Design and Implementation of FPRP on FPGA for Internet of Things

Shuning Lei, Zhongjiang Yan(✉), Xiaojiao Hu, Mao Yang, and Bo Li

School of Electronics and Information, Northwestern Polytechnical University, Xi'an, China
593154441@qq.com, {zhjyan,yangmao,libo.npu}@nwpu.edu.cn, 18392990273@mail.nwpu.edu.cn

Abstract. To guarantee low energy consumption and high efficiency for the nodes of the Internet of Things (IoT), designing and implementing the network protocols, e.g., routing and multiple access control (MAC) protocols, on FPGA become significant issues. Aiming at designing and implementing FPRP protocol on FPGA, this paper proposes a design scheme of FPGA implementation of FPRP protocol. The overall architecture includes clock counting module, reservation module, data transmission module, algorithm module and reservation result generation module, which are connected and restricted with each other. In order to verify, we divide it into two parts, i.e., single node verification and double node verification. Finally, the basic implementation of the five-step reservation is initially completed which show that the design scheme is correct and feasible.

Keywords: FPRP · Multi - access protocol · MAC · FPGA

1 Introduction

With the rapid development of communication technology and the improvement of people's communication requirements, Internet of Things (IoT) has become an important research hotspot of nowadays network. Medium Access Control (MAC) protocol has great influence on the performance of wireless sensor networks and is one of the key protocols to ensure efficient communication for IoT. Different from the contention based protocols, e.g., Carrier Sense Multiple Access with Collision Avoid (CSMA/CA) [1–3] and ALOHA [4–6], the performance of which may deteriorate when the network density becomes high. The joint contention and reservation-based protocol, e.g., Five-Phase Reservation Protocol (FPRP) [7–11], has both flexible and high efficiency advantages. To guarantee low energy consumption and high efficiency for the nodes of the IoT, how to design and implement the joint contention and reservation-based protocol, e.g., FPRP, becomes a significant issue.

Y.-B. Lin and D.-J. Deng (Eds.): SGIoT 2020, LNICST 354, pp. 178–194, 2021.
https://doi.org/10.1007/978-3-030-69514-9_15

FPRP divides the channel into frames, and each frame contains two stage, i.e., the reservation stage and the data transmission stage. The nodes that need to transmit traffic initiate the reservation process with probability P in the reservation stage, and they interactively contend channel resources by controlling the grouped broadcasting type. Once the contention is successful, the node will reserve the broadcast information slot corresponding to the reservation slot and automatically occupy the certain slot for data transmission until the next reservation period comes. If the node still needs an information slot, the slot resource is re-competed in the next reservation phase.

The node completes the reservation of information slot after five information interactions in the contention cycle. The basic process of five handshake runs in FPRP is as follows.

- Reservation Request phase (RR): Requesting Node (RN) sends reservation request with probability P.
- Contention Report phase (CR): The nodes of the channel are monitored. If multiple RR are received, CR is sent to indicate that the channel contention occurred. If the RN does not receive CR, then its RR is sent successfully, and RN becomes Transmission Node (TN).
- Reservation Confirmation phase (RC): If TN does not receive CR, it means that the reservation is successful. After that, TN sends RC, and TN's neighbor node is informed that the slot is reserved by TN. Otherwise, do not send RC.
- Reservation Acknowledgement phase (RA): The node receiving RC replies to RA and tells its neighbor TN that its reservation has been successful. Nodes that cannot receive RA are isolated nodes. RA is also used to inform nodes beyond two hops that the slot is successfully reserved. The node receiving RA sets its own state to Blocked (BK).
- Packing/Elimination phase (P/E): The two-hop neighbor node of TN sends PP and tells the three-hop neighbor node of TN that the slot has been successfully booked. The three-hop neighbor node knows that its one-hop neighbor node will not compete for the slot, so the three-hop neighbor node can increase the competition probability P in its own two-hop network. At this stage, TN also sends EP to other potential TNs with a probability of 0.5 to solve the deadlock problem of non-isolated nodes.

The FPRP protocol uses Bayes-based algorithm to calculate the probability P to send messages. The algorithm stipulates that each node should estimate the number of resource competitors around it, n, and adjust its probability P.

$$P = \frac{1}{n}$$

At the end of each competition, the node updates its estimate n based on feedback.

- If the competition is successful or the competition is idle:

$$n = n - 1$$

– If contention happened:

$$n = n + (e - 2)^{-1}$$

In the research work of improving FPRP protocol, Bayes-based algorithm is taken as the focus, which is constantly updated and improved.

It has been proved by practice that the FPRP protocol adopts the five-steps handshake mechanism, which does not produce time slot waste and effectively solves the problem of hidden terminal and non-isolated deadlock. However, the FPRP protocol does not take into account such problems as unfair slot allocation caused by different node loads, low channel utilization and excessive access delay due to single channel communication. In recent years, aiming at the problems found in the application of FPRP protocol, many targeted improvement and research have been carried out. For example, in order to simplify the complexity of the implementation of FPRP protocol system, the node should have the ability to measure the signal-to-noise ratio of wireless communication link, judge whether the communication link is reliable according to the signal-to-noise ratio, and then choose whether to select the link. At the same time, it is necessary to distinguish the transmitting power in different stages. For example, in the RR and CR phases, the wireless transmitting power of the node sending RR packets can be appropriately increased to reduce interference and conflict. Besides, aiming at the problem that FPRP can not fully use some time slots in the information frame, an improved pseudo Bayesian algorithm is proposed. The improved FPRP contention access mechanism allows nodes to compete for multiple slots in the reserved frame according to a certain probability or priority, so as to improve the slot utilization.

