FoneAstra: Making Mobile Phones Smarter

Rohit Chaudhri, Gaetano Borriello

Department of Computer Science and Engineering
University of Washington
Seattle, WA [USA]
{rohitc, gaetano}@cs.washington.edu

William Thies

Technology for Emerging Markets
Microsoft Research India
Bangalore, [INDIA]
thies@microsoft.com

ABSTRACT

FoneAstra is a low-cost, programmable device that extends capabilities of mobile phones. We show how our device extends the functionality of non-programmable, low-tier mobile phones that are most prevalent amongst people from low-income groups in developing regions. FoneAstra enables interesting mobile applications in a variety of domains ranging from participatory sensing to remote monitoring to healthcare. The paper describes several applications that we are currently developing. As a first sample application, we demonstrate location tracking capability on low-tier mobile phones that are not programmable and do not have GPS capability. In prototype quantities, FoneAstra costs only $15.

1. INTRODUCTION

Hardware miniaturization and advances in embedded systems have resulted in the evolution of mobile phones from being voice and SMS communication devices to smart devices that have several sensors (motion, location etc) and communications and computing capabilities close to the level of desktop computers. In developing countries mobile phones are the de-facto computing platform because they run on battery power, can be easily secured on the person, and have a limited but familiar user interface. Interesting mobile applications have been developed that leverage the phone's computing and communication capabilities ([10] [11], [18]). As phone prices drop, more users are able to buy smarter, more capable phones. However, surveys indicate that low-tier phones (especially Nokia phones) are the most prevalent amongst people from low-income groups in both rural and urban areas of developing countries ([19], [20]). While these phones (e.g. Nokia 1110, 1200, 1650, etc.) are cheap and affordable, they have minimal hardware and support only basic telephony. Most lack a flexible programming environment as is available for Android and Windows Mobile on mid to high-end phones. This severely restricts the services that can be delivered to the more numerous users of low-tier phones.

In this paper, we present a low-cost system we are developing to extend capabilities of low-tier mobile phones. We call our system FoneAstra (in Sanskrit, Astra means weapon or tool). Since software-only extensibility is not possible, our approach combines hardware and software to achieve the additional functionality. FoneAstra is a programmable accessory based on an ARM7 processor that connects to a phone through the phone's data port and communicates to it using a standard serial protocol. The serial interface exposes some cellular capabilities such as sending/receiving SMS, making/receiving phone calls, querying the cellular network for information, etc. We use the device for application specific computation and leverage the phone for user input/output and cellular communications. In prototype quantities, the device costs only $15. The low-cost of the device combined with its extensibility (both in enabling new hardware and software to be added to low-tier phones), makes it a versatile platform that we envision being used in many different application areas such as participatory sensing, healthcare, remote monitoring, etc. Section 4 discusses in more detail the relationship of FoneAstra to prior work.

Even though non-programmable, low-tier phones are currently more prevalent in low-income settings, it is reasonable to expect that this will change over the next several years and many more people will have access to more capable and programmable phones. It is important to understand that our approach is not restricted to low-tier phones. In fact, moving forward, we envision FoneAstra to play a critical role in extending capabilities for smart phones as well. Sensors are increasingly being integrated onto smart phones (e.g. sensors for light, motion, proximity, etc). However, there are applications for which the required sensor(s) might not be available in a phone available on the mass market. With the limited, standardized connectivity options available on smart phones (USB or Bluetooth), it might not be possible to connect arbitrary sensors to the phone. However, microcontrollers including ARM7, offer several connectivity options for interfacing with hardware (UART, I2C, SPI etc.). A future version of the FoneAstra will have a standard interface for connecting to smart phones (using USB so as not to require a separate power source as for a Bluetooth device) and will have capabilities to easily host arbitrary sensors. Healthcare in developing regions is one particular domain that we are very interested in exploring for this future work [7].

The principal contributions of the work presented here are:

- Extending the capabilities of low-tier mobile phones in a practical, low-cost manner to enable new classes of distributed sensing applications.
- Adding wider and more scalable deployment possibilities to existing approaches such as wireless sensor networks and participatory sensing.
- Demonstrate that the approach presented will also be useful for extending high-end mobile phones.

The rest of the paper is structured as follows. Section 2 gives an overview of the FoneAstra platform. Applications currently being developed and planned for the near-term future are discussed in section 3. We compare our system to related work in section 4. Section 5 concludes the paper.

