Hardware and Software Co-design of a Single-Carrier Modulation (SCM) Acoustic Communication System

Jinfeng Li, and Y. Rosa Zheng, *Fellow, IEEE*
Dept. of ECE, Lehigh University, Bethlehem, PA 18015-3084, USA.

***Abstract*—Most of the recent software-defined underwater acoustic modems are based on the commercial SDR platform. The implementation of the physical layer is either solely software-defined with high processing latency or partially hardware-defined but without adequate reconfiguration capabilities. This paper demonstrates a flexible implementation of a high-frequency acoustic transceiver with Single-Carrier Modulation (SCM) in contrast to the common orthogonal frequency division multiplexing (OFDM). We explore the hardware and software co-design based on the System on a Chip (SoC) architecture consisting of ARM processors and a Field Programmable Gate Array (FPGA). The partition of the data processing modules on the processing system (PS) and programmable logic (PL) is discussed at both the transmit and receive chains. The real-time reconfiguration capability on most of the processing modules is also described. Different data processing architectures as well as the real-time reconfiguration capabilities of the proposed design have been evaluated in the lake environments.**

I. Introduction

The development of reconfigurable, reprogrammable, Software-defined acoustic modems (SDAMs) attracts lots of interest due to the requirement of flexibility in the scientific experiments on underwater communications and networks [1]. Most of the recent work incorporates the hardware and software design based on the commercial software-defined radio (SDR)-based platform, such as the latest Universal Software Radio Peripheral (USRP) products that based on an FPGA and a CPU (host PC) [2]–[5] , and the Texas Instruments OMAP series that contain an ARM processor and a DSP [6], [7], while several works developed a platform based on the system on a chip (SoC) architecture integrating the FPGA and ARM in one chip [8]–[10].

The work in [2]–[4] and the work in [5] are based on USRP N210 and X-300, respectively. The FPGA is only responsible for the interpolation/decimation of the stream to match the fixed hardware sampling rate (100 MSPS) to the rate requested by the user. The data processing is mainly done in the host PC with high processing latency. [3] discussed the real-time reconfigurable modem based on the USRP N210, where in the experiments, the decision of the incorporation of a 1/2-rate error correction coding and the selection between QPSK and BPSK is made in real-time based on the SINR.

An SDAM based on the OMAPL138 which contains an ARM and a DSP is implemented in [6]. The DSP core in that work is responsible for the data processing while the ARM takes charge of the information interaction with external devices. The configuration parameters in the transmitter include coding types, mapping types, etc. The options for the coding scheme are Reed-Solomon (RS) codes and convolution. The support modulation schemes are BPSK or QPSK.

Different from the SDR-based platform containing two individual components, the SoC architecture provides more flexibility in the hardware and software co-design with the convenient and high-speed data exchange between the ARM and FPGA. The ”FLUMO“ developed by [8] is based on a RedPitaya board mounted with a Xilinx Zynq 7010 SoC. However, similar to the USRP, only DUC and DDC are realized on the FPGA, other data processing algorithms are implemented on the ARM using the *liquid-dsp* library. The supported tunable parameters include the sampling rate, center frequency, symbols rate, modulation index, FEC, etc. [9] described an hw/sw co-design approach based on Altera product but without much discussion of the reconfiguration. The SEANet G2, a new IoUT platform developed by [10], is based on the Xilinx Zynq SoC where all the physical layer and time-critical MAC layer functionalities are implemented in the FPGA and the software-defined functionalities and networking protocols are executed in the ARM-based processor. The author also discussed the real-time reconfiguration capability by reconfiguring the guard interval time between two ZP-OFDM symbols in adjacent packets.

Besides, all the aforementioned work adopted the orthogonal frequency division multiplexing (OFDM) technology due to its low computational complexity at the receiver end. However, OFDM has a drawback of the high peak to average power ratio (PAPR) which requires the power amplifier (PA) to work with a large power back-off and in the lowe efficiency operating regime [11]. Moreover, because of the insertion of cyclic prefix, OFDM also has low efficiency in terms of spectral and power efficiency.

Single-carrier modulation (SCM) can avoid these issues but it needs turbo equalization to improve the robustness and reliability, especially in severe Doppler and multipath wireless channels. The real-time hardware implementation of a turbo equalizer requires lots of hardware resources and low processing latency. Since the inherent parallelism as well as the flexible and configurable data movements, FPGA has been used in an implementation of a single-carrier Time-domain

978-0-692-93559-0 ©2021 MTS

LMMSE-based turbo equalizer [12].

