Modeling of Wireless Sensor Nodes Powered by Tunable Energy Harvesters: HDL-Based Approach

Tom J. Kazmierski, *Senior Member, IEEE*, Geoff Merrett, *Member, IEEE*, Leran Wang, *Member, IEEE*, Bashir M. Al-Hashimi, *Fellow, IEEE*, Alex Weddell *Member, IEEE* and Ivo N. Ayala-Garcia *Member, IEEE*

***Abstract*—This paper presents a hardware description language (HDL) approach to modeling a complete wireless sensor node system powered by a tunable vibration energy harvester including both energy generation and consumption. Tunable energy harvesters, which can adjust their own resonant frequency through mechanical or electrical methods to match the input frequency, are attracting significant research interest. We present an accurate model of a vibration-based tunable electromagnetic energy harvester including its frequency tuning algorithm. We have also developed energy consumption models of sensor node components that use the generated energy to perform different tasks, such as autonomously tuning the resonant frequency, sensing temperature and transmitting wirelessly. The modeling of a wireless sensor node powered by tunable energy harvesting has not previously been reported, and now permits the simulation of the entire node. The accuracy of the proposed approach is demonstrated by comparing simulation results with experimental validation where relative errors of less than 1% are achieved.**

***Index Terms*—Tunable energy harvester, wireless sensor node, hardware description language.**

I. Introduction

In recent years, energy efficient wireless sensor networks have attracted a great research interest. Since wireless sensor nodes can provide information from previously inaccessible locations and from previously unachievable numbers of locations, many new application areas are emerging, such as environmental sensing [1], structural monitoring [2] and human body monitoring [3]. It has become widely agreed that long-lasting sensor nodes would benefit from energy harvesters [4–6], and vibration-based energy harvesters are used in many applications since mechanical vibrations are widely present [4]. Most of the reported vibration-based energy harvester designs are based on a microgenerator using a spring-mass-damper system with its characteristic resonant frequency. These devices normally have a high Q-factor and generate maximum power when their resonant frequency matches the dominant frequency of the input ambient vibration [7]. Consequently, the output power generated by the microgenerator drops dramatically when there is a difference between the dominant ambient frequency and the microgenerator's resonant frequency. Tunable microgenerators, which can adjust their own resonant frequency through mechanical or electrical methods to match the input frequency, have become an emerging area [8].

The Authors are with the Faculty of Physical and Applied Sciences, University of Southampton, Southampton, SO17 1BJ, UK, (phone: +44 2380593520; fax: +44 2380592901; email: {tjk,gvm,lw04r,bmah,asw,inag07r}@ecs.soton.ac.uk)

Most existing modeling approaches describe the mechanical and electrical components of an energy harvester separately using specialized tools, such as ANSYS for mechanical parts and SPICE for electrical circuits. These tools cannot be used directly to simulate complete sensor nodes because they cannot incorporate both electrical and mechanical components [9]. Hardware description languages with mixed physical-domain capabilities, such as VHDL-AMS and Verilog-AMS, have been in use for more than a decade with the first IEEE Standard for VHDL-AMS released in 1999 [10]. In recent years, the new HDLs have been applied to the modelling of mixed-technology energy harvesters and their associated power conditioning electronics [11]. However, as mixed-technology systems with non-electrical parts and analog electronics increasingly include sophisticated embedded software, models written in VHLD-AMS or Verilog-AMS may become too low-level and their underlying simulation times too excessive for validating a complete system [12]. SystemC with analog extensions seems a suitable platform for modelling mixed-technology systems with embedded software. The ongoing international effort to enhance SystemC with a support for continuous-time mixed-signal and mixed-technology descriptions has recently resulted in the release of the SystemC-AMS IEEE standard [13]. However, the currently available version of SystemC-AMS lacks an general solver for general, non-linear algebro-differential equations that are necessary to model the dynamics of electromechanical microgenerators and analog electronics. Pending the development of such a solver, in this paper we have used SystemC-A [14], a proprietary extended SystemC which has recently been equipped with an analog solver that uses a linearised state-space equation formulation [15], capable of simulating efficiently intricate operational scenarios of an energy harvester system.

Recently, Boussetta et al. [16] reported a VHDL-AMS model of a MEMS (microelectromechanical system) piezoelectric microgenerator. Their model incorporates the physical and geometric parameters of the microgenerator, is reusable, reflects experimental results well and can be integrated in global simulation of multidomain and mixed signal systems such as complete wireless sensor nodes.

We demonstrate in this work that such an integration is possible and present the first HDL-based approach that includes both energy generation and energy consumption in a complex wireless sensor node system. The system is comprised of a tunable energy harvester, a power processing and energy storage circuit, digital control for microgenerator frequency tuning, a temperature sensor, and a low-power microcontroller

and radio transciever. Accurate prediction of the system's performance is carried out through simulation and verified experimentally.

The paper is organized as follows. Section II outlines the principle of our proposed modeling approach. Section III describes the overall structure of the system and the modeling details of different system components, while Section IV presents the HDL implementation of the models. Section V shows the experimental validation of the simulation results and Section VI draws conclusions and identifies areas of future work.

II. Proposed Modeling Approach

To model a complete wireless sensor node system powered by a tunable energy harvester (Fig. 1) we have incorporated both physical, i.e. mechanical and magnetic, as well as geometric parameters of the microgenerator, circuit-level parameters of the analog electronic components, digital processes that implement the underlying control algorithms, and interactions between the different physical domains. Additionally, the energy consumption of the system components has been modeled as equivalent variable resistors, whose values are dynamically changing and linked with the control scenarios executed by the digital processes.

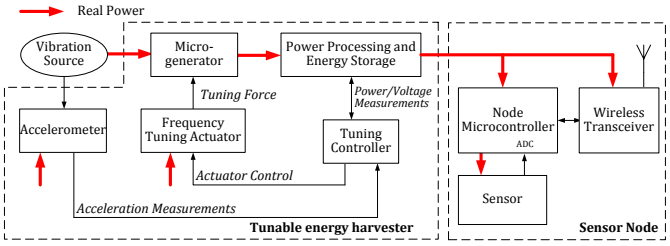


Fig. 1. Key components of a wireless sensor node system powered by a tunable energy harvester.

