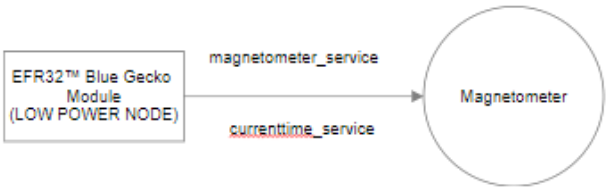
AUTOMATIC TRAFFIC CONTROL USING BLUETOOTH MESH

Individual proposal
Ayush Dhoot

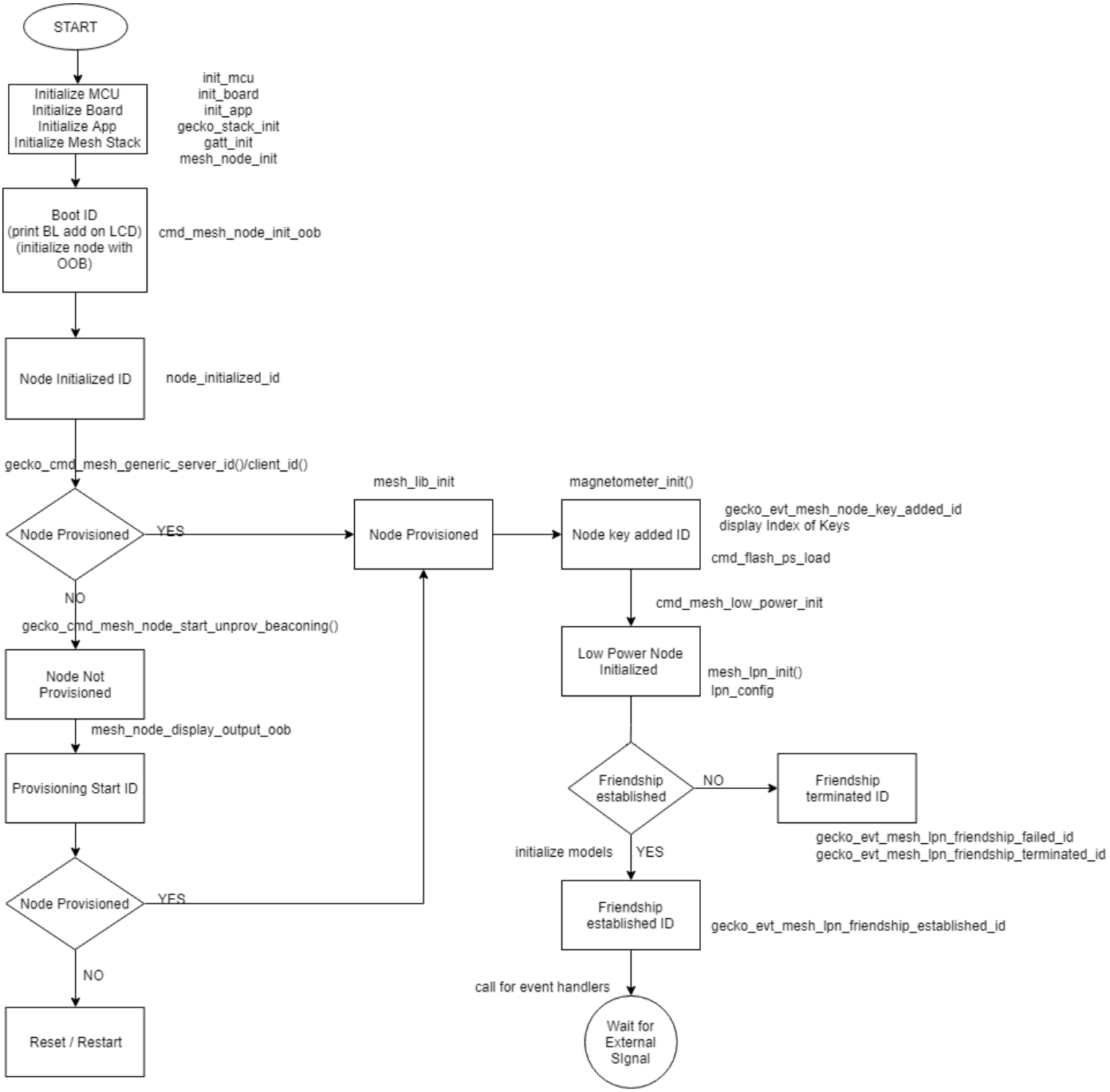
1. Describe what the problem this project or part of a team project addresses.
→My part of the team project is creating a low power node which will keep track of the car traffic in a lane and send the data to the friend node which will modify the on-time of the signal according to the traffic length.

