



Sapientia Erdélyi Magyar Tudományegyetem

Automatika és alkalmazott informatika mérnök szak

Villamosmérnöki Tanszék

Bicopter modellezése, kivitelezése és irányítása

KÉSZÍTETTE

Marton Tamás-Loránd

IV. évfolyam

KONZULENS

Dr. Kutasi Dénes Nimród

Egyetemi docens

2023

Tartalomjegyzék

1. Bevezető.....	4
2. Célkitűzések.....	4
3. Elméleti megalapozás és bibliográfiai tanulmány	4
3.1. Kétmotoros légi robot (bicopter) síkbeli matematikai modellje	7
3.2. Kétmotoros légi robot (bicopter) térbeli matematikai modellje.....	11
3.3. Bevezetés a Kálmán-szűrő elméletébe	15
4. Modell szimuláció, validáció	16
4.2. Kétmotoros légi robot (bicopter) térbeli matematikai modelljének validációja és szimulációja.....	23
5. Pályakövető szabályozó tervezése a síkbeli modell alapján	26
6. Inerciális mérőeszköz jeleinek feldolgozása Kálmán-szűrővel	29
6.1. MPU-9150.....	29
6.1.1. Szenzor adatok kiolvasása	29
6.1.2. A szenzor kalibrálása	30
6.2. Az adatok előszűrőése alpha szűrővel.....	31
6.3. Kálmán-szűrő tervezése	31
6.4. Szögek kiszámítása	32
7. Gyakorlati kivitelezés	33
7.1. Mechanikai elemek	33
7.2. A Bicopter kapcsolási rajz.....	35
7.3. ESC (Electronic Speed Control) vezérlése PWM-en keresztül	36

8.	Tesztelés.....	37
8.1.	Kálmán-szűrő tesztelése.....	37
8.2.	Szög mérése integrálással.....	40
9.	Következtetések.....	43
9.1.	Megvalósítások.....	43
9.2.	További fejlesztési irányok	44
10.	Irodalomjegyzék	45

1. Bevezető

A bicopter egy olyan légi jármű, amely két forgószárnyat használ az emelkedés és a manőverezés biztosítására. A forgószárnyak elhelyezkedése lehet párhuzamos vagy V-alakú, valamint mindkét rotor függetlenül irányítható. A bicopter könnyű, kompakt és kiváló manőverező képességgel rendelkezik, ami lehetővé teszi számára, hogy szűk helyeken is mozogjon és akár álló helyzetben is repüljön. Ez a géptípus egyre népszerűbb az áruszállítás, kutatás-fejlesztés és a katonai alkalmazások terén is. A bicopter kialakítása és működése azonban nagyon bonyolult lehet, és számos technikai kihívással jár. Például a rotorok kezelése, a súlyelosztás, a szélviszonyok hatása és a stabilizáció. Azonban a fejlesztők folyamatosan dolgoznak a bicopter továbbfejlesztésén és javításán, hogy egyre hatékonyabbá és megbízhatóbbá tegyék ezt a légi járművet.

2. Célkitűzések

Dolgozatom célja egy kétmotoros légi robot (bicopter) 2D-s és 3D-s matematikai modellezése és szimulációja, a szabályozási feladatok elméleti és gyakorlati vizsgálata, valamint a robot gyakorlati megvalósítása. A gyakorlati megvalósítás esetében meg kell oldani a már klasszikusnak mondható, az inerciális és mágneses szenzorok esetében fennálló zaj- és eltolás problémákat. Erre a feladatra a Kálmán-szűrő a megoldás. Megvalósítottam egy saját valós rendszert (bicoptert), amelyet először síkban, aztán térben is tesztelek. Gyakorlati mérésekkel támasztom alá az elméleti eredményeket.

3. Elméleti megalapozás és bibliográfiai tanulmány

A pilóta nélküli légi járművek (UAV) a közelmúltban nagyszerű eszközzé váltak az abiotikus és biotikus stressz hatások felderítésében. A kézi felderítéshez képest kevesebb időt és erőfeszítést igényelnek, és nagyobb területeket tudnak felderíteni[1]–[3]. Az UAV-k aerodinamikai jellemzők alapján különböző típusokba sorolhatók. A négy fő típus a következő: több rotoros, rögzített szárnyú, egy rotoros és hibrid függőleges fel- és leszállásos (VTOL).

Több rotoros UAV-k a legszélesebb körben elterjedtek, egyaránt használatos hétköznapi használatra és iparban. Használják fényképezésre, feltérképezésre és szabadidős sportokra is. Az UAV-n található rotorok száma alapján különböző típusba sorolhatóak: tricopter (három

rotor), quadcopter (négy rotor), hexacopter (hat rotor) és oktocopter (nyolc rotor). Általában az UAV-k által szállítható hasznos teher nagysága a rotorainak számával egyenesen arányos. Annak ellenére, hogy a több rotoros UAV-k a legolcsóbb elérhető opciót jelentik, hátrányai is vannak. A többi UAV-típushoz képest korlátozott repülési idővel rendelkeznek. Ez annak köszönhető, hogy a több rotoros UAV-nak sok energiára van szükségük ahhoz, hogy a gravitációval és a széllel szemben stabilak maradjanak a levegőben. A több rotoros UAV-k átlagos repülési ideje körülbelül 20-40 perc között mozog[4].

Egy rotoros UAV-k kialakításukban és szerkezetükben nagyon hasonlítanak a helikopterekhez. Nem szükséges sok energia a motor részéről, hiszen ezek az UAV-k felhasználják a felhajtó erőt a szerkezeti kialakításukból kifolyólag. A merevszárnyú UAV-k repülési ideje elérheti az 1-2 órát. Hátrányuk a nehezebb irányíthatóság.



ábra 1. Hibrid VTOL [5]

A hibrid VTOL típusú UAV-k egyesítik a merevszárnyú és a több rotoros UAV-k képességeit, mely lehetővé teszi a függőleges felszállást és leszállást. Ezek az UAV-k még a fejlesztés korai szakaszában vannak, és jelenleg nagyon kevés lehetőség áll rendelkezésre a piacon. Hosszú repülési idővel rendelkeznek és nagyobb hasznos terhet tudnak szállítani[6].



ábra 2. Bell Boeing V-22 Osprey[7]

Ami az UAV-kat illeti, számos tanulmányt végeztek a quadcopterek, a merevszárnyúak és más több rotoros konstrukciók esetében. Ezek konstrukcióját, dinamikáját és vezérlését a kutatások során alaposan tanulmányozták. A bicopterek esetében azonban ez nem így van. Instabil felépítésük és az irányítás elérésének bonyolultsága miatt, ezeket az UAV-kat nehéz tervezni és stabilizálni. Ezen korlátok miatt a bicopterek nem tekinthetők reprezentatívnak a jelenlegi repülőiparban összehasonlítva azokkal a quadcopterekkel, amelyek már jelenleg a piacon vannak. A bicopterek fő előnye a quadokkal vagy helikopterekkel szemben az, hogy az utóbbi modellekhez képest lényegesen nagyobb vízszintes sebességgel és hosszabb élettartammal rendelkeznek. Egy másik tulajdonság, amely a bicoptereket érdekesebbé teszi a merevszárnyú tervezésnél, az az hogy VTOL üzemmódban is üzemeltethetők. A merevszárnyú UAV-k jobb stabilitással és vízszintes sebességgel rendelkeznek, de nem képesek szűk területeken vagy korlátozott és kedvezőtlen körülmények között a felszállásra. A bicopterek viszont VTOL üzemmódban is üzemeltethetők. A helikopterek és a quadcopterek a legstabilabb VTOL-ek, amelyek jelenleg is üzemelnek, de alacsony horizontális sebességük és kis hatótávuk a nagy hatótávolságú küldetéseknél sürgetik a jobb tervezés szükségességét.



ábra 3. VTOL[8]

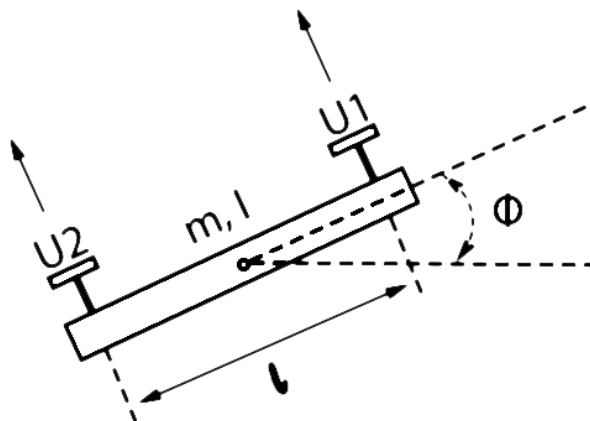
A bicoptereken belül kétfajtát különítünk el az előre-hátra aktív billenő (FAAT) és ferde aktív billenő (OAT)[9].

	Típus 1	Típus 2	Típus 3
Szárny	Igen	-	Igen
Billenő rotor	Igen	-	-
VTOL	Igen	Igen	Igen

1. táblázat A VTOL járművek osztályozása[9]

3.1. Kétmotoros légi robot (bicopter) síkbeli matematikai modellje

A 4-es ábrán látható a bicopter síkbeli rajza, írott l -el jelöltem a két motor közötti távolságot.



ábra 4. A bicopter síkbeli kinézete

Ahhoz, hogy felírhasam a rendszer matematikai modelljét a kinetikus (1) és a potenciális energia (2) fizikai összefüggéséből indultam ki.

$$T = \frac{m(\dot{x}^2 + \dot{y}^2)}{2} + \frac{I\dot{\phi}^2}{2} \quad (1)$$

$$V = mgh \quad (2)$$

A helyzeti és a potenciális energia különbsége (3) nem lesz más, mint az Euler-Lagrange-egyenlet megoldásához szükséges Lagrange-függvény.

$$L = T - V \quad (3)$$

Felírom a rendszerre ható külső erők összegét, lásd a (4) -es egyenletet.

$$Q_j = J_p^T F_p^0 + J_R^T F_R^0 \quad (4)$$

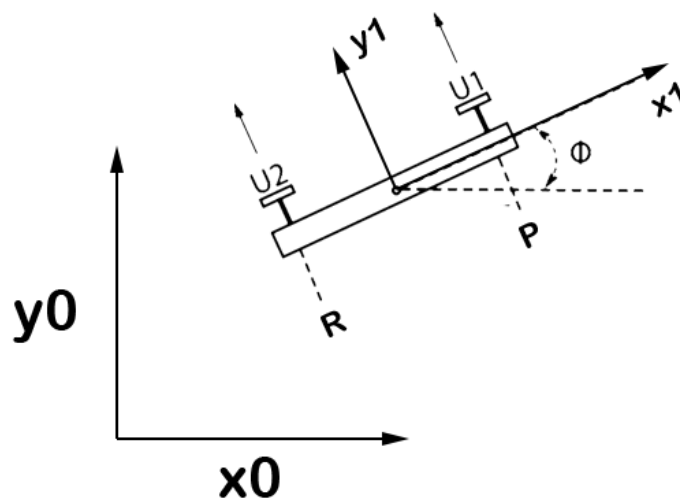
J_p^T - Jacobi-mátrix p pontban

J_R^T - Jacobi-mátrix R pontban

F_p^0 - A p pontra ható erő a 0-as koordináta rendszerben

F_R^0 - A R pontra ható erő a 0-as koordináta rendszerben

Ki kell számolni a p és az R pontok pozícióját (12)(13) az x_0 és y_0 koordináta rendszerben (ábra 5), ehhez a homogén transzformációs mátrixot (5) használom. A 5-ös ábrán a ϕ szög a bicopter dőlését írja le az (x_0, y_0) és (x_1, y_1) koordináta rendszerek között.



ábra 5. Koordináta rendszer

$$H_1^O = \begin{bmatrix} R_1^O & 0_1^O \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} \cos \phi & -\sin \phi & x \\ \sin \phi & \cos \phi & y \\ 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$P^0 = H_1^0 P^1 \quad (6)$$

$$P^1 = \frac{l}{2} \quad (7)$$

$$P^0 = H_1^0 \begin{bmatrix} l/2 \\ 0 \\ 1 \end{bmatrix} \quad (8)$$

$$R^0 = H_1^0 R^1 \quad (9)$$

$$R^1 = -\frac{l}{2} \quad (10)$$

$$R^0 = H_1^0 \begin{bmatrix} -l/2 \\ 0 \\ 1 \end{bmatrix} \quad (11)$$

$$P^0 = \begin{bmatrix} x + \frac{l}{2} \cos \phi \\ y + \frac{l}{2} \sin \phi \\ 1 \end{bmatrix} \quad (12)$$

$$R^0 = \begin{bmatrix} x - \frac{l}{2} \cos \phi \\ y - \frac{l}{2} \sin \phi \\ 1 \end{bmatrix} \quad (13)$$

A Jacobi-mátrix a p és az R pontra (14) (15).

$$J_p = \frac{\partial P^0}{\partial x} = \begin{bmatrix} 1 & 0 & -0.5l \sin \phi \\ 0 & 1 & 0.5l \cos \phi \\ 0 & 0 & 0 \end{bmatrix} \quad (14)$$

$$J_R = \frac{\partial R^0}{\partial x} = \begin{bmatrix} 1 & 0 & 0.5l \sin \phi \\ 0 & 1 & -0.5l \cos \phi \\ 0 & 0 & 0 \end{bmatrix} \quad (15)$$

A p pontban fellépő erő a nullás koordináta rendszerben (16) és az R pontban fellépő erő a nulla koordináta rendszerben (18).

$$F_p^0 = R_1^0 F_p^1 = \quad (16)$$

$$= \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} 0 \\ U_1 \end{bmatrix} = \begin{bmatrix} -U_1 \sin \phi \\ U_1 \cos \phi \end{bmatrix} \quad (17)$$

$$F_R^0 = R_1^0 F_R^1 = \quad (18)$$

$$= \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} 0 \\ U_2 \end{bmatrix} = \begin{bmatrix} -U_2 \sin \phi \\ U_2 \cos \phi \end{bmatrix} \quad (19)$$

$$Q_J = J_P^{0T} F_P^0 + J_R^{0T} F_R^0 = \quad (20)$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -0.5 \sin \phi & 0.5 \cos \phi \end{bmatrix} \begin{bmatrix} -U_1 \sin \phi \\ U_1 \cos \phi \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0.5 \sin \phi & 0.5 \cos \phi \end{bmatrix} \begin{bmatrix} -U_2 \sin \phi \\ U_2 \cos \phi \end{bmatrix} = \quad (21)$$

$$= \begin{bmatrix} -(U_1 + U_2) \sin \phi \\ (U_1 + U_2) \cos \phi \\ (U_1 - U_2) \frac{1}{2} l \end{bmatrix} \quad (22)$$

Az Euler-Lagrange-egyenlet (23) megoldása után fogom megkapni a rendszer modellemt, azonban előtte a Lagrange függvénynek meg kell határozzam az elsőrendű parciális deriváltjait x, y, ϕ alapján, amiket egyenlővé teszek a nekik megfelelő erő komponensekkel (25) (28) (30). Az így kapott egyenletekből, miután kifejezem az $\ddot{x}, \ddot{y}, \ddot{\phi}$ kapom a síkbeli rendszerként leírható matematikai modellt (33) (34) (35). A könnyebb felírás érdekében bevezettem a felhajtó erők összegére és különbségére az U_s és U_d -t (36) (37). Ezután kaptam a végső összefüggést (38).

