



Politechnika
Śląska



UCZELNIA
BADAWCZA
INICJATYWA DOSKONAŁOŚCI

Raport Końcowy

Rodzaj zajęć

Projekt

Przedmiot

Systemy Mikroprocesorowe i Wbudowane

Rok akademicki		Miasto	Kierunek	Semestr	Prowadzący	Grupa	Sekcja
2025/2026		G	INF	5	MP	2	4
Planowany termin wykonywania ćwiczenia			Faktyczny termin wykonania ćwiczenia				
Data		Godzina	Data			Godzina	
Numer ćwiczenia	Temat ćwiczenia						
	Lokalizator GPS/GSM						

Skład sekcji		
	Imię i nazwisko	Uwagi
1	Michał Figołuszka	
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		

Spis treści

Opis i funkcjonalność zrealizowanego układu.....	2
Wykorzystane moduły i elementy elektroniczne	3
Wykorzystane narzędzia i oprogramowanie	5
Specyfikacja wewnętrzna	5
Schemat blokowy	6
Schemat ideowy	6
Opis bloków układu na podstawie schematu ideowego.....	7
Schemat montażowy (PCB)	9
Algorytm oprogramowania urządzenia.....	9
Opis ważniejszych funkcji i procedur programowych urządzenia.....	11
Opis interakcji oprogramowania z układem elektronicznym	14
Specyfikacja zewnętrzna	15
Opis funkcji elementów sterowania urządzeniem	15
Opis elementów wykonawczych urządzenia.....	15
Reakcje urządzenia na zdarzenia zewnętrzne	16
Skrócona instrukcja obsługi urządzenia	17
Opis montażu i uruchamiania.....	18
Napotkane problemy oraz ich rozwiązania	18
Przeprowadzone testy urządzenia	20
Wnioski z testów	21
Literatura i załączniki.....	22

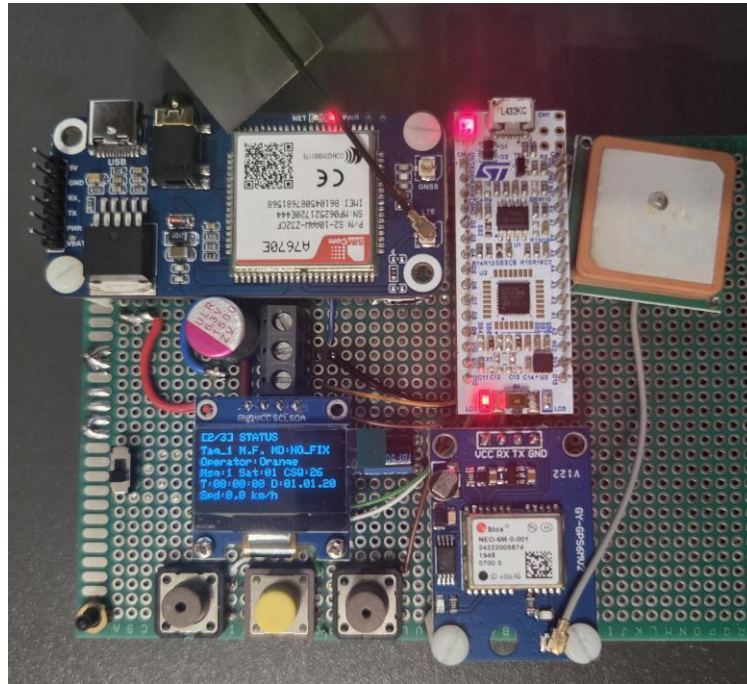
Opis i funkcjonalność zrealizowanego układu

Stworzone urządzenie o przyjętej nazwie „Lokalizator GPS/GSM”, jest systemem geolokalizacyjnym, przesyłającym swoje położenie w czasie rzeczywistym na zewnętrzny serwer brokera MQTT, poprzez konsumencką infrastrukturę sieciową GSM-LTE. Oprócz komunikacji sieciowej, urządzenie pozwala również na odczyt najważniejszych informacji i parametrów pracy na wbudowanym ekranie wyświetlającym prosty i czytelny interfejs użytkownika.

Najważniejsze komponenty urządzenia:

- Układ sterujący bazujący na mikrokontrolerze STM32, zarządzający komunikacją między innymi komponentami.
- Moduł GSM-LTE umożliwiający komunikację z siecią komórkową.
- Moduł GPS zbierający aktualne współrzędne geograficzne urządzenia.
- Ekran OLED wyświetlający interfejs użytkownika z parametrami pracy urządzenia w czasie rzeczywistym.

Architektura zastosowanego systemu opiera się na głównej pętli obsługującej wbudowane funkcje potrzebne do niezawodnej pracy urządzenia, oraz obsłudze przerwań wywoływanych przez powyższe moduły. Użytkownik ma do dyspozycji 3 pulpity, które może przełączać za pomocą wbudowanych w układ przycisków funkcyjnych, oraz ma dostępny podgląd aktualnej godziny, daty i pozycji lokalizatora z poziomu przeglądarkowego klienta brokera MQTT.

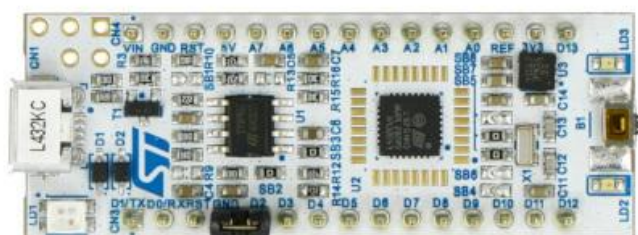


P.1 Zdjęcie układu bez obudowy podczas pracy.

Wykorzystane moduły i elementy elektroniczne

- **Mikrokontroler STM32 NUCLEO-L432KC**

Wybrany jako główny układ sterujący urządzenia ze względu na kompaktowość, niski pobór mocy, aspekt finansowy (był dostępny do zastosowania), oraz rozbudowane możliwości komunikacyjne obejmujące interfejsy: USB, SAI, 2x I2C, 2x SPI, 3x UART, CAN, SWPMI, IRTIM; w tym również 26 programowalnych pinów z tolerancją napięcia do 5V. Mikrokontroler jest wspierany przez przyjazne w użytkowaniu środowisko STM32CubeIDE pozwalające na tworzenie oprogramowania w języku C.



P.2 NUCLEO-L432KC

- **Moduł GPS „GY-NEO6MV2” oparty o układ NEO-6M-0-001**

Zastosowany ze względu na niezawodność pracy i dokładność pozycjonowania lokalizacji. Moduł komunikuje się z STM32 poprzez interfejs UART, cyklicznie przysyłając ramki NMEA z których pozyskiwana jest prędkość, data, czas UTC, oraz współrzędne geograficzne. Używany w komunikacji baudrate jest ustawiony na 9600, natomiast maksymalny czas uzyskania odczytu z satelit (Fix GPS) wynosi do kilku minut w przypadku zimnego startu w niesprzyjających warunkach zewnętrznych.



