

Artur ZAWADA<sup>1</sup>

Opiekun naukowy: Jacek RYSIŃSKI<sup>2</sup>

## ROBOT HUMANOIDALNY – PIERWSZE KROKI

**Streszczenie:** W artykule przedstawiono podejście do modelowania układu ruchu robota humanoidalnego. Obecnie budowa takiej konstrukcji nie stanowi już tak wielkiego wyzwania jak kiedyś. Dostępność gotowych podzespołów oraz urządzeń do druku 3D sprawia, że każdy może zbudować unikatową konstrukcję. Niestety wprowadzenie robota w ruch tak, aby poruszał się nie tracąc równowagi nie jest już tak proste i często z tego powodu projekt zostaje porzucony. W artykule przedstawiono na co należy zwracać uwagę podczas projektowaniu nóg robota. Porównano różne konfiguracje oraz przedstawiono ich możliwości. Przedstawiono koncepcję programowania trajektorii ruchu robota oraz opisano jak najprościej sterować ruchem jego nóg.

**Słowa kluczowe:** humanoid, pierwsze kroki, projekt

## THE FIRST STEPS OF A HUMANOID ROBOT

**Summary:** The article presents an approach to modeling the humanoid motion system. Nowadays, building your humanoid construction is not as big a challenge as it used to be, the availability of ready-made components and 3D printing means that anyone can build an almost unique structure. Unfortunately, putting the robot in motion without losing the balance is more complicated, and the project is often abandoned for this reason. The article presents what you should pay attention to when designing the robot's legs. Various configurations were compared, and their capabilities were presented. The concept of programming the driving trajectory is presented and the easiest way to control the course of its legss.

**Keywords:** humanoid, first steps, walkthrough

### 1. Wprowadzenie

Czasy, w których roboty humanoidalne były jedynie telewizyjną fikcją są już dawno za nami. Stopniowo w naszym życiu pojawiają się nowe konstrukcje przypominające człowieka, lecz do tej pory ich budową i programowaniem zajmowała się wąska grupa

---

<sup>1</sup> inż., Akademia Techniczno-Humanistyczna w Bielsku-Białej, Wydział Budowy Maszyn i Informatyki, email: artur.zawada88@gmail.com

<sup>2</sup> dr. inż., Akademia Techniczno-Humanistyczna w Bielsku-Białej, Wydział Budowy Maszyn i Informatyki, email: jrysinski@ath.bielsko.pl

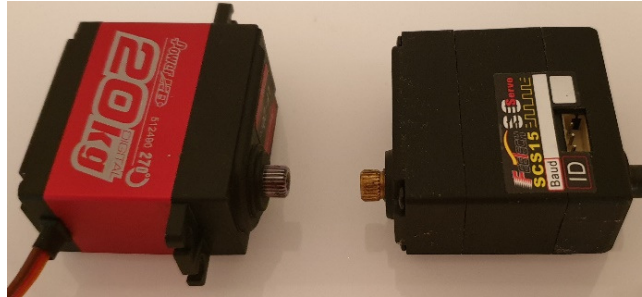
osób, które specjalizowały się w tej dziedzinie. Sam przygotowując się do tego artykułu przeczytałem wiele prac, w których opisano, jak powinien wyglądać poprawny ruch dwunożnego robota o ludzkiej strukturze – humanoida. W większości z tych prac przedstawiono sposób poruszania się oraz wyznaczaniu trajektorii ruchu wykorzystując metody „zero moment point” lub genetyczne. Prace te kierowane są do osób o wykształceniu technicznym, dla których pojęcia takie jak kinematyka prosta, odwrotna czy moment bezwładności nie stanowią problemu. Oczywiście nie jest tak bez powodu, gdyż bez tej wiedzy budowa robota dwunożnego jest wręcz nie możliwa. Nie mniej jednak te informacje nie są wystarczające do tego, aby bez wyznaczonej drogi samemu zaprojektować i wprowadzić w ruch robota dwunożnego. W artykule podjęto próbę przedstawienia kierunku rozpoczęcia prac tak, aby w możliwie najprostszy sposób dojść do etapu, w którym funkcjonalność wybranej na początku metody wyznaczania trajektorii będzie miała znaczenie. Aktualnie na rynku znajduje się szeroka gama produktów, które możemy zastosować w naszym projekcie. W poniższej pracy opisane zostaną podzespoły oraz oprogramowanie wykorzystane przy budowie robota „Achilles”.

