

Sistema embarcado automotivo - Diagnóstico via interface serial

Ruan Luiz Alves da Silva¹; Ricardo Barz Sovat¹ ;
1 - Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – *Campus* Campinas;

Objetivo

Este trabalho pretende estudar as características envolvidas em um sistema embarcado automotivo, entender este tipo de diagnóstico e explorar seu monitoramento utilizando algumas tecnologias de baixo custo.

Além disto, os dados colhidos por cada sistema instalado são armazenados e podem ser tornados públicos, de modo a permitir que grupos de usuários compartilhem informações sobre modelos de veículos.

Para aplicar os conhecimentos adquiridos, está sendo implementado um sistema que permite fazer a leitura dos sensores presentes em um automóvel e guardá-las na nuvem.

Introdução

Fundamentalmente, o que o sistema proposto faz é se conectar à rede automotiva interna, efetuar algumas interações, como fazer a leitura dos principais sensores existentes, levantar e exibir localmente eventuais falhas ou anomalias eletrônicas e enviar as informações para um banco de dados.

Observando-se a indústria automotiva desde o surgimento do primeiro automóvel a manivela, até os veículos atuais, com injeção eletrônica, pode-se notar facilmente, que parte de seu funcionamento está se tornando cada vez mais informatizada (1).

Há diversos dispositivos eletrônicos no carro responsáveis por processar as informações geradas e tomar as decisões necessárias, tornando-as mais confiáveis. Assim, a informação do que acontece, por exemplo, em um carro de passeio, está cada vez mais distante do motorista, ou ao menos tornando-se menos diretamente acessível.

Portanto, é importante entender a relação de sistemas embarcados veiculares com a informática, pesquisar como a Tecnologia da Informação pode auxiliar neste ramo sem alijar o usuário do processo e ainda disponibilizar cada vez mais dados abertos para compartilhamento público.

Materiais e Métodos

Para a realização deste trabalho, foi adquirido o dispositivo ELM327 (Figura 1), que segundo a ELM Eletronics (2), faz a conversão dos principais protocolos da rede interna automotiva para uma interface serial padrão RS232, com conectividade USB ou Bluetooth.

Existem alguns protocolos que cuidam da comunicação interna veicular (1) como por exemplo o protocolo CAN. Eles permitem a condução de informações dos sensores presentes no automóvel aos dispositivos informatizados conhecidos também por ECU.

Figura 1: ELM327 Bluetooth



Fonte: Produzido pelos autores

A fim de fazer o dispositivo interagir com a rede interna automotiva, é instalado no conector On Board Diagnostics II (OBDII), localizado próximo à caixa de fusíveis (Figura 2)

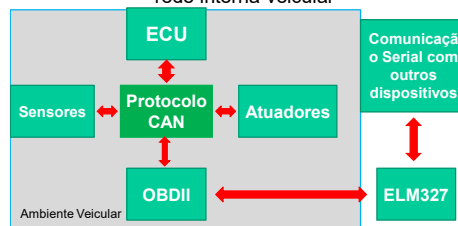
Figura 2: Conector OBDII



Fonte: Produzido pelos autores

Abaixo (Diagrama 1) encontra-se um diagrama do funcionamento deste sistema:

Diagrama 1: Representação da comunicação da rede interna veicular



Fonte: Produzido pelos autores

O software que faz a leitura dos sensores e envia os comandos via interface serial foi inicialmente construído utilizando a linguagem de programação Java, implementando as APIs Bluecove e obd-java-api. (Figura 3)

O banco de dados em nuvem encontra-se em implantação valendo-se da infraestrutura da AWS (Amazon Web Services) (3) e, no escopo deste projeto, suportará apenas o armazenamento dos dados obtidos de duas marcas de automóveis, sem permitir ainda nenhuma análise estatística ou similar. Pretende-se que este seja um trabalho futuro.

Resultados preliminares

Os desenvolvimento obtido até o momento resultou na criação de um sistema que pretende por à prova conceitos que foram estudados durante o planejamento deste projeto. O programa encontra-se em sua versão inicial, ainda sendo sucessivamente melhorado.

Também está sendo executada, no momento, uma comparação do programa gerado com uma versão em Python, uma vez que foi observada uma significativamente maior facilidade desta linguagem em obter bibliotecas relacionadas e produzir um código de compreensão mais simples.

Considerações finais

O sistema foi inicialmente desenvolvido em um notebook rodando Ubuntu. Durante a migração do ambiente de execução para um Raspberry Pi 3, foram observados alguns problemas na configuração do Bluetooth nativo do Raspbian (distribuição Linux usada).

Alguns aspectos como, por exemplo, uma certa lentidão no recebimento e envio de dados ao ELM327 para obtenção dos parâmetros, quando da execução do sistema em notebook estão sendo analisados.

Figura 3: Software responsável pela leitura dos sensores



Fonte: Produzido pelos autores

Referências

- (1) SMITH, Craig. **The Car Hacker's Handbook: a guide for the penetration tester**. San Francisco: No Starch Press, 2016.
- (2) ELM Eletronics. **The ELM327 Data Sheet**. London, Ontario, CA: Elm Electronics Inc., 2012
- (3) **AWS Storage Services Overview**. Disponível em: [https://d1.awsstatic.com/whitepapers/Storage/AWS Storage Services Whitepaper-v9.pdf](https://d1.awsstatic.com/whitepapers/Storage/AWS%20Storage%20Services%20Whitepaper-v9.pdf)
Amazon Web Services, Inc., 2016