Energy Efficient Architecture of Sensor Network Node Based on Compression Accelerator

Jue Wang, Beihua Ying, Yongpan Liu, Huazhong Yang and Hui Wang
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
{wangjue03, yingbh04}@mails.tsinghua.edu.cn, {ypliu, yanghz, wangh}@tsinghua.edu.cn

ABSTRACT
In this paper, we propose an energy efficient architecture of wireless sensor network node. It consists of a general-purpose processor and several compression accelerators. To verify the low energy consumption of this architecture, we implement a baseband chip of sensor node by 1-poly 6-metal 0.18um CMOS technology, in which a hardware accelerator is realized based on a distributed wavelet compression algorithm. Our measurements show that the compression accelerator based architecture reduces over 98% energy consumption compared with the traditional solution.

Categories and Subject Descriptors
B.7.1 [**Hardware**]: Integrated Circuits - *Algorithm implemented in hardware*

General Terms
Design, Performance.

Keywords
Wireless sensor network; Compression accelerator; Energy efficient; Chip design

1. INTRODUCTION
Wireless Sensor Networks (WSNs) is an emerging technology to monitor the physical world via a spatially distributed self-organized network without infrastructure. WSN consists of a large number of sensor nodes. The primary resource of a node is the battery supply, which limits its lifetime. Therefore, minimizing each node's energy consumption is quite important.

According to the measurements from [1], transmitting energy for a single bit data is over 1000 times larger than a 32-bit computing energy. Furthermore, data from neighbor sensors are likely to be correlated in WSNs. Therefore, it is reasonable to compress the original data before transmitting to reduce the sensor node's energy consumption.

The traditional solution is executing the compression algorithm by processor. But it has been estimated that the processor consumes up to 64% of the power of a node during the execution of relatively simple applications [2]. So reducing the processor's energy consumption by executing the compression algorithm with hardware accelerator and putting the processor into sleep mode is relevant. However, some issues require solutions, such as combining the general-purpose processor with a compression accelerator, the communication architecture between them, etc. They are also the problems to address in this paper.

This article proposes an energy efficient scalable architecture of sensor node which is based on a general-purpose processor and a tunable number of compression accelerators (CA). Furthermore, a power efficient CA communication architecture is described. The proposed node architecture is implemented by real silicon and measurements show up to 98% energy savings compared with the pure software approach.

The rest of this paper is organized as follows. Section 2 illustrates the related work in this area. In Section 3, we describe the proposed energy efficient scalable node architecture. Section 4 describes overall architecture of our sensor node and the hardware design of CA unit. We present the chip implementation and measurement results in Section 5. Finally, Section 6 concludes the paper and points out the future work.

2. RELATED WORK
There are already some groups working on hardware-based compression because it is more energy efficient than the software-based one. A hard-wired unit to compress data had been proposed in [3], which tried to reduce the main memory traffics in the cache-to-memory path. However, they only verified the design by architecture level simulations. Kim et al. implemented a specific low power compression processor for body sensor network [4], which specially aimed at biologic data compression. Their coprocessor based architecture led to the difficulties in software and compiler development. In this paper, we focus on designing a universal scalable architecture of sensor nodes based on a processor with compatible instruction set and CA unit which can be easily reused.

In order to connect the processor and CA, low power communication architecture should be used. There have been plenty of existing SoC bus specifications, such as AMBA [5], Avalon [6], Wishbone [7], GALS [8], etc. However, all of them are too complex for WSNs application. According to the characteristics of CA in sensor node, such as limited data width and power efficiency, the communication protocols should be simple and low power. We proposed a scalable communication architecture for accelerator extension with independent RAM access.

GLSVLSI'09, May 10–12, 2009, Boston, Massachusetts, USA.

3. PROPOSED ARCHITECTURE

Figure 1 shows the proposed architecture of a sensor node after adding the CA units. Each CA unit in the figure denotes a compression hardware accelerator which can compress a section of data.

This architecture is proposed to minimize energy consumption, and its characteristics are summarized as below:

- Separate SoC bus and CA bus. Besides the CA, some other units are needed in sensor nodes, such as interface controllers, IP cores and other hardware accelerators. These units can communicate with the processor via the SoC bus in Figure 1. Since the data compression is a common function module which is used frequently, this feature enables the CA units and the processor to be executed independently.
- Separate processor and data RAM. In this way, CA can access data RAM by the control of an arbiter. The processor only needs to decide which CA unit to execute the compression and which section of data to be compressed. It doesn't interact with the compression process after signaling the CA unit to run.
- The expansibility of CA bus. The number of CA units and the algorithms running on them are configurable. We can use more than one CA unit in one sensor node to compress in parallel. Otherwise, if we need to compress many different kinds of data, a number of CA units realizing various algorithms can be used for different data respectively. Furthermore, any CA unit not in use can be power gated independently.

