vBox: Vehicle Blackbox

Project Proposal

Team Members:

Christian Loth

Jose C. Garza

Lidu Su

Prisha Srivastava

Department of Computer Science

Texas A&M University

2/8/2022

Table of Contents

1 Executive summary

In our project, the objective is to create a vehicle blackbox that can record many different aspects of the vehicle's behavior. Some of these aspects include vehicle speed, location, acceleration, and driving inputs. Through this data, we intend to create a web-application that reads and infers a driver's overall driving behavior. Additionally, this platform creates a user-interface to advise them on better driving performance based on different kinds of road conditions as well as provide insight on the vehicle's state of functionality.

To accomplish this task, we will need a GPS, IMU, an OBD, as well as a Raspberry PI to power all of the devices. The OBD will work to gather information from the processor of the car and feed it right over to our raspberry pi for documentation and comparison. The GPS will record the coordinates of the route taken throughout the trip. Finally, the IMU will be used to determine the g-forces, velocity, and acceleration of the vehicle.

Additionally, we will be transferring all raw data from the car to our web application's backend in order to translate and reflect a user's driving profile. The front-end of the application will display a user's past trips, their average speed and driving score that is determined by an algorithm that observes a driver's behavior for a period of time. Furthermore, this profile will also reflect useful data regarding the car's status such as engine check, engine readiness, fuel usage to determine behaviors such revving, speeding, excessive idling, fuel consumption etc.

The personal experiences each team member has will be used throughout the project. Since there is development in both hardware and software, the team will split into two in order to achieve as most progress as possible while using their main skills. Once both branches are developed, the whole team will be responsible for integrating them together and ensuring proper intercommunication. Finally, the team as a whole will continue working in the development and testing of the platform,

The project will have a prototyping cost of around $345.50 dollars. However, the volume production cost would be around $250 dollars. The market for a project like this is emerging therefore it would be highly profitable. Additionally, the parts are commonly available as the project does not use specialty sensors and can be found by multiple vendors. This is helpful because it makes the project highly modularized, especially as the analyzing of the data occurs on the cloud and not the device itself.

Overall, the team will be able to develop a product that will obtain data to determine the driving behaviors of drivers to encourage a safer drive practice.

2 Introduction

There is a necessity to educate drivers of their bad driving behaviors. Furthermore, bad driving habits are developed over time and are difficult to identify as they become a routine that is often ignored. Unfortunately, accidents are more likely to be expected when a driver is reckless or does not follow safe procedures when driving.

However, accidents are preventable. The Vehicle Blackbox is expected to be able to improve driving practices. The blackbox will tell us everything about the vehicle during its complete trip. RPM, location, speed, as well as many other readings will be recorded. This will help us to directly identify maneuvers made by the driver that were considered dangerous. Then the information will be presented to the driver in a way that they benefit from the information present and are encouraged to change the way they drive.

The blackbox contains a GPS module that allows the location to be tracked throughout the entire trip. Knowing all of the different locations and routes someone took can help to better understand their driving behavior. It can help determine if they take congested roads, the speed limit of the road they were traveling in. The blackbox also connects to the OBD port of the car, which provides several measurements by the ECU of the car. Some of these are useful in a driving profiling system such as the vehicle speed and engine rpms. The speed of the vehicle could be compared to the speed recorded in the database for that road.

Additionally, the blackbox contains an IMU (Inertial Measurement Unit) which will measure the aspects of the movement of the car by itself. This IMU is not plugged into the car to take data off of it but will measure its own data through inertia. If the car accelerates, the IMU will detect this. If the car decelerates, the IMU will also detect this. It can also detect the angular acceleration of the vehicle. The great thing about the IMU is that it contains gyroscopes that can allow it to do what it needs to do, and allow for very precise measurements.

Overall, the blackbox will contain the necessary components to obtain data to draw conclusions from the behavior of the driver. The conclusions will be displayed to the driver through a web application. A scoring system will be presented to them depending on their actions.

2.1 Needs statement

The problem our design will address is the necessity of drivers to recognize their bad driving habits to determine the areas the drivers need to improve. This will make them aware of the consequences their driving could cause. Additionally, the drivers need a way to have access to the data in a way it shows them the decisions they took at certain times.

2.2 Goal and objectives

The goal of this project is to encourage healthy and responsible driving practices through automatic, real-time monitoring systems. Incorporating a user-specific interface creates an accurate profile such that it increases self-awareness. We hope that by creating a time sensitive profile algorithm, we are able to realistically grasp a driver's current driving behavior and through consistent monitoring, reduce their chances of collisions or accidents.

Our objectives are to record all of this data onto a Raspberry PI to document it, as well as convert this data into quantitative data to display on a web server output. This way, the durable blackbox data can easily be exported and converted into words which describe the vehicle's behavior.

2.3 Design constraints and feasibility

Technical: We are working with four prominent sensors/ GPIO -- IMU, OBD, GPS and Camera. Our purpose is to create a blackbox such that allows extension of our sensors to their required spots outside the box (if necessary) without affecting the connectivity with Raspberry PI microcomputer. We can achieve this through spaced out device connectivities and proper soldering. Additionally, we will be relying on a combination of wired and wireless sensor implementation (i.e OBD) to reduce wiring issues.

Physical: We are designing our prototype with compactness in mind, in order to make our product accessible for various vehicles. Our outside “hull” for our microcomputer will be 3-D printed such that it contains ports for outside connections and a space to fit the raspberry pi inside.

Economical: For our project, we are provided with a $600 budget that will be used towards purchase of required materials and unexpected costs in manufacturing, prototyping and testing. Additionally, our remaining funds will be saved for emergency purposes such as software purchases, deployment and testing costs, if necessary.

Time: The estimated completion date for this Project is April 30,2022. We will achieve this timely development through bi-weekly sprint meetings and burn-up charts to help us identify our weekly tasks and necessary insight on how to maintain a realistic and efficient project timeline while complying with Agile methodology,

3 Literature and technical survey

1. In 2010, there was a vehicle black box system designed by the team of *Ahmad Asi, Benjimin Chang, Mehdi Dadfarnia, Serge Kamta, and Ifzalul Khan*. The vehicle black box designed by them was supposed to withstand heavy impact and record audio and video data with high quality and resolution. The design was composed of a high-speed solid-state drive(SSD) to store the data, a vehicle data recorder to collect the pressure sensor data and GPS/date time data, and a data rotator software to help refrain from potential data loss. Once periodical data collection is complete, the software downloads data from the black box and reliefs the local data.

The black box was designed for military vehicles and field engineers, and allows remote monitoring of vehicle status. The black-box system needs multiple cameras to record video data from any angle around the vehicle, and should be made robust so it can endure extremely high temperature and any vibration in the field environment. It should also be able to endure explosions.

As for the budget, since it is funded by the U.S military and requires reliable impact tolerance and high-quality data, it is relatively more expensive than most similar products in the market. Compared to our project, which requires only one camera in the front (might need another one for the rear view) and relies primarily on a Raspberry Pi 4 to process the data, this vehicle black box design is more complicated, since it takes into account more occasions and situations that ordinary vehicles are not going to encounter.

2. Another design in 2021 took a different approach, the team built a vehicle black box system based on IoT. The hardware components are rather simple compared to the previous one. The process of sending data to IoT is handled by an ESP8266 chip. Proposed system uses an Arduino board that provides an easy burning/uploading of a program. To monitor the various sensors such as alcohol sensor, temperature sensor, light sensor, accelerometer, ultrasonic sensor, GPS are connected to Arduino board. The Arduino board is connected to the cloud. The output of the sensors is read from Arduino and output values are displayed in LCD. The data is stored in the cloud the given system is proposed in IoT. This project also uses a solid-state drive for data storage which possibly makes it a little more expensive.

3. Another paper published by IEEE also provides inspiration in system design based on IoT. However, this black box system has not only a cloud platform that can track the data real-time, but also mail and short message services for data retrieval. Just like the other products, The crash and its likely location are

sent for medical assistance. This paper mainly focuses on improving the care of victims of the crash, helping, and easily detect fraud.

