The Mote is Dead. Long Live the Discarded Smartphone!

Geoffrey Challen, Scott Haseley, Anudipa Maiti, Anandatirtha Nandugudi,
Guru Prasad, Mukta Puri and Junfei Wang
Department of Computer Science and Engineering
University at Buffalo
{challen,shaseley,anudipam,ans25,gurupras,mpuri2,junfeiwa}@buffalo.edu

ABSTRACT

As the rapid pace of smartphone improvements drives consumer appetites for the latest and greatest devices, the hidden cost is millions of tons of e-waste containing hazardous chemicals that are difficult to dispose of safely. Studies show that smartphone users are replacing their devices every 18 months, almost *three times* faster than desktop computers [1, 3], producing millions of discarded smartphones each year that end up lying in desk drawers, buried in landfills, or shipped to third-world countries where they are burned to extract precious metals, a process that damages both the health of those involved and the environment.

Fortunately, the capabilities of discarded smartphones make them ideal for reuse. Instead of ending up in a landfill, a discarded smartphone could be integrated into a home security system or transformed into a health care device for the elderly. In this paper, we evaluate using discarded smartphones to replace traditional sensor network "motes". Compared with motes, discarded devices have many advantages: price, performance, connectivity, interfaces, and ease of programming. While the main question is whether their energy consumption is low enough to enable harvesting solutions to allow continuous operation, we present preliminary results indicating that this may be possible.

1. INTRODUCTION

Smartphone technologies are advancing rapidly, bringing power into users' pockets that is changing the way we live. The rapid rate at which consumers purchase new smartphones can be seen as primarily a response to the speed at which this technology is improving. Short device lifetimes, while unfortunate from a sustainability perspective, help support companies that build and sell smartphone hardware and software. Unfortunately smartphones, like most other electronics, are difficult to dispose of properly. Many end up unused in desk drawers, discarded in landfills, or shipped to poor countries where they are dangerously dismantled in an effort to extract precious metals.

HotMobile '14, February 26 - 27 2014, Santa Barbara, CA, USA

http://dx.doi.org/10.1145/2565585.2565598.

Given smartphones' current role in bringing about transformative technological change, it is hard to argue that consumers should hang on to outdated devices in the name of sustainability. Instead, we believe it will be more effective to focus on how to reuse the devices we currently discard. There are three reasons why the time is right for this effort. First, unlike previous generations of "feature phones", the smartphone market is coalescing around a small set of platforms, with this homogeneity reducing the burden of reusing discarded devices. Today, each phone in an electronics recycling bin runs a different OS; in three years, half of the smartphones in the same bin may run Android.

Second, current smartphones have an attractive feature set for many non-phone applications: size and power requirements facilitating easy deployment; microphones, cameras, and other sensors built-in; touch screens for interacting with users. And the volume at which they are produced combined with the rate at which consumers are replacing them produces an extremely competitive price point for discarded devices given their capabilities.

Finally, smartphones are well-integrated into existing communication infrastructures. They can transmit data using text messages, Wifi networks, and high-speed mobile communication technologies like 3G. If Wifi is available, no service plans are required to allow recycled smartphones to become part of the "Internet of Things". And with carriers increasingly interested in "machine-to-machine" applications [13], we expect to see increasing service flexibility allowing discarded devices to be cheaply connected to pervasive mobile cellular and data networks.

To provide an idea of the potential of discarded devices, the U.S. Environmental Protection Agency (EPA) estimates that 141 million mobile devices became ready for end-of-life management in 2009, of which only 11.7 million (8%) were collected for recycling [16]. The 129 million discarded phones are enough to place a phone in half of the registered vehicles in the United States [15].

In this paper, we investigate reusing smartphones sensor network "motes". Compared to motes, discarded smartphones have many advantages, which we outline in Section 3. And while power consumption is a concern and the discarded phone's major weakness, we show in Section 4 that a simple and unoptimized sense-and-send application running on a Nexus S phone can last over a week on a full battery charge, even while preserving the familiar and powerful Android programming environment. To begin, the next section reviews the current state of smartphone sustainability.

