The Optimal Deploy Method of Multi Redundancy FPGA Gateway Design

Changjian Deng, Yaozhong Zhang

***Abstract:** Prevention of geological disasters is essential in high-altitude or semi-high mountain plateaus and hilly areas. To monitor disasters, the manuscript presents a design method for a distributed redundancy FPGA (Field Programmable Gate Array) gateway. This gateway can test a series of parameters to estimate the likelihood of geological disasters. However, detecting geological disasters is a complex task, as the test fields are located at different monitoring sites, and each site has unique environmental conditions. Meanwhile, the same type of monitoring parameters should be monitored in more than two places, and the location may change, resulting in the problem of distributed redundant monitoring. There are three types of monitoring actors: active detection, auxiliary detection, and redundancy. And these actors can transform one into another. To achieve a balance between communication resources and the effective detection of geological disasters, the design of a distributed redundancy gateway should be dynamically deployed based on changes in environmental and detection conditions. The article employs a redundant gateway design utilising multiple FPGAs in hardware and implements a distributed, redundant decision-making method in software. It uses a wireless redundancy decision-making optimization method, and gives a design example for 'the atomic cloud' application of infrasound monitoring.*

***Keywords:** Redundancy; Optimization; Distributed Gateway; Geological Disasters*

Abbreviations:
FPGA: Field Programmable Gate Array
EDF: Earliest Deadline First
MAC: Media Access Control
IoT: Internet of Things
MCU: Microcontroller Unit
HART: Highway Addressable Remote Transducer
WIFI: Wireless Fidelity
UART: Universal Asynchronous Receiver/Transmitter
RTL: Register Transfer Language
MQTT: Message Queuing Telemetry Transport
HTTP: Hypertext Transfer Protocol
AMQP: Advanced Message Queuing Protocol
CPU: Central Processing Unit
I/O: Input/Output
LED: Light-emitting diode

I. INTRODUCTION

The network gateway is a node which connects two or more different networks with different transmission protocols and simplifies the communication connectivity into one electronic device [1]. The gateway node acts as a firewall and proxy server for some applications [2], for example, the detection of debris flow. To monitor debris flows, a test plan should be developed based on the collection of data and on-site investigations. In the rural field, it may measure rainfall, infrasound, level of mud, velocity of flow, Soil moisture content, soil heaviness and viscosity, as well as visual observations, among others. These test devices should be compatible with various protocol standards, such as Zigbee, serial bus, Process Fieldbus, Wi-Fi, and others.

Manuscript received on 12 February 2025 | First Revised Manuscript received on 17 February 2025 | Second Revised Manuscript received on 25 March 2025 | Manuscript Accepted on 15 April 2025 | Manuscript published on 30 April 2025.
*Correspondence Author(s)

Changjian Deng*, Associate Professor, School of Automation, Chengdu University of Information Technology, Chengdu, China. Email ID: chengli_dcj@163.com, ORCID ID: 0000-0002-6471-8623

Yaozhong Zhang, Student, School of Automation, Chengdu University of Information Technology, Chengdu, China. Email ID: 1102070325@qq.com

The intelligent network gateway has the basic function of store and forward [3]. Also, it is the basic function of the multi-FPGA node of computer networks [4]. To do this, it first caches the incoming packets from the input port, checks whether the packets are correct, and filters out conflicting packet errors. After confirming that the package is accurate, it finds the desired output port address and then sends the package out.

The intelligent FPGA Gateway can communicate better with different network protocols, and it has high reconfigurability, fast conversion speed, high flexibility, and so on [5].

The contributions of the paper include:

A. It proposes an optimal model using a redundancy structure based on Type-2 fuzzy systems with redundant supplementary characteristics.

B. It presents the redundancy arithmetic of a distributed multi-FPGA network gateway.

The paper is organised as follows: Section 1 presents the related work; Section 2 discusses the main issues of the multi-FPGA network gateway design; Section 3 provides details of the hardware and software design; Section 4 presents the results; Section 5 concludes the paper.

II. THE RELATED WORK

A. FPGA Redundancy Technology

FPGAs are widely used in communication, automatic control, image, video, and other fields due to their advantages of high computing density and flexible, configurable functions. It is one of the essential platforms for realizing large-scale digital systems. In the paper [6], an accelerated routing mechanism using a Field-Programmable Gate Array (FPGA) to reduce the latency for a gateway was proposed. A modular gateway prototype based on an FPGA is a powerful alternative for rural gateways. To minimise bandwidth requirements, a scheduling algorithm based on the earliest deadline first (EDF) approach was proposed. EDF forwarding was able to achieve real-time capability using fewer resources. In [7], a configurable low-latency gateway architecture on a hybrid FPGA was proposed. A translation mechanism based on a virtual media access control (MAC) address was used.

Meanwhile, with the advancement of integrated circuit process technology, the continuous improvement of FPGA capacity and performance also makes it more susceptible to ageing effects (resulting in increased failure rates and reduced reliability), and this is the same case in many harsh environments [8]. Therefore, in some research, it is of great significance to realize the self-healing mechanism [9]. And it can be combined with self-test, self-diagnosis, self-healing, and self-mitigation by using the re-configurable capability of FPGA at runtime to ensure long-term reliable operation of FPGA-based digital systems [10].

