

## Chapter 12

---

# Energy Consumption Profile for Energy Harvested WSNs

---

T. V. Prabhakar, R Venkatesha Prasad, H. S. Jamadagni, and Ignas Niemegeers<sup>+</sup>

### Contents

12.1 Introduction.....	318
12.2 Energy Harvesting .....	318
12.2.1 Motivations for Energy Harvesting.....	319
12.2.2 Energy Harvesting: A Possible Solution .....	320
12.3 Energy Harvesting: Beyond the Solar Harvester—Is It a Viable Option?..	321
12.4 Storing Harvested Energy .....	324
12.4.1 Energy Harvesting System.....	325
12.4.2 Experimental Measurement .....	328
12.5 Energy Budgets: System and Network Operations .....	329
12.5.1 Energy Harvested Applications: Challenges.....	331
12.5.2 Storage and Retrieval of System State .....	335
12.5.3 Toward a Distributed Smart Application: Challenges .....	336
12.6 Summary .....	338
References .....	338

## 12.1 Introduction

Miniaturization and improvements in the communication and computation capabilities of devices is bringing about steep growth in wireless communication devices. The exploding number of devices comes with the immediate task of supplying them with energy. Thus, energy harvesting has become an important aspect. Energy that is harvested from sources, such as solar panels, vibration, and thermogenerators, provide varying instantaneous power. We focus on the performance of energy harvested embedded sensors and study the problems associated for applications that assume constant power harvesting. We begin the chapter by taking a simple environment-sensing application and show its baseline performance under the trivial assumption of constant power availability. We then propose simple mechanisms specific to harvested power-enabled wireless sensor nodes to constantly adjust their operational schedule based on the incoming energy profile. In the initial part, we begin with system-related node operations and further extend this to a network of wireless sensor nodes. Since energy availability can be different across the nodes in the network, even if similar harvesting is used, network setup and collaboration is a nontrivial task. At the same time, in the event of excess harvested energy, the collaboration between nodes is possible, which is exciting, but also challenging. Operations, such as sensing, computation, storage, and communication, are required to achieve the common goal for any sensor network.

The chapter introduces the example of a “smart application” assisted by a decision engine that transforms itself into an “energy-matched” application. The chapter also explains the architecture of such a decision engine. We provide a few performance results of such applications. The results are based on measurements using Crossbow’s IRIS motes [1] as well as custom motes running on solar energy. To accurately measure the energy consumption, we have done away with batteries; instead, we used low-leakage supercapacitors to store harvested energy. The last part of the chapter has a section with results of a “distributed smart application” where network-related information is fed as an additional input to provide networking-related decisions.

## 12.2 Energy Harvesting

Energy harvesting devices capture small amounts of energy over a long time from sources, such as ambient light, wind, vibration, linear motion, temperature differential, radio frequency (RF) energy, etc. Such harvested energy, when converted to an electric charge, can be stored in storage devices, such as batteries and supercapacitors. Although the modern definition of energy harvesting means conversion from one form of energy to electricity, this is not necessarily true in a broader context. For example, the well-known mechanical watches and wall clocks of the past have generated mechanical energy and stored it in a coiled spring. There was

no requirement for conversion into electricity. Moreover, while there are many harvesting sources, not all could be regarded as “energy for free” sources. For instance, energy scavenging from sources, such as waste heat from industrial plants, vibrations from machinery in large industrial manufacturing plants, temperature differentials in automobiles and aircrafts, are all examples of secondary sources of energy harvesting. One may regard them as sources where energy is made to work twice [2]. Other examples, some quite fanciful, are those gadgets that use human energy conversion to electricity. Energy generated from human action, such as running, walking, cycling, lifting, and pressing is commonly sighted. We might have seen many people, especially little kids, wearing shoes with inserts where a light glows for every step walked. The bicycle dynamo, batteryless winding radio, and the hand-cranked flashlight are other examples. Sometimes environmental energy conversion into electricity can happen in two steps. As an example, a wind belt created with a speed of 1 to 2 meters/second can generate a vibration. This vibration can be converted into electricity that will be sufficient to power a wireless sensor node. In our present context, we will examine conversion from one form of energy into electricity. We particularly concentrate on energy requirements to drive embedded communication devices used for environmental monitoring and control. We regard electricity as the purest form of energy and its efficient harvesting from the environment is necessary to provide relief from the ever-growing number of embedded devices used by humans. In a recent survey, it is predicted that by year 2017, the world’s 7 billion population will be strapped with 7 trillion embedded devices [3]. Thus, it is important to find other ways to power a large portion of those devices. Furthermore, in the context of this chapter, we limit our discussions on power generated to be several hundreds of milliwatts (mW), sufficient to drive low-power electronics used for embedded applications. Such embedded applications could be in healthcare, industrial monitoring and control, security, agriculture, structural monitoring, automotive, and infotainment.

### ***12.2.1 Motivations for Energy Harvesting***

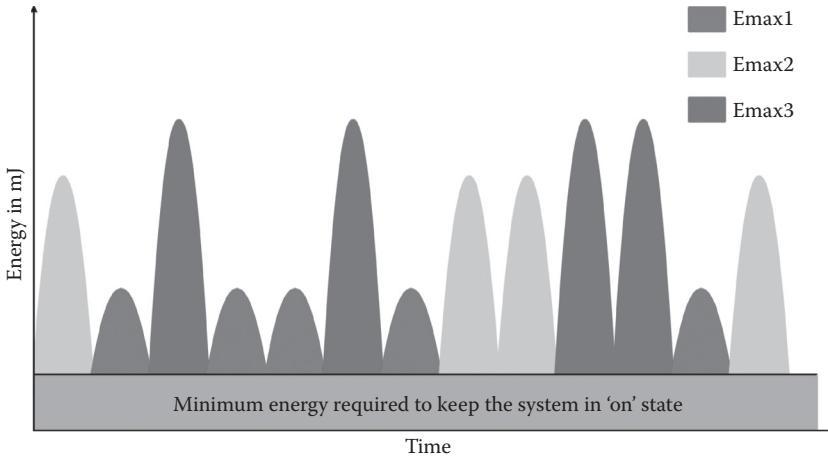
We have mentioned above about the trillion devices that are likely to be used by 2017 [3]. Let us assume for the time being that this estimate is true. We then will have to address a few issues. Firstly, what are the associated technologies to support such a large number? For instance, each device should have a unique address in a manner that it would be accessible from anywhere at anytime. While the address problem seems to be well geared toward large-scale IPv6 (Internet protocol, v. 6) deployments, the battery technology to support perennial or everlasting operation is unlikely to support all the devices. The well-known and commercially available lithium–manganese dioxide coin cell-2450 (use and discard) used in sensor devices costs a little over \$2 apiece and supports a nominal capacity of about 600 mA/h. They have a shelf life of about five years at room temperature. This figure doesn’t have credence since sensor nodes might be deployed in outdoor and, thus, subject

to large temperature swings. If we assume that 7 billion batteries are fitted to the devices, the effect of their replacement and disposal can be enormous—cost wise as well as environmentally. If one now replaces the power source with rechargeable lithium batteries, complicated electronics supporting overvoltage, short circuit protection and a cell temperature monitored design is required to prevent battery explosions. Moreover, we need power to charge the battery and this manual activity itself would be cumbersome. Such batteries also suffer from limited charge–discharge cycles not exceeding around 500 to a maximum of about 1,000 cycles. Again, cost of replacement of these batteries is an issue. Furthermore, a recent survey indicates that by year 2020, the European Union is committed to reducing the carbon emissions by 20 percent. It is expected that this figure is possible due to use of information communication technologies (ICTs). Also, over 17 percent of power generated worldwide is required to power ICTs [4]. In summary, even if we assume a very moderate number of devices per person (about 100), the explosion of devices by year 2017 and a need to equip each one with a battery is a serious challenge. Given this, the main focus of this chapter is how to guarantee a lifetime support from a sensor device.

