International Academic Institute for Science and Technology

International Academic Journal of Science and Engineering
Vol. 11, No. 3, 2024, pp. 40-43.

ISSN 2454-3896

International Academic Journal of Science and Engineering

www.iaiest.com

An Energy Conservation Technique Utilizing FPGA in Dynamic Wireless Sensor Networks

Pranay Luthra [a], Kritika Phadke [b]

[a] *Automated Legal Reasoning Unit, Ajmer Institute of Law and Computing, India.*
[b] *Automated Legal Reasoning Unit, Ajmer Institute of Law and Computing, India.*

Abstract

Wireless sensor networks (WSN) have evolved as a transforming tool for large-scale monitoring. WSNs are becoming more and more popular and employed in many different fields, including academic research, environmental monitoring, industrial, military, and surveillance, due to the rising accessibility and sophistication of the underlying technology. The complexity of such applications requires changes in node requirements, which generates Dynamic-WSN (D-WSN). This work presented an Energy Conservation Technique (ECT) used on DWSN utilizing FPGA. For improved operational efficiency, a sensor node could need customization. Therefore, dynamic methods stress enabling a WSN to control energy consumption to improve lifetime, fidelity, and latency. Under Variable Voltage Frequency Conversion (VVFC), the proposed architecture describes the conception and implementation of ECT for WSN under control by a cooperative, customized unit (CCU) with concurrent execution capability on an FPGA. The CCU is based on adjusting operating system (OS) amplification using specialized hardware and then applying it to a soft-core microprocessor (SCMP). This device will reduce the CPU complexity of deploying processor-based sensor nodes in the operating system. The energy control unit controls hardware peripherals, manages the CPU clock, and configures them suitably, depending on the hardware needs of the operations.

Keywords: Wireless Sensor Networks, Energy Conservation, Variable Voltage Frequency Conversion, FPGA, Collaborative Customized Unit.

DOI: 10.71086/IAJSE/V11I3/IAJSE1160
Received: 25 June 2024/Revised: 29 July 2024/Accepted: 29 August 2024/Published: 30 September 2024

Introduction

WSN has evolved into a transforming instrument for continuous observation (Tan et al., 2021). WSNs are becoming increasingly popular and applied in many fields because of the increasing accessibility and sophistication of the technology behind them. The complexity of these kinds of projects calls for a change in node parameters. Strong processing units required for information compression techniques, encryption, and communication protocols drive higher node energy consumption. The power limitations of sensor nodes hamper the correct usage of these battery-powered sensors in WSNs. Many research projects recorded in the literature have been carried out to solve this critical issue WSN nodes (Loukil, 2023) face. The lifetime and value of a WSN node directly depend on the drained state of the battery. WSNs have limited resources; hence, they must be able to rebuild and change with the times and make contextual modifications. Reorganizing means that the system might flexibly adapt the operational mode or application parameters depending on the set constraints. WSNs depend on battery-powered sensors. Hence, their limiting capacity is related to energy limits (Gupta et al., 2021).

Therefore, extending node lifetime depends on reducing energy consumption and improving network dependability. Meeting the strict requirements of energy usage, safety, and dependability depends on controlling the limited power of a WSN node, which presents a difficult task. Moreover, physical sensor nodes are getting more complex and want the integration of strong components to support complex operations, including compression, encryption, and communication protocols, optimizing power usage (Aziz et al., 2022). Effective energy management strategies are urgently required to enable the node to use energy judiciously.

Designing power-efficient wireless sensor networks necessitates an analysis of power consumption characteristics inside the network. Numerous variables contribute to power consumption in wireless sensor networks, including energy consumption from communication protocols and energy use at the sensor node level. The energy-efficient communication protocol is regarded as a significant concern in WSN. A literature study reveals that the proportion of energy needed for transmission to the energy required for processing exceeds one (Gupta & Kumar, 2023).

Considering this, the majority of academics developed energy-saving communication protocols. Nevertheless, the duration needed for information processing at nodes within a WSN significantly exceeds the time necessary for communication at the transmitting node. This results in significant usage of electricity by the CPU in nodes relative to the power needed for transmitting or receiving via a protocol for communication. Therefore, it is essential to diminish energy usage at the WSN node.

