. A camada de transporte faz justamente a ligação entre estas camadas de alto e baixo nível, que cuidam mais da forma e como os dados vão ser transmitidos do que dos próprios dados.

Já a camada três, é a camada de rede. Esta camada realiza o endereçamento dos pacotes e transforma endereços digitais em endereços físicos, assegurando que os dados cheguem ao destino correto. Esta camada pode também determinar rotas de trajeto para os dados, conforme esteja o tráfego da rede ou conforme a prioridade.

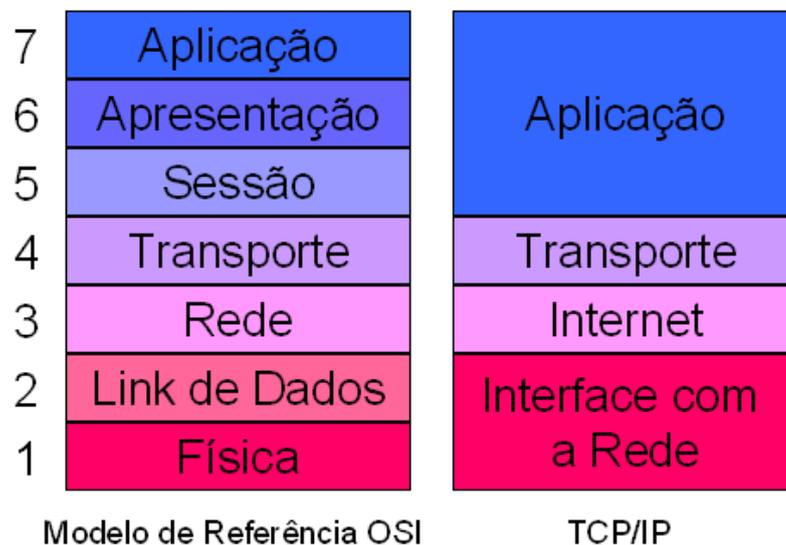
A camada dois é denominada de link de dados ou de enlace. Transforma os dados da camada de rede em quadros para navegar, acrescentando informações como, por exemplo, os endereços das placas de rede de origem e destino, dados de controle, os próprios dados, dentre outros. Posteriormente, esses dados vão para a camada física, que irá convertê-los em dados de sinais elétricos para serem enviados pelos cabos. Quando esses dados chegam ao receptor, a camada de enlace do mesmo verifica se os dados não foram corrompidos e envia uma confirmação da origem.

Finalmente, a camada física transforma os dados em sinais correspondentes aos meios da rede. Por exemplo, se o meio for óptico, a camada converterá o código lógico em sinais luminosos e se forem cabos elétricos, em sinais elétricos. Este processo de conversão é normalmente realizado pela placa de rede dos equipamentos presentes na rede. [3]

2.3.4. Modelo TCP IP

Atualmente o protocolo TCP/IP é o protocolo mais utilizado em redes de computadores devido ao aumento de usuários de internet, e também é a que mais possui investimentos na área industrial. Este protocolo foi criado pensando-se em grandes redes de computadores, e possui uma arquitetura aberta que pode ser adaptada para os sistemas operacionais dos fabricantes.

Este protocolo utiliza o protocolo OSI como referência, porém o protocolo TCP/IP possui apenas quatro, como mostra a figura abaixo:



Fonte: < <http://www.clubedohardware.com.br/artigos/1351/> >

Figura 11: Comparação do Modelo OSI com o protocolo TCP/IP.

O protocolo TCP/IP é constituído de vários protocolos, dois deles dão o nome do protocolo TCP/IP: TCP (*Transmission Control Protocol*, Protocolo de Controle da

transmissão) para as aplicações de transporte e IP (*Internet Protocol*) para operações em nível de internet. Porém, existem outros protocolos que fazem parte deste conjunto. [3]

2.3.4.1. Camada de Aplicação

Esta camada tem as funções equivalentes às camadas 5, 6 e 7 presentes no modelo OSI. Assim como no modelo OSI, esta camada se comunica com o software ou aplicativo e posteriormente com o protocolo de transporte. Esta camada é composta por vários protocolos, os principais são o DNS (*Domain Name System*), o FTP (*File Transfer Protocol*), o SMTP (*Simple Mail Transfer Protocol*), o SNMP (*Simple Network Management Protocol*), o HTTP (*Hypertext Transfer Protocol*) e o Telnet.

Um exemplo prático seria quando um aplicativo de e-mail deseja baixar os e-mail contidos no servidor, ele solicitará a camada de aplicação do TCP/IP, e então será acionado o protocolo SMTP, ou quando se deseja acessar uma página na internet utilizando www, será acionado no TCP/IP o protocolo HTTP.

Cada protocolo da camada de aplicação utiliza uma porta diferente para se comunicar com a camada de transporte, por exemplo, o protocolo HTTP se comunica sempre pela porta 80, de maneira que cada protocolo utilizando números de portas distintos a camada de transporte pode determinar qual o tipo de pacote de dados vai transmitir. No lado do receptor, sua respectiva camada de transporte reconhecerá para qual protocolo de aplicação deverá entregar o pacote de dados. [3]

2.3.4.2. Camada de Transporte

A camada de transporte do modelo TCP/IP corresponde a camada 4 do modelo OSI, e como no modelo OSI sua função é transportar os dados da camada de aplicação e enviá-los pela rede. Transforma os dados em pacotes para serem repassados para as camadas seguintes. [3]

No modelo TCP/IP, ocorre a multiplexação de dados de várias aplicações ao mesmo tempo. Neste esquema, várias aplicações podem se comunicar com a rede ao mesmo tempo. Os dados são transmitidos intercaladamente, dessa forma, não é necessário terminar uma aplicação para então começa outra. [3]

Na recepção destes dados, que são repassados pela camada de internet, os quadros de dados podem percorrer vários caminhos diferentes. Os mesmos podem chegar fora de ordem, pois a camada de transporte organiza os dados e verifica se eles chegaram corretamente. O protocolo IP não realiza esta verificação dos pacotes de dados, somente o

protocolo TCP o faz, e se caso estiver faltando alguma informação, o mesmo solicita uma nova transmissão de dados. [3]

2.3.4.3. Camada de Internet

Esta camada do modelo TCP/IP é correspondente a camada 3 de rede do modelo OSI. Nesta camada também existem vários protocolos que podem operar, tais como: IP (*Internet Protocol*), ICMP (*Internet Control Message Protocol*), ARP (*Address Resolution Protocol*) e RARP (*Reverse Address Resolution Protocol*). [3]

Os pacotes recebidos da camada de TCP são divididos em outros pacotes, que são normalmente conhecidos como datagramas. Estes datagramas vão para a camada de interface com a rede e depois são enviados pela rede através de quadros, a camada de internet não verifica se os dados chegaram ao destino, a camada TCP é que se encarregará desta tarefa.

A principal função desta camada é o roteamento dos pacotes, adicionando informação sobre o trajeto que o pacote deverá realizar.

2.3.4.4. Camada de Interface com a Rede

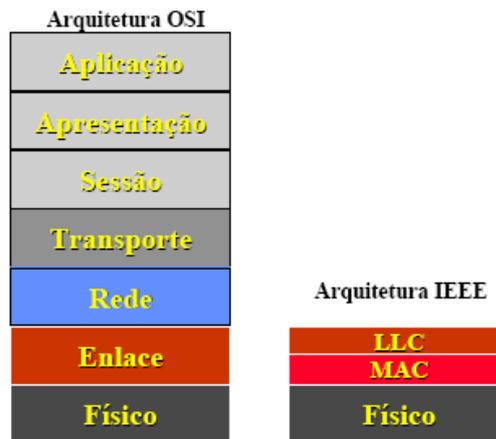
Correspondente às camadas 1 e 2 do modelo OSI, a camada de interface com a rede recebe os datagramas da camada de internet e transforma em quadros e então os envia pela rede. Seu funcionamento é similar ao descrito anteriormente no modelo OSI.