2. SYSTEM OVERVIEW

FoneAstra is based on NXP's LPC2148 processor [24], an ARM7–

NSDR '10, June 15th, 2010, San Francisco, CA.USA.

based microcontroller. The ARM7 processor was chosen because of its low-cost, support for low-power modes, and rich I/O interfaces (UART, SPI, I2C, USB and GPIO). We added a memory card interface as well to enable the device to have persistent storage of data. One of the 2 UARTs available on the LPC2148 is used for connecting to mobile phone serial ports, while the other UART is used for firmware upgrades and application debugging. [2] provides details of the platform, including the schematics.

The serial interface exposed by low-tier mobile phones is the key to extending the phone's capabilities. GSM and CDMA phones typically implement an AT command interface ([25], [26]) that is accessible over a serial link. Additionally, if AT commands are not implemented on the phone, vendors usually implement their own proprietary protocols. For instance, low-tier Nokia phones implement the FBUS (Fast-Bus) [27] serial protocol instead of the standard AT command-set. These serial interfaces expose a rich set of capabilities including being able to query phone information such as the phone's IMEI identifier and phone number, sending and receiving SMS messages, and initiating and answering phone calls programmatically, over the serial link. This enables FoneAstra to communicate with other user's phones as well as services hosted in the "cloud". Cellular information such as the current cell tower-ID and signal strength is also accessible over this interface; this information can be used to build location-aware services. Key presses on the phone generate DTMF tones; these can be captured on the device by using a DTMF decoder. This enables the use of key presses for user interactions with FoneAstra. Since FoneAstra can also be battery-powered, it can function independently when not connected to a phone. In this "disconnected" mode of operation, FoneAstra can be used to aggregate data in the field and then, when it is later connected to a phone, use the newly available connection to upload the data to a remote server. In the current implementation, FoneAstra implements a subset of the FBUS protocol to communicate with low-tier Nokia phones. The supported phones include currently shipping models like the 1209, 1661/2 etc, as well as older models like the 1110 and 1200.

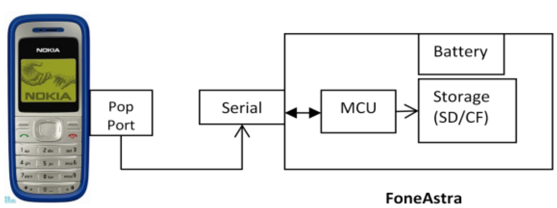

Figure 1: Basic FoneAstra Architecture

We now discuss a few architectural instantiations of our system. Figure 1 shows the basic architecture of the system in which FoneAstra is connected to a mobile phone over the data port (this connector is called "Pop Port" on low-tier Nokia phones). The phone's serial command interface is accessible over this link. Figure 2 shows the basic system enhanced with audio capabilities. The device in this figure has a DTMF decoder connected to the processor. This enables it to accept and interpret numeric keypad input from the phone. When the user is on the phone's home screen, each numeric key press on the phone generates a unique DTMF tone. In addition to being transmitted over the cellular channel these tones are also available on the phone's audio port. Hence, as shown in the figure, if the phone's audio port is connected to FoneAstra, it would then be able to capture and decode the key presses from the phone. Additionally, FoneAstra can be connected to a headset enabling it to send audio prompts to the user. This capability would be especially useful for illiterate/semi-literate users.

Figure 3 shows the system further enhanced with sensing capabilities. Sensors are shown connected to FoneAstra over the I/O peripheral interfaces available on the LPC2148. The configuration of Figures 1, 2, and 3 highlight different levels of capabilities possible with the same basic approach.

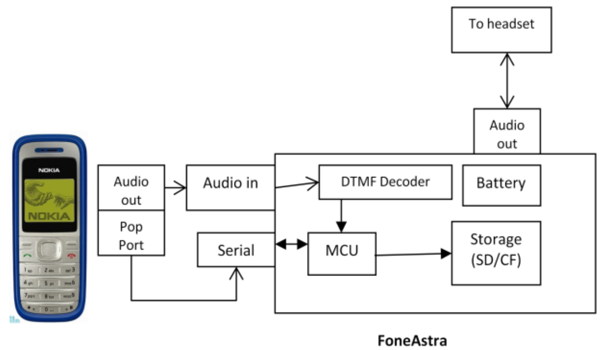

Figure 2: FoneAstra Enhanced with Audio Capabilities

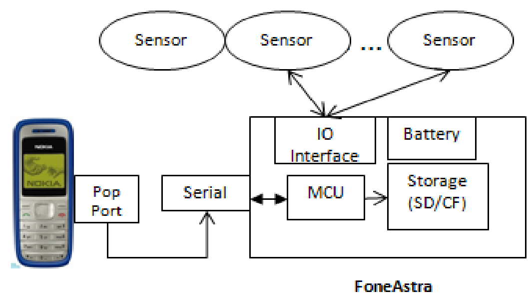

Figure 3: FoneAstra Enhanced with Sensing Capabilities

Figure 4 shows FoneAstra connected to a Nokia 1650 phone. This configuration adheres to the architecture depicted in Figure 1.

