

Politechnika Wrocławska

Wydział Elektroniki, Fotoniki i Mikrosystemów

KIERUNEK: Automatyka i Robotyka (AIR)

PRACA DYPLOMOWA INŻYNIERSKA

TYTUŁ PRACY:
Wektorowanie ciągu w zadaniu sterowania
bezzałogowym obiektem latającym

AUTOR:
Eryk Możdżeń

PROMOTOR:
Dr inż. Robert Muszyński,
Katedra Cybernetyki i Robotyki

Spis treści

1	Wstęp	3
2	Model obiektu	4
2.1	Układy współrzędnych	4
2.2	Model układów wykonawczych	5
2.2.1	Układ napędowy	6
2.2.2	Łopatki	7
2.3	Model jednowirnikowego drona	8
3	Algorytm sterowania	10
3.1	Regulator LQR	10
3.1.1	Zastosowanie	10
3.1.2	Kryterium optymalności	11
3.1.3	Człon całkujący	11
3.2	Implementacja regulatora	12
3.2.1	Punkt równowagi	12
3.2.2	Linearyzacja modelu	13
3.2.3	Strategia sterowania	14
3.3	Symulacje	15
4	Platforma testowa	16
4.1	Konstrukcja mechaniczna	16
4.1.1	Przegląd rozwiązań	16
4.1.2	Analiza modelu matematycznego	16
4.1.3	Projekt CAD	18
4.2	Układy elektroniki	23
4.2.1	Wymagania funkcjonalne	23
4.2.2	Elementy składowe	23
4.2.3	Schemat oraz projekt sterownika lotu	25
4.3	Oprogramowanie	28
4.3.1	Program sterownika lotu	28
4.3.2	Program stacji pilota	33
4.3.3	Programy do kalibracji	34
4.4	Identyfikacja parametrów modelu	37
4.4.1	Identyfikacja współczynnika ciągu i prędkości obrotowej	37
4.4.2	Identyfikacja współczynnika momentu obrotowego	38
5	Eksperymenty	40

5.1	Zdolność do poruszania się	40
5.2	Lot swobodny	40
6	Podsumowanie	42
	Literatura	43
A	Rysunki techniczne	47
B	Schematy sterownika lotu	56
C	Pomiary identyfikacji	62
D	Galeria zdjęć	66

Do składu pracy wykorzystano system przygotowania dokumentów \LaTeX , opracowany przez L. Lamport [Lam94], będący nakładką systemu \TeX [Knu86a, Knu86b]. Matematyczne czcionki o nazwie AMS Euler, których używam w tej pracy, zostały przygotowane przez H. Zapf [KZ86], przy współpracy z D. Knuthem i jego studentami, na zlecenie Amerykańskiego Towarzystwa Matematycznego. Czcionka składu tekstu, zwana URW Palladio jest klonem zapfowskiej rodziny czcionek o nazwie Palatino [LPn05] i zdaniem autora świetnie współgra z kształtem czcionki AMS Euler. Autorami szablonu pracy są R. Muszyński i R. Orozco [MO22], którzy wykorzystali klasę mgr.cls stworzoną przez A. Ratajczaka [Rat08].

- [Knu86a] D. E. Knuth, The \TeX book, volume A of Computers and Typesetting. Addison-Wesley, Reading, 1986.
- [Knu86b] D. E. Knuth, \TeX : The Program, volume B of Computers and Typesetting. Addison-Wesley, Reading, 1986.
- [KZ86] D. E. Knuth i H. Zapf, AMS Euler — A new typeface for mathematics. Scholarly Publishing, 20:131–157, 1986.
- [Lam94] L. Lamport, \LaTeX : A Document Preparation System. Addison–Wesley, Reading, 1994.
- [LPn05] Linotype Palatino nova: A classical typeface redesigned by Hermann Zapf, Linotype Library GmbH, 2005.
- [MO22] R. Muszyński i R. Orozco, Przykład i wytyczne formatowania pracy dyplomowej. https://kcir.pwr.edu.pl/~much/Pracki/praca_dyplomowa_wzor.pdf, 2022.
- [Rat08] A. Ratajczak, Klasa dokumentu - mgr.cls. <http://diablo.kcir.pwr.edu.pl/~ar/LaTeX/mgr.php>, 2008.

Rozdział 1

Wstęp

Od najdawniejszych czasów ludzkość pragnie zrzucić ograniczenia nadane przez naturę w sposobie poruszania się. Dzięki dokonaniom braci Montgolfier ludzie oderwali się od ziemi za sterami balonu na gorące powietrze (1783) [Wikd]. Jednak dopiero wojny XX wieku oraz pojawiające się możliwości technologiczne skłoniły konstruktorów do przeprowadzania lotów bez narażania życia pilotów. Początkowo bezzałogowe statki powietrzne UAV (ang. *Unmanned Aerial Vehicle*), popularnie nazywane dronami, rozwijane były przez przemysł wojskowy jako nośniki ładunków wybuchowych, jednak wraz z miniaturyzacją układów sensorycznych zaczęto wykorzystywać je także w zadaniach inspekcji lub patrolu [Arm12, Air21]. Współcześnie drony rozpowszechniły się na rynku cywilnym, pozwalając na rozwój aerofotografii, transportu oraz rozrywki.

Szczególnym typem statków powietrznych są wirnikowce, w których siła nośna generowana jest przez niezależnie napędzane śmigła. Mimo posiadania wielu zalet, takich jak zdolność do pionowego startu i lądowania VTOL (ang. *Vertical Take-Off and Landing*) oraz prosta budowa mechaniczna, wymagają one dużej liczby jednostek napędowych, co zwiększa koszty produkcji drona. Rozwiązaniem tego problemu, jest wywodzące się z technologii rakietowych sterowanie wektorem ciągu TVC (ang. *Thrust Vector Control*), które polega na zmianie wartości ciągu oraz odchyleniu kierunku jego działania w płaszczyźnie pionowej, poziomej, lub w obu płaszczyznach jednocześnie [R607]. W tym rozwiązaniu już pojedynczy wirnik wraz z łopatkami odchylającymi strumień powietrza pozwala na zachowanie stabilności podczas lotu. Zastosowanie dużego śmigła pozwala na osiągnięcie większej sprawności układu w porównaniu do wielu niewielkich wirników, co może prowadzić do dłuższego czasu pracy w terenie.

Celem pracy jest skonstruowanie jednowirnikowego drona oraz wykorzystanie jego modelu w zadaniu sterowania przez wektorowanie ciągu. W dokumencie dron rozpatrywany będzie jako robot mobilny. Układ pracy jest następujący. W drugim rozdziale opisany został proces modelowania obiektu. Rozdział trzeci przybliży proponowaną strategię sterowania robotem. Czwarty rozdział zawiera szczegółowy opis procesu identyfikacji parametrów modelu oraz budowy platformy testowej z podziałem na konstrukcję mechaniczną, układy elektroniki i oprogramowanie. Rozdział piąty traktuje o eksperymentach przeprowadzonych celem sprawdzenia działania układu sterowania w warunkach lotu. Rozdział szósty podsumowuje pracę i przedstawia możliwe kierunki jej kontynuacji.

Rozdział 2

Model obiektu

Wiedzę o obiekcie zawarto w matematycznym opisie zwanym modelem, który opracowywany jest według zasady Brzytwy Ockhama [PWN]. Poziom jego skomplikowania powinien być wystarczający do osiągnięcia celu, ale nie wyższy. Zdefiniowanie wystarczającego modelu robota pozwoli na skuteczną realizację sterowania nim. Model jest podstawą „wirtualnego bliźniaka”, dzięki któremu możliwa jest powtarzalna predykcja zachowania obiektu bez ryzyka narażania rzeczywistego układu na szwank. W tym rozdziale przedstawiony został proces modelowania jednowirnikowego drona oraz użyte do tego celu narzędzia.

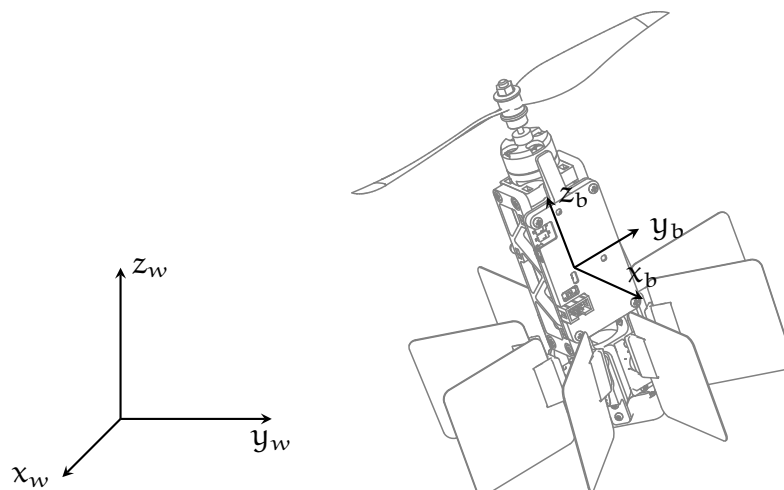
2.1 Układy współrzędnych

Bryła sztywna w przestrzeni posiada 6 stopni swobody, więc do jej opisu konieczne jest 6 liczb – 3 składowe opisujące położenie oraz 3 elementy określające orientację. Układy współrzędnych zostały wybrane analogicznie jak w pracy [Jac21], co przedstawia rysunek 2.1. Zdefiniowano prawoskrętny kartezjański układ współrzędnych $X_w Y_w Z_w$ zwany dalej układem świata. Oś Z_w jest równoległa do kierunku działania siły grawitacyjnej i ma w stosunku do niej przeciwny zwrot. Płaszczyzna XY umownie oznacza poziom gruntu. Lokalny układ współrzędnych drona $X_b Y_b Z_b$ ma początek w środku ciężkości maszyny a oś Z_b pokrywa się z osią obrotu wirnika. Orientacja lokalnego układu współrzędnych została opisana parametryzacją Eulera ZYX zwanej kiwaniem–kołysaniem–myszkowaniem KKM (ang. *Roll Pitch Yaw*) [TM18]. Wyraża ona orientację za pomocą obrotu globalnego układu współrzędnych kolejno w osiach Z , Y , X odpowiednio o kąty ψ , θ i ϕ . Macierz rotacji \mathbf{R}_w^b można wyrazić jako

$$\mathbf{R}_w^b = \mathbf{R}_z(\psi)\mathbf{R}_y(\theta)\mathbf{R}_x(\phi) = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix},$$

gdzie c_x oraz s_x oznaczają odpowiednio $\cos x$ i $\sin x$. Pozwala to przeliczać położenie wektorów wyrażonych w lokalnym układzie współrzędnym \mathbf{v}_b na wektor w układzie globalnym \mathbf{v}_w za pomocą zależności

$$\mathbf{v}_w = \mathbf{R}_w^b \mathbf{v}_b.$$



Rysunek 2.1 Przyjęte układy współrzędnych

Prędkości kątowe ω oraz postępowe \mathbf{v} korpusu robota zostały wyrażone w jego lokalnym układzie współrzędnych. Ich relację pomiędzy prędkością postępową w globalnym układzie współrzędnych oraz pochodną kątów KKM można określić jako

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \mathbf{R}_w^b \mathbf{v}, \quad \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \mathbf{W}_{KKM}^b \omega,$$

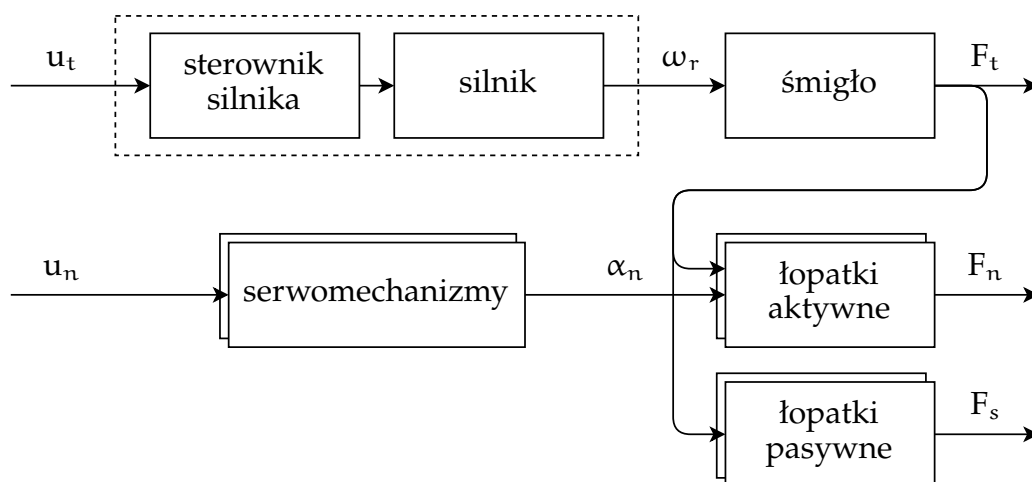
gdzie

$$\mathbf{W}_{KKM}^b = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & \frac{s_\phi}{c_\theta} & \frac{c_\phi}{c_\theta} \end{bmatrix},$$

x , y , z to składowe pozycji drona w globalnym układzie współrzędnych a t_θ oznacza $\tan \theta$.

2.2 Model układów wykonawczych

Wektorowanie ciągu realizowane zostało za pomocą wirnika napędowego oraz zestawu łopatek odchylających strumień powietrza. Rysunek 2.2 przedstawia schemat blokowy układów wykonawczych jednowirnikowca. Sterowania u_t i u_n zadają wartość prędkości obrotowej rotora silnika ω_r oraz wychylenie orczyka serwomechanizmu α_n . Wartość siły ciągu oznaczona została symbolem F_t i wpływa na wartości sił F_n oraz F_s generowanych przez łopatki.



Rysunek 2.2 Schemat blokowy układów wykonawczych

2.2.1 Układ napędowy

Źródłem siły ciągu jest pojedyncze śmigło napędzane przez silnik elektryczny. W pracach [Jac21, DGSAB⁺22, CFAN16] śmigła reprezentowano modelem

$$F_t = K_\omega \omega_r^2, \quad (2.1)$$

gdzie F_t jest wartością generowanej siły ciągu, ω_r jest prędkością obrotową wirnika a K_ω jest stałą charakteryzującą układ. Wyniki identyfikacji przedstawione w dodatku C na wykresie C.2 potwierdzają stosowność takiego modelu również w omawianym obiekcie.

Badania identyfikacyjne przeprowadzone na jednowirnikowcu, których wyniki pokazane są na wykresie C.1 dowodzą, że dla rozpatrywanego obiektu występuje zależność

$$F_t = K_t u_t,$$

gdzie K_t jest stałą charakteryzującą układ. Według literatury [Jac21] para sterownik-silnik stosowana w małych dronach charakteryzuje się niewielką stałą czasową i modeluje się ją jako człon proporcjonalny, gdzie prędkość obrotowa wirnika ω_r jest proporcjonalna do sterowania u_t . Jednak po przyrównaniu powyższego równania do (2.1) uzyskujemy model pary sterownik-silnik w postaci

$$\omega_r = \sqrt{\frac{K_t}{K_\omega}} u_t.$$

W konstrukcjach wielowirnikowych zjawiska dotyczące zachowania momentu pędu oraz oporów powietrza znoszą się, ponieważ stosuje się w nich pary wirników obracające się w przeciwnych kierunkach. W przypadku omawianego drona konieczne jest zawarcie w modelu wymienionych zjawisk.

Wirnik podczas obrotu wytwarza moment obrotowy M_r spowodowany oporami powietrza. Moment ten oddziałuje na korpus w osi obrotu wirnika i posiada zwrot

przeciwny do jego prędkości kątowej. Jego wartość została zamodelowana jako proporcjonalna do wartości siły ciągu [CFAN16]. Jako że siła ciągu również jest liniowo zależna od sterowania u_t zastosowano model

$$M_r = K_\tau u_t,$$

gdzie K_τ jest stałą charakteryzującą układ. Wyniki identyfikacji zgromadzone na wykresie C.3 potwierdzają stosowność takiego modelu.

Zmiana osi obrotu szybko obracającej się masy powoduje występowanie efektu żyroskopowego. Zjawisko to sprzęga prędkości i przyspieszenia kątowe ciała w osiach równoległych do osi obrotu wirnika. W literaturze [DGSAB+22] wartość generowanego momentu M_{gyr} zamodelowano jako

$$\mathbf{M}_{gyr} = \begin{bmatrix} J_r \omega_r \omega_y \\ -J_r \omega_r \omega_x \\ 0 \end{bmatrix},$$

gdzie J_r to moment bezwładności całego wirnika (śmigło oraz rotor silnika) w osi obrotu, czyli osi Z.

2.2.2 Łopatki

Wektorowanie ciągu możliwe jest dzięki łopatkom odchylającym jego wektor. Modeluje się to poprzez dodanie do wektora siły ciągu składowych prostopadłych F_n , generowanych przez obrócone łopatki. Założono, że serwomechanizm jest obiektem statycznym, więc stosując odpowiednią funkcję odwrotną, można sterować bezpośrednio kątem wychylenia orczyka od pionu α_n . Dla niewielkich kątów α_n mieszczących się w zakresie $\pm 15^\circ$ sprawdza się model łopatki zdefiniowany jako [Jac21, CFAN16]

$$F_n = C_l F_t \alpha_n,$$

gdzie C_l jest współczynnikiem siły nośnej łopatki. Opór powietrza stawiany przez łopatki został pominięty z powodu niewielkiego wpływu na układ w rozpatrywanym zakresie pracy [Jac21].

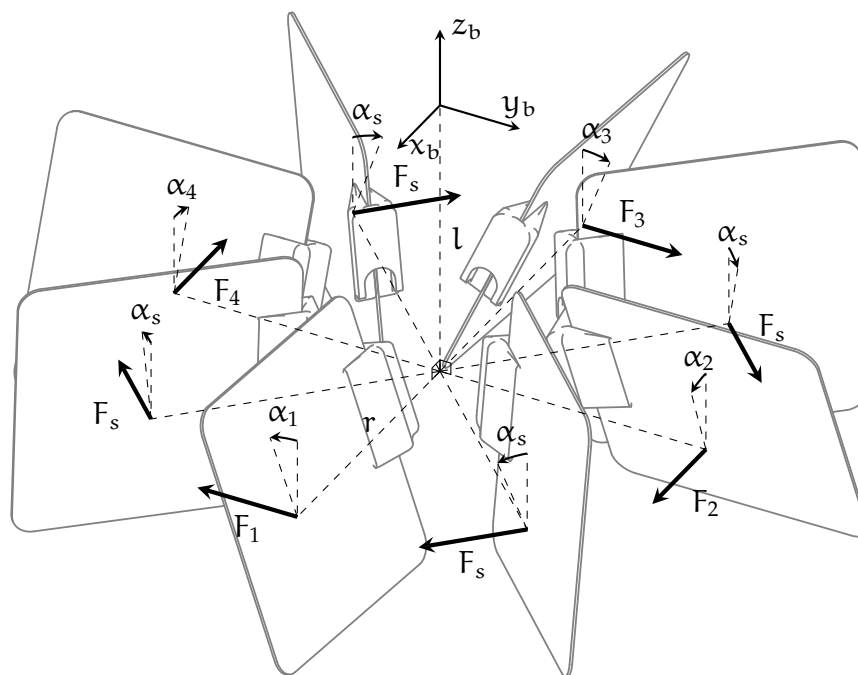
Współczynnik siły nośnej C_l jest jednym z podstawowych parametrów opisujących charakterystykę skrzydła o danym przekroju. Zawiera informację o tym, jak duża jest generowana siła nośna przy danym kącie natarcia skrzydła. W dalszych rozważaniach przyjęto wartość [Jac21]

$$C_l = \frac{P_l}{2P_r} C_{L\alpha},$$

gdzie P_l to powierzchnia skrzydła (tutaj łopatki), P_r to pole przekroju poprzecznego strumienia powietrza otaczającego skrzydło a $C_{L\alpha}$ jest współczynnikiem charakteryzującym geometrię skrzydła. Dla skrzydeł w postaci płaskiej płyty o nieskończonej długości krawędzi przyjmuje się wartość $C_{L\alpha} = 2\pi$ [Bre06]. Pole przekroju poprzecznego strumienia powietrza obliczono ze wzoru

$$P_r = \pi R_r^2,$$

gdzie R_r jest promieniem śmigła.



Rysunek 2.3 Wektory sił generowanych przez łopatki

2.3 Model jednowirnikowego drona

Model dynamiki jednowirnikowego drona bazuje na wyprowadzeniach opracowanych przez autora w ramach kursu Roboty Mobilne [Moż23f, Moż23e, Moż23d] oraz pozycji [Jac21]. Dynamika ruchu obrotowego wyprowadzona jest z wykorzystaniem formalizmu Newtona–Eulera, mianowicie

$$\begin{bmatrix} \mathbf{F} \\ \boldsymbol{\tau} \end{bmatrix} = \begin{bmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{J} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} \end{bmatrix},$$

gdzie macierz bezwładności całego drona

$$\mathbf{J} \approx \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix}$$

zawiera najbardziej znaczące elementy na głównej przekątnej a pozostałe można przybliżyć do zera. Symbole \mathbf{F} i $\boldsymbol{\tau}$ to wektory sił i momentów działających na obiekt wyrażone w lokalnym układzie współrzędnych a m to masa całkowita maszyny.

Rysunek 2.3 przedstawia punkty zaczepienia i kierunki wektorów sił generowanych przez łopatki. Parametry l i r określają odległości geometrycznego środka łopatki od środka ciężkości obiektu odpowiednio w osi pionowej i poziomej, $F_i, i = 1, \dots, 4$ to wartości sił generowanych przez łopatki aktywne a F_s to wartość siły generowana przez łopatki pasywne. Poczyniono następujące założenia dotyczące łopatek:

- są zamontowane osiowosymetrycznie względem osi obrotu wirnika,

- charakteryzują się równym co do wartości polu powierzchni,
- znajdują się poniżej środka ciężkości robota.

Przy takich założeniach model ruchu obrotowego obiektu można wyrazić jako

$$\begin{cases} \dot{\phi} = \omega_x + \omega_y s_\phi t_\theta + \omega_z c_\phi t_\theta \\ \dot{\theta} = \omega_y c_\phi - \omega_z s_\phi \\ \dot{\psi} = \omega_y \frac{s_\phi}{c_\theta} + \omega_z \frac{c_\phi}{c_\theta} \\ \dot{\omega}_x = \frac{1}{J_{xx}}(J_{yy} - J_{zz})\omega_y \omega_z + \frac{1}{J_{xx}}l(F_3 - F_1) + \frac{J_r}{J_{xx}}\omega_r \omega_y \\ \dot{\omega}_y = \frac{1}{J_{yy}}(J_{zz} - J_{xx})\omega_x \omega_z + \frac{1}{J_{yy}}l(F_4 - F_2) - \frac{J_r}{J_{yy}}\omega_r \omega_x \\ \dot{\omega}_z = \frac{1}{J_{zz}}(J_{xx} - J_{yy})\omega_x \omega_y - \frac{1}{J_{zz}}r(F_1 + F_2 + F_3 + F_4 + 4F_s) + \frac{1}{J_{zz}}M_r \end{cases} \quad (2.2)$$

Model ruchu postępowego został wyrażony jako

$$\begin{cases} \dot{x} = v_x c_\theta c_\psi + v_y (s_\phi s_\theta c_\psi - c_\phi s_\psi) + v_z (c_\phi s_\theta c_\psi + s_\phi s_\psi) \\ \dot{y} = v_x c_\theta s_\psi + v_y (s_\phi s_\theta s_\psi + c_\phi c_\psi) + v_z (c_\phi s_\theta s_\psi - s_\phi c_\psi) \\ \dot{z} = -v_x s_\theta + v_y s_\phi c_\theta + v_z c_\phi c_\theta \\ \dot{v}_x = \frac{1}{m}(F_2 - F_4) + g s_\theta \\ \dot{v}_y = \frac{1}{m}(F_3 - F_1) - g s_\phi c_\theta \\ \dot{v}_z = \frac{1}{m}F_t - g c_\phi c_\theta \end{cases} \quad (2.3)$$

gdzie g to wartość przyspieszenia ziemskiego.

Rozdział 3

Algorytm sterowania

Jednowirnikowiec z wektorowaniem ciągu jest układem z natury niestabilnym. Z tego powodu konieczne jest stworzenie algorytmu sterowania, które umożliwi kontrolę nad nim. W tym rozdziale przedstawiono zastosowany regulator i jego wariacje, opisano sposób jego implementacji w zadaniu sterowania dronem oraz podsumowano wyniki testów symulacyjnych układu sterowania.

3.1 Regulator LQR

Regulator liniowo-kwadratowy (ang. *Linear-Quadratic Regulator*) [[WIKI](#)] jest liniowym regulatorem realizującym sterowanie ze sprzężeniem zwrotnym dynamicznymi układami liniowymi. Regulator LQR można interpretować jako odpowiednik MIMO (ang. *Multi Input Multi Output*) skalarne regulatora typu P, w którym parametry są dobrane optymalnie.

3.1.1 Zastosowanie

Regulator LQR stosowany jest do liniowych układów dynamiki, które można opisać równaniem

$$\dot{x} = Ax + Bu,$$

gdzie x jest wektorem stanu układu a u jest wektorem sterowań. Macierze A i B to macierze charakteryzujące obiekt. Tak opisany układ posiada jeden punkt równowagi $x = 0$. W przypadku gdy układ jest niestabilny (lub na granicy stabilności) celem stabilizacji układu proponuje się sprzężenie zwrotne w postaci

$$u = -Kx,$$

gdzie K jest stałą macierzą współczynników. Aby ustabilizować obiekt w punkcie innym niż zero, należy przesunąć punkt stabilności do pożądanej wartości zadanej x_z

$$u = -K(x - x_z) = -Ke,$$

na co pozwala zasada superpozycji spełniana przez układy liniowe. Występujący w powyższej formule wektor e definiuje się jako uchyb regulacji.

