

Received 10 June 2025, accepted 30 June 2025, date of publication 7 July 2025, date of current version 11 July 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3586645

RESEARCH ARTICLE

A mmWave JCAS System for Real-Time High-Data Rate Communication and RADAR Sensing

MARKUS PETRI¹ AND NEBOJSA MALETIC¹, (Senior Member, IEEE)

IHP-Leibniz-Institut für innovative Mikroelektronik, 15236 Frankfurt (Oder), Germany

Corresponding author: Markus Petri (petri@ihp-microelectronics.com)

This work was supported by the German Federal Ministry of Research, Technology, and Space (BMFTR) within the Project Open6GHub under Grant 16KISK009.

ABSTRACT This paper presents a real-time millimeter-wave (mmWave) joint communication and sensing (JCAS) system, supporting bi-directional data communication with up to 5.8 Gbit/s coded data rate as well as mono-static radio detection and ranging (RADAR) with a range resolution of 6.7 cm and up to 2.1 kHz sensing rate. The same hardware resources, i.e., an orthogonal frequency-division multiplexing (OFDM) baseband processor and a beam-steering phased array antenna frontend, the same signal waveform and the same frequency channel are used for both sensing and communication. After a description of one intended use case and the design constraints, the system architecture is described in detail. The focus is on two main modules, i.e. the advanced extensible interface 4 (AXI4) stream crossbar switch and the sensing controller, while changes of the basic OFDM baseband processor design are also explained. For the sensing function, the used hardware-based OFDM baseband processor does not provide a sufficiently precise synchronization of received frames, so a method for the temporal alignment of the received channel impulse responses (CIRs) for different beams is developed and assessed. The evaluation of the system performance and the measurement results highlight that it is possible to perform very precise sensing with an OFDM baseband processor purely developed for data communication. The presented method for the CIR alignment does not depend on the baseband signal processing and can be used in any JCAS system with full duplex sensing.

INDEX TERMS 6G, joint communication and sensing (JCAS), ISAC, ICAS, RADAR, mmWave, real-time.

I. INTRODUCTION

Joint communication and sensing (JCAS, JCS) is one of the hot topics of current research, especially in the context of the development of the sixth generation of mobile networks, 6G [1], [2]. The highlight of JCAS is the combination of wireless data communication and radio sensing functionalities in the same system, by leveraging commonalities of both. A JCAS system does not only physically integrate a communication system and a radio detection and ranging (RADAR) device. Instead, the same resources including radio frequency (RF) spectrum and hardware are used for both sensing and communication. The goal is to develop a system where both functionalities operate in a cooperative

manner, to minimize interference and enable new use cases. Furthermore, sensing is not limited to RADAR and therefore object detection and localization, but also comprises a complete (RF) environmental sensing. Thus, the provision of channel state information (CSI) for further processing in higher layers or applications is also a JCAS functionality.

In the literature, JCAS is also called integrated sensing and communication (ISAC) [3], [4], integrated communication and sensing (ICAS) [5] or joint radar and communication (JRC) [6], depending on the main focus of the application scenario. In this paper, the term JCAS is used to emphasize that no additional RADAR sensing components are added to the basic communication system. The communication system, its data communication waveform and its signal processing are used for both communication and RADAR sensing.

The associate editor coordinating the review of this manuscript and approving it for publication was Walid Al-Hussaibi¹.

A. RELATED WORK

Although there is a lot of literature on different aspects of JCAS / ISAC like waveform design [7], [8], [9] and signal processing [10], [11], there are only a few publications presenting the system architecture, implementation aspects and system performance of integrated real-time JCAS systems.

Recently, experimental verification of a method for coherent multi-band ranging has been performed, using an AMD UltraScale+ RFSoc platform [12]. While the experimental verification includes the implementation of the developed method, the RFSoc platform is only used for storing the send and receive waveforms. The waveforms are processed offline on a PC. Thus, the implemented system is not capable of real-time operation.

A prototype of a real-time millimeter-wave (mmWave) ISAC system is presented in [13]. There are no details on the system implementation given. Besides communication, it supports mono-static RADAR sensing. This prototype uses orthogonal frequency-division multiplexing (OFDM) at a carrier frequency of 28 GHz, but a rather small bandwidth of 100 MHz. From the parameters given in the publication, the coded data rate can be estimated to be lower than 160 Mbit/s. For the sensing, no performance values are presented. It is expected that the range resolution is rather low due to the small bandwidth.

Another joint communication and RADAR proof-of-concept platform is described in [14]. Here, an existing mmWave communication set-up was extended with an additional full-duplex RADAR receiver. It operates in the 71-76 GHz range and uses a software-defined radio (SDR) approach with a fully digital multiple antenna processing and therefore a huge amount of computational resources. The paper focuses on the RADAR aspects and not on the joint system architecture.

Real-time capable SDR prototyping platforms are presented in [15] and [16]. While JCAS applications can be realized with them, their focus is on the development support for and evaluation of multiple input multiple output (MIMO) antenna systems or signal processing algorithms. This first one allows only capturing a signal of 10 seconds, so it cannot be used in a continuously operating system. The second platform supports a continuous sample stream of 2.5 GSps sent over a 100 Gbit/s Ethernet link to a host computer (PC). Therefore, the host PC must be able to process this amount of data in software.

B. CONTRIBUTION

In contrast to the related work, this paper presents a real-time mmWave JCAS system with the digital processing integrated in one single system-on-chip, resulting in a small form factor. It uses hardware-based processing to reach a high data throughput up to several Gbit/s and a short sensing time below half a millisecond. The main hardware consists of a commercial off-the-shelf (COTS) field

programmable gate array (FPGA) evaluation board and a COTS antenna front-end module. Furthermore, all details on system implementation aspects are given, together with a performance evaluation including the necessary trade-offs between communication and sensing functionality.

In summary, the main contributions are:

- A JCAS system architecture for real-time operation, using an existing baseband processor designed for communication
- Full details on the hardware implementation on a system-on-chip (SoC)
- A method to deal with potentially imprecise frame synchronization of baseband processors
- Measurement campaigns to evaluate the system, showing that a very good sensing performance can be reached with baseband processors designed for communication
- A discussion on the trade-offs between communication and sensing functionality

C. OUTLINE

This paper is organized as follows: First, a potential use case for the JCAS system and its design constraints are presented in Section II. This section also includes the derived requirements and the description of the mode of operation. In Section III, the system architecture is presented and the functionality and implementation of the main building blocks, especially the advanced extensible interface 4 (AXI4) stream crossbar switch and the sensing controller, is explained. Afterwards, the method used for the alignment of the channel impulse responses (CIRs) from different beams is presented in Section IV. The performance evaluation of the implemented JCAS system is given in Section V, where potential improvements are also derived. Conclusions are drawn in Section VI, together with an outlook on future work.

II. SYSTEM OPERATION AND DESIGN CONSTRAINTS

A lot of use cases for future JCAS systems are currently under discussion and evaluation. Some of them are presented in [17]. The present work does not target a specific use case. Instead, the implemented JCAS system combines high data rate communication with RADAR-like sensing. Based on the system parameters and performance with a carrier frequency of 60 GHz and a measured range distance of up to 17 m, indoor use cases are currently more suitable. The range limitation comes from specific algorithms implemented in the baseband processor. Larger RADAR distances of around 50 m are in principle possible (see Sec. V). One potential use case is a sensing-assisted predictive beam steering. Here, the sensing is used to detect and to track objects which may block the line-of-sight (LOS) communication link. If such a potential blockage is predicted, the communication beam is switched to another direction, using a previously identified reflection path. Thus, an interruption of the communication link is prevented.

A similar approach of link-blockage prediction for mmWave communication is demonstrated in [18]. Here, a light detection and ranging (LIDAR) system is used to detect moving objects which might block the LOS path.

A. DESIGN CONSTRAINTS AND REQUIREMENTS

The idea of this work is to enhance a real-time mmWave communication system by (mono-static) RADAR sensing functionality in an JCAS approach. According to the mentioned understanding of JCAS, no specific RADAR components should be added. As one main goal, impacts of the JCAS functionality on the system implementation should be derived. Furthermore, it should be (and is) shown that a very good sensing performance can be reached even with a system which was optimized for communication. Thus, an existing OFDM baseband processor and an existing SDR-based JCAS platform are combined and enhanced with sensing-specific components (see Section III) to create a real-time capable JCAS system.

An additional design constraint of our current work is that changes on the basic OFDM baseband processor should be avoided. Therefore, there is limited flexibility in choosing the best-suited components and algorithms for each building block.

The basic SDR platform contains an analog beamforming mmWave antenna frontend, which is designed for communication. The existence of only one RF chain in the antenna frontend module prevents MIMO-based sensing approaches like the one presented in [5], having access to the individual signals of all antenna elements. Instead, a horizontal beam scan is used to obtain a range-angular RADAR map.

B. MODE OF OPERATION

Before starting the detailed elaboration of the system architecture, it is essential to briefly explain the operational mode of the JCAS system. While it is a *joint* communication and sensing system meaning that the RADAR functionality is realized just by using the components of the communication part, sensing and communication do not operate simultaneously. Instead, separate time slots are used. In each slot, either the communication or the sensing functionality is active. For the sensing, the OFDM ranging method presented in [19] is used to estimate the distance of reflectors from the acquired CIR. The angular position is derived based on a beam scan using analog beam-steering antenna frontend modules (see Sec. III).

In the communication slot, data frames are sent and received by the OFDM baseband processor. The structure of these OFDM frames is depicted in Fig. 1. The first part of a frame is an OFDM preamble used for frame synchronization (SYNC), channel frequency offset (CFO) compensation and channel estimation (CE). The preamble is followed by a signal field (SF) symbol containing information for the processing of the data symbols, e.g. the payload modulation and the payload length. Finally, there is a number of OFDM data symbols carrying the data payload. More details about

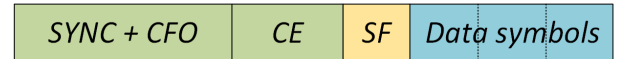


FIGURE 1. OFDM frame structure (green: OFDM preamble, orange: signal field OFDM symbol, blue: OFDM data symbols).

the OFDM parameters and the baseband processing are given in Sec. III-E.

