

Article

Design of Refueling Control System with Multi-Control Terminal Based on Network

Bowen Zeng¹, Zhongmin Zhang^{2,*}, Xiao Zhang¹ and Jing Xiang¹

¹ College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin 150001, China; zengbowen@hrbeu.edu.cn (B.Z.); zhangx1997@126.com (X.Z.); xiangjing@hrbeu.edu.cn (J.X.)

² College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China

* Correspondence: zhangzhongmin@hrbeu.edu.cn

Abstract: This paper discusses a network-based refueling control system with multiple control terminals that achieves the control input and display of near-aircraft position, refueling station, and centralized console. The system can achieve the closed-loop control of the refueling input flow, automatic control of the fixed refueling amount, and collection of environmental parameters, such as pipeline pressure, according to the refueling aircraft model. Moreover, the system can dynamically display parameter information of the refueling process in real time on three terminals. To meet the requirements of system functions and the real-time performance of tasks, this study selected the LPC2378 controller with the ARM core. The controller uses the $\mu\text{C}/\text{OS-II}$ embedded real-time operating system as the software operating platform. To achieve the special requirements of the explosion-proof system, the explosion-proof 485 bus was employed to communicate with the display control unit (DCU) of the intrinsically safe contact aircraft stand and refueling station. The communication adopts the dual-redundant Ethernet communication and full-duplex 485 communication. To avoid the influence of external equipment on the control circuit, the power isolation method was adopted to ensure the stability of the system. After comprehensive debugging of hardware and software, we deem that the system has good performance and can be applied in engineering.

Keywords: refueling control; embedded system design; 485 communication; ethernet communication; $\mu\text{C}/\text{OS-II}$



Citation: Zeng, B.; Zhang, Z.; Zhang, X.; Xiang, J. Design of Refueling Control System with Multi-Control Terminal Based on Network. *Electronics* **2021**, *10*, 1559. <https://doi.org/10.3390/electronics10131559>

Academic Editor: Manohar Das

Received: 8 April 2021

Accepted: 19 June 2021

Published: 28 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the advancement of science and technology, automated aircraft refueling methods have gradually developed. The automatic refueling method can save manpower and material resources and can accurately control the refueling amount concurrently with the current expansion of the scale of civil airports and the diversification of aircraft types. It is worth studying to design a reliable aircraft refueling control system that can refuel many different types of aircraft.

In recent years, research about refueling systems has developed rapidly. Many scholars have studied the refueling control system. The study on refueling systems includes an Autonomous Aerial Refueling (AAR) system, system operational of a refueling station, and refueling station design. Dong et al. [1] studied the AAR system and proposed a failsafe mechanism. A simulation environment verified the feasibility of this mechanism. Sun et al. [2] presented a position measurement system for the AAR system. The method has strong robustness. In addition, there are many other studies on aerial refueling, including binocular-vision-based AAR platform [3], on-board visual navigation system [4], and pose estimation of AAR system through visual measurement [5]. Brisse et al. [6] investigated the system operation of refueling stations. They described the H₂ refueling station and gave the structure of the station on-site. The development of refueling station design was

studied by Riedl SM [7]. Simultaneously, some other investigations about the refueling system were also performed. Xu et al. [8] studied the optimal operational strategy for a hybrid refueling station. Williamson et al. [9] focused on the sensor fusion applied to the AAR system. Chen et al. [10] presented an AAR ground test demonstration with a sensor description. Cai et al. [11] described a hardware-in-loop simulation system during aerial refueling. They established a simulation and experiment platform and conducted tests. The results show that the design of the hardware system is effective. Plaza and Santos [12] established a general and optimized procedure for the initial commercial aircraft refueling. They designed an optimized algorithm and verified the algorithm with a case. The results show that the designed algorithm is effective and optimal.

Those literature views carried out different studies on refueling systems. However, these investigations did not give detailed hardware design and on-site debugging. In our study, we design a refueling control system with a multi-control terminal based on a network. The system adopts the hybrid communication methods include 485 communication and Ethernet communication mode. The detailed hardware design is given in this paper.

The main contributions of this paper can be described as follows: (1) The refueling system can control different refueling flow rates according to different aircraft models and equipment, which improves the refueling efficiency. (2) A refueling system with multiple control terminals is designed. The refueling system includes a refueling station MCU, a refueling station DCU, a near-aircraft position DCU. Moreover, the refueling system has a centralized control room supervision and management unit. The system is easy to operate, saves manpower, and also improves the degree of automation. (3) The fueling system uses redundant Ethernet, full-duplex 485 that is 422 communication, and uses a reliable communication protocol 3964(R). The communication ways ensure the reliability of system communication. (4) To prevent the influence of external equipment on the control circuit, the system adopts the power isolation to ensure the stability of the system.

The outline of the remainder of this article is organized as follows. Section 2 introduces the system function and overall plan; Section 3 summarizes the hardware design of the main control unit (MCU); system software design is described in Section 4; experiments are illustrated in Section 5; and, finally, Section 6 summarizes the current work.

2. System Function and Overall Plan

The system can achieve the closed-loop control of the refueling input flow rate, the automatic control of the set refueling volume, and the collection and alarm of environmental parameters such as pipeline pressure according to the refueling model. Moreover, the system can dynamically display the refueling process information in real-time on three terminals. Figure 1 shows the overall plan. Since the equipment works in an inflammable and explosive environment with refueling, the MCU of refueling is designed as explosion-proof equipment, and the DCU is intrinsically safe. The two ways of communication are designed separately. The 485 communication form of intrinsic safety and explosion-proof conversion is used to connect for control and display. The system consists of five parts: the MCU for refueling, the DCP of the near-aircraft, the control unit of the refueling station, the centralized display control console, and the on-site sensors. The main function of the refueling MCU is to control the corresponding flow rate according to the type of refueling equipment (such as the model). The system can realize the automatic control of the preset refueling volume, the oil outlet pipeline pressure collection, alarm, and valve closing linkage control. Moreover, the system can provide the power required by various peripherals. The refueling station DCU and the centralized console can be set and displayed: refueling model, refueling amount, and system parameters; the near-aircraft DCU can display system information and simple start-stop function input buttons without setting functions. The priority of the control parameters set by the DCU of the local refueling station is higher than that of the centralized console, and the local refueling DCU can release the priority to the centralized console by pressing a button. The refueling station DCU and the near-aircraft DCU communicate with the MCU through 422 bus. The

centralized control center communicates with the MCU through dual redundant Ethernet. The MCU of refueling is the core part of the system.

