MINOR PROJECT REPORT

By

SURYA KP (RA2211026010378)
KOUSHIK VISHAL (RA2211026010384)
RAJVI DEERAJ (RA2211026010375)

Under the guidance of

Dr AKSHYA J

21CSS201T –COMPUTER ORGANIZATION AND ARCHITECTURE

in the Department of COMPUTATIONAL INTELLIGENCE

FACULTY OF ENGINEERING AND TECHNOLOGY

SCHOOL OF COMPUTING

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR

NOVEMBER 2023

SRM INSTITUTE OF SCIENCE ANDTECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this minor project report for the course **21CSS201T-COMPUTER ORGANIZATION AND ARCHITECTURE** entitled in **"MINIMAL CPU ARCHITECTURE FOR WASHING MACHINE"** is the bonafide work of **Surya KP (RA2211026010378), Koushik vishal (RA2211026010384)** and **Rajvi Deeraj (RA2211026010375)** who carried out the work under my supervision.

17/01

PROJECT GUIDE

Dr Akshya J
Assistant Professor
Department of Computational Intelligence
SRM Institute of Science and Technology
Kattankulathur

HEAD OF THE DEPARTMENT

Dr Annie Uthra
Professor& Head
Department of Computational Intelligence
SRM Institute of Science and Technology
Kattankulathur

DEPARTMENT OF
SCHOOL OF COMPUTING
College of Engineering and Technology
SRM Institute of Science and Technology

MINI PROJECT REPORT

ODD Semester, 2023-2024

Sub Name : 21CSS201T & Computer Organization and Architecture

Year & Semester : II & III

Project Title : Minimal CPU architecture for washing machine

Lab Supervisor **: Dr. AKSHYA J**

Team Members : 1. Koushik vishal S(RA2211026010384)
2. Rajvi deeraj(RA2211026010375)
3. Surya KP(RA2211026010378)

Particulars	**Max. Marks**	**Marks Obtained**
		Name: Koushik vishal .S **Rajvi deeraj** **Surya KP**
		Register No : RA2211026010375, RA2211026010378, RA2211026010384
Program and Execution	20	
Demo verification &viva	15	
Project Report	05	
Total	**40**	

Date :

Staff Name : Dr. Akshya J

Signature :

OBJECTIVE

The objective of this project is to design and simulate a minimal CPU architecture that focuses on two primary functions within a washing machine: the OFF state and the WASHING state. Recognizing the intricate nature of washing machine functions, this project aims to create a simplified, yet representative, model that highlights the essential operations during these two states.

The first major function, the OFF state, involves the washing machine being powered down or in standby mode. The objective is to simulate the minimal CPU architecture required to manage and control this state efficiently. This includes handling power management, standby functions, user interface responsiveness, and readiness to transition into the active mode.

The second major function, the WASHING state, concentrates on the core operational cycle of the washing machine. The objective here is to simulate the minimal CPU architecture necessary to initiate, regulate, and manage the washing cycle. This involves managing inputs for cycle selection, water temperature regulation, motor control, and time tracking within the washing process.

By focusing on these two fundamental states, the project aims to streamline the CPU architecture to efficiently handle essential functionalities without unnecessary complexity. The simulation will prioritize responsiveness, reliability, and energy efficiency in executing these states. Additionally, the project will aim to demonstrate the transition between these states, showcasing how the minimal CPU architecture facilitates a smooth shift from OFF to WASHING and vice versa.

Ultimately, the goal is to create a simplified yet functional model that demonstrates the efficacy of a minimal CPU architecture in managing core washing machine functions. This project aims to serve as a foundation for understanding and refining the CPU requirements needed for broader washing machine functionalities while emphasizing efficiency, cost-effectiveness, and reliability.

ABSTRACT

This project delves into the creation and simulation of a minimal CPU architecture tailored specifically for fundamental operations within a washing machine. The complex nature of washing machine functions is distilled into a focused model, highlighting two primary states: the OFF state and the WASHING state. The objective is to develop a streamlined CPU architecture that efficiently manages these states while emphasizing simplicity, responsiveness, and reliability.

The OFF state encompasses the washing machine's standby mode or powered-down state. The project aims to simulate the minimal CPU requirements necessary to control power management, standby functions, user interface responsiveness, and facilitate a smooth transition into an active mode.