In the sensing slot, a full beam scan is done in the following way: The preamble also used for communication is sent in every beam direction. Simultaneously, the received signal is processed by the receiver chain of the OFDM baseband processor and the estimated channel coefficients (CHEs) for each beam are stored. In the current measurement setup, the CHEs are transferred to a PC after a beam scan is finished. There, the software-based post-processing consists of an inverse Fast Fourier Transform (FFT) operation to transfer the frequency-domain CHEs into time-domain CIRs. Furthermore, an alignment procedure (see Section IV) is applied. The transfer of the CHE data and the post-processing do not need to be performed within a sensing time slot, it can be independently done at any time. Thus, the duration of the sensing time slot can be shortened, as will be discussed in Section V-A. Furthermore, the post-processing can be also implemented on an edge server or the JCAS transceiver itself as a part of the RADAR-based object detection and tracking. In this work, the focus is on the implementation of the JCAS physical layer (PHY). The required data rates for the transmission of the sensing data are given in Section V-A. Options for the reduction of the amount of data to be transferred and for reducing the processing time on the PC respectively the edge server are discussed in Section V-C.

The necessity for using different time slots for communication and sensing results from the already mentioned use of analog beam-steering frontends. Such frontends are common in mmWave frequency bands. Contrary to MIMO-based sensing approaches, the RADAR sensing with analog beam-steering frontends requires a sequential beam scan over all possible beams. This impedes simultaneous data communication, since for the communication, one (fixed) transmit/receive beam has to be set to optimize the link budget. Although it is possible to transmit data frames during the beam-scan, in most cases the signal-to-noise ratio (SNR) would be too low for the receiver to decode them. Furthermore, using complete data frames would increase the time needed for the sequential beam-scan, as shown in Section V-A.

While using frames with data payload is not recommended during the sensing slot, communication beams might be omitted during the sensing beam scan. Their CIRs are already known. Thus, omitting those beams would reduce the time for the beam scan. But as shown in Section V-A, the required channel coherency time would still increase.

III. SYSTEM ARCHITECTURE

The developed real-time mmWave JCAS system is based on the mmWave SDR platform from [20]. The platform

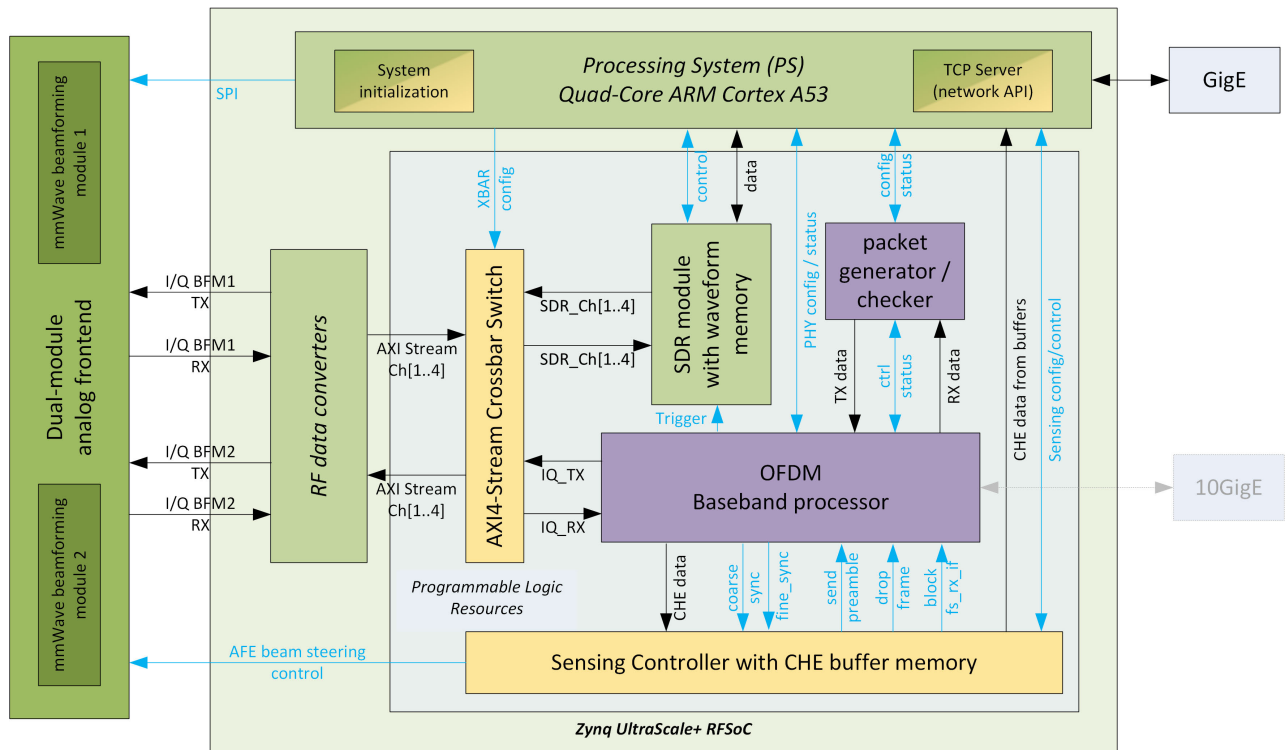


FIGURE 2. Architecture of the mmWave JCAS system (blue arrows: control signals; black arrows: data signals).

consists of an AMD ZCU111 evaluation board [21] and a dual module analog antenna frontend equipped with two Sivers BFM06005 60 GHz phased array beam-steering modules [22]. Each frontend module has independent TX and RX patch array antennas with 16 dual-patch elements in a line. This allows beam-steering in azimuth direction. More details on the beam-steering are given in Sec. V-A. It is possible to change the frontend modules to the successor module BFM06009, which enables beam steering in azimuth and elevation. More general, the presented JCAS system is independent from the used analog mmWave frontend, as long as the frontend supports the baseband interface and signal bandwidth and is capable of digitally controlled analog beam-steering.

In this work, the SDR platform was extended with the real-time OFDM baseband processor from [23] and [24] and some new components for enabling the real-time JCAS functionality. The development of the new components follows the hardware-software co-design approach from the base platform.

The central element of the SDR platform is an AMD UltraScale+ RFSoc [25]. This SoC is made of a quad-core ARM Cortex-A53 processing system (PS), 8 high-speed analog-digital converters (ADCs) and 8 high-speed digital-analog converters (DACs) as well as a huge amount of programmable logic resources (i.e. the FPGA part of the SoC). The data converters support sample rates up to around 6.5 GSps (DAC) and 4.1 GSps (ADC), respectively.

The architecture of the real-time mmWave system is shown in Fig. 2. For the sake of clarity, only the relevant blocks in the context of this paper are included. Other blocks required for the general board operation, e.g. the clocking structure and the external dynamic random access memory (DRAM), are omitted. Green color indicates blocks from the basic SDR hardware platform [20], whereas purple blocks mark the existing OFDM baseband processor [23], [24] and its infrastructure. Orange blocks show the new blocks developed for the JCAS functionality. The mmWave JCAS system operation is controlled from a PC application through a transmission control protocol (TCP) connection over Gigabit-Ethernet (GigE). In the following subsections, the shown blocks are described in more detail. For the blocks taken from other works, the focus is put on implications due to their specific parameters as well as necessary changes for an efficient system integration. Implementation results for the whole mmWave JCAS system are presented in Section V-A.

A. PROCESSING SYSTEM AND TCP SERVER (NETWORK API)

The firmware running on the PS implements routines for the initial system configuration in a bare metal (standalone) application. It furthermore includes a TCP server with a *network API*. Through this application programming interface (API), firmware routines to configure the parameters of the implemented blocks, to start and stop the different operational modes and to readout the status and results can be called.

Algorithm 1 Pseudo-code to perform sensing using *network API*

```

1: tcp_open(ip_address)
2: tcp_call(start_sensing, start_beam, stop_beam)
3: while status  $\neq$  finished do
4:   status = tcp_call(get_sensing_status)
5: end while
6: che_data = tcp_call(read_che_memory)
7: radar_heatmap = postprocess(che_data)
8: plot(radar_heatmap)
9: tcp_close(ip_address)

```

A pseudo-code program to perform sensing and to display the resulting angular/range RADAR heatmap is shown in Alg. 1. The function *tcp_open* establishes a TCP connection to the JCAS station specified by its IP address. The function *tcp_call* is used to send different commands, e.g. to start the sensing, to get the current status or to readout memory buffers. Together with the command, necessary parameters like the start and stop beam for a scan are sent to the JCAS station.

B. RF DATA CONVERTERS

The used AMD UltraScale+ RFSoc contains 16 high-speed data converters (8 ADCs, 8 DACs). The basic SDR design supports 4 ADCs and 4 DACs, which is sufficient for the dual-module beam-steering frontend with an in-phase and quadrature-phase (IQ) modulated baseband signal interface. Although the sampling clock of the data converters can be freely adjusted, the basic SDR design requires a sampling clock which is an integer multiple of 160 MHz. Otherwise, the sample synchronization of the data converters would not properly work, leading to a sample offset in the IQ signal. As a consequence, the target sampling frequency for the existing OFDM baseband processor (2.16 GSps) cannot be generated. Thus, the sample rate is slightly increased to 2.24 GSps. The AXI4-Stream interface [26] of each data converter provides 8 samples in parallel and operates at a clock rate of 280 MHz.

C. SDR MODULE

The SDR module contains the memory to store the signal waveforms which should be sent out through the DACs and the signal waveforms which are sampled from the ADCs. Its current implementation supports 4 ADCs and 4 DACs. Furthermore, the SDR module contains trigger logic. One trigger input channel is connected to the OFDM baseband processor. Thus, it is possible to analyze the IQ signal generated by the OFDM baseband processor in a loopback mode (see Section III-D).

D. AXI4-STREAM CROSSBAR SWITCH

The AXI4-Stream crossbar switch (*XBAR*) is the main extension of the basic SDR platform. Its general structure is shown in Fig. 3. The *XBAR* allows to switch between different sources for the AXI4-Stream data input of each individual

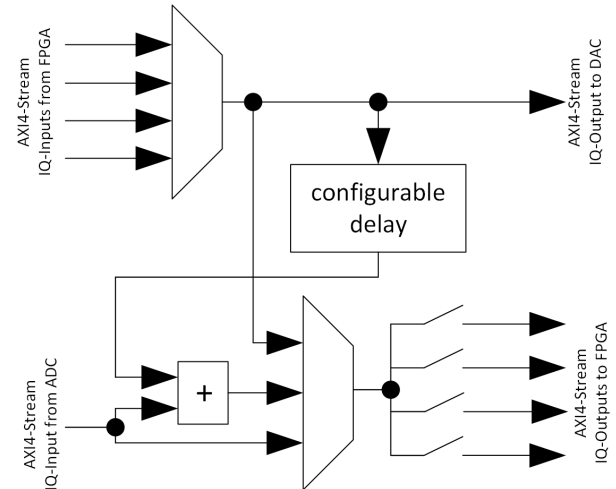


FIGURE 3. Block schematic of AXI4-Stream crossbar switch.

