Cheap, accurate, and privacy-preserving contact tracing chip

Team 64 - Kapil Kanwar, Anshul Sanamvenkata, Abhinav Singh

Introduction

Objective

COVID is a deadly and highly infectious disease, and given the current trend of globalization and environmental destruction, such pandemics will only become more common. Testing and contact tracing are one of the best ways to fight a highly infectious disease while allowing people to maintain some semblance of a normal life.

We propose a small, cheap chip that can be easily carried which will automatically communicate with other nearby chips over ultra-wideband (UWB) to perform contact tracing, detecting potential transmissions of up to 10 feet away (adjustable depending on the nature of the pandemic it is being used for). Additionally, it must be regularly docked with a PC with Internet access to charge and upload contact information to a server. While there are existing solutions that use smartphones, these solutions are less than ideal for reasons that will be outlined below.

Background

Current contact tracing solutions either rely on manual effort, or mobile apps, which are both flawed. Manual methods typically involve calling someone who has tested positive and asking them to recall whom they met, which obviously is highly imperfect, since people oftentimes provide insufficient information, and many contact tracers must be hired[1].

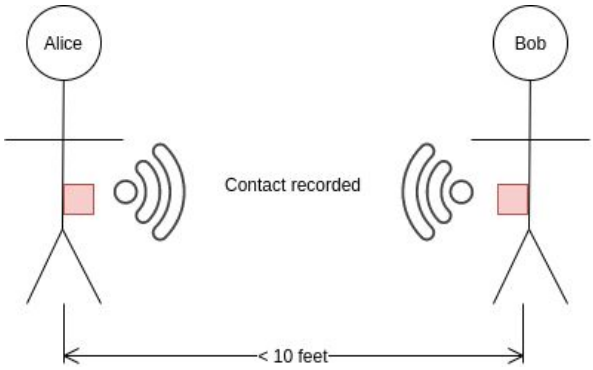
Mobile contact tracing apps, although a great improvement over manual contact tracing, still have serious flaws. Apps that use GPS suffer from the fact that GPS is not always available and also quite inaccurate, not to mention the privacy concerns of mass surveillance of everyone's locations. Apps that use NFC, or Bluetooth, to address the privacy and availability concerns of GPS, still fall short. In the case of NFC, the range is far too small, and in the case of Bluetooth, the ability to measure distance accurately is sorely lacking, which inevitably leads to high false positive rates[2]. Finally, modern smartphones are simply too expensive in many parts of the world, and few people have sufficiently sophisticated smartphones that can perform effective contact tracing.

Physical Design

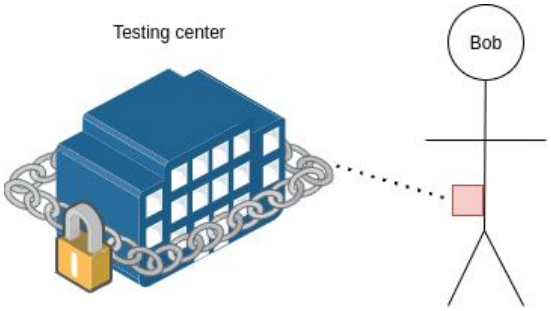
[1]https://www.nature.com/articles/d41586-020-03518-4

[2]https://www.forbes.com/sites/ramseyfaragher/2020/04/21/the-hidden-trade-offs-inside-contact-tracing-apps/?sh=dd085eeea07a

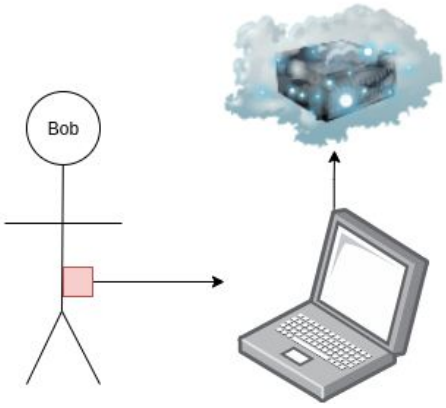
The following series of images demonstrates how the device is intended to be used, and how it can effectively detect and notify users of potential transmissions.



