

UNIVERSIDADE ESTÁCIO DE SÁ
Curso de engenharia elétrica com ênfase em
telecomunicações

IGOR MARQUES CARNEIRO DA SILVA

SISTEMA DE CONTROLE REMOTO DE
EQUIPAMENTOS DE TELECOMUNICAÇÕES PARA
CONTINGENCIAMENTO EM CASO DE FALHA

Rio de Janeiro

2013

IGOR MARQUES CARNEIRO DA SILVA

**SISTEMA DE CONTROLE REMOTO DE
EQUIPAMENTOS DE TELECOMUNICAÇÕES PARA
CONTINGENCIAMENTO EM CASO DE FALHA**

Trabalho de Conclusão de Curso apresentado a Universidade Estácio de Sá, sob orientação do Professor Alexander Mazolli Lisboa como requisito parcial para obtenção do grau de Engenheiro em Telecomunicações.

Rio de Janeiro

2013

IGOR MARQUES CARNEIRO DA SILVA

**SISTEMA DE CONTROLE REMOTO DE
EQUIPAMENTOS DE TELECOMUNICAÇÕES PARA
CONTINGENCIAMENTO EM CASO DE FALHA**

Trabalho de Conclusão de Curso apresentado a Universidade Estácio de Sá, sob orientação do Professor Alexander Mazolli Lisboa como requisito parcial para obtenção do grau de Engenheiro em Telecomunicações.

Aprovado em
BANCA EXAMINADORA

Prof. Dr. José Brant de Campos
Universidade Estácio de Sá

Prof. Dr. Francisco Jadson Miranda Viana
Universidade Estácio de Sá

Prof. Dr. Gilberto Rufino de Santana
Universidade Estácio de Sá

DEDICATÓRIA

Dedico este trabalho, em especial, aos meus familiares, principalmente a minha esposa Bárbara Cristina Chaves de França Marques Carneiro, e minha filha Alice Chaves de França Marques Carneiro que estar por vir a este mundo, por todo apoio dado no período em que estive desenvolvendo este projeto.

Por fim, a todos que acreditaram e acreditam no meu potencial.

AGRADECIMENTOS

Ao Professor e Orientador Alexander Mazolli Lisboa, professor da Universidade Estácio de Sá, pelo apoio e orientação.

Ao Professor da Pontifícia Universidade Católica do Rio de Janeiro e Diretor da MLS Wireless S/A, Rogério Passy.

À MLS Wireless S/A, no qual possibilitou o desenvolvimento deste projeto e testes piloto em suas dependências.

RESUMO

O principal objetivo deste projeto é desenvolver um sistema que seja capaz de controlar equipamentos de telecomunicações que estejam inacessíveis pela rede da operadora, de forma que possibilite desligar e religar os equipamentos através do corte da alimentação elétrica. Para isso, o sistema estará ligado com um aparelho de telefone celular usado no qual, ao receber uma chamada, atenderá automaticamente esta chamada para receber comandos enviados por qualquer outro telefone nesta simples chamada de voz. Estes comandos oferecem a possibilidade de desligar ou religar uma tomada e também verificar se uma tomada esta ligada ou desligada. Para manter a segurança do sistema, estes comandos serão precedidos de uma senha de quatro dígitos que poderá ser alterada também através da ligação. Este sistema permite controlar separadamente oito tomadas, possibilitando assim a reinicialização apenas do equipamento que não esteja respondendo, de forma contingencial para reestabelecer o link de acesso e principalmente fornecendo um meio alternativo de acesso, que não seja o link da operadora.

Palavras-chave: Controle remoto, equipamentos de telecomunicações, alimentação elétrica, telefone celular, telefone celular usado, tomadas, segurança, senha, reinicialização, contingencial, reestabelecer o link.

RÉSUMÉ

L'objectif principal de ce projet est celui de développer un système qui soit capable d'effectuer un contrôle des équipements de télécommunications qui sont inaccessibles à travers le réseau de l'opérateur téléphonique, permettant d'éteindre et de rallumer les équipements en coupant l'alimentation électrique. Pour cela, le système sera connecté à un téléphone mobile usagé, qui en recevant un appel, répondra automatiquement pour recevoir des commandements envoyés depuis n'importe quel autre téléphone par un simple appel vocal. Ces commandements offrent la possibilité de débrancher ou de rebrancher une prise et également de vérifier si une prise est branché ou débranché . Pour maintenir la sécurité du système, ces pilotes seront précédés par un mot de passe à quatre chiffres qui peut également être changé par un appel. Ce système contrôle séparément huit prises , permettant de redémarrer seulement le matériel qui ne répond pas , étant une mesure d'urgence pour rétablir le lien d'accès et principalement pour offrir un moyen alternatif d'accès, qui n'est pas le réseau de l'opérateur.

Mots-clés: Télécommande, équipements de télécommunications, électricité, téléphone cellulaire, téléphone cellulaire utilisé, sockets, sécurité, mot de passe, redémarrage, urgence, rétablir le lien.

LISTA DE FIGURAS

Figura 1.1 – Topologia típica da operadora.....	12
Figura 2.1 – Pinagem MT8870DE.....	14
Figura 2.2 – ATmega168.....	16
Figura 2.3 – Pinagem do ATmega168.....	17
Figura 2.4 – Definição de pinagem segundo o arduino.....	17
Figura 2.5 – Tela do Software de desenvolvimento Arduino.....	19
Figura 2.6 – Esquema do opto acoplador 4n25.....	21
Figura 2.7 – Régua de tomadas.....	22
Figura 2.8 – Esquemático do circuito.....	24
Figura 2.9 – Chave de seleção alimentação dos relés.....	24
Figura 3.1 – Circuito final obtido através do Eagle.....	25
Figura 3.2 – Tela de Silk Screen pronta.....	26
Figura 3.3 – Placa com o desenho do circuito.....	27
Figura 3.4 – Corrosão da placa.....	28
Figura 3.5 – Placa após a corrosão.....	28
Figura 3.6 – Placa com a mascara de solda.....	29
Figura 3.7 – Furação da placa.....	30
Figura 3.8 – placa pronta.....	31
Figura 3.9 – montagem final.....	32
Figura 3.10 – Instalação POP-BARRA06.....	33

LISTA DE TABELAS

Quadro 2.1 – Tabela DTMF.....	14
Quadro 2.2 – Tabela da saída binária dos terminais.....	15
Quadro 2.3 – conversão binário para decimal.....	19
Quadro 2.4 – lista de comandos.....	20
Quadro 3.1 – Confecção por placa.....	34
Quadro 3.2 – Outros custos.....	35
Quadro 3.3 – Custos Hh e ART.....	35

SUMÁRIO

INTRODUÇÃO.....	9
CAPÍTULO 1 OBJETIVO.....	11
CAPÍTULO 2 DESENVOLVIMENTO.....	13
2.1 SISTEMA DTMF.....	13
2.2 MICROCONTROLADOR.....	15
2.3 SOFTWARE DE CONTROLE.....	18
2.4 DRIVER DE POTÊNCIA.....	20
2.5 RÉGUA DE TOMADAS.....	21
2.6 CIRCUITO FINAL.....	23
CAPÍTULO 3 IMPLEMENTAÇÃO.....	25
3.1 TRANSFERÊNCIA DA IMAGEM PARA A PLACA.....	26
3.2 CORROSÃO DA PLACA.....	27
3.3 APLICAÇÃO DA MASCARA DE SOLDA.....	29
3.4 FURAÇÃO DA PLACA.....	30
3.5 MONTAGEM FINAL DO SISTEMA.....	31
3.6 APLICAÇÃO PRÁTICA.....	33
3.7 CUSTO DO PROJETO.....	34
CAPÍTULO 4 SUSTENTABILIDADE.....	36
CAPÍTULO 5 CONCLUSÃO.....	38
REFERÊNCIAS BIBLIOGRÁFICAS.....	39
APÊNDICE A CÓDIGO FONTE DO PROGRAMA.....	40

INTRODUÇÃO

O mundo online já é uma realidade. A cada dia novas aplicações e equipamentos são lançados no mercado a característica de serem controlados ou monitorados via rede Internet. Podemos enumerar diversos eletrodomésticos tais como, geladeiras, máquinas de lavar e até cafeteiras que já possuem suas versões com acesso Internet integrado. Além desses equipamentos, podemos também facilmente encontrar em residências outros equipamentos altamente dependentes de acesso como câmeras de monitoramento IP, alarmes, telefones VoIP, e principalmente os roteadores de saída para rede Internet. Por outro lado, as empresas de telecomunicações estão aumentando suas redes em capacidade e capilaridade de modo a ofertar serviços com maior capacidade para o maior número possível de assinantes. Essas expansões geram um aumento do número de pequenas centrais remotas conhecidas como PoP “Point of presence [1]”, onde são abrigados equipamentos que controlam e distribuem acessos para uma determinada área ou até mesmo servindo de trânsito a outros PoPs. Alguns desses PoPs chegam a abrigar dezenas de equipamentos tais como roteadores, switches, modems óticos, rádios e outros equipamentos de telecomunicações. Apesar do padrão desses equipamentos de telecomunicações obedecerem normas mais restritas que os equipamentos domésticos, frequentemente algum desses equipamentos podem sofrer um “travamento”, perdendo todo o seu controle via rede. Nesses casos, o problema pode gerar uma paralisação do PoP que afetará todos os assinantes daquela região. Grande parte desses travamentos poderiam ser rapidamente resolvidos com uma simples reinicialização do equipamento (comumente conhecida como reboot), efetuada por um técnico no local onde o equipamento se encontra. Em casos onde PoP pode servir como trânsito para outros PoPs, qualquer paralisação pode produzir um efeito em cascata para milhares de assinantes.

