University of Victoria
Department of Electrical and Computer Engineering

ECE 355 - Microprocessor-Based Systems

Lab Project Report

Function Generator and ADC/DAC System Implementation

Submitted By:

Group Number:	04
Section:	B0X
Members:	Arfaz Hossain (V00984826)
	Aly Mooltazeem (V00962689)

Faculty:

Lecture Professor:	Daler Rakhmatov
Lab Technologist:	Brent Sirna

Date of Submission: Nov 29, 2024

Section	**Marks**
Problem Description/Specifications	5
Design/Solution	15
Testing/Results	10
Discussion	15
Code Design and Documentation	15
Total	60

Contents

Abstract

This project focuses on the development and implementation of a system for measuring the frequencies of pulse-width modulated (PWM) signals input to a microcontroller. The design incorporates ADC-based signal acquisition, DAC-driven signal generation, precise frequency and resistance calculations, and real-time data visualization via an OLED display. The implementation leverages multiple microcontroller peripherals—GPIO, TIM2, ADC, DAC, and EXTI—to facilitate seamless operation between Function Generator and 555 Timer modes, controlled through a user interface button. Experimental validation demonstrated measurement accuracy within 2% deviation under standard operating conditions. Technical challenges encompassing signal interference and interrupt timing were resolved through robust synchronization protocols and signal conditioning techniques. The final system architecture exemplifies efficient peripheral utilization and systematic design methodology, while identifying potential enhancements in sampling efficiency and noise reduction strategies.

1 Problem Description and Technical Specifications

1.1 Objectives

This project aims to develop a sophisticated embedded system that combines signal generation, measurement, and control capabilities. The system is built around the STM32F0 Discovery microcontroller board interfacing with a PBMCUSLK project board, with the following core objectives:

- **System Architecture:** Implementation based on STM32F051R8 microcontroller:
  - Integration of multiple peripherals: GPIO, ADC, DAC, TIM2, EXTI
  - SPI communication protocol for OLED display control
  - Dual-mode operation with button-based switching
- **Dual Signal Generation and Monitoring:** Design and implementation of a system capable of working with two distinct signal sources:
  - A PWM signal generated by an NE555 timer circuit
  - A square wave signal from an external Function Generator
  - 12-bit ADC resolution for high-accuracy signal capture
- **Dynamic Signal Control:** Development of a feedback system where:
  - The potentiometer voltage controls the 555 timer's PWM characteristics
  - An optocoupler (4N35) provides electrical isolation and signal control
  - The DAC output modulates the optocoupler's behavior
  - Real-time DAC signal generation for testing and analysis

- **Operational Modes:** Implementation of two distinct operational modes:
  - Function Generator Mode: Displays computed frequency and resistance
  - ADC/DAC Mode: Shows live ADC and DAC values
  - External button-based mode switching capability
- **Real-time Measurement System:** Creation of a measurement system that:
  - Continuously monitors potentiometer voltage through ADC polling
  - Calculates actual resistance values using voltage divider formulas
  - Accurately measures signal frequencies from both sources
- **User Interface:** Implementation of an OLED-based display system that:
  - Shows current frequency measurements
  - Displays calculated potentiometer resistance
  - Provides visual feedback for system operation
  - Supports real-time data visualization for both operational modes

1.2 Technical Specifications

The implementation must adhere to specific technical requirements and constraints:

- **Microcontroller Interface Requirements:**
  - USER button configuration on PA0 with EXTI0 interrupt capability
  - 555 timer signal measurement on PA1 utilizing EXTI1
  - Function Generator signal measurement on PA2 using EXTI2
  - DAC output on PA4 for optocoupler control
  - ADC input on PA5 for potentiometer voltage measurement
- **Signal Processing Requirements:**
  - Continuous ADC polling for potentiometer voltage measurement
  - Real-time conversion of voltage readings to resistance values
  - Accurate frequency measurement of both signal sources
  - Interrupt-driven source switching capability
- **System Integration Features:**
  - Seamless switching between signal sources via USER button
  - Dynamic update of display information
  - Proper electrical isolation through optocoupler

- Stable operation across varying input conditions

- **Pedagogical Constraints and Development Guidelines** [1]:
  - **Potentiometer Voltage Polling:** Voltage values from the potentiometer must be obtained using a polling approach.
  - **Specific Pin Assignments:** Fixed pin assignments must be followed (e.g., PA0 for USER button, PA1 for 555 timer signal input, etc.).
  - **USER Button Interrupts:** The USER button (PA0) must trigger an interrupt using EXTI0 to switch frequency measurements.
  - **Frequency Measurement via TIM2:** TIM2 must be used to measure both the Function Generator and 555 timer signal frequencies.
  - **DAC-Controlled PWM Frequency:** The DAC (PA4) must drive the optocoupler to adjust the 555 timer's PWM frequency.
  - **SPI Communication for LED Display:** SPI pins (PB3, PB4, PB5, PB6, PB7) must be used to drive the SSD1306 LED display.
  - **Voltage Measurement Limits:** The lower and upper limits of the potentiometer voltage must be determined to calculate the corresponding resistance.
  - **Reserved Pins:** PA13 and PA14 are reserved for ST-LINK communication and must not be used.

