Integrated Mechatronic Project

Group D
Group members: Filippo Dal Ben, Alemseghed Ghebrezghi, Donghao Qiu, Zhaoxi Yu, Yangyi Shi
Date report written: 04/2018

UNIVERSITY OF BIRMINGHAM

INTRODUCTION

This report describes how the integrated design project has been accomplished: designing, building and demonstrating a wire following autonomous vehicle that almost elegantly follow a wire of 10mA on a course able to turn no more than 90 degrees.

The vehicle has been built over the period from the end of the second semester to the final race (26^{th} of April).

The first weeks of projects have been dedicated to some technical speed dating sessions and activity sessions, such as Sensor Coil design: for the signal changing every time we were positioning to the left, right, up or down of the wire; Signal Processing with different filters; High-voltage Construction: for circuit construction and the safety from the vehicle, because **at any point during the project may the voltage be greater than 20V be exposed**, for this reason robust power supply model has been designed.

AIMS

- Build a vehicle, which follows 10mA wire on a course with turns, turns not more severe than 90 degrees
- Working in a group of 6
- Time frame – accomplished the final design by the final race 26^{th} of April 26
- Budget £40.00

OBJECTIVES

- Develop a management plan detailing tasks to achieve and tasks that should be undertaken by each team member
- Design how the robot following the given specification
- Order components needed
- Test all the components given and purchased (within the budget)
- Make sure the vehicle causes no arm during the construction time and during the test/final race
- Vehicle must be finished before the final race deadline
- Report containing how this project has been accomplished and technical details about the vehicle

REQUIREMENTS

- Vehicle should be able to follow 10mA wire on a course and stay on the course
- Vehicle should be able to follow the course faster than all the other vehicles
- Vehicle should be able to turn not more severe than 90 degrees
- External power supply shouldn't be greater than 30V

FITNESS FOR PURPOSE

Our vehicle can sense the current through thanks to the magnetic field produced by the wire and detected by the two coils placed at the front by the universal wheel. This signal is the converted by the Arduino board and transformed in data that will give to our robot direction, on the position of the car in respect with the wire. The car is able to move thanks to a control loop that gives direction and inputs when a turn is encounter, so that it is possible to turn right or left following the direction of the wire.

SYSTEM ARCHITECTURE

I. The electrical structure of the vehicle

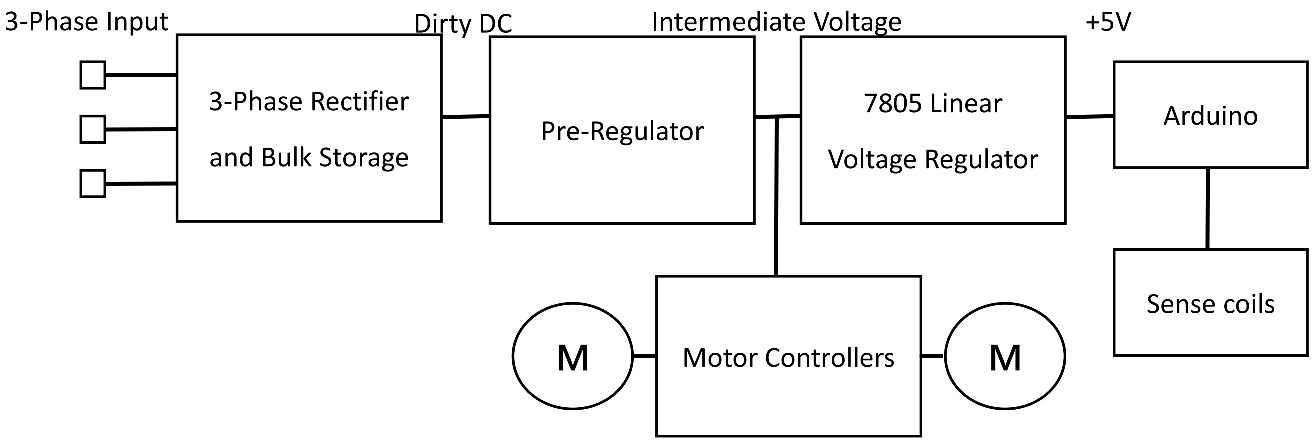


Figure 1: system structure

The electrical part of system structure is illustrated in above figure. Because all the subsystems (Microprocessor System, DC motor Controllers) require DC voltage supply, a 3-Phase rectifier should be introduced to convert the 3-Phase input to a dirty DC output. The dirty DC obtained at the output of the bridge rectifier can vary rapidly between zero and the maximum applied input potential. In order to smooth this output, a large-value bulk-storage capacitor is used following the rectifier. Then the regulator should be introduced to step down the DC voltage because of a high-purity DC +5V supply required to supply the microprocessor. Considering the absolute maximum input voltage of a 7805 regulator(+35V) which is less than the supply voltage for the vehicle and the absolute maximum temperature of regulator (60 ℃), we cannot use a single voltage regulator to derive our microprocessor power supplies. Thus, a pre-regulator is placed before the 7805 regulator to roughly step down the dirty dc voltage to an intermediate voltage. This intermediate voltage is used to supply the +12V motors at the same time. Two heatsinks are placed with two regulators respectively because voltage regulators operate by dissipating the required voltage drop as heat. Two sense coils are connected to the Arduino(Microprocessor) directly to transmit the induced current due to the magnetic field around the wire on the ground. The amount of the induced current indicates the magnetic field strengths. We use this value to adjust the direction of the robot car.

II. The body structure of the vehicle

- Components: 3mm Acrylic board, Aluminum mesh, 2*motor, 2*wheel, a universal wheel
- The main body of the vehicle is a piece of 3mm rectangle Acrylic board. All other components are placed on this board. Two sense coils are positioned at the front of the board as well as the universal wheel. Two driving wheels are placed in the back of the car. The Aluminum mesh is used to fix the motor and other components on the board.

