**CENTRO PAULA SOUZA**

**FACULDADE DE TECNOLOGÍA DE TATUÍ**

**CURSO SUPERIOR DE TECNOLOGIA EM AUTOMAÇÃO INDUSTRIAL**

**ANDRÉ LUIS GUERRA DE LIMA JUNIOR**

**GUILHERME DE ALMEIDA GILLI**

**CONJUNTO DIDÁTICO PARA MICROCONTROLADOR COM ATUADORES**

**Tatuí, SP**

**2º Semestre/2015**

**ANDRÉ LUIS GUERRA DE LIMA JUNIOR**

**GUILHERME DE ALMEIDA GILLI**

**CONJUNTO DIDÁTICO PARA MICROCONTROLADOR COM ATUADORES**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Tatuí, como exigência parcial para obtenção do grau de Tecnólogo em Automação Industrial sob a orientação do Professor Otavio Gaizutis.

**Tatuí, SP**

**2º Semestre/2015**

**ANDRE LUIS GUERRA DE LIMA JUNIOR**

**GUILHERME DE ALMEIDA GILLI**

**CONJUNTO DIDÁTICO PARA MICROCONTROLADOR COM ATUADORES**

Trabalho de Graduação apresentado à banca examinadora da Faculdade de Tecnologia de Tatuí, para a obtenção do grau de tecnólogo em Automação Industrial.

( ) APROVADO ( ) REPROVADO

Com média:................

Tatuí......de.......de 2015

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Dr.

FATEC – Tatuí

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Dr.

FATEC – Tatuí

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Dr.

FATEC – Tatuí

**AGRADECIMENTOS**

Agradecemos a Deus por nos fortalecer e mostrar o caminho, agradecemos à nossa família por sempre nos apoiar em nossos projetos e agradecemos aos nossos professores por nos orientar e transmitir o conhecimento necessário para alcançar a linha de chegada.

**RESUMO**

Este trabalho tem por objetivo facilitar a interação dos alunos com o microcontrolador, configurando assim a montagem de um kit didático com microcontrolador onde será possível obter um melhor entendimento das aulas. De forma que o aluno ira realizar toda a parte estrutural do programa e poderá realizar os testes na bancada, obtendo assim melhor visualização e entendimento do estudo. Com esse projeto reunimos informações de escolas técnicas e faculdades técnicas para saber em qual área se encontravam mais dificuldades no aprendizado, a fim de desenvolver um protótipo que acarreta no aumento da qualidade de aprendizado e na qualidade da aula que ganha mais dinâmica e se torna mais didática, uma vez que se utiliza de muitas ferramentas abordadas ao longo do curso de Tecnologia em Automação Industrial como, lógica de programação, elétrica, eletrônica, microcontroladores.

**PALAVRAS CHAVES:** Kit didático, Microcontrolador, Automação.

**ABSTRACT**

This work aims to facilitate student interaction with the microcontroller, thus creating the assembly of a didactic kit with microcontroller where it will be possible to gain a better understanding of the classes. So the student will perform all the structural part of the program and will be able perform the tests on the bench, thus obtaining better visualization and understanding of the study. With this project we gather information from technical schools and technical colleges to know in which area were found more Difficulties in learning, in order to develop a prototype that led to the increase of quality of learning and the quality of the class that gets more dynamic and becomes more didactic, since it uses many tools Discussed throughout the course of Technology in Industrial Automation, the programming logic, electrical, electronics, microcontrollers.

**KEYWORDS:** Didatic Kit, Microcontroller, Automation.

**LISTA DE ILUSTRAÇÕES**

[Figura 1 – Arquitetura de um Microcontrolador . 1](#_Toc435580536)4

Figura 2 – Diagrama de bloco PIC18F4550 15

Figura 3 – Registrador W 23

Figura 4 – Diagrama de Pinos PIC18F4550 encapsulamento PDIP 24

Figura 5 - Motor de passo híbrido 40  
Figura 6 – Diagrama de blocos simplificado de um LCD 43

Figura 7 – Forma de um LED 7 segmentos 44

Figura 8 – Regulador de Tensão 5V 46

Figura 9 – Microcontrolador PIC18F4550 47

Figura 10 – Entradas analógicas 48

Figura 11 – Cristal de frequência 49

Figura 12 – Conector USB 49

Figura 13 – Dipswitch DSW2 50

Figura 14 – Entradas digitais e botão de reset 51

Figura 15 – Saídas digitais, LEDs 52

Figura 16 – Displays 7 segmentos 53

Figura 17 – Tabela da verdade CI 4511 54

Figura 18 – Display de LCD 55

Figura 19 – Relê, Cooler e Dipswitch 55

Figura 20 – DSW1 e L293 57

Figura 21 – Tabela verdade para meio passo L297 58

Figura 22 – Tabela da verdade para passo normal 58

Figura 23 – Layout da placa 60

Figura 24 – Ferro de solda , solda “Estanho” e sugador de solda 60

Figura 25 – Pasta de solda e fluxo de solda 60

Figura 26 – Placa fenol 15x20cm 61

Figura 27 – Pasta de solda e Fluxo de solda 61

Figura 28 – Ci, semicondutores e display LCD 62

Figura 29 – Motor de passo, relê. LED’s e display 7 segmentos 62

Figura 30 – Dipswithces, conectores e botões 63

Figura 31 – Resistores, capacitores, cabo usb e cooler 64

**LISTA DE QUADROS**

Quadro 1 – Mapa dos Bancos de memória PIC18F4550 20

Quadro 2 – Mapa dos Registradores de função Especial 21

Quadro 3 – Descrição do pino Reset e Osciladores 24

Quadro 4 – Descrição do pino PORTA 25

Quadro 5 – Descrição do pino PORTB 27

Quadro 6 – Descrição dos pinos PORTC 29

Quadro 7 – Descrição dos pinos PORTD 31

Quadro 8 – Descrição dos pinos PORTE 32

Quadro 9 – Diferentes modulações de PWM 37

Quadro 10 – Módulos LCD 41

Quadro 11 – Ambiente e desempenho de LCD 42

Quadro 12 – Diagrama de Blocos 45

Quadro 13 – Orçamento dos componentes 65

**LISTA DE SIGLAS DE ABREVIATURAS**

CCP - Capture, Cimpare e PWM

CISC - Complex Instruction Set Computer

DIP - Dual In-Line Package

DRAM - Dynamic RAM.

EEPROM - Electrically-Erasable Programmable Read Only-Memory

I/O - Input/Output (Entrada/saída)

LCD - Liquid Cristal Display

LED - Light-Emitting Diode

PLCC - Power Line Carrier Communication

RAM - Random-Access Memory

RISC - Reduced Instruction Set Computer

ROM - Read Only Memory

SOIC - Small Outline Integrated Circuit

SPI - Serial Peripheral Interface

SPP - Streaming Parallel Port

SRAM - Static RAM

TQFP - Thin Quad Flat Pack

UART - Universal Asynchronous Receiver Trasmiter

ULA - Arithmetic Logic Unit

USART - Universal Synchronous Asynchronous Receiver Transmitter

**SUMÁRIO**

[1 INTRODUÇÃO 10](#_Toc437537580)

[2 REFERENCIAL TEÓRICO 11](#_Toc437537581)

[2.1 MICROCONTROLADOR 11](#_Toc437537582)

[**2.1.1 História dos Microcontroladores** 11](#_Toc437537583)

[**2.1.2 Arquitetura de um Microcontrolador** 13](#_Toc437537584)

[**2.1.3 Unidade Central de Processamento** 16](#_Toc437537585)

[**2.1.4 Memórias de um Microcontrolador** 16](#_Toc437537586)

[2.1.4.1 Memória de Programa 17](#_Toc437537587)

[2.1.4.2 Memória de Dados e Registradores 18](#_Toc437537588)

[2.1.4.3 Registradores 20](#_Toc437537589)

[**2.1.5 Terminais de Entrada e/ou Saída** 23](#_Toc437537590)

[**2.1.6 Interrupções** 33](#_Toc437537591)

[**2.1.7 Modulação por largura de pulso** 36](#_Toc437537592)

[2.2 LINGUAGEM DE PROGRAMAÇÃO C 38](#_Toc437537593)

[2.3 MOTOR DE PASSO 39](#_Toc437537594)

[**2.3.1 Motor Hibrído – (HB Hybrid)** 39](#_Toc437537595)

[**2.3.2 Passo Normal** 40](#_Toc437537596)

[**2.3.3 Meio Passo** 40](#_Toc437537597)

[2.4 DISPLAYS 40](#_Toc437537598)

[2.5 LED 43](#_Toc437537599)

[3 MATERIAIS E MÉTODOS 45](#_Toc437537600)

[3.1 DESENVOLVENDO O CONJUNTO 59](#_Toc437537601)

[4 RESULTADOS E DISCUSSÕES 67](#_Toc437537602)

[5 CONCLUSÃO 68](#_Toc437537603)

[REFERÊNCIAS 69](#_Toc437537604)

1. INTRODUÇÃO

Em 1971, a INTEL lança ao mercado o primeiro microprocessador designado “4004”. Este foi o primeiro microprocessador de 4 bits e era aplicado para realizar operações matemáticas das calculadoras BUSICOM. Logo no ano seguinte, a Intel desenvolve os microprocessadores de 8 bits à pedido da companhia americana CTC. Estes eram denominados “8008”. Esse microprocessador foi o pioneiro para chegarmos até os que encontramos no mercado hoje em dia. Desde então os microprocessadores tiveram papel fundamental na evolução de computadores e de dispositivos com unidades de processamento, como os microcontroladores.

Com a evolução dos microcontroladores, estes foram se destacando em projetos na automação juntamente com a eletrônica. Sendo assim, é de fundamental questão que o profissional que trabalhe na área saiba como utilizar esta ferramenta que é o microcontrolador. A facilidadede manuseio da ferramenta, fara com que os profissionais programe contadores simples com displays até projetos mais complexos envolvendo comunicação serial, lógica e controle de variáveis.

Apesar dos computadores já conterem simuladores e compiladores próprios para a didática aplicada a microcontroladores, um layout onde se possa interagir com equipamentos e dispositivos reais pode facilitar o aprendizado de alunos e interessado a ampliar seus conhecimentos em microcontroladores.

Pensando nesta questão e na própria experiência do grupo durante a formação acadêmica, tornou se necessária então a elaboração de uma placa eletrônica didática onde é possível visualizar e se familiarizar com aplicações voltadas à automação.

Esta placa terá como elemento principal um microcontrolador da família PIC, modelo 18F4550 e através de suas entradas e saídas poderá se comunicar e controlar equipamentos como displays 7 segmentos, displays LCD, led’s, motor de passo, etc.

O trabalho está organizado de forma que no capítulo 2 são definidos elementos e termos que fazem parte do projeto. No capitulo 3 são apresentados os materiais e as especificações do projeto. No capitulo 4 serão apresentados os resultados que foram obtidos, e finalmente, no capitulo 5 capítulo, a conclusão.

1. REFERENCIAL TEÓRICO

## 2.1 MICROCONTROLADOR

Microcontroladores, são pastilhas inteligentes que possuem processador, memória e interface com o meio externo através de terminais de entrada e saída. Estes podem ser programados para que suas saídas sejam alteradas de acordo com o valor de suas entradas (MARTINS, 2005).

De outra maneira, são pequenos componentes eletrônicos dotados de inteligência programável, utilizado no controle de processos lógicos. O controle de processo deve ser compreendido como o controle de periféricos. Os microcontroladores são programáveis, ou seja, é possível alterar a sua lógica (suas ações) de acordo com um processo chamado gravação, do qual é responsável que todas as vezes que o microcontrolador for alimentado, o programa (a sua lógica) será executado (SOUZA, 2010).

Vale ressaltar que diferentemente dos microprocessadores, os microcontroladores não necessitam de memórias externas ou componentes de envio e recebimento de dados para o seu funcionamento uma vez que todos os periféricos estão contidos nele (exemplo: memórias, processador, temporizadores)(MATIC,2005). Ou seja, em uma única pastilha de silício tem-se todos os componentes necessários para o seu funcionamento. Por exemplo: memória de programa, memória de dados, portas de entrada e saída, timers, contadores, conversores analógico-digitais, etc (SOUZA,2010).

### 2.1.1 História dos Microcontroladores

Com a proposta de trazer um circuito integrado cuja função seria determinada por um programa armazenado no mesmo, estudos iniciaram em 1969 com engenheiros japoneses – integrando a tecnologia à calculadoras - passaram pela BUSICOM até chegar a INTEL onde o primeiro microprocessador nasceu.

Em 1971 aparece ao mercado o primeiro microprocessador de 4 bits designado por 4004. No ano seguinte, após pedidos da companhia Americana CTC, a INTEL lança ao mercado microprocessadores de 8 bits, estes com o nome de 8008 que já possuía velocidade 50 vezes superior aos do modelo antigo (300.000 operações por segundo contra 6000). A partir daí, houve uma grande agitação no mercado, onde ocorreram disputas entre a *MOS Technology* e a INTEL com os seus microcontroladores em relação a tecnologia e preço. A *MOS Technology* chegou a vender 15 milhões de microprocessadores por ano e teve o seu modelo 6502 utilizados em computadores como Apple I, Apple II e Atari (MATIC; ANDRIC, 2005).

O histórico dos microprocessadores tem reflexo direto nos microcontroladores, pois a partir deste, existiu a possibilidade de melhoria no processamento de dados e redução dos periféricos a um único componente, o microcontrolador. Estes, depois de avanços da tecnologia, boas relações custo/benefício a projetos que demandam processamento, baixo custo e pequena necessidade de espaço físico. Hoje, os grandes nomes no mercado são: INTEL (8051), ATMEL, DALLAS e MICROCHIP (família PIC) (MARTINS, 2005).

Neste trabalho, o microcontrolador utilizado será o PIC18F4550 da Microchip, as principais características deste modelo são:

1. 36 pinos de I/O;
2. Frequência de até 48MHz;
3. Oscilador interno de 8Mhz/31KHz;
4. 32kbytes de memória FLASH de programa;
5. 2048 bytes de memória RAM para dados;
6. 256 bytes de memória EEPROM;
7. Módulo TIMER 0 (*timer*/contador de 8 ou 16 bits);
8. Módulo TIMER 1 (*timer*/contador de 16 bits);
9. Módulo TIMER 2 (*timer*/contador de 8 bits);
10. Módulo TIMER 3 (*timer*/contador de 16 bits);
11. Dois módulos CCP (modos Capture, Compare e PWM);
12. Dois comparadores analógicos com referência interna programável de tensão;
13. Comunicação serial: MSSP, USART;
14. Um *watchdog* *timer*;
15. Comunicação USB V2.0;
16. Capacidade de corrente de 25mA por pino de I/O;
17. 75 instruções – podendo extender para 83;
18. Vinte fontes de interrupção independentes;
19. Modo de funcionamento *sleep*;
20. Tensão de operação entre 2,0V a 5,5V.

(ZANCO, 2008; MICROCHIP, 2009)

Características elétricas:

1. Temperatura de trabalho iiiiiiiiii.....................................................-40°C a +85°C
2. Temperatura de armazenamentoiiiiiii.........................................-65°C a +150°C
3. Tensão de trabalhoi..........................................................................4,2V a 5,5V
4. Tensão máxima no pino VDD (em relação ao VSS).................. -0,3V até 7,5 V
5. Tensão máxima no pino MCRL (em relação ao VSS)................. 0V até 13,25V
6. Tensão máxima nos demais pinos (em relação ao VSS)...............................................................................-0,3V até (VDD + 0,3V)
7. Dissipação máxima de energia................................................................. 1,0 W
8. Corrente máxima de saída no pino VSS................................................. 300mA
9. Corrente máxima de entrada no pino VDD............................................. 250mA
10. Corrente máxima de entrada de um pino (quando em VSS).....................25mA
11. Corrente máxima de saída de um pino (quando em VDD) .......................25mA
12. Corrente máxima de entrada de todos os ports..................................... .200mA
13. Corrente máxima de saída de todos os ports......................................... 200mA

(MICROCHIP, 2009; SOUSA, SOUZA, LAVINIA, 2010)

Nos capítulos a seguir, estas características serão detalhadas.

