

**Centro Federal de Educação Tecnológica de Minas Gerais Engenharia  
Mecatrônica**

---

**Estudo e Elaboração de uma Rede CAN para Aplicação em um  
Sistema Automotivo**

Paulo Vitor de Souza

Orientador: Christian Herrera

Divinópolis

2013

**Centro Federal de Educação Tecnológica de Minas Gerais Engenharia  
Mecatrônica**

---

**Estudo e Elaboração de uma Rede CAN para Aplicação em um  
Sistema Automotivo**

Paulo Vitor de Souza

Monografia referente à disciplina de TCCII do 10º período do Curso de Engenharia Mecatrônica do Centro Federal de Educação Tecnológica de Minas Gerais. Orientado pelo professor Christian Herrera. Realizado pelo aluno Paulo Vitor de Souza.

Divinópolis, março de 2013

## **Resumo**

Este projeto visou à elaboração de módulos didáticos de comunicação CAN para sensoriamento, e acionamento de um sistema automotivo. Este trabalho inicia-se com uma breve introdução ao sistema eletrônico automotivo. Logo após, compara-se um sistema descentralizado conectado em rede e um sistema centralizado autônomo. Em seguida, explicam-se as características do microcontrolador PIC18F4580 e do software LabView. Posteriormente é feita uma revisão bibliográfica sobre os seguintes tópicos: Rede CAN e suas normas. Já no desenvolvimento, são detalhados os projetos de hardware, software e rede. Dentro destes projetos podemos destacar alguns pontos importantes: No projeto de hardware destaca-se o dimensionamento de componentes, no projeto de software destaca-se a definição das mensagens, e no projeto de rede destaca-se a topologia lógica de comunicação e o dimensionamento do barramento. Ao final do desenvolvimento, estes projetos são integrados e formam o sistema proposto com aplicação em um veículo baja. Por fim, este trabalho traz a análise de dados juntamente com a conclusão.

Palavras-chaves: Rede CAN, LabView, Microcontrolador.

## **Abstract**

This project aims at developing educational modules CAN communication for sensing and actuation system of an automobile. This paper begins with a brief introduction to automotive electronics. Soon after, compares a decentralized networked standalone and a centralized system. Then explain the characteristics the microcontroller PIC18F4580 and LabView software. Later a literature review is made on the following topics: CAN Network and its norms. Already in development, the projects are detailed hardware, software and network. Within these projects we highlight some important points: In hardware design highlights the design of components, the design of software highlights the definition of messages, and network design highlights the logical communication topology and sizing bus . At the end of development, these projects are integrated and form the proposed system with application in a vehicle baja. Finally, this paper presents data analysis with the conclusion.

Keywords: CAN Network, LabView, Microcontroller.

## Sumário

1.	Introdução	9
2.	Revisão Bibliográfica	12
2.1	ARQUITETURA CENTRALIZADA	12
2.2	ARQUITETURA DESCENTRALIZADA	12
2.3	SISTEMA ELETRÔNICO AUTOMOTIVO	13
2.4	COMUNICAÇÃO CAN	16
3.	Projeto de Hardware	18
3.1	IDEALIZAÇÃO DO CIRCUITO	18
3.2	ESPECIFICAÇÃO DOS COMPONENTES	19
3.3	CONSTRUÇÃO DOS MÓDULOS	24
3.4	TESTES	34
4.	Projeto de rede	38
4.1	TOPOLOGIA DA REDE	38
4.2	BARRAMENTO	39
4.3	PROJETO FÍSICO DA REDE	44
4.4	TOPOLOGIA LÓGICA DE REDE	46
4.5	ESTRATÉGIA DE TROCA DE INFORMAÇÕES	47
4.6	PROJETO LÓGICO DA REDE	47
5.	Projeto de Software	49
5.1	MICROCONTROLADORES	49
5.2	LABVIEW	52
6.	Conclusões	55
	Referências Bibliográficas	56

## Lista de Figuras

Figura 1: ECU com arquitetura centralizada. ....	12
Figura 2: Arquitetura Descentralizada.....	13
Figura 3: Modelo OSI da ISO.....	16
Figura 4: Sinal diferencial.....	17
Figura 5: Data frame.....	17
Figura 6: Primeira idealização de um módulo genérico.....	19
Figura 7: LT3682 .....	20
Figura 8: PIC18F4580 .....	22
Figura 9: Transceiver CAN MCP2551 .....	22
Figura 10: Conector de 9 vias com trava e cabo.....	23
Figura 11: Conector de 16 vias com trava e cabo.....	23
Figura 12: Conector de 4vias com trava e cabo.....	23
Figura 13: Circuito de Alimentação.....	24
Figura 14: Circuito do microcontrolador.....	25
Figura 15: Circuito de condicionamento CAN.....	26
Figura 16: Slope control através de R3 .....	26
Figura 17: Diagrama elétrico do display.....	27
Figura 18: Sensor de temperatura MTE4053.....	27
Figura 19: Diagrama elétrico para ligação do sensor de temperatura.....	29
Figura 20: Sensor de velocidade SD7063.....	29
Figura 21: Circuito interno do sensor SD7063 .....	30
Figura 22: Condicionamento da onda quadrada.....	30
Figura 23: Montagem de dois módulos em protoboard.....	31
Figura 24: Montagem em funcionamento.....	31
Figura 25: Interface do Ares.....	32
Figura 26: Layout PCB módulo 1.....	33
Figura 27: Layout PCB módulo2.....	33
Figura 28: Módulo 1.....	33
Figura 29: Módulo 2.....	34
Figura 30: Fonte de Alimentação.....	34
Figura 31: Placa PCI-8513.....	35
Figura 32: Barramento CAN.....	35

Figura 33: Computador.....	35
Figura 34: Bus monitor.....	36
Figura 35: Bus monitor.....	36
Figura 36: Frame CAN.....	37
Figura 37: Topologias físicas de rede.....	38
Figura 38: Velocidade de comunicação vs comprimento do barramento. ....	40
Figura 39: Propagação de atraso no bit. ....	40
Figura 40: Definição de bit time e sincronização.....	41
Figura 41: Modelo de queda de tensão no barramento.....	42
Figura 42: Sem resistor de terminação. ....	42
Figura 43: Com resistor de 60Ω.....	43
Figura 44: Com resistor de 120Ω.....	43
Figura 45: Vista lateral da gaiola do mini baja.....	44
Figura 46: Vista isométrica da gaiola do mini baja.....	44
Figura 47: Rede CAN dimensionada para o mini baja.....	46
Figura 48: Topologia Lógica. ....	46
Figura 49: Modelo Cliente/Servidor.....	47
Figura 50: Modelo Produtor/Consumidor. ....	47
Figura 51: Topologia lógica projetada para o mini baja.....	48
Figura 52: Ajuste exponencial para o MTE4053.....	50
Figura 53: Painel Frontal LabView.....	53
Figura 54: Diagrama de blocos LabView.....	54

## **Lista de Tabelas**

Tabela 1: Volume de publicações sobre Eletrônica de Potência Automotiva

Tabela 2: Relação de sensores e atuadores no powertrain.

Tabela 3: Relação de sensores e atuadores no chassi.

Tabela 4: Conversores buck para aplicações automotivas com seus respectivos fabricantes.

Tabela 5: Microcontroladores com comunicação CAN e seus respectivos fabricantes.

Tabela 6: Valores de resistência de acordo à temperatura.

Tabela 7: Tabela comparativa entre módulos.

Tabela 8: Valores típicos para a Comunicação CAN.

Tabela 9: Parâmetros de construção da rede.

# 1. Introdução

Este trabalho de conclusão de curso de Engenharia Mecatrônica começou quando o professor Christian Herrera apresentou alguns artigos sobre Eletrônica de Potência. Os mesmos continham pesquisas que mostram várias mudanças na arquitetura elétrica dos automóveis. Dentre estas modificações propostas ressaltam-se duas essenciais: criação de um barramento de alimentação e criação de um barramento de comunicação. E destas modificações propostas surgiu o interesse de aplicá-las em um veículo baja juntamente com a equipe do CEFET-MG campus V.

Com início das pesquisas surgiram algumas questões, sendo elas:

- Quais as vantagens e desvantagens de um sistema descentralizado utilizando rede CAN quando comparado com um sistema autônomo centralizado?
- Quais as mudanças que devem ser realizadas na arquitetura atual para que se possa desenvolver um barramento de alimentação?
- Quais os benefícios de ter uma alimentação por barramento em relação à alimentação por conexões ponto a ponto?

Avançando um pouco mais no trabalho, notou-se que se fosse desenvolvido as mudanças propostas alguns resultados poderiam ser obtidos, sendo estes:

- Devido à nova arquitetura, haverá redução de peso no baja.
- Com as soluções adotadas, os veículos exigirão menos conexões.
- Desenvolvendo-se os barramentos de comunicação e alimentação, diminuirá a complexidade na montagem e manutenção do sistema elétrico do baja.
- Com o desenvolvimento do barramento de alimentação, será necessária a utilização de componentes mais robustos.

Para responder as questões levantadas e obter os resultados esperados precisaram-se traçar alguns objetivos. Como objetivo principal adotou-se projetar uma arquitetura automotiva descentralizada e alimentada através de barramento e comparar com a arquitetura centralizada alimentada com conexões ponto a ponto, ambas idealizadas para o baja. Além deste objetivo, escolheu-se também estudar o protocolo de comunicação CAN, aprender sobre sistemas elétricos automotivos, estudar eletrônica de potência, construir módulos simples e didáticos que se comunicam via CAN, e testar e validar as soluções propostas.

Uma vez com o projeto está fechado, o mesmo torna-se relevante para o CEFET-MG campus V enquanto pesquisa pelos seguintes aspectos: Esta proposta vem para suprir a necessidade da

equipe baja do CEFET-MG campus V, o trabalho tem relevância acadêmica, e utiliza soluções largamente utilizadas na indústria.

Como o projeto do baja está em desenvolvimento, e ainda não possui nenhuma arquitetura elétrica, este trabalho vem para preencher esta lacuna no projeto. Já a relevância acadêmica deste trabalho, consiste no fato da introdução do barramento de alimentação no veículo. O qual é uma solução que ainda não foi empregada na indústria automobilística, porém é uma área aonde tem grande concentração de pesquisa como mostra a tabela 1:

**Tabela 1: Volume de publicações sobre Eletrônica de Potência Automotiva**

Publicações sobre “Eletrônica de Potência Automotiva”	
Ano	Número de Publicações
2008	322
2009	413
2010	488
2011	532
2012	288

Fonte: IEEEEXPLORE, 2012.

A tabela 1 mostra o volume de publicações de 2008 a 2012 nessa área. Analisando a tabela 1, nota-se que a pesquisa em Eletrônica de Potência Automotiva é muito relevante. Uma solução que já está padronizada e é utilizada largamente na indústria automobilística é a comunicação CAN, pois tem baixo custo, confiabilidade e segurança.

Este trabalho será realizado dentro das instalações do CEFET-MG campus V, mais especificamente nos laboratórios 315, 319 e no laboratório do galpão de mecânica juntamente com utilização dos computadores e instrumentos presentes nos mesmos. Os principais instrumentos que serão utilizados são: osciloscópio digital, programador de microcontrolador, estação de solda, multímetro e ferramentas em geral.

Inicialmente, foram realizadas pesquisas sobre rede CAN, eletrônica de potência e sistema elétrico automotivo. As quais tiveram como referências principais os seguintes livros: “Understanding and Using the Controller Area Network Communication Protocol: Theory and Practice, Marco Di Natale, 2012”, “Eletrônica de Potência, José Antenor Pomilio, 2006”, “The Handbook of Automotive Power Electronics and Motor Drives, Ali Emadi, 2005”, e “Automotive Software Engineering: Principles, Methods, and Tools, Jörg Schäuffele, 2003”.

Após o levantamento teórico, iniciou o projeto da arquitetura proposta. A qual consistiu nas etapas de projeto de hardware, projeto de software, e projeto de rede.

Essas etapas por sua vez, foram divididas em sub-etapas. O projeto de hardware foi dividido da seguinte forma: Dimensionamento de sensores; Dimensionamentos de componentes; e Confeção dos módulos. O projeto de software contou com as seguintes sub-etapas: Definição das mensagens CAN, e Tipo de comunicação lógica. Por fim, o projeto de rede foi executado da seguinte maneira: Projeto do barramento, Topologia da rede, e Definição da estratégia de acesso ao barramento.

Os módulos foram confeccionados com duas topologias diferentes. O primeiro modelo utilizou um microcontrolador PIC18F4580 juntamente com o software MPLAB, já o segundo modelo utilizou a placa PCI-8513 da National Instruments com o software LabView.

Após a conclusão das etapas de projeto, fez-se a análise de dados que inclui: Estimativa de uma aplicação real para um veículo baja, e comparação a desta proposta com uma arquitetura autônoma centralizada.

## 2. Revisão Bibliográfica

### 2.1 ARQUITETURA CENTRALIZADA

Schäuffele (2005) define esta organização como Arquitetura Autônoma Centralizada. Este tipo de organização caracteriza-se pela convergência do fluxo de informação para um único núcleo. A figura 1 ilustra o conceito:

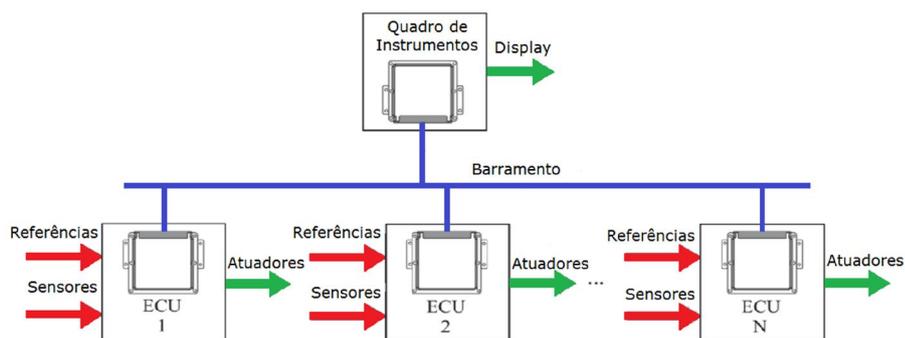


Figura 1: ECU com arquitetura centralizada. Fonte: Próprio autor.

