

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ  
CENTRO DE CIENCIAS EXATAS E DE TECNOLOGIA  
ENGENHARIA DA COMPUTAÇÃO

# PROJETO Radar

CURITIBA,

2010

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ  
CENTRO DE CIENCIAS EXATAS E DE TECNOLOGIA  
ENGENHARIA DA COMPUTAÇÃO

Projeto apresentado como requisito para avaliação do programa de aprendizagem de Microprocessadores II, do curso de graduação em Engenharia da Computação da Pontifícia Universidade Católica do Paraná, referente ao sexto período do mesmo. Com instrução do professor Afonso Ferreira Miguel. Tendo como alunos responsáveis pelo desenvolvimento: Carlos Alexandre Gouvêa da Silva, Edson Leonardo dos Santos e Rafael Veiga de Moraes.

CURITIBA,  
2010

## Abstract

The radar is something that is present in our daily life but is usually invisible. The air traffic control uses radar to track planes on the ground and in the air, in addition to also use it in time to guide the pilots to make soft landings. Besides, who also uses the radar is the police, but in order to detect the speed of automobiles. Since NASA uses to map the Earth and other planets, to track satellites and space debris and to help in time to maneuver their aircraft. The military, in turn, use radar to detect enemies and guide weapons to targets. Main project features are detecting speed of a car, processing speed and demonstration in a pack with display seven segments and record the plate of the vehicle to go above the speed limit.

## Sumário

|   |    |
|---|----|
| 1. Resumo   | 06 |
| 2. Objetivos  | 07 |
| 2.1. Objetivo Geral                                 | 07 |
| 2.2. Objetivo Especifico                            | 07 |
| 3. Lista de componentes                             | 08 |
| 4. Utilidades                                       | 09 |
| 4.1. Lista de materiais, equipamentos e ferramentas | 09 |
| 4.2 Softwares                                       | 09 |
| 5. Descrição do projeto                             | 10 |
| 5.1. Alimentação componentes eletrônicos            | 10 |
| 5.2. Controle sensores capacitivos                  | 10 |
| 5.3. Torre e display de sete segmentos              | 11 |
| 5.4. Maquete  | 13 |
| 5.5. Base 8051                                      | 14 |
| 5.6. Comunicação serial                             | 15 |
| 5.7. Código C                                       | 16 |
| 5.8. Módulo de disparo e câmera                     | 19 |
| 5.9. Código Visual Basic 6                          | 20 |
| 6. Outras Imagens                                   | 22 |
| 7. Conclusão  | 23 |
| 8. Referencias Bibliográfica                        | 24 |

## Índice de fotos

|   |    |
|---|----|
| Figura 1: Fonte alimentação                     | 10 |
| Figura 2: Schematic fonte alimentação (Eagle)   | 10 |
| Figura 3: Módulo controle sensores capacitivos  | 11 |
| Figura 4: Display de sete segmentos.            | 12 |
| Figura 5: Schematic display 7 segmentos (Eagle) | 12 |
| Figura 6: Torre para display sete segmentos     | 13 |
| Figura 7: Maquete. Pista e Torre                | 13 |
| Figura 8: Base 8051                             | 14 |
| Figura 9: Placa do Circuito Serial              | 15 |
| Figura 10: Schematic módulo serial              | 16 |
| Figura 11: Acoplamento câmera                   | 20 |
| Figura 12: Circuito disparo rápido              | 20 |
| Figura 13: Gráfico de monitoramento             | 21 |

## 1. Resumo

**P**rojetos de radar tem como finalidade, detectar e processar a velocidade média de um veículo. Tendo acoplado em seu conjunto, um sistema de detecção por imagem do indivíduo que ultrapassou um determinado limite de velocidade. Em muitos projetos de radares, a tecnologia utilizada é a de sinais sonoros para detecção de eco, sendo que os mesmos são do tipo radar móvel, e ainda radares que utilizam bobinas eletromagnéticas, que são do tipo radar fixo. Esse projeto tem como base usada sensores capacitivos que detectam o ponto de início e o ponto final de um corpo, sendo assim, calculando sua velocidade média em milímetros por segundo (mm/s). A criação de uma torre com display de sete segmentos para mostrar a velocidade medida também faz parte do desenvolvimento do projeto. O detector de imagem constituirá de uma câmera digital que através de um sinal do micro controlador 8051, irá capturar ou não a imagem do veículo em movimento.

