



CENTRO UNIVERSITÁRIO DE BRASÍLIA -UniCEUB
CURSO DE ENGENHARIA DE COMPUTAÇÃO

CAROLINA DROLHE HOLANDA SILVA
CONTROLE DE CAIXA D'ÁGUA COMPUTADORIZADO

Orientador: Prof. Ms C. Francisco Javier De Obaldía Díaz

BRASÍLIA / DF
2º SEMESTRE/2013

CAROLINA DROLHE HOLANDA SILVA

CONTROLE DE CAIXA D'ÁGUA COMPUTADORIZADO

Monografia apresentada ao Curso de Engenharia da Computação, como requisito parcial para obtenção do grau de Engenheiro de Computação.

Orientador: Prof. MsC Francisco Javier De Obaldía Díaz

BRASÍLIA / DF

2º SEMESTRE/2013

Carolina Drolhe Holanda Silva

CONTROLE DE CAIXA D'ÁGUA COMPUTADORIZADO

Trabalho apresentado ao Centro
Universitário de Brasília
(UniCEUB) como pré-requisito
para a obtenção de Certificado de
Conclusão de Curso de Engenharia
de Computação. Orientador: Prof.
MsC Francisco Javier De Obaldía
Díaz

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.

Prof. Abiezer Amarília Fernandes
Coordenador do Curso

Banca Examinadora:

Prof. Francisco Javier De Obaldía Díaz
Orientador

Prof. Marco Antônio Araújo
Mestre em Ciência da Computação

Prof. Flávio A. Klein
Mestre em Estatística e Métodos Quantitativos

Prof. Sidney Cerqueira B. dos Santos
Doutor

AGRADECIMENTOS

Agradeço a Deus por ter me dado forças, à minha família pela compreensão e apoio. Agradecimento especial ao meu pai, Alan, meu super-herói, à minha mãe Bianca pelo exemplo de mulher guerreira e batalhadora. À Manuela, minha pequena, que mesmo sendo pequena teve a grandeza de compreender o momento que a mamãe está passando. Ao meu Chefe Fernando Quintans por ter acreditado em mim. Finalmente, agradeço ao Jean Carlos pelo companheirismo e compreensão.

CONTEÚDO

LISTA DE FIGURAS	6
LISTA DE SIGLAS	7
RESUMO	8
ABSTRACT	9
1 CAPÍTULO 01 – INTRODUÇÃO.....	10
1.1 Motivação.....	10
1.2 Objetivos	11
1.3 Escopo do Projeto.....	12
1.4 Resultados esperados.....	12
1.5 Estrutura da monografia	13
1.6 Apresentação do Problema.....	13
2 CAPÍTULO 02 – REFERENCIAL TEÓRICO.....	15
2.1 Automação	15
2.2 Delphi 7.....	16
2.2.1 Principais características	16
2.3 Potenciômetro	17
2.4 Porta serial.....	18
2.4.1 RS 232.....	19
2.5 Relé	19
2.6 Micro controlador.....	20
2.7 <i>Software</i>	21
3 CAPÍTULO 03 – PROPOSTA DE SOLUÇÃO E PROTÓTIPO	27
3.1 Apresentação geral do protótipo.....	27
3.2 Descrição das etapas do protótipo (como foi desenvolvido).....	29
4 CAPÍTULO 04 – AMBIENTE DE SIMULAÇÃO E IMPLEMENTAÇÃO DE RESULTADOS	
31	
4.1 Apresentação do ambiente e montagem.....	31
4.2 Descrição da aplicação da solução	32
4.3 Avaliação geral do projeto	32
5 CAPÍTULO 05 – CONCLUSÃO.....	34
5.1 Sugestão de Projetos futuros	34
6 CAPÍTULO 06 – REFERÊNCIAS	35
APÊNDICE: CÓDIGO-FONTE DO PROGRAMA DESENVOLVIDO	36

LISTA DE FIGURAS

Figura 1.1 – Poço Artesiano.....	10
Figura 1.2 – Proposta de Projeto.....	10
Figura 2.1 – Automação Residencial x Automação Industrial.....	15
Figura 2.2 – Tela de interação do <i>software</i>	17
Figura 2.3 – Potenciômetro.....	17
Figura 2.4 – Porta serial RS232.....	19
Figura 2.5 – Relé.....	20
Figura 2.6 – Microcontrolador PIC12F675.....	21
Figura 2.7 – Tela indicando botão “conectar”.....	22
Figura 2.8 – Tela indicando leitura do volume.....	23
Figura 2.9 – Tela indicando botão “Liga/Desliga”.....	26
Figura 3.1 – Projeto.....	27
Figura 3.2 – O protótipo.....	28
Figura 3.3 – Diagrama de atividades.....	32
Figura 4.1 – O protótipo.....	33
Figura 4.2 – Placa eletrônica.....	34

LISTA DE SIGLAS

CTS – *Clear to Send*

DCE – *Data Communication Equipament*

DLL – *Dynamic-link Library*

DTE – *Data Terminal Equipament*

DTR – *Data Terminal Ready*

RAD – *Rapid Aplication Development*

RS-232 – Porta serial utilizada para comunicação de dados.

RTS – *Request to Send*

USB – *Universal Serial Bus*

VCL – *Visual Component Library*

RESUMO

O presente trabalho mostra a solução para controle de caixa d'água abastecida por poço artesiano. O foco foi desenvolver um protótipo, *hardware e software*, onde a bomba d'água seja ligada ou desligada conforme os volumes mínimo e máximo da caixa, respectivamente.

O protótipo é constituído por placa eletrônica que conectada a um computador com software irá enviar comando de ligar ou desligar para a bomba d'água, conforme leitura do volume de água efetuada.

Como esperado, o protótipo obtido demonstrou ser um sistema simples, que permite, de forma automática e parametrizada, ligar e desligar a bomba de abastecimento de água, evitando desperdício e falta d'água.

Palavras-chave: Sustentabilidade, poço artesiano, automação.

ABSTRACT

This study shows the solution to control water tank supplied by artesian wells. The focus was to develop a prototype, hardware and software, where the water pump is on or off according to the minimum and maximum volumes of the tank, respectively.

The prototype is made up of electronic board and connected to a computer with software that will send command to turn on or off the water pump, as reading the volume of water carried.

As expected, the prototype obtained proved to be a simple system that allows for an automatic and parameterized, on and off the water pump, avoiding waste and water shortages.

Keywords: sustainability, artesian well, automation.

1 CAPÍTULO 01 – INTRODUÇÃO

1.1 Motivação

De acordo Censo-Demográfico 1991-2000, do Instituto Brasileiro de Geografia e Estatística – IBGE, entre os anos de 1991 e 2000, Brasília teve um aumento absoluto de 442 mil habitantes, esse aumento populacional desordenado teve início em meados da década de 80 e grande parte dessas pessoas passaram a viver em condomínios. O fenômeno dos condomínios surgiu para atender a demanda habitacional da classe média, que ficou de fora das políticas habitacionais. No fim dos anos 1980, logo após o término da ditadura militar, as equipes de fiscalização do solo desapareceram das ruas e os grileiros aproveitaram para lotear grandes extensões de terras públicas e particulares. Não sendo possível atender às demandas imediatas que surgiram nesses condomínios quanto à distribuição de água e tratamento de esgoto, a alternativa foi utilizar meios naturais de conseguir água potável. Surgia, assim, um grande volume de poços artesianos como o poço mostrado na Figura 1.1.