Com a ocorrência deste fato, é fácil perceber que as tarefas executadas pelo processador serão realizadas de forma quase paralela (SCHÄUFFELE, 2005), sendo este um resultado oriundo da execução sequencial das instruções no microcontrolador. A arquitetura centralizada é mais simples de ser desenvolvida, pois não há comunicação com outras centrais, sendo assim, não necessita de nenhum protocolo de comunicação. Porém exige uma montagem complexa devido às várias conexões presentes no sistema.

### 2.2 ARQUITETURA DESCENTRALIZADA

“A ideia de conectar em rede está além do periférico físico no veículo ela habilita uma nova classe de funções, como por exemplo, driver assistance systems” (SCHÄUFFELE, 2005). Este tipo de estrutura é a ideal para os veículos que objetivam segurança, alto desempenho, fácil montagem, e fácil manutenção. A arquitetura descentralizada só tornou-se possível com a inclusão de uma rede de comunicação no sistema automotivo. Atualmente a rede de comunicação mais utilizada em aplicações automotivas é a rede CAN (Controller Area Network). A figura 2 mostra a topologia deste tipo de rede.



**Figura 2: Arquitetura Descentralizada. Fonte: Figura adaptada (SCHÄUFFELE, 2005)**

Como podemos notar as ECUs trocam informações através do barramento de comunicação, sendo assim, qualquer ECU pode ter acesso a qualquer leitura de sensor sem que o mesmo esteja conectado a nela. Com isto, são notórias as vantagens: Diminuição do cabeamento no chicote, montagem modular, aumento de funcionalidades no veículo e maior integração do projeto (EMADI, 2005; SCHÄUFFELE, 2005). Em contra partida, é necessário a inclusão de um protocolo de comunicação e o software torna-se mais complexo.

## 2.3 SISTEMA ELETRÔNICO AUTOMOTIVO

O sistema eletrônico automotivo está distribuído em todo o veículo, monitorando muitas variáveis e controlando várias ações. Para ter uma visão geral do quanto a eletrônica está presente no veículo, inicialmente dividiu-se o mesmo em subsistemas: Powertrain, Chassi, Body e Multimídia. Posteriormente pesquisou-se sobre os sensores em cada subsistema.

### 2.3.1 Powertrain

Esse subsistema engloba o motor, embreagem e transmissão manual ou transmissão automática, caixa de transmissão e sistemas de ignição auxiliares, como motor de arranque e alternador (SCHÄUFFELE, 2005). A tabela 1 mostra a relação de sensores e atuadores deste subsistema:

**Tabela 2: Relação de sensores e atuadores no powertrain.**

Sensores	Atuadores
Posição da válvula de aceleração.	Vela de ignição.
Tensão da bateria.	Válvula de controle eletrônico.
Temperatura do ar de admissão.	Injetores de combustível.
Temperatura do motor.	Relé da bomba de alimentação.
Intensidade de batida.	Ventilação do tanque de combustível.

Velocidade do virabrequim. Velocidade do veículo. Controle de árvore de cames.	
--	--

Como resultado deste monitoramento, o sistema eletrônico está presente no subsistema de powertrain da seguinte forma:

- ECUs do motor.
- ECUs da transmissão.

### 2.3.2 Chassi

Neste subsistema está incluso os seguintes componentes: eixos, rodas, freios, suspensão e amortecedores, e sistemas de direção (SCHÄUFFELE, 2005). A tabela 2 mostra a relação de sensores e atuadores deste subsistema:

**Tabela 3: Relação de sensores e atuadores no chassi.**

Sensores	Atuadores
Tensão da bateria.	Válvulas solenoides das rodas.
Tensão do motor da bomba.	Relé da bomba.
Tensão do relé da válvula.	Relé da válvula.
Sensores de velocidade das rodas.	
Chave da luz de freio.	

Como consequência, o sistema eletrônico está presente no subsistema de chassi da seguinte maneira:

- ABS.
- Distribuição de força-frenagem eletrônica.
- Programa de Estabilidade Eletrônica.
- Freio de mão.
- Sistema de monitoramento das pressões dos pneus.
- Suspensão pneumática.
- Direção assistida.
- Direção ativa.
- Freio eletro hidráulico ou eletromecânico.

### **2.3.3 Body**

Os sistemas eletrônicos inclusos no subsistema de body são divididos em segurança passiva e conforto (SCHÄUFFELE, 2005). O grupo de segurança passiva aplica-se a todos os sistemas que contribuem para o aumento da segurança e proteção dos ocupantes em caso de acidentes (SCHÄUFFELE, 2005). Estes sistemas incluem:

- Sistemas de retenção como tensores de cintos de segurança.
- ECU do Airbag, incluindo o banco de detecção de ocupante.

Já o grupo de conforto inclui os sistemas de acesso ao veículo, incluindo sistemas de travas, sistema de alarme antifurto, bem como os sistemas que controlam as seguintes funções:

- Vidro elétrico.
- Teto solar.
- Capota conversível.
- Limpadores de para-brisa e sensores de chuva.
- Ajuste da coluna de direção.
- Aquecimento interior e ar condicionado.
- Iluminação interna.
- Controle de faróis e sistema de limpeza dos faróis.
- Recursos de ajuda ao estacionamento.

### **2.3.4 Multimídia**

O grupo de multimídia é relativamente recente, porém é um sistema que teve grande aceitação. O sistema eletrônico está presente no subsistema de multimídia da seguinte forma:

- Amplificadores e sistemas de áudio.
- Sistemas de vídeo.
- Sistemas de navegação.
- Comando de voz.
- Acesso à internet.

## 2.4 COMUNICAÇÃO CAN

### 2.4.1 História

A rede CAN foi desenvolvida em meados dos anos 80 pela Bosch GmbH para proporcionar uma boa relação custo eficácia para um barramento de comunicação com aplicações automotivas (NATALE, 2012). Hoje esta comunicação é utilizada também em indústrias, plantas de controle, robótica, e em dispositivos médicos. Natale (2012) define Controller Area Network (CAN) como um protocolo de comunicação baseado em múltiplos mestres caracterizado pela resolução determinística da conexão, baixo custo, e implantação simples.

Segundo Schäuffele (2005) a introdução desta tecnologia em veículos iniciou-se em 1990, e está presente até hoje em veículos que requerem alto desempenho.

A rede CAN é um barramento digital projetado para operar de 20 Kbit/s até 1 Mbit/s, padronizado pela norma ISO11898 para altas velocidades (500 Kbit/s) e pela norma ISO11519-2 para aplicações de baixas velocidades (até 125 Kbit/s). Sendo assim, a norma ISO11898 define um modelo de camadas para o protocolo CAN e um modelo de arquitetura para aplicações em alta velocidade. A figura 3 mostra a organização em camadas do protocolo CAN:

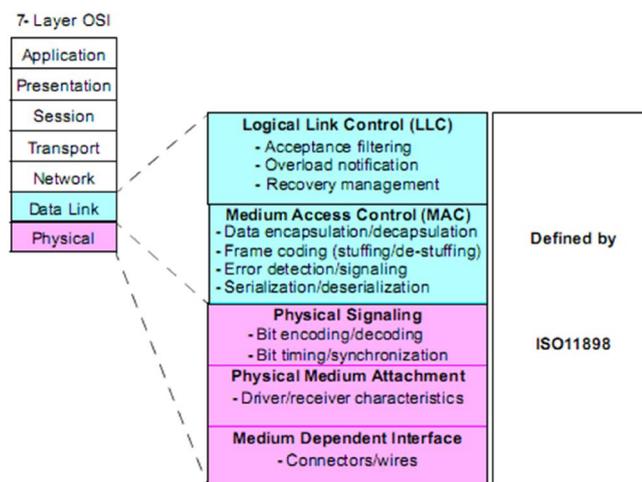
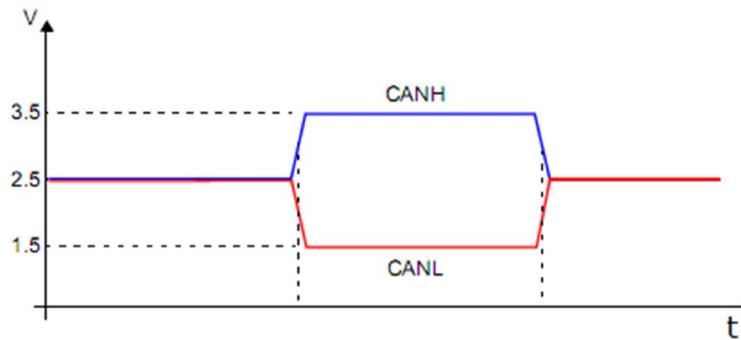


Figura 3: Modelo OSI da ISO. Fonte: (RICHARDS, 2002).

Na figura 3 nota-se que a norma define somente as primeiras camadas: camada física, e camada de enlace de dados. Definindo a camada física, a norma aborda alguns temas como: características do transceiver, meio físico de transmissão, e o sinal transmitido. Por sua vez, a figura 4 mostra o sinal elétrico no barramento de acordo com a norma, um sinal diferencial com duas componentes CANH e CANL.



**Figura 4: Sinal diferencial. Fonte: (RICHARDS, 2002).**

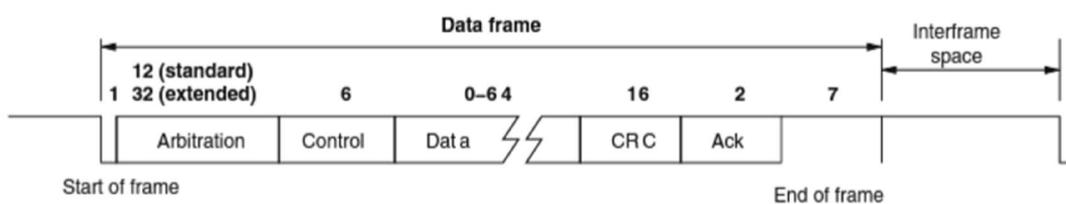
Por outro lado, definindo a camada de enlace de dados a norma engloba alguns aspectos como: frame da mensagem, serialização de dados, e detecção de erros.

### 2.4.2 Frame CAN

Segundo Natale (2012), existem quatro tipos de frames de mensagens no protocolo CAN definidos por conteúdo e função:

- Data frame. É o tipo de mensagem que contém dados de informações de um nó para outro(s).
- Remote frame. É a mensagem utilizada para pedir uma transmissão do correspondente data frame.
- Error frame. Mensagem transmitida sempre que um nó detecta um erro na rede.
- Overload frame. Tipo de mensagem utilizada para controle de fluxo para pedir um tempo adicional de atraso antes de transmissão do data frame e remote frame.

Na imagem 5 tem-se um frame completo caracterizando um data frame:



**Figura 5: Data frame. Fonte: (NATALE, 2012).**

Nota-se na imagem 5 temos campos definidos dentro do frame, sendo eles: Arbitration, Control, Data, CRC, e Ack.

### 3. Projeto de Hardware

Optou-se por incluir um projeto de hardware neste trabalho pelo fato do funcionamento correto dos módulos ao final deste trabalho estar diretamente ligado ao sucesso desta etapa. Seguindo nesta vertente, alguns aspectos foram priorizados, sendo estes: funcionamento correto dos módulos, segurança, e custo. O primeiro aspecto é o mais importante, pois o mesmo define o sucesso do projeto. Já o segundo aspecto, refere-se ao tipo de aplicação a qual o módulo será empregado. E por fim, o terceiro parâmetro ressalta-se pela importância de ser eficaz com os recursos disponíveis. Sendo assim, trabalhando com este foco, os objetivos deste projeto foram:

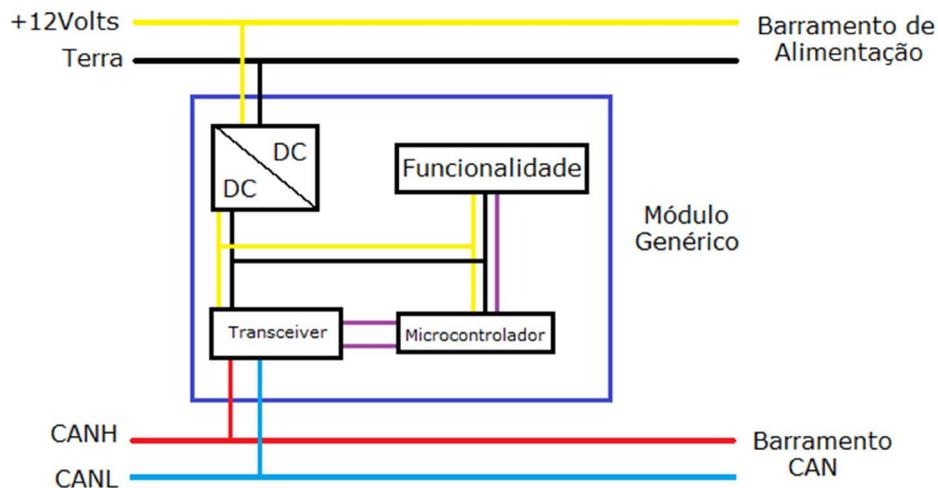
- Idealizar o circuito.
- Especificar dos componentes.
- Desenhar o layout PCB.
- Construir os módulos.
- Testar os módulos.

#### 3.1 IDEALIZAÇÃO DO CIRCUITO

Para idealizar o circuito partiu-se das informações iniciais especificadas no projeto, entre elas:

- Uso de microcontrolador.
- Uso de transceiver CAN.
- Uso de conversores DC-DC.
- Todos os módulos devem conter conectores para o barramento de alimentação e comunicação e funcionalidade.
- Cada módulo tem uma funcionalidade específica como:
  - Monitorar eventos.
  - Interface.
  - Mensurar variáveis.
  - Acionar cargas.
- Todos os módulos são alimentados com 12 Volts.
- Todos os módulos comunicam-se via rede CAN.
- Todos os módulos devem conter proteção à sobretensão.
- Todos os módulos devem conter proteção à subtensão.
- Todos os módulos devem conter proteção à polaridade invertida.

