**Raport do zadania 3**

Imię i nazwisko: Jakub Dmochowski

Numer Albumu: 169236

**Konfiguracja sprzętowa**

**A table with black text

AI-generated content may be incorrect.**

POSCMOD = XT - wybór zewnętrznego kwarcu jako źródła zegara

OSCIOFNC = ON - funkcja wyjścia oscylatora włączona

FCKSM = CSDCMD - wyłączenie monitorowania zegara i przełączania

FNOSC = PRI - Podstawowe źródło zegara jako domyślne

IESO = ON - włączenie wewnętrznego/zewnętrznego przełączania oscylatora

**Watchdog timer, debugowanie**

**A table of text with different colored letters

AI-generated content may be incorrect.**

WDTPS = PS32768 - Prescaler watchdog timera ustawiony na 1:32768

FWPSA = PR128 - Prescaler A watchdog timera ustawiony na 1:128

WINDIS = ON - Windowed watchdog timer wyłączony

FWDTEN = OFF - Watchdog timer całkowicie wyłączony

ICS = PGx2 - Komunikacja z debuggerem przez piny PGC2/PGD2

GWRP = OFF - Ochrona zapisu do pamięci programu wyłączona

GCP = OFF - Ochrona odczytu kodu wyłączona

JTAGEN = OFF - Interfejs JTAG wyłączony

**Biblioteki**

**A group of yellow text

AI-generated content may be incorrect.**

#include <xc.h> - Główna biblioteka dla kompilatorów XC

#include <libpic30.h> - Biblioteka specyficzna dla PIC30, zawiera funkcje opóźnień

#include <stdbool.h> - Biblioteka standardowa C dla typu bool (true/false

**Definicje i stałe**

**A close-up of a computer code

AI-generated content may be incorrect.**

#define FCY 4000000UL - Częstotliwość procesora 4 MHz

#define DEBOUNCE\_DELAY 200000 - Opóźnienie dla eliminacji drgań styków

#define button\_wylacz PORTDbits.RD6 -Przycisk wyłączenia alarmu

#define NASTAWA\_ALARM 512 - Próg załączenia alarmu (połowa zakresu ADC)

#define OPOZNIENIE\_MRUGANIA 500000 - Opóźnienie dla mrugania LED w fazie 1

**Konfiguracja wejść/wyjść i ADC**

**A white background with black text

AI-generated content may be incorrect.**

TRISA = 0x0000; - Port A jako wyjście (sterowanie LED)

TRISD = 0xFFFF; - Port D jako wejście (odczyt przycisków)

AD1PCFG = 0xFFDF; - Pin AN5 jako analogowy, pozostałe cyfrowe

AD1CON1 = 0x00E0; - Format danych, tryb próbkowania

AD1CON2 = 0; - Referencja napięcia i tryb skanowania

AD1CON3 = 0x1F3F; - Czas konwersji i próbkowania

AD1CHS = 5 - Wybór kanału AN5 dla konwersji

AD1CON1bits.ADON = 1 - Włączenie przetwornika ADC

\_\_delay32(1000);  *- STABILIZACJA ADC*

**potencjometr() - Odczyt wartości z przetwornika ADC**

**A screen shot of a computer code

AI-generated content may be incorrect.**

Definicje:

* AD1CON1bits.SAMP = 1; - Rozpoczęcie próbkowania
* \_\_delay32(100); - Czas stabilizacji próbkowania
* AD1CON1bits.SAMP = 0; - Zakończenie próbkowania, start konwersji
* while (!AD1CON1bits.DONE); - Oczekiwanie na zakończenie konwersji
* return ADC1BUF0; - Zwrócenie wyniku konwersji

Zasada działania:

* Rozpoczyna próbkowanie sygnału analogowego z potencjometru
* Czeka 100 cykli na stabilizację
* Kończy próbkowanie i rozpoczyna konwersję ADC
* Oczekuje na zakończenie konwersji
* Zwraca 10-bitową wartość z bufora

**Algorytm systemu alarmowego**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**Definicje**

wartosc\_pot = potencjometr() - Odczyt aktualnej wartości ADC

Przejście ze stanu normalnego do ostrzegawczego

* stan\_alarmu = 1; - Aktywacja fazy ostrzegawczej
* licznik\_5s = 0; - Reset licznika czasu
* dioda\_stan = 0; - Reset stanu mrugania

Dezaktywacja alarmu przez potencjometr

* stan\_alarmu = 0; - Powrót do stanu normalnego
* LATA = 0x0000; - Wyłączenie wszystkich LED
* licznik\_5s = 0; - Reset licznika

**Wyłączenie przyciskiem z debouncing**

**A computer code with text

AI-generated content may be incorrect.**

Definicje:

* \_\_delay32(DEBOUNCE\_DELAY); - Pierwsze opóźnienie 200000 cykli
* if (!button\_wylacz) - Potwierdzenie stanu przycisku
* stan\_alarmu = 0; - Reset automatu
* LATA = 0x0000; - Wyłączenie LED
* licznik\_5s = 0; - Reset licznika
* while (!button\_wylacz); - Oczekiwanie na zwolnienie
* \_\_delay32(DEBOUNCE\_DELAY); - Drugie opóźnienie

Zasada działania:

* Dwukrotna weryfikacja stanu przycisku z opóźnieniem
* Eliminacja fałszywych wyzwoleń spowodowanych drganiami styków
* Blokowanie do momentu zwolnienia przycisku
* Działa tylko gdy system jest w stanie alarmowym (stan\_alarmu > 0)

**Stan alarmu**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**Stan 0 - Tryb normalny:**

* LATA = 0x0000 - Wszystkie LED na porcie A wyłączone
* System oczekuje na przekroczenie progu potencjometru

**Stan 1 - Mruganie:**

* Algorytm toggle: Zmienna dioda\_stan kontroluje stan LED
* dioda\_stan == 0 - Włącz LED, dioda\_stan = 1
* dioda\_stan == 1 - Wyłącz LED, dioda\_stan = 0
* OPOZNIENIE\_MRUGANIA = 500000 cykli
* Częstotliwość mrugania: około 4 Hz
* licznik\_5s++ po każdym cyklu mrugania
* Automatyczne przejście po 5 sekundach

**Stan 2 - Alarm:**

* LATA = 0x00FF - Wszystkie 8 LED świecą jednocześnie (bity 0-7)
* \_\_delay32(100000) - Krótkie opóźnienie dla odciążenia procesora
* Stan utrzymuje się do wyłączenia