

PROJEKT

STEROWNIKI ROBOTÓW

Dokumentacja

Antena śledząca do sondy stratosferycznej ANSTRA

Skład grupy:

Adam ŚLIWKA, 250137

Michał WIŚNIEWSKI, 247647

Termin: poniedziałek 18:55 TP

Prowadzący:

dr inż. Wojciech DOMSKI

19 czerwca 2022

Spis treści

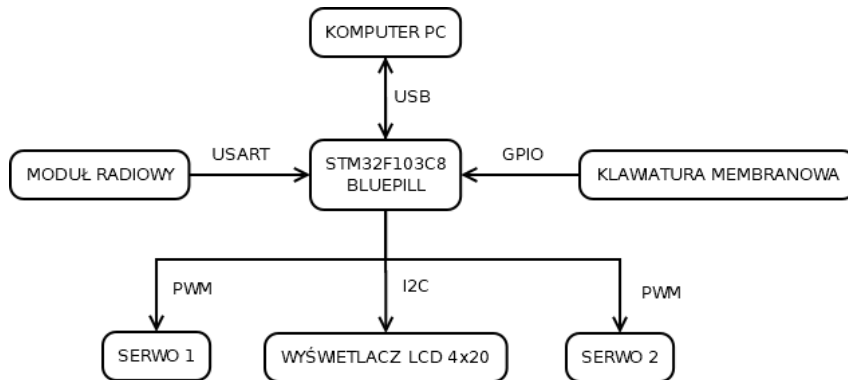
1	Opis projektu	2
2	Konfiguracja mikrokontrolera	3
2.1	Konfiguracja pinów	3
2.2	I2C	4
2.3	RCC	4
2.4	USART3	6
2.5	USB	6
2.6	TIM3	6
3	Urządzenia zewnętrzne	7
3.1	Klawiatura membranowa	7
3.2	Wyświetlacz LCD 4x20	7
3.3	Servomechanizm FT5715M	7
3.4	Servomechanizm OKY8015	7
3.5	Moduł radiowy HM-TRLR-S-868	7
4	Projekt elektroniki	8
5	Konstrukcja mechaniczna	9
6	Opis działania programu	10
6.1	Sterowanie servomechanizmem	10
6.2	Obsługa klawiatury	10
6.3	Wyświetlacz LCD	10
6.4	Obsługa modułu radiowego	10
6.5	Opis działania	12
7	Aplikacja webowa	13
8	Symulator sondy stratosferycznej	14
9	Podsumowanie	15
	Bibilografia	16

1 Opis projektu

Celem projektu jest zbudowanie anteny śledzącej, pełniącej funkcję stacji odbiorczej, umożliwiającej odbieranie sygnału od sondy stratosferycznej przy dużych odległościach. Stacja odbiorcza wyposażona będzie w antenę kierunkową o dużym wzmocnieniu, której orientacja będzie stale korygowana, aby maksymalnie wykorzystać zysk kierunkowy anteny [2].

W niektórych publikacjach [1] możemy zauważyć, iż korzyści płynące z zastosowania sterowanej anteny nie są tak wysokie jakbyśmy oczekiwali (lub nawet niezauważalne), w porównaniu z zastosowaniem anteny dookólnej. Warto jednak zauważyć, że przy badaniach, dających tego typu wyniki, odległość pomiędzy stacją bazową a np. robotem jest niewielka (we wspomnianej publikacji [1] - kilkaset metrów), w porównaniu do odległości rzędu kilkunastu-kilkudziesięciu kilometrów, z którymi będziemy mieć do czynienia w przypadku łączności z sondą stratosferyczną.

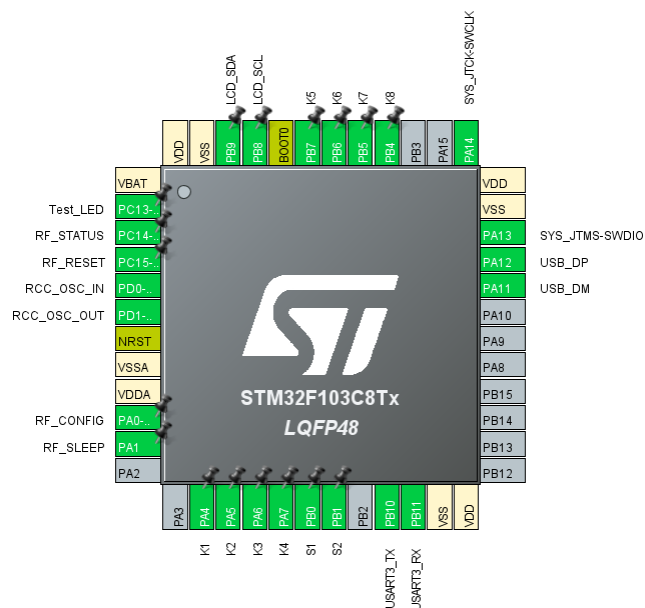
Niniejszy projekt będzie oparty o mikrokontroler z rodziny STM32103C8, znajdujący się na płytce STM32 Blue Pill (dokumentacja płytki ze strony producenta [7]) oraz moduł radiowy HM-TRLR-S-868 [4], pracujący w paśmie 868 MHz. Mikrokontroler będzie również połączony z wyświetlaczem alfanumerycznym LCD, klawiaturą oraz serwami, umożliwiającymi zmianę orientacji anteny. Dodatkowo zostanie stworzona aplikacja webowa, z poziomu której będzie można odczytać na komputerze - podłączonym do mikrokontrolera poprzez interfejs USB - położenie sondy, kąty nachylania serw czy moc sygnału. Z poziomu aplikacji będzie możliwe również „ręczne” sterowanie anteną.