$$\frac{d}{dL} \left(\frac{\partial L}{\partial \dot{q}_j} \right) - \frac{\partial L}{\partial q_j} = Q_j \quad (23)$$

$$L = \frac{1}{2} [m(\dot{x}^2 + \dot{y}^2) + I\dot{\phi}^2] - mgy \quad (24)$$

$$\frac{d}{dL} (m\dot{x}) - 0 = -(U_1 + U_2) \sin \phi \quad (25)$$

$$m\ddot{x} = -(U_1 + U_2) \sin \phi \quad (26)$$

$$\ddot{x} = -\frac{(U_1 + U_2)}{m} \sin \phi \quad (27)$$

$$\frac{d}{dL} (m\dot{y}) - (-mg) = (U_1 + U_2) \cos \phi \quad (28)$$

$$m\ddot{y} = -mg + (U_1 + U_2) \cos \phi \quad (30)$$

$$\ddot{y} = -g + \frac{(U_1 + U_2)}{m} \cos \phi \quad (29)$$

$$\frac{d}{dL}(I\dot{\phi}) = (U_1 - U_2)0.5l \quad (31)$$

$$\ddot{\phi} = \frac{(U_1 - U_2)}{I} 0.5l \quad (32)$$

$$\ddot{x} = -\frac{(U_1 - U_2)}{m} \sin \phi \quad (33)$$

$$\ddot{y} = -g + \frac{(U_1 + U_2)}{m} \cos \phi \quad (34)$$

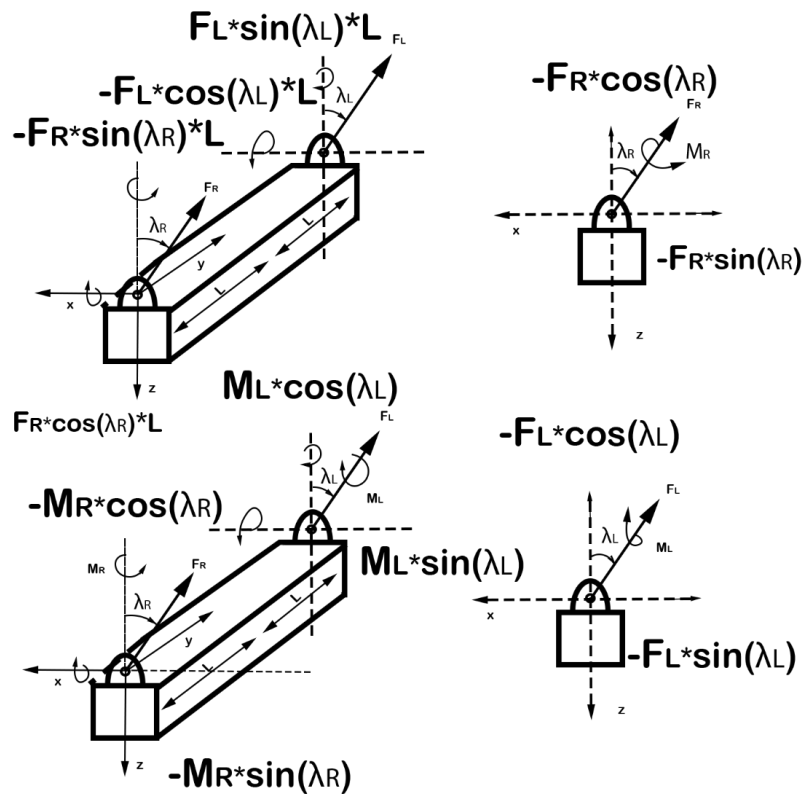
$$\ddot{\phi} = \frac{(U_1 - U_2)}{I} 0.5l \quad (35)$$

$$U_s = U_1 + U_2 \quad (36)$$

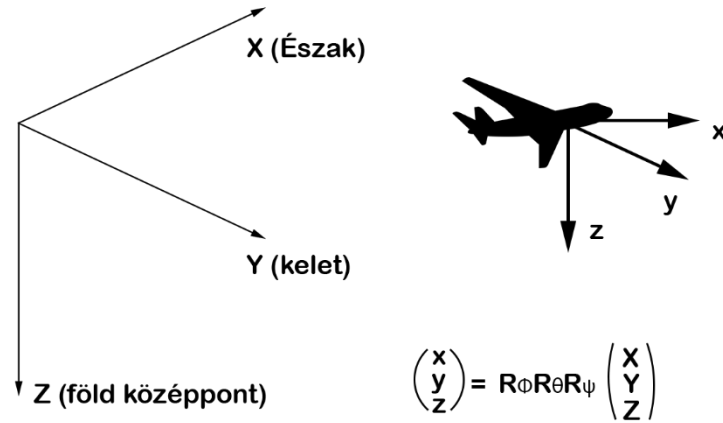
$$U_d = U_1 - U_2 \quad (37)$$

$$\begin{cases} \ddot{x} = -\frac{U_s}{m} \sin \phi \\ \ddot{y} = -g + \frac{U_s}{m} \cos \phi \\ \ddot{\phi} = \frac{1}{2} \frac{U_d}{I} l \end{cases} \quad (38)$$

3.2. Kétmotoros légi robot (bicopter) térbeli matematikai modellje



ábra 6. A bicopter térbeli kinézetei



ábra 7. A bicopter koordináta rendszerei

Az orientáció, a pozíció mellett egy test térbeli elhelyezkedésének leírására szolgál egy referenciakoordináta-rendszerben. Egyik leírási formája Roll(ϕ) – Pitch(Θ) – Yaw(ψ) (Billenés – Bólintás – Elfordulás) szögek, nagyon gyakran használt módszer a repülőgép irányításban. Három elemi rotáció eredménye $Rot(x, \phi)Rot(y, \Theta)Rot(z, \psi)$, így kapjuk a rotációs mátrixot (39).

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} \cos \Theta \cos \psi & \cos \Theta \sin \psi & -\sin \Theta \\ \sin \phi \sin \Theta \cos \psi - \cos \phi \sin \psi & \sin \phi \sin \Theta \sin \psi + \cos \phi \cos \psi & \sin \phi \cos \Theta \\ \cos \phi \sin \Theta \cos \psi + \sin \phi \sin \psi & \cos \phi \sin \Theta \sin \psi - \sin \phi \cos \psi & \cos \phi \cos \Theta \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (39)$$

A (39) rotációs mátrix fordítottja látható a (40) -es rotációs mátrix.

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} \cos \Theta \cos \psi & \sin \phi \sin \Theta \cos \psi - \cos \phi \sin \psi & \cos \phi \sin \Theta \cos \psi + \sin \phi \sin \psi \\ \cos \Theta \sin \psi & \sin \phi \sin \Theta \sin \psi + \cos \phi \cos \psi & \cos \phi \sin \Theta \sin \psi - \sin \phi \cos \psi \\ -\sin \Theta & \sin \phi \cos \Theta & \cos \phi \cos \Theta \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (40)$$

Az erők translációjának esetében a 40-es rotációs mátrixot használom. Felírom a három tengely mentén az erők összegét; lásd a (41) (42) (43) -es egyenleteket.

$$\sum F_X = -F_R \sin \lambda_R - F_L \sin \lambda_L \quad (41)$$

$$\sum F_Y = 0 \quad (42)$$

$$\sum F_Z = -F_R \cos \lambda_R - F_L \cos \lambda_L \quad (43)$$

Newton második törvénye értelmében ($ma = \sum F$), tömeg szorozva gyorsulás egyenlő az erők összegével. A(41) (42) (43) -as egyenleteket egyenlővé tehetem a tömeg szorozva a gyorsulással felbontva x,y,z tengelyek mentén és megkapom a (44) (45) (46) -os egyenleteket.

$$m\ddot{x} = -F_R \sin \lambda_R - F_L \sin \lambda_L \quad (44)$$

$$m\ddot{y} = 0 \quad (45)$$

$$m\ddot{z} = -F_R \cos \lambda_R - F_L \cos \lambda_L \quad (46)$$

Ha végig osztom a (44) (45) (46) -os egyenleteket m-el és megszorozom ezeket a 40-es rotációs mátrixszal, lásd (47), akkor megkapom a x,y,z gyorsulásokat R-P-Y szög orientációban; lásd a (48) (49) (50).

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = R_{bg} \begin{pmatrix} \frac{-F_R \sin \lambda_R - F_L \sin \lambda_L}{m} \\ 0 \\ \frac{-F_R \cos \lambda_R - F_L \cos \lambda_L}{m} \end{pmatrix} \quad (47)$$

A gravitációs gyorsulást csak az (50) -nél adom hozzá.

$$\ddot{x} = -\frac{T_1}{m} (\cos \phi \sin \Theta \cos \psi + \sin \phi \cos \psi) - \cos \Theta \cos \psi \frac{T_3}{m} \quad (48)$$

$$\ddot{y} = -\frac{T_3}{m} \cos \Theta \sin \phi + \frac{T_1}{m} (\sin \phi \cos \psi - \cos \phi \sin \Theta \sin \psi) \quad (49)$$

$$\ddot{z} = \frac{T_3}{m} \sin \Theta - \frac{T_1}{m} \cos \phi \cos \Theta + g \quad (50)$$

F_R és F_L a jobb- és baloldali felhajtóerő (55) (56). Az ω vektor tartalmazza a szögsebességeket (57). Az (48) (49) (50) -es egyenletekben a $T_1 T_2 T_3 T_4$ egyenlőek, lásd (51)(52)(53)(54).

$$T_1 = K_T (\omega_R \cos \lambda_R + \omega_L \cos \lambda_L) \quad (51)$$

$$T_2 = K_T (\omega_R \cos \lambda_R - \omega_L \cos \lambda_L) \quad (52)$$

$$T_3 = K_T (\omega_R \sin \lambda_R + \omega_L \sin \lambda_L) \quad (53)$$

$$T_4 = K_T (\omega_R \sin \lambda_R - \omega_L \sin \lambda_L) \quad (54)$$

$$F_R = C_T \omega_R^2 \quad (55)$$

$$F_L = C_T \omega_L^2 \quad (56)$$

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} = \omega \quad (57)$$

Felírom az Euler egyenletek segítségével a test koordináta rendszerben a forgó mozgásokat (58) (59) (60).

$$\sum M_x = I_{xx}\dot{p} - (I_{yy} - I_{zz})qr \quad (58)$$

$$\sum M_y = I_{yy}\dot{q} - (I_{zz} - I_{xx})rp \quad (59)$$

$$\sum M_z = I_{zz}\dot{r} - (I_{xx} - I_{yy})pq \quad (60)$$

Az (58) (59) (60) -ból kiemelem a szöggyorsulásokat és jobb oldalra hozom (61) (62) (63).

$$\dot{p} = \frac{\sum M_x + (I_{yy} - I_{zz})qr}{I_{xx}} \quad (61)$$

$$\dot{q} = \frac{\sum M_y + (I_{zz} - I_{xx})rp}{I_{yy}} \quad (62)$$

$$\dot{r} = \frac{\sum M_z + (I_{xx} - I_{yy})pq}{I_{zz}} \quad (63)$$

A külső nyomatékok lehetnek a felhajtó erők által létrehozott forgató nyomatékok (64) (65) (66).

$$\sum M_x = F_L \cos \lambda_L L - F_R \cos \lambda_R L \quad (64)$$

$$\sum M_y = F_L \cos \lambda_L L - F_R \cos \lambda_R L \quad (65)$$

$$\sum M_z = F_L \cos \lambda_L L - F_R \cos \lambda_R L \quad (66)$$

Még figyelembe lehet venni a giroszkópikus hatásokat (67) (68) (69). A (67) (68) (69) -et még át kell vinni inerciális rendszerbe (70).

$$\dot{p} = \frac{LT_2}{I_{xx}} + \frac{(I_{yy} - I_{zz})}{I_{xx}} qr \quad (67)$$

$$\dot{q} = \frac{-hT_3}{I_{yy}} + \frac{(I_{yy} - I_{xx})}{I_{yy}} rp \quad (68)$$

$$\dot{r} = \frac{LT_4}{I_{zz}} + \frac{(I_{xx} - I_{yy})}{I_{zz}} pq \quad (69)$$

$$\begin{pmatrix} \ddot{\phi} \\ \ddot{\Theta} \\ \ddot{\psi} \end{pmatrix} = J \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} 1 & \sin \phi \tan \Theta & \cos \phi \tan \Theta \\ 0 & \cos \phi & \sin \phi \\ 0 & \frac{\sin \phi}{\cos \Theta} & \frac{\cos \phi}{\cos \Theta} \end{pmatrix} \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} \quad (70)$$

$$\ddot{\phi} = \frac{LT_2}{I_{xx}} + \sin \phi \tan \Theta \frac{-hT_3}{I_{yy}} + \cos \phi \tan \Theta \frac{LT_4}{I_{zz}} \quad (71)$$

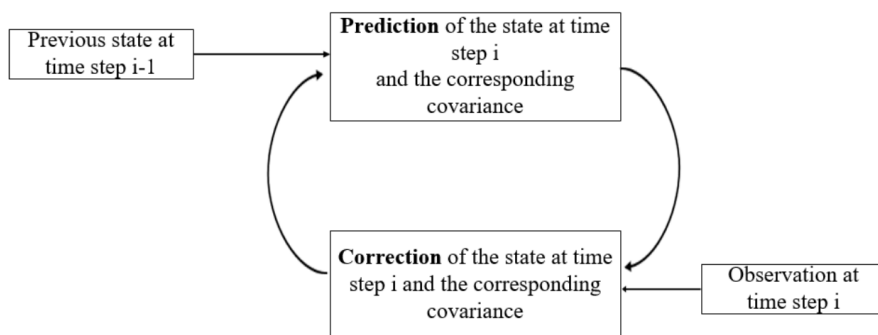
$$\ddot{\Theta} = \cos \phi \frac{-hT_3}{I_{yy}} + \sin \phi \frac{LT_4}{I_{zz}} \quad (72)$$

$$\ddot{\psi} = \frac{\sin \phi}{\cos \Theta} \frac{-hT_3}{I_{yy}} + \frac{\cos \phi}{\cos \Theta} \frac{LT_4}{I_{zz}} \quad (73)$$

3.3. Bevezetés a Kálmán-szűrő elméletébe

A Kálmán-szűrő egy rekurzív prediktív szűrő, amely állapotter technikákat és rekurzív algoritmust használ. A zajos bemenő adatok rekurzív mérésével egy optimális becslést ad a Kálmán-szűrő a mérés állapotáról. Ezt a dinamikus rendszert többnyire fehér zaj zavarhatja. A becsült állapot javítása érdekében a Kálmán-szűrő az aktuális kalkulált állapotot, és az aktuális mérési eredményeket veszi figyelembe. A Kálmán-szűrő két lépésből áll: predikcióból és korrekcióból.

Első lépésben az állapot előrejelzése történik a dinamikus modell alapján. A második lépésben a megfigyelési modellel korrigáljuk, így a becsülő hiba kovarianciája minimálisra csökken. Ebben az értelemben a Kálmán-szűrő egy optimális becsülő. Ez az eljárás minden időlépésben megismétlődik, kiindulásnak az előző lépés állapotát tekinti. A Kálmán-szűrő visszacsatolási mechanizmust használ a folyamat becslésére. A zajos méréseket visszacsatolja és felhasználja a folyamat becslésére. Tehát általánosan a Kálmán-szűrő alapvető összetevői az állapotvektor, a dinamikus modell és megfigyelési modell [10].



ábra 8. Kálmán-szűrő algoritmus

A (74) -es képlet a Kálmán-szűrő által megbecsült állapotot írja le. A (75) -ös a kovariancia predikcióját adja meg.

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k, 0) \quad (74)$$

$$P_{k+1}^- = A_k P_k A_k^T + W_k Q_k W_k^T \quad (75)$$

A (76)-os képlet a kimenet becslését, a (77)-es képlet az optimális Kálmán-szűrő erősítését adja meg és a (78)-as képlet kovariancia értékét frissíti.

$$\hat{x}_k = f(\hat{x}_k^-, u) + K[z_k - h(\hat{x}_k^-, 0)] \quad (76)$$

$$K_k = P_k^- H_K^T (H_K P_k^- H_K^T + R_K)^{-1} \quad (77)$$

$$P_k = (I - H_K K_k) P_k^- \quad (78)$$

4. Modell szimuláció, validáció

A 3.1 és a 3.2-es alfejezetekben levezetett bicopter matematikai modelleket a Simulink modellező szoftver segítségével építettem fel, hogy megbizonyosodjak arról, hogy a szimulációban szereplő modellek és adatok valósághűen tükrözik a való életben tapasztalt jelenségeket. Minden matematikai modell felállítása után szükséges bizonyítani, hogy a modell élethűen reprodukálja a fizikai rendszert. A validáció során megvizsgálom, hogy a szimuláció eredményei hogyan viszonyulnak a valós életben tapasztalt adatokhoz, és hogy azok mennyire pontosan és megbízhatóan reprodukálják az adott jelenséget.