We envision 2 different kinds of deployments of our system (cf. section 3 for applications). In one class of deployments, FoneAstra would be used as an extension module for phones that are carried by people. For instance, this could be used for tracking the location of users or act as a platform to enable urban sensing via mobile users. Since humans are involved in such deployments; usability and form factor of the accessory will require special attention. We plan to provide an integrated packaging via a simple phone clip-on that includes FoneAstra, making the system more usable/wearable. In another class of deployments, the system (phone + FoneAstra) will act as a programmable communications gateway for existing enterprise systems (e.g. asset tracking, cold chain monitoring, WSNs, etc).

Our system offers some distinct advantages over currently available alternatives. The mobile phone ecosystem is already established and growing rapidly. By creating a small, low-cost plug-in for existing phones and cellular connections, we will be able to deliver value added services to mobile phone users and other systems that require cellular connectivity. For instance, within the WSN and urban sensing communities, systems have been proposed that leverage the cellular network for communicating with backend servers (cf. section 4). However, these systems either require a programmable mobile phone or a custom cellular modem module, often making the system cost prohibitive. Surveying the web yields that cellular modems cost around $100 [22]. Our system enables similar capabilities at a much lower cost, without requiring *any changes* to existing phones. When FoneAstra is manufactured at a larger scale, we estimate the total cost of the system to be under $30, including the low-tier phone. Additionally, since FoneAstra is based on a programmable platform, we are able to remotely reconfigure the system over the cellular network thus making the application easily upgradeable. Adding similar capabilities to a system based on a custom cellular modem further increases the cost of the system.

Figure 4: FoneAstra Connected to a Nokia 1650

Table 1 compares some relevant capabilities of our system with generic mid-tier J2ME and high-end smart phones (e.g. iPhone, NexusOne etc). It shows that with these capabilities, our platform has characteristics comparable to those of a smart phone at a much lower price point, while providing an extensible interface for connecting sensors. This makes it an attractive platform to build applications that require special-purpose sensing along with computing and communication capabilities (cf. section 3). While J2ME phones offer a rich set of APIs and have deep market penetration, FoneAstra offers some capabilities that are not present in J2ME. First, J2ME offers limited/variable support for daemon operation, i.e., running programs in the background for continuous data collection. The J2ME standard has no provision for daemon operation, though some phone vendors have proprietary extensions to enable it. Due to this limitation, it is not possible to track a user's day-to-day usage of the phone (e.g. numbers dialed on the keypad) using J2ME. A second limitation of J2ME is performance; Java may remain unsuitable for real-time operations such as audio and speech processing. Campo et al [16] compare the image processing performance of a J2ME midlet to that of a native Symbian application. They show that the native application performs significantly better than the midlet. Finally, most of the capabilities in the table require midlets to be signed due to J2ME's security model. While this is possible for advanced developers, it represents a practical hurdle for many projects.

Table 1 Capability Comparison

Capability	Low Tier	LowTier+ FoneAstra	MidTier (J2ME)	High Tier
Send/recv SMS programmatically	No	Yes	Yes (signed)	Yes
Make/recv calls programmatically	No	Yes	Yes (make calls)	Yes
GPS	No	No	Optional (signed)	Optional
Location sensing by cell tower-ID	No	Yes	Optional (signed)	Yes
Log numeric key presses	No	Yes	Limited	Yes
Interfacing to external sensors	None	Extensible (UART, I2C, SPI, ADC, GPIO)	Serial, BT, USB (signed)	Serial, BT, USB
Extensible storage	No	Yes	Yes	Yes
Computational power	-	Fast	Slow	Fast
Daemon operation	No	Yes	Limited	Yes
Cost	$20	$30	>=$50	>=$200

3. APPLICATIONS

Having a general purpose, programmable extension to a mobile phone creates several interesting application possibilities. We discuss a few classes of applications in this section. Some of these are works in progress, while others are planned for future work.