2. SUSTAINABILITY TODAY

We spoke to Sprint about their efforts in the area of smartphone sustainability. Sprint has been recognized as one of the greenest companies in the US by Newsweek's annual Green Rankings [17], and has ambitious goals for greening mobile devices. Sprint offers users credit for their old phones during the purchase of a new device, with the amount depending on the phone model and condition. They aim to recover 90% of their users phones at end-of-life by 2017, an aggressive target given today's industry average of recovering only 10% and Sprint's current rate of 44%. Sprint's goal may also be difficult to achieve because many users choose to retain their old phone as a spare device. And the user demand for new smartphones shows no sign of abating, with multiple carriers offering new plans tailored at users that want to replace their devices even more frequently.

When we asked Sprint specifically about what they do with devices that they repurchase, their answer focused on enabling reuse of smartphones as smartphones. After paying to test and, if necessary, refurbish the returned phone, they resell it as used to a second user, either in the US or overseas. Of the phones they have recovered from their buyback program, 80% can be rebranded and reused, 10% are desirable phones but not compatible with Sprint's network, and the final 10% are so broken that their only value is the $1.82 of gold they contain [7].

While creating a market for used smartphones is an appropriate first step, it ignores what happens when the second user returns the doubly-used but still functional device. As with other electronics, the value of smartphones drops extremely quickly. Sprint informed us that they were offering users only $22 for a Samsung Nexus S 4G in good condition, three years after it sold for $529 unlocked. After several iterations either one of two things will happen: the buyback price will be too low to incentivize the user to return the phone, or the phone will be old enough to not be attractive to users in the market for a smartphone.

If a phone is too old or broken to be reused, Sprint first manually disassembles the phones apart to recover valuable parts that can be reused again—plastics, glass, batteries—prior to sending the phones for recycling. Any part that cannot be reused is classified as e-waste. According to Sprint, 1,180 metric tons of e-waste was collected for recycling in 2010 [12]. The e-waste is recycled in recycling facilities such as Sipi Metals that process scrap non-ferrous or precious metals. At these facilities, the cell phones are first finely shredded and then smelted to extract the precious metals such as gold, silver, palladium, and copper. These metals are then captured into metal bars which are sold in the market to be reused in other products.

Sprint tries to ensure that all of its recycling partners are environmentally certified and that trans-boundary shipments of e-scrap from developed countries to underdeveloped countries is prevented. Nevertheless, studies have shown that a large amount of electronic waste continues to be shipped to poor countries without regulations protecting the workers that dismantle it or the environment [5].

3. MOTES V. DISCARDED PHONES

As part of determining whether discarded smartphones can replace sensor motes for some sensor networking applications, we perform an attribute-by-attribute comparison of the two options. We compare the Epic mote, a common sensor node platform, with one discarded phone, the Samsung Nexus S 4G model that was used for our preliminary feasibility study. Table 1 presents numbers used in the discussion below, drawn from both datasheets and experiments. As an additional comparison point, we also include numbers for the Raspberry Pi Model B, since this is a popular, cheap and powerful single-board computer. Our comparison considers multiple aspects involved in deploying sensor networks, including cost, difficulty, capabilities, and power provisioning.

3.1 Cost and Availability

We would expect the rapid turnover and high production volumes for consumer devices like smartphones to cause their prices to start low and fall quickly, and our data shows that this is the case. The Samsung Nexus S was released in 2010 at $529 unlocked, but only three years later Sprint offers customers $22 as a trade-in value for a returned device in good condition, making it cheaper than both the Raspberry Pi Model B ($35) and the Epic Mote ($69), despite offering many more features as detailed below.