Figura 1.1 – Poço artesiano

Considerando que a utilização de poço artesiano ocasiona danos irreversíveis ao meio ambiente e que, até o momento, a CAESB (Companhia de Água e Esgoto de Brasília) ainda

não conseguiu implantar a distribuição de água em 100% das residências do Distrito Federal, esse projeto vem contribuir com a melhor utilização da água, além de auxiliar na construção de uma consciência ambiental, buscando otimizar o uso dos poços artesianos e reduzindo o desperdício causado pelo mau uso das bombas d'água, que têm a função de recolher a água dos poços artesianos e armazená-las em caixa d'água. Como é constatado, em muitos casos não existe um sistema automático que desligue a bomba tão logo a caixa de água atinja seu nível máximo, e em grande parte deles, a mesma somente é desligada após intervalos de tempo com a caixa de água transbordando.

A água é uma necessidade básica vital para todos os seres vivos na Terra. Um ser humano pode sobreviver várias semanas sem comida, mas sem água somente alguns dias. O consumo da água é necessário para repor fluidos perdidos através das atividades fisiológicas básicas, como transpiração e urina. Além de utilizar a água para consumo direto, o homem utiliza a água para manter a higiene básica evitando proliferação de doenças.

A necessidade de conscientização quanto à utilização da água é tão importante que é responsabilidade do Distrito Federal, conforme Item IV, Artigo 2º, da Lei Distrital nº 2.725, de 13 de junho de 2001, manter a população constantemente informada sobre a situação quantitativa e qualitativa de recursos hídricos, além de promover ação permanente de educação ambiental e de conscientização sobre a importância da preservação, da conservação e do uso racional dos recursos hídricos.

Diante desse cenário, percebe-se a necessidade de utilizar a tecnologia em favor do meio ambiente. A automação residencial se tornou um aliado do ser humano, tanto para solucionar problemas do dia-a-dia quanto para otimizar e informatizar atividades rotineiras. Esse projeto vem ao encontro desse anseio de informatizar a rotina do ser humano e ajuda-lo na preservação do meio ambiente.

1.2 Objetivos

Construir um protótipo que permita ligar e desligar a bomba de abastecimento de água, utilizada no bombeamento de água a partir de um poço artesiano. Tal desligamento deve ocorrer conforme variação do nível de água na caixa, que quando atingir o limite máximo deverá desligar a bomba e quando atingir o limite mínimo a bomba deve ser ligada.

O objetivo específico é evitar desperdício de água por transbordo da caixa o derramamento pelo cano conhecido como ladrão, já que a bomba será desligada quando a caixa d'água estiver cheia. Assim, também, evitar falta de água, já que a bomba será religada quando o nível de água estiver baixo. O usuário irá parametrizar o limite máximo e mínimo.

1.3 Escopo do Projeto

O projeto consiste em produzir um protótipo composto de hardware e software que simule o ambiente em termos sistêmicos mostrando como seria o funcionamento no ambiente real. O software, desenvolvido como parte deste projeto, recebe a informação do volume de água na caixa d'água e, de acordo com a leitura emite comandos para ligar/desligar o motor. A construção do protótipo reduzido consistindo de caixa de água, com boia, interface hidráulica, com circuito controlador e a bomba, também faz parte do escopo do projeto, como mostrado na Figura 1.2.

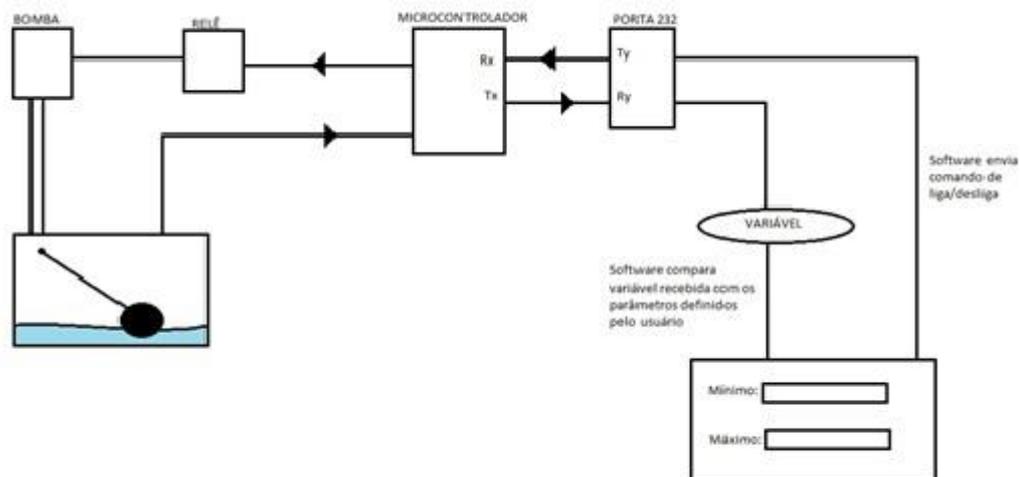


Figura 1. 2 – Proposta de projeto

1.4 Resultados esperados

Espera-se construir um sistema capaz de simular uma caixa d'água que será abastecida por poço artesiano ou similar, onde a boia d'água estará conectada a um sistema que irá detectar o nível da água dentro da caixa e comparará o volume existente com as variáveis inseridas pelo usuário a fim de executar o comando de liga/desliga, conforme necessidade. Evita-se assim o monitoramento humano para execução dessa tarefa.

1.5 Estrutura da monografia

O trabalho está dividido em 6 capítulos, conforme a seguir:

- O Capítulo 1 – trata do introdutório do trabalho, contendo a motivação, a justificativa, os objetivos e escopo, além da estrutura da monografia.
- O Capítulo 2 – contextualiza e apresenta o problema, o impacto da situação atual, danos ambientais e possíveis soluções e a proposição deste trabalho no intuito de contribuir para pelo menos evitar o desperdício de água, com o derramamento de água devido ao esquecimento de desligar a bomba e a caixa vir a transbordar.
- O Capítulo 3 – trata do referencial teórico e metodologia utilizada no desenvolvimento do projeto.
- O Capítulo 4 – apresenta o protótipo do projeto com sua solução de hardware e software, assim como a implementação.
- O Capítulo 5 – apresenta os resultados alcançados, aplicação e avaliação do projeto mostrando que os objetivos foram alcançados.
- E, finalmente, no Capítulo 6 são apresentadas as conclusões e proposta de trabalhos futuros.

1.6 Apresentação do Problema

Esse capítulo tem por objetivo contextualizar e apresentar o problema, o impacto da situação atual, danos ambientais e possíveis soluções e a proposição deste trabalho no intuito de contribuir para pelo menos evitar o desperdício de água, com o derramamento de água devido ao esquecimento de desligar a bomba e a caixa vir a transbordar.

A água encontra-se disponível sob várias formas e é uma das substâncias mais comuns existentes na natureza, cobrindo cerca de 70 por cento da superfície do planeta. Do total de

265 400 trilhões de toneladas de água disponível, somente 0,5 por cento representa água doce explorável sob o ponto de vista tecnológico e econômico, que pode ser extraída dos lagos, rios e aquíferos. É necessário ainda subtrair aquela parcela de água doce que se encontra em locais de difícil acesso ou já muito poluída, restando assim, para utilização direta, apenas 0,003 por cento do volume total de água do planeta. (Braga, 2002)

Do total de água disponível para utilização, de 60 a 80 por cento é constituído de água subterrânea. Partindo desse ponto, fica mais claro entender a grande necessidade de utilização de poços artesianos para o abastecimento de residências. Infelizmente, o uso desenfreado desse recurso gera um desperdício inestimável para toda a população. Além disso, ainda existe a dificuldade de acesso ao abastecimento fornecido pelo órgão público destinado a esse fim, a CAESB (Companhia de Água e Esgoto de Brasília). Apesar de haver promessa de aumento na área de cobertura para fornecimento de água e esgoto, ainda existem muitas regiões que dependem de meios próprios para suprir a necessidade de utilização de água.

A água é uma necessidade básica vital para todos os seres vivos na Terra. O consumo da água é necessário para repor fluidos perdidos através das atividades fisiológicas básicas, como transpiração e urina. Além de utilizar a água para consumo direto, o homem utiliza a água para manter a higiene básica evitando proliferação de doenças.