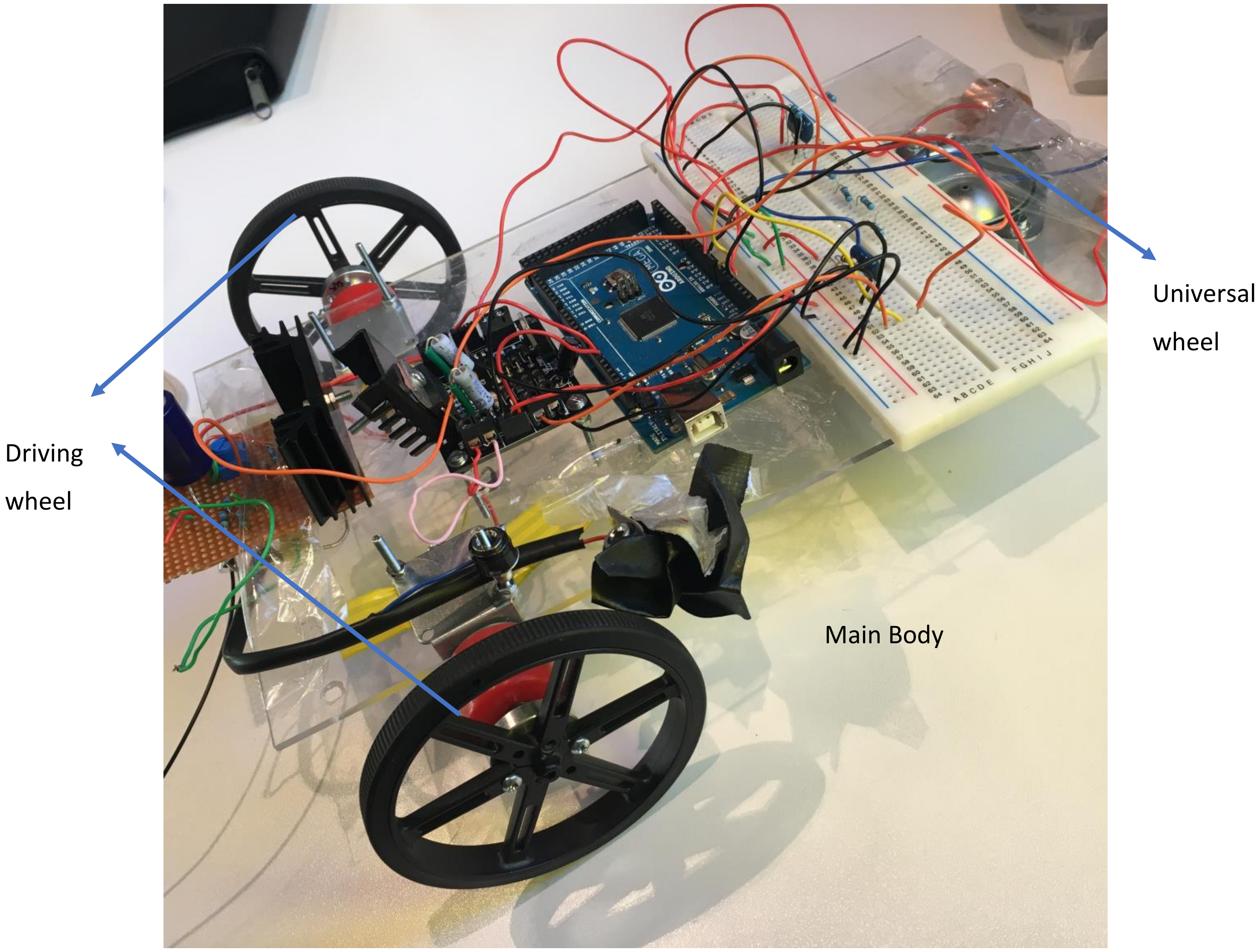


Figure 2: the body structure of the vehicle

Power Conversion and Supply Conditioning

I. First design:

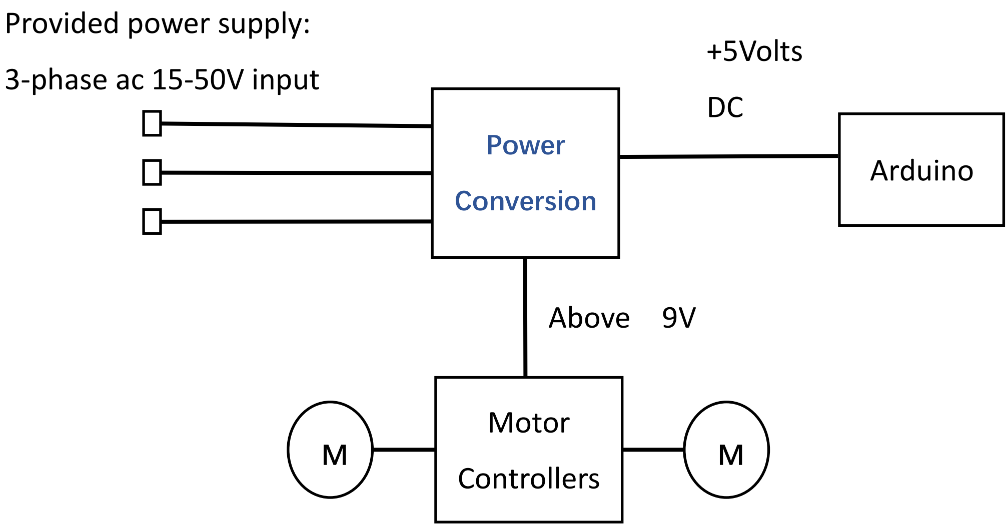


Figure 3: power conversion

The provided power source is a 3-phase ac 15-50V input, but what we need is a +5V DC voltage and an +9V DC voltage to supply the Arduino and motor controllers separately. Thus, we need to build a power conversion module to obtain the desired voltage.

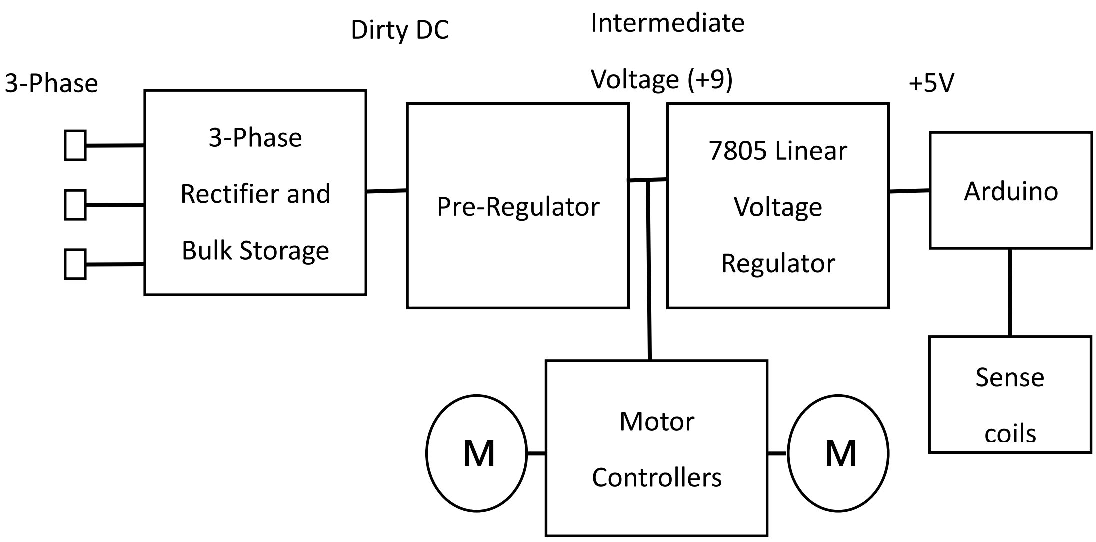


Figure 4: power unit-first design

- 3-phase rectifier

  First, we need to convert 3-phase AC input to a DC output using a 3-phase rectifier. As each motor of our vegucle can draw at least 2A, we use six, 3Amp diodes which can bear such high current.