Our proposed approach has led to an accurate description of the complete system, with all its parts integrated as shown in Fig. 1: the mechanical microgenerator, its magnetic coupling with the electrical part, the power processing and storage circuitry, the accelerometer, tuning controller and mechanical actuator, and the sensor node (formed of a temperature sensor, microcontoller and radio transceiver). The mechanical microgenerator model incorporates the geometric parameters of the tuning magnets. The energy consumption models are dependent on the algorithms executed by the digital processes and the corresponding power consumption of the microcontrollers. Table I summarizes the modeling techniques adopted for each system component and shows where in the paper detailed descriptions of the models are presented.

The SystemC-A implementation presented below allows simulation and exploration of the design space comprising not only the energy harvester design parameters but also parameters concerning the sensor node operation. Thus, the whole sensor node system can be explored holistically with all the parts interacting together so that the performance of the system can be investigated accurately.

The presented approach is generic and is based on identifying the appropriate modelling level (e.g. behavioral or circuit level) or modelling type and physical domain (e.g. analog electrical, digital, mechanical, magnetic), combined with models of energy generation and consumption for the individual components of a sensor node. The approach has been applied, as a proof of concept, to the case study of a temperature sensor powered by a tunable microgenerator. The simulated operational scenarios with varying vibration frequencies and resulting energy consumptions have been experimentally verified.

III. A Tunable Energy Harvester Powered Wireless Sensor Node System

Fig. 2 illustrates the case study of a wireless sensor node system powered by a tunable energy harvester [17] which was used in this work. The wireless sensor node consists of a temperature sensor, microcontroller (MSP430 from Texas Instruments) and 2.4GHz radio transceiver (CC2500 from Texas Instruments). The microgenerator converts the input vibration into electrical energy. The generated AC voltage is rectified by a diode bridge and stored in a 0.55F supercapacitor (GS206 from CAP-XX). The supercapacitor acts as the energy source for the sensor node and the tuning controller (PIC16F884 from Microchip) which controls the frequency tuning of the microgenerator. In order to tune the resonant frequency of the microgenerator to the frequency of the vibration source, the tuning controller uses two input signals, one from the microgenerator and one from the accelerometer (LIS3L06AL from ST Microelectronics). The operational amplifier (MCP6141 from Microchip) acts as a comparator to generate square waves from the microgenerator output so that it is easy for the microcontroller to calculate the frequency. The detailed tuning algorithms are presented in Section III-C. The tuning controller also provides energy for the accelerometer, the operational amplifier and the tuning actuator (2100 Series Size 8 stepper motor from Haydon) so that these devices can be turned off when not in use. The sensor node periodically samples its supply voltage and the temperature sensor and transmits these data to a receiver connected to a remote PC.

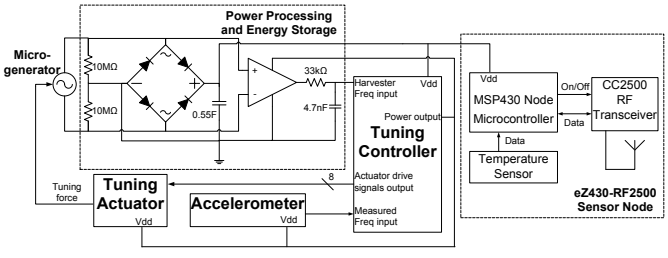


Fig. 2. System diagram showing the components of the wireless sensor node system powered by a tunable energy harvester used in this case study.

A. Tunable microgenerator

Fig. 3 shows the arrangement of the electromagnetic microgenerator and its tuning mechanism. The microgenerator is based on a cantilever structure. The coil is fixed to the

TABLE I
MODELING THE COMPONENTS OF A WIRELESS SENSOR NODE POWERED BY TUNABLE ENERGY HARVESTING

Component	Behavior Modeled	Model Abstraction Level	Section No.
Microgenerator	Interactions between mechanical, magnetic and electrical domains with component geometry	Nonlinear differential equations	Section III-A
Accelerometer	Analog input acceleration converted to output measurement and energy consumption model	Ordinary algebra equations and equivalent variable resistance	Section III-C
Tuning actuator	Digital input drive signal converted to output analog tuning force and energy consumption model	Ordinary algebra equations and equivalent variable resistance	Section III-C
Power processing	Nonlinear analog circuits	Nonlinear differential equations	Section IV-C
Tuning controller	Digital control algorithms and energy consumption model dependent on operation	SystemC digital processes and equivalent variable resistance	Section IV-B
Sensor node	Energy consumption model dependent on operation	Equivalent variable resistance	Section III-B

base, and four magnets (which are located on both sides of the coil) form the proof mass. The tuning mechanism uses magnetic force to change the effective stiffness of the cantilever which leads to a change of resonant frequency. One tuning magnet is attached to the end of the cantilever beam and the other tuning magnet is connected to a linear actuator. The linear actuator moves the magnet to the calculated desired position so that the resonant frequency of the microgenerator matches the frequency of the ambient vibration. The control algorithm is modeled as a SystemC digital process described in Section III-C.

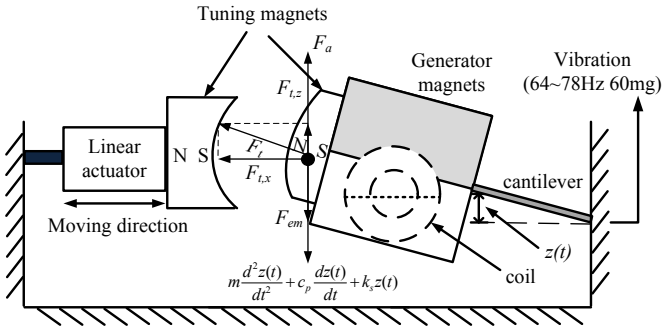


Fig. 3. Diagram of the tunable electromagnetic microgenerator showing its components and parameters.