4. The majority of vehicle black box products available online are merely dashcams, the only functionality they have is recording video and audio data. Take the Vehicle Black Box- Full HD1080p DVR from Beehive Cheese as an example, it is composed of a memory card, a camera, and a microphone – a rather simple structure. The functionality of it is obviously quite limited to meet the needs of a proper vehicle black box. Compared to the black box designs we mentioned previously, the only advantage of this kind of dashcams is their relatively low price, for their typical price range is below 30$.
Although not a black box, there are a lot of vehicles that come with an event data recorder (EDR). The 5. EDR provides vehicle data such as engine mpr, gas mileage, steering wheel rotation angle and brake paddle status. The difference between an EDR and a black box is that it does not record video or audio data. The list of vehicles with built-in EDR can be found at: https://rimkus.com/media/pdfs/Event-Data-Recorder-Vehicle-List-Rimkus-1.11.21.pdf.

In general, the goal of our project is to build an affordable vehicle black box with the functionality of collecting vehicle data as well as monitoring it. From the products we've reviewed above, we are confident that the product of ours is going to be more powerful and efficient than most of the products available online, and also cheaper than the ones that exceed daily usage.

4 Proposed work

4.1 Evaluation of alternative solutions

1. Dash Cams can only detect the cars in front of the driver, limiting the amount of data around overall driving. The integral purpose of dash cams allows users to record collisions for insurance purposes. However, when identifying a driver's behavior, it is important to look at many different factors to make a logical conclusion. For instance, our vehicle blackbox incorporates various elements of driving such as location tracking, speed monitoring and IMU to understand a driving pattern holistically. However, standard dash cams lack the implementation of GPS, IMU and OBD. Often, They are only helpful in the case of collision, as they keep videographic proof of lanes leading upto the collision. Therefore, it cannot identify the driver's behavior because anyone can get into a bad accident, whether they are good or bad drivers.
2. Front and rear camera- Front and rear camera combination is useful in providing more data on lanes and surrounding environment. However, in order to make this system useful for the project,

we need to process these recordings through image recognition. Image recognition is expensive and difficult, making it a tough starting point for our project. Since Image recognition is a machine learning algorithm, it does not guarantee accuracy and often requires training models several times on lane detection to get it to work well. Image recognition cannot solely be a good indicator of driving behavior because it cannot detect speeding. Although lane detection can be implemented through cameras to make predictions about potential collisions, it limits insightful data on a user's holistic driving behavior.

3. A standard Inertial Measurement Unit System can allow users to identify the car's acceleration, current speed, turn rate or inclination. However, receiving such data solely creates a lot of gaps and concerns, such as "What conditions caused the driver to speed a certain way?". When we observe speed and turn rate of cars without context, we lose a lot of data on circumstantial evidence on a driver's behavior. Hypotheticals such as a user making a sharp turn to avoid incoming collision can be seen as a responsible move. But solely depending on IMU may allow the system to conclude that a user is possibly a harsh driver. In conclusion, IMU lacks insight from road conditions (i.e bumpy, traffic giving inconsistent speed reading) needed for an advanced driving detection system.
4. GPS based web-application: Although GPS is useful in detecting speed and road traveled to destination, A GPS based driving detection system might not be so useful after all. The OBD port in the car provides many helpful features such as real-time speed monitoring, status of "check engine", emission readiness and freeze frame of data at the time of collision. The issue with only using GPS is that it limits the understanding of car conditions that can affect a user's driving ability or profile. Location tracking and speed detection through GPS may lack context in situations where driving above or below the speed limit is realistic. For instance, driving below 20 mph in a neighborhood because of people driving 10mph above speed limit on the highway to match everyone else on the highway. Additionally, OBD derived data can help us understand if a user has been ignoring car issues, something that a GPS system cannot provide insight on. Overall, having a real-time data collection of cars and driving is needed for this design problem to be solved.
5. Speeding Alert System: Creating an alarm system that sounds an alarm each time a user goes above speed limit can result in distracted drivers, causing them to be frustrated and reduce their focus from responsible driving. highway driving would be difficult. Alert systems do not accurately conclude a driver's ability as they lack a lot of situational awareness and context that need to be factored in when assessing someone's driving behavior.

4.2 Design specifications

At the top level, the vBox will be used while driving and will store information provided by the GPS, IMU, Camera, and OBD port and store it locally within the microcontroller, which will then be sent to the server and then displayed to the user as analyzed data. The microcontroller will run a script that calls the AWS cloud service to upload the data, and the data stored locally will be wiped out for further data collection.

High-Level Block Diagram:

Hardware:

Our hardware will consist of the following:

Raspberry Pi: This microcontroller will continuously obtain data from the OBD Interface, IMU, GPS, and the camera module. The data will be stored locally in an SD card until it is transmitted to the web server.

OBD Bluetooth Interface: The OBD port that is present on all cars in the United States will provide data from the ECU to the Raspberry Pi. This port will provide data such as the speed of the vehicle, the rpms of the engine, the throttle position, fuel consumption, among others. These metrics will be important to determine driving behaviors as higher rpms could mean higher accelerations which in turn is an unsafe driving behavior.

IMU (Gyroscope and Accelerometer): The IMU will be connected to the Raspberry Pi and will serve to measure g-forces, orientation, position and velocity. This will help determine abrupt change of lanes or high speed corners. In addition, it can be used to determine a hard deceleration or an abrupt acceleration.

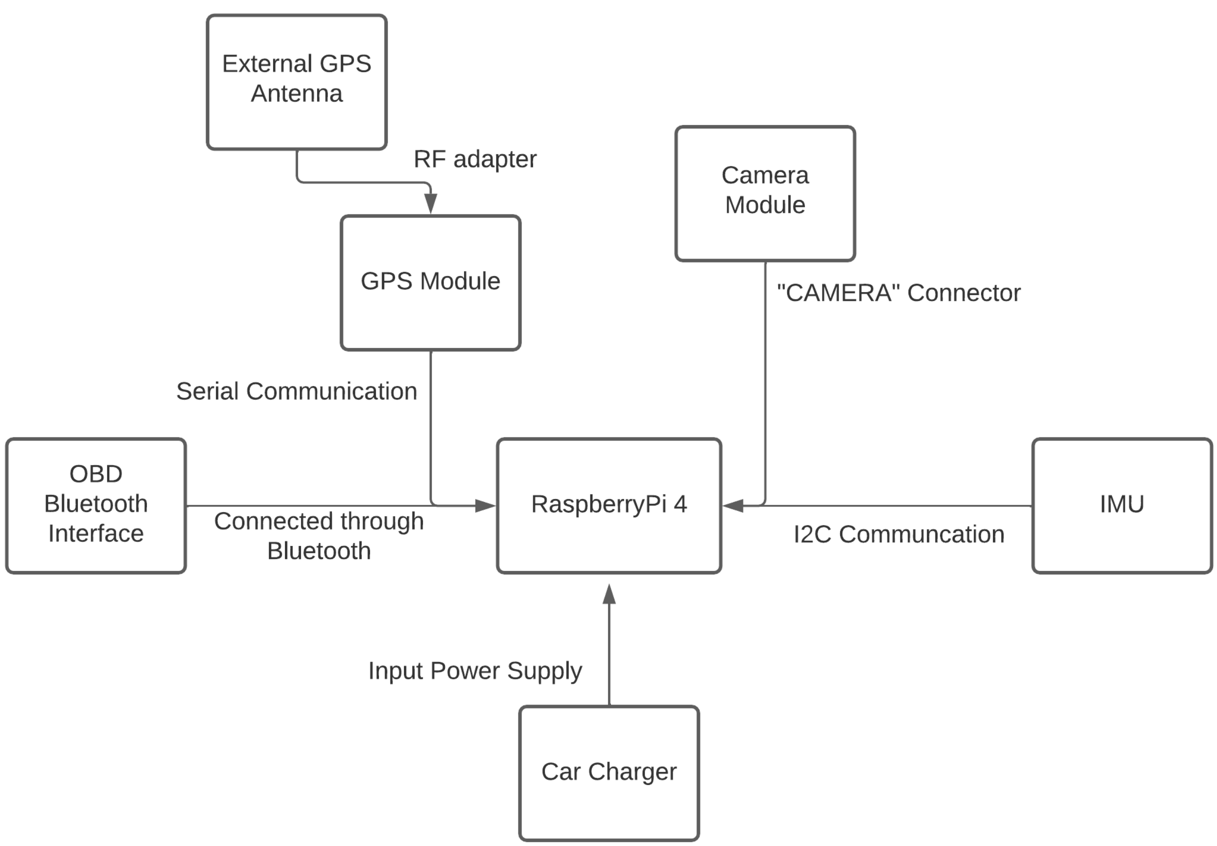
The GPS Hat: The GPS hat for the Raspberry Pi will provide us with raw data in the form of NMEA sentences. This will be then parsed to obtain time, date, latitude, longitude, altitude, estimated land speed.