According to different fault manifestations, FPGA redundancy technology research can also be implemented at varying levels of abstraction [11]. Architecture-level redundancy research can make full use of the advantages of complex digital system software and hardware coordination, relatively large space for optimization and management, and to a certain extent, it is separated from the details of the underlying circuit structure of FPGA, and has strong flexibility and applicability [12]. And, the digital system is decomposed into memory subsystem, communication subsystem and operation subsystem at the architecture level [13].

In the memory redundancy subsystem, researchers [14] integrates fault testing, fault repair, and aging mitigation. The method realizes the software-insensitive online test, avoids the intrusiveness to the memory subsystem, ensures the real-time performance of the FPGA calculation, and can realise the early warning of faults [15].

In the communication redundancy subsystem, the study [16] takes the on-chip network as the core, and proposes a self-healing method that combines the test method oriented to the transmission path with the degraded reconfigurable repair, refresh reconfigurable repair and other FPGA dynamic partial reconfigurable repair methods. An online self-healing in redundancy communication subsystem is designed, and an FPGA fault injection experiment is carried out [17].

In the computing redundancy subsystem, the paper [18] propose a self-healing method that combines three-mode redundancy test. The paper [19] dynamic partial reconfigurable repair of reserved resources, and roll-forward synchronization strategy, which has a small time overhead.

For dedicated array hardware accelerators, a self-healing method combining functional testing, mapping adjustment and repair of target algorithms is proposed [20]. The self-healing methods of the above two types of typical computing subsystems are integrated to form a self-healing process. An online self-healing computing subsystem is designed, and an FPGA fault injection experiment is carried out.

B. The FPGA Gateway Design

Yang Hang Lee proposed an IoT gateway framework to implement functions such as data storage, analysis, and computation [21]. Jutila Mirjami proposed an adaptive algorithm based on regression access control and fuzzy weighted queuing to solve the resource allocation problem in large-scale network systems such as industrial Internet [22]. Narayanan Revathy proposed a heterogeneous gateway that includes a protocol probability model, which can predict protocol transition probabilities by analysing protocol data, thereby improving gateway processing efficiency. Applying predictive algorithms to protocol loading scheduling provides a new approach [23].

Some works of intelligent gateway are described below: Zhang Xu and others designed a multi-core Linux-based approach to address the scheduling problem of network communication protocols [24]. Akasiadis Charilaos proposed a multi-protocol IoT platform that combines several open-source frameworks (RabbitMQ, Ponte, OM2M, and RDF4J) for loading and scheduling problems of several application layer protocols such as MQTT, HTTP, AMQP, etc [25]. Song Xin and others proposed an industrial Internet gateway based on microservice architecture, and studied the task unloading and resource allocation of the industrial Internet gateway. A multi-level task unloading model was proposed, and the optimal industrial Internet gateway resource scheduling strategy was solved through a heuristic algorithm [26]; Shi Yuntao and others proposed a method for parsing multiple protocol data and converting it into a unified format. A protocol conversion method for standard data formats and an asynchronous processing mechanism that supports breakpoint continuation and can quickly handle concurrent tasks are proposed to ensure the real-time reliability of protocol conversion [27].

As described in sections 1.1 and 1.2, research on multi-FPGA network gateway design should be studied in both rural and industrial applications, and the optimal method of redundancy subsystem may be explored in greater depth. The paper addresses the central issue of multi-FPGA network gateway design, including its hardware and software aspects.

III. THE MAIN ISSUE OF THE MULTI FPGA NETWORK GATEWAY DESIGN

A. The Basic Consideration

- In many applications, a network gateway has demands for large quantities (data stream) of real-time storage and re-transmission (its performance parameters: data processing throughout, processing time, saving amount, etc.)

The technical principal of the real time data processing in network gateway involves many concepts:1)Streaming Data: Refers to a continuously generated stream of data, such as sensor data, transaction data, etc.2) Real time Data Processing Engine: refers to a software platform used for processing real-time data, typically using a streaming computing framework.3) Message Queue: Used for storing and managing data streams, such as Kafka, RabbitMQ, etc.4) Window: Used for grouping and summarizing data streams, such as sliding windows, scrolling windows, etc.5) Event Driven Architecture: refers to a software architecture based on event triggered processing logic, typically used for real-time data processing.

- The arithmetic for the construction of multiple MCUs and FPGAs. There are three kinds of sharing Schema: Shared Everything - Completely transparent shared CPU, memory, and I/O, with the worst parallel processing capability; performance improvement heavily relies on upgrading single-machine hardware and always has a ceiling.

Shared Storage: Each processing unit uses its private CPU and Memory, sharing the disk system.

Shared Nothing: Each processing unit has its private CPU, memory, and hard disk, and communication between processing units is achieved through protocols, resulting in improved parallel processing and expansion capabilities.

The paper uses multiple MCUs to perform data processing. The paper adopts the Shared Nothing architecture, a non-shared architecture where each node has its CPU, memory, and storage, and information exchange between nodes can only be achieved through network connections.

