

# FoneAstra: Enabling Remote Monitoring of Vaccine Cold-Chains Using Commodity Mobile Phones

Rohit Chaudhri<sup>1</sup>, Eleanor O'Rourke<sup>1,2</sup>, Shawn McGuire<sup>2</sup>, Gaetano Borriello<sup>1</sup>, Richard Anderson<sup>1,2</sup>

<sup>1</sup>Department of Computer Science and Engineering  
University of Washington, Seattle, WA [USA]

<sup>2</sup>PATH, Seattle, WA [USA]

{rohitc, eorourke, gaetano, anderson}@cs.washington.edu, smcguire@path.org

## ABSTRACT

We present a low-cost, energy-efficient system to remotely monitor the temperature and location of vaccines in a country-wide “cold-chain”. Our system is based on FoneAstra [11] – a low-cost, microcontroller-based, programmable device that extends capabilities of low-tier mobile phones that are commonly used in developing countries. In the system discussed in this paper, FoneAstra is enhanced with a digital temperature sensor and integrated with a vaccine cold-box used to store vaccines in a temperature controlled environment. FoneAstra continuously monitors the temperature of the cold-box, aggregating readings over a period of time. It uses the mobile phone to which it is coupled to send periodic SMS messages with routine temperature reports or immediate alerts if it detects abnormal temperature conditions. Additionally, it enables location-tracking of vaccines in transit, based on the mobile phone’s cell tower-IDs. We present results from an ongoing lab-deployment done at PATH [18], our Seattle-based partner NGO for this project. Over the next few months, we will deploy this temperature and location monitoring system for vaccine cold-chains in several countries in which PATH operates. The client in this system, which includes a temperature sensor, FoneAstra and a mobile phone, costs \$50; while the server, which includes a Netbook and a GSM modem, costs \$500. We discuss how our system can scale up to enable large-scale monitoring while incurring low overhead costs.

## Categories and Subject Descriptors

J.7 [Computer Applications]: Computers in Other Systems: *command and control, consumer products, process control.*

## General Terms

Measurement, Design, Experimentation.

## Keywords

Remote Monitoring, Large-Scale Distributed Sensing, Location Tracking, Mobile Systems.

## 1. INTRODUCTION

Effective management and delivery of vaccines is an important aspect of improving healthcare for under-served communities in developing countries. The World Health Organization (WHO) estimates that vaccines prevent over 2.5 million child deaths from fatal diseases annually [23]. Access and availability of vaccines has improved tremendously over the last 5 to 10 years with the

influx of donor resources and increased supplies from several new manufacturers. However, delivery and storage of these temperature and time sensitive vaccines continues to pose challenges in many developing countries. The WHO specifies the procedures to be followed for safe storage of vaccines, which include routine monitoring and reporting of alarm conditions (e.g. storage temperatures deviating from the allowable range for a prolonged period of time) [21, 27, 28]. Vaccines typically need to be stored in a controlled +2° to +8°C environment in order to ensure their potency (e.g. Figure 1 shows a few pictures of equipment used to store vaccines at a SILIAS Health Clinic in Nicaragua). However, country cold-chains (the supply chains for goods requiring constant refrigeration) are not always reliable; as a result, vaccines may be exposed to temperatures that are either too hot or too cold, causing vaccines to be spoiled [8, 12, 13, 16]. Intermittent power outages, limited availability of skilled staff for equipment maintenance, lack of spare parts, and interruptions during transportation are common causes for these losses [4].



Figure 1: Cold-boxes for vaccine storage at a SILIAS Health Clinic in Nicaragua

Commonly used procedures to monitor the temperature of vaccines stored in cold-boxes or refrigerators at storage facilities require the facility staff to manually record temperature twice daily (per WHO guidelines [21, page 40]) on paper. This method produces accurate results (depending on the accuracy of the thermometer used and compliance of the staff performing the procedure); however, manual recording on paper makes it difficult for organizations to aggregate such data especially when facilities are not geographically co-located. Figure 2 shows the paper-based method used at a SILIAS health clinic to record the temperature of equipment used for vaccine storage.

There is significant potential to reduce individual instances of spoilage and help diagnose problem areas of a cold-chain by improving the monitoring of storage temperatures and locations

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ACM DEV'10, December 17-18, 2010, London, United Kingdom.  
Copyright 2010 ACM 978-1-4503-0473-3/10/12... \$10

and providing a way to report this data in real-time. For example, continuous monitoring systems could enable timely detection of changes in the temperature profile of equipment. This information could be indicative of impending equipment-failure.



**Figure 2: Paper-based Cold-box Temperature Reporting at a SILIAS Health Clinic**

In this paper, we present a low-cost, energy-efficient system to remotely monitor the temperature and location of vaccines in a country-wide cold-chain. Our system is based on FoneAstra [11] – a low-cost, microcontroller-based, programmable device that extends the capabilities of low-tier, non-programmable Nokia mobile phones that are commonly used in developing countries. FoneAstra is enhanced with sensors to enable continuous monitoring of temperature and it tracks the location of vaccines in transit based on recording the ID of cell towers that associate with the mobile phone to which it is connected. Routine reports, containing temperature readings accumulated over a period of time and the current location, are sent as SMS messages to a back-end server via the mobile phone coupled to FoneAstra. Additionally, the device generates immediate SMS alerts if temperatures deviate outside the allowable range. This is critical for diagnosing problems in the cold-chain and enabling timely intervention to prevent vaccines from being spoiled. For instance, if a problem occurs while vaccines are being transported by truck, the device will send an SMS alert to the truck driver, who can then try to rectify the problem. The low cost of the monitoring device (~USD15 in prototype scale) and its ability to leverage cheap mobile phones (~USD25) for communication reduces the cost of the overall system significantly thereby significantly lowering the barrier for large-scale deployments.

