

Lokalizator GPS/GSM

Wygenerowano przez Doxygen 1.10.0

1 Topic Index	1
1.1 Topics	1
2 Indeks struktur danych	3
2.1 Struktury danych	3
3 Indeks plików	5
3.1 Lista plików	5
4 Topic Documentation	7
4.1 Definicje Sprzętowe	7
4.1.1 Opis szczegółowy	7
4.2 Zmienne Interfejsu Użytkownika	7
4.2.1 Opis szczegółowy	8
4.3 Zmienne Obsługi Przycisków	8
4.3.1 Opis szczegółowy	8
4.4 Zmienne Modułu GSM	8
4.4.1 Opis szczegółowy	9
4.5 Słowniki GSM	9
4.5.1 Opis szczegółowy	9
4.5.2 Dokumentacja zmiennych	9
4.5.2.1 op_list	9
4.6 Zmienne Modułu GPS	10
4.6.1 Opis szczegółowy	10
4.7 Logika Wysyłania	10
4.7.1 Opis szczegółowy	11
4.8 Funkcje GSM	11
4.8.1 Opis szczegółowy	11
4.8.2 Dokumentacja funkcji	11
4.8.2.1 GSM_CheckNetwork_Reg()	11
4.8.2.2 GSM_CheckSignal_Blocking()	12
4.8.2.3 GSM_CheckSIM_Status()	12
4.8.2.4 GSM_GetOperator_Blocking()	12
4.8.2.5 GSM_Init_Sequence()	12
4.8.2.6 GSM_Power_Off()	13
4.8.2.7 GSM_Power_On()	13
4.8.2.8 GSM_PublishMQTT_Async()	13
4.8.2.9 GSM_SendAT_Checked()	13
4.8.2.10 GSM_Watchdog_Check()	14
4.9 Funkcje GPS	14
4.9.1 Opis szczegółowy	14
4.9.2 Dokumentacja funkcji	14
4.9.2.1 GPS_Parse_GPRMC()	14

4.9.2.2 NMEA_To_Decimal()	15
4.10 Funkcje Interfejsu	15
4.10.1 Opis szczegółowy	15
4.10.2 Dokumentacja funkcji	15
4.10.2.1 Handle_Buttons()	15
5 Dokumentacja struktur danych	17
5.1 Dokumentacja struktury OperatorMapping	17
5.1.1 Opis szczegółowy	17
6 Dokumentacja plików	19
6.1 Dokumentacja pliku main.c	19
6.1.1 Opis szczegółowy	22
6.1.2 Dokumentacja funkcji	22
6.1.2.1 Error_Handler()	22
6.1.2.2 HAL_UART_RxCpltCallback()	23
6.1.2.3 main()	23
6.1.2.4 SystemClock_Config()	23
Skorowidz	25

Rozdział 1

Topic Index

1.1 Topics

Here is a list of all topics with brief descriptions:

Definicje Sprzętowe	7
Zmienne Interfejsu Użytkownika	7
Zmienne Obsługi Przycisków	8
Zmienne Modułu GSM	8
Słowniki GSM	9
Zmienne Modułu GPS	10
Logika Wysyłania	10
Funkcje GSM	11
Funkcje GPS	14
Funkcje Interfejsu	15

Rozdział 2

Indeks struktur danych

2.1 Struktury danych

Tutaj znajdują się struktury danych wraz z ich krótkimi opisami:

[OperatorMapping](#)

Struktura mapująca kod operatora na nazwę 17

Rozdział 3

Indeks plików

3.1 Lista plików

Tutaj znajduje się lista wszystkich udokumentowanych plików z ich krótkimi opisami:

main.c	: Główny plik programu dla projektu "Lokalizator GPS/GSM"	19
------------------------	---	----

Rozdział 4

Topic Documentation

4.1 Definicje Sprzętowe

Definicje

- `#define GSM_UART &huart1`
Uchwyt UART dla modemu GSM (SIM7670E)
- `#define DEBUG_UART &huart2`
Uchwyt UART dla GPS (GPS_NEO-6M-0-001)
- `#define GSM_PWR_PIN GPIO_PIN_1`
Pin sterujący zasilaniem GSM (PWRKEY)
- `#define GSM_PWR_PORT GPIOB`
Port sterujący zasilaniem GSM.
- `#define BTN_MID_PIN GPIO_PIN_1`
Przycisk Środkowy (PA1/A1) - ON/OFF Ekran.
- `#define BTN_LEFT_PIN GPIO_PIN_4`
Przycisk Lewy (PA4/A3) - Poprzedni pulpit.
- `#define BTN_RIGHT_PIN GPIO_PIN_7`
Przycisk Prawy (PA7/A6) - Następny pulpit.
- `#define BTN_PORT GPIOA`
Port przycisków.

