

UNIVERSIDADE FEDERAL DO PARANÁ

BRUNO CUETO

**PROTOCOLO DE COMUNICAÇÃO CAN APLICADO A VEÍCULOS:
ESTUDO, PROJETO E IMPLEMENTAÇÃO DE UM EQUIPAMENTO COM
FINALIDADE DIDÁTICA**

CURITIBA

2018

BRUNO CUETO

**PROTOCOLO DE COMUNICAÇÃO CAN APLICADO A VEÍCULOS:
ESTUDO, PROJETO E IMPLEMENTAÇÃO DE UM EQUIPAMENTO COM
FINALIDADE DIDÁTICA**

Trabalho de Conclusão de Curso apresentado ao
Curso de Engenharia Elétrica da Universidade
Federal do Paraná como requisito à obtenção do
título de obtenção do grau de Engenheiro
Eletricista.

Orientador: Prof. Dr. Ewaldo Luiz de Mattos Mehl

CURITIBA

2018

TERMO DE APROVAÇÃO

BRUNO CUETO

PROTOCOLO DE COMUNICAÇÃO CAN APLICADO A VEÍCULOS: ESTUDO, PROJETO E IMPLEMENTAÇÃO DE UM EQUIPAMENTO COM FINALIDADE DIDÁTICA

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia Elétrica da Universidade Federal do Paraná como requisito à obtenção do título de obtenção do grau de Engenheiro Eletricista, pela seguinte banca examinadora:

Orientador:

Prof. Dr. Ewaldo Luiz de Mattos Mehl

Departamento de Engenharia Elétrica, UFPR

Prof. Dr. Carlos Marcelo Pedroso

Departamento de Engenharia Elétrica, UFPR

Prof. Dr. Eduardo Parente Ribeiro

Departamento de Engenharia Elétrica, UFPR

Curitiba, de junho de 2018

AGRADECIMENTOS

Profundo agradecimento aos meus pais José Roberto Bresser Cueto e Cynthia V.P.O. Cueto e familiares pelo suporte oferecido durante todo o período de minha formação, onde sempre acreditaram em meu sucesso.

À minha namorada, Paola Zanardi, por sempre ter me apoiado nos momentos mais difíceis, pela motivação em todos eles, pelo companheirismo e amor.

Ao meu orientador, Prof. Dr. Ewaldo Luiz de Mattos Mehl, por todo o incentivo e suporte oferecido durante o projeto.

Aos meus amigos e colegas que ajudaram de diversas formas e estiveram presentes em toda a trajetória da graduação.

Aos profissionais do SENAI da sede Automotiva, Valério Mendes Marochi e Luiz Carlos Konrath, pelo suporte no auxílio para desenvolvimento da bancada didática.

Aos professores do curso de Engenharia Elétrica da Universidade Federal do Paraná por todos os conhecimentos transmitidos para a formação e crescimento pessoal que possibilitaram.

RESUMO

Nos veículos automotores da atualidade estão presentes tecnologias e soluções de conectividade com alto nível de desempenho que tornam possível o desenvolvimento tecnológico e agregam segurança e conforto aos veículos. Dentre estas tecnologias, figuram as redes de comunicação, que viabilizam a troca de informações entre centrais de controle, através de arquiteturas confiáveis e resistentes a interferências eletromagnéticas. O presente trabalho pretendeu estudar um dos protocolos de comunicação veicular (Rede CAN), explicitando o seu funcionamento, elementos e características do mesmo. Para tanto, confeccionou-se uma bancada de finalidade didática de aplicação automotiva, que replica, de forma simplificada, uma arquitetura de protocolo CAN, tendo como objetivos, facilitar o aprendizado de estudantes de cursos técnicos e universitários das áreas a fim, bem como incentivar a pesquisa e desenvolvimento de novos dispositivos e soluções inovadoras para o segmento automotivo em escala regional.

Palavras-Chave: Protocolos de Comunicação. Rede CAN. Redes Veiculares. Conectividade Automotiva. Bancada didática. MCP2515. Arduino.

ABSTRACT

In today's automotive vehicles there are technologies and connectivity solutions with high level of performance that make possible the technological development and add safety and comfort to the vehicles. Among these technologies, there are communication networks, which enable the exchange of information between control centers, through reliable architectures and resistant to electromagnetic interference. This paper intends to study one of the communication protocols (CAN Network), explaining its operation, elements and characteristics of the same. For that, a didactic design was created for automotive application, which replicates, in a simplified way, a CAN protocol architecture, with the objective of facilitating the students' learning of technical and university courses in the areas in order to encourage research and development of new devices and innovative solutions for the automotive segment on a regional scale.

Palavras-Chave: Communication Protocols. CAN network. Vehicle Networks. Automotive Connectivity. Didactic bench. MCP2515. Arduino.

LISTA DE ILUSTRAÇÕES

Figura 1 – Padrão OSI de sete camadas (ISO 7498).....	13
Figura 2 - Arquitetura do CAN, divisões em camadas e as respectivas tarefas.....	14
Figura 3 - Formato de mensagem com identificador padrão a partir do quadro de bits do CAN.....	16
Figura 4 - Formato de mensagem com identificador estendido.....	16
Figura 5 - Diagrama elétrico da interligação dos componentes eletroeletrônicos da bancada.....	23
Figura 6 - Diagrama Elétrico de Alimentação da bancada.....	24
Figura 7 - Diagrama Elétrico do Arduino 1 e MCP2515 de leitura de dados CAN.....	25
Figura 8 - Diagrama Elétrico do Arduino 2 e MCP2515 do comando de iluminação externa.....	25
Figura 9 - Diagrama elétrico do módulo relé, faróis e lanternas.....	26
Figura 10 - Fonte de alimentação chaveada.....	27
Figura 11 – Arduino UNO.....	27
Figura 12 - Módulo CAN BUS MCP2515.....	28
Figura 13 - Painel de instrumentos do Peugeot 308.....	29
Figura 14 – Alavanca de setas do Peugeot 308.....	29
Figura 15 - BSI, unidade de bordo do Peugeot 308.....	30
Figura 16 – Porta fusíveis automotivo.....	31
Figura 17 - Módulo Relé 4 canais para Arduino.....	32
Figura 18 – Interruptor de porta.....	32
Figura 19 – LEDs de sinalização.....	33
Figura 20 - Sensor de nível de combustível automotivo.....	34
Figura 21 - Primeiro esboço da posição dos componentes da bancada.....	34
Figura 22 - Componentes e diagramas elétricos do Peugeot 308.....	35

Figura 23 - Primeiro teste de funcionamento do painel de instrumentos e BSI.....	35
Figura 24 - Fixação de componentes na chapa de MDF.....	36
Figura 25 - Montagem do chicote na parte traseira da bancada.....	36
Figura 26 - Posição final de cada componente.....	37
Figura 27 - Posição final do roteamento do chicote.....	37
Figura 28 - Pintura da chapa de MDF.....	38
Figura 29 - Bancada didática em funcionamento.....	39
Figura 30 - Chicote finalizado.....	39
Figura 31 - Funções de iluminação externa através do conjunto de alavancas.....	40
Figura 32 - Funções de abertura de portas.....	41
Figura 33 - Funções de posição de pedal de acelerador e embreagem.....	41
Figura 34 - Funções da central do motor do Peugeot 307.....	42
Figura 35 - Declaração das bibliotecas CAN.....	42
Figura 36 – Bibliotecas de identificadores de mensagens.....	43
Figura 37 – Definição de pinos de acionamento dos relés.....	43
Figura 38 - Leitura de mensagens e posição da chave de seta.....	44
Figura 39 – Leitura das mensagens e posição da chave de seta.....	45
Figura 40 - Alunos do curso de eletrônica automotiva em aula prática do SENAI.....	46
Figura 41 - Alunos do curso de eletrônica automotiva em aula prática do SENAI.....	47
Figura 42 – Osciloscópio ilustrando mensagem CAN.....	47

LISTA DE TABELAS

TABELA 1 – Protocolos de comunicação veicular.....	12
TABELA 2 – Principais padrões baseados no CAN.....	19
TABELA 3 – Tabela de componentes eletroeletrônicos.....	23

LISTA DE ABREVIATURAS

CAN	–	<i>Controller Area Network</i>
PSA	–	<i>Peugeot Citroën Moteurs</i>
MCP	–	<i>Master Control Program</i>
SPI	–	<i>Serial Peripheral Interface</i>
BSI	–	<i>Boîtier de Servitude Intelligent</i>
CAN H	–	<i>CAN High</i>
CAN L	–	<i>CAN Low</i>
OSI	–	<i>Open System Interconnection</i>
ISO	–	<i>Internacional Organization for Standardization</i>
LLC	–	<i>Logical Link Layer</i>
MAC	–	<i>Medium Access Control</i>
MDF	–	<i>Medium Density Fiberboard</i>
RPM	–	<i>Rotações por minuto</i>
OBD	–	<i>On-Board Diagnostic</i>
CSMA/CR	-	<i>Carrier Sense Multiple Access/Collision Resolution</i>
RTR	–	<i>Requisição de transmissão remota</i>
IDE	–	<i>Identificador de Extensão</i>
DLC	–	<i>Data Length Code</i>
CRC	–	<i>Campo de verificação de Redundância Cíclica</i>
ACK	–	<i>Acknowledgement</i>
REC	–	<i>Receiver Error Counter</i>
TEC	–	<i>Transmitter Error Counter</i>
SAE	–	<i>Society of Automotive Engineers</i>
DIN	-	<i>Deutsches Institut für Normung</i>
NMEA	–	<i>National Marine Electronic Association</i>

ABS – Anti-lock Brake System

CPU – Central Processing Unit

CI – Circuito integrado

PWM – Pulse Width Modulation

USB – Universal Serial Bus

SENAI – Serviço Nacional de Aprendizagem Industrial

SUMÁRIO

1. INTRODUÇÃO.....	11
2. MOTIVAÇÃO E JUSTIFICATIVA.....	11
3. CAN – CONTROLLER AREA NETWORK.....	12
3.1 MECANISMO DE ACESSO AO MEIO - CSMA/CR.....	15
3.2 MENSAGENS DO CAN	15
3.2.1 FORMATO DA MENSAGEM	15
3.3 TIPOS DE QUADROS.....	16
3.3.1 QUADRO DE DADOS.....	16
3.3.2 QUADRO DE REQUISIÇÃO REMOTA.....	17
3.3.3 QUADRO DE ERRO.....	17
3.3.4 QUADRO DE SOBRECARGA.....	17
3.4 DETECÇÃO E TRATAMENTO DE ERROS.....	18
3.5 FILTROS DE MENSAGEM	18
3.6 PADRÕES DE PROTOCOLOS DE ALTO NÍVEL	19
3.7 APLICAÇÕES.....	20
3.7.1 APLICAÇÕES AGRÍCOLAS.....	20
3.7.2 APLICAÇÕES AUTOMOTIVAS.....	20
3.7.3 APLICAÇÕES COMERCIAIS.....	20
3.7.4 APLICAÇÕES INDUSTRIAIS	20
3.8 DESENVOLVIMENTO.....	21
3.9 DOCUMENTAÇÃO DO PROJETO.....	21
3.9.1 ESCOPO.....	21
3.9.2 CRONOGRAMA.....	21
3.9.3 ORÇAMENTO.....	22
3.9.4 QUALIDADE	22
3.9.5 RECURSOS HUMANOS	22
4. ARQUITETURA ELETROELETRÔNICA.....	23
4.1 DESENVOLVIMENTO DO SOFTWARE.....	40
4.2 INTEGRAÇÃO DA BANCADA.....	45
5. VALIDAÇÃO EM AULA PRÁTICA	46
6. CONCLUSÕES E SUGESTÃO PARA TRABALHOS FUTUROS	48
6.1 CONCLUSÕES.....	48
6.2 SUGESTÕES PARA TRABALHOS FUTUROS	48
REFERÊNCIAS	49
ANEXO A – CRONOGRAMA DE EXECUÇÃO DO PROJETO	50

ANEXO B – ORÇAMENTO INICIAL DE EXECUÇÃO DO PROJETO	51
ANEXO C – ORÇAMENTO FINAL DE EXECUÇÃO DO PROJETO.....	52

1. INTRODUÇÃO

O intenso avanço tecnológico alavancou o desenvolvimento de produtos de alto desempenho, multifuncionais e de custo menor. Partindo deste princípio, temos um processo de descentralização das aplicações, o qual tem se tornado uma tendência mundial. Com isso, é necessário que as aplicações tenham partes mais independentes, o que nos leva a um modelo distribuído.

Os modelos convencionais de comunicação e controle utilizados nos sistemas industriais nas últimas décadas têm sido por meio de cabeamento, o qual conecta um dispositivo de controle à sensores e atuadores. No entanto, este método centralizado não favorece aos novos requisitos de desenvolvimento de sistemas de controle como modularidade, controle descentralizado, diagnóstico facilitado e menor custo (Yang, 2006). Atualmente, a arquitetura das redes é baseada no conceito de barramentos (*Fieldbus*), tendo em vista que melhoram a confiabilidade, eficiência, flexibilidade do sistema e ainda são capazes de reduzir o tempo e os custos de manutenção e instalação (Moyne e Tilbury, 2007). A fim de aproveitarem da melhor forma possível este modelo distribuído de comunicação e transmissão de dados, vários protocolos de rede vêm sendo desenvolvidos.

O protocolo CAN (Bosch, 2005) atualmente é um dos protocolos mais utilizados nos sistemas de controle distribuído, principalmente nas áreas de automação industrial, eletrônica embarcada (Automotiva e Agrícola) e robótica, por meio de diversos padrões tais como CANOpen, J1939, DeviceNet e SDS (CIA, 2006). Dentre as suas principais vantagens temos o baixo custo de instalação e manutenção, diversos dispositivos com controladores CAN, elevado grau de segurança na detecção de erros e priorização de mensagens.