Field Programmable Gate Array (FPGA) has a strong real-time capability to process signals in real time. In addition, it realizes various functions through software programming oriented to chip format instructions. Therefore, the original design scheme or original functions of the system can be improved by modifying the software code without changing the hardware platform, which is of great flexibility. Therefore, using FPGA as the platform to implement FPRP protocol is very convenient, practical and effective.

FPGA is a semi-custom circuit developed in recent years on the original hardware editable device. Using FPGA to implement MAC protocol can make the design products small, integrated, stable and reliable, which shorten the development cycle greatly. In addition, it is easy to debug and modify internally. In Ad Hoc networks, MAC protocol is required to be implemented on FPGA due to its advantages of high integration, fast speed and low power consumption. However, as FPGA is a hardware platform, it is a parallel working environment, while MAC protocol is usually a time-dimension sequential working process. Therefore, how to accurately describe the sequence working process of the designed MAC protocol on FPGA has become a breakthrough point that has been widely studied. In the implementation of various protocols with FPGA, for example, using FPGA to implement a subset of IEEE 802.15.4 protocol to realize point-to-point communication and compare the power consumption of wireless data

transmission system [14]. Besides, a new MAC-PHY Interface (MPI) protocol is implemented with FPGA, which realize that the throughput is 1.21 Gbps [15].

How to use FPGA to implement FPRP protocol is the key content of this paper. The main contributions of this paper are as follows.

1 Proposed a design scheme to realize FPRP protocol. Proposed to divide FPRP protocol into several modules to realize different functions, including clock counting module, algorithm module, reservation module, reservation result generation module, data transmission module, and so on. And realized various functions with Verilog language simulation.
2 Carried out simulation design for clock counting module, algorithm module and reservation module. Successfully realized the receiving and sending of double nodes and obtained correct output waveform. Proved the feasibility of the whole scheme.

This paper is divided into six sections. Section 1 introduces the research background and innovation points of the paper. Section 2 designs and divides FPRP modules on the basis of the knowledge learned. Section 3 gives the realization ideas according to the module division, and Sect. 4 gives the final simulation results. Section 5 summarizes and looks forward to the future. Section 6 gives acknowledgement.

2 Overall System Architecture Design

In order to realize FPGA design and implementation, according to different functions and MAC protocol model, FPRP protocol is divided into several key sub-modules for design simulation and implementation.

Figure 1 shows the MAC layer networking protocol module system architecture. It is mainly divided into two parts, which are MAC protocol module and memory management module. The MAC protocol module mainly realizes the logic function of MAC protocol, and the memory management module caches and transmits data to the data transmission module or the upper module according to the requirements of MAC protocol. In addition, it supports data re-transmission.

As a kind of MAC protocol, the system framework of FPRP protocol is also the same as Fig. 1. However, in order to realize the function of FPRP simply, according to Fig. 1, the simplified FPRP system is divided into five main modules as shown in Fig. 2 in addition to the memory management module.

As shown in the Fig. 2, the five parts are introduced as follows.

1 Clock counting module. It is the master control module for sending and receiving data, which decides when to send corresponding frame signals.
2 Reservation module. It contains reservation sending module and reservation receiving module. It can realize the five-step reservation mechanism, send and receive each type of frame (RR, CR, RC, RA, PP/EP), and decide whether the five handshakes proceed smoothly.