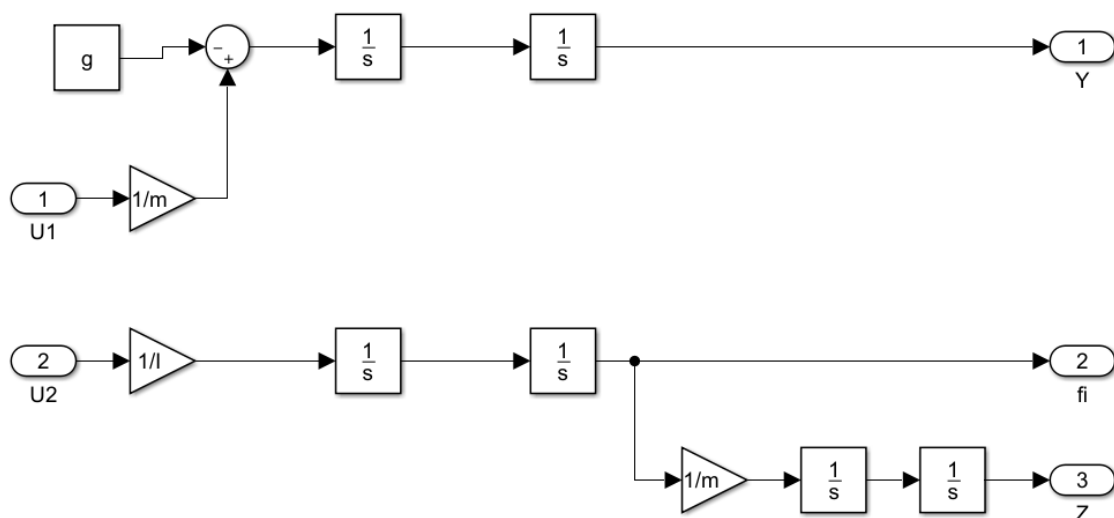
4.1. Kétmotoros légi robot (bicopter) síkbeli matematikai modelljének validációja és szimulációja

Első lépésként felépítettem Simulink környezetben a (38) matematikai egyenletrendszerből kiindulva a síkbeli bicopter modelljét. Elfogadtam, hogy a ϕ szög megközelítőleg 0 és 5 fok között lehet, ebből kifolyólag a (38) -as egyenletrendszerből lineáris egyenleteket kapok, lásd a (79) (80) (81) egyenleteket. Ez a folyamatmodell az 9-es ábrán látható.

$$\ddot{x} = -\frac{U_s}{m} \quad (79)$$

$$\ddot{y} = -g + \frac{U_s}{m} \phi \quad (80)$$

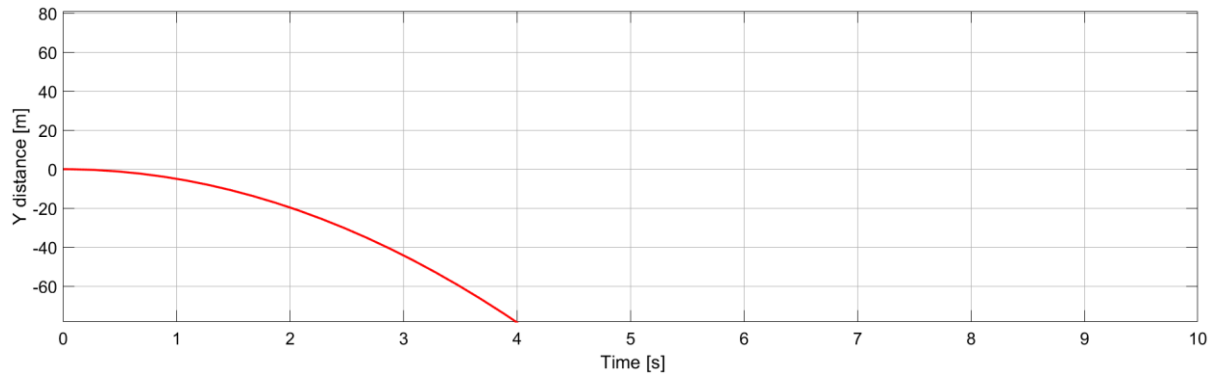
$$\ddot{\phi} = \frac{1}{2} \frac{U_d}{l} l \quad (81)$$



ábra 9. A bicopter lineáris síkbeli modellnek a Simulink megvalósítása

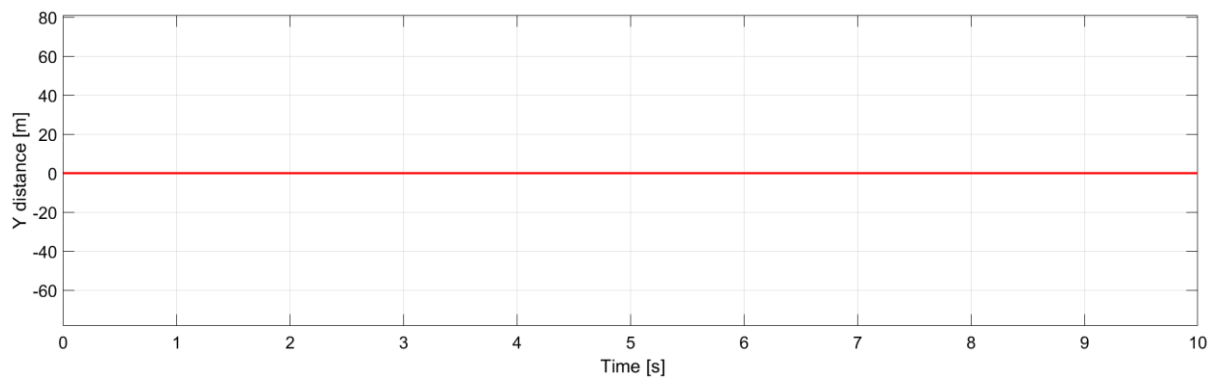
Ahhoz, hogy a modell helyességét igazoljam, a következő méréseket végeztem a rendszeren. Az U_1 -es és az U_2 -es bemenetekre 0 bemeneti paramétert állítottam be. Ez azt

jelenti, hogy nem vezérem a metrókat, ebből kifolyólag a gravitációs erő elkezd a Föld fele vonzani. Ez egy negatív elmozdulásnak felel meg az Y tengelyen, lásd a 10-es ábrát.

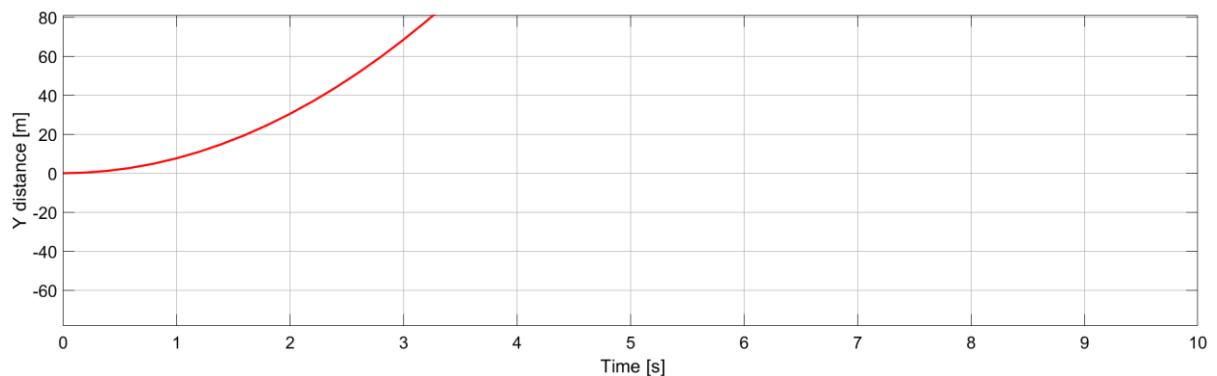


ábra 10. Gravitációs erő hatására az elmozdulása (esés)

Kiindulva a dinamika alaptörvényéből ($F = ma$), ahhoz, hogy a gravitációs erőt kiegyenlítsük $U_1 = mg$ vezérlőjelre van szükségünk, lásd a 11-es ábrát, ha ennél nagyobb a vezérlőjel, akkor Y tengelyen pozitív elmozdulás következik be (felszáll); lásd a 12-es ábrát.



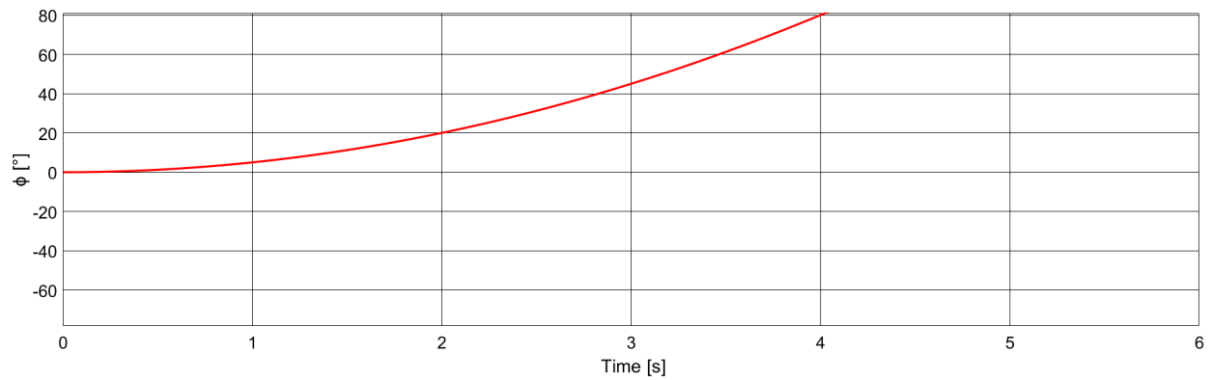
ábra 11. Bicopter gravitációs erő kompenzációja



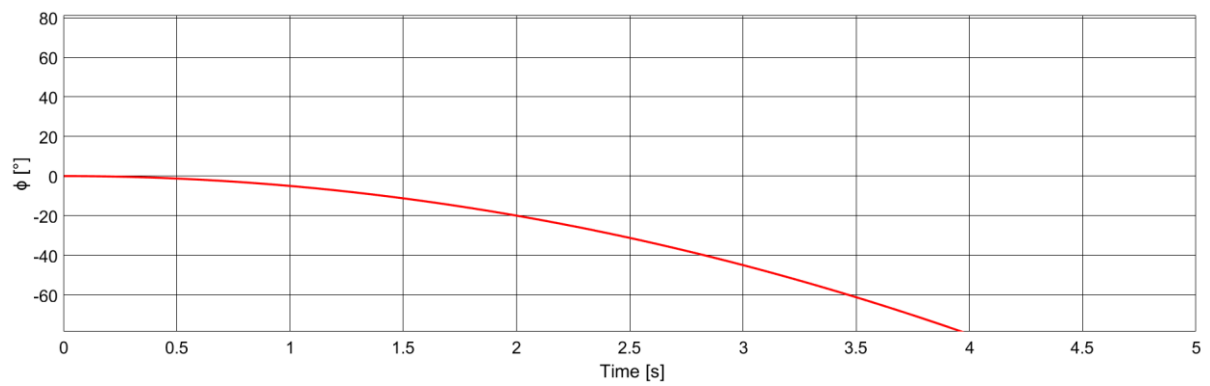
ábra 12. Bicopter Y tengelyi elmozdulása ($U_1 = 5N$, $U_2 = 5N$)

Ha $U_s > 0$, akkor a ϕ szög pozitív lesz és ha $U_s < 0$, akkor ϕ szög értéke negatív lesz. Ez annak köszönhető, hogy amikor $U_s > 0$ akkor U_1 felhajtó ereje megnő, míg az U_2 felhajtó

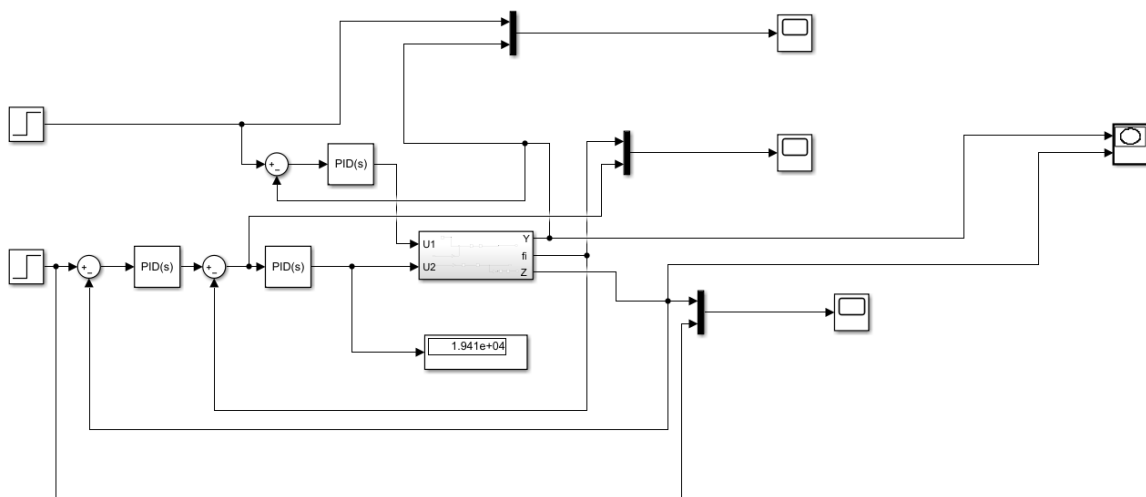
ereje lecsökken, ami egy pozitív dőlést eredményez; lásd a 13-as ábrát. Ha pedig $U_s < 0$, akkor U_1 felhajtó ereje lecsökken, míg az U_2 felhajtó ereje megnő, ami egy negatív dőlést eredményez; lásd a 14-es ábrát.



ábra 13. Pozitív ϕ szög

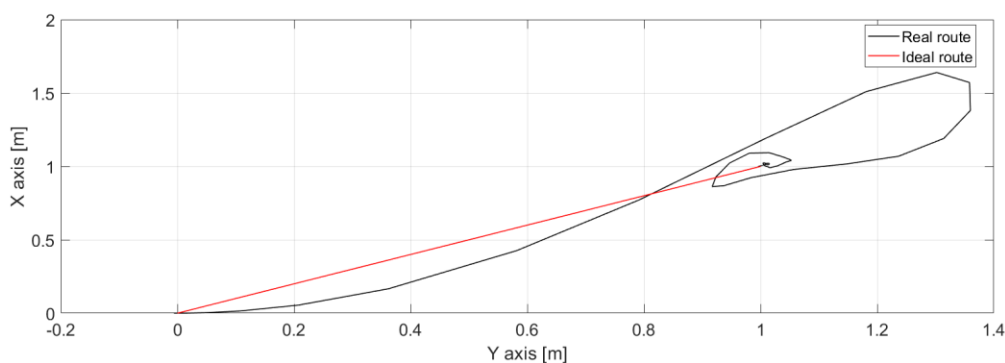


ábra 14. Negatív ϕ szög



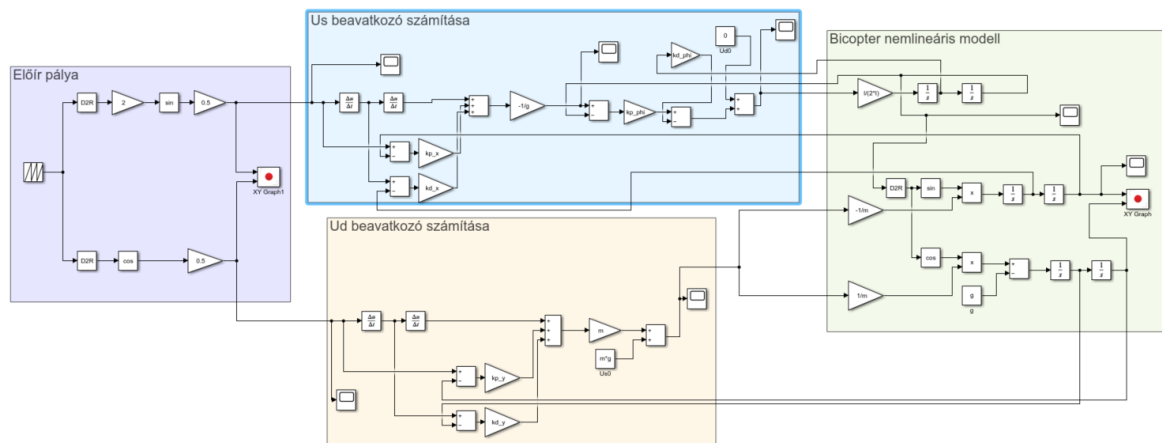
ábra 15. Bicopter PID szabályozó

Végző soron kialakítottam egy szabályzó kört PID szabályozók segítségével; lásd a 15-ös ábrát. Így tesztelhettem, hogy a rendszert lehetséges egy tetszőleges (x,y) koordinátába elvinni. A 16-os ábrán látható, hogy sikeresen elmozdult az (1,1) pontba a bicopter.