2. MOTIVAÇÃO E JUSTIFICATIVA

A rede CAN, apesar de ter sido criada em meados da década de 80 por Robert Bosch, é uma tecnologia que tem ganhado muita atenção em relação ao desenvolvimento dos sistemas embarcados. Atualmente é possível aplicar este tipo de rede em sistemas automotivos, industriais, comerciais, agrícolas. Com isso, as razões que levaram à proposição deste projeto são tanto técnicas quanto educacionais, apontando para o desenvolvimento de um módulo didático de estudo de redes veiculares, que torne o aprendizado do protocolo CAN mais fácil e prático, além de incentivar estudantes das áreas técnicas e tecnológicas a pesquisar e desenvolver dispositivos e soluções inovadoras para o segmento automotivo.

3. CAN – CONTROLLER AREA NETWORK

O CAN é um protocolo digital de comunicação serial que suporta eficientemente os conceitos de controle distribuído de tempo real com elevado grau de segurança (CIA, 2000). Foi criado pela empresa Robert Bosch em 1986 com a finalidade de aplicação na indústria automotiva para simplificar os complexos sistemas de comunicação e controle dos diversos módulos que equipavam os veículos. Além do CAN existiam em paralelo outros protocolos de comunicação veicular, como apresentado na tabela abaixo.

TABELA 1 – Protocolos de comunicação veicular.

Protocolo	Classe	Taxa de Transmissão	Instituições Diretamente Relacionada
SINEBUS	A	10Kbps	Delco Eletronics
I ² C	A	10Kbps	Phillips
SAE J1708	A	10Kbps	SAE/TCM
ACP	A	10Kbps	Chrysler
BEAN	A	10Kbps	Ford
LIN	A	10Kbps	Motorola
CAN 2.0 ISO 11898	B	10Kpbs a 125Kpbs	SAE/ISO
CAN 2.0 SAE J1939	B	10Kpbs a 125Kpbs	SAE
J1859 Class 2	B	10Kpbs a 125Kpbs	GM
J1850 SCP	B	10Kpbs a 125Kpbs	Ford
J1850 PCI	B	10Kpbs a 125Kpbs	Chrysler
CAN 2.0 ISO 11898	C	125Kpbs a 1Mbps	SAE/ISO
CAN 2.0 SAE J139	C	125Kpbs a 1Mbps	SAE

FONTE: O autor (2018).

Nas últimas décadas o protocolo vem ganhando notoriedade, nas aplicações de automação e controle. Segundo (Sousa, 2002) pode-se ressaltar algumas vantagens sobre os demais protocolos como:

- Taxas de transmissão com operação desde poucos kbps até 1Mbps;
- Comunicação através de 2 fios, minimizando custos e a complexidade de instalação, além de evitar interferências eletromagnéticas;
- Tamanho de dados por quadro otimizados, com menor tempo de ociosidade durante a transmissão;
- Possibilidade de escolha de rede com comunicação ponto a ponto (cliente-servidor) ou *multicast* (comunicação grupal);
- Flexibilidade para adicionar, remover e alterar dispositivos conectados ao barramento, facilitando a manutenção do sistema.

Alguns dos protocolos que serão mencionados neste trabalho, são baseados no padrão de comunicação OSI de sete camadas (ISO 7498), as quais são apresentadas brevemente na figura 1:

FIGURA 1 – Padrão OSI de sete camadas (ISO 7498).

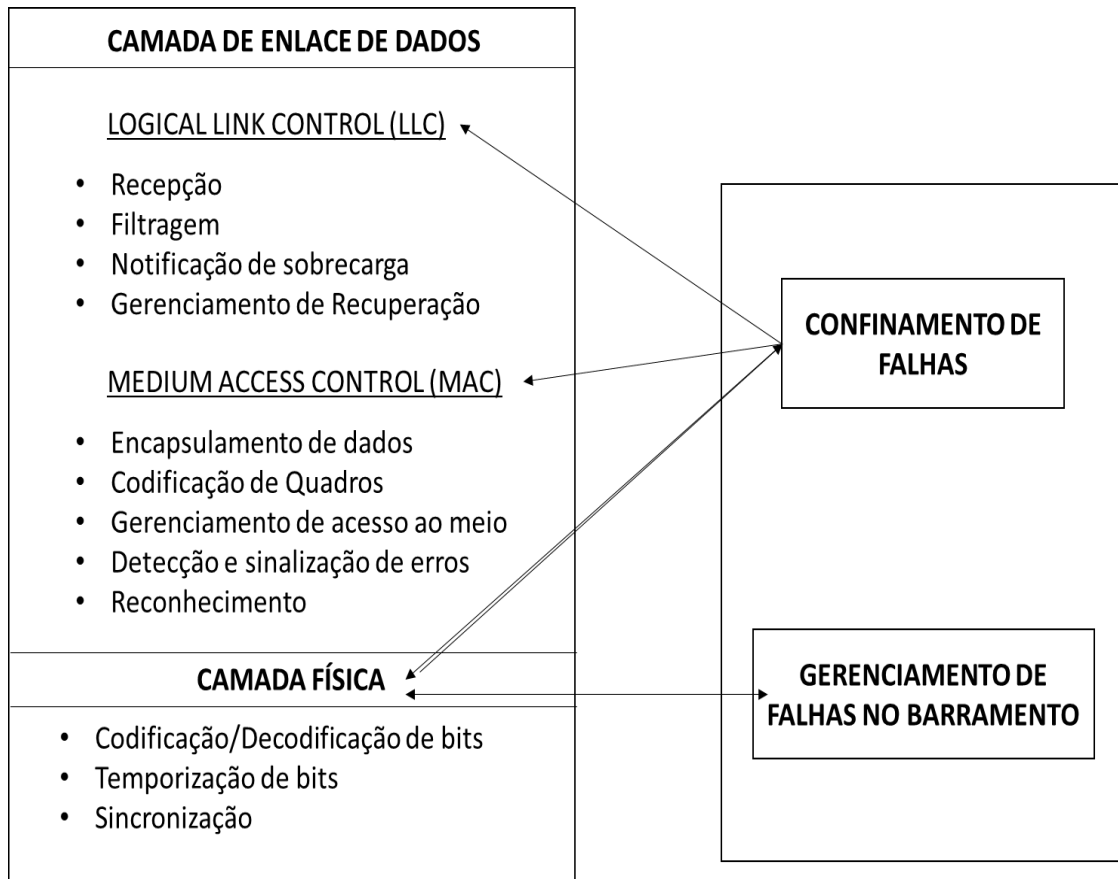


Fonte: O autor (2018).

Com o objetivo de se obter transparência no projeto e flexibilidade na implementação, o CAN foi dividido em duas diferentes camadas, obedecendo ao modelo OSI/ISO: Camada de Enlace de Dados (*Data Link Layer*) e a Camada Física (*Physical Layer*). Por sua vez, a camada de Enlace de Dados foi dividida em duas outras subcamadas: *Logical Link Layer (LLC) sublayer* e *Medium Access Control (MAC) sublayer*.

A camada física trata de aspectos como a temporização e codificação de bits, além da sincronização dos mesmos. A subcamada de acesso ao meio (MAC) representa o núcleo do protocolo, recebendo mensagens de/para a LLC, sendo responsável pela divisão das mensagens em quadros (*framing*), arbitragem, reconhecimento, detecção e sinalização de erros, sendo gerenciada por uma entidade chamada Confinamento de Dados (*Fault Confinement*), que é o mecanismo responsável pela distinção entre falhas temporárias ou permanentes. Já a subcamada lógica (LLC), está relacionada com a filtragem das mensagens, a notificação de *overload* e o gerenciamento da recuperação.

FIGURA 2 - Arquitetura do CAN, divisões em camadas e as respectivas tarefas



Fonte: Adaptado de HUBERT (2011).

As mensagens trafegam por um barramento com um formato fixo, porém podem possuir tamanhos diferentes. O campo de identificação da mensagem é quem define a prioridade da mesma no momento de acesso ao barramento. Cada sensor ou atuador conectado ao barramento, deverá transmitir em velocidade pré-definida, por exemplo, em redes automotivas uma velocidade comum é de 500kbps a 1Mbps. Em sistemas mais complexos onde existem diversos barramentos CAN, podem haver outras configurações de taxa de transmissão diferentes entre si.

Quando o barramento está livre, ou seja, quando houver os bits de fim de mensagem e sincronização, o acesso é liberado a qualquer nó. Por meio do método da arbitragem, é feita a seleção das mensagens, logo a de maior prioridade é recebida imediatamente e as próximas somente quando o barramento estiver livre novamente. Esta escolha é feita por meio da comparação bit-a-bit do campo de identificação das mensagens, caso um nó transmita um bit recessivo (1 lógico) e o barramento estiver com um bit dominante (0 lógico), o nó com bit recessivo será abortado e terá de aguardar até que ocorra a liberação do barramento para uma nova transmissão. Trabalhando desta forma, garante-se que não há perda de dados nem tempo durante a transmissão. Outra situação é quando duas mensagens possuem identificadores dominantes, logo a forma de desempate é pelo quadro de dados que tiver maior prioridade.

Os métodos de detecção de erros têm como finalidade garantir o máximo de segurança durante a transmissão. A partir do Confinamento de Falhas (*Fault Confinement*), erros temporários e permanentes podem ser diferenciados, dessa forma se um erro permanente é identificado o mesmo é retirado do sistema.

Em teoria, o número de conexões, também chamados de nós, é ilimitado em um barramento CAN. No entanto, como trata-se de um canal bidirecional de carregamento de bits, na prática as conexões são limitadas pelos tempos de atraso e cargas elétricas no barramento. A transmissão das mensagens pode ser feita de diferentes modos, porém o mais comum é através de par trançado ou fibra ótica.

3.1 MECANISMO DE ACESSO AO MEIO - CSMA/CR

Uma importante característica do CAN é que o protocolo é baseado na técnica CSMA/CR (*Carrier Sense Multiple Access/Collision Resolution*) de detecção e resolução de colisões no acesso ao meio de transmissão. Ou seja, os nós que têm acesso ao barramento possuem prioridade definida, com isso o status do barramento é verificado e em seguida é iniciada a transmissão das mensagens. O campo de identificação é quem define a prioridade de cada mensagem, dessa forma é possível fazer o escalonamento do acesso ao meio. Durante a recepção da mensagem transmitida por um determinado nó, é feito o teste de aceitação e verificação de interesse da mensagem para outros nós através do identificador. Em caso de colisão, o valor do identificador que define a prioridade na transmissão, um módulo com mensagem de menor prioridade interrompe sua transmissão para dar vez a outro módulo de maior prioridade. Este processo é feito por meio da comparação bit a bit não destrutivo, ou lógica “E” por fios, onde cada bit é comparado e o dominante sobrepõe o recessivo.

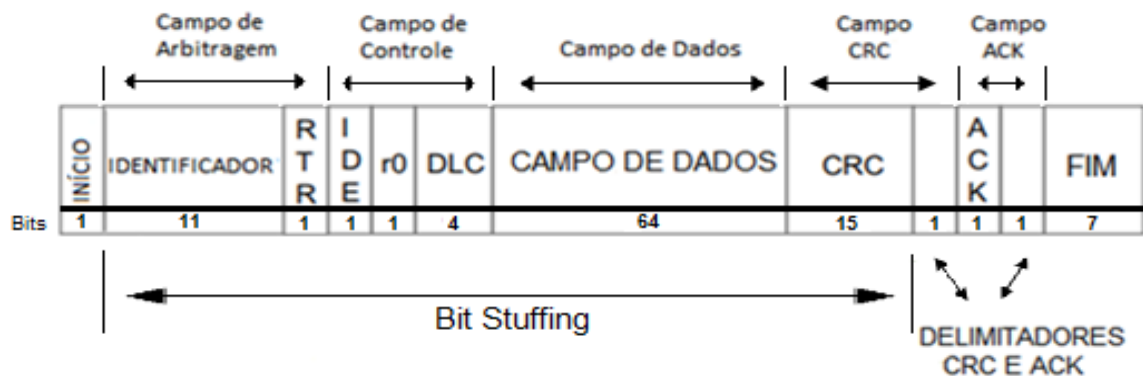
3.2 MENSAGENS DO CAN

3.2.1 FORMATO DA MENSAGEM

Para que ocorra a comunicação dentro da rede CAN é necessário que mensagens sejam enviadas e recebidas pelos diversos nós do barramento, as quais são transmitidas por quadro de bits, que por sua vez, são compostos por campos de bits, que possuem determinadas funções dentro do quadro.

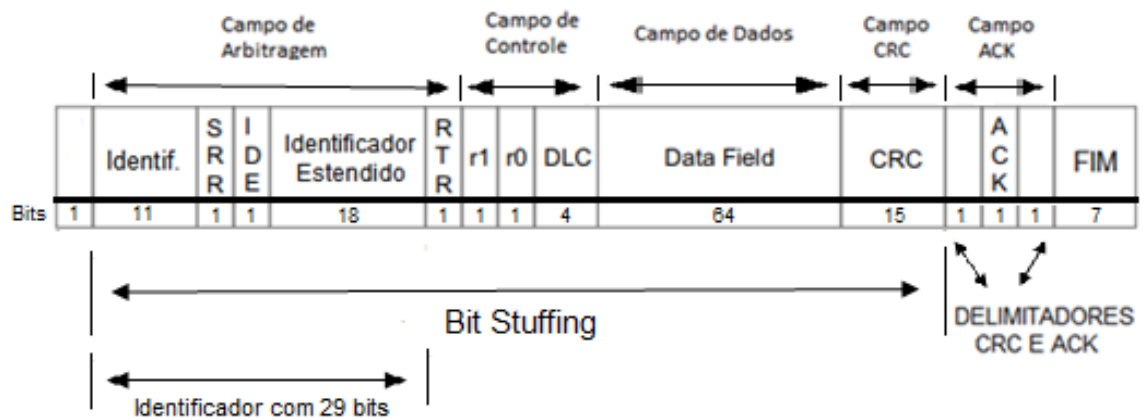
Segundo (Sousa, 2002), as mensagens podem ser divididas em dois tipos: sendo CAN A (*standard frame*) com identificador de 11 bits ou CAN B (*extended frame*) com identificador composto por 2 campos, um com 11 bits e outro com 18 bits totalizando 29 bits. As figuras 3 e 4 ilustram como são os formatos de uma mensagem CAN e o quadro para transmissão.