Au: Do you consider 100 devices per person a "moderate" number? Maybe 10 would be more like it.

### 12.2.2 Energy Harvesting: A Possible Solution

To tackle the limitation of battery power and its replacement, and to guarantee a perennial operation of the sensor node, we envisage energy harvesting to ensure a self-powered node. A state-of-art survey in micro power generators for powering wireless sensor networks (WSNs) is covered in Borca-Tasciuc, et al. [5]. The initial part of the survey takes us through several mechanical to electrical energy converters, such as electromagnetic, piezoelectric, and electrostatic converters. The second half proposes a dielectrophoretic (DEP) actuation-based electrostatic harvester. Such capacitance-based recent harvesters, which switch their dielectric constant between air and liquid, are promising to offer at least four magnitudes increase in energy conversion. Thus, harvesting efficiency needs to be improved vastly to see its application complete. It is now reasonable to assume that if environmental energy (solar and wind) is perennial, then our sensor node also is perennially powered. Energy from vibrations has recently been promising. Researchers at Cornell [6] have created the “Piezo-tree” technology where low wind speed is sufficient to convert wind energy into vibration energy. Bio-energy generated due to pH imbalance in several parts of a tree and the surrounding soil is being explored by researchers from Massachusetts Institute of Technology [7]. The scavenged energy is sufficient to power a network of temperature and humidity sensors that are expected to aid in fire management. The motivation for this effort is replacement of batteries that is expensive and impractical. The project is titled Early Wildfire Alert Network (EWAN). The bio-energy harvester also is commercially available from commercial vendors now, such as Voltree. Figure 12.1 depicts a made up diagram of harvested energy by a device that varies with time. For example, the diurnal cycle of a solar energy incident on a



**Figure 12.1** Depiction of harvested energy varying with time. A minimum energy is required to keep the WSN node functional.

photovoltaic panel is maximal at noon during the summer months. Similarly, wind energy harvested at a given time depends on the wind speed. One may attribute similar variations with other sources, such as thermo energy generators where varying temperature differential creates a voltage that varies with time.

### 12.3 Energy Harvesting: Beyond the Solar Harvester—Is It a Viable Option?

Let us now look at the possibility of energy harvesting technology being a viable option to drive wireless sensor networks (WSNs). At this point, we do not consider solar energy harvester for two reasons. Firstly, solar energy harvesting is a definite possibility if the panel size is dimensioned appropriately. Secondly, researchers and vendors are concentrating on many indoor applications where solar energy is not a possible solution. Moreover, solar photovoltaic panels are expensive to manufacture.

To look beyond solar energy harvesting, let us see the energy harvesting capacities of other methods. Table 12.1 [8] shows the power densities from different energy harvesters. It shows that the thermoelectric generator provides more power density compared to the solar source. Vibration harvesters also are promising with power sufficient to drive embedded communication devices when an electric charge from the vibration harvester is collected in a supercapacitor for a sufficiently long time (15–20 min.).

Thus, let us now turn our attention to secondary energy harvesting sources and their suitability to drive embedded wireless communication devices. Energy harvesting sources are available from several vendors. Micropelt [9] offers a Seebeck

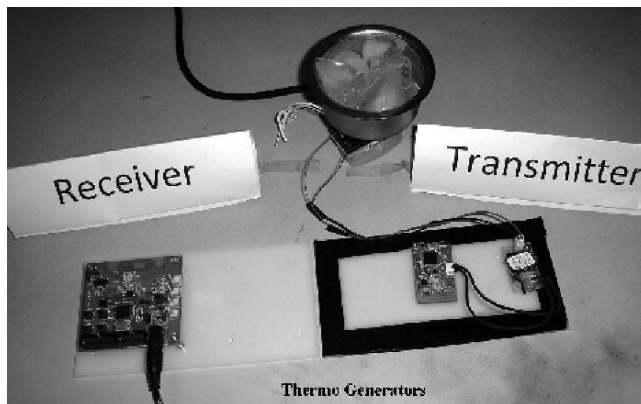
AU: Does "m" stand for minute here?

**Table 12.1 Power Densities of Different Energy Harvesters**

3 V flexible solar cell 1,000 lux	7 mW/kg
3 V flexible solar cell 10,000 lux	280 mW/kg
Vibration generator (60 Hz) $a = 0.24 \text{ m/sec}^2$	2.78 mW/kg
Vibration generator (60 Hz) $a = 0.98 \text{ m/sec}^2$	37 mW/kg
Thermoelectric generator $\Delta T = 10 \text{ K}$	8 W/kg
Thermoelectric generator $\Delta T = 40 \text{ K}$	131 W/kg
<i>Source:</i> Becker, et al. Power Management for Thermal Energy Harvesting in Aircrafts, from the <i>Proceedings of IEEE Sensors</i> , 2008.	

effect-based thermo energy generator (TEG) harvester. Seebeck effect is a phenomenon that generates a voltage when a temperature differential is applied across two dissimilar metals, such as tellurium and bismuth. The MPG-0751 is a popular thin film technology-based TEG. Figure 12.2 shows the setup of a TEG subjected to a temperature differential. One side of the semipackaged TEG is connected to a heater reaching 50°C. The other side of the TEG has an ice-filled container to develop the thermal gradient. The output power is conditioned with a low startup DC-DC converter and other maximum power point control functionality to harvest the maximum amount of energy. A DC-DC converter is a high efficient (greater than 90 percent) electronic circuit that has the ability to provide the required load voltage from very low harvested voltages. For example, the DC-DC converter (Texas Instruments' TPS 63031) can be configured to provide the required output voltage from an input voltage of 1.8 V. The energy is used

Au: Is this Texas Instruments, inc.?

**Figure 12.2 Thermo energy generation from Micropelt's MPG-0751.**

to charge a 15 mF supercapacitor. Once the output voltage across the capacitor crosses a critical value, the transmitter sends a data packet. The data packet has a typical length of about 128 bytes. The system shown in Figure 12.2 is built using a 16-bit RISC (reduced instruction set computing) microcontroller MSP430 and a radio transceiver CC2520 from Texas Instruments. The radio conforms to IEEE 802.15.4 standard that defines the lower layers (MAC and PHY layers) for low rate personal area networks (LR-**WPAN**)s.

Au: What is the W for in this abbrev.?

