Project Two
Advance Embedded System Design
Spring 2019

EMBEDDED VEHICLE SYSTEM
(VEHICLE SAFETY SYSTEM)

Harsimransingh Bindra
Yasir Aslam Shah

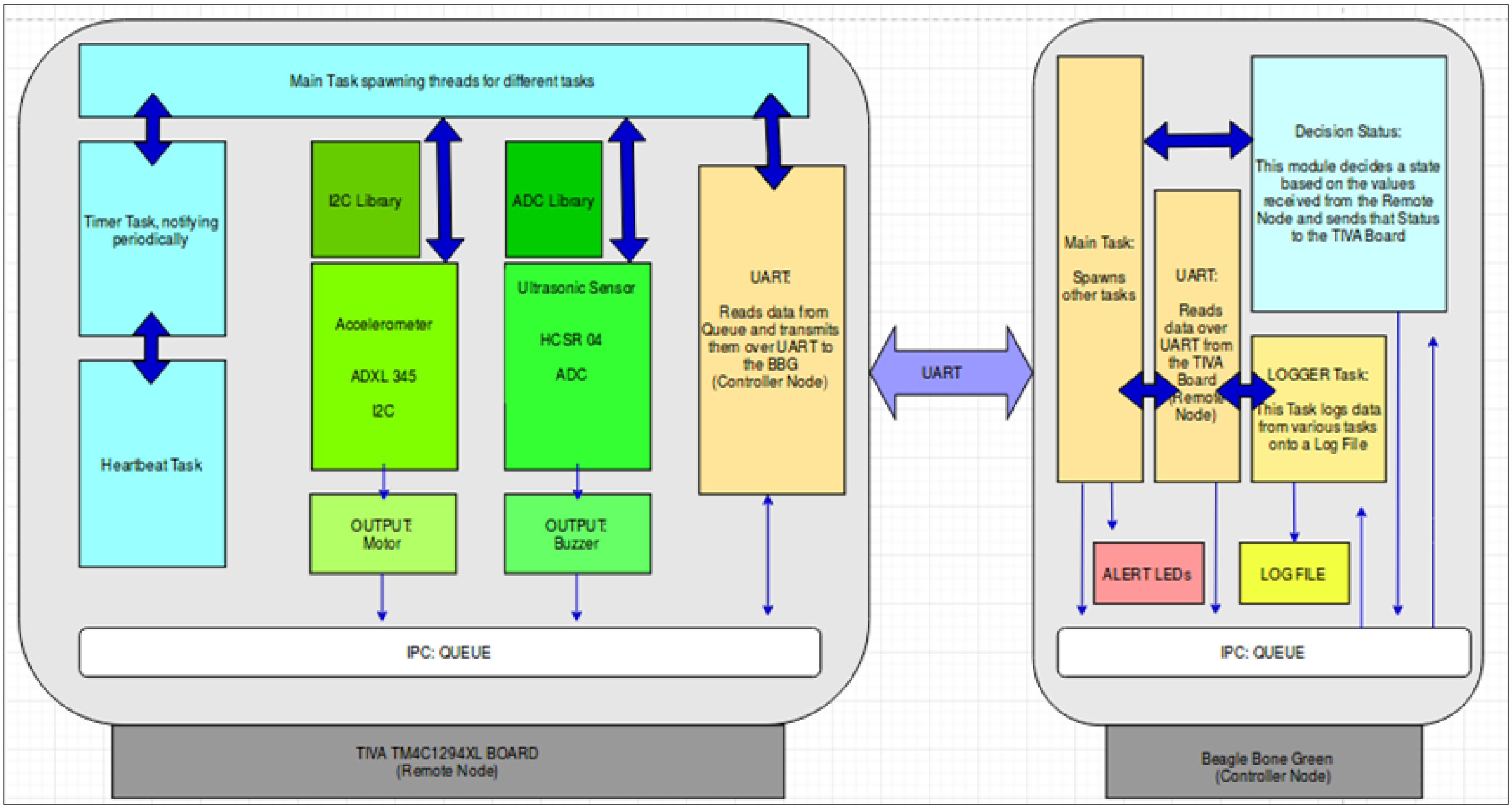
Contents

GitHub Link: https://github.com/harsimran2421/AESD-5013-002-/tree/master/Project-2
Media Link: https://drive.google.com/file/d/1Rx0ahvlc_pcPnv_vSMZyvwRARshrCd1l/view?usp=sharing