## **2. Dobór podzespołów**

Przed przystąpieniem do wykonania projektu robota konieczne jest podjęcie decyzji, którą metodę wyznaczania trajektorii ruchu będziemy używać. Nie jest to trudny wybór jednak niesie za sobą pewne konsekwencje. Większość osób budujących swoją pierwszą konstrukcję wyobraża sobie robota poruszającego się jak człowiek, przez co wybierają metody genetyczne. W tym przypadku należy uwzględnić konieczną liczbę serwomechanizmów do realizacji rzeczywistego ruchu. Takie rozwiązanie jest poprawne jednak, aplikacja do sterowania jest bardzo skomplikowana. Innym rozwiązaniem są metody „zero moment point”, które ograniczają liczbę serwomechanizmów, jednak wymagają znajomości parametrów konstrukcyjnych takich jak np. masowe momenty bezwładności poszczególnych elementów. W poniższej publikacji wykorzystano właśnie drugą metodę, aby jak najbardziej uprościć konstrukcję.

### **2.1. Podzespoły mechaniczne**

Napęd jest bardzo ważnym elementem konstrukcji robota, ponieważ determinuje jego mobilność. Głównym komponentem odpowiadającym za płynność ruchu robota jest serwomechanizm. Wraz ze zwiększeniem gabarytów naszej konstrukcji rośnie jej masa, przez co musimy dobrać odpowiedni moment obrotowy serwomechanizmów. Z pomocą przychodzi nam szeroka gama gotowych produktów, które możemy zaimplementować niemalże składając robota z gotowych klocków. Drugim rozwiązaniem jest druk 3D, dzięki któremu możemy zaprojektować własną unikatową i często lżejszą konstrukcję. Rys. 1 przedstawia przykładowe serwomechanizmy typu standard stosowane w modelarskich robotach.



Rysunek 1. Serwomechanizm

Porównując je możemy zauważyć, że różnią się sposobem montażu samego serwomechanizmu jak i kolejnego członu, a także maksymalnym momentem, co będzie miało wpływ na dobór elementów konstrukcyjnych. Istnieje jeszcze jeden ważny aspekt wspólny dla większości serwomechanizmów odróżniający je od rzeczywistego stawu człowieka, jakim jest liczba stopni swobody. Przykładem jest staw biodrowy, w którym musielibyśmy umieścić trzy takie serwomechanizmy o przecinających się osiach obrotu tak aby stworzyć napędzane połączenie kulowe co jest fizycznie niewykonalne.

