Data Acquisition on mobile devices using non-standard transfer methods

Client: Fred Lionetti

Director of Software Department
Decagon Devices, Inc.

Joran Beasley
Computer Science Undergraduate
University of Idaho

Abstract

The emerging ubiquity of smartphone devices is changing the way in which we interact with computers. Thousands of existing devices, using the standard RS-232 communication protocol (from weather sensors to refrigerators,) are unable to communicate with newer smartphones. iOS and Android complicate matters by making platform independent solutions even more challenging. Thus the investigation of a robust communications methodology and protocol would prove both useful and valuable. Ideally this solution would not require extensive setup or jail-breaking on the part of the end user.
**Problem Statement**

RS-232 is a communication standard used by many peripherals that allows communications over a serial port. However there is no standardized, platform-independent method for communicating between these devices and smartphones. While some research has proceeded in this area, no two-way cross-platform solutions have emerged. We are interested in providing a standardized, platform independent two-way communication interface between RS-232 devices and mobile computing devices.

**Project Requirements**

- Determine a method for transfer.
- Determine a message passing protocol that accepts RS-232 as client input and outputs RS-232 as host output.
- Create a proof of concept that works on a laptop with the same features as a mobile computing device (i.e. Wi-Fi, micro-USB, etc.), no need to worry about OS specific implementations at this point
- Define an API for use with the new data communication interface.
- Implement the API in a way that it is cross-platform
- Create prototype mobile application that demonstrates proof of concept while remaining cross-platform

**Determine a transfer method**

There are several potential transfer methods which require further investigation:

1. Bluetooth
2. Device specific wires (for iOS devices, use their proprietary data cable; for Android devices, use micro-USB, etc.)
3. Micro-USB and use an adapter for the iPhone
4. Ad-hoc wireless data network in which the mobile device synchronizes to the sensor over the ad-hoc network
5. Converting microphone/audio jack into a serial modem by demodulating the electrical pulses

Performance and cost must also be taken into account when evaluating transfer methods. For example, if using the microphone/audio is the least expensive method, it may still not be usable for many applications if it can only achieve a 1 kilobit per second transfer rate.
Determine Message Passing Protocol

Develop a protocol to wrap the RS-232 signal in a packet, encoding information about the message (message size, start/stop bits, setting baud rates and polarity, etc.). Since RS-232 is our primary protocol, this packet would not need much information while focusing the design effort on simplicity.

Create a proof of concept

Create an alpha quality prototype. This stage ensures we are not pursuing an impossible concept. It must be demonstrated that the protocol developed sends data and receives data through the transfer method chosen.

Defining the API

Using best practices, design an API to closely mirror serial communication APIs that are already in existence. Implementation of the API would likely be in the form of a C/C++ shared library object to ease the transition to cross-platform implementations.

Build Library

Implement API in a cross-compiled library to be used on Android and iOS.

Implementation using library

Implement an application, on both iOS and Android, demonstrating the message passing protocol developed earlier.

Results

If accepted, the initial phase will be further project requirements gathering, as well as define some test cases that will need to pass in order to consider the project a success. This will result in a communication library facilitating communication between traditional RS-232 devices and mobile computing devices. When implemented, this library will be used in many projects across multiple industries sold into many markets.

The students involved will have worked with advanced data communication protocols and developed for multiple mobile platforms. The team members will have developed the hands on experience of building easily transported libraries while tackling the challenges of designing an application that runs in complex, and diverse environments.

The Customer

Decagon Devices Inc. is a local business that manufactures a large variety of agricultural and food science sensors and devices. However, this technology would be applicable to large segments of many industries.