EMERTXE

EMBEDDED SYSTEMS INTERNSHIP PRESENTATION

# Simulation of Washing Machine using PIC16F877A

**Presented by :-**
**Name : Priyanshi Singh**
**College : JKIAPT, University of Allahabad**

# CONTENTS

- Introduction of C Programming
- Embedded Systems Concepts
- Memory Classifications
- Microcontroller
- Tools Used
- Washing Machine Requirements
- Implementation Process
- Washing Machine Simulation
- Learning Outcomes & Conclusion

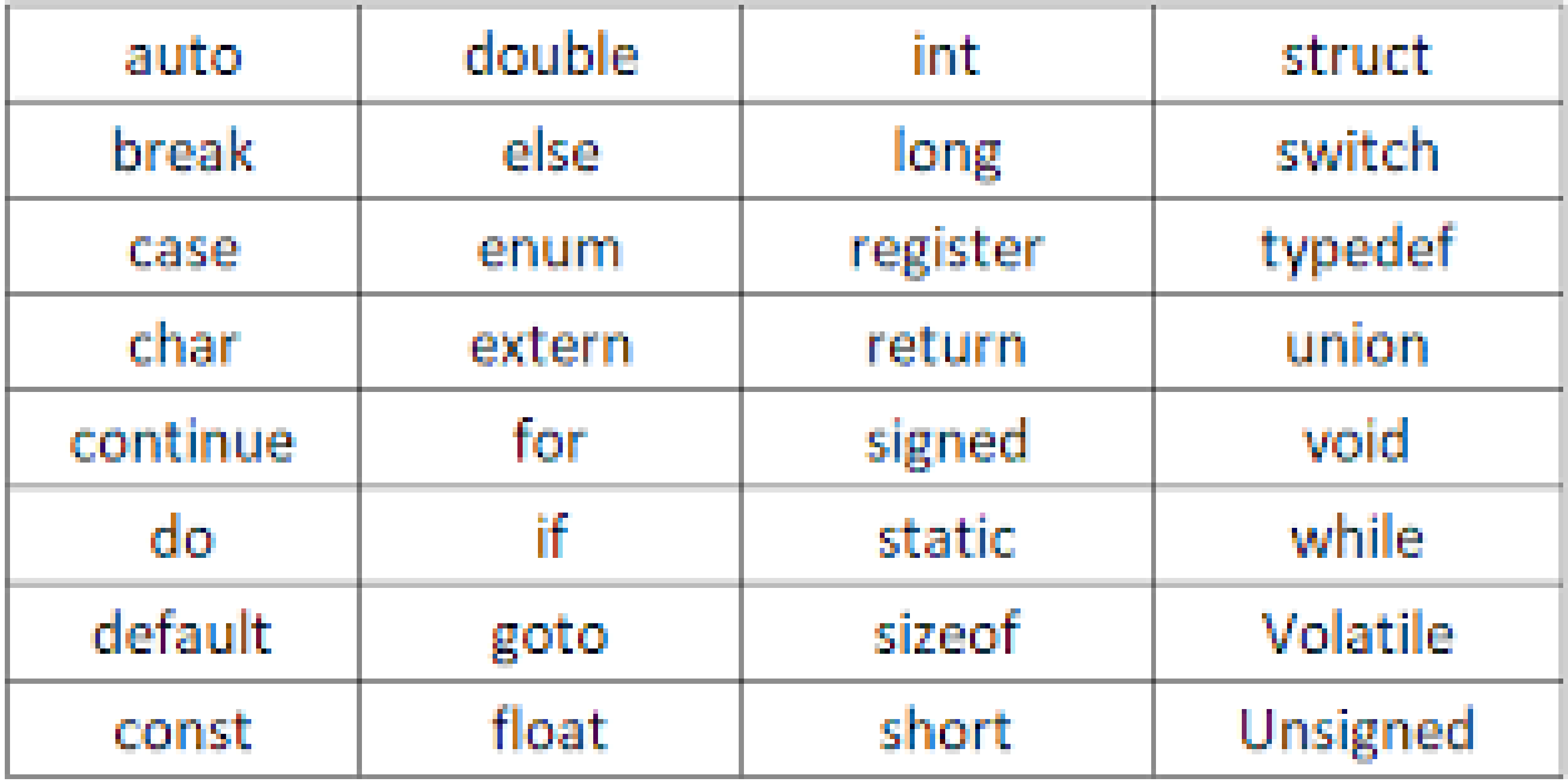
# C Programming Language

- C was created by Dennis Ritchie at the Bell Laboratories in 1972 as a system implementation language for the Unix operating system.
- It is a very popular language, despite being old. Is simple and flexible programming language that is widely used in various applications.

