

# Vehicle with Active Transformable Wheels for Various Terrains (August 2019)

VM350 Group 7

Vehicles are playing an important role in modern society in the civil, industrial and military field. However, most vehicles are equipped with common round wheels which are easy to manufacture but cannot satisfy the various kinds of terrains. For example, unlike caterpillar tracks, the round wheels are not capable of tramping over the hills or cruising through marshes, where the round wheels are prone to be trapped in muds. Moreover, when the weather condition is extremely bad, vehicles with round wheels are more likely to lose control on slippery surfaces. However, despite all these deficits, round wheels are very suitable for carrying heavy loading and help the vehicle to reach a high speed without difficulties. As a result, an idea came up in our mind that a transformable wheel can take advantages of both sides with the adaptability to different conditions and situations. In this study, we looked into several successful designs of transformable wheels with sensors to detect different terrains and manufactured our own vehicle whose wheels can automatically transform into different modes to climb the terrace as well as go through the tunnel. In detail, we used SolidWorks to build the models and 3-D printing the parts to build the vehicle. Then, we assembled them and equipped the vehicle with motors, servos, gears, and sensors necessary for transformable wheels. At last, we programmed the system and conducted field tests to adjust the vehicle to a practical and reliable situation. After that, we have manufactured our vehicle with transformable wheels that can come across different terrains successfully as we expected.

## I. INTRODUCTION

To build a transformable wheel, we are faced with two major problems. The first problem is to come up with a reliable and practical transforming mechanism between different forms of the wheel. The transformable wheel has two configurations, one is the open-wheel mode to make sure that the vehicle can climb up to the terrace which is at least 6 cm high by task, and the other is the closed-wheel mode to ensure that the vehicle can go through the tunnel which is 9 cm high. Moreover, in order to be practical, the transformation process of the wheels should be repeatable, meaning that the wheels can either expand or contract at any time, any place according to the current situations. Here comes the other problem: how to determine the exact time and place where our vehicle needs to transform its wheels? The strategy of detecting different terrains and transforming the wheels accordingly is also what we need to come up with to make the transformable wheels feasible.

We have reviewed some literature written by other research teams on transformable wheels and found some relevant literature to help us design our own transformable wheels

better. The literatures are listed as follows:

#	Method	Major findings	Unsolved issues	Ref.
L1	A possible passive transforming mechanism	Passive transforming mechanism could be used to reduce the difficulty of wheel transforming	When the loading exceeds a certain value, the passive method cannot help the transforming	[1]
L2	Active transforming mechanism using purely linkages	How to design linkages for a transformable wheel of a small size	Hard to find materials that meet our demands	[2]
L3	A possible solution for transformable wheels with passive legs	The mechanism of gears and racks transmitted by servos are suitable for transforming process	Need to fix the problems such as the unavoidable sand between the interfaces and increase friction	[3]

Based on these literature reviews, we divided our work into two parts. Two of us designed and manufactured the structure of the vehicle and the mechanism of transformable wheels, and the other two members designed the strategy of the transforming process and programmed the motors, servos and sensors. Finally, we achieved our goal: the vehicle can automatically transform the wheels according to the terrains, to climb up the terrace and pass through the tunnel.

## II. DESIGN

### A. Creative Designs

#### *Passive Petal*

The passive petal design removes the constraints on servo torques that pushes the rack to open the petals when the load is large. The adjective “passive” comes from the different mechanism from the “active” petals. The active petals are rigidly connected to the center of the wheel and form a slide crank linkage as shown in Fig 8. When part 2 is pushed forward,

the active petal, which is in part 4, will be opened. Unlike the mechanism of the active petal, the passive petals don't have a coupler, which is part 3, and connected with neighboring active petal will a rubber band instead. When the active petals are opened, the passive petal will follow the active petals as long as there is no large force hindering it.

The design makes use of the passive mechanism of one petal. The passive expansion process is illustrated in Fig 1. The passive expansion lets the other four active petals open by applying a force from rack to counter the stress of the rubber band first. Then the high-torque gear motor lifts the passive petal when the vehicle is moving forward.

This design relief the high requirement of the servo torque and transfers the task to the gear-motor which is more capable of providing high torque, when the loading is large.

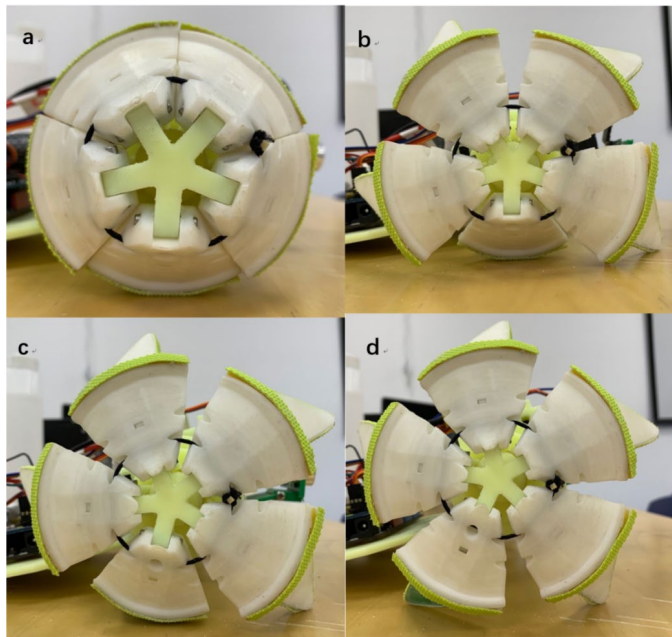


Figure 1. a) The passive petal is contacting with the table and the other four active petals are going to expand. b) The active petals are expanded, while the passive petal is not able to open. Because the rubber band only exerts limited force to the active petals and provides limited force to the passive petal to lift the vehicle. c) As the wheel turns clockwise, the space for the expansion of the passive petal becomes large. d) The wheel is fully expanded.

There are three other creative points in our design, which we call “Sand Groove”, “T-shaped Grooves” and “Tight Locking Mechanism”. The CAD plots of these parts are shown in Fig 2.

#### *Sand Grooves*

Since our vehicle will operate in the sand, it can't be avoided that some sand may go into the gap for the rack to move in and out. Chances are that, in the worst case the sand may block the movement of rack and therefore prevent the transforming mechanism to operate normally. Therefore, we designed the sand groove to remove the sand which may increase the friction and get the rack stuck.

#### *T-shaped Grooves*

To detect the surrounding of the vehicle, we need to append some distance sensors to the main body. And for different situations, we may need to adjust the position of our sensors for better precision. This means that we should design a special kind of grooves for the holders of sensors to be installed. The reason why the grooves are T-shaped is that it can reduce the degree of freedom as much as possible so that the holders can be tightly matched. The T-shaped grooves outside the main body enables us to attach sensors to the body structure and adjust them conveniently, which helps us a lot in assembly and testing process.

#### *Tight-locking Mechanism*

The last question we met with we are designing the vehicle is how to fix the motors so that they will not rotate at any case. Removing and installing the motors easily is also desired since they might be broken accidentally. Using the holders which can be found in the market is not a good choice since they may take too much space and make the vehicle wider than expected. Therefore, we decided to compact the motors into the main body to minimize the width and tightened them with the “tight locking mechanism”. This mechanism uses two screws and nuts to suppress the motors tightly so that they can not rotate due to the large friction. In that case, the motors are locked inside the body, which is shown in Fig 2.

