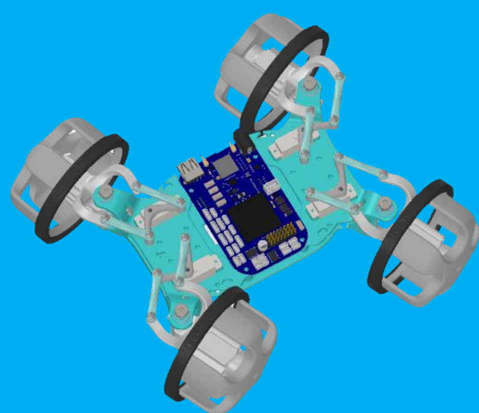
EDUROVER V2

BeagleBone Project



TEAM MEMBERS:

Zhaoliang Zheng (zhz503@eng.ucsd.edu)

Dominique Meyer (demeyer@ucsd.edu)

Neal Bansal (nkbansal@ucsd.edu)

Pengcheng Cao (p5cao@eng.ucsd.edu)

Junchao Lin (jul025@eng.ucsd.edu)

DroneLab

Catalog

0 INTRODUCTION

This is an application of the embedded system: Beaglebone. EduRover is originally designed by James Strawson (jstrawso@eng.ucsd.edu) in the Done Lab@UCSD and Coordinated Robotics Lab@UCSD. In version 2, we made some modification on it, redesigned some part of it. In our case, high torque motors could be perfectly embedded into our design and with the higher level control algorithm, we can manually drive EduRover in different modes. Meanwhile, with the high torque motors, we can balance EduRover by using only two wheels when it faces the wall.

1 STRUCTURE

This chapter describes the structure and some hardware of EduRover as well as the properties of these components. All the Solidworks files can be found in the link: https://github.com/zhz503/Beaglebone-project-EduRover-V2/tree/master/Edu%20Rover%20Solidworks%20Version2

1.1 Configuration in different modes

The figure 1-1 shows the structure of EduRover and in the balance condition.

The figure 1-2 shows the different four-wheeled EduRover driving mode.

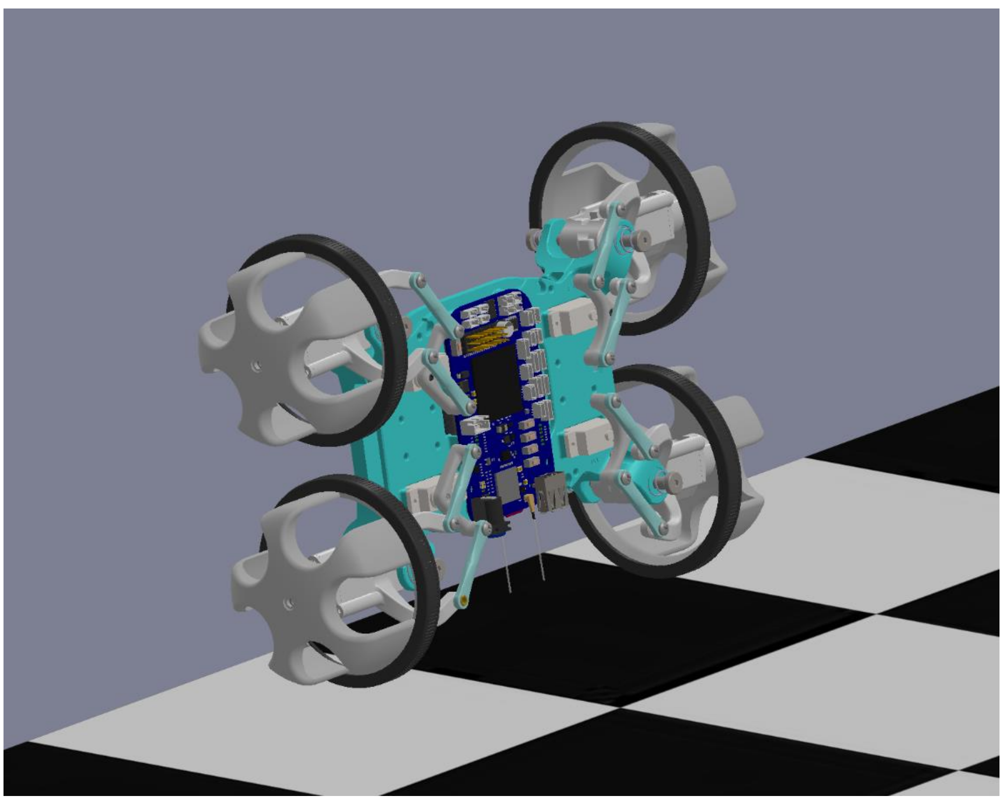
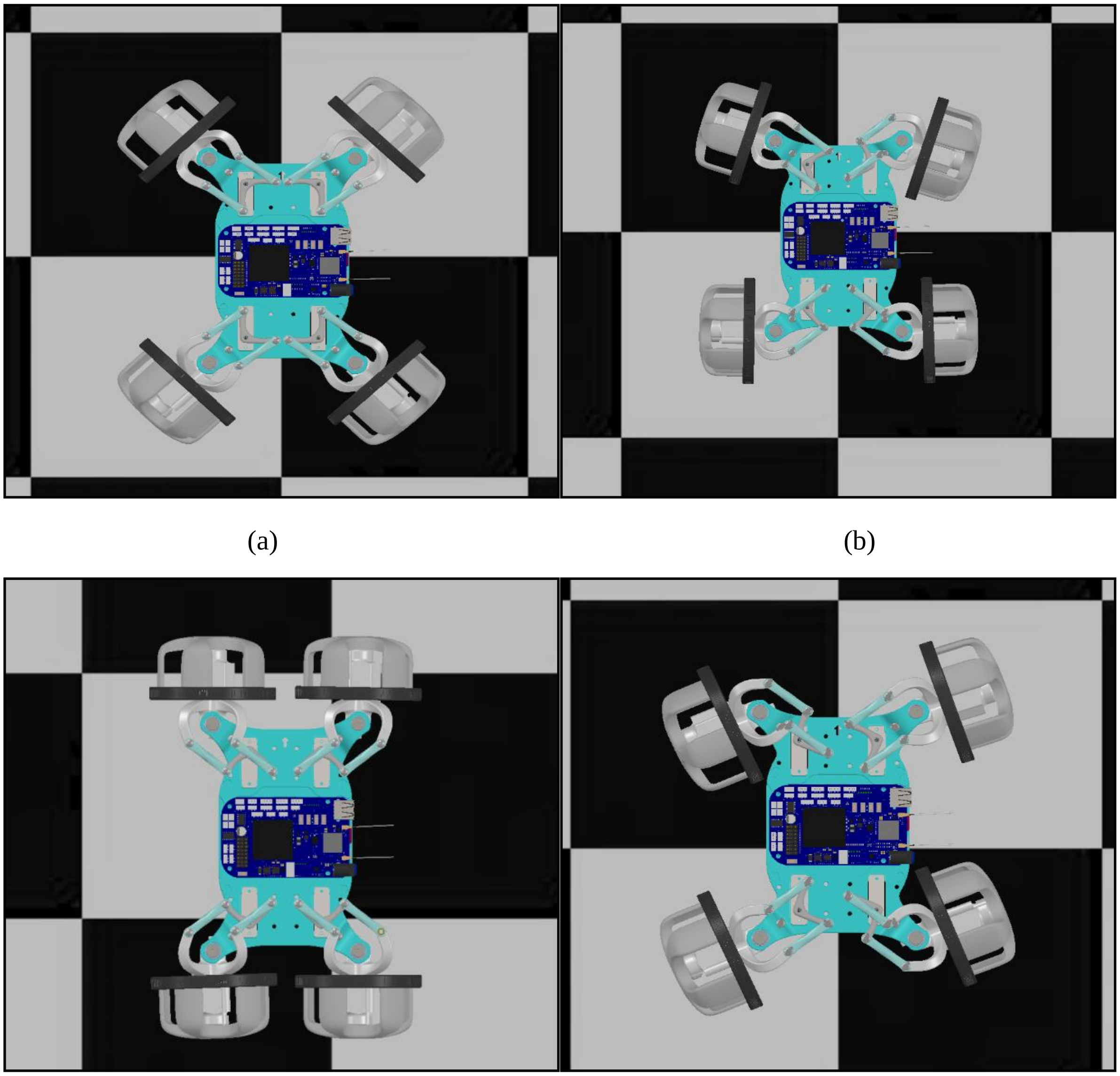


