Senior Design 1

MDAS.ai Drive-By-Wire Using V2X for Enhanced Safety

System Requirements Specification

Version 2 / 8 November 2018

Fall 2018

Team #7 - Classified:
Athanasios Argyris, Kayleigh James, Sarah Overbeck, Joshua Quejadas

Contact Information: soverbec@umich.edu, aargyris@umich.edu, jamesmk@umich.edu
Advisor: Dr. Mike Putty

Table of Contents

Section	Title	Page
1.	Introduction	2
1.1	Purpose	2
1.2	Definitions	3
1.3	System Description	3
1.4	References	6
2.0	Constraints	7
2.1	Environmental Constraints	7
2.2	MDAS.ai Vehicle Constraints	7
2.3	Reliability and Safety	7
3.0	System Requirements	8
4.0	Interface Description	11
5.0	Risk Assessment and Mitigation	13
5.1	Functional and Performance Risks	13
5.2	Programmatic Risks	13
5.3	Safety and Reliability	13
5.4	Knowledge Base and Uncertainty	13
5.5	Risk Mitigation	14
6.0	Budget Information	20
7.0	Master Schedule	21

1. INTRODUCTION

1.1 Purpose

This document define the system requirements for the MDAS.ai Drive-by-Wire system using V2X for Enhanced Safety. This document will expand upon the Concept Description by providing quantitative constraints and requirements for the system.

For this project, we will be designing a system that will convert a traditionally operated vehicle into a Drive-By-Wire (DbW) vehicle. We will be taking three main functions usually operated by a driver and computerizing the control of each as well as the communication between them. In addition, we will be incorporating Dedicated Short-Range Communication (DSRC) as a form of Vehicle-to-Everything (V2X) communication for further safety of the system as a whole. This V2X communication will notify the vehicle of an external event, such as, a pedestrian in a crosswalk. Then, the vehicle will come to a safe stop until the crosswalk has been cleared, or the pedestrian in the crosswalk message has stop being sent by the DSRC radio. The ***motivations for this project*** is to create a reliable and safe DbW system and enhancing the aforementioned safety by incorporating V2X communication. An ***overall benefit*** is learning how both these technologies work and to show that autonomous vehicles could be enhanced with this outside information to make better decisions in various environments and conditions. This vehicle will be a ***working prototype*** that the MDAS.ai group can take and turn into an autonomous vehicle that will be driven around the University of Michigan - Dearborn campus.

1.2 Definitions

MDAS.ai: Michigan Dearborn Autonomous Shuttle
DSRC: Dedicated Short Range Communication
V2X: Vehicle to X (Infrastructure, Vehicle, ect)
CAN: Controller Area Network
DbW: Drive-by-Wire
Drive PX2 : The Nvidia Drive PX2 is a computer that is aiming to provide autonomous functionality to MDAS.ai using machine learning.
RSU: Road Side Unit; DSRC module placed outside of the vehicle used to alert the vehicle of a road hazard.
OBU: On Board Unit; DSRC module placed in the vehicle used to receive alerts from the RSU.
RSSI: Received Signal Strength Indicator
GPIO: General Purpose Input Output
E-STOP: A switch, that when pressed, will notify the boards via an interrupt on a GPIO pin. This switch will be an analog device, either providing 0V or 3.3V into the microcontroller.
DAC: Digital-to-Analog Converter; a piece of hardware that converts a digital signal to an analog signal.

1. 3 System Description

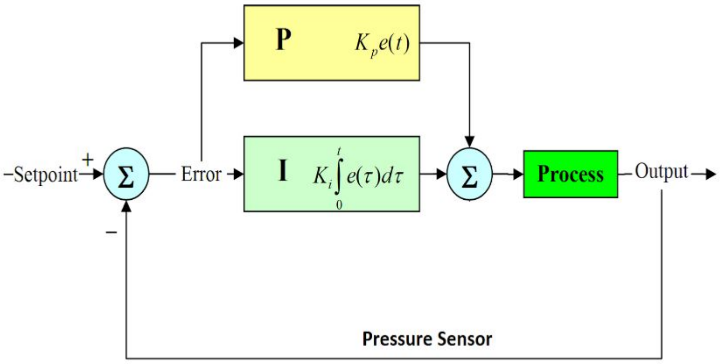
MDAS.ai (Michigan Dearborn Autonomous Shuttle) is a research project dedicating to the design, construction, test, and implementation of an autonomous shuttle vehicle. Before that can be accomplished however, there are several smaller projects that must be completed. For this project, we will be designing a system that will convert a traditionally operated vehicle into a Drive-By-Wire (DbW) vehicle. We will be taking three main functions usually operated by a driver and computerizing the control of each as well as the communication between them. In addition, we will be incorporating Dedicated Short-Range Communication (DSRC) as a form of Vehicle-to-Everything (V2X) communication for further safety of the system as a whole. The primary objectives of this system are to be a reliable and safe DbW system and enhancing the aforementioned safety by incorporating V2X communication. The major goal of using this V2X communication is to provide the vehicle with information about the outside world that could alter the state of the vehicle's operation.

The DbW system will include the following modules: RF, steering, braking and throttle, and joystick. As mentioned before, all modules will be communicating via a Controlled Area Network (CAN).

