

Radio Hardware Virtualization for Software-Defined Wireless Networks

Felipe A. P. de Figueiredo¹ · Xianjun Jiao¹ · Wei Liu¹ · Ingrid Moerman¹

Published online: 26 March 2018
© The Author(s) 2018

Abstract Software-Defined Network (SDN) is a promising architecture for next generation Internet. SDN can achieve Network Function Virtualization much more efficiently than conventional architectures by splitting the data and control planes. Though SDN emerged first in wired network, its wireless counterpart Software-Defined Wireless Network (SDWN) also attracted an increasing amount of interest in the recent years. Wireless networks have some distinct characteristics compared to the wired networks due to the wireless channel dynamics. Therefore, network controllers present some extra degrees of freedom, such as taking measurements against interference and noise, or adapting channels according to the radio spectrum occupation. These specific characteristics bring about more challenges to wireless SDNs. Currently, SDWN implementations are mainly using customized firmware, such as OpenWRT, running on an embedded application processor in commercial WiFi chips, and restricted to layers above lower Media Access Control. This limitation comes from the fact that radio hardware usually require specific drivers, which have a proprietary implementation by various chipset vendors. Hence, it is difficult, if not impossible, to achieve virtualization on the radio hardware. However, this status has been changing as Software-Defined Radio (SDR) systems open up the entire radio communication stack to radio hobbyists and researchers. The bridge between SDR and SDN will make it possible to bring the softwarization and virtualization of wireless networks down to the physical layer, which will unlock the full potential of SDWN. This paper investigates the necessity and feasibility of extending the virtualization of wireless networks towards the radio hardware. A SDR architecture is presented for radio hardware virtualization in order to facilitate SDWN design and experimentation. We do believe that by adopting the virtualization-oriented hardware accelerator design presented here, an all-layer end-to-end high performance SDWN can be achieved.

✉ Ingrid Moerman
Ingrid.Moerman@UGent.be

Felipe A. P. de Figueiredo
felipe.pereira@ugent.be

¹ Department of Information Technology, Ghent University, Ghent, Belgium

Keywords SDR · SDN · Virtualization · SDWN · PHY · FPGA

1 Introduction

SDN is a promising concept at networking level, it decouples the network control and data forwarding functions, allows directly programmable network control, and provides diverse network services to a variety of applications. Before the introduction of the SDN concept, there were an increasing amount of labels/headers appended to packets, to support various kinds of protocols for different services on the Internet, which greatly increased the processing burden on the edge routers and switches. SDN solves the issue by using a disruptive design that separates data and control planes: routers and switches become dumb devices, which are only responsible for forwarding data according to the controller's instructions. The controller applies slow-varying configurations on the data forwarding devices, in order to slice/allocate the network resources to different types of services during runtime. Such an approach allows virtualizing a single physical network into multiple and heterogeneous logical network domains, each domain serving a certain category of traffic flow in the most appropriate way. The SDN approach is very encouraging, but has been basically designed for wired networks and mainly involves the higher layers of the protocol stack, e.g. layer 4 to 7 of the Open Systems Interconnection (OSI) model. It is primarily providing transport capacity and service differentiation up to the edge router of wired networks.

Wireless networks benefiting from the flexibility offered by virtualization in SDN are known as SDWN, however, it must be emphasized that the wireless medium, i.e., 'radio spectrum', has very different properties than the ones exhibited by wired networks. In wired networks, a port of a switch or router is always connected to fiber or UTP cables. Multiple ports are equivalent to multiple isolated non-interfering communication links with constant data rate. As a result, Ethernet is ubiquitous in the wired network. On the other hand, the wireless medium is not isolated but shared. In wireless networks, there is interference when multiple links are simultaneously running in the same or adjacent radio spectrum bands. In addition, the data rate of a wireless link is dynamic, due to the variations in distance (mobility), channel conditions (e.g., heavily shaded or LOS), unpredicted interference from other co-located wireless technologies or radiating devices (e.g. microwave ovens). Hence, unlike wired networks that are dominated by Ethernet or optical links that have a deterministic capacity, wireless networks are non-deterministic and established upon many heterogeneous PHY and MAC layer standards, with each standard serving a different type of traffic flow. For example, LoRa [1] and SigFox [2] are used for long range low rate sensor data collection, while Zigbee is designed for short-mid range low-mid rate sensor networks [3]; Bluetooth is known for short range accessory communications [4]; WiFi is devised for short range high throughput applications [5]; 2G/3G/4G mobile networks serve mid-high throughput terminals over a mid-long range [6, 7]; etc.

From an end-to-end communication point of view, the different traffic flows are characterized by different QoS requirements, such as for example, latency and throughput, which will be addressed later in this paper. The main idea here is that, in wireless communications networks, one technology can hardly meet all requirements and can not give firm guarantees to QoS requirements. The lack of coordination and interaction among all the wireless networks standards, can jeopardize the overall performance of a network. The versatility in wireless network's PHY and link layers is somewhat comparable to the era before SDN's appearance in the wired network, where various headers are appended into packets to support different services. Therefore, the logical next step in wireless networks

is the achievement of runtime configuration across the diverse wireless standards by applying the SDN concept. This implies two requirements: (i) the lower layer radio stack needs to be more flexible in order to support runtime configuration and virtualization; (ii) the conventional SDN paradigm needs to be extended to counteract the uncertainties in wireless networks, by taking measurements in order to optimize the radio resource allocations (e.g., spectrum, time, space).