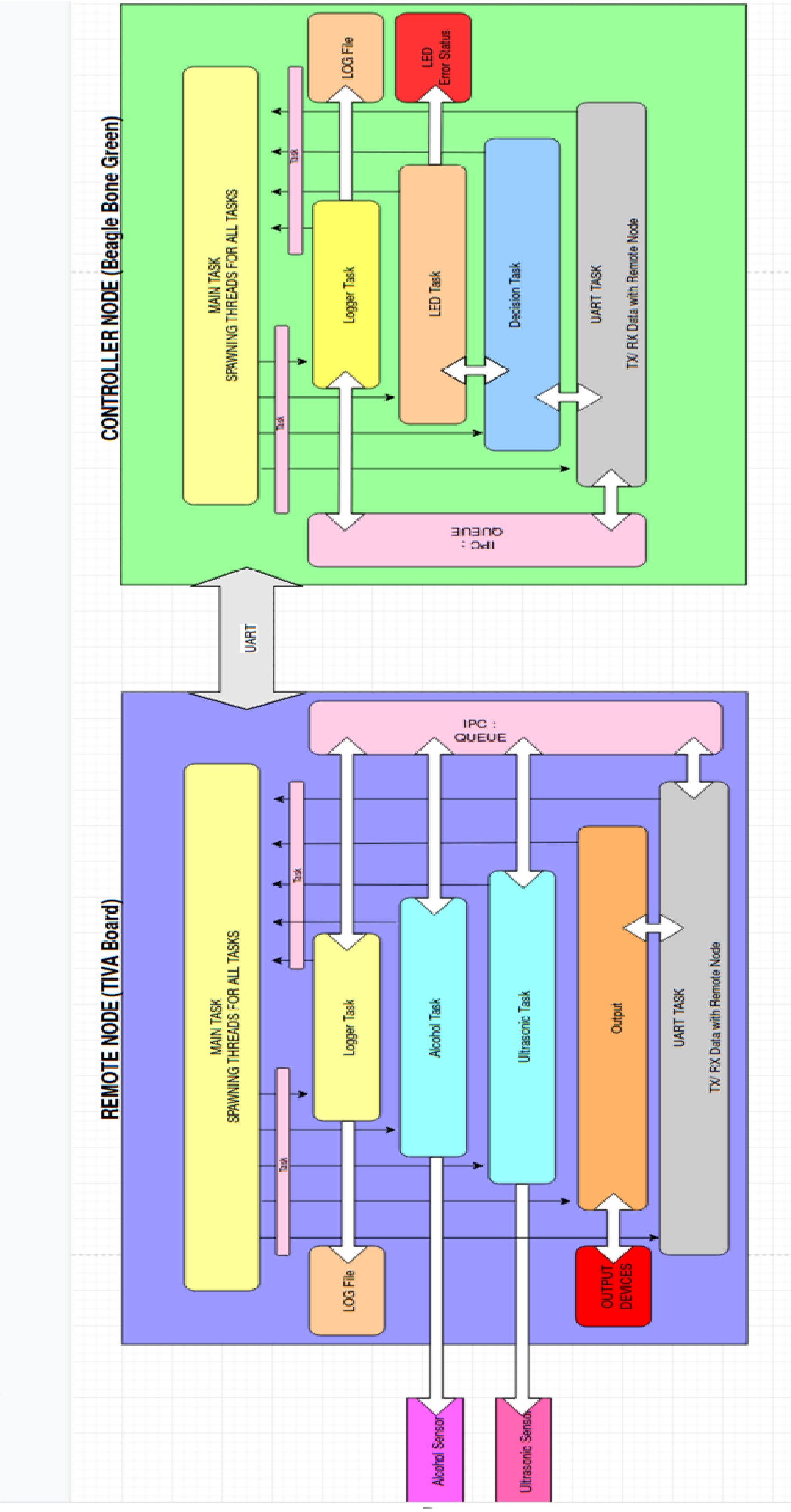
In this project, we are designing a **Vehicle Safety System** in which the system informs a user about any obstacles nearby within a threshold value using output devices. The system also monitors the presence of alcohol and alerts a user. Both sensor applications provide extra vehicle safety to user. In this project we are implementing LEDs, Motor and Buzzer as output devices. The project is based on TIVA Board (FreeRTOS) and Beagle Boone Green.

SOFTWARE ARCHITECTURE

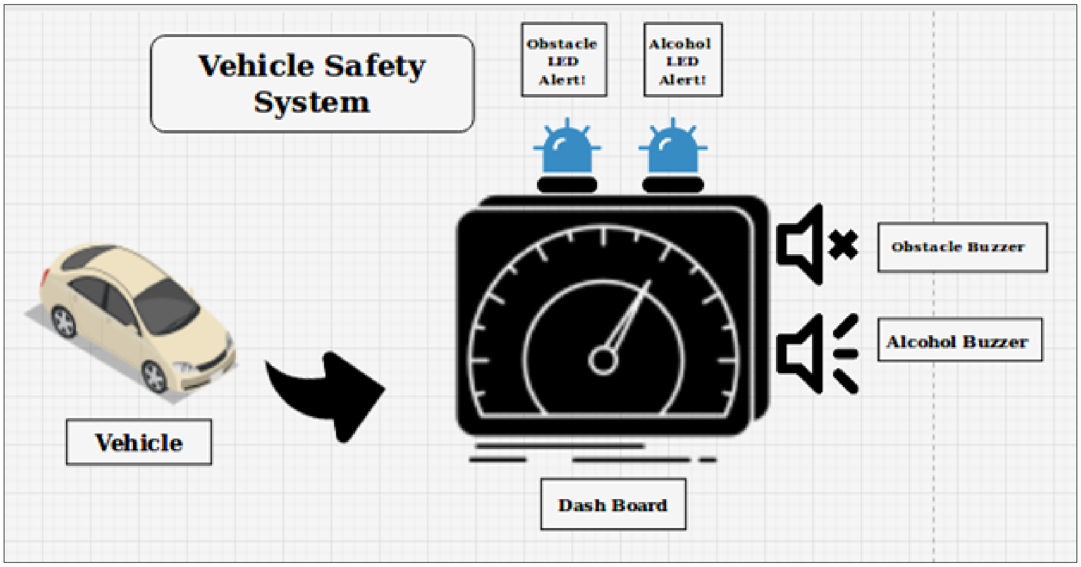
The initial software architecture included in the Project proposal is as follows:


Main Task spawning threads for different tasks
Timer Task, notifying periodically
Heartbeat Task
I2C Library
Accelerometer
ADXL 345
I2C
OUTPUT: Motor
ADC Library
Ultrasonic Sensor
HC SR 04
ADC
OUTPUT: Buzzer
UART:
Reads data from Queue and transmits them over UART to the BBG (Controller Node)
IPC: QUEUE
TIVA TM4C1294XL BOARD
(Remote Node)
UART
Main Task:
Spawns other tasks
UART:
Reads data over UART from the TIVA Board (Remote Node)
Decision Status:
This module decides a state based on the values received from the Remote Node and sends that Status to the TIVA Board
LOGGER Task:
This Task logs data from various tasks onto a Log File
ALERT LEDs
LOG FILE
IPC: QUEUE
Beagle Bone Green
(Controller Node)

The updated Software Architecture:



INTRODUCTION:



The project is designed to implement a **Vehicle Safety System** in which the vehicle dash board notifies a user in case of any obstacle nearby within threshold values set by the user. The dash board represented by the TIVA board as Remote node also monitors the alcohol quantity near the driver so to alert that the driver is drunk. The alert is created using LEDs, Buzzer and a Motor.

In this project Beagle Bone Green acts as a Controller Node that receives sensor values from the Remote node and decides the status of the output peripherals.

This entire project is developed on a TIVA Board and a Beagle Bone Green Board as Remote node and Controller Node respectively. The TIVA board runs on Free RTOS, bare metal firmware architecture and all the sensors are interfaced on this board. Each sensor on TIVA Board is ran initially to test various sensor unit modules like reading and writing configuration values. The values read from the sensor are sent to a logger task using IPC message queues. The two sensor tasks share a common message queue that serve the logger task. The logger task logs each entry (e.g. start up, significant events, state machine changes, etc.), as well as error/faults/failures with time stamp. Each logged information at the TIVA Board is forwarded to the Beagle Bone Green Board using UART.

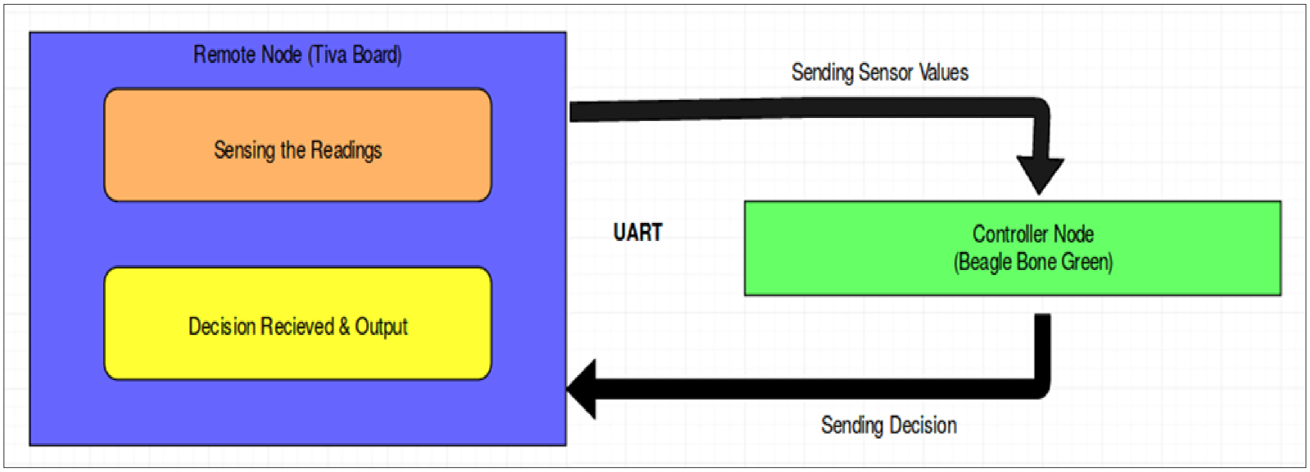
At the Beagle Bone Green (Controller Node) end, the data sent by the TIVA board is received through UART. The BBG module also logs the incoming data into a logger task. Each task in the BBG share a common message queue. The sensor data received from the UART is processed and the sensor values are compared with the threshold values set initially for each sensor by the user and flag values are set and communicated to the remote node which processes the flag values to define the state of the output devices.

