The Assembly and Manipulation of the Intelligent Transport Vehicle Equipped Robotic Arm

Yuheng Jiao [1], Dingyan Zhang [2, *]

[1] Shenzhen College of International Education, Shenzhen, 518043, China

[2] Gonville, Eastbourne College, Eastbourne, BN214JY, The United Kingdom

* Corresponding Author Email: deanzhzdy@mailbox.org

Abstract. With the advancement of artificial intelligence technology, robots are increasingly replacing human labor and playing an increasingly important role in production. The research focused on goods transportation, which plays a great role in society. However, the traditional method for goods transportation was only to use the labor force, which was very inefficient and expensive if applied on a large scale and could contain life risks under dangerous working conditions. Therefore, the research attempts to explore a new method by using robotics and automation in the transportation system, where robots are the ones to carry the heavy goods, whereas human is the controller remotely. This can substantially increase productivity, and accuracy and lower the costs for transportation, leading to enormous economic benefits, as well as safety. At the end of the research, we successfully created a small remote-controlled transport cart assembled with an artificial arm, with the cart that can be moved omnidirectionally. The research carries out potential benefits for future skills development, applies theory to reality, and builds up a correlation between them.

Keywords: Transport vehicle; Robotic arm; Remote control; Algorithms and data conversion.

1. Introduction

The remote-controlled transport cart and the robotic arm can be applied to many real-life problems in many fields, from manufacturing to agriculture, if associated with goods transportation. For example, they are widely used in warehouses and logistics for moving goods between different areas, in manufacturing, such as transporting material in assembly lines, especially for heavy components like engines, in agriculture to transport crops from field to processing areas, and in the mining industry to replace the worker to carry heavy loads in confined and poorly ventilated underground mining space. Therefore, in place of workers, carrying out these physical tasks by robots can increase productivity by not feeling tired, precision by seldom making mistakes, and safety by subsisting workers to work in extreme conditions. So far, a lot of research has been done on the remote-controlled transport carts, and they are widespread in many domains. Research demonstrates the application of remote-controlled transport carts to carry the purchased goods to the cashier in the supermarket so that people don’t have to wait in queues with heavy items, especially in peak shopping hours when queues are quite long but instead, they can sit in a comport place to control the robot for transporting their groceries to the cashier [1]. There is also research about the remote control of the long truck and remote-controlled airport luggage carts [2-3]. They all showcase the great utility of remote-controlled transport carts in solving real-life problems. However, there are some limitations and problems in their projects. For instance, the airport luggage carts are “bulky to the target client” and lack a security system to protect the luggage. The smart shopping cart can only carry objects lower than 5 kilograms. The motion of their objects is limited to eight defined directions instead of moving freely in all possible directions; “This luggage cart is limited to stop, forward, left and right directions, as well as limited turning” [2].

In this research, we fixed the problem of the limited direction of movement by assembling the Mecanum wheels and applying the McNamm wheel kinematics, which makes cart able to move omnidirectionally based on the direction of the rocker in the controller. The robotic arm on the cart also achieves the omnidirectional direction by the inverse kinematic robotic arm equation. In addition, we have used Fusion 360's 3D modeling software and Mixly's programming software to assist in the creation of the vehicle and the writing of its program. Additionally, we have used 3D printing to modify the vehicle's structure and appearance, making it more suitable for various application scenarios. Furthermore, we have attempted to solder the mainboard and display screen of the remote controller ourselves. During the experiment, we reduced instructional delay by programming the chips of both the transport vehicle and its remote controller, thereby making the vehicle's operation smoother. Finally, we have summarized the experimental results, discussed the future application prospects of this transport vehicle equipped with a robotic arm, and offered suggestions for further improvements.

2. Cart Assembly

The group constructed a four-wheeled cart within 3 days. All the screws and nails are manually assembled, also including the build-up of the motherboard and the soldering process. Some of the process was worth considering and discussing.

2.1. Fundamental Structure

Constructing the basic structure of the cart required the manipulation of screws, drills, and the ability to read the graphs and instructions precisely.