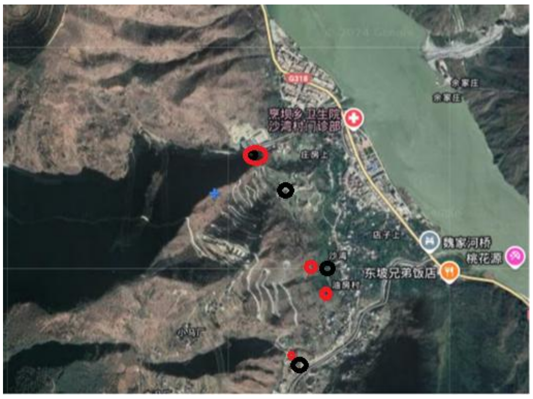
B. The Optimization Deploy Method of Dynamic Multi-FPGA Network Gateway

- Optimisation of deployment in field environments involves issues related to the static and dynamic reliability optimisation of multi-point measurement results, as well as the optimisation of replacement locations for measurement deployments.

In the deployment of flood prevention or fire prevention gateways in rural areas, the static deployment problem involves arranging detection points at key locations to ensure timely alarms, especially in unmanned environments. From this, technical parameters, including detection probability, detection places, favourable false alarm rates, negative false alarm rates, reliability, cost, and long-term sustainability, are used to determine the number, location, and usage of the selected monitoring points.

The general provisions for monitoring of debris flows should be based on the collection of data and on-site investigations to develop a monitoring plan. The collection of data and on-site investigations should meet the following requirements:

- Geological surveys, mapping, and exploration data should be collected to reflect the basic characteristics, hazard objects, hazard forms, and hazard levels of debris flow gullies;
- Analyze the formation conditions of debris flows, the development of adverse geological phenomena, source types, distribution, reserves, drainage conditions, and other disaster-causing factors;
- A topographic map should be drawn to scale with a scale of no less than 1:10,000.



[Fig.1: The Network Gateway Deployment of Monitoring of Debris Flow]

Figure 1 illustrates a distributed network gateway in a rural village. There are two rivers named the Zhaojia River and the Weijia River, as well as four large streams. The town is situated on a steep hillside, which is prone to landslides. Landslide and debris flow monitoring points are sometimes not needed at all, and occasionally, many are required depending on weather and other disaster conditions. Even if multiple monitoring points are arranged, the fusion of these monitoring points only provides probabilistic information, and sometimes, redundancy needs to be eliminated. Sometimes, due to insufficient monitoring points, practical control information such as forecasts and warnings cannot be provided. For example, there are three black circles; they typically propose an active alarm signal or valid information. And the four red circles are sometimes necessary, and some are used solely to prevent the loss of black detection points.

The monitoring level of debris flows should be classified into three levels, as shown in Table I, based on their susceptibility and the degree of harm they pose. The assessment of the susceptibility and hazard level of debris flows shall be determined by DB11/T 893.

Table I: Monitoring Level of Debris Flow

Susceptibility	**Hazard Level**		
	High	**Middle**	**Low**
High Occurrence Probability	Level 1	Level 1	Level 2
Middle Occur Probability	level 1	Level 2	Level 3
Low Occurrence Probability	level 2	Level 2	Level 3

The monitoring method should be determined based on the specific monitoring items, environmental conditions, and other relevant factors.

For example, Level 1 should measure rainfall, infrasound, Mud level, velocity of flow, Soil moisture content, heavy metal content, viscosity, and visual observations, among other parameters. The level 2 should test rainfall, infrasound, Mud level, and velocity of flow. The level 3 should test rainfall, infrasound, and mud level.

Dynamic deployment optimization

It generally involves selecting effective deployment locations at different monitoring points based on changing conditions, or leaving limited channels for the most effective locations. This is a redundant deployment optimization problem.

This is a switching problem between parallel or primary/secondary networks. In Figure 1, it is necessary to obtain effective places all the time.

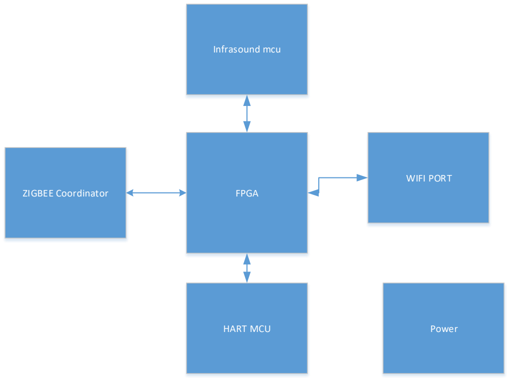
For dynamic optimization of wireless redundant networks, it is necessary to:

- Using wireless heartbeat signals to detect the effectiveness of the other party's activities;
- Using self-inspection to ensure fault avoidance;
- Utilize mutual inspection to select suitable nodes;
- Wireless Triple Redundancy Technology
- Wired and wireless switching technology.

IV. MATERIALS AND METHODS

A. Hardware Design

The construction of a network The gateway system is shown in Figure 2. The FPGA gateway receives signals from Wi-Fi, HART, Zigbee, and the infrasound test signal. It can receive signals from four different types, and has a large backplane bandwidth to ensure that data can be transmitted inside the gateway without blocking, even during peak network load periods. Therefore, it offers high throughput and low latency, meeting the demands of various applications.