P.3 GY-NEO6MV2

- **Moduł GSM-LTE „A7670E Cat-1/GNSS HAT” firmy Waveshare oparty o układ SIMCom A7670E**

Zaimplementowany w urządzeniu moduł został wybrany, gdyż zastosowany w nim układ SIMCom A7670E wypadł w testach pomyślnie, spełniając wszystkie wymagania potrzebne w projekcie. Moduł wspiera szeroki zakres pasm od 2G do LTE, natomiast zastosowana w projekcie karta SIM operatora ORANGE z aktywowaną taryfą „ORANGE IoT na karte” wspiera komunikację za pomocą protokołu MQTT. Moduł GSM-LTE jest najbardziej energochłonnym elementem całego układu, lecz dzięki logice programowej jego aktywność jest kontrolowana przez układ sterujący, z którym komunikacja odbywa się poprzez pin „PWR”, oraz interfejs UART.



P.4 A7670E Cat-1/GNSS HAT

- **Ekran OLED 0,96" 128x64px ze sterownikiem SSD1306**

Zastosowany w urządzeniu niewielki wyświetlacz OLED, pozwala na wyświetlanie w czasie rzeczywistym istotnych informacji przedstawiających pracę układu. Ekran wyświetla piksele w kolorze jasnoniebieskim/białym, używając do tego sterownika SSD1306 który posiada pełne wsparcie w postaci oprogramowania dla STM32. Kontrola nad wyświetlaczem przez układ sterujący odbywa się poprzez magistralę I2C.



P.5 Wyświetlacz OLED SSD1306

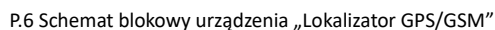
Wykorzystane narzędzia i oprogramowanie

- **IDE:** STM32CubeIDE – tworzenie oprogramowania dla układu sterującego STM32, debugowanie, kompilacja, flash i zarządzanie bibliotekami wykorzystanymi w projekcie.
- **Broker MQTT:** HiveMQ – obsługa odbierania wiadomości z danymi przesyłanymi przez urządzenie, tymczasowe przechowywanie danych poprzez klienta.
- **Testowanie GSM:** SSCOM.exe by Tintin – testy komend AT i komunikacji z modułem GSM.
- **Dokumentacja kodu:** Doxygen – generowanie dokumentacji w formacie HTML/LaTeX na podstawie komentarzy kodu.
- **Dokumentacja etapowa i końcowa:** Pakiet Microsoft Office – prezentacja wstępna, karta projektu, raport końcowy.
- **Projekt schematów/PCB:** Autodesk EAGLE – schemat ideowy oraz schemat montażowy (PCB) projektu.
- **Realizacja fizyczna projektu:** Przyrządy lutownicze oraz elementy układu – montaż układu na uniwersalnej płytce PCB.
- **Pomiary i testy obwodów:** Multimetr cyfrowy UNI-T, Oscyloskop – testowanie napięć i stabilności pracy układu zasilania, oraz testy stanów logicznych na portach modułów.

Specyfikacja wewnętrzna

Poniżej znajdują się szczegółowe informacje związane z tematem specyfikacji wewnętrznej urządzenia:

Schemat ideowy

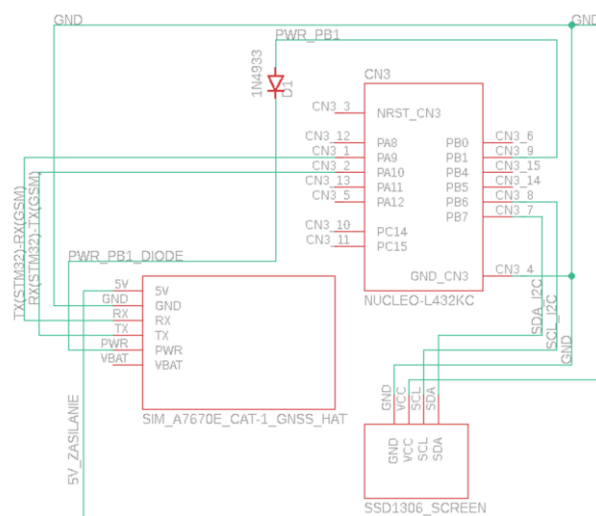


Schemat ideowy

[illegible]

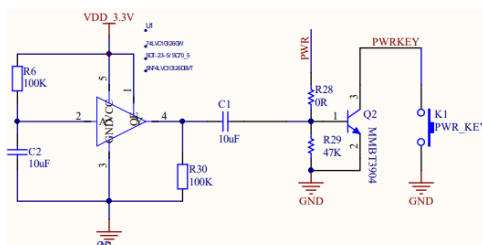
6

Opis bloków układu na podstawie schematu ideowego



P.7 Zbliżenie na omawiany poniżej fragment schematu ideowego.

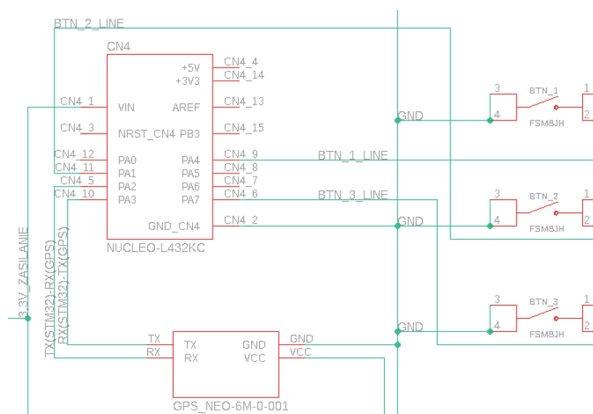
- Moduł GSM „A7670E Cat-1/GNSS HAT” wykorzystuje jeden z dwóch użytych w projekcie interfejsów UART mikrokontrolera STM32. Piny „PA_9” i „PA_10” mikrokontrolera STM32 zostały zaprogramowane jako „Serial1”, odpowiednio „TX” i „RX”, podłączone krzyżowo do pinów „TX” i „RX” w module GSM. Kontrola pracy modułu GSM odbywa się za pomocą pinu „PB_1”, zaprogramowanego jako wyjście ustawiające stały stan logiczny w celu kontroli pracy modułu GSM. Ustawienie stanu wysokiego na czas 3 sekund na pinie „PB_1” STM32 podłączonym do pinu „PWR” modułu GSM, powoduje jego włączenie lub wyłączenie, zgodnie z logiką stworzoną przez producenta modułu GSM.