One thing to mention is the analysis of the Mecanum wheel. The X-type arrangement of the wheels and the O-type arrangement of the wheels will result in different logical thinking of the analysis and kinematics of the cart’s movement. Directly linked to the wheel algorithm.

2.2. Soldering

The assembly of the chip required us to solder the junctions and the motherboard. Throughout the whole process, about 60 soldiers were killed.

One of the chips was burnt out due to a poor connection of soldering, receiving a great current, and we were forced to replace one.

2.3. Wiring

As shown in Fig. 1, after the basic structure had been assembled, the wiring between the motherboard and the servos and motors also required further planning and matching up the interfaces.

At the same time, it is directly linked to the difficulty of programming as the logic of the program will be different according to the certain type of arrangement of the cart and wires.

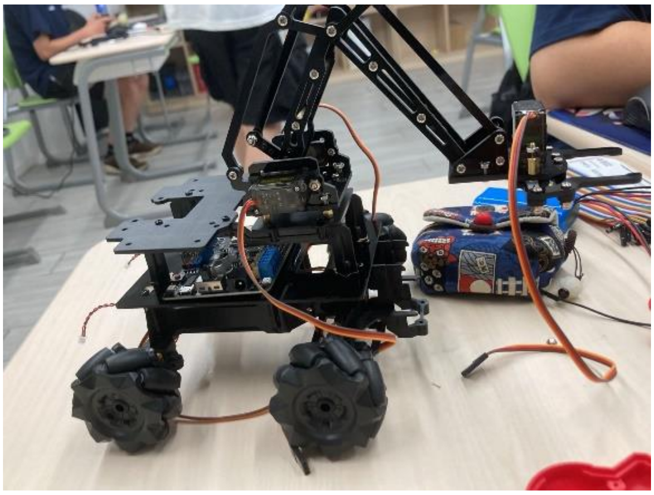

Fig. 1 The assembled cart. (Photo/Picture credit: original)

3. Cart Assembly

The program is in three sections within two different programs as we require both the controller and the cart a program each. The three sections are data collection, data execution, and data transmission.

3.1. Data Collection

3.1.1 Pull-up Resistor

As shown in Fig. 2, in the very first stage of programming, we found out that the inputting data was different from what we expected. Due to the lack of pull-up resistors, all the readings given back by the chip are a stable '0'.

The pull-up resistor affects the direction of entering current flow. It works as the current will pass the I/O while the button is not pressed. The current will, however, go in the opposite direction to GND, resulting in a '0' at I/O. Giving out the I/O a '1' while the switch is off and a '0' when the switch is on. By then, all the buttons can now present their statement of either switched on or off.

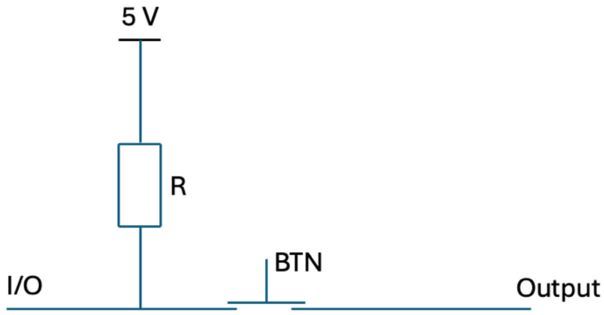

Fig. 2 A pull-up resistor. (Photo/Picture credit: original)

3.1.2 Digital and Analog Input

Digital inputs only contain values of '1' and '0', whereas the analog inputs can be varied between a certain range. In this case, it is '0' ~ '4095' for each dimension of the joystick. Overall, we have seven digital readings and four analog inputs on the controller.

3.2. Data Execution

3.2.1 Mean Calculation Algorithm

The joysticks of the controller are neither stable nor precise due to their fluctuation in the electric signal. Theoretically, the resting value of the joysticks should be maintained around 2048, whereas the value is arranged between '1700' and '1900'. Furthermore, the final plan of the controller aimed to utilize all four dimensions given by the joysticks. Methods were developed to resolve these problems. Data collected from '1700' and '1900' are corresponding to a speed of 0, and a mean calculation algorithm was developed.