DAC channel. Similarly, it forwards the sample streams received from the ADCs to different sinks, which can be individually specified for each ADC channel. Thus, the *XBAR* allows the integration of other signal processing modules like the OFDM baseband processor, without removing the SDR functionality from the system. While there can be only one signal source for a DAC channel, the samples of one ADC channel can be duplicated and forwarded to several signal sinks. In this way, it is e.g. possible to process or analyze the received signal in both the hardware-implemented baseband processor and the SDR signal processing on a PC.

Besides routing to and from the data converters, the *XBAR* supports a loopback mode as well as an overlay mode. In the loopback mode, the input from the selected signal source is locally routed to the enabled FPGA fabric outputs. This digital loop simplifies the debugging of hardware modules.

In the overlay mode, the output signal to the sinks is the summed signal of the ADC sample stream and the delayed signal of the selected input source. The delay line is realized with distributed memory. In the current implementation, the delay can be configured in a range of 0 to 60 clock cycles. The overlay mode is essential for the proposed CIR alignment method (see Section IV).

The AXI4-Stream interface of the data converters operates at a high clock frequency of 280 MHz. In pure SDR mode, the system supports even higher frequencies up to 500 MHz. To not sacrifice the maximum clock speed by including too many logic levels and to not increase the signal delay by adding heavy pipelining, the *XBAR* does not implement a full any-to-any crossbar switch. Instead, the current implementation follows a hierarchical approach. The basic *XBAR* module supports 4 different signal sources and 4 signal sinks. Each source and each sink provide two parallel AXI4 data streams, i.e. the in-phase (I) and quadrature-phase (Q) component of the sampled signal. On the converter side, two ADCs and two DACs are connected to the *XBAR*, one for the I component, the other for Q. The *XBAR* is implemented

TABLE 1. OFDM baseband parameters.

Parameter	Value
Sample rate	2.24 GSps
FFT size	1024 points
Data sub-carriers	768
Pilot / zero sub-carriers	60 / 5
Used channel bandwidth	≈ 1.825 GHz
Sub-carrier spacing	≈ 2.19 MHz
Symbol duration	≈ 572 ns
Preamble duration	≈ 3.7 μ s
Frame duration (2 kB payload, QPSK)	≈ 16.9 μ s
Frame duration (4 kB payload, QPSK)	≈ 28.8 μ s
Sub-carrier modulation	BPSK, QPSK
Coded data rate (BPSK)	≈ 1.35 Gbit/s
Coded data rate (QPSK)	≈ 2.7 Gbit/s

twice to support the four ADC respectively DAC channels available in the basic SDR platform.

The configuration of the *XBAR* is controlled from the processing system through an AXI general purpose input/output (GPIO) block [27] with an AXI4-lite interface. The firmware contains the appropriate functions, which are also included into the *network API*.

E. OFDM BASEBAND PROCESSOR AND PACKET GENERATOR

As already mentioned, an existing OFDM baseband processor [23], [24] was integrated to realize the communication functionality. The OFDM frame structure (shown in Fig. 1) and the OFDM parameters are aligned to the IEEE 802.11ad standard [28]. The major difference to the standard is that the baseband processor uses an FFT size of 1024 points, while the standard proposes an FFT size of 512 points. The OFDM parameters are summarized in Table 1. There, some minor implementation changes compared to the original specification are taken into account. These changes are described in the following. The original design supports binary phase shift keying (BPSK), quadrature phase shift keying (QPSK) and quadrature amplitude modulation (QAM) subcarrier modulation schemes with a maximum net PHY data rate of nearly 3.9 Gbit/s, but only a reduced version is used in this work: The number of parallel coding channels was reduced from 24 to 8 and accordingly, the subcarrier modulation was limited to BPSK and QPSK. Furthermore, the puncturing modes were removed, i.e. only a convolutional channel coding with a code rate of $1/2$ is supported. The reason for those changes is that this work does not target to reach the maximum possible data rate. Instead, the feasibility of a real-time mmWave JCAS system should be shown. The system can be easily scaled to higher data rates by including the removed coding channels and the puncturing modes as well as 16-QAM subcarrier modulation. The reduction of the coding channels has no other implication than a reduction of the processing time needed for the synthesis, placing and routing of the FPGA design.

The parallel processing of 8 data samples in the baseband processor directly fits to the number of parallel samples in the AXI4-Stream interface of the data converters. The slightly increased sample rate of 2.24 GSps results in a channel bandwidth of 2.24 GHz. The clocking for the other clock domains of the baseband processor is accordingly adjusted to fit the data stream requirements: 200 MHz is increased to 210 MHz, 135 MHz and 125 MHz are increased to 140 MHz. With the higher bandwidth of 2.24 GHz, the net PHY data rate is increased by $\approx 3.8\%$ from 1.3 to 1.35 Gbit/s for QPSK subcarrier modulation.

According to the mentioned understanding of *joint* communication and sensing (see Section I), the OFDM baseband processor is handled as closed box during the integration. This means, to support the additional sensing operation, no changes are made in this block or at its interfaces. The only exceptions are three additional control signals: *send_preamble*, *drop_frame* and *block_fs_rx_if*. They allow the transmission and receiver processing of a pure preamble (without following signal field and payload OFDM symbols). This change simplifies the handling and decreases the necessary time for the sensing, as shown in Section V. As an alternative, a usual OFDM data frame with a payload size of 0 bytes - or even with a payload - could have been used to enable the JCAS functionality without any change of the baseband processor.

Asserting the signal *send_preamble* will directly start the transmission of an OFDM preamble. Since no signal field is transmitted after the preamble, the signal processing at the receiver is stopped after the channel estimation by setting the *drop_frame* signal from the sensing controller. Usually, this signal is internally set after the signal field processing, in case the signal field could not be correctly decoded. Since no valid signal field is transmitted during sensing, this signal would also be set by the baseband processor, but after an additional processing of the supposed signal field sequence. The modification of the *drop_frame* signal makes it necessary to block the internal data transmission between the two parts of the baseband receiver, i.e. the OFDM symbol processing and the datapath processing starting with the demapper. Otherwise, unexpected behavior could occur. For this purpose, the *block_fs_rx_if* signal is added, which is assigned during the whole sensing process. It is emphasized again that those changes are just made for a shorter sensing time. They are not necessary to enable the JCAS functionality.

Other necessary signals for sensing, i.e. the coarse and fine synchronization indication, are already provided by the existing OFDM baseband implementation. The same is true for the interface signals to read out the estimated channel coefficients from the baseband processor.

The data payload for the OFDM communication frames is generated by a dedicated packet generator. It also controls the start of the individual packet transmission through a configurable frame gap counter. OFDM parameters like the modulation scheme and the number of payload bytes can be configured through the *network API*. The packet

generator furthermore includes a packet checker to evaluate the received packets and to determine the packet error rate.

F. SENSING CONTROLLER

The hardware-implemented sensing controller consists of a finite state machine (FSM), a timestamp counter and a memory buffer structure to store the estimated channel coefficients. Its structure is shown in Fig. 4. The dual-port memory block for the amplitude and phase data of the channel coefficients (CHE buffer) is split into 64 regions of 128 memory rows each, according to the 64 different beam settings of the analog frontend. The baseband processor provides 8 channel coefficients (*che_data*) in parallel in one clock cycle. Thus, each buffer could store 1024 channel coefficients. Since the baseband just provides the channel coefficients for the 828 data carriers, 24 memory rows are unused. This overhead of 23 % is tolerated to ease the buffer selection by just using the address bits 8 and 9 to specify the buffer. Furthermore, the highest memory address is used to store the timestamps of the frame synchronization event (*ts_coarse_sync*) and the fine synchronization event (*ts_fine_sync*). Finally, there are only block random access memory (BRAM) primitives with fixed sizes available in the FPGA, so that the real implementation overhead in terms of used primitives will be much smaller, as discussed in Section V-C. The stored channel coefficients can be read out from the PS through the signals *che_buffer_addr* and *che_buffer_data*.

In addition to the channel coefficients itself, the baseband processor provides an address (*che_ram_addr*) for each *che_data* word. Since the provided channel coefficients are ordered with increasing OFDM subcarrier indices, *che_ram_addr* corresponds to the lowest subcarrier index of the current eight channel coefficients. Therefore, the CHE buffer address is generated by concatenating the index of the current buffer region (*buffer_addr*) with the provided address *che_ram_addr*. For storing the timestamps, the highest memory address in the current buffer region is selected with the *ram_addr* signal.

Similar to the baseband processor, three different clock domains are used. The timestamp counter operates at 280 MHz to provide the best resolution and minimum jitter of the events. The CHE buffer memory is clocked with 210 MHz, since the channel estimator of the baseband operates at this frequency. The control FSM operates at 140 MHz, to avoid a large number of clock domain crossing signals at the interface.

The FSM of the sensing controller implements the whole sensing process, as presented in Alg. 2. For the sake of clarity, necessary wait cycles are not shown. To allow fast beam scanning, a configurable time-out counter is included in the sensing controller FSM block. For each step, the maximum time to wait for an expected event is set. If it is overdue, an “all ones” timestamp will be stored at the highest buffer address to mark this buffer as invalid. E.g. if no frame is received or the fine synchronization fails, the coarse sync

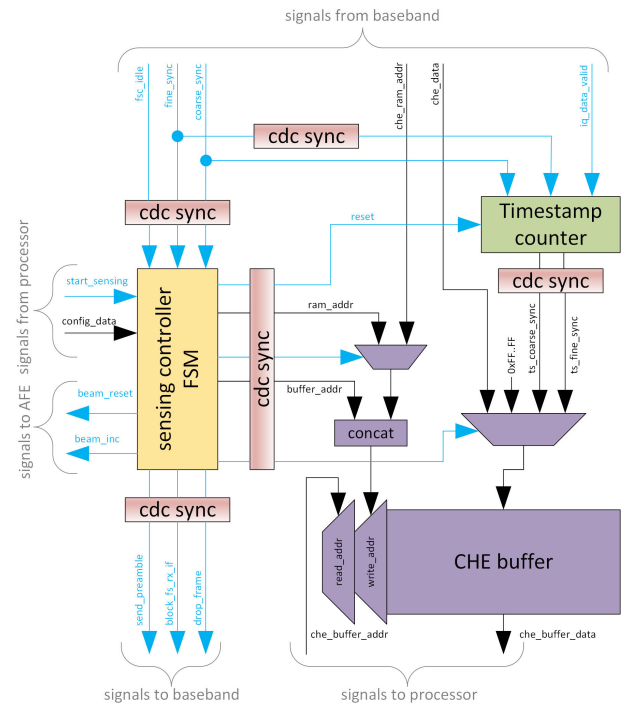


FIGURE 4. Block schematic of sensing controller (block colors indicate different clock regions, blue arrows: control signals, black arrows: data signals, cdc sync: clock domain crossing synchronization stages).

and/or fine sync signals will not be set by the baseband processor. The time-out counter is furthermore re-used to set some wait cycles in different states of the process, e.g. during the initial beam reset. Parameters for the sensing operation like the start and stop beam index are provided from the PS through the *config_data* interface.