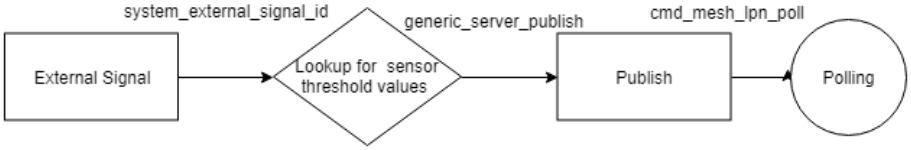
2. How does this project alleviate or solve the problem?
→My part will have a 3-axis magnetometer interfaced with the EFR32 BlueGecko board. The Low power node will have a magnetometer which will be waking up after a particular time to check the traffic length. According to the traffic data from all of the rest low power node, the friend node will give priority to the respective node. Along with the custom traffic service, there will also be a current time service which will make the project real-time.

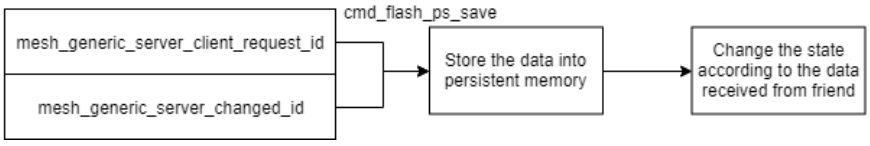
3. Functional block diagram of the individual project.



4. Software Block Diagram. (Flow Chart)







5. List of sensors for this project a. If a sensor needs to be acquired, provide Sparkfun breakout board p/n
→ I will be needing a 3-axis magnetometer.
SparkFun Triple Axis Magnetometer Breakout - MLX90393 (Qwiic)

6. What exposed services and client profiles will be implemented (a minimum of 2 not including TX power or Health Temperature)
→Current-Time service [GATT]
Magnetometer service [Custom-Traffic]

7. What persistent data will be stored to enable the project?
--> The Out-Of-Band verification details
Sensor data values
Threshold values for determining the change in the state
The Keys (handled automatically by the stack)
The states of the LEDs

8. Proposed development schedule for this project broken down to discreet components including target implementation dates.

TASK	START DATE	TIME (IN DAYS)	STATUS
STUDY THE DATASHEETS AND MESH MANUAL	10/29/2018	4	Done
IMPLEMENT PERSISTANT MEMORY ROUTINE	11-02-2018	4	Done
INTERFACING MAGNETOMETER SENSOR TO THE NODE	11-07-2018	4	Connection Successful
INTEGRATING SENSOR TO THE CODE	11-11-2018	4	I2C Code implemented and receiving data.
LOAD POWER MANAGEMENT OF THE NODES	11/15/2018	4	Done
FRIENDSHIP ESTABLISHMENT	11-18-2018	3	Friendship established
INTEGRATING LCD	11/19/2018	1	Done
DEVELOPING THE MESH NODES RELATIONSHIP	11/20/2018	7	Developed
INTEGRATING ALL THE NODES	11/28/2018	4	Integrated
VALIDATION OF THE PROJECT	12-03-2018	4	Validated