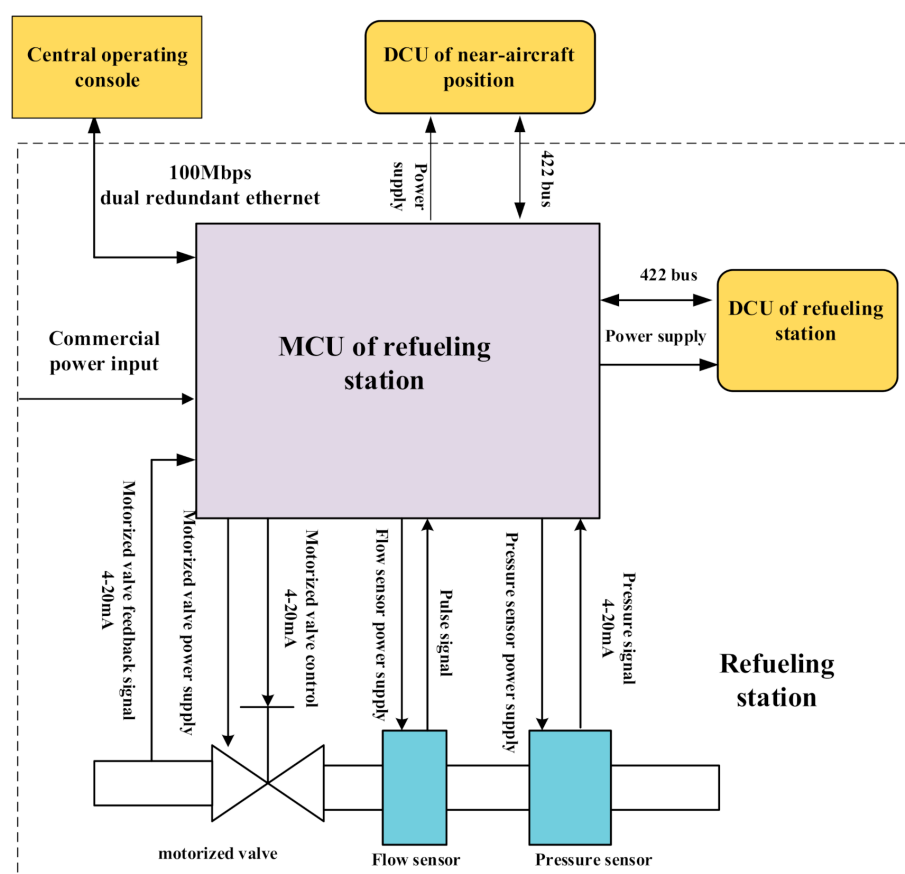


Figure 1. System structure. MCU: main control unit; DCU: display control unit.

3. Hardware Design of MCU

3.1. Overall Introduction of Hardware

The MCU adopts LPC2378 based on the ARM core, which has a powerful communication function and industrial control ability. The hardware composition of the MCU is shown in Figure 2. The internal circuit of the refueling MCU consists of two main circuit boards: the former mainly completes the power supply of each module and sensor of the system and standardizes the external input and output signals; the other carries on the standard signal sampling, processing, and output, as shown in Figure 3.

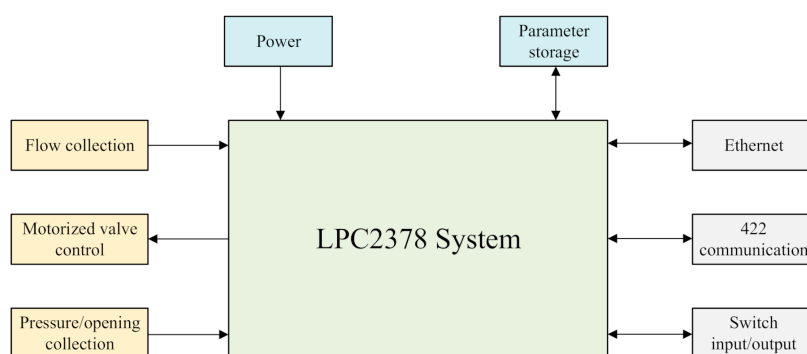


Figure 2. Hardware framework of MCU.

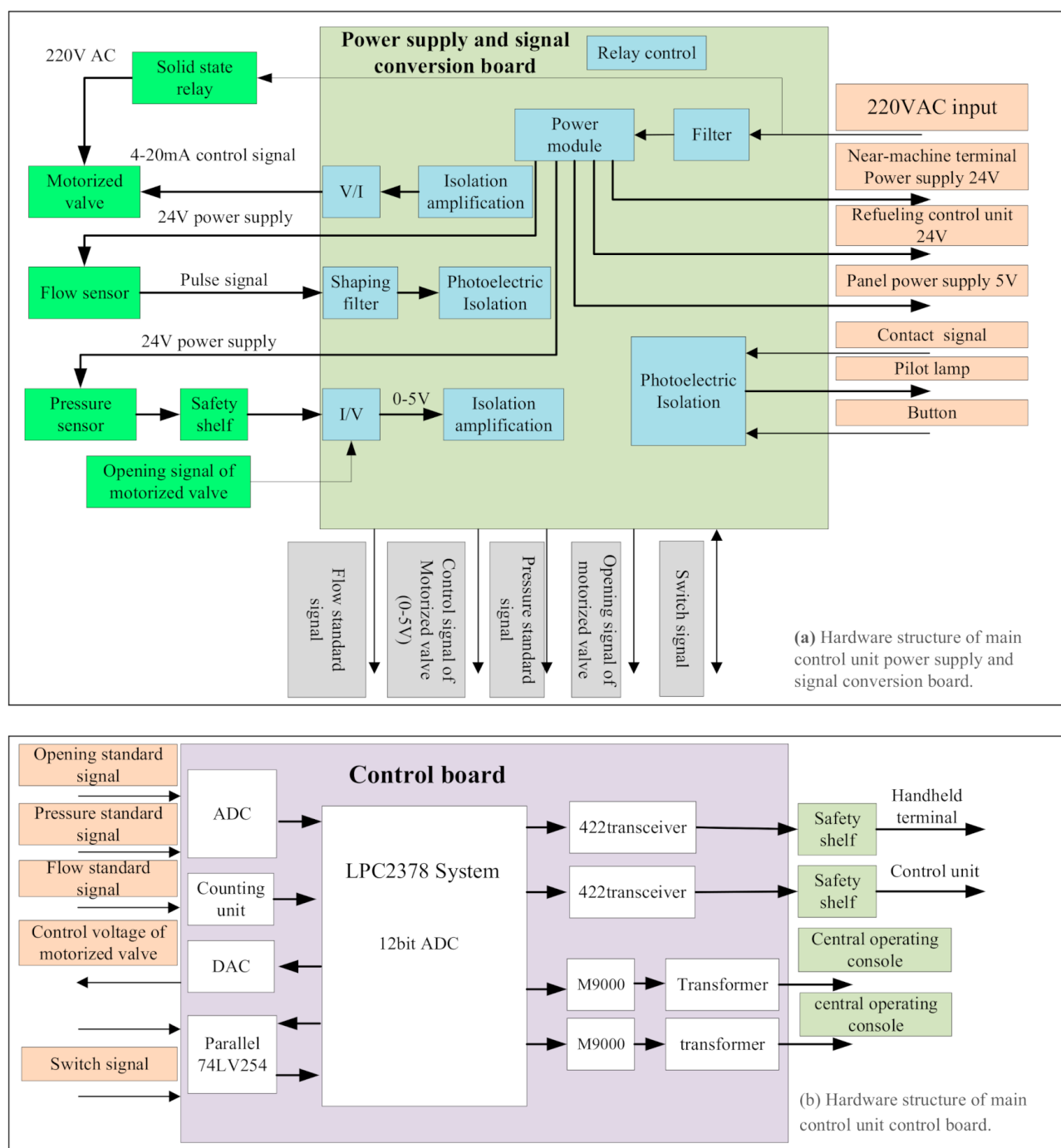


Figure 3. Internal circuit structure of hardware.

