

ESTUDO DO SISTEMA DE ACC E CRIAÇÃO DE UM MODELO DIDÁTICO EM ARDUINO

Lucas Carvalho Costa¹, Fabio Delatore²

¹ Engenharia de Automação e Controle, FEI

² Engenharia Elétrica, FEI

lucas_fc@hotmail.com, fdelatore@fei.edu.br

Resumo: Um problema recorrente na sociedade é o dos acidentes de trânsito, que vêm se tornando mais frequentes a cada ano de acordo com Nakagami et al [1]. Por isso, é importante que o sistema de controle de velocidade (ACC), já existente em alguns veículos, seja aprimorado. Um modelo didático de simulação do sistema permite um mais fácil entendimento da população em geral. Este estudo implementou o modelo com Arduino na linguagem C e, posteriormente, um modelo virtual no Matlab e Simulink.

1. Introdução

O sistema de *Adaptive Cruise Control*, mais conhecido por sua sigla ACC, está presente em cada vez mais automóveis modernos. Objetivamente, o mecanismo controla automaticamente a velocidade do veículo baseando-se na distância livre à frente, podendo servir para prevenir acidentes ou proporcionar uma experiência mais tranquila ao motorista.

Os dispositivos de ACC mais comuns atualmente funcionam em dois níveis, alto e baixo. O nível alto calcula a aceleração que será aplicada, com base no ritmo do veículo à frente e na distância entre eles. Já o nível baixo tem a função de converter o valor de aceleração calculado em comandos de aumento e redução de velocidade gradualmente até que o ritmo se estabilize. Se houver alguma mudança no comportamento do tráfego, o sistema se ajusta automaticamente.

No presente momento, os sistemas existentes utilizam um método linear de detecção da distância entre os dois veículos. Todavia, esse modelo ainda não proporciona uma experiência totalmente natural para os condutores pois possui irregularidades em seu funcionamento, principalmente em curvas com raio de curvatura variável e situações que envolvem mais do que dois veículos, como afunilamentos de pista e rampas de acesso. Os estudos relacionados a essas áreas ainda não foram desenvolvidos [1].

Devido aos motivos citados, é essencial que sejam feitos estudos mais específicos para solucionar esses problemas. Para isso, deve existir um modelo didático do sistema, de fácil entendimento, como ponto de partida para pesquisas futuras. O presente estudo tem esse objetivo, por meio da implantação, utilizando a plataforma Arduino, deste modelo na linguagem C e posteriormente no Simulink.

O software Matlab é um dos mais utilizados do mundo no meio acadêmico. Especializado em cálculo numérico e programação, utiliza linguagem própria para operar matrizes, funções, gráficos e dados. Dentro do programa existe o Simulink, uma ferramenta gráfica

para a modelagem e simulação de sistemas dinâmicos. É muito utilizado em processamento de sinais e controle de sistemas automáticos.

O Arduino é uma plataforma utilizada para construir sistemas digitais que podem detectar sinais e controlar objetos. É uma plataforma de hardware livre e código aberto, ou seja, personalizável e de fácil entendimento. Este componente possui pinos digitais e analógicos nos quais podem ser conectados sensores, botões, LEDs, entre outros. Esses pinos podem ser de entrada ou de saída, dependendo do componente conectado.

2. Metodologia

A proposta principal do projeto é idealizar um modelo físico e funcional que simula um sistema de ACC real. O modelo é de nível baixo, opera apenas com leitura de dados imediatos.

O modelo é constituído de uma base de madeira, um motor CC de 12V, um sensor ultrassônico HC-SR04 que detecta a distância livre à frente, um driver de potência ponte H L298N e a plataforma Arduino. De acordo com a leitura obtida pelo sensor, o driver de potência é ativado e controla o motor CC, aumentando ou reduzindo sua rotação. A ideia do sistema é reduzir progressivamente a velocidade do motor de acordo com a distância ao obstáculo mais próximo, lida pelo sensor e processada pelo Arduino.

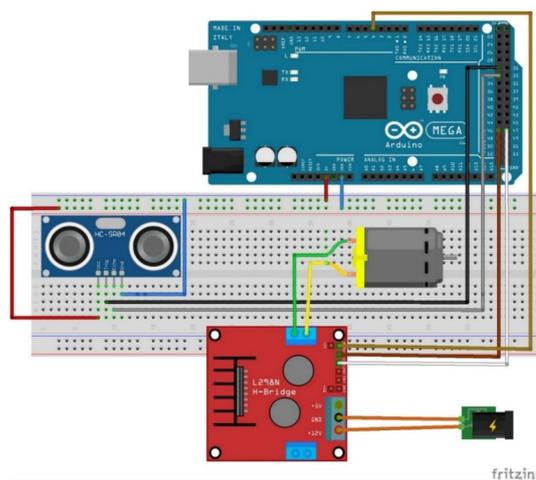


Figura 1 – Representação do sistema.

Como primeiro passo, foi efetuada a programação do sistema diretamente no software do Arduino, utilizando linguagem C. Para isso, determinou-se que as distâncias de transição, ou seja, o valor nos quais o estado do sistema varia, seriam 20 cm e 50 cm. Se a distância for maior que 50 cm o modelo funciona na

velocidade máxima. Se estiver entre 20 e 50 cm, o ritmo cai pela metade. Se alcançar um valor inferior a 20 cm, a velocidade é reduzida progressivamente até o repouso, seguindo relação diretamente proporcional à distância.