## **2.1.2 Arquitetura de um Microcontrolador**

Arquitetura de um microcontrolador é a forma com que este é estruturado. Os microcontroladores da Família PIC possuem a arquitetura do tipo Harvard (sendo uma outra arquitetura também conhecida a Von-Neumann). A primeira citada desenvolveu-se da necessidade de aumentar a velocidade de trabalho de um microcontrolador, pois conforme a figura 1, apresenta separada cada memória de dados e da memória de programa, sendo assim, possível uma maior fluência de dados através da unidade de processamento (SOUZA,2010; MATIC, ANDRIC,2005).

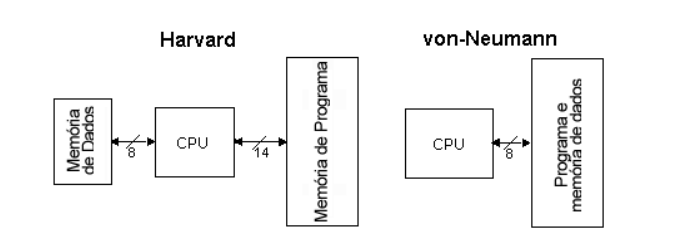


Figura 1 – Arquitetura de um Microcontrolador (MATIC, ANDRIC, 2005)

Enquanto a arquitetura tipo Von-Neumann possui apenas um barramento interno (bus), geralmente de 8 bits, para trafico das instruções e dados na arquitetura Harvard, existem dois barramentos internos, sendo um de dados e outro de instruções. Nos microcontroladores PIC, o barramento de dados sempre é de 8bits e o de instruções pode ser de 12, 14 ou 16 bits. Este modelo de arquitetura permite que, enquanto uma instrução é executada, outra seja resgatada da memória, acelerando assim o processo (SOUZA,2010; MATIC, ANDRIC,2005).

A arquitetura Harvard também tem como característica um escopo de instruções reduzidos devido a tecnologia RISC (*Reduced* *Instruction* *Set* *Computer*), se comparado a arquitetura Von-Neumann – que pertence a tecnologia CISC (*Complex* *Instruction* *Set* *Computer*). Desta maneira, os microcontroladores que utilizam a tecnologia RISC acabam tornando o aprendizado de programação “dinâmico” pois cabe ao programador construir funções que não possuem em instruções diretas (SOUZA,2010; MATIC, ANDRIC,2005).

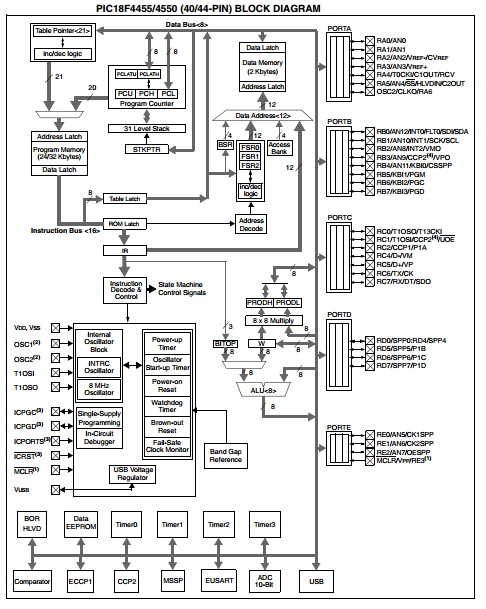


Figura 2 – Diagrama de bloco PIC18F4550 (MICROCHIP, 2009).

Na figura 2 pode ser visto como é feita a comunicação interna de um microcontrolador PIC18F4550 da Microchip, que será o modelo utilizado nesse trabalho. Alguns destes itens serão abordados nos próximos capítulos.

## **2.1.3 Unidade Central de Processamento**

A Unidade Central de Processamento através da comunicação com a memória e os dispositivos de I/O é responsável em gerenciar todo o sistema e executar os programas. Pode-se dividir a CPU em três módulos: Unidade Lógica e Aritmética, Unidade de Instrução e Controle e Rede de Registradores (ZANCO, 2005).

A ULA é a parte responsável pelos cálculos matemáticos e lógicos. Exemplos: Soma, subtração, lógica OU, E, XOU, etc (ZANCO, 2005).

A Unidade de Instrução e Controle é responsável por gerar os sinais necessários para que as instruções possam ser executadas pela CPU (ZANCO, 2005).

A Rede de Registradores é subdividida em duas: Registradores de uso geral e com finalidade específica. O primeiro citado armazenam dados temporários que serão manipulados pelo programa em algum momento, o segundo é utilizado em tarefas predeterminadas (ZANCO, 2005).

Junto à CPU existem as Memórias e os Dispositivos de I/O que serão citados nos próximos capítulos.

## **2.1.4 Memórias de um Microcontrolador**

A memória de um microcontrolador é a parte designada a reter, recuperar, armazenar e evocar informações codificadas digitalmente. Estas informações podem representar números, letras, caracteres quaisquer, comandos de operações e endereços. Ela pode ser comparada com uma grande prateleira cheia de gavetas. Considerando que estas sejam identificadas, o conteúdo que possuem pode ser facilmente acessado, basta saber em qual gaveta está a informação que deseja (SILVA, 2006; CAPUANO, IDOETA, 2000).

Existem diferentes modelos de memórias, por isso, estas geralmente são classificadas de acordo com 4 itens: Acesso, Volatilidade, Troca de Dados e Tipo de Armazenamento.

1. Acesso: Pode-se ter acesso aos endereços de memória de forma sequencial e aleatório, sendo a que a primeira passa por todas as localidades intermediárias até chegar a desejada e a segunda chegam na localidade diretamente.
2. Volatilidade: Dividem-se entre as voláteis e as não voláteis. Como o próprio nome indica, as não voláteis ao se encerrar a alimentação elétrica não perde o seu conteúdo e as voláteis ao se cortar a alimentação perdem sua informação.
3. Troca de Dados: Neste quesito, existem as memórias que permitem leitura e escrita (leitura do dado e alteração da informação) e as que permitem apenas leitura.
4. Tipo de Armazenamento: Em relação ao armazenamento, existem memórias de armazenamento estático e dinâmico. O primeiro, uma vez armazenado uma informação, esta lá permanece. No segundo modelo é necessário inserir a informação com certa frequência, pois de acordo com os seus componentes, perdem esta informação.

No caso dos microcontroladores, usualmente a informação da memória é tratada de duas maneiras:

a) *Read* (Ler), o que significa que ele não irá sofrer alterações, apenas informará ao programa o seu conteúdo.

b) *Write* (Escrever), neste caso, o valor é alterado, o programa está escrevendo alguma informação que deve ser armazenada. (MATIC, ANDRIC,2005; SILVA,2006; IDOETA, CAPUANO,2000; ZANCO,2005).

O PIC18F4550 possui três memórias distintas: A memória FLASH PROM, a memória RAM, e a memória EEPROM. A primeira é responsável em armazenar o programa que será executado pelo microcontrolador. A segunda é utilizada para dados temporários e a terceira, utilizada para armazenamento de dados que não podem ser perdidos em casos de quedas de energia. Detalhes sobre essas tecnologias estão nos próximos sub-capítulos. (MATIC, ANDRIC,2005; SILVA,2006; IDOETA, CAPUANO,2000; ZANCO,2005).

### 2.1.4.1 Memória de Programa

São utilizadas para armazenar programas de sistemas operacionais em computadores e, em microcontroladores, com a tecnologia EEPROM (*Eltrically* *Erasable* *Programmable* *Read* *Only* *Memory* – em tradução livre: Memória Programável Apagável Somente de Leitura), são utilizadas pelos programadores para armazenar informações importantes e não perde-las em uma queda de energia ou falta de alimentação (MATIC, ANDRIC,2005; IDOETA,CAPUANO, 2000).

Este tipo de memória utiliza como base as memórias ROM (*Read* *Only* *Memory*, traduzindo: Memória de Apenas Leitura) que tem como principal característica o que seu nome sugere, é uma memória que permite apenas leitura de dados gravados na sua fabricação. Caracterizam-se também como memória de acesso aleatório e seus dados não são voláteis. (MATIC, ANDRIC,2005; IDOETA,CAPUANO, 2000).

Além da tecnologia ROM, as memórias de programa podem utilizar a tecnologia FLASH, que são componentes que permitem ser apagados e regravados milhares de vezes por meio de sinais elétricos aplicados em alguns de seus terminais (DA SILVA ZANCO,2005).

O microcontrolador PIC18F4550 utiliza como memória de programa a tecnologia FLASH, regravável eletronicamente com escrita rápida o que possibilita ser acessada via software, possibilitando assim que o programa seja reescrito dinamicamente ou que a memória FLASH seja usada como expansão da memória E²PROM de dados. (SOUSA, SOUZA, LAVINIA, 2010).

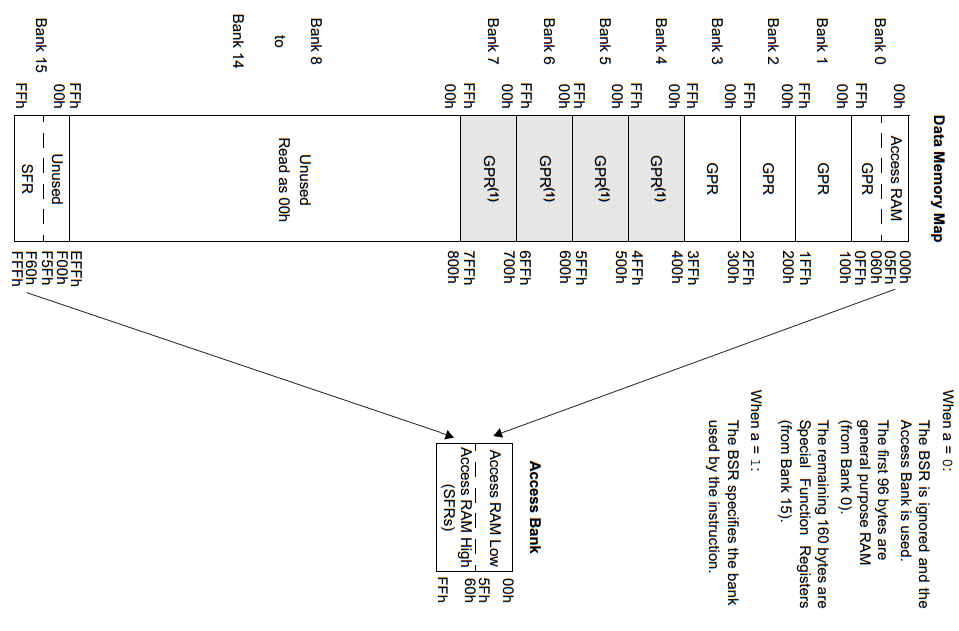
### 2.1.4.2 Memória de Dados e Registradores

A memória de dados de um microcontrolador é responsável por armazenar as variáveis e os registradores utilizados pelo programa. Este tipo de informação, devido a tecnologia RAM, é perdida automaticamente ao se desenergizar o microcontrolador. (SOUSA, SOUZA, LAVINIA, 2010).

Como dito anteriormente, esta memória utiliza tecnologia RAM (*Random*-*Access* *Memory*) que são do tipo voláteis e de acesso randômico. Tem como característica tempo de acesso reduzido, sendo assim utilizada principalmente como memória de programa e dados para armazenamento temporário de sistemas digitais. Este tipo de memória pode ser encontrado com armazenamento estático ( SRAM - *Static* RAM) e dinâmico (DRAM – *Dynamic* RAM). Sua utilização nos microcontroladores são para guardar dados intermediários ou temporários durante a execução do programa (MATIC,ANDRIC,2005;IDOETA,CAPUANO,2000).

Nos microcontroladores da Microchip, uma memória RAM é compartilhada pelos registradores com finalidades específicas e pelos registradores de propósito geral. Dos Registradores, temos os mais comuns: STATUS, INTCON, OPTION\_REG, PORT, TRIS, LAT. Os registradores serão assunto de um futuro capítulo. No caso dos registradores de propósito geral, também chamados de memória de dados, são utilizados para armazenamento temporário de informações. O PIC 16F628A tem 224 localidades de memória de dados com capacidade de armazenamento de 1 byte em cada localidade ( ZANCO,2005).

A memória RAM de dados de um microcontrolador da família PIC18 é dividida em 16 bancos – esses bancos são as divisões da memória de dados em grupos de 256 bytes - e é nessa memória que encontramos os registradores de propósito geral e pelos registradores com finalidades específicas. Conforme o quadro 1 nos mostra, o registrador responsável pela seleção direta dos bancos é o registrador BSR (*Bank* *Select* *Register*) de tal forma que os quatro bits menos significativos do registrador são concatenados com os 8 bits do endereço imediato. O banco 15 tem a particularidade de ser dedicado parcialmente aos registradores de função especial. (SOUSA, SOUZA, LAVINIA, 2010; MICROCHIP,2009; PEREIRA, 2010).

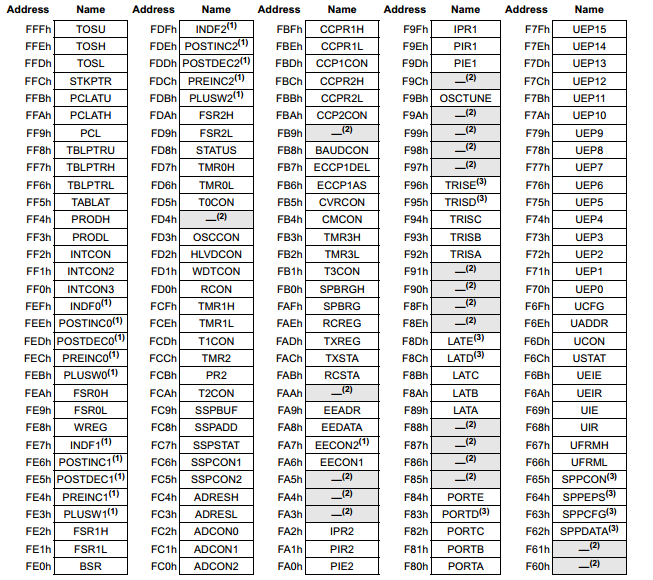


Quadro 1 – Mapa dos Bancos de memória PIC18F4550 (MICROCHIP, 2009).

### 2.1.4.3 Registradores

Nos microcontroladores da Microchip, a memória RAM pode ser divida em dois grupos: registradores de uso comum e registradores de funções especiais. O primeiro grupo trata-se de uma área destinada ao armazenamento de variáveis definidas pelo programador para serem escritas e lidas pelo programa. (SOUSA, SOUZA, LAVINIA, 2010; MICROCHIP, 2009)

Já os registradores de função especial no quadro 2, também denominados de SFRs (*Special* *Functions* *Registers*) são utilizados pelo microcontrolador para execução do programa e processamentos da unidade lógica aritmética - ULA. Esses registradores podem ser lidos e escritos pelo programa e pelo hardware e configuram funções e a utilização dos periféricos.



Quadro 2 – Mapa dos Registradores de função Especial (MICROCHIP, 2009).