3.1.2 Kryterium optymalności

Macierz współczynników \mathbf{K} jest optymalnie dobrana tak, aby minimalizować kwadratową funkcję kosztu, która w ogólnej postaci zdefiniowana jest jako [Wikf]

$$J = \int_0^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u} + 2\mathbf{x}^T \mathbf{N} \mathbf{u}) dt,$$

gdzie macierze \mathbf{Q} , \mathbf{R} oraz \mathbf{N} zawierają wagi kosztu odpowiednio dla kwadratu elementów wektora stanu, kwadratów sterowań oraz ich iloczynu mieszanego. Zwyczajowo macierze \mathbf{Q} oraz \mathbf{R} zawierają niezerowe elementy wyłącznie na swojej głównej przekątnej. Dobór wartości elementów macierzy \mathbf{Q} , \mathbf{R} i \mathbf{N} dokonywany jest doświadczalnie. Macierz \mathbf{K} wyrażona jest zależnością

$$\mathbf{K} = \mathbf{R}^{-1} (\mathbf{B}^T \mathbf{P} + \mathbf{N}^T),$$

gdzie \mathbf{P} jest rozwiązaniem algebraicznego równania Riccatiego podanego jako [Wika]

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} - (\mathbf{P} \mathbf{B} + \mathbf{N}) \mathbf{R}^{-1} (\mathbf{B}^T \mathbf{P} + \mathbf{N}^T) + \mathbf{Q} = \mathbf{0}.$$

3.1.3 Człon całkujący

Regulator proporcjonalny często charakteryzuje się występowaniem niezerowego uchybu w stanie ustalonym, co w przypadku niektórych zastosowań jest niedopuszczalne. Rozszerzenie regulatora LQR o dodatkowe elementy całkujące pozwala interpretować go jako odpowiednik MIMO regulatora typu PI [MC14].

Aby wyeliminować uchyb w stanie ustalonym, możliwe jest rozszerzenie wektora uchybu o elementy

$$\mathbf{e}_{PI} = \begin{bmatrix} \mathbf{e} \\ \mathbf{e}_I \end{bmatrix},$$

gdzie

$$\mathbf{e}_I = \int_0^t \mathbf{G} \mathbf{e} dt,$$

\mathbf{G} to macierz z elementami o wartościach 0 lub 1 decydująca, które składowe wektora uchybu należy poddawać całkowaniu. W ten sposób aby, obliczyć rozszerzoną macierz współczynników \mathbf{K}_{PI} , należy sztucznie powiększyć model obiektu i funkcję kosztu o dodatkowy integrator

$$\mathbf{A}_{PI} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{G} & \mathbf{0} \end{bmatrix}, \quad \mathbf{B}_{PI} = \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix}, \quad \mathbf{Q}_{PI} = \begin{bmatrix} \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_I \end{bmatrix}, \quad \mathbf{N}_{PI} = \begin{bmatrix} \mathbf{N} \\ \mathbf{N}_I \end{bmatrix},$$

gdzie \mathbf{Q}_I i \mathbf{N}_I to macierze zawierające nowe wagi kryterium optymalizacyjnego, dzięki którym możliwe jest dostosowanie wpływu wartości całek z wybranych uchybów na sterowanie.

3.2 Implementacja regulatora

Implementacja regulatora w jednowirnikowym dronie bazuje na implementacji opracowanej przez autora w ramach kursu Roboty Mobilne [Moż23f, Moż23e, Moż23d]. Model drona przedstawiony w rozdziale 2 jest modelem nieliniowym. Zaproponowano więc sterowanie obiektem w bliskim otoczeniu jednego z punktów równowagi modelu i wykorzystanie jego liniowego przybliżenia w tym punkcie do implementacji regulatora LQR z elementem całkującym.

3.2.1 Punkt równowagi

Punktem, w którym dokonana została linearyzacja modelu obiektu, jest punkt równowagi. Z definicji jest to punkt, w którym przy stałym wymuszeniu nie występuje ruch ciała, co równoznaczne jest z przyrównaniem prawej strony równań dynamiki drona (2.2), (2.3) do zera

$$\left\{ \begin{array}{l} 0 = \omega_x + \omega_y s_\phi t_\theta + \omega_z c_\phi t_\theta \\ 0 = \omega_y c_\phi - \omega_z s_\phi \\ 0 = \omega_y \frac{s_\phi}{c_\theta} + \omega_z \frac{c_\phi}{c_\theta} \\ 0 = \frac{1}{J_{xx}} (J_{yy} - J_{zz}) \omega_y \omega_z + \frac{1}{J_{xx}} l C_l K_t u_t (\alpha_3 - \alpha_1) + \frac{J_r}{J_{xx}} \sqrt{\frac{K_t}{K_\omega}} u_t \omega_y \\ 0 = \frac{1}{J_{yy}} (J_{zz} - J_{xx}) \omega_x \omega_z + \frac{1}{J_{yy}} l C_l K_t u_t (\alpha_4 - \alpha_2) - \frac{J_r}{J_{yy}} \sqrt{\frac{K_t}{K_\omega}} u_t \omega_x \\ 0 = \frac{1}{J_{zz}} (J_{xx} - J_{yy}) \omega_x \omega_y - \frac{1}{J_{zz}} r C_l K_t u_t (\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + 4\alpha_s) + \frac{1}{J_{zz}} K_\tau u_t \\ 0 = v_x c_\theta c_\psi + v_y (s_\phi s_\theta c_\psi - c_\phi s_\psi) + v_z (c_\phi s_\theta c_\psi + s_\phi s_\psi) \\ 0 = v_x c_\theta s_\psi + v_y (s_\phi s_\theta s_\psi + c_\phi c_\psi) + v_z (c_\phi s_\theta s_\psi - s_\phi c_\psi) \\ 0 = -v_x s_\theta + v_y s_\phi c_\theta + v_z c_\phi c_\theta \\ 0 = \frac{1}{m} K_t u_t C_l (\alpha_2 - \alpha_4) + g s_\theta \\ 0 = \frac{1}{m} K_t u_t C_l (\alpha_3 - \alpha_1) - g s_\phi c_\theta \\ 0 = \frac{1}{m} K_t u_t - g c_\phi c_\theta. \end{array} \right.$$

Przykładowym rozwiązaniem powyższego układu równań jest zbiór punktów \mathbf{x}_0 przy wymuszeniu \mathbf{u}_0

$$\mathbf{x}_0 = \begin{bmatrix} \phi \\ \theta \\ \psi \\ \omega_x \\ \omega_y \\ \omega_z \\ \chi \\ y \\ z \\ v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ * \\ 0 \\ 0 \\ 0 \\ * \\ * \\ * \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{u}_0 = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ u_t \end{bmatrix} = \begin{bmatrix} \alpha_{n0} \\ \alpha_{n0} \\ \alpha_{n0} \\ \alpha_{n0} \\ u_{t0} \end{bmatrix},$$

gdzie

$$\alpha_{n0} = -\frac{K_\tau}{4C_l K_t r} - \alpha_s, \quad u_{t0} = \frac{mg}{K_t}, \quad (3.1)$$

* oznacza dowolną liczbę rzeczywistą. W dalszych dywagacjach wektor u_0 będzie stanowić punkt pracy urządzeń wykonawczych, przyjęto także

$$x_0 = \mathbf{0}.$$

3.2.2 Linearyzacja modelu

Aby umożliwić implementację regulatora, konieczna jest znajomość liniowego układu dynamiki, będącego liniowym przybliżeniem modelu drona wyrażonego w postaci

$$\dot{x} = f(x, u).$$

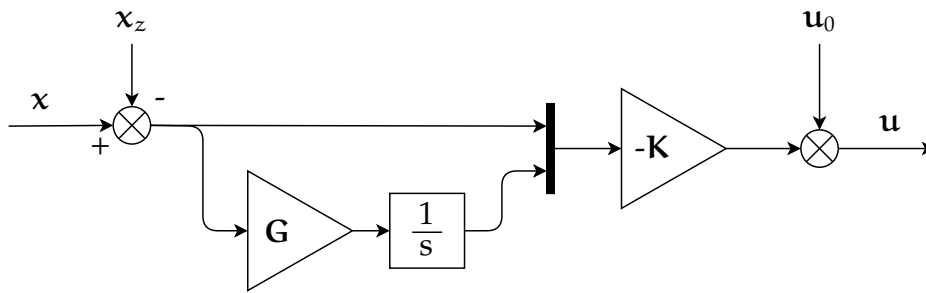
W tym celu obliczono pochodne cząstkowe odpowiednio po stanie obiektu x oraz sterowaniu u w punkcie równowagi

$$\mathbf{A} = \left. \frac{\partial f(x, u)}{\partial x} \right|_{x_0, u_0}, \quad \mathbf{B} = \left. \frac{\partial f(x, u)}{\partial u} \right|_{x_0, u_0},$$

gdzie \mathbf{A} oraz \mathbf{B} to macierze opisujące dynamiczny obiekt liniowy. W wyniku obliczeń otrzymano

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{J_r}{J_{xx}} \omega_{t0} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{J_r}{J_{yy}} \omega_{t0} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & g & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -g & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{J_{xx}} F_{t0} C_l l & 0 & \frac{1}{J_{xx}} F_{t0} C_l l & 0 & 0 \\ 0 & -\frac{1}{J_{yy}} F_{t0} C_l l & 0 & \frac{1}{J_{yy}} F_{t0} C_l l & 0 \\ -\frac{1}{J_{zz}} F_{t0} C_l r & -\frac{1}{J_{zz}} F_{t0} C_l r & -\frac{1}{J_{zz}} F_{t0} C_l r & -\frac{1}{J_{zz}} F_{t0} C_l r & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{m} F_{t0} C_l & 0 & -\frac{1}{m} F_{t0} C_l & 0 \\ -\frac{1}{m} F_{t0} C_l & 0 & \frac{1}{m} F_{t0} C_l & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{m} K_t \end{bmatrix},$$



Rysunek 3.1 Diagram układu sterowania

gdzie

$$\omega_{t0} = \sqrt{\frac{K_t}{K_\omega}} u_{t0}, \quad F_{t0} = K_t u_{t0}.$$

3.2.3 Strategia sterowania

Rysunek 3.1 przedstawia układ sterowania złożony z regulatora LQR, którego wyjście jest przesunięte do punktu pracy urządzeń wykonawczych. Zadaniem sterowania jest utrzymanie:

- zadanego wychylenia w osi X,
- zadanego wychylenia w osi Y,
- zadanej prędkości kątowej w osi Z,
- zadanej wysokości nad ziemią.

Z wektora stanu x oraz macierzy A i B usunięto zbędne składowe pozostawiając

$$x = \begin{bmatrix} \phi \\ \theta \\ \omega_x \\ \omega_y \\ \omega_z \\ z \\ v_z \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{J_r}{J_{xx}} \omega_{t0} & 0 & 0 & 0 \\ 0 & 0 & -\frac{J_r}{J_{yy}} \omega_{t0} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{J_{xx}} F_{t0} C_{1l} & 0 & \frac{1}{J_{xx}} F_{t0} C_{1l} & 0 & 0 \\ 0 & -\frac{1}{J_{yy}} F_{t0} C_{1l} & 0 & \frac{1}{J_{yy}} F_{t0} C_{1l} & 0 \\ -\frac{1}{J_{zz}} F_{t0} C_{1r} & -\frac{1}{J_{zz}} F_{t0} C_{1r} & -\frac{1}{J_{zz}} F_{t0} C_{1r} & -\frac{1}{J_{zz}} F_{t0} C_{1r} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{m} K_t \end{bmatrix}.$$

Niedoskonałości konstrukcji mechanicznej, takie jak położenie środka ciężkości poza pionową osią robota oraz zależność siły ciągu od napięcia zasilania baterii, unaocniły

Rozdział 4

Platforma testowa

4.1 Konstrukcja mechaniczna

Konstrukcja mechaniczna drona powinna charakteryzować się ograniczoną masą oraz wysoką wytrzymałością. Poziomą integracją układów elektroniki wymusił zaprojektowanie dedykowanego korpusu robota. W tym podrozdziale przedłożone zostały możliwe rozwiązania konstrukcyjne, przeanalizowano model matematyczny obiektu oraz opisano budowę mechaniczną jednowirnikowca.

4.1.1 Przegląd rozwiązań

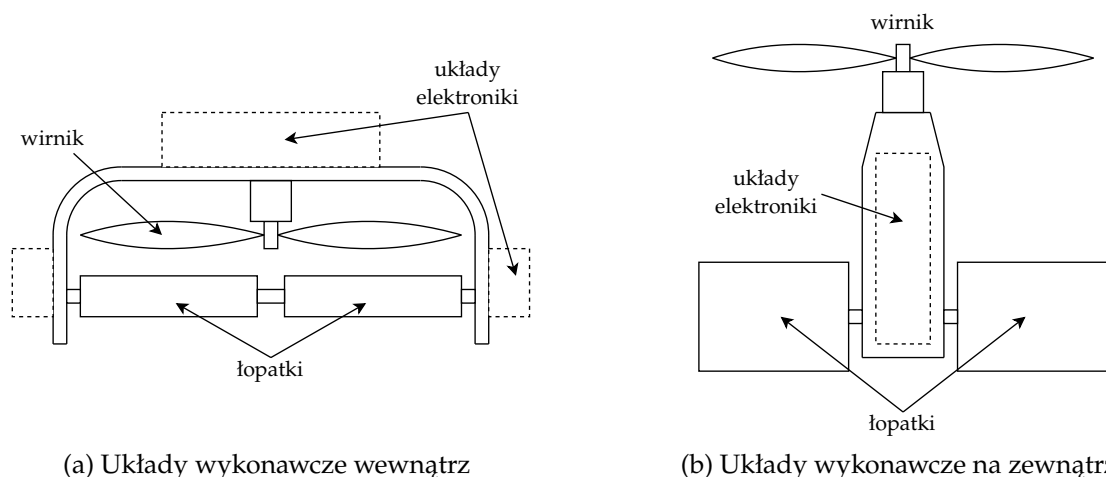
Tematyka budowy drona TVC poruszana była w opracowaniach naukowych oraz militarnych wielokrotnie. Wśród opisanych w literaturze pojazdów można wyróżnić dwa odrębne podejścia konstrukcyjne przedstawione na rysunku 4.1.

Wariant występujący w literaturze [[Jac21](#)] oraz realizacjach militarnych [[Arm12](#), [Air21](#)] został przedstawiony na rysunku 4.1a. Charakteryzuje się on całkowitą zabudową elementów ruchomych, co pozwala na sprawniejsze wykorzystanie aktuatorów robota. Układy elektroniki znajdują się na zewnątrz względem urządzeń wykonawczych.

Drugi wariant, opisywany w literaturze [[DGSAB+22](#), [CFAN16](#)] został przedstawiony na rysunku 4.1b. Zakłada on umiejscowienie wszystkich układów elektroniki w obrysie kadłuba, pozostawiając elementy ruchome na zewnątrz. Takie podejście wymaga uzyskania niewielkiej powierzchni przekroju poprzecznego drona, co implikuje wysoki stopień integracji układów elektroniki (niewiele dostępnej przestrzeni). W tym rozwiązaniu elementy ruchome są bardziej narażone na awarie związane z upadkami na ziemię a lotki z powodu dużej odległości od wirnika muszą posiadać większą powierzchnię.

4.1.2 Analiza modelu matematycznego

Model przedstawiony w rozdziale 2, oprócz stanowienia bazy do zaprojektowania układu sterowania jest podstawą do zrozumienia zjawisk rządzących robotem. Analiza modelu obiektu pozwala osiągnąć wiedzę na temat kluczowych parametrów w celu



Rysunek 4.1 Najczęściej stosowane warianty konstrukcyjne jednowirnikowców

zaprojektowania konstrukcji tak, aby możliwie ułatwić proces wdrażania układu sterowania.

Analiza modelu ruchu postępowego

Model przyśpieszeń liniowych (2.3) zależny jest od masy robota oraz siły ciągu. Zależność siły ciągu od sterowania została opisana wzorem

$$F_t = K_t u_t$$

gdzie $u_t \in [0; 1]$ jest sterowaniem zadawanym sterownikowi silnika a K_t współczynnikiem siły ciągu. W każdym przypadku warunkiem koniecznym do utrzymania możliwości lotu jest

$$K_t > mg.$$

Im wartość współczynnika siły ciągu K_t jest większa, tym większy jest zakres siły ciągu możliwej do wygenerowania. Układ powinien być możliwie lekki, aby zwiększyć zakres możliwych do uzyskania przyśpieszeń liniowych.

Analiza modelu ruchu obrotowego

Z modelu przyśpieszeń kątowych (2.2) można wywnioskować, że aby zminimalizować zjawisko efektu żyroskopowego stosunek momentu bezwładności wirnika J_r do momentów bezwładności całego drona (odpowiednio J_{xx} i J_{yy}) powinien być minimalizowany. Zwiększenie wartości współczynnika prędkości obrotowej K_ω pozwoli na osiąganie mniejszej prędkości obrotowej wirnika, co również zmniejszy wpływ oddziaływania efektu żyroskopowego. Ograniczenie zmian prędkości wirnika pozwoli na zminimalizowanie momentu generowanego w osi pionowej.

Zwiększenie odległości łopatek od środka ciężkości (l i r) oraz zwiększenie wartości współczynnika oddziaływania łopatek C_l pozwoli na zwiększenie generowanych przez łopatki momentów obrotowych. Bezwładność układu powinna być możliwie niska, aby zwiększyć zakres możliwych do uzyskania przyśpieszeń kątowych.

Analiza punktu pracy modelu

Występowania niezerowego momentu obrotowego oddziałującego na korpus wygenerowanego przez wirnik, powoduje konieczność wygenerowania momentu o równej wartości, lecz przeciwnym zwrocie. Kąt wychylenia łopatek aktywnych w punkcie pracy opisany równaniem (3.1) pomniejszony jest o wartość α_s , która stanowi kąt nachylenia łopatek pasywnych (pomocniczych).

Kąt α_s powinien zostać dobrany tak, aby punkt pracy łopatek aktywnych był możliwie blisko środka zakresu ich pracy (zera). Należy także zadbać, aby α_s mieścił się w zakresie obowiązywania modelu. W tym celu zastosowano przybliżenie

$$\alpha_s \approx -\frac{K_\tau}{4C_l K_t r}.$$

4.1.3 Projekt CAD

Do wykonania platformy testowej zdecydowano się wybrać wariant z zewnętrznymi układami wykonawczymi. Przemawia za tym lepsze wypełnienie warunków postawionych w podrozdziale analizującym model obiektu, a także potencjalnie prostsza w zaprojektowaniu i wykonaniu konstrukcja mechaniczna o mniejszej masie.

Do wykonania elementów mechanicznych robota została użyta technologia FDM (ang. *Fused Deposition Modeling*) [Wikg]. W tym celu wszystkie części oraz użyte podzespoły zostały zamodelowane za pomocą platformy Onshape [PTC]. Proces druku 3D przebiegał przy wykorzystaniu plików G-code wygenerowanych programem Cura [Ult].

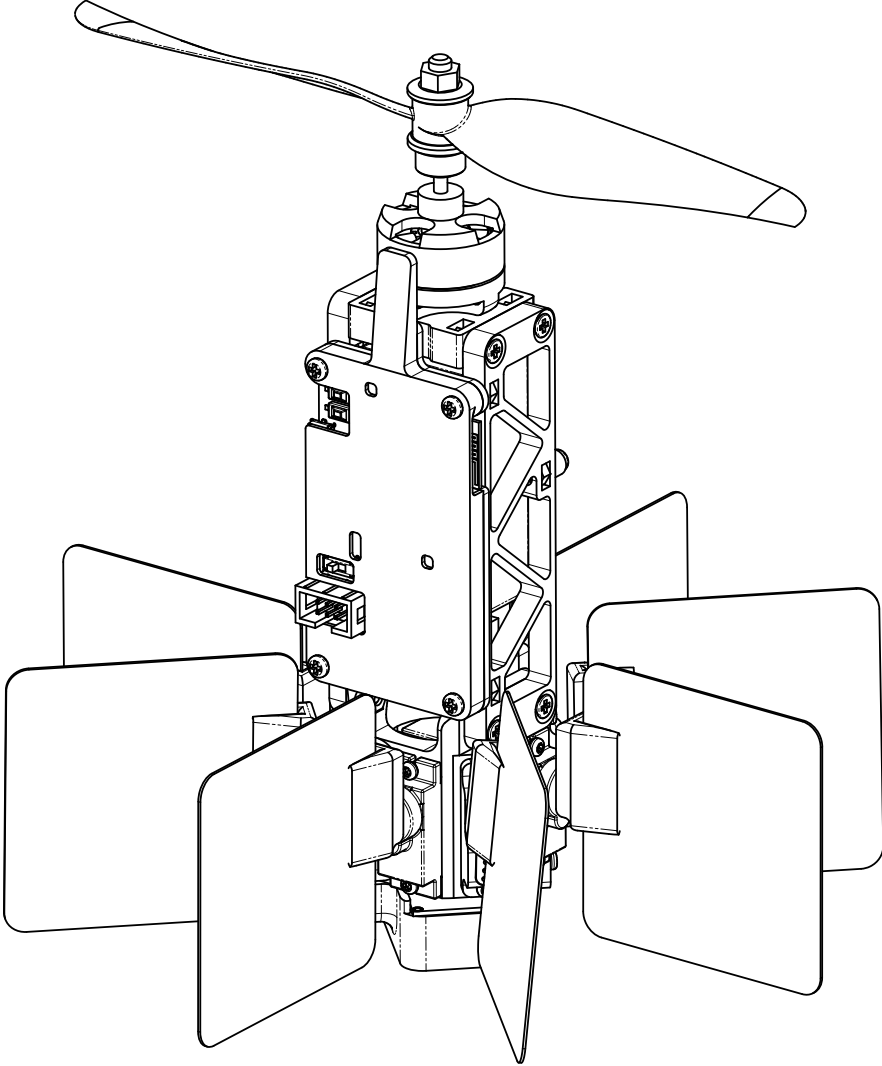
Rysunki techniczne części zaprojektowanych na potrzeby projektu znajdują się w dodatku A. Aby obliczenia położenia środka ciężkości oraz wartości elementów macierzy bezwładności były możliwie zbliżone do rzeczywistych, waga każdego elementu składowego drona została określona za pomocą wagi jubilerskiej oraz nadpisana w programie CAD. Rysunek 4.2 przedstawia kompletną konstrukcję, natomiast rysunek 4.3 prezentuje sposób połączenia jej wyszczególnionych fragmentów: bloku serwomechanizmów, ramy i sekcji dolnej.

Blok serwomechanizmów

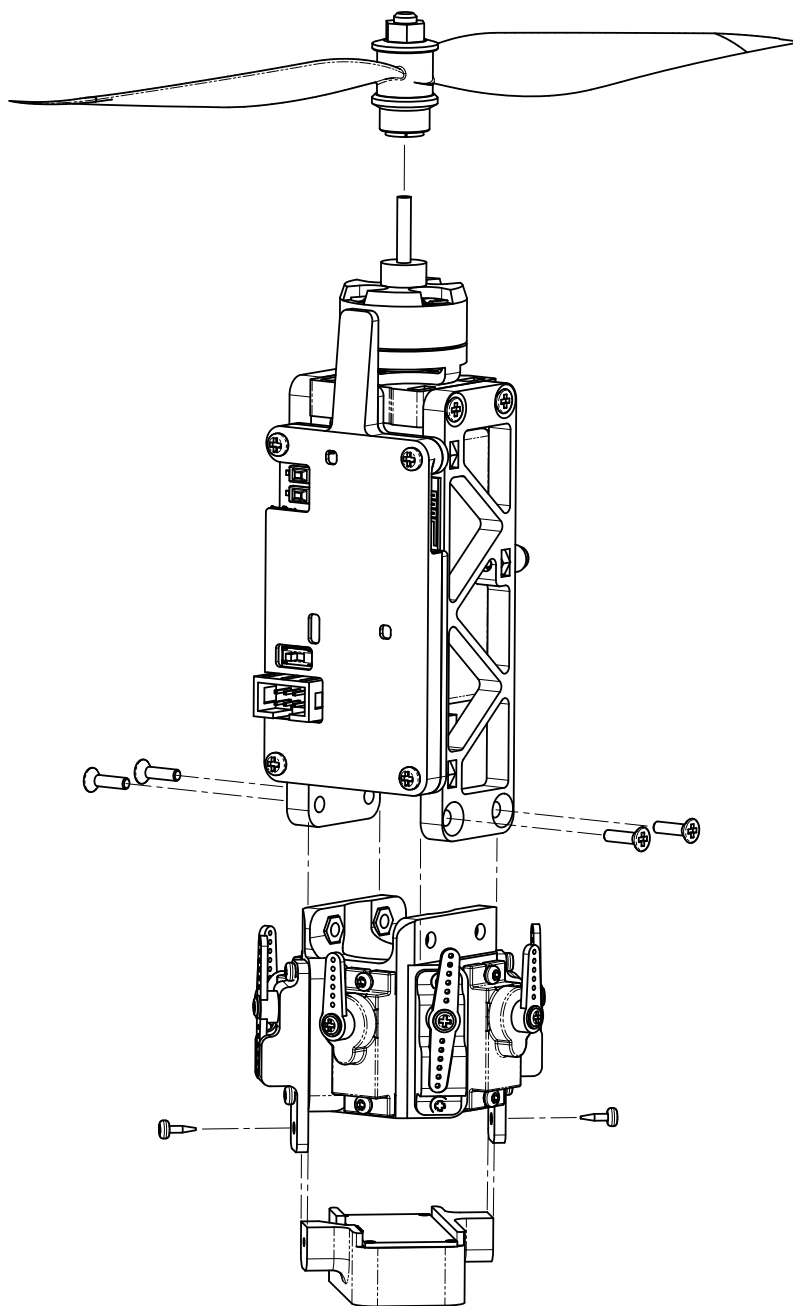
Rysunek 4.4 przedstawia złożenie bloku serwomechanizmów będącego centralnym elementem konstrukcji. Blok promieniście łączy cztery serwomechanizmy lotek umieszczone w specjalnych gniazdach. Z czterech stron bloku znajdują się mocowania lotek pasywnych. Napędy oraz mocowania przytwierdzone są do bloku za pomocą niewielkich wkrętów. Blok łączy ze sobą ramę, sekcję dolną oraz lotki.

