Towards a Wireless Sensor Network Platform for the Internet of Things

Sprouts WSN Platform

Ahmad El Kouche
Telecommunications Research Lab, School of Computing
Queen's University
Kingston, Ontario, Canada, K7L3N6
Email: elkouche@cs.queensu.ca

Abstract—**This paper describes a WSN platform architecture uniquely designed and implemented for the Internet of Things (IoTs). The paper elaborates on all the architectural design decisions and challenges across the three major divisions of the platform, that is, the middleware, hardware, and network layer. The result of this research is a unique WSN platform, *Sprouts*, which is rugged, cost effective, versatile, open source, and multi-standard WSN platform that offers a step forward towards interoperable WSN platforms for the IoTs. Sprouts' architecture leverages state of the art technologies in hardware and network standards and builds upon our module-oriented DREAMS middleware architecture. Sprouts presents a much needed new approach that is different than the de-facto MSP430/AVR and Zigbee-based WSNs, and we discusses the reasons behind the necessary changes to meet the needs of IoT. Sprouts was tested in the harsh industrial environment of the Oil-Sands and showcased at the Ontario Centre of Excellence (OCE) Discovery of 2011.**

Keywords- WSN; Platform; Internet of Things; Architecture; Middleware; Network; Embedded Systems; Industrial; Energy Harvesting; Remote Trigger; RFID; Applications; Oil Sands

I. INTRODUCTION

The internet of things (IoTs) is the massive deployment of trillions of low cost wireless internet protocol (IP)-based sensor nodes to identify and monitor every object, or *thing*, around us. The conglomeration of the diverse assortment of sensing entities will bring a new perspective to the way we interact with the environment around us on multiple levels including social, environmental, cultural, business, etc. [1]. Every device capable of connecting to the internet, such as smart-phones, laptops, tablets, desktops, routers, internet radios, internet TVs, etc., requires a unique IP address. The current widely deployed IP revision is IPv4, which has a 32-bit address-space, or approximately 4.3 billion unique IP addresses, a number much smaller than the world population today. The 32-bit address-space of IPv4 has already been exhausted and succeeded by a sixth revision, IPv6, with a much wider address-space. IPv6 plays a very important role in the IoTs due to its very wide 128-bit address-space capable of addressing over 3.4×10^{38} unique addresses. With trillions of uniquely identifiable sensor nodes, and an address space larger than IPv4, an adaptation of IPv6 for WSN, known as 6LowPAN, becomes necessary [2].

This paper addresses inherent requirements and challenges for implementing a unique WSN platform for the IoTs with the current state-of-the-art technologies. In addition, we discuss necessary aspects of the platform and the decision making stages in building the Sprouts platform, which is most valuable for those seeking to do the same. Furthermore, the paper describes the application of Sprouts platform in the harsh industrial environments of the Oil-Sands in Fort McMurry, Canada. Sprouts' advantages include: operation in harsh environments, physical ruggedness, small dimensions, water proof, ultra low power, low cost, network and hardware standard compliant, ease of use, and conveniently customizable to easily meet various application requirements for maximum deployment flexibility.

The paper is organized as follows: Section II investigates widely used WSN platforms and discusses why a new platform architecture approach is needed for the IoTs. Section III discusses the general architecture of the Sprouts platform. Section IV compares DASH7, Zigbee, and BLE and discusses which network is most suitable for an IoT-based WSN platform. Section V discusses energy harvesting and remote wakeup operations of Sprouts. Section VI compares state of the art ARM Cortex M3 microcontrollers. Section VII builds upon our previous work with DREAMS middleware architecture for WSNs. Section VIII describes Sprouts' application in the Oil-Sands. Finally, Section IX concludes our work on Sprouts.

II. RELATED WORK

Most of the commercially available WSN platforms in use today are very identical in terms of their hardware architecture [3], which is either based on the 16bit MSP430 microcontroller unit (MCU) or the 8bit ATmega128 MCU, in combination with an 802.15.4 Zigbee based network architecture, such as IRIS, Mica2, and MicaZ, TelosB, Tmote Sky, and EPIC [4]. The popular use of the MSP430 and ATmega128 is due to several factors including: ULP energy consumption, community support, open source compilers based on GNU-GCC, and TinyOS support, which has constrained many of us to design our WSN platforms around these two MCUs due to difficulties in porting TinyOS to other MCU architectures [5]. Therefore, we attempt to break this limitation with Sprouts platform by using an ARM Cortex M3 (ARM-C M3), which also provides the same advantages as MSP430 and ATmega128, in addition to industry-standard 32bit ARM core architecture, vendor-independent hardware abstraction layer (HAL), known as CMSIS (Cortex Microcontroller Software Interface Standard), and GNU-GCC compiler for 32-bit ARM-C MCUs [7]. Section VI elaborates more on this topic.

