



Projetos de Automação

*Édilus de Carvalho Castro Penido
Pedro Luis Almeida de Oliveira Costa*

Presidência da República Federativa do Brasil
Ministério da Educação
Secretaria de Educação Profissional e Tecnológica

© Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais
Este caderno foi elaborado em parceria entre o Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – Campus Ouro Preto e a Universidade Federal de Santa Maria para a Rede e-Tec Brasil.

Equipe de Elaboração
Instituto Federal de Educação, Ciência e
Tecnologia de Minas Gerais – IFMG-Ouro Preto

Reitor
Caio Mário Bueno Silva/IFMG-Ouro Preto

Direção Geral
Arthur Versiani Machado/IFMG

Coordenação Institucional
Sebastião Nepomuceno/IFMG-Ouro Preto

Coordenação de Curso
Cristiano Lúcio Cardoso Rodrigues/IFMG-Ouro Preto

Professor-autor
Édilus de Carvalho Castro Penido/IFMG-Ouro Preto
Pedro Luis Almeida de Oliveira Costa/IFMG-Ouro Preto

Equipe de Acompanhamento e Validação
Colégio Técnico Industrial de Santa Maria – CTISM

Coordenação Institucional
Paulo Roberto Colusso/CTISM

Coordenação de Design
Erika Goellner/CTISM

Revisão Pedagógica
Elisiane Bortoluzzi Scrimini/CTISM
Jaqueline Müller/CTISM

Revisão Textual
Carlos Frederico Ruviano/CTISM

Revisão Técnica
Guilherme Dhein/CTISM

Ilustração
Marcel Santos Jacques/CTISM
Ricardo Antunes Machado/CTISM

Diagramação
Emanuelle Shaiane da Rosa/CTISM
Tagiane Mai/CTISM

Ficha catalográfica elaborada por Márcia Margarida Vilaça – CRB-6- 2235

C355a Penido, Édilus de Carvalho Castro
Projetos de Automação. Édilus de Carvalho Castro Penido.
Ouro Preto: 2016.
67 p.; il.
ISBN: 978-85-9450-027-4

1. Automação Industrial. 2. Circuitos Eletrônicos de Controle
3. Microcontroladores. I. Penido, Édilus de Carvalho Castro. II.
Costa, Luis Almeida de Oliveira. III. Instituto Federal de Minas
Gerais, Campus Ouro Preto. IV. Universidade Federal de Santa
Maria/Rede e-Tec Brasil. V. Título.

CDU: 621.38

Apresentação e-Tec Brasil

Prezado estudante,
Bem-vindo a Rede e-Tec Brasil!

Você faz parte de uma rede nacional de ensino, que por sua vez constitui uma das ações do Pronatec – Programa Nacional de Acesso ao Ensino Técnico e Emprego. O Pronatec, instituído pela Lei nº 12.513/2011, tem como objetivo principal expandir, interiorizar e democratizar a oferta de cursos de Educação Profissional e Tecnológica (EPT) para a população brasileira propiciando caminho de o acesso mais rápido ao emprego.

É neste âmbito que as ações da Rede e-Tec Brasil promovem a parceria entre a Secretaria de Educação Profissional e Tecnológica (SETEC) e as instâncias promotoras de ensino técnico como os Institutos Federais, as Secretarias de Educação dos Estados, as Universidades, as Escolas e Colégios Tecnológicos e o Sistema S.

A educação a distância no nosso país, de dimensões continentais e grande diversidade regional e cultural, longe de distanciar, aproxima as pessoas ao garantir acesso à educação de qualidade, e promover o fortalecimento da formação de jovens moradores de regiões distantes, geograficamente ou economicamente, dos grandes centros.

A Rede e-Tec Brasil leva diversos cursos técnicos a todas as regiões do país, incentivando os estudantes a concluir o ensino médio e realizar uma formação e atualização contínuas. Os cursos são ofertados pelas instituições de educação profissional e o atendimento ao estudante é realizado tanto nas sedes das instituições quanto em suas unidades remotas, os polos.

Os parceiros da Rede e-Tec Brasil acreditam em uma educação profissional qualificada – integradora do ensino médio e educação técnica, – é capaz de promover o cidadão com capacidades para produzir, mas também com autonomia diante das diferentes dimensões da realidade: cultural, social, familiar, esportiva, política e ética.

Nós acreditamos em você!
Desejamos sucesso na sua formação profissional!

Ministério da Educação
Abril de 2016

Nosso contato
etecbrasil@mec.gov.br



Indicação de ícones

Os ícones são elementos gráficos utilizados para ampliar as formas de linguagem e facilitar a organização e a leitura hipertextual.



Atenção: indica pontos de maior relevância no texto.



Saiba mais: oferece novas informações que enriquecem o assunto ou “curiosidades” e notícias recentes relacionadas ao tema estudado.



Glossário: indica a definição de um termo, palavra ou expressão utilizada no texto.



Mídias integradas: sempre que se desejar que os estudantes desenvolvam atividades empregando diferentes mídias: vídeos, filmes, jornais, ambiente AVEA e outras.



Atividades de aprendizagem: apresenta atividades em diferentes níveis de aprendizagem para que o estudante possa realizá-las e conferir o seu domínio do tema estudado.



Sumário

Palavra do professor-autor	9
Apresentação da disciplina	11
Projeto instrucional	13
Aula 1 – Automação de sistemas	15
1.1 Considerações iniciais.....	15
1.2 Fases da automação de um sistema.....	16
1.3 Exemplo de uma automação passo a passo.....	16
Aula 2 – Diagrama elétrico	21
2.1 Função e importância.....	21
2.2 Passos para a elaboração.....	21
Aula 3 – Programação do controlador	27
3.1 Função e importância.....	27
3.2 Elaboração do algoritmo.....	28
3.3 Codificação em uma linguagem específica.....	29
Aula 4 – Lista de material	35
4.1 Função e importância.....	35
4.2 Observações para a elaboração.....	35
4.3 Itens indiretos necessários.....	36
4.4 Itens não relacionados.....	37
Aula 5 – Variações de projeto	41
5.1 Alterações no projeto original.....	41
5.2 Sinalização de funcionamento com LEDs.....	41
5.3 Utilização de módulos LCD.....	43
Aula 6 – Projetos com CLPs	55
6.1 Automação com CLPs.....	55
6.2 Descrição da solução proposta.....	55
6.3 Diagrama elétrico.....	56
6.4 Programação do elemento de comando.....	58
6.5 Lista de material.....	58

Aula 7 – Projetos finais	61
7.1 Sistemas automatizados.....	61
7.2 Automação de um sistema de esteiras.....	61
7.3 Elevador com três paradas.....	63
Referências	66
Currículo do professor-autor	67

Palavra do professor-autor

Vivemos, atualmente, num mundo onde há uma mudança contínua nos processos industriais, objetivando a melhoria da qualidade dos produtos e a redução dos custos envolvidos nesta industrialização. Comumente, cada atividade ou etapa é revista várias vezes na busca pela excelência, sendo que a procura por novas soluções para os problemas envolvidos é uma constante.

Na área da automação, que funciona como um suporte para todo o processo industrial, isso também não seria diferente. Com isso, os profissionais deste segmento precisam estar aptos a assimilar mudanças tecnológicas significativas que podem ocorrer em pequenos espaços de tempo.

Uma base sólida de conhecimentos teóricos e práticos, além de imaginação e da capacidade de se adaptar ao novo, são grandes facilitadores nesse processo. A disciplina de Projetos de Automação pretende então colaborar com isso ao abordar temas complementares aos demais tópicos apresentados em todo o curso, fornecendo aos alunos informações que permitam a combinação dos conhecimentos adquiridos para o desenvolvimento de sistemas automatizados mais versáteis.

A primeira parte da disciplina pretende fornecer aos alunos uma visão geral dos bancos de dados e da sua interação com a internet. Eles são largamente utilizados hoje em dia para armazenar todas as informações ligadas aos processos industriais, permitindo que as tomadas de decisão (automáticas ou não) sejam bem mais embasadas.

A segunda parte da disciplina propõem, para os futuros profissionais da área, uma série de atividades para que eles aprendam, de forma eminentemente prática, uma metodologia para a automação de sistemas. Inclusive, há no final dela um projeto final para aplicação do que foi estudado.

Esperamos que todos vocês tenham um grande sucesso, não só na disciplina e no curso de modo geral, mas principalmente na vida profissional, seguindo essa grande carreira que escolheram, atuando com empenho e dedicação em tudo que fizerem.

Édilus de Carvalho Castro Penido
Pedro Luis Almeida de Oliveira Costa



Apresentação da disciplina

Caros alunas e alunos,

É com muita alegria que iniciamos hoje a disciplina Projetos de Automação. Ela é uma das etapas finais para a conclusão do curso e representa para você a oportunidade de aplicação de vários conhecimentos adquiridos ao longo do curso.

Teremos a chance de ver aqui o que são os bancos de dados e como eles podem se relacionar com a internet, além de aprender uma metodologia para automação de sistemas com microcontroladores e micro CLPs.

Estas áreas têm um enorme potencial e são largamente aplicadas no mundo moderno. Sendo assim, não perca a oportunidade de consolidar estes novos conhecimentos.

Desejamos que vocês tenham um ótimo curso e colocamo-nos à disposição para ajudar no que for preciso para uma boa assimilação dos temas que serão tratados aqui.

Com muita alegria, sejam bem-vindos!



Projeto instrucional

Disciplina: Projetos de Automação (carga horária: 75h).

Ementa: Desenvolvimento de projetos para a automação de sistemas utilizando microcontroladores e CLPs, aplicando os conhecimentos adquiridos em várias disciplinas do curso.

AULA	OBJETIVOS DE APRENDIZAGEM	MATERIAIS	CARGA HORÁRIA (horas)
1. Automação de sistemas	Compreender o que representa a automação de um sistema. Conhecer os métodos de automação baseados em microcontroladores e CLPs para operações simples. Conhecer as fases do processo de automação de sistemas de pequeno porte. Propor soluções de automação para pequenos projetos.	Ambiente virtual: plataforma Moodle. Apostila didática. Recursos de apoio: <i>links</i> , exercícios.	08
2. Diagrama elétrico	Compreender a importância de um diagrama elétrico para um projeto de automação. Conhecer os passos para a elaboração do diagrama elétrico. Analisar e compreender o diagrama elétrico de outros projetos. Desenvolver os primeiros diagramas elétricos para a automação de pequenos sistemas.	Ambiente virtual: plataforma Moodle. Apostila didática. Recursos de apoio: <i>links</i> , exercícios.	08
3. Programação do controlador	Conhecer as vantagens de um sistema automatizado programável em relação aos não programáveis. Compreender a importância da programação para o correto funcionamento de um sistema automatizado. Desenvolver os algoritmos que servirão de base para a elaboração de programas. Analisar e compreender os programas de outros projetos. Desenvolver os primeiros programas para a automação de pequenos sistemas.	Ambiente virtual: plataforma Moodle. Apostila didática. Recursos de apoio: <i>links</i> , exercícios.	08

AULA	OBJETIVOS DE APRENDIZAGEM	MATERIAIS	CARGA HORÁRIA (horas)
4. Lista de material	<p>Compreender a importância da lista de material para um projeto de automação.</p> <p>Conhecer os pontos que devem ser observados para a elaboração da lista de material.</p> <p>Aprender quais são as partes mínimas e quais são as partes recomendáveis para fazer parte de uma lista de material.</p> <p>Desenvolver as primeiras listas de materiais para a automação de pequenos sistemas.</p>	<p>Ambiente virtual: plataforma Moodle.</p> <p>Apostila didática.</p> <p>Recursos de apoio: <i>links</i>, exercícios.</p>	08
5. Variações de projeto	<p>Compreender a necessidade de se modificar um projeto para facilitar a sua utilização na prática.</p> <p>Desenvolver alterações que propiciem ao usuário de um sistema um conjunto maior de informações visuais sobre o seu estado atual.</p> <p>Utilizar LEDs para indicar de forma simples o estado de um sistema.</p> <p>Aprender como usar módulos LCD para exibir mensagens em projetos que empreguem microcontroladores.</p>	<p>Ambiente virtual: plataforma Moodle.</p> <p>Apostila didática.</p> <p>Recursos de apoio: <i>links</i>, exercícios.</p>	08
6. Projetos com CLPs	<p>Compreender as diferenças na metodologia de automação proposta entre a utilização de microcontroladores e CLPs como elemento principal de comando.</p> <p>Desenvolver diagramas elétricos para sistemas de automação onde o elemento principal de comando seja um CLP.</p> <p>Desenvolver programas em Ladder para sistemas automatizados de pequeno porte.</p> <p>Elaborar listas de material para sistemas automatizados de pequeno porte baseados em CLPs.</p>	<p>Ambiente virtual: plataforma Moodle.</p> <p>Apostila didática.</p> <p>Recursos de apoio: <i>links</i>, exercícios.</p>	08
7. Projetos finais	<p>Revisar muitos conceitos vistos ao longo de várias disciplinas que compõem o Curso de Técnico em Automação Industrial.</p> <p>Aplicar os conhecimentos adquiridos em uma possível situação onde haja a necessidade de automação de um sistema.</p> <p>Compreender como o processo de automação de um sistema se desenvolve na prática.</p>	<p>Ambiente virtual: plataforma Moodle.</p> <p>Apostila didática.</p> <p>Recursos de apoio: <i>links</i>, exercícios.</p>	27

Aula 1 – Automação de sistemas

Objetivos

Compreender o que representa a automação de um sistema.