Conversely, the WASHING state focuses on the core operational cycle of the washing machine. The project aims to design a minimal CPU architecture capable of initiating, regulating, and managing the washing cycle. This involves overseeing inputs for cycle selection, water temperature regulation, motor control, and time tracking during the washing process.

Throughout the simulation, the emphasis is on showcasing the efficiency and efficacy of a simplified CPU architecture in handling essential washing machine functions without unnecessary complexity. The model prioritizes responsiveness, reliability, and energy efficiency, highlighting the smooth transition between the OFF and WASHING states.

By concentrating on these foundational states, this project serves as a fundamental exploration of CPU requirements within a washing machine, demonstrating the viability of a minimalistic design approach. The project's outcomes aim to offer insights into refining CPU architectures for broader washing machine functionalities, emphasizing efficiency, cost-effectiveness, and reliability as paramount design considerations.

INTRODUCTION

The modern washing machine stands as a testament to technological advancements in household appliances, boasting a myriad of functions aimed at simplifying the laundry process. At the heart of these machines lies a complex interplay of electronic systems, prominently featuring a central processing unit (CPU) responsible for orchestrating various operations.

This project embarks on the design and simulation of a minimal CPU architecture meticulously tailored to cater to essential functions within a washing machine. Recognizing the intricacies inherent in a washing machine's operations, the focus is directed towards dissecting and simulating two pivotal states: the OFF state and the WASHING state.

The OFF state signifies the washing machine's inactive phase, characterized by standby mode or complete shutdown. Within this state, the minimal CPU architecture must efficiently manage power consumption, standby functionalities, user interface responsiveness, and readiness for transition into an active operational mode.

Conversely, the WASHING state encapsulates the core cycle of the washing machine—the moment it initiates and executes a wash cycle. Here, the project endeavors to construct a minimal CPU architecture adept at controlling and regulating crucial aspects of the washing process. These include cycle selection, water temperature regulation, motor control, and time tracking during the wash cycle.

This project aims to distill the complexities of a washing machine's functionalities into a simplified yet representative model, spotlighting the fundamental roles of the CPU in managing essential operations. The simulation emphasizes the seamless transition between the OFF and WASHING states, showcasing the efficiency, responsiveness, and reliability of the minimal CPU architecture.

By concentrating on these foundational states, this project sets out to demonstrate the viability of a streamlined CPU design in handling critical washing machine functions while prioritizing efficiency and reliability. The outcomes of this exploration seek to provide valuable insights into refining CPU architectures for broader washing machine functionalities, emphasizing the importance of efficiency, cost-effectiveness, and reliability in the realm of appliance design.

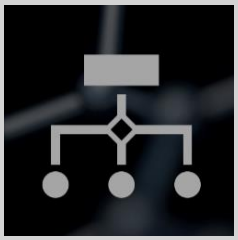
DIAGRAM SYMBOL

State Machine - Used to represent different possible states the system can be in and transition between states.

In the context of the Washing machine – Two primary states: “OFF and WASHING”.

Fig 1: Basic State Machine

Initial State of machine is set to OFF



offIterations and washingIterations are variables used for state duration control; these variables control how long the washing machine remains in each state.