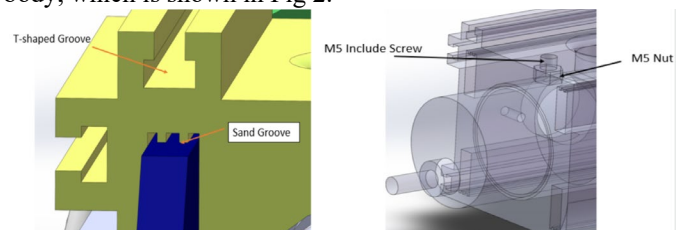


Figure 2. T-shaped groove design and tight locking mechanism.

#### *B. Graphical Linkage Synthesis*

For our design, there is a two-position synthesis, which is as shown below in Fig 3. This is a slider-crank four bar linkage mechanism. The push hoop slides along the column and pushed the coupler inside its push range. When the push hoop pushes the coupler, the crank linked to the coupler rotates about an axis which is fixed at the tip of the column. The length of the coupler is designed to match with the push range so that the crank can rotate exactly 90 degrees and reach a vertical position. In that case, our vehicle can “stand up” and complete the process of transforming.

We used SolidWorks to build the model of our linkage system and simulated the two configurations of transformable wheel. The red petals are the cranks that linked to the coupler while the purple one is the passive leg driven by elastic band which we have already introduced in the creative part. The diameter of the closed-wheel mode of our transforming wheel

is 85.12 mm, which is less than 90 mm to ensure the vehicle to run through the tunnel. The radius of the open-wheel mode of the wheel is 85.00 mm, which is more than 60mm to guarantee that the vehicle is able to climb up the terrace.

The linkage size for both expansion and shrink mode are marked and in Fig 3.

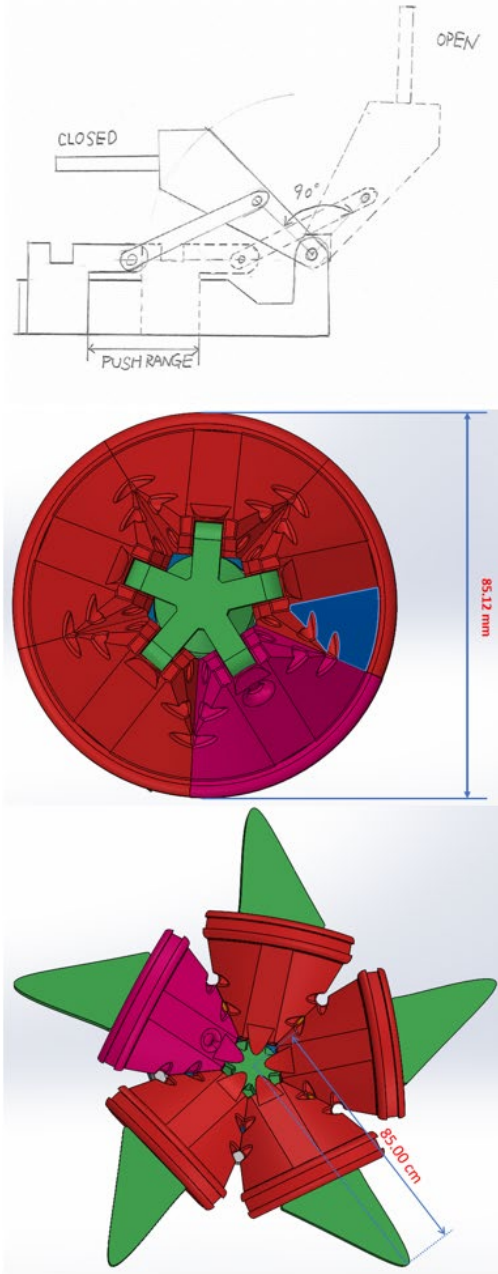
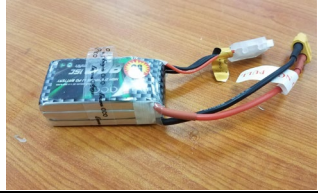

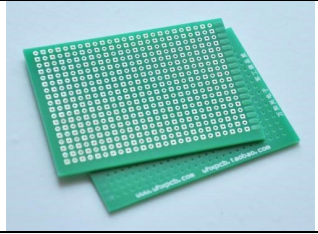

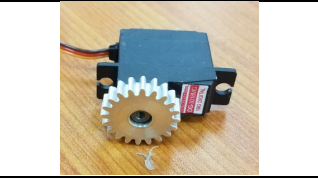

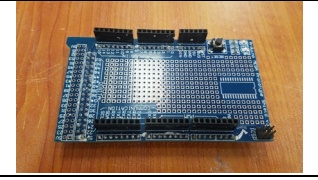
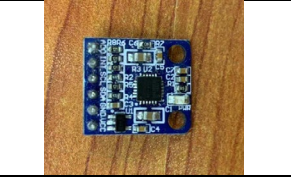
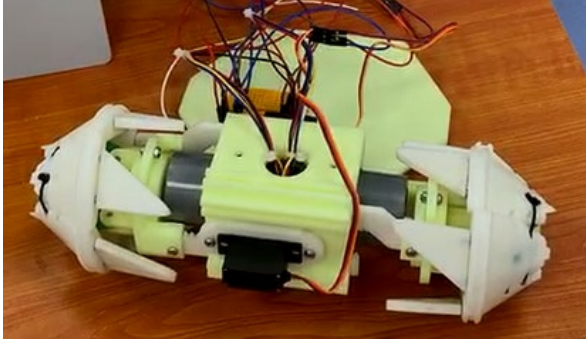


Figure 3. a) The synthesis diagram for designed linkage. b) The closed-wheel mode and its dimension. c) The open-wheel mode and its dimension.

Besides the linkage synthesis, we also bought and manufactured other components to assemble the whole vehicle. The initial prototype and its components are shown in the Table 1.

**Table 1. Electronic components and prototype**

	
Battery	Distance Sensor
	
Perf Board	Motor
	
Servo	Arduino Mega
	
Arduino Mega Shield	Gyroscope
	
Prototype	

### III. MANUFACTURING

#### A. Selection of Materials

The main body of our vehicle, as Fig 5a, requires high accuracy for the T-shaped groove and well surface finish to reduce the friction with the racks. The pushing hoops, as Fig 5b, requires good surface finish so that it can rotate and slide with little friction. The racks require high strength and good abrasive resistance to fit with aluminum gears. The rest of the parts don't have specific requirements. Based on material



properties and design requirements as analysed above, the selection of material is listed in Table 1.

Some of the material for specific parts of the vehicle is shown in Fig 4.

**Table 1. Material Selection of components**

Component	Material	Reason
Rack	Nylon	Strength ++ Weight --
Gear	Aluminum	Abrasive Resistance ++ Strength ++ Weight --
Main Body	Resin	Printing Precision ++ Friction --
Wheel	PLA	Price --

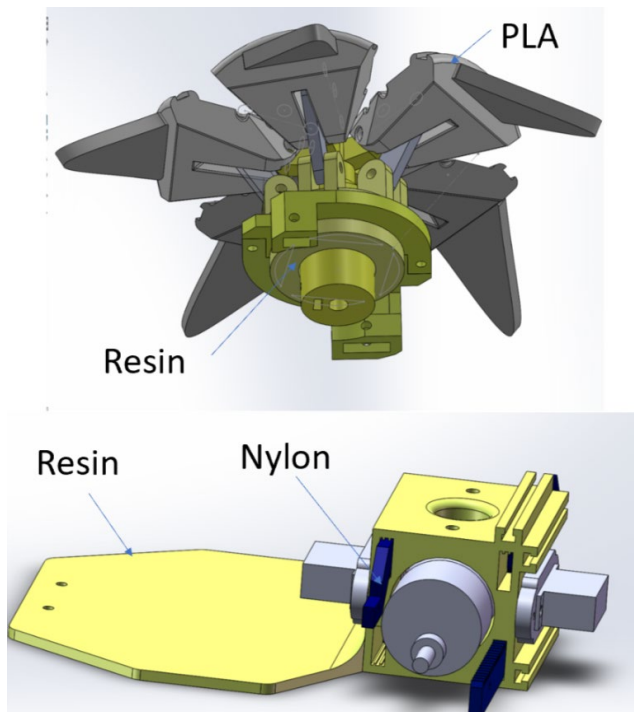


Figure 4. Material selection for the transformable wheels and the main body.