2.3.5. Ethernet

O padrão Ethernet se tornou o mais utilizado atualmente em redes locais e este padrão trata da parte física e de como os dados devem ser enviados pela rede. Este padrão está presente nas camadas 1 e 2 do modelo OSI e podem ser trocadas pelas três camadas do padrão Ethernet. [3]

A principio a Ethernet operava a 10 Mbps, porém com os novos padrões que foram desenvolvidos esta rede pode trabalhar a velocidades muito altas como 100Mbps, 1Gbps e até 10Gbps. [4]

Sua função é receber os pacotes de dados provenientes das camadas superiores e montá-los em quadro para serem enviados pela rede. O protocolo Ethernet define também a forma de como esses quadros trafegarão na rede como frequência, nível e formato do sinal. Abaixo na figura 10 segue uma representação das camadas do padrão Ethernet. [3]



Fonte: < [http://estudeccna.com.br/tutoriais/redes-ethernet-cabeamento />](http://estudeccna.com.br/tutoriais/redes-ethernet-cabeamento/)

Figura 12: Comparação camadas físicas do Modelo OSI com o padrão Ethernet.

As camadas e suas características da arquitetura Ethernet são:

- LLC (*Controle do Link Lógico*): Esta camada agrega informações ao quadro de dados sobre o protocolo de alto nível e seus aplicativos, que transmitiu o pacote de dados. Dessa forma, quando a máquina de destino receber o quadro de dados, a mesma identificará para qual protocolo deve entregar os pacotes.
- MAC (*Controle de Acesso ao Meio*): Esta camada realiza a montagem do quadro de dados com várias informações para transmissão do quadro pela rede, incluindo também um cabeçalho próprio.
- Camada Física: É responsável por transmitir os quadros recebidos da camada de acesso ao meio utilizando o método CSMA/CD (*Carrier Multiple Access with Collision Detetion*). E também determina toda a parte física da rede, como cabos e conectores.

A partir da Ethernet comercial, o grupo 802.3 do comitê de padrões LAN IEEE 802 definiu uma série de padrões de controle de acesso ao meio que foi designado de CSMA/CD. [4]

Através do método CSMA/CD, antes de uma estação transmitir um dado, primeiramente ela verificará se não existe outra transmissão em andamento, se estiver livre, então a estação iniciará a transmissão. Se caso duas estações transmitirem ao mesmo tempo, ocorrerá uma colisão, o que impedirá que os dados de ambas as estações cheguem ao destino. Se ocorrer uma colisão ambos os transmissores enviarão um sinal de colisão

para as outras estações e aguardarão um tempo aleatório para voltar a transmitir. Uma das grandes vantagens desse protocolo é sua facilidade de implementação. [4]

2.3.6. Placas de Rede

Basicamente as placa de rede Ethernet transmitem os quadros de dados vindos da camada de Controle de Acesso pelo cabeamento, dessa maneira as placas de rede são responsáveis pela codificação e a utilização do protocolo CSMA/DC.

Atualmente, o conector mais comum instalado nas placas de rede é o RJ-45, porém também existem conectores coaxiais finos e grossos ou ainda de fibra óptica. [3]

2.4. MICROCONTROLADORES

Este dispositivo vem sendo cada vez mais utilizado comercialmente, é considerado um dispositivo eletrônico que possui inteligência programável capaz de executar operações de controle lógico. [6]

O microcontrolador pode ser considerado como um computador em um circuito integrado, e vem contribuindo muito para a automação do mundo atual. Estes dispositivos estão presentes principalmente em equipamentos portáteis de baixo custo. [5]

Este dispositivo trata-se de um semicondutor em forma de CI, onde estão presentes todas as unidades básicas que constituem um microcomputador como, por exemplo, microprocessador, memórias não voláteis (ROM, EEPROM, etc), memórias voláteis (RAM, Flash Ram, etc), portas de entradas e saídas digitais, analógicas, de comunicação (Serial, Paralela), dentre outros. Possui, normalmente, uma limitação na memória de dados, por isso este dispositivo é mais utilizado em aplicações pequenas e específicas. [5]

2.4.1. Arquiteturas

As duas principais arquiteturas atualmente são as arquiteturas Harvard e von Neumann, onde a arquitetura von Neumann é possui a característica de uma máquina digital armazenar os programas na mesma área de memória de dados na comunicação com a CPU. Devido a essa arquitetura, quando a CPU está acessando a memória de programa não é possível acessar a memória de dados, pois se utiliza um único barramento. Utiliza separadamente uma unidade central de processamento e outra de memória.

Na arquitetura Harvard, o microcontrolador processa os dados e o programa de execução através de dois barramentos internos, por onde passam as funções e os dados. Um barramento é destinado para os dados e o outro para as instruções. Neste tipo de

arquitetura, o processamento é mais rápido por utilizar dois barramentos, pois enquanto uma instrução é executada, outra pode ser acionada na memória interna. [6]

2.4.2. Processador com núcleo ARM Cortex-M3

O processador ARM Cortex-M3 proporciona um alto desempenho e uma plataforma de baixo custo, que utiliza menos memória, menos pinos e baixo consumo de energia. Apesar dessas características, este processador proporciona um excelente desempenho computacional e um bom sistema de resposta para interrupções. Algumas características deste processador são:

- Núcleo Compacto;
- Rápida execução de aplicação, através da arquitetura Harvard por dois barramentos;
- Possibilidade de migração para família de processadores ARM7 para obter um melhor desempenho;
- Manuseio de interrupções com implementação do registrador de manipulação para manobra das interrupções no hardware;
- Processamento rápido e determinístico de interrupções.

2.4.3. Microcontrolador Stellaris LM3S6965

A família de microcontroladores chamada de Stellaris é fabricada pela Texas Instruments Incorporated, traz dispositivos de alto desempenho para aplicações com microcontroladores de 32 bits e baixo custo. Esta família oferece ainda uma larga aplicação e facilidade de integração com outros sistemas e é capaz de atender algumas aplicações com um considerável nível de conectividade e processamento.

Atualmente, o microcontrolador Stellaris LM3S6965 é largamente utilizado para muitas aplicações industriais, tais como: monitoramento e controle remoto, máquinas eletrônicas de vendas no comércio, equipamentos de teste e medição, aparelhos de rede como switches gerenciáveis, automação de fábricas, controle de movimento, equipamentos médicos e hospitalares e segurança residencial e predial.

