

# Design of Reconfigurable System-on-Chip Architecture for Optical Wireless Communication

Syifaul Fuada<sup>1</sup>, Trio Adiono<sup>2</sup>, Angga Pratama Putra<sup>2</sup>, and Erwin Setiawan<sup>2</sup>

<sup>1</sup> Universitas Pendidikan Indonesia, Indonesia

<sup>2</sup> Institut Teknologi Bandung, Bandung, Indonesia

Email: syifaulfuada@upi.edu; tadiono@stei.itb.ac.id; {anggapratap, erwinsetiawan}@pme.itb.ac.id

**Abstract**—To meet the growing demands of the data communication infrastructure in the Internet-of-Things era, alternative methods are needed to complement the current technology, one of which employs optics-based communication. In this paper, we develop optical wireless communication (OWC) infrastructure focuses on digital signal processing (DSP) part. We design System-on-Chip (SoC) architecture based on the Orthogonal Frequency-Division Multiplexing (OFDM) technique with reconfigurable hardware resources. The system developed combines ARM microprocessors with FPGAs. For accelerating the digital processing, several essential parts such as Viterbi decoder, FFT, and time synchronizer are applied to the hardware IP (H/W SoC). While the scheduling is carried out on the software (S/W SoC). With this system, the data communication with other devices can be practiced easily, using various peripherals, i.e., Ethernet, UART, and serial-based connection. Afterward, we exploit the system performance in terms of the hardware resources utilization both for DSP Transmitter and DSP Receiver, also the system latency.

**Index Terms**—Digital Signal Processing (DSP), Optical Wireless Communication (OWC), OFDM, System-on-Chip (SoC)

## I. INTRODUCTION

In recent years, the IoT technology is overgrowing. The growth of this technology demands capable communication infrastructures to provide data connectivity. The Radio Frequency (RF) spectrum which is used frequently today has limited bandwidth. Moreover, it was shared massively in many gadgets and other applications. We need an alternative technology to complement the existing wireless communication technology like RF. One of the solutions is by using another spectrum, e.g., light spectrum; it can be demonstrated on the OWC system.

Research and development in OWC has been done by many works as in [1]-[3]. The OWC technology can be realized by using single-carrier modulation, ex. on-off-keying (OOK) and pulse position modulation (PPM). Besides, the OWC has successfully been carried out using

multi-carrier modulation, ex. OFDM technique [4]-[7]. The OFDM implementation for OWC requires some adjustments by modifying several blocks in OFDM-based RF communication. The OWC can only transmit the real-valued data, different with the RF communication which is able to transmit real as well as imaginary valued data. We can solve it by using the Hermitian symmetry characteristic before the data is modulated in the IFFT processing block, thus the system can produce the real-valued data only [8].

In this paper, we design and develop the SoC architecture for digital processing on the OWC system. We implement the SoC architecture on the Xilinx Zynq-7000 SoC platform. Our contribution is to propose a reconfigurability SoC architecture. Hence it can be used as the base for further development of the OWC technology using OFDM multi-carrier modulation.

Our paper is organized as follows. Section II discusses three topics: overview of the OFDM system, the target platform specifications for the system implementation purpose, and the SoC architecture for the transmitter and receiver sides. Section III presents the results. The conclusion of our work is given in section IV, and the next development plan is discussed in the future works, also at section IV.

## II. METHODS

### A. Overview the OFDM System

The OFDM is used in 802.16 worldwide interoperability for microwave access (WiMAX) standard. The WiMAX standard especially uses OFDM-FDMA [9], also referred to OFDMA. In addition, the OFDM is also employed in digital video broadcasting (DVB) standards. The use of OFDM was first discussed in [10]. In this system, the data stream is split into a set of lower rate sub-stream. The split of serial data offer some advantages compared to the serial system, such as in inter-symbol interference (ISI) and complex equalization. The modulation in OFDM is done by using IFFT operation. The orthogonality is achieved by selecting a unique equidistant set of carrier frequencies or the number of IFFT point. The modulated data is then demodulated by FFT operation on the receiver side.