4.1.1 Opis szczegółowy

Mapowanie pinów i peryferiów.

4.2 Zmienne Interfejsu Użytkownika

Zmienne

- `uint8_t current_desktop = 0`
Aktualny pulpit (0=GPS, 1=Status, 2=Debug)
- `uint8_t screen_on = 1`
Stan ekranu (1=Włączony, 0=Wylączony)
- `uint32_t last_display_time = 0`
Czas ostatniego odświeżenia ekranu (ms)

4.2.1 Opis szczegółowy

Stan ekranu i nawigacji.

4.3 Zmienne Obsługi Przycisków

Zmienne

- uint32_t **btn_check_timer** = 0
Timer do próbkowania przycisków (debounce)
- uint8_t **btn_mid_last** = 1
Ostatni stan przycisku środkowego (1=puszczony)
- uint8_t **btn_left_last** = 1
Ostatni stan przycisku lewego.
- uint8_t **btn_right_last** = 1
Ostatni stan przycisku prawego.

4.3.1 Opis szczegółowy

Zmienne dla maszyny stanów przycisków i debouncingu.

4.4 Zmienne Modułu GSM

Zmienne

- volatile int **signal_strength_csq** = 99
Moc sygnału (99 = nieznaną)
- char **network_operator** [32] = "Szukam..."
Nazwa operatora sieci.
- volatile uint32_t **rx_byte_counter** = 0
Licznik odebranych bajtów.
- uint32_t **tx_msg_counter** = 0
Licznik wysłanych wiadomości MQTT.
- uint8_t **rx_char_buffer**
Pojedynczy znak z przerwania UART.
- char **current_line** [64]
Bufor roboczy do składania linii odpowiedzi.
- uint8_t **line_idx** = 0
Indeks w buforze linii.
- char **gsm_last_message** [64]
Ostatnia pełna linia (do debugowania)
- char **at_buffer** [512]
Bufor dla funkcji blokujących (GSM_SendAT_Checked)
- uint8_t **gsm_powered_on** = 0
Flaga stanu zasilania GSM (1=Włączony, 0=Wylączony)

4.4.1 Opis szczegółowy

Bufory i flagi komunikacji z modemem.

4.5 Słowniki GSM

Struktury danych

- struct [OperatorMapping](#)
Struktura mapująca kod operatora na nazwę.

Zmienne

- const [OperatorMapping](#) `op_list` []
Lista znanych operatorów w Polsce.

4.5.1 Opis szczegółowy

Mapowanie kodów na nazwy.

4.5.2 Dokumentacja zmiennych

4.5.2.1 `op_list`

```
const OperatorMapping op_list[]
```

Wartość początkowa:

```
= {  
    {"26001", "Plus"},  
    {"26002", "T-Mobile"},  
    {"26003", "Orange"},  
    {"26006", "Play"},  
    {"26012", "Cyfrowy Polsat"},  
    {"26017", "Aero2"},  
    {"26098", "Play (Roaming)"}  
}
```

Lista znanych operatorów w Polsce.

4.6 Zmienne Modułu GPS

Zmienne

- uint8_t **gps_rx_char**
Znak z przerwania GPS.
- char **gps_line_buffer** [128]
Bufor na jedną linię NMEA.
- uint8_t **gps_line_idx** = 0
Indeks bufora linii.
- volatile uint8_t **gps_data_ready** = 0
Flaga: nowa linia gotowa do parsowania.
- char **gps_main_buffer** [128]
Bufor bezpieczny dla pętli main.
- char **gps_debug_raw** [20] = "NO DATA"
Podgląd surowych danych dla ekranu DEBUG.
- float **gps_lat_decimal** = 0.0
Szerokość geograficzna (format dziesiętny)
- float **gps_lon_decimal** = 0.0
Długość geograficzna (format dziesiętny)
- char **gps_time_str** [16] = "00:00:00"
Czas UTC.
- char **gps_date_str** [16] = "01.01.2000"
Data.
- char **gps_sats** [4] = "0"
Liczba widocznych satelitów.
- uint8_t **gps_fix_valid** = 0
Status Fixa (1 = Jest Fix, 0 = Brak)
- float **gps_speed_kmh** = 0.0
Aktualna prędkość w km/h (uśredniona z kilku pomiarów)

4.6.1 Opis szczegółowy

Dane pozycyjne i bufor NMEA.