Evolution of vaccine technology might make it possible to safely store vaccines over a larger temperature range [25]. For instance, it might be possible in the near-term future to store vaccines at room temperature. However, monitoring systems to ensure that vaccines always remain within the acceptable temperature range would still be required. This fact needs to be considered as we build monitoring systems today. Our systems need to be flexible and reconfigurable, so as to easily adapt to changes in the requirements for vaccine storage.

The principal contributions of the work presented here are:

- A scalable, low-cost system for continuous monitoring of country-wide vaccine cold-chains.
- A technique to extend capabilities of low-tier mobile phones to enable large-scale distributed sensing.

- A demonstration that it is possible to remotely detect anomalies or failures in the cold-chain in a timely manner.

The rest of the paper is structured as follows. Section 2 describes the end-to-end architecture of our cold-chain monitoring system. We discuss the implementation of our system and deployment of a test-bed at PATH [18] in section 3. Experimental results obtained from the test-bed are discussed in section 4. We compare our system to related work in section 5. Section 6 concludes the paper with a discussion of future work and the upcoming deployments of our system in several developing countries.

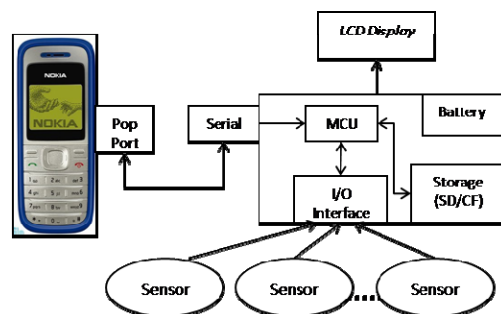
## 2. System Architecture

Our system is based on the client/server computing model in which multiple, geographically dispersed FoneAstra-based monitoring clients communicate to a single remote server.

### 2.1 Client

The client consists of a low-tier Nokia phone connected to FoneAstra to which a temperature sensor has also been attached. FoneAstra is based on NXP’s LPC2148 processor [15], an ARM7-based microcontroller. The ARM7 processor was chosen because of its low-cost, support for low-power operating modes, and rich I/O interfaces (UART, SPI, I2C, USB and GPIO). We added a memory card interface so that the device has ample and easily accessed bulk memory for the persistent storage of sensor data. Sensors are connected to FoneAstra using the available I/O interfaces. One of the 2 UARTs available on the LPC2148 is used for connecting to the mobile phone’s serial port, while the other UART is used for firmware upgrades and application debugging. [1] provides details of the platform, including the schematics.

FoneAstra communicates to the coupled mobile phone using a serial protocol called FBUS [9] that is implemented on Nokia phones and is accessible over its data port (known as a “Pop Port” on low-tier Nokia phones). This serial interface exposes a rich set of capabilities such as the ability to query phone information (e.g. IMEI identifier and phone number), sending and receiving SMS messages, and initiating and answering phone calls programmatically. This enables FoneAstra to communicate with the remote server via SMS messages or phone calls. Cellular information such as the ID of the currently associated cell tower and its signal strength is also accessible over this interface. This information will be used to track the location of vaccines as they are being transported from central warehouses to remote healthcare facilities. The supported mobile phones include currently shipping low-tier models like the Nokia 1209, 1661/2 etc, as well as older models like the Nokia 1110 and 1200. It is expected that Nokia will continue to provide this interface as it serves important functions for users and cellular service providers.



**Figure 3: FoneAstra Functional Block Diagram**

Figure 3 shows an architectural block diagram of FoneAstra connected to a Nokia phone. It shows the hardware extensibility of the platform. While we have currently integrated only temperature sensors for the vaccine cold-chain monitoring system; it is possible to extend the device with other sensing capabilities as well. For example, in future work we will add an LCD display and a sensor to monitor the electrical power of cold-boxes or refrigerators Figure 4 shows FoneAstra connected to a Nokia 1650 phone.

[11] discusses other possible architectural instantiations of FoneAstra as well the different classes of applications it enables.



Figure 4: FoneAstra connected to a Nokia 1650 phone

## 2.2 Server

The server in our system runs on an off-the-shelf Dell Latitude 2110 Netbook with an attached MultiModem GPRS [17] cellular modem that enables it to communicate with remote clients. The server receives routine reports and alarms from clients via SMS messages. It processes this information and stores it in a local database, sending additional SMS messages as necessary. For example, the server can be configured to send SMS notifications to the appropriate health officials and maintenance staff when a refrigerator reports out-of-range temperatures. End-users of the system (e.g. supervisory staff) view information stored in the database via an internet portal that presents graphical visualizations of the data aggregated from clients over a period of time. Moving forward, the server would also be able to remotely control certain aspects of the clients – e.g. it would be able to reconfigure system parameters such as the sampling rate of the sensor(s).

## 2.3 End-to-End System

Figure 5 depicts the end-to-end architecture of the system. It shows clients monitoring multiple cold-boxes in two locations (A and B). However, a real deployment of the system will have several client systems installed in the field. Some of the clients will be at fixed locations such as healthcare facilities, while others will be mobile systems that monitor vaccines as they are distributed from central warehouses. From an initial hardware-cost perspective, deploying such a monitoring system will incur a relatively higher, one-time cost of setting up the server infrastructure, while each client installation will incur a significantly lower incremental cost. We discuss detailed costs of our current implementation in the next section.

## 3. Implementation

Over the next few months we will deploy our cold-chain monitoring system in several developing countries. This work will be done in collaboration with PATH, our Seattle-based NGO partner in this project. Towards that end, we have deployed a test-

bed at PATH’s engineering lab and are currently testing the system before the upcoming deployments.