The OFDM is often combined with constellation mappers such as BPSK, QPSK, 16-QAM, and 64-QAM

---

Manuscript received March 15, 2019; revised September 5, 2019.

This work was supported by the Royal Academy of Engineering (RAENG), United Kingdom, through *Industry-Academia Partnership Programme* under Grant No. IAPP1\100074 collaborated with University of Edinburgh, UK.

Corresponding author email: syifaulfuada@upi.edu  
doi:10.12720/jcm.14.10.965-970

in which all of these modulations are applied as a standard for MAC and PHY layer [11]. To keep the data from the burst error, the interleaving method is usually implemented on the transmitter side. On the open channel of OFDM implementation, the noise distortion can cause ISI. To solve it, we can set the guard interval between symbol by using a cyclic prefix in each subchannel. Therefore, the orthogonality characteristic can be reserved. Besides that, the high peak-to-average power ratio (PAPR) is an exciting topic of the OFDM signal.

In OWC, the high signal variations in the time domain are applied to modulate the LED using intensity modulation. Due to the high-speed of this variation, the human eye cannot notice flashing LED. Therefore, it will not interfere and harmful to the human eye if the LED modulation frequency can be maintained high enough. That is, variations in LED cannot be seen.

**B. Target Hardware and Software**

For the implementation purpose, we employed the Xilinx Zynq-7000 as a target platform. The Zynq-7000 combines ARM processor and FPGA fabric. Both of these resources are connected through Advances eXtensible Interface (AXI) bus. In this paper, the ARM processor will be called as Processing System (PS), and FPGA fabric will be called as Programmable logic (PL). For the target board, we employed the Arty-Z20 which contains the Z-7020 chip of the Zynq-7000 variants. The detail of the hardware resources is shown in Table I [12].

In our architecture, all of the processing blocks for the OFDM are implemented on the PL side. The software is used to initialize the resources that are implemented on the PL. The PS controls them through the AXI bus. The initialization is done for direct memory access (DMA) and the AXI4-Stream Interconnect. In addition, the software side is also used to control the scheduling of each processing block on the PL through the interrupt signal which states “start and end” of the process. In this work, all of the OFDM blocks are implemented using AXI-Stream interface. The interrupt is generated by the DMA block once the data processing is done, and the data is then transferred to the DDR memory.

TABLE I: ARTY Z-20 HARDWARE RESOURCES

Parameters	Descriptions
Target Board	Arty Z-20
Integrated Circuit	XC7Z020-1CLG400C
FPGA Device	XC7Z020
Processor	650 MHz Dual ARM Cortex-A9
Logic Slices	13300
LUTs	53200
Flip-flops	106400
DSP Slices	220
Block RAM	630 KB
DDR Capacity	512 MB
DDR Type	DDR3

We used the Xilinx Vivado software to design the hardware architecture, which includes the PS and PL

blocks, also the interconnection between them. In software development, we used Xilinx SDK to make the application program. It is used to control the designed hardware architecture. Most of the program codes are written using the library from the Xilinx SDK. This program is used to handle three tasks as follow: 1) controlling the data transfer to the on-chip memory; 2) switching each of stream-based IP, and 3) controlling the hardware interrupt signals.

**C. SoC Architecture Design**

The designed SoC architecture is divided into two blocks, i.e., transmitter (Tx), and the receiver (Rx). The SoC transmitter is designed to encode the data. Thus, the data can be corrected if the noise on the channel distorts it, then it is modulated using the IFFT block. While the SoC Rx side is designed to reverse the process on the SoC Tx side, i.e., to demodulate and decode the data.

