

Urządzenia peryferyjne

Prowadzący zajęcia: dr inż. Jan Nikodem

Termin zajęć: Czwartek TN, godz. 14:10

Osoby wykonujące ćwiczenie:

Mateusz Gawłowski, Bartosz Szymański

Oznaczenie grupy:

B

Tytuł ćwiczenia:

SYSTEMY NAWIGACJI SATELITARNEJ (GPS)

Data wykonania ćwiczenia:

09.11.2023

Spis treści

1. Cel ćwiczenia	2
2. Wstęp teoretyczny	2
2.1. Systemy nawigacji satelitarnej, struktura i zasady działania.....	2
2.2. Budowa i zasada działania systemu GPS	3
2.3. Istotne pojęcia geograficzne	4
2.4 Struktura zdań NMEA.....	5
3. Użyte oprogramowanie	6
4. Rozwiązania zadań wraz z krótkim opracowaniem	7
4.1 Odczyt oraz podział zdań NMEA.....	7
4.2 Program do obsługi transmisji GPS	8
Opracowanie.....	12
5. Wnioski	12
6. Literatura	12

1. Cel ćwiczenia

- Zapoznać się z zestawem GPS oraz podłączyć via Bluetooth
 - W ramach testu uruchomić transmisję z urządzenia GPS przy użyciu wybranego terminala (np. Hyper Terminal, Putty)
 - Odczytać uzyskane zdania NMEA, podzielić je wg. Typów. Dla ustalonych z prowadzącym zdań NMEA, sprawdzić ważność uzyskanych danych i omówić wyniki.
- Napisać program w dowolnym (Windows, Android) środowisku, który będzie obsługiwał transmisję z urządzenia GPS i pozwoli na czytelne (rozkodowane) przedstawienie uzyskanych danych oraz zlokalizuje na mapie świata (np. Google Maps) punkt, w którym znajduje się urządzenie.

2. Wstęp teoretyczny

2.1. Systemy nawigacji satelitarnej, struktura i zasady działania

Systemy nawigacji satelitarnej to globalne infrastruktury składające się z zestawu sztucznych satelitów umieszczonych na orbicie wokół Ziemi, które umożliwiają dokładne określenie położenia i czasu na powierzchni ziemi. Każdy z wymienionych systemów ma własną strukturę i zasadę działania, ale ich ogólna idea polega na wykorzystaniu sygnałów emitowanych przez satelity do określania pozycji i nawigacji użytkowników na Ziemi.

- **GLONASS (Globalnaja Nawigacyonaja Sputnikowaja Sistema) [Rosja]:**
 - Struktura: GLONASS składa się z kilkudziesięciu satelitów umieszczonych na trzech orbitach okołoziemskich.
 - Zasada działania: GLONASS działa na podobnej zasadzie co amerykański system GPS. Sygnały z satelitów są odbierane przez urządzenia nawigacyjne na Ziemi, które obliczają swoją pozycję na podstawie trójwymiarowego sygnału.
- **Galileo [Unia Europejska]:**
 - Struktura: System Galileo składa się z konstelacji około 30 satelitów umieszczonych na orbitach średniej wysokości.
 - Zasada działania: Galileo jest niezależnym systemem nawigacji satelitarnej Unii Europejskiej. Sygnały z satelitów pozwalają na dokładne określenie pozycji użytkowników na Ziemi, a także oferują usługi w zakresie czasu i synchronizacji.
- **BeiDou (BDS) [Chiny]:**
 - Struktura: BeiDou składa się z dwóch rodzajów satelitów - nawigacyjnych i komunikacyjnych, umieszczonych na różnych orbitach.
 - Zasada działania: System BeiDou oferuje globalną nawigację i lokalne usługi komunikacyjne. Sygnały z satelitów BeiDou pozwalają na określenie pozycji oraz oferują usługi czasu i synchronizacji.
- **QZSS (Quasi-Zenith Satellite System) [Japonia]:**
 - Struktura: QZSS składa się z kilku satelitów umieszczonych na orbitach wokół Ziemi, w tym jednego satelity Zenitowego (QZS).
 - Zasada działania: System QZSS ma na celu zwiększenie dokładności nawigacji w obszarach miejskich i górskich w Japonii. Satelita

Zenitowy jest umieszczany na szczycie swojej orbity, co zwiększa czas widoczności w regionie.