In the remainder of this paper, we first present a comprehensive view on the status of various efforts towards SDWN in Sect. 2; next an end-to-end view of the SDN-enabled wireless network from the ORCA project [8] is given in Sect. 3; then Sect. 4 presents a novel architecture for radio hardware virtualization to support the ORCA vision, followed by initial experimental validations; finally we conclude this work in Sect. 5.

2 State of the Art Analysis

This section begins with the recent progress in the field of flexible and generic physical layer radio implementations, and the trend towards more dynamic spectrum allocation schemes; then we move on to two representative ways of real-life SDWN practices.

2.1 Evolution Towards Flexible PHY

At the radio level, we have observed the emergence of SDR. An SDR is a radio communication system where transceiver components that are typically implemented on Application-Specific Integrated Circuit (ASIC), e.g., digital mixers, filters, equalizers, modulators/demodulators, multiple antenna techniques etc., are instead implemented on software on a host computer or on an embedded system equipped with programmable hardware like Application-Specific Instruction set Processor (ASIP) or Field-Programmable Gate Array (FPGA). The concept behind SDR is very encouraging for the development of state-of-the-art physical layer (PHY) functionalities, because software programming allows faster development cycles. Therefore, many advanced and flexible physical layer techniques are available on SDR platforms, including Massive MIMO, full duplex, mmWave, and various novel waveforms.

The main problem with software implementation is the slower sequential execution of algorithms, even when multi-core or many-core Central Programming Unit (CPU) platforms or Graphics Processing Units (GPUs) are used, in contrast to a very fast execution and a very high degree of parallelization achieved with implementations on ASIC, ASIP or FPGA. For this reason, SDR development has so far mostly been limited to non real-time physical layer development, as software implementations do not always offer the fast execution times that are required for true networking experimentations, e.g., experiments requiring acknowledgment of MAC frames within a few microseconds. Recently, we have been observing limited yet increasing efforts to code more and more transceiver functionality on hardware, e.g., FPGA, trading off software flexibility for faster execution times, at the cost of higher design time. In both cases, SDR implementations are much more open, which gives potential to support functionalities such as network virtualization on the lower communication stack.

2.2 Evolution Towards Generic PHY

Current wireless networks are composed of many different standards below the transport layer, which are necessary due to the versatility in the wireless medium and the traffic demands. In recent years, we have been noticing that individual physical layer standards are evolving towards each other to become more generic/homogeneous. For instance, Narrow Band Internet of Things (NB-IoT) has been developed as a subset of LTE to support low power and long range IoT applications, which has similar capabilities as LoRA [9]. In addition, LTE also supports smaller Transmission Time Intervals (TTI), which are complementary to the standard 1 ms TTI for serving low-latency applications [10]. Conventional WiFi standards don't support flexible sub-carrier allocation, meaning that the spectrum resource cannot be sliced besides the selection of channels. The latest WiFi standard 802.11ax is supposed to support the sub-carrier allocation feature, which is comparable to the resource blocks in LTE [11]. According to the standardization of 5G New Radio (NR), "scalable OFDM numerology" is proposed to support sub-carrier spacing ranging from 15 kHz (same as LTE) to 240 kHz (close to WiFi 802.11a, 312.5 kHz), to enable the operation in a much wider radio spectrum and coverage areas [11, 12]. Without doubt, the evolution towards a more generic/homogeneous physical layer in wireless network will make the support of SDWN more convenient.

2.3 Evolution Towards Dynamic Spectrum Allocation

As stated previously, the radio communication link is not isolated by nature, but it could be achieved by enforcing the usage of a chunk of the spectrum only for a specific application, this is referred to as the licensed spectrum usage. This is the simplest approach, however, it is not efficient, since the static allocation causes waste of spectrum when there is no traffic demand from a given application. Some efforts are already being pushed to increase the utilization rate of the licensed spectrum, by allowing secondary usage of the spectrum without sacrificing the communication quality of the incumbents, e.g., TV white space [13] or spectrum sharing in radar bands [14]. The alternative to the simplest approach is the unlicensed spectrum access, where technologies share the medium with equal privileges. This approach is best represented by the current situation in ISM bands, where several technologies compete for spectrum access. However, the chances of over-the-air collisions increase when there is no coordination among devices/technologies, which is likely to trigger extended back off periods, leading to poor spectrum efficiency and lower QoS observed in the end-to-end communication links. The control and management functionalities in SDN could be borrowed to improve coordination between wireless network entities for spectrum usage. In addition to improving the efficiency of allocated radio spectrum, huge bandwidth can be harvested by extending wireless signal spectrum to mmWave band [15], such as 5G [11, 12] and 802.11ad [16].

Generally speaking, there is a trend going on for dynamic spectrum allocation, and many of the pioneer works in this area are based on SDR, focusing on its ability to rapidly adapt the operational parameters in order to achieve the optimal performance [17].