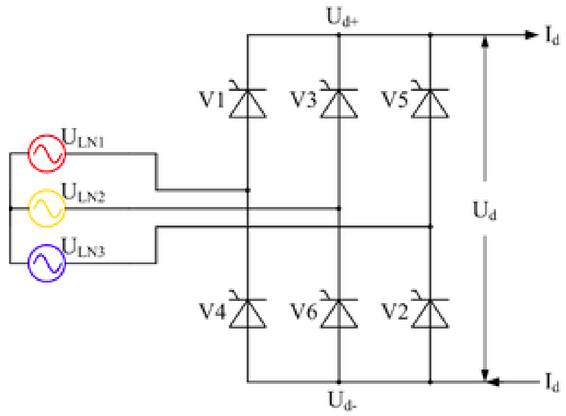


Figure 5: the legend diagram of 3-phase rectifier

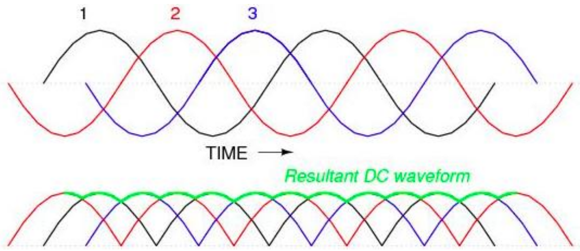


Figure 6: the resultant waveform of 3-phase rectifier

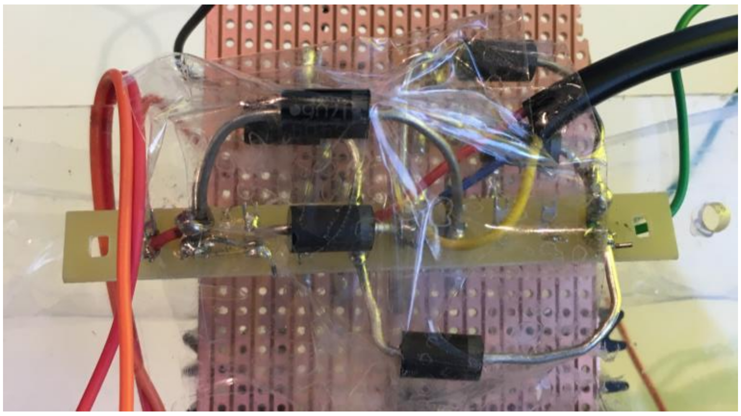


Figure 7: picture of real product:

- 7805 regulator

  A high-purity +5V supply is required to supply the sensor suite and microprocessors. This is achieved using a linear 7805 regulator.

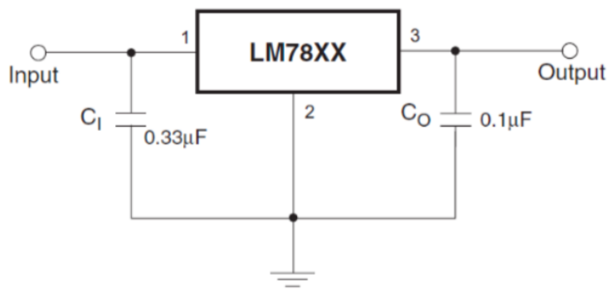


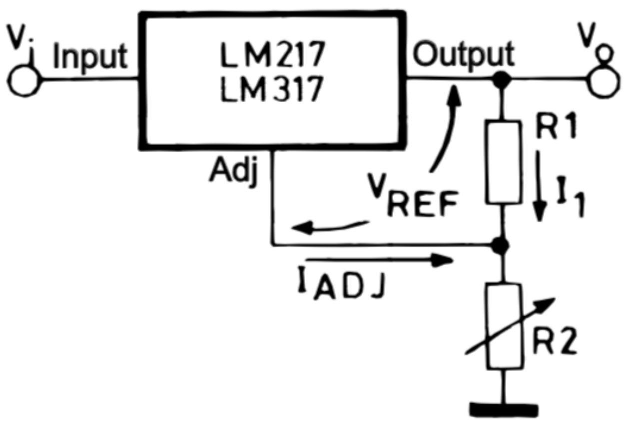
Figure 8: the legend diagram of 7805 regulator

Temperature considerations:

Voltage regulators operate by dissipating the required voltage drop as heat. From the datasheet, we know that for 7805, the temperature rise is 5°C/W with an infinite heat sink connected. The input voltage is +9V, so the maximum current drawn from the regulator to ensure the temperature does not exceed 60°C is 60/5/9=1.33A.

- Pre-regulator

  Because the maximum input voltage of a 7805 regulator in the following stage is +35V and this is less than the supply voltage for the vehicle(15-50V). Besides, we need a +9V DC voltage to supply the motors. Thus, a pre-regulator module should be built. We use LM317 to construct this part as legend diagram below.



From the datasheet:

$$V_O = V_{REF}\left(1 + \frac{R2}{R1}\right) + I_{ADJ}R_2$$

$$V_{REF} = 1.25V$$

$$I_{ADJ} = 100\mu A$$

$$V_O = 9V$$

We choose R1 = 220Ω, R2 = 0-5k Ω.

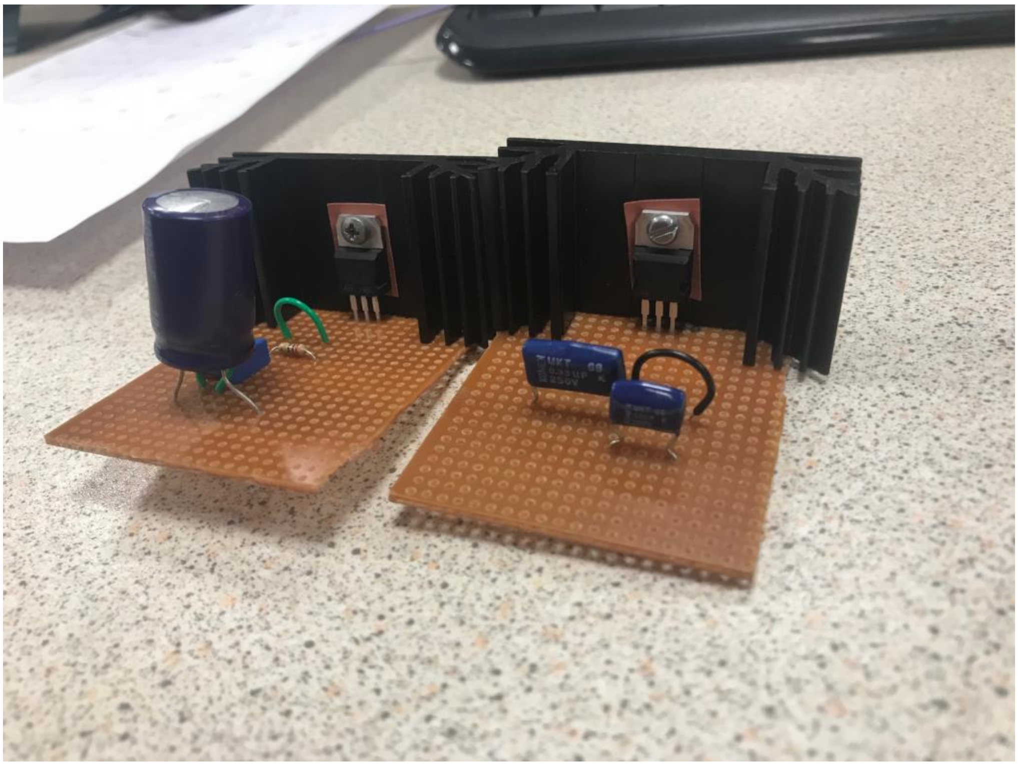


Figure 9: pre-regulator(left) and 7805 regulator(right)

II. Modified design:

When we built power conversion module and started to test it in the whole project, we found that Arduino needs 7-12V power supply and when Arduino is powered it can provides +5V voltage from one of its pin. This +5V voltage can be used to supply the sensor suits.