While both the Raspberry Pi and Epic can be purchased new in unlimited quantities, one concern about the use of discarded devices is availability. However, as stated earlier the EPA estimates that 141 million mobile devices were discarded in 2009. In several years, even if only half are Android devices and only 10% of those are in working condition, that still leaves millions available for reuse. **Advantage: discarded smartphone.**

3.2 Packaging and Human Interface

Both the Epic Mote and the Raspberry Pi are shipped as bare circuit boards and lack any human interface. In contrast, the discarded Nexus S comes packaged in plastic and features a familiar touch-screen interface. Even if they are only used during deployment debugging or maintenance, a screen provides a powerful maintainability advantage over sensor nodes, which must communicate by flashing their LEDs. **Advantage: discarded smartphone.**

3.3 Sensors and Sensor Interface

The Epic Mote is designed as a sensor platform, and so includes multiple sensor interfaces while integrating no onboard sensors. The Raspberry Pi is not designed for sensing applications but still exposes GPIO ports and a USB interface allowing external sensors to be attached.

On the Nexus S and other discarded phones the situation is more complicated. On the plus side, many phones include multiple integrated sensors, although the sensor suite is designed around smartphone and mobile computing applications and its composition varies across devices. In addition, the increasingly-ubiquitous μUSB port creates the possibility of adding additional sensors, or the even more ubiquitous audio jack can be "hijacked" [10] for this purpose.

Unfortunately, our experience is that many smartphone USB controllers are not designed for this purpose. If they can enter USB host mode at all, their power consumption when acting as a host is prohibitive, possibly because they are expecting to receive power when the USB cable is connected. We are investigating this problem in more detail and hope to be able to develop software solutions on devices where the hardware has the necessary capabilities.

An alternative approach to extending a device's default

	Epic Mote	Raspberry Pi Model B	Discarded Nexus S 4G
Cost	$69	$35	$22[1]
Microprocessor	4/8 MHz MSP 430	800 MHz[2] ARM1176JZ-F	1 GHz[2] ARM Cortex A8
Memory	10 KB	512 MB	512 MB
Storage	2 MB	SD card sold separately[3]	16 GB
Wireless Connectivity	802.15.4	None	SMS, 3G data, Wifi (802.11 b/g/n), 4G WiMax
Packaging	Open circuit board	Open circuit board	Plastic case
Human Interface	None	LEDs, HDMI output	480 x 800 pixel touch screen
Onboard Sensors	None	None	Location (GPS), accelerometer, gyroscope, proximity, compass, GPS, camera, light
Sensor Interface	8 ADC channels, 8 GPIO ports, OneWire	2 USB ports, GPIO	1 microUSB port
Operating System	TinyOS	Linux	Linux
Programming	NesC	Python, C	Java, Android
Sleep Power	27 μW	500 mW[4]	4.2 mW
Battery	None	None	Li-Ion 1500 mAh ($5 replacement)

[1] Customer buyback price quoted by Sprint for a smartphone in good condition.
[2] Processor is capable of dynamic voltage and frequency scaling (DVFS).
[3] The cheapest 16 GB SD card we could locate on NewEgg cost $9, increasing the total cost of the Raspberry Pi to $44.
[4] With the onboard hub removed [18].

Table 1: Comparison between potential sensing platforms. The discarded Nexus S 4G smartphone has multiple advantages compared with both the Epic mote and the Raspberry Pi Model B.

sensor suite is to use a tiered approach. Instead of connecting a single additional sensor via USB, we could connect multiple sensors to an additional piece of hardware that is in turn connected to the Android device. In practice, this can be accomplished using the Android Open Accessory protocol (AOA) which provides a way to integrate custom hardware accessories with Android devices over USB. A suitable candidate for a daughterboard in this tiered approach is the Arduino ADK board. There are a wide range of sensors available for the Arduino platform such as temperature, humidity, light, and water sensors. Using the AOA protocol, it is possible to access data from these sensors on an Android device over USB.