2.4 SDWN Experiments on Commercial WiFi Chipset

Some SDN experiments have already been carried out in the wireless network domain. An off-the-shelf WiFi Access Point (AP) device can be split into multiple virtual APs on

demand by flashing customized firmware, e.g., OpenWRT, [18]. In this way, the bandwidth supported by the physical AP is sliced and can be allocated to different users or services. Seamless mobility among physical APs can be achieved by managing the virtual APs across multiple physical APs [19]. These functionalities are implemented in layers above upper-MAC. It means that multiple virtual entities share a single physical layer and lower-MAC layer by Time Division Multiplexing (TDM).

Although these efforts are important progresses towards SDWN, there are a number of significant limitations caused by the lack of SDN oriented physical and lower-MAC layers. As the virtual APs are created in a pure software manner, the additional overhead caused by running multiple link services, e.g., authentication status maintenance, context switching, etc., upon a weak processor causes a severe performance degradation in throughput [20, 21]. Furthermore, extra latency and jitter are introduced for a specific service/user, due to different virtual entities accessing the single physical link through time division multiplexing, meaning that when one entity is accessing the physical link, the rest of the users/services are waiting for their turn.

2.5 SDWN Experiments on SDR and Cloud Computing

Essentially, SDR aims to implement radio transceiver functionalities, which are traditionally realized in hardware, on software domain. It is a promising candidate for physical layer implementation of SDWN, as demonstrated in use cases such as Cloud based Radio Access Network or Centralized Radio Access Network (C-RAN) [22]. In C-RAN systems, a Remote Radio Head (RRH) only performs conversion between the digitized baseband signal and the analog RF signal. The Baseband Unit (BBU) is implemented in the servers on the cloud, which in turn performs all the necessary processing tasks of the physical layer. Empowered by the rather mature virtualization technologies in computer science domain, the software BBU can be created, allocated, migrated and deleted on the fly. The BBU in the cloud might achieve comparable throughput as their hardware counterparts, however, the performance in terms of latency is generally much worse. Fortunately, existing mobile network standards can tolerate a relatively large latency. For instance, LTE has a 1 ms TTI and 4 ms Hybrid Automatic Repeat Request (HARQ) feedback delay [7].

Although centralized BBU functions work well for some applications, it is difficult to serve applications with tight latency requirements. For example, self-driving functions relaying on vehicle-to-vehicle communications require the base station to be located as close as possible to the vehicle in order to minimize the reaction time of the vehicle. For this type of use case, Mobile Edge Cloud (MEC) is a more suitable architecture than C-RAN. However, even MEC cannot softwarize the wireless standards with extremely low latency requirement, such as WiFi. WiFi's low MAC requires a node to acknowledge a successfully received packet within the duration of Short Inter Frame Space (SIFS) [5]. SIFS ranges from 16 μ s down to 3 μ s, depending on the specific variant of WiFi standards (IEEE802.11a/b/g/n/ad,etc). It is evident that this requirement cannot be met with the BBU entirely implemented in software. The lower-MAC and hardware coded physical layer need therefore to be tightly integrated in order to fulfill the necessary level of latency requirement.

3 ORCA's Vision

The overall vision of the H2020 ORCA project is to drive end-to-end wireless network innovation by bridging real-time SDR and SDN. The project aims at exploiting the maximum flexibility at radio, medium access and network levels, in order to meet a very diverse application requirements [23].

This vision is illustrated in Fig. 1 and is further explained step-by-step using factory-of-the-future as the driving scenario. The manufacturing industry is one of the most demanding verticals with respect to ultra-low latencies, ultra-high reliability, ultra-high data rates, ultra-high availability, reliable indoor coverage in harsh environments (with a lot of metal structures) as well as energy-efficient and ultra-low communication costs for produced and connected goods. At the top of Fig. 1 (beige color) different traffic classes can be observed corresponding to different application requirements. For the manufacturing scenario, a non-exhaustive list of traffic classes (TCs), can be identified. These TCs were inspired by [24, 25].

TC1 Time-critical sensor/actuator control loop: bidirectional communication, low data rate (in the order of kbps), stringent timing requirements (below 1 ms cycle time, order 100 μ s response time, below 1 μ s jitter), ultra-high reliability (99.999999 %), indoor, very short range (in the order of 10 m). Examples: motion control in printing machines, textile weaving machines, paper mills.