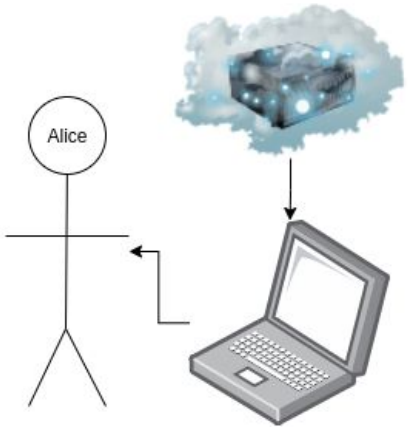
Alice and Bob's chips detect each other within 10 feet, and record each others' anonymous IDs in their internal storage.



Bob receives the bad news from a trusted testing center that he has tested positive. The testing center connects to Bob's chip and uploads a cryptographically signed message with his positive status.



Bob connects his device to his computer to charge it and to upload his positive status to a server.



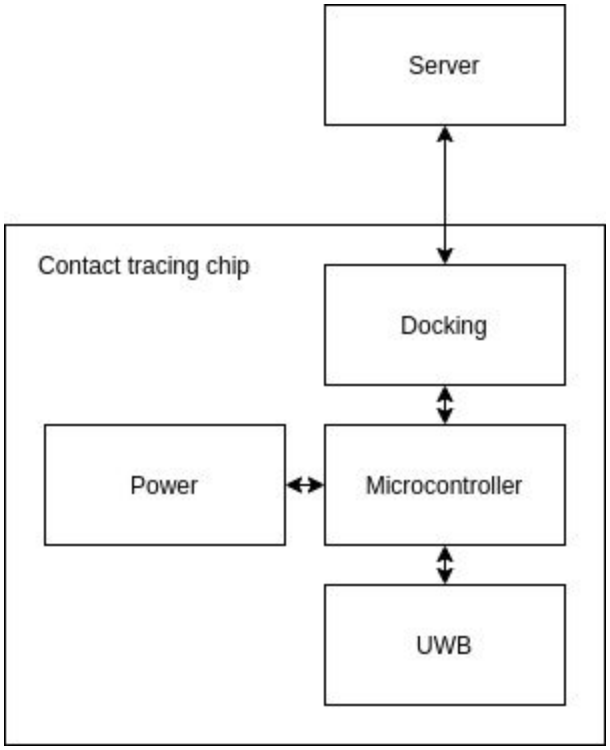
Alice receives the bad news immediately from the server that she was in contact with someone who was positive. She must now follow whatever public health policies have been set by her local government.

High-Level Requirements

- The chance of a false negative, which is defined as the device failing to record a contact despite two users being less than 10 feet apart, must be less than 25%. The chance of a false positive, which is defined as the device recording a contact despite two users being more than 10 feet apart, must be less than 25%.
- The device must be capable of operating for at least 12 hours without having to be charged.
- The protocol must provably prevent malicious attackers from faking a positive status. It must also provably prevent anyone with access to the server from deducing any individual's identity, without access to additional information, such as when any given user tends to connect to the server. This property can be formally proven, assuming a bug-free implementation.

Design

Block Diagram



This design ensures that the three high-level requirements are met. Since we use UWB, the chip will be capable of measuring distances with an accuracy of up to 10 centimeters[3]. This capability is enough to prevent false negatives and false positives, and if any additional accuracy is desired, the microcontroller can perform repeated measurements to gain confidence. The power subsystem will be using small rechargeable li-ion batteries with an average capacity of 560mAh[4]. Finally, privacy is ensured by the fact that the only information the server has about the user comes from the chip, and the only information the chip will send is an anonymous ID, positive status messages, and a list of anonymous contacts.

Functional Overview

Communication Subsystem (UWB)

- This subsystem contributes to the overall system by allowing our contact tracing card to be able to communicate and find the distance between other contact tracing cards in the area. The UWB module interfaces with the microcontroller subsystem through SPI and is powered through the stable power supply from the power subsystem. This subsystem is by far the most difficult to implement due to UWB being such a new technology with little documentation and support. We will have to do extensive testing and development to properly utilize the capabilities of ultra wideband communication.