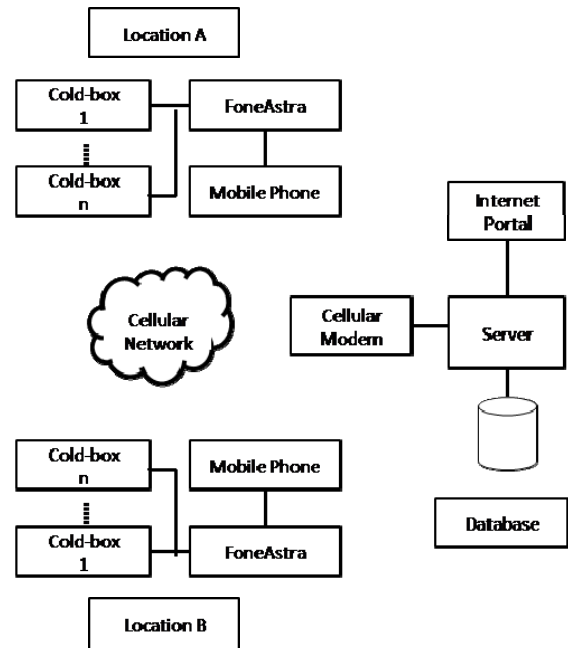


Figure 5: End-to-End System Architecture

## 3.1 System Setup

We are using FoneAstra with a DS18B20 digital temperature sensor produced by Maxim [7] (Figure 7) to monitor and report temperatures of a solar-powered cold-box called TwinBird [26] (Figure 6). The total cost of our monitoring client (not including the cold-box) is ~USD50.

FoneAstra and the Nokia phone are powered by separate lithium-ion batteries. To charge these batteries we have tapped into the power-supply of the TwinBird. The energy requirements of FoneAstra and the mobile phone are relatively low compared to those of the cold-box, which can draw up to 2 amps of current when it is actively cooling (as opposed to maintaining a fixed temperature). Therefore, we expect that the battery chargers will always be connected to the devices in our field deployments. In its active-mode (e.g. while sampling the sensor, sending SMS etc) FoneAstra draws about 50 milli-amps (mA) of current, while in its low-power mode, it draws under 100 micro-amps ( $\mu$ A) of current. In real deployments, we expect to sample the temperature sensor once every few minutes and send SMS reports 2-4 times per day. Therefore, FoneAstra will be programmed to be in its low-power mode for most of the time and only wake up periodically to sample the sensor or send an SMS. This will ensure that the system’s overall power consumption very low (dominated by power consumption in FoneAstra’s low-power mode). The mobile phone on average consumes  $\sim$ 5mA per hour. Given the low energy-overhead of the monitoring system, we will explore the possibility of using a single battery to power both the devices in the near future. This will significantly simplify the deployment and maintenance of our system.

The application code running on FoneAstra is implemented in C. We use the “CrossWorks for ARM” tools from Rowley Associates for development and debugging. Currently FoneAstra does not have an operating system and all the hardware-specific



initialization is done by a Board Support Package (BSP) provided by CrossWorks; the BSP is implemented in assembly language. This has been sufficient for our needs so far because the software running on FoneAstra is not too complex. It is an interrupt-driven system that handles interrupts from a limited number of sources, namely: UART (to communicate with the phone), Real Time Clock and Timers. However, as our system continues to evolve and we add more sensors to the platform, we will need an operating system to manage the increasing complexity and concurrency. Hence, we are currently investigating a few light-weight Real Time Operating Systems (RTOS) as possible options. We expect to have an RTOS running on FoneAstra very soon and we will redesign the monitoring application to be a collection of cooperating “tasks” on the RTOS.

The server machine is running Ubuntu Linux, version 10.04. The application is built using RapidSMS [20], an open-source framework for developing SMS driven applications that is based on the Django [6] web application framework. We use MySQL as our database engine. The total cost of the server is ~USD500.



**Figure 6: Solar-powered Cold-box Enhanced with Temperature Monitoring**

### 3.2 Messaging Protocol

The communication requirements between the clients and server in our system are fairly straightforward and will not generate very high traffic under normal operating conditions. In a basic monitoring system where messages are only sent from the clients to the server, clients will periodically send routine temperature reports aggregated over a period of time and generate SMS alerts whenever alarm conditions are detected. So at the very least, the system requires application-level support to communicate these two types of messages.

Each temperature reading received from the DS18B20 is a 12-bit number (1 sign bit, 7 bits for the decimal value and 4 bits for the fractional value) [7]. In order to minimize SMS costs we attempt to maximize the number of temperature readings that can be included in each routine report message. To achieve this, we split each 12-bit temperature reading into 2 6-bit numbers. Each 6-bit number is encoded as a readable ASCII character before being included in the SMS message sent to the server. This allows each temperature reading to be transmitted as two ASCII characters. With this encoding scheme we are able to transmit a maximum of 80 temperature readings (160 characters) in a single SMS message.

In our initial tests we tried to report 48 temperature readings in 1 SMS message (this translates to a sampling interval of 15 minutes with a 12-hour reporting interval). We used a simply formatted message payload in which the 1<sup>st</sup> byte indicates the number of temperature readings that follow in the SMS message (a total message size of 97 bytes).