P.8 Fragment schematu ideowego modułu GSM „A7670E Cat-1/GNSS HAT” firmy Waveshare. Pełny schemat modułu jest dostępny za pośrednictwem linku zamieszczonego w sekcji „Literatura i załączniki”,

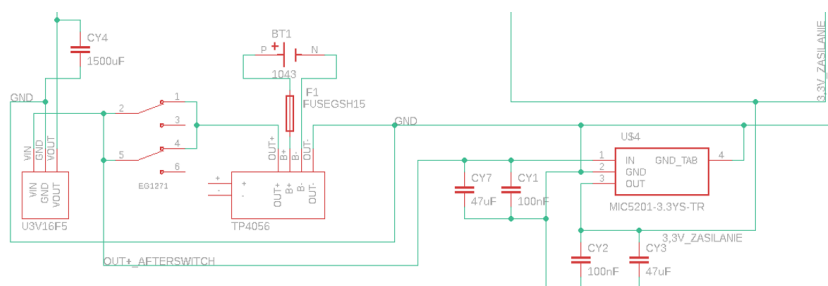
Zastosowanie diody na linii pinów „PB_1” – „PWR” z katodą skierowaną w stronę pinu „PWR” modułu GSM, zapobiega upływności sygnału startowego do masy poprzez mikrokontroler STM32 podczas uruchomienia urządzenia. Napięcie logiczne w wymienionych powyżej rozwiązaniach wynosi 3,3V. Napięcie zasilania modułu GSM wynosi 5V.

- Wyświetlacz OLED ze sterownikiem SSD1306 jest podłączony poprzez magistralę I2C do tożsamo zaprogramowanych pinów „PB_6” i „PB_7” w mikrokontrolerze STM32. Z racji zastosowania przez producenta wyświetlacza rezystorów podciągających na liniach z pinami „SDA” i „SCL”, piny te mogą być podłączone bezpośrednio do mikrokontrolera. Napięcie logiczne w komunikacji wyświetlacz – STM32, oraz napięcie zasilania wyświetlacza wynosi 3,3V.



P.9 Zbliżenie na omawiany poniżej fragment schematu ideowego.

- Moduł GPS „GY-NEO6MV2” oparty o układ NEO-6M-0-001, wykorzystuje drugi użyty w projekcie interfejs UART mikrokontrolera STM32. Piny „PA_2” i „PA_3” mikrokontrolera STM32 zostały zaprogramowane jako „Serial2”, odpowiednio „TX” i „RX”, podłączone krzyżowo do pinów „TX” i „RX” w module GSM. Napięcie logiczne w komunikacji GPS – STM32, oraz napięcie zasilania modułu GPS i mikrokontrolera STM32 wynosi 3,3V.
- Wejścia przycisków „BTN_1”, „BTN_2” i „BTN_3” zostały podłączone odpowiednio do pinów „PA_4”, „PA_1” i „PA_7” mikrokontrolera STM32, zaprogramowanych jako wyjścia ze stałym stanem logicznym o napięciu 3,3V, oraz posiadające rezystory podciągające. Poprzez wciśnięcie przycisków, piny mikrokontrolera są zwierane do masy i następuje upłynność napięcia która jest wykrywana programowo.



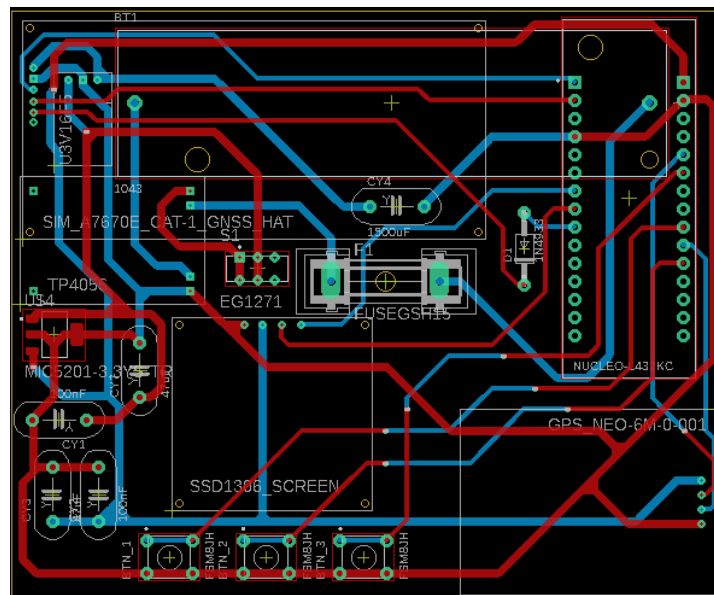
P.10 Zbliżenie na omawiany poniżej fragment schematu ideowego.

- Urządzenie jest zasilane poprzez akumulator litowo-jonowy 3,6V podłączony do modułu „TP4056”, pełniącego rolę systemu BMS zarządzającego ładowaniem i rozładowywaniem akumulatora. Dodatkowo na dojściu napięcia z akumulatora został zastosowany bezpiecznik topikowy 2A zabezpieczający resztę urządzenia przed niekontrolowanym działaniem akumulatora.
- Na wyjściu modułu „TP4056”, linia zasilania rozdziela się na dwa bloki o różnych napięciach wyjściowych. Pierwszym jest przetwornica step-up Pololu U3V16F5 która podnosi napięcie z 3,6V na 5V. Dodatkowo równolegle na wyjściu przetwornicy został zastosowany kondensator o pojemności 1500uF, w celu zapewnienia stabilności napięciowej zasilania w przypadku piku poboru energii przez moduł GSM.

- Drugim blokiem zasilania jest stabilizator napięciowy MIC5201 3.3YS-TR z równolegle podłączonymi w parach na jego wejściu i wyjściu kondensatorami o pojemnościach 47uF i 100nF, w celu filtrowania zakłóceń, oraz wygładzania napięcia wyjściowego 3,3V.

Moduł:	Napięcie zasilania:	Napięcie logiki/wyjściowe
STM32 NUCLEO-L432KC	3,3V	3,3V
GPS GY-NEO6MV2	3,3V	3,3V
GSM A7670E Cat-1/GNSS HAT	5V	3,3V
Ekran OLED SSD1306	3,3V	3,3V
BMS TP4056	5V (USB)	3,6V

Schemat montażowy (PCB)



P.11 Schemat montażowy (PCB) urządzenia „Lokalizator GPS/GSM”. Dostępny również jako plik „PCB.Lokalizator.GPS.GSM_Michał.Figołuska.png” w zakładce „Hardware”.

