



**Centro Universitário de Brasília - UNICEUB**  
**Faculdade de Ciências Exatas e Tecnologia - FAET**  
**Engenharia de computação**

**DANIEL ALVES FERREIRA**

**SISTEMA DE ALARME INTEGRANDO**  
**GPS E CELULAR**

BRASÍLIA - DF  
1º SEMESTRE DE 2010

**DANIEL ALVES FERREIRA**

**SISTEMA DE ALARME INTEGRANDO GPS E CELULAR**

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 O. Diaz

**4 de julho de 2010**

**DANIEL ALVES FERREIRA**

**SISTEMA DE ALARME INTEGRANDO GPS E CELULAR**

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 O. Diaz

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 Amarilia Fernandez  
Coordenador do Curso

**Banca Examinadora:**

---

Prof. MsC. Francisco Javier de O. Diaz  
Orientador

---

Prof. MsC. Fabiano Mariath D'Oliveira  
Mestre em Gestão do Conhecimento e da Tecnologia da Informação

---

Prof. João Marcos Souza Costa  
Especialista em Matemática

---

Prof. MsC. Maria Marony Souza Farias Nascimento  
Mestre em Engenharia Elétrica

Dedico este trabalho a todos que correm atrás de seus sonhos,  
sabendo que nunca é tarde demais para alcançá-los.

## **AGRADECIMENTOS**

Agradeço em primeiro lugar a Deus, que é a razão da minha vida e de minha fé, e que sempre esteve a iluminar meus caminhos.

Agradeço a minha esposa (Andréa) e aos meus filhos (Álex e Wendy), pela compreensão, diante de tantos momentos em que minha ausência foi sentida e mesmo assim continuaram a me apoiar.

Agradeço aos meus pais (João e Audelina) que me ensinaram a ser uma pessoa de bem e buscar meus objetivos com trabalho e honestidade.

Agradeço aos meus irmãos (Dilma, Natanael, Thaysa e Tharley), minha cunhada (Eliane) e minhas sobrinhas (Letícia e Rayssa), que me acompanharam de perto, sempre estando ao meu lado.

Agradeço aos meus amigos e colegas, que me ajudaram e incentivaram, cada uma a sua maneira, durante essa caminhada.

# SUMÁRIO

## Conteúdo

<b>LISTA DE FIGURAS</b> .....	<b>8</b>
<b>LISTA DE TABELAS</b> .....	<b>9</b>
<b>LISTA DE SIGLAS E ABREVIATURAS</b> .....	<b>10</b>
<b>RESUMO</b> .....	<b>11</b>
<b>ABSTRACT</b> .....	<b>12</b>
<b>1. CAPÍTULO Nº 1 - INTRODUÇÃO</b> .....	<b>11</b>
1.1 – IDENTIFICAÇÃO DO PROBLEMA.....	11
1.2 - OBJETIVOS DO TRABALHO.....	11
1.2.1 - <i>Objetivo geral</i> .....	11
1.2.2 - <i>Objetivos Específicos</i> .....	12
1.3 – JUSTIFICATIVA E IMPORTÂNCIA DO TRABALHO.....	12
1.4 – ESCOPO DO TRABALHO.....	12
1.5 – RESULTADOS ESPERADOS.....	14
1.6 – ESTRUTURA DO TRABALHO.....	14
<b>2. CAPÍTULO Nº 2 – APRESENTAÇÃO DO PROBLEMA</b> .....	<b>15</b>
<b>3. CAPÍTULO Nº 3 – BASES METODOLÓGICAS PARA A RESOLUÇÃO DO PROBLEMA</b> .....	<b>17</b>
3.1 - SISTEMA DE POSICIONAMENTO GLOBAL (GPS).....	17
3.2 - COORDENADAS GEOGRÁFICAS.....	18
3.3 – COORDENADAS UTM.....	21
3.4 - DATUM.....	21
3.4.1 – <i>Datum horizontal WGS84</i> .....	21
3.5 – MÓDULO GPS ME-1000RW.....	22
3.5.1 – <i>Protocolo de comunicação do módulo GPS: Padrão NMEA 0183</i> .....	24
3.6 – DISCADORA WESTRON EV300.....	25
3.6.1 – <i>Celular Siemens M50</i> .....	26
3.7 – KIT ACEPIC 16F628A.....	26
3.7.1 - <i>Microcontrolador PIC16F628A</i> .....	28
3.8 – SISTEMA DE COMUNICAÇÃO ENTRE O MICROCONTROLADOR E O MÓDULO GPS.....	29
3.8.1 – <i>Comunicação serial do microcontrolador</i> .....	29
3.8.2 - <i>RS232</i> .....	30
3.8.3 - <i>MAX232</i> .....	30
3.9 – LINGUAGEM DE PROGRAMAÇÃO.....	31
<b>4. CAPÍTULO Nº 4 – MODELO PROPOSTO</b> .....	<b>33</b>
4.1 - APRESENTAÇÃO GERAL DO MODELO PROPOSTO.....	33
4.2 – DESCRIÇÃO DAS ETAPAS DO MODELO.....	34
4.2.1 - <i>Comunicação do microcontrolador com o módulo GPS</i> .....	34
4.2.2 - <i>Discadora e celular GSM</i> .....	34
4.2.3 - <i>Software</i> .....	35
4.3 - DESCRIÇÃO DA IMPLEMENTAÇÃO.....	40
4.3.1 – <i>Montagem do Protótipo</i> .....	40
4.3.2 - <i>Compilação e alteração no código fonte</i> .....	42
4.4 – PROBLEMAS ENCONTRADOS.....	43
<b>5. CAPÍTULO Nº 5 - APLICAÇÃO E DEMONSTRAÇÃO PRÁTICA DO MODELO PROPOSTO</b> .....	<b>45</b>
5.1 - APRESENTAÇÃO DA ÁREA DE APLICAÇÃO DO MODELO.....	45
5.2 – DESCRIÇÃO DA APLICAÇÃO DO MODELO.....	45
5.3 – AVALIAÇÃO GLOBAL DO MODELO.....	47
5.4 – VALIDAÇÃO.....	47

<b>6. CAPÍTULO Nº 6 - CONCLUSÃO .....</b>	<b>54</b>
6.1 - CONCLUSÕES .....	54
6.2 - SUGESTÕES PARA TRABALHOS FUTUROS .....	54
6.2.1 - <i>Envio das coordenadas para rastreamento</i> .....	55
6.2.2 – <i>Interação com módulos de bloqueio do veículo</i> .....	55
6.2.3 – <i>Instalação de um sistema de escuta ou gravação</i> .....	55
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>56</b>
<b>APÊNDICE .....</b>	<b>58</b>
CÓDIGO FONTE DO PROGRAMA .....	58
<b>ANEXOS .....</b>	<b>62</b>
BIBLIOTECA <i>RS232.C</i> .....	62

## LISTA DE FIGURAS

Figura 1.1 - Visão geral do projeto .....	13
Figura 3.1 - Constelação de Satélites .....	17
Figura 3.2 - Latitude e Longitude.....	18
Figura 3.3 - Imagem do <i>Google Earth</i> : Bloco 07 .....	19
Figura 3.4 - Módulo ME-1000RW .....	22
Figura 3.5 - Ligações do ME-1000RW .....	22
Figura 3.6 - Discadora Westron e Celular Siemens M50 .....	26
Figura 3.7 - Kit de Desenvolvimento ACEPIC 18 e 28 pinos.....	27
Figura 3.8 - Pinagem do PIC16F628A .....	28
Figura 3.9 - Conector DB9 .....	30
Figura 3.10 - Ligação com o MAX232 .....	31
Figura 4.1 – Fluxograma do monitoramento das coordenadas .....	36
Figura 4.2 – Trecho do código para captura da latitude e longitude pela porta RS232 .....	37
Figura 4.3 - Trecho do código para comparação das coordenadas .....	39
Figura 4.4 – Port A de expansão do Kit de desenvolvimento .....	40
Figura 4.5 – Detalhes das ligações no Kit de desenvolvimento .....	41
Figura 4.6 – Ligação do celular com a discadora .....	41
Figura 4.7 – Tela do PIC C Compiler .....	42
Figura 4.8 – Tela de gravação do ICProg .....	43
Figura 5.1 - Conector para alimentação do modelo .....	48
Figura 5.2 – Acondicionamento dos componentes do modelo no veículo.....	48
Figura 5.3 – Tela inicial do modelo.....	49
Figura 5.4 – Tela informando que está fixando as coordenadas.....	49
Figura 5.5 – Tela informando as coordenadas que foram fixadas(sem os separadores) .....	49
Figura 5.6 – Tela informando que está procurando coordenadas válidas .....	50
Figura 5.7 – Coordenada capturada para comparação.....	50
Figura 5.8 – Tela informando a coordenada depois do deslocamento.....	51
Figura 5.9 – Tela informando que o movimento foi detectado.....	51
Figura 5.10 – Tela do celular realizando a ligação para o dono do veículo.....	51
Figura 5.11 – Tela do LCD avisando que está realizando a ligação. ....	52
Figura 5.12 – Celular do dono do veículo recebendo a ligação.....	52
Figura 5.13 – Tela mostrando o número da ligação que está sendo realizada .....	53
Figura 5.14 – Tela informando que já foram realizadas 5 ligações e que está saindo do sistema.....	53



## LISTA DE TABELAS

Tabela 3.1 – Comparação entre as coordenadas com 1 segundo de variação .....	20
Tabela 3.2 - Especificações técnicas do ME-1000RW .....	23
Tabela 3.3 - Significado de cada campo da linha \$GPRMC .....	25
Tabela 3.4 - Designação dos pinos do DB9 .....	30
Tabela 4.1 – Custo de Componentes.....	33
Tabela 5.1 – Consumo máximo dos componentes utilizados .....	46

## LISTA DE SIGLAS E ABREVIATURAS

ASCII: Acrônimo do inglês – *American Standart Code for Interchange of Information* ,  
Código Padrão Americano para Intercambio de Informação.

GPS: Acrônimo do inglês – *Global Positioning System*, Sistema de Posicionamento Global.

GSM: Acrônimo do inglês – *Global System for Mobile Communications*, Sistema Global  
para Comunicação Móvel.

Km: **Kilômetro** – equivalente a 1000 metros.

LVTTL: Acrônimo do inglês - *Low Voltage Transistor-to-Transistor Logic*, Lógica  
Transistor-Transistor Baixa Tensão

LED: Acrônimo do inglês - *Light Emitting Diode*, Diodo Emissor de Luz

NASA: Acrônimo do inglês - *National Aeronautics and Space Administration*,  
Administração Nacional do Espaço e da Aeronáutica, também conhecida como Agência  
Espacial Americana.

NMEA: Acrônimo do inglês – *National Marine Electronics Association*, Associação  
Nacional Eletrônica da Marinha.

RAM: Acrônimo do inglês – *Random Access Memory*, Memória Apenas de Leitura.

ROM: Acrônimo do inglês – *Read Only Memory*, Memória de Acesso Aleatório.

SIM: Acrônimo do inglês – *Subscriber Identity Module*, Módulo de Identificação do  
Assinante.

SMS: Acrônimo do inglês – *Short Message Service*, Serviço de Mensagens Curtas.

TTL: Acrônimo do inglês - *Transistor-to-Transistor Logic*, Lógica Transistor-Transistor.

## RESUMO

Neste trabalho é apresentado o projeto de um sistema que integra um microcontrolador a um módulo GPS e um telefone celular. Trata-se de um sistema de alerta que avisa ao proprietário de um veículo quando o mesmo entrar em movimento não autorizado, através de uma ligação para o número de telefone previamente armazenado no sistema. Se o movimento não autorizado acontecer, o proprietário é avisado para que possa tomar as providências imediatamente após o ocorrido. O sistema desenvolvido faz tudo de forma automática, capturando as coordenadas, analisando e fazendo a ligação se necessário. O alerta através de uma ligação telefônica é feito imediatamente após ser detectado o movimento, aumentando as chances de o proprietário reaver o seu veículo. O proprietário pressionará um botão ao chegar ao local onde deixará o seu carro estacionado e a partir daí o sistema começa a funcionar automaticamente, realizando as funções de captura das coordenadas, análise das mesmas e realização de ligação telefônica. No modelo aqui apresentado, o alerta é realizado a partir de uma movimentação de 15 metros em função da precisão do GPS aqui utilizado.

**Palavras-chave:** microcontrolador, módulo GPS, módulo GSM, alerta, veículo.

## **ABSTRACT**

This paper presents the design of a system that integrates a microcontroller with a GPS module and a cell phone. This is a warning system that notifies the owner of a vehicle when it enters into unauthorized via a link to the telephone number previously stored in the system. If the unauthorized movement occurs, the owner is notified so that you can let to it immediately after it happened. The developed system does everything automatically, capturing the coordinates, analyzing and making the connection if necessary. The warning by a telephone call is made immediately after being detected the movement, increasing the chances of the owner get his vehicle. The owner will press a button to get to where they leave their car parked and since then the system begins to operate automatically, performing the functions of the coordinates of capture, analyze and carry out the same phone call. In the model presented here, the alert is made from a drive fifteen meters depending on the accuracy of GPS used here.

**Keywords:** microcontroller, GPS module, GSM module, alert, vehicle.

## **1. CAPÍTULO Nº 1 - INTRODUÇÃO**

Nas próximas seções deste capítulo serão apresentados os pontos iniciais deste trabalho, mostrando o problema abordado e seu objetivo. Traz também as justificativas sobre a importância deste trabalho, bem como os resultados esperados e a estrutura adotada na monografia.

### **1.1 – Identificação do Problema**

Uma das preocupações dos donos de veículos é com o mesmo, que ao ficar estacionado, não está isento de ações que visem o deslocamento do mesmo do local, quer seja por meliantes ou por autoridade pública. Em ambos os casos, para o projeto aqui apresentado, considera-se que o deslocamento foi não autorizado pelo proprietário do veículo.

Assim, o dono só fica sabendo que o seu veículo foi roubado ou entrou em movimento não autorizado, quando volta ao local onde o deixou. Seria possível o dono ser avisado que o seu carro entrou em movimento não autorizado, assim que o evento ocorresse, através de uma chamada para o seu celular?

### **1.2 - Objetivos do Trabalho**

#### **1.2.1 - Objetivo geral**

Desenvolver um sistema integrando um microcontrolador, GPS e telefone celular, que capture e analise as coordenadas geográficas e identifique se houve alteração, para determinar se o veículo entrou em movimento. Se o movimento for detectado, ligar para o celular do proprietário para alertá-lo.

### 1.2.2 - Objetivos Específicos

Criar um software que captura as coordenadas do GPS através da porta serial do microcontrolador, separa apenas as coordenadas geográficas adquiridas no padrão NMEA e compara se houve alteração acima da tolerância. Se houver alteração fazer uma ligação para o proprietário para alertá-lo que o veículo entrou em movimento.

Implementar um protótipo com o microcontrolador, o GPS e o telefone celular, mostrando através do display, as funções e ações do sistema implementado.

### 1.3 – Justificativa e Importância do Trabalho

Se as pessoas fossem avisadas de imediato que o seu veículo foi furtado, poderiam tomar providências para reaver o veículo. Então as pessoas deveriam ter um sistema instalado em seus veículos para que fossem avisadas assim que o seu veículo entrasse em movimento não autorizado. Isso seria muito importante para o proprietário, que saberia que o seu veículo foi furtado ou rebocado assim que o evento acontecesse, podendo tomar imediatamente as providências.