De posse de todas essas informações, a primeira visão do que seria um módulo proposto neste projeto é mostrado na figura 6:



**Figura 6: Primeira idealização de um módulo genérico. Fonte: Próprio autor.**

Na imagem 1, nota-se um módulo genérico alimentado com 12 Volts, um conversor buck abaixando a tensão para 5 Volts, um microcontrolador sendo o núcleo de processamento, um transceiver CAN condicionando os sinais da comunicação segundo a norma ISO 11898, e também foi atribuída ao módulo um funcionalidade conforme descrito acima. Esta última pode receber dados do microcontrolador ou enviar algum sinal analógico para o mesmo.

### 3.2 ESPECIFICAÇÃO DOS COMPONENTES

Para especificar os componentes, partiu-se do princípio que o módulo teria que suportar o ambiente automotivo. Sendo este, um ambiente muito hostil contendo as seguintes características:

- Variações de temperaturas.
- Poeira.
- Umidade.
- Vibrações.
- Atmosfera salina.
- Tensão de alimentação poluída.

#### 3.2.1 Conversor DC-DC

O conversor DC-DC tipo buck é um circuito muito usual e existem fabricantes de semicondutores que possuem linhas destes produtos voltados para o setor automotivo. Desta

forma, inicialmente, fez-se uma busca por esses fabricantes com seus respectivos produtos, e o resultado desta busca pode ser visto abaixo, na tabela 4:

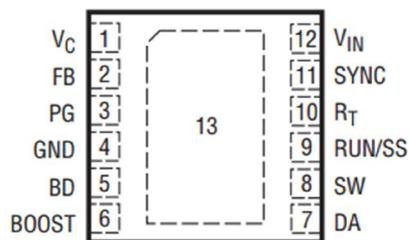
**Tabela 4: Conversores buck para aplicações automotivas com seus respectivos fabricantes.**

Fabricante	Componente
Linear Technology	LT3433
	LT3748
	LTC3115-1
	LT3682
On Semi	NCV890101
	NCV890130
	NCV890131
	NCV890200
	NCV890201
	NCV890230
	NCV890231

Após esta busca escolheu-se o LT3682 por possuir as seguintes características:

- Opera entre 3.6V até 36 V.
- Opera entre as temperaturas de -40°C até 125°C.
- Tem proteção contra transientes de até 60 V.
- Proteção contra curto circuito.
- Tem o menor preço quando comparado com os demais.

A figura 7 mostra a organização dos pinos juntamente com suas funcionalidades presentes no mesmo:



**Figura 7: LT3682. Fonte: (TECHNOLOGY, L., 2008).**

### 3.2.2 Microcontrolador

O microcontrolador é parte fundamental deste projeto, pois este componente será responsável por todo processamento de dados do sistema, tanto aquisições oriundas de sensores quanto troca de informações via rede CAN. Desta forma, realizou-se uma pesquisa por fabricantes de hardwares com comunicação CAN, e o resultado desta busca é mostrado na tabela 5:

**Tabela 5: Microcontroladores com comunicação CAN e seus respectivos fabricantes.**

Fabricante	Componente
Atmel	AT90CAN32
	ATmega32C1
	ATmega16M1
Microchip	PIC18F2480
	PIC18F2580
	PIC18F4480
	PIC18F4580

Ao final desta pesquisa notou-se que os componentes da Atmel são de fato voltados para o setor automotivo e em alguns casos são superiores aos componentes da Microchip. Entretanto, não optou-se pelos produtos da Atmel pela dificuldade de compra, e também pelo fato do CEFET-MG não possuir as ferramentas necessárias para trabalhar e desenvolver aplicações com este hardware. Portanto, escolheu-se o PIC18F4580 pelos seguintes motivos:

- Possui comunicação CAN.
- Opera entre as temperaturas de  $-40^{\circ}\text{C}$  até  $125^{\circ}\text{C}$ .
- Possui baixo consumo de potência.
- Possui 11 conversores A/D de 10-bit.
- Possui 4 timers.
- Opera em até 40MHz.
- A instituição possui as ferramentas necessárias para trabalhar e desenvolver aplicações.

A figura 8 mostra a organização dos pinos juntamente com suas funcionalidades presentes no mesmo:

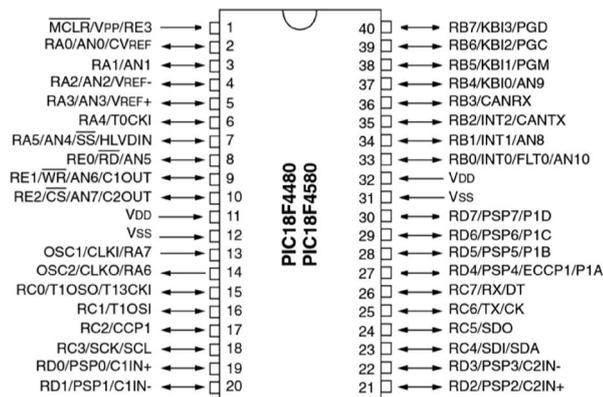


Figura 8: PIC18F4580. Fonte: (TECHNOLOGY, M., 2006).

### 3.2.3 Transceiver

O transceiver tem um papel muito importante na comunicação CAN, pois é este elemento que converte os bits vindos do microcontrolador em um sinal diferencial para o barramento e vice-versa. Então, para o bom funcionamento da comunicação CAN é necessário ter um transceiver de qualidade e compatível com o microcontrolador PIC18F4580. Sendo assim, buscou-se um transceiver da própria Microchip, sendo este o MCP2551. O MCP2551 possui as seguintes características:

- Opera com taxa de transmissão de dados de até 1Mbis/s.
- Opera entre as temperaturas de -40°C até 85°C.
- Até 112 nós podem ser conectados ao barramento.
- Proteção contra transientes de alta tensão.
- Alta imunidade a ruídos vindos do barramento diferencial.
- Atende aos padrões ISO 11898.

A figura 9 mostra a organização dos pinos juntamente com suas funcionalidades presentes no mesmo:

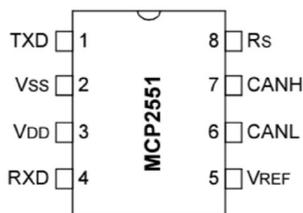


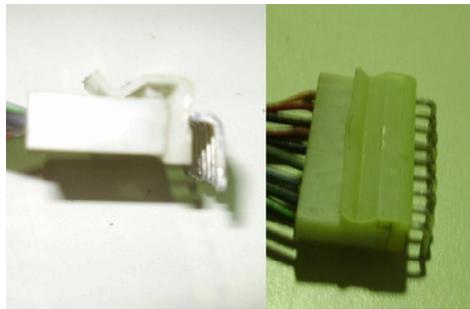
Figura 9: Tranceiver CAN MCP2551. Fonte: (TECHNOLOGY, M., 2003).

### 3.2.4 Conectores

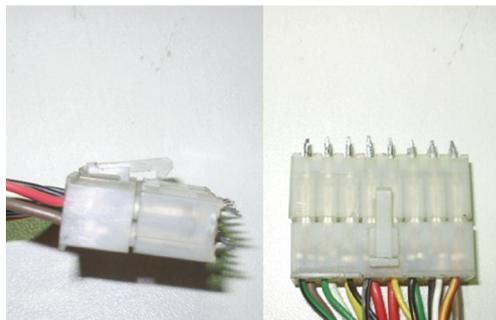
Os conectores são igualmente importantes no projeto. Uma vez mal dimensionados podem causar ruídos aleatórios através de mau contato comprometendo o sucesso da aplicação. O protocolo CAN juntamente com as normas que a regulamentam não definem conectores, porém algumas observações foram feitas em torno deste item, e as conclusões foram as seguintes:

- O conector deve suportar a corrente máxima do módulo 1A.
- O conector deve ter uma geometria de encaixe único.
- O conector deve ter travas suportando as trações exercidas durante a competição.
- Deve-se ter um conector soldado na placa e outro no cabo.

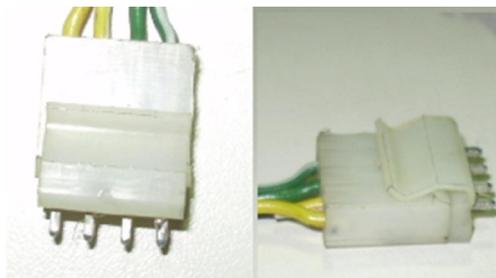
As figuras 10, 11 e 12 mostram os conectores automotivos:



**Figura 10: Conector de 9 vias com trava e cabo. Fonte: Próprio autor.**



**Figura 11: Conector de 16 vias com trava e cabo. Fonte: Próprio autor.**



**Figura 12: Conector de 4 vias com trava e cabo. Fonte: Próprio autor.**

### 3.3 CONSTRUÇÃO DOS MÓDULOS

A construção dos módulos consiste em três sub etapas. A primeira sub etapa é o desenvolvimento de um diagrama elétrico a partir das informações vindas da etapa de especificação. Já a segunda sub etapa, é desenhar um layout PCB (Printed Circuit Board) com base no diagrama elétrico proposto. Por fim, a última sub etapa é a montagem física dos módulos.

#### 3.3.1 Diagramas Elétricos

Para a montagem do diagrama elétrico inicialmente utilizou-se da ideia da figura 6 que é o módulo dividido por funções, sendo elas: alimentação, processamento de dados, condicionamento de sinais para a comunicação, conectores para alguma eventual funcionalidade. Seguindo desta forma, deu-se prioridade à alimentação.

Alimentação:

Como especificado anteriormente, a alimentação utilizaria o LT3682, entretanto houve atraso na liberação do orçamento para a compra dos componentes. Os componentes que não necessitavam de importação não houve problemas. Porém, para este componente, cuja importação seria necessária ocorreu à inviabilização do uso do mesmo por causa do prazo de entrega que seria muito longo, em torno de cinco semanas. Desta forma, resolveu-se perder um pouco em desempenho e adotar uma solução mais simples, porém confiável e de rápida aquisição. A alternativa encontrada foi utilizar o regulador de tensão 7805 juntamente com capacitores filtrando variações de tensões oriundas do barramento. Como estes módulos são para testes e fins didáticos notou-se que esta modificação atenderia as expectativas. A figura 13 mostra esta modificação no projeto junto com os filtros capacitivos, e os leds D1 e D2.

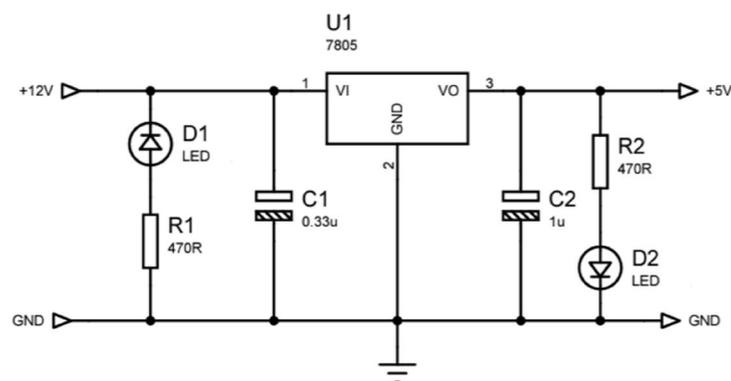


Figura 13: Circuito de Alimentação. Fonte: Próprio autor.

Circuito do microcontrolador:

Neste circuito o principal componente é o PIC18F4580. Contudo, este microcontrolador não funciona sozinho, o mesmo requer alguns circuitos auxiliares, como:

- Circuito oscilador.
- Circuito de reset.
- Filtros de tensão.

O circuito oscilador é o responsável por gerar o clock do sistema, influenciando assim todo o tráfego de dados. O circuito de reset é o circuito utilizado para reiniciar o microcontrolador, não é obrigatório tê-lo no projeto, todavia incluiu-se esta funcionalidade em caso de erro no programa. Outro motivo para esta escolha é fato do pino 1 ter somente as funções de master clear (reset) e ser a referência de tensão de programação. Já os filtros de tensão são capacitores colocados perto dos pinos de alimentação como intuito de diminuir a variação de tensão. A figura 14 mostra o diagrama do circuito do microcontrolador:

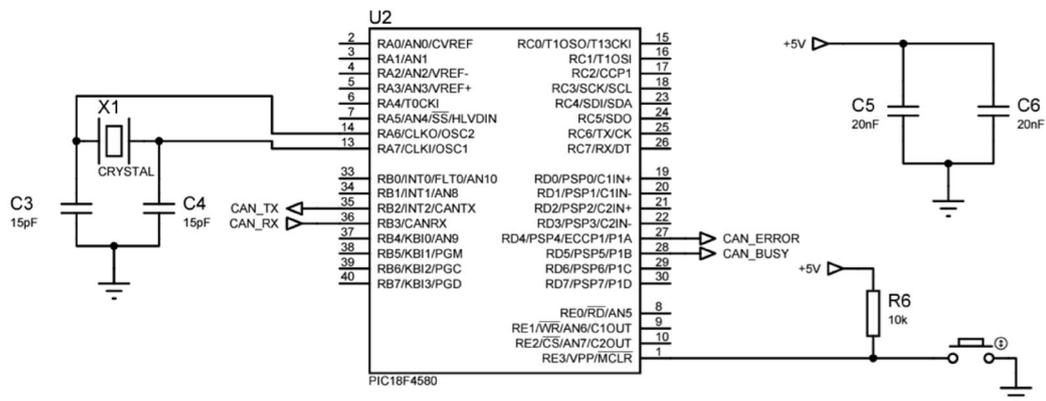


Figura 14: Circuito do microcontrolador. Fonte: Próprio autor.

Como observar-se na figura 14, o circuito oscilador é composto por C3, C4 e X1. Sendo C3 e C4 capacitores cerâmicos de 15 pF e o cristal oscilador X1 de 20 MHz. Já o filtro de tensão é realizado por C5 e C6 que são capacitores de 20nF localizados próximos aos pinos de VCC. E por fim, o circuito de reset é formado por R6 e um botão. Uma vez pressionado o botão causa-se o reset do microcontrolador, caracterizando o reset por estado baixo, ou low state.

