Circuit and System Design Guidelines for Ultra-Low Power Sensor Nodes

Yoonmyung Lee, Yejoong Kim, Dongmin Yoon, David Blaauw, Dennis Sylvester University of Michigan, Ann Arbor

{sori, yejoong, dmyoon, blaauw, dmcs}@umich.edu

ABSTRACT

Designing an ultra-low power sensor node requires careful consideration of the system-level energy budget. Depending on applications, various components can dominate total energy. In this paper, we review three different system energy budget scenarios where any of the microprocessor, memory, and timer of a sensor node can dominate the energy budget. The design space and corresponding trade-offs for these three components are explored to suggest guidelines for the design of ultra-low power sensor nodes.

Categories and Subject Descriptors

B.7.1 [Types and Design Styles]

General Terms

Design

Keywords

Ultra-low Power, Wireless Sensor Node, Low Voltage

1. INTRODUCTION

Sensor nodes collect useful data from their environment in a distributed fashion and process the collected raw data to extract essentials to be transferred to user or central data collection point. Recently, with advances in circuit design, packaging and battery technologies, tiny sensor nodes whose volume is as small as a few cubic-millimeters are demonstrated [1].

Figure 1 shows key building blocks of ultra-low power sensor nodes. Sensor nodes typically consist of sensors for measurement, a microprocessor for data processing, a memory for data and execution code storage, a radio for data transfer, a timer for wake up and communication control, and a power management unit. Although sensor nodes commonly includes these function units, each function unit's frequency or length of activation can largely vary from applications to applications which makes the system energy budget varies significantly. Other function units, such as wake up receiver, energy harvesting unit can be added for more complete system, but scope of discussion in this paper will be limited to the key function units illustrated in Figure 1 for simplicity.

Table 1 shows an example energy/power consumption of function blocks in state-of-the-art literatures [1]. Table 2 shows usage models of these function blocks in three different scenarios. Scenario 1 represents a surveillance application. A sensor node wakes up every 5 seconds and takes an image of a monitored object. The image is then processed with microprocessor (MP) to determine if there was significant

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 2012, June 3-7, 2012, San Francisco, California, USA. Copyright 2012 ACM 978-1-4503-1199-1/12/06 ...\$10.00.



Figure 1. Simplified block diagram of a typical sensor node.

change from past images. The result is then transferred to base station. For image processing, a large (16kB) memory (compared to the other scenarios) has to be implemented in this scenario. Scenario 2 represents a temperature monitoring application where a sensor node wakes up every 10 minute and measures temperature periodically. Measured temperature is stored in memory and transferred to a base station hourly. Due to the small data size, a smaller 1kB memory can be used for this scenario. Scenario 3 is temperature monitoring sensor identical to Scenario 2, except that measured data is collected by symmetric communication among sensor nodes instead of centralized base station.

With power/energy numbers given in Table 1 and usage models in Table 2, energy usage in each scenario can be estimated as shown in Figure 2. With Scenario 1, which is relatively computationally intensive, the microprocessor dominates system energy. Therefore, optimizing the microprocessor for minimum energy operation is the key approach for minimizing overall energy consumption and maximizing lifetime of the sensor node.

With sensor nodes that wake up infrequently and spend most of their time in standby mode, memory leakage power dominates total energy as shown in Scenario 2. This is because, unlike other function blocks, memory cannot be power-gated in standby mode for minimizing leakage power since measurement data and execution code stored in the memory will be lost with power-gating. Therefore, to design a sensor node with infrequent

 Table 1. Example energy/power consumption of function blocks in ULP sensor nodes.