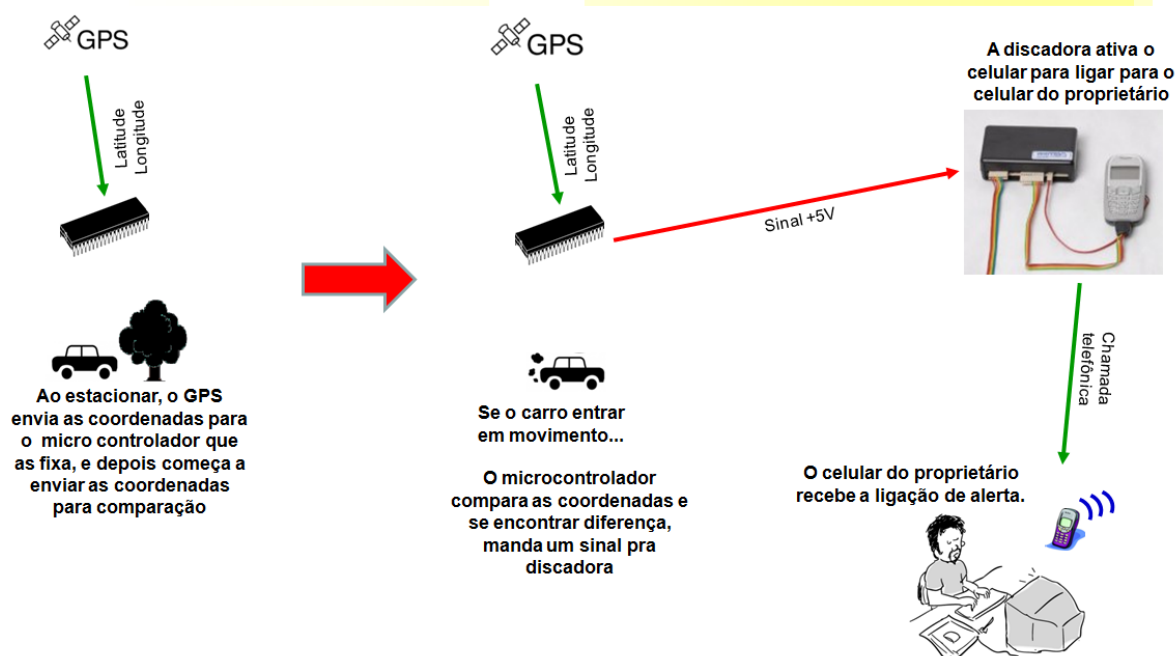
Isso além de trazer tranquilidade para o proprietário, inibiria o furto, pois o ladrão ficaria mais temeroso ao roubar um carro, pois a polícia poderia ser avisada imediatamente e tomar as providências necessárias para a procura desse veículo.

### 1.4 – Escopo do Trabalho

Nesse projeto é implementado um sistema que captura as coordenadas do GPS. Este por sua vez, conectado a dispositivo micro-controlador, envia estas coordenadas, que são analisadas em seguida. Assim que o sistema for acionado, o microcontrolador captura as coordenadas e as armazena para comparação. A partir daí, ele captura as coordenadas e as compara com as coordenadas que foram armazenadas para detectar se houve alteração acima da tolerância estabelecida, que em caso positivo, o sistema fará uma ligação para o

proprietário do veículo, alertando-o sobre o ocorrido. Na figura 1.1 é mostrada uma visão geral dessa solução.

## Visão Geral do Projeto



**Figura 1.1 - Visão geral do projeto**  
(Fonte: O autor)

O sistema apenas armazena as coordenadas que são utilizadas para fazer a comparação entre as coordenadas que ele fixou e as coordenadas atuais, não armazenando as coordenadas para uma futura consulta ou monitoramento.

O objetivo desse sistema não é enviar as coordenadas para rastreamento, e sim enviar uma alerta para o proprietário do veículo, visto que a aplicação focará a comunicação com o módulo GPS e o tratamento das informações capturadas do mesmo. O alerta será enviado através de uma ligação feita através do celular integrado ao sistema.

O sistema não vai monitorar o veículo, apenas avisar através de uma ligação se as coordenadas que foram fixadas se alteraram e avisar que o veículo entrou em movimento não autorizado.

## **1.5 – Resultados Esperados**

Criar um protótipo que integre o módulo GPS e um telefone celular, e que consiga capturar pela porta serial as coordenadas vindas do módulo GPS, analisando as alterações e fazendo uma ligação usando o telefone celular ligado a uma discadora que será acionada através de um sinal positivo enviado pelo microcontrolador. Espera-se que as coordenadas capturadas do módulo GPS realmente mostrem com exatidão, dentro da precisão do equipamento, a posição em que se encontra e que a discadora e o celular faça a ligação para o telefone do proprietário quando necessário.

## **1.6 – Estrutura do Trabalho**

Capítulo 1 – Introdução: Neste capítulo estão apresentadas as idéias básicas sobre o projeto com os objetivos, abrangência e sua organização.

Capítulo 2 – Apresentação do Problema: Neste capítulo é detalhado o problema, mostrando dados estatísticos e situações que exemplificam a necessidade de uma solução. Mostra também a relação entre o problema e a solução apresentada nesse projeto.

Capítulo 3 – Bases metodológicas para resolução do problema: Contém aplicação dos conteúdos vistos nas disciplinas pertinentes ou em outras publicações, visando a resolução do problema. Descreve as técnicas, métodos, metodologias e outras ferramentas que serão utilizadas na formulação do modelo de resolução do problema.

Capítulo 4 – Modelo proposto: Contém toda a parte de implementação do projeto, mostrando os passos necessários para obter a resolução do problema de pesquisa, entre outros detalhes.

Capítulo 5 – Aplicação do modelo proposto: Neste capítulo podem ser verificados os resultados sobre a realização do projeto, mostrando sua implementação prática, simulação e testes.

Capítulo 6 – Conclusões: Aqui estão descritas as conclusões retiradas sobre o projeto desenvolvido, se os resultados foram satisfatórios, mostrando as vantagens e limites do projeto, entre outros.



## 2. CAPÍTULO Nº 2 – APRESENTAÇÃO DO PROBLEMA

Com o crescimento das cidades, cada vez mais a segurança é uma preocupação da população de forma geral. Com os constantes roubos de veículos, as pessoas não se sentem seguras em deixar seus carros estacionados em lugares públicos, nem mesmo com os sistemas de trava e alarmes instalados nos veículos.

Pesquisa da policia civil revelam que mais de 50% dos veículos roubados são recuperados hoje em dia, mas a velocidade em se tomar providências é muito importante, já que em menos de 24 horas muitos carros já estão depenados, em muitos casos. O roubo de camionetes diminui 90%, devido a maioria de elas possuírem sistema de monitoramento, fazendo com que os ladrões as abandonem logo após o furto. (Regis Alex - Tribuna do Norte, 2008).

Muitos carros são recuperados por seus donos, mas o tempo de aviso para que as autoridades tomem providências é muito importante. A maioria dos crimes acontece entre 18h e 6h, fazendo com que o dono só perceba o furto de manhã. 70% dos crimes cometidos são furtos, quando a vítima não está presente, e na maioria das vezes quando estão na aula ou trabalho, onde ficam várias horas até voltar ao seu carro, só percebendo o desaparecimento muito tempo depois, diminuindo a eficácia da tomada de providências. (Pedro Ferreira - Estado de Minas, 2009)

Quando se estaciona o carro e vai trabalhar, estudar ou fazer alguma atividade em que ficará um longo tempo longe do seu carro, no tempo em que se fica distante, ele fica sem nenhum tipo de vigilância. Mesmo que o seu carro tenha um alarme ou tranca, existe a possibilidade de furto; e vindo a ser furtado, só se fica sabendo que o sinistro aconteceu, quando retorna ao local onde foi estacionado o carro, o que em alguns casos podem durar horas.

Sabendo que os primeiros momentos depois que acontece o roubo são os mais importantes para que se avise as autoridades, para tomar providências no sentido de tentar reaver o bem. Acontece que na maioria dos casos, só se toma conhecimento do ocorrido muito tempo depois, diminuindo ou mesmo anulando possíveis atitudes a serem tomadas. Se ficasse sabendo de imediato do ocorrido, as providências também seriam tomadas muito mais rápidas e imediatas, aumentando a eficácia dos meios a serem utilizados.

A maioria dos alarmes emite uma sirene como um tipo de alarme para assustar o infrator e fazê-lo desistir do feito, mas os infratores desligam essas sirenes de imediato e até mesmo o alarme do carro, conseguindo roubá-lo em poucos instantes. E mesmo que o alarme tocasse durante algum tempo, dependendo do local onde você deixar seu carro, nem sempre terá uma pessoa passando ou mesmo próximo ao local, o que tornará a sirene sem efeito.

Existem no mercado, empresas que fazem o serviço de rastreamento e monitoramento do seu veículo, avisando de qualquer movimento não autorizado, e até mesmo bloqueando funções do veículo, informando a posição atual em que o veículo se encontra. O problema é que o cliente tem que pagar mensalidades pelo serviço que inclui o monitoramento, suporte e assistência e muitas pessoas prefeririam adquirir um sistema que não precisasse de mensalidades e mesmo que não tivesse todas as funcionalidades que estas empresas oferecem, pudessem ser avisados de movimentos não autorizados para tomada de providências.

Este projeto propõe uma solução alternativa do problema com um sistema integrando GPS e celular, que avisa o dono através de uma ligação telefônica se a posição do seu carro se alterar, sendo uma solução barata, pois o celular poderá fazer uma ligação para o dono a cobrar, e este nem precisará atender a ligação, já sabendo com isso que o seu veículo está em movimento. O celular integrado ao sistema só precisará ser mantido com as recargas mínimas para não perder a linha, trazendo um gasto baixo para o proprietário, trazendo uma vantagem interessante no sentido de não precisar pagar uma mensalidade para uma empresa que forneça esse tipo de serviço.

No caso da aquisição de um modelo como este, o uso de componentes como um módulo GPS e um celular torna o modelo um produto também de baixo custo para aquisição, já que estes componentes têm o valor mais acessível em relação ao produto oferecido pelas empresas de monitoramento. Estes componentes são facilmente encontrados para a compra, tanto para importação como também em distribuidores nacionais.

Este problema apresentado neste capítulo pode ser resolvido de forma muito eficiente pelo modelo proposto, e com um baixo custo em relação aos produtos existentes no mercado, trazendo uma solução acessível a empresas e pessoas físicas.

### 3. CAPÍTULO Nº 3 – BASES METODOLÓGICAS PARA A RESOLUÇÃO DO PROBLEMA

Nas próximas seções deste capítulo, são apresentadas as tecnologias e métodos referenciais aplicadas no desenvolvimento desta monografia. O Desenvolvimento e implementação do projeto fazem referência ao uso do GPS na obtenção das coordenadas geográficas, da integração com módulo GSM e microcontrolador PIC, sendo necessária a integração desses componentes, a qual deve seguir especificações que envolvem interfaces físicas, protocolos e desenvolvimento de software. Estes aspectos, quanto ao funcionamento de cada componente individualmente são expostos a seguir.

#### 3.1 - Sistema de Posicionamento Global (GPS)

O GPS tem permitido uma variedade de aplicações civis nos últimos anos. Criado pelo Departamento de Defesa Norte Americano, na década de 70 para atender interesses militares, foi muito usado pelos Estados Unidos da América, principalmente na Guerra do Golfo, em 1991. Mas o grande potencial desse sistema no uso das mais diferentes aplicações que necessitem de um posicionamento rápido e preciso fez com que ele fosse explorado nas mais diversas áreas.

O sistema é composto por uma coleção de 24 satélites distribuídos em seis órbitas diferentes, distantes da terra aproximadamente a 20000 km de altitude. As órbitas foram calculadas para que a qualquer momento, pelo menos 4 satélites sejam visíveis em qualquer parte do globo terrestre, como está sendo exemplificado na figura 3.1. Atualmente a NASA divulga que essa constelação consta de 32 satélites, com alguns deles sendo uma reserva. (Hasegawa, Galo, Monico, & Imai, 1999)



**Figura 3.1 - Constelação de Satélites**  
(Fonte: Hasegawa, Galo, Monico, & Imai, 1999)

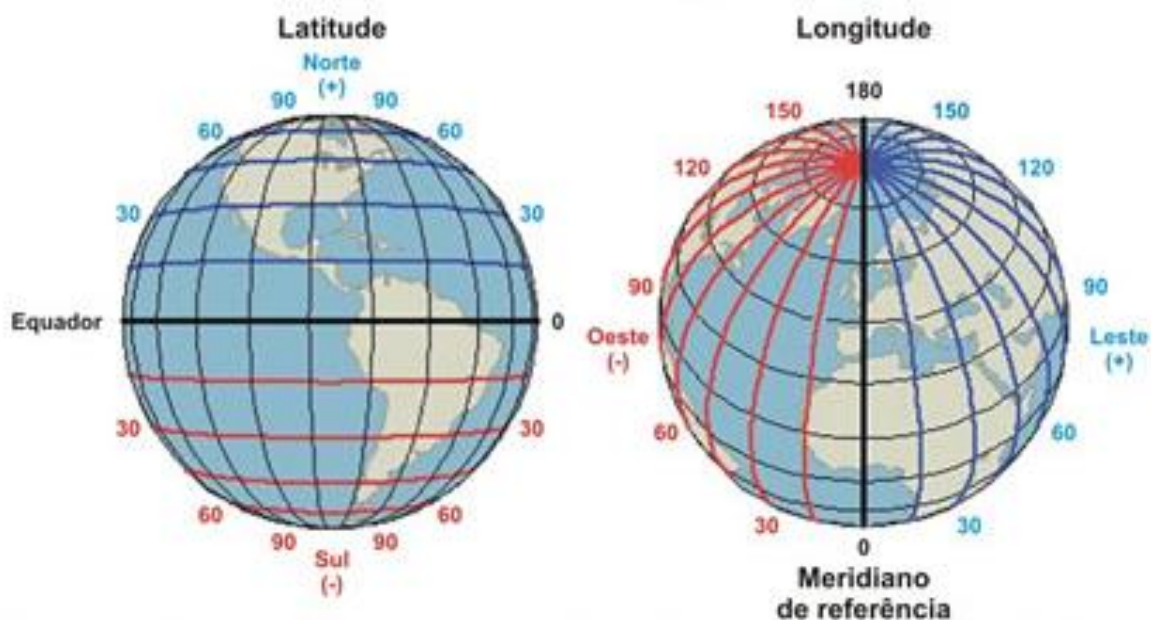
### 3.2 - Coordenadas Geográficas

Coordenadas geográficas são linhas imaginárias verticais e horizontais, que dividem a terra em graus, minutos e segundos, tornando possível definir qualquer ponto no globo terrestre. As latitudes são as linhas paralelas ao Equador, que é o ponto 0° ou ponto de origem. As longitudes são as linhas paralelas ao meridiano de Greenwich que é o ponto 0° ou ponto de origem.

A latitude é dividida em linhas paralelas, que começam no ponto de origem, que é o Equador, como podemos ver na figura 3.2, e seguem de 0° a 90°, nos dois sentidos (N=Norte e S=Sul). A longitude é dividida em meridiano que começam no ponto de origem, o meridiano de Greenwich, e segue de 0° a 180° nos dois sentidos (L=Leste e O=Oeste).

Tanto na latitude como na longitude as linhas são infinitas, sendo traçados de grau em grau, minuto em minuto e segundo em segundo ou fração deste.

Os pontos na superfície terrestre podem ser definidos pelo cruzamento da latitude e longitude.