Presently, MDAS.ai only has pieces of two of the modules which have been implemented in an Ad-Hoc fashion: the steering module and the joystick module. These two are currently interfaced over CAN, but have no defined message set. This is because they primarily work together for demonstration purposes. Previously, the vehicle utilized a rudimentary throttle module, however, it was never interfaced over CAN and was activated by flipping a toggle switch. The brake and DSRC module have yet to be developed and will be fully developed by our team. The individual modules that are partially developed have been tested, but never integrated to communicate as an operational system with a defined message set. Our goal is to integrate all of these modules, use CAN as our main communication protocol, build a message set and introduce V2X using DSRC to provide the vehicle with vital information that could alter the way the vehicle will operate.

The RF module will be composed of two parts, the On-Board Unit (OBU) located on the vehicle and the Roadside Unit (RSU) located on a crosswalk, intersection, or other infrastructure. In the event of a traffic incident (e.g. reduced speed ahead or a crash) the RSU will send a message to the OBU in order to alert the vehicle of the incident ahead. The OBU will interpret the RSU's messages in order to generate a CAN message. This CAN message will be sent to the modules that are affected by the incident. For example, if a pedestrian is inside the crosswalk, The RSU should send a message to the OBU indicating this event. The CAN message sent by the OBU should trigger a change in all modules in order to properly respond to the pedestrian. The brake module will bring the vehicle to a stop and the other modules will enter a safety state. Once the OBU stops receiving this message, the safety state will be lifted and the vehicle will resume normal operation.

The next module in our system is the joystick module. The joystick will be capable of controlling the vehicle meaning it will have control over steering, brakes, and throttle modules. We will be utilizing CAN to communicate between the joystick module and all other modules. This device will have a dead man switch in order to ensure safe operation of the vehicle. This module will serve



two additional purposes in our system: the primary purpose being a diagnostic tool and our secondary purpose being system verification. Due to the safety concerns that come with working on a vehicle, the joystick will be our primary method for demonstrating the vehicle operating as intended.

The brake module in our system will be controlled via a PID controller. This means it will be receiving a demanded "Setpoint" pressure via CAN. It will be receiving a feedback "Output" pressure via the brake pressure sensor, and apply the correct amount to achieve the setpoint. It will apply the pressure by generating a CAN message and sending it to the brake actuator. For this module, we will have to define a message set for both receiving and sending CAN data.

The steering module will receive a CAN message and do meaningful scaling. After, it will use this scaling and generate a CAN message that will be sent to the steering actuator. It will also be actively listening for a message from the input torque sensor. If there is an input torque applied during DbW mode, it will safely deactivate DbW and notify the other modules. The other modules will be notified via CAN and enter a safety mode that will not be deactivated until the vehicles power is cycled. A well-defined message set will be in order, to ensure proper and safe communication for the DbW system.

Another example of our intended functionality is when the DbW modules receive a signal indicating an emergency. The manual emergency stop button, if pressed, will trigger the brake board to stop the vehicle in a quick but safe manner. Simultaneously, the steering and throttle boards will go into safety mode such that they do not accept messages from any other sources. To ensure that the E-stop is not easily disengaged, the button will have to be released and the vehicle power- cycled before DbW can be re-engaged.

As part of our message set and safety features of our DbW system, we will implement a "Keep Alive" check. Meaning, if our modules don't see a message within a set amount of time, they will enter a safety mode and disable DbW mode until communication is re-established. On the system requirements documents, the details and specific numbers will be presented.

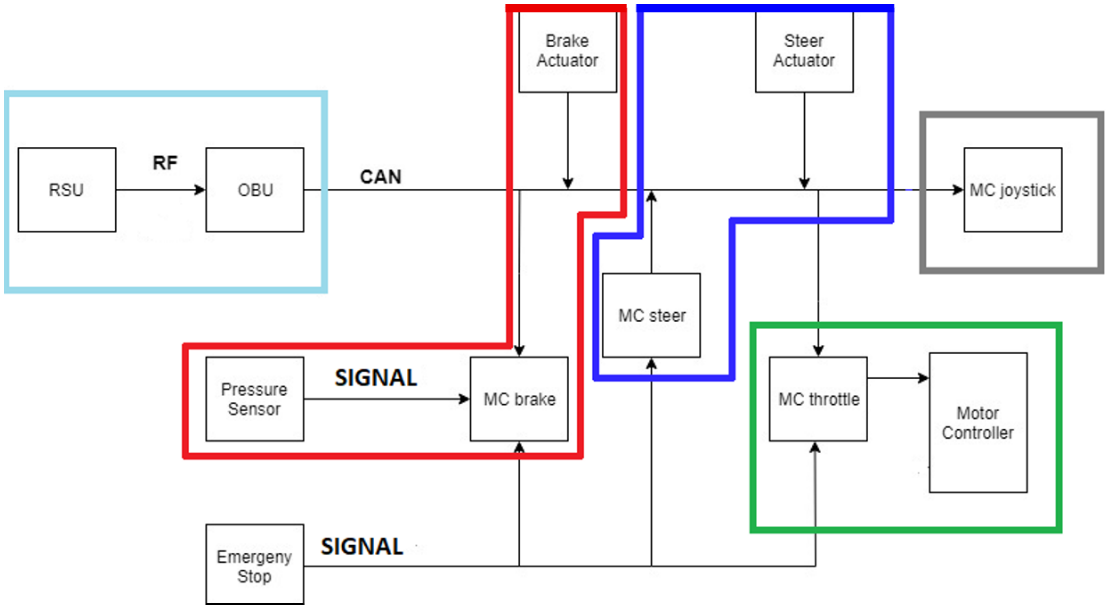


Figure 1: MDAS.ai Drive-by-Wire System Block Diagram
DSRC module (light blue), Brake Module (red), Steering Module (dark blue), Throttle Module (green), and Joystick Module (gray)

1.4. References:

1. Bertoluzzo, M., et al. "Drive-by-Wire Systems for Ground Vehicles." *2004 IEEE International Symposium on Industrial Electronics*, 2004, doi:10.1109/isie.2004.1571893.