Figure 1-1 Balance Configuration Mode



(a) (b)

(c) (d)

Figure 1-2 (a) is Spinning Mode, (b) is Normal-car Mode, (c) is Quarter Turn Mode, (d) is Sidling Mode

1.2 Sensors and Actuators

Table 1-1 Sensors

Sensor	Output	Unit
Encoder	Angle	rad or deg
IMU: Gyroscope	Angular velocity	deg/sec
IMU: Accelerometer	Linear velocity	m/sec

Table 1-2 Actuators

Actuator	Output	Unit
Servo Motor	Position or duty circle	%

DC Motor	PWM	%

The illustration of DC motor[1] and Servo Motor can be found in appendix A and reference.

2 MATHEMATICAL MODELING

This chapter describes the mathematical model and motion equations of EduRover. Based on the basic knowledge, the EduRover balance mode can be simplified as a two-wheeled inverted pendulum model[2], and EduRover four-wheeled normal-car mode can be simplified as Ackermann drive (bicycle) model. Spining Mode, Quarter Turn Mode, Sidling Mode can be simplified as Omni Wheel model.

2.1 Notation

Items	Parameter	Value	Unit	Meaning
EduRover Parameter	g	9.81	m/sec^2	Gravity acceleration
	m	0.03852	kg	Wheel weight
	R	$D/2$ =0.0425	m	Wheel radius
	J_w	$mR^2/2$	kgm^2	Wheel inertia moment
	M	0.6	kg	Body weight
	W	0.150	m	Body width
	D	0.048	m	Body depth
	H	0.18282	m	Body height
	L	0.08869	m	Distance of the center of mass from the wheel axle
	J_ψ	$ML^2/3$	kgm^2	Body pitch inertia moment
	J_ϕ	$M(W^2+D^2)/12$	kgm^2	Body yaw inertia moment
	f_m	0.0022		Friction coefficient between body & DC motor
	f_w	Depends[4]:0.1		Rolling Friction between wheel & floor
DC motor	T_{or}	0.1765	Nm	Motor torque
	I_m	1.6	A	Motor working current
	J_m	$2.7565e^{-}7$	kgm^2	DC motor inertia moment
	R_m	2.7	Ω	DC motor resistance
	$\mathrm{K_b}$	$7.407e^{-}4$	Vsec/rad	DC motor back EMF constant
	$\mathrm{K_t}$	T_{or}/I_m	Nm/A	DC motor torque constant
	n	99		Gear ratio

Note:

DC motor parameters can be found in [1].

f_m, f_w are hard to verify, so they are a approximation.

[4] Rubber & Concrete(Dry)

2.2 Two-wheeled Inverted Pendulum Model and Differential-drive Model

The EduRover can be simplified as a two-wheeled inverted pendulum model, as described in Figure 2.2-1

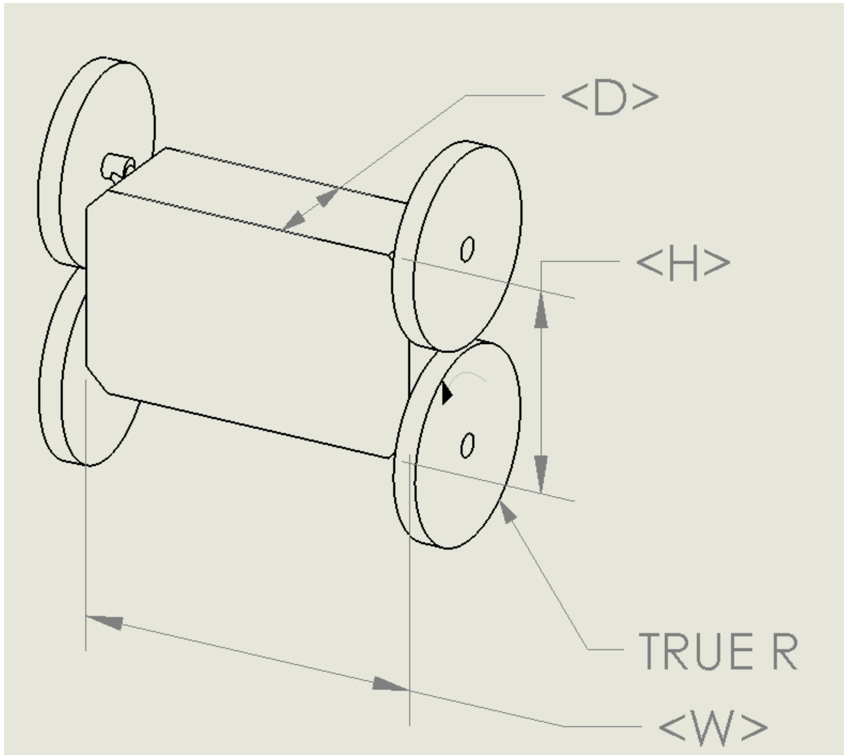
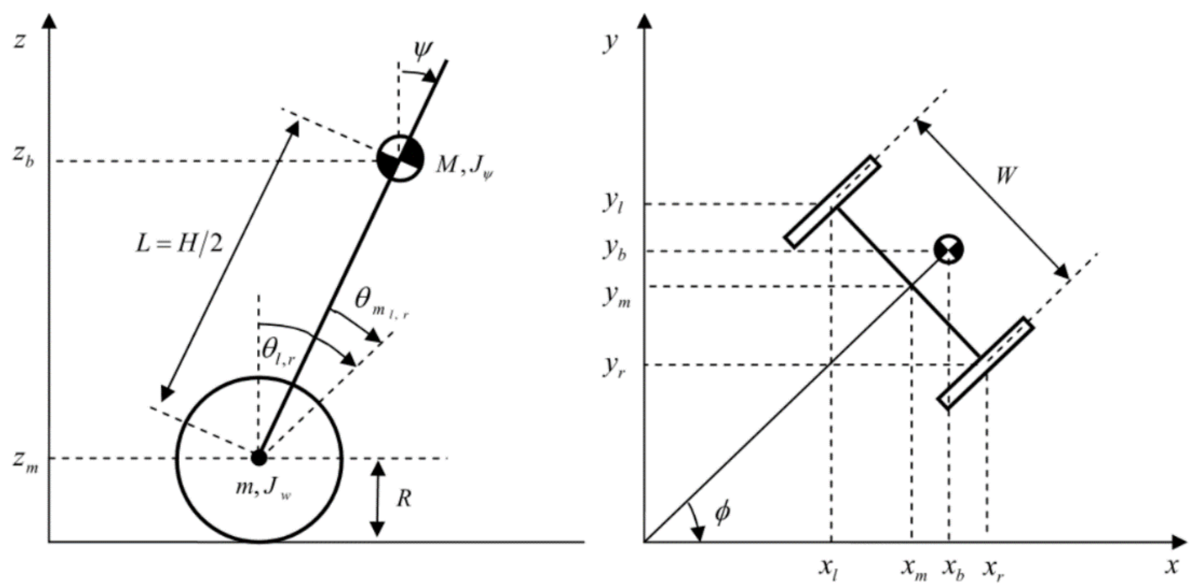