- **NavIC (Navigation with Indian Constellation) [Indie]:**
 - Struktura: NavIC to system nawigacji satelitarnej Indie, który składa się z kilku satelitów umieszczonych na różnych orbitach.
 - Zasada działania: NavIC oferuje usługi nawigacyjne i czasowe na obszarze Indii i okolicznych regionów. Sygnały z satelitów pozwalają na dokładne określenie pozycji i czasu na powierzchni ziemi.

2.2. Budowa i zasada działania systemu GPS

Budowa i zasada działania systemu GPS (Global Positioning System) można opisać w kilku głównych krokach:

- **Satelity GPS:**
 - System GPS składa się z co najmniej 24 satelitów umieszczonych na orbitach okołoziemskich.
 - Te satelity są rozmieszczone w sposób, który pozwala na równomierne pokrycie całej powierzchni Ziemi.
- **Zegary atomowe:**
 - Każdy satelita GPS jest wyposażony w bardzo dokładny zegar atomowy.
 - Zegary te generują sygnały o stałej częstotliwości, co jest niezbędne do precyzyjnego pomiaru czasu.
- **Sygnały GPS:**
 - Każdy satelita GPS emituje sygnały radioelektroniczne, które zawierają informacje o czasie oraz identyfikatorze satelity.
 - Te sygnały są rozprzestrzeniane w przestrzeni.
- **Odbiorniki GPS:**
 - Odbiorniki GPS to urządzenia dostępne dla użytkowników na Ziemi.
 - Odbiorniki te odbierają sygnały od co najmniej trzech satelitów GPS w danej chwili.
- **Zasada działania triangulacji:**
 - Odbiornik GPS odbiera sygnały od satelitów i dokładnie mierzy czas, który upłynął od momentu wysłania sygnału przez satelitę.
 - Na podstawie różnicy czasu między wysłaniem sygnału a jego odbiorem od kilku satelitów, odbiornik oblicza odległość do każdego satelity.
- **Wyliczanie pozycji:**
 - Korzystając z odległości do co najmniej trzech satelitów, odbiornik GPS może wykonać obliczenia w celu określenia swojej dokładnej pozycji na powierzchni ziemi.
 - Dla jeszcze większej dokładności, odbiornik może korzystać z sygnałów od więcej niż trzech satelitów.
- **Korekta zegara:**
 - W trakcie działania systemu GPS, odbiornik musi uwzględniać korekty zegara satelitów, które mogą ulegać zmianom ze względu na efekty

związane z ogólną teorią względności Einsteina oraz wpływem grawitacji satelitów.

Ostatecznie, dzięki tym obliczeniom, odbiornik GPS jest w stanie określić swoją pozycję z bardzo dużą dokładnością, często na poziomie kilku metrów. System GPS jest szeroko wykorzystywany w nawigacji, badaniach geodezyjnych, lotnictwie, transporcie, telekomunikacji i wielu innych dziedzinach. Warto zauważyć, że system GPS jest utrzymywany i nadzorowany przez rząd Stanów Zjednoczonych, co zapewnia jego ciągłą dostępność i precyzję.