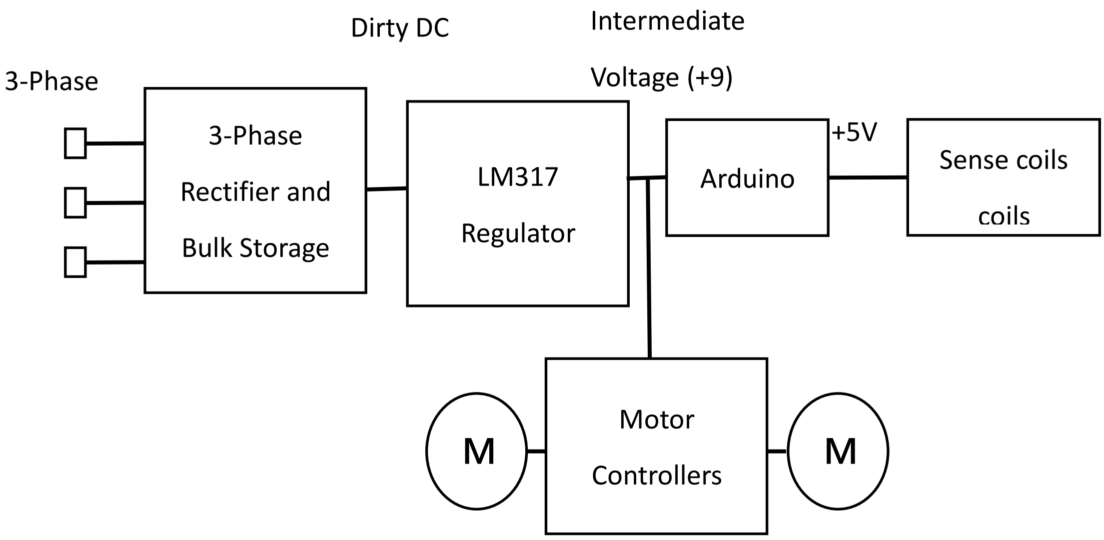


Figure 10: power unit-first design-modified design

Closed-loop DC Motor Armature Current Control

The goal of armature current control is to keeping the flux stable though the motor. With standard DC motors, the desired torque can be obtained by a certain current. Using feedback loop to control DC motor can make our vehicle pace steadily.

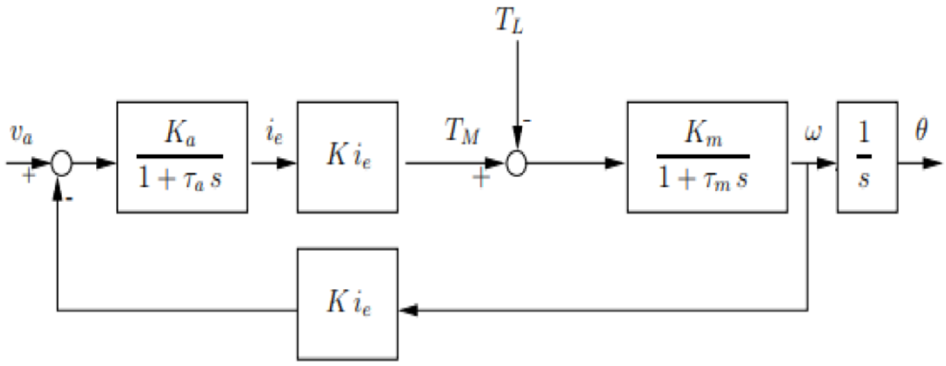


Figure 11: Block diagram of the linearization of the DC motor under armature-current control

In the project, we change the motors. The motors used in the EE2A experiment 3 rotating fast, unfortunately the torque it can provide is unstable, the current is oscillating all the time. As a result, we change those into a pair of stable small motors, whose velocity is relatively small but more powerful in torque.

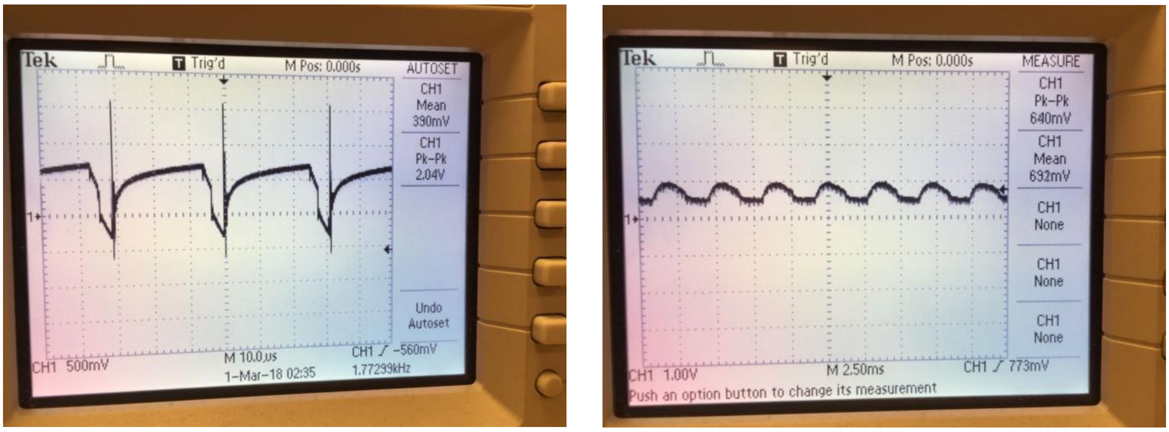


Figure 12

Those pictures were token during EE2A experiment3, which clearly indicate that with the closed-loop control, the armature current is much more smooth and stable than those without.

In our project, we cannot directly control the analogue value of the Va (input voltage), we use H-Bridge instead and control PWM to increase or decrease the input voltage.

