



CENTRO UNIVERSITÁRIO DE BRASÍLIA – UniCEUB  
FACULDADE DE TECNOLOGIA E CIÊNCIAS SOCIAIS APLICADAS  
CURSO DE ENGENHARIA DA COMPUTAÇÃO

MANUELLA THEREZA PEREIRA POUSO

**SISTEMA DE AUTOMAÇÃO E CONTROLE DE UM SISTEMA DE  
IRRIGAÇÃO**

**Orientador (a):** Professora M.C. Maria Marony Sousa Farias.

Brasília - DF

2º Semestre de 2012



MANUELLA THEREZA PEREIRA POUSO

SISTEMA DE AUTOMAÇÃO E CONTROLE DE UM SISTEMA DE IRRIGAÇÃO

Trabalho de conclusão de curso apresentado ao UniCEUB – Centro Universitário de Brasília como pré-requisito à obtenção de Certificação de Conclusão do Curso de Engenharia de Computação.

Orientadora: Prof.<sup>a</sup> M.C. Maria Marony Sousa Farias Nascimento.

Brasília - DF  
2º Semestre de 2012

MANUELLA THEREZA PEREIRA POUSO

SISTEMA DE AUTOMAÇÃO E CONTROLE DE UM SISTEMA DE IRRIGAÇÃO

Trabalho de conclusão de curso apresentado ao UniCEUB – Centro Universitário de Brasília como pré-requisito à obtenção de Certificação de Conclusão do Curso de Engenharia de Computação.

Este Trabalho foi julgado adequado para a obtenção do Título de Engenheiro de Computação, e aprovado em sua forma final pela Faculdade de Tecnologia e Ciências Sociais Aplicadas - FATECS.

Abiezer Amarília Fernandes

Prof. Abiezer Amarília Fernandes  
Coordenador do Curso

**Banca Examinadora:**

Maria Marony Sousa Farias

Prof.<sup>a</sup> Maria Marony Sousa Farias, Mestre em Engenharia Elétrica – UFPB – PB.  
Orientadora

Flávio Antonio Klein

Prof. Flávio Antonio Klein, Mestre - UNB

João Marcos Souza Costa

Prof. João Marcos Souza Costa, Especialista, Matemática, UNB

Vera Lucia Fiarini Alves Duarte

Prof.<sup>a</sup> Vera Lucia Fiarini Alves Duarte, Matemática

## **Dedicatória**

*Dedico este projeto a todos que me ajudaram a concluí-lo e a mais uma superação da minha vida.*

## **AGRADECIMENTO**

A minha família, pela ajuda na vida e pela de esperança e o suporte que os mesmos me deram para que eu conseguisse enfrentar todos os desafios que eu sofri durante esse período de aprendizado e crescimento profissional.

Aos amigos que me ajudaram a superar meus medos e me fizeram acreditar que eu podia conseguir chegar ao fim do curso de Engenharia da Computação.

Agradecer principalmente a todas as pessoas que me ajudaram na conclusão do projeto e da monografia, pois vocês foram quem mais me incentivaram a crescer.

“A dor é inevitável.

O sofrimento é opcional...”

**(Drummond)**

## RESUMO

Este trabalho apresenta um sistema de controle de irrigação, cujo objetivo é otimizar o consumo de água na irrigação de jardins. O sistema proposto deve manter a umidade do solo dentro da faixa apropriada, evitando desperdício de água, além de permitir que o usuário acompanhe o seu consumo durante o funcionamento. O objetivo foi alcançado através da construção de um dispositivo para controlar a umidade do solo, abrindo e fechando válvula de controle de fluxo de água, conforme seja necessário, e para registrar o consumo de água no processo. Para a criação de tal dispositivo, foi necessário o estudo e aplicação de sensores de umidade do solo, temperatura e de fluxo de água. O resultado deste trabalho é um protótipo do dispositivo proposto, utilizando um microcontrolador para realizar a leitura dos sensores, ativar a válvula de controle de fluxo quando necessário, registrar o consumo de água e apresentar as informações relevantes em um painel LCD.

Palavras chaves: Irrigação, Microcontrolador, Consumo de água, Fluxo de água, Umidade do solo.

## **ABSTRACT**

This paper presents a control system of irrigation, which aims to optimize the consumption of water to irrigate gardens. The proposed system must maintain soil moisture within the proper range, avoiding water wastage, and allows the user to track their consumption during operation. The goal was achieved through the construction of a device for controlling soil moisture, opening and closing the flow control valve for water as necessary and for recording water consumption in the process. To create such a device, it was necessary the study and application of soil moisture sensors, temperature and water flow. The result of this work is a prototype of the proposed device using a microcontroller to read the sensors activate the flow control valve when needed, to record water consumption and to provide relevant information on an LCD panel.

Keywords: Irrigation, Microcontroller, Water consumption, Water flow, Soil Moisture.

## SUMÁRIO

<b>AGRADECIMENTO .....</b>	<b>VI</b>
<b>RESUMO.....</b>	<b>VIII</b>
<b>ABSTRACT.....</b>	<b>IX</b>
<b>SUMÁRIO .....</b>	<b>X</b>
<b>LISTA DE FIGURAS.....</b>	<b>12</b>
<b>1. INTRODUÇÃO .....</b>	<b>14</b>
1.1 Consumo de água .....	14
1.2 Água doce do Planeta .....	15
1.3 Métodos convencionais de irrigação.....	15
1.4 Vegetação.....	16
<b>2. REFERENCIAL TEÓRICO E TECNOLÓGICO.....</b>	<b>18</b>
2.1 Microcontroladores .....	18
2.1.1 Microcontroladores da Família PIC.....	19
2.2 Sensores.....	21
2.2.1 Sensor de Umidade .....	23
2.2.2 Sensor de Temperatura.....	26
2.2.3 Sensor de Fluxo de Fluídos .....	26
2.3 Válvulas .....	26
2.3.1 Válvula de Controle .....	27
2.3.2 Válvula Solenóide .....	28
<b>3. IMPLEMENTAÇÃO (HARDWARE E SOFTWARE).....</b>	<b>31</b>

3.1	Sistema.....	31
3.2	Apresentação do Circuito.....	32
3.3	Microcontrolador.....	34
3.4	Comando do Microcontrolador.....	36
3.5	Sensor de Umidade.....	37
3.6	Sensor LM35.....	40
3.7	Sensor de Fluxo e Fluido.....	42
3.8	Válvula EVA 15.....	45
3.9	LCD.....	47
<b>4.</b>	<b>TESTES E RESULTADOS OBITIDOS .....</b>	<b>49</b>
4.1	Teste do Sensor de Umidade.....	49
4.2	Teste do Sensor de Temperatura.....	50
4.3	Teste do Sensor de Fluxo.....	50
4.3.1	Teste Isolado.....	50
4.3.2	Teste com o Sistema.....	53
4.4	Teste do Sistema.....	54
<b>5.</b>	<b>CONSIDERAÇÕES FINAIS .....</b>	<b>57</b>
<b>6.</b>	<b>REFERÊNCIAS DE PESQUISAS .....</b>	<b>58</b>
<b>7.</b>	<b>APÊNDICES.....</b>	<b>62</b>
Apêndice A	Código Fonte do Projeto (sistema_irrigacao.c).....	62
Apêndice B	Biblioteca do LCD (lcd-v2.0.h).....	72
<b>8.</b>	<b>ANEXOS.....</b>	<b>79</b>
Anexo 1	Pinagem detalhada do PIC16F877A.....	79
Anexo 3	Sensor de Fluxo de Fluido.....	82
Anexo 4	Válvula EVA 15.....	84

## LISTA DE FIGURAS

Figura 2.1 – Arquitetura Harvard .....	19
Figura 2.2 – Diagrama do PIC16F877A .....	20
Figura 2.3 – Ilustração das formas de energia em um sensor.....	21
Figura 2.4 – Ilustração da leitura da posição de um objeto por um <i>encoder</i> incremental.....	22
Figura 2.5 – Ilustração da variação de uma grandeza física de um sensor analógico	22
Figura 2.6 – Medidor Watermark Meter.....	25
Figura 3.1 – Fluxograma do protótipo .....	32
Figura 3.2 – Simulação Projeto ISIS .....	33
Figura 3.3 – Pinagem do PIC16F877A.....	34
Figura 3.4 – Sensor Watermark (Modelo 200SS).....	37
Figura 3.5 – Sensor Watermark (Modelo 200SS-V) .....	37
Figura 3.6 – Simulação do sensor de umidade (ISIS) .....	39
Figura 3.7 – Sensor LM35.....	40
Figura 3.8 – Simulador do Sensor LM35 (ISIS).....	41
Figura 3.9 – Sensor de Fluxo de Fluídos.....	42
Figura 3.10 – Conector do sensor de Fluxo de Fluídos.....	42
Figura 3.11 – Simulador do sensor YF-G21-8(ISIS).....	44
Figura 3.12 (a) – Foto visão superior da válvula EVA 15. ....	45

Figura 3.12 (b) – Foto visão lateral da válvula EVA 15 .....	45
Figura 3.13 – Simulador da válvula (ISIS) .....	45
Figura 3.14 – Placa de circuito com relé .....	46
Figura 4.1 – Teste do sensor de umidade com um multímetro e uma fonte de tensão de 5V. ....	49
Figura 4.2 – Osciloscópio emitindo um sinal constante de 0,5V, com uma fonte de tensão de 5V e o sensor de fluxo de fluído. ....	51
Figura 4.3 – Osciloscópio com medida de onda quadrada de 4,5V .....	52
Figura 4.4 – Sensor de fluxo conectado ao osciloscópio e à fonte de tensão .....	52
Figura 4.5 – Medida no LCD do fluxo de água .....	53
Figura 4.6 – Sistema completo com balde de água .....	54
Figura 4.7 – Protótipo montado (1).....	55
Figura 4.8 – Protótipo montado (2).....	56

# 1. INTRODUÇÃO

## 1.1 Consumo de água

Neste capítulo é abordada a questão do uso da água nos centros urbanos. A relevância do enfrentamento desta matéria está diretamente ligada aos altos custos no processamento da água potável.

De toda água doce consumida no Brasil, 80% (oitenta por cento) é destinada à agropecuária e 13% (treze por cento) às residências, sendo 11% (onze por cento) consumida na área urbana. ([http://www.cultivando.com.br/saude\\_meio\\_ambiente\\_agua\\_reduzindo\\_o\\_consumo.html](http://www.cultivando.com.br/saude_meio_ambiente_agua_reduzindo_o_consumo.html))

Os estudos indicam que, enquanto o volume de água destinado à irrigação das grandes lavouras é bem maior que o consumo urbano, nas cidades o custo se eleva, visto que a água geralmente utilizada na manutenção de jardins é potável.

Outro índice que também estarrece é a notícia de que 59% da água doce são desperdiçadas por conta da metodologia arcaica utilizada no Brasil para torna-la própria para consumo humano.

Isso eleva sobremaneira os custos no tratamento da água, revelando maior comprometimento das verbas públicas e, conseqüentemente, maiores encargos tributários para o cidadão.

Nesse sentido, em que pese o maior consumo na agropecuária em relação ao consumo residencial urbano, o fato é que, economicamente, o consumo urbano merece igual atenção, uma vez que é qualitativamente mais oneroso.

Confira-se o comentário:

*Devemos lembrar que o tratamento de água nas ETA's (Estações de Tratamento de Água) é um processo difícil e muito caro, pesando no orçamento público. A redução de gastos residenciais reduz os gastos públicos, reduzindo os impostos, além de ajudar no orçamento doméstico. ([http://www.cultivando.com.br/saude\\_meio\\_ambiente\\_agua\\_reduzindo\\_o\\_consumo.html](http://www.cultivando.com.br/saude_meio_ambiente_agua_reduzindo_o_consumo.html) Acessado em: 17 set. 2012.)*

Assim, na apresentação do problema é necessário abordar os seguintes tópicos: (1.2) água doce do planeta; (1.3) métodos convencionais de irrigação; e (1.4) vegetação.

## **1.2 Água doce do Planeta**

A escassez de água doce no mundo está cada vez maior, o que gera a necessidade de estudos e pesquisas relacionados à otimização de seu consumo.

“De toda a água do planeta apenas 3% é água doce e, destes, somente 0,03% estão disponíveis para nós...” <[http://www.jardimdasideias.com.br/505-economia\\_de\\_agua\\_no\\_jardim](http://www.jardimdasideias.com.br/505-economia_de_agua_no_jardim)>. Acessado em: 01 set de 2012. O mesmo autor também cita o quão complicado e de custo elevado é o processo de tratamento de água, sendo que o método utilizado no Brasil favorece um desperdício de 41% de toda água tratada, fazendo uma comparação com outros países como Estados Unidos (EUA), que é de 12%, e França, em que o desperdício é de apenas 9%.

Utilizando Brasília como campo de abrangência territorial de observação, o presente projeto busca a comprovação de que a utilização de um sistema de irrigação monitorado por meio de medidores de umidade do solo podem gerar economia e maior eficiência na irrigação dos imensos espaços públicos da capital federal.

A escolha dessa cidade como objeto de análise deve-se, especialmente, pelos longos períodos de estiagem enfrentados nessa região do cerrado, bem como a combinação do grande número de áreas verdes que embelezam os espaços públicos.

As situações de extrema seca em relação ao elevado consumo de água potável na conservação dos jardins possibilita melhor visualização dos resultados da pesquisa.

## **1.3 Métodos convencionais de irrigação**

Os métodos convencionais de irrigação sugerem que o sistema seja regulado de modo a atender às alterações climáticas com base, apenas, nas estações do ano.

Ora, em se tratando de Brasil, a premissa utilizada por esses métodos é falha, uma vez que é característico do clima tropical brasileiro, certa irregularidade nas temperaturas e volume de chuvas nas diferentes estações do ano. Sendo estas, também, bastante irregulares.

Nesse sentido, a mera adoção dos fatores atmosféricos e de previsão do volume pluviométrico para determinado período do ano é absolutamente ineficaz, podendo gerar encharcamento do solo, danificando a vegetação e desperdiçando a valiosa água potável, bem como pode causar seu ressecamento excessivo em outras ocasiões.

Outro método, como a utilização de timer isoladamente, dentre outros, tendem a causar idênticos prejuízos.

#### **1.4 Vegetação**

A vegetação utilizada no embelezamento dos centros urbanos pode, e deve ter outras funções, além do mero aformoseamento dos espaços públicos.

A escolha ideal da vegetação que será incluída nesses ambientes deve estar relacionada à pesquisa de espécies que melhor suportem a poluição, podendo, inclusive, auxiliar na descontaminação do meio ambiente, e, também, deve levar em conta aquelas que exijam menores cuidados na sua manutenção, seja com água, fertilizantes, podagem, etc. Tudo visando o racionamento dos gastos públicos.

Esses fatores, aqui tratados como externalidades, não serão aprofundados no âmbito deste projeto, mas podem servir como sugestão para o desenvolvimento de outras iniciativas. Isso porque é um tema correlato à área de botânica e paisagismo, escapando à matéria ora abordada.

No entanto, há trabalhos de pesquisas que esclarecem, em conclusão, que a densidade do solo também é fator que influencia na correta irrigação das coberturas vegetais.

Densidade do solo está associada à estrutura, à densidade de partícula e à porosidade do solo. Para início de trabalho é necessário explicar como variação da densidade e da umidade do solo estão ligados à irrigação de um ambiente.

A Densidade do solo ( $d_s$ ) é percebida por meio da medição da massa de uma unidade de volume de solo incluindo o espaço poroso. Quando relacionada com a densidade da partícula ( $d_p$ ) permite calcular a porosidade do solo. (Carvalho, João Paulo).