The dynamic model of the microgenerator is [18]:

$$m\frac{\mathrm{d}^2z(t)}{\mathrm{d}t^2}+c_p\frac{\mathrm{d}z(t)}{\mathrm{d}t}+k_sz(t)+F_{em}+F_{t,z}=F_a \quad (1)$$

where m is the proof mass, $z(t)$ is the relative displacement between the mass and the base, c_p is the parasitic damping factor, k_s is the effective spring stiffness, F_{em} is the electromagnetic force, $F_{t,z}$ is the z component of tuning force F_t, and F_a is the input acceleration force. The z component of tuning force is:

$$F_{t,z}=F_t\frac{z(t)}{l_c} \quad (2)$$

where l_c is the length of the cantilever.

The resonant frequency ω_0 and damping coefficient ζ are:

$$\omega_0 = \sqrt{\frac{k_s}{m}} \quad (3)$$

$$\zeta = \frac{c_p}{2\sqrt{mk_s}} \quad (4)$$

Fig. 3 also shows all the forces acting on the generator. $F_{t,z}$ represents the z component of the tuning force. It is typically omitted in the dynamic equation of the tunable microgenerator for two reasons [18]. Firstly, when the two tuning magnets are far apart, the tuning force is small and its z component is negligible. Secondly the calculation of the tuning force between the two tuning magnets is usually performed by finite element (FE) analysis and standard FE tools cannot simulate $F_{t,z}$ as it varies over time [18]. However, we found that simulation results obtained without $F_{t,z}$ differ from experimental measurements when the two tuning magnets are closing together, i.e. the tuned resonant frequency is increasing, because the tuning force is becoming larger and its z component begins to affect the microgenerator's behavior. The z component of the tuning force has therefore been included in equation (1) to ensure the accuracy of the model.

As the two tuning magnets can be approximately treated as regular cuboids, the method developed by Akoun and Yonnet [19] can be adopted here to calculate F_t. For two cuboid magnets sharing the same central line along their thickness and with the area where these two magnets face each other, as shown in Fig. 4, the magnetic force between them is [18]:

$$F_t=\frac{M_1\cdot M_2}{4\pi\mu_0}\sum_{i=0}^{1}\sum_{j=0}^{1}\sum_{k=0}^{1}\sum_{l=0}^{1}\sum_{p=0}^{1}\sum_{q=0}^{1}(-1)^{i+j+k+l+p+q}\phi(u_{ij},v_{kl},w_{pq},r) \quad (5)$$

where M_1 and M_2 are the magnetization of these two magnets, μ_0 is the magnetic constant and $\phi(u_{ij},v_{kl},w_{pq},r)$ is a function of dimensions of the two magnets and their relative position. For the magnet configuration shown in Fig. 4, the interactive force between the two magnets is parallel with their polarization and ϕ is given by:

$$\phi(u_{ij},v_{kl},w_{pq},r)=-u_{ij}\cdot w_{pq}\cdot\ln(r-u_{ij})-v_{kl}\cdot w_{pq}\cdot\ln(r-v_{kl})+u_{ij}\cdot v_{kl}\cdot\tan^{-1}\frac{u_{ij}\cdot v_{kl}}{r\cdot w_{pq}}-r\cdot w_{pq} \quad (6)$$

where $u_{ij}=(-1)^j\cdot H-(-1)^i\cdot h$, $v_{kl}=(-1)^l\cdot W-(-1)^k\cdot w$, $w_{pq}=d+\frac{L+l}{2}+(-1)^q\cdot L-(-1)^p\cdot l$ and $r=\sqrt{u_{ij}^2+v_{kl}^2+w_{pq}^2}$.

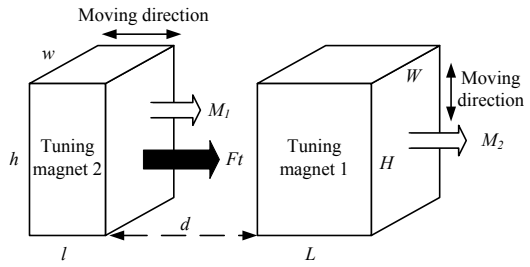


Fig. 4. Configuration and parameters of the two tuning magnets.

Fig. 5 shows the calculated attractive force between the two magnets.

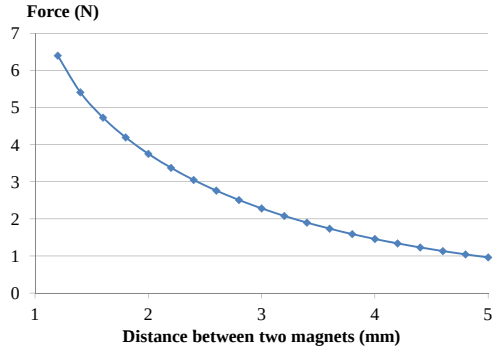


Fig. 5. Variation in the magnetic force between the two tuning magnets as a function of spacing.

The resonant frequency of the tuned microgenerator (f_r') is:

$$f_r' = f_r\sqrt{1+\frac{F_t}{F_b}} \tag{7}$$

where f_r is the un-tuned resonant frequency, F_t is the tuning force between two magnets and F_b is the buckling load of the cantilever.

The electromagnetic voltage generated in the coil is:

$$V_{em} = -\Phi\frac{\mathrm{d}z(t)}{\mathrm{d}t} \tag{8}$$

where $\Phi = NBl_{eq}$ is the transformation factor and N is the number of coil turns, B is the magnetic flux density and l_{eq} is the effective length. The output voltage is:

$$V_m(t) = V_{em} - R_c i_c(t) - L_c\frac{\mathrm{d}i_L(t)}{\mathrm{d}t} \tag{9}$$

where R_c and L_c are the resistance and inductance of the coil respectively and $i_c(t)$ is the current through the coil. The electromagnetic force is calculated as:

$$F_{em} = \Phi i_c(t) \tag{10}$$

To demonstrate the effect of including $F_{t,z}$ into the dynamic model, simulation results are compared with existing experimental tests [20]. The input frequencies have been chosen as 68, 76, 84, 92 and 98Hz. The simulation results of the output RMS power are superimposed onto the original experimental results, as shown in Fig. 6. The "X"s show the simulation results (alongside their respective values). It can be seen that the new simulation results match well with the experimental tests along the entire frequency range, and provide considerably better correlation than the previous model (depicted as "theory" in Fig. 6). It is clear that when the resonant frequency is high, the output power drops far below that of simulation with constant damping, i.e. without $F_{t,z}$, as when the two tuning magnets are close together the z component of the tuning force increases the overall damping of the cantilever oscillation and therefore less energy is generated.

