



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

SÉRGIO RICARDO DE OLIVEIRA BRANDÃO

SISTEMA PARA USO DA TECNOLOGIA CAN EM VEÍCULOS AUTOMOTIVOS

Campina Grande, Brasil

Agosto de 2022

SÉRGIO RICARDO DE OLIVEIRA BRANDÃO

SISTEMA PARA USO DA TECNOLOGIA CAN EM VEÍCULOS AUTOMOTIVOS

Relatório de trabalho de conclusão de curso submetido à Unidade Acadêmica de Engenharia Elétrica na Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica.

Professor Orientador: Raimundo Carlos S. Freire

Campina Grande, Brasil

Agosto de 2022

SÉRGIO RICARDO DE OLIVEIRA BRANDÃO

SISTEMA PARA USO DA TECNOLOGIA CAN EM VEÍCULOS AUTOMOTIVOS

Relatório de trabalho de conclusão de curso submetido à Unidade Acadêmica de Engenharia Elétrica na Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica.

Aprovado em ____ / ____ / ____

Raimundo Carlos S. Freire
Orientador

Convidado

Campina Grande, Brasil

Agosto de 2022

AGRADECIMENTO

Agradeço à minha família por todo suporte e dedicação a este filho que teve tantos percalços dentro da graduação, sempre batalhando e vencendo os desafios pelo, e tão somente, esforço. À Dona Maria de Fátima, minha mãe, e ao Senhor Petrônio Brandão, meu pai, me faltam as devidas palavras para tanta fé em um filho. Para vocês, dedico as minhas lágrimas de agradecimento.

Agradeço à todo o corpo de professores do DEE, meus grandes mestres, em especial ao Professor Raimundo Carlos Silvério Freire, pela imensa gentileza de me conceder uma oportunidade em fazer parte do Laboratório de Metrologia e Instrumentação Científicas – LIMC -, onde tive meu primeiro contato com pesquisas acadêmicas relacionadas à eletrônica e interações com o mundo profissional, me possibilitando o aprendizado de habilidades humanas e técnicas, vitais ao engenheiro, e que hoje, fazem parte do meu repertório.

Agradeço, também, ao Professor Raimundo Carlos S. Freire por todo suporte, apoio e boa vontade enquanto orientador do presente trabalho. Sua paciência e disposição foram de suma importância.

RESUMO

A tecnologia CAN é usada na maior parte da indústria automotiva. Essencialmente é dividida em dois contextos: O protocolo de comunicação e o barramento físico. Surgiu em meados da década de 80, inicialmente, como uma tentativa de substituir a enorme quantidade de cablagem usada nos projetos automotivos para tráfego de dados. O protocolo de comunicação de início foi concebido para um identificador de 11 bits, classificado como CAN Padrão, e posteriormente foi estabelecido um modelo com 29 bit, conhecido como CAN Estendido. Em termos físicos, a nível de circuito eletrônico, é implementado via *chips* e seus circuitos periféricos. Atualmente, a tecnologia segue uma linha de evolução, gerando aperfeiçoamento, novos modelos de tecnologia e aplicabilidades. Este presente trabalho tem por objetivo propor um projeto de placa de circuito impresso que viabilize a aplicação da tecnologia para captura e tráfego de dados em aplicações automotivas.

Palavras-chave: CAN, Microcontrolador, Protocolo de Comunicação, Firmware, Hardware

ABSTRACT

CAN technology is used in most of the automotive industry. Essentially it is divided into two contexts: the communication protocol and the physical bus. It emerged in the mid-1980s initially as an attempt to replace the massive amount of wiring used in automotive designs for data traffic. The communication protocol at first was designed for an 11-bit identifier, classified as Standard CAN, and later a 29-bit model was established, known as Extended CAN. In physical terms, at the electronic circuit level, it is implemented via chips and their peripheral circuits. Currently, the technology follows a line of evolution, generating improvements, new technology models, and applications. This paper aims to propose a printed circuit board project that enables the application of the technology for data capture and traffic in automotive applications.

Keywords: CAN, Microcontroller, Communication Protocol, Firmware, Hardware.

LISTA DE FIGURAS

FIGURA 1: LÓGICA INVERTIDA DO BARRAMENTO CAN.....	15
FIGURA 2: ARBITRAÇÃO NO BARRAMENTO CAN.....	16
FIGURA 3: BARRAMENTO CAN.....	19
FIGURA 4: ESTADOS NO BARRAMENTO CAN.....	20
FIGURA 5: CONECTOR OBD.....	21
FIGURA 6: DIAGRAMA DE BLOCOS DO TJA1050.....	23
FIGURA 7: DIAGRAMA DE BLOCOS DO MCP2515.....	25
FIGURA 8: DIAGRAMA DE BLOCOS DO ATMEGA328P.....	27
FIGURA 9: PINOUT DO LM7805.....	28
FIGURA 10: PINOUT DO LM1117.....	29
FIGURA 11: MODELOS DE OPTOACOPLADORES.....	30
FIGURA 12: FOLHA DOS REGULADORES DE TENSÃO.....	31
FIGURA 13: FOLHA DA CONECTIVIDADE USB.....	32
FIGURA 14: FOLHA DO MÓDULO CAN.....	33
FIGURA 15: FOLHA DO MÓDULO MCU.....	34
FIGURA 16: AVR ISP.....	35
FIGURA 17: FOLHA DO OPTOACOPLADOR.....	35
FIGURA 18: VISTA SUPERIOR.....	36
FIGURA 19: VISTA INFERIOR.....	37
FIGURA 20: RECORTE DO OSCILOSCÓPIO DOS SINAIS CAN.....	37
FIGURA 21: SINAIS CAN TX/RX.....	37

LISTA DE TABELAS

TABELA 1: CAMADAS DA ARQUITETURA PADRÃO DEFINIDA NA ISO 11898 ...	11
TABELA 2: CAN PADRÃO	13
TABELA 3: CAN ESTENDIDO	14
TABELA 4: MAPEAMENTO DE BLOCOS	38

SUMÁRIO

1	Introdução	9
1.1	Objetivos	10
1.2	Metodologias	10
2	Fundamentação Teórica	11
2.1	CAN Padrão e CAN Estendido	12
2.2	Os campos de Bits	12
2.2.1	CAN Padrão	12
2.2.2	CAN Estendido	14
2.3	A Mensagem CAN	15
2.3.1	Arbitragem	15
2.3.2	Tipos de Mensagens	16
2.3.2.1	<i>Data Frame</i>	16
2.3.2.2	<i>Remote Frame</i>	17
2.3.2.3	<i>Error Frame</i>	17
2.3.2.4	<i>Overload Frame</i>	17
2.3.3	<i>Valid Frame</i>	17
2.3.4	Verificação de Erros e Confinamento de Falhas	18
2.4	O Barramento CAN	18
3	O Projeto	20
3.1	A Concepção	20
3.2	Circuitos Integrados	21
3.2.3	Transceptor de Alta Velocidade	22
3.2.3.1	TJA1050	22
3.2.4	Controlador CAN	23
3.2.4.1	MCP2515	24
3.2.5	Microcontrolador	25
3.2.5.1	ATMEGA328P	25
3.3	Circuitos periféricos	27
3.3.1	Regulação de tensão	27
3.3.1.1	LM7805	28
3.3.1.2	LM1117	28
3.3.2	Optoacopladores	29
3.3.3	Filtros de desacoplamento	30
4	Esquemáticos	31

4.1	Folha 01: Reguladores de Tensão	31
4.2	Folha 02: Conectividade USB	32
4.4	Folha 03: Módulo CAN	33
4.5	Folha 04: Módulo MCU	34
4.6	Folha 05: Módulo OptoAcoplador	35
5	Sistema Final	36
6	Considerações Finais.....	38
7	Referências.....	39

1 Introdução

No passado, os fabricantes de automóveis conectavam dispositivos eletrônicos em veículos fazendo uso de uma quantidade volumosa de cabos ponto-a-ponto, que, em grande maioria eram pesados e custosos. Com o decorrer dos anos, à medida que o número de dispositivos eletrônicos aumentava, a indústria constatou ser impraticável o método adotado e novas ideias começaram a ser estudadas sobre o tema.

Em 1980, engenheiros da Bosch estavam estudando os barramentos seriais existentes em relação ao seu possível uso em sistemas automotivos para o usuário comum. Nenhuma das tecnologias existentes à época era capaz de atender as especificações dos engenheiros. Deu-se início ao desenvolvimento de um novo sistema de barramento serial em 1983, que mais tarde seria conhecido como tecnologia CAN.

A nova tecnologia, entre outros fatores, deveria principalmente adicionar novos recursos. Engenheiros da Mercedes também se envolveram na fase de especificações e a Intel, no fornecimento de semicondutores e em 1986 nasceu a tecnologia *Controller Area Network (CAN)*, apresentada no congresso SAE, em Detroit e foi padronizada internacionalmente no início da década de 1990. O protocolo padronizou uma camada física para taxas de bits de até 1Mbit/s.