Unit	Description	Energy /Power	Ref.
Microprocessor	Commercial ARM [®] Cortex-M3 29 pJ/in		[2]
Memory	8T SRAM (65nm)	14 pW/bit	[1]
Timer	Gate-leakage timer including LDO 8.6 nW		[1]
Radio – TX	ULP UWB transmitter energy	1.65 nJ/bit	[3]
Radio – RX	ULP UWB receiver power 1.64 mV		[3]
Image Sensor	28×128 pixel low power image sensor 140 nJ/frame		[4]
Temperature Sensor	PTAT-based	806 nJ/meas.	[1]

Unit	Scenario 1 (Surveillance)	Scenario 2 (Temp. monitor)	Scenario 3 (Temp. monitor /w symmetric communication)
(wake up period)	5 sec	10 min	10 min
Microprocessor	100k inst. / wake up	2k inst. / wake up	2k inst. / wake up
Memory	16kB, 229nW	1kB, 14.3nW	1kB, 14.3 nW
Timer	Not used	Not used	8.6 nW
Radio – TX	1 kb / wake up	100 b / 1 hr	100 b / 1 hr
Radio – RX	Not used	Not used	1.64mW for mismatch
Image Sensor	1 frame / wake up	Not used	Not used
Temperature Sensor	Not used	1 meas. / wake up	1 meas. / wake up

Table 2. Example usage model of ULP sensor nodes.



Figure 2. Energy consumption by functional unit in example sensor node scenarios.

activity, minimizing memory leakage with careful selection on memory topology is required.

Sensor nodes are required to form a wireless sensor network in some applications, which requires symmetric communication among the sensor nodes. Scenario 3 shows how energy budget could change with such sensor nodes. With an ULP sensor node, symmetric radio communication should be periodically activated since its energy budget does not allow continuous radio activation. This radio activiation timing has to be precisely determined by a low power timer. If the activation time determined by one sensor node is different from the other sensor node, receiving sensor node has to activate radio receiver for mismatched amount of time until transmitter is activated. Due to the high receiver power (in the order of mW), energy wasted for receiver activation during this mismatch can dominate energy budget of this types of sensor nodes as shown in Figure 2. Therefore, accuracy and power trade-off of the timer should be carefully explored for designing sensor nodes with symmetric communication.

These three sensor node usage scenario confirms that function block which dominates the energy can vary. Once dominating function block is determined at system level, circuit level design optimization has to be done on the dominating function blocks. Therefore, in this paper, design guidelines and trade-offs for three different function blocks, namely – microprocessor, memory and timer – are discussed at the circuit level.

2. MICROPROCESSOR DESIGN CONSIDERATIONS IN ULP SENSOR NODES

For sensor nodes where microprocessor dominates system energy budget, optimizing operation conditions of microprocessor is of critical concern to reduce energy consumption. There are two key knobs that can be adjusted in design time, namely supply voltage scaling [7] and fabrication technology selection [8].

2.1 Supply Voltage

E

Voltage scaling is a popular approach for microprocessor energy reduction [5][6]. Although voltage scaling incurs a performance penalty, typical sensor node performance requirements aremuch lower than that of commercial microprocessors shown in [5][6], allowing voltage scaling to the near-threshold or sub-threshold regions. However, excessive voltage scaling can actually increase energy consumption. Therefore, for sensor nodes whose microprocessor takes significant portion of system energy budget, minimum energy supply voltage (V_{min}) [7] should be carefully determined.

Microprocessor energy consumption can be represented as the sum of switching (E_{switch}) and leakage energy (E_{leak}) [EQ1]. Since switching energy is proportional to the square of supply voltage [EQ2], voltage scaling enables a quadratic E_{switch} reduction.

$$E_{Total} = E_{switch} + E_{leak}$$
[EQ1]
$$E_{switch} \propto V_{DD}^{2}$$
[EQ2]

 E_{leak} can be represented as product of execution time (T) and leakage current (I_{leak}) of the microprocessor [EQ3]. In the superthreshold regime, E_{leak} has only a minor impact on overall energy consumption. However, as voltage scaling is extended towards the sub-threshold regime, the exponential reduction of on-current incurs an exponential increase in execution time (T), making E_{leak} non-negligible compared to the quadratically reduced E_{switch} . Therefore, with excess voltage scaling, the quadratic E_{switch} reduction can be outweighed by the exponential increase in E_{leak} .

$$I_{leak} = T \times I_{leak}$$
[EQ3]

Figure 3 shows the simulated energy–supply voltage relationship for a single transition in a 51 stage inverter chain. The minimum energy operation point is approximately 220mV where the increase in E_{leak} overcomes the reduction of E_{switch} .