FIGURA 3 – Formato de mensagem com identificador padrão a partir do quadro de bits do CAN



Fonte: O autor (2018).

FIGURA 4 – Formato de mensagem com identificador estendido



Fonte: O autor (2018).

3.3 TIPOS DE QUADROS

São utilizados diversos tipos de quadros para envio de dados, requisição de dados, propagação de erros e mensagens de sobrecarga (*overload*).

3.3.1 QUADRO DE DADOS

Existem variações na configuração dos quadros de dados nas versões *Standard* (Quadro Padrão) e *Extended* (Quadro Estendido) conforme visto nas figuras 3 e 4 acima. A partir delas são definidos os campos:

- Campo de Início: composto por um bit dominante, o qual é responsável pela primeira sincronização entre os nós;
- Campo de Arbitragem: campo responsável pelo arbítrio das mensagens recebidas. É composto pelo identificador e o bit RTR (Requisição de Transmissão Remota), caso possua bit 0 é denominado quadro de dados, caso possua bit 1 será do tipo quadro remoto;

- Campo de Controle: possui 6 bits. O primeiro, chamado de IDE (Identificador de Extensão) informa se o quadro é padrão (bit dominante) ou estendido (bit recessivo). O bit r0 foi designado para aplicações em versões futuras do CAN. Os últimos quatro bits denominados DLC (*Data Length Code*), são responsáveis por indicar o número de bytes no campo de dados;
- Campo de dados: é formado por 8 bytes (64 bits), onde o bit mais significativo do primeiro byte é o primeiro a ser transmitido. Estes bytes podem representar a informação completa, parte da informação ou várias informações;
- Campo de Verificação de Redundância Cíclica (CRC): composto por 15 bits para o código de detecção de erros e possui um bit recessivo que delimita este campo. O código produz uma redundância capaz de permitir que o receptor identifique que houve um erro durante a transmissão da mensagem;
- Campo de Confirmação (ACK): composto por dois bits, um bit chamado ACK, o qual é recessivo, sobrescrito por bits dominantes caso a mensagem tenha sido transmitida com sucesso. Assim como no CRC, o campo ACK termina com um bit recessivo;
- Campo de Fim de Quadro: indica o fim de quadro, através de 7 bits recessivos.

3.3.2 QUADRO DE REQUISIÇÃO REMOTA

Quando existe a necessidade de requisição de dados em um nó, o mesmo envia um quadro remoto, o qual executa a tarefa de forma eficiente e rápida. Este tipo de quadro é idêntico ao quadro de dados, porém não é composto pelo Campo de Dados em sua estrutura e o bit RTR é recessivo.

3.3.3 QUADRO DE ERRO

O quadro de erro é produzido por qualquer nó que identifique erro no barramento e é formado por dois campos. Um campo sinalizador de erro, chamado de *Error Flag*, e um campo delimitador, chamado *Error Delimiter*.

Quando ocorre a detecção de erro, o nó envia um quadro composto por 6 bits, o qual transgredir a regra do *Bit Stuffing* fazendo com o que os outros nós também transmitam um quadro de erro.

O campo sinalizador é composto por 6 bits, de forma que podem ser todos recessivos tanto para erros passivos quanto para erros ativos. Já o campo delimitador é composto por 8 bits recessivos.

3.3.4 QUADRO DE SOBRECARGA

Tem como finalidade sinalizar que o receptor atual precisa de mais algum tempo até o recebimento do próximo quadro, em geral por falta de processamento ou memória.

Análogo ao quadro de erros, este quadro consiste de dois campos, o campo de sinalizador (*overload flag*) possui 6 bits dominantes e o campo delimitador (*overload delimiter*) possui 8 bits recessivos.

3.4 DETECÇÃO E TRATAMENTO DE ERROS

A estrutura do protocolo CAN é composta por mecanismos de detecção e tratamento de erros, a fim de garantir que um dispositivo com falha não provoque erro no barramento.

Cada nó possui dois contadores: o REC (*Receiver Error Counter*), de 7 bits, o qual guarda a contagem de erros de recepção com falha e o TEC (*Transmitter Error Counter*) responsável pela contagem de erros de transmissão de mensagens. Pode-se dividir em dois níveis de tipos de erros.

Nível de Bit possui dois tipos de erros possíveis:

- Erro de bit (*Bit Error*): um bit transmitido por um determinado nó, é diferente do bit monitorado no barramento pelo mesmo nó. Caso essa diferença ocorra no campo de arbitragem não é considerado erro;
- Erro de Codificação (*Stuff Error*): são transmitidos mais de cinco bits de mesmo nível, o que viola a regra de *Bit Stuffing*. Caso uma mensagem seja recebida com pelo menos seis bits seguidos iguais, ocorreu algum erro no barramento;

Nível de mensagem possui três tipos de erros possíveis:

- Erro de Formação (*Form Error*): bit que tem valor pré-definido, porém durante a transmissão apresenta valor incorreto;
- Erro no CRC (*CRC Error*): quando o cálculo do CRC recebido em um nó difere do valor transmitido junto aos dados, apresentando erro no valor de um ou vários bits da mensagem;
- Erro de Confirmação (*ACK Error*): o nó que transmitiu a última mensagem não recebe confirmação dos outros nós, ou seja, nenhum módulo recebeu a mensagem ou a mensagem transmitida estava corrompida.

3.5 FILTROS DE MENSAGEM

Os filtros de mensagens têm como principal função separar as mensagens que interessam e que não interessam a um determinado nó. Com isso, o processador de mensagens é otimizado de forma que não é necessária a interrupção do processamento. Existem dois tipos de filtros de mensagens:

- Filtro de aceitação: possui valores pré-definidos que são comparados com os valores de identificadores de outras mensagens no barramento para que se determine se a mensagem será ou não validada para transmissão;
- Máscara: permite definição de quais bits de identificadores de mensagens do barramento serão comparados com os valores de filtros de aceitação, para determinar se a mensagem será ou não carregada pelo nó.

3.6 PADRÕES DE PROTOCOLOS DE ALTO NÍVEL

Segundo (HUBERT, 2001) o CAN por padrão, define somente a camada física e a camada de enlace de dados, ou seja, é especificado apenas o transporte de pequenos pacotes de dados entre diferentes pontos de um meio comum de comunicação.

Para que fosse possível gerenciar a comunicação dentro do protocolo CAN, foi necessário a implementação dos Protocolos de Alto Nível, termo derivado do modelo OSI e seus sete níveis. Os protocolos operam na camada de aplicação do modelo OSI, dentre suas principais funções podemos citar:

- Inicialização de componentes do sistema;
- Distribuição dos identificadores das mensagens entre os diversos nós;
- Tradução dos conteúdos dos quadros de dados;
- Gerenciamento do *status* do sistema.

De acordo com a necessidade das empresas, foram implementados variados protocolos conforme tabela abaixo:

TABELA 2 – Principais padrões baseados no CAN

PADRÃO	DESCRIÇÃO
CANopen	Desenvolvida por um grupo de empresas, sendo uma das mais completas e diversificadas do mercado. Utilizada no mercado automobilístico, engenharia médica e principalmente em automação industrial.
DeviceNet	Desenvolvida pela Allen Bradley, sua principal aplicação é na automação industrial.
CAN Kingdom	A empresa sueca KVASER em conjunto com o grupo CAN Textile Users Group desenvolveram o protocolo para a automação fabril.
OSEK/VDX	Originado de um projeto conjunto de empresas da área automotiva, protocolo de padrão aberto para sistemas de controle distribuído em veículos.
SAE J1939	Desenvolvido para aplicação em veículos de terra de grande porte como ônibus tratores, caminhões e máquinas agrícolas.
DIN 9684	Atende as normas alemãs DIN, utilizado em aplicações agrícolas. Baseado no CAN 2.0A adotou e influenciou algumas especificações do padrão ISO.
ISO 11783	Implementado em redes de máquinas e implementos agrícolas, o qual possui variadas especificações iguais ao padrão SAE J1939 e ao DIN 9684
NMEA 2000	A National Marine Electronic Association desenvolveu este padrão para redes em dispositivos em embarcações navais. Tem como base o padrão ISO 11783 e SAE J1939.

FONTE: O autor (2018).

3.7 APLICAÇÕES

A partir dos conceitos apresentados anteriormente, torna-se interessante utilizar uma rede CAN quando temos ambientes hostis ou que possuam a necessidade de transmissão de mensagens com prioridade e a possibilidade de serem transmitidas em tempo real. Contudo, é possível aplicar CAN em redes comuns onde vários microcontroladores e sensores estejam interligados. Dessa forma, observamos que a rede CAN está presente em variados setores, em especial os que seguem abaixo.

3.7.1 APLICAÇÕES AGRÍCOLAS

O avanço da tecnologia no campo contribuiu para que os equipamentos se tornassem cada vez mais interligados, com isso surgiu a necessidade da criação de um padrão para a interconexão dos equipamentos agrícolas, o ISO 11783, sendo implementado no mercado o mínimo do padrão como ISOBUS. Este padrão ainda é recente, o qual se encontra em fase de desenvolvimento por vários fabricantes internacionais. No Brasil, foi criado um grupo chamado “Força Tarefa ISOBUS” a fim de ajudar as pesquisas já em andamento em território nacional para que o padrão seja utilizado.

3.7.2 APLICAÇÕES AUTOMOTIVAS

Foi o setor responsável pela criação da rede CAN, logo diversas aplicações utilizam as vantagens do sistema para melhorar as funções dos veículos. Um exemplo de atuador que pode ser beneficiado com o uso da rede CAN são os freios ABS, os quais enviam e recebem informações sobre o funcionamento global do sistema de sensores/atuadores, com isso pode ser utilizado usando uma estratégia mais adequada de acionamento. Também é utilizada em toda a comunicação entre vidros elétricos, interruptores de portas, iluminação externa, regulagens de bancos entre outras funções. Outra vantagem são os sistemas de diagnóstico de componentes e sensores do veículo, que também são implementados utilizando CAN.

3.7.3 APLICAÇÕES COMERCIAIS

Embora não seja comum o uso em aplicações comerciais, dada a grande disponibilidade de componentes, com baixo custo e boa aceitação em outras áreas, tornou-se uma alternativa atrativa entre as tecnologias de redes que competem ao setor comercial. Um bom exemplo são os projetos em automação residencial e predial, onde pode ser utilizada a rede CAN para a interligação do controle de abertura de portões e portas, ventilação, estado dos elevadores, iluminação, detectores de fumaça, entre outras. A grande maioria das aplicações comerciais, baseiam-se na utilização dos protocolos de alto nível CANopen.

3.7.4 APLICAÇÕES INDUSTRIAIS

Por ser uma área a qual exige alta robustez e resistência a interferências eletromagnéticas, diversas aplicações utilizam a rede CAN. Por ser um ambiente onde podem existir diversos motores elétricos e equipamentos de microondas de alta potência, a interferência eletromagnética pode causar falhas e até mesmo acarretar o fim da operação. O principal protocolo utilizado na indústria é o DeviceNet, pois foi criado e mantido como uma camada de aplicação padrão, para produtos industriais.

3.8 DESENVOLVIMENTO

Para o desenvolvimento da bancada didática foram levados em consideração alguns aspectos relacionados a definição do hardware e software a serem utilizados. Sendo assim, foi possível buscar as opções que melhor atendessem as especificações com os componentes existentes no mercado nacional. Dentro dos parâmetros de livre escolha temos como exemplo, velocidade de comunicação, tamanho do barramento e número de nodos. Já na camada física e de componentes eletrônicos, os microcontroladores, *Shields* transceptores e controladores, cabos e conectores, sensores automotivos.

Ao implementar um padrão baseado no CAN, três módulos são elementares:

- Controlador CAN: módulo principal de controle e implementação do protocolo;
- Transceptor CAN: tem como função adaptar os níveis de tensão entre circuito do nó e barramento CAN.
- Sistema Computacional Central: módulo de implementação dos protocolos de alto nível. Composto por CPU, memória, interface compatível com o controlador CAN e os demais sensores e atuadores.

Com isso, os controladores e transceptores são considerados elementos de extrema importância para uma rede CAN, pois definem a taxa de transferência de dados, número de nós, imunidade a interferências, detecção, sinalização e distinção de erros, custo e robustez. Segundo as normas da ISO, recomenda-se que mesmo em caso de um dos dois fios do barramento esteja rompido ou em curto-circuito com a fonte ou à terra, os transceptores sejam desenvolvidos para que a comunicação continue. Estes módulos podem ser encontrados no mercado como CI's ou podem ser desenvolvidos com componentes discretos. Já os controladores CAN são responsáveis pela definição dos diferentes tipos de quadros, processo de arbitragem, pelo gerenciamento do envio e recebimento de mensagens, além do processo de identificação e sinalização de erros.

3.9 DOCUMENTAÇÃO DO PROJETO

3.9.1 ESCOPO

As atividades do projeto podem ser divididas em três grupos principais:

- a) Pesquisa e Documentação;
- b) Programação;
- c) Construção da bancada.

3.9.2 CRONOGRAMA

A fim de controlar o desenvolvimento das atividades e também melhor administrar o tempo disponível entre a inscrição do projeto e sua entrega, foi desenvolvido um cronograma (Anexo A), contendo as macro etapas das atividades, um registro das datas de desenvolvimento das mesmas, bem como observações a respeito

de situações ocorridas durante o projeto. De acordo com o cronograma observa-se que algumas das atividades foram desenvolvidas simultaneamente, otimizando o tempo disponível entre as etapas.

3.9.3 ORÇAMENTO

Também foi desenvolvido um orçamento base, para o controle do custo total do projeto. O orçamento inicial estimado (Anexo B) e o final realizado (Anexo C) ficaram muito próximos, o que representa um ponto positivo no que diz respeito ao planejamento do projeto. É válido observar que nos dois casos o item de maior impacto no orçamento é a mão-de-obra, uma vez que foram investidas aproximadamente 80 horas de desenvolvimento na confecção da bancada e também no desenvolvimento da programação dos controladores Arduino.