3.1 Location Tracking

As mentioned previously, FoneAstra has access to the connected phone's cell tower-IDs over the serial interface. We have used this information to create a (relatively low-resolution) location tracking application for the Nokia 1650 phone. The 1650 is a low-tier phone that does not have a GPS unit or a programmable run-time system to aid in collecting location data. In this application, FoneAstra is programmed to periodically query the phone's current cell tower-ID. This information is written into a file on the on-board memory card. We process the data offline to get the geographical coordinates of the cell towers using a Google Web Service [28], which gives an approximation of the trip route of the user. Figure 5 shows a location trace collected using this system from a 10 mile round trip that originated at Microsoft's office building in Bangalore. While this is a sample application to demonstrate the capabilities of the system, a more realistic application might upload cell tower-IDs to a server in real time via SMS messages. The server could be part of a location tracking system deployed by an enterprise. The tracked entities could be users of mobile phones or assets in a supply chain. Operation ASHA [13] and LabourNet [5], two non-governmental organizations (NGO) operating in India, are interested in piloting our system to track their workforce who use predominantly low-tier mobile phones.

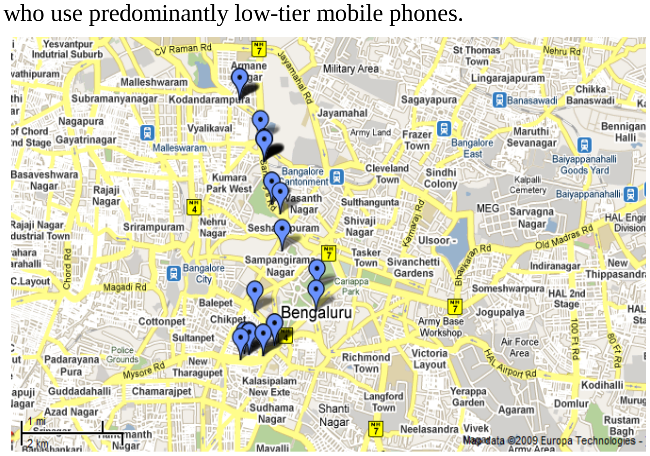

Figure 5: Location Trace Obtained from FoneAstra Connected to a Nokia 1650

3.1.1 *Tracking in Disconnected Mode*

A slightly different flavor of tracking is enabled if the accessory is time-shared across multiple phones. For instance, in a shipment tracking application, the accessory could be attached to the cargo. As the cargo crosses predetermined checkpoints, the staff could plug in their low-tier phones to the accessory, which would query the phone's cell tower-ID and IMEI. This information could either be logged to the on-board memory card or sent to a remote server via an SMS. Such a system could also be used by NGOs to deploy a low-cost attendance tracking system. In this application, FoneAstra would be installed at a fixed location (perhaps at the NGO's office) and the staff would plug in their phones into the accessory to log their attendance.

3.2 Cold Chain Monitoring

PATH [15] is a Seattle-based NGO that works to deliver healthcare services to the underprivileged in developing countries. They state: "*Temperature and geography can get in the way of delivering lifesaving vaccines to children in remote areas.*" One of their projects involves improving the vaccine delivery system or *cold-chain,* which consists of refrigerators and cold storage units to hold vaccines as they are moved from the national store to health facilities. Since vaccines have strict temperature requirements, there is a need to remotely monitor the temperature of these storage units. Towards this, PATH is developing a custom, SMS-accessible, temperature monitoring device called SmartConnect [21].

In collaboration with PATH, we are currently working on integrating temperature sensors onto FoneAstra for monitoring cold-chains. This will be an alternate solution to SmartConnect that leverages our device used in conjunction with a commodity, low-tier mobile phone. This application will use a FoneAstra device to monitor up to 5 co-located cold-chain units. Under normal conditions, the device will periodically send SMS messages containing temperature data. These messages will be aggregated and logged at a backend server. Additionally, if temperatures deviate from a predefined normal range, the device will send warning messages. We expect to do a pilot deployment of this system during the summer of 2010.

3.3 Phone Usage Logging

Phone usage information (e.g. phone calls, SMS, location) is accessible to FoneAstra over the serial interface. This can be used to track phone usage and users' travel patterns. This information is of great interest to researchers who study the use of mobile phones in developing countries. Current methods have relied on in-person interviews and retrieving call logs from carriers [1]. Using an automated, electronic logger for capturing usage information will facilitate new insights into how low-tier phones are being used. For instance, it would be possible to determine if correlations exist between phone usage and the location at which it is being used. With sufficiently large datasets it might even be possible to determine if correlations exist between phone usage and human dynamics or socio-economic status of users (similar to the Reality Mining project [8].)