**Figure 7: FoneAstra Enhanced with a Temperature Sensor Connected to a Nokia 1661 phone**

To simulate this scenario in the lab, we used a sampling rate of 15 seconds and a reporting interval of 12 minutes. In order to debug systems issues, we generated a short SMS debug message (14-bytes long) every minute that had a timestamp and the current temperature reading. In our first test-run, which lasted for 5 hours, we observed that the more frequent debug messages were delivered reliably but the longer report messages were often lost or delayed in transit. We initially suspected that these errors might be related to either the message-size or the fact that the test-run spanned two service-provider (SP) networks (i.e. the SMS sender and receiver had service from different SPs). Oliver [5] and Zerfos et al. [2] have done extensive studies to characterize SMS behavior in cellular networks. However, we didn’t find any data correlating the size of the SMS message to network performance. To confirm our suspicion we ran tests that generated SMS messages of variable sizes over a 12-hour period. We performed two sets of experiments to understand if using different SPs had any impact on the performance of SMS delivery. For these experiments we varied the message size and SP, however maintained the other conditions of our initial 5-hour test-run such as the testing locations and late-night testing hours. Interestingly, we did not encounter any problems in these two experiments. We summarize results from our experiments in

Table 1. The “Delayed” row shows messages that were delayed by over 3 minutes (in the worst-case we saw delays of about 10 minutes).

Despite the performance in our initial test-run, these results are quite encouraging. However, this experience helped us to understand the dynamic nature of the cellular network. Given that our field deployments will be in different developing countries; the exact nature of the cellular networks in those countries needs to be better understood so that we can properly provision the system on the local SP networks. This has motivated us to design some flexibility into our system. Our current protocol includes metadata, such as a sequence number and retry-count. This information will help us understand characteristics of the

networks during our deployments. System parameters such as the sampling rate, reporting interval and report size are configurable. We will tune these parameters for optimal performance based on the network characteristics that we discover through the metadata in our protocol.

**Table 1: Summary of Network Characterization Tests (SP=Service Provider)**

	SP1-SP2		SP1-SP1	SP1-SP2
<b>Duration (hours)</b>	5		12	12
<b>Messages</b>	275	25	500	500
<b>Size (bytes)</b>	14	97	75-160	75-160
<b>Lost</b>	0	4 (16%)	0	0
<b>Delayed</b>	0	5 (20%)	0	0
<b>Replayed</b>	17 (6.2%)	0	0	0

## 4. Experimental Results

For experimentation and data collection in the lab, we configured the sampling interval on FoneAstra to be higher (30 seconds – 2 minutes) than what a real cold-chain monitoring system might require. Additionally, routine reports are generated every hour to provide us with detailed measurements sooner, rather than being generated once in several hours, as might be of the norm for a typical system deployed in the field.

### 4.1 Temperature Monitoring

#### 4.1.1 Ground Truth Validation

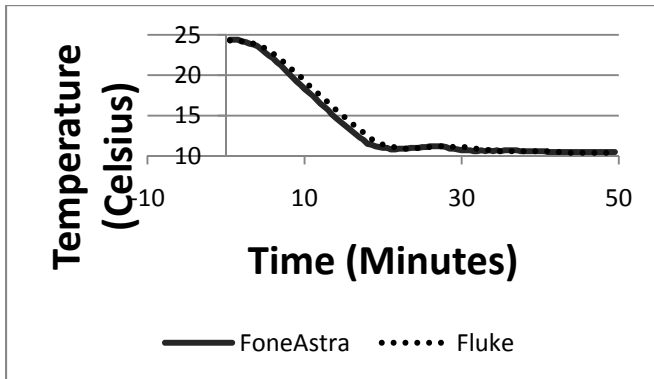


Figure 8: Ground Truth Validation of Temperature Sensor

The datasheet of the DS18B20 specifies that the sensor is accurate to within 0.5°C in the -10°C to +85°C temperature range; however, the sensor is not waterproof as it ships. Since we are using the sensor in environments where moisture content is high, we waterproofed the sensor by insulating the sensor’s body and connecting leads with brush-on electrical tape. Figure 7 shows the sensor after it has been treated with the brush-on tape. To ensure that the waterproofed sensor performs at its specified accuracy we compared the sensor’s temperature readings with the readings from a type-K thermocouple connected to a Fluke multimeter

[10]. The graph in Figure 8 shows the results of our ground-truth validation experiments. For this experiment, the thermocouple and the DS18B20 were taped at the same location inside the TwinBird. FoneAstra was configured to sample the sensor every 30 seconds. The internal temperature of the TwinBird was +24.5°C (room temperature) when the experiment was started. Temperature readings from the Fluke thermocouple were manually recorded every time FoneAstra queried its temperature sensor (it is programmed to blink an LED when it samples the temperature sensor). We ran this experiment for about an hour (the graph shows readings from the first 50 minutes), until the TwinBird’s internal temperature stabilized to approximately +10.0°C. As can be seen from the graph, the temperatures reported by FoneAstra and the Fluke thermocouple were very close throughout the experiment. FoneAstra reported temperatures that were either slightly lower than or equal to Fluke’s readings, but the readings from the two devices eventually converged to +10.4°C, when the TwinBird’s internal temperature stabilized. One possible reason for the minor difference in readings could be that the temperature sensors were being sampled at different rates and the exact time at which each sensor was sampled was most likely different for each device. Based on this experiment, we concluded that waterproofing had not modified the thermal characteristics of our temperature sensor.

#### 4.1.2 Remotely Detecting Power Failure

During our in-lab deployment, the power supply to the TwinBird was inadvertently disconnected for a few hours. This provides us with data simulating a power failure in the field. Before the power was disconnected, the cold-box had been running continuously for about a week and FoneAstra had been monitoring the temperature during this time. FoneAstra was configured to sample its temperature sensor every 2 minutes and report readings every hour. Figure 9 shows an almost flat temperature curve until Time=132. This had been the stable temperature over the week-long continuous operation. The power supply was disconnected at Time=132 and as the graph shows, the temperature starts rising rapidly after this point in time. One author was monitoring the temperature logs remotely at home and could infer that the rapid rise in temperature indicated power failure and verified it in-person upon reaching the lab. Power was resumed at Time=442 and the cold-box returned to its stable temperature shortly thereafter. Having this capability to detect power failures in cold-chains will be important to help reduce vaccine spoilage.

