

Wydział Informatyki, Laboratorium Architektury Komputerowej	Data 07.12.2023r.
Ćwiczenie nr.4 Temat: Wprowadzenie do wyświetlacza LCD Część 2: LCD i ADC Grupa LAB.02 Imię i Nazwisko: Dominik Gąsowski	Prowadzący: Dr inż. Mirosław Omieljanowicz Ocena:

Zadanie 1.

Treść zadania:

- Korzystając ze schematu płytki LCD Keypad Shield, odszukaj na schemacie drabinę rezystorową (tj. wielokrotny dzielnik napięciowy) składającą się z 5 rezystorów oraz 4 przycisków.
- Korzystając ze schematu płytki Nucleo, zidentyfikuj wyprowadzenie płytki Nucleo, za pomocą którego można zmierzyć napięcie wyjściowe drabiny rezystorowej.
- Następnie odszukaj pin mikrokontrolera, który jest połączony z wcześniej znalezionym wyprowadzeniem.
- Skonfiguruj ADC1 w taki sposób, aby możliwe było zmierzenie napięcia wyjściowego drabinki rezystorowej poprzez wcześniej znaleziony pin mikrokontrolera.
- „Surowy” wynik pomiaru wyświetl w pierwszej linii wyświetlacza LCD.
- W drugiej linii wyświetl wynik przekonwertowany na napięcie (V) korzystając z następującego wzoru: $U = X \cdot U_{ref} / 2^{n-1}$, gdzie X oznacza odczyt z konwertera analogowo-cyfrowego, U_{ref} oznacza napięcie referencyjne (domyślnie 3.3V), natomiast n oznacza rozdzielczość (domyślnie 12).
- Przetestuj poprawność działania programu. Sprawdź, jak zmienia się napięcie w zależności od wciśniętego przycisku.

Wykonanie:

```
int main(void)
{
    /* USER CODE BEGIN 1 */
    uint16_t raw; // Zmienna przechowująca surową wartość odczytana z ADC
    char msg[10]; // Bufor przechowujący wiadomość tekstowa
    /* USER CODE END 1 */

    /* MCU Configuration-----*/

    /* Reset wszystkich periferiów, inicjalizacja interfejsu Flash i licznika SysTick. */
    HAL_Init();

    /* USER CODE BEGIN Init */

    /* USER CODE END Init */

    /* Konfiguracja zegara systemowego */
    SystemClock_Config();

    /* USER CODE BEGIN SysInit */

    /* USER CODE END SysInit */

    /* Inicjalizacja wszystkich skonfigurowanych periferiów */
    MX_GPIO_Init();
    MX_USART2_UART_Init();
    MX_ADC1_Init();
    /* USER CODE BEGIN 2 */
    lcd_init(4, 5 * 8, 2); // Inicjalizacja wyświetlacza LCD
    lcd_clear(); // Wyczyszczenie zawartości wyświetlacza

    char msg_raw[16];
    char msg_volts[16];
    /* USER CODE END 2 */

    /* Pętla nieskończona */
    /* USER CODE BEGIN WHILE */
    while (1)
    {
        /* USER CODE END WHILE */

        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_10, GPIO_PIN_SET); // Ustawienie pinu GPIOA_10 na stan wysoki
        HAL_ADC_Start(&hadc1); // Rozpoczęcie konwersji ADC
        HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY); // Oczekiwanie na zakończenie konwersji ADC
        raw = HAL_ADC_GetValue(&hadc1); // Odczytanie surowej wartości z ADC

        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_10, GPIO_PIN_RESET); // Wyzerowanie pinu GPIOA_10
        lcd_print(1, 1, msg); // Wyświetlenie wiadomości na wyświetlaczu LCD
        float napięcie_referencyjne = 3.3f; // Napięcie referencyjne dla ADC
        float maksymalna_wartość_ADC = 4095.0f; // Maksymalna wartość ADC dla rozdzielczości 12-bitowej
        float napięcie = (raw * napięcie_referencyjne) / maksymalna_wartość_ADC; // Obliczenie napięcia na podstawie surowej wartości ADC
        uint16_t czesc_calkowita = (uint16_t)napięcie; // Część całkowita napięcia
        uint16_t czesc_ulkamkowa = (uint16_t)((napięcie - czesc_calkowiec * 100); // Część ułamkowa napięcia, np. dwa cyfry po przecinku

        sprintf(msg, "%hu", raw); // Konwersja surowej wartości ADC na tekst
        sprintf(msg_volts, "%hu.%02hu V", czesc_calkowita, czesc_ulkamkowa); // Konwersja wartości napięcia na tekst z formatowaniem do dwóch miejsc po przecinku
        lcd_clear(); // Wyczyszczenie zawartości wyświetlacza
        lcd_print(1, 1, msg); // Wyświetlenie surowej wartości ADC
        lcd_print(2, 1, msg_volts); // Wyświetlenie wartości napięcia w woltach
        HAL_UART_Transmit(&huart2, (uint8_t *)msg, strlen(msg), HAL_MAX_DELAY); // Wysłanie wiadomości przez interfejs UART
        HAL_Delay(300); // Opóźnienie 300 milisekund
    }
    /* USER CODE BEGIN 3 */
    /* USER CODE END 3 */
}
```

Działanie:





Porównanie wartości wyświetlanych na wyświetlaczu z rzeczywistymi (zmierzone multymetrem):

Brak wciśniętego przycisku	2.89 V	2.99 V
Select	2.31 V	2.39 V
Left	1.73 V	1.78 V
Down	1.2 V	1.25 V
Up	0.55 V	0.59 V
Right	0 V	0 V

Wnioski:

Różnice między napięciami rzeczywistymi, a tymi, które są widoczne na płytce wynikają z kilku czynników. Moduł ADC, którego użyłem ma rozdzielczość 12-bitową, co oznacza że może on osiągnąć maksymalną wartość wynoszącą $2^{12} - 1$. Mikrokontrolery mają również wewnętrzne napięcie referencyjne, które wykorzystują do pomiarów analogowych. Jeśli to napięcie nie jest stabilne, może wpływać na dokładność pomiarów.