Home Security System with LIFI

Intro to Embedded Systems – CSE221

Submitted to

Prof. Sherif Hammad

Eng. Mohamed Tarek

Eng. Hesham Salah

Group: 15

Ahmed Wael Ibrahim Mohamed	20P3343
Mustafa Usama Abdelrahman	20P5556
Ahmed Sameh Mahmoud Nabieh	20P2914
Farah Ahmed AbdelRehim Tharwat	20P1269
Salma Mohamed ElSoly	20P6375

Table of Contents

List of Figures

System Design

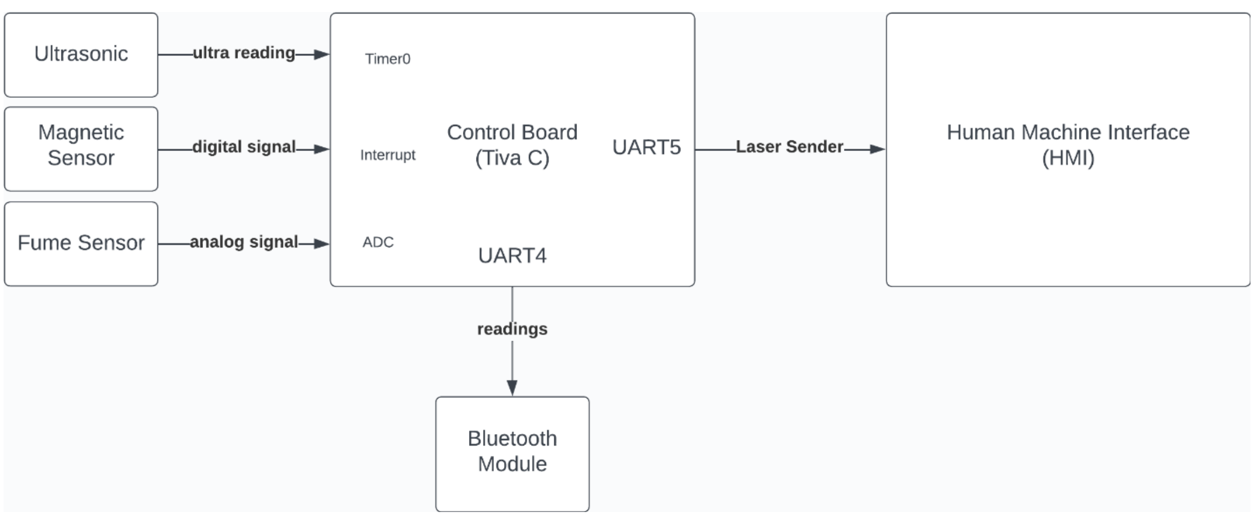
Introduction

Introducing our cutting-edge Home Security System, harnessing the power of Li-Fi (Light Fidelity) technology to deliver unparalleled safety and control within your home environment. Designed to safeguard your space comprehensively, our innovative system combines advanced functionalities to detect fire hazards, intrusions, and door open/close actions with swift precision.

At the core of this state-of-the-art solution lies a sophisticated detection mechanism that not only identifies potential threats but also empowers users with versatile interaction methods for seamless control and monitoring. With our intuitive Human Machine Interface (HMI) and a user-friendly mobile application linked via Bluetooth connectivity, you can effortlessly oversee and manage the system's functionalities from the convenience of your fingertips.

Experience a new level of peace of mind as our Home Security System prioritizes your safety through swift detection, efficient response, and user-centric control, all powered by cutting-edge Li-Fi technology. Welcome to a secure and intelligent home environment redefined by our comprehensive and adaptable security solution.

System Layout



As illustrated in the above figure, the system is composed of 2 parts: The **control microcontroller** is responsible for managing sensor readings and reporting alerts to the other system components or to the application. And the **Human Machine Interface (HMI)** which is the microcontroller that enables the user to control the system functionalities.

The user functionalities include:

- Turn off/on the system.
- User is notified with LED and Buzzer
- Check current alert on LCD.
- Mute the buzzer.

