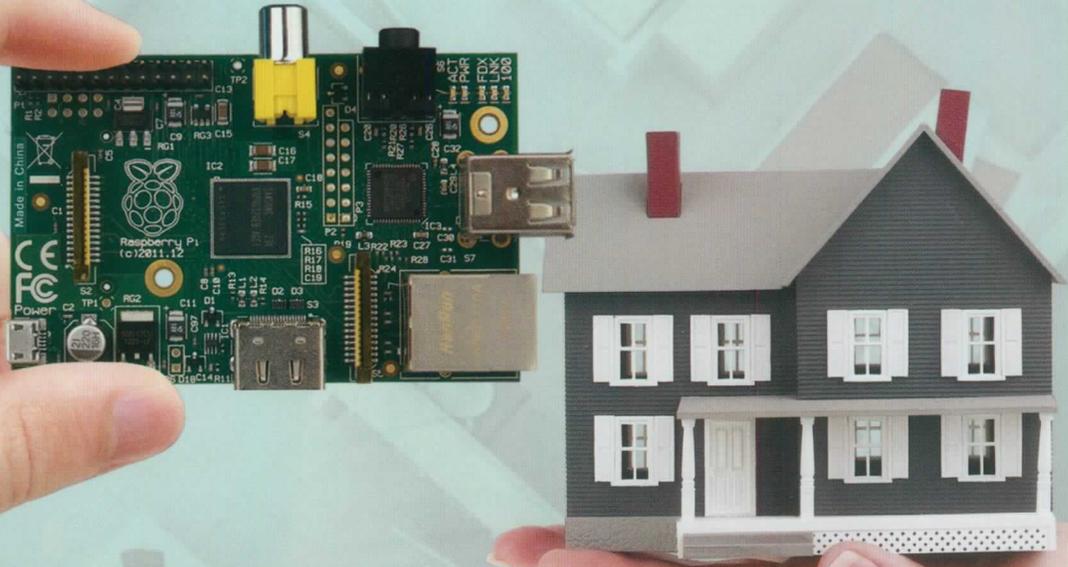


ELETRÔNICA TOTAL



Ano 24 - Nº 158
Europa € 4,30 Brasil R\$ 12,90



Raspberry Pi: Controle sua casa com comando de VOZ

Automatize a climatização, iluminação e entretenimento,
utilizando reconhecimento de voz e infravermelho



ISSN 0103-4960

Braço Robótico

Com Arduino
e LabVIEW

Inglês para Eletrônicos

Como escrever
data-sheets

Ensinando FFT

Na prática com
o LabVIEW

Detector de passagem

Configurações mais
comuns

Mais de 4 milhões de componentes
eletrônicos oferecidos.





Assistência em Português

GO br.mouser.com

Os mais novos produtos para os projetos mais inovadores®

Acesso instantâneo.

Mais de 4 milhões de componentes eletrônicos oferecidos por mais de 500 fabricantes líderes reconhecidos da indústria.



Mouser é sua fonte autorizada para os mais novos semicondutores e componentes eletrônicos para os seus projetos mais inovadores



Distribuímos semicondutores e componentes eletrônicos para Engenheiros de Desenvolvimento e Projetistas



MOUSER
ELECTRONICS



Editora Saber Ltda.

Diretor
Hélio Fittipaldi

ELETRÔNICA
TOTAL

www.eletronicatotal.com.br

twitter.com/editora_saber

Editor e Diretor Responsável
Hélio Fittipaldi

Conselho Editorial
Luiz Henrique C. Bernardes;
Newton C. Braga

Revisão Técnica
Eutíquio Lopez

Designers
Carlos C. Tartagliani;
Diego M. Gomes

Redação
Rafaela Turiani

Publicidade
Caroline Ferreira;
Marileide de Oliveira

Colaboradores
Caio Carneiro Silva Rocha;
Eduard Montgomery Meira Costa;
Eduardo Guimarães Zola;
Guilherme Kenji Yamamoto;
Hector Alfonso Petez Mateus;
Maurício Petterson Gulicz;
Newton C. Braga;
Renan Airosa Machado de Azevedo;
Rafaela Turiani;
Roberto Mathias Susin

PARA ANUNCIAR: (11) 2095-5339
publicidade@editorasaber.com.br

Capa
Arquivo Editora Saber; Stock.XCHNG

Impressão
EGB

Distribuição
Brasil: DINAP
Portugal: Logista tel.: 121 926-7800

ASSINATURAS

www.eletronicatotal.com.br
fone (11) 2095-5335/fax (11) 2098-3366
atendimento das 8:30 às 17:30h

Eletrônica Total é uma publicação bimestral da Editora Saber Ltda, ISSN 0103-4960. Redação, administração, publicidade e correspondência: Rua Jacinto José de Araújo, 315, Tatuapé, CEP 03087-020, São Paulo, SP, tel./ fax (11) 2095-5333. Edições anteriores (mediante disponibilidade de estoque), solicite pelo site www.sabermarketing.com.br, ou pelo tel. 2095-5330, ao preço da última edição em banca.

Associada da:

anatec

www.anatec.org.br

Associação Nacional das Editoras de Publicações Técnicas, Dirigidas e Especializadas

Editorial

Você, que acompanha nosso trabalho através das edições impressas nestes 24 anos de existência da revista, pode encontrar na internet os artigos publicados desde 2005 em www.sabereletrônica.com.br. Na seção Educação estão os artigos da Eletrônica Total, e se você se cadastrar, poderá ver até 30% de todo o portal gratuitamente.

Para ter acesso a tudo, é só fazer a assinatura do portal Saber Eletrônica por R\$48,00 por ano, podendo ver inclusive os artigos da Saber Eletrônica sem pagar mais nada.

Por estarem na nuvem, como muitos dizem, você pode acessá-los de qualquer lugar apenas com seu login e senha. É muito prático e barato, mas se mesmo assim o seu interesse for o de acessar tudo por apenas uma semana, temos um plano especial de assinatura por R\$15,00.

Estamos trabalhando para que você tenha à disposição da maneira mais rápida possível, o conteúdo das nossas revistas. Dentro em breve teremos também todo o nosso material feito exclusivamente para ser acessado do seu smartphone. Muita gente está trocando o antigo celular pelos modernos smartphones, podendo assim acessar a internet.

Nosso novo programa detecta quando é acessado, se é através do smartphone e, automaticamente, disponibiliza o formato ideal no tamanho da sua tela. Isto já acontece atualmente, quando você nos acessa através do tablet, notebook ou desktop.

Sabemos que nos dias de hoje temos de oferecer os diversos formatos, pois os leitores têm hábitos diferentes e é preciso respeitá-los. Muitas revistas de diversos segmentos no mundo têm deixado de circular justamente por não entenderem isso, e não se adaptarem aos novos comportamentos impostos pela tecnologia.



Hélio Fittipaldi

Hélio Fittipaldi

Submissões de Artigos

Artigos de nossos leitores, parceiros e especialistas do setor serão bem-vindos em nossa revista. Vamos analisar cada apresentação e determinar a sua aptidão para a publicação na Revista Eletrônica Total. Lemos trabalhar com afinco em cada etapa do processo de submissão para assegurar um fluxo de trabalho flexível e a melhor apresentação dos artigos aceitos em versões impressa e online.

Índice

- 10 Controle sua Casa com o Raspberry Pi
- 14 Braço Robótico com Arduino e LabVIEW
- 19 Sensores e Amplificadores Operacionais
- 22 Algoritmo de Mapeamento de Labirintos
- 29 Controles Automáticos de Temperatura
- 36 Inglês para Eletrônicos
- 38 Ensinando FFT na Prática com o NI LabVIEW
- 41 Filtro de Linha
- 43 Switches On-Off com o Arduino
- 52 Usando o Osciloscópio no Service de TV e Vídeo
- 56 Sistema de Medição de Som e Vibração
- 58 Entenda de fato o que é um Sistema de Tempo Real
- 60 Detectores de Passagem

Atendimento ao Leitor: atendimento@eletronicatotal.com.br

Os artigos assinados são de exclusiva responsabilidade de seus autores. É vedada a reprodução total ou parcial dos textos e ilustrações desta Revista, bem como a industrialização e/ou comercialização dos aparelhos ou ideias oriundas dos textos mencionados, sob pena de sanções legais. As consultas técnicas referentes aos artigos da Revista deverão ser feitas exclusivamente por cartas, ou e-mail (A/C do Departamento Técnico). São tomados todos os cuidados razoáveis na preparação do conteúdo desta Revista, mas não assumimos a responsabilidade legal por eventuais erros, principalmente nas montagens, pois tratam-se de projetos experimentais. Tampouco assumimos a responsabilidade por danos resultantes de imperícia do montador. Caso haja enganos em texto ou desenho, será publicada errata na primeira oportunidade. Preços e dados publicados em anúncios são por nós aceitos de boa fé, como corretos na data do fechamento da edição. Não assumimos a responsabilidade por alterações nos preços e na disponibilidade dos produtos ocorridas após o fechamento.

CNPq abre Chamadas para PD&I em Fotovoltaica, LEDs e TI

O Ministério da Ciência, Tecnologia e Inovação (MCTI), por intermédio do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), torna públicas a Chamada MCTI/CNPq/CT-Energ n° 49/2013 - PD&I em Energia Solar Fotovoltaica e LEDs para Aplicações em Iluminação e Eficiência Energética, e a Chamada MCTI/CT-Info/CNPq n° 59/2013.

O objetivo principal das chamadas é o apoio financeiro não reembolsável para projetos de pesquisa científica, tecnológica e de inovação, respectivamente, em tecnologias para geração de energia solar fotovoltaica, LEDs e fontes luminosas inovadoras e para

sistemas de iluminação de alta eficiência; e a atração de Centros Globais de P&D em software e serviços de TIC, no âmbito do Programa Estratégico de Software e Serviços de Tecnologia de Informação.

O proponente deverá possuir ao menos o título de mestre, ser obrigatoriamente o coordenador do projeto, além de ter vínculo formal com a instituição de execução do projeto. As propostas aprovadas serão financiadas com recursos no valor global estimado de R\$ 8 milhões do CT-Energ, sendo que cada projeto terá o valor máximo financiado de R\$ 400 mil, com prazo de execução de até 24 meses para a Cha-

mada em Energia Solar Fotovoltaica e LEDs. E, para a Chamada em TI, o valor global estimado é de R\$ 14 milhões do FNDCT/CT-Info, sendo que o valor máximo financiado em cada projeto é de R\$ 2 milhões, que serão destinados integralmente ao financiamento de bolsas, com prazo de execução de até 36 meses.

Os interessados em participar devem submeter suas propostas até 4 de novembro, para ambas as chamadas.

Acesse a Chamada Pública em Energia Solar Fotovoltaica e LEDs em Chamadas Públicas no site: www.cnpq.br.

MediaTek desenvolve o primeiro cartão 3-SIM para Smartphones 3G

A MediaTek Inc., uma empresa de semicondutores para comunicações sem fio e soluções de multimídia digital, anunciou a primeira tecnologia 3-SIM para smartphones Android 3G. A solução de ponta, que é uma resposta à demanda generalizada de dispositivos multi-SIM em mercados emergentes, foi recentemente comercializada no Brasil pelo smartphone Optimus L4II, da LG Electronics.

Aproveitando o sucesso de sua popular antecessor dual-SIM, a tecnologia 3G 3-SIM permite que três cartões SIM sejam usados simultaneamente. Cada um deles pode receber chamadas, transmitir dados, enviar mensagens SMS e permanecer no modo *stand-by* simultaneamente sem consumir bateria em excesso.

Uma característica fundamental da solução 3G 3-SIM é seu alto nível de estabilidade, que evita a

ocorrência de queda de chamadas ou de má qualidade, e reafirma a reputação da MediaTek como o principal produtor de tecnologias de chipset multi-SIM.

"Há muito tempo a MediaTek fornece soluções de ponta para dispositivos móveis multi-SIM. O 3G 3-SIM é mais um exemplo disso", diz Brian Kwon, chefe de grupo de planejamento de produtos da LG Electronics. "Graças à sua tecnologia inovadora, os novos smartphones vão ampliar significativamente as opções de conectividade de nossos clientes, e com isso eles poderão aproveitar vários planos tarifários".

Além de sua tecnologia pioneira, a tecnologia 3G 3-SIM também serve como uma solução integral para os fabricantes através de sua interface embutida de usuário, que permite que os usuários finais ajustem ou

desliguem diretamente a função multi-SIM, se e quando desejado.

"Em muitos mercados emergentes, o suporte multi-SIM tornou-se uma exigência absoluta. Com a criação de uma solução completa abrangente, repleta de funcionalidades superiores e tecnologias avançadas, a MediaTek atende não só às necessidades do usuário final, mas também atrai a atenção de fabricantes de aparelhos de primeira linha do mundo", diz Jeffrey Ju, gerente-geral da Unidade de Negócios de Smartphones da empresa. "Com a nossa longa linha de novos produtos instigantes em desenvolvimento, esperamos trazer muitas soluções de dispositivos móveis superiores a preços mais acessíveis para os mercados emergentes e outros mercados, em um futuro próximo."

CNI e Sebrae abrem inscrições para Prêmio Nacional de Inovação 2013

Estão abertas as inscrições para o Prêmio Nacional de Inovação 2013 até o dia 31 de outubro. Empresas de micro, pequeno, médio e grande porte de todo o Brasil poderão concorrer a R\$ 900 mil pré-aprovados no Edital SENAI SESI de Inovação, além de missão técnica internacional e cursos de educação executiva. As inscrições devem ser feitas exclusivamente pela internet, onde o regulamento está disponível. Os vencedores serão conhecidos em maio do ano que vem.

O Prêmio Nacional de Inovação é uma iniciativa da Mobilização Empresarial pela Inovação (MEI), realizado

pela CNI e o Sebrae – Serviço de Apoio às Micro e Pequenas Empresas, com apoio do Instituto Euvaldo Lodi (IEL), Serviço Nacional de Aprendizagem Industrial (SENAI), Movimento Brasil Competitivo (MBC), Financiadora de Estudos e Projetos (FINEP) e Ministério de Ciência, Tecnologia e Inovação (MCTI).

As empresas podem concorrer em quatro categorias:

- **Gestão da Inovação:** busca reconhecer métodos, técnicas e ferramentas de gestão da inovação que estabeleçam um ambiente propício ao desenvolvimento de projetos inovadores.

- **Agente Local de Inovação:** direcionada a micro e pequenas empresas atendidas pelo Programa Agentes Locais de Inovação (ALI) do Sebrae nos setores de indústria, comércio e serviços.

- **Inovação Tecnológica:** valoriza projetos de produtos (ou processos) que contribuem para o aumento dos níveis de competitividade da empresa.

- **Inovação em Modelo de Negócio:** premia a forma como projetos são levados ao mercado a fim de aprimoramento de modelos de negócio capazes de gerar impactos na empresa.

Tecnologia garante economia de até 35% no consumo de energia elétrica

A cada dia que passa, as pessoas ficam mais dependentes da energia elétrica e dos milhares de aparelhos que dela dependem para funcionar, mas se esquecem de calcular os prejuízos que o uso desenfreado da tecnologia causa tanto para o bolso quanto para o meio ambiente.

Aos que prezam por minimizar os impactos negativos do consumo de energia como apagar a luz quando não estiverem no cômodo, escolher eletrodomésticos de baixo consumo energético para aquisição, tirar da tomada os aparelhos que não estiverem em

uso, a On Eletrônicos apresenta uma solução inovadora, prática e rentável: o *Economicus*.

Trata-se de um pequeno dispositivo capaz de economizar até 35% do consumo de energia. Isso acontece porque ele funciona como um dosador que fornece somente a carga necessária para o funcionamento do aparelho, evitando desperdícios.

Além da economia, proporciona uma série de vantagens ao consumidor: torna a partida do motor mais suave, reduz a temperatura, a vibração e o ruído, o que aumenta a vida útil de

refrigeradores, freezers, hidromassagens etc.

Para fazê-lo funcionar, basta ligá-lo ao aparelho e, em seguida, à tomada. Depois disso, basta esperar a conta de energia elétrica no final do mês para comprovar a economia!

Características:

- Indicado para geladeiras, freezers, bombas de piscinas etc;
- Tensão: Bivolt automático;
- Corrente: 6 A;
- Refrigeradores até 550 litros;
- Motores de indução até 1 CV.

ERRATA / ELETRÔNICA TOTAL 157

Página 44: Na figura 14, onde está escrito "Voltage AC", leia-se "Voltaje AC"; Onde está escrito "Resistencia", leia-se "Resistencia" e na mesma linha, substitua "200 MW" por "200 MΩ"; Onde está escrito "Protección", leia-se "Protección".

Página 44: Na figura 15, onde foi desenhado o símbolo de um resistor, o certo é o símbolo de um capacitor.

Página 45: Na legenda da figura 17, leia-se "Estrutura interna de um JFET – Canal N". Na legenda da figura 18, leia-se "Vista de cima do JFET - Canal N".

Pen drive para smartphones e tablets com plataformas Android, Mac e PC

A Leef Technology acaba de anunciar o produto Leef Bridge™, o primeiro *flash drive USB* para compartilhamento de arquivos entre smartphones e tablets com plataformas Android, Mac e PC. O Leef Bridge é um “pen drive” simples e inovador que permite aos usuários transferir facilmente arquivos para outros dispositivos sem necessidade de cabos, serviços de computação em nuvem, Wi-Fi ou qualquer tipo de conexão de dados.

Através de conexão USB, o ele permite que os usuários compartilhem conteúdo, fotos, vídeos, músicas e documentos de/para qualquer dispositivo compatível. A memória utilizada dentro do Leef Bridge possui velocidades de leitura e escrita bastante rápidas para garantir excelente reprodução de vídeos e músicas diretamente do *drive USB*, sem as interrupções ou atrasos que acompanham produtos com “flash drive USB” de baixa qualidade. Projetado na Califórnia pela equipe de *design* da Leef, o Bridge se destaca também pelo seu design exclusivo, apresentando uma bandeja *slide-and-lock* (desliza e trava), que permite ao usuário trocar facilmente para o conector USB apropriado para conexão de seus dispositivos.

“O Leef Bridge é uma importante realização técnica”, afirmou Jonny Anderson, sócio e diretor de design de produtos da Leef. “Como todos os nossos produtos, queremos que ele seja um objeto de desejo para as pessoas, algo que o usuário queira possuir por sua funcionalidade, design e estilo. Ele torna o gerenciamento de conteúdo em um telefone tão simples quanto conectar um *pen drive*. É realmente um produto revolucionário”, completa.

O Leef Bridge é atualmente compatível com dispositivos móveis com sistema operacional Android Jelly Bean OS 4.1 ou superior, e com conector microUSB On-The-Go (OTG). Também

roda em Windows® 8, Windows® 7, Windows® XP (SP3), Windows Vista® (SP1, SP2), Windows 2000®, Linux Kernel 2.6 ou superior, Apple® Mac OS X ou superior.

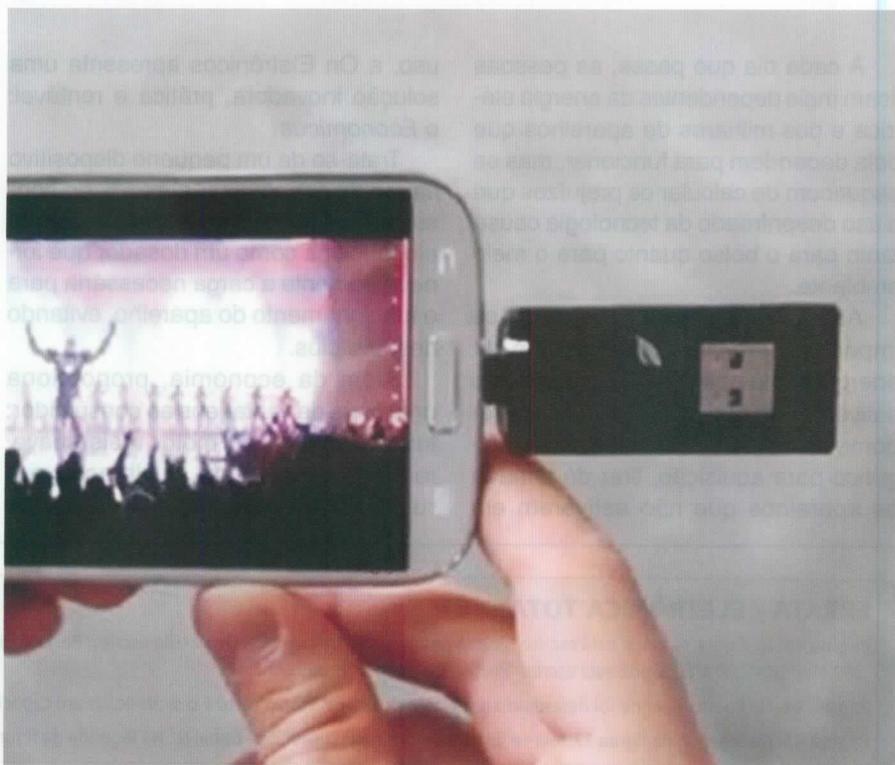
A Metago, desenvolvedora do ASTRO File Manager, o aplicativo de gerenciamento de conteúdo mais baixado no Android, fechou uma parceria com a Leef para lançar uma versão atualizada do seu *app*, permitindo uma experiência perfeita entre o Leef Bridge e ASTRO File Manager. As instruções para *download* do *app* do Google Play estão incluídas na embalagem do produto da Leef. Apesar do ASTRO File Manager não ser obrigatório para que o Leef Bridge funcione, a equipe de design da Leef o recomenda por seu sistema intuitivo de gerenciamento de arquivos.

O produto Leef Bridge é o mais recente lançamento da Leef, que ainda inclui outros modelos de pen drives de

alta qualidade, com uma grande gama de materiais e recursos de design inovadores.

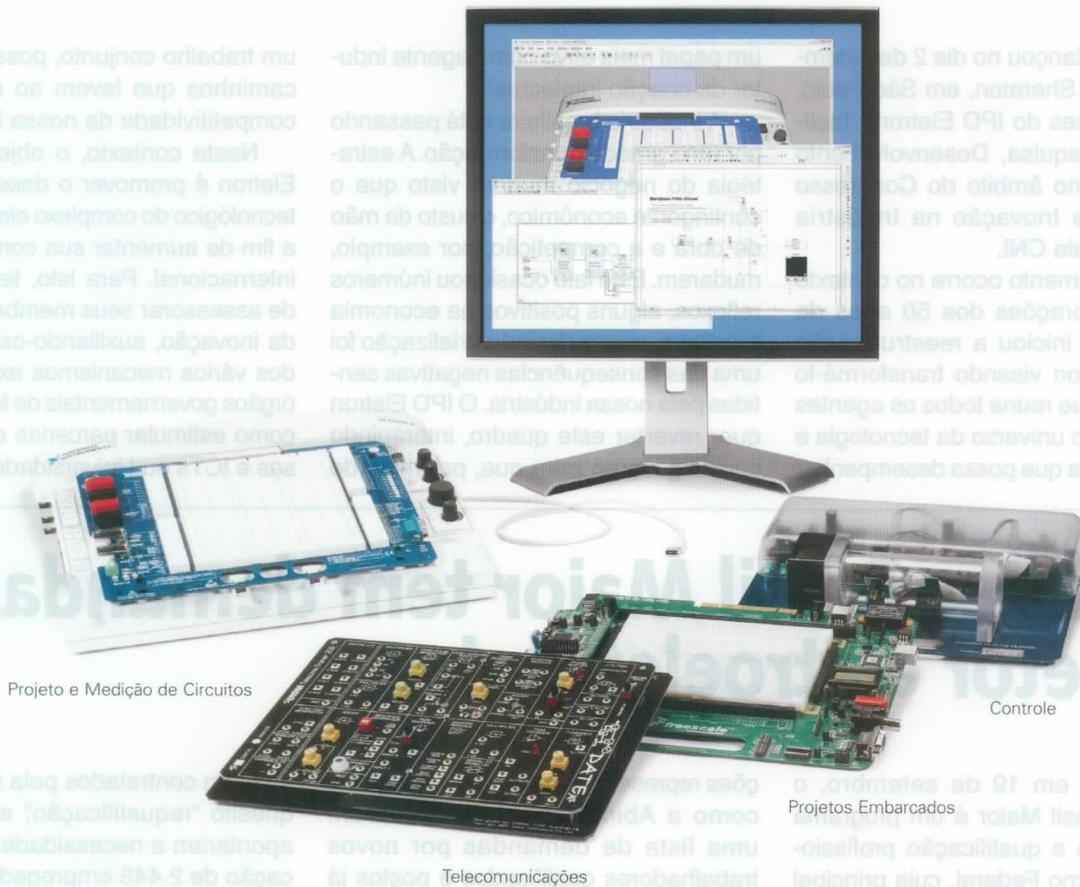
O produto ainda não está à venda no Brasil, mas a empresa estuda seu lançamento no mercado nacional.

Para o diretor executivo da Leef no país, João Marcelo Fernandes, os produtos Leef, com seus desenhos inovadores e únicos, podem promover uma verdadeira revolução na categoria de drives USB (pen drives) junto ao mercado nacional, agregando valor aos revendedores e dando a eles alternativas para sair do ponto comum. “Hoje, muitos varejistas buscam diferenciar seu portfólio de produtos, procurando itens mais exclusivos, com grande apelo de design, e principalmente para se diferenciar em relação a seus concorrentes diretos. É justamente este parceiro que estamos buscando no mercado brasileiro”, finaliza Fernandes.



Reduza os custos de seu laboratório

Plataforma flexível para projeto e protótipo NI ELVIS



Placas para NI ELVIS



O NI ELVIS é uma plataforma educacional que integra 12 dos instrumentos mais utilizados em um hardware compacto ideal para experimentos em sala de aula ou laboratórios.

Educadores ao redor do mundo podem estender seus laboratórios além da instrumentação e do projeto de circuitos, através das placas para NI ELVIS e material didático de líderes do segmento acadêmico como Emona Instruments, Freescale Semiconductors, Quanser e Labtools para ensino de telecomunicações, microcontroladores, controle e mais – tudo utilizando uma única e prática plataforma de projeto e protótipo.



inscreva-se
no Academic Day

NI Academic Day

Fórum de Inovação Tecnológica
para Pesquisa e Ensino

29/10 - Escola Politécnica da USP
Prédio do Biênio - 1º andar - São Paulo



- >> Faça o download dos recursos para ensino em ni.com/nielvis
- >> Inscreva-se no NI Academic Day em brasil.ni.com/eventos/academic-day



IPD Eletron - Instituto para Pesquisa, Desenvolvimento e Inovação

A Abinee lançou no dia 2 de setembro, no WTC Sheraton, em São Paulo, as novas bases do IPD Eletron - Instituto para Pesquisa, Desenvolvimento e Inovação, no âmbito do Congresso Brasileiro de Inovação na Indústria promovido pela CNI.

O relançamento ocorre no contexto das comemorações dos 50 anos da Abinee, que iniciou a reestruturação do IPD Eletron visando transformá-lo num fórum que reúna todos os agentes que atuam no universo da tecnologia e inovação, para que possa desempenhar

um papel mais ativo como agente indutor da criação intelectual.

A indústria brasileira está passando por uma grande transformação. A estratégia do negócio mudou, visto que o contingente econômico, o custo da mão de obra e a competição, por exemplo, mudaram. Este fato ocasionou inúmeros reflexos, alguns positivos na economia brasileira, mas a desindustrialização foi uma das consequências negativas sentidas pela nossa indústria. O IPD Eletron quer reverter este quadro, interagindo com o governo para que, por meio de

um trabalho conjunto, possa encontrar caminhos que levem ao aumento da competitividade da nossa indústria.

Neste contexto, o objetivo do IPD Eletron é promover o desenvolvimento tecnológico do complexo eletroeletrônico a fim de aumentar sua competitividade internacional. Para isto, tem a missão de assessorar seus membros na busca da inovação, auxiliando-os a fazer uso dos vários mecanismos existentes nos órgãos governamentais de fomento, bem como estimular parcerias entre empresas e ICTs ou Universidades.

Pronatec Brasil Maior tem demandas do setor eletroeletrônico

Lançado em 19 de setembro, o Pronatec Brasil Maior é um programa de formação e qualificação profissional do Governo Federal, cuja principal missão é a solução dos “gargalos” de recursos humanos em nível técnico, para setores estratégicos da economia nacional identificados pelo Plano Brasil Maior. A iniciativa visa oferecer, em uma primeira fase, 118 mil vagas em cursos de qualificação para ocupação de novas vagas e requalificação de trabalhadores em atividade de forma gratuita.

O programa, que atende às necessidades específicas de mão de obra apresentadas pelas empresas, é resultado da parceria entre o Ministério da Educação (MEC), o Ministério do Desenvolvimento, Indústria e Comércio Exterior (MDIC) e o Sistema S (Sesi, Sesc, Senai). Todos os cursos estão inseridos no Programa Nacional de Acesso ao Ensino Técnico e Emprego (Pronatec).

O mapeamento da demanda por qualificação profissional foi realizado pelo MDIC com o apoio das associa-

ções representativas do setor produtivo, como a Abinee, que apresentaram uma lista de demandas por novos trabalhadores qualificados e postos já preenchidos, mas com empregados que precisam de requalificação. No caso do setor eletroeletrônico, foram identificadas, na categoria qualificação, 836 vagas para potenciais candidatos

a serem contratados pela indústria. No quesito “requalificação”, as empresas apontaram a necessidade de requalificação de 2.448 empregados.

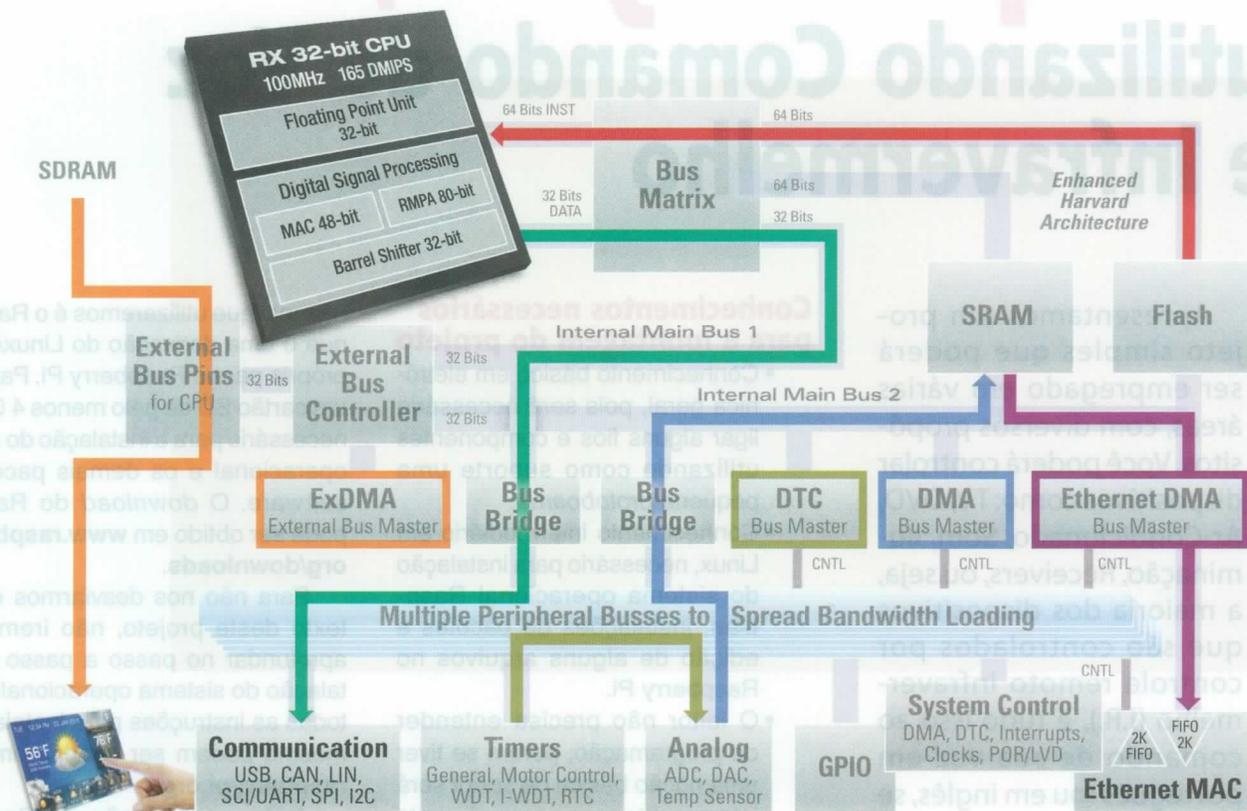
Os cursos abrangem diversas áreas, como Técnico em Eletrônica, Técnico em Mecatrônica, Técnico em Segurança do Trabalho, Montador de Equipamentos eletrônicos, entre outros.



MCU RX600 de Alta-Performance em 32 bits Para as Suas Mais Exigentes Aplicações de Conectividade



Dotado de múltiplos controladores de dados e da única Flash embarcada de 100MHz, o microcontrolador RX600 processa dados em alta velocidade, é capaz de lidar com todas as suas necessidades de conectividade, incluindo acionar diretamente displays TFT ("Direct Drive LCD"), Ethernet, USB e CAN.



Acionamento Direto de TFTs Duas USB CAN Ethernet

Experimente!

Registre-se para obter um kit de demonstração do RX62N GRATUITO.

- Conecta-se a uma porta USB do PC para exibir as características e capacidades das MCUs RX600
- Placa da MCU RX com "debugger" J-Link integrado e um enorme conjunto de periféricos, incluindo Ethernet, CAN e USB
- Display gráfico
- Entrada/Saída de Áudio
- Acelerômetro de 3 eixos
- O CD de instalação contém o IDE da Renesas (HEW), ferramentas, software demo da Micrium e muito mais



Nosso Representante de Vendas



Nossos Distribuidores



am.renesas.com/rx

© 2013 Renesas Electronics America Inc.

RENESAS

Controle sua casa com o Raspberry Pi, utilizando Comando de Voz e Infravermelho

Apresentamos um projeto simples que poderá ser empregado em várias áreas, com diversos propósitos. Você poderá controlar dispositivos como: TV, DVD, Ar-Condicionado, Som, Iluminação, Receivers, ou seja, a maioria dos dispositivos que são controlados por controle remoto infravermelho (I.R.), e tudo isso ao comando de sua voz em português (ou em inglês, se assim preferir).

Eduardo Guimarães Zola
www.zola.com.br

Conhecimentos necessários para a montagem do projeto

- Conhecimento básico em eletrônica geral, pois será necessário ligar alguns fios e componentes utilizando como suporte uma pequena *protoboard*.
- Conhecimento intermediário em Linux, necessário para instalação do sistema operacional Raspbian, instalações de pacotes e edição de alguns arquivos no Raspberry Pi.
- O leitor não precisa entender de programação, porém se tiver uma noção básica de *script* será interessante para o entendimento do código-fonte.

Tecnologias utilizadas

- Raspberry Pi;
- Google Speech Recognition API;
- LIRC (Linux Infrared Remote Control);
- Emissor e Receptor Infravermelho (I.R.).

Hardware e Sistema Operacional

Rotulado como o menor PC do mundo, o Raspberry Pi foi lançado em 2012 e desenvolvido na Grã-Bretanha. Inclui um processador ARM, interface de rede, entrada de cartão SD, HDMI, USB e saídas de áudio e vídeo.

Para a montagem do projeto, utilizaremos como plataforma o Raspberry Pi/ Modelo B, pois este modelo possui 512 MB de memória RAM. O sistema ope-

racional que utilizaremos é o Raspbian, que é uma derivação do Linux/Debian própria para o Raspberry Pi. Para tanto um cartão SD de pelo menos 4 GB será necessário para a instalação do sistema operacional e os demais pacotes de software. O *download* do Raspbian pode ser obtido em www.raspberrypi.org/downloads.

Para não nos desviarmos do contexto deste projeto, não iremos nos aprofundar no passo a passo da instalação do sistema operacional, porém todas as instruções para instalação do mesmo podem ser obtidas em www.raspberrypi.org.

Após a instalação do Raspbian, efetue o *login* utilizando o usuário "pi" que, por padrão, possui a senha "raspberrypi". Para facilitar a montagem do projeto e instalação de todos os pacotes, utilizaremos o usuário "root", pois assim evitaremos contratemplos no meio do caminho. Este, por padrão, é o usuário com maior privilégio em um ambiente Linux.

Após a instalação do sistema operacional, o Raspbian abrirá um menu. Sugiro que alguns passos sejam executados neste menu:

- Expand Filesystem: recomendo que esta opção seja selecionada para que haja um melhor aproveitamento no espaço do cartão SD.
- Change User Password: neste momento esta opção não é necessária. Desconsiderar.
- Enable Boot to Desktop: não recomendo esta opção para este

3V3	5V
SDA	NC
SCL	GND
GPIO 4	TX
NC	RX
GPIO 17	GPIO 18
GPIO 21	NC
GPIO 22	GPIO 23
NC	GPIO 24
MOSI	NC
MISO	GPIO 25
SCLK	CEO
NC	CE1

F1. Utilizando apenas 4 pinos do Raspberry Pi.

projeto, pois queremos entrar em modo linha de comando. Desconsiderar.

- Internationalisation Options: opcional, caso queira configurar teclado e acentuação.
- Enable Camera: Desconsiderar.
- Add to Rastrack: Desconsiderar.
- Overclock: Desconsiderar.
- Advanced Options: conecte o cabo de rede, estando com acesso a internet, selecione a opção A5 Update, para que o Raspbian possa ser atualizado. Este processo pode levar algum tempo, pois as atualizações serão "baixadas" via internet. Ao finalizar a atualização, clique em "Finish".

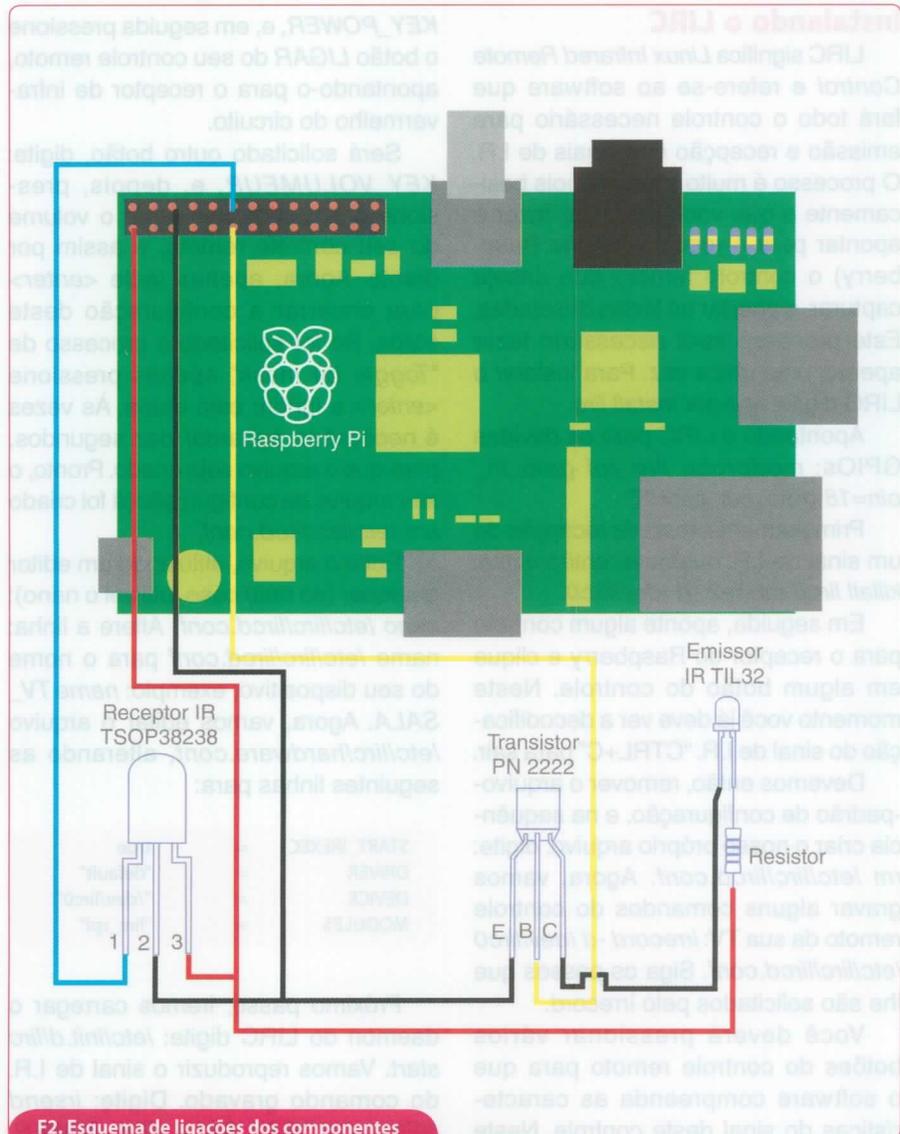
Agora vamos definir uma senha para o usuário root, então digite na linha de comando: `sudo passwd root`. Digite a senha duas vezes para confirmar. Na sequência, vamos efetuar o login com o usuário root, então digite: `su root`. Entre com a senha de root, cadastrada no passo anterior.