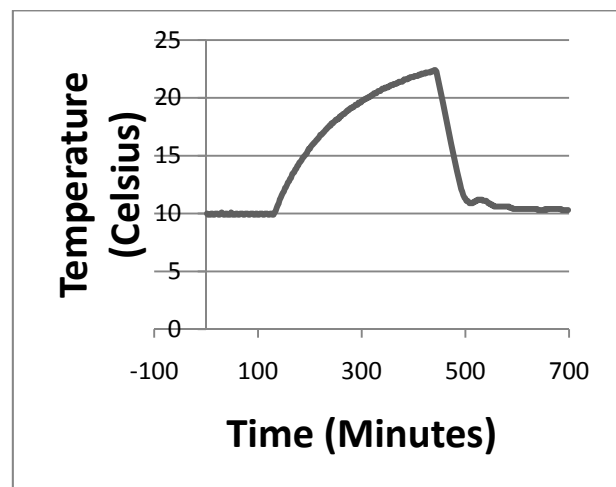


Figure 9: Power Failure

### 4.1.3 Remotely Detecting Lid Open/Close Events

Figure 10 shows a graph from another interesting experiment we conducted in the lab. FoneAstra was configured to sample the sensor every 30 seconds. In this experiment, we opened the lid of the cold-box after it had reached its stable operating temperature and left the lid open for about 30 minutes. As expected, the temperature begins to rise very soon after the lid is opened. In fact, within a couple of minutes we are able to remotely detect that the lid had been opened. We also see a steeper temperature curve during the time the lid is left open than what we saw when the power to the cold box was disconnected (the temperature curve shown in Figure 9). This is because the cold-box's internal temperature rises rapidly when the lid is left open due to the large difference between the internal temperature and room temperature. When the lid was closed (Time=28), we observe a steeper-than-normal cooling curve (see Figure 8 for comparison). This is because the cold-box aggressively tries to recover and reach its stable operating temperature. This result is interesting because we are able to detect if the lid has been left open for an abnormally long time period. Additionally, we hypothesize that features of this temperature curve might be indicative of the operational efficiency of the cold-box. We need to experiment further and consult with refrigeration experts to validate this hypothesis.

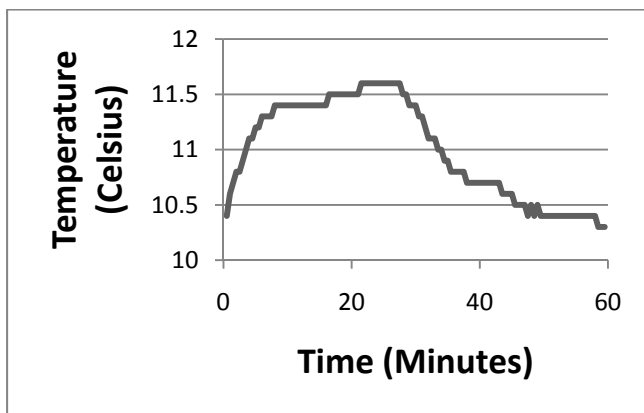


Figure 10: Cold-box Lid Left Open for ~30 Minutes

We also did some additional experiments in which we opened the lid for short durations of time (30 seconds – 2 minutes); simulating normal usage scenarios for the cold-box. However, instead of shutting the lid completely, we left it ajar for an extended period of time, thus simulating scenarios where the cold-box lid was not closed properly. As expected, we saw the temperature rise (by 0.2°C-0.5°C) when the lid was opened and it remained high for a short duration as the lid was left ajar. Interestingly, the cold-box recovered from this anomaly quickly and its internal temperature returned to its stable value. With the lid ajar, the cold-box was losing energy while maintaining its internal temperature, so its current drain increased slightly (0.1-0.2 amps); this was also as expected. However, when the lid was closed, we saw that the current drain did not drop. We monitored the current for about 30 minutes after closing the lid and did not see the current consumption decrease. We do not know yet if this is normal behavior, but suspect that this might give some insights into the operational efficiency of the cold-box. We need to consult with refrigeration experts about this behavior as well.

Based on these experiments, we have concluded that additional sensors are needed to monitor the cold-box's current consumption and lid open/close events. In future work, we will explore whether it is possible to determine impending failures of the refrigeration system based on data gathered by these types of sensors.

## 4.2 Location Tracking

Our lab test-bed is not mobile so we do not track the cold-box's location in this deployment. However, in our field deployments we will track the location of vaccine carriers while they are being transported from central warehouses to healthcare facilities. Instead of using a GPS-based device that will raise the overall cost of our system, we will provide coarse-grained location tracking based on the locations of cell towers that are associated with FoneAstra's mobile phone. FoneAstra will be programmed to periodically query the connected mobile phone's current cell tower ID. This information will be sent to the server via SMS, which will then obtain the cell tower's location from a Google Web Service [14]. Valid requests sent to this Web Service contain a cell tower's ID and its Location Access Code (LAC); this information uniquely identifies the cell tower. Valid responses from the Web Service contain the latitude and longitude of the cell tower.