In summary, while smartphones were not designed to be sensor platforms, they include built-in sensors and can be extended to control a variety of others, albeit with the overheads of including a second device. In contrast, motes lack the built-in sensors but are designed to integrate with external ones easily. **Advantage: even.**

3.4 Programming Environment

Both the TinyOS [8] and Contiki [6] sensor node programming frameworks are notoriously difficult to learn and use, a challenge that the sensor network research community has been trying to address for years. In contrast, there are over 1 million applications hosted on the Google Play store, evidence of a large and growing Android developer community built on pre-existing familiarity with the Java programming language. We anticipate that the capabilities of these developers can be harnessed in developing applications that reuse discarded phones as sensors and in other ways. **Advantage: discarded smartphone.**

3.5 Capabilities

When comparing the core device capabilities, the discarded phone has the clear advantage, with a three orders-of-magnitude faster processor, four orders-of-magnitude more memory, and an one order-of-magnitude more storage. The discarded phone actually has more memory than the Epic has Flash storage, allowing it to cache information in RAM and avoid the energy overhead of writing to or reading from Flash. While Flash has the benefit of persisting across failures, motes tiny memory sizes typically cause data that could be storage in RAM to be move to Flash simply due to lack of space. The Raspberry Pi's core specifications are similar to the discarded smartphone, with the exception that storage is not included and must be added separately at additional cost. **Advantage: discarded smartphone.**

3.6 Connectivity

With their multiple connections to widely-deployed networks, connectivity is another important area where the discarded Nexus S 4G has a distinct advantage. The amount of the world not served by cellular, mobile data, and Wifi networks is shrinking, meaning that a smartphones can be deployed almost anywhere without provisioning network infrastructure. In contrast, the Epic mote has only an 802.15.4 low power, low bitrate radio, relying on a standard that has yet to be widely deployed. While some have called for a Wifi-scale deployment of 802.15.4 [21], we consider it unlikely that anyone will build a new networking infrastructure at that scale to support low-power sensor nodes. In addition, when data aggregation is possible Wifi becomes more much more energy-efficient than 802.15.4 [2].

The ad-hoc routing and networking required when nodes lack direct connections to the infrastructure takes a severe toll on the real lifetime of deployed sensor networks, with an energy breakdown compiled by Klues et. al [9] of a sensor node showing that periodic idle listening reduced lifetime by an order-of-magnitude. Specialized networks also complicate deployment logistics, with several experience papers [20, 4] documenting the difficulties establishing and maintaining 802.15.4 networks and bridges with the outside world. As an anecdote, one of the authors spent several days of a short deployment working to establish a point-to-point serial link between a volcano-monitoring sensor network and a monitoring site located 8 km away [19]. At one point, as he and a colleague were struggling high on the volcano to diagnose a networking problem, the colleague's phone rang with a call from movers in Berkeley, CA, some 6000 km away.

Utilizing these ubiquitous networks, however, requires carriers willing to tailor cost-effective plans allowing sharing across large numbers of nodes. In our conversations with Sprint, they expressed interest preliminary willingness in offering such plans. Interestingly, the "all you can eat" data plans offered for years by US cellular carriers may have discouraged this use model, since without a way to meter data carriers may fear that larger numbers of devices mean more data. Now multiple carriers offer metered data plans that can be shared across multiple devices, indicating that they have technology in place to implement this option.

Overall, by avoiding ad-hoc networking and tapping into the most widely-deployed networking infrastructures available, the discarded phone is better-connected than the mote. **Advantage: discarded smartphone.**

3.7 Power Consumption

As the table shows, the idle power consumption of the mote is two orders-of-magnitude lower than the discarded phone, unsurprising given that motes are designed to be energy-efficient. The Raspberry Pi's much higher power consumption reflects its intended use as a powered device. **Advantage: mote.**

4. PRELIMINARY RESULTS

To explore the potential to transform our discarded Nexus S 4G smartphones into low-power sensors, the authors divided into two teams for a lifetime programming competition. Each team was provided five discarded Nexus S 4G phones and given two weeks to write a program that recorded battery and light levels every 15 minutes and transmitted them to a server over Wifi. The goal was to implement a sensing application that would last as long as possible, while maintaining data delivery to the server. A gap of over two hours in the data values as observed by the other team rendered the node as dead, regardless of the amount of energy it had reported, with the two hour delay chosen to represent the potential requirements of a somewhat delay-tolerant application.