2.3. Istotne pojęcia geograficzne

- **UTC (Coordinated Universal Time):**
 - UTC jest standaryzowanym czasem, który jest używany jako punkt odniesienia do określania czasu na całym świecie. Jest to międzynarodowy czas atomowy, oparty na międzynarodowym czasie atomowym (TAI) z korektą dodaną, aby utrzymać zgodność z rotacją Ziemi. Jest to wspólny punkt odniesienia dla wszystkich regionów i czasów na świecie, co ułatwia synchronizację różnych systemów czasu.
- **Odległość satelitów od Ziemi (MEO, LEO) i ich ilość:**
 - MEO (Medium Earth Orbit): Satelity w MEO znajdują się na średniej orbicie okołoziemskiej, zazwyczaj w odległości około 10 000-20 000 kilometrów od powierzchni Ziemi.
 - LEO (Low Earth Orbit): Satelity w LEO okrążają Ziemię na niskiej orbicie, zazwyczaj w odległości od 160 do 2 000 kilometrów nad powierzchnią Ziemi.
 - Liczenie odległości satelitów od Ziemi może obejmować zarówno odległość od powierzchni Ziemi, jak i od jej środka, w zależności od kontekstu i potrzeb pomiaru. Zazwyczaj odległość od powierzchni jest używana w przypadku określania parametrów transmisji, natomiast odległość od środka Ziemi jest używana w naukowych obliczeniach orbitalnych.
- **Ilość satelitów na orbicie i ich prędkość obrotowa:**
 - Liczba satelitów na orbicie może być różna w zależności od konstelacji satelitarnej. Na przykład system GPS posiada ponad 30 satelitów w różnych płaszczyznach orbitalnych. Prędkość obrotowa satelitów zależy od ich wysokości i orbity. Satelity w LEO mogą poruszać się z prędkościami od kilkunastu do kilkudziesięciu tysięcy kilometrów na godzinę, podczas gdy satelity w MEO poruszają się z mniejszymi prędkościami.
- **Układ wysokościowy Kronsztad i wysokość nad poziomem morza w Polsce:**
 - Układ wysokościowy Kronsztad jest jednym z wielu systemów referencyjnych wykorzystywanych do określania wysokości nad poziomem morza. W Polsce wysokość nad poziomem morza jest obliczana na podstawie tego układu, co umożliwia dokładne określenie wysokości terenu względem poziomu morza.
- **Elipsoida a geoida:**
 - Elipsoida: W geodezji i kartografii elipsoida jest modelem matematycznym opisującym kształt Ziemi jako elipsę obrotową. Jest to uproszczony model, który pozwala na dokładne określenie kształtu Ziemi do celów pomiarowych i kartograficznych.
 - Geoida: Geoida jest modelem opisującym równopotencjalną powierzchnię grawitacyjną Ziemi, czyli powierzchnię, na której siła grawitacji jest taka sama we wszystkich punktach. Jest to "idealna" powierzchnia referencyjna dla pomiarów wysokości nad poziomem morza, ponieważ odzwierciedla naturalne nieregularności grawitacyjne Ziemi.

2.4 Struktura zdań NMEA

- **Zdanie RMC (Recommended Minimum Navigation Information):**

- Zdanie RMC zawiera podstawowe informacje nawigacyjne, takie jak pozycja, prędkość, kierunek i czas.
- Przykład:

mathematica Copy code

```
$GPRMC,123519.00,A,4807.038,N,01131.000,E,022.4,084.4,230394,003.1,W*6A
```

- ``$GPRMC`` to identyfikator zdania RMC.
- ``123519.00`` to czas w formacie godzina:minuta:sekunda.milisekunda UTC.
- ``A`` oznacza status aktywności (A - aktywne, V - nieaktywne).
- ``4807.038`` to szerokość geograficzna w formacie stopnie i minuty.
- ``N`` to wschodnia (N - półkula północna, S - półkula południowa).
- ``01131.000`` to długość geograficzna w formacie stopnie i minuty.
- ``E`` to wschodnia (E - wschodnia półkula, W - zachodnia półkula).
- ``022.4`` to prędkość w węzłach.
- ``084.4`` to kierunek kursu w stopniach.
- ``230394`` to data w formacie dzień/miesiąc/rok.
- ``003.1`` to wariacja magnetyczna.
- ``W`` oznacza kierunek wariacji (W - zachód, E - wschód).
- ``*6A`` to suma kontrolna.