O Experimento realizado pelos pesquisadores Hugo Roldi Guariz, Wesley Augusto Campanharo, Marcelo Henrique Savoldi Picoli, Roberto Avelino Cecílio e Maycon Patrício de Hollanda, divulgado no artigo “Variação da umidade e da densidade do solo sob diferentes coberturas vegetais”, demonstra que o nível de desenvolvimento da vegetação tem relação direta com a manutenção dos níveis de umidade do solo, no entanto, não existe uma forma de calcular o volume de água necessário para manter a umidade do solo com base apenas na quantidade de vegetação da área estudada.

Nesse sentido, a instalação de sensores de umidade conectados ao sistema de irrigação garante maior eficiência na hidratação ideal do solo.

## **2. REFERENCIAL TEÓRICO E TECNOLÓGICO**

### **2.1 Microcontroladores**

O microprocessador – usualmente conhecido por CPU (Unidade Central de Processamento) – consiste em uma parte de um sistema que processa os dados, busca instruções na memória, decodifica-as e as executa. Para sua efetiva finalidade, deve ser composto de chips adicionais, como por exemplo, no caso de um sistema que possa ser utilizado para controle, necessita de memória para armazenamento de dados e portas de entrada/saída para permitir comunicação com o mundo externo e receber sinais.

O microcontrolador é a integração do microprocessador com interfaces de entrada/saída (I/O) e outros periféricos, como temporizadores, em um único chip.

Um microcontrolador geral tem pinos para conexões externas de entradas e saídas, alimentação, clock e sinais de controle. Os pinos para as entradas e saídas são agrupados em unidades denominadas portas. Geralmente estas portas têm oito linhas para serem capazes de transferir palavras de 8 bits. Duas portas podem ser usadas para palavras de 16 bits, uma para transmitir os 8 bits inferiores e a outra para os 8 bits superiores. As portas podem ser apenas de entrada, apenas de saída ou programáveis como entrada ou saída.

Os microcontroladores possuem uma quantidade de RAM e ROM limitadas e são muito usadas em sistemas de controle embarcados. Um sistema microprocessado com chips separados de memória e I/O é mais adequado para o processamento de informações em um sistema de computador.

### 2.1.1 Microcontroladores da Família PIC

A empresa Microchip™ criou o termo PIC (Peripheral Interface Controller) para denominar microcontroladores de pastilha única, que são projetados com a arquitetura Harvard.

Esta arquitetura permite que as instruções sejam buscadas em uma memória de programa através de barramentos distintos dos usados para acessar variáveis (Figura 2.1). Fazendo com que as instruções possam ser buscadas a cada ciclo sem espera, sendo a execução realizada no próximo ciclo de busca. Permitindo o PIC ter velocidades de execução maiores para uma dada frequência de clock. (W.Bolton, 2010)



Figura 2.1 – Arquitetura Harvard

Fonte: <http://sistemasuniban.blogspot.com.br/2010/04/arquiterura-von-neumann-vs-harvard.html>, Acesso em 11/11/2012.

O microcontrolador utilizado no projeto foi o PIC16F877A, a Figura 2.2 ilustra o diagrama em blocos do mesmo.

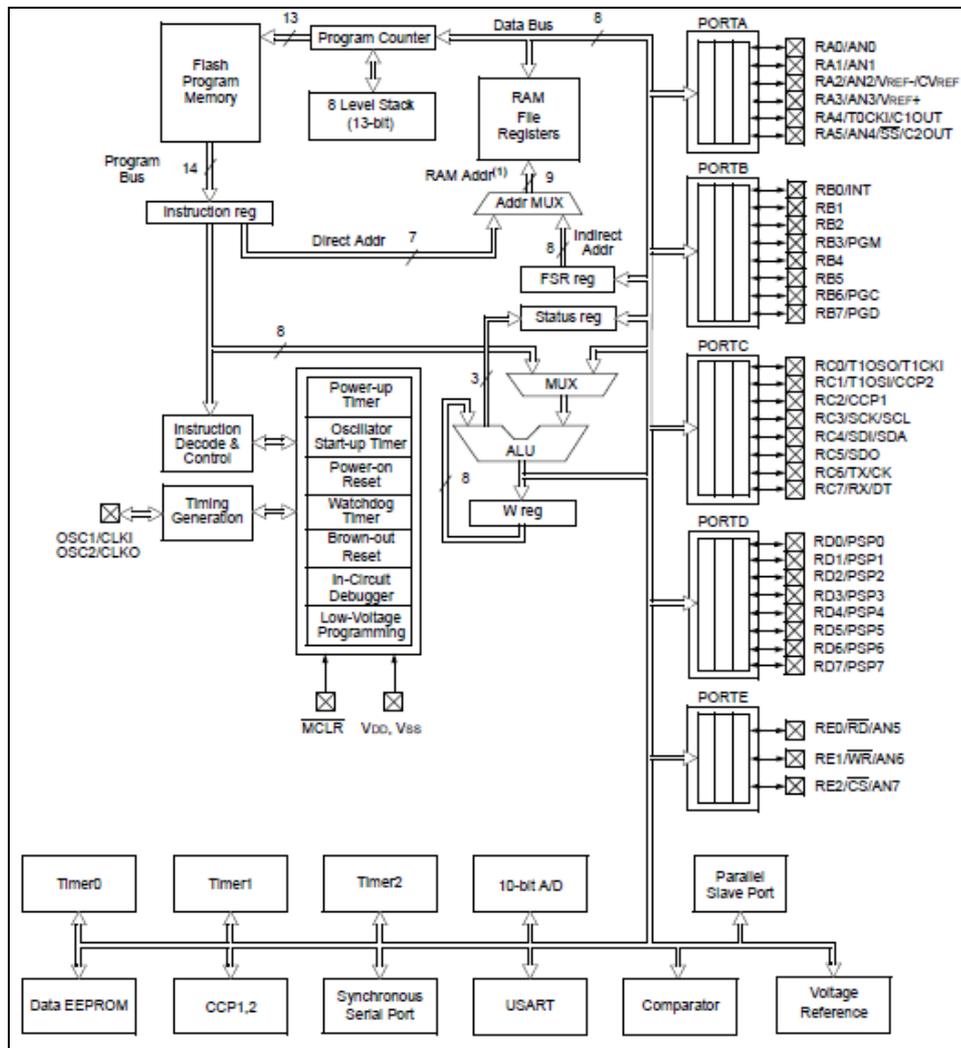


Figura 2.2 – Diagrama do PIC16F877A

Fonte: PIC16F87XA Data Sheet, disponível em

<http://ww1.microchip.com/downloads/en/DeviceDoc/39582b.pdf>, Acesso em 01/10/2012.

Cada pino possui uma característica e função no microcontrolador. Os pinos utilizados no projeto estarão com suas descrições mais bem estabelecidas no capítulo 3.

## 2.2 Sensores

Sensor, de acordo com Daniel Thomazini e Pedro Urbano Braga de Albuquerque, é um dispositivo sensível a alguma forma de energia do ambiente que pode relacionar informações sobre uma grandeza que precisa ser medida, como: temperatura, pressão, velocidade, corrente, aceleração, posição, etc. A Figura 2.3 ilustra as formas de energia de um sensor.

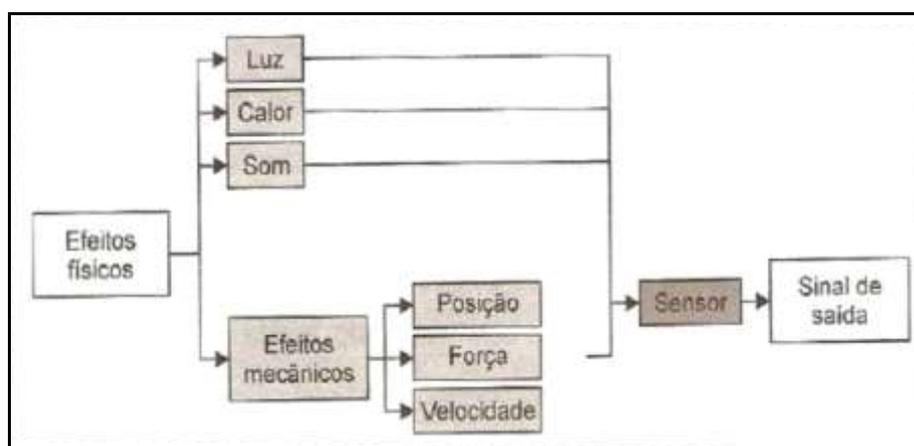


Figura 2.3 – Ilustração das formas de energia em um sensor  
 Fonte: Sensores Industriais, Thomazini, 2007.

Em um sistema com mais de um sensor, o elemento que estiver em contato com o processo será chamado de *elemento sensor primário*, outros de *elementos sensores secundários*. Já em casos que o sensor pode estar protegido por algum outro dispositivo, para não ficar em contato direto com o processo, os nomes alternativos são: *elemento transdutor*, *elemento primário*, *detector*, *probe*, *pickup* ou *pickoff*. (Ribeiro, 2002).

Dentre as classificações de sensor, destacam-se a de sensor digital e sensor analógico.

Sensor digital pode apenas assumir dois valores no seu sinal de saída ao longo do tempo que podem ser interpretado como zero (0) e um (1). Naturalmente grandezas físicas não assumem esses valores, porém eles são representados assim pelo sistema quando convertidos pelo circuito eletrônico do transdutor.

A Figura 2.4 ilustra a variação da posição de um objeto lida por um *encoder* incremental. Sensores ópticos, sensores de presença, entre outros, são exemplos de sensores digitais.

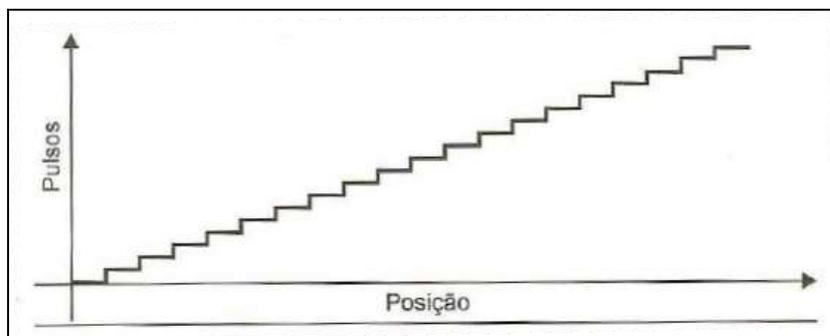


Figura 2.4 – Ilustração da leitura da posição de um objeto por um *encoder* incremental  
Fonte: Sensores Industriais, Thomazini, 2007. (p19)

Sensores Analógicos, também chamados de sensores de saída linear, são mais complexos que os digitais, porém oferecem muito mais dados e precisão sobre um processo. Esse tipo de sensor admite qualquer valor no seu sinal de saída ao longo do tempo, desde que esteja dentro da sua faixa de operação.

Algumas grandezas físicas que podem assumir qualquer valor ao longo do tempo são: temperatura, pressão, velocidade, umidade, vazão, força, ângulo, distância, toque, luminosidade. Essas variáveis são mensuradas por elementos sensíveis com circuitos eletrônicos não digitais. A Figura 2.5 ilustra a variação de uma grandeza física (temperatura) de forma analógica. (Thomazini, 2007)

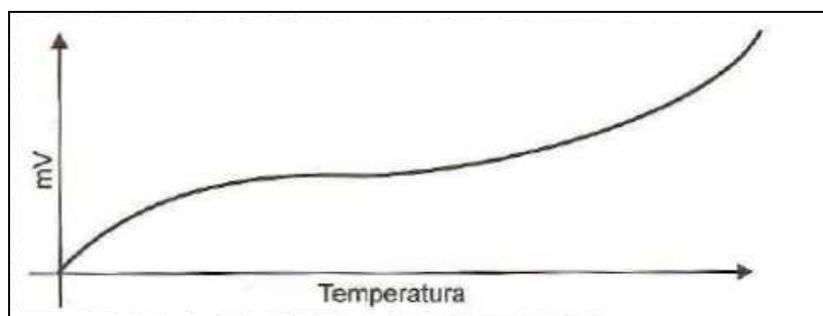


Figura 2.5 – Ilustração da variação de uma grandeza física de um sensor analógico  
Fonte: Sensores Industriais, Thomazini, 2007. (p18)

É possível converter um sinal analógico em um sinal digital ou vice-versa através de conversores analógico-digital ou digital-analógico. Porém estas conversões quando realizadas, podem gerar perda de parte do sinal e realizar pequenas distorções na grandeza realmente medida, além de um conversor mal dimensionado poder tornar um projeto mais lento durante o processo de conversão.

No projeto, por utilizar o microcontrolador PIC16F877A, essa perda de sinal é insignificante, pois o PIC possui uma resolução de conversão do valor 5V (Volts) com um ADC de 10 Bits igual 4,88mV. Esse valor, também mostra o quanto o microcontrolador possui um alto valor de precisão.

### 2.2.1 Sensor de Umidade

Existem diversos modos utilizados para determinar a umidade do solo, no entanto não há se falar em um melhor, especificadamente, a escolha deverá levar em conta os objetivos desejados pelo pesquisador e/ou produtor, do projeto ou produto, do nível de precisão desejada, entre outros fatores que possam limitar sua escolha. Klar (1988, apud Coelho 2008) relata que todos os métodos utilizados na determinação da umidade do solo apresentam limitações: ou quanto à precisão, ou por serem dispendiosos, ou excessivamente morosos. Libradi (1999, apud Coelho 2008) confirma as limitações citadas e ressalta o elevado nível de complexidade de alguns métodos.

Estes autores classificam os métodos em diretos e indiretos; os diretos permitem a determinação direta da umidade do solo, destacando-se o método da estufa e o de pesagem. Já os indiretos baseiam-se em propriedades químicas ou físicas do solo, conforme a quantidade de água presente, isto é, tomam como base a moderação de nêutrons, a resistência à passagem de uma corrente elétrica, a constante dielétrica e a tensão da água no solo. Alguns destes métodos necessitam de uma calibração para serem utilizados.

#### 2.2.1.1 Método Direto

O método direto mais utilizado para a determinação da umidade do solo é o gravimétrico. Este método consiste basicamente na coleta a campo de um volume de solo conhecido e na determinação das suas massas úmida e seca. Esse método,

apesar de ser o mais preciso, apresenta limitação por necessitar de mais de 24h para fornecer o resultado da amostragem, além da necessidade de se utilizar balança e estufa.

#### 2.2.1.2 Método Indireto

Há vários métodos indiretos para determinar a umidade do solo, como por exemplo: método de resistência elétrica, método tensiométrico, método nuclear ou sonda de nêutrons, método químico, método de capacitância elétrica, método do TDR, entre outros.

Neste projeto foi utilizado o método de resistência elétrica, que consiste no princípio de que o solo úmido oferece menor resistência à passagem da corrente elétrica do que o solo seco. Os principais métodos que utilizam a resistência elétrica são:

- **Método de Bouyoucos – Blocos de Gesso**

Este método foi desenvolvido por Bouyoucos e Mick (1940), é baseado na medição da resistência elétrica entre dois eletrodos inseridos em um bloco, geralmente de gesso. A resistência elétrica é medida por uma ponte de Wheatstone (corrente alternada), dando uma indicação indireta de umidade do bloco e, portanto, do conteúdo de água no solo.

Os blocos de gesso, quando enterrados no solo, absorvem umidade ou a perdem para o solo, até que a solução dentro dos blocos entre em equilíbrio com a água contida no solo.