The timestamp counter is reset from the FSM in each iteration before the *send_preamble* signal is set. Afterwards, it waits until a rising edge on the *iq_data_valid* signal from the baseband processor indicates the start of the preamble output. Then, it counts until it is reset again, while the timestamps of the above mentioned synchronization events (indicated through *coarse_sync* and *fine_sync* signals from the baseband processor) are saved in registers included in the timestamp counter block. At the end of each iteration, the transfer of the timestamps from the registers to the CHE buffer is controlled by the FSM.

The analog frontend module allows to store the weights for up to 64 beams in an internal memory. The setting to be used is specified by an index register accessible through a serial peripheral interface (SPI) connection. For fast beam switching, the frontend also supports a beam increment functionality, controlled by two dedicated input signals (*beam_reset*, *beam_inc*). A short pulse on the first signal line resets the beam index to the first beam table entry, while a pulse on the second one increments the beam index by one. Thus, the switching from one beam to another is done in less than 40 ns, compared to 300 ns for a SPI register write access.

Algorithm 2 Sensing process implemented in state machine

```

1: switch (fsm_state)
2:   case st_idle:
3:     if start_sensing = 1 then
4:       beam_reset = 1
5:       current_beam = 1
6:       fsm_state = st_prepare_sensing
7:     end if
8:   case st_prepare_sensing:
9:     while current_beam  $\neq$  start_beam do
10:      beam_inc = 1
11:    end while
12:    send_preamble = 1
13:    fsm_state = st_wait_coarse_sync
14:    timeout_cnt_reset = 1
15:   case st_wait_coarse_sync:
16:     if coarse_sync = 1 then
17:       fsm_state = st_wait_fine_sync
18:       timeout_cnt_reset = 1
19:     end if
20:   case st_wait_fine_sync:
21:     if fine_sync = 1 then
22:       fsm_state = st_wait_che
23:       timeout_cnt_reset = 1
24:     end if
25:     if fine_sync_failed = 1 then
26:       next_beam = 1
27:     end if
28:   case st_wait_che:
29:     if che_written = 1 then
30:       next_beam = 1
31:     end if
32:   end switch
33:   if timeout_cnt = 0 or next_beam = 1 then
34:     if current_beam = last_beam then
35:       fsm_state = st_idle
36:       sensing_finished = 1
37:     else
38:       beam_inc = 1
39:       timeout_cnt_reset = 1
40:       send_preamble = 1
41:       fsm_state = st_wait_coarse_sync
42:     end if
43:   end if

```

As already mentioned in Section III-E, three control signals (*send_preamble*, *drop_frame*, *block_fs_rx_if*) were added to the baseband processor to speed up the sensing process. The achieved reduction of the sensing time is presented in Section V-A.

IV. CIR ALIGNMENT METHOD

The RADAR map of the environment is generated by a sequential beam scan and the collection of the CIRs for every beam. For the correct generation of the angular/range map,

the collected CIRs of all beams need to be phase-aligned to the same reference. Assuming a perfect synchronization in the receiver and a known fixed delay in the transmit and receive processing chain, the received CIRs can be easily aligned by just correcting the fixed delay with an appropriate phase shift of the channel coefficients (or a time shift of the CIR itself), as done in [20].

While the used OFDM baseband processor has a fixed processing delay, the included auto-correlation receiver does not provide a perfect synchronization. In case of noisy signals, the frame synchronization signal jitters by several samples. This jitter is partly corrected by the fine synchronization during the channel estimation, but a residual jitter of the fine synchronization pulse is still present. For the communication system using a cyclic-prefix (CP) OFDM modulation, the remaining jitter is not critical. Due to the cyclic prefix, the FFT window has just to be roughly aligned within the CP-extended OFDM symbol. The remaining phase shift is corrected symbol-wise by the examination of the included pilot subcarriers.

In contrast to communication, the remaining jitter of the synchronization is very critical for the presented sensing application. The estimation of the distance of a reflector is done by evaluating the time delay between the transmission start and the appropriate peak in the received channel impulse response (taking the fixed processing delays into account.) Thus, a jitter of the synchronization leads to errors in the distance estimation. This effect is even amplified by the parallel processing. The receiver chain processes eight data samples in parallel, so the synchronization pulse (frame / fine sync, see. Fig. 2) is just provided with a resolution of eight samples. This can result in a range mismatch of around 0.5 m between different CIRs. Since the baseband processor is handled as a closed box within this work, there is no option to change its behavior and to get a sample-synchronous frame sync signal. Furthermore, such an additional signal would not solve the jitter issue of the auto-correlation receiver.

For a proper alignment of the CIRs, the following two-step method is proposed. It is based on an intentionally inserted digital crosstalk as well as the existing self-interference from the antenna crosstalk. First, the delayed overlay mode of the AXI4-Stream crossbar switch (see Sec. III-D) is used. Thus, the non-distorted transmit frame is directly fed into the receiver processing chain. This frame is overlaid with the incoming signal from the receiver (RX) antenna array, making use of the properties of CP-OFDM communication: Any reflection path delay falling in the cyclic prefix period does not distort the frame evaluation. Instead, it can be seen as peak in the estimated CIR. The delay of the overlay mode is optimized so that the synchronizer always synchronizes on the directly fed frame, while the additional time delay of the DACs and ADCs as well as the analog antenna frontend is compensated as much as possible. (I.e. a frame reflected directly at the antenna should have a very small time offset.)

As said, the auto-correlation receiver is sensitive to noise. Therefore, the frame synchronization signal will still jitter in

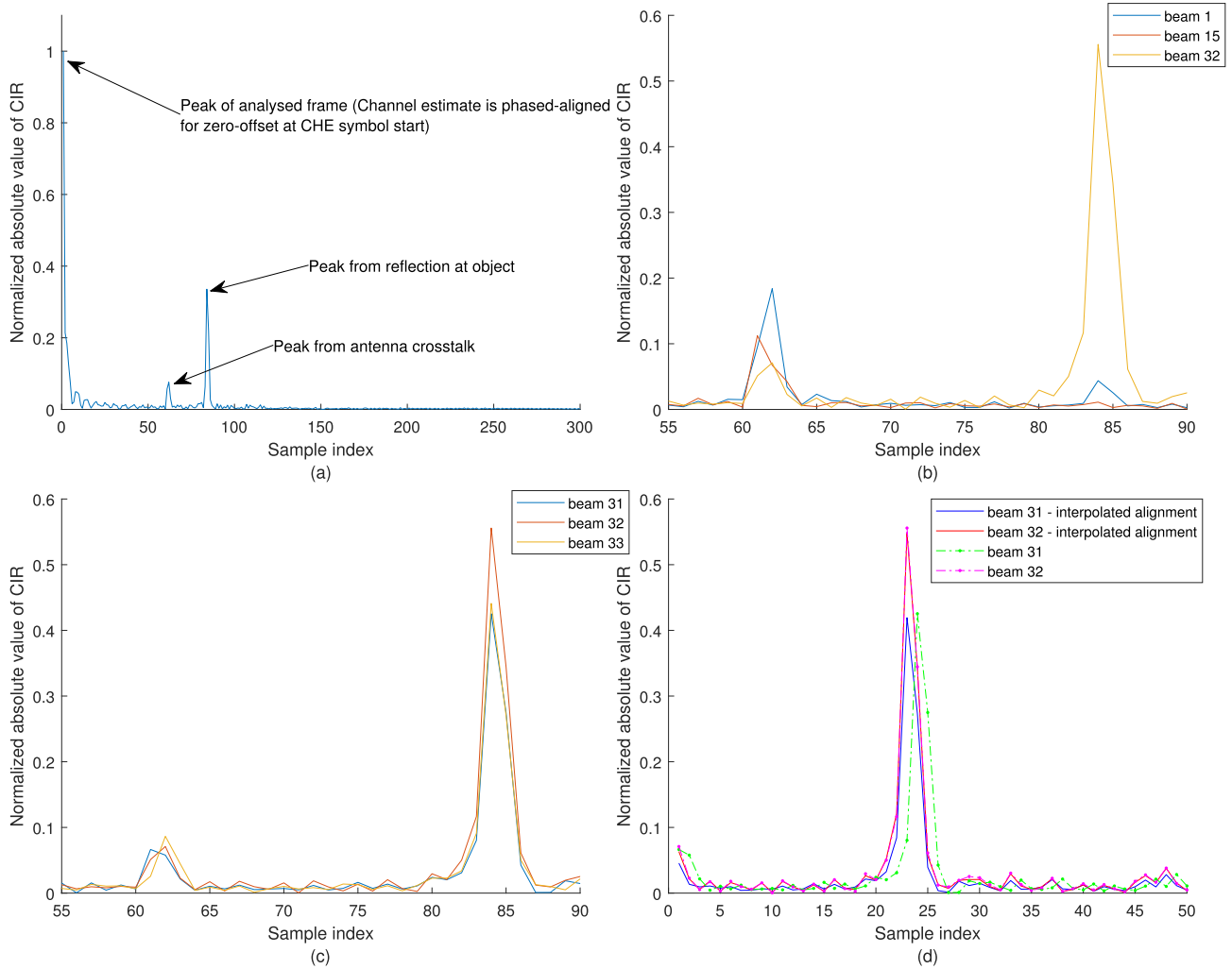


FIGURE 5. (a): Example of a CIR for one beam, indicating the antenna crosstalk and one reflection peak; (b): Cut-out of unaligned CIRs of 3 beams, indicating the reference peak used for alignment on the left and a reflection on the right; (c): Cut-out of unaligned CIRs of 3 neighbored beams with the same reflection; (d): Comparison of CIR alignment without and with interpolation.

the overlay mode, since the noise from the RX antenna array is digitized by the ADCs and overlaid with the fed-through frame. To overcome this, the self-interference of the antenna frontend is used. This self-interference appears as a first peak in the estimated CIR. It is labeled ‘antenna crosstalk’ in Fig. 5a. Since the processing pipeline with the DACs and ADCs as well as the signal traces on the board and the cables to the antenna frontend introduce a fixed delay, the jitter of the antenna self-interference peak is identical to the jitter of the frame synchronization. An example is given in Fig. 5b, where the jittering self-interference peak and one reflection peak is shown for several beams.