Condicionamento de sinais CAN:

Os sinais que caracterizam o protocolo CAN seguem a norma ISO 11898-2 na qual são muito bem detalhados. O transceiver MCP2551 foi desenvolvido baseando-se nesta norma, porém alguns circuitos foram incluídos no projeto para um melhor desempenho, sendo estes:

- Filtro na alimentação.
- Slope control.

- Filtro no barramento.
- Circuito de indicação.

O filtro de alimentação, novamente, foi colocado para impedir variações de tensão evitando o mau funcionamento do dispositivo. O slope control é um meio para diminuir a emissão de EMI limitando o tempo de subida e descida dos sinais CAN\_H e CAN\_L. Já o filtro no barramento foi uma forma de eliminar eventuais ruídos presentes na comunicação. E inclui-se o circuito de indicação como uma forma de acompanhar a comunicação CAN e a execução do programa, visto que não simulou-se esta etapa. O diagrama da comunicação CAN é mostrado na figura 15:

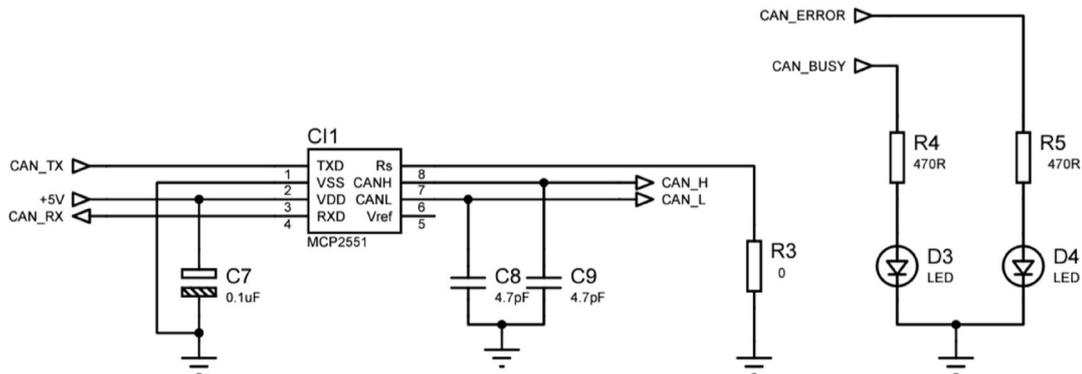


Figura 15: Circuito de condicionamento CAN. Fonte: Próprio autor.

Observa-se na figura 15, que os capacitores C7, C8 e C9 são os filtros capacitivos. Sendo que, C7 é filtro de alimentação, e C8 com C9 são os filtros de comunicação. Ainda na figura 10, o resistor R3 está indicado com valor 0, isso refere-se ao fato de que o pino Rs deve ser aterrado em caso de comunicações a 1Mbit/s. A mudança de valores deste resistor causa o slope control referido acima, a figura 16 mostra um gráfico de como o valor de R3 influencia os tempos de subida e descida dos sinais CAN\_H e CAN\_L:

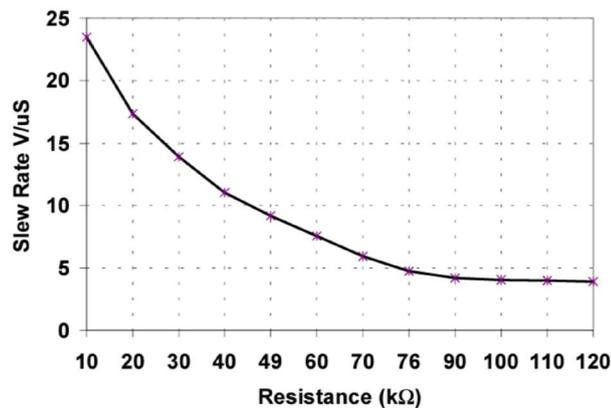


Figura 16: Slope control através de R3. Fonte: (TECHNOLOGY, M., 2003).

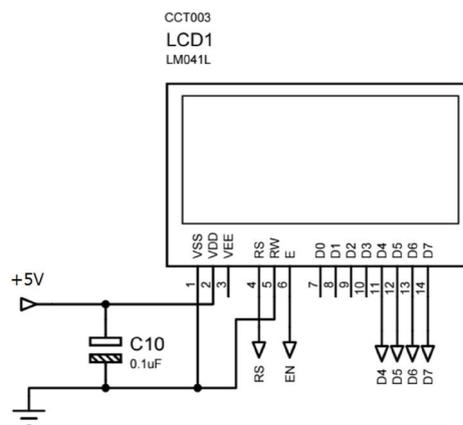
Continuando na figura 15, tem-se ainda os leds D3 e D4 indicando barramento em uso e erro, respectivamente, servindo de referência para correções no software.

Display:

O circuito requerido pelo display é bastante simples, sendo este um filtro de alimentação feito pelo capacitor C10 e um conector de 10 vias. Para que fosse possível ligar o display usando um menor número de vias fizeram-se alguns ajustes, sendo estes:

- Não realiza-se leituras na memória do display.
- Usa-se envio de dados via nibble, ou meio byte.
- Não usa-se controle de contraste.

Desta forma, o diagrama elétrico para o display ficou conforme a figura 17:



**Figura 17: Diagrama elétrico do display. Fonte: Próprio autor.**

A divisão das vias do conector ficou da seguinte forma: 2 vias para alimentação, 2 vias para sinais de controle, 4 vias para dados e 2 vias para o backlight.

Sensor de Temperatura:

O sensor que utilizou-se neste projeto foi o MTE4053 que é um sensor de linha automotiva usado para medir a temperatura de água do motor. Tal sensor é mostrado na figura 18:



**Figura 18: Sensor de temperatura MTE4053. Fonte: Próprio autor.**

Após uma extensa pesquisa devido à falta de informação presente neste setor, conseguiu-se achar um manual com as características deste tipo de sensor, sendo elas:

- Faixa de operação de -40°C até 125°C.
- 5 Volts é máxima tensão de operação.
- Sensor do tipo NTC (Negative Temperature Coefficient)

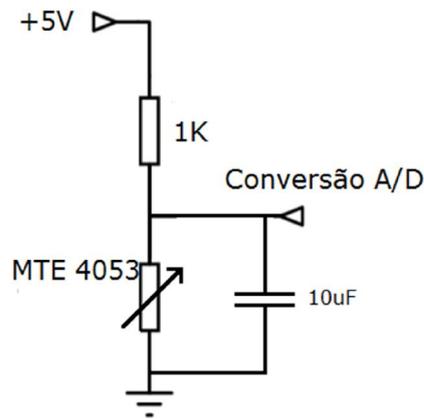
Desta forma, o funcionamento deste tipo de sensor baseia-se na queda exponencial de sua resistência interna de acordo com o aumento da temperatura. A tabela 6 mostra alguns valores de resistências para determinadas temperaturas:

**Tabela 6: Valores de resistência de acordo à temperatura.**

Temperatura (°C)	Resistência (Ω)
-40	100950
-30	53100
-20	29120
-10	16600
0	9750
10	5970
20	3750
25	3000
30	2420
40	1600
50	1080
60	750
70	530
80	380
90	280
100	200
110	150
125	100

**Fonte: (MARELLI, 2001).**

Uma técnica comum para trabalhar com este tipo de sensor é achar uma região linear de operação e construir um circuito para o condicionamento de sinais. Neste projeto fez-se diferente, observou-se que era melhor retirar a complexidade do hardware e aumentar a complexidade do software. Inicialmente, fez-se um divisor de tensão para que o sensor tivesse uma queda de tensão variável e menor que 5 Volts, conforme mostra a figura 19:



**Figura 19: Diagrama elétrico para ligação do sensor de temperatura. Fonte: Próprio autor.**

Já o tratamento do sinal Conversão A/D será comentado mais a frente na seção de Projeto de Software.

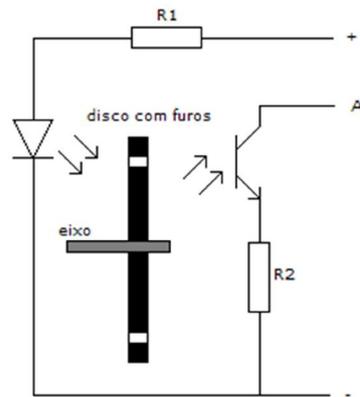
Sensor de Velocidade:

O sensor escolhido para este projeto foi o SD7063 mostrado na figura 20:



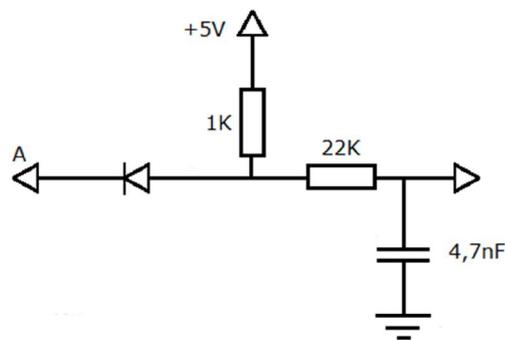
**Figura 20: Sensor de velocidade SD7063. Fonte: Próprio autor.**

Este sensor é fabricado para aplicações em linhas automotivas, e achar informações sobre o mesmo foi muito complicado. O que sabia-se a princípio sobre este sensor era que em uma revolução o mesmo deveria liberar 6 pulsos. Olhando inicialmente a figura 20, supõe-se que o mesmo tem dois pinos de alimentação e uma saída para os pulsos, entretanto isso não ocorre. Após vários testes e pesquisas notou-se que o pino central que supôs-se ser uma saída, era na verdade uma entrada. Então, outra abordagem foi realizada e percebeu-se que o circuito interno do sensor tem o mesmo comportamento da figura 21:



**Figura 21: Circuito interno do sensor SD7063. Fonte: Próprio autor.**

De posse desta informação, desenvolveu-se o circuito para a geração da onda quadrada, pois é a partir deste sinal que torna-se possível estimar a velocidade do veículo, a figura 22 mostra o diagrama elétrico:

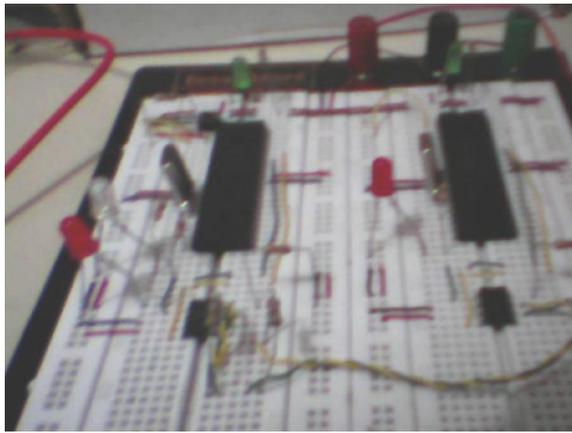


**Figura 22: Condicionamento da onda quadrada. Fonte: Próprio autor.**

Já o tratamento da onda quadrada será explicado posteriormente na seção de Projeto de Software.

### 3.3.2 Layout PCB:

Concluída a sub etapa anterior, que são os diagramas elétricos, já pode-se fazer a montagem de protótipos utilizando protoboards. Este procedimento não é obrigatório e nem sempre é bem sucedido devido aos maus contatos presentes na montagem. Porém, a partir desta montagem já é possível visualizar a localização dos componentes, identificar fontes de erros, verificar se o circuito proposto realmente funciona, e aumenta a confiança para uma montagem definitiva. Neste projeto, dedicou-se um tempo a este tipo de procedimento devido ao fato do circuito não possuir modelos para simulação, a figura 23 mostra a montagem inicial em protoboard:

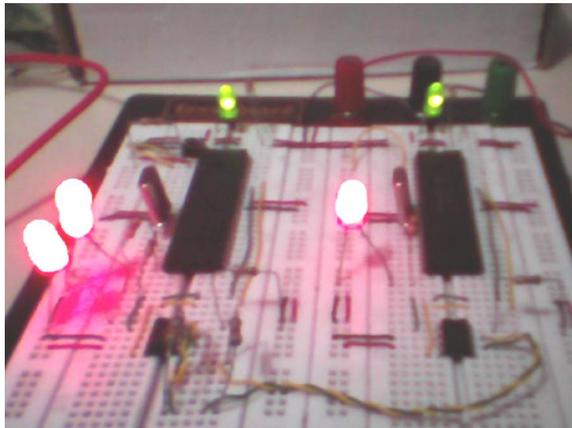


**Figura 23: Montagem de dois módulos em protoboard. Fonte: Próprio autor.**

A partir do protótipo visto na figura 23, fez-se algumas observações:

- Os resistores de terminação realmente influenciam o sinal.
- A bitola do barramento interfere no sinal.
- Os transceivers são sensíveis à descarga estática.
- Os cristais necessitam de filtros.

Apesar de todos os estes problemas ressaltados, a montagem obteve êxito como mostra a figura 24:



**Figura 24: Montagem em funcionamento. Fonte: Próprio autor.**

Nesta montagem objetivou-se o funcionamento da rede CAN, ou seja, os dois microcontroladores teriam somente que comunicar-se entre si. Neste protótipo, o software era bem simples, os microcontroladores liam suas respectivas entradas e acionavam os leds do outro microcontrolador.

Passada esta etapa de prototipação rápida, deu-se início ao desenho do layout PCB. Este layout foi produzido com o uso do software Ares que é uma ferramenta profissional para este tipo de trabalho, a figura 25 mostra a interface deste programa:

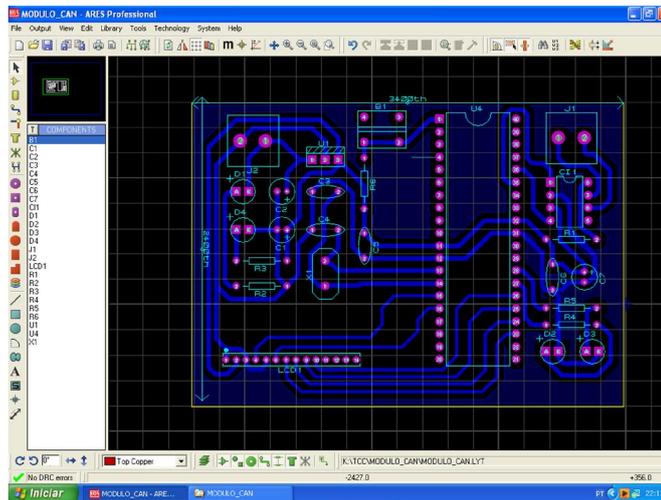


Figura 25: Interface do Ares. Fonte: Próprio autor.