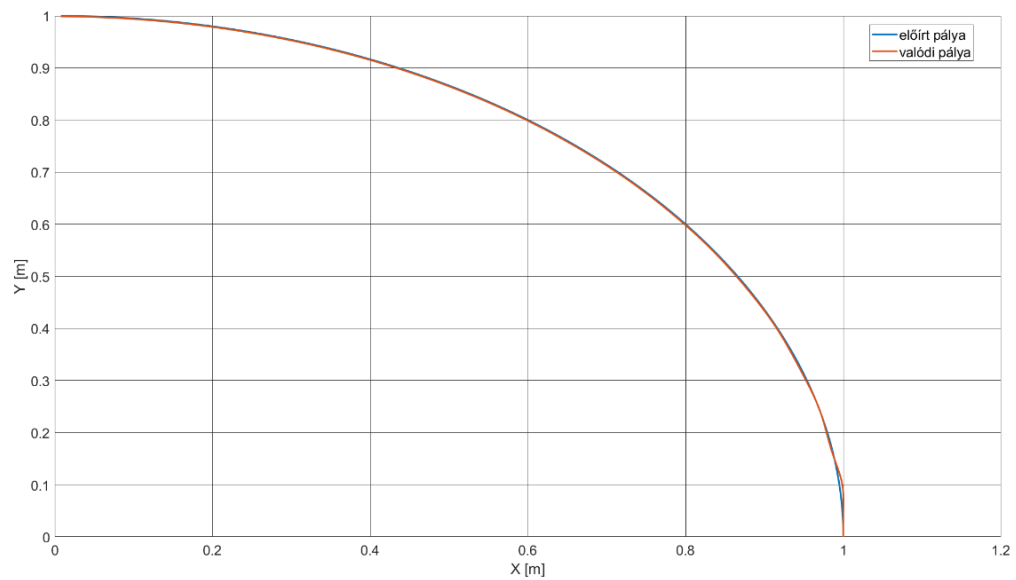


ábra 16. Bicopter (1,1) pozícióba való elmozdítása

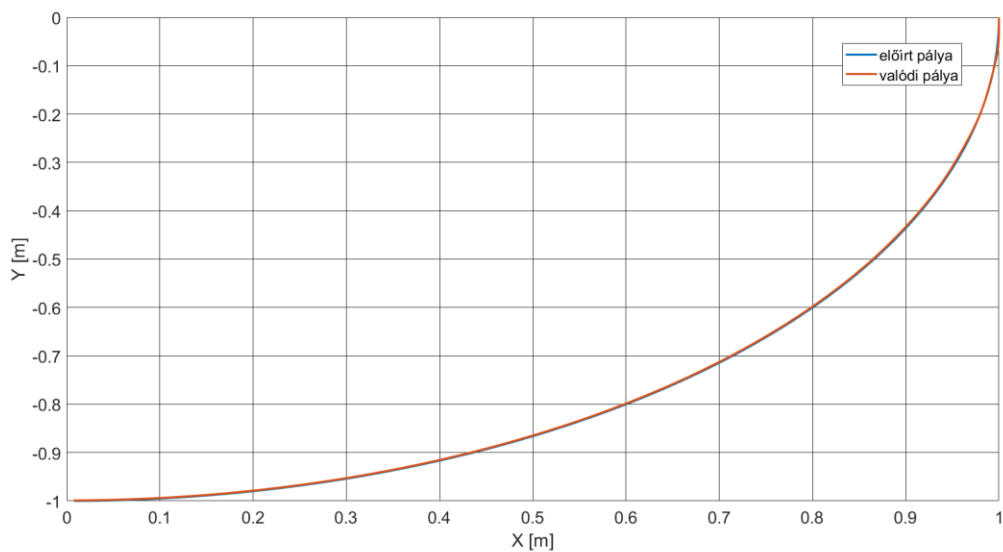
A 17-es ábrán látható a bicopter nem lineáris síkbeli modelljének a felépített változata Simulink szimulációs környezetben. A modell mellé van helyezve egy pálya követést megvalósító szabályozó is. A $K_{p_y} = 300$, $K_{p_x} = 300$, $K_{p_{\phi}} = 2500$, $K_{d_y} = 2^2 \sqrt{K_{p_y}}$, $K_{d_x} = 2^2 \sqrt{K_{p_x}}$, $K_{d_{\phi}} = 2^2 \sqrt{K_{p_{\phi}}}$. A 14-es ábrán látható hogy 0-tól -pi/2-es negyedkört írok le.



ábra 17. Nem lineáris bicopter matematikai modelljének pályaszabályozása

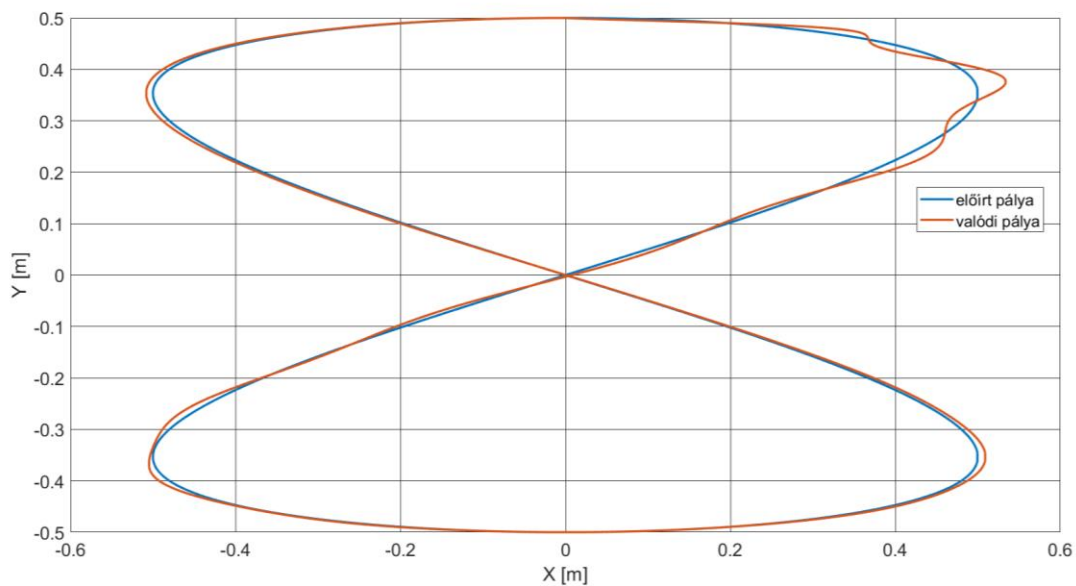


ábra 18. Bicopter pályakövetése lenti irányba



ábra 19. Bicopter pályakövetése fenti irányba

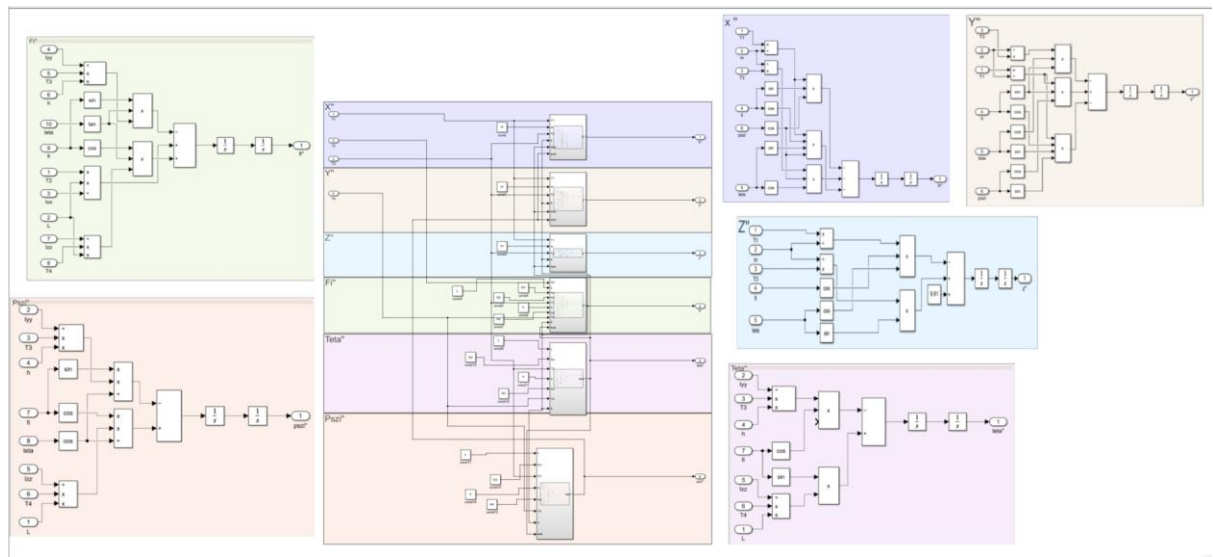
A fentiek alapján elmondható, hogy a 3.1 alfejezetben levezettet síkbeli bicopter matematikai modellje nagymértékben tükrözi a bicopter fizikai valósághű modelljét. A 5-ös fejezetben tervezett pályakövető szabályozóval leírtam egy 8-as pályát sikeresen lásd a (20)-as ábrát.



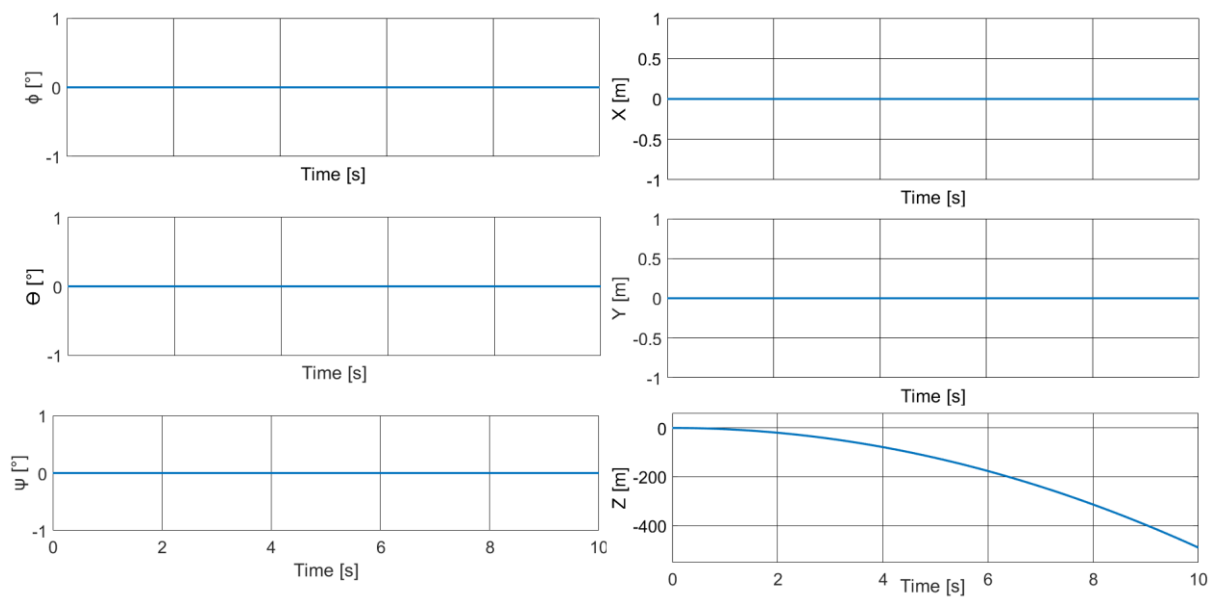
ábra 20. 8-as pálya

4.2. Kétmotoros légi robot (bicopter) térbeli matematikai modelljének validációja és szimulációja

A 21-es ábrán látható a bicopter térbeli matematikai modellje felépítve a 3.2-es alfejezetben levezetett egyenletekből. Ha bemenetekre (T_1 , T_2 , T_3 , T_4) 0 értéket teszek, akkor megfigyelhető egy vízszintes esés a Z tengelyen. Ez annak tudható be, hogy ha nincs felhajtó erő a rendszeren, akkor csak a gravitációs gyorsulás fog hatni rá; lásd a 22-es ábrát [11].

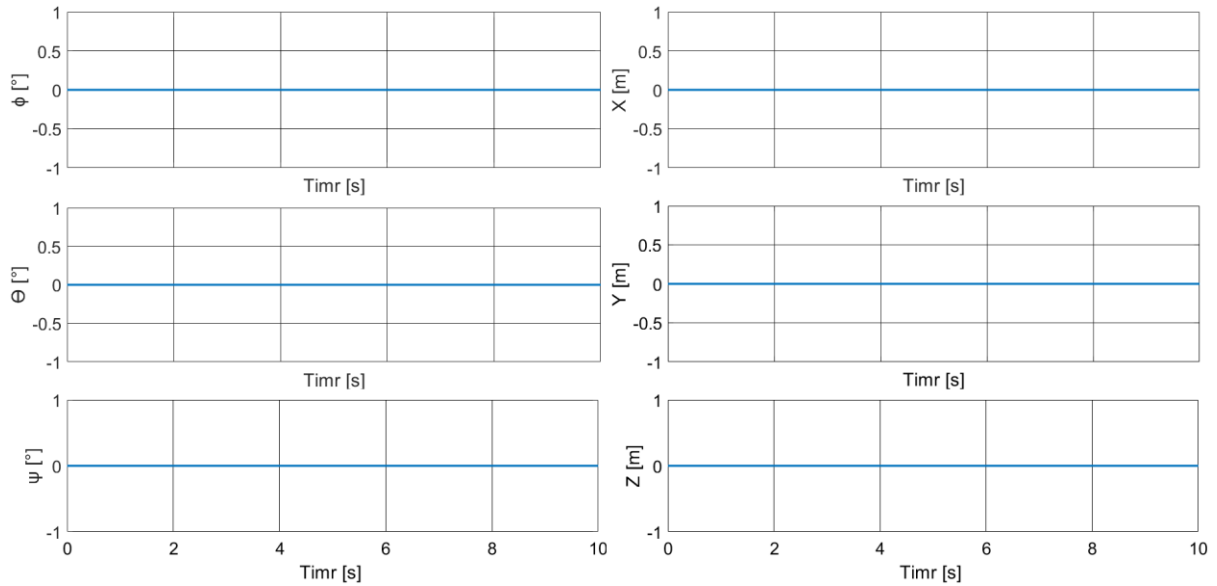


ábra 21. Bicopter térbeli modell Simulink megvalósítása



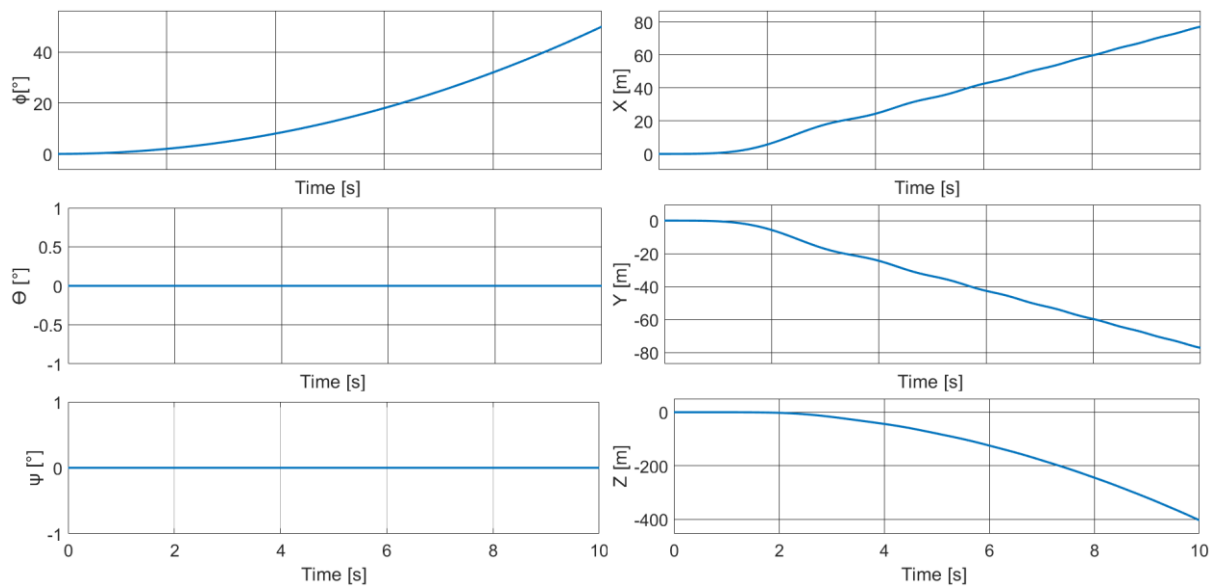
ábra 22. Bicopter térbeli modelljére ható gravitációs erő

Ahhoz, hogy kiegyenlítsem a gravitációs erőt, amint látható az 51-es egyenlet alapján is, hogy $T1 = g/m$ nagyságú vezérlő jelre van szükség (m- a bicopter tömege és a g- a gravitációs gyorsulás); lásd a 23-as ábrát.