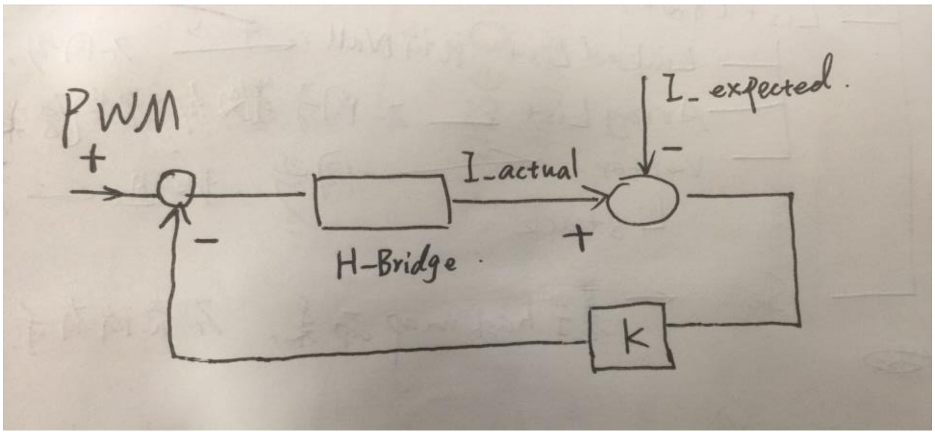


Figure 13

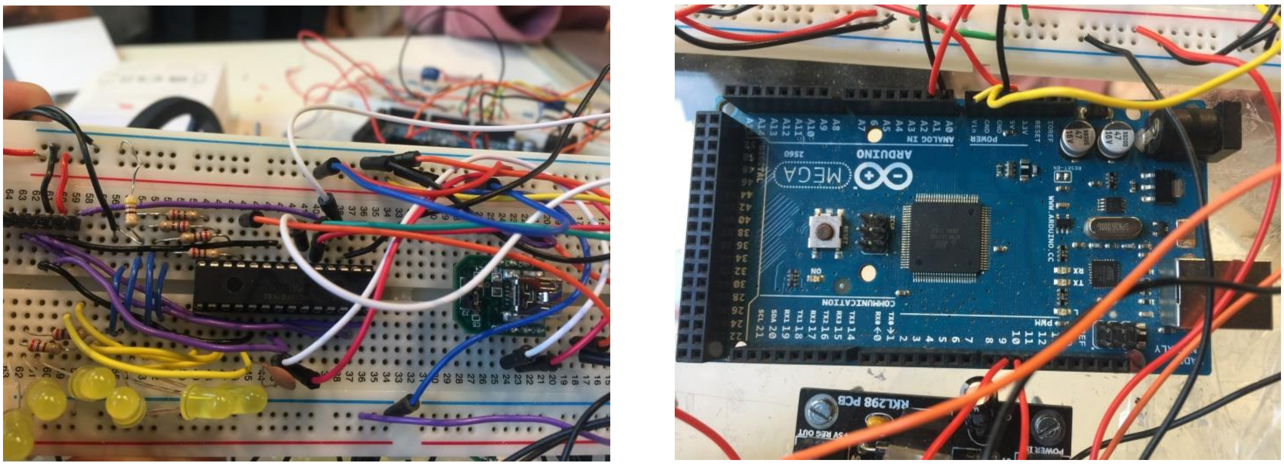
In our case, the PWM is initial to be 250 (max is 255), then whenever a difference happened, the system would rapidly adjust PWM, then maintain current to a constant number. During experiment 3 of EE2A, we successfully fulfil the goal of closed-loop control. Therefore, in this project, we can simply transfer the old code from PIC to Arduino board. Due to the differences between PIC and Arduino board, the code may seem different.

What need to be mentioned here is that we didn't use a constant number as coefficience of feedback loop, instead, we built a map which match the current error and the PWM change. In this way, PWM can change quickly in several oscillate clock periods, rather than change a constant number every loop regardlessly.

Higher-Level Closed-loop Controllers

Except for the DC motor current control, there are a lot of Closed-loop controllers in our vehicle.

One of the most important components is the navigation suite. In order to build



vehicle which can actually follow the wire, there is a lot of sacrifice need to be done. First, we use Arduino board in the project. Actually the reason we abandon PIC as a control core device is not about speed (the oscillate clock in PIC is impressively quicker than Arduino board, we will talk about it later) it is the peripheral. The peripheral of Arduino board is thoughtful; it not only includes PWM, ADC converter ports (rather than a generic port, which need to implement myself) but also provides us a much easier way to use them. In most case, one line of code can drive one part. It saves me a lot of time and let me have more time thinking about the algorithm to control the vehicle.

Figure 14 (It can easily judge the difficulty between using Arduino and using PIC to implement an ADC converter, let alone we need at least two ADC converters in our project)

However, there is one problem with this board, the oscillate clock is slower than PIC, as a result, the ADC convert either be slow but accurate or quick but inaccurate. We do not want to sacrifice accuracy to pursuit the speed. It is the follow the hidden line that define if the robot successful not its speed. In order to get an accurate magnetic field data, we did not reduce the detection time, instead we allocate 128 periods for Arduino board to detect data. And setup two array both storing 100 data within corresponding to two coils. Then calculate the average value of these numbers, and the variance as well. Because when the coil come close to signal, the variance would become greater while average value stays still.

(In ideal situation, we can detect the clear waveform of the signal through coils, then we can conclude it’s in the left or right with only one coil, while in our test, the data is quite blur, hardly read any useful information, so the same method used in experiment 5 EE2A is no longer suitable for this situation)

When the signal is closer to one coil than the other is, the variance in that coil would greater than the other's variance. Through comparing the difference between two coil, we can tell whether the signal wire is on the left or on the right side of the vehicle.

This is not enough to get an accurate judge, because the existence of noise, not every measure is accurate, but statistically most of the judge is convincible. So we come up with this plan. Whenever a left judgement is made, which means the vehicle is on the left side of the wire need to go right. Then the right-side motor's PWM would reduce 40, but when nor right neither left is detected, then the left and right motors would all increase 8. In this way, any effect of misjudge would be eliminate within 5 loop periods, while most of the right judgement would still affect.

During the race day, I assume a lot of audience would curious why our vehicle seems broken. The vehicle moves a little bit then pause a little bit then again move a little bit. It's designed on purpose. vehicle in one hand need to be sensible to signal while in the other hand need to ignore the noise. So, we decided it is only until the coil come really close to the wire then the turn left or right decision can be made. Otherwise, the misjudging situation would be too often. This decision brought up another problem. Even though we set PWM to be small, the vehicle still moves too quick. The time that one of the coil hit the signal is too short for Arduino board to process. As a compromise, we have to sacrifice the speed to gain accuracy of the detection. We designed the vehicle firstly stay still for five loops to "think" carefully to make the decision (to move straight, move left, or move right) then, the vehicle act based on the decision for one loop. Then repeat.

10.70	2.72	7.98	0	250	left	o
10.47	7.02	3.46	0	250	left	o
7.70	4.18	3.52	0	250	left	o
4.67	3.25	1.42	5	250		o
9.50	5.54	3.96	0	250	left	o
12.08	4.49	7.58	0	250	left	o
9.06	6.44	2.63	0	250	left	o
28.85	25.67	3.18	0	250	left	o
7.54	4.58	2.97	0	250	left	o
6.01	2.18	3.83	0	250	left	o
4.62	2.38	2.24	0	250	left	o
5.20	2.93	2.27	0	250	left	o
7.30	3.72	3.58	0	250	left	o
12.90	2.93	9.97	0	250	left	o
10.46	7.76	2.70	0	250	left	o
11.86	4.45	7.41	0	250	left	o
8.30	4.40	3.89	0	250	left	o
13.02	3.58	9.44	0	250	left	o
4.28	2.50	1.78	0	250	left	o
6.52	2.42	4.10	0	250	left	o
35.18	35.27	-0.10	5	250		o
7.48	3.05	4.42	0	250	left	o
12.26	7.87	4.38	0	250	left	o
8.37	4.54	3.83	0	250	left	o
8.57	4.14	4.44	0	250	left	o
6.47	3.28	3.20	0	250	left	o
12.17	11.76	0.41	5	250		o
4.30	4.86	-0.56	10	250		o
5.20	3.20	2.00	0	250	left	o
6.80	3.50	3.30	0	250	left	o
8.68	4.22	4.45	0	250	left	o
11.07	6.54	4.54	0	250	left	o
8.85	3.30	5.55	0	250	left	o
4.67	2.60	2.08	0	250	left	o
6.20	3.02	3.17	0	250	left	o
3.01	3.08	-0.06	5	250		o
5.82	2.40	3.42	0	250	left	o