Temos um terceiro grupo de registradores, os registradores de configuração, esses estão localizados na memória flash entre os endereços 0x300000 a 0x300000D. Esses registradores só podem ser alterados no instante em que se grava o chip. Alguns dos itens que podem ser configurados por esses registradores são:

1. Com o registrador CONFIG1H: modo de partida, monitoramento falha de clock, seleção do oscilador;
2. Com o registrador CONFIG2L: habilitação do reset por queda de tensão, habilitação do temporizador de partida;
3. Com o registrador CONFIG2H: seleção do pós-divisor da saída do watchdog, habilitação do watchdog;
4. Com o registrador CONFIG3H: seleção do pino MCLR/RE3, seleção do modo de baixa potência, configuração dos pinos da porta B, multiplexação do pino de entrada/saída do CCP2;
5. Com o registrador CONFIG4L: habilitação do depurador em background, habilitação do conjunto de instruções estendidas, programação utilizando baixa tensão, reset por estouro da pilha;
6. Com o registrador CONFIG5L: proteção contra leitura externa da área de memória dos blocos 0, 1, 2 e 3;
7. Registrador CONFIG5H: proteção contra leitura externa da EEPROM interna, proteção contra leitura externa da área de memória do bloco de boot;
8. Registrador CONFIG6L: proteção contra escrita dos blocos 0, 1, 2 e 3;
9. Registrador CONFIG6H: proteção contra escrita na EEPROM interna, proteção contra escrita no bloco de boot, proteção contra escrita nos registradores de configuração;
10. Registrador CONFIG7L: proteção contra leitura dos blocos 0, 1, 2 e 3 via instrução TBLRD localizadas em outros blocos.
11. Registrador CONFIG7H: proteção contra leitura do bloco de boot via instruções TBLRD localizadas em outros blocos (PEREIRA, 2010; MICROCHIP, 2009).

Outro registrador que merece destaque é o registrador W (*Work* *Register*) na figura 3. Todas as operações aritméticas ou booleanas são feitas por meio dele. Os resultados de suas operações podem ser destinadas a outros registradores ou ao próprio registrador W (MARTINS, 2005; PEREIRA,2010).

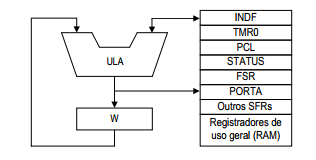


Figura 3 – Registrador W (MARTINS, 2005).

### 2.1.5 Terminais de Entrada e/ou Saída

Os terminais são as vias em que o microcontrolador é alimentado e por onde se comunicará com o meio externo. O PIC18F4550 possui 40 pinos agrupados em por PORTs para facilitar o gerenciamento da máquina. Temos então, três portas de E/S com 8 bits, uma porta com 7 bits e uma porta com 4 bits. São eles: PORTA (RA0:RA7), PORTB (RB7:RB0), PORTD (RD0:RD7) de oito bits; PORTC (RC0:RC7 excluindo o RC3 que não é implementado neste componente) com sete bits; PORTE (RE0:RE3) com 4bits. Como mostrado na figura 4, conforme os encapsulamentos PDIP, cada PORT está associado a um respectivo pino do microcontrolador e pode ser configurado como entrada ou saída independente do restante por um registro chamado TRIS - TRISA para o PORTA, TRISB para o PORTB, TRISC para o PORTC, TRISD para o PORTD e TRISE para o PORTE. O registro TRIS funciona de que, configurando um bit = 1 fara o pino correspondente PORT uma entrada (pino correspondente em modo de alta impedância), e, limpando esse valor (bit = 0) o correspondente pino será uma saída.

Além do registro PORT e TRIS, há o registro LAT que foi criado para resolver problemas encontrados em modelos anteriores em relação alterações feitas na porta de forma sequencial. A leitura do registrador LAT corresponde à leitura do último estado na porta antes de configurar como entrada. A escrita no registrador LAT corresponde à escrita na porta do microcontrolador. Têm-se como sugestão: utilizar o registrador PORT para leitura de uma porta e utilizar o registrador TRIS para operação de escrita em um port. (DA SILVA ZANCO, 2008; SOUSA, SOUZA, LAVINIA, 2010; MICROCHIP, 2009).

A figura 4 mostra a disposição dos pinos de um microcontrolador PIC18F4550 no encapsulamento PDIP.

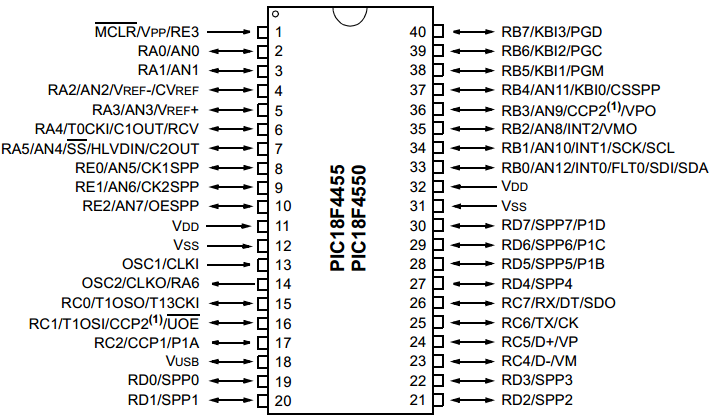


Figura 4 – Diagrama de Pinos PIC18F4550 encapsulamento PDIP (MICROCHIP, 2009).

Os quadros de 3 a 8 detalharão cada pino do PIC18F4550 conforme encapsulamento PDIP. O primeiro quadro detalha o pino de reset e Osciladores do microcontrolado:

|  |  |  |  |
| --- | --- | --- | --- |
| Nome do Pino | Número do Pino | Tipo do Pino | Descrição |
| /VPP/RE3    VPP  RE3 | 1 | E  A  E | *Master* *Clear* (entrada) ou tensão de programação (entrada).  *Master* *Clear* (Reset) entrada. Este pino é um Reset para o dispositivo baixo ativo.  Entrada da tensão de programação.  Entrada digital. |
| OSC1/CLK1  OSC1  CLK1 | 13 | E  E | Entrada cristal oscilador ou clock externo  Entrada cristal oscilador ou entrada fonte de clock externo.  Entrada fonte clock externa. Sempre associada com a função do pino OSC1. |
| OSC2/CLKO  /RA6  OSC2  CLKO  RA6 | 14 | S  S  E/S | Cristal Oscilador ou saída clock  Saída do cristal oscilador. Conectado ao cristal ou ressonador. No modo Cristal Oscilador  No modo RC, o pino de saída OSC2 terá CLKO que tem ¼ da frequência de OSC1 e denota a taxa de instrução de ciclo.  Pino de entrada e saída de propósito Geral. |

Quadro 3 – Descrição do pino Reset e Osciladores (Adaptado de MICROCHIP, 2009).

Legenda:

E = Entrada;

S = Saída;

A = Alimentação elétrica;

Neste trabalho, será utilizado oscilador externo de 20MHz, alimentação de +5V e botão para reset do microcontrolador.

Descrição dos Pinos PORTA com microcontrolador PIC18F4550 modelo PDIP

|  |  |  |  |
| --- | --- | --- | --- |
| Nome do Pino | Número do Pino | Tipo do Pino | Descrição |
| RA0/AN0  RA0  AN0 | 2 | E/S  S | PORTA é uma porta de entradas/saídas bidirecionais  Entrada/saída Digital.  Entrada analógica 0. |
| RA1/AN1  RA1  AN1 | 3 | E/S  S | Entrada/saída Digital.  Entrada analógica 1. |
| RA2/AN2/VREF-/CVREF  RA2  AN2  VREF-  CVREF | 4 | E/S  E  E  S | Entrada/saída Digital.  Entrada analógica 2.  Entrada tensão (baixa) de referência A/D.  Saída de referência do comparador analógico. |
| RA3/AN3/VREF+  RA3  AN3  VREF+ | 5 | E/S  E  E | Entrada/saída Digital.  Entrada analógica 3.  Entrada tensão (alta) de referência A/D. |
| RA4/T0CKI/C1OUT/RCV  RA4  T0CKI  C1OUT  RCV | 6 | E/S  E  S  E | Entrada/saída digital.  Entrada clock externoTimer0  Saída comparador 1.  Entrada transceptor USB externo RCV. |
| RA5/AN4//  HLVDIN/C2OUT  RA5  AN4    HLVDIN  C2OUT | 7 | E/S  E  E  E  S | Entrada/saída digital.  Entrada analógica 4.  Entrada SPI seleção escravo.  Entrada do detector de tensão Alta/Baixa.  Saída do comparador 2. |
| RA6 | - | - | Ver pino OSC2/CLKO/RA6. |

Quadro 4 – Descrição do pino PORTA (Adaptado de MICROCHIP, 2009).

Legenda:

E = Entrada;

S=Saída;

Neste trabalho, será ligado ao PORTA um display LCD e o oscilador externo de 20MHz.

Descrição dos Pinos PORTB com Microcontrolador modelo PDIP

|  |  |  |  |
| --- | --- | --- | --- |
| Nome do Pino | Número do Pino | Tipo do Pino | Descrição |
| RB0/AN12/INT0/FLT0/SDI/SDA  RB0  AN12  INT0  FLT0  SDI  SDA | 33 | E/S  E  E  E  E  E/S | PORTB é uma porta de entradas/saidas bidirecionais. PORTB pode ser programada via software para pull-ups internos em todas as entradas.  Entrada/saída digital.  Entrada analógica 12.  Interrupção externa 0.  Entrada Falha PWM aprimorada.  Entrada de dados SPI.  Entrada/saída de dados I²CTM. |
| RB1/AN10/INT1/SCK/SCL  RB1  AN10  INT1  SCK  SCL | 34 | E/S  E  E  E/S  E/S | Entrada/saída digital.  Entrada analógica 10.  Interrupção externa 1.  Entrada/saída de clock serial síncrono para modo SPI.  Entrada/saída de clock serial síncrono para modo I²C. |
| RB2/AN8/INT2/VMO  RB2  AN8  INT2  VMO | 35 | E/S  E  E  S | Entrada/saída digital.  Entrada analógica 8.  Interrupção externa 2.  Saída transceptor USB externo VMO. |
| RB3/AN9/CCP2/VPO  RB3  AN9  CCP2  VPO | 36 | E/S  E  E/S  S | Entrada/saída digital.  Entrada analógica 9.  Entrada Capture 2/ saída Compare 2/ saída PWM2.  Saída transceptor USB externo VPO. |
| RB4/AN11/KBI0/CSSPP  RB4  AN11  KBI0  CSSPP | 37 | E/S  E  E  S | Entrada/saída digital.  Entrada analógica 11.  Pino interrupção de mudança de estado.  Saída seleção de controle chip SPP. |
| RB5/KBI1/PGM  RB5  KBI1  PGM | 38 | E/S  E  E/S | Entrada/saída digital.  Pino interrupção de mudança de estado.  Pino habilitação programável baixa-tensão ICSP™. |
| RB6/KBI2/PGC  RB6  KBI2  PGC | 39 | E/S  E  E/S | Entrada/saída digital.  Pino interrupção de mudança de estado.  Pino clock programável ICSP e In-Circuit Debugger. |
| RB7/KBI3/PGD  RB7  KBI3  PGD | 40 | E/S  E  E/S | Entrada/saída digital.  Pino interrupção de mudança de estado  Pino dados programáveis ICSP e In-Circuit Debugger. |

Quadro 5 – Descrição do pino PORTB (Adaptado de MICROCHIP, 2009).

Legenda:

E = Entrada;

S=Saída;

No PORTB serão conectados os leds, o motor de passo.

Descrição dos Pinos PORTC conforme encapsulamento PDIP.

|  |  |  |  |
| --- | --- | --- | --- |
| Nome do Pino | Número do Pino | Tipo do Pino | Descrição |
| RC0/T1OSO/T13CKI  RC0  T1OSO  T13CKI | 15 | E/S  S  E | PORTC é uma porta de entradas/saídas bidirecionais.  Entrada/saída digital.  Saída do oscilador do Timer 1.  Entrada clock externo Timer1/Timer3. |
| RCT1/T1OSI/  CCP2/  RC1  T1OSI  CCP2 | 16 | E/S  E  E/S  S | Entrada/saída digital.  Entrada do oscilador do Timer 1.  Entrada Capture 2/saída Compare 2 / saída PWM2.  Saída transceptor USB externo |
| RC2/CCP1/P1A  RC2  CCP1  P1A | 17 | E/S  E/S  S | Entrada/saída digital.  Entrada Capture 1/saída Compare 1/ saída PWM1.  Saída CCP1 PWM aprimorada, canal A. |
| RC4/D-/VM  RC4  D-  DVM | 23 | E  E/S  E | Entrada digital.  USB differential minus line (input/output).  External USB transceiver VM input. |
| RC5/D+/VP  RC5  D+  VP | 24 | E  E/S  E | Entrada digital.  USB differential plus line (input/output).  External USB transceiver VP input. |
| RC6/TX/CK  RC6  TX  CK | 25 | E/S  S  E/S | Entrada/saída digital.  Transmissão EUSART assíncrono.  Clock EUSART síncrono. |
| RC7/RX/DT/SDO  RC7  RX  DT  SDO | 26 | E/S  E  E/S  S | Entrada/saída digital.  Recepção EUSART assíncrono.  Dados EUSART síncrono.  Saída de dados SPI. |

Quadro 6 – Descrição dos pinos PORTC (Adaptado de MICROCHIP, 2009)

Legenda:

E = Entrada;

S = Saída;

Neste trabalho serão conectados a essa porta: botões, comunicação USB, uma ventoinha (para aplicação PWM) e led’s para sinalização.

Descrição dos pinos PORTD conforme encapsulamento PDIP.

|  |  |  |  |
| --- | --- | --- | --- |
| Nome do Pino | Número do Pino | Tipo do Pino | Descrição |
| RD0/SPP0  RD0  SPP0 | 19 | E/S  E/S | PORTD é uma porta de entradas/saídas bidirecionais ou Transmissor de Porta Paralela (SPP na sigla em inglês).  Entrada/saída digital.  Dados da transmissão de porta paralela. |
| RD1/SPP1  RD1  SPP1 | 20 | E/S  E/S | Entrada/saída digital.  Dados da transmissão de porta paralela. |
| RD2/SPP2  RD2  SPP2 | 21 | E/S  E/S | Entrada/saída digital.  Dados da transmissão de porta paralela. |
| RD3/SPP3  RD3  SPP3 | 22 | E/S  E/S | Entrada/saída digital.  Dados da transmissão de porta paralela. |
| RD4/SPP4  RD4  SPP4 | 27 | E/S  E/S | Entrada/saída digital.  Dados da transmissão de porta paralela. |
| RD5/SPP5/P1B  RD5  SPP5  P1B | 28 | E/S  E/S  S | Entrada/saída digital.  Dados da transmissão de porta paralela.  Saída aprimorada CCP1 PWM, canal B. |
| RD6/SPP6/P1C  RD6  SPP6  P1C | 29 | E/S  E/S  S | Entrada/saída digital.  Dados da transmissão de porta paralela.  Saída aprimorada CCP1 PWM, canal C. |
| RD7/SPP7/P1D  RD7  SPP7  P1D | 30 | E/S  E/S  S | Entrada/saída digital.  Dados da transmissão de porta paralela.  Saída aprimorada CCP1 PWM, canal D. |

Quadro 7 – Descrição dos pinos PORTD (Adaptado de MICROCHIP, 2009).

Legenda:

E = Entrada;

S = Saída;

Neste trabalho serão conectados ao PORTD os displays de 7 segmentos.

Descrição dos pinos PORTE e alimentação elétrica.

|  |  |  |  |
| --- | --- | --- | --- |
| Nome do Pino | Número do Pino | Tipo do Pino | Descrição |
| RE0/AN5/  CK1SPP  RE0  AN5  CK1SPP | 8 | E/S  E  S | PORTE é uma porta de entradas/saídas bidirecionais.  Entrada/saída digital.  Entrada analógica 5.  Saída do clock 1 SPP. |
| RE1/AN6/  CK2SPP  RE1  AN6  CK2SPP | 9 | E/S  E  S | Entrada/saída digital.  Entrada analógica 6.  Saída do clock 2 SPP. |
| RE2/AN7/OESPP  RE2  AN7  OESPP | 10 | E/S  E  S | Entrada/saída digital.  Entrada analógica 7.  Saída de habilitar saída SPP. |
| RE3 | - | - | Ver pino /VPP/RE3. |
| VSS | 12,31 | A | Referência Terra para lógica e pinos de E/S. |
| VDD | 11,32 | A | Alimentação positiva para lógica e pinos de E/S. |
| VUSB | 18 | A | Regulador de saída tensão USB interno 3,3V, alimentação positiva para transceptor USB. |

Quadro 8 – Descrição dos pinos PORTE (Adaptado de MICROCHIP, 2009).