No início de 2011, a General Motors e a Bosch iniciaram o desenvolvimento de algumas melhorias no protocolo CAN em relação ao maior rendimento. A indústria automotiva sofreu com o *download* de pacotes de software cada vez mais crescentes nas unidades de controle eletrônico (ECU¹), uma vez que a tarefa se tornara extremamente demorada. No intuito de aumentar a velocidade de transmissão do CAN, e dessa forma, a Bosch, apresenta em 2012 uma evolução da tecnologia, chamada CAN FD.

No final de 2018, surge a terceira geração da tecnologia CAN, denominada CAN XL. Entre suas principais características, destaca-se um maior campo de dados destinado ao tráfego de informações.

¹ *Engine Control Unit*. Dispositivo eletrônico que recebe dados do veículo.

1.1 Objetivos

O objetivo deste trabalho de conclusão de curso é apresentar uma proposta de hardware que faça uso da tecnologia CAN a ser utilizado em veículos automotivos e realizar a captura de dados, tais como: Rotações do motor, temperatura, leitura de sensores, dentre outros. Para tal fim, faz-se necessário o estudo da tecnologia, a seleção dos componentes eletrônicos adequados, o desenvolvimento das folhas de esquemáticos e o roteamento do projeto em PCB – *Printed Circuit Board* – em *software*.

1.2 Metodologias

O desenvolvimento do projeto envolve essencialmente cinco etapas.

Na primeira etapa, realizou-se uma revisão bibliográfica sobre a tecnologia CAN, que teve por finalidade entender seu surgimento e sua linha de evolução temporal em termos de hardware e implementação do protocolo, afim de definir as diretrizes iniciais do projeto, como os circuitos integrados, componentes passivos e ativos a serem utilizados.

A segunda etapa é caracterizada pelo estudo, validação e projeto dos circuitos eletrônicos. Os circuitos são definidos em três categorias, essencialmente:

- i. Circuitos de alimentação de tensão.
- ii. Circuitos de filtros eletrônicos.
- iii. Circuitos para a interface da tecnologia CAN.

A terceira etapa é definida como “Integração com software de design de PCBs”. Nela, todos os esquemáticos de circuitos eletrônicos são importados para os softwares de desenvolvimento de PCB – *Printed Circuit Board* -. Este tipo de ferramenta automatiza o design eletrônico para a disposição dos elementos na placa.

A quarta etapa é destinada ao roteamento dos componentes eletrônicos. Na concepção da eletrônica, o roteamento compreende a disposição de todos os componentes eletrônicos do projeto na placa de circuito impresso e as suas devidas interconexões, avaliando as disposições necessárias para mitigar efeitos adversos como interferências eletromagnéticas. Este contexto também envolve a criação de planos de referência por meio de polígonos, possibilitando a referência dos circuitos.

A última fase consiste na finalização do projeto, compreendendo a soldagem dos componentes na placa de circuito impresso e os testes necessários para a validação.

2 Fundamentação Teórica

CAN é definido pela *International Standardization Organization (ISO)* como uma tecnologia de comunicação serial originalmente desenvolvida para a indústria automotiva no intuito de substituir os complexos cabeamentos outrora utilizados. A tecnologia é especificada como sendo de alta imunidade a interferências eletromagnéticas e com capacidade de autodiagnosticar e reparar erros de dados. Estas características conduziram à popularidade do CAN numa imensa variedade de indústrias.

O protocolo de comunicação CAN ISO-11898:2003 descreve como a informação é transmitida entre dispositivos de uma rede. A comunicação real entre dispositivos ligados pelo meio físico é definida por meio da camada física do modelo. A arquitetura ISO 11898 define as duas camadas mais baixas como sendo as camadas de *data-link* e *physical layer* como se nota na tabela 1:

Tabela 1: Camadas da arquitetura padrão definida na ISO 11898

<i>Application Layer</i>	
<i>Data-Link Layer</i>	<i>Logic Link Control</i>
	<i>Medium Access</i>
	<i>Physical Signaling</i>

<i>Physical Layer</i>	<i>Physical Medium Attachment</i>
	<i>Medium-Dependant Interface</i>

Fonte: Autoria própria

2.1 CAN Padrão e CAN Estendido

O CAN, do ponto de vista da tecnologia, possui duas vertentes: o protocolo de comunicação e a camada física. Classicamente, o protocolo de comunicação pode ser aplicado tendo em vista dois modelos: CAN Padrão e CAN Estendido. O protocolo de comunicação CAN é de acesso múltiplo com detecção de colisão e arbitragem sobre prioridade de mensagens. Cada nó do barramento deve esperar por um período de inatividade antes de tentar enviar uma mensagem. As colisões são resolvidas através de uma arbitragem baseada numa prioridade pré-programada de cada mensagem no campo identificador. O identificador de maior prioridade ganha sempre o acesso ao barramento. Em outras palavras, o último nível alto ativo no identificador continua a transmitir porque é a prioridade máxima e uma vez que cada nó de um barramento participa na escrita de cada bit, um nó arbitrário sabe se colocou um bit nível alto ativo.

A norma ISO-11898:2003, com o identificador padrão de 11 bits, prevê taxas de sinalização a partir de 125 kbps a 1 Mbps. A norma foi posteriormente alterada com o identificado estendido de 29 bits. O padrão de 11 bits prevê 2048 identificadores de mensagens diferentes, enquanto o campo de identificação estendido, prevê 537 milhões de identificadores.

2.2 Os campos de Bits

2.2.1 CAN Padrão

O CAN Padrão possui o seguinte pacote de dados:

Tabela 2: CAN Padrão

SOF	11 bit Id	RTR	IDE	R0	DLC	0...8 Bytes Data	CRC	ACK	EOF	IFS
-----	-----------	-----	-----	----	-----	---------------------	-----	-----	-----	-----

Fonte: Autoria própria

- SOF: O único início dominante de pacote (*single dominant start of frame – SOF*) marca o início de uma mensagem e é usado para sincronizar os nós de um barramento depois de estar inativo.
- Identificador: O identificador do CAN Padrão é de 11 bits e estabelece a prioridade da mensagem. Quanto menor o valor binário, maior é sua prioridade.
- RTR: O único bit de pedido de transmissão (*single remote transmission request – RTR*) é dominante quando a informação é requerida a outro nós. Todos os nós recebem o pedido, mas o identificador determina o nó especificado. Os dados de resposta são também recebidos por todos os nós e utilizados por qualquer nó interessado. Desta forma, todos os dados são usados de forma uniforme.
- IDE: Uma extensão de identificador único (*identifier extension - IDE*) dominante, significa que um identificador CAN padrão sem extensão está a ser transmitido.
- R0: Bit reservado para implementações futuras da tecnologia.
- DLC: O código de comprimento de dados de 4 bits (*data length code – DLC*) contém o número de bytes de dados a serem transmitidos.
- Data: 64 bits de dados para aplicação.
- CRC: A verificação de redundância cíclica (*cyclic redundancy check – CRC*) de 16 bits (15 bits mais delimitador) contém o *checksum* dos dados da aplicação anterior para a detecção de erros.

- ACK: Cada nó que recebe uma mensagem precisa substituir este bit recessivo na mensagem original com um bit dominante, dessa forma, indicando que uma mensagem sem erros foi enviada. ACK é de 2 bits, sendo um bit o de reconhecimento e o outro, o delimitador.
- EOF: Este campo de fim de quadro (*end of frame – EOF*) de 7 bits marca o fim de um quadro CAN.
- IFS: Este espaço entre quadros de 7 bits (*interframe space – IFS*) contém o tempo requerido pelo controlador para mover corretamente um *frame* para a sua posição adequada.

2.2.2 CAN Estendido

Tabela 3: CAN Estendido

S	11 bit ID	S	I	18 bit ID	R	R1	R0	D	0...8 Bytes Data	C	A	E	I
O		R	D		T			L		R	C	O	F
F		R	E		R			C		C	K	F	S

Fonte: Autoria própria

A mensagem do CAN Estendido segue igual ao CAN Padrão, com as seguintes adições:

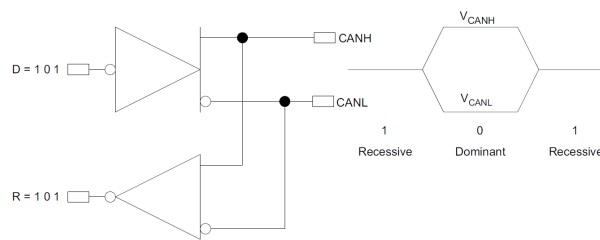
- SRR: O bit do pedido remoto substituto (*substitute remote bit - SRR*) substitui o bit do RTR no local padrão da mensagem como titular de um lugar no formato estendido.
- IDE: Um bit recessivo na extensão do identificador (*identifier extension – IDE*) indica que se seguem mais bits identificadores.
- R1: Seguindo RTR e R0 bits, diz respeito a um bit de reserva adicional incluído antes do bit DLC.

2.3 A Mensagem CAN

2.3.1 Arbitragem

A tecnologia CAN apresenta estados de níveis lógicos opostos entre o barramento, *driver* de entrada e *receiver* de saída. Usualmente, na maior parte das aplicações, o nível lógico alto diz respeito ao dígito binário “1”, e um nível lógico baixo, ao dígito binário “0”. Em termos de barramento, essencialmente, as mensagens são transmitidas por meio de dois canais intitulados CAN HIGH e CAN LOW.