As we established in the previous section, motes are designed to provide extremely low energy consumption, whereas smartphones are designed around daily charging cycles. Thus, it is not our intention to claim that discarded phones will ever achieve the multi-year lifetimes promised by motes, no matter how carefully they are programmed—indeed, the idle current of deep-sleep mode alone will exhaust the battery in only two months. Instead, our goal is to establish whether and how easily we could reduce the energy usage to a point where energy-harvesting solutions, such as solar panels, could potentially allow continuous operation in an outdoor setting (discussed in section 5.1). This would allow discarded phones to replace motes for many of the applications originally considered for sensor networks, such as bridge [11], habitat [14], and volcano [19] monitoring.

Broadly speaking the teams explored two different options with important implications for reuse: starting with a stock AOSP platform build and the familiar Android API, or using a super-minimal Tiny Android build that discards most of the platform components. We refer to the first approach as 'Sensor Android" and the second as "Tiny Sensor Android". From a programming perspective, we were hopeful that we could preserve the familiar Android environment that many programmers today are learning. But from an energy management perspective, we were worried that the platform contained features designed around short lifetimes that would prove unhelpful.

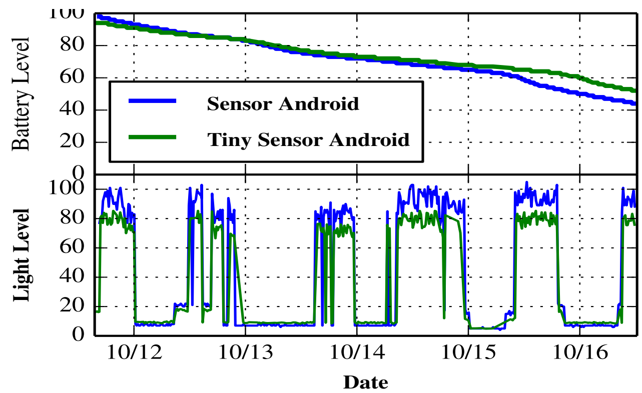

Figure 1: Lifetime results for two light sensing applications. Both approaches could achieve an 8–9 day lifetime.

Overall we were pleased to discover that both approaches were able to achieve 8–9 days of lifetime on a full battery, as shown in Figure 1, despite each approach suffering from a significant limitation affecting its performance. We believe that this lifetime may allow perpetual operation with commercially-available solar panels. We describe both approaches and our findings in more detail below.

4.1 Tiny Sensor Android

Tiny Android is a development option enabling a stripped-down build intended for testing new devices, and was not suitable for our application without modifications. Wifi drivers along with `dhcpcd` and `wpa-supplicant` had to be added to the build process. The only dependency introduced was `openssl`. The total package count was increased by 6, from 11 to 17. The implementation is primarily in C, consisting of 1500 lines-of-code (LOC). With the smartphone configured with a static IP address, during each loop iteration an alarm is set to wake up for the next sample. Then, the light sensor is enabled and a sample is obtained from the sensor as well as the battery. The Wifi interface is then enabled for a short period to send the message and disabled immediately after. The kernel is then asked to suspend the device until it is woken up by the alarm. Measurements showed that each iteration kept the device awake for approximately one second.

Figure 2 shows current output for one sense-and-send cycle of our sensing application on Tiny Sensor Android. While the sensing and transmission complete quickly, allowing the phone to rapidly return to idle, there was an extra 8 mA of current during the idle state which we have yet to explain. This experience demonstrates the difficulty of working with a stripped-down Tiny Android build, which does not fully enter sleep mode despite receiving commands identical to those provided by the full AOSP platform.