## 2.2. Podzespoły elektroniczne

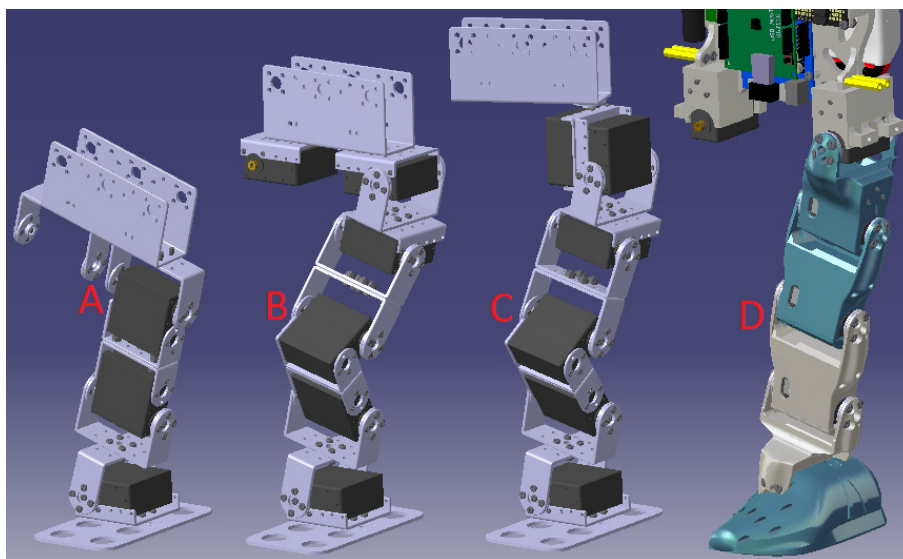
Dobór podzespołów elektronicznych często sprowadza się do wyboru technologii, której będziemy używać. Wynika to z faktu, iż musimy skomunikować wszystkie podzespoły ze sterownikiem, czyli powinny one obsługiwać te same standardy napięć oraz protokoły komunikacyjne. Przedstawione na Rys.1 serwomechanizmy różnią się pod względem napięcia zasilania. I tak, dla lewego wynosi do 6,4VDC natomiast dla prawego do 8,4VDC. Nie są to duże różnice jednak wpływają na dobór zasilania naszej konstrukcji. Dobierając akumulatory należy również uwzględnić ich wydajność prądową, a także pojemność, tak aby zapewnić możliwie jak najdłuższy czas pracy. Niestety pojemność wpływa na wagę samych pakietów, przez co często skracamy czas pracy na rzecz lżejszej konstrukcji. Opisane wcześniej serwomechanizmy komunikują się ze sterownikiem na logice 5VDC, przez co idealnie nadają się do małych i lekkich mikrokontrolerów takich jak Arduino, Rasbery PI czy STM32. Różnica polega na sposobie tej komunikacji, gdyż lewy serwomechanizm sterowany jest sygnałem PWM, przez co możemy jedynie wysłać do niego pozycje na jaką ma się ustawić, nie możemy odczytać jego aktualnej pozycji, a prędkość ruchu zawsze będzie dążyć do maksymalnej. Dodatkowo każda zamontowana szuka wymaga niezależnego „pinu” na mikrokontrolerze, przez co ilość przewodów jakie musimy umieścić na kończynach robota wynosi 3n, gdzie n to liczba serwomechanizmów. Odpowiednik po prawej stronie do komunikacji używa transmisji szeregowej, przez co niezależnie od ich ilości zawsze będą to 3 przewody. Dodatkowym argumentem jest fakt, iż możemy uzyskać takie informacje jak aktualna pozycja, napięcie, temperatura a także możemy sterować jego prędkością. Przydatną funkcją jest załączanie i wyłączanie momentu obrotowego, którą wykorzystamy w późniejszym etapie.

### 3. Projekt robota

Wiedząc już czym się kierować przy doborze podzespołów możemy przejść do rzeczywistej fazy projektowania. Najlepszym rozwiązaniem jest użycie programu umożliwiającego projektowanie 3D nawet jeżeli korzystamy z gotowych podzespołów. Zaletą tego rozwiązania jest fakt, że możemy zweryfikować czy nie występują kolizje podczas poruszania się oraz jakie maksymalne ruchy możemy wykonać. Jednak najważniejszą zaletą jest to, że przy odpowiednio wprowadzonych materiałach, z których wykonane są dane części lub ich wagi program automatycznie wyliczy nam momenty bezwładności danego komponentu potrzebnego później w metodzie „zero moment point”.

#### 3.1 Porównanie konfiguracji nóg robota

Na początku należy wprowadzić pojęcia fazy podporowej oraz przenoszenia kończyny. Faza podporowa występuje, gdy dana noga styka się z podłożem i robot utrzymuje dzięki niej równowagę, faza przenoszenia natomiast występuje, gdy noga nie ma styczności z podłożem i dana kończyna przemieszczana jest do przodu w celu wykonania kolejnego kroku. Obie fazy występują naprzemiennie wyjątkiem jest, gdy robot nie przemieszcza się wówczas obie kończyny są w fazie podporowej. Przedstawione na rys. 2 konfiguracje lewej nogi podczas fazy podporowej różnią się od siebie nie tylko ilością zastosowanych serwomechanizmów, lecz przede wszystkim możliwościami robota, a dokładniej odwzorowaniem ludzkiej nogi.