The protocol of CA bus is shown in Figure 2. We describe the process as the following:

1. The processor decides which CA unit to use and put the CA id on signal *Sel*. The width of this signal is decided by the number of CA units. If there is only one CA in the node, we even don't need this signal in CA bus.
2. The processor gives information to the arbiter before compression, such as the CA id, data location, and some other parameters used in the compression.
3. The processor sets the *Ctrl* line to signal the CA start to work.
4. The CA unit sets the *State* line and starts to compress the appointed data in data RAM. At the same time, the processor can do other work or go into sleep mode.
5. The CA unit pulls down the *State* line to notify processor that the work is done. Then the CA bus returns to the idle state.

The arbiter is designed for controlling the operations of the protocol. It gives the address to RAM and gets the data from RAM. The arbiter can choose the address signal which comes from the processor in normal case or from the CA unit in compressing case. It also can decide to transfer the data to a specific unit: processor or one of the CA units according to the requirement.

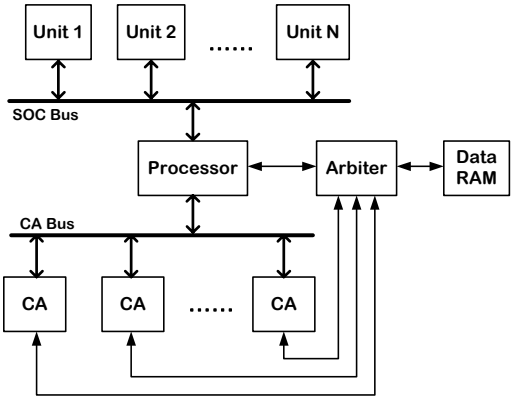

Figure 1. Proposed architecture of a sensor node

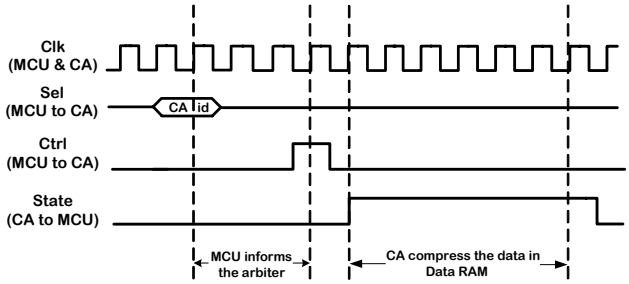

Figure 2. Protocol of CA bus

4. HARDWARE DESIGN

The proposed architecture of a sensor node is implemented in silicon. We first describe the overall node architecture in Section 4.1. Furthermore, the selected compression algorithm and the hardware design of the CA unit are illustrated in Section 4.2.

4.1 Node Architecture

We implement a baseband chip containing a general processor, one CA unit and some other controllers to verify the proposed architecture. The overall architecture of the sensor node is shown in Figure 3. The units are described as followings:

- Processor: we adopt mc8051 as our 8-bit processor. The instruction set is compatible to the industry 8051 microcontroller.
- SPI controller: it is connected to the transceiver to exchange commands and data.
- I^2C controller: it is connected to the sensor to acquire the sampling data from sensors.
- UART controller: it is connected to PC and used for debugging.
- CA unit: it is used for compression operation and the details are described in Section 4.2.
- Wishbone Bus: it connects the processor to other hardware controllers.

It should be noted that only one CA unit in this design will not influence the validation of the proposed node architecture. The interface and control signals are the same for different CA units.

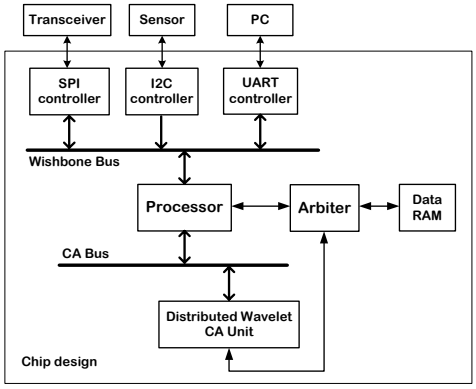

Figure 3. Architecture of our chip design

The only difference between one CA unit and multi-CA units is the setting of signal *Sel* before compression.

4.2 Compression Accelerator

There are some requirements to choose a proper compression algorithm in WSNs. First, the algorithm should have low computation complexity to save battery energy. Second, a distributed compression algorithm is better to balance the power distribution in WSN to prolong its lifetime.