3.2. Main Hardware Design

The main hardware design includes the following steps:

Step 1. Description of the minimum system.

The external crystal oscillator is 30 MHz, and that is 100 MHz after the phase-locked loop frequency doubling. In the minimum system, the control core requires a 1.8 V power supply, and the I/O port requires a 3.3 V power supply. Two sets of power supplies are provided by the regulator chip AMS1117. To meet system control tasks and the memory requirements of operating systems, the external bus of LPC2378 expands the memory space. The 4M SRAM memory IS61WV102416BLL is selected. The 74LVC16245A device is added

between the SRAM data line and the LPC2378 bus, which plays a two-way buffer. The description of the minimum system is shown in Figure 4a.

Step 2. Channel of pulse flow acquisition.

The external counting unit on the MCU chip LPC2378 is used to collect the flow sensor signal. The output information of the flow sensor is as follows: the amplitude is 0–10 V, the duty cycle is 50%, and the frequency is 0–850 HZ pulse signal. The circuit of the flow collection preprocessing section is shown in Figure 5a.

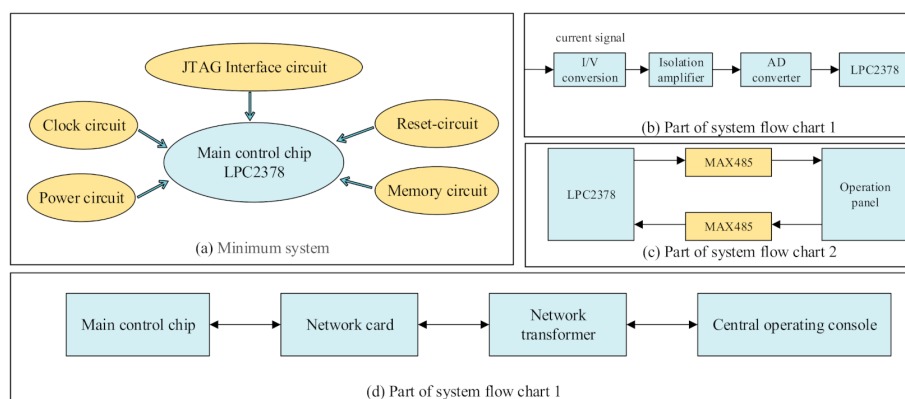


Figure 4. Composition of the MCU.

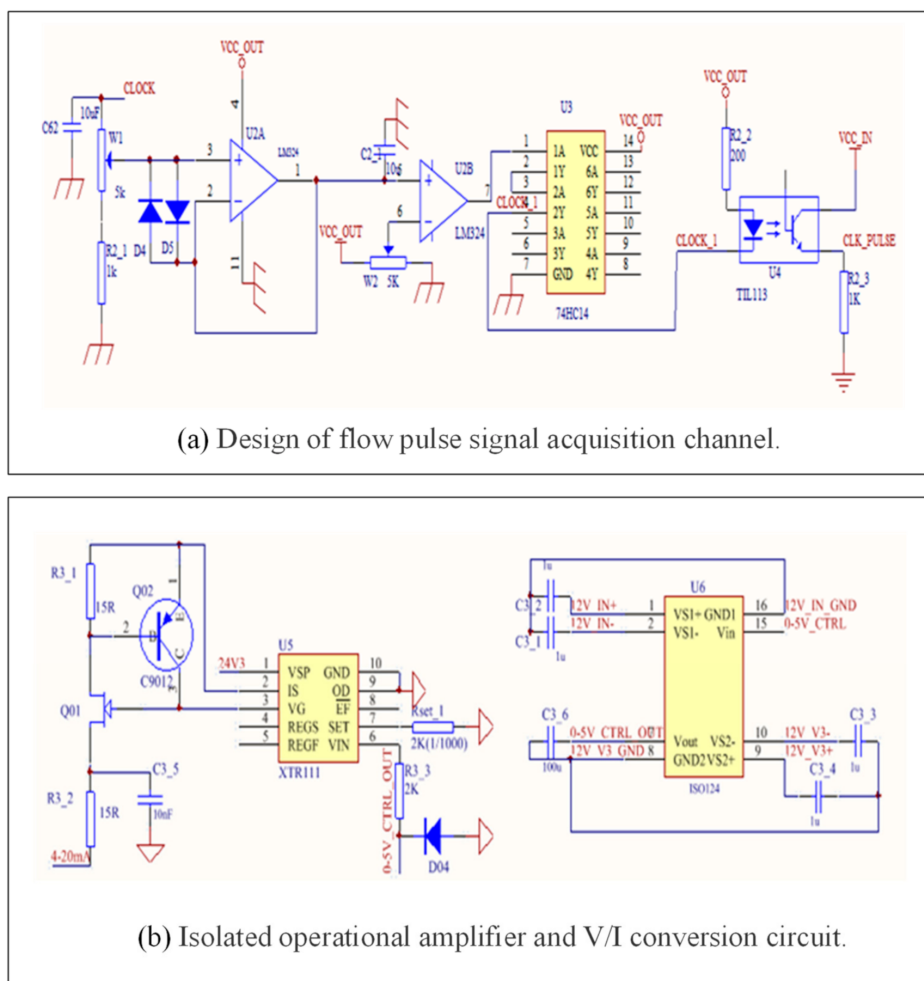


Figure 5. Composition of the MCU (circuit).

From the left signal input, the sliding resistor and a fixed resistor constitute a voltage divider circuit; the two diodes are in reverse parallel to stabilize the voltage and limit the amplitude. The first operational amplifier of LM324 is used as the isolation buffer stage. The second set of operational amplifiers constitute a voltage comparator to limit the input voltage, which can eliminate the high frequency and low amplitude pulse signal in the process of starting. The shaping effect of 74HC14 with Schmitt triggers the LM324 output signal. The signal passes through the optocoupler TL113 from outside to the acquisition end. The two sets of power supply of external working shaping circuit and internal pulse acquisition circuit are independent, and the power supply of external working circuit and pulse flow sensor is common.

Step 3. Electric valve control.

Using 12-bit DAC, SPI interface, output 0–5 V voltage, through ISO122P, V/I conversion, output 4–20 mA current control valve opening. Isolated operational amplifier and V/I conversion as shown in Figure 5b, the controller output voltage signal 0–5 V from DAC output ISO output, through XTR111 and auxiliary circuit conversion to 4–20 mA current signal. Two groups of independent three-terminal power supply ± 12 V and ground are isolated, one group of internal power supply and DAC are ground, and the other group of external power supply and I/V are ground. This separates the peripherals from the controller to prevent the phase of the external circuit and the digital control circuit mutual influence.