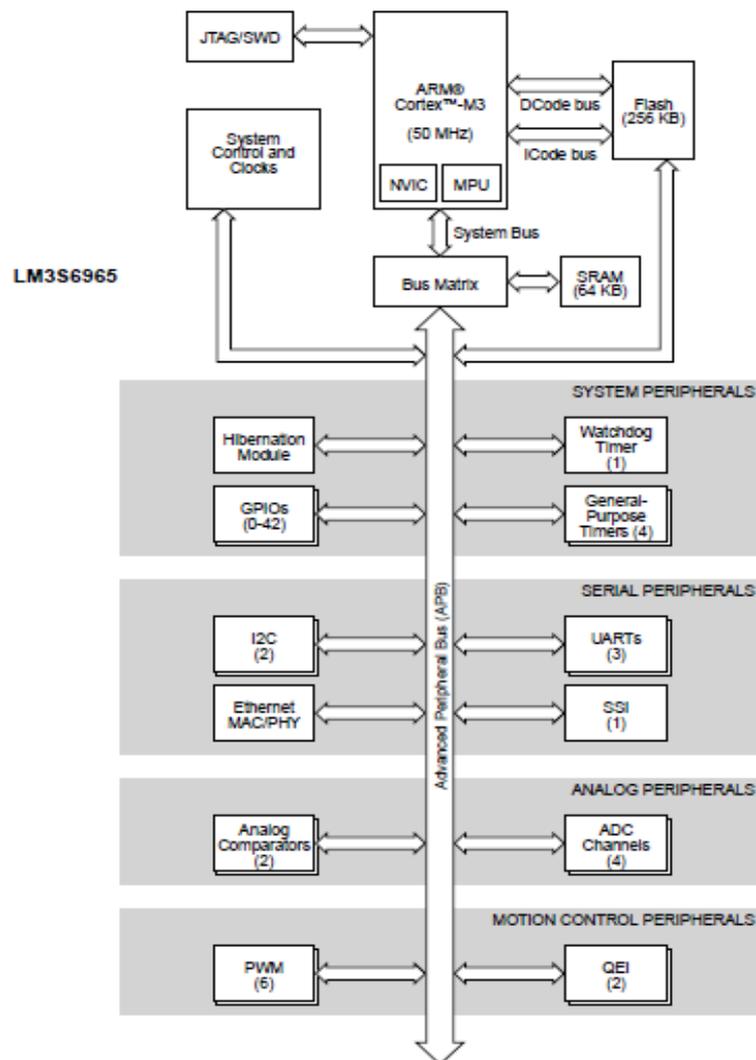
Quando alguma aplicação necessita de extrema economia de energia, este microcontrolador possui funções para entrar em estado de hibernação, o que alcança um

baixo consumo da fonte de alimentação. Quando o mesmo passa por longos períodos de inatividade, este módulo de hibernação torna o microcontrolador LM3S6965 totalmente viável para aplicações com pilhas ou baterias.

Este dispositivo possui muitas características, dentre elas algumas são:

- Processador ARM Cortex-M3;
- Composto por 100 pinos;
- Frequência de operação de 50 Mhz ;
- 42 pinos que podem ser utilizados como entradas ou saídas;
- Memória de programa do tipo Flash de 256 KB;
- 64 KB de memória SRAM;
- Controlador Ethernet 10/100 conforme especificação IEEE 802.3;
- Quatro canais analógicos;
- Também contém várias características especiais, como módulo de hibernação, JTAG, UART, SSI, I2C, comparadores analógicos, dentre outros.

A figura a seguir apresenta a arquitetura do micro controlador LM3S6965:



Fonte: <Datasheet Microcontrolador LM3S6965 Texas Instruments >

Figura 13: Arquitetura interna Microcontrolador LM3S6965.

2.4.4. Controlador Ethernet

O controlador Ethernet, nos microcontroladores da família Stellaris, são integrados ao controle de acesso ao meio (MAC) e interface de rede física (PHY) e suportam os padrões 10BASE-T e 100BASE-TX.

Algumas características do módulo Controlador Ethernet são:

- Auto negociação;
- Múltiplos modos de operação;
 - - Full/Half Duplex 100 Mbps
 - - Full/Half Duplex 10 Mbps
- Altamente configurável;

- Endereço MAC programável
- Controle de correção de erro CRC
- Interrupções configuráveis
- Manipulação física de mídia;
 - Correção automática cross-over MDI/MDI-X
 - Registro programável de transmissão de sinal

Devido ao controlador ethernet, que é completamente integrado ao microcontrolador LM3S6965 para estabelecer uma interface ethernet basta somente um conector RJ45 especial com transformador, leds e alguns componentes passivos. Como estes microcontroladores suportam correção automática MDI/MDI-X é possível conectar-se diretamente a uma rede ou a outro dispositivo ethernet sem a necessidade de utilizar um cabo *cross-over*.

2.5. LINGUAGEM HTML

A Linguagem HTML é uma abreviação de *Hyper Text Markup Language*, e se trata de uma linguagem muito utilizada para criação, edição e exibição de páginas Web. Baseia-se em textos comuns e códigos específicos que também são conhecidos como tags, estes tags nada mais são do que comandos da linguagem que serão tratado pelo navegador na página Web.

Em resumo, pode ser considerada uma linguagem simples que permite formatar textos, inserir imagens ou criar ligações entre páginas. Possibilitando assim que as páginas tragam muita informação, dos mais variados gêneros.

Como em qualquer outra linguagem, na HTML o código-fonte é conforme as regras de sintaxe da linguagem, que uma vez elaborada, será posteriormente interpretada pelo navegador, que executará os comandos durante a exibição da página. Não é necessário estar conectado à internet para elaboração, pois o browser trabalha independentemente de haver uma conexão com a internet. [9]

Normalmente, o código das páginas web em HTML pode ser escrito até mesmo nos mais simples editores de textos. Entretanto, é altamente recomendável a utilização de editores específicos de HTML, pelo fato de haver mais recursos pré-determinados, o que permite que o tempo de edição da página seja encurtado.

Os documentos que estão acessíveis em sites da web podem ser vistos por qualquer pessoa no mundo, basta ter acesso à internet. Estes documentos podem conter textos de diferentes tamanhos, cores e estilos, assim como imagens, dados e links para outros

documentos associados. O HTML é uma linguagem destinada a essa categoria de documentos. [10]

2.5.1. Formulários

Os formulários têm a função de serem preenchidos com informações para serem enviados mais tarde. Os formulários Web seguem o mesmo conceito, porém são mais rápidos e dinâmicos. Contudo, os servidores Web necessitam de scripts CGI para estabelecer uma conexão entre o navegador e o servidor. Neste processo, o formulário recebe informações vindas do usuário e os scripts CGI realizam o processamento das informações pertinentes. [14]

Dentre as tags mais utilizadas, uma que se destaca é **<FORM>** e **</FORM>**, utilizadas em pares. Definindo dessa maneira o início e o fim do formulário. Este tag tem dois atributos, **ACTION** e **METHOD**.

O atributo **ACTION** nada mais é do que a URL do script CGI, para onde serão enviadas as informações. [14]

O atributo **METHOD** controlará a maneira como as informações chegarão ao destino indicado no atributo Action. As opções no atributo Method são duas: "POST" e "GET". Estas opções devem ser escolhidas conforme o scripts CGI selecionado.

