**Hexapod robot**

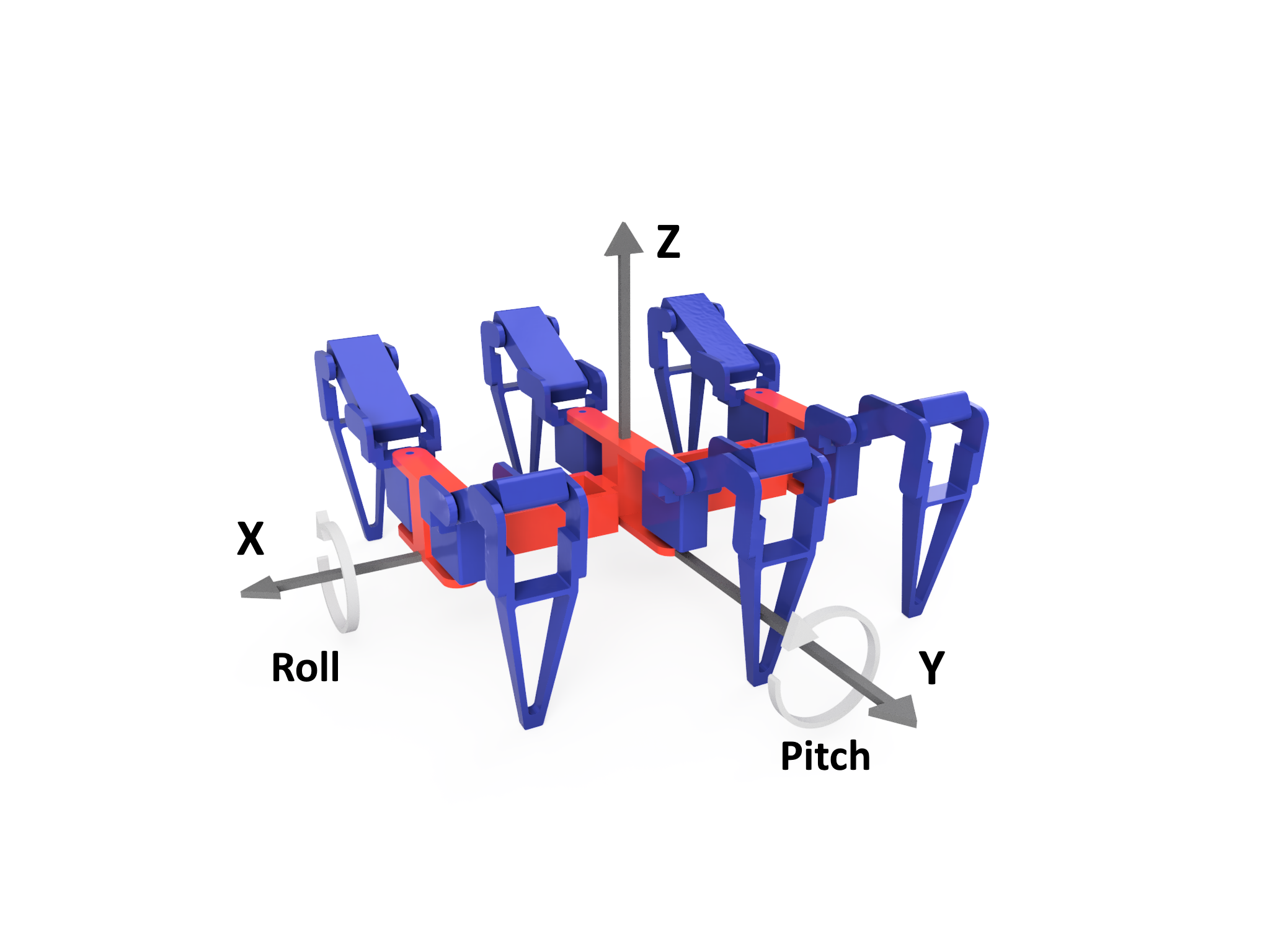
**Dokumentacija**

**Funkcije robota**

**Definirane funkcije:**

* **initializeRobot() –** inicijalizacija varijabli robota
* **robotStand() –** postavljanje robota u početnu poziciju
* **moveAtAngle(String move\_object, int n\_object, double n\_angle, double n\_stride, int n\_delay) –** kretanje robota u bilo kojem smjeru
* **turnRobot(String side, double n\_stride, int n\_delay) –** okretanje robota oko z osi
* **moveLegZTo(double z, int legNum) –** podizanje ili spuštanje noge na određenu visinu
* **robotToZ(double z) –** podizanje ili spuštanje robota na određenu visinu
* **angleBase(double n\_pitch, double n\_roll, double n\_mid\_height) –** postavljanje nagiba baze
* **angleBaseOverTime(double n\_pitch, double n\_roll, double n\_mid\_height, double n\_time) –** postavljanje nagiba baze kroz vrijeme
* **gesture\_wave() –** robot izvršava gestu „mahanja“

**Koordinatni sustav robota:**



**NAPOMENA: koordinatni sustav je nepomičan u odnosu na tijelo robota!**

**Inicijalizacija robota**

Za početnu inicijalizaciju varijabli potrebnih za rad robota, koristimo funkciju ***initializeRobot()***.

Ovu funkciju pozivamo u ***setup()*** dijelu *Arduino* programa te ju je potrebno pozvati samo jedanput.

Sintaksa za poziv ove funkcije je:

***initializeRobot()***

Funkcija ne traži unos dodatnih varijabli.

**Postavljanje robota u početnu poziciju**

Za postavljanje robota u početnu poziciju definirana je funkcija ***robotStand()***.

Sintaksa za pozivanje ove funkcije glasi:

***robotStand()***

Funkcija ne traži unos dodatnih varijabli.

**Okretanje robota**

Za okretanje cijelog robota oko z osi koristimo funkcije ***turnRobot()***.

Pozivom na ovu funkciju robot izvršava postepeno okretanje kroz nekoliko koraka. Pozivanje funkcije se mora ponavljati sve dok robot ne dođe u željenu poziciju.