- **Zdanie GGA (Global Positioning System Fix Data):**

- Zdanie GGA dostarcza informacje o pozycji geograficznej, dokładności pomiaru i czasie.
- Przykład:

mathematica Copy code

```
$GPGGA,123519.00,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47
```

- ``$GPGGA`` to identyfikator zdania GGA.
- ``123519.00`` to czas w formacie godzina:minuta:sekunda.milisekunda UTC.
- ``4807.038`` to szerokość geograficzna w formacie stopnie i minuty.
- ``N`` to wschodnia (N - półkula północna, S - półkula południowa).
- ``01131.000`` to długość geograficzna w formacie stopnie i minuty.
- ``E`` to wschodnia (E - wschodnia półkula, W - zachodnia półkula).
- ``1`` to jakość fixa (0 - brak, 1 - fix jednopunktowy, 2 - fix DGPS, 6 - fix RTK).
- ``08`` to liczba używanych satelitów w fixie.
- ``0.9`` to dokładność pozioma.
- ``545.4`` to wysokość nad poziomem morza w metrach.
- ``M`` oznacza jednostkę wysokości (M - metry).
- ``46.9`` to różnica między wysokością geoidy a elipsoidy (geoidal separation).
- ``M`` oznacza jednostkę różnicy między wysokością geoidy a elipsoidy (M - metry).
- ``*47`` to suma kontrolna.

- **Zdanie GLL (Geographic Position - Latitude/Longitude):**

- Zdanie GLL przekazuje informacje o aktualnej pozycji w formacie szerokości i długości geograficznej.
- Przykład:

```
bash Copy code  
  
$GPGLL,4916.45,N,12311.12,W,225444,A,*33  
  
• ` $GPGLL ` to identyfikator zdania GLL.  
• ` 4916.45 ` to szerokość geograficzna w formacie stopnie i minuty.  
• ` N ` to półkula północna.  
• ` 12311.12 ` to długość geograficzna w formacie stopnie i minuty.  
• ` W ` to półkula zachodnia.  
• ` 225444 ` to czas w formacie godzina:minuta:sekunda UTC.  
• ` A ` oznacza status aktywności (A - aktywne, V - nieaktywne).  
• ` *33 ` to suma kontrolna.
```

- **Zdanie ZDA (Time and Date):**

- Zdanie ZDA przekazuje informacje o aktualnym czasie.
- Przykład:

```
bash Copy code  
  
$GPZDA,160012.71,11,03,2004,-1,00*7D  
  
• ` $GPZDA ` to identyfikator zdania ZDA.  
• ` 160012.71 ` to czas w formacie godzina:minuta:sekunda.milisekunda UTC.  
• ` 11,03,2004 ` to data w formacie dzień,miesiąc,rok.  
• ` -1 ` to różnica czasu UTC (czas letni/zimowy).  
• ` 00 ` to numer strefy czasowej (00 - UTC).  
• ` *7D ` to suma kontrolna.
```

3. Użyte oprogramowanie

- Putty Terminal
- CLion

4. Rozwiązania zadań wraz z krótkim opracowaniem

4.1 Odczyt oraz podział zdań NMEA

Przebieg zadania można ująć w 8 krokach:

1. Połączenie z urządzeniem GPS z komputerem przy użyciu interfejsu Bluetooth
2. Otwarcie terminala Putty:
3. Wybór ustawień połączenia:
 - W Putty wybrano opcję "Serial" (szeregowy) jako rodzaj połączenia.
4. Konfiguracja portu szeregowego:
 - Wybrano port COM8. (W razie problemów należy sprawdzić dostępne porty w menedżerze urządzeń lub podobnej aplikacji, aby znaleźć prawidłowy, jeśli nie wiadomo, który jest używany przez urządzenie GPS)
 - Ustawiono prędkość transmisji na 9600 bps
5. Rozpoczęcie połączenia:
 - Naciśnięto przycisk "Open", w celu nawiązania połączenia z urządzeniem GPS.
6. Odczytanie danych:
 - Po nawiązaniu połączenia w terminalu Putty, dane z urządzenia GPS będą wyświetlane w oknie terminala.
7. Analiza danych:
 - Wyświetlone w terminalu dane zostały odczytane i zinterpretowane na podstawie objaśnień zawartych w punkcie 2.4 *Struktura zdań NMEA*
8. Zakończenie połączenia:
 - Po zakończeniu pracy z urządzeniem GPS, zakończono połączenie w terminalu Putty, naciskając przycisk "Close".

Przykładowy odczytany fragment:

```
$GPGSV,4,3,13,23,11,264,18,14,08,064,21,32,06,325,,25,05,244,*7B
$GPGSV,4,4,13,43,,, *7C
$GPRMC,155837.000,A,5106.5360,N,01703.6376,E,0.07,190.60,250304,,,A*69
$GPGGA,155838.000,5106.5360,N,01703.6375,E,1,9,0.92,154.4,M,42.6,M,,*58
$GPRMC,155838.000,A,5106.5360,N,01703.6375,E,0.07,190.60,250304,,,A*65
$GPGGA,155839.000,5106.5359,N,01703.6374,E,1,9,0.92,154.4,M,42.6,M,,*52
$GPRMC,155839.000,A,5106.5359,N,01703.6374,E,0.07,190.60,250304,,,A*6F
$GPGGA,155840.000,5106.5358,N,01703.6374,E,1,9,0.92,154.4,M,42.6,M,,*5D
$GPGSA,A,3,12,13,19,17,10,22,23,24,14,,,,,1.21,0.92,0.78*08
$GPGSV,4,1,13,24,80,279,23,19,49,094,32,12,41,239,16,15,37,201,*7E
$GPGSV,4,2,13,17,37,058,35,22,27,061,19,13,19,163,18,10,16,300,25*7B
$GPGSV,4,3,13,23,11,264,18,14,08,064,21,32,06,325,,25,05,244,*7B
$GPGSV,4,4,13,45,,, *7A
$GPRMC,155840.000,A,5106.5358,N,01703.6374,E,0.07,190.60,250304,,,A*60
$GPGGA,155841.000,5106.5358,N,01703.6373,E,1,9,0.92,154.4,M,42.6,M,,*5B
$GPRMC,155841.000,A,5106.5358,N,01703.6373,E,0.07,190.60,250304,,,A*66
$GPGGA,155842.000,5106.5356,N,01703.6374,E,1,9,0.92,154.4,M,42.6,M,,*51
$GPRMC,155842.000,A,5106.5356,N,01703.6374,E,0.07,190.60,250304,,,A*6C
$GPGGA,155843.000,5106.5355,N,01703.6374,E,1,9,0.92,154.4,M,42.6,M,,*53
$GPGSA,A,3,12,13,19,17,10,22,23,24,14,,,,,1.21,0.92,0.78*08
$GPGSV,4,1,13,24,80,279,24,19,49,094,32,12,41,239,16,15,37,201,*79
$GPGSV,4,2,13,17,37,058,35,22,27,062,19,13,19,163,20,10,16,300,24*72
$GPGSV,4,3,13,23,11,264,18,14,08,064,19,32,06,325,,25,05,244,*70
$GPGSV,4,4,13,49,,, *7
```