The GPS hat will be used in combination with the Google Maps API to map trips and determine if the driver was possibly speeding. In addition, depending on the availability of the information provided by the Google Maps API, it is possible to determine if the driver takes routes that are typically congested.

External GPS Antenna: The external GPS antenna provides a more accurate positioning of the device. We will be using the antenna attached to the very top of the windshield in order to not obstruct the view of the driver.

A Camera module: The camera will point towards the outside of the car. It's capability of being wide angle will provide us with the ability of having a view as close as possible to the one of the driver. The camera will serve as a constant recording device that will keep track of any dangerous situations encountered throughout the drive.

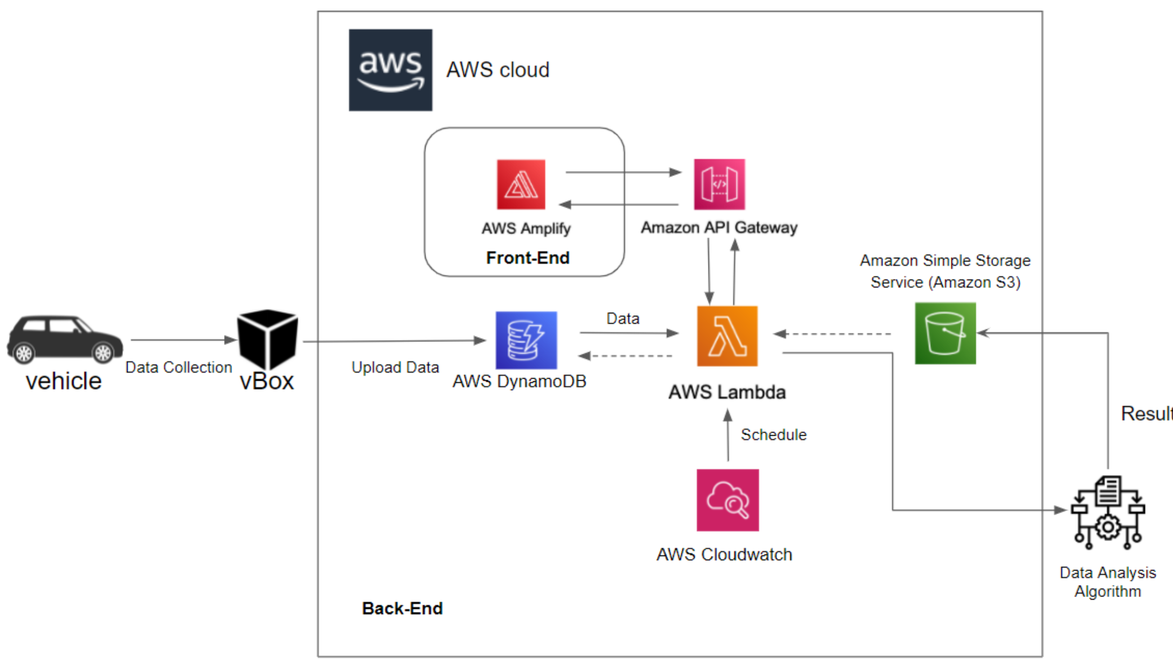
The following diagram displays the hardware interfaces interconnected and their method of communication:



Software:

The back-end of the project will be built by using AWS cloud services. AWS provides us with various tools that will be useful:

- AWS Amplify: AWS Amplify provided us with efficient and convenient tools to deploy our website application.
- Amazon API Gateway: This tool lets us create our own API that will be called by the web application. The API Gateway invokes the lambda functions to process the data stored in DynamoDB and returns the results to our front end.
- AWS Lambda: Lambda functions are powerful and user-friendly scripts that allow us to run complicated methods on the cloud side instead of locally. It improves efficiency and saves computing resources for our local devices.
- AWS DynamoDB: This is going to be our major storage for vehicle data. The powerful NoSQL allows fast transfer of data and is also easy to use.



The front-end will call the AWS API Gateway for the analysis of the driver. Since the results are already stored in the S3 bucket, we only need to retrieve the results from it via the lambda function. The result is going to be a json file, so we can easily parse the data using javascript and embed it into our web application.

4.3 Approach for design validation

Our design is intended to transition from prototype to final deployment through a series of layered testing that allows us to thoroughly examine the functionality of hardware and software aspects of our project. For the purpose of our approach, we will divide our testing focus into three different categories--Hardware Testing, Software Testing and User Case Scenario.

1. Physical Hardware Testing: Enforcing the reliability of our blackbox by ensuring that the box is able to fit all ports, Raspi is connected to all GPIO and USB/USB-C connections. Run simple python commands on Raspi terminal to check each sensor status. Use multimeter to check soldering connection
2. Web Application Testing: We need to validate real-time data retrieval by checking if DBMS and other Backend Services are receiving new data. This can be achieved through viewing 'Transaction Logs' in SQL Server. They record timestamps of each new entry. Then check if the profile on the frontend displays these changes through local testing.
3. Usage Scenarios: Brainstorming a few testing scenarios will allow us to identify edge cases and train our software model better based on the data it receives from the car.
    a. Going through a slow-speed area, such as a school zone: Testing in this area allows us to identify how the web application assesses slower and more brake-induced driving. We can test this by creating a manual baseline for this scenario and compare our results with this.
    b. Going through a smooth, fast-lane road (i.e highway/tollway): Testing in this area allows us to identify how the software assesses faster driving due to circumstantial reasons. Since, drivers typically maintain a speed higher than the limit on tollways and highways, we can use the GPS to locate if the user is currently on a highway and use OBD data to check if the user is within 1.10% of the speed limit.
    c. Going through high-traffic areas at peak hours: Using our IMU and GPS, we will analyze the road condition and braking behaviors to assess if the user is rightfully slowing down or displaying abnormal driving behavior due to circumstances.
    d. Going through rough terrain: Rough terrain can cause sharp turns, sudden/consistent brakes or slower speeds. When our GPS detects such roads, we will change our metrics for driver performance to test if the user is adjusting to change of road conditions.

5 Engineering standards

5.1 Project management

The team consists of four members: Jose C. Garza, Lids Su, Prisha Srivastava and Christian Loth. Each one of the members is a Computer Engineering major and has the qualifications as well as the determination necessary to accomplish this project.

During the project, the team will use the following helper tools to manage the information and progress of the project:

- Github repository: The repository will contain all the documents, code and hardware design material that the team develops throughout the project..
- Google Shared Drive: It will be used as a collaboration tool for documents as it provides a simple way of all members to contribute.
- Slack Channel: It will be our main method of communication for project management decisions such as meeting times, quick questions, suggestions, among others. In this platform the members will communicate daily updates on the tasks they were assigned.

Furthermore, the main oversight of the main components of the project will be divided between the team members in the following way:

Project Team Leader: Jose C. Garza. Throughout the project he will be responsible for turning in any necessary materials before the deadline. He will ensure that the project is on track for completion by the end of the semester. Additionally, he will be the main point of contact for the project in the event of any inquiries by the faculty, ordering parts, etc. During the project, the team leader will keep an updated Gann chart, project budget, and this information will be easily accessible for the other members. Besides management duties, Jose will be responsible for interconnecting all the hardware together with the main microcontroller. Jose has the qualifications necessary to be the team leader as he has experience in previous leadership roles. Additionally, his internship experience consisted of connecting high-level hardware with multiple other modules.

System Designer/Engineer: Prisha Srivastava will be responsible for the System Design ensuring the hardware and the software integrate work with each other. Prisha will also work closely with the design and production of the 3D Printed enclosure for the product. Prisha has the analytical skills necessary to understand the system from a higher perspective and identify any issues that could develop due to her internship experience as a Software Engineer. Additionally, her experience in SolidWorks will be essential for the modeling of the enclosure of the Raspberry Pi and its modules.