3.4 Hosting SMS-Accessible Services

Using FoneAstra in conjunction with a low-tier phone enables users to create and publish services that are accessible over SMS. For instance, the device could interact with a user to aggregate information about crop prices and make this information available to buyers over SMS. Essentially, turning any phone into an SMS service that responds to requests automatically (without a separate server).

3.5 Healthcare

The Change group [3] at the University of Washington is developing information and communication technologies (ICT) to help with the problems of under-served communities around the world. MDPhone [7], one of the projects within the group, envisions creating a smart phone-based medical platform to enable telemedicine and basic diagnostic assistance. Moving forward; FoneAstra will play a critical role in the MDPhone project as the platform that provides the required sensing capabilities. A future version of FoneAstra will have a standard interface for connecting to smart phones (using USB so as not to require a separate power source as for a Bluetooth device) and will have capabilities to easily host sensors that enable physiological measurement/monitoring and point-of-care diagnostics. This evolution will demonstrate the utility of FoneAstra in a smart phone-based ecosystem as well.

4. RELATED WORK

FoneAstra complements existing systems and approaches in different research areas. In this section, we discuss our system in the context of this related work.

4.1 Wireless Sensor Networks

Wireless Sensor Networks (WSN) have several low-cost sensing nodes and aggregator nodes that act as base stations. A base station is more expensive compared to the sensing nodes because it has more computational capabilities and typically has a long-haul connection to communicate with a backend server. Since FoneAstra provides low-cost computing and communications, with the appropriate RF interface, it could easily be deployed as part of a WSN and act as a networked base station. Basha et al [9] deployed a WSN to enable early warning of floods in Honduras. They used a custom RF solution for communications. Kim et al [4] deployed a WSN on the Golden Gate Bridge in San Francisco to monitor its structural health and used a laptop for the base station. In both of these deployments (and other similar deployments), FoneAstra could be programmed appropriately to be the base station of the WSN.

4.2 Sensing on Smart Phones

Currently shipping smart phones already have several sensors integrated into the device. Researchers have built novel platforms to further enhance the sensing capabilities of these devices and enable them to interact with existing WSNs and Body Sensor Networks (BSN).

4.2.1 Extensions to Smart Phones

Systems like [6] and [17] have been proposed to enhance sensing capabilities of smart phones. However, these systems have required modifications to the phone's operating system. Our system, on the other hand, enables similar capabilities on low-cost phones, without requiring any changes to the phone itself. While we do not see any overlap here with our work on extending low-cost phones, for our future work with smart phones, we will leverage lessons learned from these systems.

4.2.2 Urban Sensing

People-centric Urban Sensing has generated much interest within the sensor networking community in recent years [14]. The NSMARTS [12] project is building mobile phone centric platforms to enable large-scale urban sensing. However, their current work relies on programmable mobile phones that have Bluetooth.

Although this is a promising approach, it limits wide scale deployments because it does not include users who own low-tier phones. We believe our system can fill this gap. This will be important especially for deployments in developing countries.

4.3 Cellular Modems for SMS-based services

Custom cellular modems are used in the industry to enable fleet management, cargo tracking and remote service delivery [23], much like the applications enabled by our system. However, since these modems are not manufactured at the same scale as mobile phones, their cost tends to be high (~$100) even for very basic models that do not offer a programmable run-time environment. Our solution can offer similar services (and much more because we use a general purpose programmable platform) at significantly lower costs, since we leverage commodity, low-cost mobile phones for communications.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented FoneAstra – an ARM7-based low-cost, programmable accessory that we have built to extend capabilities of mobile phones. While applications currently being developed target low-tier mobile phones, our future work will show that FoneAstra adds value to the smart phone ecosystem as well. In prototype quantities, our device costs only $15.

We have discussed how our system can add value to existing application areas like participatory sensing, remote monitoring, and healthcare. Moving forward, we will explore possibilities of integrating our system (phone + FoneAstra) with other elements of these application domains.

With the flexibility and extensibility offered by our system, we believe it is highly likely to further both ICTD research and practice. Our next crucial step is to deploy applications in the field in collaboration with our NGO partners.