3.9.4 QUALIDADE

A qualidade do projeto foi focada em obter sucesso nas premissas do projeto definidas inicialmente. Apesar da não possibilidade de desenvolver todas as funcionalidades previstas no início do projeto (como a aquisição de sinal dos sensores de rotação e temperatura do motor), em parte devido à complexidade e peculiaridade da arquitetura veicular adotada no projeto (linha Peugeot PSA) em parte devido à dificuldade em adquirir informações detalhadas sobre os dispositivos automotivos utilizados, obteve-se um bom resultado no projeto, podendo ser empregado um tempo para desenvolvimento do acabamento estético (pintura personalizada e prototipagem de elementos em manufatura aditiva). Sendo assim, do ponto de vista da qualidade, a bancada didática apresenta bom acabamento estético e também funcionalmente.

3.9.5 RECURSOS HUMANOS

No levantamento de recursos humanos consta o realizador do projeto e os consultores externos que auxiliaram por meio de fornecimento de informações ou desenvolvimento de atividades específicas (como pintura e programação).

DESENVOLVEDOR DO PROJETO

Bruno Cueto

CONSULTORES EXTERNOS

Eletroeletrônica Automotiva e Programação

Profº Valério Mendes Marochi

Pintura

Profº Luiz Carlos Konrath

4. ARQUITETURA ELETROELETRÔNICA

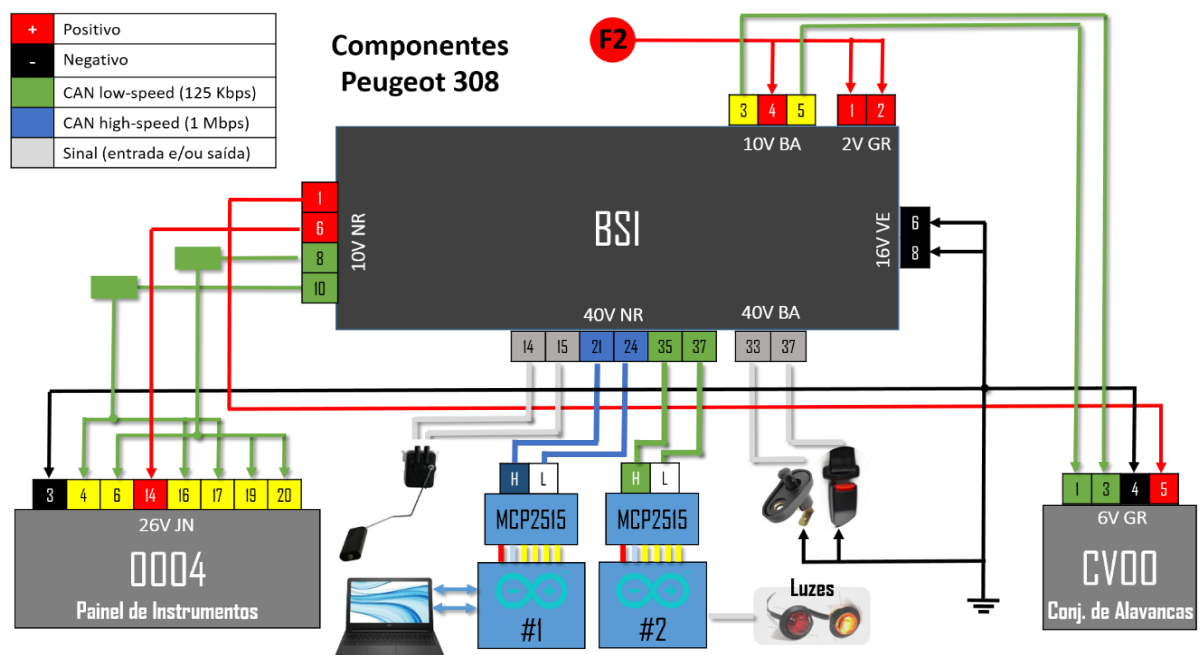
A arquitetura eletroeletrônica da bancada é composta por diversos componentes conforme a tabela 3, tendo em vista que cada um possui uma função dentro do sistema do veículo, entretanto compartilham informações entre si por meio da rede CAN.

TABELA 3 – Tabela de componentes eletroeletrônicos

Item	Descrição
1	MCP 2515
2	Placas Arduino UNO
3	Painel de Instrumentos Peugeot 308
4	Conjunto de alavancas Peugeot 308
5	BSI - Unidade de Bordo
6	Sensor de nível de combustível
7	Módulo Relé
8	Leds de Sinalização
9	Interruptor de porta
10	Fonte de alimentação

FONTE: O autor (2018)

FIGURA 5 – Diagrama elétrico da interligação dos componentes eletroeletrônicos da bancada

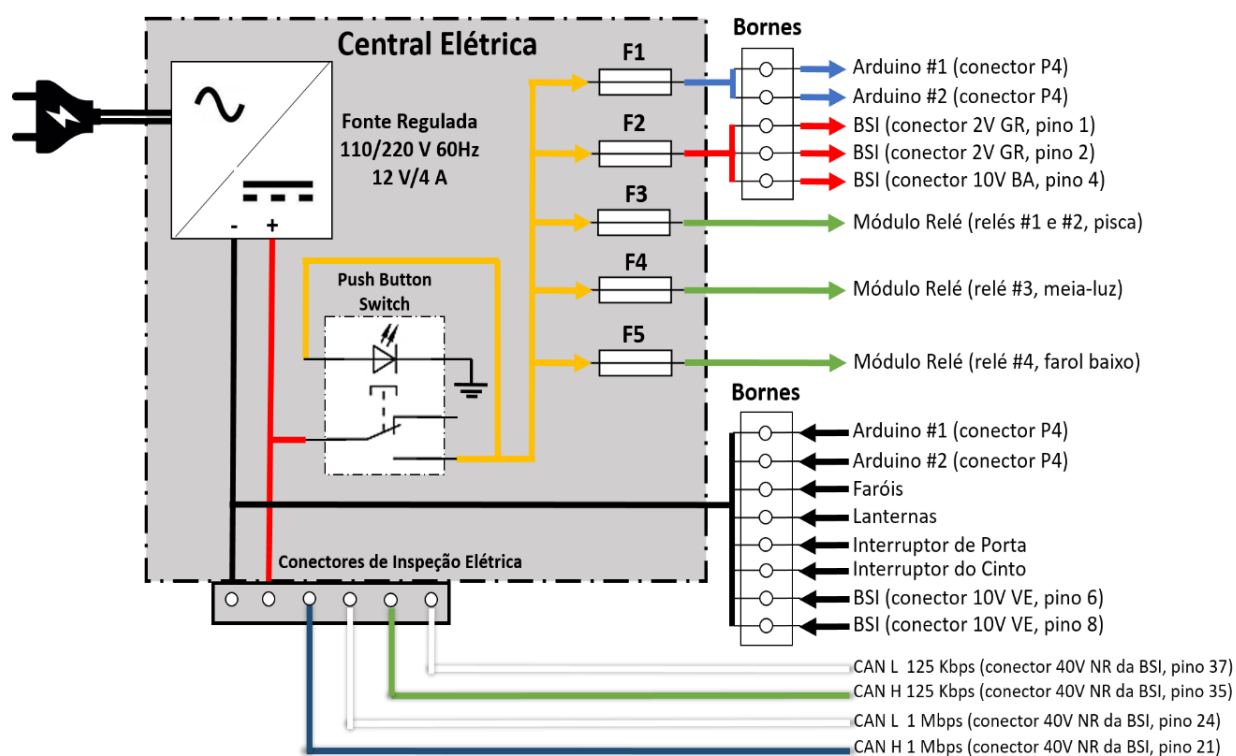


Fonte: O autor (2018)

O diagrama elétrico da interligação entre a BSI, painel de instrumentos, conjunto de alavancas, placas Arduino, os módulos MCP2515 e sensores é exemplificada na figura 5, o qual foi baseado nos diagramas elétricos da fabricante Peugeot, onde são representados por números os pinos para a ligação dos sinais de alimentação positivo, negativo, CAN *High* e CAN *Low*.

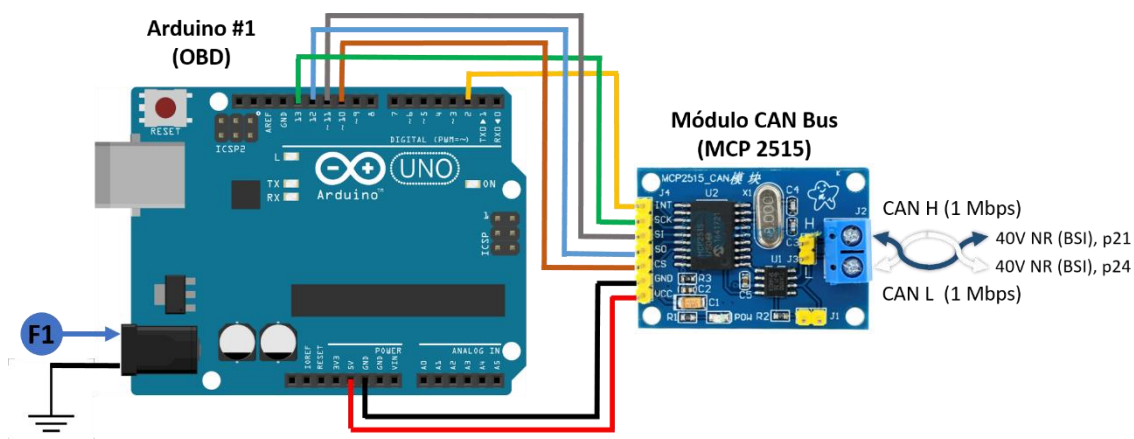
Com o intuito de facilitar o entendimento e facilitar o acesso aos conectores de inspeção, foi desenvolvida na região da bateria do carro, uma área que compõe pinos de inspeção da rede CAN H e CAN L, além dos sinais de alimentação. Também foi inserido um botão liga/desliga do sistema, juntamente com fusíveis de proteção individuais de cada circuito.

FIGURA 6 – Diagrama Elétrico de Alimentação da bancada



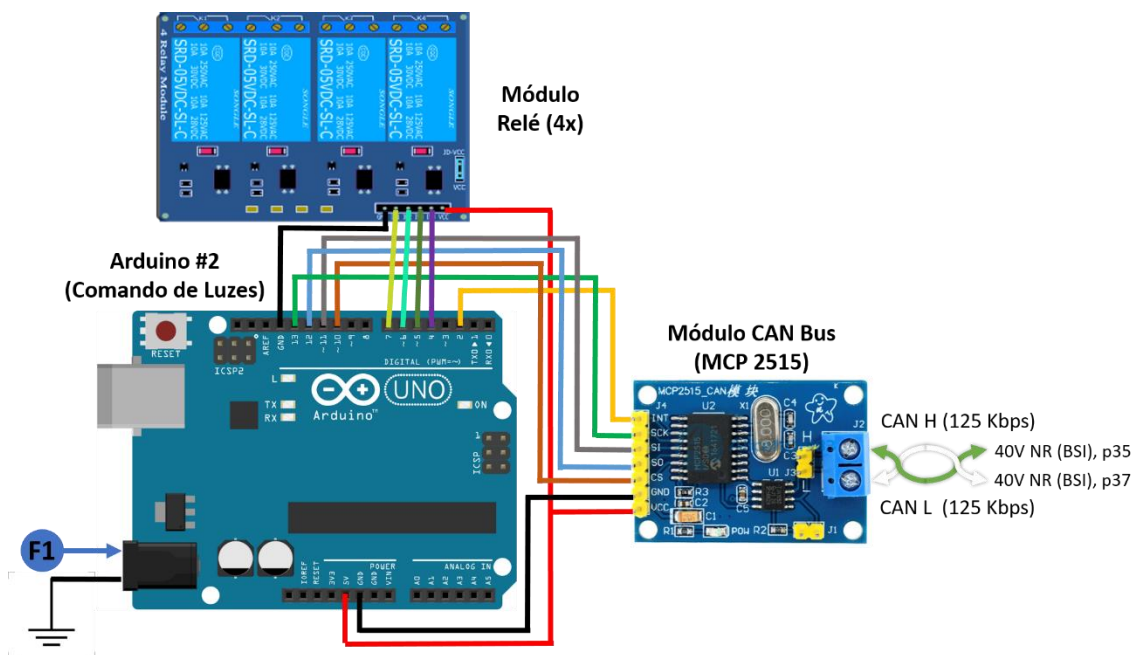
Fonte: O autor (2018)

FIGURA 7 – Diagrama Elétrico do Arduino 1 e MCP2515 de leitura de dados CAN



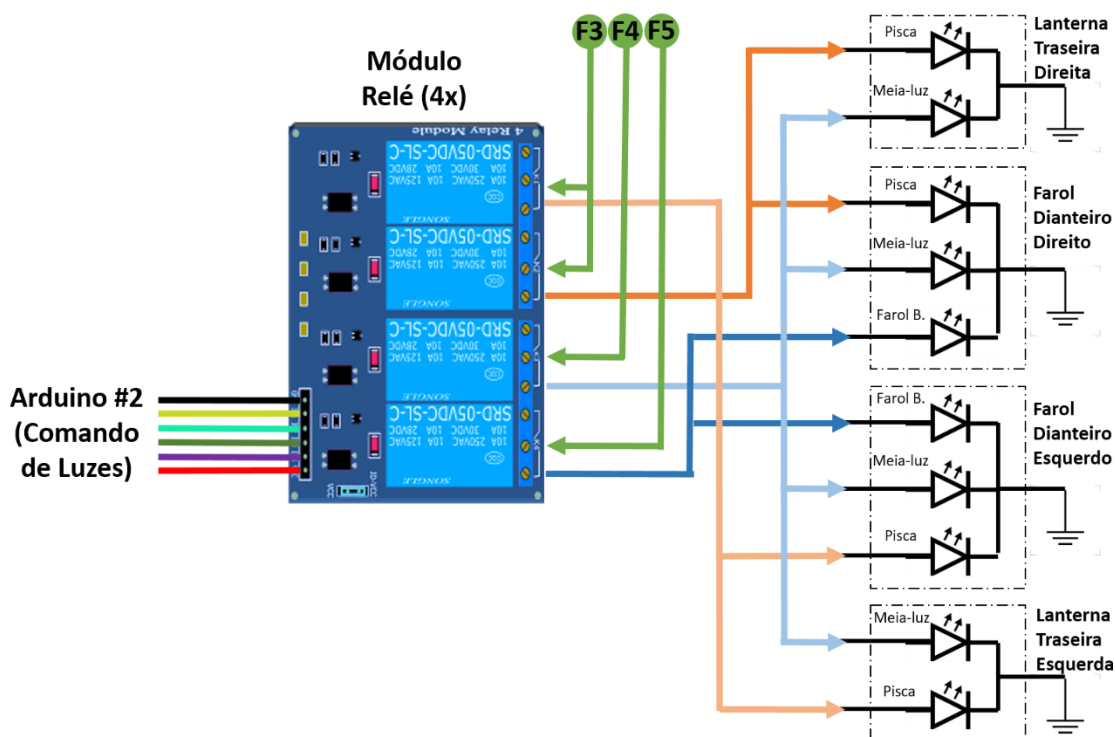
FONTE: O autor (2018)