Although Sprouts platform can support a Zigbee network, we have opted to deviate from the norm Zigbee network associated with IEEE 802.15.4 standard, due to its memory intensive stack, complexity, and associated cost increase. Instead, Sprouts adopts the new Bluetooth Low Energy (BLE) IEEE 802.15.1, due to its service discovery architecture most suitable for service oriented applications in the IoTs [1], low power, and low memory footprint of the network stack, which

This project is funded by Natural Sciences and Engineering Research Council of Canada (NSERC).

is embedded in the nRF8001 radio. Thus, Sprouts achieves lower implementation complexity and lower associated cost by using the nRF8001 transceiver. In Table I, we compare the prominent WSN platforms commercially available, actively supported by the WSN community [3]. As seen in Table I, Sprouts is uniquely different than other platforms and offers a much needed fresh-start architecture for the IoTs. Sun Spot and Imote2 use ARM cores that were designed for high speed mobile phones, thus, not intended for ULP-based WSNs.

TABLE I. COMPARISON BETWEEN SPROUTS AND OTHER PLATFORMS

Platform Name	*MCU*	*Core*	*Size (mm)*	*Network*		*Battery Power*	*Remote Wakeup*	*ULP MCU*	*EH*
				Radio	*w/s*[a]				
Sprouts	EFM32G 230F128	ARM v7	23x10 Cylind.	BLE nRF8001	Yes	20mm CoinBatt.	ULP RF	Yes	RF + VEH
IRIS	ATmega 128	AVR 8-bit	58x32 x7	Zigbee RF230	No	2 x AA	No	Yes	No
MicaZ	ATmega 128	AVR 8 bit	58x32 x7	Zigbee CC2420	No	2 x AA	No	Yes	No
TelosB	MSP430 F1611	RISC 16bit	65x31 x6	Zigbee CC2420	No	2 x AA	No	Yes	No
SunSpot	ARM 920T	ARM v4	64x38 x25	Zigbee CC2420	No	Lith.Ion 3.7 V	No	No	No
IMote2	XScale PXA271	ARM v5	36x48 x9	Zigbee CC2420	No	3 x AA	No	No	No

a. w/s: Network stack is embedded in the radio, which does not require the MCU to implement it.

III. SPROUTS GENERAL SYSTEM ARCHITECTURE

The work presented in this paper focuses on the low cost deployable source node aspect of the Sprouts WSN platform. The middleware architecture of Sprouts is discussed in section VII. The sensor node is designed for maximum modularity using interchangeable hardware layers as seen in Figure 1. The modular layers of the platform are described as follows:

1) Magnetic Layer: an optional magnetic attachment layer important for industrial applications and easy deployment.

2) Sensor Layer: in order to address a large number of applications, a sensor attachment bus is provided in the sensor layer, which serves as a dasiy-chain I^2C bus, UART, or SPI.

3) CPU and RF Layer: a mandaory layer for any sensor node as it contains the BLE radio, the remote wakeup circuit, and the ARM cortex MCU.

4) Custom Layer: this is a package contained and optional customizable layer dependent on the application.

5) Harvesting Layer: this is the energy harvesting layer capable of exclusively powering the entire node. The Sprouts platform currently uses RF and vibrational energy harvesting.

6) Backup Battery: an optional ML-2020 battery is recharged when excesive energy harvesting levels are reached.

7) Antenna: a small low cost PCB patch antenna which also serves as the cover for the platform package, and allows the node to be mounted on metal.

8) Packaging: a low cost cyledrical-aluminum tube-type measuring 23mm in diameter. Various cylendrical metals and hights can be used depending on the number of layers used.

