The University Of Sheffield.

Department of Electronic and Electrical Engineering

Using Near Field Communication (NFC) to enhance an automated low cost device for the remote monitoring of Glucose for type-1 Diabetics using mobile phones

Electronic and Electrical Engineering, MEng

Student: Ben Trevett

Registration No.: 110231253

Word Count: 1084

Date: 28/08/2015

Supervisor: Dr. Mohammed Beniassa

Introduction

The project I carried out for the Sheffield Undergraduate Research Experience (SURE) was an enhancement on the project I completed as my individual research project towards the third year of my MEng degree in Digital Electronics. The project was the research and development of a device that would aid diabetics by reading data from their glucose measuring device (known as a glucometer) and transfer the data via Bluetooth to a mobile phone application that would display their data, with the ability to upload it to a webserver where the patient's medical supervisor can view it. My goal for the duration of the SURE scheme was to improve the project by using ideas originally conceived during the research project, but were unable to implement due to time constraints. The main enhancement I wanted to add was making use of near field communication (NFC) technology. NFC is a short range data transfer technology most commonly used in contactless payment systems. I believed that the addition of NFC would make the device easier to use for patients who were unfamiliar with Bluetooth technology, due to NFC's ease of use. The project had 3 main implementations of NFC:

1. Using NFC tags to store data
2. Using NFC for data transfer
3. Using NFC to enable Bluetooth communication

Project Summary

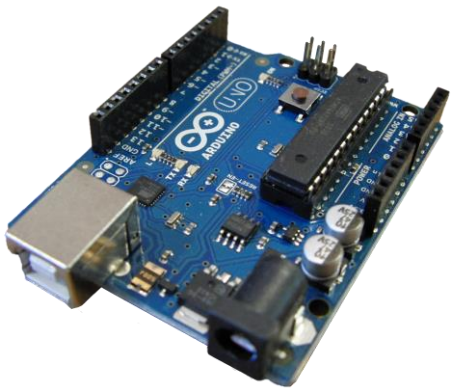

Figure 1 – Arduino Uno

The project made use of an Arduino Uno microcontroller. This was chosen due to them being open source, relatively low-cost (around £6) and having a wide variety of functionality available through the use of 'shields'. Shields are boards that can be connected on top of the Arduino and two shields used for this project were a USB host shield and an NFC shield. The USB host shield allows for data to be transferred between devices connected to the USB port, in this case reading data from the glucometer. The data is then processed on the Arduino board. The NFC shield allows data to be transferred over NFC, allowing data to flow between Arduino and NFC enabled devices, such as smartphones.

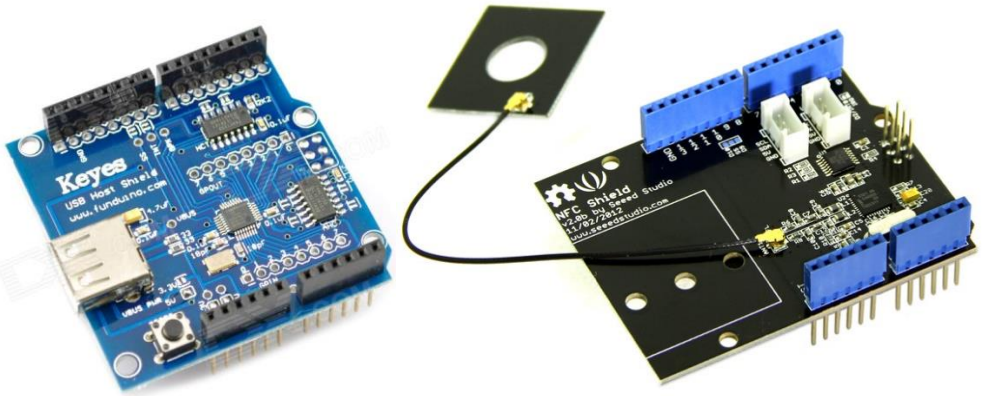

Figure 2 – USB host shield (left) and NFC shield (right)