Conhecer os métodos de automação baseados em microcontroladores e CLPs para operações simples.

Conhecer as fases do processo de automação de sistemas de pequeno porte.

Propor soluções de automação para pequenos projetos.

1.1 Considerações iniciais

Segundo a definição encontrada no Dicionário Aurélio (FERREIRA, 1999), um sistema automático é aquele no qual “os mecanismos controlam seu próprio funcionamento, quase sem a interferência do homem”.

Em relação especificamente à automação industrial, encontramos a seguinte afirmação:

A automação industrial de uma máquina/processo consiste essencialmente em escolher, de entre as diversas tecnologias que se encontram ao nosso dispor, as que melhor se adaptam ao processo a desenvolver e a melhor maneira de as interligar para garantir sempre a melhor relação custo/benefício (WIKIPÉDIA, 2014).

Portanto, cabe ao profissional da área escolher uma tecnologia adequada para a automação de sistema quando se confrontar com uma determinada situação após analisar suas características. Podemos citar alguns pontos muito importantes nesta análise que devem ser observados:

- Desempenho esperado.
- Robustez de todo o conjunto.



Para saber mais sobre automação, acesse:
<http://pt.wikipedia.org/wiki/Automação>

http://pt.wikipedia.org/wiki/Automação_industrial

- Investimento inicial.
- Tipos e formas das manutenções requeridas.
- Treinamento do pessoal técnico e operacional.
- Possíveis alterações do modo de funcionamento.
- Monitoração remota.

1.2 Fases da automação de um sistema

Existem várias formas de automatizar um sistema ou processo. Optamos em apresentar nesta aula uma metodologia que acreditamos ser mais didática e adequada para aqueles que estão se iniciando no mundo da automação. Ela também apresenta a vantagem de poder ser adequada para sistemas de vários portes, além de, também, poder ser aplicada tanto em sistemas baseados em microcontroladores quanto em CLPs (Controladores Lógicos Programáveis).

Esta metodologia se inicia com a análise do sistema que se deseja automatizar e é baseada nas seguintes fases:

- Descrição da solução proposta.
- Desenvolvimento do **diagrama elétrico**.
- Programação do controlador (microcontrolador ou CLP) do sistema.
- Elaboração da lista de material.

A-Z

diagrama elétrico
Também é conhecido
como esquema elétrico.

1.3 Exemplo de uma automação passo a passo

Para que você possa compreender melhor esta metodologia e possa aplicá-la em outros casos, vamos partir de uma situação de exemplo. Então, consideremos que desejamos automatizar o processo de enchimento de uma caixa d'água a partir da água bombeada de um rio. Suponhamos também que o proprietário deste sistema também deseja que um alarme toque no caso de falha do sistema de controle de nível e a caixa transborde.

O primeiro passo a ser feito é uma descrição do que pretendemos fazer para solucionar o problema. Ela deve ser o mais detalhada possível, já que norteará todos os nossos próximos passos e deverá ser apresentada para o proprietário do sistema e/ou quem irá utilizar o sistema para verificarmos se ela realmente atende às expectativas deles e se não esquecemos de observar algum pormenor importante.

Uma das possíveis formas de automatizar a situação proposta de controle do nível de uma caixa d'água é a seguinte:

O sistema automatizado terá um sensor para indicar que a caixa está no nível máximo permitido, representado na Figura 1.1 através do símbolo com a letra **H** (**High** → **Alto**).

Para a indicação do nível mínimo, também haverá um sensor representado na Figura 1.1 através do símbolo com a letra **L** (**Low** → **Baixo**).

Os sinais fornecidos pelos sensores serão usados para determinar o momento do acionamento da bomba (**P** – **Pump** → **Bomba**).

Se o nível da água estiver abaixo da posição do sensor **L**, a bomba **P** ficará ligada enquanto este nível não atingir o sensor **H**, que será então desligada quando isto ocorrer.

A bomba **P** só será ligada novamente quando o nível da água voltar a atingir o sensor **L**.

Haverá um sensor de falha (**F** – **Fail** → **Falha**), que será montado na borda da caixa d'água e servirá para indicar um eventual transbordamento.

Se o nível da água ultrapassar o sensor **H** e atingir o sensor de falha **F**, uma sirene de alarme (**A** – **Alarm** → **Alarme**) será soada e a bomba **P** ficará desativada até que um operador pressione o botão de desarme (**D** – **Disarm** → **Desarme**).

Podemos representar a solução proposta em desenho esquemático para facilitar a compreensão do que foi descrito para todos os envolvidos. Uma possibilidade de desenho para esta situação é mostrada na Figura 1.1.

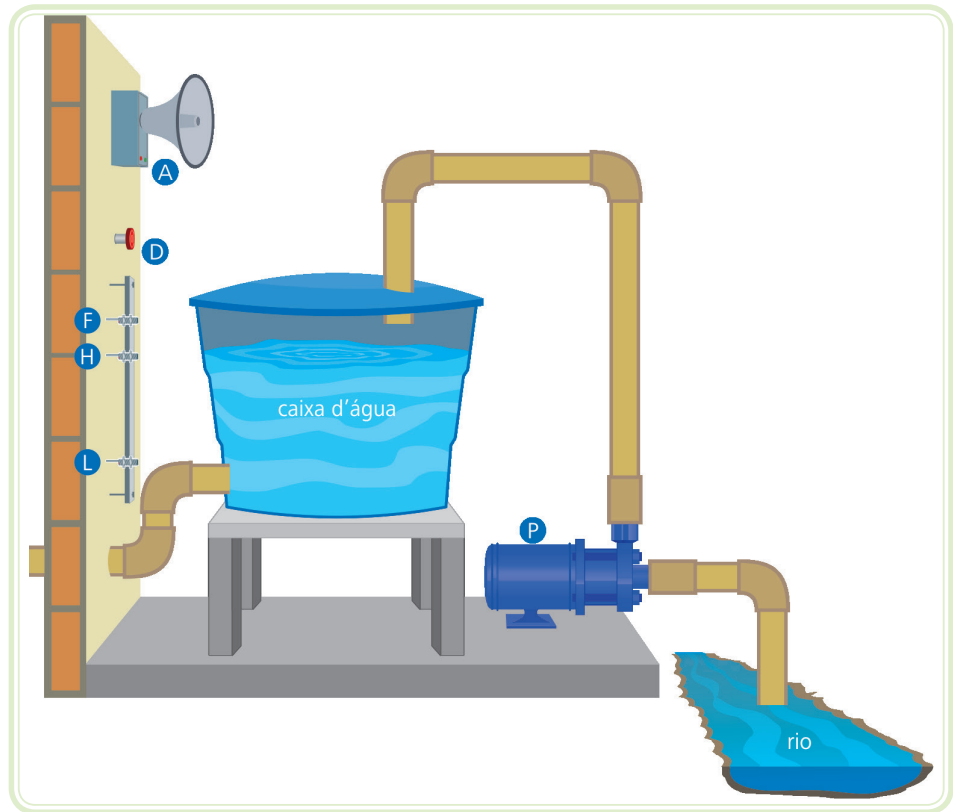


Figura 1.1: Representação da solução de automação proposta

Fonte: CTISM, adaptado dos autores

Os outros passos da metodologia sugerida para automação de sistemas serão abordados nas próximas aulas com maior nível de detalhamento. Vamos estudá-los juntos em breve!

Resumo

Nessa etapa, estudamos o que representa a automação de sistemas e apresentamos uma metodologia (modo de agir) que poderá ser usada para automatizar sistemas de diversos portes.

Para compreender melhor esta forma de realizar uma tarefa de automatização, começamos a aplicá-la na situação onde desejamos controlar o nível de uma caixa d'água.

Nas próximas aulas, abordaremos os outros passos da metodologia de automação proposta. Você terá então uma ótima oportunidade de exercitar e aplicar vários conceitos aprendidos em outras disciplinas do curso. Portanto, faça com dedicação os exercícios de aprendizagem desta aula e continue firme nos seus estudos!

Atividades de aprendizagem



1. Explique, com suas palavras, o que é automação e qual é sua importância para o mundo moderno.
2. Descreva detalhadamente uma solução para a automação de cada uma das situações a seguir:
 - a) A caixa d'água da situação mostrada na Figura 1.1 deverá ter agora seu nível máximo controlado em três alturas diferentes, sendo que esta seleção (limite do nível máximo) poderá ser feita pelo operador do sistema de forma manual.
 - b) A quantidade de aparelhos de ar-condicionado ligados para resfriar uma sala que só possui uma porta de acesso dependerá da quantidade de pessoas dentro dela da seguinte maneira:
 - Nenhuma pessoa presente → nenhum aparelho ligado.
 - De uma a cinco pessoas presentes → um aparelho ligado.
 - De seis a dez pessoas presentes → dois aparelhos ligados.
 - De onze a quinze pessoas presentes → três aparelhos ligados.
 - Dezesesseis ou mais pessoas presentes → quatro aparelhos ligados.
 - c) Uma outra sala, só que agora com duas portas de acesso, deverá ter seus quatro aparelhos controlados da mesma forma do que a sala do item anterior.
3. Faça um desenho representativo das soluções de automação propostas para cada uma das situações do exercício anterior.

Aula 2 – Diagrama elétrico

Objetivos

Compreender a importância de um diagrama elétrico para um projeto de automação.

Conhecer os passos para a elaboração do diagrama elétrico.

Analisar e compreender o diagrama elétrico de outros projetos.

Desenvolver os primeiros diagramas elétricos para a automação de pequenos sistemas.

2.1 Função e importância

O diagrama elétrico simboliza, de forma esquemática, a ligação entre todos componentes que formam o sistema de automação. A partir da sua análise, os profissionais que serão responsáveis pela montagem ou por uma eventual manutenção futura deste sistema podem compreender melhor como as partes se interligam e como é o funcionamento do todo.

2.2 Passos para a elaboração

A elaboração do diagrama elétrico se inicia com o levantamento da quantidade e do tipo das entradas e saídas (E/S) requeridas pelo sistema a ser automatizado. No nosso exemplo, a partir da solução proposta para o controle de nível da caixa d'água mostrada anteriormente, podemos relacionar os seguintes pontos de entrada e saída, indicados nos Quadros 2.1 e 2.2, respectivamente.

Quadro 2.1: Relação de entradas do projeto

	Descrição	Tipo
1	Sensor de nível alto da caixa d'água (H)	Digital
2	Sensor de nível baixo da caixa d'água (L)	Digital
3	Sensor de falha – transbordamento (F)	Digital
4	Botão de desarme do alarme (D)	Digital
Total		04

Fonte: Autores

Quadro 2.2: Relação de saídas do projeto

	Descrição	Tipo
1	Sirene de alarme (A)	Digital
2	Motor de bomba da caixa d'água (B)	Digital
Total		02

Fonte: Autores

A partir desta análise, observamos que serão necessários seis pontos digitais no total. Como o modo de funcionamento do sistema proposto é relativamente simples, optamos que o seu gerenciamento seja feito por um microcontrolador PIC 12F508 da Microchip, mostrado de forma esquemática na Figura 2.1, que possui justamente seis pinos de E/S (oito pinos no total). Também poderíamos usar um micro CLP como elemento de controle, como será mostrado em aulas posteriores.



Para saber mais sobre CLPs e microcontroladores, acesse:
<http://pt.wikipedia.org/wiki/Microcontrolador>

[http://pt.wikipedia.org/wiki/Controlador_lógico_programável](http://pt.wikipedia.org/wiki/Controlador_l%C3%B3gico_program%C3%A1vel)

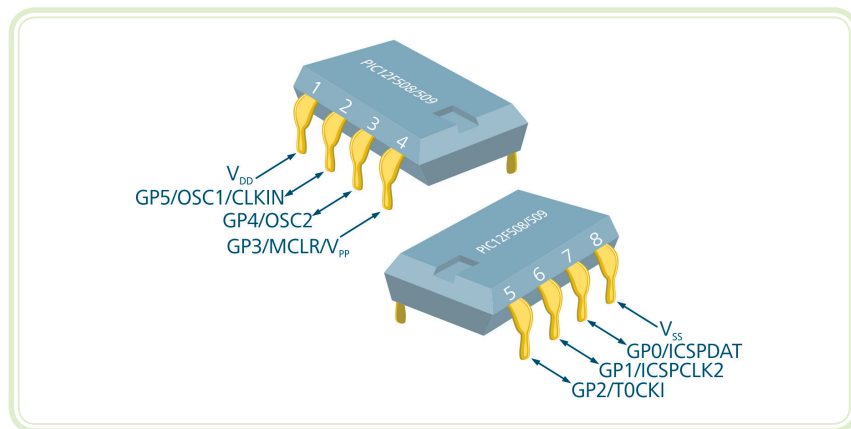


Figura 2.1: Microcontrolador PIC12F508

Fonte: CTISM, adaptado de Microchip, 2014

Se houver a previsão de expansões futuras no número de entradas e saídas de um sistema ou o aumento do nível de complexidade do seu funcionamento, podemos especificar inicialmente um outro microcontrolador mais completo. No entanto, devemos nos lembrar que ele será provavelmente mais caro.