Front-End Software Designer/Engineer: Christian Loth will be responsible for working closely with Lide Su to achieve a working web application that displays the information and conclusions gathered from the trip data. Christian has the necessary skills to work in the front-end of the platform as he has developed multiple web applications. He also has experience using API calls and the back-end that will be useful for him to interconnect the front-end with the back-end.

Back-End Software Designer/Engineer: Lide Su has experience using AWS and other Amazon AWS services, therefore he will be responsible for ensuring the back-end software design of the project is in the direction intended to. Some of his tasks include: setting up the AWS server, ensuring integration with the APIs we choose for the project, as well as working with Christian on how to obtain and analyze the data for the web application

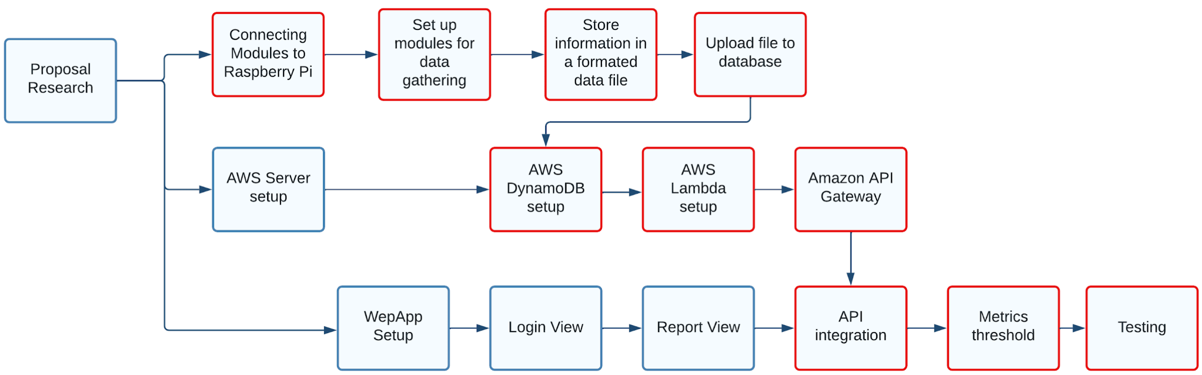
5.2 Schedule of tasks, Pert and Gantt charts

The table below shows the schedule of tasks:

Task #	Task Description	Start Date	End Date
1	Web Application starter code	02/17/2022	02/22/2022
2	Dashboard Design	02/22/2022	02/24/2022
3	Assemble and install Camera	02/24/2022	02/28/2022
4	Assemble and install IMU	02/24/2022	02/28/2022
5	Assemble and install GPS	02/24/2022	02/28/2022
6	Connect Raspberry Pi to Server	02/24/2022	02/28/2022
7	Connect OBD port interface	02/24/2022	02/28/2022
8	Setup AWS Database	02/24/2022	03/10/2022
9	Setup AWS Lambda & API	02/24/2022	03/10/2022
10	Obtain raw data from modules	02/28/2022	03/03/2022
11	Integrate OBD python library	02/28/2022	03/07/2022
12	Concurrently obtain data from GPS and OBD	03/08/2022	03/15/2022
13	Format data file	03/10/2022	03/24/2022
14	Record Camera video and store as needed	03/10/2022	03/20/2022
15	Retrieve data and make error handlings	03/15/2022	03/29/2022
16	WebApp Google API integration	03/10/2022	03/17/2022
17	WebApp: Login View	03/10/2022	03/17/2022

18	WebApp: Report View	03/10/2022	03/22/2022
19	Implement uploading data to server	03/17/2022	03/24/2022
20	Develop analyzing algorithm	03/22/2022	04/05/2022
21	Testing-WebApp displaying retrieved data	03/24/2022	04/07/202
22	Determine metrics for reckless behavior	04/05/2022	04/12/2022
23	Final testing	04/05/2022	04/19/2022
24	Correct/Perform minor improvements	04/14/2022	04/21/2022

Pert Chart displaying critical path:

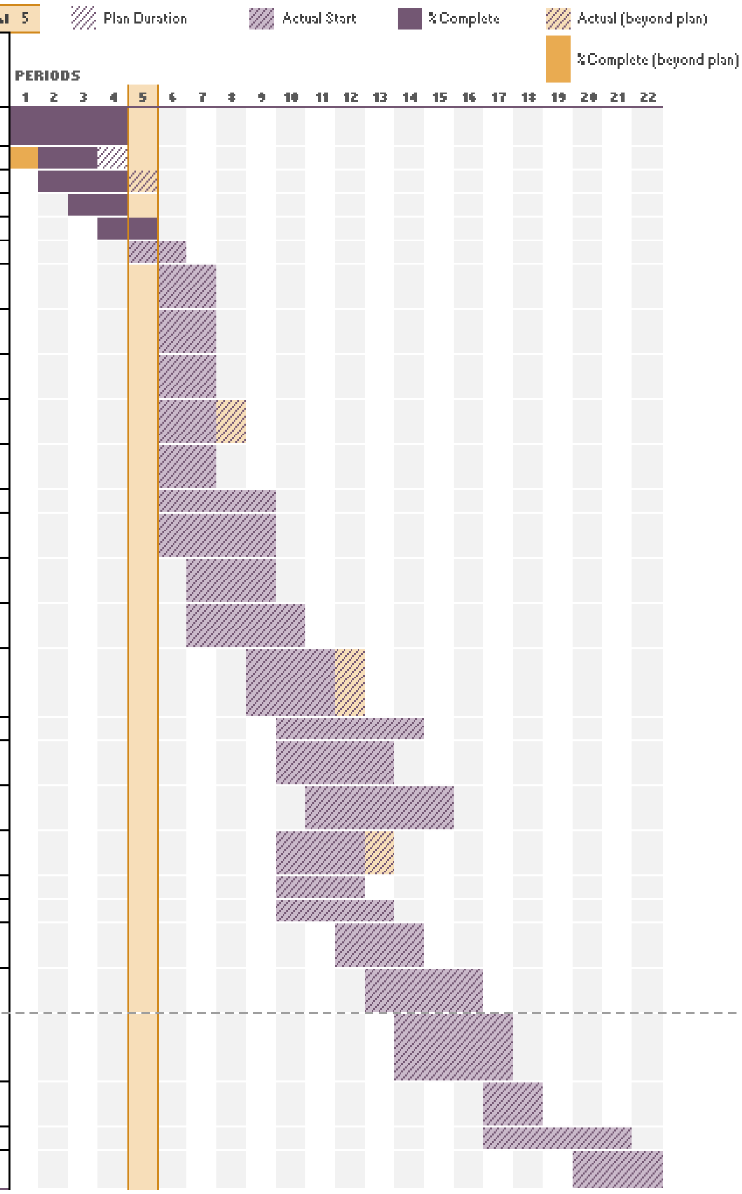


The Gantt Chart for the project is the following (next page):

Project Planner: Vehicle Blackbox

Select a period to highlight at right. A legend describing the charting follows. **Period Highl** 5

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Proposal	1	4	1	4	100%
Research	2	3	1	3	100%
Research	2	3	2	4	75%
Order materials	3	2	3	2	100%
WebApp starter	4	2	4	2	100%
Dashboard	5	2	5	2	0%
Assemble and install Camera	6	2	6	2	0%
Assemble and install IMU	6	2	6	2	0%
Assemble and install GPS	6	2	6	2	0%
Connect Raspberry Pi to	6	2	6	3	0%
Connect OBD port interface	6	2	6	2	0%
Setup AWS	6	4	6	4	0%
Setup AWS Lambda & API	6	4	6	4	0%
Obtain raw data from modules	7	3	7	3	0%
Integrate OBD python library	7	4	7	4	0%
Concurrently obtain data from GPS and	9	3	9	4	0%
Format data file	10	5	10	5	0%
Record Camera video and store	10	4	10	4	0%
Retrieve data and make error	11	5	11	5	0%
WebApp Google API integration	10	3	10	4	0%
WebApp: Login	10	3	10	3	0%
WebApp:	10	4	10	4	0%
Implement uploading data	12	3	12	3	0%
Develop analyzing	13	4	13	4	0%
Testing-WebApp displaying	14	4	14	4	0%
Determine metrics for	17	2	17	2	0%
Final testing	17	5	17	5	0%
Correct/Perform minor	20	3	20	3	0%



5.3 Economic analysis

Economical Viability: Our project is a complete system that incorporates both hardware and web platform. To date, similar projects with relatively low cost and powerful functionality appear to be pretty rare in the market. Therefore, our project has a huge potential of commercialization given the possible lower cost per product when put into volume production. The prototype costs about $345.50, but our expected cost of volume production would be around $250 or lower.

