Pan Siriviriyakul
20150951

Final report
Capstone design I

Group 7 배달의 민족

Advisor: Professor 김진환

Pan Siriviriyakul
20150951

Table of content

Pan Siriviriyakul
20150951

Final report

I am in charge of LabView part for our team.

Part I : basic study on the program and device.

Before starting the project, our team started by learning the basic function of Lab View and the devices. The main device for Lab View part is myRIO. The first assignment of Labview is to blink LED of myRIO and use accelerometer inside myRIO. Another task is creating wifi configuration of myRIO to match with PC and running as wireless device. The second assignment of Labview is basic coding of myRIO. The first task is to blink 4 LED of myRIO in different time interval. In this task, our team used quotient and remainder function to blink LED with different time and different duration. For example, in 1 second or 1000ms, the first LED would blink 1 time with 1 seconds, the second LED would blink 2 time with 500ms duration for each blink and so on. The quotient was used to find the amount of blink for each LED within one seconds. The second task is blinking 4 LED in order as 1 to 2 to 3 to 4 and go back to 1. For this task, our team used quotient and remainder function again in order to find the duration that each LED would blink in 1 second. For example, the first LED would blink during the period of 0-250ms, the second would blink during period of 250-500ms and so on. To sum up, for these two tasks, quotient and remainder function was used in order to divide the period of time as different range depending on remainder of time iteration divided by some range. Other approach such as waiting time could be used also but it has to be done with different while loop so in order to complete these tasks within one loop, this solution might be suitable.

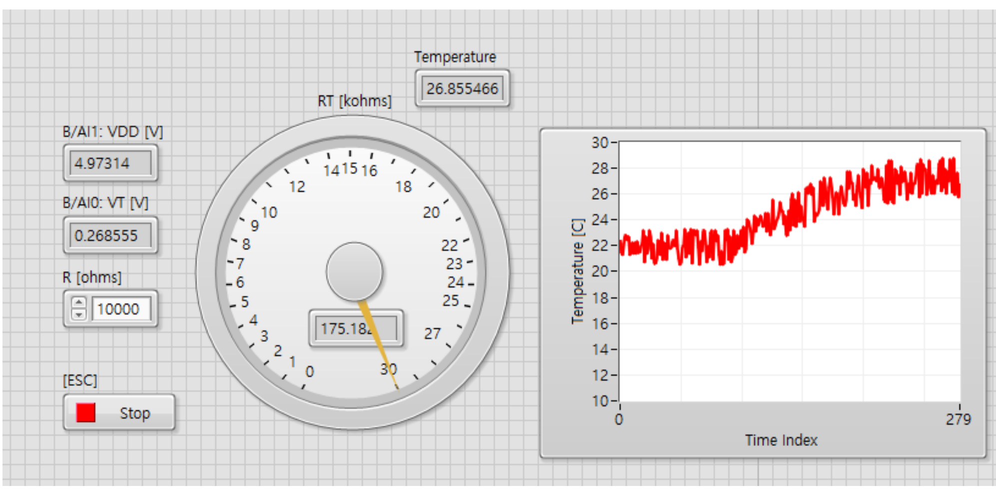
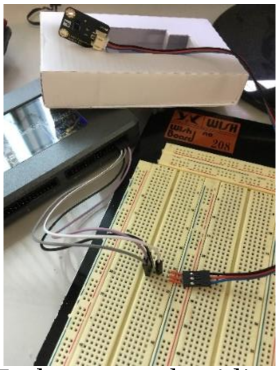


Figure 1 represents the temperature's measurement by using Dfrobot lm35, the temperature has linear relation with VT from the panel.

Figure2 represents the circuit of myRIO with Dfrobot lm35.



The third task for the second assignment of Labview are measuring acceleration by using accelerometer inside myRIO and measuring temperature by using Dfrobot lm35, the linear thermistor as addition device in the circuit. The subtask is to blink LED when acceleration or temperature crossed some threshold. In this task, we could use the properties of resistance which varied by the temperature as thermistor device and Labview example in order to measure the changed of voltage due to the changed resistance. The problem in this task is the circuit. By studying from datasheet and manual of Dfrobot lm35, there must be 4 lines to connect from myRIO and this device. 2 lines must be connected as the constant voltage and ground as digital voltage. The other two line would be ground and analog output voltage. According to this setting, in Labview, there would be two voltage represents as Vt and Vd. The Vt would be constant due to constant voltage applied on one line. Another voltage as Vd would be analog data from analog output line. This voltage would be changed due to the changed of resistance from temperature. When temperature raise higher, this voltage would raise also. The sensitivity of this device from data sheet is about 10mV per degree celcius. As a result, we could convert this voltage in the temperature scale. After acquiring this temperature scale, we could plot it as graph. The accelerometer inside myRIO could provide the information of acceleration directly so the acceleration could be plotted as graph.