## 2. Objetivos

### 2.1- Geral

Utilizando os conhecimentos adquiridos nas disciplinas de Eletrônica e Microprocessadores, construir um projeto que integre essas disciplinas e traga um entendimento prático e claro de cada recurso aprendido teoricamente. Criar um Radar que detecte a passagem de um veículo por sensores, calcule sua velocidade, demonstre a velocidade na torre e capture a imagem do veículo.

### 2.2 - Específicos

- I. Desenvolvimento de torre de velocidade utilizando display de sete segmentos;
- II. Desenvolvimento de um código em Visual Basic 6 para monitoramento da velocidade de todos os veículos;
- III. Desenvolvimento código em C (8051) para comunicação serial com micro computador, envio da velocidade calculada para a torre e tratamento dos sinais dos sensores capacitivos;
- IV. Desenvolvimento dos módulos eletrônicos como: comunicação serial, acionador da camera, display de sete segmentos unidade e dezena, fonte de tensão 12v e 5v.

### **3. Lista de componentes**

1 Regulador de tensão 7812  
1 Regulador de tensão 7805  
2 Diodos N4007  
2 Capacitores eletrolíticos 1000uF  
1 Capacitor eletrolítico 2200uF  
3 Capacitores de cerâmica 100nF  
2 Led's auto brilho  
1 Cristal oscilador 5,5MHz  
14 Resistores 330Ω  
2 CI CD4511  
Soquetes para CI's  
2 Displays 7 segmentos Catodo comum  
5 Capacitores de cerâmica 1μF  
1 CI Max232  
4 Resistores 1KΩ  
1 CI 4066  
1 Conector DB9  
2 reles 12Volts  
Conectores diversos modelos  
Soquetes



## 4. Utilidades

### 4.1 Lista de materiais, equipamentos e ferramentas

Fenolite;

Fios para conexão;

Multímetro digital;

Isopor;

Computador;

Transformador;

Microprocessadores 8051;

2 sensores capacitivos;

Alicate bico e corte;

Gerador de funções;

Osciloscópio.

### 4.2. Softwares

HyperTerminal – configuração e teste comunicação serial;

MicroShep – Gravação código C 8051;

µVision - desenvolvimento código C 8051;

Eagle – layout placa de circuito impresso e esquemático;

Visual Basic 6 – layout monitoramento de velocidades;

Video Maker – edição vídeo de apresentação.

## 5. Descrição do Projeto

### 5.1. Alimentação componente eletrônicos

Inicialmente para alimentação de todos os componentes do projeto tem-se usado uma fonte de alimentação do projeto anterior “CLP”, que constitui um transformador 127V/15V acoplado a um circuito regulador de tensão de saídas 12V e 5V. Usaremos a saída de 5 volts para alimentação do micro controlador 8051, módulo serial e módulo display sete segmentos. Para controle do sinal recebido dos sensores capacitivos é usado a tensão contínua de 12V.



Figura 1: Fonte alimentação.

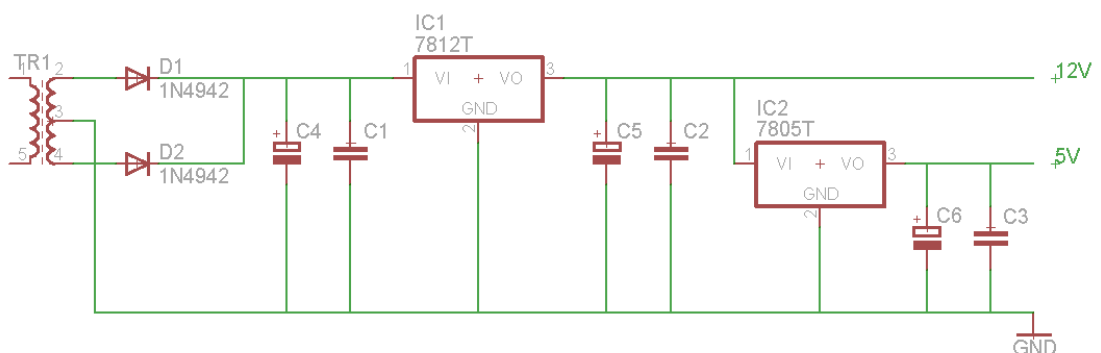


Figura 2: Schematic fonte alimentação (Eagle)

### 5.2. Controle sensores capacitivos

Para controle dos sensores capacitivos um circuito simples de entrada composto por dois reles é usado, isso se deve, pois o sensor capacitivo apenas disponibiliza o sinal lógico 0, e como é necessário a utilização do sinal 1 para reconhecimento do código em C o rele tem como finalidade mudar o sinal de 0 para 1. Esse módulo também é reutilizado do projeto anterior “CLP”. Como dito anteriormente esse circuito com reles é alimentado com 12V.