Esquema e Montagem

Iniciaremos agora a parte eletrônica do projeto. O esquema de montagem é muito simples, usaremos apenas quatro pinos do Raspberry Pi. Basta seguir o modelo, ligando os fios e componentes.

Os pinos utilizados na **figura 1** são:

- Pino 1: Saída de 3,3V
- Pino 6: GND
- Pino 11: GPIO 17 (servirá para escrita no emissor de I.R.)



F2. Esquema de ligações dos componentes do circuito ao Raspberry Pi.

- Pino 12: GPIO 18 (servirá para leitura do receptor de I.R.)

De acordo com cada cor, os fios da **figura 2** são utilizados da seguinte forma:

- Fio Vermelho: alimentação 3,3VDC.
- Fio Preto: Ground.
- Fio Azul: Receptor I.R.
- Fio Amarelo: Emissor I.R.

É importante lembrar que os componentes sejam posicionados na forma como mostra a figura, pois existe uma ordem correta na polaridade dos mesmos.

Para facilitar a montagem do projeto, sugiro ao leitor utilizar uma pequena *proto-board*. Agora conecte o circuito, conforme segue:

- Pino 1 do receptor I.R. no pino 12 do Raspberry (GPIO 18).
- Pino 2 do receptor I.R. no GND (ou pino 6) do Raspberry.
- Pino 3 do receptor I.R. no Positivo (ou pino 1) do Raspberry.
- Pino E do transistor no GND (ou pino 6) do Raspberry.
- Pino B do transistor no pino 11 do Raspberry (GPIO 17).
- Pino C do transistor no "perna" mais curta do LED emissor I.R.
- Conecte a "perna" mais comprida do LED emissor I.R. no resistor, e na sequência, conecte no positivo (ou pino 1) do Raspberry.

Observe o aspecto do circuito montado na **figura 3**.

Instalando o LIRC

LIRC significa *Linux Infrared Remote Control* e refere-se ao software que fará todo o controle necessário para emissão e recepção dos sinais de I.R. O processo é muito simples, pois basicamente o que você precisará fazer é apontar para o receptor IR (do Raspberry) o controle remoto que deseja capturar, e apertar as teclas desejadas. Este processo será necessário fazer apenas uma única vez. Para instalar o LIRC digite: `apt-get install lirc`.

Apontando o LIRC para as devidas GPIOs: `modprobe lirc_rpi gpio_in_pin=18 gpio_out_pin=17`.

Primeiramente, teste de recepção de um sinal de I.R. qualquer, então digite: `killall lircd mode2 -d /dev/lirc0`.

Em seguida, aponte algum controle para o receptor do Raspberry e clique em algum botão do controle. Neste momento você já deve ver a decodificação do sinal de I.R. "CTRL+C" para sair.

Devemos então, remover o arquivo-padrão de configuração, e na sequência criar o nosso próprio arquivo, digite: `rm /etc/lirc/lircd.conf`. Agora, vamos gravar alguns comandos do controle remoto da sua TV: `irrecord -d /dev/lirc0 /etc/lirc/lircd.conf`. Siga os passos que lhe são solicitados pelo `irrecord`.

Você deverá pressionar vários botões do controle remoto para que o software compreenda as características do sinal deste controle. Neste momento, será questionado o nome do botão que deseja configurar. Neste caso, como exemplo, podemos digitar:

`KEY_POWER`, e, em seguida pressione o botão *LIGAR* do seu controle remoto, apontando-o para o receptor de infravermelho do circuito.

Será solicitado outro botão, digite: `KEY_VOLUMEUP`, e, depois, pressione o botão de aumentar o volume do seu controle remoto, e assim por diante. Agora, apenas teclé <enter> para encerrar a configuração deste etapa. Se for solicitado o processo de "Toggle bit mask", apenas pressione <enter> e ignore esta etapa. Às vezes é necessário aguardar dez segundos, para que o arquivo seja criado. Pronto, o seu arquivo de configuração já foi criado em `/etc/lirc/lircd.conf`.

Edite o arquivo, utilizando um editor qualquer (no meu caso, utilizei o nano): `nano /etc/lirc/lircd.conf`. Altere a linha: `name /etc/lirc/lircd.conf` para o nome do seu dispositivo, exemplo: `name TV_SALA`. Agora, vamos editar o arquivo `/etc/lirc/hardware.conf`, alterando as seguintes linhas para:

```
START_IREXEC = true
DRIVER       = "default"
DEVICE       = "/dev/lirc0"
MODULES      = "lirc_rpi"
```

Próximo passo, iremos carregar o daemon do LIRC digite: `/etc/init.d/lirc start`. Vamos reproduzir o sinal de I.R. do comando gravado. Digite: `irsend SEND_ONCE TV_SALA KEY_POWER`. Note que, neste momento, a sua TV deve ligar. Caso não tenha ligado, verifique se o emissor de I.R. do Raspberry

está bem localizado em relação a sua TV. Outra dica é verificar se o LED emissor está acendendo. Como a luz I.R. é invisível ao olho humano, você pode utilizar a câmera de vídeo de um celular qualquer. Ao apontar a câmera para o LED, deverá ser possível ver a luz sendo emitida.

Configurando o IREXEC

O IREXEC, também é um *daemon* que faz parte do pacote do LIRC, porém este é responsável por mapear um sinal de I.R. a ser recebido de um comando a ser executado no Linux. Exemplo: Ao clicar no play do botão de um controle remoto, fazer com que o Raspberry reproduza um MP3, ou apareça uma mensagem na tela. Exemplo de conteúdo do arquivo `/etc/lirc/lircrc`:

```
begin
  remote = TV_SALA
  button = KEY_VOLUMEUP
  prog = irexec
  repeat = 0
  config = echo "CLICOU O BOTAO VOLUME UP"
end
```

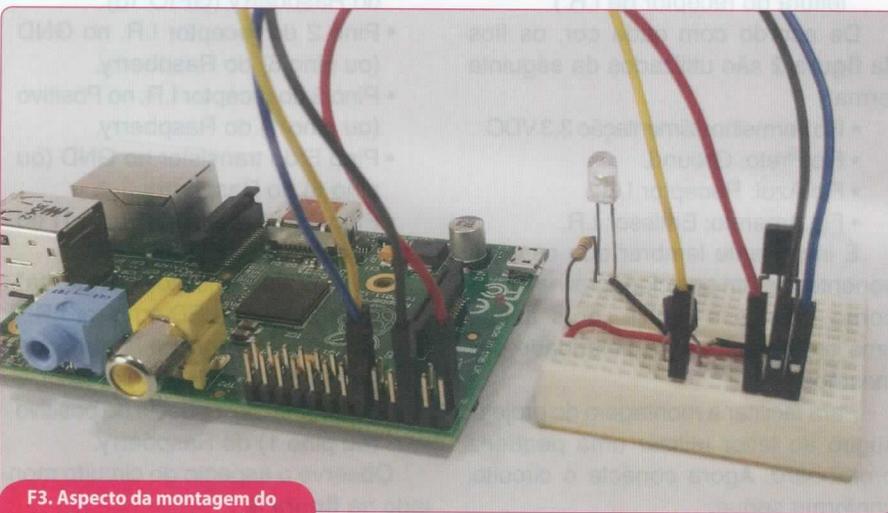
Salve o arquivo e reinicie o serviço do `lircd`, execute: `/etc/init.d/lirc restart`. Pronto, agora você será capaz de comandar o seu Raspberry através de um controle remoto I.R.

Observação: Você pode ter vários blocos de `begin/end` dentro do mesmo arquivo de configuração. Este arquivo de configuração geralmente fica em `/etc/lirc/lircrc`. Outro ponto é que o serviço do `lircd` deve ser reiniciado após qualquer alteração nos arquivos da pasta `/etc/lirc`. Para isso, execute: `/etc/init.d/lirc restart`.

Reconhecimento de voz

Utilizaremos uma técnica bem simples para o reconhecimento de voz, onde deixaremos todo o trabalho a ser feito para o Google. Criaremos um *script bash* que irá capturar sua voz, através de sua câmera (*webcam*), e em seguida enviá-la ao Google, e retornar um texto (*string*) das palavras reconhecidas.

O primeiro passo será instalar os utilitários necessários, para isso digite:



F3. Aspecto da montagem do protótipo em protoboard.

```
apt-get install sox
apt-get install flac
apt-get install mpg123
```

Crie um script chamado `voz.sh` e digite a seguinte linha do **box 1**. Repare que para a gravação do áudio, utilizamos o dispositivo `plughw:1` que geralmente corresponde à webcam conectada à USB do seu Raspberry.

Observação: Este arquivo pode ser baixado em www.zolalab.com.br/repos/voz.sh. O seguinte comando irá efetuar o download: `wget www.zolalab.com.br/repos/voz.sh`.

Após criar o arquivo, não esqueça de adicionar o atributo de execução ao mesmo, utilizando o comando: `chmod +x voz.sh`.

Uma dica importante é a regulagem do ganho do microfone, que pode ser feita através do software `alsamixer`, que é um *mixer* de som. Digite `alsamixer` e após executá-lo, tecle em F6, e escolha o dispositivo de entrada do microfone, no caso a sua webcam. Agora, tecle "F5" para que todos os canais sejam mostrados. Pronto, é só regular o ganho de entrada.

Neste momento, você já pode controlar o seu Raspberry através de dois tipos de entrada:

- Infravermelho;
- Áudio/voz.

Para testar, execute o arquivo `voz.sh` e após executá-lo, diga em voz alta uma ou duas palavras, e assim, obterá o retorno das palavras reconhecidas, em formato texto. Como passamos o parâmetro `-d 3`, estamos informando o gravador (`arecord`) que grave apenas 3 segundos de áudio. Você pode configurar isso aumentando (ou diminuindo) este valor.

Outro ponto a ser observado é que o áudio gravado só será enviado ao Google quando algum som for detectado, evitando assim consumo desnecessário do *link* de internet. A condição para que o áudio seja submetido ao Google, é a intensidade do volume detectado no microfone, que pode ser modificado no trecho `"if [$volume > 3.0]"` do *script*, conforme a sua necessidade. **T**

Box 1

```
clear
while [ true ]
do
echo "Aguardando comando de voz"
arecord -q -D plughw:1 -d 3 -f cd -t wav -r 16000 /dev/shm/voz.wav
volume=`sox /dev/shm/voz.wav -n stats -s 16 2>&1 | awk '/Max level/ {print $3}`
echo "Volume detectado: $volume"
if [ $volume > 3.0 ]
then
flac /dev/shm/voz.wav -f --best --sample-rate 16000 -o /dev/shm/voz.flac 2>/dev/null
palavra=`wget -O - -o /dev/null --post-file /dev/shm/voz.flac --header="Content-Type: audio/x-flac; rate=16000" http://www.google.com/speech-api/v1/recognize?lang=pt-BR | iconv -f UTF-8 -t ISO-8859-1 | awk -v txt="text" '{sz=split($0,arr,""); for (p=1; p<=sz; p++) print arr[p]; exit }' | awk -F: 'NR==3 { print $3; exit }'`

palavra=`echo $palavra | tr -d ^`
echo
echo $palavra
echo
if [ "$palavra" == "ligar tv" ]
then
echo "enviando comando para ligar TV"
irsend SEND_ONCE TV_SALA KEY_POWER
fi
if [ "$palavra" == "música" ]
then
mpg123 /mp3/musica.mp3
fi
echo
fi
```

Lista de Materiais

Raspberry Pi - Modelo B, configurado e rodando o sistema operacional Raspbian. Além disso, deve possuir conexão com a internet para que o reconhecimento de voz possa ser realizado.

Webcam USB - Servirá apenas como microfone, portanto, sugiro que utilize uma câmera barata. No meu caso, estou usando uma Webcam Microsoft VX-2000, pois ela é reconhecida automaticamente no Linux.

LED Emissor I.R. 5 mm TIL32 940 nm - Servirá para controlar os seus dispositivos eletrônicos.

LED Receptor I.R. TSOP38238 (2,5V a 5,5V) - Servirá para absorver/ aprender os comandos dos seus controles remotos. Por exemplo, você poderá copiar o botão de Volume Up do seu controle da TV para dentro do Raspberry.

Resistor - 41 ohms - amarelo, marrom, preto

Transistor - PN2222

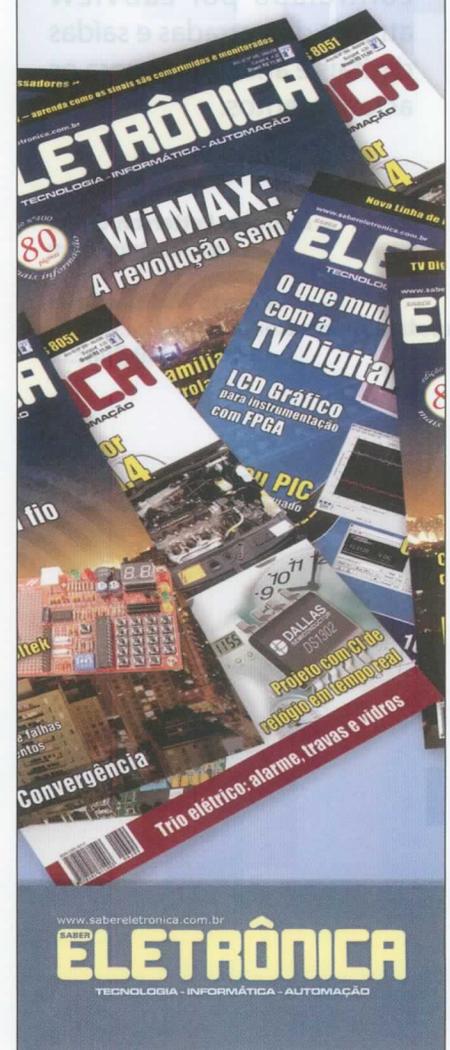
Receba a Saber Eletrônica em casa

Assine já!

(11) 2095-5335

ou acesse:

www.editorasaber/assinaturas



Braço Robótico com Arduino e LabVIEW

Este artigo apresenta o protótipo de um braço mecânico com 2 graus de liberdade, com movimentos de rotação, elevação, controle de abertura e fechamento da garra. Este braço robótico terá um controle utilizando a plataforma ARDUINO, e será controlado por LabVIEW através de entradas e saídas digitais e analógicas para os acionamentos.

Mauricio Petterson Gulicz
gulicz@msn.com

Roberto Mathias Susin
contato@laboratorioderobotica.com.br
eletrica.cmi@unc.br

Este projeto tem como objetivo deslocar uma peça de uma esteira e colocá-la com o braço robótico em uma caixa de classificação de produtos. O braço será acionado por servomotores e a esteira com motores DC através da porta USB de um computador com a utilização de um software de controle C++. A finalidade deste protótipo é servir de apoio didático na construção de conhecimentos através de um plano de aula definido na matéria "Automação e Robótica" do Curso de Engenharia Elétrica, da UnC/FUSM.

A disciplina de Automação e Robótica possui assunto referente a braços mecânicos e suas teorias, e no laboratório se encontra à disposição dos acadêmicos um *kit* de treinamento ou um protótipo de braço mecânico para provar, testar, medir, e comparar os conceitos teóricos com a prática.

Após realizar estudo juntamente com o professor Roberto Susin, o acionamento do braço robótico desenvolvido pelo acadêmico do curso de Engenharia Elétrica tem pouca precisão nos movimentos, e está sendo acionado com motores DC, com programação em LOGO. A solução que propomos é implementar o braço com servomotores e uma programação para que se tenha confiabilidade nos movimentos. Com isso, será implementada uma esteira de seleção de produtos usando plataforma ARDUINO com uma linguagem do software C++.

Referencial Teórico

Na robótica moderna há pesquisas e desenvolvimentos de robôs intitulados humanoides, ou antropomórficos. Estes são criados com semelhança humana e com capacidade de interagir com o ambiente, construídos pela monta-

dora japonesa Honda Motor. Citam-se ainda diversos brinquedos articulados com feições que lembram animais de estimação como cães, por exemplo, e que se destinam ao entretenimento. Contudo, tais robôs são incapazes de realizar quaisquer tipos de tarefas, e apenas respondem a estímulos externos. Com a evolução do tempo, muitas definições e soluções surgiram para os mais diversos problemas na constituição de um robô, e a robótica desenvolveu-se muito, tendo sido criados tipos de robôs para a aplicação nos mais variados segmentos envolvendo a realização de tarefas.

A Robótica aplicada

Apresentamos exemplos de aplicação em que o homem tem dificuldades na sua execução, como: nas montagens em série, que são difíceis para o homem pela sua repetição; na manipulação de materiais, sendo eles tóxicos e/ou considerados de risco; na substituição de peças pesadas; em soldagem e em pinturas. Ainda dentro do rol de tarefas executadas em ambientes de risco podemos citar as usinas nucleares, onde são realizadas tarefas de limpeza, remoção de resíduos nucleares, desativação de bombas, limpeza de campos minados etc., justificando assim a necessidade da existência da robótica na realização destas tarefas, porém com manipulação por controle remoto ou programação.

Portanto, a autonomia em manusear os robôs a longas distâncias é marcada como um grande passo para a história da robótica, onde a aplicação é caracterizada pela dificuldade de acesso a determinados locais como, por exemplo, no fundo do mar, dentro de crateras vulcânicas etc.

Robótica na Educação

A Robótica Educacional (ou Pedagógica) é desenvolvida para estimular a criatividade dos alunos devido à natureza dinâmica e interativa de construir e manipular experimentos criados a partir dos conceitos de robótica, além de ser motivacional para estimular o interesse dos estudantes no seu processo criativo e lógico (CADORI, 2012).

A robótica proporciona uma maior interação com os usuários, e no aprendizado entre os alunos, acontecendo de forma mais acelerada, mesmo que de forma oculta, conseguindo ainda relacionar-se com as demais áreas do conhecimento e do saber, na qual todos possam usufruir dos benefícios por ela oferecidos. (Castilho, 2002).

Braço Robótico

Um braço mecânico é formado pela base, braço e punho. O braço é ligado à base, que é fixada ao chão, à parede, ao teto, ou ainda a outra base. É o braço que efetua os movimentos e posiciona o punho. O punho é dotado de movimentos destinados a orientar (apontar) o órgão terminal. Este, por sua vez, executa a ação, mas não faz parte da anatomia do braço robótico, pois depende da aplicação a ser exercida pelo braço.

Motor de Corrente Contínua

Os motores de corrente contínua, muito utilizados na robótica, têm um par de terminais que devem ser ligados a uma fonte de alimentação. A polaridade da fonte é que determina o sentido de rotação do eixo do motor. Seu movimento é suave e contínuo e, com uma redução mecânica apropriada, são capazes de desenvolver um alto torque em volumes reduzidos. (PAZOS F., 2002).

Servomotores

Em geral, eles são muito utilizados na robótica, pois o controlador é um tipo dedicado que consiste num comparador em um amplificador com integrador. O servomotor tem uma precisão excelente, podendo pegar peças pequenas e levar para um determinado lugar e, assim sendo, existem outras aplicações para os servomotores.



F1. Garra em Alumínio.



F2. Arduino Uno.

Foi utilizado um servomotor de modelo SG – 5010, pois é de precisão, e foram feitos testes e verificações para elevar as peças que iriam entrar em sobrecarga. Então, com um servo de aplicação mais robusta conseguiu-se levantar a peça.

Materiais e métodos

Garra: A garra de alumínio, mostrada na **figura 1**, conforme foi utilizada está sendo de grande benefício para a implementação do braço, pois a anterior era muito frágil e não erguia muito peso, além do alumínio ou qualquer tipo de material, sendo a de alumínio mais resistente.

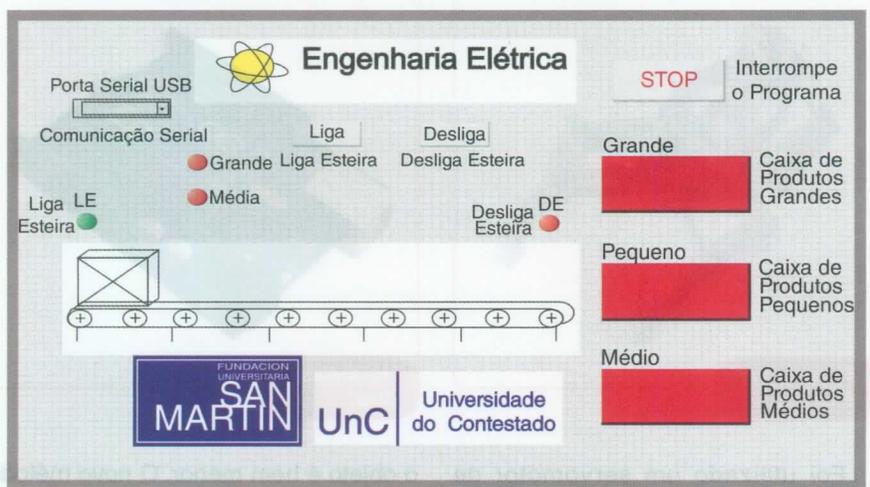
Os efetadores são destinados a pegar e segurar objetos para serem deslocados para dentro do espaço de trabalho do manipulador e, com isso, se tem maior desempenho no projeto. Tais objetos podem ser pequenos, frágeis como componentes eletrônicos na montagem de uma placa de circuito, ou pesados e grandes como carros, máquinas, que podem ser deslocados de um lado para outro em uma linha de produção. (PAZOS F., 2002).

Esta nova versão da Garra Robótica está representada na figura 1, a MK-II. Esta garra robótica é ótima para segurar diversos objetos. É feita de metal e é muito sólida. A garra abre por volta de 5 cm e pode segurar objetos de 200 g. O braço da garra se move paralelamente para você ter um melhor encaixe nos objetos que quiser segurar. Esta nova versão possui algumas melhorias. Ela tem uma nova liga, o que torna tudo mais rígido e a possibilidade de soltar

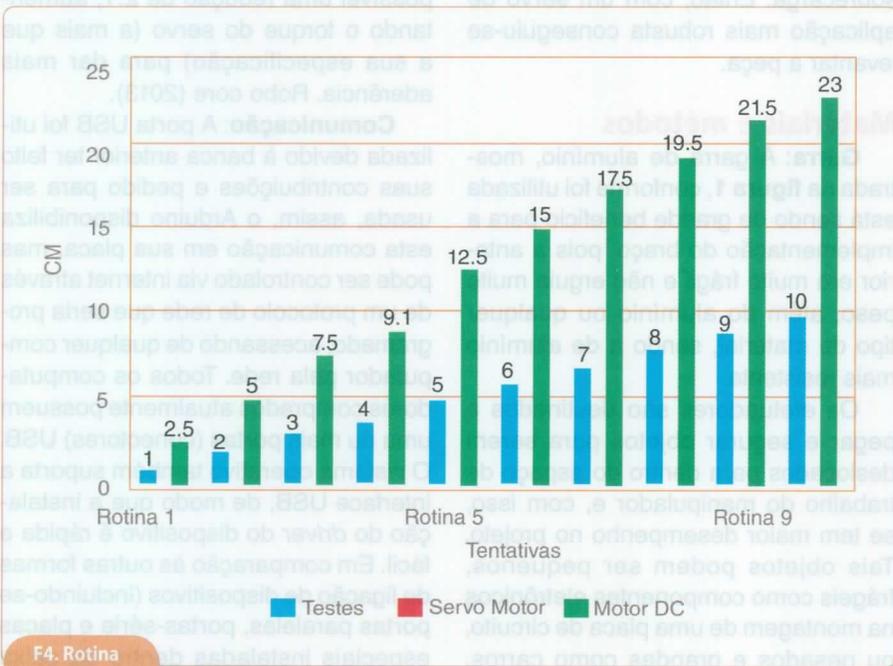
o objeto é bem menor. O novo método em que a garra foi feita é para ajudar a proteger as engrenagens de seus servos contra os danos. Além disso, é possível uma redução de 2:1, aumentando o torque do servo (a mais que a sua especificação) para dar mais aderência. Robo core (2013).

Comunicação: A porta USB foi utilizada devido à banca anterior ter feito suas contribuições e pedido para ser usada, assim, o Arduino disponibiliza esta comunicação em sua placa, mas pode ser controlado via internet através de um protocolo de rede que seria programado, acessando de qualquer computador pela rede. Todos os computadores comprados atualmente possuem uma ou mais portas (conectores) USB. O sistema operativo também suporta a interface USB, de modo que a instalação do *driver* do dispositivo é rápida e fácil. Em comparação às outras formas de ligação de dispositivos (incluindo-se portas paralelas, portas-série e placas especiais instaladas dentro da caixa da máquina), os dispositivos USB são incrivelmente de simples instalação no computador.

Software de Controle: O Arduino, ilustrado na **figura 2**, é uma placa de controle de entrada de dados (IN) como sensores, e saída de dados (OUT) como motores e LEDs, com cristal oscilador de 16 MHz, um regulador de tensão de 5 V, botão de *reset*, plugue de alimentação, pinos conectores, e alguns LEDs para facilitar a verificação do funcionamento. Neste projeto foram usadas as entradas digitais 2, 4, 7, 8 para os sensores, e para os servos foram utiliza-



F3. Supervisão em LabVIEW.



F4. Rotina de testes.

das as saídas com PWM 3, 5, 9, além de ser usada a digital 13 para ligar a esteira. A porta USB já fornece alimentação enquanto estiver conectada ao computador, e a tensão de alimentação quando desconectada pode variar de 7 V a 12 V, devido ao regulador presente na placa. (ARDUINO.CC 2013).

O Arduino é uma plataforma para acadêmicos e pessoas que gostam de fazer uma programação rápida e possui aplicações consideráveis, pois, atribuindo-se um kit ARDUINO

básico já se tem vários exemplos de aplicação na prática, e faz parte do conceito de hardware e software livre e está aberta para uso e contribuição de toda sociedade.

Comunicação e controle: O LabVIEW permite a análise e interpretação virtual através da construção de instrumentos virtuais, tais como indicadores gráficos, analógicos e/ou digitais. Possibilita também a criação de interfaces amigáveis muito semelhantes aos painéis de máquinas, podendo ser

utilizado como uma interface homem-máquina. Desta forma, por ser muito similar a instrumentos reais, os programas em LabVIEW são chamados de VI's (*Virtual Instruments*). A linguagem de programação utilizada para ele é chamada de linguagem G, por se tratar de uma linguagem gráfica que utiliza blocos de funções.

Todo instrumento virtual é composto, basicamente, de três partes principais: o Painel Frontal, o Bloco Diagrama e o Conector de Ícones. O painel frontal pode ser entendido como uma interface do usuário, ele está representado na **figura 3**, onde estão presentes os controles dos sensores (entrada de dados e valores) e os indicadores (saídas de dados e resultados). O diagrama de blocos é equivalente ao código-fonte do programa, onde estão presentes todas as estruturas de cálculo, de repetição, captura de sinais e outras, as quais são conectadas por "fios" que representam fluxos de dados, segundo Jeff Kodoskyv (2002).

Resultados

O projeto de implementação foi feito para melhorar o trabalho com base na rotina de testes que foi desenvolvida com o braço sem nenhuma mudança. Verificou-se que, com uma rotina de testes de 10 movimentos de precisão, ele conseguiu na décima rotina que o braço ficasse fora da precisão, então era de grande importância a confiabilidade dos movimentos, uma vez que haveria uma esteira de produtos, com isso, temos que ter uma precisão melhor nos movimentos.

A diferença constatada é devida ao acionamento do braço ser realizado por motores de corrente contínua sem realimentação, que, por não serem precisos nos seus movimentos de parada, causam em cada movimento pequenas diferenças que se acumulam e prejudicam a precisão dos movimentos. Após mudar os acionamentos dos motores para servo, notou-se diretamente as mudanças, pois a precisão é um fator que se tornou um objetivo do projeto, assim a precisão é de extrema importância. Com isso, os servomotores mostram como o seu controle atrelado

a uma boa programação fica de excelente precisão.

Rotina de testes

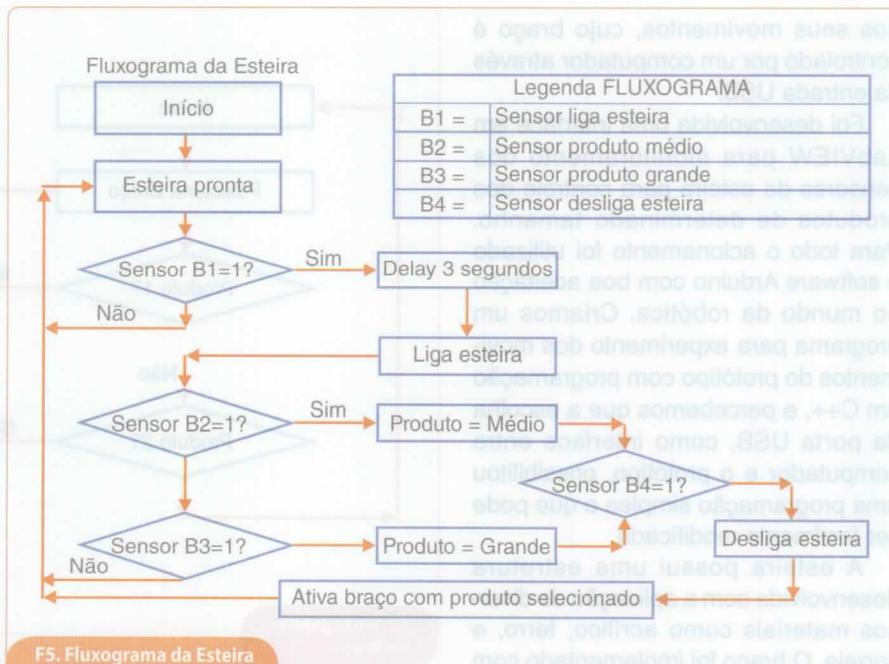
Com base na rotina de testes da **figura 4**, podemos considerar que a precisão, com programação adequada e mecanismos que auxiliem o projeto, irá ter um grau elevado. Foi realizada uma Rotina com 10 tentativas, sendo comparado em relação ao ponto inicial.

Com base nesta rotina, temos o motor de corrente contínua sem realimentação. Na décima tentativa a precisão do braço ao retornar ao ponto inicial estava com 77 %, não retornando ao local de origem, e cada vez mais se defasaria em centímetros do ponto inicial. Em dados estatísticos, a média de erro era de 13,31 cm e o desvio padrão 6,76 cm. Após isso, foi realizado um estudo de caso e foi definido que não havia confiabilidade no braço.

A opção escolhida foi o servomotor sendo controlado com Arduino. Como sabemos, o servomotor se desloca em graus para o ponto de origem, havendo na décima tentativa um erro de 0,34 cm e, com relação aos dados obtidos, devemos ter uma média de erro de 0,34 cm e o desvio padrão de 0,12 cm. Todos os mecanismos podem ser programados e ter mais precisão com uma programação ideal. O estudo do caso elaborou que a precisão é de 99,66% com servomotores e a programação. Além da precisão existente no projeto, há um mecanismo de proteção caso o servo não pare no lugar onde deve deixar o produto, atuando dois interruptores fim de curso e desligando o braço, cortando a energia do circuito, e assim, retornando os micros fim de curso, de modo que o projeto retorna ao seu funcionamento correto, isto é, com apenas um mecanismo de segurança.

Fluxograma da esteira

Na **figura 5** temos um fluxograma de funcionamento da esteira, utilizando sensores fotodiodos e fototransistor para leitura de produtos da esteira, para ligá-la e desligá-la. Após isso, o supervisor em LabVIEW tem um monitoramento da esteira que seleciona 3 produtos: pequeno, médio, grande.



F5. Fluxograma da Esteira de produtos.



F6. Esteira de Produtos.

Em determinado lugar, os produtos são colocados na esteira e levados até seu ponto de destino. Com o fluxograma podemos entender o funcionamento da esteira.

Fluxograma do Braço Robótico

Na **figura 6** mostramos uma esteira de produtos, e na **figura 7** vemos um fluxograma de funcionamento do braço que mostra que em cada seleção diferente, ele executará uma determinada função através dos sensores já comen-

tados anteriormente na figura 5. Temos no braço robótico servoacionamentos para uma melhor confiabilidade nos movimentos.

A **figura 8** mostra uma foto do braço robótico. Atente, agora para a **tabela 1**.

Conclusão

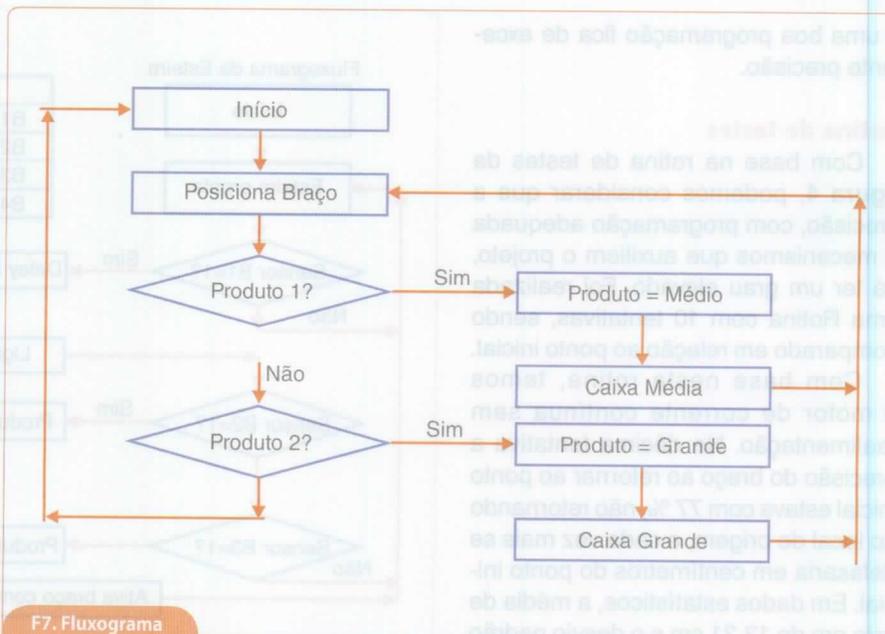
Implementou-se, conforme detalha o artigo, através de um estudo de caso, um braço mecânico com dois graus de liberdade e com um efetuador, uma garra de alumínio que trouxe mais força

aos seus movimentos, cujo braço é controlado por um computador através da entrada USB.

Foi desenvolvida uma interface em LabVIEW para monitoramento dos sensores da esteira para controle dos produtos de determinado tamanho. Para todo o acionamento foi utilizado o software Arduino com boa aceitação no mundo da robótica. Criamos um programa para experimento dos movimentos do protótipo com programação em C++, e percebemos que a escolha da porta USB, como interface entre computador e o protótipo, possibilitou uma programação simples e que pode ser facilmente modificada.

A esteira possui uma estrutura desenvolvida com a aplicação de diversos materiais como acrílico, ferro, e correia. O braço foi implementado com servomotores para uma maior confiabilidade nos movimentos. Os servos foram fixados com parafusos nos eixos. O efetuator consiste de uma garra de alumínio.

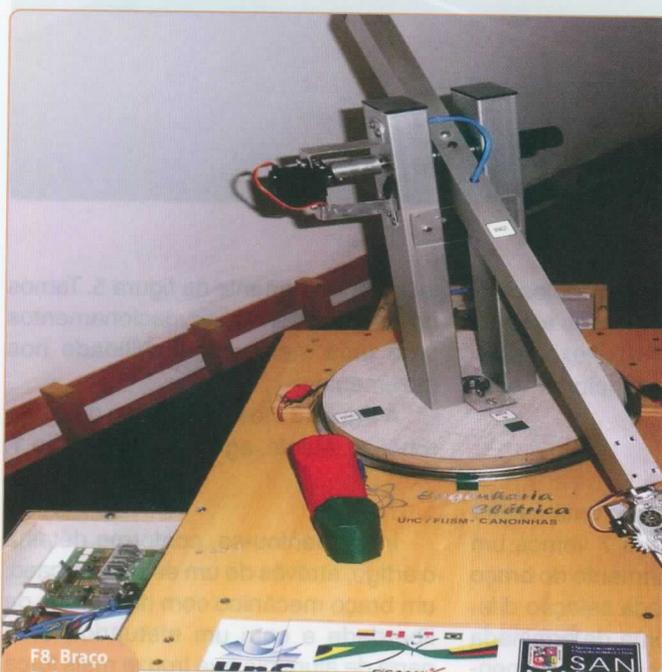
Ao implementar este projeto percebe-se a interdisciplinaridade, que é a atitude da busca (é um conceito e uma prática que está em processo de construção) e a multidisciplinaridade, que é um conjunto de disciplinas a serem trabalhadas simultaneamente com um objetivo único.



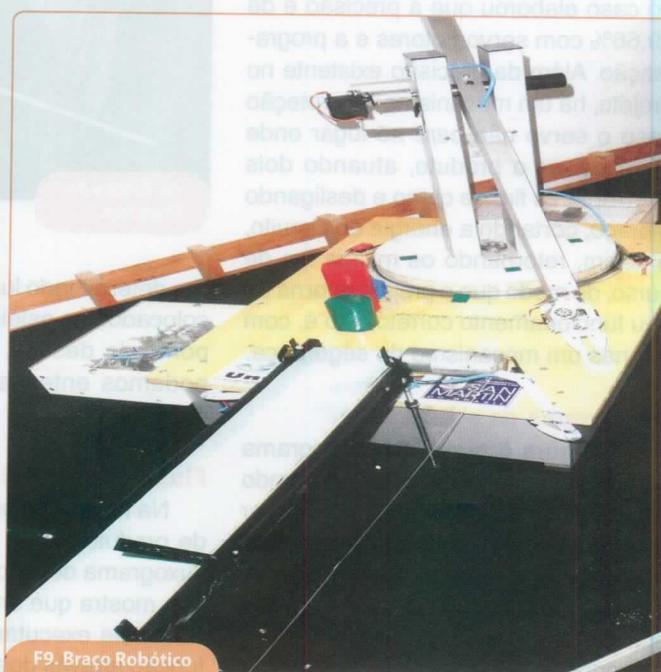
F7. Fluxograma do Braço.

Tempo	Desenvolvimento	Materiais e métodos	Avaliação
10 min.	Introdução Conceito do braço Robótico	PowerPoint, Funcionamento do Braço com Servomotores.	
20 min.	ARDUINO e LabVIEW	PowerPoint, Controladores, definições entrada e saída digital, comunicação, e tela de supervisão.	Realizando perguntas para verificar o acompanhamento da matéria
30 min.	Prática; Programas	Laboratório	Acompanhar alunos
10 min.	Conclusões	Relatório do Plano de Aula	Teste Final

T1. Plano de Aula.