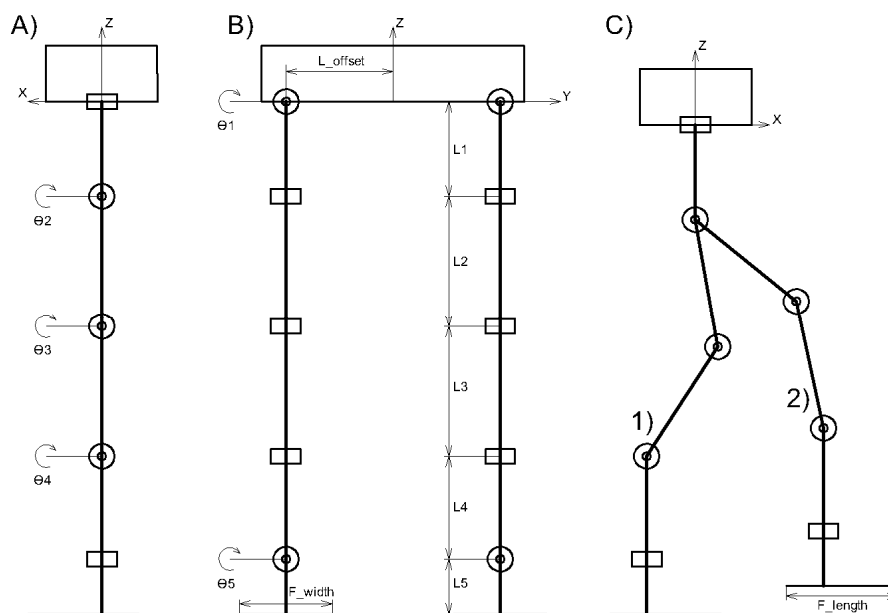
Rysunek 2. Porównanie konfiguracji lewej nogi robota

Nie trudno zauważyć, że konfiguracja „A” składająca się z trzech serwomechanizmów sprawia, że tułów robota przechyla się na boki ze względu na brak biodra. Ruch ten sprzyja utrzymywaniu równowagi, przez co programowanie takiej konstrukcji nie jest trudne jednak łatwo wytracić ją z równowagi podczas fazy przenoszenia. Konfiguracje „B” oraz „D” są niemal identyczne. Różnica w ułożeniu

serwomechanizmów nie wpływa na znacząco na możliwości danej konstrukcji, ponieważ w obu przypadkach istnieje możliwość zginania nogi w kolanie oraz nie występuje już przechyłanie tułowia. Nadal jednak brakuje możliwości obrotu nogi wokół własnej osi jaka znajduje się w ludzkim biodrze sprawiając, że obrót robota w miejscu jest bardzo utrudniony. Wady tej nie ma ostatnia z porównywanych konfiguracji „C” jest ona najbardziej zbliżona do ludzkiej nogi i jako jedyna umożliwia zastosowanie metody genetycznej. Konfiguracja „D” różni się od pozostałych złożonych z gotowych podzespołów montażowych dostępnych na rynku, ponieważ została zaprojektowana w środowisku CATIA umożliwiającym wykonanie własnej unikatowej konstrukcji wydrukowanej później za pomocą druku 3D.

### 3.2. Kinematyka kończyny

Po wyborze odpowiedniej konfiguracji nadszedł czas na opisanie jej geometrycznie i od tej chwili konieczna jest znajomość pojęć kinematyka prosta i odwrotna. Tematyka ta jest bardzo obszerna i można się z nią zaznajomić w innej mojej publikacji [6], w której opisuje te zagadnienia na prostym przykładzie. Kinematykę prostą wykorzystujemy, gdy znając wymiary oraz pozycje kątowne poszczególnych serwomechanizmów obliczamy położenie końca kończyny w układzie kartezjańskim. Natomiast kinematykę odwrotną używamy, aby obliczyć pozycje kątowne serwomechanizmów tak aby kończyna znalazła się w zadanym miejscu.



Rysunek 3. Kinematyka nogi robota

Na rys. 3 A) oraz B) przedstawiono wymiary poszczególnych ogniw oraz kąty obrotu  $\Theta$ . Odnosząc się do geometrii ludzkiej nogi kąty  $\Theta_1$  oraz  $\Theta_2$  opisująby obrót biodra,  $\Theta_3$  – kolana, natomiast  $\Theta_4$  oraz  $\Theta_5$  kostki. Na rysunku zaznaczony został również środek układu współrzędnych, a wymiar „L\_offset” oznacza odsunięcie kończyny od jego początku w osi Y. Rys. 3 C przedstawia, jak będzie wyglądała kończyna podczas

fazy 1) Podpory, 2) Przenoszenia. Możemy zauważyć, że środek układu współrzędnych znajduje się niżej niż na rysunkach A oraz B, jest to normalne zjawisko i wynika z faktu, iż jeżeli robot poruszałby się na wyprostowanych nogach trajektorie środka układu współrzędnych w osi Z można by porównać do wykresu wartości bezwzględnej z funkcji cosinus. Dodatkowo podczas przemieszcza się na ugiętych nogach zawsze istnieje możliwość korekty pozycji w przypadku wytrącenia robota z równowagi zapobiegając jego przewróceniu się.