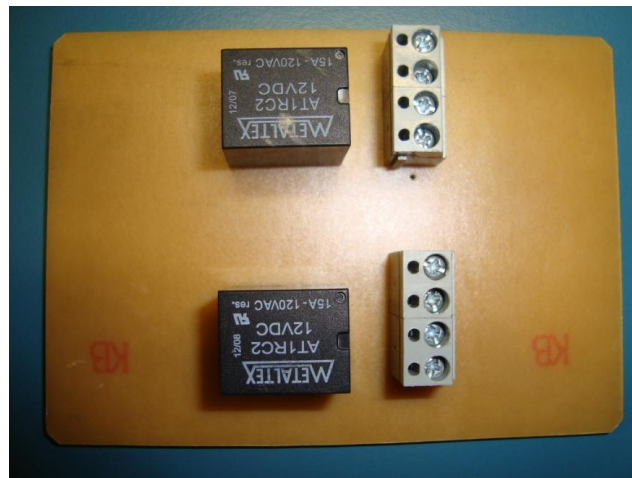


Figura 3: Módulo controle sensores capacitivos

### 5.3. Torre e display de sete segmentos

A torre do radar para acoplamento do display foi desenvolvido com madeira e pintada com tinta preta. Seu interior é oco de forma que se possa inserir o módulo do display de sete segmentos. O display de sete segmentos constitui-se de unidade e dezena acionada por 4 bits cada display. Cada conjunto de 4 bits é enviado para decodificadores CD4511 que codificam o valor binário de entrada (unidade ou dezena) em seu respectivo valor de saída que é enviado para cada led do display (a, b, c, d, e, f, g). Além de dois leds comuns acoplados logo acima de cada conjunto de resistores, refere-se à função de indicar que a torre está em funcionamento e outro indica quando a velocidade é excedida. As entradas do display são descritas da

seguinte forma: quatro bits de entrada acopladas no micro controlador 8051 da unidade e quatro bits de entrada acoplados no micro controlador 8051 da dezena, dois bits entrada acoplada no 8051 um para o led de ultrapassagem de limite de velocidade e outro para o led indicando radar em operação, uma entrada para alimentação 5 Volts e uma entrada de GND.

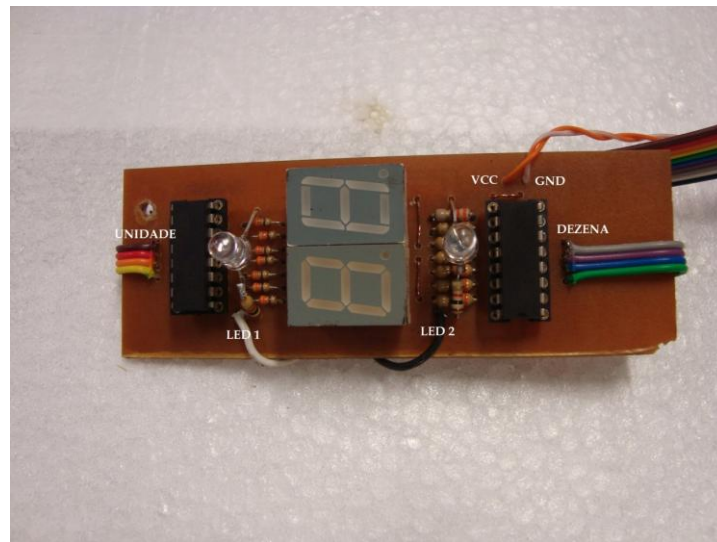


Figura 4: Display de sete segmentos.