6. ACKNOWLEDGMENTS

Initial platform development for FoneAstra happened while the first author was doing an internship at Microsoft Research, India. We would like to acknowledge MSRI, and especially Kentaro Toyama, for this opportunity and funding the platform development.

7. REFERENCES

[1] I. Medhi, S.N. Gautama, and K. Toyama, "A comparison of mobile money-transfer UIs for non-literate and semi-literate users," *Proceedings of the 27th international conference on Human factors in computing systems*, Boston, MA, USA: ACM, 2009, pp. 1741-1750.

[2] R. Chaudhri and K. Toyama, "A Programmable Accessory for Cell Phones", *MSR-TR-2009-178*.

[3] Change Group: *http://change.washington.edu*

[4] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Health monitoring of civil infrastructures using wireless sensor networks," *Proceedings of the 6th international conference on Information processing in sensor networks*, Cambridge, Massachusetts, USA: ACM, 2007, pp. 254-263.

[5] LabourNet: *http://www.labournet.in/*

[6] B. Priyantha, D. Lymberopoulos, and J. Liu, "LittleRock: Enabling Energy Efficient Continuous Sensing on Moble Phones", *MSR-TR-2010-14*.

[7] G. Borriello and W. Brunette, "MDphone: A Doctor's Tools in a Phone," *IN: Globicomp 2009: Taking Ubicomp Beyond Developed worlds. Workshop at the UbiComp 2009*, Orlando, FL. USA: 2009.

[8] Reality Mining: *http://reality.media.mit.edu/*

[9] E.A. Basha, S. Ravela, and D. Rus, "Model-based monitoring for early warning flood detection," *Proceedings of the 6th ACM conference on Embedded network sensor systems*, Raleigh, NC, USA: ACM, 2008, pp. 295-308.

[10] M-Pesa: *http://en.wikipedia.org/wiki/M-Pesa*

[11] Nokia Life Tools: *http://www.nokia.com/NOKIA_COM_1/Microsites/Entry_Event/phones/Nokia_Life_Tools_datasheet.pdf*

[12] R. Honicky, E.A. Brewer, E. Paulos, and R. White, "N-smarts: networked suite of mobile atmospheric real-time sensors," *Proceedings of the second ACM SIGCOMM workshop on Networked systems for developing regions*, Seattle, WA, USA: ACM, 2008, pp. 25-30.

[13] Operation ASHA: *http://s01.opasha.org/*

[14] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M.B. Srivastava, "Participatory sensing," *IN: WORKSHOP ON WORLD-SENSOR-WEB (WSW'06): MOBILE DEVICE CENTRIC SENSOR NETWORKS AND APPLICATIONS*, 2006, pp. 117--134.

[15] PATH: *http://www.path.org/*

[16] C. Campo, C. Navarrete, and C. Garcia-Rubio, "Performance Evaluation of J2ME And Symbian Applications in Smart Camera Phones," *2007 Digest of Technical Papers International Conference on Consumer Electronics*, Las Vegas, NV, USA: 2007, pp. 1-2.

[17] T. Pering, P. Zhang, R. Chaudhri, Y. Anokwa, and R. Want, "The PSI Board: Realizing a Phone-Centric Body Sensor Network," *4th International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2007)*, 2007, pp. 53-58.

[18] Txteagle: http://txteagle.com/

[19] E. Cutrell, Microsoft Research India, "Mobile Phone and SMS Survey for LaboutNet"

[20] Personal conversation with Cliff Schmidt, Founder, Literacy Bridge

[21] R. Anderson, E. Blantz, D. Lubinski, N. Rourke, M. Summer, K. Yousoufian, "SmartConnect: Last mile data connectivity for rural health clinics", *Submitted to NSDR 2010*.

[22] *http://www.roundsolutions.com/pdf/Price-Modules-GSM.pdf*

[23] http://www.aplicom.com/

[24] LPC2148: *http://www.nxp.com/#/pip/pip=[pip=LPC2141_42_44_46_48_4]|pp=[t=pip,i=LPC2141_42_44_46_48_4*

[25] GSM AT Commands: *http://www.3gpp.org/ftp/Specs/html-info/27007.htm*

[26] CDMA AT Commands: *http://www.forum.nokia.com/info/sw.nokia.com/id/537978cf-3542-41ea-a220-9a0bbbb83580/AT_Command_Set_For_Nokia_CDMA_Products_v1_1_en.pdf.html*

[27] FBUS: *http://en.wikipedia.org/wiki/FBus*

[28] Google Web Service: *http://www.google.com/glm/mmap*