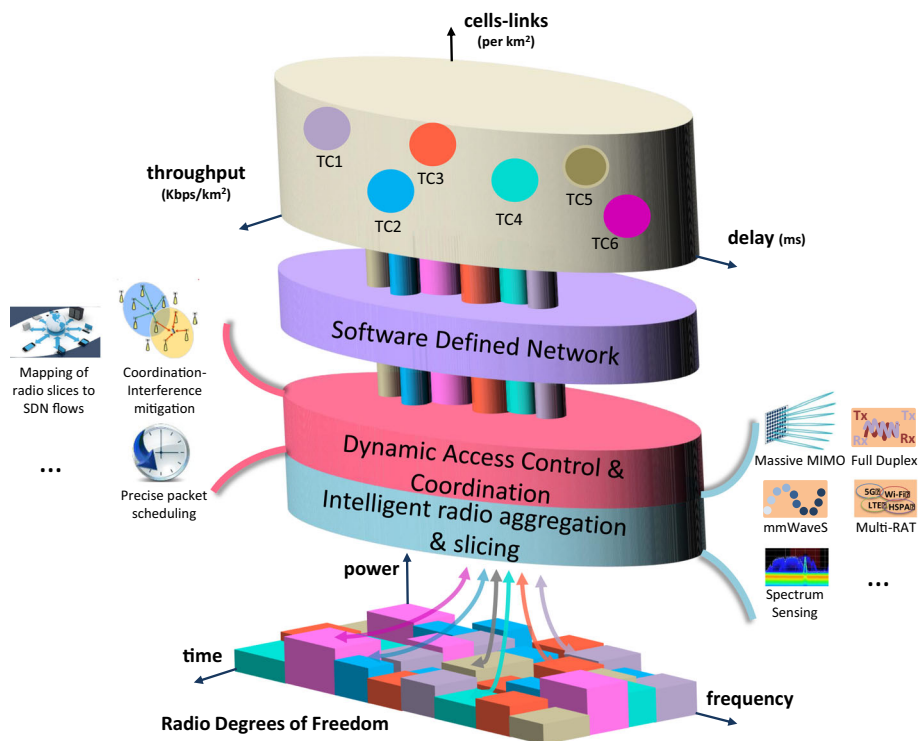


Fig. 1 Network innovation driven by ORCA—the end-to-end view on SDN enabled wireless network [23]. (Color figure online)

TC2 Time-critical vision-controlled processes: bidirectional asymmetric communication ultra-high data rate (up to 10 Gbps), low latency (below 0.5 ms), high reliability (99.99999 %), indoor, short range (10–100 m). Example: vision-controlled robot arms, vision-controlled quality inspection, wearables and augmented reality on the shop floor.

TC3 Low-latency continuous medium throughput: point-to-point and point-to-multi-point, moderate data rate (in the order of 10–100 kbps), low latency and jitter (both below 10 ms), ubiquitous coverage and high availability (indoor + on-site outdoor), mobility support, large autonomy. Example: voice communications between workers with headsets in the manufacturing hall.

TC4 Correlated data capturing: moderate data rates (in the order of kbps up to 100 Mbps), moderate latency (10–100 ms), ultra-high time synchronization accuracy (below 100 ns), and moderate reliability (99.999 %), ubiquitous indoor coverage. Example: capturing of time-correlated sensor data on the shop floor to facilitate virtualized design processes that integrate simulator data with real-life data sensed during production.

TC5 Non time-critical in-factory communication: moderate data rates (in the order of kbps up to 100 Mbps), latency in the order of 100 ms (limited by human response times), moderate reliability (99.999 %) ubiquitous coverage and high availability (indoor + on-site outdoor), mobility support. Examples: interactions between humans and machines or robots, localization of assets and goods.

TC6 Bursty traffic: non-time critical (very large latencies allowed), large data volumes (in the order of MB up to 100 GB). Examples: sporadic software/firmware updates of machines, temporary reconfiguration of machines.

TC7 Best effort: low priority, no firm guarantees on data rates or latency, minimal shared capacity, ubiquitous coverage (indoor–outdoor). Example: typical Internet application (email, web surfing).

The current radio technologies lack capabilities with respect to wireless performance, management of heterogeneous devices, technology interoperability and application (traffic) demands. Flexible and seamless connectivity across different Radio Access Technologies (RATs) will be required in order to instantaneously adapt the capacity and mobility needs to changing environments and application demands. A first approach to deal with such very diverse traffic demands would be the application of SDN techniques. Instead of employing one physical network infrastructure to deal with all the different traffic classes, applying complex traffic algorithms or QoS scheduling mechanisms, the network infrastructure can be virtualized into separate and independent network infrastructures, applying the most appropriate protocols and resource sharing mechanisms to deal with a specific traffic class. This approach is called network slicing or vertical slicing. This is illustrated by the vertical, colored pipes in Fig. 1. Each pipe in the figure represents a single network slice, architected and optimized for the specific requirements of the applications supported by its traffic class. For the manufacturing scenario described above, this results in 7 different pipes. The main focus of SDN today is on wired networks (Ethernet, optical transport networks) and on layer 3. ORCA offers a wireless SDN, by extending the current SDN vertical slicing capabilities with lower layer wireless capabilities.

To that end, the vertical pipes (corresponding to different traffic classes) need to be mapped onto the radio resource grid (bottom of Fig. 1), hereby maximally exploiting the radio degrees of freedom like time, frequency and space. It is important to note that the space dimension allows the reuse of spectrum and time resources through space division multiple access (not shown in Fig. 1). The radio resource grid corresponds to the overall capacity of the radio infrastructure. Each block in the radio resource grid represents a

chunk of radio resources consuming a certain part of the airtime, spectrum and space (controlled by the power setting for omni-directional antennas or by directional beam in 3D MIMO case) with a certain PHY configuration (modulation and coding scheme) providing a certain dynamic capacity (in terms of data payload it can carry). This capacity is dynamic, as it changes over time due to changes in the wireless environment (requiring adaptations to the PHY). The mechanism of mapping vertical pipes to radio resource blocks is called radio slicing. It is responsible for the dynamic allocation of available resource blocks in the radio resource grid over the different traffic classes.