Sintaksa za pozivanje ove funkcije glasi:

***turnRobot( String side , double n\_stride , int n\_delay )***

**side** – definira stranu u koju želimo da se robot okreće. Vrijednosti upisa su „left“ i „right“. Upisom vrijednosti „left“ robot se okreće ulijevo. Upisom vrijednosti „right“ robot se okreće udesno.

**n\_stride** – definira duljinu „koraka“ tijekom okretanja robota. Vrijednost se upisuje u milimetrima te utječe na brzinu okretanja robota.

**n\_delay** – definira pauze između izvođenja operacija u ciklusu okretanja. Utječe na brzinu izvođenja naredbe i cjelokupnog ciklusa okretanja.

**Primjer korištenja funkcije *turnRobot()*:**

* turnRobot( „left“ , 40 , 500);
* turnRobot( „right“ , 25 , 600);

**Kretanje robota**

Za kretanje robota u bilo kojem smjeru, koristimo funkciju ***moveAtAngle()***.

***moveAtAngle()*** ima sposobnost pomicanja robota u svim smjerovima, bez promjene orijentacije robota.

Sintaksa za pozivanje ove funkcije glasi:

***moveAtAngle( String move\_object, int n\_object, double n\_angle, double n\_stride, int n\_delay )***

**move\_object** – definira koji dio robota pokrećemo. Može biti „*leg*“, „*leg\_set*“ ili „*robot*“. Korištenjem objekta „*leg*“ pomičemo samo jednu nogu, korištenjem objekta „*leg\_set*“ pomičemo odabrani set od triju nogu, a najčešće korišteni objekt, „*robot*“ pomiče cijelog robota u željenom smjeru.

**n\_object** – definira segment objekta kretanja. Ako je move\_object postavljen u vrijednost „*leg*“, ovime odabiremo broj noge koju pomičemo(0 – 5). Ako je move\_object postavljen u vrijednost „*leg\_set*“, ovime odabiremo koji set pomičemo. Dostupni setovi nogu su 1 i 2. Set 1 sastoji se od nogu 0, 3, 4 dok se set 2 sastoji od nogu 1, 2, 5. Kod vrijednosti „*robot*“ ova varijabla nema nikakvu funkciju ali mora biti zadana (postavljena barem u 0).