### B. Procedure of manufacturing

We used SolidWorks to draw the CAD models for the parts we need and 3-D printed them with Resin, Nylon or ABS materials. Some of the parts we manufactured with 3-D printing technology are listed in Fig 5.

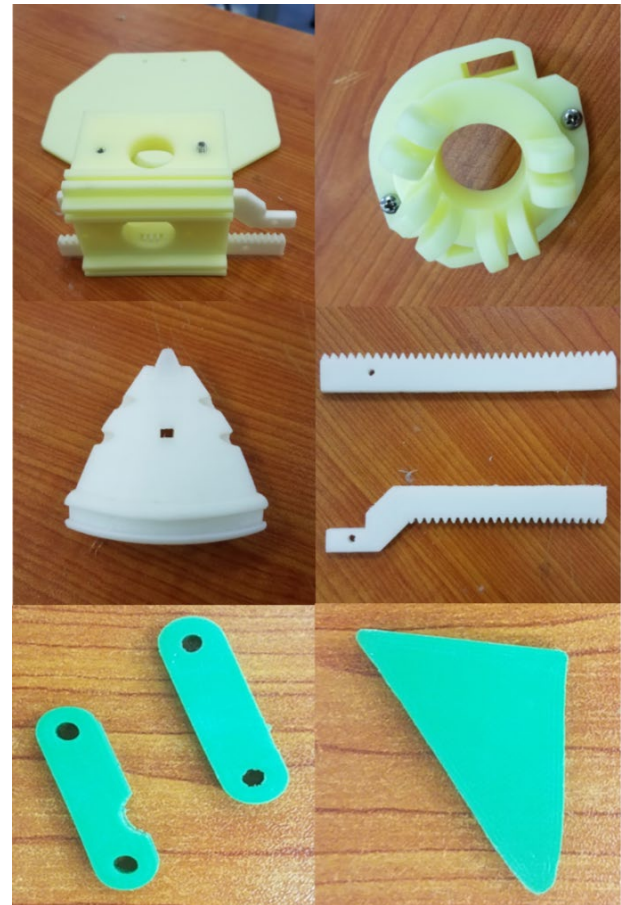


Figure 5. a) Main body with racks inserted in. b) Pushing hoops. c) The petal component. d) A pair of racks. e) The couplers. f) The horn.

### C. Procedure of assembly

#### 1. Assembly of the Transformable Wheels

First, we linked the petals with couplers using steel pins. The steel pins are smooth enough for couplers to rotate around.

Then, we used steel pins to link the petals with the holes at the tip of the central column to make a pivot. The petals will rotate around the pivot during expansion or shrinkage. The position of pins are shown in Fig (a) and the outcome is shown in Fig (b).

Thirdly, we hoped the central column with a pushing hoop and connected the couplers with the holes on the hoop with steel pins. This is the joint between the slider and the coupler. The result of assembly is shown in Fig (c).

After finishing with the linkage part, we added horns to the end of the petals and connected them with two steel pins each. The horns are used for catching the edge of the terrace to facilitate the process of climbing.

Here comes our transformable wheel, the assembly is shown in Fig (d).

#### 2. Connecting the Wheels to the Main Body

First, we put the motors into the main body from the holes on both sides.

Then, we connected the wheels and the body with racks and inserts the motor shaft into the holes on the column of wheels by inserting the shaft into the D-shaped holes on the central

column. We used screws to tighten the connections. The result is shown in Fig (e).

### 3. Connecting the Main Body and the Servos

Firstly, we use screws to attach the servo to the main body.

Then match the gears of the servo with the racks to achieve an efficient transmission as is shown in Fig 6f.

With the T-shaped groove, we can place the sensors at the front of the vehicle. The prototype is shown in Fig 6g.

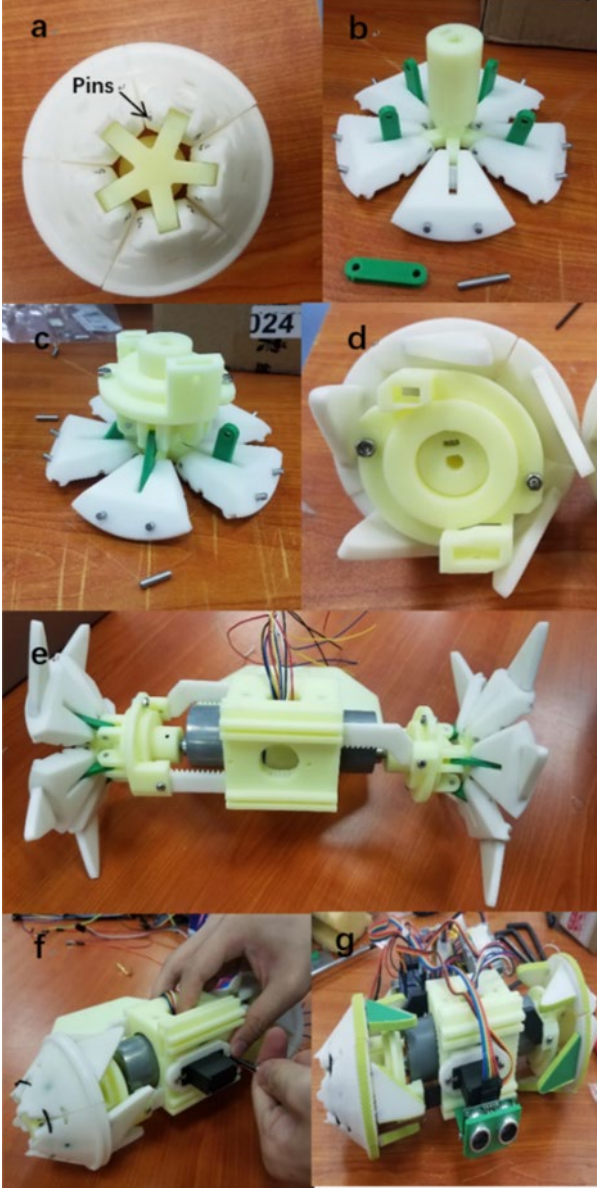


Figure 6. Assembly steps.

## IV. ELECTRONIC COMPONENTS AND PROGRAMMING

### A. Design of Electronic Circuits

After finishing with the mechanical parts of manufacturing and assembling, we came to the electronic part. As we all know, the bread boards are unreliable since the inserted wires may loosen and drop out due to all kinds of fluctuations, so we decided to make an integrated circuit

board and soldering all signal, power and ground lines together to an Arduino Shield board. The shield board is very much useful because you can directly insert it onto your Arduino Mega board so as to minimize the space.

We also soldered several pins for exchanging batteries and a switch to turn on and off with ease. These components may seem useless, but they really act an important role in the testing to save our energy and reduce unexpected accidents. The circuit diagram of all the devices and the picture of how we soldered the components together on Arduino Shield board connected with the Arduino mega board are shown below in Fig 7.

