



CENTRO UNIVERSITÁRIO DE BRASÍLIA – UNICEUB
FATECS – FACULDADE DE TECNOLOGIA E CIÊNCIAS SOCIAIS APLICADA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

DANIEL ATAIDE LEITE CAMPOS

SISTEMA DE ALARME DE INTRUSÃO UTILIZANDO O SMS

Orientadora: Professora MS Maria Marony S. F. Nascimento

Brasília
Junho, 2010

DANIEL ATAÍDE LEITE CAMPOS

SISTEMA DE ALARME DE INTRUSÃO UTILIZANDO O SMS

Trabalho apresentado ao Centro Universitário de Brasília como pré-requisito para a obtenção de Certificado de Conclusão do Curso de Engenharia de Computação.

Orientadora: Professora MS Maria Marony S. F. Nascimento

Brasília
Julho, 2010

DANIEL ATAÍDE LEITE CAMPOS

SISTEMA DE ALARME DE INTRUSÃO UTILIZANDO O SMS

Trabalho apresentado ao Centro Universitário de Brasília como pré-requisito para a obtenção de Certificado de Conclusão do Curso de Engenharia de Computação.

Orientadora: Professora MS Maria Marony S. F. Nascimento

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 Fernandez
Coordenador do Curso

Banca Examinadora:

Prof. Maria Marony Sousa Farias Nascimento, mestre.
Orientador

Prof. Antonio Barbosa Junior, especialista.
UniCEUB

Prof. Francisco Javier de Obaldia Diaz, mestre.
UniCEUB

Prof. Vera Lucia Farini Alves Duarte, mestre.
UniCEUB

DEDICATÓRIA**Dedico**

*Aos meus pais Jorge e Cristina e ao meu irmão Diogo,
pela força, paciência, incentivo e apoio em todas as situações,
na busca desta minha realização tão desejada.
Compartilho a vitória com meus tios, primos, padrinhos e amigos.*

AGRADECIMENTOS

A Deus todo poderoso, pela vida.

A professora MS Maria Marony S. F. Nascimento, minha orientadora, pelo respeito, idéias criativas, paciência, amizade e perseverança que me inspiraram a prosseguir.

Ao professor MS Francisco Javier De Obaldía, da disciplina Projeto Final, pela grande contribuição ao meu trabalho.

Ao professor MS Abiezer Amarília Fernandes, Coordenador Acadêmico do Curso de Engenharia de Computação da FATECS.

Aos professores do Curso de Engenharia de Computação da FATECS.

Aos meus colegas de Curso pela troca de experiências enriquecedoras, sugestões e ajuda nas várias fases do trabalho.

Aos funcionários da secretaria da FATECS.

Àqueles que, de alguma maneira, contribuíram com este trabalho e para a sua realização e não estão, aqui, nominalmente citados.

Apesar de todo apoio e auxílio recebidos, eventuais equívocos e imperfeições – e eles certamente existem – são de responsabilidade exclusiva do autor.

SUMÁRIO

LISTA DE FIGURAS.....	VII
LISTA DE SIGLAS.....	IX
RESUMO.....	XI
ABSTRACT.....	XII
1 - INTRODUÇÃO.....	12
2- APRESENTAÇÃO DO PROBLEMA.....	15
3 - BASES METODOLÓGICAS PARA RESOLUÇÃO DO PROBLEMA.....	18
4 - MODELO PROPOSTO.....	40
4.1 – Apresentação Geral do Modelo Proposto.....	40
4.2 – Descrição das Etapas do Modelo.....	41
5 - APLICAÇÃO DO MODELO PROPOSTO.....	61
5.1 - Apresentação da área de Aplicação do modelo.....	61
5.2 – Descrição da Aplicação do Modelo.....	61
5.3 – Avaliação Global do Modelo.....	62
6 - CONCLUSÕES.....	64
6.1 - Conclusões.....	64
6.2 - Sugestões para Trabalhos Futuros.....	65
REFERÊNCIAS	66
ANEXOS	68

LISTA DE FIGURAS

Figura 2.1 – Topologia básica de um sistema de alarme.....	16
Figura 3.1 – Funcionamento da Arquitetura GSM.....	20
Figura 3.2 – Modem G24 GSM da iTech.....	23
Figura 3.3 – Estrutura básica de uma linha de comando AT.....	25
Figura 3.4 – <i>Microsoft Hyperterminal Versão 5.1</i>	25
Figura 3.5 – Microcontrolador Intel da família 8051.....	27
Figura 3.6 – Layout padrão do microcontrolador 8051 – Encapsulamento DIP40.....	30
Figura 3.7 – Diagrama de blocos do microcontrolador 8051.....	32
Figura 3.8 – Ambiente de Desenvolvimento Keil μ Vision3 V3.80.....	36
Figura 3.9 – Tela principal do Proteus 6.2 ISIS Professional.....	37
Figura 3.10 – <i>Software</i> para gravação do arquivo no microcontrolador da família 8051 – <i>SPI Flash Programmer Versão 3.7</i>	38
Figura 3.11 - Cabo serial RS-232 db9.....	39
Figura 4.1 – Circuito de proximidade desenvolvido em <i>Protoboard</i>	41
Figura 4.2 – Conexão utilizando um cabo flat de 10 vias.....	43
Figura 4.3 – Esquema elétrico.....	44
Figura 4.4 – Criação de um novo projeto.....	45
Figura 4.5 – Nomear e salvar pasta e projeto.....	46
Figura 4.6 – Fabricante e modelo do microcontrolador a ser utilizado.....	46
Figura 4.7 – Copiando para o projeto o arquivo do Startup.....	47
Figura 4.8 – Criando um novo arquivo.....	47
Figura 4.9 – Anexo do arquivo ao projeto.....	48
Figura 4.10 – Habilitar a geração do arquivo em hexadecimal.....	48
Figura 4.11 – Compilação do arquivo escrito em linguagem C.....	54
Figura 4.12 – Gravação do arquivo no microcontrolador AT89S52 que se encontra no Kit8051LS.	54
Figura 4.13 – Acendimento do LEDs gerados pela interrupção externa.....	55
Figura 4.14 – SIM Card e o Modem G24.	55
Figura 4.15 – Realizada a conexão, são realizadas as configurações e testes para a comunicação com o <i>Hyperterminal</i>	56
Figura 4.16 – Configuração dos parâmetros necessários para a comunicação com o modem no envio e recebimento de mensagens e os comandos executados no	

<i>hyperterminal</i> , com as respectivas respostas de confirmação pelo modem.....	57
Figura 4.17 – Armazenando a seguinte mensagem: “INTRUSAO DETECTADA!”.....	58
Figura 4.18 – Envio de SMS através do Hyperterminal.....	59
Figura 4.19 - Integração entre os dispositivos: Circuito com sensor, Kit8051LS que contém o microcontrolador AT89S52, o <i>modem</i> e dois aparelhos de telefone celular.....	59
Figura 4.20 – Recebimento do SMS que foi enviado pelo modem G24, após a detecção do sensor do circuito de proximidade.	60

LISTA DE SIGLAS

3G	Terceira Geração
ALE	Address Latch Enable
ANATEL	Agência Nacional de Telecomunicações
ANSI	American National Standards Institute
API	Aplication Programming Interface
ASCII	American Standard Code for Information Interchange
AT	Hayes AT Commands
AT&T	American Telephone and Telegraph
CDMA	Code Divison Multiple Access
CI	Circuito Integrado
CPU	Central Processing Unit
DCE	Data Circuit-terminating Equipment
DIP	Dual in Pack
DTE	Data Terminal Equipment
EA	External Access Enable
EDGE	Enhanced Data Rates for GSM Evolution
EEPROM	Electrically-Erasable Programmable Read-Only Memory
EGPRS	Enhanced General Packet Radio Service
GSM	Global System for Mobile Communications
GMSC	Gateway Mobile Switching Center
GPRS	General Packet Radio Service
HLR	Home Location Register
IDE	Integrated Development Environment
ISDN	Integrated Service Digital Network
IMSI	International Mobile Subscriber Identity
IP	Internet Protocol
ISP	In-System Programmable
ISUP	ISDN User Part
LAI	Location Area Identity
MAP	Mobbility Application Part
MSC	Mobile Switching Center

MSISDN	Mobile Station Integrated Service Digital Network
PC	Personal Computer
PCB	Printed Circuit Board
PSEN	Program Store Enable
QFP	Quad Flat Pack
RAM	Random Access Memory
ROM	Read-Only Memory
SCCP	Signalling Connection Control Part
SFR	Special Function Register
SIM	Subscribe Identity Module
SMP	Serviço Móvel Pessoal
SMS	Short Message Service
SMSC	Short Message Service Center
SMS-MO	Short Message Service – Mobile Originated
SMS-MT	Short Message Service – Mobile Terminated
SoC	System-on-Chip
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
TMSI	Temporary Mobile Subscriber Identity
UART	Universal Asynchronous Receiver/Transmitter
UCP	Unidade Central de Processamento
ULA	Unidade Lógica e Aritmética
USB	Universal Serial Bus
VLR	Visitor Location Register
VSM	Virtual System Modelling

RESUMO

O presente estudo teve como objetivos: a) desenvolver e implementar um sistema de alarme utilizando a troca de dados por meio da rede de telefonia celular; b) promover a interligação entre o *modem* de dados, o circuito de proximidade infravermelho e o microcontrolador 8051 e o telefone celular; c) desenvolver um circuito que utilize um sensor que seja capaz de identificar a presença de um ser humano no ambiente “protegido”; d) enviar um sinal ao microcontrolador para que o mesmo identifique, processe, execute e envie uma informação para o *modem* de dados; e) fazer com que o *modem* envie uma mensagem (SMS - *Short Message Service*), cujo número do aparelho telefônico móvel já está previamente armazenado, alertando-o sobre a intrusão em seu ambiente. A seguir, apresenta-se o problema e descrevem-se todas as bases metodológicas para a resolução deste problema. Na sequência, apresenta-se o modelo proposto com a descrição detalhada das etapas do projeto. Finalmente, a aplicação do modelo proposto, as áreas de aplicação do projeto e descrição de sua aplicação. Entre as sugestões de estudos adicionais para trabalhos futuros estão o desenvolvimento e implementação de um sistema de alarme de controle relacionado à abertura e o fechamento de entradas ou saídas de um ambiente, como portas e/ou janelas, bem como o desenvolvimento de um sistema capaz de se comunicar com uma rede de sensores sem fio, possibilitando a aquisição da informação por meio da rede de sensores e envio dos resultados a uma central que trabalhe com o monitoramento de grandes extensões territoriais e que necessite o controle em diferentes localidades de variáveis como umidade, temperatura e pressão. Do ponto de vista acadêmico, espera-se ter contribuído para o debate teórico e prático em torno do tema do projeto, bem como estimulado a realização de novos projetos, contribuindo assim para o avanço da engenharia.

Palavras Chave: *Modem*, SMS, 8051, Sensor, Sistema de Alarme de Intrusão.

ABSTRACT

This study aimed to: a) develop and implement an alarm system using the data exchange through mobile phone network, b) promote the interconnection between the data modem, the circuit of microcontroller and infrared proximity 8051, and the cell phone, c) develop a circuit that uses a sensor that is capable of identifying the presence of a human environment "protected", and d) sending a signal to the microcontroller so that it identify, process, execute and submit information to the data modem; e) cause the modem to send a message (SMS – Short Message Service), the number of mobile handset is already stored in advance, alerting you to the intrusion into their environment. The following is presented and described the problem to all the methodological bases for solving this problem. In the following, presented the proposed model with detailed description of the stages of the project. Finally, the application of the proposed model, the application areas of project and description of your application. Among the suggestions for further studies for future work are the development and implementation of an alarm system control related to the opening and closing entrances or exits an environment such as doors and / or windows, as well as developing a system capable of communicate with a network of wireless sensors, enabling the acquisition of information through the network of sensors and sending the results to a central monitor that works with large tracts of land and requiring the control in different localities of variables such as humidity, temperature and pressure. From an academic standpoint, it is expected to have contributed to the theoretical and practical debate on the theme of the project and encouraged the creation of new projects, thus contributing to the advancement of engineering.

Keywords: *Modem*, SMS, 8051, Sensor, Intrusion Detect System

CAPÍTULO 1 – INTRODUÇÃO

1.1 – Apresentação do Problema

Atualmente, há uma necessidade de se garantir maior segurança aos ambientes comerciais e residenciais para se manter e se preservar a integridade das pessoas e dos seus bens materiais. O uso de recursos tecnológicos para o desenvolvimento de um sistema de alarme via SMS ajudaria a detectar a intrusão.

– Mas quais os tipos de ferramentas e dispositivos poderiam ser empregados para que esse sistema fosse implementado?

1.2 – Objetivos do Trabalho

Objetivo Geral – desenvolver e implementar um sistema de alarme utilizando a troca de dados (SMS) por meio da rede de telefonia celular.

Objetivos Específicos:

- a. Promover a interligação entre o *modem* de dados, o circuito de proximidade infravermelho e o microcontrolador 8051, e o telefone celular;
- b. Desenvolver um circuito que utilize um sensor que seja capaz de identificar a presença de um ser humano no ambiente “protegido”;
- c. Enviar um sinal ao microcontrolador para que o mesmo identifique, processe, execute e envie uma informação para o *modem* de dados;
- d. Fazer com que o *modem* envie uma SMS, cujo número do aparelho telefônico móvel já está previamente armazenado, alertando-o sobre a intrusão em seu ambiente.

1.3 – Justificativa e Importância do Trabalho

O crescente aumento no uso da comunicação, voz e dados, por meio da rede de telefonia celular e a vasta utilização dos microcontroladores na criação e implementação de sistemas, tem sido utilizadas em diversas áreas como, segurança, automação, entretenimento, militar, entre outras.

As utilizações dessas tecnologias vêm se tornando cada vez mais freqüentes, principalmente, em relação ao custo benefício, no qual é possível aliar baixo custo aos componentes utilizados, confiabilidade e flexibilidade na manutenção dos sistemas criados a partir da integração das telecomunicações e dos circuitos que empregam microcontroladores.

1.4 – Escopo do Trabalho

Será abordado, no desenvolvimento deste projeto, o uso de tecnologias tais como: a troca de dados por meio de mensagens SMS, utilizando a rede GSM de telefonia celular; a utilização do circuito contendo um sensor infravermelho; e a integração destes com o microcontrolador da família 8051.

Não foram considerados outros tipos de sensores, microcontroladores e o uso de outras linguagens de programação.

1.5 – Resultados Esperados

Detectar, informando sobre a invasão, uma vez ativado o *led* infravermelho, bem como a ativação de uma sirene e o envio e recebimento de dados, por meio da rede GSM de telefonia celular.

1.6 – Estrutura do Trabalho

A estrutura principal está dividida em três capítulos, conforme breve descrição apresentada a seguir.

No capítulo 2 apresenta uma descrição profunda do problema que se pretende resolver, bem como os fatores, parâmetros e ambientes associados.

No capítulo 3 estão apresentados os principais conceitos teóricos relativos aos dispositivos empregados, bem como a comunicação com esses dispositivos e a apresentação das ferramentas utilizadas, explicando o funcionamento das mesmas e justificando o emprego de tais ferramentas, seja hardware ou software.

No capítulo 4 está apresentado o modelo proposto, utilizando fluxogramas, diagramas, softwares. Implementação do sistema, bem como a explicação do funcionamento, a análise dos resultados esperados e demonstrações práticas.

No capítulo 5 está apresentada a aplicação prática envolvendo um caso real, tendo como objetivo principal mostrar a viabilidade da proposta de resolução sugerida no trabalho, bem como permitir que novos conhecimentos sejam incorporados.

No capítulo 6 são colocadas as conclusões do trabalho como um todo e não apenas sua aplicação, bem como os objetivos geral e específicos apresentados na introdução do trabalho alcançados, total ou parcialmente, e em quais subitens. Também é avaliado se os resultados obtidos foram satisfatórios, ressaltado as principais vantagens e limites do modelo proposto.

CAPÍTULO 2 – APRESENTAÇÃO DO PROBLEMA

O excesso de intrusão a determinados ambientes residenciais e comerciais está acarretando a necessidade de se desenvolver outros tipos de sistemas de alarme utilizando diferentes dispositivos, de diferentes tecnologias, para a construção de novos sistemas oferecendo uma maior segurança ao local a ser protegido.

Alguns sistemas atuais de segurança não intimidam tanto uma intrusão. Se algum “intruso” quer adentrar em ambientes residenciais e comerciais encontrará uma maneira de fazê-lo, pois normalmente ele já conhece os principais modelos de sistemas disponíveis e como eles funcionam.

Usualmente, há dois tipos de sistemas de segurança: um monitorado, que alerta uma Central, toda vez que o alarme dispara, mantendo imediatamente contato com o responsável, para verificar e informar sobre a situação, sempre por meio de ligações telefônicas. Se o responsável não atender eles seguem imediatamente para verificar *in loco*.

Uma desvantagem de um sistema monitorado é que, o alarme ao ser acionado, proporciona ao “intruso” a informação de que há um sistema de segurança, o que faz com que ele escape levando alguns bens de valor. Mas, nem sempre isso pode vir a acontecer, pois o “intruso” poderá se assustar e fugir sem levar nada.

Se a central não recebe a senha correta ou o telefone do responsável não atende, ela contata a polícia, que levará alguns minutos para chegar ao local. Um “intruso” experiente pode escapar com alguns bens de valor, num curto espaço de tempo.

O outro sistema é o não monitorado que, por meio de um alarme, toca uma sirene alta e acende luzes de alerta, tanto no ambiente interno como no ambiente externo, chamando a atenção da vizinhança que, normalmente, avisa a Polícia.

Neste sistema os “intrusos” se assustam e saem correndo levando algum bem mais próximo. A desvantagem é a de que os vizinhos, em geral, não se envolvam por receio de retaliação. Outra desvantagem é que este sistema só pode ser utilizado em centros urbanos, com grande vizinhança.