Rysunek 1: Architektura systemu

2 Konfiguracja mikrokontrolera

Do wykonania projektu użyty zostanie zestaw ewaluacyjny wyposażony w mikrokontroler STM32F103C8T6.



Rysunek 2: Konfiguracja wyjść mikrokontrolera w programie STM32CubeMX

2.1 Konfiguracja pinów

Numer pinu	PIN	Tryb pracy	Funkcja/etykieta
2	PC13-TAMPER-RTC	GPIO _Output	RF_STATUS RF_RESET
3	PC14	GPIO _Input	
4	PC15	GPIO _Output	
5	PD0-OSC_IN	RCC_OSC_IN	
6	PD1-OSC_OUT	RCC_OSC_OUT	RF_CONFIG RF_SLEEP
10	PA0	GPIO _Input	
11	PA1	GPIO _Output	
12	PB10	USART3_TX	
13	PB11	USART3_RX	K8 K7 K6 K5
14	PA4	GPIO _Input	
15	PA5	GPIO _Input	
16	PA6	GPIO _Input	
17	PA7	GPIO _Input	LCD_SCL LCD_SDA
18	PB0	GPIO _Analog	
19	PB1	GPIO _Analog	
21	PB10	USART3_TX	
22	PB11	USART3_RX	USB_DM USB_DP
32	PA11	USB_DM	
33	PA12	USB_DP	
40	PB4	GPIO _Input	
41	PB5	GPIO _Input	K8 K7 K6 K5
42	PB6	GPIO _Input	
43	PB7	GPIO _Input	
45	PB8	I2C1_SCL	
46	PB9	I2C1_SDA	

Tabela 1: Konfiguracja pinów mikrokontrolera

Za pomocą pinów oznaczonych jako K1 - K8 odczytywane będzie wejście klawiatury. Piny S1 i S2 służyć będą do sterowania serwomechanizmami. Pozostałe piny służą do komunikowania się z urządzeniami zewnętrznymi.

2.2 I2C

Standard transmisji wykorzystywany do komunikacji z wyświetlaczem LCD.

Parametr	Wartość
I2C Speed Mode	Standard Mode
I2C Clock Speed (Hz)	100000
Clock No Stretch Mode	Disabled
Primary Address Length selection	7-bit
Dual Address Acknowledged	Disabled
Primary slave address	0
General Call address detection	Disabled