Após a determinação da quantidade e do tipo das E/S, podemos passar a seleção dos componentes que gerarão (entradas) ou receberão (saídas) os sinais para o microcontrolador escolhido.

As entradas do sistema são sensores de nível (alto, baixo e falha) do tipo boia e o modelo escolhido foi CB2002, fabricado pela Margirius. Já o botão de desarme do alarme selecionado é do tipo de microinterruptor, modelo 18531 MF T3F ST S E, também fabricado para MarGirius.

As Figuras 2.2 e 2.3 mostram, respectivamente, um sensor do tipo boia e como ele deve ser utilizado para indicar o nível de um reservatório.

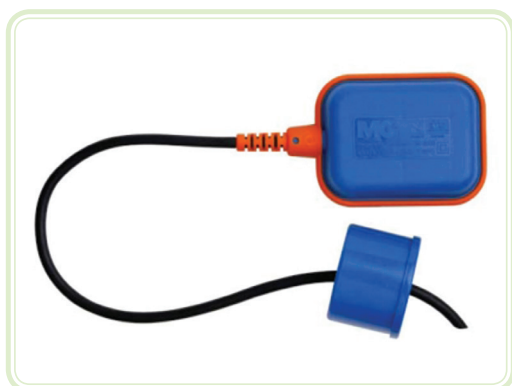


Figura 2.2: Sensor de nível do tipo boia

Fonte: MarGirius, 2014a

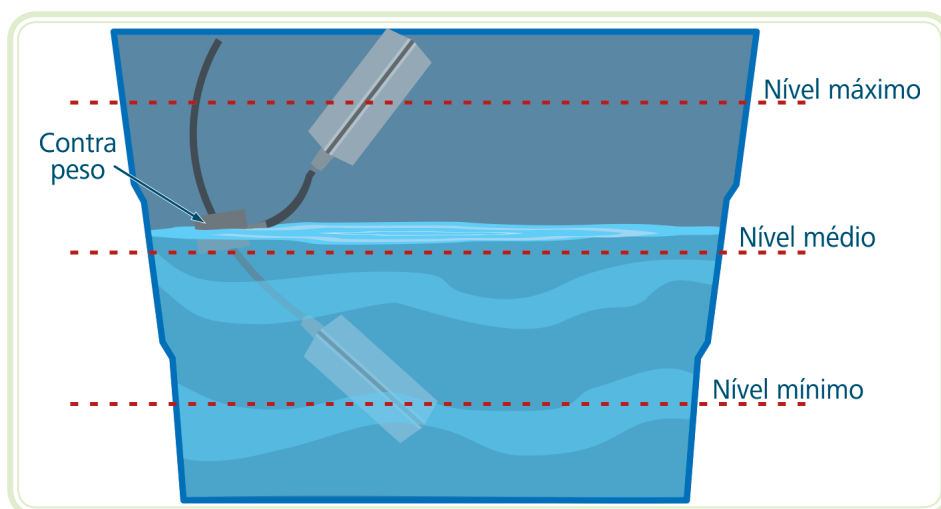


Figura 2.3: Forma de utilização do sensor do tipo boia

Fonte: CTISM

Já a Figura 2.4 mostra o botão do tipo microinterruptor escolhido para o projeto usado como exemplo.



Figura 2.4: Botão do tipo microinterruptor

Fonte: MarGirius, 2014b

Do ponto de vista funcional, os sensores de nível e o botão se comportam com chaves abertas quando não estão acionados. Sendo assim, para impedir que o pino do microcontrolador fique “flutuando” (sem um nível definido de tensão presente) nesta condição, foram acrescentados resistores de “pull-up” no valor de 10 k Ω para cada uma das entradas como mostrado na Figura 2.5.

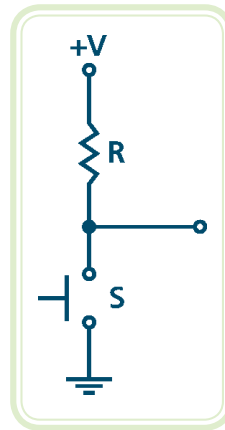


Figura 2.5: Ligação de uma botão com um resistor de “pull-up”

Fonte: CTISM, adaptado dos autores



Para saber mais sobre relés, acesse:

<http://pt.wikipedia.org/wiki/Relé>

<http://eletronicos.hsw.uol.com.br/rele.htm>

<http://www.newtonbraga.com.br/index.php/como-funciona/597-como-funcionam-os-reles?showall=&limitstart=>

Para a especificação adequada dos acionamentos das saídas, devemos conhecer as características das cargas que serão alimentadas. Considerando que a bomba de enchimento da caixa d’água é acionada por um motor monofásico de 127 Vca/2 H.P. (~ 11,8 A) e que a potência da sirene é 1200 W também em 127 Vca, serão usados dois relés de estado sólido da Metaltex, modelo TSZC-25A (mostrado na Figura 2.6), já que a característica de acionamento deste dispositivo (tensão de entrada: 4 a 32 Vcc/corrente de entrada: máx. 12 mA) permite que ele seja acionado diretamente pelo microcontrolador escolhido.



Figura 2.6: Relé de estado sólido

Fonte: Metaltex, 2014b

A Figura 2.7 apresenta o diagrama elétrico com as conexões entre todos os componentes comentados até o momento.

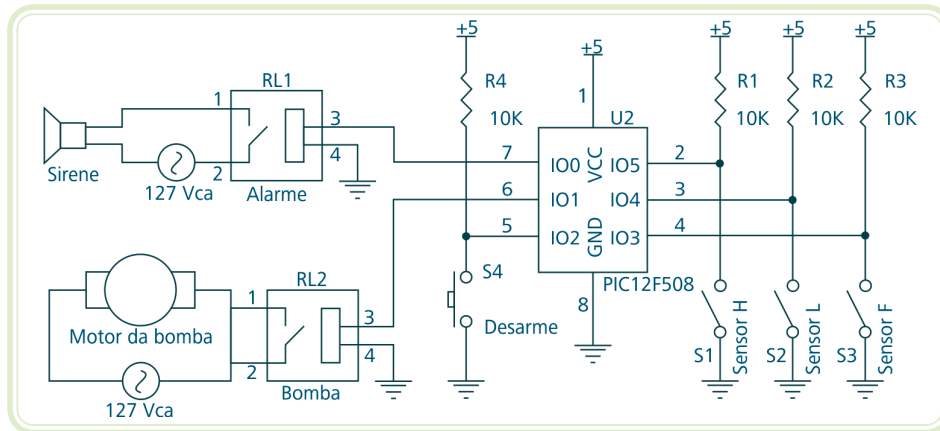


Figura 2.7: Diagrama elétrico para o controle de nível da caixa d'água

Fonte: CTISM, adaptado dos autores

A última parte do circuito a ser especificada é a fonte de alimentação. Utilizamos uma fonte linear convencional (mostrada na Figura 2.8) devido à intenção de um baixo custo para o sistema. Ela foi baseada num Circuito Integrado (CI) regulador de tensão LM7805, fabricado pela Fairchild e outras empresas, já que a tensão de alimentação escolhida foi 5 Vcc.

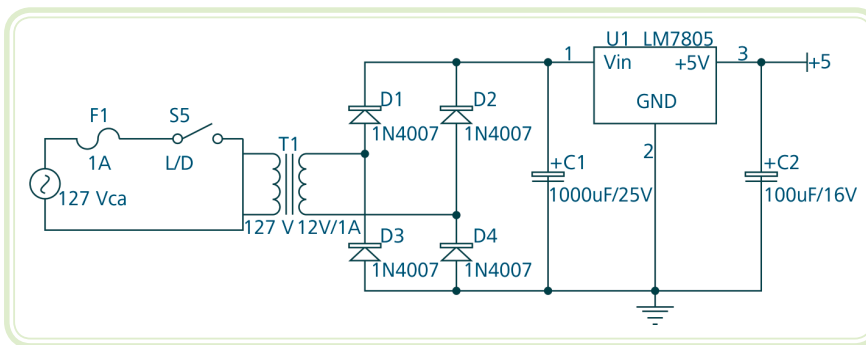


Figura 2.8: Fonte de alimentação linear convencional sugerida

Fonte: CTISM, adaptado dos autores

A função dos componentes que fazem parte desta fonte de alimentação é mostrada a seguir:

- **F1** – fusível de proteção contra sobre-corrente, oriunda normalmente de curtos-circuitos indevidos entre partes do sistema.
- **S5** – chave liga-desliga geral do sistema de automação.
- **T1** – transformador usado para a redução da tensão da rede elétrica (127 Vca) para uma outra mais adequada para a alimentação do circuito (12 Vca).
- **D1 a D4** – diodos ligados em ponte para a retificação em onda completa da tensão alternada presente no secundário do transformador.

- **C1** – capacitor eletrolítico de filtragem da tensão retificada pela ponte de diodos.
- **U1** – circuito integrado regulador de tensão (5 Vcc).
- **C2** – capacitor para a estabilização do funcionamento do circuito integrado regulador de tensão (uso indicado pelo fabricante do CI).

É importante observar que a escolha de uma fonte linear para a alimentação do circuito também foi influenciada pelo seu pequeno consumo. Em situações onde este consumo é mais elevado devido ao acionamento de elementos de alta potência ou outras características específicas, pode ser recomendado o uso de fontes chaveadas, já elas apresentam normalmente uma eficiência maior do que as lineares para as mesmas condições de operação. No entanto, seu custo também é usualmente mais alto.

Resumo

Nessa aula, o aluno(a) teve a oportunidade de analisar o diagrama elétrico básico para um projeto de automação baseado em microcontrolador, compreendendo as etapas envolvidas na sua elaboração e na especificação dos componentes periféricos de entrada (sensores) e de saída (atuadores), bem como da fonte de alimentação de todo o circuito.

Esperamos que este conhecimento tenha despertado em você o desejo de elaborar o diagrama elétrico dos seus próprios projetos e que esteja com a vontade firme de prosseguir na aprendizagem deste mundo fantástico chamado automação. Vamos continuar então?



Atividades de aprendizagem

1. Explique com suas palavras o que é um diagrama elétrico e qual é a sua importância para a automação de um sistema.
2. Desenvolva um diagrama elétrico para cada uma das três situações que serão automatizadas e propostas no item 2 das Atividades de aprendizagem da Aula 1.
3. Indique a função de cada um dos componentes usados nos três diagramas elaborados no item anterior, para os projetos de automação propostos.

Aula 3 – Programação do controlador

Objetivos

Conhecer as vantagens de um sistema automatizado programável em relação aos não programáveis.

Compreender a importância da programação para o correto funcionamento de um sistema automatizado.

Desenvolver os algoritmos que servirão de base para a elaboração de programas.

Analisar e compreender os programas de outros projetos.

Desenvolver os primeiros programas para a automação de pequenos sistemas.

3.1 Função e importância

Em sistemas não programáveis, a forma de operação do conjunto é determinada basicamente pelo modo que as partes que o compõem são interligadas. Dessa forma, a mudança desta forma de operar normalmente é muito complicada, ou até mesmo, inviável, sendo que em alguns casos extremos é mais recomendável substituir todo o sistema do que tentar alterá-lo.

Já nos sistemas programáveis, a forma de operação do conjunto é determinada por uma sequência estruturada de comandos ou instruções que comumente chamamos de programa.

Este programa pode ser escrito normalmente de várias formas de acordo com o tipo do elemento de controle utilizado. Em sistemas microcontrolados, as linguagens de programação Assembly, C e BASIC são muito utilizadas. Já para sistemas baseados em CLPs, a linguagem mais utilizada é o Ladder.

3.2 Elaboração do algoritmo

O desenvolvimento do programa de controle do sistema é baseado no diagrama elétrico elaborado anteriormente. No caso específico dos microcontroladores, ele é usualmente chamado de *firmware*.



Para saber mais sobre algoritmos, acesse: <http://pt.wikipedia.org/wiki/Algoritmo>

Antes da programação propriamente dita, é altamente recomendado que elaboremos previamente um algoritmo, que representará de forma organizada todo o funcionamento esperado para o sistema. Dessa forma, além de facilitar a futura codificação, podemos nos assegurar, com um maior grau de confiança, que previmos todas as relações de funcionamento entre as entradas e saídas do sistema.