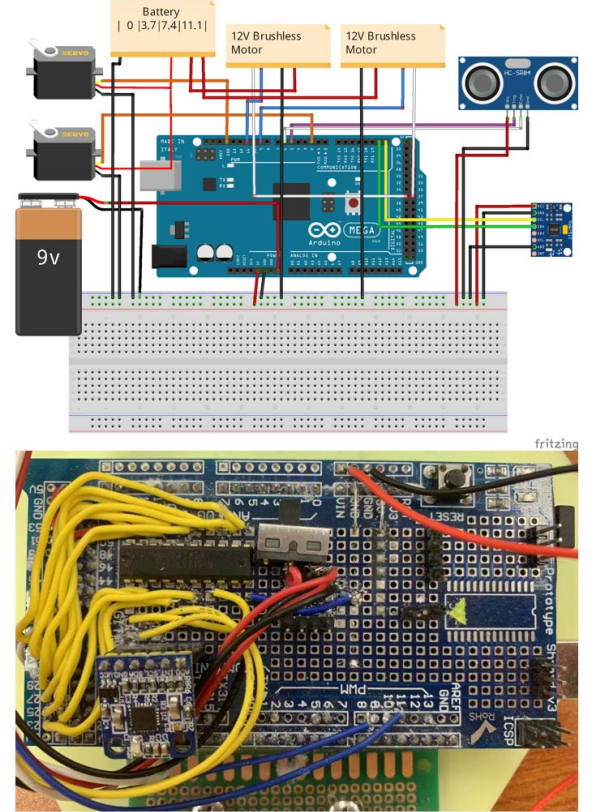


Figure 7. The circuit diagram and the soldered Arduino Shield board.

### B. Programming the Motors, Servos and Sensors

It is hard to believe that the motor we have chosen is so good that we can program it to adjust the speed of left and right motor with a closed-loop control. The advance of real-time algorithm enables our vehicle to run straight under the feedback from the gyroscope, which measures the orientation, angular velocity and angular acceleration of the vehicle. The servos and distances sensors are controlled by Arduino Mega. The gyroscope receives data of orientation and the distance sensor receives data of the distance from the front. With the feedback from the gyroscope, the left and right motor adjusts their speed of rotation automatically to ensure that the orientation always towards the initiated one. When the vehicle needs to rotate for a degree, the gyroscope can also measure the orientation accurately so as to rotate just as we expected. The distance sensor and the servos are used to determine when to transform between the modes. When the sensor detects a terrace, the Arduino Mega will give signal to the servo and use

gear-rack mechanism to open the wheels. The detailed algorithm of how our vehicle reacts to the surroundings is shown in Fig. 8.

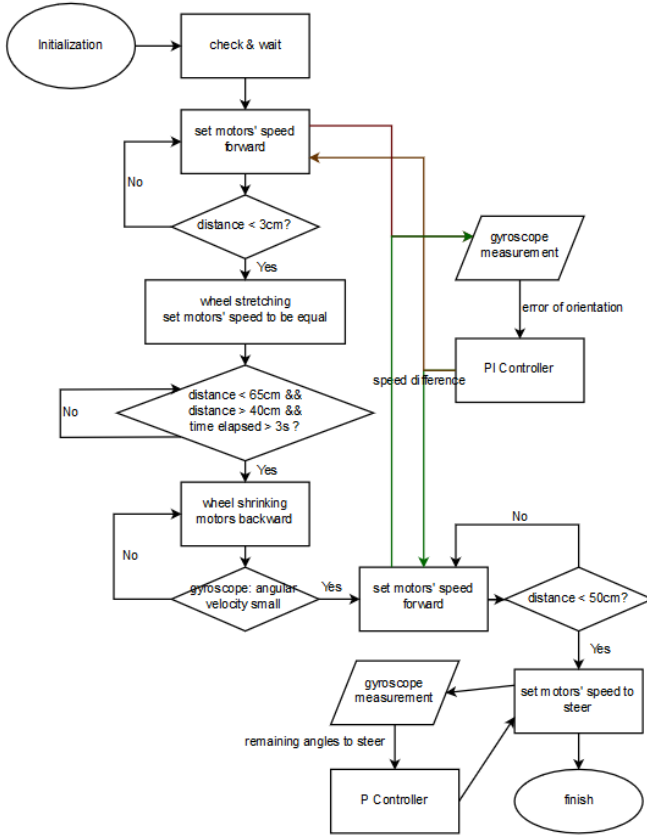


Figure 8. The flowchart of algorithm of the vehicle.

## V. ANALYSIS

### A. Classification of the Designed Linkage

For the DOF of the linkage, as is shown in figure x, we can get:

$$L=4, J_1=4, J_2=J_m=0$$

$$DOF=3*(4-1) - 2*4=1$$

For the transforming part, it is a RRRP slider-crank linkage.

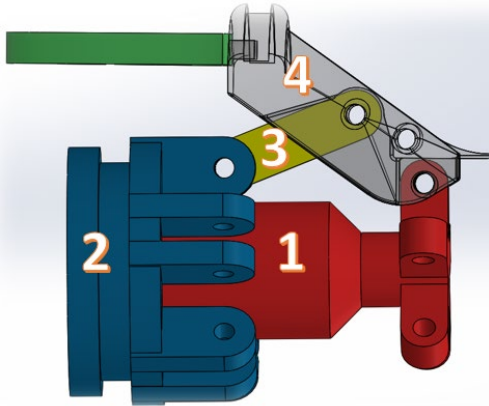


Figure 8. RRRP linkage synthesis.

### B. Position Analysis for Transformation

As is shown in Fig 9, the unknown parameter for the process of transformation is the  $\theta_3$  and the position of the slider  $d$  under the condition that we have already known the input angle  $\theta_2$ .

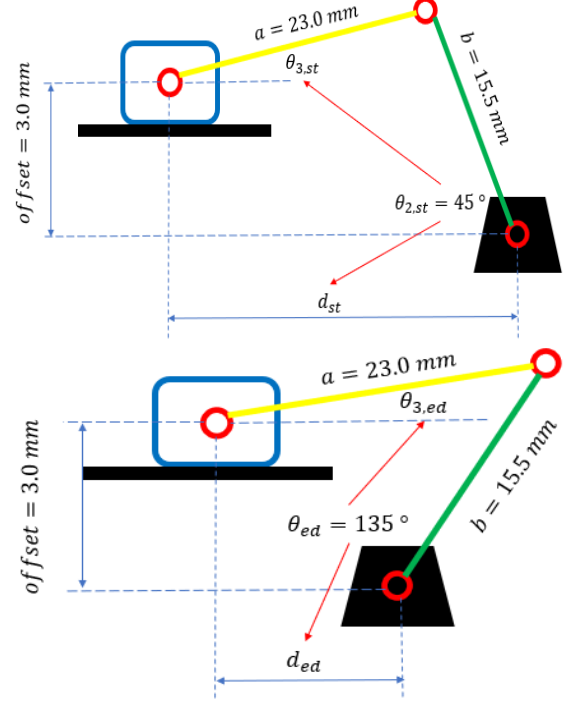
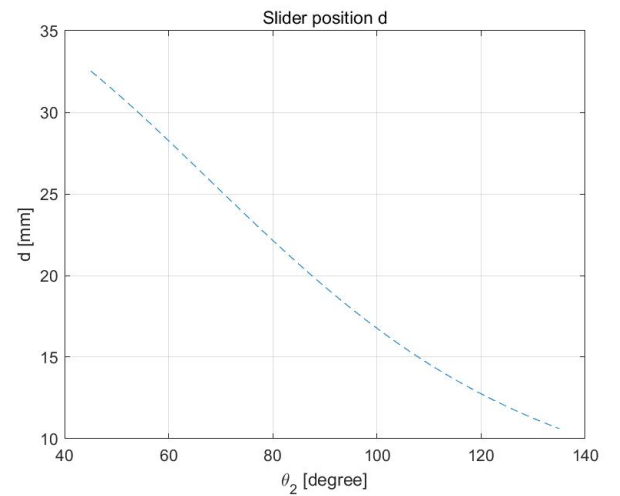


Figure 9. The position analysis for transformation.