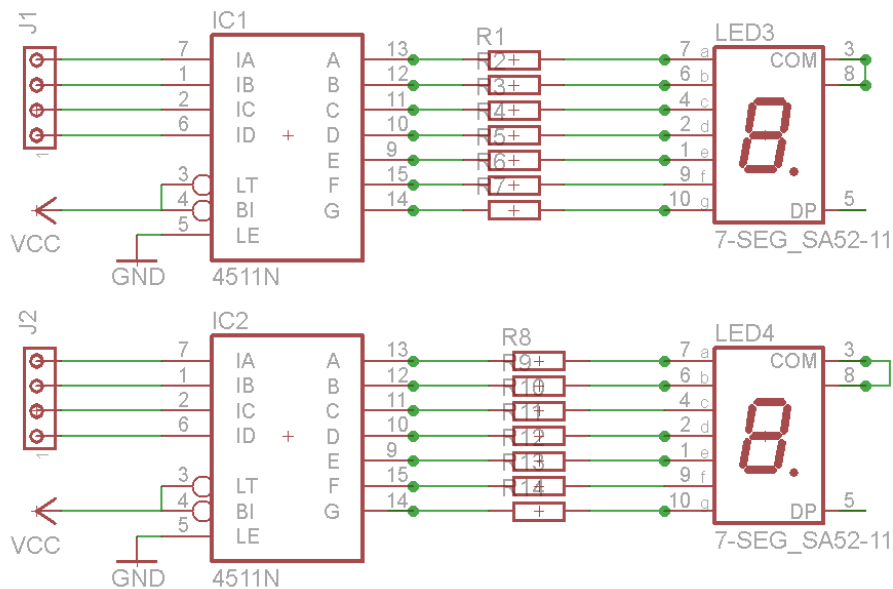


Figura 5: Schematic display 7 segmentos (Eagle)

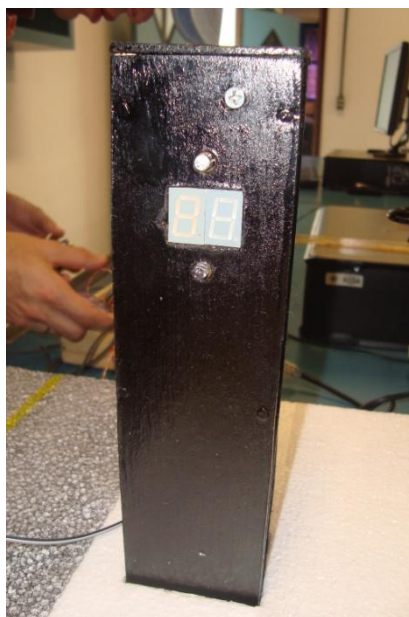


Figura 6: Torre para display sete segmentos

#### 5.4. Maquete

A base da maquete foi criada com uma chapa de isopor com o desenho similar de uma rua de duas vias. Em uma das vias é inserido dois sensores capacitivo com a distância de 10 centímetros um para o outro. Além disso é inserido na extremidade da pista a torre do display.