For the alignment of the CIRs, the first peak within a defined window is searched and all CIRs are aligned so that this peak occurs at time 0. The search window is defined by the known processing delay and the expected limits of the synchronization jitter, including some margin.

As a side effect, this procedure directly aligns the time axis of the CIR and, therefore, the origin of the range axis to the position of the antenna frontend. Thus, no additional

compensation of the fixed delays is necessary. The range can be directly calculated from the sample index of the peak in the aligned CIR by

$$r = i \cdot \frac{c_0}{2f_s}, \quad (1)$$

where r indicates the range, i the sample index of the peak, f_s the sampling frequency and c_0 the speed of light. The scaling factor of 2 in the denominator results from the doubled path delay to the reflecting object and back.

Although the presented alignment procedure works well, there is an issue with possible sub-sample shifts of the reference peak, as indicated in Fig. 5c. Here, the self-interference peaks of beam 31 and beam 32 are broad, while the reflection from the object is already aligned. A sample shift according to the maximum of the self-interference peak would lead to a one sample difference of the reflection peaks, as shown with the dashed lines in Fig. 5d. This issue is solved by interpolating the CIR with a factor of two before searching the self-interference peak and an appropriate decimation after

TABLE 2. Resource usage of mmWave JCAS transceiver.

Module	LUT	FF	DSP	BRAM	URAM
AXI crossbar	4605	13167	0	0	0
Sensing controller	408	516	0	54	0
SDR module	1548	3600	0	20	80
OFDM baseband	111965	109525	895	251	0
Packet gen/check	228	436	0	0	0
Total	142914	151002	895	325	80

LUT: lookup table, FF: flip-flop, DSP: digital signal processing block, BRAM: block memory, URAM: UltraRAM block

TABLE 3. Timing of OFDM baseband processor signals (without data payload and without noise).

Signal name	Time
Start preamble	0 us
Coarse frame sync	2.848 us
Fine sync	5.789 us
CHE write start	6.498 us
CHE write end	6.998 us
Start Signal field demapping	8.679 us
Signal field decoded	12.158 us
Receiver idle	12.496 us

the alignment. The resulting aligned CIRs are also included in Fig. 5d with the solid lines. Please note that the two curves of beam 32 are on top of each other, proving that the interpolation, shifting and decimation does not change the CIR.

V. PERFORMANCE EVALUATION

In this section, the performance of the presented real-time mmWave JCAS system is evaluated. The focus is on the sensing functionality and system integration aspects. There are already a number of publications evaluating the performance of the used OFDM baseband processor in different communication-only scenarios. A complete real-time mmWave communication system including a MAC processor targeting machine vision applications is evaluated and measured in [29]. There, a detailed data throughput and packet error rate analysis is given, while a different analog frontend is used. An outdoor deployment showcasing a mmWave communication link with an average throughput of 936 Mbit/s over a distance of 42 m is presented in [30], using a single Siivers BFM06005 module. Both works use the same OFDM baseband processor on a different FPGA board and without the sensing functionality.

The performance evaluation starts with a system analysis in terms of resource usage, timing, feasible sensing rates and object velocity limits as well as necessary data rates for the transfer of sensing data in subsection V-A. In the following subsection V-B, the setup used for measurements is explained and measurement results are presented. Finally, potential improvements are discussed in subsection V-C.

TABLE 4. Sensing performance indicators without and with OFDM baseband modifications.

Performance indicator	without modification	with modification
Sensing slot duration	819 us	472.5 us
Sensing slot duration	819 us	472.5 us
Max. sensing rate	1221 Hz	2116 Hz
Angular range / angular scanning resolution	$\pm 45^\circ / 1.4516^\circ$	$\pm 45^\circ / 1.4516^\circ$
Max. distance / resolution	47.2 m / 6.7 cm	47.2 m / 6.7 cm
Max. radial velocity	2577 m/s	4466 m/s
Max. lateral velocity	1007 m/s	1745 m/s
Necessary transfer rate		
at max. sensing rate	1.91 Gbit/s	3.31 Gbit/s
at 100 Hz sensing rate	156.5 Mbit/s	156.5 Mbit/s
Throughput loss		
at 100 Hz sensing rate	8.2 %	4.7 %
and including data transfer	19.8 %	16.3 %

A. SYSTEM ANALYSIS

The used FPGA resources for the presented modules and the complete system are shown in Table 2. Not listed in detail is the glue logic necessary for controlling the RF data converters and for the AXI interface to the processing system, which adds around 24k lookup tables and flip-flops to the design.

The timing of the baseband processing for a frame without data payload is presented in Table 3. It is obtained from a clock-cycle accurate digital simulation with structural simulation models and reflects the real system behavior. For the simulation, a direct digital connection between the transmitter and the receiver part was used (XBAR loopback mode without delay). Since the frame does not contain data payload, the baseband processor returns to its idle state after decoding of the signal field. The assertion of the *start_preamble* signal was set at time 0. As can be seen, the signal field decoding adds a huge delay, while the CHEs are already available. Without the modifications of the OFDM baseband processor described in Section III-E, one sensing beam scan over 63 beams would take $12.5 \text{ us} \cdot 63 = 787.5 \text{ us}$. With the modifications, the necessary time is reduced by 44 % to $7 \text{ us} \cdot 63 = 441 \text{ us}$. To take the delay of the data converters and the on-board signal traces into account, a 500 ns wait cycle is added to each transmission. The duration of one full beam scan with and without the OFDM baseband processor modification and the resulting sensing rates are shown in Table 4.

The timing analysis also shows that the usage of communication frames for sensing would increase the required channel coherency time, i.e. the time in which the environment is assumed to be static. The required channel coherency time will increase not only when communication frames are used in the sensing slot, but also when the sensing is done just with preambles, while communication beams are omitted during the beam scan. For the following discussion, the MAC protocol is neglected and it is assumed that the communication beams are directly switched after sending

the communication frame. Furthermore, the timing of the modified baseband processor is used. Since the CIRs of the communication beams are already known, these beams might be omitted during the beam scan. Assuming five different communication beams, this would reduce the time for a full scan by roughly $5/63 \approx 7.9\%$. This corresponds to 435 us. But as shown in Table 1, a frame with data payload is much longer than a pure preamble (e.g. 16.9 us for 2 kB payload, compared to ≈ 4 us for a preamble). Thus, the required channel coherency time will increase a lot. For the assumed setup with five communication beams and just including the frame duration, the required channel coherency time would increase by at least 10 % to 519.5 us (435 us + 5 · 16.9 us).

To not increase the required channel coherency time (either by using communication frames during sensing or by reusing CIRs from the communication slot and omitting the corresponding beam), the duration of a communication frame has to be shorter than the time for a single sensing in one direction (i.e. 7 us). The preamble and the signal field symbol have a length of ≈ 4.3 us. Each data symbol adds ≈ 0.57 us (see Table 1). Thus, the communication frame could only contain four data symbols, corresponding to a maximum data payload of 384 bytes in case of QPSK modulation. This would severely decrease the data throughput, due to the PHY overhead of more than 60 %. In addition, this analysis does not take the overhead of the medium access control (MAC) protocol with acknowledgments and the corresponding latency of the OFDM payload processing into account. Eventually, the required channel coherency time would increase much more, since the beam cannot be switched directly after sending a communication frame.

Table 4 also summarizes the angular and range resolution as well as their limits. The angular scan range and the angular scanning resolution are defined by the characteristics of the used analog antenna frontend. The beambook defines 63 evenly spaced beams with 6° horizontal beam width in a range of $\pm 45^\circ$, resulting in an angular scanning resolution of $\approx 1.5^\circ$. The range resolution is determined by the sampling frequency and can be derived from (1) by setting i to 1. The maximum distance for an unambiguous detection is determined by the length of the cyclic prefix l_{cp} . For the channel estimation symbols, the cyclic prefix consists of 768 samples, resulting in a maximum distance of

$$d_{max} = \frac{(l_{cp} - l_{off})}{f_s} \cdot \frac{c_0}{2} = 47.2 \text{ m}, \quad (2)$$

where c_0 determines the speed of light, f_s the sampling frequency and l_{off} the worst case sample index of the antenna crosstalk peak. In practise, the maximum achievable distance is additionally limited by the RADAR cross-section of the reflectors and the channel smoothing filter of the implemented OFDM baseband processor (see Sec. V-B).

Due to the beam scan procedure, moving objects should not change their detectable position given by the range resolution and angular scanning resolution within the time needed for one transmission. This leads to the maximum velocities given

in Table 4, distinguishing both cases: moving directly to and from the transceiver and moving in parallel to the antenna plane. For the latter case, the velocity is given for a distance of 1 m. Increasing the distance would lead to higher possible lateral velocities. The maximum velocities are calculated by

$$v_{rmax} = \frac{r_r}{2 \cdot t_{bs}} \quad (3)$$

$$v_{lmax} = \frac{d \cdot \tan(r_a)}{2 \cdot t_{bs}} \quad (4)$$

where v_{rmax} denotes the maximum radial velocity, v_{lmax} the maximum lateral velocity, t_{bs} the duration of a single sensing in one beam direction (i.e. 13 us resp. 7.5 us), d the distance to the object, r_r the range resolution and r_a the angular scanning resolution. The factor two in the denominator ensures that the object moves less than half of the sensing resolution grid. Please note that this analysis is only valid for a sequential beam scan. For out-of-order scans, e.g. omitting communication beams during the beam scan and reusing their CIRs, moving objects should not change their position from the transmission start of the reused communication frame until the end of the whole sensing.