A necessidade de se garantir maior segurança aos ambientes comerciais e residenciais para manter e preservar a integridade das pessoas e dos seus bens materiais, tem sido um grande desafio para todos os envolvidos com a Segurança Pública e Privada. Desenvolver um sistema eficaz, de baixo custo, com a utilização de ferramentas simples para o usuário final, como o Serviço de Mensagens Curtas (SMS – Short Message Service) é o principal objetivo deste trabalho.

O conceito de SMS foi inventado pelo engenheiro finlandês Matti Makkonen (*Mobile Internet for Dummies*, 2008), em meados da década de 1980. O Serviço de Mensagens Curtas permite aos usuários enviarem e receberem mensagens a partir de um aparelho de telefone celular.

Uma arquitetura básica de um sistema de alarme, figura 2.1, pode ser descrita em quatro blocos fundamentais que estão interligados. Cada bloco desempenha uma função específica, detectando, acionando e, com os dados adquiridos, delatando ou informando sobre a ocorrência de que houve uma invasão, para que o objetivo principal de um sistema possa ser alcançado com sucesso.

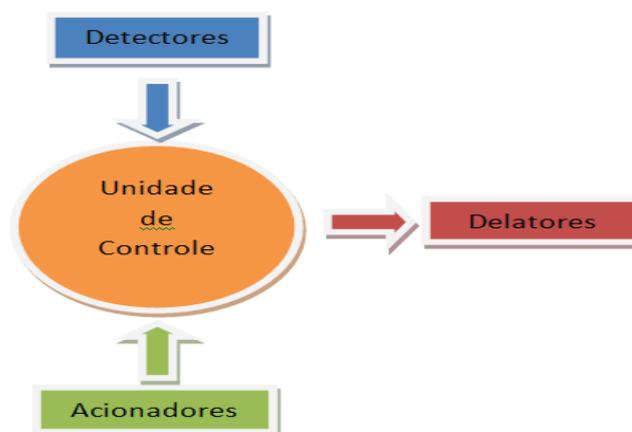


Figura 2.1 – Topologia básica de um sistema de alarme. Fonte (AUTOR DO PROJETO,2010)

No desenvolvimento deste projeto **o detector será um sensor infravermelho**, que utilizará o princípio da reflexão difusa no seu funcionamento. **Na unidade de controle** estará presente **o microcontrolador**. **O acionador** será o **circuito que contém o sensor**. **O delator** será representado **pelo alto-falante e pelo *modem***.

O circuito a ser montado, que contém o sensor de proximidade infravermelho e o microcontrolador que se comunica com o *modem*, através da comunicação serial. Outra forma de comunicação que está envolvida é a rede GSM de telefonia celular, no envio e recebimento de mensagens por SMS, entre o modem e o telefone celular.

O celular tem a função de receber uma mensagem (SMS) se o sistema foi acionado. O *modem*, por sua vez, deve estar preparado para que, ao receber um comando do circuito microcontrolado, analisa e encaminha as mensagens ao destino previamente estabelecido.

A mobilidade que a rede de telefonia celular GSM proporciona, através da utilização do aparelho de telefone celular, e o uso associado na implementação de um sistema de segurança, possibilita a informação sobre o mesmo de qualquer lugar, a qualquer hora, quase que instantaneamente.

De acordo com a Agência Nacional de Telecomunicações – ANATEL (<http://www.anatel.gov.br/Portal/exibirPortalInternet.do>), foram habilitados 1.655.637 celulares em abril de 2010, um crescimento de 0,92% em relação a março de 2010. O Brasil chega a 180.765.438 de acessos do Serviço Móvel Pessoal (SMP) e densidade de 93,80 acessos por 100 habitantes, com crescimento de 0,85% sobre o mês anterior. O crescimento nos quatro primeiros meses do ano é o maior da série histórica. Do total de acessos, 159.689.505 estão relacionados à utilização da tecnologia GSM, resultando numa participação de 88,34%.

CAPÍTULO 3 – BASES METODOLÓGICAS PARA RESOLUÇÃO DO PROBLEMA

3.1 A Rede GSM de Telefonia Celular

De acordo com a *GSM Association's, Global System for Mobile Communications*, sigla para GSM, é uma tecnologia de celular digital utilizada na transmissão de serviços de voz e dados móveis. (<http://www.gsmworld.com/technology/gsm/index.htm>)

GSM suporta chamadas de voz e dados com velocidades de transferência de até 9,6 Kbit/s, juntamente com a transmissão de SMS (Short Message Service). Opera na faixa entre 900 MHz e 1,8 GHz na Europa e na faixa entre 1,9 GHz e 850 MHz nos Estados Unidos. A faixa de 850 MHz também é utilizada pela rede GSM e 3G (*Third Generation*) na Austrália, Canadá e em muitos países da América do Sul. (<http://www.gsmworld.com/technology/gsm/index.htm>)

Por ter um espectro harmonizado na maior parte do globo, a capacidade de *roaming* internacional da rede GSM permite aos usuários acessarem os mesmos serviços que usam nos países que residem quando viajam pro exterior.

As redes GSM já abrangem mais de 80% da população do mundo. O *roaming* GSM via satélite também ampliou o acesso aos serviços em áreas onde a cobertura terrestre não está disponível. (<http://www.gsmworld.com/index.htm>)

3.2 Short Message Service – SMS

O Serviço de Mensagens Curtas (*Short Message Service*) permite aos usuários enviarem e receberem mensagens a partir de um aparelho de telefone celular. Cada mensagem pode ser de até 160 caracteres e ser enviada e recebida por usuários de diferentes redes de operadoras. Todos os telefones celulares têm suporte ao SMS.

Bilhões de mensagens de texto são enviadas semanalmente em todo o mundo. Estima-se que um total mundial de um trilhão de mensagens de textos foram enviadas em 2005 (SVZERZUT, 2008).

Além do uso do SMS de pessoa para pessoa, uma variedade dos serviços de mensagens baseadas nos conteúdos dos textos estão disponíveis. A maioria das operadoras GSM permite aos usuários se inscreverem nos serviços que enviam conteúdos sobre notícias, esportes e entretenimento para os telefones móveis na forma de um SMS.

3.3 – Arquitetura de rede GSM e o SMS

Uma rede GSM utiliza duas redes separadas: uma para chamadas de telefone, usando o protocolo ISUP, que é a base do protocolo ISDN, para configuração e liberação de chamadas, e uma para sinalização, principalmente para mensagens relacionadas com mobilidade (caso contrário, não seria uma rede móvel) e SMS.

De acordo com (Henry-Labordère, 2004), cada operadora móvel tem as suas próprias formas de separar as duas redes privadas que ela constrói com linhas e equipamentos. Para se comunicarem uma com a outra, em particular, para serem capazes de oferecerem os serviços de roaming, as prestadoras utilizam as operadoras internacionais (France Telecom, Belgacom, Teleglobe, Swisscom, British Telecom, e assim por diante), que também lhes fornece uma rede de telefonia internacional (ISUP) e uma rede de sinalização para a troca de mensagens de sinalização.

Em uma determinada rede móvel, como na rede mostrada na figura 3.1, o *gateway* entre as redes privadas e as operadoras internacionais é uma peça de equipamento denominado *Gateway Mobile Switching Center* (GMSC), que é tanto um *switch* para chamadas telefônicas e um *router* para sinalização de mensagens SCCP. Os serviços de SMS envolvem apenas a rede de sinalização de dados SCCP, uma rede de pacotes datagrama reservada aos operadores móveis. Conforme (Henry-Labordère, 2004), um SMS utiliza apenas cerca de 1Kb e devido a isso traz cerca de 0,1 milhões de euros de receitas para o operador, que proporciona uma margem elevada em comparação com uma chamada de voz de 30 segundos.

O telefone celular (1) da figura 3.1 envia um SMS com o texto para um número de destino (MSISDN). O SIM (*Subscribe Identity Module*) é um cartão inteligente com processador e memória que carrega todas as informações de assinatura. Se alguém está usando outro aparelho de telefone celular armazena o número do SMSC (*Short Message Service Center*).

O SMS-MO atinge este SMSC através do sistema de sinalização nº 7 (SS7) da rede de sinalização (3); este é o procedimento SMS-MO (*Short Message Service – Mobile Originated*). Então o SMSC o envia para o telefone celular de destino, quer seja no mesmo âmbito de rede ou em outra rede usando o processo SMS-MT (*Short Message Service – Mobile Terminated*).

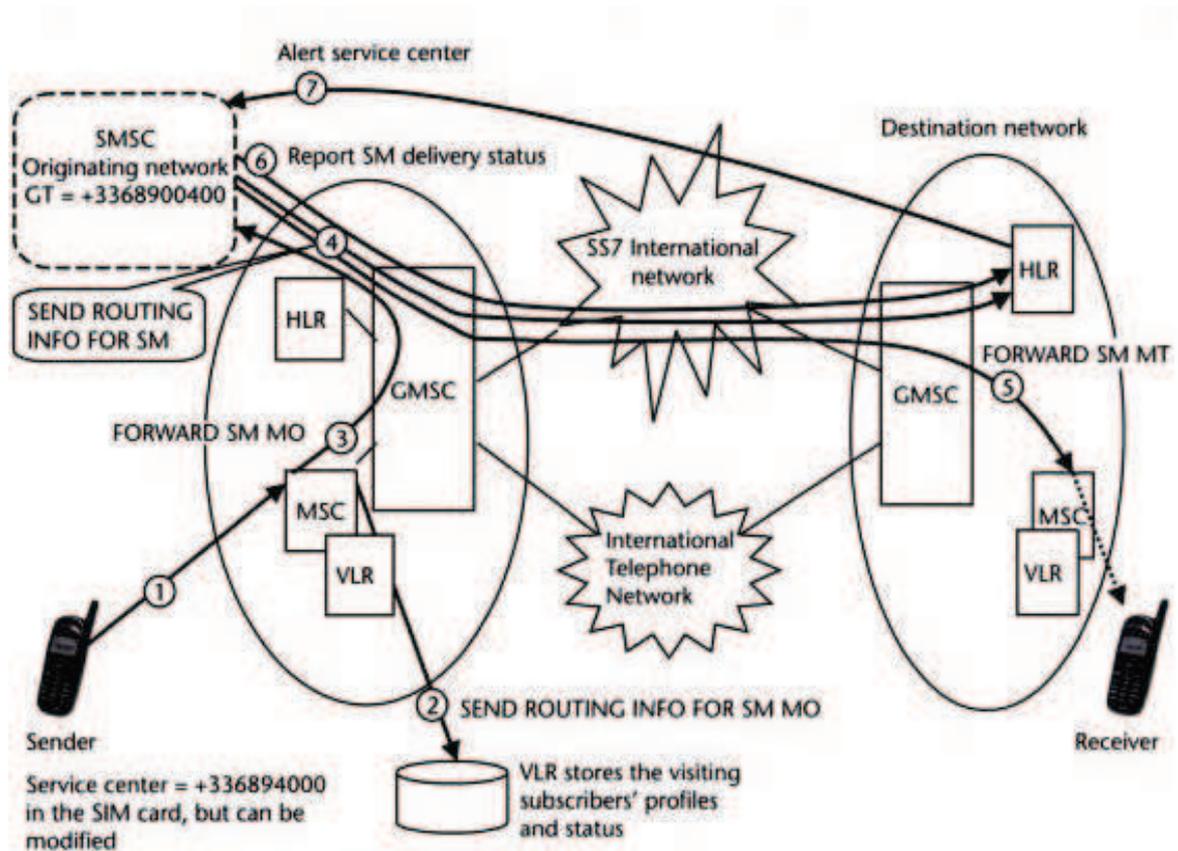


Figura 3.1 – Funcionamento da Arquitetura GSM. Fonte (Henry-Labordère, 2004)

Conforme (Henry-Labordère, 2004), para enviar o SMS, o SMSC interroga o HLR (*Home Location Register*) do telefone celular de destino (4) para obter o endereço do equipamento (o MSC/VLR que pode estar em outro país) onde este telefone celular está atualmente visitando. Então ele pode enviar o sinal do SMS-MT para este MSC (5). Se o sinal falhar, por alguns dos vários motivos (isto é, o celular está fora da área de cobertura, a memória do celular está cheia), o SMSC tem um esquema de repetição, incluindo o desencadeamento (6) por alertas automáticos sempre que as condições necessárias para entregar o SMS possam estar reunidas novamente (7). Este é o propósito dos mecanismos de repetição e alerta.

Neste nível, não há nenhuma diferença fundamental entre os processos do SMS na rede GSM, que utiliza o protocolo MAP (*Mobility Application Part*) e o IS-41, o padrão ANSI (*American National Standards Institute*) usado pelos telefones celulares CDMA (*Code Division Multiple Access*) e o TDMA (*Time Division Multiple Access*).

O mesmo (Henry-Labordère, 2004), na rede GSM, o SMSC pode facilmente endereçar o HLR de seus parceiros de *roaming* porque uma MSISDN internacional é utilizada para perguntar os HLRs. Isto é o porquê de haver uma maior diferença no CDMA ou TDMA quando se trata de enviar um SMS para um assinante em outra rede (CDMA ou TDMA). O SMS é enviado da SMSC de origem o SMSC da rede de destino. Na rede GSM, não há nenhuma diferença entre enviar um SMS para um assinante de mesma rede ou de outra rede.

3.4 – Modem

A palavra modem vem da junção de duas palavras do inglês, *MODulating* (modulador) e *DEModulating* (demodulador).

É um equipamento que tem a função de modular, converter um sinal digital em um sinal analógico, para que seja transmitido através da linha telefônica onde, ao chegar ao destino, este sinal será demodulado e convertido para o sinal digital original.

O processo de conversão dos sinais binários em analógicos é denominado conversão/modulação digital-analógico.

3.4.1 O *Modem* de Dados G24 da Informat Technology

O G24 é o módulo embarcado GSM, que utiliza o microcontrolador HCS08, ambos fabricados pela Motorola. Suporta todas as quatro faixas da rede GSM – 850/900/1800/1900 MHz e pode operar em qualquer rede GSM/GPRS/EGPRS, para fornecer uma comunicação de voz e dados.

O módulo é semelhante ao núcleo condensado de um telefone celular e pode ser integrado juntamente com qualquer sistema ou produto que precise transferir informações de voz ou dados através da rede de telefonia celular.

Assim, aumenta significativamente as capacidades do sistema, transformando-o a partir de um produto isolado e autônomo para um poderoso sistema de alta performance com capacidades de comunicações globais.

Abaixo estão especificadas as principais características do G24 (*Datasheet - Motorola G24 Developer's Guide – Developer's Kit*, 2006):

- Banda GSM: 850/900/1800/1900 MHz;
- Java;
- GPRS/EDGE Multi-Slot classe 10;
- Tamanho: 24.4 x 48.2 x 6.0 mm;
- MO/MT SMS;
- FAX;
- Pilha TCP/IP;
- STK sobre RS232;
- API – *Application Programming Interface*;
- Larga escala de temperatura funcional.

O modem de dados da empresa *Informat Technology*, visualizado na figura 3.2, é um modem GSM que contém o módulo embarcado G24, uma antena, uma fonte AC/DC, entrada para *SIM Card* e comunicação serial RS232 e USB 2.0.



Figura 3.2 – Modem G24 GSM da iTech. Fonte (AUTOR DO PROJETO, 2010)

3.5 – SIM Card – Subscriber Identity Module (Módulo de Identificação do Assinante)

De acordo com Sverzut (2008), o módulo de identificação do usuário (SIM) é um cartão inteligente (*smart card*) conectado internamente ao equipamento móvel. Esse cartão contém informações sobre a estação móvel, tais como:

- Identidade internacional do assinante móvel – IMSI;
- Identidade temporária do assinante móvel (*Temporary Mobile Subscriber Identity - TMSI*);
- Identidade da área de localização (*Location Area Identity - LAI*);
- Chave de autenticação do assinante (*Subscriber Authentication Key - Ki*);
- Número internacional ISDN (*Integrated Service Digital Network*) da estação móvel (*Mobile Station Integrated Services Digital Network - MSISDN*).

Para Sverzut (2008), o processamento e a tarifação das chamadas são realizados a partir das informações contidas no cartão SIM e não no terminal móvel.

O assinante para habilitar uma estação móvel precisa apenas adquirir um cartão SIM da operadora desejada para ser utilizado em qualquer terminal GSM compatível com o sistema da operadora.

O cartão SIM consiste em um microcontrolador, pois possui memória RAM, ROM e EEPROM, além de UCP e ULA, *Timer* e portas E/S. A função básica de um cartão SIM é a autenticação do cliente.

Quando o telefone celular é ligado, o aparelho procura a rede GSM que está registrada no cartão SIM. Ao encontrar a rede, o sistema procura e define a localização do cliente automaticamente.

A identificação e o login são realizados através do chip e não do aparelho de telefone celular, devido ao fato de, o número de telefone e o número do cartão SIM serem únicos no mundo.

3.6 – Comandos AT – *Hayes AT commands*

Os comandos AT são o conjunto de linhas de comandos que são utilizadas para comunicação e configuração de modems. São constituídos por conjuntos de caracteres ASCII no qual são inicializadas pelo prefixo AT. O prefixo AT deriva da palavra *Attention*, no qual é solicitada ao *modem* uma atenção às requisições (comandos) correntes.

Os comandos AT no modem G24 são utilizados para solicitarem alguns serviços, como (*Datasheet - Motorola G24 Developer's Guide – AT Commands Reference Manual*, 2006):

- Serviços de chamada: discagem, resposta e de desligar;
- Utilidades de telefone celular: enviar/receber SMS;
- *Modem profiles*: auto resposta;
- Consultas às redes de telefonia celular: Qualidade do sinal GSM.