É comum utilizar os atributos juntos, como no exemplo abaixo:

```
<form method="get" action="iocontrol.cgi" name="iocontrol">
...
</Form>
```

Outras tags muito utilizadas são as seguintes:

- **<TEXTAREA>** e **</TEXTAREA>** - Este tag permite que seja digitado textos livres em um determinado campo de edição. Seus atributos são:
 - **NAME** – Determina o nome do campo de edição.
 - **ROWS e COLS** – Determina a largura e altura do campo de edição.
- **<SELECT>** e **</SELECT>** - Disponibiliza uma lista de opções para serem enviadas. Seus atributos são:
 - **NAME** – Determina o nome que será relacionado com o valor escolhido. [14]

- **SIZE** – Determina a quantidade de opções que estarão disponíveis ao usuário.
 - **MULTIPLE** – Determina se um grupo de select irá escolher várias opções ao mesmo tempo.
 - **OPTION** – Determina cada um das opções para serem escolhidas.
- **<INPUT>** - Disponibiliza uma variedade de opções de entradas para serem inseridas no formulário. Esta tag possui uma linha de comando, como no exemplo abaixo:

```
<input name="LEDOn" value="1" type="checkbox">
```

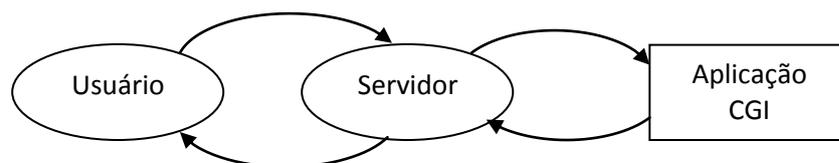
No exemplo acima, o atributo "name" corresponde ao nome que será relacionado com a entrada inserida. Value determina o valor que poderá ser inserido no campo de entrada. E por fim, o atributo type determina o tipo de controle. No caso do exemplo será um checkbox. [14]

2.5.2. CGI (*Common Gateway Interface*)

Atualmente, existe cada vez mais a necessidade de sites e páginas com animações e interações mais dinâmicas. Para proporcionar uma interação desse nível com a Web, é necessário haver uma interface executando programas por trás do servidor. O CGI (*Common Gateway Interface*) pode ser considerado um meio de acesso entre interfaces web e outras fontes de dados.

Com este recurso, define-se uma interface que executará aplicações e programas externos que serão utilizados no servidor Web. Através de aplicações CGI pode-se incluir funções externas que deixam as interfaces Web diferenciadas.

Estas aplicações CGI manipulam informações solicitadas ou fornecidas pelo servidor. O servidor manuseia as informações solicitadas pelo usuário e então aciona as aplicações CGI, que por sua vez trabalham segundo este canal de entrada de dados e retornaram um resultado para o servidor. Logo em seguida, o usuário poderá visualizar na interface Web.



Fonte: Elaborado pelo autor.

Figura 14: Interação entre aplicação CGI, Servidor e Usuário.

O CGI basicamente é um mediador entre o servidor e outras aplicações, como fontes de dados e informações externas. Os programas e scripts CGI são como extensões de um servidor, que executam tarefas com informações específicas do processo.

O principal propósito do CGI é auxiliar na comunicação com informações e serviços que estão fora do domínio do servidor Web, abrindo um canal com incontáveis informações e recursos que seriam muito complicados com outros meios.

Através da utilização de variáveis, o servidor pode utilizar uma linha de comando para transmitir dados com informações solicitadas pelo usuário. Isso permite realizar controles específicos sobre parâmetros do programa CGI, com o intuito de gerar novas ações. Este recurso fornecerá ao usuário um meio de acesso para entrada de dados e alterar algumas funções no programa CGI envolvido.

2.5.2.2. Método GET

O envio de dados para programas CGI é realizado por Formulários HTML. Estes formulários são criados com um tag conhecido como <FORM>, estes formulário contém sempre um botão de envio de formulário, que enviará as informações para o script indicado no atributo Action do tag FORM.

O método GET é utilizado para enviar informações do formulário através de um URL do próprio script. Uma URL é um endereço único de um arquivo em uma rede ou na internet, e são usadas para localizar páginas e outros tipos de arquivos.

Com um formulário HTML pode-se acionar o método GET com uma derivativa `method=get`, que faz parte do tag FORM. Através deste método, os dados do formulário são transmitidos a princípio para o servidor que as armazenará em uma variável chamada de `QUERY_STRING`.

Conforme dito acima, um script CGI será chamado através da derivativa Action do tag Form, que constituirá a parte inicial do formulário. O script irá extrair os dados da variável armazenada pelo servidor, e assim poderá receber as informações enviadas pelo usuário. [13]

2.5.2.3. Linguagem C para CGI

Cada linguagem de programação possui sua própria característica e sua filosofia, sendo cada uma delas destinadas a soluções de tipos de problemas específicos. A linguagem utilizada normalmente é escolhida por ser mais compreensível, mais eficiente e mais confiável.

Existem algumas vantagens em se utilizar a linguagem C para criar programas CGI. Primeiro, que utilizando esta linguagem pode-se gerar um código binário bem pequeno se comparado a outras linguagens de alto nível. Segundo, que estes códigos binários são executados mais rapidamente. Porém, uma desvantagem da linguagem C na programação de CGI's é que se torna mais complicado para se manipular Strings.

2.5.3. SSI (Server Side Includes)

O termo SSI está relacionado a capacidade do servidor Web misturar vários arquivos juntos dentro de um único arquivo.

Basicamente, SSIs são utilizados em aplicativos para preencher páginas HTML com conteúdos variáveis, mais conhecidos como conteúdos dinâmicos. Quando um servidor está executando um código em HTML e este código conter SSIs, no momento que o servidor passar por um comando SSI, o servidor irá executar um comando ao invés de apresentar a linha em si. O servidor analisa constantemente os tags SSIs que podem dessa maneira, executar várias ações antes das páginas serem carregadas e até mesmo quando as mesmas estão sendo visualizadas. [11]

Através deste recurso, pode-se executar códigos e scripts remotamente na página HTML que está sendo visualizada. Com este recurso, pode-se também reduzir bastante o código fonte de sua página ou site, pois uma linha SSI pode substituir dezenas de linhas com códigos em HTML. Além disso, os arquivos são menores, não sobrecarregando muito o servidor. [11]

SSIs possuem um vasto número de diferentes Funções. Dependendo do servidor, pode-se utilizar SSI para incluir arquivos, executar programas, mostrar parâmetros, dentre outros. Estas diretivas são inseridas em uma página web e posteriormente o servidor interpretará antes de enviar ao usuário. [15]

Se um usuário ativa uma página que contém SSI, o servidor pesquisará por essas diretivas em todo o código HTML e substituirá por valores determinados.

Normalmente, quando o servidor realiza essa busca por diretivas, tal processo pode exigir um processamento maior. Devido a este detalhe é utilizado uma extensão especial para que o servidor encontre estas diretivas mais rapidamente. Assim, somente os arquivos que estiverem com esta extensão, que são como estas marcações, serão interpretados.

Tais diretivas vêm sempre na forma de comentários, como no exemplo abaixo:

```
<!--#LED1txt-->
```

No exemplo acima a diretiva será usada para se comunicar diretamente com o micro controlador e irá indicar o status de uma das saídas do microcontrolador.

Um dos comandos básicos mais utilizados é o comando Config, este comando pode configurar mensagens de erro, de data, dentre outros. Por exemplo, para realizar a configuração da data o comando seguiria a seguinte sintaxe: `<!--#config timefmt=" formato" -->`. O formato pode ser determinado por:

- %a seguido do nome abreviado do dia da semana;
- %A seguido do nome completo do dia da semana;
- %b seguido do nome abreviado do mês;
- %B seguido do nome completo do mês;
- %d seguido do dia do mês;
- %d seguido do número do mês;
- %c Representação de data e hora;
- %S Segundos no formato decimal;

Outro comando básico muito utilizado é o comando Include, este comando é utilizado para incluir um arquivo dentro de uma página HTML. Sua estrutura fica a seguinte: `<!--#include virtual="nome_do_arquivo.html" -->`.

O comando echo também está entre os mais utilizados, através deste comando pode-se incluir dados de uma variável na página. Sua estrutura fica assim: `<!--#echo var="variável" -->`.

Este recurso SSI vem sendo amplamente utilizado pelos profissionais desenvolvedores de páginas e sites em HTML, pois facilita muito na criação. Poupa tempo, reduz o tamanho do código e agiliza alterações em muitas páginas. [15]

3. METODOLOGIA

Durante o desenvolvimento deste trabalho, foi necessário a pesquisa de inúmeros itens bibliográficos, como livros, sites, revistas, manuais e trabalhos publicados, que serão a base de todo conhecimento teórico necessário para o desenvolvimento do projeto.