**Characteristics:**

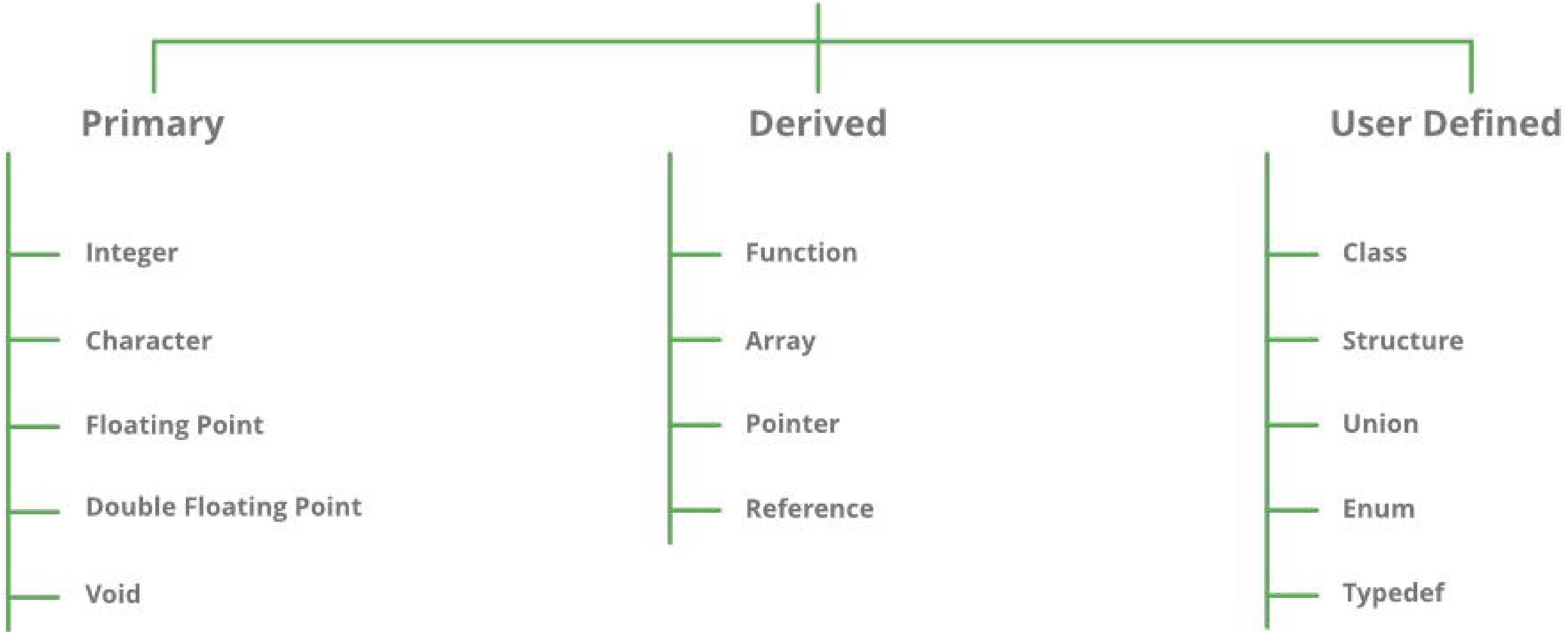
- Is a general-purpose language
- Is a free-formatted language
- Efficiency
- Portability
- Library Facilities

# C Keywords



| | | | |
|---|---|---|---|
| auto | double | int | struct |
| break | else | long | switch |
| case | enum | register | typedef |
| char | extern | return | union |
| continue | for | signed | void |
| do | if | static | while |
| default | goto | sizeof | Volatile |
| const | float | short | Unsigned |

# DataTypes in C



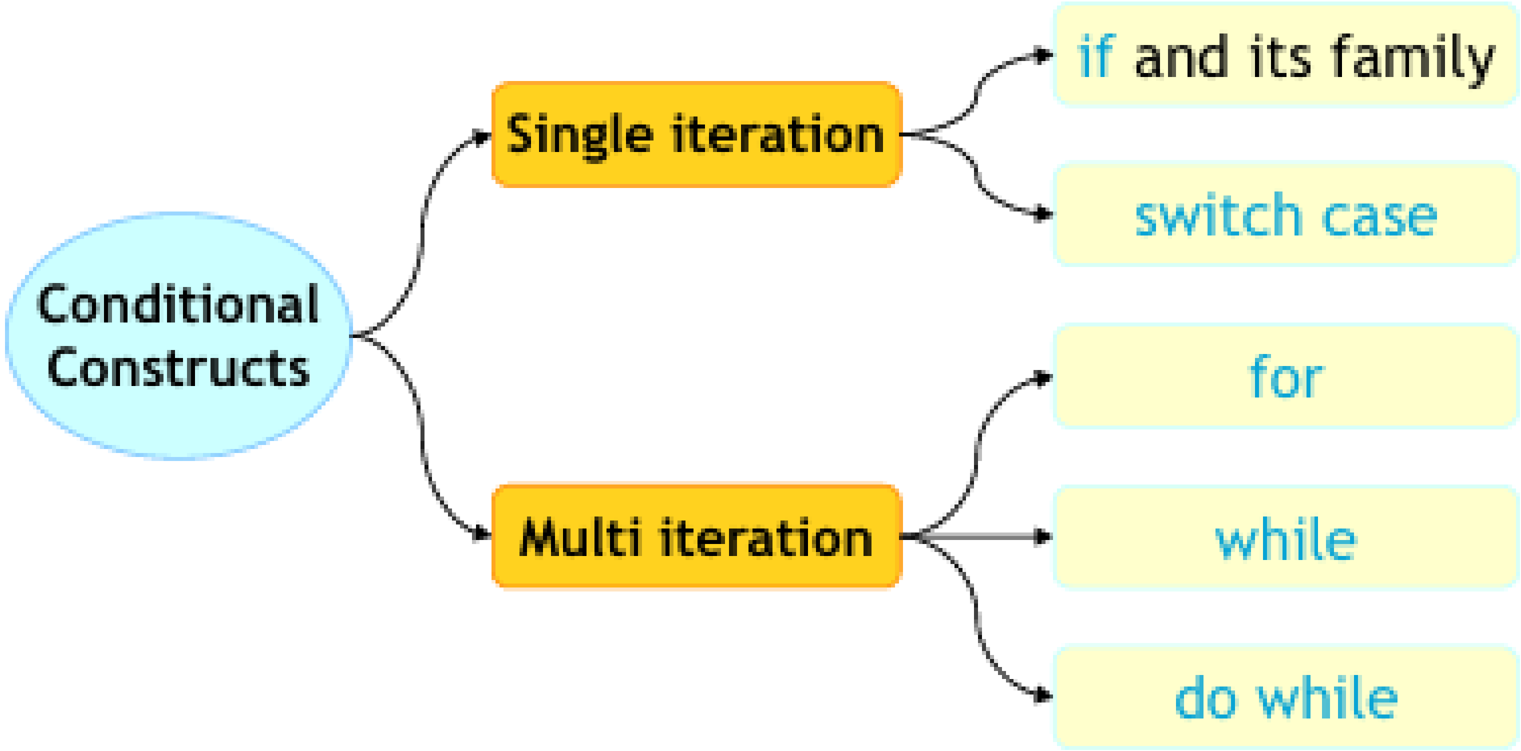
- Primary
  - Integer
  - Character
  - Floating Point
  - Double Floating Point
  - Void
- Derived
  - Function
  - Array
  - Pointer
  - Reference
- User Defined
  - Class
  - Structure
  - Union
  - Enum
  - Typedef

