



AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

**WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,
INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ**

KATEDRA Automatyki i Inżynierii Biomedycznej

Praca dyplomowa inżynierska

*Elektroniczny kontroler lotu dla platformy latającej typu
quadrocopter*

Electronic flight controller for a quadrocopter

Autor:	<i>Karol Kaim</i>
Kierunek studiów:	<i>Automatyka i Robotyka</i>
Opiekun pracy:	<i>Dr inż. Paweł Skruch</i>

Kraków, 2015

Oświadczam, świadomy odpowiedzialności karnej za poświadczenie nieprawdy, że niniejszą pracę dyplomową wykonałem osobiście i samodzielnie i że nie korzystałem ze źródeł innych niż wymienione w pracy.

<podpis dyplomanta>

Spis treści

1. Wstęp.....	5
1.1 Wprowadzenie	5
1.2 Cel i zakres pracy.....	5
2. Przegląd zagadnień wykorzystanych w pracy.....	7
2.1 Platforma latająca typu quadrocopter	7
2.1.1 Quadrocopter – wprowadzenie.....	7
2.1.2 Dynamika platformy typu quadrocopter	8
2.2 Niezbędne urządzenia pomiarowe	9
2.2.1 Urządzenia MEMS.....	9
2.2.2 Opis zastosowanych urządzeń pomiarowych.....	9
2.3 Reprezentacja orientacji przestrzennej obiektu	11
2.3.1 Wprowadzenie.....	11
2.3.2 Kąty Eulera.....	12
2.3.3 Kwanterniony	12
2.4 Sterowanie systemów dynamicznych	14
2.4.1 Zamknięty układ regulacji	14
2.4.2 Regulator PID	15
3. Budowa platformy typu quadrocopter.....	17
3.1 Wykaz podzespołów	17
3.2 Opis podzespołów.....	17
4. Implementacja sterownika kontroli lotu.....	21
4.1 Sterownik kontroli lotu – wprowadzenie.....	21
4.2 Sterownik kontroli lotu – komponenty	21
4.2.1 Texas Instruments Tiva C-Series.....	21
4.2.2 Pololu AltIMU-10	22
4.2.3 Przetwornica napięcia step-down Pololu- D24V50F5 5V, 5A	23
4.2.4 Dedykowany obwód drukowany	23
4.3 Opis algorytmu	24
4.4 Stabilizacja lotu.....	25
4.4.1 Zadawanie i odczyt sterowań	25
4.4.2 Odczyt danych pomiarowych	27
4.4.3 Wyznaczenie orientacji.....	28
4.4.4 Wyznaczenie sygnałów sterujących	30

5.	Testy i dobór parametrów	33
5.1	Komunikacja z jednostką pomiarową IMU.....	33
5.2	Orientacja quadrocoptera	33
5.2.1	Testy statyczne	34
5.2.2	Testy dynamiczne	35
5.3	Dobór nastaw regulatorów PID	36
6.	Uwagi końcowe i wnioski	39
7.	Bibliografia :	41

1. Wstęp

1.1 Wprowadzenie

Szybki rozwój techniki oraz miniaturyzacja sprawiły, że właściwie w każdej dziedzinie życia napotykamy na urządzenia zbudowane w oparciu o zaawansowaną elektronikę. Głównym zadaniem tego typu systemów wbudowanych jest poprawa jakości życia codziennego lub, co nie mniej istotne, zapewnienie rozrywki użytkownikom.

Doskonałym przykładem spełniającym te dwie cechy wydaje się być mulitokopter. Helikoptery tego typu wyposażone są, w zależności o konfiguracji, w 4,5,6 a nawet 8 silników. Prosta konstrukcja w połączeniu z coraz tańszymi układami kontroli lotu sprawia, że urządzenia te zyskują na popularności. Dzieje się tak również dzięki ogromnym możliwościom tych maszyn. Jednym z pomysłów, który już doczekał się realizacji, jest używanie multicopterów do szybkiej lokalizacji ofiar lawin śnieżnych. Nie trudno też sobie wyobrazić inspekcję budynków na terenach dotkniętych przez katastrofy naturalne, np. trzęsienia ziemi czy fale tsunami, dokonywaną przez zdalnie sterowane urządzenia wyposażone w systemy wizyjne. Multikopter wyposażony w kamerę 3D może również zostać wykorzystany do sporządzenia szczegółowych planów budynku. Jest to wreszcie idealne narzędzie do kręcenia bardzo ciekawych ujęć filmowych, zdecydowanie obniżające koszt ich powstania. Doskonale sprawdza się w filmach przyrodniczych, produkcjach amatorskich czy filmach o tematyce sportowej, w których multikopter podąża np. za zjeżdżającym narciarzem.

1.2 Cel i zakres pracy

Celem niniejszej pracy jest zbudowanie latającej platformy typu quadcopter, sterowanej zdalnie przy użyciu aparatury modelarskiej. Prace związane z budową platformy można podzielić na dwa etapy. Pierwszy z nich dotyczy wyboru odpowiednich podzespołów oraz zaprojektowania dedykowanego obwodu drukowanego, łączącego wszystkie elementy elektroniczne, które zostaną zamontowane na ramie quadcoptera. Drugi etap to wykonanie sterownika kontroli lotu w oparciu o jednostkę pomiarową IMU (z ang. *Inertial Measurement Unit*), mikrokontroler i odbiornik radiowy.

W kolejnych rozdziałach pracy zostały podjęte zagadnienia teoretyczne związane z budową platformy. Poruszone zostaną takie kwestie jak : zamknięty układ regulacji, niezbędne przyrządy pomiarowe, sposoby przedstawienia orientacji obiektu w przestrzeni oraz sterowania platformą latającą typu quadcopter.

Następnie zaprezentowana zostanie krótka historia budowy platformy. Od listy wszystkich niezbędnych komponentów, poprzez ich krótką specyfikację, kończąc na integracji wszystkich elementów.

Dalej przedstawione zostanie rozwiązanie głównego i najbardziej czasochłonnego problemu związanego z budową quadcoptera, jakim jest implementacja sterownika kontroli lotu.

Algorytm sterowania można podzielić na kilka etapów. Wszystkie z nich zostaną dokładnie zaprezentowane, wraz z krótką charakterystyką mikrokontrolera i jednostki pomiarowej IMU.

Ostatni etap prac to testy zaimplementowanych rozwiązań. Każdy etap budowy związany jest z bardziej lub mniej szczegółowymi testami. Dotyczą one min. komunikacji z jednostką pomiarową IMU, odczytów z aparatury modelarskiej, wyznaczania orientacji obiektu czy doboru nastaw regulatorów odpowiedzialnych za stabilizację quadrocoptera w powietrzu.

Wyniki pracy związane z budową platformy wraz z wnioskami zaprezentowane zostaną w ostatnim rozdziale.

2. Przegląd zagadnień wykorzystanych w pracy

2.1 Platforma latająca typu quadcopter

2.1.1 Quadcopter – wprowadzenie

Quadcopter to rodzaj statku powietrznego. Jest zaliczany do kategorii helikopterów, które wytwarzają siłę nośną dzięki ruchowi obrotowemu wirników napędzanych przez silniki. Quadcopter zbudowany jest z 4 silników, wraz z regulatorami i śmigłami, ramy, sterownika kontroli lotu, odbiornika RC oraz pakietu zasilającego.

Zdecydowana większość konstrukcji to quadcoptery w tzw. układzie „+”. Rama tego typu przypomina znak dodawania, każde z czterech ramion rama ma identyczną długość, a kąt pomiędzy sąsiednimi ramionami to 90^0 . Silniki przykręcone są na końcach ramion, natomiast sterownik kontroli lotu i pakiet zasilający zamontowany jest centralnie. Dużą zaletą takiego układu jest to, że można łatwo wyznaczyć orientację quadcoptera znajdującego się w powietrzu. Dzięki temu pilotaż jest stosunkowo prosty, nawet dla osób zaczynających swoją przygodę z modelarstwem RC.

Rzadziej spotykane są quadcoptery w układzie „H”. Wykorzystuje się je zazwyczaj do latania z użyciem systemu wizyjnego FPV(z ang. *First Person View*). Taki układ sprawia, że pole widzenia kamery nie obejmuje śmigieł, dzięki czemu pilot może w pełni skupić się na kontrolowaniu quadcoptera.



Rys. 1 Przykład platformy typu quadcopter.

2.1.2 Dynamika platformy typu quadcopter

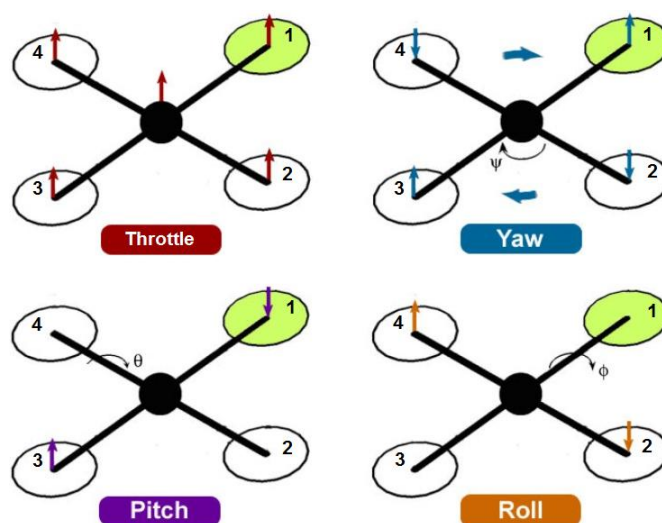
Każdy statek powietrzny wyposażony jest w odpowiedni zespół napędowy, odpowiedzialny za wytworzenie ciągu na skutek przyspieszenia strumienia powietrza. Ciąg umożliwia pokonanie siły bezwładności przy starcie, przyspieszaniu lub zmianie kierunku ruchu, siły ciężkości przy wznoszeniu oraz oporu aerodynamicznego.

Quadcopter wyposażony jest w cztery silniki prądu stałego, w których moment obrotowy powstaje na skutek sił działających w polu magnetycznym na uzwojenia przewodzące prąd elektryczny. Moment obrotowy przetwarzany jest na siłę ciągu przy użyciu śmigła stałobrotowego zamocowanego do piasty silnika.

Aby uniknąć ciągłego obracania się quadcoptera wokół osi *yaw* (kąty *RPY* opisane są w pkt. 2.3.2), wypadkowy moment obrotowy, działający na platformę, powinien wynosić 0 Nm. W quadcopterze, w przeciwieństwie do konwencjonalnego helikoptera, można to zrealizować w prosty sposób. Przyjmując oznaczenie silników jak na rys.2, wystarczy aby silniki 1 i 3 obracały się zgodnie z ruchem wskazówek zegara - CW (z ang. *Clockwise*), a silniki oznaczone nr 2 i 4 obracały się przeciwnie do ruchu wskazówek zegara - CCW (z ang. *Counterclockwise*).

Sterowanie quadcopterem polega na zadawaniu przez pilota wartości kątów *RPY* oraz wartości *throttle*. Zwiększenie wartości *throttle* powoduje równomierny wzrost ciągu poprzez jednakową zmianę prędkości obrotowej każdego z silników. Dzięki temu możliwe jest poderwanie quadcoptera w powietrze oraz utrzymanie go na odpowiedniej wysokości.

Zmiana wartości kątów *RPY*, umożliwia kołysanie boczne, kołysanie wzdłużne oraz zbaczanie quadcoptera [1]. Zmiana kąta *roll* polega na jednoczesnym zwiększeniu prędkości obrotowej silnika nr 4 oraz zmniejszeniu prędkości obrotowej silnika nr 2. Zmiana wartości kąta *pitch* jest analogiczna do przypadku zmiany wartości *roll*. Różnica polega na tym, że zmiana prędkości obrotowej dotyczy silników nr 1 i 3. Zmiana wartości kąta *yaw* jest możliwa, jeśli wypadkowy moment obrotowy, działający na quadcopter, jest różny od zera. Aby to osiągnąć, należy zwiększyć prędkość silników CW oraz zmniejszyć prędkość obrotową silników CCW.



Rys. 2 Sterowanie platformą latającą typu quadcopter [2].

2.2 Niezbędne urządzenia pomiarowe

2.2.1 Urządzenia MEMS

Układy MEMS (z ang. *Micro Electro-Mechanical Systems*) to zintegrowane urządzenia łączące elementy elektroniczne i mechaniczne, wykonane w miniaturowej skali. Główne zalety takich układów to: niska cena, małe rozmiary, dobra integracja z układami mikroprocesorowymi oraz niskie zużycie energii. Te cechy sprawiły, że układy MEMS znalazły bardzo szerokie zastosowanie. Układy MEMS najczęściej wykorzystywane są w charakterze sensorów, przetwarzających wielkości mechaniczne na wielkości elektryczne.

2.2.2 Opis zastosowanych urządzeń pomiarowych

Sterownik kontroli lotu został zbudowany w oparciu o trzy urządzenia pomiarowe - akcelrometr, żyroskop i magnetometr. Stabilizacja quadrocoptera wymaga zastosowania akcelrometru i żyroskopu. W przypadku, gdy sterownik kontroli lotu ma umożliwiać zbaczanie quadrocoptera, niezbędne jest zastosowanie magnetometru.