A necessidade de conscientização quanto à utilização da água é tão importante que é responsabilidade do Distrito Federal, conforme Item IV, Artigo 2º, da Lei Distrital nº 2.725, de 13 de junho de 2001, manter a população constantemente informada sobre a situação quantitativa e qualitativa de recursos hídricos, além de promover ação permanente de educação ambiental e de conscientização sobre a importância da preservação, da conservação e do uso racional dos recursos hídricos.

Assim, surge a ideia de se construir um protótipo que mostre que é possível automatizar o abastecimento de uma caixa d'água que utiliza poço artesiano através da utilização de solução mecânica combinada com uma placa elétrica e comandada por um *software*.

Os próximos capítulos descreverão como o protótipo foi construído e quais tecnologias foram necessárias.

2 CAPÍTULO 02 – REFERENCIAL TEÓRICO

Neste capítulo serão abordados os conceitos e fundamentos técnicos que subsidiaram o desenvolvimento da proposta deste trabalho, desde o conceito de automação até os componentes de *hardware* e *software* necessários à implementação do projeto.

2.1 Automação

Num contexto industrial, pode se definir a automação como a tecnologia que se ocupa da utilização de sistemas mecânicos, eletro-eletrônicos e computacionais na operação e controle da produção. (Pazos, 2002) A Figura 3.1 mostra automação residencial x automação industrial.

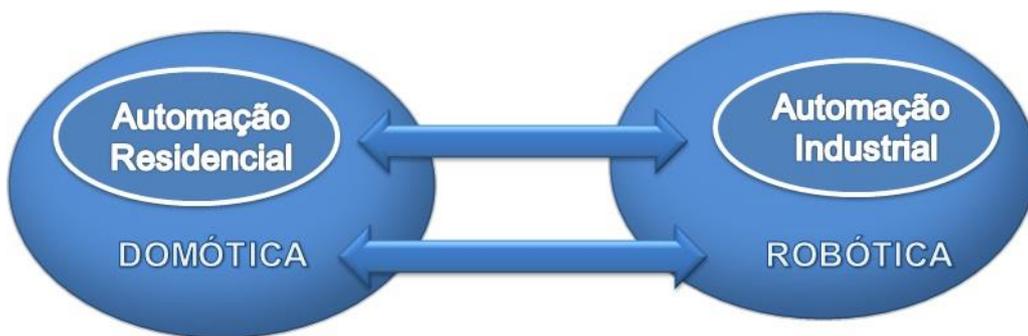


Figura 2. 1 Automação Residencial x Automação Industrial

Há anos a automação vem sendo empregada no uso residencial. Sendo denominada domótica (vem da junção da palavra em latim *domus* – significa “casa” – e robótica), consiste em novas tecnologias que procuram oferecer conforto, praticidade, produtividade, economia e eficiência aos moradores de uma residência. Diante disso, percebe-se um aumento no uso da domótica, substituindo os recursos humanos por recursos tecnológicos (*hardware* e *software*), no intuito de diminuir erros.

2.2 Delphi 7

2.2.1 Principais características

O Delphi é uma ferramenta orientada a objeto, que utiliza o conceito RAD (*Rapid Application Development*), que permite associar, de maneira simples, um elemento da interface e o código da aplicação. Tirando o foco do desenvolvimento da interface e focando no que realmente é necessário, na análise, no projeto e na codificação do sistema. O Delphi apresenta uma biblioteca de componentes, inteiramente desenvolvida em Object Pascal – a VCL (*Visual Component Library*) -, na qual cada componente é representado por uma classe.

Aliando a facilidade do Visual Basic ao poder da linguagem Object Pascal, o Delphi se tornou uma opção de linguagem muito atrativa. Já que o desenvolvedor consegue desenvolver em menos tempo e com um desempenho final melhor.

Com a intenção de ampliar a biblioteca de componentes a Borland disponibiliza o código-fonte da VCL, para que os desenvolvedores possam entender a estrutura hierárquica e codificação, podendo auxiliar na criação de novos componentes.

Uma característica muito atrativa é a portabilidade, o *software* desenvolvido em Delphi não depende de nenhuma dll do Windows, não é necessário se preocupar em qual versão do Windows está sendo instalado. Podendo ser comercializado facilmente, não obrigando o usuário a adaptar seus equipamentos nem a ter instalação de um sistema operacional específico e limitado. Além disso, possui um compilador muito rápido e eficiente. Foram essas características que fizeram com que o Delphi fosse a linguagem de programação escolhida.

Na Figura 3.2 é mostrada a tela desenvolvida para ser usada pelo usuário. Nela é possível verificar se o motor está ligado ou desligado, é possível verificar o volume de água na caixa, é possível alterar os valores máximo e mínimo, além de poder ligar ou desligar o motor manualmente (pode ser utilizado em caso de manutenção).

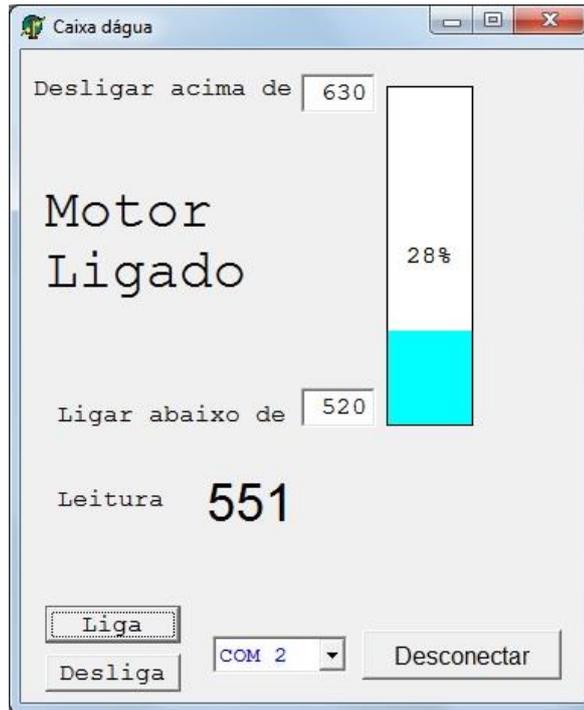


Figura 2. 2 – Tela do software

2.3 Potenciômetro

Um potenciômetro (português europeu) ou potenciômetro (português brasileiro) é um componente eletrônico que possui resistência elétrica ajustável. Geralmente, é um resistor de três terminais onde a conexão central é deslizante e manipulável. Conforme é possível observar na Figura 3.3. Se todos os três terminais são usados, ele atua como um divisor de tensão.

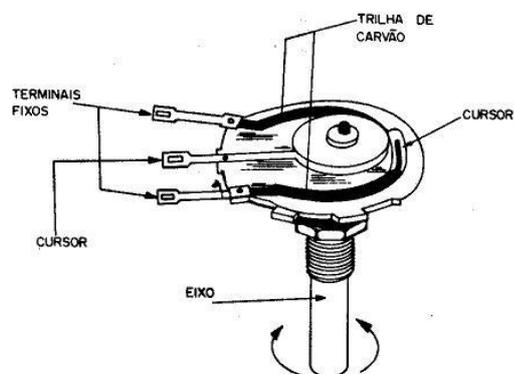


Figura 2. 3 - Potenciômetro

Em relação à curva de resposta em função do ângulo de giro do eixo, existem dois tipos de potenciômetros, os lineares (sufixo B ao final do código) e os logarítmicos (sufixo A ao final do código comercial do valor).

Exemplo de especificação de potenciômetro linear: 50 kB, ou seja, de 50.000 ohms, linear.