Rama

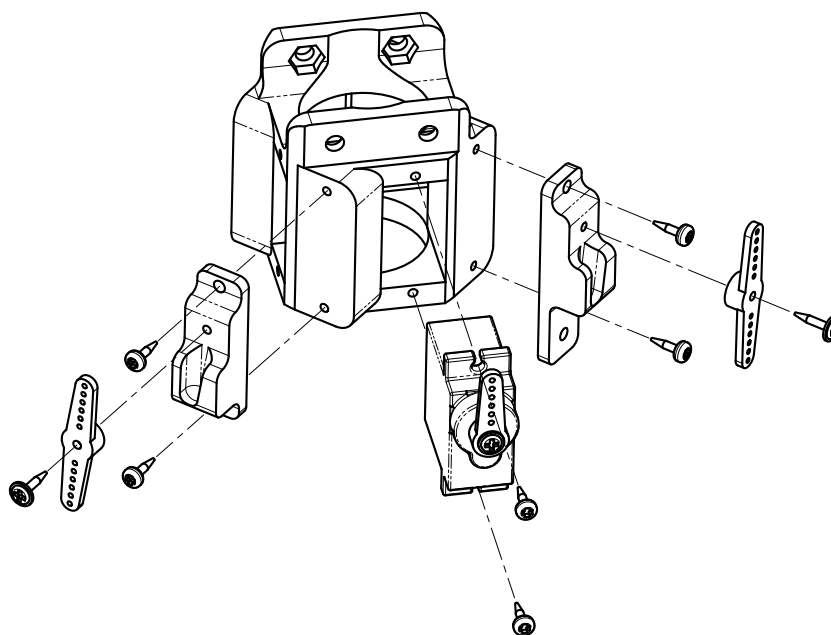
Uzyskanie małej powierzchni przekroju korpusu w płaszczyźnie pracy śmigła wymagało zachowania podłużnej sylwetki konstrukcji. W tym celu została zaprojektowana trzyczęściowa rama o kształcie zbliżonym do litery U, otaczająca baterię. Rama złożona



Rysunek 4.2 Kompletna platforma



Rysunek 4.3 Złożenie głównych elementów konstrukcji



Rysunek 4.4 Złożenie bloku serwomechanizmów

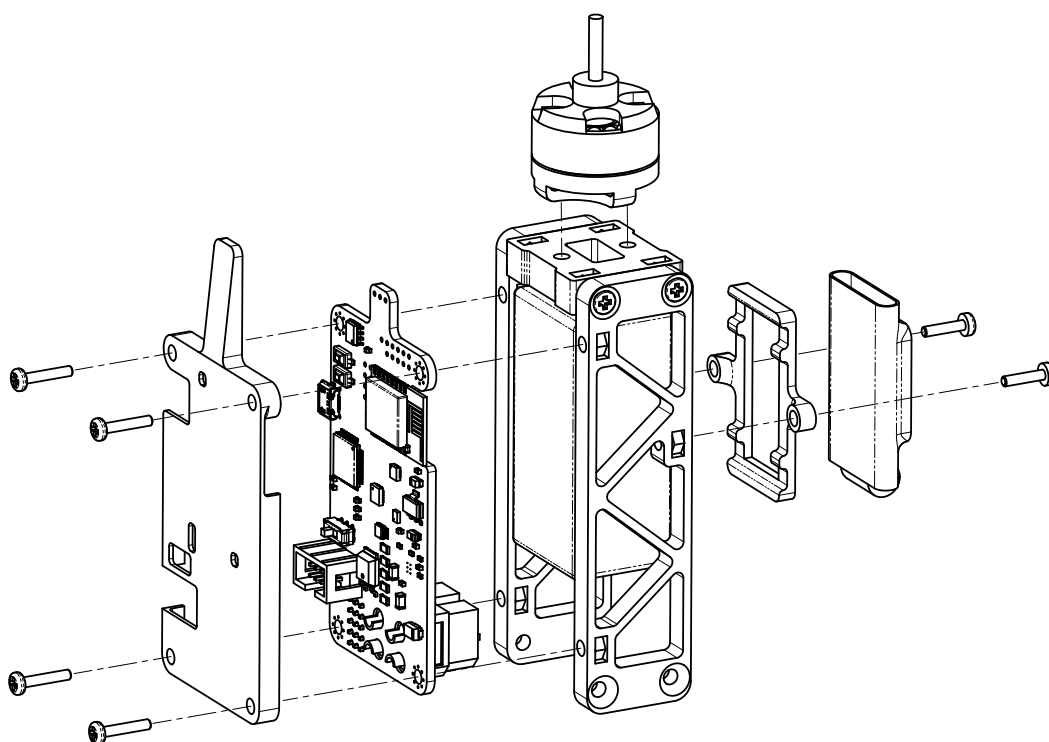
jest z dwóch symetrycznych boków oraz łącznika połączonych czterema śrubami M3. Rysunek 4.5 przedstawia sposób przytwierdzenia sterownika lotu oraz sterownika silnika do ramy. Sterownik lotu został otoczony dodatkową obudową. Silnik, sterownik lotu oraz uchwyt sterownika silnika zostały przytwierdzone do ramy za pomocą śrub i nakrętek M3. Aby zredukować masę, w elementach ramy wykonano specjalne otwory na wzór żeber kratownicy.

Sekcja dolna

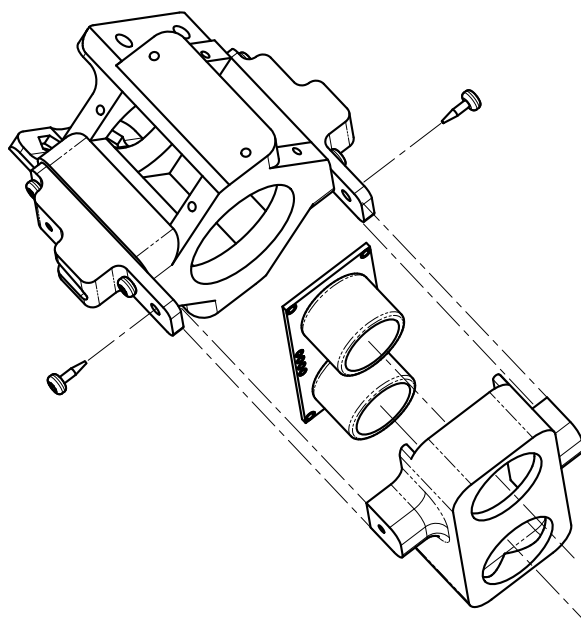
Sekcja dolna składa się z czujnika odległości skierowanego w dół. Element pokazany na rysunku 4.6 osłania czujnik, jednocześnie przytwierdzając go do reszty maszyny. Osłona przymocowana jest do wydłużonych fragmentów mocowań lotek pasywnych, poprzez niewielkie wkręty.

Łopatki

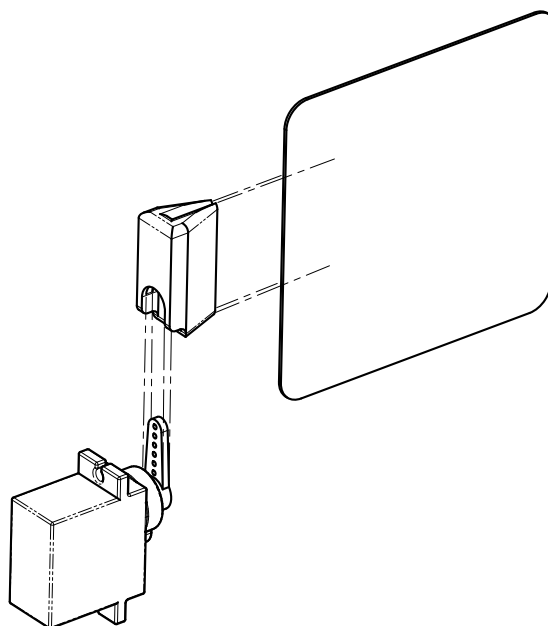
Łopatki sterujące wektorem ciągu zostały wykonane z plastikowego pudełka do produktów spożywczych. Dzięki temu charakteryzują się wystarczającą sztywnością a także łatwością obróbki. Połączenie łopatek z dronem zostało zaprojektowane tak, aby kontakt z podłożem lub inną przeszkodą prowadził do łatwej separacji łopatek od osi serwomechanizmów. Jest to działanie konieczne w celu ochrony podatnych na uszkodzenia plastikowych kół zębatach napędów. Rysunek 4.7 pokazuje montaż łopatki. Łopatka umieszczona jest we wcięciu uchwytu wyprofilowanego tak, aby stabilnie utrzymywał się na orczyku serwomechanizmu. Dowolna siła działająca w górę strąca uchwyt z orczyka, chroniąc wrażliwy napęd. Aby ułatwić proces znajdowania łopatek w trawie lub zaroślach, elementy zostały pomalowane farbą w jaskrawym kolorze.



Rysunek 4.5 Złożenie ramy



Rysunek 4.6 Złożenie sekcji dolnej



Rysunek 4.7 Montaż łopatki

4.2 Układy elektroniczne

W niewielkich obiektach latających rolę nadrzędnego układu sterowania często pełni zintegrowany sterownik lotu stanowiący centralną część ich układów elektronicznych. W tym podrozdziale opisany jest projekt i realizacja układów elektronicznych drona oraz zastosowany protokół komunikacyjny.

4.2.1 Wymagania funkcjonalne

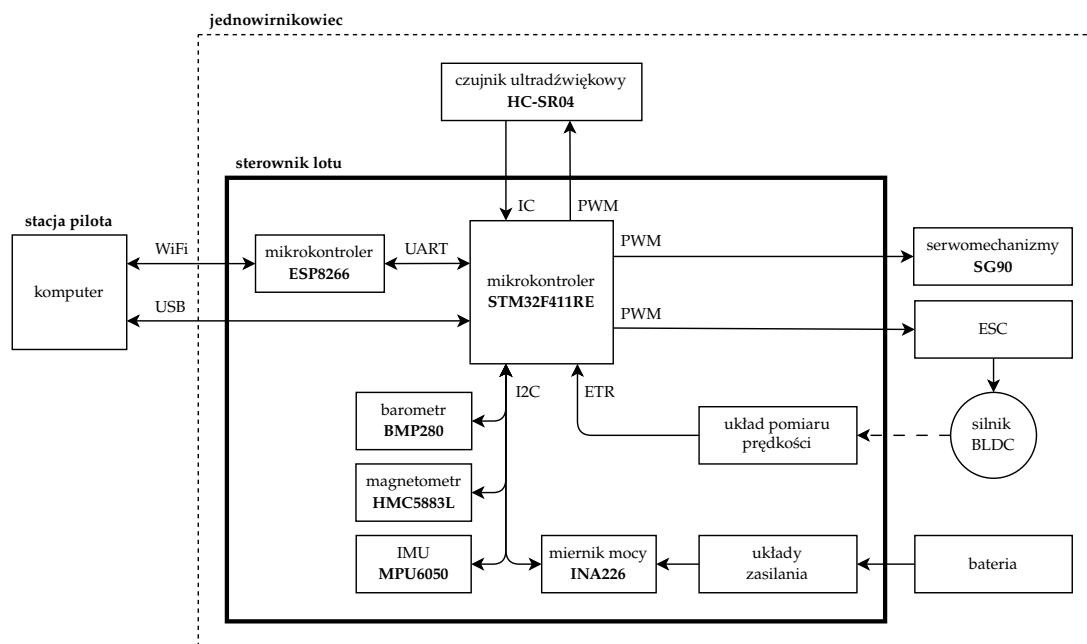
Zadaniem sterownika lotu bezzałogowca jest kontrolowane utrzymanie go w powietrzu przy zachowaniu łączności z pilotem. Aby poprawnie zrealizować to zadanie, sterownik powinien odczytywać dane z układów sensorycznych oraz układów komunikacyjnych, dokonywać ich analizy oraz mieć możliwość sprawnego wpływu na elementy wykonawcze – akulatory robota.

4.2.2 Elementy składowe

Rysunek 4.8 przedstawia schemat układów elektronicznych drona odgrywających rolę sterownika. W skład sterownika wchodzi sensory, układy zasilania oraz sekcja komunikacji i sterowania.

Układy sensoryczne

W celu stabilizacji lotu oraz utrzymania zadanej wysokości nad ziemią dron powinien być wyposażony w czujniki pozwalające na estymację orientacji oraz wysokości przelotowej. Do uzyskania pełnej informacji o położeniu kątowym wymagane będzie



Rysunek 4.8 Diagram układów elektronicznych jednowirnikowca

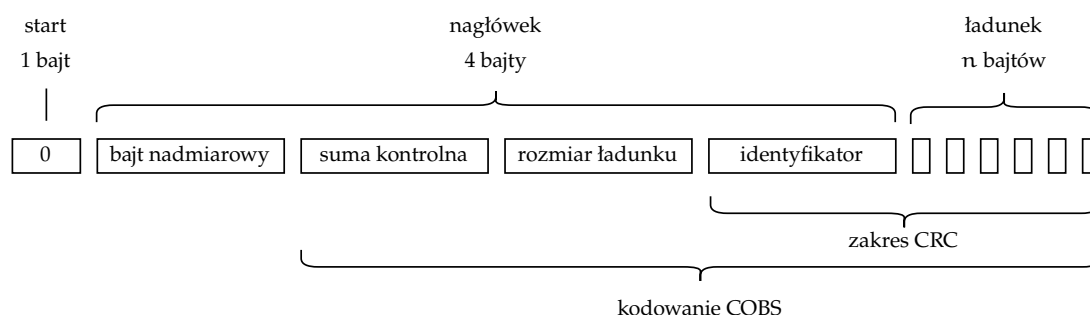
użycie IMU (ang. *Inertial Measurement Unit*) zrealizowanej tu jako połączenie cyfrowego akcelerometru, żyroskopu (układ MPU6050 [Inv]) oraz magnetometru (układ HMC5883L [Hon]). Wysokość lotu drona jest estymowana na podstawie fuzji odczytów barometru (czujnik BMP280 [Bos]), ultradźwiękowego czujnika odległości (moduł HC-SR04) oraz wymienionego wcześniej akcelerometru. Oprócz czujników niezbędnych do utrzymania jednowirnikowca w powietrzu, w sterowniku został umieszczony czujnik mierzący napięcie zasilania oraz całkowity prąd pobierany z baterii (układ INA226 [Texa]), jak i zintegrowany prędkościomierz na bazie czujnika Halla.

Układy wykonawcze

Utrzymanie jednowirnikowca w powietrzu wymaga precyzyjnego sterownia urządzeniami wykonawczymi. Główny wirnik generujący ciąg napędzany jest silnikiem BLDC (ang. *Brushless Direct-Current Motor*) Redox BL400/1400, którego kontrolę umożliwia dedykowany sterownik ESC (ang. *Electronic Speed Controller*). Do kontrolowania kąta natarcia łopatek w strumieniu powietrza, zastosowane zostały cztery serwomechanizmy SG90. Wszystkie wymienione akulatory posiadają standardowy modelarski interfejs wykorzystujący modulację PWM (ang. *Pulse Width Modulation*), w którym sygnał o częstotliwości 50Hz przyjmuje wypełnienie od 1ms do 2ms. Tak określone wartości wypełniania odpowiadają sterowaniu prędkością w zakresie 0–100% (w przypadku silnika BLDC) lub położeniem od -90° do 90° (w przypadku serwowymechanizmów).

Komunikacja

Aby możliwie uprościć złożoność systemu, komunikacja zrealizowana została na bazie lokalnej sieci WiFi tworzonej przez stację pilota. Mikrokontroler ESP8266 [Esp]



Rysunek 4.9 Ramka danych użyta do komunikacji

z oprogramowaniem Blackmagic-espifd [Pal] automatycznie łączy się z siecią o ustalonej nazwie. Do komunikacji bezprzewodowej z dronem użyta została usługa Telnet skojarzona z peryferium UART na mikrokontrolerze ESP. Zastosowanie zdalnej sondy GDB SWD, podłączonej do głównego mikrokontrolera z serii STM32F4, umożliwiło bezprzewodowe programowanie oraz debuggowanie platformy. Dla ułatwienia pracy podczas tworzenia projektu została dodatkowo zaimplementowana możliwość połączenia sterownika lotu bezpośrednio do komputera za pomocą portu USB. Komunikacja przewodowa podlega identycznemu kodowaniu i obsłudze jak wariant radiowy.

W celu uzyskania niezawodnej komunikacji pomiędzy stacją pilota a dronem został opracowany binarny protokół komunikacyjny. Pokazana na rysunku 4.9 ramka danych składa się z trzech części: startu, nagłówka oraz ładunku. Dzięki zastosowaniu kodowania COBS [CB99] (ang. *Consistent Overhead Byte Stuffing*), które pozwala na usunięcie z danych bajtów o wartości 0, dzięki czemu bajt 0 może być wykorzystywany do sygnalizacji początku ramki danych. Bajt nadmiarowy wskazuje miejsce pierwszego występowania zera w następującej po nim sekwencji. Na pozycji tak wskazanego zera, zamiast niego umieszczona zostaje wartość wskazująca względną pozycję kolejnego zera. W przypadku wystąpienia pomiędzy zerami odstępu większego niż 254 bajty, wymagane jest wstawienie dodatkowych bajtów nadmiarowych. Z powodu braku potrzeby wysyłania ładunków dłuższych niż 252 bajty, implementacja została ograniczona do jednego bajtu nadmiarowego.

Możliwość sprawdzenia poprawności otrzymanych danych została zrealizowana poprzez umieszczenie w nagłówku wartości sumy kontrolnej CRC-8. Zakres, z którego liczona jest suma kontrolna, obejmuje identyfikator wiadomości oraz cały ładunek ramki. W razie niezgodności sumy kontrolnej wiadomość jest ignorowana.

4.2.3 Schemat oraz projekt sterownika lotu

Schematy ideowe, jak i projekt płytki drukowanej zostały wykonane w programie KiCad 6 [Cha] i znajdują się w Dodatku B. Przedstawiony na rysunku B.1 schemat główny został podzielony na omówione poniżej sekcje przedstawione w postaci schematów hierarchicznych.

Sekcja sterowania i komunikacji

Jako główna jednostka sterownika został wybrany STM32F411RE firmy STMicroelectronics [STMb]. W celu zapewnienia jego stabilnej pracy w układzie umieszczono kondensatory filtrujące zasilanie zgodnie z zaleceniami producenta [STMa] oraz zewnętrzny oscylator kwarcowy o częstotliwości 8MHz. Wyprowadzenia BOOT0 oraz BOOT1 zostały podłączone tak, by użytkownik mógł wybrać tryb rozruchu mikrokontrolera za pomocą przycisku (wykonywanie programu lub wejście w tryb DFU). Do interakcji jednostki z otoczeniem zostały użyte następujące peryferia:

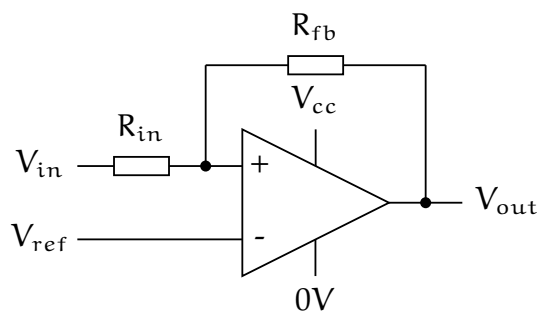
- I²C – konfiguracja i odczyt czujników wewnątrz sterownika lotu,
- UART – komunikacja z ESP8266,
- USB – komunikacja przewodowa oraz programowanie DFU,
- kanały PWM – sterowanie aktuatorami,
- wejście ETR – taktowanie zegara licznika sygnałem prędkościomierza,
- zewnętrzne przerwania z czujników,
- SWD – programowanie oraz debugowanie,
- GPIO – sterowanie kolorowymi diodami informacyjnymi,
- UART – komunikacja z zewnętrznym modułem GPS (brak implementacji),
- SPI – komunikacja z zewnętrznym modułem nRF24L01+ (brak implementacji).

Mikrokontroler ESP8266 stanowi podstawę modułu ESP-12E zawierającego wymagane układy pamięci oraz antenę 2.4GHz umożliwiającą komunikację WiFi. Moduł został zaprogramowany przed wlutowaniem na powierzchnię płytki drukowanej, dzięki czemu wyprowadzenie GPIO0 może być na stałe połączona poprzez rezystor podciągający do potencjału wysokiego. Linie komunikacji UART oraz GPIO emulujące działanie sondy GDB zostały połączone z jednostką główną poprzez zestaw zworek umożliwiających rozwarćcie obwodów do przeprogramowania mikrokontrolerów w razie potrzeby.

Sekcja układów sensorycznych

Z powodu małej dostępności na rynku 9-osiowych IMU został wybrany wariant z osobnymi układami: MPU6050 (akcelerometr oraz żyroskop) oraz HMC5883L (magnetometr). Czujniki zostały połączone do wspólnej magistrali I²C, w której każde urządzenie posiada unikalny adres, dzięki czemu liczba linii sterujących jest zminimalizowana. Sygnały zakończenia konwersji zostały połączone do pinów GPIO jednostki głównej skonfigurowanych jako zewnętrzne przerwania.

Aby mierzyć prędkość obrotową wirnika napędowego w sterowniku lotu, w niewielkiej odległości od silnika został umieszczony czujnik Halla CS49E [F.W]. Podczas obrotu magnesy neodymowe, będące częścią rotora silnika BLDC, wymuszają przenienny strumień magnetyczny przenikający czujnik. Tak wygenerowany sygnał trafia na wejście komparatora LM393 [ons] w konfiguracji z histerezą. Opisana metoda pozwala na wytworzenie sygnału prostokątnego, którego częstotliwość jest proporcjonalna do prędkości obrotowej wirnika. Współczynnikiem proporcjonalności jest liczba par



Rysunek 4.10 Schemat ideowy komparatora z histerezą



Rysunek 4.11 Oscylogram przebiegu sygnału wejściowego i wyjściowego komparatora

pól magnetycznych magnesów w rotorze silnika. Rysunek 4.10 przedstawia schemat ideowy komparatora z histerezą zaimplementowany w układzie. Wartości elementów pasywnych układu zostały dobrane według zależności

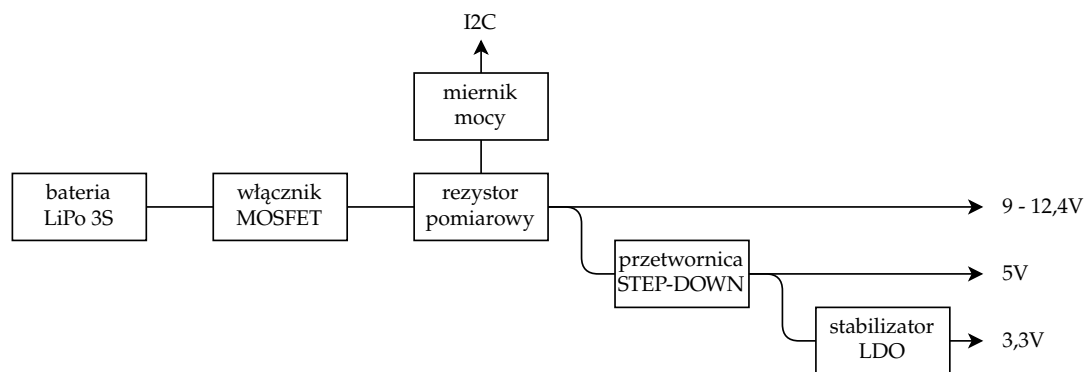
$$\frac{R_{in}}{R_{fb}} = \frac{V_{TH} - V_{TL}}{V_{cc}},$$

$$V_{ref} = \frac{V_{TH} + V_{TL}}{2},$$

gdzie R_{in} oraz R_{fb} oznaczają wartości oporników w układzie, natomiast V_{ref} oznacza napięcie referencyjne, otrzymywane z dzielnika napięcia. Wartości V_{TH} oraz V_{TL} są progami przełączania komparatora odpowiednio przy tranzycji ze stanu niskiego na wysoki oraz z wysokiego na niski. Progi zostały dobrane na podstawie obserwacji przebiegów w układzie podczas obrotu silnika, które przedstawia rysunek 4.11. W rozważaniach przyjęto wartości $V_{TH} = 1,75V$, $V_{TL} = 1,7V$, $R_{fb} = 100k\Omega$, z których wynika w przybliżeniu wartość $R_{in} = 1,5k\Omega$.

Sekcja układów zasilania

Schematyczna budowa sekcji zasilania została przedstawiona na rysunku 4.12. Składa się ona z beziskrowego włącznika zrealizowanego z użyciem tranzystora MOSFET, rezystora pomiarowego, przetwornicy i stabilizatora. Źródłem zasilania jest bateria litowo-polimerowa 3S o pojemności 850mAh.



Rysunek 4.12 Diagram układów zasilania jednowirnikowca

Rezystor pomiarowy służy układowi INA226 do pomiaru prądu pobieranego z baterii, poprzez badanie spadku napięcia na dwójniku. Z powodu małej dostępności profesjonalnych rezystorów pomiarowych, charakteryzujących się małą rezystancją, w układzie zastosowano 5 połączonych równolegle rezystorów 10mΩ.