4.7 Logika Wysyłania

Zmienne

- uint32_t **last_mqtt_time** = 0
Czas ostatniej wysyłki MQTT.
- uint32_t **current_interval** = 30000
Aktualny interwał wysyłania (ms)
- char **current_mode** [10] = "INIT"
Aktualny tryb pracy (FAST/SLOW/STATIC/NO_FIX)
- uint32_t **last_movement_time** = 0
Zmienna do histerezy czasowej (opóźnienie wejścia w tryb STATIC)
- float **speed_buffer** [5] = {0.0}
Bufor kołowy do wygładzania prędkości GPS (średnia krocząca)
- uint8_t **speed_buf_idx** = 0

4.7.1 Opis szczegółowy

Zmienne sterujące zachowaniem lokalizatora.

4.8 Funkcje GSM

Funkcje

- `uint8_t GSM_SendAT_Checked` (char *cmd, char *expect, uint32_t timeout)
Wysyła komendę AT i czeka na oczekiwaną odpowiedź.
- `int GSM_CheckSignal_Blocking` (void)
Sprawdza siłę sygnału (AT+CSQ) w trybie blokującym.
- `void GSM_CheckSIM_Status` (char *status_buffer)
Sprawdza status karty SIM (AT+CPIN?).
- `int GSM_CheckNetwork_Reg` (void)
Sprawdza rejestrację w sieci (AT+CEREG?).
- `void GSM_GetOperator_Blocking` (void)
Pobiera kod operatora (AT+COPS?) i tłumaczy na nazwę.
- `void GSM_Init_Sequence` (void)
Główna sekwencja inicjalizacyjna modemu GSM.
- `void GSM_PublishMQTT_Async` (char *payload)
Publikuje wiadomość MQTT w sposób asynchroniczny.
- `void GSM_Watchdog_Check` (void)
Watchdog dla modemu GSM.
- `void GSM_Power_On` (void)
Bezpieczne włączenie zasilania modemu.
- `void GSM_Power_Off` (void)
Wyłączenie zasilania modemu (Sleep).

4.8.1 Opis szczegółowy

Obsługa modemu, komend AT i MQTT.

4.8.2 Dokumentacja funkcji

4.8.2.1 GSM_CheckNetwork_Reg()

```
int GSM_CheckNetwork_Reg (  
    void )
```

Sprawdza rejestrację w sieci (AT+CEREG?).

Zwraca

Status rejestracji: 1=Home, 5=Roaming, 0/2=Szukam/Brak.

4.8.2.2 GSM_CheckSignal_Blocking()

```
int GSM_CheckSignal_Blocking (
    void )
```

Sprawdza siłę sygnału (AT+CSQ) w trybie blokującym.

Zwraca

Wartość CSQ (0-31) lub 99 (błąd).

4.8.2.3 GSM_CheckSIM_Status()

```
void GSM_CheckSIM_Status (
    char * status_buffer )
```

Sprawdza status karty SIM (AT+CPIN?).

Parametry

<i>status_buffer</i>	Bufor wyjściowy na opis statusu (np. "READY", "PIN LOCK").
----------------------	--

4.8.2.4 GSM_GetOperator_Blocking()

```
void GSM_GetOperator_Blocking (
    void )
```

Pobiera kod operatora (AT+COPS?) i tłumaczy na nazwę.

Aktualizuje globalną zmienną `network_operator`.

4.8.2.5 GSM_Init_Sequence()

```
void GSM_Init_Sequence (
    void )
```

Główna sekwencja inicjalizacyjna modemu GSM.

1. Sprawdza komunikację (AT).
2. Ponawia zapytania w przypadku braku odpowiedzi ze strony modemu.
3. Konfiguruje parametry sieci (APN, GPRS).
4. Konfiguruje klienta MQTT i SSL.
5. Łączy się z brokerem.