The system also support a degraded level of service in the absence/failure of any single sensor and the level of degradation is represented by means of on board LEDs and LED Matrix. In case any sensor fails the initial Startup module, the level of degradation is represented by external LEDs at the controller node side. Each external Led represents each sensor and glows in case of any sensor failing the startup test. In case the system passes the Startup module, the system then forwards to thread and task creation. If by any chance any of the sensor is disconnected by user, the level of degradation is displayed by glowing on board LEDs at controller node. In case the sensors work correctly but the UART fails during the system runtime, the level of degradation is represented by turning on a on board LED at controller node.

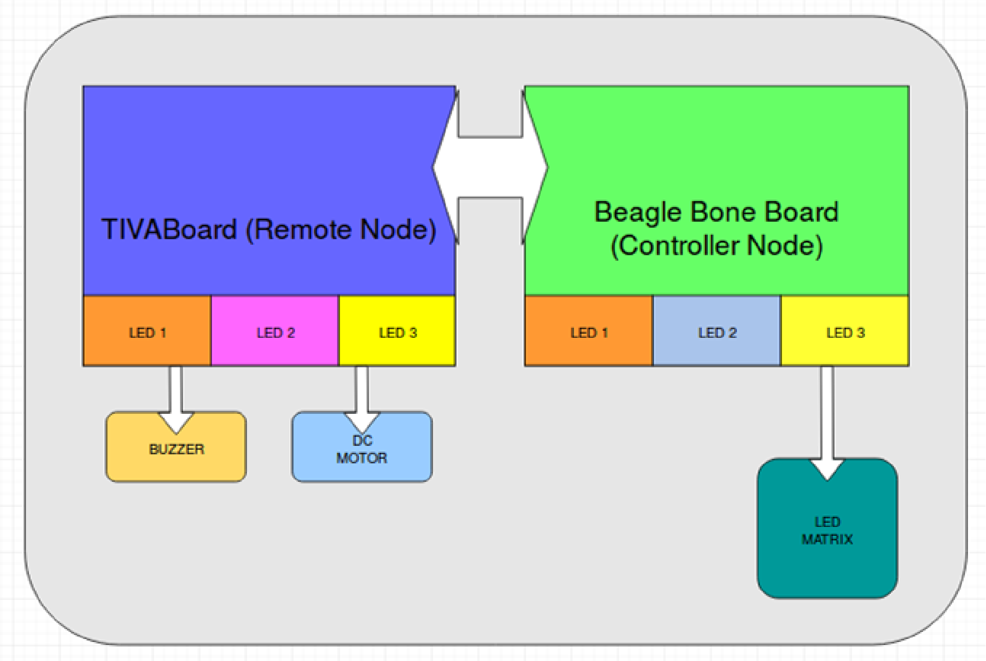
In this project, we are implementing Buzzer, Motor and LEDs as output devices.

The various modes of operation is also represented using a LED Matrix. The number on the LED Matrix denotes the level of operation. 1 stands for normal operation, 2 stands for degraded operation, 3 Stands when both sensors are disconnected and 4 depicts UART failure/unavailability of controller node.

The control node provides a feedback to the remote node as follows:



HARDWARE ARCHITECTURE:



The hardware architecture is divided into two modules as Controller Node and Remote Node

Controller Node:

It contains the following components:

- Beagle Bone Green
- LED Matrix

Remote Node:

- TIVA Board
- Buzzer (Output Device)
- Motor (Output Device)

ARCHITECTURE:

The system is divided into the following two nodes as below:

Controller Node (Beagle Bone Green)
Remote Node (TIVA Board)

The Controller Node:

This node acts as a main task and is based on Linux-based Beagle Bone Green. This task also contains an alert mechanism using on board LEDs for various tasks operational on the remote node based on FreeRTOS TIVA Board. This task also sends decision flag values to the TIVA board to generate output on real Output devices like LEDs and a Buzzer.
The controller node contains a UART module, a logger task, a decision task and a LED task.

- The LED Task alerts a user using on board LEDs in case of any violation like, sensor disconnect or UART fail by turning on different LEDs for Alcohol sensor disconnect, Ultrasonic sensor disconnect and UART disconnect.

- The decision Task receives sensor values from the UART Task decides the values of decision flags based on threshold values. The flags are sent back over UART in a full duplex mode to the TIVA board to manipulate the state of the output devices.

- The Controller Node also contains a Logger task that logs every activity in a log file. Every state change, Error handling, Fault occurrence and decision made is logged in a logger file.

- The UART task communicates with the UART module at the remote end and communicates with the TIVA Board in a full duplex mode.