# Modifiers in C

## C has five types modifiers

- Short (stores 2 bytes of memory)
- Long (stores 4 bytes of memory)
- Long long (stores 8 or 16 bytes of memory)
- Signed (stores both negative and positive numbers)
- Unsigned (stores only positive numbers)

# Conditional Constructs



# Operators in C

| Operatos | Type of Operatos | Operation Type |
|---|---|---|
| ++, -- | Increments/Decrements Operators | Unary Operator |
| +, -, *, /,% | Arithmetic Operators | Binary Operator |
| <, <=, >, >=,==, != | Relational Operators | |
| &&, \|, ! | Logical Operators | |
| &, \|, <<, >>, ~, ^ | Bitwise Operators | |
| =, +=, -=, *=, /=, %= | Assignment Operators | |
| sizeof () , & * | Special Operaots | |
| ?: | Ternary or Conditional Operator | Ternary Operator |

# Arrays and Strings in C

**ARRAYS :**

- An array is a collection of data of same data type. First element will be with the lowest address and the last element with the highest address.
- Each element in the array is accessed by index

**STRINGS :**

- A String in C programming is a sequence of characters terminated with a null character '\0’.
- The difference between a character array and a C string is that the string in C is terminated with a unique character '\0'.

# Pointers and Functions in C

**POINTERS :**

- A pointer is a variable that stores the memory address of another variable.
- Instead of holding a direct value, it has the address where value is stored in memory.
- This allows us to manipulate the data stored at a specific memory location without actually using its variable.

**FUNCTIONS :**

- A function in C is a set of statements that, when called, perform some specific tasks.
- It is the basic building block of a C program that provides modularity and code reusability.
- We have 2 methods of parameters passing:

- Pass by value
- Pass by reference

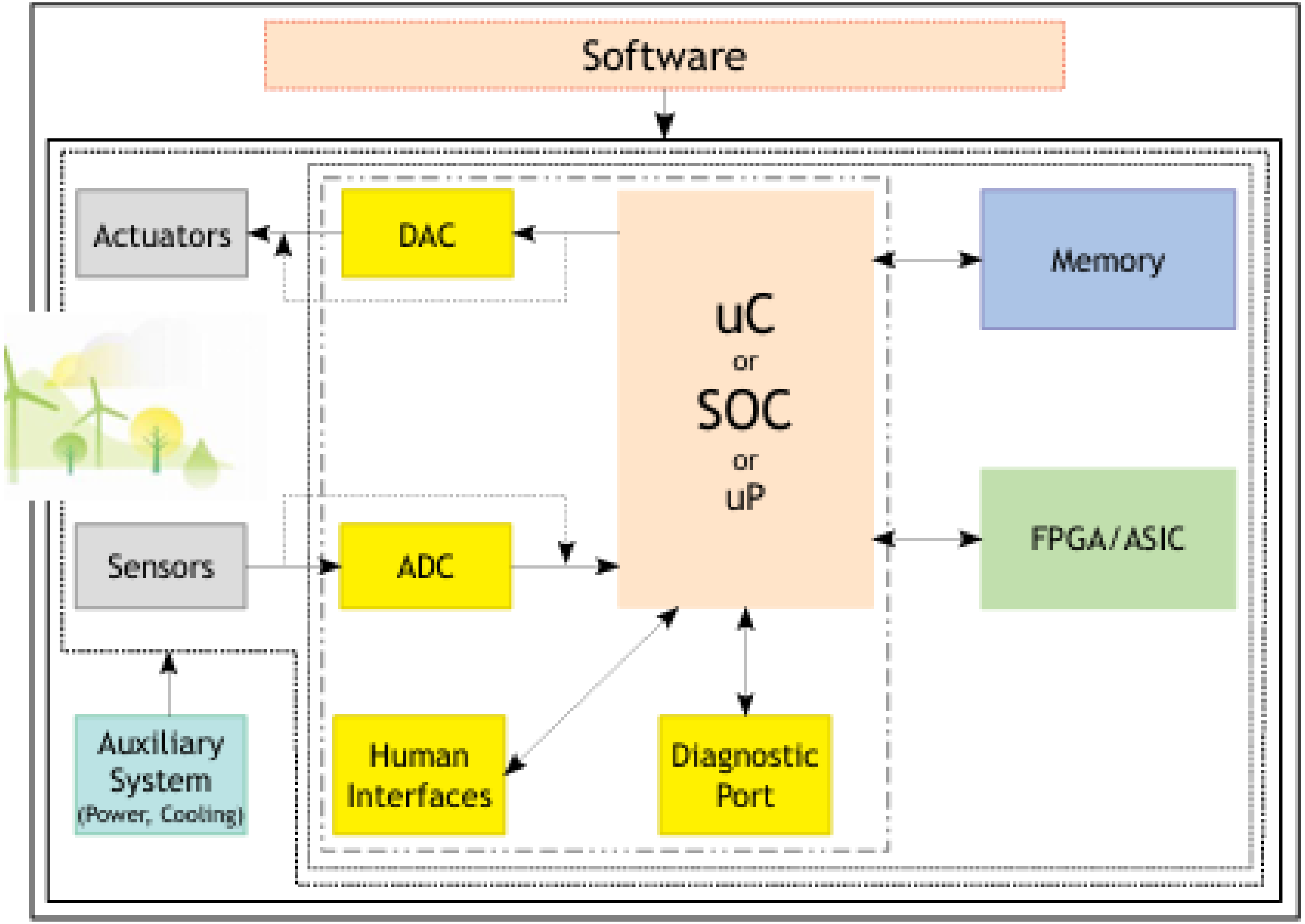
# Embedded Systems

- Any combination of Hardware and Software which is intended to do a Specific Task can be called as an Embedded System.
- Example: Washing Machine, Micro Oven, Fridges etc.

**Types of Embedded Systems**

- Stand-alone
- Real Time
- Networked
- Mobile

# Embedded System Components



# Embedded System

## REQUIREMENTS :

- Reliability
- Cost-effectiveness
- Low Power Consumption
- Efficient Usage of Processing Power
- Efficient Usage of Memory

## CHALLENGES :

- Efficient Inputs/Outputs
- Embedding an OS
- Code Optimization
- Testing and Debugging

# Memory Classification

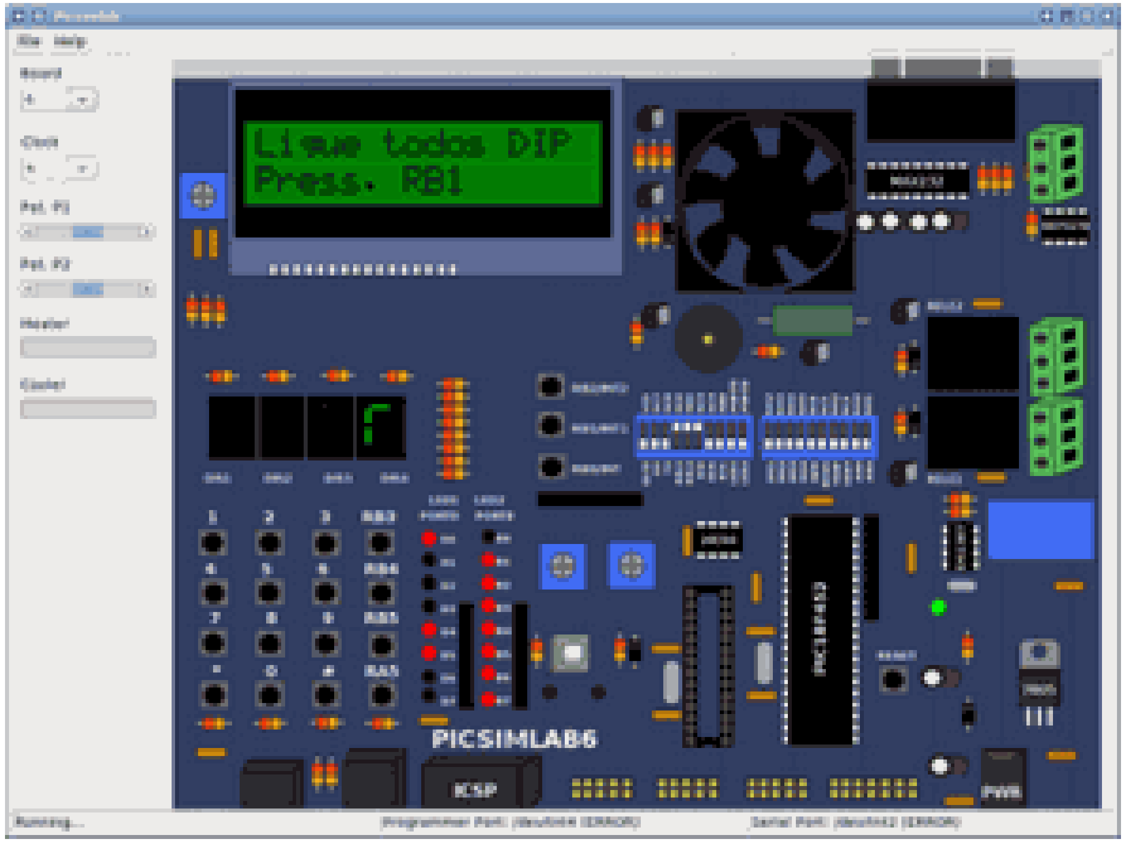
- ROM
  1. Programmable ROM (PROM)
  2. Erasable Programmable ROM (EPROM)
  3. Masked ROM
- RAM
  1. Dynamic RAM (DRAM)
  2. Static RAM (SRAM)
- Hybrid Memories
  1. Flash
     i. NOR Flash
     ii. NAND Flash
  2. EEPROM (Electrically Erasable Programmable Read-Only Memory

# Microcontroller

- Microcontroller is an integrated circuit which is capable of being programmed to perform a specific task.
- Unlike Microprocessor, a Microcontroller has all components integrated in a single chip.
- It is less flexible and require less design complexity.
- To build this project, we are using PIC Microcontroller.