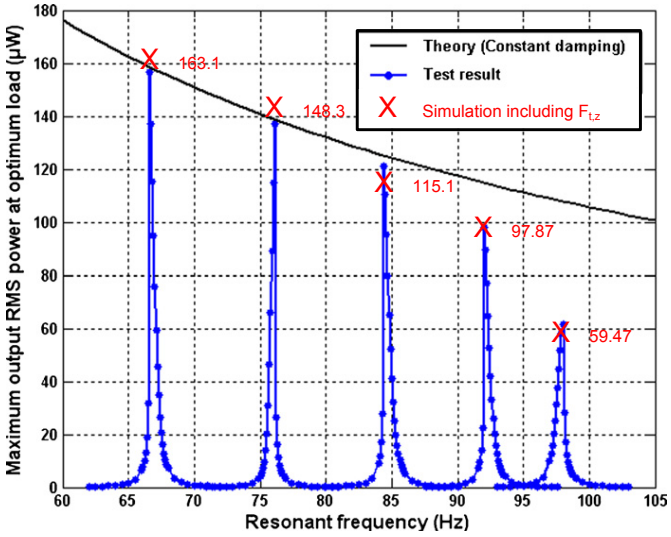


Fig. 6. The resonant frequencies of a tunable microgenerator, from experimental test results (marked by the peaks of the line with solid circles), and simulation with (marked by crosses) and without (marked by the solid line) the z component of the tuning force (original figure from [20] annotated with new simulation results).

The numerical values of the microgenerator parameters are listed in Table II. Some of the listed parameters were obtained from measurements as they are difficult to be calculated accurately, such as the parasitic damping c_p, the magnetic flux density B, the effective length l_{eq}, the coil inductance L_c and the buckling force F_b. Others can be calculated from the dimensions of the device components. The proof mass m equals material density times volume. In the above equations, the coil parameters are given in number of turns N and resistance R_c. However, when manufacturing a coil, the specification is often given by the thickness t_k, inner radius R_i, outer radius R_o and wire diameter d_w [18]. The total wire length is:

$$l_{tot} = 4f_f t_k(R_o^2 - R_i^2)/d_w^2 \tag{11}$$

where f_f is the fill factor. The number of turns is:

$$N = l_{tot}/(2\pi R_{ave}) \tag{12}$$

where $R_{ave} = (R_o - R_i)/2 + R_i$ is the average radius. The coil resistance is given by:

$$R_c = 4l_{tot}\rho/(\pi d_w^2) \tag{13}$$

where ρ is the resistivity of the material.

TABLE II
NUMERICAL VALUES OF MICROGENERATOR PARAMETERS

Symbol	Value	Unit	Symbol	Value	Unit
m	2.4e-3	kg	l	1.3e-3	m
c_p	1.3e-3	$\mathrm{Nm^{-1}s^{-1}}$	R_c	4500	Ω
N	6000		L_c	0.58	H
B	0.45	T	F_b	0.5	N

B. Energy-aware sensor node

The eZ430-RF2500 wireless sensor node from Texas Instruments has been used as an example to demonstrate the proposed approach. The on-board microcontroller is the MSP430F2274 and is paired with the CC2500 multi-channel RF transceiver, both of which are low-power components intended for use in embedded systems. The sensor node monitors the environmental temperature as well as the supercapacitor voltage. Once activated, it transmits the temperature and voltage values via a radio link. Transmissions are unacknowledged. Software on the node's microcontroller configures the sensor node in an energy-aware manner, by adjusting its duty cycle in response to the energy available in the supercapacitor. This operation is summarized in Table III.

TABLE III
SENSOR NODE BEHAVIOR AS A FUNCTION OF SUPERCAPACITOR VOLTAGE

Supercapacitor voltage	**Node behavior**
Below 2.7V	Wake up every 1 minute, no transmission
Between 2.7 and 2.8V	Wake up and transmit every 1 minute
Between 2.8 and 2.9V	Wake up and transmit every 5 seconds
Above 2.9V	Wake up and transmit every 1 second

In order to characterize the energy consumption model of the sensor node, its current draw was measured during a transmission cycle; the measured results are shown in Fig. 7. Although the current draw changes as the supply voltage varies, our experiments found that this variation was less than 1% across the operating voltage range of these tests (leading to the equivalent resistance changing by less than 2%), and hence is omitted from the modeling in this work.

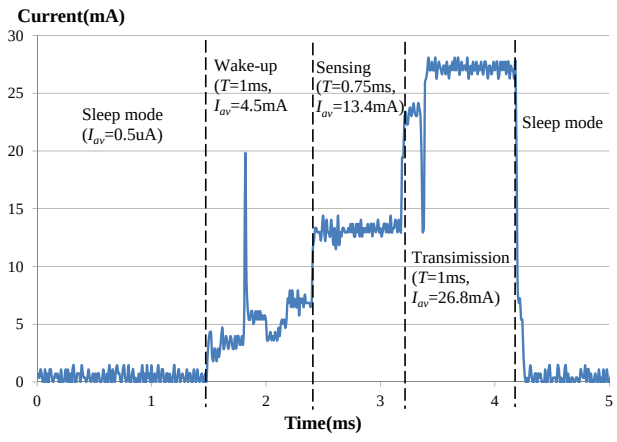


Fig. 7. Experimental measurements obtained for the current draw of the sensor node during an active cycle (T is duration and I_{av} average current).

With the supply voltage kept at 2.8V, and with each transmission lasting 2.75 ms, the sensor node consumes approximately 116 μJ of energy. Modelling this as an equivalent variable resistance, the node energy consumption model becomes:

$$R_{node} = \begin{cases} 186\ \Omega & \text{when active} \\ 5.6\ \mathrm{M}\Omega & \text{when sleeping} \end{cases} \quad (14)$$

C. Tuning control

In order for an energy harvester powered wireless sensor node system (Fig. 2) to work autonomously, all the system components need to be powered by the harvested energy. Standard SystemC modules were used to model the digital control process.