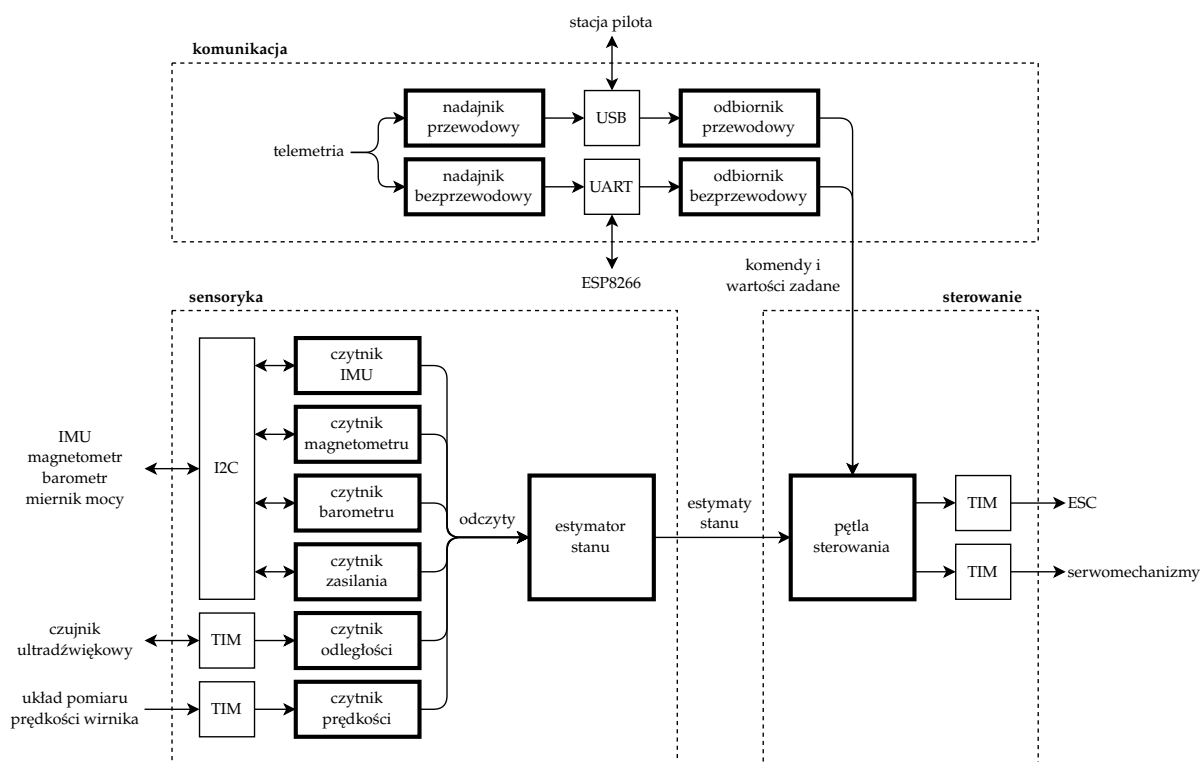
Linia zasilania wirnika (9 ÷ 12,4V) została bezpośrednio wyprowadzona na wyjściowe złącze XT60. Serwomechanizmy oraz ultradźwiękowy czujnik odległości, wymagają napięcia zasilania 5V, do czego służy zmniejszająca napięcie przetwornica z układem MP1584 [Mon], będącym zintegrowanym kluczem tranzystorowym oraz generatorem. Zewnętrzne komponenty pasywne przetwornicy zostały dobrane na bazie przykładu producenta zawartego w nocie katalogowej układu. Zasilanie mikrokontrolerów oraz reszty układów scalonych napięciem 3,3V zostało zrealizowane stabilizatorem liniowym. Dzięki zastosowaniu stabilizatora typu LDO (ang. *Low Dropout*) LM1117 [Texb] możliwe było wykorzystanie linii 5V jako źródła napięcia wejściowego.

4.3 Oprogramowanie

Nietypowy charakter konstrukcji wyklucza zastosowanie gotowego oprogramowania dedykowanego do sterowania bezzałogowymi obiektami latającymi. Z tego względu konieczne było opracowanie własnego oprogramowania umożliwiającego sterowanie jednowirnikowcem oraz komunikację z nim. W tym podrozdziale opisano proces akwizycji danych, estymacji stanu oraz szczegóły implementacji regulatora.

4.3.1 Program sterownika lotu

Program sterownika lotu jednowirnikowego drona bazuje na oprogramowaniu opracowanym przez autora w ramach kursu Sterowniki Robotów [Moż23c, Moź23a, Moź23b]. Dla uproszczenia składni złożonych operacji matematycznych, takich jak operacje macierzowe, głównym językiem w którym napisany jest program jest C++. Zastosowany został system czasu rzeczywistego FreeRTOS [Ama], który stanowi standard w branży Embedded, wraz z nakładką zapewniającą obiektowy interfejs FreeRTOScpp [Dam].



Rysunek 4.13 Diagram architektury programu sterownika lotu

System RTOS (ang. *Real-Time Operating System*) pozwala na emulację wielowątkowości na urządzeniach jednordzeniowych. Możliwie jest to dzięki podziałowi czasu pracy rdzenia na odcinki – kwanty, w tym wypadku o długości 1ms. Na początku każdego kwantu czasu, planista decyduje które zadanie powinno zająć obliczeniami rdzeń, wybierając zadanie o najwyższym priorytecie. Mechanizm usypiania zadań pozwala na zwolnienie rdzenia dla zadań o niższym priorytecie.

Zawarty na rysunku 4.13 diagram przedstawia budowę programu sterownika lotu z podziałem na zadania RTOS. Architektura programu została utworzona tak, aby umożliwić łatwą skalowalność (dodawanie nowych czujników) oraz wprowadzanie modyfikacji (zmiana algorytmu sterowania lub czujników). Program został podzielony na część obsługi układów sensorycznych, estymacji stanu, obsługi komunikacji oraz pętlę sterowania.

Sensoryka

Do obsługi każdego czujnika, w postaci układu scalonego lub zewnętrznego modułu, dedykowane jest osobne zadanie systemu czasu rzeczywistego, realizujące jego początkową konfigurację, odczyt oraz wyliczenie skalibrowanych wartości pomiarów. Zdefiniowano zadania odczytujące czujniki wymienione w tabeli 4.1. Zostały użyte dwa sposoby wybudzania zadań oraz wyzwalania pomiaru: spowodowane zewnętrznym przerwaniem generowanym przez czujnik oraz cykliczne, charakteryzujące się stałym interwałem czasowym.

Zadanie estymacji stanu drona podzielone zostało na dwie części: estymację orienta-

czujnik	wyzwalacz	częstotliwość pracy [Hz]
IMU	przerwanie	200
magnetometr	przerwanie	75
barometr	cykliczne	20
miernik mocy	cykliczne	20
czujnik ultradźwiękowy	przerwanie	50
prędkościomierz	cykliczne	50

Tabela 4.1 Spis obsługiwanych czujników

cji w przestrzeni oraz estymację jego pozycji. Do realizacji obydwu zadań zastosowano rozszerzony filtr Kalmana [Wike]. Umożliwia on fuzję wielu sygnałów z czujników w celu uzyskania bardziej wiarygodnej informacji o niedostępnym stanie obiektu. Proces estymacji można podzielić na dwa etapy: predykcję oraz korekcję. Na etapie predykcji, aby określić przewidywany stan, wykorzystuje się wiedzę o wejściach modelu. W tym celu formułuje się funkcję $f(\mathbf{x}, \mathbf{u})$, gdzie \mathbf{x} jest wektorem stanu systemu a \mathbf{u} jego wejściem, której wartość będzie wskazywała następny krok predykcji. Etap korekcji odpowiednio dopasowuje model tak, aby jego wyjścia pokrywały się z danymi zebranymi z obiektu, poprzez funkcję $h(\mathbf{x})$, która określa wskazania czujników przy danym stanie obiektu. Taki algorytm pozwala na przeprowadzanie wielokrotnego etapu predykcji, gdzie wejściem są dane pochodzące z czujników pracujących z wysoką częstotliwością, zakończonego pojedynczą korekcją.

Estymacja orientacji

Implementacja filtra została zaczerpnięta z bloga [The], gdzie orientację opisano za pomocą kwaternionów. Fuzja łączy wskazania żyroskopu, akcelrometru oraz magnetometru. Model rotacji robota $\mathbf{f}_{\text{rot}}(\mathbf{x}, \mathbf{u})$ opisany jest równaniem

$$\begin{bmatrix} q_{wk} \\ q_{ik} \\ q_{jk} \\ q_{kk} \\ b_{xk} \\ b_{yk} \\ b_{zk} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & \frac{1}{2}\Delta t q_{ik-1} & \frac{1}{2}\Delta t q_{jk-1} & \frac{1}{2}\Delta t q_{kk-1} \\ 0 & 1 & 0 & 0 & -\frac{1}{2}\Delta t q_{wk-1} & \frac{1}{2}\Delta t q_{kk-1} & -\frac{1}{2}\Delta t q_{jk-1} \\ 0 & 0 & 1 & 0 & -\frac{1}{2}\Delta t q_{kk-1} & -\frac{1}{2}\Delta t q_{wk-1} & \frac{1}{2}\Delta t q_{ik-1} \\ 0 & 0 & 0 & 1 & \frac{1}{2}\Delta t q_{jk-1} & -\frac{1}{2}\Delta t q_{ik-1} & -\frac{1}{2}\Delta t q_{wk-1} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_{wk-1} \\ q_{ik-1} \\ q_{jk-1} \\ q_{kk-1} \\ b_{xk-1} \\ b_{yk-1} \\ b_{zk-1} \end{bmatrix} + \frac{1}{2}\Delta t \begin{bmatrix} -q_{ik-1} & -q_{jk-1} & -q_{kk-1} \\ q_{wk-1} & -q_{kk-1} & q_{jk-1} \\ q_{kk-1} & q_{wk-1} & -q_{ik-1} \\ -q_{jk-1} & q_{ik-1} & q_{wk-1} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \omega_{xk-1} \\ \omega_{yk-1} \\ \omega_{zk-1} \end{bmatrix},$$

gdzie elementy q_w , q_i , q_j i q_k tworzą kwaternion reprezentujący rotację jednowirnikowca, natomiast b_x , b_y oraz b_z stanowią wektor przesunięcia wskazań magnetometru.

Podstawą algorytmu jest całkowanie prędkości obrotowych pochodzących z żyroskopu (ang. *Dead Reckoning*). Korekcja osi pionowej uzyskiwana jest na podstawie znormalizowanych danych z akcelerometru, których długość wektor przyspieszenia mieści się w przedziale $[0,97g; 1,03g]$. Takie ograniczenie pozwala na zminimalizowanie wpływu przyspieszeń liniowych na estymatę orientacji. Model akcelerometru $\mathbf{h}_a(\mathbf{x})$ został wyznaczony jako

$$\mathbf{h}_a(\mathbf{x}) = \begin{bmatrix} -2q_j & 2q_k & -2q_w & 2q_i & 0 & 0 & 0 \\ 2q_i & 2q_w & 2q_k & 2q_j & 0 & 0 & 0 \\ 2q_w & -2q_i & -2q_j & 2q_k & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} q_w \\ q_i \\ q_j \\ q_k \\ b_x \\ b_y \\ b_z \end{bmatrix}.$$

Korekcja kierunku północnego możliwa jest dzięki wskazaniom magnetometru, którego model $\mathbf{h}_m(\mathbf{x})$ został wyznaczony jako

$$\mathbf{h}_m(\mathbf{x}) = \begin{bmatrix} -2q_k & -2q_j & -2q_i & -2q_w & 0 & 0 & 0 \\ -2q_w & 2q_i & -2q_j & 2q_k & 0 & 0 & 0 \\ 2q_i & 2q_w & -2q_k & -2q_j & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} q_w \\ q_i \\ q_j \\ q_k \\ b_x \\ b_y \\ b_z \end{bmatrix}.$$

Estymacja pozycji

Estymata pozycji jednowirnikowca pozyskiwana jest na podstawie fuzji odczytów akcelerometru, czujnika odległości oraz barometru. Krok predykcji polega na dwukrotnym całkowaniu przyspieszenia liniowego ciała wyrażonego w układzie świata. Przyspieszenie liniowe obliczane jest z równania

$$\mathbf{a}_{lin} = \mathbf{R}_b^w \mathbf{a}_{pomiar} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix},$$

gdzie \mathbf{R}_b^w jest macierzą obrotu uzyskaną z estymacji orientacji a g to wartość przyspieszenia grawitacyjnego na powierzchni ziemi. Z powodu występowania znacznych fluktuacji ciśnienia atmosferycznego w czasie lotu, konieczna okazała się być również estymacja ciśnienia referencyjnego jako jednego z elementów wektora stanu. Dyskretny model przemieszczenia ciała $\mathbf{f}_{lin}(\mathbf{x}, \mathbf{u})$ został wyrażony jako

$$\begin{bmatrix} z_k \\ \dot{z}_k \\ p_{0k} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_{k-1} \\ \dot{z}_{k-1} \\ p_{0k-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \\ 0 \end{bmatrix} \mathbf{a}_{zk-1},$$

gdzie Δt oznacza okres próbkowania akcelerometru, a_z jest składową pionową wektora przyspieszenia liniowego \mathbf{a}_{lin} a p_0 to ciśnienie atmosferyczne na poziomie gruntu, będące wartością referencyjną podczas fuzji odczytów barometru. Model został zdefiniowany na podstawie analizy ruchu jednostajnie przyspieszonego wzdłuż pionowej osi układu współrzędnych.

Etap korekcji estymaty pozycji jest przeprowadzany na podstawie danych przychodzących z dwóch źródeł. Pierwszym z nich jest ultradźwiękowy czujnik odległości skierowany w dół, którego model ma postać

$$h_d(\mathbf{x}) = [1 \ 0 \ 0] \begin{bmatrix} z \\ \dot{z} \\ p_0 \end{bmatrix} = z.$$

Drugim elementem umożliwiającym etap korekcji jest cyfrowy barometr. Z powodu nieliniowej zależności wysokości od ciśnienia zastosowano rozszerzony filtr Kalmana. Model barometru $h_p(\mathbf{x})$ został wyrażony wzorem barometrycznym jako [Wiki]

$$h_p(\mathbf{x}_k) = p_0 \left(1 - \frac{z}{44330}\right)^{5,255}.$$

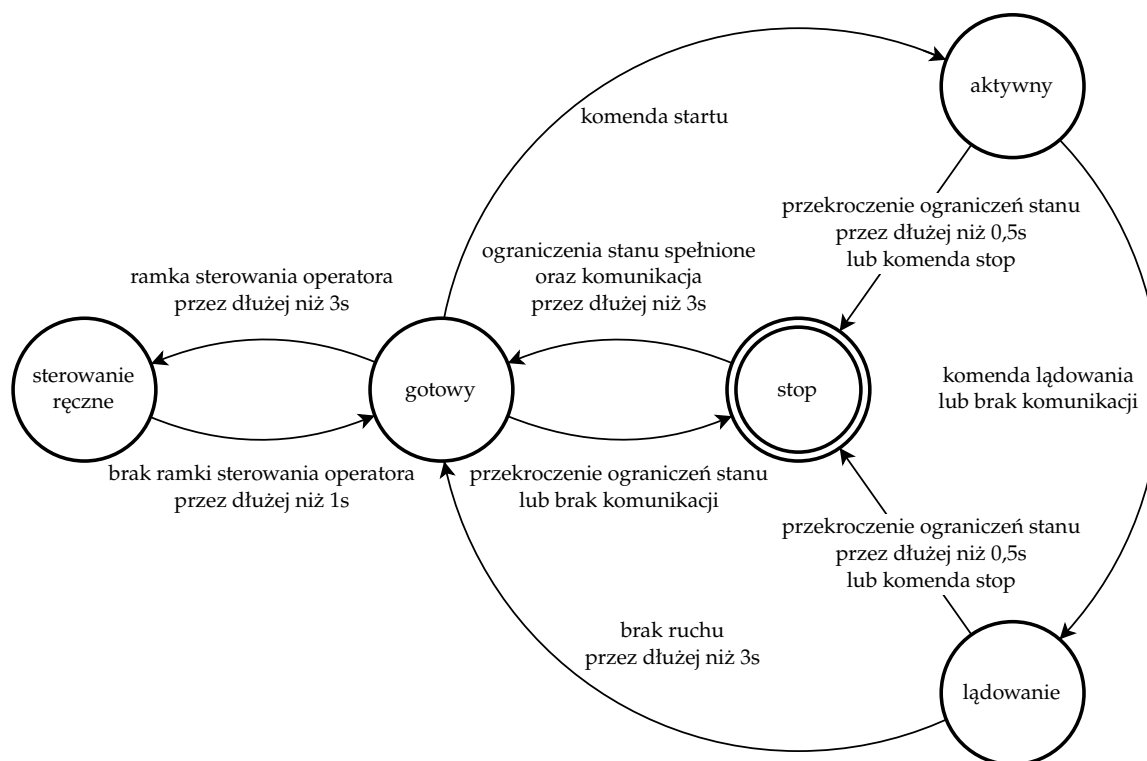
Komunikacja

Dwustronna komunikacja drona ze stacją pilota została zrealizowana dwoma kanałami. Każdy kanał zbudowany jest z dwóch zadań systemu czasu rzeczywistego. Nadajnik pobiera dane z kolejki nadawczej oraz wysyła je przez skojarzone peryferium. Odbiornik dokonuje nasłuchu i odczytu peryferium, a następnie umieszcza tak pobrane wiadomości w kolejce odbiorczej. Podstawowym, używanym podczas lotu, jest kanał bezprzewodowy realizowany za pomocą peryferium UART oraz mikrokontrolera ESP8266, tak jak opisano to w podrozdziale 4.2.1. Kanał przewodowy, szczególnie poręczny podczas wczesnej fazy rozwoju sterownika lotu, wykorzystuje wbudowane peryferium USB OTG FS, umożliwiające bezpośrednie połączenie z komputerem. Podczas nadawania informacji do stacji pilota podejmowana jest próba użycia każdego z dostępnych kanałów, dzięki czemu nie ma potrzeby wybierania aktywnego kanału komunikacji.

Interfejs pozwala na wysyłanie dwóch głównych typów wiadomości: ramek telemetrycznych oraz wiadomości tekstowych – logów. Telemetria, przesyłana w postaci binarnej, to zbiór wiadomości nadawanych przez poszczególne zadania, zawierających informacje liczbowe na temat pomiarów, estymat i sterowań. Kod źródłowy opisujący strukturę poszczególnych ramek jest współdzielony podczas procesu kompilacji programu mikrokontrolera oraz aplikacji stacji pilota. Wiadomości tekstowe skierowane są do operatora, informują o stanie maszyny lub wyniku wykonywanych operacji.

Sterowanie

Zarządzanie trybami pracy urządzenia zostało zrealizowane automatem skończonym, którego diagram przejść między stanami wraz z warunkami tranzycji przedstawia rysunek 4.14. Wyróżniono następujące stany urządzenia:



Rysunek 4.14 Automat stanów sterownika lotu

- stop – natychmiastowe zatrzymanie urządzeń wykonawczych,
- gotowy – stan pośredni, powolne zatrzymanie elementów ruchomych,
- aktywny – dążenie do utrzymania zadawanego stanu,
- lądowanie – powolne obniżenia zadanej wysokości lotu,
- sterowanie ręczne – bezpośrednie sterowanie aktuatorami przez operatora.

Podczas lotu (stan aktywności i lądowania) realizowane jest sterowanie ze sprzężeniem zwrotnym, szerzej opisane w rozdziale 3. Stany zatrzymania (stop) oraz gotowości zostały zaimplementowane w celu zwiększenia bezpieczeństwa pracy z maszyną. Dane dotyczące aktualnego stanu obiektu oraz wartości zadane stanu są pobierane z odpowiednich kolejek. Człon całkujący regulatora działa z częstotliwością 200Hz, a sterowanie zadawane jest z częstotliwością 100Hz. Skrypt w języku Python generuje plik nagłówkowy zawierający wartości macierzy regulatora K oraz wektora punktu pracy urządzeń wykonawczych. Z powodu występowania gwałtownych sterowań na początku pracy regulatora, na wirnik został nałożony ogranicznik narastania i opadania sygnału (rampa). Na łopatkach zostało nałożone ograniczenie maksymalnego wychylenia o $\pm 15^\circ$ od pionu.

4.3.2 Program stacji pilota

Stacja pilota oraz wszystkie inne programy okienkowe opracowane na potrzeby projektu została napisana z użyciem biblioteki Qt [Gro]. Program opisany w tym podrozdziale zawiera fragmenty kodu źródłowego rozwianego przez autora w ramach

typ	kolor wyświetlania	zastosowanie
error	czerwony	błędy komunikacji i odczytów
warning	żółty	odczyty z poza przewidywanego zakresu
info	niebieski	inicjalizacja, tranzycja stanu
debug	szary	testy

Tabela 4.2 Typy komunikatów

kursu Wizualizacja Danych Sensorycznych [DM23]. Zadaniem aplikacji stacji pilota są:

- nawiązanie komunikacji z dronem poprzez dostępne kanały (USB lub WiFi),
- wizualizacja danych telemetrycznych,
- wyświetlenie komunikatów tekstowych,
- odczyt sterowań użytkownika (kontroler USB lub interfejs graficzny).

Komunikacja odbywa się z użyciem klasy `QSerialPort` oraz `QTcpSocket` odpowiednio dla komunikacji przewodowej oraz bezprzewodowej. Interfejs użytkownika pozwala na zmianę domyślnego portu COM oraz adresu IP mikrokontrolera ESP8266. Do odczytu zewnętrznego kontrolera USB została użyta klasa `QGamepad`.

Wyświetlanie komunikatów tekstowych sprowadza się do wysłania tekstu na standardowe wyjście (terminal) poprzedzone kodem wyjścia terminala [Wikb] służącym do zmiany koloru tekstu. Typy komunikatów tekstowych zostały opisane w tabeli 4.2.

Najważniejsze elementy stanu robota są wizualizowane za pomocą wykresów odświeżanych w czasie rzeczywistym. Są to sterowania (kąty wychylenia łopatek, siła ciągu), pozycja (orientacja i wysokość nad ziemią) oraz prędkość (kątowna i składowa pionowa liniowej). Wykresy zostały opracowane się z użyciem klasy `QChart`. W przypadkach występowania więcej niż jednej serii danych na jednym wykresie, stosowane są kolory dla rozróżnienia wartości. Liniami przerywanymi zostały oznaczone wartości zadane regulatora.

4.3.3 Programy do kalibracji

Do budowy robota zastosowano elementy łatwo dostępne na rynku oraz charakteryzujące się niską ceną. Przyspiesza i ułatwia to proces wymiany elementów, uwzględniając budżet i czas dostawy. Jednak często wiąże się z tym także ich niewielka dokładność, wysoka awaryjność oraz duża rozbieżność parametrów poszczególnych egzemplarzy tego samego produktu. Z tego powodu koniecznym okazało się zastosowanie technik kalibracji czujników, jak i urządzeń wykonawczych w celu osiągnięcia zadowalających wyników.

Program do kalibracji magnetometru

Magnetometr jest urządzeniem mierzącym wektor pola magnetycznego w danym miejscu w przestrzeni. Jednak sam fakt umieszczenia takiego czujnika wewnątrz ro-

bota zniekształca jego odczyty. Zastosowany magnetometr nie posiada wbudowanych mechanizmów kalibracji, dlatego należało takie zaimplementować.

W idealnym przypadku wartości wektora pola magnetycznego dla każdej orientacji urządzenia zakreślałyby sferę o środku w początku układu współrzędnych, jednak przed dokonaniem kalibracji w rzeczywistości powstaje elipsoida. Wyróżniane są dwa rodzaje zniekształceń pola magnetycznego: twarde (ang. *Hard Iron Distortions*) i miękkie (ang. *Soft Iron Distortions*) [Vec]. Zniekształcenia twarde powodują przesunięcie odczytów o stałą wartość. Natomiast zniekształcenia miękkie wpływają na skalę i kierunek pola magnetycznego. Poniższa formuła pozwala na uzyskanie informacji niezawierającej zniekształceń

$$\mathbf{m}_{\text{corr}} = \mathbf{A}(\mathbf{m}_{\text{raw}} - \mathbf{b}),$$

gdzie \mathbf{m}_{corr} jest skalibrowaną wartością odczytu, \mathbf{m}_{raw} wartością odczytaną z czujnika, \mathbf{A} to stała macierz transformacji przestrzeni, natomiast \mathbf{b} to stały wektor przesunięcia przestrzeni.

Aby określić macierz \mathbf{A} i wektor \mathbf{b} zaimplementowano algorytm znajdowania najlepiej dopasowanej elipsoidy do danych [Jud], bazujący na metodzie najmniejszych kwadratów. Powierzchnię elipsoidy można opisać równaniem

$$C_1x^2 + C_2y^2 + C_3z^2 + C_4xy + C_5xz + C_6yz + C_7x + C_8y + C_9z = 1,$$

gdzie $C_i, i = 1, \dots, 9$ są współczynnikami elipsoidy. Dowiedziono, że optymalny wektor współczynników $\mathbf{C} = [C_1, \dots, C_9]^T$ można uzyskać z następującej zależności:

$$\mathbf{C} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \mathbf{K},$$

gdzie

$$\mathbf{J} = \begin{bmatrix} x_1^2 & y_1^2 & z_1^2 & x_1y_1 & x_1z_1 & y_1z_1 & x_1 & y_1 & z_1 \\ x_2^2 & y_2^2 & z_2^2 & x_2y_2 & x_2z_2 & y_2z_2 & x_2 & y_2 & z_2 \\ x_3^2 & y_3^2 & z_3^2 & x_3y_3 & x_3z_3 & y_3z_3 & x_3 & y_3 & z_3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N^2 & y_N^2 & z_N^2 & x_Ny_N & x_Nz_N & y_Nz_N & x_N & y_N & z_N \end{bmatrix},$$

\mathbf{K} to wektor N -wymiarowy jedynek, a \mathbf{J} to macierz $N \times 9$ powstała z danych, N to liczba próbek, dla których szukana jest najlepiej dopasowana elipsoida, wynik uzyskuje się dla $N \geq 9$. Na podstawie znalezionej macierzy \mathbf{C} możliwe jest obliczenie położenia, rozmiarów i orientacji elipsoidy. Wektor \mathbf{b} bezpośrednio odpowiada położeniu środka elipsoidy. Macierz \mathbf{A} uzyskiwana jest z zależności

$$\mathbf{A} = \mathbf{R}_e^T \begin{bmatrix} \frac{1}{L_x} & 0 & 0 \\ 0 & \frac{1}{L_y} & 0 \\ 0 & 0 & \frac{1}{L_z} \end{bmatrix} \mathbf{R}_e$$

gdzie \mathbf{R}_e to macierz rotacji elipsoidy, a \mathbf{L} wektor, którego wartości reprezentują długość, szerokość i wysokość elipsoidy wyrażone w jej układzie współrzędnych.