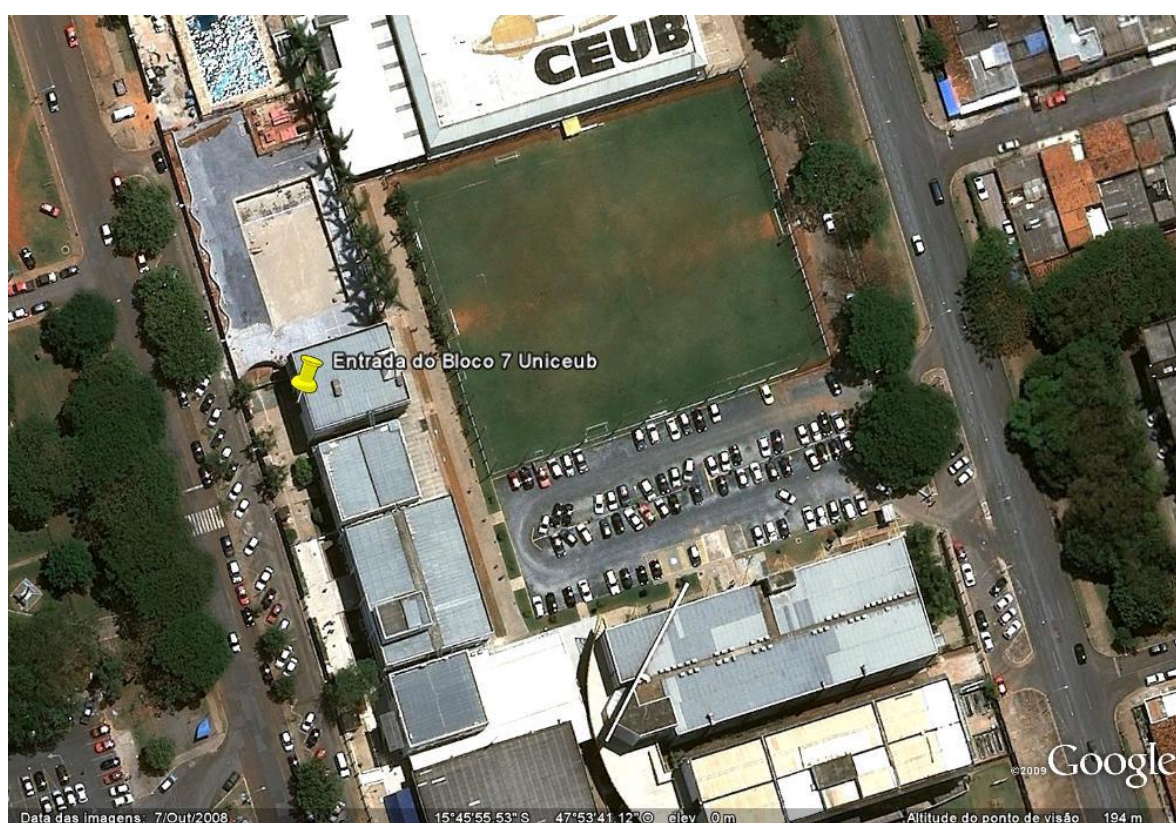


**Figura 3.2 - Latitude e Longitude**  
(fonte: Amarante, 2008)

O equador divide a terra ao meio, sendo a metade do caminho entre os pólos, dividindo a terra em dois hemisférios: Norte ou Setentrional e Sul ou Meridional. Já o meridiano de Greenwich, foi definido tendo como referência o meridiano que passa pelo

Observatório Astronômico Real Inglês, na cidade de Greenwich, próxima a Londres, Inglaterra. (Amarante, 2008)

Para exemplificar a forma que as coordenadas geográficas são apresentadas, vamos analisar as coordenadas tiradas a partir do *Google Earth*<sup>1</sup>, como se pode observar na figura 3.3, da entrada do bloco 7 do UniCeub. As coordenadas geográficas podem ser mostradas das seguintes formas: latitude 15°45'55.53" S em graus-minutos decimais ou latitude -15.765422° em graus decimais, sendo seguido do S e do negativo(-), por estar abaixo da linha do Equador. E longitude 47°53'41.15" O em graus-minutos decimais ou longitude -47.894764°, sendo seguido do O e do negativo(-), por estar à esquerda do Meridiano de Greenwich.



**Figura 3.3 - Imagem do *Google Earth*: Bloco 07**

Apesar de o valor ser dado em ângulos, no caso da latitude é possível determinar o valor em metros, pois o grande círculo tem 20.003,93 km, dividindo esse valor por 180°, considerando os dois hemisférios norte e sul, chega a conclusão que 1 grau ( 1° ) equivale a 111,133 km. Dividindo 1° por 60, chegamos ao valor de 1 minuto ( 1' ) equivalente a 1.852,22 metros. Dividindo esse valor por 60, temos que 1 segundo ( 1'' ) equivale a 30,87 metros.

<sup>1</sup> *Google Earth*: Software desenvolvido pela empresa *Google*, que permite ao usuário visualizar imagens de satélites, mapas e terrenos de qualquer lugar ao redor do globo terrestre.

No caso da longitude, o valor em metros varia de acordo com a posição do ponto em relação aos pólos, visto que esse valor na linha do Equador vale 0 (zero), chegando ao seu valor máximo nos pólos. No Brasil, esse valor varia de 30,91 metros na linha do Equador até 25,83 metros no extremo sul do país, conforme vemos os dados retirados da tabela 3.1 e feitas as comparações entre as coordenadas geográficas e as coordenadas UTM. Os cálculos de conversão de coordenadas geográficas para UTM foram feitos segundo (R&D, 2007).

**Tabela 3.1 – Comparação entre as coordenadas com 1 segundo de variação**  
(Fonte: o autor)

<b>Linha do Equador</b>			
Longitude	15°15'15"	X	471.717,4182
Latitude	00°00'00"	Y	0,0000
Longitude	15°15'16"	X	471.686,5081
Latitude	00°00'00"	Y	0,0000
Diferença		30,91	
<b>Brasília - DF</b>			
Longitude	47°25'25"	X	239.895,6837
Latitude	15°25'25"	Y	8.293.361,8762
Longitude	47°25'26"	X	239.865.8570
Latitude	15°25'25"	Y	8.293.361,8762
Diferença		29,82	
<b>Extremo Sul do Brasil</b>			
Longitude	53°25'25"	X	274.656,7562
Latitude	33°25'25"	Y	6.299.132,8238
Longitude	47°25'26"	X	274.630,9228
Latitude	15°25'25"	Y	6.299.132,8238
Diferença		25,83	

### 3.3 – Coordenadas UTM

O Universal Transversal de Mercator, UTM, é um sistema de localização referencial com base no sistema métrico, e é dividido em 60 zonas, múltiplas de 6 graus de longitude. Os eixos cartesianos de coordenadas têm origem no Equador para as coordenadas Y e o meridiano central de cada zona para as coordenadas X, nesse caso devendo ser indicada o número da zona. Os valores das coordenadas são dados em metros, e são decrescentes do norte para o sul e do oeste para o leste. Os valores iniciam em 10.000.000, para o hemisfério Sul, na linha do Equador, para não usar números negativos, e iniciam em 500.000 no limite de cada zona. (IBGE, 2010)

### 3.4 - Datum

Para que a posição seja definida sobre a superfície terrestre, é necessário que a Terra possa ser tratada matematicamente. Simplificando, aproximamos a Terra de um elipsóide de revolução, que é gerado pela rotação de uma elipse em torno dos eixos dos pólos, ou seja, pelo eixo menor. Um datum é caracterizado pela superfície de referência formada por um elipsóide em relação à Terra.

Um datum planimétrico ou horizontal é estabelecido usando cinco parâmetros: dois são usados para definir o elipsóide de referência e três são usados para definir o vetor de translação entre o centro da Terra real e o do elipsóide de referência. (INPE, 2010)

#### 3.4.1 – Datum horizontal WGS84

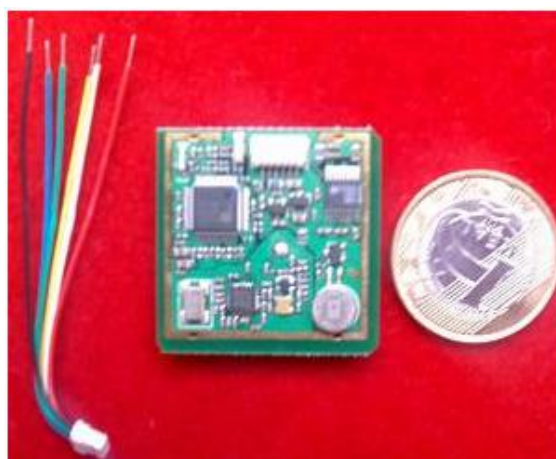
O *World Geodetic System* 1984, WGS84, foi criado em 1984 a partir do WGS72, que por sua vez foi criado a partir do WGS66, que foram criados para serem usados em todo o mundo. O WGS84 já foi melhorado, mas não alterado na sua base, sendo considerado um datum universal.

O WGS84 é um elipsóide de referência usado como modelo global e foi desenvolvido pela NASA E *The Ohio State University*, e apresenta a precisão de um metro ou menos em

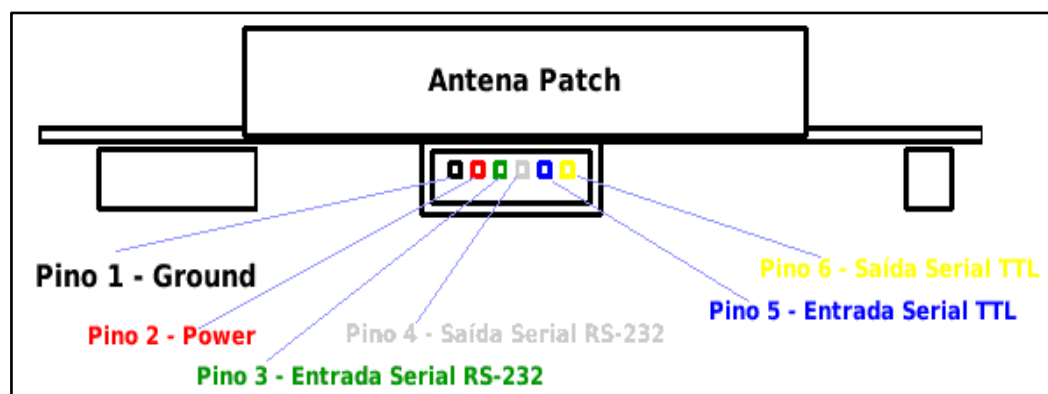
qualquer ponto da superfície terrestre, e é um sistema tridimensional de coordenadas simples, comum e acessível.

### 3.5 – Módulo GPS ME-1000RW

O módulo GPS escolhido foi o ME-1000RW, de altíssima sensibilidade com antena embutida e duas saídas seriais LVTTL e RS-232, além de sua alimentação ser bastante flexível, podendo variar de 3,3 V a 6 V. É um módulo altamente sensível, estável e tem ótima qualidade de recepção, detectando sinais de satélite de até -161 dBm. A saída dos dados é padrão NMEA 0183, e o datum é o WGS84. A taxa de atualização é de 1 Hz, ou seja, ele atualiza a sua posição a cada segundo. Por esse módulo já vir com antena embutida, é desnecessário o uso de antena externa, reduzindo o tamanho do módulo e do projeto. (datasheetME-1000RW, 2009)



**Figura 3.4 - Módulo ME-1000RW**  
Fonte: (DatasheetME-1000RW, 2009)



**Figura 3.5 - Ligações do ME-1000RW**  
Fonte: (DatasheetME-1000RW, 2009)



Para melhor entendimento, as especificações técnicas são mostradas na Tabela 3.2 e uma imagem é mostrada na figura 3.4, comparando com uma moeda de 1 Real, para se ter uma noção do tamanho real do módulo. Na figura 3.5 é mostrada a pinagem de ligação do módulo, com as respectivas cores de cada fio e sua aplicação.

**Tabela 3.2 - Especificações técnicas do ME-1000RW**

Fonte: (DatasheetME-1000RW, 2009)

Chipset	Skytraq V634LPX
Tipo de Receptor	65 canais de paralelização, código L1 C/A
Precisão	5 metros (CEP)
Velocidade	0.1m/sec
Tempo de Início	Partida Quente ( <i>hot start</i> ) < 10 segundos Partida Morna ( <i>warm start</i> ) < 32 segundos Partida Fria ( <i>cold start</i> ) < 35 segundos
Reaquisição	1s
Sensibilidade	-158dBm aquisição -161dBm rastreamento
Taxa de Atualização	1Hz
Dinâmicas	4G (39.2m/s <sup>2</sup> )
Limites Operacionais	Altitude < 18,000m ou velocidade < 515m/s (limite COCOM, um ou outro pode ser excedido, mas não ambos)
Interface Serial	Dupla saída serial: 1 Nível CMOS (LVTTTL = 0~3,3V) 1 Nível RS-232
Protocolo	NMEA-0183 V3.01 GPGGA, GPGSA, GPRMC, GPVTG, GPGSV 9600 baud, 8, N, 1 (Default)
Datum	Padrão WGS-84
Interface de Conexão	Conector Molex (part number 51021-0600) Plugue macho (part number 53261-0671) Obs. Já acompanha o cabo com o conector macho.
Voltagem de Entrada	3,3V ~ 6,0V DC
Consumo	Quando rastreando <= 35mA a 3,3V Máximo de 76mA a 3,3V
Dimensões	33,9 mm x 33,9 mm x 9,2 mm
Peso	14g
Temperatura	Armazenamento: -40°C ~ +80°C Operação: -20°C ~ +65°C
Humidade	5% ~ 95%
Outros	Led indicador de recepção Bateria de backup para partida rápida

Um dos fatores que devem ser observados nesse tipo de módulo é que em ambientes fechados ou com obstáculos sobre ele podem influenciar a qualidade de recepção, visto que são fabricados para serem usados a céu aberto. Ao serem usados dentro de um carro, deve-se procurar colocá-lo em um local que a antena embutida tenha a melhor visão possível do céu.

O módulo tem um LED indicador que se manterá ligado continuamente se a qualidade de recepção for insuficiente para determinar sua posição e piscará uma vez por segundo se o módulo conseguir determinar sua posição.

### 3.5.1 – Protocolo de comunicação do módulo GPS: Padrão NMEA 0183

O NMEA 0183 é um protocolo serial de interface que foi criado pela Marinha americana com o propósito de organizar e padronizar as informações recebidas dos satélites. O nome faz referencia ao seu órgão de criação, National Marine Electronics Association, que nasceu como uma associação de fabricantes de produtos eletrônicos norte-americanos, mas que hoje tem membros espalhados por todo o mundo. (NMEA, 2008)

Com a unificação da comunicação conseguida pelo padrão NMEA, facilitou-se muito o desenvolvimento e integração de equipamentos eletrônicos no campo da navegação. Esse padrão é usado pela maioria dos receptores GPS. Segue-se um exemplo de como esses dados são capturados:

```
$GPGSA,A,3,01,20,19,13,,,,,,,,,40.4,24.4,32.2*0A<CR><LF>
$GPGGA,060932.448,2447.0959,N,12100.5204,E,1,08,1.1,108.7,M,,,,0000*0E<CR><LF>
$GPRMC,092204.999,A,4250.5589,S,14718.5084,E,0.00,89.68,211200,,A*25<CR><LF>
$GPVTG,89.68,T,,M,0.00,N,0.0,K,A*5F<CR><LF>
$GPGSV,3,1,09,28,81,225,41,24,66,323,44, 20,48,066,43, 17,45,336,41*78<CR><LF>
```

Os dados são enviados através de strings em que cada linha identifica certo tipo de dados do GPS. Alguns dos tipos de dados dessas linhas são os seguintes:

- GPGGA – Global Positioning System Fix Data
- GPGLL – Geographic position, Latitude and Longitude
- GPGSA - Satellite status
- GPGSV - Satellites in view
- GPRMC - Recommended Minimum sentence C
- GPVTG - Track made good and ground speed
- GPZDA - Data and Time

Nesse projeto, a linha desse protocolo que será usada é a linha \$GPRMC, de onde serão tiradas a latitude e longitude. Na tabela 3.3, temos a explicação de cada campo do exemplo a seguir:

**Estrutura:**

\$GPRMC, hhmmss.sss, A, ddm. mmmm, a, dddmm.mmmm, a, x.x, x.x, ddmmyy, x.x, a, a, \*hh <CR>  
                   1          2          3          4          5          6  7  8          9          10 11 12  13

**Exemplo:**

\$GPRMC,092204.999,A,4250.5589,S,14718.5084,E,0.00,89.68,211200,,A\*25<CR><LF>

**Tabela 3.3 - Significado de cada campo da linha \$GPRMC**

Fonte: (NMEA, 2008)

Campo	Nome	Exemplo	Descrição
1	Tempo UTC	060932.448	Horário UTC* no formato hhmmss.sss Variação de (000000.000 ~ 235959.999)
2	Status	A	'V' = GPS aquecendo 'A' = Dados válidos
3	Latitude	2447.0959	Latitude no formato ddm. mmmm
4	Indicador N/S	N	Hemisfério, 'N' = Norte, 'S' = Sul
5	Longitude	12100.5204	Longitude no formato dddmm.mmmm
6	Indicador E/W	E	Hemisfério, 'E' = Leste, 'W' = Oeste
7	Velocidade	000.0	Velocidade em nós (000.0 ~ 999.9)
8	Curso	000.0	Curso em graus (000.0 ~ 359.9)
9	Data UTC	211200	Data UTC de uma posição fixa no formato, ddmmyy
10	Varição magnética		Em graus (000.0 ~ 180.0)
11	Varição magnética		Em Direção 'E' = Leste; 'W' = Oeste
12	Indicador de Modo	A	'N' = Dados não válidos 'A' = Modo autônomo 'D' = Modo Diferencial 'E' = Modo Estimado (dead reckoning DR) 'M' = Modo de entrada manual 'S' = Modo de Simulação
13	Checksum	0E	Começa com * e consiste de 2 caracteres e representam um número hexadecimal. É o ou exclusivo de todos os caracteres entre "\$" e o "*"

### 3.6 – Discadora Westron EV300

A discadora Westron EV300 é usada para monitorar aparelhos eletrônicos através de um telefone celular. Ela realiza as ligações de alerta automaticamente quando o evento ocorrer. Ela oferece uma versatilidade para acompanhar qualquer tipo de aparelho eletrônico, monitorando qualquer evento que possa ser controlado por um sinal positivo compatível. Além do sinal positivo, também pode contar com alerta de pânico através de um botão ou

corte de linha telefônica ligada a ela. Como se pode ver na figura 3.6 ela é ligada a um celular Siemens M50. (Westron, 2010)

A tensão para detecção de evento deve ser uma tensão maior que 2,0V por pelo menos três segundos e menor que 16,0V. Quando a discadora é acionada, com um sinal positivo de pelo menos 3 segundos, ela efetua as chamadas para os números cadastrados e depois volta ao modo de monitoramento.



