

Obsługa Akcelerometru z Przerwaniami

Dodatkowo wykorzystanie LCD 16x2 oraz wbudowanego pola dotykowego

Projekt wykonali: Jakub Płoskonka, Grzegorz Krawczyk

Prowadzący: dr inż. Sebastian Koryciak

Data wykonania: 15.01.2025r.

Spis treści

Cel projektu.....	3
Wymagania techniczne	3
Opis połączeń.....	3
Obsługa przerwań	4
Kod programu	4
Testy i weryfikacja	5
Podsumowanie	6

Cel projektu

Projekt polega na stworzeniu oprogramowania do obsługi akcelerometru MMA8451Q na mikrokontrolerze KL05Z. Komunikacja z akcelerometrem odbywa się przez magistralę I2C, a przerwanie INT2 podłączone jest do pinu PTA10. Wyniki pomiarów osi X, Y, Z są wyświetlane na wyświetlaczu LCD 2x16.

Wymagania techniczne

- Mikrokontroler: MKL05Z32VFM4 z rdzeniem ARM Cortex-M0+ (48 MHz).
- Akcelerometr: MMA8451Q z obsługą przerwań.
- Magistrala komunikacyjna: I2C.
- Wyświetlacz: LCD 2x16 z interfejsem zgodnym z HD44780.
- Pin przerwania: PTA10 (INT2).

Opis połączeń

1. Akcelerometr

- SCL: PTB3
- SDA: PTB4
- INT2: PTA10

2. Wyświetlacz LCD

- SLC: PTB3
- SDA: PTB4
- VCC: +5V
- GND: GND mikrokontrolera

3. Pole dotykowe (pojemnościowe)

- Pierwsza elektroda: PTB12
- Druga elektroda: PTB13

Obsługa przerwań

W momencie wystąpienia przerwania INT2 na pinie PTA10 zostaje wywołana funkcja obsługi. Funkcja ta odczytuje wartości osi X, Y, Z z akcelerometru, formatuje dane i wyświetla je na LCD razem z graficzną wizualizacją aktualnego przyspieszenia. Dodatkowo umożliwia zmianę aktualnie wyświetlanej osi.

Kod programu

Struktura projektu składa się z następujących plików:

- main.c. Główna funkcja programu, która inicjalizuje działanie mikrokontrolera i obsługuje pętlę główną.
- accelerometer.c Kod odpowiedzialny za komunikację z akcelerometrem i przetwarzanie danych z jego czujników.
- accelerometer.h Nagłówek definiujący funkcje i struktury używane do obsługi akcelerometru.

Nazwa funkcji	Krótki opis
Accelerometer_WriteRegister(uint8_t reg, uint8_t value)	<ul style="list-style-type: none">• Wysyła wartość do wybranego rejestru akcelerometru.
Accelerometer_ReadRegister(uint8_t reg)	<ul style="list-style-type: none">• Odczytuje wartość z określonego rejestru akcelerometru.
Accelerometer_Init()	<ul style="list-style-type: none">• Konfiguruje akcelerometr, ustawiając odpowiednie parametry pracy, takie jak tryb działania i czułość.

- dispData.c Odpowiadają za przetwarzanie danych wyświetlanych na ekranie.
- dispData.h Nagłówek definiujący funkcje i struktury używane do wyświetlania danych na ekranie

Nazwa funkcji	Krótki opis
display_progress_bar(int16_t value)	<ul style="list-style-type: none"> Wyświetla dynamiczny pasek postępu na LCD, wizualizujący wartość jako wypełnienie paska. Pasek rozróżnia wartości dodatnie (wypełnienie w prawo) i ujemne (wypełnienie w lewo).
display_acceleration_data(int16_t x, int16_t y, int16_t z)	<ul style="list-style-type: none"> Wyświetla aktualne wartości przyspieszenia na osi X, Y i Z.
clear_display()	<ul style="list-style-type: none"> Czyści wyświetlacz LCD, przygotowując go do nowych danych.

- lcd1602.c Kod obsługujący wyświetlacz 2x16 znaków (LCD 1602).
- lcd1602.h Nagłówek definiujący funkcje i zmienne związane z wyświetlaczem.
- i2c.c Kod implementujący komunikację z wykorzystaniem magistrali I2C.
- i2c.h Nagłówek dla modułu I2C
- tsi.c Obsługa interfejsu dotykowego TSI.
- tsi.h Nagłówek definiujący funkcje i struktury używane do obsługi interfejsu dotykowego TSI
- frdm_bsp.h Nagłówek zawierający podstawowe definicje sprzętowe dla płytki FRDM-KL05Z

Testy i weryfikacja

1. Sprawdzenie komunikacji z akcelerometrem przez magistralę I2C.
2. Test poprawności obsługi przerwań.
3. Weryfikacja wyświetlania danych na LCD.

Podsumowanie

Projekt realizuje funkcjonalność odczytu danych z akcelerometru i ich wyświetlania w czasie rzeczywistym. Użycie przerwań umożliwia efektywną obsługę zdarzeń bez nadmiernego obciążania procesora. Dodatkowo użytkownik ma możliwość manipulacji wyświetlanymi danymi, w postaci zmiany wyświetlanych osi, które dodatkowo w dolnym rzędzie LCD obrazują aktualne przyspieszenie skalując je w granicach 0 do 1g.