Żyroskop

Żyroskop to urządzenie pomiarowe mierzące prędkość kątową obiektu, na którym został zamontowany. Żyroskopy wykonane w technologii MEMS to czujniki wykorzystujące efekt Coriolisa. Drgająca masa, umieszczona na obrotowej tarczy, zmienia swoje położenie względem osi obrotu, w zależności od kierunku prędkości kątowej. Jednocześnie tarcza, która ulega odkształceniom, jest jedną z okładek kondensatora. Zmiana położenia masy powoduje odkształcenie tarczy, co implikuje zmianę pojemności kondensatora, która przetwarzana jest na sygnał napięciowy proporcjonalny do prędkości kątowej [3].

Nowoczesne żyroskopy wykonane w technologii MEMS to urządzenia umożliwiające pomiar prędkości kątowej wokół trzech osi jednocześnie. W połączeniu z niewielkimi rozmiarami, stanowią grupę urządzeń o bardzo szerokim zastosowaniu. Kontrolowanie urządzeń mobilnych, stabilizacja obrazu w aparatach i kamerach cyfrowych czy systemy bezpieczeństwa montowane w samochodach, to tylko nieliczne przykłady aplikacji wykorzystujących żyroskopy [4].



Rys. 3 Żyroskop wykonany w technologii MEMS oraz przykładowa aplikacja wykorzystująca urządzenie [5].

Do budowy kontrolera lotu wykorzystałem 3-osiowy żyroskop L3GD20, oferowany przez firmę *STMicroelectronics*.

Tab1. Wybrane parametry żyroskopu ST L3GD20 :

<i>Parametr</i>	<i>Wartość</i>	<i>Jednostka</i>	<i>Informacje dodatkowe :</i>
Rozdzielczość	16	bit	<ul style="list-style-type: none"> • Wbudowane filtry dolno i górno przepustowe • Obsługa magistrali I2C, SPI
Zakres pomiarowy	250/500/2000	dps	
Czułość dla zakresu 250 dps	8.75	mdps/LSB	

Akcelerometr

Akcelerometr to kolejne urządzenie niezbędne do wyznaczenia orientacji obiektu w przestrzeni. Akcelerometry wykorzystywane są do pomiarów statycznego przyspieszenia grawitacyjnego oraz przyspieszenia dynamicznego obiektu. Są to zazwyczaj przetworniki pojemnościowe. Do pomiaru przyspieszenia używają kondensatorów różnicowych, których pojemność jest mierzona i przekształcana na napięcie proporcjonalne do przyspieszenia [3].

Akcelerometry, podobnie jak żyroskopy, znalazły zastosowanie w bardzo wielu dziedzinach. Wykorzystywane są np. w branży motoryzacyjnej w układach sterujących poduszkami powietrznymi, w układach alarmowych czy systemach *hill-start assist*. Inne zastosowania to pomiary drgań, aplikacje wykrywające ruch człowieka lub kardiostymulatory [4].



Rys. 4 Akcelerometr wykonany w technologii MEMS oraz przykładowa aplikacja wykorzystująca urządzenie [6].

Magnetometr

Magnetometr to urządzenie umożliwiające pomiar wielkości i kierunku pola magnetycznego Ziemi. Urządzenia pomiarowe wykonane w technologii *MEMS* zazwyczaj wykorzystywane są do budowy kompasów elektronicznych oraz systemów *AHRS* (z ang. *Attitude and Heading Reference System*).

Do budowy kontrolera lotu wykorzystałem 3-osiowy akcelerometr i magnetometr LSM303DLHC, oferowany przez firmę *STMicroelectronics*.

Tab2. Wybrane parametry akcelerometru i magnetometru ST LSM303DLHC :

<i>Parametr</i>	<i>Wartość</i>	<i>Jednostka</i>	<i>Informacje dodatkowe :</i>
Rozdzielczość	16	bit	<ul style="list-style-type: none"> • Wbudowane czujnik temperatury • Obsługa magistrali I2C • Gwarantowana dokładność pomiarowa dla szerokiego zakresu temperatury : od -40°C do 85°C
Zakres pomiarowy przyspieszenia	$\pm 2/\pm 4/\pm 8/\pm 16$	g_n	
Zakres pomiarowy pola magnetycznego	$\pm 1.3/\pm 2.5 / \pm 4.0 / \pm 4.7/\pm 5.6 / \pm 8.1$	Gs	
Czułość dla zakresu $\pm 4g$	2	mg/LSB	

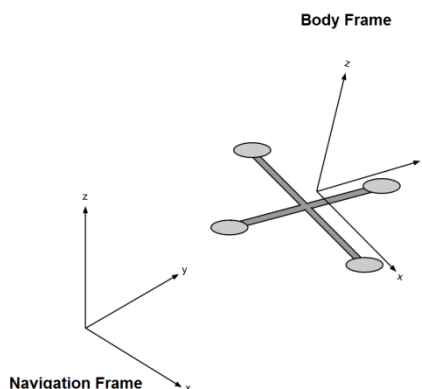
2.3 Reprezentacja orientacji przestrzennej obiektu

2.3.1 Wprowadzenie

Znajomość usytuowania obiektu w przestrzeni jest niezbędną informacją we wszystkich aspektach manipulacji robotów. Umieszczenie obiektu w przestrzeni może zostać opisane za pomocą jego położenia oraz orientacji.

W przypadku sterowania platformą latającą, niezbędna jest znajomość orientacji obiektu. Wyznaczenie orientacji związane jest ze zdefiniowaniem dwóch układów odniesienia w przestrzeni, w której odbywa się ruch. Układ bazowy definiowany jest jako nieruchomy układ przestrzeni nawigacyjnej *NF* (z ang. *navigation-frame*). Drugim niezbędnym układem odniesienia jest układ związany z obiektem *BF* (z ang. *body-frame*). Określenie orientacji obiektu, jest równoznaczne z opisem położenia układu związanego z obiektem względem układu przestrzeni nawigacyjnej [7].

Istnieje kilka sposobów opisu orientacji obiektu. Są to m.in.: *macierz rotacji*, *reprezentacja oś/kąt*, *kąty Eulera* lub *kwaterniony jednostkowe* [1]. W dalszej części rozdziału zostaną przybliżone dwie metody, które znalazły zastosowanie przy budowie kontrolera lotu. Są to *kąty Eulera* oraz *kwaterniony jednostkowe*.



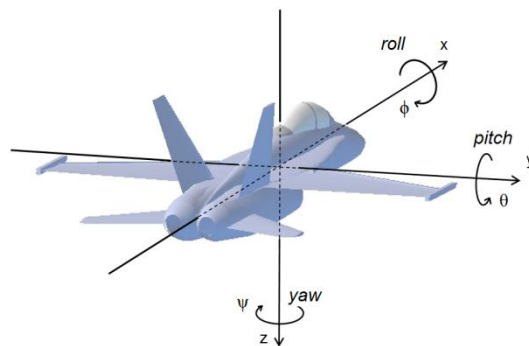
Rys. 5 Przykład zdefiniowania układu odniesienia navigation-frame, oraz układu związanego z obiektem body-frame.

2.3.2 Kąty Eulera

Kąty *Eulera* to minimalna reprezentacja rotacji, składająca się z trzech niezależnych parametrów [1]. *Kąty Eulera* opisują rotację pomiędzy dwoma układami współrzędnych, którą można przedstawić jako złożenie trzech elementarnych obrotów względem wybranych osi. Taki sposób reprezentacji nie jest jednoznaczny, ponieważ istnieje dwanaście różnych zbiorów *kątów Eulera*, odpowiadających możliwym sekwencjom elementarnych rotacji [8].

Kąty *RPY* (z ang. *roll, pitch, yaw*) to postać, która zyskała dużą popularność w nawigacji. Wypadkowa rotacja odpowiada złożeniu elementarnych rotacji względem układu odniesienia *navigation-frame*, o początku umiejscowionym w środku masy statku powietrznego:

- *roll* – obrót wokół osi x układu odniesienia
- *pitch* – obrót wokół osi y układu odniesienia
- *yaw* – obrót wokół osi z układu odniesienia



Rys. 6 Reprezentacja orientacji w postaci kątów *roll* ϕ , *pitch* θ , *yaw* ψ [28].

Kąty *RPY* są bardzo intuicyjne, znając ich wartość można bez problemu zinterpretować orientację obiektu względem układu odniesienia. Niestety, ten sposób reprezentacji jest ograniczony przez efekt zwany blokadą przegubów (z ang. *gimbal lock*). W przypadku reprezentacji *RPY* efekt ten pojawia się w momencie, gdy kąt *pitch* wynosi $\pm 90^\circ$. Wówczas tracony jest jeden stopień swobody, w wyniku czego rotacje *pitch* i *yaw* dają ten sam efekt.

2.3.3 Kwaterniony

Kwaterniony, będące uogólnieniem liczb zespolonych, które umożliwiają reprezentację obrotów na płaszczyźnie, mogą zostać wykorzystane do opisu orientacji obiektu w przestrzeni trójwymiarowej [9].

Kwaterniony stanowią nadmiarową reprezentację rotacji, ponieważ wykorzystują cztery parametry. Jednak w przeciwieństwie do kątów Eulera, umożliwiają parametryzację globalną.

Kwaternion definiowany jest w następujący sposób :

$$q = \eta + \varepsilon_1 i + \varepsilon_2 j + \varepsilon_3 k \quad (1)$$

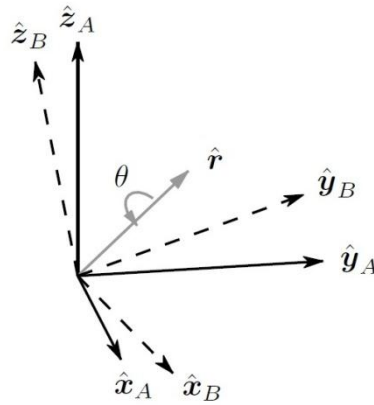
gdzie i, j, k są wersolami ortogonalnego układu współrzędnych.

Często stosuje się bardziej zwarty zapis :

$$q = \{\eta, \epsilon\} \quad (2)$$

gdzie $\eta \in \mathbb{R}$ i jest składową skalarną, $\epsilon = [\epsilon_1, \epsilon_2, \epsilon_3]^T \in \mathbb{R}^3$ i jest składową wektorową kwaternionu.

Przy użyciu kwaternionu można opisać dowolną rotację obiektu, ponieważ każdą serię obrotów można wyrazić jako pojedynczy obrót wokół danej osi. Poniżej znajduje się schemat ilustrujący takie wykorzystanie kwaternionów. Orientacja obiektu związanego z układem $B(x_B, y_B, z_B)$ względem układu odniesienia $A(x_A, y_A, z_A)$, może zostać wyrażona jako obrót o kąt θ , wokół wektora r , opisanego w układzie odniesienia A .



Rys. 7 Przykład zastosowania kwaternionu do opisu orientacji [10].

Kwaternion opisujący taki obrót zdefiniowany jest w następujący sposób :

$${}^A_B q = \left\{ \cos \frac{\theta}{2}, r_x \sin \frac{\theta}{2}, r_y \sin \frac{\theta}{2}, r_z \sin \frac{\theta}{2} \right\} \quad (3)$$

Sprzężenie kwaternionów może zostać wykorzystane do opisu orientacji obiektu związanego z układem A względem układu B :

$${}^A_B q^* = {}^B_A q = \left\{ \cos \frac{\theta}{2}, -r_x \sin \frac{\theta}{2}, -r_y \sin \frac{\theta}{2}, -r_z \sin \frac{\theta}{2} \right\} \quad (4)$$

W celu wykonania obrotu wektora v przy użyciu kwaternionu ze składową skalarną równą 0, należy wykonać:

$${}^B v = {}^A_B q * {}^A v * {}^A_B q^* \quad (5)$$

Iloczyn kwaternionów może zostać wykorzystanych do składania obrotów. Na przykład, dla dwóch orientacji opisanych przy użyciu ${}^A_B q$ i ${}^B_C q$, złożenie dwóch obrotów można wyznaczyć w następujący sposób:

$${}^A_C q = {}^B_C q * {}^A_B q \quad (6)$$

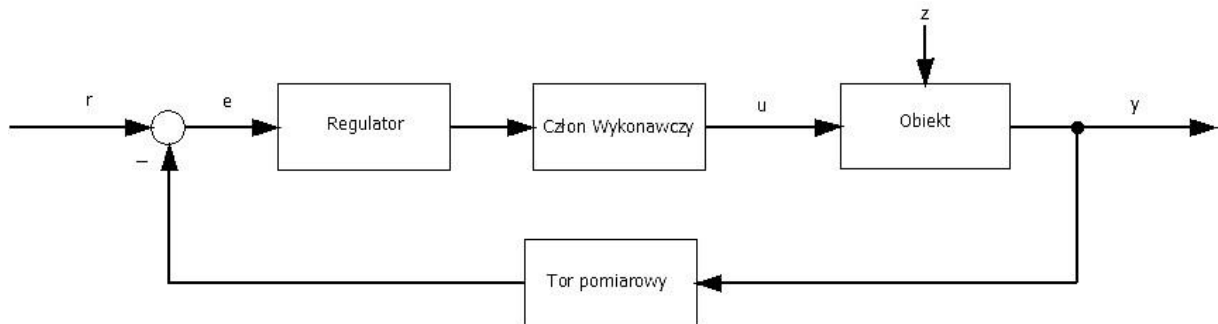
Ze względu na problem z interpretacją orientacji wyrażonej w postaci kwaternionu, często orientacja reprezentowana jest w postaci kątów *roll*, *pitch*, *yaw*. Konwersja kwaternionu ${}^A_B q$ na kąty RPY:

$$\begin{bmatrix} roll \\ pitch \\ yaw \end{bmatrix} = \begin{bmatrix} \text{atan2}(2(\eta\varepsilon_1 + \varepsilon_2\varepsilon_3), 1 - 2(\varepsilon_1^2 + \varepsilon_2^2)) \\ \arcsin(2(\eta\varepsilon_2 - \varepsilon_3\varepsilon_1)) \\ \text{atan2}(2(\eta\varepsilon_3 + \varepsilon_2\varepsilon_1), 1 - 2(\varepsilon_2^2 + \varepsilon_3^2)) \end{bmatrix} \quad (7)$$

