

TDPS Initial Report by Team 07

Members:

2613968Z Zou Hanlin 2614344G Gu Xingci 2613973G Guo Linhong
2613949C Chen Xi 2613977S Sun Linhan 2613977S Liu Cehan 2614353Y Yuan Ye
2613951L Li Chenghao 2614354Y Yang Chun 2614329S Sheng Dian

Date: March 2023

Content:

Abstract	Page 1
1. Introduction	Page 1
2. Task Analysis	Page 1
2.1 Patio 1	Page 1
2.2 Patio 2	Page 1
3. Top-level system design	Page 1
4. Sensor design	Page 1
4.1 Open Mv camera programming design	Page 2
4.1.1 Edge detection	Page 2
4.1.2 Filtering, Adaptive binarized image	Page 2
4.1.3 Arrow identification	Page 2
4.2 Ultrasonic detectors design	Page 2
4.2.1 Distance measurement:	Page 2
4.2.2 Navigation	Page 2
5. Control design	Page 2
5.1 Arduino Chassis design	Page 2
5.1.1 Motor control	Page 2
5.1.2 PID adjustment	Page 2
5.1.3 UART communication	Page 2
5.2 Robotic arm design	Page 2
5.2.1 Servo design	Page 2
5.2.2 Robotic arm design	Page 2
6. Integration design	Page 3
6.1 Communication approaches	Page 3
6.1.1 Communication module	Page 3
6.1.2 Data sending	Page 3
6.2 PCB main controller design	Page 3
7. Conclusion	Page 3
References	Page 3

List of Figures:

Figure 1: Flowchart of system design	Page 1
Figure 2: Parts of the robotic arm	Page 3

Team Design Project and Skills Initial Report

Abstract: The team design project requires 10 team members to design, build and present a rover that accomplishes activities on East Lake, UESTC, with two patios, each containing three tasks. Our original designed rover is equipped with an OpenMV camera, three ultrasonic detectors as its sensors, Nucleo-H7 as the central controller, a single robotic arm, a servo and others. These have form a unified system in the top-level system design flowchart, which is initially subdivided into sensor design, control design, and integration design. Our rover could accomplish the preliminary completion of the six tasks.

1. Introduction:

Intelligent small rover is a prominent topic of research. It has a chassis, wheels, motor, micro controller, and battery pack. It avoids impediments and drives independently using sensors, controls, and algorithms [1].

In this course, we build an autonomous rover that can complete three tasks on each of the two patios at East Lake of UESTC.

The remainder of this paper proceeds as follows. Section 2 discusses functions of rover. Section 3 creates a rover system flowchart. Sections 4, 5, and 6 present rover sensor, control, and integration designs. In Section 7 the report is concluded and future improvement is given.

2. Task Analysis:

2.1 Patio1: To perform tasks 1.1 and 1.3, where the rover must follow colorful tiles and stop at a gate, OpenMV needs to have visual identification and pavement processing capabilities to determine the optimal route. Task 1.2 involves locating and crossing a bridge, requiring three ultrasonic detectors to ensure proper alignment with the bridge and sidewall.

2.2 Patio2: Task 2.1 involving identifying arrow shapes and directions, the OpenMV camera in patio1 must be modified. To complete Task 2.2, a robotic arm attached to the servo is needed to launch a tennis ball into a basket. Task 2.3's main requirement is communication, which involves using a wireless transceiver to send messages to a laptop computer.

3. Top-level system design:

The flowchart diagram of overall rover system is shown as Figure 1:

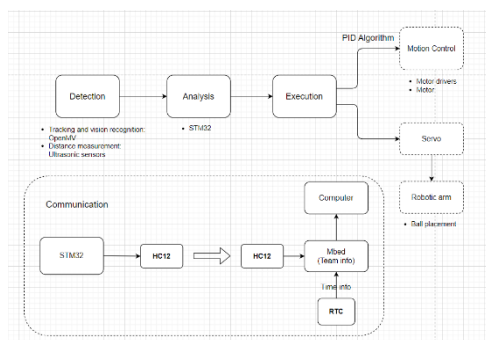


Figure 1: Flowchart of system design

For detection phase, an OpenMv camera and ultrasonic detectors are utilised as sensors to identify the patio tiles and estimate the robot's distance from the handrail.

STM32 MCU receives information gathered by sensors for analysis. At execution part, a servo is instructed to drop the ball into the hoop via a mechanical arm, and stable functioning of the mecanum wheels is controlled by PID algorithm [2].

In the communication section, an HC12 module is activated to transmit team information together with the current time given by an RTC module.

4. Sensor design:

The sensor module detects physical parameters, sends data to the car's control system, and helps the rover adjust behavior.

4.1. OpenMv camera programming design:

4.1.1 Edge detection: Rover must navigate a scarce gravel pavement and a flat stone pavement. Canny operator is selected to acquire the thick and intricate patrol pavement edge and the sparse standard pavement edge.