Baseando-se na explicação da solução proposta do projeto do controle de nível da caixa d'água, um dos algoritmos possíveis que podem ser usados nesta situação é mostrado a seguir:

```
Início:
  Se Sensor_Falha == 0 Então
    Alarme = 1
    Bomba = 0
  Senão
    Se Sensor_Nivel_Baixo == 1 Então
      Bomba = 1
    Fim do Se
    Se Sensor_Nivel_Alto == 0 Então
      Bomba = 0
    Fim do Se
  Fim do Se
  Se Botao_Desarme == 0 Então
    Alarme = 0
  Fim do Se
Salto p/ Início
```

Neste algoritmo, o algarismo "0" para as entradas representa que elas estão acionadas (contato fechado), enquanto o algarismo "1" representa que elas estão não acionadas (contato aberto).

Já para as saídas, o algarismo "0" representa que elas estão ativadas (ligadas), enquanto o algarismo "1" representa que elas estão desativadas (desligadas).

3.3 Codificação em uma linguagem específica

A conversão do algoritmo em uma linguagem de programação qualquer costuma ser uma operação direta, com algumas poucas e eventuais adequações. No nosso caso, como optamos por um microcontrolador PIC12F508 da Microchip, o mais comum é que a linguagem de programação escolhida seja o Assembly, o C ou o BASIC. A translação do algoritmo elaborado em cada uma destas linguagens é mostrada a seguir.

- Código na linguagem Assembly

```
;INCLUSÃO DE ARQUIVOS
#include <p12f508.inc>

;CONFIGURAÇÃO DO MODO DE FUNCIONAMENTO
__CONFIG _MCLRE_OFF & _WDT_OFF & _PWRTE_OFF & _CP_OFF
& _BODEN_OFF & _INTRC_OSC_NOCLKOUT & _CPD_OFF

;DEFINIÇÃO DAS ENTRADAS DIGITAIS
#define Botao_Desarme GPIO, 2 ;Pino 5
#define Sensor_Falha GPIO, 3 ;Pino 4
#define Sensor_Nivel_Baixo GPIO, 4 ;Pino 3
#define Sensor_Nivel_Alto GPIO, 5 ;Pino 2

;DEFINIÇÃO DAS SAÍDAS DIGITAIS
#define Bomba GPIO, 1 ;Pino 6
#define Alarme GPIO, 0 ;Pino 7

;CONFIGURAÇÃO DOS SFR'S
BSF STATUS, RPO
MOVLW B'00111100'
MOVWF TRISIO
BCF STATUS, RPO

;INICIALIZAÇÃO DO PROGRAMA
BCF Bomba
BCF Alarme
```



Para saber mais sobre a linguagem Assembly, acesse: <http://pt.wikipedia.org/wiki/Assembly>

```

;ROTINA PRINCIPAL
Principal:
    BTFSS Sensor_Falha
    CALL Transbordamento

    BTFSC Sensor_Falha
    CALL Funcionamento_Normal

    BTFSS Botao_Desarme
    BCF Alarme

    GOTO Principal ;Salto para o início

;SUBROTINAS
Transbordamento:
    BSF Alarme
    BCF Bomba
    RETURN

Funcionamento_Normal:
    BTFSC Sensor_Nivel_Baixo
    BSF Bomba

    BTFSS Sensor_Nivel_Alto
    BCF Bomba
    RETURN

END

```

- Código na linguagem C (compilador MICROCHIP XC8)

```

//INCLUSÃO DE ARQUIVOS
#include<xc.h>

//CONFIGURAÇÃO DO MODO DE FUNCIONAMENTO
__CONFIG(FOSC_INTRCIO & WDTE_OFF & CP_OFF & CPD_OFF &
MCLRE_OFF & BOREN_OFF & PWRTE_OFF);

```

```

//DEFINIÇÃO DAS ENTRADAS DIGITAIS
#define Botao_Desarme GPIO2 //Pino 5
#define Sensor_Falha GPIO3 //Pino 4
#define Sensor_Nivel_Baixo GPIO4 //Pino 3
#define Sensor_Nivel_Alto GPIO5 //Pino 2

//DEFINIÇÃO DAS SAÍDAS DIGITAIS
#define Bomba GPIO1 //Pino 6
#define Alarme GPIO0 //Pino 7

//FUNÇÃO PRINCIPAL
void main(void) {
    //CONFIGURAÇÃO DOS SFR'S
    TRISIO = 0b00111100;

    //INICIALIZAÇÃO DO PROGRAMA
    Bomba = 0;
    Alarme = 0;

    //LAÇO CONTÍNUO
    while(1) {
        if (Sensor_Falha == 0) {
            Alarme = 1;
            Bomba = 0;
        }
        else {
            if (Sensor_Nivel_Baixo == 1) {
                Bomba = 1;
            }
            if (Sensor_Nivel_Alto == 0) {
                Bomba = 0;
            }
        }
        if (Botao_Desarme == 0) {
            Alarme = 0;
        }
    }
}

```



Para saber mais sobre a linguagem C, acesse: http://pt.wikipedia.org/wiki/Linguagem_c

- Código na linguagem BASIC (compilador PIC BASIC PRO)



Para saber mais sobre
linguagem Basic, acesse:
<http://pt.wikipedia.org/wiki/BASIC>

```

'INCLUSÃO DE ARQUIVOS E CONFIGURAÇÃO INICIAL
@ DEVICE PIC12F508, WDT_OFF, PWRT_OFF, PROTECT_OFF, BOD_OFF,
MCLR_OFF, INTRC_OSC_NOCLKOUT

'DEFINIÇÃO DAS ENTRADAS DIGITAIS
Botao_Desarme VAR GPIO.2 'Pino 5
Sensor_Falha VAR GPIO.3 'Pino 4
Sensor_Nivel_Baixo VAR GPIO.4 'Pino 3
Sensor_Nivel_Alto VAR GPIO.5 'Pino 2

'DEFINIÇÃO DAS SAÍDAS DIGITAIS
Bomba VAR GPIO.1 'Pino 6
Alarme VAR GPIO.0 'Pino 7

'CONFIGURAÇÃO DOS SFR'S
TRISIO = %00111100

'INICIALIZAÇÃO DO PROGRAMA
Bomba = 0
Alarme = 0

'ROTINA PRINCIPAL
Principal:
  IF Sensor_F == 0 THEN
    Alarme = 1
    Bomba = 0
  ELSE
    IF Sensor_L == 1 THEN
      Bomba = 1
    ENDIF
    IF Sensor_H == 0 THEN
      Bomba = 0
    ENDIF
  ENDIF
  IF Botao_Desarme == 0 THEN
    Alarme = 0
  ENDIF
GOTO Principal
END

```


Resumo

Vimos nessa aula a grande versatilidade dos sistemas de automação programáveis, quando comparados com os não programáveis, além de compreender a importância da elaboração de algoritmo previamente antes de partirmos para a codificação propriamente dita.

Você também teve a oportunidade de analisar o processo de translação de um algoritmo para três linguagens de programação (Assembly, C e BASIC) diferentes para microcontroladores a partir do exemplo do controle de nível da caixa d'água.

Para a fixação de todo este conteúdo, é muito importante que faça as atividades de aprendizagem. De qualquer forma, não podemos deixar de parabenizá-lo neste momento, pois, avançamos bastante em nossas atividades e você continua firme conosco!

Atividades de aprendizagem

1. Indique as vantagens dos sistemas automatizados programáveis em relação aos não programáveis.
2. Explique com suas palavras qual a importância de se elaborar um algoritmo antes de iniciar a codificação para um projeto.
3. Desenvolva um algoritmo para cada uma das três situações que serão automatizadas e propostas no item 2, das Atividades de aprendizagem da Aula 1.
4. Converta cada um dos algoritmos elaborados no item anterior para as seguintes linguagens de programação de microcontroladores: Assembly, C e BASIC.



Aula 4 – Lista de material

Objetivos

Compreender a importância da lista de material para um projeto de automação.

Conhecer os pontos que devem ser observados para a elaboração da lista de material.

Aprender quais são as partes mínimas e quais são as partes recomendáveis para fazer parte de uma lista de material.

Desenvolver as primeiras listas de materiais para a automação de pequenos sistemas.

4.1 Função e importância

A lista de material, também chamada de BOM (do inglês *Bill of Material*), é utilizada tanto para a aquisição inicial das peças e componentes, que serão usados na montagem do circuito de automação do sistema, quanto a reposição de partes no caso da sua manutenção.

4.2 Observações para a elaboração

A lista de material de um projeto deve ser a mais completa possível, contendo, no mínimo, a indicação da quantidade e a descrição detalhada de todas as partes e de todos os componentes utilizados. Além disso, a indicação de um ou mais fabricantes dos itens, com os seus respectivos códigos de designação, é altamente recomendada, já que os profissionais do setor de compras das empresas não tem de ter conhecimentos técnicos sobre os produtos adquiridos.

A indicação dos códigos de designação também facilita a comparação de preços e a busca por similares no caso da indisponibilidade temporária ou permanente de um determinado item.

O Quadro 4.1, mostrado a seguir, exibe uma sugestão de lista de material que segue estas orientações para a situação proposta do controle de nível de uma caixa d'água.

Quadro 4.1: Lista de material do projeto baseado em microcontrolador			
Item	Descrição	Código	Fabricante
C1	Capacitor 1000 μ F/25 V	ECA-1EM102B	Panasonic
		UVZ1E102MPD	Nichicon
C2	Capacitor 100 μ F/16 V	ECA-1CM101I	Panasonic
		UVR1C101MDD	Nichicon
D1 a D4	Diodo retificador	1N4007	NXP
			Fairchild
R1 a R4	Resistor 10 k Ω – 1/8 W	CF18JT10K0	Stackpole
RL1 e RL2	Relé de estado sólido	TSZC-25A	Metaltext
	Dissipador de calor	TSZC-DT2	Metaltext
S1 a S3	Chave de nível tipo boia	CB2002	MarGirius
S4	Microinterruptor	18531 MFT3F ST S E	MarGirius
S5	Interruptor de tecla	13101 MF AF S1 S	MarGirius
T1	Transformador 127 V/12 V	DP-241-5-12	Signal
U1	Regulador de tensão	LM7805	Fairchild
U2	Microcontrolador	PIC12F508-I/P	Microchip
F1	Fusível de vidro – 500 mA	GMC-500-R	Bussmann
	Porta fusível	I02	Metaltext

Fonte: Autores



A indicação de códigos de outros fabricantes, quando possível, é recomendada por facilitar a aquisição inicial e eventuais reposições.

4.3 Itens indiretos necessários

Um erro comum dos profissionais, que estão iniciando na atividade de elaboração da lista de material para projetos, é não relacionar componentes complementares àqueles que aparecem diretamente no diagrama elétrico.

Geralmente, os componentes complementares permitem que outros elementos funcionem corretamente ou servem de base para que esses possam ser adequadamente usados.

Como exemplo da primeira situação, no nosso caso, temos os dissipadores de calor (mostrado na Figura 4.1) para os relés de estado sólido, já que sem eles estes componentes irão queimar devido à alta temperatura gerada durante a sua operação.

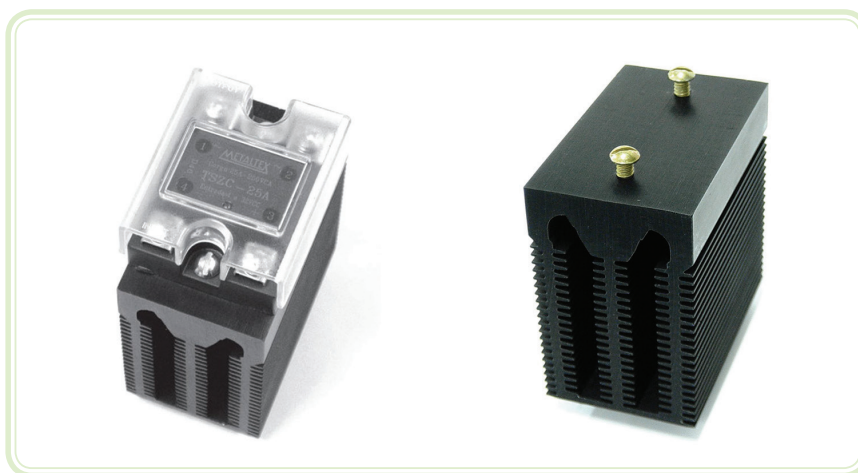


Figura 4.1: Dissipador de calor para relé de estado sólido

Fonte: Metaltex, 2014b

Em relação à segunda situação, citamos o exemplo do porta fusível (mostrado na Figura 4.2) para o fusível de vidro, usado como elemento de proteção contra uma elevação anormal da corrente que circula pelo circuito.



Figura 4.2: Porta fusível

Fonte: Metaltex, 2014a



Para saber mais sobre fusível, acesse: <http://pt.wikipedia.org/wiki/Fusível>

4.4 Itens não relacionados

Aprendemos a elaborar, nesta aula, uma lista de material conhecida como lista de componentes, já que ela relaciona todos os componentes que serão usados em um projeto.