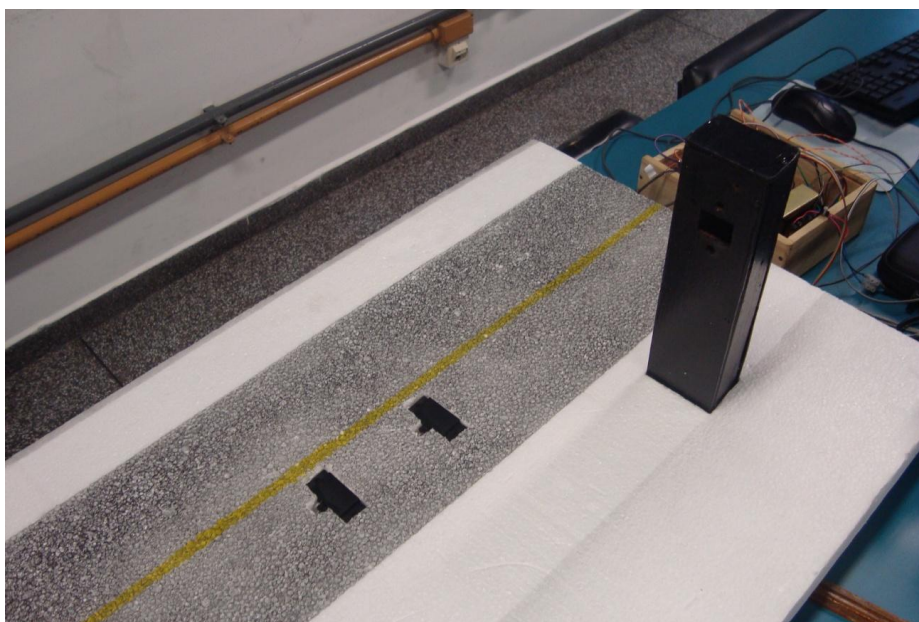


Figura 7: Maquete. Pista e Torre

## 5.5. Base 8051

O microprocessador 8051 é acoplado em uma base simples construída com um soquete de 40 pinos. Suas conexões são descritas na seguinte tabela:

| Pino    | Descrição   | Sinal   |
|---------|-------------|---|
| 1 a 8   | P1.0 à P1.7 | Saída velocidade para display sete segmentos  |
| 9       | RST         | Reset microprocessador                        |
| 10      | RXD         | Registrador RX para configuração serial       |
| 11      | TXD         | Registrador TX para configuração serial       |
| 12      | INT0        | Interrupção externa                           |
| 19      | XTAL1       | Frequência de uso comunicação serial          |
| 20      | GND         | Ponto de aterramento                          |
| 21 e 22 | P2.0 e p2.1 | Acionamento disparo Câmera                    |
| 31      | EA          | Ligado a VCC para desabilitar memória interna |
| 40      | VCC         | Alimentação positiva                          |

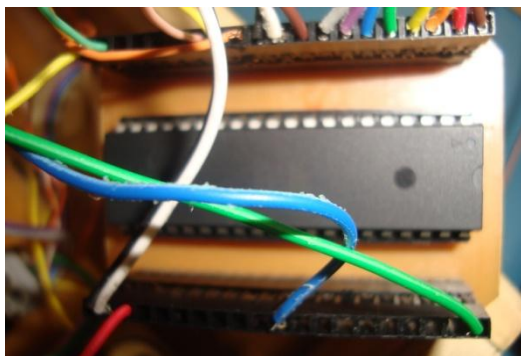


Figura 8: Base 8051

## 5.6. Comunicação serial

A comunicação baseada em um circuito MAX232 é usada para fazer a interface com o micro computador. É transmitido para o computador todos os dados que foram sendo registrados pelo micro controlador. Após ocorrer uma velocidade acima do permitido, pré definida pelo programador, é enviado através do 8051 um pulso para o módulo de acionamento que faz com que a câmera digital dispare e faça o registro do veículo em questão. Após testes realizados constatou-se que a frequência ideal para funcionamento da serial é de aproximadamente 5,52MHz. E em tempo real na tela do PC, aparece o registro de todas as datas e horas das velocidades registradas.