2. Kenney, John B. "Dedicated Short-Range Communications (DSRC) Standards in the United States." *IEEE*, Proceedings of the IEEE, July 2011, ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=5888501.

3. TIVA C Series Microcontroller Data Sheet: http://www.ti.com/lit/ds/symlink/tm4c123gh6pm.pdf

4. CAN ISO Standard 11898: https://www.iso.org/obp/ui/#iso:std:iso:11898:-1:ed-2:v1:en

5. Technical Specification for DSRC: https://www.imda.gov.sg/-/media/imda/files/regulation-licensing-and-consultations/ict-standards/telecommunication-standards/radio-comms/imda-ts-dsrc.pdf?la=en

6. IEEE 1609 Spec for DSRC: https://standards.ieee.org/standard/1609_2-2016.html

7. I^{2}C Standard: https://www.nxp.com/docs/en/user-guide/UM10204.pdf

2. CONSTRAINTS

2.1 Environmental Constraints

The system will be developed in a laboratory environment during this project. It will not be exposed to significant temperature changes, vibration, or impact.

2.2 MDAS.ai Vehicle Constraints

The system must be able to run off of the MDAS.ai vehicle's power supply while the vehicle is operating. The system's size and weight shall not degrade the operational characteristics of the shuttle. The system (excluding the RSU and the antennas for the OBU) must be able to fit within the shuttle.

2.3 Reliability and Safety Constraints

The system must maintain safe operation in the event of component failure or communication failure. The software must maintain reliability in the event of software errors.

2.4 MDAS.ai Development Site Constraint

The system will be developed in a laboratory environment (IAVS high bay) as a test bed and the vehicle will be up on blocks so that movement from test area is prevented. This project is also not easily mobile, preventing ability to work in other labs.

2.5 MDAS.ai Other Project Teams Considerations Constraint

The vehicle is part of a large research project with several other teams working on it at once. It must be considered that at times, components may or may not be worked on while other teams are completing a demonstration or working on the vehicle at the same time. This will constrain time while completing the project on the vehicle.

3. SYSTEM REQUIREMENTS

3.1. DSRC

The system will use DSRC V2X communication to alter the system's state in order to enhance the safety of the DbW system by relaying information about the surroundings. The OBU radio will receive a message from the RSU radio, and generate a CAN message to update the other modules on the vehicle state. The message that will be transmitted via the RSU radio is 'pedestrian in crosswalk,' and it will notify the steering, brake, throttle modules. Once the modules receive this CAN message, it will bring the vehicle to a safe stop and disengage the Drive-by-Wire mode until the RSU stops broadcasting that a pedestrian is in crosswalk. After this occurs, the vehicle will re-engage the Drive-by-Wire mode and continue normal operation.

- 3.1.1 range estimate: < 100 feet
- 3.1.2 noise floor estimate: -80 dBm to -90 dBm
- 3.1.3 frequency band of operation: 5.9 Ghz
- 3.1.4 latency estimate: < 5 ms
- 3.1.5 message estimate: 1 DSRC sent/received and 1 CAN message generated
- 3.1.6 Activate Pedestrian crosswalk message via laptop through socket

3.2. Throttle Module

The system have a throttle module in order to control the state of the motor controller. It will read the throttle pedal analog sensor and listen a for CAN message for control information. We will also publish the TPS sensor reading onto the CAN bus such that the neural network in the MDAS vehicle can be trained. This is a requirement from the neural team that is working on the MDAS vehicle. The message will be parsed and control two DAC's that will be inputted into the motor controller. If the E-STOP is engaged, the module will not provide any analog signals to the motor controller. After the E-STOP is disengaged and the vehicle's power is cycled, the modules will return to their normal operating state. This module will have a feature, such that when it misses 3 consecutive CAN messages, the DbW system will disengage. DbW will not re-engage until the vehicle's power has been cycled. Additionally, when a message from the RSU is received stating that there is a pedestrian in the crosswalk, the throttle module will stop providing an input voltage signal to the motor controller.

- 3.2.1 DAC1 output voltage: 1V to 4V, DAC2 output voltage: .5V to 2V (datasheet specified the acceptable range of operation)
- 3.2.2 DAC2 will be ½(DAC1) at all times. (As specified in the motor controller datasheet)
- 3.2.3 E-STOP input voltage estimate: 0V to 3.3V
- 3.2.4 If 3 consecutive CAN messages are not received, regardless of their source, the DbW system will cutoff and go into a safe mode, similar to that of the emergency stop.
- 3.2.5 DAC's will control the vehicles motor controller

- 3.2.6 The throttle module will stop operation in a safe manner in the event of an emergency stop button being pressed.
- 3.2.7 The module will take two analog input signals from the throttle pedal sensor (TPS).
- 3.2.8 Send TPS sensor reading on the CAN bus

3.3. Steering Module

The system will have a steering module to provide steering actuation for a safe DbW system. The steering module will respond to CAN communication from the OBU and the joystick module. When the steering module responds to notifications from the OBU or the joystick module, meaningful scaling will be performed and a CAN message will be sent to the steering actuator. The steering module will also be actively listening for a message from the input torque sensor. If there is an input torque applied to the steering wheel by the safety driver during DbW mode, it will safely deactivate DbW and notify the other modules. The other modules will be notified via CAN and enter a safety mode that will not be deactivated until the vehicle's power is cycled. This module will have a feature, such that when it misses 3 consecutive CAN messages, the DbW system will disengage. DbW will not re-engage until the vehicle's power has been cycled.