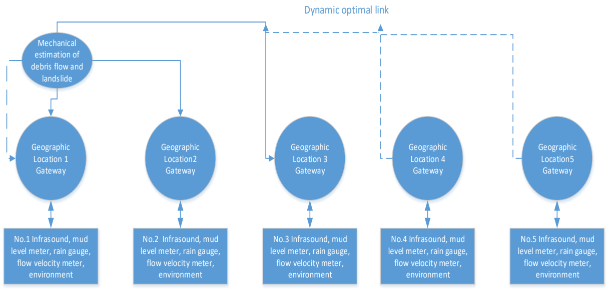


[Fig.2: The System Functions Figure]

B. Software Design

The concept of redundancy in arithmetic originates from the ecosystem. The redundant structure of an ecosystem is a multi-parallel system, and when the system is disturbed, it may lead to the loss of a specific component (or level). However, the system can be restored through redundancy supplementation. This is because biological systems exhibit a rapid self-healing function through redundancy supplementation.

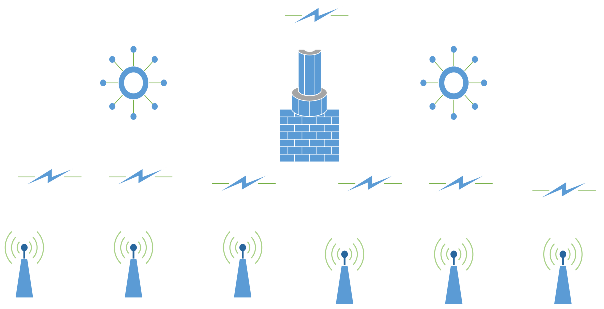
In complex and ever-changing environments, the lifespan of specific components in biological systems is limited. Therefore, failure will occur, and if a component fails, new components must be switched in and supplemented to maintain the system's functionality and structure. In the long-term evolution of a species or community, it is through the continuous acquisition of new components with the same (or similar) function to replace ineffective components. This redundant replenishment characteristic is a fundamental aspect of the entire biological succession process, manifesting as the automatic replenishment of redundancy. This crucial mechanism enables organisms or communities to recover quickly. In Figure 3, the real line represents the active link, and the dashed line represents the dynamic optimal link; they can be transformed into each other.


Dynamic optimal link
Mechanical estimation of debris flow and landslide
Geographic Location 1 Gateway
Geographic Location2 Gateway
Geographic Location 3 Gateway
Geographic Location 4 Gateway
Geographic Location5 Gateway
No.1 Infrasound, mud level meter, rain gauge, flow velocity meter, environment
No.2 Infrasound, mud level meter, rain gauge, flow velocity meter, environment
No.3 Infrasound, mud level meter, rain gauge, flow velocity meter, environment
No.4 Infrasound, mud level meter, rain gauge, flow velocity meter, environment
No.5 Infrasound, mud level meter, rain gauge, flow velocity meter, environment

[Fig.3: The Optimal Redundancy Link Diagram]

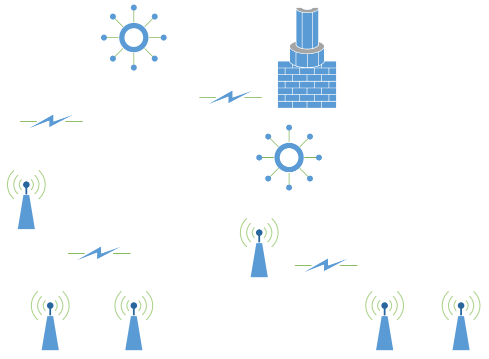
A natural tendency of redundancy is to maximise the use of limited resources to achieve maximum functionality. Detection components are combined in parallel, and the failure of one device will not affect the system's function.

Normally, the greater the redundancy of devices at the same level of a community, the higher the stability of that level. Redundancy at the network level - the combination of species redundancy and hierarchical redundancy forms a multiple parallel redundancy structure. The more complex the parallel-structured system, the higher its reliability, and the stronger its ability to operate stably will be. It is shown in Figure 4.



[Fig.4: The Two Redundancy Network Gateways with Full Link]

To analyse the working status, a mathematical model of redundant multiple parallel structures should be considered. As more link is built, as shown in Figure 5, it consists of two redundant network gateways with an optimal link. The communication resources are discreet, and the redundancy is also maintained.



[Fig.5: The Two Redundancy Network Gateways with Optimal Link]

This redundant mathematical model has a more complex fuzzy set or membership function. Then, a Type-2 fuzzy system may be suitable for these redundant supplementary characteristics. For type-2 fuzzy sets, this overcomes the difficulties in constructing conventional membership functions. Usually, when it is difficult to determine the membership function of a fuzzy set, Type-2 fuzzy sets are very effective, as it is more effective in describing these complex things [28]. The reasons are below:

Firstly, according to the Type-2 fuzzy set, it has unique uncertainty and the "broadband" effect of membership functions.