Algorytm oprogramowania urządzenia

Ze względu na znaczną ilość funkcji zaimplementowanych w kodzie źródłowym oprogramowania mikrokontrolera STM32, plik raportu zawiera jedynie opis najważniejszych fragmentów kodu. Pełna dokumentacja kodu źródłowego znajduje się w zakładce „Dokumentacja” w pliku:

- Doxygen_Lokalizator.GPS.GSM_Michał.Figołuska.pdf

Wybrane fragmenty kodu źródłowego:

- Rozruch urządzenia na początku funkcji main:

```
int main(void)
{
    /* Reset of all peripherals, Initializes the Flash interface and the Systick.
    */
    HAL_Init();

    /* Configure the system clock */
    SystemClock_Config();

    /* Initialize all configured peripherals */
    MX_GPIO_Init();
    MX_USART2_UART_Init();
    MX_I2C1_Init();
    MX_USART1_UART_Init();
    ssd1306_Init();
    ssd1306_Fill(Black);
    ssd1306_WriteString("Starting...", Font_11x18, White);
    ssd1306_UpdateScreen();
    HAL_Delay(10000); // Oczekiwanie na start GSM

    // 1. INICJALIZACJA GSM (DIAGNOSTYKA + KONFIGURACJA)
    GSM_Init_Sequence();

    // 2. WŁĄCZENIE PRZERWAŃ
    HAL_UART_Receive_IT(&huart1, &rx_char_buffer, 1); // GSM
    HAL_UART_Receive_IT(&huart2, &gps_rx_char, 1);      // GPS
    uint32_t last_mqtt_time = 0;
    uint32_t current_time = 0;

    // 3. WYSŁANIE JEDNORAZOWEJ WIADOMOŚCI
    GSM_PublishMQTT_Async("Turned ON");
    HAL_Delay(1000);
    char display_buff[32];
    char mqtt_msg[128];
    /* Infinite loop */
    while (1)
    {
        ... Rozruch zakończony, dalsza obsługa programu w pętli while.
    }
}
```

- Logika zasilania modułu GSM i wysyłania wiadomości na serwer brokera MQTT:

```
// 1. Ustalenie interwału czasowego na podstawie stanu
if (gps_fix_valid) {
    if (gps_speed_kmh > 20.0) {
        current_interval = 10000; // FAST (10s)
        strcpy(current_mode, "FAST");
    } else if (gps_speed_kmh > 3.0) {
        current_interval = 30000; // SLOW (30s)
        strcpy(current_mode, "SLOW");
    } else { // Jeśli bez ruchu krócej niż 2 minuty (120000 ms) -> tryb SLOW
        if (HAL_GetTick() - last_movement_time < 120000) {
            current_interval = 30000; // Oczekiwanie na uśpienie
            strcpy(current_mode, "SLOW*");
        } // Jeśli bez ruchu dłużej niż 2 minuty -> tryb STATIC
        else {
            ...
        }
    }
}
```

```

        current_interval = 3600000; // STATIC (1h)
        strcpy(current_mode, "STATIC");
    }
} else {
    current_interval = 3600000; // NO FIX (1h)
    strcpy(current_mode, "NO_FIX");
}

// 2. Obsługa usypiania GSM przy długim interwale
if (current_interval == 3600000) {
    if (gsm_powered_on && (current_time - last_mqtt_time > 10000) &&
        (current_time - last_mqtt_time < current_interval - 30000)) {
        GSM_Power_Off();
    }
    else {
        if (!gsm_powered_on) GSM_Power_On();
    }
}

// 3. WYSYŁANIE
if (current_time - last_mqtt_time > current_interval){
    last_mqtt_time = current_time;
    if(!gsm_powered_on) {
        GSM_Power_On();
    }
    GSM_Watchdog_Check();
    if (gps_fix_valid) {
        // Format wiadomości: "GPS: 50.055721, 18.927776 Time: 16:00:08 Date: 03.01.26"
        sprintf(mqtt_msg, "GPS: %.6f, %.6f Time: %s Date: %s",
            gps_lat_decimal, gps_lon_decimal, gps_time_str, gps_date_str);
    }
    else {
        // Informacja o braku fixa + ostatnie znane satelity
        sprintf(mqtt_msg, "No Fix. Satelites: %s", gps_sats);
    }
    GSM_PublishMQTT_Async(mqtt_msg);
    HAL_Delay(200);
    HAL_UART_Transmit(GSM_UART, (uint8_t*)"AT+CSQ\r\n", 8, 100);
}

```

Opis ważniejszych funkcji i procedur programowych urządzenia

- Funkcja odpowiedzialna za obsługę wysyłania wiadomości na serwer brokera MQTT:

```

void GSM_PublishMQTT_Async(char *payload)
{
    char cmd[64];
    int len = strlen(payload);
    // Funkcja wysyła na interfejs UART modułu GSM komendy AT z zawartością
    // 1. TEMAT
    HAL_UART_Transmit(GSM_UART, (uint8_t*)"AT+CMQTTTOPIC=0,4\r\n", 19, 100);
    HAL_Delay(100); // Krótki delay do synchronizacji
    HAL_UART_Transmit(GSM_UART, (uint8_t*)"Tag1", 4, 100);
    HAL_Delay(100);
    // 2. DŁUGOŚĆ WIADOMOŚCI
    sprintf(cmd, "AT+CMQTTTPAYLOAD=0,%d\r\n", len);
}

```

```

HAL_UART_Transmit(GSM_UART, (uint8_t*)cmd, strlen(cmd), 100);
HAL_Delay(100);
// 3. TREŚĆ WIADOMOŚCI
HAL_UART_Transmit(GSM_UART, (uint8_t*)payload, len, 100);
HAL_Delay(100);
// 4. WYŚLIJ
HAL_UART_Transmit(GSM_UART, (uint8_t*)"AT+CMQTPUB=0,0,60\r\n", 20, 100);
tx_msg_counter++; // Inkrementacja licznika wysłanych wiadomości
}

```

- Funkcja parsująca ramki NMEA cyklicznie przesyłane przez moduł GPS:

```

void GPS_Parse_GPRMC(char *nmea)
{
    char *p = strstr(nmea, "$GPRMC,"); // Szukanie początku ramki
    if (!p) p = strstr(nmea, "RMC,"); // Zabezpieczenie na wypadek ucięcia
    ramki
    if (!p) return;

    p = strchr(p, ','); // Przesunięcie wskaźnika za nagłówek ramki
    if (!p) return;
    p++;
    // Bufory pomocnicze i zmienne tymczasowe na odczyt danych
    char field_buffer[20];
    int field_index = 0;
    int char_index = 0;
    char t_time[16] = {0};
    char t_status = 'V';
    char t_lat[16] = {0}; char t_ns = 'N';
    char t_lon[16] = {0}; char t_ew = 'E';
    char t_speed[10] = {0};
    char t_date[16] = {0};
    // Parsowanie znak po znaku całej ramki aż do sumy kontrolnej „*”
    while (*p != '*' && *p != '\0' && *p != '\r' && *p != '\n')
    {
        if (*p == ',')
        {
            field_buffer[char_index] = 0;
            switch(field_index) {

                // Czas UTC
                case 0: strncpy(t_time, field_buffer, 15); break;
                // Status połączenia
                case 1: if(char_index > 0) t_status = field_buffer[0]; break;
                // Szerokość geograficzna
                case 2: strncpy(t_lat, field_buffer, 15); break;
                // Półkule N/S
                case 3: if(char_index > 0) t_ns = field_buffer[0]; break;
                // Długość geograficzna
                case 4: strncpy(t_lon, field_buffer, 15); break;
                // Półkule E/W
                case 5: if(char_index > 0) t_ew = field_buffer[0]; break;
                // Prędkość w węzłach
                case 6: strncpy(t_speed, field_buffer, 9); break;
                // Data
                case 8: strncpy(t_date, field_buffer, 15); break;
            }
            char_index = 0;
        }
        p++;
    }
}