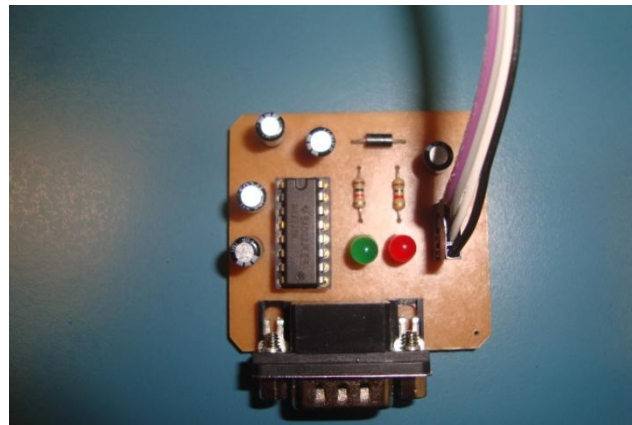


Figura 9: Placa do Circuito Serial

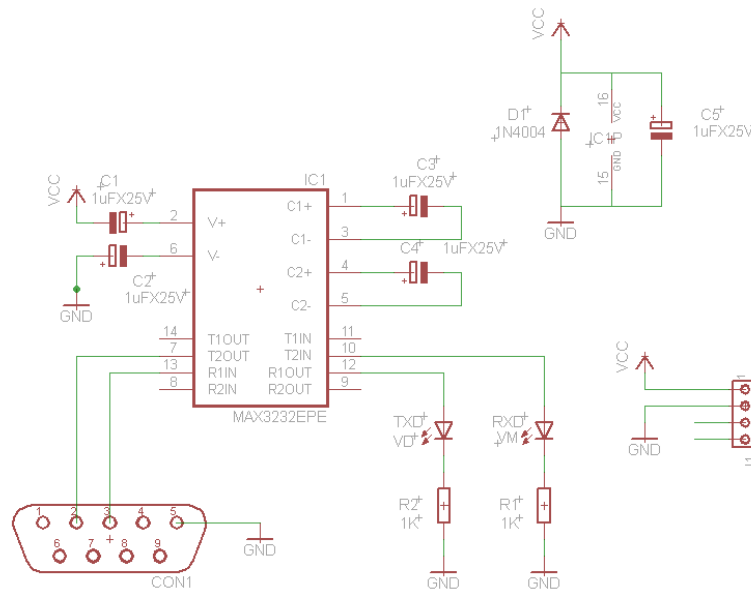


Figura 10: Schematic módulo serial

## 5.7. Código C

Para controle e monitoramento da velocidade além do monitoramento de todos os circuitos do projeto, criou-se um código em linguagem C que realiza a comunicação serial com PC, calcula e envia velocidade do veículo para o display por 8 bits de sinal, sinal de radar em operação (LED Amarelo), sinal de ultrapassagem de velocidade (LED Vermelho) e sinal de disparo da câmera.

```

*****
PROJETO RADAR
*****

CONFIGURAÇÃO SERIAL
Clock de 5,52M Hz / Baudrate -> 9600 / PCON -> SMOD = 1 / TH1 -> 253
*****

#include <REGX51.H>
#include <math.h>
#include <stdio.h>

// ----- Variáveis globais
float time; // Variável para armazenar o tempo
float speed; // Variável para armazenar a velocidade
unsigned short int frequencia; // Variável para armazenar a frequência de clock
unsigned short int distance; // Variável para armazenar a distancia entre os sensores em mm
int i, k, aux; // Variáveis auxiliares

```



```

// ----- IMPORTANTE:
// Para testar na serial eliminar '\r' e '\n'
char msg[15] = {'S', 't', 'a', 't', 'u', 's', ' ', 'O', 'K', ' ', ' ', ' ', ' ', ' '};
char msgv[15] = {'V', 'e', 'l', 'o', 'c', 'i', 'd', 'a', 'd', 'e', ' ', ' ', ' ', ' '};

// Delay
void delay(void) {
    for(k=0; k<10000; k++)
        continue;
    for(k=0; k<10000; k++)
        continue;
    for(k=0; k<10000; k++)
        continue;
    for(k=0; k<10000; k++)
        continue;
    for(k=0; k<10000; k++)
        continue;
    for(k=0; k<10000; k++)
        continue;
}

// Transmite a velocidade pela serial
void trans_velocidade() {
    i = 0;
    while(msgv[i] != NULL) {
        // Config timer 1
        TR1 = 0;
        TH1 = 0XFD;
        TL1 = 0;
        TR1 = 1;
        TI = 0;
        SBUF = msgv[i];
        while(!TI);
        i += 1;
    }
}

// Interrupção externa 0
void i0_irq(void) interrupt 0
{
    if( TR0 == 0 ) {
        // Zera timer
        TH0 = 0;
        TLO = 0;
        // Abaixa a bandeira do timer
        TFO = 0;
        // Dispara o relógio
        TR0 = 1;
    }
}

```

```

// Interrupção externa 1
void i1_irq(void) interrupt 2
{
    if (TR0 == 1) {
        // Pausa o relógio
        TR0 = 0;
        // Transfere o tempo do timer para a variável time em segundos
        time = (TH0<<8) + TL0;
        time = time/frequencia;
        // Calcula a velocidade do veículo
        speed = abs(distance/time);
        // Como serão dois displays, velocidades acima de 99 serão mostradas com 99mm/s no mesmo
        if(speed > 99)
            speed = 99;
        // Concatena o valor da velocidade na sua respectiva mensagem
        aux = abs(speed/10);
        msgv[11] = (aux + 48);
        msgv[12] = ((speed - aux*10) + 48);
        // Converte para BCD
        speed = speed + aux*6;
        P1 = speed;
        // Envia mensagem com o valor da velocidade pela serial
        trans_velocidade();
        // Tolerância de 10%
        // Acima de 44mm/s dispara a câmera e aciona leds
        if(speed > 44) {
            P2_2 = 0x01;
            P2_0 = 0x01;
            delay();
            P2_2 = 0x00;
            P2_0 = 0x00;
        }
        // Delay para mostrar velocidade no display
        else
            delay();
        // Display em alta impedância
        P1 = 0xff;
    }
}

// transmite caractere pela serial
void rx_serial(char c) {
    // Config timer 1
    TR1 = 0;
    TH1 = 0XFD;
    TL1 = 0;
    TR1 = 1;
    TI = 0;
    SBUF = c;
}