4.8.2.6 GSM_Power_Off()

```
void GSM_Power_Off (
    void )
```

Wyłączenie zasilania modemu (Sleep).

Wysyła impuls na PWRKEY, aby uśpić modem. Zeruje flagę `gsm_powered_on`.

4.8.2.7 GSM_Power_On()

```
void GSM_Power_On (
    void )
```

Bezpieczne włączenie zasilania modemu.

Wywołuje Watchdoga, który sprawdza czy modem żyje. Jeśli nie żyje -> Watchdog wykona procedurę startową (impuls PWRKEY). Ustawia flagę `gsm_powered_on`.

4.8.2.8 GSM_PublishMQTT_Async()

```
void GSM_PublishMQTT_Async (
    char * payload )
```

Publikuje wiadomość MQTT w sposób asynchroniczny.

Parametry

<i>payload</i>	Treść wiadomości do wysłania.
----------------	-------------------------------

Wysyła sekwencję komend AT: TOPIC, PAYLOAD LENGTH, PAYLOAD, PUB. Nie czeka na odpowiedź (Fire and Forget), aby nie blokować pętli głównej.

4.8.2.9 GSM_SendAT_Checked()

```
uint8_t GSM_SendAT_Checked (
    char * cmd,
    char * expect,
    uint32_t timeout )
```

Wysyła komendę AT i czeka na oczekiwaną odpowiedź.

Parametry

<i>cmd</i>	Komenda do wysłania.
<i>expect</i>	Oczekiwany ciąg znaków (np. "OK").
<i>timeout</i>	Czas oczekiwania w ms.

Zwraca

1 - Sukces, 0 - Błąd/Timeout.

Nota

Funkcja blokująca. Czyści flagi błędów UART przed wysłaniem.

4.8.2.10 GSM_Watchdog_Check()

```
void GSM_Watchdog_Check (
    void )
```

Watchdog dla modemu GSM.

1. Zatrzymuje przerwania UART.
2. Wysyła "AT".
3. Jeśli brak "OK" -> Wykonuje twardy reset (PWRKEY) i reinicjalizację `GSM_Init_Sequence`.
4. Wznawia przerwania.

4.9 Funkcje GPS

Funkcje

- void [GPS_Parse_GPRMC](#) (char *nmea)
Parsuje ramkę NMEA GPRMC (Recommended Minimum Navigation Information).
- float [NMEA_To_Decimal](#) (char *nmea_coord, char direction)
Konwertuje współrzędne NMEA na format dziesiętny (Decimal Degrees).

4.9.1 Opis szczegółowy

Parsowanie NMEA i konwersja danych.

4.9.2 Dokumentacja funkcji**4.9.2.1 GPS_Parse_GPRMC()**

```
void GPS_Parse_GPRMC (
    char * nmea )
```

Parsuje ramkę NMEA GPRMC (Recommended Minimum Navigation Information).

Parametry

<i>nmea</i>	Wskaźnik do bufora z surową ramką.
-------------	------------------------------------

- Odporna na puste pola (np. ",") dzięki własnej pętli parsowania.
- Wyciąga Czas, Datę, Współrządne i Prędkość.
- Oblicza średnią kroczącą prędkości z 5 ostatnich pomiarów (filtr szumu).

4.9.2.2 NMEA_To_Decimal()

```
float NMEA_To_Decimal (
    char * nmea_coord,
    char direction )
```

Konwertuje współrządne NMEA na format dziesiętny (Decimal Degrees).

Parametry

<i>nmea_coord</i>	String ze współrządną w formacie NMEA (np. "5003.3430").
<i>direction</i>	Znak kierunku ('N', 'S', 'E', 'W').

Zwraca

float Współrządna w formacie dziesiętnym (np. 50.055716).

Nota

Obsługuje ujemne współrządne dla półkul S i W.

4.10 Funkcje Interfejsu

Funkcje

- void [Handle_Buttons](#) (void)
Obsługa przycisków fizycznych (Maszyna Stanów).

4.10.1 Opis szczegółowy

Obsługa przycisków i logiki ekranu.

4.10.2 Dokumentacja funkcji

4.10.2.1 Handle_Buttons()

```
void Handle_Buttons (
    void )
```

Obsługa przycisków fizycznych (Maszyna Stanów).