No entanto, ela não contempla outros materiais que serão necessários para a montagem e/ou instalação física do sistema especificado em campo, como fios, solda, parafusos, caixas, eletrodutos, etc. Sendo assim, é fundamental que o profissional da área da automação esteja atento a este pormenor.

Um último ponto que gostaríamos de salientar nessa aula é a forma da montagem dos componentes eletrônicos no caso de optarmos por controlar o sistema de automação através de um microcontrolador.



Para saber mais sobre placas de circuito impresso, acesse: http://pt.wikipedia.org/wiki/Placa_de_circuito_impreso

Usualmente, esta montagem é feita numa Placa de Circuito Impresso (PCI) dedicada, especialmente projetada e fabricada de acordo com o nosso projeto, ou numa placa de circuito impresso universal ou padrão, onde os componentes são interligados comumente através de pequenos pedaços de fio, mostradas nas Figuras 4.3 e 4.4, respectivamente.

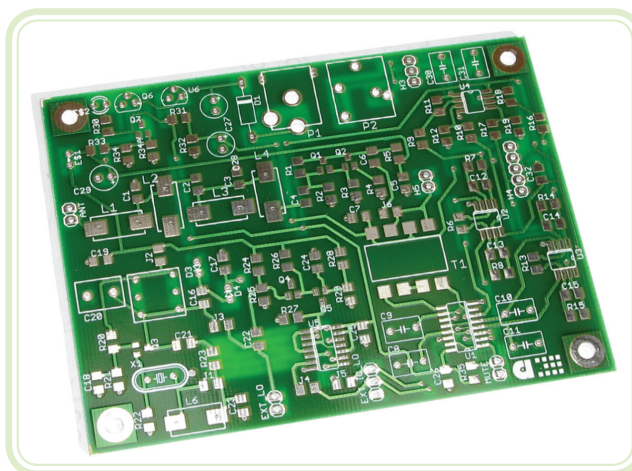


Figura 4.3: Placa de circuito impresso (PCI) dedicada

Fonte: http://2.bp.blogspot.com/-4H11x-GPPUc/UFMGiP8KVPI/AAAAAAAAAJQ/O_a-5R5vCM8/s1600/IMG_0832.jpg

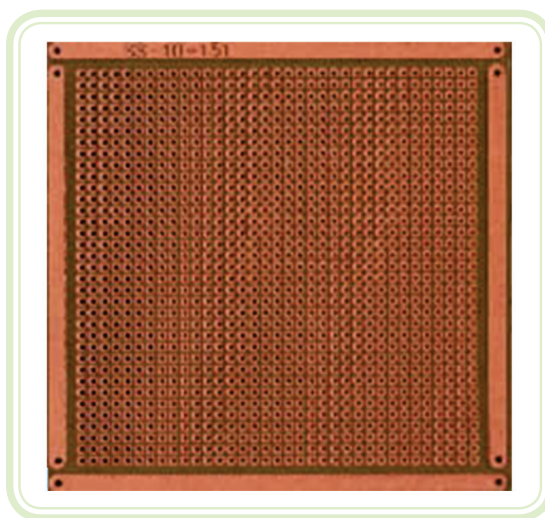


Figura 4.4: Placa de circuito impresso (PCI) universal ou padrão

Fonte: Eletrônica Digital, 2014

A primeira solução apresenta como vantagens principais: a confiabilidade, o aspecto profissional da montagem e um tempo bem menor para a montagem. A sua desvantagem fica por conta do custo mais elevado. Já em relação a segunda solução, a única vantagem realmente é o custo, pois ela perde em todos os outros pontos analisados.

Resumo

Nessa aula, você foi apresentado a um tópico muito importante para a automação de um sistema, que normalmente (infelizmente e erroneamente) é subestimado: a lista de material.

Você também teve a oportunidade de compreender o valor de descrição detalhada dos componentes, além de perceber que existem componentes (elementos complementares) que não aparecem diretamente no diagrama elétrico, mas que são muito importantes para o funcionamento de todo o sistema.

Ao concluir esta lição, você aprendeu os passos básicos da metodologia proposta para a automação de sistemas. Parabéns e continue firme!

Atividades de aprendizagem



1. Explique com suas palavras qual a importância de uma lista de material bem elaborada para um projeto de automação.
2. Indique os fornecedores e outros códigos de requisição para todos os itens da lista de material apresenta no Quadro 4.1.
3. Elabore uma lista de material para cada uma das três situações que serão automatizadas e propostas no item 2 das Atividades de aprendizagem da Aula 1. Cite, sempre que possível, pelo menos o código de dois fabricantes para cada item.

Aula 5 – Variações de projeto

Objetivos

Compreender a necessidade de se modificar um projeto para facilitar a sua utilização na prática.

Desenvolver alterações que propiciem ao usuário de um sistema um conjunto maior de informações visuais sobre o seu estado atual.

Utilizar LEDs para indicar de forma simples o estado de um sistema.

Aprender como usar módulos LCD para exibir mensagens em projetos que empreguem microcontroladores.

5.1 Alterações no projeto original

O projeto de exemplo, desenvolvido ao longo das Aulas 1 a 4, priorizava apenas o funcionamento do sistema de acordo com as especificações, não prevendo, no entanto, uma forma de sinalização visual do seu estado para os usuários.

Em várias situações essa sinalização pode ser fundamental, e uma variação do projeto original que continha apenas o mínimo para o funcionamento básico pode ser implementada para atender a este novo requisito.

Nessa aula, veremos como implementar uma sinalização visual de estado do sistema mais simples, usando apenas LEDs e uma mais sofisticada usando um módulo LCD.

5.2 Sinalização de funcionamento com LEDs

Em relação ao projeto original, foram acrescentados três LEDs com as seguintes funções:

- LD1 – indicação visual de funcionamento do sistema de controle.
- LD2 – indicação visual de acionamento da bomba.

- LD3 – indicação visual de acionamento da sirene.

Neste caso, onde os LEDs LD2 e LD3 foram colocados em paralelo com os relés de estado sólido, é importante verificar se o microcontrolador utilizado possui a capacidade de fornecer corrente elétrica diretamente para todas estas cargas ou se será necessário empregar um dispositivo de potência, como um transistor para acioná-las.

Como o PIC12F508 pode fornecer ou absorver até 25 mA em cada um dos seus pinos de E/S (informação obtida no datasheet do dispositivo), foi possível fazer a ligação dos LEDs diretamente, só usando os resistores de 560 Ω como limitadores de corrente.

O novo diagrama elétrico com a inclusão dos LEDs é mostrado na Figura 5.1, enquanto a nova lista de material para essa situação é mostrada no Quadro 5.1. É importante salientar que como não houve alterações no modo de funcionamento do sistema, o programa de controle do sistema não sofreu modificações e pode ser o mesmo desenvolvido na Aula 3.

Quadro 5.1: Lista de material do projeto com LEDs de indicação de estado			
Item	Descrição	Código	Fabricante
C1	Capacitor 1000 μ F/25 V	ECA-1EM102B	Panasonic
		UVZ1E102MPD	Nichicon
C2	Capacitor 100 μ F/16 V	ECA-1CM101I	Panasonic
		UVR1C101MDD	Nichicon
D1 a D4	Diodo retificador	1N4007	NXP
			Fairchild
LD1	LED verde de 3 mm	MV5774C	Everlight
LD2	LED amarelo de 3 mm	MV5474C	Everlight
LD3	LED vermelho de 3 mm	MV5374C	Everlight
R1 a R4	Resistor 10 k Ω – 1/8 W	CF18JT10K0	Stackpole
R5 a R7	Resistor 560 Ω – 1/8 W	CF18JT560R	Stackpole
RL1 e RL2	Relé de estado sólido	TSZC-25A	Metaltext
	Dissipador de calor	TSZC-DT2	Metaltext
S1 a S3	Chave de nível tipo boia	CB2002	MarGirius
S4	Microinterruptor	18531 MF T3F ST S E	MarGirius
S5	Interruptor de tecla	13101 MF AF S1 S	MarGirius
T1	Transformador 127 V/12 V	DP-241-5-12	Signal
U1	Regulador de tensão	LM7805	Fairchild
U2	Microcontrolador	PIC12F508-I/P	Microchip
F1	Fusível de vidro – 500 mA	GMC-500-R	Bussmann
	Porta fusível	I02	Metaltext

Fonte: Autores

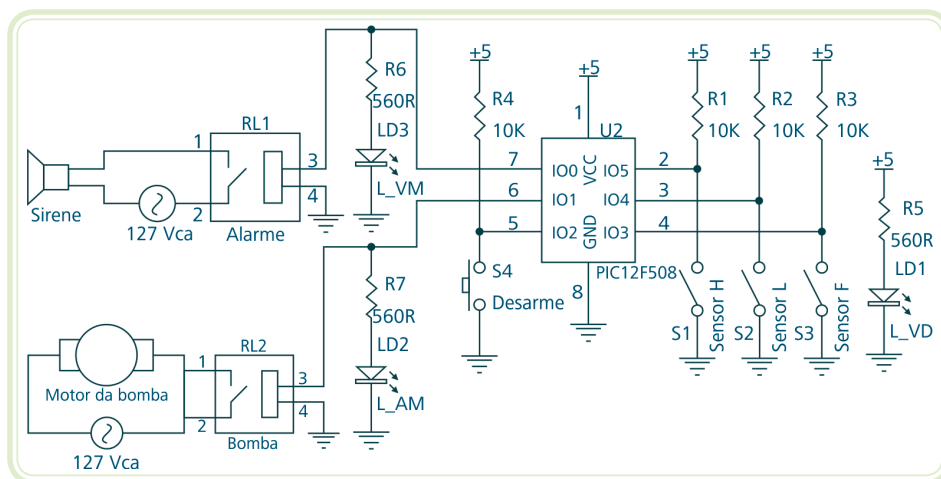


Figura 5.1: Diagrama elétrico do projeto com LEDs de indicação de estado

Fonte: CTISM, adaptado dos autores

5.3 Utilização de módulos LCD

Em muitas aplicações, somente a utilização de LEDs é o suficiente para passar informações do sistema aos usuários. No entanto, há situações que só isso não é o bastante, principalmente quando a quantidade e variedade de informações é grande.

Quando isso ocorre, pode ser conveniente a utilização de um Display de Cristal Líquido (LCD – *Liquid Cristal Display*) para a exibição de mensagens de texto apropriadas para os usuários.

Os módulos LCD típicos mais comuns no mercado possuem um processador próprio e podem se comunicar usualmente com um microcontrolador por meio de seis pinos. Sendo assim, para a inclusão de um módulo desses no projeto, foi necessário substituir o PIC12F508 por um outro microcontrolador com uma disponibilidade maior de pinos de E/S. Optamos, dessa forma, por um PIC16F877A, cuja utilização também é muito difundida.

O novo diagrama elétrico com a inclusão do módulo LCD é mostrado na Figura 5.2, enquanto a nova lista de material para essa situação é mostrada no Quadro 5.2.

Um novo código-fonte, escrito em C e mostrado a seguir, foi desenvolvido para essa nova variação de projeto. Para que possa haver a comunicação entre o PIC16F877A e o módulo LCD, foi inserida uma biblioteca (`flex_lcd.h`) no início do programa que contém todas as rotinas necessárias para isso.

No caso da programação em BASIC, também mostrada na sequência, não é necessário incluir uma biblioteca no código-fonte porque o compilador já prevê comandos para controlador em um módulo LCD. No entanto, é necessário indicar quais pinos do microcontrolador serão utilizados na comunicação entre os dois dispositivos.