4.2 Sensor Android

The image for the Sensor Android build was built from the latest available AOSP code from the JRO03R branch.

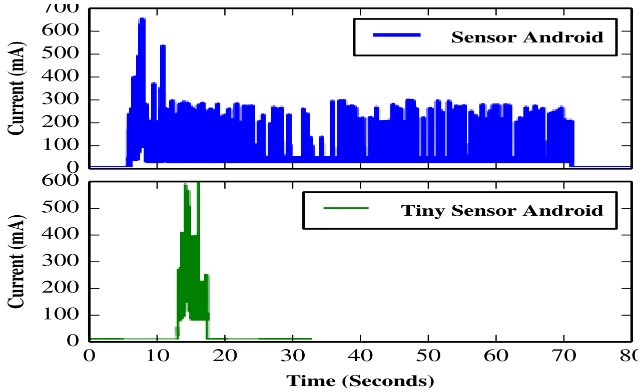

Figure 2: Current draw for a single sense-and-send cycle. Flaws in both approaches are visible. Tiny Sensor Android sleeps at a high 8 mA, whereas Sensor Android does not return to sleep for almost one minute.

No changes were made to the platform as we wanted to measure stock performance. However, we manually disabled all the default apps that run in the background like Browser, Calendar, and Email (an Android platform designed around programming sensors would not include these applications in the first place). The sensing application was written in 291 LOC using the standard Android API's for accessing sensors and using the network. Before the start of the experiment, we assigned a static IP to the Wifi interface and the phone was put into airplane mode to disable the cellular radio interface. The sensing application controlled the enabling and disabling of the Wifi radio interface for each sensing period.

Figure 2 shows the current draw for one sense-and-send cycle of our sensing application on Sensor Android. While it completes the sense-and-send operation as quickly as Tiny Sensor Android and sleeps in the lowest-power state, there is a 60 s delay before the phone reaches the idle state. On further investigation, we found that the *ConnectivityService* in the Android framework was keeping the phone awake to allow other radio interfaces to connect to a network after one of the interfaces is disconnected. For our sensing application this is not required and the platform code can be easily modified to disable this behavior.

4.3 Discussion

Overall we found our results encouraging, particularly the fact that an unmodified stock Android platform could equal the performance of the stripped-down Tiny Android build. We believe that this indicates that the Java programming framework used by Android is efficient enough to use to program discarded devices, even ones with energy constraints. In addition, our estimates indicate that fixing the long-tail problem on the Sensor Android approach will double its lifetime, from 8 days to almost 18.

Inspired by our results we are beginning the process of designing a dedicated Android build for sensor programming and discarded device reuse. Because many of these use cases consist primarily of a single application controlling the entire device, we can disable all extra included applications, as well as reduce resource limitations such as memory limits. Ideally we would like to remove all unnecessary platform components as well by examining the applications usage of the Android API, but our attempts to do this manually demonstrated that dependencies exist which must be carefully identified.

(a) Outdoor Monitoring (b) Urban Monitoring

Figure 3: Sensing applications. Two of the sensing applications we are currently investigating.

Given the wide variety of Android devices currently available on the market, there are certain concerns to consider with regard to the heterogeneity of discarded smartphones. The discarded Android devices will vary in OS version, types of sensors, and precision and accuracy of sensors. With the number of devices that we anticipate to be available, it is not unreasonable to think that heterogeneity can largely be avoided. However, even if it is not possible to find enough of the same desired device model for a particular deployment, variation in devices is still acceptable. Variations in OS version does not matter since the deployed version can be controlled. While Android does not require manufacturers to include specific sensors, the trend is to include more sensors, not less. Manufacturers have incentive to include more sensors to provide a better user experience and enable application developers to create more robust applications. The same incentive applies to sensor accuracy, but existing differences can be overcome by sensor calibration which should be done regardless to ensure accurate data.