Sustainability: The system parts are ordinary parts that could be found from more than one vendor in the market. The parts need to be taken care of regularly, and replaced should there be anything failing. As for the source code, the host needs to regularly check the functionality and update if needed. The product needs support from AWS.

Manufacturability: Our project is highly modularized and flexible. The data is handled on the cloud side so the processor part can also be replaced into other models or products satisfying basic computing needs without harming system performance. The worst case scenario occurs when the processing unit (Raspberry Pi 4 in our case) fails, and therefore this most expensive part has to be replaced. We expect the production yield to be higher than 95%, for we use simple yet reliable parts to ensure affordable replacement and instant repair. Our source code would be transparent and we welcome any investigation should there be any proper requests from authority. We promise we will not collect any data exceeding the needs of our product without authorization from the client.

5.4 Societal, safety and environmental analysis

Society would benefit from the vehicle blackbox because it can be used as a product to encourage safe driving practices. Since the blackbox records information from the car and analyzes it to determine hard breaks, close proximity with other vehicles, vehicle speed, rpms, among other things. However, there is a detrimental aspect which is the necessity to record data of the user's driving habits. While the data will be as secure as possible within the server.

While working on this project there will be several precautions to have in mind. The device will have to be tested to ensure it is capable of achieving its intended purpose. The team should be aware at all times of the driving conditions to ensure there is minimal risk of damages and distractions to the driver. The driver should be focused on driving, while the rest of the team can interact with the device for debugging or data collection purposes. In addition, there might be the possibility of necessary unsafe maneuvers in order to determine metrics to quantify data for visualization and determine the driving behavior threshold of the driver.

The project will have minimal damage to the environment. In production, the device could be reassigned to other drivers. During testing, driving for testing purposes could possess the biggest impact on the environment. However, this could be minimized by recording raw data from a trip and reusing the data to simulate a current driving situation.

5.5 Itemized budget

The product to be developed will use the following hardware:

Description	Price	Quantity
Raspberry Pi 4 Camera	$24.59	1
Adafruit GPS Hat	$29.95	1
External GPS antenna	$19.95	1
IMU (Accelerometer & Gyro)	$14.95	1
Standoffs for Pi HATs	$0.75	1
RF Adapter cable	$3.95	1
Bluetooth interface OBD port	$13.99	1
Raspberry Pi 4	$167.95	1
SD card with 512 GB	$29.99	1
Car charger for Raspberry Pi 4	$14.44	1
Suction Cup Mount	$24.99	1
3D Printed Enclosure	$25.00	1

This yields a total of approximately $345.50 dollars in parts. The 3D printed enclosure will be fabricated at the FEDC.The GPS Hat for the Raspberry Pi and the IMU will require minor soldering work in order to neatly and space efficiently connect directly on top of the Raspberry Pi. However, the camera and the external GPS are plug-and-play. Since the soldering work is minor, it can be considered insignificant to the cost of the project. Finally, in the software side of the product, it is expected that only the following two components will be used:

Description	Price
AWS Server	Free Credits
Damoov API (tentative)	Free Credits
Google API	Free Credits
Open Source OBD python library	Free - Open Source

Due to the credits provided by both platforms, there will not be a cost in the software side of the project.

With the cost outlined above, there still $255 dollars left that could be used if any roadblocks or unexpected costs are encountered during the course of the product development.

6 References

S. Sethuraman and S. Santhanalakshmi, "Implementing Vehicle Black Box System by IoT based approach," *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, 2020, pp. 390-395, doi: 10.1109/ICOEI48184.2020.9142906.

Ahmad Asi, Benjimin Chang, Mehdi Dadfarnia, Serge Kamta, and Ifzalul Khan, “Black Box System Design,” Accessed: Feb, 15. 2022. [online]. Available: *https://user.eng.umd.edu/~austin/enes489p/projects2010b/BlackBox-FinalReport.pdf*

S.Monika, S.Miruthulasri, R.Mano Priya, and D.Murugesan, “Automobiles Based Black -Box System Using IoT,” *International Research Journal for Engineering and Technology*, *volume 8, issue 4,* April 2021, https://www.irjet.net/archives/V8/i4/IRJET-V8I449.pdf

7 Appendices

7.1 Product datasheets

- Raspberry Pi 4

Specification

Processor:	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
Memory:	1GB, 2GB, 4GB or 8GB LPDDR4 (depending on model) with on-die ECC
Connectivity:	2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless LAN, Bluetooth 5.0, BLE Gigabit Ethernet 2 × USB 3.0 ports 2 × USB 2.0 ports.
GPIO:	Standard 40-pin GPIO header (fully backwards-compatible with previous boards)
Video & sound:	2 × micro HDMI ports (up to 4Kp60 supported) 2-lane MIPI DSI display port 2-lane MIPI CSI camera port 4-pole stereo audio and composite video port
Multimedia:	H.265 (4Kp60 decode); H.264 (1080p60 decode, 1080p30 encode); OpenGL ES, 3.0 graphics
SD card support:	Micro SD card slot for loading operating system and data storage
Input power:	5V DC via USB-C connector (minimum 3A[1]) 5V DC via GPIO header (minimum 3A[1]) Power over Ethernet (PoE)–enabled (requires separate PoE HAT)
Environment:	Operating temperature 0–50°C
Compliance:	For a full list of local and regional product approvals, please visit https://www.raspberrypi.org/documentation/hardware/raspberrypi/conformity.md
Production lifetime:	The Raspberry Pi 4 Model B will remain in production until at least January 2026.

- MakerFocus Raspberry Pi 4 Camera

Description:
The camera is equipped with photosensitive resistance, which enables the camera to automatically detect and recognize light to automatically switch between the night-vision and day-time shooting mode.
Manual settings are not required, plug-in then itcan be used.
The advantage is that during the day, the photographs taken will not be reddish, which also eliminates the need for manual switching.

Specifications:
Infraed Camera:
Pixels: 5 Megapixels
Photosensitive chip: OV5647
Focal Length: 1.7 mm
Aperture (F): 2.0
FOV (Diagonal): 175 degrees
Sensor optimum resolution: 1080p
Size:11.7*6.6*3.6 cm
Product weight: 0.028kg

Infrared Light: can fill flash and it's waveband is 850 nm.
Photoresistor: to detect ambient light intensity.

Package Including:
1 * Fisheye Infraed Camera
2 * LED Lights
1 * 15cm FFC cable
4 * screws and nuts

- GPS Module

	Description
GPS Solution	MTK MT3339
Frequency	L1, 1575.42MHz
Sensitivity[1]	Acquisition: -148dBm, cold start Reacquisition: -163dBm, Hot start Tracking: -165dBm
Channel	66 channels
TTFF	Hot start: 1 second typical Warm start: 33 seconds typical Cold start: 35 seconds typical (No. of SVs>4, C/N>40dB, PDop<1.5)
Position Accuracy	Without aid:3.0m (50% CEP) DGPS(SBAS(WAAS,EGNOS,MSAS)):2.5m (50% CEP)
Velocity Accuracy	Without aid : 0.1m/s DGPS(SBAS(WAAS,EGNOS,MSAS,GAGAN)):0.05m/s
Timing Accuracy (1PPS Output)	10 ns(Typical)
Altitude	Maximum 18,000m (60,000 feet)
Velocity	Maximum 515m/s (1000 knots)
Acceleration	Maximum 4G
Update Rate	1Hz (default), maximum 10Hz
Baud Rate	9600 bps (default)
DGPS	SBAS(defult) [WAAS, EGNOS, MSAS,GAGAN]
QZSS	Support(Ranging)
AGPS	Support
Power Supply	VCC：3.0V to 4.3V；VBACKUP：2.0V to 4.3V
Current Consumption	25mA acquisition, 20mA tracking
Working Temperature	-40 °C to +85 °C
Dimension	16 x 16x 4.7mm, SMD
Weight	4g

- GPS External Antenna

Ceramic Path Specification:

Operating Frequency	T1 1575.42±1.023 MHz
Output Impedance	50 ohms
Polarization	R.H.C.P.
Bandwidth	10 MHz min. @S11<=-10 dB
Gain at 10° elevation	-1 dBic Typ.
Axial Ratio	3.0 dB Typ.