Legenda:

E = Entrada;

S = Saída;

Neste trabalho serão conectados ao PORTE os resistores variáveis e um sensor de temperatura.

### 2.1.6 Interrupções

Uma interrupção é um evento de hardware que, quando ocorre, provoca um desvio no programa, para que este evento seja tratado de forma prioritária em um vetor de interrupção. Ao término do tratamento do evento que deu origem à interrupção, o programa retorna ao ponto em que parou quando ocorreu o desvio. Pode-se considerar a interrupção como uma chamada de sub-rotina originada pelo hardware interno do microcontrolador. Este tipo de tratamento opera de modo inverso ao *scanning*. Nos microcontroladores da família PIC18, diferentemente do que existe na família PIC16, existem dois vetores de interrupção: um de alta e outro de baixa prioridade, sendo assim, melhora-se o mecanismo que permite atribuir prioridades diferentes às interrupções (sendo que a interrupção de maior prioridade pode parar uma interrupção de menor prioridade que esteja sendo tratada). No PIC18F4550 temos 20 formas diferentes de interrupções e como elas acontecem (ZANCO, 2008; PEREIRA,2010; SOUSA,SOUZA, LAVINIA 2010).

São elas:

1. Interrupções externas: geradas pelos pinos RB0, RB1 e RB2 configurados como entrada podendo ser configuradas para borda de subida (nível lógico de 0 para 1) e borda de descida (nível lógico de 1 para 0);
2. Interrupção por mudança de estado: portas RB4, RB5, RB6 e RB7 configuradas como entrada. Acontece em ambas as bordas comparando o nível existente entre o pino e o *latch* interno (se estiverem diferentes causa interrupção);
3. Interrupção do Timer 0: acontece quando o *timer* (TMR0) estoura o seu contador. No caso de 8bits de FFh para 00h e no caso de 16 bits de FFFFh para 0000h. Utilizado para controles precisos de tempo;
4. Interrupção do Timer 1: Ocorre quando este contador (TMR1) de 16 bits estoura;
5. Interrupção do Timer 2: Ocorre quando este contador (TMR2) de 8 bits acrescido do seu Potscaler estoura;
6. Interrupção do Timer 3: Ocorre quando este contador (TMR3) de 16 bits estoura;
7. Interrupção da porta paralela (PSP): ligado à porta paralela tipo escravo (PSP – *Parallel* *Slave* *Port*, em português, Porta paralela escrava). Ocorre quando uma operação de escrita ou leitura é finalizada;
8. Interrupção dos conversores Analógico/Digital: Ocorre quando uma conversão Analógica/Digital é completada;
9. Interrupção de recepção da USART: Ocorre ao término da recepção de um dado pela USART (*Universal* *Synchronous* *Asynchronous* *Receiver* Transmitter);
10. Interrupção de transmissão da USART: Ocorre quando o buffer relacionado à transmissão de um dado pela USART foi esvaziado.
11. Interrupção da comunicação serial (SPI e I²C): Ocorre quando o segundo sistema de comunicação serial ( no modo SPI ou I²C) recebe ou transmite um dado pela porta serial.
12. Interrupção do CCP1 (Captura/Compare/PWM): Associado ao primeiro sistema CCP.
13. Interrupção do CCP2: Associado ao segundo sistema CCP.
14. Interrupção do FSCM (*Fail* *Save* *Clock* *Monitor*): Ocorre quando o oscilador principal falhar, possibilitando a mudança do oscilador e impedindo que o programa pare de rodar.
15. Interrupção de colisão de dados: Acontece quando há colisão de dados na comunicação serial I²C.
16. Interrupção dos comparadores: Está ligada à alteração do nível lógico dos comparadores internos do microcontrolador.
17. Interrupção de fim de escrita na EEPROM/FLASH: Ocorre quando é detectado o final de uma rotina de escrita nessa memória.
18. Interrupção do HLVD (*High*/*Low* *Voltage* *Detector*): Ocorre para informar queda ou subida de tensão na alimentação do microcontrolador.

(SOUSA, SOUZA, LAVINIA, 2010; MICROCHIP, 2009).

Uma interrupção pode ser tratada, ou seja, interromper o programa, porém o microcontrolador pode estar configurado para ignorar essas tratativas. Sendo assim, deve-se configurar os registradores relacionados a interrupção no momento de programar o microcontrolador para obter o resultado esperado. Lembrando que, o vetor de interrupção de baixa prioridade desvia o programa para o endereço 0x000018 e o vetor de interrupção de alta prioridade desvia o programa, ou interrupção de baixa prioridade, para o endereço 0x000008. (ZANCO, 2008).

Para habilitar as interrupções do PIC18F4550, deve-se:

1. Setar o bit RCON<IPEN> - desta forma haverá prioridades diferentes para as interrupções
2. Ativar as chaves INTCON<GIE/GIEH> e INTCON<PEIE/GIEL> - em caso de utilizar o vetor de baixa prioridade.
3. Para interrupções de alta prioridade o bit INTCON<PEIE/GIEL> = 0 habilitará apenas as interrupções externas, mudança de estado e timer 0.

Para cada fonte de interrupção existe uma configuração diferente (SOUSA, SOUZA, LAVINIA, 2010; ZANCO 2008).

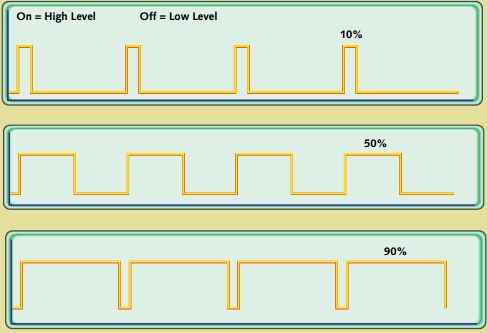
Se uma determinada interrupção estiver habilitada e ocorrer o evento responsável por ela, o flag de sinalização correspondente ao evento é setado, a execução do programa principal é interrompida, o endereço da próxima instrução que seria executada, que está armazenada no contador de programa, é colocado em uma pilha e o programa é desviado para um vetor, dependendo da prioridade da interrupção, na qual será executada uma rotina responsável pelo tratamento da interrupção. Quando a instrução RETFIE é encontrada, o endereço é resgatado da pilha e o programa volta ao ponto em que ocorreu o desvio. (ZANCO, 2008).

### 2.1.7 Modulação por largura de pulso

A modulação por largura de pulso, ou PWM (*Pulse* *Width* *Modulation*) é uma técnica poderosa para controle e circuitos analógicos utilizada em instrumentos de medição, aparelhos de comunicação e controle e conversão de energia elétrica. (BARR, 2008)

Esta é implementada digitalmente, com microcontroladores e microprocessadores, para controle de circuitos analógicos, para redução de custo e de consumo de energia. Desta maneira então, o PWM está codificando discretizadamente níveis analógicos de forma que o sinal se mantenha digital pois, em qualquer momento, a fonte de energia de corrente contínua estará totalmente ligada ou totalmente desligada. A corrente ou tensão é fornecida a carga analógica por meio de uma série de repetições de pulsos de nível alto e baixo, assim fazendo, ligando e desligando a carga e com isso, controlando o valor da alimentação na carga. (BARR,2008)

O quadro 9 retrata 3 sinais de modulações diferentes .



Quadro 9 – Diferentes modulações de PWM (BARR, 2008).

No primeiro quadro, observa-se a saída de uma a modulação está 10% do tempo ligado e 90% desligado do seu ciclo de funcionamento. A segunda tem seus valores em 50% ligado e 50% desligado e na terceira e última 90% do ciclo de funcionamento ligado e 10% desligado. Essas 3 saídas diferentes codificam três valores de sinais analógicos de 10%, 50% e 90% de toda a grandeza da fonte de energia. Por exemplo, se a fonte de energia for uma bateria de 9V e um microcontrolador estiver modulando o seu sinal como no primeiro caso da figura, em 10%, 0,9V da bateria, vão para a carga. O mesmo acontece com 50% e 90% (BARR, 2008).

Como dito anteriormente, com o recurso PWM em um microcontrolador é possível obter uma tensão analógica variável controlando a largura do pulso a partir de um único pino do microcontrolador (SOUSA, SOUZA, LAVINIA, 2010).

Genericamente, a tensão média de uma forma de onde é dada por:

Onde:

Vdc – Tensão Média;

T – Período da forma de onda;

V(t) – é a função da tensão no tempo; (SOUSA, SOUZA, LAVINIA, 2010).

As vantagens de se utilizar a modulação por largura de pulso é a permanência do sinal digital do processador para o controlador, não havendo necessidade de conversão digital-analógico, diminuindo os efeitods de ruídos. (BARR, 2008)

Este trabalho utilizará uma das duas saídas PWM do microcontrolador PIC18F4550 para variar a velocidade de um ventilador.

## 2.2 LINGUAGEM DE PROGRAMAÇÃO C

Um microcontrolador pode ser programado em diversas linguagens de programação como: *Assembly*, Linguagem C, *Ladder*, etc. Este trabalho usa para a programação do PIC18F4550 a Linguagem de programação C++ que é derivada da Linguagem de programação C.

Usando como exemplo a definição de programa para computadores, segundo Marco Medina e Cristina Fertig, programa de computador é “ [...]um conjunto de instruções que será executado pelo processador em uma determinada seqüência. Esse programa leva o computador a executar alguma tarefa.”. Nos microcontroladores temos a mesma função, um programa será responsável em guiar o microcontrolador a executar uma atividade. Neste trabalho, a Linguagem utilizada será a Linguagem C ++ (MEDINA, FERTIG,2005).

Criada em 1972 por Dennis Ritchie e Brian W. Kernighan, a partir da linguagem “B” de Ken Thompson, a Linguagem de Programação C tinha como propósito ser uma linguagem de alto nível, diferentemente da linguagem *Assembly* conhecida como de baixo nível. Tem como vantagem maior simplicidade – no quesito de ser facilmente interpretada -, poder e velocidade o que a tornou muito popular. É característica da Linguagem C ser de uso geral, isso significa que pode fazer parte da desenvolvimento de diversos tipos de aplicações. (SANTEE,2005;SILVA,2006).

## 2.3 MOTOR DE PASSO

Por meio de imãs e indutores os motores elétricos transformam energia elétrica em movimento rotativo, no mercado existe uma grande margem de motores elétricos sejam eles de corrente continua ou corrente alternada, servo motores e os de passo.

Motores de passos são dispositivos mecânicos e eletro-magnéticos que podem ser controlados digitalmente através de um hardware específico ou através de softwares. Eles são encontrados em aparelhos onde a precisão é um fator muito importante. São usados em larga escala em impressoras, *plotters*, *scanners*, *drivers* de disquetes, discos rígidos e muitos outros aparelhos. (BRINO, 2008).

O Motor de passo é muito usado em casos onde seja necessário maior precisão ou rotação em angulo. Existem 3 tipos básicos de motores de passo: relutância variável , ímã permanente e híbrido. O mais utilizado e que foi utilizado em nesse projeto é o Hibrído. aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

### 2.3.1 Motor Hibrído – (HB Hybrid)

O motor híbrido é de longe o motor de passo mais utilizado, como o nome ja diz, hibrido, é a combinação dos motores de Relutância Variável e Imã Permanente, têm ângulos de passo que variam de 0.9 a 5 graus. (BRINO, 2008; QUEIROZ, 2002).

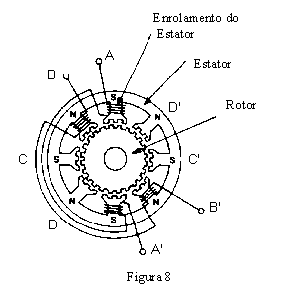


Figura 5 - Motor de passo híbrido (QUEIROZ, 2002)

### 2.3.2 Passo Normal

Excitação Única de fase – Uma fase energizada de cada vez, sendo mais utilizado em casos onde a velocidade e o torque não são importantes, apresenta baixo consumo de energia (BRINO, 2008; QUEIROZ, 2002).

Excitação Dupla - Onde são energizadas duas fases de cada vez, com isso o torque e a velocidade aumentam e também o consumo de energia (BRINO, 2008; QUEIROZ, 2002).

### 2.3.3 Meio Passo

Meio Passo é a excitação única e dual alternada que resulta em passos com a metade de um tamanho de um passo normal. Onde o consumo de energia é maior e a velocidade menor, porem a precisão é muito maior. A rotação pode ser modificada pela precisão necessária, sendo ela por numero de dentes do eixo ou em ângulo (QUEIROZ, 2002).

## 2.4 DISPLAYS

Os displays gráficos são interfaces de saída muito úteis em sistemas microprocessados, tem a função de apresentar a informação de forma mais pratica (visual ou táctil), podem ser encontrados em resoluções por exemplo: 122x32, 128x64, 240x64 e 240x128 *dot* *pixel*. Desde que as CPU’s que atendam as temporizações de leitura e escrita de dados, fornecida pelo fabricante do display, elas podem ser utilizadas (BARBACENA, 1996).

O LCD Caractere (*Liquid* *Crystal* *Display*) é responsável pela consolidação da escrita como forma de interação entre equipamento e ser humano. Começaram a ser usados em calculadoras e relógios e hoje são amplamente utilizados nos mais diversos equipamentos eletrônicos, eletrodomésticos e em instrumentos de medição de alta precisão. No mercado, temos disponível variados modelos desde LCDs capazes de exibir uns poucos caracteres até LCDs gráficos, coloridos e nos tamanhos mais variados (BARBACENA, 1996).

Estes dispositivos normalmente são compatíveis com o código ASCII, podendo gerar letras, números e caracteres especiais, caracteres katakana, europeu e gregos além da possibilidade de ar caracteres especiais ou símbolos devido a sua memória RAM interna (BARBACENA, 1996).

Os LCDs caracteres podem ser de vários tipos, diferenciados pelo número de caracteres por linha, e pelo número de linhas existentes. A tabela abaixo mostra os tipos de LCDs caracteres mais comumente encontrados no mercado.

|  |  |
| --- | --- |
| Caracteres | Linhas |
| 8 | 1 |
| 16 | 1 |
| 16 | 2 |
| 20 | 2 |
| 20 | 4 |
| Caracteres | Linhas |
| 24 | 1 |
| 24 | 2 |
| 40 | 1 |
| 40 | 2 |
| 40 | 4 |

Quadro 10 – Módulos LCD (BARBACENA, 1996).

Um LCD pode operar de três modalidades ópticas diferentes: reflexivo, transflexivo e transmissivo.

a) Reflexivo: exibe uma imagem positiva, ou seja, caracteres escuros sobre um fundo claro.

b) Transflexivo: exibe uma imagem positiva com possibilidade da utilização de um painel iluminado.

c) Transmissivo: mostra uma imagem negativa, ou seja, caracteres claros sobre fundo escuro. Sua maior performance é obtida com o painel iluminado.

Cada uma dessas modalidades é indicada para um ambiente e possuem melhor desempenho, conforme o quadro 11:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Reflexivo | Transflexivo | Transmissivo |
| Uso Exterior | Excelente | Excelente | - |
| Uso Interior | Bom | Bom (painel desligado) | Bom (painel ligado) |
| Uso na Penumbra | Não recomendado | Bom (painel desligado | Excelente (painel ligado) |
| Uso no Escuro Total | - | Excelente ( painel ligado) | Excelente (painel ligado) |

Quadro 11 – Ambiente e desempenho de LCD (ZANCO, 2008).