Figure 2.2-1 Two-wheeled Inverted Pendulum Model



ψ : body pitch angle $\theta_{l,r}$: wheel angle (l, r indicates left and right) $\theta_{m_{l,r}}$: DC motor angle

Figure 2.2-2 Side view and plane view of the model

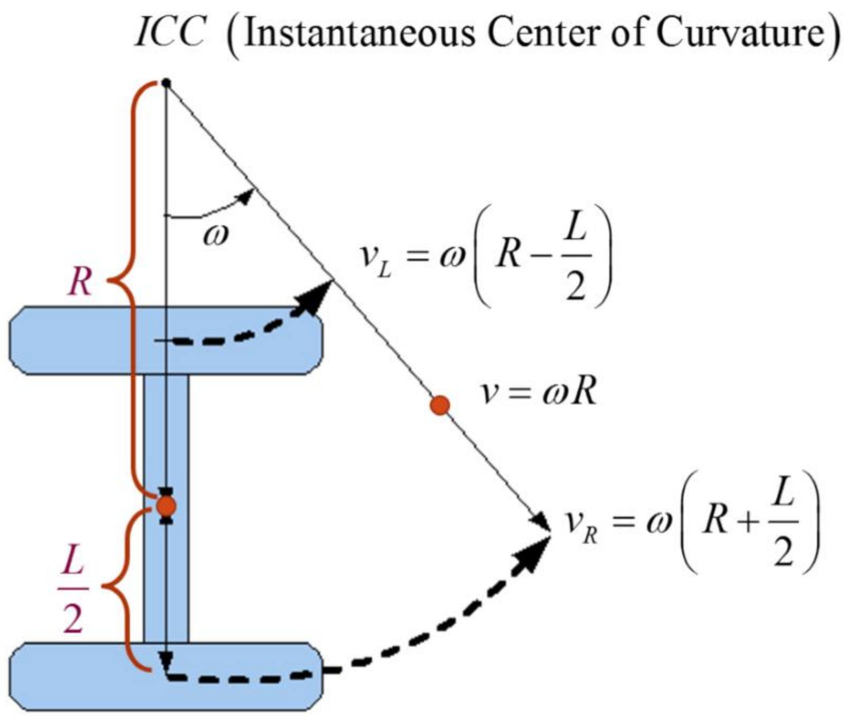


Figure 2.2-3 Differential-drive Model

2.2.1 Motion equations of two-wheeled inverted pendulum

We can derive motion equations for inverted pendulum based on figure 2.2-2 and the differential model in figure 2.2-3 differential-drive model [3].

$$[F_\theta \quad F_\psi \quad F_\phi]^{\mathrm{T}} = \left[\frac{1}{2}(F_l + F_r) \quad F_\psi \quad \frac{R}{W}(F_r - F_l)\right]^{\mathrm{T}}$$

$$[x_m \quad y_m \quad z_m]^{\mathrm{T}} = [R\theta\cos\phi \quad R\theta\sin\phi \quad R]^{\mathrm{T}}$$

$$[\theta \quad \phi]^T = \left[\frac{1}{2}(\theta_l + \theta_r) \quad \frac{R}{W}(\theta_r - \theta_l)\right]^T$$

$$[x_l \quad y_l \quad z_l]^{\mathrm{T}} = \left[x_m - \frac{W}{2}\sin\phi \quad y_m + \frac{W}{2}\cos\phi \quad z_m\right]^{\mathrm{T}}$$

$$[x_r \quad y_r \quad z_r]^{\mathrm{T}} = \left[x_m + \frac{W}{2}\sin\phi \quad y_m - \frac{W}{2}\cos\phi \quad z_m\right]^{\mathrm{T}}$$

$$[x_b \quad y_b \quad z_b]^{\mathrm{T}} = [x_m + L\sin\psi\cos\phi \quad y_m + L\sin\psi\sin\phi \quad \mathrm{z_m} + \mathrm{L}\cos\psi]^{\mathrm{T}}$$

Where,

θ : average rotational angle of left and right wheel

ψ: Body pitch angle

ϕ: Body yaw angle

Based on the kinetic energy theorem and the potential energy theorem, the translational kinetic energy $\mathrm{E_{k1}}$ and the rotational kinetic energy $\mathrm{E_{k2}}$, the potential energy $\mathrm{E_p}$ are:

$$\mathrm{E_{k1}} = \frac{1}{2}m\left(\dot{x_l}^2 + \dot{y_l}^2 + \dot{z_l}^2\right) + \frac{1}{2}m\left(\dot{x_r}^2 + \dot{y_r}^2 + \dot{z_r}^2\right) + \frac{1}{2}M\left(\dot{x_b}^2 + \dot{y_b}^2 + \dot{z_b}^2\right)$$

$$\mathrm{E_{k2}} = \frac{1}{2}J_w\dot{\theta_l}^2 + \frac{1}{2}J_w\dot{\theta_r}^2 + \frac{1}{2}J_\psi\dot{\psi}^2 + \frac{1}{2}J_\phi\dot{\phi}^2 + \frac{1}{2}n^2J_m(\dot{\theta}_l - \dot{\psi})^2 + \frac{1}{2}n^2J_m(\dot{\theta}_r - \dot{\psi})^2$$

$$\mathrm{E_p} = mgz_l + mgz_r + Mgz_b$$

Note: the last two terms in $\mathrm{E_{k2}}$ equation are the rotational kinetic energy of armature in the left and right DC motor.