```

```

        field_index++;
    }
    else
    {
        if (char_index < 19) field_buffer[char_index++] = *p;
    }
    p++;
}

// Analiza i przetwarzanie danych
if (t_status == 'A') {
    gps_fix_valid = 1;
    gps_lat_decimal = NMEA_To_Decimal(t_lat, t_ns);
    gps_lon_decimal = NMEA_To_Decimal(t_lon, t_ew);

    float current_kmh = 0.0f;
    if (strlen(t_speed) > 0) {
        float knots = atof(t_speed);
        current_kmh = knots * 1.852f; // Konwersja na km/h
    }

    if (current_kmh < 1.5f) current_kmh = 0.0f; // Filtr driftu GPS

    speed_buffer[speed_buf_idx] = current_kmh; // Średnia krocząca
    speed_buf_idx++;
    if (speed_buf_idx >= 5) speed_buf_idx = 0;

    float sum = 0.0f;
    for(int i=0; i<5; i++) sum += speed_buffer[i];
    gps_speed_kmh = sum / 5.0f;

    if(strlen(t_time) >= 6) { // Formatowanie czasu
        sprintf(gps_time_str, "%c%c:%c%c:%c%c", t_time[0],t_time[1],
t_time[2],t_time[3], t_time[4],t_time[5]);
    }
    if(strlen(t_date) >= 6) { // Formatowanie daty
        sprintf(gps_date_str, "%c%c.%c%c.%c%c", t_date[0],t_date[1],
t_date[2],t_date[3], t_date[4],t_date[5]);
    }

} else { // Na wypadek braku Fixa
    gps_fix_valid = 0;
    gps_speed_kmh = 0.0;

    for(int i=0; i<5; i++) speed_buffer[i] = 0.0f; // Reset prędkości

    if(strlen(t_time) >= 6) {
        sprintf(gps_time_str, "%c%c:%c%c:%c%c", t_time[0],t_time[1],
t_time[2],t_time[3], t_time[4],t_time[5]);
    }
}
}

```

Opis interakcji oprogramowania z układem elektronicznym

Warstwa sprzętowa:

- **STM32 NUCLEO-L432KC:** główny układ sterujący, UART1 (GSM), UART2 (GPS), I2C (Ekran), piny programowalne „PA_4”, „PA_1”, „PA_7”, oraz „PB_1” (Przyciski i GSM).
- **GPS GY-NEO6MV2:** moduł GPS przesyłający ramki NMEA z danymi, UART (STM32), baudrate 9600.
- **GSM A7670E Cat-1/GNSS HAT:** moduł GSM do komunikacji z serwerem brokera MQTT, UART (STM32), pin „PWR” (STM32 – pin „PB_1”).
- **Ekran OLED SSD1306:** wyświetla interfejs użytkownika z parametrami pracy urządzenia, I2C (STM32).
- **Układ zasilania:** ładowanie akumulatora, regulacja napięcia akumulatora na napięcia pracy poszczególnych modułów urządzenia.

Warstwa programowa:

- **Rozruch systemu:** konfiguracja zegara, konfiguracja peryferiów, inicjalizacja modułu GSM, inicjalizacja połączenia z brokerem MQTT, uruchomienie przerwań UART1 i UART2.
- **Pętla while(1) we funkcji main:** obsługa GPS, obsługa interfejsu użytkownika, obsługa logiki wysyłania wiadomości MQTT, obsługa logiki pracy modułu GSM.
- **Funkcje komunikujące się z modułem GPS:** parsowanie ramek NMEA wysyłanych przez moduł w sekundowym odstępie czasowym, czyszczenie bufora UART2.
- **Funkcje komunikujące się z modułem GSM:** utrzymywanie stabilnego połączenia z siecią GSM-LTE, utrzymywanie połączenia z serwerem brokera MQTT, wysyłanie wiadomości MQTT, czyszczenie bufora UART1.
- **Funkcja „Handle_Buttons”:** obsługa logiki wykrywania stanów przycisków kontrolnych, tworzenie i zarządzanie zmiennymi dla logiki obsługi interfejsu użytkownika.

Specyfikacja zewnętrzna

Poniżej znajdują się szczegółowe informacje związane z tematem specyfikacji zewnętrznej urządzenia:

Opis funkcji elementów sterowania urządzeniem

- **GPS GY-NEO6MV2:** za pośrednictwem interfejsu UART przesyła do mikrokontrolera STM32 ramki NMEA z których jest pobierana aktualna data, czas UTC, współrzędne geograficzne, liczba wykrytych satelit, prędkość (przeliczana z węzłów na km/h), oraz status Fix GPS.
- **GSM A7670E Cat-1/GNSS HAT:** komunikacja obustronna odbywa się za pośrednictwem interfejsu UART, poprzez przysyłanie komend AT dotyczących wstępnej konfiguracji modułu, ustanawiania połączenia, wysyłania wiadomości, oraz pobierania informacji o operatorze i sile sygnału sieci GSM-LTE. Aktywność modułu GSM jest kontrolowana poprzez logikę programową.
- **Przyciski funkcyjne:** za pomocą dwóch skrajnych przycisków, użytkownik może zmieniać aktualnie wyświetlany pulpit, natomiast środkowy przycisk służy do wygaszania ekranu i wznowiania jego pracy.
- **Układ zasilania:** stabilność zasilania przekłada się na stabilność pracy urządzenia. Gdy urządzenie przeprowadza rozruch, wysyła o tym informację na serwer brokera MQTT.

Opis elementów wykonawczych urządzenia

- **Ekran OLED SSD1306:** wyświetla ekran powitalny, sekwencję rozruchu oraz pulpity z informacjami w zależności od wyboru. Dostępne są trzy pulpity: 1. GPS, 2. STATUS, 3. DIAGNOSTICS. Ekran może być wygaszany przez użytkownika środkowym przyciskiem, nie wpływa to na pracę urządzenia.



P.12 Wygląd głównych pulpików lokalizatora w trybie NO_FIX.