Vibration harvesters V21BL from Mide Vulture [10] are attractive because they have two piezo fibers packaged for serial or parallel connection. The parallel connection offers a peak-to-peak of 20 V. Figure 12.3 shows the fiber mounted on the box that houses a motor and cam for generating vibrations. To ensure that the system resonates effectively with the source, a suitable tipping mass is applied. The piezo fiber generates peak power at resonance.

Enocean [11] offers ECO-100 linear motion harvester for wireless switch applications. Figure 12.4 shows the 2.3 msec. pulse generated from a single operation

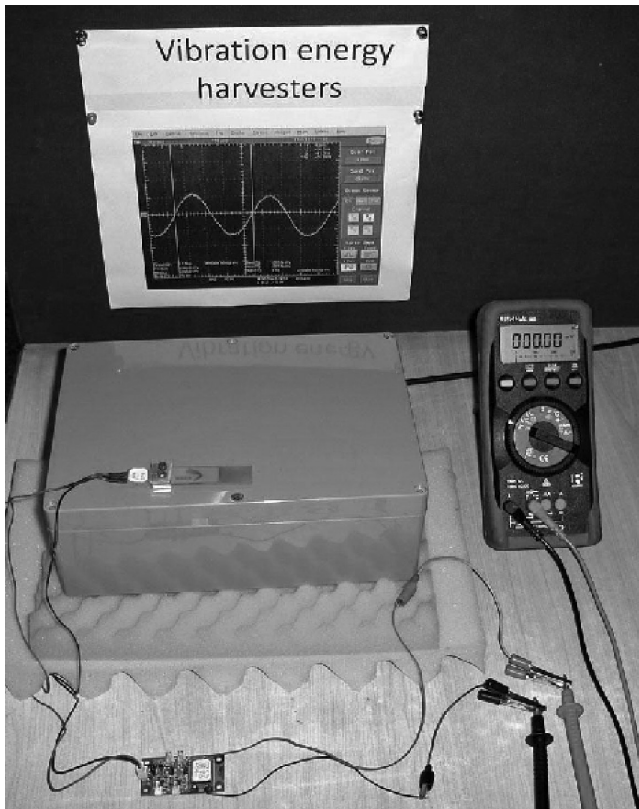
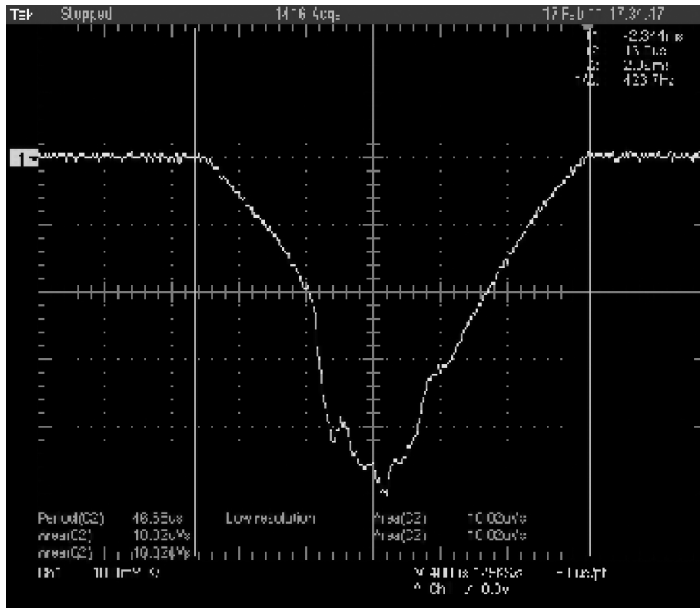


Figure 12.3 Vibration energy generation from Mide Vulture’s V21BL–piezo fiber.



**Figure 12.4** Current waveform from EnOcean's ECO-100 linear motion harvester.

of a mechanical switch. The current peak reaches 50 mA, although momentarily. The energy generated is found to be sufficient to transmit three data packets of size 50 bytes.

## 12.4 Storing Harvested Energy

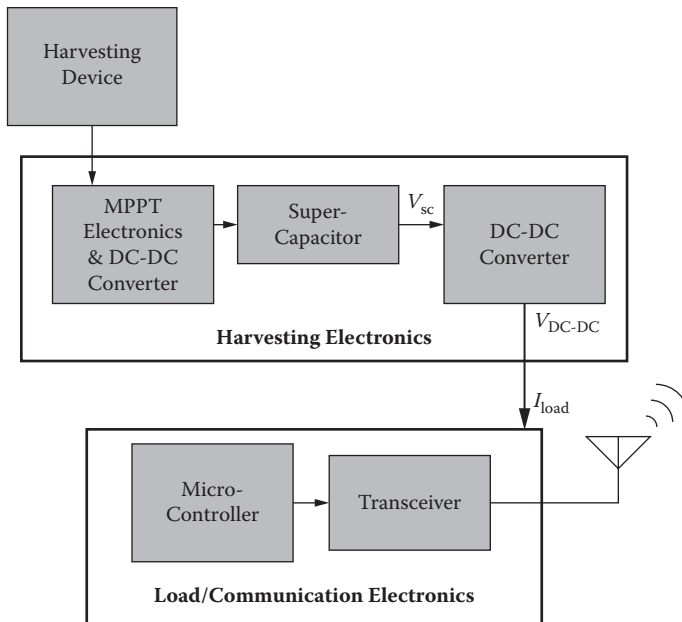
Having now increased the basket of energy harvesting sources beyond the solar harvester, the next problem is to explore the energy storage options. As we stated earlier, one of the biggest drawback of battery storage is the number of charge–discharge cycles that are supported. The battery beyond these cycles becomes a source of energy leakage instead of exhibiting the ideal storage characteristics. In recent times, vendors have started offering thin film batteries for storing harvested energies. However, the nominal capacity is not comparable to normal batteries. The Enerchip CBC-050 rechargeable solid state battery from Cymbet [12] offers 50 microampere-hour capacities with over five thousand charge – discharge cycles if the depth of discharge is about 10 percent. The thin film battery from Excellatron [13] offers 1 mA/h and 10 mA/h batteries. STMicroelectronics's [14] EnFilm is another thin film battery.

Recent advances in supercapacitors have provided another alternative to store energy. Perhaps one of the biggest and best known drawbacks with capacitors is the

problem of their extremely low energy density compared to the batteries. Now days, with the use of electric double layer capacitors (EDLC) in these ultracapacitors, the energy density is reasonable for energy harvested WSNs to consider them as a viable option. A 10 F capacitor can be easily mounted on a printed circuit board (PCB). However, supercapacitors suffer from high leakage and, thus, result in high discharge. It is observed that the capacitors discharge significantly in the first couple of hours after charging. However, one of the major advantages of supercapacitors is the efficiency of storage, which is exceptional, and a capacitor ideally has an infinite number of charge–discharge cycles. These advantages outweigh the discussed disadvantages. If we now consider an embedded communication system with associated harvesting electronics and supercapacitor storage, we question whether this is a viable system. If not, what are the means and ways to make the system viable? We demonstrate the problem at a higher level and then discuss the options available to make the sensor nodes sustainable under varying energy profiles.

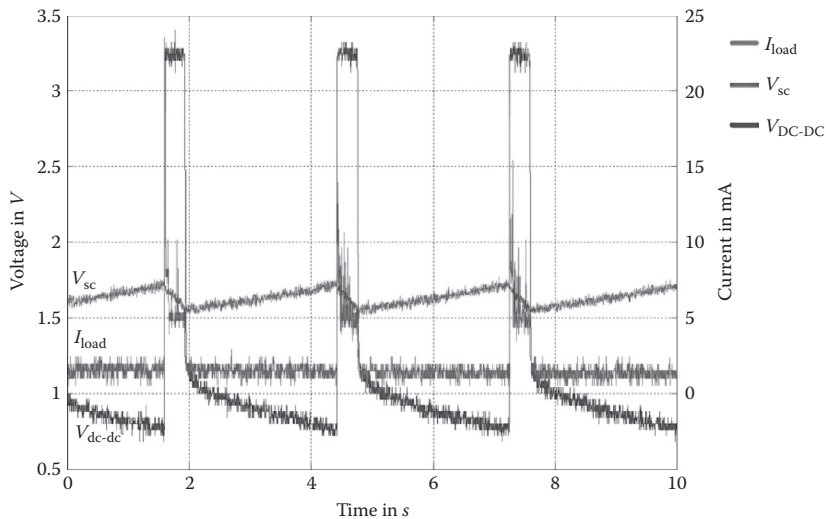
### 12.4.1 Energy Harvesting System

Figure 12.5 shows the block diagram of typical harvesting electronics. because harvesting sources provide variable power output, it becomes necessary to operate the system at the maximum power availability point. However, it is easy to



**Figure 12.5** Block diagram of an energy harvested embedded communication system.

see that, in order to perform maximum power point tracking (MPPT), the harvesting electronics require power. Therefore, the harvesting electronics initially perform power conditioning and store a small amount of energy to generate their own power. Once this power is available, the input power stored is conditioned successfully and stored in supercapacitors. The small input circuit matches the impedance of the energy source, rectifies the voltage whenever necessary, and delivers the power to the supercapacitor. This output voltage is shown as  $V_{sc}$ . The output DC-DC converter is a voltage regulator that feeds power to the load and is shown as  $V_{dc-dc}$ . MPPT algorithms are required to ensure that the source impedance variations due to the varying input power and load impedance are matched for transferring the maximum power from the source to the storage device. Due to efficiency issues in harvesting electronics with DC-DC converters, a critical input power is required to go over the *tipping point* for perennial functioning of the load. Figure 12.6 shows the voltages across the DC-DC converter and the supercapacitor (as shown in Figure 12.5) as well as the current drawn ( $I_{load}$ , load current) by a communication system in a low input power scenario. The initial start-up current requirement is approximately 23 mA, as seen in Figure 12.6. Each time the output DC-DC converter switches on, the load attempts to draw  $I_{load}$  indicated by the voltage waveform. The voltage builds up across the storage device; the super capacitor is also shown. As soon as the voltage across the supercapacitor builds up to the minimum input voltage for the DC-DC converter (1.8 V for TPS63031), the output switches on (3.3 V), and almost immediately loads the input harvester. This behavior repeats itself as long as the input power is low, leading to a large



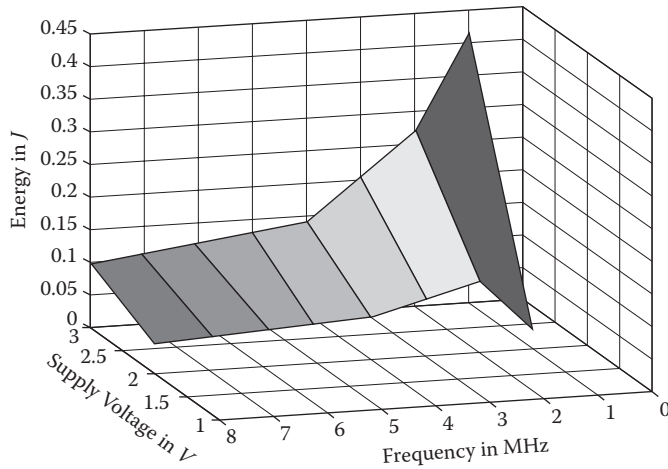
**Figure 12.6** Current and voltage waveforms when the embedded communication attempts to power on.

wastage of energy that could otherwise be accumulated for later use. This is a typical problem that requires adequate attention from system designers. Now, an important task is to search for simple solutions. Assuming for a moment that the system does power up, we now try to find how to ensure sustainable operation of the sensor node.

Many chip manufacturers and vendors recently have addressed this specific problem. For instance, Linear Technologies' [15] LTC 3108 is a DC-DC converter ideally suited for energy harvesting applications, particularly when harvester output is in the range of about 20 mV. The IC additionally provides a "power good" signal. The power good is a logic high signal that can be connected to any of the central processing units (CPUs) General Purpose Input/Output (GPIO) or Interrupt pins. The power good signal goes to logic high whenever the output voltage buildup of about 92 percent of the target value is achieved. This signal can then be used to bring the CPU out of the deep sleep modes at the exact time when there is energy for a specific activity. The power good signal indicates that the output voltage from the harvester electronics is within regulation.

$$E_{tot} = P_{tot} \Delta t = C_{tot} V_{dd}^2 f \Delta t + V_{dd} I_{leak} \Delta t \quad (12.1)$$

Equation (12.1) shows that the total energy consumed by the system has two components. The first part is the dynamic power consumption where the operating frequency, supply voltage, and device capacitance of the specific technology contribute. The second part is the static leakage power dissipated in every electronic component. This means that one may now have to go beyond the low power modes by monitoring and altering the supply voltage and altering the system operating frequency [16]. These are two important handles to adapt the system to the time varying, available power. Almost all low power microcontrollers including the MSP430 [17] used in our experiments work over a range of operating frequencies. The radio (CC2520) used in our communication experiments is IEEE 802.15.4 compliant [18]. As one can observe, higher operating voltage and frequency increases the power dissipation. The next question that arises is when one should alter these system parameters. From equation (12.1), since the product of  $f \cdot t$  is a constant for a given task, one way of reducing power dissipation is by varying the voltage. However, if the processor is awake for a longer time, perhaps altering the frequency is the right approach. Several IC manufacturers provide power and clock gating technique options to ensure that dynamic and static power to processor subsystems is minimized. An example of clock gating is the serial peripheral interface (SPI) clock used between a microcontroller and its peripherals, such as a radio transceiver. The clock circuit is started only when the data are available. Other examples include a camera clock that may not be required when there is no image processing. Similarly, a USB clock is not required when the system is not communicating. Thus, powering embedded communication systems with energy harvesting brings



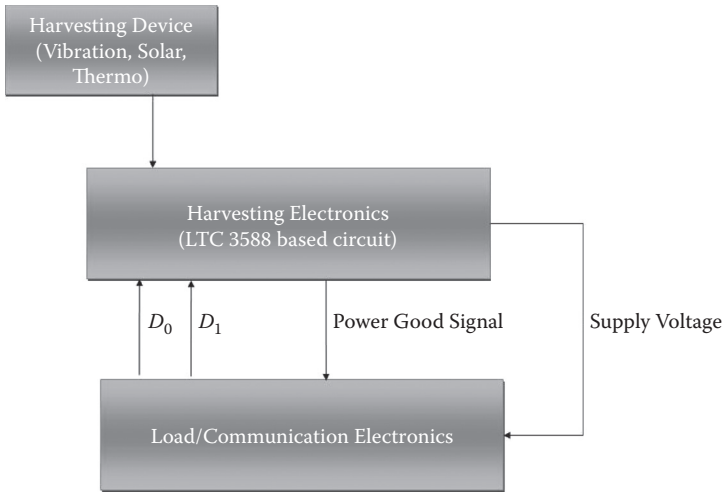
**Figure 12.7** Energy variation with frequency and voltage.

to focus dynamic tuning of system parameters well beyond the system's simple linearly polarized modes (LPMs).