LNA/Filter Specification:

Operating Frequency	T1 1575.42±1.023 MHz
Gain	28 dB
Noise Figure	1.5 dB
Filter	DR Filter
	20dB 30dB min @ fo±50MHz
	30dB 35dB min @ fo±50MHz
	* fo=1575.42 MHz
Output VSWR	2.0 Max
Voltage	2.3~5.5V
Current	2.5V : 6.6mA Typical
	3V : 8.6mA Typical
	4V : 12.6mA Typical
	5V : 16.6mA Typical

General specification:

Dimensions	L 41.2xW38.5xH13.3 mm
Mount	Magnetic Antenna
Antenna Color	Black
Coaxial Cable	RG174 Length=5M (Option)
Cable Connector	SMA MALE (Option)
Operating Temperature	-30°C to +85°C
Storage Temperature	-40°C to +90°C

- IMU Accelerometer and Gyro

InvenSense
Sensing Everything

ICM-20649

World's First Wide-Range 6-Axis MEMS MotionTracking™ Device for Sports and High Impact Applications

GENERAL DESCRIPTION

Many of today's wearable and sports solutions, which analyze the motion of a user's golf or tennis swings, soccer ball kicks, or basketball activities, require higher than currently available ±2000 dps (degrees per second) FSR for gyroscope and ±16*g* FSR for accelerometer to better insure that critical data is not lost at the point of high impact or high speed rotation. The ICM-20649 6-axis inertial sensor offers the smallest size, lowest profile and lowest power in conjunction with industry leading high FSR.

With an extended FSR range of ±4000 dps for gyroscope and ±30*g* for accelerometer, the ICM-20649 enables precise analysis of contact sports applications providing continuous motion sensor data before, during and after impact providing more accurate feedback

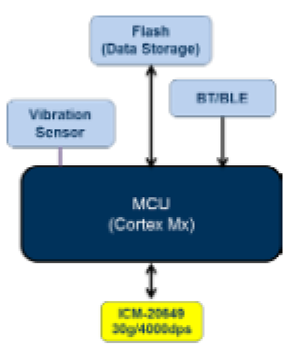
The ICM-20649 is the world's first wide-range 6-axis MotionTracking device for Sports and other High Impact applications. It is available in a 3x3x0.9 mm 24-pin QFN package.

ORDERING INFORMATION

PART	TEMP RANGE	PACKAGE
ICM-20649†	−40°C to +85°C	24-Pin QFN

†Denotes RoHS and Green-Compliant Package

BLOCK DIAGRAM



Motion Analysis Pod Architecture

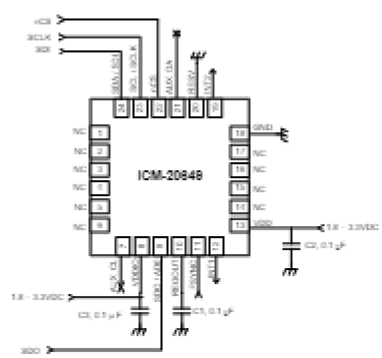
APPLICATIONS

- Sports
- Wearable Sensors
- High Impact Applications

FEATURES

- 3-Axis gyroscope with programmable FSR of ±500 dps, ±100 dps, ±2000 dps, and ±4000 dps
- 3-Axis accelerometer with programmable FSR of ±4*g*, ±8*g*, ±16*g*, and ±30*g*
- User-programmable interrupts
- Wake-on-motion interrupt for low power operation of applications processor
- 512-byte FIFO buffer enables the applications processor to read the data in bursts
- On-Chip 16-bit ADCs and Programmable Filters
- DMP Enabled:
  - SMD, Step Count, Step Detect, Activity Classifier, RV, GRV
  - Calibration of accel/gyro/compass
- Host interface: 7 MHz SPI or 400 kHz I²C
- Digital-output temperature sensor
- VDD operating range of 1.71V to 3.6V
- MEMS structure hermetically sealed and bonded at wafer level
- RoHS and Green compliant

TYPICAL OPERATING CIRCUIT



This document contains information on a pre-production product. InvenSense Inc. reserves the right to change specifications and information herein without notice.

InvenSense Inc.
1745 Technology Drive, San Jose, CA 95110 U.S.A
+1(408) 988-7339
www.invensense.com

Document Number: DS-000192
Revision: 1.0
Revision Date: 12/13/2016

- OBD Bluetooth

ELM327

OBD to RS232 Interpreter

Description

Almost all of the automobiles produced today are required, by law, to provide an interface for the connection of diagnostic test equipment. The data transfer on these interfaces follow several standards, but none of them are directly usable by PCs or smart devices. The ELM327 is designed to act as a bridge between these On-Board Diagnostics (OBD) ports and a standard RS232 serial interface.

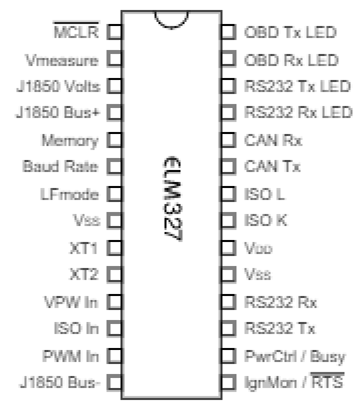
In addition to being able to automatically detect and interpret nine OBD protocols, the ELM327 also provides support for high speed communications, a low power sleep mode, and the J1939 truck and bus standard. It is also completely customizable, should you wish to alter it to more closely suit your needs.

The following pages discuss all of the ELM327's features in detail, how to use it and configure it, as well as providing some background information on the protocols that are supported. There are also schematic diagrams and tips to help you to interface to microprocessors, construct a basic scan tool, and to use the low power mode.

Features

- Power Control with standby mode
- Universal serial (RS232) interface
- Automatically searches for protocols
- Fully configurable with AT commands
- Low power CMOS design

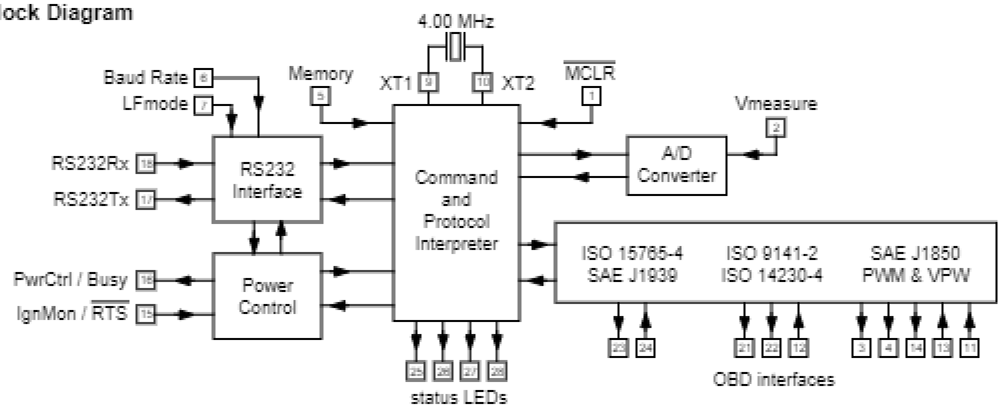
Connection Diagram
PDIP and SOIC
(top view)



Applications

- Diagnostic trouble code readers
- Automotive scan tools
- Teaching aids

Block Diagram



ELM327DSJ | Elm Electronics – Circuits for the Hobbyist | 1 of 94
www.elmelectronics.com

- Micro SD card

Product Description

512GB Micro SD Card Memory Card with Adapter 512GB Ultra microSDXC Memory Card 512GB-XHL1

Excellent performance transfer speed Up to 100MB/s,Full HD video displaying/recording
You are sure to spend more time watching movies,4K & Full HD videos,listening to music and less time transferring them between your devices