FIGURA 8 – Diagrama Elétrico do Arduino 2 e MCP2515 do comando de iluminação externa



FONTE: O autor (2018)

FIGURA 9 – Diagrama elétrico do módulo relé, faróis e lanternas



FONTE: O autor (2018)

FONTE DE ALIMENTAÇÃO

A fonte chaveada de alimentação é um produto de alta eficiência utilizado na alimentação de circuitos eletrônicos. Possui um circuito controlador interno, chaveando a corrente, de modo a alternar a passagem de energia, estabilizando a tensão de saída e garantindo maior estabilidade para o funcionamento dos circuitos. O modelo em questão é bivolt (127V ou 220V) e tem capacidade de alimentar circuitos que atinjam correntes de até 10A, ou seja, é possível alimentar vários circuitos com a mesma fonte. Outra característica importante é a sua construção, pois tem peso bem inferior ao de um transformador com as mesmas especificações, possui blindagem externa (caixa de metal) o que possibilita evitar interferências de outros circuitos, além de em alguns pontos são utilizados alumínio para uma melhor dissipação de calor.

FIGURA 10 – Fonte de alimentação chaveada

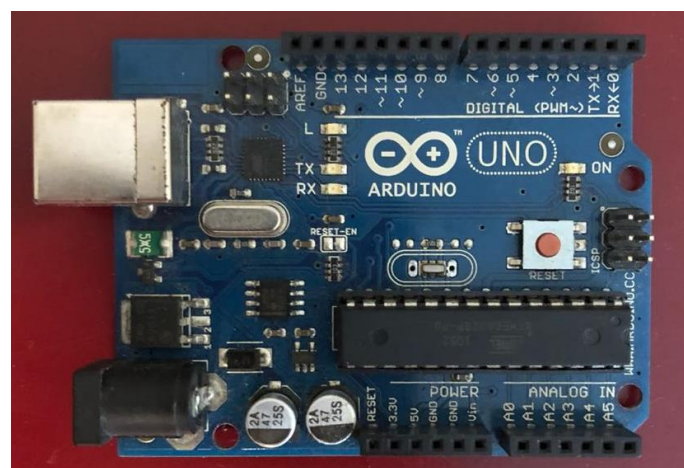


FONTE: O autor (2018)

ARDUINO UNO

O Arduino UNO é uma plataforma com hardware livre e com uma placa baseada no microcontrolador ATmega328, com suporte de entrada/saída embutido, linguagem de programação padrão, em geral C/C++. Tem como principal objetivo ser uma ferramenta de baixo custo, criar ferramentas acessíveis e ser flexível de modo a atender os mais variados tipos de projetos. Possui 14 pinos digitais de entrada/saída (6 podem ser utilizados como saídas PWM), 6 entradas analógicas, um pino com frequência de 16 MHz ressonador cerâmico, conexão USB, um conector de alimentação e um botão de reset.

FIGURA 11 – Arduino UNO



FONTE: O autor (2018)

MÓDULO CAN BUS MCP2515

O Módulo CAN BUS MCP2515 é uma placa eletrônica capaz de permitir a integração de um microcontrolador à rede CAN, apresentando excelente desempenho. É composto por um controlador CAN MCP2515 e um transceptor TJA1050, onde a interface de comunicação entre o módulo e o microcontrolador é SPI (*Serial Peripheral Interface*). Fisicamente possui sete pinos especificados na placa e um par de bornes de conexão. É importante salientar que a conexão é garantida por meio dos pinos CAN *HIGH* e CAN *LOW*, permitindo desta forma a troca de informações entre os módulos conectados ao barramento. O dispositivo é constituído basicamente por três blocos:

- **Módulo CAN:** processa todas as funções para garantir a transmissão e recepção de mensagens no barramento. A transmissão é iniciada usando bits de registro de controle através da interface SPI ou usando os pinos de habilitação para transmissão. Qualquer mensagem detectada no barramento passa por uma verificação de erros e em seguida, comparada com filtros definidos pelo usuário para saber se será enviada a um dos dois buffers de recebimento.
- **Controle Lógico:** responsável pelo controle da configuração e operação do MCP2515, transmitindo estas informações aos outros blocos. Possui um pino de interrupção utilizado para determinar quando uma mensagem válida é recebida. Existem ainda outros três pinos disponíveis para iniciar transmissão imediata, de uso opcional.
- **Protocolo SPI:** Comandos SPI padrão para ler e escrever mensagens, além de comandos especializados.

A priorização dos dados é feita internamente no módulo MCP2515 e por meio da biblioteca `mcp_can.h`, onde ocorre a arbitragem dos identificadores das mensagens, os quais são comparados bit a bit e o que possuir maior nível de prioridade será transmitido primeiro no barramento.

Especificações:

- Modelo: MCP2515 CAN;
- Tensão de Operação: 5VDC;
- Corrente de Trabalho: 5mA (1 μ A em espera);
- Campo de Dados: 0-8 bytes;
- CAN v2.0b em até 1 Mb/s;
- Dimensões: 40x28x11mm

FIGURA 12 – Módulo CAN BUS MCP2515.



FONTE: O autor (2018)

PAINEL DE INSTRUMENTOS E ALAVANCA DE SETAS PEUGEOT 308

O painel de instrumentos tem como função indicar ao condutor as principais informações do veículo tais como velocidade, rotações por minuto, temperatura da água, nível de combustível, indicadores de mal funcionamento, travamento das portas. Já as alavancas de setas são responsáveis pela indicação das setas, ligar e desligar os faróis e os limpadores dos vidros. A bancada didática é composta pelo painel de instrumentos e conjunto de alavanca de setas do Peugeot 308 conforme imagens abaixo:

FIGURA 13 – Painel de instrumentos do Peugeot 308.



FONTE: O autor (2018).

FIGURA 14 – Alavanca de setas do Peugeot 308.



FONTE: O autor (2018).

BSI - Boîtier de Servitude Intelligent

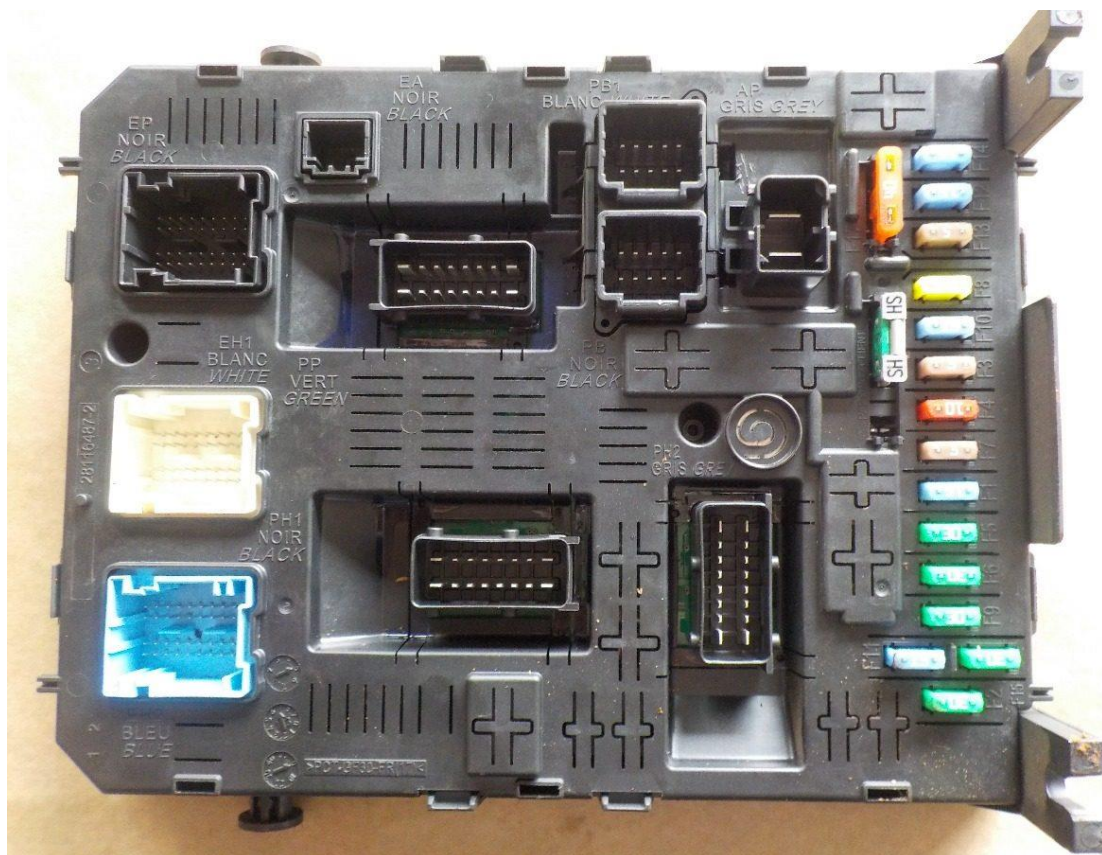
É conhecida como Unidade de Bordo do Veículo, pois integra todos os sistemas CAN do veículo. Caracterizada como central de distribuição elétrica, localizada no interior do veículo (sob o painel na região do motorista), tendo como funções principais:

- Centralizar e distribuir as alimentações elétricas dos sistemas e dispositivos do veículo (carroceria);

- Funcionar como gateway da rede CAN, centralizando e distribuindo todas as mensagens dos diversos subsistemas (carroceria, conforto e trem de força);
- Controlar os sistemas interligados via CAN, possibilitando funções como o modo ECO (hibernação da Rede);
- Disponibilizar acesso aos equipamentos de diagnóstico com o veículo, por meio do modulo OBD.

Para os veículos mais antigos da linha PSA, as centrais em rede são ligadas em série, ou seja, se uma central for removida do Sistema toda a rede para de funcionar.

FIGURA 15 – BSI, unidade de bordo do Peugeot 308.



FONTE: O autor (2018).

PORTA FUSÍVEL

O porta fusível tem como função a de proteger o circuito, ou seja, garantir que os componentes não sejam danificados caso ocorra alguma falha. O modelo utilizado na bancada foi o fusível tipo lâmina, pois é o modelo utilizado nas aplicações automotivas. Possui elemento condutor protegido por capa termoplástica de alta resistência térmica e a mesma varia de cor, a fim de indicar o valor de capacidade de corrente do fusível.

FIGURA 16 – Porta fusíveis automotivo



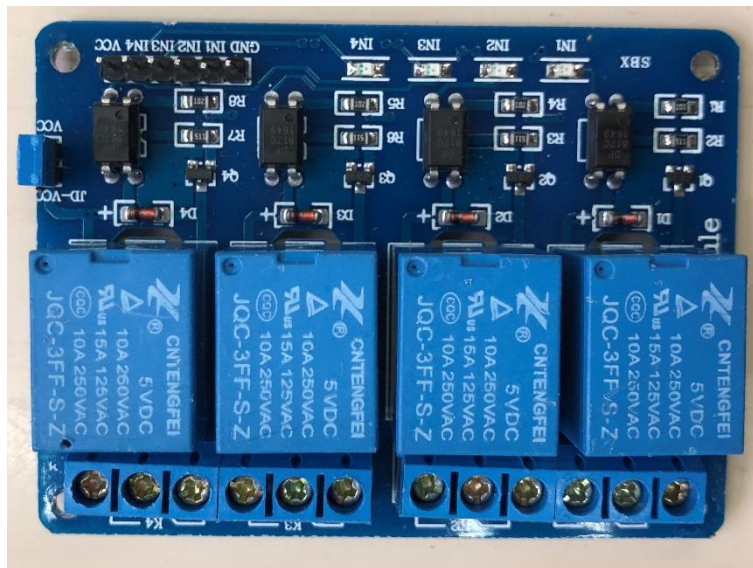
FONTE: O autor (2018)

MÓDULO RELÉ

O módulo relé de 4 canais é uma excelente alternativa para utilização com placas Arduino, visto que são feitos para que haja perfeita comunicação entre ambos, dispensando a necessidade de adicionar outros circuitos. Seu funcionamento inicia quando recebe tensão de 5V pelo comando do Arduino, assim fazendo o chaveamento para Normal Aberto, Normal Fechado ou neutro.

- Tensão de operação da bobina: 5VDC;
- Tensão de sinal: 5VCD (IN1);
- Tensão de comutação: 250VAC ou 24VDC;
- Corrente de comutação: 10A;
- Corrente típica de cada relê: 15-20mA;
- Modelo do Relê: SRD-05VDC-SL-C;
- Tempo de resposta: 5-10ms;
- LED's de indicação de funcionamento;
- Dimensões: 75mm x 55mm x 18mm.

FIGURA 17 – Módulo Relé 4 canais para Arduino



FONTE: O autor (2018)

INTERRUPTOR DE PORTA

O interruptor de porta utilizado no painel é de aplicação universal, possibilitando a regulação de altura do interruptor em relação à superfície de contato que acionará o mesmo. Seu funcionamento é em geral associado ao aterramento de circuitos de alimentação de lâmpadas de teto condicionado ao fechamento do contato com o negativo quando da abertura da porta.

