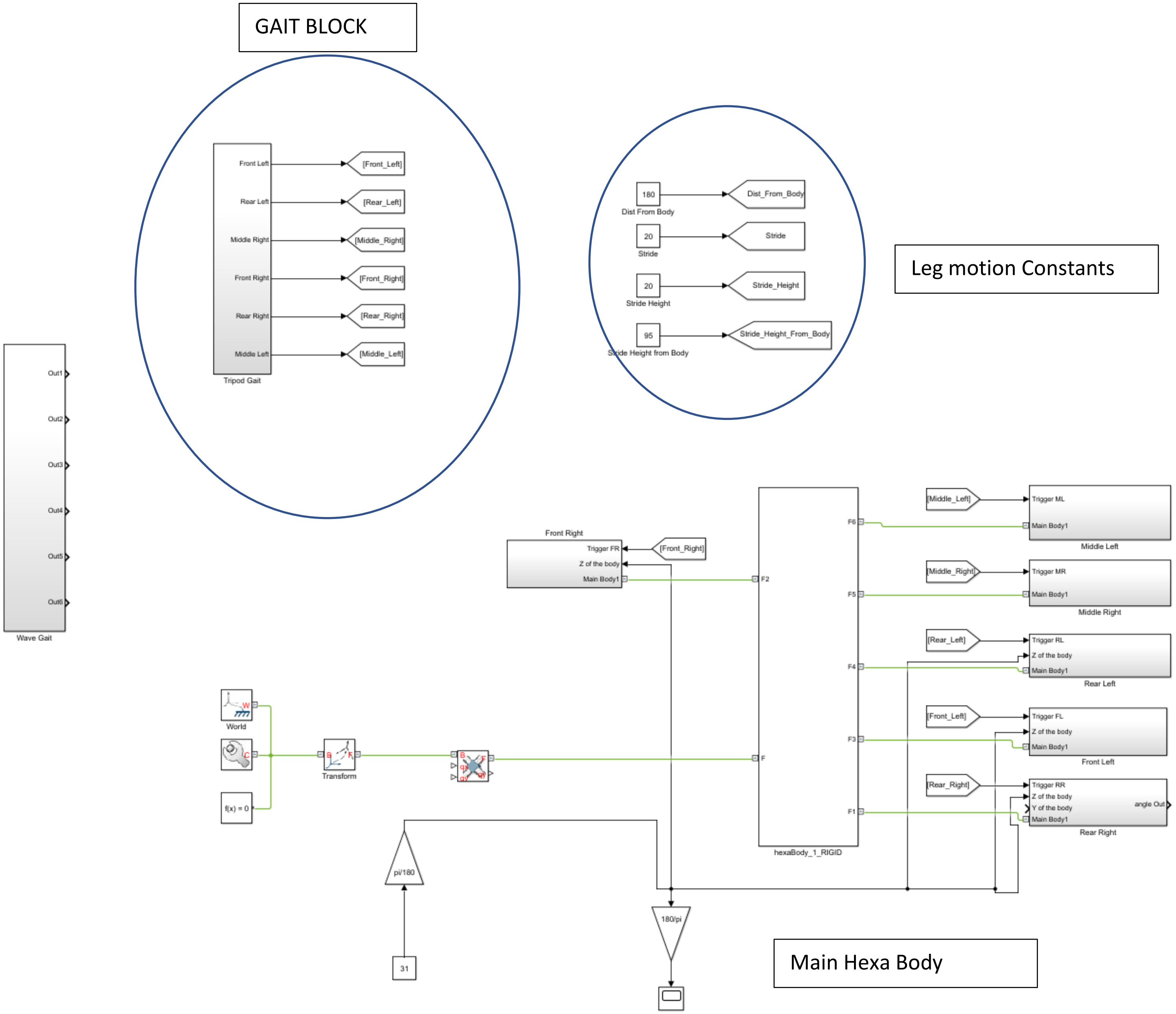


GAIT BLOCK

Leg motion Constants

Main Hexa Body

Gait block

```
[Tripod Gait]
Run the step → Delay 2
    Front Left
    Rear Left
    Middle Right
Run the Step → Delay 1
    Front Right
    Middle Left
    Rear Right
```