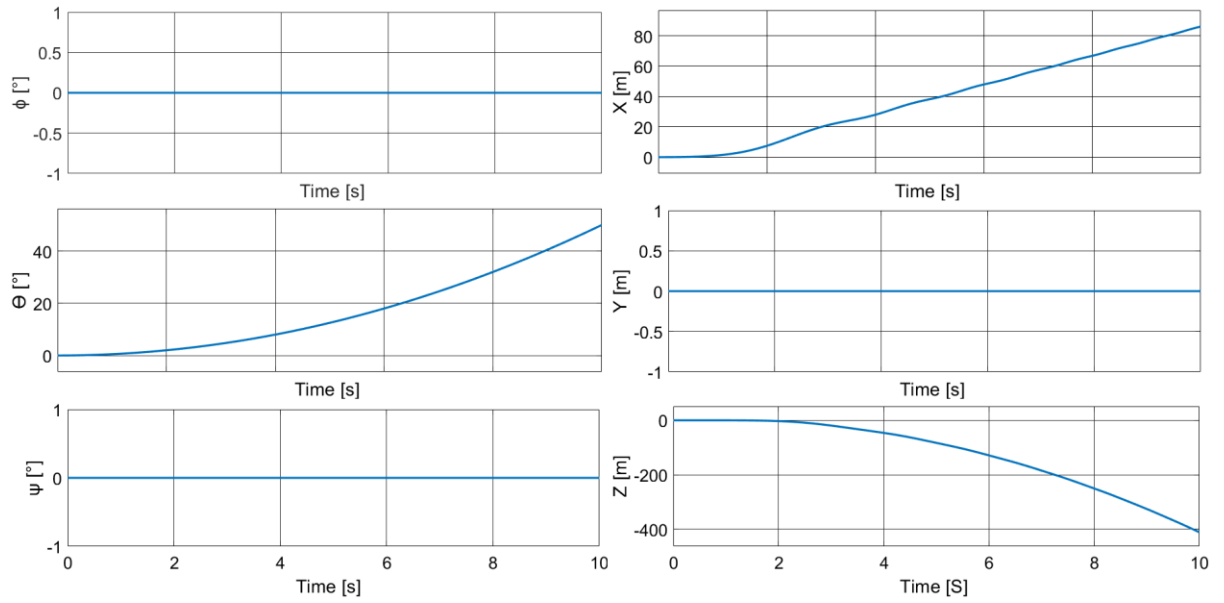


ábra 23. Térbeli gravitációs kompenzálás

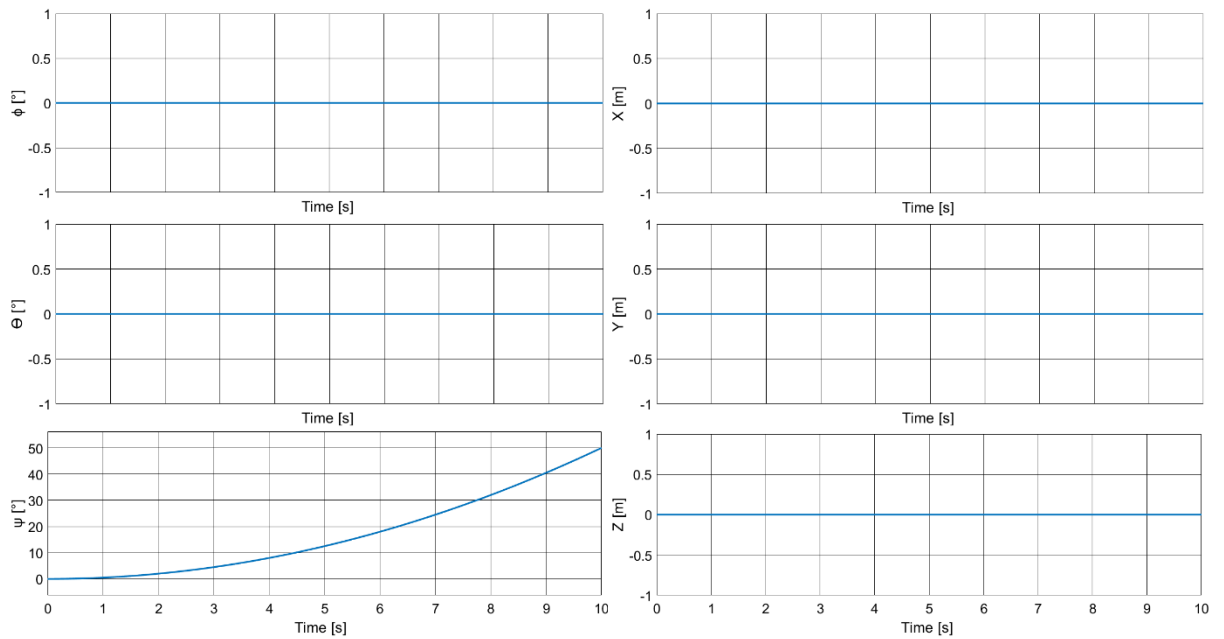
A (68)(68)(70) egyenletek lapján látható, hogy T2 a ϕ , a T3 a θ és a T4 a ψ szög értékét lehet változtatni; lásd a 24, 25, 26 -os ábrákat.



ábra 24. A T2, ami 2N, vezérlőjel hatása a modellre

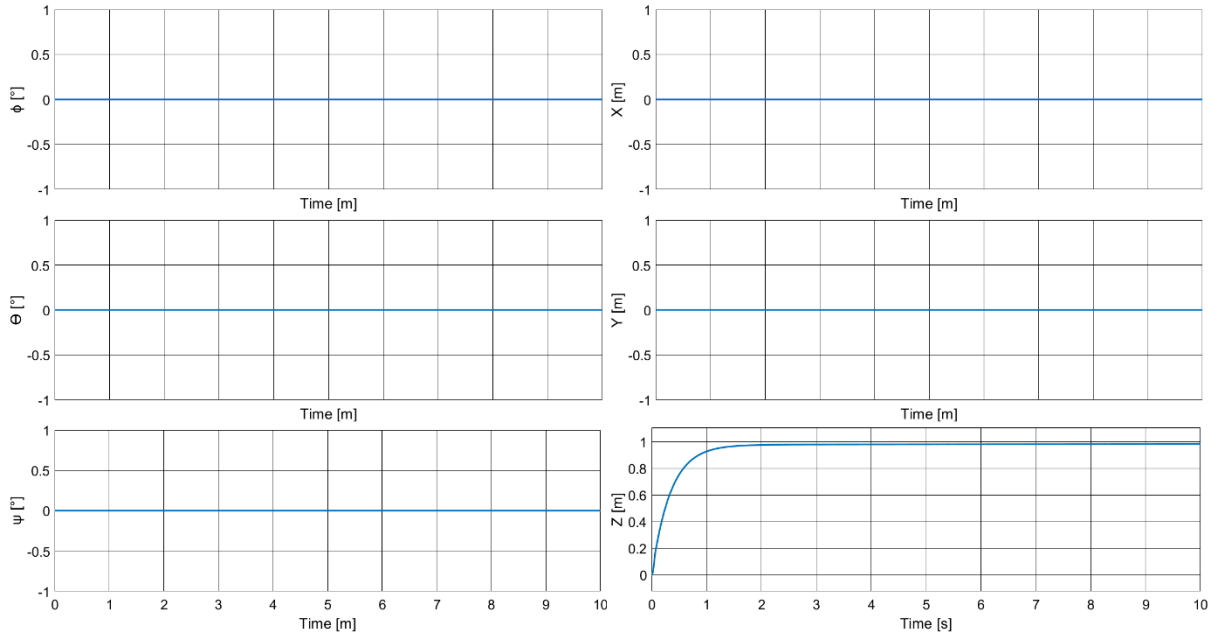


ábra 25. A T3, ami 2N, vezérlőjel hatása a modellre



ábra 26. A T4, ami 2N, vezérlőjel hatása a modellre

Egy PID szabályozó segítségével megvalósítottam a magasság szabályozását ($P = 300$, $I = 10$, $D = 100$); lásd a 27-es ábrát. Látható, hogy ezek a paraméterek mellett nincs túllövés, csak hiba jelenik meg.



ábra 27. Z tengelyen levő elmozdulás szabályozása PID-el

A fentiek értelmében levonhatom azt a következtetést, hogy a matematikai modellem helyesen tükrözi a fizikai valóságot.

5. Pályakövető szabályozó tervezése a síkbeli modell alapján

Általában egy nemlineáris folyamat a következő formában írható fel.

$$\dot{x} = f(x, u) \quad (82)$$

$$y = h(x, u) \quad (83)$$

Az x állapotok változásai az állapotok és az u bemenetek nemlineáris kombinációi, a z kimenetek ugyancsak szintén az állapotok és bemenetek nemlineáris kombinációi. A folyamat dinamikáját az f nemlinearitás írja le. A nemlineáris rendszerek fontos osztályát képezik az úgynevezett kanonikus irányíthatósági alakban felírt rendszerek. Ha a rendszernek n állapota van, akkor a következőképpen alakul a rendszer modellje.

$$\dot{x}_1 = x_2 \quad (84)$$

$$\dot{x}_2 = x_3 \quad (85)$$

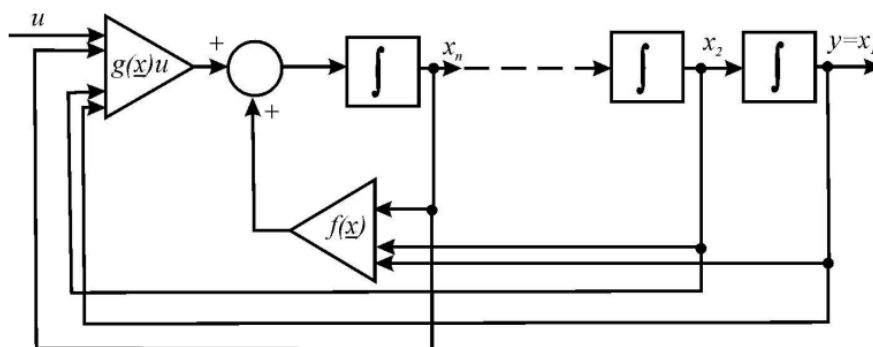
...

$$\dot{x}_{n-1} = x_n \quad (86)$$

$$\dot{x}_n = f(x) + g(x)u \quad (87)$$

$$y = x_1 \quad (88)$$

A rendszer kimenete az első állapot. A rendszerosztály fontos tulajdonsága, hogy a bemenetben lineáris, vagyis az állapotváltozás dinamikájában a bemenet additív komponensként jelenik meg. A bemenetet állapotfüggő erősítés $g(x)$ szorozza. Észrevehető, hogy az irányíthatóság feltétele, hogy a $g(x)$ ne legyen nulla az állapotter egyetlen pontjában sem. Máskülönben a bemenetnek nincs hatása a rendszer dinamikájára.



ábra 28 Nemlineáris rendszer kanonikus alakjának tömbvázlata

A lineáris visszacsatolás egy módszer arra, hogy a nemlineáris modelleket liberalizáljuk. Az alap ötlet az, hogy az irányítási algoritmusba olyan nemlineáris elemeket építsünk be, amely kiejti a folyamatban levő nemlineáris hatásokat. Az így kapott rendszerhez lineáris szabályozót tervezünk.

A szabályozó jelet a következő formában tudjuk általánosan felírni, lásd (89) ahol U_0 -állandósult állapotbeli szabályzó jel és δU -a visszacsatolt szabályzó jelből tevődik össze, az utóbbi linearizálja a rendszert. Első lépésben megkeresem az állandósultbeli jelet ($\ddot{x} = \dot{y} = \ddot{\phi} = \dot{x} = \dot{y} = \dot{\phi} = \phi = 0$).

$$U = U_0 + \delta U \quad (89)$$

A bicopter síkbeli egyenleteit (90)(91)(92) egyenlővé teszem nullával (93) (94) (95).

$$m\ddot{x} = -U_s \sin \phi \quad (90)$$

$$m\ddot{y} = -mg + U_s \cos \phi \quad (91)$$

$$I\ddot{\phi} = \frac{1}{2}lU_d \quad (92)$$

A vezérlőjelek a következő formába néznek ki (100) (102).

$$0 = -U_s \sin \phi \quad (93)$$

$$0 = 0 \quad (94)$$

$$0 = -mg + U_s \cos \phi \quad (95)$$

$$U_s^0 = mg \quad (96)$$

$$0 = \frac{1}{2} l U_d \quad (97)$$

$$U_d^0 = 0 \quad (98)$$

$$U_s = U_s^0 + \delta U_s \quad (99)$$

$$U_s = mg + \delta U_s \quad (100)$$

$$U_d = U_d^0 + \delta U_d \quad (101)$$

$$U_d = 0 + \delta U_d \quad (102)$$

A vezérlőjeleket behelyettesítve a modell egyenletekbe kapjuk (104) (107) (110).

$$m\ddot{x} = -(mg + \delta U_s) \sin \phi = -mg \sin \phi - \delta U_s \sin \phi \quad (103)$$

$$\ddot{x} = -g \phi \quad (104)$$

$$m\ddot{y} = -mg + (mg + \delta U_s) \cos \phi \quad (105)$$

$$m\ddot{y} = \delta U_s \quad (106)$$

$$\ddot{y} = \frac{\delta U_s}{m} \quad (107)$$

$$I\ddot{\phi} = \frac{1}{2} l (U_d^0 + \delta U_d) \quad (108)$$

$$\ddot{\phi} = \frac{1}{2I} l (0 + \delta U_d) \quad (109)$$

$$\ddot{\phi} = \frac{1}{2I} l \delta U_d \quad (110)$$

Feltételezve, hogy a pálya n-szer deriválható időfüggvény, választhatjuk a beavatkozó jel lineáris részét:

$$u_{linearis} = y_d^{(n)} + k_0(y_d - x_1) - k_1(y_d^{(1)} - x_2) - \dots - k_{n-1}(y_d^{(n-1)} - x_n) \quad (111)$$

A (104) (107) (110) és (108) alapján következik a (112) (113) (114).

Feedback linearizáció:

$$\phi_{ref} = -\frac{1}{g} \left(\ddot{x}_{ref} + K_{px}(x_{ref} - x) + K_{dx}(\dot{x}_{ref} - \dot{x}) \right) \quad (112)$$

$$\delta U_s = m \left(\ddot{y}_{ref} + K_{py}(y_{ref} - y) + K_{dy}(\dot{y}_{ref} - \dot{y}) \right) \quad (113)$$

$$\delta U_d = -K_d \dot{\phi} + K_p(\phi_{ref} - \phi) \quad (114)$$

$$U_s = U_s^0 + \delta U_s = mg + \delta U_s \quad (115)$$

$$U_d = 0 + \delta U_d = \delta U_d \quad (116)$$

6. Inerciális mérőeszköz jeleinek feldolgozása Kálmán-szűrővel

Az inerciális mérő egység (IMU) az inerciális navigációs rendszerek összetevője – egy navigációs eszköz, amellyel egy mozgó objektum helyzetét, sebességét és tájolását külső hivatkozások nélkül számítják ki. Továbbá, képes a térbeli gyorsulás és elfordulás mérésére az inerciális referencia-koordináta-rendszer három tengelye mentén. Az IMU tartalmaz 3 darab gyorsulásmérőt, 3 darab giroszkópot (elfordulás mérőt) és ez még kiegészülhet magnetométerekkel. A Föld közelében végzett mérések esetén az IMU által mért navigációs értékeket korrigálni kell a gravitációs erőhatások és a Föld forgása miatt.

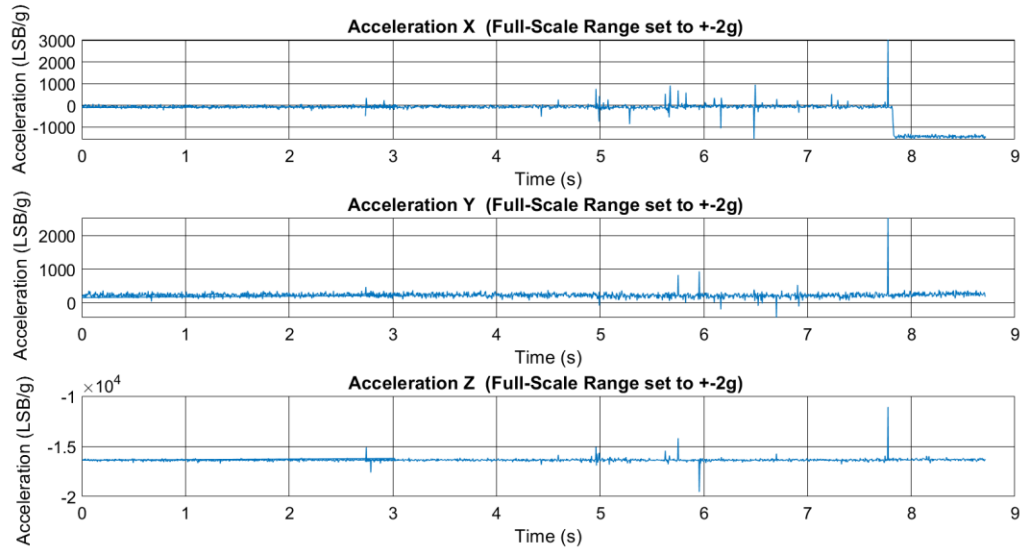
6.1. MPU-9150

Az általam használt IMU szenzor a hobbi elektronikában is használatos MPU-9150, egy 9 tengelyes MotionTracking eszköz, amely egy 3 tengelyes giroszkópot egy 3 tengelyes gyorsulásmérőt és egy 3 tengelyes magnetométert tartalmaz. A giroszkóp segítségével mérhetőek a csavaró-billentő-faroló sebességek ez azt jelenti, hogy nem az abszolút szögeket méri fokban, hanem a szögsebességet másodpercenként fokban (fok/sec). A 10 fok/sec szögsebesség azt jelenti, hogy másodpercenként 10 fokot fordul el és a teljes kört 36 másodperc alatt hajtja végre.