Proposed Method

Energy conservation becomes a significant difficulty in WSN nodes. This suggested study selects the Altera FPGA platform because of its adaptability, temporal considerations, and developments in VLSI design. The proposed design for the ECT-based WSN node is displayed in Fig. 1.

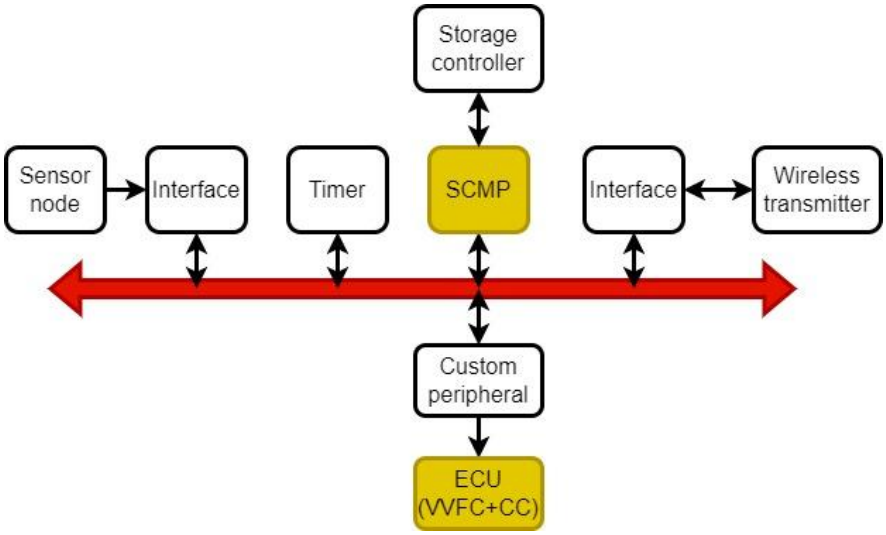


Fig. 1: Proposed design for the ECT-based WSN node

The fundamental principle of notional design is tailoring the SCMP to effectively serve the OS and energy control unit (ECU). The CCU, namely customized peripherals, regulates every peripheral device's active status and manages the program's electrical energy on the CPU. The CCU will alleviate the burden on the operating system by assuming specific duties of commonly conducted processes. Power conservation is achieved by minimizing OS overhead using a CCU and employing an ECU, namely VVFC. VVFC will adjust the rate and voltage according to the application's requirements. Clock control (CC) facilitates the selective activation of peripherals connected to the CPU, allowing them to be toggled between active and inactive states based on the application's device resource needs. A CCU for OS customization and an ECU including VVFC and CC are integrated with the SCMP for energy control.

ECU (VVFC + CC)

The primary role of this module is to deliver regulated frequency and voltage to the SCMP following signalling from the CCU. FPGA implementations use the CC approach to achieve power savings. This device may modify the transmission of the CC signal and deactivate peripheral devices not used for the specific application.

CCU

Constructing an OS for WSN nodes is challenging. Only a limited number of OS, such as Lite OS, Mantis, RET OS, and Enix OS, have been produced in the past 10 years. OS maintenance is crucial for the conception and deployment of WSNs since it facilitates the integration of energy-efficient features inside these networks. Although OS studies and development for WSN is an ongoing topic of inquiry, several trade-offs, such as CPU overhead, energy economy, system dependability, adaptability, and user interface layout, need additional study. This customized device is affixed to the SCMP. This item will enhance the speed of commonly used OS processes. The data about the need for poor performance or optimal condition is delivered to the ECU for additional intervention.