Figura 1: Lógica invertida do barramento CAN



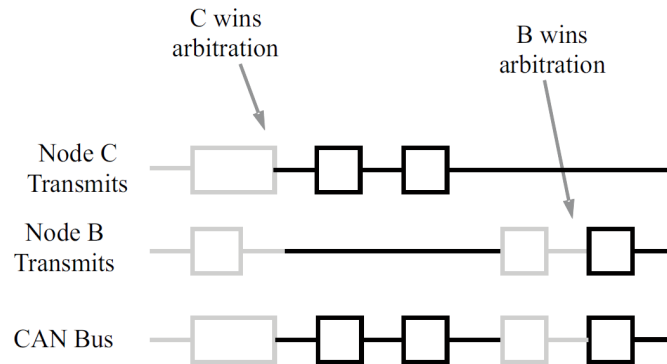
Fonte: Texas Instruments

O acesso ao barramento é conduzido por eventos e tem lugar de forma aleatória. Se dois nós tentarem ocupar o barramento simultaneamente, o acesso é implementado com uma arbitragem não destrutiva, o que significa que o nó vencedor apenas continua com a mensagem sem que seja destruída ou corrompida por outro nó.

A atribuição de prioridade às mensagens no identificador é uma característica da tecnologia CAN que o torna atraente para a utilização dentro de um ambiente de controle em tempo real. Quanto mais baixo for o número de identificação da mensagem binária, maior é a sua prioridade. Um identificador composto inteiramente de zeros é a mensagem de maior prioridade numa rede, porque detém o barramento dominante por maior tempo. Portanto, se dois nós começarem a transmitir em simultâneo, o nó que envia um último bit identificador como um zero (dominante) enquanto os outros nós enviam um dígito binário “um” (recessivo) mantém o controle

do barramento CAN e prossegue para complementar a sua mensagem. Um bit dominante sobrescreve sempre um bit recessivo no barramento CAN.

Figura 2: Arbitração no barramento CAN



Fonte: Texas Instruments

A atribuição da prioridade da mensagem cabe ao *designer* de sistemas, mas os grupos industriais concordam mutuamente sobre o significado de certas mensagens. Por exemplo, um fabricante de acionamentos motorizados pode especificar essa mensagem 0010 como um sinal de corrente de retorno de um motor numa rede CAN e que 0011 é a velocidade do contador de rotações. Como 0010 tem o identificador binário mais baixo, as mensagens relativas aos valores atuais têm sempre uma prioridade mais elevada no barramento.

2.3.2 Tipos de Mensagens

2.3.2.1 Data Frame

O *data frame* é o tipo de mensagem mais comum, e compreende o campo de arbitragem, o campo de dados, o campo CRC e o campo de reconhecimento. O campo de arbitragem contém um identificador de 11 bits no CAN padrão e o bit RTR, que é dominante para os *data frames*. O CAN Estendido contém o identificador de 29 bits e o bit RTR. A seguir há o campo de dados que contém até 8 bytes de dados, e campo

CRC que contém o *checksum* de 16 bits utilizado para a detecção de erros. O último é o campo de reconhecimento.

2.3.2.2 Remote Frame

O objetivo do *remote frame* é solicitar a transmissão de dados de outro nó. É semelhante ao *data frame*, com duas diferenças importantes: em primeiro lugar, este tipo de mensagem é explicitamente marcado como um *remote frame* por um bit RTR recessivo no campo de arbitragem, e em segundo lugar, há sem dados.

2.3.2.3 Error Frame

O *error frame* é uma mensagem especial que viola as regras de formatação de uma mensagem CAN. É transmitida quando um nó detecta um erro numa mensagem, e faz com que todos os outros nós da rede enviem um erro também a moldura. O transmissor original retransmite automaticamente a mensagem. Um sistema elaborado de contadores de erros no controlador CAN assegura que um nó não pode amarrar um barramento transmitindo repetidamente quadros de erro.

2.3.2.4 Overload Frame

O *overload frame* é análogo ao *error frame* no que diz respeito ao formato e é transmitido por um nó que se torna ocupado. É utilizado sobretudo para fornecer um atraso entre mensagens.

2.3.3 Valid Frame

Uma mensagem é considerada livre de erros quando a última parte do campo EOF de uma mensagem é recebido no estado recessivo sem erro. Um bit dominante no campo EOF faz com que o transmissor repita uma transmissão.

2.3.4 Verificação de Erros e Confinamento de Falhas

A robustez do CAN pode ser atribuída, em parte, aos seus abundantes procedimentos de verificação de erros. O CAN incorpora cinco métodos de verificação de erros: três ao nível da mensagem e dois ao nível do bit. Se uma mensagem falha em qualquer um desses métodos de verificação de erros, não é aceita e o quadro de erro é gerado a partir de um nó receptor. Isto força o nó transmissor a reenviar a mensagem até que esta seja recebida corretamente. No entanto, se um nó defeituoso perdurar continuamente, a capacidade de transmissão é removida pelo controlador após um determinado limite de erros ter sido alcançado.

A verificação de erros ao nível da mensagem é executada pelo CRC e os slots ACK apresentados na CAN Padrão e Estendido. O CRC de 16 bits contém o *checksum* dos dados da aplicação anterior para detecção de erros com um *checksum* de 15 bits e um delimitador de 1 bit. O campo ACK é de 2 bits de comprimento e consiste no bit de reconhecimento e de 1 bit de delimitador de reconhecimento.

Também ao nível da mensagem é realizada uma verificação de formulário. Esta verificação procura campos na mensagem que devem sempre ser pedaços recessivos. Se um bit dominante for detectado, é gerado um erro. Os bits verificados são SOF, EOF, o delimitador ACK e os bits do delimitador CRC.

Ao nível do bit, cada bit transmitido é monitorizado pelo transmissor da mensagem. Se um bit de dados (não bit de arbitragem) é escrito no barramento e o seu oposto é lido, é gerado um erro.

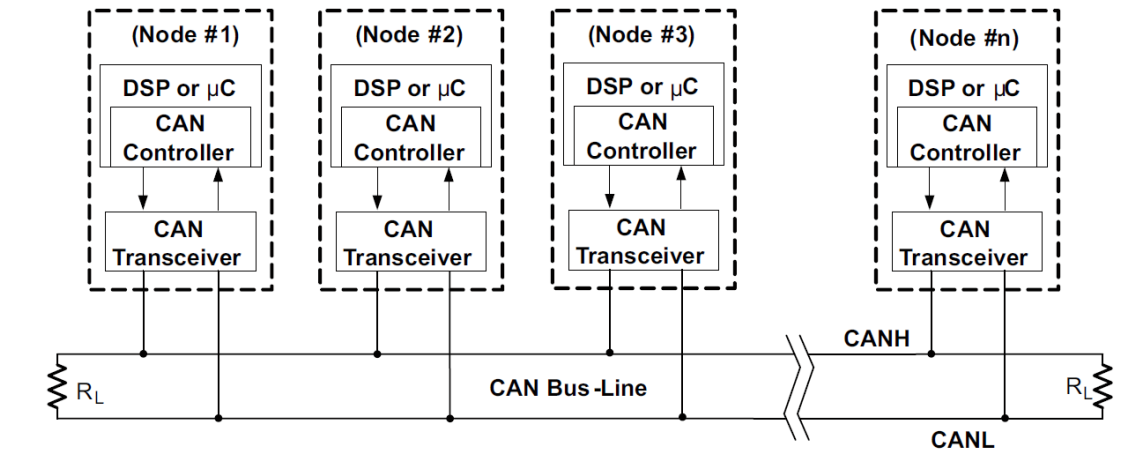
O método final de detecção de erros é com a regra de retificação de bits, onde após cinco bits consecutivos do mesmo nível lógico, se o bit seguinte não for um complemento, é gerado um erro.

2.4 O Barramento CAN

As camadas de ligação de dados e de sinalização física vistas na tabela 1, estão incluídas em qualquer controlador que implemente o protocolo CAN. A conexão

com o meio físico é implementada por meio de um transceptor no intuito de formar um nó de sistema como mostrado na figura abaixo:

Figura 3: Barramento CAN



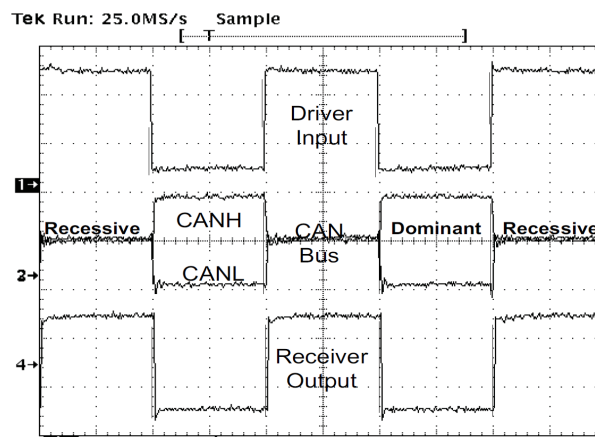
Fonte: Texas Instruments

A sinalização é do tipo diferencial, dessa forma o CAN demonstra uma robusta imunidade ao ruído e tolerância a falhas. A sinalização diferencial reduz o acoplamento de ruído e permite altas taxas de sinalização sobre o par trançado. Idealmente, a corrente que flui em cada linha de sinal é igual, mas em direção oposta, resultando em um cancelamento de campo, que é essencial para o controle do ruído. A utilização da tecnologia diferencial e a cablagem de par trançado elevam a imunidade ao ruído no barramento CAN.