Some spatial distributed compression algorithms have been proposed for sensor networks, such as LTC [9], MCS [10], TREG [11], DPCM [12], but they are too complex to be realized by hardware or need pre-knowledge of the total network. For the low-cost platform, we choose a light computation compression algorithm: Distributed Wavelet Compression [13]. This method has low computation complexity and can be executed in a distributed manner, which is well suitable for the nature of WSN. The algorithm consists of two steps:

1. Wavelet transforms computation

The biorthogonal wavelets are computed based on the lifting scheme. We split the sensor nodes into odd and even subsequences. Each node will compute both the high-pass and low-pass wavelet coefficients. The formulas of the 5/3 wavelet coefficients using lifting are shown as following equations:

$$\begin{cases} d(2n+1) = D(2n+1) - \frac{1}{2}[D(2n) + D(2n+2)] \\ s(2n) = D(2n) + \frac{1}{4}[d(2n-1) + d(2n+1)] \end{cases} \quad (1)$$

where d(n) denotes the high-pass coefficient, s(n) denotes the low-pass coefficient, D(n) denotes the data measured by node n. Partial coefficients are used to guarantee all sensor nodes only need communicate in one direction [13].

2. High-pass coefficients encoding

The output of wavelet transforms computation consists of low-pass and high-pass coefficients. High-pass coefficients are typically very small because the data from spatially close sensors are correlated. So we can represent them using fewer bits. In our design, Run-length encoding (RLE) is used to encode the high-pass coefficients.

Figure 4 shows the architecture of the CA. It includes two components: hard-wired controller and special data-path. The hard-wired controller is a state machine. The special data path

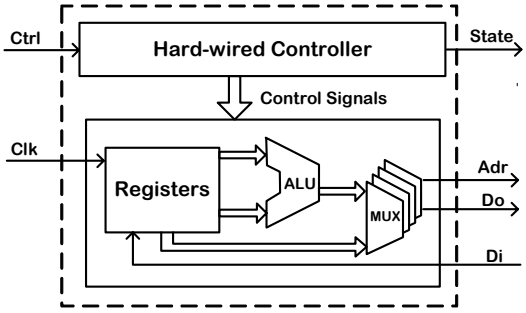

Figure 4. Architecture of the compression accelerator

contains registers, multiplexers and special arithmetic logical units (ALU) designed for the wavelet compression algorithm. It defines some special operations to adapt the algorithm, such as comparison operation for encoding.

Corresponding with the protocol of CA bus, signal *clk* is the system clock. Signal *Ctrl* gets the control command from the processor, and signal *State* notifies processor the states of compression when it is completed. Signals *Adr*, *Do*, *Di* are used to communicate with the arbiter.

5. IMPLEMENTATION AND RESULTS

The sensor node's baseband chip is implemented using 1-poly 6-metal 0.18um CMOS technology, which is shown in Figure 5. The dimensions of overall chip are 1.5mm x 3mm. The volumes of code memory, internal data memory and external data memory are 32 kbyte, 256 byte and 8 kbyte, respectively.

In our design, the CA unit contributes to 5.2% of the total core area. Compared with its significant enhancements in both performance and energy savings, the increased area overhead due to the CA is reasonable.

We also implement a demonstration board system to test the chip. The test environment and the board are shown in Figure 6.

The chip is operating at a supply voltage of 1.8V and a clock frequency of 10MHz. In our experiments, we use real data as the inputs for the compression, which are climate measurements from

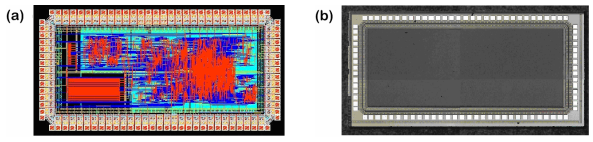

Figure 5. (a) Chip layout in Cadence (b) Die photo

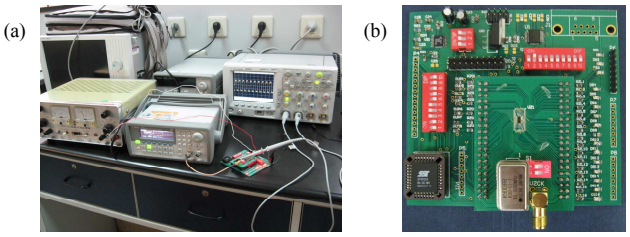

Figure 6. (a) Test environment (b) Demonstration board

moored ocean buoys [14], such as temperature, humidity, etc. The average measured power consumption of CA unit is 0.98mW, whereas the average power consumption is 2.61mW when the same compression algorithm is running on the processor. So the CA unit can save over 62% power.

Figure 7 shows the measurement results when compressing one byte data of different types. It shows that the proposed CA architecture can provide about 96% execution time and over 98% energy consumption savings compared with the pure software solution.