Essa dependência crescente da rede Internet para uso doméstico ou residencial gera uma cobrança também crescente da qualidade e principalmente de disponibilidade de acesso sobre as operadoras que oferecem o serviço. Assim, qualquer falha de um PoP, pode produzir uma verdadeira avalanche de reclamações nos call-centers das operadoras e até sanções de suas agências reguladoras. Portanto, a agilidade na recuperação dos acessos se torna um indicador importante na qualidade dos serviços prestados aos assinantes além de produzir uma diminuição da quantidade de reclamações a serem atendidas pelas operadoras.

CAPÍTULO 1 OBJETIVO

A motivação para esse trabalho foi o desenvolvimento e implementação de um equipamento com capacidade de controle da alimentação elétrica de equipamentos de uma operadora de Internet banda larga e telefonia VoIP em seus PoP. Um dos principais problemas encontrado nessa operadora era a demora na recuperação de seus PoPs e até mesmo acessos de clientes, devido ao tamanho de sua rede.

Podemos ver a topologia de acesso típico dessa operadora na Figura 1.1. Atualmente o deslocamento urbano em certos lugares da cidade, torna um deslocamento de poucos quilômetros em uma viagem de muitas horas. Além deste deslocamento, alguns PoPs se encontram em telhados de prédios comerciais e residenciais de difícil acesso e até restrições em certos horários. Desta forma, em caso de falha, o serviço poderá ficar parcialmente ou totalmente inoperante durante muitas horas até que a equipe consiga chegar e ter acesso ao local. Em certos casos o tempo de recuperação pode chegar há mais de 6 horas, conseqüentemente causando uma grande quantidade de reclamações e até cancelamentos. Com dito anteriormente, no caso de PoPs onde existe retransmissão para outros PoPs o problema pode afetar milhares de assinantes. Grande parte desses eventos ocorriam simplesmente devido a paralisação de um equipamento onde uma simples reinicialização solucionava, temporariamente, o problema até a solução definitiva do problema ou troca do equipamento a ser efetuada. A utilização de tomadas controladas via IP como por exemplo a “IP Power Controller 9258T Aruca Electronics, [2]” tem como acesso o link da própria rede da operadora, que em muitas vezes, não solucionaria o problema, pois a paralisação pode afetar os equipamentos de borda da rede no PoP, isolando assim, o próprio dispositivo de recuperação, impedindo qualquer reinicialização remota. Assim, pensou-se em um equipamento que possuísse um acesso independente da rede da operadora que efetuasse uma reinicialização remota até a chegada do técnico ao local. Sabendo que nesses PoPs

a única rede de internet disponível era a da própria operadora, pensou-se em um equipamento que utilizasse a rede de telefonia celular, para controlar a alimentação elétrica dos equipamentos de telecomunicações. Assim, devido a essa necessidade, foi idealizado o sistema que será descrito a seguir nesse trabalho.

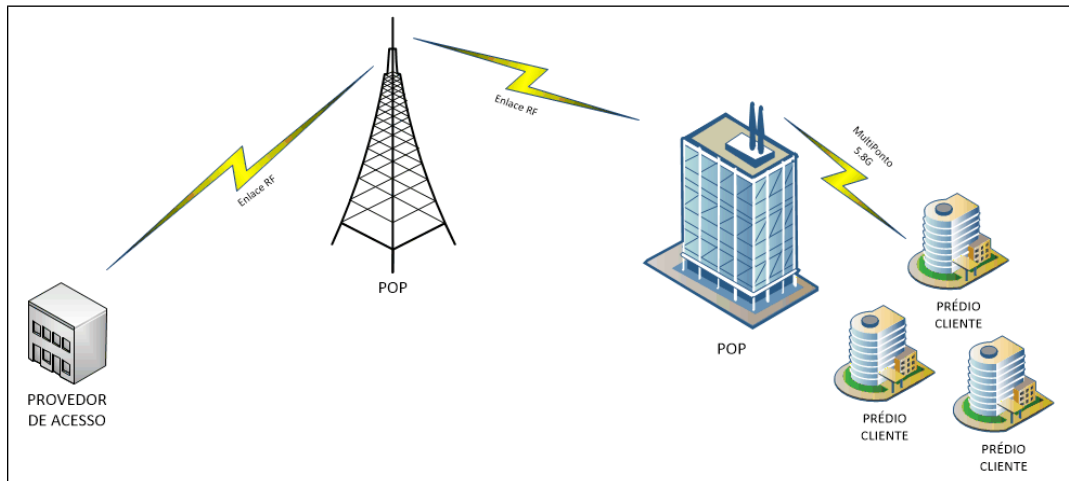


Figura 1.1 – Topologia típica da operadora

CAPÍTULO 2 DESENVOLVIMENTO

Nessa parte, serão descritos os detalhes técnicos utilizados no desenvolvimento desse projeto. Como base inicial de acesso foi utilizado um aparelho celular simples, podendo ser um aparelho usado, com a função de autoatendimento. A partir desse ponto foi considerado o acesso ao circuito através da interface Headset (headphones combinado com um microphone) do aparelho. O circuito foi desenhado para reconhecer tons DTMF [3] (*Dual-Tone Multi-Frequency*) no sinal de áudio transmitido através da chamada atendida automaticamente pelo aparelho celular. Assim, de qualquer ligação feita para o celular acoplado ao circuito, foi possível transmitir comandos para o circuito.

2.1 SISTEMA DTMF

Segundo a Wikipédia 2013, A sinalização DTMF foi desenvolvida nos laboratórios Bell (Bell Labs) visando permitir a discagem DDD, que usa enlaces sem fio como os de micro-ondas e por satélite.

DTMF é a sigla em inglês de "Dual-Tone Multi-Frequency", os tons de duas frequências utilizados na discagem dos telefones mais modernos.(Wikipédia, 2013);

A combinação das frequências que formam os tons DTMF, mostradas no quadro 2.1, onde o cruzamento de uma frequência alta, descrita na primeira linha e uma frequência baixa descrita na primeira coluna da esquerda. O cruzamento da linha com a coluna, ou seja o cruzamento das duas frequências, formam o respectivo tom DTMF.

Hz	1209	1336	1477	1633
697	1	2	3	A
770	4	5	6	B
852	7	8	9	C
941	*	0	#	D

Quadro 2.1 – Tabela DTMF (Fonte: Wikipédia,2013)

Para a detecção do tom DTMF, foi utilizado o CI(Circuito Integrado) MT8870DE [4] mostrado na figura 2.1. Esse CI tem a característica de detectar e decodificar todos os 16 pares dos tons DTMF e colocar em seus terminais de saída(Q4,Q3,Q2,Q1), um código de 4 bits (mostrado no quadro 2.2),

Segundo seu datasheet:

The MT8870D/MT8870D-1 monolithic DTMF receiver offers small size, low power consumption and high performance. Its architecture consists of a bandsplit filter section, which separates the high and low group tones, followed by a digital counting section which verifies the frequency and duration of the received tones before passing the corresponding code to the output bus. (p.4).

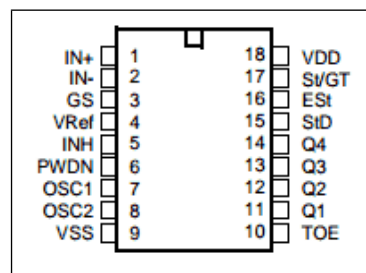


Figura 2.1 – Pinagem MT8870DE (Fonte: datasheet MT8870DE)

Digito	Q4	Q3	Q2	Q1
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
0	1	0	1	0
*	1	0	1	1
#	1	1	0	0

*Quadro 2.2 – Tabela da saída binária dos terminais.
(Fonte: datasheet MT8870DE)*

2.2 MICROCONTROLADOR

O microcontrolador utilizado foi o ATmega168 [5] (figura 2.2 e figura 2.3) fabricado pela Atmel Corporation. Apesar de ser um microcontrolador menos presente no mercado do que controladores baseados no PIC, o ATmega168 apresenta uma vantagem de ser o mesmo utilizado pelo projeto Arduino [6], uma ferramenta de programação, compilação e gravação de fácil utilização e vasta documentação e exemplos. Assim, foi possível a utilização de todas as ferramentas de desenvolvimento presente no Arduino, na fase inicial do desenvolvimento do software embarcado do projeto. As principais características do microcontrolador ATmega168 são:

- 16k de memória flash;
- 512 Bytes de EEPROM;
- 1k Byte de RAM;
- dois contadores de 8bit e um 16bit;
- 6 canais PWM(Pulse-width modulation);
- 8 canais 10-bit ADC em TQFP e MLF
- 6 canais 10-bit ADC em PDIP;
- 23 portas de entrada e saída programáveis;
- tensão de operação 2.7 a 5.5V;
- processador: 0 – 12 MHz @ 2.7 – 5.5V, 0 – 24 MHz @ 4.5 – 5.5V;
- baixo consumo: 1 MHz, 1.8V: 240µA, 32 kHz, 1.8V: 15µA;



Figura 2.2 – ATmega168 (Fonte: HWTech, 2008)

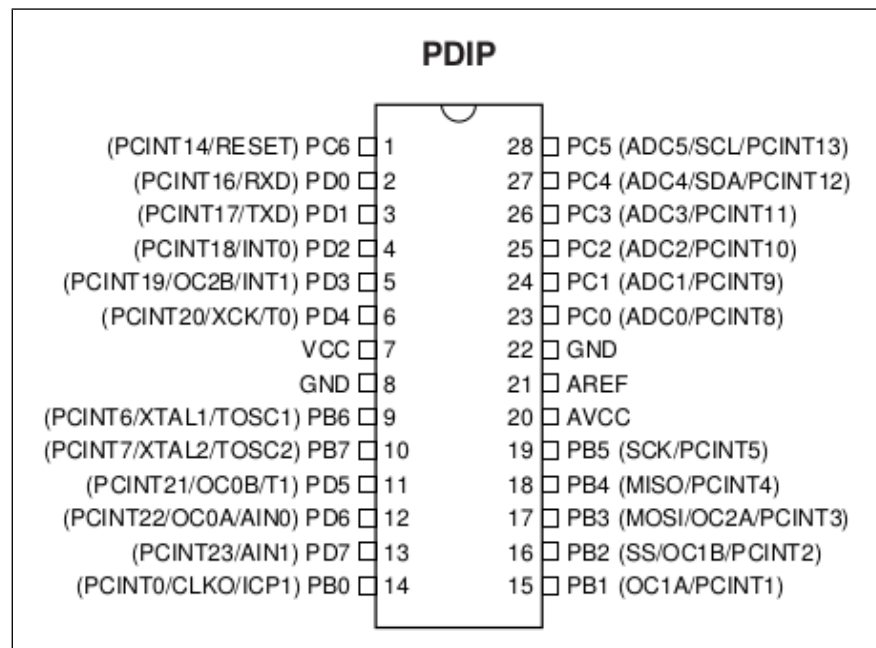


Figura 2.3 – Pinagem do ATmega168 (Fonte: datasheet ATmega168)

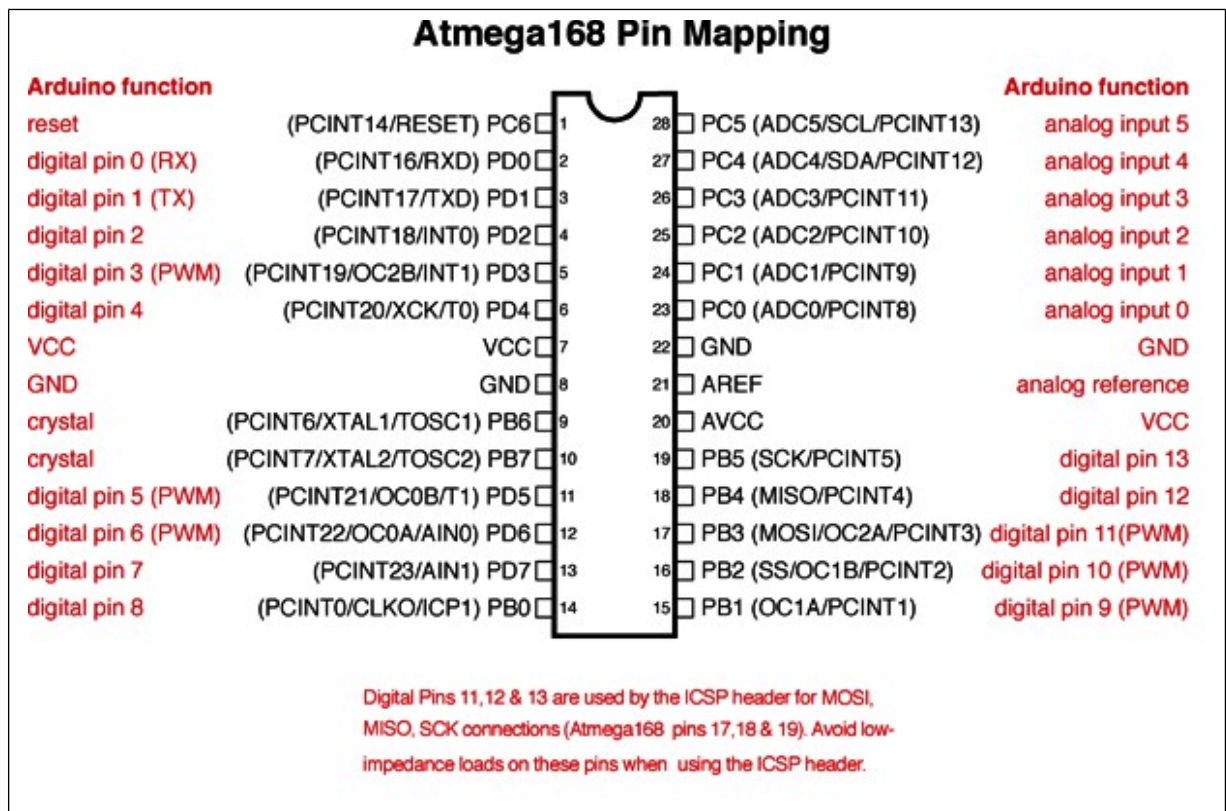


Figura 2.4 – Definição de pinagem segundo o arduino(Fonte: <http://arduino.cc>)

2.3 SOFTWARE DE CONTROLE

Como dito antes, foi utilizado o Arduino no desenvolvimento do software devido facilidade da interface de programação (Figura 2.5). Por sua vez, foi necessário gravar o bootloader do Arduino no microcontrolador. A linguagem utilizada no desenvolvimento do software embarcado foi a linguagem C, no qual encontra-se na íntegra no APÊNDICE A, que posteriormente foi compilado, pela IDE do Arduino, em instruções assembly em seguida gravada diretamente no microcontrolador.

Basicamente, o software funciona em um loop infinito, ou seja, ao final da instrução ele retorna ao começo consultando as entradas de informações, quando encontra uma entrada esperada, a instrução é direcionada para o devido tratamento em outras funções. O software verifica dois tipos de entradas, uma é a entrada no pino 25 que no código é definido pelo nome “A2” (ver figura 2.3), esta entrada está ligada em um botão do tipo push-button¹ que ao ser pressionado, coloca um sinal alto(5 volts), neste momento, no software, a intrusão ao reconhecer este sinal, aguarda 10 segundos e checa novamente, caso ainda assim o sinal continue alto, ou seja, o botão ainda está pressionado, inicia-se o processo de reset para a senha padrão, que neste caso foi definido como sendo a senha “1234”, diretamente na memória EEPROM em seguida retorna um sinal de áudio que será explicado a seguir. A outra checagem é a entrada no pino 15, que no código é chamado de porta “9”, esta entrada está ligada ao CI MT8870DE e recebe um sinal alto quando um código de 4 bits está disponível para leitura. Neste momento o software faz a leitura dos 4 bits nas portas Q1, Q2, Q3 e Q4 em seguida converte os bits em decimal utilizando a fórmula do quadro 2.3 e chama a função “criar_frase()” passando como parâmetro o número obtido que se refere ao dígito DTMF. Esta função primeiro retorna um sinal em formato de áudio com um tom breve indicando assim que o dígito foi recebido. Com isso, podemos saber se o dígito foi reconhecido ou até mesmo se foi recebido duplicado, muito comum em casos de echo² na telefonia.