Posteriormente, foi iniciada a modelagem do circuito no Matlab e no Simulink. Neste programa o sistema foi projetado de uma forma um pouco diferente, por meio de um diagrama de blocos. O sensor ultrassônico funciona como entrada no Arduino e o driver de potência funciona como saída. Como esse último componente possui três saídas, sendo duas digitais, para girar o motor, e uma PWM para controlar sua velocidade, é necessário utilizar três blocos para a modelagem completa. Todos os sinais de entrada são processados pelo Arduino de acordo com o modelo, convertidos para tensão e enviados para as portas de saída. Com esses valores, o H L298N opera o motor, que adquire rotação real.

Para determinar os parâmetros de funcionamento do sistema, foi utilizado o bloco *Stateflow*, que opera como uma máquina de estados controlada por transições. Essas transições são ativadas por meio de valores escolhidos manualmente. Em vez dos 20 cm e 50 cm da fase inicial, foram utilizados 5 cm e 30 cm.

Por motivos de segurança, o estado inicial do modelo é o parado e assim permanece em distâncias até 5 cm. Para valores entre este e 30 cm, a velocidade do motor é controlada e varia de acordo com a distância detectada. Se este número for maior que 30 cm, a rotação do motor é máxima.

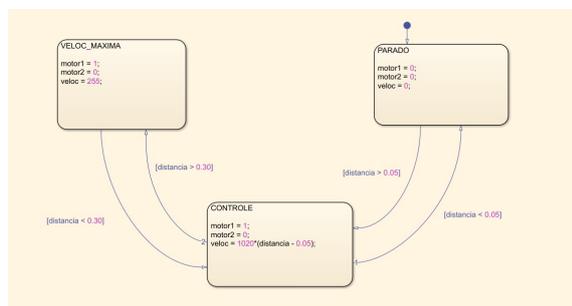


Figura 2 – Diagrama de estados *Stateflow*.

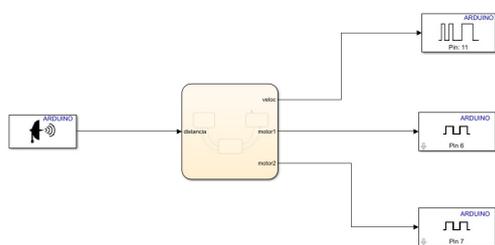


Figura 3 – Diagrama de blocos do sistema.

3. Resultados

O primeiro passo do projeto, a implementação em C, foi bem-sucedida. O código foi montado, compilado para o Arduino e foi possível observar o funcionamento

do sistema no modelo real. Nesta fase, a placa utilizada foi o Arduino Mega 2580.

Na fase seguinte, que envolveu a modelagem no Simulink, foi necessária a troca da placa para um Arduino Uno pois o Arduino Mega era propriedade de um terceiro, que o pediu de volta. Também se percebeu que a versão do Matlab que vinha sendo utilizada, a R2016a, não possuía os blocos do Simulink necessários para modelar o sistema. Por isso, houve um atraso na execução dos passos até a aquisição de uma versão mais atualizada, a R2018b.

Com a modelagem feita, houve um problema ao enviar o programa ao Arduino, pois o computador no qual o sistema estava sendo feito não reconheceu a placa e foi incapaz de se comunicar com ela. Como alternativa, realizou-se o teste em outra máquina, com o mesmo resultado. Como o Arduino Mega que havia sido utilizado na fase inicial não apresentou esse problema, a hipótese mais provável é a ocorrência de uma falha na placa Arduino Uno. Devido a essa condição, não foi possível testar o modelo do Simulink na placa física.

4. Conclusões

A fase inicial do projeto consistiu em estudos sobre o sistema ACC e a montagem do circuito físico. A FEI não possuía todos os componentes necessários para tal, portanto, foi necessário adquiri-los externamente. A montagem, subsequente programação em C e teste na placa ocorreram sem problemas.

No momento da modelagem no Simulink, houve um atraso significativo no projeto pois a versão utilizada estava desatualizada e foi necessário substituir o Arduino Mega pela sua versão Uno, mais comum. Com esses problemas resolvidos, pôde-se compor o sistema por meio de diagrama de blocos. Porém, infelizmente não foi possível realizar o teste no circuito físico por causa dos problemas acima mencionados.

Os sistemas de controle de velocidade automotivos revolucionaram a maneira com que a automação se insere no meio automobilístico. Como os estudos mais específicos ainda estão sendo desenvolvidos, a existência de uma alternativa didática faz-se necessária para que os projetos sejam mais acessíveis. Mesmo sem a realização dos testes físicos, a experiência com o estudo foi satisfatória.

5. Referências

[1] NAKAGAMI, S. *et al*, 2010: *Improvement of Adaptive Cruise Control Performance*. Shigeharu Miyata, Takashi Nakagami, Sei Kobayashi e outros. EURASIP Journal on Advances in Signal Processing, 2010. Traduzido do inglês.

Agradecimentos

À própria FEI pelo empréstimo de equipamentos e aos sites FilipeFlop e RoboCore pela aquisição de componentes.

¹ Aluno da IC 128/18 do Centro Universitário FEI. Projeto com vigência de 09/18 a 08/19.