Therefore, in this work, we adopt the Xilinx Zynq MPSoC and explore the hardware and software co-design for SCM underwater acoustic communication system. At the transmitter, three modes are implemented based on the partition of the data processing modules on the Processing system (PS) and programmable logic (PL), Full-PL mode, partial-PL mode, and Full-PS mode. At the receiver, a software-defined mode and a high-speed real-time processing mode are discussed. In addition, the real-time reconfiguration capability is described in most of the processing modules. In summary, this modem supports: i) different carrier frequencies up to 200 kHz; ii) different convolutional encoding with 1/2 code rate, constraint length up to 9; iii) different sizes for block interleaving/de-interleaving schemes; iv) different single-carrier M-PSK modulation schemes; v) different symbol rates.

II. System Block

This design leverages the capabilities of the Zynq MPSoC, which integrates ARM-based processing system (PS) and programmable logic (PL) on one chip. The high-performance AXI_HP port provides high-throughput data transfer between the PS and the PL. The PS can run at the bare-metal domain or the Linux domain for general-purpose computing. The PL is used to accelerate compute-intensive tasks. Therefore, the proposed design is able to provide not only high-data-rate characteristics of hardware implementations but also the reconfiguration capabilities of software implementations.

Fig. 1 shows the block-level structure of the SCM underwater acoustic modem employing LMMSE turbo equalization which has the lowest computational complexity and fast convergence. It has a transmit chain and a receive chain, each of which consists of the data processing modules implemented in the Soc either in the PS or in the PL, analog front-end circuits, and an omnidirectional acoustic transducer. Notably, the Digital to Analog Converter (DAC) is not needed in this design.

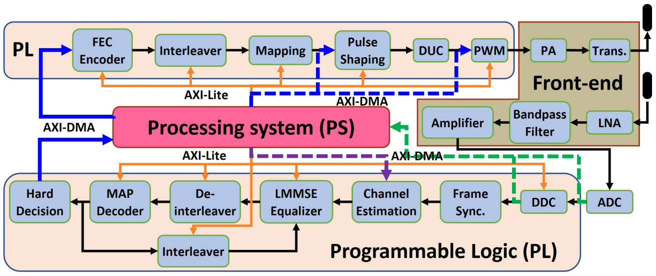


Fig. 1. Block diagram of the SCM underwater acoustic communication system

As shown in Fig. 1, the data exchange between PS and PL is through the AXI Direct Memory Access (DMA) and most of the data processing modules are configured in real-time through the AXI4-Lite interface from the PS.

A. Transmit chain

The transmit chain supports three working modes based on the partition of the data processing modules in the PS and PL.

1) Full-PL Mode: In this mode, PS loads the information bits through AXI DMA to the PL. All the transmitting processing are implemented in the PL, including Forward Error Correction (FEC) encoder, interleaver, mapping, pulse shaping, Digital Upper Converter (DUC), and Pulse Width Modulation (PWM). A detailed description of this mode can refer to [13].

2) Partial-PL Mode: In this mode, the FEC encoder, interleaver, and mapping are implemented in the PS while the time-critical module Pulse-shaping filter is implemented in the PL followed by DUC and PWM.

3) Full-PS mode: This mode is dedicated to the transmit chain that using square-wave as the carrier. Therefore, the pulse shaping module, mapping, and DUC module are not needed. The FEC encoder and interleaver are directly implemented in the PS except for the PWM module. The PWM module has another function in this mode that only outputs half-duty square wave, but the output toggling, i.e. bit one or zero, in every PWM period is controlled by two thresholds which dependents on the symbol phase.

Moreover, the direct access to the PWM module in the PL from PS offers compatibility with other signals using different modulation schemes, for example, the linear frequency modulation (LFM) signal, or the JANUS standard using frequency-shift keying (FSK).

B. Real-time Reconfiguration

1) FEC Encoder: The constraint length of the encoder as well as the generation polynomial can be configured by the PS. In this work, the constrain length is limited to 9 and the supported code rate is 1/2.

2) Interleaver: Interleaver is used to combat the burst error caused by the channel. Block interleaver and random interleaver are two popular methods used in the application. In this work, the addressing of the interleaver is calculated in the PS and then loaded into the block RAM in the PL. In this way, for different block interleaver diagrams, the logic in PL only needs to read out an address from the block RAM and writes the coded bits into the corresponding address of the data block RAM. The interleaved bits are then read out sequentially from the data block RAM.