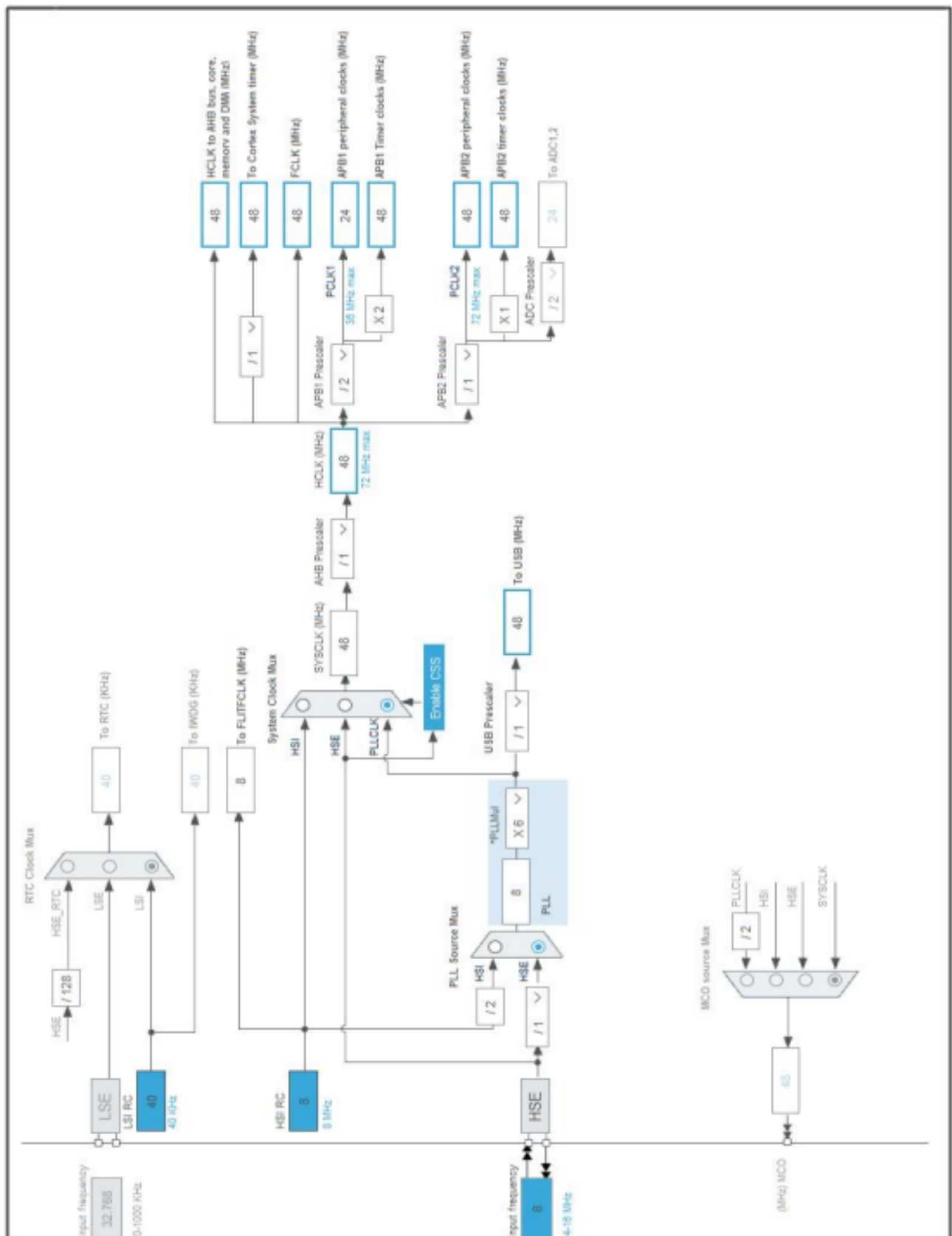
Tabela 2: Konfiguracja peryferium I2C

2.3 RCC

Peryferium zostanie wykorzystane do zapewnienia taktującego sygnału zegarowego.

Parametr	Wartość
VDD voltage (V)	3.3
Prefetch Buffer	Enabled
Flash Latency(WS)	1 WS (2 CPU cycle)
RCC ParametersHSI Calibration Value	16
HSE Startup Timeout Value (ms)	100
LSE Startup Timeout Value (ms)	5000

Tabela 3: Konfiguracja peryferium RCC



Rysunek 3: Konfiguracja zegarów mikrokontrolera

2.4 USART3

Standard transmisji wykorzystywany do komunikacji z modułem radiowym.

Parametr	Wartość
Baud Rate	9600
Word Length	8 Bits (including Parity)
Parity	None
Stop Bits	1

Tabela 4: Konfiguracja peryferium USART2

2.5 USB

Standard transmisji wykorzystywany do komunikacji z komputerem.

Parametr	Wartość
Speed	Full Speed 12MBit/s
Low Power	Disabled
Link Power Management	Disabled
Battery Charging	Disabled

Tabela 5: Konfiguracja peryferium USB

Komunikacja z komputerem odbywa się z wykorzystaniem *Virtual COM Port* z prędkością 115200 bps.

2.6 TIM3

Timer wykorzystywany jako źródło sygnału PWM sterującego serwomechanizmem

mode: Clock source
Channel3: PWM Generation CH3

Parametr	Wartość
Counter Settings:	
Prescaler (PSC - 16 bits value)	48-1
Counter Mode	Up
Counter Period (AutoReload Register - 16 bits value)	20000-1
Internal Clock Division (CKD)	No Division
auto-reload preload	Disable
Trigger Output (TRGO) Parameters:	
Master/Slave Mode (MSM bit)	Disable (Trigger input effect not delayed)
Trigger Event Selection	Reset (UG bit from TIMx_EGR)
PWM Generation Channel 3:	
Mode	PWM mode 1
Pulse (16 bits value)	0
Output compare preload	Enable
Fast Mode	Disable
CH Polarity	High

Tabela 6: Konfiguracja peryferium USB

3 Urządzenia zewnętrzne

3.1 Klawiatura membranowa

Klawiatura została wykorzystana jako prosta bezpośrednia forma komunikacji ze stacją odbiorczą. Jej obsługa odbywa się z wykorzystaniem multipleksacji.

3.2 Wyświetlacz LCD 4x20

Wyświetlacz [8] został wykorzystany do prostej komunikacji ze stacją odbiorczą.

Komunikacja z wyświetlaczem odbywa się za pomocą 8-bitowego ekspandera PCF8574 do magistrali I2C [9]. Adres PCF8574: 0x3F.

3.3 Serwomechanizm FT5715M

Serwomechanizm [5] o zakresie 180° do sterowania wysokością anteny. Ze względu na mechaniczne ograniczenia zakres zmniejszony został do 90°.