A especificação da norma ISO 11898 de alta velocidade são dadas para uma taxa máxima de sinalização de 1 Mbps com um comprimento de barramento de 40 metros com uma quantidade máxima de 30 nós. O cabo é especificado para ser do tipo par trançado blindado ou não, com 120 ohm de impedância característica. A norma ISO 11898 define uma única linha de cabo de par trançado como a topologia da rede, como mostra a figura acima, terminada em ambas as extremidades com 120 ohm, que correspondem à impedância característica da linha para evitar reflexões do sinal.

As duas linhas de sinal do barramento, CAN HIGH e CAN LOW, no estado recessivo, operam em aproximadamente 2.5 V. O estado dominante no barramento leva CAN HIGH para 3.5 V e CAN LOW para 1.5 V, criando um sinal diferencial de 2 V.

Figura 4: Estados no barramento CAN



Fonte: Texas Instruments

O CAN Padrão define uma rede de comunicação que liga todos os nós ligados a um barramento e permite-lhes se comunicar uns com os outros. Pode ou não haver um nó de controle central, e os nós podem ser acrescentados a qualquer momento.

Os nós da figura 3 poderiam teoricamente estar configurados para enviar mensagens de sensoriamento inteligente e um controlador de motor. Uma aplicação real pode incluir um sensor de temperatura que envie uma atualização dessa grandeza que pode ser utilizada para ajustar a velocidade do motor de um exaustor. Se um nó de um determinado outro sensor quiser enviar uma mensagem ao mesmo tempo, a arbitragem assegura que ela é enviada.

3 O Projeto

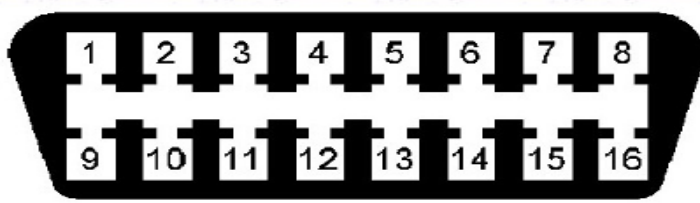
3.1 A Concepção

A tecnologia CAN clássica é amplamente validada pelo meio acadêmico e industrial. A nível de eletrônica, para os devidos fins, o projeto essencialmente envolve o estudo dos circuitos integrados a serem utilizados, seus circuitos periféricos, a construção dos esquemáticos, e o roteamento dos componentes numa placa de circuito impresso via *software*.

3.2 Circuitos Integrados

Na maior parte dos veículos automotivos as informações lógicas podem ser acessadas por meio do conector OBD. Esta conectividade está presente na maior parte dos veículos, e por meio dela, é possível acessar os dados provenientes dos sinais *CAN HIGH* e *CAN LOW*.

Figura 5: Conector OBD



PIN	DESCRIPTION	PIN	DESCRIPTION
1	Vendor Option	9	Vendor Option
2	J1850 Bus +	10	j1850 BUS
3	Vendor Option	11	Vendor Option
4	Chassis Ground	12	Vendor Option
5	Signal Ground	13	Vendor Option
6	CAN (J-2234) High	14	CAN (J-2234) Low
7	ISO 9141-2 K-Line	15	ISO 9141-2 Low
8	Vendor Option	16	Battery Power

Fonte: *Trustedsec.com*

Os sinais *CAN HIGH* e *CAN LOW* por sua vez, necessitam de um meio para serem interfaceados no projeto, e terem dados tratados de forma inteligente, via de regra, por um microcontrolador, executando tarefas determinadas, um transceptor de alta velocidade e um controlador CAN.

3.2.3 Transceptor de Alta Velocidade

O transceptor, enquanto circuito integrado, tem a função de conduzir e detectar dados para e a partir de um determinado barramento. Ele traduz a lógica utilizada pelo barramento para um determinado controlador, emitindo um sinal lógico aceitável na forma de TX e RX.

Para aplicações automotivas, um transceptor necessita ter algumas características próprias, como:

- Compatibilidade com a norma ISO.
- Alta velocidade.
- Oferecer proteção contra transientes provenientes de aplicações automotivas.
- Imunidade eletromagnética.
- Níveis de tensão compatíveis com 3.3 V, usado em aplicações industriais.
- Proteção térmica.
- Proteção contra curtos-circuitos.
- Elevado número de nós permitidos.

3.2.3.1 TJA1050

O TJA1050 é um transceptor de alta velocidade fabricado pela PHILIPS. Possui todas as características importantes em se tratando de implementação em projetos automotivos. Possui taxa de dados de até 1Mbaud, proporciona uma capacidade de transmissão diferencial para a tecnologia, sendo totalmente compatível com a norma ISO 11898.

Um circuito limitador de corrente protege o estágio da saída do transmissor de eventuais danos causados por curtos-circuitos.

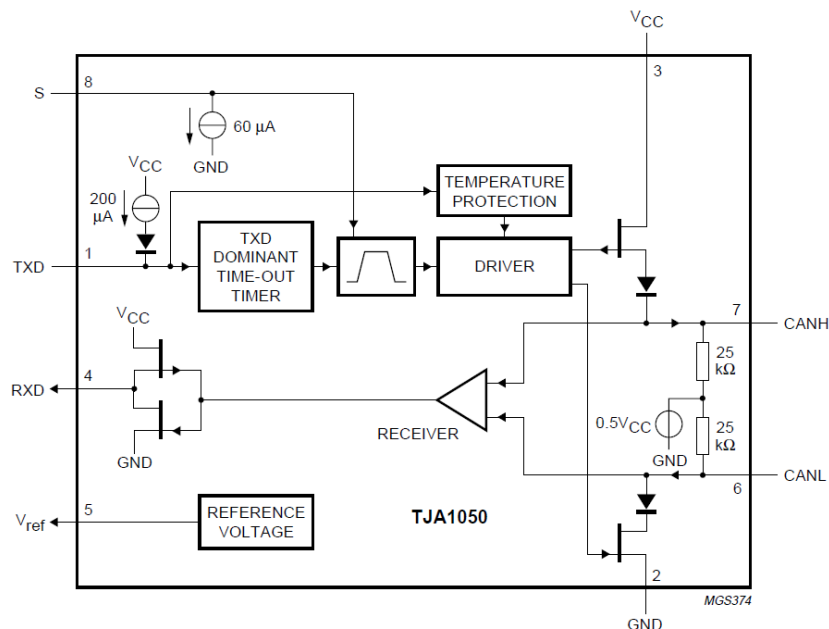
Um circuito de proteção térmica protege o CI de danos se a temperatura de junção exceder um valor aproximado de 165 °C. Uma vez que o transmissor dissipa a maior parte da potência, a temperatura geral do CI é reduzida.

As linhas *CAN HIGH* e *CAN LOW* estão protegidas contra transientes oriundos do veículo automotivo e da mesma forma, contra descargas eletrostáticas de até 4 kV do corpo humano.

A linha de controle S permite a seleção de dois modos de operação: modo de alta velocidade e modo silencioso. No modo silencioso, o transmissor é desativado ligando o pino ao VCC e todas as outras funcionalidades se mantêm.

O TJA1050 também possui um circuito temporizador que impede o barramento de ser conduzido para um estado dominante permanente (bloqueando todas as comunicações em rede) se o TXD for forçado permanentemente para nível lógico baixo por uma falha de hardware e ou aplicação de software. O temporizador é acionado por uma borda negativa no TXD. Se a duração do nível lógico baixo no TXD excede a duração interna do temporizador, o transmissor é desativado, conduzindo o barramento para um estado recessivo. O temporizador é reiniciado por uma margem positiva em TXD.

Figura 6: Diagrama de blocos do TJA1050



Fonte: Philips

3.2.4 Controlador CAN

Para a implementação do projeto de *hardware* faz-se necessário a escolha de um controlador CAN. Este tipo de controlador é responsável por receber os sinais TX e RX oriundos do transceptor e enviá-los a, tipicamente, um microcontrolador, que trata os dados e executa a rotina lógica das regras de negócio do projeto.

3.2.4.1 MCP2515

Em aplicações industriais, comumente o circuito integrado MCP2515 da *Microchip Technology* é usado para desempenhar o papel de controlador CAN para projetos de pequeno porte. É um controlador que implementa a especificação CAN e é capaz de transmitir e receber o CAN Padrão e Estendido. A tecnologia do circuito integrado possui características para interfacear microcontroladores através do barramento *SPI (Serial Peripheral Interface)*. O seu diagrama de blocos pode ser entendido por meio de três módulos principais: CAN, lógica de controle e protocolo SPI.