3) Mapping: This design supports $M-psk$, where $M=2,4,8$. The interleaved bits are grouped based on the selected modulation scheme from PS and then mapped into the I and Q with different amplitude factors.

4) Pulse Shaping: The pulse-shaping filter is used to limit the bandwidth of the transmit signal and maximum the SNR of the received signal together with the match filter at the receiving end. The square-root-raised cosine filter is adopted in this work. The span symbol is limited to $-3T_s$ to $3T_s$. However, the symbol rate R_{sym} can be adjusted by the interpolation ratio r_i during the generation of the filter.

$$R_{sym}(SPS) = \frac{f_c}{r_i} \tag{1}$$

where f_c is the center frequency.

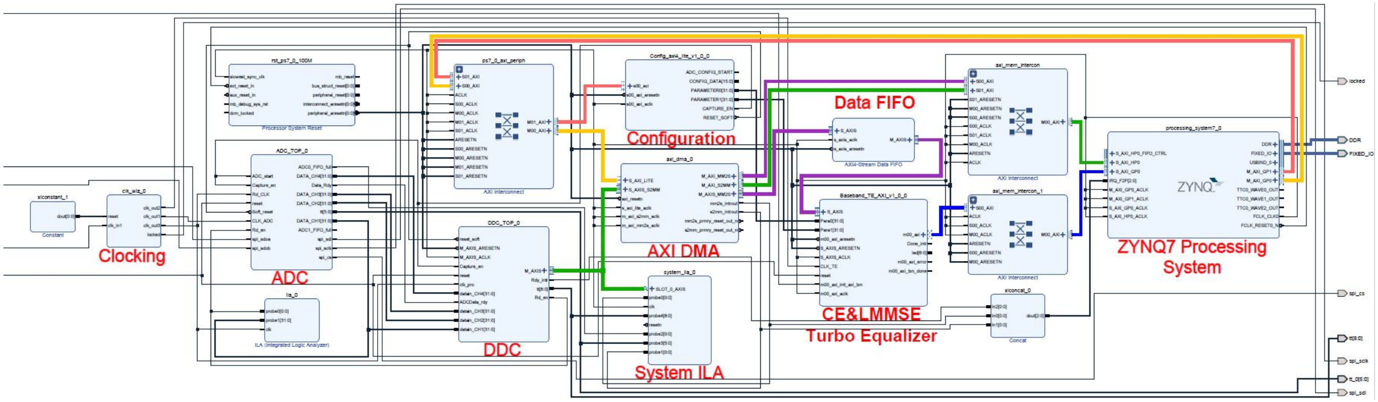


Fig. 2. The block implementation of the receive chain in Vivado.

Besides, the interpolation for the following DUC module is also considered here. The filter taps again are loaded into a block RAM in the PL to complete the pulse shaping.

5) DUC: In the DUC module, the interpolated baseband samples at a ratio of r_s are converted to the passband samples by multiplying the digital local oscillation. However, in this design, the r_s is fixed as 4 so that every carrier is sampled 4 times at frequency f_s. The values are separately $\{0, 1, 0, -1\}$. Therefore, during the multiplying process, only one multiplier in every carrier period is needed to multiply the corresponding interpolated baseband samples with -1.

6) PWM: PWM is a popular method to directly generate the passband waveform in digital circuit [14], [15], which enables a design of an all-digital transmitter.

To represent a passband sample with a binary pulse chain, the PWM module needs to work at a frequency of f_{pwm} which is much higher than the sampling frequency f_s. Denoting the carrier frequency as f_c, the sampling ratio r_s equals to f_s/f_c. Fixing the f_{pwm} in the design, the carrier frequency f_c can be adjusted with the range of the amplitude representation in the PWM module A_r. The equation is given by:

$$\begin{aligned} f_c &= \frac{f_{pwm}}{A_r \times r_s} \\ R &\geq 2^{N_b} \end{aligned} \tag{2}$$

where N_b is the number of bits used in the fixed-point operation before the PWM module.

Moreover, the amplitude of the passband waveform after the DUC module can be also adjusted in the PWM module by multiplying an amplitude factor.

