IoT Project Report – Smart Parking: Friend Node

By Sarthak Jain

1. **Describe what problem this project or part of a team project addresses**

   a) The first problem is that of fires in an enclosed area, which aren't easily detected.

   b) The second problem is that of energy optimization. It is not viable to keep the motion and proximity sensors online all day long, as too much energy would be wasted. Also, it needs to be decided when the light allotted to each parking slot comes online.

   c) The available slots need to be displayed at a location which can be seen by each driver on entry.

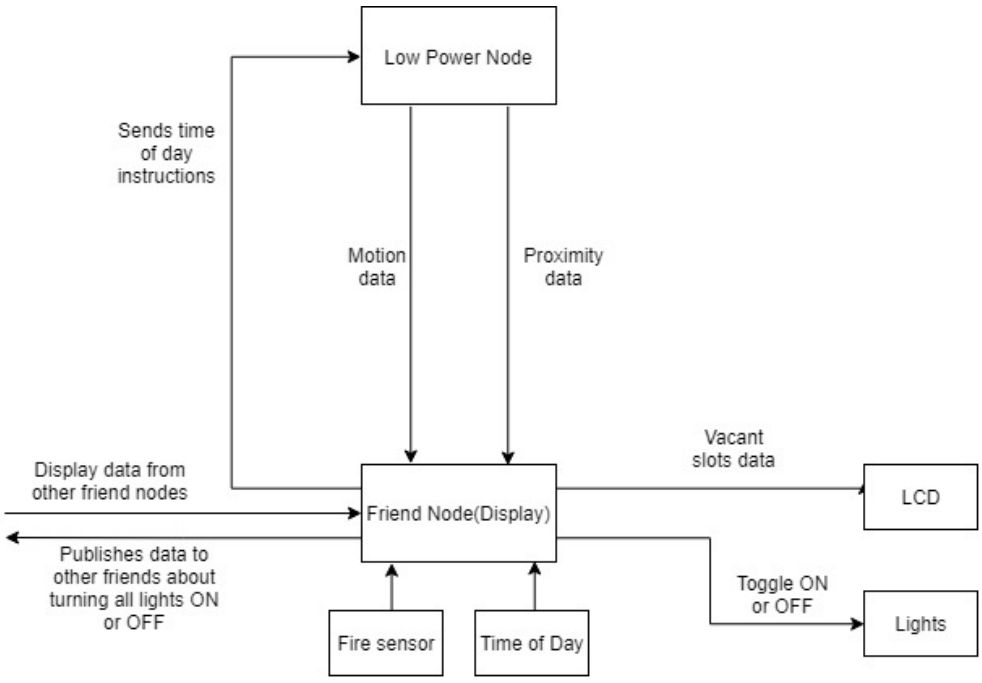
2. **How does this project alleviate or solve the problem?**

   a) The friend node has a fire sensor on it, which uses gas (carbon monoxide) detection. In this way, a single fire sensor for 12 parking slots is sufficient to detect any outbreak.

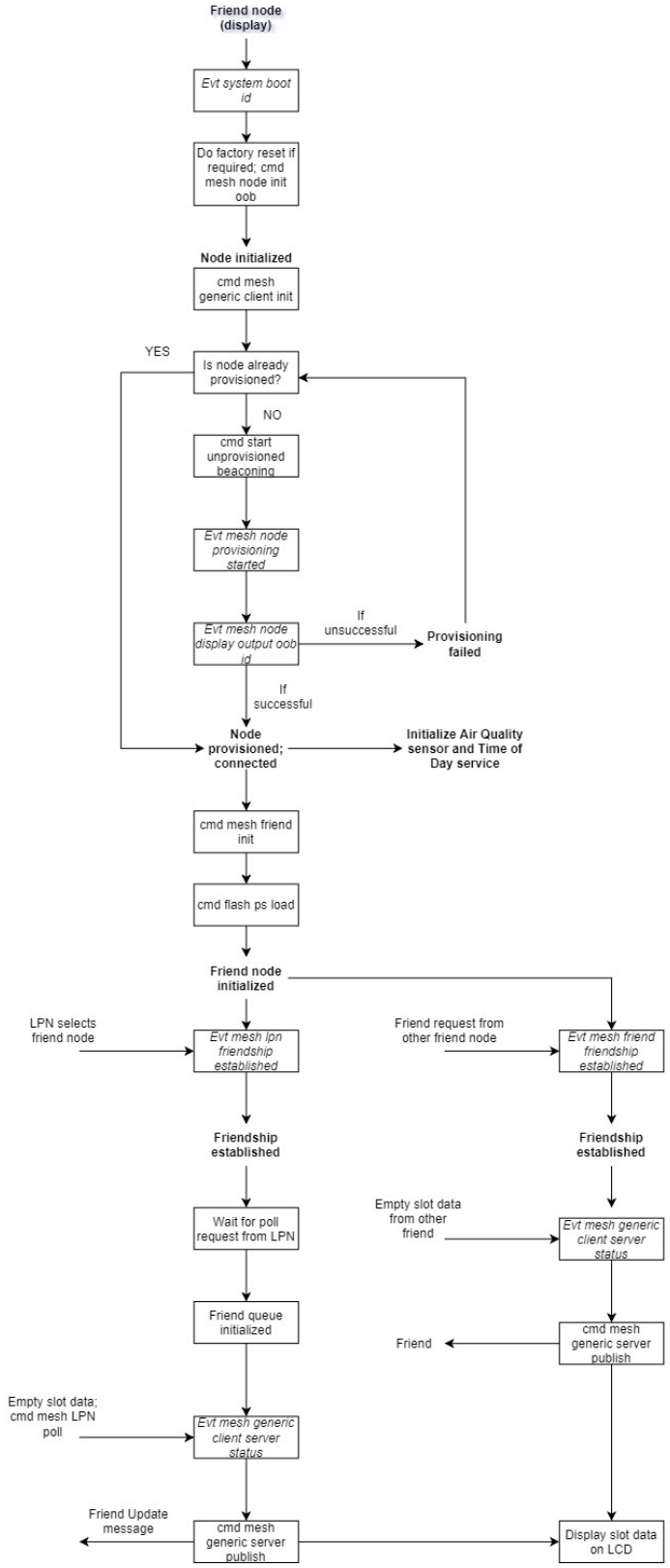
   b) Energy optimization is done during the day by taking data from the motion and proximity sensors on the LPN. If motion is detected, the friend turns on the light. Also, during the day, the Friend tells the LPN to take proximity sensor readings at a high frequency (eg. 30 seconds).
   At night, according to the Time of Day service on the Friend node, the lights are turned on, and the motion sensor is turned off. Also, since at night the frequency of cars is less, the proximity sensor can take readings at a slower rate (eg. 5 minutes). In this way power management is achieved.