# Specifications of PIC Microcontroller

- PIC Stand for Peripheral Interface Controller.
- Very easy to use Flash memory technology so that we can be write-erase until thousand times.
- For support serial communication it has 2 pins Tx and Rx.
- It can also support SPI Protocol.
- To build this project, we are using PIC16F877A Microcontroller.



# PIC16F877A

- It has 40 pins.
- It has 5 Ports in total.
  -PORT A [6 pins]
  -PORT B[8 pins]
  -PORT C[8 pins]
  -PORT D[8 pins]
  -PORT E [3 pins]).
- It supports Serial Communication for TX and RX.
- It can also supports SPI protocol.
- We have oscillator frequency range from 4MHZ-40MHZ.

# Software Used

- PICSIMLAB
- MPLAB-X IDE
- XC8 COMPILER
- PICGENIOUS-PIC16F877A MC BOARD

# Washing Machine

- This project focuses on simulating a washing machine using PICSimLab, aiming to replicate real-world washing machine functionality in a virtual environment.
- The project begins with a comprehensive understanding of washing machine components, cycles, and operations.
- Parameters defining laundry status, such as fabric type, load size, and water level, are identified.
- Peripherals Used :

➔CLCD

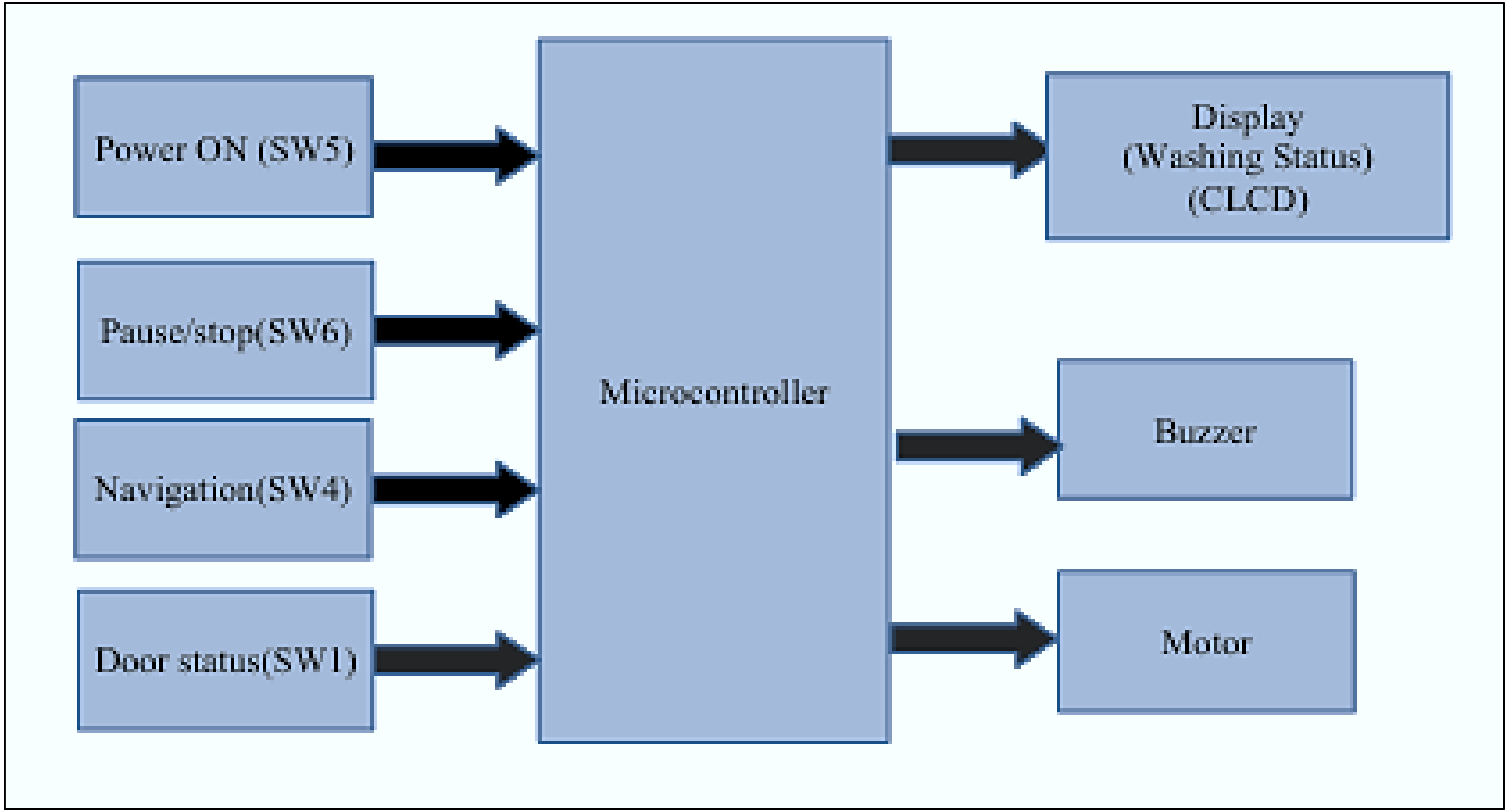
➔FAN

➔BUZZER

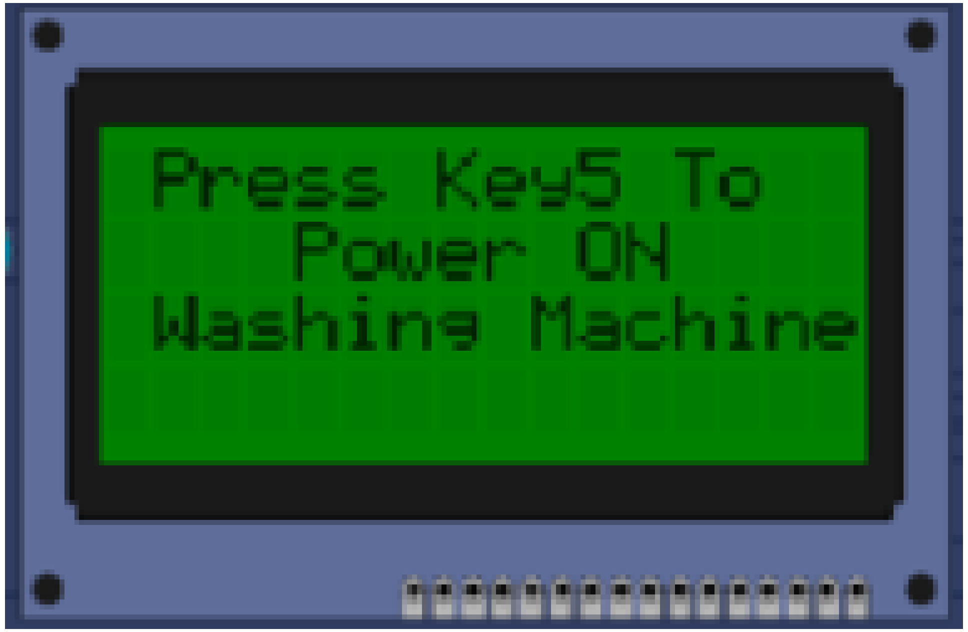
➔ SWITCHES

➔Timer

# Functional Requirements

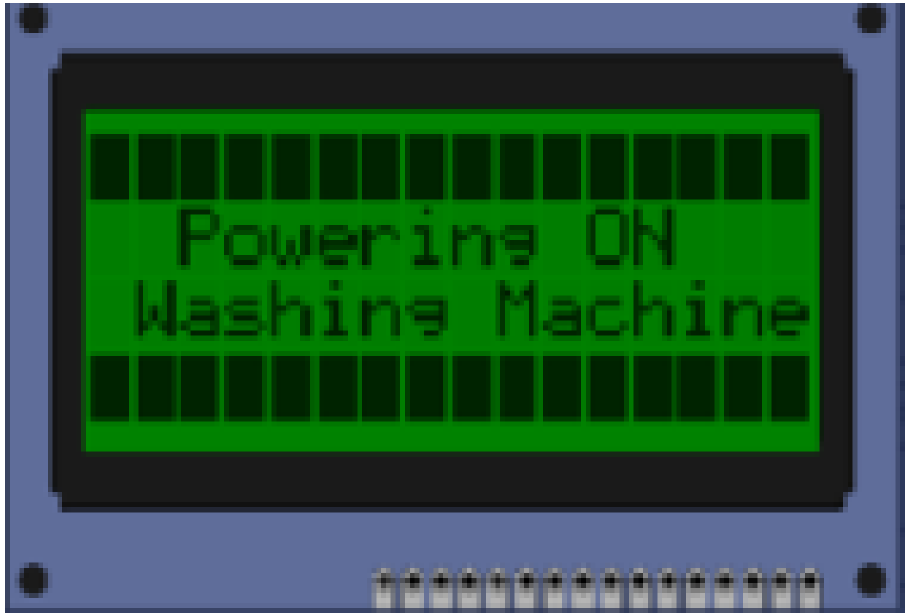


# POWER ON SCREEN



Press Key5 To
Power ON
Washing Machine

POWER ON SCREEN – Press SW5 to power ON Washing Machine



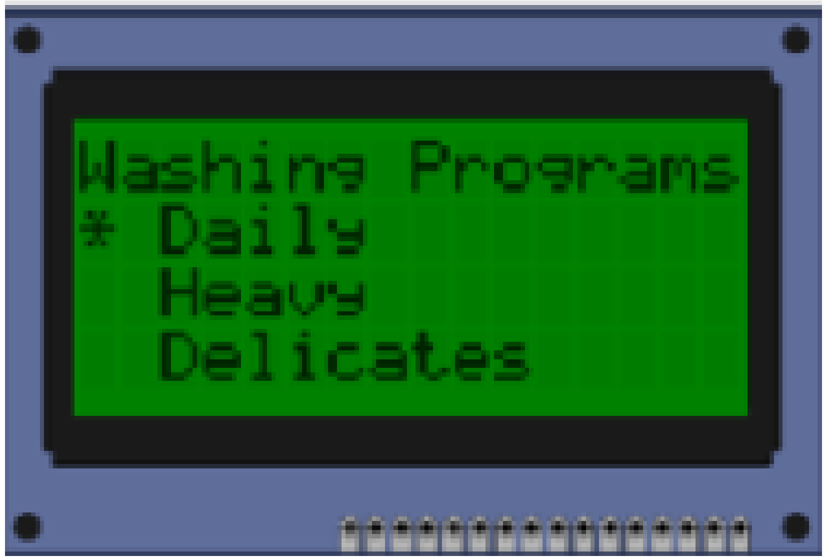
Powering ON
Washing Machine

POWERING ON SCREEN

# Washing Program Screen

**Washing Program Menu :**

1. Daily
2. Heavy
3. Dedicates
4. Whites
5. Stain wash
6. Eco cottons
7. Woolens
8. Bed sheets
9. Rinse + Dry
10. Dry only
11. Wash only
12. Aqua store



```
Washing Programs
* Daily
  Heavy
  Delicates
```