3.4 Serwomechanizm OKY8015

Serwomechanizm [6] o zakresie 270° do sterowania kierunkiem zwrotu anteny.

3.5 Moduł radiowy HM-TRLR-S-868

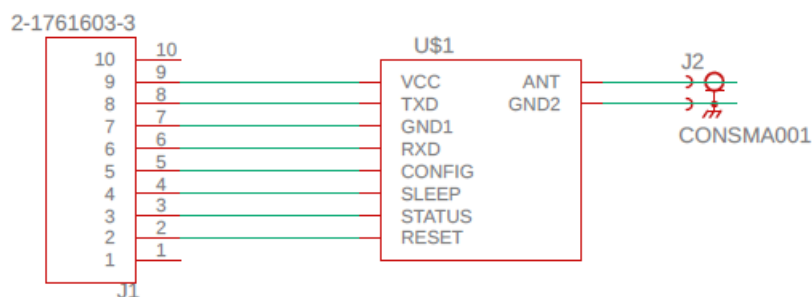
Moduł radiowy [4] działający na paśmie 868 MHz. Komunikuje się z mikrokontrolerem za pośrednictwem interfejsu UART. Oferuje m.in. pracę w modulacji LoRa, FSK, GFSK, OOK; nadawanie z różną mocą sygnału i szerokością pasma. Konfiguracja odbywa się za pomocą komend AT. Wykorzystaliśmy domyślne parametry pracy modułu, zamieszczone poniżej:

Command	default	remark
AT+SPR=n	n=3	Baud rate 9600pbs
AT+SPC=n	n=0	None check
AT+POWER=n	n=0	Power 20dbm
AT+CS=n	n=A	868MHz
AT+SYNL=n	n=6	6 bytes
AT+NODE=n,mode	n=0,mode=0	Disable ID Node function
AT+LRCRC=n	n=1	Lora mode,CRC enable
AT+LRSBW=n	n=7	SBW = 125KHz
AT+LRSF=n	n=9	SF = 9
AT+LRRC=n	n=0	CodeRate=4/5
AT+LRHF=n	n=0	FHSS is disable
AT+LRPL=n	n=32	Package lenth 32bytes
AT+LRHPV=n	n=10	Hopping period
AT+LRFSV=n	n=1638	Frequence step 100KHz
AT+MODE=n	n=0	LoRa mode
AT+BAND=n	n=2	868MHz band

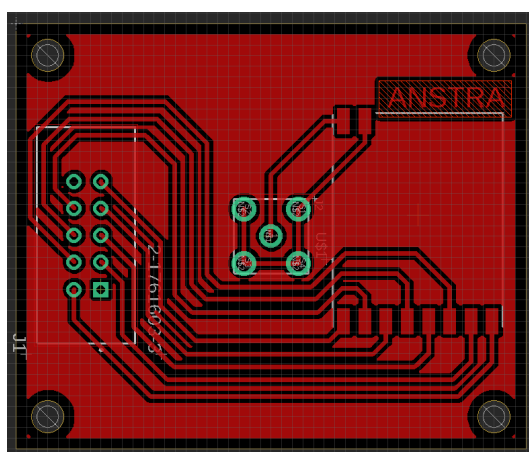
Rysunek 4: Parametry komunikacji wykorzystane w projekcie

4 Projekt elektroniki

W celu umożliwienia korzystania z modułu radiowego, zaprojektowano płytkę PCB, umożliwiającą przy-mocowanie anteny 50 Ω zakończonej wtykiem SMA, oraz stanowiącą przejściówkę na raster 2,54 mm. Na płytce znajdziemy moduł HM-TRLR-S-868, gniazdo proste SMA żeńskie oraz złącze IDC 10 pin.