A list of ten spaces available was created in the very first stage of the program. Each was given a value '1800' as the list will give a mean between the '0 velocities' range. Along with the progress of the code within each cycle, one value in the list will be replaced by the new incoming data. By then, the flow of the alternation of the mean is more fluent and continuous, also, in comparison, more stable and precise. The balance between stability and the delay of the instructions has been tested throughout and finally determined by using six values in the data and a minor delay within each cycle.

3.2.2 Kinematic Analysis of Mecanum Wheels and the Robot Arm

The aim of the program is for the user to manage the motion of the whole cart instead of each motor or servo. So, we need to find the correlation between the whole cart and each wheel. This involves the analysis of the motion of Mecanum wheels. The rollers on the wheels formed a 45-degree angle between the wheel's motion and itself, leading to a sideway force. As a result, the resultant force is 45 degrees deviated. Each wheel speed can be expressed using the motion of the cart.

$$\omega \cdot V_{front\_left} = V_x - V_y - \omega \cdot A \tag{1}$$

$$\omega \cdot V_{front\_right} = V_x + V_y + \omega \cdot A \tag{2}$$

$$\omega \cdot V_{rear\_left} = V_x + V_y - \omega \cdot A \tag{3}$$

$$\omega \cdot V_{rear_{right}} = V_x - V_y + \omega \cdot A \tag{4}$$

A=0.5*(length+ width between the centers of the wheels)

where Vx is the motion of the cart on the x-axis, representing forwards and backward. Vy is the motion of the cart along the y-axis, representing left and right shifts. ω is the rotation of the cart. Clockwise being positive and vice versa.

By utilizing the formulas in the program, we can control our cart by simply ordering the motion of the whole thing, and each wheel will automatically be converted into its required speed.

On the other hand, the robotic arm is controlled by three servos. One of them takes charge of the claw at the very end of the arm. The other two control the motion of the arm. By simplifying the situation into simple graphs, equations were found and put into use immediately. Trigonometry took the most portion in this stage. The controller was finished and could switch modes to choose whether wheel data or robot arm data had been controlled by joysticks.

3.3. Data Transmission

3.3.1 Mean calculation algorithm

It is impossible to connect a wire between the cart and the controller as it disobeyed the initial target of developing the cart. As a result, to overcome this difficulty, we decided to use Wi-Fi signal transmission. Each device that can connect the Wi-Fi has a unique MAC address that can be used to identify each device. Our cart's motherboard also has its own MAC address. Using the program, we received the address and coded our controller to send the joysticks' data to the cart. However, the transmitting data can only be in the form of a 'string,' a combination of characters. Whereas we have four isolating data that need to be transferred. Each can be a number between '0' to '4095' or simply '1' or '0'. The combination of numbers and uppercase letters was developed. We used upper case letters, including 'A' and 'E' as labels to separate the data. The receptor can decode the string, extract each data, and use them to manipulate the cart.

4. PN Junctions

A PN junction exhibits unidirectional conductivity. When a P-type semiconductor and an N-type semiconductor are fabricated on the same silicon wafer, a PN junction forms at their interface. The P-type semiconductor region contains holes (vacancies left after free electrons break away from covalent bonds, carrying a positive charge), free electrons, and negatively charged ions. In the P-type semiconductor, the concentration of holes is greater than that of free electrons, making holes the majority carriers (often referred to as "majority carriers"), while free electrons are the minority carriers (referred to as "minority carriers"). Conversely, in the N-type semiconductor region, there are free electrons (formed by intrinsic excitation breaking away from covalent bonds), holes, and positively charged ions. In the N-type semiconductor, the concentration of free electrons is higher, making free electrons the majority carriers and holes the minority carriers. The structure of the PN junctions is shown in Fig. 3.