Para demonstração prática, o projeto será constituído de duas partes, o hardware e a interface gráfica. No hardware se utilizará o kit de desenvolvimento LM3s6965 do fabricante Texas, que possui as características necessárias para as aplicações desejadas, como um

micro controlador Stellaris com núcleo Arm Cortex M3, capaz de suportar aplicações com rede Ethernet e TCP/IP. O kit também já possui os conectores RJ45 e USB instalados para acesso direto a um micro controlador.

Finalmente, para que o controlador seja acessado e parametrizado, haverá uma interface gráfica, que será acessada diretamente entre o hardware e o micro computador via conexão ethernet. Esta interface será uma página WEB desenvolvida em HTML, e nesta página será possível realizar toda a configuração do hardware, como definir quais saídas serão acionada e também seus tempos de acionamentos. Também será possível monitorar os Status do dispositivo.

3.1. PLACA DE DESENVOLVIMENTO STELLARIS LM3S6965 TI

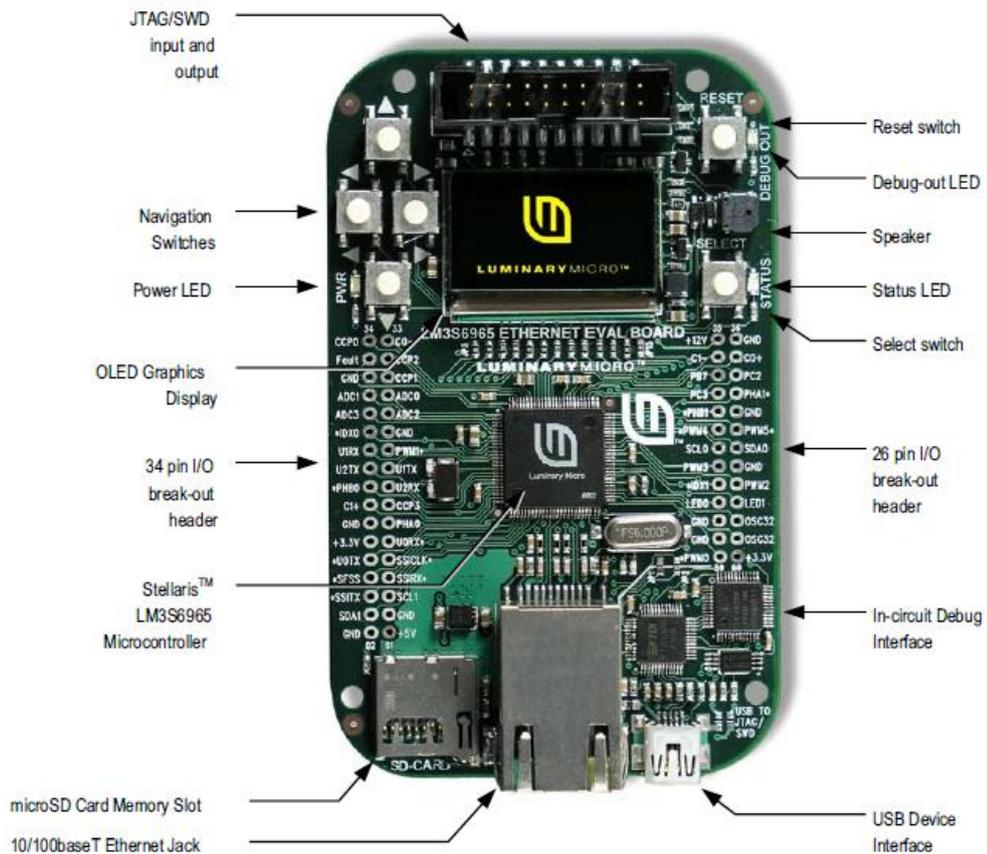
Esta placa é versátil e compacta, contemplada com o microcontrolador Stellaris LM3S6965 integrado com controlador 10/100 Ethernet que contém um servidor web.

Esta placa de desenvolvimento pode ser usado não só para testes mas como interface para depuração, no modo de depuração é desabilitado, dessa forma é possível programar ou depurar um outro dispositivo externo.

Este kit de desenvolvimento realiza testes rápidos, desenvolvimentos de protótipos e criação de aplicações para rede Ethernet. [8]

O kit de desenvolvimento apresenta as seguintes características:

- Micro controlador Stellaris LM3S6569 integrado com controlador Ethernet;
- Conexão USB para comunicação serial, depuração e alimentação;
- Display gráfico com resolução 128 X 96 pixel;
- Leds para operação, chaves e botões de navegação;
- Alto-falante;
- Pontos de IO's disponibilizados para utilização;
- Conector JTAG com entradas e saídas;
- Slot para cartão SD de memória.

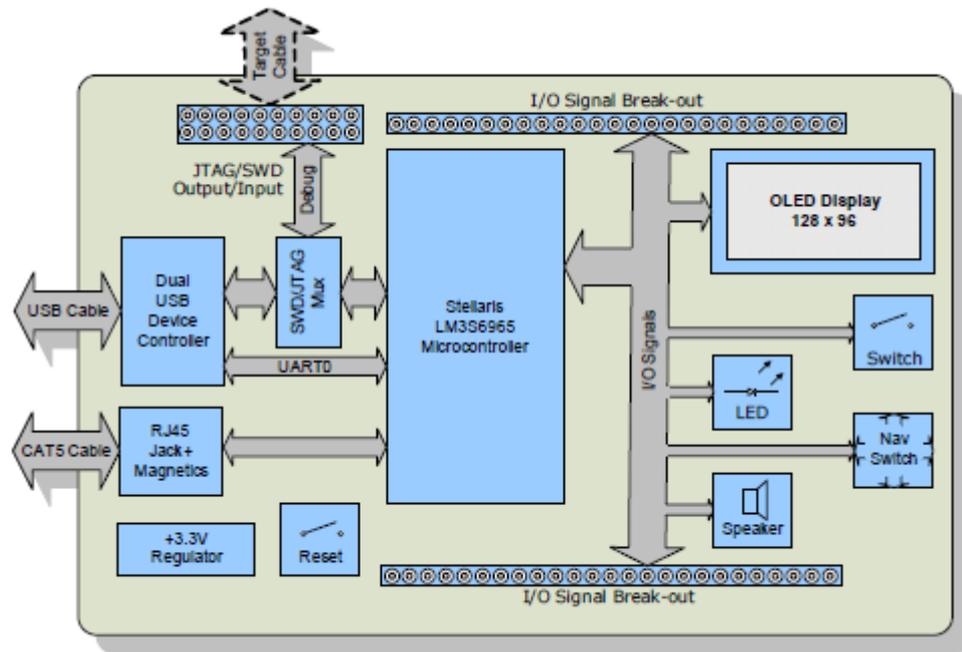


Fonte: <Users Manual Evaluation Board LM3S6965 Texas Instruments >

Figura 15: Layout Placa de Desenvolvimento Microcontrolador LM3S6965.

O kit de desenvolvimento ainda conta com vários pontos para serem utilizados. Sendo 44 pontos de I/O's, 14 pontos de energia e alimentação e 2 pontos para utilização de cristais de clock. Para utilização destes pontos basta soldar diretamente fios nas ilhas ou também soquetes ou cabeças de pinos. [8]

Abaixo, na figura 13, pode-se visualizar a arquitetura da placa de desenvolvimento.