Os potenciômetros lineares possuem curva de variação de resistência constante (linear) em relação ao ângulo de giro do eixo. Os potenciômetros logarítmicos, por sua vez, apresentam uma variação de resistência ao ângulo de giro do eixo mais adaptada à curva de resposta de audibilidade do ouvido humano. Considerando um aparelho de som, os potenciômetros lineares são recomendados para uso em controle de tonalidade (graves, médios e agudos) já os logarítmicos são mais recomendados para controles de volume.

No protótipo, foi utilizado potenciômetro linear, potenciômetro 10kB, multivoltas. Já que a relação entre a posição do eixo e o aumento de volume é uma curva linear. O eixo multivoltas foi conectado à haste que sustenta a boia d'água. À medida que o eixo se movimenta, levantando a boia e girando o eixo, o potenciômetro envia ao circuito leitura efetuada em unidade de medida de resistência - kohms, para que fosse possível simular o volume máximo e mínimo da caixa d'água, foi utilizada a faixa entre 420 e 600 kohms, respectivamente.

2.4 Porta serial

A interface serial ou porta serial, também conhecida como RS-232, é uma porta de comunicação utilizada para conectar mouses, impressoras, scanners e outros *hardwares*. Nesse tipo de interface os bits são transferidos em fila, um bit de cada vez.

A comunicação serial entre dispositivos se difundiu por ser um padrão de cabeamento simples e de fácil utilização.

No *software* desenvolvido foi utilizado o componente *Tcomport*, componente que facilita a programação da Porta Serial tanto para enviar ou receber dados, como também para controlar sinais RTS, CTS, DTR.

A função `ComPort1.open` executa a abertura da porta COM que será utilizada no sistema e a `ComPort1.close` desconecta o componente da porta serial.

2.4.1 RS 232

RS-232 (também conhecido por EIA RS-232C ou V.24) é um padrão para troca serial de dados binários entre um DTE (terminal de dados, de *Data Terminal Equipment*) e um DCE (comunicador de dados, de *Data Communication Equipment*). É comumente usado nas portas seriais dos computadores.

No protótipo foi utilizada a porta serial mostrada na Figura 3.4, que tem suas funcionalidades definidas conforme abaixo:

- 2 – RD – Recepção de dados (dados provenientes do circuito eletrônico)
- 3 – TD – Transmissão de dados do computador
- 5 – SG – Terra

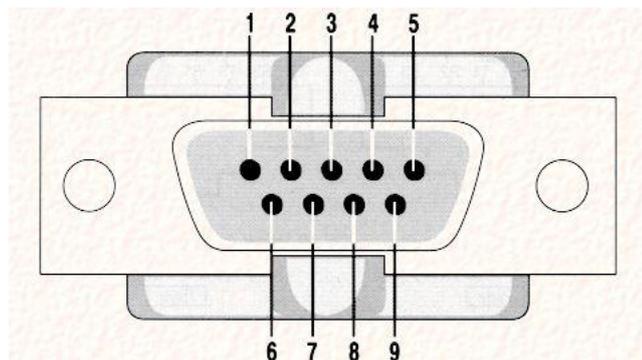


Figura 2. 4 – Porta serial RS232

2.5 Relé

O Relé eletromagnético é constituído basicamente de uma bobina envolvendo um núcleo magnético, cujo entre ferro é formado por uma peça móvel na qual é fixado um contato elétrico que atua sobre um contato fixo, permitindo a continuidade de um circuito elétrico de acionamento do disjuntor. (Mamede, 2005).

Comumente utilizado para ligar e desligar dispositivos, a mudança de estado dos contatos de um relé ocorre apenas quando há presença de tensão na bobina que leva os contatos a movimentarem-se para a posição normal fechado (NF) ou normal abertos (NA) quando esta tensão é retirada.

No protótipo, a função do relé é ligar e desligar a bomba d'água, conforme comando recebido do *software*, que compara a leitura do volume de água aos limites máximo e mínimo, enviando comando de desligar e ligar a bomba. Na Figura 3.5 é possível verificar o relé utilizado no projeto.



Figura 2. 5 - Relé

2.6 Micro controlador

Um microcontrolador (também denominado MCU) é um computador-num-chip, contendo um processador, memória e periféricos de entrada/saída. É um microprocessador que pode ser programado para funções específicas, em contraste com outros microprocessadores de propósito geral (como os utilizados nos computadores). (Wikipedia, 2013).

Na Figura 3.6 é mostrada a imagem do microcontrolador PIC 12F675 utilizado nesse projeto, ele tem as seguintes características:

- Arquitetura RISC;
- Pode operar em até 20Mhz;
- Possui varias fontes de interrupção;
- Pilha de 8 níveis;

- Pode endereçar a memória RAM de forma direta ou indireta
- Possui apenas 35 instruções. Todas executadas em apenas um ciclo de instrução
- Oscilador interno de 4Mhz com 1% de precisão
- Possui memória RAM

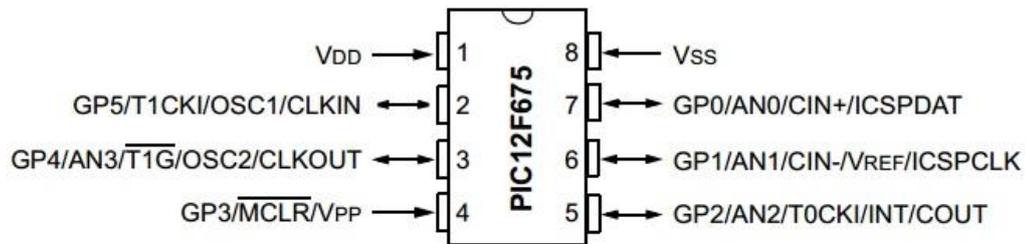


Figura 2. 6 – Microcontrolador PIC12F675

Os conceitos, fundamentos e componentes citados nesse Capítulo são insumos importantes para o desenvolvimento e implementação do protótipo. Nos próximos capítulos serão descritas as etapas de elaboração do projeto.

2.7 Software

Para iniciar o *software* é necessário conectar fisicamente o computador à porta serial através de um cabo com entrada USB, em seguida, verificar em qual porta que o *hardware* foi conectado. Isso é possível no Windows > Painel de controle > Gerenciador de Dispositivos. Ao abrir o programa será necessário selecionar a COM que será utilizada e clicar no botão “Conectar” ao conectar a *label* é alterada para “Desconectar”, durante a execução do programa o usuário pode clicar no botão a qualquer momento para desconectar o computador da placa eletrônica. A Figura 3.5 mostra a tela evidenciando o botão “conectar”.