F8. Braço Robótico.



F9. Braço Robótico com a Esteira.

Sensores e amplificadores operacionais

As características elétricas dos amplificadores operacionais tornam estes circuitos ideais para aplicações que envolvem sensores. De fato, na indústria, em eletrônica de consumo e em muitos outros campos da Eletrônica encontramos amplificadores operacionais exercendo funções importantes no interfaceamento de sinais de sensores com microprocessadores, DSPs e conversores analógicos digitais. Neste artigo apresentamos alguns circuitos básicos de amplificadores operacionais usados com sensores.

Newton C. Braga

Muitos sensores empregados em aplicações industriais de todos os tipos fornecem sinais fracos demais para que possam ser usados por circuitos convencionais. Esses sensores, para serem utilizados na prática, precisam de um interfaceamento apropriado que consiste em um circuito amplificador.

De todos os circuitos amplificadores que podem ser usados com esses sensores, os amplificadores operacionais, por suas características, são os mais requisitados.

Embora o amplificador operacional real esteja longe de apresentar as características do amplificador operacional ideal, as configurações que

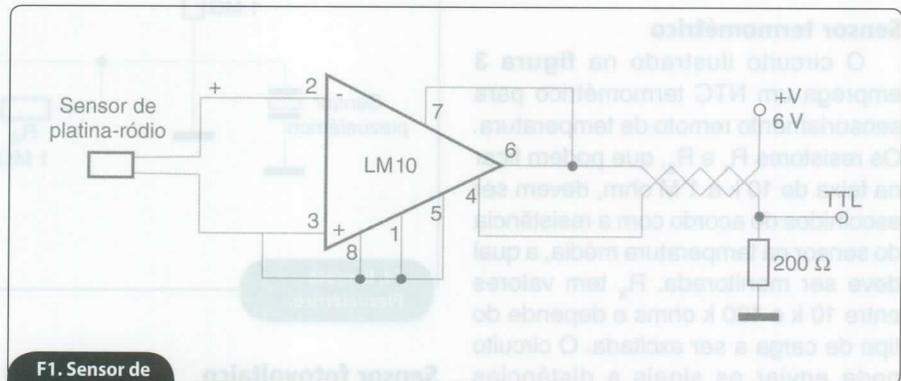
podem ser obtidas atendem perfeitamente à maioria das aplicações e por isso são largamente empregadas.

Os circuitos

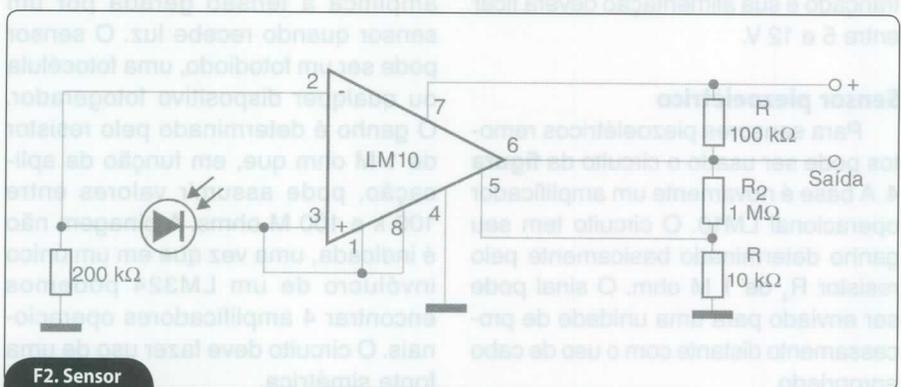
Observamos que muitos dos circuitos podem ser alterados tanto em função do desempenho desejado, já que as características dos sensores podem mudar sensivelmente, como também do possível uso de equivalentes para os amplificadores operacionais.

Sensores de chama

O primeiro circuito que apresentamos, mostrado na **figura 1**, é um sensor de chama piloto que pode ser usado em sistemas de aquecimento, em fornos



F1. Sensor de chama.



F2. Sensor de luz.

industriais e em muitas outras aplicações semelhantes. Ele faz uso de um sensor de ródio-platina que gera uma tensão, a qual depende de sua temperatura (par termoelétrico), fornecendo uma saída compatível com lógica TTL. Utiliza também um amplificador operacional LM10 que pode enviar os sinais a uma unidade de processamento via par trançado. Esta possibilidade permite que seja usado no monitoramento remoto de uma chama.

Sensor de luz

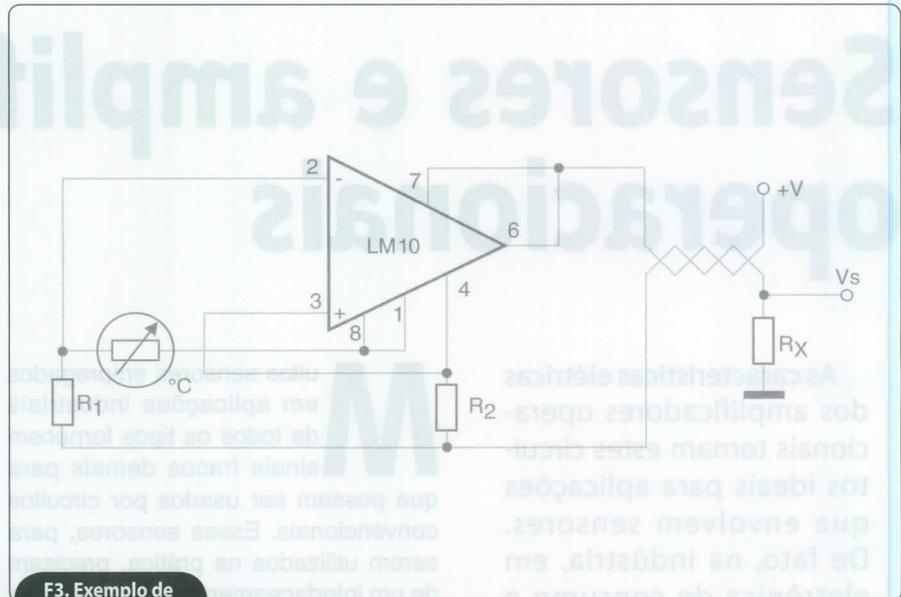
Na **figura 2** apresentamos um circuito que faz uso de um fotodiodo como sensor de luz. Ele se baseia também em um amplificador operacional LM10, sendo que os resistores R_2 e R_3 são responsáveis pela histerese do circuito. Nessa aplicação, o fotodiodo funciona como um gerador de tensão ligado de modo diferencial nas entradas do amplificador operacional. Os valores dos componentes, principalmente de R , podem ser modificados dependendo do tipo de carga que deva ser excitada. A alimentação pode ser feita com tensões de 5 a 12 V.

Sensor termométrico

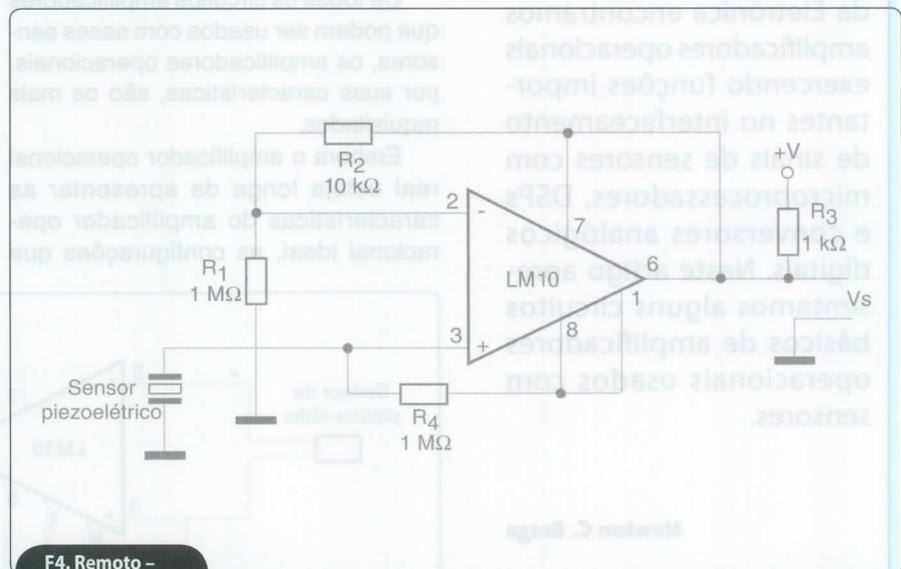
O circuito ilustrado na **figura 3** emprega um NTC termométrico para sensoriamento remoto de temperatura. Os resistores R_1 e R_2 , que podem ficar na faixa de 10 k a 1 M ohm, devem ser escolhidos de acordo com a resistência do sensor na temperatura média, a qual deve ser monitorada. R_x tem valores entre 10 k e 100 k ohms e depende do tipo de carga a ser excitada. O circuito pode enviar os sinais a distâncias relativamente grandes através de par trançado e sua alimentação deverá ficar entre 5 e 12 V.

Sensor piezoelétrico

Para sensores piezoelétricos remotos pode ser usado o circuito da **figura 4**. A base é novamente um amplificador operacional LM10. O circuito tem seu ganho determinado basicamente pelo resistor R_4 de 1 M ohm. O sinal pode ser enviado para uma unidade de processamento distante com o uso de cabo apropriado.



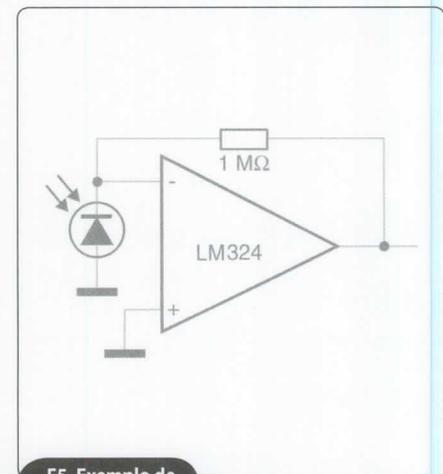
F3. Exemplo de termométrica.



F4. Remoto - Piezoelétrico.

Sensor fotovoltaico

O circuito exibido na **figura 5** amplifica a tensão gerada por um sensor quando recebe luz. O sensor pode ser um fotodiodo, uma fotocélula ou qualquer dispositivo fotogerador. O ganho é determinado pelo resistor de 1 M ohm que, em função da aplicação, pode assumir valores entre 100 k e 100 M ohms. A pinagem não é indicada, uma vez que em um único invólucro de um LM324 podemos encontrar 4 amplificadores operacionais. O circuito deve fazer uso de uma fonte simétrica.



F5. Exemplo de fotovoltaico.

Sensor em ponte

Existem diversos tipos de sensores que operam em pontes e que possuem características resistivas. Dentre eles, destacamos sensores de pressão mecânica, deformação e efeito Hall.

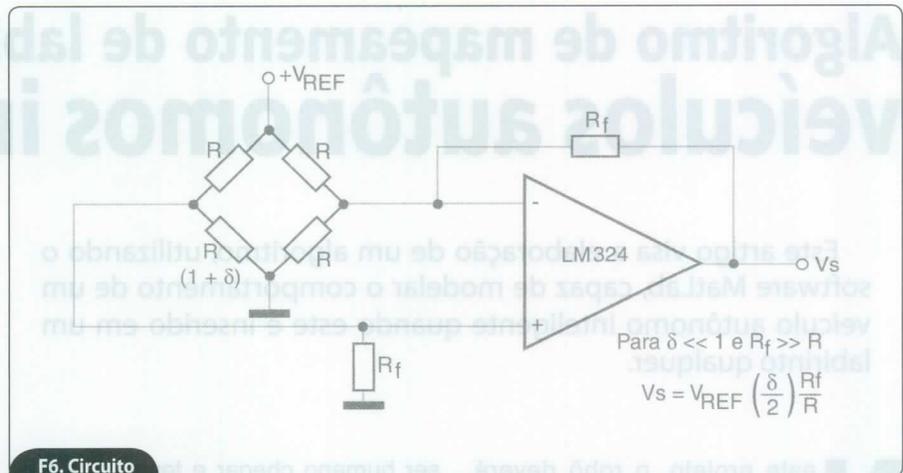
O circuito mostrado na **figura 6** pode ser usado eficientemente com esse tipo de sensor. A base é um dos 4 amplificadores operacionais encontrados em um LM324, ou equivalente. O ganho do circuito é determinado por R_f na realimentação, sendo que suas características de saída, em função dos valores dos componentes, são dadas pelas fórmulas junto ao diagrama. O circuito deve ser alimentado por fonte simétrica de 5 a 12 V.

Sensor de relutância variável

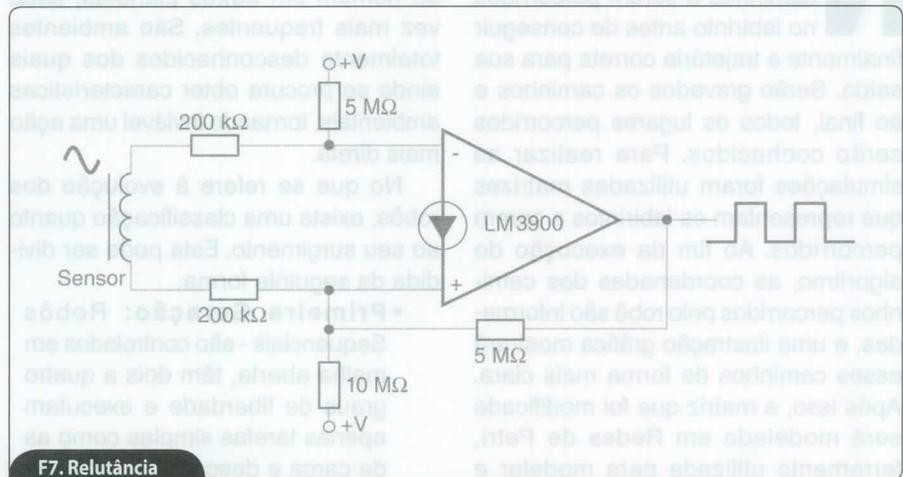
Sensores para medidas de velocidade de peças mecânicas, oscilações mecânicas e outras podem ser de tipos que se baseiam na variação da tensão gerada por uma bobina na presença de peças de metal em movimento. O circuito ilustrado na **figura 7** serve para aplicações desse tipo e tem por base um amplificador operacional de transcondutância (LM3900). Como em um invólucro do LM3900 encontramos 4 amplificadores desses, a pinagem não é dada. A alimentação deve ser feita com fonte simétrica e o resistor de 5 M ohms determina o ganho. Este ganho deve ser função da sensibilidade do sensor empregado e o valor do componente pode oscilar entre 1 M e 22 M ohms, tipicamente.

Seguidor de tensão

Um seguidor de tensão é um amplificador cujo o ganho de tensão é unitário. Como a impedância de entrada do circuito nesta aplicação é extremamente elevada (muitos megaohms) e a impedância de saída muito baixa (entre alguns ohms e algumas centenas de ohms), o circuito apresenta um elevado ganho de potência. Com sensores, este circuito é utilizado como um casador de impedâncias, podendo ser empregado para excitar convenientemente etapas que precisam de sinais intensos a partir de



F6. Circuito em ponte.

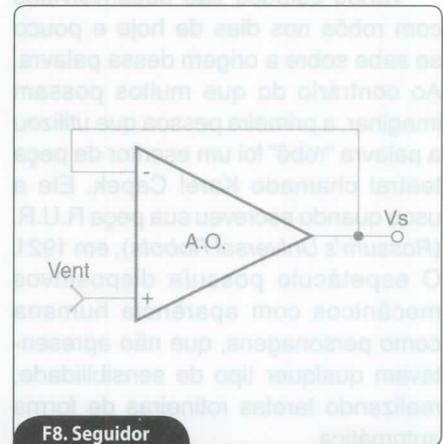


F7. Relutância variável.

fontes de alta impedância, as quais poderiam ser usadas diretamente. O circuito deve fazer uso de uma fonte simétrica. O seguidor de tensão na sua configuração básica é apresentado na **figura 8**.

Conclusão

Os amplificadores operacionais, apesar de fazerem parte de uma geração antiga de componentes, ainda são indispensáveis em uma grande quantidade de aplicações. Com o desenvolvimento de novas tecnologias, os amplificadores vão adquirindo características que se aproximam cada vez mais do ideal como, por exemplo, as elevadíssimas impedâncias de entrada dos tipos CMOS e JFET, ou as altas velocidades conseguidas com alguns tipos especiais para aplicações em transmissão de dados. No entanto,



F8. Seguidor de tensão.

em aplicações industriais onde os sensores ainda são os tradicionais ou mesmo na eletrônica de consumo em que o baixo preço dos tipos comuns é importante, esses componentes continuam marcando sua presença. **T**

Algoritmo de mapeamento de labirintos aplicado a veículos autônomos inteligentes

Este artigo visa a elaboração de um algoritmo, utilizando o software MatLab, capaz de modelar o comportamento de um veículo autônomo inteligente quando este é inserido em um labirinto qualquer.

Caio Carneiro Silva Rocha
Eduard Montgomery Meira Costa

Neste projeto, o robô deverá encontrar todos os possíveis caminhos a serem percorridos no labirinto antes de conseguir finalmente a trajetória correta para sua saída. Serão gravados os caminhos e ao final, todos os lugares percorridos serão conhecidos. Para realizar as simulações foram utilizadas matrizes que representam os labirintos a serem percorridos. Ao fim da execução do algoritmo, as coordenadas dos caminhos percorridos pelo robô são informadas, e uma ilustração gráfica mostrará esses caminhos de forma mais clara. Após isso, a matriz que foi modificada será modelada em Redes de Petri, ferramenta utilizada para modelar e analisar sistemas.

Vários estudos são desenvolvidos com robôs nos dias de hoje e pouco se sabe sobre a origem dessa palavra. Ao contrário do que muitos possam imaginar, a primeira pessoa que utilizou a palavra “robô” foi um escritor de peça teatral chamado Karel Capek. Ele a usou quando escreveu sua peça R.U.R. (*Rossum's Universal Robots*), em 1921. O espetáculo possuía dispositivos mecânicos com aparência humana como personagens, que não apresentavam qualquer tipo de sensibilidade, realizando tarefas rotineiras de forma automática.

Existem algumas atividades, nesse mundo de descobertas, que são de difícil acesso ao ser humano, isso quando não se fala na impossibilidade da intervenção direta do homem. Um exemplo típico são as perfurações de petróleo feitas em profundezas cada vez maiores, onde é impossível um

ser humano chegar e tentar qualquer ação. Outro exemplo são as investidas do homem em outros planetas, cada vez mais frequentes. São ambientes totalmente desconhecidos dos quais ainda se procura obter características ambientais, tornando inviável uma ação mais direta.

No que se refere à evolução dos robôs, existe uma classificação quanto ao seu surgimento. Esta pode ser dividida da seguinte forma.

- **Primeira Geração:** Robôs Sequenciais - são controlados em malha aberta, têm dois a quatro graus de liberdade e executam apenas tarefas simples como as de carga e descarga de prensas e máquinas ferramenta.
- **Segunda Geração:** Robôs a Ciclos Programáveis - são um pouco mais sofisticados e possuem de quatro a oito graus de liberdade.
- **Terceira Geração:** Robôs Inteligentes - são capazes de se adaptar às modificações do ambiente mediante sistemas de controle, percepção, comunicação e decisão.

Aqui é abordada a modelagem de um robô que se comporta como um sistema a eventos discretos. Dessa forma, são considerados quatro sensores capazes de detectar um caminho livre ou fechado, disparando uma transição quando encontrado um caminho livre. O disparo da transição define que um passo deve ser realizado. Esse passo é entendido como um evento discreto, o que caracteriza um sistema a eventos discretos.

Teoria de sistemas a eventos discretos

Um Sistema a Eventos Discretos pode ser classificado como dinâmico, invariante no tempo, não linear, com espaço de estados discreto e dirigido por eventos. O SED tem seu estado alterado em instantes discretos no tempo. Quando se fala da modelagem desse tipo de sistemas, se possível identificar um conjunto de eventos no qual cada evento possa causar uma transição de estado, a variável “tempo” não pode ser considerada uma variável independente.

Um SED é considerado dinâmico, pois sua saída em qualquer instante t depende de valores passados da entrada. Em sistemas invariantes no tempo, se sua função de entrada é aplicada ao sistema unidades de tempo depois de t , a função de saída deve ser igual à obtida em t , deslocada de unidades de tempo. A não linearidade dos sistemas está associada com a não validade do princípio da superposição. Quando se fala em sistemas com espaço de estados discretos, tem-se que as variáveis de estado pertencem a um conjunto discreto podendo ter variáveis com valores simbólicos, por exemplo, alto e baixo. E finalmente para um sistema dirigido por eventos, estes ocorrem de forma assíncrona no tempo, ou seja, a ocorrência é independente do tempo.

A principal diferença entre um sistema a eventos discretos e outro a eventos contínuos está no modo de interação com o ambiente no qual tal sistema está inserido. Para o primeiro caso, os eventos gerados no ambiente

provocam alterações na sua configuração interna determinando sua evolução dinâmica. Para os sistemas a eventos contínuos as modificações de estado são provocadas da troca de energia com o ambiente.

Um sistema a eventos discretos pode ser definido por três características básicas. São elas:

- Ciclo de funcionamento descrito através do encadeamento de eventos, que determinam as tarefas realizadas ou em realização;
- Ocorrência de eventos simultâneos, ou seja, diversos eventos podem ocorrer no mesmo período de tempo para modificar o estado do sistema;
- Necessidade de sincronização, uma vez que para se ter uma evolução dinâmica correta, o início de algumas atividades requer o término de outras.

Um SED pode ser definido como um sistema cuja evolução dinâmica depende da ocorrência de eventos. Um evento pode ter um caráter proposital, espontâneo ou pode ser um resultado da verificação de uma condição. Os eventos são responsáveis pela modificação de um estado presente em um estado futuro.

Um evento discreto pode ser considerado instantâneo, pois seu tempo de duração é desprezível para o sistema em geral. Um caso que pode exemplificar bem o que foi dito anteriormente, é o simples acionamento de uma lâmpada. O intervalo de tempo que a lâmpada leva para ser acesa entre o acionamento do interruptor e o momento em que a mesma foi finalmente ligada, é totalmente desprezível para o sistema em si.

Como exemplos de SEDs pode-se citar sistemas de manufaturas, sistemas de computação, sistemas de comunicação, sistemas de tráfegos, sistemas robóticos, entre outros diversos.

Teoria de Redes de Petri

“Redes de Petri” foi um modelo criado a partir da tese de doutorado de um alemão chamado Carl Adam Petri, a qual foi intitulada *Kommunikation mit Automaten* (Comunicação com Autôma-

tos) e apresentada na Universidade de Bonn em 1962. Tal modelo foi criado com o objetivo de modelar sistemas com componentes concorrentes.

As Redes de Petri são consideradas uma importante ferramenta matemática e gráfica na análise de diversos tipos de sistemas, inclusive em sistemas a eventos discretos, pois torna fácil a visualização do modelo de uma forma geral e o estado deste em qualquer instante de tempo.

Utilizando-se as Redes de Petri em modelagem de SEDs são observadas diversas vantagens em relação a outros modelos, dentre as quais algumas são listadas a seguir:

- Simplicidade na modelagem das principais características do sistema, por exemplo, concorrência, sincronismo e assincronismo, conflito, exclusão mútua, relações de precedência, não determinismo e bloqueio.
- Ótima visualização de dependência de sistemas. Focalização na informação local.
- Abordagem de modelagem do tipo refinamento e do tipo composição modular.
- Possibilidade de gerar código de controle supervisorio diretamente de sua representação gráfica.
- Possibilidade de testar propriedades indesejáveis para os sistemas, tais como bloqueio e reinicialização.
- Análise de desempenho sem simulação é possível para muitos sistemas, desde que taxas de produção, utilização de recursos, segurança e desempenho possam ser avaliadas.
- Simulação de eventos discretos diretamente a partir do modelo e informação do estado atual do sistema que permite monitoração em tempo real.

A Rede de Petri, ferramenta utilizada para modelar e analisar sistemas que apresentam concorrências e conflitos em sua dinâmica, é representada graficamente por grafos direcionados, bipartidos e ponderados com uma marcação inicial. A característica denominada de bipartidos faz alusão aos seus dois

tipos de nós, que são denominados de lugar e transição.

Os lugares são utilizados para modelar os componentes passivos do sistema, assim correspondendo às variáveis de estado. As transições são usadas para modelar os componentes ativos do sistema, ou seja, são os eventos que levam o sistema de um estado a outro. Os arcos sempre serão direcionados de um lugar para uma transição, ou de uma transição para um lugar.

Na **figura 1** pode-se observar os elementos de uma Rede de Petri, que foram descritos no parágrafo anterior. Assim, pode-se perceber que um arco parte do Lugar A para a Transição T e outro arco parte da Transição T para o Lugar B.

Em se tratando de representação gráfica, os círculos representam um lugar e as transições são representadas por uma barra ou um retângulo. Cada arco tem um peso discriminado pela letra k , que representa o número de arcos paralelos que ligam dois nós. A marcação de uma Rede de Petri atribui um número não negativo m , que são denominadas fichas. Essas fichas são representadas graficamente por círculos pretos. Essa marcação é um vetor $M = [m_1, m_2, m_3, \dots, m_k]T$, onde k é o número de lugares da rede, e m_i o número de fichas do lugar i .

O estado do sistema é dado pela divisão das fichas nos lugares da Rede de Petri, sendo que cada lugar representa um estado parcial do sistema. Cada evento ocorrido no sistema está associado a uma transição. A ocorrência de um evento é representada pelo disparo da transição ao qual este está associado. O disparo de uma transição consiste na retirada de fichas dos lugares de entrada, indicando que esta



F1. Componentes básicos de uma Rede de Petri.

condição não é mais válida após a ocorrência de um evento, e em seguida depositar fichas lugares de saída, indicando que estas atividades estarão, após o evento, sendo executadas.

Uma Rede de Petri é uma sêxtupla $RP = (P, T, Ar, K, W, M_0)$, onde:

- $P = \{p_1, p_2, p_3, \dots, p_m\}$ é um conjunto finito de lugares;
- $T = \{t_1, t_2, t_3, \dots, t_n\}$ é um conjunto finito de transições;
- $Ar \subseteq (P \times T) \cup (T \times P)$
- $K: P \rightarrow \mathbb{N} \cup \{\infty\}$ é a função de capacidade;
- $W: Ar \rightarrow \mathbb{N}^+$ é a função de ponderação;
- $M_0: P \rightarrow \mathbb{N}$ é a função de marcação inicial, que satisfaz $\forall p \in P: M_0(p) \leq K(p)$.

Levando em conta que em um sistema se deseja estudar somente o comportamento lógico, e ainda este apresenta alterações de estados definidos por transições discretas, pode-se utilizar uma metodologia para realizar a modelagem deste sistema. Essa metodologia de modelagem define que cada evento físico do sistema que resulta em uma mudança de estado é representado por uma transição na Rede de Petri. Assim, os estados do sistema são representados pela marcação da rede e os lugares são as partes que formam o sistema.

Com isso, tem-se que um lugar pode ser compreendido como o estado de um recurso ou de uma atividade. Se um lugar for interpretado como sendo o estado de um recurso, o número inicial de fichas pode ser constante para representar que existe uma quantidade fixa de recursos no sistema ou variável para representar o número de tarefas realizadas no sistema.

Sendo um lugar entendido como o estado de um recurso, a presença mínima de uma ficha nesse lugar indica que o recurso está disponível, entretanto a ausência de fichas indica justamente o inverso, ou seja, que o recurso é indisponível. No entanto, se um lugar for interpretado como o estado de uma atividade, a presença da ficha indica que a atividade está sendo realizada e a ausência de fichas indica que a atividade não está sendo realizada.

Por fim, uma transição pode ser entendida como o início de um processo ou uma atividade, e também como seu término. Tendo o conhecimento desses conceitos, pode-se modelar um sistema qualquer em Rede de Petri seguindo alguns passos, que serão listados a seguir:

1. Identificar todos os recursos e atividade suficientes para se ter o funcionamento do sistema em questão;

2. Determinar uma ordem de ocorrências das atividades de acordo com as relações de precedência definidas na descrição do funcionamento do sistema;

3. Para cada atividade determinada no tópico anterior deve-se:

- Criar e etiquetar um lugar para representar a condição da atividade;
- Criar uma transição para representar o início da atividade com arcos direcionados para os lugares de saída;
- Criar uma transição que represente o término da atividade com arcos direcionados para os lugares de entrada. De uma forma geral, a transição de término de uma atividade será a mesma transição de início da próxima atividade que foi listada. Quando se tem a rede sendo executada, uma ficha em um lugar indica que a atividade está sendo executada e várias fichas indicarão sua execução na multiplicidade do número de fichas. O disparo de uma transição de inicialização representa o início do processo e o disparo de uma transição de finalização indica a complementação da atividade e pode também representar o início da próxima atividade;

4. Para cada atividade ordenada: se um determinado lugar não foi criado, este deve ser criado e rotulado para cada recurso que deve estar disponível para iniciar a atividade. Todos os lugares de disponibilidade de recursos apropriados devem ser conectados com arcos a cada transição de entrada para a inicialização da atividade. Arcos de saída devem ser criados para conectar

as transições de finalização seguintes à atividade para algum lugar de recurso representando recursos que se tornem disponíveis na complementação da próxima atividade;

5. Especificar a marcação inicial para o sistema.

Desenvolvimento do algoritmo e modelagem DO LABIRINTO em Redes de Petri

Neste artigo são explorados os robôs móveis. Pode-se definir um robô móvel como sendo um veículo munido de um sistema de sensoriamento e um sistema atuador, controlado por uma estrutura capaz de executar tarefas determinadas pelo seu projetista. Todo robô móvel autônomo desenvolve um conjunto de atividades em que se faz necessário o uso de dispositivos de entradas, que são os sensores, e um conjunto de dispositivos de saída. Para fazer o controle de todo o sistema geralmente são utilizados microcontroladores ou microcomputadores.

A estrutura básica de um robô é composta por três sistemas, que são o sistema de controle, sistema de acionamento e sistema sensorial.

O sistema de controle, principal sistema da estrutura, é responsável pelo funcionamento e comportamento do robô. Ele vai gerenciar todo o sistema visando por fim uma determinada tomada de decisão. Grande parte do que se pesquisa em robôs móveis utiliza Redes Neurais Artificiais (RNA). Porém, neste artigo, serão utilizadas as Redes de Petri.

Já o sistema sensorial, também conhecido por sistema de percepção, tem a tarefa de obter informações do atual estado do robô, como também do ambiente em que este está inserido. Existem diversos tipos de sensores que são utilizados, por exemplo, os de infravermelho e os do tipo chave.

E por fim tem-se o sistema de acionamento, que são os atuadores responsáveis, de fato, pelos movimentos do robô. Os motores elétricos são muito utilizados.

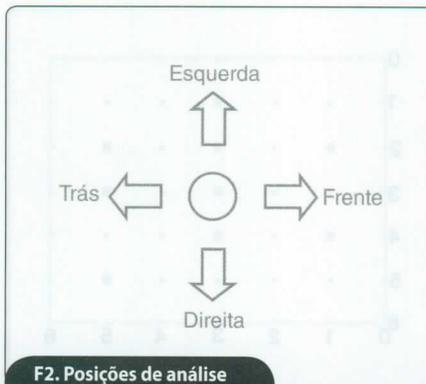
Neste trabalho é desenvolvido um algoritmo de mapeamento de labirintos através de um robô que se encaixa

perfeitamente na terceira geração. Este estará munido de sensores capazes de detectar obstáculos que irão ser desviados e, ainda, terão os caminhos obstruídos gravados para que não o faça novamente. Os sensores deverão captar sinais de quatro posições perpendiculares entre si, como mostra a **figura 2**.

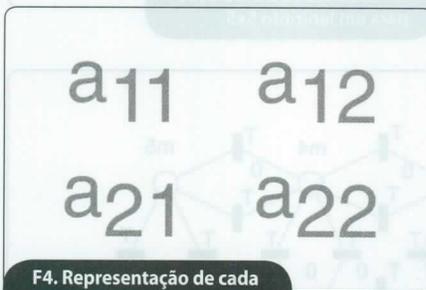
O robô se movimenta pelo labirinto, podendo assim descobrir o caminho correto a fazer para encontrar a saída. A tomada de decisão do robô é obtida através de testes realizados via sensores. Se o teste realizado for verdadeiro será realizado um passo, o que caracteriza um evento discreto. Com isso fica caracterizado um sistema a eventos discretos, que por sua vez pode ser modelado via Redes de Petri. No caso do presente trabalho, onde o objetivo é apenas a modelagem do sistema, os obstáculos encontrados pelos sensores serão identificados pelo número 1 na matriz principal. O caminho livre para o robô será dado pelo número 0. Como o algoritmo implementado tem uma ordem de execução, e para isso deve-se ter um critério de tomada de decisão, ao fim do caminho percorrido pelo robô existirão lugares onde este não passará. Isso se deve ao simples fato de o robô ter tomado um caminho que o leve para a saída, sem que seja necessário andar por um devido lugar que o levaria a um caminho fechado.

Na **figura 3** é mostrado um labirinto utilizado para realizar os primeiros testes do algoritmo. Pode-se notar que foi feito em uma matriz 5x5, tendo assim 25 lugares. Nesta, cada lugar é representado por um A_{ij} correspondente, pelo qual o índice i representa o número da linha e o índice j representa o número da coluna.

Para ilustrar o que foi dito a respeito de alguns lugares que não serão percorridos pelo robô, pode-se observar esse fato na figura 3. Os caminhos A31 e A51 não serão percorridos pelo robô, pois as direções adotadas na lógica implementada foram as apresentadas na figura 2. Assim, do ponto de vista da tomada de decisão na lógica, a direção DIREITA é tomada antes que o robô se movimenta na direção TRÁS.



F2. Posições de análise dos sensores.



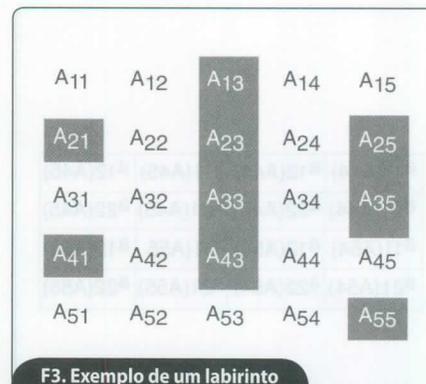
F4. Representação de cada célula da matriz principal.

Por outro lado, o robô percorrerá o lugar A_{45} , já que a direção FRENTE é tomada antes da direção DIREITA e ESQUERDA, contudo não será obtido sucesso, visto que a decisão tomada levará a um caminho fechado. Posteriormente ao teste para FRENTE será feito o teste para DIREITA, porém esse já foi um caminho percorrido pelo robô. Assim, o seguinte teste será para ESQUERDA, onde ele encontrará caminho livre, podendo assim continuar seu percurso até a saída desejada.

Para simplificar e ficar bem claro qual foi o critério utilizado para os testes de cada direção, tem-se que:

- 1º teste > Direção FRENTE
- 2º teste > Direção DIREITA
- 3º teste > Direção ESQUERDA
- 4º teste > Direção TRÁS

Cada célula da matriz principal será dividida em uma submatriz 2x2, na qual cada célula dessa sub-matriz indicará uma direção a ser tomada pelo robô. A célula a_{11} representará um passo para frente (F), a célula a_{12} um passo para direita (D), a célula a_{21} um passo para esquerda (E) e por fim se nenhum dos



F3. Exemplo de um labirinto percorrido pelo robô.

$a_{11}(A_{11})$	$a_{12}(A_{11})$	$a_{11}(A_{12})$	$a_{12}(A_{12})$
$a_{21}(A_{11})$	$a_{22}(A_{11})$	$a_{21}(A_{12})$	$a_{22}(A_{12})$
$a_{11}(A_{21})$	$a_{12}(A_{21})$	$a_{11}(A_{22})$	$a_{12}(A_{22})$
$a_{21}(A_{21})$	$a_{22}(A_{21})$	$a_{21}(A_{22})$	$a_{22}(A_{22})$

F5. Exemplificação do que ocorre no primeiro teste condicional.

testes for positivo, ou seja, encontre caminhos livres, será realizado o teste na célula a_{22} e o robô dará um passo para trás (T). A **figura 4** ilustra o vetor-linha que compõe cada célula da matriz principal.

Para ficar mais simples de visualizar o comportamento, são mostrados os valores que preenchem originalmente cada submatriz da **figura 5**.

$$A_{11} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$A_{12} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}$$

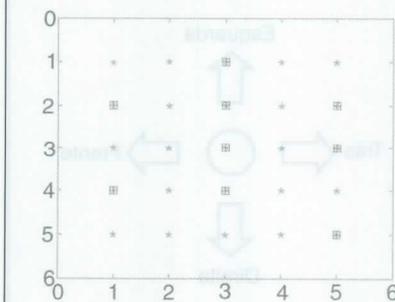
$$A_{21} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

$$A_{22} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

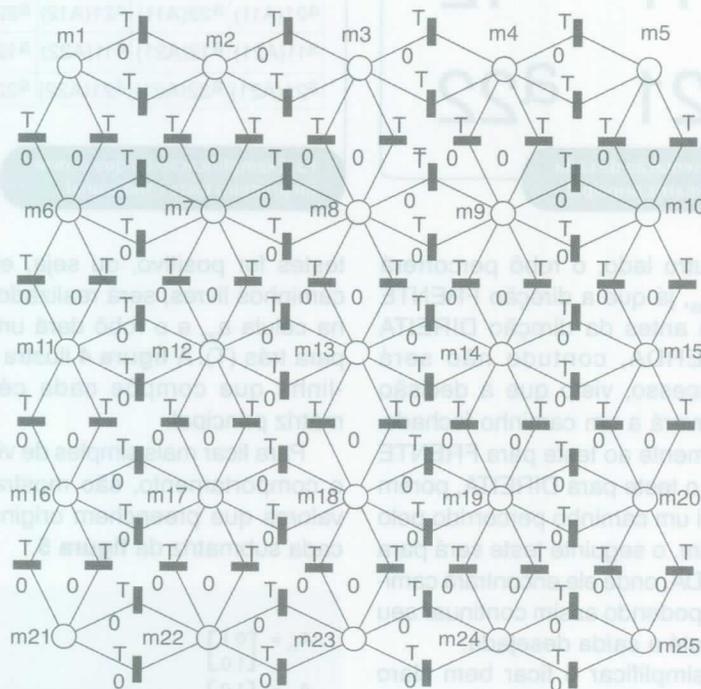
Na célula A_{11} , onde se encontra primeiramente o robô, será feito o teste para saber se existe ao menos um 0. Verifica-se que esse teste é verdadeiro e assim o programa entrará no primeiro teste condicional. Após isso, pela ordem direcional implementada, o teste será feito na célula $a_{11}(A_{11})$ que representa andar para FRENTE. Realizado esse teste se verifica que o caminho para

$a_{11}(A_{44})$	$a_{12}(A_{44})$	$a_{11}(A_{45})$	$a_{12}(A_{45})$
$a_{21}(A_{44})$	$a_{22}(A_{44})$	$a_{21}(A_{45})$	$a_{22}(A_{45})$
$a_{11}(A_{54})$	$a_{12}(A_{54})$	$a_{11}(A_{55})$	$a_{12}(A_{55})$
$a_{21}(A_{54})$	$a_{22}(A_{54})$	$a_{21}(A_{55})$	$a_{22}(A_{55})$

F6. Exemplificação do que ocorre no segundo teste condicional.



F7. Resultado da simulação para um labirinto 5x5.