5. USE CASES

The combination of low price and abundant capabilities makes discarded phones suitable for many other forms of reuse. We describe two sensing applications using discarded smartphones that we are currently investigating.

5.1 Outdoor Monitoring

Building rooftops can be equipped with discarded smartphones to monitor the environmental conditions. This in turn can be used to regulate the lighting and temperature inside the buildings to minimize the energy costs. Due to the unavailability of power sources on rooftops and the hazardous nature of power extension cables, it is unviable to deploy smartphones as monitoring instruments without any power supply to charge the battery. Based on our experiments, we think this can be made possible by using other alternate renewable energy sources like solar energy.

Figure 3(a) shows our current deployment. The solar panel trickle charges the smartphone battery. The application running in the smartphone periodically senses the temperature and light levels. It then transmits the sensed values to a central server every *transmit* interval, a configurable parameter, using 3G.

Our initial measurements indicate that a 1800mAH battery can be fully charged in 25 hours in heavily shadowed areas using the solar panel shown in Figure 3(a). Such a setup would provide enough energy for a self sustaining monitoring application. The total cost for the deployment was $67 excluding the cost for the smartphone and data services. This deployment will also give us insight into the effort required in maintaining a long-term smartphone sensor deployment.

5.2 Urban Monitoring

Cars manufactured post 1996 are equipped to provide On-Board Diagnostics(OBD) data following the OBD-II specification. With OBD-II, we can extract rich sensor data from the sensors deployed in the cars. With this data, we can design number of applications like urban traffic monitoring, pollution monitoring, etc. In order to read the OBD-II data, we need to instrument the car with a OBD reader that has Bluetooth capability. We use the smartphone in the car to control the OBD reader to retrieve the sensor data from the car's sensors. Once we have the data, we can either transmit it immediately or store it and transmit at a later time based on the application needs. The total cost for such a monitoring setup is around $15, excluding the cost for the smartphone and connectivity charges.

While OBD data can be collected only when the car's engine is running, the smartphone's sensors can be used to collect data at all times. Furthermore, Since the smartphone is in the car, it can be powered using the car's power outlets. This gives us the ability to also use the more energy hungry sensors like the camera or GPS to design smarter urban monitoring applications.

6. CONCLUSIONS

To conclude, we believe that the millions of discarded smartphones represent a significant opportunity, and can be used in many cases as replacements for sensor nodes in our effort to better instrument the world around us. Despite not being designed for ultra-low power consumption, smartphones can be operated efficiently enough to be able to operate continuously while harvesting energy. We are beginning outdoor experiments with commodity solar panels and plan to use our nodes to gather data to support the Neon citizen science project. Through intelligent reuse, we can turn techno trash into treasure.

Acknowledgments

Philippe Bonnet and Aslak Johansen provided helpful feedback on this paper. We thank Darren Beck and Blake Hosmer for discussing Sprint's sustainability efforts with us.

7. REFERENCES

[1] BIT-TECH . Most computers replaced after 4.5 years . http://www.bit-tech.net/news/hardware/2011/05/12/most-computers-replaced-after-4-5-years/1.

[2] DIGIKEY . Comparing Low-Power Wireless Technologies . http://www.digikey.com/us/en/techzone/wireless/resources/articles/comparing-low-power-wireless.html.

[3] DIGITAL TECHNOLOGY: THE TECH PORTAL . Trends in Smartphone Turnover . http://www.technology-digital.com/gadgets/-by-tim-kridel.

[4] BARRENETXEA, G., INGELREST, F., SCHAEFER, G., AND VETTERLI, M. The hitchhiker's guide to successful wireless sensor network deployments. In Proc. ACM SenSys (2008).

[5] COALITION, E. T. Responsible Recycling vs Global Dumping . http://www.electronicstakeback.com/global-e-waste-dumping/.