Aby zautomatyzować proces próbkowania i obliczeń napisano odrębny program okienkowy z wykorzystaniem biblioteki Eigen [JG], pozwalającej na zaawansowane

operacje macierzowe. Podczas gdy użytkownik obraca dronem w wielu kierunkach, program akumuluje kolejne próbki, a po otrzymaniu nowej próbki, wysyła obliczone wartości macierzy \mathbf{A} i wektora \mathbf{b} na standardowe wyjście. Jednocześnie wizualizuje zebrane dane (rzuty na płaszczyzny XY, YZ oraz ZX) w dwóch podglądach: przed i po kalibracji, w celu ułatwienia walidacji współczynników.

Program do kalibracji akcelerometru

Odczyty akcelerometru zostały przeskalowane oraz przesunięte tak, aby każda oś podczas oddziaływania przyspieszenia grawitacyjnego w jej kierunku (o obu zwrotach) wskazywała wartość $\pm g$. W tym celu, aby uzyskać skalibrowane dane, należy zastosować formułę

$$\mathbf{a}_{\text{corr}} = \mathbf{A}(\mathbf{a}_{\text{raw}} - \mathbf{b}),$$

gdzie \mathbf{a}_{corr} to skalibrowane dane, \mathbf{a}_{raw} reprezentuje odczyty czujnika, \mathbf{b} to wektor przesunięcia a \mathbf{A} to macierz zdefiniowana jako

$$\mathbf{A} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix}.$$

Wartości współczynników s_x, s_y, s_z oraz wektora \mathbf{b} można obliczyć z zależności

$$b_i = \frac{a_{\text{rawimax}} + a_{\text{rawimin}}}{2}, \quad s_i = \frac{g}{a_{\text{rawimax}} - b_i},$$

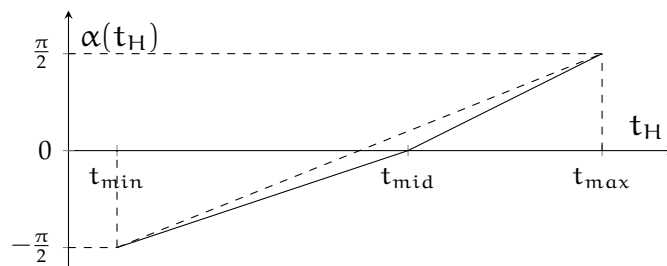
gdzie a_{rawimax} oraz a_{rawimin} są odpowiednio maksymalną i minimalną wartością odczytaną z i -tej osi czujnika w spoczynku.

Do obliczeń napisano program okienkowy, pozwalający na zapis 6 skrajnych orientacji robota. Program automatycznie wyświetla obliczone wartości macierzy \mathbf{A} oraz wektora \mathbf{b} na standardowym wyjściu.

Program do kalibracji serwomechanizmów

W sterowaniu łopatkami jednowirnikowca dużą rolę odgrywa poprawne działanie serwomechanizmów, bez którego obiekt nie ma możliwości utrzymania stabilnego lotu. Z tego powodu wymagane było wprowadzenie kompensacji nieliniowej pracy serwomechanizmów. Szczególną uwagę należało zwrócić na punkt pracy lotek (ustawienie pionowe).

Zaproponowano, aby dla każdego serwomechanizmu, znaleźć wartości czasu trwania stanu wysokiego w jednym okresie sygnału PWM (t_H) dla trzech punktów: skrajnego ujemnego ($t_{\text{min}}, -\frac{\pi}{2}$), pionowego ($t_{\text{mid}}, 0$) oraz skrajnego dodatniego ($t_{\text{max}}, \frac{\pi}{2}$). Powyższe punkty wybrano tak, aby w łatwy sposób można było określić osiągnięcie danego położenia bez dodatkowych układów pomiarowych. Zdefiniowano krzywą pracy serwomechanizmu widoczną na rysunku 4.15. Do tak zdefiniowanej krzywej pracy znaleziono funkcję odwrotną, umożliwiającą kompensację nieliniowości urządzenia,



Rysunek 4.15 Kąt wychylenia serwomechanizmu od czasu wypełnienia PWM

zdefiniowaną jako

$$t_H(\alpha) = \begin{cases} \frac{t_{\max} - t_{\text{mid}}}{\frac{\pi}{2}} \alpha + t_{\text{mid}}, & \text{dla } \alpha \geq 0 \\ \frac{t_{\text{mid}} - t_{\min}}{\frac{\pi}{2}} \left(\alpha + \frac{\pi}{2} \right) + t_{\min}, & \text{dla } \alpha < 0 \end{cases}$$

Aby ułatwić kalibrację na rzeczywistym obiekcie, został przygotowany dedykowany program okienkowy. Za pomocą 12 suwaków (trzy dla każdego serwomechanizmu) możliwe jest szybkie znalezienie wyżej omówionych punktów.

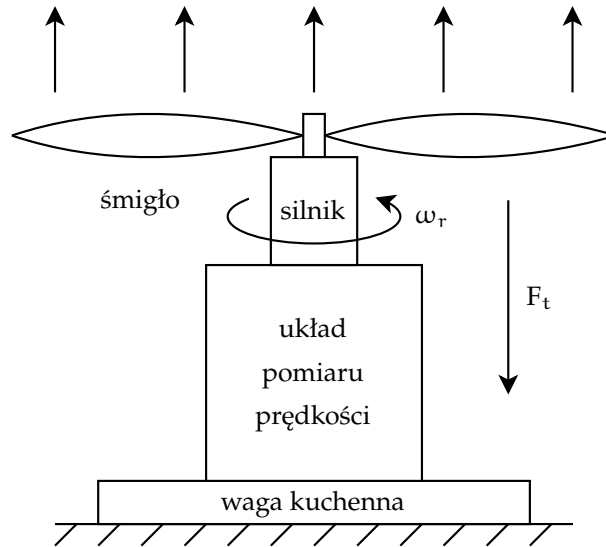
4.4 Identyfikacja parametrów modelu

Aby wdrożyć układ sterowania stworzony na podstawie modelu, należy znać parametry takiego modelu. Z kolei program CAD jest w stanie dostarczyć informacji na temat współrzędnych środka ciężkości oraz macierzy bezwładności wybranych grup elementów. Mimo to dalej pozostają współczynniki zależne od zbyt dużej ilości niewiadomych, które często najprościej jest estymować na podstawie eksperymentu. W tym podrozdziale opisano proces pozyskiwania brakujących współczynników modelu poprzez przeprowadzenie eksperymentów identyfikujących. Wszystkie pomiary oraz wyniki zawarte zostały w dodatku C.

4.4.1 Identyfikacja współczynnika ciągu i prędkości obrotowej

W rozdziale 2 zamodelowano siłę ciągu F_t jako proporcjonalną do sterowania u_t , gdzie K_t jest stałą proporcjonalności. Jednocześnie w pracach [Jac21, CFAN16] ciąg modelowany jest również jako siła proporcjonalna do kwadratu prędkości obrotowej wirnika ω_r , gdzie stałą proporcjonalności jest K_ω . Opisane stałe zależą od czynników takich jak:

- geometria śmigła,
- parametry silnika,
- możliwości ESC,
- napięcie i rezystancja wewnętrzna baterii,
- temperatura i wilgotność powietrza.



Rysunek 4.16 Budowa stanowiska testowego do identyfikacji K_t oraz K_ω

Dlatego opracowano eksperyment umożliwiający pomiar siły ciągu oraz prędkości obrotowej wirnika dla całego zakresu sterowań. Budowa stanowiska została ukazana na rysunku 4.16. Podstawą jest starowana waga kuchenna, która mierzy nacisk spowodowany wystąpieniem siły ciągu skierowanej w górę, obliczanej jako

$$F_{ti} = \frac{m_i}{1000}g,$$

gdzie m_i to wartość i -tej siły nacisku na wagę wyrażona w gramach. Do pomiaru prędkości obrotowej wirnika zastosowano układ analogiczny do opisanego w podrozdziale 4.2.3.

Poszukiwanie optymalnie dobranych współczynników K_t i K_ω zostało zrealizowane metodą najmniejszych kwadratów. Zdefiniowano następujące zadania optymalizacyjne:

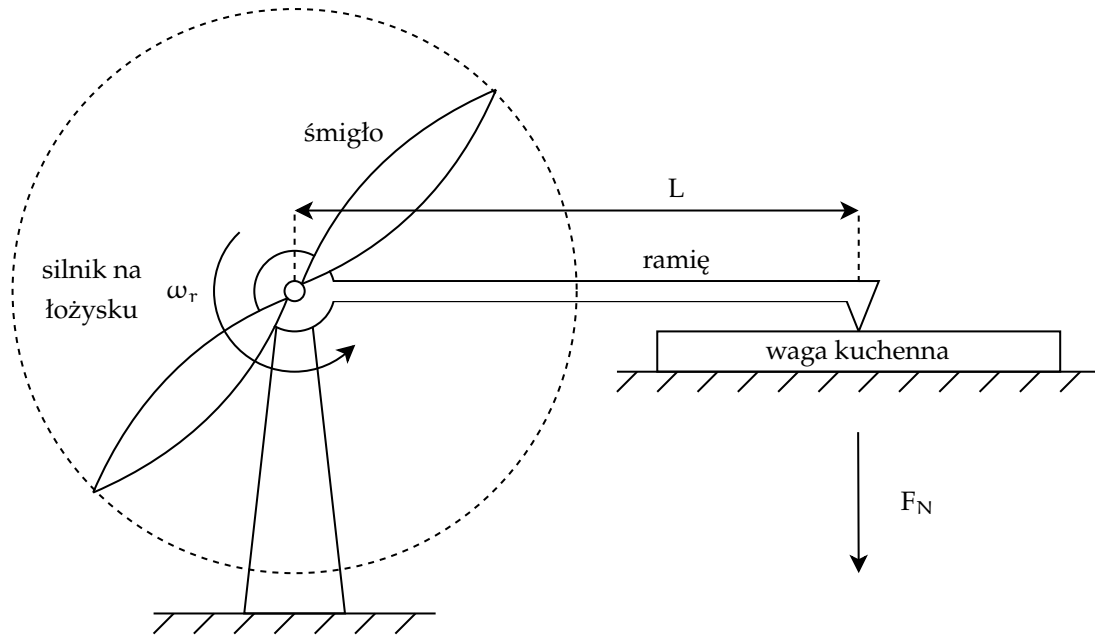
$$\min_{K_t} \sum_{i=1}^N (F_{ti} - K_t u_{ti})^2, \quad \min_{K_\omega} \sum_{i=1}^N (F_{ti} - K_\omega \omega_{ri}^2)^2,$$

których rozwiązania można przedstawić jako odpowiednio

$$K_t = \frac{\sum_{i=1}^N F_{ti} u_{ti}}{\sum_{i=1}^N u_{ti}^2}, \quad K_\omega = \frac{\sum_{i=1}^N F_{ti} \omega_{ri}^2}{\sum_{i=1}^N \omega_{ri}^4}.$$

4.4.2 Identyfikacja współczynnika momentu obrotowego

W rozdziale 2 zamodelowano moment obrotowy powstały w wyniku pracy wirnika M_r jako proporcjonalny do wartości sterowania u_t , gdzie stałą proporcjonalności jest współczynnik K_τ . Aby umożliwić zebranie wymaganych danych (wartości momentu) przygotowano stanowisko przedstawione na rysunku 4.17. Cechą szczególną stanowiska jest silnik zamontowany w sposób zapewniający swobodny obrót statora w osi obrotu śmigła. Jednocześnie do statora przytwierdzone zostało ramię stykające się

Rysunek 4.17 Budowa stanowiska testowego do identyfikacji K_τ

ze starowaną wagą kuchenną. W wyniku obrotu śmigła generowany moment obrotowy zostanie przekształcony na siłę naciskającą na wagę. Metoda pozwala na obliczenie momentu obrotowego z zależności

$$M_{ri} = \frac{m_i}{1000} gL,$$

gdzie M_{ri} to moment obrotowy generowany w i -tym pomiarze, m_i to odnotowana w nim wartość siły nacisku na wagę wyrażona w gramach a L to długość ramienia, która na potrzeby eksperymentu wynosiła 0,2m. Należy zwrócić szczególną uwagę na sposób interpretacji wyników pomiarów. Wyżej opisany układ pomiarowy wprowadza błąd systematyczny w postaci momentu obrotowego tarcia łożyska. Dlatego w tym przypadku dopasowano prostą o równaniu

$$M_r(u_t) = K_\tau u_t - M_0,$$

gdzie M_0 jest momentem obrotowym potrzebnym do przezwyciężenia tarcia łożyska. Podczas dopasowania należy zignorować wszystkie pomiary o wartości $M_r = 0$, ponieważ oznacza to, że wygenerowany moment nie był w stanie poruszyć układem pomiarowym. W celu znalezienia wartości K_τ został zdefiniowany problem optymalizacyjny w postaci

$$\min_{K_\tau, M_0} \sum_{i=1}^N (M_{ri} - K_\tau u_{ti} + M_0)^2,$$

którego rozwiązaniem jest

$$K_\tau = \frac{N \sum_{i=1}^N u_{ti} M_{ri} - \sum_{i=1}^N u_{ti} \sum_{i=1}^N M_{ri}}{N \sum_{i=1}^N u_{ti}^2 - \left(\sum_{i=1}^N u_{ti} \right)^2}.$$

Rozdział 5

Eksperymenty

Ocena zdolności do lotu konstrukcji jest podstawą weryfikacji osiągnięcia celu projektu. W tym rozdziale opisane zostały eksperymenty przeprowadzone na platformie testowej oraz wnioski płynące z otrzymanych wyników.

5.1 Zdolność do poruszania się

Przeprowadzono szereg testów udowadniających zdolność do generacji sił i momentów wymaganych do sterowania obiektem. Jednowirnikowca umieszczono w stacjonarnym uchwycie umożliwiającym mu niewielki zakres ruchu. Poprzez sterowanie w trybie ręcznym wirnikiem oraz łopatkami sprawdzono zdolność do generacji:

- siły ciągu umożliwiającej uniesienie drona,
- momentów obrotowych w osiach X i Y umożliwiających wychylenie od pionu,
- momentu obrotowego w osi Z umożliwiającego obrót w pionie.

Każde ze sterowań było w stanie przemieścić robota pomiędzy odpowiadającym im skrajnym położeniem w uchwycie. Zgodnie z przewidywaniami obrót w osiach X i Y ujawniał wpływ efektu żyroskopowego na drona. Nawet mimo zmiany nachylenia jednej pary przeciwległych łopatek obiekt poruszał się na skos, co pokrywało się z wynikami symulacji. Testy wykazały, że podniesienie łopatek w ich uchwytach (zbliżenie łopatek do wirnika) nie zwiększa znacząco generowanych momentów obrotowych i przyczynia się do skrócenia ramienia działania siły oraz większego zużycia napędów.

5.2 Lot swobodny

Wykonano dziesiątki testów sprawdzających zachowanie robota podczas lotu w powietrzu. Po rozwiązaniu problemów z układami sensorycznymi oraz po kalibracji napędów łopatek jednowirnikowiec zaczął zachowywać stabilność podczas lotu.

Układ sterowania był w stanie ustabilizować obiekt na zadanej wysokości pod warunkiem, że orientacja drona była zbliżona do pionu, a bateria była dostatecznie naładowana. Ze względu na ostateczną wagę drona w zawisie silnik wirnika pracował blisko

swoich maksymalnych obrotów, co znacznie ogranicza możliwości zwiększania siły ciągu. Sterowanie prędkością kątową drona w osi Z przebiegło pomyślnie, jednak nie udało się znaleźć takich współczynników macierzy \mathbf{Q} , aby wyeliminować drobne oscylacje. Układ sterowania był w stanie utrzymać maszynę w pionie (punkt równowagi), jednak zadanie innej orientacji powodowało utratę stabilności układu. Uniemożliwiło to lot w zadanych przez pilota kierunkach.

Można stwierdzić, że efekt żyroskopowy został zamodelowany poprawnie, ponieważ zaobserwowana korelacja między uchybem w osiach X i Y była marginalna. Niedoskonałości konstrukcji mechanicznej sprawiały, że jednowirnikowiec miał tendencje do powolnego lotu w jedną ze stron. Umożliwiło to wprowadzenie alternatywnego sposobu kontroli kierunku lotu, poprzez obracanie się całym robotem w tę stronę. Testy pokazały, że obiekt jest bardzo podatny na obecność wiatru i w takich warunkach nie jest możliwa kontrola nad nim.

Rozdział 6

Podsumowanie

Celem pracy było skonstruowanie jednowirnikowego drona oraz sterowanie nim poprzez wektorowanie ciągu. Zaprojektowano i wykonano konstrukcję mechaniczną oraz dedykowany układ sterownika lotu. Napisano programy służące do sterowania robotem, wizualizacji pochodzących z niego danych sensorycznych oraz kalibracji jego podzespołów. Rozbudowano oprogramowanie wbudowane sterownika lotu oraz dokonano procesu identyfikacji parametrów modelu na rzeczywistym obiekcie. Wykonany jednowirnikowiec zachowuje stabilność podczas lotu, więc cel pracy został osiągnięty.

Konstrukcja mechaniczna okazała się wystarczająco wytrzymała, a mechanizm separacji łopatek działał, dzięki czemu całość przetrwała wielokrotne upadki. Modularność platformy pozwoliła na łatwe i szybkie wprowadzanie niekiedy drastycznych zmian, jednak kosztem konieczności zastosowania dużej ilości śrub zwiększających masę całkowitą. Sterownik lotu spełnił postawione wymagania funkcjonalne, a jedyne zmiany w układzie wymuszone były zastosowaniem czujnika odległości innego typu niż pierwotnie zakładano. Budowa aplikacji stacji pilota umożliwia łatwą modyfikację interfejsu, co w toku prac nad projektem okazało się szczególnie przydatne. Konieczne okazało się rozbudowanie oprogramowania wbudowanego w drona o obsługę miernika mocy i czujnika ultradźwiękowego, a także implementację członu całkującego regulatora. Zmodyfikowano także części algorytmu realizujące sterowanie w trybie ręcznym oraz ograniczające narastanie i opadanie sygnału sterującego silnikiem. Zastosowanie sieci WiFi w roli medium komunikacyjnego nie sprawiło problemów, a zasięg na otwartej przestrzeni okazał się wystarczający do przeprowadzenia testów platformy. Kluczem do osiągnięcia stabilnego lotu okazały się skalibrowane napędy łopatek.

Możliwym kierunkiem kontynuacji pracy byłoby realizowanie zadania śledzenia trajektorii. Wymagałoby to estymowania pełnego położenia robota w przestrzeni oraz doposażenia konstrukcji w dodatkowy czujnik, taki jak moduł GPS, optyczny czujnik przemieszczenia lub miernik odległości od wielu stacji będących w stałych, znanych lokalizacjach. Koniecznym także byłoby zwiększenie stosunku siły ciągu do masy całkowitej. Efektywniejsze ukierunkowanie strumienia powietrza na łopatki umożliwiłoby redukcję ich liczby i rozmiarów.

Literatura

- [Air21] Airforce Technology. V-BAT 128 Unmanned Aircraft System (UAS), US. <https://www.airforce-technology.com/projects/v-bat-128/>, 2021.
- [Ama] Amazon Web Services. FreeRTOS. <https://www.freertos.org/index.html>.
- [Arm12] Army Technology. Honeywell T-Hawk Micro Air Vehicle (MAV). <https://www.army-technology.com/projects/honeywell-thawk-mav-us-army/>, 2012.
- [Bos] Bosh. Nota katalogowa układu BMP280. <https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bmp280-ds001.pdf>.
- [Bre06] C. E. Brennen. Internet book of fluid dynamics. <http://brennen.caltech.edu/fluidbook/externalflows/lift/flatplateairfoil.pdf>, Caltech, 2006.
- [CB99] S. Cheshire, M. Baker, Consistent overhead byte stuffing. *IEEE/ACM Transactions on Networking*, 7(2):159–172, 1999.
- [CFAN16] O. C. Carholt, E. Fresk, G. Andrikopoulos, G. Nikolakopoulos, Design, modelling and control of a single rotor uav. W: *2016 24th Mediterranean Conference on Control and Automation (MED)*, strony 840–845, 2016.
- [Cha] J. P. Charras. KiCad. <https://www.kicad.org/>.
- [Dam] R. Damon. FreeRTOScpp. <https://github.com/richard-damon/FreeRTOScpp>.
- [Den] A. Dennevi. Board2Pdf. <https://gitlab.com/dennevi/Board2Pdf>.
- [DGSAB⁺22] V. H. Dominguez, O. Garcia-Salazar, L. Amezquita-Brooks, L. A. Reyes-Osorio, C. Santana-Delgado, E. G. Rojo-Rodriguez, Micro coaxial drone: Flight dynamics, simulation and ground testing. *Aerospace*, 9(5), 2022.
- [DM23] J. Delicat, E. Możdżeń, *Wizualizacja danych z drona*. Sprawozdanie, https://github.com/Eryk-Mozdzen/uav-tvc-goose/blob/master/docs/reports/WDS_Goose.pdf, Politechnika Wroclawska, 2023.
- [Esp] Espressif. Nota katalogowa układu ESP8266. https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf.

- [F.W] F.W. Bell. Nota katalogowa układu CS49E. <http://rutronic.eu/opisy/inne/CS49X.pdf>.
- [Gro] Qt Group. Qt. <https://www.qt.io/>.
- [Hon] Honeywell. Nota katalogowa układu HMC5883L. <https://www.farnell.com/datasheets/1683374.pdf>.
- [Inv] InvenSense. Nota katalogowa układu MPU6050. <https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>.
- [Jac21] E. B. Jacobsen, *Modelling and Control of Thrust Vectoring Mono-copter*. Praca magisterska, https://projekter.aau.dk/projekter/files/421577367/Master_Thesis_Emil_Jacobsen_v5.pdf, Aalborg University, 2021.
- [JG] B. Jacob, G. Guennebaud. Eigen. https://eigen.tuxfamily.org/index.php?title=Main_Page.
- [Jud] T. Judd. Fit data points to a 3D Ellipsoid. http://www.juddzone.com/ALGORITHMS/least_squares_3D_ellipsoid.html.
- [Mata] MathWorks. Matlab. <https://www.mathworks.com/products/matlab.html>.
- [Matb] MathWorks. Simulink 3D Animation. <https://www.mathworks.com/products/3d-animation.html>.
- [MC14] H. G. Malkapure, M. Chidambaram, Comparison of two methods of incorporating an integral action in linear quadratic regulator. *IFAC Proceedings Volumes*, 47(1):55–61, 2014. 3rd International Conference on Advances in Control and Optimization of Dynamical Systems (2014).
- [Mon] MonolithicPower. Nota katalogowa układu MP1584. https://www.monolithicpower.com/en/documentview/productdocument/index/version/2/document_type/Datasheet/lang/en/sku/MP1584EN-LF-Z/document_id/204/.
- [Moż23a] E. Możdżeń, *Sterownik lotu drona sterowanego wektorem ciągu „Goose” – I kamień milowy*. Sprawozdanie, https://github.com/Eryk-Mozdzen/uav-tvc-goose/blob/master/docs/reports/SteRoP_3_TVCG_EM_2.pdf, Politechnika Wrocławska, 2023.
- [Moż23b] E. Możdżeń, *Sterownik lotu drona sterowanego wektorem ciągu „Goose” – II i III kamień milowy*. Sprawozdanie, https://github.com/Eryk-Mozdzen/uav-tvc-goose/blob/master/docs/reports/SteRoP_3_TVCG_EM_3.pdf, Politechnika Wrocławska, 2023.
- [Moż23c] E. Możdżeń, *Sterownik lotu drona sterowanego wektorem ciągu „Goose” – założenia projektowe*. Sprawozdanie, https://github.com/Eryk-Mozdzen/uav-tvc-goose/blob/master/docs/reports/SteRoP_3_TVCG_EM_1.pdf, Politechnika Wrocławska, 2023.