4.1.2 Filtering, Adaptive binarized image: The Mean operator blurs the image in order to reduce the following computing burden. Then, use mean pooling on grayscale images to detect pavement only in pixels with dense edges.

4.1.3 Arrow identification: Traditional digital image processing methods are able to extract the geometrically weighted centre of the arrow, but are unable to identify it from a distance. We choose machine learning in the end, photographing acrylic standing plates beside black standing plates, manually labelling them, then cropping the model with Edge Impulse to determine arrow direction.

4.2. Ultrasonic detectors design:

4.2.1 Distance measurement: Three ultrasonic detectors are going to be installed, and two of them would be installed parallel to the rover's right side at railing column height. The front ultrasonic detector measures the distance to the bridge, while the side sensors measure two lengths to make sure it's parallel to the bridge and side wall to complete task 1.2.

4.2.2 Navigation: During the basket searching process, it is planned to use multiple ultrasonic detectors to ensure that the rover can navigate stably along the railing's edge, as the cobblestone texture on the ground is difficult to extract reliably to find the trash can.

5. Control design:

The control module employs the PID algorithm [2] to control the motors, and controls the servo-controlled robotic arm with a PWM output.

5.1 Arduino Chassis design:

5.1.1 Motor control : Basic motor control is driven by a PWM output to alter the duty cycle, consequently adjusting the output size to control the motor speed, and changing the voltage of the two digital pins, high and low, to control the direction of the motor.

5.1.2 PID adjustment: As the PID concept is simple, robust, and applicable, it significantly enhances the accuracy of wheel movement. In our rover, we employ the PID algorithm to verify that the actual speed of the wheel corresponds to the necessary speed.

5.1.3 UART communication: UART communication is characterized by receiving host controller-sent messages. This communication from the host controller handles and allocates the speeds and directions of four Mecanum wheels by receiving data from an OpenMV camera.

5.2 Robotic arm design:

5.2.1 Servo design: One MG995 servo, a metal servo that rotates more forcefully and steadily than average servos, is used to drive the robotic arm. A PWM signal activates the servo, which rotates the arm for 180 degrees from its initial position and drops the ball into the basket.

5.2.2 Robotic arm design:

The 3D model of robotic arm is shown as Figure 2:

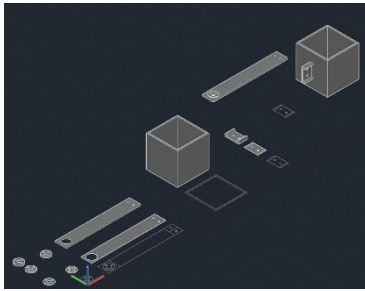


Figure 2: Parts of the robotic arm

The robotic arm is designed using 3D modelling and printing, allowing the adjustment of the length, thickness, and interface type of the driving rod. The robotic arm transmission rod is on the right, while the robotic arm disc is on the left. The right side is attached to the pool table's frame, the basket and connection are connected by screws and nuts, and 3mm redundant space is reserved at the front and back to reduce jamming and damping.

6. Integration design:

Communication approaches and PCB main controller design are the foundation of the Integration design, forming the backbone of the system design, making it robust and efficient.

6.1 Communication approaches:

6.1.1 Communication module : HC12 radio module is chosen to transmit information from the rover, which works at a frequency of 433MHz and is controlled by UART. A pair of HC12 are used to transmit data between the rover and the computer.

6.1.2 Data sending : As the rover reaches its destination, the main controller STM32 (H7 series) sends an enable signal to the outside through its HC12. The HC12 attached to the Nucleo-L432KC transfers data to the computer through UART after receiving the signal. By using this design, the load of the main controller is lessened and the communication stability is improved. RTC (Real Time Clock) is a module that could provide real time for digital devices. After Nucleo-L432KC receives the enable signal, it would send current time obtained by RTC and team information to the computer by UART.

6.2 PCB main controller design:

The PCB serves as a stable connector among components of the system, including battery, sensors, STM32 micro controller, servo, motor drivers and communication module. The specific design of the PCB is going to be done after testing the system.

7. Conclusion: The rover uses sensor, control, and integration modules to complete six tasks on two patios in this course. Early planning and coordination helped these modules provide a reliable hardware platform for the software to accomplish basic functions. Future enhancements should add ultrasonic sensors to the OpenMV camera's sensor part to stabilise the rover's navigation while finding trash cans in Patio2. For the control design, the robotic arm's tennis ball would fall out after 90 degrees, resulting in an insufficient drop point and requiring improvement.

References:

- [1] S. Cherian, W. O. Troxell, and M. M. Ali, "Design of behavior-based micro-rover robot," *Proceedings of the Intelligent Vehicles '92 Symposium*.
- [2] Y. Xie and J. Meng, "PID control for the vehicle suspension optimized by the PSO algorithm," *3rd International Conference on Electromechanical Control Technology and Transportation*, 2018.