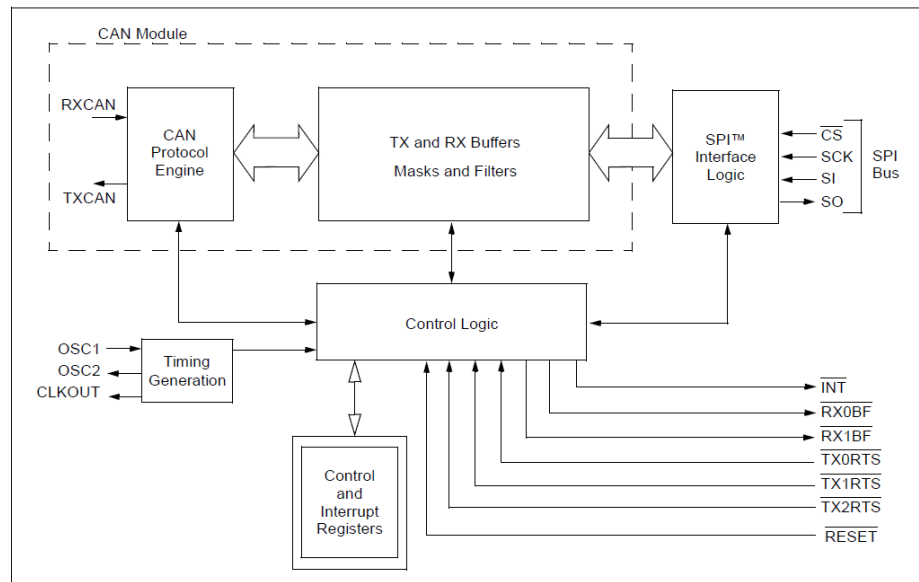
O módulo CAN trata de todas as funções de recepção e transmissão de mensagens acerca do barramento CAN. As mensagens são transmitidas por meio do primeiro carregamento de *buffer*. A transmissão é iniciada por meio da utilização de bits de registradores via SPI ou utilizando pinos de ativação de transmissão. Estados e erros podem ser verificados. Qualquer mensagem detectada no barramento CAN tem sua verificação de erros checada.

O módulo da lógica de controle controla a configuração e a operação do MCP2515, fazendo a interface com os outros blocos em ordem para passar a informação e o controle.

Os pinos de interrupção são fornecidos para permitir uma maior flexibilidade ao sistema. A utilização dos pinos de interrupção de propósito geral, bem como os registradores de status, pode ser utilizada para determinar quando uma mensagem válida foi recebida. Também existem três pinos disponíveis para iniciar transmissão imediata de uma mensagem que tenha sido carregada nos registradores de transmissão. A utilização desses pinos é opcional.

O microcontrolador pode interagir com o dispositivo por meio da interface SPI. Escrita e leitura a partir dos registradores são realizados por meio dessa tecnologia.

Figura 7: Diagrama de blocos do MCP2515



Fonte: Microchip

3.2.5 Microcontrolador

Um microcontrolador pode ser entendido como um circuito integrado contendo, principalmente, um núcleo processador, memória e periféricos programáveis de entrada e saída.

Microcontroladores podem ser utilizados em aplicações de sistemas embarcados como: sistemas de controle, sistemas de monitoramento de dados, eletrodomésticos, entre outros. No geral, possuem baixa consumo de potência, geralmente, em miliwatts, são compactos, e são de uso amplamente difundido.

3.2.5.1 ATMEGA328P

O microcontrolador escolhido para o projeto é o ATMEGA328P. A escolha de um microcontrolador em um projeto de sistemas embarcados é algo crítico, devendo-se tomar por nota quais funcionalidades realmente importam para o bom funcionamento das regras de desenvolvimento para que as funcionalidades sejam implementadas com êxito. O microcontrolador em questão é fabricado pela *Microchip Technology*, de tecnologia CMOS baixa potência de 8 bits baseado na arquitetura AVR/RISC.

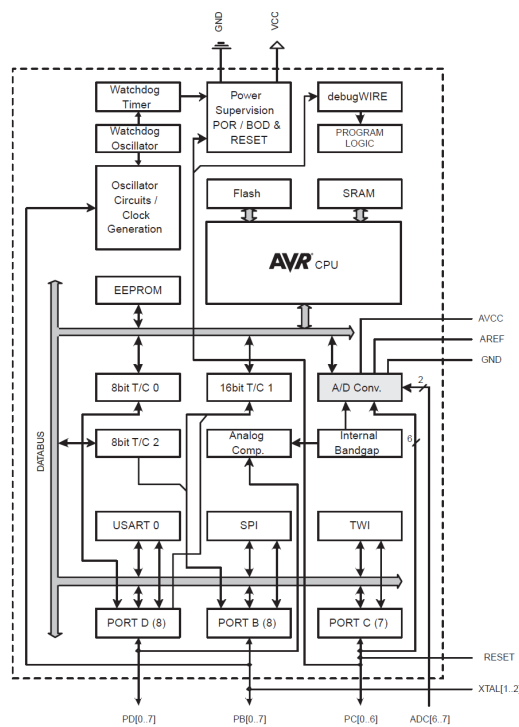
O ATMEGA328P fornece um conjunto robusto de instruções com 32 registradores de trabalho de propósito geral. Todos eles estão diretamente ligados à Unidade Lógica Aritmética (ULA), permitindo que dois registradores independentes sejam acessados numa única instrução executada num único ciclo de *clock*.

O microcontrolador fornece 32 kbytes de memória *flash* programável com capacidade de leitura-escrita, 1 kbyte de EEPROM, 2kbytes de SRAM, 23 linhas de entrada e saída de uso geral, 32 registradores para fins gerais, contador em tempo real, três temporizadores/contadores, PWM, uma USART programável, uma interface I2C, ADC, temporizador *watchdog* programável com oscilador interno, interface SPI e modos selecionáveis para consumo inteligente de potência. O modo *idle* interrompe a CPU, enquanto permite a SRAM, temporizadores/contadores e SPI funcionando. O modo *power-down* salva o conteúdo de registro, mas congela o oscilador, desativando todas as outras funções do chip até a próxima interrupção ou reinicialização do *hardware*. No modo *power-slave*, o temporizador assíncrono continua a funcionar, permitindo ao utilizador manter uma base de temporizadores enquanto o resto do dispositivo está adormecido. O modo de redução de ruído do ADC interrompe a CPU e todos os módulos de entrada e saída, exceto os temporizadores assíncronos e o ADC para minimizar o ruído de comutação durante as conversões do conversor. No modo *standby*, o oscilador do cristal funciona enquanto o resto do dispositivo está adormecido. Isto permite uma inicialização mais rápida combinada com um baixo consumo de potência. Em modo *standby*, tanto o oscilador como o temporizador principal assíncrono continuam a funcionar.

O dispositivo é fabricado fazendo uso da tecnologia de memória não volátil de alta densidade da *Microchip*. A tecnologia *ISP On-chip Flash* possibilita que a memória do programa seja reprogramada por meio de uma interface serial SPI por um

programador de memória não volátil convencional ou por um programa *On-chip boot* a ser executado do microcontrolador. O programa *boot* pode fazer uso de qualquer interface para descarregar o programa de aplicação na memória *flash*. O software continuará a funcionar enquanto a *flash* é atualizada, proporcionando uma verdadeira operação de leitura-escrita. Combinando uma CPE RISC de 8 bits com a tecnologia *In-System Flash* autoprogramável em um chip monolítico, o ATMEGA328P é um poderoso microcontrolador que fornece uma solução altamente flexível e rentável para muitas aplicações de controle.

Figura 8: Diagrama de blocos do ATMEGA328P



Fonte: Microchip

3.3 Circuitos periféricos

3.3.1 Regulação de tensão

Em quase todo projeto de dispositivos eletrônicos se faz necessário pensar sobre a alimentação de tensão de entrada. Para o projeto em questão, foi pensando

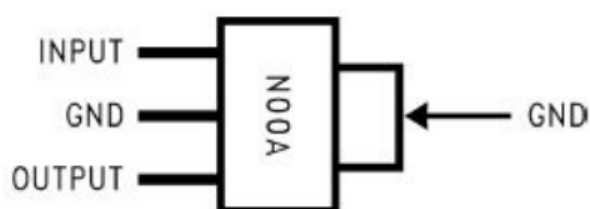
uma alimentação externa de 12/24 V, que tipifica o padrão industrial na maior parte das aplicações. Todavia, uma vez que os circuitos integrados, via de regra, são alimentados com 5 V ou 3.3 V, é necessário projetar uma topologia de circuitos que realize a regulação desejada para a finalidade necessária. Pensando nisso, o projeto utilizou, essencialmente, dois reguladores de tensão: LM7805 e LM1117. O primeiro, responsável por regular a tensão para 5 V e o segundo para 3.3 V, que é o nível de tensão usado nos circuitos integrados adotados.

3.3.1.1 LM7805

O LM7805 é um regulador de tensão que possui tecnologia para limitar a corrente interna e desligamento térmico, fazendo com que ele seja essencialmente robusto. Se no projeto eletrônico for fornecido um tratamento térmico adequado, ele pode entregar mais de 1.5 A de corrente de saída.

Em seu projeto de tecnologia foram empregados grandes esforços para facilitar sua implementação em projetos e reduzir o número de circuitos periféricos para sua empregabilidade, sendo necessário basicamente capacitores de entrada e saída.