Large Storage Capacities
With capacities up to 512GB,enough memory for up to 24 hours of 4K UHD video,78 hours of Full HD video,150,300 photos,or 77,300 songs
So go ahead and savor all of what life has to offer,and keep it all too.and whatever else your creative mind can think of.All your files
large or small in just one amazingly fast Memory Card

Extended Compatibility
Includes full size adapter for use in Laptop,Tablet,PC,Smartphones,Camera,DSLR,Dash Cam,Camcorder,Surveillance
e-Reader,Drone,Gaming.Files,Videos,Music.Compatible with Nintendo Switch GoPro Andoroid Samsung Canon Nikon

Package Included
512GB Micro SD Card
SD Adapter

100% Risk-free Satisfaction Guarantee from the Date of Purchase
12 months manufacturer guarantee and Free technical support

- RF adapter

RF Cables Technical Data Sheet — RG178B/U

Configuration

- Flexible Cable
- 1 Shield(s)

Electrical Specifications

Description	Minimum	Typical	Maximum	Units
Frequency Range	DC		3	GHz
Impedance		50		Ohms
Velocity of Propagation		70		%
Operating Voltage (AC)			1,000	Vrms
Nominal Capacitance		29.4 [96.46]		pF/ft [pF/m]

Performance by Frequency Band

Description	F1	F2	F3	F4	F5	Units
Frequency	0.1	0.4	1	3		GHz
Attenuation, Typ	13.81	27.8	44.41	78.4		dB/100ft
	45.31	91.21	145.7	257.22		dB/100m

Mechanical Specifications

Diameter	0.072 in [1.83 mm]
Weight	0.005 lbs/ft [0.01 Kg/m]
Min. Bend Radius (Repeated)	0.4 in [10.16 mm]

Construction Specifications

Description	Material and Plating	Diameter
Inner Conductor	Copper Clad Steel, Silver, 7 Strands	0.012 in [0.3 mm]
Conductor Type	Stranded	
Dielectric	PTFE	0.034 in [0.86 mm]
First Shield	Silver Plated Copper Braid	0.051 in [1.3 mm]

7.2 Bios and CVs

7.2.1 Bios

Jose C. Garza

I was born in Laredo, Texas on January 8th, 2000. Currently working in my Bachelors of Science in Computer Engineering - Computer Science Track. Throughout my college career I led the logistics of a service organization “BUILD” that builds medical clinics out of shipping containers over the fall semester with the help of the student body. Additionally, I had an internship in the summer of 2021 at Raytheon Technologies in which I had the ability to work on a development of a next generation radar used in military airplanes.

Christian Loth

My name is Christian Loth. I was born in Plano, TX on April 7th, 1999. I am currently a 5th year here at Texas A&M and am majoring in Computer Engineering - Computer Science Track. I have had internships at L3Harris as well as a smaller company called ShadowDragon in the past. When I graduate in May, I will be starting with Cisco as a software engineer in Dallas.

Prisha Srivastava

My name is Prisha Srivastava and I’m from Toronto, Canada. I was born in Mumbai, India on August 10, 2000. I am currently pursuing a Bachelor of Science in Computer Engineering - CS Track and a Minor in Cybersecurity. I am passionate about mentorship and leadership through service. I recently completed a role as Director of Mentorship for a Texas A&M Freshman Leadership Organization called IDEAAL and currently serving as Program Ambassador for Texas A&M’s Courageous Conversations. Additionally, I am very interested in data analytics and software development. This past summer, I worked with a cybersecurity start-up called CyberGRX and will be starting a full time role at Goldman Sachs as a Data Engineering Analyst in Dallas.

Lide Su

My name is Lide Su and I am from Inner Mongolia, China. I was born in Hohhot, China on January 6, 2000. I am currently a senior in Computer Engineering major, and Mathematics minor. I am currently working as an undergraduate researcher in Texas A&M University's Embedded Signal Processing Lab. I am interested in Data Analysis, Software Engineering and Cybersecurity. I am going to extend my education further to pursue a Master's Degree after graduation.

7.2.2 CVs

Jose C. Garza

(956) 893-2576 | **josecgarza@hotmail.com** |
https://www.linkedin.com/in/josecgarza

EDUCATION

Texas A&M University College Station, TX
Bachelor of Science in Computer Engineering with emphasis in Computer Science Aug 2018 – May 2022
Minor in Mathematics & Cybersecurity
Courses: Data Structures, Algorithms, Discrete Mathematics, Logic Circuits, CMOS design

Saint Augustine High School Laredo, TX
Top 5% of the Class |Obtained 5 TEA Performance Acknowledgements Aug 2014 – May 2018

RELATED EXPERIENCE

Internship - Raytheon Technologies McKinney, TX
Systems Engineer June 2021 – Aug 2021
- Provided support in the building of a development station for a software team.
- Collaborated with analysts to model the new weather mode for Release 4 of the APY-10 Radar.
- Analyzed and modified software to the new weather mode specifications.

BUILD Texas A&M University College Station, TX
Team Leader – Leadership Jan 2021 – Present
- Officer position in Materials Procurement.
- Coordinated the logistics of obtaining materials from suppliers and donors to the construction site.
- Led teams of volunteers in construction tasks; including: framing, painting, paneling, among other tasks.
- Developed a database system for construction site sign in and sign out management.

Business Immersion Program for Engineers College Station, TX
Sponsored by Mays Business School and Dell EMC Aug 2020
- Specialized high-impact program for students in engineering disciplines to gain an in-depth understanding of multiple aspects of an entrepreneurial curriculum.
- Learned soft-skills such as self-awareness and emotional intelligence as they are applicable to all professional careers.
- Undertook intense courses in Accounting, Finance, Management, Marketing, and Supply Chain Management.

Aggie Challenge, Disaster City Digital Twin (DCDT) College Station, TX
Software Developer Aug 2019 – Dec 2019
- **Technologies:** React Native, Postman, Expo, Android Studio
- The DCDT system utilizes an integration of Artificial Intelligence software and crowd-sourced data for emergency disaster response planning and operation
- Developed an application for the data gathering of relief organizations in the event of natural disasters.
- Implemented server responses to "CALLS" and "POSTS" from information obtained from the mobile app user.

SKILLS

Programming Languages: C++, Java, Python, Javascript, HTML, CSS
Technologies: Git, Bash, Linux, Matlab, Verilog, Multisim, React Native, Postman, Expo, Microsoft Office Suite, Raspberry Pi, Arduino.
Languages: English, Spanish

AWARDS AND MEMBERSHIPS

- Distinguished Student Spring 2020
- National Society of Professional Engineers (NSPE) Jan 2020 – Present
- Dean's Honor Roll Fall 2019
- Principal's Award May 2018

REFERENCES AVAILABLE UPON REQUEST

Christian Loth

2000 Brenham Dr, McKinney, Texas 75072 | 214-592-6603 | christian@loth.me

OBJECTIVE

Exceptional Computer Engineer seeking a full-time job (starting in the summer of 2022) that will allow me to utilize my deep problem-solving skills, critical thinking, attention to detail, exceptionally fast learning curve, and proven ability to work in teams to advance my skills in the field of software engineering while continually adding business value.

EDUCATION

TEXAS A&M UNIVERSITY – COLLEGE STATION, TEXAS — **MAY 2022**

- Bachelor of science in Computer Engineering (Computer Science track). Minor in Cybersecurity and Math.
- GPA: 3.34
- Fluent in Java, C#, C++, C, MATLAB, Batch, Python, Linux, and Git
- Currently hold a Secret Security Clearance.

WORK EXPERIENCE (www.linkedin.com/in/christianloth3)

L3HARRIS — **SUMMER 2020 – FALL 2020 & SUMMER 2021**

- Worked in the electronic intelligence (ELINT) group and created GUIs to represent many different signal processing graphs such as FFTs, Spectrograms, and IQ plots with real and complex data for analysis. This allowed the ELINT group to easily recognize hardware FPGA bugs and fix them.
- Dealt with custom data structures and multithreaded TCP servers in both C and Python allowing the many subsystems to communicate.
- For time efficiency, the subsystems were built to operate at the low level so the byte order and message byte structures are valid when communicating with each other.
- Worked with protocol buffers and modified a program to output data into an HTML document so that all of the structures could be visually represented and read.