The focus of the ORCA project is on wireless functionalities that are needed to extend the current SDN concepts. ORCA has no intention to develop new network-level SDN paradigms, but will align with other SDN-oriented initiatives (based on heterogeneous and cooperative networks integrated through SDN/NFV techniques) as to ensure that ORCA developments are compliant with common SDN mechanisms. The focus of this paper is to enhance data plane functionalities of wireless networks once this is necessary to support more advanced SDN control functionalities in the future of SDWN.

4 Radio Hardware Virtualization

To support the SDN functionalities and ORCA architecture, a requirement analysis is carried out targeting runtime reconfigurable SDR physical and lower-MAC layers. More specifically, the ORCA SDR architecture aims to meet the following requirements:

1. Requirement Analysis

- (a) *RF Resource Slicing* Radio Frequency (RF) resource slicing is used to slice wireless resources, such as spectrum, time and beams, i.e., space beams pointing to specific directions. As a generalized module, it should not stick to any specific standard. A practical choice is to use it as the last stage of the digital processing chain, just before the Analog to Digital Converter (ADC) and Digital to Analog Converter (DAC). The module multiplexes/demultiplexes IQ sample streams from/to physical layer transceivers. The transceivers could be physical entities or logical/virtual entities.

To perform multiplexing/demultiplexing in real time under control parameters, this module needs high processing throughput and precise timing control (in the case of time slicing). For instance, a 4 antenna WiFi RF front-end generates 2.56 Gbps, i.e., a data rate of 20 Msps (IQ samples) \times 32-bit per sample (16-bit I, 16-bit Q) \times 4 antennas. In order to multiplex several WiFi transceivers in the frequency domain, a link with a high data rate is required.

- (b) *Multi-channel Transceiver* Multiple concurrent transceiver instances are necessary in order to utilize radio resources for multiple concurrent beams or frequency channels, in this way, multiple simultaneous services are supported by separate radio slices. A multi-channel transceiver can be achieved by implementing multiple physical instances, or creating multiple logical instances from single or fewer physical instances. In terms of hardware/computing resources occupancy, the latter is better. When the same set of physical resources are shared by multiple logical instances, the hardware context switching speed is essential to support multiple instances.

The core part of the physical layer is the transceiver chain. In general, a receiver

should include synchronization, channel estimation, equalization, decoding, deframing, etc.; on the other hand, a transmitter should include framing, coding, modulation, etc. For low latency standards or time critical services, the transceiver should have low processing delay and should therefore be implemented in ASIC or FPGA. For relaxed latency standards or services, it could be either software or hardware implementation.

- (c) *Context Switching Support* In the computer science domain, when multiple programs/virtual-machines share the same CPU, they actually sleep and wake up frequently and quickly, triggered by user input, network packet arrival or other CPU generated interruption. Before sleep, the CPU's state needs to be saved for the instance. This can be done by saving the CPU's internal registers into the memory. Before waking up, a restoring operation is performed to make sure that the execution is resumed correctly.

Compared with CPU, the radio transceiver functionality is more complicated. There are lots of internal stages, FIFOs, buffers, state machines inside the radio transceiver, therefore, context saving and restoring are challenging operations when one radio transceiver is supposed to be shared or switched quickly among multiple users/services.

Therefore, besides traditional radio transceiver functionalities, the design should also support fast hardware level context saving and restoring. With this feature, a high performance transceiver can be used to process multiple IQ streams in fast switching TDM manner. Along with IQ buffers for each stream and transceiver consuming IQ samples much faster than IQ incoming into each buffer, buffer overflow can be avoided for each IQ stream. Through this way, multiple concurrent logical transceivers can be created from a single transceiver to serve multiple traffic classes, and therefore, multiple end-to-end virtual slices can be implemented efficiently without implementing multiple physical transceivers.

- (d) *Resource Slicing Controller* In this SDR–SDN context, there are two types of resources. The first type refers to the chunk of radio resources that are allocated to a single radio slice (such as beams, spectrum, and time) and can be used by a signal for transmitting and receiving. The second type of resource refers to the operation mode of the transceivers. This type of resource is used to deliver services/traffic within a transceiver's radio slice. We call the first type of resource 'RF resource', and the second resource 'transceiver resource'. To use resources smartly and efficiently under diverse and dynamic requirements, a control software is needed for the real-time management of resources. Although the control software in general is not computationally intensive, controlling resources in precise timing is needed when time slot is used in TDMA MAC, such as multi-frequency TDMA, multi-beam TDMA.
- (e) *SDN Agent* At the AP or edge of the wireless network, a traditional SDN controller might not offer appropriate control functionality toward the AP or base station, because wireless equipments capabilities are more complicated than "just forwarding". However, the traditional SDN controller does know the requirements of traffic classes or users. Therefore a SDN agent that incorporates wireless domain knowledge and that is capable of interpreting (more abstract) SDN requirements and mapping those into control strategies of radio domain resources is needed.