NFC is a method of wireless communication, similar to Bluetooth. However it has several major differences. In terms of data transfer speed, NFC is much lower than Bluetooth, around 500Kbits per second compared to Bluetooth's 3Mbits. NFC also has a much shorter range, around 5cm, compared to the 10's of metres offered by Bluetooth. NFC is also 'low friction', meaning it is easy to set up a connection between devices, there is no discovery or pairing required, whereas Bluetooth is 'high friction', requiring around 10 seconds for discovery and a user-interface for pairing with a device. NFC devices are also passive, meaning the receiving end requires no power as it is powered by the sending side [1] [2].

The reason I decided to use NFC during the SURE project was due to NFC's ease-of-use over Bluetooth. NFC data transfer is initiated when the device is in close proximity with an NFC enabled phone, the phone will ask the user to confirm the connection and then automatically open the application and begin the data transfer. This is much simpler than Bluetooth's process of scanning, pairing and connecting. The short range also acts as a security feature, meaning that it will only connect to devices in close proximity, which will generally done by the user and not an outsider with malicious intent.

NFC data communication is done via an NFC Data Exchange Format (NDEF). NDEF messages are formed of multiple 'records', which are essentially individual messages within the overall NDEF message [3]. Data is sent from the Arduino via the NFC shield and has two methods of transmission: data sent in a single message or data is sent in a continuous stream (peer-to-peer).

NDEF receiving was done in this project via NFC tags (single message) and an Android phone (peer-to-peer). NFC tags are passive (non-powered) devices that come a variety of shapes and sizes and effectively store data.

The first task, using NFC tags to store data, was completed by writing a program for the Arduino microcontroller, called a 'sketch'. The sketch makes use of the NFC 'library'. 'Libraries' are sections of pre-written code that can be used to easily implement complex functions, such as writing to an NFC tag. An example of such code:

```
NfcAdapter nfc = NfcAdapter(interface);          //declaring the adapter as an interface
NdefMessage msg = NdefMessage();                 //declaring the message
message.addTextRecord("Hello World!");           //adding text to the message
nfc.write(message)                               //writing the message
```

This is done in a similar approach for peer-to-peer connections. This is handled on the Android side by reading the NDEF message and writing it to the screen [4].

The process for using NFC to enable Bluetooth is done by configuring the NDEF record in such a way that specifies the Universally Unique Identifier (UUID) for the Bluetooth chip on the phone and can be automatically created using an Excel form produced by Texas Instruments [5]. During this method, when the phone is held close the NFC shield, an alert appears to the user asking to confirm if they want to connect via Bluetooth.

Conclusion

The first 2 tasks were completed successfully. The application to receive the results was made, however it only has basic functionality of reading and displaying the data in a simple list. If more time were permitted, the application would be updated to include a simple user-interface.

The third task proved to be more difficult, and even though would sometimes successfully connect, other times it would inexplicably not. I was unable to diagnose the problem within the time.

This research will can be used in further in remote monitoring of glucose levels by acting as a proof of concept for NFC enabled glucometers. These glucometers are attached to the patient's body as a 'patch', and readings could be periodically taken from them simply by placing the phone close to the patch.

Arduino code for the project can be found at: https://github.com/bentrevett/glucoduino-nfc

Android application can be found at: https://github.com/bentrevett/glucoduino-nfc-application

Initial project report can be found at: https://bentrevett.github.io/glucoduino.pdf

References

[1] [Online]. Available: https://www.youtube.com/watch?v=49L7z3rxz4Q.

[2] [Online]. Available: https://www.youtube.com/watch?v=HkzPc8ZvCco.

[3] [Online]. Available: http://developer.android.com/reference/android/nfc/NdefRecord.html.

[4] [Online]. Available: http://developer.android.com/reference/android/nfc/package-summary.html.

[5] [Online]. Available: http://www.ti.com/lit/zip/sloa187.