Durante a utilização do Ares alguns ajustes foram realizados, como:

- A precisão do programa foi ajustada para 1th que igual um milésimo de polegada.
- As trilhas são padronizadas em 1mm e 0,5mm ou 40th e 20th, respectivamente.
- Todos os furos presentes no layout tem 1mm de diâmetro.

Feitas estas configurações desenhou-se inicialmente dois layouts distintos. A tabela 7 mostra as características dos dois módulos:

Tabela 7: Tabela comparativa entre módulos.

Características	Módulo 1	Módulo 2
Alimentação	LM317	7805
Microcontrolador	PIC18F4580	PIC18F4580
Transceiver	MCP2551	MCP2551
Número de Botões	8	-
Display	-	1
Conversor A/D	1	-

As figuras 26 e 27 mostram os layouts PCBs destes módulos:

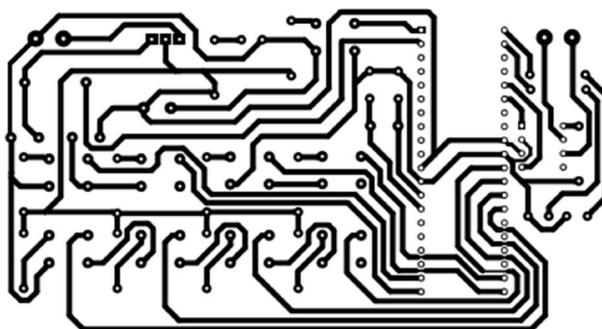


Figura 26: Layout PCB módulo 1. Fonte: Próprio autor.

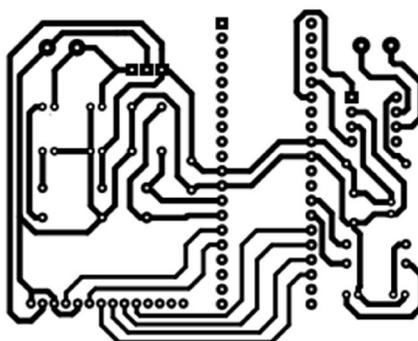


Figura 27: Layout PCB módulo 2. Fonte: Próprio autor.

### 3.3.3 Montagem dos módulos:

A construção dos módulos foi realizada de forma artesanal dentro das instalações do CEFET-MG campus V. Esta montagem artesanal consiste em transferir o layout PCB para uma placa de fenolite com a superfície cobreada, e então corroê-la. As figuras 28 e 29 mostram os módulos depois de prontos:

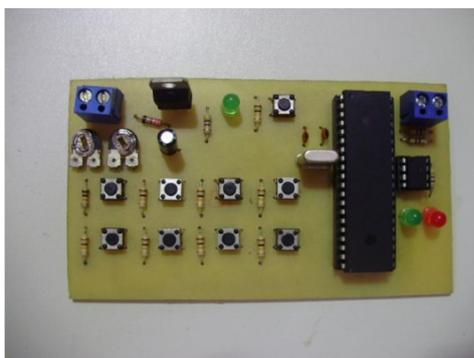
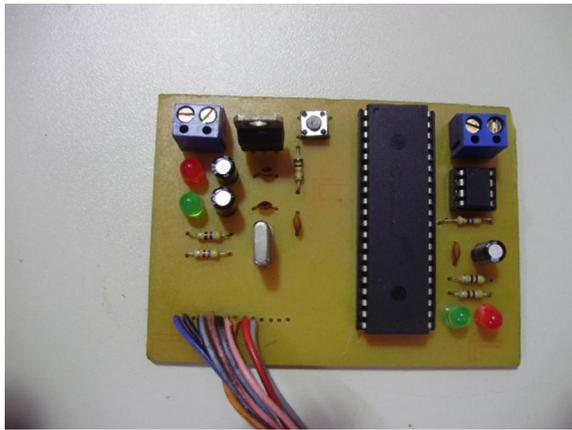


Figura 28: Módulo 1. Fonte: Próprio autor.



**Figura 29: Módulo 2. Fonte: Próprio autor.**

### 3.4 TESTES

A seção de testes é dedicada à identificação de possíveis pontos falhas os quais se por ventura ocorrerem causam a falha imediata do módulo. Após um estudo detalhado do módulo, notou-se três possíveis pontos de falha, sendo estes:

- Alimentação.
- Comunicação CAN.

Para a realização dos testes foram utilizados:

- Uma fonte de alimentação 0 a 30 Volts – 3 Ampères com limitação de corrente.
- Um osciloscópio digital.
- Uma placa PCI-8513.
- Um barramento CAN.
- Um computador.

As figuras de 30 a 33 mostram os equipamentos utilizados:



**Figura 30: Fonte de Alimentação. Fonte: Próprio autor.**



**Figura 31: Placa PCI-8513. Fonte: Próprio autor.**



**Figura 32: Barramento CAN. Fonte: Próprio autor.**



**Figura 33: Computador. Fonte: Próprio autor.**

### **3.4.1 Teste da alimentação**

Neste teste verificou-se as especificações propostas no item idealização do circuito no que refere-se à tensão de entrada. Inicialmente alimentou-se o módulo com tensões de 5 a 30 Volts e verificou-se que a tensão interna do módulo permanecia constante em 5 Volts. Depois, retirou-se a tensão e alimentou-se novamente o módulo com uma tensão inferior a 5 Volts, o módulo não funcionou, como esperado, e quando voltou a ser alimentado com 12 Volts funcionou perfeitamente. Posteriormente realizou-se o teste da tensão invertida notou-se que a corrente que circulou pelo módulo só não foi maior pelo fato da fonte de tensão utilizada limitar a mesma.

Sendo assim, se o módulo estivesse ligado em uma bateria o mesmo teria queimado o regulador de tensão. Então, viu-se a necessidade de incluir o diodo de proteção que não estava presente nos primeiros módulos confeccionados.

### 3.4.2 Teste da comunicação CAN

Neste teste verificaram-se dois aspectos. O primeiro aspecto foi se a mensagem CAN era entendida pelo computador. O segundo aspecto era a forma de onda da mensagem CAN. Para testar se a mensagem era entendida pelo computador, inicialmente, conectaram-se os módulos ao barramento CAN. Depois, geraram-se eventos para que os módulos escrevessem mensagens no barramento. Então, utilizou-se o software Bus Monitor da National Instruments para captar a mensagem como mostram as figuras 34 e 35:

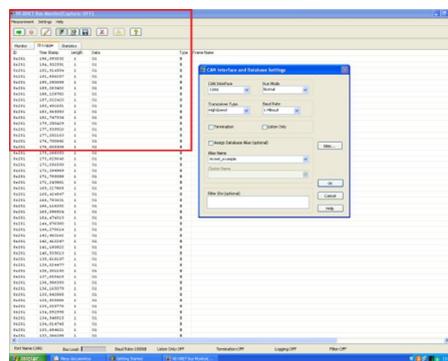


Figura 34: Bus monitor. Fonte: Próprio autor.

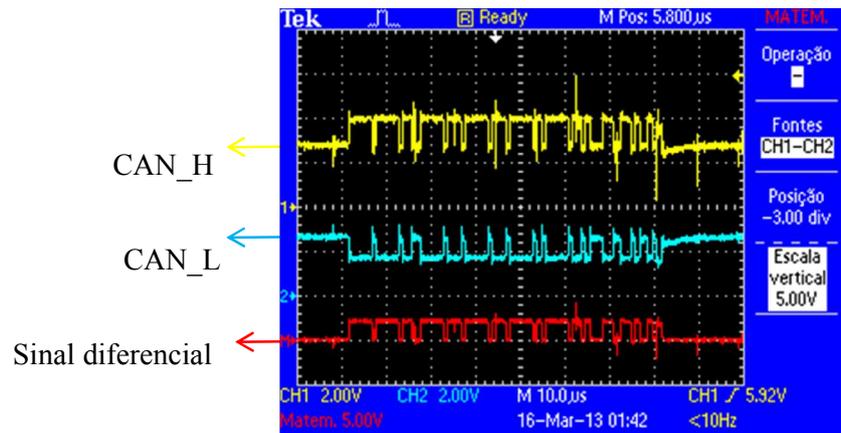
ID	Time Stamp	Length	Data	Type
0x201	196,093032	1	01	E
0x201	194,502591	1	01	E
0x201	192,914584	1	01	E
0x201	191,856287	1	01	E
0x201	190,280688	1	01	E
0x201	189,263402	1	01	E
0x201	188,128782	1	01	E
0x201	187,022420	1	01	E
0x201	185,491691	1	01	E
0x201	183,565083	1	01	E
0x201	181,747334	1	01	E
0x201	179,280429	1	01	E
0x201	177,939926	1	01	E
0x201	177,282163	1	01	E
0x201	176,700842	1	01	E
0x201	176,005308	1	01	E

Figura 35: Bus monitor. Fonte: Próprio autor.

Na figura 35 vemos que o teste foi bem sucedido, pois o software conseguiu identificar todos os frames gerados.

Para monitorar o sinal diferencial, conectaram-se os módulos ao barramento CAN. Depois, geraram-se eventos para que os módulos escrevessem mensagens no barramento. Então, utilizou-

se o osciloscópio juntamente com o software NI LabVIEW SignalExpress Tektronix Edition para captar o frame CAN conforme a figura 36:



**Figura 36: Frame CAN. Fonte: Próprio autor.**

Na figura 36 vemos os dois canais, CAN\_H e CAN\_L, e um canal diferencial que é justamente o sinal resultante no barramento. Sendo assim, sinal da comunicação foi bem sucedido.

## 4. Projeto de rede

O projeto de rede tem como função a integração do sistema, por isso ele completa tanto o projeto de hardware quanto projeto de software. Nota-se a ocorrência deste fato devido ao caráter do projeto ora físico e ora abstrato. Inicialmente caracterizou-se como projeto de rede os seguintes pontos:

- Topologia física da rede.
- Barramento.
- Topologia lógica de rede.
- Estratégia de troca de informações.

### 4.1 TOPOLOGIA DA REDE

A organização técnica da ligação dos módulos de comunicação é chamada topologia de rede (SCHÄUFFELE, 2005). A figura 37 mostra as principais topologias de rede:

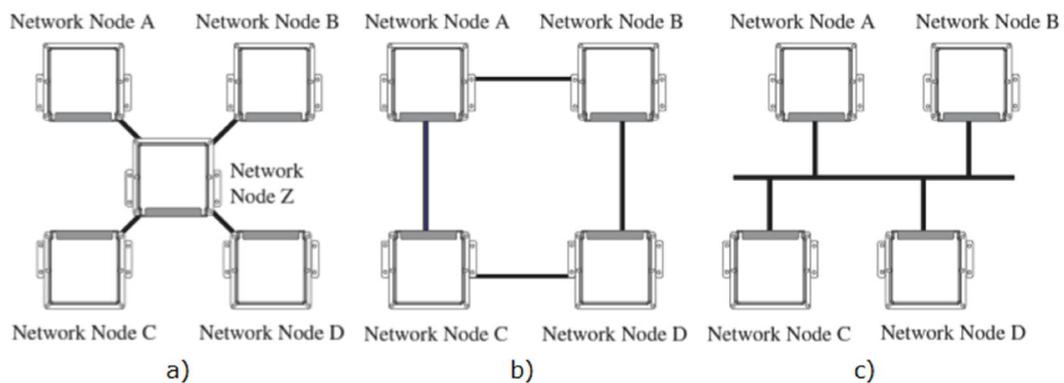


Figura 37: Topologias físicas de rede. Fonte: (SCHÄUFFELE, 2005).

#### 4.1.1 Topologia estrela

Nesta configuração nota-se que para interligar as ECUs utilizam-se conexões pontuais. Toda a comunicação é tratada pela ECU Z e o mau funcionamento dela corta toda a comunicação da rede.

### **4.1.2 Topologia Anel**

A topologia em anel é uma cadeia fechada de ligações pontuais entre as centrais. Todos os nós são capazes de regenerar a rede e transmitir informações para a próxima ECU. A falha de um nó não causa a falha da rede.

### **4.1.3 Topologia Linear**

A topologia linear compartilha o mesmo meio de comunicação, sendo assim a mensagem está disponível a todos os nós conectados ao barramento. A falha de um nó não causa necessariamente a falha da rede. A expansão deste tipo de rede é feita facilmente e várias relações lógicas podem ser implementadas sem grande esforço.

Após esta descrição rápida dos tipos de topologia física de rede pode-se notar que a escolha desta topologia pode influenciar tanto o projeto de software quanto a própria implantação física da rede. Uma vez posse destas informações, escolheu-se a topologia linear como a topologia física de rede. Pois esta possui vantagens em relações às demais nos seguintes aspectos:

- É expansível facilmente.
- Sua concepção é modular.
- Os dados são disponibilizados para toda a rede.
- E a falha de um módulo não implica na falha da rede.