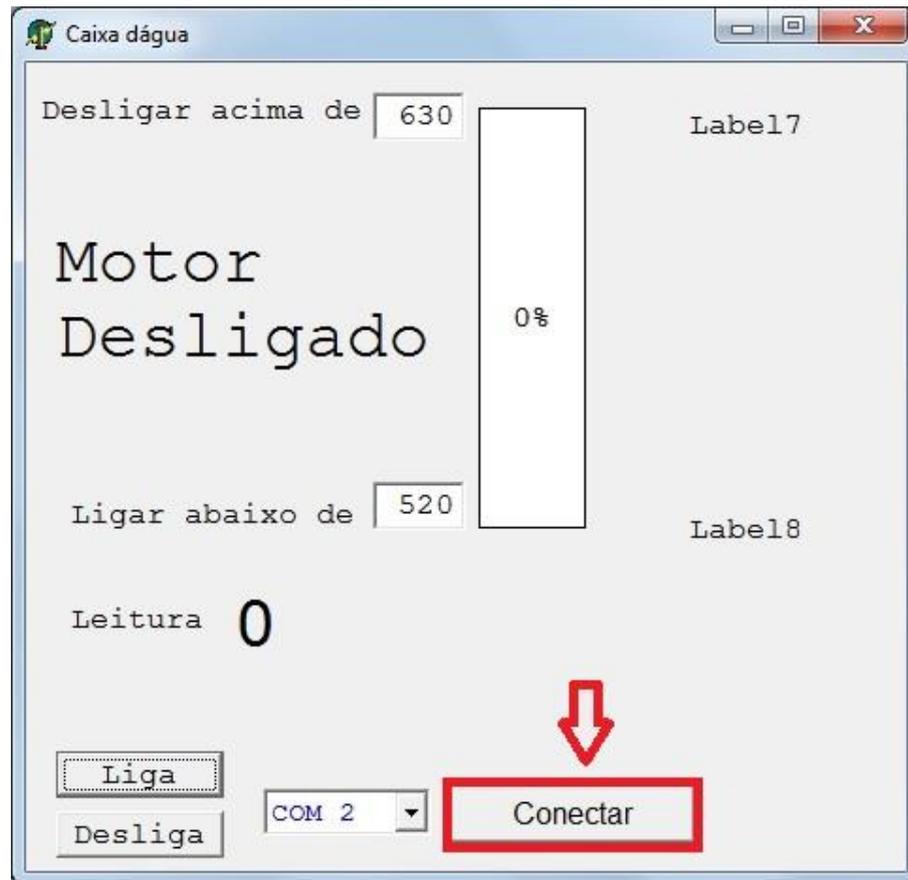


Figura 2. 7 – Tela indicando botão “Conectar”.

Ao clicar no botão, o *software* executa a *procedure* abaixo:

```

procedure TForm1.btnConectarClick(Sender: TObject);
begin
if ComPort1.Connected then
begin
ComPort1.Close;
end
else
begin
ComPort1.Open;
end;
end.

```

Assim, o *software* inicia a execução, alterando a *label* para “Desconectar” a partir da seguinte *procedure*:

```

procedure TForm1.ComPort1Open(Sender: TObject);
begin
  btnConectar.Caption := 'Desconectar';
end;

```

Iniciada a execução, o software começa a efetuar a leitura do volume de água que é feita a cada 800 milissegundos e apresentando ao lado da palavra “leitura”. Além de atualizar o gráfico de acordo com a leitura do volume. Na Figura 3.8 é possível verificar o gráfico e a leitura do volume.

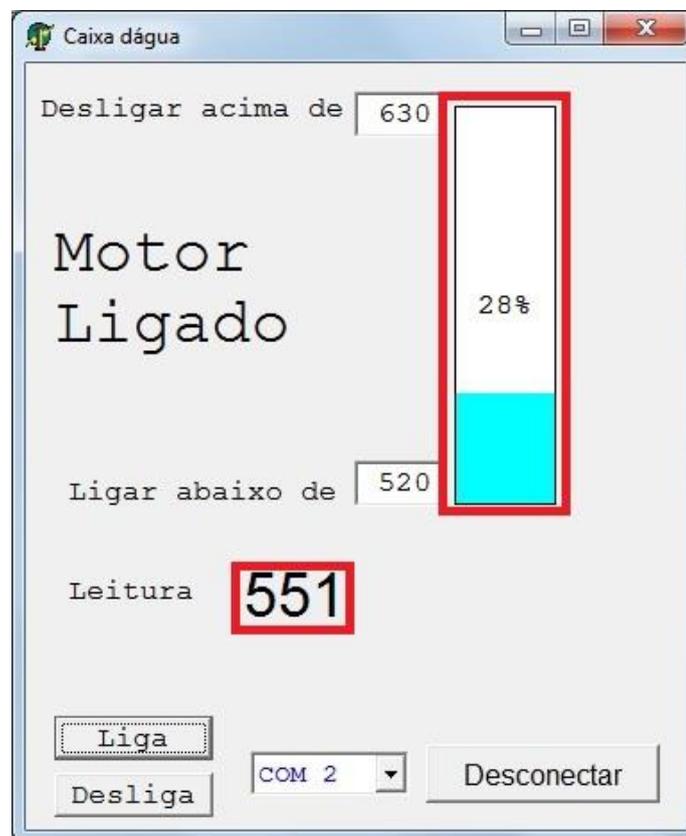


Figura 2. 8 – Tela indicando leitura do volume

Essas duas *labels* são atualizadas a partir da execução da *procedure*:

```

procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
var
  Str: String;
  i : integer;
  aux : string;

```

```

t : char;

begin

ComPort1.ReadStr(Str,count);

edit1.text := edit1.text + str; //edit1 serve para guardar temporariamente o string para ser
usado no label especifico de cada placa

if ComPort1.Connected then

begin

end;

for I := 1 to length(edit1.text) do begin

t := edit1.text[i];

case t of

#0..#31 : { não mostrar } ;

#126..#255: { não mostrar } ;

else

aux := aux + t;

end;

end;

if length(aux) = 4 then // se o string tem 13 caracteres..

begin

leitura := copy(aux,0,4); // retorna os 4 primeiros caracteres na variavel "dispositovo"

label4.caption:=(leitura); // apresenta dispositivo

GAUGE1.MaxValue:= STRTOINT(EDIT3.TEXT); //pega o valor de edit e coloca em
maxvalue de ajuste do grafico

GAUGE1.MinValue:= STRTOINT(EDIT2.TEXT); //pega o valor de edit e coloca em
minvalue de ajuste do grafico

gauge1.Progress:=strtoint(leitura); // pega o valor de leitura e joga no gauge

end;

end;

```

O *software* continua fazendo a leitura e comparando com os valores máximo e mínimo parametrizados pelo usuário, se a leitura for menor que o campo informado pelo usuário, então enviará comando de ligar o relé e mudará a *label* para “Ligado”, se a leitura for maior que o limite máximo informado pelo usuário, então o programa enviará o comando de desligar o relé e mudará a *label* para “Desligado”, seguem abaixo as linhas do código:

```

procedure TForm1.Timer2Timer(Sender: TObject);
begin
  if ComPort1.Connected then
  begin
    if
      edit2.Text >(leitura)
    then
      begin
        comport1.writestr(#73#79#82#76);// envia IOR L
        label1.caption:='Ligado';
      end;
    if
      edit3.Text <(leitura)
    then
      begin
        comport1.writestr(#73#79#82#68);// envia IOR D
        label1.caption:='Desligado';
      end;
    end;
    end;
    end;
  procedure TForm1.Timer3Timer(Sender: TObject);
  begin
    if ComPort1.Connected then
    begin

```

end;

end;

end.

Por último, a qualquer momento o usuário pode ligar ou desligar o motor manualmente, clicando nos botões mostrados na Figura 3.9.

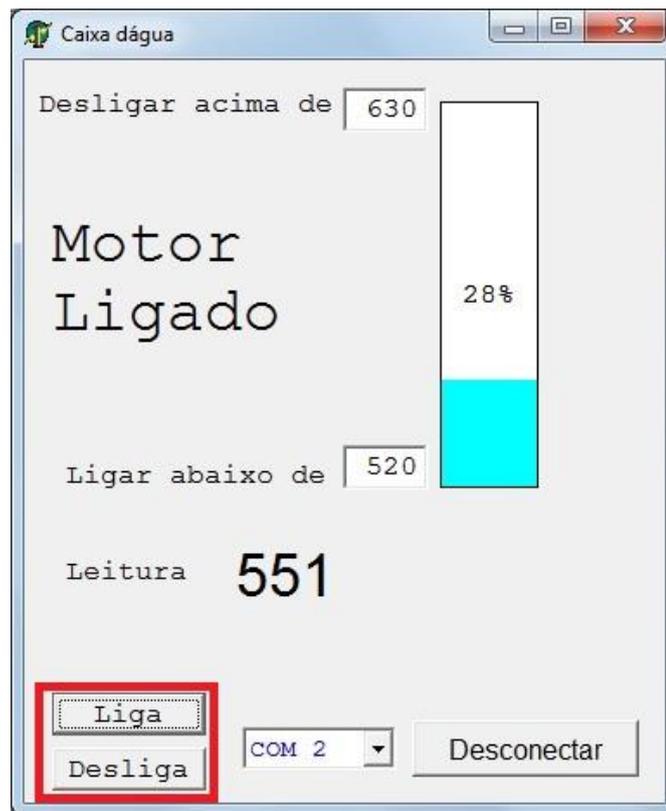


Figura 2. 9 – Tela indicando botão “Liga/Desliga”

3 CAPÍTULO 03 – PROPOSTA DE SOLUÇÃO E PROTÓTIPO

Um sensor na boia da caixa d'água, conectado a um potenciômetro, este por sua vez ligado a um computador, retorna ao computador a cada 800 milisegundos o nível de água. Com estes dados, como entrada de um aplicativo rodando no computador, será mostrado o gráfico de níveis. A mesma interface de sensor também possuirá um relê para ligar e desligar a bomba d'água, sempre que forem atingidos os níveis especificados. O software mostra dois campos para preenchimento com os níveis máximo (desligar a bomba) e mínimo (ligar a bomba). O software controla o relê da bomba ligando ou desligando a mesma de acordo com o resultado da comparação do nível da água na caixa com os níveis preenchidos pelo usuário.