A watchdog timer wakes the tuning controller periodically and the tuning controller then detects whether there is enough energy stored in the supercapacitor. If there is not enough energy, the tuning controller goes back to sleep and waits for the watchdog timer again. If there is enough energy, the tuning controller will then detect the ambient vibration frequency to see if it matches the microgenerator's resonant frequency. If there is a difference between the vibration frequency and the resonant frequency, the tuning controller will start the tuning process by controlling the actuator to move the tuning magnet to the desired position. The detailed tuning algorithms are presented in Section IV-B. To tune the resonant frequency of the microgenerator effectively, the system incorporates a tuning controller, a linear actuator and an accelerometer. These three components are powered by the energy harvester as this is an autonomous system. To characterize the energy consumption models of these components, current measurements have been taken and energy consumptions have been calculated (see Table IV). According to the current and voltage values together with their operation times, the equivalent resistances for the energy consumption models of these devices have been obtained. Note that the measurements of the actuator have only been carried out twice (1 step and 100 steps) and the values in between are linearly interpolated.

TABLE IV
ENERGY CONSUMPTION MODELS OF THE SYSTEM COMPONENTS

Component (action)	**Operation time(ms)**	**Current (mA)**	**Power (mW)**	**R_{eq} (Ω)**	**Energy (mJ)**
Accelerometer	153	5.1	13.2	509	2.02
Actuator					
(1 step)	5	312	811	8.33	4.06
(100 steps)	500	156	405	16.7	203
Microcontroller					
(Coarse tuning)	149	1.9	5.0	1.38k	0.745
(Fine tuning)	325	5.1	6.5	250	2.11

IV. HDL IMPLEMENTATION OF SYSTEM MODELS

As shown in Fig. 2, the wireless sensor node system powered by a tunable energy harvester is a mixed-domain entity that incorporates both analog and digital components. The subsections below discuss the HDL implementation of the analog and digital models respectively.

A. Analog components

Analog and Mixed Signal (AMS) extensions in the SystemC-A language [14] were used to build the analog models. The analog part of the system, consisting of non-linear differential and algebraic equations, is handled by the extended syntax of SystemC-A where the user defines the behavior of each analog component by specifying the $build$ methods, that contribute to the analog equation set of whole system. In SystemC-A, the $build$ method is provided to support the automatic equation formulation of the user-defined system models. It is a virtual method in the abstract $component$ base class and inherited by all derived components. It consists of two functions, $BuildM()$ and $BuildRhs()$. SystemC-A uses the $BuildM()$ method to add the Jacobian entries to the analog equation set and $BuildRhs()$ method to build the equations, i.e. the right hand side of the Newton-Raphson linearized equation set. For example, the SystemC-A representation of the differential equation (1) is:

```
BuildM(ztQ,ztQ,-Ks); //Jacobian of equation (1)
BuildM(ztQ,ytQ,-Cp-Mp*S);
BuildM(ztQ,itQ,-Phi);
BuildRhs(ztQ,mpytdotdot+Mp*ytdot+Cp*yt+Ks*zt+Phi*it);
//Right hand side of equation (1)
```

B. Digital components

The pseudo code of the tuning algorithm is shown in Algorithm 1. As can be seen in Algorithm 1, the tuning controller wakes up periodically to check if the microgenerator's resonant frequency matches the input vibration frequency and, if necessary, to perform the tuning process by driving the actuator to move the tuning magnet to the desired position (Fig. 3).

Algorithm 1 contains two subroutines: coarse-grained tuning (Algorithm 2) and fine-grained tuning (Algorithm 3). The coarse-grained tuning algorithm measures the frequency of the microgenerator output and moves the actuator to the optimum position according to a predefined lookup table. If coarse-grained tuning alone cannot generate the best performance, the fine-grained tuning algorithm is employed. Fine tuning involves additional use of the energy budget but is essential especially when a large phase difference is detected between the input vibration and the microgenerator motion that prevents the microgenerator from working at the resonance. The fine-grained tuning algorithm takes another input, the raw vibration data from the accelerometer and moves the actuator to minimize the phase difference between the microgenerator signal and the accelerometer signal so that the microgenerator is working at resonance. It can be seen in Table IV that the fine-grained tuning algorithm requires more calculation than the coarse-grained tuning and also additional energy is consumed by the accelerometer.

As an example, the SystemC-A code of the model constructor and the coarse-grained tuning algorithm is listed below. The coarse-grained tuning algorithm, which is purely digital, is represented as a SystemC thread.

```
SC_CTOR(testbench){
timer=0; //watchdog timer signal
system(); //analog system
```

Algorithm 1 Harvester tuning control algorithm

1: **repeat**
2: Energy generation while waiting for watchdog timer (320 seconds)
3: **if** Enough energy stored in the supercapacitor (V_s $\geq$2.6V, where 2.6V is the minimum voltage for the actuator to start) **then**
4: Turn on Timer1
5: **repeat**
6: Measure microgenerator period
7: **until** 8 cycles have been measured
8: Turn off Timer1
9: Calculate input vibration frequency from 8 measurements
10: Find optimum position (8-bit) of tuning magnet through look-up table which has been pre-obtained and stored in the microcontroller memory
11: **if** Current position of tuning magnet matches optimum position (the accuracy is $1/2^8$) **then**
12: Goto 2
13: **else**
14: Perform coarse-grained tuning (**Algorithm 2**)
15: **end if**
16: Measure the phase different between the accelerometer signal and the microgenerator signal
17: **if** The phase difference is less than 100μs **then**
18: Goto 2
19: **else**
20: Perform fine-grained tuning (**Algorithm 3**)
21: **end if**
22: **end if**
23: **until** Forever

Algorithm 2 Coarse-grained tuning algorithm

1: **repeat**
2: Send the optimum position as 8-bit control signal to the actuator
3: The actuator moves tuning magnet
4: Wait 5 seconds for the microgenerator signal to settle down
5: Compare the current position and optimum position
6: **until** Current position of tuning magnet matches optimum position