Um sistema básico de configuração contém um modem e um terminal. O G24 é o modem e pode ser referido ao DCE (*Data Circuit-terminating Equipment*), o telefone, o celular ou o rádio. O terminal pode ser referido como o DTE (*Data Terminal Equipment*).

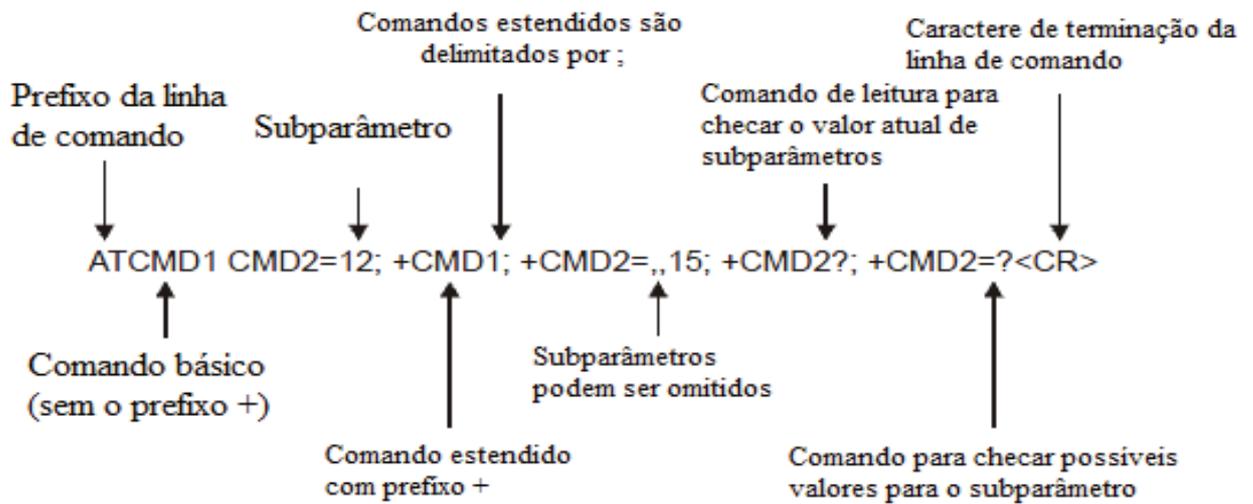


Figura 3.3 – Estrutura básica de uma linha de comando. Fonte (*Datasheet - Motorola G24 Developer's Guide – AT Commands Reference Manual, 2006*)

3.7 – Hyperterminal

O *Hyperterminal* (mostrado na figura 3.4) é um programa que pode ser utilizado para se conectar a outros computadores, *sites* de Telnet, BBSs, serviços *on-line* e computadores *host*, usando um *modem* ou um cabo de *modem* nulo, cabo especial que elimina a necessidade de comunicações assíncronas do *modem* entre dois computadores em distâncias pequenas. Um cabo de *modem* nulo emula a comunicação por *modem*.

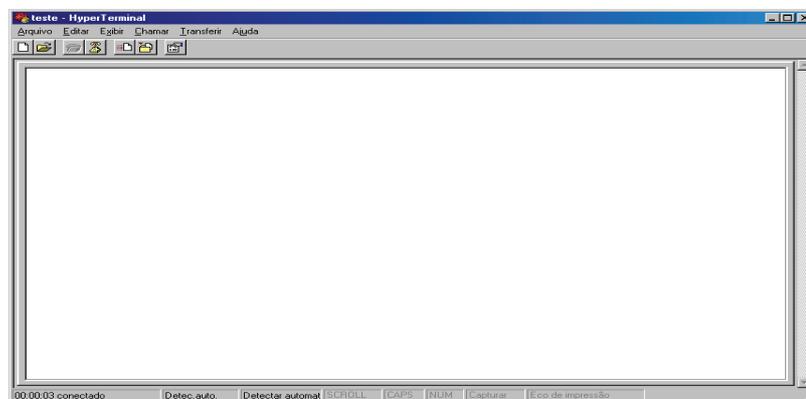


Figura 3.4 – Microsoft *Hyperterminal* Versão 5.1. Fonte (AUTOR DO PROJETO, 2010)

3.8 – Sensores

Os sensores são dispositivos que detectam, percebem ou “sentem” a alteração de uma determinada condição estabelecida, para que o mesmo possa estar informando à central de controle. Diferentes tipos de sensores podem ser encontrados para serem utilizados na construção de projetos de segurança, cada um com a sua lógica de funcionamento e com formas específicas de trabalho.

De acordo com Albuquerque e Thomazini (2005), o funcionamento básico consiste em identificar a energia de um ambiente, podendo ela ser de uma fonte luminosa, térmica, cinética e, assim, relacioná-las com suas grandezas como temperatura, pressão, posição, etc.

3.8.1 Sensor Infravermelho por Reflexão Difusa

O sensor óptico de detecção por reflexão difusa possui o emissor e o receptor montados no mesmo dispositivo. A luz emitida pelo emissor cria uma região ativa cuja presença de um objeto faz com que a luz seja refletida de forma difusa, de volta ao receptor, ativando o sensor.

O emissor de luz óptico pode ser um LED (diodo emissor de luz) ou uma lâmpada. O receptor é um componente fotossensível, isto é, sensível à luz como o fototransistor, o fotodiodo ou os resistores variáveis pela luz (LDR – *light dimmer resistor*).

Para (Rosário, 2005), o princípio de funcionamento de um sensor óptico baseia-se num circuito oscilador que gera uma onda convertida em luz pelo emissor. A luz ao encontrar um objeto, é refletida e, portanto, o receptor acaba por detectá-la. Um circuito eletrônico identifica a variação no componente fotossensível, o qual irá emitir um sinal que poderá ser utilizado para controle e inspeção.

Algumas características desses sensores de proximidade estão descritas abaixo:

- Ausência de contato mecânico para sensoriamento;
- Ausência de partes móveis;
- Pequenas dimensões;
- Chaveamento seguro;

- Insensíveis a vibrações e a choques;
- Maior variedade de configurações;
- Requerem sempre alinhamento;
- Em ambientes com alto grau de luminosidade (ex.: setor de soldagem), há a possibilidade de blindagem;
- Necessário a limpeza e o isolamento do pó e da umidade.

3.9 – O microcontrolador 8051

Em 1976, a Intel lançou seu primeiro microcontrolador: o 8048 de 4 bits. Nascia um conceito muito interessante de CI's (Circuitos Integrados) capazes de controlar um sistema ou máquina de pequeno porte de forma autônoma, pois disponibilizava em seu interior alguns periféricos e tinha como vantagem a possibilidade de ser gravado, não necessitando mais da fabricação de componentes dedicados. Esse componente fez um relativo sucesso e logo ganhou um herdeiro de peso. Assim, em meados da década de 80 a Intel lançou seu substituto o 8051, que logo se tornaria a família de microcontroladores de 8 bits mais famosa e bem sucedida da história.

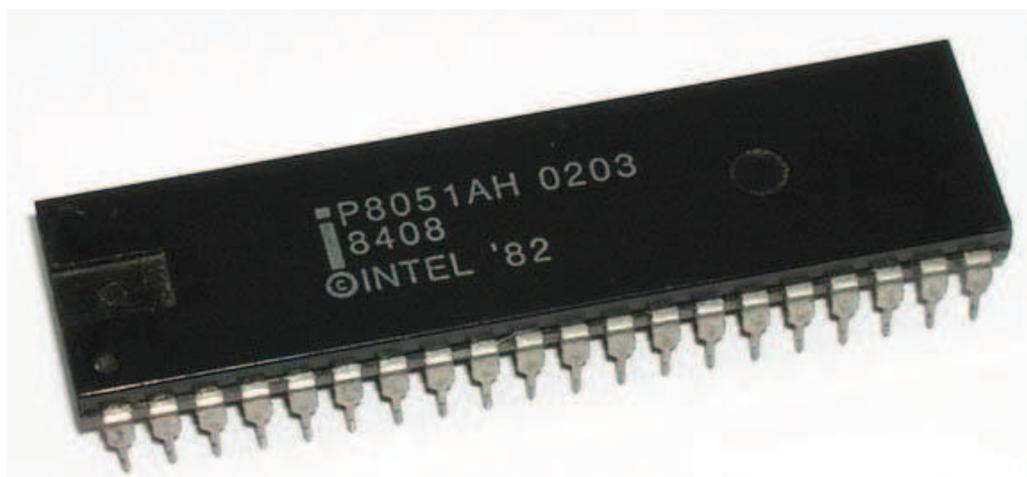


Figura 3.5 – Microcontrolador Intel da família 8051

Fonte (<http://www.cpu-world.com/CPUs/8051/MANUF-Intel.html>, 12/05/2010)

A tecnologia dos microcontroladores não é recente, possuindo cerca de 30 anos e só a família 8051 já existe há cerca de 30 anos. Alguns fatores podem ser enumerados para explicar o porquê de, somente agora, esses componentes serem vistos sendo amplamente utilizados e divulgados.

Dentre os mais importantes estão:

- Preço: atualmente é muito mais barato o projeto e fabricação com componentes deste tipo, pois, além do valor do componente ter caído muito na última década, a grande quantidade, de periféricos internos evita a utilização de mais componentes, barateando o projeto e tornando-o muito mais compacto;
- Memória de programa interna: outro fator decisivo para o crescimento e fortalecimento desta linha de produtos foi à utilização de memória de programa interna, e na maioria das vezes de tecnologia flash, o que facilita muito a programação, regravação e atualizações, atendendo assim, desde o desenvolvedor até o fabricante de larga escala;
- Ferramentas de desenvolvimento: outro tópico importantíssimo a ser mencionado são as ferramentas de desenvolvimento, as quais facilitam muito o manuseio e programação destes componentes. Hoje existem inúmeras IDE's (*Integrated Development Environment* - ambiente integrado para desenvolvimento de software) que permitem programação em várias linguagens de alto nível e, algumas até de forma visual, sem mencionar os simuladores cada vez mais práticos e reais. Isso tornou muito mais agradável o trabalho de desenvolvimento com microcontroladores.

Embora a família 8051 tenha mais de 20 anos de idade, ainda é importantíssimo seu conhecimento pelos seguintes motivos:

- Estrutura simples e de fácil entendimento. Além disso, o bom entendimento da estrutura interna destes microcontroladores auxilia no entendimento e aprendizado de outros;
- O 8051 e outros derivados da família são componentes muito confiáveis para a grande maioria das aplicações;

- O conjunto de instruções da família 8051 é bastante didático e fácil de utilizar. Além disso, é utilizado por diversos dispositivos como sistemas SoC (System-on-Chip) e controladores USB.

A Intel liberou o núcleo desta família para todos os fabricantes que quisessem criar microcontroladores similares e isso originou inúmeros componentes equivalentes, tornando o projeto bastante portátil, além de popularizar muito a utilização desta família. Hoje mais de 50 fabricantes produzem cerca de 700 variações de microcontroladores com esse núcleo e seu consumo anual é próximo de 300 milhões de unidades.

3.9.1 – Características de *Hardware* da Família 8051

O 8051 e seus equivalentes (família MCS-51) são microcontroladores de 8 bits, pois todos os seus barramentos internos, entradas da ULA (Unidade Lógica e Aritmética) e decodificadores de instruções são de 8 bits.

3.9.2 – Pinagem do 8051

As primeiras versões do 8051, de acordo com (Nicolosi, 2004), eram baseadas em encapsulamento do tipo DIP40. Hoje os dispositivos de 40 pinos ainda são muito comuns, porém, de acordo com a aplicação e os periféricos internos presentes no chip, eles podem ser encontrados em diversos encapsulamentos como:

- *DIP – Dual in Pack;*
- *QFP – Quad Flat Pack;*
- *PLCC – Leadless Chip Carrier.*

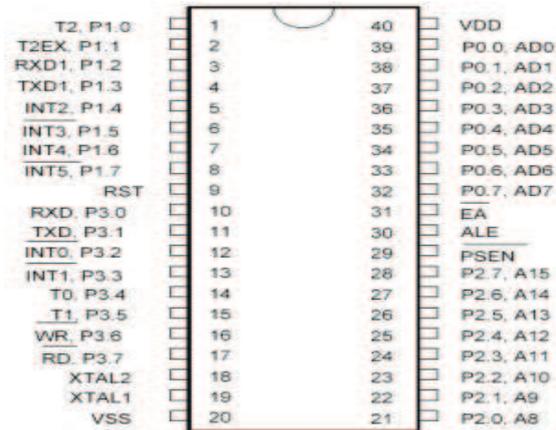


Figura 3.6 – Layout padrão do microcontrolador 8051 – Encapsulamento DIP40

Fonte (http://www.computer-solutions.co.uk/info/micro-search/8051/8051_tutorial.htm)

Abaixo, seguem as descrições das pinagens do microcontrolador 8051 padrão, conforme Nicolosi (2004):

- VDD (pino 40) – Alimentação, geralmente tensão de 5V, mas depende do componente, hoje existe uma tendência muito forte por tensões mais baixas como 3,3V e em certos casos 1,8V);
- VSS (pino 20) – Terra – GND;
- Port P0 (pinos 39 a 32) - Barramento de 8 pinos físicos que formam um 1 *byte*, utilizado como porta de entrada e saída (I/O) de dados. Pode ser utilizado como barramento de uso geral ou, pode ser utilizado para trabalhar com memória externa de dados ou de programa. Quando utilizada esta função, esse barramento tornar-se multiplexado, ou seja, por esse mesmo caminho trafegam o *byte* menos significativo (LSB) dos endereços e o também os dados que serão escritos ou lidos na memória externa;
- Port P2 (pinos 28 a 21) – Barramento de 8 pinos físicos que formam 1 *byte*, utilizado como porta de entrada e saída (I/O) de dados. Pode ser utilizado como barramento de uso geral, ou pode ser utilizado para trabalhar com memória de dados ou de programa externa. Quando utilizado esta função, por ele trafega o *byte* mais significativo (MSB) dos endereços de memória externa que será acessada;
- Port P1 (pinos 1 a 8) – Barramento de 8 pinos físicos que formam 1 *byte*, utilizado como porta de entrada e saída (I/O) de dados. Pode ser utilizada como barramento de uso geral, podendo ler ou escrever dados via software. Em alguns componentes da

família, como no 8052, esse port possui outras funções, como por exemplo, acesso externo ao *Timer/Counter 2*;

- Port P3 (pinos 10 a 17) – Esta porta possui funções especiais em seu barramento, pois é nela que estão ligados o canal de comunicação serial, as interrupções, os timers/counters e os pinos de escrita e leitura da RAM externa. Esse port também pode ser utilizado como entrada e saída de dados (I/O comum). É necessário saber se serão utilizadas as suas funções especiais ou se irá trabalhar o port como I/O comum:
 - P3.0 – RXD, entrada serial;
 - P3.1 – TXD, saída serial;
 - P3.2 – INT0, interrupção externa 0;
 - P3.3 – INT1, interrupção externa 1;
 - P3.4 – T0, entrada para o timer 0 (contador, neste caso);
 - P3.5 – T1, entrada para o timer 1 (contador, neste caso);
 - P3.6 – WR, escrita na memória de dados externa;
 - P3.7 – RD, leitura na memória de dados externa.
- RST (pino 9) – Assim como o computador, nos microcontroladores, existe um pino físico de RESET, que reinicia suas funções de maneira pré-estabelecida por *hardware*;
- ALE/PROG (pino 30) – *Address Latch Enable*. Este pino habilita o *latch*, de modo a guardar a parte menos significativa do endereço, quando a memória externa (dados ou programa) é acessada. É utilizado automaticamente pelo microcontrolador quando se utiliza as instruções de acesso à memória externa (dados ou programa);
- PSEN (pino 29) – *Program Store Enable*. É controlado automaticamente pelo microcontrolador quando se utilize a memória de programa externo. Nele são gerados os pulsos que habilitam a ROM externa;
- EA/VPP (pino 31) – *External Access Enable*. Informa a CPU se o programa tem início na memória de programa interna ou externa. Se a memória de programa interna for utilizada, será fixado esse pino físico em “1” (VCC). Se o for trabalhado somente com memória de programa externa, esse pino deve ser fixado em “0” (GND). De acordo com o microcontrolador usado, o tamanho de memória de programa interna varia. Utilizando a memória de programa interna do microcontrolador, logo que esta acabe o microcontrolador busca automaticamente o restante do programa na memória externa, não havendo necessidade de alterar o sinal do pino EA.

- XTAL1 e XTAL2 (pinos 18 e 19) – Pinos através dos quais será conectado o cristal que irá gerar o *clock* externo para o componente. O 8051 possui um oscilador interno ligado a esses pinos e todo o sincronismo de execução de tarefas será baseado na frequência de pulsos deste oscilador. Podem-se utilizar cristais como gerador de *clock* externo, respeitando a frequência máxima e mínima de trabalho de cada componente da família 8051, ou usar outro gerador de *clock* externo do tipo TTL e, neste caso, o pino 19 deve ser aterrado e o sinal injetado no pino 18.

Na figura 3.7, pode-se constatar o funcionamento da estrutura interna em diagrama de blocos do microcontrolador 8051.