Reakcje urządzenia na zdarzenia zewnętrzne

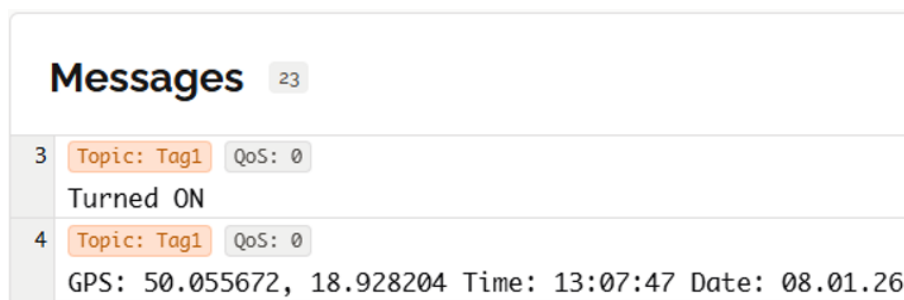
- **Wciśnięcie przycisków:** W zależności od danego przycisku: lewy i prawy przycisk przełączają pulpity, środkowy przycisk wygasza oraz wybudza ekran. Gdy ekran jest wygaszony, lewy i prawy przycisk nie są aktywne, po wybudzeniu ekranu wyświetlany jest ten sam pulpit który był przed jego wygaszeniem.
- **Utrata Fixa GPS:** Przejście do trybu NO_FIX, pulpit „1/3 GPS” wyświetla komunikat „Waiting for Fix...”, „No GPS connection” zamiast współrzędnych geograficznych. Wówczas lokalizator co godzinę wybudza moduł GSM i wysyła wiadomość na serwer brokera MQTT z komunikatem o braku Fixa GPS oraz z liczbą wykrytych satelit.
- **Synchronizacja czasu i daty z GPS:** Po pierwszym uzyskaniu stabilnego połączenia GPS, lokalizator ustawia czas systemowy wraz z datą na podstawie aktualnie przesyłanych danych z modułu GPS. W przypadku braku połączenia, aktualny czas i data są nieznane.
- **Prędkość mniejsza niż 3km/h:** W przypadku czasu krótszego niż 2 minuty od zarejestrowania prędkości mniejszej niż 3km/h, przejście do trybu SLOW, oczekując na przejście do trybu STATIC. Wówczas lokalizator wysyła co 30 sekund wiadomość na serwer brokera MQTT z aktualnymi współrzędnymi geograficznymi, datą i godziną. W przypadku przekroczenia czasu 2 minut, przejście do trybu STATIC. Wówczas lokalizator co godzinę wybudza moduł GSM i wysyła wiadomość na serwer brokera MQTT z aktualnymi współrzędnymi geograficznymi, datą i godziną.
- **Prędkość w przedziale 3km/h – 20km/h:** Przejście do trybu SLOW, lokalizator wybudza moduł GSM jeśli ten jest aktualnie uśpiony, po czym wysyła co 30 sekund wiadomość na serwer brokera MQTT z aktualnymi współrzędnymi geograficznymi, datą i godziną.
- **Prędkość większa niż 20km/h:** Przejście do trybu FAST, lokalizator wysyła co 10 sekund wiadomość na serwer brokera MQTT z aktualnymi współrzędnymi geograficznymi, datą i godziną.
- **Brak połączenia z siecią GSM-LTE / brak reakcji modułu GSM / moduł GSM jest wyłączony:** Za każdym razem przed wystaniem wiadomości na serwer brokera MQTT, watchdog sprawdza czy moduł GSM jest włączony i czy ma połączenie, jeśli nie to rozpoczyna jego restart i konfigurację. Podczas uśpienia modułu GSM, operator zmienia nazwę na „SLEEPING”, natomiast podczas rozruchu, w przypadku gdy ekran lokalizatora nie jest wygaszony, na ekranie chwilowo wyświetla się komunikat „Connection Lost! REBOOTING GSM...”, po czym pojawia się timer odliczający w dół sekundy od 15. Po tej sekwencji następuje standardowa inicjalizacja modułu GSM i konfiguracja połączenia.
- **Brak karty SIM / uszkodzona karta SIM / błąd karty SIM:** Lokalizator podczas rozruchu sprawdza kartę sim i status połączenia z siecią, gdy napotka problemy, wyświetli na ekranie komunikat „ERROR” lub „NO SIM”. Na wypadek utraty ważności karty SIM lub innego problemu z jej aktywacją już po rozruchu urządzenia, operator zmieni nazwę na „Nieznany”.

- **Problem z rejestracją w sieci GSM-LTE:** Lokalizator podczas rozruchu sprawdza rejestrację modułu GSM w sieci. Wykrywa czy połączenie jest lokalne czy też roamingowe wyświetlając stosowny komunikat „Home” lub „Roaming”. W przypadku braku połączenia lub odrzucenia przez sieć, wyświetli komunikat „Searching” lub „Denied”.
- **Ładowanie urządzenia:** Gdy urządzenie jest ładowane, przy złączu USB-C zaświeci się czerwona dioda, natomiast gdy urządzenie zostanie naładowane, to zaświeci się niebieska dioda.

Skrócona instrukcja obsługi urządzenia

1. Przygotowanie:

Przed uruchomieniem lokalizatora, zaleca się zalogowanie na stronie internetowej klienta brokera MQTT (w tym przypadku wykorzystywany broker to HiveMQ), w celu dostępu do danych przesyłanych przez lokalizator.



P.13 Przykładowe dane widoczne na stronie klienta brokera MQTT HiveMQ.

2. **Ładowanie urządzenia:** W celu ładowania, podepnij urządzenie do sieci elektrycznej za pomocą zasilacza o parametrach **5V, 1A (moc 5W)**. Gdy urządzenie będzie naładowane, zobaczysz niebieską diodę obok złącza ładowania USB-C.

3. Uruchomienie:

Aby włączyć urządzenie, przesunij przełącznik po lewej stronie od ekranu. Wówczas na ekranie zostanie wyświetlona sekwencja rozruchowa. Rozruch urządzenia w zależności od warunków zewnętrznych, może zająć od 20 sekund do kilku minut (nawiązanie połączenia GPS).

4. Obsługa urządzenia po rozruchu:

Urządzenie samoczynnie przesyła dane na serwer brokera MQTT (wspomniany w pkt.1). Lokalizator posiada wbudowane trzy przyciski, których wciśnięcie powoduje zmianę w sposobie wyświetlania ekranu:

- **Przycisk lewy:** przełączenie pulpitu wstecz (np. pulpit z 1/3 zmieni się na 3/3).
- **Przycisk prawy:** przełączenie pulpitu wprzód (np. pulpit z 1/3 zmieni się na 2/3).
- **Przycisk środkowy:** powoduje wygaszenie lub wybudzenie wyświetlacza.

5. Zakończenie pracy:

W celu zakończenia pracy urządzenia, ponownie przesunąć przełącznik po lewej stronie ekranu. Dane zarejestrowane podczas pracy lokalizatora będą dostępne za pośrednictwem strony internetowej brokera MQTT.

