Smart parking system

dmh1g19@soton.ac.uk — dmh1g19 — 31138039

March 2024

Contents

Abstract

In the wake of an increase in urbanization and 15 minute cities, this project provides an IoT low power wide area network solution to improving parking routes in parking facilities - smart parking. This report introduces a smart parking solution that leverages Long Range Wide Area Network (LoRaWAN) technology to address these challenges by providing real-time parking availability data, thereby facilitating efficient parking space management and improving the overall parking experience.

1 Introduction

Urban centers around the globe are increasingly faced with the challenge of accommodating the ever-growing number of vehicles within their confines, leading to congestion and inefficiencies in parking management. This leads to congestion and increased emissions, creating a time and financial overhead to day to day activities and events in the local area.

The aim of the technology is to improve the efficiency of parking tasks in urban areas. Moreover, the environmental impact of this search for parking cannot be overlooked, as it leads to unnecessary carbon emissions and reduced air quality.

The proposed solution is centered around deploying LoRaWAN-enabled parking sensors that detect the occupancy status of each parking spot. These sensors, characterized by their long battery life and wide coverage area, transmit data to a central system via LoRaWAN gateways. The data is processed on site through a small computer device before being sent to the TNN cloud and accessed by the end node app server.

The end user is able to view the current state of the parking space and make an informed decision, helping to lower congestion.

1.1 Background

1.2 Problem statement

There's increasing demand for parking spaces amidst growing vehicle populations. This surge contributes significantly to urban traffic congestion, with drivers spending considerable time searching for available parking spots[1]. This leads to increased fuel consumption, elevated carbon emissions, and reduced air quality. Moreover, the lack of real-time parking information and inefficient use of existing parking infrastructure exacerbates these issues, resulting in frustration for drivers.

1.3 Scope

The implementation of the system will focus on the use of networking technology more so than scalable implementations which are highly dynamic. The main goal of the system is to explore the use of different networking technologies with a focus on the use of bluetooth and LoRa technologies for low power wide area solutions.

2 Design

2.1 UML abstraction

The following UML diagram presents an overview of the different layers of abstraction of the smart parking system.

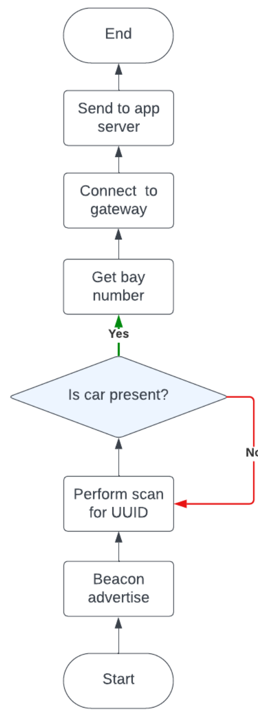

Figure 1: System abstraction

.

3 Implementation

3.1 Hardware

3.1.1 Aadafruit feather nrf52840 express

A powerful and versatile microcontroller that is part of the system's sensor nodes. Equipped with Bluetooth Low Energy (BLE) and built-in USB support, integrated in the parking environment, in this case as occupancy sensors installed in each parking spot. Occupancy data is transmitted in real time to any BLE receiver.

In this project the board is set up to follow the popular 'iBeacon' format from apple[1]. This is a very popular format used a lot in retail stores to manage products for different purposes. The reason for this decision is because the iBeacon format is designed for proximity sensing, low energy transmittal and easy of implementation by focusing on homogenising heterogeneous devices.

Each beacon is given a name and a UUID (universally identifiable ID) which can be used to differentiate them from background noise.

[1] https://developer.apple.com/ibeacon/

E.g of a UUD:

```
uint8_t beaconUuid[16] = {
    0xE2, 0xC5, 0x6D, 0xB5, 0xDF, 0xFB, 0x48, 0xD2,
    0xB0, 0x60, 0xD0, 0xF5, 0xA7, 0x10, 0x96, 0xE0
};
```

Each beacon also gets a 'major' and 'minor' value for clustering beacons within a certain area/location. For example all beacons in the system can share the same UUID (all beacons belong to the same parking facility) but perhaps there are different floors - these floors can be differentiated through these additional parameters.

E.g of major and minor values:

```
uint16_t major = 100; // Example major
uint16_t minor = 1;   // Example minor
```

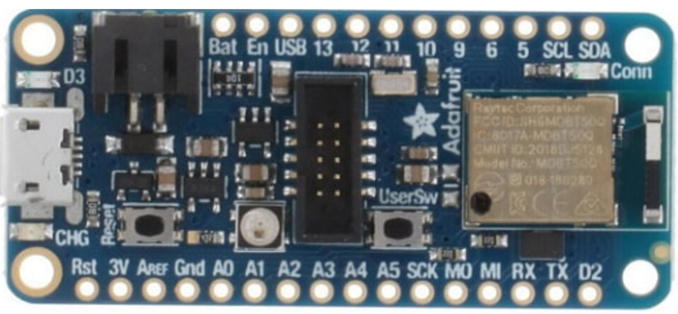

Figure 2: nRF52840

.

3.1.2 RPi Zero W 2

A small processing unit with a full operating system that acts as a compact, cost-effective gateway or central hub in the smart parking system. It collects data transmitted over BLE from nearby Adafruit Feather nRF52840 Express devices.