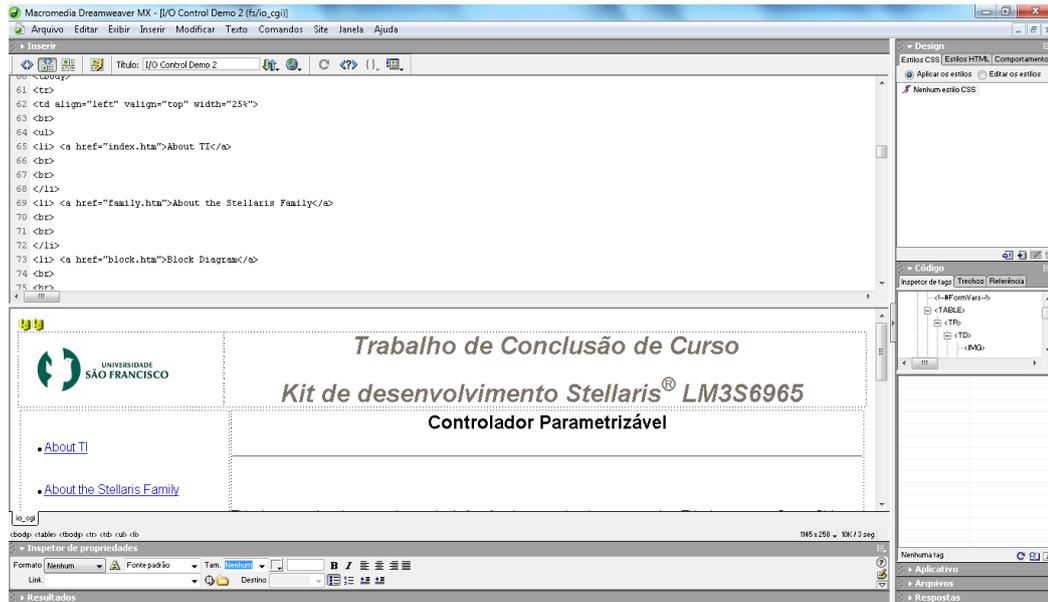
Fonte: <User's Manual Evaluation Board LM3S6965 Texas Instruments >

Figura 16: Arquitetura Placa de Desenvolvimento Microcontrolador LM3S6965.

3.2. PÁGINA WEB

Basicamente, uma página Web é um documento que pode ser exibido para qualquer computador através da internet. Este tipo de documento se constitui de textos e códigos que permitem a exibição da página. Além de textos, este documento pode conter imagens, vídeos, som, animações ou ainda links ou ligações para outras páginas. Para edição da página Web, que terá o papel de realizar a configuração do controlador, foi utilizado um software específico chamado de Dreamweaver.

O Dreamweaver MX é um dos softwares de edição em HTML mais utilizados pelos profissionais da área na edição e criação de páginas Web, sites e aplicativos para internet. Este software facilita muito o desenvolvimento dos projetos, devido a sua simplicidade. O Dreamweaver MX traz ferramentas para realizar páginas no modo texto, mas uma de suas vantagens é que ele possibilita a criação através de uma interface gráfica, facilitando bastante para quem não tem muita afinidade com códigos.



Fonte: Elaborado pelo autor.

Figura 17: Tela do software IAR Embedded Workbench.

O Dreamweaver suporta ainda a criação de animações e de páginas dinâmicas, denominadas de DHTML. Suporta também folha do estilo cascata (CSS) e comportamento de Javascript.

Para acessar a página web de interface do controlador, deve-se conectar qualquer computador ao dispositivo. O controlador poderá ser configurado e gerar um endereço de IP fixo ou buscar ainda um endereço através de um servidor através do modo de DHCP. Uma vez gerado, o endereço IP será mostrado no display do controlador e se este estiver com o cabo ethernet conectado, pode-se digitar o número do endereço diretamente no navegador e logo a página será carregada.

Esta página Web terá a função de parametrizar o controlador através do controlador ethernet, do microcontrolador Stellaris Lm3s6569 e das pilhas TCP/IP. Utilizando formulários comuns para transmitir dados para os scripts CGI que são executados internamente no microcontrolador.

Nesta pagina Web é possível habilitar um saída PWM para controle de motores, onde pode-se configurar os parâmetros de frequência e do *Duty Cycle* que constituem um sinal PWM. Também é possível selecionar as saídas digitais que serão acionadas e configurá-las para acionarem com sinal constante ou intermitente. Caso a opção de acionamento intermitente for selecionada, será possível configurar os tempos em que as saídas permanecem ligadas e também desligadas.

No modo de acionamento intermitente, que pode ser selecionado na página Web, pode-se realizar a combinação das saídas digitais disponíveis, conseguindo assim obter um sequenciador de saídas digitais, com tempos de acionamentos controlados.

Todas estas funções serão executadas pelo microcontrolador sempre que o botão de envio de dados do formulário for pressionado, pois somente quando forem transmitidas as informações com solicitações desejadas é que os scripts CGI serão executados. Fazendo assim com que o controlador trabalhe como o esperado.

Além de realizar a configuração do controlador, será possível monitorar sempre que for solicitada uma atualização. As informações sobre os acionamentos, como se o sinal PWM está ativo ou quais saídas digitais estão acionadas, e os valores de frequência e *duty cycle*.

Trabalho de Conclusão de Curso
Controlador Parametrizável Via Rede Ethernet
Kit de desenvolvimento Stellaris® LM3s6965

This demonstration shows another method of performing control and status reporting. This time, we use Server Side Include tags to replace text in the page as it is being served from the Stellaris board and standard HTML forms to send data to a CGI handler running on the board. This example does cause the page to be reloaded whenever form data is submitted but it involves less complex HTML to perform the task.

Control	Current	New
LED0 State	OFF	<input type="checkbox"/>
LED1 State	OFF	<input type="checkbox"/>
LED2 State	OFF	<input type="checkbox"/>
LED3 State	OFF	<input type="checkbox"/>
LED4 State	OFF	<input type="checkbox"/>
SEQ01 State	OFF	<input type="checkbox"/>
PWM State	OFF	<input type="checkbox"/>
PWM Frequency (Hz)	440	<input type="text" value="440"/>
PWM Duty Cycle (%)	50	<input type="text" value="50"/>
TIMER ON (s)	10	<input type="text" value="10"/>
TIMER OFF (s)	10	<input type="text" value="10"/>

Display this text on the screen:

Copyright © 2009 Texas Instruments Incorporated. All rights reserved.

Fonte: Elaborado pelo autor.

Figura 18: Interface de operação e parametrização.