Rysunek 5: Przejściówka PCB dla modułu radiowego - schemat

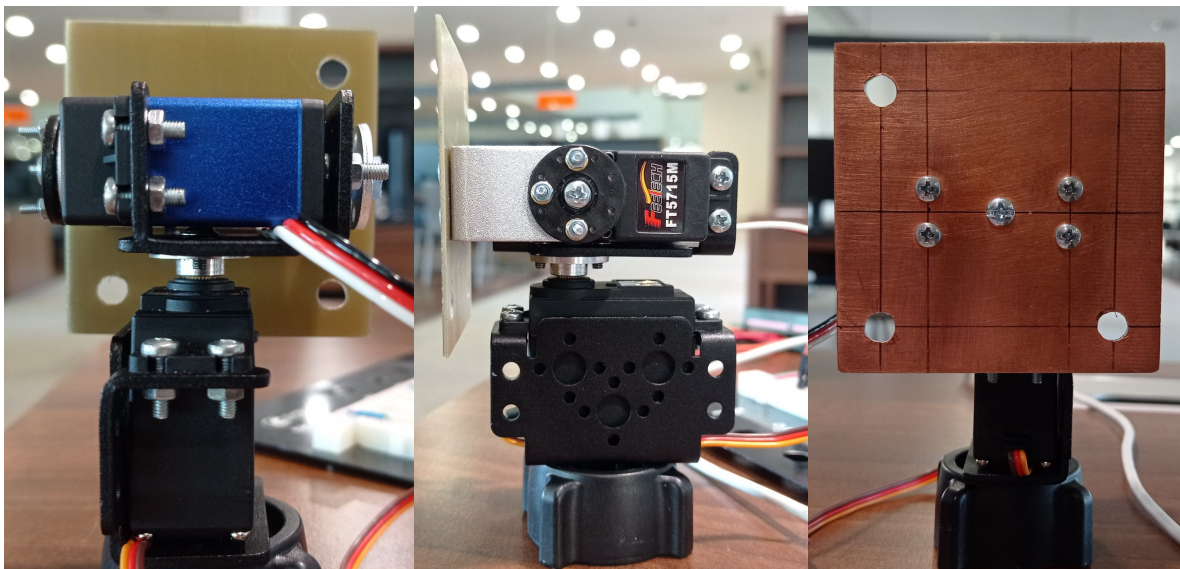


Rysunek 6: Przejściówka PCB dla modułu radiowego - widok płytki

5 Konstrukcja mechaniczna

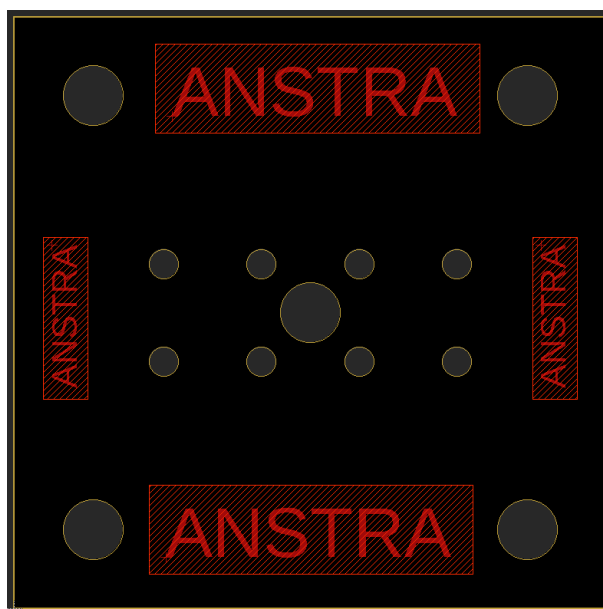
W celu sterowania orientacją anteny, stworzono konstrukcję mocowaną do statywu fotograficznego. Jej zadaniem jest sztywne umocowanie serw. Wykonana została z gotowych komponentów z własnymi modyfikacjami.

Z powodów ograniczeń konstrukcyjnych zakres górnego serwomechanizmu ograniczony jest do 90°.



Rysunek 7: konstrukcja mechaniczna układu naprowadzającego

Widoczna na zdjęciach (rys. 7) płytki PCB pozwala na zamocowanie anteny panelowej [3] do układu naprowadzającego. Została ona wykonana na podstawie projektu, przedstawionego poniżej (rys.8). Płytki pełni wyłącznie funkcję mechaniczną. Z powodu problemów z firmą, wykonującą PCB dla naszego projektu, wykonaliśmy ją samodzielnie.



Rysunek 8: Projekt PCB do montażu anteny panelowej

Na rys. 9 zamieszczono zdjęcia układu naprowadzającego - bez anteny i PCB z lewej, z zamocowaną anteną po prawej - na statywie.



Rysunek 9: Układ naprowadzający na statywie

6 Opis działania programu

6.1 Sterowanie serwomechanizmem

Serwomechanizmy są sterowane poprzez wypełnienie sygnału PWM generowanego przez Timer3 mikrokontrolera. Kąt wychylenia serwomechanizmu steruje się wypełnieniem z zakresu $[0.5, 2.5]$ ms odpowiadającego zakresowi serwomechanizmu.

6.2 Obsługa klawiatury

Funkcja sprawdza który przycisk na klawiaturze membranowej został przyciśnięty i zwraca jego numer (-1 gdy klawiatura nie jest przyciśnięta).

W trybie wpisywania wartości klawisz '*' działa jako znak '-', a klawisz '#' jako '.'. Klawisz 'D' służy do zatwierdzenia wartości.

W trybie wyświetlania klawisz 'A' służy do manualnego nastawienia kąta, a klawisz 'B' do nastawienia celu.

6.3 Wyświetlacz LCD

Z wyświetlaczem LCD mikrokontroler komunikuje się poprzez 8-bitowy ekspander I2C. Wyświetlacz działa w trybie 4-bitowym (pozostałe 4 bity ekspandera wykorzystywane są do pinów kontrolnych Enable, Read_write itp.).

Gdy ANSTRA oczekuje na wpisanie wartości na wyświetlaczu pojawia się migający kursor.