- Código na linguagem C (compilador MICROCHIP XC8)

```
//FREQUÊNCIA DO CLOCK
#define _XTAL_FREQ 4000000

//INCLUSÃO DE ARQUIVOS
#include <xc.h> //Definições do PIC utilizado
#include "flex_lcd.h" //Comunicação com o módulo LCD

//CONFIGURAÇÃO DO MODO DE FUNCIONAMENTO
__CONFIG (CP_OFF & CPD_OFF & BOREN_OFF & WDTE_OFF & PWRTE_
OFF);

//DEFINIÇÃO DAS ENTRADAS DIGITAIS
#define Botao_Desarme      PORTDbits.RD3 //Pino 22
#define Sensor_Falha      PORTDbits.RD2 //Pino 21
#define Sensor_Nivel_Baixo PORTDbits.RD1 //Pino 20
#define Sensor_Nivel_Alto  PORTDbits.RD0 //Pino 19

//DEFINIÇÃO DAS SAÍDAS DIGITAIS
#define Bomba              PORTBbits.RB0 //Pino 33
#define Alarme             PORTBbits.RB1 //Pino 34

//FUNÇÃO PRINCIPAL
void main(void) {
    //CONFIGURAÇÃO DOS SFR'S
    TRISD = 0b11111111; //Pinos do PortD = Entradas
    TRISB = 0b00000000; //Pinos do PortB = Saídas
```

```

//INICIALIZAÇÃO DO PROGRAMA
Bomba = 0; //Desligamento da Bomba
Alarme = 0; //Desligamento da Sirene
Lcd_Init( ); //Inicialização do LCD
Lcd_Cmd(LCD_CLEAR); //Apagamento do LCD
Lcd_Cmd(LCD_CURSOR_OFF); //Desligamento do cursor
__delay_ms(100); //Pausa de inicialização

//MENSAGEM DE ABERTURA
Lcd_Out(1, 3, "Projetos de");
Lcd_Out(2, 4, "Automacao");
__delay_ms(1500); //Pausa de 1500 ms (1,5 s)
Lcd_Cmd(LCD_CLEAR); //Apagamento do LCD
Lcd_Out(1, 0, "SISTEMA: Normal ");
Lcd_Out(2, 0, "BOMBA: ");

//LAÇO CONTÍNUO
while(1) {
    if (Sensor_Falha == 0) {
        Bomba = 0;
        Alarme = 1;
        Lcd_Out(1, 0, "***** ALARME *****");
        Lcd_Out(2, 7, "Desligada");
    }
    else {
        if (Sensor_Nivel_Baixo == 1) {
            Bomba = 1;
            Lcd_Out(2, 7, "Ligada ");
        }
        if (Sensor_Nivel_Alto == 0) {
            Bomba = 0;
            Lcd_Out(2, 7, "Desligada");
        }
    }
    if (Botao_Desarme == 0) {
        Alarme = 0;
        Lcd_Out(1, 0, "SISTEMA: Normal ");
    }
}
}

```

- Biblioteca **flex_lcd.h** em C (compilador MICROCHIP XC8)

```
#ifndef __flex_lcd_H
#define __flex_lcd_H

#define LCD_RD7 PORTCbits.RC7           // D7
#define TRISR7 TRISCbits.TRISC7

#define LCD_RD6 PORTCbits.RC6           // D6
#define TRISR6 TRISCbits.TRISC6

#define LCD_RD5 PORTCbits.RC5           // D5
#define TRISR5 TRISCbits.TRISC5

#define LCD_RD4 PORTCbits.RC4           // D4
#define TRISR4 TRISCbits.TRISC4

#define LCD_EN PORTCbits.RC3            // EN
#define TRISEN TRISCbits.TRISC3

#define LCD_RS PORTCbits.RC2            // RS
#define TRISRS TRISCbits.TRISC2

#define LCD_FIRST_ROW                   128
#define LCD_SECOND_ROW                   192
#define LCD_THIRD_ROW                    148
#define LCD_FOURTH_ROW                   212
#define LCD_CLEAR                         1
#define LCD_RETURN_HOME                   2
#define LCD_CURSOR_OFF                    12
#define LCD_UNDERLINE_ON                  14
#define LCD_BLINK_CURSOR_ON               15
#define LCD_MOVE_CURSOR_LEFT              16
#define LCD_MOVE_CURSOR_RIGHT             20
#define LCD_TURN_OFF                      0
#define LCD_TURN_ON                        8
#define LCD_SHIFT_LEFT                    24
#define LCD_SHIFT_RIGHT                   28
```

```

void Lcd_Init(void);
void Lcd_Out(unsigned char y, unsigned char x, const char
*buffer);
void Lcd_Out2(unsigned char y, unsigned char x, char *buffer);
void Lcd_Chr_CP(char data);
void Lcd_Cmd(unsigned char data);

void Lcd_Init(void) {
    unsigned char data;
    TRISRD7 = 0;          TRISRD6 = 0;
    TRISRD5 = 0;          TRISRD4 = 0;
    TRISEN = 0;           TRISRS = 0;
    LCD_RD7 = 0;          LCD_RD6 = 0;
    LCD_RD5 = 0;          LCD_RD4 = 0;
    LCD_EN = 0;           LCD_RS = 0;

    __delay_us(5500);
    __delay_us(5500);
    __delay_us(5500);
    __delay_us(5500);
    __delay_us(5500);
    __delay_us(5500);

    for(data = 1; data < 4; data ++) {
        LCD_RD7 = 0;   LCD_RD6 = 0;   LCD_RD5 = 1;
        LCD_RD4 = 1;   LCD_EN = 0;   LCD_RS = 0;
        LCD_RD7 = 0;   LCD_RD6 = 0;   LCD_RD5 = 1;
        LCD_RD4 = 1;   LCD_EN = 1;   LCD_RS = 0;
        __delay_us(5);
        LCD_RD7 = 0;   LCD_RD6 = 0;   LCD_RD5 = 1;
        LCD_RD4 = 1;   LCD_EN = 0;   LCD_RS = 0;
        __delay_us(5500);
    }

    LCD_RD7 = 0;   LCD_RD6 = 0;   LCD_RD5 = 1;
    LCD_RD4 = 0;   LCD_EN = 0;   LCD_RS = 0;
    LCD_RD7 = 0;   LCD_RD6 = 0   LCD_RD5 = 1;
    LCD_RD4 = 0;   LCD_EN = 1;   LCD_RS = 0;
    __delay_us(5);
    LCD_RD7 = 0;   LCD_RD6 = 0;   LCD_RD5 = 1;

```

```

LCD_RD4 = 0;      LCD_EN = 0;      LCD_RS = 0;
__delay_us(5500);
data = 40;      Lcd_Cmd(data);
data = 16;      Lcd_Cmd(data);
data = 1;      Lcd_Cmd(data);
data = 15;      Lcd_Cmd(data);
}

```

```

void Lcd_Out(unsigned char y, unsigned char x, const char
*buffer) {
    unsigned char data;
    switch (y) {
        case 1: data = 128 + x; break;
        case 2: data = 192 + x; break;
        case 3: data = 148 + x; break;
        case 4: data = 212 + x; break;
        default: break;
    }
    Lcd_Cmd(data);
    while(*buffer) {
        Lcd_Chr_CP(*buffer); buffer++;
    }
    return;
}

```

```

void Lcd_Out2(unsigned char y, unsigned char x, char *buffer) {
    unsigned char data;
    switch (y) {
        case 1: data = 128 + x; break;
        case 2: data = 192 + x; break;
        case 3: data = 148 + x; break;
        case 4: data = 212 + x; break;
        default: break;
    }
    Lcd_Cmd(data);
    while(*buffer)
        Lcd_Chr_CP(*buffer); buffer++;
    }
    return;
}
void Lcd_Chr_CP(char data){

```



```

    LCD_EN = 0; LCD_RS = 1;
    LCD_RD7 = (data & 0b10000000)>>7; LCD_RD6 = (data &
0b01000000)>>6;
    LCD_RD5 = (data & 0b00100000)>>5; LCD_RD4 = (data &
0b00010000)>>4;
    _delay(10);
    LCD_EN = 1; __delay_us(5); LCD_EN = 0;
    LCD_RD7 = (data & 0b00001000)>>3; LCD_RD6 = (data &
0b00000100)>>2;
    LCD_RD5 = (data & 0b00000010)>>1; LCD_RD4 = (data &
0b00000001);
    _delay(10);
    LCD_EN = 1; __delay_us(5); LCD_EN = 0;
    __delay_us(5); __delay_us(5500);
}
void Lcd_Cmd(unsigned char data){
    LCD_EN = 0; LCD_RS = 0;
    LCD_RD7 = (data & 0b10000000)>>7; LCD_RD6 = (data &
0b01000000)>>6;
    LCD_RD5 = (data & 0b00100000)>>5; LCD_RD4 = (data &
0b00010000)>>4;
    _delay(10);
    LCD_EN = 1; __delay_us(5); LCD_EN = 0;
    LCD_RD7 = (data & 0b00001000)>>3; LCD_RD6 = (data &
0b00000100)>>2;
    LCD_RD5 = (data & 0b00000010)>>1; LCD_RD4 = (data &
0b00000001);
    _delay(10);
    LCD_EN = 1; __delay_us(5); LCD_EN = 0;
    __delay_us(5500); //Delay_5us();
}

#endif

```

- Código na linguagem BASIC (compilador PIC BASIC PRO)

```

'INCLUSÃO DE ARQUIVOS E CONFIGURAÇÃO INICIAL
@ DEVICE PIC16F877A, PROTECT_OFF, PWRT_OFF, WDT_OFF, HS_OSC,
BOD_OFF, CPD_OFF

'DEFINIÇÃO DAS ENTRADAS DIGITAIS
Botao_Desarme VAR PORTD.3           'Pino 22
Sensor_Falha VAR PORTD.2           'Pino 21
Sensor_Nivel_Baixo VAR PORTD.1     'Pino 20
Sensor_Nivel_Alto VAR PORTD.0      'Pino 19

'DEFINIÇÃO DAS SAÍDAS DIGITAIS
Bomba VAR PORTB.0                   'Pino 33
Alarme VAR PORTB.1                  'Pino 34

'DEFINIÇÃO DA COMUNICAÇÃO COM O MÓDULO LCD
DEFINE LCD_DREG PORTC
DEFINE LCD_DBIT 4
DEFINE LCD_RSREG PORTC
DEFINE LCD_RSBIT 2
DEFINE LCD_EREG PORTC
DEFINE LCD_EBIT 3
DEFINE LCD_BITS 4
DEFINE LCD_LINES 2

'CONFIGURAÇÃO DOS SFR'S
TRISD = %11111111 'Pinos do PortD = Entradas
TRISB = %00000000 'Pinos do PortB = Saídas
'INICIALIZAÇÃO DO PROGRAMA
Bomba = 0 'Desligamento da Bomba
Alarme = 0 'Desligamento da Sirene
PAUSE 500 'Pausa de inicialização
LCDOUT $FE, 1 'Apagamento do LCD

```

```

'MENSAGEM DE ABERTURA
LCDOUT $FE, $83, "Projetos de"
LCDOUT $FE, $C4, "Automacao"
PAUSE 1500 'Pausa de 1500 ms (1,5 s)
LCDOUT $FE, 1 'Apagamento do LCD
LCDOUT $FE, $80, "SISTEMA: Normal "
LCDOUT $FE, $C0, "BOMBA: "

'ROTINA PRINCIPAL
Principal:
  IF Sensor_F == 0 THEN
    Alarme = 1
    Bomba = 0
    LCDOUT $FE, $80, "**** ALARME ****"
    LCDOUT $FE, $C0, "Desligada"
  ELSE
    IF Sensor_L == 1 THEN
      Bomba = 1
      LCDOUT $FE, $C7, "Ligada "
    ENDIF
    IF Sensor_H == 0 THEN
      Bomba = 0
      LCDOUT $FE, $C7, "Desligada"
    ENDIF
  ENDIF
  IF Botao_Desarme == 0 THEN
    Alarme = 0
    LCDOUT $FE, $80, "SISTEMA: Normal "
  ENDIF
GOTO Principal

END

```

Quadro 5.2: Lista de material do projeto com módulo LCD

Item	Descrição	Código	Fabricante
C1	Capacitor 1000 μ F/25 V	ECA-1EM102B	Panasonic
		UVZ1E102MPD	Nichicon
C2	Capacitor 100 μ F/16 V	ECA-1CM1011	Panasonic
		UVR1C101MDD	Nichicon
C3 e C4	Capacitor 100 nF/50 V	B37987F5104K000	EPCOS
		K104K10X7RF5UH5	Vishay
C5 e C6	Capacitor 22 pF/50 V	B37979N1220J051	EPCOS
		K220J15C0GF5TH5	Vishay
D1 a D4	Diodo retificador	1N4007	NXP
			Fairchild
LD1	LD1	LED verde de 3 mm	MV5774C
R1 a R4	Resistor 10 k Ω – 1/8 W	CF18JT10K0	Stackpole
R5	Resistor 560 Ω – 1/8 W	CF18JT560R	Stackpole
RL1 e RL2	Relé de estado sólido	TSZC-25A	Metaltex
	Dissipador de calor	TSZC-DT2	Metaltex
S1 a S3	Chave de nível tipo boia	CB2002	MarGirius
S4	Microinterruptor	18531 MF T3F ST S E	MarGirius
S5	Interruptor de tecla	13101 MF AF S1 S	MarGirius
T1	Transformador 127 V/12 V	DP-241-5-12	Signal
U1	Regulador de tensão	LM7805	Fairchild
U2	Microcontrolador	PIC16F877A-I/P	Microchip
Y1	Cristal de 4 MHz	AB-4.000MHZ-B2	Abracon
LCD1	Módulo LCD 16 \times 2	LCR-U01602DSF	Lumex
F1	Fusível de vidro – 500 mA	GMC-500-R	Bussmann
	Porta fusível	I02	Metaltex