2.4 Sterowanie systemów dynamicznych

2.4.1 Zamknięty układ regulacji

Podstawowym celem sterowania jest zapewnienie pożądanego stanu obiektu lub pożądanego przebiegu procesu w układzie. Ze względu na charakter procesu sterowania platformą latającą, konieczne jest zastosowanie układu automatycznej regulacji ze sprzężeniem zwrotnym. Zastosowanie ujemnego sprzężenia zwrotnego umożliwia osiągnięcie celu sterowania, pomimo występujących zakłóceń. Mogą to być zakłócenia związane z pomiarami wyjścia, zakłócenia działające na obiekt lub zakłócenia sterowania. Schemat prostego układu regulacji jest przedstawiony na rys. 8. Zasada działania układu ze sprzężeniem zwrotnym polega na porównaniu aktualnej wartości wielkości regulowanej y z wartością zadaną r , wyznaczeniu wartości uchybu regulacji e i pobudzenia do pracy regulatora, którego akcja redukuje błąd regulacji, pomimo ciągle działających zakłóceń z [11].



Rys. 8 Schemat zamkniętego układu regulacji z ujemną pętlą sprzężenia zwrotnego. Opis sygnałów: r – wartość zadana, e – uchyb regulacji, u – sterowanie generowane przez regulator, y – wyjście z obiektu, z – zakłócenia działające na obiekt.

Zarówno struktura regulatora, jak i jego nastawy dobierane są w taki sposób, aby jak najlepiej sprowadzały uchyb regulacji do zera. Błąd regulacji może występować w trakcie wykonywania różnych zadań [11]:

- Nadążania za planowaną zmianą wartości zadanej – regulacja nadążna
- Stabilizacji obiektu wokół wybranego punktu pracy – regulacja stałowartościowa

W odniesieniu do sterowania platformą latającą, regulacja nadążna odpowiada przejściu do nowego punktu pracy za sprawą zmiany wartości zadanej jednego z kątów *roll*, *pitch* lub *yaw*. Natomiast regulacja stałowartościowa odpowiada sytuacji, w której platforma stabilizuje się w zadanym punkcie pracy, pomimo występujących zakłóceń związanych z warunkami atmosferycznymi czy zakłóceniami sterowania.

2.4.2 Regulator PID

Regulator to jeden z głównych elementów wchodzących w skład układu regulacji. Zadanie regulatora polega na wypracowaniu takiego sygnału sterującego, aby obiekt osiągnął wartość zadaną w możliwie najkrótszym czasie.

Regulator *PID* (z ang. *Proportional-Integral-Derivative*) to algorytm regulacji ciągłej, składający się z liniowej kombinacji trzech członów, proporcjonalnego, różniczkującego i całkującego. Zaproponowany w 1922 roku przez Nicolasa Minorskiego, zyskał dużą popularność w układach automatycznej regulacji i jest z powodzeniem wykorzystywany do dzisiaj [12]. Regulator *PID* wyznacza sterowanie proporcjonalne do wartości błędu, proporcjonalne do wartości całki błędu, od momentu rozpoczęcia pracy aż do chwili obecnej, oraz proporcjonalne do aktualnej pochodnej błędu.

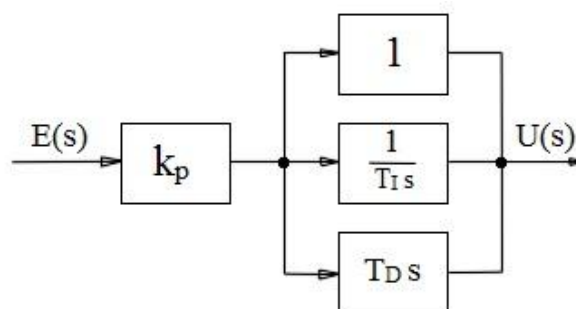
Sygnał sterujący w idealnym regulatorze PID-ISA (z ang. *Ideal Standard Algorithm*) :

$$u(t) = k_p \left(\varepsilon(t) + \frac{1}{T_I} \int_0^t \varepsilon(\tau) d\tau + T_D \frac{d\varepsilon(t)}{dt} \right) \quad (8)$$

gdzie k_p -współczynnik wzmocnienia, T_I - czas zdwojenia(całkowania), T_D - czas wyprzedzenia(różniczkowania), $\varepsilon(t)$ -sygnał błędu, $u(t)$ - sygnał sterujący

Transmitancja operatorowa idealnego regulatora *PID-ISA* :

$$G_R = k_p \left(1 + \frac{1}{T_I s} + T_D s \right) \quad (9)$$



Rys. 9 Schemat idealnego regulatora PID [13].

W cyfrowych implementacjach algorytmu *PID* wykorzystywane są różne metody wyznaczania całki i pochodnej sygnału błędu e_k . Stosując regułę całkowania prostokątów (*forward*), można wykorzystać dwa algorytmy *PID*, *pozycyjny* i *przyrostowy* [14]. Algorytm przyrostowy wykorzystywany jest częściej ze względu na brak konieczności znajomości błędu we wszystkich krokach poprzedzających. Aby wyznaczyć wartość sterowania, wymagana jest znajomość trzech ostatnich wartości błędów oraz ostatnia wartość sterowania.

$$u_k = u_{k-1} + b_0 e_k + b_1 e_{k-1} + b_2 e_{k-2} \quad (10)$$

gdzie : $b_0 = K \left(1 + \frac{T_D}{T}\right)$, $b_1 = -K \left(1 + 2 \frac{T_D}{T} - \frac{T_I}{T}\right)$, $b_2 = K \frac{T_D}{T}$, T - okres próbkowania

Dobór nastaw regulatora *PID* wiąże się z wyznaczeniem optymalnych wartości poszczególnych członów. Istnieje kilka metod strojenia regulatorów *PID*, oto niektóre z nich:

- I metoda Zieglera-Nicholsa – metoda odpowiedzi skokowej
- II metoda Zieglera-Nicholsa – metoda cyklu granicznego
- Metoda przekątnikowa Astroma i Hagglunda

W praktyce istnieje kilka wymagań stawianych przed takimi metodami. Prostota eksperymentu odgrywa duże znaczenie. Konieczność dokładnej znajomości modelu może dyskwalifikować metodę w przypadku skomplikowanych układów regulacji. Ważne jest również, aby potrzebne parametry były łatwe do zarejestrowania i zmierzenia. Często wymagane jest również to, aby eksperyment nie zaburzał procesu.

Dynamika odpowiedzi układu zależy od dobranych nastaw regulatora. Aby ocenić jakość dobranych nastaw wprowadzono kryteria jakości regulacji, wyznaczane na podstawie przebiegów regulacji [15]. Najpopularniejsze z nich:

- Całka z kwadratu uchybu regulacji - $J = \int_0^{\infty} e(t)^2 dt$, gdzie $e(t)$ - uchyb regulacji
- Czas regulacji – najmniejszy czas, począwszy od którego sygnał wyjściowy $y(t)$ wchodzi w 5-procentowe otoczenie wartości zadanej $r(t)$
- Wartość przeregulowania – odniesiona do wartości zadanej, maksymalna wartość dodatniej odchyłki sygnału wyjściowego $y(t)$
- Wielkość uchybu ustalonego - $e_{\infty} = \lim_{t \rightarrow \infty} |r(t) - y(t)|$

3. Budowa platformy typu quadcopter

3.1 Wykaz podzespołów

Platforma, którą zdecydowałem się zbudować, to typowy quadcopter przeznaczony do celów modelarskich. Kompaktowe rozmiary i rozsądny czas lotu, przy użyciu w pełni naładowanego pakietu zasilającego, to główne kryteria, którymi kierowałem się przy doborze podzespołów. Nie bez znaczenia okazała się dostępność podzespołów, jak również ich cena. W dalszej części rozdziału zaprezentowane zostały poszczególne podzespoły wraz z krótką charakterystyką.

Ostatecznie podzespoły, które użyłem do budowy platformy, to :

- Rama TAROT FY-450 V2
- 4 silniki SunnySky X2212-13
- 4 regulatory ESC ABC-Power SimonK 30A
- 4 śmigła APC MR 10x4,5MRP
- Pakiet zasilający Turnigy 3S/2200mAh – Li-pol
- Aparatura modelarska FlySky FS-CT6B

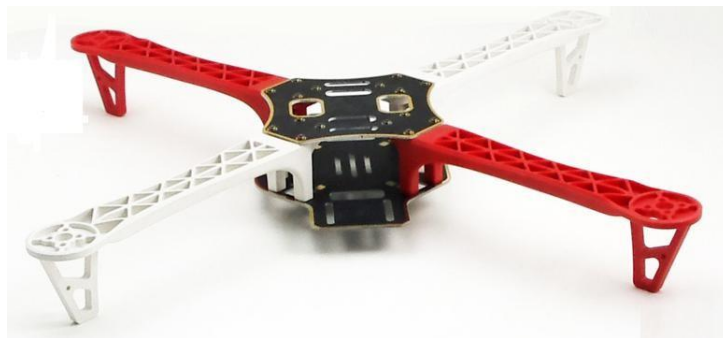
3.2 Opis podzespołów

Rama TAROT FY-450 V2

Lekka rama do platformy typu Quadrocopter, wykonana z laminatu szklano epoksydowego, zapewnia odpowiednią sztywność i wytrzymałość konstrukcji.

Tab3. Wybrane parametry ramy TAROT FY-450 V2 :

<i>Parametr</i>	<i>Wartość</i>	<i>Jednostka</i>
Rozstaw silników	450	mm
Masa własna	282	g
Masa startowa	800~1200	g
Maksymalny rozmiar śmigła	10	cal



Rys. 10 Rama Tarot FY-450 V2 [16].

Silnik SunnySky 2212-13

Silniki oferowane przez firmę SunnySky charakteryzują się wysoką precyzją wykonania. Wirniki wykonywane przez Kawasaki, magnesy wysokotemperaturowe czy japońskie łożyska o niskim współczynniku tarcia sprawiają, że silniki SunnySky to produkty wysokiej jakości.

Tab4. Wybrane parametry silnika SunnySky 2212-13 :

<i>Parametr</i>	<i>Wartość</i>	<i>Jednostka</i>
Obroty	980	rpm/V
Masa	84	g
Zasilanie	11.1/14.8	V
Moc max	146	W



Rys. 11 Silnik SunnySky 2212-13.

Regulator ESC ABC-Power SimonK 30A

Regulatory ESC (z ang. *Electronic Speed Controller*) dla silników bezszczotkowych to urządzenia regulujące pracę trójfazowych silników elektrycznych. Regulatory posiadają specjalne oprogramowanie dla kontrolerów do multikopterów.

Tab5. Wybrane parametry regulatora ESC ABC-Power SimonK 30A :

<i>Parametr</i>	<i>Wartość</i>	<i>Jednostka</i>
Prąd znamionowy	30	A
Napięcie pracy	7.4/14.8	V
Zakres częstotliwości PWM	50~400	Hz



Rys. 12 Regulator ESC SimonK 30A [16].

Śmigła APC MR 10x4.5 MRP

Platforma typu quadcopter wyposażona jest w dwa rodzaje śmigieł. Jedne przeznaczone do montażu na silnikach obracających się zgodnie z ruchem wskazówek zegara, oznaczone jako śmigła typu CW (z ang. *ClockWise*). Drugi rodzaj to śmigła obracające się przeciwnie do ruchu wskazówek zegara, tzw. śmigła CCW (z ang. *Counter-ClockWise*).

Zdecydowałem się na zakup śmigieł APC ze względu na dokładność wykonania oraz brak problemów z wyważaniem śmigieł.

Tab6. Wybrane parametry regulatora śmigła APC MR 10x4.5 MRP :

<i>Parametr</i>	<i>Wartość</i>	<i>Jednostka</i>	<i>Informacje dodatkowe :</i>
Średnica/skok śmigła	10x4.5	cal	<ul style="list-style-type: none"> Niski poziom hałasu Duża odporność na uszkodzenia



Rys. 13 Śmigło APC 10x4.5 [16].