O cristal líquido é um estado da matéria entre o sólido e o líquido. Descobriu-se em 1970 que as moléculas poderiam ser giradas pela aplicação de tensão elétrica tornando dois estados: transparente ou opaco (ZANCO, 2008).

Para que isso seja possível, é necessário um circuito eletrônico chamado LCD controler, os mais conhecidos no mercado são: KS0066 da Sansgung Eletrônics e HD4470 fabricado pela Hitachi (ZANCO, 2008).

Este circuito é responsável por ascender ou apagar os diversos segmentos do display e ainda tem a vantagem de permitir comunicação simples com usuário programador com apenas algumas instruções. A figura 6 demonstra um diagrama de blocos simplificado de um LCD Caractere (ZANCO, 2008).

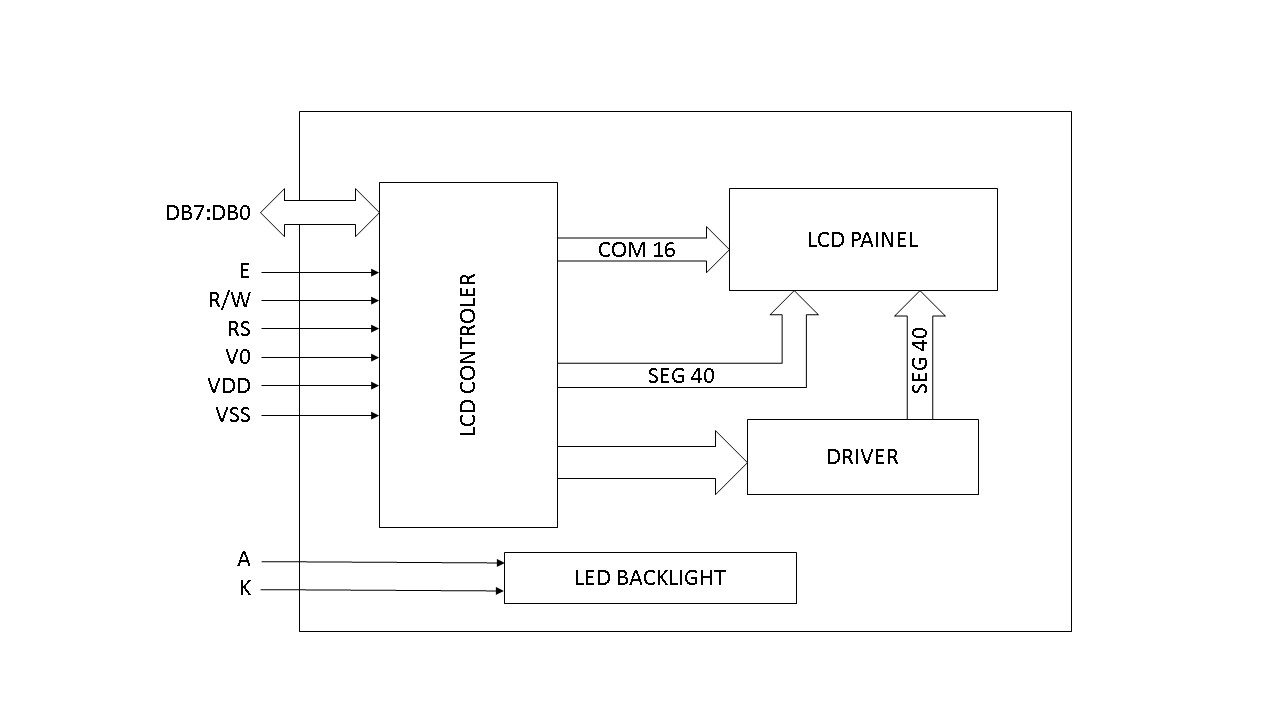


Figura 6 – Diagrama de blocos simplificado de um LCD (ZANCO, 2008).

Onde:

1. DB7:DB0 – Barramento de Dados;
2. E – Sinal habilita;
3. R/W – Leitura ou escrita de dados;
4. RS – Informações são instruções ou dados;
5. V0 – não conectado;
6. VDD – Alimentação positiva;
7. VSS – 0V;
8. A,K – Ajuste de brilho;

## 2.5 LED

Os diodos emissores de luz (LEDs) são uma junção de PN de material semicondutor que transformam energia elétrica em luz e por isso são extremamente resistentes a vibrações mecânicas, e a umidade. Uma lâmpada incandescente converte em luz apenas 5% da energia elétrica que consome, enquanto as lâmpadas LED convertem em luz até 40%. Os LEDs são utilizados desde pequenas coisas como elementos de indicação em calculadoras e aparelhos de medidas a iluminações mais modernas de arquitetura, semáforos e etc  
RANGEL, SILVA, GUEDE, 2009).

Este trabalho também conta com displays de 7 segmentos que é um display formado por 7 LEDs distribuídos em formato que ocupe todos os espaços de uma letra ou numero, o led de 7 segmentos tem por finalidade exibir uma informação seja ela binária, octa, decimal e hexadecimal em diversos tipos de aparelhos eletrônicos por se tratar de um dispositivo de fácil utilização e visualização. Cada LED corresponde a uma letra e de acordo com o seu sinal (0 ou 1) obtem-se o numero ou letra desejada (no caso de uma letra, ela estaria representando um numero em hexadecimal).

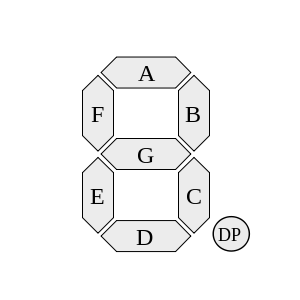
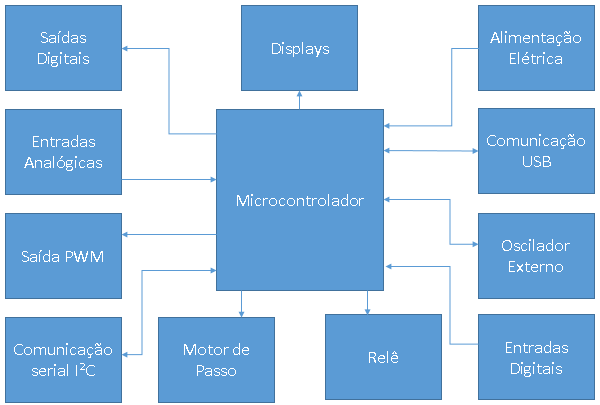


Figura 7 – Forma de um Display 7 segmentos. (LUKACHESKI, 2013).

3 MATERIAIS E MÉTODOS

Para inicio do projeto, foi elaborado um diagrama de blocos de modo a facilitar a elaboração do layout do protótipo, conforme desmontrado no quadro 12.



Quadro 12 – Diagrama de Blocos (Elaboração própria)

Nota-se que o elemento principal é o microcontrolador, no caso deste projeto o PIC18F4550 da fabricante Microchip.Tendo em mente a estrutura que o protótipo deveria possuir, partiu-se então para as ligações elétricas em simulador.

A primeira parte, é a alimentação elétrica do circuito, conforme mostra na figura 8, a alimentação virá de uma fonte de +12V que alimentará entre outros elementos um regulador de tensão 7805 (item U9 no esquema da figura 8). A saída da tensão deste regulador será de +5V que servirá para alimentar os circutios integrados, o microcontrolador, os displays de 7 segmentos e LCD entre outros conforme próximas imagens.

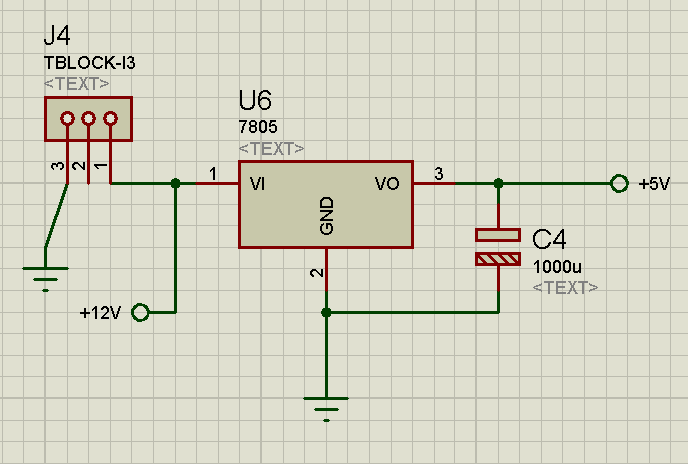


Figura 8 – Regulador de Tensão 5V (Elaboração própria)

O item J4 é um borne para circuito integrado que no terminal n°3 receberá o GND e no terminal n°1 receberá +12V de uma fonte externa. O terminal n°1 ramificará assim alimentarndo o regulador de tensão (U6) no pino n° 1. O pino n°2, será aterrado, e no pino n°3, temos +5V que alimentará outros componentes da placa. O capacitor C4 de 1000uF é utilizado para caso de quedas de tensão. Os periféricos +12V e +5V indicam conexões com periféricos de mesmo nome.

A figura 9 representa o microcontrolador PIC18F4550 – item U1 da simulação. Os seus pinos estão conectados a periféricos numerados para que fique de maneira mais clara as suas conexões. Os terminais de alimentação estão ocultos (pinos 11 e 32 para +VDD e 12 e 31 para VSS), mas estão ligados de forma que +VDD ligado +5V e VSS ligado ao GND. O capacitor C3 de 470nF conectado ao pino 18 é uma recomendação do fabricante do microcontrolador para projetos que utilizam a função USB do mesmo.

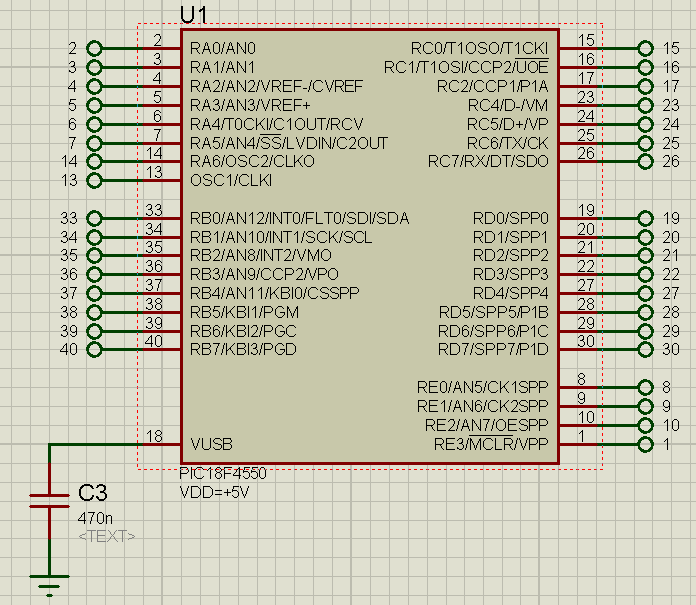


Figura 9 – Microcontrolador PIC18F4550 (Elaboração própria)

A placa do projeto foi dividida de forma a direcionar a função dos terminais com o periférico conectado. Salientando que a placa foi montada de forma à agrupar elementos para facilitar a programação da mesma.

O primeiro bloco apresentado será o bloco das entradas analógicas. Na simulação os itens RV2 e RV1 são resistores variáveis de 1K Ω e estão representando um joystick direcional analógico que possui a mesma resistência e seus terminais, serão ligados da mesma forma na placa. O item U2 é um sensor de temperatura cujo software permite mudar o valor da temperatura lida pelo sensor – o que fará com que a tensão enviada para o microcontrolador seja diferente.

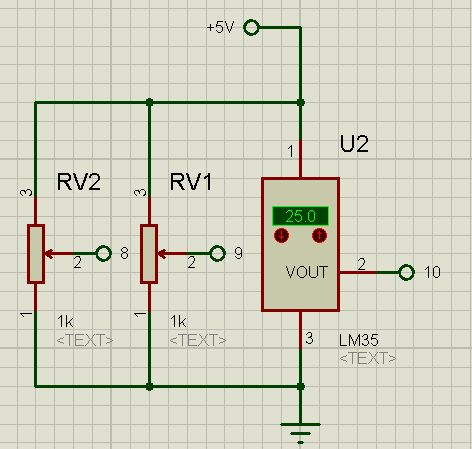


Figura 10 – Entradas analógicas (Elaboração própria)

Os terminais n°3 dos resistores variáveis estão ligados ao +5V e os terminais n°1 dos mesmos estão aterrados. Os terminais n°2 destes resistores estão ligados aos pinos 8 e 9 do microcontrolador. O sensor de temperatura LM35 – item U2 – tem sua alimentação positiva no terminal 1 e aterramento no terminal 3. O sinal analógico que irá variar com a temperatura é transmitido pelo seu terminal 2 que está ligado ao terminal 10 do microcontrolador. Os três componentes correspondem às portas RE0, RE1 e RE2 respectivamente e estes devem ser configurados como entrada analógica na programação do microcontrolador.

A próxima ligação é referente à um oscilador externo, recomendado pelo manual do microcontrolador PIC18F4550 para fonte de *clock*. Este está relacionado à velocidade do processamento do microcontrolador – seus ciclos de máquina.

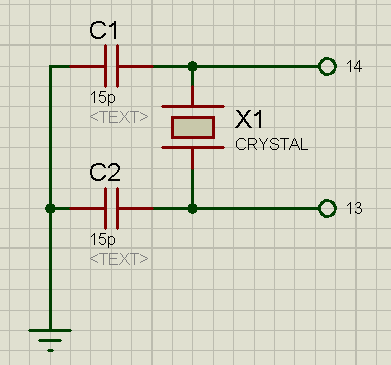


Figura 11 – Cristal de frequência (Elaboração própria)

Para um cristal de 20MHz – utilizado neste trabalho – os capacitores C1 e C2 devem assumir os valores de 15pF. A frequência de 20MHz é a maior frequência suportada por este microcontrolador. A saída do cristal está conectada aos pinos 14 e 13 do microcontrolador, nas portas para que estes itens funcionem de maneira correta, os terminais devem estar configurados para trabalhar a modo OSC1 (13) e OSC2 (14).

Para facilitar na gravação do microcontrolador de forma a se dispensar a utilização de um gravador externo e poder gravar o microcontrolador sem remove-lo do circuito utilizou-se de uma conexão USB para que a gravação seja feita por este tipo de comunicação serial com um computador. Esta etapa será detalhada quando o assunto “*bootloader*” for abordado.

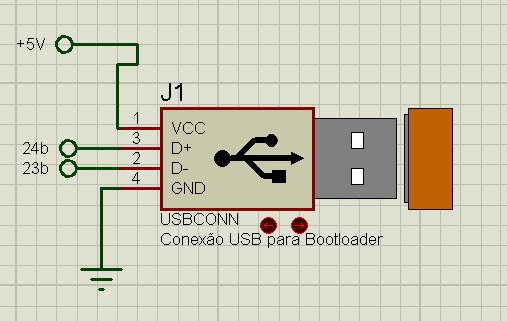


Figura 12 – Conector USB (Elaboração própria)

Como vemos na figura 12 que o conector está alimentado com +5V e aterrado em seus pinos VCC e GND respectivamente. As vias de transmissão de dados, D+ e D-, estão conectados a uma *dipswitch* que tem função de selecionar a via/periférico que estará conectado ao microcontrolador. Após a *dipswitch*, em caso das vias 23b e 24b estarem ligadas (ON), o conector USB estará ligado aos pinos 23 e 24 do microcontrolador – portas RC4 e RC5, respectivamente.



Figura 13 – Dipswitch DSW2 (Elaboração própria)

As saídas do conector USB chegam até a *dipswitch* porém só se conectarão aos seus respectivos pinos se a chave correspondente estiver na posição ON. De maneira sucinta, a dipswitch permitirá que o 23a ou 23b se conectem ao pino 23. O mesmo ocorre com as outras numerações. Este artifício foi utilizado para otimizar a placa de forma a se ter um numero maior de periféricos. O ato de ligar ou desligar as vias é de responsabilidade do programador.

Na figura 14 vemos o bloco das entradas digitais.