- **Método de Colman – Nylon**

Baseado no mesmo princípio do método de Boyoucos. Porém, o bloco onde são inseridos os eletrodos é de fibra de vidro, envolvida por duas chapas de metal “monel” perfuradas. O bloco é composto por “termistor”, permitindo, assim, determinar a temperatura da célula e fazer a correção das leituras, em função da temperatura. As células devem ser calibradas para cada tipo de solo, o que dá maior

precisão no cálculo da umidade do solo. Essa calibração pode ser em função do teor de água no solo ou, preferencialmente, da sua tensão.

Este método é mais sensível que o de Boyoucos para maiores teores de água no solo, porém é mais suscetível à concentração salina do solo.

- **Sensor Watermark**

O sensor Watermark é formado de dois eletrodos embutidos num molde, seu funcionamento utiliza a resistência elétrica para medir o nível de umidade do solo e quando a eletricidade é aplicada, o material do molde mostra uma resistência elétrica que muda com a variação do volume de umidade do molde. Esta mudança na resistência é lida e convertida em potencial de água do solo.

Este sensor necessitará de uma calibração para ser utilizado, se o projetista não usar seu aparelho de medição específico o Watermark Meter (Figura 2.6).



Figura 2.6 – Medidor Watermark Meter

Fonte:Watermark/Irrrometer. Disponível em:< <http://www.irrometer.com/sensors.html#meter>>. Acesso em: 22/10/2012

A calibração é realizada com o uso de tensiômetros e sensores. Nesse processo, são realizadas comparações das várias medidas dos tensiômetros e dos sensores ao longo do processo de saturação do solo e de solo seco. De posse dos dados é possível realizar uma regressão para encontrar a fórmula que descreve a tensão de saída no sensor em função da pressão medida no tensiômetro.

### 2.2.2 Sensor de Temperatura

Sensor de temperatura tem a função de medir a grandeza física temperatura. Existem, no mercado, vários tipos de sensores de temperaturas, dentre os quais se destacam os *termisores*, *termopares*, *termorresistores*, *par bimetálico*, *sensores eletrônicos* e *pirômetros*.

O sensor utilizado é o sensor da série LM35.

### 2.2.3 Sensor de Fluxo de Fluídos

O sensor utilizado no projeto é um sensor de vazão magnético, que utiliza a teoria da bobina de Pick-up, porém, no lugar de uma bobina de Pick-up é utilizado sensor de efeito Hall.

#### 2.2.3.1 Bobina de PICK-UP

*Uma bobina do tipo pick-up é colocada em um tubo de latão, aço inox ou plástico que está dentro de uma tubulação da qual se quer medir a vazão. Na extremidade do tubo existe um rotor que gira a uma velocidade proporcional ao fluxo do líquido. Devido à proximidade do rotor com a bobina, cada vez que o rotor corta as linhas de fluxo magnético do imã do pick-up, haverá indução de um pulso de tensão na bobina, devido à variação do fluxo. (Thomazini, 2007,p 156)*

## 2.3 Válvulas

São usadas em sistemas hidráulicos e pneumáticos para guiar e regular a vazão de um fluido, geralmente são usadas em tubulações, entradas e saídas de vasos e tanques em várias aplicações diferentes.

Sistemas pneumáticos são usados para transferir sinais de controle dos elementos finais. Possui vantagem em poder acionar grandes válvulas e outros dispositivos de alta potência e, com isso, mover cargas significativas. A desvantagem está na compressibilidade do ar.

Sistemas hidráulicos são usados em dispositivos de controle de potência mais robustos, porém mais caros e mais perigosos com relação a vazamentos de óleo que não existem nos vazamentos de ar.

Existem apenas dois tipos de válvulas:

- **Posições finitas:** possuem a ação para permitir ou bloquear a passagem de um fluido, com isso, são usadas para ligar ou desligar atuadores. (Bolton, 2010)
- **Posições infinitas:** controlam a vazão em qualquer valor entre a posição totalmente aberta e a totalmente fechada, e são usadas para controlar as forças nos atuadores ou a vazão de um fluido em uma situação de controle de processo. (Bolton, 2010)

### 2.3.1 Válvula de Controle

Válvula de controle é um equipamento usado para regular a vazão de fluidos em tubulações e máquinas, recebendo o sinal de saída do controlador e atuando na variável manipulada. O controle pode ser automático ou manual. O controle manual pode ser remoto ou local. A válvula de controle abre e fecha a passagem interna do fluido, de conformidade com um sinal de controle. Quando o sinal de controle é proveniente de um controlador, tem-se o controle automático da válvula. Quando o sinal de controle é gerado manualmente pelo operador de processo, através de uma estação manual de controle, tem-se o controle manual remoto.

Válvula de controle é usada para regular a vazão de fluidos em tubulações e máquinas, recebendo o sinal de saída do controlador e atuando na variável manipulada. Ela é dividida em três tipos de controle:

- **Controle Manual->** pode ser remoto ou local. A válvula de controle abre e fecha a passagem interna do fluido, de conformidade com um sinal de controle.
- **Controle Automático->** é um controle proveniente de um sinal de controle de um controlador.
- Quando o sinal de controle é gerado manualmente pelo operador de processo, através de uma estação manual de controle, tem-se o controle manual remoto.

O controle pode ser feito de modo contínuo, onde a válvula pode assumir infinitas posições entre totalmente aberta ou fechada, ou liga-desliga, em que apenas dois estados são possíveis de forma estável.

A válvula de controle possui vários tipos de dispositivos que são utilizados para se obter determinadas adaptações ao sistema de controle utilizado ou sofisticações quanto dos mesmos, são eles os acessórios, os mais comuns são: posicionadores, *Boosters* pneumáticos, Válvulas Solenóides (que serão comentadas mais adiante), Chaves Indicadoras de Posição, entre outros. (*Instrumentação - Elementos Finais de Controle*. Departamento Regional do Espírito Santo. SENAI – ES, 1999)

### 2.3.2 Válvula Solenóide

A válvula solenóide é a combinação de duas unidades funcionais, que são: um corpo de válvula e um pacote eletromagnético.

O pacote eletromagnético é constituído por um solenoide e seu correspondente núcleo móvel. O solenóide é uma bobina de fio isolado, energizada eletricamente para produzir um campo magnético no seu interior, que provoca um movimento mecânico em um núcleo ferromagnético, colocado no centro do campo. Quando a bobina está energizada, o núcleo está numa posição, quando desenergizada, o núcleo está em outra posição, provocando assim o sistema de abertura e fechamento.

No corpo de válvula podem conter um ou vários orifícios de entrada, passagem e saída. Esses orifícios permitem a passagem de um fluido ou não quando a haste é acionada pela força da bobina. Esta força faz com que o pino seja puxado para o centro da bobina, permitindo a passagem do fluido. O processo inverso, o de fechamento, ocorre graças a perda de energia, pois o pino exerce uma força graças ao seu peso e da mola que tem instalada.

O corpo possui um dispositivo que permite que a válvula solenóide seja aberta ou fechada pelo movimento do núcleo, que é acionado na solenóide quando a bobina é energizada e é usada para controlar a vazão de fluidos, principalmente no modo liga-desliga. Existem válvulas do tipo normalmente fechadas, que se abrem

quando uma corrente é aplicada e se fecham quando a corrente é cortada e normalmente aberta, que trabalham de forma inversa.

De acordo com as empresas ASCO, Paker e Jeffeson produtoras de válvulas solenoides. Os tipos de válvulas solenóides, mais comuns são:

- Válvulas de 2 vias: possuem uma conexão de entrada e uma de saída, abrindo ou fechando o orifício de passagem de fluido em função de um comando elétrico, disponível em duas versões:
  - **Normalmente fechada** – a válvula permanece fechada quando desenergizada e se abre quando energizada.
  - **Normalmente aberta** – a válvula permanece fechada quando energizada e se abre quando desenergizada.
- Válvulas de 3 vias: possui três conexões e dois orifícios. Um dos orifícios sempre estará fechado enquanto o outro estiver aberto. Elas são comumente usadas para aplicar pressão ou dar escape de pressão a uma válvula tip diafragma, atuador pneumático, cilindro de simples ação e selecionar ou divergir fluxos.

Também estão disponíveis nas versões:

- **Normalmente fechada** – a válvula permanece fechada quando desenergizada e se abre quando energizada.
  - **Normalmente aberta** – a válvula permanece fechada quando energizada e se abre quando desenergizada.
  - **Universal** – Pode se comportar como Normalmente Fechada, Normalmente Aberta ou desviadora de fluxo. dependendo de onde se aplica a pressão de entrada.
- Válvulas de 4 ou 5 vias: podem ter 2 ou 3 posições; no ultimo caso com uma posição estável e duas instáveis, com uma bobina para cada posição.

São geralmente utilizadas para o comando de cilindros dupla ação, podendo ser simples ou duplo solenoide. Apresentam cinco conexões: uma de pressão, duas de utilização e duas de exaustão.

### **3. IMPLEMENTAÇÃO (HARDWARE E SOFTWARE)**

#### **3.1 Sistema**

O projeto tem como objetivo promover uma maior economia de água e proporcionar ao usuário um jardim sempre em excelentes condições, sem que seja necessário a rega em horários desnecessários ou quando o solo já estiver suficientemente úmido.

Neste Capítulo é apresentado o protótipo do sistema de irrigação automatizado. Os principais itens que o protótipo utiliza são: o PIC16F877A, o sensor de umidade para solo (Waltermark), o sensor de temperatura (LM35), o sensor de fluxo de fluídos e a válvula solenóide. Também são apresentados trechos da programação realizada em linguagem C de cada item com suas respectivas funções. O código de programação completo, do protótipo está no APÊNDICE A (página 62).

O PIC16F877A realiza uma leitura contínua dos sensores e uma atualização dos valores no LCD. Após a atualização, o microcontrolador realizará uma sequência de conferências de valores: de solo seco, de temperatura e de solo saturado para saber se a válvula será aberta ou fechada. Quando aberta o medidor de consumo de água é acionado, após a medida de água ou ao fechar a válvula o programa é reiniciado para continuar todo o processo.

Para a representação da explicação realizada anteriormente foi criado um fluxograma do protótipo, apresentado na Figura 3.1.

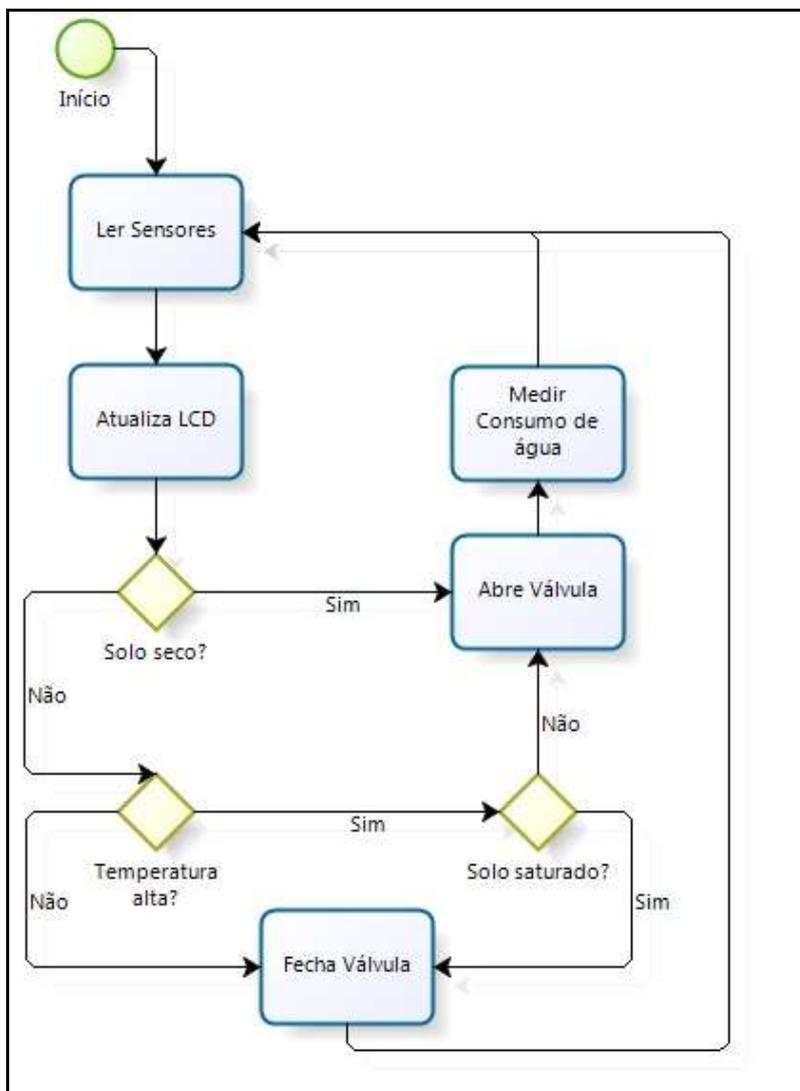


Figura 3.1 – Fluxograma do protótipo  
Fonte: (Autor)

### 3.2 Apresentação do Circuito

Antes da montagem de todo o circuito do projeto, foi criado um protótipo do projeto, que foi montado no software Proteus 7.6, que é dividido em duas partes a ISIS onde foi realizada uma simulação da programação, e a ARES que foi utilizada para desenho da placa do projeto.

A simulação do projeto, que foi desenhado no PROTEUS ISIS, está na Figura 3.2.

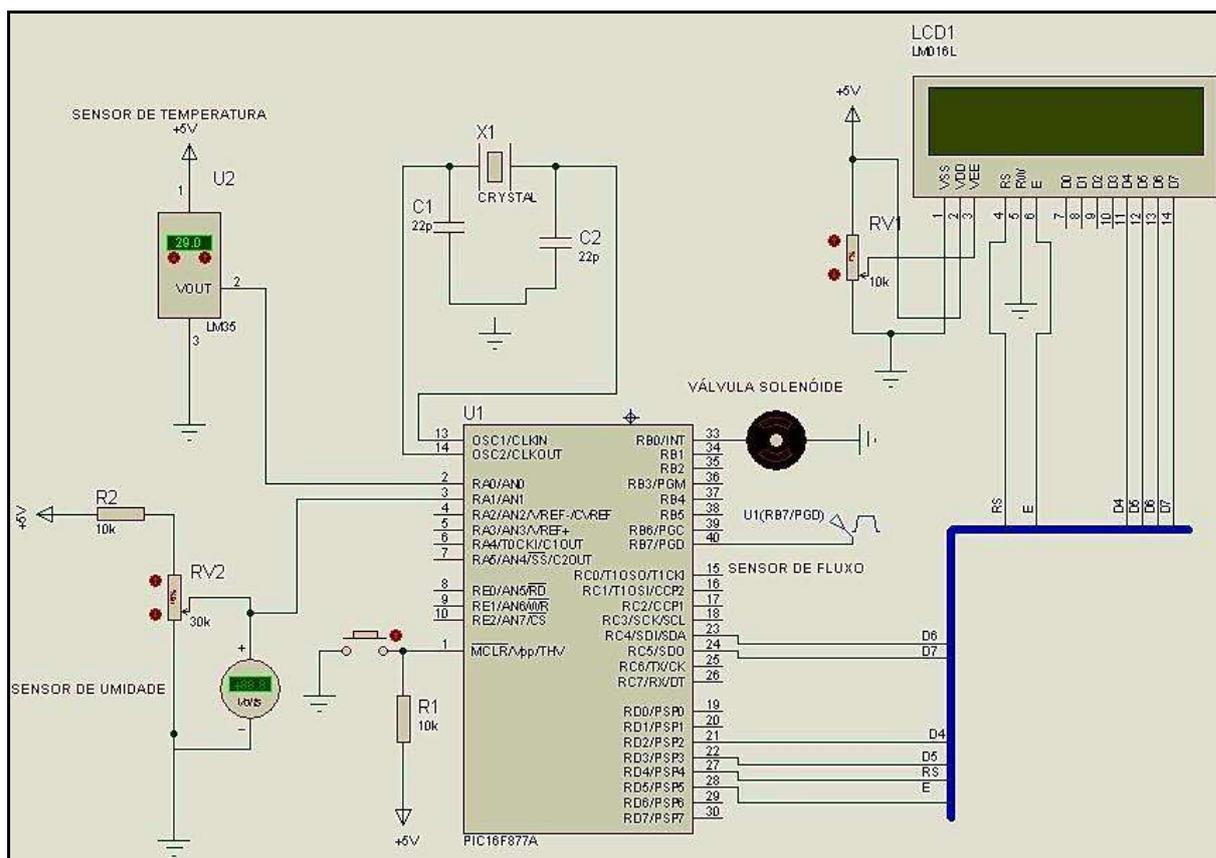


Figura 3.2 – Simulação Projeto ISIS  
Fonte: (Autor)

Na simulação do circuito no ISIS também é realizado a simulação da programação, esse teste de programação é utilizado para testar se todo o sistema poderá estar funcionando, mas não oferece garantia de 100% de que todo projeto estará funcionando quando estiver montado, pois no Proteus ele será apenas uma simulação.

