Suggest and Design a Minimal CPU Architecture for Controlling the Washing Machine.

Jonathan Guan, Clive Lewis, Liu Xia, Kalidinadi Krishna Kishore

**Seidenberg School, Pace University**

**CS 604: Computer Systems and Concepts**

Section: 73438

Instructor: Rayal Johnson

Week 10/11

**Abstract**

This project proposes the design and implementation of a minimum CPU structure, using a microcontroller model with a Java-based software program, to control the operation of a bathing machine. The primary objective is to create an efficient and reliable control system for a washing machine. The system should have full access to handle mutable tasks such as water level detection, washing times, temperature controlling, motor speed control, timer, rinse speed, etc.

The project will be executed in a couple of phases, starting with the design phase, involving the creation of a comprehensive CPU architecture, encompassing CPU design, memory allocation, and I/O port configuration. The subsequent implementation phase will utilize Java to translate these designs into functional API. then will debug and test to ensure the reliability of the system.

Expected results include a completely functional minimum CPU structure for washing system control, providing efficient multitasking abilities. Efforts to set an excessive level of inclusiveness, and allow the integration of more advanced control algorithms in the future.

*Keywords:*

Minimum CPU structure, Microcontroller, Java-based software, Washing machine, Efficiency, CPU architecture, Memory allocation, I/O port configuration, Implementation, API, Multitasking, Advanced control algorithms, and integration.

Title:

Suggest and Design a Minimal CPU Architecture for Controlling the Washing Machine.

Objective:

This project aims to create an efficient and reliable control system for a washing machine. The system should have full access to handle mutable tasks such as water level detection, washing times, temperature controlling, motor speed control, timer, rinse speed, etc.

Theory:

The earliest laundry machines simply applied a laundry action once loaded with clothing with hot water and commenced.

The washing machine is supposed to have the following functional modes:

Automatic Mode: the system operates independently as soon as it is started and should notify the user that the job is complete when the work is done. This mode instantly senses the quality of the laundry and the requirements for water volume and temperature, as well as the capacity of the laundry and the wash cycle and performs the appropriate operation.

Manual Mode: In this mode, the user must specify the operation and provide the control system with the relevant information. The system asks the user to enter the washing time, water volume, and load. After entering these data, the user should start the machine. When the specified operation is completed, the system should notify the user. If the user only wants to rinse, he must manually select the "rinse" option.

Design:

Create a microcontroller-based CPU architecture that can control the washing machine. This architecture will include the microcontroller, memory allocation, and I/O ports, ensuring the washing machine can be well controlled by different functions.

* **Microcontroller Unit:**

A microcontroller chip is the core of the system that could have features and processing capabilities.

* **Memory Allocation**:

Memories are allocated within the system.

Program memory: storing control algorithms.

Data memory: variables and settings.

* **I/O Ports:**

Input and output ports are used for interfacing with external components.

* **Control Algorithms:**

Represent the flow of control algorithms within the microcontroller.

E.g.

Water level detection

Temperature control

Motor speed control

Timer

* **User Interface:**

Allow users to change settings and timing for wash cycles.

E.g.

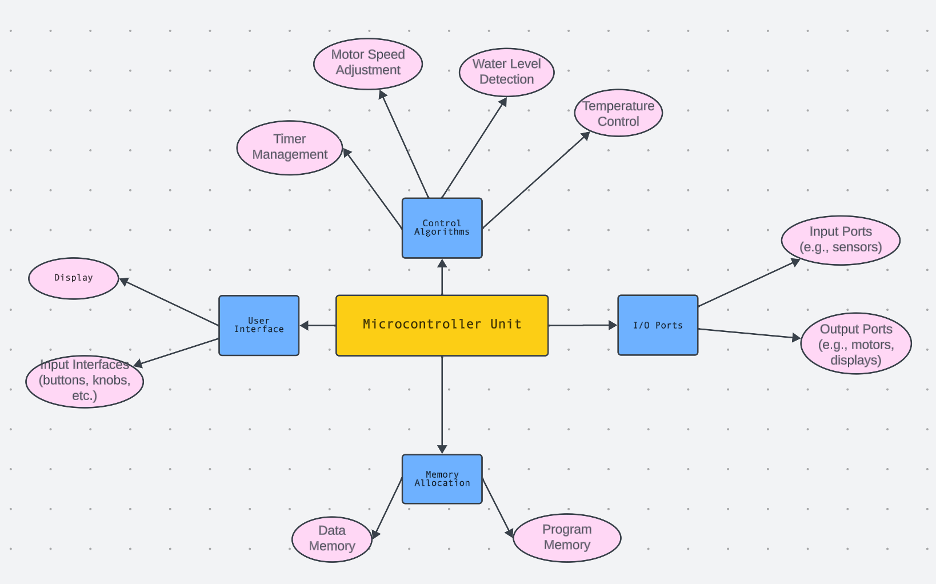
display screen and input interface.