Pakiet zasilający Turnigy 3S/2200mAh - Li-pol

Wybór pakietu zasilającego został podyktowany zachowaniem odpowiedniego kompromisu pomiędzy wagą akumulatora a pojemnością prądową. Akumulatory litowo-polimerowe charakteryzują się dużą wydajnością prądową.

Tab7. Wybrane parametry pakietu zasilającego Turnigy 3S/2200mAh :

<i>Parametr</i>	<i>Wartość</i>	<i>Jednostka</i>
Pojemność	2200	mAh
Liczba ogniw	3	-
Wydajność prądowa	55	A
Masa	188	g



Rys. 14 Pakiet zasilający 3S/2200 mAh [16].

Aparatura modelarska FlySky FS-CT6B

Aparatura FlySky FS-CT6B doskonale nadaje się do sterowania quadrocoptrem. Do dyspozycji użytkownika jest 6 kanałów, w zakresie 2.4-2.48 GHz.

Tab8. Wybrane parametry aparatury modelarskiej FlySky FS-CT6B:

<i>Parametr</i>	<i>Wartość</i>	<i>Jednostka</i>	<i>Informacje dodatkowe :</i>
Zakres RF	2.40-2.48	GHz	<ul style="list-style-type: none"> • Modulacja GFSK • Pamięć do ośmiu modeli • Wysoka czułość
Zasięg	1	km	
Zasilanie	12	V	
Liczba kanałów	6	-	

4. Implementacja sterownika kontroli lotu

4.1 Sterownik kontroli lotu – wprowadzenie

Elektroniczny kontroler lotu jest niezbędnym wyposażeniem każdego wielowirnikowca. Bardzo trudno wyobrazić sobie pilota, który byłby w stanie utrzymać platformę latającą w powietrzu przy pomocy aparatury modelarskiej, sterując każdym silnikiem z osobna. Kontroler lotu, który zdecydowałem się zbudować, umożliwia sterowanie quadropterem poprzez zadawanie wartości kątów (*roll*, *pitch*, *yaw*), opisanych w rozdziale drugim. Głównym zadaniem sterownika jest stabilizacja platformy w pozycji zadawanej z aparatury RC.

Sterownik kontroli lotu został zbudowany w oparciu o :

- Mikrokontroler Texas Instruments TM4C123GH6PM
- Inercyjną jednostkę pomiarową Pololu AltIMU-10
- Przetwornicę step-down Pololu-D24V50F5 5V, 5A
- Odbiornik RC FS-R6B 2.4 GHz
- Dedykowany obwód drukowany

Program do kontroli lotu został napisany w języku C, który jest bardzo popularnym narzędziem do programowania. Stworzony w 1972 roku przez Dennisa Ritchie'go z firmy Bell Labs w trakcie prac prowadzonych nad systemem operacyjnym UNIX [17]. Główne zalety języka C to duża elastyczność, przenośność oraz wydajność. Znajduje duże zastosowanie w programowaniu systemów operacyjnych oraz innych zadań niskiego poziomu.

Prace nad oprogramowaniem do sterownika kontroli lotu wykonałem przy użyciu zintegrowanego środowiska programistycznego *KEIL μ Vision v.4*. To środowisko umożliwia programowanie szerokiej gamy mikrokontrolerów opartych na procesorach *ARM*, *Cortex-M* i *Cortex-R*. Ponadto oferuje użytkownikom szeroką gamę narzędzi programistycznych, takich jak wbudowany debugger, emulator oraz kompilator *ARM C/C++* [18].

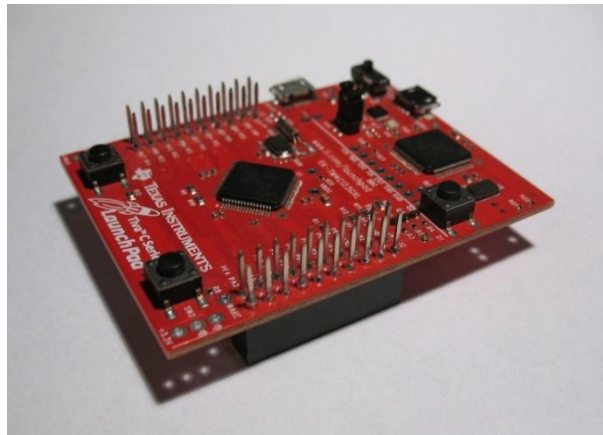
4.2 Sterownik kontroli lotu – komponenty

4.2.1 Texas Instruments Tiva C-Series

Sterownik kontroli lotu został zbudowany w oparciu o zestaw uruchomieniowy firmy Texas Instruments Tiva C-Series EK-TM4C123GXL, wyposażony w mikrokontroler TM4C123GH6PM z rdzeniem Cortex-M4 opracowanym przez ARM. Zdecydowałem się na takie rozwiązanie z kilku powodów. Mikrokontroler oferowany przez TI oferuje wysoką wydajność i niski pobór prądu. Szybka obsługa przerwań w połączeniu z dużą dokładnością i szybkością obliczeniową 32-bitowego procesora sprawiają, że jest bardzo solidną podstawą do budowy sterownika kontroli lotu. Sam zestaw uruchomieniowy charakteryzuje się kompaktowymi rozmiarami, niską wagą, wyprowadzeniem wszystkich niezbędnych portów I/O oraz łatwością programowania.

Parametry :

- Rdzeń Cortex-M4F
- Taktowanie procesora do 80MHz
- Wbudowany ICDI(z ang. *In-Circuit Debug Interface*)
- Pamięć Flash 256KB
- Pamięć SRAM 32KB
- Układ zasilany z USB lub z zewnętrznego źródła napięcia 5V
- Uniwersalne porty I/O
- Kontroler przerwań NVIC (z ang. *Nested Vectored Interrupt Controller*)
- Kontroler transmisji szeregowej(USB 2.0, SSI, I2C, UART, CAN)
- Kontroler DMA (z ang. *Direct Memmory Access*)
- Generator PWM
- Układy czasowo-licznikowe

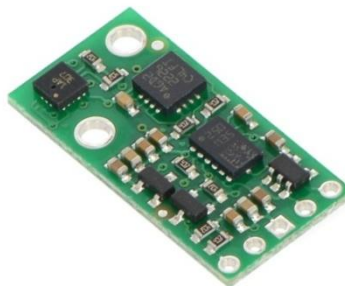


Rys. 15 Zestaw uruchomieniowy Texas Instruments TM4C123GLX.

4.2.2 Pololu AltIMU-10

Pololu AltIMU-10 to inercyjna jednostka pomiarowa, umożliwiająca pomiar przyspieszeń, prędkości kątowych, pola magnetycznego oraz wysokości. W jej skład wchodzi :

- Trzyosiowy żyroskop ST L3GD20 :
- Trzyosiowy akcelerometr ST LSM303DLHC
- Trzyosiowy magnetometr ST LSM303DLHC
- Barometr ST LPS331A



Rys. 16 Jednostka pomiarowa Pololu AltIMU-10 [20].

Wszystkie czujniki komunikują się po magistrali I²C (z ang. *Inter-Integrated Circuit*). Magistrala I²C jest dwukierunkowym, szeregowym interfejsem komunikacyjnym zaprojektowanym przez firmę Philips. Do transmisji danych wykorzystywane są dwie linie : SDA (z ang. *Serial Data Line*) – linia danych oraz SCL (z ang. *Serial Clock Line*) – linia sygnału zegarowego [19].

Moduł Pololu AltIMU-10 posiada wbudowany regulator napięcia oraz niezbędne elementy pasywne gwarantujące poprawne działanie magistrali I²C [20].

4.2.3 Przetwornica napięcia step-down Pololu- D24V50F5 5V, 5A

Regulator Pololu D24V50F5 to miniaturowa przetwornica konwertująca napięcie wejściowe, z zakresu od 6 do 38V, do napięcia wyjściowego 5V. Maksymalny pobór prądu to 5A.

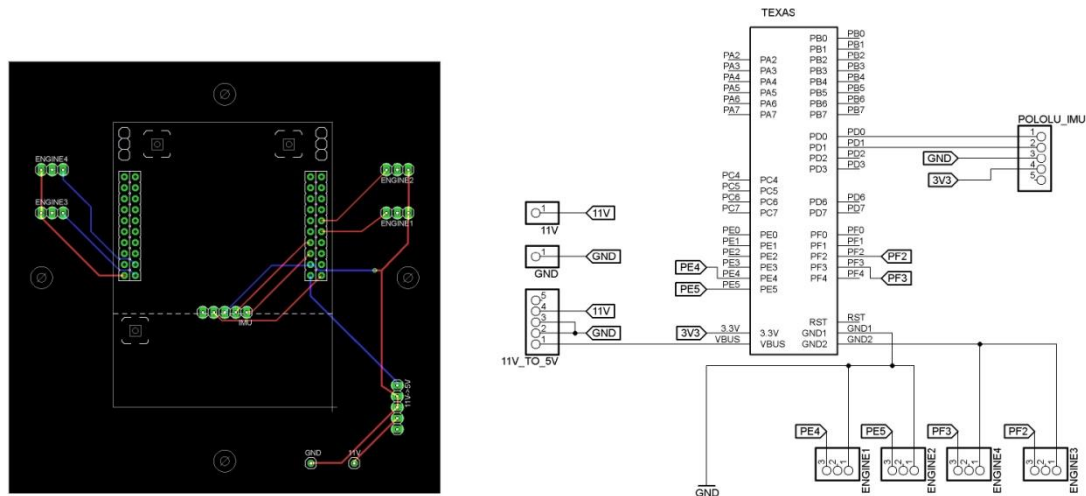
Wybór takiej przetwornicy podyktowany został przede wszystkim wysoką sprawnością układu, definiowaną jako stosunek mocy wyjściowej do mocy wejściowej. Wysoka sprawność jest istotnym parametrem w przypadku zasilania urządzeń przy użyciu akumulatorów lub baterii. Im wyższa sprawność, tym dłuższy czas cyklu pracy na pojedynczym ładowaniu. Producent deklaruje, że w przypadku układów zasilanych napięciem 12V, sprawność przetwornicy wynosi około 0.8-0.95 [21].

4.2.4 Dedykowany obwód drukowany

Schemat połączeń i projekt płytki wykonałem przy użyciu oprogramowania do projektowania PCB, Eagle 7.1.0. Dedykowany obwód drukowany to dwuwarstwowa płytka PCB o wymiarach 10x10cm, wykonana przez SATLAND Prototype [22].

Płytką stanowi podstawę, do której zamontowane zostały wszystkie urządzenia składające się na kontroler lotu. Jednostka pomiarowa IMU została na stałe przylutowana na środku płytki. Zestaw uruchomieniowy TI wpinany jest do męskiej listwy goldpin, z rastrem 2.54mm. W celu zachowania kompaktowych rozmiarów całego układu, zdecydowałem, że zestaw uruchomieniowy również zostanie zamontowany na środku, przesłaniając tym samym jednostkę pomiarową IMU, która przymocowana jest pod drugiej stronie płytki. Na płycie znalazły się listwy goldpin, umożliwiające montaż regulatora napięcia oraz podłączenie czterech przewodów od regulatorów ESC. Zamontowanie układu do ramy quadcoptera jest możliwe dzięki czterem otworom o średnicy 5mm.

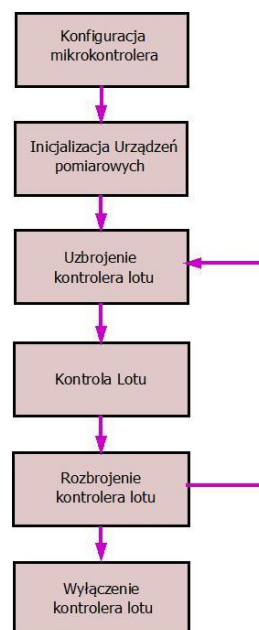
Zaprojektowany obwód zawiera wszystkie niezbędne połączenia elektryczne. Dzięki temu wszystkie urządzenia wchodzące w skład kontrolera lotu mogą zostać zamontowane bez użycia przewodów połączeniowych.



Rys. 17 Projekt płytki PCB oraz schemat połączeń.

4.3 Opis algorytmu

Algorytm sterowania można podzielić na kilka etapów wykonywanych sekwencyjnie, przedstawionych na rys.17. Rozpoczęcie pracy sterownika zaczyna się od inicjalizacji mikrokontrolera oraz jednostki pomiarowej IMU. Następnie kontroler lotu oczekuje na uzbrojenie sterownika przez pilota. Dalej wykonywany jest algorytm odpowiedzialny za stabilizację quadrocoptera. W przypadku rozbrojenia sterownika istnieje możliwość ponownego włączenia kontrolera lub wyłączenia quadrocoptera.



Rys. 18 Schemat pracy kontrolera lotu.