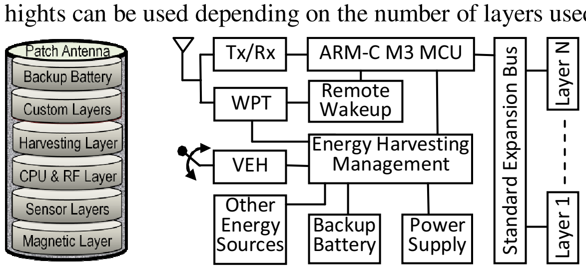

Figure 1. Sprouts modular platform Architecture

In addition to the source node architecture, Sprouts has a layered modular system architecture as seen in Figure 2. An example of the application layer is discussed in section VIII. The middleware is discussed in Section VII. The operating system resides on the source node and can support TinyOS, Contiki, or any real-time OS, which can be easily ported to other ARM Cortex based MCUs due to the standard use of a hardware abstraction layer CMSIS [7]. CMSIS standardizes the use of peripheral hardware across all ARM Cortex vendors, allowing interoperability and porting of software and operating systems across other WSN platforms. Energy harvesting management layer is implemented using CMSIS to monitor and control power consumption and energy harvesting of the vibration energy harvesting (VEH), wireless power transfer (WPT), remote wakeup trigger, and recharging a backup battery when excessive harvested energy is available. Although Sprouts' MCU has enough Flash memory (128KB) to support Zigbee, we opted for BLE using nRF8001, which contains the network stack, as discussed in section IV. Adaptation support for IPv6 is underway for BLE. Finally, we use CMSIS to support a common standard bus interface to manage the different attachable layers described earlier.

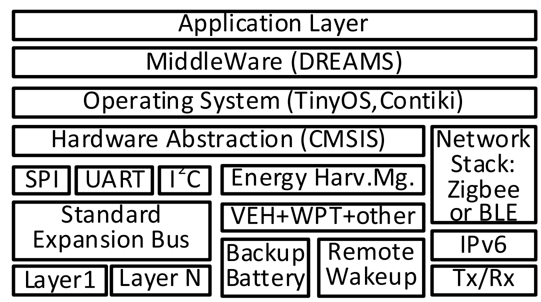

Figure 2. Sprouts general system and layered architecture overview

Depending on the application, some layers may be omitted, such that one of the simplest configurations of the platform would contain the sensor layer, CPU and RF Layer, backup battery, and the patch antenna as seen in Figure 3 C. Sprouts platform is very rugged and compact (23x10mm) to enable various application domains and the tagging of small objects.

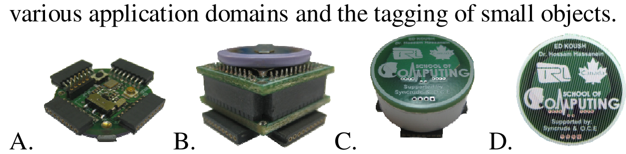

Figure 3. Sprouts sensor platform: A. Four sensor bus ports. B. Stackable C. Miniature metallic packaging D. Mount-on-metal patch antenna

IV. BLUETOOTH LOW ENERGY, ZIGBEE, AND DASH7

The lack of hardware architecture standards in WSNs is greatly made up for in organized, standard, and open source network architectures. We compare the most notable networks that are fundamentally different for the IoT, that is, Zigbee, Bluetooth Low Energy (BLE), and Dash7, as seen in Table II.

BLE is a new standard released by Bluetooth (BT) Special Interest Group (SIG) in December of 2009. BLE is the special feature of BT v4.0, which is BT v3.0 with BLE. BT v3.0 release was skipped for BT v4.0 [6], thus, offering BLE a head-start into the mass market of smart-phones, laptops, tablets, etc. Similar to previous BT radios, BT 4.0 radio can be implemented on the same WiFi radio in smart-phones. Therefore, we expect BLE to have immediate market readiness, allowing the quick realization of IoT applications. This head-start advantage will allow BT 4.0 smart-phones to become mobile collectors and routers for the IoT. In addition, BLE utilizes adaptive frequency hopping in 37 channels, which enables BLE to avoid interferences that would most likely occur in the crowded spectrum of IoT. Due to the importance of 6LoWPAN to the IoT, we assume that BT-SIG will bring IPv6 support for BLE devices in future improvements. For the reasons listed above, we expect BLE to become the dominant network standard, especially for smart-phone enabled WSNs.