1 “[...]such as press, depress, mash, and punch[...]” <http://en.wikipedia.org/wiki/Push-button>

2 É uma reflexão de um som que retorna a origem <http://en.wikipedia.org/wiki/Echo>

Depois, a função verifica qual dígito foi teclado e adiciona em uma variável externa³ para ser formado o comando. O dígito 12, que é a tecla “#”, indica que é o final daquele comando, chamando assim a função “dispara_comando()”. Esta função tem por finalidade analisar o conteúdo da variável externa, que chamamos de “frase”, onde foi salva uma certa quantidade de dígitos e executa processos diferentes para cada condição desta frase, como descrito no quadro 2.4.

$$(\text{bit04} * 8) + (\text{bit03} * 4) + (\text{bit02} * 2) + (\text{bit01} * 1);$$

Quadro 2.3 – conversão binário para decimal



```

projetoTCC_v3_4 | Arduino 1.0.5
File Edit Sketch Tools Help
projetoTCC_v3_4
#include <EEPROM.h>
#include <String.h>
#include <program.h>
#include <wiring_private.h>
#include <pins_arduino.h>
#define INTERNAL2V56 3
#define DEFAULT 3

// bits do DTMF
#define pinBit01 13 // Q0
#define pinBit02 12 // Q1
#define pinBit03 11 // Q2
#define pinBit04 10 // Q3
#define pinBitStd 9 // STD

// saída de audio
#define pinspk A0

//reset da senha
#define buttonPin A2

const int EEPROM_MTN_ADDR = 0;

Done Saving.
The sketch name had to be modified. Sketch names can only consist
of ASCII characters and numbers (but cannot start with a number).
They should also be less less than 64 characters long.
11 Arduino NG or older w/ ATmega168 on/dev/ttyACM0

```

Figura 2.5 – Tela do Software de desenvolvimento Arduino

3 Variável criada fora do contexto da função e será reconhecida por outras funções.

<p>comando para trocar a senha:</p> <p>conteúdo da frase: <senha atual>*<nova senha>*<nova senha novamente># exemplo: 1234*9876*9876#</p> <p>comando para desligar uma tomada:</p> <p>conteúdo da frase: <senha>*<número da tomada>*0# exemplo: 1234*1*0#</p> <p>comando para ligar uma tomada:</p> <p>conteúdo da frase: <senha>*<número da tomada>*1# exemplo: 1234*1*1#</p> <p>comando de reboot e ligar uma tomada:</p> <p>conteúdo da frase: <senha>*<número da tomada>*3# exemplo: 1234*1*3#</p> <p>comando para verificar o estado atual da tomada:</p> <p>conteúdo da frase: *<número da tomada># exemplo: *3#</p>
--

Quadro 2.4 – lista de comandos

Os comandos de trocar senha, desligar tomada, ligar tomada e reboot⁴, após serem executados retornam, em forma de áudio, um tom longo e agudo para informar o sucesso do comando. O comando que verifica o estado atual da tomada retorna um tom longo e agudo para informar que a tomada está no estado atual ligada ou um tom longo e grave para informar que a tomada está no estado atual desligada. Após a execução de cada comando, o conteúdo da frase é excluído, para guardar a próxima entrada.

2.4 DRIVER DE POTÊNCIA

Para a parte de driver de potência para o acionamento dos relés da régua de tomadas, foram utilizados oito opto acopladores LTV-4N25 [7] fabricado pela (Fairchild Semiconductor). O opto acoplador fornece o isolamento elétrico

4 Termo usado na área de TI que se refere a ação de reinicialização. <http://pt.wikipedia.org/wiki/Reboot>

conveniente, além de possibilitar o uso de uma fonte de tensão externa em seu segundo estágio. Desta forma, podemos usar este aparelho apenas para fornecer um curto nos 8 terminais de saída, possibilitando o acionamento do relé alimentado por uma outra fonte de tensão, além de simplesmente fechar um circuito com a alimentação do aparelho.

O opto acoplador tem um funcionamento bem simples, em seus terminais 1 e 2 esta ligado um diodo infravermelho e quando ativo por uma tensão em torno de 1.5v o diodo inicia a emissão do infravermelho ativando assim o transistor fotossensível nos terminais 4 e 5 como mostrado na figura 2.6.

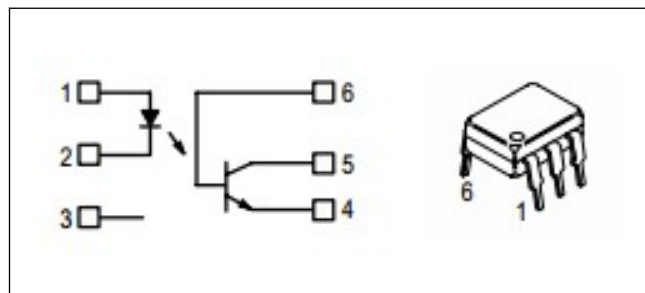


Figura 2.6 – Esquema do opto acoplador LTV-4n25 (Fonte: Datasheet)

2.5 RÉGUA DE TOMADAS

Para o controle das tomadas foram utilizados oito relés do tipo padrão de 5V, 127V 10A conectados como normalmente fechados e um diodo 1N4148 ligados em paralelo a bobina do relé, ficando dentro da estrutura da régua. Essa solução se mostrou mais prática pela possibilidade da troca e manutenção do circuito de controle sem o desligamento dos equipamentos nela ligados.

Segundo Newton C Braga [8]:

[...] O diodo tem como objetivo, proteger o circuito contra uma tensão inversa gerada no momento em que o relé é desenergizado havendo então a indução de uma tensão. Esta tensão tem polaridade oposta àquela que criou o campo e pode atingir valores muito altos. Podendo causar sérios danos ao circuito.[...] (2013)

A conexão da régua de tomadas com o circuito foi feita utilizando um cabo multi-vias com conector do tipo DB9. A caixa da régua possibilitou a instalação dos relés internamente sem necessidade de adaptações mecânicas, permitindo a proximidade entre o relés e as tomadas. Outra vantagem da régua escolhida, foi o seu padrão 19" permitindo a sua utilização em bastidor ou rack, padrões muito utilizados em telecomunicações. Figura 2.7.



Figura 2.7 – Régua de tomadas (Fonte: brdshop.com.br)

2.6 CIRCUITO FINAL

O circuito final obtido foi desenvolvido através do software “CadSoft EAGLE [9]” e que é mostrado na figura 2.8. Onde podemos notar a presença de um LED conectado na porta 5 no CI MT8870DE no qual tem a finalidade de sinalizar o reconhecimento do dígito DTMF, facilitando eventuais testes no equipamento. Foi colocado também um Trimpot⁵ na conexão dos pinos 2 e 3 do CI MT8870DE responsável pelo ajuste fino do ganho da entrada de áudio do CI. Como o circuito depende da utilização de um aparelho de telefone celular, disponibilizamos dois tipos de saída para a alimentação do aparelho de celular, uma saída do tipo plug USB devido a vasta utilização deste tipo de conexão para efetuar a recarga da bateria, e uma saída através de um conector do tipo PCF1-02, ambos com a própria alimentação de 5V do circuito. Para o acionamento dos relés, que estão dentro das tomadas, foi disponibilizado duas configurações controladas por dois jumper dispostos em paralelo e com duas posições em cada um como mostrado na Figura 2.9. Para que os relés utilizem a alimentação do próprio circuito, ou seja, a alimentação de 5V, basta ligar os jumpers na posição “INT”. Caso contrário, ligando os jumpers na posição “EXT” isolará o circuito completamente das conexões com os relés através do isolamento do opto acoplador. Desta forma será necessário que a régua de tomadas tenha a sua própria alimentação para os relés, possibilitando até mesmo a instalação de dispositivos de maior carga.