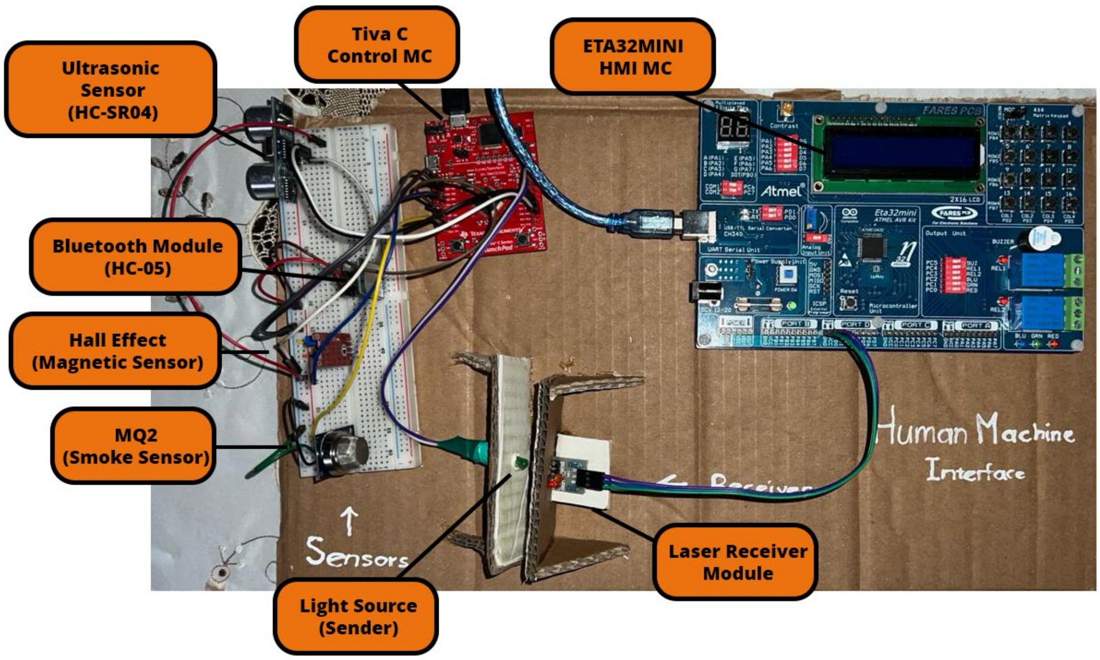


Figure 1: System Layout

Li-Fi Implementation

We've successfully integrated the Li-Fi communication protocol into our system by leveraging the UART serial communication protocol. This achievement involved establishing UART communication between two microcontrollers at (9600Hz, No parity bits and 1 stop bit). Subsequently, we replaced the wired connection between these microcontrollers with Li-Fi devices.

The setup comprises a laser module connected to the Tx pin of the control microcontroller (Tiva C Launchpad) and a laser receiver module linked to the Rx pin of the HMI microcontroller (ETA32Mini development kit). The implementation of UART ensures rapid communication between the microcontrollers, translating into swift notifications for alerts.

Furthermore, the integration of laser communication facilitates extended-distance installation for the microcontrollers, although it necessitates heightened installation precision. For further insights into our system's implementation selection, including overcoming obstacles, please refer to the **Obstacles** section on this matter.