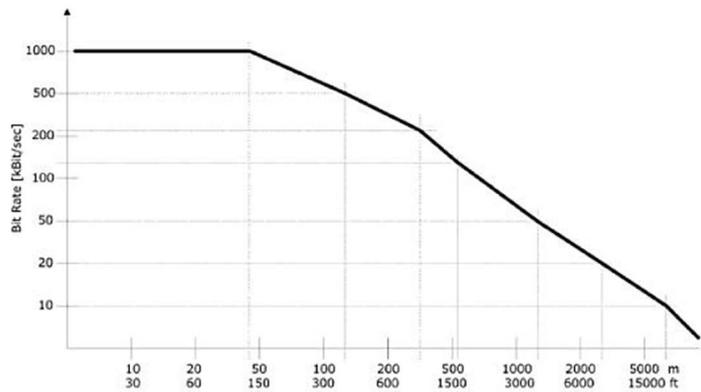
## **4.2 BARRAMENTO**

O barramento é o meio físico pelo qual passarão todos os sinais elétricos da comunicação CAN. Sendo assim, o dimensionamento correto do mesmo, torna-se fundamental para o sucesso do projeto. O barramento é composto somente por cabos e resistores de terminação, e desta forma é influenciado pelos seguintes parâmetros:

- Comprimento.
- Área da seção transversal do fio.
- Resistores de terminação.

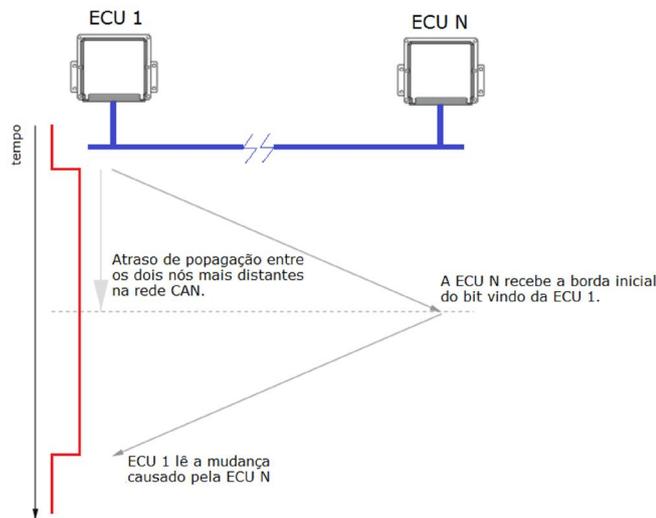
### **4.2.1 Comprimento**

O barramento influencia de forma direta a velocidade de comunicação, a figura 39 mostra a relação entre a velocidade de comunicação e o comprimento do barramento.



**Figura 38: Velocidade de comunicação vs comprimento do barramento. Fonte: (NATALE, 2012).**

Esta influência exercida pelo comprimento do barramento está relacionada com o fato de existirem atrasos na propagação do bit na comunicação CAN. Este atraso é ilustrado na figura 40:



**Figura 39: Propagação de atraso no bit. Fonte: Figura adaptada e traduzida (NATALE, 2012).**

Como nota-se na figura 40 o bit que a ECU 1 envia deve durar tempo suficiente para que a ECU N possa recebe-lo e alterar o estado do barramento de recessivo para dominante. Então, quanto mais longo for o barramento e mais distantes ficarem os módulos, maior o tempo que o bit deve durar. Este tempo de duração do bit é comumente chamado de *bit timing*. Dentro deste intervalo de bit timing ainda existe outra divisão, o *time quantum*. O time quantum são divisões do bit timing dedicadas à sincronização, sendo estas: *synchronization segment*, *propagation segment*, *phase segments*, e *sample point*. A figura 41 mostra o bit time, suas divisões e a sincronização:

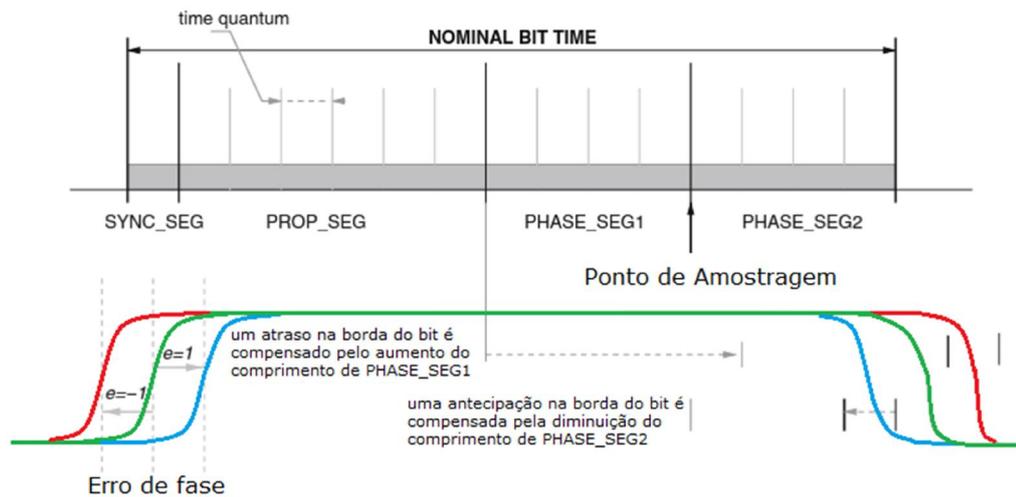


Figura 40: Definição de bit time e sincronização. Fonte: Figura adaptada e traduzida (NATALE, 2012).

Como pode-se observar na figura 41, existem 2 tipos de segmentos: os estáticos e os dinâmicos. Os estáticos são os dois primeiros, synchronization segment e propagation segment. O synchronization segment é reservado para a *Hard Synchronization* que é a sincronização que acontece logo no primeiro quantum time. O propagation segment como o próprio nome já diz é reservado para compensar o atraso devido à propagação do bit no barramento. Os segmentos dinâmicos são os dois últimos, phase segment1 e phase segment2. Estes segmentos podem aumentar ou diminuir de acordo com o especificado para realizar a *Re-synchronization*. A tabela 8 mostra alguns valores comuns para a comunicação CAN:

Tabela 8: Valores típicos para a Comunicação CAN.

Velocidade de dados	Bit time ( $\mu\text{s}$ )	Comprimento do barramento (m)
1 Mb/s	1	30
800 Kb/s	1.25	50
500 Kb/s	2	100
250 Kb/s	4	250
125 Kb/s	8	500
62.5 Kb/s	16	1000
20 Kb/s	50	2500
10 Kb/s	100	5000

Fonte: (NATALE, 2012).

#### 4.2.2 Área da seção transversal do fio:

Este parâmetro também influencia a comunicação CAN, pois quanto maior o comprimento do barramento, maior será a queda de tensão no mesmo. Desta forma é essencial o

dimensionamento correto da bitola do fio. A figura 42 ilustra este conceito de queda de tensão no barramento:

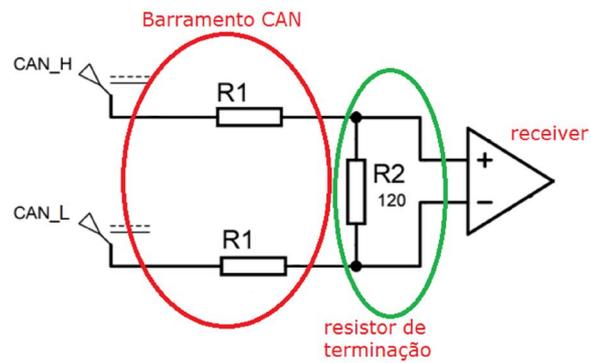


Figura 41: Modelo de queda de tensão no barramento. Fonte: Próprio autor.

Na figura 42 nota-se que o barramento está representado por dois resistores de mesmo valor R1. Este valor pode ser obtido através da equação 1:

$$R1 = \rho \frac{\ell}{A} \quad (1)$$

A equação (1) calcula a resistência elétrica de materiais, no caso cobre, aonde  $\rho = 1,72 * 10^{-8} \Omega m$ ,  $\ell$  é o comprimento do barramento e  $A$  é a área da seção transversal do fio.

#### 4.2.3 Resistor de Terminação

Este resistor é necessário no projeto de rede CAN, pois o mesmo impede a degradação do sinal não deixando o barramento comportar-se como antena. Geralmente seu valor está associado ao valor resistivo do barramento, cerca de 50 vezes maior que o valor do barramento. As figuras 43, 44 e 45 mostram a diferença entre o mesmo sinal dependendo dos resistores de terminação:

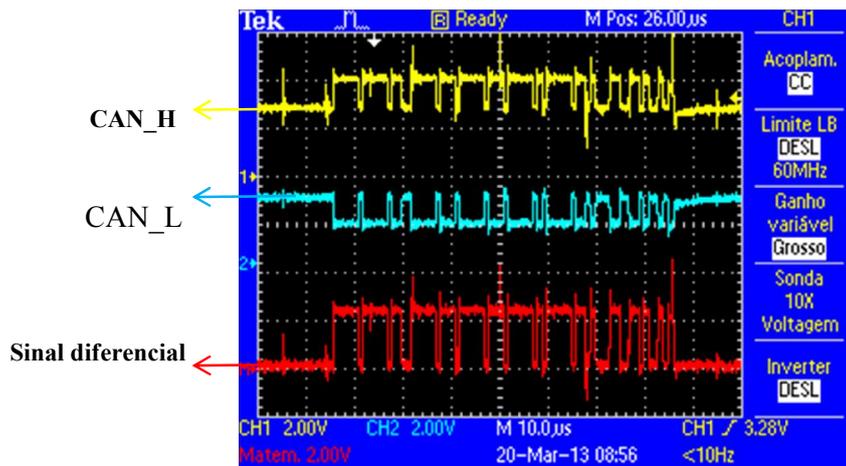


Figura 42: Sem resistor de terminação. Fonte: Próprio autor.

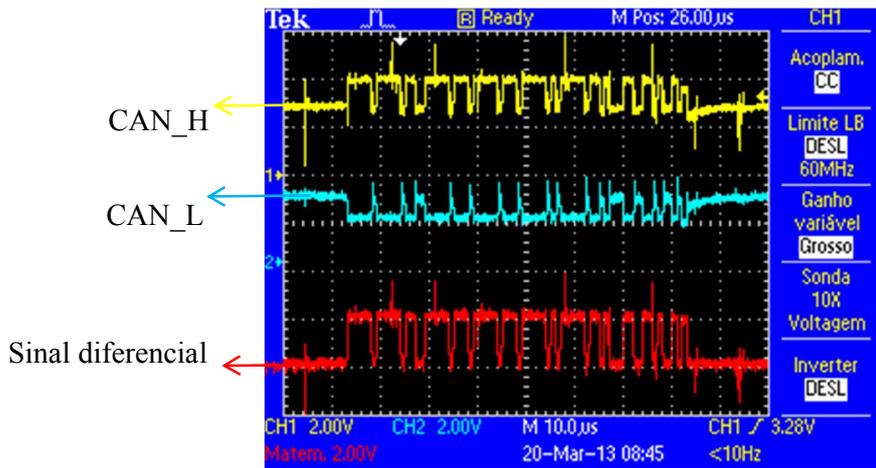


Figura 43: Com resistor de 60Ω. Fonte: Próprio autor.

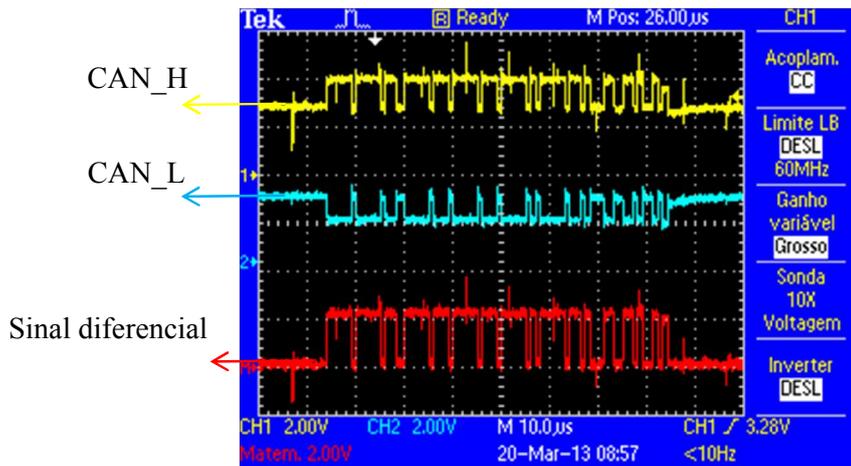


Figura 44: Com resistor de 120Ω. Fonte: Próprio autor.

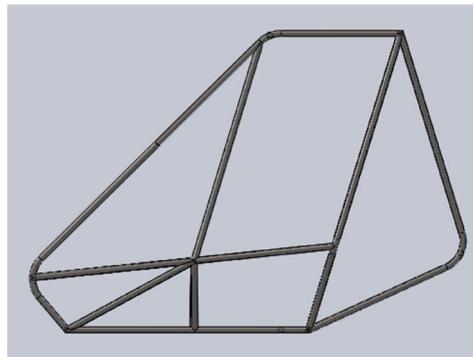
Nas figuras anteriores fez a aquisição sempre dos mesmos dados, sendo que a primeira forma de onda nas imagens é o sinal CAN\_H, a forma de onda do meio é sinal CAN\_L e o sinal diferencial é a última forma de onda. Como pode-se ver nas figuras anteriores os sinais em alguns momentos foram degradados, sendo o pior caso quando os resistores de terminação são de 60Ω.

### 4.3 PROJETO FÍSICO DA REDE

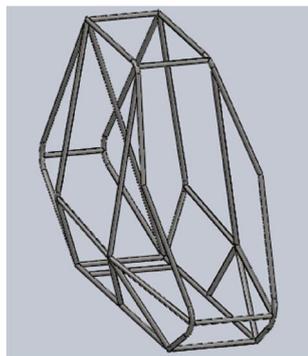
Levando em consideração que este projeto aplica-se ao mini baja, primeiramente discutiu-se com a equipe quais seriam as prioridades de montagem, qual o espaço disponível dentro do veículo. E ainda, qual seria a estrutura do mini baja para que pudesse dimensionar a rede CAN. Após esta reunião, chegou-se a conclusão que as prioridades da equipe eram:

- Aquisição de velocidade.
- Indicar frenagem.
- Aquisição de temperatura.
- Interface.

Em relação ao espaço disponível, este não seria problema, pois são poucos os elementos que compõe o mini baja. Já a estrutura do mini baja que é uma das informações mais importantes do projeto pode-se ver nas figuras 45 e 46:



**Figura 45:** Vista lateral da gaiola do mini baja. Fonte: Figura cedida pela equipe mini baja do CEFET-MG *campus V*.



**Figura 46:** Vista isométrica da gaiola do mini baja. Fonte: Fonte: Figura cedida pela equipe mini baja do CEFET-MG *campus V*.