Press SW4 to scroll menu options

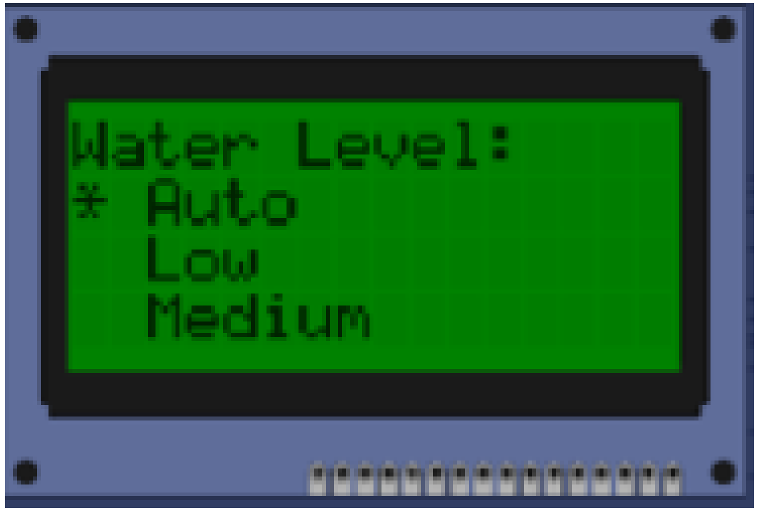
Long Press SW4 to select

# Water Level Screen

**Water Level Menu :**

1. Auto
2. Low
3. Medium
4. High
5. Max

- The user need to select water level



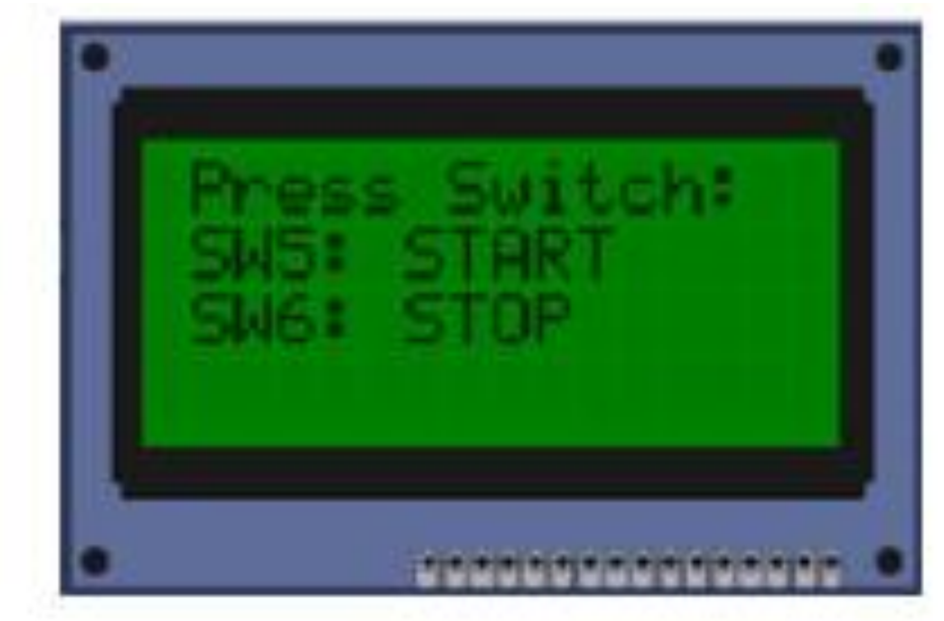
```
Water Level:
* Auto
  Low
  Medium
```

Press SW4 to scroll menu options
Long Press SW4 to select

# Start Screen

**Start or Stop Screen Display :**

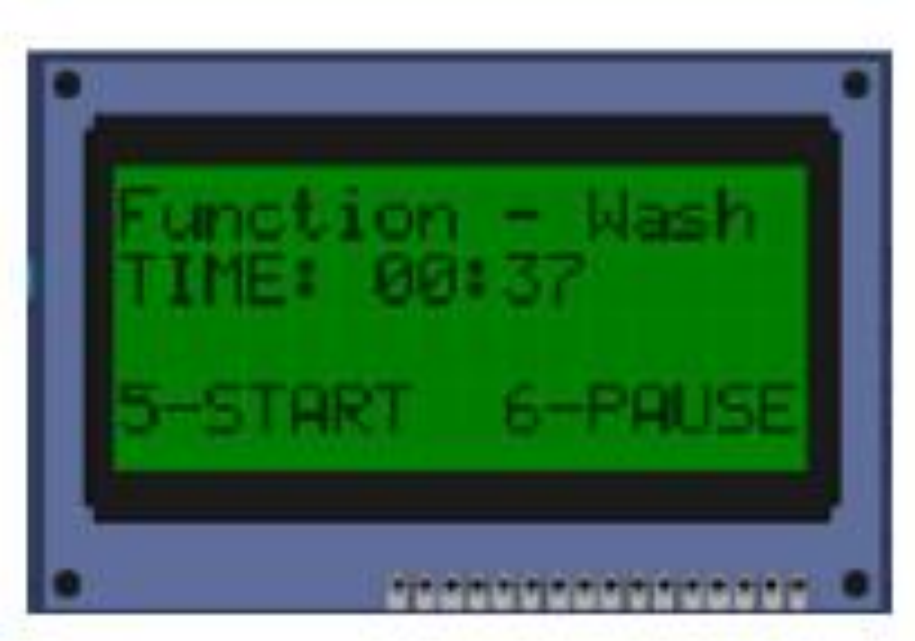
- User have options to start or stop the program.
- If SW5 is pressed, then washing run for selected menu.
- If SW6 is pressed it goes back to Washing menu screen.



```
Press Switch:
SW5: START
SW6: STOP
```