⁵ é um potenciômetro miniatura ajustável <http://pt.wikipedia.org/wiki/Trimpot>

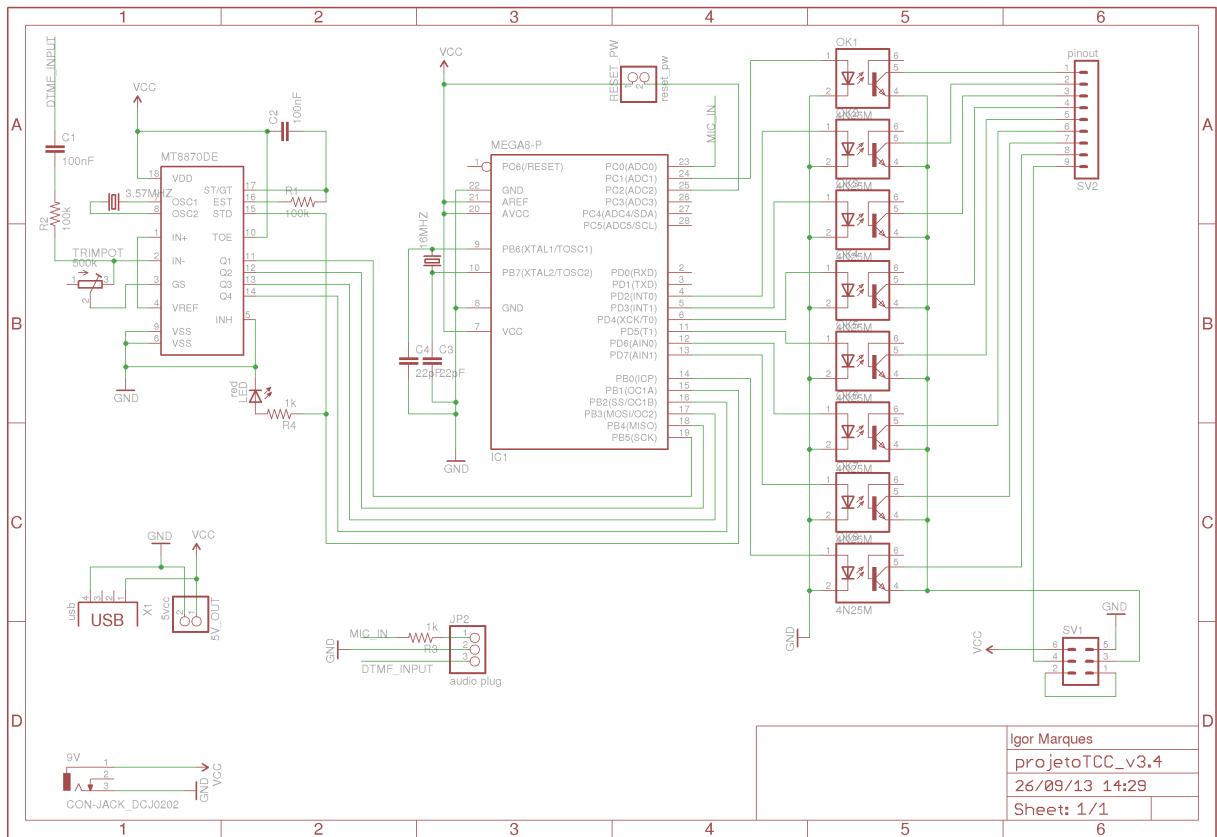


Figura 2.8 – Esquemático do circuito

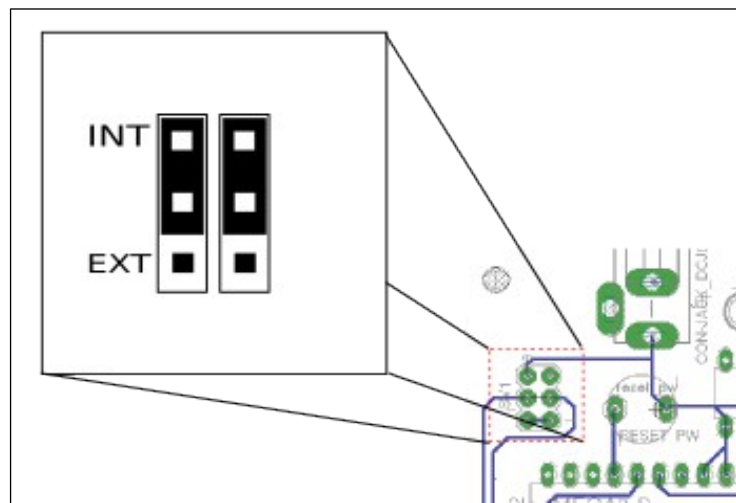


Figura 2.9 – Chave de seleção alimentação dos relés

CAPÍTULO 3 IMPLEMENTAÇÃO

Utilizado os circuitos acima descritos, foi desenhada uma placa de circuito impresso também com o uso do software “CadSoft EAGLE” (mostrado na Figura 3.1).

Segundo o site www.esquemas.org [10]:

Fisicamente, podemos descrever as placas de circuito impresso em geral, da seguinte maneira: Um substrato isolante, sendo o fenolite ou a fibra, com espessura de aproximadamente 1,6 mm; e uma superfície laminada condutora de cobre muito fina, aproximadamente de 0,05 mm, colada à base isolante, por processos industriais[...] (2013)

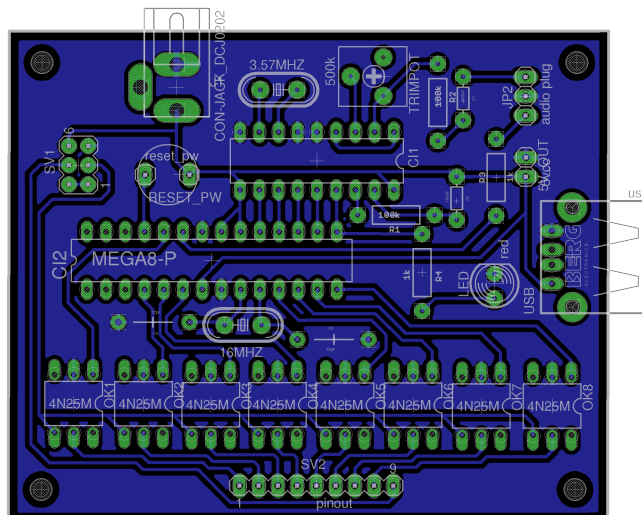


Figura 3.1 – Circuito final obtido através do Eagle

3.1 TRANSFERÊNCIA DA IMAGEM PARA A PLACA

Foram feitos testes com três formas de transferência do circuito para a placa. A transferência através de impressão a laser em papel fotográfico e transferido com a aplicação de calor para a placa. Este método se mostrou simples porém com uma qualidade em resolução muito ruim. O segundo método testado foi a utilização de DrY-Film, (uma película foto sensível a luz ultravioleta) este método se mostrou com a maior qualidade em termos de resolução porém a desvantagem deste método é que o DrY-Film é vendido em grandes quantidade e custo elevado chegando a um valor de R\$ 690,00 uma caixa com dois rolos de 20cm de largura e um comprimento de 153 metros (Dupiza,2013). Por último, e o escolhido para o desenvolvimento deste projeto, foi testado o método de Serigrafia no qual consistem na transferência da imagem impressa em papel transparente para uma tela serigráfica ou comumente chamado de SilkScreen através de um processo fotográfico (figura 3.2) para isso, foi utilizado o nylon de 180 fios por se tratar de uma imagem com muitos detalhes na ordem de 0,1mm.

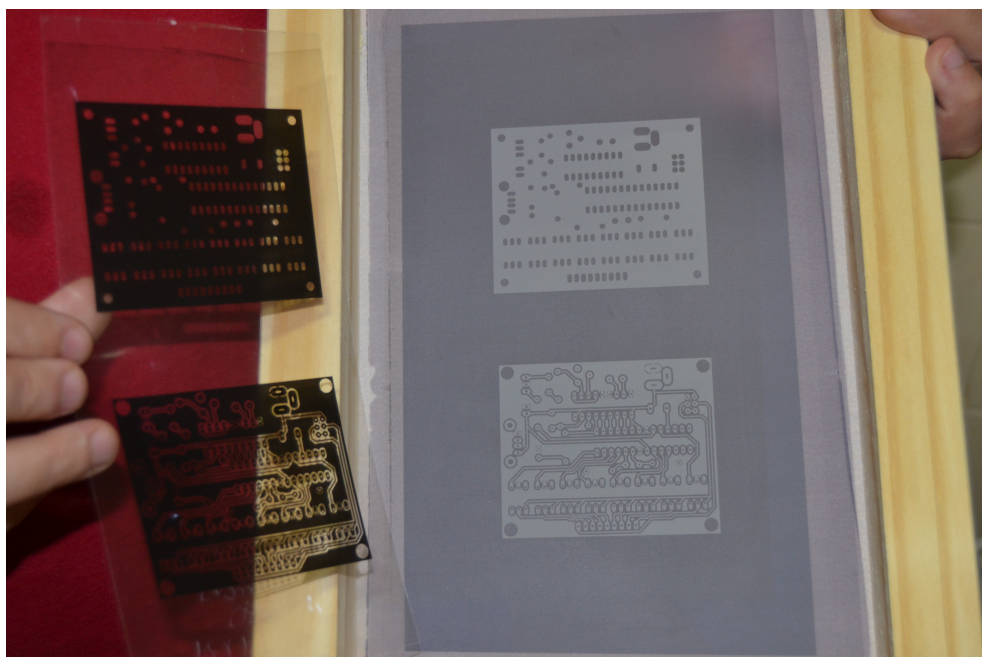


Figura 3.2 – Tela de Silk Screen pronta

Para este projeto, foi escolhido a placa do tipo “Fibra de Vidro”, apesar de ter seu custo elevado em comparação com as placas do tipo Fenolite, a sua qualidade é superior em termos de resistência mecânica e aderência do cobre a placa. Para a transferência da imagem, que já estava revelada na tela de Silk Screen, foi utilizado a tinta GN001-1 do fabricante Gênesis no qual é resistente a corrosão. Colocando a tela sobre a placa de fibra de vidro que já deve estar cortada no tamanho adequado, aplicamos a tinta por cima da tela em seguida espalhamos com um rodo próprio para Silk Screen. Ao final deste processo, obtemos a placa de fibra de vidro impresso com o desenho do circuito como mostra a figura 3.3. Após este processo, a placa apesar de ter o desenho do circuito, ainda assim continua com sua laminação de cobre por toda a placa sendo necessário o próximo passo de corrosão.

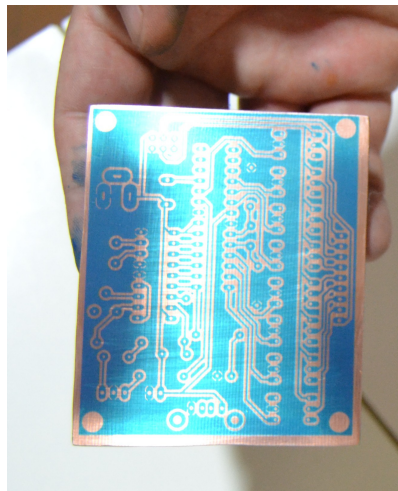


Figura 3.3 – Placa com o desenho do circuito

3.2 CORROSÃO DA PLACA

A tinta sobre a placa tem a finalidade de proteger o cobre no momento da corrosão, a parte onde o cobre da placa de fibra de vidro estiver exposto, ou seja não está coberto pela tinta GN001-1, deve ser retirado para que apenas sobre o

cobre que está coberto pela tinta formando assim o circuito final. Para ser retirado o cobre, existem diversos métodos de corrosão como a utilização de ácido nítrico (HNO_3), geralmente usado em processo de produção industrial para alta velocidade. Para processos artesanais podem ser usados o Percloroeto de ferro (FeCl_3) ou o Persulfato de amônia (NH_4). No caso de produção artesanal, o tempo de corrosão depende da área e da espessura do metal a ser corroído além das múltiplas reutilizações da solução. Em média para se corroer uma placa de dimensões 10 x 10 cm leva-se em torno de 10 minutos(Wikipedia, 2013), Para este projeto, usamos o Percloroeto de ferro como mostra a figura 3.4. Após a corrosão (figura 3.5), foi necessário retirar a tinta que ficou sobre as trilhas de cobre, para isso, colocamos a placa submersa em hidróxido de sódio “NaOH” e com o auxílio de uma pequena escova a tinta saiu com facilidade revelando assim a trilha de cobre que estava sob a tinta.