Algorithm 3 Fine-grained tuning algorithm

1: **repeat**
2: Send the direction of movement that can reduce phase difference to the actuator
3: The actuator moves tuning magnet by 1 step
4: Wait 5 seconds for the microgenerator signal to settle down
5: Measure the phase of the accelerometer signal
6: Measure the phase of the microgenerator signal
7: Calculate the phase difference
8: **until** The phase difference is less than 100μs

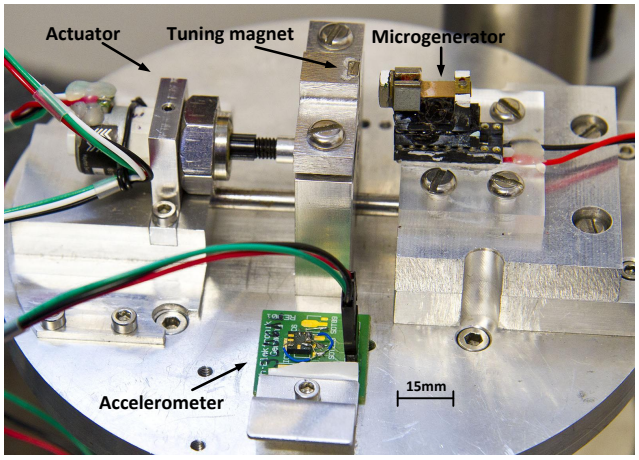
```
SC_CTHREAD(watchdog,clock.pos()); //invert every 320
seconds
SC_THREAD(coarse); //digital coarse-grained tuning
SC_THREAD(fine); //digital fine-grained tuning
sensitive<<timer; //sensitivity signal
}
⋮
void testbench::coarse(){
while (true) {
wait(); //wait for sensitivity signal
if(n6->readn()>2.6){//check if enough energy on
supercap
checking=true;//set flag as working
wait(20,SC_MS);//working time
checking=false;
double Freq;
Freq=ACM->get_Freq();//get microgenertor frequency
if(Freq<64) {ACT->desiredPos=0;}//tuning range is
from 64Hz
else if(Freq>78) {ACT->desiredPos=112;}// to 78Hz
else {ACT->desiredPos=(int)((Freq-64)*8);}//send
control signal to actuator
}}}
```

C. Complete system model

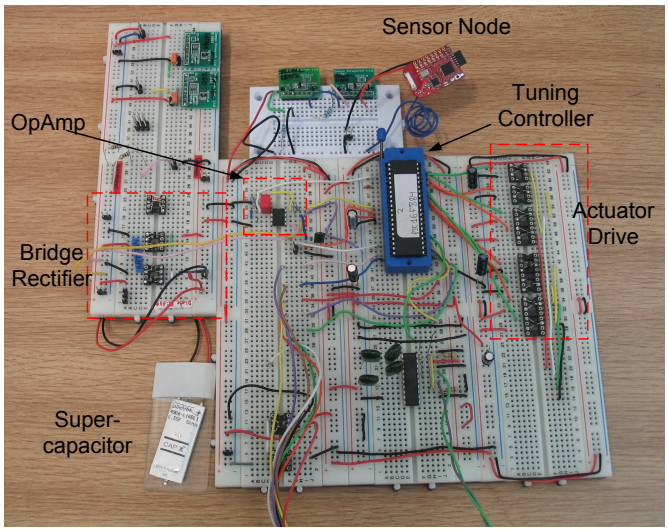
A SystemC-A model of the complete system Fig. 2 has been built and simulated. The SystemC-A code of the top-level testbench is listed below. The system components include the microgenerator, the diode bridge, the supercapacitor and the equivalent variable resistances of the actuator, the accelerometer, the tuning controller and the sensor node.

```
void testbench::system(){

ACT=new actuator;
ACM=new accelerometer;
uC=new control;
NODE=new sensor;
n0 = new Node("0");//don't write n0
n1 = new Node("n1");
n2 = new Node("n2");
n3 = new Node("n3");
n4 = new Node("n4");
n5 = new Node("n5");
n6 = new Node("n6");
//microgenerator generator *G1 =new
generator("G1",n1,n2,0.3192,64);
//diode bridge
diode *D1 =new diode("D1",n0,n1,2.117e-7,1.015);
diode *D2 =new diode("D2",n0,n2,2.117e-7,1.015);
diode *D3 =new diode("D3",n2,n3,2.117e-7,1.015);
diode *D4 =new diode("D4",n1,n3,2.117e-7,1.015);
resistor *R1 =new resistor("R1",n1,n0,10e6);
resistor *R2 =new resistor("R2",n2,n0,10e6);
//super capacitor model
resistor *Ri =new resistor("Ri",n3,n4,0.204);
resistor *Rd =new resistor("Rd",n3,n5,84.0);
resistor *Rl =new resistor("Rl",n3,n6,4375.0);
cap_ini *Ci0 =new cap_ini("Ci0",n4,n0,0.35,1.65);
cap_vary *Ci1 =new cap_vary("Ci1",n4,n0,0.21,1.65);
cap_ini *Cd =new cap_ini("Cd",n5,n0,0.21,1.65);
cap_ini *Cl =new cap_ini("Cl",n6,n0,0.06,1.65);
//energy consumption models for actuator,
accelerometer, tuning controller and sensor node
res_vary *RAct =new res_vary("RAct",n3,n0,1.0e9);
res_vary *RAcc =new res_vary("RAcc",n3,n0,1.0e9);
res_vary *RuC =new res_vary("RuC",n3,n0,1.0e9);
res_vary *RNode =new res_vary("RNode",n3,n0,1.0e9);
}
```



(a) Mechanical components: microgenerator, tuning magnet with actuator and accelerometer.



(b) Electrical components including power processing circuits, supercapacitor, tuning controller and sensor node.

Fig. 8. Complete wireless sensor node system powered by a tunable energy harvester.

V. Simulation Results and Experimental Validation

To validate the proposed approach, experimental measurements have been carried out on the actual wireless sensor node system and compared with the results of our simulations. Fig. 8(a) shows a photo of the mechanical components of the system (mounted on a mechanical shaker unit which acts as the vibration source), and Fig. 8(b) shows the electrical components.