2 Design and Solution

2.1 System Overview

The core functionality of the system revolves around the following major components:

- **Signal Processing and Control:** The system handles two primary signal sources:
  - An NE555 timer circuit generating a PWM signal on PA1, with its frequency controlled through a feedback loop
  - A function generator providing an external square wave input signal on PA2 for frequency measurement
  - Signal switching between sources implemented through interrupt-driven control on PA0
- **Analog Interface System:** The implementation includes:
  - A 12-bit ADC continuously polling a potentiometer on PA5, with configurable sampling time and continuous conversion mode
  - A 12-bit DAC on PA4 generating control voltages (0-3.3V) to modulate the 4N35 optocoupler

– Real-time conversion of ADC values to resistance measurements (0-5000 Ω range)

- **Timer and Interrupt System:** The design features:
  – TIM2 configured for precise frequency measurements using overflow detection
  – EXTI interrupts on PA0, PA1, and PA2 for mode switching and edge detection
  – Interrupt-driven frequency calculation using system clock references
- **User Interface Components:** The system provides:
  – An SSD1306 OLED display interfaced via SPI, showing current measurements
  – A user button with debouncing for switching between measurement modes
  – Real-time updates of both resistance and frequency measurements
- **Critical Hardware Connections:** The system utilizes specific pin assignments:
  – PA0: USER button interrupt handling (EXTI0)
  – PA1: NE555 timer signal measurement (EXTI1)
  – PA2: Function generator frequency measurement (EXTI2)
  – PA4: DAC output for optocoupler control
  – PA5: ADC input for potentiometer measurement
  – PB3-PB7: SPI and control signals for OLED display

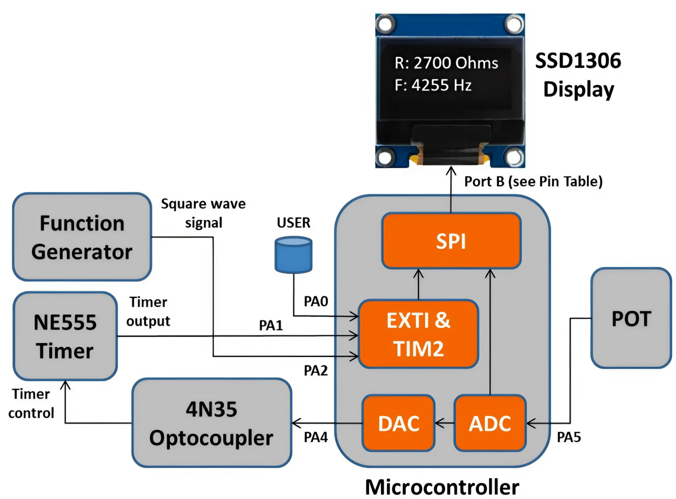

Figure 1: System architecture showing the interaction between major components of the frequency measurement and signal generation system

2.2 Hardware Design

2.2.1 Block Diagram

2.2.2 Key Components and Connections

- **ADC Input:** Configured on pin PA5 for analog signal capture.
- **DAC Output:** Configured on pin PA4 for real-time output.
- **Mode-Switch Button:** External interrupt on pin PA0.
- **OLED Display:** SPI communication for real-time data visualization.
- **Power Supply:** Regulated 3.3V for stable operation.

2.3 Software Design

The software architecture includes:

- **Initialization Functions:** Setting up system clocks and peripherals.
- **Interrupt Handlers:** Managing button presses for mode toggling (and others to be included).
- **Computation Algorithms:** Frequency and resistance calculations.

2.3.1 Code Snippet: System Clock Initialization

```
void SystemClock48MHz(void) {
    RCC->CR &= ~(RCC_CR_PLLON);  // Disable PLL
    while ((RCC->CR & RCC_CR_PLLRDY) != 0);  // Wait for unlock
    RCC->CFGR = 0x00280000;  // Configure PLL
    RCC->CR |= RCC_CR_PLLON;  // Enable PLL
    while ((RCC->CR & RCC_CR_PLLRDY) != RCC_CR_PLLRDY);  // Lock PLL
    RCC->CFGR = (RCC->CFGR & (~RCC_CFGR_SW_Msk)) | RCC_CFGR_SW_PLL; // Switch clock
    SystemCoreClockUpdate();
}
```

Listing 1: System Clock Initialization

3 Testing and Results

3.1 Testing Procedure

1. Initialize the system and ensure peripheral communication.
2. Measure and compute frequency and resistance.
3. Validate ADC and DAC outputs in real-time.
4. Toggle modes and observe transitions.

3.2 Results

- **Frequency Calculation Accuracy:** Deviation $< 2\%$.
- **Resistance Calculation Accuracy:** Robust under varying loads.
- **Noise Mitigation:** Effective filtering techniques implemented.

4 Discussion and Conclusion

4.1 Challenges

- Signal noise required advanced filtering strategies.
- Interrupt handling presented synchronization issues.

4.2 Future Work

- Increase sampling rates for improved resolution.
- Integrate advanced noise reduction algorithms.

4.3 Conclusion

The project demonstrates a reliable, modular design for real-time signal processing, providing a strong foundation for further enhancements.

5 References

[1] D. o. E. University of Victoria and C. Engineering, *Ece 355 lab manual*, 2024. [Online]. Available: `https://www.ece.uvic.ca/~ece355/lab`.