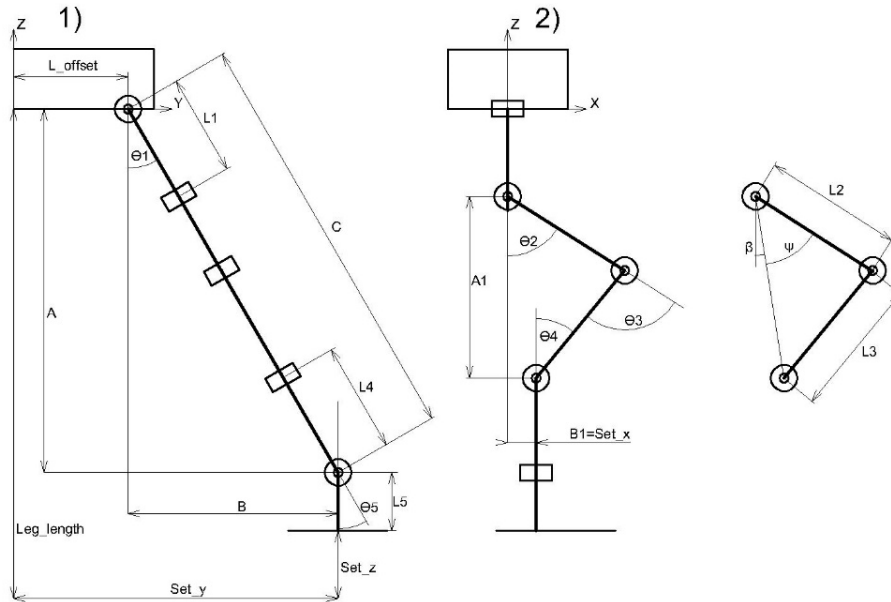
#### 4. Wyznaczanie trajektorii ruchu

Kolejnym krokiem powinno być wyznaczenie dynamiki ruchu kończyny oraz wyznaczenie masowych momentów bezwładności. Ze względu na stopień złożoności, w początkowej fazie równania ruchu zostaną uproszczone. Z własnego doświadczenia wiem również, że należy stopniowo uruchamiać funkcje robota, gdyż łatwiej w ten sposób zlokalizować pojawiające się problemy. Trajektorja robota dwunożnego najczęściej oparta jest o analizę ludzkich nóg i poszczególne fazy ruchu wyznaczane są zgodnie z tą zależnością. Jednak nim robot wykona pierwszy poprawny ruch kończynami należy odpowiednio oprogramować kończyny aby ich ruch był zależny od wprowadzonej wartości przemieszczenia, prędkości i przyśpieszenia.

##### 4.1 Przemieszczenie

Do uzyskania zadanej pozycji końca łańcucha kinematycznego (stopy) należy wykorzystać wspomnianą wcześniej kinematykę odwrotną. Istnieje kilka sposobów, w jaki można ją uzyskać. Możemy np. zastosować metody numeryczne, przekształcić równania notacji Denavita-Hartenberga tak, aby uzyskać wartości kątów obrotu. Jednak obie te metody generują dużo obliczeń, które spowolnią prace mikrokontrolera oraz wydłużą czas reakcji przy wytrąceniu z równowagi. Analizując rys. 3 można zauważyć, że zamontowane serwomechanizmy umożliwiają ruch nogi w dwóch płaszczyznach. Mianowicie zmiana wartości kątów  $\Theta_1$  oraz  $\Theta_5$  wpływa na przemieszczenie w płaszczyźnie OYZ, natomiast  $\Theta_2$ ,  $\Theta_3$ ,  $\Theta_4$  w płaszczyźnie OXZ. Sumą tych przemieszczeń jest ruch w układzie OXYZ. To samo zjawisko zachodzi w konfiguracji C (rys. 2). Należy powiązać lokalne układy współrzędnych z analizowaną kończyną, dla której orientacja osi Z –  $Z_1$  zależna by była od  $\Theta_0$ . Biorąc pod uwagę powyższe zależności kończynę można opisać geometrycznie co zostało przedstawione na rys. 4. W celu uproszczenia analizy założono, że stopa robota jest zawsze równoległa do podłoża.