### 12.4.2 Experimental Measurement

A laboratory measurement of energy consumed for various frequencies and operating voltages was recorded and is shown in Figure 12.7. The application running on the microcontroller was a quick-sort algorithm that sorted 10 numbers repeatedly about 7,000 times. The energy was calculated using a current probe amplifier. Equation (12.1) together with Figure 12.7 shows that, when the operating frequency is fixed, the total energy consumption of the system increases with the increase in operating voltage. Furthermore, at higher voltages, as frequency decreases, energy increase is near exponential compared to lower operating voltages. We now show one possibility of “on-the-fly” change in the system's operating voltage. Today's harvesting electronics are capable of generating multiple outputs based on input commands from a microcontroller. The LTC 3588 can accept input from a microcontroller to set its output voltage. Figure 12.8 shows the block diagram of the circuit where the microcontroller's GPIO pins D0 and D1 can signal the required output voltage from the harvester electronics. Four output voltages, 1.8 V, 2.5 V, 3.3 V, and 3.6 V, can be chosen by the microcontroller based on the D0 and D1 logic level combinations. The lithographic test chip (LTC) is capable of providing a continuous current of 100 mA.

Earlier, we mentioned about supercapacitors as energy storage buffers. While the capacitor value determines the quantum of energy storage, embedded communication systems running on supercapacitors have to wait longer for higher



**Figure 12.8** A schematic block diagram of an energy harvested communication system with commands to generate a specific output voltage.

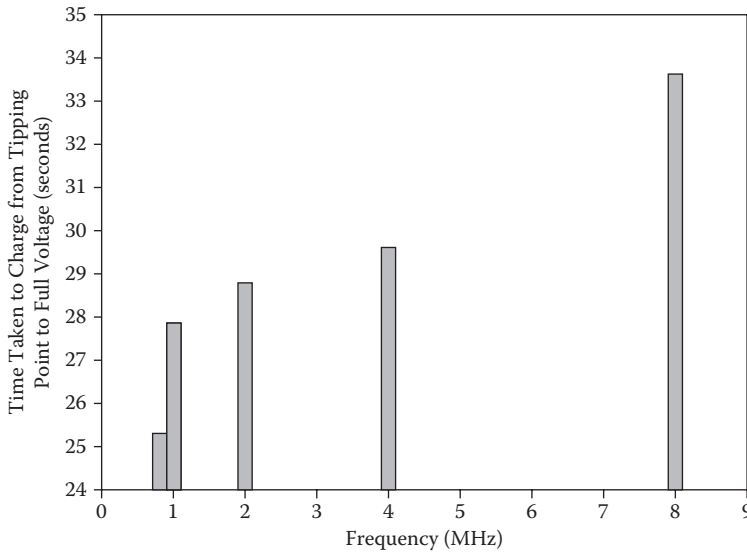
value capacitors to build up the required output voltage. To enable the capacitor to build faster, one option is to run the system at a lower frequency. To illustrate the impact of the frequency scaling on sustainable operation of the sensor node, we measured the time required for the supercapacitor to reach the operating voltage of 3 V from the tipping voltage of 1.8 V for various operating frequency. Figure 12.9 shows the time output of a node running a simple analog-to-digital conversion (ADC) sensing application with 2 sec sampling to reach 3 V; the operating voltage across the supercapacitor for several frequencies. The figure clearly shows the benefits of dynamically adjusting the frequency to the input power from the harvesting source.

Au: Incomplete thought.

## 12.5 Energy Budgets: System and Network Operations

So far, we have looked at energy harvesting technologies and the viability of such energies on commercially available, embedded communication systems. Let us now see how energy expenditure can be minimized so that the harvested energy can always remain at a sufficiently high threshold and, thus, ensuring perennial operation.

To complete a packet transmission, a node has to expend some amount of energy. This energy will have to ensure that packet transmission is successful without any midair collisions with data from other sources. Thus, our goal is to ensure that packet retransmissions are almost always avoided. This requires that we transmit the packet after making sure that the channel is free. We conducted an experiment to study the energy requirements for transmitting packets under conditions such as



**Figure 12.9** Effect of frequency scaling on harvested power.

with and without channel sensing, and with and without enabling acknowledgments from the receiver. Table 12.2 shows the summary of measurements. It turns out that by enabling acknowledgments, the transmitter consumes a significant amount of energy waiting for an acknowledgment. This is quite understandable, in the sense that the transmission is completed soon (less than 4 msec) and then the system switches to the reception mode and waits for an acknowledgement. Hence, waiting in reception state is energy intensive and should be used carefully by applications. For example, there might be some information that does not require a high degree of reliability. Such packets can be transmitted without expecting an acknowledgment.

Table 12.2 shows the energy spent for -18 dBm transmitted power. In this measurement, we considered many factors to be included when estimating the energy

**Table 12.2** Energy Requirement for Channel Sensing and Acknowledgements

Operation	Energy in $\mu J$
A: Transmit 30 bytes without channel sensing (-18 dBm)	65.0
B: Energy required for channel sensing (without backoff)	3.2
C: Transmit 30 bytes without enabling ACKs (-18 dBm)	65.0
D: Transmit 30 bytes with ACKs (-18 dBm)	110.5
Extra energy for acknowledgements: (D-C )	45.5

spent. These are channel sensing and overhead for reliability in terms of acknowledgment (ACK), backoff, and transmission. Table 12.2 shows a snapshot of energy measurements conducted for channel sensing and packet transmission with and without ACKs.