2.2.2 Lagrangian equations

Based on the Lagrangian method, The Lagrangian L has the following expression:

$$\mathrm{L} = \mathrm{E_{k1}} + \mathrm{E_{k2}} - \mathrm{E}_p$$

We use the variables θ, ψ, ϕ as the generalized coordinate. The Lagrangian equation can be derived as:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\,\dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = F_\theta$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\,\dot{\psi}}\right) - \frac{\partial L}{\partial \psi} = F_\psi$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\,\dot{\phi}}\right) - \frac{\partial L}{\partial \phi} = F_\phi$$

Expand the equations ()-(), we can derive the following equations, the whole derivation is in the Appendix:

$$\left((2\mathrm{m} + \mathrm{M})\mathrm{R}^2 + 2\mathrm{J_w} + 2\mathrm{n}^2\mathrm{J_m}\right)\ddot{\theta} + (\mathrm{MLRcos\psi} - 2\mathrm{n}^2\mathrm{J_m})\ddot{\psi} - \mathrm{MLRsin\psi}\ \dot{\psi}^2$$
$$-\left((2m + M)R^2\theta + MLR\mathrm{sin\psi}\right)\dot{\phi}^2 = \mathrm{F_\theta}$$

$$\left(ML^2 + J_\psi + 2n^2J_m\right)\ddot{\psi} + (-2\mathrm{n}^2\mathrm{J_m} + \mathrm{MLRcos\psi})\ddot{\theta} - MgLsin\psi$$
$$-(\mathrm{MLR}\theta + ML^2sin\psi)\dot{\phi}^2\mathrm{cos\psi} = F_\psi$$

$$\left(\frac{mW^2}{2} + \frac{W^2}{2R^2}(J_w + n^2J_m) + J_\phi + 2MLR\theta sin\psi + (2\mathrm{m} + \mathrm{M})\mathrm{R}^2\theta^2\right)\ddot{\phi} + 2(2m + M)R^2\theta\dot{\theta}\dot{\phi}$$
$$+ 2ML^2sin\psi\mathrm{cos\psi}\ \dot{\psi}\dot{\phi} + 2MLR\dot{\phi}(\dot{\theta}sin\psi + \mathrm{cos\psi}\ \theta\dot{\psi}) = F_\phi$$

Considering the rolling friction between motor and wheel as well as the friction between wheel and floor, the generalized force are:

$$\mathrm{F_l} = nK_ti_l + \mathrm{f_m}\left(\dot{\psi} - \dot{\theta}_l\right) - \mathrm{f_w}\dot{\theta}_l$$

$$\mathrm{F_r} = nK_ti_r + \mathrm{f_m}\left(\dot{\psi} - \dot{\theta}_r\right) - \mathrm{f_w}\dot{\theta}_r$$

$$F_\psi = -nK_t i_r - nK_t i_l - \mathrm{f_m}(\dot{\psi} - \dot{\theta}_r) - \mathrm{f_m}(\dot{\psi} - \dot{\theta}_l)$$

Where $i_{l,r}$ is DC motor current.

Since the input of our system is voltage of DC motor, in the above equations, we cannot used current $i_{l,r}$ directly. Therefore, we have to derive the relationship between current $i_{l,r}$ and voltage $\mathrm{v_{l,r}}$. From DC motor model, The applied voltage equals the voltage drop across the coil resistance, R, and the inductor, L, plus the back-EMF[5], which is as follows:[4]:

$$\mathrm{v_{l,r}} = i_{l,r}\mathrm{R_m} + L_m \frac{di_{l,r}}{dt} + \mathrm{K_b}(\dot{\psi} - \dot{\theta_{l,r}})$$

Where, $\mathrm{R_m}$:DC motor resistance, L_m:DC motor inductance.

For the steady-state torque-speed relationship, $L_m$has no effect, so $L_m$term is negligible and is approximated as zero. Therefore, rewrite equation() as:

$$i_{l,r} = \frac{\mathrm{v_{l,r}} - K_b(\dot{\psi} - \dot{\theta_{l,r}})}{R_m}$$

When substitute the equation() to equations()-(), the generalized force can be expressed as the function of voltage:

$$F_\theta = \frac{\alpha}{2}(\mathrm{v_l} + \mathrm{v_r}) - (\beta + f_w)\dot{\theta} + \beta\dot{\psi}$$

$$F_\psi = -\alpha(\mathrm{v_l} + \mathrm{v_r}) + 2\beta\dot{\theta} - 2\beta\dot{\psi}$$

$$F_\phi = \frac{R}{W}\alpha(\mathrm{v_l} - \mathrm{v_r}) - (\beta + f_w)\dot{\phi}$$

Where, $\alpha = \frac{nK_t}{R_m}$, $\beta = \frac{nK_tK_b}{R_m} + \mathrm{f_m}$

2.2.3 State equation linearization

So far, we've had all the equations we need. At the balance point, when the body angle ψ changes at about $\pm 5^\circ$, equation() can be linearized. In this case, ψ approximates to be 0, meaning that $\sin\psi \approx 0, \cos\psi \approx 1$, and $\dot{\psi}^2 \approx 0$. The linearized Lagrangian equations() can be rewritten as follows:

$$\left((2\mathrm{m} + \mathrm{M})\mathrm{R}^2 + 2\mathrm{J_w} + 2\mathrm{n}^2\mathrm{J_m}\right)\ddot{\theta} + (\mathrm{MLRcos\psi} - 2\mathrm{n}^2\mathrm{J_m})\ddot{\psi} = \mathrm{F_\theta}$$

$$(\mathrm{ML}^2 + \mathrm{J_\psi} + 2\mathrm{n}^2\mathrm{J_m})\ddot{\psi} + (MLR - 2n^2J_m)\ddot{\theta} - MgL\psi = F_\psi$$

$$\left(\frac{mW^2}{2} + \frac{W^2}{2R^2}(J_w + n^2J_m) + J_\phi\right)\ddot{\phi} = F_\phi$$

