Calculator

EE24BTECH11017 - Karthik

I. INTRODUCTION

This report presents the design and implementation of a scientific calculator using an Arduino Uno, a JHD162A LCD display, 16 input buttons, and other supporting components. The calculator is programmed using AVR-GCC on Termux (Debian) and supports various mathematical functions within the constraints of 16 push buttons. The report covers the circuit design, software implementation, and testing results.

II. HARDWARE COMPONENTS

The main components used in this project are listed in Table

Component	Quantity
Arduino Uno	1
16x2 LCD Display	1
Push Buttons	16
Resistors (220Ω, 10kΩ)	Assorted
Breadboard	1
Jumper Wires	As required
Power Source (USB or Battery)	1

TABLE 0

LIST OF HARDWARE COMPONENTS

III. CIRCUIT CONNECTIONS AND KEYPAD INTERFACE

To optimize space, the circuit connections and keypad functions are presented in Tables below respectively.

Component	Arduino Pin
LCD RS	D7
LCD E	D6
LCD D4	D5
LCD D5	D4
LCD D6	D3
LCD D7	D2
Keypad Rows	A0-A3
Keypad Columns	D8-D11
Power (VCC)	5V
Ground (GND)	GND

TABLE 0

CIRCUIT CONNECTIONS

Button	Primary / Alternate Mode Function
0-9	Numeric Input
+	Addition
-	Subtraction
×	Multiplication
÷	Division
=	Compute Result
C	Clear Input
.	Decimal Point
M+	Store in Memory
MR	Recall Memory
MC	Clear Memory
SIN	$\sin x$ / $\sin^{-1} x$
COS	$\cos x$ / $\cos^{-1} x$
TAN	$\tan x$ / $\tan^{-1} x$
EXP	e^x / $\ln x$
LOG	$\log_{10} x$ / $\log_2 x$
MODE	Toggle Between Standard and Scientific Mode

TABLE 0

KEYPAD BUTTON FUNCTIONS

IV. SOFTWARE IMPLEMENTATION

The firmware is written in Embedded C and compiled using AVR-GCC. The core functionalities include:

- Reading ADC values for digit buttons.
- Detecting arithmetic operations through digital pins.
- Implementing additional functions such as memory storage and recall.
- Displaying input and results dynamically on the LCD.
- Handling error conditions such as division by zero.

V. FEATURES

- **Basic Arithmetic**: Addition, subtraction, multiplication, and division.
- **Scientific Functions**: Trigonometry, logarithms, exponentiation.
- **Memory Functions**: Store and recall values.
- **Number Systems**: Convert between HEX, DEC, BIN, and OCT.
- **Multi-Mode Input**: Normal, Alpha, and Shift functions.
- **Parentheses Handling**: Supports complex expressions.

VI. CONCLUSION

This project successfully implements a digital calculator using an Arduino microcontroller. By leveraging ADC values and optimizing pin usage, the design maintains hardware simplicity while delivering comprehensive functionality. The use of AVR-GCC enhances efficiency, and future improvements could include higher precision and additional functions.