# Function Screen

- Selected menu starts executing.
- Timer shows how many mins/secs washing Will be done.
- User can Pause and Start the program by pressing SW6 and SW5 respectively.
- FAN is turned ON during the program

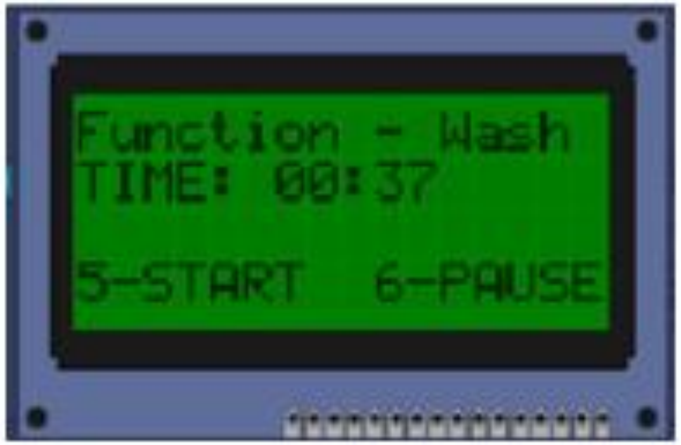


```
Function - Wash
TIME: 00:37

5-START   6-PAUSE
```

# Function Screen

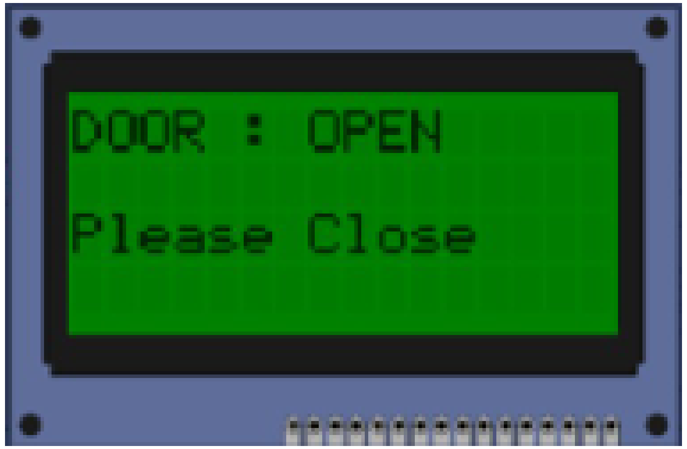
- Selected menu starts executing.
- Timer shows how many mins/secs washing Will be done.
- User can Pause and Start the program by pressing SW6 and SW5 respectively.
- FAN is turned ON during the program



```
Function - Wash
TIME: 00:37

5-START   6-PAUSE
```

- SW1 acts as door for washing machine.
- If SW1 Is pressed BUZZER turns ON and FAN turn OFF

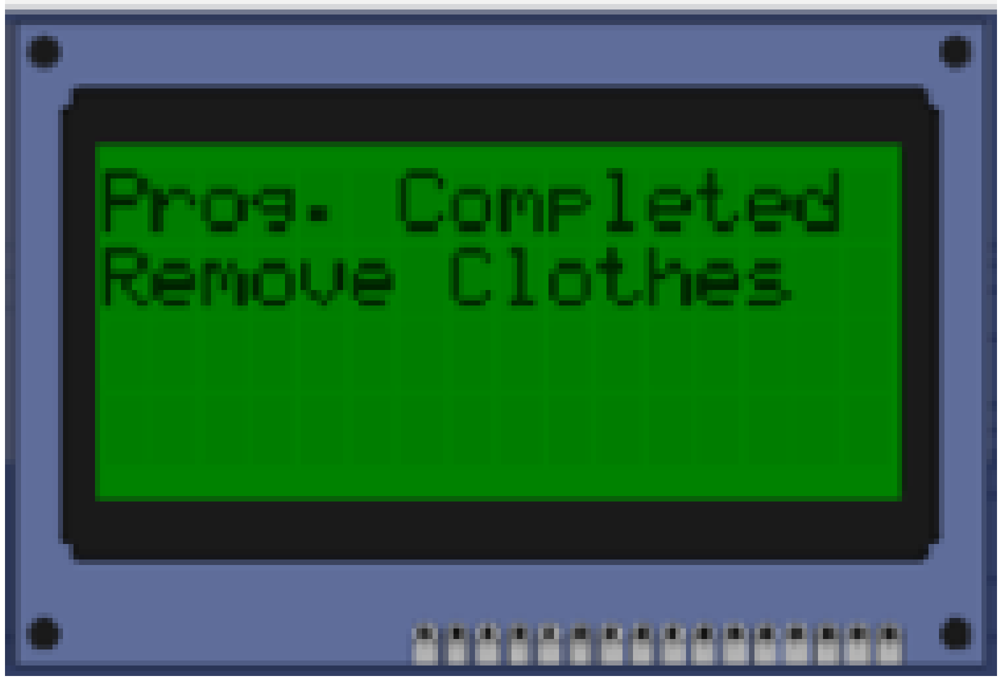


```
DOOR : OPEN

Please Close
```

# Completion Status Screen

- If program time is completed, then, PROGRAM COMPLETION SCREEN is displayed, and BUZZER turns ON.



```
Prog. Completed
Remove Clothes
```

# Washing Machine Simulation Demo

# Learning Outcomes

- Understood C for embedded use
- Applied timer interrupts
- FSM design principles
- Peripheral control using GPIO
- Debugging with PicsimLab
- Importance of documentation

# THANK YOU !