Fonte: Autores

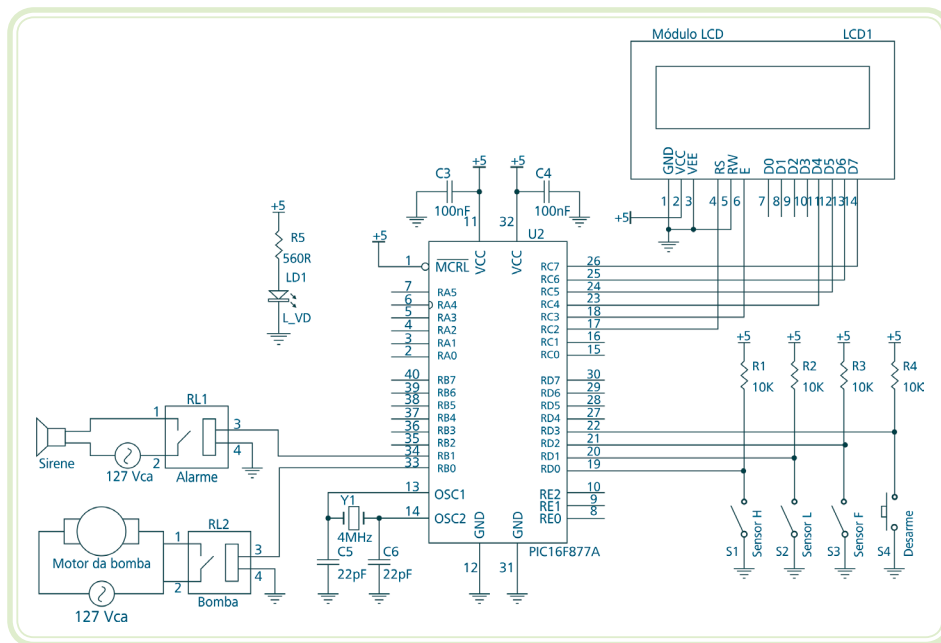


Figura 5.2: Diagrama elétrico do projeto com módulo LCD

Fonte: CTISM, adaptado dos autores

Resumo

Nessa aula, aprendemos que, em algumas situações, precisamos incluir em um projeto alguns itens a mais do que o mínimo necessário para o funcionamento do sistema, segundo as especificações iniciais para facilitar a sua utilização pelo usuário.

Especificamente, como exemplos, o projeto original de controle de nível da caixa d'água foi alterado para incluir a sinalização do seu funcionamento através de LEDs e posteriormente com a utilização de módulo LCD para a exibição de mensagens de texto.

Assim, ao terminar outra lição, podemos perceber um pouco da infinidade de opções que temos para fazer a automação de um sistema. Continuemos então perseverantes para a próxima lição.

Atividades de aprendizagem

1. Explique, com suas palavras, porque em determinadas situações é recomendável incluir itens num projeto, além do mínimo necessário para um sistema funcionar de acordo com as especificações iniciais.



2. Pesquise e indique outros elementos visuais que podem ser acrescentados em um projeto, além de LEDs e módulos LCD de texto, para facilitar a utilização do sistema pelo usuário.
3. Altere os projetos elaborados para atender as três situações propostas no item 2, das Atividades de aprendizagem da Aula 1, de tal forma a incluir um módulo LCD em cada uma delas.

Aula 6 – Projetos com CLPs

Objetivos

Compreender as diferenças na metodologia de automação proposta entre a utilização de microcontroladores e CLPs como elemento principal de comando.

Desenvolver diagramas elétricos para sistemas de automação onde o elemento principal de comando seja um CLP.

Desenvolver programas em Ladder para sistemas automatizados de pequeno porte.

Elaborar listas de material para sistemas automatizados de pequeno porte baseados em CLPs.

6.1 Automação com CLPs

Na Aula 1, propomos uma metodologia para a automação de pequenos sistemas e a usamos para automatizar uma situação onde desejávamos controlar o nível de uma caixa d'água, que era enchida por uma bomba que captava a água de um rio.

Ao longo das Aulas 2, 3 e 4, os passos deste modo de agir foram explicados em detalhes, considerando que o sistema seria comandado por um microcontrolador.

Nessa aula, veremos as diferenças entre esta situação e uma outra, onde optamos em usar como elemento de comando um micro CLP (Controlador Lógico Programável) ao invés de um microcontrolador.

6.2 Descrição da solução proposta

A descrição da solução proposta expressa basicamente a forma como o sistema irá operar após a sua automatização e não a tecnologia envolvida para a obtenção deste resultado. Em outras palavras, este passo da metodologia é mais orientado para “o que deve ser feito” do que “o como será feito”.

Portanto, a indicação de qual elemento de controle será usado (microcontrolador ou CLP) é desnecessária neste momento. Sendo assim, este passo é o mesmo para as duas opções propostas.

6.3 Diagrama elétrico

O processo de desenvolvimento do diagrama elétrico de um sistema de automação baseado em um micro CLP começa, também, com o levantamento da quantidade de entradas e saídas que serão necessárias no projeto.

Partindo do nosso exemplo, do controle de nível da caixa d'água, vemos que serão necessárias quatro entradas digitais e duas saídas digitais, conforme mostrado anteriormente nos Quadros 2.1 e 2.2.

Dessa forma, como o total de pontos digitais é seis, optamos por um micro CLP da família Easy, da Moeller: o EASY512-DA-RC. Este dispositivo apresenta oito entradas digitais, quatro saídas digitais e é alimentado com 12 Vcc. A Figura 6.1 mostra a sua aparência física.



Figura 6.1: Micro CLP

Fonte: Moeller, 2014

Com o micro CLP escolhido, podemos estabelecer como os elementos de entrada e saída serão conectados a ele. Isso normalmente é representado através de um mapa de E/S (entradas e saídas), que simboliza estas ligações. O mapa para o nosso exemplo é mostrado no Quadro 6.1.

Os componentes de entrada (sensores) são exatamente os mesmos da situação onde o elemento de comando era o microcontrolador, ou seja, as chaves de nível do tipo boia (sensor de nível baixo, sensor de nível alto e sensor de falha) e

o microinterruptor (botão de desarme). O mesmo ocorre com os componentes de saída (atuadores): os relés de estado sólido para acionamento das cargas (bomba e sirene de alarme).

Quadro 6.1: Mapa de E/S – entradas e saídas

Referência	Descrição
I01	Sensor de falha – transbordamento (F)
I02	Sensor de nível baixo da caixa d'água (L)
I03	Sensor de nível alto da caixa d'água (H)
I04	Botão de desarme do alarme (D)
Q01	Sirene de alarme (A)
Q02	Motor da bomba da caixa d'água (B)

Fonte: Autores

A interconexão entre todos estes componentes do sistema é ditada pelo micro CLP escolhido, sendo definida após a análise do seu *datasheet*. Sendo assim, o diagrama elétrico para o nosso exemplo que leva em conta esta observação importante é mostrado na Figura 6.2.

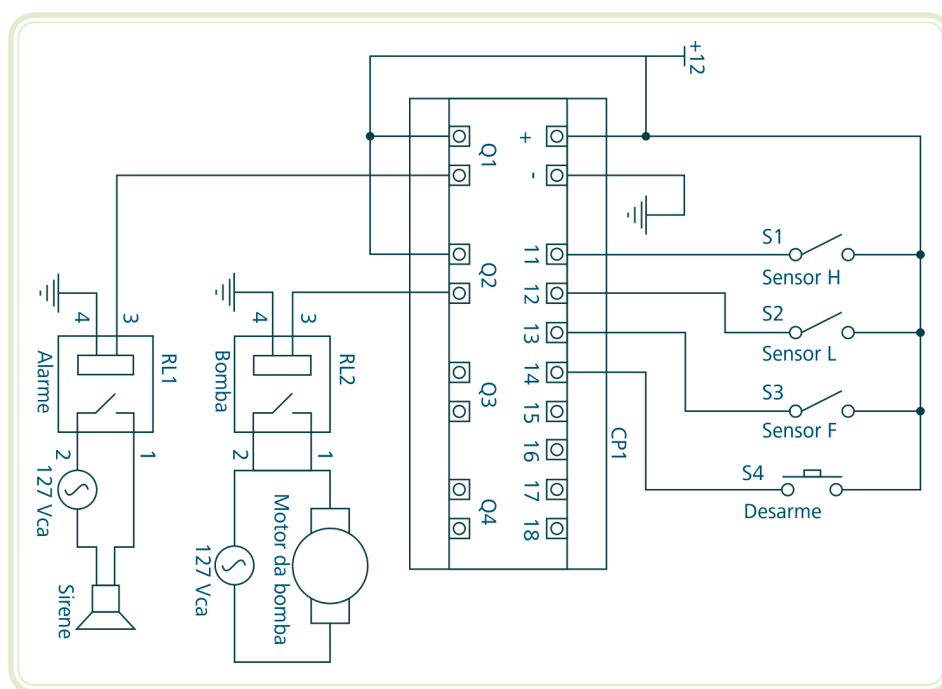


Figura 6.2: Diagrama elétrico com o micro CLP

Fonte: CTISM, adaptado dos autores

Note que desta vez optamos por utilizar uma fonte chaveada comercial (mostrada na Figura 6.3) para alimentar todo o circuito, já que o consumo de energia elétrica do micro CLP é maior do que o do microcontrolador.



Figura 6.3: Fonte de alimentação chaveada comercial

Fonte: CTISM

6.4 Programação do elemento de comando

Já que agora o nosso elemento de comando é micro CLP, faremos a sua programação em Ladder, usando o mesmo algoritmo apresentado no item 3.2 da Aula 3, como a base para a orientação do funcionamento do sistema.

O programa desenvolvido usando também as informações do mapa de E/S do sistema (Quadro 6.1) é mostrado na Figura 6.4.

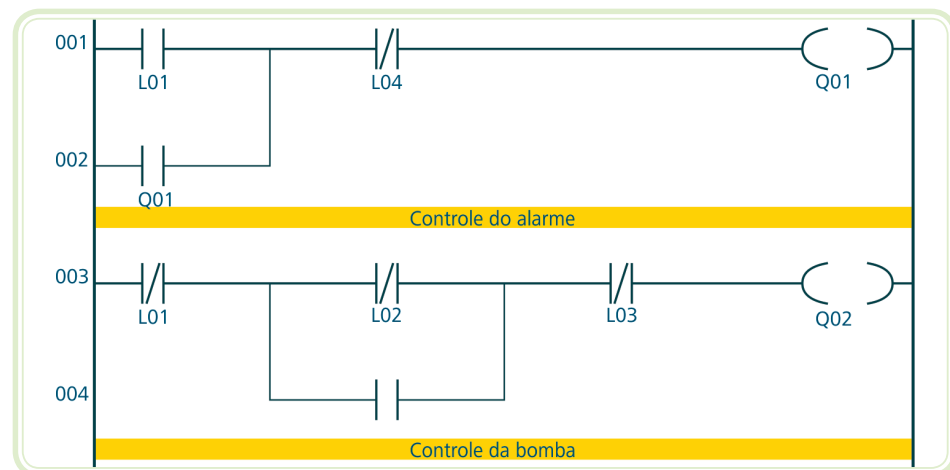


Figura 6.4: Programa em Ladder para o controle de nível da caixa d'água

Fonte: CTISM, adaptado dos autores

6.5 Lista de material

Todas as observações feitas para o projeto baseado em microcontrolador valem também para o projeto baseado no micro CLP. A nova lista de material com as alterações necessárias é mostrada no Quadro 6.2.

Quadro 6.2: Lista de material do projeto com micro CLP

Item	Descrição	Código	Fabricante
RL1	Relé de estado sólido	TSZC-25A	Metaltex
	Dissipador de calor	TSZC-DT2	Metaltex
RL2	Relé de estado sólido	TSZC-25A	Metaltex
	Dissipador de calor	TSZC-DT2	Metaltex
S1	Chave de nível tipo boia	CB2002	MarGirius
S2	Chave de nível tipo boia	CB2002	MarGirius
S3	Chave de nível tipo boia	CB2002	MarGirius
S4	Microinterruptor	18531 MFT3F ST S E	MarGirius
PS1	Fonte chaveada (12 Vcc)	CH 12 – 2	MCE
CP1	Micro CLP	EASY512-DA-RC	Moeller

Fonte: Autores

Observe que neste caso (uso do micro CLP como elemento de comando) não será necessário o emprego de uma placa de circuito impresso (PCI) para a montagem dos componentes já que eles serão interligados entre si através de fios conforme o diagrama elétrico da Figura 6.2.

Resumo

Nessa aula, aprendemos como usar a metodologia de automatização, proposta quando o elemento de comando escolhido for um micro CLP, fazendo as alterações necessárias no diagrama elétrico, na programação e na lista de material de todo o sistema.

Ao concluir mais esta lição, nos aproximamos ainda mais do final da disciplina. Sendo assim, você está próximo da conclusão do curso. Parabéns e continue firme até o final!

Atividades de aprendizagem



1. Explique com suas palavras quais são as diferenças entre a automação de sistemas, usando os microcontroladores ou os micro CLPs, enfatizando as vantagens e as desvantagens de cada método.
2. Considerando que o elemento de comando do sistema é um micro CLP, realize as seguintes tarefas para cada uma das três situações propostas no item 2, das Atividades de aprendizagem da Aula 1:
 - a) Desenvolvimento do diagrama elétrico.