F8. Rede de Petri primitiva do labirinto 5x5.

frente está livre, pois foi encontrado um 0 na célula. Com isso será atribuído o valor 2 à célula $a_{11}(A_{11})$ e só assim o robô dará o seu passo a frente. Quando esse passo for dado, o valor 2 será também atribuído a célula $a_{22}(A_{12})$ que representa andar para TRÁS.

Estando o robô na célula A_{12} , novamente será feito o teste para verificar a existência de no mínimo um 0. Confirmada a presença mínima do 0, inicia-se o teste direcional. Primeiramente testa a direção FRENTE sendo

verificada presença de um 1, o que significa um caminho bloqueado. Após isso, verifica-se a direção DIREITA, e encontra-se um 0, ou seja, um caminho livre. Assim será atribuído o valor 2 à célula $a_{12}(A_{12})$, e o robô dará um passo a DIREITA. Dado esse passo, a célula $a_{21}(A_{22})$ que representa a direção ESQUERDA receberá o valor 2. E o programa seguirá até a última célula seguindo a esse critério.

Esse valor 2 atribuído é utilizado para indicar que essa transição já foi

utilizada pelo robô, mas que é um caminho em aberto ainda. No decorrer da explicação da lógica implementada ficará mais claro o uso dessa atribuição.

Quando o robô encontra um caminho sem saída, ele tenderá a voltar a um lugar onde será realizado um novo teste para encontrar um caminho livre que possivelmente será o correto. Quando esse fato ocorrer, e o teste para saber se existe algum 0 na submatriz for feito, este será negativo, pois esta deverá estar preenchida por 1 e 2. Logo, o segundo teste condicional será realizado.

Esse segundo teste condicional funciona de forma semelhante ao primeiro, com uma pequena diferença. No primeiro teste os valores 0 eram substituídos por 2, ao passo que no segundo os valores 2 são substituídos por 1, indicando que aquele é um caminho incorreto e que o robô não deve voltar naquele lugar. Ficará mais simples o entendimento com a análise da figura 6.

Essa figura representa as células A_{44} , A_{45} , A_{54} e A_{55} da figura 3. Quando for realizado o teste na célula $a_{11}(A_{44})$ pela primeira vez será encontrado um 0, o que fará o robô dar um passo para a direção FRENTE. Com isso será atribuído o valor 2 nessa célula, o robô dará o passo a frente e será atribuído à célula $a_{22}(A_{45})$, que representa a direção TRÁS. Quando o robô estiver na célula A_{45} será realizado o primeiro teste condicional e então será verificado que não existe nenhum 0 nessa sub-matriz, levando o programa ao segundo teste condicional. Feito isso, serão analisadas as direções FRENTE, DIREITA e ESQUERDA, nessa ordem, e se constatará que todos esses valores são 1, o que indica caminho fechado. A última alternativa do programa é verificar a direção TRÁS em que será encontrado o valor 2. Assim será atribuído o valor 1 na célula $a_{22}(A_{45})$, que representa a transição para trás, o robô dará um passo para trás e então será atribuído o valor 1 na célula $a_{11}(A_{44})$, que representa o passo a FRENTE. Isso significa que o robô não mais voltará

para a célula A_{45} , já que este é um caminho fechado.

Para facilitar, pode-se resumir o significado dos números como se segue abaixo.

- 1 > Caminho livre nunca percorrido pelo robô.
- 2 > Caminho fechado definitivamente.
- 3 > Caminho livre percorrido pelo robô.

Para que fosse desenvolvido o código no MatLab foi elaborado um algoritmo, onde são descritos todos os passos do código. Esse algoritmo foi feito como está descrito a seguir.

- **1º passo:** ler o sensor que indica a posição FRENTE.
- **2º passo:** se encontrado um caminho livre, um passo a frente é dado e retorna-se ao 1º passo, se não, ler o sensor que indica a posição DIREITA.
- **3º passo:** se encontrado um caminho livre, uma passo a direita é dado e retorna-se ao 1º passo, se não, ler o sensor que indica a posição ESQUERDA.
- **4º passo:** se encontrado um caminho livre, uma passo a esquerda é dado e retorna-se ao 1º passo, se não, ler o sensor que indica a posição TRÁS, e como é o único caminho livre, ele dará um passo para o lugar que estava antes.

Resultados: simulações e modelagem

O algoritmo gera ilustrações que representam o labirinto encontrado pelo robô. Nessas ilustrações podem ser vistos três símbolos distintos que significam:

- Símbolo azul: trajetória percorrida pelo robô.
- Símbolo vermelho: paredes do labirinto.
- Símbolo preto: caminho não percorrido pelo robô.

O primeiro teste a ser realizado foi para um labirinto bastante simples composto de 25 lugares, dispostos em uma matriz 5x5, que foi mostrada na figura 3. Com o algoritmo implementado e executado, o resultado obtido

para essa simulação está mostrado na **figura 7**.

Observando essa figura, pode-se notar que pelo critério de tomada de decisão do algoritmo não foram percorridos apenas dois lugares de coordenadas (3,1) e (5,1). Além da ilustração gráfica, o programa implementado gera uma tabela com a ordem de lugares percorridos em formas de coordenadas. Na **tabela 1** estão listados os caminhos na ordem percorrida pelo robô.

Com o algoritmo implementado foi feita a modelagem do sistema em Redes de Petri. Foi feito o modelo considerado primitivo, onde aparecem todos os lugares e transições do sistema completo. Após isso, os lugares e transições que são cancelados quando o programa for compilado são mostra-

das pontilhadas em outra figura. Por fim é mostrada a Rede de Petri em uma nova figura, contendo as transições de ida e volta, o que ilustra o algoritmo compilado.

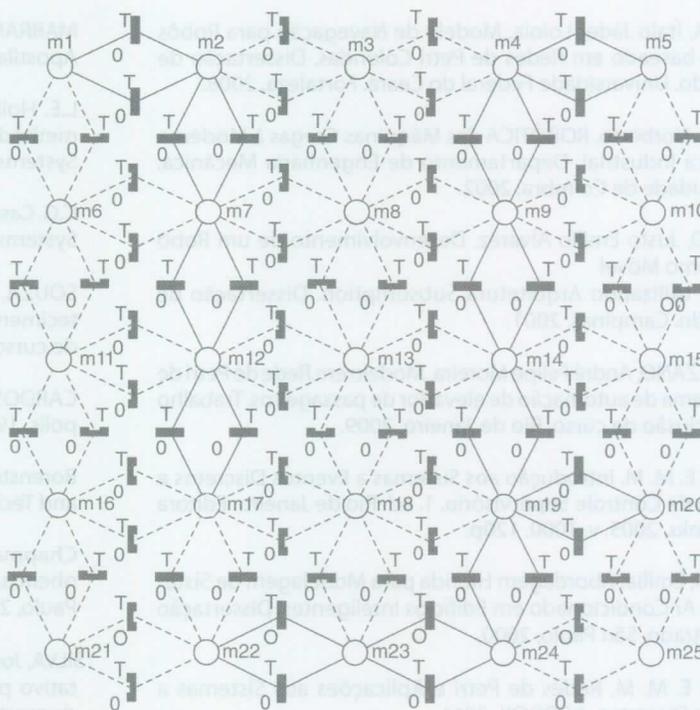
As redes primitiva, pontilhada e simplificada são mostradas nas **figuras 8, 9 e 10**, respectivamente.

Conclusão

O trabalho desenvolvido teve como objetivos a criação de um algoritmo capaz de mapear labirintos e modelar esses labirintos após o código ser compilado. Tendo em vista que existem vários lugares que o homem tenta explorar nos dias de hoje, porém onde não é possível sua presença física, pode-se constatar a importância desse trabalho.

(x y)	(x y)	(x y)
(1 1)	(5 2)	(4 4)
(1 2)	(5 3)	(3 4)
(2 2)	(5 4)	(2 4)
(3 2)	(4 4)	(1 4)
(4 2)	(4 5)	(1 5)

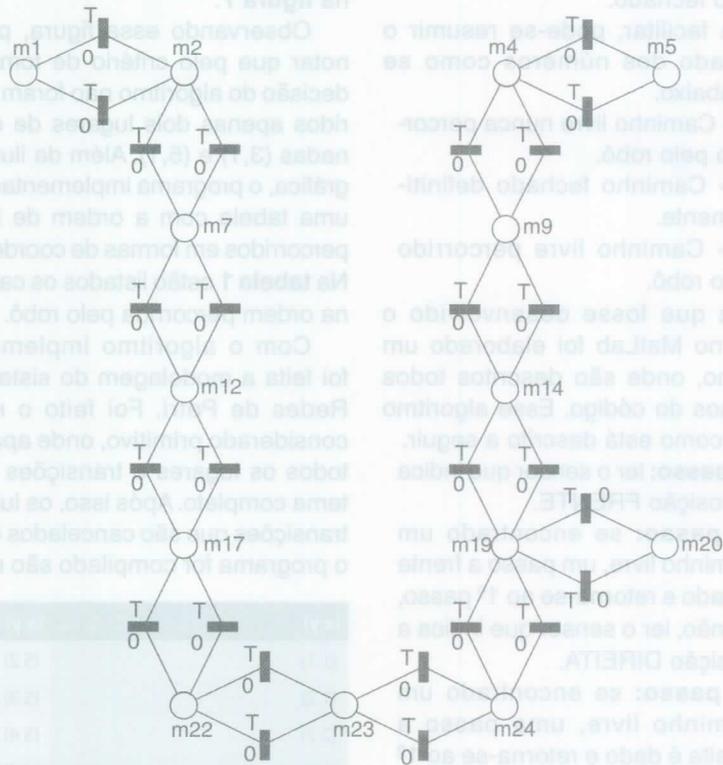
T1. Caminhos percorridos para o labirinto 5x5.



F9. Rede de Petri pontilhada do labirinto 5x5.

A escolha do MatLab para a realizar a implementação do algoritmo foi bastante apropriada, visto que este software permite uma manipulação de matrizes relativamente simples. Além disso, possui uma interface gráfica um tanto quanto ilustrativa, podendo assim gerar figuras gráficas que mostram os resultados obtidos, o que facilita o entendimento do usuário. Com o MatLab foi possível obter a sequência de lugares percorridos pelo robô, podendo assim elaborar uma tabela onde são mostrados os lugares na ordem encontrada pelo robô.

Para se obter a validação do modelo desenvolvido foram simulados, além do labirinto 5x5 mostrado, outros exemplos com maior número de lugares. Com isso pôde-se realizar testes para diversas situações, que foram impossibilitadas no labirinto de 5x5 lugares, devido à sua simplicidade. Com esses testes realizados verificou-se o bom comportamento do algoritmo para todas as situações testadas, tendo assim a comprovação do bom funcionamento do mesmo. **T**



F10. Rede de Petri simplificada do labirinto 5x5.

Referências Bibliográficas

BATISTA, Ítalo Jáder Loiola. Modelo de Navegação para Robôs Móveis baseado em Redes de Petri Coloridas. Dissertação de Mestrado. Universidade Federal do Ceará. Fortaleza, 2008.

PIRES, J. Norberto. ROBÓTICA das Máquinas Gregas à Moderna Robótica Industrial. Departamento de Engenharia Mecânica, Universidade de Coimbra, 2002.

JÁCOBO, Justo Emilio Alvarez. Desenvolvimento de um Robô Autônomo Móvel Versátil utilizando Arquitetura Subsumption. Dissertação de Mestrado. Campinas, 2001.

MONTEZANO, André Felipe Moreira. Modelo em Rede de Petri de um sistema de automação de elevador de passageiros. Trabalho de conclusão de curso. Rio de Janeiro, 2009.

COSTA, E. M. M. Introdução aos Sistemas a Eventos Discretos e à Teoria de Controle Supervisório. 1. ed. Rio de Janeiro: Editora Alta Books, 2005. v. 1000. 120p.

VILLANI, Emília. Abordagem Híbrida para Modelagem de Sistemas de Ar Condicionado em Edifícios Inteligentes. Dissertação de mestrado. São Paulo, 2000.

COSTA, E. M. M. Redes de Petri e Aplicações aos Sistemas a Eventos Discretos. AGBOOK, 2011.

MARRANGHELLO, Norian. Redes de Petri: Conceitos e Aplicações, Apostila, UNESP, 2005.

L.E. Holloway, B.H. Krogh, and A. Giua. A survey of petri net methods for controlled discrete event systems. Discrete Event Systems: Theory and Applications.

C.G. Cassandras and S. Lafortune. Introduction to Discrete Event Systems. Kluwer Academic Publishers, 1999.

SOUZA, Marlon Marques de. Sistema de Automação para Abastecimento de um Moinho de Cimento. Trabalho de conclusão de curso. Universidade Federal do Vale de São Francisco. 2011.

CARDOSO, Janette; VALETTE, Robert. Redes de Petri. Florianópolis, 1997.

Borenstein, H. R., et al., 1996. Navigation Mobile Robots: Systems and Techniques, AK Peters.

Chapman, Stephen J. Programação em MATLAB para engenheiros; tradução técnica Flávio Soares Correa da Silva. São Paulo, 2003.

SILVA, Joelson Coelho da. Uma proposta de controle neural adaptativo para a navegação de veículos autônomos. Dissertação de mestrado. Universidade Federal do Rio Grande do Sul, 1999.

Controles Automáticos de Temperatura

Diversas são as técnicas com que o leitor pode contar para o controle da temperatura de um ambiente. Os sensores de temperatura disponíveis são de vários tipos, como por exemplo: NTCs, diodos, e até mesmo transistores, o que abre ao projetista uma infinidade de possibilidades práticas, algumas das quais exploramos neste artigo.

Newton C. Braga

Manter a temperatura de um ambiente constante é uma preocupação em muitos projetos. Podemos incluir neste projeto as câmaras de refrigeração, salas de trabalho, estufas, chocadeiras, e até mesmo os transmissores cujos cristais precisam estar em uma determinada temperatura para atender as exigências legais.

O projetista de aparelhos eletrônicos, simples ou complexos, cuja finalidade seja manter a temperatura de um ambiente constante, tem muitas opções.

Além dos sensores que podem ser de diversos tipos, os próprios circuitos de acionamento admitem as mais diversas configurações em função do que devem ativar.

Neste artigo, focalizamos de forma mais detida os principais tipos de projetos que envolvem o controle da temperatura ambiente e fornecemos também algumas sugestões práticas para os leitores que estão ocupados neste tipo de trabalho.

Os sensores

Para os controles mais comuns de temperatura podemos contar com diver-

sos sensores, cuja escolha depende de diversos fatores como:

- Linearidade;
- Faixa de temperatura de operação;
- Sensibilidade;
- Custo.

Damos a seguir algumas das características desses sensores.

Diodos

Quando polarizamos um diodo no sentido inverso, ou qualquer junção PN, uma pequena corrente ainda circula. Esta corrente é devida à liberação de portadores de cargas pela agitação térmica dos átomos da junção, conforme ilustra a **figura 1**.

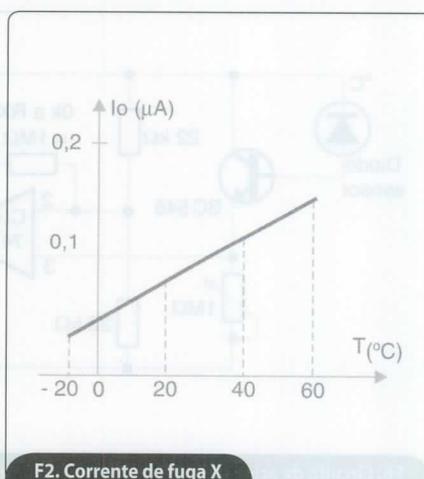
Dentro da faixa de -40 °C a $+125\text{ °C}$ a intensidade da corrente de fuga, como é denominada, varia quase linearmente com a temperatura absoluta, conforme mostra o gráfico da **figura 2**.

Em um trecho suficientemente pequeno desta curva, podemos considerar a resposta de um diodo como linear e, assim, usá-lo para a medida de temperaturas.

No entanto, as correntes obtidas de diodos comuns de silício (ou germânio), quando usados como sensores de tem-



F1. Portadores de carga na região da junção são liberados pelo calor.



F2. Corrente de fuga X temperatura.

peratura são muito baixas, da ordem de poucos microampères, o que exige o emprego de etapas amplificadoras com elevado ganho e linearidade.

A forma mais simples de se obter um aumento da corrente de tais sensores é exibida na **figura 3**, e consiste no uso de um transistor único como seguidor de emissor.

Esta configuração tem como característica o fato da elevada impedância de entrada da etapa se casar com as características do sensor, e com isso ser obtido também um elevado ganho de corrente.

Se mesmo com esta etapa ainda não tivermos um ganho suficiente para a aplicação desejada, poderemos partir para um Darlington, conforme visto na **figura 4**.

O ganho será dado aproximadamente pelo produto dos ganhos dos transistores usados, e a impedância de entrada será dada pelo produto da resistência da carga (Rx) pelo ganho da etapa.

O acionamento de um relé a partir de um sensor deste tipo, pode ser conseguido com certa facilidade, utilizando-se o circuito da **figura 5**.

Este circuito não é de grande sensibilidade, mas mesmo assim, dependendo do ponto em que deve operar, é possível acrescentar um controle que consiste de um potenciômetro (conforme mostra a mesma figura).

Para maior sensibilidade podemos utilizar a configuração da **figura 6**, que tem por base um amplificador operacional.

O sensor (diodo) é ligado a um transistor de tal modo que este componente, com um resistor, formam um divisor de tensão. Esta tensão é aplicada então à entrada não inversora de um amplificador operacional.

A tensão de referência é aplicada à entrada inversora, na qual também temos uma realimentação que fixa o ganho do circuito.

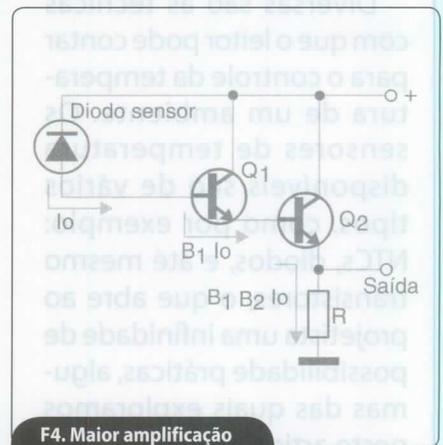
Deixando em aberto esta realimentação temos o ganho máximo e com isso uma ação de comutação mais aguda, conforme revela o gráfico da **figura 7**.

A tensão de referência determina o ponto de disparo e ele pode ser ajustado por meio de um potenciômetro.

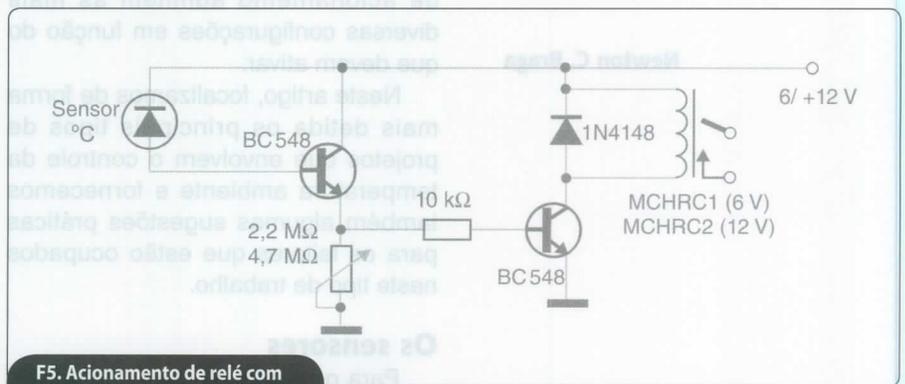
Se o transistor excitador for um NPN, teremos a energização da bobina do



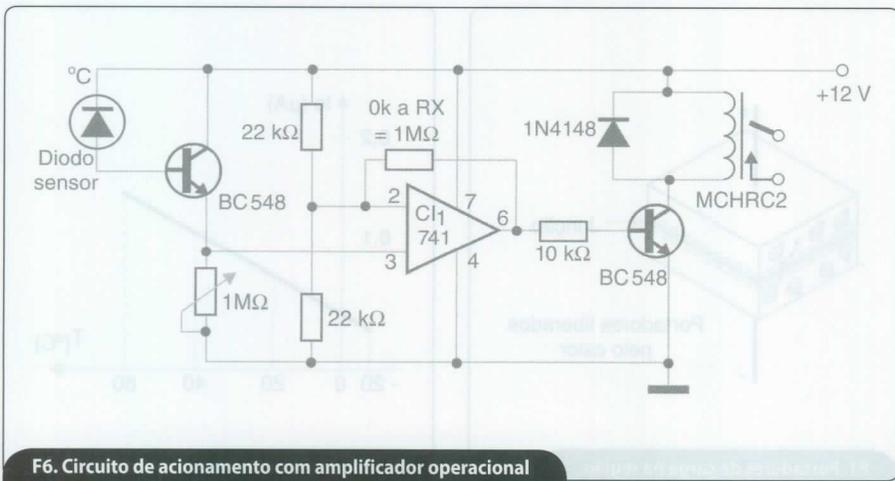
F3. Amplificando a corrente do sensor.



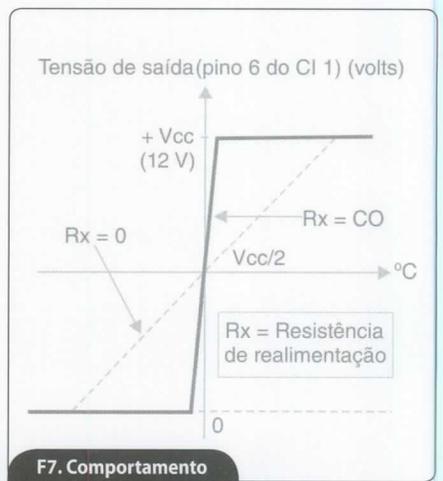
F4. Maior amplificação com 2 transistores.



F5. Acionamento de relé com controle de sensibilidade.



F6. Circuito de acionamento com amplificador operacional - liga com a elevação da temperatura.



F7. Comportamento do circuito.

relé, quando a saída do operacional for positiva. Por outro lado, se usarmos um transistor PNP na configuração mostrada na **figura 8**, o acionamento ocorrerá com a saída em 0 V. Estas duas possibilidades podem levar o circuito a controlar tanto elementos de aquecimento como de refrigeração, mantendo o relé energizado somente nos momentos em que a alimentação de tais dispositivos se fizer necessária. Isso é importante para reduzir o consumo de energia do aparelho.

Uma versão mais sofisticada inclui o disparo de um TRIAC com um acoplador óptico, conforme apresenta a **figura 9**.

Neste caso, quando o LED for energizado, quer seja pelo nível alto quer pelo nível baixo na saída do operacional, teremos o disparo do TRIAC controlando assim cargas de potência elevada. O acoplador é do tipo que inclui um optodiac.

A realimentação do operacional controla, neste caso, o ganho, e o trimpot no divisor de referência fixa a temperatura em que ocorre o disparo. Uma característica importante deste circuito é o seu isolamento.

Os optodiacs isolam eletricamente o circuito de controle da carga de alta potência, para maior segurança de operação.

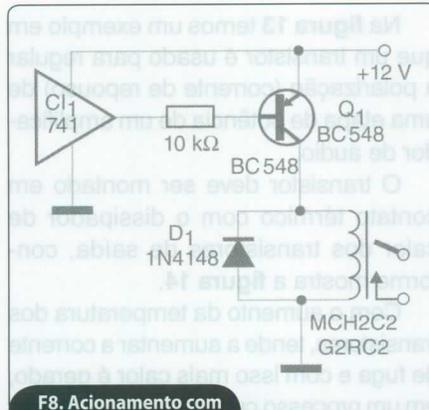
Transistores

Quando aplicamos uma tensão entre o coletor e o emissor de um transistor, mantendo a sua base aberta, flui uma pequena corrente, denominada de fuga (I_{ceo}), que também depende da temperatura.

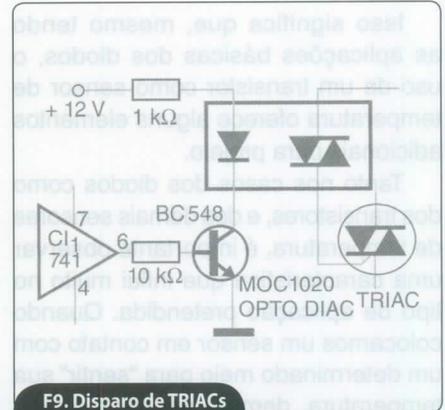
Esta corrente tem a mesma natureza da corrente de fuga dos diodos e por isso permite que transistores comuns sejam usados como sensores de temperatura.

Os circuitos de aplicações são basicamente os mesmos que vimos para os diodos, mas temos algumas possibilidades adicionais.

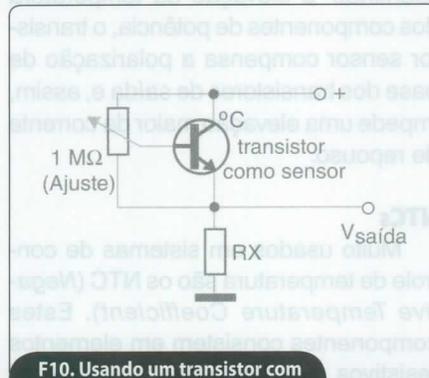
Podemos controlar a corrente de repouso e até mesmo amplificá-la com uma polarização conveniente da base do transistor, conforme mostra a **figura 10**.



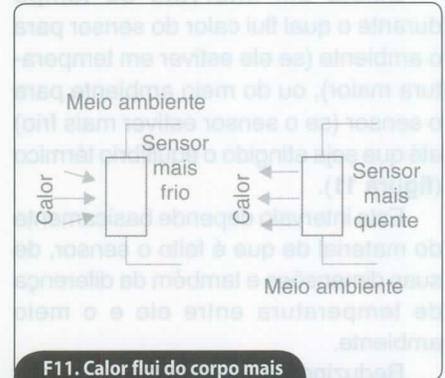
F8. Acionamento com transistor PNP.



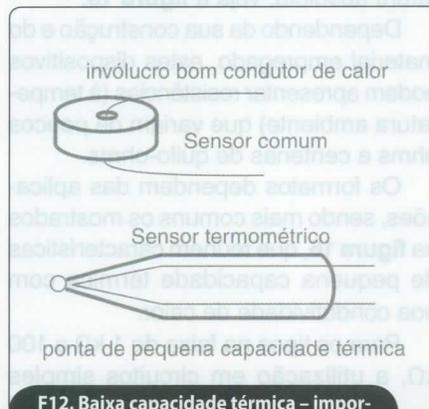
F9. Disparo de TRIACs com optodiac.



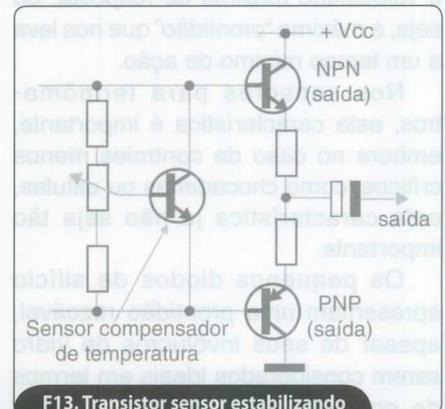
F10. Usando um transistor com sensor de temperatura.



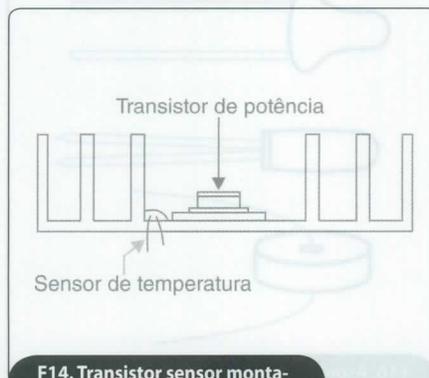
F11. Calor flui do corpo mais quente para o mais frio.



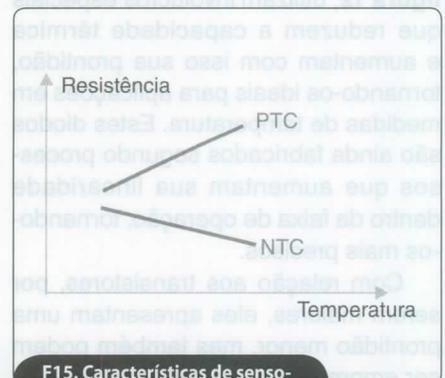
F12. Baixa capacidade térmica - importante nos sensores termométricos.



F13. Transistor sensor estabilizando termicamente uma saída de áudio.



F14. Transistor sensor montado em um dissipador.



F15. Características de sensores resistivos de temperatura.

Isso significa que, mesmo tendo as aplicações básicas dos diodos, o uso de um transistor como sensor de temperatura oferece alguns elementos adicionais para projeto.

Tanto nos casos dos diodos como dos transistores, e dos demais sensores de temperatura, é importante observar uma característica que influi muito no tipo de aplicação pretendida. Quando colocamos um sensor em contato com um determinado meio para “sentir” sua temperatura, demora um certo tempo até que isso ocorra.

Existe um intervalo de tempo durante o qual flui calor do sensor para o ambiente (se ele estiver em temperatura maior), ou do meio ambiente para o sensor (se o sensor estiver mais frio) até que seja atingido o equilíbrio térmico (**figura 11**).

Este intervalo depende basicamente do material de que é feito o sensor, de suas dimensões e também da diferença de temperatura entre ele e o meio ambiente.

Reduzindo a um mínimo a capacidade térmica do sensor, teremos então a velocidade máxima de resposta, ou seja, a máxima “prontidão” que nos leva a um tempo mínimo de ação.

Nos sensores para termômetros, esta característica é importante, embora no caso de controles menos críticos, como chocadeiras ou estufas, esta característica já não seja tão importante.

Os pequenos diodos de silício apresentam uma prontidão razoável, apesar de seus invólucros de vidro serem considerados ideais em termos de condutividade térmica. Diodos termométricos especiais como o da **figura 12**, utilizam invólucros especiais que reduzem a capacidade térmica e aumentam com isso sua prontidão, tornando-os ideais para aplicações em medidas de temperatura. Estes diodos são ainda fabricados segundo processos que aumentam sua linearidade dentro da faixa de operação, tornando-os mais precisos.

Com relação aos transistores, por serem maiores, eles apresentam uma prontidão menor, mas também podem ser empregados em muitos projetos.

Na **figura 13** temos um exemplo em que um transistor é usado para regular a polarização (corrente de repouso) de uma etapa de potência de um amplificador de áudio.

O transistor deve ser montado em contato térmico com o dissipador de calor dos transistores de saída, conforme mostra a **figura 14**.

Com o aumento da temperatura dos transistores, tende a aumentar a corrente de fuga e com isso mais calor é gerado, em um processo cumulativo que culmina com a queima dos transistores de saída. “Sentindo” a elevação da temperatura dos componentes de potência, o transistor sensor compensa a polarização de base dos transistores de saída e, assim, impede uma elevação maior da corrente de repouso.

NTCs

Muito usados em sistemas de controle de temperatura são os NTC (*Negative Temperature Coefficient*). Estes componentes consistem em elementos resistivos cuja temperatura diminui de uma forma quase linear com a temperatura absoluta, veja a **figura 15**.

Dependendo da sua construção e do material empregado, estes dispositivos podem apresentar resistências (à temperatura ambiente) que variam de poucos ohms a centenas de quilo-ohms.

Os formatos dependem das aplicações, sendo mais comuns os mostrados na **figura 16**, que reúnem características de pequena capacidade térmica com boa condutividade de calor.

Para os tipos na faixa de 1 k Ω a 100 k Ω , a utilização em circuitos simples



F16. Aspectos físicos de NTCs.

de controle pode ser feita conforme os exemplos que daremos a seguir.

Em uma aplicação mais simples, temos um medidor de temperatura com um NTC (sensor) e um microamperímetro como indicador. O circuito da **figura 17** tem uma razoável linearidade em uma faixa de temperaturas da ordem de 0 a 50 °C, o que permite usá-lo como termômetro simples.

Para o disparo de um relé, a partir da elevação ou queda da temperatura no sensor, temos o circuito da **figura 18**.

O potenciômetro ajusta o ponto de disparo, ou seja, a temperatura em que o relé fica no limiar do fechamento de seus contatos.

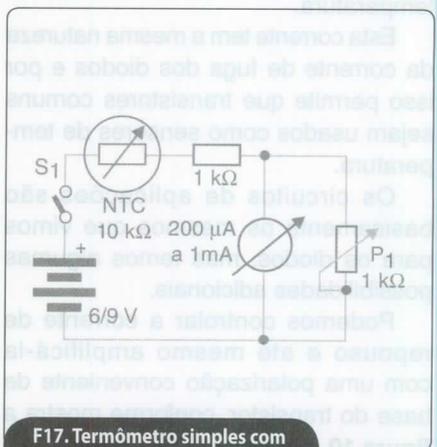
Este circuito opera bem na faixa de -10 a +100 °C, servindo como eficiente base para um projeto de controle de temperatura.

Em uma aplicação mais sofisticada podemos contar com o uso de um amplificador operacional na configuração dada na **figura 19**.

Neste circuito ajustamos a tensão de referência por P_1 , de modo que na temperatura limiar do disparo, a tensão na outra entrada seja da mesma ordem, conforme o NTC usado. A resistência do NTC deve ser da mesma ordem que a metade de P_1 na temperatura de disparo.

A grande variação de resistência que se obtém em função da temperatura tornam este tipo de sensor ideal para controles de estufas, câmaras de refrigeração etc.

Um fato importante que também precisamos analisar neste tipo de circuito é a sua histerese.



F17. Termômetro simples com NTC (escala não linear).

Conforme mostra a **figura 20**, a temperatura em que ocorre o fechamento de um relé é um pouco diferente daquela em que se obtém sua abertura. Isso ocorre porque a corrente de acionamento de um relé é maior que sua corrente de manutenção.

Se precisamos, por exemplo de 100 mA para acionar um relé, ele não abre seus contatos com os mesmos 100 mA, mas sim com um pouco menos, aproximadamente 90 mA, pois há uma corrente de histerese da ordem de 10 mA.

Em um controle de temperatura para uma estufa, por exemplo, esta histerese significa que não conseguimos mantê-lo exatamente em uma temperatura ajustada, mas sim dentro de uma faixa, conforme sugere a **figura 21**. A largura desta faixa depende, justamente, da histerese do circuito.

Com um amplificador operacional de alto ganho ou mesmo um bom comparador de tensão, podemos reduzir os efeitos pela histerese do relé, mas ainda assim ocorre o efeito da capacidade térmica do sensor.

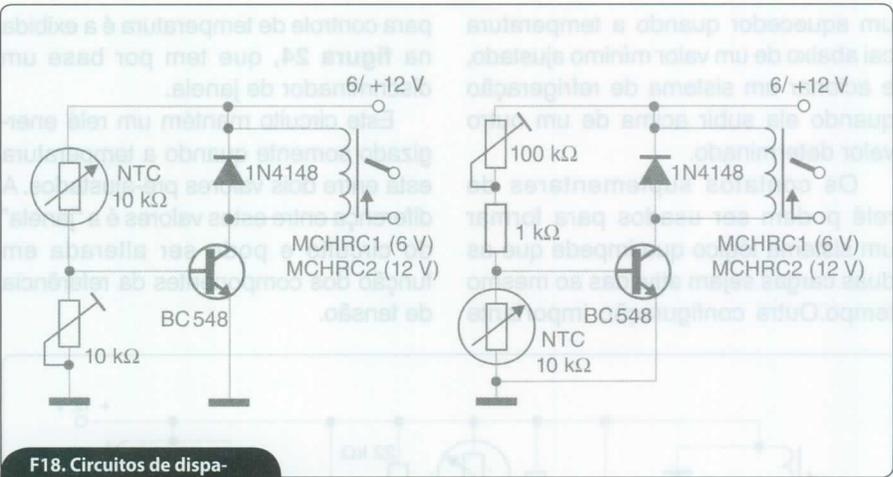
Uma vez que o sistema seja acionado para aquecer um ambiente, por exemplo, uma estufa, em função de uma queda da temperatura, demora um certo tempo para que o sensor “sinta” a recuperação desta temperatura. Este tempo faz com que o relé seja desligado mesmo quando a temperatura pré-ajustada já tenha sido um pouco superada, conforme ilustra a **figura 22**.

Este problema também faz com que a atuação do sistema sempre ocorra em uma faixa, cuja a largura irá depender da prontidão com que o sensor responde as variações de temperatura.

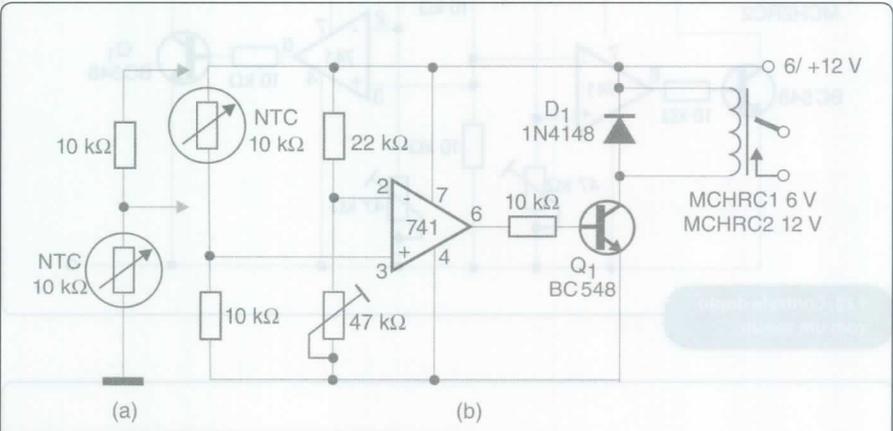
Circuitos bastante interessantes podem ser elaborados com base em NTCs, além daqueles que vimos.

Na **figura 23** temos uma aplicação dupla, ou seja, em que um único NTC é usado para acionar um relé quando a temperatura cai abaixo de certo valor e acionar o segundo relé quando ela supera outro valor.

O circuito integrado pode ser um duplo 741 como o MC1458 (ou equivalente). P_1 e P_2 são os ajustes de acionamento dos dois relés. Uma aplicação importante para este circuito é acionar



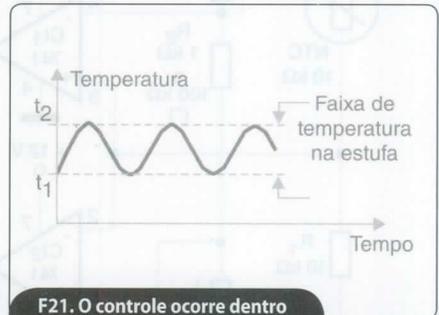
F18. Circuitos de disparo com NTCs.



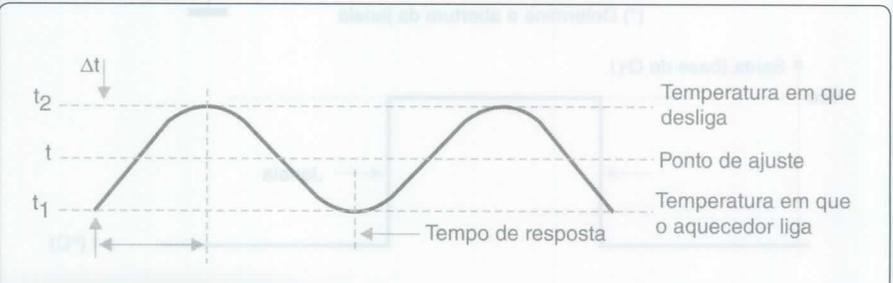
F19. Em (a) disparo com a elevação de temperatura, e em (b) disparo com a queda.



F20. Comportamento de um controle com relé.