We calculated and plotted the diagram using MATLAB. We get the following plot of  $d$  v.s.  $\theta_2$  and  $\theta_3$  v.s.  $\theta_2$ . By reading the diagram, we can figure out that the displacement  $d$  varies from 10.1 mm to 33.0 mm while the angular displacement  $\theta_3$  varies from 20.1 degrees to 32.9 degrees. The plot is shown in Fig 10. We can see that the slope of displacement change is relatively constant and the angular displacement is symmetric about the middle point, which represents that it is a very good design to even the force.





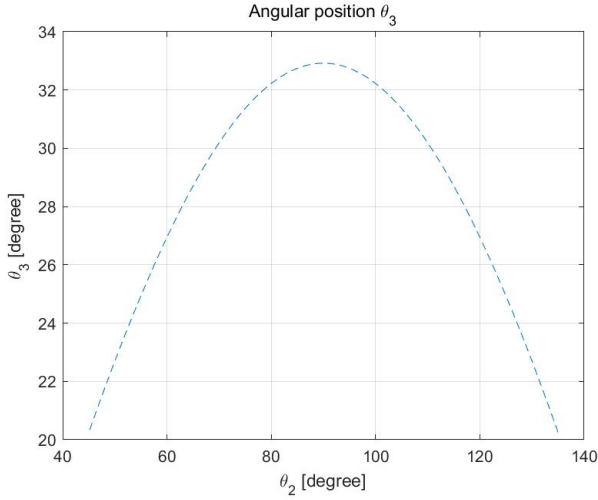


Figure 10. The plot of horizontal and angular displacements and input angle for position analysis.

### C. Force Analysis for Transformation

First, we can draw free body diagrams for both expanding and shrinking processes. The linkage is subjected to three forces basically: the friction force with the ground, the normal force for supporting and the pushing force from the pushing hoop. Both free body diagrams are shown in Fig 11.

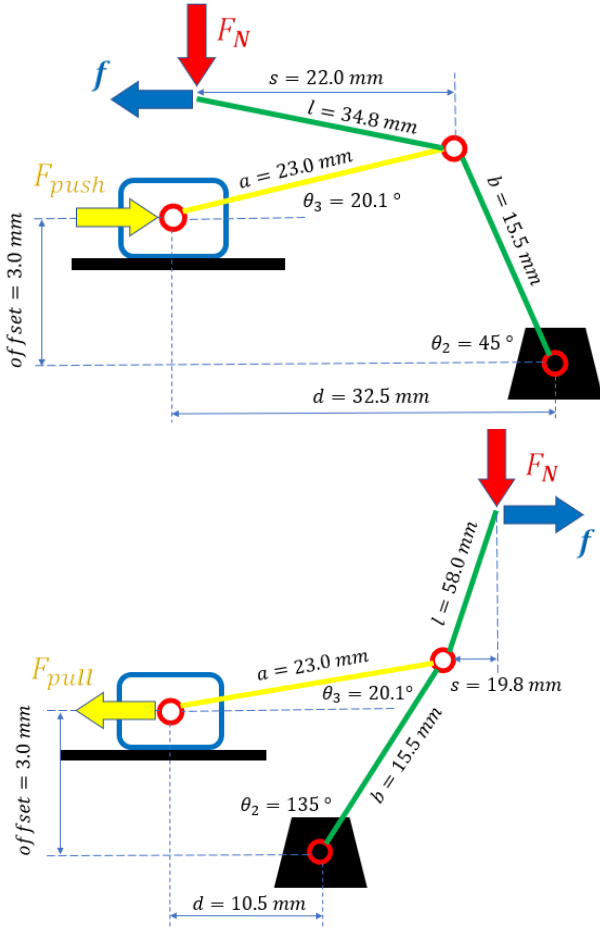


Figure 11. Free body diagram for expanding and shrinking.

Then we can use the vector method to calculate the minimum torque we need for the expanding process:

$$A = \begin{pmatrix} 15.5 * \cos\theta_2 \\ 15.5 * \sin\theta_2 \\ 0 \end{pmatrix}; B = \begin{pmatrix} 15.5 * \cos\theta_2 + 34.8 * \cos(\theta_2 + 5.82^\circ) \\ 15.5 * \sin\theta_2 + 34.8 * \sin(\theta_2 + 5.82^\circ) \\ 0 \end{pmatrix}; F = \begin{pmatrix} 2.4 \\ -4.9 \\ 0 \end{pmatrix};$$

$$T = r \times F = \begin{bmatrix} i & j & k \\ B_x & B_y & 0 \\ f & -F & 0 \end{bmatrix} = \begin{pmatrix} 0 \\ 0 \\ -4.9 * [15.5 * \cos\theta_2 + 34.8 * \cos(\theta_2 + 5.82^\circ)] - 2.4 * [15.5 * \sin\theta_2 + 34.8 * \sin(\theta_2 + 5.82^\circ)] \end{pmatrix}$$

$$\text{The required torque } T = 4.9 * [15.5 * \cos\theta_2 + 34.8 * \cos(\theta_2 + 5.82^\circ)] + 2.4 * [15.5 * \sin\theta_2 + 34.8 * \sin(\theta_2 + 5.82^\circ)] \text{ (N * mm)}$$

We calculated and plotted the diagram using MATLAB. We get the following plot of  $T$  v.s.  $\theta_2$ . By reading the diagram, we can figure out that the torque  $T$  varies from 250 N\*mm to (-100) N\*mm where the minus sign means the opposite direction and there is no need to push. We can see that the maximum torque we need is 250 N\*mm in pushing process. The plot is shown in Fig 12.

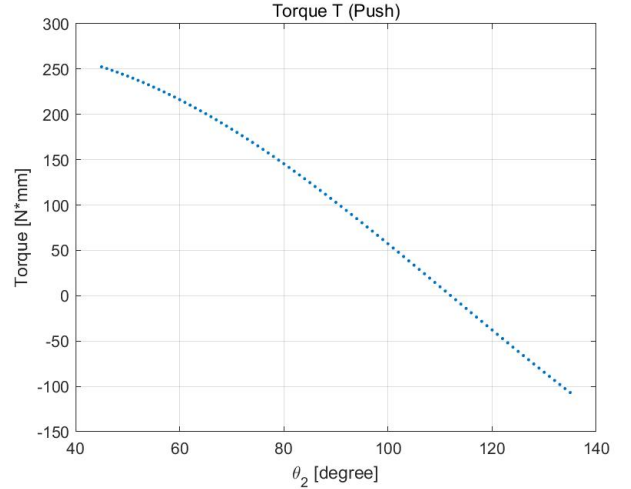


Figure 12. The plot for the torque and input angle for force analysis in expanding process.