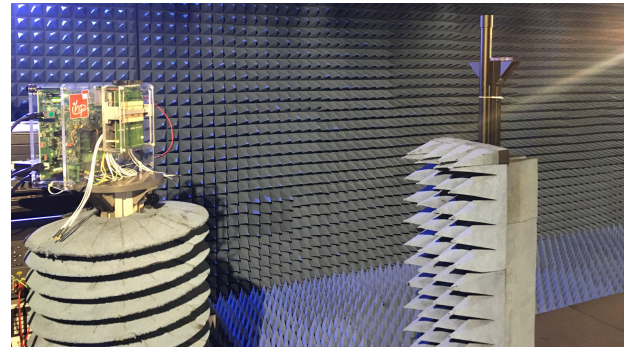


FIGURE 6. Measurement setup in the anechoic chamber with the JCAS transceiver station on the left and the reflecting metal pipe mounted on the linear positioner on the right.

In a real application scenario, the maximum sensing rate would not be used, since the medium access time slot for communication would be reduced to zero. For a typical sensing rate of 100 Hz, the throughput loss is given in Table 4. For a sensing rate below 10 Hz, the throughput loss is negligible.

Each channel coefficient for the 768 data subcarriers and 60 pilot subcarriers is provided with 18 bit amplitude resolution and 12 bit phase resolution. As such, $30 \text{ bits} \cdot (768 + 60) \cdot 63 = 1564920$ bits need to be readout and transferred to the PC after a full beam scan. This results in a necessary sensing data transfer rate of up to 3.3 Gbit/s for the maximum achievable sensing rate of 2116 Hz. For a more realistic sensing rate of 100 Hz, the necessary transfer rate is reduced to 156.5 Mbit/s. Assuming the OFDM baseband is used to transfer the sensing data, an additional throughput loss of 11.6 % would occur (with QPSK modulation). This additional throughput loss for the sensing data transfer indicates that the processing of the sensing data should be either done on the

sensing station or the sensing rate should be reduced to 10 Hz and below. In the latter case, the total throughput loss will be around 2 %.

B. MEASUREMENT SETUP AND RESULTS

As mentioned at the beginning of Section V, the communication performance of the used OFDM baseband processor was already evaluated and measured in different publications. Therefore, the current measurement focuses on the sensing functionality. Especially, the derived resolution from Subsection V-A is verified and the impact of the CIR alignment on the sensing accuracy is evaluated. Furthermore, the maximum sensing distance is examined.

For the measurements of the sensing accuracy, one transceiver station is used for mono-static RADAR sensing in an anechoic chamber. It is mounted on a rotational positioner. Three different objects are successively used as reflecting targets: A metal bracket with dimensions of $5 \times 5 \times 5$ cm, a vertically oriented metal pipe with 35 mm diameter and a RADAR corner reflector. They are mounted on a linear positioner which allows to vary the distance between the transceiver and the reflector in a range of 0 to around 4.4 m. An impression of the setup using the metal pipe reflector is shown in Fig. 6. The pole of the linear positioner is made of plastic, so the metal pipe was used to intensify the reflection.

For every selected combination of angle and distance, 100 full beam scans are performed and the obtained CHEs of every scan are individually processed. The results in terms of the mean estimated distance and mean estimated beam angle as well as their standard deviation are shown in Table 5. For estimating the distance and beam angle, the maximum peak of the CIR's is used. Furthermore, the table shows the difference between the estimated and real values.

In most cases, the standard deviation for the distance estimation is zero, meaning that the individual estimations can be repeated with a very high precision. The accuracy is also high, taking the possible resolution of 6.67 cm into account. For the angle estimation with a possible scanning resolution of 1.5° , it looks a little different. The standard deviation is up to 1.7° and the angle estimation error reaches 5.3° . Here, it must be considered that the beam scan is done in steps of 1.5° , but the 3 dB beam width is 6° . Extended objects are therefore detected in a wide angular range, but only one beam is selected by the peak search evaluation. And this might not be the beam pointing directly to the center of the object, since the gain of the beams decreases with an increasing angle. Thus, a shifted beam might result in a higher received energy. In future work, the beam pattern has to be included in the analysis for the CIR's.

Comparing the results from Table 5 for both CIR alignment methods, it seems that the method without interpolation outperforms the alignment with interpolation in terms of range and angle estimation accuracy (e.g. last row for object B). But it must be considered that the interpolation was not done to increase the resolution or detection accuracy. Instead, it is done to avoid errors during the alignment of

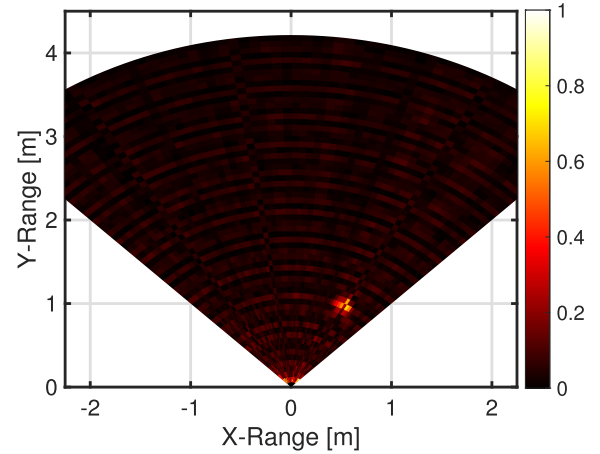


FIGURE 7. Range-angular RADAR heatmap for metal pipe without CIR interpolation.

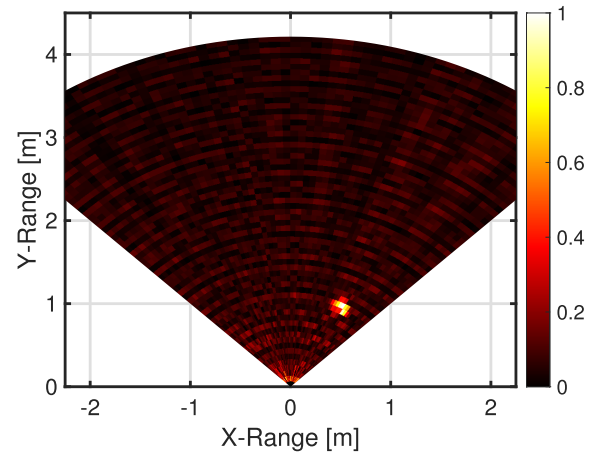


FIGURE 8. Range-angular RADAR heatmap for metal pipe with CIR interpolation.

CIRs from different beams, where a broad self-interference peak might result in a misalignment. (see Sec. IV). Thus, the interpolation during the CIR alignment is necessary to correctly estimate the shape of extended objects. An example is visualized in Fig. 7 and 8. Here, the metal pipe was placed at a distance of 1.10 m and at an angle of 30.9° (third line for object B in Table 5). As can be seen, the maximum reflection peak is broad in Fig. 8, leading to the increased error with the simple peak search for object localization. But in Fig. 7, it can be seen that the CIR of one beam is shifted by one sample, leading to a distorted shape of the object. When detecting extended objects, this is more critical, since it cannot be compensated by including the shape of the antenna pattern.

To evaluate the angular scanning resolution and the multi-target separation, two RADAR corner reflectors with an outer edge length of 6 cm were mounted on the linear positioner, which was placed at a distance of 2.9 m from the JCAS transceiver. The distance between the reflectors is 15.6 cm (center to center). This corresponds to an angular spacing of 3° . Even though the distance between both objects is smaller

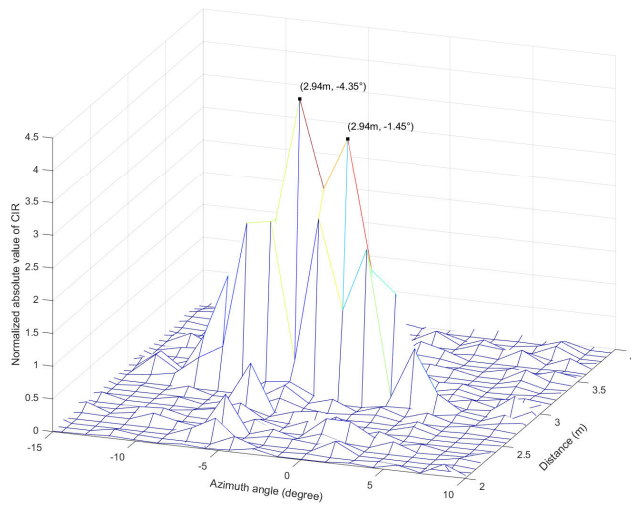


FIGURE 9. CIR vs. range and angle for two small objects with 3° spacing in a distance of 2.9 m.

than the beam width, the objects can be clearly identified in the CIR plot in Fig. 9. This plot shows the raw data of the acquired CIRs for every beam setting, after subtraction of the background (i.e. the scan result without mounted reflectors).

In addition to the detailed evaluation of the sensing performance in an anechoic chamber, measurements in a large entrance hall (see Fig. 10) have been conducted. During these measurements, a RADAR corner reflector with an edge size of 21 cm was detected at a distance of up to 17 m. Fig. 11 shows the CIRs for beam 32 (0° beam direction) for different reflector positions. For each position, 20 measurements were averaged and the averaged background was removed. As can be seen, the reduction of the peak amplitudes is much higher than what is expected from the path loss. For reflector distances above 17 m, no peak could be identified.

The observable amplitude drop and the limited sensing range result from the channel smoothing filter implemented in the real-time baseband processor's channel estimation module. It filters out highly delayed components of the channel impulse response. While this improves the channel estimation SNR [23], it limits the maximum range during the sensing operation. Thus, the measurements were repeated with a non-real-time SDR implementation of the OFDM baseband processing. In this software implementation, the channel smoothing filter was disabled. Due to the SDR mode without a digital loopback, l_{off} (see (2)) is set to zero. All other settings were kept the same. With this setup, the reflector could be detected at a distance of up to 51 m. This corresponds to the calculated maximum unambiguous range of the SDR implementation. Thus, it can be concluded that the derived maximum sensing distance for the real-time system of 47.2 m (see Table 4) can be realized with a small change in the implemented OFDM baseband processor: For future usage in a JCAS application, it should be possible to switch off the channel smoothing filter during the sensing process.



FIGURE 10. Measurement setup with up to 60 m distance (currently 10 m) between the JCAS transceiver and a RADAR corner reflector in an entrance hall.

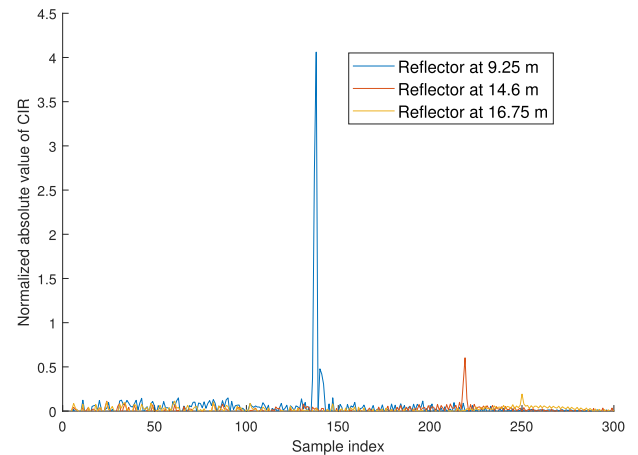


FIGURE 11. CIRs for 0° beam direction with different reflector positions after background removal.