Konfiguracja mikrokontrolera związana jest z aktywacją portów I/O. W tej fazie inicjalizowane są również układy czasowo-licznikowe odpowiedzialne za sterowanie główną pętlą algorytmu kontroli lotu, odczyt sterowań oraz generowanie sygnałów sterujących PWM. Ostatnim etapem jest konfiguracja kontrolera transmisji szeregowej I2C, używanej do komunikacji z jednostką pomiarową IMU.

Następnie inicjalizowane są urządzenia pomiarowe. Każdy z czujników jest domyślnie wyłączony, więc należy je uruchomić, ustawiając odpowiednie bity w rejestrze *CTRL_REG_1*. Należy również ustawić zakresy pomiarowe dla każdego z czujników, dla akcelerometru $\pm 4g$, żyroskopu ± 250 dps, magnetometru $\pm 8.1Gs$.

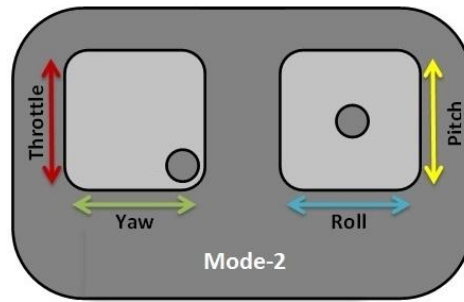
Pomyślne zakończenie etapu inicjalizacji urządzeń prowadzi do stanu oczekiwania na uzbrojenie sterownika. Ten etap polega na przytrzymaniu lewego drążka aparatury RC w pozycji przedstawionej na rys. 18 przez co najmniej dwie sekundy. Uzbrajanie sterownika jest formą zabezpieczenia przed przypadkowymi ruchami drążków, które mogą prowadzić do nieprzewidzianych i niebezpiecznych sytuacji. Uzbrojenie sterownika sygnalizowane jest zapaleniem diody, umieszczonej na mikrokontrolerze, oraz sygnałem dźwiękowym. Silniki quadcoptera zaczynają się kręcić z minimalną prędkością. Od tego momentu sterownik lotu zaczyna wykonywać operacje związane ze stabilizacją platformy, dzięki czemu quadcopter może zostać poderwany w powietrze przez pilota.

Rozpoczęcie sekwencji operacji związanych ze stabilizacją quadcoptera wyzwalane jest przerwaniem, pojawiającym się cyklicznie z częstotliwością 50Hz. Algorytm stabilizacji lotu działa na zasadzie układu zamkniętego z ujemnym sprzężeniem zwrotnym opisanym w pkt. 2.4.1. Pierwszy etap to wyznaczenie zadanej orientacji obiektu, podyktowanej przez pilota przy użyciu aparatury modelarskiej. Następnie w celu wyznaczenia błędu regulacji, należy również określić aktualną orientację platformy na podstawie danych odczytanych z jednostki pomiarowej IMU. Kolejny etap algorytmu stabilizacji to pobudzenie do pracy regulatorów. Ich zadanie polega na wyznaczeniu takich sygnałów sterujących, które ustabilizują quadcopter wokół zadanej punktu pracy.

4.4 Stabilizacja lotu

4.4.1 Zadawanie i odczyt sterowań

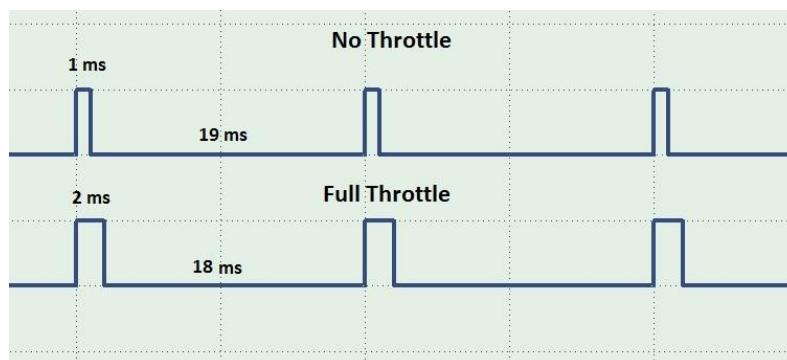
Do sterowania platformą wykorzystywana jest aparatura modelarska. Pilot zdalnie zadaje trzy wartości kątów: *roll*, *pitch*, *yaw*, oraz wartość *throttle*, poprzez odpowiednie wychylenie drążków. Zdecydowałem się na zastosowanie aparatury w układzie MODE-2. Zamieszczony poniżej schemat przedstawia układ drążków wraz z przypisanymi do nich wartościami zadanymi.



Rys. 19 Układ drążków w trybie Mode-2, w pozycji uzbrajania sterownika kontroli lotu [23].

Transmisja danych pomiędzy nadajnikiem zamontowanym w aparaturze, a odbiornikiem umieszczonym na quadcopterze wykorzystuje pasmo 2.4 GHz. Jest to tzw. system transmisji z rozproszonym widmem (z ang. *spread spectrum*). Dzięki temu wielu użytkowników może współdzielić to samo pasmo bez wzajemnego zagłuszania transmisji.

Odbiornik demoduluje sygnały transmitowane przez nadajnik do postaci sygnału PWM (z ang. *Pulse-width-modulation*). Jest to sygnał o stałej częstotliwości równej 50Hz i wypełnieniu zmieniającym się w zakresie od 5 do 10 %. Wartości zadane *roll*, *pitch*, *yaw* oraz *throttle* wyznaczane są na podstawie czterech sygnałów PWM.



Rys. 20 Przebieg sygnału sterującego wartością zadaną *throttle* : 0 % - przebieg górny, 100% - przebieg dolny.

Odpowiednie piny odbiornika połączone zostały z pinami GPIO mikrokontrolera. Ich wybór został podyktowany odpowiednim położeniem wyprowadzeń na płycie ewaluacyjnej, umożliwiającym zachowanie zwartej konstrukcji kontrolera lotu, bez konieczności prowadzenia zbędnych przewodów. Ostatecznie do komunikacji z odbiornikiem radiowym wykorzystalem:

- Wartość zadana *roll* – GPIO_PortA_P4
- Wartość zadana *pitch* – GPIO_PortB_P6
- Wartość zadana *yaw* – GPIO_PortC_P7
- Wartość *Throttle* – GPIO_PortD_P6

Odczyt wypełnienia sygnału PWM zrealizowałem przy użyciu przerwań od portów GPIO oraz czterech liczników (z ang. *timer*). Przerwania od portów wyzwalane są pojawieniem się zbocza narastającego i opadającego. Zbocze narastające powoduje uruchomienie licznika, który zaczyna zliczać czas stanu wysokiego z dokładnością 10 μ s. Taka konfiguracja licznika umożliwia wyznaczenie odczytywanej wartości z dokładnością 1%. W przypadku pojawienia się zbocza opadającego licznik jest wyłączany. Następnie od czasu, który zliczył, odejmowana jest wartość wynosząca 5 % okresu sygnału PWM. Wyznaczone w ten sposób sterowanie, zadane przez pilota, przypisywane jest do odpowiedniej zmiennej – *rollSetValue*, *pitchSetValue*, *yawSetValue* lub *throttle*. Wszystkie zmienne wykorzystane są na etapie wyznaczania sygnałów sterujących w pkt. 4.4.4.

4.4.2 Odczyt danych pomiarowych

Kolejnym etapem sekwencji stabilizującej quadrocopter jest odczyt danych pomiarowych. Komunikacja pomiędzy mikrokontrolerem a jednostką pomiarową IMU nawiązywana jest przy pomocy magistrali I2C w trybie *standard mode*, z prędkością przesyłania danych 100 kb/s. Mikrokontroler komunikuje się z jednostką pomiarową IMU wykorzystując cztery porty:

- Zasilanie układu *Vin* – GPIO_Port_3.3V
- Potencjał masy *GND* – GPIO_Port_GND
- Linia zegarowa *SCL* – GPIO_PortD_P0
- Linia danych *SDA* – GPIO_PortD_P1

Komunikacja działa na zasadzie *Master/Slave*. Mikrokontroler to urządzenie typu *Master*. Jest urządzeniem nadrzędnym, odpowiedzialnym za inicjowanie, prowadzenie oraz kontrolę transmisji.

Urządzenia pomiarowe zamontowane w jednostce pomiarowej IMU to urządzenia typu *slave*. Ich zadanie polega na prowadzeniu komunikacji z *masterem*, poprzez wysyłanie odpowiedzi na rozkazy lub zapytania urządzenia nadrzędnego. Każdy *slave* posiada unikatowy, 7-bitowy adres.

Tab. 1 Adresy urządzeń pomiarowych jednostki pomiarowej IMU

<i>Urządzenie</i>	<i>Adres</i>
Akcelerometr <i>LSM303DLHC</i>	0011001b
Magnetometr <i>LSM303DLHC</i>	0011110b
Żyroskop <i>L3GD20</i>	1101011b

Sekwencja odczytu danych z urządzenia *slave* inicjowana jest przez *mastera*, rozpoczyna się od ustawienia znacznika startu ST, wysłania 7-bitowego adresu SAD oraz bitu zapisu W.

Dalej *master* oczekuje na bit potwierdzenia SAK, wysyła adres rejestru SUB, z którego wartość chce odczytać i oczekuje na bit potwierdzenia SAK. Odczyt danych poprzedzony jest znacznikiem ponownego startu SR, adresem układu *slave* SAD oraz bitem odczytu R. *Slave* potwierdza odczyt instrukcji poprzez SAK i wysyła dane zapisane w rejestrze SUB. Po odebraniu całego bajtu *master* może zakończyć transmisję ustawiając NMAK i znacznik końca transmisji SP, lub ustawić bit potwierdzenia odebrania paczki danych MAK i oczekiwać na kolejny bajt danych przechowywanych pod adresem SUB+1.

Master	ST	SAD +W		SUB		SR	SAD +R			MAK		MAK		NMAK	SP
Slave			SAK		SAK			SAK	DATA		DATA		DATA		

Rys. 21 Przykład transmisji z użyciem magistrali I2C [24].

Przykładowo odczyt danych pomiarowych z 16-bitowego akcelerometru związany jest z odczytem 6 bajtów, z których pierwszy przechowywany jest w rejestrze o adresie 28h. Pierwsze dwa bajty składają się na mierzoną wielkość przyspieszenia w osi X, kolejne dwa dotyczą osi Y, a ostatnie dwa bajty składają się na wynik przyspieszenia w osi Z. Są to surowe wielkości, które trzeba przetworzyć w zależności od wybranego zakresu pomiarowego.

Przyjmując zakres pomiarowy $\pm 2g$, rozdzielczość akcelerometru wynosi 0.000061 g/LSB. Aby otrzymać przyspieszenie wyrażone w m/s^2 , wynik zwrócony z akcelerometru należy pomnożyć przez współczynnik skalujący akcelerometru:

$$\text{współczynnik_skalujący} = \text{rozdzielczość_akcelerometru} * \text{przyspieszenie_ziemskie} \quad (11)$$

W analogiczny sposób dokonywane są odczyty danych pomiarowych z żyroskopu i magnetometru.

Wszystkie dane pomiarowe zapisane są w trzech tablicach, z których każda przechowuje wartości pomiaru dla jednej z trzech osi odpowiedniego czujnika : *accelReadings[3]*, *gyroReadings[3]*, *magnetReadings[3]*. Wszystkie dane pomiarowe wykorzystywane są do wyznaczenia aktualnej orientacji quadcoptera.

4.4.3 Wyznaczenie orientacji

Znajomość położenia quadcoptera w przestrzeni trójwymiarowej jest niezbędna do zastosowania układu sterowania z zamkniętą pętlą sprzężenia zwrotnego.

W literaturze można znaleźć kilka filtrów służących do estymacji orientacji obiektu, na którym został zamontowany czujnik IMU. Zastosowanie filtru jest konieczne z kilku powodów.

W przypadku zastosowania akcelerometru możliwe jest wyznaczenie kąta, pod jakim znajduje się obiekt, względem pionu wyznaczonego przez wektor grawitacji. Koniecznym jednak jest, aby obiekt pozostał w bezruchu lub poruszał się ruchem jednostajnym. Jeśli te warunki nie są spełnione, pomiar z akcelerometru będzie zaburzony o wartość przyspieszenia ruchu,

który obiekt wykonuje. W takiej sytuacji nie jest możliwe odseparowanie wektora przyśpieszenia ziemskiego, co prowadzi do błędów wyznaczenia orientacji. Zakłócenia na wyjściu czujnika również prowadzą do pogorszenia dokładności wyznaczenia orientacji.

Orientacja obiektu może zostać wyznaczona w oparciu o dane pomiarowe żyroskopu. Żyroskop dostarcza informacji o prędkości kątowej obiektu. Scałkowanie prędkości kątowej jest równoznaczne z wyznaczeniem kąta obrotu obiektu wokół danej osi. Zakłócenia toru pomiarowego, oraz wrażliwość żyroskopu na warunki pracy, a szczególnie wpływ temperatury, powodują tzw. dryf żyroskopu. Dryf jest zjawiskiem niekorzystnym, związanym z ciągłym narastaniem błędu, w wyniku całkowania zakłóconych danych pomiarowych żyroskopu.