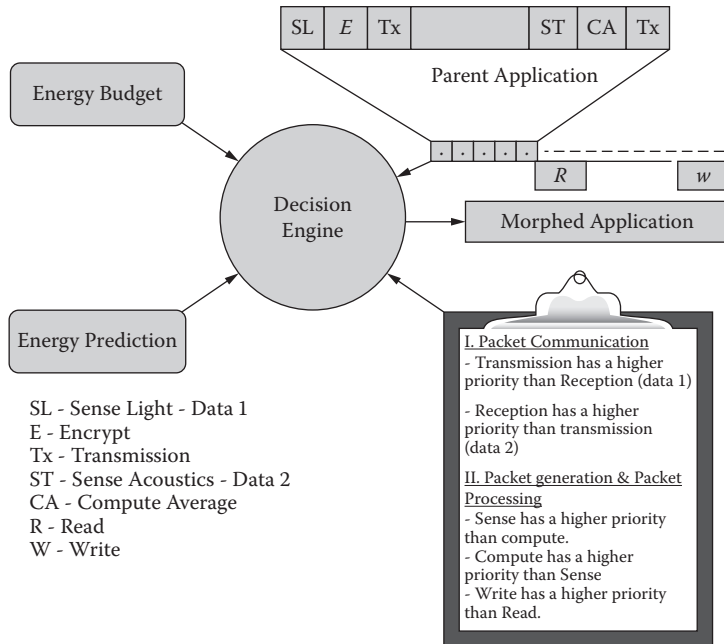
Assuming that a sensor monitoring two environment parameters, such as temperature and pressure, requires about 30 bytes for communication with a header and payload, to transmit this data, the table shows that channel sensing requires about 3.2  $\mu\text{J}$  of energy “each time” a carrier-sensing operation is carried out. On an average, at least four channel-sensing strobes have to be issued to ensure one packet transmission. This value can be substantially high if we consider a multi-node scenario with backoff mechanisms. Similarly, packet ACK is at an additional cost in energy. When the transmitter requests an ACK from its destination, it has to change its state from transmit state to receive state and wait for an ACK. Over and above the 65  $\mu\text{J}$  incurred for packet transmission, Table 12.2 shows there is an overhead of 45.5  $\mu\text{J}$  when ACKs are requested. The energy harvesting node could benefit by disabling channel sensing and ACKs for small packets.

### 12.5.1 Energy Harvested Applications: Challenges

Networking with harvested energies is fundamentally different compared to battery-driven systems because one has to look at the maximum rate at which energy can be used and not the limit on energy available from the source [19]. Applications designed over embedded communication devices have to ensure they are aware of the energy available in the system. Thus, applications have to be enabled to be “smart” to ensure that data gathering and application functioning are perennial. In our example, let us assume a base application, as shown in Figure 12.9. The application has to sense two environment parameters namely light and sound. How does the node manage all the activities, such as sensing, computing, and storing, and data transmission? To answer this, we propose a decision engine running on a sensor node to complete this task successfully.

The goal for the decision engine (DE) is to maximize the number of operations. We have done away with batteries and, instead, we have used low-leakage supercapacitors to harvest the energy. The capacitor (which acts as an energy buffer) is divided into two halves. The lower half of the energy is used for routine activities, such as neighbor discovery, route establishment, channel sensing, and other housekeeping activities. The upper half of the energy is used by the application, and our engine, in the first step whether the energy is above the lower threshold is checked. Time is divided into fixed slots and energy harvested in a slot is used by the engine to provide its recommendations. The DE utilizes two pieces of data to provide its recommendations. First, a time series, energy prediction model helps to identify how energy could be harvested. The second input is the energy cost database for operations required for energy budgeting. This database was built using real measurements. Figure 12.10 shows the architectural design of the engine.

Au: Confusing sentence. Please rework.



**Figure 12.10 Architectural overview of the decision engine.**

All operations associated with the base application are shown. The base application is comprised of two environment parameters that sense data, such as light and acoustic. Since the read from flash “R” and write “W” to flash are performed on a longer time scale, the figure displays them separately. While the light data are required to be encrypted, acoustic data can go as plain text. A forwarding node for multiple sensors has the additional role of aggregating acoustic data. Each block in the base application indicates a timer trigger to complete an activity. The energy-aware DE calculates the total energy requirement in each time slot and, assisted by a “heuristic rule book,” decides prioritization of the operations. The system finally recommends the best possible set of operations matching the energy harvested in the current time slot.

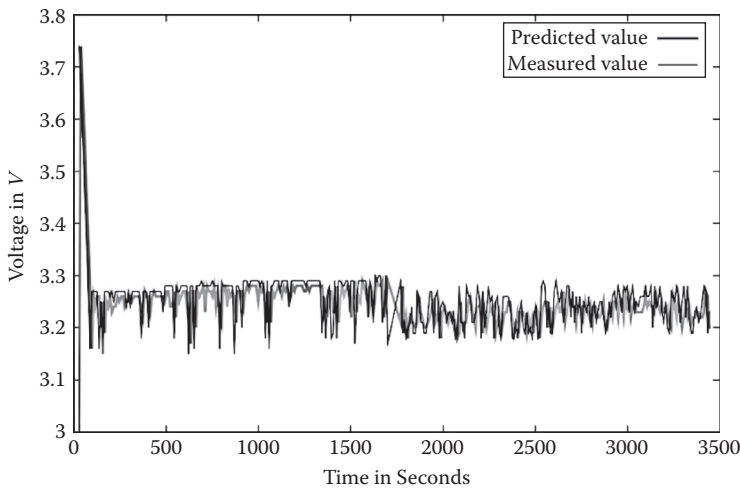
The other tool used by the DE is the energy budget calculator, which looks up a table of values of energy consumed per operation by Atmel microcontroller-based IRIS motes. The experimental setup includes a current probe amplifier connected to an oscilloscope to measure the current consumed by a node for an operation. The experimental results are documented in Table 12.3. It can be seen that the communication operation, together with writing to flash, requires significantly higher energy compared to other operations. The sum of energy consumed by the next operation is calculated every time the timer expires for a scheduled operation. Scheduled operation is performed only if the predicted energy inflow is higher than

**Table 12.3 Energy Consumed by IRIS Notes**

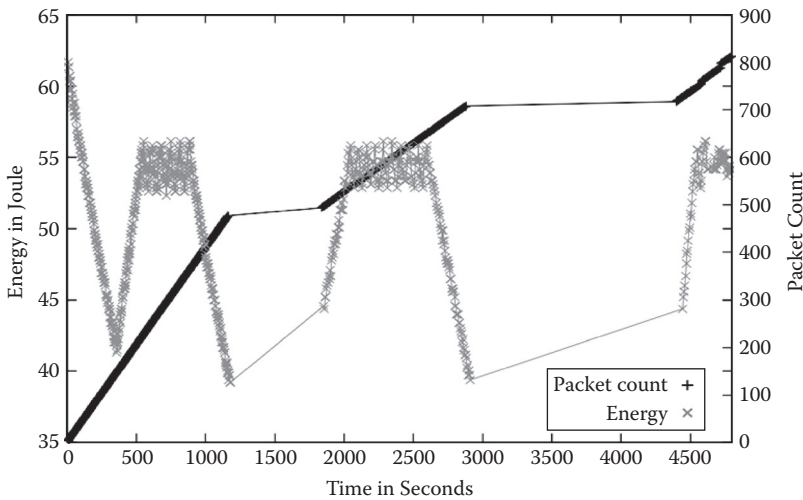
<i>Operation</i>	<i>Energy in J</i>
Average of 50 samples	7.056 $\mu$ J
Finding peak among 50 samples	7.392 $\mu$ J
Sensing once from ADC	16.128 $\mu$ J
Writing 1 byte to Flash	0.136 mJ
Reading 1 byte from Flash	28.224 $\mu$ J
Transmitting @ 0 dBm 28 bytes once	0.784 mJ
Receiving 28 bytes once	0.672 mJ

the energy required. If the predicted inflow is lower than the needed amount, the scheduled operation is ignored and other operations that can be accommodated are performed. To decide between multiple matches, the heuristic rule book is used to decide the priority. At the next scheduled timer trigger, the node checks for previously ignored operations and goes back to perform them if harvested energy is sufficient in this time slot.