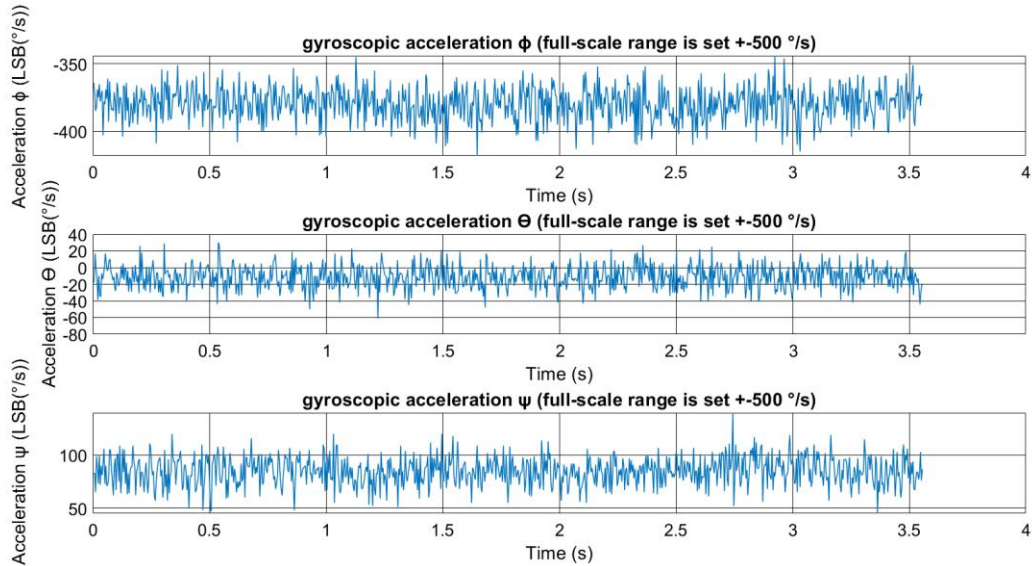
6.1.1. Szenzor adatok kiolvasása

Az MPU-9150 szenzorral I2C soros kommunikációs protokollon keresztül kérdezem le az adatokat. Minden szenzornak van egy egyedi címe. Az MPU-9150 esetében 0x68 érték. Ezzel a címmel lehet megszólítani a szenzort. Szükséges beállítani a szenzor érzékenységi skáláját. Az MPU-9150 szenzor mérései LSB-ben vannak rögzítve. Az érzékenységi skálát FS_SEL = 4-re állítottam. Ez az érték 2000 LSB/(fok/sec) felel meg. Ez azt jelenti, hogy 1 fok/sec szögsebesség +/-2000 LSB-nek felel meg. A funkció aktiválásához 0x1B címre a 0x08 értéket kell írni. Ugyanígy a gyorsulásnak is be kell állítani az érzékenységet. Ezt a következő módon tehetjük meg, ha a 0x1C regiszterbe 0x10 értéket írunk, akkor +/- 8g érzékenységet állítunk be. A szenzor érzékenysége beállítása után lekérhetjük az adatokat. Ezt úgy

tehetjük, meg hogy kiolvassuk a megfelelő regiszterekből az értékeket. A 0x68 kezdő címtől az első 3 byte a gyorsulás értéke X, Y, Z tengelyek mentén. A következő 1 byte a szenzor hőmérsékletének az értéke és a következő 3 byte pedig a giroszkóp három tengelye menti értéke [12].



ábra 29. Gyorsulási nyersadatok



ábra 30. Giroszkópikus gyorsulási nyersadatok

6.1.2. A szenzor kalibrálása

A nyersen beolvasott gyorsulás szenzor adatait mintavételeztem 16Hz-en és 250 Hz-en a giroszkópikus szenzor adatait. Középértéket számoltam 2000 minta alapján. A középértékeket használtam fel a szenzor kalibrálásához.

6.2. Az adatok előszűrése alpha szűrővel

A beolvasott adatokon előszűrést végzek annak érdekében, hogy az alacsony frekvenciás zajokat kiszűrjem. Erre a célra egy alpha szűrőt használok. Az alpha szűrő egy olyan szűrő, amelyet a digitális jelfeldolgozás során használnak a zaj csökkentésére és az alacsony frekvenciás zavarok szűrésére. Az alpha szűrő egy folytonos szűrő, amely az előző mintavételezett érték és az aktuális mintavételezett érték közötti különbséget veszi figyelembe, majd egyenletesen szűri a jelet a két érték között. Az alpha szűrőt gyakran használják az analóg- és digitális átalakítók kimenetén, hogy csökkentsék a zajt és az analóg jelek kis zavarait.

$$y(t) = \alpha y(t-1) + (1-\alpha)y(t) \quad (117)$$

6.3. Kálmán-szűrő tervezése

Mivel az IMU-t egy mozgó tárgynak tekinthetjük, emiatt a rendszer állapotai a távolság d_k és a sebesség v_k . A rendszer bemenete a gyorsulás a_k és kimenete y_k , ami a mért távolság és sebesség. Az alap fizikai törvények alapján a sebességet v_k a (118) egyenlettel lehet leírni [13].

$$v_k = v_{k-1} + \Delta t(a_{k-1}) \quad (118)$$

A pillanatnyi sebesség értéke egyenlő az egy periódussal ezelőtt mért sebesség értékével, amihez hozzáadjuk az egy periódussal ezelőtt mért gyorsulás és az idő különbség szorzatát. A távolságot a (119) egyenlettel lehet leírni.

$$d_k = d_{k-1} + \Delta t v_k + \frac{1}{2} \Delta t^2 a_{k-1} \quad (119)$$

Az állapot vektort a következőképpen definiáltam.

$$x_k = \begin{pmatrix} d_k \\ v_k \\ a_k \end{pmatrix} \quad (120)$$

Az (118) és (119) egyenletek alapján az állapotteret modellt a következőképpen lehet felírni.

$$\begin{pmatrix} d_k \\ v_k \\ a_k \end{pmatrix} = \begin{pmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} d_{k-1} \\ v_{k-1} \\ a_{k-1} \end{pmatrix} + \begin{pmatrix} \frac{\Delta t^2}{2} \\ \Delta t \\ 1 \end{pmatrix} a_k + w_k \quad (121)$$

A kimenetet a következőképpen lehet felírni.

$$y_k = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} d_k \\ v_k \\ a_k \end{pmatrix} + v_k \quad (122)$$

A rendszer zaj az IMU-hoz kapcsolódó zajnak felel meg. Magába foglalja az IMU kimeneti zaját és annak offset értékét. A Δt megfelel két mérés közötti eltelt időnek.

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (123)$$

$$y(t) = Cx(t) + Du(t) \quad (124)$$

$$x(t_0) = x_0 \quad (125)$$

Állapot-átmeneti mátrix (A)	$\begin{pmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{pmatrix}$
Belsőállapotok vektora (x)	$\begin{pmatrix} d_k \\ v_k \\ a_k \end{pmatrix}$
Belsőállapotok vektorának inicializálása (x_0)	$\begin{pmatrix} d_0 \\ v_0 \\ a_0 \end{pmatrix}$
Mérési mátrix (H)	$\begin{pmatrix} 0 & 0 & 1 \end{pmatrix}$
Kovariancia mátrix P_0 , Mérési mátrix R, Folyamat zaj Q	3 x 3 Mátrix (adott)

táblázat 2. Kálmán-szűrő algoritmusában használt rendszermodell

A fentiek alapján Matlab környezetben implementáltam a Kálmán-szűrőt[10]. Az IMU-val kétféle mérést végeztem.

6.4. Szögek kiszámítása

A bicopter szabályozásához és irányításához elengedhetetlen, hogy tudjuk milyen dőlés-, billenő- és irányyszöggel rendelkezik. Ennek a kiszámítására az IMU szenzor adatait használom fel. Mint ahogy a 6.1-es fejezetben is említettem, az IMU szenzorom a szögsebességet téríti vissza másodpercenként. A beolvasott adatokat mintavételezem. A mintavételezési frekvenciám 250 Hz-re választottam, mivel a YMFC-3D repülésirányító is ezt használja. 2000 mintát veszek, ezekből a mintákból középértéket számolok (Kódrészlet 1) és minden egyes újonnan beolvasott értékből kivonom a középértéket (Kódrészlet 2), ezzel megkapva a kalibrált szöget. Fizikailag ez felel meg szög referenciának [14].

```
ang_acc_x_cal /= 2000;
ang_acc_y_cal /= 2000;
ang_acc_z_cal /= 2000;
```

Kódrészlet 1

```
ang_acc_x_offset = ang_acc_x - ang_acc_x_cal;
ang_acc_y_offset = ang_acc_y - ang_acc_y_cal;
ang_acc_z_offset = ang_acc_z - ang_acc_z_cal;
```

Kódrészlet 2

A szögsebességet megszorozva a periódussal megkapom a szög értéket. Mivel a mintavételezési frekvenciám 250Hz, emiatt a periódusom 0.004 szekundum (Kódrészlet 3).

```
ang_pitch += ang_x_filter * 0.004;
ang_roll += ang_y_filter * 0.004;
ang_yaw += ang_z_filter * 0.004;
```

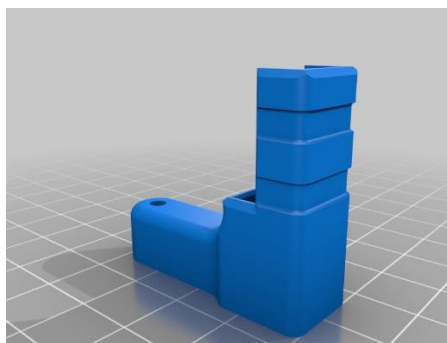
Kódrészlet 3

7. Gyakorlati kivitelezés

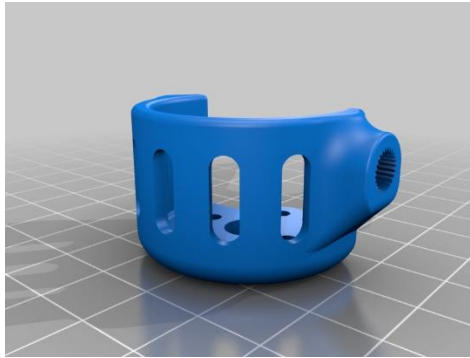
A célom a bicopter fizikai megvalósításának a tervezése és kivitelezése. Meg kell oldanom a mechanikai elemek létrehozását, összeszerelését.

7.1. Mechanikai elemek

A gyors haladás érdekében a mechanikai elemeket 3D nyomtató segítségével gyártottam le. Szükségem volt két motortartóra, ez a 35-es ábrán látható. A motor tartót össze kell kötni egy szervóval. A szervó tengelye egyenesen a motortartóba illeszkedik bele. Azért, hogy ne legyen túlságosan nagy terhelés ezen a tengelyen a bicopter két végére egy támasztó elemet rögzítettem, lásd a 34-os ábrát [15].

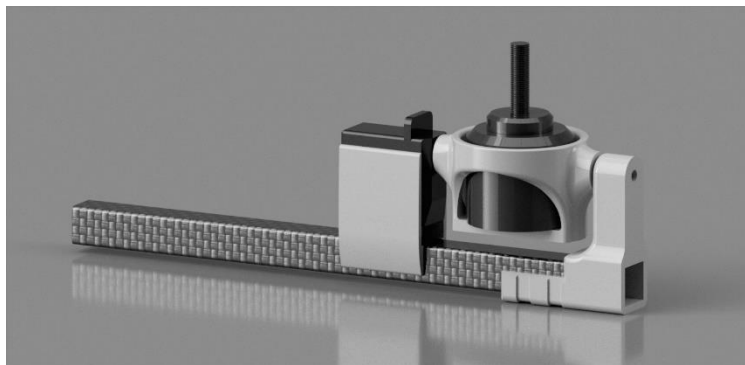


ábra 34. Tengelyrögzítő elem.



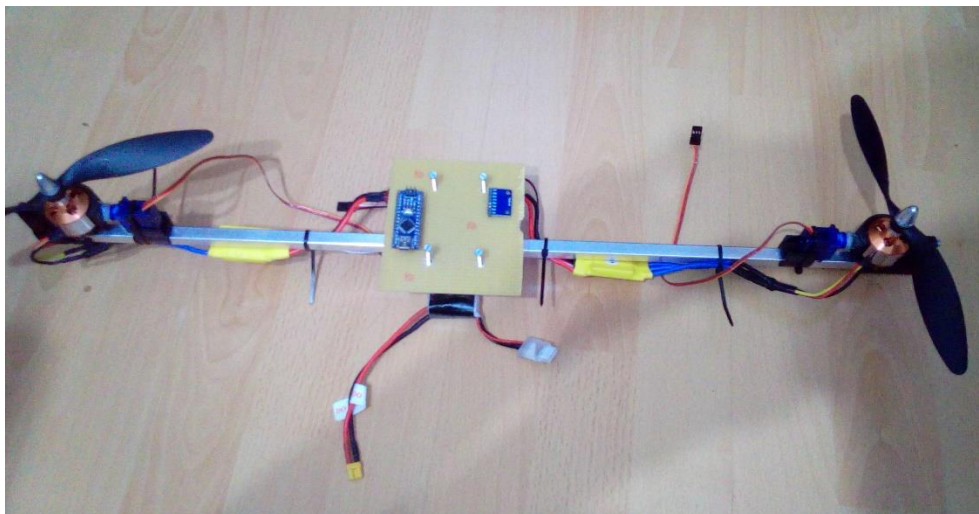
ábra 35. Motortartó elem

Az elemek összeszerelve a következő módon néznek ki (36 ábra).

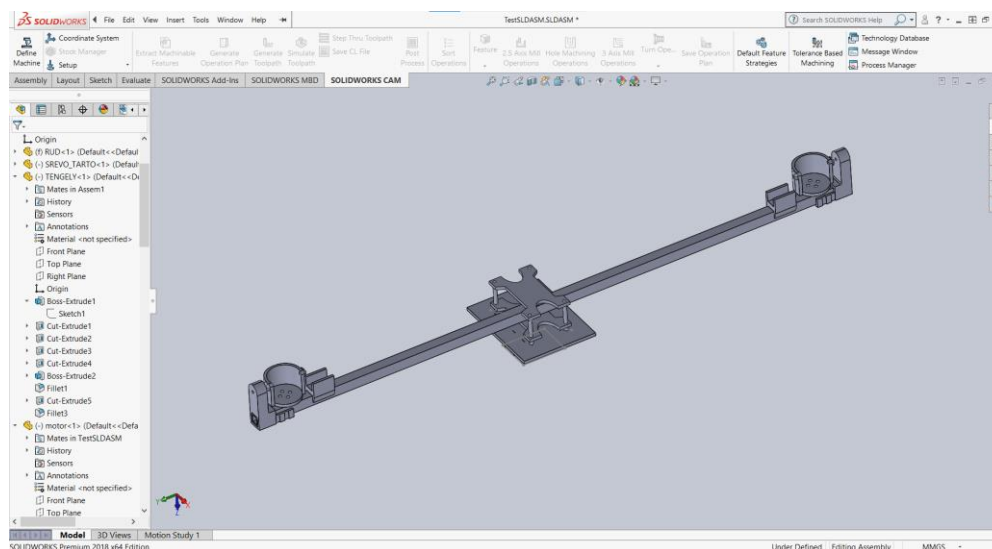


ábra 36. Billenést megvalósító mechanizmus

A bicopter teljesen összeszerelt formája a 37-as ábrán látható. A bicopter minden egyes alegysége le volt tesztelve.



