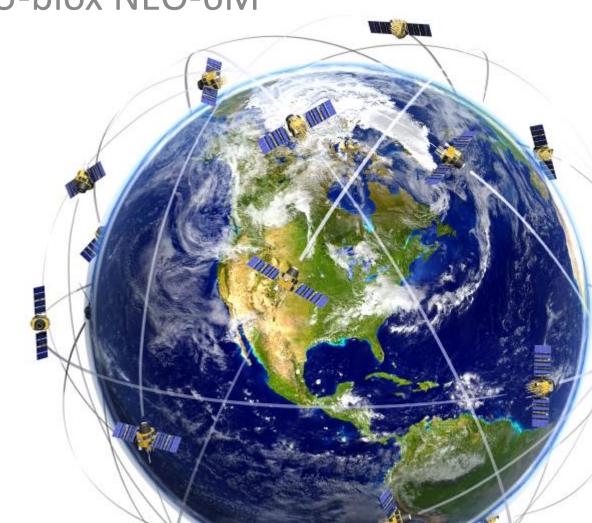
Odbiornik GPS + prezentacja danych

U-blox NEO-6M





STM32f4-discovery – wykorzystane biblioteki

- STM32DiscoveryVCP biblioteki pozwalające wykorzystać stma jako wirtualny port COM przez kabel microUSB
- Standardowe biblioteki Coocoxa, głównie stm32f4xx_usart do odbierania danych przesyłanych przez GPS
- Odebranie danych z GPSa i przesłanie ich do portu COM

```
void USART1_IRQHandler(void) {
    // check if the USART1 receive interrupt flag was set
    if( USART_GetITStatus(USART1, USART_IT_RXNE) ) {
        char t = USART1->DR; // the character from the USART1 data register is saved in t
        VCP_put_char(t); // send character to the COM port

        // store some data
        if( (t != '\n') && (cnt < MAX_STRLEN) ) {
            received_string[cnt] = t;

            cnt++;
        }
}</pre>
```

Visual Studio C# forms

- Odbieranie danych przy uzyciu biblioteki System.IO.Ports, wykorzystując funkcje Port.ReadLine(); (komunikaty wysyłane przez GPS są odzielone znakiem nowej linii \n)
- Dane odebrane z nadajnika przechowywane są w klasie NMEAData, w której znajdują się również funkcje do przetwarzania otrzymanych komunikatów.
- Dane odbierane są na bieżąco przez wątek działający w tle, naciśnięcie przycisku update aktualizuje wyświetlane informację, oraz mapę.

Parser NMEA

- Różne typy komunikatów m.in. GGA, GLL, GSA, GSV, RMC
- Każdy komunikat zaczyna się od znaku \$, i kończy 2 cyfrową sumą kontrolną poprzedzoną znakiem *

- Komunikaty przekazują informację m.in. o
 - Szerokości i Długości geograficznej
 - Ilość śledzonych i widocznych satelit
 - Wysokość geometryczną
 - Informacje o dokładności pomiaru (HDOP, PDOP, VDOP)

Parser NMEA

```
public void parse(string NMEA)
            if (!IsValid(NMEA)) return; //Checksum
            string [] split = NMEA.Split(new Char [] {','});
//Split data
            bool test = false;
            while (test == false)
                test = ParseGPRMC(split);
                test = ParseGPGGA(split);
                test = ParseGPGLL(split);
                test = ParseGPGSA(split);
                test = ParseGPGSV(split);
                test = ParseGPRMC(split);
                break;
```

Komunikat RMC Recommended Minimum sentence C

Przykładowy komunikat:

\$GPRMC,123519,A,4807.038,N,01131.000,E,022.4,084.4,230394,003.1,W*6A

Przekazuje m.in. informacje o:

- Długości i szerokości geograficznej
- Czas pomiaru
- Prędkość

Komunikat GGA Global Positioning System Fix Data

Przykładowy komunikat \$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47

Przekazuje informacje o:

- Długości i szerokości geograficznej
- Wpływ geometrii konstelacji satelitów na wyznaczenie pozycji płaskich HDOP
- Liczbę śledzonych satelit
- Wysokość geograficzną na której odbywa się pomiar
- Aktualny czas

Komunikat GSA Satellite status

Przykładowy komunikat:

\$GPGSA,A,3,04,05,,09,12,,,24,,,,,2.5,1.3,2.1*39

Przekazuje informacje o:

Jakości połączenia wynikającą z geometrii konstelacji satelitów

| DOP | Opis |
|--------|-------------|
| 1 | Idealny |
| 2 – 3 | Znakomity |
| 4 – 6 | Dobry |
| 7 – 8 | Umiarkowany |
| 9 – 20 | Słaby |
| > 20 | Zły |

Komunikat GLL Geographic Latitude and Longitude

Przykładowy komunikat:

\$GPGLL,4916.45,N,12311.12,W,225444,A,*1D

Przekazuje informacje o:

- Szerokości geograficznej
- Długości geograficznej
- Czas odebrania informacji

KOMUNIKAT GSV Satellites in view

Przykładowy komunikat \$GPGSV,2,1,08,01,40,083,46,02,17,308,41,12,07,344,39,14,22,228,45*75

Przekazuje m.in. informacje o:

- Liczbie satelitów w zasięgu
- Wysokości geometrycznej

Komunikat VTG Track made good and ground speed

Przykładowy komunikat:

\$GPVTG,054.7,T,034.4,M,005.5,N,010.2,K*48

Przekazuje informacje m.in. o prędkości i kierunku poruszania się.