Microcontroller Subsystem

[3]https://www.mouser.com/datasheet/2/412/DW1000_Datasheet-1878664.pdf
[4]https://www.mouser.com/ProductDetail/Panasonic-Battery/CR-2450A-GBN/?qs=17u8i%2FzlE8%252B2J0AMWy9Ldg%3D%3D

- This subsystem acts as the main processing of the whole contact tracing system and is responsible for initiating communication with other contact tracing systems. It leverages the SPI interface between the UWB chip to scan and discover other chips in the vicinity and do the appropriate encryption and storage of user information. It also uses the built in USB capability to communicate with the Docking Subsystem.

Power Subsystem

- This subsystem is responsible for managing the battery and utilizing USB power to safely recharge the lithium battery. It used the concept of balance charging. This process will check the voltages of each individual cell in the battery and ensure they all have the same voltage ensuring battery health and safe recharging. This is important due to the volatility of lithium batteries. We expect a roughly full day of usage with our system. The power subsystem will be using small rechargeable li-ion batteries with an average capacity of 560mAh. We see from the UWB datasheet that the nominal power consumption is 13.4mA and the nominal power consumption of an Arduino type microcontroller is approximately 11.3mA[5]. Doing some quick napkin math and ignoring other parts of the subsystem that require less power consumption we see that the contact tracing card should be operational for 18 hours.

Docking Subsystem (PC)

- This subsystem is responsible for interfacing with the chip, loading contact data, uploading it to the server, and receiving notification of potential transmissions from the server. As part of this subsystem, a special device driver to interface with the chip over USB will be necessary, as well as a simple GUI application that users can use. In order to interface with the server, a simple REST API will be used. This piece is the key component that feeds the server with the edges of the contact graph.

Server Subsystem

- The server is responsible for getting the edges of the contact graph from the Docking Subsystem, as well as positive COVID statuses. Given this information, it finds all nodes that are connected to positive users and sends a notification to their PCs (Docking Subsystem).

Block Requirements

Communication Subsystem (UWB)

- Properly send signals between UWB devices
- Ensure all chips within the given distance are noticed, and that chips outside the desired range are not noticed with at least 75% accuracy

Microcontroller Subsystem

- Conduct handshake between ultra wideband chips
- Verify and store COVID test results
- Communicate with with the Docking Subsystem over USB

Power Subsystem

- Keep system running for at least 12 hours at a time

---

[5]https://diyi0t.com/arduino-reduce-power-consumption/#:~:text=The%20power%20consumption%20of%20the,Pro%20Mini%20with%201.58mA.

- Allow users to recharge via USB

Docking Subsystem (PC)

- Communicate with chip over USB using a custom device driver
- Download and upload contact data

Server Subsystem

- Maintain a graph of anonymous contacts
- Inform users of potential transmissions

Risk Analysis

The subsystem that poses the biggest threat to the completion of this project is the communication. Although the datasheet for the UWB does state that it has the required resolution to determine distances accurately, real-life scenarios will have interference, obstacles blocking signals, and so on. There may certainly be difficulties arising from these problems, and in that case, we will have to come up with software means to increase reliability, or try different UWB chips. In addition, the UWB chips have little open source information online and sparse documentation. This means that we will have to prototype and extensively test the UWB subsystem to ensure we can have it do the requirements we noted. This is unlike using an established communication system like Bluetooth which has plenty of online documentation and support.

Ethics and Safety

A project of this nature will have several ethics and safety concerns. This is particularly so in user's privacy and how their data is used and stored. In the IEEE code of ethics this is stated as: "1. to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, to protect the privacy of others, and to disclose promptly factors that might endanger the public or the environment." In this ACM code of ethics this is stated as: "1.6 Respect privacy. The responsibility of respecting privacy applies to computing professionals in a particularly profound way. Technology enables the collection, monitoring, and exchange of personal information quickly, inexpensively, and often without the knowledge of the people affected." The privacy issues apply to our project with how user information is stored in terms of contact information and covid test results.

In order to deal with these privacy concerns we will be using a variety of methods to keep our users safe. First off we shall not store any personal information about users other than whom they have been in contact with and whether or not they are positive, and this information is stored anonymously. To accomplish this, we shall have a user ID randomly generated for each user that will be associated with their test results. This way access to the test results does not compromise the integrity of individual users. To avoid malicious actors pretending to be positive, positive status messages must be cryptographically signed by testing centers concatenated with the user ID to prevent replay attacks. We believe this project's purpose of upholding the safety and health of the public while preserving privacy is both feasible and ethical.