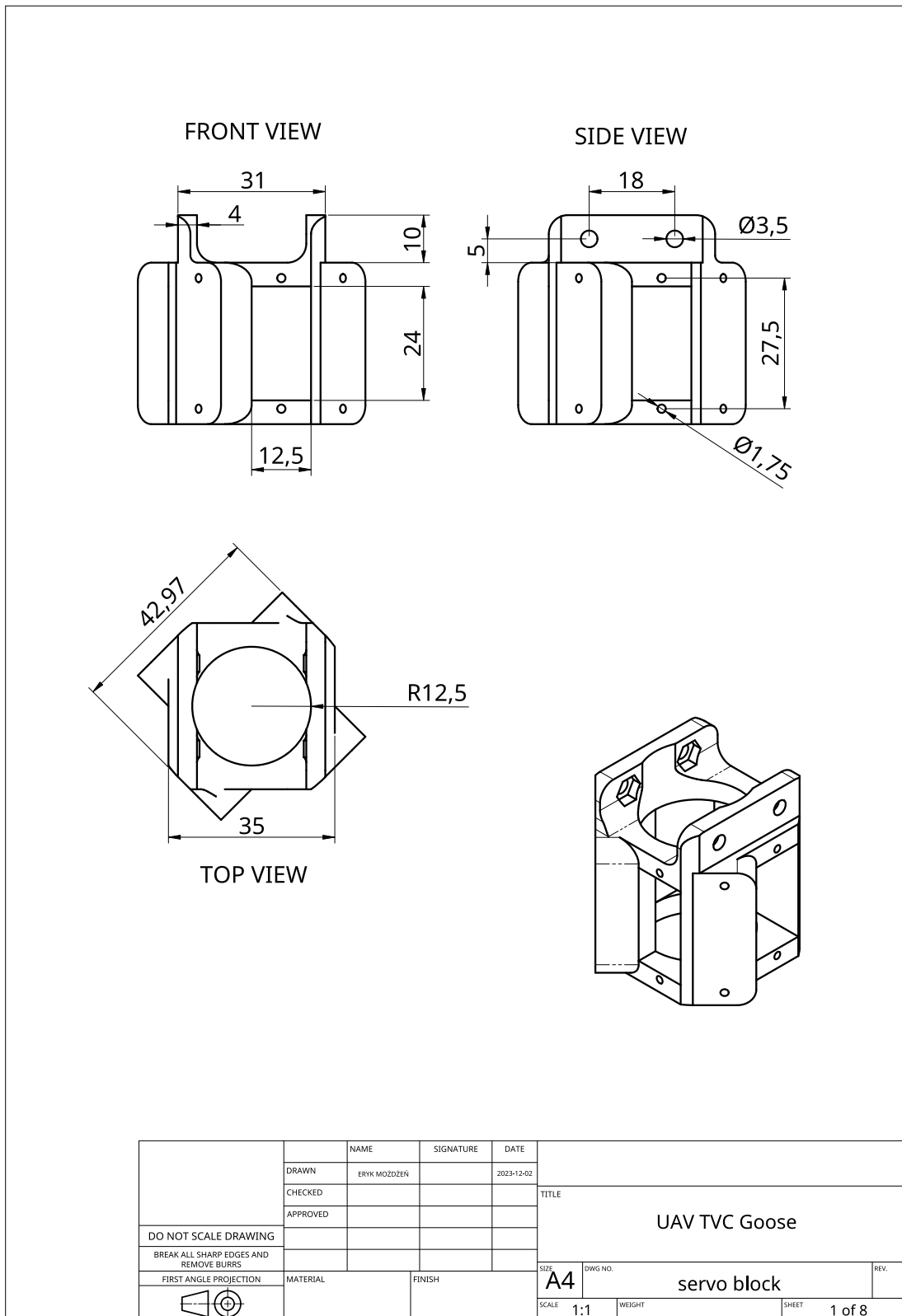
- [Moż23d] E. Możdzeń, *Symulacja bezzałogowego statku powietrznego w środowisku Simulink – sprawozdanie z finalnego projektu*. Sprawozdanie, https://github.com/Eryk-Mozdzen/uav-tvc-goose/blob/master/docs/reports/RM_etap3_finalny_projekt.pdf, Politechnika Wroclawska, 2023.
- [Moż23e] E. Możdzeń, *Symulacja bezzałogowego statku powietrznego w środowisku Simulink – sprawozdanie z przeprowadzonych prac*. Sprawozdanie, https://github.com/Eryk-Mozdzen/uav-tvc-goose/blob/master/docs/reports/RM_etap2_wstepne_rezultaty.pdf, Politechnika Wroclawska, 2023.
- [Moż23f] E. Możdzeń, *Symulacja bezzałogowego statku powietrznego w środowisku Simulink – założenia projektowe*. Sprawozdanie, https://github.com/Eryk-Mozdzen/uav-tvc-goose/blob/master/docs/reports/RM_etap1_zalozenia_projektowe.pdf, Politechnika Wroclawska, 2023.
- [ons] onsemi. Nota katalogowa układu LM393. <https://www.onsemi.com/download/data-sheet/pdf/lm393-d.pdf>.
- [Pal] V. Palikša. Blackmagic esp-idf. <https://github.com/walmis/blackmagic-espidf>.
- [PTC] PTC. Onshape. <https://www.onshape.com/en/>.
- [PWN] Encyklopedia PWN. brzytwa Ockhama. <https://encyklopedia.pwn.pl/haslo/brzytwa-Ockhama;3881454.html>.
- [R607] R. Róziecki, *Bifurkacyjna analiza dynamiki lotu samolotu supermanewrowego z wektorowaniem ciągu*. Praca doktorska, <https://www.dbc.wroc.pl/dlibra/doccontent?id=1773>, Politechnika Wroclawska, 2007.
- [STMa] STMicroelectronics. Nota aplikacyjna do układów z serii STM32F4xxxx. https://www.st.com/resource/en/application_note/an4488-getting-started-with-stm32f4xxxx-mcu-hardware-development-stmicroelectronics.pdf.
- [STMb] STMicroelectronics. Nota katalogowa układu STM32F411RE. <https://www.st.com/en/microcontrollers-microprocessors/stm32f411re.html>.
- [Texa] Texas Instruments. Nota katalogowa układu INA226. <https://www.ti.com/lit/ds/symlink/ina226.pdf>.
- [Texb] Texas Instruments. Nota katalogowa układu LM1117. <https://www.ti.com/lit/ds/symlink/lm1117.pdf>.
- [The] ThePoorEngineer. Extended Kalman filter implementation. <https://thepoorengineer.com/en/ekf-impl>.
- [TM18] K. Tchoń, R. Muszyński, *Robotyka*. Notatki do wykładów, https://kcir.pwr.edu.pl/~mucha/Skrypty/KTRM_Robotyka.pdf, Politechnika Wroclawska, 2018.

- [Ult] UltiMaker. Cura. <https://ultimaker.com/software/ultimaker-cura/>.
- [Vec] VectorNav Technologies. Magnetometer Hard & Soft Iron Calibration. <https://www.vectornav.com/resources/inertial-navigation-primer/specifications--and--error-budgets/specs-hsicalibration>.
- [Wika] Wikipedia. Algebraic Riccati equation. https://en.wikipedia.org/wiki/Algebraic_Riccati_equation.
- [Wikb] Wikipedia. ANSI escape code. https://en.wikipedia.org/wiki/ANSI_escape_code.
- [Wikc] Wikipedia. Barometric formula. https://en.wikipedia.org/wiki/Barometric_formula.
- [Wikd] Wikipedia. Bracia Montgolfier. https://pl.wikipedia.org/wiki/Bracia_Montgolfier.
- [Wike] Wikipedia. Extended Kalman filter. https://en.wikipedia.org/wiki/Extended_Kalman_filter.
- [Wikf] Wikipedia. Linear-quadratic regulator. https://en.wikipedia.org/wiki/Linear%E2%80%93quadratic_regulator.
- [Wikg] Wikipedia. Osadzanie topionego materiału. https://pl.wikipedia.org/wiki/Osadzanie_topionego_materia%C5%82u.

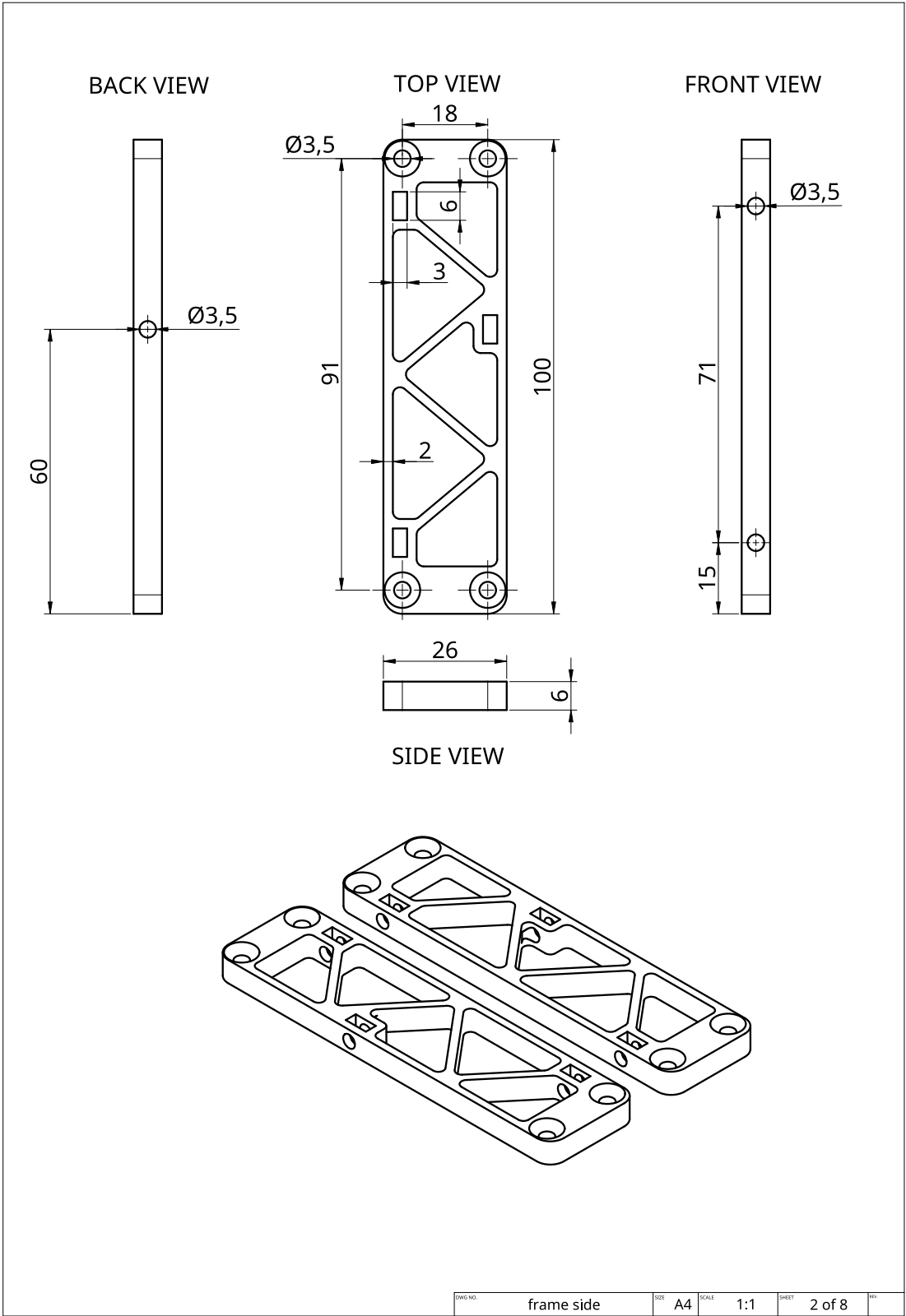
Dodatek A

Rysunki techniczne

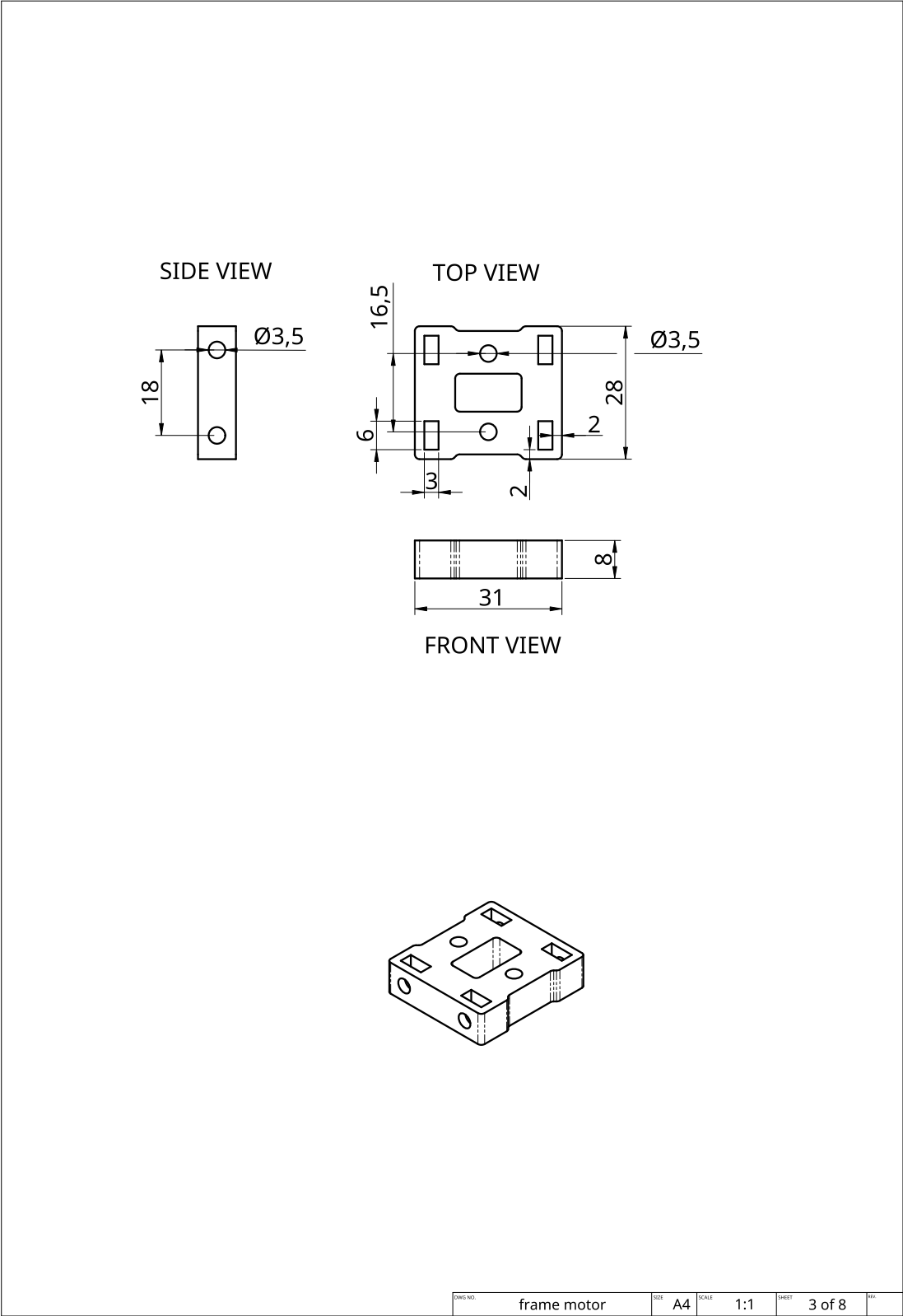
Dodatek zawiera rysunki techniczne elementów zaprojektowanych na potrzeby projektu. Projekty zostały wykonane za pomocą platformy Onshape [[PTC](#)].