At last we use the same method to calculate the minimum torque we need for the shrinking process:

$$A = \begin{pmatrix} 15.5 * \cos\theta_2 \\ 15.5 * \sin\theta_2 \\ 0 \end{pmatrix}; B = \begin{pmatrix} 15.5 * \cos\theta_2 + 58 * \cos(\theta_2 - 29.13^\circ) \\ 15.5 * \sin\theta_2 + 58 * \sin(\theta_2 - 29.13^\circ) \\ 0 \end{pmatrix}; F = \begin{pmatrix} -2.4 \\ -4.9 \\ 0 \end{pmatrix};$$

$$T = r \times F = \begin{bmatrix} i & j & k \\ B_x & B_y & 0 \\ -f & -F & 0 \end{bmatrix} = \begin{pmatrix} 0 \\ 0 \\ -4.9 * [15.5 * \cos\theta_2 + 58 * \cos(\theta_2 - 29.13^\circ)] + 2.4 * [15.5 * \sin\theta_2 + 58 * \sin(\theta_2 - 29.13^\circ)] \end{pmatrix}$$

$$\text{The required torque } T = 4.9 * [15.5 * \cos\theta_2 + 58 * \cos(\theta_2 - 29.13^\circ)] - 2.4 * [15.5 * \sin\theta_2 + 58 * \sin(\theta_2 - 29.13^\circ)] \text{ (N * mm)}$$

We calculated and plotted the diagram using MATLAB. We get the following plot of  $T$  v.s.  $\theta_2$ . By reading the diagram, we can figure out that the torque  $T$  varies from 260 N\*mm to (-300) N\*mm where the minus sign means the opposite direction and there is no need to pull. We can see that the maximum torque we need is 260 N\*mm in pulling process. The plot is shown in Fig 13.

Comparing the two maximum torque that we have calculated above, we can draw the conclusion that in order to meet the demands of both pulling and pushing, the minimum torque we require is 260 N\*mm. Therefore, it is appropriate

for us to choose a servo that can provide a torque of 3.5 kg\*cm (343 N\*mm).

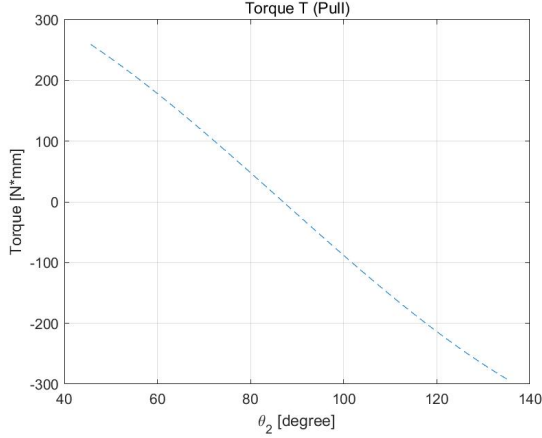
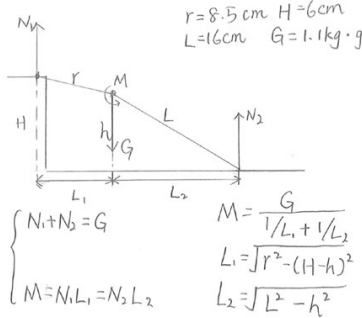


Figure 13. The plot for the torque and input angle for force analysis in shrinking process.

We also want to know where should we put our motors to generate a torque that is big enough for climbing up the barrier, so we built a physical model for the climbing procedure, measured the dimensions of the open-wheel mode and calculated the required torque according to the free body diagram. The calculating process is shown as follows.



We plotted the relationship between the torque and the height of the motors with the results we derived above. We placed our motor at 8 cm high from the ground, and the two motors we have chosen can provide a total torque of 22 kg\*cm at 25 rpm, which is sufficient to provide the required torque of 5.8 kg\*cm from our calculation. The plot is shown below in Fig 14.

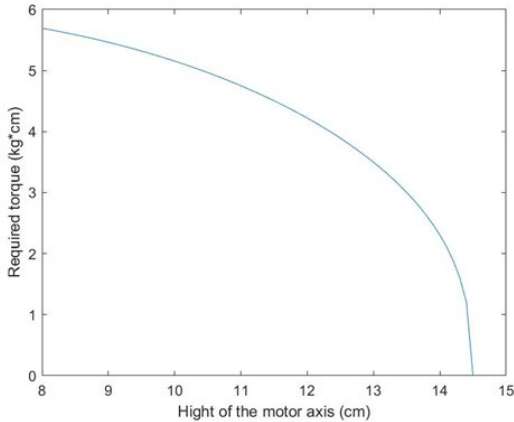


Figure 14. The plot for the required torque and the height of the motors for force analysis in expanding form.

#### D. Rolling Speed Analysis

The translational speed of our vehicle on flat surface and sand pitch was estimated based on the spec sheet of the gear motor and several assumptions. The translational speed is calculated as

$$u = R_{effective} \times \omega_{wheel} \quad (1)$$

where  $u$  is the translational speed,  $R_{effective}$  is the effective radius of the wheel that applies to the non-slip condition and  $\omega_{wheel}$  is the angular velocity of the wheel. According to the spec sheet of the gear-motor, the no-load rotational speed is 60 RPM. Since the load torque on the gear-motor is not very large, the  $\omega_{wheel}$  was assumed to be 60 RPM constant. For flat surface condition, the effective radius of the contracted wheel is the radius of the contracted wheel which is 4.4 cm. The effective radius of the expended wheel is assumed to be the distance from the center to the rare tip of the wheel which was measured as 8.3 cm. For sand pitch condition, the effective radius of the expended wheel is assumed to be the distance from the center to fat end of the petal which was measured as 6.1 cm. Because the horns planted on the fat end of the petals are thin and therefore will cut through the sand. On the other hand, the fat end of the petals will provide good support to the vehicle. Apply in Equation 1, the calculated speed is listed in parentheses in Table 2

## VI. EXPERIMENT

#### A. Load Capacity

The load capacity is defined as the maximum load the vehicle can carry without functional problems. Due to limited space to place weights, the maximum weight put on the vehicle is 3kg. The big black weight placed on the top of the vehicle weighs 2 kg and ten strip weights each weight 100g were piled on the vehicle in Fig 15. Under this weight, the vehicle succeeded in passive wheel expansion, climbing the barrier and passing through the sand pitch which proved that the load capacity of this vehicle is at least 3 kg. According to the analysis, the maximum torque required for gear-motors is the torque required to climb onto the barrier. The theoretical maximum weight of the vehicle is 6.25 kg. Subtracted by the weight of the vehicle (1.16 kg), the theoretical maximum load is about 5 kg. Our vehicle shows excellent load-bearing ability.

For active wheel expansion powered by servos, the maximum load was found as 0.7 kg in Fig 15 and thus the total weight of the vehicle is 1.86 kg. The theoretical maximum weight of the vehicle constrained by servo torque is 1.8 kg, which matches with the experimental value.



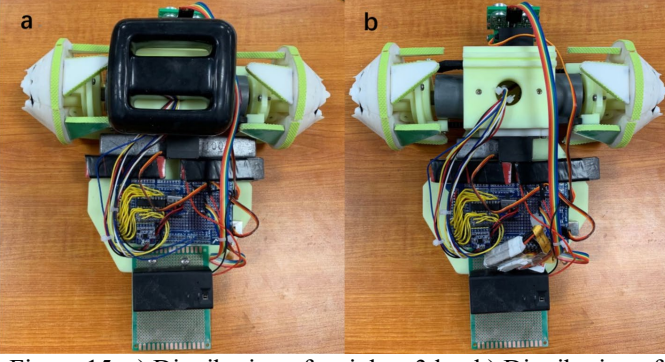


Figure 15. a) Distribution of weights: 3 kg. b) Distribution of weights: 700 g.