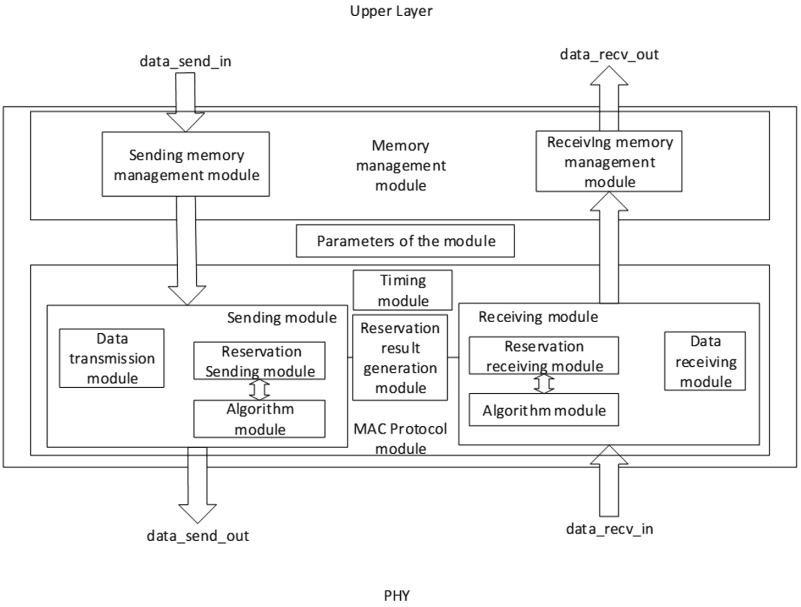

Fig. 1. MAC protocol module system architecture

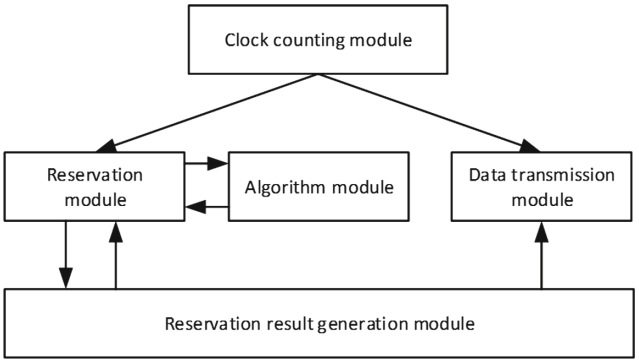

Fig. 2. Integral module division

3 Data transmission module. It contains data transmission sending module and data transmission receiving module. After the reservation is successful, the packet is transmitted.
4 Algorithm module. It can adjust the probability P to determine whether each node can make a reservation.
5 Reservation result generation module. It can generate reservation slot allocation table, which transmits the signal whether the reservation is successful.

These five modules are interrelated. Among them, the clock timing module counts according to the micro-slot of reservation and data transmission, outputs the micro-slot count, and jointly manages the reservation module with the algorithm module. The reservation results of the reservation module are saved in the reservation generation module and correlated with the algorithm module. Generate modules based on reservation results to provide relevant information for the data transfer module.

In this paper, the relationship among the reservation module, algorithm module and clock counting module is refined into several parts as shown in Fig. 3 for design.

As shown in Fig. 3, the reservation master module is composed of the total top-level module of reservation sending and reservation receiving master module. The total top-level module of reservation sending calls the sub-module of reservation sending module, the clock counting module and the algorithm module to realize the mutual restriction of the three. The sub-module of reservation sending calls the five-step reservation frame signal sending sub-module. The reservation receiving master module calls the five-step reservation receiving sub-module to realize the receiving of frames.

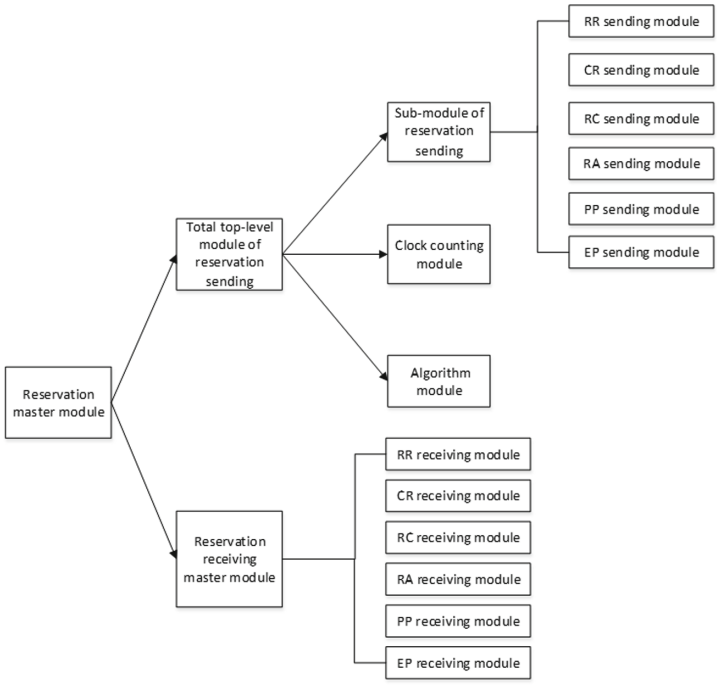

Fig. 3. The module design realized in this paper

3 Detailed Sub-module Design and Implementation

In this paper, the clock counting module, algorithm module and reservation module are presented.

3.1 Clock Counting Module

The main function of the clock counting module is to generate enabling signals needed in the Reservation Frame (RF) and Information Frame (IF), and to divide the whole time period orderly. The clock counting module is divided into two sub-modules, namely RF counting module and IF counting module. Each module generates corresponding enabling signals. The RF counting module outputs the enabling signals of the Reservation Slot (RS) counting module, which in turn outputs the enabling signals of the Reservation Cycle (RC) counting module, and then the RC module outputs the enabling signals of RR, CR, RA, RC, PP and EP phases. Each module works in an orderly manner according to the time slot.

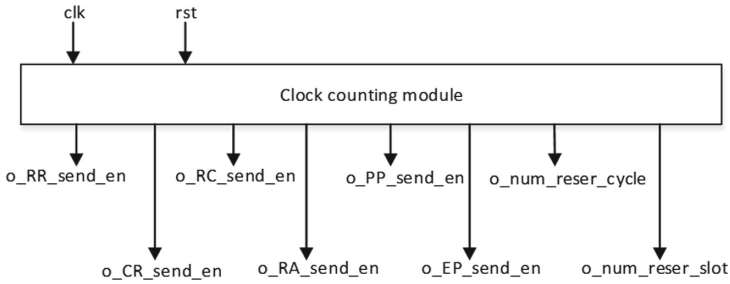

Fig. 4. Input and output signals of clock count module

Figure 4 shows the input and output of the clock counting module. Clock counting module inputs clock signal (clk) and reset signal (rst) to generate clock to set counter. Output RR packet sending enable signal (o_RR_send_en), CR packet sending enable signal (o_CR_send_en), RC packet sending enable signal (o_RC_send_en), RA packet sending enable signal (o_RA_send_en), PP packet sending enable signal (o_PP_send_en), EP packet sending enable signal (o_EP_send_en), and output current reservation cycle (o_num_reser_cycle) and current reservation slot number (o_num_reser_slot) to make time limits on reservation cycles and slots.