Now, we can combine the equ()-() and the equations()-(), and rewrite these equations in matrix form:

$$\begin{cases} \mathrm{E}\begin{bmatrix}\ddot{\theta}\\ \ddot{\psi}\end{bmatrix} + F\begin{bmatrix}\dot{\theta}\\ \dot{\psi}\end{bmatrix} + G\begin{bmatrix}\theta\\ \psi\end{bmatrix} = H\begin{bmatrix}\mathrm{v_l}\\ \mathrm{v_r}\end{bmatrix} \\ I\ddot{\phi} + J\dot{\phi} = K(\mathrm{v_l} - \mathrm{v_r}) \end{cases}$$

Where,

$$E = \begin{bmatrix} (2m+M)R^2 + 2J_w + 2n^2J_m & MLR\cos\psi - 2n^2J_m \\ MLR - 2n^2J_m & ML^2 + J_\psi + 2n^2J_m \end{bmatrix}$$

$$F = \begin{bmatrix} \beta + f_w & -\beta \\ -2\beta & 2\beta \end{bmatrix}, G = \begin{bmatrix} 0 & 0 \\ 0 & -MgL \end{bmatrix}, H = \begin{bmatrix} \alpha/2 & \alpha/2 \\ -\alpha & -\alpha \end{bmatrix}$$

$$I = \frac{mW^2}{2} + \frac{W^2}{2R^2}(J_w + n^2J_m) + J_\phi$$

$$J = \beta + f_w$$

$$K = \frac{R}{W}\alpha$$

Now, we can consider $x_1 = [\theta \quad \psi \quad \dot{\theta} \quad \dot{\psi}]^T, x_2 = [\phi \quad \dot{\phi}]$ as the state, $u = [\nu_l \quad \nu_r]^T$ as input, we can rewrite equations() as state space equations form:

$$\dot{x_1} = A_1x_1 + B_1u$$

$$\dot{x_2} = A_2x_2 + B_2u$$

The state space realization:

$$A_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & A_1(3,2) & A_1(3,3) & A_1(3,4) \\ 0 & A_1(4,2) & A_1(4,3) & A_1(4,4) \end{bmatrix}, B_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ B_1(3) & B_1(3) \\ B_1(4) & B_1(4) \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 1 \\ 0 & -J/I \end{bmatrix}, B_2 = \begin{bmatrix} 0 & 0 \\ -K/I & K/I \end{bmatrix}$$

Where, in the MATLAB:

$$\det(E) = E(1,1)E(1,2) - E(1,2)^2$$

$$A_1(3,2) = -MgLE(1,2)/\det(E)$$

$$A_1(4,2) = MgLE(1,1)/det(E)$$

$$A_1(3,3) = -\frac{(\beta + f_w)E(2,2) + 2\beta E(1,2)}{\det(E)}$$

$$A_1(4,3) = \frac{(\beta + f_w)E(1,2) + 2\beta E(1,1)}{\det(E)}$$

$$A_1(3,4) = \frac{\beta(E(2,2) + 2E(1,2))}{\det(E)}$$

$$A_1(4,4) = -\frac{\beta(E(1,2) + 2E(1,1))}{\det(E)}$$

$$B_1(3) = \frac{\alpha(E(2,2)/2 + E(1,2))}{\det(E)}$$

$$B_1(4) = -\frac{\alpha(E(1,2)/2 + E(1,1))}{\det(E)}$$

3 CONTROL SYSTEM DESIGN

3.1 Control System Theory

3.1.1 Input & Output

A. Balance Mode

Input :$\nu = [\nu_l \quad \nu_r]^T$

Output: $x = [\theta \ \psi \ \dot{\theta} \ \dot{\psi}]^T$

B. Four-wheeled modes

Input: $u = [\text{flag } K_m \ K_s]$

Output: $x = [Mo_{1234} \ Ser_{1234}]$

3.1.2 Stability

(not yet finish)

3.2 Controller design

3.2.1 Equivalent system for balance mode

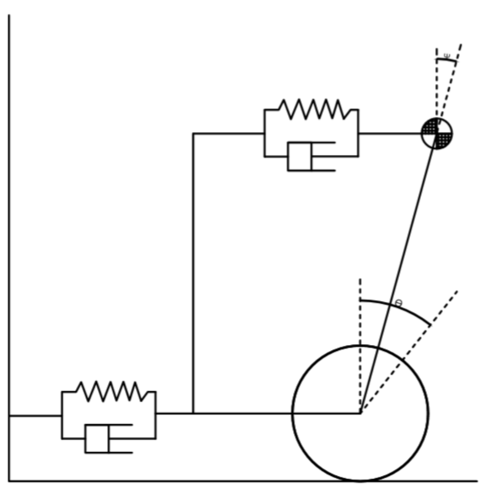


Figure 3.2-1

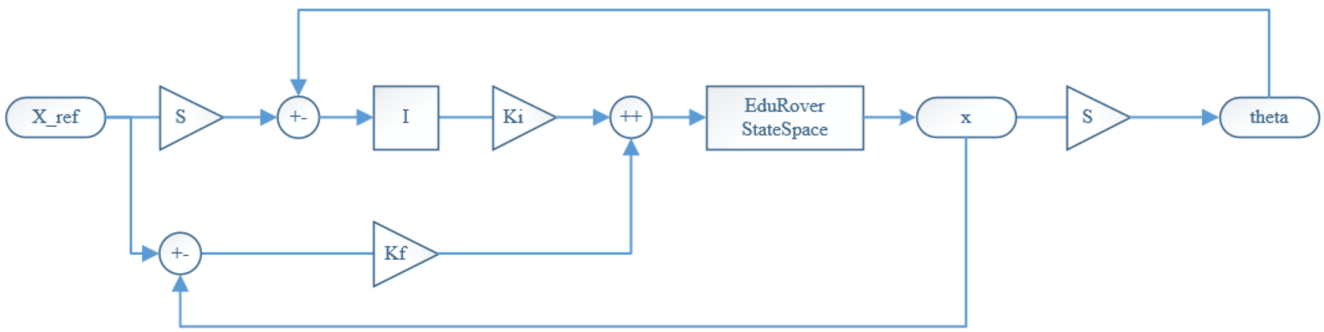


Figure 3.2-2

3.2.2 Simulation

A. Balance Mode

The EduRover balance mode can only be activated when it faces the wall. In the simulation, we only consider the small angle approximation, which is around 5°. We can set the initial condition: $\psi = 5^{\circ}$. The simulation runs in Simulink. The result is as follows.