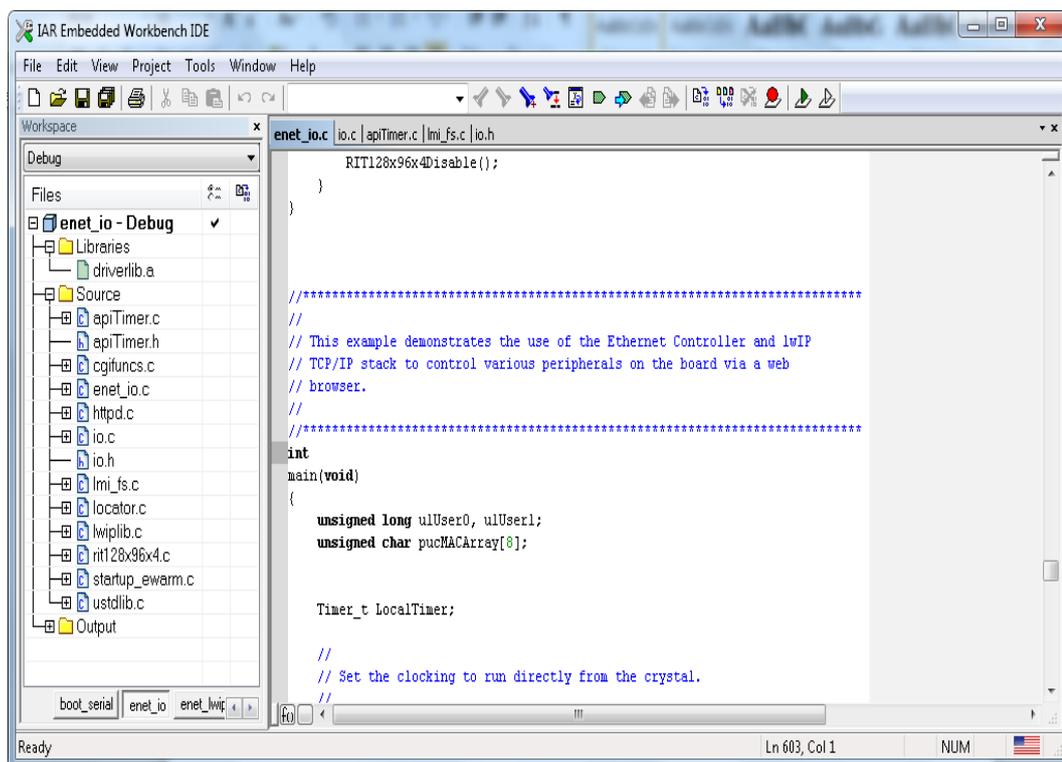
Para apresentação dos dados e status do controlador, como nível lógico das saídas digitais ou valores de tempos de acionamento, foram utilizadas variáveis comandadas por tags SSI. Estes tags mostrarão textos que serão controlados pelo servidor que receberá informações do microcontrolador stellaris Lm3s6569 na placa do kit de desenvolvimento. Assim como nos scripts CGI, a aplicação irá registrar uma lista de tags SSI e manipulará

certas funções durante a inicialização do servidor, que apresentará um conteúdo dinâmico quando acionado.

3.3. INTERFACE DE PROGRAMAÇÃO, SIMULAÇÃO E GRAVAÇÃO

Para o desenvolvimento do software que atuará no sistema microcontrolado, utilizou-se a ferramenta de programação chamada de IAR *Embedded Workbench*. Mantida pela empresa IAR Systems, especialista em software para sistemas embarcados.

Neste trabalho, foi utilizada uma versão gratuita desta ferramenta com limitação no tamanho do código do programa de aplicação. Através desta ferramenta, é possível construir aplicações em um ambiente de desenvolvimento completo, integrando um gestor de projetos, editor e ferramentas de compilação, onde é possível desenvolver, compilar e depurar a aplicação em um simulador ou através de um hardware apropriado. Todas estas funcionalidades estão disponíveis para vários microcontroladores. Até mesmo os de 32 bits, entre eles o Stellaris Lm3s6569, utilizado neste trabalho.



Fonte: Elaborado pelo autor.

Figura 19: Tela do software IAR Embedded Workbench.

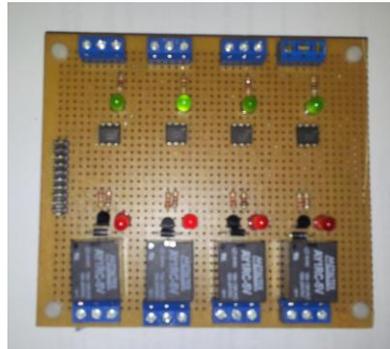
Neste compilador, os programas de aplicação são desenvolvidos nas linguagens C e C++, por possuir otimizações globais genéricas e otimizações específicas de cada chip. Dessa forma, durante o desenvolvimento pode-se aproveitar melhor as características de cada microcontrolador e gerar um código reduzido.

3.4. HARDWARE DE EXPANSÃO DE ENTRADAS E SAÍDAS

Para auxiliar na demonstração do conceito que este trabalho pretende apresentar, foi montada uma placa com o objetivo de se obter uma expansão para os sinais de entradas e saídas digitais.

Como o kit de desenvolvimento Stellaris LM3S6569 não contempla nenhum acionamento pronto, por isso a importância de se utilizar este hardware preparado para receber sinais de entradas e saídas e por sua vez realizar os devidos acionamentos.

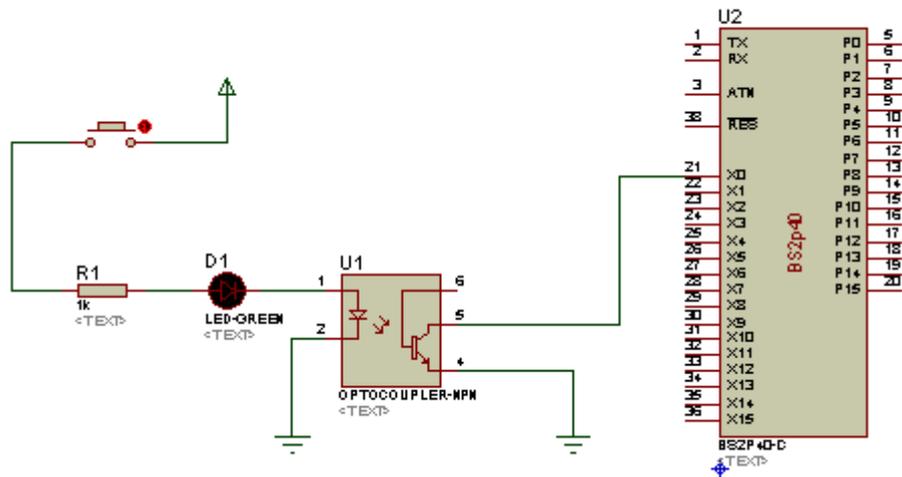
Esta placa de expansão será conectada ao kit de desenvolvimento através um flat cable, que levará até a placa de expansão todos os sinais necessários, incluindo alimentação 5 VDC e GND.



Fonte: Elaborado pelo autor.

Figura 20: Placa de expansão e entradas e saídas digitais.

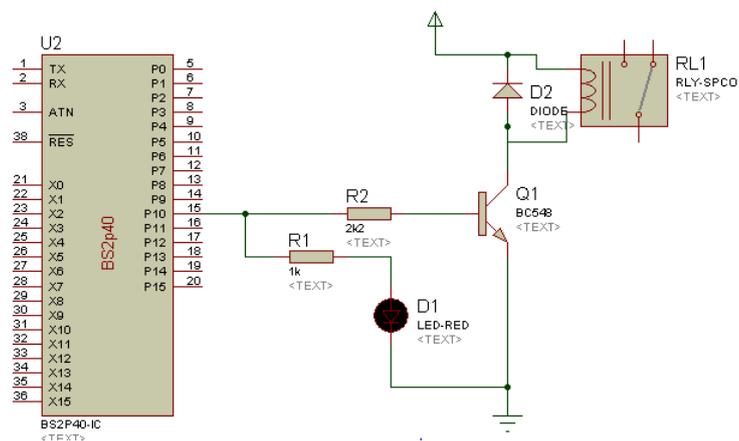
O circuito de entrada da placa de expansão foi baseado no conceito dos Clp's atuais e é constituído de quatro circuitos iguais onde os sinais externos podem ser fixados através de bornes. Estes circuitos também são contemplados com leds de indicação e opto acopladores para realizar a isolação entre o microcontrolador e o meio externo. Os pinos do micro controlador foram configurados para a função Pull-Up, neste caso compreende-se que a entrada estará ativa quando a mesma for aterrada.