Figura 9: Pinout do LM7805



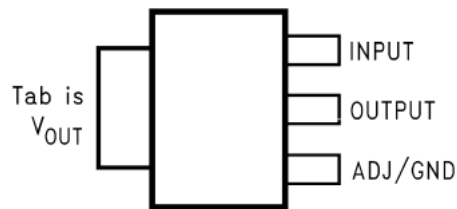
Fonte: Texas Instruments

3.3.1.2 LM1117

O LM1117 é um regulador de baixa potência de tensão positiva concebido para satisfazer uma corrente de saída de 1 A e estabelecer tensão de saída de 3.3 V. Este

dispositivo é uma excelente escolha para uso em aplicações eletrônicas embarcadas. Apresenta uma corrente quiescente muito baixa e uma tensão de saída estável. Está disponível em tensões de saída fixas de 1.8 V, 2.5 V, 3.3 V e 5 V. O regulador de tensão também oferece limitação de corrente e proteção térmica.

Figura 10: Pinout do LM1117



Fonte: Texas Instruments

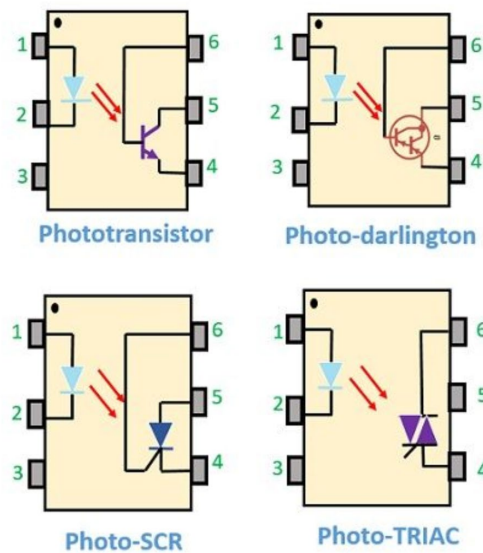
3.3.2 Optoacopladores

Optoacoplador ou também chamado optoisolador é um dispositivo semicondutor que possibilita a transmissão de um sinal elétrico entre dois circuitos isolados.

Um tipo comum de optoacoplador consiste num LED e um fototransmissor, mas combinações com outras tecnologias também estão disponíveis: fotodiodos, foto-darlington, foto-scr, foto-triac. Normalmente esses dispositivos transferem sinais digitais (on/off), mas algumas técnicas permitem sua implementação em aplicações de sinais analógicos.

Os optoacopladores contêm uma fonte emissora de luz, tipicamente um diodo emissor que emite luz infravermelha, convertendo o sinal de entrada em sinal luminoso, um canal óptico fechado e um elemento sensível que detecta a entrada de luz ou modula a corrente elétrica que flui de uma fonte de alimentação externa.

Figura 11: Modelos de optoacopladores



Fonte: Texas Instruments

3.3.3 Filtros de desacoplamento

Os dispositivos de um projeto eletrônico são ligados às suas fontes de alimentação por meio de condutores com resistência finita e indutância. Se a corrente drenada por um dispositivo mudar, as quedas de tensão das fontes de alimentação para o dispositivo também mudam devido às impedâncias. Se vários dispositivos partilham um caminho comum para a fonte de alimentação, as mudanças na corrente drenada por um elemento podem produzir mudanças de tensão suficientemente elevadas para afetar o funcionamento de outros pelo que a mudança de estado de um dispositivo é acoplada a outros por meio da impedância comum à fonte de alimentação. Um capacitor de desacoplamento fornece um caminho de desvio para correntes transitórias ao invés de fluir por meio da impedância comum.

O capacitor de desacoplamento atua como um dispositivo de armazenamento local de energia do dispositivo. O capacitor, via de regra, é colocado entre a linha de tensão e o GND, até ao circuito em que a corrente deve ser fornecida. Quando a capacitância é suficientemente elevada, é fornecida corrente apropriada para manter uma gama aceitável de queda de tensão. O capacitor armazena uma pequena quantidade de energia que pode compensar a queda de tensão nos condutores de

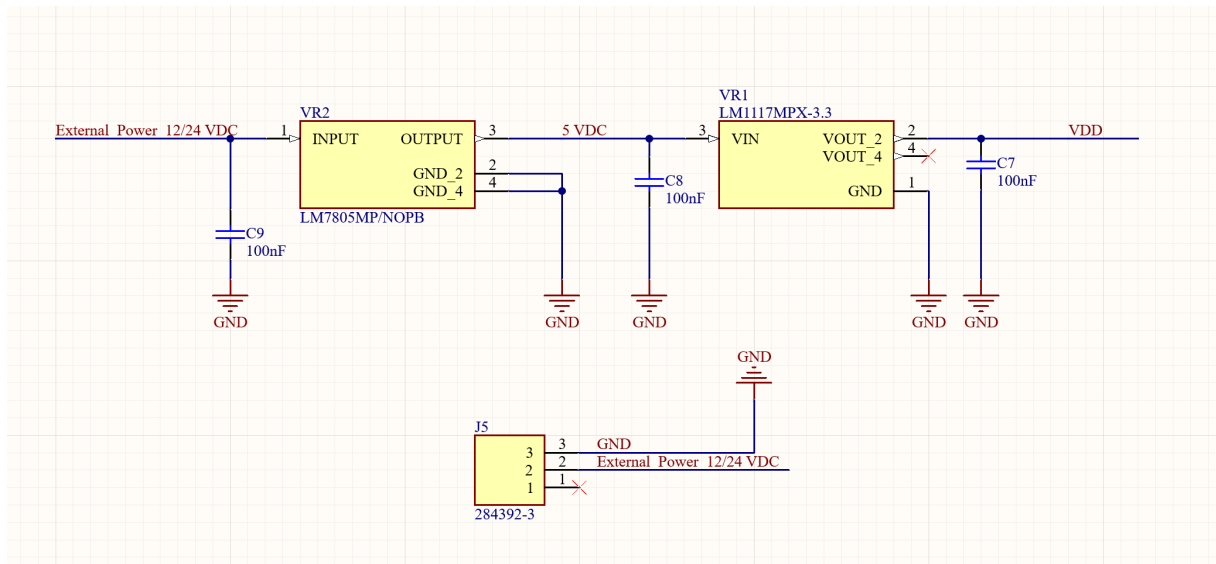
alimentação. Para reduzir a indesejável indutância parasita, os capacitores são frequentemente colocados em paralelo, adjacentes a circuitos integrados individuais.

O uso de capacitores nesse intuito é frequentemente utilizado para desacoplar um subcircuito de sinais AC ou picos de tensão numa fonte de alimentação. Pode desviar energia desses sinais, ou transitórios, para além do subcircuito a ser desacoplado, diretamente para o caminho de retorno.

4 Esquemáticos

4.1 Folha 01: Reguladores de Tensão

Figura 12: Folha dos reguladores de tensão



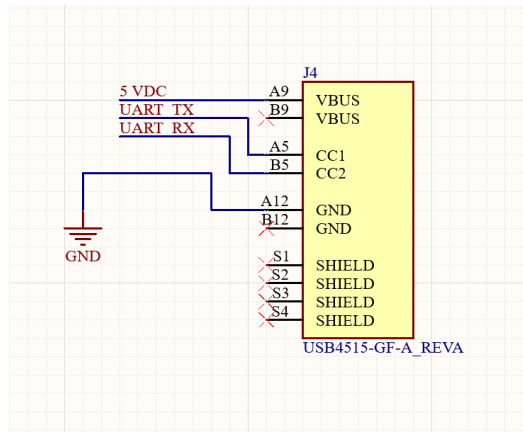
Fonte: Autoria própria

Na primeira folha de esquemáticos do projeto foi implementado a topologia para regulação de tensão. São usados os dois reguladores de tensão previamente citados. O primeiro, LM7805, recebe uma alimentação externa por meio do conector J5. A tensão de saída gerada é de 5 V, que é direcionada para o LM1117, este, por sua vez, regulando a tensão para 3.3 V, que é o nível desejado para alimentar os circuitos integrados. Os capacitores utilizados, como desacoplamento, têm a finalidade de

evitar picos de tensão indesejáveis, e dessa forma, garantir a proteção elétrica dos dispositivos.