The level of operation is represented using LED Matrix as well as on board LEDs. A blinking LED represents a healthy connection with Remote node where as a solid glow represents fault connection with remote node over UART.

The Remote Node:

The Remote Node is based on FreeRTOS on TIVA Board and is responsible for interfacing various sensors. The Remote node communicates with the controller node and generates output on real output devices. This remote task in our project consists of two sensors ; Alcohol and an Ultrasonic Sensor.

The remote node contains the following sensor task:

Alcohol Task

The alcohol sensor is used in this project to detect the alcohol level in a user's breathe or sensor ambient atmosphere. The alcohol sensor used in this project is Gas Sensor MQ3 based on ADC. One of the factors that provided a extra learning edge in this project is using a non I2C based sensor. The sensor is tested initially using Startup module. The sensor values read from the sensor are read and then pushed to a common bus that is further serviced through UART to the Beagle Bone Green Board (The Control Mode). The values are also displayed on the terminal. If the values cross threshold values at Control Mode, a Buzzer is generated at the remote end to alert a user along with an on-board LED.

Ultrasonic Task

In this task, we are using an Ultrasonic Sensor HCSR 04 based on GPIO. The basic purpose of this sensor is to detect any obstacle based on its distance reading. The sensor generates an alert whenever the distance crosses a certain threshold value defined by the user. A LED turns on to alert a user representing an obstacle within a set limit of radius. A motor also turns on alerting a user about the obstacle in its defined radius. We are turning on the motor only and only when the ultrasonic readings cross a certain threshold values and rest of the time the motor stays off so to implement energy efficiency of the system.

Logger Task

This task is responsible for accepting various logs from various on-board sources. A synchronization mechanism is implemented for protection from multiple log sources. The logger entry contains the following information in various data types as:

- Logger ID
- Log Message

Output Task

This task reads the decision flags from the UART to manipulate the state of the output devices; LEDs, Motor and Buzzer.

- Two On board LEDs on the remote node depict if the sensor reading is above a certain threshold value.
- A motor is turned on in case the ultrasonic sensor values cross the threshold values,
- A Buzzer is implementing in case the alcoholic sensor reading cross its safety value.
- A led Matrix is also implemented on the BBG side to display the mode of operation

Startup Tests:

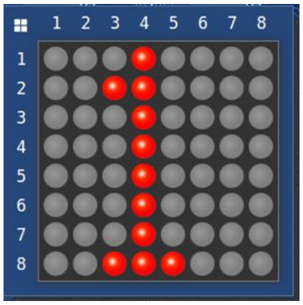
This task performs the startup tests for various sensors to make sure the sensors are connected and running. This module is ran initially as a STARTUP test. Once this test passes then only the system moves forward creating threads for different modules. Any startup failure results termination at start. Once the system starts communicating and in case of any failure, be it sensor failure, or anything the system demonstrates such a failure with LEDs so that the user gets to know about the cause of the failure. The LED matrix also displays the values related to each failure

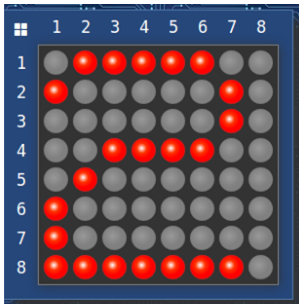
THREE LEVELS OF OPERATION:

This system consists of three level of application as follows:

Normal Operation:

In this normal operation mode, the system works just fine with Beagle Bone Green Board communicating with TIVA board through UART. The Sensors work fine and pass the sensor Startup module. This operation cab be displayed with no fault LEDs glowing on TIVA board as well as on BBG. The value represented on the LED Matrix is 1 for a normal operation



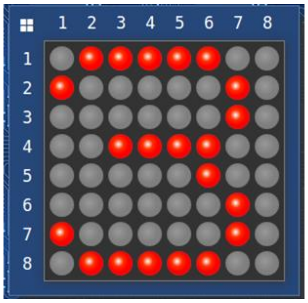


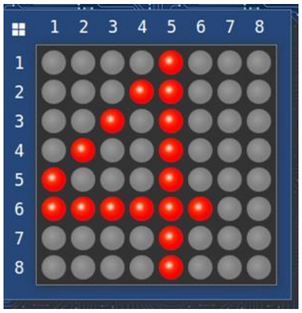
Degraded Operation:

In this degraded operation mode, the system works in a degraded operation mode. In this mode, only one of the sensors works while the other sensor doesn't work. The system depicts this operation by glowing a on board LED on BBG for each sensor. The LED Matrix displays a number 2 to represent this fault.