In this project the RPi is used to sniff the surrounding BLE traffic and identify parking spots through their UUID, major and minor parameters. The RPi is great cost effective solution for this as it's more than powerful enough to perform simple data anylysis and parsing before sending cleaner data to a better long range broadcasting solution. The main purpose of this component is to decode the data transmitted by the feather ready for transmittal through LoRaWAN. A single RPi Zero 2 W is used to collect broadcasts from all feather beacons.

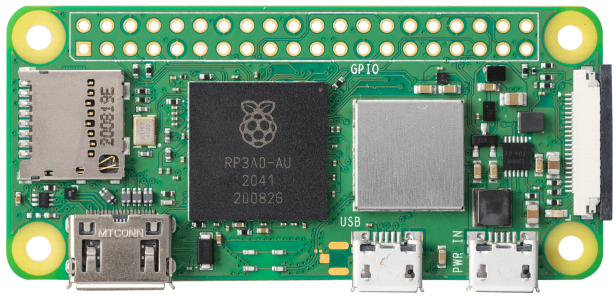

Figure 3: RPi zero 2 W

.

3.1.3 Arduino MKR WAN 1300

Utilized for its LoRaWAN communication capabilities, offering a long-range, low-power solution for transmitting parking data to the LoRa gateway for access through the cloud. In this case the MKR WAN 1300 is connected to the RPi via the mini USB serial port.

Seeing as only RPi is used for collecting all beacon broadcasts, any broadcast is passed to the MKR WAN 1300 through its serial port. The received data is then sent to the TTL. The app servers such as a smartphone, laptop or computer can access a visual application showing the current availability of the parking spots for that location.

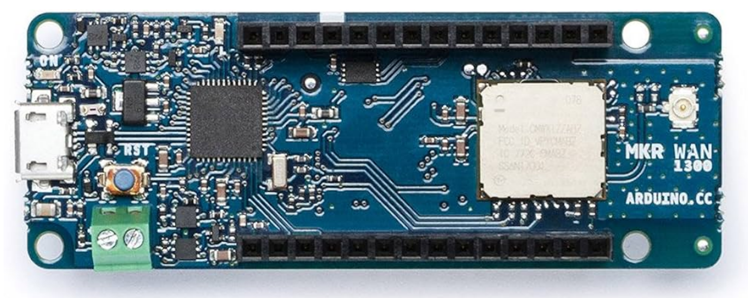

Figure 4: mkr

.

3.1.4 End server

A visual representation of each parking spot and their availability (green = available, red = unavailable). The app sever receives the beacon broadcast through the use MQTT before rendeirng a graphical web display.

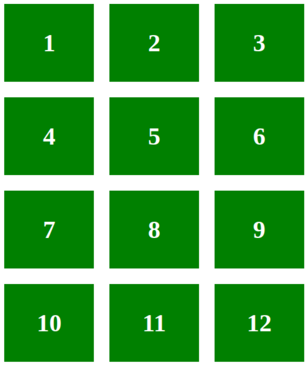

Figure 5: Visual app showing all parking spots available

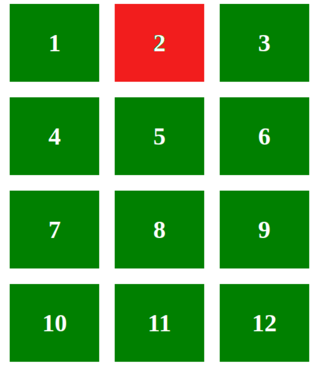

Figure 6: Visual app showing bay 2 in use

3.1.5 Data format

The following payload is sent over to the RPi for processing:

```
transmission = {
        "name": /* Name of the bay */,
        "db": /* Strength of the signal */,
        "UUID": /* The unique identifier of the beacon */
    }
```

The followingpay load is sent over loRaWAN:

$$\text{Final payload sent over LoRaWAN } p : \begin{cases} \text{If car is present} & 0, \\ \text{If car is not present} & 1. \end{cases}$$

3.2 Software

3.2.1 C++

C++ is used to program the low level micro-controllers such as the feather and the Arduino MKR WAN 1300.

3.2.2 Python

Python is used on the RPi Zero 2 W and the app server for managing data such as using the MQTT library.

4 Results

4.1 Implementation details

For simplicity the key metric to determine if a car in present in a bay is through a decrease in the signal strength given a certain threshold. The signal is blocked by the cars mass, if the signal is beyond a threshold - it is assumed a car is parked in the bay.

4.2 Results

The final networking system works as intended and successfully detects object proximity and broadcasts its beacon data and updates the final display for viewing. This is a cost effective and simple solution that can be expanded upon if needed.

The system's capability to discern the presence of a car based on signal attenuation provides a reliable metric for parking space occupancy without the need for direct visual confirmation or physical contact sensors.

This project shows that BLE and LoRaWAN networking is a viable solution to smart parking problems.

4.3 Further work

This project can be expanded upon by building on what's already available. The existing hardware can be expanded upon by including a fourth component in the car itself to receive a more reliable detection method. Another component that can be used to improve the detection of the cars in the parking bays could be image sensors in each parking bay, but this increases complexity and costs.

References

[1] Donald Shoup. "The High Cost of Free Parking". In: *Journal of Planning Education and Research* 17 (Jan. 1997), pp. 3–20.