Leg motion constants

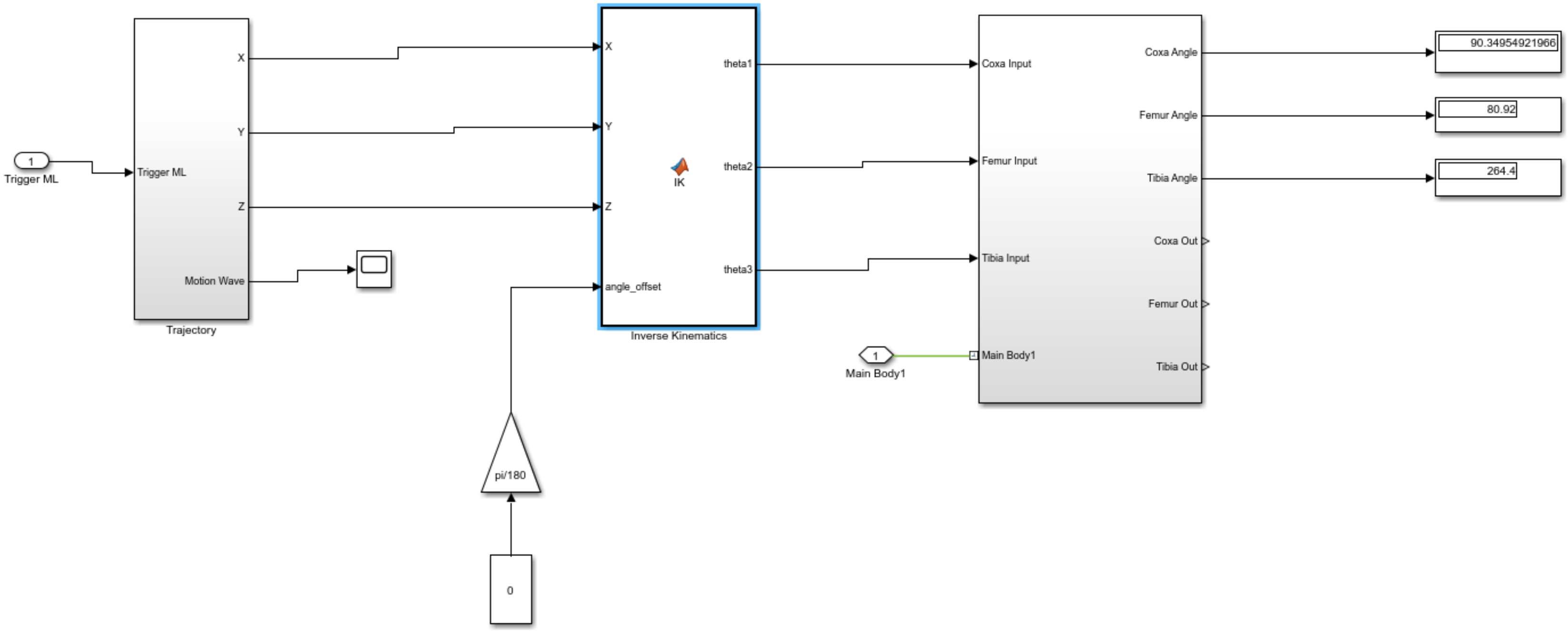
EndEffectorDistanceFromBody = float

StrideSize = float

StrideHeight = float

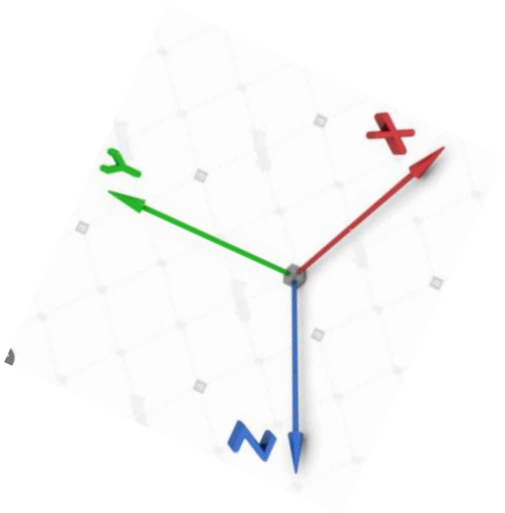
EndEffectorHeightFromBody = float

Single Leg motion



Trajectory Generation → IK → Write to Joints

Trajectory Generation



Sinusoidal Wave for Y

```
y = -stridesize*sin((pi*u)-1.6);
```

(change +- depending on the limb)

Cosine Wave for Z

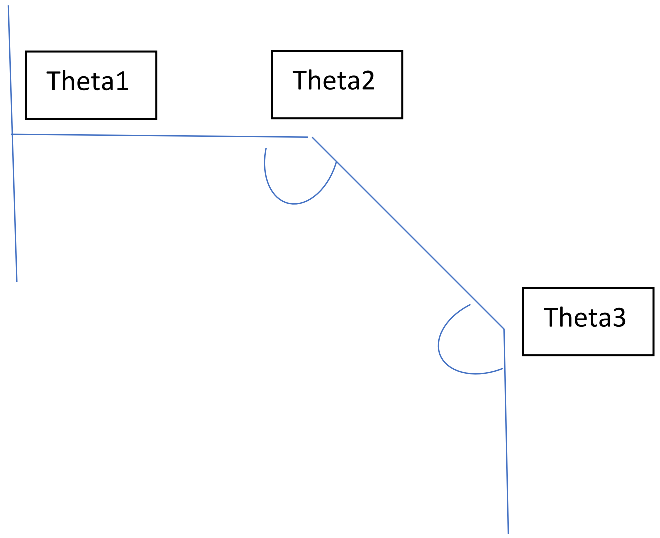
```
z =(stride_height*cos((pi*u)-1.6)+EndEffectorHeightFromBody);
```

X = EndEffectorDistanceFromBody

Inverse Kinematics

```
function [theta1,theta2,theta3]= IK(X,Y,Z,angle_offset)

a_1 = 28.45; //Coxa Length
a_2 = 76.2; //Femur Length
a_3 = 107.22; //Tibia Length

X = X;
Y = Y - 69.78;

L_1 =  sqrt((X^2)+(Y^2))-a_1;
```

```
L_2 = Z;
L_3 = sqrt((L_1)^2+(L_2)^2);
L_3_sqr = L_3^2;

alpha_1_nume = a_2^2+L_3^2-a_3^2;
alpha_1_denom = 2*a_2*L_3;

alpha_1 = acos(alpha_1_nume/alpha_1_denom);

alpha_2_nume = a_2^2+a_3^2-L_3^2;
alpha_2_denom = 2*a_2*a_3;

alpha_2 = acos(alpha_2_nume/alpha_2_denom);

alpha_3 = atan2(L_2,((L_1)));


Theta_1 =  atan2(Y,X);
Theta_2 = (pi/2)+(alpha_3)- (alpha_1);
Theta_3 = pi + alpha_2;

theta1 = (angle_offset)+ Theta_1;
theta2 = Theta_2;
theta3 = Theta_3;


** angle_offset is not applicable in the real model. Ignore that.
```



Write the Values to the motors