The major competitor to BLE is Zigbee due to its maturity. Zigbee is a well known standard today and has been constantly improving since its initial release in 2004. The exceptional advantage of Zigbee is the mesh network topology support, which comes at the significant cost of increased network-stack footprint and code complexity. However, despite the initial release of Zigbee in 2004, Zigbee never made it into smart-phones, laptops, or tablets today. Therefore, Zigbee lacks the important connectivity with smart-phones, which will play a very important role as mobile collectors and routers for the IoT.

TABLE II. COMPARISON BETWEEN BLE, ZIGBEE, AND DASH7

Standard (name)	*Frequency Bands*	*Antenna Size* [a]	*Network topology*	*Data Rate*	*Spectrum Spreading*	*Modulation*	*Channels*
802.15.1 (BLE)	2.45 GHz	12.49cm	Star	1.0Mbps	FHSS	GFSK	40
802.15.4 (Zigbee)	868 MHz	34.53cm	Star, Tree, Mesh	100Kbps	DSSS	BPSK,ASK	1
	916 MHz	31.22cm		250Kbps		OQPSK	10
	2.45 GHz	12.49cm		250Kbps		OQPSK	16
ISO18000 (DASH7)	433 MHZ	69.23cm	Star	27.8Kbps	none	FSK	$1_{mode\ 1}$
				200Kbps		GFSK	$8_{mode\ 2}$

a. Based on the wavelength equation $\lambda = c * f$ (c is the speed of light, and f is the frequency).

DASH7 is an open standard technology based on part 7 specification of the International Standardization Organization (ISO) 18000, which defines wireless communication interfaces with RFID tags in several frequency bands including 433MHz (part 7, i.e. DASH7). Since DASH7 uses active RFID technology, the tags are not passive and require a battery to operate. With only 8 channels available, DASH7 would be easily overwhelmed in a crowded IoT environment. DASH7 might seem like a good candidate for the IoT due to its support for ISO18000 active RFIDs, and claims to have the longest communication range. While lower frequencies do travel further than higher frequencies, at their respective antenna sizes, shown in Table II, we argue that with limited size restrictions of nodes suitable for the IoT, operating in the 2.4GHz will yield a longer communication range due to better antenna directivity gain (DG) within small limited spaces.

In order to settle DASH7 claims on range and the common misconception about Bluetooth being a low range network, we have simulated three antennas using Ansoft Designer in the 433MHz, 916 MHz, and 2.4GHz ISM bands. To support small size sensor nodes for the IoTs, we assume a limited space of 30mm diameter or square area as close to a CR2032 size which allows for a tapered patch antenna at 916MHz. DASH7 at 433MHz cannot utilize any type of patch antenna within this space. Therefore, we implemented a loop-type antenna instead with a ground center to simulate the underlying components. Zigbee at 916MHz can establish a tapered patch antenna, while BLE and Zigbee at 2.4GHz can utilize a full patch antenna, as seen in Figure 4 A-C. The simulation results in Figure 5 show that with limited space restrictions, DG is greatly reduced from the nominal 0dBi. Similar DG results are discussed in [11].

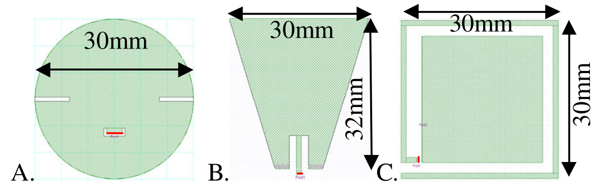

Figure 4. Antenna comparison: A. Circular patch 2.45GHz B. Trapezoidal short-ended patch 916MHz C. Loop antenna 433MHz