**n\_angle** – definira smjer kretanja robota. Smjer definiramo kutom zadanim u stupnjevima u rasponu od -360 do 360. Kada je kut postavljen u vrijednost 0 robot se kreće ravno naprijed. Kut 180 odgovara kretanju unatrag. Kut -180 isto odgovara kretanju unatrag.

**n\_stride** – definira duljinu jednog „koraka“ robota(put koji svaka noga prijeđe na tlu pri jednom ponavljanju ove funkcije). Manja vrijednost daje nam precizniji i „mirniji“ hod.

**n\_delay** – definira pauzu između pojedinog koraka. Manja pauza rezultira bržim ciklusom izvršenja koda čime ostvarujemo brži hod robota.

**Primjer korištenja funkcije *moveAtAngle()*:**

moveAtAngle( "robot" , 0 , 180 , 20 , 400 );

**Pomicanje noge na određenu visinu**

Funkcija koja nam omogućava podizanje ili spuštanje noge na željenu visinu je ***moveLegZTo()***.

Poziv ove funkcije podiže nogu u visinu po Z osi te ostale osi ostavlja nepromijenjene.

Sintaksa za pozivanje ove funkcije glasi:

***moveLegZTo( double z, int legNum )***

**z** – definira novu željenu visinu noge.

**legNum** – definira koju nogu želimo pomaknuti. Noge se nalaze pod brojevima od 0 do 5.

**Primjer korištenja funkcije *moveLegZTo()*:**

* moveLegZTo( -30, 3 );
* moveLegZTo( -45, 0 );

**Pomicanje robota na određenu visinu**

Funkcija koja nam omogućava pomicanje robota po z osi je ***robotToZ().***

Pozivom ove funkcije postavljamo robota na željenu visinu.

Sintaksa za pozivanje ove funkcije glasi:

***robotToZ( double z )***

**z** – definira novu željenu visinu robota.

**Primjer korištenja funkcije *robotToZ()*:**

* robotToZ( -50 );
* robotToZ( -45 );

**Postavljanje nagiba baze robota**

Za postavljanje nagiba baze robota koristimo funkcije ***angleBase()*** i ***angleBaseOverTime().***

***angleBase()*** mijenja kut nagiba baze robota i izvršava se u najkraćem mogućem vremenu te se to vrijeme ne može mijenjati. Ako želimo promjenu nagiba baze kroz određeni vremenski interval koristimo funkciju ***angleBaseOverTime().***

Sintaksa za pozivanje ovih funkcija:

***angleBase( double n\_pitch, double n\_roll, double n\_mid\_height )***

***angleBaseOverTime( double n\_pitch, double n\_roll, double n\_mid\_height, double n\_time )***

**n\_pitch** – definira nagib u „pitch“ osi rotacije (nagib naprijed/natrag). Nagib je definiran kutom zadanim u stupnjevima od -45 do 45 (moguće je definirati veće ili manje vrijednosti od navedenih ali robot nema sposobnosti izvršiti takav zadatak zbog svoje konstrukcije).

**n\_roll** – definira nagib u „roll“ osi rotacije (nagib lijevo/desno). Nagib je definiran kutom zadanim u stupnjevima od -45 do 45 (moguće je definirati veće ili manje vrijednosti od navedenih ali robot nema sposobnosti izvršiti takav zadatak zbog svoje konstrukcije).

**n\_mid\_height** – definira visinu na kojoj želimo da se nalazi dno sredine baze robota.

**n\_time** – definira vrijeme koje je robotu potrebnu da iz trenutne pozicije dođe u željeni nagib. Vrijednost je definirana u milisekundama.

**Primjer korištenja funkcija** ***angleBase()*** **i *angleBaseOverTime()*:**

* angleBase( -10 , 8 , -60 );
* angleBaseOverTime( 15 , 0 , -70 , 1000 );
* angleBaseOverTime( 0 , -10 , -50 , 1000 );

**Gesta „mahanja“**

Za izvršavanje geste „mahanja“ koristimo funkciju ***gesture\_wave()***.

Pozivom na ovu funkciju robot se postavlja u poziciju gdje se balansira na stražnje četiri noge te podiže prve dvije i izvršava gestu „mahanja“.

Sintaksa za pozivanje ove funkcije glasi:

***gesture\_wave()***

Funkcija ne traži unos dodatnih varijabli.