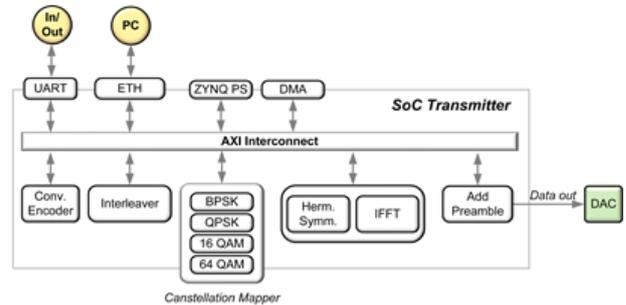


Fig. 1. SoC transmitter architecture

On the SoC Tx side as in Fig. 1, the OFDM block is divided into several blocks, i.e., the convolutional encoder, interleaved, constellation mapper, IFFT, and preamble adder. Each processing block is divided into the independent block so that the configuration for this architecture can be done quickly, in spite of the data processing latency is more significant. It happens because the processed data of each block will be buffered to the memory block before it is processed into the next processing block.

In this design, the raw data is sent from the PC via the Ethernet connection, and then it is stored into the on-chip memory. From the on-chip memory, the information is taken to be processed on the hardware accelerator. We used Xilinx based IP for each block, except for following blocks: the constellation mapper, Hermitian symmetry and preamble adder block, which are the custom based IPs. If each of the accelerator blocks is involved, the OFDM processing flow is as follows. First, the raw data is processed into the convolutional encoder block which is configured to produce the code rate of 1/2, with generator polynomials of  $g_0 = 133$  and  $g_1 = 171$  and constraint length of 7. The convolutional encoder is implemented to give the error correcting characteristic in the data. The encoded data can be decoded using the Viterbi Decoder which resides on the receiver side. The convolutional block takes four clock cycles to process data input.

Then, to keep the data from being vulnerable to burst error, the interleaver hardware accelerator block is involved. In this design, the interleaved block is configured as rectangular interleaver with the column length similar to the number of sub-carrier for data, and the row length corresponding with the amount of the data symbol, in one packet data. The distorted data (due to the burst noise) will be spread among the packet data if block interleaver is used. The interleaver block latency using this configuration is  $10 + (\text{row} * \text{column})$  clock cycles for the first data input to be displayed on the output.

After that, the data is mapped into the constellation in the mapper block. This block consists of BPSK, QPSK, 16-QAM, and 64-QAM modulation which can be chosen using the multiplexer and demultiplexer block. Both the multiplexer and demultiplexer blocks are implemented using the AXI4-Stream Interconnect IP which is controlled by an application program on the PS via the control register routing port. On the application program, the master and slave ports of the AXI4-Stream Interconnect can be enabled and disabled by using the ID number of the master and slave. We can choose the constellation mapper block through these mechanisms. The custom IP latency of the constellation mapper is one clock cycle. The multi-channel modulation process is done through the IFFT block that is configured to Radix-2 Burst I/O with 64-point transform length and with cyclic prefix insertion as a guard band.

For the OWC, the modulated data need to have real value only, so the Hermitian symmetry block is used. By using the Hermitian symmetry block, the data will be processed in the IFFT block with its conjugate. The placement of the data in the IFFT block input after the Hermitian symmetry as shown in Fig. 2.

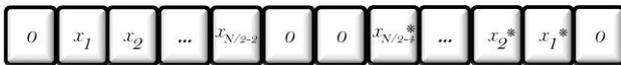


Fig. 2. Symbol sequence for IFFT

Data  $X_1$  to data  $X_{N/2-1}$  is the real data that want to be transmitted. Later,  $N/2$  is the mirror conjugate of the first  $N/2$  data. The first data,  $X_0$ , contains the DC value of the signal, so the value is set as zero. By using this input configuration, the IFFT block output will produce data with real value only. However, on the implementation, the IFFT output still contains the imaginary part but with minimal value, so it is negligible. The IFFT block latency for this configuration is 423 clock cycles.