The test scenario, used for both experiment and simulation, takes place over a 100-minute period. The input vibration frequency changes by 5Hz (about a third of the microgenerator's 14Hz tuning range), at ∼16 and ∼41 minutes (Fig. 10). During

the periods of a constant vibration frequency, the system recovers the energy expended by the retuning process and resumes operation at full capacity. This enables the complex tuning behaviour of the node to be investigated. After ∼41 minutes, the node is left to harvest energy at a constant frequency, to investigate the behaviour of the sensor node. These results focus on the simulation and experimental waveforms of the supercapacitor voltage as this reflects both the system's energy generation and consumption.

To illustrate the ability of our complete model to reflect the frequency-tuning behaviour of the energy harvester, simulation results are compared with experimental measurements (Fig. 9). After the vibration frequency changes, the supercapacitor voltage drops because the harvester is generating less energy than the system's quiescent consumption. A few minutes later, the tuning controller routinely wakes up, identifies that the generator is un-tuned, and actuates the tuning element so that the generator's resonant frequency matches that of the source vibration. This process consumes considerable energy from the supercapacitor (noticable by a steep drop in the supercapacitor voltage in Fig. 9). However, once tuned, the generator is able to begin charging the supercapacitor. The maximum relative error between simulation and experimental results occurs when the microgenerator is un-tuned (e.g. between ∼16 and ∼21 minutes); this can be explained by a simplified leakage behaviour in the supercapacitor model. However, the results presented in Fig. 9 display a maximum relative error of under 1%.

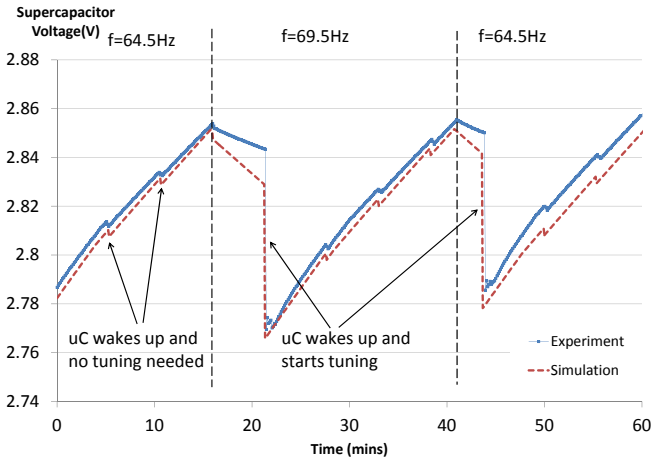


Fig. 9. Simulation and experimental results showing accurate modeling of the energy harvester's frequency tuning ability.

To demonstrate the importance of modeling a complete wireless sensor node powered by a tunable energy harvester, a series of simulations based on incomplete system models (i.e modeling the energy consumption of only a single system component) have been performed and results are presented in Fig. 10. At the beginning of the test, the charging slopes of all the waveforms are similar, demonstrating the accuracy of the energy generation model. However, as the test progresses, different components begin to consume energy to perform various tasks such as frequency tuning and wireless transmission, and the simulation results become increasingly inaccurate. It is clear from these simulations that in order to accurately predict the behavior of the complete system, the energy consumption of each system component must be included in the model.

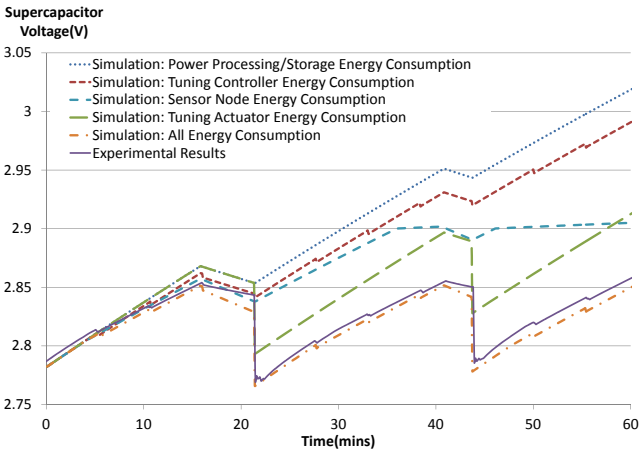


Fig. 10. Simulation and experimental results showing the effect of modeling the energy consumption of individual components (energy generation is modeled in all cases).

The results in Fig. 11 demonstrate the model's ability to capture the operation of a complete sensor node. The figure shows simulation and experimental results after ∼44 minutes (i.e. after the microgenerator is tuned to its final frequency). With the vibration frequency fixed, the behaviour of the sensor node can be better observed. At various stages (annotated on Fig. 11), the sensor node's duty cycle changes in response to the voltage across the supercapacitor (based on the behaviour specified in Table III). This change in duty cycle can be observed by the changes in the supercapacitor's charging slope (a lower gradient reflects a higher duty cycle, as more energy is being consumed). It can be viewed in detail in the insert on Fig. 11, which magnifies the simulated waveform when the duty cycle changes from waking up every 5 seconds to every 1 second. It can be observed that there is a delay of approximately one minute between this simulated duty cycle change (at ∼74 minutes) and experimental results (∼75 minutes). This error is caused by the limited resolution and non-idealities of the sensor node's ADC, resulting in it incorrectly measuring the voltage across the supercapacitor (and hence actually changes duty cycle at 2.91V). This error could therefore be reduced by incorporating these non-idealities into the model of the sensor node. However, the complete model continues to correlate well with experimental measurements and the maximum relative error is less than 1%. The CPU time for simulating the complete scenario, shown in Fig. 9 and 11, was 910 sec. with the simulation step-size of 5 μsec.