4.2 Folha 02: Conectividade USB

Figura 13: Folha da conectividade USB

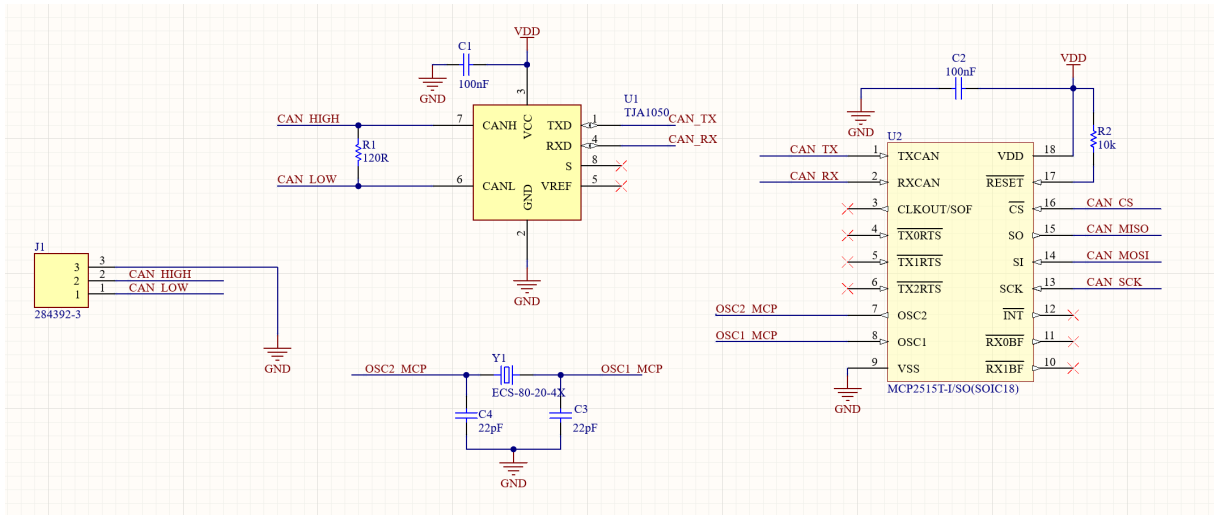


Fonte: Autoria própria

Na folha 02 foi implementado a conectividade USB. Esse tipo de conectividade possibilita o acesso das informações provenientes do microcontrolador para o computador ou algum outro dispositivo de acesso serial. Os dados que serão transmitidos via linha USB são as informações veiculares tratadas. Essencialmente, elas são disponibilizadas via pinos RX/TX do microcontrolador para o barramento *Data Minus* e *Data Plus* da conectividade USB.

4.4 Folha 03: Módulo CAN

Figura 14: Folha do módulo CAN



Fonte: Autoria própria

A folha 03 contém o principal esquemático do projeto: O módulo CAN. Nela consta a implementação de toda a conectividade CAN. O conector J1 recebe os dados oriundos do conector OBD, via *CAN HIGH* e *CAN LOW*, direcionando os sinais para o TJA1050, que por sua vez traduz os dados internamente para CAN TX e CAN RX, que são aceitáveis ao controlador CAN MCP2515, e este, envia os dados para o microcontrolador.

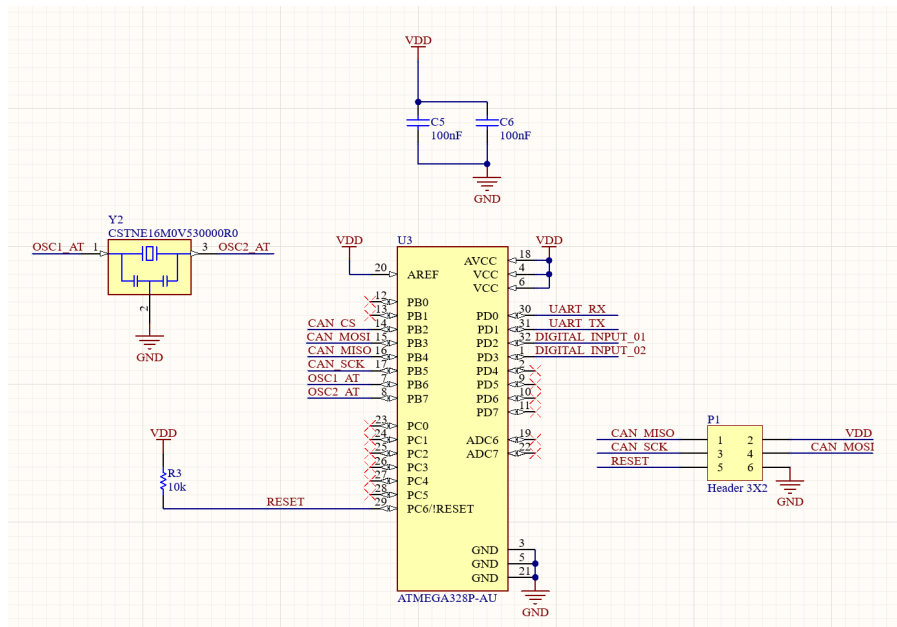
O transceptor de alta velocidade, TJA1050, necessita de um resistor terminal de 120 ohm. Este resistor tem o importante papel de evitar reflexões de sinal. Também é necessário a implementação de um capacitor de desacoplamento conectado ao pino de tensão de entrada para evitar oscilações indesejáveis de tensão, e assim garantir a proteção elétrica ao circuito.

O controlador CAN necessita de um cristal oscilador de 8 MHz para prover um *clock* estável de sinal e da mesma forma que o transceptor, capacitor de desacoplamento. Ele recebe os sinais CAN TX e CAN RX, e por meio da tecnologia SPI, os dados são acessados pelo microcontrolador via linhas: MISO, MOSI, SCK e

SCK. Analogamente aos demais circuitos integrados, também se faz necessário a implementação de capacitores de proteção.

4.5 Folha 04: Módulo MCU

Figura 15: Folha do módulo MCU



Fonte: Autoria própria

A folha 04 contém a implementação do módulo pertinente ao microcontrolador. O fabricante recomenda, numa configuração mais simples a implementação dos capacitores de desacoplamento e um cristal oscilador que por sua vez possui a função de configurar o *clock* do microcontrolador.

O projeto de firmware fez uso do ambiente de desenvolvimento da própria fabricante: *Microchip Studio*. Também foi adicionado um conector ISP para gravação dos dados no microcontrolador fazendo uso de um gravador e computador.

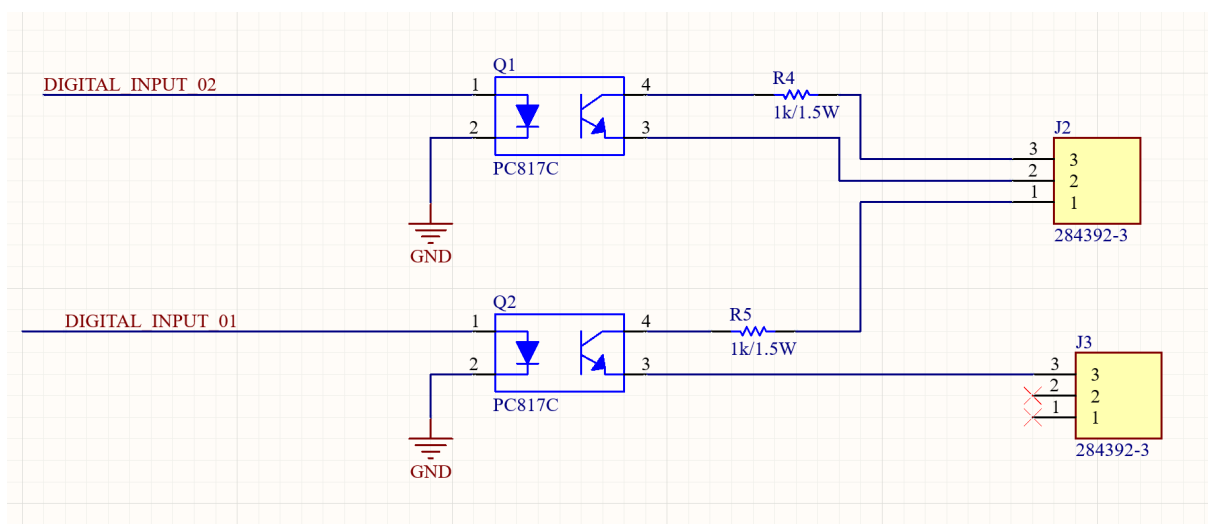
Figura 16: AVR ISP



Fonte: Autoria própria

4.6 Folha 05: Módulo OptoAcoplador

Figura 17: Folha do Optoacoplador



Fonte: Autoria própria

Adicionalmente, foi implementado no projeto uma folha com optoacopladores para leitura de sinais analógicos.

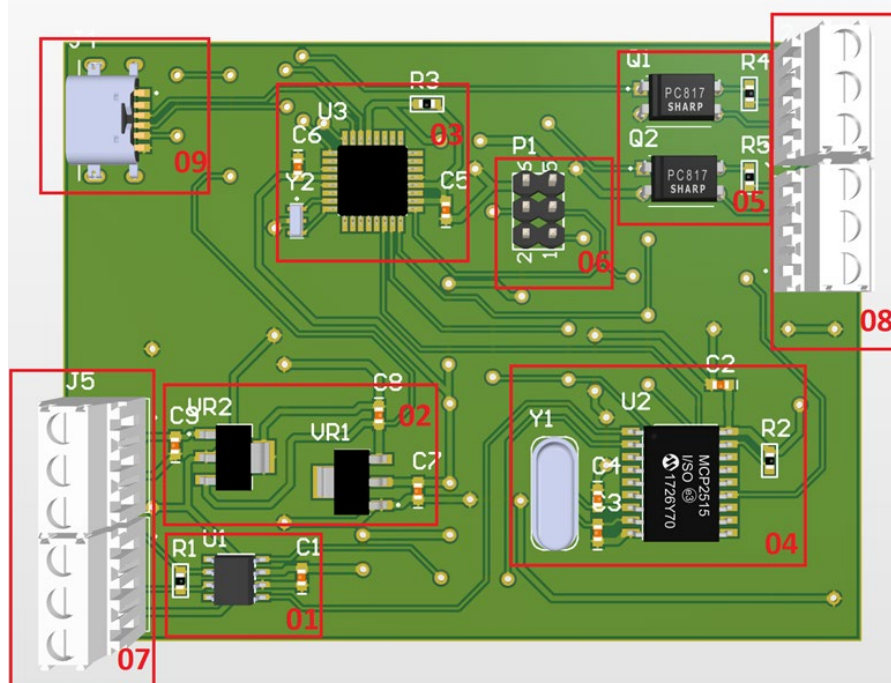
Os microcontroladores, de maneira geral, operam em 5 V e respondem a rápidas variações de sinal. Dessa forma, um possível ruído elétrico, pode ser interpretado como um tipo de sinal. Este tipo de leitura indesejada pode ser eliminado de dois modos, via *software* ou *hardware*. Comumente, como exemplo de implementação em *hardware* faz-se uso de optoacopladores.