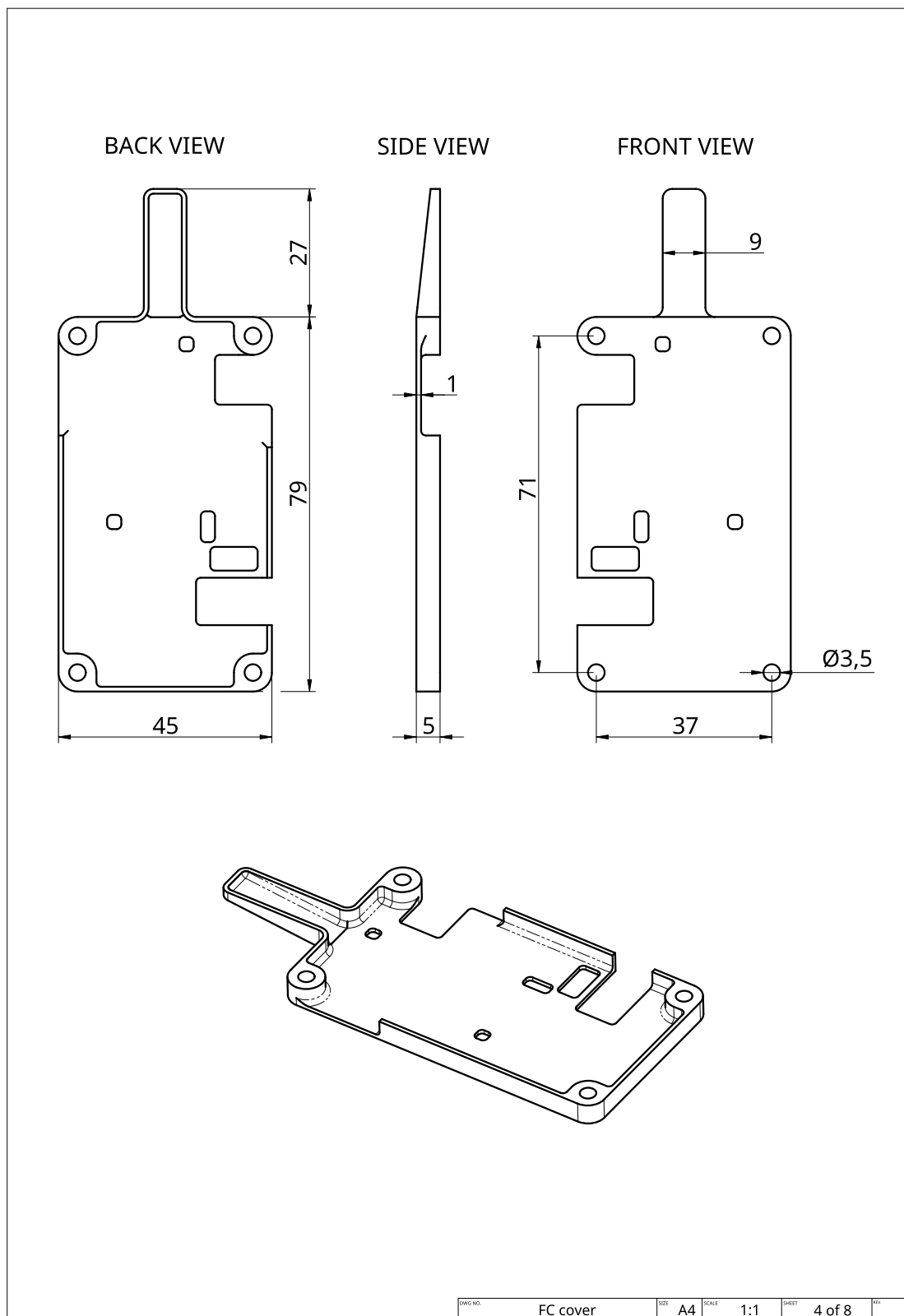
Rysunek A.1 Blok serwomechanizmów



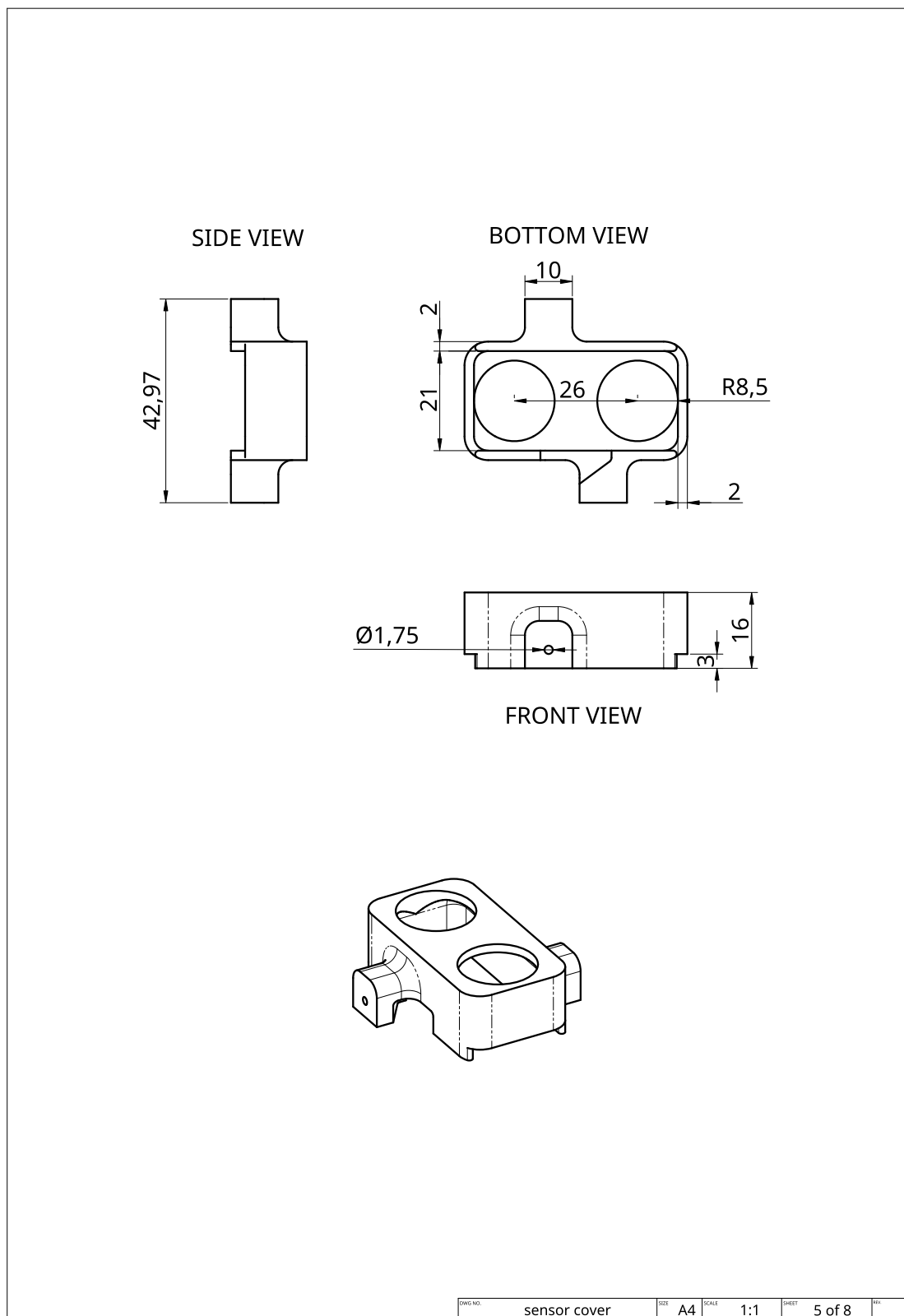
Rysunek A.2 Rama boczna



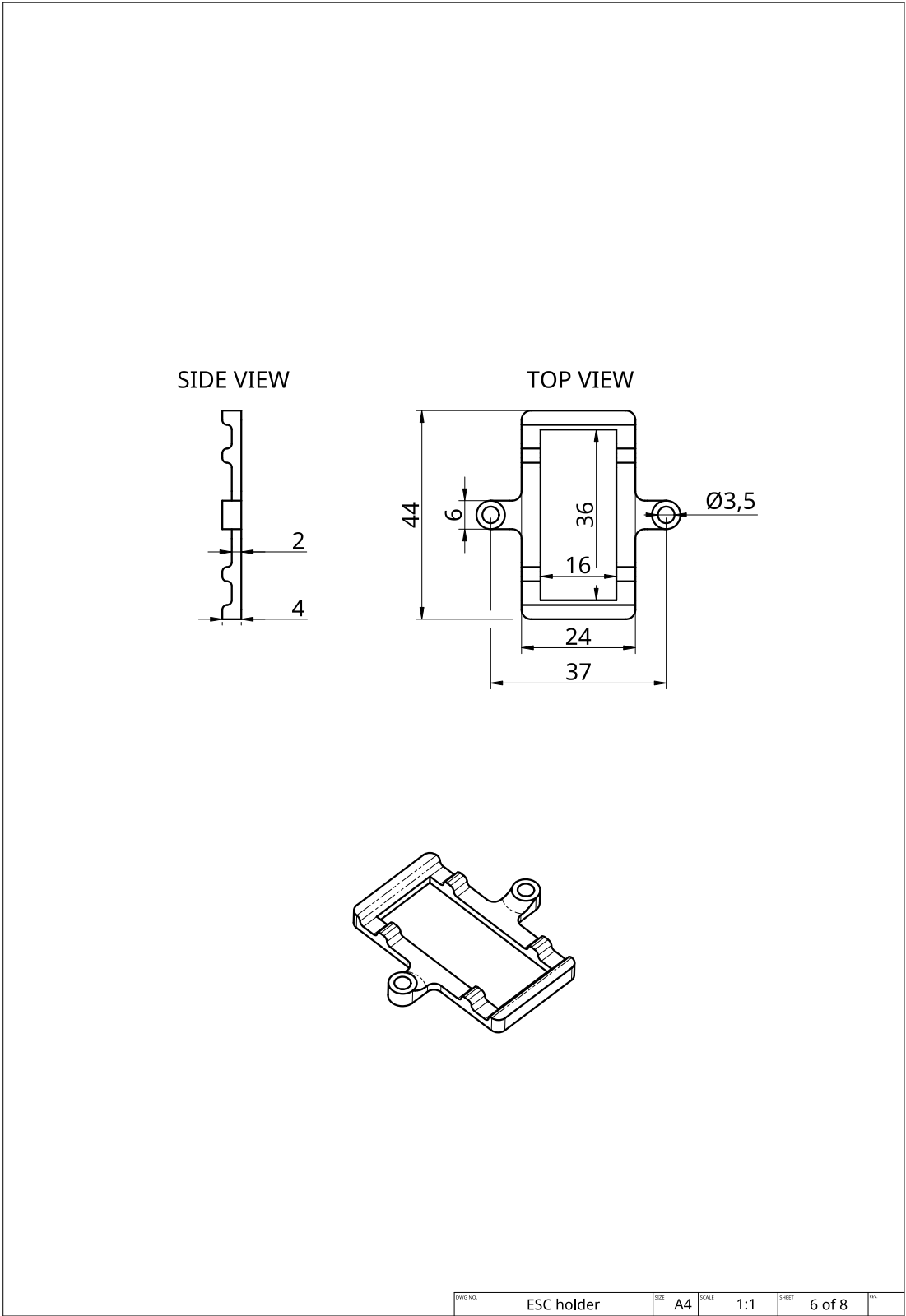
Rysunek A.3 Rama silnika



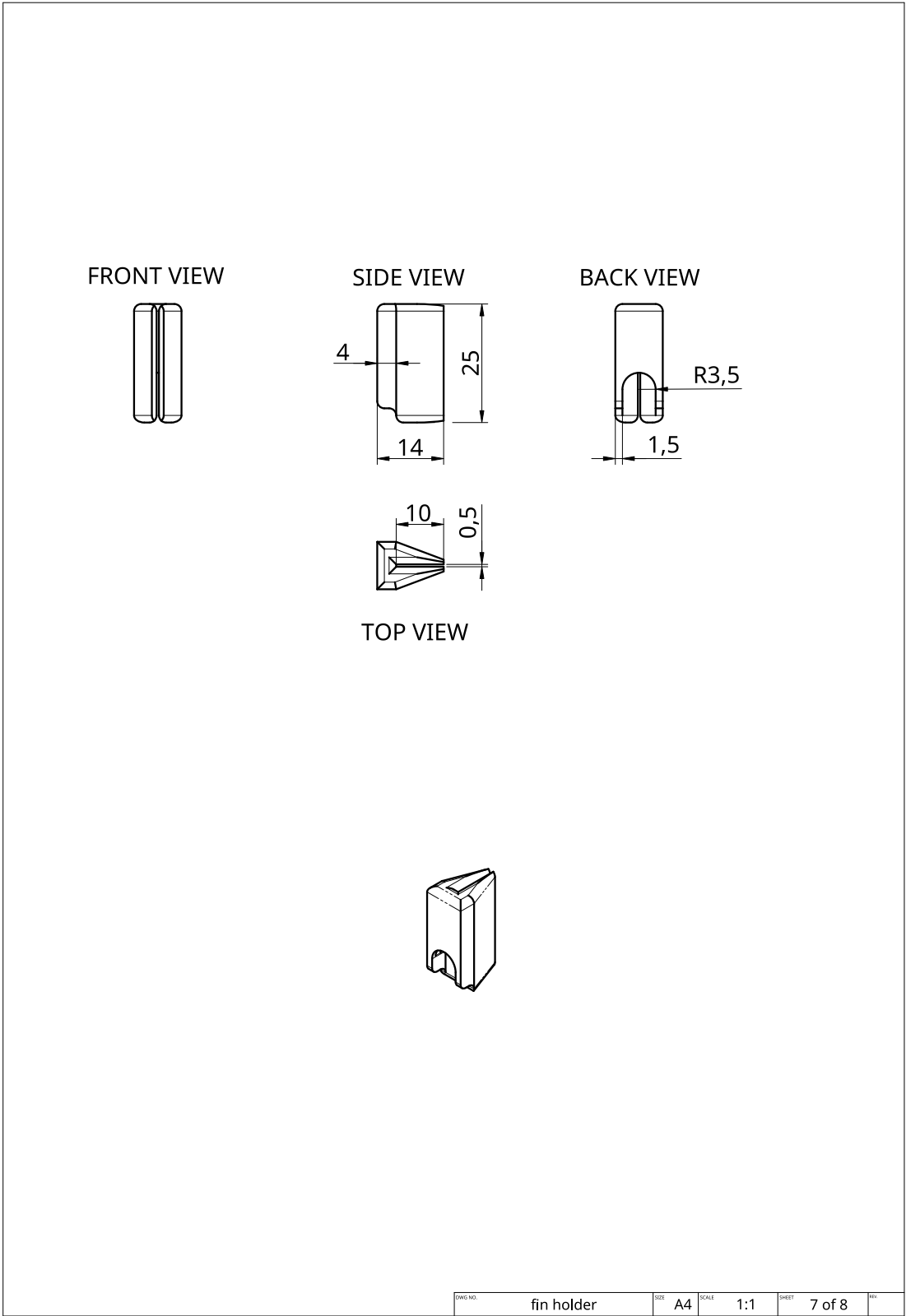
Rysunek A.4 Osłona sterownika lotu



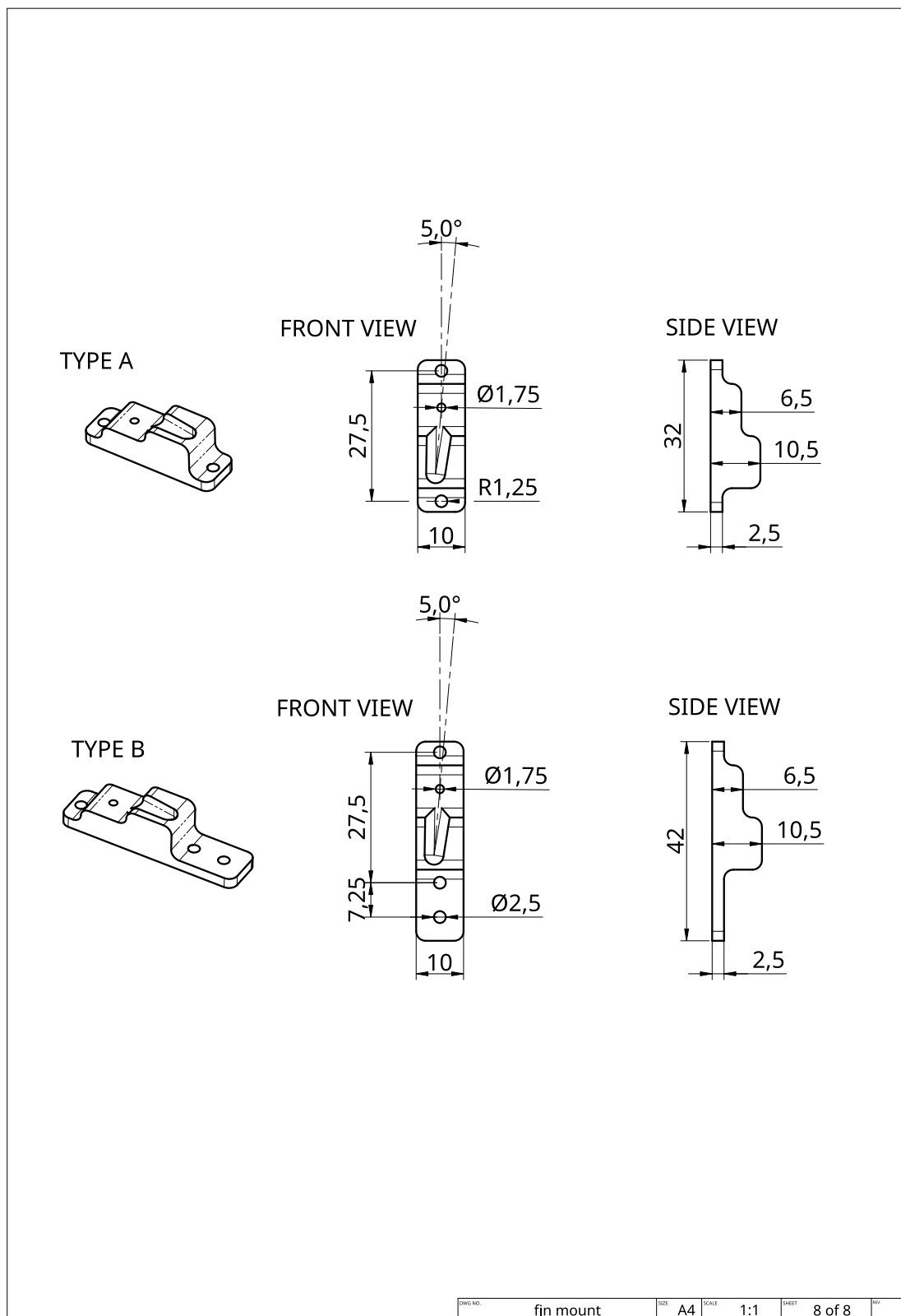
Rysunek A.5 Osłona czujnika odległości



Rysunek A.6 Uchwyt ESC



Rysunek A.7 Uchwyt łopatki

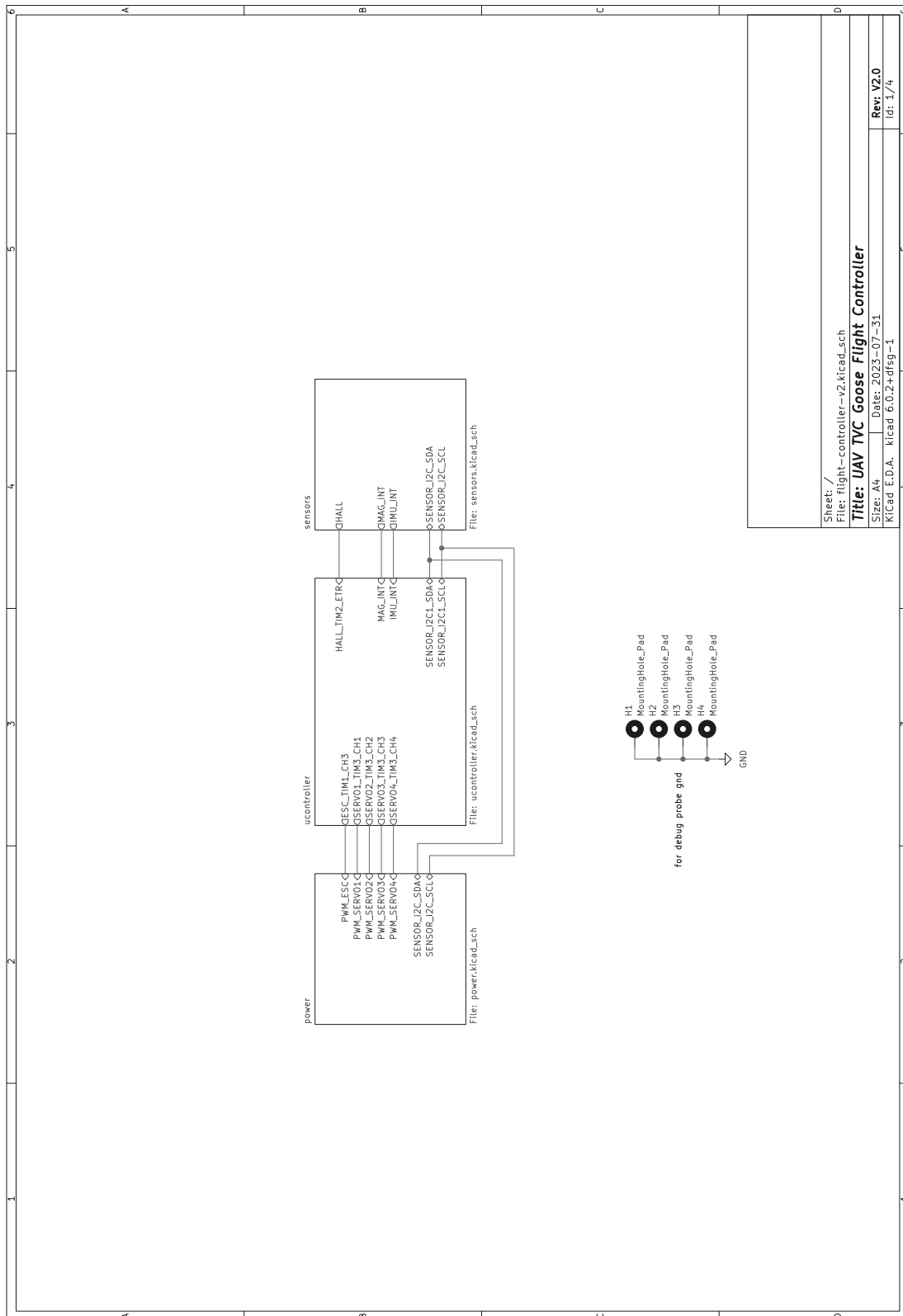


Rysunek A.8 Mocowanie łopatki

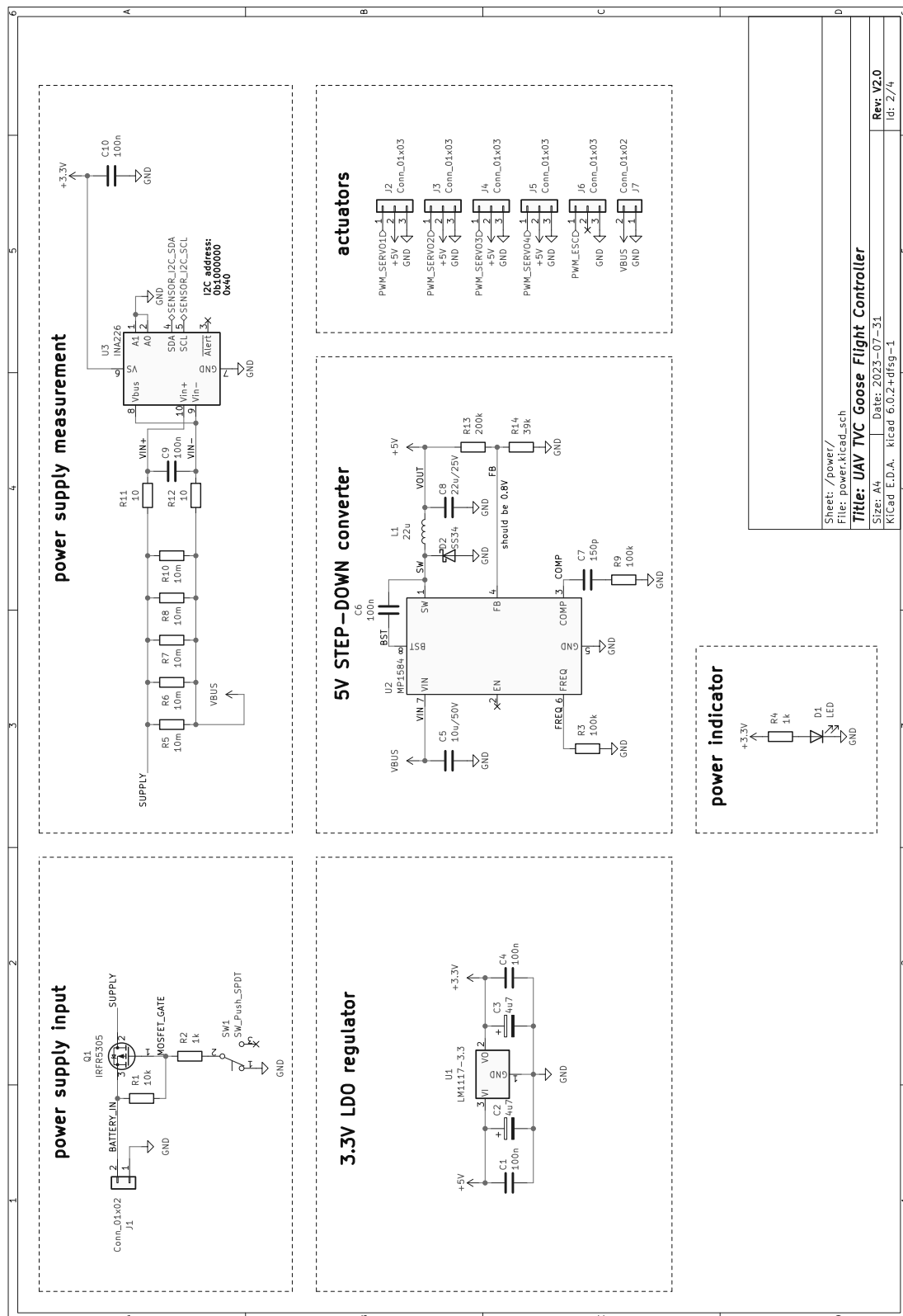
Dodatek B

Schematy sterownika lotu

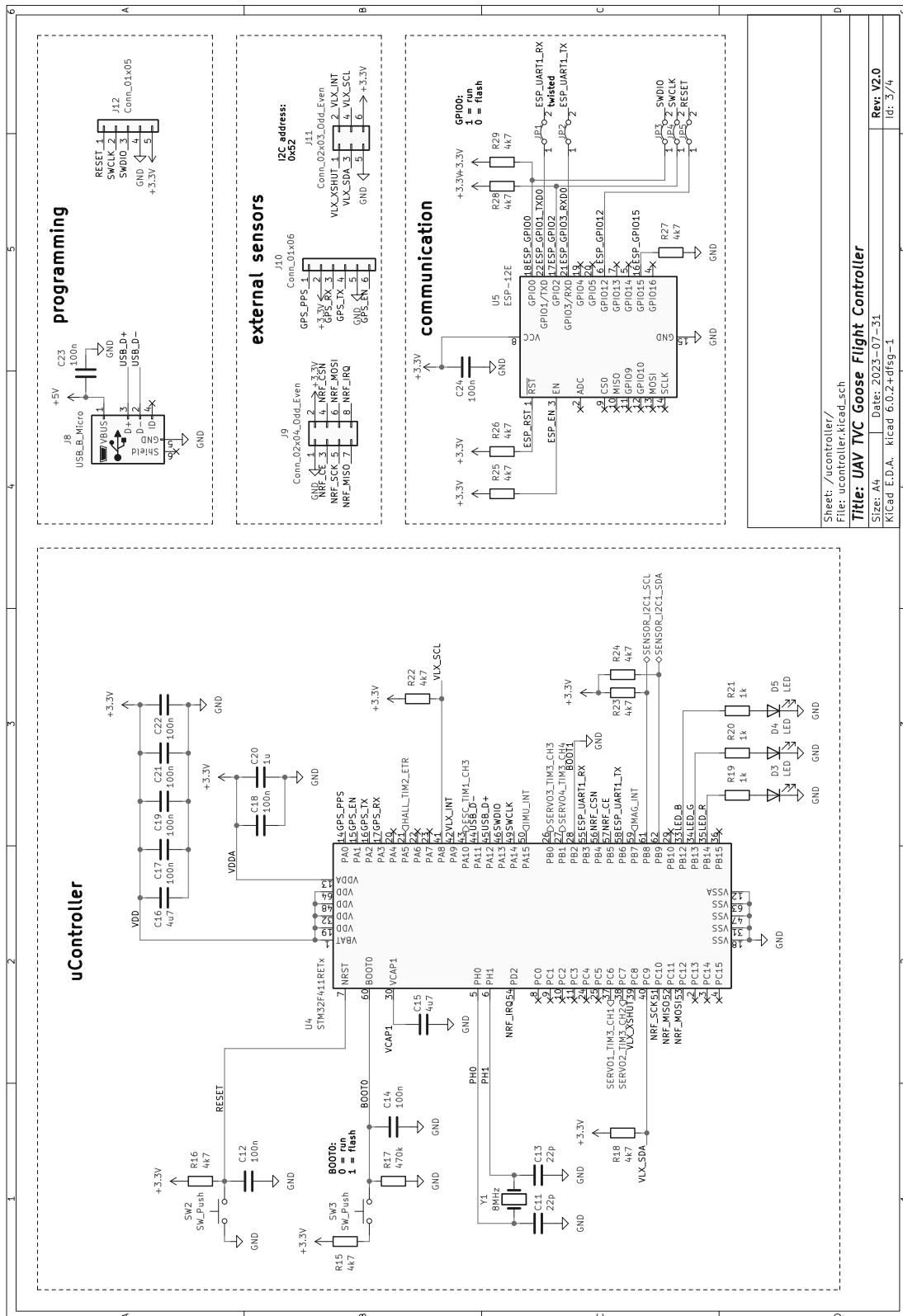
Dodatek zawiera schematy elektroniczne sterownika lotu. Projekt został wykonany za pomocą programu KiCad [[Cha](#)]. Do wygenerowania plików została użyta wtyczka Board2Pdf [[Den](#)].



Rysunek B.1 Główny schemat hierarchiczny

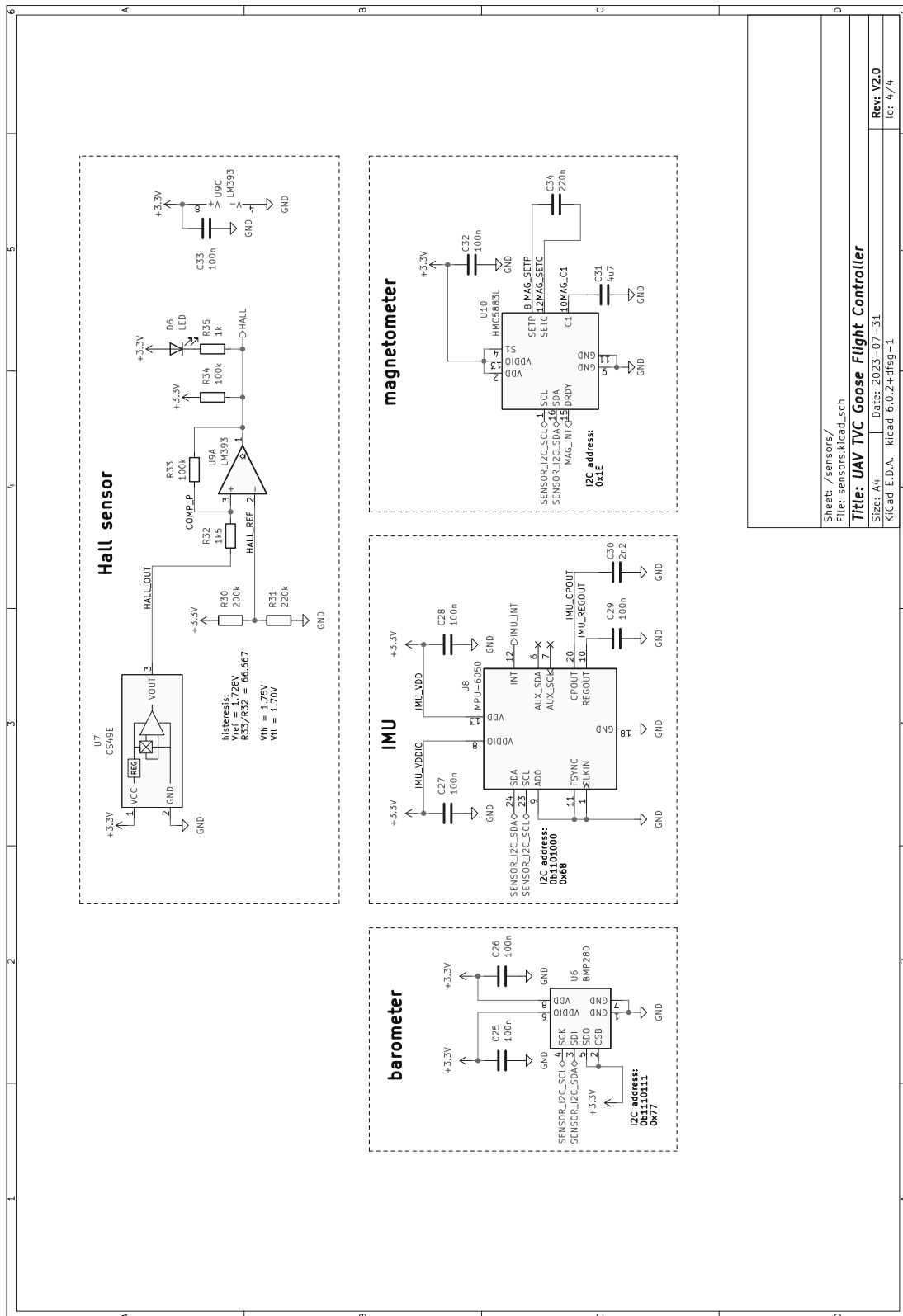


Rysunek B.2 Sekcja zasilania

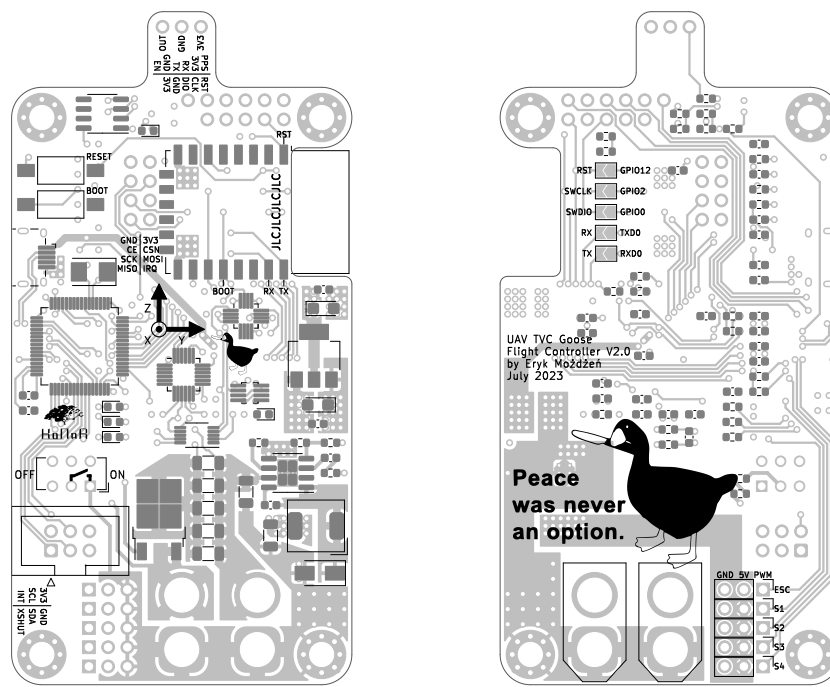


Sheet: /ucontroller/
 File: ucontroller.kicad_sch
Title: UAV TVC Goose Flight Controller
 Size: A4 Date: 2023-07-31
 Kicad E.D.A. kicad 6.0.2+dfsg-1
 Rev: V2.0
 Id: 3/4

Rysunek B.3 Sekcja komunikacji i sterowania



Rysunek B.4 Sekcja układów sensorycznych



(a) Widok od przodu

(b) Widok od tyłu

Rysunek B.5 Płytki drukowana

Dodatek C

Pomiary identyfikacji

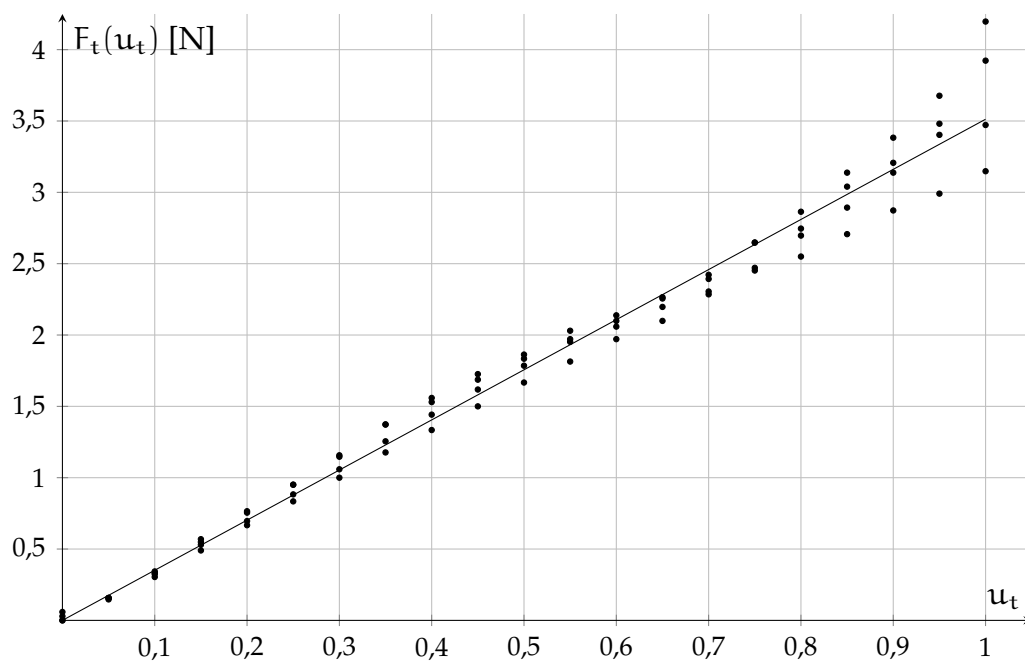
W wyniku pomiarów oraz obliczeń określono wartości parametrów modelu, które zostały zebrane w tabeli C.1. Pomiary eksperymentu identyfikującego współczynnik ciągu oraz prędkości obrotowej zostały zawarte w tabeli C.2. Wyniki pomiarów pierwszego eksperymentu oraz najlepiej dopasowane krzywe zostały przedstawione na rysunkach C.1 oraz C.2. Współczynnik momentu obrotowego został wyznaczony na podstawie pomiarów zawartych w tabeli C.3. Rysunek C.3 przedstawia wyniki drugiego eksperymentu oraz także najlepiej dopasowaną prostą. Każda seria pomiarów realizowanych w ramach identyfikacji parametrów modelu była rozpoczynana przy w pełni naładowanej baterii.