C. Receive chain

As shown in Fig. 1, the receive chain includes the Analog to Digital Converter (ADC), Digital Down Converter (DDC), Frame Synchronization, channel estimation, LMMSE-based turbo equalizer, and Hard decision module. The receiver chain supports both the software-defined mode and high-speed real-time processing mode.

1) software-defined mode: Benefiting from the high-speed data speed offered by the AXI-DMA, the passband samples after ADC or baseband samples DDC are uploaded to the PS and stored in the large volume DDR memory, where the data are processed in the PS within relatively high latency. In this mode, this modem is able to support the JANUS standard. Since the data rate of the JANUS standard is around 80 bits per second, the decoding of the JANUS signal can be done in the PS running on a Linux distribution.

In addition, since the signal saved in the memory can be dumped into files in the Vitis, this mode can be also used as a data acquisition system.

2) High-speed Real-time processing mode: The block implementation in Vivado of the high-speed real-time processing mode is shown in Fig. 1. It mainly consists of Zynq processing system IP, ADC IP, Control/configuration IP, AXI DMA IP, Data FIFO (First-in-First-out) IP, channel estimation (CE), and LMMSE-based Turbo equalizer IP, and a system ILA (Integrated Logic Analyzer) IP. The other IPs are generated automatically to support the embedded system design.

ADC IP is implemented in PL for high-speed communication with external ADC through SPI protocol. The green bold line in Fig. 1 represents the data up-link path, where the baseband samples after the DDC are written into the large volume DDR buffer at PS side by the DMA IP through AXI_HP port. PS takes charge of the signal detection, synchronization, and Doppler frequency shift estimation. Next, PS will transfer the valid signal to the Data FIFO IP at the PL side by the DMA IP, which is named as a downlink path and depicted as the purple bold line. The CE&LMMSE Turbo equalizer IP reads out the signal from the Data FIFO IP and implements channel estimation, LMMSE-based turbo equalization, and MAX-log-MAP decoding. The CE&LMMSE Turbo equalizer IP is able to support two turbo iterations for 1024-symbol block size and achieve 200 ksps transmission rate where the equalizer has a large number of taps on the order of 150. The detailed implementation can refer to [12]. The recovered information bits are then written back to the PS through AXI general-purpose port, which is shown as the blue bold line. The left two bold lines represent the control/configuration paths from

PS to PL through AXI general-purpose ports. The light brown one is corresponding the control/configuration to the DMA IP while the brown one is corresponding to the ADC IP and the CE&LMMSE Turbo equalizer IP. At last, the System ILA IP is used to monitor the important paths for debugging.

D. Real-time Reconfiguration

1) ADC: Since the carrier frequency in the underwater acoustic communication is always tens of kHz or hundreds of kHz, not like in most of the existing modems where the sampling rate is several hundreds of MHz, an external ADC that supports several MSPS sampling rate is the best solution to save power. Normally, the configuration input and sample output of this kind of ADC are through the SPI bus. In this implementation, the configuration data is written from PS through the AXI-Lite interface. Moreover, the sampling rate R_samp of ADC for different center frequencies can be configured by

$$R_{samp}(MSPS) = f_{SCLK}/((N_{ADC} + 4 + wait_{cnt}))/2 \quad (3)$$

where f_{SCLK} is the maximum frequency of SCLK that an ADC can support, N_{ADC} is the number of bits of the ADC, four more clock cycles that are used for the state transition in the RTL code, and *wait_cnt* is a variable in the PL that to control the waiting time between two ADC conversions.

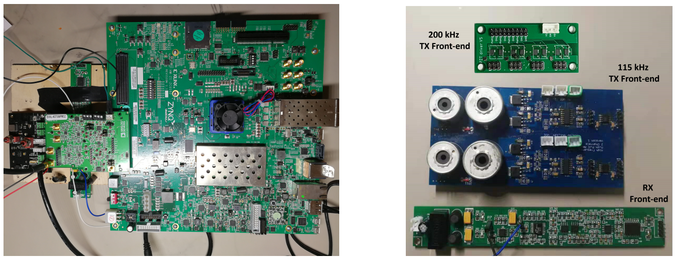
2) DDC: The function of the DDC module includes demodulation that moves the passband signal to the baseband signal, matching filter, and decimation. The samples of the carrier would be loaded into a Block RAM in the FPGA. When the ADC starts working, the signal is demodulated by multiplying the carrier samples that are read out cyclically from the Block RAM.