### 3.3 Microcontrolador

Como citado anteriormente, o microcontrolador utilizado no projeto é o PIC16F877A a pinagem que foi utilizada no projeto está representada na Figura 3.3 e estarão descritas logo após.

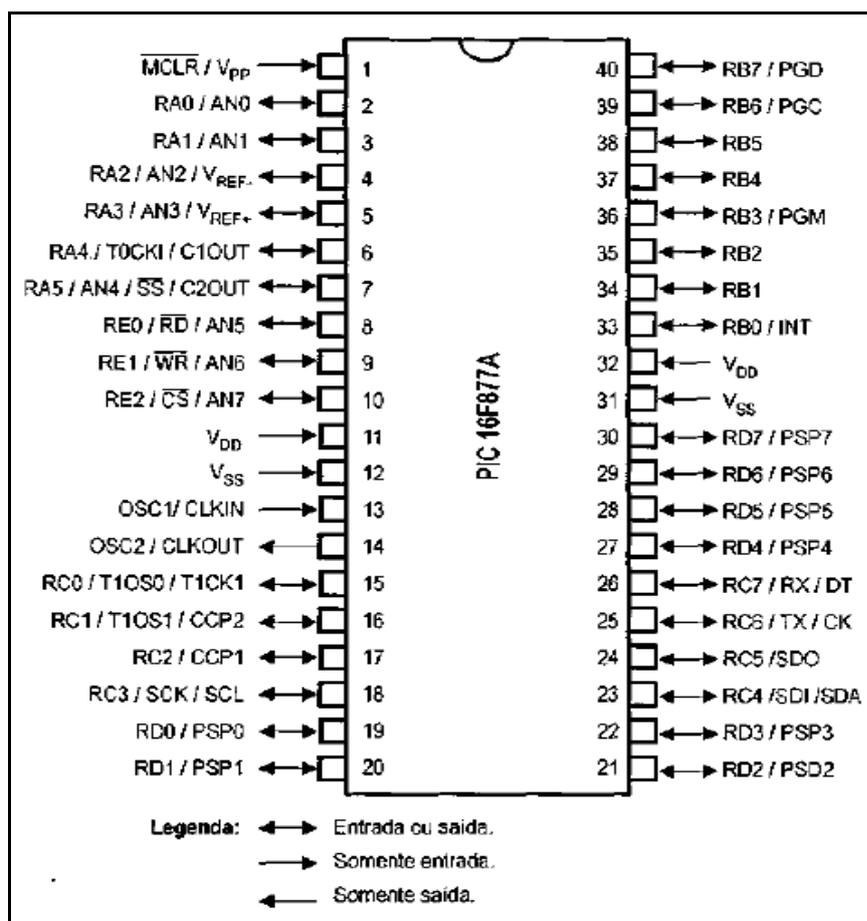


Figura 3.3 – Pinagem do PIC16F877A

Fonte: SOUZA, 2007

Os pinos utilizados são:

PORTA: porta bidirecional de entrada/saída, composto pelos pinos: RA0, RA1, RA2, RA3, RA4 e RA5.

Sendo que os que estão sendo utilizados pelo projeto são:

RA0- Pino de entrada analógica que está sendo usado no Sensor de Temperatura

- **RA1**- Pino de entrada analógica que está sendo usado no Sensor de Umidade do Solo (Waltermark).

PORTB: porta bidirecional de entrada/saída, composto pelos pinos: RB0, RB1, RB2, RB3, RB4, RB5, RB6 e RB7.

Sendo que os que estão sendo utilizados pelo projeto são:

- **RB0**- Pino de saída digital (0 ou 5V) que está sendo usado para válvula solenoide receber o sinal de comando para abertura ou fechamento do fluxo de água.
- **RB7**- Pino de entrada digital configurado para uma interrupção externa proveniente do sensor de fluxo de água. Essa interrupção é o pulso gerado que serve como base de contagem para indicar a quantidade de água que passa pelo sensor em um determinado tempo.

PORTC e PORTD: os dois estão sendo usados no projeto para barramento de dados e controle do LCD. PORTC porta bidirecional de entrada/saída, composto pelos pinos RC0, RC1, RC2, RC3, RC4, RC5, RC6 e RC7. PORTD também bidirecional de entrada/saída, porém digital, composto pelos pinos RD0, RD1, RD2, RD3, RD4, RD5, RD6 e RD7.

Sendo que os pinos que estão sendo utilizados pelo projeto são:

PORTC-

- **RC4 e RC5**: compõem o barramento de dados do LCD (D6 e D7)

PORTD-

- **RD2 e RD3**: compõem o barramento de dados do LCD (D4 e D5);
- **RD4**: compõe a linha de controle do registrador de seleção do LCD (*RS-register select*). Sendo utilizado no modo 4bits de dados;
- **RD5**: compõe a linha de controle de habilitação do LCD (*E- Enable*).

**MCLR (MASTER CLEAR RESET)**: Pino de Reset do microcontrolador. Ativado em nível lógico baixo (0V), resetando o programa na memória do microcontrolador. Este Reset é proporcionado por um botão externo que no caso de travamento do programa o usuário comprime o botão reiniciando o sistema.

OSC1 e OSC2: entrada do Cristal oscilador do microcontrolador. O mesmo cadencia o programa na memória do sistema. No projeto é utilizado um cristal de 4MHz.

Para uma descrição completa da pinagem do PIC16F877 basta conferir o Anexo1 (página 79).

### 3.4 Comando do Microcontrolador

Como já dito a linguagem de programação utilizada no projeto foi a linguagem C.O programa utilizado para realizar a compilação e geração do arquivo HEX. foi o PIC C Compiler. Segue um trecho do código onde foram apresentadas a biblioteca utilizada, tal como definição do clock utilizado, a resolução do conversor A/D, a biblioteca do módulo LCD, a definição das configurações de TRIS e as configurações necessárias para o compilador.

```
#include <16f877A.h>           //Inclui a biblioteca do PIC utilizado;
#device adc=10                //Define o conversor analógico digital na resolução 10
bits;
#use delay (clock=4000000)    //Cristal oscilador de 4Mhz;
#fuses xt,nowdt,put,nobrownout,nolvp
#use fast_io(B)              //Define que as configurações de TRIS devem ser feitas
manualmente
#include <lcd-v2.0.h>         //Inclui a biblioteca do módulo LCD
```

Para que o protótipo funcionasse após a captação dos valores dos sensores é necessário realizar a conversão A/D, pois os valores que são obtidos são valores analógicos e um microcontrolador só irá processar esses valores quando os mesmos estiverem convertidos para digitas. Com isso, foi separada uma função de conversão analógica e uma para cada sensor para que o mesmo pudesse gerar seus valores.

### 3.5 Sensor de Umidade

O sensor utilizado no protótipo desse projeto é o sensor de resistência elétrica Watermark 200SS (Figura 3.4).



Figura 3.4 – Sensor Watermark (Modelo 200SS)

Fonte: Watermark/Irrrometer. Disponível em:< <http://www.irrometer.com/sensors.html#meter>>. Acesso em: 22/10/2012

Por não ter sido possível calibrar o sensor com o uso de tensiômetros, o modelo utilizado (200SS) não ofereceu medidas confiáveis para a pressão matricial. Para realizar essas medidas foram utilizados os dados de calibração do modelo Watermark 200SS-V (Figura 3.5), que é um modelo aprimorado do 200SS.



Figura 3.5 – Sensor Watermark (Modelo 200SS-V)

Fonte: Watermark/Irrrometer. Disponível em:< <http://www.irrometer.com/sensors.html#meter>>. Acesso em: 22/10/2012

Os dados são disponibilizados pelo fabricante através de uma tabela (Anexo 2), a partir deles foi aplicada a regressão linear para encontrar a equação (1) que descreve o comportamento da voltagem no sensor.

$$y = 0.0117155x - 5 * 10^{(-9)} \quad (1)$$

Onde:

y -> Tensão (em V);

x -> Pressão (em Cb - centibars).

Como o microcontrolador só consegue ler a tensão do sensor, a equação (1) foi reescrita para obter a pressão do solo em função da tensão do mesmo:

$$x = \frac{y - 5 * 10^{(-9)}}{0.0117155} \quad (2)$$

Na programação foram utilizadas 3 faixas de umidade:

```
#define LIM_UMIDO 40           //Limite de solo saturado.
#define LIM_SECO 70           //Limite de solo seco.
#define LIM_QSECO           //Limite de solo quase seco.
```

Para compreensão das faixas de umidade utilizadas, é importante entender que a medida realizada pelo sensor é a da pressão matricial, e que, quanto mais seco o solo, maior o valor medido de tensão e pressão. E que existem tabelas de controle de pressão (cb – centibars ou kPa – kilo Pascal) informando qual a medida de pressão necessária para o tipo de cultivo que o pesquisador ou produtor estará realizando. Por esse motivo o valor do limite de solo seco é maior que o limite de solo saturado de umidade.

Os limites de faixa de umidade que foram utilizados para definir se a válvula deve ser aberta ou fechada e os intervalos de atualização do sensor de umidade são:

- **LIM\_UMIDO** é o limite superior da faixa em que o solo está excessivamente úmido.
- **LIM\_SECO** é o limite inferior da faixa em que o solo está excessivamente seco e a grama pode começar a secar.
- **LIM\_QSECO** é o limite inferior da faixa em que o solo ainda não está excessivamente seco, mas está próximo e tem seu valor atualizado com a mesma frequência que teria se estivesse excessivamente seco.

Segue trecho do código na função de atualização da medida de umidade em que o tempo entre as medidas de umidade é definido.

```

UMIDADE = (MEDIDA+(5/1000000000))/(0.0117155); // Aplicando Equação (2)
if (UMIDADE >= LIM_QSECO){
    min_leitura_umidade = TIMER_SECO;
}else{
    min_leitura_umidade = TIMER_UMIDO;
}

```

Na etapa de simulação, o sensor foi representado por um resistor ajustável e um medidor de tensão para que fosse possível testar o comportamento conforme o solo seca. A Figura 3.6 ilustra a simulação do sensor de umidade.

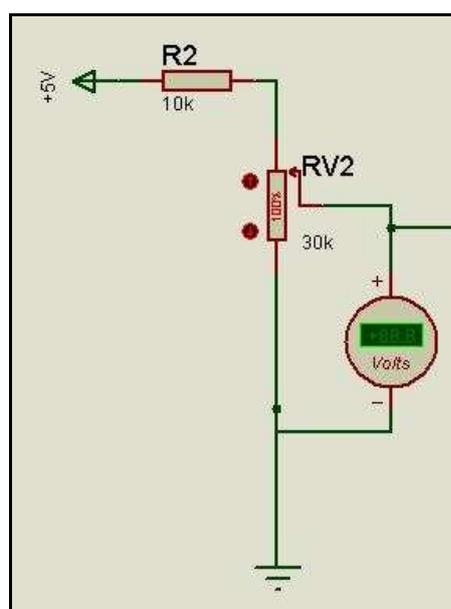


Figura 3.6 – Simulação do sensor de umidade (ISIS)  
Fonte: (Autor)

### 3.6 Sensor LM35

O sensor da série LM35, é um sensor de temperatura de circuito integrado, foi lançado pela National Semiconductor (equivalente ao da Texas Instruments, STP35A) para oferecer alta precisão por contar circuitos linearizados.

Ele possui saída de tensão linear relativa à temperatura em que ele se encontra no momento em que for alimentado (+Vs) por uma tensão de 4-20Vdc e GND, tendo em sua saída (Vout) um sinal de 10mV/°C.

O LM35 não necessita de uma calibração externa ou “trimming” para fornecer, valores de temperatura com variações de  $\frac{1}{4}^{\circ}\text{C}$  ou até mesmo  $\frac{3}{4}^{\circ}\text{C}$  dentro da faixa de temperatura de  $-55^{\circ}\text{C}$  à  $150^{\circ}\text{C}$ .

Dependendo do tipo de alimentação (simples ou assimétrica) que for estabelecida para o sensor, o sinal de saída irá alterar, mas independentemente da alteração, a saída continuará sendo de 10 mV/°C. Ele drena apenas 60µA para estas alimentações, sendo assim seu auto-aquecimento é de aproximadamente 0.1°C ao ar livre.

Composto por três pinos, conforme Figura 3.7:

- Alimentação (+Vs) - Varia de 4V-20V;
- Saída (Vout) – 0 mV/°C + 10 mV/°C;
- GND.

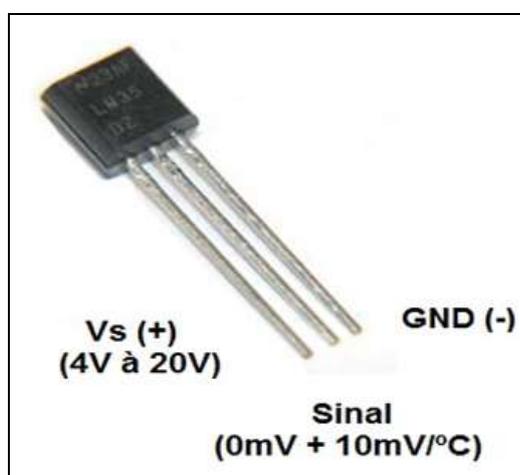


Figura 3.7 – Sensor LM35

Fonte: Disponível em <http://2.bp.blogspot.com/-kXsmv8beH7U/TIKOn6miDCI/AAAAAAAAANG/-6YY2il1pQo/s1600/2.png>, Acesso em 23/10/2012

Neste projeto o sensor de temperatura é responsável por controlar a temperatura do solo, para que o mesmo não atinja uma temperatura acima de 30°C, este limite foi estabelecido para a proteção do sensor de umidade, pois o mesmo pode ser danificado se ultrapassar essa temperatura. O código que está definindo o limite de temperatura e o trecho com o controle é:

```
#define LIM_TEMP 30          //Limite de temperatura.  
  
if (CELSIUS > LIM_TEMP){ //Este está localizado na função de “Conferência do Ajuste da  
válvula”
```

A Figura 3.8 ilustra a simulação do sensor LM35 (ISIS) utilizado no projeto, neste simulador pode-se alterar a temperatura clicando na seta da direita, para aumentar a temperatura, e na seta da esquerda, para reduzir a temperatura.

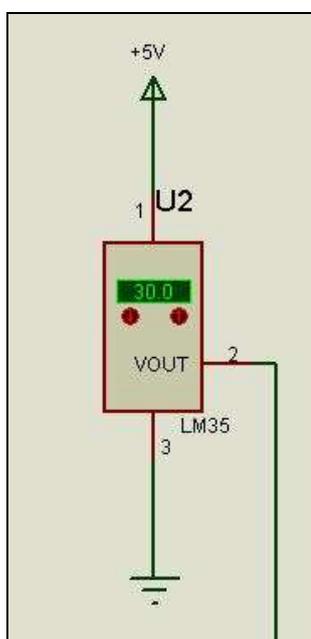


Figura 3.8 – Simulador do Sensor LM35 (ISIS)  
Fonte: (Autor)

### 3.7 Sensor de Fluxo e Fluido

O sensor de fluxo de fluido utiliza um rotor (turbina tipo “roda d’água”) de plástico, acoplado a um ímã e utiliza um sensor de efeito HALL. O modelo do sensor é o YF-G21-8 (Figura 3.9). Ele possui um conector (Figura 3.10) com três pinos:

- Alimentação (+Vcc) - Varia de 3V-24V;
- Saída (Vout) – pulsos;
- GND.

Ele trabalha com a faixa de 0,5 a 30 L/min, com pressão do fluido menor ou igual 1,75Mpa.

A saída do sensor é um sinal de onda quadrada, cuja frequência varia de acordo com o fluxo medido. Para uma alimentação de 5V, o nível alto é de 4,5V e o nível baixo 0,5V. A faixa de temperatura de operação de um sensor de fluxo varia entre -20°C à +100°C (sem formação de gelo).



Figura 3.9 – Sensor de Fluxo de Fluidos

Fonte: Disponível em <http://lojabrasilrobotics.blogspot.com.br/2011/01/sensor-de-fluxo-de-agua-r-3500.html>, Acesso em 11/11/2012.

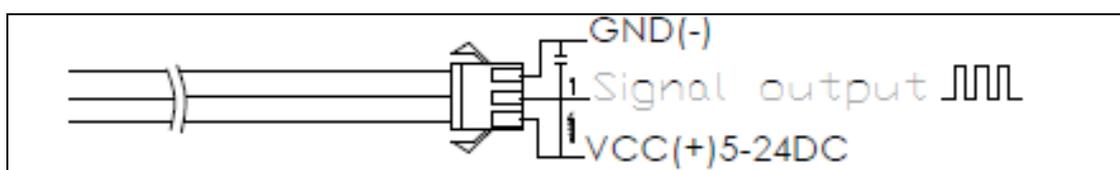


Figura 3.10 – Conector do sensor de Fluxo de Fluidos

Fonte: Disponível em [http://www.seeedstudio.com/wiki/images/b/b7/Water\\_flow\\_sensor\\_datasheet.pdf](http://www.seeedstudio.com/wiki/images/b/b7/Water_flow_sensor_datasheet.pdf), Acesso em 24/10/2012.

Este sensor no projeto é responsável por medir a quantidade de água utilizada na irrigação do jardim. O sensor de fluxo só calcula essa quantidade de água por um pulso gerado quando a água passa pela hélice dele, e para a água passar por ela a válvula tem que ser aberta pelo sistema.

De acordo com o catálogo do fabricante Yifa (ANEXO 3) , para  $F(\text{Hz})$ ,  $Q(\text{L}/\text{min})$ , o sensor usado segue a equação (3).

$$F = 8.1 * Q - 2; \quad (3)$$

F: Frequência dos pulsos. Medida em Hz.

Q: Fluxo de água. Medido em L/min.

Porém, a pesquisa do funcionamento do sensor indicou que o mesmo precisaria de um fator de calibração. Os testes aplicando a fórmula presente no catálogo confirmaram que a mesma apresenta uma margem de erro elevada (aproximadamente 20%). Após testes com vários valores, a equação que apresentou o menor erro foi a (4):

$$F = 7.0 * Q; \quad (4)$$

A fórmula coincide com a da documentação do YF-21 distribuído mundialmente pela Seeed Studio.

Como o sensor utilizado neste projeto, a medida realizada é a da frequência dos pulsos e a desejada para medir o consumo é a do fluxo de água. Reescrevendo a equação (4), o fluxo é descrito em função da frequência pela equação (5):

$$Q = \frac{F}{7.0}; \quad (5)$$

A equação (5) é aplicada para encontrar o fluxo de água e atualizar a medida de consumo a cada segundo, o fluxo de água utilizado deve estar em litros por segundo. Para isso aplica-se a relação:

$$Q_{(\text{L}/\text{min})} = 60 * Q_{(\text{L}/\text{s})}; \quad (6)$$

$$Q_{(L/s)} = \frac{Q_{(L/min)}}{60}, \quad (7)$$

Portanto a equação que descreve o fluxo de água em litros por segundo em função da frequência dos pulsos (tempo de rotação) emitidos pelo sensor é a equação (8):

$$Q_{(L/s)} = \frac{tmp\_rot}{7.0*60}, \quad (8)$$

Assim, a cada segundo, o fluxo de água medido é somado ao total do consumo.

Durante a fase de simulação, o sensor foi representado por um gerador de pulsos. A Figura 3.11 mostra o componente gerador de pulsos representando o sensor na simulação.

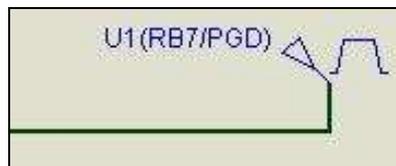


Figura 3.11 – Simulador do sensor YF-G21-8(ISIS)  
Fonte: (Autor)

O trecho de aplicação da equação (8) na programação do sistema é:

```
void atualizar_medida_agua(){
    tmp_rot=ROTACAO;
    if (tmp_rot != 0){
        QUANT = QUANT + (tmp_rot)/(7.0*60);    //Quantidade anterior + equação (8)
        ROTACAO = ROTACAO - tmp_rot;          //Remove da contagem de rotações as
        rotações já computadas
    }
}
```

### 3.8 Válvula EVA 15

A válvula solenóide usada no projeto é a válvula EVA 15, apresenta na Figura 3.12 (a e b).



Figura 3.12 (a) – Foto visão superior da válvula EVA 15.



Figura 3.12 (b) – Foto visão lateral da válvula EVA 15

Fonte: (Autor)

A descrição completa do EVA 15 estará no ANEXO 1, as principais características da válvula são:

EVA 15 – Válvula de entrada de água – reta

- Temperatura máxima do líquido: 60°C;
- Número de entradas/saídas: 1 entrada/ 1 saída;
- Bobina: 220Vca / 50-60Hz (bobina preta).

A válvula não é ligada diretamente no microcontrolador do projeto. No simulador do Proteus (Figura 3.13) apenas é colocado um motor que pode ser ligado diretamente, mas no circuito foi necessário conectar um relé antes de conectar a válvula, pois a mesma necessita de uma alta corrente, que o microcontrolador não oferece.

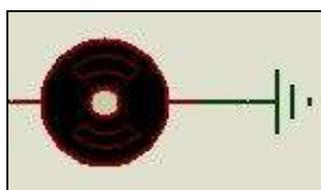


Figura 3.13 – Simulador da válvula (ISIS)

Fonte: (Autor)

A placa do circuito com relé, que fica conectada entre a válvula e o circuito principal do microcontrolador, está na Figura 3.14.



Figura 3.14 – Placa de circuito com relé  
Fonte: (Autor)

O trecho da programação que controla a válvula é separado em várias partes, mas a principal está na função de ajuste de válvula, pois a mesma controla as medidas dos sensores de temperatura e umidade, que são os principais pontos do projeto. O trecho é:

```
void ajusta_valvula()
{
  if (UMIDADE >= LIM_SECO)
  {
```

```

    abre_valvula();           //chamada da função de abertura da válvula
}else
{
    if (CELSIUS > LIM_TEMP){
        atualizar_umidade();   //chamada da função de atualização da umidade do solo
        if (UMIDADE >= LIM_UMIDO){
            abre_valvula();     //chamada da função de abertura da válvula
        }else{
            fecha_valvula();    //chamada da função de fechar a válvula
        }
    }else{
        fecha_valvula();       //chamada da função de fechar a válvula
    }
}
}
}

```

### 3.9 LCD

A programação do módulo LCD (Liquid Crystal Display ou Display de Cristal Líquido) foi feita utilizando a biblioteca de manipulação de módulo LCD, publicada no livro *Microcontroladores PIC Programação em C*, de Fábio Pereira e adaptada neste projeto (apêndice B) para utilizar a pinagem descrita na seção 3.3.

Essa biblioteca oculta a complexidade da comunicação com o módulo e torna o código mais legível utilizando chamadas a `lcd_escreve(mensagem)` ou `printf(lcd_escreve, mensagem, parâmetros)` para que uma mensagem seja mostrada no display LCD.

Para a utilização da biblioteca, além de incluir no código utilizando a diretiva `include` (no início do arquivo, `#include <lcd-v2.0.h>`), é necessário que o módulo seja inicializado com o comando `lcd_ini()`.

No código deste protótipo, a inicialização do módulo é feita dentro da função principal.

```
void main(){  
//... Partes do código foram omitidas nesta seção. Código completo disponível no Apêndice  
A  
    lcd_ini();           //inicializa LCD.  
    printf(lcd_escreve, "\nIniciando\nLeitura");  
}
```

## 4. TESTES E RESULTADOS OBITIDOS

### 4.1 Teste do Sensor de Umidade

O teste do sensor de umidade isoladamente foi feito medindo a saída com um multímetro e avaliando os valores para totalmente seco, saturado de umidade e valores intermediários.

Na etapa inicial de simulação, foram aplicadas diversas fórmulas encontradas em estudos publicados. Mas com os testes realizados apenas com um multímetro e uma fonte de tensão de 5V, apresentado na Figura 4.1, não foi possível confirmar que as medidas que o sensor Watermark 200SS estava fornecendo eram verdadeiras, e que, com isso, seria necessário realizar um processo de calibração com tensiômetros para confirmar se as fórmulas apresentadas nos estudos encontrados era adequada.

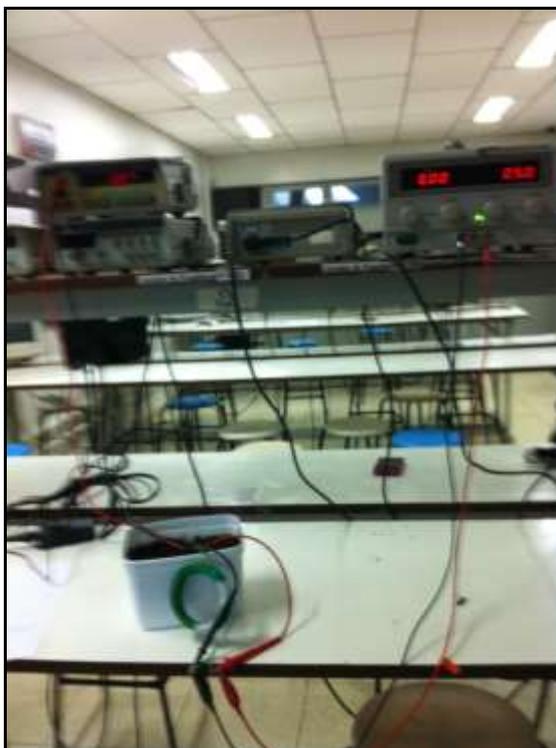


Figura 4.1 – Teste do sensor de umidade com um multímetro e uma fonte de tensão de 5V.  
Fonte: (Autor)

Foi procurada a ajuda da EMBRAPA (Empresa Brasileira de Pesquisa Agropecuária) para realizar a calibração do sensor Watermark 200SS, utilizado no protótipo. Obteve-se a resposta de um pesquisador da EMBRAPA CNPH (Centro Nacional de Pesquisa de Hortaliças) que estava por ajudar a conseguir os tensiômetros, mas por questões superiores não foi possível.

Por não ter sido possível realizar o processo com tensiômetros, foi utilizada a tabela de calibração do sensor Watermark 200SS-V, que não necessita do processo com tensiômetros por ser um sensor digital.

## **4.2 Teste do Sensor de Temperatura**

O teste do sensor de temperatura foi feito medindo a saída com um multímetro e conferindo a mudança na medida conforme a temperatura era alterada. O comportamento do sensor foi o esperado.

## **4.3 Teste do Sensor de Fluxo**

O teste do sensor de fluxo foi dividido em duas partes que serão abordados nos seguintes tópicos: (4.3.1.) Teste isolado e (4.3.2.) Teste com o sistema.

### **4.3.1 Teste Isolado**

O sensor foi ligado a um osciloscópio para a visualização dos pulsos emitidos. Com o teste foi possível verificar que o sensor emite um sinal constante em 0,5V quando não há fluxo e uma onda quadrada com amplitude de 4,5V, conforme a especificação.

Na Figura 4.2, está apresentada a imagem com o osciloscópio, a fonte de tensão de 5V e o sensor de fluxo de fluídos.

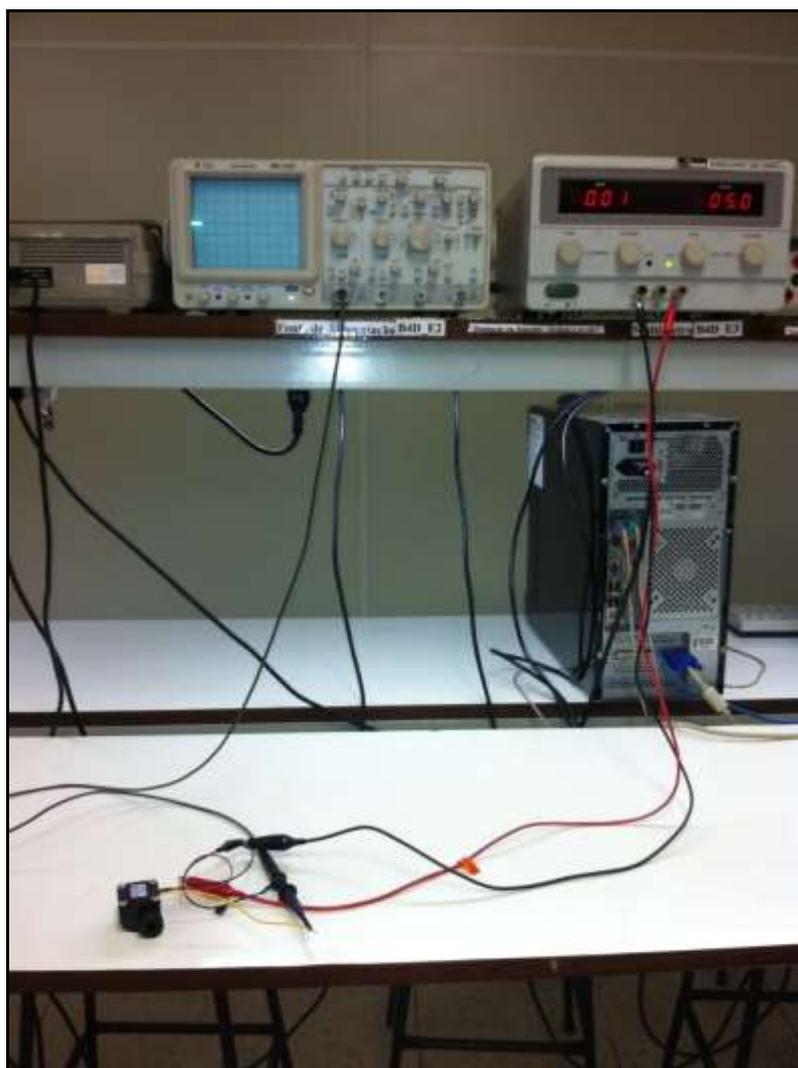


Figura 4.2 – Osciloscópio emitindo um sinal constante de 0,5V, com uma fonte de tensão de 5V e o sensor de fluxo de fluído.  
Fonte: (Autor)

A Figura 4.3 apresenta o osciloscópio com a medida da onda quadrada com amplitude de 4.5V.



Figura 4.3 – Osciloscópio com medida de onda quadrada de 4,5V  
Fonte: (Autor)

A Figura 4.4 apresenta o sensor de fluxo conectado ao osciloscópio e à fonte de tensão.



Figura 4.4 – Sensor de fluxo conectado ao osciloscópio e à fonte de tensão  
Fonte: (Autor)

#### 4.3.2 Teste com o Sistema

Com esse teste o fluxo foi medido e comparado com a marcação do volume de água de um balde, a partir das medidas feitas, foi possível perceber que o sensor necessitava ser calibrado. Essa informação já havia sido encontrada em alguns fóruns, mas como nenhuma documentação dos sensores de fluxo do fabricante mencionava uma etapa de calibração, esta só pode ser confirmada após os testes do sistema.

Após diversos testes utilizando balde com medidas em litros e protótipo, podem-se concluir fatores de calibração diferentes, mas as medidas mais corretas encontradas foram utilizando a relação da equação (4).

As Figuras 4.5 e 4.6 demonstram os testes realizados com o sistema e balde para tirar as medidas. Onde a Figura 4.5 mostra a medida no LCD do fluxo de água.

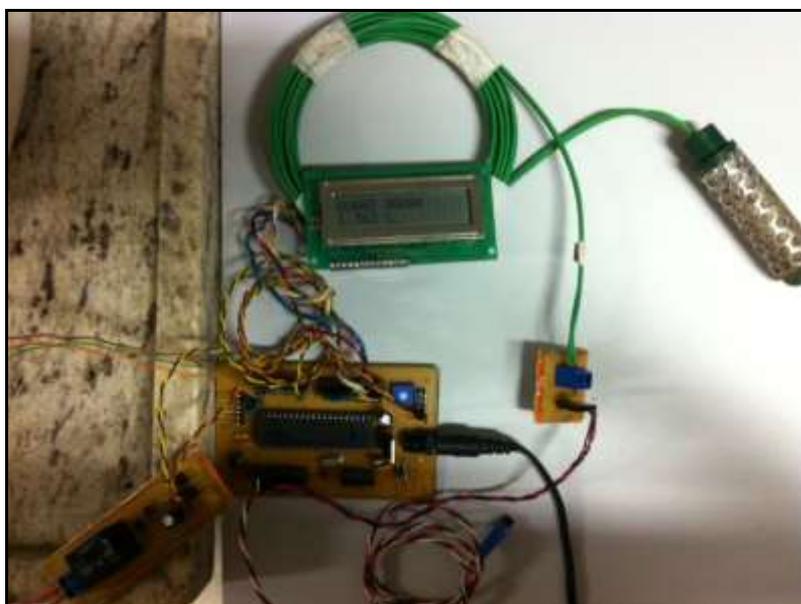


Figura 4.5 – Medida no LCD do fluxo de água  
Fonte: (Autor)

Figura 4.6 apresenta o sistema completo de teste com o balde de água.



Figura 4.6 – Sistema completo com balde de água  
Fonte: (Autor)

#### 4.4 Teste do Sistema

O teste do sistema foi realizado ligando todos os sensores e o LCD para verificar a operação do protótipo em ambiente similar ao que o dispositivo deve ser instalado. O teste realizado difere do funcionamento final do dispositivo em três pontos.

1. A versão do código utilizado teve parte do código modificada para que o tempo de leitura dos sensores de umidade e de temperatura fosse em segundos, e não em minutos. Assim foi possível verificar mais rápido o comportamento do dispositivo conforme os valores eram alterados.

2. O sensor de umidade não estava enterrado o tempo todo. Em alguns momentos foi enterrado, mas também ficou ao ar livre ou embebido em água (não danifica o sensor e é parte do processo de instalação do mesmo).

3. O sensor utilizado no protótipo é o Watermark 200SS, enquanto os dados de calibração utilizados são do 200SS-V, que é o sensor adequado para o dispositivo.

O sistema funcionou como esperado. A válvula foi aberta e fechada quando as condições de umidade e temperatura definidas foram atingidas. Enquanto isso, o display LCD mostrou os valores medidos pelos sensores e o consumo de água.

A Figura 4.7 protótipo montado (1).



Figura 4.7 – Protótipo montado (1)  
Fonte: (Autor)

Figura 4.8 protótipo montado (2).

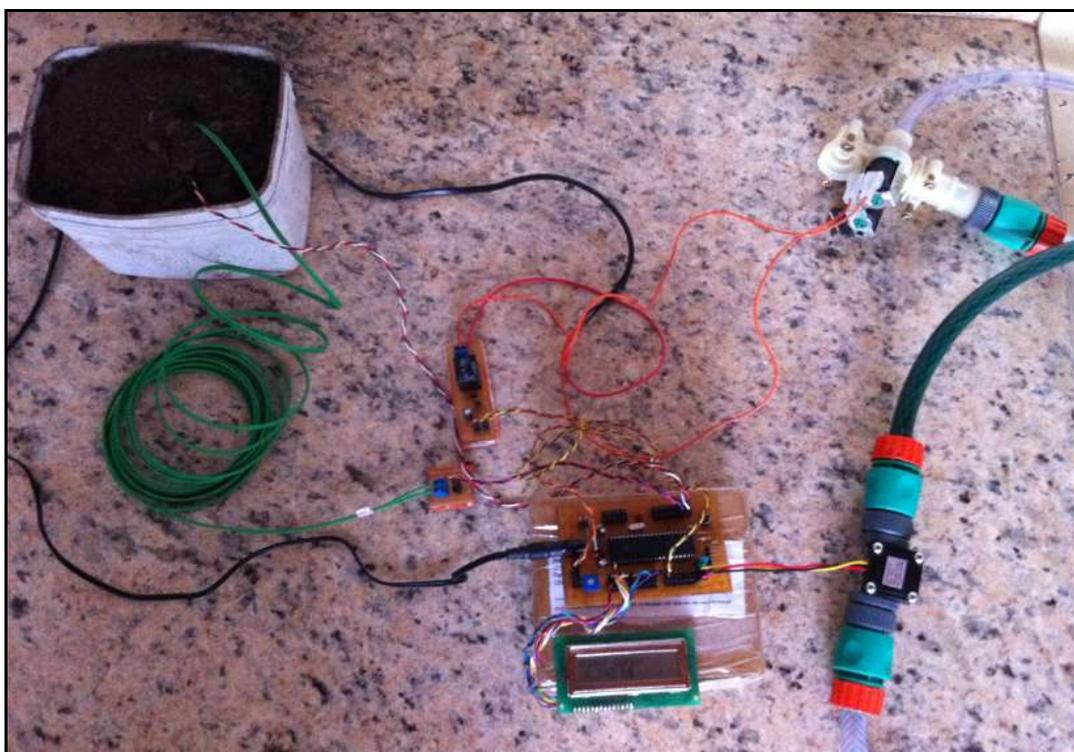


Figura 4.8 – Protótipo montado (2)  
Fonte: (Autor)

## 5. CONSIDERAÇÕES FINAIS

O objetivo deste projeto foi criar um sistema de controle de sistema de irrigação. Para tanto foi realizado o estudo do funcionamento e a aplicação de sensores de umidade do solo, temperatura e fluxo de água e de válvulas de controle. O objetivo foi atingido e o dispositivo criado pode ser utilizado, por exemplo, para controle de um sistema de irrigação por aspersão de um jardim.

A maior dificuldade encontrada no projeto foi relacionada ao sensor de umidade de solo escolhido inicialmente, que precisa de um tensiômetros para ser calibrado. Não foi possível realizar o processo de calibração e como solução foram utilizados dados de calibração do modelo 200SS-V que são disponibilizados pelo fabricante.

Portanto, ainda é necessário que o modelo do sensor de umidade seja substituído pelo Watermark 200SS-V, que não pode ser adquirido. Como alternativa, seria possível manter o sensor do protótipo atual, porém os limites das faixas de umidade precisariam ser ajustados até encontrar os valores ideais experimentalmente, o que não faz parte do escopo deste projeto.

Trabalhos futuros poderiam incluir a comunicação do dispositivo com um computador para incluir histórico de consumo, registro da umidade medida no solo ao longo do ano e configuração de parâmetros pelo usuário.

## 6. REFERÊNCIAS DE PESQUISAS

ASCO. **Informações técnicas sobre válvulas ASCO**. Disponível em: <<http://www.autoval.com.br/sites/default/files/3-inform-tecnicas.pdf>>. Acesso em: 25/10/2012.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6023**: informação e documentação – Referências - Elaboração. Rio de Janeiro, 2002.

Blog Sistemas Uniban. "**Arquitetura: Von Neumann Vs Harvard**". Disponível em: <<http://sistemasuniban.blogspot.com.br/2010/04/arquiterura-von-neumann-vs-harvard.html>>. Acesso em: 11/11/2012.

BOLTON, William. **Mecatrônica**: uma abordagem multidisciplinar. 4. ed. Porto Alegre: Bookman, 2010. 664 p.

Brasil Robotics Loja. **Sensor de fluxo de água 1/2**. Disponível em: <<http://lojabrasilrobotics.blogspot.com.br/2011/01/sensor-de-fluxo-de-agua-r-3500.html>>. Acesso em: 11/11/2012.

Carvalho, João Paulo de. "Enquete # 22 - **Densidade do solo, porosidade e água disponível**". Disponível em: <<http://www.pedologiafacil.com.br/enquetes/enq22.php>>. Acesso em: 10/10/2012.

Coelho, Silvana de Lourdes. **DESENVOLVIMENTO DE UM TENSÍOMETRO ELETRÔNICO PARA O MONITORAMENTO DO POTENCIAL DA ÁGUA NO SOLO**. Fortaleza, 2008. Dissertação de mestrado. Universidade Federal do Ceará.

Fisher Controls International. **CONTROL VALVE HANDBOOK**. 4ª Edição. 2005.

GUARIZ, H. R.; CAMPANHARO, W. A.; PICOLI, M. H. S.; CECÍLIO, R. A.; HOLLANDA, M. P. **Variação da umidade e da densidade do solo sob diferentes coberturas vegetais**. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO, 14. (SBSR), 2009, Natal. Anais... São José dos Campos: INPE, 2009. p. 7709-7716. DVD, On-line. ISBN 978-85-17-00044-7. Disponível em: <<http://urlib.net/dpi.inpe.br/sbsr@80/2008/11.18.04.40>>. Acesso em: 09/10/2012

Inácio, Augusto Ávila Do Nascimento. **Controle de Irrigação de Hortaliças por Microcontrolador**. Brasília, 2012. 70p. Monografia. Centro Universitário de Brasília.

Irrrometer/ Watermark <<http://www.irrometer.com/sensors.html#wm>> Acesso em: 10/11/2012

Jefferson Solenoid Valves. **Válvulas Solenóides Informação de Engenharia**. Disponível em: <<http://www.jeffersonvalves.com/repositorio/pdfs-portugues/general-catalog.pdf>>. Acesso em: 25/10/2012.

Light, J. E.; Mitchell, A. R.; Barnum, J. M.; Shock, C. C. "**GRANULAR MATRIX SENSORS FOR IRRIGATION MANAGEMENT**". Disponível em: <[http://oregonstate.edu/dept/coarc/sites/default/files/publication/92\\_granular\\_matrix\\_sensors.pdf](http://oregonstate.edu/dept/coarc/sites/default/files/publication/92_granular_matrix_sensors.pdf)>. Acesso em: 20/09/2012

Microchip. **PIC16F87XA Data Sheet**. 2003. Disponível em: <<http://ww1.microchip.com/downloads/en/DeviceDoc/39582b.pdf>>. Acesso em 01/10/2012.

Oliveira, Adalberto Luiz de Lima. **Instrumentação - Elementos Finais de Controle**. SENAI. 1999.

Parker Hannifin Corp. **Válvulas e Acessórios para Controle de Fluido** - Catálogo 4201-5 BR - Controle em processos críticos e segurança. Disponível em: <[http://www.norless.com.br/downloads/Valvulas\\_Solenoides.pdf](http://www.norless.com.br/downloads/Valvulas_Solenoides.pdf)>. Acesso em: 25/10/2012

Pereira, Fábio. **Microcontroladores PIC**. Programação em C. 2 ed. São Paulo: Érica, 2003.

RIBEIRO, Marco Antônio. **Instrumentação**. 9.ed. Salvador. Tek Treinamento e Consultoria. 2002.

RIBEIRO, Marco Antônio. **Válvulas de Controle e Segurança**. 5.ed (Revista). Salvador. Tek Treinamento e Consultoria. 1999.

Shock, C. C.; Barnum, J. M.; Seddigh, M. Calibration of W~!ermark. **Soil Moisture Sensors for Irrigation Management**. Ontario, Oregon. Oregon State University, 1998.

Site Cultivando. "**Como reduzir o consumo de água doce?**". Disponível em: <[http://www.cultivando.com.br/saude\\_meio\\_ambiente\\_agua\\_reduzindo\\_o\\_consumo.html](http://www.cultivando.com.br/saude_meio_ambiente_agua_reduzindo_o_consumo.html)>. Acesso em: 17/09/2012

Souza, David José de; LAVINIA, Nicolás César. **Conectando o PIC 16F877A: recursos avançados**, 4. ed., São Paulo, Érica, 2007

Thomazini, Daniel. Albuquerque, Pedro U.B. "**Sensores Industriais - Fundamentos e Aplicações**". São Paulo: Érica, 2007.

Trabalhos ABNT. "**REGRAS E NORMAS DA ABNT 2012 PARA FORMATAÇÃO DE TRABALHOS ACADÊMICOS**". 2012. Disponível em: <<http://www.trabalhosabnt.com/regras-normas-abnt-formatacao>>. Acesso em: 10/09/2012.

Vieira, Hélio. "**TUTORIAL**: Sensor de temperatura com Arduino". Blog Encomper Eletrônica e Computação. 2011. Disponível em: <<http://engcomper.blogspot.com.br/2011/08/tutorial-sensor-de-temperatura-com.html>>. Acesso em: 23/10/2012.

VIELMO, Alexandro. **Limite superior da retenção da água no solo**: método de campo e método de estimativa. Santa Maria, 2008, 82p. Dissertação de mestrado. Universidade Federal de Santa Maria.

YIFA. **Catálogo de produtos sensores de fluxo de água**. 2010.

YIFA the Plastics Ltd. **Product Introduction**: YF-21. Disponível em: <[http://www.seeedstudio.com/wiki/images/b/b7/Water\\_flow\\_sensor\\_datasheet.pdf](http://www.seeedstudio.com/wiki/images/b/b7/Water_flow_sensor_datasheet.pdf)>. Acesso em: 24/10/2012.

## 7. APÊNDICES

### Apêndice A Código Fonte do Projeto (sistema\_irrigacao.c)

```

/*=====
PROJETO FINAL - Engenharia da Computação - UniCEUB
2o. Semestre de 2012

Manuella Thereza Pereira Pouso
RA: 20817535

SISTEMA DE AUTOMAÇÃO E CONTROLE DE UM SISTEMA DE IRRIGAÇÃO
=====*/

#include <16f877A.h> //Inclui biblioteca do microcontrolador PIC 16f877A
#define adc=10 //Define o conversor analógico digital na resolução 10 bits
#define delay(clock=4000000) //Cristal Clock de 4Mhz
#define fuses xt,nowdt,put,nobrownout,nolvp
#define use fast_io(B) //Define que as configurações de TRIS devem ser feitas
//manualmente [ex: set_tris_b()]

#define byte portb=0x06

#include <lcd-v2.0.h> //Inclui a biblioteca do módulo LCD (Fabio Pereira).
//Esta biblioteca sofreu alteração para utilizar os pinos de dados, R/W, RS e
//Enable do LCD no Port C e D do PIC.

```

```
/*=====*/  
#use rs232(baud=4800,xmit=PIN_C6,rcv=PIN_C7)  
//Configura comunicação serial para:  
/* Velocidade de transmissão 4800 bits, sem paridade, 8 bits de dados.  
RC6 pino de transmissão.  
RC7 pino de recepção (não utilizado).*/  
/*=====*/  
  
#define VALVULA PIN_B0 //Define a válvula na porta B0 do PIC.  
  
#define RPM PIN_B7 //Define o contador de pulso/ Entrada p/ sensor de fluxo na  
//porta B7 do PIC.  
  
#define LIM_UMIDO 40 //Limite de solo saturado.  
  
#define LIM_SECO 70 //Limite de solo seco.  
  
#define LIM_QSECO 60 //Limite de solo quase seco.  
  
#define LIM_TEMP 30 //Limite de temperatura.  
  
#define TIMER_UMIDO 30 //Solo úmido o tempo de espera para a próxima medida.  
  
#define TIMER_SECO 5 //Solo seco o tempo de espera para a próxima medida.  
  
#define TIMER_TEMP 5 //Tempo de espera para medida do sensor de temperatura.
```

```

/*=====
DECLARAÇÃO DAS FUNÇÕES CRIADAS
=====*/

void init_adc();

void mostrar_proxima_mensagem();

void atualizar_medida_agua();

void atualizar_temperatura();

void atualizar_umidade();

void ajusta_valvula();

void abre_valvula();

void fecha_valvula();

```

```

/*=====
CRIAÇÃO DAS VARIÁVEIS GLOBAIS
=====*/

long ROTACAO = 0;

long tmp_rot=0;

int conta = 0;

int segundos = 0;

signed long min_leitura_umidade=0;

signed long min_leitura_temp=0;

int1 valvula_aberta = 0;

byte mensagem = 0; // Define a mensagem que o LCD vai mostrar

unsigned long VALOR=0, TEMP=0;

float QUANT=0.000f;//Declara variável CONTA de 16 bits inicializando-a com zero.

float UMIDADE=0, MEDIDA=0, CELSIUS=0;

```

```

/*=====
                                     FUNÇÃO DO SENSOR DE FLUXO
=====*/

#include <avr/io.h>
#include <avr/interrupt.h>

#define INT_RB 1

void RB_ISR() {
    if (input(RPM)) {
        ROTACAO++;
    }
}

/*=====
                                     FUNÇÃO TIMER (TEMPO DE ESPERA DO SISTEMA)
=====*/

#include <avr/interrupt.h>

void interrupcao_timer() {

    conta++;

    if (conta == 125) { //125 == 1 segundo
        conta = 0;
        segundos++;

        if (segundos == 60) {
            segundos = 0;
            min_leitura_umidade--;
            min_leitura_temp--;
        }

        atualizar_medida_agua(); //Chamada da função de atualização da medida
                                //de água

        if (segundos % 3 == 0) { //apresentar uma vez a cada 3 segundos

```

```

    mostrar_proxima_mensagem(); //Chamada da função de próxima mensagens
}
}
set_timer0(131-get_timer0()); //Atualiza Timer
}

/*=====
                        FUNÇÃO DE PRÓXIMA MENSAGEM
=====*/

void mostrar_proxima_mensagem(){
    switch(mensagem){
        case 0: //Escreve no LCD a quantidade de água.
            printf(lcd_escreve, "\fQUANT AGUA:\n%01.3f L", QUANT);
            break;
        case 1: //Escreve no LCD o valor que o sensor de temperatura mediu
            printf(lcd_escreve, "\fTemp:%03.1foC", CELSIUS);
            break;
        case 2: //Escreve no LCD o valor da umidade do solo e a tensão medida
            //pelo sensor de umidade
            printf(lcd_escreve, "\fPres:%3.3fcbar\nTens:%03.3fV", UMIDADE, MEDIDA);
            break;
    }

    mensagem++;
    if (mensagem > 2) {
        mensagem = 0;
    }
}
}

```

```

/*=====
                                     INICIALIZAÇÃO DA FUNÇÃO PRINCIPAL
=====*/

void main()
{
    set_tris_b(0b10000000); //Definição manual da porta B7
    output_b(0x00);        //limpa portas
    output_c(0x00);
    output_d(0x00);
    output_e(0x00);

    init_adc(); //Chamada da função de conversor analógico
    lcd_ini(); //inicializa LCD.

    setup_timer_0( RTCC_INTERNAL | RTCC_DIV_64); //Configura o timer para o usar o
        //clock interno e prescaler 64
    set_timer0(131); //define o valor inicial da contagem do timer
    clear_interrupt(INT_RB); //limpa a flag de interrupção
    ext_int_edge(L_TO_H); //borda de subida da interrupção
    enable_interrupts(INT_RB); //habilita a interrupção na porta B
    enable_interrupts(GLOBAL); //habilita a interrupção Global
    enable_interrupts(int_timer0); //habilita a interrupção do timer 0
    setup_timer_1(T1_DISABLED);
    setup_timer_2(T2_DISABLED,0,1);
    printf(lcd_escreve, "\nIniciando\nLeitura"); //LCD escreve mensagem de
        //inicialização de processo
    do{
        if (min_leitura_temp <= 0){

```

```

    atualizar_temperatura();//chamada da função de atualização da temperatura
    ajusta_valvula();//chamada da função de conferência do ajuste da válvula
}
if (min_leitura_umidade <= 0){
    atualizar_umidade();//chamada da função de atualização da umidade do solo
    ajusta_valvula();//chamada da função de conferência do ajuste da válvula
}

}while(TRUE);
}

/*=====
                        FUNÇÃO DE CONVERSÃO ANALÓGICA DIGITAL
=====*/
void init_adc(void)
{
    setup_adc_ports(ALL_ANALOG);
    setup_adc(ADC_CLOCK_INTERNAL);
}

/*=====
                        FUNÇÃO DE ATUALIZAÇÃO DA TEMPERATURA
=====*/
void atualizar_temperatura()
{
    set_adc_channel(0);
    delay_ms(10);
    TEMP=read_adc();

```

```

    CELSIUS=TEMP*100*(5.0/1023.0);
    min_leitura_temp = TIMER_TEMP;
}

/*=====
                FUNÇÃO DE ATUALIZAÇÃO DA UMIDADE DO SOLO
=====*/

void atualizar_umidade()
{
    set_adc_channel(1);
    delay_ms(10);
    VALOR=read_adc();
    MEDIDA=5.0*(VALOR/1023.0);
    UMIDADE =(MEDIDA+(5/1000000000))/(0.0117155);
    if (UMIDADE >= LIM_QSECO){
        min_leitura_umidade = TIMER_SECO;
    }else{
        min_leitura_umidade = TIMER_UMIDO;
    }
}

/*=====
                FUNÇÃO DE ATUALIZAÇÃO DA MEDIDA DE ÁGUA
=====*/

void atualizar_medida_agua(){
    tmp_rot=ROTACAO;
    if (tmp_rot != 0){
        QUANT = QUANT + (tmp_rot)/(7.0*60); //Quantidade anterior + Formula 2
    }
}

```

```

    ROTACAO = ROTACAO - tmp_rot; //Remove da contagem de rotações as rotações já
    computadas

```

```

    }

```

```

}

```

```

/*=====

```

### FUNÇÃO DE CONFÊRENCIA DO AJUSTE DA VÁLVULA

```

=====*/

```

```

void ajusta_valvula()

```

```

{

```

```

    if (UMIDADE >= LIM_SECO)

```

```

    {

```

```

        abre_valvula();//chamada da função de abertura da válvula

```

```

    }else

```

```

    {

```

```

        if (CELSIUS > LIM_TEMP){

```

```

            atualizar_umidade();//chamada da função de atualização da umidade do solo

```

```

            if (UMIDADE >= LIM_UMIDO){

```

```

                abre_valvula();//chamada da função de abertura da válvula

```

```

            }else{

```

```

                fecha_valvula();//chamada da função de fechar a válvula

```

```

            }

```

```

        }else{

```

```

            fecha_valvula();//chamada da função de fechar a válvula

```

```

        }

```

```

    }

```

```

}

```

```
/*=====
FUNÇÃO DE ABERTURA DA VÁLVULA
=====*/
```

```
void abre_valvula(){
    output_high(VALVULA);
    min_leitura_temp = TIMER_TEMP;
    min_leitura_umidade = TIMER_SECO;
    valvula_aberta = 1; // 1 -> verdadeiro
}
```

```
/*=====
FUNÇÃO DE FECHAR A VÁLVULA
=====*/
```

```
void fecha_valvula()
{
    output_low(VALVULA);
    if (valvula_aberta)
    {
        valvula_aberta = 0; //0 -> falso
        min_leitura_umidade = TIMER_UMIDO;
    }
}
```

## Apêndice B      Biblioteca do LCD (lcd-v2.0.h)

```

/*****/

/*MOD_LCD.C - Biblioteca de manipulação de módulo LCD */

/* */

/*Autor: Fábio Pereira */

/*Esta biblioteca foi modificada para utilizar os pinos do Port D do PIC*/

/*para o LCD: */

/*Pino Enable (E) ligado ao pino RD5 */
/*Pino RS ligado ao pino RD4 */
/*Pino RW ligado ao pino RD6 (não utilizado) */

/*****/

// As definições a seguir são utilizadas para acesso aos pinos do display
// caso o pino RW não seja utilizado, comente a definição lcd_rw
#ifndef lcd_enable
    #define lcd_enable    pin_d5        // pino enable do LCD
    #define lcd_rs       pin_d4        // pino rs do LCD
    //#define lcd_rw     pin_d6        // pino rw do LCD
    #define lcd_d4       pin_d2        // pino de dados d4 do LCD
    #define lcd_d5       pin_d3        // pino de dados d5 do LCD
    #define lcd_d6       pin_c4        // pino de dados d6 do LCD
    #define lcd_d7       pin_c5        // pino de dados d7 do LCD
#endif

#define lcd_type 2        // 0=5x7, 1=5x10, 2=2 linhas

```

```
#define lcd_seg_lin 0x40 // Endereço da segunda linha na RAM do LCD

// a constante abaixo define a seqüência de inicialização do módulo LCD
byte CONST INI_LCD[4] = {0x20 | (lcd_type << 2), 0xf, 1, 6};

byte lcd_le_byte()
// lê um byte do LCD (somente com pino RW)
{
    byte dado;

    // configura os pinos de dados como entradas
    input(lcd_d4);
    input(lcd_d5);
    input(lcd_d6);
    input(lcd_d7);

    // se o pino rw for utilizado, coloca em 1
    #ifdef lcd_rw
        output_high(lcd_rw);
    #endif

    output_high(lcd_enable); // habilita display

    dado = 0; // zera a variável de leitura

    // lê os quatro bits mais significativos
    if (input(lcd_d7)) bit_set(dado,7);
    if (input(lcd_d6)) bit_set(dado,6);
    if (input(lcd_d5)) bit_set(dado,5);
    if (input(lcd_d4)) bit_set(dado,4);

    // dá um pulso na linha enable
    output_low(lcd_enable);
    output_high(lcd_enable);
}
```

```
    // lê os quatro bits menos significativos
    if (input(lcd_d7)) bit_set(dado,3);
    if (input(lcd_d6)) bit_set(dado,2);
    if (input(lcd_d5)) bit_set(dado,1);
    if (input(lcd_d4)) bit_set(dado,0);
    output_low(lcd_enable);    // desabilita o display
    return dado;    // retorna o byte lido
}
```

```
void lcd_envia_nibble( byte dado )
// envia um dado de quatro bits para o display
{
    // coloca os quatro bits nas saidas
    output_bit(lcd_d4,bit_test(dado,0));
    output_bit(lcd_d5,bit_test(dado,1));
    output_bit(lcd_d6,bit_test(dado,2));
    output_bit(lcd_d7,bit_test(dado,3));
    // dá um pulso na linha enable
    output_high(lcd_enable);
    output_low(lcd_enable);
}
```

```
void lcd_envia_byte( boolean endereco, byte dado )
{
    // coloca a linha rs em 0
    output_low(lcd_rs);
    // aguarda o display ficar desocupado
```

```

//while ( bit_test(lcd_le_byte(),7) ) ;

// configura a linha rs dependendo do modo selecionado
output_bit(lcd_rs,endereco);

delay_us(100);      // aguarda 100 us

// caso a linha rw esteja definida, coloca em 0
#ifdef lcd_rw
    output_low(lcd_rw);
#endif

// desativa linha enable
output_low(lcd_enable);

// envia a primeira parte do byte
lcd_envia_nibble(dado >> 4);

// envia a segunda parte do byte
lcd_envia_nibble(dado & 0x0f);
}

```

```

void lcd_ini()

// rotina de inicialização do display
{
    byte conta;

    output_low(lcd_d4);
    output_low(lcd_d5);
    output_low(lcd_d6);
    output_low(lcd_d7);
    output_low(lcd_rs);

#ifdef lcd_rw
    output_high(lcd_rw);

```

```
#endif
output_low(lcd_enable);
delay_ms(15);
// envia uma seqüência de 3 vezes 0x03
// e depois 0x02 para configurar o módulo
// para modo de 4 bits
for(conta=1;conta<=3;++conta)
{
    lcd_envia_nibble(3);
    delay_ms(5);
}
lcd_envia_nibble(2);
// envia string de inicialização do display
for(conta=0;conta<=3;++conta) lcd_envia_byte(0,INI_LCD[conta]);
}
```

```
void lcd_pos_xy( byte x, byte y)
{
    byte endereco;
    if(y!=1)
        endereco = lcd_seg_lin;
    else
        endereco = 0;
    endereco += x-1;
    lcd_envia_byte(0,0x80|endereco);
}
```

```
void lcd_escreve( char c)
```

```
// envia caractere para o display
{
  switch (c)
  {
    case '\f' :    lcd_envia_byte(0,1);
                  delay_ms(2);
                  break;

    case '\n' :
      case '\r' :    lcd_pos_xy(1,2);
                    break;

    case '\b' :    lcd_envia_byte(0,0x10);
                  break;

    default :    lcd_envia_byte(1,c);
                break;
  }
}
```

```
char lcd_le( byte x, byte y)
// le caractere do display
{
  char valor;

  // seleciona a posição do caractere
  lcd_pos_xy(x,y);

  // ativa rs
  output_high(lcd_rs);

  // lê o caractere
  valor = lcd_le_byte();

  // desativa rs
```

```
output_low(lcd_rs);  
// retorna o valor do caractere  
return valor;  
}
```