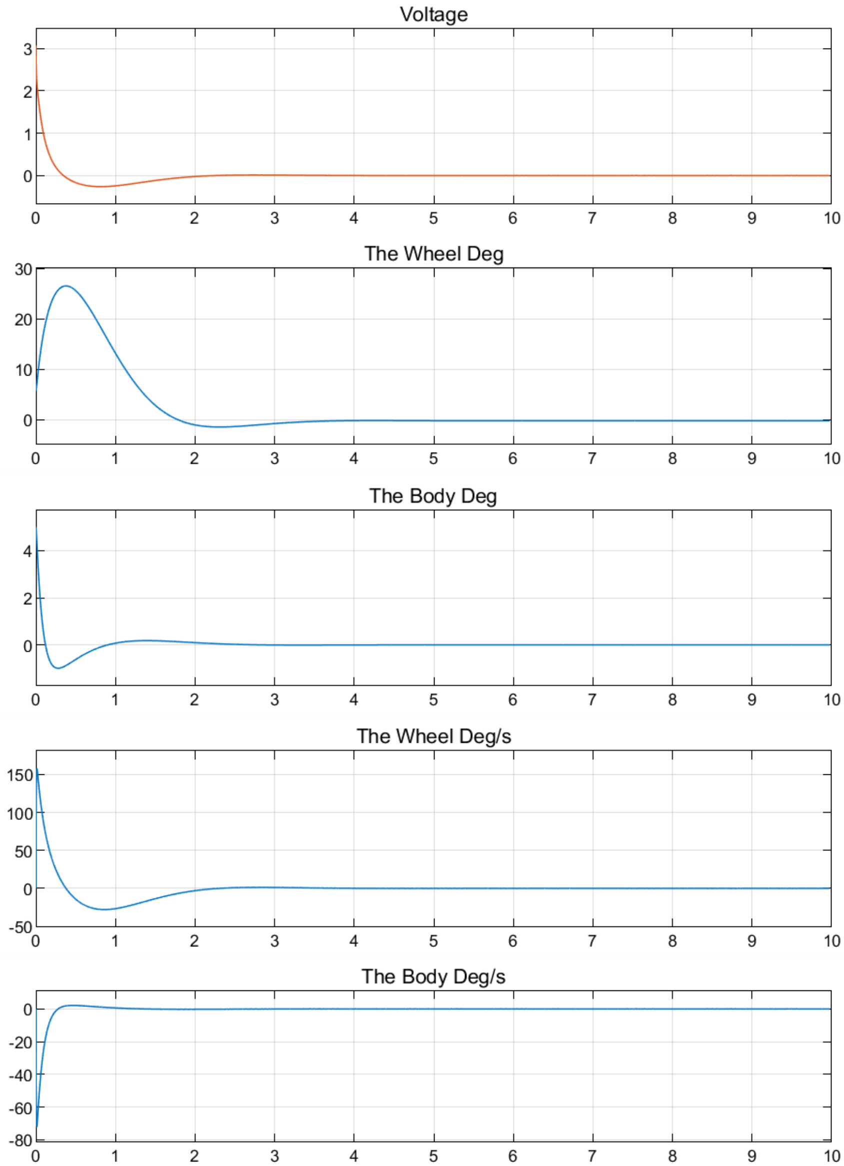


Figure 3.2-1

B. Four-wheeled driving mode

In the four-wheeled driving mode, four situation has to be consider: (a) Spinning mode, (b) Normal-car mode, (c) Quarter-turn mode, (d) Sidling mode. To switch from different modes, we have set up a servo logic for these four different modes.

For manual control, we want to minimize the number of inputs as much as possible. The figure 3.2-2 shows that the servo logics in Simulink. In this servo logics, the flag constant controls the logics switching between different modes. Flag(1): Spinning Mode, flag(2): Normal Car Mode, flag(3): Quarter Turn Mode, flag(4): Sidling Mode.

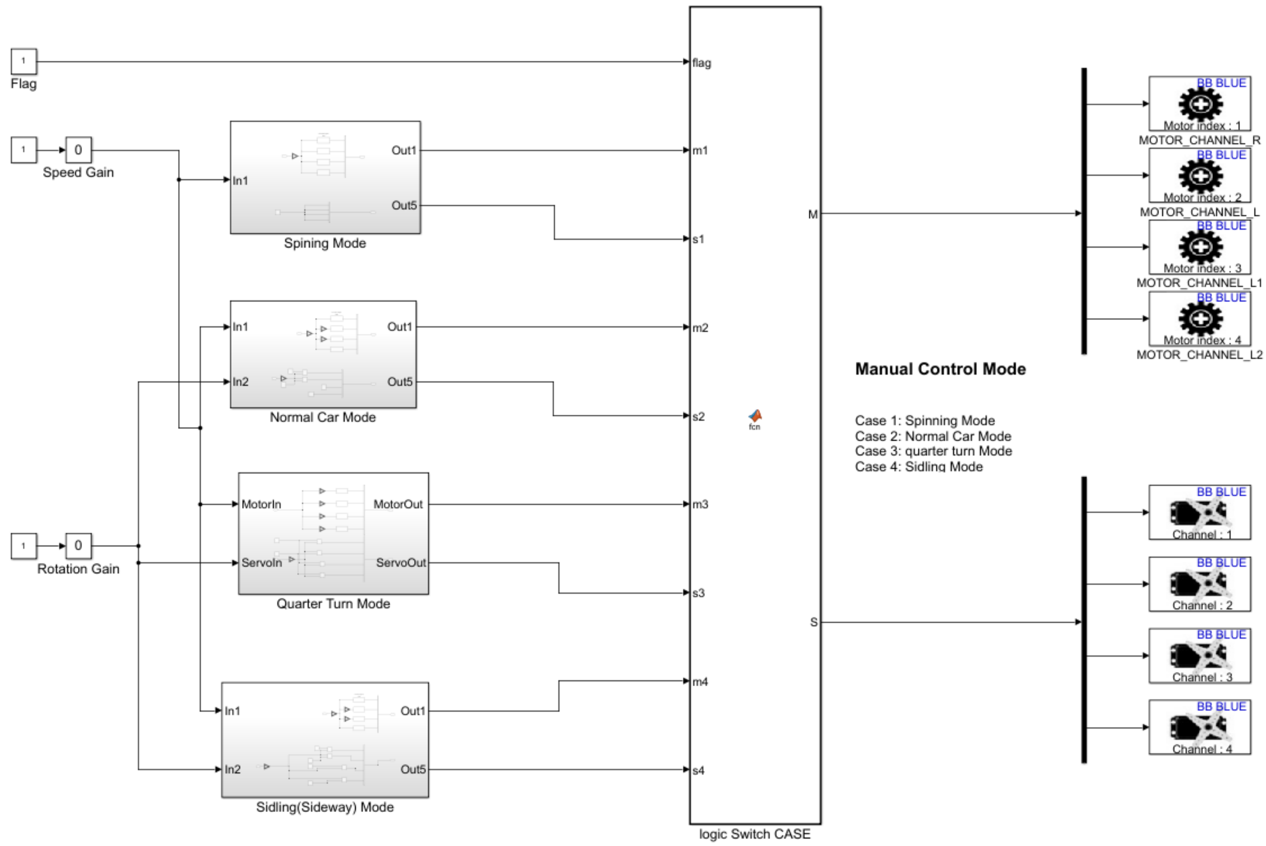


Figure 3.2-2 Servo logics

3.2.3 Real-world Experiment

A. Balance Mode

(not yet finish)