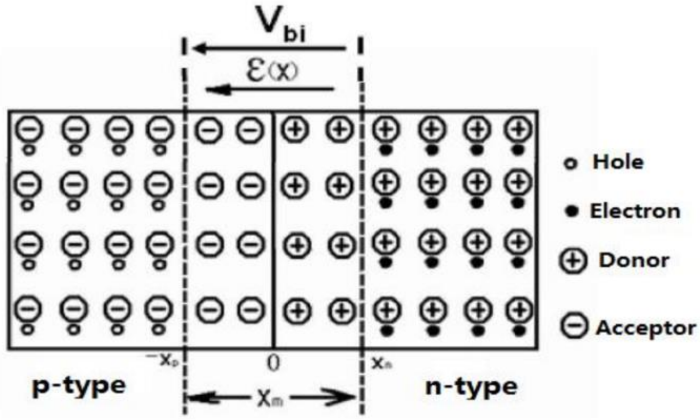

Fig. 3 Movement of charge carriers [4]

When two types of semiconductors are combined, particles will flow from the region with a higher concentration to the region with a lower concentration. This phenomenon, caused by the concentration difference, is known as diffusion. When the P-region and N-region are in contact, the majority of carriers (holes) in the P-type semiconductor will flow toward the N-region. Similarly, the majority of carriers (free electrons) in the N-type semiconductor will move toward the P-region. When these diffusing majority carriers meet at the junction, the holes and free electrons will recombine, leading to a decrease in the concentration of majority carriers at the junction. As a result, a negative ion region will form in the P-region, and a positive ion region will form in the N-region at the junction. These ion regions, which cannot move, are known as the space charge region, leading to the formation of an internal electric field. To prevent further diffusion, the space charge region widens, and the internal electric field strengthens, with its direction pointing from the N-region to the P-region. The process is clearly shown in Fig. 4.

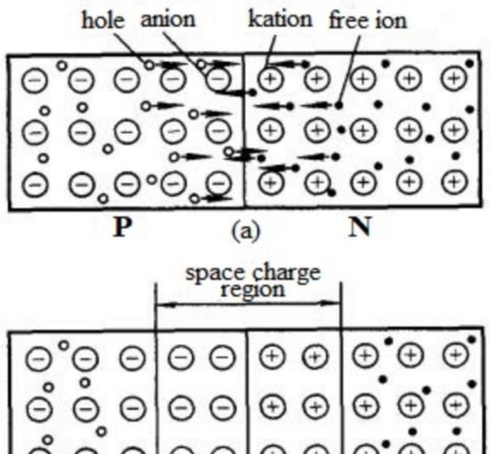

Fig. 4 PN junction in balance [5]

In the absence of external voltage, the PN junction remains in a state of equilibrium. However, when an external voltage is applied, the equilibrium of the PN junction is disrupted, causing current to flow through the junction. This leads to the conclusion that under different voltage conditions, the PN junction will exhibit distinctly different behaviors, demonstrating its unidirectional conductivity.

When the positive terminal of a power supply is connected to the P-side of the PN junction and the negative terminal to the N-side, the PN junction is said to have an applied forward voltage, also known as forward bias. The external electric field generated by the power supply pushes the majority of carriers toward the space charge region, thereby narrowing it. Due to the power supply, the majority of carriers continuously undergo this diffusion process, resulting in a forward current.

Conversely, when the positive terminal of the power supply is connected to the N-side of the PN junction and the negative terminal to the P-side, the PN junction has an applied reverse voltage, also known as reverse bias. In this case, the external electric field formed by the power supply enhances the space charge region, causing it to widen. This prevents diffusion and enhances drift (the movement caused by minority carriers), but since the number of minority carriers is extremely small, the current generated by this movement is usually negligible. Hence, the PN junction is in a cutoff state.

5. Problem Occurrence and Solutions

Unfortunately, a problem occurred abruptly during our project for the transport cart, besides those expected, like the common grammatical and logical errors in programming.

5.1. ESP32 Mainboard Damage

The problem is that the motor suddenly stopped working after it had spun successfully for a while.