9. Verification Plan of individual project.

Sr No.	Verification Plan	Expected Result	Date of test	Actual Result	Passed
1.	Check if node is being provisioned by the Silicon Labs Mesh App	The node should get provisioned after adding it to the group using the App and display 'Provisioned' on the LCD.	11/05/2018	The node is provisioned using the App and 'Provisioned' is displayed on the LCD module.	Yes
2.	Check if the friend and LPN relationship is being established	The friend and low power nodes should establish the relationship and display 'LOW POWER' on LCD.	11/11/2018	The friendship is established and 'LOW POWER' is displayed at LCD.	Yes
3.	Check if the node is being provisioned with out-of-band security authentication	The node should be provisioned only after out-of-band authentication.	11/09/2018	Out of Band authentication working with a single digit passkey.	Yes
4.	Persistent data	The sensor data, threshold values and the state values should be retained.	11/22/2018	Threshold data and states can be stored and regenerated even on reset.	Yes
5.	Testing sensor	Acquire data from the sensor Magnetometer (LMS303C)	11/11/2018	No data is being received from the sensor.	Yes
6.	Interfacing the sensor with the software	Connect the Magnetometer (LMS303C) to the EXP Header using the I2C protocol.	11/10/2018	Connected and wrote the code for I2C protocol.	Yes
7.	Checking case (Traffic Congestion scenario)	If the sensor reading is more than threshold then send message to Friend node indicating traffic congestion.	12/04/2018	If the sensor reading crosses the threshold, it enables the friend to service it first and then resume the normal operation.	Yes
8.	Checking LPN node (No Traffic congestion scenario)	The LPN node should sleep and be in the lowest possible energy mode.	12/04/2018	There is normal operation where the node sleeps and wakes up only when there is a publish message from the friend	Yes
9.	Energy Profiler verification	Make sure the sensor is working in the lowest energy mode	12/04/2018	LPN goes into EM3 with an average current of 100uA	Yes
10.	Testing the node	The low power node is functional and working according to the requirements.	12/06/2018	The LPN is functional according to the project requirements.	Yes

10. How was this project designed to optimize energy usage?

→ In this project, I have interfaced a magnetometer using I2C protocol. The data from the magnetometer is collected at certain intervals of time and therefore the node is in sleep mode for the rest of the time. Moreover, the poll timeout is set as 5 seconds, so the node wakes up every 5 seconds to check if there is any data in the friend queue. I verified the same using the Energy Profiler from the Simplicity Studio as the average current was around 100uA.

11. With security in mind, how does your project's security implementation support the end application and provide details on how it was implemented in your project?
→ The project has an implementation of Out-of-Band Authentication for solving the security problems. The OOB is used for defending against the Man-In-The-Middle Attacks. In this project, there was two digit OOB authentication key generated and it was displayed on the serial port terminal. On the mobile app, the two digit is entered and then the provisioning is completed.

12.List 5 lesson learned from doing the assignment that were not taught in lecture or an earlier assignment? (INDIVIDUAL)
→

1. Implementation of OOB authentication.
2. Learnt how to use Bluetooth SIG Models.
3. Developing of Low Power Node (Bluetooth Mesh)
4. Interfacing Magnetometer using I2C protocol and enabling it only on LETIMER interrupts
5. Persistent Data routines implemented

13.Summarize the final status of your individual project.
→ The low power node was provisioned using two digit OOB authentication key. The low power node stays in sleep mode. The magnetometer sensor is interfaced using I2C protocol and the sensor reading is taken only when there is interrupt from LETIMER. The low power node wakes up to connect with the friend node and receive the data. I have implemented that using the Level models of the Bluetooth SIG. The normal operation will have the friend node sending ON/OFF data for the LED0.(Traffic light) If the magnetometer threshold is crossed, the low power node will publish the data and the friend node will service that low power node(using unicast addressing) and then again resume the normal operation. Also, if the LUX value of the luminosity sensor at the friend node crosses the threshold, it sends a light ON message (Street Light) to the whole network (using group addressing) for the LED1. In the meantime, the low power node sleeps to lower energy modes. I verified that using the Energy Profiler. Also, I have implemented the persistent storage routine. If in case of power failure (reset), the node restarts with the same state as the one before the reset.