2.86	3.86	-1.00	250	215		o
2.46	3.78	-1.32	250	220		o
3.10	3.94	-0.84	250	225		o
3.82	3.28	0.55	250	230		o
7.25	5.90	1.35	250	235		o
2.46	2.88	-0.42	250	240		o
4.26	4.37	-0.11	250	245		o
3.82	2.48	1.35	250	250		o
2.94	3.60	-0.66	250	250		o
7.58	6.58	1.00	250	250		o
5.57	7.90	-2.33	250	210	right	o
2.34	2.90	-0.55	250	215		o
3.78	5.27	-1.50	250	220		o
3.13	3.38	-0.25	250	225		o
3.66	3.49	0.16	250	230		o
3.09	2.50	0.59	250	235		o
5.74	6.47	-0.73	250	240		o
3.90	4.27	-0.37	250	245		o
4.16	5.17	-1.01	250	250		o
5.30	4.28	1.02	250	250		o
3.06	3.34	-0.28	250	250		o
2.50	2.94	-0.44	250	250		o
22.13	22.36	-0.23	250	250		o
5.36	6.48	-1.12	250	250		o
4.22	3.10	1.12	250	250		o
8.18	11.07	-2.90	250	210	right	o
7.90	7.48	0.42	250	215		o
3.40	3.50	-0.10	250	220		o
5.50	5.45	0.05	250	225		o
5.70	7.27	-1.58	250	185	right	o
4.38	5.06	-0.67	250	190		o
5.66	6.22	-0.56	250	195		o
3.50	2.36	1.13	250	200		o
9.98	10.28	-0.30	250	205		o
4.58	4.00	0.58	250	210		o
4.50	5.72	-1.22	250	215		o
5.74	5.61	0.12	250	220		o
3.62	3.74	-0.12	250	225		o
2.78	2.08	0.70	250	230		o
3.34	3.74	-0.40	250	235		o

6.62	12.20	-5.58	250	0	right	o
2.94	7.82	-4.88	250	0	right	o
3.28	5.69	-2.42	250	0	right	o
2.14	6.20	-4.05	250	0	right	o
2.85	4.98	-2.13	250	0	right	o
2.52	5.00	-2.48	250	0	right	o
2.80	5.50	-2.70	250	0	right	o
1.94	5.67	-3.73	250	0	right	o
2.88	7.45	-4.58	250	0	right	o
2.54	3.76	-1.22	250	5		o
3.60	10.24	-6.64	250	0	right	o
13.46	16.49	-3.03	250	0	right	o
3.86	6.82	-2.96	250	0	right	o
5.12	4.53	0.59	250	5		o
1.87	5.60	-3.73	250	0	right	o
3.61	4.67	-1.06	250	5		o
5.50	6.42	-0.92	250	10		o
4.86	8.46	-3.60	250	0	right	o
2.92	5.60	-2.67	250	0	right	o
3.22	3.60	-0.39	250	5		o
2.84	6.77	-3.94	250	0	right	o
2.70	6.46	-3.76	250	0	right	o
2.27	4.60	-2.32	250	0	right	o
2.20	10.47	-8.28	250	0	right	o
4.33	7.88	-3.55	250	0	right	o
2.78	5.18	-2.40	250	0	right	o
2.14	3.67	-1.54	250	0	right	o
3.26	5.40	-2.15	250	0	right	o
5.08	5.18	-0.11	250	5		o
3.06	5.20	-2.14	250	0	right	o
5.52	9.20	-3.69	250	0	right	o
3.42	4.32	-0.90	250	5		o
5.46	8.46	-3.00	250	0	right	o
4.97	10.54	-5.57	250	0	right	o
2.61	5.38	-2.77	250	0	right	o
2.97	3.10	-0.13	250	5		o
1.96	6.05	-4.09	250	0	right	o
6.62	10.20	-3.58	250	0	right	o
14.14	20.04	-5.90	250	0	right	o

Figure 15

Above are three pictures representing the data that would be used during signal process.

The first and second columns represent the Standard Deviations of the left and right coil data. The third columns represent the difference between those two Standard Deviations. The forth, and the fifth columns represent the PWM value of the left and right motor. The sixth column represents the judgement.

As shown above, both three situation contain some misjudging situation.

Electromagnetic Position Sensors

In this project, we are supposed to design a vehicle to follow a hidden wire carrying a current. The current consists of 1kHz and 2kHz sine waves, and its peak value is 10 mA.

From the lectures we know that a changing current can cause a changing magnetic field, and vice versa.

At first, we wanted to use the algorithm in EE2A lab5. And to meet the requirement of using that algorithm, we need to place two coils vertically, but orthogonal to each other. However, we found that the noise in the hidden wire was so big that we couldn't detect the signal. So we decided to use another algorithm to process the signal. Thus, we arrange 2 sense coils vertically at the front edge to detect the changing magnetic field.

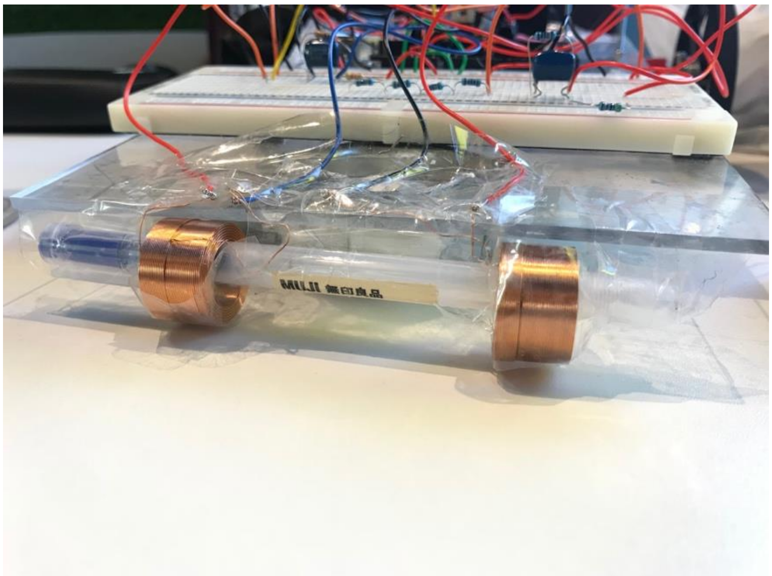


Figure 16: position of sensors

Figure 17: the track layout

Therefore, we can simplify the layout into two different cases: straight, and corner.

Now let's discuss both situations.

I. Straight

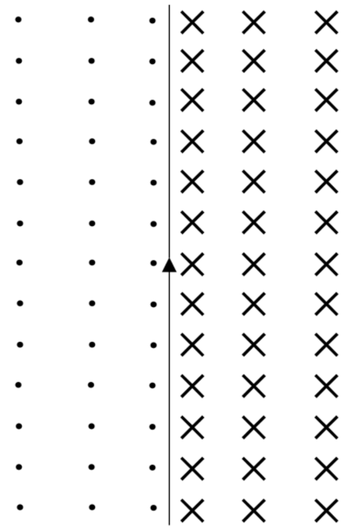


Figure 18: the magnetic field of straight line

If the vehicle is closer to the right side of the wire, the distance between left coil and the hidden wire will decrease. Therefore, the induced current in left coil will increase obviously, thus enlarge the gap between the currents in two coils. In this case, the processor(Arduino) will collect the data from coils and process it, then make the vehicle to turn left by controlling the speed of two wheels.

II. Corner

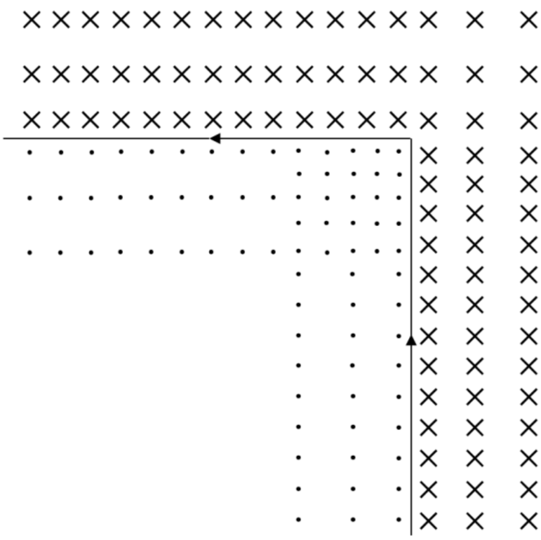


Figure 19: the magnetic field of corner

From Figure 18 we can obviously see that the magnetic field is stronger on the inside of the corner.

