**Summer Internship at BridgeThings IoT Pvt Lt**

**Project Report**

**By: K Senthur Kumaran**

During my summer internship at BridgeThings IoT Pvt. Ltd., I had the opportunity to immerse myself in cutting-edge technologies and gain practical experience in two domains: IoT Projects and Front-End Web Development for an inventory management system. This internship helped me to understand and learn real-world scenarios, develop new skills, and contribute meaningfully to ongoing projects within the company.

1. **Tank Level Sensor using ESP32 and XM125 Radar Sensor**

**1. Introduction**

**1.1 Background**

Monitoring the level of liquid in tanks is a critical task in various industries such as water treatment, oil and gas. Accurate and real-time tank level monitoring helps in efficient resource management and prevents overflows or dry runs. This project focuses on developing a tank-level sensor using the ESP32 microcontroller and the XM125 radar sensor.

**1.2 Purpose**

The purpose of this project is to design and implement a tank-level monitoring system that uses a radar-based approach to measure the distance to the liquid surface and calculate the tank level.

**1.3 Advantages of XM125 radar sensor**

The XM125 radar sensor offers high accuracy and precision with millimeter-level measurements, making it superior to ultrasonic and infrared sensors.

It remains reliable in challenging environmental conditions such as dust, fog, and varying light.

Additionally, it provides versatility for diverse applications and customizable settings for specific needs.

**2. Objectives**

* Integrate the XM125 radar sensor with the ESP32 microcontroller.
* Measure the distance from the sensor to the liquid surface in the tank.
* Calculate the tank level based on the measured distance.
* Display the tank level data in real-time using the Serial Monitor.

**3. Hardware and Software Required**

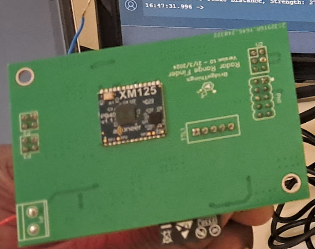
* Acconeer XM125 radar sensor
* ESP32 microcontroller
* Acconner Exploration Tool
* Arduino IDE
* SparkFun Qwiic XM125 Arduino Library

**4. System Design and Implementation**

**4.1 Flashing the XM125 Sensor**

Download the i2c\_distance\_detector.bin firmware file from the Acconeer website or the provided resources.

Connect the XM125 sensor to the computer using FT232RL USB to TTL Serial Adapter

Use the Acconeer Exploration Tool to flash the i2c\_distance\_detector.bin file into the XM125 sensor in DFU Bootloader mode. Ensure the flashing process is completed successfully

**4.2 Configuring the Arduino IDE**

To configure the Arduino IDE, first import the SparkFun Library by opening the Arduino IDE, managing libraries, searching for "SparkFun Qwiic XM125," and installing the library.

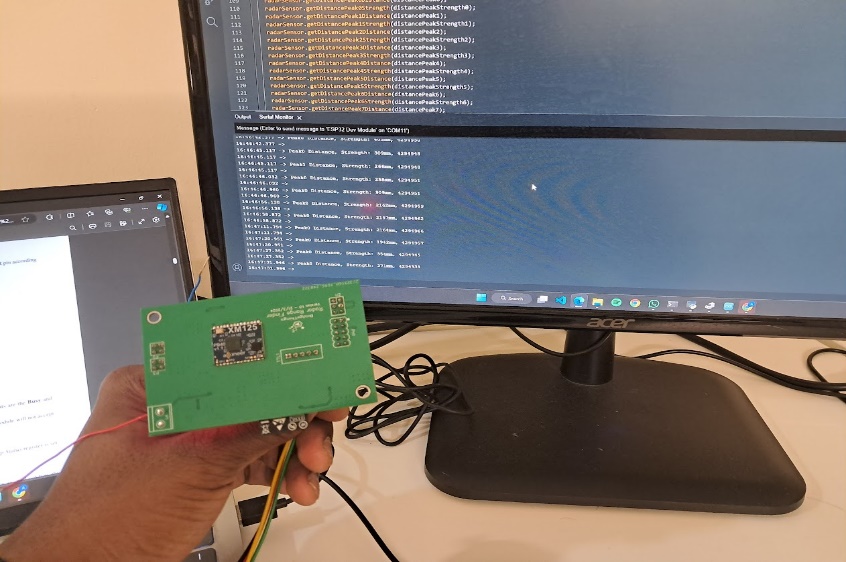
Next, import the Basic Distance Detector code by opening the Example1\_BasicDistanceDetector from the SparkFun Qwiic XM125 Arduino Library in the examples section.