### B. Mobility

The mobility was evaluated by the speed of our vehicles running on the flat surface and sand pitch, and time it requires to climb the barrier. During the experiment, the voltage of the battery ranged from 12.4 V to 12.0 V, the PWM was set as full-duty and no load was put on the vehicle. The measured speeds are listed in Table 2 and the expected speed was included in parenthesis. The measured speed of the vehicle with expended wheel running on flat surface deviated from the expected value. Because during the test, the tip of the wheel is slipping on the floor, which violated the non-slip assumption. Similarly, when the expended wheel was running in the sand pitch in Fig 16, the tip of the wheel is slipping on the floor under the sand since the depth of the sand pitch is smaller than the expended wheel radius. Due to the force that sand exerted on the wheel, the speed of our vehicle with expended wheel running in sand pitch was slightly larger than running on a flat surface.

Time elapsed for our vehicle to climb the barrier was 0.5 seconds in Fig 16.

**Table 2. Measured and Expected speeds of the vehicle**

	Sand pitch	Flat surface
<b>Contracted</b>	\	0.27 (0.276) m/s
<b>Expended</b>	0.37 (0.383) m/s	0.30 (0.522) m/s

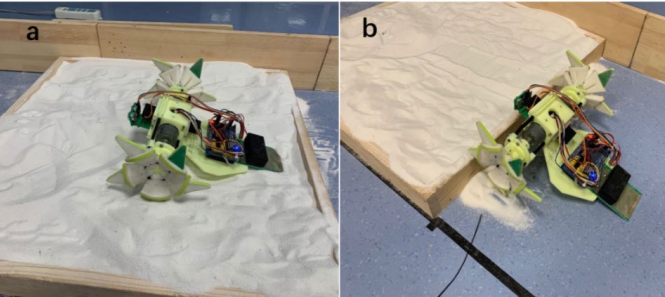


Figure 16. a) When the vehicle is running on the sand pitch, the tips of the wheel contact with the ground. b) The vehicle is climbing on the barrier.

### C. Wheel Transformation

The contracted wheel radius was measured as 4.4 cm and the expended wheel radius was measured as 8.3 cm, which agrees with analysis.

## VII. DISCUSSION

### A. Excellent Load Capacity

The most innovative design of Yu's vehicle from Ohio State is the passive petal mechanism that enable the expansion of the wheel under large load. The ratio between the maximum weight of the vehicle and load is 2.59 for our vehicle and 1.81 for Yu's vehicle since our gear-motor has a larger maximum torque. Both of the vehicles have shown excellent load carrying ability.

### B. Speed and stability of vehicle

Apart from load capacity, the speed and stability is also an important aspect of the performance of a vehicle, while Yu's paper didn't analysis and experiment the speed performance of their vehicle. We found out that when the expended wheel is running on a flat surface, the vehicle vibrates violently and make it hard to control the direction. Also, slipping between the tip of petal and hard flat surface makes the efficiency of the vehicle low. The origin of these issues is the dynamics of our vehicle.

We proposed some possible improvements to solve the stability problems.

- 1) Choosing material with moderate stiffness and large energy dissipating rate to build the horns on the petals.
- 2) Synchronizing the left wheel and left wheel to harmonize the vibration.
- 3) Sophisticated control on the rotation speed of both wheels. The idea is illustrated in Fig. 17. The speed of the wheels should be increased after the horn marked by black just touches the ground to move the vehicle forward and overcome gravity. When the horn marked by black is vertical, the horn marked by gray is going to hit the ground. The wheel speed should be decreased to compensate the released gravity potential energy in order to prevent a large impulse exerted on the horn marked by gray which will make the wheel lose contact with the ground.

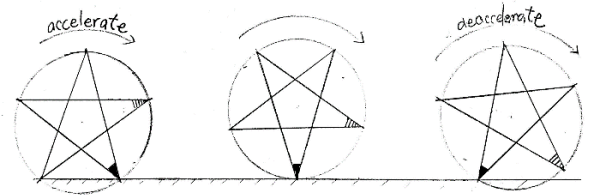


Figure 17. Illustration on the rotation of expended wheel.

### A. Intellectual merit

Our research confirmed the usefulness of Yu's passive expansion mechanism. What's more, we found a major stability issue of petal like wheels and proposed some solutions which started a new research direction of transformable wheel robotics.

### B. Practical implication

Vehicles implemented with transformable wheels are capable of crossing different territories. Our work promotes the development of this technology. In future, transformable wheels can be applied to off-road vehicles to enjoy both high efficiency on road and excellent compatibility to extreme conditions. The transformable wheel can be also applied to Rovers to explore other planets.

### VII. CONCLUSION

In order to further develop the passive mechanism of transformable wheel created by Yu, we built and tested a vehicle that applied similar mechanism. We also applied new tight locking mechanism and T-shaped grooves to our vehicle for rapid iteration. During the experiment, the maximum load and self-weight is found as 2.59 which agree with Yu's claim that the passive expansion mechanism is capable of carrying a heavy load. However, we found that the speed of our vehicle with expended wheels is much lower than the expected value, which is caused from slipping and violent vibration. In order to alleviate the vibration and slipping, we proposed the following solutions:

- Choosing material with moderate stiffness and large energy dissipating rate to build the horns on the petals.
- Synchronizing the left wheel and left wheel to harmonize the vibration.
- Sophisticated control on the rotation speed of both wheels.

These solutions open a new direction of this family of transformable wheels. We also learned some lessons from manufacturing and adjusting our vehicle:

- Simplicity promises reliability.
- Much attention needs to be paid on gear pairing.
- Multiple servos should not be applied to overconstrained mechanism.
- Magnetic compass is quite sensitive to the magnetic field and it should be used in open space.

### REFERENCES

- [1] Y.-S. Kim, G.-P. Jung, H. Kim, K.-J. Cho, and C.-N. Chu, "Wheel Transformer: A Wheel-Leg Hybrid Robot With Passive Transformable Wheels," *IEEE Transactions on Robotics*, vol. 30, no. 6, pp. 1487–1498, 2014.
- [2] L. Bai, J. Guan, X. Chen, J. Hou, and W. Duan, "An optional passive/active transformable wheel-legged mobility concept for search and rescue robots," *Robotics and Autonomous Systems*, vol. 107, pp. 145–155, 2018.
- [3] Y. She, C. J. Hurd, and H.-J. Su, "A transformable wheel robot with a passive leg," 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015

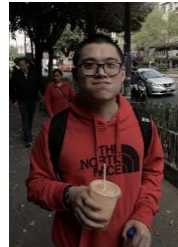
### Budget

Item	Quantity	Total price [RMB]
M2.5/3/4, Screws	300	19.4
9V Battery box	1	2.08
Pegboard	2	3.46
Pin rows	150	3.14
Switch	1	1.42
Potentiometer	10	4.62
Servo	4	156
7.4V battery	1	39
Pins	40	4.6
Hexagonal wrench	20	3.3
Motor	2	168
3D printing		770
Total		1175.02

### CONTRIBUTION OF EACH TEAM MEMBER



Wu Jiajin, designed and draw the CAD plot, assembled the prototype.



Lin Yuechuan, designed and draw the CAD plot, assembled the prototype.



He Zhirui, designed the circuit, did programming.



Sun Tianxing, designed the circuit, did programming.