FIGURA 18 – Interruptor de porta



FONTE: O autor (2018)

LEDS DE SINALIZAÇÃO

Os LEDs de sinalização foram utilizados para simular os faróis e setas do veículo. O modelo em questão é da marca TMH.

FIGURA 19 – LEDs de sinalização



FONTE: O autor (2018)

SENSOR DE NÍVEL DE COMBUSTÍVEL

Os sensores de nível de combustível automotivos possuem duas peças principais: o Flutuador, conhecido como boia, que mede o nível de combustível no tanque e o outro é o medidor do painel, o qual indica o nível analogicamente ao condutor. O funcionamento deste sensor é baseado na mudança da posição do flutuador, onde existe um contato deslizante de resistência variável, ou seja, quando estiver na posição mais elevada sua resistência é muito baixa fazendo com que a corrente seja maior, indicando que o tanque está cheio. À medida que o combustível é consumido e a posição do flutuador vai abaixando, a resistência aumenta fazendo com que a corrente diminua até que fique vazio. Dado que este tipo de sensor é um produto de baixo custo, seu nível de precisão é muito baixo.

FIGURA 20 – Sensor de nível de combustível automotivo



FONTE: O autor (2018)

MONTAGEM DA BANCADA

O início da montagem da bancada foi na semana do dia 26 de março de 2018, onde idealizou-se a alocação dos componentes na posição simulando a de um veículo real, representado na figura 21.

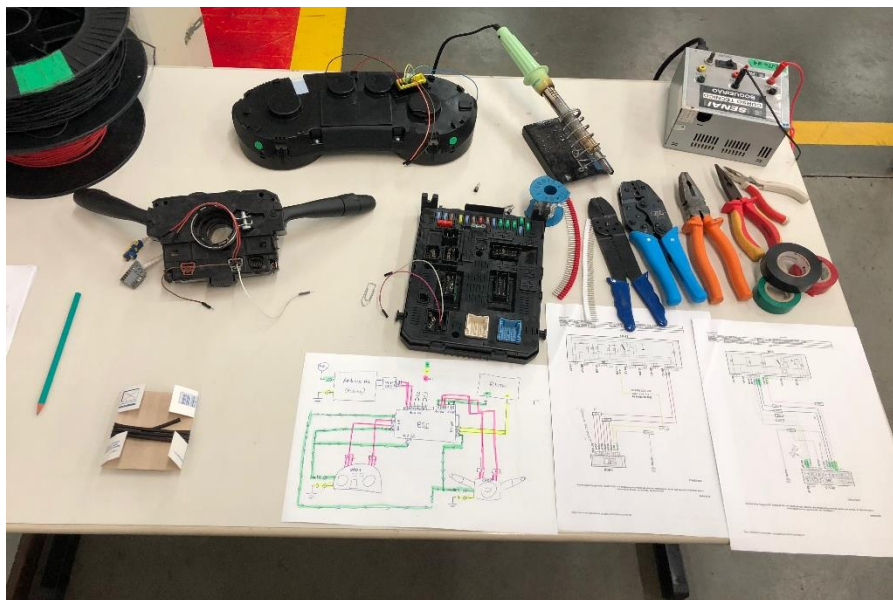
FIGURA 21 – Primeiro esboço da posição dos componentes da bancada



FONTE: O autor (2018)

Na figura abaixo, temos os diagramas elétricos do Peugeot 308, os quais foram estudados para que fosse possível iniciar as ligações para os testes de funcionamento do o conjunto de alavancas, painel de instrumentos, BSI (Unidade de Bordo do veículo).

FIGURA 22 – Componentes e diagramas elétricos do Peugeot 308



FONTE: O autor (2018)

Este estudo dos diagramas foi iniciado no dia 18 de maio de 2018, onde foram feitas as ligações necessárias para os testes de funcionamento, obteve-se sucesso conforme figura 23.

FIGURA 23 – Primeiro teste de funcionamento do painel de instrumentos e BSI



FONTE: O autor (2018)

No dia 18 de maio iniciou-se a fase de fixação dos componentes e ajustes, seguindo a ideia inicial da posição dos componentes simulando um veículo real conforme figura 24.

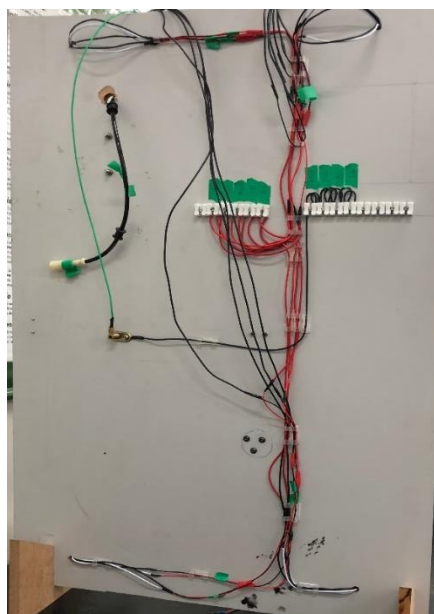
FIGURA 24 – Fixação de componentes na chapa de MDF



FONTE: O autor (2018)

No dia 6 de junho, iniciou-se o roteamento do chicote que interliga a BSI aos demais componentes veiculares e outros componentes inseridos na bancada conforme figura 25.

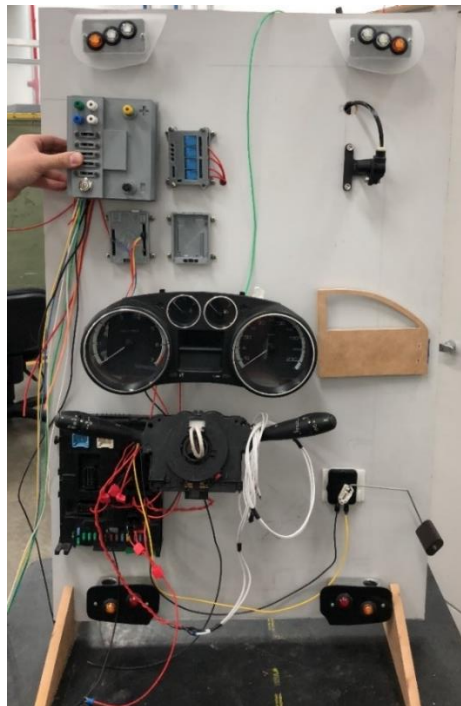
FIGURA 25 – Montagem do chicote na parte traseira da bancada



FONTE: O autor (2018)

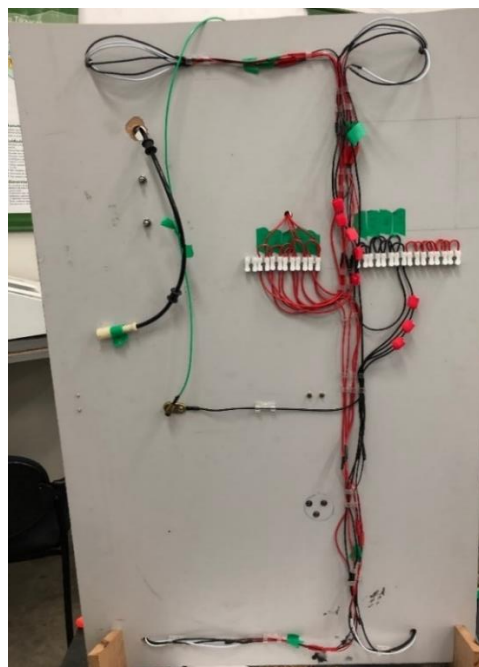
Na semana do dia 11 de junho, entrou na fase final da montagem da bancada, onde todos os componentes foram fixados em suas posições definitivas conforme figuras 26 e 27.

FIGURA 26 – Posição final de cada componente



FONTE: O autor (2018)

FIGURA 27 – Posição final do roteamento do chicote



FONTE: O autor (2018)

Na semana do dia 11 de junho a bancada foi desmontada para dar início a fase de pintura, realizada pelo consultor externo Profº Luiz Carlos Konrath, e acabamentos finais conforme figura 28.

FIGURA 28 – Pintura da chapa de MDF



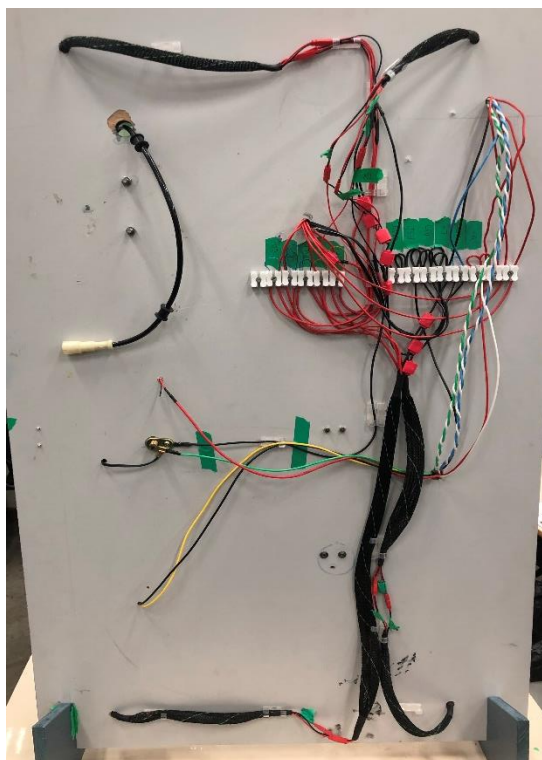
FONTE: O autor (2018)

FIGURA 29 – Bancada didática em funcionamento



FONTE: O autor (2018)

FIGURA 30 – Chicote finalizado



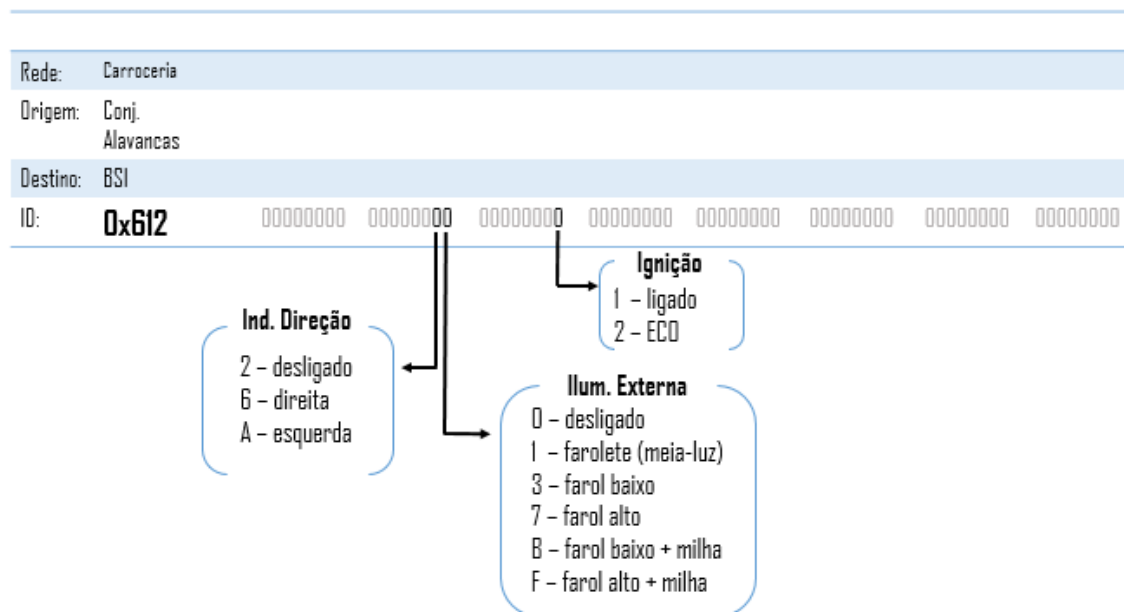
FONTE: O autor (2018)

4.1 DESENVOLVIMENTO DO SOFTWARE

Para a aplicação da bancada as plataformas de desenvolvimento utilizadas necessitam de firmware que possibilite a leitura das mensagens CAN via interface SPI, utilizando um módulo de comunicação específico para protocolo CAN (MCP2515). Houve a necessidade da utilização das bibliotecas específicas do módulo e também da comunicação SPI. Além da leitura dos dados CAN, se faz necessário a conversão desses dados para saídas digitais de sinal do Arduino. Outro fator importante é a necessidade de conhecer a taxa de transmissão da rede dos dispositivos veiculares (painel e conjunto de alavancas, BSI) para que fosse possível a leitura dos dados. Como função adicional do sketch, foi desenvolvido um programa com filtros de identificadores de mensagem, a fim de facilitar o monitoramento de um sinal específico, a partir desta função e através de técnica empírica, a descoberta de diversas funções do veículo tais como:

Através do filtro para o identificador 0x612, durante os testes empíricos em um Peugeot modelo 307, algumas funções de iluminação externa foram encontradas pela variação de bits das mensagens recebidas pela BSI, conforme indicado na figura 31.