F21. O controle ocorre dentro de uma faixa de temperatura.



F22. Ação retardada do controle de uma estufa.

um aquecedor quando a temperatura cai abaixo de um valor mínimo ajustado, e acionar um sistema de refrigeração quando ela subir acima de um outro valor determinado.

Os contatos suplementares do relé podem ser usados para formar um sistema lógico que impede que as duas cargas sejam ativadas ao mesmo tempo. Outra configuração importante

para controle de temperatura é a exibida na **figura 24**, que tem por base um discriminador de janela.

Este circuito mantém um relé energizado somente quando a temperatura está entre dois valores pré-ajustados. A diferença entre estes valores é a “janela” do circuito e pode ser alterada em função dos componentes da referência de tensão.

Um duplo comparador de tensão como os da série CA139 (podem ser usados dois comparadores dos quatro disponíveis nos integrados desta série) funciona bem neste tipo de aplicação. Operacionais comuns, como o 741, também podem ser usados.

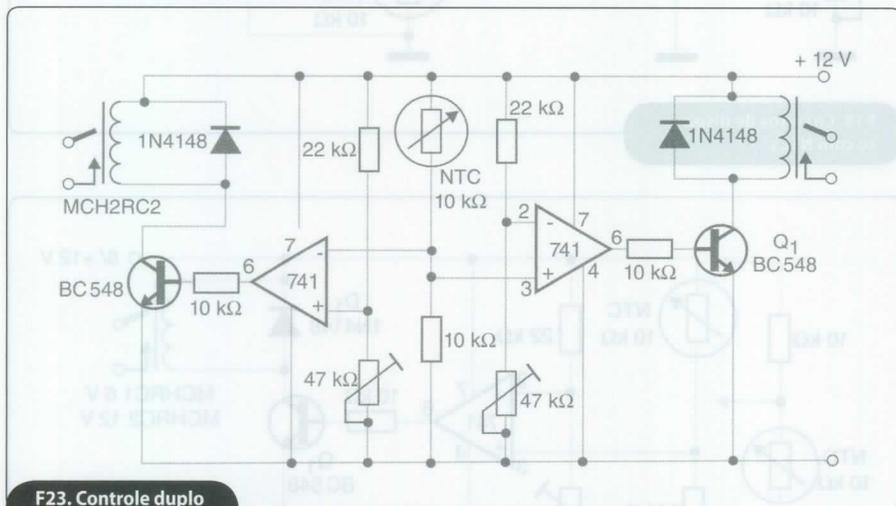
O circuito da figura 24 pode ser facilmente modificado para operar com optodiacs, obtendo-se assim, a configuração da **figura 25**.

O funcionamento é o mesmo. Quando os LEDs infravermelhos dos optodiacs são excitados, os TRIACs conduzem alimentando as cargas de potência. O isolamento da carga em relação ao circuito de controle é total.

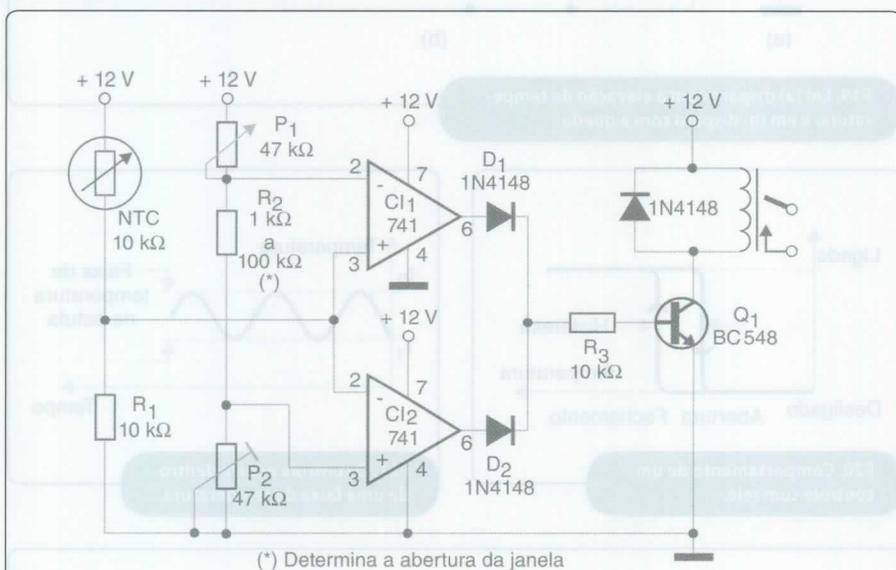
Podemos ainda modificar o circuito de modo que tenhamos uma janela de desativação, conforme revela o circuito da **figura 26**.

Neste circuito, o relé se mantém desativado quando a temperatura do sensor está entre os dois valores pré-ajustados.

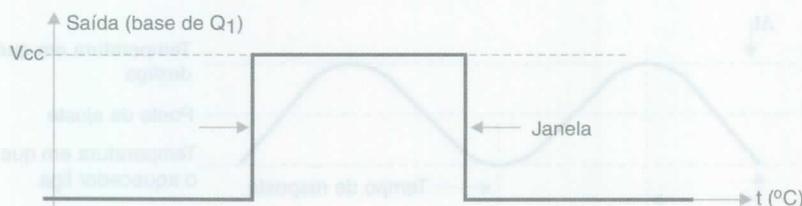
A largura da “janela” pode ser facilmente ajustada em função dos componentes da referência de tensão. Na saída, tanto podemos usar relés, conforme a aplicação indicada, como também optoacopladores do tipo com DIAC interno, caso em que podemos controlar diretamente um dispositivo de potência externo via TRIAC.



F23. Controle duplo com um sensor.



(*) Determina a abertura da janela



F24. Circuito para acionar o relé em uma faixa de temperatura.

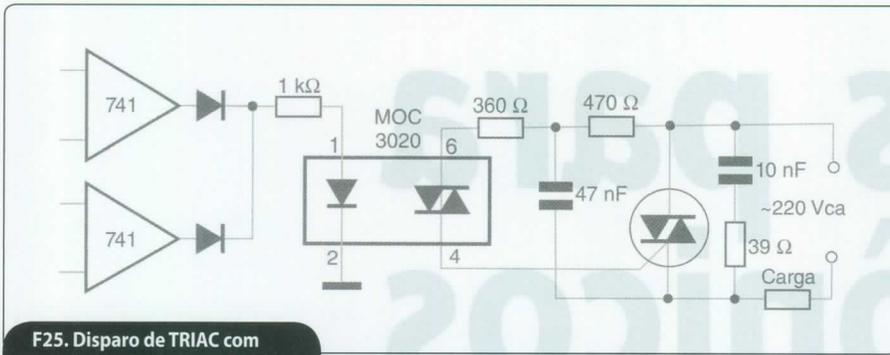
Outros circuitos

Existem circuitos integrados projetados especialmente para operar em controles de potência.

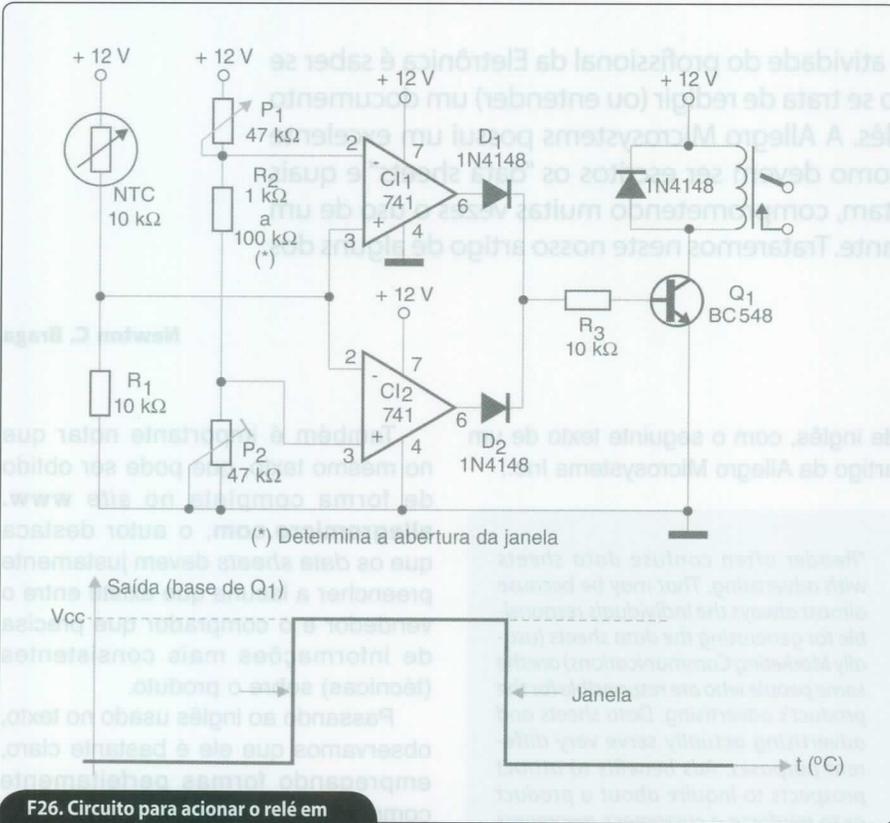
Um integrado importante neste tipo de aplicação é o TDA1023 que consiste em um controle eletrônico para TRIACs, especialmente projetado para permitir o controle muito preciso de equipamentos de aquecimento.

Na **figura 27** temos um circuito de aplicação para este integrado, indicado para potências de 1.200 W a 2.000 W, com uma faixa de temperaturas de 5 a 30 °C e usando o TRIAC BT139 da Philips Componentes. O circuito é projetado para a rede de 220 V e o NTC é de 22 kΩ na temperatura ambiente.

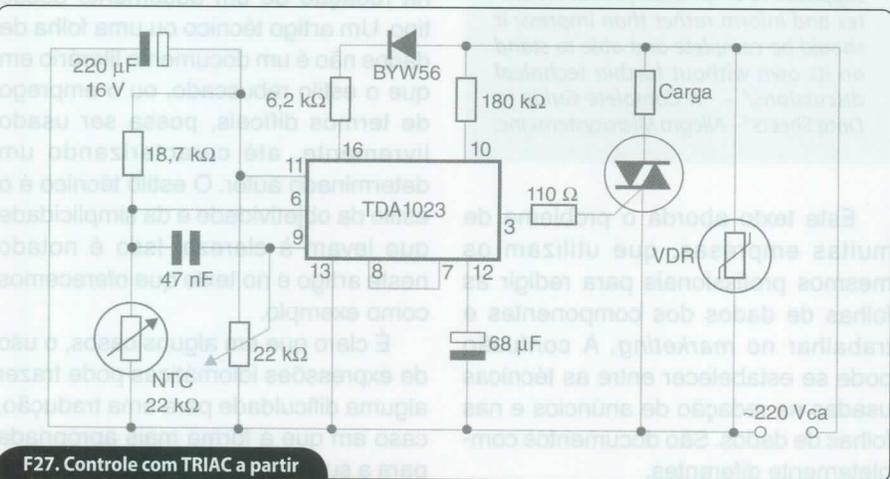
O VDR é de 250 V e o diodo é um BYW56. O resistor ligado ao pino 11 do integrado deve ter uma tolerância de 1%.



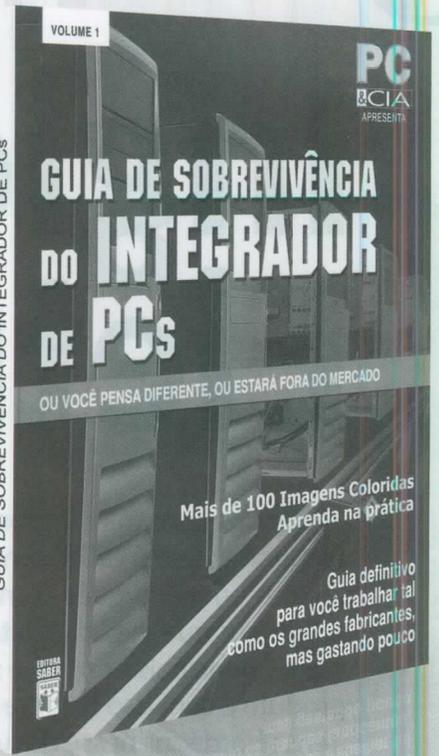
F25. Disparo de TRIAC com acoplador óptico (optodiad).



F26. Circuito para acionar o relé em uma faixa de temperatura.



F27. Controle com TRIAC a partir do TDA 1023 (Philips).



Trabalhe como os grandes, mas gastando como os pequenos!

Aprenda sobre:

- Construção de um laboratório técnico;
- Mitos e verdades sobre o aterramento elétrico, como fazer e medir corretamente;
- Equipamentos ideais para proteção elétrica;
- Montagem passo a passo de uma bancada de baixo custo protegida contra ESD;
- PCs confiáveis, o que você precisa saber para ter uma montagem à prova de falhas;
- Execução de testes de estresse e burn-in;
- Restauração automatizada do sistema: entregue mais valor para seus clientes;
- Medição de temperatura;
- Check-List para montagem e testes de micro.

Veja mais detalhes em www.novasaber.com.br

Inglês para Eletrônicos

Um dos itens mais importantes na atividade do profissional da Eletrônica é saber se comunicar. Isso é válido tanto quando se trata de redigir (ou entender) um documento técnico em português como em inglês. A Allegro Microsystems possui um excelente artigo em que explica exatamente como devem ser escritos os "data sheets" e quais são os principais erros que se constata, comprometendo muitas vezes o uso de um produto e com isso o nome do fabricante. Trataremos neste nosso artigo de alguns dos itens abordados nesse documento.

Newton C. Braga

Um dos pontos mais importantes, ao se redigir uma folha de dados ou "data sheet" para um componente, é levar em conta que esse documento pode significar um elo de ligação entre pessoas e empresas que têm necessidades opostas.

Para quem lê uma folha de dados, ela reflete a empresa que a redige. Normalmente, erros menores não são muito comentados, no entanto, erros de redação ou mesmo de concordância e, principalmente, erros técnicos refletem diretamente na credibilidade e no próprio desempenho do componente, e conseqüentemente da empresa.

Outro ponto de extrema importância a ser analisado é que uma folha de dados de um componente também é considerada pelo cliente como um documento legal de garantia da performance de um produto. Tudo isso significa que escrever uma folha de dados é uma arte, e saber interpretar seu conteúdo também.

Muitos usuários de componentes eletrônicos confundem as folhas de dados com peças de anúncios. Isso não é verdade, e é precisamente desse ponto que partimos para nossa lição

de inglês, com o seguinte texto de um artigo da Allegro Microsystems Inc.:

"Reader often confuse data sheets with advertising. That may be because almost always the individuals responsible for generating the data sheets (usually Marketing Communications) are the same people who are responsible for the product's advertising. Data sheets and advertising actually serve very different purposes. Ads benefits to attract prospects to inquire about a product or to reinforce a customer's awareness of a company. The data sheet is then supposed to emphasize product features and inform rather than impress; it should be complete and able to stand on its own without further technical discussions." – "A Complete Guide to Data Sheets" – Allegro Microsystems Inc.

Este texto aborda o problema de muitas empresas, que utilizam os mesmos profissionais para redigir as folhas de dados dos componentes e trabalhar no *marketing*. A confusão pode se estabelecer entre as técnicas usadas na redação de anúncios e nas folhas de dados. São documentos completamente diferentes.

Também é importante notar que no mesmo texto, que pode ser obtido de forma completa no site www.allegromicro.com, o autor destaca que os *data sheets* devem justamente preencher a lacuna que existe entre o vendedor e o comprador que precisa de informações mais consistentes (técnicas) sobre o produto.

Passando ao inglês usado no texto, observamos que ele é bastante claro, empregando formas perfeitamente comuns e um vocabulário não muito sofisticado, o que deve ser uma norma na redação de um documento desse tipo. Um artigo técnico ou uma folha de dados não é um documento literário em que o estilo rebuscado, ou o emprego de termos difíceis, possa ser usado livremente, até caracterizando um determinado autor. O estilo técnico é o estilo da objetividade e da simplicidade que levam à clareza. Isso é notado neste artigo e no texto que oferecemos como exemplo.

É claro que em alguns casos, o uso de expressões idiomáticas pode trazer alguma dificuldade para uma tradução, caso em que a forma mais apropriada para a sua compreensão deve ser ado-

tada. O leitor notará isso na tradução do texto que tomamos como exemplo. Sua tradução ficará como:

“Os leitores frequentemente confundem folhas de dados com anúncios. Isso pode ocorrer porque quase sempre os indivíduos responsáveis pela produção de folhas de dados (usualmente Comunicações e Marketing) são as mesmas pessoas que produzem os anúncios dos produtos. Folhas de dados e anúncios realmente servem para finalidades diferentes. Os anúncios visam ser prospectos atrativos para criar a procura de um produto, ou para reforçar a prevenção que um cliente tenha em relação a uma empresa.... A folha de dados supõe que os destaques do produto devem ser enfatizados e informar mais do que impressionar; ela deve ser completa e capaz de se suportar sem a necessidade de mais discussões técnicas”.

Veja que neste texto temos algumas formas que não permitem, ou não admitem uma tradução “ao pé da letra”. Assim, para obtermos um texto melhor em português, essas formas foram traduzidas pelo sentido e não diretamente pela maneira como apresentadas. Fica claro ainda neste texto, que existe um jogo entre os destaques e os benefícios de um produto, e quando um ou outro deve ser destacado.

Os destaques e as vantagens são importantes para uma folha de dados, mas muito mais importantes em um anúncio ou “press release”.

Documentos do ISO/IEC enfatizam o modo como os padrões internacionais devem ser usados no seguinte texto:

“The objective of a (data sheet) is to define clear and unambiguous provisions in order to facilitate international trade and communication. To achieve this objective, the (data sheet) must:

- *Be complete as necessary;*
- *Be consistent, clear and concise;*
- *Be comprehensible to qualified persons who have not participated in its preparation”- ISO/IEC Directives, Part 3, Drafting and Presentation of International Standards.*

A tradução fica:

“O objetivo de uma (folha de dados) é definir claramente e de forma não ambígua elementos de modo a facilitar o comércio e as comunicações internacionais. Para alcançar esse objetivo, a (folha de dados) deve:

- *Ser tão completa quanto necessária*
- *Ser consistente, clara e concisa*
- *Ser compreensível para pessoas qualificadas que não tenham participado de sua preparação”.*

E esses foram os itens abordados pelo nosso artigo. **T**

Acronimos

ANSI: American National Standards

Institute: Instituto Nacional Americano de Padrões

ISO: International Organization for Standardization: Organização Internacional para Padronização

JIS: Japanese Industrial Standard: Padrão Industrial Japonês

NSPE: National Society of Professional Engineers: Sociedade Nacional de Engenheiros Profissionais

OBTW: Oh, By The Way: Oh, por falar nisso...

ASAP: As Soon as Possible: O mais rápido possível

UPC: Universal Product Code: Código Universal de Produto

IEC: International Electrotechnical Commission: Comissão Eletrotécnica Internacional

IEEE: Institute of Electrical and Electronics Engineers: Instituto de Engenheiros Elétricos e Eletrônicos

III: Institute for Information Industry: Instituto para Indústria da Informação

ITI: Industrial Technology Institute: Instituto de Tecnologia Industrial.

Vocabulário

Confuse: confunde (to confuse = confundir)

Almost always: quase sempre

Data sheets: folha de dados

Responsible: responsável

Advertising: anúncio

Purpose: finalidade

Ads: anúncios

Reinforce: reforçar

Awareness: desprevenido

Emphasize: enfatizar

Able: capaz

Impress: impressionar

Guias Mangás

Cada título desta coleção de sucesso foi produzido combinando a ilustração de mangás com conhecimentos científicos e matemáticos, tornando a leitura e o estudo técnico divertido e agradável.




novasaber
www.novasaber.com.br

Ensinando FFT na prática, com o NI LabVIEW

Explicar o uso do domínio da frequência para estudantes pode ser um desafio, especialmente quando eles estão acostumados com sinais gráficos em função do tempo, como, por exemplo, no caso da temperatura. O LabVIEW, por conectar diretamente o hardware e adquirir um sinal do mundo real e ainda realizar uma FFT, permite aos estudantes verem o sinal simultaneamente nos domínios do tempo e da frequência. Fazendo isto, eles conseguem visualizar o sinal em ambos os domínios e compará-los em tempo real, permitindo-lhes entenderem intuitivamente as aplicações de cada um.

Este artigo explicará como ensinar medições do domínio da frequência, usando o NI LabVIEW e o NI myDAQ.

Guilherme Kenji Yamamoto
Renan Airoso Machado de Azevedo
National Instruments

A representação do domínio do tempo dá a amplitude do sinal no instante de tempo que você escolheu. Entretanto, em muitos casos é preciso saber o conteúdo da frequência de um sinal, em vez de um único valor da amplitude.

A transformada rápida de Fourier (FFT) fornece um método para analisar o domínio da frequência. O teorema de Fourier estabelece que qualquer forma de onda no domínio do tempo pode ser representada pela soma ponderada de senos e cossenos. A forma de onda, então, pode ser representada no domínio da frequência como um par de valores de amplitude e fase de cada componente da frequência. Veja a **figura 1**.

No domínio da frequência, você pode separar conceitualmente as senoides que formam o sinal. A figura 1 mostra componentes simples da frequência espalhados no domínio do tempo, como impulsos distintos no domínio da frequência. A amplitude de cada frequência é a mesma amplitude da forma de onda em função do tempo para cada componente da frequência. A representação do sinal em termos dos seus componentes individuais é a representação do domínio da frequência do sinal. A representação do domínio da frequência fornece mais detalhes sobre o sinal e o sistema a partir do qual ela foi gerada.

Por que isto é importante?

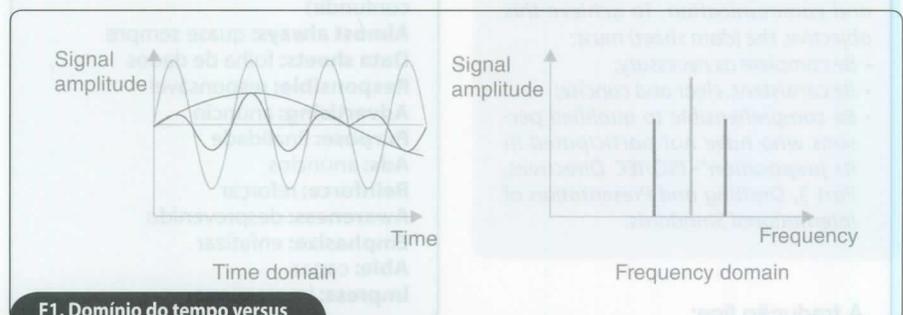
As amostras de um sinal obtidas a partir de um dispositivo de aquisição de dados constituem a representação no domínio do tempo do sinal. Algumas medições, como ruídos ou distorção harmônica, são difíceis de quantificar apenas inspecionando a forma de onda do tempo. Quando o mesmo sinal é mostrado no domínio da frequência por uma FFT, você consegue facilmente medir as frequências harmônicas e as amplitudes.

Se você possui um ruído no seu sinal, você pode movê-lo do domínio do tempo ao domínio da frequência para isolar o distúrbio na sua medição.

Medindo um sinal

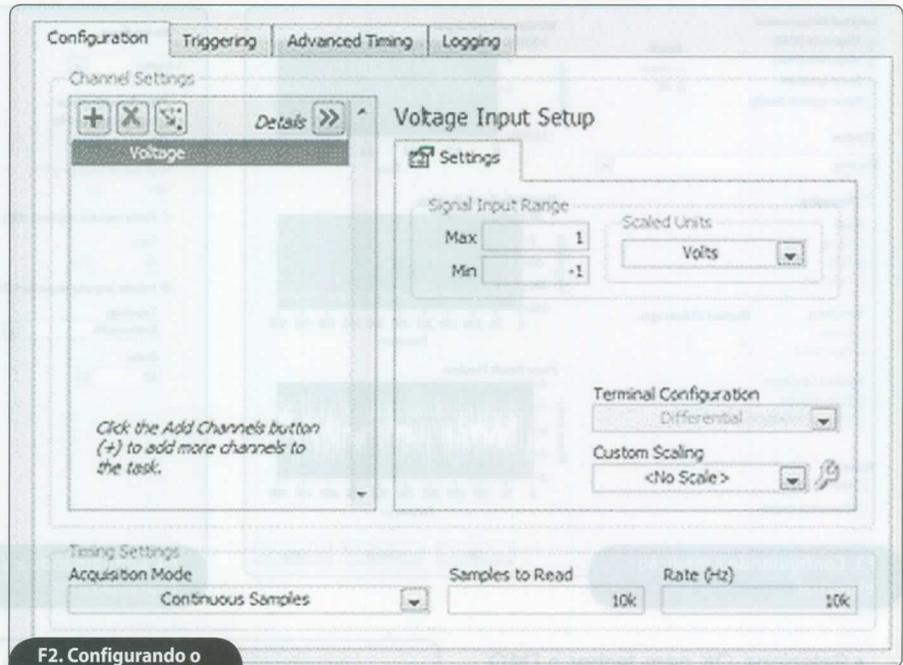
Usando o LabVIEW e os dispositivos de aquisição de dados, você pode adquirir dados e apresentá-los na sua interface de usuário em minutos, permitindo-lhe analisar e explorar o sinal medido. Usando o NI myDAQ, nós vamos configurar o dispositivo no PC e iremos adquirir um sinal analógico.

- Conectando o NI myDAQ ou um dispositivo de aquisição de dados ao seu PC;
- Conecte o sinal ao canal analógico 0 (ai0);
- Crie uma nova VI no LabVIEW:
- Abra o LabVIEW e selecione "File»New VI";
- Coloque um DAQ Assistant no diagrama de blocos;



F1. Domínio do tempo versus domínio da frequência.

- Dê um clique com o botão direito no diagrama de blocos e selecione "Express»Input»DAQ Assistant";
- Coloque o DAQ Assistant no diagrama de blocos, dando um clique com o botão esquerdo;
- Configure o DAQ Assistant Type;
- Selecione *Acquire Signals*;
- Selecione *Analog Input*;
- Selecione *Voltage*;
- Selecione *Dev1* (ou o nome do seu dispositivo caso ele não seja *Dev1*);
- Selecione *ai0* para a entrada analógica no canal zero;
- Configure o Acquisition Settings;
- Entre com 10k para *Number of Samples*;
- Entre com 10k para *Sample Rate*;
- Clique em *Run* para testar as configurações;
- Clique em *OK* para fechar o DAQ Assistant;
- Crie um indicador gráfico para ver os dados;
- Dê um clique com o botão direito no terminal de saída *Data* do DAQ Assistant;



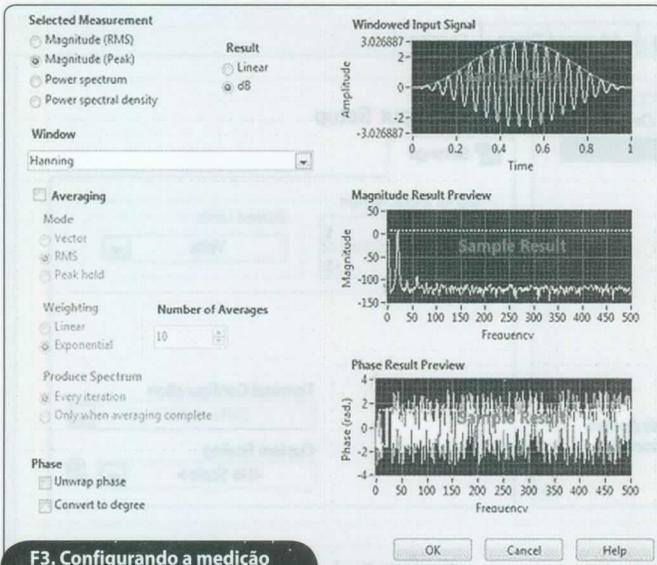
F2. Configurando o DAQ Assistant.

- Selecione *Create»Graph Indicator*;
- Rode a VI pressionando o botão *Run* e veja os resultados;
- Ajuste a aparência do *Graph Indicator* como desejar;

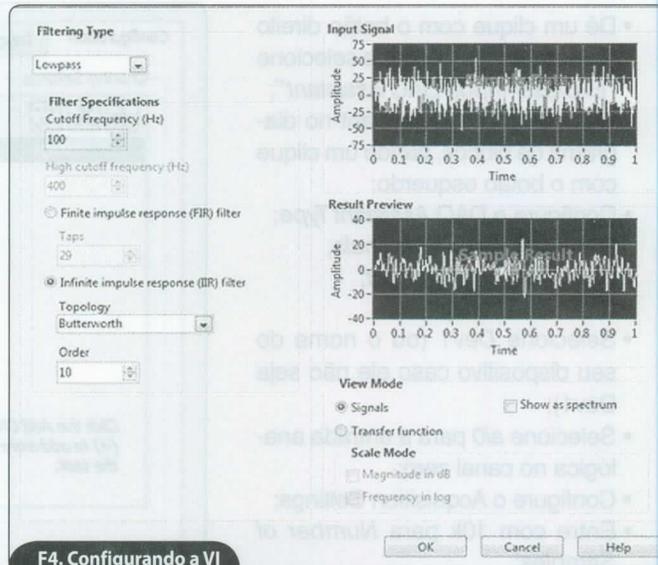
- Mude o modo de aquisição para contínuo;
- Clique duplo no DAQ Assistant;
- Mude o *Acquisition Mode* de *N Samples* (Finito) para *Continuous*;



O que você pode fazer com o Raspberry Pi, um computador de US\$ 35 do tamanho de um cartão de crédito? Todo tipo de coisas! Nesse livro você tem os primeiros passos, a linguagem Python, animação e multimídia, instalação do Arduino e muito mais. Se você está aprendendo a programar ou procurando construir novos projetos eletrônicos, este guia prático mostrar-lhe-á quão valiosa essa pequena e flexível plataforma pode ser.



F3. Configurando a medição espectral.



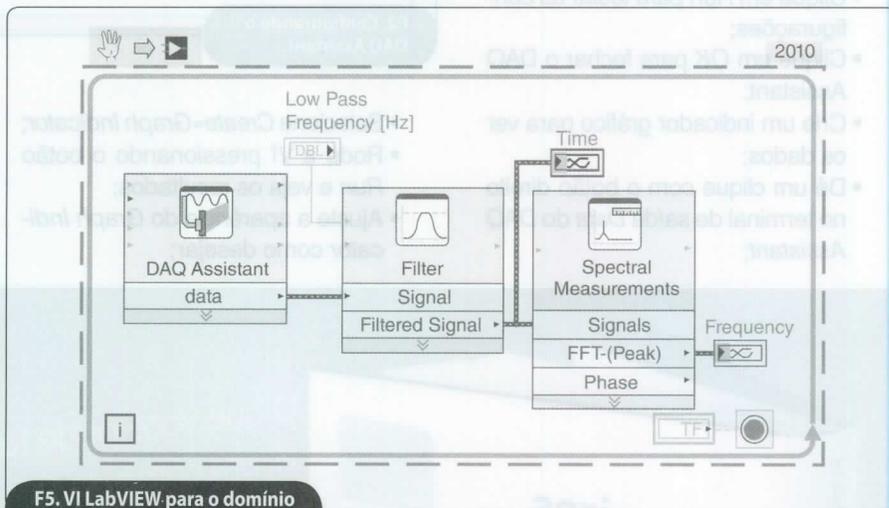
F4. Configurando a VI Expressa Filter.

- Selecione *OK* para fechar o DAQ assistant;
- Selecione *Yes* para colocar automaticamente o *While Loop* em volta do *DAQ Assistant*;
- Rode a VI para visualizar os dados no painel frontal. Observe a **figura 2**.

Converter para o domínio da frequência

Para achar o ruído do sinal medido, nós precisamos ver o sinal no domínio da frequência. Usando a VI Expressa Spectral Measurements, convertemos o sinal do domínio do tempo para o domínio da frequência. Para medir o pico, a VI analisa todas as frequências do sinal e mostra isso em um gráfico.

- Coloque a função Spectral Measurements no diagrama de blocos;
- Crie a função Spectral Measurements indo na paleta de funções e selecionando *Express»Signal Analysis»Spectral*. Coloque a função no diagrama de blocos;
- Na janela *Configure Spectral Measurements* mude o *Selected Measurement* para *Magnitude (Peak)*;
- Conecte as funções e teste a VI;
- Ligue o terminal *data* do DAQ Assistant à entrada *Signals* input do Spectral Measurements;
- Ligue a saída *FFT - (Peak)* da função Spectral Measurements à entrada do terminal *Waveform Graph*;



F5. VI LabVIEW para o domínio da frequência.

- Volte ao painel frontal e clique no botão Run button (a seta branca no canto superior esquerdo). Atente para a **figura 3**.

Filtrar o sinal indesejado

Uma vez que nós identificamos os componentes não desejados do sinal, podemos remover o ruído e manter o sinal que interessa. Um filtro remove os componentes indesejados e não altera a frequência original. Usando a VI Expressa Lowpass Filter, que passa todas as frequências que possuem um valor específico e bloqueia as frequências de valores altos enquanto o LabVIEW é executado, nós podemos interagir com a interface de usuário e modificar os valores da frequência

de corte durante o intervalo em que o sinal é adquirido.

- Coloque um filtro no diagrama de blocos;
- Retorne ao diagrama de blocos e adicione um filtro, indo em *Express»Signal Analysis»Filter*;
- Deixe o *Filtering Type* com a opção padrão, que é *Lowpass*, mas mude a *Cutoff Frequency (Hz)* para 400. Acompanhe nas **figuras 4 e 5**;
- Dê um clique com o botão direito em VI Snippet e selecione *"Save Image As..."* Localize o diretório no seu disco rígido, clique e arraste o ícone para dentro do diagrama de blocos do LabVIEW. Este irá gerar automaticamente o código para a VI Snippet.

Filtro de linha

Uma grande parte dos equipamentos elétricos e eletrônicos conectados à rede de energia faz uso de tecnologias de comutação de alta velocidade, ou mesmo de simples escovas que geram um elevado nível de interferência. Essa interferência se propaga pela rede de energia e pode causar sérios incômodos para os usuários de outros aparelhos.

Como evitar essa interferência é o que veremos neste artigo em que ensinamos o leitor a montar um filtro de linha.

Newton C. Braga

A energia que recebemos pela rede elétrica não é tão limpa quanto parece, já tivemos oportunidade de analisar este problema em muitos artigos. A forma senoidal da tensão de uma tomada que deveria ser pura está deformada pela presença de harmônicas, surtos e transientes.

Surto e transientes são pulsos na forma de trens, ou individuais, que podem atingir em certos momentos milhares de volts. As harmônicas são sinais superpostos à tensão com frequências que podem chegar a centenas de megahertz afetando receptores de rádio, intercomunicadores sem fio, televisores e muitos outros equipamentos que fazem uso de ondas eletromagnéticas.

A maioria dos problemas de uma energia “suja” é produzida pelos próprios aparelhos que são alimentados pela rede. Motores elétricos, equipamentos que fazem uso de fontes

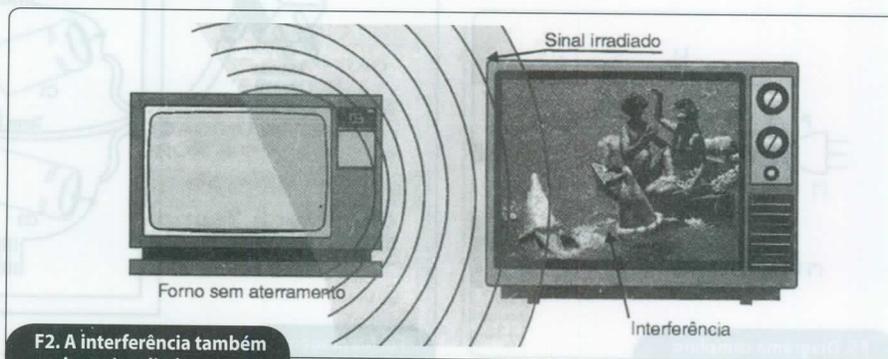
chaveadas, ou ainda controles de potência com SCRs e TRIACs geram uma grande quantidade de harmônicas que se propagam pela própria rede de energia afetando outros equipamentos a ela conectados, conforme mostra a **figura 1**.

Estas harmônicas que se propagam pela linha podem ter os seguintes efeitos sobre os equipamentos alimentados:

- Causam interferências em equipamentos que operam com ondas eletromagnéticas como, por exemplo, receptores de rádio, TV, intercomunicadores sem fio etc.;
- Podem afetar o próprio funcionamento de equipamentos sensíveis como instrumentos, computadores etc.;
- Podem afetar o consumo de um equipamento provocando perdas, ou ainda sobrecargas e aquecimento excessivo.



F1. Eletrodoméstico com motor.



F2. A interferência também pode ser irradiada.

Lista de Materiais

Capacitores:

C_1 a C_4 - 47 ou 100 nF x 250 V - capacitores supressores de transientes (ver texto).

Diversos:

L_1 , L_2 - Bobinas em bastões de ferrite (ver texto).

X_1 , X_2 - Plugue e tomada de acordo com a rede que alimenta o aparelho e o plugue usado no aparelho.

T - Conexão de terra do tipo com terminal de parafuso.

Caixa para montagem, fios, solda, cabo de alimentação, etc.

Lembramos que as interferências causadas por equipamentos que possuem dispositivos de comutação rápida não se propagam apenas pela rede de energia. Em alguns casos, elas podem ser irradiadas, conforme ilustra a **figura 2**.

Para o caso em que as interferências se propagam pela rede de energia, a solução recomendada é o uso de um filtro que pode ser intercalado entre o equipamento interferente e a rede de energia, ou entre o equipamento interferido e a rede de energia, veja na **figura 3**.

Nosso filtro

Se, quando alguém usa um barbeador elétrico no banheiro vizinho ou liga um liquidificador (ou outro eletrodoméstico), a imagem de seu televisor se enche de risquinhos ou mesmo reproduz o ruído do aparelho, conforme exhibe a **figura 4**, é sinal de que você precisa de um filtro de linha.

Descrevemos neste artigo um filtro simples que pode ser usado com eletrodomésticos de até 1000 W de consumo, tanto na rede de 110 V como na de 220 V.

Observamos que para o bom funcionamento deste filtro a ligação de um fio-terra é importante, se bem que nos casos em que a interferência não seja tão severa ele pode ainda funcionar razoavelmente sem ela.

Na **figura 5** temos o diagrama completo de nosso filtro.

A montagem pode ser feita numa pequena caixa de metal como a vista na **figura 6**.

Os capacitores devem ser de poliéster com tensão de isolamento de pelo menos 250 V, se a rede for de 110 V, ou de 400 V se a rede for de 220

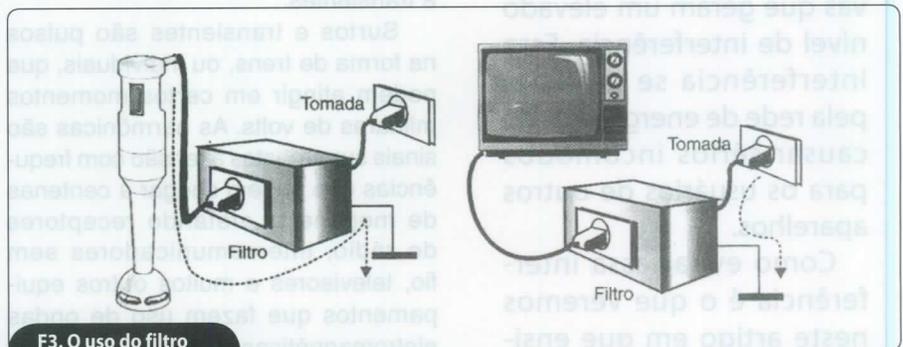
V. Os tipos especialmente indicados como supressores de interferências, tais quais os da série 330 da Philips Components, são recomendados para esta aplicação.