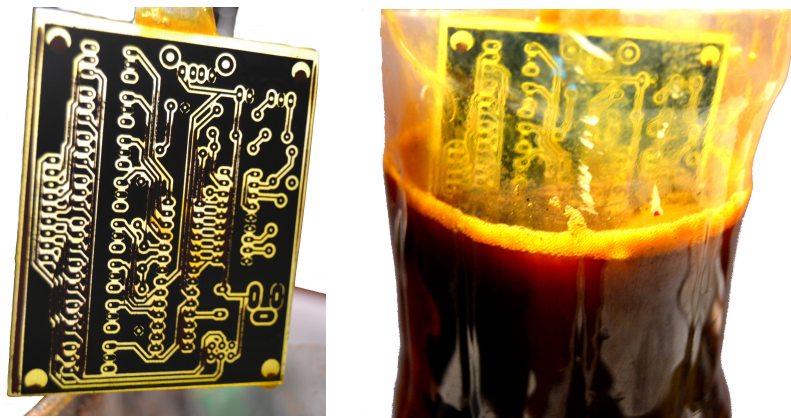


Figura 3.4 – Corrosão da placa

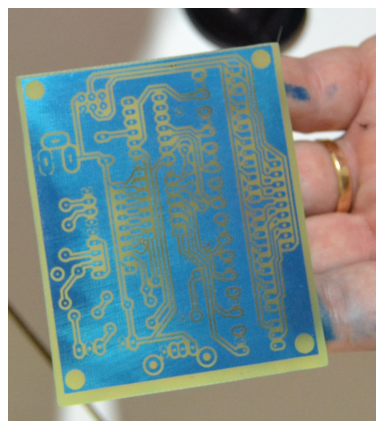


Figura 3.5 – Placa após a corrosão

3.3 APLICAÇÃO DA MASCARA DE SOLDA

Mascara de solda ou também conhecido em inglês como solder resist, é uma fina camada de tinta especial que resiste a alta temperatura no momento da solda para impedir que a solda encostes em outras vias ou no ground do circuito, no momento em que os componentes estão sendo soldados. A mascara de solda também tem como finalidade a proteção do cobre que fica exposto para evitar a corrosão com o tempo.

Existem diversos tipos de mascara de solda, como as do tipo DrY-Film já falado anteriormente e que também se encontra em forma de mascara de solda. Para este projeto foi utilizado o mesmo processo de serigrafia que usamos para transferir o desenho do circuito para a placa. A única diferença neste passo é que a tinta utilizada deve ser diferente pois a resistência necessária da tinta deixa de ser a corrosão para uma resistência ao calor e ao tempo. Para isto, utilizamos a tinta T2011-1 também do fabricante Gênesis. Podemos ver na figura 3.2 que aproveitamos a mesma tela de Silk Screen que usamos para o processo de transferência da imagem para abrigar a mascara de solda que da mesma forma transferimos para a placa do circuito pelo processo serigráfico onde podemos ver na figura 3.6 a placa com a mascara de solda.

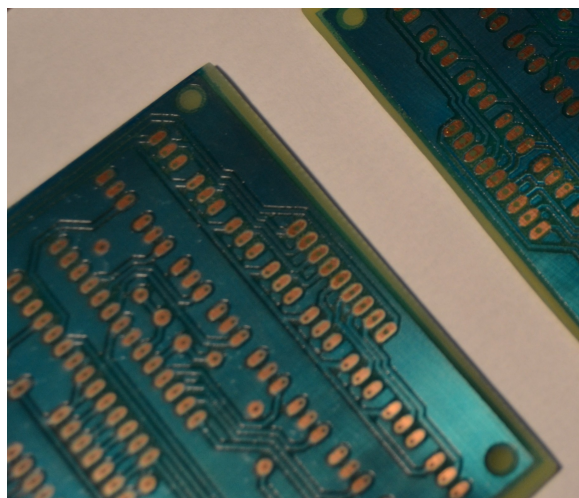


Figura 3.6 – Placa com a mascara de solda

3.4 FURAÇÃO DA PLACA

Para fazer os furos onde os componentes serão fixados, utilizamos uma minifuradeira própria para trabalhos onde exige maior precisão. Os furos foram feitos em dois tamanhos, para componentes com pinos mais grossos como os terminais dos CI's, utilizamos uma broca de 0,8mm e para os outros furos para componentes com pinos menores como por exemplo os capacitores e resistores, utilizamos uma broca de 0,6mm.

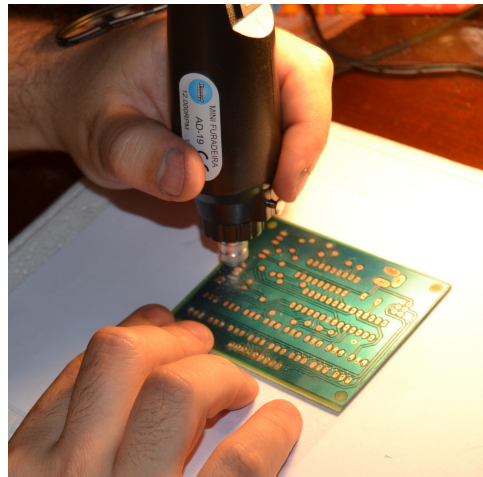


Figura 3.7 – Furação da placa

3.5 MONTAGEM FINAL DO SISTEMA

Cada componente foi soldado na placa utilizando um ferro de soldar de 60watts de potência. Para não danificar os CI's MT8870DE e o ATmega168, incluímos no projeto um socket para cada CI desta forma além de não danificarmos na soldagem, permite a posterior retirada para reprogramação no caso no CI ATmega168. Na figura 3.8 temos a placa finalizada com todos os componentes.

Para proteger a placa, a colocamos dentro de uma de plástico comumente chamada de caixa patola e muito usada para desenvolvimento de protótipos. Podemos notar na figura 3.9 que a caixa utilizada é um pouco maior que a placa, com isto temos a sobra de espaço suficiente para colocarmos um aparelho de telefone celular usado.

Para a conexão com o cabo de multi-vias, preferimos a instalação de um conector do tipo DB9, devido a sua grande resistência mecânica, acoplado apenas na caixa e interligados com fios paralelos na placa, desta forma a placa não sofreá com possíveis forças aplicadas ao se conectar o cabo externo de multi-vias.

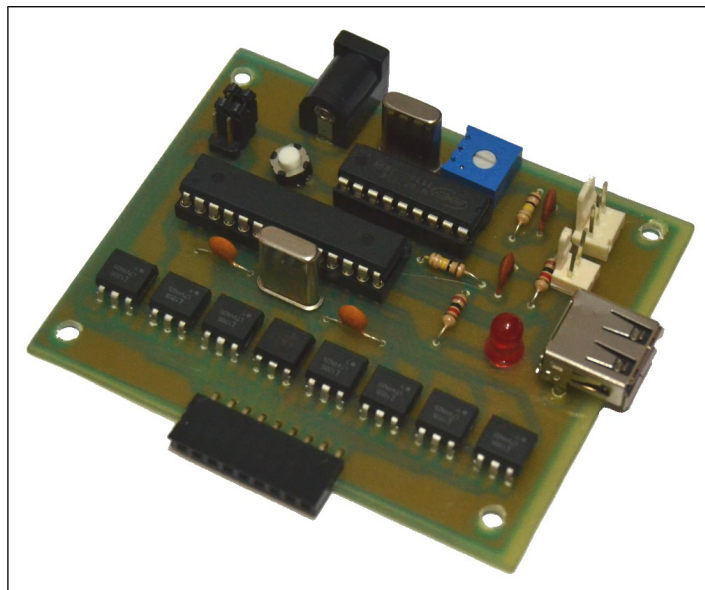


Figura 3.8 – placa pronta



Figura 3.9 – montagem final

3.6 APLICAÇÃO PRÁTICA

Foram feitos testes piloto na empresa MLS Wireless S/A, um provedor de acesso à internet banda larga e telefonia Voip no estado do Rio de Janeiro. Com a autorização e acompanhamento do atual presidente Rogério Passy, foram feitos testes, com o dispositivo proposto neste projeto, em um ponto de retransmissão na Barra da Tijuca zona oeste do Rio de Janeiro, lugar mais afetado pela demora de recuperação de acesso pelo fato de ser distante da sede da empresa que fica no bairro de Botafogo zona sul do Rio de Janeiro. Neste ponto de retransmissão mais conhecido internamente como POP foi instalado a régua em um bastidor ligando oito equipamentos, que são equipamentos de enlace de RF como o PowerBridge, NanoBridge e airFiber todos do fabricante UbiQUITi. Também foram ligados um switch Cisco-SF300 e um modem óptico. Na figura 3.10 podemos visualizar a instalação no rack de 19” onde a caixa com o sistema proposto neste projeto encontra-se instalada acima da régua de oito tomadas.