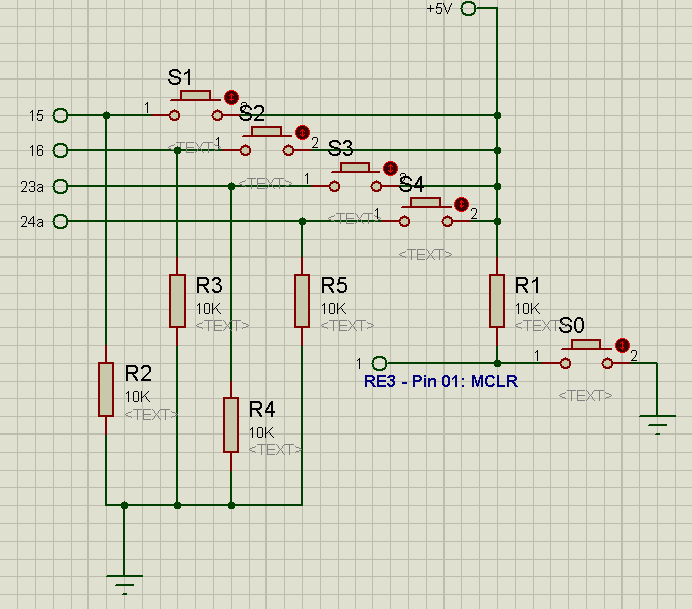


Figura 14 – Entradas digitais e botão de reset (Elaboração própria)

O bloco da figura 14 é alimentado positivamente pela saída do regulador te tensão em +5V. Nele temos os botões S0, S1, S2, S3 e S4. S0 está ligado ao pino 1 – RE3 – que é responsável em “resetar” o microcontrolador em caso de falhas. Neste caso, o nível lógico 1 se torna constante no pino 1 do microcontrolador; e caso se pressione o botão, o nível lógico enviado para o pino vai para 0, ocorrendo assim, o *reset*. Os botões S1, S2, S3 e S4 são botões de uso geral e estão ligados de forma direta aos pinos 15, 16 do microcontrolador e de forma indireta – passando por uma seleção pela *dipswitch* DSW2 da figura 13 – aos pinos 23 e 24. Estes enviam nível lógico “0” quando não estão pressionados e nível lógico 1 quando pressionados aos respectivos terminais conectados. Os resistores R1, R2, R3, R4 e R5 estão ligados à estes botões para que na condição de botões não pressionados, os terminais do microcontrolador não recebam nível lógico indefinido. Para o funcionamento correto dos botões, as portas referentes aos terminais em que os botões estão conectados, devem ser programadas como entrada de dados.

Na sequência, a ligação dos LED’s na figura 15.

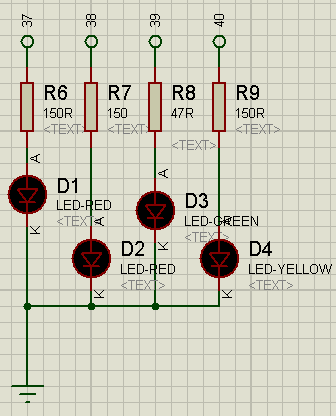


Figura 15 – saídas digitais, LEDs (Elaboração própria)

Os leds D1, D2, D3 e D4 conforme a figura 15 estão ligados de forma que o terminal catodo está aterrados e anodo conectado aos pinos 37, 38, 39 e 40 respectivamente, do microcontrolador. Desta forma, para poder ascende-los, deve-se configurar a porta referente ao pino como saída, e com nível lógico 1 (+5V) nesta saída. Os leds D1 e D2 são vermelhos e D3 verde e D4 amarelo. As cores foram variadas com a ideia de se ter alternativas de indicações luminosas. Ex: verde para “Equipamento Rodando” e amarelo para “Equipamento em Falha”, etc. Os resistores R6, R7, R8 e R9 limitam a corrente que circulam pelo led e foram calculadas da seguinte maneira:

1. Leds Vermelhos: os 2 leds vermelhos trabalham tipicamente a 1,8V e 20mA suportando até 2,5V. Definindo que o LED irá trabalhar a 2,0V, 20mA e sabendo que o microcontrolador fornece 5V temos:

O valor comercial dos resistores R6 e R7 utilizados para os leds D1 e D2 serão de 150

1. Led verde: o led verde trabalha tipicamente a 3,5V e 30mA suportando até 4,0V; Definindo que o LED irá trabalhar a 3,6V e 30mA e sabendo que o microcontrolador fornece 5V temos:

O valor comercial do resistor R8 utilizados para o led D3 será de 47ohms.

1. Led amarelo: o led amarelo trabalha tipicamente a 2,0V e 20mA suportando até 2,5 V; Definindo que o LED irá trabalhar a 2,0 e 20mA e sabendo que o microcontrolador fornece 5V temos:

O valor comercial do resistor R9 utilizado para o led D4 será de 150.

O próximo componente são os dois displays de 7 segmentos:

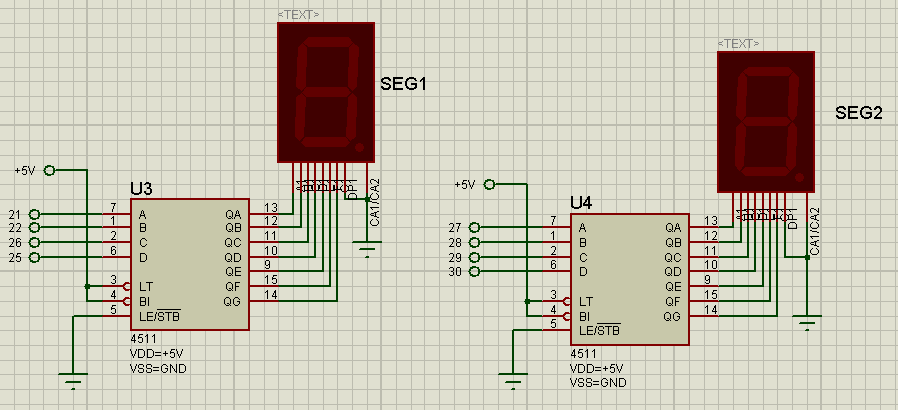


Figura 16 – Displays 7 segmentos (Elaboração própria)

Os dois *displays* da figura 16, itens SEG1 e SEG2, estão ligados a um CI 4511 cada, itens U3 e U4. Este CI é um conversor BCD para 7SEG e é utilizado nesse trabalho de modo a economizar a quantidade de portas necessárias para a comunicação com este periférico em formar números de 0 a 9. Como observado na figura 16, as entradas A, B, C e D do CI U3 estão conectadas aos pinos 21, 22, 26 e 25, respectivamente, do microcontrolador. Assim como as entradas A, B, C e D do CI U4 estão conectadas aos pinos 27, 28, 29 e 30, respectivamente, do microcontrolador.

As saídas QA, QB, QC, QD, QE, QF e QG de ambos os CIs U3 e U4 estão ligadas aos terminais A1, B1, C1, D1, E1, F1 e G1 dos displays SEG1 e SEG2, respectivamente. Por se tratar de displays de 7 segmentos de cátodo comum, um dos terminais aterrados é o comum dos leds do *display* e o outro terminal aterrado é o ponto do *display* que não será utilizado nesse trabalho.

O valor das entradas do CI 4511 e o valor de suas saídas respeita a seguinte tabela:

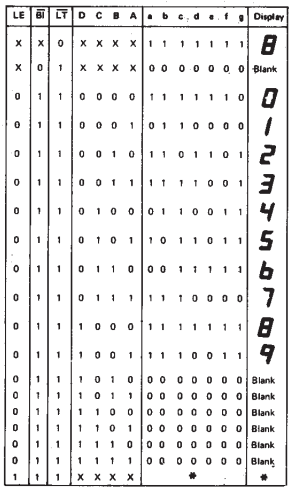


Figura 17 – Tabela da verdade CI 4511 (TEXAS, 1998)

Onde: X – Não importa o valor

\* - Depende do estado anterior

Além dos *displays* de 7 segmentos, este trabalho também conta com um *display* LCD 32x2.

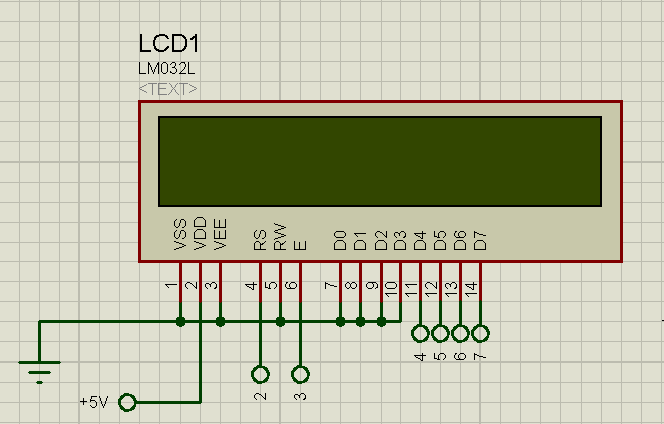


Figura 18 – Display de LCD (Elaboração própria)

O *display* conforme figura 18 está alimentado positivamente - +5V - pelo seu terminal VDD e pinos VSS, VEE, RW, D0, D1, D2 e D3 aterrados. Os pinos que terão comunicação com o microcontrolador serão: RS ligado ao pino 2 do microcontrolador; E ligado ao pino 3, D4, D5, D6 e D7 ligados aos pinos 4, 5 6 e 7 respectivamente.

Os próximos itens são dois elementos que utilizam +12V.

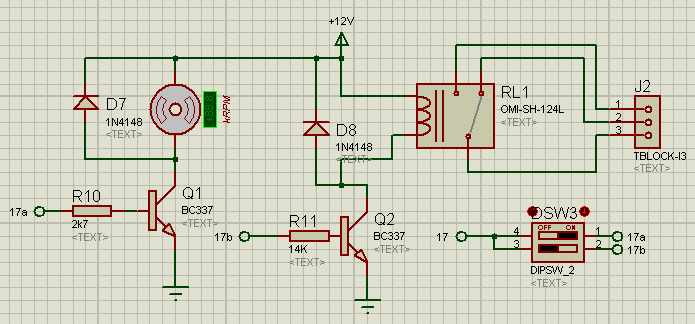


Figura 19 – Relê, Cooler e Dipswitch (Elaboração própria)

Novamente na figura 19 temos uma *dipswitch* – DSW3 – para aproveitarmos um terminal do microcontrolador e poder ligar dois periféricos diferentes que não trabalharão simultaneamente. Ao pino 17 do microcontrolador está ligado a *dipswitch* e, desta, temos duas saídas para dois circuitos com transistores. O circuito que segue a linha 17a possui um resistor R10 de 4k7, um transistor Q1 do modelo BC337, um diodo de proteção D7 do modelo 1n4148 e um cooler ligado a fonte de +12V.

O resistor de base do transistor Q1 foi definido de tal maneira:

Beta do transistor BC337 = 100

Corrente do Cooler deve ser 0,16A

Onde:

Ic – Corrente do coletor

Ib – Corrente da base

β – Ganho de corrente do transistor

Sabendo qual deve ser a Corrente da base, pode-se calcular o resistor ligado em série a base do transistor.

O valor comercial próximo ao calculado é o resistor de 2k7Ω.

Para Q2 temos:

Beta do transistor BC337 = 100

Corrente do relê RL1 do modelo CP1CR-DC12V deve ser 30mA

Onde:

Ic – Corrente do coletor

Ib – Corrente da base

β – Ganho de corrente do transistor

Sabendo qual deve ser a corrente da base, pode-se calcular o resistor ligado em série a base do transistor.

O resistor de valor aproximado comercial é de 14kΩ.

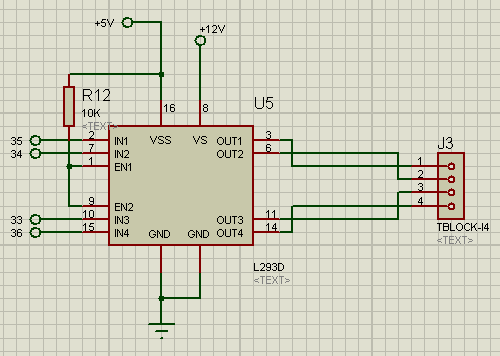


Figura 20 – L293 (Elaboração própria)

Na figura 20 encontra-se o bloco de comando do motor de passo. O L293 é um *driver* que energizará as bobinas do motor de passo fazendo-o girar dessa maneira. Este *driver* é alimentado com +12V e está conectado aos terminais 35, 34, 33 e 36. Os sinais enviados do microcontrolador para o L293D devem respeitar as seguintes tabelas para funcionamentos de meio passo e passo completo de um outro CI, o L297. Deve-se considerar que os valores de A, B, C e D do CI L297 estão associados as entradas IN1, IN2, IN3 e IN4 do L293. Também, pode-se desconsiderar os valores de INH1 e INH2, pois estarão sempre em nível lógico alto:

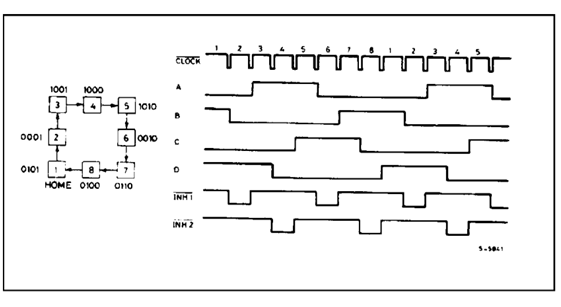


Figura 21 – Tabela verdade para meio passo L297 (STMICROELETRONICS, 2000).

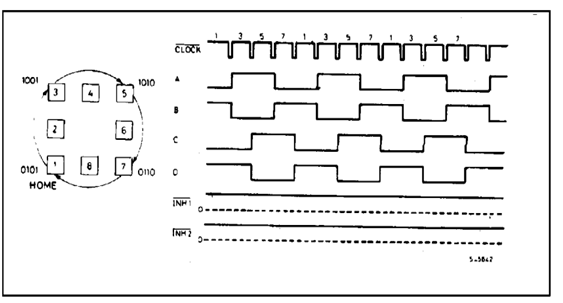


Figura 22 – Tabela da verdade para passo normal (STMICROELETRONICS,2000)

Após ligações feitas no software, foi feito a primeira possibilidade de layout da placa do projeto.

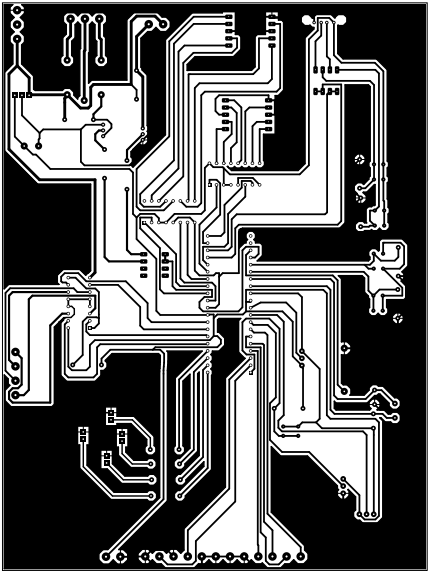


Figura 23 – Layout da placa (Elaboração própria)

Além doas trilhas, será soldado 3 fios para ligarem o terminal 5 do CI 4511 (U6) ao 8 do CI 4511 (U7); o terminal 3 do botão S2 ao terminal 4 do botão S3; o terminal 3 do resistor variável RV1 ao terminal 1 do resistor variável RV2.

## 3.1 DESENVOLVENDO O CONJUNTO

No desenvolvimento do protótipo para representação e para aplicação dos conhecimentos adquiridos foram utilizados alguns componentes eletronicos e materiais reciclados com o objetivo de minimizar os custos. Nesse capitulo é abordado o método utilizado e todos os materiais envolvidos e como eles se integraram.