Meanwhile, sensor node microprocessors can have various architectures, and even with identical architecture, execution code varies across applications, making activity factor vary. Some instructions, such as multiplication/division, demand high circuit activity whereas simple instructions, such as register reads, do not. Therefore, computationally intensive applications can increase activity factor (or vice versa), andthis activity factor should be considered when determining the minimum energy



Figure 3. Energy consumption as a function of supply voltage



Figure 6. Energy vs Vdd for an inverter chain.

operation voltage (V_{min}). Figure 4 shows the impact of activity factor on V_{min} varies . With a higher activity factor, microprocessor energy consumption is more dominated by E_{switch} , making quadratic switching energy reduction more effective down to lower voltages. Therefore, V_{min} is lower with high activity factor.

In summary, sensor node microprocessors should leverage voltage scaling as far as performance requirements allow while comprehending the limitations posed by the minimum energy point.

2.2 Technology Selection

The previous section showed that the minimum energy operating voltage can be used for microprocessor designs. However, this discussion was limited to the time when the microprocessor is active. It also assumed that an arbitrary microprocessor frequency can be chosen for minimum energy operation. In practical sensor node design, standby time energy consumption should be considered. Moreover, there may be specific performance requirements. By carefully selecting fabrication technology, these two factors can be considered in microprocessor design.

Microprocessor energy consumption is largely affected by two key metrics – duty cycle and performance. Duty cycle is defined as the ratio of active time to total time. To first order performance is simply the clock frequency that a microprocessor runs at - we model this using 40 fan-out-of-4 (FO4) inverter delays in this discussion, which represents a typical low voltage processor microprocessor cycle time [8]. Using these two metrics, the design space for a microprocessor in a sensor node is depicted in Figure 5. Microprocessor for continuous monitoring, such as biosignal monitoring [9], have high duty cycles (bottom of Figure 5), whereas microprocessors that wake up infrequently - every 10 minutes or longer -have low duty cycles (top of Figure 5) [10]. Sensor nodes with temperature or pressure sensors would not require high microprocessor performance (left of Figure 5) while sensor nodes with imaging capabilities require higher performance (right of Figure 5).

Microprocessor energy consumption in a sensor node can be represented by [EQ4].

$$E_{total} = E_{active} + E_{standby}$$
[EQ4]
= $E_{switch} + E_{leak} + E_{standby}$

 E_{active} denotes the energy consumed when the microprocessor is active. Therefore, E_{active} consists of energy consumed by switching activity (E_{swith}) and energy consumed by leakage

current during active mode (E_{leak}). $E_{standby}$ is energy consumed in standby mode. With advanced technologies, E_{switch} can be reduced with smaller switching capacitance lower supply voltage for given performance requirement. However, due to lower threshold voltage (V_{th}) of transistors, E_{leak} and $E_{standby}$ can increase significantly. To study impact of technology selection, a chain of 40 of FO4 inverters can be simulated in SPICE for 5 different commercial CMOS technologies ranging from 250nm to 65nm. Activity ratio of microprocessor circuit is assumed to be 0.2. Supply voltage is set to meet the required performance and if the performance can be achieved at minimum energy operation voltage (V_{min}) [5], the voltage is not scaled down below V_{min} .

Figure 6 shows the technology choice for the energy optimal operation of a microprocessor. There are two clear trends in the presented technology selection analysis. First, microprocessors with small duty cycles prefer old technologies due to their high device V_{th} and low leakage current. Second, high performance microprocessors prefer new technologies, due to improved E_{switch} with lower gate/parasitic capacitance and lower supply voltage for given performance requirement. These two trends are most clear with low duty cycle/low performance (top-left of Figure 6) and high duty cycle/high performance (bottom-right) microprocessors.

Relatively old technologies are optimal in microprocessors with high duty cycle and low performance (bottom left of Figure 6). Newer technologies are expected to be preferred for high duty cycles to achieve energy optimality. However, faster speed of advanced technology can lead to task completion far ahead of required microprocessor cycle time, which results in unwanted additional idle time and waste of leakage energy. This energy penalty offsets the benefit of low E_{switch} with advanced technology and makes older technologies a better choice.

On the opposite corner of the design space in Figure 6 (low duty cycle with high performance requirements), older technologies offer reduced leakage. However, the larger supply voltage needed to meet performance requirements lead to higher active energy and overall newer technologies are preferable in such scenarios.