5.1.1 Self-protection mechanism aspect

To deduce the source of the error, after a reasonable waiting time, we reinputted the code into the cart chip to check whether the self-protection mechanism, which shuts down the whole circuit board by overrunning the code, was activated. This is achieved by the EPS32 in-built temperature sensor, which measures the voltage between a certain junction to determine the internal ambient temperature, which can trigger the cooling mechanism, including shutting down if exceeding the temperature threshold [6]. But the report interface at the programming software output “uploaded unsuccessfully,” which demonstrates that it was the circuit board that went wrong.

5.1.2 Motor circuit board aspect

The circuit board on the cart can be split into three sections: The motor section, the Power section, and the ESP32 Mainboard. First, we checked the circuit related to the Motor, where there were four pairs of wires used to connect the four motors with the four fairs of pin holes on the circuit board by the blinding posts on the board site and the cold-pressed terminals on the wires site. However, the cold-pressed terminals can be broken off easily by shaking the wires, which may happen when we hold the cart. After checking, it was a pair of cold-pressed terminals for the right backward motor bent excessively. We renewed the structure and reloaded the code, which still turned to “uploaded unsuccessfully.” We next shifted to the power circuit board section.

5.1.3 Power circuit board aspect

By charging the circuit board, we saw that the LED turned on to blue, which means, based on the circuit diagram distributed by our teacher, there is nothing to do with the power section of the circuit board, such as the physical wear of wire connection interface that can be changed easily by soldering, in which LEC is located.

5.2. ESP32 Main Board

What was left now was the ESP32 Mainboard and Motor section. Theoretically, pressing the EN button at the circuit board will force the ESP32 to restart when there is a message output at the Integrated development environment indicating the subsequent boot process; “Reset button: pressing this button (EN) resets the system.” [7], and “After reset, the second line printed by the ESP32 ROM (at 115200bps) is a reset & boot mode message” [8].

Here is an example of the output message after the reset:

ETS Jun 8 2016 00:22:57

rst:0x1 (POWERON_RESET), boot:0x3 (DOWNLODING_BOOT(UART0/UART1/SDIO_REI_RE0_V2)) [9].

from website: https://docs.espressif.com/projects/esptool/en/latest/esp32/advanced-topics/boot-mode-selection.html.

Specifically, the pin used for the reset is CHIP_PU, which is pulled high (higher than VIL_nRST) to first activate the chip and then pulled down to reset the chip. Fig. 5 visualizes the process:

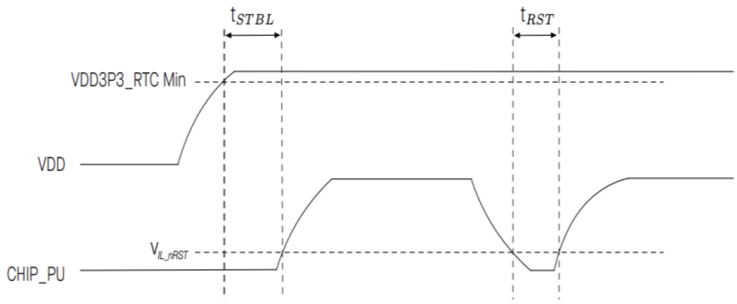

Fig. 5 Visualization of timing parameters for power-up and reset [10]

tSTBL: Time reserved for the 3.3 V rails to stabilize before the CHIP_PU pin is pulled high to activate the chip;

tRST: Time reserved for CHIP_PU to stay below VIL_nRST to reset the chip.

VIL_nRST: Low-level input voltage of CHIP_PU to shut down the chip.

However, when we pressed the button with the battery connected, no message came out, showing that the bootloader in the esp32 microcontroller unit wasn't invoked, and in other words, the esp32 mainboard was damaged.

Reason Analysis:

In engineering, it is hard to speculate the reason for an abrupt instrument malfunction, as we can't restore the previous scene to seek the specific wrong actions within millions of actions that lead to the problem. Alternatively, we could only guess.

By finding that the ESP32 mainboard temperature was higher than usual and recalling that there was no physical crash with the board, a short circuit likely happened at some point.