Figure 12.11 illustrates the operation of the exponentially weighted moving average (EWMA) time series forecasting. The EWMA time series was proposed by Raghunathan, et al. [20]. It can be seen from the figure that the predicted value seems to be adapting to the changes in the actual measurement. A weight



**Figure 12.11 Measured and predicted solar panel terminal voltage.**



**Figure 12.12** Variation in packet transmission while harvested energy depletes and replenishes.

$\alpha$  of 0.5 also ensured smoothing the curve as a correction for measurement noise. Figure 12.12 shows a plot of energy contained in the supercapacitor and illustrates the application response to the harvested energy. The lower limit prevents energy depletion from the supercapacitor.

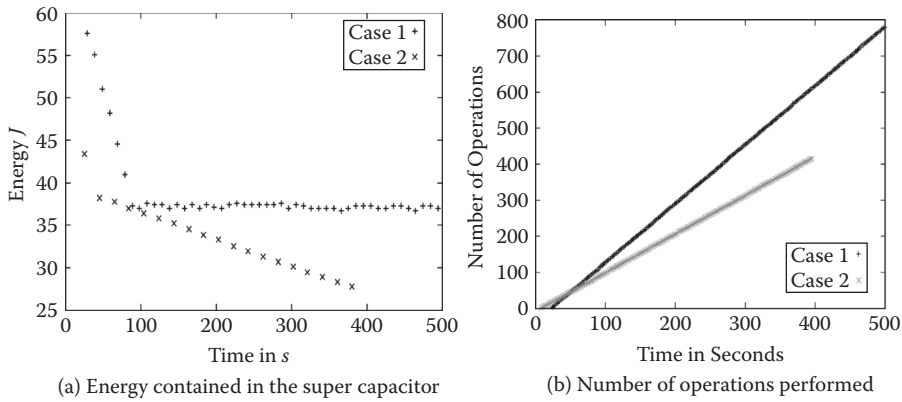
We conducted all the experiments outdoors under bright sunlight. During the course of the experiment, to emulate low light conditions, we blocked the light to the solar panels. This blocking is apparent at the times 0 to 350 sec, 935 to 1,450 sec and 2,600 to 3,770 sec. During periods of bright light, the node performed the operations matching closely with the base application.

missing value

Au: Does the "s" mean "sec" (seconds)?

During low light conditions, it modified its operation states and also stopped all operations when the supercapacitor energy level crossed the threshold for “no light” condition. When energy is available again, the node allowed for a certain amount of energy to accumulate before resuming energy-matched operations. Each point in Figure 12.12 represents a packet transmission. The oscillatory behavior during energy harvesting was found to be attributed to the heliomote. Figure 12.12 also shows the transmitted packet count (right “Y” axis) at the node. One can observe three distinct slopes, indicating the modification of the transmission interval from 2 sec to 4 sec and 5 sec. The three slopes may be attributed to fluctuation in the availability of energy.

Figure 12.13 illustrated the behavior of an application. We conducted specific experiments to study the response of the application when energy was excess, when energy was constantly replenished, and otherwise. We performed a comparison of the operation of the sensor node in two cases: (1) when the node is flooded



**Figure 12.13 Comparison of application’s behavior at high energy inflow versus low inflow.**

with harvested energy, and (2) when there is no energy to harvest, but the supercapacitor has a high amount of stored energy. While Figure 12.13a shows the energy contained in the supercapacitor in both cases throughout the experiment, Figure 12.13b illustrates the cumulative count of the number of operations that the node performed in each case. It can be seen that in case (2), the application modified itself and performed lesser number of operations than in case (1). The supercapacitor energy level remained constant in case (1), which clearly shows that the node performed energy-matched operations. For example, we looked at the behavior of the application between instances of time  $t = 200$  to  $300$  sec. The number of operations performed by the node in case (1) was 165, whereas, for the same interval of time in case (2), the node performed 108 operations. Of the 165 operations in case (1), sense and compute operations alone accounted for 54 and the remaining 111 operations were transmit operations. In case (2), however, the number of transmissions dropped to 57 and the number of sense and compute operations was 51. The application found an opportunity in case (1) to transmit as many packets as possible compared to case (2) in which it continued to operate, depleting slowly from the energy storage buffer. Calculation of energy consumption against the available energy, found manually, verified the behavior of the application from its energy consumption.

### 12.5.2 Storage and Retrieval of System State

We observe in Table 12.3 that writing data to a nonvolatile flash memory is an energy-intensive operation. At the same time, the unpredictability of the energy harvesting source might suddenly shut down the system and, thus, bring about data loss. RF energy harvesting at extremely low power levels, such as  $-20$

dBm, and discrete energy sources, such as linear switches, are examples where the energy harvested can be regarded as a discrete source. In addition to data loss, there might be certain parameters set dynamically during the course of the sensor's active state. For example, a sensor node was perhaps requested to change its sampling interval to say from "x" to "y" minutes. Thus, rollback to a set value also is lost when the node is subjected to a sudden shutdown. This problem of storage and retrieval of system state is a challenge in a network of energy harvesting nodes. Route discovery information, such as aggregator's address, are important data and "energy hard" to rediscover. Mikhaylov and Tervonen [21] describe this problem in detail and analyze the conditions under which data and system state can be stored and restored successfully when a node shutdown occurs. Thus, investigation into fast and reliable data storage and retrieval is required, particularly because recent nonvolatile memory technologies, such as Ferro Random Access Memory (FRAM), are promising in terms of speed and energy requirements.

### 12.5.3 Toward a Distributed Smart Application: Challenges

In the near future, reality will be where embedded sensor network communication is handled by energy harvesting wireless nodes. This also brings the smartness required by applications, as discussed previously. However, imagine a situation where nodes in the network are harvesting from different harvesting sources. The question is how nodes should schedule their packets. Moser, et al. [22] show that classical scheduling algorithms are not directly suitable for energy harvesting sensor nodes.

Figure 12.14 shows a typical mesh topology established by energy harvested sensor nodes, and illustrates that a collocated, solar harvesting, sensor node cannot assume energy availability from a neighboring thermo energy harvesting node. Given this scenario, we need to ensure that an application running on a system functions and data packets reach a sink node reliably. Table 12.4 shows the four possible cases that establish a node's application performance with respect to available energy and the link quality. While case 1 assures a good performance, case 4 forces a node into hibernation, and recovery from this mode is possible only when the harvested energy is sufficient to complete basic activities, such as sensing the sensor field, sensing the channel, and so on.