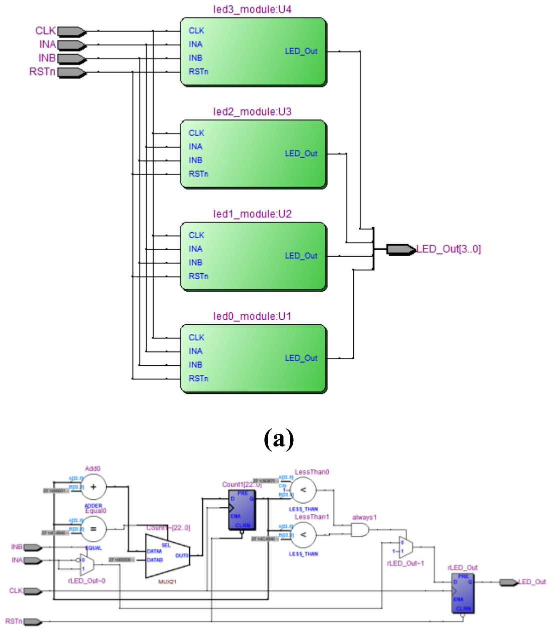
Secondly, because the Type-2 fuzzy system itself is a multiple parallel structure [29], the Type-2 fuzzy system can represent the redundant structure and redundant supplementation characteristics more effectively.

To realise the redundancy network gateway, the paper uses a hierarchical voting table based on self-testing and repair strategies. This has a fuzzy rule, but has a very suitable

MF for every input.

C. The RTL Figure of Redundancy FPGA

The multi-redundancy FPGA IO control demo is shown in Figure 6. The output is based on inputs INA and INB; for example, when INA and INB are valid within a specific period, the output LED is activated. Figure 6(a) is the top RTL figure, and Figure 6(b) is the RTL of the LED_0 module.

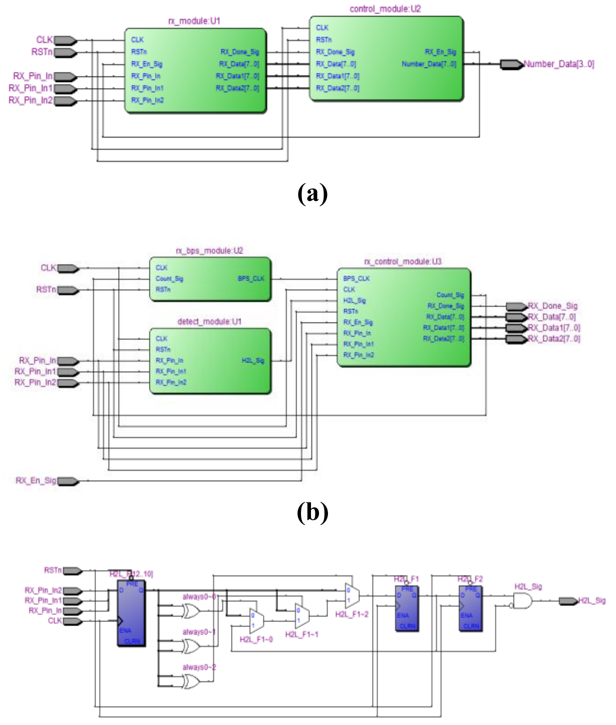


(a)

(b)

[Fig.6: The Multi Redundancy FPGA IO Control Demo]

The multi-redundancy FPGA UART control demo is shown in Figure 7. There are three UART inputs, and a hardware Voter produces the output. Figure 7 (a) is RTL of the top module, Figure 7(b) is RTL of the rx_module, and Figure 7(c) is RTL of the detect module.



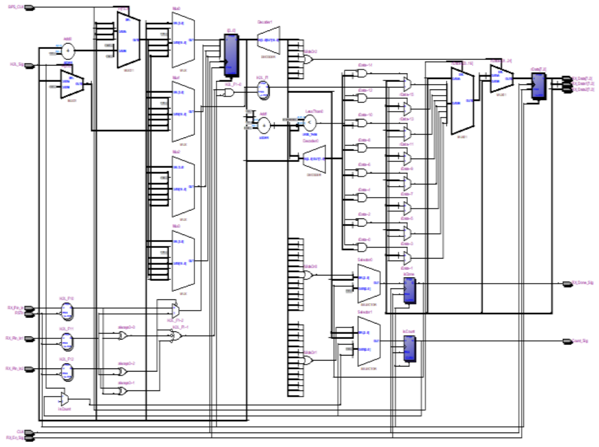
(a)

(b)

(c)

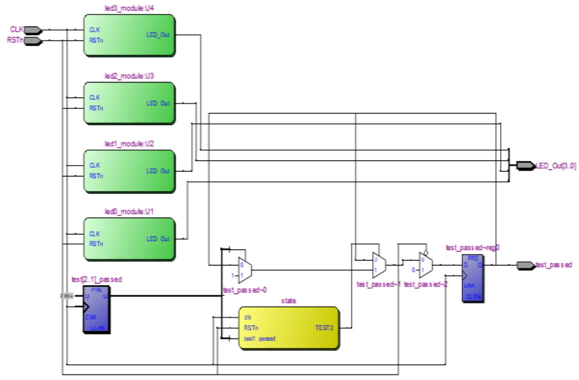
[Fig.7: The Multi Redundancy FPGA UART Control Demo]

The multi-redundancy FPGA wireless communication (UART) control demo is shown in Figure 8. In a real distributed network gateway, wireless communication is usually used, and the voter determines whether to receive a byte or not within a specific heartbeat time.



