Amphibious Spherical Explorer

Requirements and Verification

By

Kaiwen Chen

Zhong Tan

Junhao Su

ECE 445 Senior Design, Spring 2016

Project No. 30

TA: Luke Wendt

April 8, 2016

Table 1. Table of Requirements and Verification

Module or overall performance	**Requirements and Verification**	**Points**
Control panel on PC	• R: Use command line to send commands to the robot and show all the commands are executed correspondingly.(5 pts). Use game controller to send commands and show all the commands are executed correspondingly (5 pts). • V: First send four imperative commands (roll left, roll right, accelerate, and decelerate), and two assignment commands (change motor PWM, and change servo position). Then, enable gamepad control in the control panel, check whether the robot can move accordingly.	10
WiFi	• R: Packet loss rate $< 10\%$ • V: To verify this, separate the robot and the computer by 20 ± 2 m. Then, ping the WiFi module in terminal with 20 packets. Check at the terminal for the number of packets sent and received. The definition of packet loss is $$\text{Packet loss rate} = 1 - \frac{\#\text{ received}}{\#\text{ transmitted}}$$	5
Microcontroller	• R: Interrupt cycles should be executed at frequency > 20 Hz, that is, each task schedule cycle (a list of tasks that run in peroidic cycles) has to cost < 0.05 s. • V: Modify the test program, let the microcontroller pull-up an idle GPIO at the beginning of each task schedule cycle, and pull-down the same GPIO at the end of that task schedule cycle. Using oscilloscope to measure the pulse width generated on that GPIO port, so that we can measure the time cost for each task schedule cycle.	5

Continued on next page

Table 1 – continued from previous page

Module or overall performance	Requirements and Verification	Points
Servo	• R: Give the servo proper PWM signal so that it can rotate within the range about $-45°$ to $+45°$, with tolerance $\pm 5°$ (use correct pulse length, 2 pts). The servo should also converge to the desired position smoothly without random intensive oscillation (use proper pulse frequency, 3 pts). • V: Change the program to open-loop mode. The microcontroller will output pulse with width of 1000 μs to 2000 μs, according to the input on the control panel. Use an oscilloscope to view the PWM and record pulse length as well as frequency. change the pulse width using command line, check if the servo can rotate over the full range required, and observe if there is intensive oscillation at steady state.	5
Motor driver and motor	• R: The motor should be able to change speed (5 pts) and direction(5 pts). • V: Use test program, which maps an analog input from 0 to 3.3 V, to a PWM output with duty cycle from 0/255 to 255/255, to generate PWM wave given to the motor driver. Check if the speed increases monotonically as the input voltage increases, and whether the direction of rotation reversed when the voltage level on the phase pin of motor driver is flipped.	10

Continued on next page

Table 1 – continued from previous page

Module or overall performance	Requirements and Verification	Points
Magnetic encoder	• R: Communicate with the microcontroller (4 pts). The angle for speed calculation should be compensated for the modulus effect, which means $360^\circ + \alpha$ should be recorded when the angle goes to α in the next revolution, to avoid unexpected sign change in speed measurement (4 pts). The error of the speed measurement should be within 50% (2 pts). • V: To verify speed measurement, we first run the robot into steady speed. Then we measure time it takes to run 10 meters and at the same time, buffer the data inside the microcontroller and send the buffered data to computer via WiFi or serial port after the test. Take the averaged buffered data and compare with the averaged speed measured using stopwatch. If the result measured by magnetic encoder and result measured using stopwatch closely matches and the error is within 50 % , then the requirement is verified.	10
IMU	• R: Communicate with the microcontroller (5 pts). The angle measured by IMU should be within $1 \pm 20\%$ of the actual angle, within -45° to 45°. • V: Draw a collection of lines from -45° to 45° with increment of 15°. Tape this paper on the wall, and let the edge of the MPU board coincide one line each time. Read the data measured and analyze the error.	10
Power circuit	• R: Circuit take input voltage from 7.4V battery, output 3.3V $\pm$10% (3 pts). Power circuit should be decoupled for power devices (motor and servo) (2 pts). • V: Use oscilloscope to measure if the average output voltage is correct. Run the motor and add different drag force to it, and check if there is visible pulses of amplitude more than 50% of the output voltage.	5

Continued on next page

Table 1 – continued from previous page

Module or overall performance	Requirements and Verification	Points
Moving Speed	• R: The average constant maximum speed when moving forward straightly should be $\geq$ 1 m/s. • V: First run the robot with full PWM duty cycle into steady speed. Then we measure time it takes to run 10 meters using stopwatch. Take the averaged speed across the 10 meters and check if the speed above 1 m/s.	10
Acceleration/brake	• R: The rising time of the speed step response (1 m/s) of the robot should be $\leq$ 15 s (4 pts). The time taken to stop the robot from the maximum speed should also be $\leq$ 15 s (4 pts). The overshoot of both process should be $\leq$ 50% (2 pts). • V: Send command to let the robot run to the maximum speed and buffer the speed data meanwhile. Log the speed data to control panel via WiFI or serial port after the test and measure the rising time and overshoot of the both step response using MATLAB. If there is too much noise in the speed measurement, apply piecewise cubic fit to the data and then measure the response specification.	10
Wobbliness	• R: Apply a initial angle of 30° to roll angle, the robot should be able to re-balance itself to ±15° from equilibrium roll angle in less than 20 oscillations. • V: Use the protractor to place the robot with a roll angle of 30°. Then, release the robot and let IMU measure roll angle. The data will be buffered during the test and sent back computer after the test via WiFi or serial port. Log the data to MATLAB and plot the data versus time to count the total oscillations before it goes into steady state (region within ±10°).	10

Continued on next page

Table 1 – continued from previous page

Module or overall performance	Requirements and Verification	Points
Turning radius	• R: The minimum turning radius should be ≤ 2 m. • V: Set the robot to rest and let it start to turn in one direction. After the turning motion reaches its steady state, put a reference object with known dimension near the turning circle and use a camera to take three pictures from the same position right above the estimated center of the circle (from a height directly above such as the ECEB second floor looking down to the lobby). Merge the three pictures into one and use the three positions of the robot to reconstruct the turning circle. Use the relative size of the reconstructed circle and the length of the ruler in the picture, we can calculate the real size of the turning circle.	10