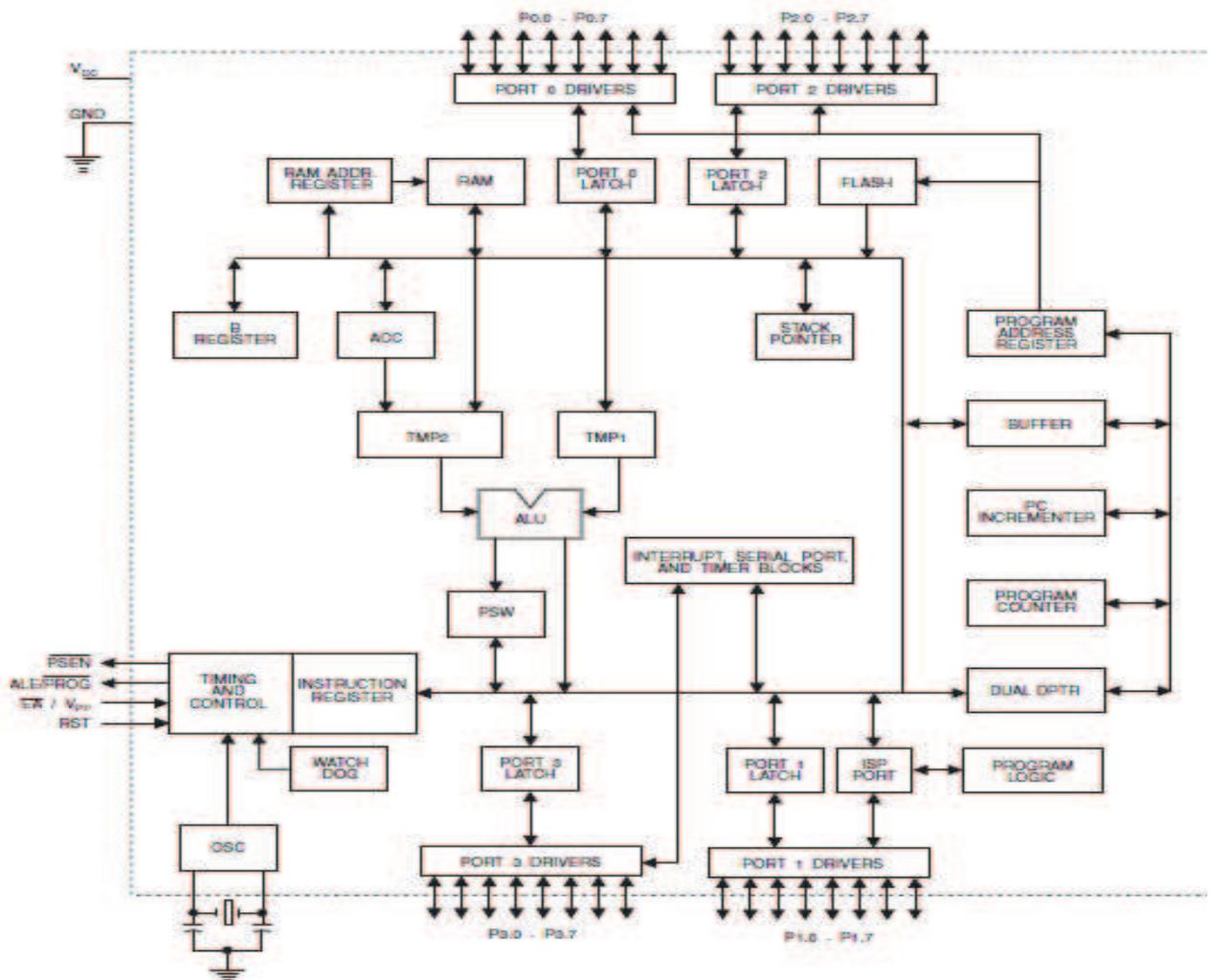


Figura 3.7 – Diagrama de blocos do microcontrolador 8051 Fonte (Atmel – Datasheet do microcontrolador AT89S52 da família 8051)

3.9.3 – Características Básicas de um Microcontrolador da Família MC8051, conforme (Zelenovsky e Mendonça, 2005):

- CPU de 8-bits otimizada para aplicações de controle;
- Processamento booleano amplo (lógica *single-bit*);
- Capacidade de endereçamento de memória de programa de 64KBytes;
- Capacidade de endereçamento de memória de dados de 64KBytes;
- 4KBytes de memória de programa *on-chip*;
- 256 bytes de RAM de dados (128 *bytes* uso geral / 128 *bytes* SFR's);
- 32 linha de I/O programáveis;
- Dois contadores/temporizadores de 16 bits;
- Porta de comunicação serial (UART *Full duplex*);
- Estrutura de interrupções com dois níveis de prioridade;
- Oscilador de relógio on-chip.

3.9.4 – O Microcontrolador AT89S52 da Atmel – Principais Características:

- Compatibilidade de 100% com a família 8051;
- 8 *KBytes* de memória flash (memória de programa);
- 256 *Bytes* de memória RAM (memória de dados);
- 32 portas de entrada/saída;
- Modo de programação serial ISP (*In-System Programmable*).

3.10 – A linguagem de Programação C para Sistemas Embarcados

De acordo com (Nicolosi e Bronzeri, 2009), a linguagem de programação C foi desenvolvida pelos pesquisadores Dennis M. Ritchie e Ken Thompson nos laboratórios Bell, empresa fundada pelo inventor do telefone Alexander Graham Bell e que hoje faz parte da empresa de telecomunicações AT&T (*American Telephone and Telegraph*). A criação da linguagem data do início da década de 70 e, posteriormente, veio sua padronização pelo ANSI (*American National Standards Institute*).

A linguagem C, embora seja de alto nível, é tão poderosa que permite acesso direto ao hardware dos processadores, por isso ela é considerada por muitos como uma linguagem de nível intermediário entre o homem e a máquina.

Programar em C requer alguns cuidados, pois a estrutura da linguagem é extremamente rígida e o programador deve estar muito certo do que está fazendo, a fim de evitar erros graves de lógica, pois não são apresentados pelo compilador. Além disso, esta linguagem de programação possui poucos comandos em sua biblioteca padrão e os comandos mais sofisticados, na verdade são funções previamente escritas e pertencem a outras bibliotecas, também padronizadas, e que quando utilizados a biblioteca deve ser chamada em seu programa.

3.10.1 A estrutura de um programa em C

Um programa em C é constituído por funções, dessa forma, todo programa, mesmo os mais simples, utilizarão pelo menos uma função, que necessariamente é chamada de *main()*. Dentro dela estará o programa. Para dizer ao compilador que o programa está dentro da função *main*, os códigos e instruções pertencentes ao programa devem estar entre {}, os quais definem todos os objetos pertencentes à função que os antecede.

Nos microcontroladores, que são máquinas de pequeno porte, não há um sistema operacional, portanto a função principal *main* de um programa, não entregará valores a ninguém. Dessa forma, nos programas, a função *main* aparecerá da seguinte maneira: *void main(void)*.

```

/*****
Exemplo de uma estrutura em linguagem C
*****/

#include "C:\Keil\C51\INC\Atmel\REGX52.H"

void main (void)
{
    long int contador1,contador2;

    P2 = 0xFE;

    while(1)

```

```
{
    for (contador1 = 0; contador1 <= 10; contador1++)
        for (contador2 = 0; contador2 <= 1000; contador2++);

    P2_0 = ~P2_0;
}
}
```

3.11 – Software Keil

O *software* Keil é uma ferramenta utilizada para edição, compilação e simulação de programas, dentre elas a linguagem C embarcada para microcontroladores 8051.

A Keil é uma empresa de origem alemã, porém hoje faz parte do grupo tecnológico inglês ARM, empresa esta que desenvolveu o núcleo de microcontroladores de 32 bits, de mesmo nome, o qual está se tornando muito utilizado.

O Keil C51 é uma ferramenta completa para desenvolvimento de aplicações para microcontroladores da família 8051, pois permite em um único software a criação de um projeto em linguagem C, o qual pode conter diversos arquivos e bibliotecas associadas. Permite a compilação, ou seja, a tradução de um programa escrito em *Assembly* ou C para o código de máquina do 8051 e, permite também, a simulação do programa, já que ele possui um *debugger* acoplado, no qual é possível o acompanhamento passo a passo de cada instrução e também a alteração dos valores contidos em variáveis e registradores.

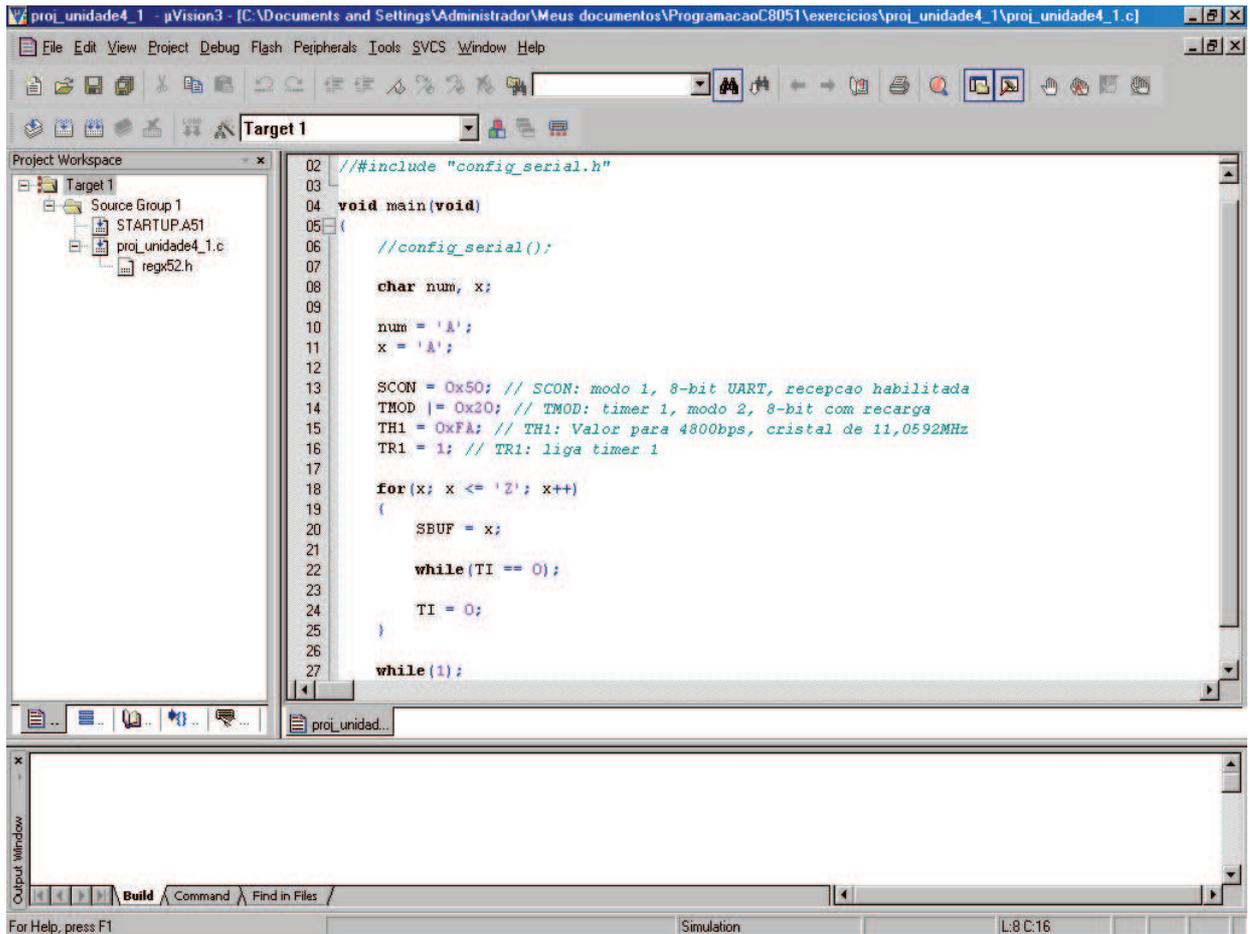


Figura 3.8 Ambiente de Desenvolvimento no Keil µVision3 V3.80. Fonte (AUTOR DO PROJETO, 2010)

Quando são criados os programas em linguagem C para 8051 é utilizada toda a estrutura padronizada do C (ANSI C) para a criação da lógica do programa.

Existem outros compiladores C para 8051 de bom desempenho, como o Ride51 e o SDCC. Como a linguagem C é padronizada, a migração para outros compiladores é fácil, bastando apenas algumas pequenas alterações no código utilizado.

3.12 O Proteus

O Proteus VSM (*Virtual System Modelling*) é uma suíte composta pelo ambiente de simulação de circuitos eletrônicos ISIS, sendo um modo misto de simulação de circuitos, componentes animados e modelos de microprocessador para facilitar a co-simulação em completos projetos baseados nos microcontroladores e um programa para desenho de circuito impresso (*PCB – Printed Circuit Board*) ARES Professional.

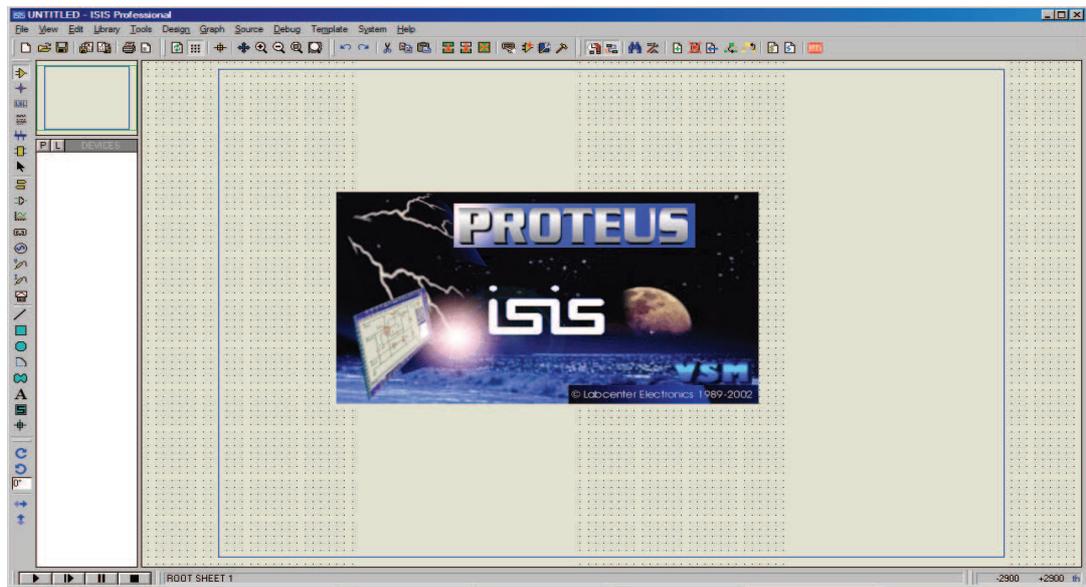


Figura 3.9 – Tela principal do software Proteus 6.2 ISIS Professional da *Labcenter Electronics*. Fonte (AUTOR DO PROJETO, 2010)

Isso é possível devido a interação com o desenho usado na tela com indicadores como o LED e o LCD e os atuadores, tais como interruptores e botões. A simulação ocorre em tempo real (ou perto o suficiente). Um Pentium III de 1 GHz pode simular um sistema básico de *clock* do 8051 em mais de 12 MHz. Proteus VSM também dispõe de instalações de depuração extensivas, incluindo pontos de parada, passo a passo, tanto para os códigos em *Assembly* como os códigos de alto nível.

3.13 Gravador SPI-Flash Programmer V3.7

O *software* de gravação SPI - *Flash Programmer* V3.7 possibilita a gravação de microcontroladores de forma serial no próprio circuito (processo ISP – *In Circuit Serial Programmer*), o qual é utilizado para gravar o programa compilado, realizada a partir do arquivo hexadecimal (*.hex), gerado após a compilação do programa. O uso do *software* é livre e sem restrições.

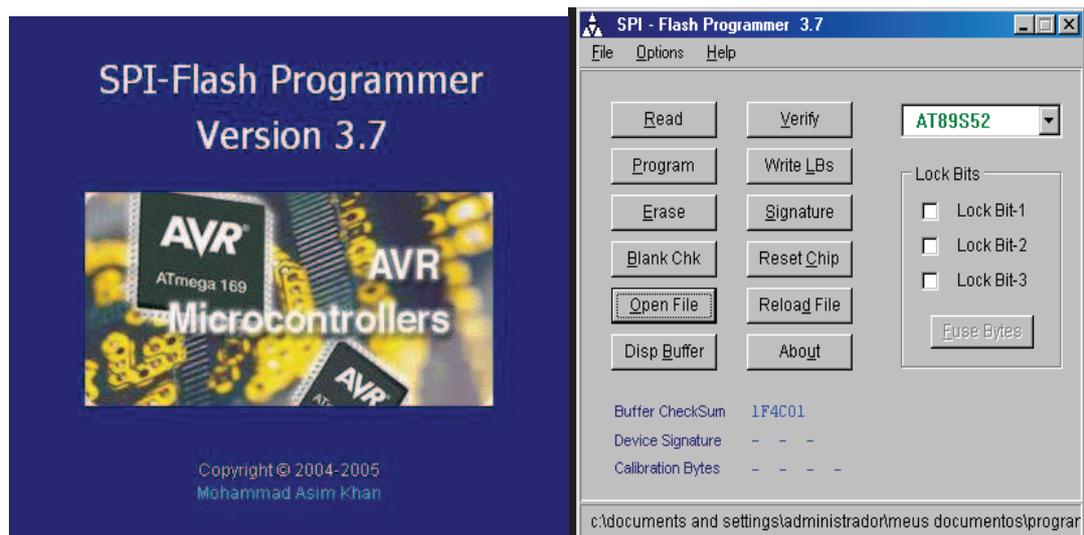


Figura 3.10 – *Software* para gravação do arquivo no microcontrolador da família 8051 – SPI *Flash Programmer* Versão 3.7. Fonte (AUTOR DO PROJETO, 2010)

3.14 - Comunicação Serial RS-232

Conforme (Soares, 2009), este padrão prevê a comunicação entre “Equipamentos de Dados – DTE - *Data Terminal Equipment*” (PC, por exemplo) e “Equipamentos de Comunicação de Dados – DCE - *Data Communications Equipment*” (modem ou outro periférico). A norma definiu as características elétricas (níveis de sinal e seus respectivos valores) e mecânicas (tipo de conectores e sua respectiva pinagem).