Figure 3 represents the controller panel which would be linked with keyboards by ICT/TP connection from PC to myRIO.

The benefit of setting some specific value for acceleration and temperature to blink LED could be used for creating emergency signal to notify that the acceleration or temperature is too high in order to activate some function to cooling the system or suspend the motion.

Another work that we are working on is the third assignment, which is using TCP/IP connection with myRIO and coding Dynamixel using PC. There are two different modes for controlling. Wheel mode to control the speed of actuator and also position mode to use in the angle of rotation. The tasks in this assignment is to creating some typical command from the PC to switching mode between wheel's mode and position's mode. To change mode in this assignment, we decide to use local variable in order to specify the speed and position of the wheel corresponding to wheel mode (speed control) and joint mode (angle control). "Insert" button would be used to choose mode speed with 'up' and 'down' buttons are used for controlling speed from keyboard to the controller in figure 3 which is connecting to myRIO and dynamixel. "Delete" button would be used to choose position mode with 'left' and 'right' buttons are used to control the position (angle) of the wheel. "End" button would be used to assign zero speed to the motor in order to stop the motion as break in the system.

Part II : Implementation with motor and wheels.

After control one wheel at a time, we started studying about using TCP/IP to control more than one motor at a time. From the last time, we adapt our code to control motor by sending command to activate the code at myRIO by using TCP/IP connection. myRIO will carry the command from code to implement to the dynamixel. The final mission is to control all of the four motors which would be used in the mobility of the system. The system would move by using four Mecanum wheels with dynamixel as motor.

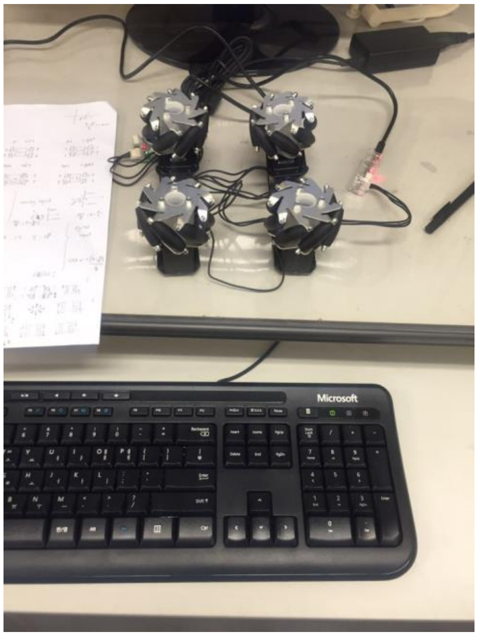


Figure 4 represents the connection of Mecanum wheels.

This figure represent all four wheel connected with dynamixel for each wheel. In this figure, the position is corresponding on each wheel position attaching to the car. From the connection as in the picture with example of dynamixel code, we could control the motor and wheel by using key board as the command in TCP/IP code.

In this project, the needed motion might be moving in all four direction as left, right, forward, back ward with the rotation as turning right and left. The movement of left and right coming from the benefits of this kind of wheel that could generate force motion as moving left or right. For example, moving in left direction, two left wheels would rotating toward each other while two right wheels would rotating outward. As a result, the moment created would generate the motion as moving to the left. We control all speed of each wheel to be the same for each direction of motion.

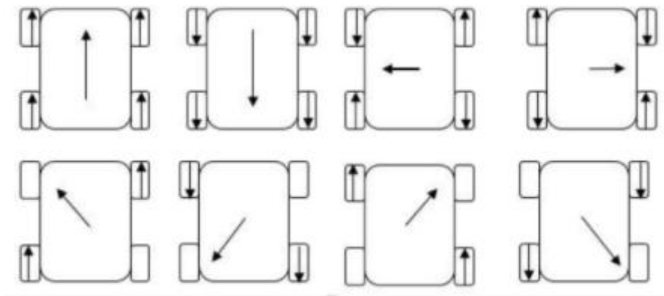


Figure 2 represents the motion of wheels corresponding to total direction.

Following these eights directional control, our system would be able to move in all eight direction around the map for final mission. This motion would be very important for the control for some fixed distance such as returning the ball to basket. Our team algorithm is rotating the car to be in parallel with the line of basket before moving towards it. It is the reason why all eight direction of motion is needed in order to keep the car moving without rotating.

After managing with motor control of wheels, our group also succeeded in creating the platform for the car. The position of camera would be in the front on top of myRIO but the angle of it might need some trial before considering.

Figure 3-4 represent the prototype of the whole system as platform.

After that we focused on the connection with TCP/IP from myRIO with computer with ROS team. ROS team provided the connection as client as in the example of LABVIEW. The client would send the information to activate such as the information from X-box controller. The X-box controller could provide information such as axis and the magnitude of the controller as x, y and z axis.

Part III : Integration with ROS

We succeeded in integration with ROS team to use the X-box controller to control wheel. From LABVIEW session, we learned that one of the most important part to integrate the part of LABVIEW to ROS team is the output from ROS part since LABVIEW code would control how much data would be read for the implementation. We adapt from the example code to receive data and read the data. From the trial with real system, we succeeded in the movement of the car. The car could be moved in all eight direction (as in the left joy stick of X-box controller) and also the rotation from using the right stick of X-box controller.

In the code, our team decided to use the X-box controller by the left joy stick for controlling the movement and using right stick in X-box controller to use as rotation. From the first step, we used the array from X-box controller as receiving the index in the array (as the output from X-box controller). We used three array as three output from the controller. The first one is the angle of right stick from X-box controller. The second one is the magnitude of the axis. The third one is the magnitude axis from the right axis (x-axis of the right axis).

For rotating the car, we decided to use the magnitude of right axis to consider the rotation right or left and control the motor following with a specific velocity of all for wheel. When we stop using stick, this magnitude would go to zero so the car will not rotate.

For the movement of the car (not rotation), we used only left stick to control. First of all, we acquired the angle of rotation from joystick of the left side in X-box controller. We divided this angle in to 4 phase as from 0 to pi/2, pi/2 to pi, pi to 3pi/2 and from 3pi/2 to 2pi. When we consider the implement with joy stick, for the first period, the car would move in around north-east direction including the forward and right hand side movement. Our main mission for LABVIEW team is to control the car as its motion following the joystick magnitude and direction. For example, if the left-side joystick, was pushed in north-east direction, the car should follow it in that direction. We completed this mission by using the magnitude of axis for controlling. There are four cases divided based on the 4 phases of angle. In each phase, 2 wheels would be controlled by a constant value as the constant velocity while other 2 wheels would be controlled proportional to the magnitude of the axis from controller. These two wheels couples would be pair up in the diagonal direction. For example, for controlling in the phase from 0 to pi/2, the left-front wheel would be controlled with the right-rear wheel as constant velocity in order to assure that the total motion of the car would be in that range of direction according to the summation of torque in the manual of dynamixel. The controller in right-front wheel and left-rear wheel would be proportional to the magnitude of the axis as control of how fast it would go in that direction and in which direction as the summation of torque depending on this pair. First, we subtract the angle value with the value of pi/4 and divided it by pi/4 in order to transfer the direction and magnitude as proper speed for the wheel to move. This value would be multiplied by minus value in order to apply with right-front wheel and left-rear wheel while the constant velocity would be implement for left-front wheel and right-rear wheel. For example, when the input angle is pi/2, the input for right-front wheel and left-rear wheel would be one multiplied with constant velocity applying with the left-front wheel and the right-rear wheel. As a result, the right wheels and left wheels would move with the same velocity as the car move forward following the controller. Other phases would be implemented in the similar way but changed in the detailed of value to subtracted due to the phase different.

Simplified version of x-box controller to the system, the left stick would be used for speed controller while right stick would be used for rotation controller.

For the sweepers for the front and the rear gate, the front one would be used for collecting the ball, the rear one would be used for deliver the ball to finish the mission. So we added two more motor in the code to control these two sweepers. The initial angles for these two sweepers would be different from the other four motors. The front sweeper would have initial position of 150 degree and rotating to 0 degree while sweeping.

From the integration with ROS team, signal from ROS would be sent in the same way as X-box controller in order to control all motor autonomously with other part such as web cam to complete the mission. The last part would be debugging some hardware problem during the trials for our mission.

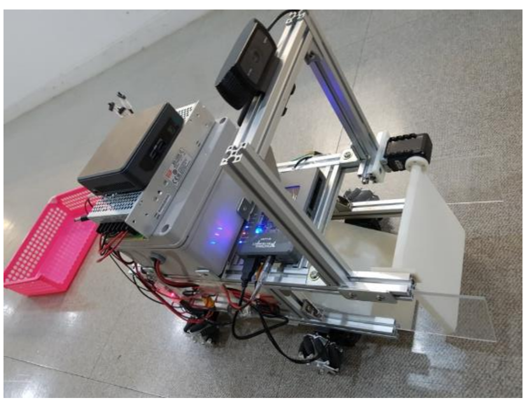
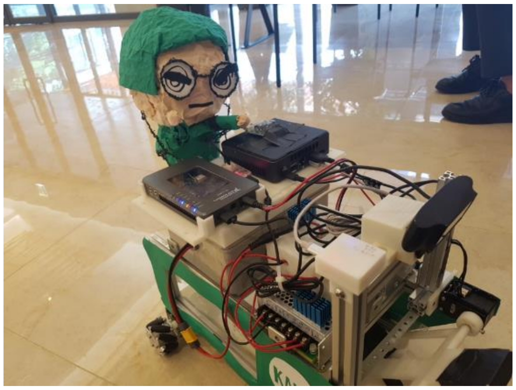


Figure 5 represents one prototype for our car.

After some trial, the wheel are struck and fell from the bearing. This is the problem from plastic type of bearing that we used from the beginning. When more load are added and after some amount of trials, the bearing become loosen. As a result, our team come up with the idea to implement 3D printer for creating the joint with the metal bearing. After applying this 3D printer with the new type of bearing, the wheels moved more fluently and consistently.

Our group also consider the temperature rise with energy management from the measured temperature through thermo-camera. We found that the highest temperature rise from the motor and myRIO. We attach fins to the part especially the sweeper motor which is the one that connect directly with motor drive and pass power to other motor in the system. After calculating about the temperature management by using fans. Our group consider not attaching the fans due to higher energy consumption with additional converter since the total amount of time for trial would be ten minutes for maximum. We have already try to run all of the system and keep the wheel rolling for ten minutes before measuring the temperature in each part of the system and consider the temperature rise. The temperature rises motor is lower than 36 degree celcius which is much lower than the maximum running temperature.

The last mission is to control the car to move back to the basket by detecting the green balls. However, it is troublesome because the distance between two balls would be different and also the size would be varied due to the different distance from the position of the car. Our group manage to solve this problem by applying the compass. The concept is setting this compass to calibrate the angle of the initial position in front of the basket. After collecting all the balls, our car would rotate back to face the basket in parallel by applying the rotation to the opposite angle with the initial point. After that, the car would move in right or left direction to face the basket before going straight and return the ball into the basket. The compass is calibrating from ROS part with taking the initial position before start.



For the final product, we also created some mascot to decorate our system since outlook is one of the most important thing that could give impression and describe our team name 배달의 민족.

Figure 6 represents the decoration on our car as final product.

It is a great experience and memory to work with the team in the course.

Pan Siriviriyakul
20150951

Extra : Heat transfer calculation

Considering the heat transfer from attaching fans and without attaching one.

Assumption : -Ambient air is quiescent

-Surface radiation effects are negligible.

- Ideal gas
- Constant properties
- Steady state conditions
- Temperature is uniform across the fin thickness

$T_f \approx 300K$, $T_s = 38.3℃ = 311.3K$ $(the\ highest\ tempt)$, $T_\infty = 25℃ = 298K$, wind velocity = 5.218 m/s

air properties at T_f : $Cp = 1007 J/kg \cdot K$, $\nu = 15.89 \times 10^{-6} m^2/s$, $\alpha = 22.5 \times 10^{-6} m^2/s$, k = $26.3 \times 10^{-3} W/m \cdot K$, Pr $= 0.707$, $\beta = \frac{1}{300}\ k^{-1}$. Aluminum fins properties : k = 180 $W/m \cdot K$

Without fan case: Consider as free convection over upper surface of a hot plate.

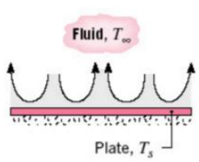


Figure 7 represent heat transfer from a hot plate.

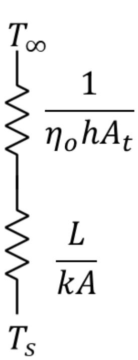
Upper surface of a hot plate:

$$\overline{Nu_L} = 0.54\, {Ra_L}^{1/4}\ (10^4 \le Ra_L \le 10^7,\ Pr \ge 0.7)$$

From $Ra_L = \frac{g\beta(T_s - T_\infty)L^3}{\nu\alpha}$, $\overline{Nu_L} = \frac{\bar{h}L}{k}$

$$q = \bar{h}A(T_s - T_\infty)$$
$$q \approx 0.05149\ W$$

With fan case: Consider external flow through fins convection.



External flow through fins:

$Re_L = \frac{VL}{\gamma}$ $< 5 \times 10^5$ Laminar flow with Pr ≥ 0.6

$\overline{Nu_L} = 0.664\, {Re_L}^{1/2} Pr^{1/3}$, $\overline{Nu_L} = \frac{\bar{h}L}{k}$

$R_t = \frac{1}{\eta_o h A_t} + \frac{L}{kA}$ $\qquad q = \frac{(T_s - T_\infty)}{R_t}$

$q \approx 3.992$