A proposed software for controlling operating system-dependent functionality

Sundos Abdulameer Alazawi1, Nadia Mahmood Hussien1, Yasmin Makki Mohialden1, Mostafa Abdulghafoor Mohammed2

1Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq
2Department of Arabic Language, Imam Aadam University College, Baghdad, Iraq

ABSTRACT

When the operating system environment temperature rises above safety, the CPU may become unresponsive or even malfunction. To address this problem, to achieve this goal a two-part system was designed. The first, consists of a controlled sensor that constantly monitors the room environment temperature and alerts the user if it rises above acceptable levels for computer use. The second part adopts a Python that uses the OS module, which provides a portable interface for OS-dependent tasks and shuts down the device to prevent it from behaving unexpectedly. A series of experiments at different temperatures demonstrated the ability of the device to alert the user.

Keywords: CPU, Microcontroller, Operating system, OS module, Python, Sensor, Temperature sensor

1. INTRODUCTION

Sensor-based real-time event monitoring is gaining popularity as the internet of things grows [11]–[13]. A temperature monitoring system with device control is required [14]. Saha et al. [14] described a practical prototype system for real-time monitoring systems that uses a DS18B20 temperature sensor, an Arduino Uno, and a GSM module to send and receive messages. This prototype collects and analyzes sensor data in order to provide appropriate input to a software system that autonomously controls the functions of the operating system. Monitoring the CPU temperature is critical for efficiency. Utilize information about the temperature of the CPU and changing environmental conditions. A second temperature sensor is required to monitor the thermal environment of the processor [15].

Python's OS module gives you access to the OS. A Python utility module that comes pre-installed with the OS. This module gives portable access to specific OS functions. It allows interaction with the OS. It is a standard utility module in Python System-dependent functionality is portable with this module [16]–[28]. The OS and OS.path modules provide several file system and Python operations [15], [29]. Because most temperature-aware systems rely on CPU temperature estimations or measurements, temperature projection is an essential component of them [30].

A number of related research works are described here, highlighting a comparative viewpoint based on their contributions. Research by Hassan et al. [31] show the result in the creation of a non-invasive, customized brain cooling device. The recommended helmet utilizes thermoelectric cooling components to keep the user's head cool to increase the operating time of the proposed system while maintaining a constant

Journal homepage: http://beei.org
A proposed software for controlling operating system-dependent functionality

(Sundos Abdulameer Alazawi)
A temperature sensor is a device that detects the temperature of the surrounding environment. The GSM model generates an alarm when the computer is turned off. The temperature was displayed on an LCD. The dependency functionality system, which represents the operating system's interface functions, is depicted in the second section. The OS module is a typical Python utility module. This module gives you access to operating system-specific functions when you're on the go. The first component, which measures the ambient temperature, starts the functioning mechanism.

The second part of the procedure begins when the temperature reaches 75 degrees Celsius. The computer's operating system is run in the second part, which also automatically shuts down the machine if the temperature sensor reaches 75 degrees. In addition, the user will be notified of an alarm. Figure 2 shows the components of the first segment.

![Figure 2. The components of the first segment of the system](image)

2.1. Hardware components
2.1.1. Arduino microcontroller board
Input and output pins include 14 digital (six PWM), 6 analog, and a 16 MHz ceramic resonator (CSTCE16M0V53-R0). Start by connecting it to a USB port, an AC adapter, or a battery. The Arduino Uno R3 was used in this study [35]–[38].

2.1.2. GSM module-global system for mobile communication
Mobile devices and computers can communicate with a GSM network through the use of a GSM module, which is a chip or circuit. As a bonus, it can be accessed online. Among the several types of liquid crystal displays, LCDs are a specific subtype. "LCD" stands for "liquid crystal display," a display technology that is utilized in computer monitors and mobile devices such as laptops, tablets, and smartphones. It displays sensor results and alarm signals for this inquiry [39], [40].

2.2. Software components
The Arduino Uno is programmed using version 1.8.12 of the Arduino IDE. The software is compatible with Microsoft Windows, Mac OS X, and Linux. The Java-based system makes extensive use of open-source technologies such as processing [41]–[43].

a. OS module: This module is included with the standard Python library, therefore there is no need to install it separately. This module contains procedures for interacting with the operating system. We'll utilize the OS module in this tutorial to shut down the computer device after receiving an alarm from the Arduino sensor when the temperature sensor records an unacceptably high temperature [44]–[46].

b. The use case diagram illustrates how the system works. The model is made up of actors and use cases that interact with one another. Figure 3 illustrates the use case diagram with the associated relationships.
A proposed software for controlling operating system-dependent functionality (Sundos Abdulameer Alazawi)

3. RESULTS AND DISCUSSION

The system is being tested in a variety of situations. The output of sensors is depicted on an LCD in Figures 4(a), (b) and Figure 5. Example 2 of a system that is being tested the computer will shut down if the temperature rises above the preset temperature threshold. The CPU time calculated throw CPU Time=I*CPI/R. Figure 6 shows the CPU performance.
4. CONCLUSION

Real-time dependent functionality is provided by an embedded system. Determinations are made based on the presence or absence of a threshold that the system monitors. In terms of the Arduino microcontroller, it is in command of all that is happening. The GSM module is a fantastic tool for telecommunications, and the second part of the system makes use of an OS module instruction to shut down the machine when the temperature exceeds the allowable value in the first portion of the program. Tests conducted at various temperatures have demonstrated the system’s efficacy.

ACKNOWLEDGMENT

The Authors would like to thank Mustansiriya University (https://uomustansiriya.edu.iq/) Baghdad -Iraq for its support in the present work.

REFERENCES


**BIOGRAPHIES OF AUTHORS**

**Sundos Abdulameer Alazawi** is Assistant Professor at the Department of Computer Science, College of Science, Mustansiriyah University. She holds a PH.D. from Iraqi Commission for Computers and Informatics in 2021 with specialization in operating systems. Her research areas are operating systems, internet of things, and image processing. She can be contacted at email: ss.aa.cs@uomustansiriyah.edu.iq.

**Nadia Mahmood Hussien** is Assistance Lecturer at College of Science, Mustansiriyah University, Iraq. She holds a Master degree in Computer Science with specialization in software engineering. Her research areas are software engineering, artificial intelligence, internet of things. She can be contacted at email: nadia.cs89@uomustansiriyah.edu.iq.

**Yasmin Makki Mohialden** is Lecturer at College of Science, Mustansiriyah University, Iraq. She holds a Master degree in Computer Science with specialization in Software Engineering. Her research areas are Software Engineering, Artificial Intelligent, Internet of Things and Cloud Computing. She is head of computer center in the College of Science. She can be contacted at email: ymmiraq2009@uomustansiriyah.edu.iq.

**Mostafa Abdulghafoor Mohammed** currently works at the Al-Imam Al Aadham University college. Mostafa does research in information technology, computer communications (networks), cloud computing and communication engineering. His current project is ‘offloading in mobile cloud computing’. He has finish his master from computer science department at BAMU university, India, and he finish Ph.D in Computer science and IT at University polytechnic of Bucharest, Romania. He can be contacted at email: alqaisy86@gmail.com.