Uma parte da norma define as características do sinal elétrico. Na RS-232 existem apenas dois níveis lógicos possíveis: “0” lógico e “1” lógico. Não há outro nível ou estado intermediário.

A norma RS-232C prevê níveis de tensão DC entre +3V e +15V para o nível lógico “0” e -3 V e -15 V para o nível lógico “1”. Essas tensões devem ser consideradas com “carga”, ou seja, somente com DTE e DCE devidamente conectados. Os níveis de tensão entre -3 V e 3 V são considerados indefinidos e devem ser evitados.

Outro ponto importante também definido pela norma é sua forma de transmissão. Ela é do tipo serial. Ou seja, os dados que na sua maioria trafegam pelos barramentos dos processadores na forma “paralela” (seqüência de *bytes* de acordo com o tamanho do barramento) devem ser “transformados” em uma seqüência serial de bits, pois o novo caminho não é mais um “barramento”, mas uma “via única” (geralmente um fio).

Sendo assim, os dados serão transferidos entre o equipamento DTE e DCE *bit a bit*. Ou seja, um *byte* é levado até a porta RS-232 transmissora (DTE), que o transformará em uma seqüência de *bits* e os enviará ao periférico receptor (DCE). Por sua vez, a porta RS-232 receptora receberá o conjunto de *bits* e os “montará” novamente na forma de *byte* .

Pode-se, então, “definir” a porta RS-232 como uma interface que transforma os dados em dados seriais durante a transmissão e vice-versa durante a recepção.

Na figura 3.10 é mostrado o cabo serial, padrão RS-232, com os conectores do tipo DB9, de 9 pinos.

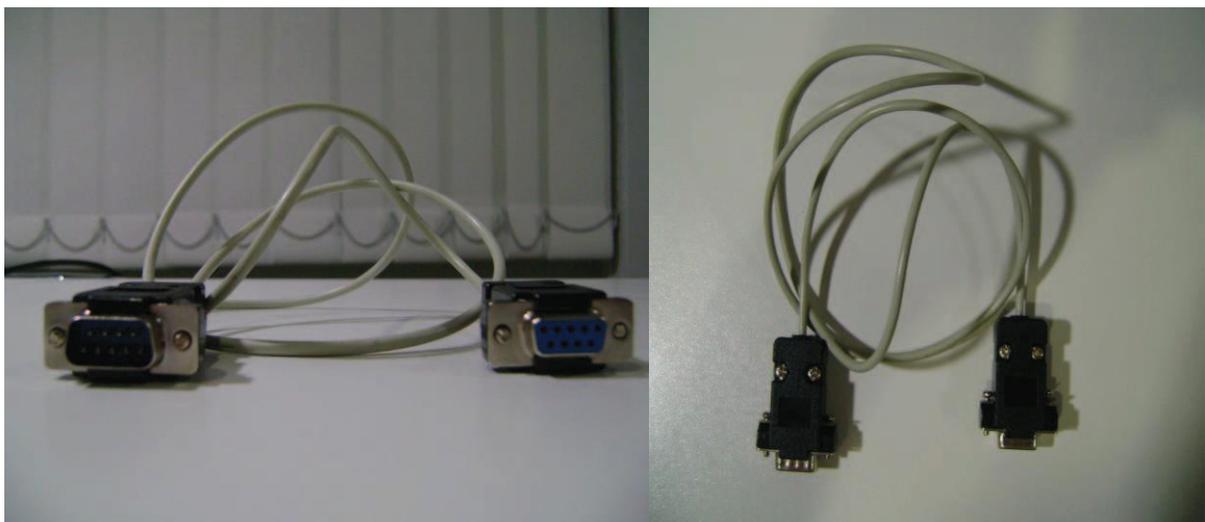


Figura 3.11 - Cabo serial RS-232 db9
Fonte (AUTOR DO PROJETO, 2010)

CAPÍTULO 4 - MODELO PROPOSTO

4.1 – Apresentação Geral do Modelo Proposto

O sistema de alarme de intrusão utilizando a tecnologia de SMS desenvolvido apresenta uma integração entre um *modem* G24, o circuito de proximidade, o microcontrolador e o aparelho de telefone celular.

Inicialmente, projetou-se e simulou-se o esquema elétrico no Proteus, construindo o circuito de proximidade para que o mesmo efetue a correta comunicação com o Kit8051LS, que contém o microcontrolador AT89S52.

Terminado o desenvolvimento e a simulação no Proteus, realizou-se a programação no ambiente de desenvolvimento Keil, onde foi criado e compilado o programa, gerando assim um arquivo pronto para gravação (através do *software SPI Flash Programmer*) no microcontrolador AT89S52.

No Hyperterminal do Windows XP, através do cabo serial padrão RS-232, acontece a comunicação com o modem G24, bem como sua configuração utilizando os comandos AT para que, posteriormente, a integração com o microcontrolador possa ser efetuada.

Finalizadas essas etapas, deu-se início a integração dos componentes anteriormente descritos, já configurados, verificando assim se a comunicação entre os mesmos acontece como esperado.

O circuito de proximidade infravermelho com o microcontrolador, se comunica com o G24 através do cabo serial que, por sua vez, utiliza a rede GSM de telefonia celular para realizar a comunicação com o aparelho celular.

4.2 – Descrição das Etapas do Modelo

4.2.1 – Desenvolvimento do Circuito de Proximidade Infravermelho

O circuito montado em *protoboard* apresentado na figura 4.1, tem como função detectar a presença de uma intrusão (alguém ou algo) por meio de um sensor infravermelho, que utiliza o princípio da reflexão difusa da luz e alerta com sinalização de *bips* com sons intermitentes, produzidos por um alto-falante de 8Ω , e o envio de um sinal para o microcontrolador.

A distância de detecção através do sensor no circuito montado e apresentado na figura 4.1 abaixo, será de aproximadamente dez centímetros.

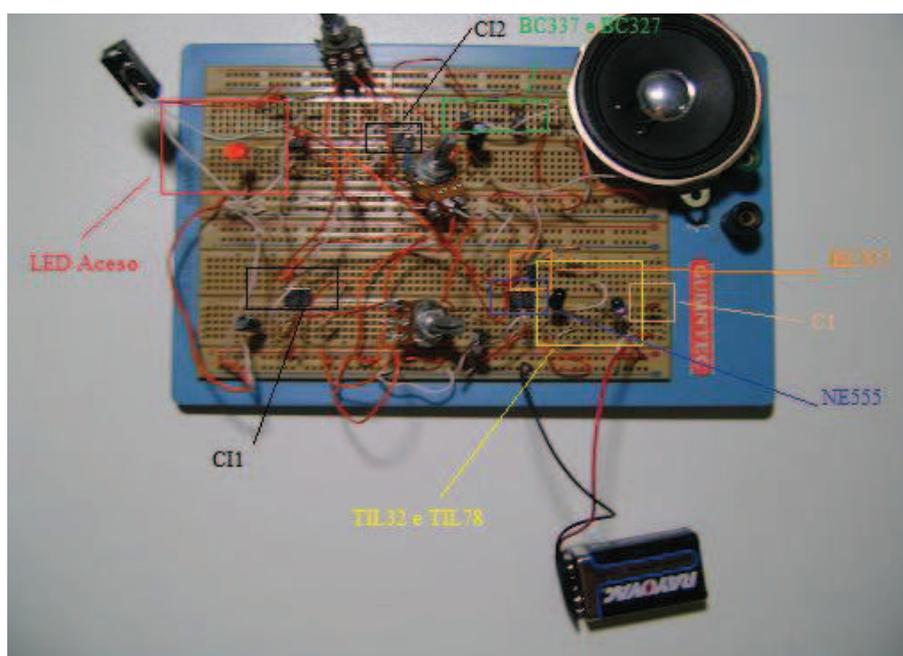


Figura 4.1 - Circuito de proximidade desenvolvido em *Protoboard* com o LED aceso, significando a detecção através do sensor.

Fonte (AUTOR DO PROJETO, 2010)

Lista de componentes utilizados para o desenvolvimento do circuito no Proteus:

- Semicondutores
 - NE555 – Timer;

- 4N25 – Acoplador Óptico;
- LED Infravermelho 5mm;
- LED Vermelho;
- 1N4148 – diodo de sinal;
- TIL 78 – fototransistor;
- BC337 – transistor NPN de uso geral;
- BC327 – transistor PNP de uso geral.
- Resistores (todos com dissipação de 1/8 W, a não ser o de 150)
 - 150Ω;470kΩ;10kΩ;18kΩ;100Ω;100KΩ;390Ω;1KΩ;2K2Ω.
- Potenciômetros
 - 100kΩ;50kΩ;10kΩ.
- Capacitores
 - 100nF – cerâmico;
 - 22μF x 35V – eletrolítico;
 - 47nF – poliéster;
 - 220μF x 16V – eletrolítico.
- Diversos
 - Bateria de 9V e conector tipo clip;
 - Alto-falante de 8Ω;
 - Conector Header de 10 pinos;
 - Cabo flat de 10 vias;
 - Fios de cobre.

Principais características de ajustes do alarme de proximidades são:

- Ajuste de temporização;
- Ajuste de intermitência;
- Ajuste de tom;
- Principais características técnicas:
- Distância de detecção de 10 centímetros;
- Entrada de ajustes: potenciômetro linear;
- Alimentação: Bateria de 9Vdc ou fonte externa de 9Vdc.

A emissão e recepção de luz infravermelha é representada por um par emissor-receptor, sendo um foto-receptor (TIL78) e um foto-transistor (TIL32).

A utilização para função do *bip* é realizada pelo circuito integrado 555 (NE555).

O CI NE555 é utilizado com duas configurações distintas, ou seja, como monoestável e como astável. A função do monoestável é a de temporizar o *bip*, que será de 1s até 4s. A função do astável é a de gerar intermitência do *bip* e a de gerar o tom (mais agudo ou menos agudo) do *bip*.

Para realizar a comunicação com o Kit8051LS, que contém o microcontrolador AT89S52, foi necessário criar um circuito de isolamento com acoplador óptico, que tem uma entrada com o sinal vindo do circuito que contém o sensor e uma saída que se conecta com um conector do tipo header de 10 vias, onde através de um cabo flat de 10 vias realiza a conexão com a porta de expansão do *Port P3* do microcontrolador, mais precisamente, ao pino P3.2, gerando uma interrupção no AT89S52.

A seguir, na figura 4.2, se pode observar a conexão entre o Kit8051LS e o circuito desenvolvido no Protoboard.

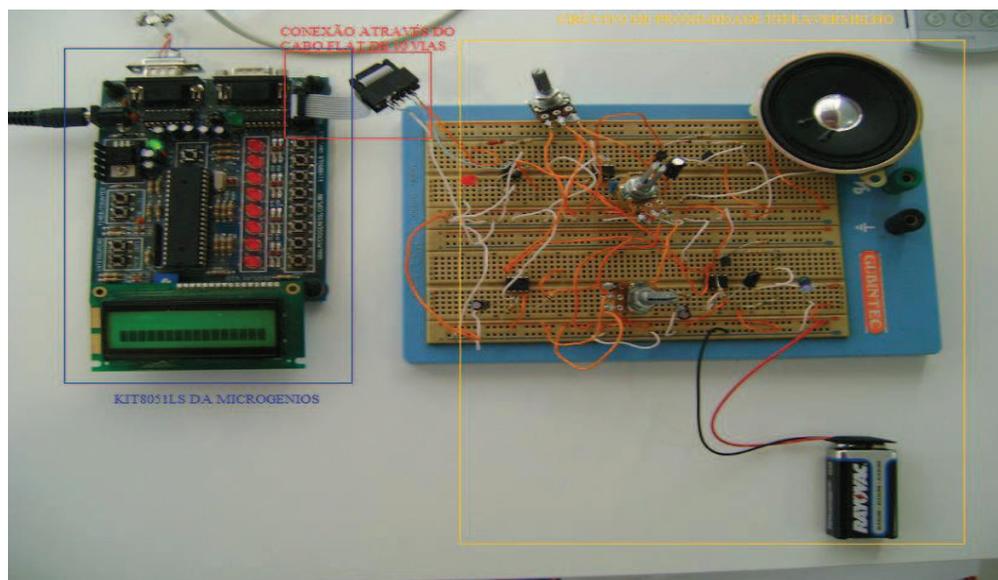


Figura 4.2 – Conexão utilizando um cabo flat de 10 vias

4.2.2 - Funcionamento do Circuito de Proximidade

O TIL32 emite ininterruptamente luz infravermelha. Devido ao fato dessa luz estar sendo emitida em linha reta, ela logo se dispersará quando uma pessoa ou objeto atravessar essa linha. O TIL78, então, mantém o transistor BC337 polarizado no corte. Logo, o monoestável NE555 está com a sua saída em nível lógico “0” e nada acontece.

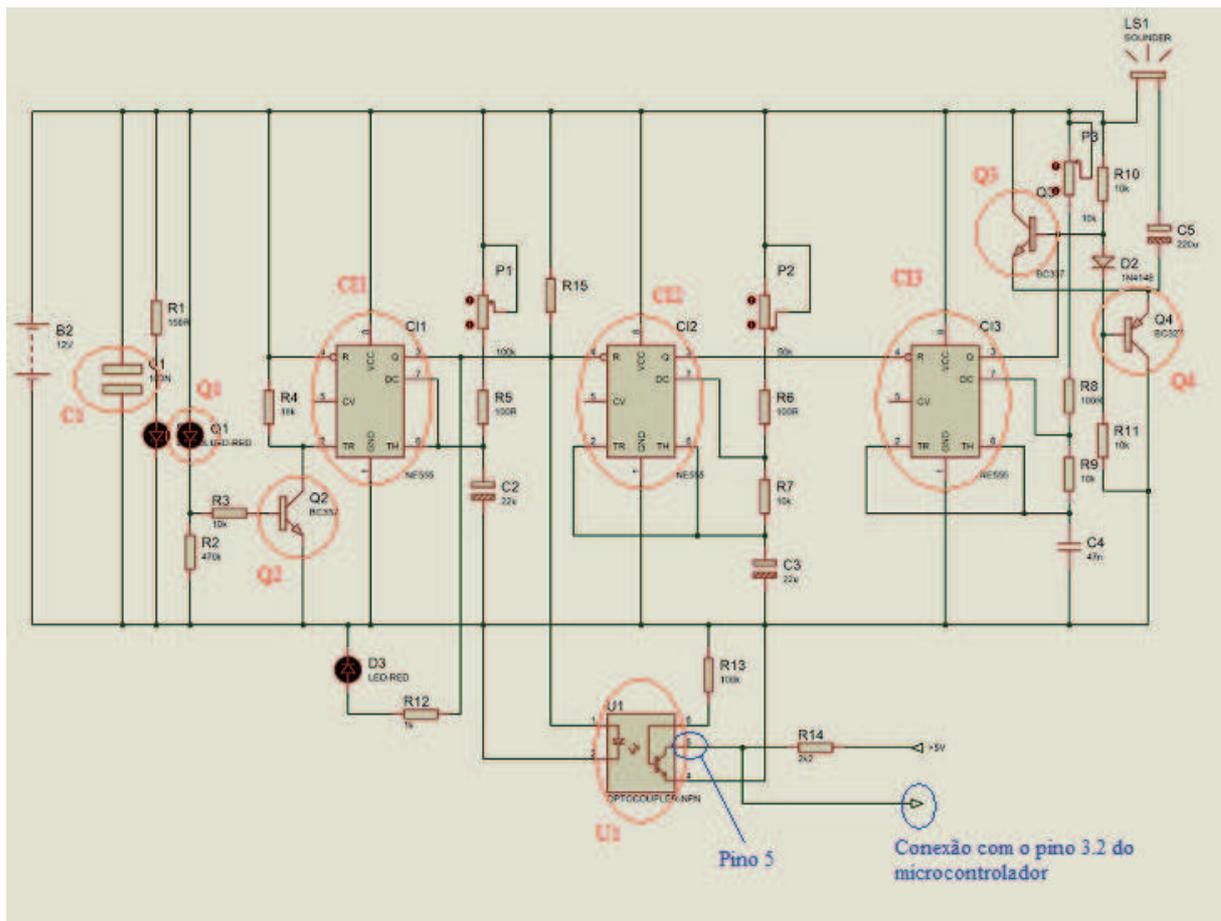


Figura 4.3 – Esquema elétrico

Fonte (Desenvolvido pelo autor do projeto no software Proteus 6.2 da *Labcenter Electronics*, 2010)

Assim que um objeto (ou uma pessoa) corte o feixe infravermelho, uma parte desse feixe será rebatida de volta e irá acionar o fototransistor TIL78 (Q1, na figura 4.3), o qual entrará em corte, onde, então, começará a conduzir corrente, o que fará com que o transistor BC337 (Q2, na figura 4.3) tenda à saturação. Com isso, teremos o disparo do timer NE555 (CI1, na figura 4.3), o qual colocará seu pino de saída em nível lógico “1”. Esse nível irá acionar o funcionamento dos osciladores de baixa frequência NE555 (CI2, na figura 4.3) e alta frequência NE555 (CI3, na figura 4.3), gerando assim os *bips*. Para que o volume de som

seja suficientemente amplificado, o sinal de saída do CI3 é conectado a um amplificador de áudio formado por BC337 e BC327 (Q3 e Q4, respectivamente na figura 4.3).

O capacitor C1, como mostrado na figura 4.3, tem a função de eliminar o nível DC, ou seja, as tensões de polarização dos transistores não passam por ele. Somente a onda quadrada alcança o alto-falante.

O acoplador óptico, mostrado na figura 4.3 pelo componente U1, tem a função de isolar as tensões do circuito de proximidade infravermelho, que é alimentado por uma bateria de 9V, do Kit8051LS que contém o microcontrolador AT89S52. É através do pino 5 do componente U1 (mostrado na figura 4.3) que é enviado o sinal de interrupção ao pino 3.2 do microcontrolador AT89S52 do Kit8051LS, quando o circuito de proximidade infravermelho é ativado.