B. Four-wheeled driving mode

We can use the controller designed in Simulink to control the EduRover in real-time in real-world. The following video links can show you how it performs:

(1) https://github.com/zhz503/Beaglebone-project-EduRover-V2/blob/master/RW%20Simulink%20Control%20and%20Sensing/ManualControl.mp4
(2) https://github.com/zhz503/Beaglebone-project-EduRover-V2/blob/master/RW%20Simulink%20Control%20and%20Sensing/logics%20switch.MP4

4 Higher Level Application Design

EduRover is a very decent platform for developer to do higher level application design. There are many applications right now.
In this report, I only list some of the on-going applications:
(1) Way points following based on Motion Capture system
(2) Path planning algorithm based on Motion Capture system
(3) Feature recognition based on web cam on board
(4) ORB-SLAM based on RGBD camera

4.1 Way points following based on Motion Capture system

4.1.1 Closed loop feedback control

(not yet finish)

4.1.2 Motion Capture System

In this project, the motion capture system we used is VICON system. According to the description [5], Vicon are the premier solution for UAV and Robotic studies because we understand what is important; A highly accurate system that provides low latency data that is easy to use. With our turnkey approach and easy to access our DataStream (via TCP, UDP, etc.) users can easily integrate accurate Vicon data into virtually any control system.

4.1.3 PID Control Algorithm

This PID control algorithm is designed by Dominique Meyer, which is available in the Github:
https://github.com/zhz503/Beaglebone-project-EduRover-V2/tree/master/way_points%20following

4.2 Path planning Algorithm on board

4.2.1 Workflow

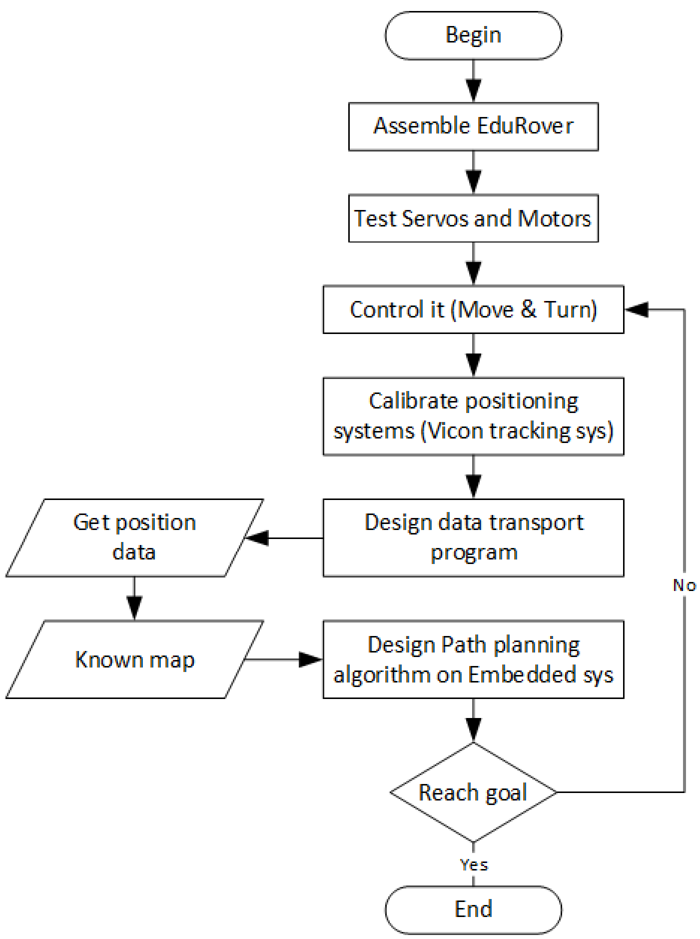


Figure 4-1

Appendix A Table 1-3 Servo Motor Properties

Table 1-3 Servo Motor Properties

Product	*Seller*	*Link*	*No-Load Speed @ 4.8V (sec/60°)*	*Price*	*Qty*	*Price/Unit*
HS-45HB	ServoCity	https://www.servocity.com/hs-45hb-servo	0.14	$15.99	1	$15.99
RB-Fit-03	RobotShop	https://www.robotshop.com/en/9g-micro-servo-motor-4-8v.html#Supplier-Product-Code	0.12	$3.56	1	$3.56
Miuzei SG90	Amazon	https://www.amazon.com/Micro-Helicopter-Airplane-Remote-Control/dp/B072V529YD?ref_=fsclp_pl_dp_1&th=1	0.09	$17.99	10	$1.80
Seamuing MG90S	Amazon	https://www.amazon.com/dp/B07F7VJQL5/ref=psdc_2234131011_t2_B072V529YD?th=1	0.11	$20.99	6	$3.50
SG90 Micro Servo	Amazon	https://www.amazon.com/dp/B07F2XDZZ7/ref=psdc_2234131011_t2_B072V529YD?th=1	0.1	$19.99	10	$2.00

Appendix B The derivation of Lagrangian equations

$$\begin{cases} x_l = R\theta cos\phi - Wsin\phi/2 \\ x_r = R\theta cos\phi + Wsin\phi/2 \\ x_b = R\theta cos\phi + L\sin\psi\cos\phi \end{cases} \Rightarrow \begin{cases} \dot{x_l} = R\dot{\theta} cos\phi - R\theta sin\phi\,\dot{\phi} - W\dot{\phi} cos\phi/2 \\ \dot{x_r} = R\dot{\theta} cos\phi - R\theta sin\phi\,\dot{\phi} + W\dot{\phi} cos\phi/2 \\ \dot{x_b} = R\dot{\theta} cos\phi - R\theta sin\phi\,\dot{\phi} + L\dot{\psi}\cos\psi\cos\phi - L\dot{\phi}\sin\psi\sin\phi \end{cases}$$

$$\dot{x_l}^2 + \dot{x_r}^2 = 2\left(R\dot{\theta} cos\phi - R\theta sin\phi\,\dot{\phi}\right)^2 + W^2\dot{\phi}^2\cos^2\phi/2$$