6.4 Obsługa modułu radiowego

ANSTRA komunikuje się z modułem radiowym z wykorzystaniem interfejsu UART. Odczytywanie danych z modułu odbywa się znak po znaku, z wykorzystaniem przerwań. Każdy komunikat przesyłany jest w formie `$<tresc>#`.

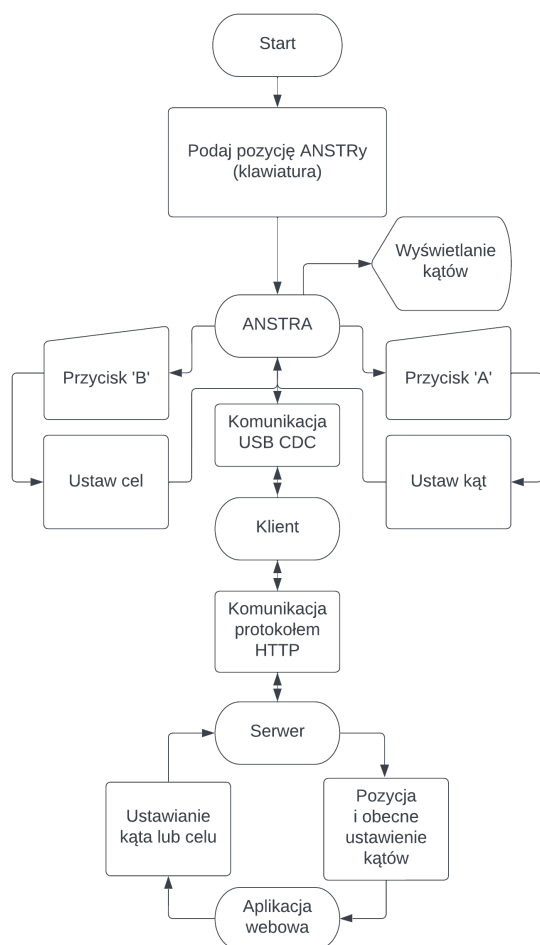
Po wykryciu zakończenia komunikatu (znak #) zapisywane są: szerokość i długość geograficzna oraz wysokość celu. Ponadto wskaźnik na bufor do którego zapisujemy odebrane przez UART dane przesuwany jest na początek tabeli RxData[]. Konwersja ciągu znaków na liczby (parametry) odbywa się z wykorzystaniem funkcji atof().

```
1 void HAL_UART_RxCpltCallback(UART_HandleTypeDef *huart){
2     if(huart->Instance==USART3){
3
4     if((RxData[RxDataIndex] == (int) '#') && (RxData[0] == (int) '$')){
5         RxData[RxDataIndex] = 0;
6         RxDataIndex=0;
7         char* num;
8         if(strstr((char*)RxData, "SET_TARGET ") != NULL){
9             num = strtok((char*)RxData, " ");
10            if(num!=NULL){num = strtok(NULL, " "); target_latitude = atof(num);}
11            if(num!=NULL){num = strtok(NULL, " "); target_longitude = atof(num);}
12            if(num!=NULL){num = strtok(NULL, " "); target_height = atof(num);}
13        }
14
15        ...
16
17        HAL_UART_Receive_IT(&huart3, &RxData[RxDataIndex], 1);
18    }
19 }
```

6.5 Opis działania

ANSTRA po starcie oczekuje na swoją pozycję geograficzną. Należy podać trzy zmienne:

- Szerokość geograficzną w stopniach
- Długość geograficzną w stopniach
- Kąt, jaki ANSTRA tworzy z z kierunkiem wschodnim idącym przeciwnie do ruchu wskazówek zegara w stopniach (np gdy ANSTRA skierowana jest w kierunku północnym kąt wynosi 90°)



Szerokość północna i długość wschodnia oznaczane są jako wartości dodatnie, a szerokości południowe i długości zachodnie jako ujemne.

Po otrzymaniu swojej pozycji ANSTRA ustawia się w swojej pozycji początkowej. Wartość "wysokość" oznacza kąt nachylenia kąta do podłoża (0° oznacza pion) i może się zawierać w przedziale 0° - 90° , natomiast wartość "kierunek" zawiera się w przedziale -135° - 135° .

ANSTRA orientuje się na ostatnią pozycję, odebraną od sondy stratosferycznej poprzez moduł radiowy. Duża częstotliwość nadawania i odbierania sygnału pozwala „śledzić” sondę poprzez odpowiednie ustawienie kątów serw. Maksymalizuje to moc odbieranego sygnału.

ANSTRA umożliwia również manualne nastawienie kąta oraz celu, na który zwrócić się ma antena. Przyciskiem 'A' na klawiaturze rozpoczynamy wpisywanie nastaw kątów. Najpierw należy podać kąt "kierunku", następnie "wysokości". By ustawić cel, należy nacisnąć przycisk 'B'. Pozycję celu definiują trzy wartości: jego szerokość i długość geograficzna oraz wysokość nad powierzchnią wyrażona w kilometrach.