Odpowiednia fuzja danych z akcelerometru i żyroskopu prowadzi do wyznaczenia dokładnej orientacji obiektu. Często spotykanym rozwiązaniem jest zastosowanie filtra Kalmana. Filtr komplementarny również cieszy się dużą popularnością, głównie ze względu na prostotę implementacji.

W swojej pracy zdecydowałem się zastosować filtr opracowany przez Sebastiana Madgwicka [10]. Metoda ta umożliwia wyznaczenie orientacji obiektu, wyrażonej w postaci kwaternionów, w oparciu o nieprzetworzone dane pomiarowe. Jej główne zalety to duża dokładność statyczna i dynamiczna, nawet przy niskich częstotliwościach próbkowania, niewielkie zużycie zasobów procesora oraz korekcja zjawiska dryfu żyroskopowego. Dodatkowo optymalizacja działania filtra polega na dobraniu wyłącznie jednego parametru β .

Dane pomiarowe z żyroskopu $\omega_x, \omega_y, \omega_z$ mogą zostać wykorzystane do wyznaczenia pochodnej kwaternionu ${}^{BF}_{NF}\dot{q}_\omega$, która opisuje szybkość zmian orientacji układu nawigacyjnego NF , względem układu związanego z czujnikiem IMU BF :

$${}^{BF}_{NF}\dot{q}_\omega = \frac{1}{2} {}^{BF}_{NF}q * r_\omega \quad (12)$$

$$r_\omega = [0 \ \omega_x \ \omega_y \ \omega_z] \quad (13)$$

Wyznaczenie estymowanej orientacji ${}^{BF}_{NF}q_{est}$, przy pomocy algorytmu zaproponowanego przez Sebastiana Madgwicka, polega na numerycznym całkowaniu ${}^{BF}_{NF}\dot{q}_{est}$:

$${}^{BF}_{NF}q_{est,t} = {}^{BF}_{NF}q_{est,t-1} + {}^{BF}_{NF}\dot{q}_{est,t}\Delta t \quad (14)$$

Ze względu na dryf żyroskopowy, konieczna jest korekcja pochodnej ${}^{BF}_{NF}\dot{q}_{est}$, w kierunku przeciwnym do estymowanego błędu ${}^{BF}_{NF}\dot{q}_\epsilon$, wyznaczonego na podstawie pomiarów akcelerometru i magnetometru:

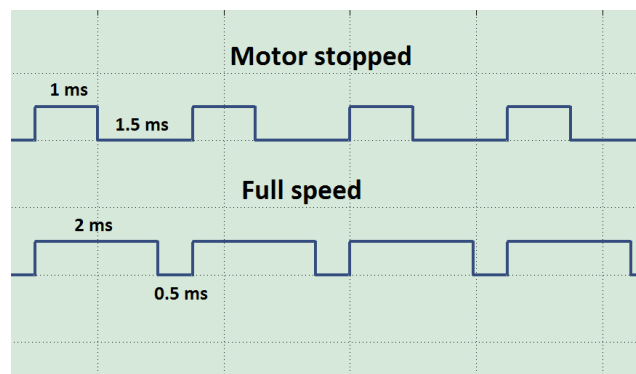
$${}^{BF}_{NF}\dot{q}_{est,t} = {}^{BF}_{NF}\dot{q}_{\omega,t} - \beta {}^{BF}_{NF}\dot{q}_{\epsilon,t} \quad (15)$$

Bardziej szczegółowy opis algorytmu Sebastiana Madgwicka dostępny jest [10]. Zaimplementowany algorytm w języku C, opublikowany na zasadach licencji *GNU General Public License*, jest dostępny [25].

Orientacja obiektu wyrażona w postaci kwaternionów jest mało intuicyjna, co jest nie bez znaczenia w przypadku przeprowadzania testów algorytmu kontroli lotu. Dodatkowo wartości zadane przedstawione są w postaci kątów Eulera. Dlatego konieczne jest przekształcenie kwaternionu do postaci kątów *RPY*, opisane w punkcie 2.3.3. Wyznaczone kąty zapisywane są do zmiennych *rollCurrnetValue*, *pitchCurrnetValue*, *yawCurrnetValue*, które wykorzystane są przy wyznaczaniu sygnałów sterujących.

4.4.4 Wyznaczenie sygnałów sterujących

Ostatnim etapem algorytmu stabilizacji lotu jest wyznaczenie sygnałów sterujących prędkością obrotową silników, które zapewnią osiągnięcie i utrzymanie stabilnej pozycji wokół zadanego punktu pracy, wyrażonego przy pomocy trzech kątów *RPY*: *rollSetValue*, *pitchSetValue*, *yawSetValue*. Urządzeniami pośredniczącymi w tym etapie są regulatory ESC, które regulują pracę trójfazowych silników elektrycznych w zależności od sygnału sterującego PWM, podawanego z mikrokontrolera. Wysoki stan sygnału utrzymywany jest przez czas mieszczący się w przedziale od 1ms do 2ms, analogicznie jak w przypadku odczytu danych z odbiornika RC. Częstotliwość generowanego sygnału to 400 Hz, więc w tym przypadku pełna skala sterowania zawiera się w przedziale od 40 do 78 % wypełnienia sygnału PWM.



Rys. 22

Przebieg sygnału sterującego prędkością obrotową silnika : 0 r/min – przebieg górny, ~10000 r/min – przebieg dolny.

Do generowania czterech sygnałów sterujących silnikami, wykorzystywany jest wbudowany w mikrokontroler generator PWM, który może wygenerować do ośmiu niezależnych przebiegów równocześnie [26]. Ustawienie wypełnienia sygnału związane jest z wpisaniem do rejestru komparatora czasu trwania stanu wysokiego, wyrażonego w taktach zegara (z ang. *clock ticks*). Dla procesora taktowanego zegarem 40 MHz, przedział 40-78 % wypełnienia sygnału PWM, odpowiada ustawieniu wartości *PWM_Width*, w rejestrze komparatora, z zakresu 40000-78000.

Sygnały PWM sterujące silnikami wystawne są na następujące porty mikrokontrolera:

- Sterowanie silnikiem nr 1 - GPIO_PortE_P4
- Sterowanie silnikiem nr 2 - GPIO_PortE_P5
- Sterowanie silnikiem nr 3 - GPIO_PortF_P2
- Sterowanie silnikiem nr 4 - GPIO_PortF_P3

Program kontroli lotu wykorzystuje cztery zmienne, *Engine1Speed*, *Engine2Speed*, *Engine3Speed* oraz *Engine4Speed* odpowiedzialne za sterowanie prędkością obrotową silników. Wszystkie zmienne przyjmują wartości z zakresu 0-100 % i wyznaczone są w następujący sposób:

$$\begin{bmatrix} Engine1Speed \\ Engine2Speed \\ Engine3Speed \\ Engine4Speed \end{bmatrix} = \begin{bmatrix} throttle + pitchPID + yawPID \\ throttle + rollPID - yawPID \\ throttle - pitchPID + yawPID \\ throttle - rollPID - yawPID \end{bmatrix} \quad (16)$$

Wartość *throttle* to wielkość odpowiedzialna za utrzymywanie quadrocoptera na odpowiedniej wysokości, zadawana ręcznie przez pilota.

Każda z wartości *rollPID*, *pitchPID* oraz *yawPID* wyznaczana jest przez mikrokontroler, w oparciu o zastosowanie pętli sprzężenia zwrotnego oraz regulator *PID*.

Działanie układu zamkniętego jest analogiczne dla wszystkich trzech kątów *RPY*. Poniżej przedstawiony zostanie schemat oraz zasada działania układu regulacji dla kąta *roll*.

Zaimplementowany regulator *PID* to regulator *przrostowy*. Zastosowanie takiego rozwiązania wymaga znajomości poprzedniej wartości sterowania *previousRollPID*, oraz wartości błędów w trzech poprzednich iteracjach algorytmu *rollError1*, *rollError2* oraz *rollError3*.

W pierwszej kolejności aktualizowane są niezbędne zmienne:

$$previousRollPID = rollPID \quad (17)$$

$$rollError3 = rollError2 \quad (18)$$

$$rollError2 = rollError1 \quad (19)$$

$$rollError1 = rollError \quad (20)$$

Dalej wyznaczany jest uchyb aktualny regulacji *rollError*:

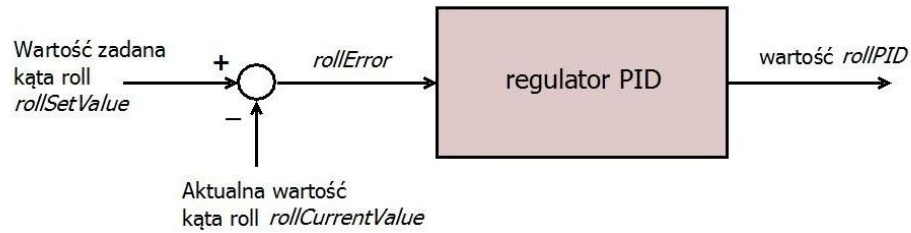
$$rollError = rollSetValue - rollCurrentValue \quad (21)$$

Następnie regulator *PID* wyznacza wartość sterowania *rollPID*:

$$rollError = previousRollPID + b_0 rollError1 + b_1 rollError2 + b_2 rollError3 \quad (22)$$

gdzie: $b_0 = K_p \left(1 + \frac{T_D}{T}\right)$, $b_1 = -K_p \left(1 + 2\frac{T_D}{T} - \frac{T_I}{T}\right)$, $b_2 = K_p \frac{T_D}{T}$, T - okres próbkowania, K_p - wzmocnienie członu proporcjonalnego, K_i - wzmocnienie członu całkującego, K_d - wzmocnienie członu różniczkującego

Struktura regulatora PID dla osi roll



Rys. 23 Stabilizacja platformy wokół zadanego kąta roll.

Ostatnim etapem wyznaczania sygnałów sterujących jest przeskalowanie zmiennych *EngineSpeed*, które przyjmują wartości z zakresu 0-100 % tak, aby do rejestru komparatora generatora PWM wpisywać wartości z zakresu 40000-78000:

$$\begin{bmatrix} PWM_Width1 \\ PWM_Width2 \\ PWM_Width3 \\ PWM_Width4 \end{bmatrix} = \begin{bmatrix} 40000 + 380 * Engine1Speed \\ 40000 + 380 * Engine2Speed \\ 40000 + 380 * Engine3Speed \\ 40000 + 380 * Engine4Speed \end{bmatrix} \quad (23)$$

Wyznaczone w ten sposób sygnały sterujące, w połączeniu z odpowiednimi nastawami regulatorów, zagwarantują osiągnięcie i utrzymanie stabilnej pozycji quadrocoptera w powietrzu.

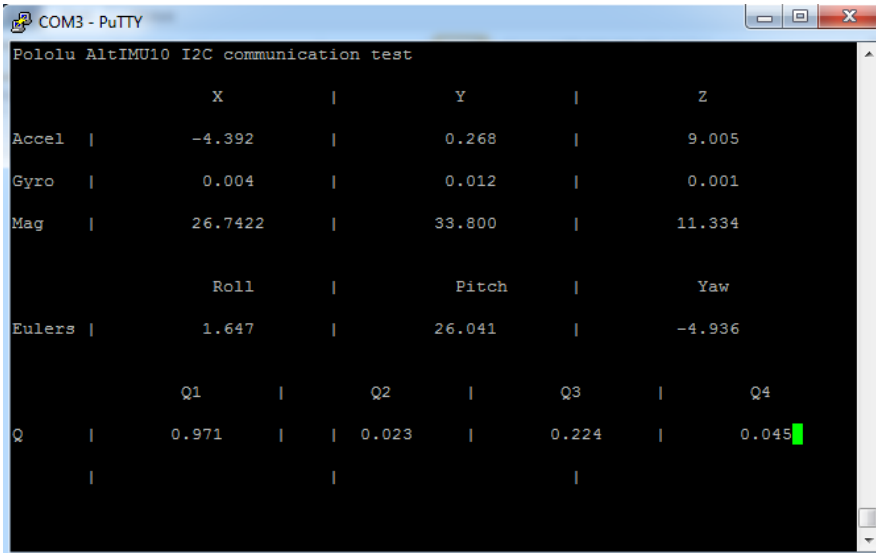
5. Testy i dobór parametrów

5.1 Komunikacja z jednostką pomiarową IMU

Komunikacja z czujnikiem Pololu AltIMU10 nawiązywana jest przy użyciu magistrali I2C. Poprawna wymiana danych uwarunkowana jest odpowiednią inicjalizacją wszystkich urządzeń pomiarowych.

Testy komunikacji wykonane zostały w początkowej fazie projektu, przy pomocy programu *PuTTY* oraz protokołu *UART* (z ang. *Universal Asynchronous Receiver and Transmitter*). *PuTTY* to bezpłatny emulator terminala tekstowego, który umożliwia połączenie z serwerem przy użyciu protokołów *TELNET*, *SSH* oraz *rlogin*. *UART* wykorzystany został do komunikacji pomiędzy mikrokontrolerem a *PuTTY*.