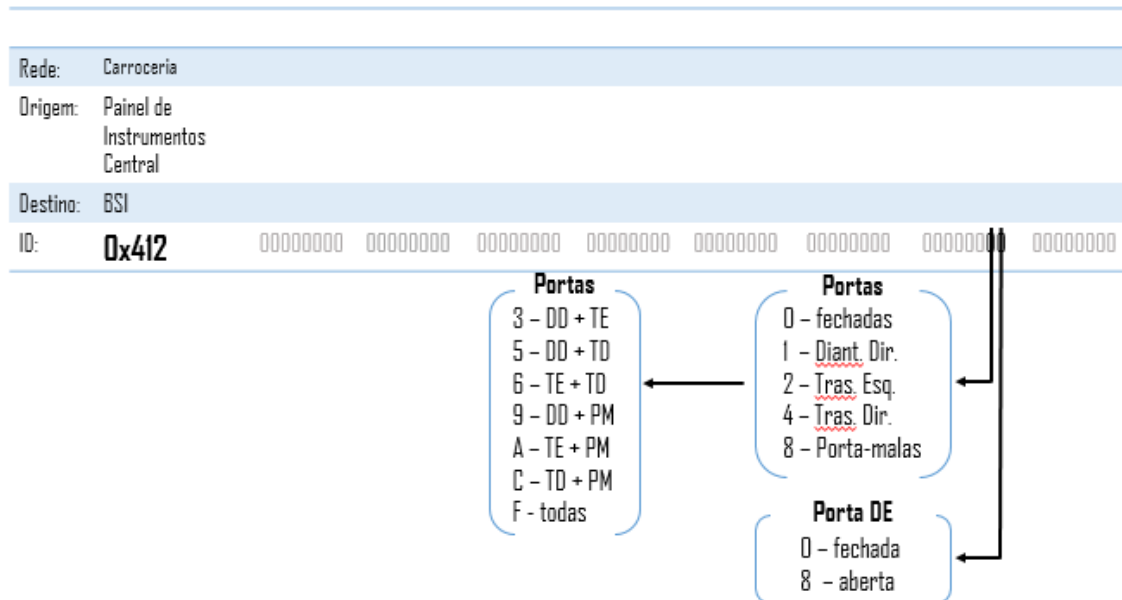
FIGURA 31 – Funções de iluminação externa através do conjunto de alavancas



FONTE: O autor (2018)

Como função adicional, nos testes empíricos no Peugeot 307 foi encontrada variação de mensagens para o identificador 0x412, o qual é responsável pela abertura de portas, conforme figura 32.

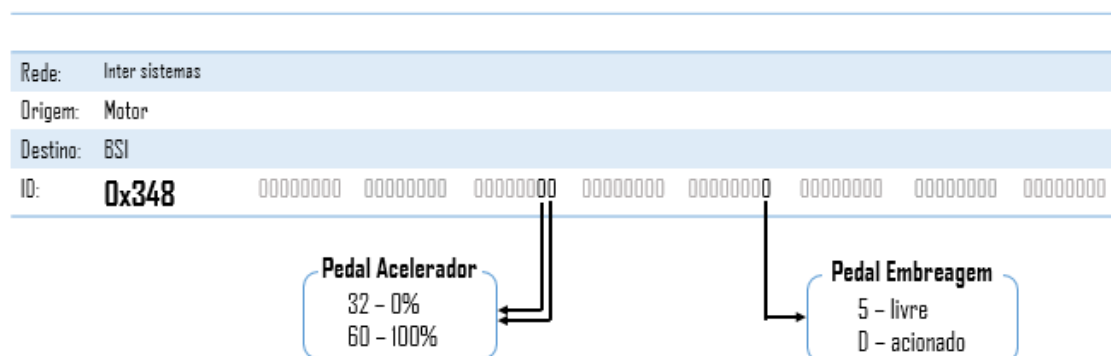
FIGURA 32 – Funções de abertura de portas



FONTE: O autor (2018)

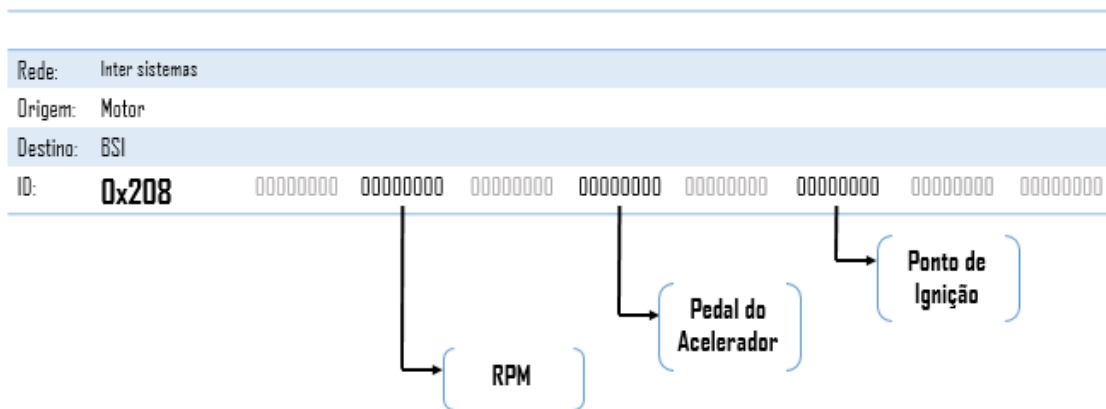
Embora não tenha sido possível adicionar o módulo de injeção eletrônica do motor, durante os testes no Peugeot 307 foram encontradas algumas funções relacionadas ao motor, entre elas estão pedal de acelerador, pedal embreagem e RPM.

FIGURA 33 – Funções de posição de pedal de acelerador e embreagem



FONTE: O autor (2018)

FIGURA 34 – Funções da central do motor do Peugeot 307



FONTE: O autor (2018)

Para a criação da comunicação CAN entre os dispositivos foram utilizadas algumas bibliotecas específicas para que ocorresse a troca de mensagens CAN. Entre elas está a biblioteca da interface de comunicação serial SPI.h e a biblioteca mcp_can.h do MCP2515. Para garantir a comunicação SPI, foi escolhido o pino 10 do Arduino para ligação com o pino CS do MCP2515. Também foi definida uma variável para armazenamento de identificadores chamado de rxId, conforme figura abaixo.

FIGURA 35 – Declaração das bibliotecas CAN e variáveis

```
//Arduino atuando como PSF1 (central de alimentação do cofre do motor)
// Teste de recebimento de mensagem e saída de alimentação para conjunto de relés

#include <SPI.h>
#include <mcp_can.h>

#define rele1 PD4      //seta E
#define rele2 PD5      //seta D
#define rele3 PD6      //meia-luz
#define rele4 PD7      //farol baixo

bool rele1_state = false; //define estado do relé 1
bool rele2_state = false; //define estado do relé 2
bool rele3_state = false; //define estado do relé 3
bool rele4_state = false; //define estado do relé 4

long unsigned int rxId;      //variável para armazenar o ID

unsigned long rcvTime;

unsigned char len = 0;      //
unsigned char buf[8];      //

const int SPI_CS_PIN = 10;
```

FONTE: O autor (2018)

Em um veículo da linha PSA, a iluminação externa é comandada pela BSI juntamente com um módulo chamado PSF1. Como não foi possível adquirir este módulo, foi implementado um programa em outro Arduino UNO em conjunto com outra placa MCP2515 a fim de substituir o módulo PSF1.

FIGURA 36 – Biblioteca de identificadores de mensagens

```
// Biblioteca de ID's mensagens:

unsigned long int ilumID = 0x612; // ID PSA (indicadores de direção e faróis)
unsigned char piscoff[8] = {73, 20, 01, 00, 00, 00, 00, 00};
unsigned char piscesq[8] = {73, A0, 01, 00, 00, 00, 00, 00};
unsigned char piskdir[8] = {73, 60, 01, 00, 00, 00, 00, 00};
unsigned char meialuz[8] = {73, 21, 01, 00, 00, 00, 00, 00};
unsigned char meiaesq[8] = {73, A1, 01, 00, 00, 00, 00, 00};
unsigned char meiadir[8] = {73, 61, 01, 00, 00, 00, 00, 00};
unsigned char luzbaix[8] = {73, 23, 01, 00, 00, 00, 00, 00};
unsigned char luzbesq[8] = {73, A3, 01, 00, 00, 00, 00, 00};
unsigned char luzbdir[8] = {73, 63, 01, 00, 00, 00, 00, 00};
```

FONTE: O autor (2018)

FIGURA 37 – Definição dos pinos de acionamento dos relés

```
MCP_CAN CAN(SPI_CS_PIN); // Set CS pin

void setup()
{
    Serial.begin(230400);
    //Definição dos pinos de acionamento dos relés:
    pinMode(rele1, OUTPUT);
    pinMode(rele2, OUTPUT);
    pinMode(rele3, OUTPUT);
    pinMode(rele4, OUTPUT);

    while (CAN_OK != CAN.begin(CAN_125KBPS)) // velocidade da CAN (low-speed) = 125Kbps (carroceria)
    {
        Serial.println("CAN BUS Module Failed to Initialized");
        Serial.println("Retrying...");
        delay(200);
    }
    Serial.println("CAN BUS Module Initialized!");
    Serial.println("Time\t\tID\tByte0\tByte1\tByte2\tByte3\tByte4\tByte5\tByte6\tByte7");
    digitalWrite(rele1, HIGH);
    digitalWrite(rele2, HIGH);
    digitalWrite(rele3, HIGH);
    digitalWrite(rele4, HIGH);
}
```

FONTE: O autor (2018)

Após a confirmação que existem mensagens disponíveis no barramento, foi desenvolvido um filtro para identificadores, onde escolhe-se o valor para o rxId, desta forma somente mensagens com identificadores com o valor escolhido irão aparecer no leitor serial. Na figura abaixo, foi escolhido como exemplo o identificador 0x612.

FIGURA 38 – Leitura das mensagens e posição da chave de seta

```
//Leitura da mensagens:
rxId= CAN.getCanId();
CAN.readMsgBuf(&len, buf);
if(rxId == 0x612){
    if(&len, buf[1] == 0x20){
        digitalWrite(rele1, HIGH);
        digitalWrite(rele2, HIGH);
        digitalWrite(rele3, HIGH);
        digitalWrite(rele4, HIGH);
        Serial.println("iluminação desligada");
    }
    if(&len, buf[1] == 0xA0){
        pisca();
        digitalWrite(rele2, HIGH);
        digitalWrite(rele3, HIGH);
        digitalWrite(rele4, HIGH);
        Serial.println("Seta esquerda");
    }
    if(&len, buf[1] == 0x60){
        digitalWrite(rele1, HIGH);
        pisca();
        digitalWrite(rele3, HIGH);
        digitalWrite(rele4, HIGH);
        Serial.println("Seta direita");
    }
    if(&len, buf[1] == 0x21){
        digitalWrite(rele1, HIGH);
        digitalWrite(rele2, HIGH);
        digitalWrite(rele3, LOW);
        digitalWrite(rele4, HIGH);
        Serial.println("Meia-luz");
    }

    if(&len, buf[1] == 0xA1){
        pisca();
        digitalWrite(rele2, HIGH);
        digitalWrite(rele3, LOW);
        digitalWrite(rele4, HIGH);
        Serial.println("Meia-luz + Seta esquerda");
    }
    if(&len, buf[1] == 0x61){
        digitalWrite(rele1, HIGH);
        pisca();
        digitalWrite(rele3, LOW);
        digitalWrite(rele4, HIGH);
        Serial.println("Meia-luz + Seta direita");
    }
    if(&len, buf[1] == 0x23){
        digitalWrite(rele1, HIGH);
        digitalWrite(rele2, HIGH);
        digitalWrite(rele3, HIGH);
        digitalWrite(rele4, LOW);
        Serial.println("Luz baixa");
    }
    if(&len, buf[1] == 0xA3){
        pisca();
        digitalWrite(rele2, HIGH);
        digitalWrite(rele3, HIGH);
        digitalWrite(rele4, LOW);
        Serial.println("Luz Baixa + Seta esquerda");
    }
}
```

FONTE: O autor (2018)

FIGURA 39 – Leitura das mensagens e posição da chave de seta

```

        if(&len, buf[1] == 0x63){
            digitalWrite(rele1, HIGH);
            pisca();
            digitalWrite(rele3, HIGH);
            digitalWrite(rele4, LOW);
            Serial.println("Luz baixa + Seta direita");
        }
    }

    void pisca(){
        digitalWrite(rele1, LOW);
        delay(320);
        digitalWrite(rele1, HIGH);
    }
}

```

FONTE: O autor (2018)

4.2 INTEGRAÇÃO DA BANCADA

A integração dos componentes da bancada com a programação empregada nos Arduinos demandou extensa pesquisa e constantes erros e acertos, tendo em vista que a ideia inicial incluía diversos sensores, porém ao decorrer do projeto constatou-se que os componentes do grupo PSA (Peugeot e Citroën) possuem certas particularidades se comparados a outros grupos automotivos. A principal delas é que todos os módulos são gerenciados por uma central eletrônica, também conhecida como BSI, a qual recebe e transmite as informações dos diversos sensores que compõe um veículo. Desta forma, foi possível receber e transmitir mensagens CAN dos módulos do painel de instrumentos, conjunto de alavancas de seta, sensor de nível de combustível, interruptor de porta e do sensor do cinto de segurança. O sensor de rotação do motor e sensor de temperatura do motor foram desconsiderados, tendo em vista que seria necessário adquirir o módulo da injeção eletrônica, o qual possui um valor muito elevado tornando-se inviável. Em contrapartida, foram adicionados outros sensores que também contemplam a rede CAN, sensor de travamento das portas e sensor do cinto de segurança.

A plataforma do grupo PSA é extremamente fechada, desta forma foram empregadas bibliotecas CAN em conjunto com *sketches* em linguagem C++ com o objetivo de receber e transmitir as mensagens. Em paralelo, para a aquisição de dados foi desenvolvido um *sketch* de leitura de identificadores e mensagens, e com ele foi feito o *sniffing* (interceptação e aquisição de dados) em um modelo Peugeot 307 para a

captura de identificadores de mensagens, em especial para a dos sensores mencionados anteriormente. Tratando-se de modelos de veículos da mesma marca, os identificadores e mensagens possuíam os mesmos valores e conteúdo, facilitando o processo de filtragem de mensagens.