4.2 Program do obsługi transmisji GPS

```
#include <iostream>
#include <windows.h>
#include <sstream>
#include "Serial.h"
//#include "stdafx.h"
#include <tchar.h>
#include <fstream>

using namespace std;

#define RX_BUFFSIZE 300 // buffer size definition

string doubleToString(double i);

void printChars(char *c, int len, string message);

bool switch_function() {
    char option;

    cout << "\n==== MAIN MENU =====" << endl;
    cout << "1. Load data from .txt file" << endl;
    cout << "2. Load data from chosen Serial port" << endl;
    cout << "0. Exit" << endl;
    cout << "Choose option: >> ";
    cin >> option;
    cout << endl;
do {
    switch (option) {
        case '1':
            return true;

        case '2':
            return false;
    }
} while (option != '0');
}

int _tmain(int argc, _TCHAR *argv[]) {

    bool run_type = switch_function();

    try {
        if (!run_type) {
            // Get information about Port
            string port;

            cout << "Enter the COM port of the GPS device [ex. COM7]: ";
            cin >> port;
            cout << "Opening selected port..." << endl;
            port += ":";

            TCHAR *portCom = new TCHAR[port.size() + 1];
            portCom[port.size()] = 0;
            copy(port.begin(), port.end(), portCom);

            // Port configure
            tstring commPortName(portCom);
            // Connect with selected port
            Serial serial(commPortName);
```



```

        cout << "Port opened successfully" << endl;
        cout << "Processing..." << endl;

        char array[RX_BUFFSIZE]; // buffer with GPS data

        // GPS data read
        int b = -1;
        int charsRead;
        do {
            b++;
            charsRead = NULL;

            for (int k = 0; k < 100; k++) array[k] = NULL;
            charsRead = serial.read(array, RX_BUFFSIZE);

            Sleep(1000);
        } while (b < 10);
    } else {
        char array[1000];

        std::ifstream file("up_lab3.txt"); // choose file name in need
        cout << "File loaded successfully" << endl;
        cout << "Processing..." << endl;

        if (!file.is_open()) {
            std::cout << "File can not be read" << std::endl;
            return 1;
        }

        // GPS data read
        int b = -1;
        int charsRead;
        do {
            b++;
            charsRead = NULL;

            for (int k = 0; k < 100; k++) array[k] = NULL;

            Sleep(1000);
        } while (b < 10);

        if (file.is_open()) { // file check
            int i = 0;
            char symbol;

            // Odczytanie symboli z pliku tekstowego i zapisanie do tablicy tab
            while (file.get(symbol) && i < 1000) {
                array[i] = symbol;
                i++;
            }
            file.close();
        }
    }
    string time, latitude, longitude; // NMEA data variables

    char GGA[72];

    // search $GPGGA sequence - Global Positioning System Fix Data and get
    its data
    for (int i = 0; i < 1000; i++) {

```

```

        if (array[i] == '$' && array[i + 1] == 'G' && array[i + 2] == 'P'
&& array[i + 3] == 'G' &&
        array[i + 4] == 'G' && array[i + 5] == 'A') {
            cout << "\nGGA found at position: " << i << endl;
            int index = i;

            for (char &g: GGA) {
                g = array[index];
                index++;
                cout << g;
            }
index = 0;

            for (int j = 0; j < 72; j++) {

                if (j == 7) {
                    for (int k = 0; k < 6; ++k) {
                        time += GGA[j + k];
                    }
                }

                if (j == 18) {
                    for (int k = 0; k < 9; ++k) {
                        latitude += GGA[j + k];
                    }
                }

                if (j == 30) {
                    for (int k = 0; k < 9; ++k) {
                        longitude += GGA[j + k];
                    }
                }
            }
            break;
        }
    }

//      EXAMPLES
//      https://www.google.com/maps/place/51°06'32.2"N+17°03'38.2"E/
//
https://www.google.com/maps/place/51%C2%B006'32.2%22N+17%C2%B003'38.2%22E/
//
$GPGGA,155841.000,5106.5358,N,01703.6373,E,1,9,0.92,154.4,M,42.6,M,,*5B

    string google = "https://www.google.pl/maps/place/";

    size_t index = latitude.find_first_not_of('0');

    if (index != std::string::npos) {
        latitude = latitude.substr(index);
    } else {
        latitude = "0";
    }

    index = longitude.find_first_not_of('0');

    if (index != std::string::npos) {
        longitude = longitude.substr(index);
    } else {
        longitude = "0";
    }
}