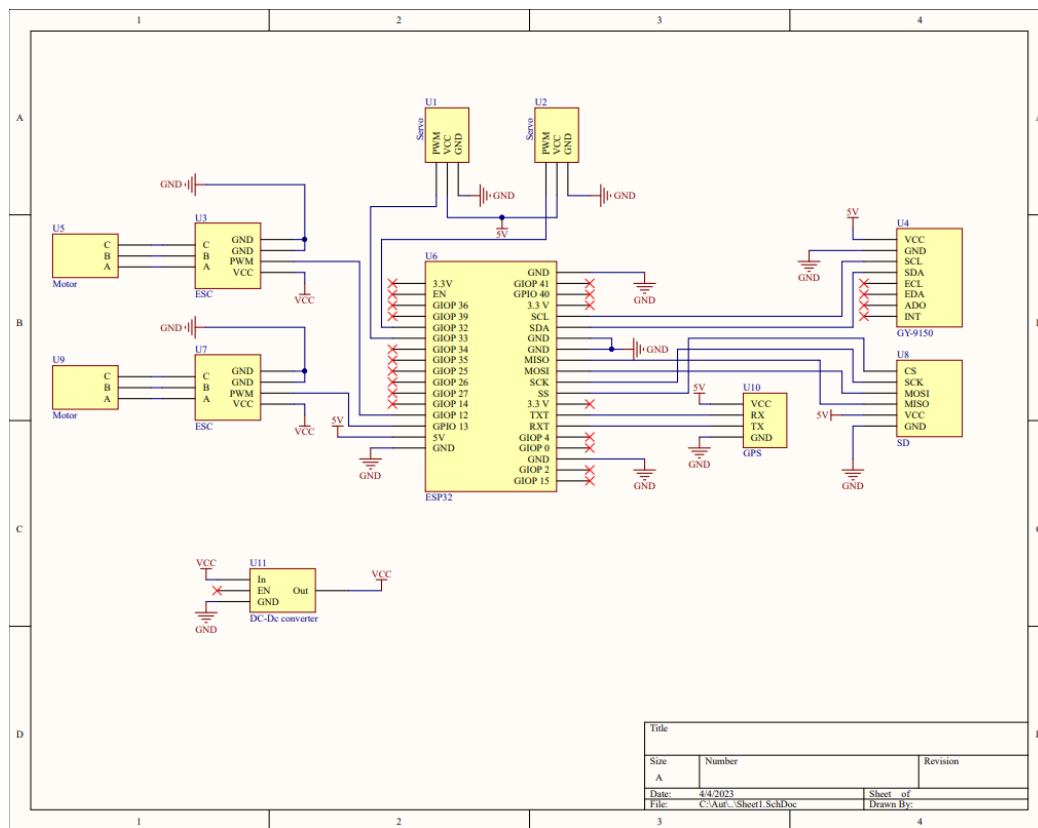
ábra 37. Bicopter



ábra 38. Bicopter vázszerkezete

7.2. A Bicopter kapcsolási rajz

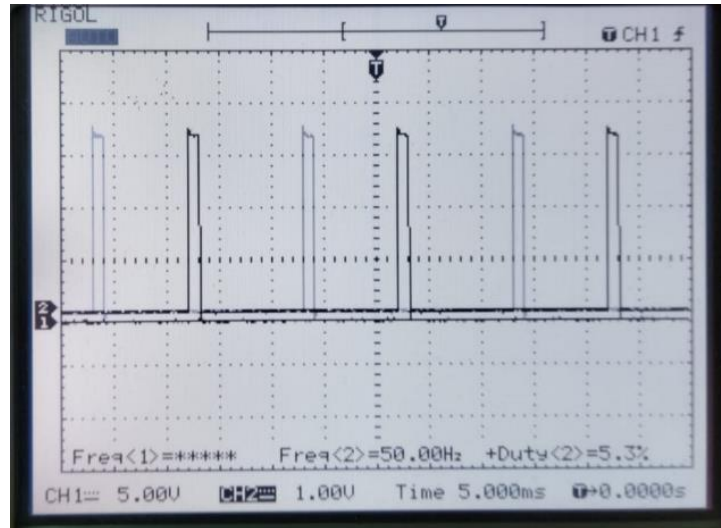
A kapcsolási rajzot altium designer program segítségével terveztem. Bekötöttem néhány olyan elemet is, ami a továbbfejlesztés során szeretnék használni. Ilyen például az SD kártya modul, amit az adatok mentésére fogok felhasználni és a GPS modul, amit szenzor fúzióként fogok felhasználni az IMU szenzorral együtt (39 ábra). A tervezési szempont az volt, hogy a PCB-n levő eszközök modulárisan cserélhetőek legyenek. Ezért minden egyes eszközt túsoron keresztül lehet csatolni.



ábra 39. Kapcsolási rajz

7.3. ESC (Electronic Speed Control) vezérlése PWM-en keresztül

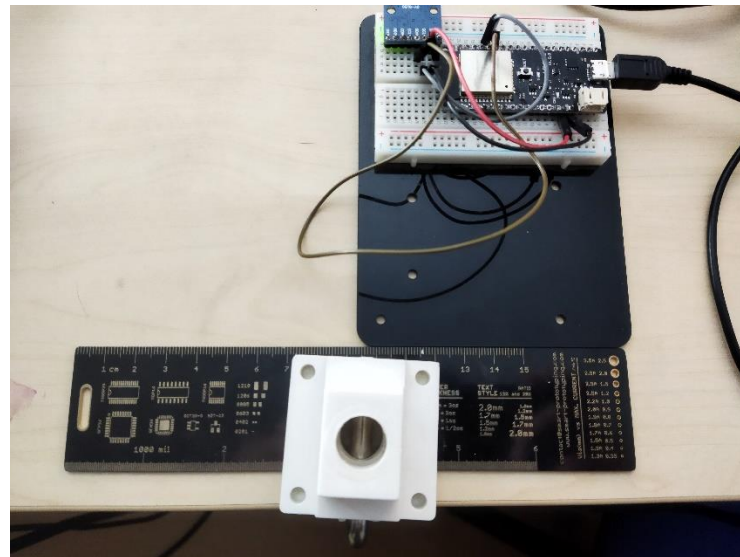
A motor fordulatszámát ESC-vel lehet vezérelni. Impulzusszélesség-modulált (PWM) jelekkel szabályozható az ESC, jelezve a kívánt fordulatszám értéket 0 és 100% között. A PWM az inerciális elektromos eszközök szabályozására elterjedten alkalmazott módszer. Tulajdonképpen, a PWM jel csak egy jelforma, amely magas és alacsony feszültség között periódikusan változik. Az ESC PWM frekvenciája 50 Hz, ami megfelel 20 ms-os periódusnak. Ha egy 1ms-os kitöltést adok az ESC-nek, akkor a legkisebb fordulaton fog forogni a motor. 2 ms-os jelnél maximum fordulaton fog forogni. A valóságban a két motor különböző kitöltésen kezdett el forogni. Ezeket oszcilloszkóp segítségével mértem le az egyik 1.06 ms-nál elkezd forogni a második motor pedig 1.26 ms-nál, lásd a (40)-es ábrát.



ábra 40. ESC motorvezérlő 1.06 ms jel

8. Tesztelés

Az IMU tesztelése során felállítottam egy mérőstandot, ami nem állt másból, mint egy vonalzóból és egy ezt rögzítő elemből, lásd a (41)-es ábrát. Ezen a rendszeren csúsztatva az IMU-t, mértem az elmozdulást. A szög mérésénél pedig, kalibrálás után, elforgattam 90 fokot.

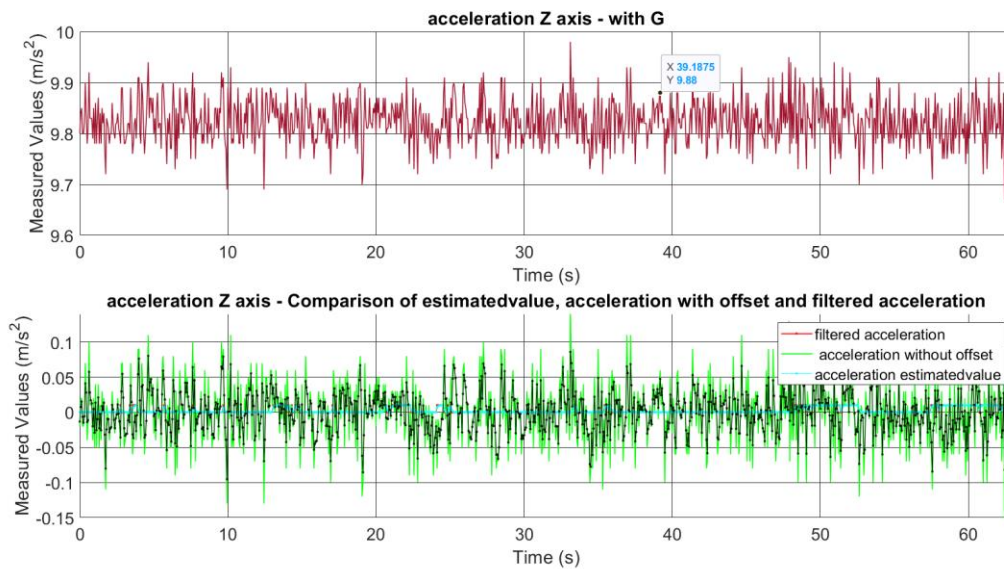


ábra 41. Tesztstand

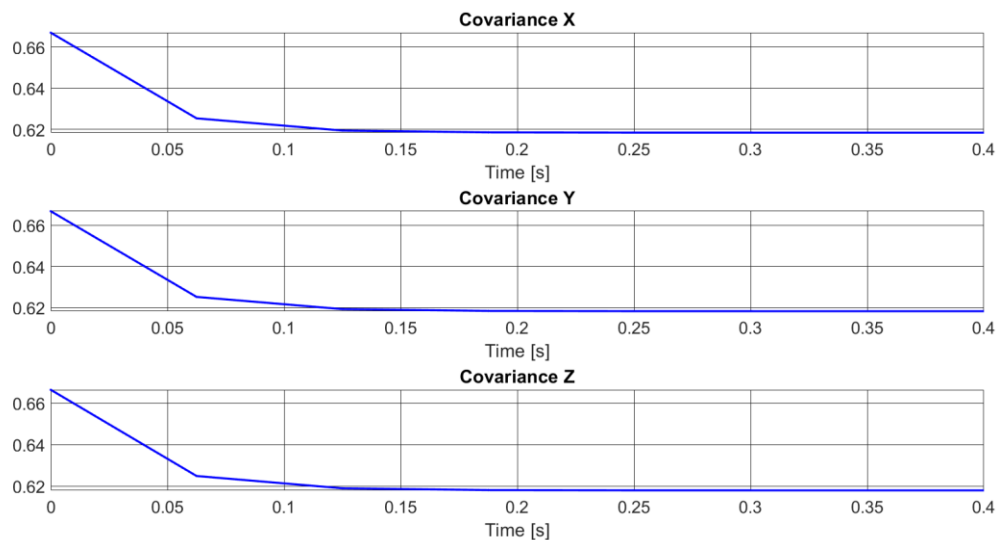
8.1. Kálmán-szűrő tesztelése

Ahhoz, hogy tesztelni tudjam a Kálmán-szűrőt, két külön mérést végeztem a rendszeren. Az első esetben elhelyeztem a szenzort egy viszonylag vízszintes felületen (asztalon). Kalibráltam a szenzort és utána egy kb. egy percen keresztül lementettem a gyorsulás értékeket

x,y,z tengely mentén. A lementett adatokat bevitettem Matlab környezetbe, ahol a Kálmán-szűrő segítségével megbecsültem és szűrtem a rendszer gyorsulását, sebességét és elmozdulását.



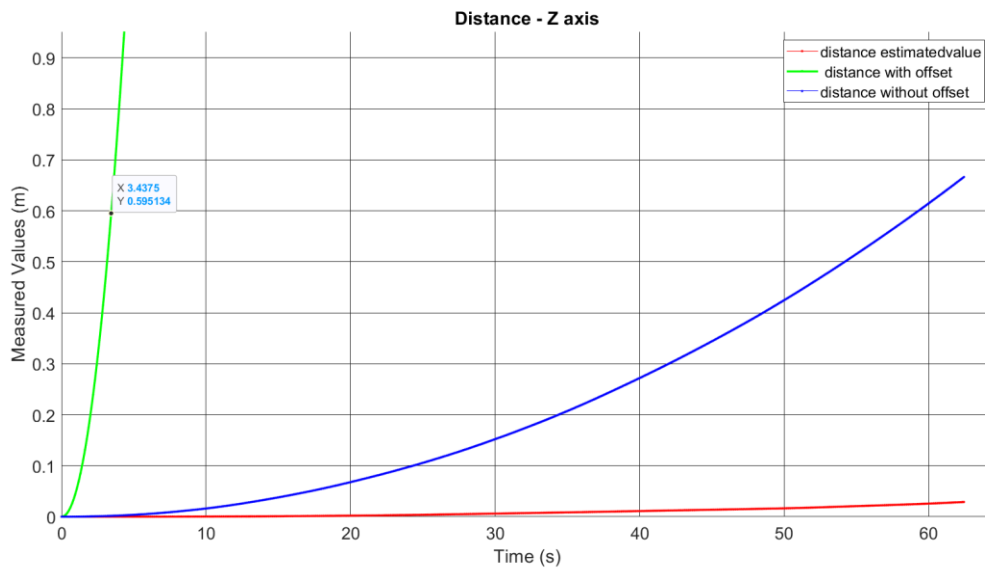
ábra 42. Kálmán-szűrő - Z tengely menti gyorsulás



ábra 43. Variancia változása x, y, z tengelyekre

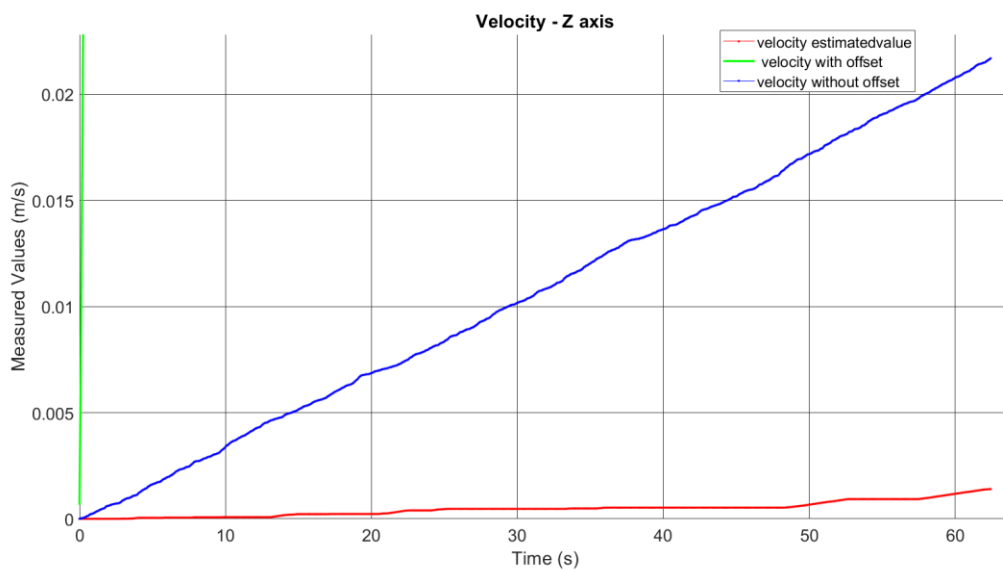
A 42-es ábrán látható a Z tengely mentén fellépő gyorsulás. A 42-es ábra felső részénél kalibráció nélkül látható a mérés. Miután a középérték kiszámításával kalibráltam a szenzort a 39-es ábra alsó részén levő zöld grafikont kaptam. Láthatom, hogy annak ellenére, hogy nem mozgatom z tengelyen, mégis fellép egy $\pm 0.1 \text{ m}^2/\text{s}$ -os gyorsulás. Ezt az értéket bevezettem a Kálmán-szűrőbe, lásd a fekete színű grafikont. Mivel nem lett sokkal jobb, ezért döntöttem

úgy, hogy használok egy szűrőt (6.2), amelyet kék színnel tüntettem fel az ábrán. Ezt a szűrt értéket vezettem be a Kálmán-szűrőbe. Az eredmény pirossal látható az ábrán.



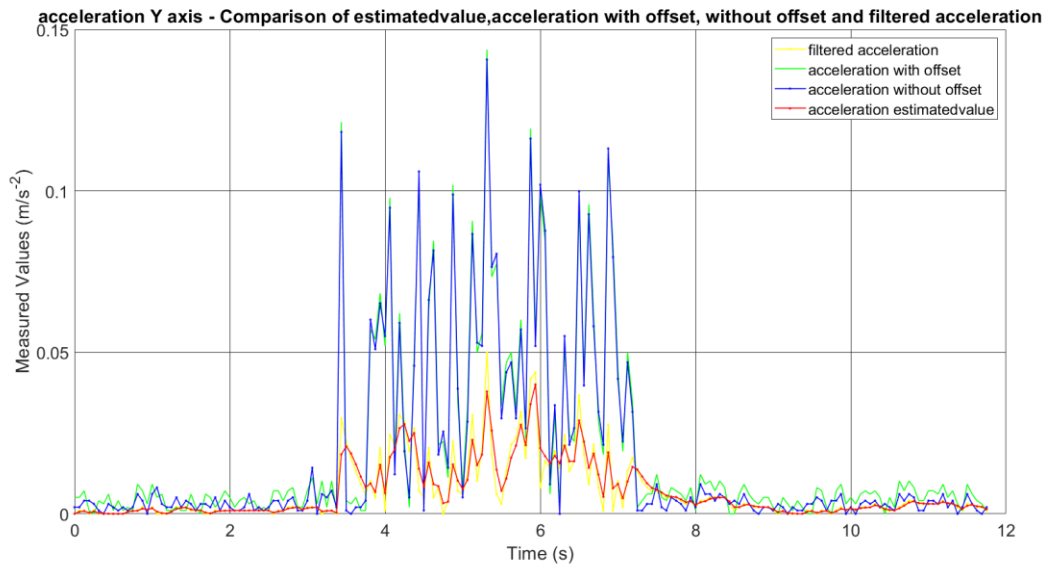
ábra 44. Kálmán-szűrő - Z tengely menti elmozdulás

A 44-es ábrán látható a rendszer által megtett távolság. Zölddel a gravitációs gyorsulás miatt látszik, hogy elintegrálódik a megtett távolság. Miután kalibráltam a szenzort azt tudom leolvasni az ábráról, hogy 0.68 méteres volt az elmozdulás kicsit több mint egy perc alatt. Ezután szűrtem (6.2) az értéket és úgy kaptam egy 0.1m alatti elmozdulást. Viszonylag jól megközelíti a valódi elmozdulást. A 4-ös ábrán látható ugyanebben a sorrendben ábrázolt sebessége a rendszernek.