We did some experiments in India to validate this approach. For data collection, FoneAstra was programmed to query the mobile phone's current cell tower ID every two minutes and store this information on its memory card. The information was processed offline to obtain the location of cell towers from the Google Web Service. One of the authors travelled with FoneAstra connected to a Nokia 1650 phone to collect data. Figure 11 shows the locations of cell towers that were "seen" by the phone during an 8-mile round-trip in Bengaluru, India. Figure 12 shows the route of the same trip generated by Google Maps. These figures show that the trip approximation obtained using cell tower IDs is very close to the trip-route generated by Google Maps. Figure 13 shows the location of cell towers, the IDs for which were obtained during a 45-mile 1-way trip from Bengaluru to Talai, a village in Tamil Nadu. Figure 14 shows the route generated by Google Maps for the same trip. The purpose of this trip, which covered some rural areas in the states of Karnataka and Tamil Nadu, was to determine if cell tower IDs can be used to effectively enable location tracking in remote, rural areas where cellular coverage is sparse. FoneAstra was powered off during parts of the journey that covered urban areas or a national highway. Hence most of the cell tower markers are seen in the lower half of the figure, which cover the rural areas of this trip (the few markers in the Bengaluru area indicate where the trip started). It can be seen that even though the cell tower density is much lower in rural areas, the trip approximation is fairly accurate.

These results demonstrate that cell tower IDs obtained from FoneAstra's mobile phone, could be used to track the location of vaccines when they are in transit. However, it is reasonable to expect that cellular coverage won't be available during parts of the journey. FoneAstra won't be able to communicate with the server during this time, and location for these parts of the journey won't be available.



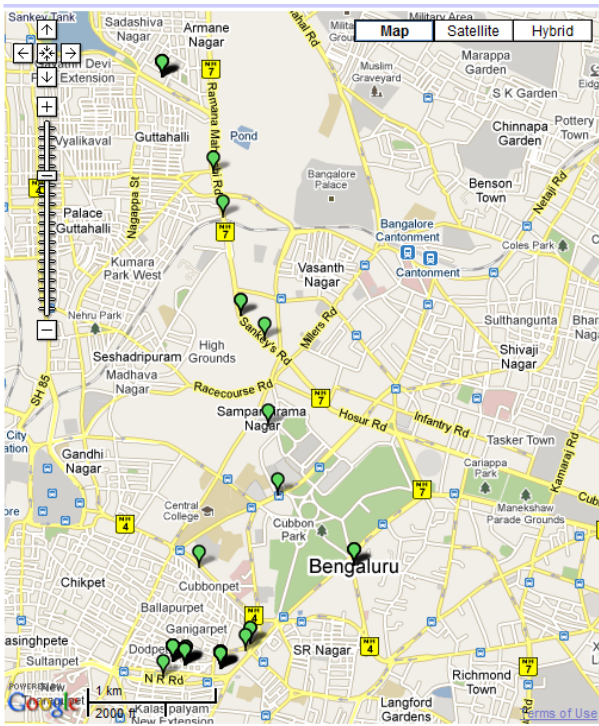


Figure 11: Route Approximation of an 8-mile Round-trip in Bengaluru, India

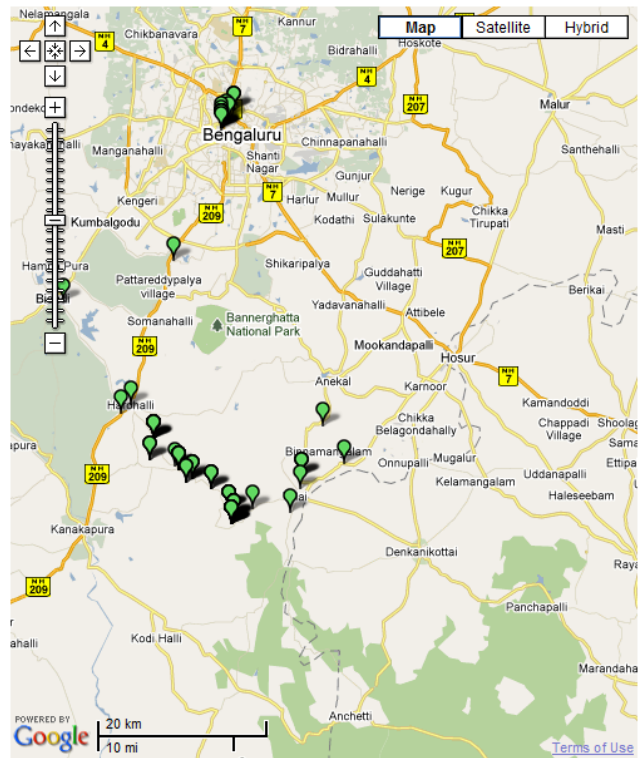


Figure 13: Route Approximation of a 45-mile trip from Bengaluru to Talai, a Village in Tamil Nadu, India