4.2.2 – Programação e Compilação no Keil

No ambiente de programação Keil é realizado a programação em linguagem C para compilação e geração do arquivo em formato .hex. Antes de utilizar o ambiente de programação, é necessário configurar o Keil para o microcontrolador que será utilizado, no caso o AT89S52 do fabricante Atmel, criando um projeto que é descrito nas etapas que se seguem abaixo:

Ao executar o *software* Keil, dentro do item *Project* é selecionada a opção *New Project*, mostrada na figura 4.4:

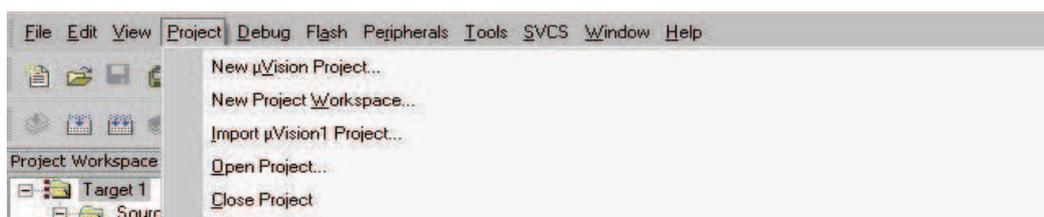


Figura 4.4 – Criação de um novo projeto. Fonte (AUTOR DO PROJETO, 2010)

Na figura 4.5 abaixo, uma nova pasta é criada, onde será salvo o projeto.

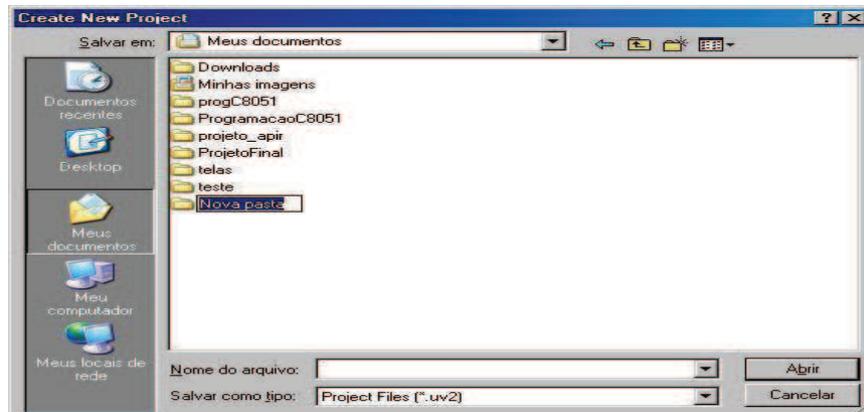


Figura 4.5 – Nomear e salvar pasta e projeto. FONTE (AUTOR DO PROJETO, 2010)

Nesta etapa (figura 4.6) é selecionado o microcontrolador AT89S52 da ATMEL.

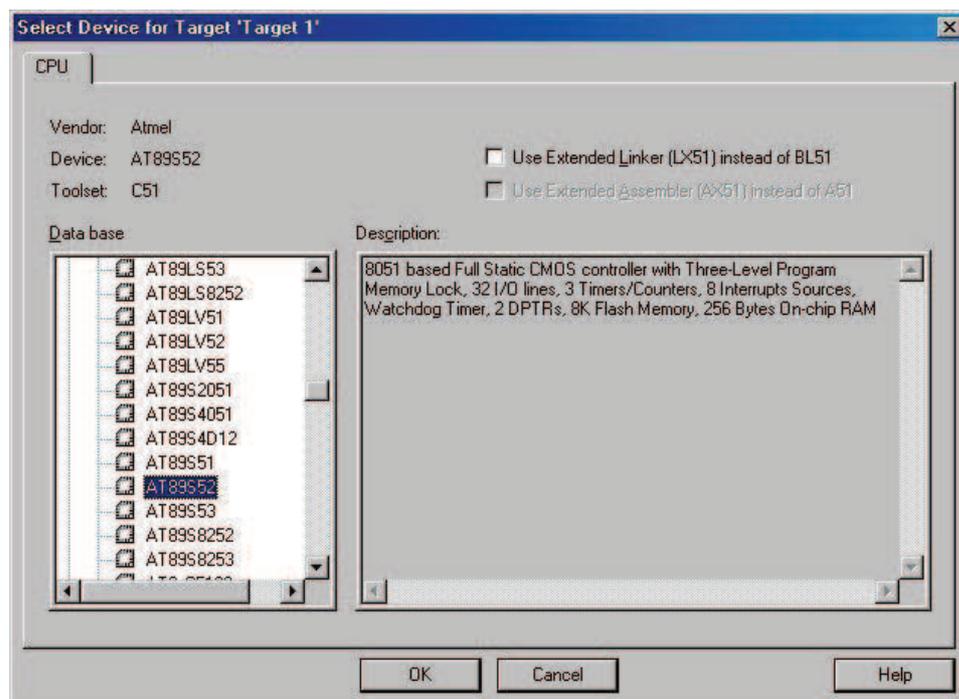


Figura 4.6 – Fabricante e modelo do microcontrolador a ser utilizado. Fonte (AUTOR DO PROJETO, 2010)

Para finalizar a criação do projeto, a opção SIM (figura 4.7) foi selecionada, visto que neste arquivo mencionado encontra-se o procedimento correto para a inicialização do microcontrolador após o *RESET*.

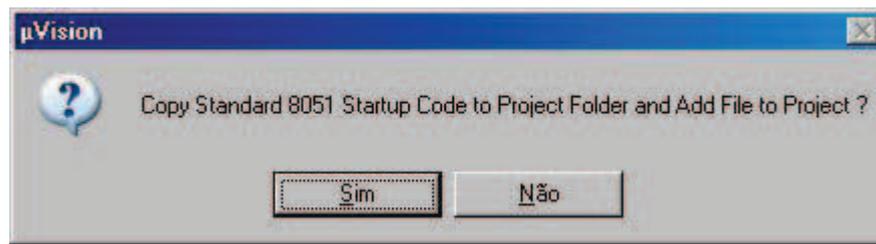


Figura 4.7 – Copiando para o projeto o arquivo do Startup. Fonte (AUTOR DO PROJETO, 2010)

Após a criação do projeto, cria-se um arquivo de texto onde será escrito o código fonte do programa acessando o no item *File*, a opção *New*. Deve-se salvar (figura 4.8) o arquivo que foi aberto, com a extensão *.c*, dentro da pasta do projeto criado inicialmente.

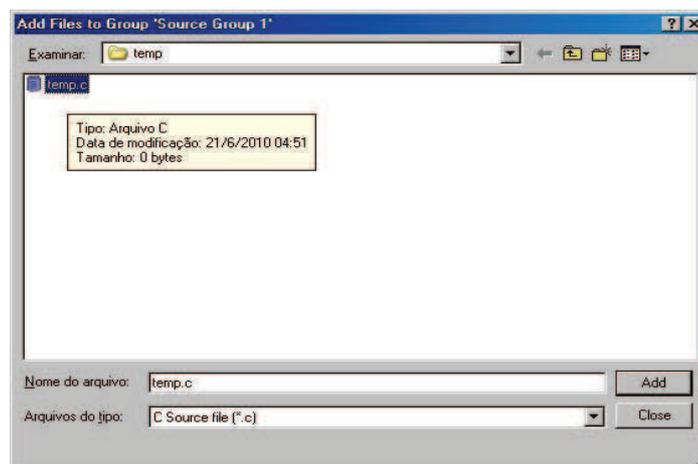


Figura 4.8 – Criando um novo arquivo. Fonte (AUTOR DO PROJETO, 2010)

Clicando com o botão direito do mouse sobre a pasta *Source Group 1*, é selecionado o item "*Add Files to Group 'Source Group 1'*" (figura 4.9), onde será realizado o anexo do arquivo ao projeto.

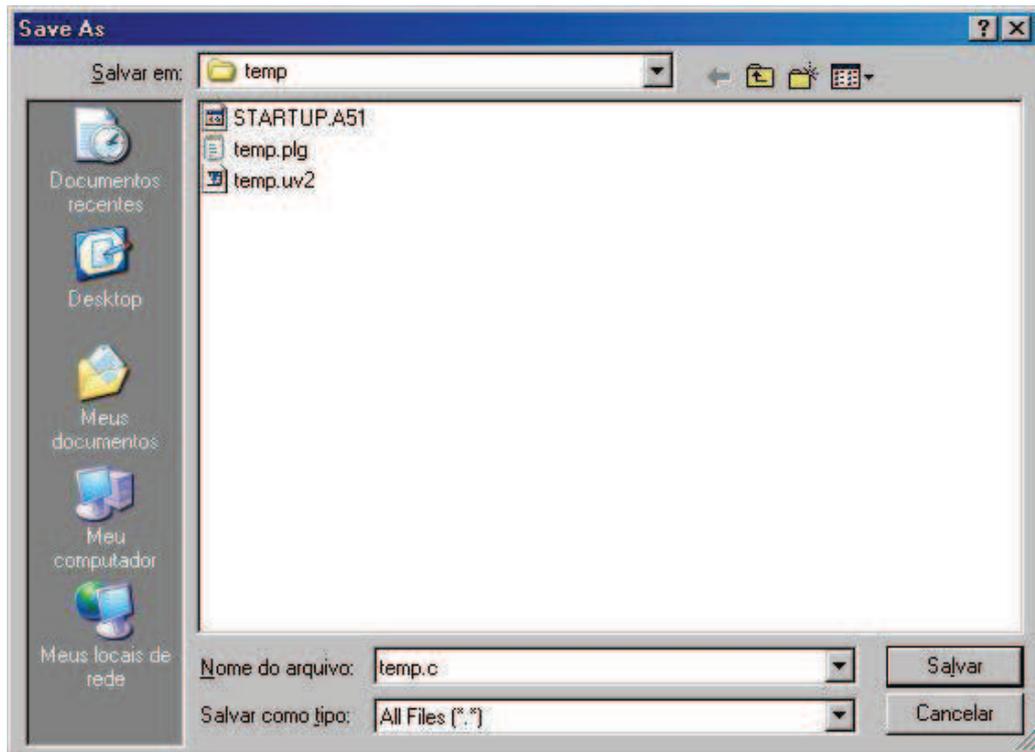


Figura 4.9 – Anexo do arquivo ao projeto. Fonte (AUTOR DO PROJETO, 2010)

Para geração do arquivo em hexadecimal (.hex) é necessário realizar o seguinte procedimento: com o botão direito do mouse sobre a pasta *Target 1*, foi selecionada a opção *Options for Target 'Target 1'*. Na janela que será mostrada (figura 4.10), acessa-se a aba *Output* e marca-se a opção *Create HEX File*.

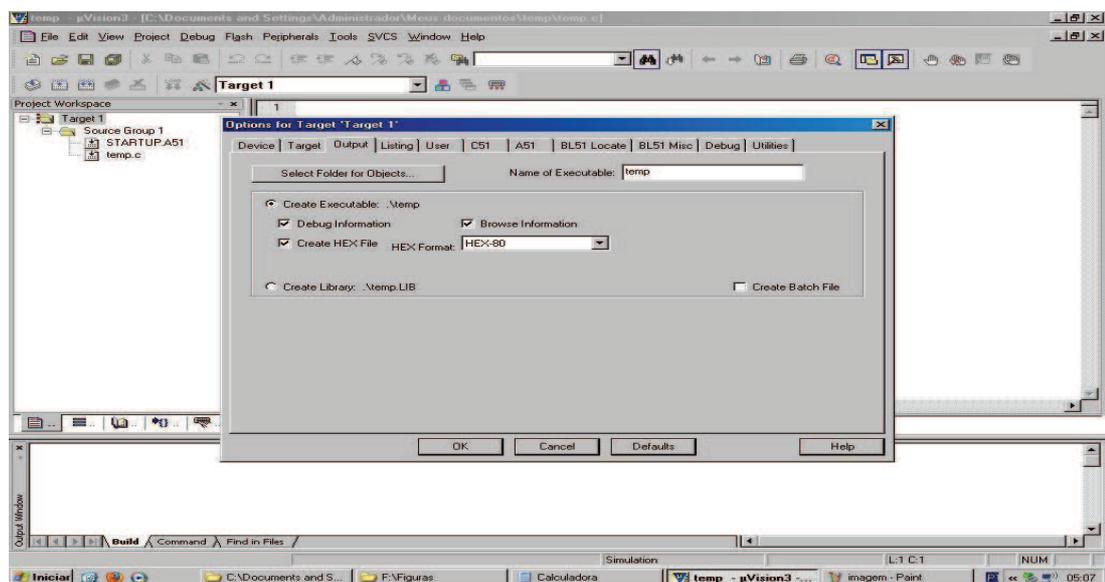


Figura 4.10 – Habilitar a geração do arquivo em hexadecimal. Fonte (AUTOR DO PROJETO, 2010)

A escrita do código, desenvolvido pelo (AUTOR DO PROJETO, 2010) em linguagem C, para configuração do modem e o envio de mensagens, após ser detectado o sinal de interrupção do microcontrolador, é mostrado abaixo:

```
#include "C:\Keil\C51\INC\Atmel\REGX52.H" // Biblioteca que contém os mnemônicos

// Protótipos das funções
void int_serial(void);
void int_detect(void);
void config_modem(void);

void main(void) // função principal do programa
{

    // Configuração dos registradores de funções especiais do microcontrolador AT89S52

    IE = 0x91;          // Habilita as seguintes chaves: EA, ES, EX0
    IP = 0x00;         // Configura as prioridades, todas em baixa prioridade
    TCON = 0x40;       // INT0 - nivel, liga Timer 1
    SCON = 0x50;       // SCON: no modo 1, 8-bit UART, recepção habilitada
    TMOD |= 0x20;      // TMOD: timer 1, modo 2, 8-bit com recarga
    TH1 = 0xFD;        // TH1: Valor para 9600bps cristal de 11,0592MHz

    while(1)          // Mantém os led's apagados
    {
        P2 = 0xFF;    // Bit 1, para que cada pino do Port P2 fique apagado.
    }

}

void int_serial(void) interrupt 4 // Função de interrupção serial chamada quando o
//circuito de proximidade infravermelho é ativado.
```

```

{
    char b = 0, cont;    // Criação das variáveis b e cont do tipo char e inicialização da
                        // variável b com o valor zero.

    b = SBUF;    // O dado recebido através da porta serial é armazenado no
                // registrador SBUF e, então, colocado na variável b.

    while(RI == 0);    // espera que o flag de recepção seja 1, ou seja, fim da
                        // recepção.

    RI = 0;    // zera o flag, permitindo nova recepção.

    if(b)    // Verifica se o dado recebido é diferente de zero, ou seja, se igual a
            // zero, FALSO, então não entra no bloco de condição if. Se 1, então
            // entra no bloco de condição if, VERDADEIRO.
    {
        for(cont = 0; cont <= 10; cont++) // enquanto a variável cont for menor ou
                                        // igual a 10, o bloco abaixo continua sendo
                                        // executado.
        {
            P2=~P2;    // inverte os estados de P2.
        }
    }

    b = 0;    // zera a variável b.
}

void config_modem(void) // Função de configuração do modem.
{
    char a, i;    // Criação das variáveis a e i do tipo char.

```

```

char modem[]={ "AT&K0;E0;+CMGF=1;+CNMI=,2;+CMGL=\"REC
UNREAD\";+CMGD=1,1"};      // String com a mensagem a ser armazenada

for(i = 0; modem[i] != '\0'; i++)    // Executa o bloco e finaliza quando for
                                        //verificado que a variável da string for igual a '\0'
{
    SBUF = modem[i];    // Coloca o valor da variável no registrador SBUF

    while(TI == 0);    // espera que o flag de transmissão seja 1, ou seja, fim da
                        // transmissão.

    TI = 0;            // Zera o flag, permitindo nova transmissão.
}

a = 0x0D;    // O valor que será armazenado na variável a, significa o carriage return,
              //equivalente ao mesmo procedimento quando é pressionada a tecla
              //enter do teclado do computador.

SBUF = a;    // Coloca no registrador SBUF o valor contido na variável a

while(TI == 0);    // Espera que o flag de transmissão seja 1, ou seja, fim da

                    // transmissão.

    TI = 0;    // Zera o flag, permitindo nova transmissão.
}

void int_ detect(void) interrupt 0    /* Função da interrupção externa INT0 – Pino 3.2 do
4051. Sinal vindo do circuito de proximidade
infravermelho, quando ativado o fototransistor (TIL 32)
pelo led emissor infravermelho (TIL 78). */
{
    char cont, a, b, i, j;    // Criação das variáveis do tipo char.