Figure 5. Optimal technology selection for microprocessor in a sensor node [8]

3. MEMORY DESIGN CONSIDERATIONS IN ULP SENSOR NODES

As wireless sensor nodes require collected data to be stored until being transmitted or processed, memories as data storage are essential elements in the system. Memories, by their nature, have a low activity factor; only a few percent of the circuit is active at a time. In addition, most wireless sensor nodes are duty-cycled and some of them spend \geq 90% of the lifetime in the sleep mode. In this context, standby energy consumption dominates the memory's total energy consumption and becomes the critical factor. Non-volatile memories (e.g., Flash) have virtually zero retention (standby) power, but their active power is typically in ~mW range, which far exceeds the energy budget of smaller battery-powered sensor nodes. Thus, embedded volatile memories (e.g., SRAM, eDRAM) are preferred in such systems, however their volatility leads to significant leakage power during standby mode. Generally, smaller devices contribute to a smaller bitcell size and more leakage power. Because the bitcell array leakage power dominates the memory standby power, there exist trade-offs between the standby power and the bitcell size. These can be clearly observed in low-power memories recently reported, which we are grouping them into three as shown in Figure 7: Regular SRAMs, Sub-V_{TH} SRAMs and eDRAMs, and ULP SRAMs.

Regular SRAMs typically consume >100pW/bit of standby power, easily exceeding 10μ W with just 1kB of capacity. However, their bitcell size is usually smaller than that of other types of SRAM, and the operating frequency is much faster with full supply voltages, resulting in up to >300MHz [12]. This type of SRAM is suitable for sensor nodes that require fast operations as well as a large amount of storage, with a relatively large battery size that can afford the large power consumption; both active and static powers can be more than100 μ W.

Sub-V_{TH} SRAMs reduce standby power down to ~1pW/bit using low supply voltages, hence allowing smaller batteries . However, their robustness requirements, such as write margins and read margins, usually cannot be met with the conventional 6T bitcell structure due to the reduced supply voltage. A typical approach to overcome this issue involves the separation of read and write paths, which requires additional transistors. Because of these added devices, such bitcells incur 50~100% area overhead compared to regular SRAMs. 10T bitcells [14] [15] from recent literature are shown in Figure 8(a) and (b); the bitcell in (a) exploits stack effect for added leakage reduction, while the bitcell in (b) supports fully differential read and write. The area overhead could be avoided at the cost of multiple supply voltages with high power active operations [17]. For the applications requiring large amount of memory capacity, embedded DRAMs (eDRAM) [18] can be an alternative choice. They have much smaller bitcell size compared to SRAMs, as shown in Figure 8(c), resulting in much denser memory. However, the refresh power greatly depends on temperature, so the standby power may easily become orders of magnitude higher than that of SRAMs at high temperature. These sub- V_{TH} SRAMs and eDRAMs are in the middle of the spectrum (Figure 7) and can be useful for sensor nodes with a relatively small battery size (~cm³) that do not require high-speed memory access.

At the other end of the spectrum is ULP SRAM. These often use thick-oxide HVT I/O devices, achieving less than 10fW/bit of



Figure 7. Low-power memories in recent publications can be grouped into three classes: Regular SRAMs, Sub-V_{TH} SRAMS and eDRAMs, and ULP SRAMs. Depending on application, designers should choose among these memory circuits.



Figure 8. (a) 10T SRAM bitcell [14], (b) 10T SRAM bitcell [15], (c) 2T eDRAM Gain Cell [18], (d) 10T SRAM bitcell [2]. Thick lines indicate HVT (I/O) devices.

standby power. This can be attractive for extremely energyconstrained systems; for example, the entire 3kB ULP SRAM in [2], whose bitcell is shown in Figure 8(d), consumes only <100pW of standby power, and this enables the use of a mm³scale battery while still providing months of lifetime. Although their operating frequency is usually limited to < 1MHz due to the combination of HVT I/O devices and the low voltage supply, they are preferred in mm³-scale sensor nodes because low-power consumption has the highest priority in such systems. However, the extremely low standby power comes with a large area penalty; the 10T HVT bitcell [2] in Figure 8(d) is \sim 3× larger than regular SRAMs due to the large size requirements of the HVT devices. This limits the maximum memory capacity in small form-factor systems.