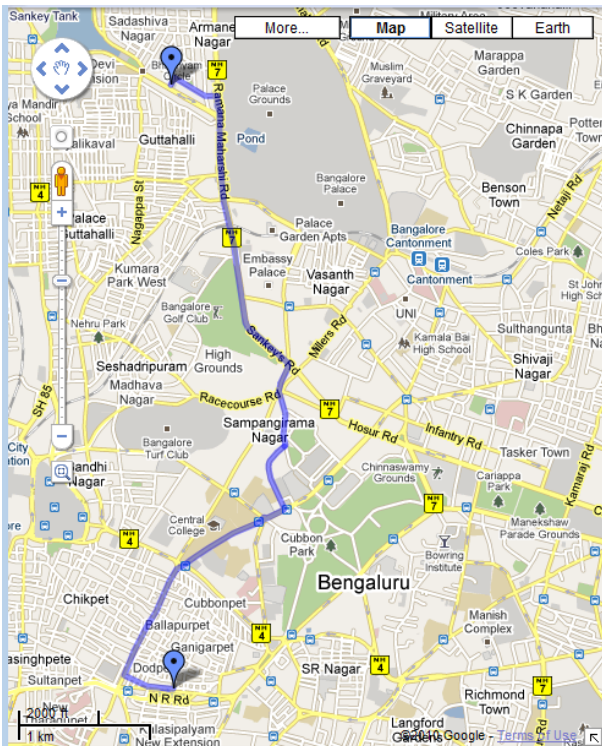


Figure 12: Trip Generated by Google Maps for the 8-mile Round-trip

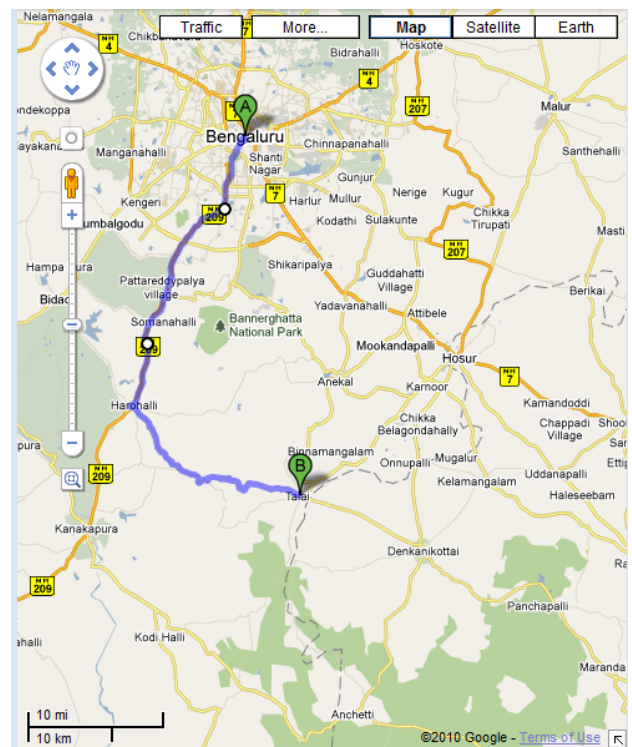


Figure 14: Trip Generated by Google Maps for the 45-mile trip

## 5. Related Work

A recent report from the WHO [24] describes two alarm-based temperature monitoring systems that have been installed at national immunization stores in Sudan and Iran. These systems enable continuous temperature monitoring of cold-stores at their respective facilities. However, the systems do not extend beyond the individual facilities so as to enable monitoring of the entire vaccine cold-chain. SmartConnect [22] proposes a facility-based temperature monitoring device that leverages a custom cellular modem for sending temperature reports and alarms to a server over SMS. While the system will enable continuous monitoring of temperature at storage facilities as well as data aggregation at the server, its cost will be higher because it uses a custom cellular modem for communication (which costs ~USD100 for basic hardware). This makes it difficult to scale up the system to cost-effectively monitor the entire vaccine cold-chain of a country. For instance, in addition to monitoring temperatures at facilities, the system also needs to monitor and report temperatures of cold-boxes as they are being transported from central warehouses to remote storage facilities.

Systems to enable location-tracking based on mobile phones' current cell tower-ID have been demonstrated in previous work [3, 19]. However, such systems rely on APIs that are available only on programmable phones. Our approach for accessing mobile phones' current cell tower-IDs is different and creates new opportunities for building location-based applications for non-programmable, low-tier phones that are commonly used in developing countries.

Systems to enable remote monitoring have been built in the Wireless Sensor Networks and Urban Sensing communities. We have discussed how FoneAstra aligns with these research areas in [11].

## 6. Conclusions and Future Work

In this paper we have presented a low-cost, energy-efficient system that enables remote monitoring of large-scale vaccine cold-chains. The monitoring client in our system, which includes a temperature sensor, FoneAstra and a mobile phone, costs \$50; while the server, which includes a Netbook and a GSM modem, costs \$500. We have deployed the system at PATH's engineering lab in Seattle, where we are performing experiments and validation tests before we deploy it in developing countries. The client has been integrated with a commercially available cold-box called TwinBird. Our initial experiments have shown positive results. The client continuously monitors the temperature of the cold-box and sends routine SMS reports to the server. Based on the data collected by the client, we are able to detect erroneous conditions in the cold-box such as power-failures and a lid that is left open for an abnormally long period of time. Temperatures reported by the client are within 0.5°C of the ground truth. Some data collected from urban and rural areas of India has shown that FoneAstra can be effectively used to enable coarse-grained location-tracking based on cell tower IDs obtained from the coupled mobile phone. We will be using this capability in our upcoming field deployments.

Over the next few months we will deploy our cold-chain monitoring system in Albania, Nicaragua, Senegal and Tunisia. This will be done in collaboration with PATH, our NGO partner in this project. Monitoring clients will be deployed at health facilities operated by PATH, where vaccines are stored in temperature controlled environments. Trucks that transport vaccine cold-boxes from central warehouses to health facilities

will also be instrumented with monitoring clients. In addition to monitoring the temperature of these mobile cold-boxes, the client will also track the location of the truck. Such a deployment will provide temperature monitoring and location tracking of the entire vaccine cold-chain. The low cost of the client makes it affordable to do large-scale deployments, and we expect that this will address one of the barriers to adoption of such systems in developing countries.

There are a number of important enhancements that we hope to incorporate into the system, some of which will be completed before our field deployments:

- Have a light-weight RTOS running on FoneAstra and redesign the application as a collection of concurrent cooperative tasks.
- Integrate additional hardware with FoneAstra. We need sensors to monitor the current consumption of the cold-box and detect when its door has been opened or closed. We need an LCD on FoneAstra to display simple statistics like the current temperature and daily maximum/minimum temperatures, as well as other feedback to users.
- Redesign the power system for FoneAstra so that only one rechargeable battery on the client powers all the hardware (phone, FoneAstra, and sensors).
- Implement 2-way messaging in the system to enable the server to query/reconfigure operational parameters of the client.