Materiais:

Para o inicio da montagem da placa foram necessários um ferro de solda comum 127v, sugador de solda e solda (estanho).



Figura 24 – Ferro de solda, Solda “Estanho” e Sugador de solda (Elaboração própria)

Foram utilizados pasta de solda e fluxo de solda, para ter uma maior facilidade com o fluxo a pasta ajuda a manter a superfície limpa, fazendo com que o fluxo não fique escorrendo. Para limpeza do ferro foi utilizado uma bucha de cozinha e também um pincel pequeno para espalhar a pasta sobre os componentes.



Figura 25 – Pasta de solda e Fluxo de solda (Elaboração própria)

Uma placa fenol 15x20, soquete p/ CI 40pinos, *Dipswitch* 02 vias e *Borne* 3 terminais.



Figura 26 – Placa fenol 15x20 (Elaboração própria)

Na figura 31 estão os circutos integrados utilizados no trabalho: soquete c/ 40 pinos, PIC18F4550, CD 4511, L293D driver de motor de passo, PCF 8574P, transistors e sensor de temperatura.

|  |  |
| --- | --- |
|  |  |
|  |  |

Figura 27 – Grupo de circutos integrados (Elaboração própria).

Na figura 32 está o regulador de tensão, cristal de frequência para o circuito oscilador externo, diodos 1N4148 e o display LCD 16x2.

|  |  |
| --- | --- |
|  |  |
|  |  |

Figura 28 – CI, Semicondutores e Display LCD (Elaboração própria).

O motor de passo escolhido foi um bipolar de 12v e 1Kgf, somente usado para demonstração não tem necessidade de um torque alto, Relê 12v para segurança dos componentes, LED’s varias cores e *display* 7 segmentos.

|  |  |
| --- | --- |
| Motor De Passo 12v 1kgf |  |
|  |  |

Figura 29 – Motor de passo, relê, LED’s e display 7segmentos. (Elaboração própria)

|  |  |
| --- | --- |
|  |  |
|  |  |

Figura 30 – *Dipswitches*, conectores e botões (Elaboração própria)

|  |  |
| --- | --- |
|  |  |
|  |  |

Figura 31 – Resistores, capacitores, cabo usb e cooler (Elaboração própria)

No desenvolvimento do conjnuto será necessário tomar muito cuidado com os caminhos que serão soldados, observando que alguns caminhos são mais grossos devidos a voltagem e se cruzam em pontos de solta perto de componentes e isso pode causar a sobreposição da solda causando curto. O uso da pasta de solda para a fixação e do sugador para remover os excessos é fundamental. Após a impressão do circuito em folha como na figura 24, tomando os cuidados necessários para transpassar o circuito para a placa, recomenda-se o uso de CNC para projetos com muitos ramos que podem ser complexos na solda ou caso quem irá executar não possuir experiência em soldas. aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa  
 Seguindo o *layout* do circuito, observe que a placa possui dois lados, um de fibra e outro do CI, a solda deve ser feita no lado do CI, aplique os componentes na placa e com o ferro pré-aquecido sobre os terminais dos componentes aplique a solda, faça isso em todos os terminais. Importante observar a posição antes da solda, as entradas e saídas dos componentes são diferentes e soldar em uma posição diferente do *layout* ira causar conflito na hora da execução, sempre limpando o ferro de solda a cada componente.

Para exemplo de programação e simulação em software (Anexo A) foi utilizado programação em linguagem C++ para teste dos *displays* e leds.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item | Descrição | Quantidade | Valor (uni) | Valor Real |
| 1 | CI PIC18F4550 | 1 | R$ 32,00 | R$ 32,00 |
| 2 | CI L293D | 1 | R$ 13,95 | R$ 13,95 |
| 3 | Cabo usb Macho + Macho | 1 | R$ 9,50 | R$ 9,50 |
| 4 | Placa fenol 15x20cm | 1 | R$ 7,00 | R$ 7,00 |
| 5 | LM035 Sensor de temperatura | 1 | R$ 4,50 | R$ 4,50 |
| 6 | Botão micro chave push buttom 4 pinos | 5 | R$ 0,35 | R$ 0,12 |
| 7 | Chave gangorra redonda | 1 | R$ 2,00 | R$ 2,00 |
| 8 | Display 7 segmentos catodo comum | 2 | R$ 1,80 | R$ 3,60 |
| 9 | Relê G1RC2-1R 12V | 1 | R$ 1,80 | R$ 1,80 |
| 10 | Conector USB fêmea | 1 | R$ 1,50 | R$ 1,50 |
| 11 | Soquete p/ CI 40pinos | 1 | R$ 1,28 | R$ 1,28 |
| 12 | Cristal frequência 20MHz | 1 | R$ 1,20 | R$ 1,20 |
| 13 | Dip switch reto 02 vias | 3 | R$ 1,00 | R$ 3,00 |
| 14 | Led 5 mm vermelho alto brilho | 2 | R$ 0,80 | R$ 1,60 |
| 15 | Led 5 mm verde alto brilho | 1 | R$ 0,80 | R$ 0,80 |
| 16 | Led 5 mm amarelo alto brilho | 1 | R$ 0,30 | R$ 0,30 |
| 17 | Capacitor cerâmica 15 pF | 2 | R$ 0,10 | R$ 0,20 |
| 18 | Motor de passo 12V | 1 | R$ 25,00 | R$ 25,00 |
| 19 | Display LCD 16x2 | 1 | R$ 16,00 | R$ 16,00 |
| 20 | Cooler Universal Intel 12V | 1 | R$ 6,50 | R$ 25,00 |
| 21 | Diodo 1N4148 | 2 | R$ 5,00 | R$ 10,00 |
| 22 | BC 337 transistor | 2 | R$ 0,60 | R$ 1,20 |
| 23 | CD 4511 | 2 | R$ 2,60 | R$ 5,20 |
| 24 | Percloreto de Ferro 250gm | 1 | R$ 8,90 | R$ 8,90 |
| 25 | 7805 Regulador de tensão | 1 | R$ 2,90 | R$ 2,90 |
| 26 | Borne 3 terminais azul KRE3 | 2 | R$ 1,95 | R$ 3,90 |
| 27 | Perfurador para CI | 1 | R$ 42,90 | R$ 42,90 |
| 28 | Borne para Ci 2 terminais | 6 | R$ 1,90 | R$ 11,40 |
| 29 | Papel Glossy | 1 | R$ 1,10 | R$ 1,10 |
| 30 | Resistor 2K2 1/2 W | 1 | R$ 0,30 | R$ 0,30 |
| 31 | Resistor 10K 1/2W | 6 | R$ 0,30 | R$ 1,80 |
| 32 | Resistor 150R 1/4W | 4 | R$ 0,15 | R$ 0,60 |
| 33 | Capacitor eletrônico 10x16V | 1 | R$ 0,40 | R$ 0,40 |
|  |  |  |  | R$ 240,95 |

Quadro 13 – Orçamento dos componentes (Elaboração própria)

O valor dos materiais utilizados é de R$ 240,95 que é um valor razoável, mas comparado com o mercado e com as funções do conjunto didático viabiliza o seu custo, tirando os instrumentos que já possuíamos esse valor ainda pode ser diminuído, devido a falta de lojas especializadas na região e na dificuldade de encontrar alguns itens do projeto. O valor do Dolar no dia das compras alcançava os R$3,873 que pode ter interferido no valor um pouco alto do projeto.

1. RESULTADOS E DISCUSSÕES

O conjunto didático irá atuar como um material complementar para uso em salas de aula nas quais são desenvolvidos programas lógicos para uso em microcontroladores. Isso trouxe resultados positivos, pois, o conjunto didático age diretamente no problema que é a falta de compreensão do aluno na aula, com esta facilidade acreditamos que o problema seja diminuído.

No desenvolvimento do projeto encontramos dificuldades para a criação do programa, que integra varias funções a um mesmo microcontrolador, o que levou a troca do mesmo diversas vezes até ser encontrado um que atenda melhor a necessidade do projeto.

Os conjuntos didádticos já existentes no mercado costumam ser encontrados como bancadas de grandes marcas a fim de desenvolver o uso exclusivo do seu produto. Muitas vezes não possibilitam o uso de diversas funções, o que limita as funções que o aluno possa executar.

Durante a montagem do protótipo constatou-se a dificuldade para encontrar componentes e peças que compõe o projeto. Exemplo: o motor de passo que em nossa região não é encontrado produtos novos, atentos ao custo beneficio do projeto, foi encontrado um motor já usado, mas em bom estado.

Focando na didática do trabalho, o projetoo está sendo oferecido totalmente livre na *Creative* *Commoms* uma organização sem fins lucrativos para compartilhamento de ideias, como um projeto de Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional, para futuros estudantes desenvolverem novas versões do projeto, sem perder as características principais que seja a didática e sem nenhum custo, totalmente livre para estudantes poderem realizar seus experimentos e ajudarem futuros estudantes que também poderão usufruir de uma tecnologia bem ampla e que gera uma vasta gama de opções para serem exploradas.

# 5 CONCLUSÃO

Através de uma pesquisa desenvolvemos um protótipo que possibilita o melhor aproveitamento de aulas que utilizam microcontroladores, produzindo um conjunto didático com varias funções básicas que o aluno poderá realizar em aula e ver sendo executado no conjunto, diminuindo dúvidas e aumentando o conhecimento.

As diversas funções que podem ser atribuídas ao microcontrolador torna a viabilidade do projeto, pois há possibilidade de melhoria, preenchendo falhas no processo e criando novas áreas de execução, assim sendo desenvolvidos por próximos trabalhos acadêmicos.

Todo o projeto será disponibilizado livre para novos alunos, desde o programa até os métodos que utilizamos, sempre o mantendo aberto e sem custos, alem de aplicar nossos conhecimentos adquiridos no curso, onde se espera ver novas versões do projeto, ajudando outros alunos.

# REFERÊNCIAS

BARBACENA, Ilton L; FLEURY, Claudio Afonso. **Display LCD.** Outubro, 1996.

BARR, Michael. Pulse width modulation. **Embedded Systems Programming**, v. 14, n. 10, p. 103-104, 2001.Disponível em [http://homepage.cem.itesm.mx/ carbajal/Microcontrollers/ASSIGNMENTS/readings/ARTICLES/barr01\_pwm.pdf](http://homepage.cem.itesm.mx/carbajal/Microcontrollers/ASSIGNMENTS/readings/ARTICLES/barr01_pwm.pdf). Acesso em 04/09/2015.

BRINO, Ricardo. PROJETO DE CONSTRUÇÃO DE UM ROBÔ EXPLORADOR. 2008. Disponivel em: <http://lyceumonline.usf.edu.br/salavirtual/documentos/1201.pdf>. Acesso em: 16 de Março.

CAPUANO, Francisco Gabriel; IDOETA, Ivan V. Elementos de eletrônica digital; 40ª Edição. **Editora: Érica**, 2000.

CARMONA, João Victor Costa; VIEIRA, Geraldo da Silva. Sistema de Monitoramento do Fluxo de Soro. **Engenharia de Computação em Revista**, v. 1, n. 4, 2010. Disponível em: [http://www3.iesampa.edu.br/ojs/index.php/computacao/ article/view/172/162](http://www3.iesampa.edu.br/ojs/index.php/computacao/%20article/view/172/162). Acesso em 26 de março de 2015.

DISCENTE, INICIAÇÃO CIENTÍFICA. Abastecimento de Estoque por Robô Microcontrolado. 2008. Disponível em: <http://dxyzmecatronica.xpg.uol.com.br/apos/estoque.pdf>. Acesso em 26 de março de 2015.

MARTINS, Nardênio Almeida. Sistemas microcontrolados. **São Paulo: Novatec**, 2005. Disponível em: [http://www.martinsfontespaulista.com.br/anexos/produtos/ capitulos/203289.pdf](http://www.martinsfontespaulista.com.br/anexos/produtos/%20capitulos/203289.pdf) Acesso em 18 de março 2015.

MATIC, Nebojsa; ANDRIC, Dragan. Microcontroladores PIC. 2005. Disponível em: <http://wlmquip.com.br/Microcontroladores.pdf>. Acesso em 27 de março de 2015.

MEDINA, Marco; FERTIG, Cristina. Algoritmos e Programação. **São Paulo, Novatec Editora**, 2005. Disponivel em: http://186.202.25.177/livros/algoritmos/capitulo 857522073X.pdf.Acesso em 29/04/2015

PEREIRA, Carla Miranda; CARLOS, Daniel Falcão Lopes Princisval; DE BARROS CABRAL, Israel Gomes. SENSOR DE OBSTÁCULOS PARA DEFICIENTES VISUAIS. Disponível em: [http://inscricaofecti.cecierj.edu.br/arquivos/trabalho 00093.docx](http://inscricaofecti.cecierj.edu.br/arquivos/trabalho%2000093.docx). Acesso em 26 de março de 2015.

PÉREZ, Fernando E. Valdés; ARENY, Ramón Pallàs. **Microcontroladores: fundamentos y aplicaciones con PIC**. Marcombo, 2007. Disponível em: <https://books.google.com.br/books?hl=ptBR&lr=&id=ODenKGOHMRkC&oi=fnd&pg=PA9&dq=microcontrolador&ots=ulqS-j2bw&sig=0YYGKrBRP_JqctZ2K8qcgcSnpF0# v=onepage&q&f=true>. Acesso em 26 de março de 2015.

PIC18F4550, DATASHEET. Microchip Technology Inc, 2009.

SANTEE, André. Programação de Jogos com C++ e DirectX. **São Paulo: Novatec**, 2005. Disponível em: [http://www.martinsfontespaulista.com.br/anexos/produtos/ capitulos/143126.pdf](http://www.martinsfontespaulista.com.br/anexos/produtos/capitulos/143126.pdf). Acessado em 29/04/2015.

SANTOS, Henrique José. Curso de linguagem c, ufmg. **Universidade Federal de Minas Gerais**. Disponível em: [http://www2.dcc.ufmg.br/disciplinas/pc/source/ introducao\_c\_renatocm\_deeufmg.pdf](http://www2.dcc.ufmg.br/disciplinas/pc/source/introducao_c_renatocm_deeufmg.pdf). Acesso em 15/05/2015.

SIEVERS JR, Fretz; GERMANO, José Silvério E.; DE ALMEIDA, Felipe. kit programe fácil–um kit educacional para subsidiar a programação de microcontroladores em sala de aula. In: **Anais do Simpósio Brasileiro de Informática na Educação**. 2009. Disponível em: [http://www.niee.ufrgs.br/ eventos/SBIE/2009/conteudo/artigos/completos/61932\_1.pdf](http://www.niee.ufrgs.br/eventos/SBIE/2009/conteudo/artigos/completos/61932_1.pdf). Acesso em 26 de março de 2015.

SILVA, Erick. **Desenvolvimento de uma metodologia para captura de imagens usando microcontroladores.** Disponível em: [http://tcc.ecomp.poli.br/20052/ ErickFranklin.pdf](http://tcc.ecomp.poli.br/20052/ErickFranklin.pdf). Acesso em 18 de março de 2015.

SILVA, Renato A.; PROGRAMANDO MICROCONTROLADOR, P. I. C. Linguagem C. **São Paulo: Ensino Profissional**, 2006. Disponível em: http://files.dtsilva. webnode.pt/ 200000087-adb28aeac7/Sebenta\_C.pdf . Acesso em: 18 de março de 2015

SILVA ZANCO, Wagner. **Micontroladores PIC: técnicas de software e hardware para projetos de circuitos eletrônicos: com base no PIC16F877A**. Érica, 2008.