The modulated data is then transmitted through the channel using an LED driver circuit. To be able to distinguish the start of the data, a preamble is added before each packet data. For the preamble detection, we used the slide correlation method to detect the beginning of the data packet (time synchronization). In this work, we used the synchronization scheme based on [13], which is time synchronization method for OFDM-based visible light communication (VLC) system. We choose this

approach because of the relatively low PAPR of the training sequences. The preamble detection is completed within 68 clock cycles.

The SoC Rx architecture as depicted in Fig. 3 is the reverse process of the SoC Tx. The SoC receiver data flow is as follows: first, the system receives sampled data (Data In) from the analog-to-digital converter (ADC). The received data is then processed in the preamble detector, by cross-correlating the data input with the known symbol to find the the packet data starts. The estimated packet is stored in the on-chip memory to be processed. Later, the stored data is demodulated using FFT block after the cyclic prefix is removed. The demodulated data is then stored on the on-chip memory and the real data is filtered out from the conjugated part. After that, the data is mapped to its value using demapper block.

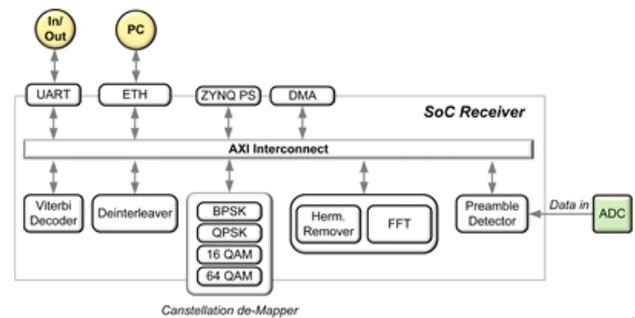


Fig. 3. SoC receiver architecture

The constellation de-mapper block uses the same mechanism with the constellation mapper block on the transmitter. The data is de-interleaved using the de-interleaver block that has the same row and column size with the interleaver block on the SoC Tx. The de-interleaved data is processed by the error correcting module, i.e., Viterbi decoder which has been configured to have the same generator polynomial value and code-rate with the convolutional encoder. The decoding process is configured to use hard coding. In this work, the Viterbi block has the most substantial latency of all the blocks used, which is 571 clock cycles.

Next, the demodulated and decoded data is sent to the PC via an Ethernet cable, to be analyzed further. The Ethernet communication process on the transmitter and receiver are done on the software layer. We employed the *lwIP* library which is provided by Xilinx SDK to configure the data transmission protocol. In this work, we employed the user datagram protocol (UDP) to transfer the data from the PC to Arty-Z20 board, and to receive the data from Arty-Z20 board to the PC. The socket program on the PC itself is done using MATLAB.

By using MATLAB socket programming, the received data can be analyzed and processed directly using available features such as bit-error-rate (BER) vs. signal-to-noise ratio (SNR) curve, constellation plot, FFT spectrum analysis, etc.

### III. RESULTS AND ANALYSIS

#### A. Implementation

TABLE II: UTILIZATION OF TRANSMITTER HARDWARE RESOURCES

Instances	LUT	SRL	FF	BRAM	DSP48
AXI DMA	1318	94	1818	2	-
Conv.	21	2	26	-	-
Coding					
Interleaver	67	1	69	1	-
BPSK Map	2	-	3	-	-
QPSK Map	3	-	5	-	-
16QAM Map	3	-	7	-	-
64QAM Map	11	-	21	-	-
Herm. Sym.	142	-	88	-	1
IFFT	564	100	1118	3	3

TABLE III: UTILIZATION OF RECEIVER HARDWARE RESOURCES

Instances	LUT	SRL	FF	BRAM	DSP48
AXI DMA	1316	94	1820	2	-
Preamble Detect	7235	86	7430	-	4
FFT	542	96	1086	3	3
Herm. Remover	7	-	39	-	-
BPSK De-Map	1	-	2	-	-
QPSK De-Map	1	-	3	-	-
16QAM De-Map	11	-	5	-	-
64QAM De-Map	26	-	7	-	-
De-Interleaver	68	1	74	1	-

Table II shows the resource utilization of the transmitter while Table III shows the resource utilization of the receiver architecture. The data for hardware resource usage is taken from the implementation report of the Xilinx Vivado in which it is generated after the synthesis and implementation process.