**Figura 3.6 - Discadora Westron e Celular Siemens M50**  
Fonte: (Westron, 2010)

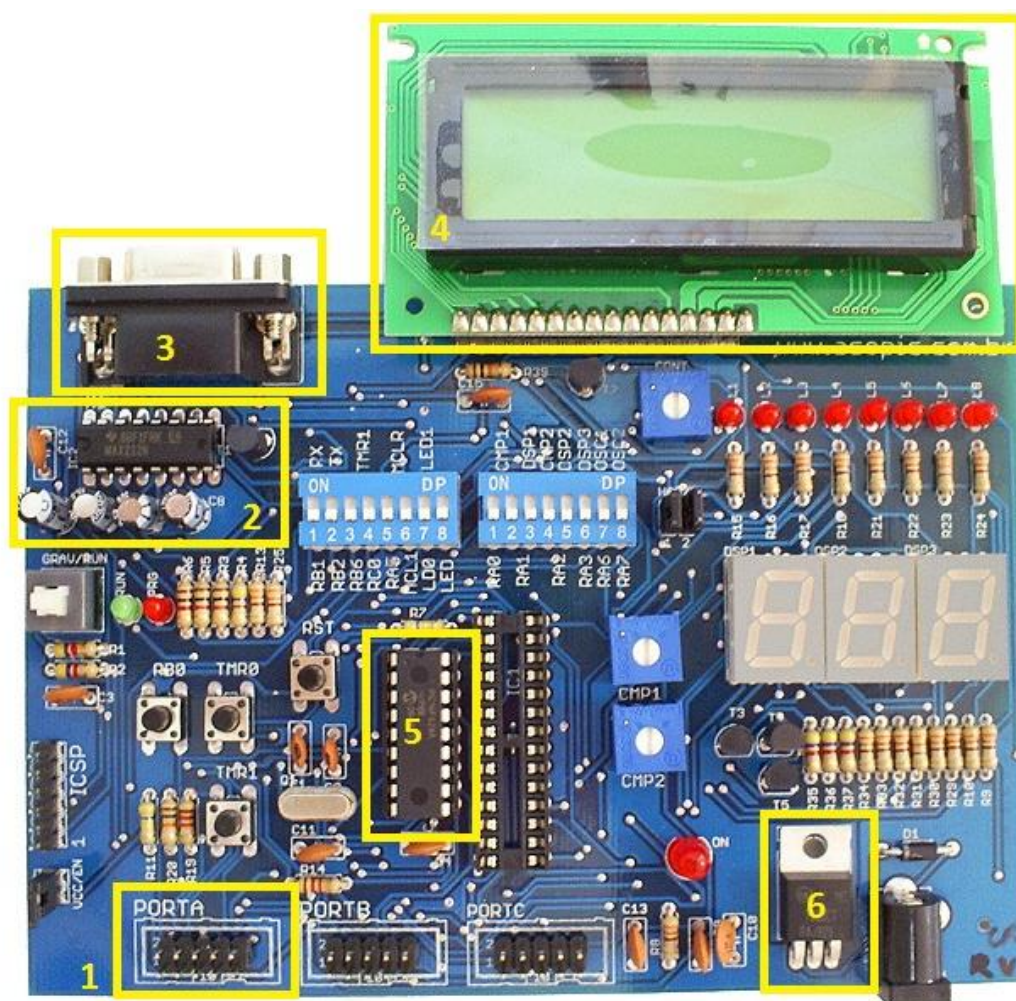
### 3.6.1 – Celular Siemens M50

O modelo do celular Siemens M50 foi escolhido por ser um modelo compatível com a discadora. O celular é alimentado por uma placa de alimentação ligada à discadora, não necessitando de bateria. Os números para os quais serão feitas as ligações em caso de evento, são registrados nas três primeiras posições do chip inserido no celular. As ligações são consecutivas do primeiro ao terceiro número.

### 3.7 – KIT ACEPIC 16F628A

Foi escolhido para o desenvolvimento da aplicação o Kit ACEPIC mostrado na figura 2.5, desenvolvido pela ACEPIC Tecnologia e Treinamento Ltda, empresa de projetos e treinamentos em eletrônica e telecomunicações. Apesar de não usar nesse projeto todos os recursos desse kit, o seu uso na aplicação, por se tratar de um protótipo, facilita pela facilidade

das ligações na fase de implementação e testes, o que se demandaria um tempo muito maior no caso de confecção da placa gráfica para o circuito. O kit é mostrado na figura 3.7 com as definições dos módulos que serão usados nesse projeto.



**Figura 3.7 - Kit de Desenvolvimento ACEPIC 18 e 28 pinos**  
 Fonte: (ACEPIC, 2009), com adaptações.

Este Kit de desenvolvimento apresenta muitas características para uso em diversas aplicações, mas as principais que serão usadas nesse projeto são as seguintes:

1. Porta de Expansão – Port A
2. MAX232
3. Conector DB9 - USART (RS232)
4. Display de Cristal Líquido (LCD) 16 colunas e 2 linhas com Backlight
5. Microcontrolador PIC16F628A
6. Regulador de tensão 12,0 V para 5,0 V

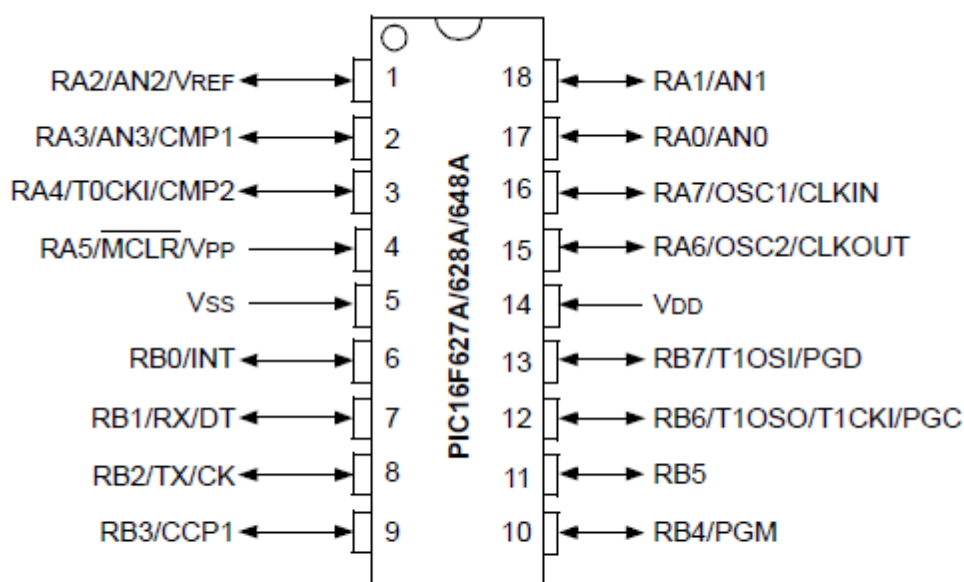
Outra facilidade desse kit é o fato da gravação da memória flash do PIC ser feita diretamente no kit, gravando o código hexadecimal que foi compilado através do código fonte escrito na linguagem C.

### 3.7.1 - Microcontrolador PIC16F628A

Os microcontroladores que existem no mercado integram as funções de um microprocessador, da memória de dados e instruções, inclusive em muitos casos portas seriais e paralelas bidirecionais, conversores A/D, timers, watchdog entre outros

Os microcontroladores PIC são fabricados pela Microchip Technology, e processam dados de 8, 16 e 32 bits, tendo uma grande variedade de modelos, com recursos de programação por memória flash, EEPROM e OTP.

O microcontrolador escolhido para esse projeto é o PIC16F628A, que tem 18 pinos com definições mostradas na figura 3.8:



**Figura 3.8 - Pinagem do PIC16F628A**

Fonte: (Microchip, 2007)

A tensão nominal de alimentação é de 5Vcc, mas a variação dessa tensão vai de 3.0V a 5.5Vcc. A memória flash de programa é de 2000 palavras de 14 bits.

### 3.8 – Sistema de comunicação entre o microcontrolador e o módulo GPS

Para que os módulos possam se comunicar com o microcontrolador, é necessário entender os níveis de tensão de cada um e como essa comunicação acontecerá. O microcontrolador se comunica usando os níveis TTL, os módulos se comunicam utilizando os níveis Lv-TTL e a entrada serial do kit utilizado nesse projeto tem o nível RS232. Para que esses módulos possam se comunicar sem que aconteça perda de dados ou mesmo a queima de algum equipamento é necessária que seja feita uma conversão entre esses níveis de tensão utilizando o MAX232, que transforma o nível serial em TTL e vice-versa.

#### 3.8.1 – Comunicação serial do microcontrolador

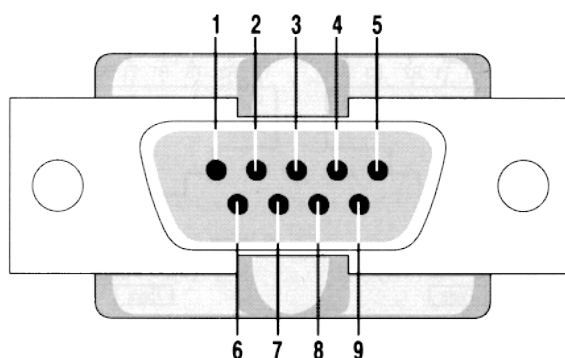
Pela necessidade de comunicação entre os mais diferentes equipamentos foi criada a transmissão serial, que consiste no envio de certo caractere composto de vários bits, de forma que cada bit de cada caractere seja transmitido de forma seqüencial, ou seja, um após o outro. Para que vários sistemas se comuniquem de forma compatível, foi criado um código binário para cada caractere chamado de código ASCII, onde cada caractere possui um correspondente binário.

Nesse projeto a comunicação é assíncrona, onde cada caractere, que é transmitido bit a bit, contém um bit de início de transmissão, chamado de *start bit* e um bit de fim de transmissão, chamado de *stop bit*. Quando o start bit é reconhecido, o sistema varre a porta para detectar os bits até chegar ao sétimo, momento em que fica esperando o *stop bit*, que ao ser detectado, chega ao fim a recepção daquele caractere e o sistema entra em repouso esperando um novo *start bit* para que possa reiniciar a recepção de um novo caractere. Nesse modo de transmissão, deve-se definir a mesma taxa de transmissão e recepção, para que a comunicação possa acontecer.

### 3.8.2 - RS232

Nesse projeto, será usado o padrão RS232 para comunicação entre o kit e o módulo GPS, que estão ligados ao MAX232 para que possam se comunicar, já que eles se comunicam no nível TTL.

Para as ligações de recepção e transmissão entre eles, são usados os pinos dos conectores DB9, conforme podemos ver na figura 3.9 e sua designação na tabela 3.4.



**Figura 3.9 - Conector DB9**  
 Fonte: (System, 2007)

**Tabela 3.4 - Designação dos pinos do DB9**

Fonte: (System, 2007)

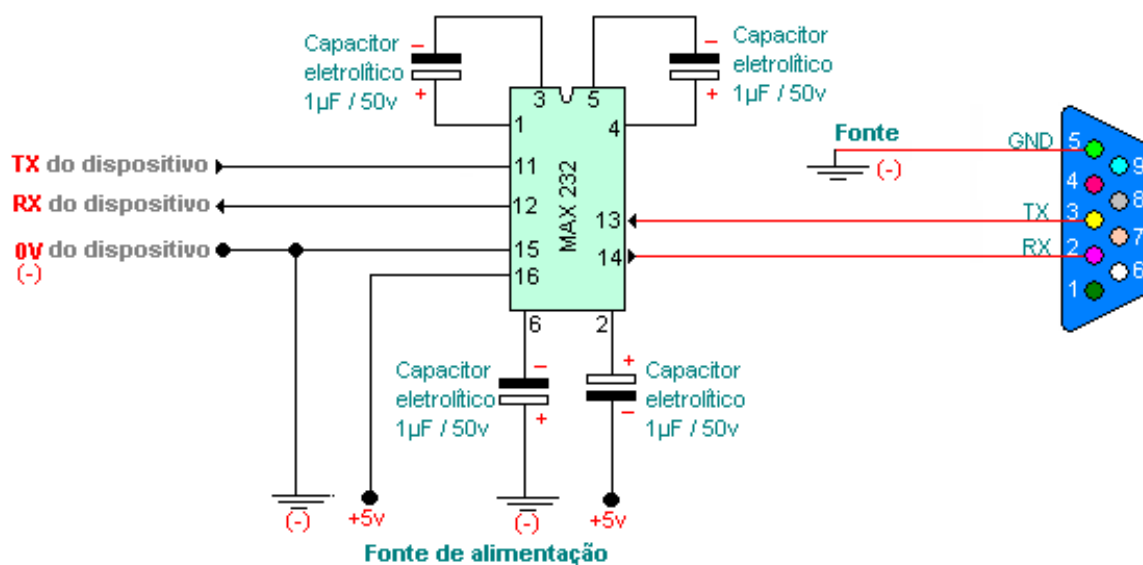
Número	Nome	Designação
1	CD - Carrier Detect	Detecção de portador
2	RXD - Receive Data	Recepção de dados
3	TXD - Transmit Data	Transmissão de dados
4	DTR - Data Terminal Ready	Terminal pronto
5	GND - Signal Ground	Massa lógica
6	DSR - Data Set Ready	Dados prontos
7	RTS - Request To Send	Pedido de emissão
8	CTS - Clear To Send	Empréstimo a emitir
9	RI - Ring Indicator	Indicador de campainha eléctrica
	Shield	Blindagem

### 3.8.3 - MAX232

Devido a diferença entre os níveis de tensão usados na comunicação entre os módulos e o microcontrolador através da porta RS232, é necessário o uso do MAX232, que converte os níveis de tensão do RS232 para TTL e vice versa.