5. VALIDAÇÃO EM AULA PRÁTICA

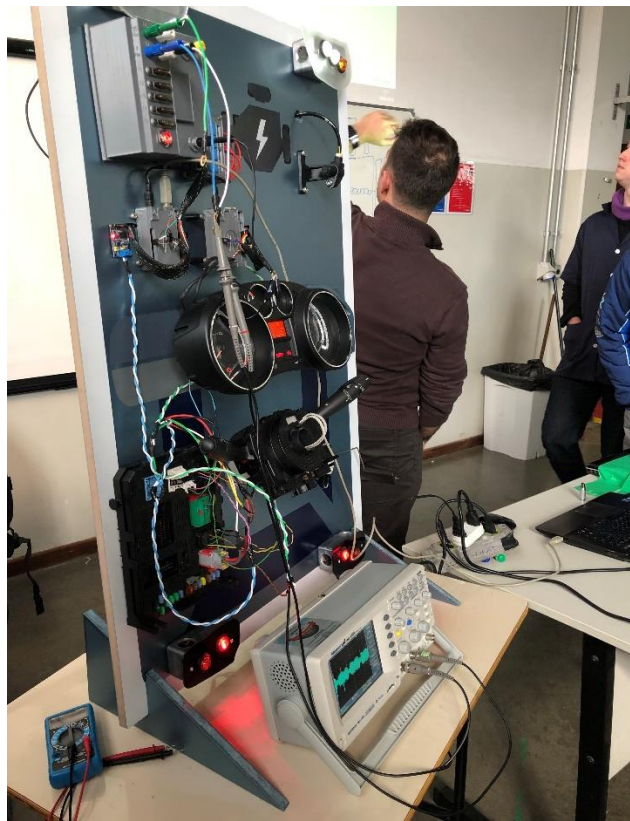
No dia 15 de junho de 2018 foi realizada uma aula expositiva a respeito do protocolo CAN numa unidade do Senai (Boqueirão) referência em ensino automotivo, onde o Prof. Valério Mendes Marochi abordou os conceitos, características e arquitetura das redes veiculares, utilizando a bancada didática produzida por meio deste trabalho para evidenciar na prática como a rede CAN funciona. Na fase final da aula, os alunos demonstraram interesse e tiveram a oportunidade de interagir com a bancada, através de práticas que incluíram visualização gráfica do sinal da rede em osciloscópio e dos dados na tela do computador via Serial Monitor – visualizador da IDE Arduino, através do qual é possível acompanhar o tráfego de mensagens em tempo real e como é o formato da mensagem. Foram registrados alguns momentos da aula conforme as imagens abaixo:

FIGURA 40 – Alunos do curso de eletrônica automotiva em aula prática do SENAI



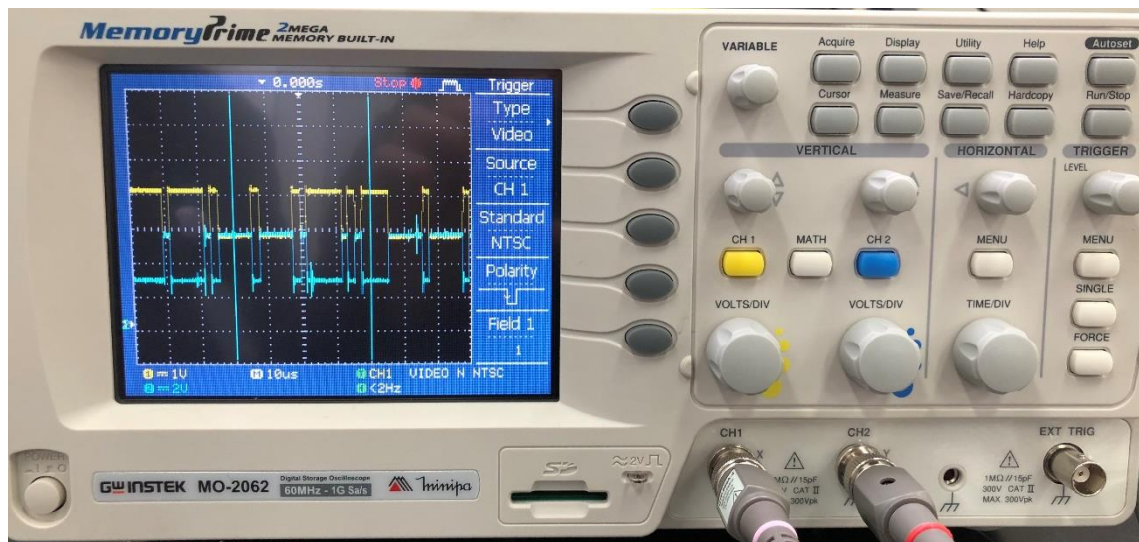
FONTE: O autor (2018)

FIGURA 41 – Alunos do curso de eletrônica automotiva em aula prática do SENAI



FONTE: O autor (2018)

FIGURA 42 – Osciloscópio ilustrando mensagem CAN



FONTE: O autor (2018)

6. CONCLUSÕES E SUGESTÃO PARA TRABALHOS FUTUROS

6.1 CONCLUSÕES

Dado o cronograma enxuto no qual o projeto foi enquadrado – aproximadamente 4 meses, levando-se em consideração a extensão do assunto abordado na proposta de pesquisa para desenvolvimento de uma solução didática e considerando a limitação de investimento financeiro previsto em orçamento, o presente trabalho logrou êxito no que diz respeito ao objetivo principal: desenvolver uma solução didática que possibilite a melhor compreensão do protocolo CAN, bem como suas características e aplicações no âmbito automotivo e fora dele. Dentre as características da bancada didática, podem-se citar: a) permite a visualização da arquitetura física de redes veiculares bem como os componentes que a constituem; b) conta com acessos destinados à captura dos dados transmitidos no barramento através de instrumentos de medição como multímetro e osciloscópio, a fim de facilitar as práticas; c) utiliza-se de uma plataforma de desenvolvimento de baixo custo e fácil implementação, através da qual é possível reconfigurar as funcionalidades e programações do sistema.

6.2 SUGESTÕES PARA TRABALHOS FUTUROS

O tema abrangido por este trabalho abre precedentes para que outros estudantes da área possam dar continuidade ao projeto, tendo em vista que até o momento não se encontra vasto conteúdo a respeito de bancadas didáticas do protocolo CAN. Sendo assim, foram levantadas possíveis melhorias e aprimoramentos.

O projeto foi desenvolvido utilizando o Arduino UNO, que possui uma plataforma aberta e de baixo custo, porém possui certas limitações quando comparado a outras plataformas mais modernas como Raspberry Pi por exemplo, o qual aliado a utilização do Codesys com bibliotecas específicas da rede CAN, pode evoluir ainda mais o resultado do projeto podendo chegar até a simulação de funcionamento do motor.

Os sensores que tiveram de ser retirados do trabalho (sensor de rotação do motor e de temperatura do motor) também ficam como sugestão para implementação, compondo o módulo do motor, sendo um grande desafio para um próximo trabalho.

REFERÊNCIAS

Yang, T.C. (2006). **Networked control system: a brief survey**. **IEEE Proceedings of Control Theory and Applications**, Vol. 153, No 4, July, pp. 403–412.

Hubert, M.K. (2001). **O protocolo CAN como solução para aplicações distribuídas, baseadas em objetos entre PCs e microcontroladores**. Disponível em: <http://www.eletronica.org/arq_artigos/mono-can.pdf>. Acesso em 15 mar. 2018.

Sousa, R.V. (2002). **CAN (CONTROLLER AREA NETWORK): Uma Abordagem para Automação e Controle na Área Agrícola**. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/18/18135/tde-30102007-115747/en.php>>. Acesso em 18 mar. 2018.

Othman, H.F., Y.R Aji, F.T. Fakhreddin and A.R. Al-Ali (2006). **Controller Area Networks: Evolution and Applications**. Proc. of the 2nd International Conference on Information and Communication Technologies, Vol. 2, pp. 3088-3093.

Moyne, J.R. & D.M. Tilbury (2007). **The Emergence of Industrial Control Networks for Manufacturing Control, Diagnostics, and Safety Data**. Proc. of IEEE Technology of Networked Control Systems, Vol. 95, No. 1, pp. 29-47.

BOSCH, Robert. **Manual de Tecnologia Automotiva**. Tradução de Helga Madjderey, Gunter W. Prokesch, Euryale de Jesus Zerbini, Sueli Pfeferman. 25° ed. São Paulo: Edgar Blücher, 2005.

CAPELLI, Alexandre. **Eletrônica automotiva: injeção eletrônica, arquitetura do motor e sistemas embarcados**. 1ª ed., São Paulo, editora Érica, 2010.

CSS ELECTRONICS. **CAN Bus Explained – A Simple Intro (2018)**. [S.n.]. CSS Electronics: Højbjerg, 2018.

HALDERMAN, James D. **Automotive Technology**. Principles, Diagnosis and Service. 4th ed. Upper Saddle River: Prentice Hall, 2012.

MICROCHIP WEBSEMINARS. **Introduction to Controller Area Network**. Microchip Technology Incorporated: [S.I.] 2012.

CORRIGAN, Steve. **Introduction to the Controller Area Network (CAN) – Application Report**. Rev. Ed. Texas Instruments: Dallas, 2016.

ANEXO A – CRONOGRAMA DE EXECUÇÃO DO PROJETO

	FEV	MAR					ABR					MAI					JUN				
Tarefa	23	5~9	12~16	19~23	26~30	2~6	9~13	16~20	23~27	30~04	07~11	14~18	21~25	28~01	04~08	11~15	18~22	25~29			
Estudo do projeto																					
Pesquisa do assunto																					
Levantamento de componentes e materiais																					
Esboço da bancada																					
Testes iniciais com placas MCP2515 e Arduino																					
Programação dos sketches da bancada																					
Desenvolvimento da arquitetura física do sistema CAN																					
Construção da bancada didática																					
Integração dos sistemas da bancada																					
Testes e ajustes finais da bancada																					
Validação da bancada em aula																					
Desenvolvimento do relatório do TCC																					
Entrega da Proposta UFPR (23/02)																					
1ª Avaliação UFPR (19/03)																					
1ª Banca UFPR (18/04)																					
Entrega do Relatório Final (18/06)																					
2ª Banca UFPR (26/06)																					

Matrizes de Responsabilidade e Marcos de Eventos		
		Execução de tarefa
		Deadline estimado
		Deadline de tarefa

ANEXO B – ORÇAMENTO INICIAL DE EXECUÇÃO DO PROJETO

Orçamento Inicial				
Item	Descrição	Valor	Qtde	Total
1	Placas MCP 2515	R\$ 45,00	2,0	R\$ 90,00
2	Painel de Instrumentos Peugeot 308	R\$ 250,00	1,0	R\$ 250,00
3	Conjunto de alavancas Peugeot 308	R\$ 300,00	1,0	R\$ 300,00
4	Sensor de temperatura	R\$ 60,00	1,0	R\$ 60,00
5	Sensor de nível	R\$ 50,00	1,0	R\$ 50,00
6	Sensor de rotação	R\$ 80,00	1,0	R\$ 80,00
9	Fonte de alimentação	R\$ 50,00	2,0	R\$ 100,00
11	Placa Arduino UNO	R\$ 45,00	3,0	R\$ 135,00
12	Eletrocalha	R\$ 20,00	1,0	R\$ 20,00
13	Condutores	R\$ 30,00	1,0	R\$ 30,00
14	Chapa de MDF	R\$ 40,00	1,0	R\$ 40,00
15	Acrílico	R\$ 50,00	1,0	R\$ 50,00
16	Pesquisa	R\$ 25,00	20	R\$ 500,00
17	Programação	R\$ 25,00	30	R\$ 750,00
18	Confecção da bancada	R\$ 25,00	10	R\$ 250,00
19	Desenvolvimento do relatório	R\$ 25,00	60	R\$ 1.500,00
Total				R\$ 4.205,00
Previsão				R\$ 6.000,00

ANEXO C – ORÇAMENTO FINAL DE EXECUÇÃO DO PROJETO

Orçamento Final				
	Materiais	Valor	Qtde	Total
1	MCP 2515	R\$ 45,00	5,0	R\$ 225,00
2	Painel de Instrumentos Peugeot 308	R\$ 300,00	1,0	R\$ 300,00
3	Conjunto de alavancas Peugeot 308	R\$ 250,00	1,0	R\$ 250,00
4	BSI - Unidade de Bordo	R\$ 450,00	1,0	R\$ 450,00
5	Sensor de nível de combustível	R\$ 45,00	1,0	R\$ 45,00
6	Módulo Relé	R\$ 20,00	8,0	R\$ 160,00
7	Leds de Sinalização	R\$ 90,00	1,0	R\$ 90,00
8	Interruptor de porta	R\$ 4,00	1,0	R\$ 4,00
9	Conector 2 vias, macho e fêmea para baterias de LiPo	R\$ 15,00	1,0	R\$ 15,00
10	Fonte de alimentação	R\$ 40,00	1,0	R\$ 40,00
11	Placa Arduino UNO	R\$ 45,00	3,0	R\$ 45,00
12	Push Button Alumínio - Liga/desliga	R\$ 15,00	1,0	R\$ 15,00
13	Impressões 3D	R\$ 50,00	1,0	R\$ 50,00
14	Rolo de filamento PLA prata	R\$ 115,00	1,0	R\$ 115,00
15	Condutores	R\$ 20,00	1,0	R\$ 20,00
16	Conector Sindal de Click	R\$ 25,00	2,0	R\$ 50,00
17	Chapa de MDF	R\$ 25,00	1,0	R\$ 25,00
18	Chapa de Acrílico Transparente	R\$ 65,00	1,0	R\$ 65,00
19	Chapa de Acrílico Fumê	R\$ 65,00	1,0	R\$ 65,00
20	Condulete em Alumínio	R\$ 4,00	4,00	R\$ 16,00
21	Fita isolante, terminais, estanho para solda, lixa	R\$ 40,00	1,00	R\$ 40,00
22	Parafusos, porcas, arruelas	R\$ 20,00	1,00	R\$ 20,00
23	Organizador de Cabos Auto-adesivo	R\$ 15,00	2,00	R\$ 30,00
24	Tarugo de Nylon 60mm torneado	R\$ 60,00	1,00	R\$ 60,00
25	Pesquisa	R\$ 25,00	20	R\$ 500,00
26	Programação	R\$ 25,00	20	R\$ 500,00
27	Confecção da bancada	R\$ 25,00	20	R\$ 500,00
28	Desenvolvimento do relatório	R\$ 25,00	20	R\$ 500,00
			Total	R\$ 4.195,00
			Previsão	R\$ 6.000,00