* **Multitasking Capabilities:**

The system needs to handle multiple tasks at the same time.

E.g.

Monitoring water temperature while running the timer.



Software Implementation:

Java was the choice of programming language due to its portability and compatibility across various format factors of devices. This is possible due to Java’s nature of being a compiled language for conversion to Java Bytecode and execution of the bytecode in Java's portable interpreter. We implement the code in a format such that the user inputs the temperature using a fixed dial to input either 25, 30, or 40 deg Celsius and set power efficiency to on or off.

If an invalid input is provided for the temperature the code throws an error to avoid any mishap of the program.

The class ‘WashingMachineController1’ is designed for the purpose of orchestration distinct sections with a washing cycle. The washing cycle comprises a series of sequential steps by doing the following phases:

1. Start cycle
2. Water intake
3. Wash
4. Rinse
5. Spin Dry
6. Stop all processes

Each step is implemented as a function in our class and when function “start()” is called all the proceeding steps are called accordingly until the “stop()” function is called post which functions named “displayEnergyConsumption()” and “displayWaterUsage()” which are implemented in the same class to track the power and water usage of the cycle are called to display the respective information.

Debugging-Test-Run:

In the main class, three distinct test cases have been incorporated to assess a range of scenarios within the codebase.

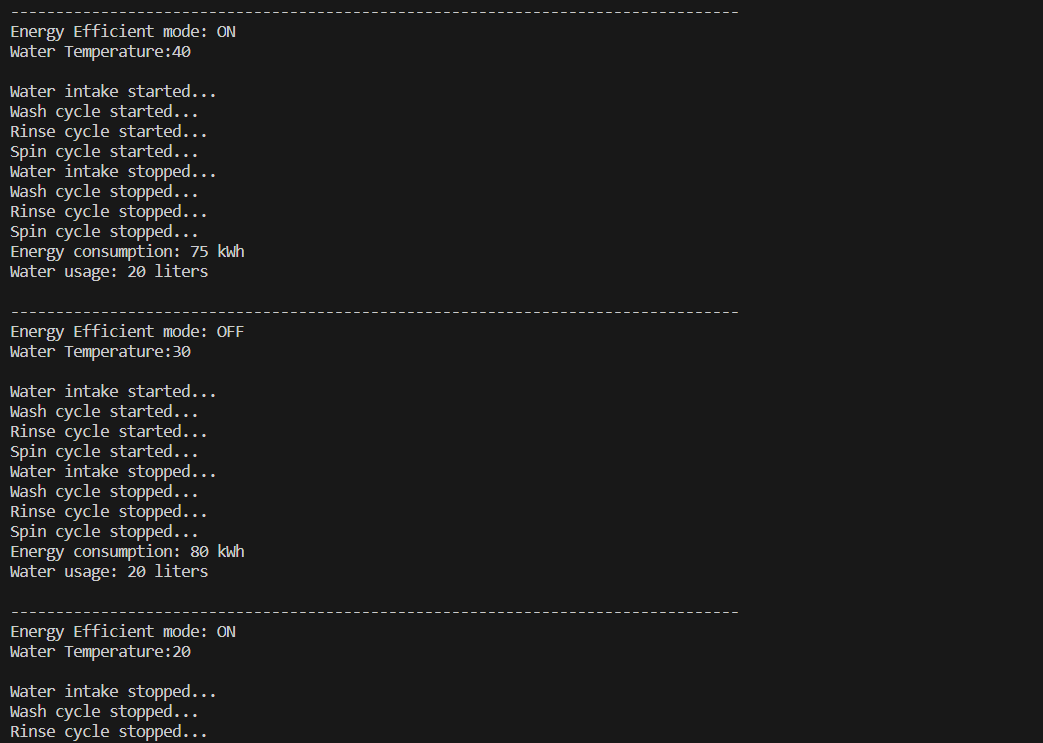
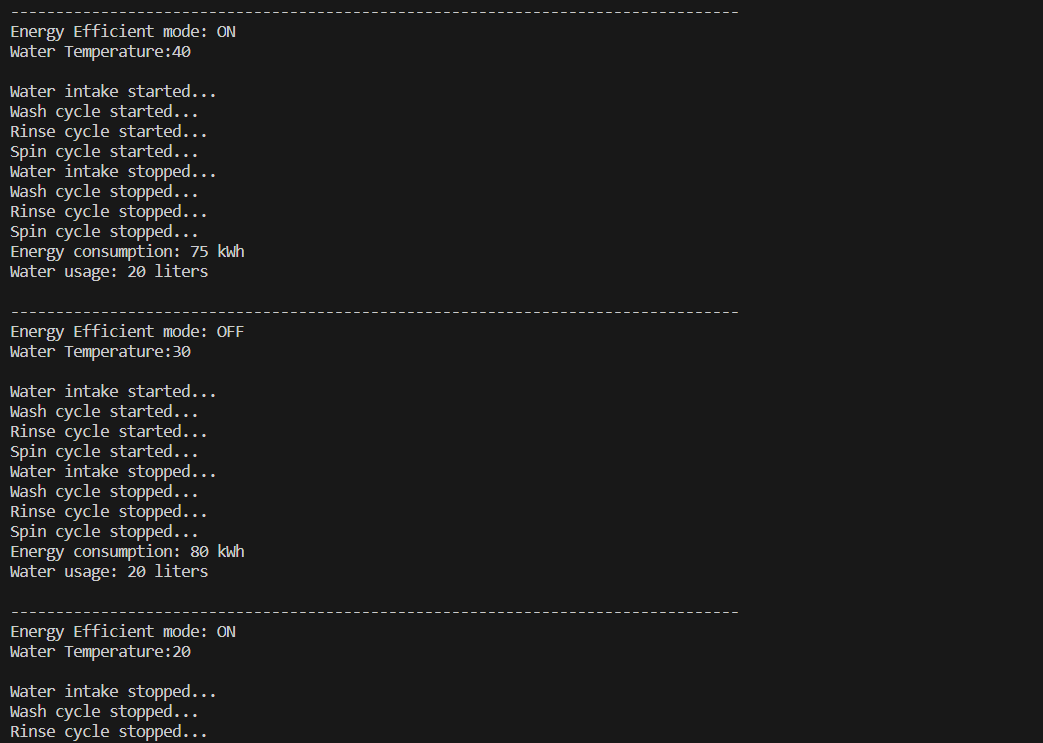
1. Temp: 40 energy efficiency: on
2. Temp: 30 energy efficiency: off
3. Temp: 20 energy efficiency: on