```

```

    while(!TI);
}

//Transmite mensagem pela serial
void trans_serial(char msg[]) {
    aux = 0;
    while(msg[aux]!= NULL) {
        rx_serial(msg[aux]);
        aux += 1;
    }
}

// Programa principal
int main()
{
    // Habilita interrupção externa 0 e externa 1
    IE = 0x85;
    // Configuração Serial
    PCON = 0x80;
    SCON = 0x40;
    // Habilita rampa da interrupção 0 e 1
    TCON = 0x07;
    TR0 = 0;
    // Habilita o timer 0 para o modo 1 e o timer 1 para o modo 2
    TMOD = 0x25;
    // Zera P2 - Porta para disparo da camera, led de infração e led de funcionamento
    P2 = 0;
    // Inicializa as variáveis
    distance = 100;
    frequencia = 400;
    // Loop
    for(;;) {
        // Led de funcionamento
        P2_1 ^= 0x01;
        // Mensagem "Status OK" pela serial
        trans_serial(msg);
        // Delay para novo ciclo
        delay();
    }
    return 0;
}

```

## 5.8. Módulo de disparo e câmera

Para o conjunto de registro de imagem, uma câmera fotográfica foi adaptada e é usada para captar a imagem dos veículos que ultrapassam a velocidade permitida. Após desmontar parcialmente a

câmera identificou-se que o mesmo é acionado através que uma passagem de corrente através de uma borracha condutiva, assim desenvolveu-se um circuito simples com CD4066, devido ao tempo de resposta do circuito. Com esse módulo de acionamento o envio do sinal para a câmera é quase de imediato, evitando um *delay* muito alto, o que na atuação real seria totalmente inviável.

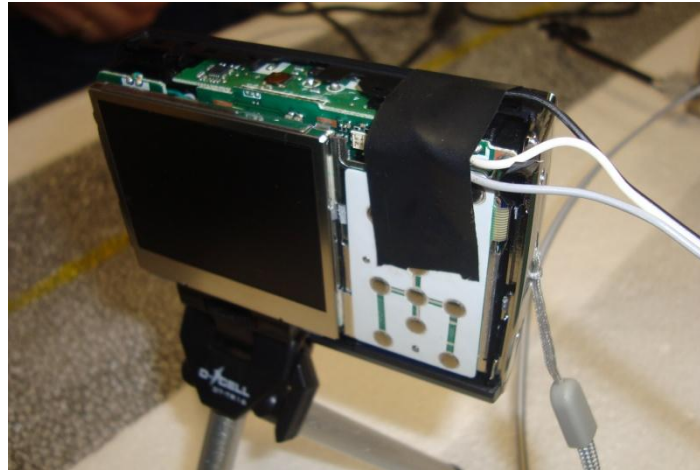


Figura 11: Acoplamento câmera

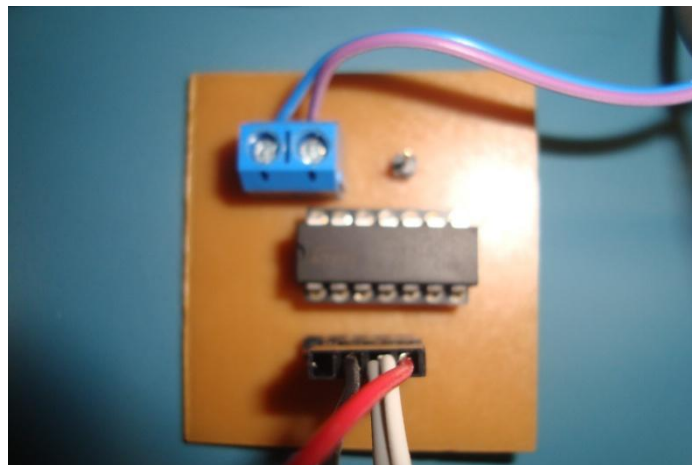


Figura 12: Circuito disparo rápido

## 5.9. Código Visual Basic 6

Um gráfico desenvolvido em Visual Basic 6 é responsável por receber as informações transmitidas via comunicação serial. A interface