Failed/out-of-service Operation:

In this mode, the system has either a failed UART connection, or the system Startup   tests fails to pass for both the sensors, resulting in the failure depiction through fault LEDs. This mode is even valid in case a remote node is not available for the controller node to communicate. In this case the remote node decides the status of its output devices. The UART failure or unavailable controller node is depicted by glowing an on-board LED. In case the UART is working and the remote and controller node are connected properly, the fault node blinks telling a user about the healthy state of the system. The LED matrix displays a number 3 if both the sensors are disconnected, and a number 4 if the controller node is missing or the UART communication is broken



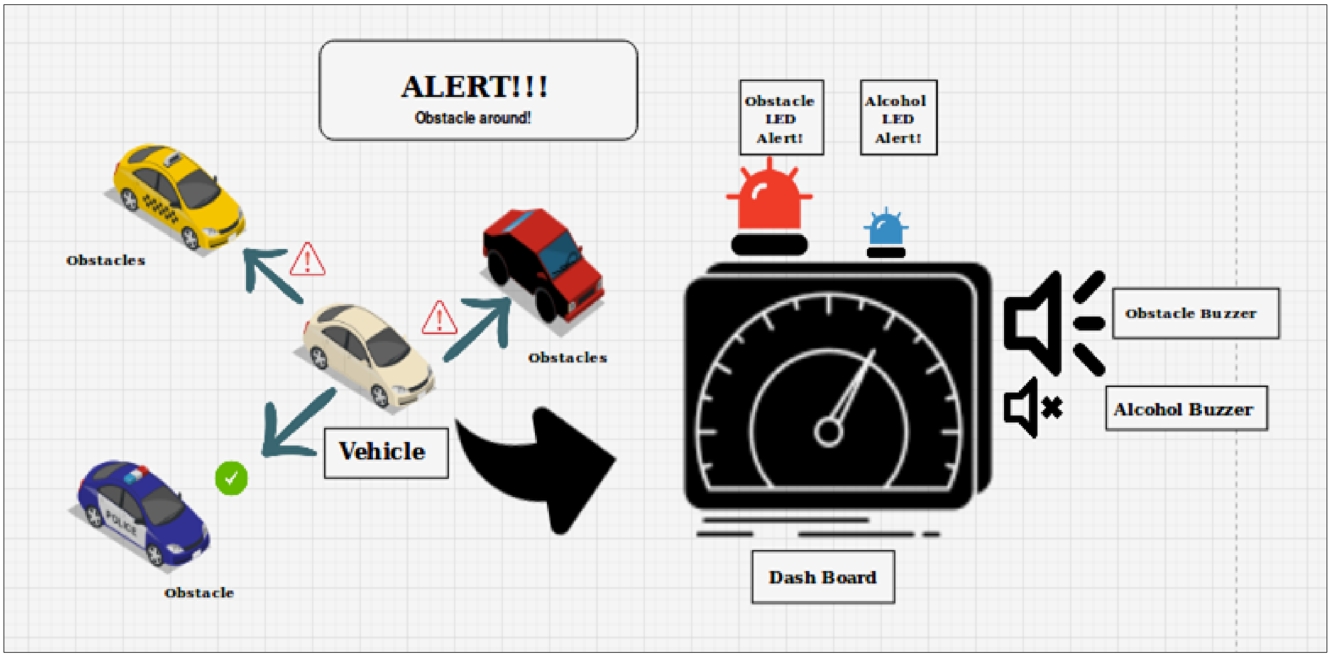


PROJECT REQUIREMENTS:

In this project based on the two sensors we have planned the following three requirements as below:

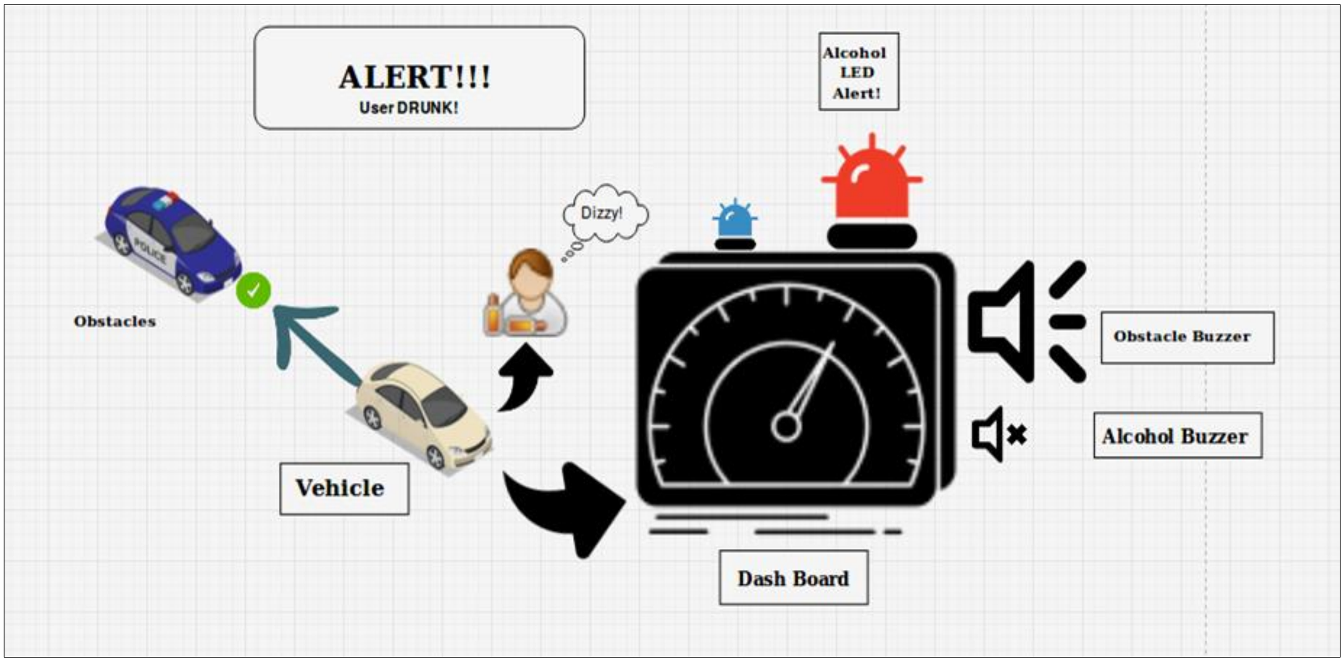
Requirement 1:

The TIVA Board (remote node) senses data from the ultrasonic sensor and sends the data over UART to the control node (BBG) for a decision that will drive the output peripheral (LEDs) at the remote node end. The control node judges the decision to be made by comparing the value from the ultrasonic sensor with the threshold values set by the user initially. In case the sensor value is less or equal to the threshold values, the remote node sets the state of its output peripheral likewise. In this project, the output state is defined using one of the on-Board LEDs and a Motor at the remote end. The LED glows in case the ultrasonic values cross the threshold value and turns on the motor. The motor is kept off for rest of the operation for energy saving



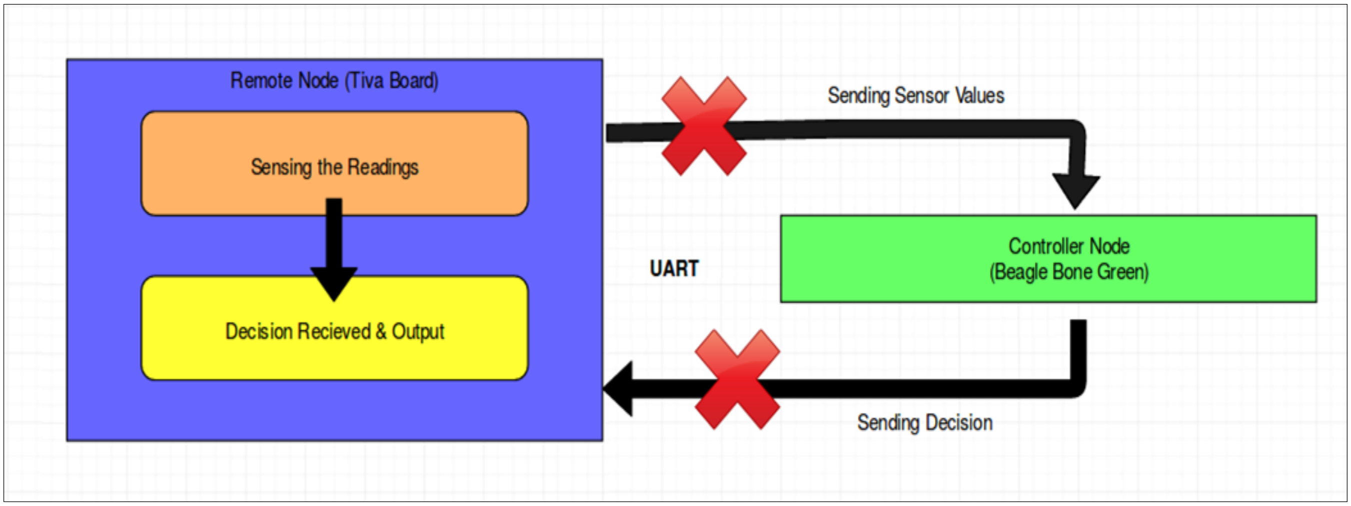
Requirement 2:

In this project we are implementing another sensor as alcohol sensor based on ADC. The TIVA board reads the value from the alcohol sensor and returns the value to the BBG. The BBG decides if the sensor values cross a threshold or not. In case the values cross the threshold value, the remote node changes the state of the output peripheral. In this project we are implementing LEDs as basic output peripherals and a buzzers to alert the user.



Requirement 3:

The third requirement is defined as a safety measure in our project. In case the remote node is disconnected from the control node, and the remote node does not have any control node to decide what to do with the output devices, the remote node in such a critical state takes care of the output devices itself as a safety measure.



CONNECTIVITY:

- In this project, the two sensors used are interfaced through ADC and GPIO (both non – I2C).
- The tasks on a Controller Node or a Remote Node communicate using message Queues
- The two boards communicate using UART in full duplex mode
- Various output sensors are interfaced using GPIO pins

TEST PLAN:

In this project we are testing the project at various levels of its runtime as well as at various level of its complexity. Initially all the sensor and communication modules are run for a Startup test. This startup test checks for all the different functionality of sensors and related communication. The system proceeds further to create different tasks and their related threads only when all the tests run successfully. In case of any failure with a single sensor or with all the sensors, the system runs in a degraded mode or failure mode.