Nastawy ANSTRY można również zmienić zdalnie z poziomu aplikacji webowej.

Algorytm wyliczający nastawy kątów na podstawie pozycji satelity:

```

1 void target_to_angle() {
2     double alf = (target_longitude - longitude)/180.0*M_PI;
3     double bet = (target_latitude - latitude)/180.0*M_PI;
4
5     double satX = cos(alf)*cos(bet)*(target_height+R)-R;
6     double satY = sin(alf)*cos(bet)*(target_height+R);
7     double satZ = sin(bet)*(target_height+R);
8
9     ha = acos( satX/( sqrt( satX*satX+satY*satY+satZ*satZ ) ) )/M_PI*180.0;
10    da = atan(satZ/satY)/M_PI*180.0 - direction;
11    if( satY < 0 )
12        if( da > 0 )
13            da -= 180;
14    else

```

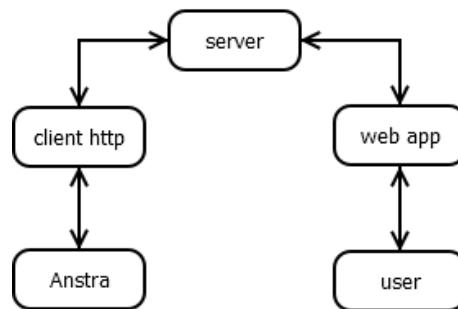
```

15     da += 180;
16 }

```

7 Aplikacja webowa

W ramach projektu stworzony został serwer REST API, umożliwiający zdalną komunikację z ANSTRĄ.



Rysunek 10: Schemat komunikacji pomiędzy Anstrą a użytkownikiem

Serwer został napisany w języku *Python3* w oparciu o bibliotekę *FastApi*. Aplikacja działa na porcie 9000. Obsługuje metody GET i POST, dzięki którym możemy:

- odebrać dane (kąty serw, pozycję sondy, pozycję Anstry)
- zadać pozycję celu (sondy)
- ręcznie ustawić kąty serw

Powyższe akcje wykonywane są przez aplikację webową. Odczyt danych z ANSTRY jest ogólnodostępny. Zadawanie pozycji i ustalanie kątów serw możliwe jest po uprzednim podaniu tokenu dostępu.

The screenshot shows a web application interface with a light green background. At the top, there are three tabs: 'GET /anstra_overview GetParameters', 'POST /angles Set Angles', and 'POST /target Set Target Position'. The 'POST /target' tab is selected and expanded. Below the tabs, there is a 'Parameters' section with a 'Try it out' button. The parameters are listed in a table:

Name	Description
token * required (query)	token
user * required (query)	user
latitude * required (query)	latitude
longitude * required (query)	longitude
height * required (query)	height

Rysunek 11: Widok z aplikacji webowej

ANSTRA komunikuje się z serwerem za pośrednictwem programu *client.py* uruchamianemu na komputerze PC, z którym połączona jest za pośrednictwem USB. ANSTRA cyklicznie wysyła informacje o pozycji sondy i aktualnych kątach serw. *Client* każdorazowo odbiera te dane i przesyła na serwer.

Serwer posiada specjalny endpoint */execute*. Jeżeli kąty serw lub pozycja celu zostały manualnie ustawione, po przesłaniu zapytania metodą GET, serwer odpowie jakie parametry zostały zmienione i jakie

są ich wartości. *Client* regularnie wysyła w/w zapytania. Kiedy w odpowiedzi serwera znajdzie klucz 'error' z wartością 'angles' albo 'target', przekaże on odpowiednią komendę do ANSTRY, co pozwoli na zadane wystrojenie.