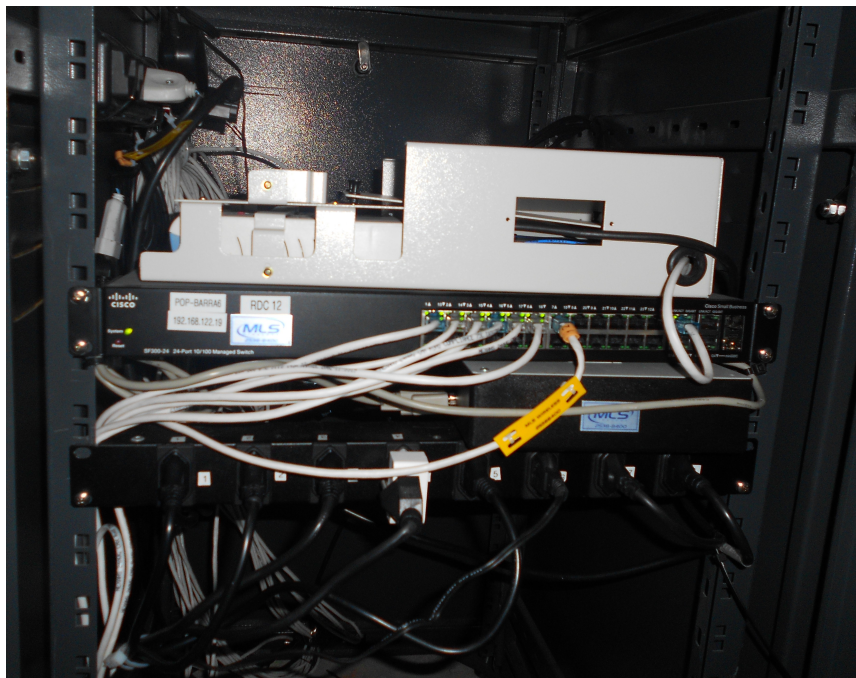


Figura 3.10 – Instalação POP-BARRA06

3.7 CUSTO DO PROJETO

N	material	quantidade	valor unitário
1	Cristal 3.57MHZ	1	R\$ 2,00
2	Cristal 16MHZ	1	R\$ 3,00
3	Capacitor 100nF	2	R\$ 0,10
4	Capacitor 22pF	2	R\$ 0,10
5	CM8870CP	1	R\$ 5,00
6	Atmega168	1	R\$ 11,00
7	DC Power Jack 2mm	1	R\$ 1,50
8	Conectores polarizados 3	1	R\$ 1,00
9	Conectores polarizados 2	1	R\$ 1,00
10	LED Vermelho	1	R\$ 0,10
11	Optoacoplador 4N25	8	R\$ 1,50
12	Resistor 100k	2	R\$ 0,10
13	Resistor 1k	2	R\$ 0,10
14	Botão push down	1	R\$ 1,50
15	Trimpot 500k	1	R\$ 1,00
16	USB Femea 90°	1	R\$ 2,50
17	Conector barra de pinos femea 9	1	R\$ 1,50
18	Conector barra de pinos macho 2x3	1	R\$ 1,50
19	Socket 18 pinos	1	R\$ 1,50
20	Socket 24 pinos	1	R\$ 2,00
21	Placa de fibra de vidro 20x30 cm	1/9	R\$ 30,00
total		32	R\$ 52,23

Quadro 3.1 – Confeção por placa

N	material	quantidade	valor unitário
1	Caixa patola	1	R\$ 5,00
2	Conector DB9 macho	1	R\$ 1,00
3	Relé 127VAC 5VDC	8	R\$ 1,50
4	Diodo 1N4148	8	R\$ 0,15
total		18	R\$ 19,20

Quadro 3.2 – Outros custos

N	tipo	UN.	Qt.	unitário	total
1	Custo Hh desenvolvimento	Hh	100	R\$ 50,00	R\$ 5000,00
2	Custo Hh implementação ~ 10 placas	Hh	5	R\$ 10,00	R\$ 50,00
3	Custo com ART	1	1	R\$ 60,00	R\$ 60,00

Quadro 3.3 – Custos Hh e ART

CAPÍTULO 4 SUSTENTABILIDADE

Você já conheceu alguém que precisava jogar fora um celular sem utilidade e não sabia onde descartar? Cerca de 40 milhões de toneladas de lixo eletrônico são gerados por ano no mundo. Entre os países emergentes, o Brasil é o país que mais gera lixo eletrônico.

Pesquisa da Fundação Santo André revela que 4.770 toneladas de celulares, incluindo baterias e carregadores, serão descartadas nos aterros sanitários do País neste ano. Em 2013, o montante chegará a 7.500 toneladas[...].

De acordo com a engenheira ambiental Nathália de Carvalho Aiolfi, que realizou o estudo, em 2010 foram vendidos 48 milhões de aparelhos, com peso médio de 100 gramas no País. "A vida útil é de dois a três anos e hoje existe muita dificuldade no descarte deste material. Grande parte vai para o lixo comum", afirma. No Estado de São Paulo são registrados 136 celulares a cada 100 habitantes. Em outubro de 2011 foram vendidos 57 mil aparelhos, aumento de 11,2% em relação ao mesmo período de 2010.

Com a Política Nacional de Resíduos Sólidos, instituída pela lei número 12.305 de 2 agosto de 2010 e que entrou em vigor no ano passado, os fabricantes, importadores, distribuidores e comerciantes são responsáveis pelo ciclo de vida dos produtos. A legislação também cria obrigações para os órgãos públicos de limpeza urbana e para os consumidores. Todos estão sujeitos a multas pelo não cumprimento da norma. As infrações variam de R\$ 500 a R\$ 10 milhões.

A nova lei obriga as indústrias e toda sua cadeia de fornecimento a gerenciar esses materiais, adotando medidas para a destinação correta e implementação de coletas seletivas.

Os consumidores também são obrigados a cumprir a legislação, devolvendo seu lixo eletrônico para a indústria.

Composição dos aparelhos celulares [11]

45% plástico;
10% cerâmicas;
20% cobre;
20% ouro, alumínio e outros metais;
5% não-metais. [9]

Problemas causados pelo descarte inadequado [12]

- Este descarte é feito quando o equipamento apresenta defeito ou se torna obsoleto (ultrapassado). O problema ocorre quando este material é descartado no meio ambiente. Como estes equipamentos possuem substâncias químicas (chumbo, cádmio, mercúrio, berílio, etc.) em suas composições, podem provocar contaminação de solo e água.
- Além do contaminar o meio ambiente, estas substâncias químicas podem provocar doenças graves em pessoas que coletam produtos em lixões, terrenos baldios ou na rua.
- Estes equipamentos são compostos também por grande quantidade de plástico, metais e vidro. Estes materiais demoram muito tempo para se decompor no solo. [10]

Neste projeto foi possível reaproveitar os celulares ultrapassado que a empresa teria que descartar para fornecer o meio de comunicação do circuito e contribuindo para a diminuição do lixo eletrônico causado rápida evolução tecnológica dos aparelhos de celulares. Um levantamento feito na empresa MLS Wireless, onde foram feitos os testes com o protótipo, levantou que em média a cada dois anos o contrato de telefonia celular em grupo da empresa é renovado e a cada renovação, a operadora oferece no pacote uma quantidade de aparelhos celulares novos e que substituem os aparelhos mais antigos dos funcionários.

CAPÍTULO 5 CONCLUSÃO

A solução apresentada mostrou-se mais eficiente, pois oferece um acesso alternativo a rede da operadora disponível no local. Além disso o custo da solução representa uma fração dos equipamentos disponíveis no mercado. A independência da régua também mostrou-se eficiente nos casos de troca e manutenção do circuito de controle, aumentando o uptime dos equipamentos controlados pelo sistema. A utilização de um telefone celular usado além de reduzir o custo com um possível módulo GSM a ser usado, contribui para a redução do lixo eletrônico causado pelo descarte de aparelhos de celular ultrapassado.