A ligação à terra deve ser feita com fio grosso para que haja a menor resistência possível para o sinal. Esta ligação será feita numa barra de metal enterrada profundamente no solo (como as usadas nos sistemas de aterramento doméstico). Eventualmente, poderá ser usado o polo neutro da tomada se houver dificuldades para um aterramento separado.

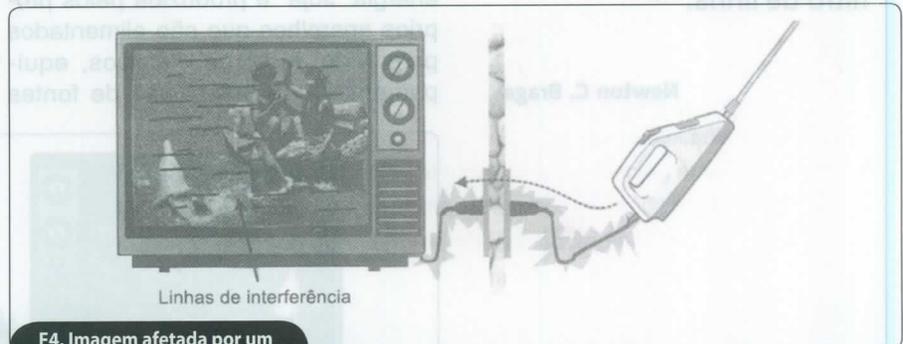
As bobinas são formadas por 30 a 50 espiras de fio comum 18 ou 20, num bastão de ferrite de 0,8 a 1 cm de diâmetro com comprimento entre 12 e 20 cm.

Usando o filtro

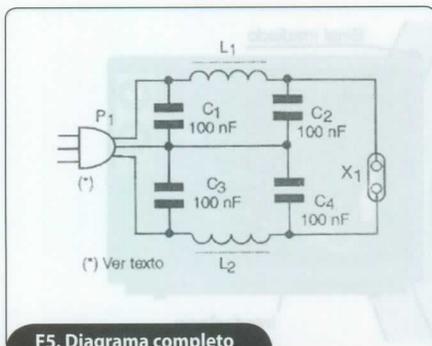
Para usar o filtro basta intercalá-lo entre o aparelho interferido e a tomada, ou entre o aparelho interferente e a tomada. Se a interferência persistir, é preciso verificar se ela não está se propagando pelo ar. Neste caso, o aparelho interferente deve ter sua carcaça aterrada para evitar a irradiação dos sinais.



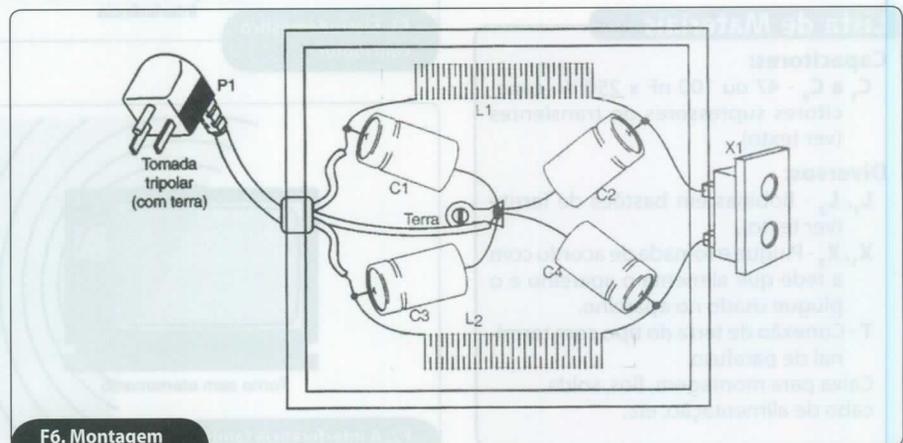
F3. O uso do filtro de linha.



F4. Imagem afetada por um eletrodoméstico.



F5. Diagrama completo do filtro.



F6. Montagem do filtro.

Switches On-Off com o Arduino

Apresentaremos 4 switches ON-OFF usando o sistema Arduino. Esse tipo de switch é construído a partir de pulsadores, que se tornam úteis quando são usados com teclados para controlar sistemas eletrônicos, como em automatizações, controle de equipamentos, etc. Alguns exemplos práticos que utilizam estes switches são: elevadores, mp3, controle remoto, alarmes, entre outros.

Hector Alfonso Perez Mateus
alfaelectronica@hotmail.com
Tradução: Rafaela Turiani

Uma das primeiras coisas que um programador de microcontroladores precisa aprender é a utilização de pulsadores (*switches*) para o controle dos circuitos eletrônicos. Eles são mais utilizados que os interruptores de duas posições por serem mais práticos. Para desenvolver o circuito, foi empregada a plataforma Arduino, que está ganhando popularidade graças ao seu baixo custo e facilidade de uso.

Os projetos que usam este sistema Arduino possuem duas rotinas (ou funções principais) para controlar o programa que é executado no microcontrolador: a função “**setup()**” e a função “**loop()**”.

A função “**setup()**” tem a seguinte semântica:

```
void setup()
{
  // put your setup code here, to run once:
}
```

Essa função é executada somente uma vez no início do programa, por isso, é o lugar onde se configuram as entradas e saídas do microcontrolador. Aqui também se configuram os periféricos especiais, como as portas, o PWM, o *timer*, etc.

A função “**loop()**” tem a seguinte semântica:

```
void loop()
{
  // put your main code here, to run repeatedly:
}
```

Essa função repete-se indefinidamente, e, assim, forma-se o programa

principal a ser executado dentro do microcontrolador. Normalmente, os sistemas de desenvolvimento com os microcontroladores que usam a linguagem C começam com a função chamada “**main()**”. Nessa função, é implementada uma rotina de inicialização dos periféricos (hardware, entradas e saídas, *timers*, conversores analógico-digitais, etc.) e um *loop* infinito no qual se executa o programa principal, visto na **figura 1**. O sistema de desenvolvimento com o Arduino já implementa essas funções para o programador nestas duas rotinas, o que é muito interessante, pois economiza tempo no momento de programar.

Outra característica importante do sistema de desenvolvimento com Arduino é que ele possui muitas bibliotecas prontas para serem usadas pelo programador, o que economiza tempo de programação. Além disso, seu sistema modular permite o desenvolvimento rápido de protótipos, pois já existe uma grande quantidade de periféricos e circuitos compatíveis com o Arduino no mercado.

A **figura 2** mostra a interface inicial de desenvolvimento que o Arduino usa para programar. Já a **figura 3** ilustra os vários exemplos de bibliotecas (prontas para serem utilizadas) do sistema de desenvolvimento Arduino, pré-instalado na interface. Algumas dessas bibliotecas são: Digital, Analógico, Comunicadores, Sensores, Control, Display, Strings (sequência de caracteres), USB, Memória Ethernet, GSM, WiFi, SPI, Robots, Motores de passo, Cartões de memória SD, Servomotores etc.

O programa

Para desenvolver este exemplo, usaremos 4 entradas digitais do microcon-

trolador para os pulsadores e 4 saídas digitais para os LEDs (figura 4). As entradas que conectam os pulsadores possuem resistores de 47 ohms para evitar a possibilidade de programar estes pinos como saídas, e, caso o pino esteja em nível alto (5 volts) e o pulsador seja pressionado, impedir que o mesmo seja queimado ou danificado.

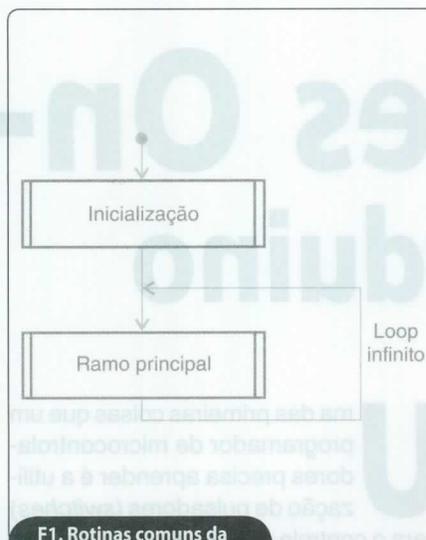
Cada LED tem um resistor de 470 ohms para limitar a corrente, pois os LEDs apenas necessitam de uns poucos miliampères para acender. O código a seguir mostra os pinos usados pelo microcontrolador:

```
const int _1_buttonPin = 2;
const int _2_buttonPin = 3;
const int _3_buttonPin = 4;
const int _4_buttonPin = 5;
const int _1_ledPin = 6;
const int _2_ledPin = 7;
const int _3_ledPin = 8;
const int _4_ledPin = 9;
const int systemLedPin = 13;
```

A instrução “**const**” diz ao compilador que este valor não é alterado ao executar o programa. Observe que o pino 13 é usado para acender e apagar (pisca, ou *to blink* em inglês) um LED, que a maioria dos circuitos com Arduino já tem conectado.

Na função “**setup()**” configuramos os pinos 2, 3, 4 e 5 como entradas, e os pinos 6, 7, 8 e 9 como saídas. Veja o código a seguir:

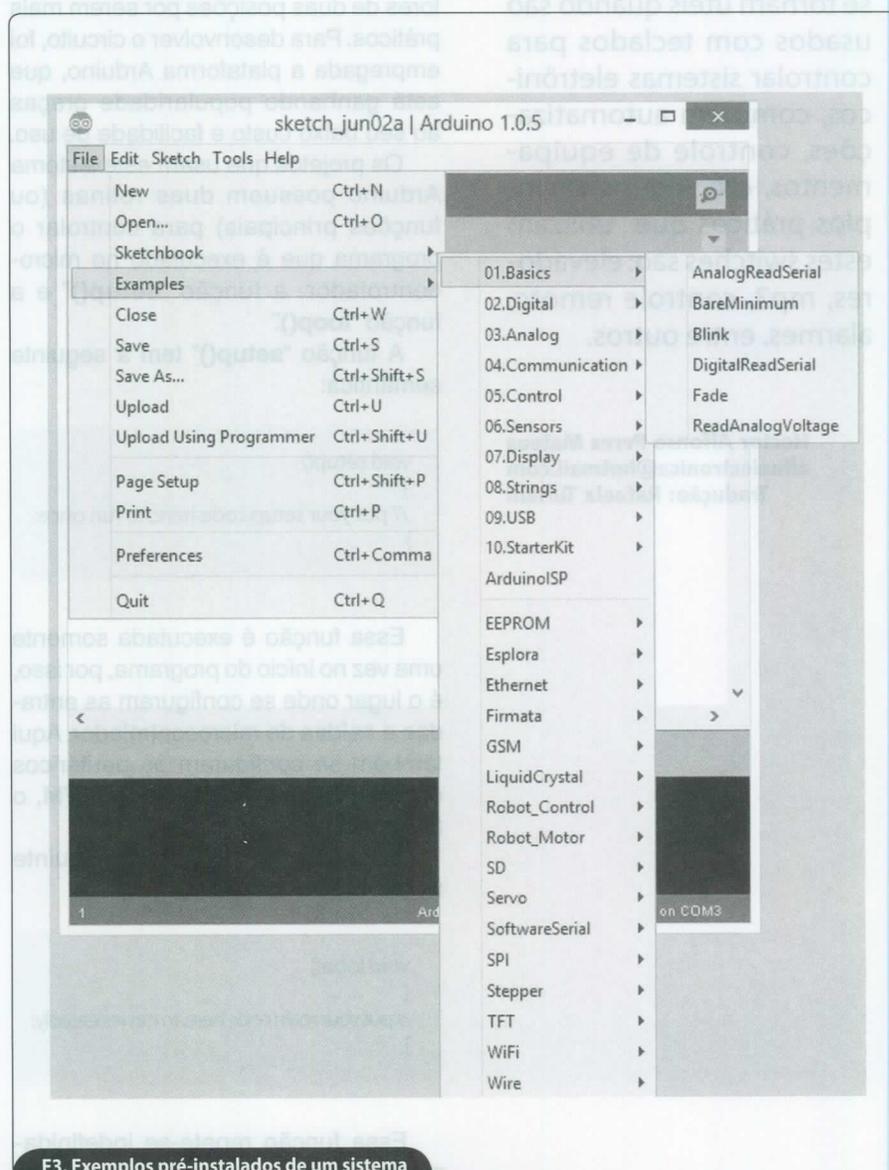
```
void setup()
{
  // put your setup code here, to run once:
  pinMode(systemLedPin, OUTPUT);
  pinMode(_1_buttonPin, INPUT_PULLUP);
  pinMode(_2_buttonPin, INPUT_PULLUP);
  pinMode(_3_buttonPin, INPUT_PULLUP);
  pinMode(_4_buttonPin, INPUT_PULLUP);
  pinMode(_1_ledPin, OUTPUT);
  pinMode(_2_ledPin, OUTPUT);
  pinMode(_3_ledPin, OUTPUT);
  pinMode(_4_ledPin, OUTPUT);
  digitalWrite(_1_ledPin, LOW);
  digitalWrite(_2_ledPin, LOW);
  digitalWrite(_3_ledPin, LOW);
  digitalWrite(_4_ledPin, LOW);
  _1_ledState = LED_OFF;
  _2_ledState = LED_OFF;
  _3_ledState = LED_OFF;
  _4_ledState = LED_OFF;
}
```



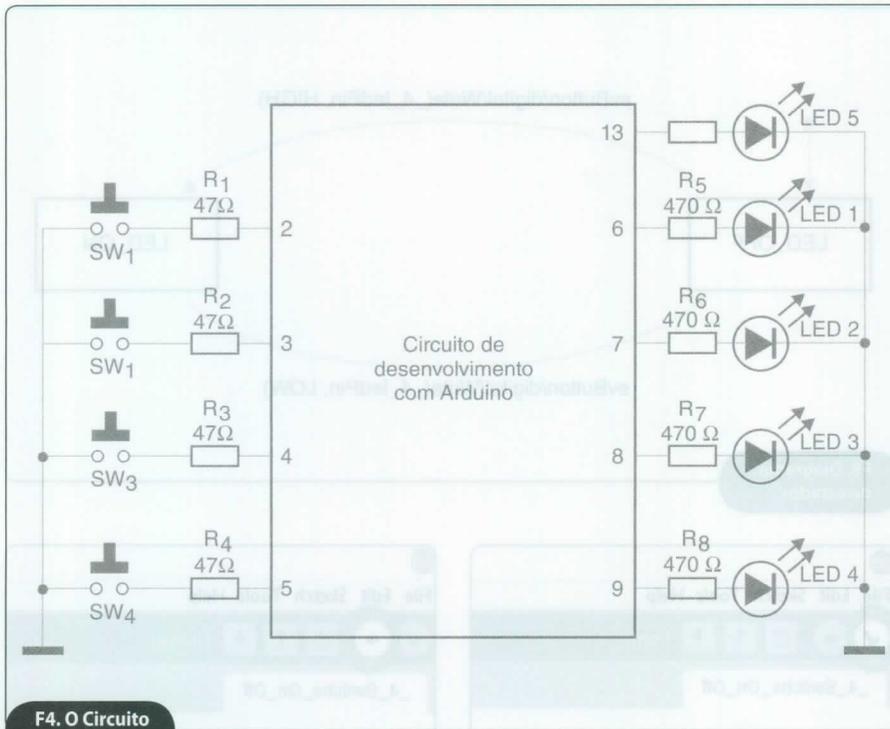
F1. Rotinas comuns da função “main()”.



F2. Interface inicial de desenvolvimento do sistema Arduino.



F3. Exemplos pré-instalados de um sistema de desenvolvimento Arduino.



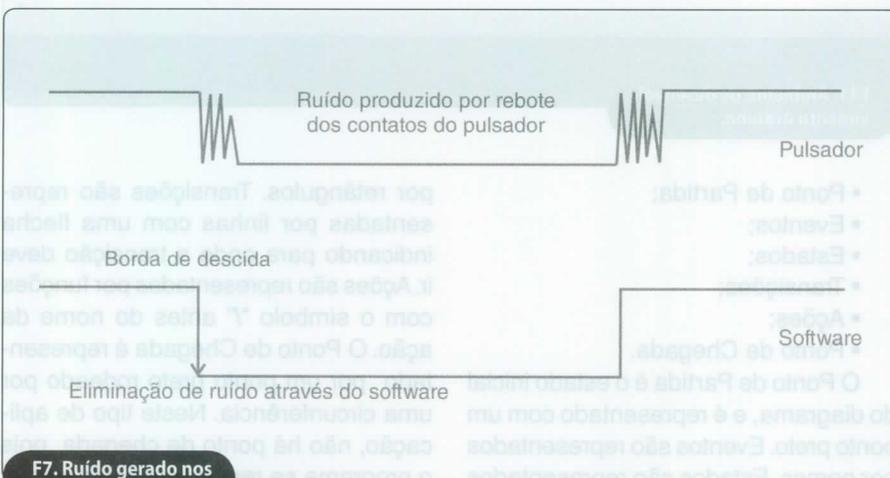
F4. O Circuito utilizado.



F5. Pino de entrada com resistor pull-up.



F6. Pino de entrada sem o resistor pull-up.



F7. Ruído gerado nos pulsadores.

A função usada para configurar um pino é **pinMode()**. Observe que a função recebe dois parâmetros. O primeiro é para indicar o número do pino, e o segundo parâmetro indica se irá ser entrada (**INPUT**) ou saída (**OUTPUT**). Para configurar as entradas existem duas opções: **INPUT**, a qual não seleciona um resistor interno que o pino tem conectado a Vcc (**figura 5**), e **INPUT_PULLUP** que seleciona esse resistor (**figura 6**).

As entradas em qualquer circuito no projeto com microcontroladores devem ser polarizadas com Vcc (positivo) e GND (negativo). Isto também se aplica aos circuitos digitais para evitar leituras erradas e nos pinos de entrada.

A função **main()** se encarrega de ler os pulsadores através da função **Event_Button_HighToLowChanged()**, na qual dá um retorno de verdadeiro (*true*), caso o respectivo pulsador já tenha sido pressionado. Para a função saber qual pulsador precisa testar, passa-se um parâmetro com o nome do pulsador. Também é passado um parâmetro que representa o tempo que o pulsador foi pressionado pela última vez. Isto também é usado para evitar os rebotes elétricos provocados pelo pulsador (**figura 7**).

Em seguida, veja o código para a função **Event_Button_HighToLowChanged()**:

```
boolean Event_Button_
HighToLowChanged(const int button-
Pin, long *p_lastDebounceTime)
{
  if (LOW == digitalRead(buttonPin))
  {
    if ((millis() - *p_lastDebounceTime) >
    debounceDelay)
    {
      *p_lastDebounceTime = millis();
      return true;
    }
    *p_lastDebounceTime = millis();
  }
  return false;
}
```

O parâmetro que controla o tempo é mostrado como um ponteiro, ***p_lastDebounceTime**, pois é necessário

atualizar esse valor dentro da função. Os ponteiros são representados pelo símbolo “*” para o nome do parâmetro, o que significa que o ponteiro armazena um endereço (*address*) na memória. Para passar o endereço certo na memória do ponteiro, coloca-se o símbolo “&” na frente da variável:

```
if (Event_Button_HighToLowChanged(_1_buttonPin, &_1_button_lastDebounceTime))
```

Um detalhe importante a notar nessa função é que a detecção do instante em que o pulsador é pressionado, se realiza na borda de descida (*edje*):

```
if (LOW == digitalRead(buttonPin))
```

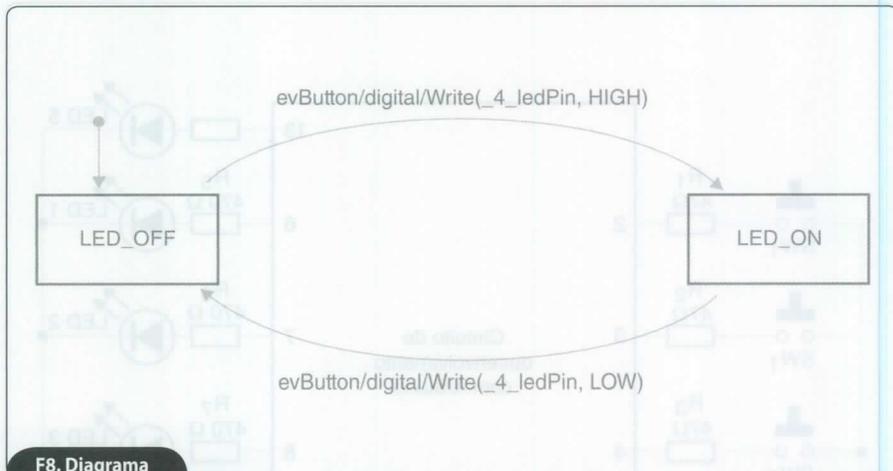
A função “*millis()*” retorna o número de milissegundos, assim que o microcontrolador comece a executar o programa. Este contador voltará a iniciar a contagem a partir de zero (*overflow*) depois de 50 dias, aproximadamente. O código que chama a função para testar os pulsadores, aparece na função “*main()*”, da seguinte maneira:

```
if (Event_Button_HighToLowChanged(_1_buttonPin, &_1_button_lastDebounceTime))
{
    _1_Led_StateMachine(evButton);
}
```

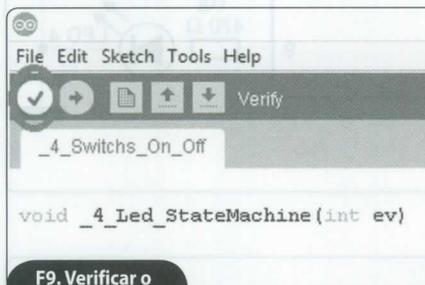
Este código se repete para cada pulsador. Caso seja detectado que o pulsador foi pressionado, a função é chamada de “*_1_Led_StateMachine(evButton);*”, na qual é treinado o controle do respectivo LED.

Máquinas de estados

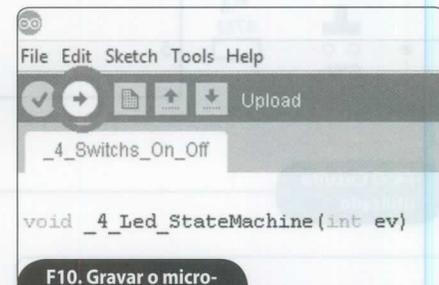
Para controlar os LEDs, foi usada uma máquina de estados. Esse tipo de controle é muito útil, pois facilita o entendimento do código. Máquinas de estado são representadas por diagramas, e, como mostra a **figura 8**, podemos vê-las controlando os LEDs. Basicamente, os diagramas de estados são formado por 6 tipos de itens:



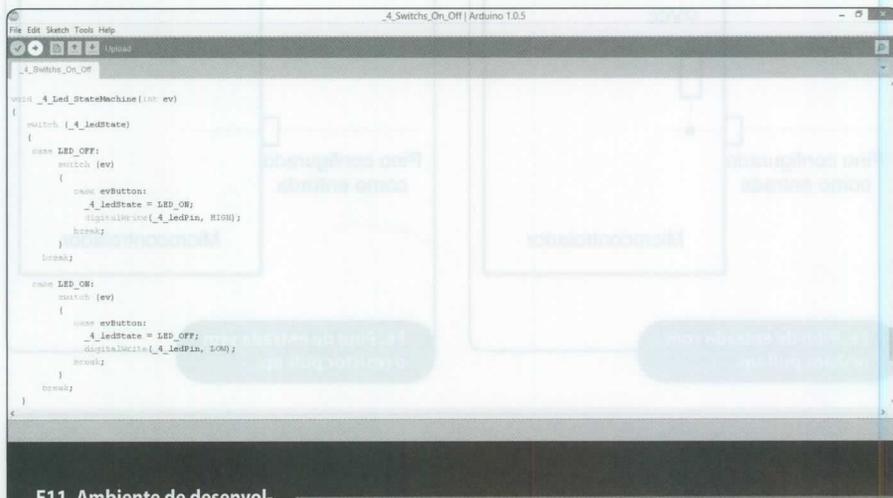
F8. Diagrama de estados.



F9. Verificar o programa.



F10. Gravar o microcontrolador.



F11. Ambiente de desenvolvimento Arduino.

- Ponto de Partida;
- Eventos;
- Estados;
- Transições;
- Ações;
- Ponto de Chegada.

O Ponto de Partida é o estado inicial do diagrama, e é representado com um ponto preto. Eventos são representados por nomes. Estados são representados

por retângulos. Transições são representadas por linhas com uma flecha indicando para onde a transição deve ir. Ações são representadas por funções com o símbolo “f” antes do nome da ação. O Ponto de Chegada é representado por um ponto preto rodeado por uma circunferência. Neste tipo de aplicação, não há ponto de chegada, pois o programa se repete indefinidamente.

O código para controlar o LED 1 é representado por:

```
void _1_Led_StateMachine(int ev)
{
  switch (_1_ledState)
  {
    case LED_OFF:
      switch (ev)
      {
        case evButton:
          _1_ledState = LED_ON;
          digitalWrite(_1_ledPin, HIGH);
          break;
      }
      break;

    case LED_ON:
      switch (ev)
      {
        case evButton:
          _1_ledState = LED_OFF;
          digitalWrite(_1_ledPin, LOW);
          break;
      }
      break;
  }
}
```

Para cada LED se usa o mesmo estado de máquina, mas ele pode ser modificado de acordo com as necessidades. Na prática são utilizadas máquinas de estados independentes, pois cada botão cumpre uma função diferente. Note que uma máquina de estados consiste basicamente de instruções “**switch**” que testam o estado.

Para o caso dos LEDs, são usado dois estados:

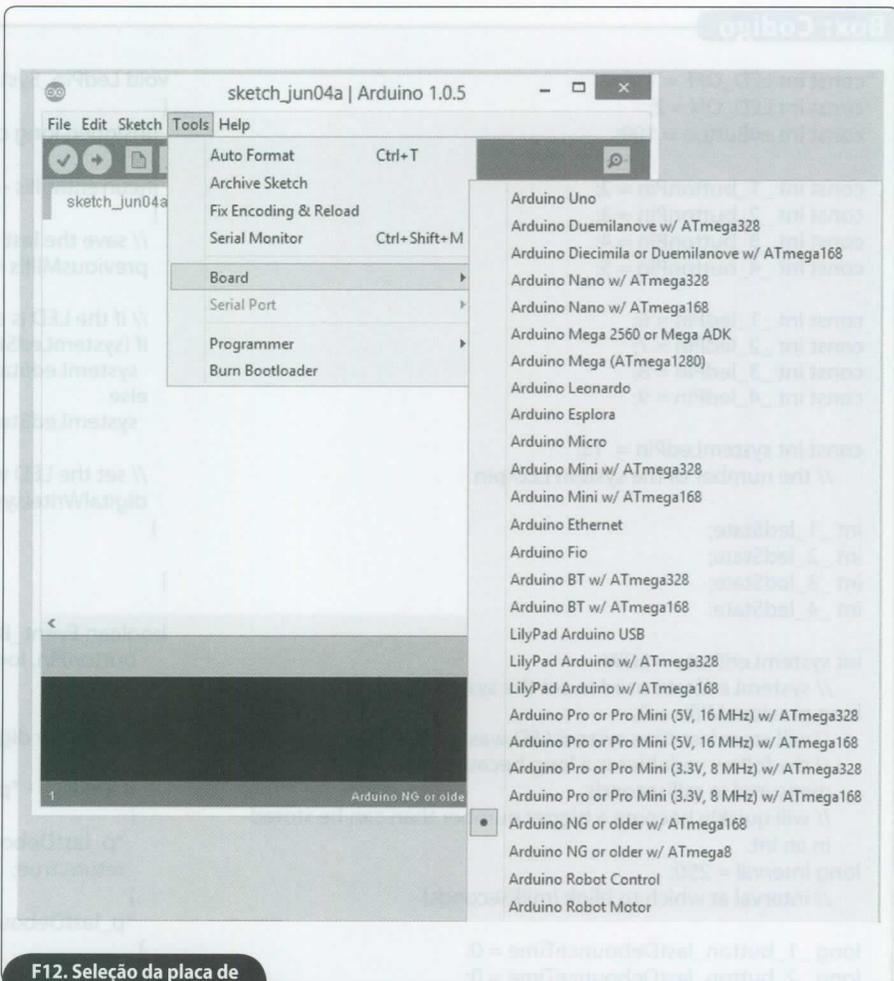
```
LED_ON,
LED_OFF.
```

Existe apenas um único evento chamado “**evButton**”, que é passado como parâmetro à função que tem o estado da máquina. As ações são: acender e apagar o LED e executar com as funções:

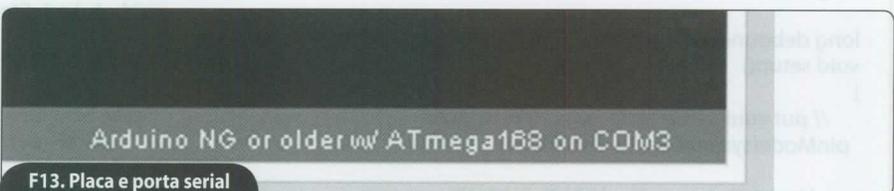
```
“digitalWrite(_1_ledPin, HIGH)”
“digitalWrite(_1_ledPin, LOW)”.
```

O código

O código completo para o microcontrolador encontra-se no **box**.



F12. Seleção da placa de desenvolvimento.



F13. Placa e porta serial selecionados.

Algumas considerações que são feitas ao desenvolver o circuito do programa devem-se ao fato de que o programa deve estar sendo executado sempre.

Por isso, a importância de usar temporizadores dinâmicos para que nunca pare a execução do programa:

```
if(currentMillis - previousMillis >
  interval)
```

Após esse processo, agora é o momento de verificar o programa e

gravá-lo no microcontrolador (*upload*). Para verificar o programa, use o ícone marcado com a cor vermelha na **figura 9**. Para gravar o microcontrolador, use o ícone marcado em vermelho na **figura 10**. Quando se está gravando o programa dentro do microcontrolador, é possível observar essa operação por causa dos LEDs TX e RX do sistema de desenvolvimento Arduino que começam a acender e apagar várias vezes em seguida.

A **figura 11** mostra o ambiente de desenvolvimento (IDE) do sistema Arduino. Ele é formado basicamente pelo

Box: Código

```

const int LED_OFF = 1;
const int LED_ON = 2;
const int evButton = 100;

const int _1_buttonPin = 2;
const int _2_buttonPin = 3;
const int _3_buttonPin = 4;
const int _4_buttonPin = 5;

const int _1_ledPin = 6;
const int _2_ledPin = 7;
const int _3_ledPin = 8;
const int _4_ledPin = 9;

const int systemLedPin = 13;
// the number of the system LED pin

int _1_ledState;
int _2_ledState;
int _3_ledState;
int _4_ledState;

int systemLedState = LOW;
// systemLedState used to set the system LED
long previousMillis = 0;
// will store last time system LED was updated
// the follow variables is a long because the time,
// measured in miliseconds,
// will quickly become a bigger number than can be stored
// in an int.
long interval = 250;
// interval at which to blink (milliseconds)

long _1_button_lastDebounceTime = 0;
long _2_button_lastDebounceTime = 0;
long _3_button_lastDebounceTime = 0;
long _4_button_lastDebounceTime = 0;

long debounceDelay = 50;
void setup()
{
// put your setup code here, to run once:
pinMode(systemLedPin, OUTPUT);

pinMode(_1_buttonPin, INPUT_PULLUP);
pinMode(_2_buttonPin, INPUT_PULLUP);
pinMode(_3_buttonPin, INPUT_PULLUP);
pinMode(_4_buttonPin, INPUT_PULLUP);

pinMode(_1_ledPin, OUTPUT);
pinMode(_2_ledPin, OUTPUT);
pinMode(_3_ledPin, OUTPUT);
pinMode(_4_ledPin, OUTPUT);

digitalWrite(_1_ledPin, LOW);
digitalWrite(_2_ledPin, LOW);
digitalWrite(_3_ledPin, LOW);
digitalWrite(_4_ledPin, LOW);

_1_ledState = LED_OFF;
_2_ledState = LED_OFF;
_3_ledState = LED_OFF;
_4_ledState = LED_OFF;
}

void LedPin_System_Toggle()
{
unsigned long currentMillis = millis();

if(currentMillis - previousMillis > interval)
{
// save the last time you blinked the LED
previousMillis = currentMillis;

// if the LED is off turn it on and vice-versa:
if(systemLedState == LOW)
systemLedState = HIGH;
else
systemLedState = LOW;

// set the LED with the ledState of the variable:
digitalWrite(systemLedPin, systemLedState);
}
}

boolean Event_Button_HighToLowChanged(const int
buttonPin, long *p_lastDebounceTime)
{
if ( LOW == digitalRead(buttonPin) )
{
if ((millis() - *p_lastDebounceTime) > debounceDelay)
{
*p_lastDebounceTime = millis();
return true;
}
*p_lastDebounceTime = millis();
}

return false;
}

void _1_Led_StateMachine(int ev)
{
switch (_1_ledState)
{
case LED_OFF:
switch (ev)
{
case evButton:
_1_ledState = LED_ON;
digitalWrite(_1_ledPin, HIGH);
break;
}
break;

case LED_ON:
switch (ev)
{
case evButton:
_1_ledState = LED_OFF;
digitalWrite(_1_ledPin, LOW);
break;
}
break;
}
}

void _2_Led_StateMachine(int ev)

```

```

{
switch (_2_ledState)
{
case LED_OFF:
switch (ev)
{
case evButton:
_2_ledState = LED_ON;
digitalWrite(_2_ledPin, HIGH);
break;
}
break;
case LED_ON:
switch (ev)
{
case evButton:
_2_ledState = LED_OFF;
digitalWrite(_2_ledPin, LOW);
break;
}
break;
}
}
void _3_Led_StateMachine(int ev)
{
switch (_3_ledState)
{
case LED_OFF:
switch (ev)
{
case evButton:
_3_ledState = LED_ON;
digitalWrite(_3_ledPin, HIGH);
break;
}
break;
case LED_ON:
switch (ev)
{
case evButton:
_3_ledState = LED_OFF;
digitalWrite(_3_ledPin, LOW);
break;
}
break;
}
}
void _4_Led_StateMachine(int ev)
{
switch (_4_ledState)
{
case LED_OFF:

```

```

switch (ev)
{
case evButton:
_4_ledState = LED_ON;
digitalWrite(_4_ledPin, HIGH);
break;
}
break;
case LED_ON:
switch (ev)
{
case evButton:
_4_ledState = LED_OFF;
digitalWrite(_4_ledPin, LOW);
break;
}
break;
}
}
void loop()
{
LedPin_System_Toggle();

if( Event_Button_HighToLowChanged(_1_buttonPin, &_1_
button_lastDebounceTime)
{
_1_Led_StateMachine(evButton);
}
if( Event_Button_HighToLowChanged(_2_buttonPin, &_2_
button_lastDebounceTime)
{
_2_Led_StateMachine(evButton);
}
if( Event_Button_HighToLowChanged(_3_buttonPin, &_3_
button_lastDebounceTime)
{
_3_Led_StateMachine(evButton);
}
if( Event_Button_HighToLowChanged(_4_buttonPin, &_4_
button_lastDebounceTime)
{
_4_Led_StateMachine(evButton);
}
}
}

```

editor de texto que é a janela na qual é escrito o código, sendo que na parte inferior há uma janela de mensagens onde se informa o resultado dos comandos dados ao ambiente de desenvolvimento. Caso tudo ocorra bem, o compilador indicará com uma mensagem que o

microcontrolador foi programado, mostrando também a quantidade de memória usada do programa pelo projeto. Se ocorrer algum erro e não for encontrado o sistema de desenvolvimento Arduino, o compilador avisará (ou reportará) o problema através dessa janela.

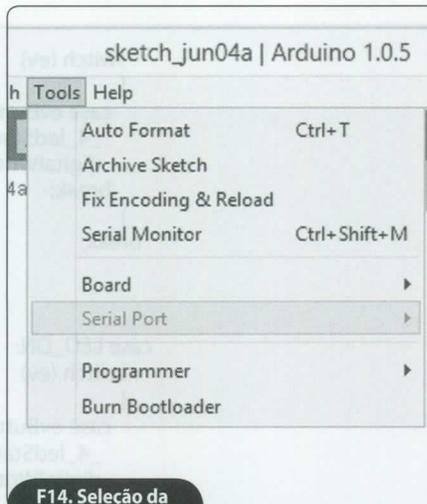
O sistema de desenvolvimento Arduino nomeia de "Sketch" cada arquivo de código que é usado no projeto. No caso deste projeto, só necessitamos de um "sketch". Verifique se a placa (*board*) correta do sistema de desenvolvimento é a que está sendo utilizada. Na **figura 12**

pode-se ver o lugar onde essa seleção é feita. Na parte inferior da direita do IDE, é possível observar essa seleção, como também qual porta serial está sendo usada para se comunicar com o cartão do Arduino (figura 13). A seleção da porta serial pode ser feita no menu “Tools->Serial Port”. Veja figura 14. Caso, ao se instalar o software para a placa Arduino, o sistema operacional não reconheça a porta virtual USB (FTDI), atualize o driver pela internet (em uma versão mais nova).

Uma grande vantagem de aplicar o sistema Arduino é a possibilidade de escrever o código usando a linguagem C/C++. Isso facilita muito mais o desenvolvimento de aplicativos, se for comparado com a linguagem “assembler”. Os itens principais da linguagem C/C++ são as funções e rotinas. Uma função, normalmente, tem a seguinte semântica:

```
void NombreFuncion()
{
}
```

```
int NombreFuncion()
{
}
```



F14. Seleção da porta serial.

Dentro dos “{ }”, escreve-se o código. A função também pode retornar algum tipo de valor. Na função seguinte é necessário retornar um valor inteiro.