Os optoacopladores realizam um isolamento elétrico entre uma fonte de entrada e uma carga de saída usando apenas luz.

5 Sistema Final

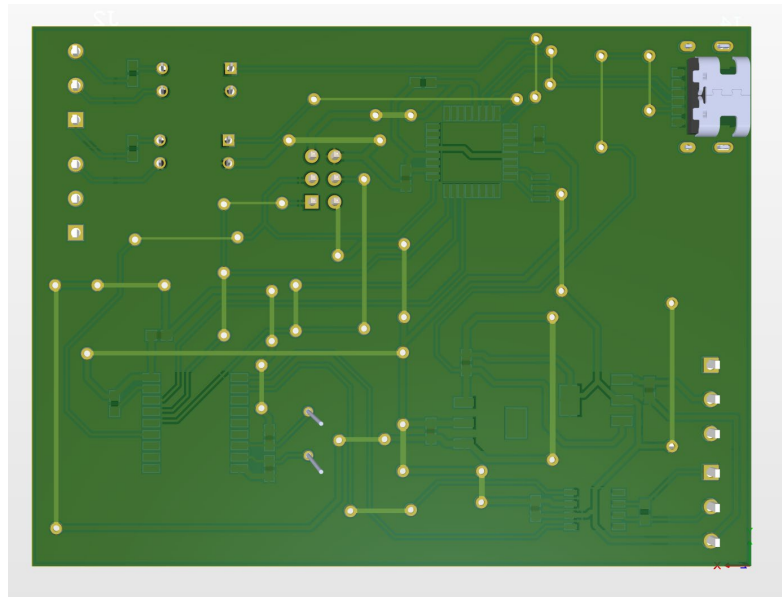
De posse de todos os esquemáticos elaborados, é possível realizar o processo de roteamento dos componentes e obter o *hardware* final.

Figura 18: Vista superior



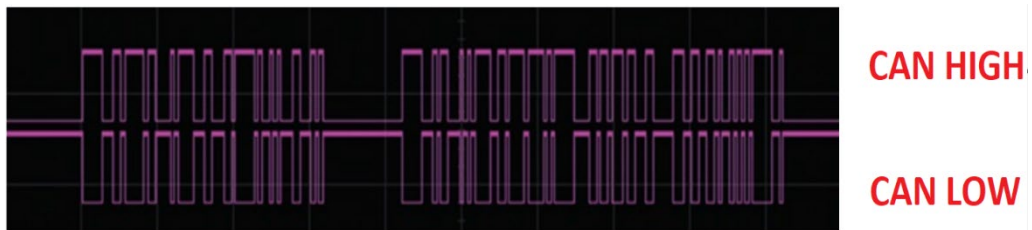
Fonte: Autoria própria

Figura 19: Vista inferior



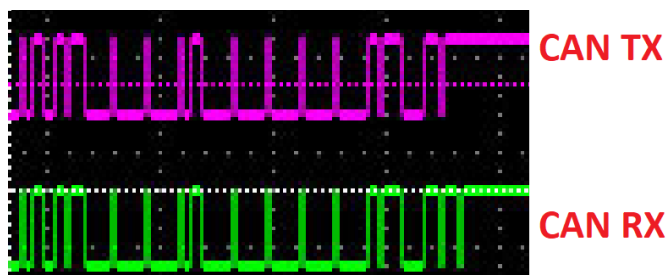
Fonte: Autoria própria

Figura 20: Recorte do Osciloscópio dos sinais CAN



Fonte: Autoria própria

Figura 21: Sinais CAN TX/RX



Fonte: Autoria própria

A placa de circuito impresso contém todos os componentes e circuitos contidos nas folhas de esquemáticos, conforme o mapeamento abaixo:

Tabela 4: Mapeamento de blocos

Bloco	Circuitos
01	Transceptor CAN.
02	Reguladores de tensão.
03	Microcontrolador.
04	Controlador CAN.
05	Optoacopladores.
06	Conectividade ISP.
07	Conector de entrada 1.
08	Conector de entrada 2.
09	Conectividade USB.

Fonte: Autoria própria

O projeto fez uso das duas faces da placa. Na face superior constam todos os componentes e conectores, e na outra, os *jumpers* necessários para conexões entre os circuitos.

6 Considerações Finais

Neste trabalho de conclusão de curso, implementou-se um projeto de *hardware* para uso da tecnologia CAN em veículos automotivos. A fundamentação teórica da tecnologia e as etapas de projeto foram amplamente comentados dentro da perspectiva de trabalho de graduação.

Verificou-se que a tecnologia CAN é de fácil implementação e requisita blocos de circuitos bem definidos, estes, essencialmente, a saber, reguladores de tensão para a correta operacionalização dos componentes e circuitos integrados, um transceptor CAN para captação de sinais oriundos do veículo automotivo e tradução em sinais TX

e RX, um controlador CAN capaz de receber tais sinais de enviá-los a um microcontrolador, onde de fato, as informações são processadas.

O principal desafio deste trabalho foi a elaboração das topologias de circuitos que atendessem as necessidades do projeto, uma vez que, isto só pode ser realizado mediante uma robusta compreensão dos aspectos físicos e lógicos da tecnologia em questão.

Durante o desenvolvimento deste trabalho de conclusão de curso, foram utilizados conhecimentos de diversas disciplinas estudadas dentro do escopo da graduação, como: Dispositivos eletrônicos, Eletrônica e Arquitetura de Sistemas Digitais.

Este trabalho também permitiu a aprendizagem de conhecimentos complementares à formação curricular, como técnicas de projetos de placas de circuitos impresso, compreendendo aspectos do *software* utilizado e boas práticas de implementação.

7 Referências

BOSCH, Robert. *CAN Specification Version 2.0*. Disponível em: <http://esd.cs.ucr.edu/webres/can20.pdf>

History of CAN Technology. Disponível em: <https://www.can-cia.org/can-knowledge/can/can-history/>

Controller Area Network. Disponível em: <https://www.ni.com/pt-br/innovations/white-papers/06/controller-area-network--can--overview.html>

Road vehicles — Controller area network (CAN). Disponível em: <https://www.iso.org/obp/ui/#iso:std:iso:11898:-1:ed-2:v1:en>

CAN and CAN FD Overview. Disponível em: <https://training.ti.com/ti-precision-labs-canlinsbc-can-and-can-fd-overview>

Introduction to the Controller Area Network. Disponível em: <https://www.ti.com/lit/an/sloa101b/sloa101b.pdf>

ATmega328P: 8-bit AVR Microcontroller with 32K Bytes In-System Programmable Flash. Disponível em: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf

TJA1050: High Speed CAN Transceiver. Disponível em: <https://www.nxp.com/docs/en/data-sheet/TJA1050.pdf>

MCP2515: Stand-Alone CAN Controller with SPI Interface. Disponível em: <https://ww1.microchip.com/downloads/en/DeviceDoc/MCP2515-Stand-Alone-CAN-Controller-with-SPI-20001801J.pdf>

ZACHARIAH, Jack. *Decoupling Capacitor and Bypass Placement Guidelines.* Disponível em: <https://resources.altium.com/p/bypass-and-decoupling-capacitor-placement-guidelines>

ZACHARIAH, Jack. *Decoupling Capacitor Optimization and Resources for Power Integrity.* Disponível em: <https://resources.altium.com/p/decoupling-capacitor-optimization-power-integrity>

Routing the PCB in Altium Designer. Disponível em: <https://www.altium.com/documentation/altium-designer/routing-the-pcb>

Altium Designer: Routing and Polygons. Disponível em: https://www.ece.ufl.edu/wp-content/uploads/AltiumDocs/Module18_RoutingandPolygons.pdf

Intelligent transport systems — Framework for collaborative Telematics Applications for Regulated commercial freight Vehicles (TARV). Disponível em: <https://www.iso.org/obp/ui/#iso:std:iso:15638:-1:ed-1:v1:en:term:4.11>

Intelligent transport systems — Cooperative ITS — Part 2: Framework overview. Disponível em: <https://www.iso.org/obp/ui/#iso:std:iso:tr:17427:-2:ed-1:v1:en:term:2.7>