4. TIMER DESIGN CONSIDERATIONS IN ULP SENSOR NODES 4.1. Bolo of Accurate Timers

4.1 Role of Accurate Timers

The circuit blocks introduced in previous sections can benefit from various circuit techniques to reduce the power consumption. However, unlike those blocks, a radio transceiver block cannot operate at such low power mainly due to physical limitations of antenna efficiency and size. Since allocating substantial area for the antenna is usually not possible for ULP wireless sensor applications, the average active power of the radio of a sensor node is still in mW regime, which is 5 to 6 orders of magnitude higher than the average power of other sensor node components [20]. Therefore the operating duration of the radio transceiver block has to be limited to minimize overall system's energy consumption. To make designing the system harder, the level of instantaneous power from the radio transceiver cannot be afforded directly from power source for some applications. To cope with the problem, the system has to have a large capacitor to act as energy reservoir. Therefore the radio transceiver's operating duration not only has significant impact on total energy consumption but may also increase system volume. That is why the system has to be designed to minimize the operating duration of the transceiver. As will be shown in the next paragraph, the accuracy of a timer has substantial impact on average operating duration of the radio transceiver.

Figure 9 shows an example scenario of an ULP wireless sensor node system. A sensor node is activated every 20 minutes to take measurement and a microprocessor processes data for 100ms at 3μ W. Then once in every hour, collected data is transmitted by radio module, consuming 1mW for 1ms. For any given time, timer is running in background at 1nW to track time change and initiate scheduled tasks at right timing. In total, the sensor node nominally consumes 5.5µJ per an hour.

However, some applications require sensor nodes to communicate with each other. In such a scenario, each sensor is equipped with a timer to initiate communication synchronously. Unfortunately, the timers may suffer from mismatch, leading to one sensor node turning on the radio earlier than others. With 200ms hourly mismatch, the total energy consumption increases to 206μ J per hour, with 200μ J arising due to the mismatch penalty.

4.2 Design Considerations

From this example, it is clear that sensor nodes require accurate timers. However several considerations should be made in the design of such timers.

The first is the power consumption of the timer module itself. Unlike other system components, the timer is always on and cannot benefit from duty cycling. For this reason, even if the timer consumes several orders of magnitude lower power than other duty-cycled high power components such as microprocessor, it may easily dominate overall energy consumption. The second one is random error in timing. The timer will suffer from various sources of noise, such as thermal noise, which leads to random mismatch of period between different timers. Finally, process, voltage, and environmental (PVT)fluctuations will impact the timer, similar to conventional circuits. PVT sensitivity must bebe distinguished from random error. Unless the sensors are used in harsh environments, variation sources such as process or temperature are likely to act as fixed offsets between different timers, rather than manifesting as random error. With the help of simple (negligible power) digital logic, fixed offsets can be calibrated to minimize their effect. Therefore, more emphasis can be allocated to the power consumption and random error characteristics of a given timer design.

Before introducing real design examples, we introduce the concept of Allan deviation [21], the standard metric used to compare the frequency stability performance of timers. Period



Figure 9. Example wireless sensor node operation scenario

deviation due to noise is often Gaussian and its magnitude of deviation can be characterized with RMS jitter, which is defined as the standard deviation of timer period. To compare the jitter of two different types of timers, normalized RMS jitter (expressed in ppm) is often used. However, the accuracy of a timer for measuring a multi-cycle synchronization period is not captured by these metrics since they ignore the averaging of jitter over multiple timer cycles. For example, two timers with identical ppm jitter can have different hourly accuraciessince the timer with a shorter period has more cycles to count, allowing for more averaging. In addition, long term stability estimation based on RMS jitter may give a false impression that longer averaging periods are always beneficial, which is refuted by measurements. Instead, timer uncertainty is well characterized by Allan deviation, which is defined by [EQ5], where τ is the observation period and y_n is n-th fractional frequency – that is, $\Delta f/f$ for Figure 10, where f is frequency – over observation time τ . Bracket \Leftrightarrow denotes time average of samples.

