University of Gothenburg

DIT168
Project: Industrial IT and Embedded Systems

Project Vision

Author:
Justinas Stirbys

Group 01

March 30th, 2018

1 Introduction

The project is an embedded systems project, focused on miniature car platooning. The project group, dashFTABs, has given a 3D printed miniature car, dubbed "Dash", to work with. To achieve the desired functionality Dash was equipped with several sensors. Out of which the group has utilized ultrasonic sensors and an internal IMU sensor within a BeagleBone Blue. Moreover, the group is required to contribute and integrate a Vehicle-to-Vehicle (or V2V for short) Protocol in order to communicate with cars from other project groups to achieve the aforementioned platooning.

2 Requirements

The project groups were provided with two main functional requirements; the first is to have an established protocol for inter-team communication and the second, platooning using the established protocol. Additionally, Dash must be able to support Leader and Follower roles while platooning.

The V2V protocol is to be created using a single member from each of the project groups. The said members are labeled as "V2V Managers", and excluding the V2V protocol creation, the managers are responsible for inter-team communication as well as sharing knowledge about the protocol with their individual teams.

Platooning shall be achieved through the Follower car mimicking the Leader car's movement. Thus, the Leader will be using the protocol to send data, such as steering angle as well as speed and heading readings, retrieved through the IMU sensor to the Follower. In turn, the Follower shall use this data to replicate the Leader's movement. The Follower shall also send its own sensor readings to the Leader, for the purpose of the Leader knowing that it still has a Follower. It has been decided that with the protocol information from the Leader will be sent 10 times a second, or at a frequency of 10 HZ. Moreover, for the correct/optimal platooning capabilities to be achieved the car times' must synchronized which shall be done via using "time.google.com" for the cars NTP servers.

The project incorporates microservices to achieve its desired functionality. Therefore, the major features of the project correspond to the designed microservices and vice versa. The development group has identified several required microservices, which include V2V Protocol, Ultrasonic, IMU, Web

Controller, and Maneuver microservices. The V2V Protocol microservice is discussed in the following section.

The Ultrasonic microservice shall use the ultrasonic sensor at the front of the car to implement distance checking and collision prevention. Simply put, the microservices responsibility is to calculate the ultrasonic readings and forward them to an internal communication channel on the car.

The IMU microservice is responsible for incorporating the IMU sensor, that can be found as part of the BeagleBone Blue, into its functionality. The main responsibility of this microservice is to keep track of the speed and direction of the car's movement as well as distance traveled. The microservice then communicates it to the internal OD4 channel. Thus, it is pivotal for the project success.

The Web Controller microservice is developed as a web page with two main responsibilities. Firstly, it will act as Dash's remote controller if Dash is the Leader in a platoon. Meaning, the user shall be able to use the web page to maneuver Dash. Secondly, the microservice will act as a request data visualizer. Meaning, the web page will provide an overview, or visual representation, of the messages that Dash receives, sends and process internally. This includes messages both from the Web Controller and the V2V Protocol.

The Maneuver microservice is responsible for the car's movement. To further breakdown the responsibilities, the Maneuver microservice is responsible for turning the car and making it move forward based on data from either the V2V or the Web Controller microservice. It is also worth mentioning that the car is not able to move backwards due to the hardware setup.

3 Communication Protocol

The V2V protocol is a pivotal part of the project. It is with this protocol that Dash communicates with other cars and sends it sensor readings. As such the protocol microservice communicates with the aforementioned mentioned microservices. This section describes the messages sent by the protocol and their purpose.

The V2V protocol contains several significant messages. These messages include Announce Presence, Follow Request, Follow Response, Stop Following, Leader Status and lastly Follower Status.

The individual cars must first begin by announcing their presence, which can be done using the Announce Presence. This indicates that the cars wish

to join a platoon. The message includes Dash's (or other cars') IP(s) and group number. It is significant to note that this message is broadcasted to all the groups using an OD4 session, whereas the later messages are done using UDP Sender and Receiver. This message is sent by any and all cars that wish to engage in platooning as it provides all cars with IP address of other cars.

Follow Request and Follow Response are empty messages used to create a handshake between two cars. The purpose of these messages is to establish direct communication, meaning a UDP Sender and Receiver, between two cars. As part of the Protocol it was decided to use UDP communication to avoid broadcasting all the messages to all of the cars and thus avoid "clutter". Moreover, is was decided that a car can only have a single Leader and a single Follower maximum. Therefore, the protocol includes logic to check for any existing followers. The direct communication is only established in case the car being followed does not have an already existing follower. The Follow Request is sent from the Follower to the Leader vehicle, whereas the Follower Response is sent from the Leader to the Follower.

Stop Following is a message sent from the Follower to the Leader. The message does not contain any content. However, it is used to end the direct car to car communication established by Follow Request and Follow Response. Once the message is done any cars wishing to engage in platooning again must re-announce their presence, otherwise they will not be discovererable.

Leader Status is a message sent from the Leader to the Follower. This message is sent at a frequency of 8HZ i.e. 8 times a second. The message contains the speed, steering angle, distance moved. As such this message contains information supplied by the Odometer microservice. Moreover, the message contains a timestamp of when the message was sent. The reasoning behind the timestamp was to create a "emergency scenario". The timestamp is used to compare the time sent and received. As part of the protocol it was decided that if a message is not received in 300ms the Follower car will stop.

Finally, Follower Status, similarly to Leader Status will be sent 8 times a second or every 125ms. The message does not contain any sensor information as it was deemed irrelevant to the Leader vehicle. Instead, the Follower Status serves as an indication to the Leader that the Follower still is following and that leader updates should still be sent.