6. Rozwiązywanie problemów:

- **Naładowane urządzenie nie działa / urządzenie nie ładuje się:** Na spodzie urządzenia (przeciwna strona do ekranu) znajduje się uchwyt z bezpiecznikiem. Sprawdź czy bezpiecznik jest sprawny, jeśli jest przepalony to wymień go na nowy bezpiecznik 2A. W takiej sytuacji może być konieczna wymiana akumulatora, znajduje się on na spodzie urządzenia i jest podłączony dwoma przewodami do dedykowanego złącza.
- **Urządzenie restartuje się podczas pracy:** Urządzenie restartuje tylko jeden ze swoich modułów, co jest oczekiwanym działaniem. W przypadku dłuższego czasu od ostatniej zmiany położenia, moduł GSM jest usypiany. Sekwencja „restartu” zachodzi wtedy, gdy zostanie wykryty błąd połączenia z siecią, lub moduł GSM jest usypiony w momencie gdy ma zostać wysłana wiadomość.
- **Brak połączenia z siecią GSM:** Sprawdź kartę SIM, możliwe że jest nieaktywna lub uszkodzona. Jeśli urządzenie znajduje się w pomieszczeniu, możliwe że nie jest w stanie znaleźć zasięgu sieci GSM, wówczas przenieś je na otwartą przestrzeń.
- **Brak połączenia GPS (brak Fixa GPS):** Przenieś urządzenie na otwartą przestrzeń aby nie znajdowało się pod zadaszeniem. Nawiązanie stabilnego połączenia przez moduł GPS może zająć nawet do kilku minut.

Opis montażu i uruchamiania

Poniżej znajdują się szczegółowe informacje związane z montażu, uruchamiania i testowania urządzenia, a także opis napotkanych błędów:

Napotkane problemy oraz ich rozwiązania

Problem z komunikacją poprzez interfejsy UART:

- **Opis:** Problem następował pomiędzy mikrokontrolerem STM32 a modułami GSM i GPS. Komendy AT nie były poprawnie odczytywane przez moduł GSM, natomiast dane odbierane z modułu GPS mogły być zakłócone.
- **Rozwiązanie:** Zastosowanie czyszczenia buforów przed przesłaniem danych i po ich odczycie dla obu interfejsów UART, usunęło zaistniałe problemy.

Brak połączenia z brokerem MQTT:

- **Opis:** Podczas wstępnych prób połączeniowych z serwerem brokera MQTT następowały problemy z odrzucaniem połączenia przez serwer, bądź braku połączenia z samą siecią.
- **Rozwiązanie:** Testowane moduły GSM oparte o układy takie jak SIMCom SIM800L i SIM7020E, ostatecznie nie nadały się do zastosowania w urządzeniu. SIM800L nie wspierał nowoczesnego szyfrowania wymaganego przez serwer brokera MQTT, natomiast SIM7020E nie wspierał polskich komercyjnych pasm transmisji danych GSM. Zmiana modułu na oparty o układ SIMCom A7670E, rozwiązała problem z połączeniem z siecią, natomiast zmiana operatora karty SIM z PLAY na ORANGE, zaowocowała odblokowaniem portów sieciowych wykorzystywanych przez protokół MQTT, rozwiązując tym samym problem z połączeniem z brokerem MQTT.

Brak automatycznego uruchomienia modułu GSM przy starcie urządzenia:

- **Opis:** Moduł GSM po podłączeniu jego pinu „PWR” do pinu mikrokontrolera STM32, nie uruchamiał się automatycznie podczas startu urządzenia.
- **Rozwiązanie:** Zastosowanie diody na linii połączeniowej pinów obu modułów, z katodą skierowaną w stronę pinu „PWR” modułu GSM, rozwiązało problem upłynności ładunku do masy poprzez mikrokontroler. Tym samym moduł GSM zaczął się poprawnie uruchamiać.

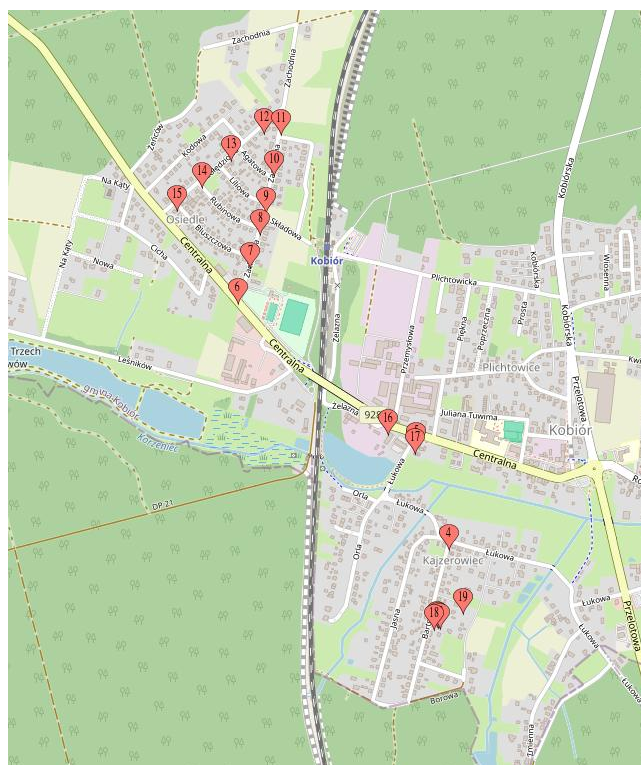
Losowe wyłączenia modułu GSM:

- **Opis:** Podczas wzmożonego zapotrzebowania energetycznego przez moduł GSM, np. podczas łączenia się z siecią, moduł losowo się wyłączał z powodu spadków napięcia na jego zasilaniu.
- **Rozwiązanie:** Zastosowanie kondensatora o pojemności 1500uF, podłączonego równolegle do linii zasilającej 5V, rozwiązało zaistniały problem spadków napięć i wyłączeń modułu GSM. Dodatkowo zastosowany programowy watchdog, zapewnia o stabilności połączenia i wymiany danych z modułem GSM.