Testy komunikacji polegały na sprawdzaniu poprawności danych pomiarowych, wysyłanych z mikrokontrolera do terminala tekstowego. W dalszej fazie projektu, w ten sposób sprawdzane były również inne etapy pracy sterownika kontroli lotu, takie jak wyznaczenie orientacji, odczyt sterowań z aparatury RC oraz poprawność działania regulatorów *PID*.



```

COM3 - PuTTY
Pololu AltIMU10 I2C communication test

      X      |      Y      |      Z
Accel |    -4.392    |     0.268    |     9.005
Gyro  |     0.004    |     0.012    |     0.001
Mag   |    26.7422   |    33.800    |    11.334

      Roll    |     Pitch    |     Yaw
Euler |     1.647    |    26.041    |    -4.936

      Q1      |     Q2      |     Q3      |     Q4
Q      |     0.971    |     0.023    |     0.224    |     0.045
      |              |              |              |
      |              |              |              |
  
```

Rys. 24 Przykładowe dane pomiarowe wyświetlane w terminalu tekstowym.

5.2 Orientacja quadrocoptera

Testy algorytmu wyznaczania orientacji zostały przeprowadzone przy użyciu dedykowanego statywu, który umożliwia swobodny obrót quadrocoptera wokół jednej z osi, *roll* lub *pitch* w zakresie $\pm 30^\circ$.

Opracowanie wyników zostało przygotowane w oparciu o środowisko *MatlabR2014a*, oferowane przez *MathWorks*, które umożliwia wykonywanie różnorodnych obliczeń numerycznych, estetyczną wizualizację wyników, programowanie własnych algorytmów oraz pakiet obszernych bibliotek. W celu wizualizacji danych został napisany skrypt, który pozwala na

prezentację wyników w czasie rzeczywistym. Do komunikacji ze sterownikiem kontroli lotu został wykorzystany port szeregowy.

Przeprowadzone testy umożliwiły dobór parametru β , który odpowiedzialny jest za optymalizację działania algorytmu wyznaczania orientacji. Sebastian Madgwick przedstawił w swojej pracy wyniki testu dokładności estymowanej orientacji, w zależności od tego parametru. Testy wykazały, że optymalne działanie algorytmu zachodzi dla $\beta \in (0.03, 0.2)$ [10]. Przeprowadzone testy miały charakter statyczny i dynamiczny.

5.2.1 Testy statyczne

Testy statyczne polegały na sprawdzeniu wyznaczonych kątów *roll* oraz *pitch* dla nieruchomego quadcoptera, gdzie prędkość obrotowa silników zmieniała się z zakresu od 5 do 60 % prędkości maksymalnej. Wstępne eksperymenty wykazały, że quadcopter podrywany jest w powietrze przy prędkości obrotowej silników równej około 40 % prędkości maksymalnej. Dlatego też przyjęty przedział wydaje się obejmować zakres sterowań stabilizujących lot.

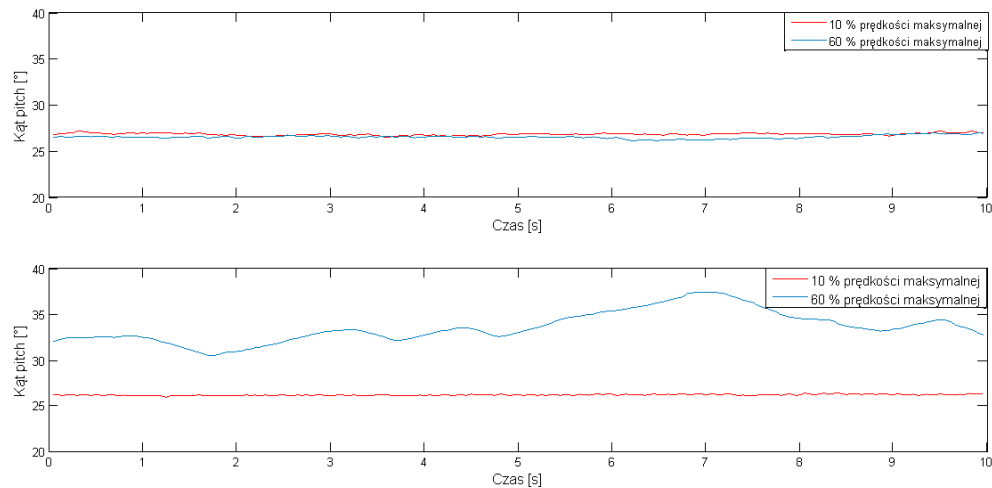
Zaprezentowane wyniki dotyczą przypadków, w których prędkość obrotowa wynosi 5% i 60 % prędkości maksymalnej. Takie zestawienie wyników pozwala ocenić działanie algorytmu, w sytuacji gdy na czujniki pomiarowe działają niskie oraz duże zakłócenia, związane z wibracjami generowanymi przez silniki.

Parametr β testowany był dla czterech przypadków. Na podstawie każdej próby wyznaczona została średnia wartość kąta $pitch_{sr}$ oraz odchylenie standardowe σ , które jest miarą rozrzutu obserwowanych wyników wokół wartości średniej. Wyniki eksperymentu zaprezentowano w tabeli nr 9.

Tab9. Wyniki testu statycznego, $pitch_{sr}$ - średnia wartość kąta *pitch*, σ – odchylenie standardowe pomiarów kąta *pitch*

β	$\omega = 10\% \omega_{max}$		$\omega = 60\% \omega_{max}$	
	$pitch_{sr}$	σ	$pitch_{sr}$	σ
0.2	26.37	0.33	29.20	1.58
0.1	26.36	0.20	27.51	0.46
0.05	26.82	0.12	26.51	0.41
0.04	26.22	0.09	26.69	1.26
0.035	26.21	0.07	33.57	1.67

Najlepsze rezultaty zostały osiągnięte dla $\beta = 0.1$ oraz $\beta = 0.05$. Dla tych przypadków zarówno σ przyjmowała najniższe wartości, jak i różnica pomiędzy wartością średnią kąta, dla skrajnych prędkości obrotowych, wynosiła najmniej. Dlatego do testów dynamicznych przyjęta została $\beta \in (0.05, 0.1)$.

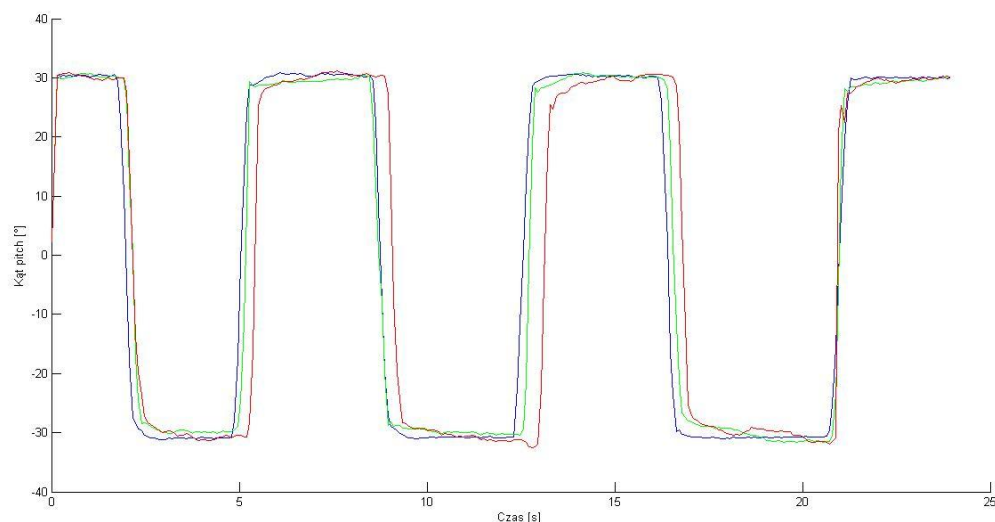


Rys. 25 Estymowana wartość kąta pitch : $\beta = 0.05$ – wykres górny, $\beta = 0.035$ – wykres dolny.

Na rys. 25 przedstawione zostały wyniki eksperymentów dla skrajnych wartości odchylenia standardowego. Różnica działania algorytmu wyznaczania orientacji, w zależności od parametru β jest wyraźna. Przebieg górny ilustruje poprawne działanie algorytmu, gdzie estymowane wartości kąta yaw praktycznie pokrywają się, a rozrzut pomiarów jest w niewielki. Na dolnym przebiegu różnica estymowanej wartości kąta yaw wynosi średnio około 7° . Wyznaczenie orientacji z taką dokładnością jest nie do zaakceptowania.

5.2.2 Testy dynamiczne

Testy dynamiczne algorytmu wyznaczania orientacji polegały na wychylaniu quadrocoptera do pozycji skrajnych, 30° i -30° . Zarejestrowane przebiegi pozwalają wyznaczyć ostateczną wartość parametru β .



Rys. 26 Przebieg testu dynamicznego : $\beta = 0.05$ – przebieg niebieski, $\beta = 0.08$ – przebieg zielony, $\beta = 0.1$ – przebieg czerwony.

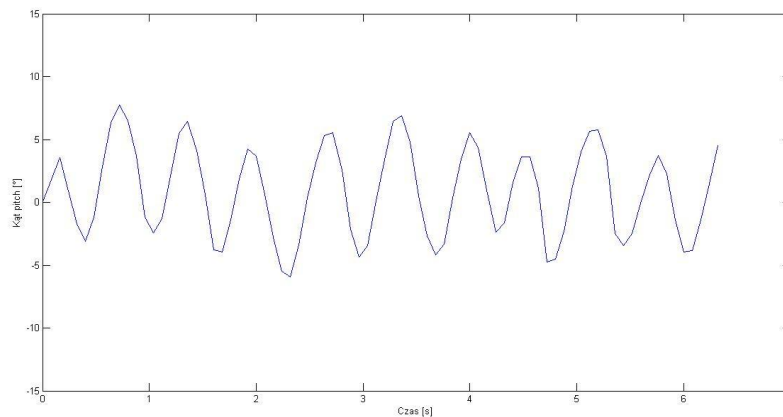
Ze względu na charakter testu, kształt przebiegu powinien przypominać falę prostokątną. Najlepszy efekt działania algorytmu można zaobserwować dla $\beta = 0.05$, gdzie widoczna jest

stabilna wartość kąta *pitch* przy niezmieniającej się pozycji $\pm 30^\circ$, oraz szybkie osiągnięcie dokładnej wartości kąta przy nagłej zmianie pozycji quadcoptera. W przypadku pozostałych przebiegów, estymowana wartość kąta *pitch* zarówno dla pozycji stabilnej, jak i szybkiej zmiany, jest wyraźnie gorsza. Dlatego też przyjąłem parametr $\beta = 0.05$.

5.3 Dobór nastaw regulatorów PID

Parametry regulatorów odpowiedzialnych za stabilizację quadcoptera zostały dobrane przy użyciu statywu, który wykorzystany był również do testów algorytmu wyznaczającego orientację. Statyw umożliwia dobranie parametrów dla dwóch regulatorów, gwarantujących osiągnięcie i utrzymanie pozycji z zadaną wartością kątów *roll* oraz *pitch*. Ze względu na analogię, przedstawię sposób postępowania przy doborze nastaw dla regulatora kąta *pitch*.

Pierwsza próba nastrojenia regulatora wiązała się z wykorzystaniem metody II metody Zieglera-Nicholsa, czyli tzw. metody cyklu granicznego. Metoda ta polega na wyznaczeniu takiego wzmocnienia w zamkniętym układzie regulacji, dla którego na wyjściu obiektu pojawia się oscylacje o stałej amplitudzie. Optymalne parametry regulatora *PID* można wyznaczyć w oparciu o znaną wartość wzmocnienia krytycznego oraz okresu oscylacji wyznaczonego na podstawie zarejestrowanego przebiegu.



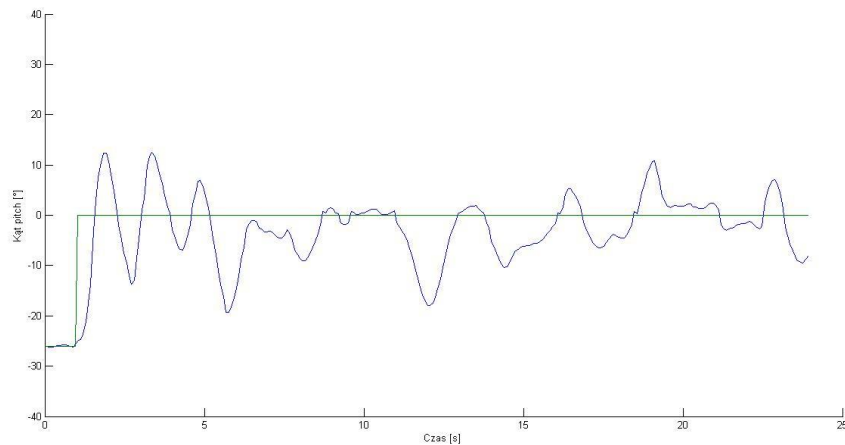
Rys. 27 Pierwszy etap II metody Zieglera-Nicholsa – wyznaczenie wzmocnienia krytycznego K_{kr} , okresu oscylacji T_{osc}

Sukcesywne zwiększanie wzmocnienia doprowadziło do oscylacji obiektu wokół wartości zadanej, dla wzmocnienia krytycznego $K_{kr} = 0.13$. Okres oscylacji wynosi $T_{osc} = 0.64$ [s]. Amplitudy drgań nie są identyczne dla kolejnych minimów/maksimów. Jednak zaprezentowany przebieg jest najlepszym przypadkiem, który udało się osiągnąć. Istotne jest to, że amplituda drgań nie rośnie w miarę upływu czasu.