## ARDUINO CODE

```

#include <Wire.h>
#include <HMC5883L.h>
#include <math.h>
#include <stdlib.h>
#include "VM350Final.h"
#include <Servo.h>

//define pins
#define LOWER_RANGING_TRIG 3
#define LOWER_RANGING_ECHO 4
#define UPPER_RANGING_TRIG 5
#define UPPER_RANGING_ECHO 6
#define REAR_RANGING_TRIG 7
#define REAR_RANGING_ECHO 8
#define LEFT_GEAR_MOTOR_PWMTIMING 9
#define LEFT_GEAR_MOTOR_FGSIGNAL 10
#define RIGHT_GEAR_MOTOR_PWMTIMING 11
#define RIGHT_GEAR_MOTOR_FGSIGNAL 12
#define LEFT_SERVO 2
#define RIGHT_SERVO 13
#define SDA 20
#define SCL 21//MEGA: SDA 20 SCL 21, UNO/NANO:
SDA A4 SCL A5

ultrasonicRanging
upperRanging(UPPER_RANGING_TRIG,UPPER_RANGIN
G_ECHO);

Servo leftServo;
Servo rightServo;
gearMotor
leftMotor(LEFT_GEAR_MOTOR_PWMTIMING);
gearMotor
rightMotor(RIGHT_GEAR_MOTOR_PWMTIMING);
//compass hmc5883l;
gyroscope mpu_6050;
PIController straightLine(0.72,0.09);//kp and ki
PIController steering(0.85,0);
timeMachine timer;
int state=0;
float motorSpeedDifference=0;

unsigned long stretchTime;
unsigned long adjustTime;
float currentOrientation_hmc5883l = 0;
float currentOrientation_mpu6050 = 0;

void setup() {
    // put your setup code here, to run once:
    pinMode(36,OUTPUT);
    pinMode(37,OUTPUT);
    digitalWrite(36,LOW);
    digitalWrite(37,HIGH);
    Serial.begin(9600);
    Serial.println("begin.");
    upperRanging.begin();
    leftMotor.begin();
    rightMotor.begin();

```

```

    leftMotor.setSpeed(0);
    rightMotor.setSpeed(0);
    delay(500);
    leftServo.attach(2);
    rightServo.attach(13);
    //hmc5883l.begin();
    mpu_6050.wakeUp();
    leftServo.write(0);
    rightServo.write(0);
    timer.begin();
    Serial.println("setup finish.");
}

void loop() {
    switch(state){
        case 0://准备阶段
            upperRanging.measure();
            //hmc5883l.measure();
            timer.readTime();
            Serial.println(upperRanging.getDistance());
            if(timer.getTime(>10000000) {
                state=1;
                timer.readTime();
                Serial.println("The vehicle can
start.");
                mpu_6050.calibrate();
                mpu_6050.measure();
                mpu_6050.measure();
                delay(300);

                leftMotor.setSpeed(50);
                rightMotor.setSpeed(50);
                //hmc5883l.calibrate();

                delay(500);
            }
            break;
            case 1://直行阶段 1
                upperRanging.measure();
                //hmc5883l.measure();
                mpu_6050.measure();
                timer.readTime();

                currentOrientation_mpu6050+=mpu_6050.getMeasur
ementData(OMEGA_Z)*((float)timer.getInterval()/1000000);

                //currentOrientation_hmc5883l=hmc5883l.getOrienta
tion();

                motorSpeedDifference =
straightLine.getResult(currentOrientation_mpu6050,((float)ti
mer.getInterval())/1000000);

                leftMotor.setSpeed(50+motorSpeedDifference);
                rightMotor.setSpeed(50-
motorSpeedDifference);

                if(upperRanging.getDistance(>1000||upperRanging.
getDistance(<3)){
                    //车停止, PI 清零
                    leftMotor.setSpeed(0);

```



```

rightMotor.setSpeed(0);
straightLine.clear();
motorSpeedDifference=0;
delay(800);
//轮子展开
leftServo.write(130);
rightServo.write(130);
delay(1300);
//电机全速转动上坡
leftMotor.setSpeed(60);
rightMotor.setSpeed(60);
//
state=2;
timer.readTime();
stretchTime=timer.getTime();
Serial.println("The vehicle detects
barrier.");
}
break;
case 2://爬坡与走沙地阶段
upperRanging.measure();
timer.readTime();

if(upperRanging.getDistance()<65&&upperRanging.
getDistance()>30&&(timer.getTime()-
stretchTime)>4000000){
    leftMotor.setSpeed(0);
    rightMotor.setSpeed(0);
    straightLine.clear();
    motorSpeedDifference=0;
    delay(800);
    leftServo.write(0);
    rightServo.write(0);

    delay(1300);
    leftMotor.setSpeed(50);
    rightMotor.setSpeed(50);
    delay(400);
    digitalWrite(37,LOW);
    digitalWrite(36,HIGH);

    delay(700);
    leftMotor.setSpeed(60);
    rightMotor.setSpeed(60);
    state=3;
    timer.readTime();
    adjustTime=timer.getTime();
    Serial.println("The vehicle down
the step.");
}
break;
case 3:
upperRanging.measure();
//hmc5883l.measure();
//mpu_6050.measure();
timer.readTime();

//currentOrientation_hmc5883l=hmc5883l.getOrienta
tion();
//Serial.println(currentOrientation_hmc5883l);

//currentOrientation_mpu6050+=mpu_6050.getMeas
urementData(OMEGA_Z)*((float)timer.getInterval()/1000000
);
//motorSpeedDifference =
straightLine.getResult(currentOrientation_hmc5883l,((float)ti
mer.getInterval())/1000000);
leftMotor.setSpeed(50);
rightMotor.setSpeed(50);
if((timer.getTime()-adjustTime)>3000000){
    straightLine.clear();
    currentOrientation_mpu6050=0;
    motorSpeedDifference=0;

    leftMotor.setSpeed(0);
    rightMotor.setSpeed(0);
    digitalWrite(36,LOW);
    digitalWrite(37,HIGH);
    delay(800);

    leftMotor.setSpeed(50);
    rightMotor.setSpeed(50);
    delay(300);

    state=4;
    timer.readTime();
    Serial.println("Begin to forward
again.");
}
break;
case 4:
upperRanging.measure();
mpu_6050.measure();
timer.readTime();

currentOrientation_mpu6050+=mpu_6050.getMeasurementDa
ta(OMEGA_Z)*((float)timer.getInterval()/1000000);
motorSpeedDifference =
straightLine.getResult(currentOrientation_mpu6050,((float)ti
mer.getInterval())/1000000);
leftMotor.setSpeed(50+motorSpeedDifference);
rightMotor.setSpeed(50-motorSpeedDifference);
if(upperRanging.getDistance()<60){
    straightLine.clear();
    motorSpeedDifference=0;
    delay(800);
    state=5;
    timer.readTime();
    Serial.println("Begin to steer.");
}

case 5:
upperRanging.measure();
//hmc5883l.measure();
mpu_6050.measure();
timer.readTime();

//currentOrientation_hmc5883l=hmc5883l.getOrienta
tion();

currentOrientation_mpu6050+=mpu_6050.getMeasur
ementData(OMEGA_Z)*((float)timer.getInterval()/1000000);

```

```
        motorSpeedDifference =  
steering.getResult(currentOrientation_mpu6050-  
87,((float)timer.getInterval())/1000000);  
  
        leftMotor.setSpeed(50+motorSpeedDifference);  
        rightMotor.setSpeed(50-  
motorSpeedDifference);  
        break;  
        default:  
        break;  
    }  
}
```