[6] DUNKELS, A., GRONVALL, B., AND VOIGT, T. Contiki: A Lightweight and Flexible Operating System for Tiny Networked Sensors. In Proc. First IEEE Workshop on Embedded Networked Sensors (EmNetS) (Tampa, FL, November 2004).

[7] GOLDMAN, D. Your dead iPhone is a gold mine—literally. http://money.cnn.com/2011/10/13/technology/iphone_trade_in/.

[8] HILL, J., SZEWCZYK, R., WOO, A., HOLLAR, S., CULLER, D. E., AND PISTER, K. S. J. System architecture directions for networked sensors. In Proc. the 9th International Conference on Architectural Support for Programming Languages and Operating Systems (Boston, MA, USA, Nov. 2000), pp. 93–104.

[9] KLUES, K., HANDZISKI, V., LU, C., WOLISZ, A., CULLER, D., GAY, D., AND LEVIS, P. Integrating concurrency control and energy management in device drivers. In Proc. the 12st Annual SOSP (2007), ACM, pp. 251–264.

[10] KUO, Y.-S., VERMA, S., SCHMID, T., AND DUTTA, P. Hijacking power and bandwidth from the mobile phone's audio interface. In Proceedings of the First ACM Symposium on Computing for Development (New York, NY, USA, 2010), ACM DEV '10, ACM, pp. 24:1–24:10.

[11] PAKZAD, S. N., KIM, S., FENVES, G. L., GLASER, S. D., CULLER, D. E., AND DEMMEL, J. W. Multi-purpose wireless accelerometers for civil infrastructure monitoring. In Proc. 5th International Workshop on Structural Health Monitoring (IWSHM 2005) (Stanford, CA, September 2005).

[12] SPRINT. Sprint Electronics Stewardship Policy. http://www.sprint.com/responsibility/ouroperations/docs/ElectronicsStewardshipPolicy.pdf.

[13] SPRINT. Sprint M2M Solutions. http://m2m.sprint.com/.

[14] SZEWCZYK, R., MAINWARING, A., POLASTRE, J., AND CULLER, D. An analysis of a large scale habitat monitoring application. In Proc. ACM SenSys (2004).

[15] UNITED STATES DEPARATMENT OF TRANSPORTATION. Number of U.S. Aircraft, Vehicles, Vessels, and Other Conveyances. http://www.rita.dot.gov/bts/sites/rita.dot.gov.bts/files/publications/national_transportation_statistics/html/table_01_11.html.

[16] UNITED STATES ENVIRONMENTAL PROTECTION AGENCY. Statistics on the Management of Used and End-of-Life Electronics. http://www.epa.gov/osw/conserve/materials/ecycling/manage.htm.

[17] GREENBIZ.COM. Newsweek's 2012 Green Rankings. http://www.greenbiz.com/blog/2012/10/20/newsweeks-2012-green-rankings-time-its-serious.

[18] WWW.RASPBERRYPI.ORG. RasPi power usage measurements. http://www.raspberrypi.org/phpBB3/viewtopic.php?f=63&t=6050&start=50.

[19] WERNER-ALLEN, G., DAWSON-HAGGERTY, S., AND WELSH, M. Lance: Optimizing high-resolution signal collection in wireless sensor networks. In Proc. ACM SenSys (Raleigh, NC, USA, 2008).

[20] WERNER-ALLEN, G., LORINCZ, K., JOHNSON, J., LEES, J., AND WELSH, M. Fidelity and yield in a volcano monitoring sensor network. In Proc. USENIX OSDI (Seattle, WA, 2006).

[21] YERVA, L., CAMPBELL, B., BANSAL, A., SCHMID, T., AND DUTTA, P. Grafting energy-harvesting leaves onto the sensornet tree. In Proc. ACM IPSN (New York, NY, USA, 2012), ACM, pp. 197–208.