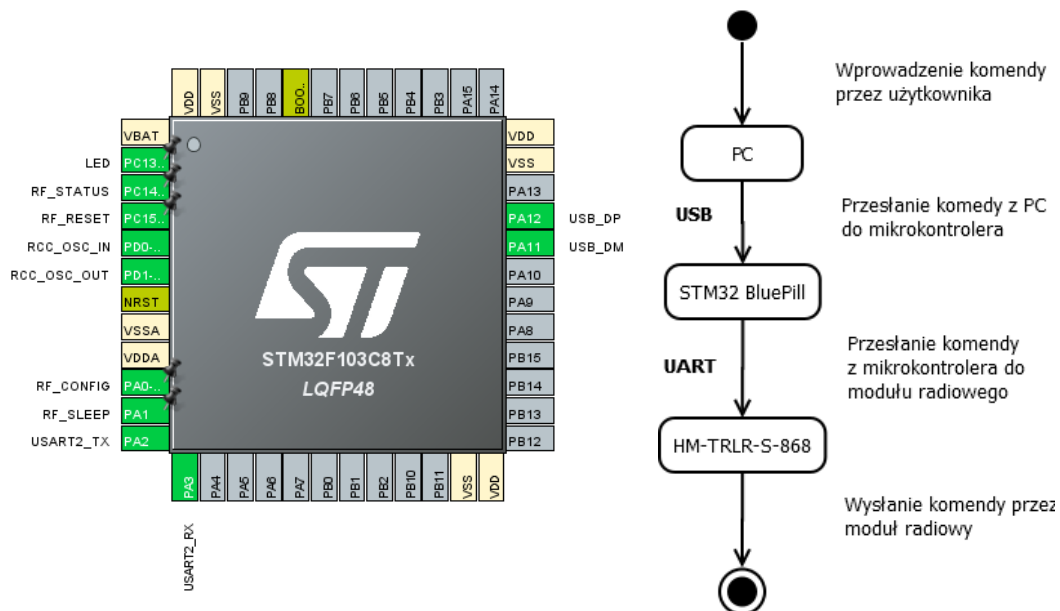
```

1  @anstra_app.get('/execute')
2  async def execute_set_values(token):
3  if token != app_token: return {'error' : 'bad token!'}
4  global servo1_angle, servo2_angle, target_latitude, target_longitude,
   target_height, changed, changed_parameter
5  if changed == False: return {'error' : 'no changes'}
6  match changed_parameter:
7      case 'angles' :
8          response = {'error' : 'no error', 'changed_parameter' : 'angles', '
9                      servo' : {'servo_1' : servo1_angle, 'servo_2' : servo2_angle}}
10         case 'target' :
11             response = {'error' : 'no error', 'changed_parameter' : 'target', '
12                         target' : {'latitude' : target_latitude, 'longitude' :
13                                     target_longitude, 'height' : target_height}}
14
15     changed = False
16     changed_parameter = None
17     return response

```

8 Symulator sondy stratosferycznej

W ramach projektu wykonano symulator sondy stratosferycznej, składający się z płytki STM32 BluePill [7] oraz modułu radiowego HM-TRLR-S-868 [4] (elementy identyczne z wykorzystanymi w ANSTRZE).



Rysunek 12: Pinout nadajnika - z lewej oraz schemat działania nadajnika - z prawej

Parametry konfiguracji interfejsu USB i UART2 są identyczne z zastosowanymi w ANSTRZE.

Działanie nadajnika opiera się głównie na zmodyfikowanej funkcji CDC_Receive_FS(). Jej kod zamieszczono poniżej:

```

1  static int8_t CDC_Receive_FS(uint8_t* Buf, uint32_t *Len)
2  {
3      /* USER CODE BEGIN 6 */
4      Buf[*Len] = 0;
5      sprintf((char*)USB_RxData, "%s#", (char*)Buf);
6      HAL_UART_Transmit_IT(&huart2, USB_RxData, strlen((char*)USB_RxData));

```

```

7  memset(Buf, 0, *Len);
8  HAL_GPIO_TogglePin(LED_GPIO_Port, LED_Pin);
9
10
11  USBD_CDC_SetRxBuffer(&hUsbDeviceFS, &Buf[0]);
12  USBD_CDC_ReceivePacket(&hUsbDeviceFS);
13  return (USBD_OK);
14  /* USER CODE END 6 */
15 }

```

9 Podsumowanie

Udało się spełnić wszystkie założenia projektowe. ANSTRę możnaysterować manualnie, z aplikacji webowej oraz bezpośrednio przez sondę wysyłającą swoją lokalizację.

Literatura

- [1] Antena; Active Antenna Tracking System with Directional Antennas for Enhancing Wireless Communication Capabilities of a Networked Robotic System. https://www.ri.cmu.edu/pub_files/2015/5/Min_et_al-2015-JFR.pdf.
- [2] Antena; Działanie na przykładzie radaru. <http://www.radary.az.pl/anten.php>. Dostęp: 21.03.2022.
- [3] Antena panelowa GSM 850 - 900 MHz. <https://yagi.pl/antena-gsm-cyklon-14hv-900-mhz-ze-zlaczem-nz>.
- [4] Moduł radiowy; HM-TRLR-S. <https://www.tme.eu/Document/cf7d155a690ebd02d216fa6dd8d39ccf/HM-TRLR-S-868.pdf>.
- [5] Serwomechanizm FT5715M. <https://botland.com.pl/serwa-typu-standard/9182-serwo-feetech-ft5715m-standard-5904422312756.html>. Dostęp: 25.04.2022.
- [6] Serwomechanizm OKY8015. <https://www.tme.eu/pl/details/oky8015/serwomechanizmy/okystar/>. Dostęp: 15.06.2022.
- [7] STM32 BluePill. https://stm32-base.org/assets/pdf/boards/original-schematic-STM32F103C8T6-Blue_Pill.pdf.
- [8] Wyświetlacz LCD; Datasheet. https://uk.beta-layout.com/download/rk/RK-10290_410.pdf.
- [9] Wyświetlacz LCD; ekspander PCF8574. https://botland.com.pl/index.php?controller=attachment&id_attachment=210.