Brak dokładności i słaby sygnał wykrywany przez moduł GPS:

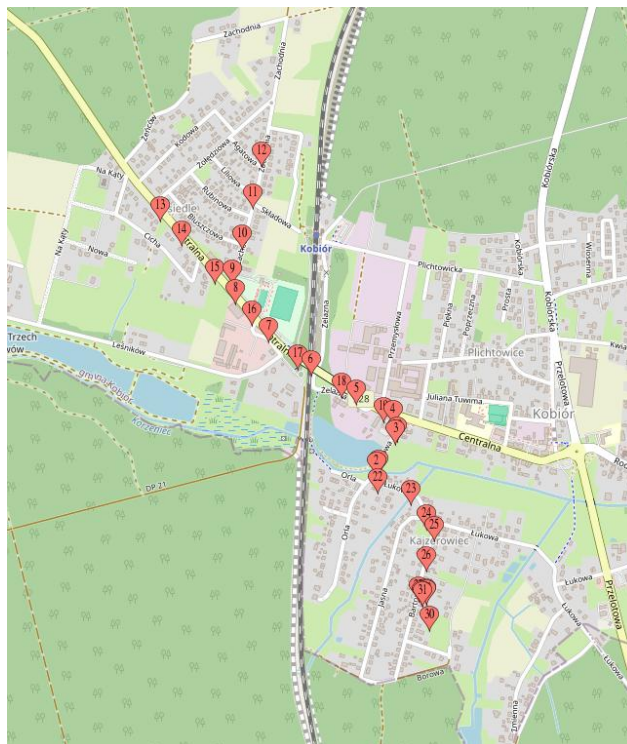
- **Opis:** Podczas wstępnych testów modułu GPS, następowały problemy z jego stabilnym połączeniem z siecią, oraz dokładność lokalizacji podawanej przez moduł okazała się być niezadowalająca.
- **Rozwiązanie:** Wymiana modułu GPS z modułu „FGPMMOPA4” na „GY-NEO6MV2” pozwoliła na rozwiązanie problemu z dokładnością lokalizacji, oraz zapewniła możliwą stabilność połączenia. Nowy moduł zapewnia dokładność podawanej lokalizacji nawet do 2,5m w zależności od liczby wykrytych satelit.

Pierwszy test lokalizatora na trasie w okolicy domu (prędkości 0 - 50km/h):



P.14 Mapa z naniesionymi współrzędnymi przesłanymi przez lokalizator.

Test lokalizatora po dodaniu opóźnienia czasowego pomiędzy przełączaniem trybów pracy (prędkości 0 - 50km/h):



P.15 Mapa z naniesionymi współrzędnymi przesłanymi przez lokalizator.

Wnioski z testów

Przeprowadzone testy pomogły udoskonalić oprogramowanie urządzenia, oraz wykazały poprawne działanie lokalizatora, nawet w przypadku szybszej zmiany jego lokalizacji (lokalizator był testowany w samochodzie).

Obie wyżej ukazane mapy przedstawiają tę samą trasę pokonaną przez lokalizator.

Przed modyfikacją oprogramowania, która polegała na dodaniu 2 minutowego opóźnienia w przełączeniu trybu pracy lokalizatora z trybu SLOW na tryb STATIC, lokalizator wysłał 19 wiadomości ze współrzędnymi. Przełączenie się lokalizatora na tryb STATIC np. podczas postoju przy skrzyżowaniu, powodowało wyłączenie modułu GSM i tym samym konieczność oczekiwania około minuty na ponowną możliwość przesłania wiadomości z lokalizacją.

Test po tej zmianie w oprogramowaniu, wykazał pozytywny efekt modyfikacji, przekładający się na poprawę częstotliwości przesyłania współrzędnych przez lokalizator. Wówczas lokalizator przesłał 31 wiadomości ze współrzędnymi, czyniąc cały rejestr pokonanej trasy dokładniejszym niż podczas pierwszego testu. Krótka przerwa pomiędzy punktami 12 i 13 na drugiej mapie, wynikała z chwilowej utraty połączenia GPS (utrata Fixa GPS). Po przywróceniu połączenia z siecią GPS, urządzenie ponownie zaczęło podawać swoją lokalizację.

Stworzony system geolokalizacyjny z łącznością GSM-LTE, spełnia postawione założenia projektowe i może zostać wykorzystany w przyszłości jako podstawa do dalszego udoskonalania i rozbudowy projektu, np. o dodatkowe czujniki, sieć kilku lokalizatorów lub automatyczną wizualizację wyników w oparciu o aplikację z bazą danych przesyłanych lokalizacji.

Literatura i załączniki

Główna część dokumentacji dotyczącej kodu źródłowego dla mikrokontrolera STM32 znajduje się w zakładce „Dokumentacja” w pliku:

- Doxygen_Lokalizator.GPS.GSM_Michał.Figołuska.pdf

Linki do dokumentacji modułów, oraz informacji przydatnych podczas projektowania urządzenia (dostęp 09.01.2026r.):

- <https://www.st.com/resource/en/datasheet/stm32l432kc.pdf>
- <https://os.mbed.com/platforms/ST-Nucleo-L432KC/>
- <https://forums.mbed.com/t/uart2-on-nucleo-l432kc/15788/2>
- <https://forbot.pl/blog/kurs-stm32l4-wejscia-wyjscia-czyli-gpio-stm32-id46571>
- <https://www.onetransistor.eu/2020/03/usb-multimedia-keys-on-arduino-stm32.html>
- <https://www.st.com/en/development-tools/stm32cubeide.html>
- https://content.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf
- <https://safaelectronics.com/public/uploads/all/VfOdrP3OT3SV6vsZJqgh2KQIW34yyTtjM5vwr vem.pdf>
- https://www.waveshare.com/wiki/A7670E_Cat-1/GNSS_HAT
- https://files.waveshare.com/wiki/A7670E-Cat-1-GNSS-HAT/A76XX_Series_AT_Command_Manual_V1.09.pdf
- https://files.waveshare.com/upload/7/74/A7600_Series_MQTT%28S%29_Application_Note_V1.00.pdf
- https://files.waveshare.com/wiki/A7670E-Cat-1-GNSS-HAT/A7670E_Cat-1-GNSS_HAT.pdf
- https://files.waveshare.com/wiki/A7670E-Cat-1-GNSS-HAT/A7672X_A7670X_Series_Hardware_Design_V1.03.pdf
- <https://kamami.pl/moduly-gps/1189472-cat-1-gsm-gprs-gnss-hat-for-raspberry-pi-based-on-a7670e-module-lte-cat-1-2g-support-gnss-posit-5906623466177.html>
- <https://console.hivemq.cloud/>
- <https://www.orange.pl/view/iot-na-karte>
- <https://botland.com.pl/wyswietlacze-i-ekrany/8866-wyswietlacz-oled-niebieski-graficzny-096-128x64px-i2c-niebieski-5904422337421.html>
- <https://github.com/afiskon/stm32-ssd1306>
- <https://blog.boochow.com/article/upy-oled-128-32-i2c-stm32.html>
- <https://www.autodesk.com/br/products/eagle/overview.acessado>

- <https://botland.com.pl/przetwornice-step-up/22475-u3v16f5-przetwornica-step-up-5v-2a-pololu-4941.html>
- <https://www.pololu.com/product/4941>
- <https://www.mouser.pl/ProductDetail/Microchip-Technology/MIC5201-3.3YS-TR?qs=U6T8BxXiZAWK93UtE5g7UQ%3D%3D>
- <https://www.elektroda.pl/rtvforum/topic3313893.html>
- <https://www.elektroda.pl/rtvforum/topic3486633.html>
- <https://www.mapcustomizer.com/>