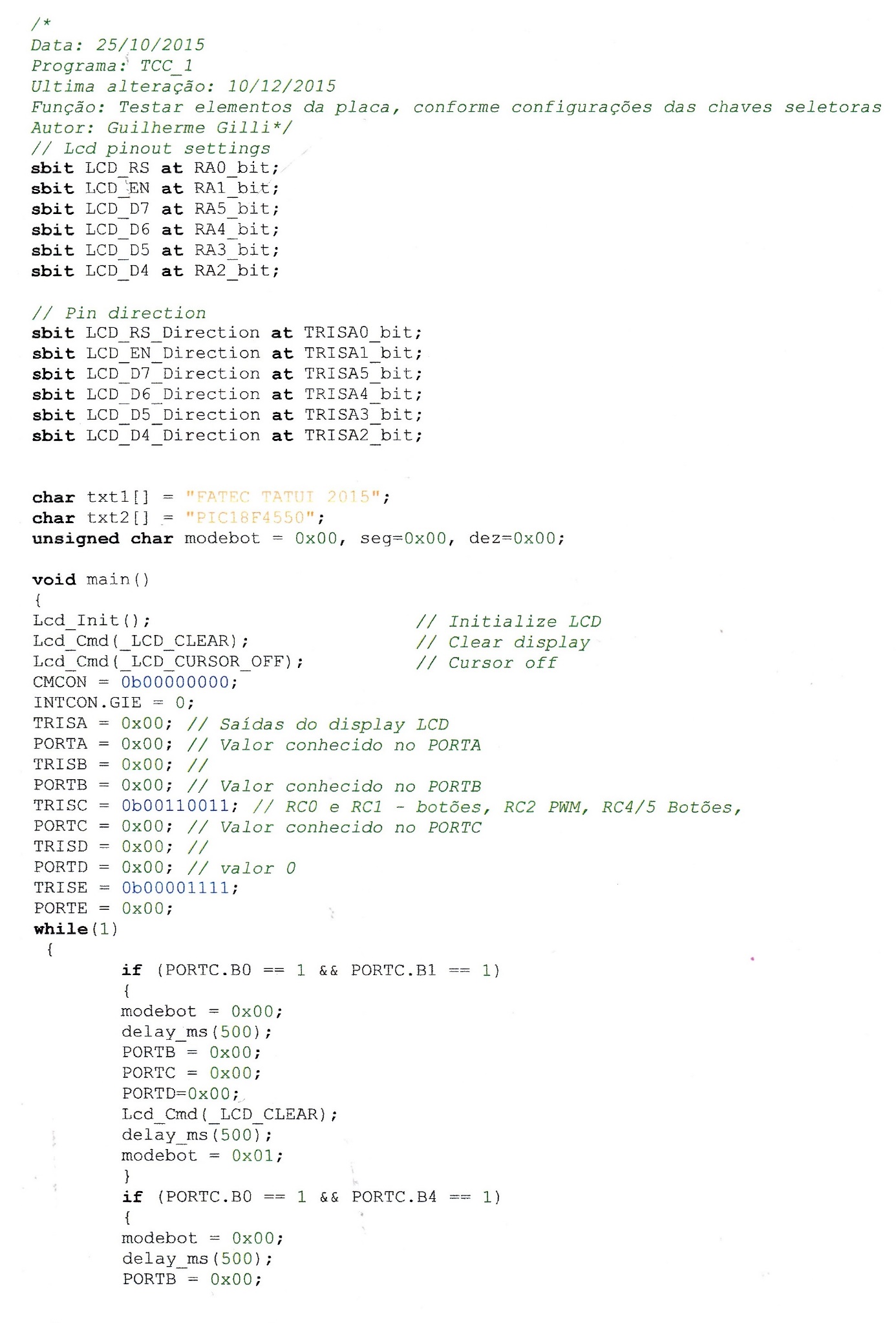
SILVA ZANCO, WAGNER. Microcontroladores PIC16F628A/648A. **Editora Erica**, 2005.

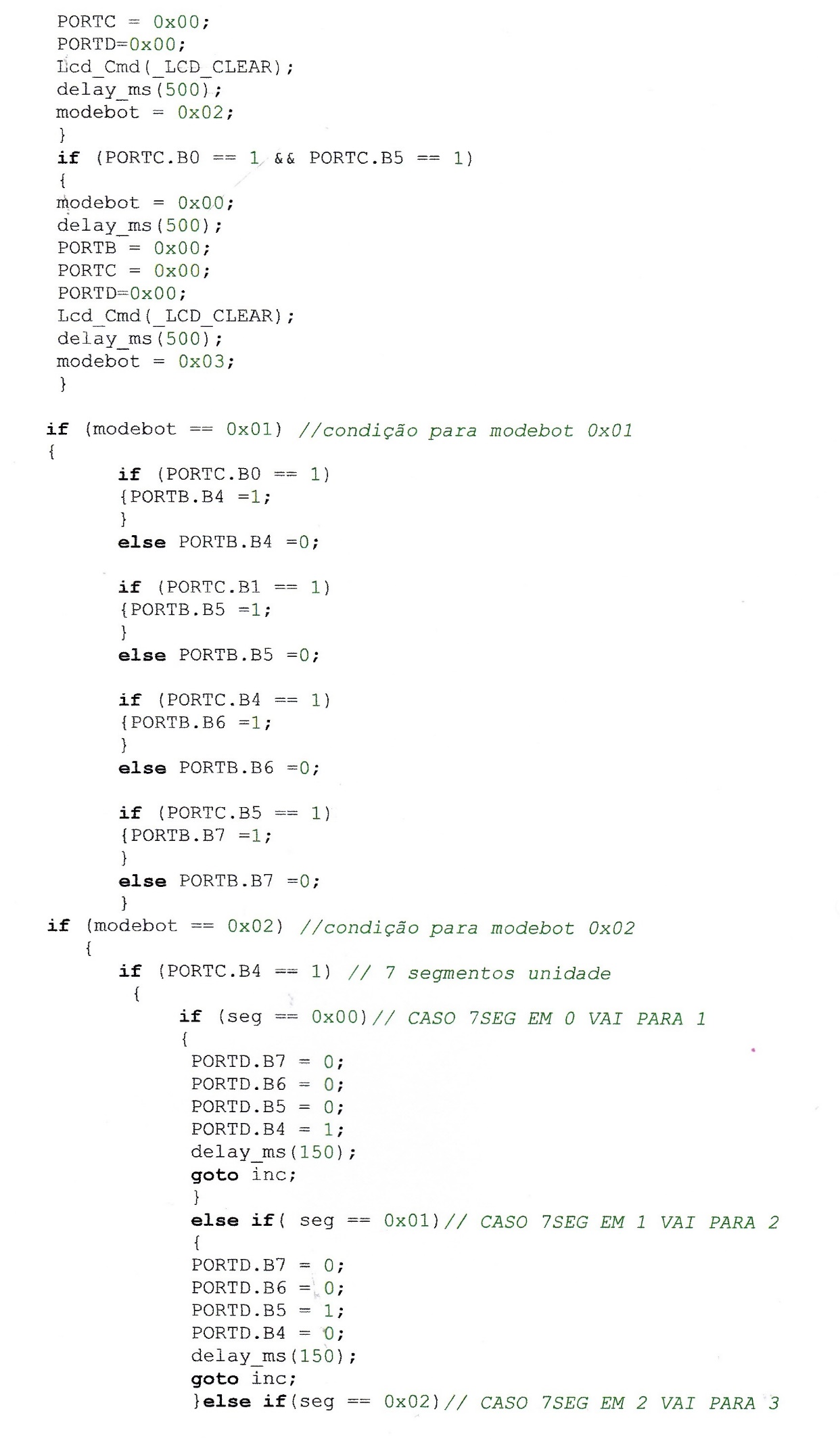
SOUSA, Daniel Rodrigues de; SOUZA, David José de; LAVINIA, Nicolás César. Desbravando o Microcontrolador PIC18-Recursos Avançados. **São Paulo: Érica**, 2010.

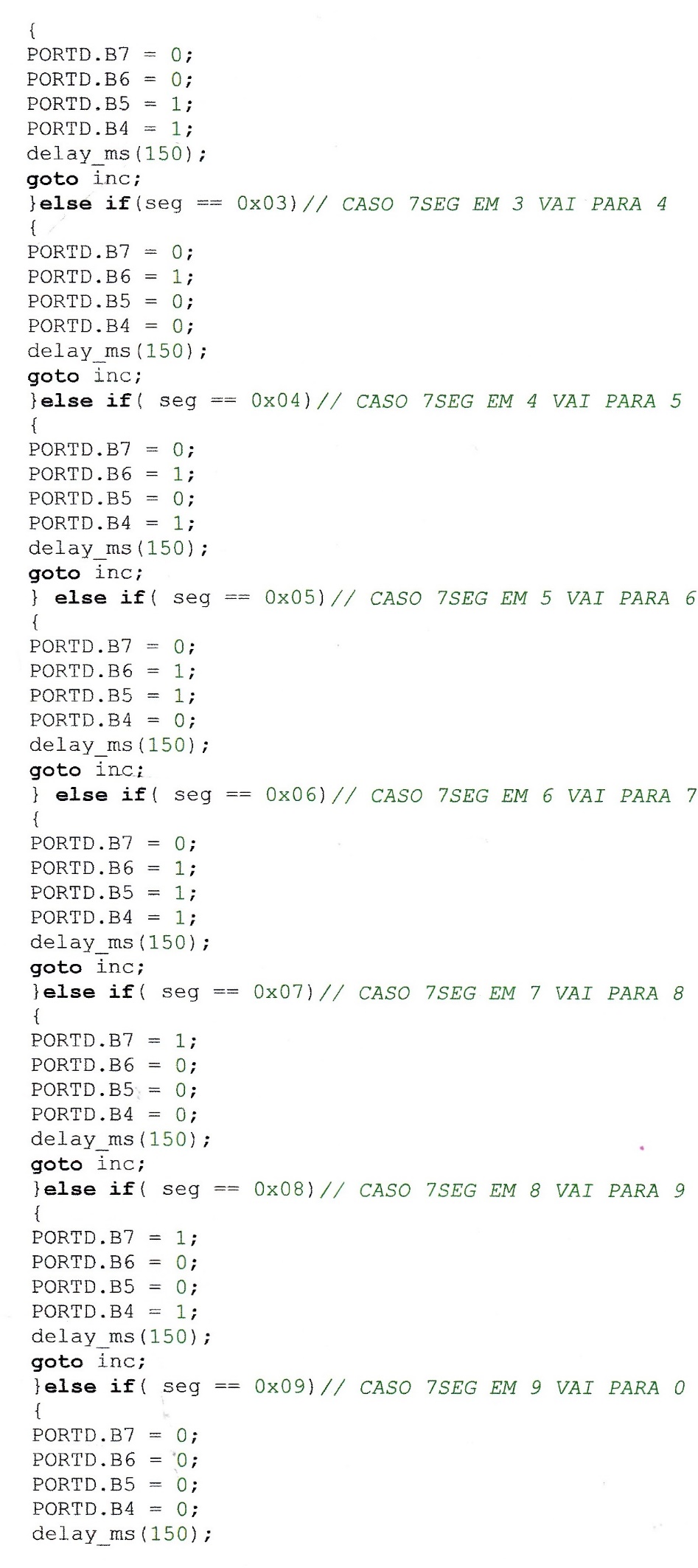
SOUZA, David José. **Desbravando o PIC: Ampliado e atualizado para PIC16F628A**, 12ª Edição. Érica, 2010.

QUEIROZ, Ricardo Alexandre. **Motores de Passo.** Universidade de Salvador. Disponível em: http://www.ppgel.ufsj.edu.br/uaisoccer/downloads/1272062510.pd. Acessado em: 18 de abril de 2015.

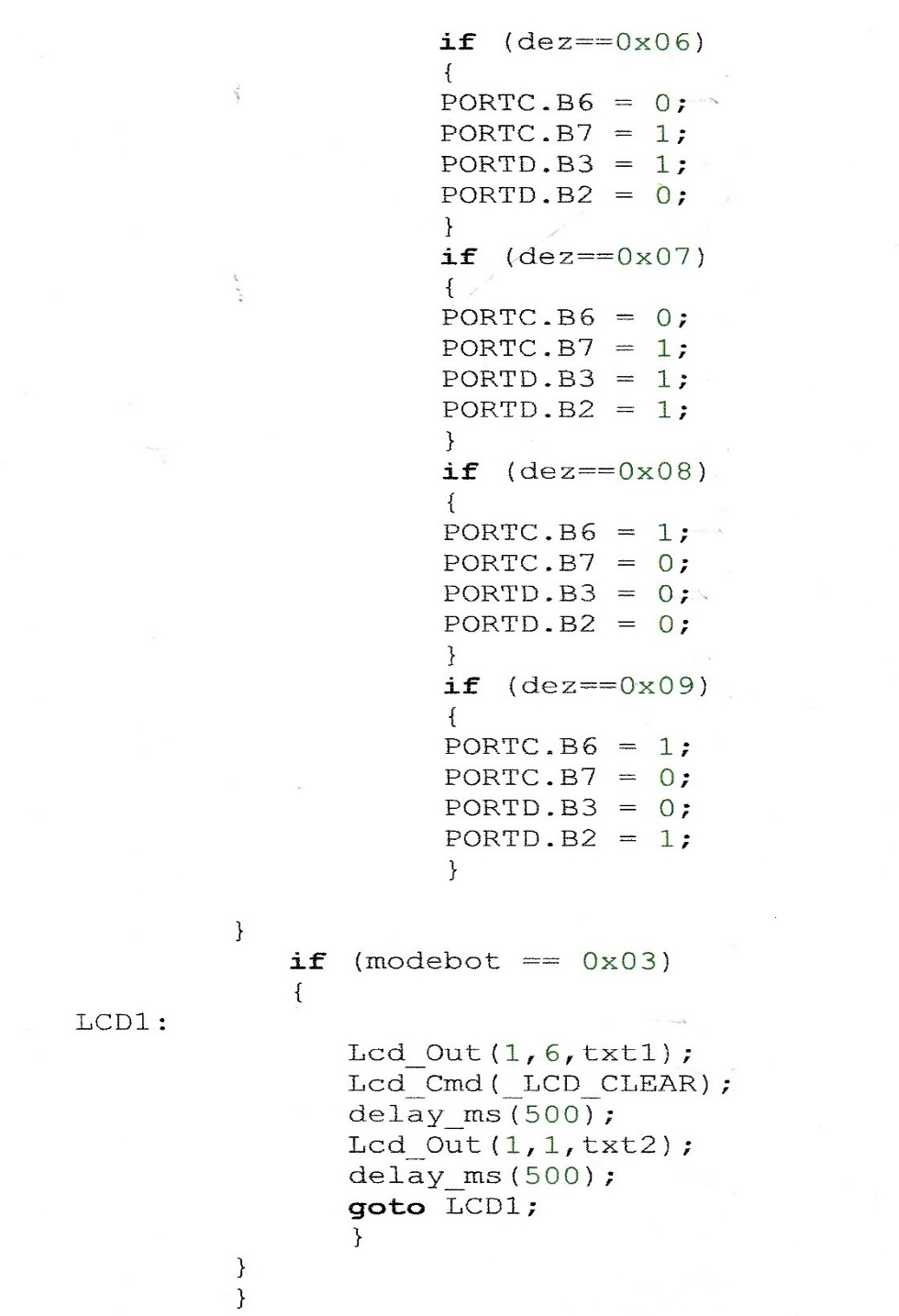
ANEXO A – Exemplo de programação para o conjunto didático

****

****

****

****

****

**Configuração dos Registros:**

**CONFIG1L :$300000 : 0x0000**

**CONFIG1H :$300001 : 0x000E**

**CONFIG2L :$300002 : 0x001F**

**CONFIG2H :$300003 : 0x001E**

**CONFIG3H :$300005 : 0x0083**

**CONFIG4L :$300006 : 0x0081**

**CONFIG5L :$300008 : 0x000F**

**CONFIG5H :$300009 : 0x00C0**

**CONFIG6L :$30000A : 0x000F**

**CONFIG6H :$30000B : 0x00E0**

**CONFIG7L :$30000C : 0x000F**

**CONFIG7H :$30000D : 0x0040**