gráfica “Radar.exe” monitora em tempo real todos os veículos que passam pela pista, todos os registros são armazenados na tela do aplicativo com suas datas e horários. Não foi possível implementar uma base que registrasse a imagem captada, pois essa possui um delay alto. Logo para consultar os veículos que ultrapassaram o limite, os mesmos deveriam ser consultados direto na memória interna da câmera digital.

A mesma interface permite ao usuário configurar a porta (COM1), velocidade (9600), bits de dados (8), paridade (n), bit de parada (1) e timer (1000 - Indica a velocidade com que ira mostrar registro na tela).

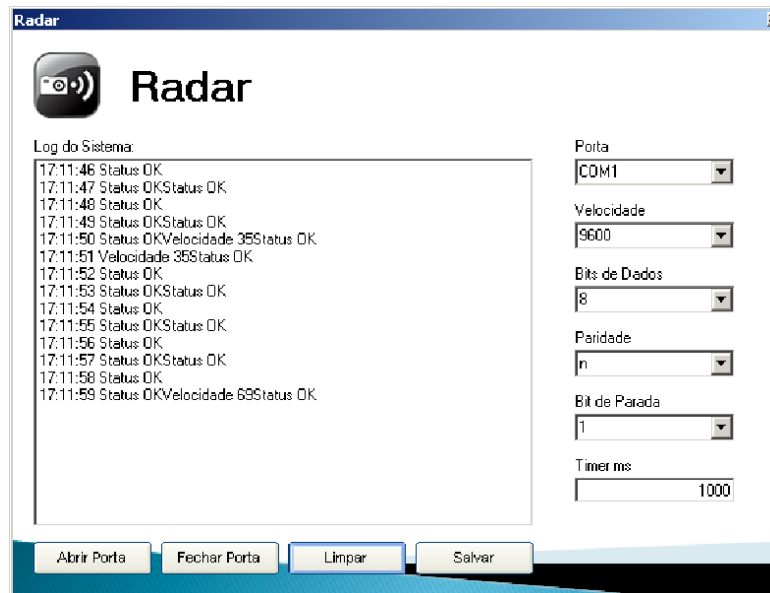
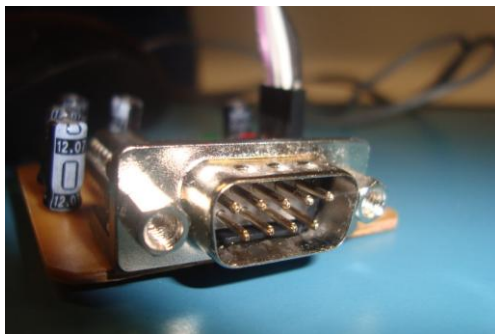


Figura 13: Gráfico de monitoramento.

## 6. Outras Imagens



## 7. Conclusão

**A**ssim, após a conclusão do projeto, chegou a conclusões que através dos conhecimentos pratico ou teórico adquiridos das disciplinas, pode-se desenvolver e implementar um projeto integrado. Além de agregar conhecimentos novos na resolução rápida e pratica de problemas que podem ocorrer em um sistema ou máquina.

Ao longo de todo o processo de desenvolvimento, dificuldades ocorreram em relação às etapas de implementação e integração hardware e software. Informações exatas, como freqüência, para configuração de serial necessitaram de ajustes precisos.

Foi possível programar códigos de controle, não somente em Assembly, em C para gravação em microprocessadores na implementação de diversos recursos em indústrias ou projetos didáticos. Buscaram-se todas as informações teóricas possíveis e necessárias para um bom resultado no termino do mesmo.

## 8. Referencias Bibliográficas

Data sheets

<http://www.datasheetcatalog.com/>

8051

<http://afonsomiguel.com/>

Livros

SILVA JUNIOR, Vidal Pereira. Aplicações praticas do micro controlador 8051. 12<sup>a</sup> edição. Editora Erica. São Paulo. 2004.