Sufficient attention should be given toward the process of the system, which impacts the application performance. Figure 12.15 shows three process sets, namely: Urgent, Periodic, and Random-Time. An example of urgent set includes sense or read, followed by compute and transmit. The process becomes an urgent process due to emergency events.

For example, events such as fire detection, intruder detection, failure of a subsystem, and etc. have to be done in real time. Figure 12.15 shows that when an urgent process is started, priority toward its completion is highest and, therefore,

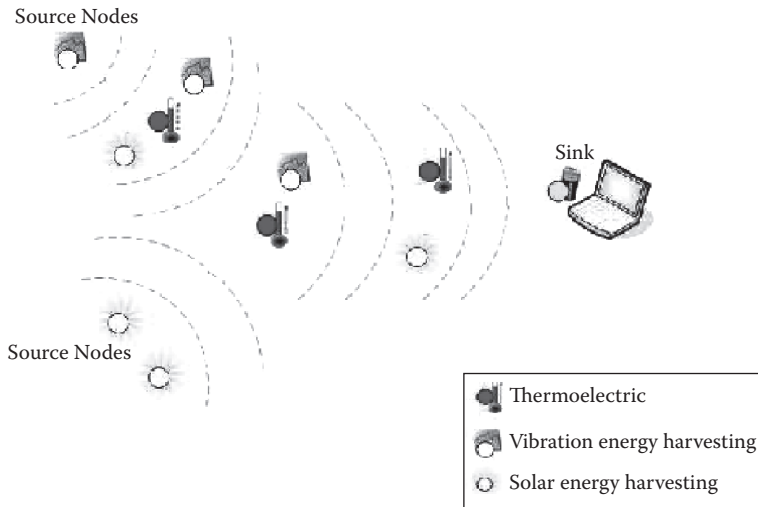


Figure 12.14 A network of energy harvesting sensor nodes powered with solar, thermo, and vibration harvesters.

Table 12.4 Impact of Available Energy and Link Quality on the Performance of Applications

Possible Cases	Node's Energy	Parent's Energy	Link to Parent
Case 1	Good	Good	Good
Case 2	Good	Good	Bad
Case 3	Good	Bad	Good
Case 4	Bad	Cannot be determined	Cannot be determined

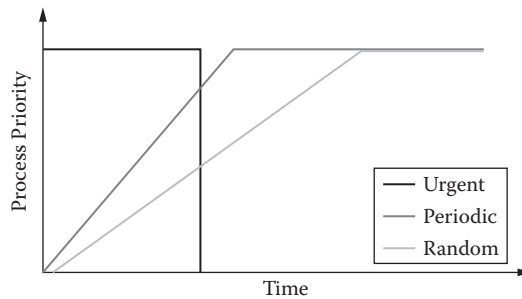


Figure 12.15 The priority setting of three type of processes.

energy usage preemption toward this is required. The choice of routes to relay such packets should ensure that packets do not get dropped or put into a queue in a waiting state at intermediate nodes. Therefore, choice of a relay and performing cooperative communication is, perhaps, a way to ensure that energy harvested network of nodes run a distributed smart application successfully.

## 12.6 Summary

We provided various ways and avenues for energy harvesting for ICT devices. We also discussed the amount of energy each type of harvesting could provide. It was shown that the varying energy levels could throw tough challenges for using them in the nodes, especially WSNs. The networking with energy harvesting devices is nontrivial. It calls for several algorithms that are of low cost in terms of overheads and also in the *energy sense*. However, these issues could be circumvented by using necessary intervention when the input energy is time varying. We explained a system with energy harvesting and how to use the varying sources of energy in WSNs. We provided a basic method for the energy harvested WSNs with measurement results. The ideas, though, are basic if performed well in terms of available energy and the required packet transmission. The ideas presented here are useful for not only the researchers, but also for realistic implementation of energy harvested devices and network. The input power adjustment software strategies coupled with the hardware options to work with lower clock frequency makes it sufficiently clear that energy harvesting technology is here to stay.

## REFERENCES

- [1] Crossbow IRIS motes. Online at: <http://www.xbow.com/Products/productdetails.aspx?sid=264>
- [2] David Lindley, The Energy Should Always Work Twice. *Nature*, vol. 458, March 12, 2009.
- [3] Jefferies, "Global vision for a wireless world," *18th Meeting of WWRF*, Helsinki, Finland, June 2007.
- [4] Online at: <http://www.eubusiness.com/topics/internet/ict-low-carbon/>
- [5] Borca-Tasciuc, D-A; Hella, M. M. Kempitiya A., "Micro-power generators for ambient intelligence applications," *4th International Workshop on Soft Computing Applications*, Arad, Romania, June 15–17, 2010.
- [6] Online at: <http://www.news.cornell.edu/stories/May10/VibroWind.html>
- [7] Online at: <http://web.mit.edu/newsoffice/2008/trees-0923.html>
- [8] T. Becker, M. Kluge, J. Schalk, T. Otterpohl, and U. Hilleringmann, "Power management for thermal energy harvesting in aircrafts," *Proceedings of IEEE Sensors*, 2008.
- [9] Online at: [www.micropelt.com/](http://www.micropelt.com/)
- [10] Online at: [www.mide.com/products/volture/v21bl.php](http://www.mide.com/products/volture/v21bl.php)

Au: Supply first name of Jefferies.

Au: Please sort out initials for these names.

Au: Supply full name of conference and where and when it was held.

- [11] EnOcean, The EnOcean GmbH, Germany. Online at: <http://www.enocean.com/en/home/>
- [12] Online at: [www.cymbet.com/](http://www.cymbet.com/)
- [13] Online at: [www.excellatron.com](http://www.excellatron.com)
- [14] Online at: <http://www.st.com/stonline/products/literature/ds/17370/efl700a39.pdf>
- [15] Online at: [www.linear.com/](http://www.linear.com/)
- [16] Amit Sinha, Energy Aware Software, master's thesis, Massachusetts Institute of Technology, Cambridge, MA, December 1999.
- [17] Texas Instruments, MSP430. Online at: <http://focus.ti.com/docs/prod/folders/print/msp430f1612.html>
- [18] IEEE 802.15.4-2006 Standard. Online at: <http://standards.ieee.org/getieee802/download/802.15.4-2006.pdf>
- [19] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power Management in Energy Harvesting Sensor Networks," *ACM Transactions on Embedded Computing Systems*, vol. 6, no. 4, September 2007.
- [20] Vijay Raghunathan, Aman Kansal, Jason Hsu, Jonathan Friedman, and Mani Srivastava, "Design considerations for solar energy harvesting wireless embedded systems," *Proceedings of IPSN*, 2005
- [21] Konstantin Mikhaylov and Jouni Tervonen. "Energy efficient data restoring after power-downs for wireless sensor networks nodes with energy scavenging," *Proceeding of the 4<sup>th</sup> IFIP International Conference: New Technologies, Mobility and Security (NTMS)*, Paris, France, February 2011.
- [22] Clemens Moser, David Brunelli, Lothar Thiele, and Luca Benini, "Real time scheduling for energy harvesting sensor nodes," *ACM Real Time Systems*, vol. 37, no. 3, December 2007.

Au: Supply full name of conference, location, and dates.

