Journal of Computer Science and Cybernetics, V.39, N.2 (2023), 191–206
DOI no 10.15625/1813-9663/18172

AN IMPROVEMENT DESIGN OF MULTI-FUNCTION CONTROLLER FOR HIGH-TECH SHRIMP FARM

CAO THU BUI

Industrial University of Ho Chi Minh City, 12 Nguyen Van Bao Street, Ward 4, Go Vap District, Ho Chi Minh City, Viet Nam

Abstract. Shrimp farming is one of the highest potential areas in the coastal provinces of Vietnam. The technical level of shrimp farming is developing strongly in the direction of high technology based on applying modern equipment and machinery. However, the efficiency of shrimp farming is still not high, the monitoring and control of the actuators are mainly manual. Shrimp farmers are not interested in using smart controllers for shrimp farms. In this article, the author evaluates the limitations of the current controllers and proposes a multi-function controller which is suitable for the practical requirements of high-tech shrimp farms (HTSFs) with significant functions such as soft configuration, overload protection, and engine damage warning for actuator devices. In addition, the proposed controller also allows monitoring and automatically controlling HTSFs on a mobile application.

Keywords. High-tech shrimp farm, IoT gateway, smart controller.

1. INTRODUCTION

Viet Nam has strengths in aquaculture with a coastline of over 2000 km. Shrimp farming is one of the highest potential areas in the maritime provinces. The technical level of shrimp farming is developing strongly in the direction of high technology based on applying modern equipment and machinery. Shrimp farming techniques have two main forms: extensive farming with low density and super intensive farming with high stocking density. In super-intensive farming with high density, there is a great demand for devices in the environmental treatment of pond water.

1.1. Introduction of an HTSF

The equipment for an HTSF is seen in Figure 1. The block diagram for actuator devices is arranged in Figure 2, including the actuators: water supply pump, aerator pump, aerator fans, waste siphon pump, and shrimp feeder. The water fans make the water circulate in the shrimp pond. The aerator makes oxygen dissolve in water for shrimp to breathe. The supply pump drives water from the external settling pond to the shrimp farm for exchanging periodically. The siphon pump sucks the waste of shrimp and over food. The actuators used for shrimp farms usually have a large capacity from a few to ten HP (horse power). The

Corresponding author.
E-mail addresses: buithucaoc@iuh.edu.vn (C.T. Bui)

Figure 1: Illustration of high-tech shrimp pond equipment system

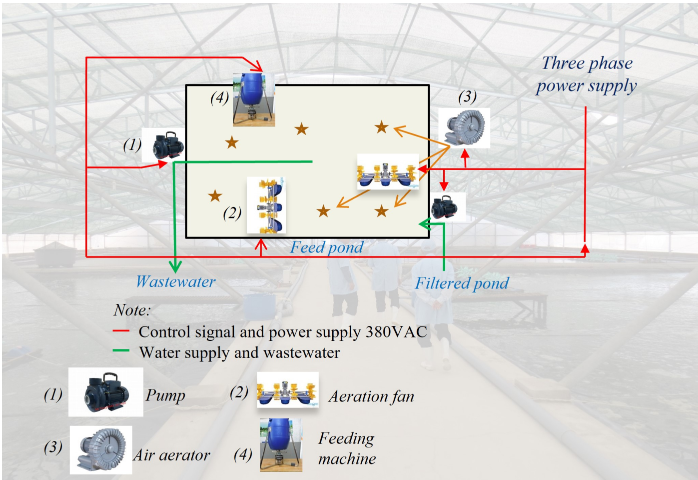


Figure 2: Block diagram of equipment layout of high-tech shrimp pond

monitoring and control of these actuators are mainly done manually. The use of automation controllers for actuators for shrimp ponds are very rare in HTSFs today.

1.2. Evaluation of current controllers

There are two types of HTSFs:

- Large farm with many ponds. This farm uses professional automation devices such as DCS (distributed control system) and PLC (programmable logic controller) to operate actuators as presented in [1, 2]. This farm has a large cost and requires automation technicians for operating the systems.
- Small farm with few ponds. In this farm, an integrated controller is used to control actuators as described in [3–8]. The controller uses electromagnetic relays and an IoT gateway to connect to the cloud for supervising and controlling the system through a mobile application. This farm model has a low cost and simple operation without the requirement of automation technicians.

(a) HYDRONEO, Thailand (b) SENECT, Germany

Figure 3: The shrimp farm foreign controllers

There are some foreign suppliers for integrated controllers in the international market, such as the Smart Shrimp Farm Management System of HYDRONEO [9] in Figure 3.a) and the Aquaculture Controller of SENECT [10] in Figure 3.b). In Vietnam, there is a typical controller FARMEXT of TEPBAC company [11] as shown in Figure 4. All these controllers have the same control functions such as supervising the environment parameters and controlling on/off actuators via mobile application.