```

/* Criação das variáveis do tipo char com as string's a serem armazenadas, que são os comandos seguidos do número gerado pelo modem após a configuração e a armazenagem da mensagem no modem. */

```
char comando1[]={"AT+CMSS=180"}, comando2[]={"AT+CMSS=181"};
```

```
for(i = 0; comando1[i] != '\0'; i++) // Executa o bloco for até que o valor da variável
// contida no índice i seja igual a '\0'.
```

```
{
```

```
    SBUF = comando1[i]; // Copia para o registrador SBUF, o valor da
//variável contida no índice i da string.
```

```
    while(TI == 0); // Espera que o flag de transmissão seja 1, ou seja, fim da
// transmissão.
```

```
    TI = 0; // Zera o flag permitindo nova transmissão.
```

```
}
```

```
a = 0x0D; // O valor que será armazenado na variável a, significa o carriage return,
```

```
//equivalente ao mesmo procedimento quando é pressionada a tecla
//enter do teclado do computador.
```

```
SBUF = a; // Coloca no registrador SBUF o valor contido na variável a
```

```
while(TI == 0); // Espera que o flag de transmissão seja 1, ou seja, fim da
// transmissão.
```

```
TI = 0; // Zera o flag permitindo nova transmissão.
```

```
for(j = 0; comando2[j] != '\0'; j++) // Executa o bloco for até que o valor da variável
// contida no índice i seja igual a '\0'.
```

```
{
```

```
SBUF = comando2[j];    // Copia para o registrador SBUF, o valor da
                        //variável contida no índice i da string.

while(TI == 0);       // Espera que o flag de transmissão seja 1, ou seja, fim da
                        // transmissão.

TI = 0;               // Zera o flag permitindo nova transmissão.
}

a = 0x0D;             // O valor que será armazenado na variável a, significa o carriage return,

                        //equivalente ao mesmo procedimento quando é pressionada a tecla
                        //enter do teclado do computador.

SBUF = a;             // Coloca no registrador SBUF o valor contido na variável a.

while(TI == 0);       // Espera que o flag de transmissão seja 1, ou seja, fim da
                        // transmissão.

TI = 0;               // Zera o flag permitindo nova transmissão.

config_modem();       // Chama a função para executar a configuração do modem.
}
```

A Compilação do código escrito, gerando o arquivo em hexadecimal que será gravado no microcontrolador, é visualizado na figura 4.11.

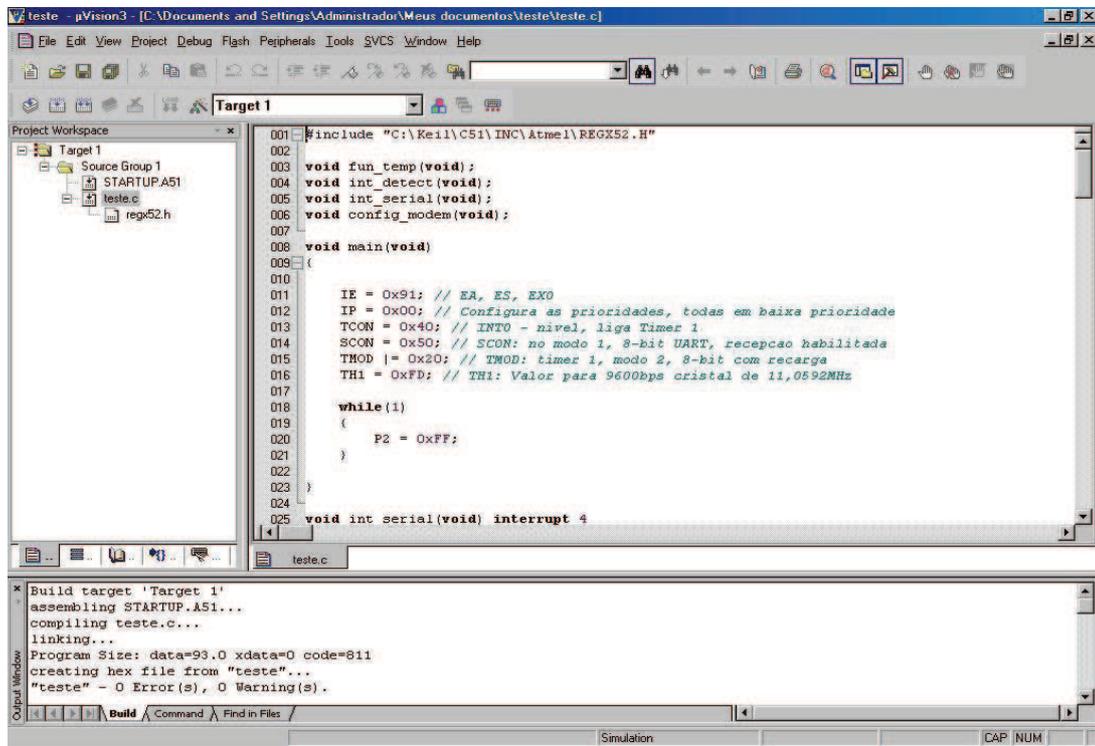


Figura 4.11 – Compilação do arquivo escrito em linguagem C. Fonte (AUTOR DO PROJETO, 2010)

4.2.3 – Gravação no SPI

Após a compilação do arquivo no Keil é executado o software SPI – *Flash Programmer*, onde será selecionado o microcontrolador AT89S52 e aberto o arquivo que foi gerado em hexadecimal, o qual se encontra na pasta do projeto que foi trabalhado. É, então, gravado o arquivo no 8051.

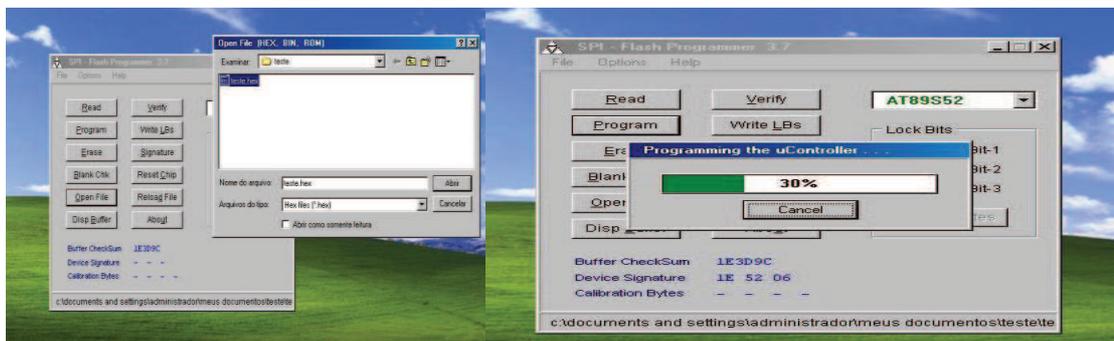


Figura 4.12 – Gravação do arquivo no microcontrolador AT89S52 que se encontra no Kit8051LS.

Após a gravação, o botão do Kit8051LS referente à interrupção externa (no caso, o INT0) é pressionado, resultando no acendimento dos led's, relacionado ao número 20H em hexadecimal, tendo como binário a sequência 00100000 (o bit zero corresponde ao led aceso).

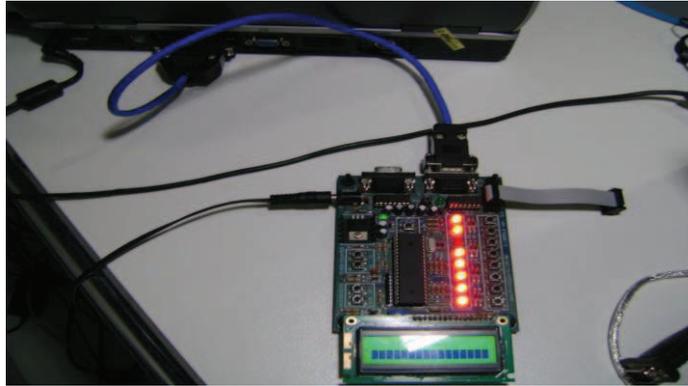


Figura 4.13 – Acendimento dos led's gerado pela interrupção externa. Fonte (AUTOR DO PROJETO, 2010)

4.2.4 Configuração do Modem G24 no Hyperterminal

Para que fosse possível realizar a configuração foi necessário incluir um chip GSM no aparelho, mostrado na figura 4.14.



Figura 4.14 – SIM Card e o Modem G24. Fonte (AUTOR DO PROJETO, 2010)

Para a configuração do modem e dos comandos utilizados para o envio de SMS, foi utilizado o programa *Hyperterminal* no Windows XP.

Antes do programa ser executado, o modem foi ligado ao computador, através do cabo USB. Na figura 4.15 verifica-se que o modem e o notebook estão conectados.

A tela do *Hyperterminal* mostra que o modem está se comunicando com o computador, onde foi digitado o comando AT e pressionada a tecla *ENTER*.



Figura 4.15 – Realizada a conexão, são realizadas as configurações e testes para a comunicação com o *Hyperterminal*. Fonte (AUTOR DO PROJETO, 2010)

Os seguintes comandos foram necessários para a configuração de envio e recebimento de SMS:

1º - A configuração de um terminal para controlar o modem, deve ser de 1 bit de *start*, 1 bit de *stop*, 8 bits de dados e sem paridade. A porta de comunicação usada é a RS232, a taxa é de 9600 bits/s e não há controle de fluxo. Esta é a configuração principal a ser feita no *Hyper Terminal*.

2º - O modem inicia com controle de fluxo ativado, ou seja, usando os pinos CTS e RTS, que não são conectados ao microcontrolador, por esse motivo o primeiro comando enviado ao modem deve ser o de desativar o controle de fluxo. Isto é feito através do comando AT&K0.

3º - É necessário configurar ainda o eco do modem, que por padrão inicia ativado, ou seja, o modem ecoa (repete) os caracteres recebidos por ele. Isto é feito através do comando ATE0 e precisa ser refeito sempre que o modem for ligado.

4º - Envia-se o comando AT+CMGF=1(Modo Texto)

5º - Envia-se o comando AT+CNMI=,2;(indicação de nova SMS)

6º - Envia-se o comando AT+CMGL="REC UNREAD"(Le toda sms recebida)

7º - Envia-se o comando AT+CMGD=1,1(apaga a sms depois de lida)

Resumidamente pode-se mandar o comando em uma linha somente:

Assim: **AT&K0;E0;+CMGF=1;+CNMI=,2;+CMGL="REC UNREAD";+CMGD=1,1**

Dessa forma, o modem está preparado para enviar e receber SMS. Só que após cada rotina ou sub-rotina essa configuração tem que ser mandada novamente para o modem, que periodicamente reseta as configurações

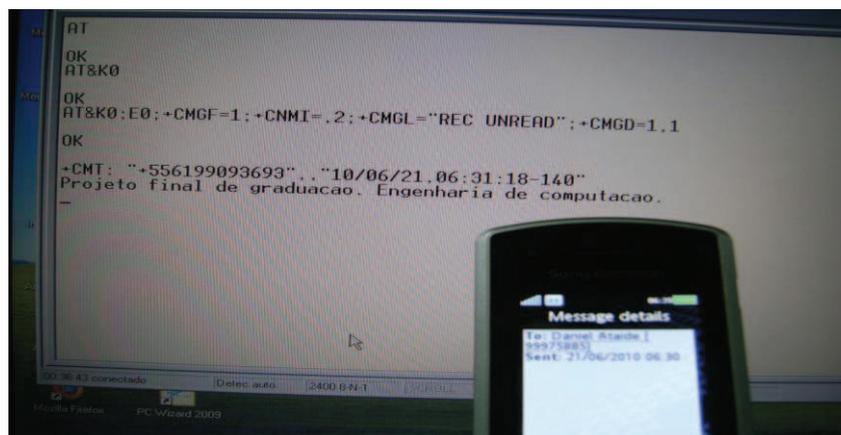
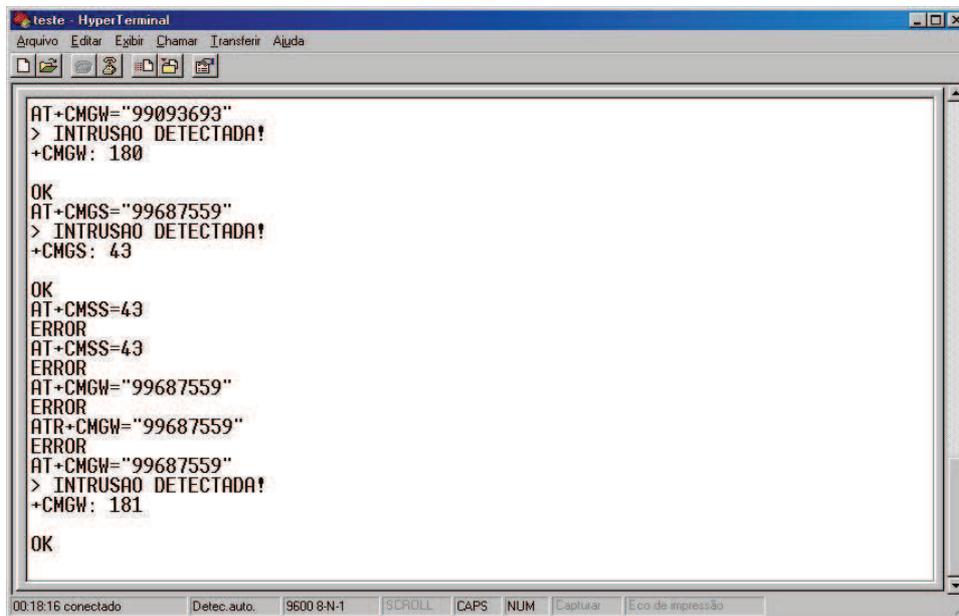


Figura 4.16 – Configuração dos parâmetros necessários para a comunicação com o modem no envio e recebimento de mensagens e os comandos executados no *hyperterminal*, com as respectivas respostas de confirmação pelo modem. Fonte (AUTOR DO PROJETO, 2010)

Configurado o modem, deverá ser configurado o mesmo para armazenar as mensagens que serão enviadas, quando o sistema for acionado, bem como cadastrar os números dos destinatários.

Na figura 4.17 abaixo, pode ser visualizado o comando AT+CMGW, com o respectivo número cadastrado. Ao pressionar a tecla *ENTER*, uma nova linha é acrescentada, onde será digitado o corpo da mensagem que será enviado para transmissão. Pressionando as teclas Ctrl+Z, retorna ao modo AT. O modem, então, retorna como resposta o número que deverá ser adicionado ao comando de envio de SMS.



```
teste - HyperTerminal
Arquivo  Editar  Exibir  Chamar  Transferir  Ajuda

AT+CMGW="99093693"
> INTRUSAO DETECTADA!
+CMGW: 180

OK
AT+CMGS="99687559"
> INTRUSAO DETECTADA!
+CMGS: 43

OK
AT+CMSS=43
ERROR
AT+CMSS=43
ERROR
AT+CMGW="99687559"
ERROR
ATR+CMGW="99687559"
ERROR
AT+CMGW="99687559"
> INTRUSAO DETECTADA!
+CMGW: 181

OK

00:18:16 conectado   Detec. auto.   9600 8-N-1   SCROLL   CAPS   NUM   Capturar   Eco de impressão
```

Figura 4.17 – Armazenando a seguinte mensagem: “INTRUSAO DETECTADA!”. Fonte (AUTOR DO PROJETO, 2010)

O exemplo de envio de SMS pode ser visualizado na figura 4.18, no comando AT+CMSS = 180.

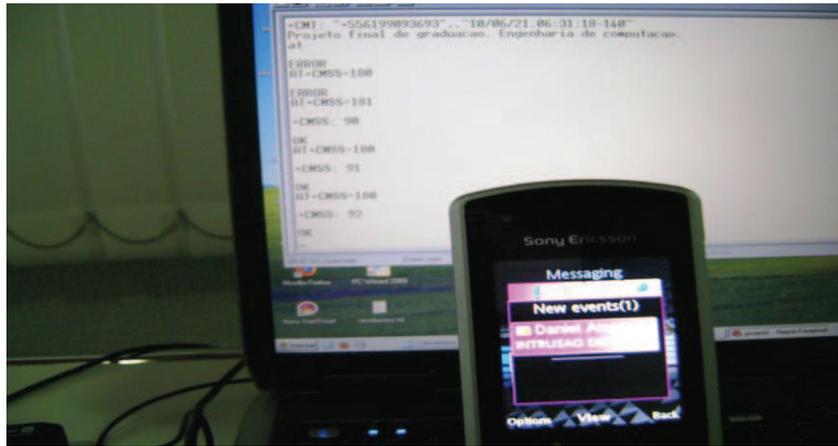


Figura 4.18 – Envio de SMS através do Hyperterminal. Fonte (AUTOR DO PROJETO, 2010)

4.3 – O Sistema de Alarme de Intrusão

Realizadas as configurações necessárias, mostradas anteriormente em cada etapa, tem-se como resultado a integração entre os dispositivos desenvolvidos (visualizada na figura 4.19), bem como a sua comunicação.

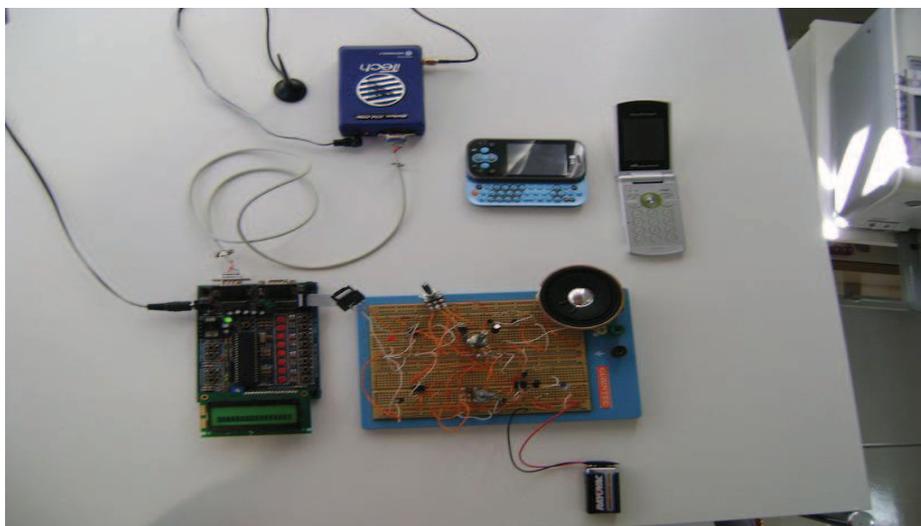


Figura 4.19 - Integração entre os dispositivos: Circuito com sensor, Kit8051LS que contém o microcontrolador AT89S52, o *modem* e dois aparelhos de telefone celular. Fonte (AUTOR DO PROJETO, 2010)

O sensor detecta a presença, quando algo ou alguém é posicionado próximo ao sensor, que envia ao microcontrolador, que por sua vez encaminha ao *modem*, onde é enviada uma SMS ao celular que foi previamente cadastrado, informando com a seguinte mensagem:

“INTRUSAO DETECTADA!”.