Energy use is often considered a hardware constraint; however, further study has shown that software significantly influences power usage. The structure and features of the OS have an unexpected influence on power usage. The increasing use of WSN systems has increased reliance on OS to abstract intricate hardware. The OS serves as a software intermediary between software applications and physical assets. The OS utilizes most CPU cycles; hence, it is regarded as a computer burden on the CPU inside the WSN node, necessary for managing virtual and physical assets. It was observed that, on the median, around 33% of the system's power had been used by the OS for a specific job. In systems that operate in real-time, timely responses to immediate requests are hindered by the cost of the CPU's Real-Time OS (RTOS). Therefore, minimizing RTOS cost and enhancing its timeliness is a critical concern. OS speed is suggested to mitigate operating system overhead. Four distinct methodologies are proposed for OS inclination:

1. Operating System as a Co-processor: The OS is executed as a co-processor. Consequently, the actions of the OS are delegated to the co-processor, enabling the co-processor to execute complete OS functions.
2. Features of OS in Hardware as an Element: The intricate and comprehensive algorithm for operating system tasks is executed in hardware to minimize OS overhead.
3. Hardware OS: This method entails the complete implementation of the software OS as a hardware entity.
4. OS as tailored commands: Commonly performed functions such as task planning and clock tick control are conducted as customized procedures.

OS speed is achieved by modification of specialized hardware and implementation on SCMPs. The suggested concept utilizes a specific training package and hardware components to accelerate the OS, reducing CPU overhead. This increases computation time for user-defined applications due to decreased OS-CPU overhead. During the procedure, commonly conducted OS activities will be detected and transformed into customized actions (CAs) or hardware components. Thus, an orderly string of commands is converted into reduced procedures, thereby decreasing the CPU cost of the OS. Additional time is allocated for customer job completion. All components of customized commands operate concurrently during execution. The customized command may be accessible in either a high-level or assembly language.

The execution of a customized command or element entails the subsequent steps:

Design of the Framework:

1. Examine the system requirements, determine the often performed operating system functions, such as job planning, and design accordingly.
2. Define the NIOS processing subsystem, including the CPU, accessories, and storage interfaces.

Hardware Flow:

1. Construct a NIOS microprocessor System Model via the QSYS integration tool.

2. Incorporate the HDL design of a bespoke instruction or custom component.
3. Synthesize and build hardware with Quartus II software.
4. Configure for the designated FPGA hardware board.

Software Workflow:

1. Produce C/C++ code for custom instructions or components inside the NIOS processor IDE.
2. Utilize the NIOS-build program to translate the initial code into a usable format.
3. Create drivers and headers for bespoke devices.
4. Troubleshoot the code.
5. Configure the FPGA platform using the compiled file.

The comprehensive outcome of performance metrics, including resource utilization, implementation duration, and dissipated energy, is aggregated to analyze the customized design.

Conclusion

The proposed framework delineates the design and implementation of ECT for WSN using Variable Voltage Frequency Conversion (VVFC) regulated by a collaborative, customized unit (CCU) with simultaneous execution capabilities on an FPGA. The CCU relies on the adaptation of operating system (OS) enhancement via specialized hardware then implements it on a soft-core microprocessor (SCMP). This unit will reduce the CPU complexity of the operating system related to deploying processor-based sensor nodes. The energy regulation unit performs operations like CPU clock regulation, hardware peripheral management, and proper configuration based on the hardware needs of the activities.

References

Tan, B. L., Mok, K. M., Chang, J. J., Lee, W. K., & Hwang, S. O. (2021). RISC32-LP: Low-power FPGA-based IoT sensor nodes with energy reduction program analyzer. *IEEE Internet of Things Journal*, *9*(6), 4214-4228.

Loukil, K. (2023). Energy saving multi-relay technique for wireless sensor networks based on Hw/Sw MPSoC system. *IEEE Access*, *11*, 27919-27927.

Gupta, N., Jain, A., Vaisla, K. S., Kumar, A., & Kumar, R. (2021). Performance analysis of DSDV and OLSR wireless sensor network routing protocols using FPGA hardware and machine learning. *Multimedia Tools and Applications*, *80*, 22301-22319.

Aziz, S. M., Hoskin, D. H., Pham, D. M., & Kamruzzaman, J. (2022). Remote reconfiguration of FPGA-based wireless sensor nodes for flexible Internet of Things☆. *Computers and Electrical Engineering*, *100*, 107935.

Gupta, N., & Kumar, A. (2023). Study on the wireless sensor networks routing for Low-Power FPGA hardware in field applications. *Computers and Electronics in Agriculture*, *212*, 108145.