Uma vez que tem-se o comprimento exato do barramento, pode-se dimensionar o mesmo seguindo os procedimentos anteriores. Para tal tarefa, primeiro tem-se que arbitrar uma variável,

ou o resistor de terminação, ou a área de seção transversal do fio. Para esta análise resolveu-se arbitrar os resistores de terminação em  $120\Omega$ , então a área mínima do fio deve ser de:

$$A = \frac{1,72 \times 10^{-8} \times 5}{\frac{120}{50}}$$

$$A = 0,036mm^2$$

Porém este só se aplica ao barramento de 5m. Então resolveu-se calcular a bitola do fio para o barramento de 40m, pois qualquer extensão feita ao barramento estaria prevista no projeto.

Desta forma, a nova bitola do fio é:

$$A = \frac{1,72 \times 10^{-8} \times 40}{\frac{120}{50}}$$

$$A = 0,286mm^2$$

Como informado anteriormente, a velocidade de comunicação depende do comprimento do barramento. Entretanto, com 40 metros ainda é possível transmitir dados a 1Mb/s. Desta maneira o barramento ficou dimensionado conforme a tabela 9, e pode ser visto na figura 47:

**Tabela 9: Parâmetros de construção da rede.**

Projeto Físico da rede	
Comprimento	Até 40m
Velocidade	1Mb/s
Bit time	1 $\mu$ s
Topologia	Linear
Número de módulos	5
Seção transversal do fio	22AWG
Resistores de terminação	120 $\Omega$

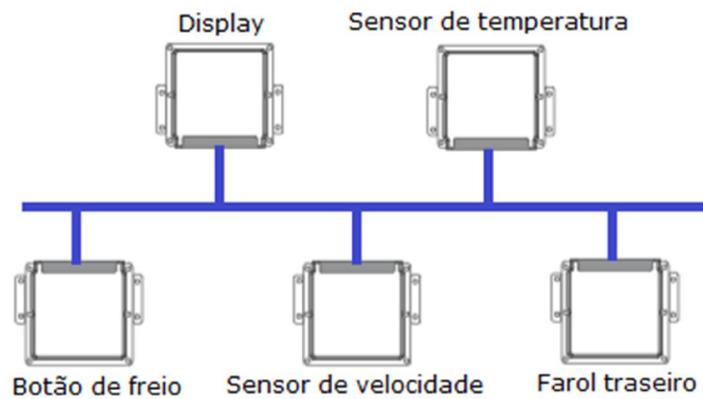


Figura 47: Rede CAN dimensionada para o mini baja. Fonte: Próprio autor.

#### 4.4 TOPOLOGIA LÓGICA DE REDE

A topologia física é muito importante, pois ela define algumas características importantes, como: Uso de repetidores, capacidade de regeneração, disponibilidade dos dados, quantidade de cabos gasta. Porém, a topologia lógica é muito mais poderosa, pelo fato de ser abstrata e estar presente somente em software. No nível físico tem-se um número limitado de conexões, já no nível lógico, ou abstrato, tem-se um número de ligações muito maior. No caso da rede CAN, apesar do transceiver não permitir este número de nós sem o uso de repetidores ou pontes, pode-se usar  $2^{11} = 2048$  endereços no modo padrão e  $2^{29} = 536870912$  endereços no modo estendido. Estes endereços podem ser funcionalidades diferentes presentes no mesmo módulo. A figura 48 ilustra este conceito:

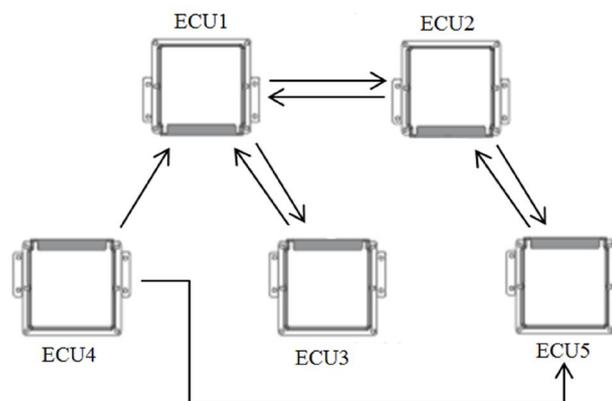


Figura 48: Topologia Lógica. Fonte: Próprio autor.

Na figura anterior observe-se a camada lógica da rede, nota-se que uma central pode receber e trocar dados com várias centrais.

#### 4.5 ESTRATÉGIA DE TROCA DE INFORMAÇÕES

Para a troca de informações tem-se duas abordagens: cliente/servidor, e produtor/consumidor. O modelo cliente/servidor constitui-se de uma comunicação bilateral aonde o cliente pede uma informação em um instante inicial e o servidor fornece o dado no momento posterior, como ilustra a figura 49:

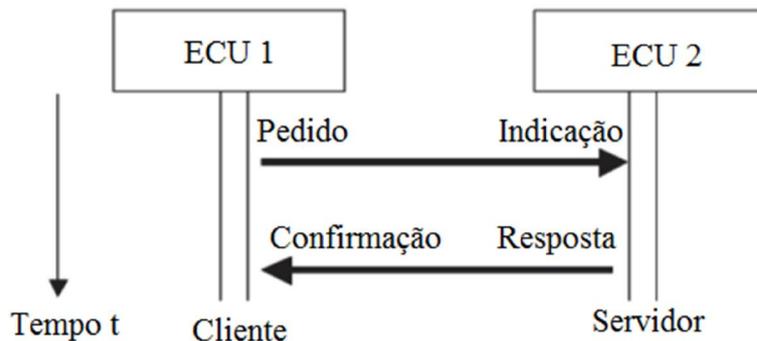


Figura 49: Modelo Cliente/Servidor. Fonte: (SCHÄUFFELE, 2005).

Já o modelo produtor/consumidor funciona com o produtor colocando dados no barramento sem nenhum pedido prévio, como mostra a figura 54:

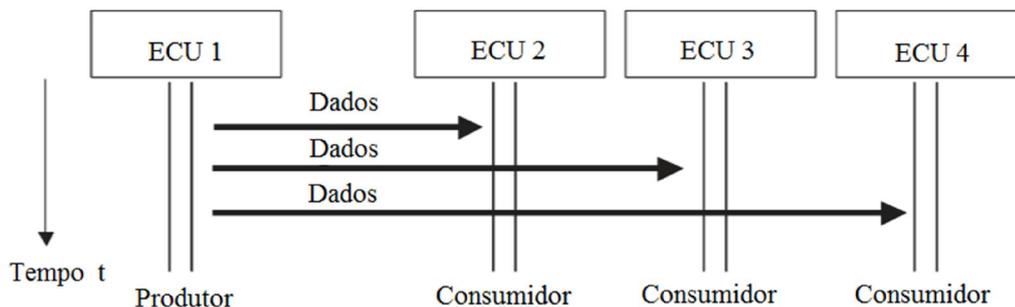
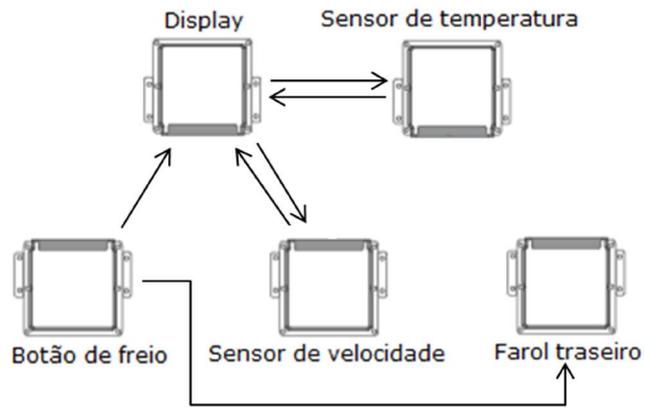


Figura 50: Modelo Produtor/Consumidor. Fonte: (SCHÄUFFELE, 2005).

#### 4.6 PROJETO LÓGICO DA REDE

Neste projeto especificou-se os modelos de troca de informações, e como elas acessam o barramento. Através da figura 51, pode-se ver que entre os módulos de display, temperatura, e velocidade o modelo de troca de informações é o cliente/servidor. Já entre botão de freio farol traseiro e display o modelo utilizado foi o produtor/consumidor.



**Figura 51: Topologia lógica projetada para o mini baja. Fonte: Próprio autor.**

Em relação ao modo de acesso ao barramento, escolheu-se o modo temporizado. Para o sensor de velocidade escolheu-se intervalos de 10ms e para o sensor de temperatura intervalos de 1s. Já para o botão de freio não tem intervalo de tempo, uma vez que o próprio evento dispara a comunicação.

## 5. Projeto de Software

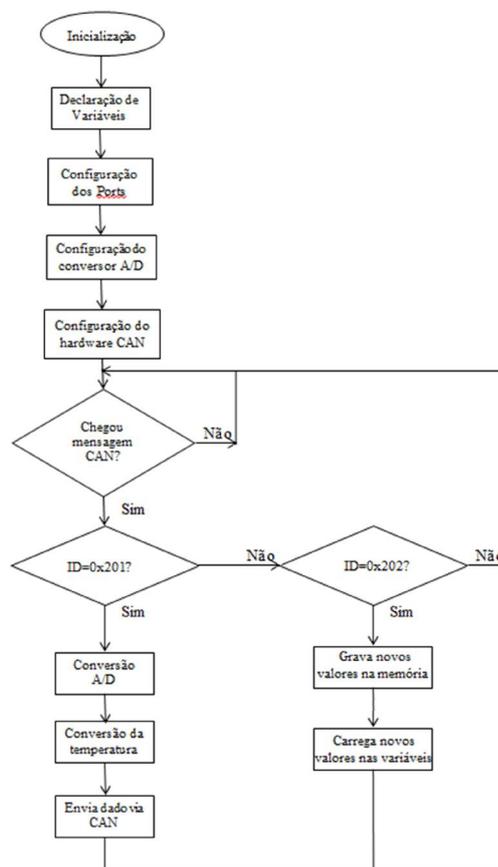
Dentro deste projeto inclui-se dois tipos de programação. O primeiro tipo de programação é a linguagem C, que foi utilizada para programar os microcontroladores. E o segundo tipo de programação é a linguagem G utilizada no LabView.

### 5.1 MICROCONTROLADORES

Para a programação dos microcontroladores utilizou-se o software MPLAB juntamente com o compilador CCS e suas bibliotecas, sendo que para cada módulo os programas variam devido às funcionalidades diferentes.

#### 5.1.1 Sensor de temperatura:

O programa para este módulo tem a seguinte característica:



Na seção hardware não define-se como se dá o processo de conversão da temperatura deixando-a a cargo do software, desta maneira esse processo será explicado nesta seção. Inicialmente, analisou-se o circuito da seguinte forma: como a tensão da conversão A/D é sempre conhecida, então, pode-se achar a corrente  $I_t$  do circuito facilmente, da seguinte forma:

$$I_t = \frac{5 - V_{ad}}{1000}$$

Como o comportamento do MTE4053 é uma queda exponencial supõe-se que sua resistência tem a seguinte característica:

$$R = K_1 e^{K_2 Temp}$$

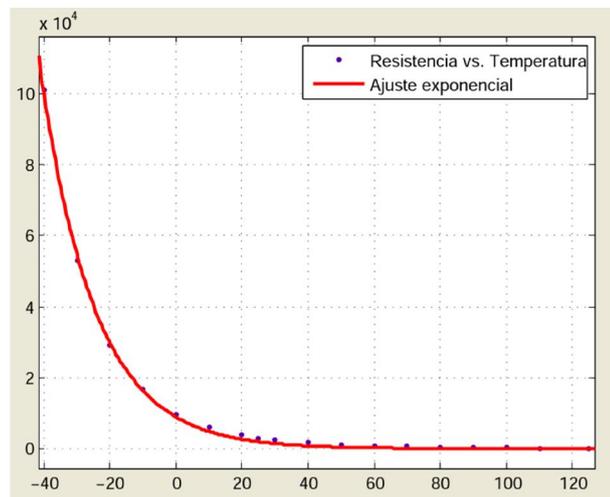
Sendo assim a tensão mensurada na conversão A/D possui a seguinte equação:

$$V_{ad} = I_t K_1 e^{K_2 Temp}$$

Desta forma a temperatura pode ser encontrada da seguinte maneira:

$$Temp = \frac{\ln\left(\frac{V_{ad}}{I_t K_1}\right)}{K_2}$$

Para estimar-se os ganhos  $K_1$  e  $K_2$  utilizou-se o Toolbox Curve Fitting do Matlab, usando a tabela 3 como dados de entrada. Feita a entrada de dados, ajustou-se a curva com uma aproximação exponencial, como mostra a figura 15:

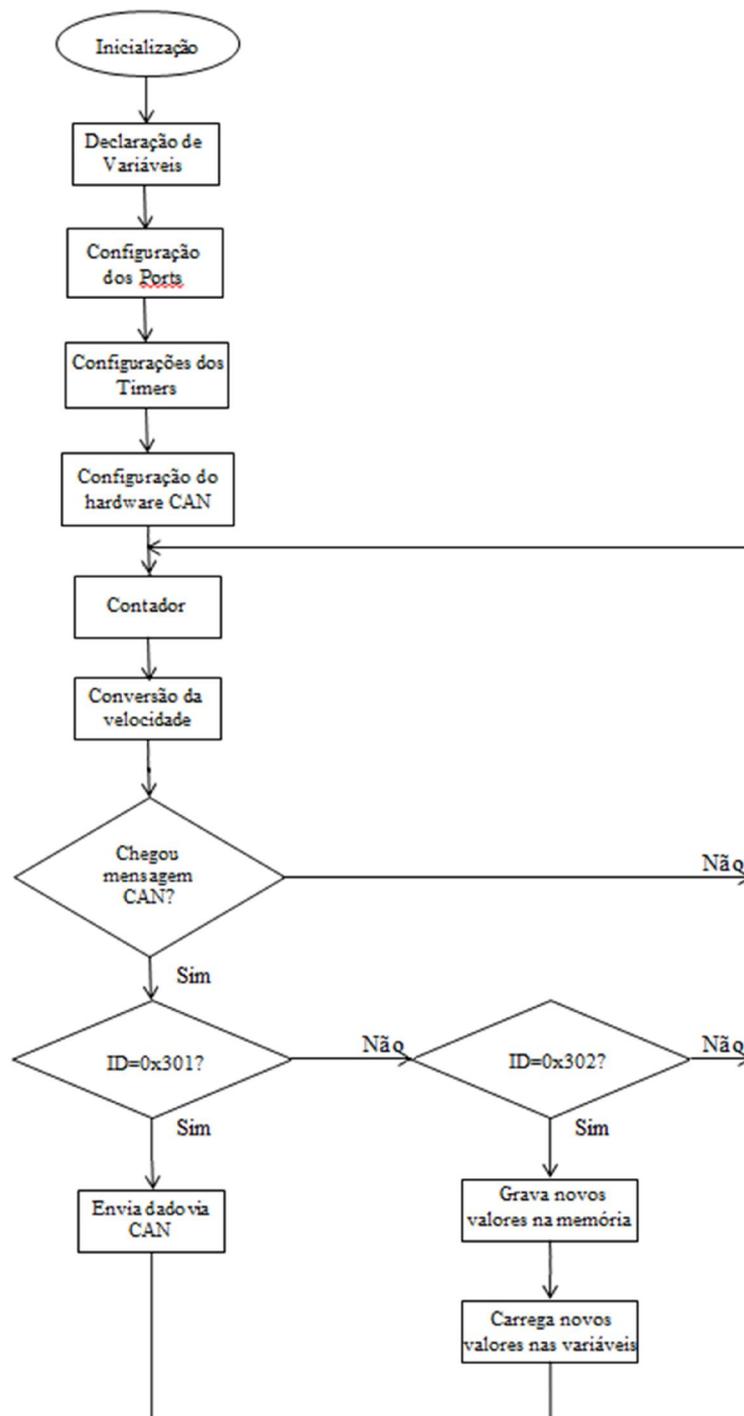


**Figura 52: Ajuste exponencial para o MTE4053. Fonte: Próprio autor.**

Como pode-se ver, a curva vermelha é a aproximação exponencial tendo como ganhos  $K_1 = 8872$  e  $K_2 = -0,0606$ .

### 5.1.2 Sensor de velocidade:

O programa para este módulo tem a seguinte característica:



Na seção hardware não define-se como se dá o processo de conversão da velocidade deixando-a a cargo do software, desta maneira esse processo será explicado nesta seção. Inicialmente, configura-se um timer para incrementar com sinal externo, e outro para incrementar com sinal interno conhecido.

$$Velocidade = \frac{timer2}{6 \times timer1} \times K_d$$

Na equação acima  $timer2$  é incrementado pelo sinal externo vindo do sensor, e  $timer1$  é incrementado com uma base de tempo periódica conhecida, e  $K_d$  é uma constante que relaciona a distância percorrida pelo mini baja com 1 revolução do CVT.

## **5.2 LABVIEW**

Com o uso do LabView desenvolveu-se dois conceitos muito importantes no que diz respeito ao desenvolvimento de hardware, estes conceitos são:

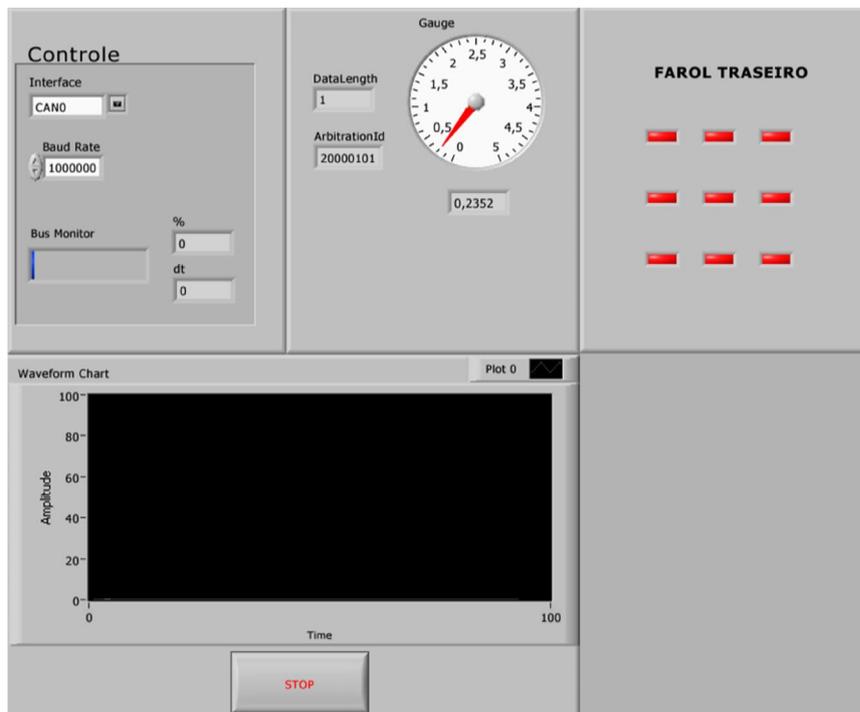
- Atualização de parâmetros.
- Hardware in the Loop.

### **5.2.1 Atualização de Parâmetros:**

Esta atualização tem o objetivo de alterar os parâmetros do software de aquisição de dados com o próprio módulo funcionando, esses parâmetros são:  $K_1$ ,  $K_2$ , e  $K_d$ . E o programa que coordena estas alterações é mostrado na figura 2:

### **5.2.2 Hardware in the Loop:**

Com a utilização de HIL é possível emular qualquer hardware, sendo assim não há necessidade de construir-se um protótipo. Sem a construção do protótipo, o projeto fica mais barato e mais rápido. Neste projeto emulou-se o farol traseiro. O farol traseiro comunica-se CAN e tem dois estados: aceso quando o freio não está pressionado, e piscando quando o freio está pressionado. A figura 3 mostra o programa em LabView:



**Figura 53: Painel Frontal LabView. Fonte: Próprio autor.**

Neste programa procurou-se emular dois hardwares, o farol traseiro e o segundo a comunicação entre uma conversão A/D e o computador. O primeiro já foi descrito acima, e o segundo foi importante, pois a partir dele notou-se que para este tipo de aplicação deve-se ter uma conversão no módulo que recebe os dados uma vez que os dados enviados via CAN são de 8 bits e do tipo inteiro. Além deste detalhe, também observou-se qual seria o comportamento da rede com essas aplicações e viu-se que as mesmas não causaram alterações na rede CAN. A figura 3 traz o diagrama de blocos desta aplicação, para fazer a comunicação CAN via LabView utilizou-se o pacote XNET 1.1, pois era o único compatível com o LabView 8.5. Neste pacote XNET 1.1 tem os dois modos de programação CAN para Labview: A Channel API, e a Frame API. Para esta aplicação vimos que seria melhor utilizar a Frame API por obter configurações quase em nível de hardware, dessa forma pode-se ler e alterar quase todos os campos da comunicação CAN.

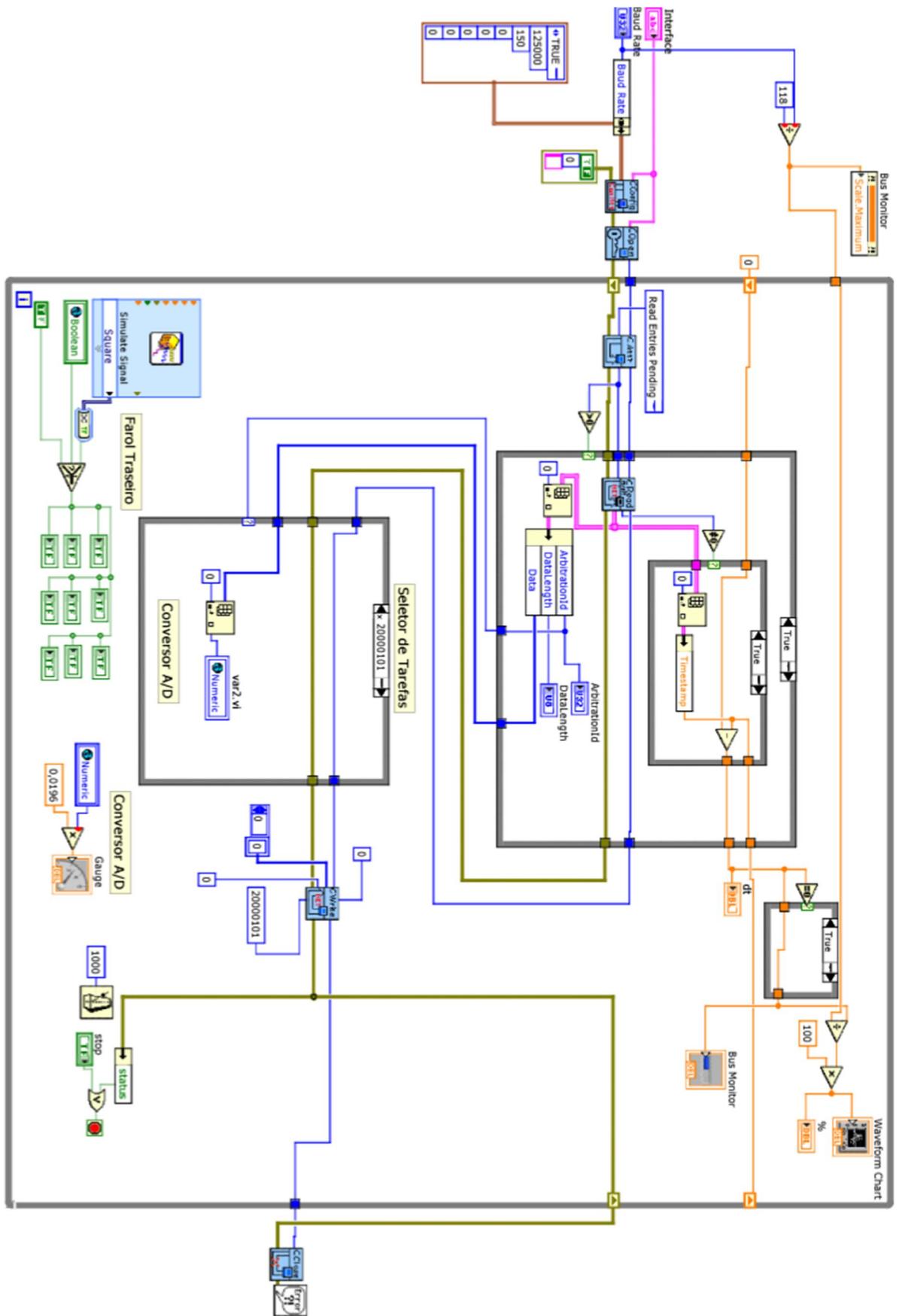


Figura 54: Diagrama de blocos LabView. Fonte: Próprio autor.

## 6. Conclusões

Analisando os resultados obtidos juntamente com o projeto desenvolvido, observa-se que cada módulo custa em torno de quarenta reais, somente componentes. Nota-se também que o número total de conexões cai para o valor de quatro vezes o número de módulos. Já o peso dos cabos não chega a ser representativo no projeto do mini baja, uma vez que o peso do próprio mini baja é muito superior ao peso dos cabos. Outra observação importante a ser realizada é o fato de o microcontrolador executar menos tarefas, desta forma pode-se escolher um microcontrolador mais lento e conseqüentemente mais barato. Uma vez que o módulo executa menos tarefas, o software também fica mais simples, entretanto são necessários mais módulos para realizar as mesmas tarefas e desta forma são feitos mais softwares. A característica mais importante deste projeto é a modularidade, todos os nós da rede são feitos a partir de módulos genéricos o que facilita tanto a montagem quanto a manutenção. Contudo, conclui-se que este projeto preencheu a lacuna do projeto da equipe do mini baja do CEFET-MG *campus V*. Este projeto demonstrou-se ser mais caro, porém trabalha com uma tecnologia atual, de alto desempenho e presente na indústria automobilística. O campo de desenvolvimento de hardware automotivo mostrou-se bastante desafiador, porém trabalhando com comprometimento e foco no resultado foi possível realizar este trabalho.

Em trabalhos futuros sugerem-se estudos relacionados a:

- Desenvolvimento de uma ECU para freio ABS.
- Desenvolvimento de uma ECU para suspensão ativa.
- Desenvolvimento de uma ECU para injeção eletrônica.
- Desenvolvimento de novos módulos utilizando microcontroladores da linha Atmel.
- Desenvolvimento de novos módulos utilizando o LT3682.
- Utilização de Hardware In The Loop para desenvolver toda a arquitetura eletrônica do mini baja.
- Desenvolvimento de gateways para obter-se diferentes velocidades de rede CAN.
- Desenvolvimento de gateways para obter-se comunicação wireless.

## Referências Bibliográficas

- NATALE, M. D. **Understanding and Using the Controller Area Network Communication Protocol: Theory and Practice**. 2012.
- SCHÄUFFELE, J. **Automotive Software Engineering: Principles, Processes, Methods, and Tools**. 2005.
- ISO. **INTERNATIONAL STANDARD ISO 11898-2**. 2003.
- RICHARDS, P. A **CAN Physical Layer Discussion**. 2002.
- EMADI, A. **Handbook of Automotive Power Electronics and Motor Drives**. 2005.
- Barbosa, L. R. G. **Rede CAN**. 2003.
- TECHNOLOGY, M. **MCP2551 Data Sheet**. 2003.
- TECHNOLOGY, M. **PIC18F4580 Data Sheet**. 2006.
- TECHNOLOGY, L. **LT3682 Data Sheet**. 2008.
- MARELLI, M. **MTE4053 Data Sheet**. 2001.
- IEEEXPLORE. Disponível em: <http://ieeexplore.ieee.org>. Acesso em: 05/05/2012