SHADOWDRAGON (www.shadowdragon.io) — **SUMMER 2019 – SUMMER 2020**

- Worked with the ShadowDragon development team to create scripts and processes used by law enforcement to help track and capture individuals performing illegal activities over many different social media platforms.
- Wrote scripts to configure Nagios Core Server which checked and pinged different social media search services to ensure they were working properly.
- Wrote Python scripts to send GET, POST, PUT, and DELETE HTTP requests to automate attributes in the backend.

TEXAS A&M UNIVERSITY (TEACHING ASSISTANT) — **FALL 2018 – SPRING 2019**

- Teaching Assistant (TA) for Engineering 102 – Intro to Engineering
- Assisted professor by working with students in class, teaching them Python, and maintaining office hours to provide students with additional help.
- Maintained information in eCampus class portal where students download and submit assignments, view grades, and see project deadlines.

ICODE (INSTRUCTOR) — **SUMMER 2018**

- Lab mentor and tech lead for iCode coding camp.
- Helped teach younger students the fundamentals of coding and computer logic.
- Explained coding concepts in an easily understandable way for young students to both use and remember.

CHICK-FIL-A (CASHIER, DINING AREA MAINTENANCE) — **SPRING 2017**

- Reinforced brand expectations as a customer greeter.
- Worked as a cashier while maintaining the integrity of the corporation.
- Maintained, cleaned, and organized dining and cashier areas as needed.

ACCOMPLISHMENTS & INTERESTS

- Information Technology Officer and manager of Aggie Men's Alliance organization webpage.
- Chosen by professor to participate in the Texas A&M Engineering Showcase.
- Participated in the Texas A&M Engineering Honors MATLAB competition and performed substantially well.
- Competed in the TAMUHack hackathon on a team of 4 developers.
- FAA licensed pilot with current endorsements.
- Accomplished musician, percussionist, and drummer.

TEXAS A&M ACTIVITIES

- Aggie Men's Alliance IT Officer – A men's leadership and social organization at Texas A&M that is very involved with community service; selected as website developer and manager, developed a new mobile app for the group and used APIs to structure the member payment system.
- Texas A&M Computer Architecture Study Abroad Program – Based in Sydney Australia, this program examined the path from C programming language down to assembly language. It then examined the design of a simple machine that executed a given assembly language, including processor memory systems and storage. The program provided new cultures, Australian college visits, and new geographies to be learned and explored over a 6-week time period.

Prisha Srivastava

(972)-704-9856 | sprisha00@tamu.edu | http://people.tamu.edu/~sprisha00/ | www.linkedin.com/in/sprisha

EDUCATION

Texas A&M University, College of Engineering — **College Station, TX**
BS in Computer Engineering | Minor in Cybersecurity, Mathematics — August 2018 - May 2022
GPA: 3.36

WORK EXPERIENCE

CyberGRX — **Denver, CO (Remote)**
Software Engineering Intern (Full-Time) — June 2021 - August 2021

- **Designed and developed** custom components for improved user interface using typescript, custom storybook components and css libraries
- Collaborated with **10+ team** directly alongside Product to analyze user requirements and develop features
- Resolved issues in select, platform services through **coverage testing** and troubleshooting testing environments
- Familiarity and full compliance with **Agile**, **Scrum XP,** and iterative development approach

Texas A&M Dept. of Computer Science & Engineering — **College Station, TX**
Undergraduate Research Student (Part-time) — March 2021 - Present

- **Lead** full-stack development for **SpectoCyber web-app** and **research** on **NIST NICE Framework** to offer insight on training and experience needed for a career in cybersecurity
- Build scalable infrastructure for two different user pools and integrate customized backend services and UI features.

Texas A&M Secure & Trustworthy Hardware Lab — **College Station, TX**
Undergraduate Research Student — October 2019 - December 2020

- **Led lab testing** of RISC-V SoCs to detect security bugs and investigated vulnerabilities for processor embedding fuzzing techniques
- Developed **python scripts** to detect discrepancies between golden ratio model to **identified privilege access vulnerabilities** and replicated hardware fuzzing attacks on SoCs

SKILLS & RELEVANT COURSES

- Languages: Typescript, Node.js, Cypher, GraphQL, C++, Python, JavaScript, Verilog, HTML/CSS
- Software: Jira, WireShark, Microsoft Visual Studio, MS Office, PyCharm, Vivado, MATLAB
- Platform & Tools: Docker, Kubernetes, SwaggerUI, React, Github, LinuxOS, AWS, Google Cloud Platform
- Courses: Operating Systems, Computer & Network Security, Cybersecurity Risk, Data Analytics for Cybersecurity, Recommended Systems, Computer Architecture, Programming Studio, Data Structures & Algorithms.

PROJECTS

SpectoCyber - NIST Analysis Tool — **In Progress**

- Developing a **React-based** web application using **GraphQL** and **Python ML** Libraries to perform statistical analysis on user-based surveys, and offer empirical curriculum recommendations for TAMU Faculty and employment insights for job seekers.

Snap Planner App — **May 2021**

- A full stack **React web application** to organize calendar events, create automated group event time suggestions and chat feature using **Google Cloud Platform** as a way to authenticate unique users and create cloud-base database system

Subreddit Recommender System — **May 2021**

- A Recommendation system web app created with **Python Django** and **React** suggesting subreddits based on posts/comment history of similar users using hybrid user-item based collaborative filtering ML model

LEADERSHIP EXPERIENCE

Institute for the Education and Development of Asian-American Leaders (IDEAAL) — **College Station, TX**
Director of Mentorship (April 2020- May 2021) | Secretary (19-20) — September 2018 - May 2021

- Conceptualized ten educational workshops over Asian-American identity and socio-political issues **to increase mentee engagement by 55%**
- Implemented organization restructure to virtual transition through online workshops and interactive mentorship environment with **Aggie Mentoring Network**
- Provided mentorship training for **twelve officers** through Summer Virtual Training and weekly 1-1s with outlined agenda to **supervise** leadership development

Leadership Education for APIDA Development Conference (Remote)

April 2021 Individuality Workshop Speaker

- Presented a **45-minute workshop** over empathic leadership through personal development by engaging **50+** attendees

Lide Su

311 Stasney St, College Station, TX 77840 | 979-922-0916 | seward0106@outlook.com

Education

TEXAS A&M UNIVERSITY | COLLEGE STATION, TEXAS | MAY 2022

- Major: Bachelor of Science in Computer Engineering (Computer Science Track)
- Minor: Mathematics
- GPA: 3.57

Employment

TEXAS A&M UNIVERSITY | TEACHING ASSISTANT | AUGUST 2020 – DECEMBER 2020

- Teaching Assistant (TA) for ENGR/PHYS 216: Experimental Physics and Engineering Lab II – Mechanics, and ENGR/PHYS 217: Experimental Physics and Engineering Lab III - Electricity and Magnetism.

TEXAS A&M UNIVERSITY | RESEARCH ASSISTANT | JUNE 2021 – PRESENT

- Constructed the research study coordinator page along with the authorization functionality, showing the registration verification states and the usage heatmap of over 150 days from 15 users' data.
- Constructed and maintained the database of the research program related to wearable devices such as Oura Ring and Garmin Watch, made sure that the database with over 500 users' data is secure and ready.
- Designed the algorithms and a machine learning model to baseline and predict fever based on the biological data of study participants.

Projects

AGGIEPASS - A HARDWARE PASSWORD MANAGER https://github.tamu.edu/minh-luu/AggiePass

- A password manager that secures user passwords and usernames using an RFID tag.

GALAXY BUCKS - AN ORDER-MENU APPLICATION

- This is an application which integrates menu, order, user information and management functionalities. The project is developed based on postgreSQL database, and the UI is developed with Java swing.

Skills & Relative Courses

PROGRAMMING LANGUAGES

- C++, Java, Python, HTML/CSS, JavaScript, PostgreSQL, Verilog, Node.js

SOFTWARE

- WireShark, Microsoft Visual Studio, Microsoft Office, PyCharm, MATLAB, Vivado

PLATFORMS & TOOLS

- GitHub, LinuxOS, AWS, Google Cloud Platform, Bootstrap, Pandas, Bokeh

RELATIVE COURSES

- Operating System, Computer Architecture, Programming Studio, Data Structure & Algorithms, Machine Learning, Network & Cybersecurity, Cryptography

LANGUAGES

- English, Mongolian and Chinese