For the SoC Tx, the total on-chip power is 1.462 Watt, whereas for the receiver the total on-chip power is 1,526 Watt. The SoC Rx uses more hardware resources compared to SoC Tx because it has a more complex operation to do, such as synchronizing and decoding the data. In particular, it can be seen from the difference of the amounts of DSP48 and logic slices used on the receiver. In summary, the number of used resources in the SoC Tx is less than SoC Rx.

#### B. Performance

Table IV lists the system performances when processing one symbol of data. The system is run sequentially for each processing block, as shown in Fig. 4, so the system accomplishments have a lot of room for optimization. Several IPs can be combined to form a single hardware accelerator, so the process can become faster by eliminating the buffering data process in on-chip memory, despite it comes with the disadvantages of reduced the system flexibility.

For the FFT block, the latency is measured by the time it takes to process one symbol of data using 64-point butterfly operation. Hence, the high latency of the FFT block comes from the operation complexity, and so does for the Viterbi decoder, which takes the most extended clock cycles to process the data. The Viterbi decoder is very crucial to determines the BER value between the transmitted and received data, besides the preamble detector block.

TABLE IV: PERFORMANCE RESULTS WITH 100 MHZ OF FPGA CLOCK

PL Blocks	Time
(i)FFT	4 $\mu$ s
Const. (De)Mapper	$\approx$ 0 $\mu$ s
Convolutional Encoder	$\approx$ 0 $\mu$ s
Interleaver	$\approx$ 0 $\mu$ s
Viterbi Decoder	6 $\mu$ s
Preamble Detector	$\approx$ 1 $\mu$ s

The preamble detector uses a CORDIC algorithm to find the peak of the magnitude and phase in slide correlation of the training symbol. The CORDIC block is implemented by using Xilinx IP as sub-process of the preamble detection. The CORDIC algorithm is done for ten iterations to execute the *arctan* operation. It was used to detect the phase and rectangular. So that the magnitude can be found by the polar conversion.

In this paper, the synchronizer block which is used for preamble identifying is not elaborated, because this paper is more focused on reporting the SoC Tx-Rx architecture, not detailing the discussion for each processing block.

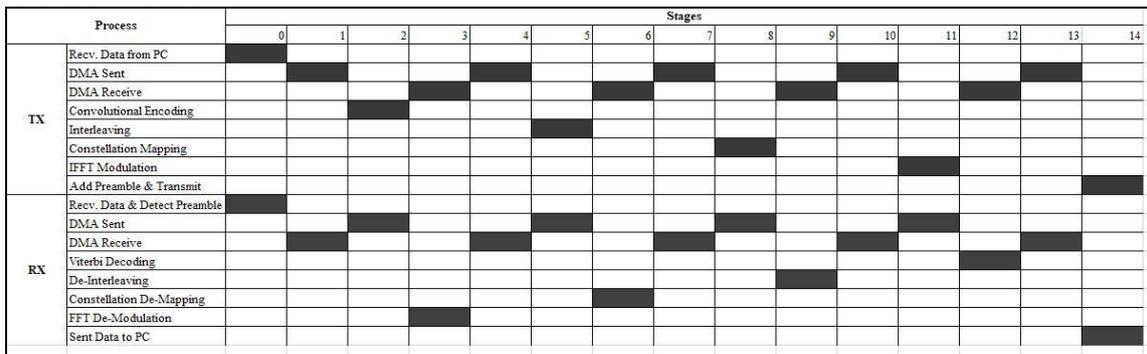


Fig. 4. Timing diagram

### IV. CONCLUSION AND FUTURE WORK

The reconfigurable SoC architecture for OWC has been designed and implemented carefully on the Xilinx