Theoretically, we should have placed both coils horizontally (to sense the magnetic field which is generated by the turning wire). That's because if we configure two coils vertically, the magnetic induction line will be parallel to them, which means the coil can't sense it. We tried to mount them horizontally but it failed. The signal we got was too weak, and the noise was too big to analyze. We tried many ways to solve this problem. Then we found it by accident that vertical coils work better. We guess it is because the vehicle has a high probability of not being parallel to the wire. Thus it could receive the induction line from both directions. That's why we choose to fix the coils vertically instead.

For instance, in the case of Figure 2, when the vehicle is close to the corner, the induction current in the left coil will increase whereas the induction current in the right coil will decrease. In other words, the difference between two currents will be larger. To eliminate this gap, the vehicle will turn left.

In actual operation, we need to provide a bias voltage to drive the sensors. The bias we used in this project is 0.9 V. Therefore, we reduce the 5 V supply with resistances of 4.7 kΩ and 1 kΩ.

Besides, we need to eliminate the high frequency noise in the coils.

$$f_c = \frac{1}{2\pi R_1 C_1}$$

The cut-off frequency we use is 2 kHz. Therefore, the capacitor should be 16.9 nF. Thus we use two 10 nF capacitors for decouple.

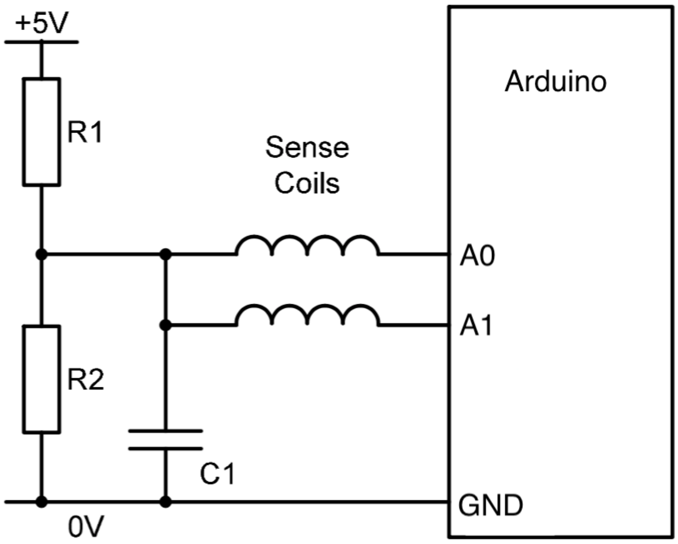


Figure 20

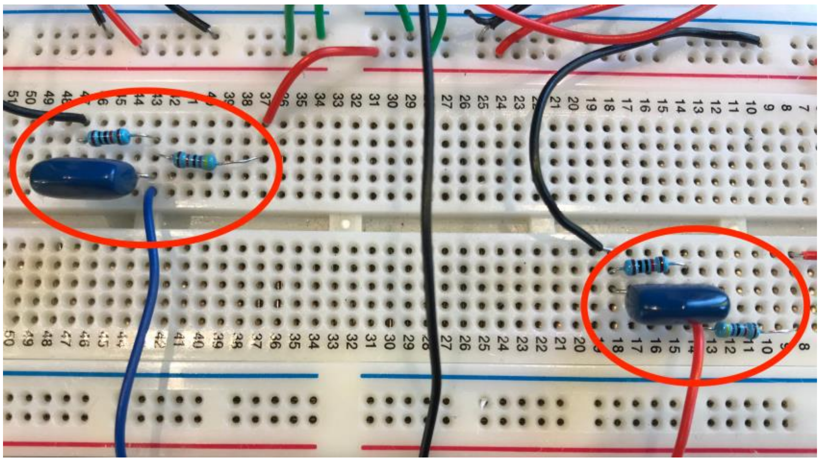


Figure 21

Numerical calculation:

$B = \frac{\mu_0 I}{2\pi r}$ (1)

$E = -N\frac{d\emptyset}{dt}$ (2)

$\emptyset = BA$ (3)

$I = 0.375 \times [\sin(2000\pi t) + \sin(4000\pi)]\ A$ (4)

$I_{coil} = \frac{E}{R}$ (5)

$I_{coil} = \frac{-NA\mu_0}{R2\pi r}\frac{dI}{dt} = \frac{-375NA\mu_0}{Rr} \times [\cos(2000\pi t) + 2\cos(4000\pi)]\ A$ (6)

From the formula (6) we know that, as the distance r increases, the induction current in the coil decreases.

CONCLUSION

After the final race we are proud to say that the second fastest vehicle is our vehicle with 2.57.88 seconds for completing the track. All the requirements have been met, our vehicle can follow a 10mA wire, can turn with no difficulties and reached the final line without stopping. Most important the vehicle is completely safe, no alarm can be caused by the robot at any point during its use or if not in use, this because it has been design with the right specification as required and all the dangerous components have been isolated from contact.

The robot car has been built for the purpose of taking part in the final race achieving the best score possible in the rank. Also, this has been a great opportunity for us to learn more about systems and control and their applications in real life and at university projects.

PERSONAL STATEMENT

Filippo Dal Ben

During this project my role has been of manging the group for meetings and dialogue with each other or with external members; I helped over the testing of each component finding key parameter for the vehicle and I have contributed with the design and solder of the Power supply construction: 3 Phase rectifier, Bulk storage capacitor and regulators; I have helped for the electrical construction and the safety of the vehicle; I have helped deciding what components where key for the robot, purchasing them from the technicians and incorporate them in the final design.

Main Role: Team Manager/Electronic Specialist

Alemseghed Ghebrezghi

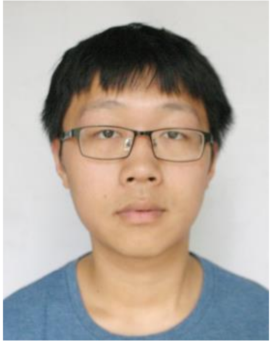
Since the wire following vehicle is working under DC power supply, this is achieved by converting the alternating current to DC current using three phase rectifiers. Hence, in this power conversion, I was involved in welding the diodes to 10-way tag-strip and make three phase rectifiers.

I was involved in wiring of the welded three phase rectifier (bridge rectifier). Since such direct application of the converted output is applied can harm the system, I was involved in connecting bulk capacitor after bridge rectifier so that ripples will be removed and to smooth the dirty DC supply and provide a steady power to the motor and micro controller unit and all other the electrical components.

Since the output can vary rapidly between zero and the maximum I was involved in installing voltage regulator with in circuit specially after decoupling capacitor so to avoid the transient noise associated with switching regulators.

I was involved in component selection for order.

I was involved in mechanical construction of the robot.



Donghao Qiu

During this project, my job was to make the vehicle to work during the race day. I wrote the code to make the vehicle could follow the hidden wire. I worked out how to make the vehicle detecting the signal wire. I tested and debugged the vehicle after the combination of software and hardware with the help of Yangyi Shi and Zhaoxi Yu. I helped to construct the body and assemble hardware components. I helped to make shopping list and decide the layout of the vehicle.