Powyższe zależności są wystarczające, aby obliczyć kąty  $\Theta$  przyjmujące wartości od  $-90^\circ$  do  $90^\circ$ , przy czym pozycje zerowe przedstawione zostały na rys. 3. Należy również uwzględnić kolizje z innymi elementami konstrukcji, przez co wartości graniczne będą różne dla każdej osi. Z rys. 4 - 1 wynika, że kąty  $\Theta_1$  oraz  $\Theta_5$  są takie same natomiast do obliczenia ich wartości należy wykorzystać twierdzenie pitagorasa (1). Stosując to twierdzenie należy pamiętać, że obliczona wartość funkcji arccos zawsze będzie większa od zera, ponieważ wykorzystuje ona wartość bezwzględną B przez co w zależności (2) została ona przemnożona przez  $B/|B|$  dzięki czemu możliwe będzie osiągnięcie wartości ujemnych.



Rysunek 4 Analiza kinematyki odwrotnej nogi robota

Ze względu na fakt, iż wartości funkcji trygonometrycznych wyrażone są w radianach wynik należy przemnożyć przez  $180/\pi$ , aby uzyskać wynik w stopniach. Kąty  $\Theta_5$  oraz  $\Theta_1$  różnią się kierunkiem stąd w równaniu (3) pojawił się minus.

$$C = \sqrt{A^2 + B^2} = \sqrt{(Leg_{Length} - L_5 - Set_z)^2 + (Set_y - L_{offset})^2} \quad (1)$$

$$\Theta_1 = \arccos\left(\frac{A}{C}\right) \cdot \frac{B}{|B|} \cdot \frac{180}{\pi} \quad B \neq 0 \quad (2)$$

$$\Theta_5 = -\Theta_1 \quad (3)$$

Zależności (1) - (3) pozwoliły na obliczenie wartości kątów w płaszczyźnie OYZ kolejnym krokiem jest przejście do płaszczyzny OXZ przedstawionej na rys. 4 - 2, z którego wynika, że wartość kąta  $\Theta_2$  (6) jest sumą kątów  $\psi$  oraz  $\beta$  do których obliczenia wykorzystane zostały zależności geometryczne (4) oraz (5).

$$\beta = \arctan 2(B_1, A_1) = \arctan 2(Set_x, C - L_1 - L_4) \quad (4)$$

$$\psi = \arccos\left(\frac{A_1^2 + B_1^2 + L_2^2 - L_3^2}{2 \cdot L_2 \cdot \sqrt{A_1^2 + B_1^2}}\right) \quad (5)$$

$$\Theta_2 = (\beta + \psi) \cdot \frac{180}{\pi} \quad (6)$$

Podobną zależność geometryczną (7) wykorzystano do wyznaczenia kąta  $\Theta_3$ . Przy czym z rys. 3 wynika, że kąt ten zawsze będzie mniejszy lub równy 0 stąd w równaniu (7) pojawił się minus:

$$\Theta_3 = \arccos\left(\frac{A_1^2 + B_1^2 - L_2^2 - L_3^2}{2 \cdot L_2 \cdot L_3}\right) \cdot \left(-\frac{180}{\pi}\right) \quad (7)$$

W ostatniej zależności (8) wykorzystano założenie, że stopa jest równoległa do podłoża, przez co wartość kąta  $\Theta_4$  będzie sumą  $\Theta_2$  oraz  $\Theta_3$ :

$$\Theta_4 = -(\Theta_2 + \Theta_3) \quad (8)$$

Powyższe zależności są również prawdziwe, gdy usuniemy założenie o równoległości stopy do podłoża jednak należy wtedy uzupełnić je o odpowiednie rzuty długości  $L_4$ ,  $L_5$ ,  $F_{Length}$  oraz  $F_{Width}$  na płaszczyzny OYZ oraz OXZ. Będą one miały bezpośredni wpływ na równania (1), (3), (4) oraz (8) i pośredni na pozostałe. Należy pamiętać, że zależności te wyznaczają pozycje środka stopy (przy założeniu osiowego umieszczenia serwomechanizmów) stąd przy braku równoległości stopy do podłoża konieczne jest również uwzględnienie jej wymiarów ( $F_{Length}$ ,  $F_{Width}$ ). Jest to konieczne, aby założona pozycja w osi Z (*Setz*) dotyczyła najniższego punktu stopy.