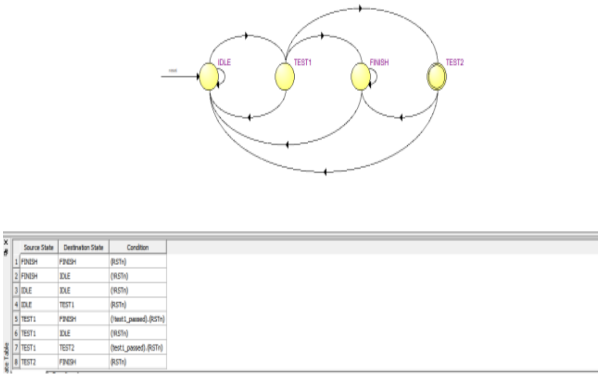
[Fig.8: The Multi Redundancy FPGA Wireless Communication (UART) Control Demo]

The self-test demo is shown in Figure 9.



[Fig.9: Self-Test Demo]

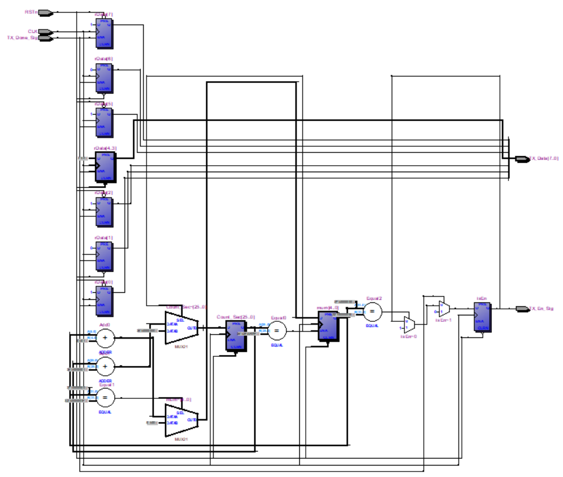
The state machine for self-test is shown in Figure 10.



	Source State	Destination State	Condition
1	FINISH	FINISH	(RSTn)
2	FINISH	IDLE	(!RSTn)
3	IDLE	IDLE	(!RSTn)
4	IDLE	TEST1	(RSTn)
5	TEST1	FINISH	(!test1_passed).(RSTn)
6	TEST1	IDLE	(!RSTn)
7	TEST1	TEST2	(test1_passed).(RSTn)
8	TEST2	FINISH	(RSTn)

[Fig.10: The State Machine of Self-Test]

The multi-redundancy FPGA wireless heartbeat communication (UART) transmitting control demo is shown in Figure.



[Fig.11: Wireless Heartbeat Communication (UART) Transmitting Control Demo]

V. RESULTS

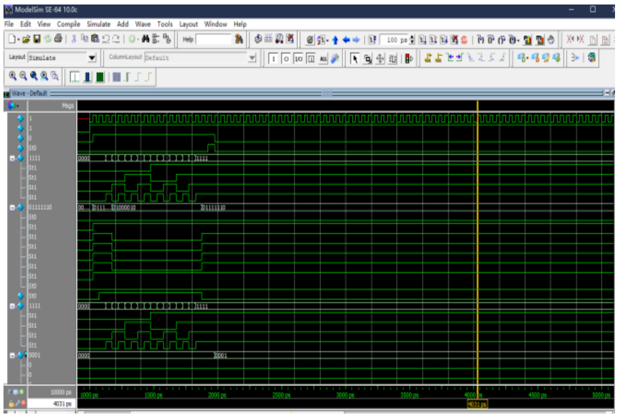
A. Simulation Results

i. Compilation

All components generated for the redundancy design were simulated in three FPGA hardware platforms. Compilation of the code generated for the function is successful. Compilation is successful for different FPGAs, which are identified by their connections.

ii. Simulation

All the software has passed the simulation. Figure 12 is a demo of UART sending.



[Fig.12: The Simulation of UART Send]

B. Test Results

i.Prepare Work. First, Should Connect it to the Atomic Cloud Server

The Atomic Cloud, also known as the Atomic Cloud Server, is an Internet cloud service platform launched by Atom on schedule, which enables remote data monitoring, forwarding, and management functions. Atomic Cloud Domain: https://cloud.alientek.com. Its port number is 59666.

Before creating a device, log in to the Atomic Cloud website at https://cloud.alientek.com/. Create and log in to an account.

To create a device on the Atomic Cloud platform, first click on "Device Management" in the left navigation bar of the Atomic Cloud homepage. Then, on the current page, click on "Add Device”. Next, in the pop-up window, select "ESP8266" based on the device type of the ATK-MW8266D module, and then root it. Fill in the "device name" and "password" according to personal preferences.

After creating the device, you can connect to the atomic cloud using the ATK-MW8266D module, specifying the device number and password, and through AT commands. The specific AT command (set in the FPGA setting port or internally) is as follows:

AT+CWMODE=1
AT+CWJAP=” XXXX-XXXX”,” XXXXXXXXXX”
AT+ATKCLDSTA=”YYYYYYYYYYYY”,”YYYYYY”

X corresponds to the router name and password, and Y corresponds to the atomic cloud account and password.

ii.Connect the Device to the Network Gateway

According to the IO assignment, connect it to every detection device.