Zynq-7000 board. The results show that the developed SoC could be used as a base of the DSP for OWC application. The architecture is designed for system reconfigurability, by separating each processing block as

an independent hardware accelerator. On SoC Tx and SoC Rx side, several block IPs can be combined, thus it will reduce the on-chip memory buffering before the finished data process of that block, processed into another block. Hence, the disadvantages of this architecture are on the buffering latency, which is controlled by the software layer when managing the data transfer “to” and “from” on-chip memory with the help of DMA. In this work, all of the data processing is done on the PL block thus causing a large number of resource usages which in turn increases the total power of the SoC.

In the future works, hardware/software co-design of this system will be explored, which cover the topic of task partitioning between the hardware layer and the software layer. Also, the whole system, integrated with the analog front end (AFE) circuit as in [14]-[17] and LED driver as in [18]-[20] will be discussed, in which the BER against SNR will indicate the complete system performances.

## REFERENCES

- [1] E. Sağ and A. Kavas, “Modelling and performance analysis of 2.5 Gbps Inter-satellite Optical Wireless Communication (IsOWC) system in LEO Constellation,” *J. of Communications*, vol. 13, no. 10, pp. 553-558, 2018.
- [2] J. H. Liu, *et al.*, “Cellular coverage optimization for indoor visible light communication and illumination networks,” *J. of Communications*, vol. 9, no. 11, pp. 891-898, 2014.
- [3] M. S. M. Gismalla and M. F. L. Abdullah, “Optimization of received power and SNR for an indoor attocells network in visible light communication,” *J. of Communications*, vol. 14, no. 1, pp. 64-69, 2018.
- [4] T. Adiono, A. Pradana, S. Fuada, and Y. Aska, “Desain dan implementasi Real-time visible light communication systems berbasis BPSK,” *Jurnal Elinvo*, vol. 3, no. 2, 2018.
- [5] T. Adiono, A. Pradana, and S. Fuada, “A Low-complexity of VLC System using BPSK,” *Int. J. of Recent Contributions from Engineering, Science & IT*, vol. 6, no. 1, pp. 99-106, 2018.
- [6] T. Adiono, A. Pradana, and S. Fuada, “Visible light communications system using quadrature phase-shift keying,” *Unpublished*.
- [7] T. Adiono, A. Pradana, and S. Fuada, “A low-cost hardware implementation for QAM-16 visible light communications,” *Unpublished*.
- [8] T. Adiono and A. P. Putra, “Hardware/Software model of DCO-OFDM based visible light communication SoC using DMA,” *Proc. of ISOCC*, pp. 92-93, 2017.
- [9] H. Yaghoobi, “Scalable OFDMA physical layer in IEEE 802.16 WirelessMAN,” *Intel Technology Journal*, vol. 8, no. 3, pp. 201-212, 2004.
- [10] Y. Tanaka, *et al.*, “Indoor visible communication utilizing plural white LEDs as lighting,” in *Proc. 12<sup>th</sup> IEEE Int. Symposium on Personal, Indoor and Mobile Radio Communications*, 2001, pp. F81-F85.
- [11] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer in the 5 GHz Band*, IEEE Std. 802.11a-1999.
- [12] Digilent, Inc. (2016) Arty Z7 Reference Manual. [Online]. Available: <https://reference.digilentinc.com/reference/programmable-logic/arty-z7/reference-manual>
- [13] M. F. G. Medina, *et al.*, “Timing synchronization for OFDM-based visible light communication system,” in *Proc. Wireless Telecommunications Symposium (WTS)*, 2016, pp. 1-4.
- [14] S. Fuada and T. Adiono, “Visible light communication kits for educations,” *J. of Education and Training*, vol. 5, no. 2, pp. 39-49, May 2018.
- [15] T. Adiono, A. Pradana, R. V. W. Putra, and S. Fuada, “Analog filters design in VLC analog front-end receiver for reducing indoor ambient light noise,” in *Proc. IEEE Asia Pacific Conf. on Circuit and Systems (APCCAS)*, October 2016, pp. 581-584.
- [16] S. Fuada, A. P. Putra, Y. Aska, and T. Adiono, “A First approach to design mobility function and noise filter in VLC system utilizing low-cost analog circuits,” *Int. J. of Recent Contributions from Engineering, Science, & IT (iJES)*, vol. 5, no. 2, pp. 14-30, 2017.
- [17] T. Adiono, S. Fuada, and A. Pradana, “A circuit for robust visible light communication systems in indoor environment,” in *Proc. 10<sup>th</sup> Int. Conf. on Information Technology and Electrical Engineering (ICITEE)*, July 2018, pp. 68-72.
- [18] S. Fuada, T. Adiono, A. P. Putra, and Y. Aska, “A low-cost Analog Front-End (AFE) transmitter designs for OFDM visible light communications,” in *Proc. IEEE Int. Symposium on Electronics and Smart Devices (ISESD)*, October 2016, pp. 371-375.
- [19] T. Adiono and S. Fuada, “Desain dan Implementasi LED driver linier untuk aplikasi visible light communication,” *Unpublished*.
- [20] S. Fuada, T. Adiono, A. P. Putra, and Y. Aska, “LED driver design for indoor lighting and low-rate data transmission purpose,” *Optik-Int. J. for Light and Electron Optics*, vol. 156, pp. 847-856, 2018.