2. **Architecture Design for Implementation:** In order to meet the requirements mentioned in Sect. 4, Fig. 2 depicts an initial architecture design for implementation. The proposed architecture supports both hardware-like low latency performance and software-like flexibility [26]. The platform is composed of RF front-end and digital baseband. The RF front-end can be any of the widely used devices, such as the FMCOMMS2 [27] or USRP [28]. To make a highly efficient design, the digital baseband chip should include not only hardware/FPGA for high-performance low-latency operation but also a processor system to support control and management functionalities in higher network layer. Therefore, System-on-Chip (SoC) architectures are good candidates, such as the Xilinx Zynq SoC [29]. The Zynq SoC consists of two parts, the Programmable Logic (PL) part is mainly the traditional FPGA, and the Processor System (PS) part includes an ARM based multiprocessor system. Two parts are proposed to be implemented in FPGA/HW: the RF resource slicing module and the transceiver resource pool. The first part is used to construct the RF resources, such as beams, channels/bands and time slots, which set the boundaries for transceiver operation in the second block. Multi-channel transceivers are implemented in the second part, with hardware-level fast context switching support. Transceivers construct data path between diverse network traffic/service/user and RF resources under control from software side. For the high-speed and low-latency on-chip connection between hardware blocks and hardware–software, an Advanced Extensible Interface (AXI) stream bus can be used. On-chip software runs in the processor system. Three main software modules are needed: MAC and network protocol; resource slicing controller; SDN agent. As the hardware design supports virtualization, the corresponding MAC and network protocols should also support creating corresponding multiple instances to handle diverse traffic streams in line with the SDN agent. To control the resources in real-time, the resource slicing controller software communicates with the hardware block via the AXI_LITE register interface.
3. **Initial Validation:** A proof of concept demonstration [26] has been developed based on FMCOMMS2 RF front-end and Xilinx ZC706 [29] board as shown in Fig. 3. In this demonstration, a Digital Down-converter (DDC) bank is implemented for the RF spectral resource slicing part. It slices the 40 MHz spectrum (partial 2.4 GHz ISM band) into two adjacent 20 MHz WiFi channels, overlapping with eight 5 MHz ZigBee channels [30]. A dual-standard preamble detector (part of the baseband receiver), with fast hardware context maintenance support, is implemented for the transceiver

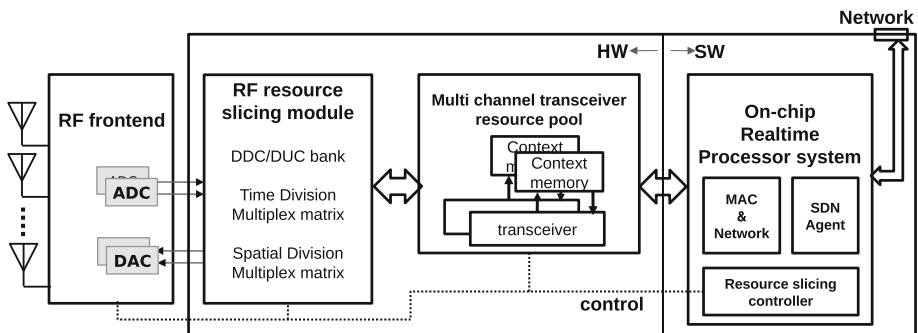


Fig. 2 Architecture proposed to meet the requirements presented in Sect. 4

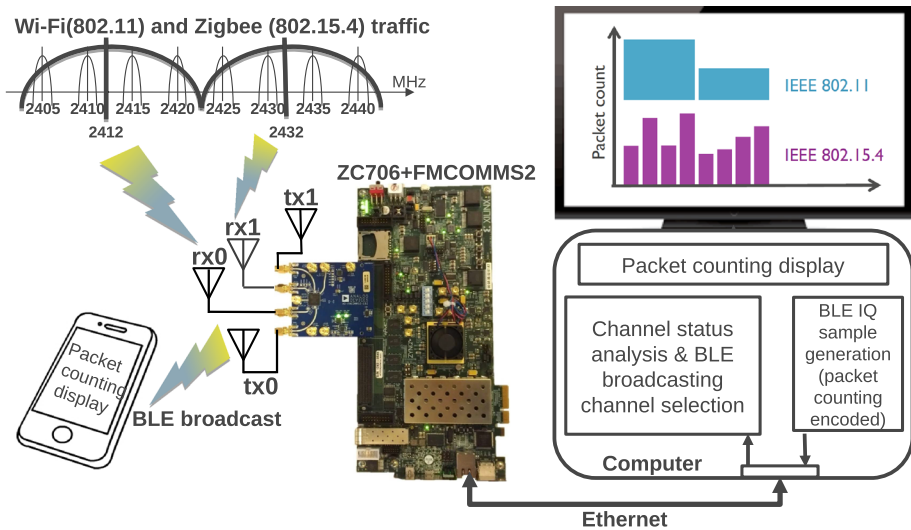


Fig. 3 Demonstration of multiple virtual radios on a single chip

resource part. Based on the FPGA design, the resource slicing controller software in the processor creates 10 virtual preamble detector instances out of the single FPGA preamble detector block to serve 10 input IQ sample streams (2 WiFi, 8 ZigBee). From the user point of view, it is the same as having 10 parallel preamble detectors running concurrently in full-time, which can show packet count statistics of 10 concurrent live traffics in the air. In addition, in order to make the demonstration more user friendly, a Bluetooth Low Energy (BLE) transmitter is implemented in the FPGA. It encodes the packet count statistics information into the BLE broadcasting packet, and broadcasts it over the less busy channel according to packet count detected by 10 virtual preamble detector instances. Then any general purpose BLE scanner/sniffer can read the message on user's devices (phone, notepad, computer, etc.).