Furthermore, more energy can be saved because of the shorter transmitted data, but we need to add the energy consumption for compression. We assume the compression ratio as 50%, which is the average value of our measurements. Table 1 shows the energy consumption of the node which receives and transmits 1 byte data without or with compression under different transceiver configurations. It is noticed that when the transceiver is ultra-low power, the node without CA can't save energy by software compression because the energy saved due to shorter data transmitting is smaller than the increased energy consumed by compression. However, the proposed CA-based architecture can always provide significant energy savings.

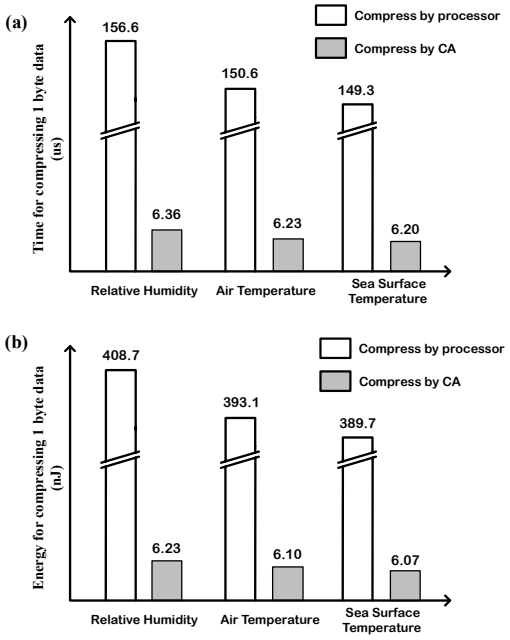

Figure 7. (a) Comparison of execution time (b) Comparison of energy consumption

Table 1. Energy of total node to deal with 1 byte data (nJ)

Transceiver	CC2420	TR1000	ZL70250
Normal Node	2100	1250	206
Compress by processor	1450	1025	503
Proposed Architecture	1081	631	109

6. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a novel architecture of WSN node including a general-purpose processor and tunable number of CA units. We fabricated the sensor node by 1-poly 6-metal 0.18um CMOS technology to verify its low energy consumption features. A spatial distributed wavelet CA is implemented in the node. The chip operates at a supply voltage of 1.8V and a clock frequency of 10MHz. Measurement results show that the proposed architecture reduces over 62% power and 98% energy consumption compared with the pure software solution. Furthermore, the CA-based chip gets significant energy savings while the processor based architecture failed in saving energy due to high computation overhead to compress data for ultra-low power transceivers, such as ZL70250. Our future work is focused on the implementation of control and arbitration with multi-CA units.

7. ACKNOWLEDGMENTS

Authors would like to thank the TAO Project Office of NOAA/PMEL for the database, and HJTC Microelectronics for chip fabrication. This work is partly supported by 863 Program under 2009AA01Z130 and by NSFC under 60871005.

8. REFERENCES

[1] K. Barr and K. Asanovic. Energy aware lossless data compression. MobiSys, pp.231-244, 2003.

[2] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler and K. Pister. System Architecture directions for networked sensors. ASPLOS, pp. 93-104, 2000.

[3] L. Benini, D. Bruni, A. Macii, and E. Macii. Hardware-assisted data compression for energy minimization in systems with embedded processors. DATE, pp.449-454, 2002.

[4] Hyejung Kim, Sungdae Choi and Hoi-Jun Yoo. A low power compression processor for body sensor network. BSN, pp. 65-69, 2007.

[5] ARM Limited. AMBA AXI Protocol v1.0 Specification. 2003, 2004.

[6] Altera Corporation. Avalon Interface Specification. 2005.

[7] Wade D. Peterson. WISHBONE System-on-Chip (SoC) Interconnection Architecture for Portable IP Cores. OpenCores.org, 2002.

[8] C.H. Fernandez, Rajkumar K.Raval and C.J. Bleakley. GALS SOC interconnect bus for wireless sensor network processor platforms. GLSVLSI, pp. 132-137, 2007.

[9] T. Schoellhammer, E. Osterweil, B. Greenstein, D. Estrin and M. Wimbrow. Lightweight temporal compression of microclimate datasets. LCN, pp. 516-524, 2004.

[10] Y.C. Wang, Y.Y Hsieh and Y.C Tseng. Compression and storage schemes in a sensor network with spatial and temporal coding techniques. VTC, pp.148-152, 2008.

[11] T. Banerjee, K. Choudhury, and D. P. Agrawal. Tree based data aggregation in sensor networks using polynomial regression. IEEE FUSION, July 2005.

[12] H. Luo, Y. Tong and G. Pottle. A two-stage DPCM scheme for wireless sensor networks. ICASSP, pp. 661-664, 2005.

[13] A. Ciancio and A. Ortega. A distributed wavelet compression algorithm for wireless multihop sensor networks using lifting. ICASSP, pp. 825~828, 2005.

[14] http://www.pmel.noaa.gov/tao