- b)** Desenvolvimento do algoritmo de controle.
- c)** Conversão dos algoritmos do item anterior para a linguagem Ladder.
- d)** Elaboração da lista de material, citando sempre que é possível, ao menos, o código de dois fabricantes para cada item.

Aula 7 – Projetos finais

Objetivos

Revisar muitos conceitos vistos ao longo de várias disciplinas que compõem o Curso de Técnico em Automação Industrial.

Aplicar os conhecimentos adquiridos em uma possível situação onde haja a necessidade de automação de um sistema.

Compreender como o processo de automação de um sistema se desenvolve na prática.

7.1 Sistemas automatizados

A área da automação de sistemas é bem vasta e pode oferecer vários tipos de soluções para as mais diversas situações. Cabe, então, ao Técnico em Automação Industrial escolher e aplicar qual destas respostas possíveis é a mais adequada para um determinado problema.

Nessa aula, você terá a oportunidade de compreender melhor este mecanismo ao propor soluções completas para sistemas a serem automatizados. Sendo assim, tente empregar ao máximo os conhecimentos adquiridos nesta e em outras disciplinas para que obtenhamos os resultados mais proveitosos possíveis.

7.2 Automação de um sistema de esteiras

Você deverá desenvolver um projeto de automação completo para um sistema selecionador de peças baseado em três esteiras (uma principal e duas auxiliares), uma balança e dois cilindros pneumáticos conforme representado no desenho da Figura 7.1.

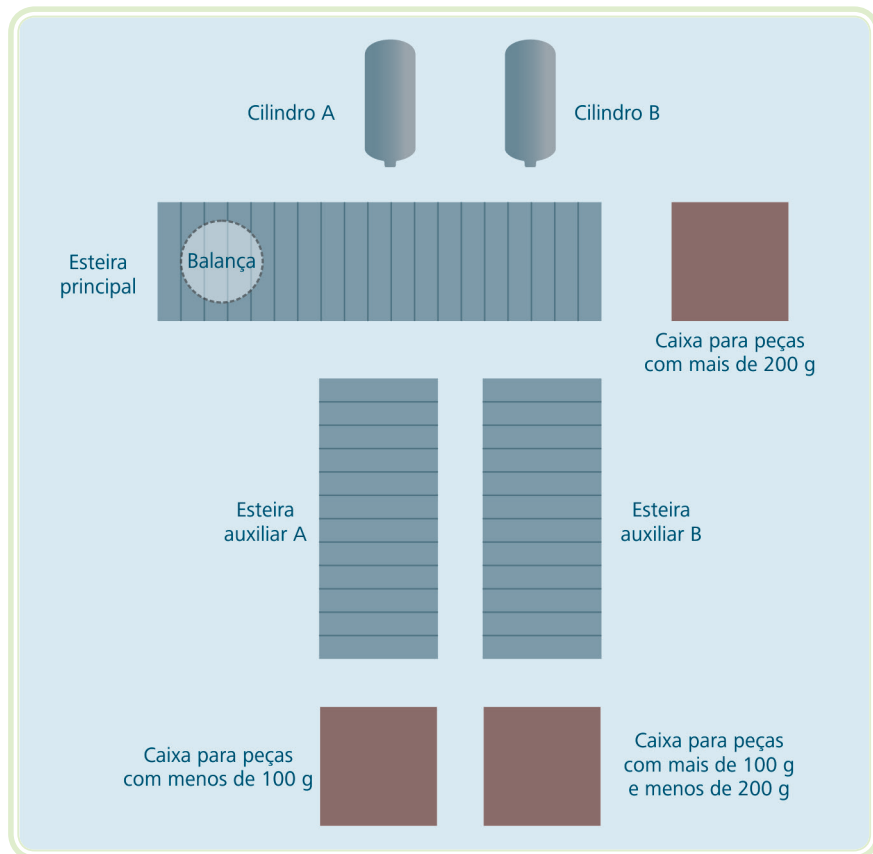


Figura 7.1: Sistema de esteiras para ser automatizado

Fonte: CTISM, adaptado dos autores

Os motores das esteiras são monofásicos (127 Vca/2 cv de potência). Em relação a todos outros componentes, eles devem ser escolhidos e especificados por você com o auxílio do professor.

Devido às características deste projeto final, recomendamos que esta tarefa seja realizada em grupo para que haja um maior aproveitamento sinérgico das capacidades e aptidões individuais.

No início da esteira principal, há uma balança que irá pesar as peças colocadas pelos operadores. A partir desta informação, o sistema irá decidir qual o procedimento a ser executado automaticamente.

a) Procedimento para peças com 100 g ou menos:

I – Parar a **esteira principal** quando a peça estiver em frente ao **cilindro pneumático A**.

II – Ligar a **esteira auxiliar A**.

III – Avançar o **cilindro pneumático A** por dois segundos (a peça consequentemente cairá na caixa).

IV – Recuar o **cilindro pneumático A**.

V – Desligar a **esteira auxiliar A**.

VI – Religar a **esteira principal**.

b) Procedimento para peças com mais de 100 g e menos de 200 g:

I – Parar a **esteira principal** quando a peça estiver em frente ao **cilindro pneumático B**.

II – Ligar a **esteira auxiliar B**.

III – Avançar o **cilindro pneumático B** por três segundos (a peça consequentemente cairá na caixa).

IV – Recuar o **cilindro pneumático B**.

V – Desligar a **esteira auxiliar B**.

VI – Religar a **esteira principal**.

c) Procedimento para peças com mais de 200 g:

I – Manter a **esteira principal** para que a peça caia na caixa no final da linha.

Poderão ser propostas melhorias extras em relação a esta especificação mínima, de tal forma a maximizar a aprendizagem durante a realização do projeto.

7.3 Elevador com três paradas

Desenvolva um projeto de automação completo para um elevador de carga com três paradas, cuja cabine será movimentada por um motor trifásico de 5 cv de potência. Assim, como no caso anterior, a sugestão é que este projeto seja feito em grupo com o auxílio e a orientação do professor.

A descrição básica do seu funcionamento é a seguinte:

- Cada andar possui um botão de chamada da cabine; dois botões para enviar a cabine para um dos outros dois andares; uma chave de bloqueio de movimentação da cabine e um sensor para indicar que a cabine está parada naquele andar.
- Não há botões no interior da cabine.
- Quando a cabine estiver parada em um andar, a porta será aberta automaticamente; quando ela estiver movimentado, tanto subindo quanto descendo, a porta da cabine deverá estar fechada.
- A cabine será movimentada pelo motor trifásico com um acionamento que permitirá que ele gire tanto para um lado (subida da cabine) quanto para o outro (descida da cabine).
- Um usuário chama a cabine para o seu andar apertando o respectivo botão. Se ela já estiver no andar onde o botão de chamada foi pressionado, nada acontecerá.
- Quando a cabine chegar ao seu andar, o usuário acionará uma chave de bloqueio de movimentação que impedirá que a cabine seja chamada para o outro andar, permitindo assim que o usuário carregue-a em segurança.
- Se a carga colocada na cabine exceder o peso máximo de 200 kg (há uma balança no seu piso para medi-la), uma lâmpada vermelha será acesa indicando esta condição e o motor não poderá ser acionado nem para subir e nem para descer.
- Ao terminar o processo de carga da cabine, o operador desligará a chave de bloqueio do seu andar podendo ele mesmo enviar a cabine para um dos outros dois andares através dos botões que estão no seu andar. Se ele não enviar a cabine, ela poderá posteriormente ser chamada por um dos outros dois andares.

Resumo

Essa foi a última aula da disciplina de Projetos de Automação. Nela tivemos a oportunidade de aplicar os conhecimentos adquiridos aqui e em outros momentos do curso, ao realizar um projeto de automação mais consistente.

Ao concluir esta lição e encerrar mais uma disciplina, você está muito próximo da conclusão da jornada para se tornar um Técnico em Automação Industrial. Congratulações!

Atividades de aprendizagem



1. Repita a execução total dos projetos propostos nos itens 7.2 e 7.3 dessa aula, substituindo o elemento principal de comando pelo outro que não foi utilizado inicialmente, ou seja, se você usou um microcontrolador em primeiro lugar, utilize agora um micro CLP ou vice-versa.
2. Desenvolva os projetos propostos nos itens 7.2 e 7.3 dessa aula de modo a incluir um módulo LCD em cada um deles.
3. Pesquise na internet sobre outras soluções possíveis para o elemento principal de comando, além dos microcontroladores e dos CLPs, para este caso e refaça o projeto para aquelas que forem encontradas.

Referências

CLARYDON. **Site institucional**. Disponível em: <<http://www.clarydon.com/wp-content/uploads/2010/12/circuit-board.jpg>>. Acesso em: 20 jan. 2014.

ELETRÔNICA DIGITAL. **Site institucional**. Disponível em: <<http://www.eletronicadigital.com/site/user/images/pcb-experimental/pcb-universal-desligada-verso.jpg>>. Acesso em: 20 jan. 2014.

FAIRCHILD. **Manual do regulador de tensão LM7805**. Disponível em: <<http://www.fairchildsemi.com/ds/LM/LM7805.pdf>>. Acesso em: 20 jan. 2014.

FERREIRA, Aurélio Buarque de Holanda. **Aurélio Século XXI: o dicionário da língua portuguesa**. Rio de Janeiro: Nova Fronteira, 1999.

MARGIRIUS. **Manual do sensor de nível tipo boia CB2000**. Disponível em: <<http://www.margirius.com.br/p/83.pdf>>. Acesso em: 20 jan. 2014a.

MARGIRIUS. **Manual da série 18.000** (microinterruptores). Disponível em: <<http://www.margirius.com.br/p/32.pdf>>. Acesso em: 20 jan. 2014b.

MCE. **Manual da linha de fontes chaveadas CH**. Disponível em: <<http://www.mctecnica.com.br/pdfs/ch.pdf>>. Acesso em: 20 jan. 2014.

METALTEX. **Manual da série de porta-fusíveis**. Disponível em: <http://www.metaltex.com.br/downloads/Porta_Fusivel.pdf>. Acesso em: 20 jan. 2014a.

METALTEX. **Manual da série TSZC de relés de estado sólido**. Disponível em: <<http://www.metaltex.com.br/downloads/TSZC.pdf>>. Acesso em: 20 jan. 2014b.

MICROCHIP. **Manual do microcontrolador PIC12F508**. Disponível em: <<http://www1.microchip.com/downloads/en/DeviceDoc/41236E.pdf>>. Acesso em: 20 jan. 2014.

MOELLER. **Manual do micro CLP EASY512-DA-RC**. Disponível em: <http://datasheet.moeller.net/Eaton-274106-EASY512-DA-RC-en_GB.pdf?model=274106&locale=en_GB&type=pdf>. Acesso em: 20 jan. 2014.

Currículo do professor-autor

Édilus de Carvalho Castro Penido, nasceu em Belo Horizonte – MG, iniciou a sua formação técnica com o curso de Eletrônica no Colégio Técnico do Centro Pedagógico (COLTEC) da Universidade Federal de Minas Gerais (UFMG) em 1989. Graduou-se em Engenharia Elétrica pela Pontifícia Universidade Católica de Minas Gerais (PUC-MG) em 1996. Concluiu em 2008 o Curso de Especialista em Projeto de Circuitos Integrados pela da Universidade Federal de Minas Gerais (UFMG). Finalizou em 2013 o Curso de Mestrado em Sustentabilidade Socioeconômica e Ambiental pela Universidade Federal de Ouro Preto (UFOP). Desde 1997, desenvolve suas atividades profissionais como docente em várias instituições de formação técnica e superior, com atuação constante na capacitação e qualificação de profissionais nas áreas de eletrotécnica, eletrônica e automação. A partir de 2008, passou a atuar como professor do Curso Técnico em Automação Industrial do Campus Ouro Preto do Instituto Federal de Minas Gerais (IFMG).



Pedro Luis Almeida de Oliveira Costa, meu primeiro contato com um computador se deu entre as meus dezesseis e dezessete anos de idade. Era um computador IBM, modelo 360, um dos primeiros modelos a chegar ao Brasil. Dispunha de 64 Mbytes de memória RAM (a mesma capacidade de memória de uma calculadora dessas que compramos nas ruas da cidade). Sou de uma geração que se formou academicamente junto ao mercado de trabalho. Poucos eram os cursos nessa área. Evidente, não havia educação a distância. Passei pelas faculdades de Análise de Sistemas (assim se chamava naquele tempo), Engenharia, História e concluí com Direito. Leciono as disciplinas de Linguagem de Programação, Banco de Dados e Ferramentas para Internet desde a criação do curso, no CEFET-Ouro Preto, hoje IFMG. Todo e qualquer área de atividade humana exige atualização constante dos profissionais, mas sempre tive a impressão de que, com a informática, essa exigência é ainda maior.