VI. Conclusion

Wireless sensor networks are fast developing and nodes powered by a tunable energy harvester have attracted great research interest. In order to design energy efficient wireless sensor nodes, we believe that it is crucial to consider all the components of a complete, autonomous wireless system in the context of both energy generation and consumption. This paper

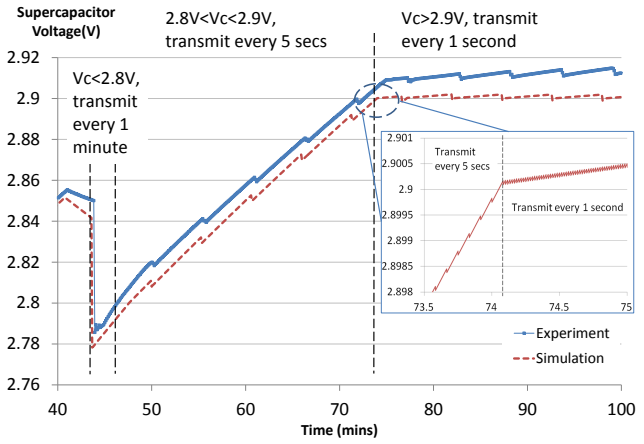


Fig. 11. Simulation and experimental results showing accurate modeling of the sensor node's behaviour.

presents the first HDL-based modeling approach that links the system's energy generation and consumption with its analog parts and digital processes. Simulation results of the SystemC-A models agree well with the experimental measurements and correctly reflect the changing energy flow when the digital processes are carrying out different operations. Our future work will focus on the optimization of both the energy harvester and digital control algorithms so that the system's overall energy efficiency can be improved.

Acknowledgment

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC), UK under grant number EP/G067740/1 "Next Generation Energy-Harvesting Electronics: Holistic Approach," website: (*http://www.holistic.ecs.soton.ac.uk*).

References

[1] C. Alippi, R. Camplani, C. Galperti, and M. Roveri, "A robust, adaptive, solar-powered WSN framework for aquatic environmental monitoring," *IEEE Sensors Journal*, vol. 11, no. 1, pp. 45–55, 2011.

[2] Q. Ling, Z. Tian, Y. Yin, and Y. Li, "Localized structural health monitoring using energy-efficient wireless sensor networks," *IEEE Sensors Journal*, vol. 9, no. 11, pp. 1596–1604, 2009.

[3] A. Sapio and G. Tsouri, "Low-power body sensor network for wireless ECG based on relaying of creeping waves at 2.4GHz," in *International Conference on Body Sensor Networks (BSN)*, 2010, pp. 167–173.

[4] S. Roundy, P. K. Wright, and J. M. Rabaey, *Energy scavenging for wireless sensor networks: with special focus on vibrations*. Springer, 2004.

[5] S. Ergen, A. Sangiovanni-Vincentelli, X. Sun, R. Tebano, S. Alalusi, G. Audisio, and M. Sabatini, "The tire as an intelligent sensor," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 7, pp. 941–955, 2009.

[6] J. A. Paradiso and T. Starner, "Energy scavenging for mobile and wireless electronics," *IEEE Pervasive Computing*, vol. 4, no. 1, pp. 18–27, 2005.

[7] P. Mitcheson, T. Green, E. Yeatman, and A. Holmes, "Architectures for vibration-driven micropower generators," *Journal of Microelectromechanical Systems*, vol. 13, no. 3, pp. 429–440, 2004.

[8] D. Zhu, J. Tudor, and S. Beeby, "Strategies for increasing the operating frequency range of vibration energy harvesters: a review," *Measurement Science and Technology*, vol. 21, no. 2, p. 022001, 2010.

[9] R. Amirtharajah, J. Wenck, J. Collier, J. Siebert, and B. Zhou, "Circuits for energy harvesting sensor signal processing," in *43rd ACM/IEEE Design Automation Conference*, 24-28 July 2006, pp. 639–644.

[10] E. Christen and K. Bakalar, "Vhdl-ams-a hardware description language for analog and mixed-signal applications," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 46, no. 10, pp. 1263 –1272, oct 1999.

[11] L. Mateu and F. Moll, "System-level simulation of a self-powered sensor with piezoelectric energy harvesting," in *International Conference on Sensor Technologies and Applications (SensorComm 2007)*, October 14-20, 2007, pp. 399–404.

[12] F. Pecheux, C. Lallement, and A. Vachoux, "Vhdl-ams and verilog-ams as alternative hardware description languages for efficient modeling of multidiscipline systems," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 24, no. 2, pp. 204 – 225, feb. 2005.

[13] K. Einwich, "Introduction to the systemc ams extension standard," in *Design and Diagnostics of Electronic Circuits Systems (DDECS), 2011 IEEE 14th International Symposium on*, april 2011, pp. 6 –8.

[14] H. Al-Junaid and T. Kazmierski, "Analogue and mixed-signal extension to SystemC," *IEE Proceedings on Circuits, Devices and Systems*, vol. 152, no. 6, pp. 682–690, 2005.

[15] L. Wang, T. Kazmierski, B. Al-Hashimi, A. Weddell, G. Merrett, and I. Ayala-Garcia, "Accelerated simulation of tunable vibration energy harvesting systems using a linearised state-space technique," in *Design, Test and Automation in Europe (DATE 2011)*, March 14-18, 2011, pp. 1267–1272.

[16] H. Boussetta, M. Marzencki, S. Basrour, and A. Soudani, "Efficient physical modeling of MEMS energy harvesting devices with VHDL-AMS," *IEEE Sensors Journal*, vol. 10, no. 9, pp. 1427–1437, 2010.

[17] I. A. Garcia, D. Zhu, J. Tudor, and S. Beeby, "Autonomous tunable energy harvester," in *PowerMEMS 2009*, 1-4 December 2009, pp. 49–52.

[18] D. Zhu, S. Roberts, J. Tudor, and S. Beeby, "Design and experimental characterization of a tunable vibration-based electromagnetic micro-generator," *Sensors and Actuators A: Physical*, vol. 158, no. 2, pp. 284–293, 2010.

[19] G. Akoun and J. Yonnet, "3D analytical calculation of the forces exerted between two cuboidal magnets," *IEEE Transaction on Magnetics*, vol. 20, no. 5, pp. 1962–1964, 1984.

[20] M. Zhu, E. Worthington, and A. Tiwari, "Design study of piezoelectric energy-harvesting devices for generation of higher electrical power using a coupled piezoelectric-circuit finite element method," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 57, no. 2, pp. 427–437, 2010.