Failure mode is when the remote node is missing or not connected to the controller node.

Degraded mode is when one of the sensors is not working and the system is running with only one sensor.

Test case scenarios:

Case 1:

System running correctly without any human intervention at start. The system takes sensor readings and the controller node is connected with the remote node. The system alerts user using output devices about the decisions made by the controller node based on the sensor values received from the remote node. The output devices respond as per the sensor values. The LED Matrix displays a number 1 for this kind of operation

Case 2:

System is initialized and the startup returns an error. Such a failure is demonstrated by turning on one on board LED at BBG (Controller Node) and displaying a number 3 at the Led Matrix.

Case3:

The system runs normally, however during the runtime if any sensor is disconnected, such a failure results in a degraded mode in which the system runs with only one sensor at its operation. Such a failure is demonstrated by turning on one on board LED at BBG (Controller Node) as well as by turning external LEDs at Remote Node. The LED matrix displays a number 2 for such kind of an error.

Case 4:

The system is tested when the remote node is not present or available or gets disconnected and is represented by LEDs. A blinking LED is present at the Remote End as well as at the controller end while the UART or the connection between the two nodes is healthy. In case of a UART failure or Controller Node unavailability, the blinking LED at both the nodes stops and a number 4 is displayed at the controller end on the LED matrix

TEST RESULTS:

The following table tabulates the overall performance of the system when different test cases are run. Here X depicts OFF and O depicts ON. The status of each LED, external LED and Output Device is depicted as follows:

		TIVA (Remote Node)					BBG (Controller Node)			
Case	Scenario	LED1	LED2	LED3	Buzzer1	Motor	LED1	LED2	LED3	Matrix
1	Normal Operation	X	X	X	X	X	X	X	X	1
2	Alcohol Threshold Cross	O	X	X	O	X	X	X	X	1
3	Ultrasonic Threshold Cross	X	O	X	X	O	X	X	X	1
4	Startup Failed (Alcohol disconnected)	X	X	X	X	X	O	X	X	2
5	Startup Failed (Ultrasonic disconnected)	X	X	X	X	X	X	O	X	2
6	UART working (Remote – Controller Connected)	X	X	X/O	X	X	X	X	X/O	1
7	Alcohol Sensor Disconnected at Runtime	X	X	X	X	X	O	X	X	2
8	Ultrasonic Sensor Disconnected at Runtime	X	X	X	X	X	X	O	X	2
9	UART Failed (Remote – Controller Connected)	X	X	O	X	X	X	X	O	4
10	Both sensors Disconnected	X	X	X	X	X	O	O	X	3

X = OFF
O = ON
X/O = Blinking

KEY LEARNINGS:

The following is the summary of the overall key learnings in the project:

- The half-duplex UART communication is simple to follow and doesn't require synchronization in comparison to full duplex communication where the receiver and transmission occur out of synchronization.
- The UART communication accessed by multiple tasks proved to be tricky as it also resulted to be out of sync at first.
- One of the most key learnings that we will be fruitful for future is not to reply on single hardware module and keep extras for backup as a single mis connection can burn your sensor up and cause a considerate delay in a project progress.
-

EXTRA CREDIT PART:

- Extra credit part done by representing operational status indicated by on board LED as well as on LED Matrix at BBG.
- Extra credit part done for each non-I2C based sensor.
- Extra credit part done by demonstrating Output using Buzzer, LEDs and Motor.
- Extra credit part done by using a total of 6 on board LEDs, a Buzzer, a Motor and a LED Matrix
- Extra credit part done for restoration of communication between the nodes, the Remote Node data/events is forwarded to the Control Node.
- Extra credit part Fast recovery of the system, the connection resumes communication within 100ms after the connection is restored
- Extra credit part done as a safety measure where the remote node takes control over output devices in case the control node is disconnected or unavailable.

REFERENCE:

- https://www.sparkfun.com/products/8880
- https://www.sparkfun.com/products/13959
- https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=2ahUKEwinvsnAlfXhAhULVa0KHW8wD9EQFjABegQICRAC&url=https%3A%2F%2Fwww.sparkfun.com%2Fdatasheets%2FSensors%2FMQ-3.pdf&usg=AOvVaw1EuCIQcfY8bNJRl84u4FUw
- http://users.ece.utexas.edu/~valvano/Volume1/E-Book/C6_MicrocontrollerPorts.htm
- http://beaglebone.cameon.net/home/serial-ports-uart
- https://www.geeksforgeeks.org/ipc-using-message-queues/
- https://www.geeksforgeeks.org/bits-manipulation-important-tactics/

-

PROJECT MEMBERS:

Harsimransingh Bindra	Yasir Aslam Shah
habi1225@colorado.edu	yash8307@colorado.edu