5 Conclusion

In this paper, a radio hardware virtualization oriented transceiver architecture is designed to bridge the gap between the diverse real-world applications and the scarce RF resources. This architecture softwarizes the lowest wireless network stack such as PHY and low MAC, while maintaining equally high performance and low latency as in the conventional hardware-defined network. With this radio hardware virtualization feature, the control plane can make efficient RF and hardware resource utilization according to dynamic network traffic/service requirements. The initial proof-of-concept demonstration shows the feasibility of radio hardware virtualization with limited hardware resources. As the next step in the ORCA project, we will bridge real-time SDR and SDN with the help of radio hardware virtualization and exploit maximum flexibility at PHY, MAC and network levels, as a way to support very diverse application requirements by efficiently sharing limited RF and transceiver resources.

Acknowledgements The project leading to this application has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 732174 (ORCA project).

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

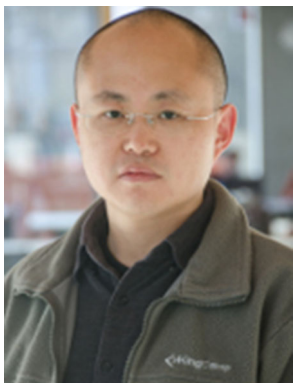
References

1. Sinha, R. S., Wei, Y., & Hwang, S.-H. (2017). A survey on LPWA technology: LoRa and NB-IoT. *ICT Express*, 3(1), 14–21.
2. Lauridsen, M., Vejlgård, B., Kovacs, I. Z., Nguyen, H., & Mogensen, P. (2017). Interference measurements in the European 868 MHz ISM band with focus on LoRa and SigFox. In *IEEE wireless communications and networking conference (WCNC)*.
3. Elahi, A., & Gschwendner, A. (2009). *ZigBee wireless sensor and control network*. Upper Saddle River: Prentice Hall Press. ISBN: 0137134851.
4. Sauter, M. (2017). *From GSM to LTE-advanced Pro and 5G: An introduction to mobile networks and mobile broadband*. New York: Wiley. ISBN: 9781119346869.
5. IEEE Standards Association. (1999). IEEE Standard for Information Technology - Telecommunications and information exchange between systems - Local and Metropolitan networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher Speed Physical Layer (PHY) Extension in the 2.4 GHz band, IEEE Std 802.11b-1999, pp. 1–96. <https://doi.org/10.1109/IEEESTD.2000.90914>.
6. Mishra, A. R. (2006). *Advanced cellular network planning and optimisation: 2G/2.5G/3G...evolution to 4G*. New York: Wiley. ISBN: 9780470014714.
7. Sesia, S., Toufik, I., & Baker, M. (2011). *LTE—The UMTS long term evolution: From theory to practice*. New York: Wiley. ISBN: 9780470660256.
8. ORCA Project. ORCA orchestration and reconfiguration control architecture. Retrieved September 25, 2017, from <https://www.orca-project.eu/>.
9. Eric Wang, Y.-P., Lin, X., Adhikary, A., Grovlen, A., Sui, Y., Blankenship, Y., et al. (2017). A primer on 3GPP narrowband Internet of Things (NB-IoT). *IEEE Communications Magazine*, 55(3), 117–123.
10. Takeda, K., Wang, L. H., & Nagata, S. (2017). Industry perspectives: Latency reduction toward 5G. *IEEE Wireless Communications*, 24(3), 2–4.
11. Shahwaiz Afaqui, M., Garcia-Villegas, E., & Lopez-Aguilera, E. (2016). IEEE 802.11ax: Challenges and requirements for future high efficiency WiFi. *IEEE Wireless Communications*, 24(3), 130–137.
12. Zaidi, A. A., Baldemair, R., Tullberg, H., Björkegren, H., Sundström, L., Medbo, J., et al. (2016). Waveform and numerology to support 5G services and requirements. *IEEE Communications Magazine*, 54(11), 90–98.
13. Holland, O., Bogucka, H., & Medeis, A. (2015). *Opportunistic spectrum sharing and white space access: The practical reality*. New York: Wiley. ISBN: 978-1-118-89374-6.
14. Paisana, F., Kaminski, N. J., Marchetti, N., & DaSilva, L. A. (2017). Signal processing for temporal spectrum sharing in a multi-radar environment. *IEEE Transactions on Cognitive Communications and Networking*, 3(2), 123–137.
15. Xiao, M., Mumtaz, S., Huang, Y., Dai, L., Li, Y., Matthaiou, M., et al. (2017). Millimeter wave communications for future mobile networks. *IEEE Journal on Selected Areas in Communications*, 35(9), 1909–1935.
16. Nitsche, T., Cordeiro, C., Flores, A. B., Knightly, E. W., Perahia, E., & Widmer, J. C. (2014). IEEE 802.11ad: Directional 60 GHz communication for multi-Gigabit-per-second WiFi [Invited Paper]. *IEEE Communications Magazine*, 52(12), 132–141.
17. Wang, B., & Ray Liu, K. J. (2011). Advances in cognitive radio networks: A survey. *IEEE Journal of Selected Topics in Signal Processing*, 5(1), 5–23.
18. Bhanage, G., Vete, D., Sesar, I., & Raychaudhuri, D. (2010). SplitAP: Leveraging wireless network virtualization for flexible sharing of WLANs. In *IEEE global telecommunications conference (GLOBECOM)*.