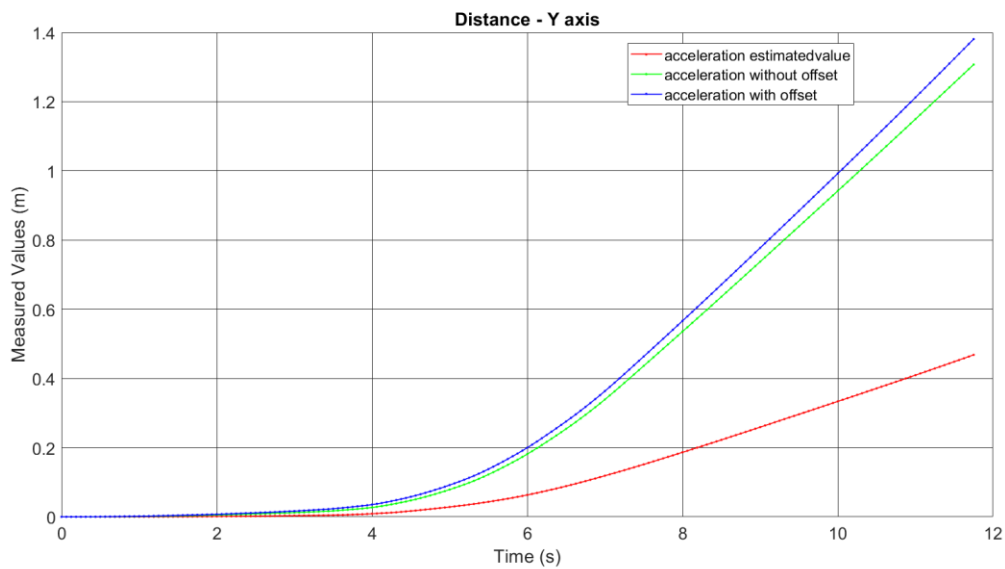


ábra 45. Kálmán-szűrő - Z tengely menti sebesség

A második esetben elmozdítottam a szenzort az y tengelyen 15cm-t, látható a 46-as ábrán, hogy észleli a gyorsulást, mikor elmozdítottam. Ha megnézzük a 47-es ábrán látható a kék grafikonon, hogy 15cm helyett 45cm-es elmozdulást mértem, ami kis elmozdulásokra jól megközelíti a valóságot.



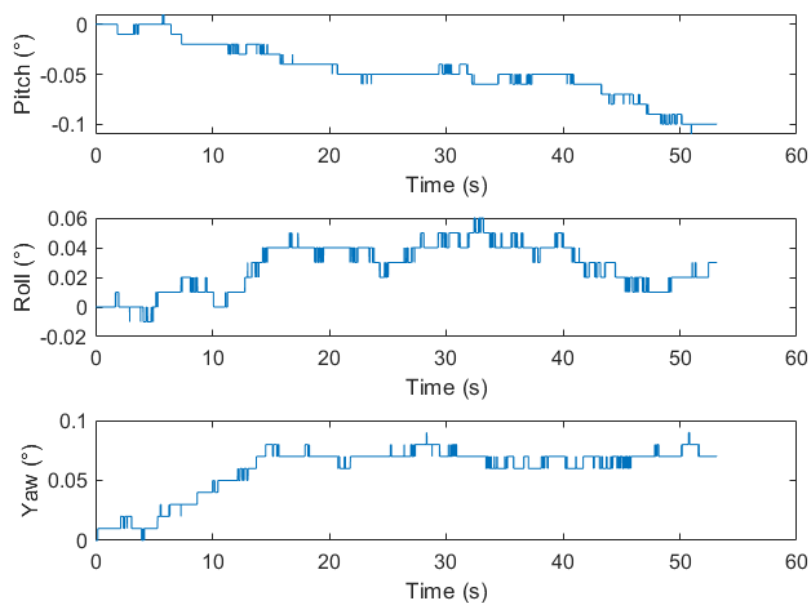
ábra 46. Kálmán-szűrő - Y tengely menti gyorsulás



ábra 47. Y tengely menti elmozdulás, amit Kálmán-szűrővel kaptunk

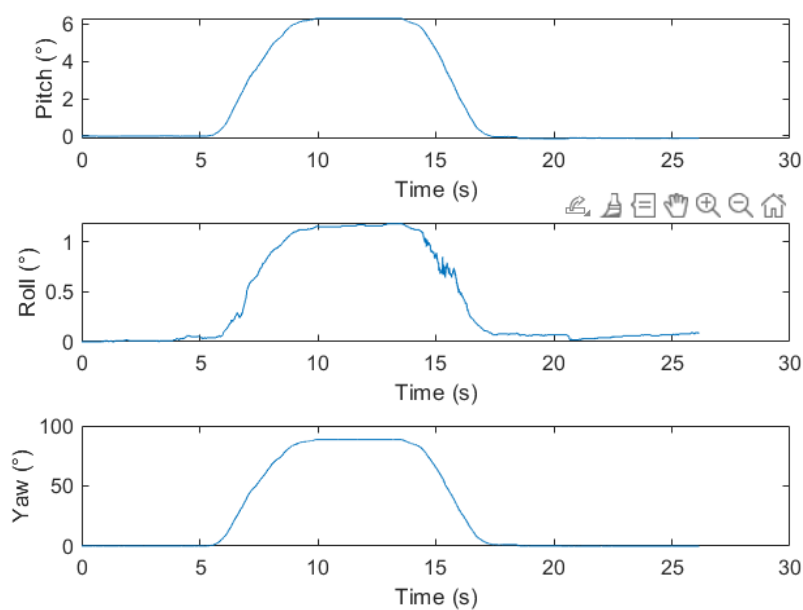
8.2. Szög mérése integrálással

Miután implementáltam a szög méréséhez szükséges elveket, lásd a 6.4-es fejezetben, az IMU-rendszeren két mérést végeztem. Az első mérést nyugalmi állapotban mértem, lásd a 48-as ábrát. Az ábrán megfigyelhető egy kis eltolódás nullához képest.



ábra 48. Nyugalmi állapotbéli szög eltolódása

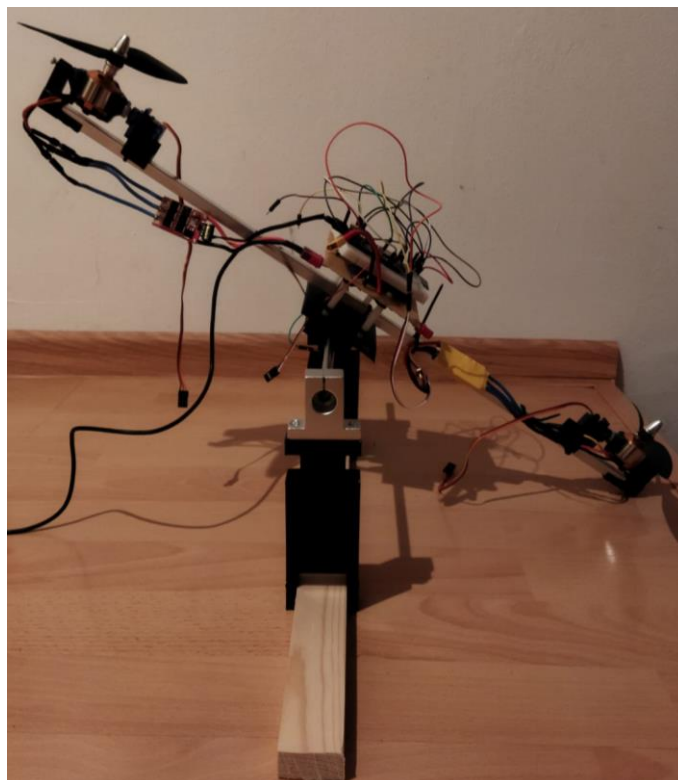
A második mérésnél a rendszernek az elfordulás (Yaw) szögét elforgattam 90 fokot, egy ideig tartottam az értéket és utána visszavittem 0 fokra, lásd a 46-os ábrát. Látható, hogy a billenés és a bólintás szögem is elmozdult. Ez annak tudható be, hogy nem vízszintes a felület, amire helyezve volt a rendszer. A 90 fokos forgatást 88.63 foknak észlelte a rendszer.



ábra 46. 90-fokos elmozdulás

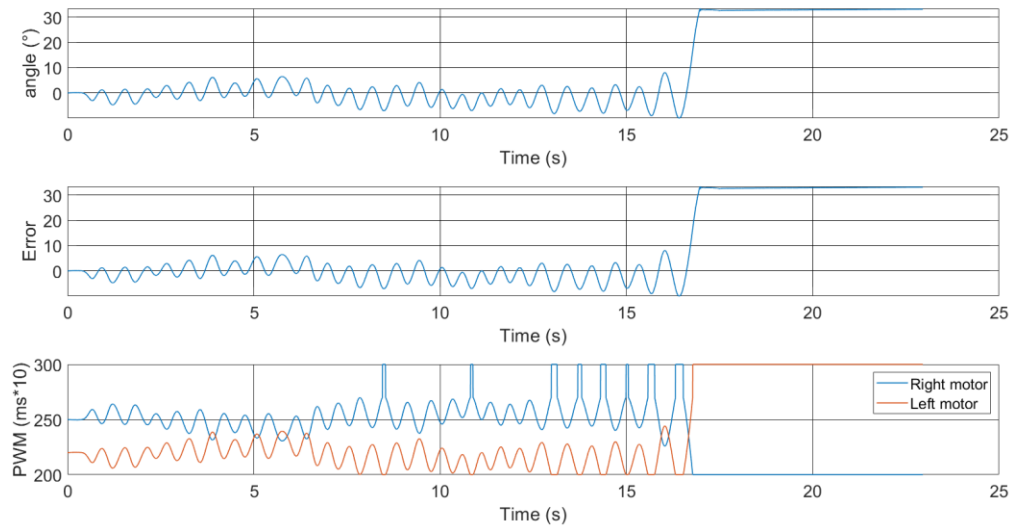
8.3. Φ szög körüli egyensúlyozás

Egy proporcionális ($P = 3$) taggal rendelkező szabályozó segítségével $\phi = 0$ referencia szöget szerettem volna tartani, ezt a szöget a 8.2-es alfejezet alapján határoztam meg. Ehhez a méréshez egy mérő standot hoztam létre, lásd (47)-es ábrát. A rendszer a referencia szöget az IMU kalibráció során a bicopter térbeli elhelyezkedéséhez rögzíti.



ábra 47. billenő rendszer

A (48)-as ábrán látható a mérések eredménye. A szög és az error egyenlő, mivel 0 referencia értéket adtam meg. Az is látható, hogy a két motor vezérlő jele különböző értékekről indul. Ez a két motor különbsége és az ESC motor vezérlők különbségéből adódik, lásd a 7.3-as alfejezetet. Látható az is, hogy a végén megszakítom a kimeneti vezérlést, emiatt leesik az egyik felére a rendszer. A vezérlő korlátozva van.



ábra 48. billenő rendszer mérések

9. Következtetések

A dolgozatom a kétmotoros légi robot (bicopter) modellezése, tervezése, irányítása és kivitelezése köré épült. A munka során több problémával is találkoztam. Az interneten található szakirodalmakban a térbeli bicopter matematikai modellek levezetésében hibákat észleltem, amit újra levezettem. A Kálmán-szűrő által kivitelezett IMU nem bizonyult megfelelően pontosnak.

A felmerülő problémák ellenére a dolgozat többnyire elérte céljait.

9.1. Megvalósítások

A dolgozatom első célja egy kétmotoros légi robot (bicopter) 2D-s és 3D-s matematikai modellezése és szimulációja volt, amely beteljesült. A második cél, ami a szabályozási feladatok elméleti és gyakorlati vizsgálata volt, részben realizáltnak mondható. Az elméleti oldalát sikerült megvalósítani, azonban a gyakorlati oldala elmaradt. A dolgozat utolsó célja a robot gyakorlati megvalósítása volt. Sikerült összeszerelnem egy fizikai rendszert, aminek különálló részei működőképesek. Viszont együttesen a rendszer nem volt tesztelve. A rendszernek az inerciális és mágneses szenzora esetében fennálló zaj- és eltolás problémákat csökkentettem a Kálmán-szűrő segítségével.

9.2. További fejlesztési irányok

A dolgozatot több irányba is tovább lehet fejleszteni. A térbeli bicopter matematikai modelljének szimulációját ki lehet bővíteni giroszkópikus hatásokkal. Továbbá, lehetne egy pályakövető szabályozót tervezni. A szabályozó gyakorlati kivitelezése és tesztelése a rendszeren egy újabb fejlesztési irányt adhat. A biztonságos tesztelés érdekében ajánlott a tesztstand építése. Az IMU-t lehetne pontosítani egy Kálmán-szűrővel a szögek becslésére. A pozíció pontosítására lehetne szenzorfúziót használni egy GPS és egy barometrikus szenzor segítségével. A 7.2-es fejezeten szereplő schematik rajz alapján lehetne egy új PCB-t tervezni, ami rövidzár védelemmel rendelkezik.

10. Irodalomjegyzék

- [1] J. Abdulridha, Y. Ampatzidis, S. C. Kakarla, and P. Roberts, “Detection of target spot and bacterial spot diseases in tomato using UAV-based and benchtop-based hyperspectral imaging techniques,” *Precis Agric*, vol. 21, no. 5, pp. 955–978, Oct. 2020, doi: 10.1007/s11119-019-09703-4.
- [2] J. Abdulridha, Y. Ampatzidis, P. Roberts, and S. C. Kakarla, “Detecting powdery mildew disease in squash at different stages using UAV-based hyperspectral imaging and artificial intelligence,” *Biosyst Eng*, vol. 197, pp. 135–148, Sep. 2020, doi: 10.1016/j.biosystemseng.2020.07.001.
- [3] J. Abdulridha, Y. Ampatzidis, J. Qureshi, and P. Roberts, “Laboratory and UAV-Based Identification and Classification of Tomato Yellow Leaf Curl, Bacterial Spot, and Target Spot Diseases in Tomato Utilizing Hyperspectral Imaging and Machine Learning,” *Remote Sens (Basel)*, vol. 12, no. 17, p. 2732, Aug. 2020, doi: 10.3390/rs12172732.
- [4] S. Salazar, H. Romero, R. Lozano, and P. Castillo, “Modeling and Real-Time Stabilization of an Aircraft Having Eight Rotors,” in *Unmanned Aircraft Systems*, Springer Netherlands, 2008, pp. 455–470. doi: 10.1007/978-1-4020-9137-7_24.
- [5] L. Ambroziak *et al.*, “Experimental tests of hybrid VTOL unmanned aerial vehicle designed for surveillance missions and operations in maritime conditions from ship-based helipads,” *J Field Robot*, vol. 39, no. 3, pp. 203–217, May 2022, doi: 10.1002/rob.22046.

- [6] S. C. Kakarla and Y. Ampatzidis, “Types of Unmanned Aerial Vehicles (UAVs), Sensing Technologies, and Software for Agricultural Applications,” *EDIS*, vol. 2021, no. 5, Oct. 2021, doi: 10.32473/EDIS-AE565-2021.
- [7] “The Best New Quad Is A Bicopter | Hackaday.”
<https://hackaday.com/2018/08/27/the-best-new-quad-is-a-bicopter/> (accessed Apr. 12, 2023).
- [8] “Bicopter - The journey so far - Blog - ArduPilot Discourse.”
<https://discuss.ardupilot.org/t/bicopter-the-journey-so-far/59328> (accessed Apr. 12, 2023).
- [9] “Stability and control of VTOL tilt-rotor bi-copter UAVs.”
<http://reports.ias.ac.in/report/21006/stability-and-control-of-vtol-tilt-rotor-bi-copter-uavs> (accessed Apr. 08, 2023).
- [10] P. P. Naveen, “Filtering of IMU data using Kalman Filter,” 2016.
- [11] Q. Zhang, Z. Liu, J. Zhao, and S. Zhang, “Modeling and attitude control of Bi-copter,” in *AUS 2016 - 2016 IEEE/CSAA International Conference on Aircraft Utility Systems*, Institute of Electrical and Electronics Engineers Inc., Nov. 2016, pp. 172–176. doi: 10.1109/AUS.2016.7748042.
- [12] “MPU-9150 Product Specification Revision 4.3 MPU-9150 Product Specification”.
- [13] “Kalman Filter.”
https://ece.montana.edu/seniordesign/archive/SP14/UnderwaterNavigation/kalman_filter.html (accessed Apr. 12, 2023).
- [14] “Carbon aeronautics quadcopter build and programming manual electronics aeronautics programming.”

[15] “RCExplorer Tilt Mechanism by rsheldiii - Thingiverse.”

<https://www.thingiverse.com/thing:4086482> (accessed Apr. 12, 2023).