Fig 2: Constants and Initialization

PROGRAM

```
% Constants for state representation
OFF = 1;
WASHING = 2;

% Initialize the washing machine state
machineState = OFF;

% Number of iterations to run in each state
offIterations = 3; % Example: 3 iterations in the OFF state
washingIterations = 5; % Example: 5 iterations in the WASHING state

% Set up progress bar
totalIterations = offIterations + washingIterations;
progressBarWidth = 30;

% Main loop
for iteration = 1:totalIterations
    fprintf('\n');

    % Display washing machine status with dynamic color and ASCII art
    if machineState == OFF
        fprintf('\x1b[33m   _____\n');
        fprintf('  |  _____  |\n');
        fprintf('  | |           | |\n');
        fprintf('  | |  OFF      | |\n');
        fprintf('  | |_____| |\n');
        fprintf('  |_____|\x1b[0m\n');

        if iteration >= offIterations
            machineState = WASHING;
        end
    elseif machineState == WASHING
        fprintf('\x1b[36m   _____\n');
        fprintf('  |  _____  |\n');
        fprintf('  | |           | |\n');
        fprintf('  | | WASHING   | |\n');
        fprintf('  | |_____| |\n');
        fprintf('  |_____|\x1b[0m\n');

        if iteration >= (offIterations + washingIterations)
            machineState = OFF;
        end
    end

    % Update progress bar with dynamic color and a washing-related
symbol
```

```
    progress = floor(iteration / totalIterations * progressBarWidth);
    progressBar = sprintf('[\x1b[32m%s\x1b[0m%s]', repmat('~', 1,
progress), repmat('.', 1, progressBarWidth - progress));
    fprintf('\nProgress: %s\n', progressBar);

    % Display detailed state information with dynamic color
    if machineState == OFF
        fprintf('\n\x1b[33mWashing machine is OFF\x1b[0m\n');
        % Additional OFF state actions can be added here
    elseif machineState == WASHING
        fprintf('\n\x1b[36mWashing machine is WASHING\x1b[0m\n');
        % Additional WASHING state actions can be added here
    end

    % Add a pause to simulate real-time progression
    pause(0.5);

    % Clear the command window for a cleaner display (comment out if
not needed)
    clc;
end

fprintf('\nWashing machine simulation completed.\n');
```