## 7. REFERENCES

- [1] R. Chaudhri and K. Toyama, "A Programmable Accessory for Cell Phones - Microsoft Research.", MSR-TR-2009-178.
- [2] X. Meng, P. Zerfos, V. Samanta, S.H. Wong, and S. Lu, "Analysis of the reliability of a nationwide short message service," *IEEE INFOCOM 2007. 26th IEEE International Conference on Computer Communications*, 2007, pp. 1811–1819.
- [3] P. Javid and T.S. Parikh, "Augmenting rural supply chains with a location-enhanced mobile information system," *Information and Communication Technologies for Development*, 2006, pp. 25–26.
- [4] John Snow Inc. "Cold chain and logistics management: an essential part of safe and effective vaccination programs", *IMMUNIZATION basics SnapShots*. 2008; 8:1–4. Available at: [http://immunizationbasics.jsi.com/Newsletter/Archives/snapshots\\_volume8.pdf](http://immunizationbasics.jsi.com/Newsletter/Archives/snapshots_volume8.pdf)
- [5] E. Oliver, "Characterizing the transport behaviour of the short message service," *Proceedings of the 8th international conference on Mobile systems, applications, and services*, San Francisco, California, USA: ACM, 2010, pp. 223-238.
- [6] Django: <http://www.djangoproject.com>
- [7] DS18B20 - Programmable Resolution 1-Wire Digital Thermometer: <http://www.maxim-ic.com/datasheet/index.mvp/id/2812>
- [8] Techathawat S, Varinsathien P, Rasdjarmrearnsook A, Tharmaphornpilas P, "Exposure to heat and freezing in the vaccine cold chain in Thailand", *Vaccine*. 2007; 25:1328–1333.
- [9] FBUS: <http://en.wikipedia.org/wiki/FBUS>
- [10] Fluke 170 Series Digital Multimeters: <http://us.fluke.com/fluke/usen/digital-multimeters/fluke-170.htm?PID=56027>
- [11] R. Chaudhri, G. Borriello, and W. Thies, "FoneAstra: Making Mobile Phones Smarter," *4th ACM Workshop on Networked Systems for Developing Regions (NSDR)*, 2010.



- [12] Setia S, Mainzer H, Washington ML, Coil G, Snyder R, Weniger BG, "Frequency and causes of vaccine wastage", *Vaccine*. 2002; 20:1148–1156.
- [13] Matthias DM, Robertson J, Garrison MM, Newland S, Nelson C. "Freezing temperatures in the vaccine cold chain: a systematic literature review", *Vaccine*. 2007; 25(20):3980–3986.
- [14] Google Web Service:  
<http://www.google.com/glm/mmap>
- [15] LPC214x Series Device Highlight:  
<http://ics.nxp.com/products/lpc1000/lpc11xx/>
- [16] Nelson C, Froes P, Dyck AM, et al. "Monitoring temperatures in the vaccine cold chain in Bolivia", *Vaccine*. 2007; 25:433–437.
- [17] MultiModem GPRS:  
[http://www.multitech.com/en\\_US/products/families/multimodemgprs/](http://www.multitech.com/en_US/products/families/multimodemgprs/)
- [18] PATH: <http://www.path.org/>
- [19] A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, J. Tabert, P. Powledge, G. Borriello, and B. Schilit, "Place Lab: Device Positioning Using Radio Beacons in the Wild," *Pervasive Computing*, 2005, pp. 116-133.
- [20] RapidSMS: <http://www.rapidsms.org>
- [21] "Safe vaccine handling, cold chain and immunizations", *World Health Organization*. 1998. Available at:  
<http://www.who.int/vaccines-documents/DocsPDF/www9825.pdf>
- [22] R. Anderson, E. Blantz, D. Lubinski, N. Rourke, M. Summer, and K. Yousoufian, "SmartConnect: Last mile data connectivity for rural health clinics," *4th ACM Workshop on Networked Systems for Developing Regions (NSDR)*. 2010.
- [23] "State of the world's vaccines and immunization. Third edition", *World Health Organization*. 2009. Available at:  
<http://www.who.int/immunization/sowvi/en/index.html>
- [24] M. Haghgou, "Temperature Recording System in National Vaccine Stores Khartoum, Sudan, and Tehran, Iran", *World Health Organization, PATH*. 2010.
- [25] "Temperature Sensitivity of Vaccines", *Dept. of Immunization, Vaccines and Biologicals, World Health Organization*. 2006. Available at:  
[http://whqlibdoc.who.int/hq/2006/WHO\\_IVB\\_06.10\\_eng.pdf](http://whqlibdoc.who.int/hq/2006/WHO_IVB_06.10_eng.pdf)
- [26] TwinBird, SC-C925 Portable Freezer Cooler 25L:  
[http://fpssc.twinbird.jp/en/sc\\_super\\_box\\_e.html](http://fpssc.twinbird.jp/en/sc_super_box_e.html)
- [27] "WHO/PQS/E06/TR03.1: Programmable electronic temperature and electronic logger systems with integral alarm and auto-dialer options". Available at:  
[http://www.who.int/immunization\\_standards/vaccine\\_quality/pqs\\_e6\\_temp\\_monitoring/en/index.html](http://www.who.int/immunization_standards/vaccine_quality/pqs_e6_temp_monitoring/en/index.html)
- [28] "WHO/PQS/E06/TR07.1: Electronic shipping indicators". Available at:  
[http://www.who.int/immunization\\_standards/vaccine\\_quality/pqs\\_e6\\_temp\\_monitoring/en/index.html](http://www.who.int/immunization_standards/vaccine_quality/pqs_e6_temp_monitoring/en/index.html)