In order to realize this module, after the interface definition, set up a 3-bit counter, cycled counting from 0 to 4, corresponding RR, CR, RC and RA, PP/EP five time slot. At different time slot (different counter value), the corresponding frame can pull up the enable signal, e.g., o_RR_send_en, then the certain frame is allowed ti be sent. Set a reservation cycle count (o_num_reser_cycle) counter.

Every time the micro-counter counts to 4, add one. And keep counting from 0 to 2, indicating 3 reservation cycles. Set a reservation slot counter. Every time the number of reservation slots (o_num_reser_cycle) is added to 2 and then it added 1. The count from 0 to 9 is repeated, indicating 10 reservation slots.

3.2 Algorithm Module

According to the interaction of the reservation control group, the module adjusts the node's reservation probability based on the Bayes algorithm to improve the success rate of the reservation and accelerate the convergence rate of the protocol.

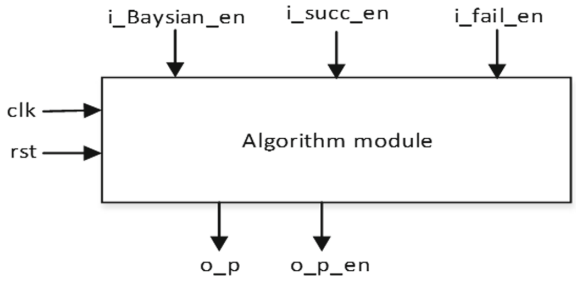

Fig. 5. Input and output signals of algorithm module

Figure 5 shows the input and output of the algorithm module. The algorithm module inputs clock signal (clk) and reset signal (rst) to generate the clock. The enable signal of the input algorithm module (i_Baysian_en) determines whether the algorithm module is called, that is, the algorithm module is called when the enabling signal of the algorithm module is high. Input the successful reservation enable signal (i_succ_en) and the failed reservation enable signal (i_fail_en) for conditional judgment, and calculate the probability P with different calculation methods according to the success or failure of the reservation. Output the send probability P (o_p) and the output P effective enable signal (o_p_en).

When the nodes receive RC group, RA group, PP group or EP group, they will generate reservation success signal to the algorithm module, that is, input i_succ_en to the algorithm module. The algorithm module will adjust the sending probability of its output through success or failure signal. If successful, refer to the formula $P = \frac{1}{n}$, $n = n - 1$. If it fails, refer to the formula $P = \frac{1}{n}$, $n = n + (e-2)^{-1}$. Firstly, use an intermediate variable register to calculate the two formulas, such as $\frac{24}{n}$ and $\frac{2520}{n}$, so that it is still a positive integer. Then use ">> 2" operation to reduce the corresponding multiple to make it a decimal. Output probability P. Generate a random number. If the probability P is less than the random number, a reservation can be made. Otherwise, no reservation will be made.

3.3 Reservation Module

Reservation module should be able to make reservation according to the probability generated by the algorithm module. When reservation, different slots output by clock counting module and the deference of the received frames are taken into consideration in order to send the five - step reservation frame in order accurately. And it should judge the frame type when the frame comes, call the corresponding frame receiving module, and receive the frame accurately.

3.3.1 RR Module

The RR frame format is shown in Fig. 6.

Fig. 6. RR frame

Type: RR group, set to 0001. The previous bit 0 means that the frame is in the reservation stage and 001 means RR frame.

Src_Addr: The source address of reservation node S.

3.3.1.1 RR Sending

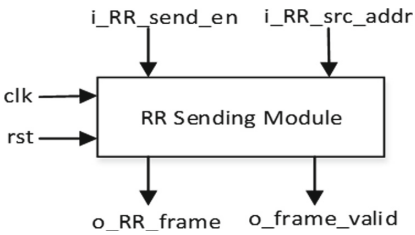

Fig. 7. Input and output signals of RR sending module

Figure 7 shows the input and output of the RR sending module. RR sending module inputs clock signal (clk), reset signal (rst) to generate the clock. Input i_RR_send_en enable signal to decide whether to call sending module. RR frame can be sent when i_RR_send_en is high. Enter i_RR_src_addr to display the source address of the node which sends the RR packet and tell the surrounding nodes who is sending the frame. Output RR packet (o_RR_frame) and the output signal display valid field o_frame_valid.

In order to realize this module, firstly set up the interface. Then define a 3-bit counter to count from 0 to 3, one for each clock cycle. When the enable signal is

high, RR frames are sent in three clock cycles. The RR frame Type field is sent in the first clock cycle. The first 32bit of the source address is sent in the second clock cycle, and the last 16bit of the source address is sent in the third clock cycle and 0 is supplemented. Once 32bit is sent, pull the o_frame_valid high.

3.3.1.2 RR Receiving

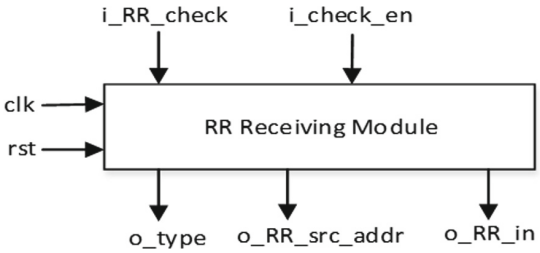

Fig. 8. Input and output signals of RR receiving module

Figure 8 shows the input and output of the RR receiving module. RR receiving module inputs clock signal (clk), reset signal (rst) to generated clock. Then input receive enable signal (i_check_en) determines whether the RR receive module is called, that is, when the receive enable signal is high, RR packet will be received. Input the received 32-bit RR group (i_RR_check) to receive. Output the type field extracted from the received RR group (o_type) and the source address that sent RR (o_RR_src_addr). After all the above points are received, the enable signal (o_RR_in) of RR group receiving is output.

In order to realize this module, firstly, set up the interface. Then define a 3-bit counter to count from 0 to 5, one clock cycle for each count. When the enable signal is high, RR frames are received in 4 clock cycles. The type field of RR group is extracted from i_RR_check in the first clock cycle. The second clock cycle extracts the first 32 bits of the sending node's source address from i_RR_check. In the third clock cycle, extract the last 16 bits of the source address of the sending node from i_RR_check and make up with 0. At the fourth clock cycle, RR group has been already received, and the enable signal (o_RR_in) is set to a high position.

3.3.2 Other Frame Types Send and Receive Modules

The design of CR, RC, RA, PP and EP sending and receiving modules are carried out in the same principle as those of RR sending and receiving module. However, the o_type of each frame is different. It is 0010 to CR, 0011 to RC, 0100 to RA, 0101 to PP and 0110 to EP.

3.3.3 Total Top-Level Module of Reservation Sending

This module call algorithm module, clock counting module and reservation sending module. Realize the normal delivery of the reservation frame under the following three constraints.

1 Time slot constraints. Call the corresponding frame sending sub-module in five steps to reserve the corresponding time slot.
2 Algorithm constraints. Start booking according to the probability generated by the algorithm module.
3 Received frame constraints. Determine which step of the five-step reservation is made according to the received frame and send the next reservation frame.

Figure 9 shows the input and output of the total top-level module of reservation sending. Input clock signal (clk), reset signal (rst) to generate the clock. Input the overall enable signal of reservation sending (i_reser_send_en) to determine whether to call the overall sending module. That is, when the overall enable signal of reservation sending is high, it is called. Input the source addresses (i_RR_src_addr, i_CR_src_addr, i_RC_src_addr, i_RA_src_addr, i_PP_src_addr, i_EP_src_addr). Input the enable signal of contend (i_contention_en) to simulate the occurrence of contend and send CR frame. Input the completion enable signals (o_RR_in, o_CR_in, o_RC_in, o_RA_in, o_PP_in, o_EP_in) of each frame, connecting to the receiving module, and determine which frame to send in the next handshake by the received frame. Output the frame signal that was sent (frame_out), and print out the frame's valid field (frame_invalid).

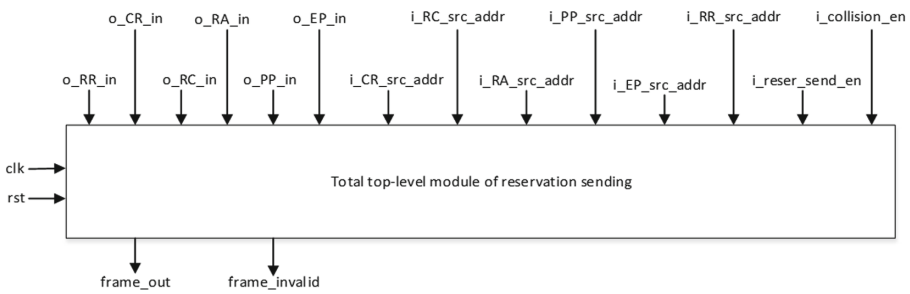

Fig. 9. Input and output signals of total top-level module of reservation sending

In order to realize this module, firstly set up the interface. Then input source address. Input contend enable signal (i_contention_en) to simulate contend situation. Input o_RR_in and so on to enable module to know which frame has been received. Call the timing module to integrate the constraint condition 1, time slot constraint, to send enable signals for each frame, that is, only the corresponding frames can be sent in the corresponding time slot. Call the sub-module of reservation sending, which indicates that only when the sending enable signal

is high can the sending module of each frame be called to send the frame. The algorithm module is called to calculate the reservation probability and determine whether the node should make an appointment or not. In this overall module, another constraint condition to send enable signals of each frame is set, that is, sending according to the received packet. For example, if only the RC packet (i.e., the input o_RC_in is high) is received, the sending module of the next frame, i.e., RA frame, can be sent. Output the packages the node sent.

3.3.4 Reservation Receiving Master Module

It can judge the type of frame and then call the corresponding sub-module to receive the certain frame signal.

Figure 10 shows the input and output of the reservation receiving master module. Input clock signal (clk), reset signal (rst) to generated clock. Input the total enable signal of reservation receiving (i_reser_recv_en) to determine whether to call the total module of reservation receiving. Input the received frame signal (frame_in) and call each receiving sub-module to receive the frame. Output the completion enable signals (o_RR_in, o_CR_in, o_RC_in, o_RA_in, o_PP_in, o_EP_in) of each frame after receiving, which is connected to the sending module and determine which frame to be sent in the next handshake by the received frame.

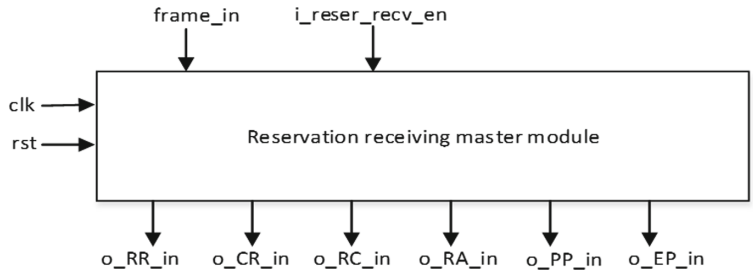

Fig. 10. Input and output signals of reservation receiving master module

In order to realize it, firstly set up the interface. Then set the frame signal received as the input. Every time one frame comes, the first 32bit of the incoming frame, namely the type code of the frame, is judged. And the receiving module of the corresponding frame is called according to the different type code and the receiving of the frame is carried out. After each frame is successfully received, output completion enable signals.

3.3.5 Reservation Master Module

The total top-level module of reservation sending and the Reservation receiving master module are called in this module to realize that after a node receives a frame, the sending master sends the next reservation frame according to the

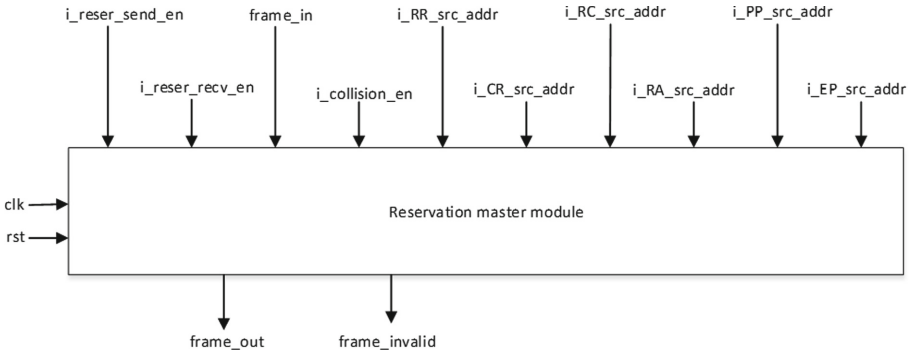

Fig. 11. Input and output signals of reservation master module

signals generated by the receiving master to realize the connection of sending and receiving.

Figure 11 shows the input and output of the reservation master module. Input clock signal (clk) and reset signal (rst) to generate the clock. Input the overall enable signal of reservation sending (i_reser_send_en) to determine whether to call the overall top-level module of reservation sending. Input the total enable signal of reservation receiving (i_reser_recv_en) to determine whether to call the total module of reservation receiving. Input the source addresses (i_RR_src_addr, i_CR_src_addr, i_RC_src_addr, i_RA_src_addr, i_PP_src_addr, i_EP_src_addr). Input the enable signal of contend (i_contention_en) to simulate the occurrence of contend, that is, when the enable signal of contend is high, contention happens and send CR frame. Input the received frame signal (frame_in) and call each receiving sub-module to receive the frame. Print out the frame signal that was sent (frame_out), and print out the frame's valid field (frame_invalid).

In order to realize it, set up the interface. Call the total top-level module of reservation sending and the Reservation receiving master module. After receiving each frame output by the Reservation receiving master module, completion enable signals, such as o_RR_in, etc., are connected to the input port of the total top-level module of reservation sending with several intermediate linear variables to connect the two modules. Output the frames sent by the overall sending module.

4 Performance Evaluation

4.1 Single Node Validation

4.1.1 Total Top-Level Module of Reservation Sending Module

Enable signal will be generated by three conditions – the clock module output enable signals, the algorithm module calculate probability to decided when to send the RR, the judging of the type of the received frame to decide the next

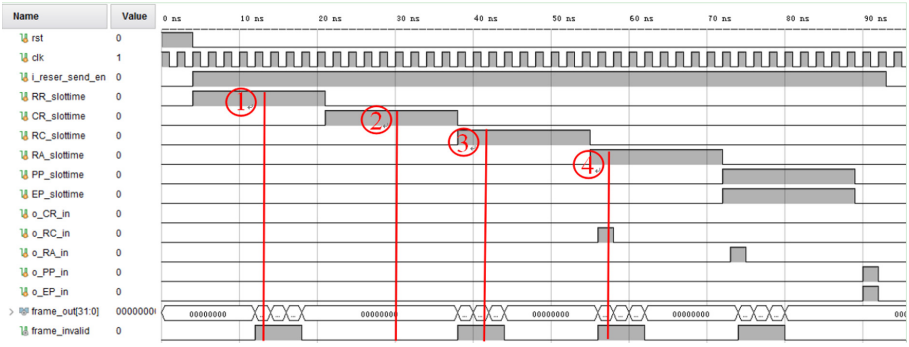

Fig. 12. Waveform of non-contention situation

handshake step and send certain frame. Integrated them into the overall module, input node source address, pull contention enable signal up, simulating the situation of contention, check the waveform. Pull down the contention enable signal to simulate the non-contention situation, and check the output waveform.

The results are shown as below.

- Non-contention situation in Fig. 12
  The result of this situation displays as Fig. 12 and the explanation of label 1 to 4 is shown as below.
  The label 1 shows that it send the corresponding frame in the corresponding slot. For example, send the RR frame in the RR slot. The label 2 shows that no CR frame is sent in non-contention situation. The label 3 shows that if no CR frame is received, the RC frame is sent in the RC slot. The label 4 shows that After receiving the RC frame, the RA frame is sent in the RA slot.
- Contention situation in Fig. 13

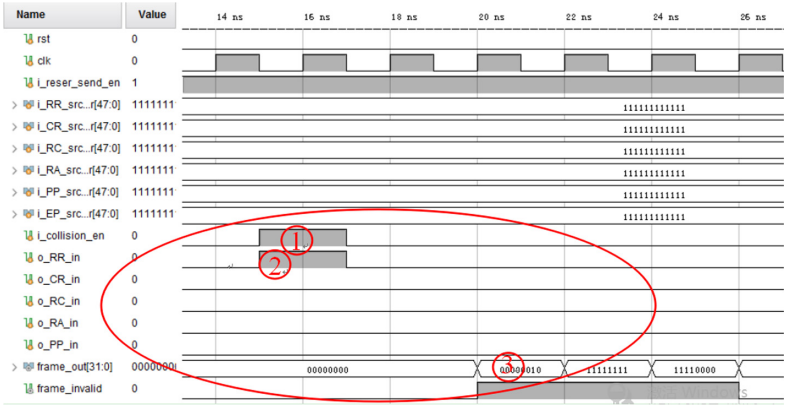

Fig. 13. Waveform of contention situation

The result of this situation displays as Fig. 13 and the explanation of the circled part shows that on the premise that the contention occurs(1) and the RR frame has been received(2), the CR frame is sent(3).

4.1.2 Reservation Receiving Master Module

In order to make a implantation, firstly pull up the reservation receiving enable signal. Then input the received signal frame (frame_in), automatically recognize the frame type to receive, and view the generated waveform.

The result is as below in Fig. 14.

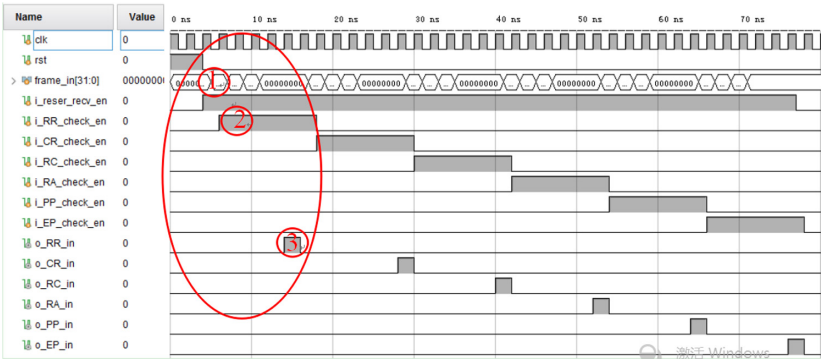

Fig. 14. Waveform of Reservation receiving master module

The result of this situation displays as Fig. 14 and the explanation of the circled part shows that for the incoming frame, firstly judge the frame type(1). Then call the corresponding frame receiving sub module to receive(2). After receiving, output the receiving completion enable signal (e.g. o_RR_in(3)).

4.2 Two-Node Validation

In order to make a implantation, firstly set up two different node source addresses. Then call two reservation master modules to simulate the state of mutual dispatching between the two nodes.

The expected result is that of the invoked two reservation master modules, frame_out of one output is sent to the other module as its input. Both sub-modules work well and can receive and send frame signals normally.

The result is shown as below in Fig. 15.

The result of this situation displays as Fig. 15 and the explanation of the circled part shows that the external node inputs the signal value to node 1 (1). Then the outputs of node 1 are sent to node 2 as the inputs of node 2 (2). And the outputs of node 2 is the outputs of the total module(3) and the display is normal.

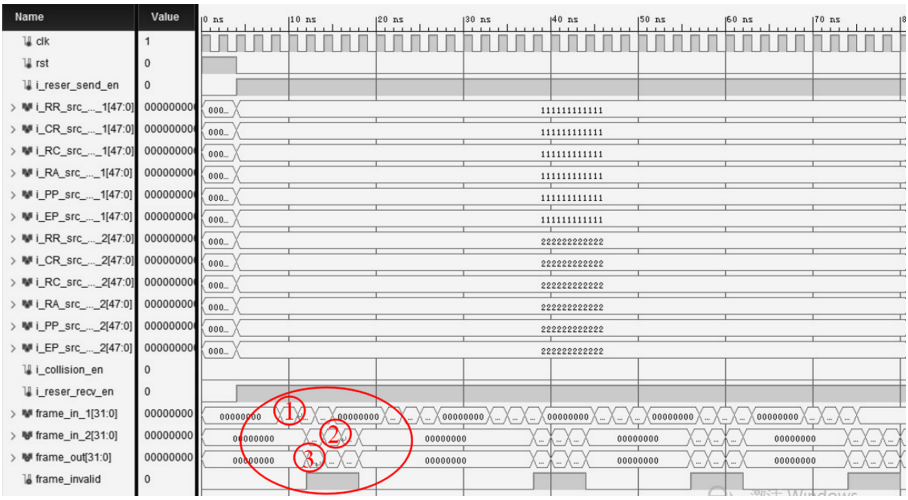

Fig. 15. Waveform of two nodes

5 Conclusion and Future Works

In this paper, the FPGA implementation scheme of FPRP protocol is presented. The design and simulation ideas of algorithm module, clock counting module and reservation module are detailed. And it is proved that this scheme is feasible.

In the simulation practice, we design a new signal to realize a contention. That is because that the principle shows that when multiple RR frames arrive, CR frames are sent to indicate contend. However, in the two-node validation, we receive at most one RR frame, and there will never be a contend. In this case, a contention signal is directly set in the simulation test file to indicate the occurrence of the contention. When the contention occurs, CR frame is sent under the simulate the occurrence of the contention. However, in fact, it cannot be realized. In the network, only when there are multiple RR frames coming or the node cannot parse out RR frames or fails to receive them successfully, the contend will be considered.

In addition, we would like to list the following points for the future work outlook.

In the following work, the result of fractional generation of algorithm module can be improved, and the remaining unrealized modules such as reservation result generation module and data transmission module can be fully realized. Besides, for some reasons, this paper does not calculate the maximum clock frequency and other parameters of the program, and hope to improve it in the future research work.

Acknowledgment. This work was supported in part by the National Natural Science Foundations of CHINA (Grant No. 61771392, No. 61771390, No. 61871322 and No. 61501373), and Science and Technology on Avionics Integration Laboratory and the Aeronautical Science Foundation of China (Grant No. 201955053002, No. 20185553035).

References

1. Cao, B., et al.: How does CSMA/CA affect the performance and security in wireless blockchain networks. IEEE Trans. Ind. Inform. **16**(6), 4270–4280 (2019)
2. El Korbi, I., Saidane, L.A.: Performance evaluation of unslotted CSMA/CA for wireless sensor networks: energy consumption analysis and cross layer routing. Int. J. Comput. Netw. Inf. Secur. **9**(6), 1–12 (2017)
3. Mawlawi, B., et al.: Analysis of frequency channel division strategy for CSMA/CA with RTS/CTS mechanism. In: International Conference on Sensing Technology, p. 5 (2014)
4. Pejoski, S., Hadzi-Velkov, Z.: Slotted ALOHA wireless networks with RF energy harvesting in Nakagami-m fading. Ad Hoc Networks **107**, 102235 (2020)
5. Baccelli, F., Blaszczyszyn, B., Muhlethaler, P.: An Aloha protocol for multihop mobile wireless networks. IEEE Trans. Inf. Theory **52**(2), 421–436 (2006)
6. Ishii, K.: Modified slotted ALOHA protocol for average consensus problem. In: IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), pp. 1–5. IEEE (2017)
7. Mustafa, M.M., Parthasarathy, V.: A design and implementation of polling TDMA with a comparative analysis with time division multiple access for sporting application. Wirel. Netw. **26**(3), 1897–1904 (2018). https://doi.org/10.1007/s11276-018-1879-9
8. Yang, Q., Zhuang, Y., Shi, J.: An improved contention access mechanism for FPRP to increase throughput. Etri J. **35**(1), 58–68 (2013)
9. Zhu, C., Corson, M.S.: A five-phase reservation protocol (FPRP) for mobile ad hoc networks. Wirel. Netw. **7**(4), 371–384 (2001)
10. Anuradha, B., Dutta, P.: Fuzzy-controlled power-aware routing protocol (FPRP) for mobile ad hoc networks. Int. J. Comput. Appl. **11**(7), 39–43 (2010)
11. Zhenghua, L., et al.: Improvement research on FPRP for MAC of wireless Ad Hoc network. J. Chengdu Univ. p. 2 (2015)
12. Bhatia, I.S., Randhawa, D.K.K.: Verilog-A modeling of a silicene-based p-n junction logic device: simulation and applications. J. Comput. Electr. **19**(1), 387–395 (2020)
13. Ciletti, M.D.: Advanced Digital Design with the Verilog HDL. Advanced digital design with the Verilog HDL =. Pub, House of Electronics Industry (2004)
14. Bhat, N.S.: Design and implementation of IEEE 802.15. 4 Mac protocol on FPGA. arXiv preprint arXiv:1203.2167 (2012)
15. Wang, G., Kang, G.X., Wang, H.: Design and FPGA implementation of MAC-PHY interface based on PCI express for next-generation WLANs. In: Wireless Communications, Networking and Mobile Computing (WiCOM), 2012 8th International Conference on, pp. 1–4. IEEE (2012)