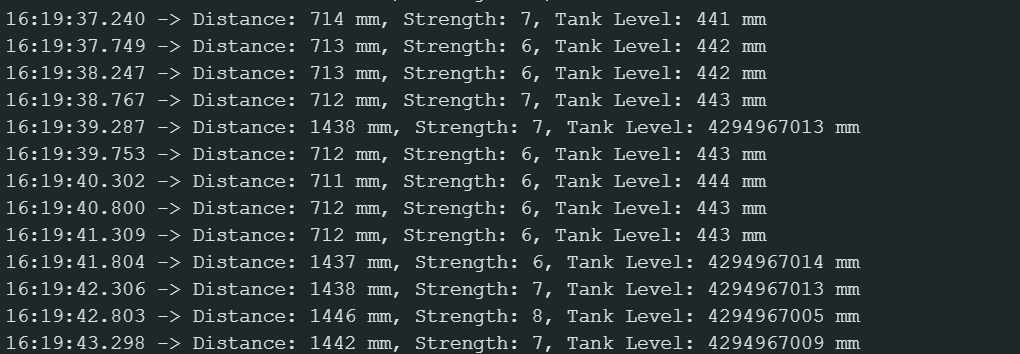
**4.3 Modifications in the Code**

1. **Modify Code as per Acconeer’s Specifications**:
   * Modify the example code based on the specifications in the Acconeer’s A121-Tank-Level-Reference-Application-User-Guide-5 document.
   * Ensure the following configurations:
     + Set the distance reflector shape to planar.
     + Set the peak sorting method to the strongest.
     + Set the presence start and end values based on the tank dimensions.
     + Configure the signal quality, threshold sensitivity, threshold method, max step length, and max profile according to the guidelines.
2. **Calculate and Display Tank Level**:
   * Initialize the tank height as a constant value.
   * Calculate the tank level by subtracting the detected distance from the tank height.
   * Display the tank level, detected distance, and signal strength on the Serial Monitor.

**5. Results**

The tank height, as physically measured, is 1155 mm, and the tank level physically measured is 712 mm. When measuring with the sensor, the tank level readings are 712 mm, 713 mm, 714 mm, 715 mm, and 711 mm. These readings are within the acceptable error margin.





**6. Challenges Faced**

However, there are occasional erroneous readings around 1400 mm. These readings occur due to internal reflections within the tank when exposed to open sunlight.

These erroneous readings are captured by the getDistancePeak7Distance method, which is designed to handle such cases.

The proper tank level measurements are obtained using the getDistancePeak1Distance method, which accurately reflects the actual tank level. The sensor's consistency in measurements close to the physically measured value confirms its reliability within the specified error margin.

Below is an example of the sensor output, demonstrating both the accurate readings and the erroneous reflections:

1. **Front-End Web Development for Inventory Management**

**1. Introduction**

* 1. **Background**

Efficient inventory management is vital for businesses to maintain optimal stock levels, reduce costs, and ensure smooth operations. Previously, the company managed inventory manually, which presented numerous challenges in tracking both incoming and outgoing stock. This manual process often resulted in difficulties locating items and determining the specifics of what was received and what was dispatched.

To address these issues, the development of an inventory management system became necessary. This system enables the detailed entry of stock specifications upon receipt and similarly tracks finished products as they are dispatched. This project focuses on the development of a front-end web application for the inventory management system, aimed at providing real-time tracking and insightful analytics to streamline inventory processes.

**1.2 Purpose**

The purpose of this project is to design and implement a user-friendly and efficient front-end for an inventory management system. The system aims to enhance the user experience, provide real-time data visualization, and facilitate easy management of inventory.

**1.3 Key Features**

* Detailed Data Entry: Allows for the entry of detailed specifications for various components, such as resistors, capacitors, and other electrical equipment needed in IoT boards.
* Stock Management: Includes functionalities to add new stock, view existing stock, and create new products from the existing stock.
* Order Management: Enables shipping orders, tracking order dates, and managing supplier information.
* Intuitive Dashboard: A comprehensive dashboard displaying key metrics, charts, and summaries of inventory status.
* Responsive Design: Ensures the application works seamlessly across various devices and screen sizes.
* Real-Time Data Updates: Provides real-time updates of inventory data to keep users informed of the latest stock levels and changes.
* Search and Filter Functions: Allows users to quickly find specific inventory items.

**2. Objectives**

* Design and develop a responsive user interface for the inventory management system.
* Utilize local storage for data storage and retrieval.
* Implement real-time data visualization and updates.
* Enhance user experience through intuitive design and easy navigation

**3. Technologies Used**

* HTML, CSS, JavaScript: Core web technologies for building the front-end.
* Bootstrap: Framework for responsive design.
* Chart.js: Library for data visualization.
* Local Storage: For storing and retrieving data within the browser.