19. Dely, P., Vestin, J., Kassler, A., Bayer, N., Einsiedler, H., & Peylo, C. (2012). CloudMAC—An OpenFlow based architecture for 802.11 MAC layer processing in the cloud. In *IEEE globecom workshops (GC Wkshps)*.
20. Zahid, T., Yousuf Dar, F., Hei, X., & Cheng, W. (2016). A measurement study of a single-BSS software defined WiFi testbed. *IEEE international conference on computer communication and the internet (ICCCI)*.
21. Zahid, T., Hei, X., & Cheng, W. (2016). Understanding the design space of a software defined WiFi network testbed. In *International conference on frontiers of information (FIT)*.
22. Chen, K., & Duan, R. (2011). C-RAN the road towards green RAN. China Mobile Research Institute, white paper.
23. Kazaz, T., Liu, W., Jiao, X., Moerman, I., Paisana, F., Felber, C., Kotzsch, V., Seskar, I., Vermeulen, T., Pollin, S., Danneberg, M., & Bomfin, R. (2017) Orchestration and reconfiguration control. In *European conference on networks and communications (EuCNC)*.
24. 5G-PPP White Paper. (2015). 5G and the factories of the future. Available from: <http://clear5g.eu/blog/5g-ppp-white-paper-factories-future>. Accessed 25 Sept 2017.
25. Sabella, R. (2015). 5G & cloud robotics For industrial IoT. Ericsson Presentation.
26. Jiao, X., Moerman, I., Liu, W., & de Figueiredo, F. A. P. (2017). Radio hardware virtualization for coping with dynamic heterogeneous wireless environments. In *EAI international conference on cognitive radio oriented wireless networks (CROWNCOM)*.
27. Analog Devices. (2016). AD-FMCOMMS2/3/4/5-EBZ Zynq and ZED quick start guide. <https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms2-ebz/quickstart/zynq>. Accessed 26 Sept 2017.
28. Ettus Research. List of products. <https://www.ettus.com/product>. Accessed 25 Sept 2017.
29. Xilinx. Xilinx Zynq-7000 all programmable SoC ZC706 evaluation kit. <https://www.xilinx.com/products/boards-and-kits/ek-z7-zc706-g.html>. Accessed 26 Sept 2017.
30. IEEE Standards Association. (2016). IEEE Standard for Low-Rate Wireless Networks, IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011), pp. 1–709. <https://doi.org/10.1109/IEEESTD.2016.7460875>.



Felipe A. P. de Figueiredo received the B.S. and M.S. degrees in Telecommunications from Instituto Nacional de Telecomunicações (INATEL), Minas Gerais, Brazil, in 2004 and 2011 respectively. He is currently working toward the Ph.D. degree with the Internet Technology and Data Science Lab, Ghent University, Gent, Belgium. He has been working in R&D of telecommunications systems for more than 10 years. His research interests include digital signal processing, digital communications, mobile communications, MIMO, multicarrier modulations and FPGA development.



Xianjun Jiao received his bachelor degree in Electrical Engineering from Nankai University in 2001 and Ph.D. degree on communications and information system from Peking University in 2006. After his studies, he worked in industrial research institutes and product teams in the domain of wireless technology, such as Radio System Lab of Nokia Research Center (senior researcher), devices department of Microsoft (senior researcher) and Wireless Software Engineering department of Apple (RF software engineer). In 2016, he joined IDLab (<http://www.ugent.be/ca/idlab/en>), a core research group of imec (<http://www.imec.be/>) with research activities embedded in Ghent University and University of Antwerp. He is working as postdoc researcher at Ghent University on flexible realtime SDR platform. His main interests are SDR and parallel/heterogeneous computation in wireless communications. On his research track, 20+ international patents and papers have been authored/published.



Wei Liu was born in China in 1986. She received the master's degree in electronic engineering from the University of Leuven, Campus GroepT, in 2010, and the Ph.D. degree from the IDLab, a core research group of IMEC with research activities embedded in Ghent University and the University of Antwerp, in 2016. During her doctoral education, she participated in multiple research projects, she became familiar with several software-defined radio platforms, and gained experiences in wireless testbed operations. She is a Post-Doctoral Researcher with Ghent University. Her research is conducted in the field of cognitive radio, focusing on spectrum analysis and interference prevention.



Ingrid Moerman received her degree in Electrical Engineering (1987) and the Ph.D. degree (1992) from the Ghent University, where she became a part-time professor in 2000. She is a staff member at IDLab, a core research group of imec with research activities embedded in Ghent University and University of Antwerp. Ingrid Moerman is coordinating the research activities on mobile and wireless networking, and she is leading a research team of about 30 members at IDLab-Ghent University. Her main research interests include: Internet of Things, Low Power Wide Area Networks (LPWAN), High-density wireless access networks, collaborative and cooperative networks, intelligent cognitive radio networks, real-time software defined radio, flexible hardware/software architectures for radio/network control and management, and experimentally-supported research.