C. POTENTIAL IMPROVEMENTS

A good opportunity to reduce the time necessary for sensing - and, therefore, to either increase the sensing rate or to reduce the throughput loss - is a hierarchical beam scan. The 3 dB beam width of the antenna frontend is 6°, while a 1.5° step size is used for scanning 63 beams. For many application cases, a scan with 6° step size would be sufficient. This reduces the scanning time by a factor of 4. If a higher resolution is needed, it could be restricted to an area of interest, e.g. to a small number of beams in the region an object was detected using the coarse resolution. The sensing controller already supports the definition of a start beam index and a stop beam index. Thus, using only 16 beams could be implemented with a modified beam table in the analog frontend. But flexible beam scanning, meaning that beams can be omitted in an adaptive way, would need some changes in the FSM of the sensing controller. A reprogramming of the analog frontend with an adapted beam table for every beam scan would take too long due to the relatively slow SPI connection.

TABLE 5. Measurement results of the sensing evaluation in the anechoic chamber with different reflecting objects.

Obj.	d_r	d_i	σ_{di}	d_n	σ_{dn}	$ d_r - d_i $	$ d_r - d_n $	a_r	a_i	σ_{ai}	a_n	σ_{an}	$ a_r - a_i $	$ a_r - a_n $
A	1.08	1.07	0.0	1.07	0.0	0.02	0.02	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	1.08	1.07	0.0	1.07	0.0	0.02	0.02	-5.0	-5.8	0.2	-4.4	0.0	0.8	0.7
	1.08	1.07	0.0	1.12	0.03	0.02	0.04	-10.1	-7.4	0.5	-8.3	0.6	2.6	1.7
	1.08	1.07	0.0	1.09	0.03	0.02	0.01	-12.0	-13.1	0.0	-11.7	2.0	1.0	0.3
	1.88	1.94	0.0	1.94	0.0	0.05	0.05	0.0	0.5	1.7	-2.8	0.7	0.5	2.7
	2.06	2.01	0.0	2.01	0.0	0.05	0.05	0.0	-0.2	0.8	0.0	0.0	0.2	0.0
	2.14	2.07	0.0	2.07	0.0	0.06	0.06	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	2.20	2.14	0.0	2.14	0.0	0.06	0.06	0.0	0.1	0.3	0.0	0.0	0.1	0.0
B	1.10	1.07	0.0	1.07	0.0	0.03	0.03	-45.0	-40.6	0.0	-40.7	0.1	4.3	4.3
	1.10	1.07	0.0	1.07	0.0	0.03	0.03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	1.10	1.05	0.03	1.07	0.01	0.04	0.02	30.9	27.7	0.5	29.0	0.01	3.3	1.9
	1.10	1.05	0.03	1.07	0.0	0.05	0.03	33.0	27.8	0.5	29.0	0.0	5.3	4.0
	1.10	1.07	0.0	1.07	0.0	0.03	0.03	0.1	0.0	0.0	-1.0	1.4	0.1	1.0
	1.16	1.14	0.0	1.14	0.0	0.03	0.03	0.1	0.0	0.0	0.0	0.0	0.1	0.1
	1.23	1.20	0.0	1.20	0.0	0.03	0.03	0.1	0.0	0.0	0.0	0.0	0.1	0.1
	1.31	1.27	0.0	1.27	0.0	0.04	0.04	0.1	0.0	0.0	0.0	0.0	0.1	0.1
	1.31	1.27	0.01	1.27	0.02	0.03	0.03	-9.1	-7.3	0.2	-7.3	0.3	1.7	1.7
	1.31	1.27	0.0	1.27	0.0	0.04	0.04	-11.9	-13.1	0.0	-13.1	0.0	1.1	1.1
	2.44	2.34	0.0	2.41	0.0	0.09	0.03	8.9	6.7	0.7	6.5	0.7	2.2	2.4
	1.30	1.34	0.0	1.34	0.0	0.04	0.04	0.1	0.0	0.2	0.0	0.0	0.1	0.1
C	1.30	1.34	0.0	1.34	0.0	0.04	0.04	1.5	0.0	0.0	0.0	0.0	1.5	1.5
	1.30	1.34	0.0	1.34	0.0	0.04	0.04	3.0	1.5	0.0	1.5	0.0	1.5	1.5
	1.30	1.34	0.0	1.34	0.0	0.04	0.04	4.5	1.5	0.0	4.4	0.0	3.1	0.2
	1.30	1.34	0.0	1.34	0.0	0.04	0.04	6.0	4.4	0.0	4.4	0.0	1.6	1.6
	1.37	1.41	0.0	1.41	0.0	0.03	0.03	0.1	0.0	0.0	0.0	0.0	0.1	0.1
	1.97	2.01	0.0	2.01	0.0	0.04	0.04	24.6	21.8	0.0	21.8	0.0	2.8	2.8
	2.99	3.01	0.0	3.01	0.0	0.02	0.02	7.6	7.3	0.0	7.3	0.0	0.3	0.3
	3.50	3.55	0.0	3.55	0.0	0.05	0.05	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	4.16	4.21	0.0	4.21	0.0	0.06	0.06	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	1.30	1.34	0.0	1.34	0.0	0.04	0.04	0.1	0.0	0.2	0.0	0.0	0.1	0.1

Obj. = object, A: metal bracket, B: metal pipe, C: corner reflector, d : distance, a , σ : standard deviation; Indices: r: true, i: using interpolation for CIR alignment, n: using no interpolation for CIR alignment. All distances are given in meters (m) and all angles in degrees (°).

The current limitation of the maximum sensing distance (see Section V-B) could be avoided by switching off the channel smoothing filter during the sensing or by selecting other filter coefficients for the sensing operation.

As discussed in Section III-F, the chosen buffer structure with 128 rows in each of the 64 buffers leads to a calculated memory overhead of 23 %. In summary, 54 BRAM tiles are used to implement the buffer structure (see Table 2). The FPGA provides BRAM primitives with fixed sizes of either 1Kx36, 2Kx18 and 4Kx9. A reduction of the memory structure to 105 rows for each buffer would reduce the number of used primitives to 48. This corresponds to an overhead of 11.1 % for the current implementation. This overhead can be reduced by a changed buffer addressing scheme.

Another potential improvement is the support of bi-static sensing. This would require a small change in the sensing controller. In bi-static sensing, one station should either act as transmitter or as receiver, so these individual modes need to be supported by the state machine. Together with this change, the buffer structure to store CHEs could be extended to application cases like PHY layer security. In addition to these changes, bi-static sensing would require a very good timing synchronization of the separated TX and RX stations. This can be realized by using state-of-the-art methods for precise timing synchronization like *White Rabbit* [31].

Finally, the hardware-implemented 1024-point FFT module of the baseband processor might be used for performing the inverse FFT on the channel coefficients. This would significantly decrease the time for software-based post-processing of the CHEs on the PC or an edge server. Currently, the PC needs 1.5 ms for calculating all 63 FFTs, so only sensing rates up to 600 Hz are feasible. Furthermore, a hardware-based FFT accelerator is required to be able to do some additional post-processing like the CIR alignment on the PS of JCAS transceiver. The PS itself has not enough performance for software-based FFT processing. A single 1024 point FFT on one ARM core with a common C FFT library needs 0.5 ms. For processing the results of 63 beams by using all 4 ARM cores in parallel, 8 ms are needed. This will only allow sensing rates much lower than 100 Hz, since the PS is also needed for the handling of the network API and the control of the system. By using a hardware-based FFT-processing and a CIR alignment on the PS, the maximum sensing rates of the PHY are supported.

The post-processing of the generated CIRs on the sensing node also reduces the amount of data to be transferred by just transmitting identified peaks or objects, e.g. the range and angle of a peak above a certain threshold. The background modeling and clutter removal could also be realized on the PS. Nevertheless, the PS has limited computing capabilities

compared to an edge server. Thus, it is beneficial to change the currently used Ethernet interface with 1 Gbit/s data rate to a 10 Gbit/s or 100 Gbit/s version, to be able to send the raw data. In addition, sending the CIR data in a user datagram protocol (UDP) stream instead of the currently used TCP stream may also improve the throughput on the interface to the edge server.

VI. CONCLUSION AND OUTLOOK

The presented real-time mmWave JCAS system fulfills the design constraints defined in Section II, mainly the addition of a RADAR sensing functionality to a communication system without changing the hardware implementation of the baseband processing. It was shown that the hardware-implemented sensing controller adds only little overhead in terms of hardware resources and medium occupancy, while the system delivers a very good sensing performance.

During the implementation of a JCAS system, several challenges might occur which were not foreseen during the system design in a simulation environment. One implementation challenge is the misalignment of the CIRs due to the jitter of the auto-correlation receiver and the parallel processing of eight samples. An easy-to-implement method for aligning the CIRs of several beams was presented. This method does not require changes in the baseband processing, e.g. a sample-index counter or a sample-accurate frame synchronization signal. Instead, the existing self-interference from the full-duplex antenna frontend is exploited, together with a digital loop-back of the sent frame.

Besides the potential improvements discussed in Section V-C, our future work includes the integration of a medium access control processor to realize a whole JCAS transceiver and to test the JCAS functionality in a wireless network of several stations. Further work will be dedicated to the realization of the mentioned predictive beam-steering use case. Finally, the mmWave JCAS system shall be integrated in a 6G end-to-end application demonstration developed within the Open6G-Hub project [32].

ACKNOWLEDGMENT

The authors acknowledge the support of their colleagues Jens Lehmann, Jörg Domke, Jürgen Berthold, and Darko Cvetkovski in the design and assembly of the RF boards, the assembly of the transceiver stations and in the preparation of the measurement campaign.