#### 4.2. Prędkość

Znając już docelowe wartości kątów  $\Theta$  możemy przystąpić do testów. W tym celu przeprowadzono kilkakrotnie dwa testy przy różnych prędkościach ruchu. Test A do którego wykorzystane zostały obliczenia kinematyki odwrotnej oraz test B, w którym manualnie ustawiając kończynę zapamiętano pozycje startową oraz docelową. Oba testy przeprowadzono stopniowo zwiększając prędkości ruchu. Uzyskane wyniki pozwoliły na stwierdzenie, że zarówno w teście A jak i B czas ruchu był taki sam dla danej prędkości. Stwierdzono również, że w obu testach wszystkie serwomechanizmy zaczynają ruch w tym samym czasie jednak każdy kończy osobno. Wynika to ze stałej prędkości kątowej i różnej drogi ruchu skąd nasuwa się kwestia wyznaczania różnych prędkości, dodatkowo można pokusić się o stwierdzenie czy można by było wykorzystując test B pominąć obliczenia kinematyki odwrotnej. Jak najbardziej jest to możliwe jednak musielibyśmy zapamiętać wszystkie możliwe pozycje dla każdej kończyny a następnie przełączać się pomiędzy nimi. Wadą tego rozwiązania jest brak możliwości korekty pozycji w przypadku wytrącenia z równowagi, przez co taki robot nie byłby odporny na działanie sił zewnętrznych i mógłby się poruszać jedynie po znanej wcześniej trajektorii. Wracając do tematu wyznaczania prędkości poszczególnych osi konieczna jest znajomość różnicy w pozycjach startowej oraz docelowej dla każdej osi. Pozycje docelową wiemy już jak wyznaczyć, natomiast startową możemy wyznaczyć na dwa sposoby. Możemy pamiętać ostatnią wysłaną pozycję lub skorzystać ze sprzężenia zwrotnego odpytując o nią serwomechanizm. Kolejnym krokiem jest znalezienie maksymalnej różnicy (9) wykorzystując funkcje max. Następnie wystarczy przeskalować procentowo prędkości w zależności od różnicy pozycji (10) wykorzystując do tego funkcje map.

$$Max_{diff} = Max(Max_{diff}, |Last\Theta_i - Act\Theta_i|); \quad i \in 1...5 \quad (9)$$

$$Velocity\Theta_i = map(|Last\Theta_i - Act\Theta_i|, 0, Max_{diff}, 0, Set_{velocity}) \quad (10)$$