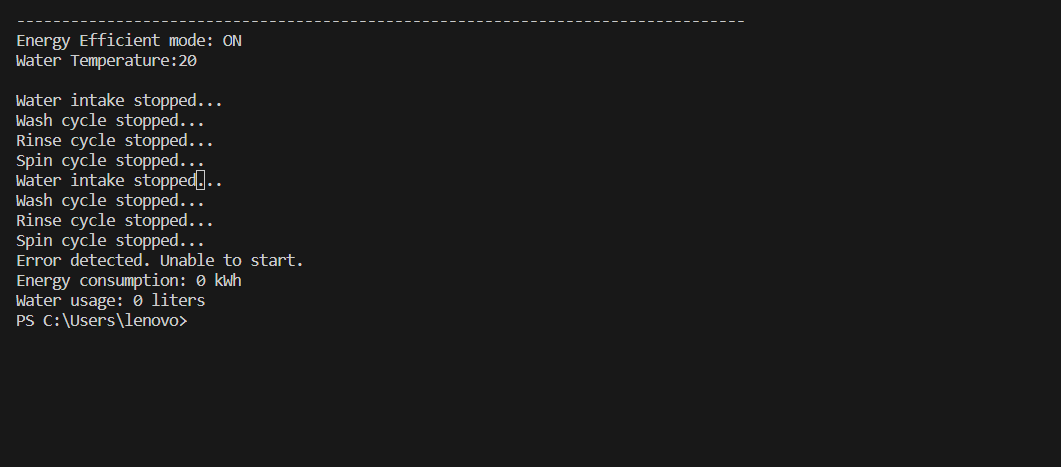
The first case tests an allowable temperature with energy efficiency mode on.

The second case tests an allowable temperature with energy efficiency mode off.

The third case is a temperature that is not allowed which should ideally throw an error.

In the following case, the output of the conducted test results is presented:

****

****

Results Analysis:

1. The washing machine is designed only to accept water temperature settings of 25, 30, and 40 degrees Celsius. Selecting any other temperature will trigger an error, preventing the initiation of the water intake process.
2. The energy consumption of the washing machine is influenced by both the water temperature and the energy consumption mode (on/off). Higher water temperatures result in increased energy consumption, making it directly proportional. Energy consumption mode will help to reduce the consumption when the mode is on.
3. If no error is detected, the washing machine will operate with a minimal energy consumption of 10 kWh. This represents the lowest energy usage scenario, ensuring efficient performance and adherence to the specified energy consumption level when the machine functions without issues.

Future Improvement:

1. Design different washing models to meet various needs:

Quick Wash Model: the washing time and washing times will be reduced under this model.

Large Capacity Model: users can have both automatic and manual control over water level, washing times, rinse time, and dry time under this model.

Customized model: users can choose how many times to wash and rinse on the washing machine setting.

Heavy-duty model: the machine will automatically add more time.

Gentle wash model: The motor of the washing machine will slower adjust the specific fragile clothes washing requirements.

Child lock safety model: Child lock safety feature.

Time setting model: users can choose the time to run the washing machine in advance.

1. Additional functions of dry level choice and detergent manually setting into the design of the washing machine such as advanced dry level choice, a quick dry model with dry level choice, and detergent automatic control setting These functions can improve the performance of the washing machine.
2. Application control: The users can remotely manipulate the machine by choosing different models.

Conclusion:

This project serves as a valuable in advancing our expertise in washing machine design, specifically focusing on the minimal CPU architecture of washing machines. Furthermore, it contributes to a broader comprehension of diverse machine designs, extending its applicability to areas such as computer design during the research. We aim to elevate our mastery in the field of CPU design, with a particular emphasis on enhancing our understanding of both existing and innovative washing machine technologies.

Bibliography:

1. D Patterson and J. Hennessy, (2017), *Computer Organization and Design: The hardware/Software Interface*. ARM Edition, Elsevier-Morgan Kauffmann.
2. Ament, P (2005). *History of Washing Machine*. http://www.ideafinder.com. Engdahl, T (2005). *Parallel Port Interfacing Made Easy*. http://www.epanorama.net /circuits/parallel\_output.html.
3. Akpobi, JA; Ayeke, E O (2003). *Design and Manufacture of A Horizontal Axis Automatic Washing Machine with Emphasis on the Critical Parts*. Journal of Engineering for Development, Vol. 5. pp. 9-16.
4. AKPOBI, J A,OKONTA, J A. June( 2007). *Design and Construction of a Computer Controlled Clothes Washing Machine*. Journal of Production Engineering Department Vol. 11 (3) 249 - 251
5. Milecki, A., & Regulski, R. (2017). Washing Machine Controller with a new programming method*. Acta Mechanica et Automatica, 11(4),* 328–332*.* https://doi.org/10.1515/ama-2017-0051