Since there are a large number of taps in the matching filter, the filtering process would consume lots of resources and cause a large amount of latency. Moreover, most of the filtered samples are discarded in the decimation. Therefore, the poly-phase structure is used in this design as a way of doing sampling-rate conversion that leads to very efficient implementations.

3) Channel Estimation and LMMSE-based Turbo Equalizer: The improved proportionate NLMS (IPNLMS) algorithm is selected in this work. However, due to the high computational complexity in this module, the parameter that can be configured by the PS is only the channel length L which is limited to 100. Therefore, the number of taps N_t of the equalizer is also limited to 151 where $N_t = 3L/2 + 1$. The MAX-log-MAP decoding is fixed to the FEC encoder with the generator polynomial $[G_1, G_2] = [17, 13]_{oct}$ [12].

III. Hardware Platform

The hardware platform for this design consists of the ZCU102 evaluation board with Zynq MPSoc, AD7386FMCZ evaluation board, analog front-end modules, and omnidirectional acoustic transducers, which are shown in Fig. 3.



(a) (b)

Fig. 3. Hardware implementation of the proposed design: (a) ZCU102 with AD7386FMCZ; (b) 200 kHz TX front-end, 115 kHz TX front-end, and RX front-end from top to the bottom.

A. MPSoc FPGA

The ZCU102 evaluation board has an XCZU9EG-2FFVB1156E MPSoC, which features a quad-core ARM Cortes-A53 processor, a dual-core Cortex-R5 real-time processor, and 4 GB DDR4 SODIMM. The high-performance AXI_HP port between PS and PL provides up to 1,200 MB/s read or write bandwidth.

B. External ADC

EVAL-AD7386FMCZ was selected as the external ADC module. It is a full-featured evaluation board designed to evaluate all the features of the AD7386 ADC. The AD7386 is a 16-bit successive approximation register (SAR) ADC that operates from a 3.0 V to 3.6 V power supply. It supports dual simultaneous sampling with throughput rates of up to 4 MSPS. The analog input types are single-ended and are sampled and converted on the falling edge of (CS) signal [16].

In this implementation, the AD7386 was configured as a 4-channel input mode, 2 MSPS per channel. Since the $N_{ADC} = 16$, 16 cycles of *SCLK* are needed to read out a converted sample, from the MSB bit to LSB bit. In this design, we adopted the maximum frequency of SCLK that the ADC7386 could support, 80 MHz. Each clock cycle t_{SCLK} lasts 12.5 ns. For example, to obtain a sampling rate of 1.6 MHz that are used for 400 kHz transducer, the variable *wait_cnt* needs to be set as 5 from the PS; to obtain a sampling rate of 800 kHz that is used for 200 kHz transducer, the variable *wait_cnt* needs to be set as 30. In addition, for other specific sampling rates, the clock frequency of SCLK can be also changed. For example, the clock frequency of SCLK f_{SCLK} and the variable *wait_cnt* can be separately set as 46 MHz and 30 to get 460 kHz sampling rate that can be used for 115 kHz transducer.

C. Front-end circuit

To test the real-time reconfiguration of the center frequency of the carrier, two analog front-end circuits in the transmit chain are used to support 115 kHz and 200 kHz, as shown in Fig. 3.

The analog front-end circuit in the receive chain is also shown in Fig. 3, which is composed of multi-stage amplifiers and band-pass filters. The combination of the resistors and capacitors in the amplifiers and filters circuits are replaced depended on different center frequencies.

IV. Synthesis and Experiment Results

We prototyped this design on Xilinx Zynq UltraScale+ MPSoC ZCU102 Evaluation board which has an Ultrascale+ FPGA (XCZU9EG-2FFVB1156, 274,080 LUTs, 548,160 FFs, 912 36Kb BRAMs, 2,520 DSP48Es). Table I lists a detail of the resource utilization of both of the transmit chain and receive chain.

TABLE I
Resource utilization summary when $L = 100, N_t = 151$

module	BRAM	DSP48E	LUT	FF
TX	66	2	2282	2717
DDC	12	22	2242	4607
CE&Turbo Equalizer	936	1264	162741	237409
Total	56%	51%	61%	45%

A. Experiment Results

We have conducted the lake experiments to test the design at the marina of the Lake Nockamixon in Bucks County, southeastern Pennsylvania. The experiment location is depicted in Fig. 4. Two prototypes are separately used in the experiments as the TX and RX. TX is deployed apart from the RX of 120 m. Both of the transducers are placed at a depth of 1 m.