Power these devices, and then all the test data from the multi-redundancy FPGA can be sent to the atomic cloud. The redundancy arithmetic was carried out well.

VI. CONCLUSIONS

The paper proposes a distributed, multi-redundant FPGA gateway to monitor disasters, which can test a series of parameters based on estimates of geological disasters.

The FPGA gateway receives signals from Wi-Fi, HART, Zigbee, and the infrasound test signal. It can receive signals from four different types, and has a large backplane bandwidth to ensure that data can be transmitted inside the gateway without blocking, even during peak network load periods. Therefore, it offers high throughput and low latency, meeting the demands of various applications.

Primarily, it involves distributed redundancy optimisation problems. The article adopts a redundant gateway design that utilises multiple FPGAs (and MCUs) in hardware and employs distributed redundant decision-making in software. It analyzes and compares various wireless redundancy decision-making optimization methods, and can transmit data to ‘the atomic cloud’.

DECLARATION STATEMENT

After aggregating input from all authors, I must verify the accuracy of the following information as the article's author.

- **Conflicts of Interest/Competing Interests:** Based on my understanding, this article does not have any conflicts of interest.
- **Funding Support:** This article has not been sponsored or funded by any organization or agency. The independence of this research is a crucial factor in affirming its impartiality, as it was conducted without any external influence.
- **Ethical Approval and Consent to Participate:** The data provided in this article is exempt from the requirement for ethical approval or participant consent.
- **Data Access Statement and Material**

Availability: Adequate resources for this article are publicly accessible.