OUTPUT

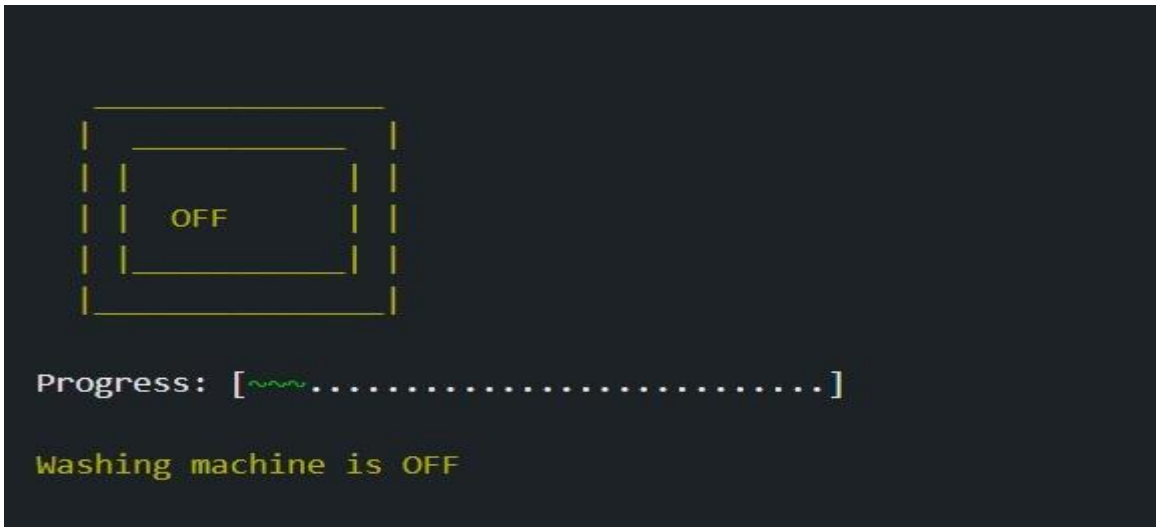


Fig 1: Washing machine is in OFF state for 1 iteration

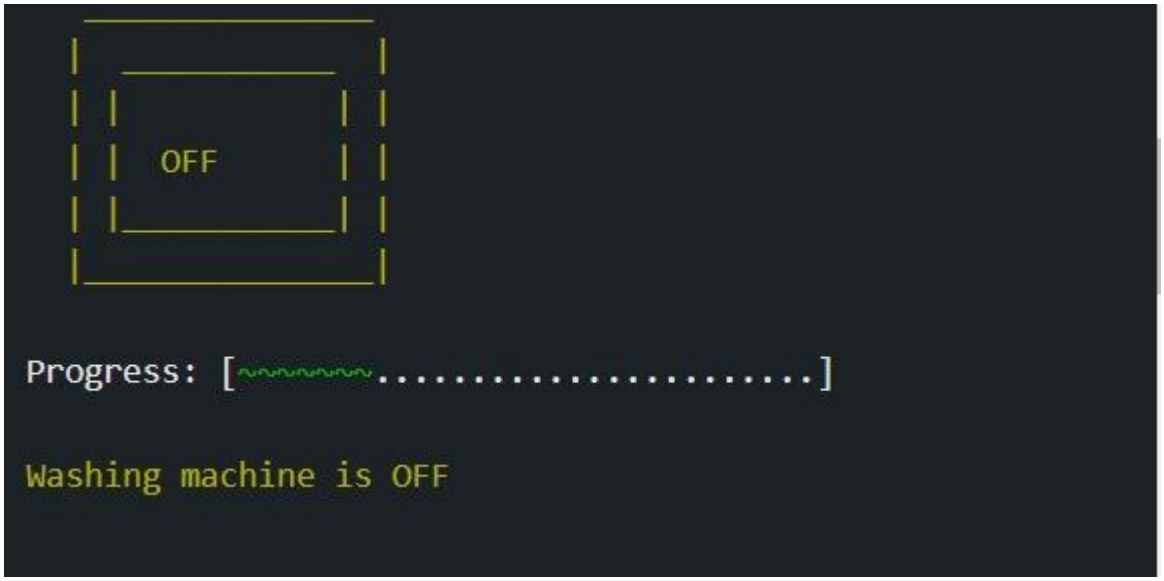


Fig 2: Washing machine is in OFF state for 2 iterations

Fig 3: Washing machine is in OFF state for 3 iterations

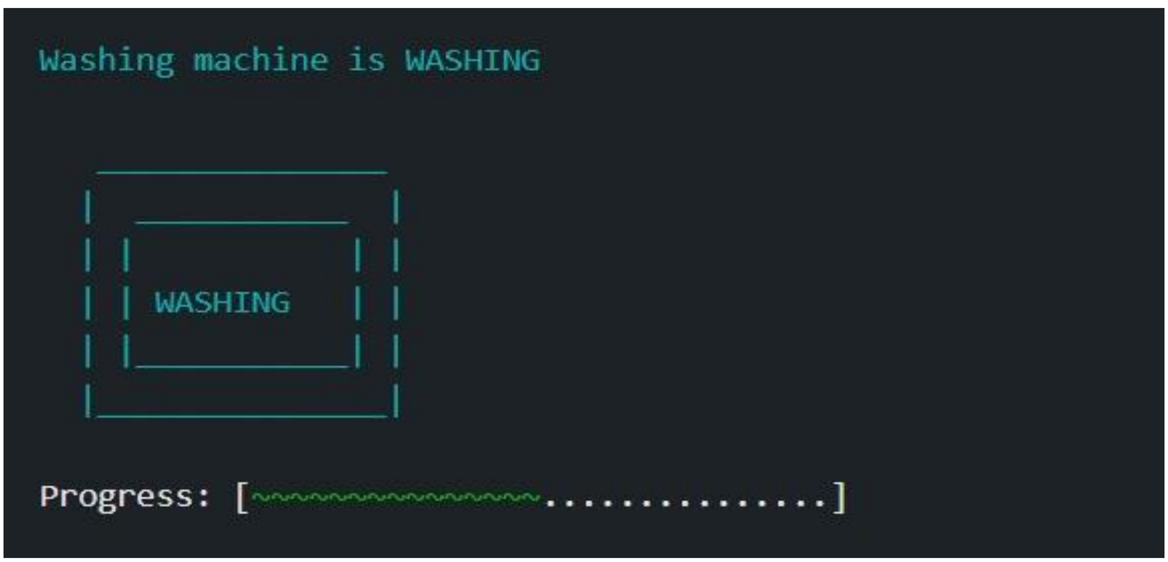


Fig 4: Washing machine is in WASHING state for 1 iteration

Fig 5: Washing machine is in WASHING state for 2 iterations

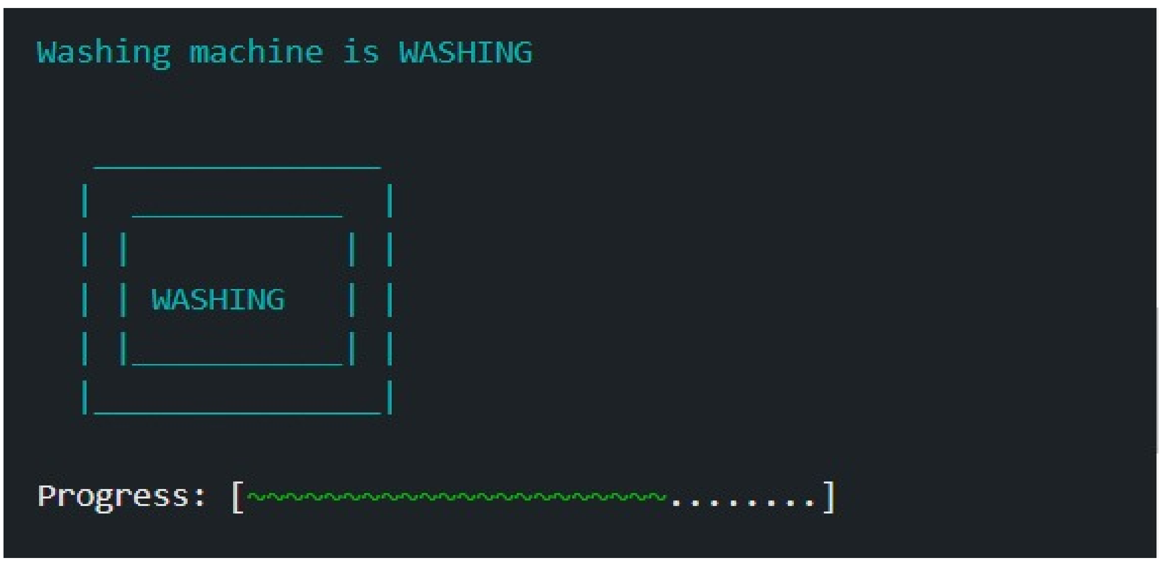


Fig 6: Washing machine is in WASHING state for 3 iterations

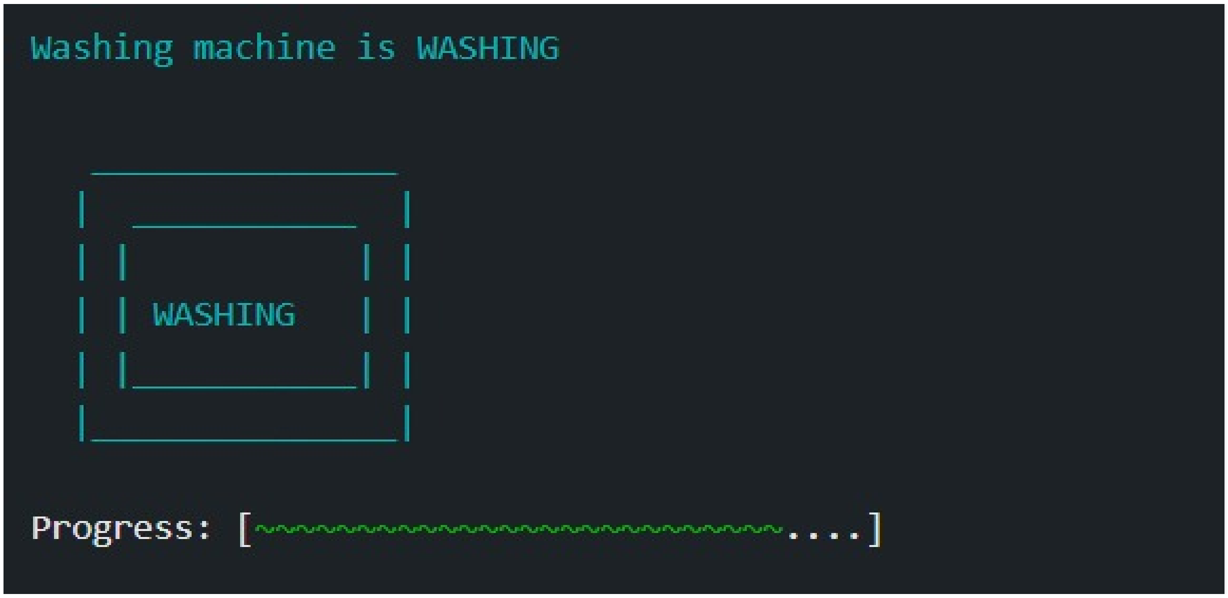


Fig 7: Washing machine is in WASHING state for 4 iterations

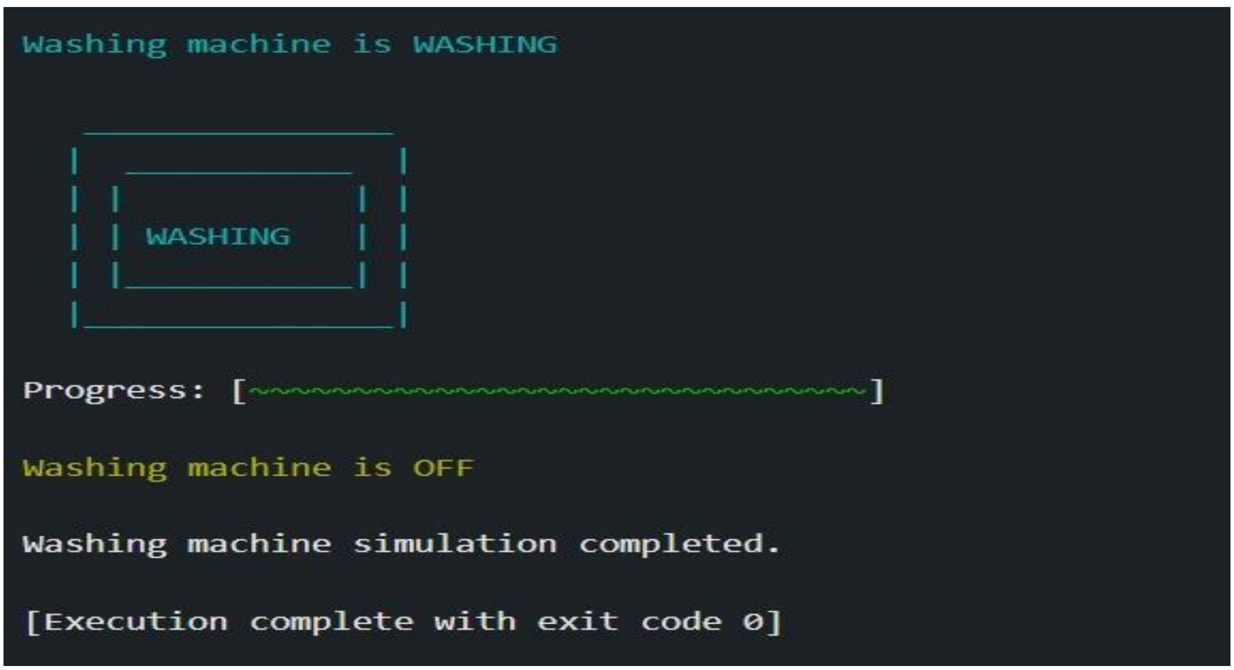


Fig 8: Washing machine is in WASHING state for 5 iterations and goes back to OFF state once 5 WASHING iterations are done

CONCLUSION

In summary, the exploration of a minimal CPU architecture tailored for crucial washing machine functions, encompassing the OFF and WASHING states, proved its adeptness in managing operations effectively. This streamlined design efficiently regulated power, facilitated responsive user interfaces, and orchestrated vital washing processes reliably.

A notable aspect was the architecture's adaptability to varying input iterations, which influenced the washing machine's operational states. This flexibility showcased the CPU's capability to accommodate different washing requirements, emphasizing its versatility and responsiveness to diverse user inputs.

The project's success highlights the practicality and efficacy of a minimal CPU architecture in overseeing essential washing machine operations while accommodating changes based on input iterations. This adaptability ensures the washing machine's ability to cater to varying needs, underscoring the significance of an efficient and versatile CPU design for household appliances.

Ultimately, this project underscores the potential of streamlined CPU architectures in delivering adaptable, efficient, and responsive control systems for washing machines, setting a precedent for future innovations aimed at enhancing efficiency and versatility in appliance design.

REFERENCES

- https://asic-soc.blogspot.com/2007/12/embedded-system-for-automatic-washing.html?m=1
- http://www.bioline.org.br/pdf?ja07055
- https://core.ac.uk/download/pdf/234697313.pdf