**4. System Design and Implementation**

**4.1 User Interface Design**

* **Dashboard**: Designed a comprehensive dashboard displaying key metrics, charts, and summaries of inventory status using HTML, CSS, and JavaScript. Ensured the dashboard is intuitive and easy to navigate.
* **Dashboard HTML Structure**: The dashboard (dashboard.html) includes sections for total inventory items, total stock value, orders processed, and most used components, each represented with cards. Additionally, it includes charts for inventory status, monthly stock changes, and component usage to visualize data.

**4.2 Responsive Design**

* Utilized Bootstrap to ensure the application is responsive and works seamlessly across different devices and screen sizes. This involved creating flexible grid layouts and responsive components.

**4.3 Data Storage and Retrieval**

* Utilized local storage for storing and retrieving data within the browser, avoiding the need for server-side storage and RESTful APIs.
* **Index Page Integration**: The index.js file handles the loading of recent activities and statistics from local storage and dynamically updates the HTML content on the index page (index.html). This includes functions to load recent activities, statistics, and initialize data in local storage for demonstration purposes.

**4.4 Data Visualization**

* Implemented data visualization using Chart.js to create interactive charts and graphs displaying inventory trends, stock levels, and other key metrics.
* **Dashboard.js**: This JavaScript file is responsible for creating and managing the charts on the dashboard page using Chart.js. It includes functions to load chart data and render them on the page.

**4.5 Real-Time Data Updates**

* Developed functionality to provide real-time updates of inventory data, keeping users informed of the latest stock levels and changes using local storage.
* **Purchase Orders Management**: The purchase-orders.js file handles loading, adding, and deleting purchase orders, as well as exporting purchase order data to Excel. It interacts with local storage to manage the data and dynamically updates the HTML content in the purchase orders page (purchase-orders.html).

**4.6 Detailed Inventory Management**

* **Detailed Specifications Entry**: Implemented features to enter detailed specifications for various components, such as resistors, capacitors, and other electrical equipment. For example, for resistors, users can enter resistance, power rating, voltage rating, and other specifications separately. This allows for precise tracking and management of each component.
  + **New Products Page (new-products.html)**: Users can add new products to the Bill of Materials (BOM) by selecting the component type and entering detailed specifications. This includes attributes such as resistance, power rating, voltage rating for resistors, and similar attributes for other components like capacitors, inductors, diodes, transistors, etc. The corresponding JavaScript file new-products.js manages the dynamic addition of these components to the stock.
* **Adding and Viewing Stock**: Functions to add new stock items, view existing stock, and manage inventory with detailed specifications were implemented.
  + **View Stock Page (view-stock.html)**: Allows users to view and manage the current inventory, displaying detailed specifications of each item. The view-stock.js file handles the display and management of stock items using data from local storage.
* **Creating New Products**: Allowed the creation of new products from existing stock, including specifying detailed attributes for each component.
  + **New Products Management**: Users can combine existing components to create new products, ensuring that all attributes and specifications are accurately tracked.
* **Order and Supplier Management**: Included features for creating and managing shipping orders, tracking order dates, and managing supplier information.
  + **Order Information Page (order-information.html)**: Users can enter order details, including BOM name, address, city, postal code, phone number, and country. The entered data is stored in local storage for order tracking.
  + **Suppliers Page (suppliers.html)**: Manages supplier data, allowing users to add, view, and delete supplier information. The suppliers.js file manages supplier data, including loading, adding, and deleting suppliers, as well as importing and exporting supplier data. It dynamically updates the HTML content in the suppliers page.
  + **Order Summary Page (order-summary.html)**: Provides a summary of orders, including detailed information on each order, grouped by component type. The order-summary.js file handles the display of order summaries and allows users to minimize or delete orders as needed.