- 3.3.1 Center to max turn on steering wheel time should be within: 4 seconds
- 3.3.2 If 3 consecutive CAN messages are not received, regardless of their source, the DbW system will cutoff and go into a safe mode, similar to that of the emergency stop.
- 3.3.3 The steering module will stop operation in a safe manner in the event of an emergency stop button being pressed.
- 3.3.4 If the Input torque exceeds ±7Nm, the system will disengage DbW mode.
- 3.3.5 Steering module will control the steering actuator via CAN with control messages at 10 Hz.

3.4. Brakes Module

The fourth module of the system will control braking of the vehicle. This module will receive and respond to CAN communication from the OBU and the joystick simulation. It will receive a message over CAN requesting a set position (i.e. brake pressure) and the PID loop will handle brake actuation in order to achieve the set position. If a DSRC message is received by the OBU via the RSU, the message will generate a CAN message to apply braking in order to bring the vehicle to a safe stop and disengage DbW mode. Similar action will be taken with the E-STOP; braking the vehicle in a safe manner. In the event the E-STOP is pressed, the brake module will also send a CAN message to the other modules notifying them that DbW mode has been disengaged. This safe mode will not be disengaged until the E-STOP is not longer being depressed and the power on the vehicle is cycled. If 3 consecutive signals are not received from any source, the DbW system will cut off and go into safe mode.

- 3.4.1 Brake pressure voltage: No Pressure=0.5V, Max Pressure = 1.3V
- 3.4.2 If 3 consecutive CAN messages are not received, regardless of their source, the DbW system will cutoff and go into a safe mode, similar to that of the emergency stop.
- 3.4.3 The brake module will bring the vehicle to a stop in a safe manner in the event of an emergency stop button being pressed.
- 3.4.4 Estimated start of actuation to full actuation time: 2 to 3 seconds
- 3.4.5 The PID loop response time estimate: 100 ms <= response <= 300ms
- 3.4.6 The PID loop will control brake pressure as measured by the sensor to: ± 0.1V
- 3.4.7 Brake module will control brake actuator via CAN with control messages at 10 Hz.
- 3.4.8 Put the brake pressure on the CAN bus at a rate of 10 Hz
- 3.4.9 Apply full brake pressure in the event of a DSRC message being received (i.e. wheels should stop spinning completely) within 2 seconds

3.5. Joystick Module

The last module of the system will be the joystick module. This will be used as a way to simulate autonomy in the vehicle in addition being a debugging tool. It will also be used to verify the operational integrity of the DbW system.   This control will also take priority over PX2 for safety and testing purposes.

- 3.5.1 The joystick module will send control message to the brake, throttle, and steering modules individually. It will send 3 separate, consecutive CAN messages every 10 Hz
- 3.5.2 The Joystick Module will have a dead man switch. If this button is not being pressed it will not send CAN messages to the other modules

3.6. Power Requirements

The 4 modules will be using TIVA TM4C123GXL Microcontroller and the maximum current the board can supply is 323 mA. We know the TIVA must also draw current to run itself so we assume the effective max draw to be around 400 mA. We have four TIVA boards in the system leading us to the following calculation: Power estimate: (0.400A)*5V * 4 = 8W. Each board will be powered via a 5V DC-DC Step down source. The battery array in the MDAS vehicle provides 48V DC and there will be a 12V step down, followed by another 5V step down to the TIVA boards.

- 3.6.1 Power consumption <= 8W
- 3.6.2 The system will be powered via the vehicles 48V battery array. It will power the microcontrollers via DC-DC step down to 5V.

3.7. Timing Budget

The modules will send a new can message every 100ms, making our update rate 10 Hz. This number was derived from the fact that we don't want to drive up utilization of the CAN bus. The CAN bus on the vehicle is 250k Baud from the manufacture. This gives us an effective bit rate of 1/250k = 4 µsec/bit . The vehicle's CAN bus we measured a utilization without our modules of 30% utilization. .3(4 µsec/bit) = 1.4 µsec/bit. We want to keep the utilization of the CAN bus lower than 40%, thus we cannot exceed .4(4 µsec/bit) - 1.4 µsec/bit. This gives us .2 µsec/bit to work with for our design. The size in bits for a CAN message is 129 bits for extended CAN. This means, for every update we will put 129 bits on the bus. The joystick module will be sending 3 messages every 10 Hz, and the other 3 boards will send 1 message every 10 Hz. At 129 bits *(3+3) * 10 Hz *1/10^9 = 7.74 * 10^-6 µsec/bit. This falls well within our .2 µsec/bit. criteria. At an update rate of 10 Hz, it is fast enough to control the vehicle as planned because the response time of the mechanical components. Also, the vehicle's top speed is 15 mph and at 10Hz our system will respond accordingly. Therefore, we have mitigated the risk of driving the bus utilization too high which would cause too many bit errors and the 100 ms is practical because that is faster than the reaction time of an average person.