- **Author's Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. Aseeri M A S,Alasows A A,Ahmad M R(2016). Design of DRFM system based on FPGA with high resources[C].Design & Test Symposium (IDT),2016 11th International. IEEE: DOI: https://doi.org/10.1109/IDT.2016.7843036
2. MAGDALENO E,RODRÍGUEZ M,PÉREZ F,et al(2013). A FPGA embedded web server for remote monitoring and control of smart sensor networks [J]. sensors,14(1): 416-430. DOI: https://doi.org/10.3390/s140100416
3. Mittal, S. (2018). A survey of FPGA-based accelerators for convolutional neural networks. Neural Computing and Applications. DOI: https://doi.org/10.1007/s00521-018-3761-1
4. Zhao, L. (2024). A scalable multi-fpga platform for hybrid intelligent optimization algorithms. Electronics, 13. DOI: https://doi.org/10.3390/electronics13173504
5. Bai, L. , Zhao, Y. , & Huang, X. . (2018). A CNN accelerator on an FPGA using depthwise separable convolution. Circuits and Systems II: Express Briefs, IEEE Transactions on, PP(99), 1-1. DOI: https://doi.org/10.1109/TCSII.2018.2865896
6. Gholami, M. , Farsa, E. Z. , & Karimi, G. . (2021). Reconfigurable field-programmable gate array-based on-chip learning neuromorphic digital implementation for nonlinear function approximation. International Journal of Circuit Theory and Applications. DOI: https://doi.org/10.1002/cta.3075
7. Sahoo S S, Ranjbar B, Kumar A(2021). Reliability-Aware Resource Management in Multi-/Many-Core Systems: A Perspective Paper[J]. Journal of Low Power Electronics and Applications, 2021, 1 1(1). DOI: https://doi.org/10.3390/jlpea11010007
8. Li X, Yan G, Ye J, et al(2018). Fault tolerance on-chip: a reliable computing paradigm using self-test, self-diagnosis, and self-repair (3S) approach[J]. Science China Information Sciences, 2018, 61(11): 1-17. DOI: https://doi.org/10.1007/s11432-017-9290-4
9. Wirthlin, M. (2015). High-reliability FPGA-based systems: space, high-energy physics, and beyond. Proceedings of the IEEE, 103(3), 379-389. DOI: https://doi.org/10.1109/JPROC.2015.2404212
10. Reorda, M. S., Sterpone, L., & Ullah, A. (2013). An error-detection and self-repairing method for dynamically and partially reconfigurable systems. IEEE. DOI: https://doi.org/10.1109/ETS.2013.6569377
11. Lee, J. , Bhagavatula, S. , Bhunia, S. , Roy, K. , & Jung, B. (2012). Self-healing design in deep-scaled CMOS technologies. Journal of Circuits Systems & Computers, 21(06), 1240011-1240011-15. DOI: https://doi.org/10.1142/S0218126612400117
12. Narasimhan S, Paul S, Chakraborty R S, et al(2010). System-level self-healing for parametric yield and reliability improvement under power bound[C]//2010 NASA/ESA Conference on Adaptive Hardware and Systems. IEEE, 2010: 52-58. DOI: https://doi.org/10.1109/AHS.2010.5546231
13. Huang, X. , Zhao, Y. , Wang, X. , Wang, F. , Liu, L. , & Yang, H. , et al. (2024). Implementing versatile programmable logic functions using two magnetization switching types in a single device. Advanced Functional Materials, 34(3). DOI: https://doi.org/10.1002/adfm.202308219
14. Ilwoong, Kim, Woosik, Jeong, Dongho, & Kang, et al. (2015). Fully programmable memory bist for commodity drams. ETRI Journal, 37(4), 787-792. DOI: https://doi.org/10.4218/etrij.15.0115.0040 0
15. Yuan, C. , Huang, L. , Wang, J. , & Li, Q. . (2018). Microarchitecture design for low-overhead fault-tolerant network-on-chip. IEEE. DOI: https://doi.org/10.1109/ISCAS.2018.8351501
16. Liu, J., Harkin, J., Yuhua, L., & Maguire, L. P. (2014). Online fault detection for networks-on-chip interconnect. IEEE. DOI: https://doi.org/10.1109/AHS.2014.6880155
17. Wang J, Ebrahimi M, Huang L, et al(2019). Efficient design-for-test approach for networks-on-chip[J]. IEEE Transactions on Computers, 68(2): 198-213. https://ieeexplore.ieee.org/document/8440731
18. Zhan J, Huang L, Wang J, et al(2019). Online path-based test method for network-on-chip[C]//2019.IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, 2019: 1-5. DOI: https://doi.org/10.1109/ISCAS.2019.8702409
19. Pavan, Poluri, Ahmed, & Louri. (2016). Shield: a reliable network-on-chip router architecture for chip multiprocessors. IEEE Transactions on Parallel and Distributed Systems, 27(10), 3058-3070. DOI: https://doi.org/10.1109/TPDS.2016.2521641
20. Vip in K, Fahmy S A(2018). FPGA dynamic and partial reconfiguration: A survey of architectures, methods, and applications[J]. ACM Computing Surveys (CSUR),5 1(4): 1-39. DOI: https://doi.org/10.1145/3193827
21. Lee, Y. H. , & Nair, S. . (2017). A Smart Gateway Framework for IOT Services. IEEE International Conference on Internet of Things. IEEE. pp. 107-114. DOI: https://doi.org/10.1109/iThings-GreenCom-CPSCom-SmartData.2016.44
22. Jutila M(2016). An Adaptive Edge Router Enabling Internet of Things[J]. Internet of Things Journal, IEEE, 2016, 3(6):1061-1069. DOI: https://doi.org/10.1109/JIOT.2016.2550561
23. N Arayanan R, Murthy C(2017). A probabilistic framework for protocol conversions in IoT networks with heterogeneous gateways[J]. IEEE Communications Letters, 2017:1-1. https://ieeexplore.ieee.org/document/7990138
24. L. Yang, Y. Wei, F. R. Yu and Z. Han(2022), Joint Routing and Scheduling Optimization in Time-Sensitive Networks Using Graph-Convolutional-Network-Based Deep Reinforcement Learning, in IEEE Internet of Things Journal, vol. 9, no. 23, pp. 23981-23994, 1 Dec.1, 2022, DOI: https://doi.org/10.1109/JIOT.2022.3188826
25. Akasiadis, C., Pitsilis, V., & Spyropoulos, C. D. (2019). A multi-protocol IoT platform based on open-source frameworks. Sensors, 19(19), 4217-. DOI: https://doi.org/10.3390/s19194217
26. Song Xin(2020). Design and Implementation of Industrial Internet Gateway Based on Microservice Architecture [D]. Beijing Jiaotong University. https://cdmd.cnki.com.cn/Article/CDMD-10056-1021826347.htm
27. Zhang, Y. , Sun, W. , & Shi, Y. . (2020). Architecture and Implementation of Industrial Internet of Things (IIoT) Gateways. IEEE. DOI: https://doi.org/10.1109/ICCASIT50869.2020.9368773
28. Sun Xihao, Li Yimin, Hua Jing, etc(2007). Type-2 mathematical model for redundant structures in biological systems [J]. Fuzzy systems and mathematics, 21(5):8. https://0.3969/j.issn.1001-7402.2007.05.028
29. Liang Q， Mendel J M(2001)．Interval type-2 fuzzy logic system：theory and design〔J〕．IEEE Trans．Fuzzy systems，2001，132：195～220. DOI: http://dx.doi.org/10.1109/91.873577

AUTHOR'S PROFILE

Changjian Deng is an associate professor at the School of Automatic Engineering, Chengdu University of Information Technology. He obtained a Ph.D. degree from the University of Electronic Science and Technology of China in 2015. His research focuses on intelligent detection technology and devices, sensor networks, and their applications. He is a project leader of Sichuan Education Department project: "Research on Acoustic Photography Technology for Valve Leakage Detection In Adverse Environments", Talent Introduction Project of the Chengdu University of Information Technology: "Research on Wireless Image Data Terminals for Ad Hoc Networks". His research is notable for its application of wireless acoustic photography technology in valve leakage monitoring, particularly in harsh environments.

Yaozhong Zhang is a Master's student in Control Science and Engineering at Chengdu University of Information Science, graduating in 2020 with a major in Measurement and Control Technology and Instruments. Currently engaged in FPGA industrial gateway design and LVDT bus design, and have completed the infrasound measurement device and preliminary on-site testing. He is a master's degree holder from the School of Automation, Chengdu University of Information Technology, Chengdu, China.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of the Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)/ journal and/ or the editor(s). The Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP) and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.