(a) FARMEXT controller (b) Inside cabinet

Figure 4: The shrimp farm controller of FARMEXT

There are some disadvantages of the present integrated controllers:

1) The present controllers are designed to fix the output power for each actuator. If we assemble an actuator device with a capacity greater than the designed capacity, the actuator device will not work. If we assemble an actuator with a capacity lower than

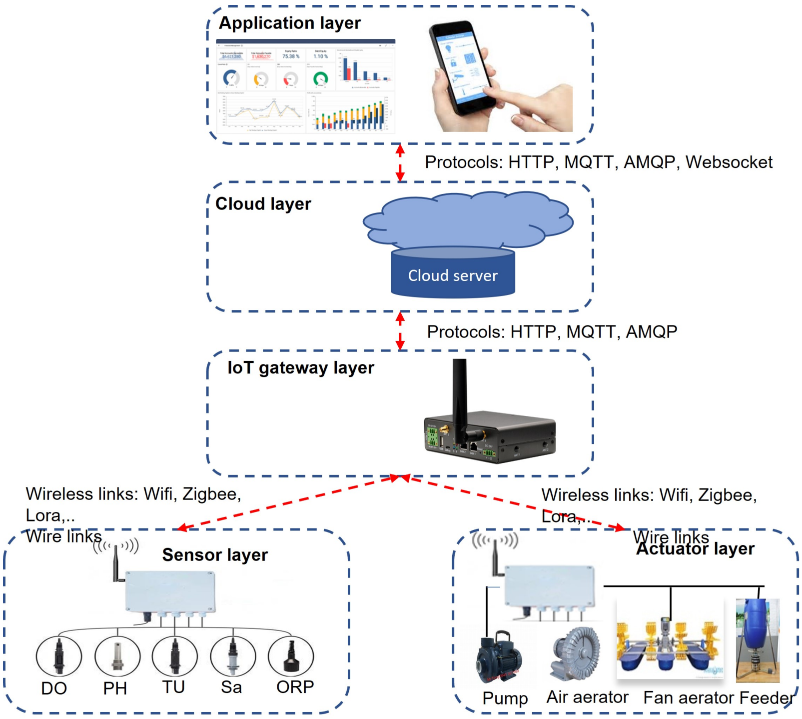


Figure 5: The general structure of the monitor and control system of HTSF

the designed capacity, the controller will not be able to protect against overload. This is a difficulty for shrimp farmers in practice, because in practical, shrimp farms use many types of motors with different power. For example, the long-ago aerator was used with a capacity of 5HP, but with new technology, the current aerators only require a capacity of 2HP to provide enough oxygen for the shrimp pond.

2) Due to operating in sea water, the large power connections and coils inside the actuator devices will corrode over time. Therefore, the impedance of the connections will increase, and this makes the current through the motor decrease. The result is a weaker engine over time.

3) Most current controllers use contactors to switch power for actuator devices. This will cause some disadvantages. The contactors can create ignition and contact corrosion when switching large power. Thus, the device lifetime will decrease, and ignition noise will create errors for the microcontroller.

To overcome the above disadvantages, a design improvement of the multifunction controller for HTSF is proposed.

2. DESIGN OF THE MULTIFUNCTION CONTROLLER

2.1. Idea of designing the multifunction controller (MFC)

As the result of synthesizing from the previous studies, the layer structure of a controller for HTSFs is described in Figure 5. The system has 5 layers. The sensor layer includes a data acquisition module and sensors which measure the parameters of the water environment such

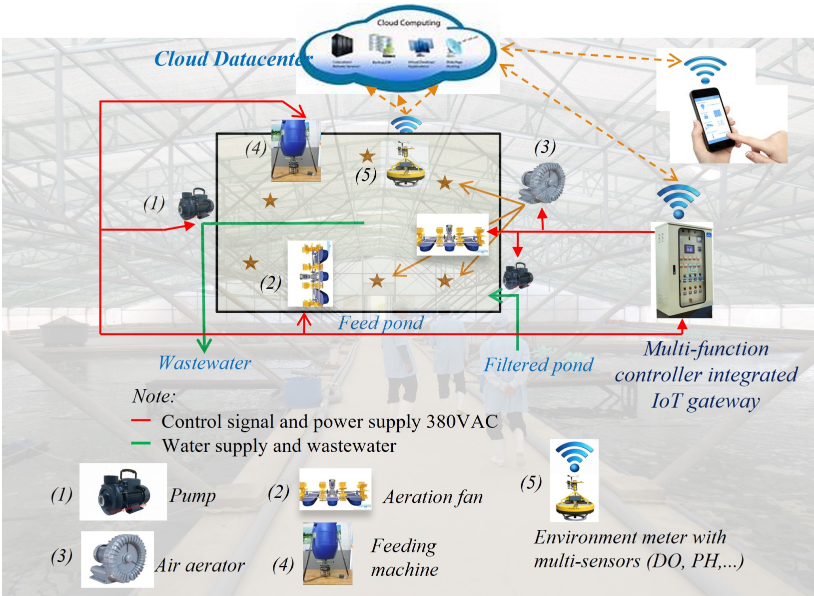


Figure 6: The proposed model for HTSF equipment and systems

as dissolved oxygen, salinity, pH, and turbidity. These water environmental parameters are sent to the IoT gateway layer and then sent to a database in the cloud. The proposed model for HTSF equipment and systems is shown in Figure 6. Data in the cloud is used to supervise and control the mobile application. Control commands from the mobile application will be sent to the cloud to update the changing status, and from the cloud to the IoT gateway and then to the MFC to control the operation of devices such as water supply pumps, aerators, water fans and shrimp feeding machines. Based on the general structure as shown in Figure 5 and the disadvantages of currently integrated controllers, we propose the design idea for the MFC with some features as follows:

a) Replace the contactor with a solid-state relay (SSR). Controllers using SSRs with phase-current sensors will allow users to reconfigure (or soft-configure) voltage and power parameters without changing the device's control hardware.