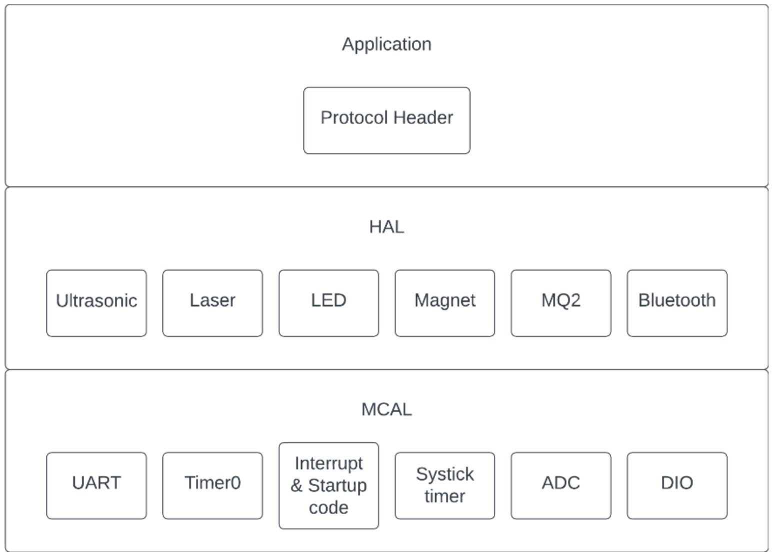


Figure 2: Control Microcontroller Layered Architecture

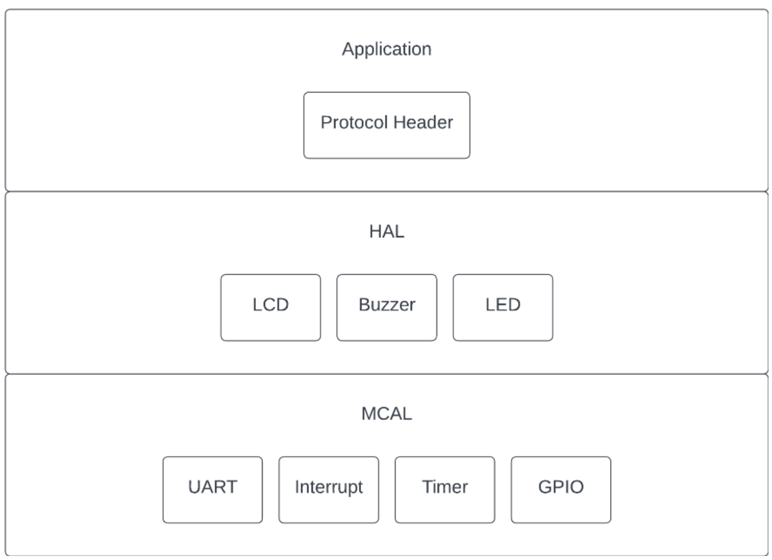


Figure 3: HMI Microcontroller Layered Architecture

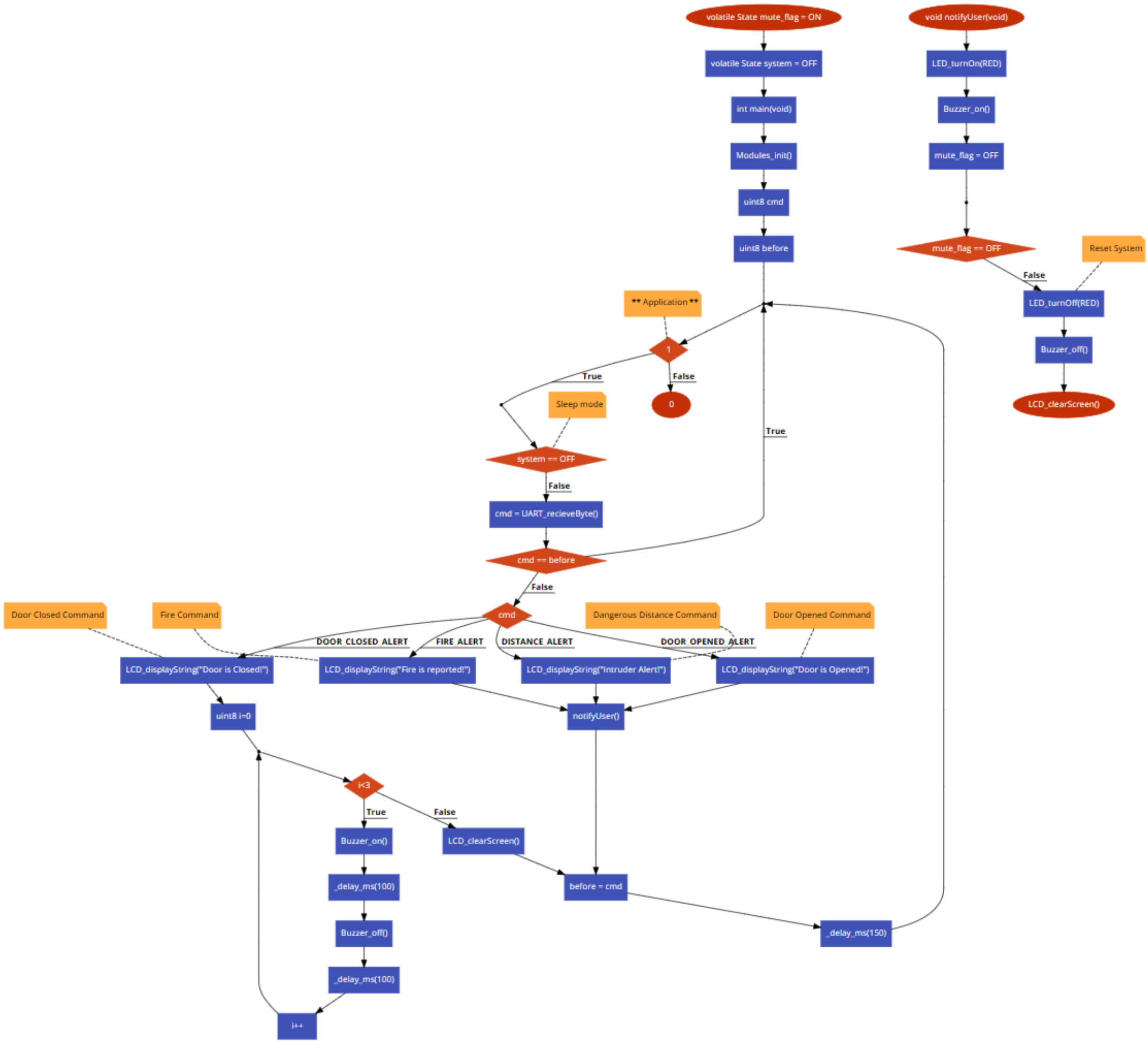


Figure 4: HMI Program Flowchart

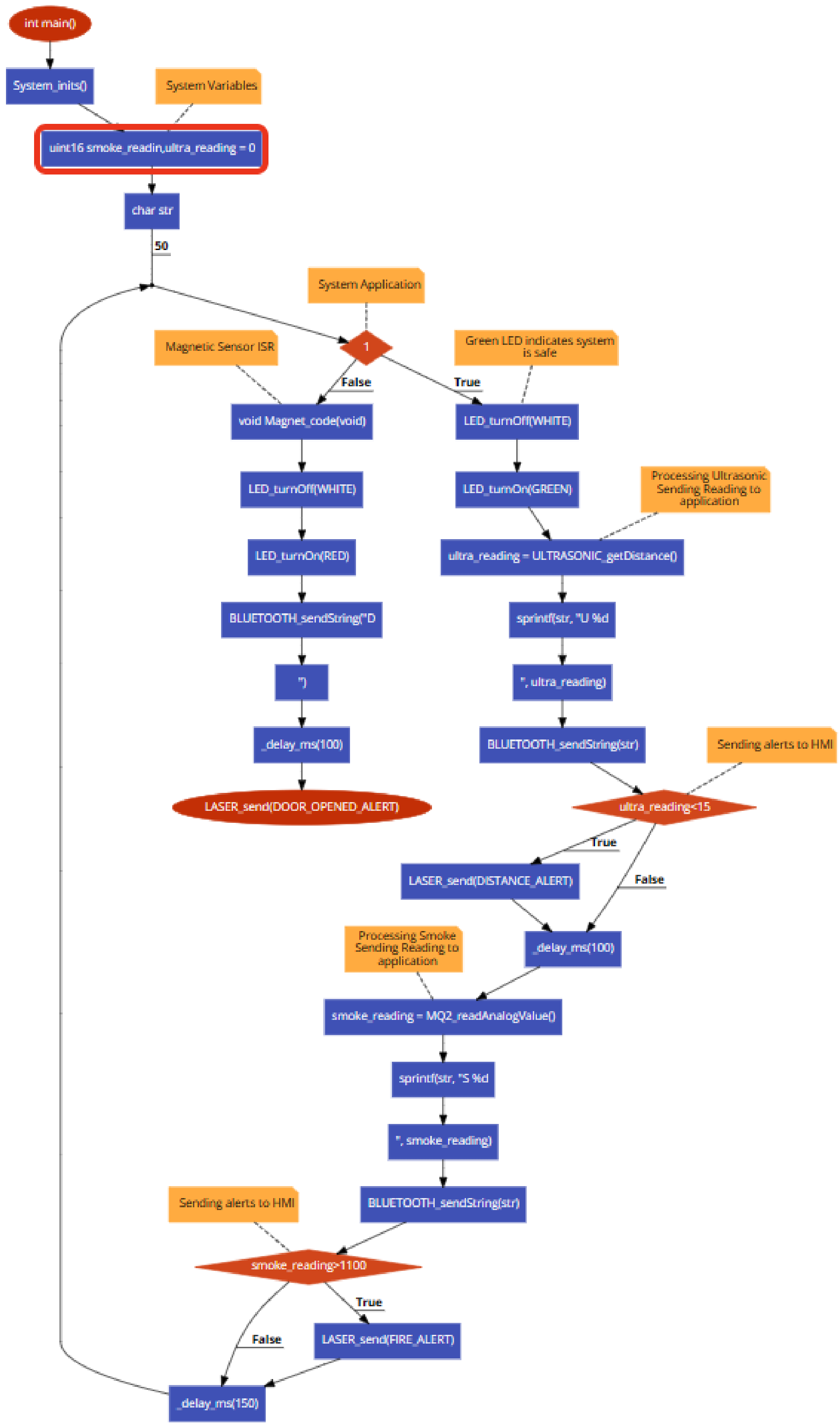


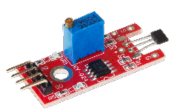
Figure 5: Control Microcontroller Flowchart

List of Components

In this section we will talk about the list of components for both the transmitter board and the receiver board.

Transmitter Board

This is the list of components added to the Tiva C Launchpad board to configure sensors and send data through LIFI.

	HC-SR04 (Ultrasonic Sensor)	Used for detecting intruders by measuring distances to the person and how far they are from the sensor