Step 4. Analog signal acquisition channel.

Pressure and electric valve opening signal acquisition LPC2378 IO port extension 8 channels 12 bit serial ADC7888. The sensor signal input is 4–20 mA, first through 250 Ω precision resistance conversion to 1–5 V voltage, through the isolation amplifier ISO122P input ADC.

Step 5. Full duplex 485 communication.

The MCU should communicate with two display and control terminal devices, full-duplex communication with two SCIs, and two 485 channels on LPC2378.

Step 6. Redundant Ethernet communication.

Because LPC2378 has no network interface card (NIC) controller, using its open external data bus, a control bus, an address bus to expand the DM9000 network controller, and then through the transceiver and Ethernet bus connection.

Step 7. Parameter storage.

Using I2C bus to expand 16 K-bit EEPROM.

4. System Software Design

The system collects the flow rate and pressure signal to realize the flow control and preset fuel automatic control. Simultaneously, RS-485 bus communication and Ethernet communication are also required to realize. The system has large tasks and requires high real-time performance. Therefore, the corresponding program design and development work are carried out on the embedded real-time operating system $\mu C/OS-II$.

The system software design includes the following: (1) the task allocation and control logic of embedded software based on real-time operating system $\mu C/OS-II$; (2) communication mechanism of each communication module; and (3) the algorithm of flow and flow velocity control

4.1. Embedded Software Design

The control system is divided into seven tasks according to the functional requirements of the system, which consists of 0–6, and the priority is reduced in turn. The division and connection of tasks are shown in Figure 6.

- (1) Task 0 is the highest priority and mainly completes the flow, pressure collection, and velocity closed-loop control. The sampling period is 200 ms in the system. The process information of the control task is transmitted to Tasks 3, 4, and 5 through three message queues to realize the information output of the three terminals.

- (2) Task 1 is the second priority and mainly achieves the network packet query reception and redundant switching of Ethernet main equipment network.
- (3) Task 2 receives the UDP message of the application layer, and the working message queue of Task 1 is passed to Task 2. Task 2 passes control commands to Task 0 through global variables.
- (4) Task 3 sends site control information UDP message to the central console. Task 3 receives information from Task 0 through the queue buffer.
- (5) Task 4, an information exchange with the DCU of refueling station mutual, including sending information and receiving setting instructions. The queue buffer receives control information from Task 0 and sets control instructions for Task 0 through global variables.
- (6) Task 5 interacts with the near-machine DCU. The queue buffer receives control information from Task 0 and sends simple control instructions to Task 0 through global variables.
- (7) Task 6 is a low real-time task, such as indicator light control.

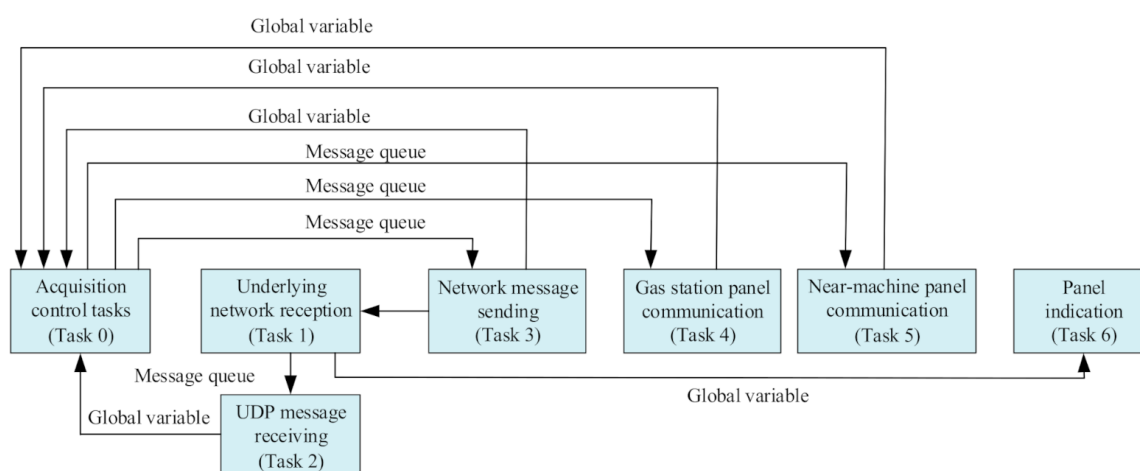


Figure 6. Division and connection among system tasks:

4.2. Communication Part

4.2.1. The 485 Communication

The MCU needs to communicate with two display control terminals, and the physical layer uses 485 communication. The transceiver between the MCU and the display control of the refueling station or between the MCU and the near-computer display control uses two independent channels to form a full duplex and uses 3964(R) protocol. The data-link layer uses a 3964 protocol to ensure the reliability of communication. The MCU characters of the 3964 protocol are shown in Table 1. The communication process of the 3964(R) protocol is shown in Figure 7.

Table 1. Basic control character description of 3964(R).

No.	Control Character	Value	Explanation
1	STX	02 H	The starting point of the text being transmitted
2	DLE	10 H	Positive response
3	ETX	03 H	The endpoint of a transmitted message
4	BCC		Longitudinal parity test results of data(DLE and ETX)
5	NAK	15 H	Negative response

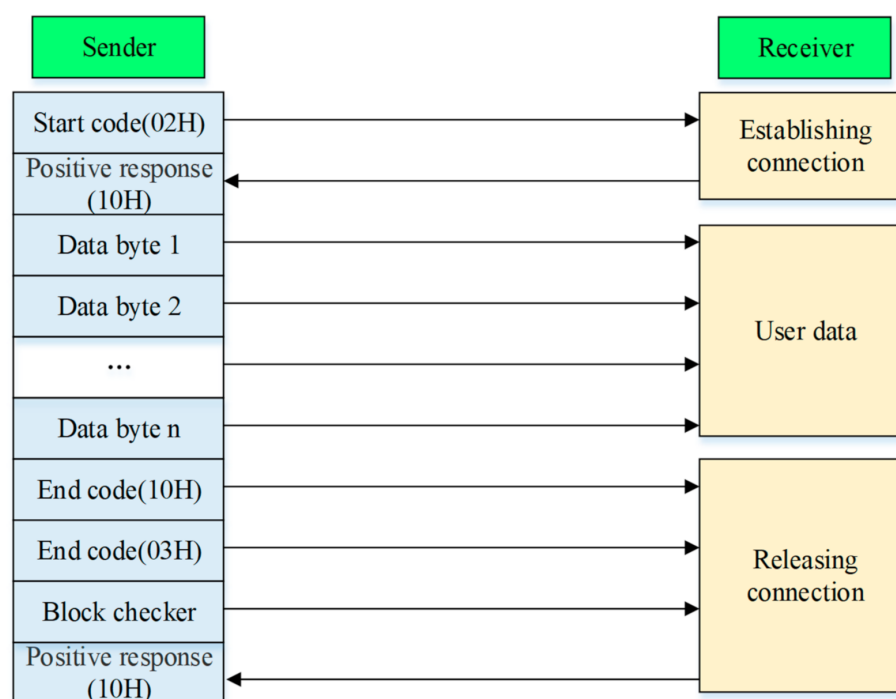


Figure 7. Diagram of 3964(R) protocol communication process.