- 3.7.1 CAN Baud Rate = 250k Baud
- 3.7.2 CAN Update rate from each module = 10 Hz
- 3.7.3 Module response time estimate: <= 10ms

4. INTERFACE DESCRIPTION

The majority of this project will be focused on creating a software interface between steering, brake, throttle, and joystick modules. This will come in the form of a CAN message set that we will create. CAN will be the communication protocol between the Onboard Unit and the brake actuator, steering actuator, brake module, steering module, throttle module, and joystick. We will be utilizing the CAN ISO Standard 11898 to construct our message set. There will also have to be an interface between the Roadside Unit and Onboard Unit in the form of DSRC in the 5.9 GHz band. For this communication standard we will be utilizing the IEEE 1609 Spec for DSRC. There will also be I^2C communication between the DACs and throttle module. We will be using the I^2C Standard from NXP. All of these documents have links associated with them in the References section of this report. Finally, the communication between the Emergency Stop and the brake, steering, and throttle modules will be a digital signal.

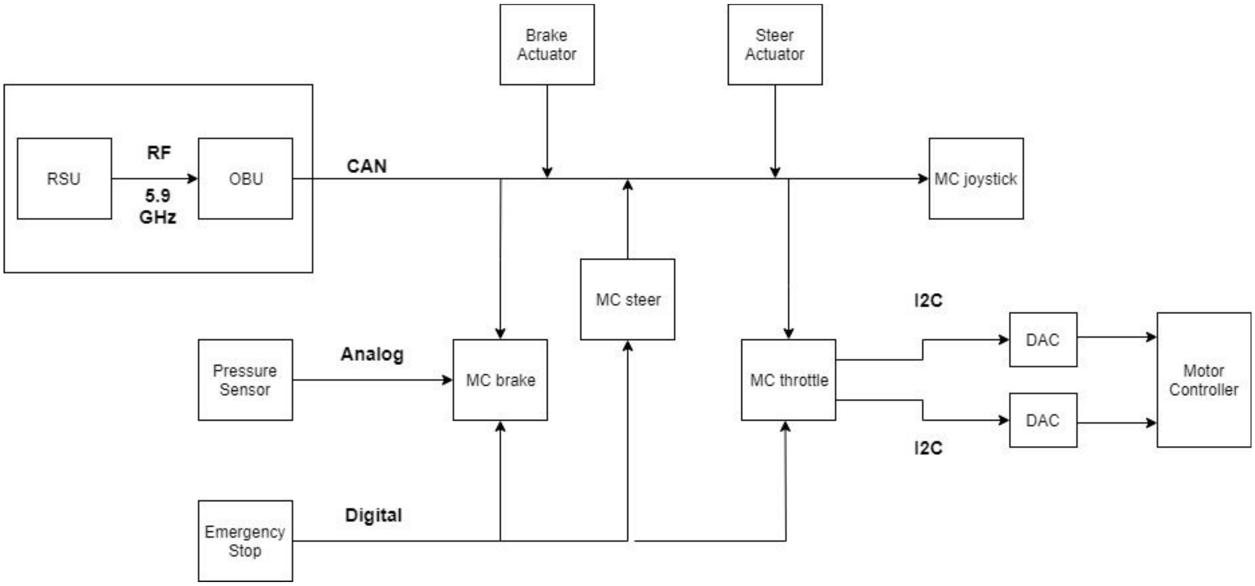


Figure 2: MDAS.ai Drive-by-Wire System Block Diagram With Interfaces Between Components.

5. RISK ASSESSMENT AND RISK MITIGATION

5.1 Functional and Performance Risks

A. Improper Communication Protocols (6)
B. Noisy Environment (8)

5.2 Programmatic Risks

A. Poor planning and under/overestimating tasks (3)
B. Loss of data (computer/document files) (4)
C. Starting late/falling behind because of other homework assignments (1)
D. Becoming disengaged, ignoring project communications, misunderstanding requirements, under communication, inaccurate expectations, individuals aren't kept informed (2)

5.3 Safety and Reliability

A. Faulty Wiring/Wire Harness (5)
B. Lack of signal from RSU (low RSSI) (7)

5.4 Knowledge Base and Uncertainty

A. Lack of Subject Matter Knowledge (9) - All team members lack some of the information necessary to complete this project. This will mostly be alleviated by excursions.

Risk Assessment Table:

red = alert yellow = warning green = caution	**Penalty of Occurrence**	10	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
		9	🟨 8	🟥	🟥	🟥	🟥 5	🟥	🟥	🟥	🟥	🟥
		8	🟨 4	🟨	🟨	🟥	🟥	🟥	🟥	🟥	🟥	🟥
		7	🟨 6	🟨	🟨	🟨	🟥	🟥	🟥	🟥	🟥	🟥
		6	🟨	🟨	🟨	🟨	🟨	🟥	🟥	🟥	🟥	🟥
		5	🟩 7	🟨	🟨	🟨	🟨	🟨	🟥 9	🟥	🟥	🟥
		4	🟩	🟩	🟨	🟨	🟨	🟨	🟨	🟥	🟥	🟥
		3	🟩	🟩	🟩	🟨	🟨	🟨	🟨	🟨	🟥	🟥
		2	🟩	🟩	🟩	🟩	🟨 2,3	🟨	🟨	🟨 1	🟨	🟥
		1	🟩	🟩	🟩	🟩	🟩	🟩	🟨	🟨	🟨	🟨
			1	2	3	4	5	6	7	8	9	10
			Risk of Occurrence									
			very unlikely					very likely				

Table: Risk Assessment (2 November 2018)

5.5 Risk Mitigation Steps for any Warning or Alert Level Risks:

Risk Mitigation Steps

- 1,3: Creating a project plan to decrease likelihood of falling behind or missing deliverables
- 2: Miscommunication can be mitigated by creating a dependable communication channel for group meetings and assignment of tasks.
- 7,8,9: Excursions and component-testing before implementation and connection with other components
- 3,6: Detailed documentation
- 4: Redundant storage of all documents and project files in Google Drive Cloud
- 5: Consistent wiring color index to identify type of signal

Excursions

Excursion 1: Exploration of the Current Drive Systems in the MDAS.ai Vehicle

(1) Description of Risk Area:

The MDAS.ai vehicle was, originally, a traditionally operated car which is being converted into an autonomous shuttle. In order to translate the existing system to Drive-by-Wire, the team must have an understanding of what systems and components present in the vehicle. This excursion mitigates the risk of the

1. Lack of familiarity with electric vehicle systems
2. Lack of knowledge of what functions Drive-by-Wire controls

Current Risk Assessment

- Risk 9: Lack of Subject Matter Knowledge
  - Risk of Occurrence: 7
  - Penalty of Occurrence: 5

(2) Objective of Excursion

The objective of this excursion is to perform a set of small experiments on elements of the vehicle in order to reduce the risk of potentially disastrous mistakes while adapting traditional function to DbW. Lack of experience with vehicle systems will be mitigated by exploring all relevant systems in their entirety.

(3) Relevant Requirements

- In-depth mapping of the current drive system in the MDAS.ai vehicle
  - Mapping the 48 Volt system to the 12 Volt system
  - Signals that controls the vehicles motor controller & steering actuator

(4) Excursion Definition:

Description of Steps for Excursion

1. Draw a system block diagram on a whiteboard with the use of the vehicle data sheets.
2. Understand each of the blocks that are relevant to this project.

Possible Outcomes

1. Familiarize team with all vehicle motion systems and see what functionality is already in place.
2. Decide the scope of the project or the system description needs to be altered.

Excursion 2: Integrating a Digital-Analog Converter (DAC) with the Microcontroller

(1) Description of Risk Area:

This project involves conversion of a traditional vehicle to a Drive-by-Wire (DbW) system. One part of this project involves controlling the throttle of the vehicle. One possible approach to control this function is to interface a DAC with a microcontroller which will operate the motor controller already in the vehicle. This excursion mitigates, in part, multiple risks within our project:

1. Faulty wiring and wire harnesses
2. Improper Communication between the microcontroller and a DAC
3. Lack of knowledge of DACs

Current Risk Assessment

- Risk 5: Faulty Wiring/Wire Harness
  - Risk of Occurrence: 5
  - Penalty of Occurrence: 9
- Risk 6: Improper Communication Protocols
  - Risk of Occurrence: 1
  - Penalty of Occurrence: 7
- Risk 9: Lack of Subject Matter Knowledge
  - Risk of Occurrence: 7
  - Penalty of Occurrence: 5

(2) Objective of Excursion

Perform an experiment which integrates the DAC with the microcontroller to reduce the above risks to ensure that the system performance requirements can be met. Additionally, this experiment will provide the opportunity for the team to learn about Digital-to-Analog converters.

(3) Relevant Requirements

- Control the vehicle speed from 0-100%
- Appropriate analog voltages output given digital input.
- Correct wire connections between the DAC and the microcontroller
- Investigate I2C as a communication protocol for the DAC
- Verify voltage output is corresponding with the 12 bit digital value (0 to 4095 in decimal)

(4) Excursion Definition:

Description of Steps for Excursion

1. Obtain microcontroller, DAC, and connectors
2. Integrate DAC to microcontroller using I2C
3. Provide digital values to the DAC and verify the output voltages

Possible Outcomes

1. Decide that this approach is a viable way to meet our system requirements.
2. Decide that using a DAC is not a viable way to meet our system requirements.

Excursion 3: Investigate CAN to create a proper message set

(1) Description of Risk Area:

This project involves modules communicating with one another via CAN (Controller Area Network). In order to build a safe Drive-by-Wire system for the MDAS.ai vehicle, the team will have to develop a CAN message set. This requires an in-depth understanding of the CAN protocol and how to use it to build our message set. This excursion mitigates the risk of:

1. Creating an improper message set
2. Using correct communication protocols for CAN
3. Lack of subject matter knowledge involving CAN

Current Risk Assessment

- Risk 6: Improper Communication Protocols
  - Risk of Occurrence: 1
  - Penalty of Occurrence: 7
- Risk 9: Lack of Subject Matter Knowledge
  - Risk of Occurrence: 7
  - Penalty of Occurrence: 5

(2) Objective of Excursion

Perform research and learn about CAN protocol in order to reduce the above risks to ensure that the system performance requirements can be met.

(3) Relevant Requirements

- Learn how to assign the CAN ID such that the priority is as intended
- Verify that we have 8 Bytes of data per message and 29 bit addressing

(4) Excursion Definition:

Description of Steps for Excursion

1. Research what CAN is from a high level of abstraction.
2. Read more in depth papers and resources in order to grasp the underlying details of how CAN works.
3. Explore code examples of CAN protocol.

Possible Outcomes

1. Learn enough about CAN through our research and exploration in order to perform a future excursion where we work with CAN ourselves.
2. Decide we need to seek help from our faculty advisor so that we can learn enough to perform a future excursion where we work with CAN ourselves.

Excursion 4: Use the DSRC radio to generate a CAN message

(1) Description of Risk Area:

A major goal of our project is to integrate V2X communication between the vehicle and infrastructure. We are going to use Dedicated Short Range Communication (DSRC) as our V2X protocol. The idea is to receive a message on an onboard DSRC radio and generate a CAN message that will notify the DbW system of an external event. This excursion mitigates the risk of the

1. Lack of familiarity with DSRC radios
2. Lack of knowledge with DSRC radio & CAN
3. Lack of Understanding with communication protocols for V2X

Current Risk Assessment

- Risk 6: Improper communication protocols
  - Risk of Occurrence: 1
  - Penalty of Occurrence: 7
- Risk 9: Lack of Subject Matter Knowledge
  - Risk of Occurrence: 7
  - Penalty of Occurrence: 5

(2) Objective of Excursion

The objective of this excursion is to investigate the ability of the DSRC radio to generate a CAN message to provide the vehicle with external data.

(3) Relevant Requirements

- Integrate SocketCAN into the DSRC radio
- Generate a CAN message and verify that it is being transmitted properly
- Investigate the J2735 Standard for DSRC radio communication

(4) Excursion Definition:

Description of Steps for Excursion

1. Setup and connect the DSRC radio.
2. Do research on SocketCAN and write a basic application to transmit CAN messages.
3. Verify it is sending CAN message by using a CAN-to-USB device.
4. Pick out a message from the J2735 standard and modify the application to only transmit a CAN message for a particular DSRC message.
5. Verify it is sending CAN message by using a CAN-to-USB device.

Possible Outcomes

1. Having a DSRC radio that can generate CAN messages.
2. Being able to parse DSRC messages and generating unique CAN messages.
3. CAN may not work on the DSRC radio as it has never been tested.

Excursion 5: GitHub as a Collaboration and Repository Tool

(1) Description of Risk Area:

The source code will be modified many times at different sections by multiple people. This could lead to unintended overlaps and/or overwrites. There is also the possibility of losing or damaging personal computers thus preventing access to the modified code. The proposed approach is to learn to use Git. GitHub will be used to host the source code. This mitigates the risk of losing data.

Current Risk Assessment

- Risk 4: Loss of Data/Project Files
  - Risk of Occurrence: 1
  - Penalty of Occurrence: 8

(2) Objective of Excursion

Learn to use Git commands to maintain a centralized source code in order to improve collaboration and reduce the risk of loss of project files.

(3) Relevant Requirements

- Organized code with comments and updates for each revision

(4) Excursion Definition:

Description of Steps for Excursion

1. To create a local repositories
2. Branch the master code
3. Learn to push and pull source code from GitHub.

Possible Outcomes

1. A centralized source code in GitHub.
2. A centralized source code in the personal computer.

Excursion 6: Closing a PID loop for brake system

(1) Description of Risk Area:

The brake system in the MDAS.ai vehicle does not have a proper control system in place to govern the braking of the vehicle by wire. One approach to alleviate this issue is to use a PID loop to provide a regulated amount of actuation to the brake pressure sensor (BPS) output as compared to the demanded BPS value. This excursion mitigates the risk of:

1. Lack of subject matter knowledge in PID controllers
2. Lack of experience with brake pressure sensors and brake actuators

Current Risk Assessment

- Risk 9: Lack of Subject Matter Knowledge
    - Risk of Occurrence: 7
    - Penalty of Occurrence: 5

(2) Objective of Excursion

Our primary objective for this excursion is to learn how to close a PID loop. Lack of experience in PID controllers makes separate experimentation with them essential to mitigate risk. This will also improve the team's knowledge of vehicle braking systems.

(3) Relevant Requirements

- Learning to close a PID loop & Closing the loop
- Apply 0 to 100% brake given the input to the loop

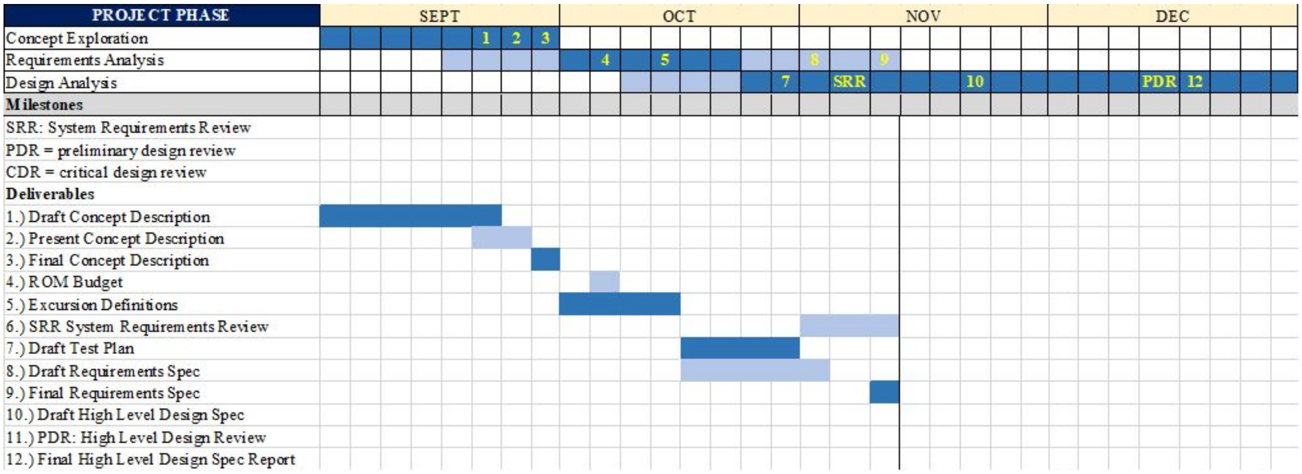
6. BUDGET INFORMATION

Cost Estimate (Version 1, Requirements Specification, 8 November 2018)