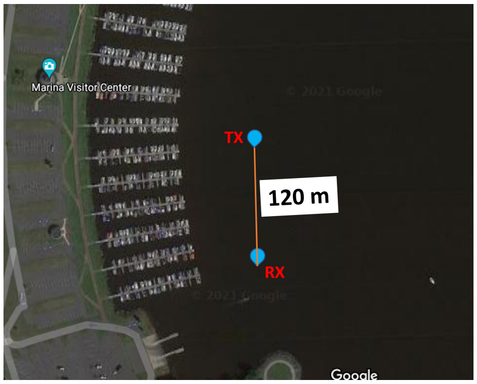
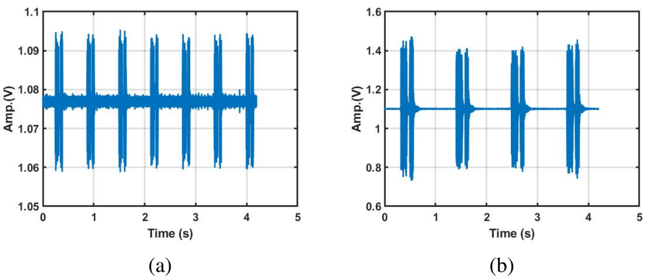


Fig. 4. The experiment location at the marina of Lake Nockamixon

1) Different Center Frequency: We tested the Full-PL mode of the transmit chain separately on the carrier frequency of 115 kHz and 200 kHz. The RX is used in the DAQ mode. The signal is saved into files and processed on the PC.

Fig. 5 shows the received signal using the center frequency of 200 kHz and 115 kHz at 2 MSPS sampling rate, respectively. For 115 kHz, the symbol rate is 23 ksps, while for 200 kHz, the symbol rate is 40 ksps. Since the higher symbol rate, more packets using 200 kHz are received in the same time interval. Please note that the amplitude difference between these two signals is due to the different amplitude gains offered by different analog front-end circuits.

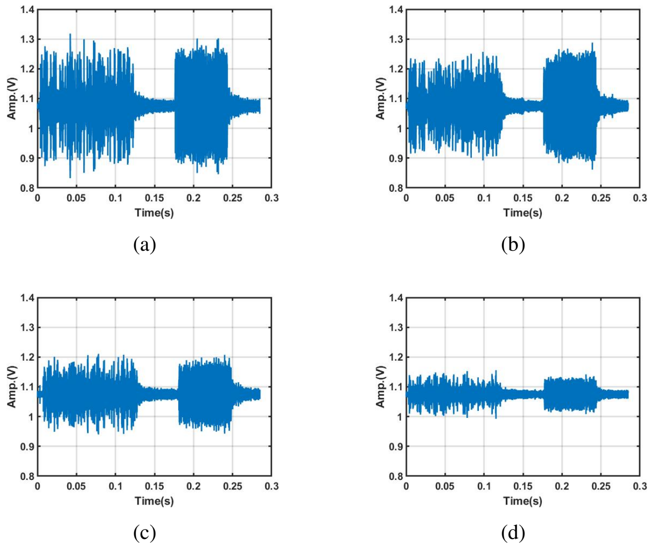
In addition, each packet contains two parts. The first part is the JANUS signal which is directly loaded to the PWM module in the PL from PS, the second part is the BPSK signal which is processed in the Full-PL mode.



(a) (b)

Fig. 5. The received signal with center frequency of (a) 200 kHz (b) 115 kHz at 2 MSPS sampling rate.

2) Amplitude Adjustment: Taking the center frequency of 115 kHz as an example, We adjusted the amplitude factor of the passband signal from 1 to 0.25. Meanwhile, the sampling rate is also reduced to 460 kSPS from 2 MSPS. The attenuation of the received signal is depicted in Fig. 6.



(a) (b) (c) (d)

Fig. 6. Different received signal of 115 kHz with different amplitude factor (a) 1, (b) 0.75, (c) 0.5, (d) 0.25 at 460 kSPS sampling rate.

V. Conclusion

In this work, we proposed a hardware and software codesign for the SCM acoustic communication system based on the Xilinx Zynq MPSoC. Benefiting from the convenient and high-thourghput data transfer between the PS and the PL, we explored different data processing architecture by partitioning the different processing modules into the PS and the PL. Besides, we demonstrated the real-time reconfiguration from the PS to the PL in the lake experiments. In this way, the proposed design is able to provide not only the high-performance real-time hardware implementation but also the reconfiguration capabilities of software implementations.