Lista de Materiais

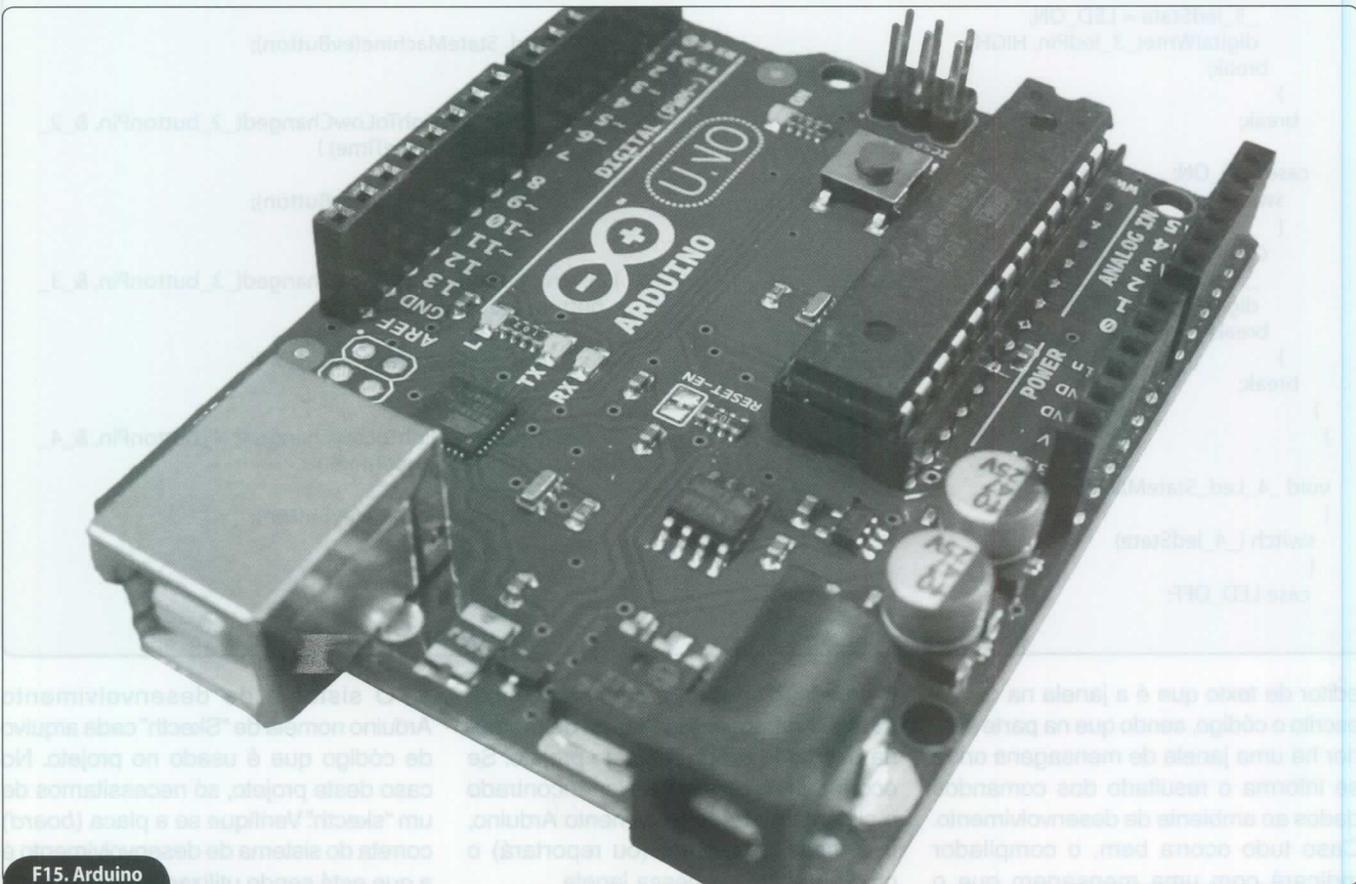
Sistema
Arduino

Resistores
R₁ a R₄ - 47 Ω - ¼ W
R₅ a R₈ - 470 Ω - ¼ W

Semicondutores
LED₁ a LED₄ - LED 5 mm
LED₅ - LED do circuito Arduino

Diversos
SW₁ a SW₄ - Pulsadores

Em edições passadas da revista Saber Eletrônica, foram publicados vários artigos sobre a linguagem C. Visite o portal www.sabereletronica.com.br para saber mais sobre esse assunto. O código-fonte dessa matéria está na seção “download” do portal. Para saber mais sobre o Arduino (Figura 15), acesse: www.arduino.cc.



F15. Arduino Uno.

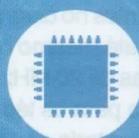


TREINAMENTOS PROFISSIONAIS

Agora a SABER está promovendo treinamentos técnicos, fazendo aquilo que mais sabe fazer que é ensinar. Nosso diferencial é saber muito antes, as novidades tecnológicas e preparar um bom material didático teórico, nunca esquecendo da prática (que é a nossa principal característica), para passar para você, leitor.



Treinamento Básico e Avançado de Programação de Microcontroladores em linguagem C



Treinamento de Programação de Microcontroladores Atmel / Freescale / Holtek / Infineon / Renesas



Treinamento Prático em SMD com Estações de Solda/Retrabalho



Treinamento de Instrumentação e Controle Aplicados em Caldeiras à Óleo

Aulas em três sedes:

SÃO PAULO - Rua Jacinto José de Araújo, 309 - Parque São Jorge - Tatuapé - Próximo à estação do Metrô Carrão

CARAPICUIBA - Av. Rui Barbosa, 540 – Piso 1 sala 40 Centro - Carapicuíba próximo à estação de trem e também em frente de Tamboré e Alphaville.

SÃO BERNARDO DO CAMPO - SP -Rua Ângelo Dusi, 326 - Vila Dusi

CONSULTE A PROGRAMAÇÃO DE 2013

Ligue para: (11) 2095-5330 – Miriam das 10h às 16h.

Mais informações : (11) 2095-5333 - Mari das 10h às 16h.

Usando o osciloscópio no service de TV e vídeo

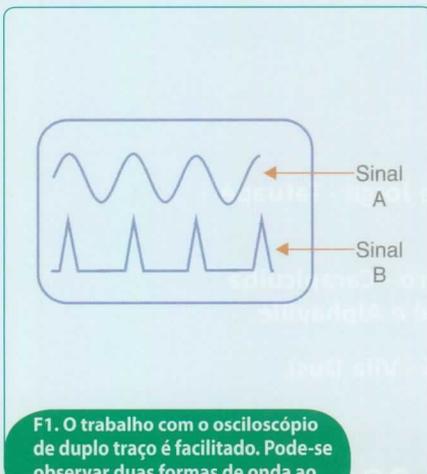
A utilização do osciloscópio na descoberta de problemas em monitores de vídeo, equipamentos de vídeo em geral e televisores, exige o conhecimento de algumas técnicas que nem todos os profissionais dominam. Usado corretamente nesta função, o osciloscópio constitui-se em uma valiosa ferramenta de diagnóstico que não deve ser desprezada por todos aqueles que pretendem ter um trabalho agilizado, maximizando seus lucros.

Osciloscópio, apesar de seu custo ainda algo elevado, é uma ferramenta indispensável em qualquer oficina que trabalha com equipamentos eletrônicos.

Em especial, os equipamentos de vídeo que operam com sinais numa faixa de frequências relativamente alta e com formas de onda complexas podem ser muito melhor analisados com diagnóstico e ajuste mais rápido, se contarem com a presença de um osciloscópio.

Para trabalhos com sinais de vídeo não é preciso contar com um osciloscópio caro, uma vez que as frequências menores das etapas intermediárias são as mais importantes no diagnóstico de problemas. Assim, até mesmo osciloscópio com frequências de 20 MHz podem ser de muita utilidade para os técnicos que souberem aproveitar todo seu potencial.

Se o osciloscópio for de duplo traço, melhor ainda, pois teremos a dupla utilidade da observação simultânea de formas de onda na entrada e saída de uma etapa (ou em dois pontos do mesmo equipamento). Observe a **figura 1**.



F1. O trabalho com o osciloscópio de duplo traço é facilitado. Pode-se observar duas formas de onda ao mesmo tempo.

Formas de onda dos diagramas

Nem todos os técnicos conseguem ter em mãos os diagramas dos equipamentos que reparam com as formas de onda que devem observar em cada etapa.

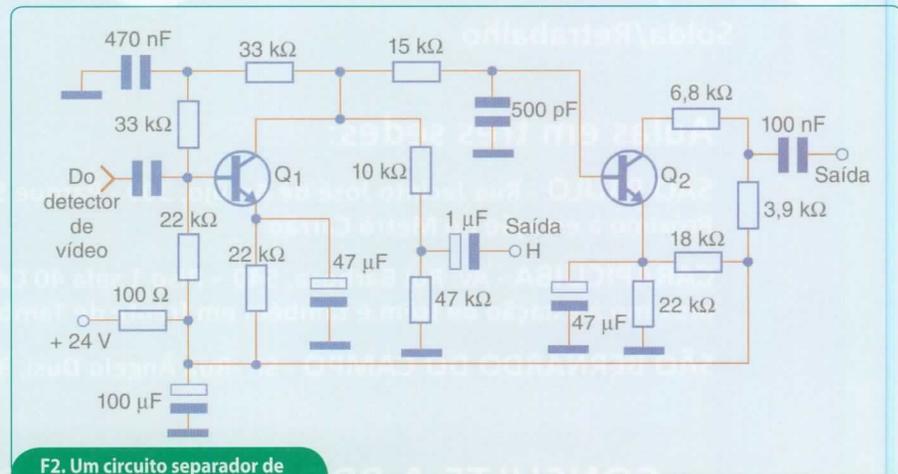
Se bem que na maioria dos casos as formas de onda tenham semelhanças quando tratamos de equipamentos com a mesma finalidade (televisores, por exemplo) é preciso estar atento para situações em que existem peculiaridades de um projeto que mudam tudo.

Em especial, chamamos a atenção para televisores que usam microprocessadores em certas etapas e que injetam sinais com informações digitais que devem ser superpostas à imagem.

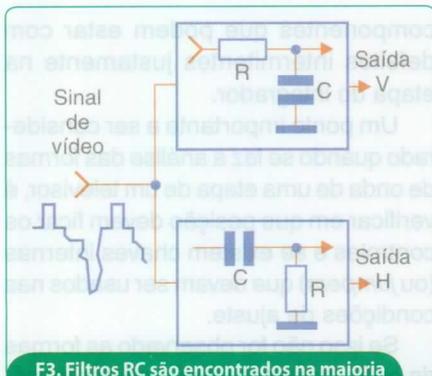
No entanto, vale a descrição geral que possibilita ao técnico avaliar quando um sinal está (ou não) correto num ponto de um circuito.

Verificando o separador de sincronismo

Na **figura 2** temos o circuito típico de um separador de sincronismo de um televisor.



F2. Um circuito separador de sincronismo com transistores.



F3. Filtros RC são encontrados na maioria dos separadores de sincronismo.

O que este circuito faz, é separar o sinal de menor frequência do sincronismo vertical do sinal de frequência muito mais alta do sincronismo horizontal.

O circuito usado nos televisores comuns nada mais é do que um filtro como o mostrado na **figura 3**.

A ligação do osciloscópio no ponto de prova com um sinal de um gerador na entrada deve levar à produção de uma imagem como a vista na **figura 4 (a)**. Lembramos que para a observação destes sinais é preciso ter uma ponta de prova de baixa capacitância.

Se o sinal estiver com a forma dada na **figura 4 (b)**, teremos problemas de funcionamento.

Outra imagem que pode aparecer na observação deste sinal é a mostrada na **figura 5** em que temos a superposição do ruído da rede de energia de 60 Hz.

Outro ponto importante de prova em que se pode observar a forma de onda é no integrador dos pulsos de sincronismo vertical.

Na maioria dos televisores comuns este circuito consiste numa rede RC simples como a presente na **figura 6**.

O uso de uma ponta de prova de baixa capacitância é fundamental para trabalhar com sinais como os encontrados nesta etapa.

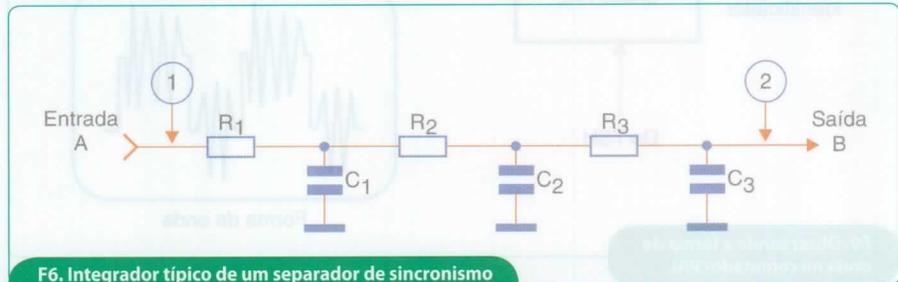
O que este circuito faz, basicamente, é reduzir a intensidade dos sinais de frequências mais altas e manter a intensidade dos sinais de frequências mais baixas. Assim, antes do integrador o sinal de sincronismo horizontal terá uma intensidade maior, sendo atenuado quando o observamos depois.



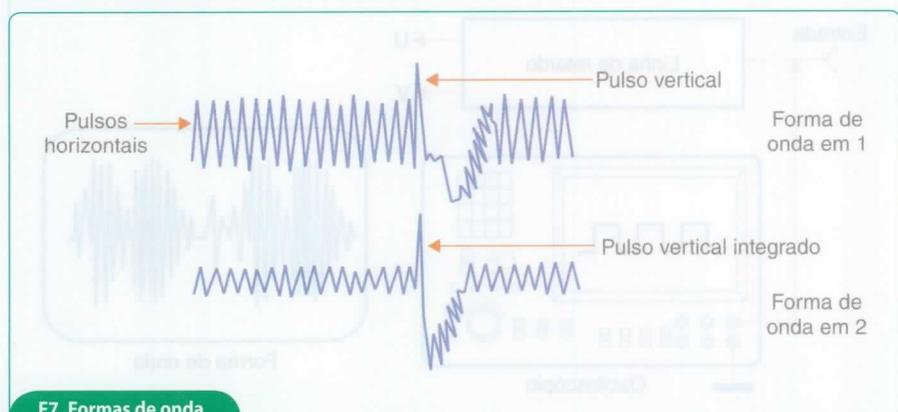
F4. Sinal do separador de sincronismo visto no osciloscópio.



F5. Sinal do separador com forte componente de zumbido de 60 Hz.



F6. Integrador típico de um separador de sincronismo com os pontos de ligação do osciloscópio.



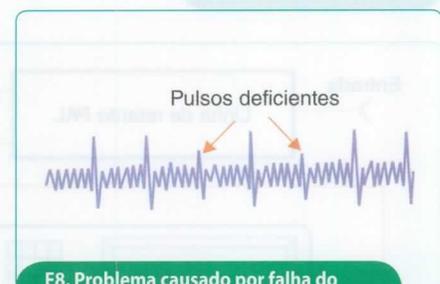
F7. Formas de onda no integrador.

Por outro lado, os pulsos de frequência menor do sincronismo vertical quase não são atenuados, conforme ilustra a **figura 7**.

No ajuste do circuito, ou verificação de uma falha de sincronismo, é importante que o técnico tenha nas folhas de serviço o valor da tensão que o pulso vertical deve ter.

Um capacitor com fugas (ou alterado) pode fazer com que os pulsos tenham sua amplitude modificada e, com isso, surgir um problema de sincronismo no equipamento analisado.

Problema algo comum é a perda de sincronismo vertical de modo aleatório



F8. Problema causado por falha do circuito de separação de sincronismo.

(de vez em quando a imagem rola) o que caracteriza uma instabilidade do circuito.

Analisando a forma de onda de um aparelho com este problema vemos

que a amplitude do pulso vertical varia em determinados instantes, quando o problema acontece (**figura 8**).

Este defeito pode ser causado pelo acoplamento indevido de sinais com a etapa, ou ainda por problemas de

componentes que podem estar com defeitos intermitentes justamente na etapa do integrador.

Um ponto importante a ser considerado quando se faz a análise das formas de onda de uma etapa de um televisor, é verificar em que posição devem ficar os controles e se existem chaves internas (ou *jumpers*) que devam ser usados nas condições de ajuste.

Se isso não for observado as formas de onda que aparecem num osciloscópio podem ser distorcidas, dando a falsa indicação de um problema de funcionamento nas etapas do equipamento.

Em muitos televisores as provas desta etapa são feitas desligando-se o oscilador vertical e, além disso, os controles de sincronismo vertical devem ser colocados em determinadas posições.

Também deve ser considerada a hipótese de que os pulsos do oscilador horizontal, por algum problema de seu circuito, sejam acoplados ao circuito vertical causando assim problemas de funcionamento.

Analizando os circuitos de cor

Usando uma ponta de prova de baixa capacitância e um gerador de barras coloridas PAL-M pode-se analisar a forma de onda do comutador da chave PAL, conforme mostra a **figura 9**.

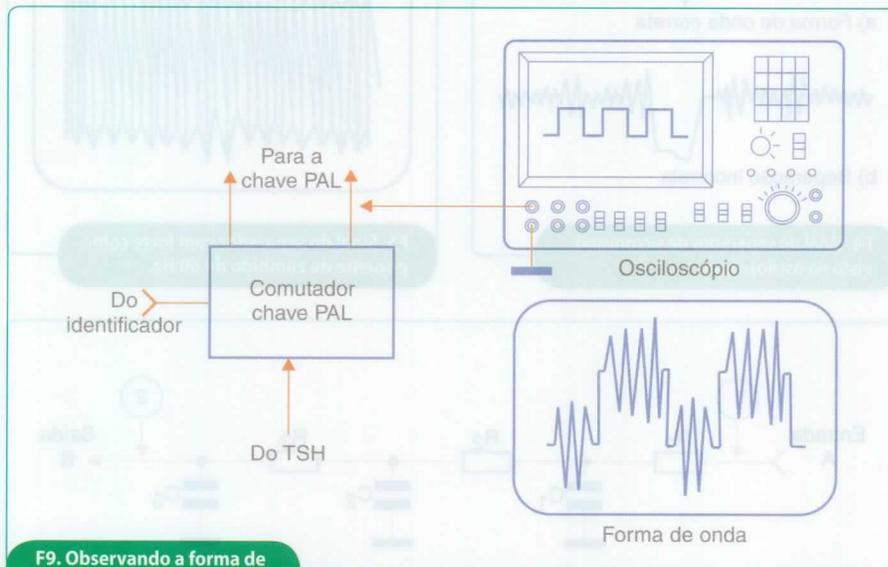
Nesta mesma figura temos a forma de onda que deve ser observada. A prova deve ser feita com um gerador de imagens coloridas e os ajustes devem ser feitos de acordo com o indicado pelo fabricante.

A frequência do sinal deve ser a metade da frequência de varredura horizontal e a amplitude deve ser a indicada pelo fabricante em seu manual.

Para observar a forma de onda na entrada da linha de retardo PAL temos a disposição de ligações dada na **figura 10**.

O gerador de sinais PAL deve ser ajustado para produzir um padrão de barras coloridas normal.

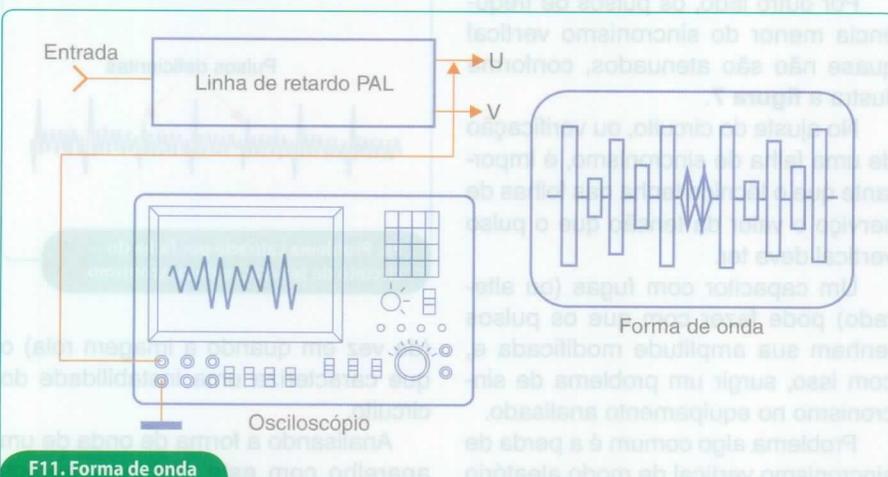
A figura mostra a forma de onda que deve ser observada. As amplitudes dos sinais dependem do aparelho analisado devendo o técnico verificar seus manuais.



F9. Observando a forma de onda no comutador PAL.



F10. Forma de onda na entrada da chave PAL.



F11. Forma de onda na saída U.

Na **figura 11** temos as ligações para se observar o mesmo sinal depois da linha de retardo (no ponto U).

O gerador de padrões PAL deve ser ajustado para produzir o padrão de barras coloridas normal.

Os sinais correspondentes às barras amarelas e azuis devem apresentar, praticamente, a mesma amplitude pico a pico. Os sinais que correspondem às barras ciano e vermelho e para as barras verde e magenta também.

No entanto, os sinais correspondentes às barras amarela e azul são os que têm a menor amplitude, e os sinais das barras ciano e vermelha são os de maior amplitude.

Para observar as formas de onda no ponto V temos as ligações mostradas na **figura 12**, juntamente com a forma de onda.

O gerador de padrões PAL deve estar gerando o padrão de barras normais.

Para um circuito normal, as barras amarela e azul devem estar com amplitudes bem próximas. O mesmo deve ocorrer com as barras ciano e vermelha e para as barras verde e magenta.

Novamente, os valores corretos das amplitudes são os indicados nos manuais de cada equipamento.

Conclusão

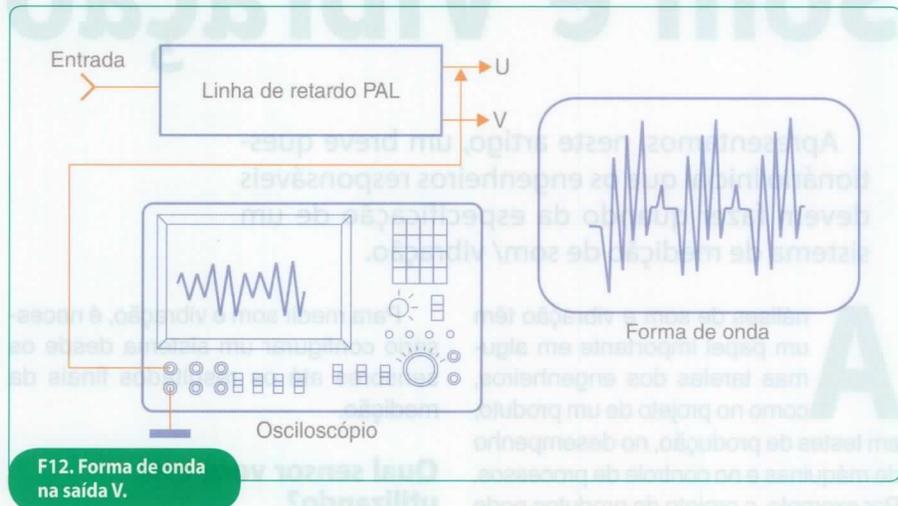
Todas estas provas preveem que o técnico saiba fazer os ajustes básicos de um osciloscópio para observar uma determinada forma de onda.

A simples comparação da forma de onda encontrada num equipamento com a que está especificada no seu

manual pode ser suficiente para se diagnosticar um defeito.

A prática do técnico (que deve vir com o tempo) vai ser muito importante para se saber exatamente em cada caso o que está ruim num circuito.

Pequenas distorções na forma de onda, alterações de amplitude, ou ainda imagens suplementares que indicam ruídos podem levar o técnico experiente exatamente aos componentes responsáveis por um problema. **T**



F12. Forma de onda na saída V.

38 Soluções Completas em Caixas e Gabinetes Técnicos Fabricados em Polímeros de Engenharia

Para outros modelos visite nosso site

Caixas para fontes e filtros de linha • caixas com tampa e abas de fixação • caixas para sensores e iluminação • caixas e acessórios norma din • caixas para sirenes e alarmes • caixas para coletor de dados • caixas para estabilizadores • caixas para sinalizadores • caixas com painel e alça • luminárias para barco • automação de portões • caixas para controles • caixas para reatores • caixas para elétrica • caixas para racks • módulo bcd • telefonia

(11) 2193-7500 | www.patola.com.br
vendas@patola.com.br

PATOLA
NÃO TEM CARA COM QUALIDADE E PREÇO

TATUINO

Programação Super Fácil
Código Aberto
Comunidade na internet
Centenas de projetos prontos
16k de memória
13 Inputs/Outputs
Comunicação USB

TATO
Equipamentos Eletrônicos

www.tato.ind.br

Algumas questões que você deve considerar antes de selecionar seu Sistema de Medição de Som e Vibração

Apresentamos, neste artigo, um breve questionário inicial que os engenheiros responsáveis devem fazer quando da especificação de um sistema de medição de som/ vibração.

Análises de som e vibração têm um papel importante em algumas tarefas dos engenheiros, como no projeto de um produto, em testes de produção, no desempenho de máquinas e no controle de processos. Por exemplo, o projeto de produtos pode necessitar de modificações no nível de som ou ruído para atender padrões como ANSI, IEC, e ISO.

O trabalho pode envolver também projeto de ajuste fino para atender expectativas de marketing. Aqui, os exemplos incluem um mecanismo de travamento da porta de um automóvel para impressionar o consumidor com um simples “click” ou modificar as várias maneiras de mudar o nível de ruído do motor.

Análises de som e vibração também podem aumentar o desempenho e a confiabilidade da máquina. Reduzindo a vibração, você pode obter benefícios como alta velocidade de operação, melhor precisão, vida estendida do componente, redução de efeitos térmicos e economia de energia. Áreas de aplicação podem incluir - acústica, medições de som, análises de vibração, ruído, vibração e medições em ambientes agressivos (NVH), análises de áudio, estruturas dinâmicas, avaliação/medição de qualidade de som, monitoramento de máquinas, manutenção preditiva, e análise de máquinas rotativas.

Para medir som e vibração, é necessário configurar um sistema desde os sensores até os resultados finais da medição.

Qual sensor você está utilizando?

O tipo do sensor que você escolhe depende do sinal que está medindo. Para aplicações de som e vibração, os sinais mais comuns são de aceleração, deslocamento e nível de pressão do som. Os sensores mais comuns para medir estes parâmetros são acelerômetros, sensores de proximidade e microfones, respectivamente.

Você deve primeiro saber o tipo do sensor que pretende usar, pois isto influencia diretamente na escolha do *hardware*. Algumas das questões em que a escolha do seu sensor reflete no *hardware* são as seguintes:

- “Qual é o tipo de condicionamento de sinal necessário?”
- “Qual é o range de frequência do sensor?”
- “Quanto o sensor necessita de range dinâmico?”

Qual condicionamento de sinal é necessário?

Antes de ser digitalizado pelo *hardware* de aquisição de dados, o sinal do sensor quase sempre precisa de

algum tipo de condicionamento como amplificação, filtro, excitação do sensor e configuração de entrada.

Qual é o range de frequência de seu interesse?

Todos os sensores possuem um range de frequência onde cada um foi projetado para operar. Seu sensor deve ter um range de frequência suficiente para abranger a faixa de frequências de seu interesse, da mesma maneira que seu *hardware* digitalizador deve ter um range de frequência suficiente para abranger os sinais de interesse.

Qual é o range dinâmico necessário?

Range dinâmico é a medida de quão pequena é a medição que você pode fazer de um sinal em relação ao máximo valor de entrada que o dispositivo pode medir. Exibido em decibéis, o range dinâmico é $20 \log(V_{\max}/V_{\min})$. Por exemplo, um produto deste mercado é a placa NI-4472 que possui um range de entrada de ± 10 V e um range dinâmico maior do que 110 dB, no qual fornece uma relação de tensão de 10^6 . Assim, com um sinal máximo de 10 V, o menor sinal que você pode ver na NI-4472 é $10 \mu\text{V}$.

O range de entrada e o range dinâmico especificados são importantes para determinar as necessidades de seu sistema.

Guilherme Kenji Yamamoto
Renan Airoso Machado de Azevedo
National Instruments

Você precisa de informação de fase (amostragem simultânea)?

Amostragem simultânea é importante em aplicações onde é necessária a informação de fase entre dois canais medidos. Informação de fase é utilizada em aplicações que necessitam da resposta em frequência de um objeto, ou em aplicações que utilizam medições entre canais como intensidade de som ou análise estrutural. Amostragem simultânea está disponível na maioria dos produtos da NI para medição de som e vibração.

Quanto você precisa de processamento e disco?

“Operação real-time” ou “operação on-line” para sistemas de medição de som e vibração são tipicamente definidas como a capacidade de executar todas as tarefas de análises, visualização e armazenamento desejadas pelo menos tão rápido quanto o dado for adquirido.

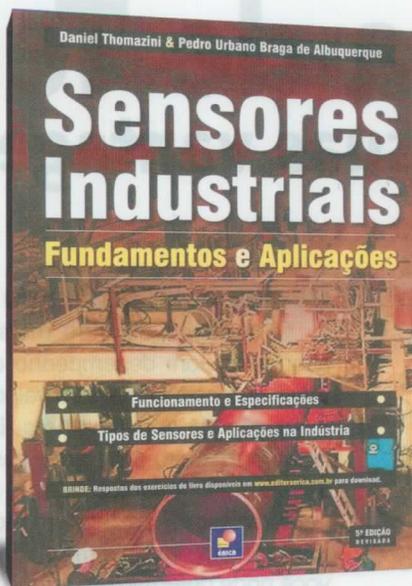
Processamento “gap-free” significa que todos os dados adquiridos são submetidos a análises *on-line*. Em algumas aplicações, análises *gap-free* podem não ser um requisito. Por exemplo, é comum para aplicações com monitoramento de muitos canais, armazenar todos os dados em disco, mas executar somente uma análise rudimentar em uma pequena parte dos dados. Outras análises podem ser executadas de maneira *off-line*, com os dados armazenados.

É importante especificar configurações suficientes de computador e de disco rígido em seu sistema de medição para atingir os requisitos de processamento e armazenamento da aplicação. Alguns dados pertinentes são comentados abaixo, tenha em mente o fato de que estes exemplos não assumem outros requisitos (por exemplo, casos de armazenamento em disco não incluem FFTs *on-line* ou outro processamento), e eles incluem somente o mínimo (tolerância de segurança).

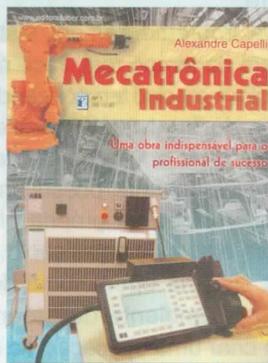
Conclusão

Respondemos, assim, às primeiras seis questões dirigidas aos engenheiros encarregados de especificar um sistema para medições de som e/ou vibração na indústria. **T**

Sensores Industriais Fundamentos e Aplicações



Essa obra é destinada a alunos de graduação em engenharia ou tecnologia, técnicos e engenheiros que atuam em processos de automação de sistemas. Apresenta os tipos de sensores utilizados na indústria e alguns de uso doméstico. Aborda tanto o fundamento físico aplicado a eles quanto algumas de suas aplicações em campo.



Mecatrônica Industrial

A tecnologia de automação integrou-se definitivamente aos processos de produção industrial. Hoje, a necessidade de alta produtividade em conjunto com a "excelência" da qualidade somente pode ser atendida com equipamentos rápidos, precisos e confiáveis.



Injeção eletrônica de motores diesel

O conteúdo dessa obra refere-se aos sistemas de Gerenciamento Eletrônico dos motores Diesel, e basicamente estão apresentados de forma cronológica. Pretende colaborar com os Técnicos que atuam com motores de ciclo Diesel, discorrendo sobre funcionamento, Características, Regulagens, Diagnósticos.



novasaber
www.novasaber.com.br

Entenda de fato o que é um Sistema de Tempo Real e quais são suas vantagens

Muitas aplicações de medição e automação requerem um desempenho de tempo real. Para diferenciar estas aplicações de tempo real, você precisa primeiramente entender o núcleo do desempenho de tempo real e como este é utilizado para controle, processamento de sinais e resposta a eventos em tempo real.

Guilherme Kenji Yamamoto
Renan Airoso Machado de Azevedo
National Instruments

Desempenho de Tempo Real

O conceito errôneo mais comum associado com desempenho de tempo real é que este aumenta a velocidade de execução do seu programa. Enquanto isto é verdadeiro em alguns casos, este, de fato, melhora sua aplicação provendo características de temporização mais precisa e previsível. Com estas melhorias, você pode determinar o tempo exato em que um determinado evento irá ocorrer.

O desempenho de tempo real pode ser atingido através de *hardware* ou *software*.

Para implementar um sistema de hardware de tempo real, você necessita de um *clock* preciso temporizando alguns circuitos do hardware. Por exemplo, o hardware para aquisição de dados, da National Instruments, possui um circuito de temporização na própria placa que configura a taxa de aquisição.

Um sistema operacional de tempo real e um ambiente de programação podem criar uma aplicação de software de tempo real. Esta solução de software oferece mais flexibilidade e ferramentas convenientes para depuração. Com isto, você pode incorporar tomadas de decisão em tempo real em sua aplicação com um algoritmo personalizado ao invés de confiar em circuitos de hardware já existentes.

Com alguns sistemas, você pode conseguir um híbrido, no qual, pode-se ter a temporização precisa de hardware controlado em tempo real combinado com a flexibilidade de software de tempo real.

Controle de Tempo Real

Com um controle de tempo real, você pode continuamente monitorar e simular um sistema físico. Aplicações de controle

de tempo real desempenham repetidamente uma tarefa definida pelo usuário com um intervalo de tempo específico separando-as. Existem diversos sistemas de controle de tempo real ao nosso redor, como o controle de velocidade em um automóvel (*cruise control*), ou o termostato controlando a temperatura em sua casa. Existem diferentes maneiras de medir ou descrever uma aplicação de controle em tempo real – tempo de ciclo de malha de controle, determinismo, *jitter* e *soft real-time*.

Tempo de Ciclo de malha de Controle

Muitos sistemas de controle de tempo real monitoram o sistema físico, comparam o estado atual com o estado desejado e depois simulam o sistema físico baseado nesta comparação. O tempo necessário para executar esta interação é considerado o tempo de ciclo de malha. Este tempo de ciclo do loop de controle varia com base na complexidade do sistema.

Jitter e Determinismo

O determinismo mede a consistência do intervalo de tempo especificado entre os eventos. Muitos algoritmos de controle, como o PID, requerem um grande comportamento altamente determinístico. Por exemplo, um elevador se move gradualmente para o andar correto devido ao comportamento determinístico de malha de controle. Sem o determinismo, o elevador iria atingir o andar correto, porém sem estabilidade.

Como todos os sistemas de tempo real, existem alguns conjuntos de erros chamados *jitter*. O *jitter* é outra maneira de medir o determinismo de um sistema

de tempo real. Você pode calcular o jitter como a máxima diferença entre qualquer tempo de atraso individual e o tempo de atraso desejado em um sistema.

Aplicações de Tempo Real Crítico e Não Crítico

Aplicações de controle de tempo real podem ser mapeadas em um espectro de desempenho.

De um lado, os sistemas críticos de tempo real são muito determinísticos e nunca perdem eventos. Um exemplo crítico de um sistema de tempo real é um dinamômetro de motor. Se um evento foi perdido, os dados coletados ou as condições simuladas da estrada estarão incorretas.

No outro lado do espectro, sistemas de crítico de tempo real não requerem o mesmo grau de determinismo e ocasionalmente perdem eventos sem afetar o funcionamento. Por exemplo, perder apenas um ponto em um sistema de monitoração de temperatura não causa impacto na tendência total porque a tendência geral de temperatura muda lentamente.

Processamento de Sinais de Tempo Real

Processamentos de sinais de tempo real possuem muitas das mesmas características de controle de tempo real. Este requer intervalos de tempo determinísticos entre tarefas repetitivas. Mas ao invés de calcular a resposta, são desempenhados processamentos de sinais nos dados adquiridos.

Um exemplo é como o sistema faz o cálculo da frequência de um sinal. O exemplo adquire deterministicamente uma série de valores que representam uma forma de onda e, depois, obtém o espectro de potência dos dados em uma estrutura de repetição. Enquanto este exemplo analisa a forma de onda em blocos de dados, ele poderia também ter analisado cada valor quando este foi adquirido através de uma rotina de análise "ponto a ponto."

Em muitas aplicações, rotinas de análise ponto a ponto provêm um melhor desempenho. Ao invés de analisar blocos de dados, estas rotinas mantêm uma memória com os dados anteriores e calculam o novo valor de saída baseado no histórico dos dados e no valor

atual. Nestes sistemas é necessário um desempenho de hardware de tempo real porque a perda de valores ou a aquisição dos valores após um pequeno atraso, destrói a integridade dos dados no histórico utilizado para cálculos futuros.

Estas rotinas reduzem o tempo de atraso entre a aquisição dos dados e a geração dos resultados. Por exemplo, uma aplicação simples de filtro digital adquire 1000 pontos de dados por segundo, filtra os dados e gera a forma de onda filtrada. Enquanto a forma de onda filtrada é precisa, é gerado um segundo inteiro após ser inicialmente adquirido, introduzindo um grande tempo de atraso ou diferença de fase. Com um filtro ponto a ponto, a forma de onda é filtrada após cada dado ser adquirido, reduzindo o tempo de atraso para microssegundos.

Filtrando após cada ponto ser adquirido, você também pode aumentar a resolução no processo de tomada de decisão e aumentar o desempenho total do sistema.

Resposta a Eventos de Tempo Real

Com uma resposta a eventos de tempo real, você pode responder a um único evento dentro de um determinado intervalo de tempo. O sistema de tempo real garante alguns tempos máximos de resposta para um único evento. O evento pode ser periódico ou aleatório. Um exemplo de aplicação com resposta a evento em tempo real é um sistema seguro de monitoramento. Se uma planta física entra em um estado de perigo, o sistema de tempo real deve responder ao evento "perigo" dentro de um intervalo de tempo garantido.

A latência é usada para descrever o tempo necessário para responder a um evento. É semelhante ao determinismo em aplicações de controle em tempo real. Com uma resposta a eventos em tempo real, você tem garantia do pior caso de latência.

Sistemas Operacionais de Tempo Real

No núcleo de um sistema de software de tempo real está o sistema operacional de tempo real. Uma das maiores diferen-

ças entre os sistemas operacionais de tempo real e os sistemas operacionais de propósito geral é a habilidade de garantir o pior caso de latência. Em SOs de propósito geral, uma interrupção externa poderia ser inserida em uma fila e executada após o SO ter finalizado sua operação atual e outras interrupções existentes na fila.

Por outro lado, um SO de tempo real pode paralisar o processo atual para gerenciar uma interrupção imediatamente. Em essência, o SO de tempo real garante a resposta ao evento dentro de um determinado intervalo. Para fazer isto, o SO de tempo real define prioridades para processar as interrupções. Se uma interrupção de alta prioridade ocorrer, o SO paralisa imediatamente o processo e executa esta interrupção.

Aplicações Embarcadas e Headless

Muitas aplicações de tempo real são embarcadas (ou embutidos), mas nem todas as aplicações embarcadas possuem desempenho de tempo real. Enquanto aplicações de tempo real sempre aderem a características de temporização estrita, sistemas embarcados não são simples de categorizar.

Um sistema embarcado é geralmente pequeno em dimensões físicas e possui menos características do que um computador padrão. Enquanto não existe uma característica definida para sistemas embarcados, estes frequentemente possuem menos memória, pequenos processadores e poucos periféricos. Devido a estes não possuírem a natureza de propósito geral dos computadores *desktop*, seu programa atua como dispositivos dedicados.

Dispositivos *Headless* são sistemas embarcados sem periféricos. Os programas são frequentemente inseridos nestes dispositivos através de uma conexão de rede. Esta mesma conexão de rede é comumente utilizada como meio para configurar estes dispositivos. Enquanto este parece ser um fator limitante, com redes de banda larga e um software para aplicações inteligentes, você pode agora criar dispositivos *headless* com um simples menu *pull down*.

Detectores de passagem

Sensores ou detectores de passagem são empregados em máquinas industriais, eletrodomésticos, eletroeletrônicos, alarmes e muitas outras aplicações. Como funcionam e quais as configurações mais comuns, são assuntos importantes para o profissional da Eletrônica e de Automação. Neste artigo trataremos desses sensores, fornecendo uma coletânea de circuitos práticos.

Newton C. Braga

Sensores ou detectores de passagem são circuitos que detectam a passagem de pessoas ou objetos pelo corte de um feixe de luz visível ou de infravermelho, conforme mostra a **figura 1**.

Com o corte de luz por um instante, um sinal de comando é produzido servindo para acionar os mais diversos tipos de dispositivos como contadores, automatismos, sistemas de aviso ou alarme etc.

Podemos encontrar dispositivos em linhas de montagem, onde a passagem do objeto que está sendo produzido faz o acionamento de algum tipo de dispositivos como os citados no parágrafo anterior.

Outra aplicação é em alarmes, onde a passagem de uma pessoa entre o sensor e a fonte de luz provoca o disparo.

Finalmente temos o mesmo tipo de dispositivo em uma infinidade de eletroeletrônicos, tais como para detectar

a introdução de um CD em um *drive*, de uma fita em um videocassete (ou quando chega o seu final), e em muitas outras aplicações.