   c) The friend node at the display end also controls an LCD on which it displays the data relayed to it by the other friend nodes, which shows up as vacant slots on the LCD, at a convenient location.

3. **Functional block diagram of project:**



4. Software Block Diagram



5. **List of sensors for project**
   a) Adafruit CCS811 Air Quality Sensor Breakout - VOC and eCO2

6. **What exposed services and client profiles will be implemented?**
   Two exposed services and profiles will be implemented:
   a) Time of Day – We obtain the time of day and convey the same information to the LPN. Based on the data obtained in the Time characteristic, the LPN takes a decision to turn off the motion sensor and change the frequency of readings taken from the proximity sensor.
   b) Fire Sensing – A fire sensor located near the Friend node conveys information about the presence of a fire, by making decisions basis the gas composition of the surrounding air. The Friend node takes some action (eg. Toggling all LEDs at the same time) based on the Fire characteristic value.

7. **What persistent data will be stored to enable the project?**
   The persistent data stored at friend node will be the various keys for mesh implementation (eg. Device Key, Network Key, Application Key). Other data stored in flash will be sensor threshold values, LPN state information and RTCC counter value.

8. **Update development schedule**

Sr. No.	Description	Estimated date of completion	Actual date of completion	Whether Passed
a.	Integrating LCD to application	11/1/18	11/1/18	Yes
b.	Developing mesh model and service code	11/20/18	11/22/18	Yes
c.	Developing Friend/LPN relationships	11/15/18	11/15/18	Yes
d.	Developing persistent memory routines	11/18/18	11/30/18	Yes
e.	Interfacing software to new sensor	11/22/18	11/27/18	Yes
f.	Integrating sensor to application code	11/27/18	12/2/18	Yes
g.	Integrating team member projects	12/2/18	12/5/18	Yes
h.	Project validation	12/9/18	12/9/18	Yes

9. Verification Plan

No.	To be Verified	Definition of passing	Date Test Performed	Measured result	Passed
1.	All Devices provisioned with the help of OOB	Displayed successfully as provisioned devices on the silicon labs app by entering the passcode	11/9/2018	Provisioned using app and check the result of the API command	YES
2.	Air Quality sensor Interfaced with the board	Receiving output values correctly as viewed on display LCD	11/12/2018	One sensor interfaced	YES
3.	Persistent Data routine implemented as expected	Sensor data, threshold values and state information retained on system boot id	11/30/18	Persistent data stored and retrieved successfully in Flash	YES
4.	Friendship established between LPN and friend	Making sure it enters the friendship_established event on both nodes by printing on serial port	11/15/2018	Checking the result of the api commands and viewing on terminal	YES
5.	Friend Queue implemented, and friend update messages sent	LPN receives update messages accurately.	11/27/18	Messages received on LPN side only when LPN wakes up, regardless of when Friend send the message	YES