$$\sigma_y^2(\tau) = \frac{1}{2} \langle (y_{n+1} - y_n)^2 \rangle, \ \sigma_y(\tau) = \sqrt{\sigma_y^2(\tau)}$$
 [EQ5]

4.3 Design Examples

Here we briefly describe three recent ultra-low power timers [22-24]. Trade-off exists among three key parameters discussed in previous section, namely power consumption, random error,



Figure 10. Allan deviation at τ =1000s and power consumption of low power timers [22-24]

and PVT variation. Figure 10 plots Allan deviation at averaging time of 1000 seconds and power consumption of the three designs. Please note that conventional XO means a 32.768kHz Pierce oscillator built with a discrete inverter running at 1V. The performance summary of the three timers is shown in Table 3. CMOS based circuits have lower power consumption at smaller area. However, the circuits need to oscillate at low frequency since its oscillation is not based on a resonant component. Inevitably, CMOS-based timers have worse performance with random error and PVT variation compared to a quartz crystal based timer. Quartz crystal based timers - commonly denoted as crystal oscillator (XO) - on the other hand, show much better frequency stability characteristics. These merits come at the expense of higher power and extra system volume due to the quartz crystal. The smallest 32.768kHz quartz crystal package available at the date of publication is 1.44mm³ [27]. There are also MEMS-based resonators available that can be integrated on-chip while providing better accuracy than CMOS circuitbased timers [25, 26]. Power consumption of these approaches is still higher than other timers, but the numbers are expected to reduce with advances in MEMS and circuit techniques.

If the application does not require communication between sensor nodes, or synchronization period is short, CMOS based timers can save power and system volume. On the other hand, if timing has to be tightly controlled, a quartz crystal can be a good option.

	[22]	[23]	[24]	[25]	[26]
Oscillation source	CMOS circuit	CMOS circuit	Quartz crystal	MEMS	MEMS
Frequency	≈11Hz	3.7~16Hz	32.768kHz	32.768kHz	32.768kHz
Power	150pW	660pW	5.58nW	600nW	3.2µW
σ_{y}	9.89×10^{-5} (τ =1400s)	5.28×10^{-5} (τ =1400s)	1.16×10^{-8} (τ =1000s)	N/A	N/A
V _{DD} sensitivity	400ppm/mV	4200ppm/mV	17.6ppm/V	N/A	<1ppm/V
Temp. stability	490ppm/°C	31ppm/°C	0.04ppm/ °C ²	± 80ppm (-20°C-80°C)	± 10ppm (0°C-50°C)
Area (mm ²)	0.019	0.015	0.3 (w/o crystal)	2.25	0.156

Table 3. Low power timers for ULP sensor nodes

5. CONCLUSIONS

In this paper, we have outlined guidelines for designing an ULP sensor node system. The energy budget of an ULP sensor node can vary significantly depending on application. Therefore, functional blocks that dominate system energy should be assessed early on and the design space for dominating blocks should be explored to balance and minimize the total system energy.

For microprocessors, voltage scaling down to minimum energy operation voltage (V_{min}) and technology selection for optimizing microprocessor design in consideration of performance requirement and duty cycle is discussed. Various memory topologies are presented to show trade-off between standby power and bitcell area overhead. State-of-the-art low power timers suitable for ULP sensor nodes are also discussed to illustrate the accuracy versus power trade off.

6. **REFERENCES**

 Y. Lee, et. al., "A Modular 1mm³ Die-Stacked Sensing Platform with Optical Communication and Multi-Modal Energy Harvesting," *IEEE ISSCC*, 2012.