Parametr	Wartość	Jednostka	Źródło
m	0,332	kg	waga
l	0,0500	m	CAD
r	0,0665	m	CAD
J_{xx}	0,0009887	kgm ²	waga + CAD
J_{yy}	0,0009817	kgm ²	waga + CAD
J_{zz}	0,0002177	kgm ²	waga + CAD
J_r	0,00001366	kgm ²	waga + CAD
α_s	-0,08727 (-5)	rad (deg)	obliczenia
C_l	0,4203	N/N	obliczenia
K_t	3,512	N	eksperyment
K_ω	0,000003133	Ns ²	eksperyment
K_τ	0,04629	Nm	eksperyment

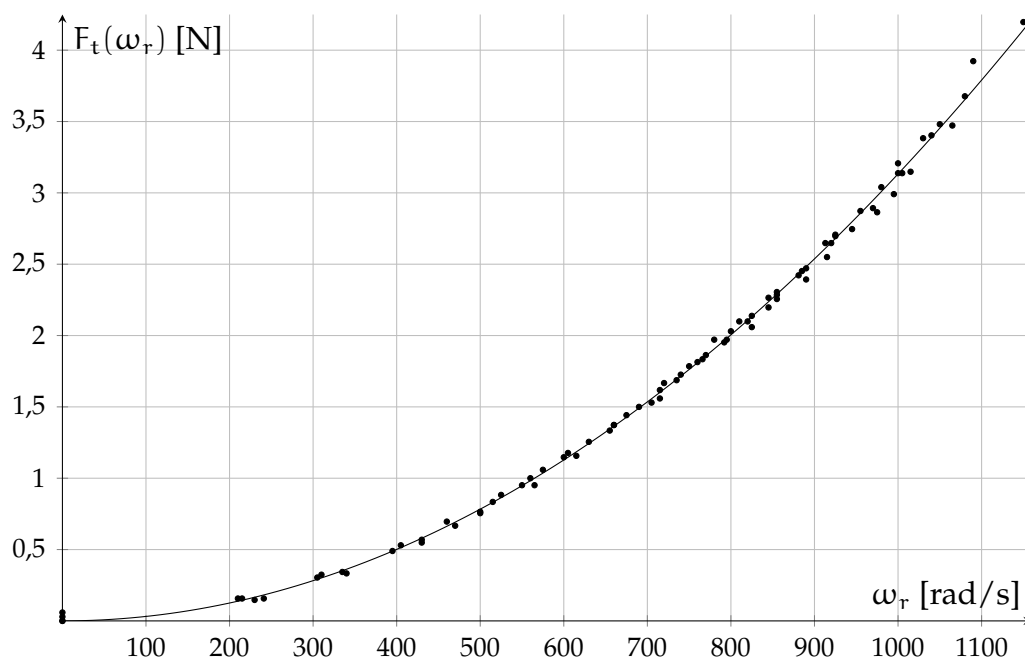
Tabela C.1 Przyjęte wartości parametrów modelu

Sterowanie [%]	Nacisk [g]	Prędkość [rad/s]	Sterowanie [%]	Nacisk [g]	Prędkość [rad/s]
0	0	0	0	3	0
5	15	230	5	16	215
10	34	340	10	33	310
15	56	430	15	54	405
20	77	500	20	71	460
25	97	550	25	90	525
30	117	600	30	108	575
35	140	660	35	128	630
40	159	715	40	147	675
45	176	740	45	165	715
50	190	770	50	182	750
55	207	800	55	201	780
60	218	825	60	210	825
65	230	855	65	231	845
70	244	890	70	247	881
75	270	920	75	270	913
80	280	945	80	292	975
85	310	980	85	320	1005
90	327	1000	90	345	1030
95	347	1040	95	375	1080
100	354	1065	100	428	1150
5	16	210	5	16	241
10	31	305	10	35	335
15	50	395	15	58	430
20	68	470	20	78	500
25	85	515	25	97	565
30	102	560	30	118	615
35	120	605	35	140	660
40	136	655	40	156	705
45	153	690	45	172	735
50	170	720	50	187	766
55	185	760	55	199	792
60	201	795	60	214	820
65	214	810	65	224	845
70	233	855	70	235	855
75	252	890	75	250	885
80	275	925	80	260	915
85	295	970	85	276	925
90	320	1000	90	293	955
95	355	1050	95	305	995
100	400	1090	100	321	1015

Tabela C.2 Pomiary identyfikacji współczynników ciągu i prędkości wirnika



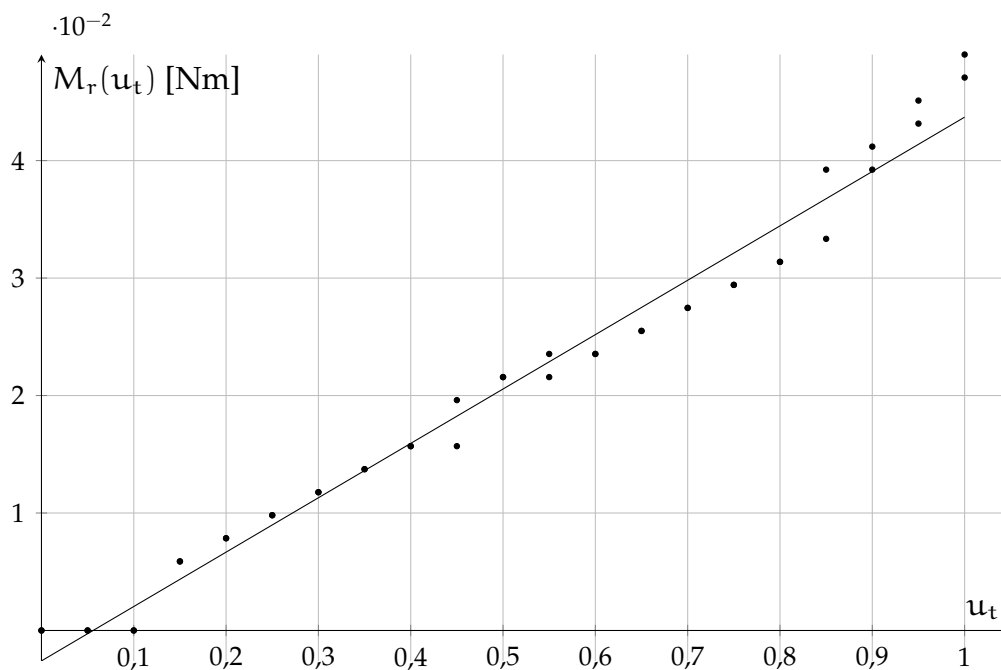
Rysunek C.1 Ciąg w zależności od sterowania



Rysunek C.2 Ciąg w zależności od prędkości obrotowej wirnika

Sterowanie [%]	Nacisk [g]	Sterowanie [%]	Nacisk [g]
0	0	0	0
5	0	5	0
10	0	10	0
15	3	15	3
20	4	20	4
25	5	25	5
30	6	30	6
35	7	35	7
40	8	40	8
45	8	45	10
50	11	50	11
55	11	55	12
60	12	60	12
65	13	65	13
70	14	70	14
75	15	75	15
80	16	80	16
85	17	85	20
90	20	90	21
95	22	95	23
100	24	100	25

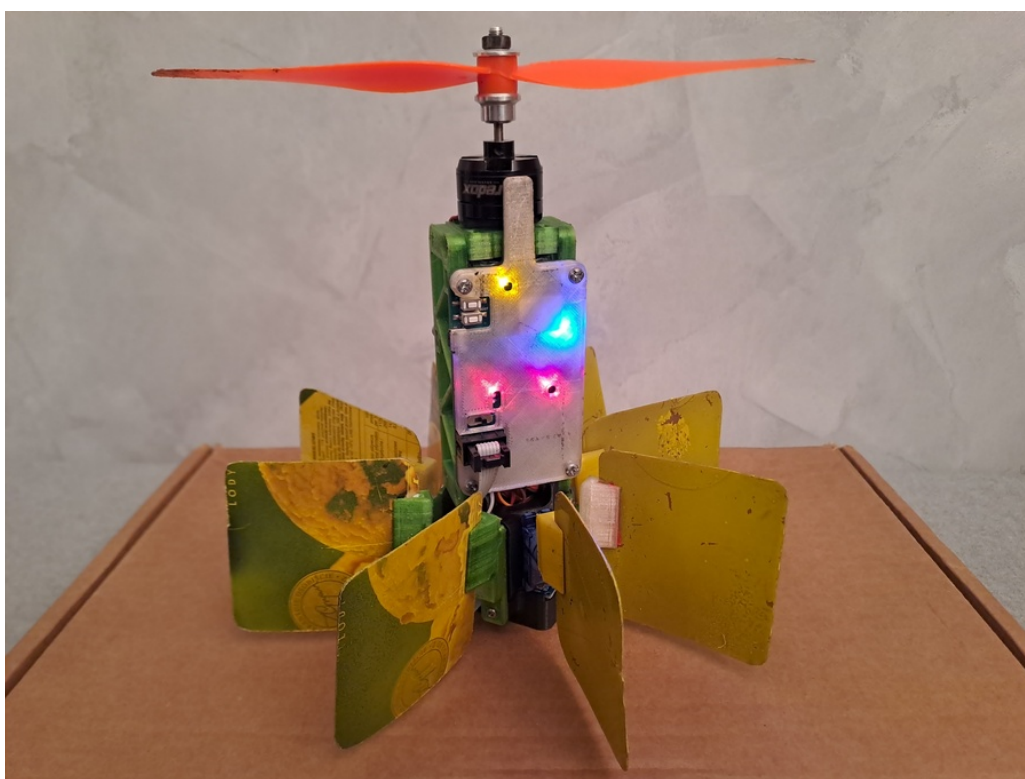
Tabela C.3 Pomiary do identyfikacji współczynnika momentu obrotowego



Rysunek C.3 Moment obrotowy generowany przez wirnik od sterowania

Dodatek D

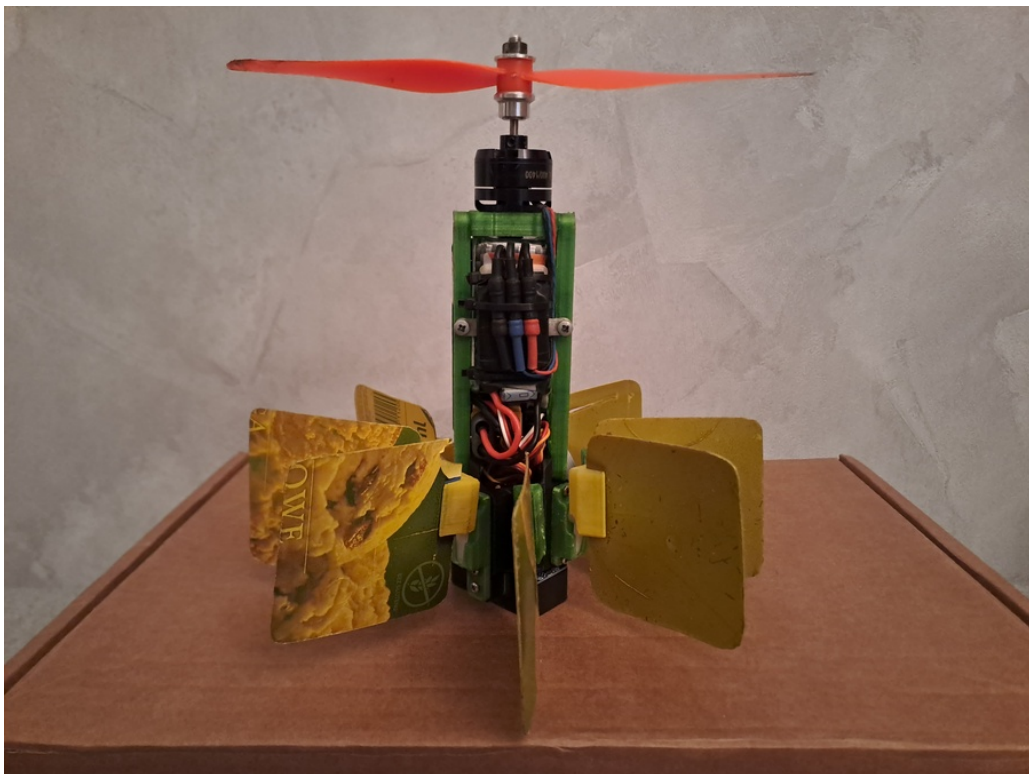
Galeria zdjęć



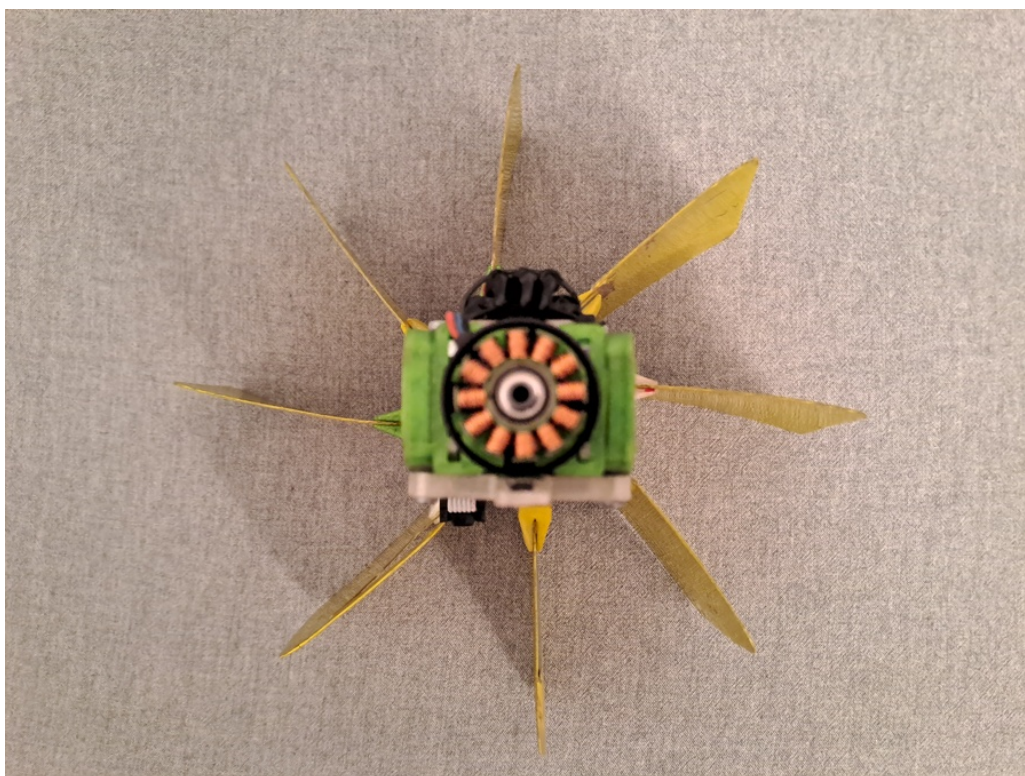
Rysunek D.1 Jednowirnikowiec gotowy do lotu



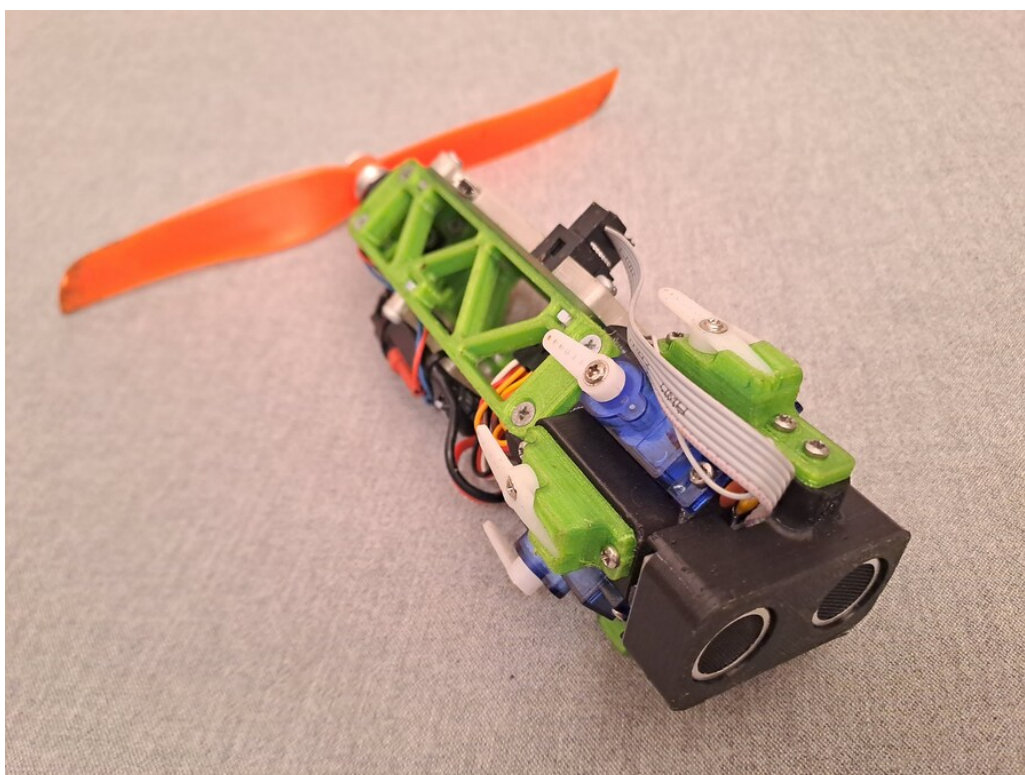
Rysunek D.2 Widok z boku



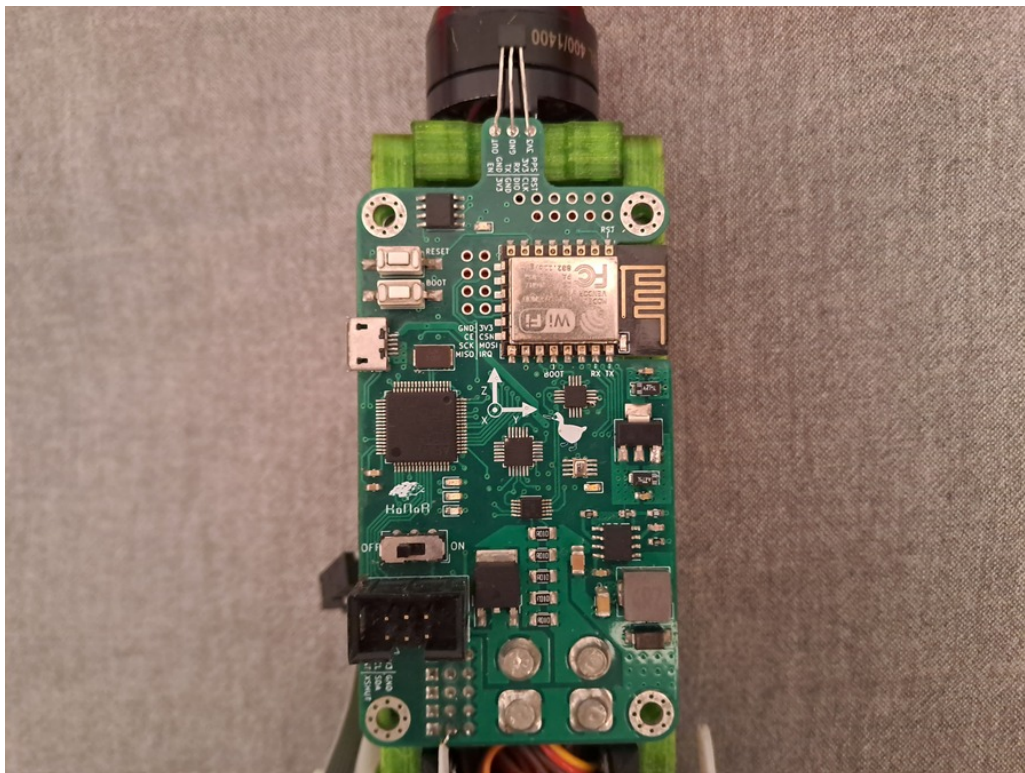
Rysunek D.3 Widok z tyłu



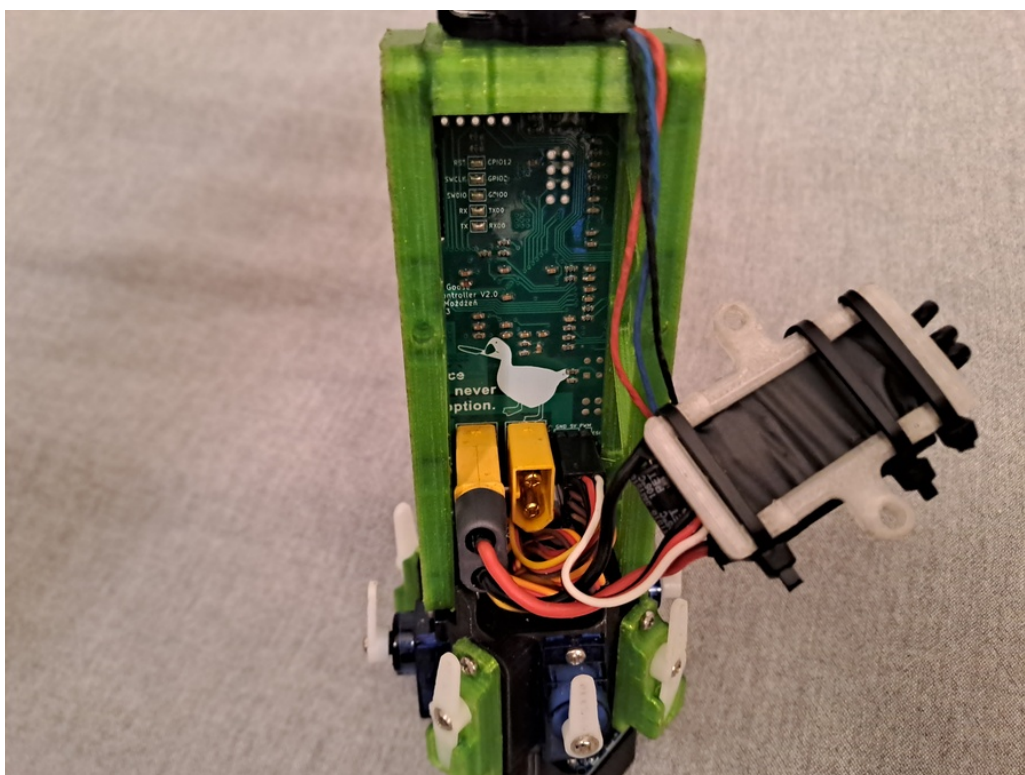
Rysunek D.4 Widok z góry (bez wirnika)



Rysunek D.5 Widok z dołu (bez łopatek)



Rysunek D.6 Widok sterownika lotu z przodu (bez obudowy)



Rysunek D.7 Widok sterownika lotu z tyłu (bez baterii)