The response delay time (ADT) of the start code should not exceed 100 ms; otherwise, the connection will fail. Moreover, the interval of transmitting characters should not exceed 20 ms; otherwise, the data transmission is regarded as a failure. At most, 6 attempts are made for the initiating connection part, as shown in Figure 8a. The connection is successfully established once, and there are 6 unsuccessful attempts to send data, that is, 36 unsuccessful attempts to send data at most. The flow of the data-sending function is shown in Figure 8b.

In the receiving and sending function, the main work is to splice messages. The key point is to find the ending characters 10H and 03 H. If the 10 H in the data field is sent twice, one 10 H must be eliminated at the end, and then the data verification is performed. According to the verification results, the response information is sent. The process of sending the message function is shown in Figure 8c.

The function of receiving and sending data is required when receiving and sending a message. The underlying function of serial data sending is directly called when sending data; When receiving data, the first is to wait for the serial port message in the task and use the UCosII operating system's generation delay OSMboxPend that used in Figure 9. Tasks 4 and 5 are the communication tasks between the MCU and the refueling station panel. According to the 3964 protocol, the communication program for message sending and receiving is written. The basic content of Task 4 and Task 5 is similar, and their flowchart is shown in Figure 8d.

4.2.2. Dual-Redundant NIC

The dual-redundant NIC is an important part of the system. The NIC in the network is composed of two mirrored NICs on the hardware. Each NIC has the same IP MAC address and supports all functions of the ordinary NIC. Simultaneously, only one channel is working, and the other channel is in a redundant state. The dual-redundant channel switching is realized in the dual-redundant NIC device driver. For the network layer and above, the dual-redundant channel switching is completely transparent, which improves the flexibility and reliability of the module. The network topology of dual-redundant networks is shown in Figure 9.

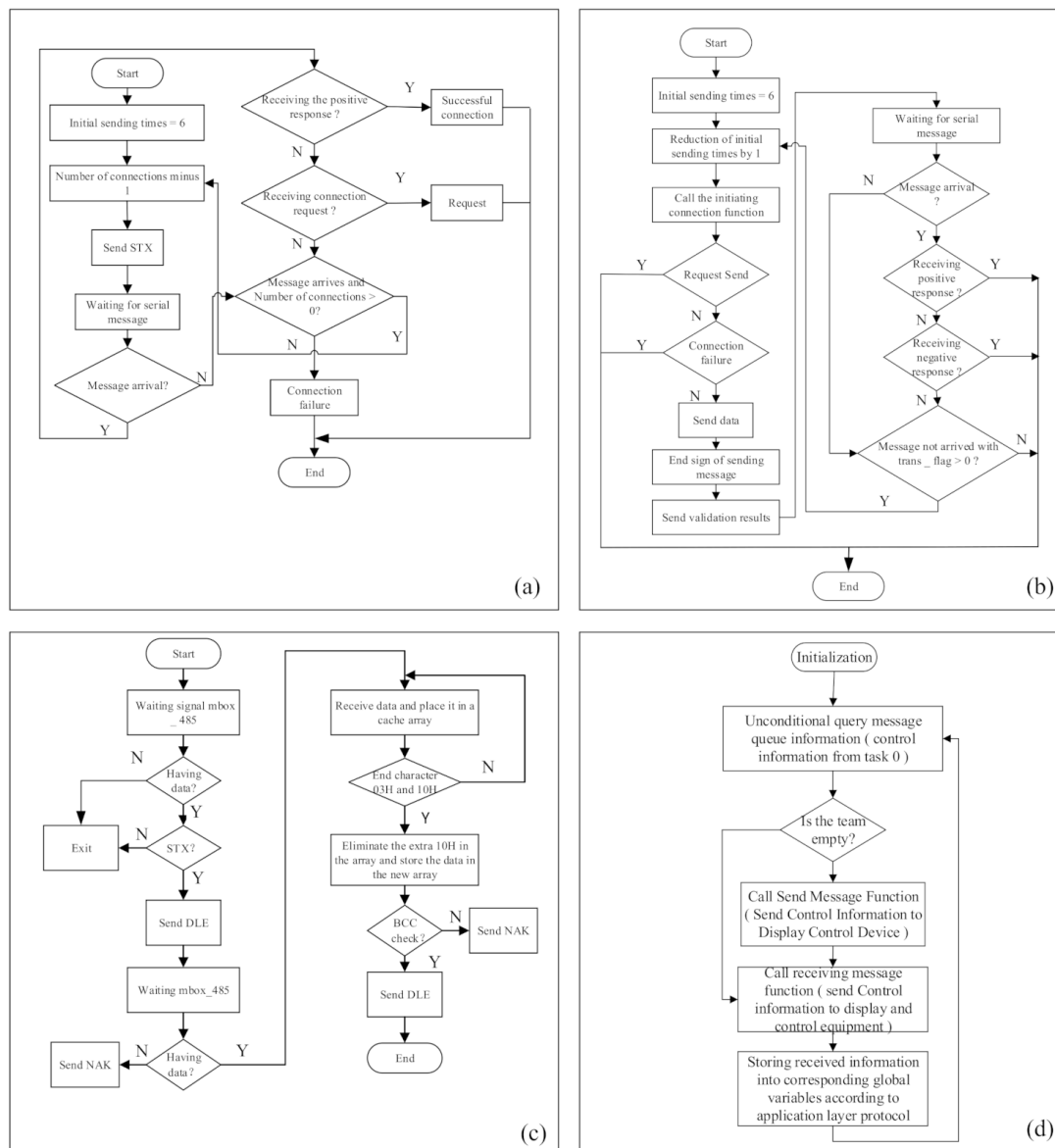


Figure 8. Flowchart of task execution: (a) initiating connection flowchart, (b) partial flowchart of sending message function, (c) partial flowchart of receiving message function, and (d) flowchart of Tasks 4 and 5.

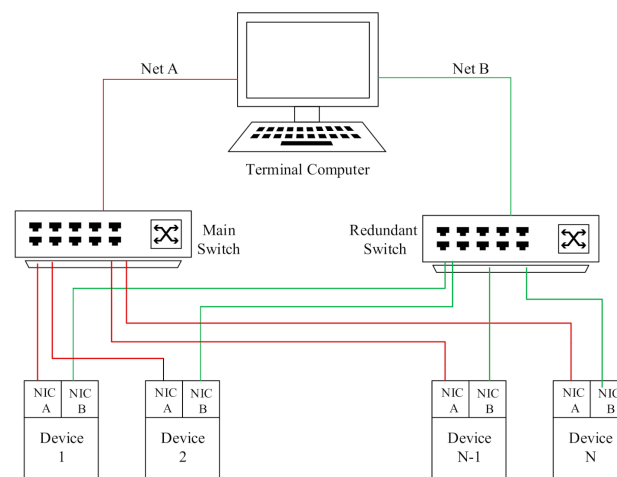


Figure 9. Topology structures of dual-redundant network.

Full fault detection and switching are provided in the driver. The dual-redundant NIC driver belongs to the link-layer software of the TCP/IP protocol stack, which has all the functions of the standard NIC driver. The dual-redundant NIC realizes the network redundancy function according to the requirements of the network. In addition to normal switching, the driver also has the functions of passive switching and abnormal message switching.

(1) Normal switching

Normal switching refers to the active switching of the NIC. When the NIC detects that the line on the current working channel is disconnected, the standby channel needs to be used as the working channel, and the original working channel is used as the standby channel.

(2) Passive switching

The passive switching means that when the dual-redundant switchboard switches to the standby channel, the dual-redundant NIC also needs to switch to the standby channel at a certain time. Then, the state that all the NICs and switches in the system work on the standby channel.

(3) Increased switching of abnormal messages

Due to poor contact between network lines and sockets, or the decline of electrical properties of NICs and network processing modules, the communication quality will be significantly reduced, thus affecting the normal communication of the network. This kind of fault can be identified by the software in the unit time statistics abnormal message. An exception message addition can be judged by the error message counter inside the chip, and channel switching occurs when the counter exceeds the threshold.

4.3. Sampling Control Part

4.3.1. Flow Control

(4) Flow collection: the external pulse counter is used to calculate the number of external pulses; the sampling period is set to 200 ms; the counter is 16 bits. Two situations need to be considered when accumulative fueling statistics:

- (a) When the current 16-bit calculated value is greater than the previous 16-bit value, the cumulative pulse number N_T in 200 ms can be expressed as follows:

$$N_T = CNT_{r_now} - CNT_{r_pre} \quad (1)$$

where CNT_{r_now} is the value of the current sampling period counter, and CNT_{r_pre} is the value of the last sampling period counter.

- (b) When the current 16-bit value is less than the previous 16-bit value within 200 ms, the accumulated pulses within 200 ms is as follows:

$$N_T = 65535 - CNT_{r_pre} + CNT_{r_now} + 1 \quad (2)$$

The refueling amount needs to sum the number of pulses in all sampling periods and then convert the pulse into flow:

$$Oil_{total} = \sum_{i=0}^k N_T(i) / Cnt_num \quad (3)$$

where Cnt_num is the number of pulses per liter of oil determined by a flow sensor, generally 37 pulses per liter.

4.3.2. Flow Velocity Control

The real-time flow velocity can be expressed as follows:

$$Flu = 5 \times N_T = 60/Cnt_num \text{ (L/min)} \quad (4)$$

The measurement range of the flow sensor is 0–1000 L/min, and the actual flow peak is 850 L/min. The PID controller with the dead zone for velocity control is shown in Figure 10.

In Figure 10, $r(k)$ is the flow velocity allowed for a given type of refueling tanker, that is, the given value; $r(k)$ ranges from 250 to 700 L/min. To prevent the small range of the electric valve, the flow dead zone is set at 5% of the setting value, that is, $|\varepsilon| = r \times 5\%$. This is the flow velocity control accuracy of the system. The actuator is the motorized valve. The plant is the pipeline. Moreover, $y(t)$ is the measurement value of flow velocity; $u(k)$ is the output of the controller, and corresponding to the opening of the electric valve.

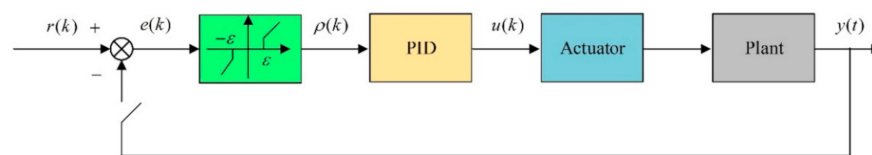


Figure 10. PID with dead-zone.

The increment PID controller can be written as follows:

$$\Delta u(k) = K_p[(e_k - e_{k-1}) + \frac{T}{T_i}e_k + \frac{T_d}{T}(e_k - 2e_{k-1} + e_{k-2})] \quad (5)$$

In Task 0, the sampling time (T) is 200 ms. In the program design, the value of the sampling period is not considered, and the controller parameters can be set through Equation (6) as follows:

$$u(k) = K_p e(k) + K_i \sum_{n=0}^k e(n) + K_d(e(k) - e(k-1)) \quad (6)$$

$$u(k-1) = K_p e(k-1) + K_i \sum_{n=0}^{k-1} e(n) + K_d(e(k-1) - e(k-2)) \quad (7)$$

$$\Delta u(k) = u(k) - u(k-1) = K_p(e(k) - e(k-1)) + K_i e(k) + K_d(e(k) - 2e(k-1) + e(k-2)) \quad (8)$$

The output of the controller is as follows:

$$u(k) = \Delta u(k) + u(k-1) \quad (9)$$

where $u(k)$ is the setting opening value of the motorized valve servo system. Firstly, the output voltage of 12-bit DAC is 1–5 V, and then the V/I is converted to 4–20 mA for the electric valve. The output control range, $u(k)$, is 819–4095.

5. Experiments

The communication process between the MCU and the DCU or near-machine unit of the refueling station is shown in Figure 11. The DCU sends the handshake message to contact the MCU after the equipment is powered actively. Then, the DCU sends the self-check message after the MCU responds. After the connection is established, the self-check result of the MCU is sent to the equipment to show the self-check state of the equipment in two DCUs.

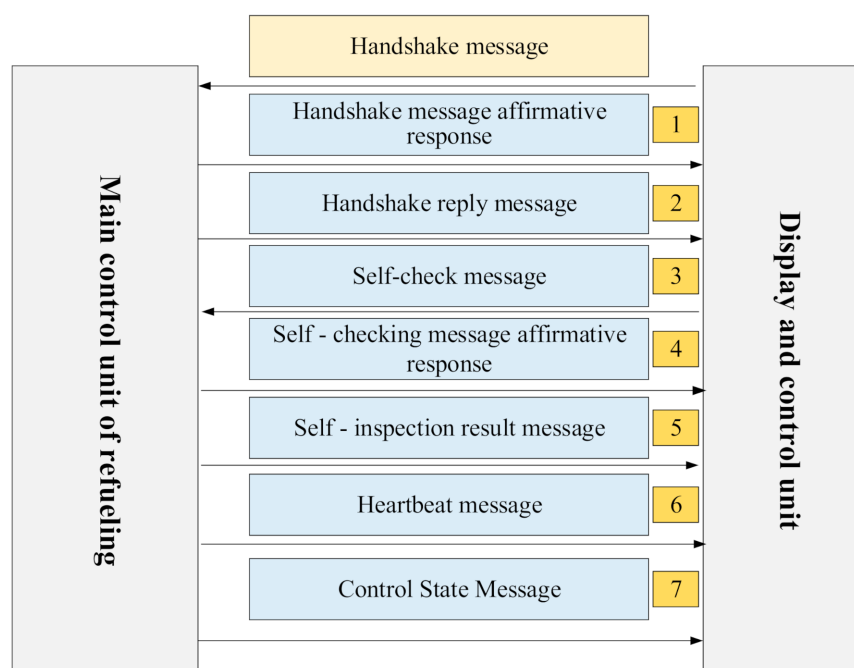


Figure 11. The communication flow between MCU and DCU.

The following part describes the communication between the MCU and the refueling DCU. The MCU adopting the 485 grab packet is used to send the communication data to the refueling DCU. The software is displayed on the computer, which is shown in Figure 12.

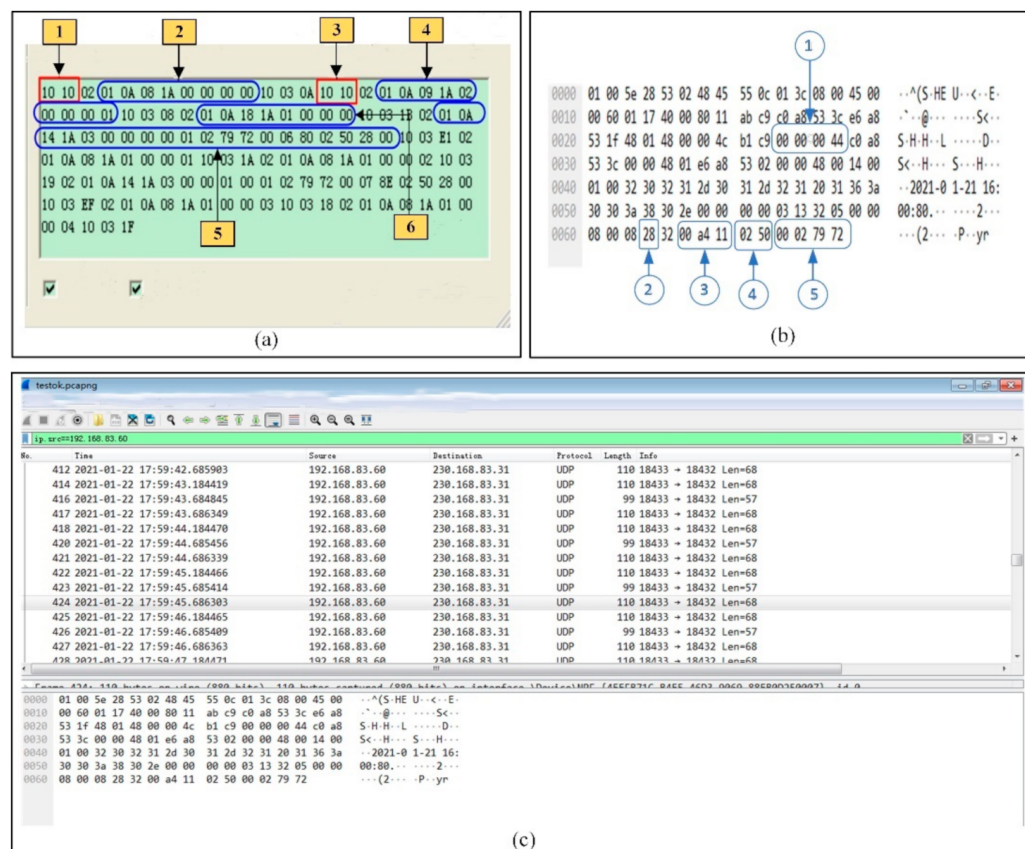


Figure 12. System grab packet: (a) send and reply message for 3964(R) protocol, (b) the specific content of the UDP message, and (c) use the packet capture software to obtain the UDP packet.

In Figure 12a, “1” is a handshake message response sent by the refueling station display control device. The first 10 H is the connection response, and the second 10 H is the handshake message positive response. In the message “6” of Figure 12a, the 10th to 12th bytes represent the real-time flow velocity, with the high bit in the front and the low bit in the back. In the experiment, the input is a 1 KHz pulse (There are 60,000 pulses in one minute; 37 pulses/1 L of oil; the flow velocity is 1621.62 L/min). The flow velocity in the message is 162162 L/min (0×0.27972), 100 times the real-time flow velocity, which is consistent with the experimental data. The 13th and 15th bytes represent the cumulative flow (initial flow velocity) and showing 10 times more fuel than the actual. The 16th and 17th represent the pipeline pressure, which is 592 Pa.

The UDP message sent by the refueling control system to the centralized console is shown in Figure 12b. The control state message is twice per second, and the heartbeat message is once per second. In Figure 12b, “1” represents the length of data; “2” represents the electric valve opening is 40; “3” represents the cumulative fueling volume with 4200.1 L (communication data are expanded by 10 times); “4” depicts the pressure is 592 Pa; “5” shows the flow velocity value is 1621.62 L/min (communication data are expanded by 100 times, which is consistent with the process control amount of the serial port data in Figure 12a).

The system interface is shown in Figure 12c. The experimental setup and the debugging process of the refueling system are shown in Figure 13. The given value is set as 300, 500, and 800 L/min, respectively. Figure 14 gives the results of debugging. Simultaneously, the switch time is an important indicator of the dual-redundant NIC. Our test method is to connect a test computer (centralized console) in this network and exchange messages to each device in the network. Figure 9 shows the test environment. The average switching time of each device is taken. The actual measured channel switching time is within 50 ms. The switching time test record is shown in Table 2. Table 2 shows the results of six groups of experimental tests. “A” represents the main NIC, and “B” represents the backup NIC. The experiment tested the switching time from A to B and from B to A. Ten tests per group and the tests mean time of each group was counted. In engineering applications, the normal switching time is 90 ms. The average value of the experimental tests that we have performed is better than the normal switching time of normal application. The designed system can meet the requirements of engineering applications.

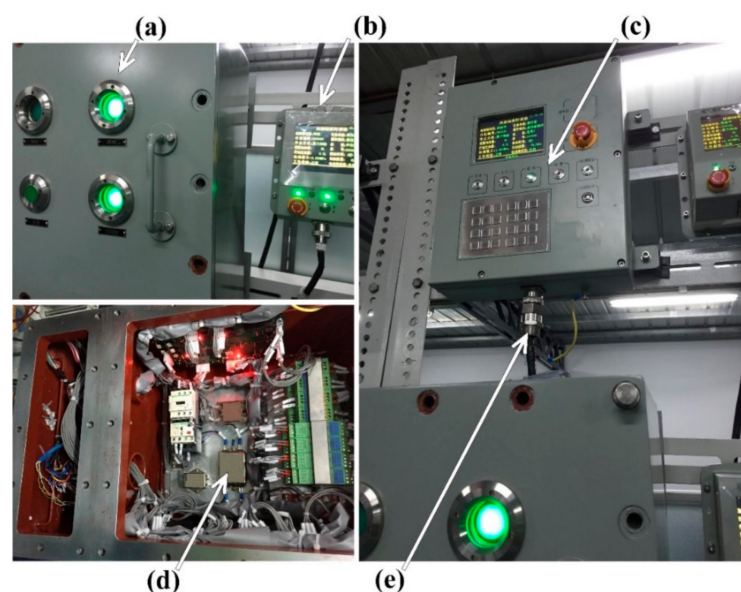


Figure 13. Refueling control system equipment: (a) MCU of the refueling system, (b) DCU of the near-machine, (c) DCU of the refueling system, (d) inside of the MCU, and (e) RS485 communication interface.