Top of Form

Bottom of Form

**5. Challenges Faced**

I faced several challenges during the development of the inventory management system:

* **Implementing Data Tables in Local Storage**: Maintaining the design and format of the data tables after refreshing the page was a significant challenge. The design and layout of the tables would often change upon reloading, making it difficult to ensure a consistent user experience.
* **Lack of Backend Implementation**: I was unable to implement a backend for the system. This limitation meant that I had to rely entirely on local storage for data management, which posed its own set of challenges, including data persistence and security.
* **Feature Setup Difficulties**: Setting up various features such as detailed data entry, stock management, and order tracking was challenging. Ensuring that all these features worked seamlessly together required careful planning and troubleshooting.
* **Analyzing Component Specifications**: Understanding and analyzing the specifications for various components was challenging. Different components such as resistors, capacitors, and inductors have unique specifications that needed to be accounted for. Designing the system to handle these varied specifications required a detailed and thoughtful approach.
* **Designing Detailed Specifications**: Each component had its own set of specifications, and designing the system to accommodate all these details was complex. I had to ensure that the system could handle the detailed entry and management of attributes like resistance, power rating, voltage rating for resistors, and similar specifications for other components. This required creating flexible and robust data structures and user interfaces to support all necessary attributes.

1. **Attended GreenCo 2024 Summit on Water and Resource Management:**

During my internship, I got an opportunity with the BridgeThings team to attend the GreenCo 2024 Summit on Water and Resource Management at Chennai held on 26th and 27th June 2024, hosted by the Confederation of Indian Industry.

#### **GreenCo in CII**

GreenCo, an initiative by the Confederation of Indian Industry (CII), is a comprehensive framework designed to encourage and evaluate sustainability practices within industries. The GreenCo rating system assesses various parameters including energy efficiency, water conservation, waste management, renewable energy use, and the implementation of green supply chains. It aims to help companies integrate sustainable practices into their operations, thereby reducing their environmental impact and promoting long-term ecological balance.

#### **Concept of Net Zero**

Net Zero refers to achieving a balance between the amount of greenhouse gases emitted into the atmosphere and the amount removed. This is accomplished by reducing emissions as much as possible and offsetting any remaining emissions through natural processes like reforestation or technological solutions such as carbon capture and storage. The primary goal of Net Zero is to significantly reduce carbon emissions and mitigate climate change.

#### **Principles of Sustainability**

Sustainability involves meeting the needs of the present without compromising the ability of future generations to meet their own needs. It encompasses a broad range of practices aimed at preserving natural resources, protecting ecosystems, and ensuring a high quality of life for all people. Sustainable practices include responsible resource management, waste reduction, and promoting social and economic well-being.

#### **Importance of Working Towards Sustainability and Net Zero**

Working towards sustainability and Net Zero is crucial for several reasons:

* **Environmental Protection**: Reducing emissions and conserving resources help protect natural ecosystems and biodiversity.
* **Climate Change Mitigation**: Achieving Net Zero is vital for limiting global warming and preventing the severe impacts of climate change.
* **Economic Benefits**: Sustainable practices can lead to cost savings through increased efficiency and innovation.
* **Social Responsibility**: Companies that prioritize sustainability demonstrate their commitment to social and environmental responsibility, which can enhance their reputation and stakeholder trust.

#### **Showcase of Skills and Products by BridgeThings IoT Pvt. Ltd.**

#### At the GreenCo 2024 Summit, BridgeThings IoT Pvt. Ltd., the company where I interned, showcased their innovative products and skills. The company demonstrated its commitment to sustainability through its IoT solutions, which optimize resource usage and improve operational efficiency. Products such as smart sensors and automation systems for energy management and environmental monitoring were highlighted, showcasing the company's expertise in integrating IoT technology with sustainable practices.

#### also started to get a better understanding of the company’s projects and aims. By listening to interactions between the CEO and business customers, I learned how he described the products and engaged with clients. Additionally, I had the opportunity to interact with a customer and explain some aspects of the products, using knowledge gained from conversations with the CEO and other employees.

#### **Insights and Learnings from the Summit**

Attending the GreenCo 2024 Summit provided me with valuable insights into how different companies, especially startups, approach design and innovation with a focus on sustainability and Net Zero. I learned about the importance of working towards sustainability and the various strategies companies employ to achieve these goals. The summit also offered a platform to understand how different companies operate, their unique product ideas, and their contributions to a sustainable future.

#### **Conference and Industrial Insights**

During the summit, I attended various conferences where industrial experts and professors discussed the latest trends and challenges in sustainability. These sessions provided a deep understanding of the importance of sustainable practices and the role of technology in achieving environmental goals. I gained insights into the innovative approaches adopted by industries to minimize their environmental footprint and the significance of collaboration between academia and industry in driving sustainable development.

#### **Learning from Startups and Industry Leaders**

Another key learning from the summit was understanding how different companies, especially startups, come up with their designs and product ideas. I observed the importance of integrating sustainability and Net Zero goals into business models from the ground up. It was insightful to see how these companies function, their innovative product ideas, and their commitment to sustainable development. This experience underscored the necessity of working towards sustainability not just as a regulatory requirement but as a core aspect of business strategy.