Funkcja nieblokująca. Wykonuje debouncing programowy oparty na czasie (50ms). Obsługuje 3 przyciski: Środkowy (On/Off ekranu), Lewy/Prawy (Zmiana pulpitów). Działa na pinach PA1, PA4, PA7 skonfigurowanych jako Pull-Up.

Rozdział 5

Dokumentacja struktur danych

5.1 Dokumentacja struktury OperatorMapping

Struktura mapująca kod operatora na nazwę.

Pola danych

- char * **code**
Kod numeryczny (np. "26001")
- char * **name**
Nazwa czytelna (np. "Plus")

5.1.1 Opis szczegółowy

Struktura mapująca kod operatora na nazwę.

Dokumentacja dla tej struktury została wygenerowana z pliku:

- [main.c](#)

Rozdział 6

Dokumentacja plików

6.1 Dokumentacja pliku main.c

: Główny plik programu dla projektu "Lokalizator GPS/GSM".

```
#include "main.h"
#include "ssd1306.h"
#include "ssd1306_fonts.h"
#include <stdio.h>
#include <string.h>
#include <stdarg.h>
#include <stdlib.h>
```

Struktury danych

- struct [OperatorMapping](#)
Struktura mapująca kod operatora na nazwę.

Definicje

- #define **GSM_UART** &[huart1](#)
Uchwyt UART dla modemu GSM (SIM7670E)
- #define **DEBUG_UART** &[huart2](#)
Uchwyt UART dla GPS (GPS_NEO-6M-0-001)
- #define **GSM_PWR_PIN** GPIO_PIN_1
Pin sterujący zasilaniem GSM (PWRKEY)
- #define **GSM_PWR_PORT** GPIOB
Port sterujący zasilaniem GSM.
- #define **BTN_MID_PIN** GPIO_PIN_1
Przycisk Środkowy (PA1/A1) - ON/OFF Ekran.
- #define **BTN_LEFT_PIN** GPIO_PIN_4
Przycisk Lewy (PA4/A3) - Poprzedni pulpit.
- #define **BTN_RIGHT_PIN** GPIO_PIN_7
Przycisk Prawy (PA7/A6) - Następny pulpit.
- #define **BTN_PORT** GPIOA
Port przycisków.

Funkcje

- void [SystemClock_Config](#) (void)
System Clock Configuration.
- uint8_t [GSM_SendAT_Checked](#) (char *cmd, char *expect, uint32_t timeout)
Wysła komendę AT i czeka na oczekiwaną odpowiedź.
- int [GSM_CheckSignal_Blocking](#) (void)
Sprawdza siłę sygnału (AT+CSQ) w trybie blokującym.
- void [GSM_CheckSIM_Status](#) (char *status_buffer)
Sprawdza status karty SIM (AT+CPIN?).
- int [GSM_CheckNetwork_Reg](#) (void)
Sprawdza rejestrację w sieci (AT+CREG?).
- void [GSM_GetOperator_Blocking](#) (void)
Pobiera kod operatora (AT+COPS?) i tłumaczy na nazwę.
- void [GSM_Init_Sequence](#) (void)
Główna sekwencja inicjalizacyjna modemu GSM.
- void [GSM_PublishMQTT_Async](#) (char *payload)
Publikuje wiadomość MQTT w sposób asynchroniczny.
- void [GSM_Watchdog_Check](#) (void)
Watchdog dla modemu GSM.
- void [GSM_Power_On](#) (void)
Bezpieczne włączenie zasilania modemu.
- void [GSM_Power_Off](#) (void)
Wyłączenie zasilania modemu (Sleep).
- void [GPS_Parse_GPRMC](#) (char *nmea)
Parсуje ramkę NMEA GPRMC (Recommended Minimum Navigation Information).
- float [NMEA_To_Decimal](#) (char *nmea_coord, char direction)
Konwertuje współrzędne NMEA na format dziesiętny (Decimal Degrees).
- void [Handle_Buttons](#) (void)
Obsługa przycisków fizycznych (Maszyna Stanów).
- void [HAL_UART_RxCpltCallback](#) (UART_HandleTypeDef *huart)
Callback przerwania odbioru UART.
- int [main](#) (void)
Główny punkt wejścia programu.
- void [HAL_UART_ErrorCallback](#) (UART_HandleTypeDef *huart)
- void [Error_Handler](#) (void)
This function is executed in case of error occurrence.