6.	Implementation of group addressing mode	Message sent by friend node is implemented on all nodes subscribing to the group (eg. Switching lights on after 12 am)	12/3/18	Message received by all nodes which subscribe to the group address	YES
7.	Friend node displays empty slot numbers on the LCD	The Display node receives the data from the other friend node and the correct slot numbers are displayed on the LCD.	12/4/18	Data as sent by LPN correctly displayed on LCD	YES
8.	The friend node correctly determines time of day and defines model.	Intermittently displays time of day data on friend node's side.	12/2/18	Time of Day correctly measured by comparing with a clock	YES
9.	The Friend node is able to detect fire correctly and send fire alarm	If by increasing CO2 air content around the sensor, other friend receives a fire message	12/4/18	Other friend receives fire detection alarm	YES
10.	Implementing On/Off and On/Off control models	State of nodes can be changed accurately as and when required	12/5/18	Other nodes correctly change states according to data sent	YES
11.	Implementing mesh hopping	If messages received by receiving node when transmitting node is moved out of range and a relay node is placed in between	12/8/18	Messages relayed even when transmitting node is moved out of range of receiving node	YES

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

The project optimized energy usage as a whole, and not on the Friend side. Energy optimization is not required or wanted on the Friend side. On the LPN's side, however, it is required. The LPN does so by implementing both of its sensors as interrupt-based sensors. It wakes up in one of two situations, either when one sensor gives an interrupt, or when it must wake up to poll the friend node for any messages in the queue. In the latter case, the Friend node has a small role to

play, in that the LPN can be configured to wake up less frequently during night time, since there would be no frequent movement of cars. This configuration of less frequent wake up time is provided by the Real Time Service on the Friend's side.

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?**
    Out of Band authentication was implemented in our project. At the time of provisioning, an OOB passkey is required to be entered by the user to authenticate whether the device being used is same as the one being provisioned. In this way, only a device part of the mesh network can receive or send messages. If the device is deleted from the network, it can no longer receive those messages. In this way, some initial form of security is implemented.

12. **List 5 lessons learned from doing the assignment that were not taught in lecture or an earlier assignment**
    a. Team project management and how to balance assignment of work in a medium-sized group. I learned that one has to keep the group's target in mind, and not the individual's target, since eventually the demonstration counts as a group demonstration.
    b. I learned that although the mesh documentation is not quite adequate or precise, the implementation of it can be quite flexible. The stack supports a variety of functions, and can in a way, be fooled to think it is performing one implementation while it is actually performing a whole group of them. Case in point: although the app does not technically support provisioning of a node as a control model, the node can still be initialized as a client a well as server model and both sets of API's can be used by simply changing the functionality of the node in the app's configurations.
    c. Use of software timers – A very helpful concept which I discovered from the example codes. Although one does need to be wary to not overuse software timers, they can be quite useful in quickly setting up timing interrupts, without having to go through the entire procedure of setting up a peripheral. I found them to be a powerful tool if one is not too pedantic about the accuracy of time measured.
    d. I learned that although sensors can have a lot of features, the basic way of setting up and using most of them is quite similar. It's just a matter of reading the datasheet and marking out the more relevant pieces of information. For someone like me with little practice in interfacing sensors, this was a fascinating learning experience, realizing that the logic for using sensors stays the same, though the language might vary.
    e. Learned further about the use of structures and pointers in embedded coding. The mesh examples really dig deep into these concepts and use nested structures extensively. Understanding the same without any documentation and simply by reading code helped extend my own understanding of how structures can be used.

13. **Summarize the final status of your individual project.**
    In summary, my friend node has the capability to receive messages sent from 3 Low Power Nodes and can set the status of each LPN to increase or decrease the wake-up time based on the Time of Day service (after 10 pm or after 8 am) using RTCC. It has a fire sensor interfaced to it, which detects fires based on the CO_2 content in the air and sends a fire alert to all the other

friend nodes if a fire is detected. The Friend node receives parking slot data sent to it from the LPN and displays the same on an LCD. The Friend node also has a central button press control on it, which allows the Friend to send out a message to all LPNs commanding them to turn their lights on or off.

14. **What were the difficulties encountered on the project?**

    a. Constant issues with provisioning the node, as the Silicon Labs app is not quite cooperative at certain times.

    b. For some unknown reason, my node would refuse to enter the friendship established event towards the beginning of the project. I still haven't figured out the reason for the same, as the node simply started to bond with LPNs one day.

    c. Even after friendship was established, my friend node would refuse to enter the client server status event or even send messages to the LPN for a time. Eventually, it turned out I had not added the Client On/Off functionality in the Bluetooth SIG models of the .isc files. A silly overlook, but one that cost me over three days.

    d. My friend node has the capability to set the LPN's state(unicast) and publish to it as well. After almost the entire functionality was up, the publish simply stopped working one day. That was our own experience of the fact that the node cannot have more than one functionality at a time, ie. the node can be provisioned as either an On/Off model, or an On/Off Control mode, not both.

15. **WOW factor**

    The WOW factor in the friend node is that the RTCC has been configured to keep a real-time calendar. Even if the friend node is given power and left on, it will keep track of and display the time, day, month and year for the span of over a year, giving the effect of a calendar.