$$\begin{cases} y_l = R\theta sin\phi + Wcos\phi/2 \\ y_r = R\theta sin\phi - Wcos\phi/2 \\ y_b = R\theta sin\phi + L\sin\psi\sin\phi \end{cases} \Rightarrow \begin{cases} \dot{y_l} = R\dot{\theta} sin\phi + R\theta cos\phi\,\dot{\phi} - W\dot{\phi} sin\phi/2 \\ \dot{y_r} = R\dot{\theta} sin\phi + R\theta cos\phi\,\dot{\phi} + W\dot{\phi} sin\phi/2 \\ \dot{y_b} = R\dot{\theta} sin\phi + R\theta cos\phi\,\dot{\phi} + L\dot{\psi}\cos\psi\sin\phi + L\dot{\phi}\sin\psi\cos\phi \end{cases}$$

$$\dot{y_l}^2 + \dot{y_r}^2 = 2\left(R\dot{\theta} sin\phi + R\theta cos\phi\,\dot{\phi}\right)^2 + W^2\dot{\phi}^2\sin^2\phi/2$$

$$\begin{cases} z_l = R \\ z_r = R \\ z_b = R + L\cos\psi \end{cases} \Rightarrow \begin{cases} \dot{z_l} = 0 \\ \dot{z_r} = 0 \\ \dot{z_b} = -L\dot{\psi} sin\psi \end{cases}$$

$$\mathrm{E_{k1}} = \frac{1}{2}m\left(\dot{x_l}^2 + \dot{y_l}^2 + \dot{z_l}^2\right) + \frac{1}{2}m\left(\dot{x_r}^2 + \dot{y_r}^2 + \dot{z_r}^2\right) + \frac{1}{2}M\left(\dot{x_b}^2 + \dot{y_b}^2 + \dot{z_b}^2\right) = ?$$

$$\begin{cases} \theta = \frac{1}{2}(\theta_l + \theta_r) \\ \phi = \frac{R}{W}(\theta_r - \theta_l) \end{cases} \Rightarrow \begin{cases} \theta_l = \theta - \frac{W}{2R}\phi \\ \theta_r = \theta + \frac{W}{2R}\phi \end{cases} \Rightarrow \begin{cases} \dot{\theta_l} = \dot{\theta} - \frac{W}{2R}\dot{\phi} \\ \dot{\theta_r} = \dot{\theta} + \frac{W}{2R}\dot{\phi} \end{cases}$$

$$\begin{cases} \left(\dot{\theta}_l - \dot{\psi}\right)^2 = \left(\dot{\theta} - \frac{W}{2R}\dot{\phi} - \dot{\psi}\right)^2 \\ (\dot{\theta}_r - \dot{\psi})^2 = \left(\dot{\theta} + \frac{W}{2R}\dot{\phi} - \dot{\psi}\right)^2 \end{cases} \Rightarrow \left(\dot{\theta}_l - \dot{\psi}\right)^2 + (\dot{\theta}_r - \dot{\psi})^2 = 2\left(\dot{\theta} - \dot{\psi}\right)^2 + \left(\frac{W}{2R}\dot{\phi}\right)^2$$

$$\mathrm{E_{k2}} = \frac{1}{2}J_w\dot{\theta}_l^{\ 2} + \frac{1}{2}J_w\dot{\theta}_r^{\ 2} + \frac{1}{2}J_\psi\dot{\psi}^2 + \frac{1}{2}J_\phi\dot{\phi}^2 + \frac{1}{2}n^2J_m(\dot{\theta}_l - \dot{\psi})^2 + \frac{1}{2}n^2J_m(\dot{\theta}_r - \dot{\psi})^2 = ?$$

$$\mathrm{E_p} = mgz_l + mgz_r + Mgz_b = 2\mathrm{m}g\mathrm{R} + \mathrm{M}g(R + L\cos\psi)$$

$$\mathrm{L} = \mathrm{E_{k1}} + \mathrm{E_{k2}} - \mathrm{E}_p = ?$$

$$\frac{\partial L}{\partial\,\dot{\theta}} = \frac{1}{2}\left((2m + M)R^2\left(2\dot{\theta}\right) + MLR\dot{\psi}cos\psi\right) + 2\mathrm{J_w}\dot{\theta} + 2n^2J_m(\dot{\theta} - \dot{\psi})$$

$$\frac{\partial L}{\partial\theta} = ?$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\,\dot{\theta}}\right) = ?$$

$$\frac{\partial L}{\partial\,\dot{\psi}} = ?$$

$$\frac{\partial L}{\partial\psi} = ?$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\,\dot{\psi}}\right) = ?$$

$$\frac{\partial L}{\partial\,\dot{\phi}} = ?$$

$$\frac{\partial L}{\partial\phi} = ?$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\,\dot{\phi}}\right) = ?$$

$$\begin{cases} \left((2\mathrm{m} + \mathrm{M})\mathrm{R}^2 + 2\mathrm{J_w} + 2\mathrm{n}^2\mathrm{J_m}\right)\ddot{\theta} + (\mathrm{MLRcos}\psi - 2\mathrm{n}^2\mathrm{J_m})\ddot{\psi} - \mathrm{MLRsin}\psi\ \dot{\psi}^2 - \left((2m + M)R^2\theta + MLR\mathrm{sin}\psi\right)\dot{\phi}^2 = \mathrm{F_\theta} \\ \left(ML^2 + J_\psi + 2n^2J_m\right)\ddot{\psi} + (-2\mathrm{n}^2\mathrm{J_m} + \mathrm{MLRcos}\psi)\ddot{\theta} - MgLsin\psi - (\mathrm{MLR}\theta + ML^2sin\psi)\dot{\phi}^2\mathrm{cos}\psi = F_\psi \\ \left(\frac{mW^2}{2} + \frac{W^2}{2R^2}(J_w + n^2J_m) + J_\phi + 2MLR\theta sin\psi + (2\mathrm{m} + \mathrm{M})\mathrm{R}^2\theta^2\right)\ddot{\phi} + 2(2m + M)R^2\theta\dot{\theta}\dot{\phi} + 2ML^2sin\psi\mathrm{cos}\psi\ \dot{\psi}\dot{\phi} + 2MLR\dot{\phi}(\dot{\theta}sin\psi + \mathrm{cos}\psi\ \theta\dot{\psi}) = F_\phi \end{cases}$$

REFERENCE

[1] https://www.servocity.com/270-rpm-micro-gear-motor-w-encoder

[2] NXTway-GS Model-Based Design, http://www.pages.drexel.edu/~dml46/Tutorials/BalancingBot/files/NXTway-GS%20Model-Based_Design.pdf

[3] UCSD ECE motion and observation model, https://natanaso.github.io/ece276a/ref/ECE276A_8_MotionObservationModels.pdf

[4] https://www.motioncontroltips.com/faq-whats-relationship-voltage-dc-motor-output-speed/

[5] https://www.vicon.com/motion-capture/engineering