Zmienne

- I2C_HandleTypeDef **hi2c1**
Uchwyt do obsługi I2C (Ekran OLED)
- UART_HandleTypeDef **huart1**
Uchwyt do obsługi UART1 (GSM)
- UART_HandleTypeDef **huart2**
Uchwyt do obsługi UART2 (GPS)
- uint8_t **current_desktop** = 0
Aktualny pulpit (0=GPS, 1=Status, 2=Debug)
- uint8_t **screen_on** = 1
Stan ekranu (1=Włączony, 0=Wylłączony)

- `uint32_t last_display_time = 0`
Czas ostatniego odświeżenia ekranu (ms)
- `uint32_t btn_check_timer = 0`
Timer do próbkowania przycisków (debounce)
- `uint8_t btn_mid_last = 1`
Ostatni stan przycisku środkowego (1=puszczony)
- `uint8_t btn_left_last = 1`
Ostatni stan przycisku lewego.
- `uint8_t btn_right_last = 1`
Ostatni stan przycisku prawego.
- `volatile int signal_strength_csq = 99`
Moc sygnału (99 = nieznaną)
- `char network_operator [32] = "Szukam..."`
Nazwa operatora sieci.
- `volatile uint32_t rx_byte_counter = 0`
Licznik odebranych bajtów.
- `uint32_t tx_msg_counter = 0`
Licznik wysłanych wiadomości MQTT.
- `uint8_t rx_char_buffer`
Pojedynczy znak z przerwania UART.
- `char current_line [64]`
Bufor roboczy do składania linii odpowiedzi.
- `uint8_t line_idx = 0`
Indeks w buforze linii.
- `char gsm_last_message [64]`
Ostatnia pełna linia (do debugowania)
- `char at_buffer [512]`
Bufor dla funkcji blokujących (GSM_SendAT_Checked)
- `uint8_t gsm_powered_on = 0`
Flaga stanu zasilania GSM (1=Włączony, 0=Wylączony)
- `const OperatorMapping op_list []`
Lista znanych operatorów w Polsce.
- `uint8_t gps_rx_char`
Znak z przerwania GPS.
- `char gps_line_buffer [128]`
Bufor na jedną linię NMEA.
- `uint8_t gps_line_idx = 0`
Indeks bufora linii.
- `volatile uint8_t gps_data_ready = 0`
Flaga: nowa linia gotowa do parsowania.
- `char gps_main_buffer [128]`
Bufor bezpieczny dla pętli main.
- `char gps_debug_raw [20] = "NO DATA"`
Podgląd surowych danych dla ekranu DEBUG.
- `float gps_lat_decimal = 0.0`
Szerokość geograficzna (format dziesiętny)
- `float gps_lon_decimal = 0.0`
Długość geograficzna (format dziesiętny)
- `char gps_time_str [16] = "00:00:00"`
Czas UTC.
- `char gps_date_str [16] = "01.01.2000"`

Data.

- char **gps_sats** [4] = "0"
Liczba widocznych satelitów.
- uint8_t **gps_fix_valid** = 0
Status Fixa (1 = Jest Fix, 0 = Brak)
- float **gps_speed_kmh** = 0.0
Aktualna prędkość w km/h (uśredniona z kilku pomiarów)
- uint32_t **last_mqtt_time** = 0
Czas ostatniej wysyłki MQTT.
- uint32_t **current_interval** = 30000
Aktualny interwał wysyłania (ms)
- char **current_mode** [10] = "INIT"
Aktualny tryb pracy (FAST/SLOW/STATIC/NO_FIX)
- uint32_t **last_movement_time** = 0
Zmienna do histerezy czasowej (opóźnienie wejścia w tryb STATIC)
- float **speed_buffer** [5] = {0.0}
Bufor kołowy do wygładzania prędkości GPS (średnia krocząca)
- uint8_t **speed_buf_idx** = 0

6.1.1 Opis szczegółowy

: Główny plik programu dla projektu "Lokalizator GPS/GSM".

Autor

: Michał Figołuszka

Data

: 06.01.2026

Wersja

: 1.0

Ten plik zawiera główną pętlę programu oraz logikę sterowania lokalizatorem GPS. Główne funkcjonalności:

- Obsługa modułu GSM (SIM7670E) poprzez komendy AT (MQTT, SSL).
- Obsługa modułu GPS (GPS_NEO-6M-0-001) poprzez parsowanie ramek NMEA (GPRMC).
- Inteligentne zarządzanie energią (Smart Beacon) i interwałami wysyłania wiadomości.
- Interfejs użytkownika na ekranie OLED SSD1306 (3 pulpity, sterowane poprzez 3 przyciski).
- Watchdog programowy i sprzętowy reset modemu.