Figura 4.20 – Recebimento do SMS que foi enviado pelo modem G24, após a detecção do sensor do circuito de proximidade. Fonte (AUTOR DO PROJETO, 2010)

CAPÍTULO 5 – APLICAÇÃO DO MODELO PROPOSTO

5.1 – Apresentação da Área de Aplicação do Projeto

O sistema de detecção de intrusão utilizando o SMS deve ser implementado em ambientes fechados, que necessitem de proteção de bens (máquinas, utensílios e equipamentos), que ali se encontram. As áreas e/ou setores e/ou departamentos que podem ser protegidos:

- Um departamento de uma empresa,
- Um *DATA CENTER*;
- Uma residência;
- Uma sala-cofre de um banco ou de uma organização;
- Um setor de uma empresa que necessita de sigilo;
- Um setor industrial, que guarda bens de valores expressivos;
- Etc.

5.2 – Descrição da Aplicação do Projeto

Antes de iniciar a implementação do projeto é necessário que seja cadastrado o número do destinatário que será avisado quando o evento de intrusão ocorrer.

Para a implementação do sistema desenvolvido tem-se a necessidade de conectá-lo a uma fonte de energia e, caso a mesma necessite proteger um ambiente que requer extrema proteção, é necessário que esteja preparado com recursos que possibilitem mantê-lo conectado após uma falta de energia, como *no-break*, bateria, geradores ou outras fontes de alimentação.

O sistema uma vez conectado a uma fonte de alimentação entra em funcionamento e é ativado.

O sistema de alarme será capaz de perceber a proximidade de algo ou alguém e, imediatamente, enviar uma SMS ao celular que foi previamente cadastrado.

Para a inclusão de outro número a ser cadastrado para o envio da mensagem quando o alarme é ativado, será necessário conectar o modem ao hyperterminal e realizar esta configuração, anotando o número gerado onde, posteriormente, deve-se abrir o código do programa que foi gravado no microcontrolador AT89S52 e alterar o número no comando de envio da mensagem.

5.3 – Avaliação Global do Projeto

O modelo proposto de **Sistema de Alarme de Intrusão** e a utilização de um SMS, presente em qualquer aparelho de telefonia móvel, apresenta como diferencial competitivo a simplicidade em sua produção.

O principal problema apresentado foi a comunicação do modem com o Kit8051LS, que contém o microcontrolador AT89S52. Várias tentativas foram realizadas para descobrir o problema. O primeiro erro estava relacionado com o cabo serial empregado, visto que não era possível detectá-lo através do Hyperterminal, pois os dois conectores do Kit e do modem eram do tipo fêmea e a conexão com o computador sempre acontecia na forma serial. Foi, então, confeccionado um cabo que pudesse realizar a correta comunicação entre os equipamentos, pois ao usar um adaptador que emendasse dois cabos do tipo serial, constatou-se que um dos cabos não estava com o fio conectado no pino correto. O segundo erro foi detectado na programação, pois não estava sendo possível enviar a SMS, assim que o sistema detectava a intrusão. Depois de corrigido o problema, o modem foi capaz de receber o comando pelo microcontrolador e enviar a mensagem sobre a intrusão detectada.

A principal vantagem do projeto desenvolvido é a utilização de um serviço comumente utilizado atualmente, que é o serviço de mensagens curtas, que se utiliza de um aparelho largamente difundido em todo o mundo e que realiza a comunicação através de uma rede com cobertura em quase todo o mundo.

A desvantagem de implementação do sistema desenvolvido está relacionado ao custo do envio da SMS pelo modem. No caso do SIM Card utilizado no projeto, o custo avulso para o envio da SMS é de R\$0,39 para o celular de mesma operadora em âmbito nacional e de R\$0,80 para o envio fora do país de origem. Fonte

http://www.vivo.com.br/portal/servicos_vivo_torpedo_sms.php?WT.ac=euvivo.servicos-mensagem.icone.vivotorpedosms).

Como pontos fortes destacam-se a forma de comunicação entre o sistema e o usuário, através da rede GSM e a forma de instalação.

Como ponto fraco, o ausente controle do Sistema de Alarme de Intrusão através do SMS pelo usuário final.

CAPÍTULO 6 – CONCLUSÃO

6.1 – Conclusões

Este trabalho teve como objetivos desenvolver e implementar um sistema de alarme utilizando a troca de dados (utilizando o envio e recebimento de SMS) por meio da rede de telefonia celular, promovendo a interligação entre o *modem* de dados, o circuito de proximidade infravermelho e o microcontrolador 8051 e o telefone celular; desenvolvendo um circuito que utilize um sensor que seja capaz de identificar a presença quando ativado, em um ambiente “protegido”, para enviar um sinal ao microcontrolador para que o mesmo identifique, processe, execute e envie uma informação para o *modem* de dados; fazendo com que o *modem* envie uma SMS, cujo número do aparelho telefônico móvel já está previamente armazenado, alertando-o sobre a intrusão em seu ambiente.

O desenvolvimento do Sistema de Alarme de Intrusão utilizando o SMS proporcionou a integração de diferentes tecnologias na comunicação com os componentes envolvidos, como a utilização da rede de telefonia celular GSM e o controle do mesmo, através de um microcontrolador, que resultou em uma ponte para a comunicação do circuito que contém o sensor com o modem.

A primeira conclusão obtida neste trabalho é de que é possível desenvolver um Sistema de Alarme de Intrusão utilizando o SMS,

A segunda conclusão é de que é possível, após a configuração e montagem dos dispositivos envolvidos, a comunicação entre o *modem* e o circuito tendo como “ponte” o microcontrolador.

A terceira conclusão é a de que, após a detecção por meio de um sensor de um Intrusão, o microcontrolador foi capaz de identificar o sinal, processar e executar enviando um comando já previamente configurado no *modem* que enviará um alerta para o usuário final.

Considerando-se que este projeto foi elaborado e produzido com ferramentas e dispositivos facilmente disponíveis no mercado, acredita-se que o mesmo pode contribuir de forma prática para todos àqueles envolvidos na Segurança Pública e Privada.

Do ponto de vista acadêmico, espera-se ter contribuído para o debate teórico e prático em torno dos temas aqui abordados, bem como estimulado a realização de novos projetos, contribuindo assim para o avanço da Engenharia.

6.2 – Sugestões para Trabalhos Futuros

Como sugestões para trabalhos futuros que poderão ser desenvolvidos, pode-se implementar um sistema de alarme de controle relacionados a abertura e o fechamento de entradas ou saídas de um ambiente, como portas e/ou janelas.

Uma segunda sugestão de estudos adicionais está no desenvolvimento de um sistema capaz de se comunicar com uma rede de sensores sem fio, possibilitando a aquisição da informação por meio da rede de sensores e envio dos resultados a uma central que trabalhe com o monitoramento de grandes extensões territoriais e que necessite o controle em diferentes localidades de variáveis como **umidade, temperatura e pressão**.

Tem-se como terceira sugestão, a inclusão de todo o sistema de alarme em um único invólucro, onde estarão presentes o circuito de proximidade infravermelho, o microcontrolador e o modem e, também, a confecção de uma placa de circuito impressa.

Como quarta sugestão, a possibilidade de alteração remotamente pelo SMS através do celular, para a alteração do número a ser configurado no envio da mensagem quando o sistema for ativado.

REFERÊNCIAS

ALBUQUERQUE, P. U. B., THOMAZINI, D. Sensores Industriais: fundamentos e Aplicações 1ª Ed. Érica, São Paulo, 2005.

FARREL, M. J., LEVINE J. R., JOSTEIN A., PEARCE J., APPLEQUIST D.. Mobile Internet for Dummies Ed. Wiley, 2008.

HENRY-LABORDÈRE, A. SMS and MMS Internetworking in Mobile Networks 1ª Ed Artech House mobile communications series, Massachussets, 2004.

NICOLOSI, D. E. C., BRONZERI, R. B.. Microcontrolador 8051 com Linguagem C: Prático e didático: Família AT89S8252, 2ª Ed. Érica, São Paulo, 2009.

PAZ, G. R., LERMES, R. S. R. Microcontroladores 8051: Programação em C, 2009.

ROSÁRIO, J. M.. Princípios de mecatrônica 1ª Ed. Prentice Hall, São Paulo, 2005.

SOARES, M. J.. Saber Eletrônica Nº 424 Ed. Saber, 2008.

SVERZUT, J. U. Redes GSM, GPRS, EDGE e UMTS: Evolução a Caminho da Quarta Geração (4G) 2ª Ed. Érica, São Paulo, 2008.

ZELENOVSKY, R., MENDONÇA, A.. Microcontroladores – Programação e Projeto com a família 8051 Ed. MZ, 2005.

8051 – Tutorial for Embedded Engineers. Disponível em:

<http://www.computer-solutions.co.uk/info/micro-search/8051/8051_tutorial.htm>

Acesso: 15 de maio de 2010

ANATEL (notícia: Brasil possui mais de 180 milhões de acessos móveis 20 de maio de 2010). Disponível em:

<<http://www.anatel.gov.br/Portal/exibirPortalInternet.do>>

Acesso: 25 de maio de 2010

GSM. Disponível em:

<<http://www.gsmworld.com/technology/gsm/index.htm>>

Acesso: 10 de maio de 2010.

Vivo – Tarifas SMS. Disponível em:

<http://www.vivo.com.br/portal/servicos_vivo_torpedo_sms.php?WT.ac=euvivo.servicos-mensagem.icone.vivotorpedosms>

Acesso: 20 de maio de 2010.

ANEXOS

Biblioteca REGX52.H

```
/*-----
```

AT89X52.H

Header file for the low voltage Flash Atmel AT89C52 and AT89LV52.
Copyright (c) 1988-2002 Keil Elektronik GmbH and Keil Software, Inc.
All rights reserved.

```
-----*/
```

```
#ifndef __AT89X52_H__
```

```
#define __AT89X52_H__
```

```
/*-----
```

Byte Registers

```
-----*/
```

```
sfr P0    = 0x80;
```

```
sfr SP    = 0x81;
```

```
sfr DPL   = 0x82;
```

```
sfr DPH   = 0x83;
```

```
sfr PCON  = 0x87;
```

```
sfr TCON  = 0x88;
```

```
sfr TMOD  = 0x89;
```

```
sfr TL0   = 0x8A;
```

```
sfr TL1   = 0x8B;
```

```
sfr TH0   = 0x8C;
```

```
sfr TH1   = 0x8D;
```

```
sfr P1    = 0x90;
```

```
sfr SCON  = 0x98;
```

```
sfr SBUF  = 0x99;
```

```
sfr P2    = 0xA0;
```

```
sfr IE    = 0xA8;
```

```

sfr P3    = 0xB0;
sfr IP    = 0xB8;
sfr T2CON = 0xC8;
sfr T2MOD = 0xC9;
sfr RCAP2L = 0xCA;
sfr RCAP2H = 0xCB;
sfr TL2   = 0xCC;
sfr TH2   = 0xCD;
sfr PSW   = 0xD0;
sfr ACC   = 0xE0;
sfr B     = 0xF0;

/*-----
P0 Bit Registers
-----*/
sbit P0_0 = 0x80;
sbit P0_1 = 0x81;
sbit P0_2 = 0x82;
sbit P0_3 = 0x83;
sbit P0_4 = 0x84;
sbit P0_5 = 0x85;
sbit P0_6 = 0x86;
sbit P0_7 = 0x87;

/*-----
PCON Bit Values
-----*/
#define IDL_    0x01

#define STOP_   0x02
#define PD_    0x02 /* Alternate definition */

#define GF0_   0x04
#define GF1_   0x08

```

```
#define SMOD_ 0x80

/*-----
TCON Bit Registers
-----*/
sbit IT0 = 0x88;
sbit IE0 = 0x89;
sbit IT1 = 0x8A;
sbit IE1 = 0x8B;
sbit TR0 = 0x8C;
sbit TF0 = 0x8D;
sbit TR1 = 0x8E;
sbit TF1 = 0x8F;

/*-----
TMOD Bit Values
-----*/
#define T0_M0_ 0x01
#define T0_M1_ 0x02
#define T0_CT_ 0x04
#define T0_GATE_ 0x08
#define T1_M0_ 0x10
#define T1_M1_ 0x20
#define T1_CT_ 0x40
#define T1_GATE_ 0x80

#define T1_MASK_ 0xF0
#define T0_MASK_ 0x0F

/*-----
P1 Bit Registers
-----*/
sbit P1_0 = 0x90;
sbit P1_1 = 0x91;
```

```

sbit P1_2 = 0x92;
sbit P1_3 = 0x93;
sbit P1_4 = 0x94;
sbit P1_5 = 0x95;
sbit P1_6 = 0x96;
sbit P1_7 = 0x97;

```

```

sbit T2 = 0x90;    /* External input to Timer/Counter 2, clock out */
sbit T2EX = 0x91; /* Timer/Counter 2 capture/reload trigger & dir ctl */

```

```

/*-----

```

```

SCON Bit Registers

```

```

-----*/

```

```

sbit RI = 0x98;
sbit TI = 0x99;
sbit RB8 = 0x9A;
sbit TB8 = 0x9B;
sbit REN = 0x9C;
sbit SM2 = 0x9D;
sbit SM1 = 0x9E;
sbit SM0 = 0x9F;

```

```

/*-----

```

```

P2 Bit Registers

```

```

-----*/

```

```

sbit P2_0 = 0xA0;
sbit P2_1 = 0xA1;
sbit P2_2 = 0xA2;
sbit P2_3 = 0xA3;
sbit P2_4 = 0xA4;
sbit P2_5 = 0xA5;
sbit P2_6 = 0xA6;
sbit P2_7 = 0xA7;

```

```

/*-----
IE Bit Registers
-----*/
sbit EX0 = 0xA8;    /* 1=Enable External interrupt 0 */
sbit ET0 = 0xA9;    /* 1=Enable Timer 0 interrupt */
sbit EX1 = 0xAA;    /* 1=Enable External interrupt 1 */
sbit ET1 = 0xAB;    /* 1=Enable Timer 1 interrupt */
sbit ES  = 0xAC;    /* 1=Enable Serial port interrupt */
sbit ET2 = 0xAD;    /* 1=Enable Timer 2 interrupt */

sbit EA  = 0xAF;    /* 0=Disable all interrupts */

/*-----
P3 Bit Registers (Mnemonics & Ports)
-----*/
sbit P3_0 = 0xB0;
sbit P3_1 = 0xB1;
sbit P3_2 = 0xB2;
sbit P3_3 = 0xB3;
sbit P3_4 = 0xB4;
sbit P3_5 = 0xB5;
sbit P3_6 = 0xB6;
sbit P3_7 = 0xB7;

sbit RXD = 0xB0;    /* Serial data input */
sbit TXD = 0xB1;    /* Serial data output */
sbit INT0 = 0xB2;    /* External interrupt 0 */
sbit INT1 = 0xB3;    /* External interrupt 1 */
sbit T0  = 0xB4;    /* Timer 0 external input */
sbit T1  = 0xB5;    /* Timer 1 external input */
sbit WR  = 0xB6;    /* External data memory write strobe */
sbit RD  = 0xB7;    /* External data memory read strobe */

/*-----

```

IP Bit Registers

```
-----*/
```

```
sbit PX0 = 0xB8;
```

```
sbit PT0 = 0xB9;
```

```
sbit PX1 = 0xBA;
```

```
sbit PT1 = 0xBB;
```

```
sbit PS = 0xBC;
```

```
sbit PT2 = 0xBD;
```

```
/*-----
```

T2CON Bit Registers

```
-----*/
```

```
sbit CP_RL2= 0xC8; /* 0=Reload, 1=Capture select */
```

```
sbit C_T2 = 0xC9; /* 0=Timer, 1=Counter */
```

```
sbit TR2 = 0xCA; /* 0=Stop timer, 1=Start timer */
```

```
sbit EXEN2= 0xCB; /* Timer 2 external enable */
```

```
sbit TCLK = 0xCC; /* 0=Serial clock uses Timer 1 overflow, 1=Timer 2 */
```

```
sbit RCLK = 0xCD; /* 0=Serial clock uses Timer 1 overflow, 1=Timer 2 */
```

```
sbit EXF2 = 0xCE; /* Timer 2 external flag */
```

```
sbit TF2 = 0xCF; /* Timer 2 overflow flag */
```

```
/*-----
```

T2MOD Bit Values

```
-----*/
```

```
#define DCEN_ 0x01 /* 1=Timer 2 can be configured as up/down counter */
```

```
#define T2OE_ 0x02 /* Timer 2 output enable */
```

```
/*-----
```

PSW Bit Registers

```
-----*/
```

```
sbit P = 0xD0;
```

```
sbit F1 = 0xD1;
```

```
sbit OV = 0xD2;
```

```
sbit RS0 = 0xD3;
```

```
sbit RS1 = 0xD4;
sbit F0 = 0xD5;
sbit AC = 0xD6;
sbit CY = 0xD7;
```

```
/*-----
```

Interrupt Vectors:

Interrupt Address = (Number * 8) + 3

```
-----*/
```

```
#define IE0_VECTOR    0 /* 0x03 External Interrupt 0 */
#define TF0_VECTOR    1 /* 0x0B Timer 0 */
#define IE1_VECTOR    2 /* 0x13 External Interrupt 1 */
#define TF1_VECTOR    3 /* 0x1B Timer 1 */
#define SIO_VECTOR    4 /* 0x23 Serial port */

#define TF2_VECTOR    5 /* 0x2B Timer 2 */
#define EX2_VECTOR    5 /* 0x2B External Interrupt 2 */
```

```
#endif
```