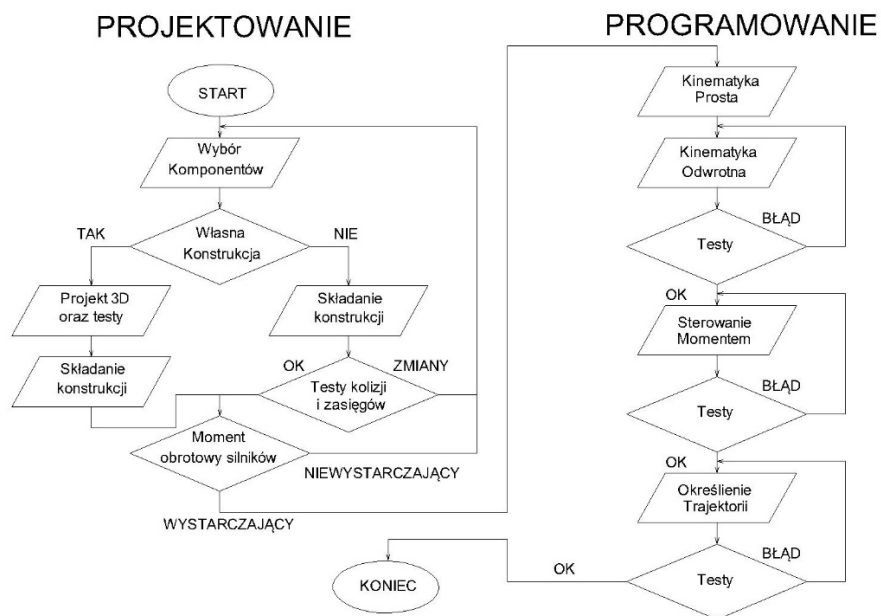


### 4.3. Przyspieszenie

Wcześniejsze zależności pozwoliły na wyznaczenie pozycji oraz prędkości dla każdej osi naszej kończyny. Kolejnym etapem jest wyznaczenie ich przyspieszeń niestety serwomechanizmy modelarskie nie posiadają takiej możliwości można jednak rozwiązać ten problem w inny sposób dzieląc zadaną trajektorię na mniejsze części stopniowo zwiększając lub zmniejszając prędkość dla każdej z nich. Wyznaczanie przyspieszenia jest bardzo ważne z punktu widzenia dynamiki robota i konieczne przy stosowaniu metody „zero moment point” jednak jest to zbyt rozległy temat, aby umieścić go w tej publikacji. Więcej informacji na ten temat można znaleźć w książce [3].

### 5. Wnioski

W artykule przedstawiono chronologicznie prace konieczne do wykonania podczas projektowania robota humanoidalnego. Ze względu na złożoność analizy chodu, przed przystąpieniem do pracy należy zapoznać się m.in. z fizjonomią ruchu człowieka, modelowaniem, dynamiką, doбором układu wykonawczego wraz ze sterowaniem itp..



Rysunek 5. Algorytm postępowania podczas projektowania humanoida – opracowanie własne

Na rys. 5 przedstawiono algorytm postępowania krok po kroku z podziałem na etap projektowania oraz programowania, opracowany na podstawie własnych doświadczeń.

**LITERATURA**

1. BABIARZ A., JASKOT K.: Analiza kinematyki dwunożnego robota kroczącego, Wydawnictwo Politechniki Śląskiej, Zeszyty Naukowe. Automatyka, (2008)150, 107—114.
2. Botland.com Feetech SCS15 Standard: <https://botland.com.pl/pl/serwa-typu-standard/9169-serwo-feetech-scs15-standard.html>, data dostępu: 1.08.2020.
3. CRAIG J.J.: Introduction to robotics: Mechanic and control, Third edition, Pearson Education International inc. 2005.
4. TROJNACKI M.: Generowanie ruchu robota dwunożnego z wykorzystaniem danych opisujących chód człowieka, Pomiary Automatyka Robotyka 3(2009), 15-16.
5. ZAWADA A.: Projekt i budowa mobilnego robota z manipulatorem, Praca dyplomowa - Akademia Techniczno-Humanistyczna w Bielsku-Białej, 2018.
6. ZAWADA A., JANUSZ J.: Design and implementation arm controlled by microprocessor, Designing, researches and exploitation, Akademia Techniczno-Humanistyczna w Bielsku-Białej, 1(2018).
7. SUDHEER A.P., VIJAYAKUMAR R., MOHANDA K.P.: Stable Gait Synthesis and Analysis of a 12-degree of Freedom Biped Robot in Sagittal and Frontal Planes, Journal of Automation, Mobile Robotics & Intelligent Systems, 6(2012)4.
8. da Silva F.M.T.P, José A.T.M.: Kinematic analysis and modelling of biped locomotion systems, Journal of the Brazilian Society of Mechanical Sciences 21(1999)3.
9. LUCERO D.M.A: Kinematic and Dynamic Analysis for Biped Robots Design, Tesis Doctoral, Universidad Carlos III de Madrid, Leganes, 2012.
10. MENG L., MACLEOD C.A., PORR B., GOLLEE H.: Bipedal robotic walking control derived from analysis of human locomotion, Biological Cybernetics (2018)112, 277–290.
11. GUPTA A., SHAMRA A.: Modelling and Analysis of Walking Pattern for a Biped Robot, [https://www.researchgate.net/publication/280970165\\_Modeling\\_and\\_Analysis\\_of\\_Walking\\_Pattern\\_for\\_a\\_Biped\\_Robot](https://www.researchgate.net/publication/280970165_Modeling_and_Analysis_of_Walking_Pattern_for_a_Biped_Robot), data dostępu: 01.07.2020.