- [2] G. Chen, et. al., "Millimeter-Scale nearly Perpetual Sensor System with Stacked Battery and Solar Cells," *IEEE ISSCC*, 2010.
- [3] M. Crepaldi, et. al., "An Ultra-Low-Power Interference-Robust IR-UWB Transceiver Chipset Using Self-Synchronizing OOK Modulation," *IEEE ISSCC*, 2010.
- [4] S. Hanson, et. al., "A 0.45-0.7V sub-microwatt CMOS image sensor for ultra-low power applications," *IEEE Symp. on VLSI Circuits*, 2009.
- [5] Intel XScale. http://www.intel.com/design/intelxscale/
- [6] IBM PowerPC. http://www.chips.ibm.com/products/powerpc/
- [7] B. Zhai, et. al., "Theoretical and Practical Limits on Dynamic Voltage Scaling", *ACM/IEEE DAC*, 2004.
- [8] M. Seok, et. al., "Optimal Technology Selection for Minimizing Energy and Variability in Low Voltage Applications," ACM/IEEE ISLPED, 2008.
- [9] A. Dogan, et. al., "Multi-Core Architecture Design for Ultra-Low-Power Wearable Health Monitoring Systems," ACM/IEEE DATE, 2012
- [10] M. Seok, et. al., "The Phoenix Processor: A 30pW Platform for Sensor Applications," *IEEE Symp. on VLSI Circuits*, 2008.
- [11] N. Verma, et. al., "A High-Density 45nm SRAM Using Small-Signal Non-Strobed Regenerative Sensing," *IEEE JSSC*, vol. 44, no. 1, Jan. 2009.
- [12] M. Yamaoka, et. al., "A 300-MHz 25-µA/Mb-Leakage On-Chip SRAM Module Featuring Process-Variation Immunity and Low-Leakage-Active Mode for Mobile-Phone Application Processor," *IEEE JSSC*, vol. 40, no. 1, Jan. 2005.
- [13] N. Verma, et. al., "A 256kb 65nm 8T Subthreshold SRAM Employing Sense-Amplifier Redundancy," *IEEE JSSC*, vol. 43, no. 1, Jan. 2008.
- [14] B. H. Calhoun, et. al., "A 256-kb 65-nm Sub-threshold SRAM Design for Ultra-Low-Voltage Operation," *IEEE JSSC*, vol. 42, no. 3, Mar. 2007.
- [15] I. J. Chang, et. al., "A 32kb 10T Sub-Threshold SRAM Array with Bit-Interleaving and Differential Read Scheme in 90nm CMOS," *IEEE JSSC*, vol. 44, no. 2, Feb. 2009.
- [16] T. Kim, et. al., "A Voltage Scalable 0.26V, 64kb 8T SRAM with V_{min} Lowering Techniques and Deep Sleep Mode," in *IEEE CICC*, 2008.
- [17] Y. Wang, et. al., "A 1.1GHz 12μA/Mb-Leakage SRAM Design in 65nm Ultra-Low-Power CMOS Technology with Integrated Leakage Reduction for Mobile Applications," *IEEE JSSC*, vol. 43, no. 1, 2008.
- [18] Y. Lee, et. al., "A 5.42nW/kB Retention Power Logic-Compatible Embedded DRAM with 2T Dual-Vt Gain Cell for Low Power Sensing Application," *IEEE ASSCC*, 2010.
- [19] G. Chen, et. al., "A Cubic-Millimeter Energy-Autonomous Wireless Intraocular Pressure Monitor," *IEEE ISSCC*, 2011.
- [20] Y. Lee, et. al., "Synchronization of ultra-low power wireless sensor nodes." *IEEE MWSCAS*, 2011.
- [21] D.W. Allan, "Time and frequency (time-domain) characterization, estimation, and prediction of precision clocks and oscillators," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control.* Vol. UFFC-34, No. 6, pp. 647-654, Nov. 1987.
- [22] Y. Lin, et. al., "A I50pW program-and-hold timer for ultra-lowpower sensor platforms," *IEEE ISSCC*, 2009
- [23] Y. Lee, et. al., "A 660pW multi-stage temperature compensated timer for ultra-low-power wireless sensor node synchronization," *IEEE ISSCC*, 2011.
- [24] D. Yoon, et. al., "A 5.58nW 32.768kHz DLL-assisted XO for real time clocks in wireless sensing applications," *IEEE ISSCC*, 2012.
- [25] J. Chang, et. al., "32kHz MEMS-based oscillator for implantable medical devices," *International Symposium on Integrated Circuits*, 2011
- [26] D. Ruffieux, et. al., "Silicon resonator based 3.2μ W real-time clock with ± 10 ppm frequency accuracy," *IEEE JSSC*, vol. 45, no. 1, 2010
- [27] Fox Electronics, 2010. FX122. http://www.foxonline.com/pdfs/FX122.pdf