Uwaga

Copyright (c) 2026 STMicroelectronics. All rights reserved.

6.1.2 Dokumentacja funkcji

6.1.2.1 Error_Handler()

```
void Error_Handler (
    void )
```

This function is executed in case of error occurrence.

Zwracane wartości

<i>None</i>	
-------------	--

6.1.2.2 HAL_UART_RxCpltCallback()

```
void HAL_UART_RxCpltCallback (
    UART_HandleTypeDef * huart )
```

Callback przerwania odbioru UART.

Parametry

<i>huart</i>	Wskaźnik do struktury UART.
--------------	-----------------------------

Obsługuje odbiór znak po znaku dla GSM (UART1) i GPS (UART2). Dla GSM: Składa linie i ignoruje echo. Dla GPS: Wykrywa koniec linii () i ustawia flagę `gps_data_ready`.

6.1.2.3 main()

```
int main (
    void )
```

Główny punkt wejścia programu.

Zwracane wartości

<i>int</i>	
------------	--

6.1.2.4 SystemClock_Config()

```
void SystemClock_Config (
    void )
```

System Clock Configuration.

Zwracane wartości

<i>None</i>	
-------------	--

Configure the main internal regulator output voltage

Configure LSE Drive Capability

Initializes the RCC Oscillators according to the specified parameters in the `RCC_OscInitTypeDef` structure.

Initializes the RCC Oscillators according to the specified parameters in the `RCC_OscInitTypeDef` structure.

Initializes the CPU, AHB and APB buses clocks

Skorowidz

Definicje Sprzętowe, [7](#)

Error_Handler
main.c, [22](#)

Funkcje GPS, [14](#)
GPS_Parse_GPRMC, [14](#)
NMEA_To_Decimal, [15](#)

Funkcje GSM, [11](#)
GSM_CheckNetwork_Reg, [11](#)
GSM_CheckSignal_Blocking, [11](#)
GSM_CheckSIM_Status, [12](#)
GSM_GetOperator_Blocking, [12](#)
GSM_Init_Sequence, [12](#)
GSM_Power_Off, [12](#)
GSM_Power_On, [13](#)
GSM_PublishMQTT_Async, [13](#)
GSM_SendAT_Checked, [13](#)
GSM_Watchdog_Check, [14](#)

Funkcje Interfejsu, [15](#)
Handle_Buttons, [15](#)

GPS_Parse_GPRMC
Funkcje GPS, [14](#)
GSM_CheckNetwork_Reg
Funkcje GSM, [11](#)
GSM_CheckSignal_Blocking
Funkcje GSM, [11](#)
GSM_CheckSIM_Status
Funkcje GSM, [12](#)
GSM_GetOperator_Blocking
Funkcje GSM, [12](#)
GSM_Init_Sequence
Funkcje GSM, [12](#)
GSM_Power_Off
Funkcje GSM, [12](#)
GSM_Power_On
Funkcje GSM, [13](#)
GSM_PublishMQTT_Async
Funkcje GSM, [13](#)
GSM_SendAT_Checked
Funkcje GSM, [13](#)
GSM_Watchdog_Check
Funkcje GSM, [14](#)

HAL_UART_RxCpltCallback
main.c, [23](#)

Handle_Buttons
Funkcje Interfejsu, [15](#)

Logika Wysyłania, [10](#)

main
main.c, [23](#)
main.c, [19](#)
Error_Handler, [22](#)
HAL_UART_RxCpltCallback, [23](#)
main, [23](#)
SystemClock_Config, [23](#)

NMEA_To_Decimal
Funkcje GPS, [15](#)

op_list
Słowniki GSM, [9](#)
OperatorMapping, [17](#)

SystemClock_Config
main.c, [23](#)
Słowniki GSM, [9](#)
op_list, [9](#)

Zmienne Interfejsu Użytkownika, [7](#)
Zmienne Modułu GPS, [10](#)
Zmienne Modułu GSM, [8](#)
Zmienne Obsługi Przycisków, [8](#)