```

```

        cout << "\nTime: " << time;
        cout << "\nLatitude: " << latitude;
        cout << "\nLongitude: " << longitude;
index = latitude.find('.') - 2;

        for (int j = 0; j < latitude.size(); ++j) {
            if (j == index) {
                google += " ";
                google += latitude[j];
            } else {
                google += latitude[j];
            }
        }

        google += ",";

        index = longitude.find('.') - 2;

        for (int j = 0; j < longitude.size(); ++j) {
            if (j == index) {
                google += " ";
                google += longitude[j];
            } else {
                google += longitude[j];
            }
        }

        google += '/';

        cout << "\nGoogle Maps (please copy and paste following line):
" << google << endl << endl;
    }

    fseek(stdin, 0, SEEK_END);
    getchar();

    return 0;
} catch (const char *error) {
    cout << error << endl;
}
}

```

Opracowanie

Program przeznaczony jest do odczytywania danych z urządzenia GPS. Oto ogólny opis jego działania:

- Wybór źródła danych GPS:
 - Program zaczyna od wyboru, czy dane mają zostać odczytane z pliku tekstowego (.txt) lub z wybranego portu szeregowego, który jest połączony z urządzeniem GPS.
- Połączenie z portem szeregowym lub odczyt z pliku:
 - Jeśli wybór padnie na port szeregowy, użytkownik podaje numer portu COM, a program próbuje się połączyć z wybranym urządzeniem GPS.
 - Jeśli wybór padnie na odczyt z pliku tekstowego, program otwiera wybrany plik z danymi.
- Odczyt danych GPS:
 - Program odczytuje dane GPS (NMEA) z urządzenia lub pliku tekstowego i przechowuje je w buforze.
 - Szuka linii danych NMEA typu GGA (Global Positioning System Fix Data), które zawierają informacje o czasie, szerokości i długości geograficznej.
- Analiza danych:
 - Gdy linia GGA zostanie znaleziona, program pobiera z niej informacje o czasie, szerokości i długości geograficznej.
 - Wybiera znaki z danej linii, które odpowiadają za te informacje i zapisuje je w zmiennych.
- Tworzenie linku do Google Maps:
 - Na podstawie odczytanych danych o szerokości i długości geograficznej, program tworzy link, który można skopiować i wkleić do Google Maps. Link ten otwiera mapę z zaznaczonym konkretnym miejscem odczytanym z urządzenia GPS.
- Zakończenie programu:
 - Program kończy swoje działanie po wyświetleniu danych i linku do Google Maps.

5. Wnioski

Realizacja ćwiczenia w znaczący sposób wpłynęła na nasz poziom rozumienia tematu w obszarze sposobu działania systemu GPS oraz jego komunikacji z urządzeniami. Na laboratoriach niestety nie udało nam się zaimplementować w pełni działającego programu do odczytu i analizy sygnału GPS, jednakże na podstawie pliku tekstowego z zapisanymi odczytami transmisji z terminalu Putty został on przez nas dopracowany i wydaje się spełniać wymagane założenia.

6. Literatura

Poprzez literaturę należy rozumieć wszelkie materiały potrzebne do zrozumienia założeń laboratoriów, sposobu realizacji ćwiczenia oraz źródeł pomocy podczas rozwiązywania problemów napotkanych w trakcie realizacji zajęć.

- Instrukcja obsługi urządzenia GPS (Nokia):
https://www.cs.put.poznan.pl/wswitala/download/pdf/Nokia_LD-3W_UG_en.pdf
- Odczyt sygnału GPS w terminalu Putty
https://www.youtube.com/watch?v=qJvIGIsToBE&ab_channel=TechnicalTransistor