VI. Acknowledgment

The work is supported in part by NSF projects # CPS-1853257 and # IIP-1853258 and in part by ONR project N00014-19-1-2688. The authors would also like to thank Xiyuan Zhu and Yukang Xue for their help on field experiments.

References

[1] H. S. Dol, P. Casari, T. Van Der Zwan, and R. Otnes, “Software-defined underwater acoustic modems: Historical review and the nilus approach,” *IEEE Journal of Oceanic Engineering*, vol. 42, no. 3, pp. 722–737, 2016.

[2] M. Wang, D. Wang, H. Li, and L. Fu, “A real-time software-defined acoustic modem based on usrp for underwater communications,” in *Global Oceans 2020: Singapore–US Gulf Coast*. IEEE, 2020, pp. 1–6.

[3] E. Demirors, G. Sklivanitis, T. Melodia, S. N. Batalama, and D. A. Pados, “Software-defined underwater acoustic networks: toward a high-rate real-time reconfigurable modem,” *IEEE Communications Magazine*, vol. 53, no. 11, pp. 64–71, 2015.

[4] D. Torres, J. Friedman, T. Schmid, M. B. Srivastava, Y. Noh, and M. Gerla, “Software-defined underwater acoustic networking platform and its applications,” *Ad Hoc Networks*, vol. 34, pp. 252–264, 2015.

[5] M. Rahmati, A. Gurney, and D. Pompili, “Adaptive underwater video transmission via software-defined mimo acoustic modems,” in *OCEANS 2018 MTS/IEEE Charleston*. IEEE, 2018, pp. 1–6.

[6] J. Zhong, L. Liu, W. Yu, and Y. Wang, “Design a reconfigurable modem for underwater acoustic communications,” in *Global Oceans 2020: Singapore–US Gulf Coast*. IEEE, 2020, pp. 1–6.

[7] F. Campagnaro, R. Francescon, O. Kebkal, P. Casari, K. Kebkal, and M. Zorzi, “Full reconfiguration of underwater acoustic networks through low-level physical layer access,” in *Proceedings of the International Conference on Underwater Networks & Systems*, 2017, pp. 1–8.

[8] G. E. Galioto, D. Garlisi, D. Croce, L. Mistretta, R. Badalamenti, I. Tinnirello, C. G. Giaconia, C. Petrioli, and P. Gjanci, “Flumo: Flexible underwater modem,” in *OCEANS 2019-Marseille*. IEEE, 2019, pp. 1–7.

[9] M. Rieß, C. Bock, and F. Slomka, “Generic reusable hardware/software co-design implementation of a complete fh-fsk modem for robust multi-user acoustic underwater communication and system validation on a fpga.” in *MBMV*, 2018.

[10] E. Demirors, J. Shi, R. Guida, and T. Melodia, “Seanet g2: Toward a high-data-rate software-defined underwater acoustic networking platform,” in *Proceedings of the 11th ACM International Conference on Underwater Networks & Systems*, 2016, pp. 1–8.

[11] A. Pitarokoilis, S. K. Mohammed, and E. G. Larsson, “On the optimality of single-carrier transmission in large-scale antenna systems,” *IEEE Wireless Communications Letters*, vol. 1, no. 4, pp. 276–279, 2012.

[12] J. Li and Y. R. Zheng, “Hardware implementation of single-carrier time-domain turbo equalization for severe multipath acoustic communication channels,” in *Global Oceans 2020: Singapore–US Gulf Coast*. IEEE, 2020, pp. 1–7.

[13] J. Li, J. Halt, and Y. R. Zheng, “Utilizing janus for very high frequency underwater acoustic modem,” in *Global Oceans 2020: Singapore–US Gulf Coast*. IEEE, 2020, pp. 1–6.

[14] D. Markert, X. Yu, H. Heimpel, and G. Fischer, “An all-digital, single-bit rf transmitter for massive mimo,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 3, pp. 696–704, 2016.

[15] R. F. Cordeiro, A. S. Oliveira, and J. M. Vieira, “All-digital transmitter with a mixed-domain combination filter,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 63, no. 1, pp. 4–8, 2015.

[16] A. Device, “4-Channel, 4 MSPS, 16-Bit Dual Simultaneous Sampling SAR ADC,” in *Data Sheet*. Analog Device, 2019, pp. 1–31.