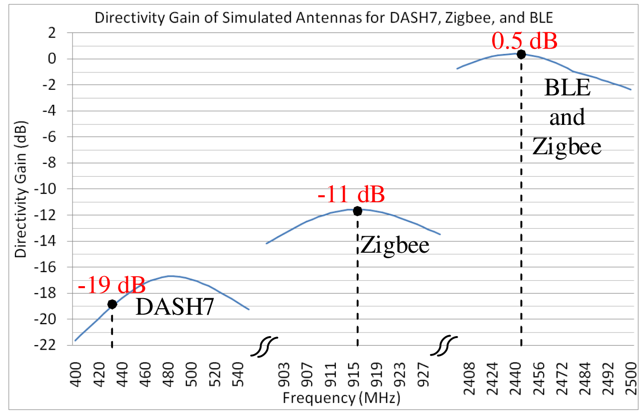

Figure 5. Directivity gain simulation results for DASH7, Zigbee, and BLE

Using Friis' free space transmission equation (1), we show how communication distance between two identical antennas is greatly reduced from the nominal 0dBi when sensor node size restrictions are applied from Figure 5 results. A sensitivity level of -85dBm is assumed for the sake of comparison, which can vary depending on transmission data-rates.

$$P_r(R,f) = P_t + 20.Log\left(\frac{c}{f}\right) - 20.Log(4.\pi.R) + G_t + G_r \quad (1)$$

The P_r is the received signal strength or the free space signal loss, P_t is the transmitter output power (assumed 0dBm), c is the speed of light, f is frequency, R is distance or range, G_t and G_r are the directivity gains of the transmitter and receiver antennas simultaneously.

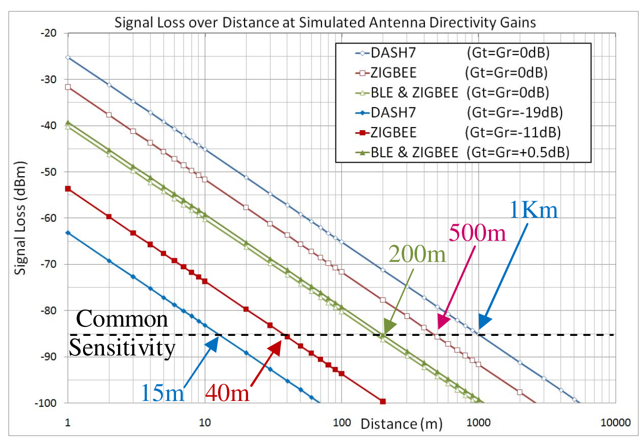

Figure 6. Signal loss over distance for nominal antenna gain (Gt=Gr=0dBm)

From Figure 6 above, we show that DASH7 range is reduced from 1Km to 15m and Zigbee 916MHz is reduced from 500m to 40m, when the space limitation is applied, while BLE and Zigbee operating in the 2.4GHz ISM band remain almost unaffected at 200m communication range.

V. MULTI-ENERGY HARVESTING WITH REMOTE WAKEUP

Energy harvesting is difficult to realize efficiently without increasing total sensor cost and size. Therefore, Sprouts utilizes a backup battery in addition to vibration energy harvesting (VEH) and wireless power transfer (WPT). To reduce cost, Sprouts utilizes a widely available low cost piezoelectric buzzer elements such as the 20mm AB2040B available for under 1.0$ each. The AB2040B is modified with lead weights to shift the physical resonant frequency as described in our previous paper [8]. This method yields a much cheaper and smaller alternative than the VEH sensors from Mide Technology ranging from $40 to $80 each. VEH is a mechanical process in nature, and the survivability of such an energy source over several years of constant mechanical movement is yet to be studied and evaluated for sustainability and long term reliability in the IoTs. Therefore, Sprouts also utilizes an RFID-like non-mechanical RF-to-DC WPT without having to increase sensor size. We use a pair of schottky diodes HSMS-2852 for RF-to-DC energy harvesting [10], and an ultra low power (0.4μA) comparator LTC1540 for remote wakeups. Another great advantage of using LTC1540 is the ability to use it as an ULP remote UART receiver using low 4200bps baud-rate ASK modulation generated by the interrogator, or smart-phone, to allow ULP addressing before enabling the RF radio.

VI. SPROUTS ARM CORTEX M3 BASED PLATFORM

The ultra low power market of deeply embedded systems have been long dominated by proprietary 8-bit and 16-bit architecture markets, such as MCUs provided by Atmel, Microchip, TI, Motorola, and many more. This has resulted in serious setbacks in practical development of WSN platforms due to the limited interoperability among various MCUs [6]. Sprouts uses an ARM-C M3 MCU which offers a much needed standard hardware architecture for price sensitive embedded markets such as WSNs [7]. The most important advantage of using ARM-C based MCUs is the ability to liberate developers and software code from being confined to one MCU vendor and proprietary core architectures [7]. A common 32-bit ARM MCU core architecture standard will enable competition among various vendors, reduction in prices, and greater innovation in the development community of WSNs without sacrificing code interoperability or code porting complexities. In other words, when we implement TinyOS using CMSIS for any ARM-C MCU, the operating system will work across all ARM-C MCUs from any vendor with minimal porting efforts.

We have surveyed state-of-the-art ARM-C MCUs available in the market today, and presented one MCU from each top leading ARM MCU vendors including: STMicroelectronics, Texas Instruments, NXP, ATMEL, and Energy Micro.

Based on Table III comparison we note the following:

1) LM3S1J16: some of the state of the art ARM-C M3 are not yet ultra low power optimized, and the LM3S1J16 is a good example. Although LM3S1J16 contains a battery-packed hibernation module, the LM3S1J16 consumes the most power.

2) ATSAM3S2BA: the disadvantages are the high power internal voltage regulator, high active current per MHz, low number of timers, and lack of SMBus standard.

3) EFM32G230F128: exceeds competitors with its ultra low power operation in all states. In addition, it contains several peripherals optimized for low energy (LE) operation such as 2 LE-UARTs, one LE-timer, 2 LE-comparators, LE-BOR, and more. The EFM32G230F128 has lower power consumption than the MSP430F5x MCUs, making it a good replacement candidate. Furthermore, Energy Micro provides an energy consumption monitoring software (EnergyAware Profiler), which visually debugs how much energy a certain code segment consumes. This is extremely important for ULP applications. Therefore, the EFM32G230F128 is a great candidate for an IoT-based WSN platform.

4) STM32L151RBT6: STMicroelectronics did a great job producing an MCU that is low power and low price. In addition, it contains the most communication interfaces making it the best value for the price. This chip is well suited for an IoT-based WSN platform and directly competes with EFM32G230F128 with slightly higher power consumption.

5) LPC1754FBD80: has the highest operating speed and a CAN controller. Has the same disadvantages as ATSAM3S2, lacks a low speed RC oscillator, and lacks a 64-pin package.

The STM32L151RBT6 and the EFM32G230F128 are both great candidates for an IoT-based WSN platform. The STM32L family is very low power and offers the best value for price. However, when it comes to ULP consumption and energy optimized peripherals, the EFM32G230F128 excels over competitors. Therefore, Sprouts uses EFM32G230F128 MCU to implement an ultra low power, cost efficient, energy harvesting, and widely applicable IoT-based WSN platform.

TABLE III. COMPARISON BETWEEN STATE-OF-THE-ART ARM CORTEX M3 MICROCONTROLLERS FOR WSN PLTAFORMS IN THE IOT

Vendor: Microcontroller [a]	*Max Clock Speed*	*Internal Oscillators*		*Memory*			*DMA Channels*	*Communication Interfaces*								*Analog Comparator*	*ADC*			*PWM Channels*	*Timers*			*AES (bit keys)*	*Voltage Range*	*Package*	*Price each (US) at a quantity of 100*	*Avg. Sleep Currents*				*Max I/O Leakage*
		Fixed Low Speed	*Tunable High Speed*	*Flash*	*SRAM*	*EPROM*		*UART*	*SPI*	*I^2C/SMBus*	*CAN*	*LIN*	*USB*	*Smart Card*	*Max GPIOs*		*Resolution*	*Channels*	*M Sample/s*		*General Timers*	*Sys Tick Timer*	*WDT+RTC*					*SRAM-ON*	*SRAM-OFF*	*RAM+RTC POR+BOD*	*Current / MHz at max speed*	
Atmel: ATSAM3S2BA [12]	64 MHz	32.768 KHz	4-12 MHz	128 KB	32 KB	-	23	4	3	2/0	-	-	✓	✓	47	1	12 bit	10	1.0	4	3 16bit	32 bit	✓	-	1.62-3.6V	64-QFN	$5.80	>3 µA	>1 µA	33 µA	561 µA	30 ηA
Energy Micro: EFM32G230F128 [13]	32 MHz	32.768 KHz	1-28 MHz	128 KB	16 KB	-	8	5	3	1/1	-	-	-	✓	56	2	12 bit	8	1.0	3	4 16bit	24 bit	✓	✓	1.8-3.8V	64-QFN	$5.70	0.6 µA	0.02 µA	0.9 µA	180 µA	25 ηA
STMicroelectronics: STM32L151RBT6[14]	32 MHz	37 KHz	64KHz-4.2MHz	128 KB	16 KB	4 KB	7	3	2	2/2	-	✓	✓	✓	51	2	12 bit	20	1.0	4	6 16bit	24 bit	✓	-	1.8-3.6V	64-LQFP	$4.81	0.6 µA	0.3 µA	4.5 µA	300 µA	50 ηA
Texas Instruments: LM3S1J16 [15]	50 MHz	30 KHz	16 MHz	128 KB	20 KB	-	32	2	2	2/2	-	✓	-	✓	33	2	10 bit	8	1.0	6	6 16bit	24 bit	✓	✓	2.25-3.0V	64-LQFP	$5.82	38 µA	26 µA	128 µA	1010 µA	2.0 µA
NXP LPC1754FBD80 [16]	100 MHz	-	4-100 MHz	128 KB	32 KB	-	8	4	1	2/0	✓	-	✓	-	52	-	12 bit	6	0.2	6	4 32bit	24 bit	✓	-	2.40-3.6V	80-LQFP	$6.11	30 µA	0.6 µA	31 µA	420 µA	40 ηA

a. All microcontrollers include the following features: power-on reset (POR), brownout detect (BOD), memory protection unit (MPU), -40º to 85ºC operation, and an internal temperature sensor

VII. SPROUT'S DYNAMICALLY RECONFIGURABLE ENERGY AWARE SOFTWARE (DREAMS) ARCHITECTURE

The DREAMS architecture [9] is our middleware architecture approach towards remotely reconfigurable and upgradable software modules for energy harvesting WSNs. The DREAMS architecture is a unique energy-harvesting aware middleware composed of interchangeable software modules. The DREAMS architecture simplifies middleware architecture implementation complexity, decreases faults associated with low energy sources, and increases design flexibility associated with remote upgrades for energy-harvesting WSNs. In [9], we discuss how DREAMS is uniquely different than other middleware approaches for energy-harvesting WSNs.

Furthermore, the module-based middleware architecture of DREAMS lends itself towards a graphical-based programming environment similar to that of LabView. Graphical based programming is an intuitive method of programming, and easier for non engineering users. The trillions of IP-based sensor nodes would be better managed if anyone and everyone could graphically program and update sensor nodes in the IoT.

VIII. OIL-SANDS APPLICATION OF SPROUTS PLATFORM

Sprouts platform is funded by an oil production company in Fort McMurry, Alberta. Therefore, Sprouts is also designed to operate in harsh and industrial environments supported by its compact and rugged packaging. The huge vibration screen, seen in Figure 7, filters large ores form the slurry sand flow. After approximately 2,000 hours of constant sand flow, the tungsten layer on the metal mesh is etched. This exposes the underlying metal, which risks the breakage of the ligament causing larger rocks to sift through and disturb the oil production process. Therefore, we use Sprouts to monitor the thickness of the vibration screen ligaments using sensor probes from the sensor layer, as described in [8].

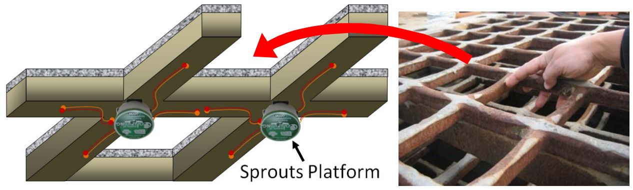

Figure 7. Sprouts application in the Oil-Sands. Sprouts platform is located underneath the intersection ligaments of the vibration screen.

Results of the thickness readings are collected and stored on a remote server accessible through any internet-enabled. Figure 8 shows the graphical results displayed on a smart phone, such as the normal operating thickness level (green color), a break in the ligament (red color), or a dangerously low thickness level (orange color). This application project proves the successful working architecture of Sprouts to reliably operating in harsh industrial environments. Therefore, Sprouts would be able to operate in similar or less stringent applications in the IoT.

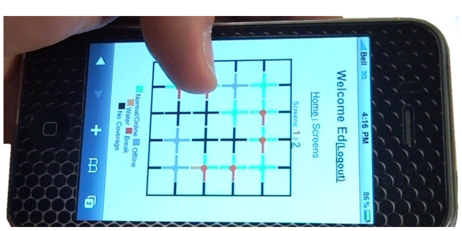

Figure 8. Graphical representation of the vibration screen status

IX. CONCLUSION

The Sprouts WSN platform represents a unique and new direction for WSNs in the IoT, and a much needed fresh start for WSNs in general. Sprouts platform abandons the outdated use of MSP430/AVR proprietary architecture for ARM Cortex M3 standard architecture and adopts BLE network standard as a better candidate for the IoTs, while optionally supporting Zigbee. Sprouts platform closely represents the operation of an RFID-like sensor node allowing for remote RF wakeups, wireless power transfer, and ultra low power addressing. In addition, Sprouts was tested in the harsh operating environments of the Oil-Sands to monitor huge vibration filter screens while utilizing energy harvesting from vibration to extend the lifetime of the sensor node. Sprouts uses DREAMS middleware architecture, which is a module-based architecture geared for graphical-based programming. Finally, we hope to encourage the use of Sprouts platform in universities working on WSNs research and applications for the Internet of Things.

ACKNOWLEDGMENT

We would like to acknowledge Prof. Hossam Hassanein for his excellent supervision. We would like to acknowledge the Natural Sciences and Engineering Research Council of Canada (NSERC) for funding this project. We would like to thank our colleagues for their support: Louai Al-Awami, Abdulmonem Rashwan, and Abdallah Almaaitah.

REFERENCES

[1] L. Atzori, A. Iera and G. Morabito, "The Internet of Things: A survey," Computer Networks, Vol. 54, No. 15, pp. 2787-2805, October 2010.

[2] A. Dunkels and J. P. Vasseur, "IP for Smart Objects," IPSO Alliance White Paper No. 1, Sept. 2008.

[3] J. Beutel, "Metrics for sensor network platforms," In ACM RealWSN'06, Uppsala, Sweden, June 2006.

[4] Diana Bri, Hugo Coll, Miguel Garcia, Jaime Lloret, "A Wireless IP Multisensor Deployment," Journal On Advances in Networks and Services, Vol. 3, Issue 1&2, pp. 14. 2010.

[5] C. Lynch, F. O'Reilly, "PIC-based TinyOS Implementation", presented at the 2004 IT&T Annual Conference, 2004

[6] Bluetooth SIG, Bluetooth Core Version 4.0, Volume 6, "Core System Package: Low Energy Controller," pp. 2165-2300, June 30 2010.

[7] J. Yiu. The Definitive Guide to the ARM Cortex-M3. Elsevier Newnes, March 2010.

[8] A. El Kouche, L. Al-Awami, H. Hassanein, K. Obaia, "WSN application in the harsh industrial environment of the oil sands," IWCMC, July 2011

[9] A. El Kouche, L Al-Awami, H. Hassanein,"Dynamically Reconfigurable Energy Aware Modular Software (DREAMS) Architecture for WSNs in Industrial Environments," Ambient Sys.Net.and Tech. (ANT), Aug 2011

[10] J. Ansari, D. Pankin, and P. Mahonen, "Radio-triggered wake-ups with addressing capabilities for extremely low power sensor network applications," International Journal of Wireless Information Networks, vol. 16, no. 3, pp. 118–130, 2009.

[11] K. Smith, "Antennas for Low Power Applications," RF Monolithics, http://www.rfm.com/corp/appdata/antenna.pdf

[12] Atmel SAM3S Microcontroller ATSAM3S2BA http://www.atmel.com/dyn/resources/prod_documents/6500s.pdf

[13] Energy Micro EFM32 Microcontroller EFM32G230F128 cdn.energymicro.com/dl/devices/pdf/d0005_efm32g230_datasheet.pdf

[14] STMicroelectronics STM32L Microcontroller STM32L151RBT6 http://www.st.com/internet/mcu/product/248821.jsp

[15] Texas Instruments Stellaris Microcontroller LM3S1J16 http://www.ti.com/lit/ds/symlink/lm3s1j16.pdf

[16] NXP LPC1700 Microcontroller LPC1754FBD80 www.nxp.com/documents/data_sheet/LPC1759_58_56_54_52_51.pdf