Fonte: Elaborado pelo autor.

Figura 21: Circuito de entrada.

Para os circuitos de saída, também foram utilizados quatro circuitos iguais como nas entradas, porém para o circuito de saída utilizou-se acionamentos a relés, permitindo acionar dispositivos com potências mais elevadas. Além disso, os dispositivos de saídas também estão contemplados com leds de indicação e podem ser fixados por bornes.



Fonte: Elaborado pelo autor.

Figura 22: Circuito de saída.

4. RESULTADOS

Através da finalização do trabalho, foram obtidos resultados muito satisfatórios, além de terem sido uma forma prática para adquirir uma maior compreensão dos recursos fornecidos pelos modernos microcontroladores de 32 bits que podem utilizar pilhas TCP/IP, o que representa um grande diferencial para dispositivos e novos produtos.

Através do conceito proposto neste trabalho, que se baseou na ferramenta do Kit desenvolvimento Stellaris Lm3s6569 do fabricante Texas Instruments, citado anteriormente, nota-se que apesar de ser apenas um conceito, pode-se calcular previamente uma redução de custo relativa, de pelo menos 20 por cento, se comparado com outras soluções de automação controladas por CLP. Algumas aplicações foram alcançadas, tais como, monitoração e acionamentos de relés, sinais digitais e acionamento e controle de um sinal PWM.

Também foi constatada a eficiência da interface de parametrização, onde a utilização do recurso de formulários HTML, que foi responsável por executar todos os acionamentos e configurações, além de apresentar os status das variáveis do dispositivo. Foi observado que tanto a parametrização quanto os acionamentos foram executados conforme o desejado, com grande rapidez por estar utilizando o protocolo ethernet.

Além dos resultados alcançados neste trabalho, também se encontra uma justificativa para que cada vez mais fabricantes de equipamentos estejam utilizando em seus mais variados produtos as tecnologias dos protocolos ethernet e TCP/IP, pois permitem conexão direta com qualquer computador pessoal ou rede local e proporcionam inúmeras vantagens, tais como baixo custo e elevado desempenho.

5. CONCLUSÃO

Devido à necessidade cada vez mais latente de se automatizar e aumentar os processos produtivos, com um mercado cada vez mais competitivo, a redução de custos é um fator primordial na conquista de novos mercados. Consequentemente a essas circunstâncias, entram em cena novas tecnologias que vem sendo implementadas com produtos inovadores e cada vez mais acessíveis às empresas e às pessoas. Com base nessas novas tendências, este trabalho contribuiu significativamente para propor uma solução prática, de baixo custo e fácil de ser implantada, porém, capaz de atender com eficiência uma demanda do mercado atual para automatizações de pequeno porte.

Além de ter proposto uma solução utilizando um conceito de baixo custo, este trabalho apresentou uma interface de fácil acesso, onde o operador pode de forma intuitiva, configurar o controlador para atuar conforme o desejado. Uma vez configurado, o controlador poderá executar suas funções constantemente até que haja uma nova intervenção, além de ser capaz de trabalhar separadamente de um computador.

5.1. DIFICULDADES ENCONTRADAS

No decorrer do trabalho, surgiram várias dificuldades que impediam alcançar o objetivo final. Dentre essas dificuldades, a principal foi a falta de suporte e de material de apoio no Brasil do fabricante Texas Instruments. Dessa forma, durante o desenvolvimento do trabalho, à medida que surgiam as dúvidas sobre o kit de desenvolvimento deste fabricante, foi consumido um tempo muito maior do que o esperado para solucioná-las. Para sanar as dúvidas ou corrigir eventuais erros, foi preciso executar várias pesquisas em muitos sites, estas interrupções dificultaram muito o andamento do trabalho.

O programa de aplicação no microcontrolador foi desenvolvido na linguagem C, devido ao nível de complexidade do programa de aplicação e isto representou outra dificuldade. Normalmente, em cursos de graduação dificilmente se adquire o domínio em linguagem C necessário para aplicações mais avançadas, como a do presente trabalho. Devido a esse obstáculo, para o desenvolvimento do software do microcontrolador foi preciso bastante empenho e esforço nas pesquisas.

5.2. PROJETOS FUTUROS

Mesmo atingindo o objetivo proposto neste trabalho, existem muitas oportunidades para o aprimoramento do projeto. Uma melhoria muito relevante seria a implementação de um módulo WiFi, pois o microcontrolador utilizado permite que este tipo de módulo, dentre outros possa ser implantado. Assim, com a implementação do módulo, será possível se conectar com o dispositivo a uma determinada distância e sem a necessidade de cabos.

Outro grande aprimoramento seria a inclusão de um módulo GPRS (Serviço de Rádio de Pacote Geral), pois este módulo permite o transporte de dados por comutação de pacote e pode ser utilizado para enviar e receber dados em uma rede de celular GSM. Dessa maneira, com acesso a internet, será possível acessar o dispositivo a distância e obter informações e parametrizar suas funções através de um servidor na Web.

6. REFERÊNCIAS

- [1] **Automação e controle discreto** / Paulo Rogério da Silveira, Winderson E. dos Santos. – São Paulo: Érica, 1998.
- [2] **Automação Aplicada Descrição e Implementação de Sistemas Sequenciais com PLCs** / Marcelo Georgini. São Paulo: Érica, 2003.
- [3] **Redes de Computadores Curso Completo** / Gabriel Torres – Rio de Janeiro: Axcel Books do Brasil, 2001.
- [4] **Rede e sistemas de comunicação de dados: teoria e aplicações corporativas** / Willian Stallings. – Rio de Janeiro: Elsevier, 2005.
- [5] **Microcontroladores 8051: Teoria do hardware e do software** / Gimenez, Salvador Pinillos. – São Paulo: Peason Education do Brasil Ltda, 2002.
- [6] **Desbravando o PIC: ampliado e atualizado para PIC 16F628A** / David José de Souza. – 12. Ed. - São Paulo: Érica, 2008.
- [7] **TEXAS INSTRUMENTS INCORPORATED. Stellaris LM3S6965 Microcontroller Data Sheet.** Rev H. Austin, TX Jun. 2012. **Disponível em:** <<http://www.ti.com/stellaris>>. **Acesso em:** 15 Jul. 2012.
- [8] **TEXAS INSTRUMENTS INCORPORATED. Stellaris LM3S6965 Evaluation Board User's Manual.** Rev H. Austin, TX Jan. 2010. **Disponível em:** <<http://www.ti.com/stellaris>>. **Acesso em:** 18 Jul. 2012.
- [9] **Iniciando em HTML** / José Antônio Ramalho. São Paulo: Makron Books, 1996.
- [10] **Usando HTML** / Tom Savola; Alan Westenbroek; Joseph Heck; Tradução de ARX Publicações. Rio de Janeiro: Campus, 1997.
- [11] WEINMAN, Lynda; WEINMAN, Willian. **Creative HTML Design.** Indianapolis: New Riders, 1998.
- [12] TITTEL, Ed et al. **CGI Bible:** Foster City CA: IDG Books Worldwide, 1996.

[13] OS METODOS GET E POST. **Disponível em:** <<http://www.htmlstaff.org/ver.php?id=1664>>. **Acesso em:** 20 Out. 2012.

[14] PROGRAMANDO COM HTML. **Disponível em:** <<http://minerva.ufpel.edu.br/~bira/html/formularios.html>>. **Acesso em:** 25 Out. 2012.

[15] SSI. **Disponível em:** <<http://www.hotserver.com.br/ajuda/ssiwin.htm>>. **Acesso em:** 28 Out. 2012.