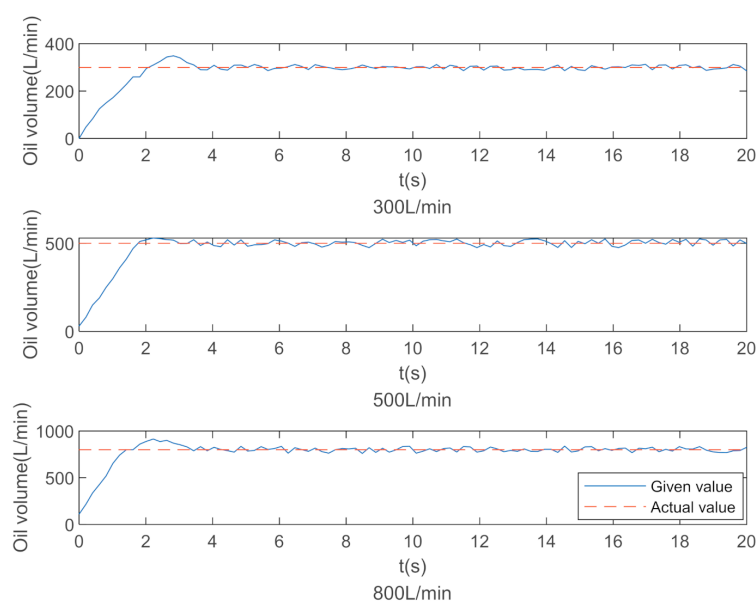


Figure 14. Refueling situation response under different given values.

Table 2. Test record of dual-redundant NIC switch.

Group	Switch Direction	Measured Time (ms)	Variance
1	A to B	32.1	1.89
2	B to A	46.3	1.85
3	A to B	35.2	1.87
4	B to A	48.2	1.88
5	A to B	32.3	1.90
6	B to A	48.2	1.89

6. Conclusions

This study describes the design of a fuel replenishment control system based on a hybrid network. The subject is mainly to design a control system that can meet the refueling requirements of multiple models.

The study was based on the embedded real-time operating system $\mu\text{C}/\text{OS-II}$. ADS 1.2 was used as the development environment for the programming of related modules, and the functions of various modules were realized in a task manner. The control part adopts the PID control law to realize the control of the valve opening.

The hybrid network mainly refers to the RS-485 bus communication network and Ethernet communication network. RS-485 bus communication is used between the control panel and the control system while Ethernet communication is used between the centralized console and the control system. The two communication networks are mainly used to realize the function of the control system.

The final result of this study is that each functional module of the refueling system works normally; the control system can refuel according to the actual flow after receiving the control message sent by the control end. From Figure 14, in different given values, the refueling control system can control the refueling volume to fluctuate around a given value. Table 2 shows that the switching time of this system is fast, which can realize the system stabilization.

Author Contributions: Conceptualization, B.Z.; Methodology, B.Z.; Software, Z.Z.; Validation, Z.Z.; Formal analysis, J.X.; Writing-Review, X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The study was conducted according to the guidelines of the Declaration of Helsinki, and approved by the Institutional Review Board of Harbin engineering university.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data is contained within the article or supplementary material.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dong, K.; Quan, Q.; Wonham, W.M. Failsafe Mechanism Design for Autonomous Aerial Refueling using State Tree Structures. *Unmanned Syst.* **2019**, *7*, 261–279. [[CrossRef](#)]
2. Sun, S.; Yin, Y.; Wang, X.; Xu, D. Robust Landmark Detection and Position Measurement Based on Monocular Vision for Autonomous Aerial Refueling of UAVs. *IEEE Trans. Cybern.* **2019**, *49*, 4167–4179. [[CrossRef](#)] [[PubMed](#)]
3. Duan, H.; Li, H.; Luo, Q.; Zhang, C.; Li, C.; Li, P.; Deng, Y. A binocular vision-based UAVs autonomous aerial refueling platform. *Sci. China Inf. Sci.* **2016**, *59*, 1–7. [[CrossRef](#)]
4. Xu, Y.; Duan, H.; Li, C.; Deng, Y. On-board visual navigation system for unmanned aerial vehicles autonomous aerial refueling. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2017**, *233*, 1193–1203. [[CrossRef](#)]
5. Duan, H.; Zhang, Q. Visual Measurement in Simulation Environment for Vision-Based UAV Autonomous Aerial Refueling. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 2468–2480. [[CrossRef](#)]
6. Brisse, A.; Zeller, M.; Ludwig, B.; Brabandt, J. Solid Oxide Electrolyzer System Operational at the H2 Refueling Station of Karlsruhe. *Fuel Cells* **2019**, *19*. [[CrossRef](#)]
7. Riedl, S.M. Development of a Hydrogen Refueling Station Design Tool. *Int. J. Hydrogen Energy* **2020**, *45*, 1–9. [[CrossRef](#)]
8. Xu, X.; Hu, W.; Cao, D.; Huang, Q.; Liu, W.; Jacobson, M.Z.; Chen, Z. Optimal operational strategy for an offgrid hybrid hydrogen/electricity refueling station powered by solar photovoltaics. *J. Power Sources* **2020**, *451*, 227810. [[CrossRef](#)]
9. Williamson, W.R.; Glenn, G.J.; Dang, V.T.; Speyer, J.L.; Stecko, S.M.; Takacs, J.M. Sensor Fusion Applied to Autonomous Aerial Refueling. *J. Guid. Contr. Dynam.* **2015**, *32*, 262–275. [[CrossRef](#)]
10. Chen, C.-I.; Koseluk, R.; Buchanan, C.; Duernner, A.; Jeppesen, B.; Laux, H. Autonomous Aerial Refueling Ground Test Demonstration—A Sensor-in-the-Loop, Non-Tracking Method. *Sensors* **2015**, *15*, 10948–10972. [[CrossRef](#)] [[PubMed](#)]
11. Cai, X.; Yuan, D.; Yan, J.; Qu, Y. Hardware-in-loop Simulation System for the Disturbed Movement of Hose-drogue During Aerial Refueling. In Proceedings of the 2018 IEEE CSAA Guidance, Navigation and Control Conference (GNCC), Xiamen, China, 10–12 August 2018.
12. Plaza, E.; Santos, M. Knowledge based approach to ground refuelling optimization of commercial airplanes. *Expert Syst.* **2021**, *38*. [[CrossRef](#)]