5.3. Short Circuit

5.3.1 Theory

A short circuit is a time when current in an electrical circuit flow with little or no presence of resistance. By Ohm's law:

$$R = \frac{V}{I} \tag{5}$$

With the constant voltage level, when there was little resistance, the current surged substantially so that the wires were unable to handle the enormous current, which then flowed out of them to the outside electronic components that had resistance. By Joule's law:

$$P = I^2 R \tag{6}$$

The power generated would be large too, all exerting on the electronic component and thus became damaged.

5.3.2 Formation of the Short Circuit

After careful inference, the short circuit may form when the tip of the screwdriver, whose resistance is very low, was accidentally inserted into the wire hole without the battery off during the repair of the broken c at the motor stage.

5.3.3 Refection

Although the result was dramatic, it informs us of the importance of turning off the battery when repairing any electronic device and the vulnerability of the equipment in electrical and electronic engineering, which means we should be careful.

Based on the new functional circuit by which information can be successfully transmitted between the transport cart, we can finally fulfill the motion of the cart in all directions and the robotic arm freely controlled by our remote controller through the programming.

6. Conclusion

The whole task is divided into four main parts as demonstrated: soldering, programming, research of PN junction, and analysis of the unexpected problem.

The outcome shows that the motion of the transport cart and the robotic arm on it can be controlled successfully and freely in all reachable directions and destinations. The entire research has enhanced my understanding and ability in soldering, programming, practical use of mathematical and physics formulas, knowledge of PIN junction and circuit, and related problem-solving and practical operation, which are very important in robotic and electrical engineering. The transport cart and the robotic arm can both be used as the base for more complicated applications of robotics, whether for existing uses, including robot vacuum cleaners and excavators, or uses under exploration, like artificial limbs for the disabled and more advanced underwater robots for deeper marine research.

However, due to the availability of technology, there was a delay between the controller and the operation of the cart and arm by slow wireless transmission in the air. In future research, this issue will be further studied.

Authors Contribution

All the authors contributed equally and their names were listed in alphabetical order.

References

[1] Ritzkal R, Prakosa B A, Munandar I P A, et al. Remote-controlled Bluetooth-enabled smart shopping cart: prototype and evaluation. Journal homepage: Ingénierie des Systèmes d’Information, 2023, 28(5): 1221-1228.

[2] Abas M C A, Cerna P D. Design and Development of Microcontroller Remote Controlled Airport Luggage Cart. Science and Education, 2017, 4(2): 49-54.

[3] Weinmann O, Bitzer F, Boos N, et al. Maneuvering of long trucks by remote control. ATZ Worldwide, 2014, 116(9): 4-9.

[4] Dong_gwu_0075M_12027.pdf, Last accessed 2024/10/25, http:// Users/ericjiaoyuheng/Downloads/Dong_gwu_0075M_12027.pdf.

[5] How is PN Junction Formed? Basics and Examples, Last accessed 2024/10/25, https://www.jakelectronics.com/blog/how-is-pn-junction-formed-basics-and-examples.

[6] Testing ESP32 capabilities with Internal Temperature Sensor, Last accessed 2024/9/25, https://www.espboards.dev/blog/esp32-inbuilt-temperature-sensor/.

[7] ESP32-DevKitC Getting Started Guide — ESP-IDF Programming Guide v3.0.8-30-gf3704f027 documentation, Last accessed 2024/9/29, https://docs.espressif.com/projects/esp-idf/en/release-v3.0/get-started/get-started-devkitc.html.

[8] ESP32-DevKitC Getting Started Guide — ESP-IDF Programming Guide v3.0.8-30-gf3704f027 documentation, Last accessed 2024/9/29, https://docs.espressif.com/projects/esp-idf/en/release-v3.0/get-started/get-started-devkitc.html.

[9] Boot Mode Selection - ESP32 - — esptool.py latest documentation, Last accessed 2024/9/29, https://docs.espressif.com/projects/esptool/en/latest/esp32/advanced-topics/boot-mode-selection.html.

[10] esp32_datasheet_en.pdf, Last accessed 2024/9/30, https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf.