A Low-Latency Wireless Network for Cloud-Based Robot Control

Seyed Ali Hassani(✉) and Sofie Pollin

Department of Electrical Engineering, KU Leuven, Heverlee 3001, Belgium
{seyedali.hassani,sofie.pollin}@esat.kuleuven.be
https://www.esat.kuleuven.be/telemic/research/NetworkedSystems

Abstract. We demonstrate a reliable network for robot remote control in which a cross-layer PHY-MAC architecture is exploited to establish a low-latency and time-critical data transmission. In our demo, three reverse pendulum robots share the spectrum to communicate their sensory data to a processing unit which can instantly generate and transmit appropriate commands to maintain the robots' balance.

To this end, we upgrade CLAWS (Cross-Layer Adaptable Wireless System) with a two-layer MAC platform which accelerates and facilities interrupt handling. To grant the network operational reliability, we elaborately coupled the FPGA-based IEEE 802.15.4 PHY in the CLAWS architecture with a set of hardware blocks that play the role of the low-level MAC. CLAWS also offers a run-time programmable module in which we deploy the high-level functionalities of the MAC protocol. Jointly with the implemented bi-layer MAC structure, we demonstrate how the CLAWS' flexibility allows either standard compliant or ad-hoc network prototyping to establish a reliable cloud-based robot remote control.

Keywords: Software-defined radio · Cross-layer architecture
Cloud-based processing

1 Introduction

Software-defined radio (SDR) enables reconfigurability in wireless communication systems, and as such it has played a key role in technology development in the last decade. In addition to programmability, the latest communication technologies (e.g., 5G, wireless edge computing) have to provide a high level of performance to satisfy new and emerging applications such as cloud-based processing.

Today's challenge on SDR is to bring together both flexibility and high-performance functionality in one platform. To overcome this challenge, the platform has to provide a wide variety of capabilities in the hardware-reliant network layers. The PHY and MAC, for instance, differ in their data-flow and control-flow operations: PHY works with simple pipeline stages to deal with heavy data-flow

I. Moerman et al. (Eds.): CROWNCOM 2018, LNICST 261, pp. 46–51, 2019.
https://doi.org/10.1007/978-3-030-05490-8_5

and is dominated by processing latency, whereas the MAC functionality needs complex control and complicated branch instructions to perform event handling and control functions [1].

1.1 The Programmable CLAWS Architecture for Reliable Communication

Recently, large FPGA-based SDRs have been introduced to fill the gap between latency requirements and reconfigurability. CLAWS [3] is one of the presented platforms which offers an IEEE 802.15.4 compliant transceiver including the physical and MAC layers respectively implemented by LabVIEW FPGA and C. The PHY is split into several functional adjustable blocks enabling granular modification and on-the-fly changes without recompiling of the FPGA code [5]. The MAC is also fully deployed in a Xilinx MicroBlaze softcore [4] to allow run-time reprogramming and facilitate MAC development.

The demo presented in [5] demonstrates the CLAWS PHY and network layers performance in a standard compliance network. The main objective of our demo is to illustrate how one can combine the CLAWS' pieces to prototype a high-reliable wireless system for time-critical messaging.

A crucial prerequisite for Ultra-Reliable Low Latency Communication (URLLC) is to deliver a packet within a certain time. One can define the reliability as the probability that the latency does not exceed a predescribed deadline [2]. Such a deadline might be dictated by the application itself or determined by the standard, e.g., the fixed time interval that within the receiver has to acknowledge a data receipt. A running MAC on CPU, may not fulfill this obligation as it can not handle multiple and successive interrupts properly.

In the case of the cloud-based wireless network, it can be shown that occurrence of consecutive interrupts, such as generating multiple succeeding Immediate Acknowledge (Imm-Ack) packets in the central node, affects the CPU performance negatively as it has to reinitiate the interrupted task each time. In full-duplex or multi-channel MAC protocols, the capability of handling events becomes even more essential. For instance, in a multi-channel vehicular network, the MAC CPU is more likely to be suspended by a high-priority packet at the control channel while it has to react to the service channels concurrently.

To mitigate this problem, we combine the CLAWS' CPU with a scalable set of low-level modules which enables agile interrupt handling while does not degrade the CLAWS flexibility. Then we employ the enhanced platform to establish a cloud-based system that controls three balancing robots. The resulting timing enhancement is presented.

2 Two-Level MAC Structure for Improving Interrupt Handling Capability

The CLAWS platform utilizes a MicroBlaze soft processor to run the MAC protocol. In addition to run-time reprogramability of this CPU, it significantly helps implementation of multi-stage procedures and sequential algorithms. In fact, some basic tasks in the MAC layer do not need deep flexibility and hence, can be implemented in hardware more efficiently.

Figure 1 depicts our proposed two-layer MAC structure in the CLAWS framework. This model benefits from a set of hardware level blocks which are interfaced carefully with the high-level flexible MAC running in the MicroBlaze. Operating close to the PHY layer, the low-level MAC can deliver a set of basic and interrupt-based tasks of the MAC layer and accelerate interrupt handling as well as facilitate the development of complicated networking protocols and retransmission algorithms.

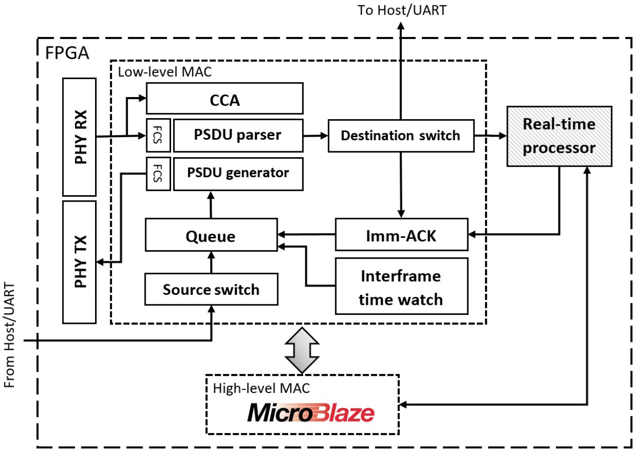

Fig. 1. High/Low level MAC in the CLAWS architecture.

2.1 High-Level MAC

The CLAWS cross-layer architecture enables to deploy the high-level MAC on MicroBlaze. In the enhanced platform, this CPU can initiate packet transmission process, run retransmission algorithm, drive the backoff timer to estimate the backoff time, trigger Clear Channel Assessment (CCA) module, configure the payload sources and destination switches (e.g., UART, host PC), etc.

2.2 Low-Level MAC

The hardware realized low-level MAC in the upgraded CLAWS reduces the workload of the CPU as it can perform somewhat non-sequential straightforward jobs of the MAC protocol and react to interrupts instantly. For instance, the CPU can configure the destination switch in the low-level MAC to pass a packet to the host, the UART interface, the MicroBlaze itself or any other hardware-implemented processor. This mechanism also allows spontaneous Imm-ACK packet transmission.

As shown in Fig. 1, these blocks are elaborately coupled with the related sections in PHY on one side and interfaced with the high-level MAC one the other side. The following functionalities can be accomplished by the presented low-level MAC; generating the PHY Service Data Unit (PSDU) and Imm-ACK packet, parsing the PSDU, controlling the packet queue, and measuring the interframe spacing times. This set of components is implemented by LabVIEW FPGA, running at 150 MHz and needs slight flexibility which is guaranteed by the CPU.

3 Demo Set-Up Overview

We show a star-topology network in which four SDRs run the upgraded CLAWS architecture. Each SDR has an NI USRP RIO [6] which is equipped with a Xilinx Kintex 7 FPGA. This SDR can transmit from 400 MHz to 4.4 GHz and provides a 120 MHz IQ sample rate at IF stage. We use the integrated PCIe interface to monitor the FPGA's internal signals and the results.

Figure 2 (Left) illustrates the structure of the test set-up. Each of the three SDRs is interfaced with a reverse pendulum robot which requires updating its balance status every 7 ms. The robots send their sensory data through the UART interface to the SDRs. The SDRs then transmit this information to the fourth node which plays the role of cloud processor and can produce appropriate commands to drive the robot's DC motors in such a way that they maintain balance.

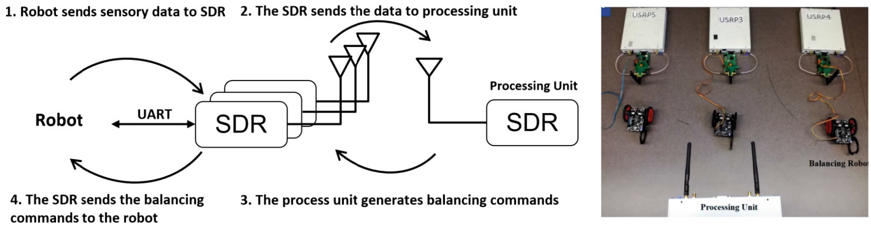
1. Robot sends sensory data to SDR
2. The SDR sends the data to processing unit
Robot
UART
SDR
Processing Unit
SDR
4. The SDR sends the balancing commands to the robot
3. The process unit generates balancing commands
USRP5
USRP3
USRP4
Balancing Robo
Processing Unit

Fig. 2. (Left): The structure of the cloud-based wireless network. (Right): The test set-up including four SDRs and three balancing robots.

The payload carries 4 bytes of sensory data and jointly with the MAC's header and footer it forms a 12-Byte PSDU. In order to share the spectrum fairly, the nodes obey the CSMA/CA MAC scheme to transmit their sensory data. The processor in our set-up can either attach the balancing commands to the Imm-ACK in the low-level MAC or generate a new packet in the CPU. Due to the CCA procedure and the backoff mechanism in CSMA/CA scheme, the latter is accompanied by a non-deterministic latency. The processing unit can also acquire a control command, e.g., moving forward and backward, through an Ethernet connection and broadcast it to all robots. A video demonstrating this set-up is available in [7].

4 Experimental Results

To compare the performance of the presented two-layer MAC and the fully CPU-deployed MAC, we ran both structures at 150 MHz and interrupted them to produce 12-Byte Imm-ACK. Depending on its working state, the single-layer MAC needs 10 µs to generate and place the Imm-ACK packet in the queue. However, this time may exceed up to 16 µs depending on CPU's working status. Whereas, the bi-layer MAC behaves deterministically as it generates and inserts the packet, including the balancing command, into the queue within 0.41 µs.

In order to compare our ad-hoc command transmission scheme with a standard complaint MAC, we measured the time that the CPU needs to acquire the sensory data from the PHY's FIFO, generate and place the Imm-ACK, trigger the CCA module and apply the CMSA's exponential backoff time to transmit the commands. Depending on the network traffic load, this interval may vary from 156 µs to several milliseconds in a non-deterministic fashion.

5 Conclusion

In this paper, we illustrated how the flexibility of CLAWS architecture helps to prototype a standard compliant or an ad-hoc network which effectively satisfies the obligations of low-latency and time-critical messaging.

Using the CLAWS architecture, it is shown that the developed two-layer MAC model enables robust communication while it preserves the CLAWS' run-time configuration capability. Due to its efficient implementation, this low-level MAC can take up the time-sensitive workload of the MAC layer and enable reliable ad-hoc network prototyping.

Besides, our low-level MAC fits in <%2 of the Kintex7 FPGA while the MicroBlaze occupies %8 of the total logic resources. Therefore, a combination of multiple low-level MACs on top of one CPU module can form a practical and efficient solution for multi-channel and full-duplex MAC protocols as it delivers adequate processing power and multiple interrupt handling ability.

Acknowledgement. This work was partially funded by the European Union's Horizon 2020 under grant agreement no. 732174 (ORCA project).

References

1. Wu, H., et al.: The tick programmable low-latency SDR system. In: MobiCom, Snowbird, Utah, USA, October 2017
2. Popovski, P., et al.: Ultra-reliable low-latency communication (URLLC): principles and building blocks. CoRR, abs/1708.07862 (2017)
3. Van den Bergh, B., Vermeulen, T., Verhelst, M., Pollin, S.: CLAWS: cross-layer adaptable wireless system enabling full cross-layer experimentation on real-time software-defined 802.15. 4. EURASIP J. Wirel. Commun. Netw. **2014**(1), 187 (2014)
4. MicroBlaze soft processor. http://www.xilinx.com/tools/microblaze.htm.. Accessed 04 Apr 2018

5. Vermeulen, T., Van den Bergh, B., Pollin, S.: Demo: a software defined radio platform for rapid cross-layer prototyping, pp. 1–4 (2015)
6. Ni usrp-2942r - national instruments. http://sine.ni.com/nips/cds/view/p/lang/nl/nid/212434
7. https://www.youtube.com/watch?v=hP7aXoF9FVU