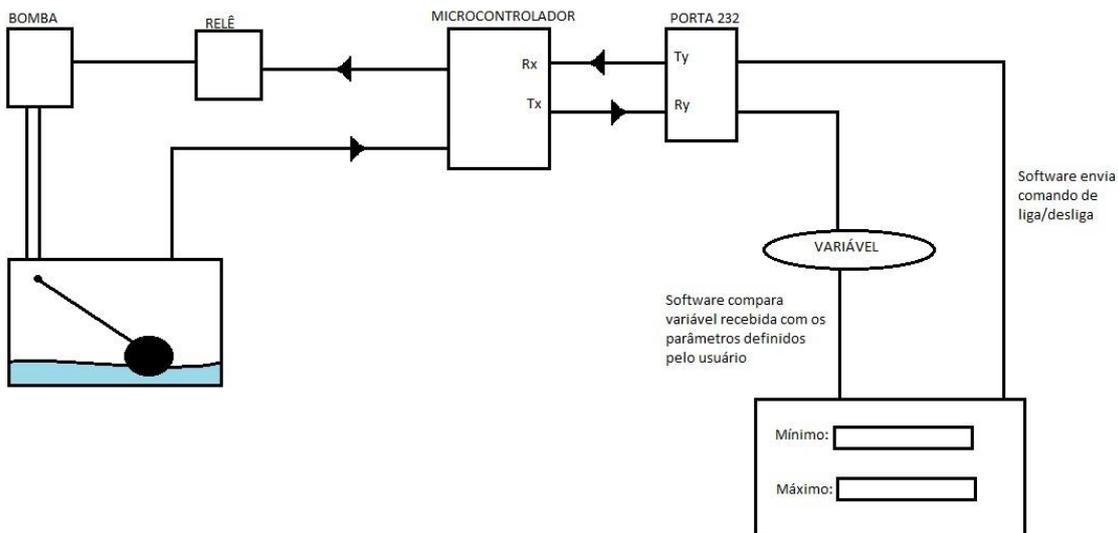


Figura 3.1 - Projeto

3.1 Apresentação geral do protótipo

Para montar o protótipo foi necessária a utilização de uma caixa com tampa, 2 baldes, 1 torneira pequena, um potenciômetro, uma boia, uma bomba injetora de água (utilizada em automóveis), um computador com entrada USB, uma placa eletrônica contendo: um relê, 2 leds, uma porta serial RS-232 e 1 microcontrolador.

Na Figura 4.2 é possível verificar todos esses elementos, conforme descritos abaixo:

- Item 01 – Balde que simula o poço artesiano. Nele está conectado o Item 03;
- Item 02 – Balde que receberá a água que sairá da torneira, simulando o consumo de água;
- Item 03 – Bomba injetora de água substitui a bomba d'água;
- Item 04 – Torneira que simula o consumo de água da caixa d'água;
- Item 05 – Boia d'água, conectada ao potenciômetro, informará o volume de água da caixa;
- Item 06 – Caixa com tampa, simula uma caixa d'água em dimensão reduzida;
- Item 07 – Potenciômetro, conectado à uma haste que segura a boia. É ele que enviará a leitura do volume de água ao *software*.
- Item 08 – Placa eletrônica.



Figura 3. 2 – O protótipo

3.2 Descrição das etapas do protótipo (como foi desenvolvido)

O protótipo surgiu com o intuito de simular o consumo e armazenamento de água, advinda de poço artesiano. Considerando ser impossível transportar um projeto que tivesse uma caixa de água, mesmo a menor disponível no mercado, foi definida a substituição por uma caixa plástica com tampa.

Para que seja possível efetuar a leitura do volume de água existente dentro da caixa plástica foi utilizada uma haste, adaptada para receber uma boia d'água. Essa haste está conectada a um potenciômetro.

O potenciômetro, por sua vez, está conectado a um circuito elétrico. Através de uma porta serial, o circuito envia informações do volume de água existente na caixa (potenciômetro informa valores inteiros entre 0 e 1023, variação de 0V a 5V). Como a boia d'água não efetua um giro de 360° para aferir o volume, foi utilizada uma faixa entre 420 (limite mínimo) a 620 (limite máximo) kohms. Após a leitura, o software faz uma comparação para verificar se a caixa d'água está cheia, ou seja, se a leitura é maior ou igual ao limite máximo especificado pelo usuário, no exemplo foi utilizado o limite máximo de 620 kohms para indicar a posição em que a caixa está cheia. Se a leitura for menor que o limite mínimo, que no exemplo é de 420 kohms, o circuito elétrico receberá o comando para ligar o relé que está conectado à bomba d'água, e iniciar o abastecimento da caixa.

A cada 800 milissegundos é feita a leitura do volume de água, a bomba ficará ligada até que a leitura seja maior ou igual ao volume máximo estabelecido. Ao atingir o limite máximo, o relé receberá o comando para desligar o circuito e a bomba será desligada. Uma torneira situada na parte inferior da caixa d'água, simulará o consumo de água, o sistema continua fazendo leitura do volume de água até que o limite mínimo seja atingido e recomeça o processo, conforme figura 4.3.

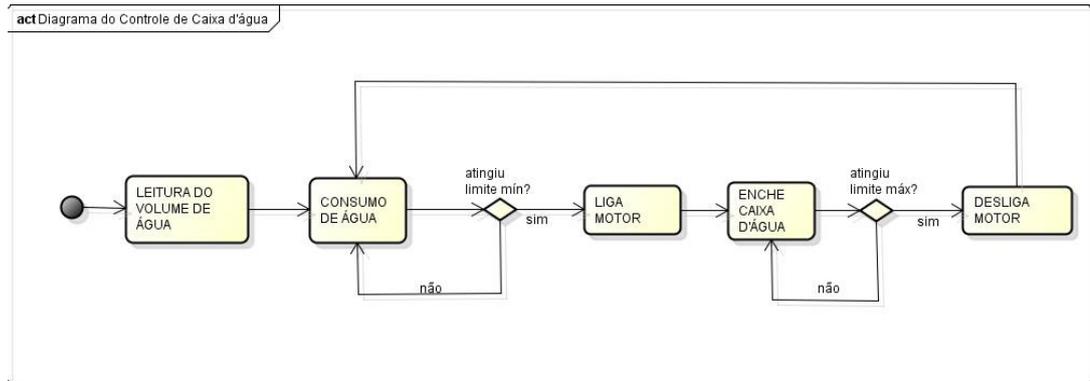


Figura 3.3 – Diagrama de atividades

O *software* continuará executando as tarefas de forma cíclica até que o usuário desconecte a porta serial do computador ou até que não haja energia elétrica para ligar a bomba injetora.

4 CAPÍTULO 04 – AMBIENTE DE SIMULAÇÃO E IMPLEMENTAÇÃO DE RESULTADOS

O objetivo desse capítulo é apresentar os resultados alcançados, aplicação e avaliação do projeto mostrando que os objetivos gerais foram alcançados.

4.1 Apresentação do ambiente e montagem

Para a montagem do protótipo, foi necessário pensar numa forma de simular o abastecimento de uma caixa d'água a partir de um poço artesiano. Considerando ser impossível montar um protótipo utilizando uma caixa d'água (normalmente de 500 litros) e uma bomba d'água (que normalmente pesaria algo em torno de 15 quilogramas), para ser apresentado no ambiente acadêmico. Como desafio do projeto tem-se, então, a questão de criar soluções reais que conseguissem simular esse ambiente, e optou-se por um protótipo ou modelo reduzido. Para simular a caixa d'água foi utilizada uma caixa plástica com tampa, a qual foi acoplada a placa eletrônica e o potenciômetro. Ao potenciômetro foi conectada uma haste adaptada que receberia uma boia d'água, a mesma utilizada nas caixas d'águas tradicionais. Para simular o poço artesiano e a bomba d'água, foram utilizados balde e bomba injetora de água (utilizada em carros), respectivamente. E para simulação do consumo, foi acoplada à caixa com tampa uma pequena torneira, conforme é possível verificar na Figura 5.1.



Figura 4. 1 – Protótipo

Na Figura 5.2 é possível verificar a placa que controla o projeto. À esquerda verifica-se o relé que liga e desliga a bomba d'água. Foram colocados 2 leds, à esquerda o que indica se a bomba está ligada ou desligada, e à direita, indica se o circuito está ligado ou desligado.

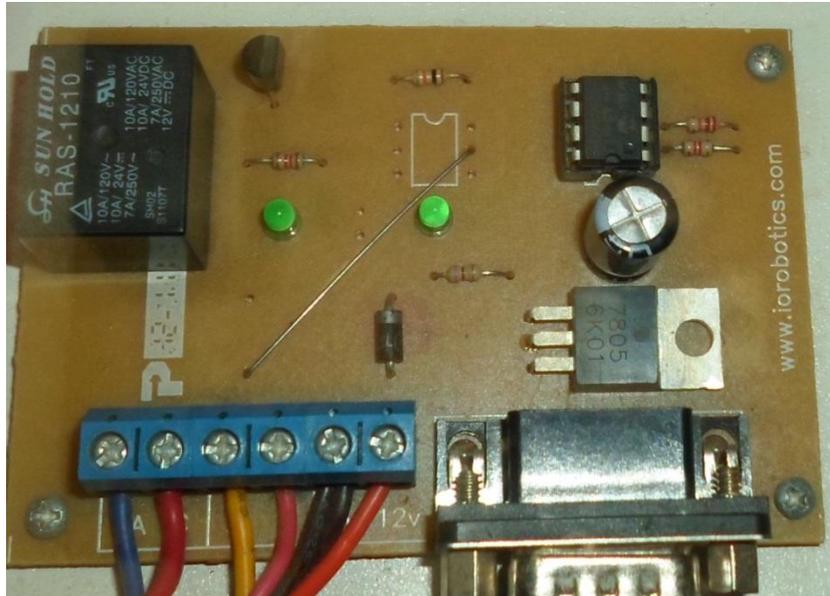


Figura 4. 2 – Placa eletrônica

4.2 Descrição da aplicação da solução

O projeto foi desenvolvido para ser aplicado em residências que sejam abastecidas por poço artesiano, sejam residências individuais ou condomínios. Essa automação substitui a função de uma pessoa que precisaria desligar a bomba de abastecimento em caso de transbordamento de água e ligar a bomba d'água quando percebesse a falta d'água.

Um projeto simples e acessível, de fácil utilização. Pode ser instalado em qualquer computador, sendo a adaptação da boia d'água a haste que fica conectada ao potenciômetro a parte mais complicada da implantação. O sistema é parametrizado conforme as necessidades do usuário.

4.3 Avaliação geral do projeto

Ao longo do desenvolvimento do sistema foi levado em conta a necessidade de um sistema simples e de fácil manuseio. Podendo ser utilizado em residências, condomínios, e pequenas empresas que tenham sua água potável advinda de poço artesiano.

Ao efetuar testes, os limites máximo e mínimo são respeitados e a bomba d'água foi ligada e desligada conforme desejado. Verificou-se que o protótipo atende aos objetivos propostos, no entanto não pode haver parada abrupta do *software*, e caso isso aconteça é necessário reiniciar o sistema operacional. Além disso, em testes foi observado que não é possível inserir parâmetros de configuração enquanto o programa está sendo executado e é necessário esperar o término do processamento para reiniciar com novos parâmetros de entrada, caso o usuário deseje.

A avaliação global do projeto é positiva, tendo o protótipo mostrado um bom desempenho ao longo do seu desenvolvimento. O modelo atende às necessidades levantadas inicialmente, tanto no que tange à sua função de automação quanto ao baixo custo de implementação.

5 CAPÍTULO 05 – CONCLUSÃO

Do que foi abordado pode-se verificar que a sociedade moderna tem exigido de todas o uso racional dos meios que a natureza oferece. A água é um bem finito, ou seja, ela não se reproduz, por isso deve ser tratada com parcimônia, sem que haja desperdícios. A automação é uma importante ferramenta que existe para ajudar a sociedade a solucionar os seus problemas, por isso o trabalho ora concluído apresenta um projeto que pode contribuir no controle e uso devido da água.

Como foi visto, o projeto foi criado com o intuito de mostrar como controlar o uso da água para população desprovida desse bem fornecido pelo Estado e que não pode viver, ou sobreviver sem ela, tendo como única fonte o poço artesiano. Além disso, o projeto trará qualidade de vida a essa população, pois os tirará da “escravidão” gerada pela necessidade de ligar e desligar a bomba que abastece a caixa d’água a partir de um poço artesiano. Dessa forma a automação, mais uma vez, se apresenta como a solução de um problema e como uma ferramenta em benefício do bem estar.

O meio ambiente não foi ignorado neste trabalho. O projeto apresentado caminha lado a lado com o pensamento atual da sociedade. A automação aplicada neste projeto evitará os desperdícios desnecessários provenientes da água que se joga fora quando utilizado o processo manual. Assim, pode-se concluir que este trabalho tem papel muito importante para a sociedade, pois procura evitar a retirada em excesso desse precioso bem existente na natureza, preservando-a.

Outro aspecto importante neste projeto é o seu custo. Para a utilização do projeto há a necessidade de um investimento inicial e, apenas, a manutenção dele ao longo do tempo. Assim, pode-se concluir que depois de um investimento inicial, o que vem depois é o retorno em termos de qualidade de vida e no uso racional dos bens oferecidos pela natureza.

Sobre o trabalho conclui-se que apresenta um projeto que vem ao encontro dos anseios da sociedade e da população que faz uso de poço artesiano e utilização do sistema convencional. Proporcionará o uso racional desse precioso bem – a água, e oferecerá qualidade de vida às populações que não recebem água do Estado.

5.1 Sugestão de Projetos futuros

Finalmente, cabe-se ressaltar a intenção futura de: (i) Projeto que verifique o volume de água consumido; (ii) Projeto que utilize aquecimento solar para aquecer a água que será consumida. (iii) Projeto que associe esse projeto ao projeto do aluno Adams, que controla a temperatura da água em aquários, dando autonomia aos proprietários de aquário quanto à manutenção e troca da água. A automação residencial é um leque muito amplo de possibilidades, sendo possível incrementar o projeto apresentado para que ele possa facilitar ainda mais a vida das pessoas que o utilizam.

6 CAPÍTULO 06 – REFERÊNCIAS

- BRAGA, Benedito et al. Introdução À Engenharia Ambiental. São Paulo: Prentice Hall, 2002.
- LEÃO, Marcelo. Borland Delphi 7 Curso Completo. Rio de Janeiro: Axcel Books do Brasil Editor, 2003.
- LEI Nº 2.725, DE 13 DE JUNHO DE 2001.
- MAMEDE FILHO, João. Manual de equipamentos elétricos. 3ª ed. Rio de Janeiro: LTC, 2005.
- PAZOS, Fernando. Automação de sistemas e robótica. Rio de Janeiro: Axcel Books do Brasil Editora Ltda, 2002.
- <http://www.msseletronicaimports.com.br/arquivos/12f675.pdf>- acessado em 10/06/2013.
- <http://www.dsc.ufcg.edu.br/~sampaio/cursos/2007.1/Graduacao/SI-II/Uml/diagramas/usecases/usecases.htm> - acessado em 03/04/2013
- <http://www.rogercom.com/PortaSerial/ControleAcesso/ControlePag4.htm> - acessado em 08/04/2013
- <http://www.simonsen.br/its/pdf/apostilas/base-tecnica/2/linguagem-de-programacao-1-capitulo-2-ano-informatica.pdf> - acessado em 11/04/2013
- <http://www.agsolve.com.br/noticias/o-perigo-do-uso-indiscriminado-dos-pocos-artesianos-uma-abordagem-hidrica>, acesso em 01/05/2013
- <http://pt.wikipedia.org/wiki/Microcontrolador>, acessado em 02/05/2013
- <http://www.arcelect.com/rs232.htm>, acessado em 02/05/2013
- <http://pt.wikipedia.org/wiki/Potenci%C3%B4metro>, acessado em 01/05/2013
- <http://www.unb.br/noticias/unbagencia/cpmod.php?id=19292>, acessado em 21/05/2013
- http://sraengenharia.blogspot.com.br/2013/01/historico-da-automacao-residencial_10.html, acessado em 17/06/2013.
- http://www.ibge.gov.br/home/estatistica/populacao/tendencia_demografica/analise_resultados/sinopse_censo2000.pdf - acessado em 12/06/2013

APÊNDICE: CÓDIGO-FONTE DO PROGRAMA DESENVOLVIDO

```
unit Serial01;

interface

uses

  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, Buttons, CPort, ExtCtrls, Gauges;

type

  TForm1 = class(TForm)

    ComPort1: TComPort;

    Button1: TButton;

    Button2: TButton;

    Label2: TLabel;

    Label4: TLabel;

    Edit1: TEdit;

    Timer1: TTimer;

    Timer2: TTimer;

    Edit2: TEdit;

    Edit3: TEdit;

    Label3: TLabel;

    Label5: TLabel;

    btnConectar: TSpeedButton;

    cboxPort: TComboBox;

    Gauge1: TGauge;

    Label1: TLabel;

    Label6: TLabel;

    Label7: TLabel;

    Label8: TLabel;
```

```
procedure btnSairClick(Sender: TObject);  
procedure ComPort1Close(Sender: TObject);  
procedure ComPort1Open(Sender: TObject);  
procedure ComPort1RxChar(Sender: TObject; Count: Integer);  
procedure FormCreate(Sender: TObject);  
procedure cboxPortChange(Sender: TObject);  
procedure btnConectarClick(Sender: TObject);  
procedure FormClose(Sender: TObject; var Action: TCloseAction);  
procedure Button1Click(Sender: TObject);  
procedure Button2Click(Sender: TObject);  
procedure Timer1Timer(Sender: TObject);  
procedure Timer2Timer(Sender: TObject);  
procedure Timer3Timer(Sender: TObject);
```

```
private  
{ Private declarations }
```

```
public  
{ Public declarations }
```

```
end;
```

```
var
```

```
Form1: TForm1;
```

```
Z : string;
```

```
leitura: string;
```

```
valor : string;
```

```
CONST
```

```
X:INTEGER=0;

implementation

{$R *.DFM}

procedure TForm1.FormCreate(Sender: TObject);

begin

cboxPort.ItemIndex := Integer(ComPort1.Port);

    label1.caption:='Desligado';

end;

procedure TForm1.btnConectarClick(Sender: TObject);

begin

// Conecta/Desconecta programa da COMMX

//porta conectada a 2400 N,8,1

if ComPort1.Connected then // se a porta conectada ...

begin

ComPort1.Close; // desconecta

end

else // senão

begin

ComPort1.Open; // conecta

end;

end;

procedure TForm1.ComPort1Close(Sender: TObject);

begin

btnConectar.Caption := 'Conectar'; // label do botão

end;

procedure TForm1.ComPort1Open(Sender: TObject);

begin
```

```
btnConectar.Caption := 'Desconectar'; //muda label do botão

end;

procedure TForm1.cboxPortChange(Sender: TObject);

begin

ComPort1.Port := TPortType(cboxPort.ItemIndex); //seleciona a porta

end;

procedure TForm1.btnSairClick(Sender: TObject);

begin

// Encerra a execução do programa

Close;

end;

procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);

begin

// Ao fechar o programa, assegura a desconexão

if ComPort1.Connected then

begin

ComPort1.Close; // fecha a porta antes de encerrar o programa

end;

end;

procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);

var

Str: String;

i : integer;

aux : string;

t : char;

begin

ComPort1.ReadStr(Str,count);
```

```

//Memo1.Text := Memo1.Text + Str; // mostra todos os strings que chegam

edit1.text := edit1.text + str; //edit1 serve para guardar temporariamente o string para ser usado no
label especifico de cada placa

if ComPort1.Connected then

begin

    end;

for I := 1 to length(edit1.text) do begin

t := edit1.text[i];

case t of

    #0..#31 : {não mostrar};

    #126..#255: {não mostrar};

else

    aux := aux + t;

end;

end;

if length(aux) = 4 then // se o string tem 13 caracteres..

begin

leitura := copy(aux,0,4); // retorna os 4 primeiros caracteres na variavel "dispositovo"

label4.caption:=(leitura); // apresenta dispositivo

GAUGE1.MaxValue:= STRTOINT(EDIT3.TEXT); //pega o valor de edit e coloca em maxvalue de
ajuste do grafico

GAUGE1.MinValue:= STRTOINT(EDIT2.TEXT); //pega o valor de edit e coloca em minvalue de
ajuste do grafico

// abaixo só para confirmação

LABEL7.CAPTION:=INTTOSTR(GAUGE1.MAXVALUE); // de max value e coloca em label 7

LABEL8.CAPTION:=INTTOSTR(GAUGE1.MINVALUE); // de minvalue e cololca no label 8

gauge1.Progress:=strtoint(leitura); // pega o valor de leitura e joga no gauge

end;

end;

```

```
procedure TForm1.Button1Click(Sender: TObject);
begin
if ComPort1.Connected then
    begin
comport1.writestr(#73#79#82#76); // envia IOR L comando para ligar relê
        end;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
if ComPort1.Connected then
    begin
comport1.writestr(#73#79#82#68); // envia IOR D Comando para desligar relê
        end;
end;

procedure TForm1.Timer1Timer(Sender: TObject);
begin
edit1.text := ""; // apaga edit1 para dar espaço ao proximo byte
end;

procedure TForm1.Timer2Timer(Sender: TObject);
begin
if ComPort1.Connected then
    begin
if
edit2.Text >(leitura)
then
        begin
comport1.writestr(#73#79#82#76); // envia IOR L
```

```
    label1.caption:='Ligado';  
end;  
if  
edit3.Text <(leitura)  
then  
begin  
comport1.writestr(#73#79#82#68);// envia IOR D  
    label1.caption:='Desligado';  
end;  
    end;  
end;  
procedure TForm1.Timer3Timer(Sender: TObject);  
begin  
if ComPort1.Connected then  
begin  
end;  
end;  
end.
```