b) The MFC using SSR will not cause ignitions when switching contact points. It will enlarge the lifetime and quality of the controller.

c) The phase current sensors at the power output of the MFC allow supervising and timely detection of equipment failures.

d) The MFC is designed for a HTSF model in 500m2 as seen in Figure 8. The actuators include one 3-phase supply pump, two 3-phase water fans, one 3-phase aeration, one 1-phase waste siphon pump, and one 1-phase automatic shrimp feeder.

e) The MFC is integrated with Lora or Zigbee interfaces to enable the expansion of the control link with other controllers.

To backup DC power for the MFC, a solar panel battery is utilized. When the main supply power is lost, the controller still works and sends an alarm to the user.

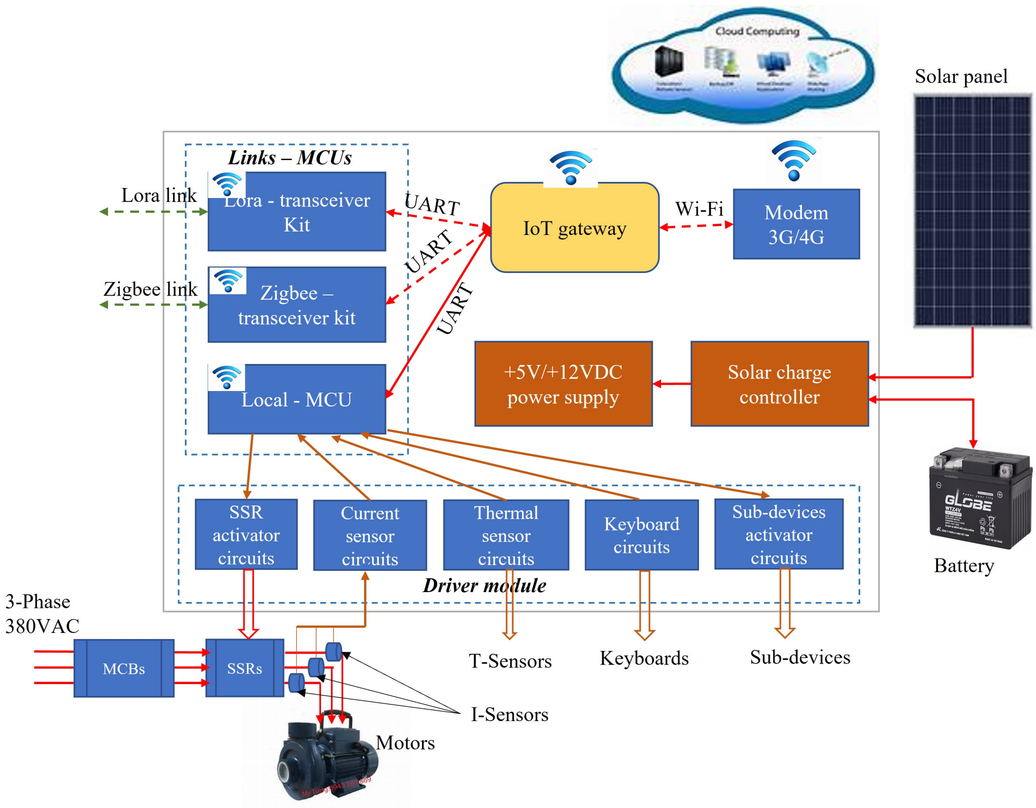


Figure 7: Structure design of the MFC

2.2. Structure design of the MFC

As shown in Figure 7, the structure design includes the following modules:

- Link-MCUs module: It encompasses Local – MCU, Lora – Transceiver kit, and Zigbee transceiver kit as shown in Appendix 1. The Local – MCU is the main micro-controller for the MFC. The Lora – Transceiver kit and Zigbee transceiver kit are used for linking to other MFC. Arduino mega 2560 is used for the Local-MCU because the MFC uses 21 DI/O which are 9 digital pins for 9 buttons, 4 digital pins for temperature sensors, and 8 digital pins for sub-devices. The MFC also uses 12 analog input pins for the phase current sensors.
- IoT gateway module: This module uses NODEMCU ESP8266 for the IoT gateway.
- Modem 3G/4G: This Wifi modem uses Alcatel MW40 4G LTE at a low cost.
- Driver module: This module is designed as in Figure 7, with the following components:

  - The activator circuits are designed by using transistors to switch ON/OFF SSRs. The SSRs are controlled by +5VDC voltage.
  - The current sensor circuits are RC filter circuits that adapt the output signal from the phase current sensors at the SSR output to the analog input of the Local - MCU.

– The thermal sensor circuits include signal acquisition circuits from temperature sensors on the SSR heat sinks.

– The keyboard circuits are state sensing circuits of buttons on the keyboard.

– The Sub-device activator circuits are buffer circuits for digital control signals from the Local – MCU to another Sub-diver.

Table 1: Statistics of current consumption

No.	Name of modules	Current (mA)
1	Lora-transceiver kit	20
2	Zigbee-transceiver kit	30
3	Arduino mega 2560	27
4	NODEMCU ESP 8266	126
5	Modem Wifi 4G Alcatel MW40	200
6	4 HSR-3D404Z	56
7	4 DS18B20	6
8	4 Sensor YHDC STC013	200
8	Relay for sub-devices	100
	Total current	765

- SSR modules: We use 3-phase solid state relays which are controlled by DC voltage to switch power for actuators. The actuator capacity of HTSFs is in the range of 2HP to 10HP and the starting current of actuator motors is viewed in [12], we can calculate the rated current of SSRs as follows

$$I = \frac{10P}{3U} = \frac{10 \cdot 10 \cdot 750}{3 \cdot 380} = 70A.$$

  Therefore HSR-3D704Zs are selected with its specification is shown in [13].

- I-Sensor modules: 12 current sensors for four 3-phase SSRs are used for protecting against overload and detecting the consumption power of each actuator motor. We choose a 100A AC current sensor, as shown in Appendix 1.

- T-Sensor modules: 4 temperature sensors are used for supervising, warning, and protecting against overheating for SSRs. DS18B20 temperature sensors are selected as shown in Appendix 1.

- Sub-device module: We use an 8-channel optocoupler relay to switch 1-phase 220VAC power for 1 waste siphon pump, 4 heat sink fans for the SSRs, and a shrimp feeding machine 0.25HP.

- MCBs (Miniature Circuit Breaker): We use MCB to assist the assembly process as well as create the safety system. We select an MCB type B 16A because the maximum starting current of the SSRs is 70A. It is MCB LS BKN 16A.

- Power supply module: The power module is a +5V DC voltage regular. Based on the list of maximum power consumption for the modules in the MFC as shown in Table 1, we choose the XL4015 power module as shown in Appendix 1. This power module provides enough power for the controller.

- Solar charging module: Solar cells are used to allow the controller to be connected online to the cloud for 24/7 supervising and control of the actuators. The parameter values of the selected solar charging module are 130W and 10Ah to be able to charge a 7A ACCU battery. This battery can remain the mains power for more than 7 hours in the event of power line failure.

2.3. Design of database, the control algorithm for IoT gateway and local - MCU

The control program is designed with functions as follows:

1. Permit the user to soft-configure the rated voltage and power of devices on the mobile application.
2. Read and process the changing state of the keyboard with 9 control buttons, 4 buttons for 3-phase equipment, 4 buttons for sub-devices, and 1 button to set AUTO or MANUAL mode. AUTO mode allows controlling automatically all devices. MANUAL mode allows users to control all devices directly through the keyboard.
3. Get the updated data from the cloud and control the devices such as aeration systems, water supply pumps, water fans, waste siphon pumps, and automatic shrimp feeders.
4. Read sensors and process overheating and overload:
   (a) Handle the overheating protection. The temperature range of SSRs is described in Appendix 2. When overheated, the temperature of SSR is greater than 60 Celsius degrees, the MFC will release an alarm buzzer and update the overheating status, ErrorDev = 1, on the cloud and the mobile application.
   (b) Handle the overload protection. In starting time, when the phase current is more than 10 times the rated current of the configured device, the controller will switch off the power of the device, release an alarm buzzer and update the overload state, ErrorDev = 2, on the cloud and the mobile application.
   (c) Handle phase out of sync. Phase loss will cause an increasing surge in current for the coils relative to the remaining phases in the motors. This leads to overload and burns all motors of the shrimp farm. To solve the problem, the MFC will check if any phase current in the 3 phases is zero while the remaining phases are not. If so, the MFC will switch off the power of all devices, alert with an alarm buzzer and update the phase-out of sync status, ErrorDev = 3.
5. Send data received from Lora/Zigbee modules to the cloud database to extend the control connection.

2.4. Design of mobile control application

Blynk.io is a very powerful IoT platform for iOS or Android mobile devices. We selected Blynk for storing the database and developing mobile control application for the MFC on the cloud. The graphical user interface (GUI) for a mobile application is shown in Figure 10 with the functions as follows:

- OPEN SYS button is used to switch ON/OFF the whole system.
- Wifi icon displays the Wi-Fi status. If it is white color, the Wi-Fi connection is a failure and if it is yellow color, the Wi-Fi is connected.

Table 2: Declaring system variables on the cloud database

1)	IS[4][3] is three-phase current sensing value variable for 4 motors.
2)	Dev[4] is operation state variable of four main motor devices.
3)	Power, Voltage, and Phase are the variables of the rated power, voltage, and phase number of the motors.
4)	Temp[4] is the temperature value variable of SSRs.
5)	DevState is an actual operating state variable of the devices (conventionally 1 is ON and 0 is OFF)
6)	ModeState is an operation mode variable (0 is AUTO and 1 is MANUAL)
7)	ErrorDev is a device's fault status variable (0 is no error, 1 overheating warning level, 2 overload warning levels, 3 phase out of sync)
8)	Sub_Dev[8] is operation state variable of 4 cooling fans for SSR and 4 motors for other auxiliary devices such as shrimp feeders, and siphon suction, ...
9)	OPEN_SYS is a control system variable, ON/OFF the whole MFC system.

Table 3: IoT gateway control algorithm

1)	Declare libraries and parameters for IoT gateway.
2)	Read the updated data from the cloud and transmit it to the corresponding UART links.
3)	Read command data from UART links and update variables to a cloud database.
4)	Repeat step 2).

Table 4: The main control algorithm of the local-MCU module

1)	Declare the library and system variables for the Local-MCU.
2)	Read command data from the UART link, and update the operating status and motor device parameters.
3)	Scan states and handle keyboard (Subroutine below).
4)	Read sensor values and handle overheating and overload (Subroutine below).
5)	Repeat step 2).

- Alarm icon shows an alarm state. In white color, there is no alarm. If it is yellow, the device has problems or failure.
- Voltage, Power, and Phase are input texts used for setting the device parameters.
- Dev1, Dev2, Dev3, and Dev4 are control buttons to switch on/off power for the three-phase actuator devices.
- I_Dev1, I_Dev2, I_Dev3, and I_Dev4 are the phase current indicators of the devices.
- When we switch on/off the buttons, the result of the command execution is displayed in a two-line text box. The above line presents the progress of the command execution, and the below line presents the existent status of the switched devices. The existent state of the devices is represented by an array of five characters. The first four weights are in 1 (ON) or 0 (OFF) status for the respective devices, and the last weight is in

Table 5: Subroutine for scanning states and handling the keyboard

1)	Read the state of the pressed keys, update on MODE and key[8].
2)	If key state (MODE! = ModeState) {update ModeState}
3)	For (k=1; k¡=4; k++) {
	If (key[k]! = Dev[k]) and (OPEN_SYS=1) {
	If key[k] = 0 {deactivate SSR[k], stop cooling fan [k] and update motor status Dev[k] = 0, Sub_Dev[k] = 0}.
	If key[k] = 1 {enable SSR[k], turn on cooling fan [k] and update motor status Dev[k] = 1, Sub_Dev[k] = 1}.
	Update the status of DevSate devices.
	}
	}
4)	For (k=5; k¡=8; k++) {
	If (key[k]! = Sub_Dev[k]) and (OPEN_SYS=1) {
	If key[k] = 0 {deactivate Relay[k] and update motor state Sub_Dev[k] = 0}.
	If key[k] = 1 {enable Relay[k] and update motor state Sub_Dev[k] = 0}.
	Update the status of Sub_DevSate devices.
	}
	}

Table 6: Subroutine for reading sensor values, handling overheating and overload

1)	Read the temperature sensors of the SSRs and store them at temp[4].
2)	Read the phase current sensor of the SSRs and store it at IS[4][3].
3)	Calculate device nominal current: IF = Device Power/(3×Voltage).
4)	For (k=1; k¡=4; k++) {
	Hint: j is phase index and in the range [1...3]
	//Overheat warning
	If (Temp[k] > 60) {ErrorDev[k] = 1; Deactivate the SSR[k], turn off the cooling fan [k], update the motor status Dev[k] = 0, Sub_Dev[k] = 0 and trigger the overheat alarm}
	//Overload warning
	If (IS[k][j] > 10IF) {ErrorDev[k] = 2; Deactivate the SSR[k], turn off the cooling fan [k], and update the motor status Dev[k] = 0, Sub_Dev[k] = 0 and trigger the overload alarm}
	//Phase sync loss warning on the supply
	If ([(min (IS[k][j]) = 0] and [max (IS[k][j]) > 0]) {ErrorDev[k] = 3; Deactivate the SSR[k], turn off the cooling fan [k] and update the motor status Dev[k] = 0, Sub_Dev[k] = 0 and trigger the phase sync loss alarm}
	}

AUTO (A) or MANUAL (M) mode. Example: In Figure 8, on the left, the text box indicates that only Dev1 is enabled, and the system is operating in MANUAL mode.

- Temp1, Temp2, Temp3, and Temp4 show the temperature of the SSR heat sinks.
- pH, DO, TU, and Sa display the pH, dissolved oxygen, turbidity, and salinity values of the water environment of the shrimp pond.
- The function of setting the "Schedule". This function allows the system to automatically turn on or off the actuators according to a preset schedule, automatically turn

on and off the aerator when the dissolved oxygen is below the allowable threshold, or automatically feed the shrimp at a preset time.

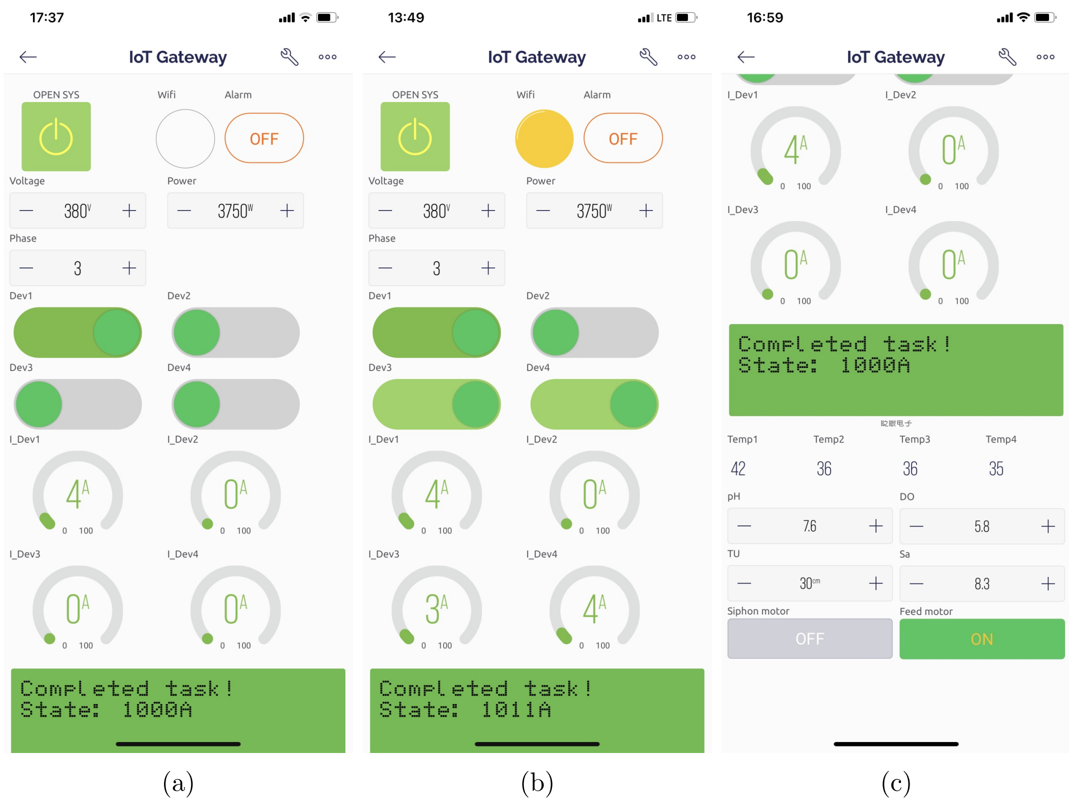
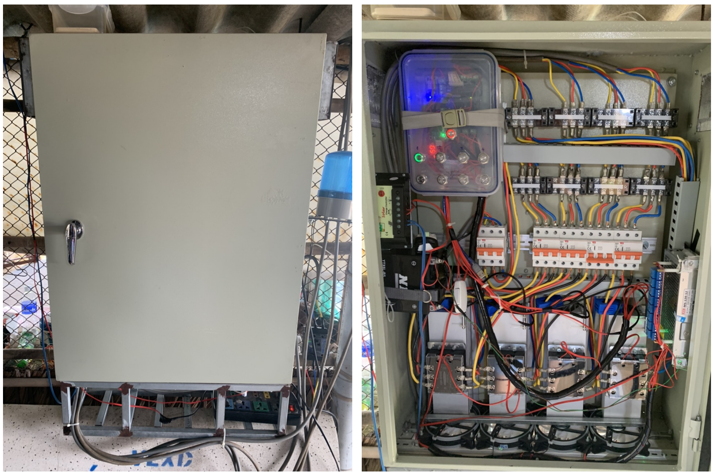


Figure 8: GUI of mobile application for the MFC. On the left, a) is a GUI of MANUAL mode. In the middle, b) and the right c) are GUI of AUTO mode.

2.5. Installation of the control cabinet

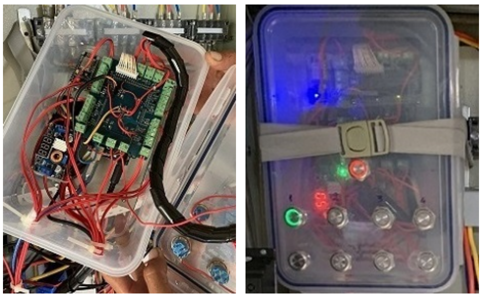
In Figure 9, they include the central control box, Wi-Fi module, the terminal blocks connecting the actuators to the SSRs output below, the terminal blocks connecting the mains input power to the MCBs below, the outputs of the MCBs connected to the SSRs below, and 4 cooling fans in the bottom. Besides, the left-side middle attaches the solar charger and the battery 7A. The right-side middle attaches the sub-devices module. In addition, the control cabinet is arranged with 2 vents on the upper 2 sides and 4 vents at the bottom. The vents have the function of cooling the SSRs.

Figure 10 a) presents the assembled modules in the central control block. They include a driver module superimposed on the Arduino Atmega 2560 module, an IoT gateway module, and power module. Figure 10 b) presents the keyboard with 9 buttons: the top button is used to set the AUTO/MANUAL mode, the four upper buttons are used to control the large power actuators, and the lower 4 buttons are used to control the sub-devices.



(a) (b)

Figure 9: Image of the control cabinet, image a) is an outside image, and image b) shows inside components.



(a) (b)

Figure 10: Image of the central control block, image a) shows the installation of component modules, and image b) shows the front panel of the central control box with the keyboard.

2.6. Results

The MFC was tested at a shrimp farm in Binh Dai district, Ben Tre province, as seen in Figure 11. The shrimp pond has 4 aeration fans 5HP (3.75KW), 1 shrimp feeding machine, and 1 siphon machine. Through the testing process since April 2022, the shrimp farm owner assessed that the MFC has been working well, installing and configuring the devices were simple with many convenient functions such as allowing surveillance, warning damage, programming automatic control in diverse and at ease. The testing process was performed with three 5HP devices and a 7HP device for 30 experimental times. We used a multimeter Uni-T UT201+ for measuring phase current. It is shown in the statistics in Table 7. The average operating current ranges from 3.5A to 3.38A for 5HP devices and 5.2A for 7HP devices. These values are reasonable because all devices have been rusted over time. So, the operating current of devices is different. The tolerance of current measurement between the real current measured by the multimeter Uni-T UT201+ with the MFC is about ±0.5A. Because the displayed value is rounded one unit for convenience of observation, the tolerance of the current measurement is reasonable.

The function of overheating protection for the SSR worked well. The temperature is

Figure 11: Field test image of the MFC at a shrimp pond in Binh Dai district, Ben Tre province, Viet Nam.

displayed on the smartphone application and timely alerts when there is an SSR overheating problem. The test is performed by burning the temperature sensor probe mounted on the SSR's heat sink. When the temperature exceeds 60 Celsius degrees, the system has an alarm with overheating status. This function is important because of working in the field, under sunlight and large power equipment, the SSR will be very hot. If it is not cooled in time, the SSR will die.

Table 7: Statistic of the phase current sensors

Device	Power	Reated current	Average operating current	Tolerance of measurement
Dev1	5HP	3.3A	3.8A	±0.5A
Dev2	5HP	3.3A	3.7A	±0.5A
Dev3	5HP	3.3A	3.5A	±0.5A
Dev4	7HP	4.6A	5.2A	±0.5A

The overload protection function worked well in the tests. The overload protection test is performed by entering the device's rating as 3HP, but connecting the 7HP device to the power output. When starting, the MFC immediately alarms the overload and stops the motor. Besides, during operation time, there are several times when the aeration fans get stuck. This makes the motor unable to rotate and suddenly increases the load current. In this situation, the MFC also has an overload alarm and automatically disconnects power.

Testing phase synchronization loss was performed by disconnecting one in three phase wires of the operating device, the mobile application will display equipment failure, and the MFC releases buzzer alarms. On the day or during the night, when a shrimp farmer sleeps, he cannot detect when the water fans or the aerators are damaged or stopped. Loss of phase synchronization will make the motor run slowly down and increase the current of the remaining phases in the motor. This will lead to motor damage if it is not detected in time. Furthermore, stopping aeration for only an hour in the shrimp pond with high stocking density can make shrimps die from a lack of oxygen. In this case, the MFC is useful to help him to detect and promptly warn whether the motors are overloaded, phase synchronization loss or not working. This function is especially important and very useful.

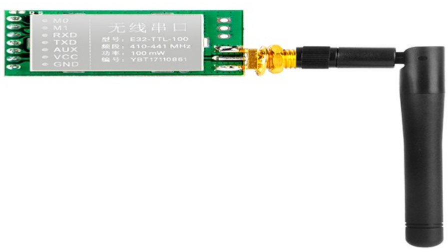
In addition, during installation and commissioning, the MFC controller helped to detect a problem with a 5HP motor because it has a current consumption of over 5A, instead of the rated current value of this device of 3.3A. (When there is a load of the 18-blade fan, the actual device current can be from 3.3 to 4A). After replacing the good motor, this motor works the same function as the previous motor, but the steady current consumption is equal to the other motors, below 4A. The detection of this equipment failure has saved 1/5 of the power consumption cost of this motor.

3. CONCLUSIONS

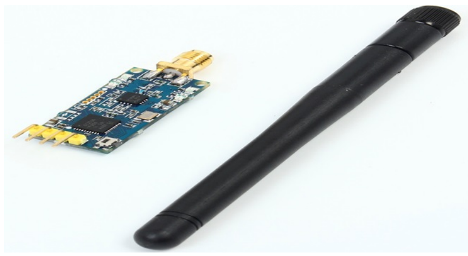
MFC test results and actual operation show that the control features have been completed and meet the actual requirements for HTSF. The controller has significant improvements to compare with existing controllers as follows:

- Using SSR instead of a contactor makes the control process safer, and more stable and increases the durability of the actuator devices.
- Using phase current sensors for each actuator motor helps users to detect damage in time and avoid loss of power during the operation.
- Allowing users to reconfigure device parameters according to actual actuators at HTSF makes MFC more flexible, and easy to install and operate without automation technicians.

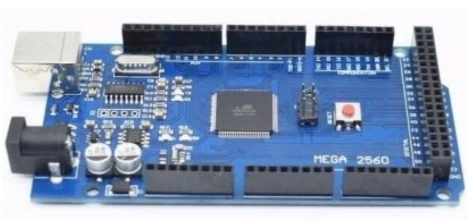
The MFC is designed to control devices for a shrimp pond. In fact, if the farm has a system of many shrimp ponds with diverse functions such as nursery ponds, culture ponds, sedimentation ponds, filter ponds, and wastewater treatment ponds. The big challenge is that the MFC needs to be designed to expand with the ability to link and control synchronously between the MFC controllers of each pond. At that time, farmers only needed to install the farming process on the smartphone application to be able to control the entire system of ponds without having to install each pond.



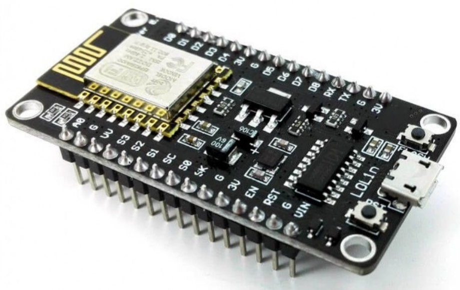
(a) UART Lora SX1278



(b) UART Zigbee CC2530+PA V2



(c) Arduino mega 2560 module



(d) NODEMCU ESP 8266 module

Figure 12: Images of the RF modules, Local-MCU and IoT gateway

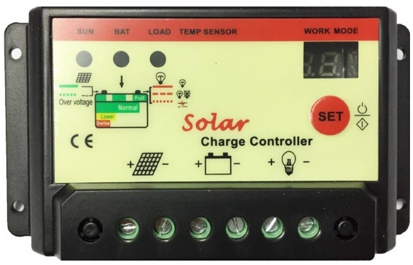
(a) Modem Wifi 4G, Alcatel MW40

(b) Solid state relay HSR-3D704Z

(c) T-Sensor module, DS18B20

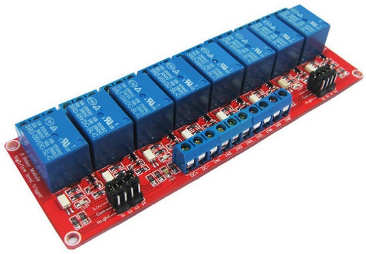
(d) I- Sensor module, YHDC STC013

(e) Power supply module



(f) Solar charging module

(g) Heat sink fan for SSRs



(h) Relay module for sub-devices

Figure 13: Pictures of the remaining related modules installed

4. APPENDIX

List of components assembled in the MFC.

REFERENCES

[1] J.-H. Huh, "PLC-based design of monitoring system for ICT-integrated vertical fish farm," *Human-Centric Computing and Information Sciences*, vol. 7, no. 1, p. 20, Jul 2017.

[2] N. P. Mahalik and K. Kim, "Aquaculture monitoring and control systems for seaweed and fish farming," *World Journal of Agricultural Research*, vol. 2, no. 4, pp. 176–182, 2014.

[3] D. S. Simbeye, J. Zhao, and S. Yang, "Design and deployment of wireless sensor networks for aquaculture monitoring and control based on virtual instruments," *Computers and Electronics in Agriculture*, vol. 102, pp. 31–42, 2014.

[4] P. S. Sneha and V. S. Rakesh, "Automatic monitoring and control of shrimp aquaculture and paddy field based on embedded system and IoT," in *2017 International Conference on Inventive Computing and Informatics (ICICI)*, 2017, pp. 1085–1089.

[5] R. Rahayani and A. Gunawan, "Proposed design of an automatic feeder and aerator systems for shrimps farming," *International Journal of Materials, Mechanics and Manufacturing*, vol. 6, pp. 277–280, 08 2018.

[6] T. H. S. N. T. K. Duy, N. D. Tu and L. H. D. Khanh, "Automated monitoring and control system for shrimp farms based on embedded system and wireless sensor network," in *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 2015, pp. 1–5.

[7] T. Khaoula, R. A. Abdelouahid, I. Ezzahoui, and A. Marzak, "Architecture design of monitoring and controlling of iot-based aquaponics system powered by solar energy," *Procedia Computer Science*, vol. 191, pp. 493–498, 2021.

[8] J. Capelo, E. Ruiz, V. Asanza, T. Toscano-Quiroga, N. N. Sánchez-Pozo, L. L. Lorente-Leyva, and D. H. Peluffo-Ordóñez, "Raspberry Pi-based IoT for shrimp farms real-time remote monitoring with automated system," in *2021 International Conference on Applied Electronics (AE)*, 2021, pp. 1–4.

[9] HydroNeo, "Smart shrimp farm management system," Tech. Rep., 2022. [Online]. Available: https://hydroneo.net/

[10] SENECT, "Senect aquaculture technology," Tech. Rep., 2022. [Online]. Available: https://www.senect.de/en/

[11] TEPBAC, "Cabinext for shrimp farm," Tech. Rep., 2022. [Online]. Available: https://tepbac.com

[12] V. K. Mehta and R. Mehta, *Principles of Electrical Machines.* S. Chand Publishing, December, 2006.

[13] HANYOUNGNUX, "Specification for hsr-3d," Tech. Rep., 2022. [Online]. Available: https://www.hanyoungnux.com/

Received on March 16, 2023
Accepted on May 14, 2023