Para melhor entendimento de como essa ligação é feita, podemos ver o esquema da figura 3.10:



**Figura 3.10 - Ligação com o MAX232**  
 Fonte: (Messias, 2004)

### 3.9 – Linguagem de Programação

À medida que aumenta a complexidade da aplicação desenvolvida, dificulta-se a criação dos programas para microcontroladores. Os primeiros programas eram escritos com códigos chamados de códigos de máquinas, que consistiam de dígitos binários que eram inseridos por meio de cartões, fitas perfuradas ou discos magnéticos.

Com a evolução da programação de sistemas, surgiu a linguagem Assembly, que usava mnemônicos, ou seja, abreviações de termos que descrevem os comandos em código de máquina. Sem dúvida, simplificou muito a vida dos programadores, mas não resolveu todos os problemas.

Depois disso, surgiu a Linguagem C, tornando-se uma escolha natural para a programação de microcontroladores, pois permite uma grande velocidade na criação de projetos, devido às facilidades de programação oferecidas pela linguagem. Hoje em dia, a maioria dos microcontroladores disponíveis no mercado conta com compiladores de linguagem C para o desenvolvimento de software. (Pereira, 2004)

Para esse projeto foi escolhida a Linguagem C, devido à facilidade no uso dos microcontroladores PIC com essa linguagem, permitindo que a preocupação fosse com a programação da aplicação em si, já que para integração dos dispositivos, contamos com uma grande biblioteca de apoio tanto em livro como na própria internet. Também essa linguagem permite certa complexidade para o uso de funções muito maior que o Assembly, por exemplo.

No próximo capítulo será apresentado e explicado o modelo proposto para a solução do problema, Muitos dos conceitos e definições aqui apresentados serão de grande utilidade no desenvolvimento desse modelo, já que trazem um referencial tecnológico para um bom entendimento das soluções apresentadas.

## 4. CAPÍTULO Nº 4 – MODELO PROPOSTO

Neste capítulo serão mostradas todas as etapas de implementação desse projeto, para chegar à solução do problema apresentado. É detalhada cada passo dessas etapas, mostrando desde o planejamento até a execução do projeto de hardware e software.

### 4.1 - Apresentação Geral do Modelo Proposto

Para a solução do problema, foi levado em conta que cada item da solução devia ser analisado separadamente, para depois chegar-se à solução geral. O monitoramento começa fixando uma coordenada para comparação. Depois se começa a monitorar as coordenadas e comparar se não houve uma alteração fora da tolerância estabelecida. Se o veículo entrar em movimento, o GPS identifica uma alteração nas coordenadas e aciona o celular que faz a ligação para o celular do dono do veículo.

Este modelo tem um custo também acessível, conforme estimativa baseada nos valores da Tabela 4.1, que mostra os valores dos principais componentes utilizados.

**Tabela 4.1 – Custo de Componentes**  
(Fonte: O Autor)

<b>Produto</b>	<b>Valor (em Reais)</b>
Módulo GPS ME-1000RW	130,00
Discadora Westron	90,00
Celular Siemens M50	70,00
PIC 16F628A	15,00
Display LCD 16x2	35,00
<b>Total</b>	<b>340,00</b>

## 4.2 – Descrição das Etapas do Modelo

A seguir são mostradas as etapas para se chegar à solução geral, mostrando-se o funcionamento e uso de cada componente.

### 4.2.1 - Comunicação do microcontrolador com o módulo GPS

A primeira coisa feita foi conseguir a comunicação com o módulo GPS, pois é a partir daí que se conseguem os dados para toda a análise das ações a serem tomadas. O módulo é alimentado com 5,0 V, a partir do Kit de desenvolvimento ACEPIC, em que são ligados os fios Vcc e o aterramento (GND), através dos pinos de expansão da Port A, definidos no Kit. O consumo máximo do módulo GPS é de 70 mA, segundo o manual, a corrente fornecida pela fonte de alimentação do kit é suficiente para alimentação do módulo GPS em conjunto com os outros componentes.

A comunicação serial é feita ligando o TX do módulo GPS (fio laranja) com o RX do kit, através do conector DB9. O RX do conector DB9 é ligado ao MAX232 do Kit e posteriormente a RX do PIC16F628A, no pino RB0.

O módulo começa a enviar a informação automaticamente ao ser alimentado, a uma velocidade de 9600 bauds<sup>2</sup> por segundo, com 8 bits de dados, sem paridade e com bit de parada. Essas informações devem ser configuradas na programação da comunicação assíncrona no código.

### 4.2.2 - Discadora e celular GSM

A alimentação da discadora e do celular acoplado a ela é feita através do kit de desenvolvimento. A alimentação da discadora deve ser de 9,0 V a 16,0 V. No caso desse projeto foi usada uma alimentação de 12,0 V, conseguida ligando o fio vermelho da discadora ao pino de entrada do regulador de tensão do kit de desenvolvimento e o e o fio marrom da

---

<sup>2</sup> *Baud*: medida de velocidade de transmissão, e mostra um evento ou mudança de sinal. A velocidade é mostrada em *bauds* por segundo ao invés de bits por segundo, pois em um *baud* pode haver mais de um bit.

discadora ao GND da placa. Como o consumo máximo da discadora é de 230 mA, segundo o manual, a corrente fornecida pela fonte de alimentação é suficiente para alimentação da discadora em conjunto com os outros componentes.

No caso de encontrar diferença nas coordenadas, um sinal positivo entre 2,0 V e 16,0 V por pelo menos três segundos deve ser enviado à discadora. Nesse projeto, o sinal enviado será de 5,0 V, a partir do pino RA7, da port A do kit de desenvolvimento, que é ligado ao fio laranja da discadora.

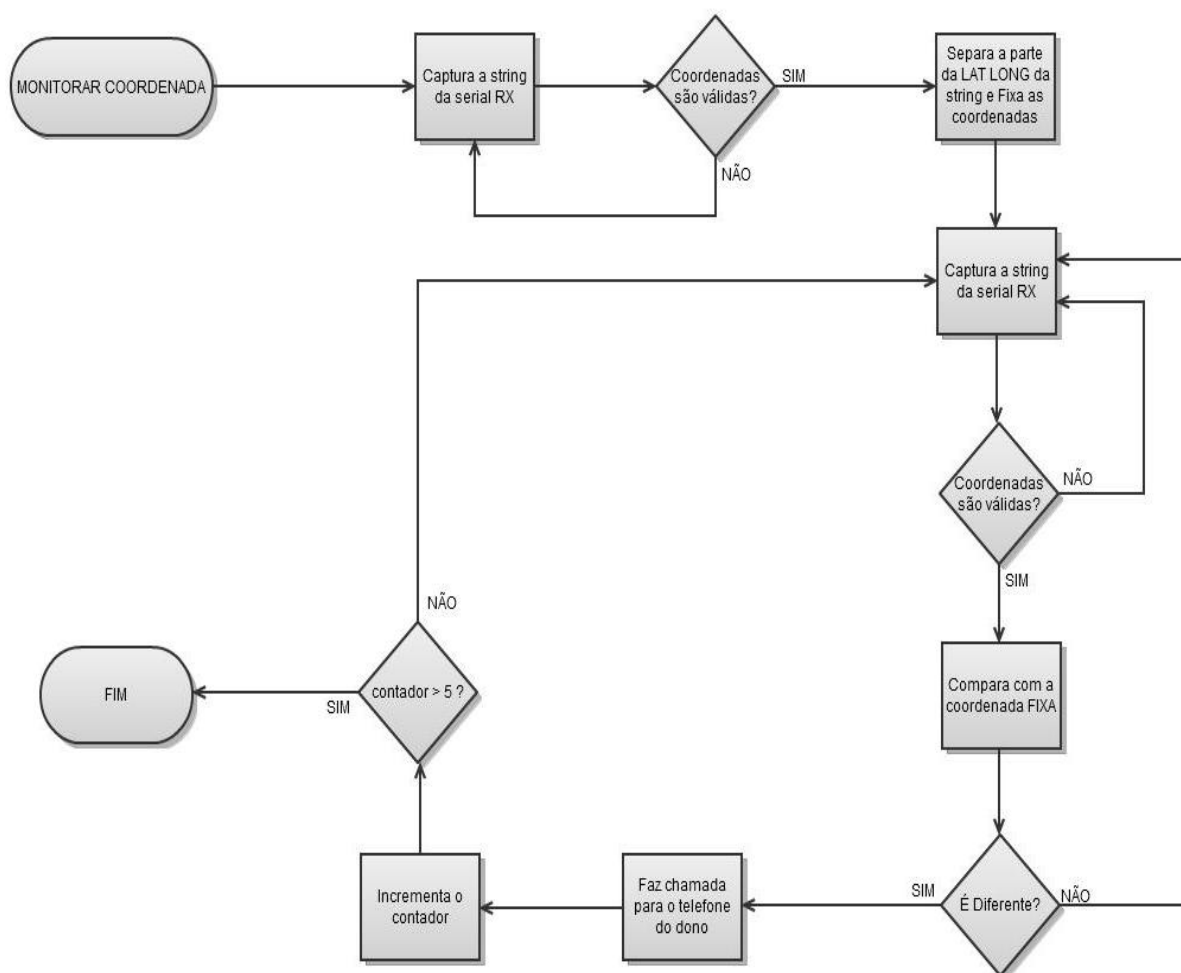
O celular Siemens M50, é ligado diretamente à discadora, sendo o uso de bateria, pois é alimentado em conjunto com a discadora na qual está acoplado através de uma cabo de comunicação e uma placa de alimentação para eliminação da bateria. O celular é ligado automaticamente ao ser alimentada a discadora pelo circuito.

Os telefones para os quais serão feitas as ligações são cadastrados nas três primeiras posições na agenda do chip GSM, sendo as ligações feitas automaticamente para esses números. Para alterar esses números, basta alterar os números na agenda do chip.

#### 4.2.3 - Software

O software foi desenvolvido em linguagem C, mas antes de começar a implementação do código, foi necessário fazer um fluxograma para esquematizar como chegar a solução do problema e integrar todos os componentes desse projeto. Foram seguidos os passos e decisões mostrados na figura 4.1.

Nesse fluxograma, vemos que ao estacionar o veículo, é necessário fixar a coordenada. Para isso, é feita a captura da coordenada pela porta serial e é necessário testar se essa coordenada é válida, pois usaremos apenas a linha \$GPRMC do protocolo NMEA, sendo necessário testar se é essa a linha capturada. Se a linha não for essa, ou seja, as coordenadas não são válidas, daí captura-se novamente até encontrar coordenadas que sejam válidas. Encontrando coordenadas válidas, separa-se da linha apenas a latitude e a longitude, que serão os dados a serem analisados. Essas primeiras coordenadas são fixadas e servem para comparação com as demais que são capturadas e comparadas com as coordenadas que foram fixadas.



**Figura 4.1 – Fluxograma do monitoramento das coordenadas**  
(Fonte: O autor)

Para o monitoramento, o teste segue a mesma ordem que foi usada para separar a latitude e longitude das coordenadas fixas. São verificadas se as coordenadas são válidas e depois separa a latitude e longitude. Quando comparar as coordenadas capturadas com as coordenadas fixas e não encontrar diferença (o veículo não se movimentou), volta a capturar e continua o monitoramento. No caso de encontrar diferença (houve deslocamento do veículo), faz uma ligação para o telefone do dono e começa a contar as ligações até o máximo de cinco, para que o celular do dono não fique recebendo ligações por tempo indeterminado.

Depois de fazer a ligação, é incrementado o número de ligações e analisado se não chegou ao número limites de ligações. Se não tiver chegado ao limite, captura novamente as coordenadas e compara novamente. Se continuar diferente, faz novamente uma ligação para o dono. Ao chegar ao número de cinco ligações, desliga automaticamente o monitoramento.

O monitoramento é desligado se chegar ao número máximo de cinco ligações depois de acontecer um evento. Isso acontece para evitar que o celular do dono fique recebendo

ligações por tempo indeterminado e com isso não consiga realizar ligações enquanto o sistema de alerta não fosse desligado.

Na figura 4.2 é mostrado um trecho do código constante no APÊNDICE, que mostra como é feita a captura e a condição de que o tamanho da string tenha pelo menos 25 caracteres, tamanho mínimo para se conseguir capturar a latitude e longitude. Os 25 caracteres são contados a partir do caractere A, no trecho da linha &GPRMC e que mostra que os dados do GPS são válidos, até o fim da longitude.

Também foram retirados os pontos (.), da string capturada, para se trabalhar com os valores inteiros ao invés de flutuante, para economizar memória.

```

while(tam<=25){          //tamanho minimo para se capturar lat e longit

printf(lcd_escreve,"\f SISTEMA ALERTA ");
printf(lcd_escreve,"\n  FIXANDO...  ");

delay_ms(2000);

  for(pos=0; pos<79;pos++){

    captura=rs232_recebe(); //captura pela comunicacao serial
    latlong[pos]=captura;   //coloca o caractere na string
  }

pass1=strstr(latlong,teste); //passagem para testar se eh a linha RMC
pass2=strchr(pass1,'A');     //passagem para testar se os dados sao validos
tam=strlen(pass2);          //tamanho do A ate o fim da string

if(tam>25){                //so entra se for o tamanho minino exigido

  for(pos=0,latpos=2; pos<8 ;latpos++,pos++){
    if(pass2[latpos]=='.'){ //para retirar os pontos da string
      latpos++;
    }
    latFIXA[pos]=pass2[latpos];
  }
  latFIXA[pos]='\0';       //para se definir o fim da string

  for(pos=0,latpos=14; pos<9;latpos++,pos++){
    if(pass2[latpos]=='.'){ //para retirar os pontos da string
      latpos++;
    }
    longitFIXA[pos]=pass2[latpos];
  }
  longitFIXA[pos]='\0';   //para se definir o fim da string

  printf(lcd_escreve,"\f"); //Limpa Display
  printf(lcd_escreve,"\fLtfix:%s", latFIXA); // Escreve Lat Fixa
  printf(lcd_escreve,"\nLgfix:%s", longitFIXA); // Escreve long Fixa
  delay_ms(8000);        //aguardar 8 segundos
} //fim do IF(tam>25)
} //fim do While(tam<=25)

```

**Figura 4.2 – Trecho do código para captura da latitude e longitude pela porta RS232**  
(Fonte: O autor)

Como podemos observar na figura 4.2, os caracteres são capturados um a um e colocados em uma string com 79 caracteres. Apesar de ser suficiente capturar 44 caracteres da linha \$GPRMC até chegar ao fim da latitude e longitude, na maioria das vezes captura-se o restante de uma linha junto com a linha desejada, motivo pelo qual se faz necessário um número adicional de caracteres para se ter uma margem de segurança para maior agilidade na captura.

Mesmo depois de capturada a string, é necessário testar se existe a string \$GPRMC, no programa definida por teste, para saber se aquela é uma string válida. Se for válida, são feitas algumas passagens através das variáveis *pass1* e *pass2*, para separar a latitude e longitude e colocá-las nas variáveis específicas. Se a string não for válida, é feita nova captura até conseguir uma string válida.

Esse teste é feito inicialmente para fixar as coordenadas que serão usadas para comparação, mas o mesmo teste é feito para captura das coordenadas que serão usadas para comparação durante o monitoramento.

Na figura 4.3, é mostrado o trecho do código em que é feita a comparação entre as coordenadas para ver se existem diferenças, caracterizando assim movimento do veículo. De início, foi necessário converter o valor da *string* pra *signed int*, para que fossem mostrados os valores negativos também, já que a diferença pode ser positiva ou negativa, dependendo da coordenada capturada.

Para comparar o valor das coordenadas, o teste é feito com a diferença entre as coordenadas, que não pode passar de 50, correspondente a meio segundo. Como mostrado no Capítulo 3, na latitude, meio segundo corresponde a aproximadamente 15,00 metros e a longitude pode variar de aproximadamente 13,00 metros a 15,00 metros, essa variação fica dentro de uma tolerância aceitável. Nos testes realizados com o protótipo parado, a variação encontrada foi sempre abaixo de meio segundo, com algum tipo de obstáculo próximo, motivo pelo qual foi escolhido esse valor como tolerância aceitável.

Se encontrada a diferença, o número de ligações é incrementado, é informado pelo LCD que foi detectado movimento e qual é o número daquela ligação. Daí, coloca o estado lógico do pino RA7 em 1, enviando um sinal positivo para a discadora fazer a ligação para o telefone cadastrado no chip. Esse sinal é enviado por dez segundos, e o estado do pino volta a zero, para que seja feita nova captura de coordenada e nova comparação.



Quando forem efetuadas cinco ligações, é informado que já foram feitas as cinco ligações e sairá do loop que está condicionado ao número de ligações ser um valor menor que cinco.

```

Ulat=atoi(lat);           //converte string em signed int
UlatFIXA=atoi(latFIXA);   //converte string em signed int
Ulongit=atoi(longit);     //converte string em signed int
UlongitFIXA=atoi(longitFIXA); //converte string em signed int

DifLat=(Ulat-UlatFIXA);     //diferença entre a lat atual e a fixa
DifLongit=(Ulongit-UlongitFIXA); //diferença entre a long atual e a fixa

if ( (DifLat>50) || (DifLat<-50) || (DifLongit>50) || (DifLongit<-50) ){

    ContLig++;

    printf(lcd_escreve,"\f"); //Limpa Display
    printf(lcd_escreve,"\f  MOVIMENTO  ");
    printf(lcd_escreve,"\n  DETECTADO!! ");
    delay_ms(3000);

    printf(lcd_escreve,"\f"); //Limpa Display
    printf(lcd_escreve,"\f  LIGACAO N.%d  ", ContLig);
    printf(lcd_escreve,"\n  PARA O DONO! ");
    delay_ms(3000);

    OUTPUT_HIGH(PIN_A7);    //coloca o pino em 1
    delay_ms(30000);        //aguarda 30 segundos
    OUTPUT_LOW(PIN_A7);     //volta o pino para estado logico ZERO.
    delay_ms(30000);        //aguarda 30 segundos

    if( ContLig==5){        //testa se já efetuou as 5 ligações
        printf(lcd_escreve,"\f");
        printf(lcd_escreve,"\f  5 LIGACOES!!! ");
        printf(lcd_escreve,"\n  SAINDO!!! ");

        delay_ms(10000);
        printf(lcd_escreve,"\f"); //Limpa Display

        ContLig++;          //incrementa as ligações para sair,
                            //pois a ContLig fica igual a 6
    }

}

} // fim do WHILE(ContLig<5)

```

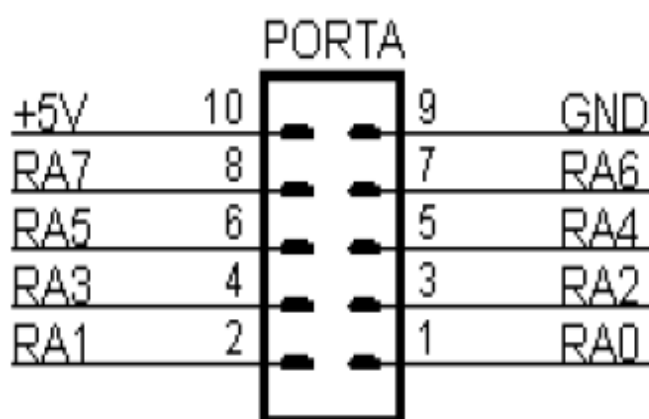
**Figura 4.3 - Trecho do código para comparação das coordenadas**  
(Fonte: O autor)

### 4.3 - Descrição da Implementação

Nesse tópico são mostrados os passos para a implementação específica de cada componente. Serão mostradas as ferramentas e metodologias usadas, bem como os cuidados tomados ao serem usadas nas aplicações.

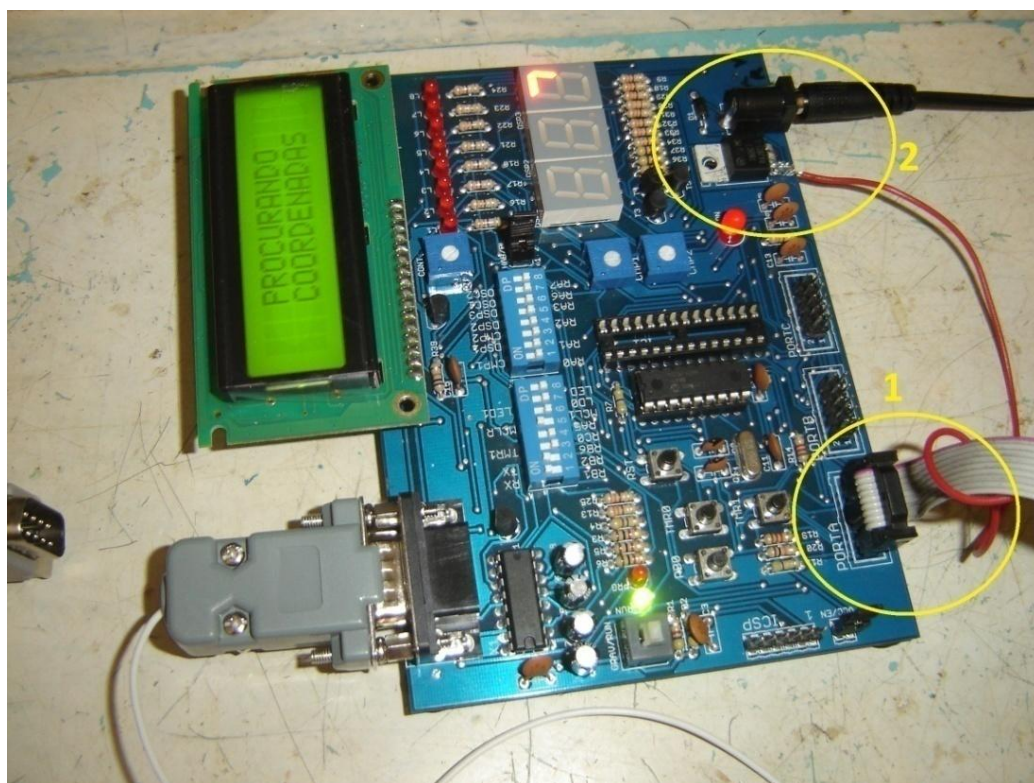
#### 4.3.1 – Montagem do Protótipo

Para montar o protótipo, as alimentações dos módulos do kit de desenvolvimento, do módulo GPS, da discadora e do celular foram feitas a partir da alimentação do kit de desenvolvimento. O Vcc da alimentação é de 12,0 V+, passando por um regulador de tensão, que transforma a tensão para 5,0 V. O módulo GPS foi alimentado com 5,0 V, a partir do pino 10 e o GND a partir do pino 9 da PortA, como mostrado na figura 4.4. O pino 8, ligado ao RA7, serviu para envio do sinal para acionamento da discadora. A ligação na Port A do kit de desenvolvimento pode ser vista no detalhe 1, da figura 4.5.



**Figura 4.4 – Port A de expansão do Kit de desenvolvimento**  
(Fonte:ACEPIC, 2009)

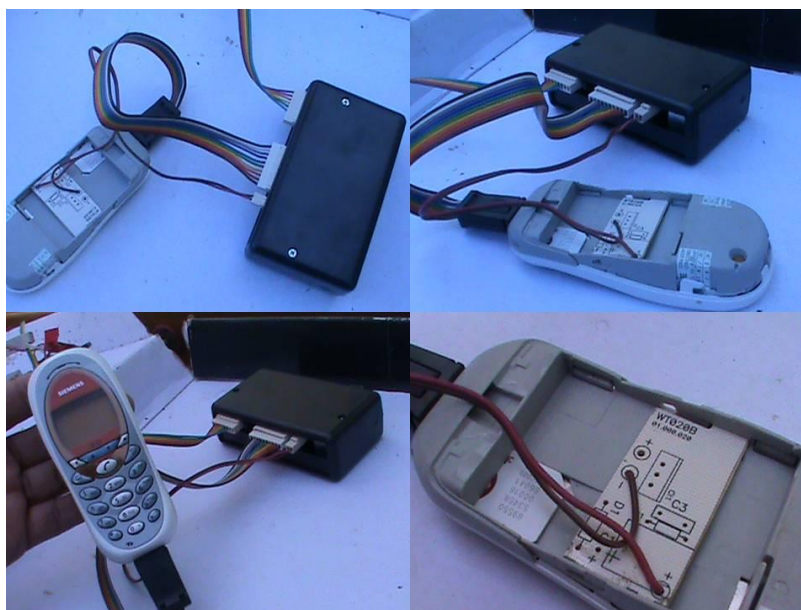
A alimentação de 12,0 V, da discadora foi feita através do pino de entrada do regulador de tensão do Kit de desenvolvimento, e pode ser vista no detalhe 2, da figura 4.5.



**Figura 4.5 – Detalhes das ligações no Kit de desenvolvimento**  
(Fonte: O autor)

A comunicação do módulo GPS e o microcontrolador foi feita ligando o fio RS232 TX (branco) ao pino 3 do conector DB9.

O celular é alimentado automaticamente pela discadora e sua ligação é feita pelos cabos da discadora como pode ser visto na figura 4.6.



**Figura 4.6 – Ligação do celular com a discadora**  
(Fonte: O autor)

### 4.3.2 - Compilação e alteração no código fonte

Para a compilação e as alterações no código fonte, que foram sendo necessárias durante a implementação, foi usado o programa PIC C Compiler. Esse programa além de compilar, mostra os avisos e erros, além do uso de memória RAM e ROM do programa, como mostrado na figura 4.7.

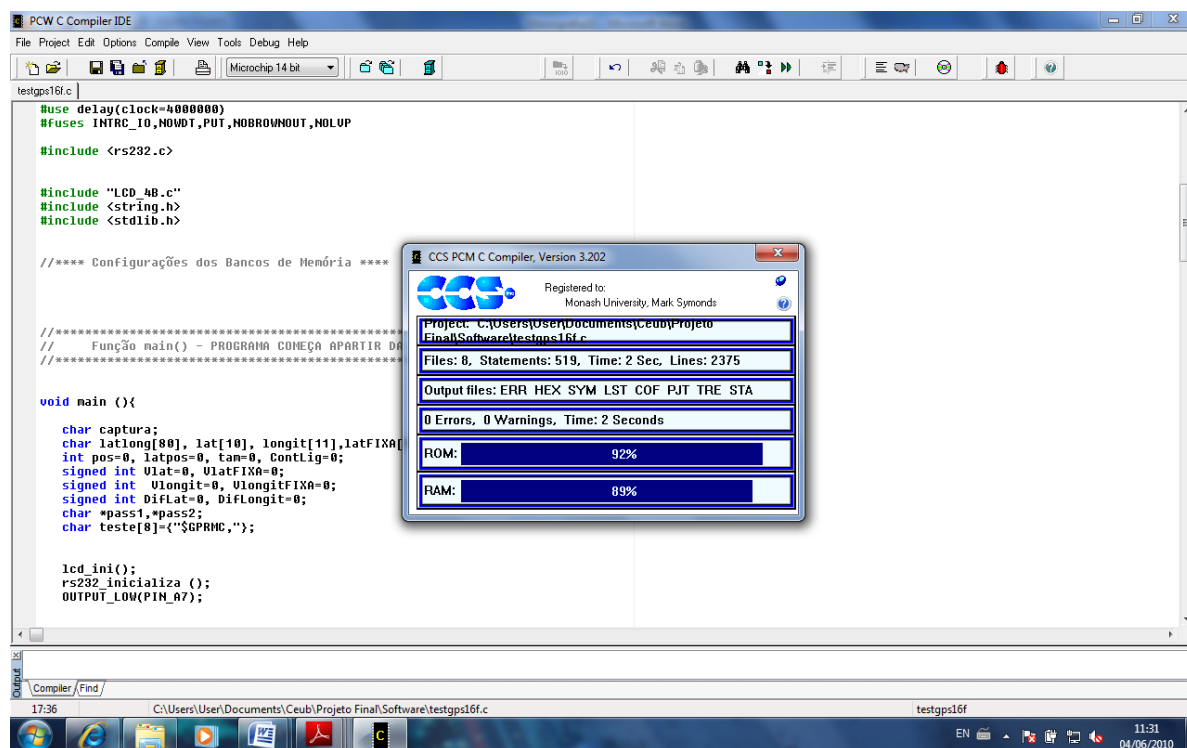


Figura 4.7 – Tela do PIC C Compiler  
(Fonte: O autor)

Depois de gerado o código hexadecimal, deve-se gravar o programa no PIC16F628A. Para a gravação no PIC, foi usado o programa ICProg, em que o programa é gravado através da porta serial do computador ligada diretamente através do conector DB9 do Kit de desenvolvimento. Na figura 4.8, pode ser visto o ICProg gravando no microcontrolador. Depois da gravação, o programa faz a verificação dos dados gravados.

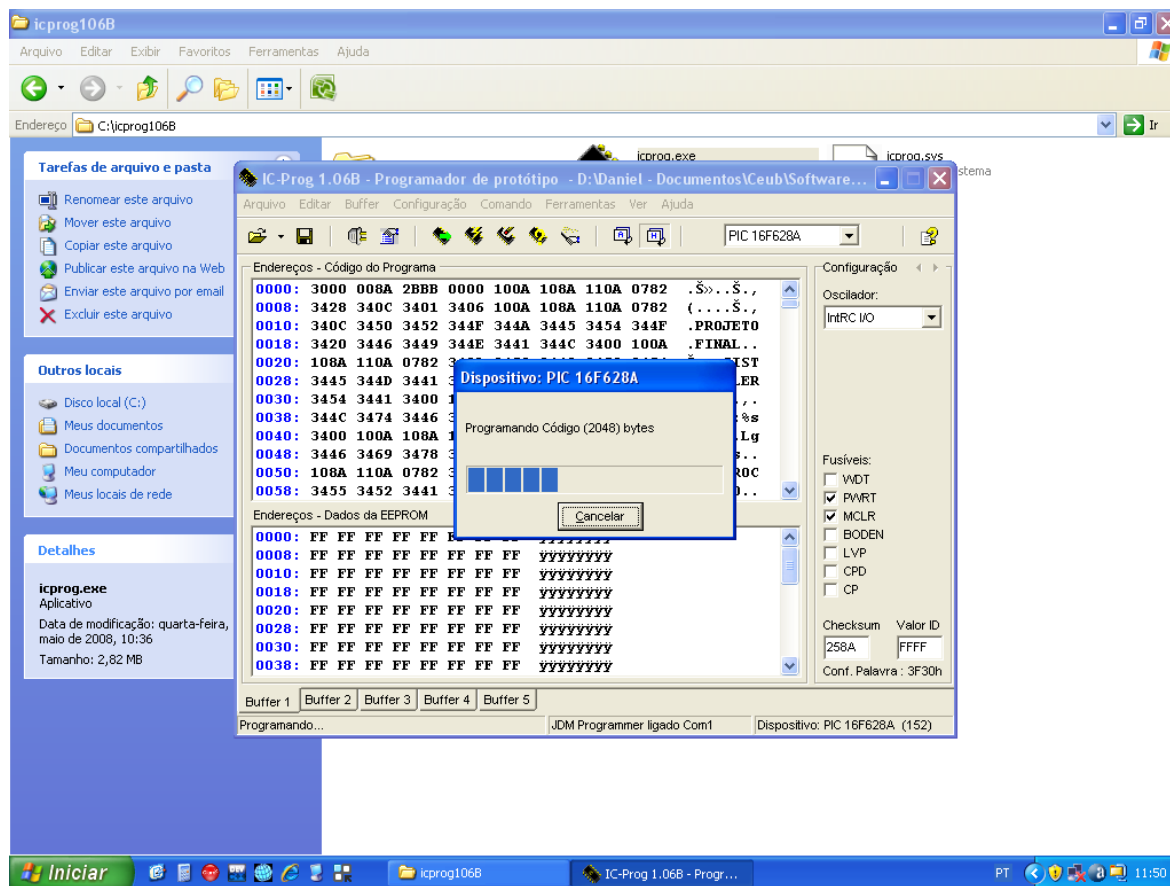


Figura 4.8 – Tela de gravação do ICProg  
(Fonte: O autor)

#### 4.4 – Problemas Encontrados

Um dos primeiros problemas encontrados foi no uso do kit de desenvolvimento. O primeiro kit utilizado foi o kit da Labit, com um PIC16F877A. As ligações entre os componentes do kit eram feitas através de fios ligando os pinos do PIC ao MAX232, ao LCD, a alimentação. Como não se conseguia a comunicação entre alguns componentes e o PIC, e até mesmo entre o microcontrolador e os módulos, ficava a dúvida se era problema de software ou problema de hardware, por algum componente queimado, ou mesmo problema

nas ligações feitas através dos fios. Depois de alguns testes, foi verificado que alguns componentes do kit não funcionavam, o que leva a concluir que um possível motivo para o não funcionamento é um mau contato nos conectores e soquetes do kit.

Como estava perdendo muito tempo testando onde estava o problema, resolvi trocar o uso desse kit da Labit e usar um Kit da ACEPIC, resolvendo esse problema, pois no kit ACEPIC, essas ligações eram soldadas e ligadas através de chaves.

Outra dificuldade foi conseguir a comunicação com o módulo GPS, pois as definições da USART interna não estavam funcionando, já que o registrador de recepção não estava sendo esvaziado, fazendo com que não conseguisse capturar a string. Daí resolvi implementar a USART através da biblioteca *usart.c*, sugerida pelo Fábio Pereira (Pereira, 2004), mas também não funcionou. Aí usei a biblioteca *rs232.c*, sugerida também pelo Fábio Pereira (Pereira, 2004), funcionando finalmente apenas fazendo as adaptações para as portas utilizadas no meu microcontrolador.

A intenção inicial era usar um módulo GSM, mas devido ao tempo, e pelo fato do uso do módulo ser apenas para realizar uma ligação, mudei para o uso de uma discadora, que resolveu o problema melhor que o módulo, pois ela discaria automaticamente para até 3 números, que poderiam ser alterados a qualquer momento sem a necessidade de reprogramar o PIC.

Outro problema foi quando a string era capturada e depois que terminava o tamanho da string, trazia algum “lixo”, ou seja, dados que estavam gravados na memória anteriormente e que não eram lidos como caracteres. Esses dados alteravam alguns valores das coordenadas. Para resolver esse problema foi colocado no código um teste para verificar se o tamanho da string era suficiente para chegar até o fim da longitude, o que resolveu esse problema, pois o sistema não considera a string válida enquanto ela não tiver um tamanho válido.

## **5. CAPÍTULO Nº 5 - APLICAÇÃO E DEMONSTRAÇÃO PRÁTICA DO MODELO PROPOSTO**

Neste capítulo é mostrado uma aplicação prática desse modelo proposto, mostrando que ele pode ser uma das soluções para o problema apresentado no Capítulo 2. Através de testes práticos foi mostrada a funcionalidade técnica e também o custo deste projeto.

### **5.1 - Apresentação da área de Aplicação do modelo**

Empresas de segurança vendem alarmes, bloqueadores, rastreadores, entre outros. Para prevenir problemas relativos à segurança de veículos, residências ou até mesmo das pessoas. Esse modelo pode ser fabricado e usado sozinho como um sistema de alerta ou pode ser usado em conjunto com algum outro equipamento como bloqueadores ou rastreadores.

Pelo fato desse modelo não precisar de que nenhuma operadora que controle o serviço, apenas que se tenha um celular com um chip GSM habilitado, ele torna-se um produto com um custo inferior aos oferecidos no mercado, por não ter a necessidade de pagamento de uma mensalidade, o que atrairá interesse de pessoas proprietárias de veículos.

### **5.2 – Descrição da Aplicação do Modelo**

A aplicação básica desse modelo é o monitoramento de um veículo e o envio de um alerta no caso do veículo realizar um movimento não autorizado. Ao estacionar o veículo em local em que o mesmo ficará sem vigilância por um longo período de tempo, o sistema do modelo proposto é ligado e começará a monitorar as coordenadas do veículo. Com o veículo parado, as alterações nas coordenadas são mínimas e o sistema tem uma tolerância para considerar um movimento. Testes foram realizados e mesmo com o veículo próximo a prédios ou árvores, essa variação ficou dentro da tolerância.

Quando o veículo entra em movimento não autorizado, esse movimento é detectado pela comparação das coordenadas e uma ligação é feita para o telefone cadastrado no chip. A

ligação é feita sem nenhum tipo de alarme para quem está no veículo. Só o dono receberá o alerta, e não haverá nenhum sinal sonoro que avise quem está no veículo.

O sistema deve ser instalado como um alarme convencional, em lugar escondido do carro. Somente o módulo GPS que deve ser colocado em um lugar do carro que esteja o menos bloqueado possível, para não atrapalhar a recepção do sinal e para diminuir ao máximo qualquer tipo de interferência.

Quando o sistema for ligado, deve-se ter em mente que se ficar debaixo de uma laje ou árvores muito densas pode prejudicar o funcionamento do modelo, pois o módulo de GPS deve ter o máximo de visibilidade do céu, já que os sinais são obtidos através de satélites.

Também deve ser lembrado que se deve ter o sinal para o celular usado, porque senão o alerta não poderá ser enviado. O sistema não tem uma memória para que faça o alerta posteriormente em caso do celular não ter sinal. O sinal para acionamento do celular é enviado no momento em que o evento acontece.

A alimentação pode ser feita através da bateria do veículo, observando a polaridade para que não queime alguns dos componentes. Deve-se dar atenção especial no caso do veículo ficar estacionado por longos períodos, pois se deve observar a autonomia da bateria, apesar do baixo consumo do sistema.

O consumo máximo dos principais componentes é mostrado na Tabela 5.1.

**Tabela 5.1 – Consumo máximo dos componentes utilizados**

(Fonte: O Autor)

<b>Componente</b>	<b>Consumo</b>
Módulo GPS ME-1000RW	70 mA
Discadora Westron e Celular	300 mA
Kit de desenvolvimento	100 mA
<b>Total</b>	<b>470 mA</b>



### 5.3 – Avaliação Global do Modelo

Esse modelo tem uma grande aplicabilidade no ramo da segurança, pois é um equipamento muito eficiente e tem suas vantagens no ponto de vista financeiro, pois um modelo como este, sendo produzido em escala industrial, poderá reduzir os valores apresentados na tabela 4.1., e se tornando também um modelo muito econômico, pois a partir do momento que o usuário adquire o sistema, o mesmo terá um gasto mínimo com o celular, podendo pagar apenas os valores para manter a linha. Quando acontecer um evento, as ligações de alerta podem ser feitas a cobrar, e sem necessidade de atendê-las, pois assim que o usuário receber uma ligação e identificar que a ligação é do celular ligado ao modelo, nem precisará atender para saber que aconteceu um evento.

Além disso, tem a possibilidade de acrescentar funções de rastreamento através do envio de SMS, apenas integrando um módulo GSM, já que as coordenadas para envio já são colocadas em variáveis, apenas precisando configurar os comando AT. Também podem ser inseridas funções de bloqueio ou escuta interna, através de interrupções a serem programadas no PIC utilizado ou em um com mais pinos.

Um dos pontos positivos mais relevantes desse modelo é o fato do alerta em caso de um evento ocorrer ser feito imediatamente após o evento ter ocorrido, mostrando uma funcionalidade dentro do esperado, permitindo tomar providências imediatas. Traz um ganho em relação aos modelos existentes por ser um modelo eficaz e econômico.

O módulo GPS utilizado tem uma margem de precisão muito boa, que vai de 5 metros em condições ideais com a maior visibilidade do céu até cerca de 10 metros, mesmo com alguns obstáculos parciais, o que aumenta a confiabilidade do sistema.

### 5.4 – Validação

Mostrando o uso prático desse modelo em um veículo, veremos a seguir os passos e figuras mostrando os testes e resultados obtidos, mostrando de forma prática a funcionalidade do modelo.

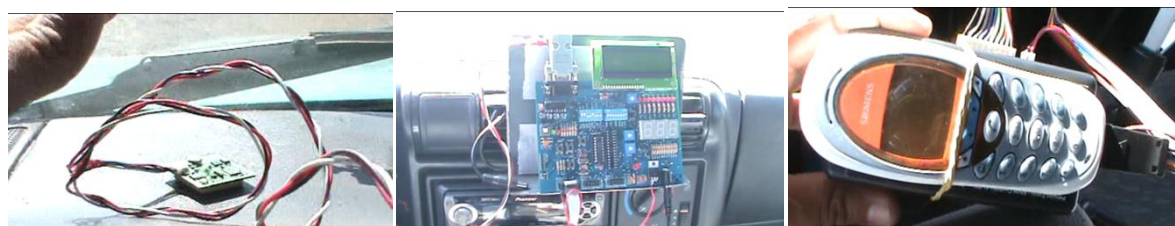
Na figura 5.1, vemos como pode ser conseguida a alimentação do modelo. No caso dos testes realizados, foi conseguida uma alimentação através da bateria do veículo, pois a

tensão de alimentação do kit é de 12,0 V, que também é a alimentação da discadora e do celular. Devido à necessidade de regular essa tensão, ela é regulada internamente no kit, através do regulador de tensão, para 5,0 V, que é a tensão de alimentação dos componentes do kit e do módulo GPS.



**Figura 5.1 - Conector para alimentação do modelo**  
(Fonte: O autor)

Para a realização dos testes, o módulo GPS foi colocado sobre o painel do veículo, e o kit com a discadora e o celular acondicionados no painel para melhor visualização, como é visto na figura 5.2. Ao ser ligado o conector ao acendedor de cigarros do carro, e com isso sendo alimentado o modelo, já começa automaticamente o seu funcionamento. O módulo liga, mostrando a mensagem que é mostrada na figura 5.3, e fica mostrando essa mensagem durante 30 segundos, que é o tempo necessário para que as coordenadas do GPS sejam enviadas e que sejam consideradas válidas, conforme especificações do fabricante e que foram confirmadas nos testes realizados.



**Figura 5.2 – Acondicionamento dos componentes do modelo no veículo**  
(Fonte: O autor)



**Figura 5.3 – Tela inicial do modelo**  
(Fonte: O autor)

Depois disso começa o monitoramento das coordenadas para se fixar as coordenadas para comparação, aparecendo a mensagem mostrada na figura 5.4. O sistema procura as coordenadas que sejam consideradas válidas, conforme foi explicado no capítulo anterior e quando encontradas, são colocadas nas variáveis para latitude e longitude fixas e são mostradas no LCD, conforme vimos na figura 5.5. As coordenadas ao mostradas sem os separadores de graus, minutos e segundos, devido o número possível de caracteres a serem escritos no LCD. No exemplo da figura, o valor da latitude é de 15°32'65.92".



**Figura 5.4 – Tela informando que está fixando as coordenadas**



**Figura 5.5 – Tela informando as coordenadas que foram fixadas(sem os separadores)**

Depois que são fixadas as coordenadas que servem de base para a comparação, começa a captura das coordenadas para comparar se houve alteração acima da tolerância. Enquanto são procuradas coordenadas válidas, é mostrada a mensagem conforme mostrado na figura 5.6. Quando forem encontradas coordenadas válidas, elas são mostradas no LCD, como pode ser visto na figura 5.7, e é feita a comparação. Se não for encontrada diferença acima da tolerância, é feita nova captura até que se encontre diferença.



**Figura 5.6 – Tela informando que está procurando coordenadas válidas**

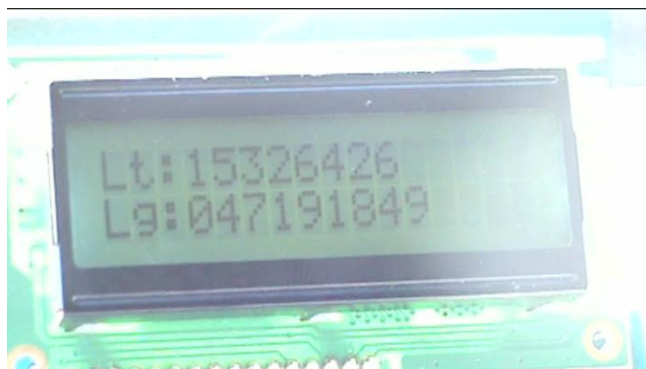
---



**Figura 5.7 – Coordenada capturada para comparação**

Foram feitas várias capturas, por vários minutos, em lugares diferentes, e enquanto o veículo permaneceu parado, as capturas eram feitas e não se encontravam diferenças acima da tolerância, conforme esperado. Ao colocar o veículo muito próximo a prédios ou árvores, essa precisão diminuía e as diferenças encontradas aumentavam, mas sempre ficando dentro da tolerância pré-estabelecida. As coordenadas só não foram conseguidas ao usar o modelo em ambiente fechado ou coberto por telhado ou laje, onde o módulo GPS não conseguia capturar o sinal dos satélites.

Depois de feitos estes testes com o veículo parado, testando a precisão e o funcionamento do modelo, partiu-se para os testes para detecção do movimento. Foi realizado um deslocamento com o veículo e assim que o veículo se deslocou cerca de 20 metros, foram capturadas as coordenadas e detectado o movimento, conforme vimos nas figuras 5.8 e 5.9.



**Figura 5.8 – Tela informando a coordenada depois do deslocamento**



**Figura 5.9 – Tela informando que o movimento foi detectado**

Assim que o movimento foi detectado, é acionada a discadora para que realize a ligação para o celular cadastrado no chip do celular acoplado, conforme é mostrado na figura 5.10, onde é feita a ligação para o número cadastrado em nome de “dono”. Enquanto é feita a ligação, é mostrada a mensagem que a ligação está sendo realizada para o dono, conforme é visto na figura 5.11, e mostra o número daquela ligação, já que o sistema realiza 5 ligações.



**Figura 5.10 – Tela do celular realizando a ligação para o dono do veículo**



**Figura 5.11 – Tela do LCD avisando que está realizando a ligação.**

A ligação é realizada durante 30 segundos, tempo necessário para que a ligação não caia na caixa postal do celular do dono. Na figura 5.12, mostra o celular do dono recebendo a ligação do celular acoplado na discadora. Como o dono conhece o número do celular acoplado a discadora não é necessário atender a ligação, pois o alerta já foi realizado. Depois que é passado o tempo de discagem, o sistema realiza nova captura das coordenadas e testa novamente. Encontrando diferença, realiza nova ligação, sempre informando o número daquela ligação, conforme vimos na figura 5.13. Quando o sistema realizar 5 ligações, é mostrada a mensagem que já foram realizadas as 5 ligações e que estará saindo do monitoramento, conforme é mostrado na figura 5.14.



**Figura 5.12 – Celular do dono do veículo recebendo a ligação**



**Figura 5.13 – Tela mostrando o número da ligação que está sendo realizada**



**Figura 5.14 – Tela informando que já foram realizadas 5 ligações e que está saindo do sistema**

Os testes foram realizados e o modelo mostrou-se com o desempenho dentro do esperado, realizando as funções programadas e dentro da tolerância pré-estabelecida. O seu funcionamento mostrou que o modelo tem aplicações práticas e resolveu o problema apresentado no capítulo 2.

## **6. CAPÍTULO Nº 6 - CONCLUSÃO**

### **6.1 - Conclusões**

Esse trabalho alcançou seu objetivo geral, pois consegue capturar as coordenadas do módulo GPS, analisar se houve alteração comparando com as coordenadas que foram fixadas e no caso de encontrar diferença, ligar para o dono enviando um sinal de alerta.

O protótipo foi implementado dentro do esperado, e consegue realizar a integração entre os componentes realizando as funções esperadas. Os resultados foram alcançados satisfatoriamente em todos os pontos destacados nos objetivos.

Um dos resultados obtidos mais relevantes se deve ao fato da análise das coordenadas acontecer de forma muito criteriosa, não dando margem para falsos positivos por motivo de software, trazendo um ganho muito grande na maneira de analisar os dados obtidos pelo padrão NMEA.

Uma vantagem desse projeto é que os números cadastrados no celular podem ser alterados, em caso de necessidade, sem que o microcontrolador seja reprogramado, só precisando alterá-los na agenda do chip do celular.

### **6.2 - Sugestões para Trabalhos Futuros**

Algumas aplicações poderiam ser incorporadas ao projeto ou mesmo poderia ser alterado para melhorias em algumas aplicações. A seguir algumas sugestões para projetos futuros.



### 6.2.1 - Envio das coordenadas para rastreamento

As coordenadas poderiam ser enviadas através de SMS, pelo uso de um módulo GSM ligado ao pino TX do PIC, já que o mesmo não está sendo usado nesse projeto. Essas mensagens seriam enviadas através de comandos AT, usando as latitude e longitude que ficam nas variáveis mostradas no código fonte.

### 6.2.2 – Interação com módulos de bloqueio do veículo

Um módulo GSM integrado ao sistema poderia receber SMS com comandos e uma senha, para realizar bloqueio da alimentação do veículo, assim que fosse detectado um movimento não autorizado. Esse sistema funcionaria com o mínimo de custo, pois o proprietário não gastaria com mensalidades, apenas com o valor do SMS.

### 6.2.3 – Instalação de um sistema de escuta ou gravação

Através de módulo GSM integrado ao sistema, assim que detectasse um movimento, acionaria um sistema de escuta ou gravação de imagens, para saber o que está ocorrendo dentro do carro. Para isso usaria o próprio sistema de escuta do módulo GSM, ou uma micro câmera instalada em um local estratégico.

## REFERÊNCIAS BIBLIOGRÁFICAS

ACEPIC. (2009 de Março de 2009). Kit de desenvolvimento ACEPIC 18\_28 (PIC16F628A). Acesso em 05 de Maio de 2010, disponível em ACEPIC Tecnologia e Treinamento Ltda: [http://www.acepicvirtual.com.br/produto-72-kit\\_de\\_desenvolvimento\\_acepic\\_18\\_28\\_%28pic16f628a%29](http://www.acepicvirtual.com.br/produto-72-kit_de_desenvolvimento_acepic_18_28_%28pic16f628a%29)

Amarante, D. (1 de Abril de 2008). Coordenadas Geograficas. Acesso em 07 de Abril de 2010, disponível em Geografia para todos: <http://www.google.com.br/imgres?imgurl=http://www.geografiaparatodos.com.http://geografiageralebiblica.blogspot.com/2008/04/coordenadas-geogrificas.html&h=265&w=425&sz=42&tbnid=l4Tr6RzHFgR3rM:&tbnh=79&tbnw=>

DatasheetME-1000RW. (20 de Janeiro de 2009). Technical Data Sheet ME-1000RW. Versão 1.2 .

Hasegawa, J. K., Galo, M., Monico, J. F., & Imai, N. N. (1999). SISTEMA DE LOCALIZAÇÃO E NAVEGAÇÃO APOIADO POR GPS. Acesso em 05 de Abril de 2010, disponível em Unesp: [http://www2.fct.unesp.br/dcartog/galo/pdf/1999\\_gps\\_cbc.pdf](http://www2.fct.unesp.br/dcartog/galo/pdf/1999_gps_cbc.pdf)

IBGE. (2010). Representação Cartográfica. Acesso em 02 de Junho de 2010, disponível em Instituto Brasileiro de Geografia e Estatística: [http://www.ibge.gov.br/home/geociencias/cartografia/manual\\_nocoas/representacao.html](http://www.ibge.gov.br/home/geociencias/cartografia/manual_nocoas/representacao.html)

INPE. (2010). Conceitos Cartograficos. Acesso em 12 de Maio de 2010, disponível em Ministerio da Ciencia e Tecnologia: <http://www.dpi.inpe.br/terraview/docs/pdf/ProjecaoCartografica.pdf>

Messias, A. R. (2004). Comunicação com a porta serial. Acesso em 13 de Maio de 2010, disponível em Instituto Federal de Educação, Ciência e Tecnologia: <http://www.cefetsp.br/edu/garabed/c/serial/Porta%20Serial.htm>

Microchip. (2007). datasheet PIC16F628A.

NMEA. (2008). National Marine Electronics Association. Acesso em 22 de Abril de 2010, disponível em The NMEA mission: <http://www.nmea.org/>

Pedro Ferreira - Estado de Minas. (11 de Agosto de 2009). Índice de recuperação de carros roubados cai 23% em BH. Acesso em 01 de Abril de 2010, disponível em Portal Uai: [http://www.uai.com.br/UAI/html/sessao\\_2/2009/08/11/em\\_noticia\\_interna,id\\_sessao=2&id\\_noticia=122336/em\\_noticia\\_interna.shtml](http://www.uai.com.br/UAI/html/sessao_2/2009/08/11/em_noticia_interna,id_sessao=2&id_noticia=122336/em_noticia_interna.shtml)

Pereira, F. (2004). Microcontroladores PIC Programação em C. São Paulo: Érica.

R&D. (2007). Conversor de coordenadas. Acesso em 04 de Junho de 2010, disponível em R&D Tecnologia de Software: <http://www.rdtec.com.br/rdgeomg/localmaster.htm>

Regis Alex - Tribuna do Norte. (13 de Abril de 2008). A cada 3 horas e meia um automóvel é roubado no RN. Acesso em 1 de Abril de 2010, disponível em Tribuna do Norte: <http://tribunadonorte.com.br/noticia.php?id=72507>

System, M. I. (2007). RS232 Pinouts designation. Fonte: Machine Informatio System: [http://www.machine-information-systems.com/RS232\\_Pinouts.html](http://www.machine-information-systems.com/RS232_Pinouts.html)

Westron. (2010). Acesso em 31 de Maio de 2010, disponível em Westron Equipamentos Eletrônicos: <http://www.westron.com.br/www/home.php?W=1280>

## APÊNDICE

A seguir encontra-se o código fonte do programa que faz toda a integração entre o microcontrolador PIC, o módulo GPS e a discadora com o celular.

### Código fonte do Programa

```

/*
*****

                PROJETO FINAL - Eng Computação - UniCEUB
                1o. Semestre 2010

                DANIEL ALVES FERREIRA
                RA: 2041629-9

*****

*****
                SISTEMA DE ALERTA INTEGRANDO MICROCONTROLADOR, GPS E CELULAR
*****

*/

#include <16F628A.h>                //Definições do microcontrolador utilizado

#include <delay.h>                //biblioteca de delay para geração de atraso
#include <INTRC_IO,NOWDT,PUT,NOBROWNOUT,NOLVP>

#include <rs232.c>                // biblioteca para controle da comunicação serial

#include "LCD_4B.c"                //biblioteca para controle do LCD
#include <string.h>                //biblioteca para comparações ações nas strings
#include <stdlib.h>                //biblioteca para conversão de tipos

*****
// Função main() - PROGRAMA COMEÇA APARTIR DA MAIN()
*****

void main (){

    char captura;
    char latlong[80], lat[10], longit[11],latFIXA[10], longitFIXA[11];
    int pos=0, latpos=0, tam=0, ContLig=0;
    signed int Vlat=0, VlatFIXA=0;
    signed int Vlongit=0, VlongitFIXA=0;

```

```

signed int DifLat=0, DifLongit=0;
char *pass1,*pass2;
char teste[8]={"$GPRMC,"};

lcd_ini(); //inicialização do LCD
rs232_inicializa (); //inicialização do RS232
OUTPUT_LOW(PIN_A7); //coloca o pino A7 em ZERO, para nao enviar sinal
//em momento indesejado

printf(lcd_escreve,"f PROJETO FINAL ");
printf(lcd_escreve,"n DANIEL ALVES ");
delay_ms(30000); //tempo para esperar o "aquecimento" do GPS

while(tam<=25){ //tamanho minimo para se capturar latitude e longitude

printf(lcd_escreve,"f SISTEMA ALERTA ");
printf(lcd_escreve,"n FIXANDO... ");

delay_ms(2000);

for(pos=0; pos<79;pos++){

captura=rs232_recebe(); //captura pela comunicacao serial
latlong[pos]=captura; //coloca o caractere na string
}

pass1=strstr(latlong, teste); //passagem para testar se eh a linha RMC
pass2=strchr(pass1,'A'); //passagem para testar se os dados sao validos
tam=strlen(pass2); //tamanho do A ate o fim da string

if(tam>25){ //so entra se for o tamanho minino exigido

for(pos=0,latpos=2; pos<8 ;latpos++,pos++){
if(pass2[latpos]!='.'){ //para retirar os pontos da string
latpos++;
}
latFIXA[pos]=pass2[latpos];
}
latFIXA[pos]='\0'; //para se definir o fim da string

for(pos=0,latpos=14; pos<9;latpos++,pos++){
if(pass2[latpos]!='.'){ //para retirar os pontos da string
latpos++;
}
longitFIXA[pos]=pass2[latpos];
}
longitFIXA[pos]='\0'; //para se definir o fim da string

printf(lcd_escreve,"f"); //Limpa Display
printf(lcd_escreve,"fLtFix:%s", latFIXA); // Escreve Lat Fixa
printf(lcd_escreve,"nLgFix:%s", longitFIXA); // Escreve long Fixa

```

```

    delay_ms(8000);                //aguardar 8 segundos
} //fim do IF(tam>25)
} //fim do While(tam<=25)

while(ContLig<5){                //só entra enquanto as ligações pro dono forem < 5
tam=0;

while(tam<=25){
printf(lcd_escreve,"f PROCURANDO ");
printf(lcd_escreve,"\n COORDENADAS ");

delay_ms(2000);

for(pos=0; pos<79;pos++){

    captura=rs232_recebe();        //captura pela comunicação serial
    latlong[pos]=captura;          //coloca o caractere na string
}

pass1=strstr(latlong,teste);      //passagem para testar se eh a linha RMC
pass2=strchr(pass1,'A');          //passagem para testar se os dados sao validos
tam=strlen(pass2);                //tamanho do A ate o fim da string

if(tam>25){                       //so entra se for o tamanho minino exigido

for(pos=0,latpos=2; pos<8 ;latpos++,pos++){
    if(pass2[latpos]==''){          //para retirar os pontos da string
        latpos++;
    }
    lat[pos]=pass2[latpos];
}
lat[pos]='\0';                    //para se definir o fim da string

//pass3=strchr(pass2,'S');

for(pos=0,latpos=14; pos<9;latpos++,pos++){
    if(pass2[latpos]==''){        //para retirar os pontos da string
        latpos++;
    }
    longit[pos]=pass2[latpos];
}
longit[pos]='\0';                //para se definir o fim da string

printf(lcd_escreve,"\f"); //Limpa Display
printf(lcd_escreve,"\fLt:%s", lat); // Escreve a Lat atual
printf(lcd_escreve,"\nLg:%s", longit); // Escreve a Long atual
delay_ms(8000);                //aguardar 8 segundos
} //fim do IF(tam>25)

```

```

} //fim do While(tam<=25)

Vlat=atoi(lat); //converte string em signed int
VlatFIXA=atoi(latFIXA); //converte string em signed int
Vlongit=atoi(longit); //converte string em signed int
VlongitFIXA=atoi(longitFIXA); //converte string em signed int

DifLat=(Vlat-VlatFIXA); //diferença entre a lat atual e a fixa
DifLongit=(Vlongit-VlongitFIXA); //diferença entre a long atual e a fixa

if ( (DifLat>50) || (DifLat<-50) || (DifLongit>50) || (DifLongit<-50) ){ //compara se a diferença nao é maior
                                                                    //que o minino estabelecido
    ContLig++;

    printf(lcd_escreve,"%f"); //Limpa Display
    printf(lcd_escreve,"%f MOVIMENTO ");
    printf(lcd_escreve,"%n DETECTADO!! ");
    delay_ms(3000);

    printf(lcd_escreve,"%f"); //Limpa Display
    printf(lcd_escreve,"%f LIGACAO N.%d ", ContLig);
    printf(lcd_escreve,"%n PARA O DONO! ");
    delay_ms(3000);

    OUTPUT_HIGH(PIN_A7); //coloca o pino em 1, para enviar o sinal pra discadora atraves do pino RA7
    delay_ms(30000); //aguarda 30 segundos
    OUTPUT_LOW(PIN_A7); //volta o pino para estado logico ZERO.
    delay_ms(30000); //aguarda 30 segundos

    if( ContLig==5){ //testa se já efetuou as 5 ligações
        printf(lcd_escreve,"%f");
        printf(lcd_escreve,"%f 5 LIGACOES!!! ");
        printf(lcd_escreve,"%n SAINDO!!! ");

        delay_ms(10000);
        printf(lcd_escreve,"%f"); //Limpa Display

        ContLig++; //incrementa as ligações para sair, pois a ContLig fica igual a 6
    }
}

} // fim do WHILE(ContLig<5)

} // fim do VOID MAIN

```

## ANEXOS

A seguir a biblioteca usada para conseguir a comunicação serial assíncrona.

### Biblioteca *rs232.c*

```

/*****/
/* RS232.C Biblioteca de Comunicação serial assíncrona por */
/* software */
/* */
/* Autor: Fabio Pereira */
/* */
/* Velocidade : 9600 Bps, 1 Start, 1 Stop, Sem paridade */
/* */
/* Estas rotinas funcionam sem modificação para velocidades de 9600 e */
/* 19200 Bps. */
/* */
/*****/

// Definições de comunicação
/*
Para alterar a velocidade de comunicação, basta alterar o valor da
constante baud_rate
*/
#ifdef baud_rate
const long int baud_rate = 9600;
#else
const int tempo_bit_dado = 1000000/ baud_rate - 10; // tempo do bit de dado
const int tempo_bit_start = 1500000/ baud_rate; // tempo do bit de start

// Definições dos pinos de comunicação
// Para utilizar outros pinos, basta incluir novas definições
// no arquivo do programa onde esta biblioteca for incluída
#define pino_tx = 0x06.0 // pino de transmissão é o RB0
#define pino_rx = 0x06.1 // pino de recepção é o RB1
#define dir_tx = 0x86.0 // direção do pino de tx
#define dir_rx = 0x86.1 // direção do pino de rx

void rs232_inicializa (void)
{
dir_tx = 0; // pino de tx como saída
pino_tx = 1; // coloca o pino de tx em nível 1
dir_rx = 1; // configura o pino de rx como entrada
}

void rs232_transmite (char dado)
{
boolean result;
int conta;
// primeiro o bit de start
pino_tx = 0;
delay_us(tempo_bit_dado); // aguarda o tempo de start
conta = 8; // são 8 bits de dados
while (conta)
{
// desloca o dado à direita e dependendo do resultado
// seta ou não a saída

```



```
if (shift_right ( &dado, 1, 0)) pino_tx=1; else pino_tx=0;
delay_us ( tempo_bit_dado); // aguarda o tempo do bit de dado
conta--; // decreenta um no número de bits a transmitir
}
pino_tx = 1; // seta a saída TX (bit de stop)
delay_us(tempo_bit_dado); // aguarda o tempo de 1 bit
}

char rs232_recebe (void)
{
int conta,dado;
while (pino_rx); // aguarda o bit de start
delay_us(tempo_bit_start); // aguarda o tempo de start
conta = 8; // são 8 bits de dados
dado = 0; // apaga o dado recebido
while (conta)
{
shift_right( &dado, 1, pino_rx); // insere o bit recebido
// deslocando à direita a
// variável com o dado
// recebido
delay_us(tempo_bit_dado); // aguarda o tempo de 1 bit
conta--; // decreenta o número de bits
}
delay_us(tempo_bit_dado); // aguarda o tempo de 1 bit
return dado; // retorna o dado recebido
}
```