REFERENCES

- [1] T. Wild, V. Braun, and H. Viswanathan, "Joint design of communication and sensing for beyond 5G and 6G systems," *IEEE Access*, vol. 9, pp. 30845–30857, 2021.
- [2] J. A. Zhang, Md. L. Rahman, K. Wu, X. Huang, Y. J. Guo, S. Chen, and J. Yuan, "Enabling joint communication and radar sensing in mobile networks—A survey," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 1, pp. 306–345, 1st Quart., 2022.
- [3] J. Lee, A. A. Badrudeen, and S. Kim, "6G integrated sensing and communication: Recent results and future directions," in *Proc. 13th Int. Conf. Inf. Commun. Technol. Conver. (ICTC)*, Oct. 2022, pp. 1219–1221.
- [4] D. K. P. Tan, J. He, Y. Li, A. Bayesteh, Y. Chen, P. Zhu, and W. Tong, "Integrated sensing and communication in 6G: Motivations, use cases, requirements, challenges and future directions," in *Proc. 1st IEEE Int. Online Symp. Joint Commun. Sens. (JC&S)*, Feb. 2021, pp. 1–6.
- [5] C. Xie, S. Ma, and B. Zhou, "Environment mapping based on mmWave MIMO OFDM communication systems towards 6G integrated communication and sensing," in *Proc. IEEE 11th Int. Conf. Inf. Commun. Netw. (ICIN)*, Aug. 2023, pp. 194–200.
- [6] F. Liu, C. Masouros, A. P. Petropulu, H. Griffiths, and L. Hanzo, "Joint radar and communication design: Applications, state-of-the-art, and the road ahead," *IEEE Trans. Commun.*, vol. 68, no. 6, pp. 3834–3862, Jun. 2020.
- [7] D. Tagliaferri, M. Mizmizi, S. Mura, F. Linsalata, D. Scazzoli, D. Badini, M. Magarini, and U. Spagnolini, "Integrated sensing and communication system via dual-domain waveform superposition," *IEEE Trans. Wireless Commun.*, vol. 23, no. 5, pp. 4284–4299, May 2024.
- [8] J. Zhu and J. Yang, "Design and implementation of integrated sensing and communication waveform based on LFM-CPM," in *Proc. Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Nov. 2023, pp. 438–442.
- [9] W. Zhou, R. Zhang, G. Chen, and W. Wu, "Integrated sensing and communication waveform design: A survey," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 1930–1949, 2022.
- [10] J. A. Zhang, F. Liu, C. Masouros, R. W. Heath Jr., Z. Feng, L. Zheng, and A. Petropulu, "An overview of signal processing techniques for joint communication and radar sensing," *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 6, pp. 1295–1315, Nov. 2021.
- [11] C. Zhang, Z. Zhou, H. Wang, and Y. Zeng, "Integrated super-resolution sensing and communication with 5G NR waveform: Signal processing with uneven CPs and experiments: (Invited paper)," in *Proc. 21st Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, Aug. 2023, pp. 681–688.
- [12] J. Pegoraro, J. O. Lacruz, M. Rossi, and J. Widmer, "HiSAC: High-resolution sensing with multiband communication signals," in *Proc. 22nd ACM Conf. Embedded Netw. Sensor Syst.*, Nov. 2024, pp. 549–563.
- [13] Z. Zhou, C. Zhang, and Y. Zeng, "Prototype of real-time integrated sensing and communication with millimeter-Wave OFDM," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Aug. 2023, pp. 1–2.
- [14] P. Kumari, A. Mezghani, and R. W. Heath Jr., "JCR70: A low-complexity millimeter-Wave proof-of-concept platform for a fully-digital SIMO joint communication-radar," *IEEE Open J. Veh. Technol.*, vol. 2, pp. 218–234, 2021.
- [15] M. Neu, C. Karle, B. Nuß, P. Groeschel, and J. Becker, "A scalable and cost-efficient antenna testbed using FPGA-server compound structures for prototyping 6G applications," in *Proc. 19th Int. Conf. Distrib. Comput. Smart Syst. Internet Things (DCOSS-IoT)*, Jun. 2023, pp. 171–178.
- [16] M. Engelhardt, S. Giehl, M. Schubert, A. Ihlow, C. Schneider, A. Ebert, M. Landmann, G. Del Galdo, and C. Andrich, "Accelerating innovation in 6G research: Real-time capable SDR system architecture for rapid prototyping," 2024, *arXiv:2402.06520*.
- [17] V. Shatov, B. Nuss, S. Schieler, P. K. Bishoyi, L. Wimmer, M. Lübke, N. Keshtiarast, C. Fischer, D. Lindenschmitt, B. Geiger, R. Thomä, A. Fellan, L. Schmalen, M. Petrova, H. D. Schotten, and N. Franchi, "Joint radar and communications: Architectures, use cases, aspects of radio access, signal processing, and hardware," *IEEE Access*, vol. 12, pp. 47888–47914, 2024.
- [18] A. Ichkov, A. Schott, N. Beckmann, and L. Simić, "Multi-band mm-Wave measurement platform towards environment-aware beam management," 2024, *arXiv:2405.00714*.
- [19] J. Guan, A. Paidimarri, A. Valdes-Garcia, and B. Sadhu, "3-D imaging using millimeter-Wave 5G signal reflections," *IEEE Trans. Microw. Theory Techn.*, vol. 69, no. 6, pp. 2936–2948, Jun. 2021.
- [20] N. Maletic, M. Petri, M. Appel, and E. Grass, "A software-defined radio solution for integrated mmWave communication and sensing," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2025, pp. 1–6.
- [21] AMD. *Zynq UltraScale+ RFSoC ZCU111 Evaluation Kit*. Accessed: Jun. 16, 2024. [Online]. Available: <https://www.xilinx.com/products/boards-and-kits/zcu111.html>
- [22] Siivers Semiconductors AB. *BFM06010 Data Sheet*. Accessed: Jul. 16, 2024. [Online]. Available: <https://www.sivers-semiconductors.com/5g-millimeterwave-mmwave-and-satcom/wireless-products/RF-modules/bfm06010-and-bfm06011>

- [23] M. Piz, E. Grass, M. Marinkovic, and R. Kraemer, "Next-generation wireless OFDM system for 60-GHz short-range communication at a data rate of 2.6 GBit/s," presented at 14th Int. OFDM-Workshop, 2009.
- [24] M. Petri, "Configurable, modular and scalable OFDM baseband processor for data rates up to 4 GBPS," in *Wireless Communications*. Calgary, AB, Canada: ACTA Press, Jun. 2011.
- [25] AMD. *AMD Zynq UltraScale RFSoc*. Accessed: Jul. 16, 2024. [Online]. Available: <https://www.amd.com/de/products/adaptive-socs-and-fpgas/soc/zynq-ultrascale-plus-rfsoc.html>
- [26] ARM. *AMBA AXI-Stream Protocol Specification*. Accessed: Aug. 2, 2024. [Online]. Available: <https://developer.arm.com/documentation/ih0051/latest/>
- [27] AMD. *AXI General Purpose IO IP-Block*. Accessed: Aug. 2, 2024. [Online]. Available: https://www.xilinx.com/products/intellectual-property/axi_gpio.html
- [28] *Amendment: Enhancements for Very High Throughput in the 60GHz Band*, Standard 802.11ad-2012, 2012. [Online]. Available: <http://standards.ieee.org>
- [29] M. Ehrig and M. Petri, "60 GHz broadband MAC system design for cable replacement in machine vision applications," *AEU-Int. J. Electron. Commun.*, vol. 67, no. 12, pp. 1118–1128, Dec. 2013.
- [30] N. Maletić, M. Ehrig, D. Cvetkovski, J. Gutiérrez, E. Grass, M. Krstić, and M. Petri, "SDR-based 60 GHz solution for mmWave applications: Implementation and evaluation," in *Proc. 30th Telecommun. Forum (TELFOR)*, Nov. 2022, pp. 1–4.
- [31] J. E. Gilligan, E. M. Konitzer, E. Siman-Tov, J. W. Zobel, and E. J. Adles, "White rabbit time and frequency transfer over wireless millimeter-Wave carriers," *IEEE Trans. Ultrason., Ferroelectr., Freq. Control*, vol. 67, no. 9, pp. 1946–1952, Sep. 2020.
- [32] Open6G-Hub project. *OpenLabs Und Experimentierfelder-6G Für Mensch, Umwelt & Gesellschaft*. Accessed: Jul. 16, 2024. [Online]. Available: <https://www.open6ghub.de/project/openlabs-und-experimentierfelder/>



MARKUS PETRI received the Dipl.-Ing. (M.S.E.E.) degree from TU Berlin, Germany, in 2006, and the Dr.-Ing. (Ph.D.) degree from BTU Cottbus, Germany, in 2012.

In 2009, he worked in the field of vision sensors for industrial automation. From 2009 to 2020, he was a Scientist with IHP-Leibniz-Institut für innovative Mikroelektronik, Frankfurt (Oder), Germany. From 2020 to 2023, he was the Head of technology transfer of IHP Solutions GmbH,

Germany. He coordinated the Horizon Europe Project COCHISA, until June 2023, and participated in a number of collaborative research projects, such as fastSecure, 5G-XHaul, ProWiLan, Prelocate, and EASY-A. Since 2023,

he has been with IHP, where he is leading IHP's research activities in the BMFTR-funded project Open6G-Hub. Since May 2025, he has been leading IHP's research group "real-time communication and sensing systems." His research interests include joint communication and sensing, appropriate PHY-layer hardware architectures, real-time implementations, and resilient systems.

Dr. Petri is an Awardee of the Erwin-Stephan-Price from the Technical University of Berlin.



NEBOJSA MALETIC (Senior Member, IEEE) received the Dipl.-Ing. degree in electrical engineering (major in microwave engineering) and the M.Sc. degree in electrical engineering and computer science from the University of Belgrade, Serbia, in 2008 and 2010, respectively. Currently, he is pursuing the Ph.D. degree with the Humboldt University of Berlin.

In 2010, he joined the Faculty of Electrical Engineering, University of Banja Luka, Bosnia-Herzegovina, as a Teaching and Research Associate. In 2015, he joined the IHP-Leibniz-Institut für innovative Mikroelektronik, Frankfurt (Oder), Germany, where he is a member of the Wireless Broadband Communications Group. In Summer 2024, he joined Qualcomm, San Diego, CA, USA, for three months as an Engineering Intern. He participated in several European H2020 projects (5G-XHaul, 5G-PICTURE, WORTECS, 5GENESIS, and 5G-VICTORI) related to millimeter waves and their applications. His current research interests include millimeter-wave communications, sensing, MIMO, signal processing algorithms, hardware impairments, and the design of wireless communication systems operating in millimeter-wave and sub-terahertz bands.

Mr. Maletic is a member of the IEEE ComSoc and MTT-S societies.

...