## 8. ANEXOS

### Anexo 1 Pinagem detalhada do PIC16F877A

Nome do Pino	Núm. Pino	I/O/P	Tipo	Descrição
MCLR/Vpp	1	I/p	ST	Master Clear (reset) externo. O microcontrolador só funciona quando este pino encontra-se em nível alto. Entrada para tensão de programação (13V).
Vss	12/3	p	-	GND.
vdd	11/3	p	-	Alimentação positiva.
RA0/AN0 RA1/AN1  RA2/AN2/ V <sub>ref</sub> /CV <sub>ref</sub>  RA3/AN3/V <sub>REF+</sub>  RA4 / T0CKI / C1OUT  RA5/SS/AN4/ C2OUT	2 3  4  5  6  7	I/O I/O  I/O  I/O  I/O	TTL TTL  TTL  TTL  ST  TTL	<b>PORTA</b> (I/Os digitais bidirecionais e sistema analógico): RA0: I/O digital ou entrada analógica AN0. RA1 : I/O digital ou entrada analógica AN1 .  RA2: I/O digital ou entrada analógica AN2 ou tensão negativa de referência analógica.  RA3: I/O digital ou entrada analógica AN3 ou tensão positiva de referência analógica. RA4: I/O digital (quando saída é open drain, isto é, não consegue impor nível alto) ou entrada externa do contador TMR0 ou saída do comparador 1 . RA5: I/O digital ou entrada analógica AN4 ou habilitação externa (slave select) para comunicação SPI ou saída do comparador 2.
RB0/INT RB1 RB2  RB3/PGM RB4 RB5  RB6/PGC  RB7/PGD	33 34 35  36 37 38  39  40	I/O I/O I/O  I/O I/O I/O  I/O  I/O	TTL/ST <sup>(1)</sup> TTL TTL  TTL TTL TTL  TTL/ST  TT17ST <sup>(2)</sup>	<b>PORTB</b> (I/Os digitais bidirecionais). Todos os pinos deste PORT possuem pull-up interno que podem ser ligados/desligados pelo software: RB0: I/O digital com interrupção externa. RB1: I/O digital. RB2: I/O digital. RB3: I/O digital ou entrada para programação em baixa tensão (5V). RB4: I/O digital com interrupção por mudança de estado. RB5: I/O digital com interrupção por mudança de estado. RB6: I/O digital com interrupção por mudança de estado ou clock da programação serial ou pino de in-circuit debugger. RB7: I/O digital com interrupção por mudança de estado ou data da programação serial ou pino de in-circuit debugger.
RC0/T10SO/ T1CKI RC1/T10SI/CCP2 RC2/CCP1  RC3/SCK/SCL RC4/SDI/SDA	15 16 17  18  23	I/O I/O I/O  I/O  I/O	ST ST ST  ST  ST	<b>PORTC</b> (I/Os digitais bidirecionais): RC0: I/O digital ou saída do oscilador externo para TMR1 ou entrada de incremento para TMR1 . RC1: I/O digital ou entrada do oscilador externo para TMR1 ou entrada do Capture2 ou saídas para Compare2/PWM2. RC2: I/O digital ou entrada do Capture1 ou saídas para Compare1/PWM1. RC3: I/O digital ou entrada/saída de clock para comunicação serial SPI / I <sup>2</sup> C. RC4: I/O digital ou entrada de dados para SPI ou via de dados (entrada/saída) para I <sup>2</sup> C.

Nome do Pino	Núm. Pino	I/O/P	Tipo	Descrição
RC5/SDO	24	I/O	ST	RC5: I/O digital e saída de dados para SPI.
RC6/TX/CK	25	I/O	ST	RC6: I/O digital ou TX (transmissão) para comunicação USART assíncrona ou clock para comunicação síncrona.
RC7/RX/DT	26	I/O	ST	RC7: I/O digital ou RX (recepção) para comunicação USART assíncrona ou data para comunicação síncrona.
RD0/PSP0 RD1/PSP1 RD2/PSP2 RD3/PSP3 RD4/PSP4 RD5/PSP5 RD6/PSP6 RD7/PSP7	19 20 21 22 27 28 29 30	I/O I/O I/O I/O I/O I/O I/O I/O	TTL/ST <sup>(3)</sup> TTL/ST <sup>(3)</sup> TTL/ST <sup>(3)</sup> TTL/ST <sup>(3)</sup> TTL/ST <sup>(3)</sup> TTL/ST <sup>(3)</sup> TTL/ST <sup>(3)</sup> TTL/ST <sup>(3)</sup>	<b>PORTD</b> (I/Os digitais bidirecionais) ou porta de comunicação paralela. RD0: I/O digital ou dado 0 (comunicação paralela). RD1: I/O digital ou dado 1 (comunicação paralela). RD2: I/O digital ou dado 2 (comunicação paralela). RD3: I/O digital ou dado 3 (comunicação paralela). RD4: I/O digital ou dado 4 (comunicação paralela). RD5: I/O digital ou dado 5 (comunicação paralela). RD6: I/O digital ou dado 6 (comunicação paralela). RD7: I/O digital ou dado 7 (comunicação paralela).
RE0/RD/AN5 RE1/WR/AN6 RE2/CS/AN7	8 9 10	I/O I/O I/O	TTL/ST <sup>(3)</sup> TTL/ST <sup>(3)</sup> TTL/ST <sup>(3)</sup>	<b>PORTE</b> (I/Os digitais bidirecionais e sistema analógico): RE0: I/O digital ou controle de leitura da porta paralela ou entrada analógica AN5. RE1: I/O digital ou controle de escrita da porta paralela ou entrada analógica AN6. RE2: I/O digital ou habilitação externa da porta paralela ou entrada analógica AN7.

**Legenda:**

I	=	Input (entrada)
O	=	Output(saída)
I/O	=	Input/Output (entrada ou saída)
P	=	Power (alimentação)
-	=	Não-utilizado
TTL	=	Entrada tipo TTL
ST	=	Estrada tipo <i>Schmitt Trigger</i>

**Notas:**

- (1) Esta entrada é do tipo ST, somente quando configurado como interrupção externa.
- (2) Esta entrada é do tipo ST, somente durante o modo de programação serial.
- (3) Esta entrada é do tipo ST, quando configurado como I/O de uso geral e TTL quando usado em modo de porta paralela.
- (4) Esta entrada é ST quando em modo RC e CMOS nos demais casos.

## Anexo 2 Medidas de dados de calibração do Sensor 200SS-V

### 200SS-V Calibration

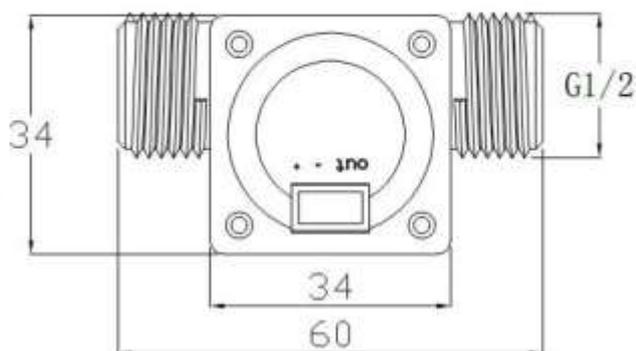
0-2.8V represents 0-239 CB

0.0117155 V to 1 Centibar (kPa)

Centibar (kPa)	Voltage
0	0
10	0.117155
20	0.23431
30	0.351464
40	0.468619
50	0.585774
60	0.702929
70	0.820084
80	0.937238
90	1.054393
100	1.171548
120	1.405858
130	1.523013
140	1.640167
150	1.757322
160	1.874477
170	1.991632
180	2.108787
190	2.225941
200	2.343096
210	2.460251
220	2.577406
230	2.694561
240	2.811715

Fonte: Imagem editada do site: <http://www.irrometer.com/pdf/supportmaterial/sensors/voltage-WM-chart.pdf> acessado em 20/11/2012

### Anexo 3 Sensor de Fluxo de Flúido



型号:YF-21-8(进水口内径9mm)

#### 1、使用条件:

工作电压范围 **DC3-24V**  
 额定工作电压 **DC5V**高电平**4.5V**以上, 低电平**0.5V**以下  
 使用温度范围 **-20~+100℃**(无结冰状态)  
 允许耐压 **1.75MPA**

#### 2、流量脉冲特性:

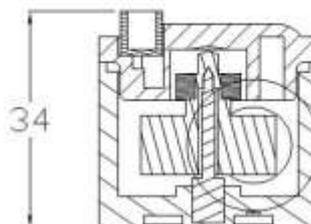
脉冲频率 (HZ)  $\approx 6 \cdot Q \pm 2\%$ , Q=流量(L/min)

#### 3、密封性:

封闭各孔, 加**1.75MPA**水压试验1分钟无漏水和变形现象

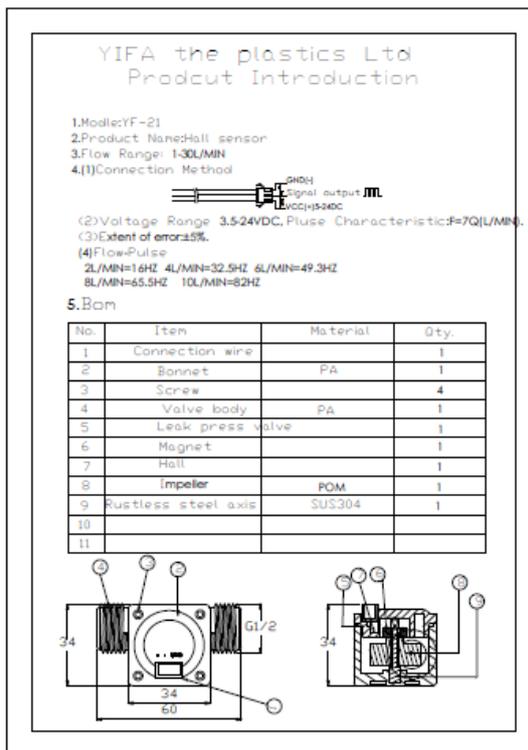
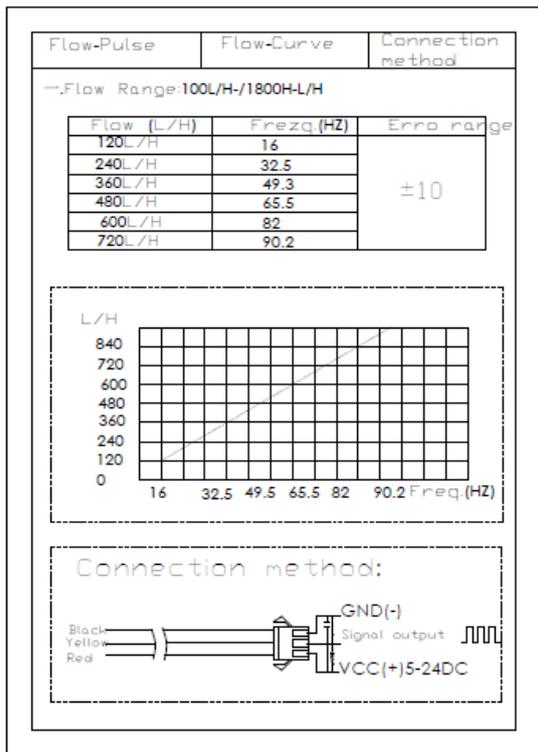
#### 备注:

流量计进水口尺寸和连接线接线方式, 线长均可按客户要求定做!



Tel: 0757-22109057 Fax: 0757-28118188 Http://www.fandyifa.com.cn

Fonte: YIFA. Catálogo de produtos sensores de fluxo de água. 2010.



Fonte: [http://www.seedstudio.com/wiki/images/b/b7/Water\\_flow\\_sensor\\_datasheet.pdf](http://www.seedstudio.com/wiki/images/b/b7/Water_flow_sensor_datasheet.pdf). Acessado em: 24/10/2012

## Anexo 4 Válvula EVA 15

Descrição completa da válvula EVA 15, que utilizada no projeto.



EVA 15 - Válvula de entrada de água - reta

**Materiais:** Corpo: termoplástico  
 Filtro: plástico (removível)  
 Partes metálicas: aço zincado  
 Membrana: borracha (padrão) e silicone (atóxica)  
 Terminais: latão

**Pressão de operação:**  
 de 0,2 à 8 kgf/cm<sup>2</sup>  
 à 0,2kgf/cm<sup>2</sup> , vazão mínima= 7 l/min;  
 à 8 kgf/cm<sup>2</sup>; vazão máxima= 40 l/min;  
*Opcional - regulador de vazão (5l/min., 6,5 l/min., 8 l/min., 12 l/min., 20 l/min.)*

**Temperatura máxima do líquido:** 60°C

**Rigidez dielétrica:** 1.500Vca - 1 min.

**Saída de água (conexão para mangueira)**  
 Disposição geométrica em relação à entrada: 90°

**Número de entradas / saídas:** 1 entrada / 1 saída

**Vida útil:** 50.000 operações

**Tipo de terminal (alimentação):**  
 Padrão reto; dobrado (sob consulta)  
 Faston 6,3mm x 0,8mm  
 Faston 4,8 x 0,8mm

**Bobina:**  
 Tensão: 127Vca / 60Hz (bobina amarela)  
 220Vca / 50-60Hz (bobina preta)  
 220-240Vca / 50Hz (bobina vermelha)  
 12Vcc (bobina azul)  
 24Vcc (bobina verde)  
*Outras tensões sob consulta.*

**Posições de montagem:**  
 da bobina em relação à entrada d' água: alojamentos: 0 a 7  
 do suporte em relação à bobina: alojamentos: 1 a 4

**Entrada de água:** rosca  
 1...3/4" x 11,5 - NHT  
 2...3/4" x 14 - NBR-8133  
*Opcional: terminal terra*

**Dimensões de saída (mm):**

Código	Ø B	Ø C
11	10.2	12.0

**Gravações:** conforme especificações do cliente

Fonte: Imagem editada do site: [http://emicol.com/produtos.php?menu\\_id=3&pro\\_id=149](http://emicol.com/produtos.php?menu_id=3&pro_id=149), acessado em 10/11/2012

Descrição de Série EVA da válvula:

**Série EVA**

**Mod. EVA 15 - Válvula de entrada de água - reta**  
1 entrada e 2 saídas

**Codificação das opções disponíveis**  
(consulte o check-list para amostras e cotações)

<b>EVA</b>	<b>15</b>	<b>1</b>	<b>A</b>	<b>1</b>	<b>0</b>	<b>4</b>	<b>2</b>	<b>0</b>	<b>0</b>	<b>ZZ</b>	<b>11</b>	<b>Z</b>	<b>12</b>
Série	Modelo	Tamanho	Posição da entrada de água em relação ao suporte	Posição da bobina A em relação ao suporte	Posição da bobina B em relação ao suporte	Posição da bobina C em relação ao suporte	Raça da entrada de água	Verão	Terminais	Suporte	Dimensões do suporte de água	Posição do suporte em relação à entrada de água	Regulador de Vazão
		1. 127CA 810 2. 228CA 84010 3. 12V02 4. 24V02 5. 220440100	A	(veja Especificações Técnicas)	(veja Especificações Técnicas)	Z: não aplicável	(veja Especificações Técnicas)	0: normal 1: atóxico	0...sem	(veja Especificações Técnicas) ou ZZ: sem suporte	(veja Especificações Técnicas)	(veja Especificações Técnicas) ou Z: não aplicável	05: 51mm 06: 61mm 12: 121mm 20: 201mm A: 61mm ZZ: sem regulador (quadrado 60x60mm) *verificar dimensões para cada raça da sob consulta

05

06

08

09

\* Reservamos o direito de alterar as especificações sem prévio aviso. Reservamos a responsabilidade por aplicações que não atendam as nossas especificações e não as tenham validadas. Emcol Brato Belfortina S.A.