Module	**#NO**	**Item**	**Vendor**	**Model/Part No.**	**Unit Cost**	**Qty**	**Sub Total Cost**
RF Module	1	DSRC Radio	Cohda	MK5 OBU/RSU	$2000.00	2	$4000.00
Microcontroller	1-5	Tiva Launchpad	TI	TM4C123G XL	$13.49	5	$67.45
Digital-to-Analog	1-2	DAC	Microchip	MCP4725	$2.49	2	$4.99
Steering Actuator	1	Steering Actuator	Global Motors	GMA12	$1499.99	1	$1499.99
Linear Actuator	1	Linear Actuator	Kartec	1A0011BJ	$1499.99	1	$1499.99
Total Cost Estimates							**$7072.00***

* All components provided by MDAS.ai and DSRC budget

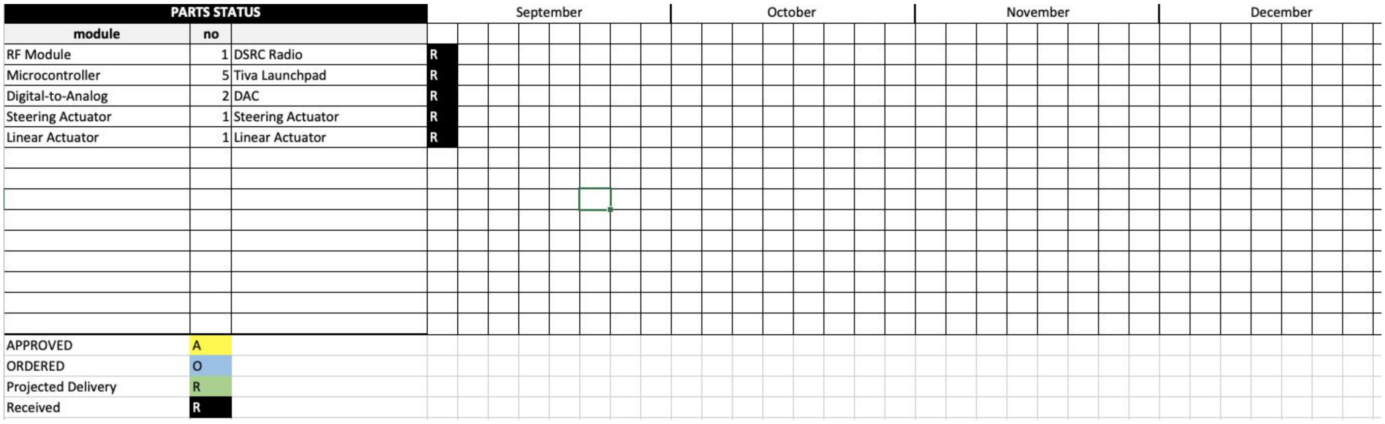
7. MASTER SCHEDULE (Requirements Specification) - 2 November 2018



PROJECT PHASE	SEPT	OCT	NOV	DEC
Concept Exploration	1 2 3			
Requirements Analysis		4 5	8 9	
Design Analysis		7	SRR 10	PDR 12
Milestones				
SRR: System Requirements Review				
PDR = preliminary design review				
CDR = critical design review				
Deliverables				
1.) Draft Concept Description				
2.) Present Concept Description				
3.) Final Concept Description				
4.) ROM Budget				
5.) Excursion Definitions				
6.) SRR System Requirements Review				
7.) Draft Test Plan				
8.) Draft Requirements Spec				
9.) Final Requirements Spec				
10.) Draft High Level Design Spec				
11.) PDR: High Level Design Review				
12.) Final High Level Design Spec Report				

7.1 PROJECT STATUS (8 November 2018)

Concept exploration phase was completed following completion of the concept description and presentation at which point requirements analysis began. The project is still in this phase as the system requirements are developed in this document. Excursions definitions and budget have been developed at this time. The test plan is also in progress which is part of the design analysis phase.



PARTS STATUS			September	October	November	December
module	no					
RF Module	1	DSRC Radio	R			
Microcontroller	5	Tiva Launchpad	R			
Digital-to-Analog	2	DAC	R			
Steering Actuator	1	Steering Actuator	R			
Linear Actuator	1	Linear Actuator	R			

APPROVED	A
ORDERED	O
Projected Delivery	R
Received	R

Component Status (Version 1, Requirements Specification, 8 November 2018)

Module	#NO	Comments	Status	Availability Date
RF Module	1		ORD	HAVE
Microcontroller	1-5		ORD	HAVE
Digital-to-Analog	1-2		ORD	HAVE
Steering Actuator	1		ORD	HAVE
Linear Actuator	1		ORD	HAVE

Status:
REQ – requested
APP – approved
ORD– ordered
TBD – to be determined