No futuro outras melhorias poderão ser feitas neste projeto como por exemplo a ampliação de portas de controle para abranger um número maior de tomadas a serem controladas. Ou também inserir a funcionalidade de checar se uma tomada efetivamente está sendo usada, ou seja passando corrente. Outra funcionalidade que poderia ser criada é o desenvolvimento de um aplicativo mobile para Android e iOS no quando pode ser cadastrado os números e definir botões referenciados a cada tomada, e quando o usuário clicasse no botão, o aplicativo iria efetuar uma ligação para aquele número e enviaria os tons DTMF pré-selecionado.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] , Point of presence,, Disponível em:
<http://en.wikipedia.org/wiki/Internet_service_provider>, Acesso em: 2 jul. 2013. .
- [2] Aruca Electronics, IP Power Controller 9258T,, Disponível em:
<<http://www.amazon.com/IP-Power-9258T-Network-Controller/dp/B005D55HKU>>, Acesso em: jul. 2013.
- [3] WIKIPÉDIA, a enciclopédia livre, DTMF,, Disponível em:
<<http://pt.wikipedia.org/w/index.php?title=DTMF&oldid=34485343>>, Acesso em: jul. 2013.
- [4] MITEL, Integrated DTMF Receiver,,Mai. 1995.
- [5] Atmel, Atmel Microcontroller,,Abr. 2011.
- [6] Arduino S/A, Arduino,, Disponível em: <<http://www.arduino.cc>>, Acesso em: Ago. 2012.
- [7] LITEON, General Purpose Type Photocoupler,,Set. 2013.
- [8] Segundo o Instituto Newton C Braga, ,, Disponível em:
<<http://www.newtoncbraga.com.br>>, Acesso em: Jul. 2012.
- [9] CadSoft Computer GmbH and CadSoft Inc, CadSoft EAGLE,, Disponível em:
<<http://www.cadsoftusa.com>>, Acesso em: fev. 2012.
- [10] Esquemas.org, ,, Disponível em: <<http://www.esquemas.org>>, Acesso em: Fev. 2013.
- [11] TIM CELULAR S.A, Composição dos aparelhos celulares,, Disponível em:
<<http://www.tim.com.br>>, Acesso em: Nov. 2013.
- [12] Pnuma - Programa da ONU para o Meio Ambiente, Problemas causados pelo descarte inadequado,, Disponível em: <http://www.suapesquisa.com/o_que_e/lixo_eletronico.htm>, Acesso em: Nov. 2013.

APÊNDICE A CÓDIGO FONTE DO PROGRAMA

```

#include <EEPROM.h>
#include <String.h>
#include <wprogram.h>
#include <wiring_private.h>
#include <pins_arduino.h>

#define INTERNAL2V56 3
#define DEFAULT 3

// bits do DTMF
#define pinBit01 13 // Q0
#define pinBit02 12 // Q1
#define pinBit03 11 // Q2
#define pinBit04 10 // Q3
#define pinBitStd 9 // STD

// saida de audio
#define pinspk A0

//reset da senha
#define buttonPin A2

const int EEPROM_MIN_ADDR = 0;
const int EEPROM_MAX_ADDR = 511;

const int BUFSIZE = 5;
char buf[BUFSIZE];
String myString;
char myStringChar[BUFSIZE];

String frase = "";

int bit01 = 0;
int bit02 = 0;
int bit03 = 0;
int bit04 = 0;
int tecla = 0;

// portas das tomadas
int tomadas[] = {0,A1,2,3,4,5,6,7,8};

// checagem se a tomada esta desligada ou ligada
int checar_on_off(uint8_t pin){
    uint8_t bit = digitalPinToBitMask(pin);
    uint8_t port = digitalPinToPort(pin);
    if (port == NOT_A_PIN)
        return LOW;

```

```

    return (*portOutputRegister(port) & bit) ? HIGH : LOW;
}

/***** instrucoes de leitura e gravacao na EEPROM *****/

boolean eeprom_is_addr_ok(int addr) {
    return ((addr >= EEPROM_MIN_ADDR) && (addr <= EEPROM_MAX_ADDR));
}

boolean eeprom_write_bytes(int startAddr, const byte* array, int numBytes) {
    int i;
    if (!eeprom_is_addr_ok(startAddr) || !eeprom_is_addr_ok(startAddr + numBytes))
        return false;

    for (i = 0; i < numBytes; i++)
        EEPROM.write(startAddr + i, array[i]);

    return true;
}

boolean eeprom_write_string(int addr, const char* string) {
    int numBytes;
    numBytes = strlen(string) + 1;
    return eeprom_write_bytes(addr, (const byte*)string, numBytes);
}

boolean eeprom_read_string(int addr, char* buffer, int bufSize) {
    byte ch;
    int bytesRead;
    if (!eeprom_is_addr_ok(addr)) {
        return false;
    }

    if (bufSize == 0) {
        return false;
    }

    if (bufSize == 1) {
        buffer[0] = 0;
        return true;
    }

    bytesRead = 0;
    ch = EEPROM.read(addr + bytesRead);
    buffer[bytesRead] = ch;
    bytesRead++;

    while ( (ch != 0x00) && (bytesRead < bufSize) && ((addr + bytesRead) <=
EEPROM_MAX_ADDR) ) {
        ch = EEPROM.read(addr + bytesRead);
        buffer[bytesRead] = ch;
        bytesRead++;
    }
}

```

```

    }

    if ((ch != 0x00) && (bytesRead >= 1)) {
        buffer[bytesRead - 1] = 0;
    }

    return true;
}

void comando_no(){
    tone(pinspk, 200,1000/4);
    delay(1000/2);
    noTone(pinspk);
    tone(pinspk, 200,1000/4);
}

void comando_yes(){
    tone(pinspk, 200,1000);
}

void criar_frase(int digito){

    if(digito == 12){
        dispara_comando();
        frase = "";
    }else if(digito == 11){
        frase += "*";
        tone(pinspk, 200,1000/7);
    }else if(digito == 10){
        frase += "0";
        tone(pinspk, 200,1000/7);
    }else{
        frase += String(digito);
        tone(pinspk, 200,1000/7);
    }
}

void dispara_comando(){
    char senha_salva[BUFSIZE];

    Serial.println("comando: ");
    Serial.print(frase);

    //consultar senha na EEPROM
    eeprom_read_string(0, senha_salva, BUFSIZE);

    if(String(senha_salva) == frase.substring(0,4)){
        // senha digitada confere
    }
}

```

```

if(frase.length() == 14){
    //procedimento de troca de senha

        if(frase.substring(5,9) == frase.substring(10,14)){
            // senhas iguais pode salvar
            myString = frase.substring(5,9);
            myString.toCharArray(myStringChar, BUFSIZE);
            strcpy(buf, myStringChar);
            eeprom_write_string(0, buf);
            comando_yes();
        }

    }else if(frase.length() == 8){
        //comando das tomadas

        if( (frase.substring(5,6).toInt() >= 1 ) && ( frase.substring(5,6).toInt() <= 8 ) ){

            if(frase.substring(7,8) == "0"){
                /***** desliga tomada *****/
                digitalWrite(tomadas[frase.substring(5,6).toInt()], HIGH);
                comando_yes();
            }else if(frase.substring(7,8) == "1"){
                /***** liga tomada *****/
                digitalWrite(tomadas[frase.substring(5,6).toInt()], LOW);
                comando_yes();
            }else if(frase.substring(7,8) == "3"){
                /***** desliga tomada por 3 s depois liga *****/
                digitalWrite(tomadas[frase.substring(5,6).toInt()], HIGH);
                delay(3000);
                digitalWrite(tomadas[frase.substring(5,6).toInt()], LOW);
                comando_yes();
            }
        }
    }

}

}else if(frase.length() == 2){
    if(frase.substring(0,1) == "**"){
        /*** comando para checar se a tomada esta ligada ou desligada ***/

        if( (frase.substring(1,2).toInt() > 0 ) && ( frase.substring(1,2).toInt() < 9 ) ){

            if(checar_on_off(tomadas[frase.substring(1,2).toInt()]) == 0){
                // tomada ligada
                tone(pinspk, 800,1000);
            }else{
                // tomada desligada
                tone(pinspk, 100,1000);
            }
        }
    }
}

}
}

void setup() {

```

```

Serial.begin(9600);

analogReference(DEFAULT);
pinMode(buttonPin, INPUT);
analogWrite(buttonPin, LOW);
pinMode(pinspk, OUTPUT);
noTone(pinspk);

// colocando estado inicial para todas ligadas
pinMode(A1, OUTPUT);
digitalWrite(A1, LOW);
pinMode(2, OUTPUT);
digitalWrite(2, LOW);
pinMode(3, OUTPUT);
digitalWrite(3, LOW);
pinMode(4, OUTPUT);
digitalWrite(4, LOW);
pinMode(5, OUTPUT);
digitalWrite(5, LOW);
pinMode(6, OUTPUT);
digitalWrite(6, LOW);
pinMode(7, OUTPUT);
digitalWrite(7, LOW);
pinMode(8, OUTPUT);
digitalWrite(8, LOW);
}

void loop(){

if(analogRead(buttonPin) > 1000 ){
  delay (10 * 1000);
  if(analogRead(buttonPin) > 1000 ){
    myString = "1234";
    myString.toCharArray(myStringChar, BUFSIZE);
    strcpy(buf, myStringChar);
    eeprom_write_string(0, buf);
    comando_no();
  }
}

if (digitalRead(pinBitStd) == HIGH) {
  bit01 = digitalRead(pinBit01);
  bit02 = digitalRead(pinBit02);
  bit03 = digitalRead(pinBit03);
  bit04 = digitalRead(pinBit04);
  tecla = (bit04 * 8) + (bit03 * 4) + (bit02 * 2) + (bit01 * 1);

  criar_frase(tecla);
  delay (500);
}
}

```