	Hall effect sensor (Magnetic Sensor)	Used for detecting Door Open/Close action by detecting the magnetic field from a magnet attached to the door
	MQ2 (Fume Sensor)	Used for detecting fire by detecting fume emitted from fire
	Laser Module	Used to send data by light to the receiver board
	HC-05 (Bluetooth Module)	Used to send data to the mobile application by Bluetooth

Receiver Board (HMI)

This is the list of components added to the ETA32MINI development kit to receive the data transmitted through LI-FI

	Laser Receiver Module
	Button (1) – Configured as Mute Button
	Button (2) – Configured as On/Off Button

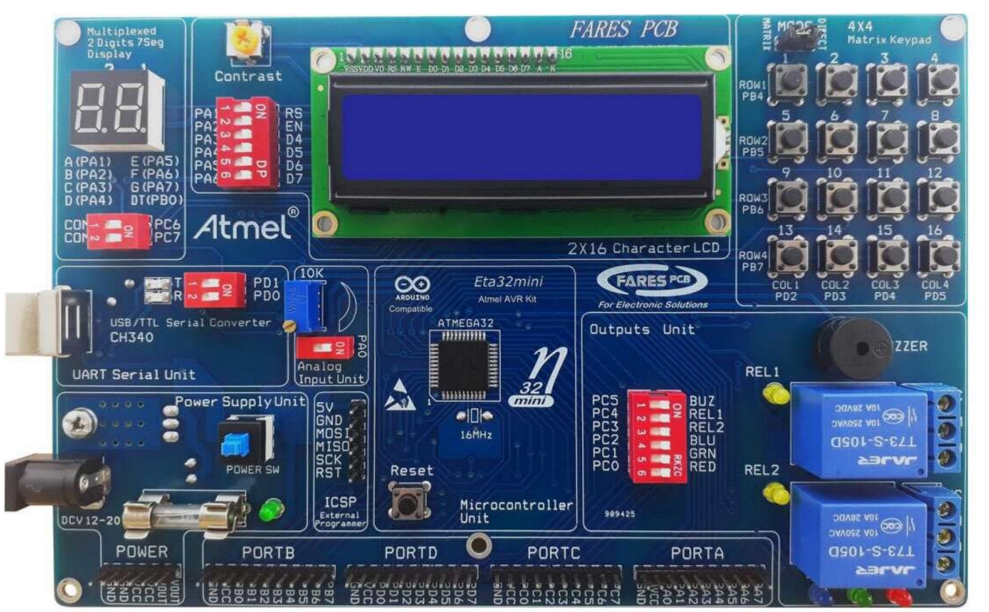


Figure 6: ETA32Mini Development Kit (Receiver Board)

Circuit wiring

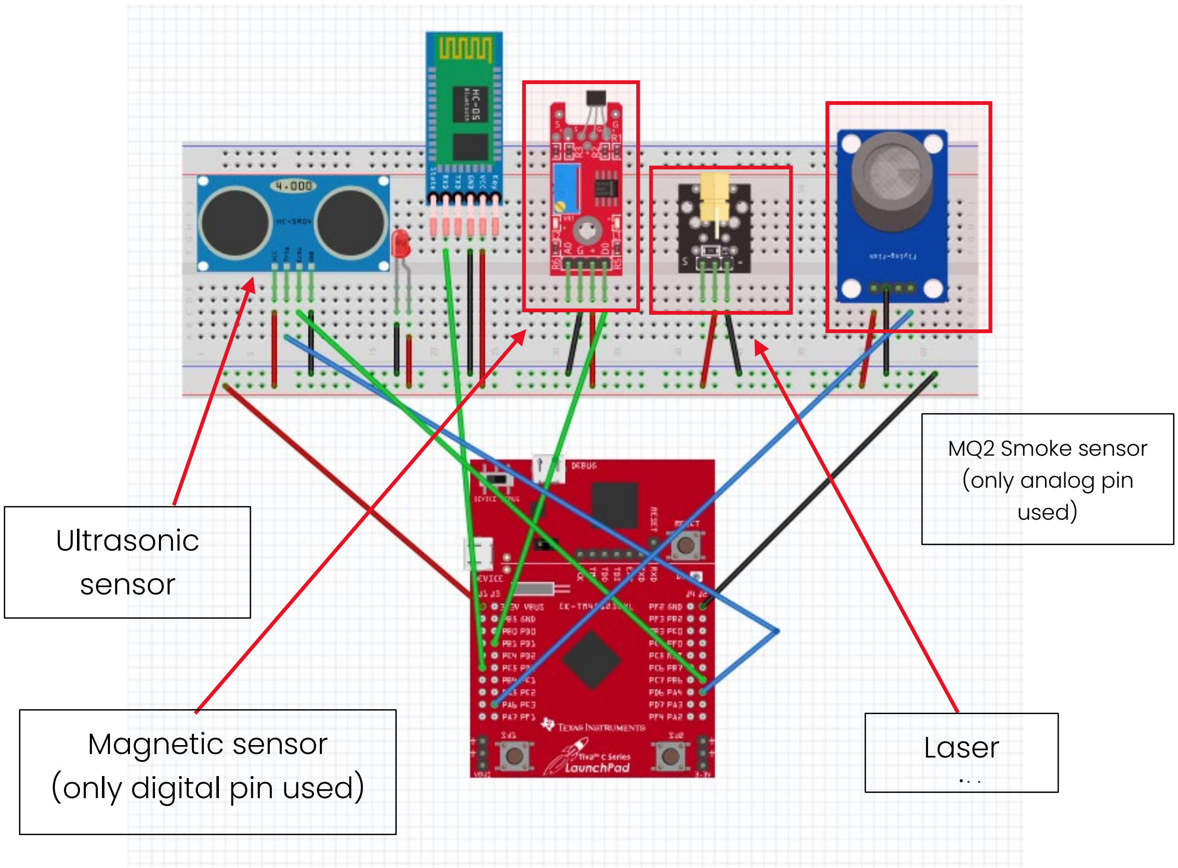


Figure 7: Tiva C Circuit Wirings (Control)

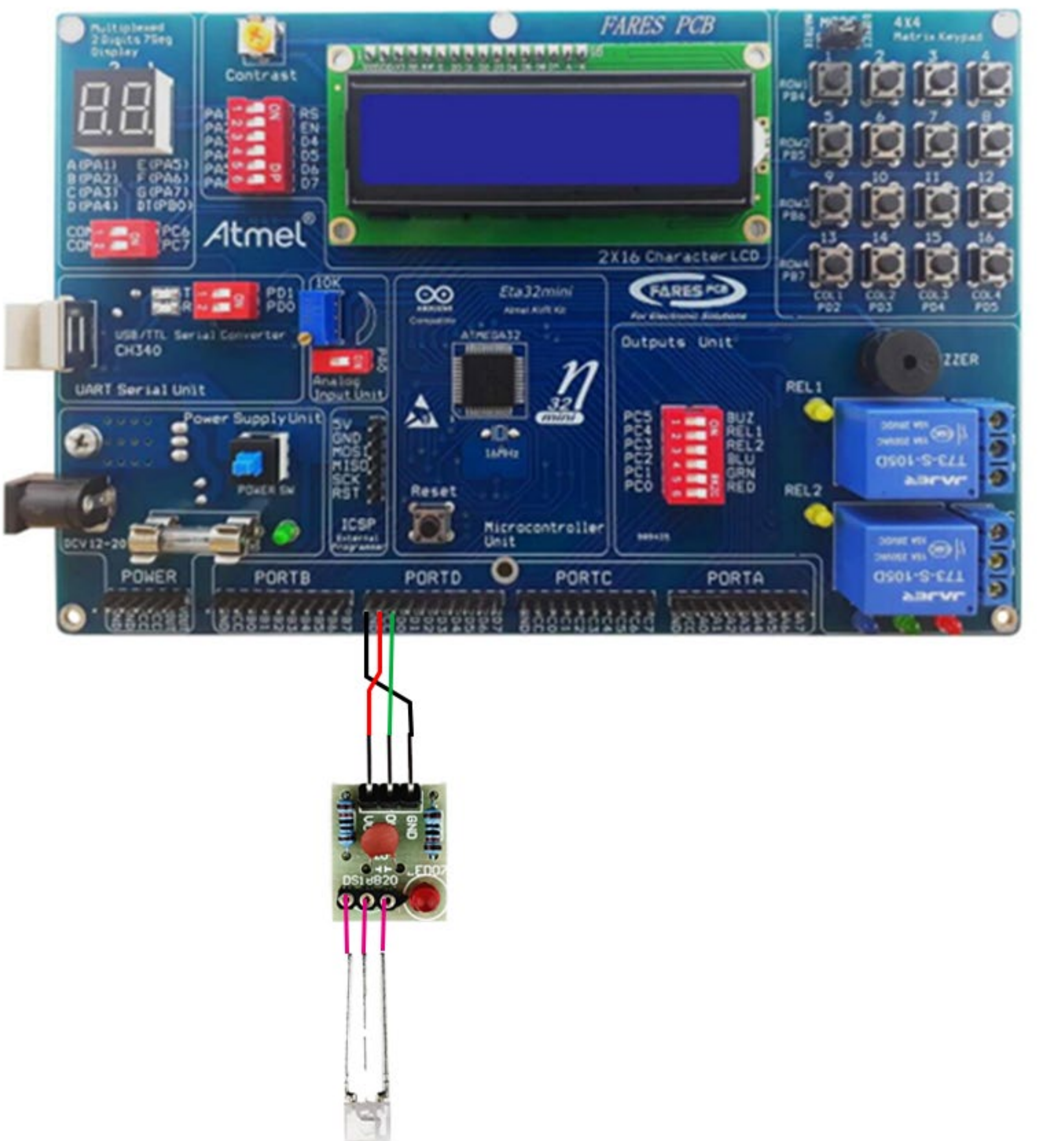


Figure 8: ETA32Mini Circuit Wirings (HMI)

Mobile App

For the mobile app development, we have used MIT app inventor.

App Development

App Inventor, developed by MIT, provides a visual and drag-and-drop interface for building Android applications. When working with Bluetooth communication in App Inventor, it primarily uses BluetoothClient and BluetoothServer components to establish communication between devices

1. Bluetooth Client Component:

The Bluetooth Client component is responsible for managing the client-side of a Bluetooth connection.

a. Connecting to a Bluetooth Device:

- Use the **BluetoothClient.Address** property to set the Bluetooth address of the device you want to connect to.
- Use the **BluetoothClient.Connect** method to establish a connection.

Here is an example of how we used them to show on the screen that the Bluetooth is connected or disconnected.

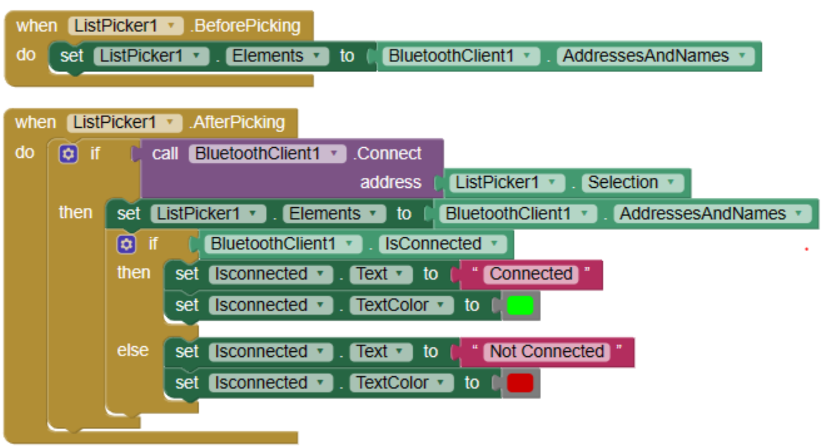


Figure 9: checking Bluetooth connection on MIT app inventor

b. Sending Data:

- Use the **BluetoothClient.SendText** method to send textual data.
  Ex: BluetoothClient.SendText("Hello, World!")
- Use the **BluetoothClient.SendBytes** method to send binary data (byte array).
  Ex: BluetoothClient.SendBytes("Hello, World!")

2. BluetoothServer Component:

The BluetoothServer component sets up a server to listen for incoming Bluetooth connections.

a. Starting the Server:

- Use the **BluetoothServer.StartListening** method to begin listening for incoming connections.

b. Receiving Data:

- The **BluetoothServer.AfterReading** event is triggered when data is received.

A side note: To be able to use the App correctly , you should first make request Bluetooth-related permissions in your App Inventor project as shown below:

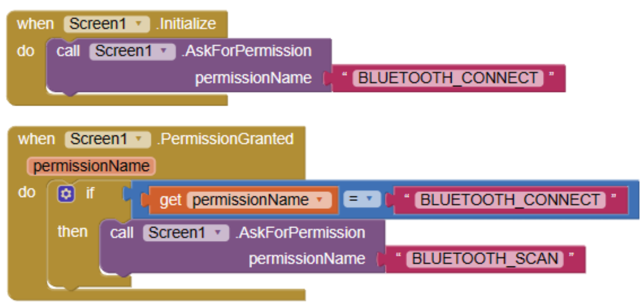


Figure 10: Bluetooth request permission in App inventor

Sending data from multiple sensors

To distinguish between which sensor has sent the data as to process it on the app side and show it to the user in its corresponding label , we send first a letter corresponding to each sensor (for example: M for magnetic) ; and in the app code , we have handled this by when detecting 'M' , take the data coming after it and show it on its corresponding label.

App Installation

To be able to run the app you just need the .apk file associated with this app and then to enable Installation from Unknown Sources:

1. On your Android device, go to "Settings."
2. Navigate to "Security" or "Privacy" settings.
3. Enable the option for "Install unknown apps" or "Install from unknown sources." This allows you to install apps from sources other than the Google Play Store.

4. Then, you will need to open the file manager on your Android device then locate the transferred APK file. After locating it, tap on the APK file to start the installation process. Lastly , Follow the on-screen instructions to complete the installation.

App Preview

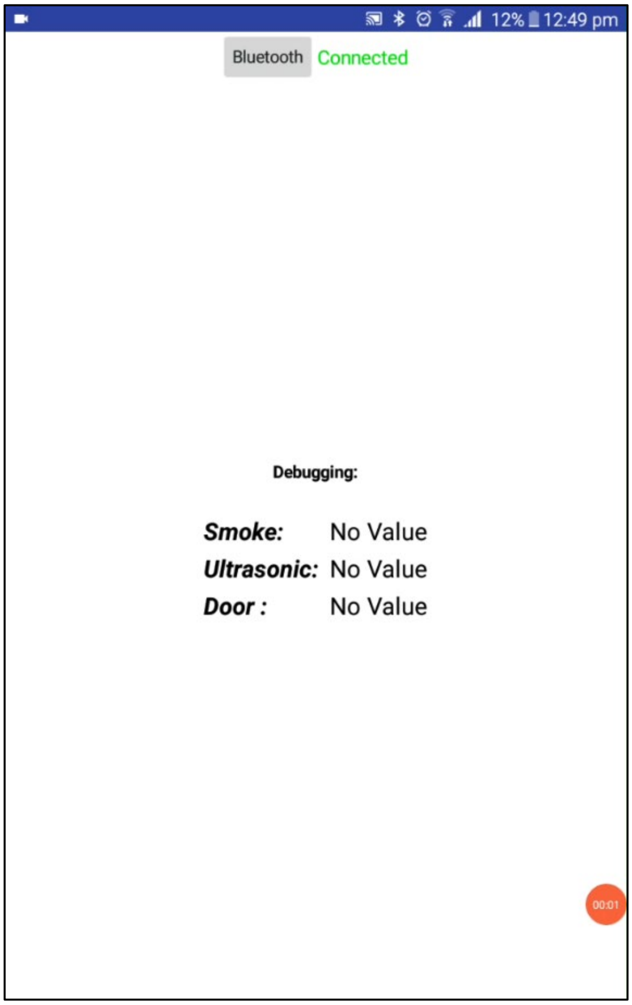


Figure 12: System Initialization

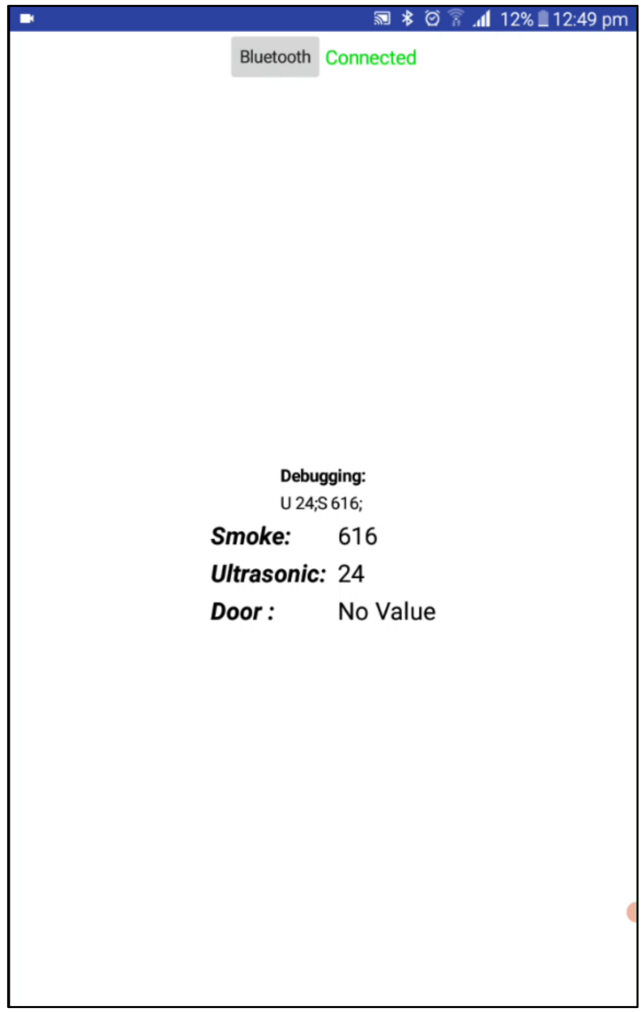


Figure 11: Continuous Readings of Smoke and Ultrasonic Sensors

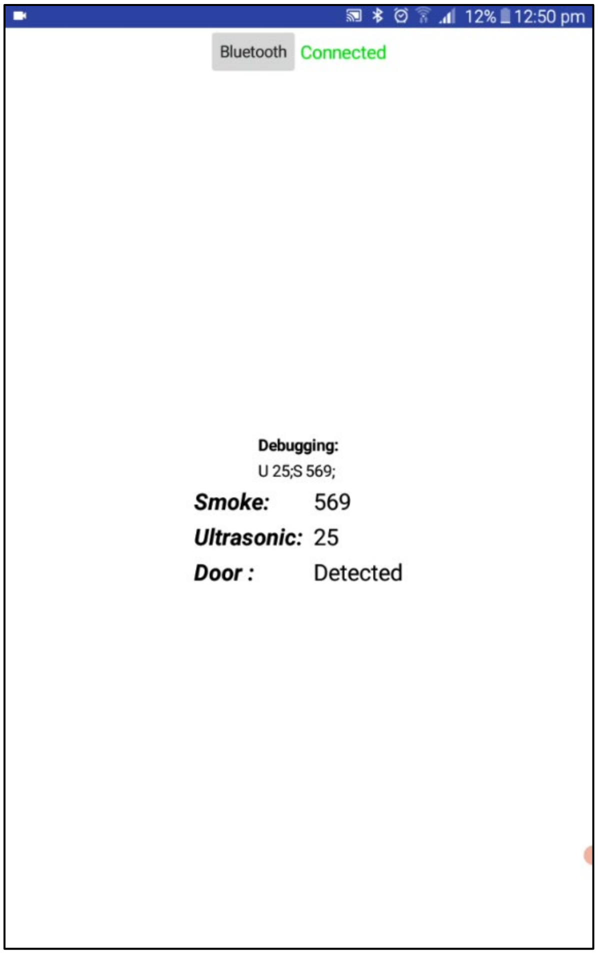


Figure 14: System has detected a magnetic sensor alert

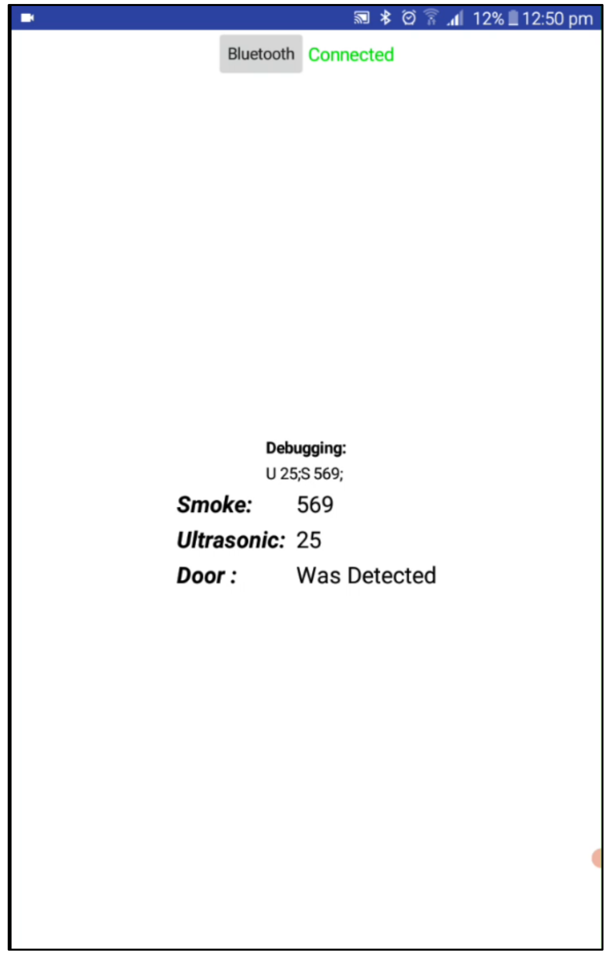


Figure 13: Magnetic sensor alert was muted

Obstacles

Li-Fi Receiver Hardware Selection

When implementing the idea of transmitting the UART data over light we encountered some issues with the receiving latency, as some modules couldn't withstand the speed of UART data transmission relative to its receiving latency. Our observations were as follows:

- Solar Panel Module: We have tried to receive light on a solar panel that would produce current relative to the speed of the light blinking speed, however, the solar panel was highly affected by the surrounding light and couldn't product 3.3v signals that would be readable on Tiva C GPIO pins.
- LDR Photoresistor: We have tried to implement an inhouse LDR configuration by using a power source and a photoresistor to receive the light signals. Although the module produced the required voltage, the module could not withstand the speed of UART transmission (9600 Hz) resulting in a constant (Logic High) reading.

Our solution was to implement the receiver using Laser Receiver Module as it possessed the required latency specifications that suits our implementation.

Light Source Selection

When transmitting light we faced the distance issue, as the intensity of light decreases with distances which restricts the installation of the system. We solved this issue by using a laser module to transmit laser signal for longer distances, however, due to the small size of the laser receiver module and the laser signal thickness, another problem raised which was the accuracy of installation.

In the demonstration of the project an LED was used rather than the laser module to demonstrate the issue.

Li-Fi Data Corruption

When transmitting data using the laser configuration, the data was somehow corrupted. Where the data sent with constant with constant input but with a shifted value. We have sent a sequence of numbers and displayed the received bytes on an LCD and the following pattern was observed:

- Odd numbers (1,3,5,...) produced the output(127, 126, 125, ...)
- Some even numbers were shifted to (63, 62, 61, ...)
- Some even numbers produced a different pattern

The relationship we concluded was as the follows, the number was transferred to 2's complement then shifted right till the first 1 is removed.

The reversed engineered solution was to shift left till the first 1 then perform a 2's complement. The solution was implemented as a function called at the sender to convert the data to be received correctly. In addition, the function was abstracted in the Laser driver so that the user would only be concerned with sending the correct message.

Bluetooth Reading Transmission

When sending the data to the Bluetooth module, we can send a message at a time, so the application can parse the message into displaying it, however, due to the large speed of the laser transmission we were capable of encoding the data by sending a header followed by the reading. This implementation enabled fast reading data transmission with Bluetooth. The messages looked like this:

- U 200;S 100;D;
- U 200 → Ultrasonic reading
- S 100 → Fume sensor reading
- D → Door opened message

Contribution

A group meeting was conducted to construct and assemble the Li-Fi communication devices, in addition, unit and integration testing of modules were implemented along the project in separate online meetings.

Multiple design decisions were discussed during tutorials with Eng. Mohamed Tarek.

Ahmed Wael Ibrahim Mohamed	- HMI Configurations and main program - Control (Tiva C) Main Program - DIO, UART, ADC drivers - Documentation & Presentation
Mustafa Usama Abdelrahman	- Data parsing and communication protocol of mobile app - Delay(SysTick), LED and ultrasonic drivers
Ahmed Sameh Mahmoud Nabieh	- Bluetooth driver configurations - Laser driver and data corruption fix - Hardware Assembly
Farah Ahmed AbdelRehim Tharwat	- Mobile application interface - Bluetooth data communication with application - MQ2 Smoke sensor driver
Salma Mohamed ElSoly	- Interrupt configurations - Startup code configurations and interrupt handlers - Magnetic sensor driver