**Syifaul Fuada** received a B.A. in Electrical Engineering Education from Universitas Negeri Malang (UM), Indonesia, and an M.Sc. in Electrical Engineering option Microelectronics from the School of Electrical Engineering and Informatics, Institut Teknologi Bandung (ITB), Indonesia. He was with the University Center of Excellence at Microelectronics ITB from 2016-2018 as a main researcher. Now, he is with the Program Studi Sistem Telekomunikasi Universitas Pendidikan Indonesia (UPI) as a Lecturer. His research interests include analog circuit design and instrumentation, circuit simulation, engineering education, IoT, multimedia learning development and Visible Light Communication.



**Trio Adiono** received a B.Eng. in electrical engineering and an M.Eng. in microelectronics from Institut Teknologi Bandung (ITB), Indonesia, in 1994 and 1996, respectively. He obtained his Ph.D. in VLSI Design from the Tokyo Institute of Technology, Japan, in 2002. He holds a Japanese Patent on a High-Quality

Video Compression System. He is now a Full professor and a senior lecturer at the School of Electrical Engineering and Informatics, and formerly serves as the Head of the Microelectronics Center, Institut Teknologi Bandung. His research interests include VLSI design, signal and image processing, VLC, smart cards, and electronics solution design and integration.



**Angga Pratama Putra** received his B.Sc. degree on Electrical Engineering from School of Electrical Engineering and Informatics, Institut Teknologi Bandung (ITB), Indonesia (2015) and M.Sc in Electrical Engineering with specialization in Computer Engineering, with same campus (2017). His research

interests include: Embedded System, Software Engineering, VLSI, System-on-Chip, Internet of Things, Digital Signal

Processing (DSP) and Visible Light Communication (VLC). His research topic about reconfigurable SoC design for OFDM-based VLC.



**Erwin Setiawan** received the B.Eng. degree in Computer Engineering from Maranatha Christian University, Bandung, Indonesia, in 2014, and the M.Sc. degree in Microelectronics Engineering from Institut Teknologi Bandung (ITB), Indonesia, in 2018 with Cumlaude. He is currently a researcher at

Microelectronics Center of ITB. His main research interests are in VLSI and SoC design, embedded system, and visible light communications.