Na podstawie [27] wyznaczone zostały optymalne parametry :

Tab. 10 Parametry regulatora kąta *pitch*, wyznaczone II metodą Zieglera-Nicholsa.

K_p	K_i	K_d
0.078	0.32	0.08



Rys. 28 Odpowiedź obiektu przy zmianie wartości zadanej kąta *pitch*, dla parametrów regulatora wyznaczonych metodą Zieglera-Nicholsa, przebieg zielony – wartość zadana, przebieg niebieski – wartość aktualna kąta *pitch*.

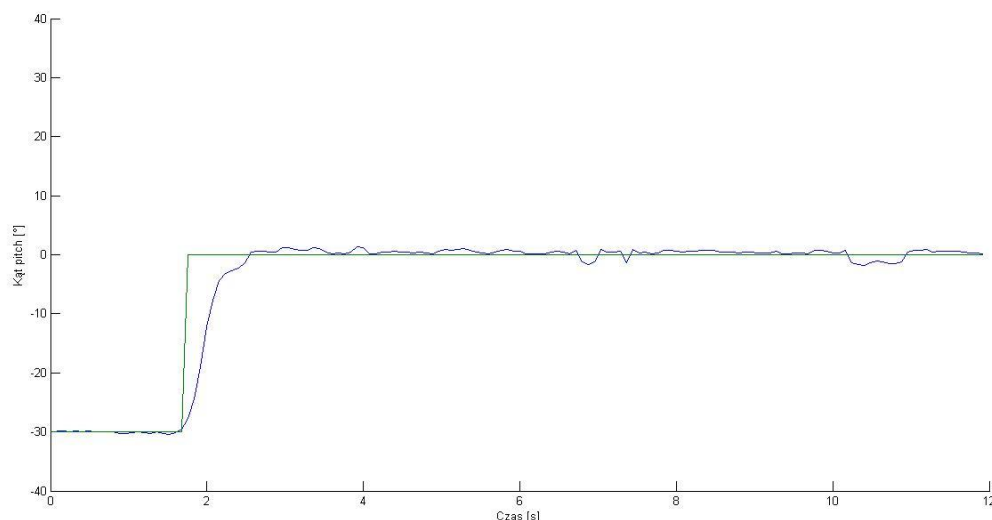
Zarejestrowana odpowiedź obiektu na zmianę wartości zadanej, o charakterze skoku jednostkowego, w sposób jednoznaczny odrzuca wyznaczone parametry regulatora kąta *pitch*. Bardzo duże przeregulowanie w połączeniu z ciągłymi oscylacjami wokół wartości zadanej, to nie są cechy opisujące poprawnie działający układ regulacji.

Druga próba nastrojenia regulatorów polegała na eksperymentalnym wyznaczeniu parametrów, w oparciu o odpowiedź quadcoptera na zmianę wartości zadanej, o charakterze skoku jednostkowego. Eksperyment zdecydowanie bardziej czasochłonny, ale przyniósł zaskakująco dobre rezultaty.

Ostateczne nastawy regulatorów kąta *roll* oraz *pitch* wynoszą :

Tab. 11 Parametry regulatora kąta *pitch*, wyznaczone eksperymentalnie.

	K_p	K_i	K_d
Kąt <i>roll</i>	0.22	0	0.36
Kąt <i>pitch</i>	0.22	0	0.37



Rys. 29 Odpowiedź obiektu przy zmianie wartości zadanej kąta pitch, dla parametrów regulatora wyznaczonych eksperymentalnie,
przebieg zielony – wartość zadana, przebieg niebieski – wartość aktualna kąta pitch.

Zarejestrowany przebieg świadczy o poprawnie dobranych parametrach regulatora. Krótki czas regulacji w połączeniu z niewielkimi, chwilowymi odchyleniami wartości aktualnej kąta, od wartości zadanej, powinny zagwarantować stabilny lot quadcoptera.

W rzeczywistości tak dobrane parametry pozwalają na bezpieczny lot quadcoptera nawet przy niekorzystnych warunkach atmosferycznych, tj. przy silnym, zmiennym wietrze. Przeprowadzone testy w otwartej przestrzeni wykazały, że quadcopter można stabilnie podnieść w powietrze, nawet jeśli ustawienie startowe nie jest idealnie poziome. Warto jednak zauważyć, że doświadczenie i umiejętności pilota, mają duży wpływ na bezpieczeństwo i powodzenie lotu.

6. Uwagi końcowe i wnioski

Celem projektu inżynierskiego przedstawionego w tej pracy, było wykonanie elektronicznego kontrolera lotu dla platformy latającej typu quadcopter. Pomimo kilku trudności napotkanych w trakcie realizacji projektu, zadanie zostało wykonane. W ramach projektu przygotowane zostały :

1. Kompletna platforma latająca typu quadcopter, zbudowana w oparciu o wybrane przeze mnie podzespoły, dostępne w sklepach specjalistycznych.
2. Dedykowany obwód drukowany, wykonany zgodnie z przygotowanym przeze mnie projektem, zapewniający integrację wszystkich urządzeń pokładowych. Umożliwia również montaż sterownika kontroli lotu na platformie latającej.
3. Warstwa aplikacyjna sterownika kontroli lotu, oparta na mikrokontrolerze Texas Instruments. Zadaniem funkcjonalnym oprogramowania jest kontrola lotu quadcoptera, sterowanego zdalnie.

Zastosowane rozwiązania umożliwiają stabilny lot platformy, kontrolowanej zdalnie przez pilota, przy użyciu aparatury modelarskiej.

Warto jednak zaznaczyć, że quadcopter to urządzenie, które może stanowić poważne niebezpieczeństwo. Z własnego doświadczenia wiem, że należy zachować najwyższe środki ostrożności przy pracach związanych z urządzeniami tego typu.

W trakcie prac nad projektem najpoważniejsze problemy sprawiły komunikacja z jednostką pomiarową IMU przy użyciu magistrali I2C, prawidłowe wyznaczenie orientacji quadcoptera oraz dobór nastaw regulatorów stabilizujących lot.

Początkowo zakładałem napisanie własnej biblioteki do obsługi magistrali I2C. Jednak ze względu na pojawiające się problemy z komunikacją, zdecydowałem się na zastosowanie biblioteki udostępnionej przez *Texas Instruments*, która umożliwia obsługę tej magistrali.

Problemy związane z wyznaczeniem prawidłowej orientacji quadcoptera polegały na błędnym wyznaczaniu wartości kąta *yaw*. Przyczyną był uszkodzony magnetometr, którego pomiar wartości pola magnetycznego wokół osi *x* był błędny. Wymiana czujnika rozwiązała problem. Należy również zwrócić uwagę na poprawną kalibrację czujników, zwłaszcza akcelerometru i żyroskopu.

Kontrola lotu quadcoptera wymaga użycia trzech niezależnych regulatorów wartości kątów – *roll*, *pitch* oraz *yaw*. Eksperymentalny dobór odpowiednich nastaw dla regulatorów stanowił trudne zadanie. Niemniej osiągnięty efekt jest w pełni zadowalający. Jedynym ograniczeniem tego rozwiązania jest to, że dobór parametrów regulatora kąta *yaw* możliwy jest tylko wtedy, gdy quadcopter poderwany jest w powietrze, lub może się swobodnie obracać wokół osi prostopadłej do powierzchni Ziemi. Dlatego też sterowanie wartością kąta *yaw* może nie być w pełni optymalne.

Sterowniki kontroli lotu dla multikopterów to zagadnienie, które w ostatnim czasie mocno zyskało na popularności. Temat zrealizowany w ramach projektu inżynierskiego stanowi dobrą podstawę do dalszej rozbudowy. Optymalizacja algorytmów sterowania w oparciu o model symulacyjny, czy próba implementacji autonomicznego algorytmu sterowania z zastosowaniem systemu operacyjnego czasu rzeczywistego w połączeniu z systemem wizyjnym, lub

odbiornikiem GPS, to tylko niektóre pomysły dalszego rozwoju projektu. Możliwości tego typu maszyn są ogromne, a ich granice zdaje się wyznaczać jedynie ludzka wyobraźnia.

7. Bibliografia :

- [1] **Krzysztof Kozłowski, Piotr Dutkiewicz, Waldemar Wróblewski** – *Modelowanie i Sterowanie Robotów*, str. 26, Wydawnictwo Naukowe PWM, Warszawa 2003
- [2] **Intelligent flight control of an autonomous quadrotor** – *sterowanie quadrocopterem, opracowanie własne na podstawie :*
<http://www.intechopen.com/books/motion-control/intelligent-flight-control-of-an-autonomous-quadrotor>
- [3] **Żyroskopy i akcelerometry MEMS w elektronice użytkowej**
<http://elektronikab2b.pl/technika/12098-zyroskopy-i-akcelerometry-mems-w-elektronice-uzytkowej#.VI3r-n8zPcc>
- [4] **Stmicroelectronics** – *materiały promocyjne - Motion MEMS, Smart multi-axis solutions for enhanced user experience*
- [5] **Embedded systems news** – *Witryna sieci Web -*
<http://embeddedsystemnews.com/stmicroelectronics-announces-l3gd20-new-three-axis-digital-output-gyroscope.html>
- [6] **Stmicroelectronics** – *Witryna sieci Web -* <http://www.st.com/web/en/press/en/p3460>
- [7] **John J. Craig** – *Wprowadzenie do robotyki : Mechanika i sterowanie*, str. 36, Wydawnictwo Naukowo-Techniczne, Warszawa 1995
- [8] **John J. Craig** – *Wprowadzenie do robotyki : Mechanika i sterowanie*, str. 459, Wydawnictwo Naukowo-Techniczne, Warszawa 1995
- [9] **Krzysztof Kozłowski, Piotr Dutkiewicz, Waldemar Wróblewski** – *Modelowanie i Sterowanie Robotów*, str. 29, Wydawnictwo Naukowe PWM, Warszawa 2003
- [10] **Sebastian O.H. Madgwick** – *An efficient orientation filter for inertial and inertial/magnetic sensor arrays*
- [11] **Witold Byrski** – *Obserwacja i sterowanie w systemach dynamicznych*, str. 180, Wydawnictwa AGH, Kraków 2007
- [12] **Witold Byrski** – *Obserwacja i sterowanie w systemach dynamicznych*, str. 193, Wydawnictwa AGH, Kraków 2007
- [13] **Asimo** – *Witryna sieci Web -*
http://www.asimo.pl/teoria/regulatory_dzialanie_ciagle.php
- [14] **Witold Byrski** – *Obserwacja i sterowanie w systemach dynamicznych*, str. 199, Wydawnictwa AGH, Kraków 2007
- [15] **J. Baranowski, K. Hajduk, A. Korytowski, W. Mitkowski, A. Tutaj** – *Teoria Sterowania – Materiały do ćwiczeń laboratoryjnych*, str. 82-84, Wydawnictwa AGH, Kraków 2007

- [16] **Internetowy sklep modelarski** – *Witryna sieci Web* – <http://abc-rc.pl>
- [17] **Stephen Prata** – *Szkola programowania – Język C*, str. 23-26, Helion, 2006
- [18] **Keil Embedded Development Tools** – *Witryna sieci Web* - <http://www.keil.com/uvision/>
- [19] **Paprocki Krzysztof** : *Mikrokontrolery STM32 w praktyce*, BTC, Legionowo, 2011
- [20] **POLOLU Robotics & Electronics** – *AltIMU10* <http://www.pololu.com/product/1269>
- [21] **POLOLU Robotics & Electronics** – *Step down voltage regulator* <http://www.pololu.com/product/2851>
- [22] **Satland Prototype** – *Producent układów drukowanych* http://prototypy.com/t/1,Strona_glowna
- [23] **Aparatura RC** – *Tryb mode-2, opracowanie własne na podstawie :* <http://aeroquad.com/showwiki.php?title=Using+the+transmitters+sticks+and+switches+to+operate+your+AeroQuad>
- [24] **Stmicroelectronics** – *LSM303DLHC 3D 3D accelerometer and 3D magnetometer datasheet*
- [25] **X-IO Technologies** - *Open Source IMU and AHRS Algorithms* - <http://www.x-io.co.uk/open-source-imu-and-ahrs-algorithms/>
- [26] **Texas Instruments** - *TM4C123GH6PM Microcontroller datasheet*
- [27] **Teoria Sterowania** – *metoda Zieglera-Nicholsa*- http://home.agh.edu.pl/~o_iwona/ins_ts4.pdf
- [28] **Understanding Euler angles** – *reprezentacja orientacji przestrzennej przy użyciu kątów Eulera – opracowanie własne na podstawie :* <http://www.chrobotics.com/library/understanding-euler-angles>