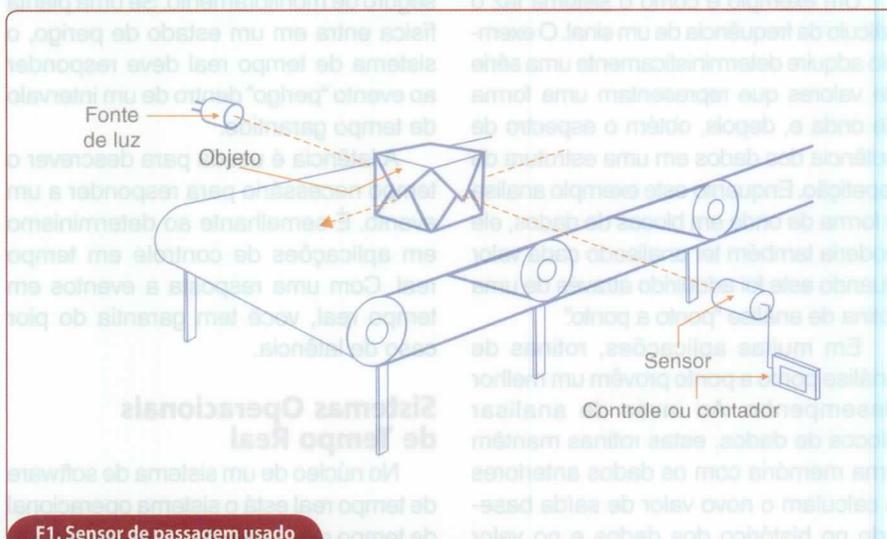
O tipo de configuração e de sensores que se usa para fazer um detector de passagem depende de diversos fatores, tais como:

Distância

O tipo de fonte de luz e de sensor depende em primeiro lugar da sua separação.

Para pequenas distâncias, como por exemplo num acoplador óptico ou numa linha de montagem, ou dentro de uma máquina em que ela não supera alguns centímetros, podemos usar LEDs como fontes de luz e sensores como fotodiodos, fototransistores ou LDRs, conforme ilustra a **figura 2**.

Para distâncias maiores a fonte de luz precisa ser intensa, o sensor pode ser qualquer um dos anteriores, mas eventualmente devem ser usados recur-



F1. Sensor de passagem usado em uma linha de produção.

sois ópticos, como lentes, para se obter maior sensibilidade e maior diretividade também.

Velocidade

A velocidade com que o objeto a ser detectado passa diante do sensor é outro fator a ser considerado.

Se o sensor for lento e o objeto passar rapidamente, a detecção não ocorrerá.

Os LDRs são sensores relativamente lentos, servindo para aplicações em que não temos uma frequência de passagem maior do que alguns quilohertz.

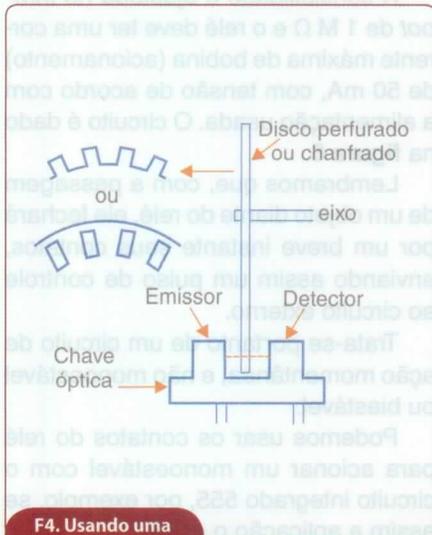
Já os fotodiodos e fototransistores podem detectar passagens em velocidades superiores a 10 MHz, dependendo do tipo.

Tipo de radiação usada

Para se evitar a interferência da luz ambiente, ou ainda para se obter maior alcance e maior diretividade, pode-se usar uma fonte de luz modulada.

Desta forma, o sensor deve ser capaz de responder à frequência da modulação, caso em que as aplicações exigem o uso de fotodiodos e fototransistores, uma vez que os LDRs são lentos para esta aplicação.

Na prática, a montagem do sistema consiste em uma fonte de luz que aponta para um sensor, deixando espaço livre para que o objeto a ser detectado passe entre ambos, veja na figura 3.



F4. Usando uma chave óptica.

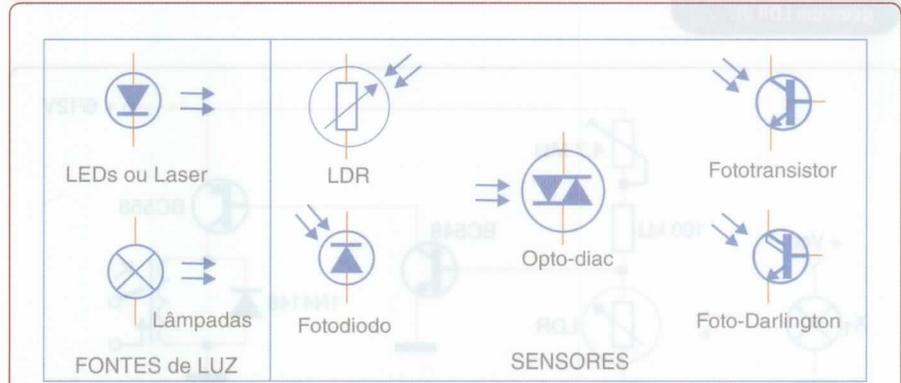
Esta montagem pode incluir recursos ópticos e ser provida de dois dispositivos isolados como mostra a figura. No entanto, para montagens industriais em que uma engrenagem ou um disco perfurado, ou ainda chanfrado, deve ser o agente de controle do circuito, pode-se usar uma chave óptica como a exibida na figura 4.

Chaves ópticas que já incorporam internamente tanto o emissor de luz infravermelha como circuitos sensores

(que vão desde fototransistores comuns até foto-Darlington e fotodisparadores lógicos) como as mostradas na figura 5, podem ser encontradas no comércio especializado. O tipo de sensor a ser escolhido, conforme já explicamos, depende de diversos fatores.

Circuitos práticos

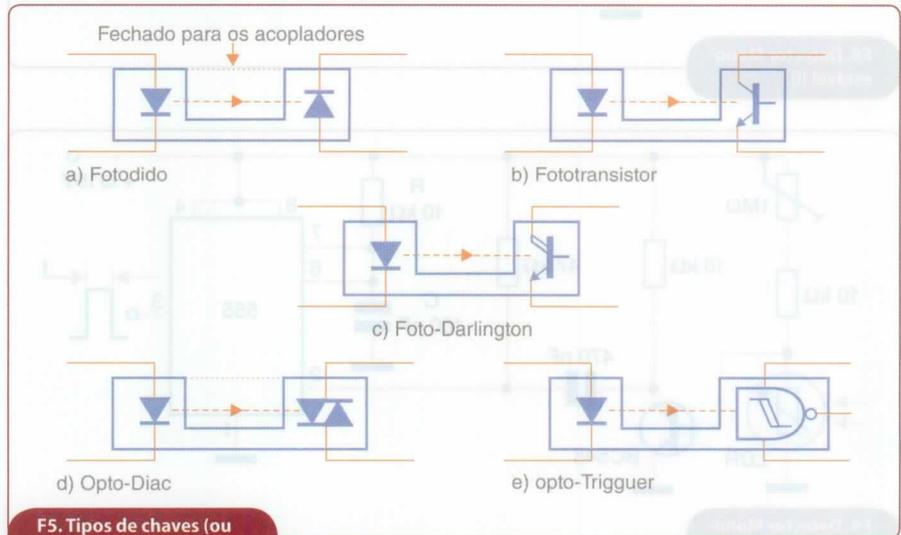
Damos a seguir uma coletânea de circuitos práticos que usam desde LDRs



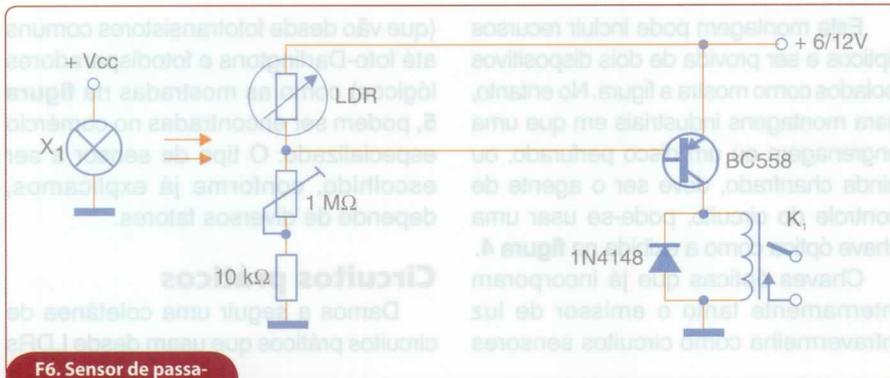
F2. Sensores e fontes de luz.



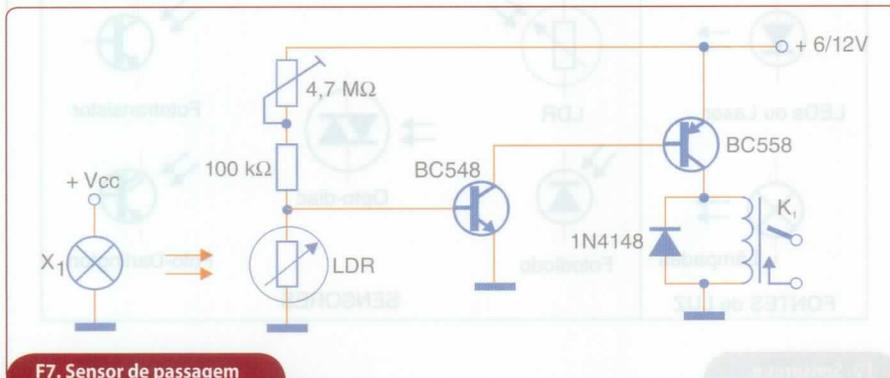
F3. Arranjo típico para um detector de passagem.



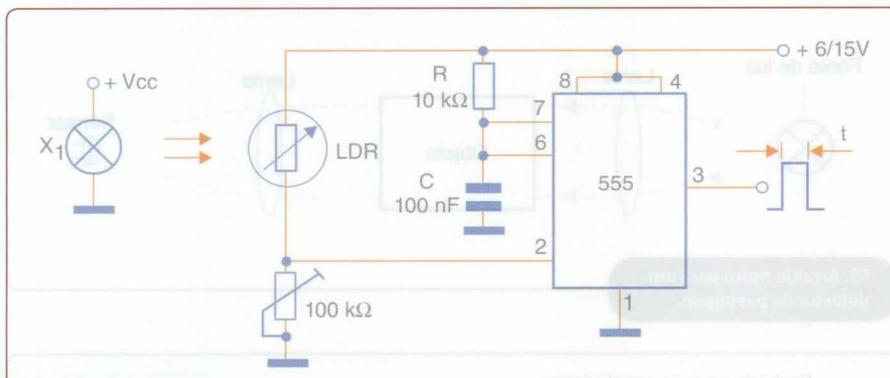
F5. Tipos de chaves (ou acopladores) ópticos.



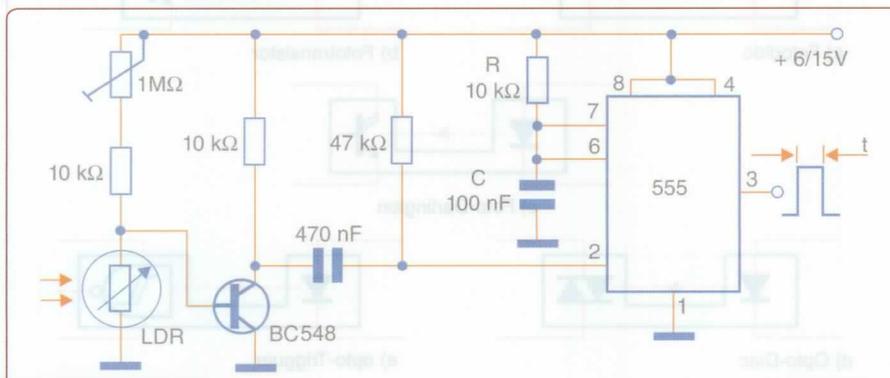
F6. Sensor de passagem com LDR (I).



F7. Sensor de passagem com LDR (II).



F8. Detector Monoestável (I).



F9. Detector Monoestável (II).

de baixa velocidade e acionam relés, até circuitos de alta velocidade que fazem uso de fontes de luz modulada.

Também daremos alguns circuitos acessórios como detectores de tom e moduladores que podem ser usados em projetos práticos.

Lembramos que os valores dos componentes usados devem ser eventualmente alterados em função da aplicação específica para se otimizar o desempenho, e que os transistores marcados com (*) são equivalentes indicados a partir da fonte original que sugere o circuito, o qual faz uso de componentes que não mais podem ser encontrados facilmente no nosso mercado.

Casualmente, em função do uso de equivalentes, a alteração de valores de resistores de polarização pode ser necessária para se obter o melhor desempenho do circuito.

Sensor de Passagem com LDR (I)

Este primeiro circuito é bastante simples e se o LDR for montado em um tubo opaco com lente, sua sensibilidade permite uma separação de até mais de 1 metro entre o sensor e a fonte.

A fonte de luz é uma lâmpada de 5 a 12 V com pelo menos 100 mA de corrente, dependendo da distância de separação.

Uma lâmpada incandescente de 5 a 40 W também pode ser usada para conexão na rede de energia.

A sensibilidade é ajustada no *trim-pot* de 1 MΩ e o relé deve ter uma corrente máxima de bobina (acionamento) de 50 mA, com tensão de acordo com a alimentação usada. O circuito é dado na **figura 6**.

Lembramos que, com a passagem de um objeto diante do relé, ele fechará por um breve instante seus contatos, enviando assim um pulso de controle ao circuito externo.

Trata-se portanto de um circuito de ação momentânea, e não monoestável ou biestável.

Podemos usar os contatos do relé para acionar um monoestável com o circuito integrado 555, por exemplo, se assim a aplicação o exigir.

Sensor de Passagem com LDR (II)

Este circuito é mais sensível do que o anterior, no sentido de que podemos usar fontes de luz mais fracas e mais distantes.

O sensor deve ser montado em um tubo opaco com lente que focalize a fonte distante. Distâncias de até mais de 5 metros podem ser cobertas com esta configuração.

O ajuste é feito no trimpot de 4,7 MΩ, e o relé deve ter uma bobina de acordo com a tensão de alimentação e corrente máxima de 50 mA.

O circuito também é de ação lenta, produzindo um breve fechamento dos contatos do relé quando acionado.

Recomenda-se este circuito para detecção de objetos extensos e lentos.

A fonte de luz pode ter as mesmas características do circuito anterior. A configuração é mostrada na **figura 7**.

Detector Monoestável (I)

A ação monoestável pode ser um fator importante em muitos projetos.

Entende-se por ação monoestável a produção de um pulso de saída de duração constante que independe da velocidade ou extensão do objeto detectado, ou seja, do tempo de interrupção do feixe de luz.

Esta ação pode ser obtida facilmente com a ajuda de um circuito integrado 555 na configuração monoestável.

O tempo de duração do pulso de saída depende de R e de C no circuito da **figura 8**. A constante de tempo é dada por:

$$T = 1,1 \cdot R \cdot C$$

O ajuste do ponto de disparo do circuito depende da intensidade da fonte de luz, e pode ser feito por um trimpot de 100 kΩ no pino 2. Para fontes fracas este potenciômetro pode ser aumentado para 1 MΩ.

Lembramos que o circuito integrado 555 pode excitar diretamente uma etapa com um relé, ou ainda etapas de circuitos lógicos de tecnologia TTL ou CMOS.

As fontes de luz dependem da sensibilidade desejada. O circuito completo deste detector é apresentado na **figura 8**.

Detector Monoestável (II)

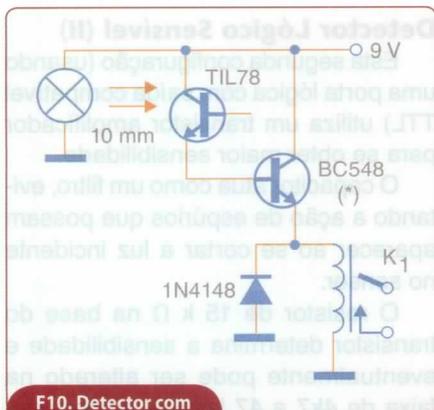
Trata-se de uma versão equivalente à anterior, mas com um tipo de ação diferente e com maior sensibilidade.

Este circuito contém recursos para que o pulso de disparo do 555 seja de duração constante, evitando problemas de repiques com objetos que tenham fendas ou ainda formatos irregulares.

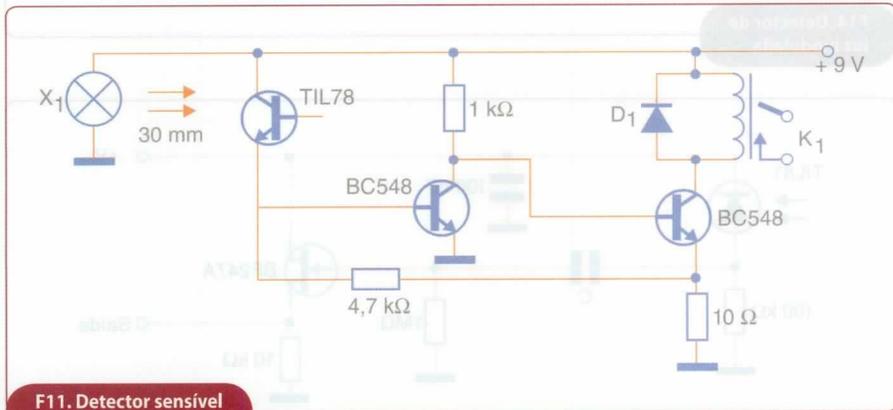
Ao mesmo tempo, a adição do transistor aumenta a sensibilidade do circuito que pode cobrir distâncias maiores.

Da mesma forma que o anterior, temos um comportamento monoestável em que o tempo no qual a saída permanece no nível alto depende de R e de C (conforme fórmula já dada).

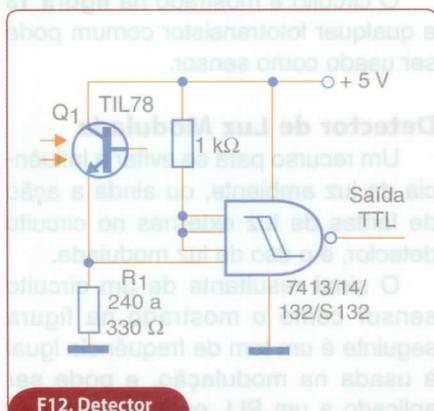
O circuito, que pode ser alimentado com tensões de 5 a 15 V, excita etapas com relés ou outras cargas e é compatível com tecnologias TTL e CMOS.



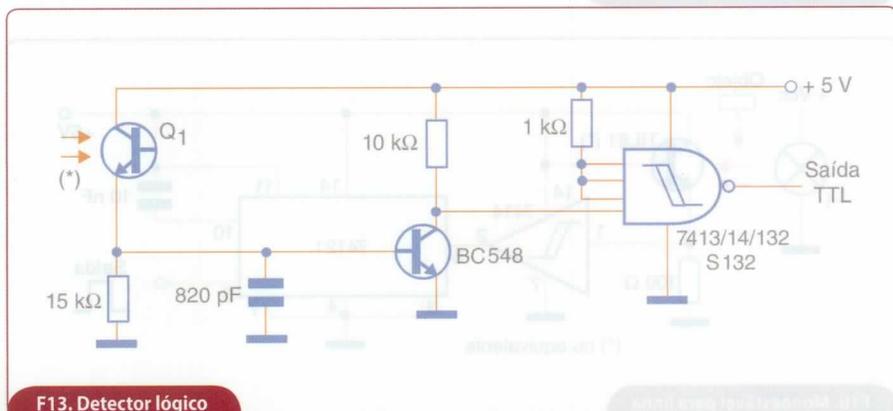
F10. Detector com fototransistor.



F11. Detector sensível com fototransistor.



F12. Detector lógico (I).



F13. Detector lógico sensível (II).

O circuito completo para este detector é ilustrado na **figura 9**.

Detector com Fototransistor

Um primeiro circuito que faz uso de um fototransistor como detector e que é indicado para a cobertura de pequenas distâncias (até 10 mm), podendo também ser usada uma chave óptica, é exibido na **figura 10**.

Este circuito aciona um relé, quando o feixe de luz que incide no fototransistor é cortado.

Para distâncias bem pequenas, a fonte de luz pode ser um diodo emissor de luz visível ou infravermelha.

O relé tem bobina de no máximo 50 mA e tensão de acordo com a usada para a alimentação da etapa.

Lembramos que este circuito produz um pulso apenas durante o tempo em que o objeto corta o feixe de luz e que, portanto, não tem ação monoestável.

Detector Sensível Com Fototransistor

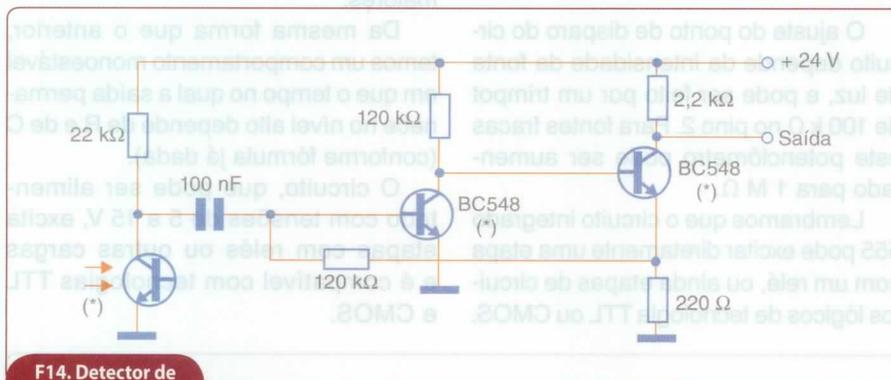
Este circuito faz uso de dois transistores numa configuração de acoplamento direto, e aciona diretamente um relé cujas características dependem da tensão de alimentação usada. A corrente de acionamento do relé deve

ser no máximo de 50 mA, devendo ser considerada a queda de tensão no resistor de realimentação de 10 Ω.

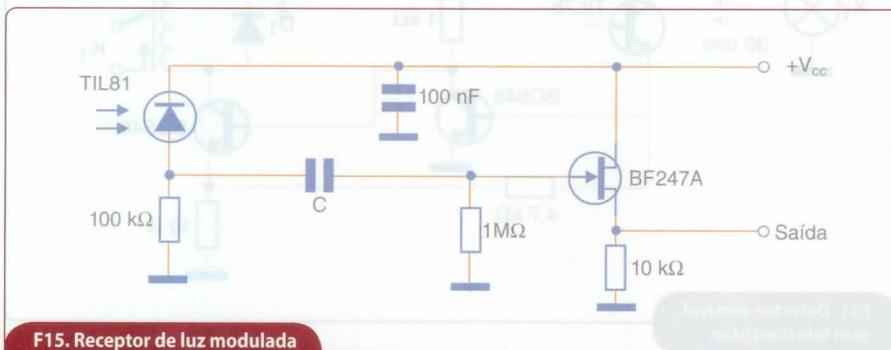
A fonte de luz é uma pequena lâmpada de baixa tensão, que tem sua luz focalizada no fototransistor a uma distância de até 3 cm.

Fontes mais potentes e recursos ópticos podem aumentar esta distância.

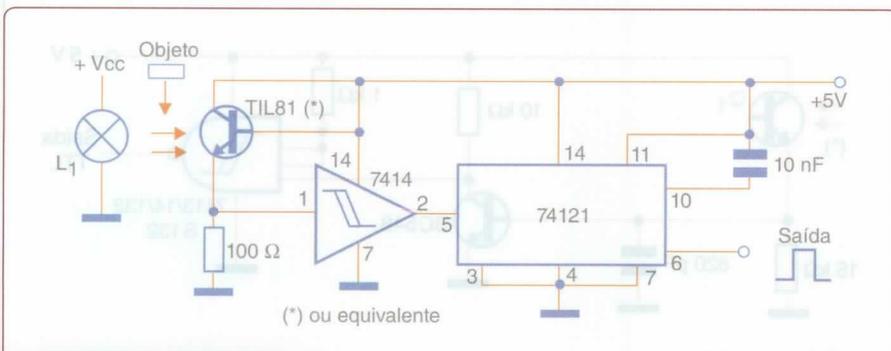
O resistor de 1 k ohm, eventualmente, pode ser alterado conforme as características de acionamento desejadas para o circuito. Na **figura 11** temos o diagrama completo deste detector de passagem.



F14. Detector de luz modulada.



F15. Receptor de luz modulada de alta velocidade.



F16. Monoestável para linha de montagem.

Detector Lógico (I)

O circuito apresentado na **figura 12** aciona diretamente uma porta lógica TTL, exigindo pois alimentação de 5 V.

O resistor R_1 determina a sensibilidade do circuito. O nível lógico de saída depende do fato da luz no sensor ser cortada ou incidir no mesmo, lembrando que a porta TTL atua como um inversor.

Outras funções inversoras-disparadoras (Schmitt) podem ser usadas na mesma configuração.

Detector Lógico Sensível (II)

Esta segunda configuração (usando uma porta lógica com saída compatível TTL) utiliza um transistor amplificador para se obter maior sensibilidade.

O capacitor atua como um filtro, evitando a ação de espúrios que possam aparecer ao se cortar a luz incidente no sensor.

O resistor de 15 k Ω na base do transistor determina a sensibilidade e eventualmente pode ser alterado na faixa de 4k7 a 47 kohms, conforme a aplicação.

O circuito é mostrado na **figura 13** e qualquer fototransistor comum pode ser usado como sensor.

Detector de Luz Modulada

Um recurso para se evitar a influência da luz ambiente, ou ainda a ação de fontes de luz externas no circuito detector, é o uso de luz modulada.

O sinal resultante de um circuito sensor como o mostrado na figura seguinte é um tom de frequência igual à usada na modulação, e pode ser aplicado a um PLL para detecção ou

ainda a um circuito reconhecedor de outro tipo, incluindo os que fazem uso de DSPs.

No circuito indicado, o resistor de 22 k Ω em série com o sensor pode ser alterado em função da sensibilidade desejada.

O fototransistor usado deve ser de tipo escolhido de acordo com a aplicação, mas em geral os tipos comuns como o TIL78 admitem operação com sinais modulados em até algumas centenas de quilohertz.

O circuito completo para esta aplicação está na **figura 14**.

Receptor de Luz Modulada de Alta Velocidade

Sinais modulados em até alguns megahertz podem ser detectados com o circuito mostrado na **figura 15**, que faz uso de um fotodiodo como sensor e um transistor de efeito de campo de baixa potência como elemento amplificador.

A sensibilidade depende do resistor de 100 k Ω em série com o fotodiodo, o qual pode ser alterado conforme a aplicação.

O capacitor C tem seu valor escolhido na faixa de 1 nF a 100 nF, conforme a frequência de modulação do sinal recebido.

A alimentação pode ficar entre 5 V e 15 V, tipicamente. O sensor pode ser qualquer fotodiodo ou mesmo uma junção PN de um fototransistor.

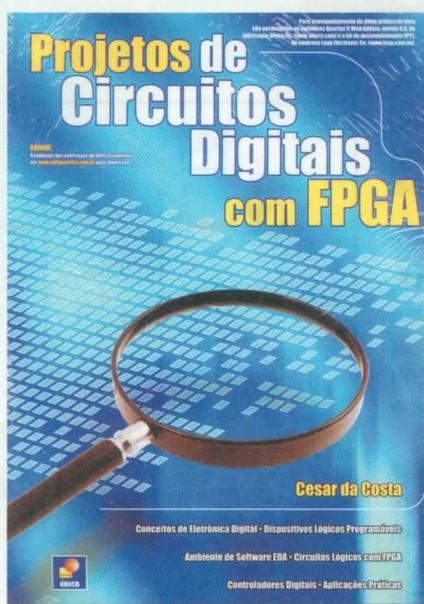
Monoestável Para Linha de Montagem

O circuito da **figura 16** é sugerido pela Texas Instruments em seu manual de optoeletrônica e consiste num sensor de passagem para objetos numa esteira de linha de montagem.

Este circuito usa um fototransistor para detectar a passagem do objeto, o qual gera uma transição lógica numa porta disparadora do tipo 7414.

Esta transição é usada para disparar um monoestável TTL, que gera um pulso de contagem ou controle de duração constante no pino 6 de saída.

O tempo de duração deste pulso depende do resistor de 15 kohms e do capacitor de 10 nF, componentes estes que podem ser alterados em função da aplicação desejada.

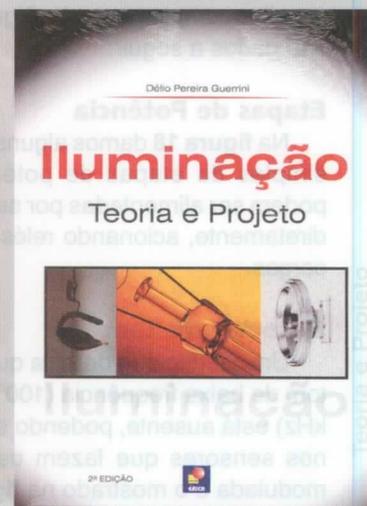
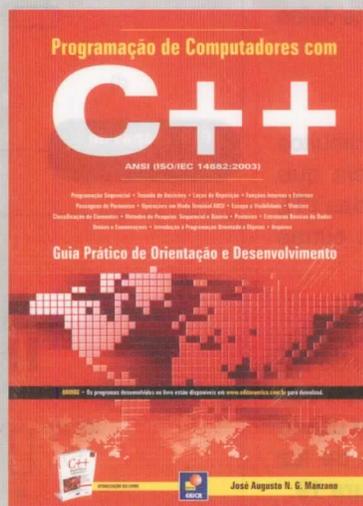
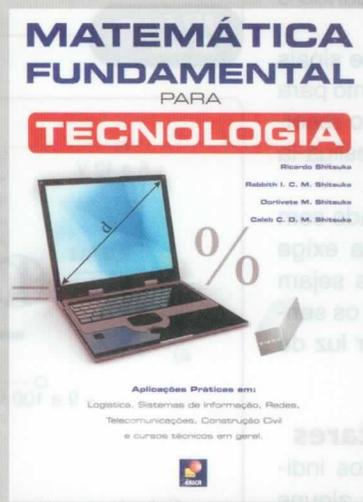


Projetos de Circuitos Digitais com FPGA

Mostra o emprego de novas ferramentas computacionais no desenvolvimento de projetos por meio de exemplos e casos práticos do dia-a-dia do projetista de circuitos digitais.

Destina-se a estudantes, professores, mestrandos e doutorandos, técnicos, autodidatas e profissionais da área.

Outros Destaques



novasaber

www.novasaber.com.br

Recursos ópticos junto à fonte de luz e ao sensor podem ser usados para se aumentar a sensibilidade e alcance.

Sensor de Boia

Este circuito, visto na **figura 17**, também é sugerido pela Texas Instruments e mostra como se pode controlar o nível de um reservatório de água (ou outro líquido transparente) utilizando-se sensores de passagem.

O que temos são dois sensores que são posicionados em níveis diferentes do reservatório, sendo focalizados em fontes de luz que no caso são LEDs.

O corte do feixe de luz por uma boia faz com que os circuitos sensores atuem sobre um *flip-flop* TTL montado com portas de um 7402, mudando-o de estado.

Assim, temos a produção de sinais nos níveis A e B, que servem tanto para ligar uma bomba para encher o reservatório no nível B como para desligá-la quando a boia atingir o nível A.

O circuito deve ser alimentado com 5 V e a sensibilidade elevada exige que os focos de luz dos LEDs sejam rigorosamente apontados para os sensores, que não podem receber luz de outras fontes.

Circuitos complementares

Para operar com os circuitos indicados podem ser necessários alguns circuitos complementares. Alguns deles são dados a seguir:

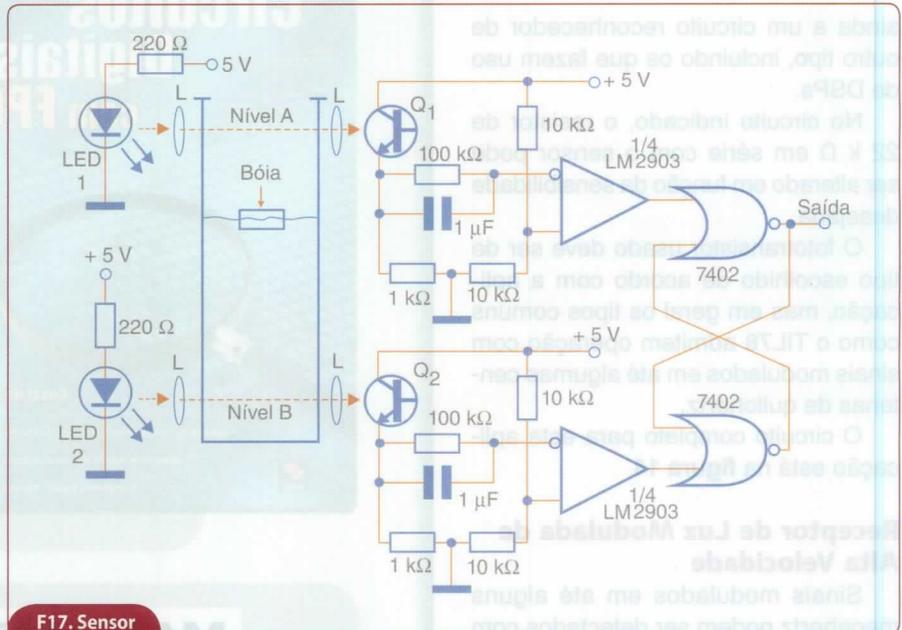
Etapas de Potência

Na **figura 18** damos alguns circuitos simples de etapas de potência que podem ser alimentadas por saídas TTL diretamente, acionando relés e outras cargas.

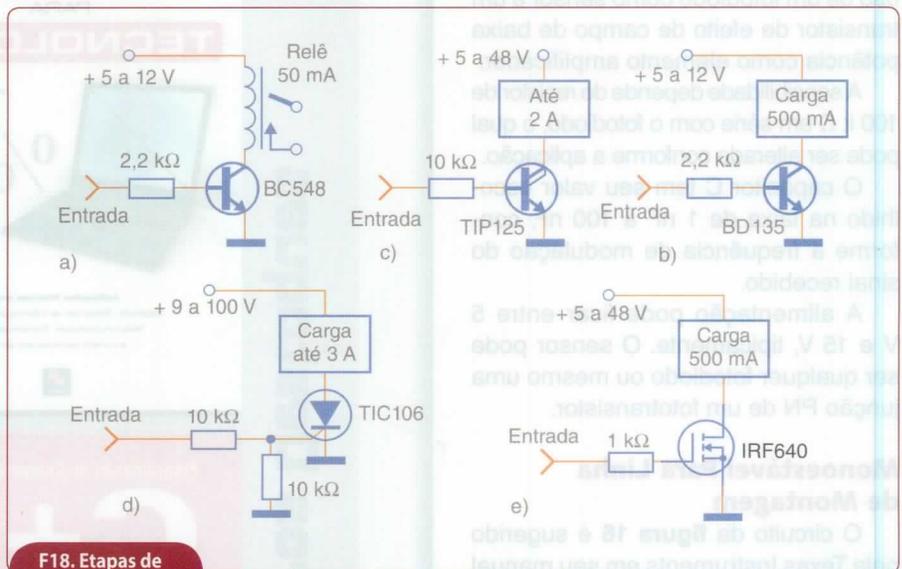
Detector de Tom

Um circuito que detecta quando um tom de baixa frequência (100 Hz a 500 kHz) está ausente, podendo ser usado nos sensores que fazem uso de luz modulada é o mostrado na **figura 19**.

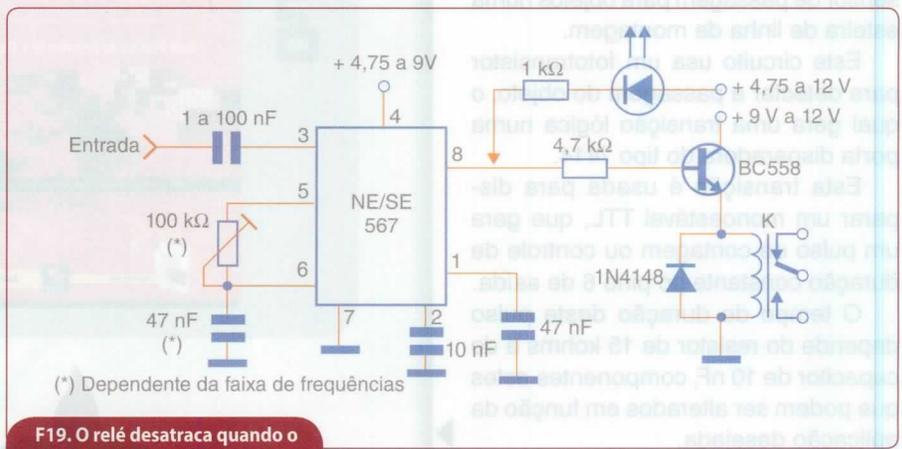
Este circuito faz uso de um PLL NE567 que é sintonizado no trimpot. Quando o tom de entrada deixa de ser reconhecido, o relé ligado na saída é acionado.



F17. Sensor de boia.



F18. Etapas de potência.



F19. O relé desatraca quando o sinal sintonizado desaparece.

criatividade e inovação

FEBRACE¹²



feira brasileira de ciências e engenharia

atenção criativos de todo Brasil

Já estão abertas as inscrições para a
12ª Feira Brasileira de Ciências e Engenharia



Para participar, basta ser estudante do 8º ou 9º ano do Ensino Fundamental, do Ensino Médio ou do Ensino Técnico, criar um projeto investigativo e enviá-lo até **29 de outubro de 2013**

INFORMAÇÕES E SUBMISSÕES

11 3091 5430

febrace@lsi.usp.br

www.febrace.org.br

MOSTRA DE FINALISTAS

18, 19 e 20 de
março de 2014

Você ainda tem a chance de representar o Brasil na Intel International Science and Engineering Fair nos EUA

Acompanhe a FEBRACE:



ORGANIZAÇÃO / REALIZAÇÃO



POLI USP

USP

Holtek agora também é ARM

NOVA LINHA DE MICROCONTROLADORES HT32Fxxx

Holtek inova com o lançamento de sua nova linha de microcontroladores HT32Fxxx que une todo o poder de processamento do core ARM Cortex-M3 com sua experiência de mais de 20 anos desenvolvendo microcontroladores de 8 bits, resultando em um microcontrolador poderoso com um set de periféricos abrangente altamente competitivos em preço.

CARACTERÍSTICAS:

Com a integração do núcleo Cortex-M3 em seus dispositivos os desenvolvedores contam com as seguintes vantagens:

- Frequência de operação de até 72MHz.
- 1,25 DMIPS/MHz
- Multiplicação e divisão em um único ciclo por hardware.
- Interrupções vetoradas e aninhadas através do controlador NVIC
- Timer de 24-bits

A arquitetura ARM presente em seus novos microcontroladores de 32 bits, proporciona o uso dos melhores ambientes de programação da atualidade como:

- Keil MDK-ARM
- IAR EWARM

Visite o site da Holtek e conheça toda linha:

www.holtek.com

HOLTEK 

ANYTEK
conectores

FORDATA
FOR DATA DISPLAY

módulos RF
WENSHING 

GlobalSat
Wired by Wireless

PLUS
CI reproduzidor de sons

FAGOR 

SIMCom
A company of SIM Tech

 **Cika**
Eletrônica

Seu Distribuidor de Componentes Eletrônicos

Tel.: 55 11 2696-6428 / Fax: 55 11 2696-6422
vendas@cikaeletronica.com
www.cikaeletronica.com