Yangyi Shi

In this project, my role is to build the structure of the robot and construct the hardware part as well as debugging the whole system with other members. I have helped for the designing the system architecture and selecting the fitted components used on the robot car. I built the main body of the vehicle and assemble the motor controller module with other members. Then I helped design and construct the power conversion part. After combining the software and hardware, I helped debug the parameter of the motor to improve the accuracy of changing the direction.

Zhaoxi Yu

During this project my job was to construct the body and hardware components. I helped make the shopping list and decide the layout of the vehicle. I helped cut the acrylic board, drill and fix the components. I helped test and calculate the parameters of motors. I constructed the motors, wheels, H-bridge, sensors, etc. with other group members together. I did the calculation for the regulators. I soldered the 3-phase rectifier and the circuits for regulators. I tested the hardware components and resolved the bugs. I helped the testing of vehicle performance.

Appendices

- Code:

```
int left_full=250;
int right_full=250;
int left_now;
int right_now;
int ifs=0;
double coco=0;
char asd[100];
char asd1[100];
double hightotal;
double hightotal1;
double highindex=0;
double highindex1=0;
int left=0;
int right=0;
int c=0;
double change;
double total;
double total1;
int start=1;
int inde=0;
void setup() {
  // put your setup code here, to run once:
  analogReference(INTERNAL1V1);
   Serial.begin(9600);
   //setP16( );     // Prescaler = 16
  //setP128( ); // Prescaler = 128 = default
   left_now=left_full;
   right_now=right_full;
    analogWrite(9, 0);//9 ports use for left pwm
  analogWrite(10, 0);//10 ports use for right pwm
  /*
  for(int j=0;j<5;j++)
  {
    total=0;
    total1=0;
    hightotal=0;
    hightotal1=0;
    for(int i=0;i<100;i++)
    {
       asd1[i] = analogRead(A1);
       asd[i] = analogRead(A0);
       total+=asd[i];
       total1+=asd1[i];
    }
    total=total/100;
    total1=total1/100;
    for(int i=0;i<100;i++)
    {
hightotal+=(total-asd[i])*(total-asd[i]);
hightotal1+=(total1-asd1[i])*(total1-asd1[i]);
    }
   //   Serial.print("all is ");
       //Serial.println(highindex);
       hightotal/=10;
       hightotal1/=10;
       hightotal=abs(hightotal);
```

```
        hightotal1=abs(hightotal1);
        coco+=hightotal-hightotal1;
  }

  coco/=5*/
  coco=0;
  Serial.println("coco");   // 100

  Serial.println(coco);   // 100

}
void setP16( ) {
  Serial.println("ADC Prescaler = 16");
// 100
  ADCSRA |=  (1 << ADPS2);    // 1
  ADCSRA &=  ~(1 << ADPS1);   // 0
  ADCSRA &=  ~(1 << ADPS0);   // 0
}
void loop() {
  // put your main code here, to run
repeatedly:

   if (Serial.available() > 0) {
              // read the incoming
byte:
              inde              =
Serial.parseInt();
              if(inde==0)
              {
                if(start==0)
                {
                  start=1;
                }
                else
                {
                  start=0;
                }
              }

            }
      total=0;
      total1=0;
      hightotal=0;
      hightotal1=0;
      highindex=0;
      highindex1=0;
      for(int i=0;i<100;i++)
      {
         asd1[i] = analogRead(A1);
         asd[i] = analogRead(A0);
         total+=asd[i];
         total1+=asd1[i];
      }
      total=total/100;
      total1=total1/100;
      for(int i=0;i<100;i++)
      {

    hightotal+=(total-asd[i])*(total-asd[i]);

    hightotal1+=(total1-asd1[i])*(total1-asd
    1[i]);
      }
    //  Serial.print("all is ");
         //Serial.println(highindex);
         hightotal/=10;
```

```
        hightotal1/=10;
        hightotal=abs(hightotal);
        hightotal1=abs(hightotal1);
        change=hightotal-hightotal1;
        //hightotal1+=18;

    Serial.print(hightotal);
    Serial.print("      ");
    Serial.print(hightotal1);
    Serial.print("      ");
    Serial.print(change);
//    if(hightotal>39&&hightotal1<26)
//    {
//        Serial.println("      right");
//    }
//                                          else
if(hightotal<37&&hightotal1>27)
//    {
//        Serial.println("      left");
//    }
//    else if(hightotal>47)
//    {
//        Serial.println("      right super");
//    }
//    else if(hightotal1>30)
//    {
//        Serial.println("      left super");
//
//    }
    if(change>coco+1.5)
    {
        if(right_now==right_full)
        {
            left_now-=40;
            if(left_now<0)
            {
                left_now=0;
            }
        }
        else
        {
            right_now+=40;
            if(right_now>right_full)
            {
                right_now=right_full;
            }
        }
        Serial.print("      ");
        Serial.print(left_now);
        Serial.print("      ");
        Serial.print(right_now);
        Serial.print("      left");
    }
    else if(change<coco-1.5)
    {
        if(left_now==left_full)
        {
            right_now-=40;
            if(right_now<0)
            {
                right_now=0;
            }
        }
        else
        {
            left_now+=40;
```

```
        if(left_now>left_full)
        {
          left_now=left_full;
        }
      }
      Serial.print("      ");
      Serial.print(left_now);
      Serial.print("      ");
      Serial.print(right_now);
      Serial.print("      right");
}
else
{

    if(right_now!=right_full)
    {
      right_now+=5;
    }
    else
    {
      left_now+=5;
    }
    if(right_now>right_full)
    {
      right_now=right_full;
    }
    if(left_now>left_full)
    {
      left_now=left_full;
    }
    Serial.print("      ");
    Serial.print(left_now);
    Serial.print("      ");
    Serial.print(right_now);
    Serial.print("      ");
  }

  if(start==0)
  {
    analogWrite(9, 0);//9 ports use for
left pwm
    analogWrite(10, 0);//10 ports use
for right pwm
     Serial.println("      c");

  }
  else
  {
    if(ifs%5<1)
    {
      if(left_now<(left_full*14/15))
      {
        analogWrite(9, 0);//9 ports
use for left pwm
        analogWrite(10,
right_now);//10 ports use for right pwm
      }
      else
if(right_now<(right_full*14/15))
      {
        analogWrite(9, left_now);//9
ports use for left pwm
        analogWrite(10, 0);//10 ports
use for right pwm
      }
      else
```

```
            {
                analogWrite(9,    left_now);//9 ports use for left pwm
                analogWrite(10, right_now);//10 ports use for right pwm
            }
        }
        else
        {
            analogWrite(9,  0);//9  ports  use for left pwm
            analogWrite(10,    0);//10    ports use for right pwm
        }
        ifs++;
        Serial.println("      o");
    }
//    if(hightotal>hightotal1)
//    {
//        change=hightotal-hightotal1;
//        analogWrite(9, left_full);//9 ports use for left pwm
//        analogWrite(10, right_full-change*80);
//    }
//    else
//    {
//        change=hightotal-hightotal1;
//        analogWrite(9, left_full-change*80);
//        analogWrite(10, right_full);
//    }
      // Serial.println(total);
      //    Serial.print(",");

}
```

- Picture of vehicle