

Comparison study of 8-PPM, 8-DPIM, and 8-RDH-PIM modulator FPGA hardware design in term of bandwidth efficiency and transmission rate

M. A. Ilyas¹, Maisara Othman², Rahmat Talib³, R. Yahya⁴, M. Yaacob⁵,
S. M. Mustam⁶, M. B. Jaafar⁷, C. B. M. Rashidi⁸

^{1,2,3,4,5,6,7}Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, Malaysia

⁸Advanced Communication Engineering, Centre of Excellence-School of Computer and Communication Engineering, Universiti Malaysia Perlis, Malaysia

Article Info

Article history:

Received Oct 23, 2019

Revised Dec 30, 2019

Accepted Feb 1, 2020

Keywords:

Digital pulse interval modulation (DPIM)
FPGA modulation design
pulse-position modulation (PPM)
Reverse dual header-pulse interval modulation (RDH-PIM)

ABSTRACT

In this paper, a performance study of 8-Pulse-Position Modulation (PPM), 8-Digital Pulse Interval Modulation (DPIM), and 8-Reverse Dual Header-Pulse Interval Modulation (RDH-PIM) implementation in Verilog hardware design language is presented. The hardware design is chosen over software design since it could provide much more flexibility in term of transmission rate and reduce the workload of the processor in the complete system. Using 50 MHz clock as the reference data clock speeds, the transmission rate recorded are 11.11 Msymbol/second or 33.33 Mbps, 9.09 Msymbol/s or 27.27 Mbps, and 6.25 Msymbol/s or 18.75 Mbps for 8-RDH-PIM, 8-DPIM, and 8-PPM respectively. We conclude that 8-RDH-PIM modulator design provides better performance in term of bandwidth utilization and transmission rate as compared to 8-PPM and 8-DPIM.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Maisara Othman,
Faculty of Electrical and Electronic Engineering,
Universiti Tun Hussein Onn Malaysia,
Parit Raja, Batu Pahat, Malaysia.
Email: maisara@uthm.edu.my

1. INTRODUCTION

In recent years, many researchers who studied the modulation hardware experimental works such as free space optical communication have shifted from experimental setup to rapid hardware prototyping especially on FPGA platform [1, 2]. In the experimental setup, each component has its own particular hardware. PRBS for example, can be created using PRBS generator machine. The same situation is also true for modulator, sampling, demodulator, laser, and photo detector. This set up requires a lengthy step and some spaces to keep the hardware. This method is also inflexible especially in electrical components that use pre made circuit. With developments in semiconductor and computing field, it is possible to simplify the optical communication system by transforming all the electrical components in a single FPGA chip implementation which ultimately saves the set up time and cost [3].

Many applications for example digital modulation can be implemented in both software and hardware design. However, the performance of both designs yielding a different result. In [4] and [5], on-off keying (OOK) modulation is implemented in Arduino using software design. The highest data rate recorded for the design is around 100 Kbps using 16 MHz system clock speed. While [6] shows the hardware

design of PPM using FPGA that produce data rate of 5 Mbps using 160 MHz system clock speed. Although software could provide a flexible design, the transmission rate is significantly lower than hardware design as some of the clock speed is used for processing purposes. Hence, it can be inferred that in term of implementing a time sensitive application, hardware design much more preferable as it could provide a better performance.

In this paper, the performance of bandwidth efficiency and transmission rate comparison of several digital modulations for instance Pulse-Position Modulation (PPM), Digital Pulse Interval Modulation (DPIM), and Reverse Dual Header-Pulse Interval Modulation (RDH-PIM) has been investigated. This modulation is chosen since it could provide a better performance in term of bit error rate as mentioned in [7, 8] as compared to other modulation.

2. BACKGROUND STUDY

In this section, the background study is presented. The section starts with explanation of digital modulation followed by FPGA structure, previous implementation, and the key challenges.

2.1. Digital modulation

Digital modulation is somewhat similar to the analog modulation except base band signal is of discrete amplitude level. For binary signal it has only two level, either high or logic 1 or low or logic 0. For this paper, digital modulation for 8-level such as Reverse Dual Header-Pulse Interval Modulation (RDH-PIM), Pulse-Position Modulation (PPM), Digital Pulse Interval Modulation (PIM) performance is investigated. Table 1 shows the actual data frame conversion for 8-PPM, 8-PIM, 8-DH-PIM and 8-RDH-PIM. PPM operates by varying the position of logic 1 within a certain transmission period. For 3 bits input, the transmission period is divided by 8 partitions and each input have different logic 1 position. For this application, a synchronous timing between transmitter and receiver is crucial in order the modulation to work perfectly [8-10]. Next, DPIM is modulation technique that sent data using a variable size of time interval. Every level has different interval which end up with variable transmission data rate. This technique significantly improves the speed of data transmission as compared to PPM for the same M-level [11-12].

Lastly, RDH-PIM is basically an improvise version from the Digital Pulse Interval Modulation (DPIM) and Dual Header-Pulse Interval Modulation (DH-PIM). Reverse indicates the inversion of data bit of DH-PIM which is 0 to 1 and vice versa. Basically, it starts with a dual header, which represents the weight of the decimal value of the input binary word followed by the number of time slots representing the information carried by the input frame. It can be seen that DH-PIM not only removes the redundant time slots following the pulse as in PPM frame, but it also reduces the average frame duration to around half that of DPIM and a quarter that of PPM, thus resulting in a much higher bit rate [7].

Table 1. Modulation data frame

OOK	8-PPM	8-DPIM	8-DHPIM	8-RDHPIM
000	10000000	10	100	011
001	01000000	100	1000	0111
010	00100000	1000	10000	01111
011	00010000	10000	100000	011111
100	00001000	100000	110000	001111
101	00000100	1000000	11000	00111
110	00000010	10000000	1100	0011
111	00000001	100000000	110	001

2.2. FPGA

Field Programmable Gate Arrays (FPGAs) are semiconductor devices that are based around a matrix of configurable logic blocks (CLBs) connected via programmable interconnects. FPGAs can be reprogrammed to desired application or functionality requirements after manufacturing [13, 14]. The FPGA architecture consists of three major components which are I/O blocks, programmable routing, and programmable logic blocks [14]. Programmable logic blocks, which implement logic functions, provide basic computation and storage elements used in digital systems. A basic logic element consists of programmable combinational logic, a flip-flop, and some fast carry logic to reduce area and delay cost [15-17]. Modern FPGAs contain a heterogeneous mixture of different blocks such as dedicated memory

blocks and multiplexers. Configuration of memory is used throughout the logic blocks in order to control the specific function of each element [18].

Next, the programmable routing (interconnects) which implements functions establishes a connection between logic blocks and Input/output blocks to complete a user-defined design unit. It consists of multiplexers pass transistors and tri-state buffers. Pass transistors and multiplexers are used in a logic cluster to connect the logic elements [19].

FPGA also has I/O blocks which are used to make off-chip connections. The programmable I/O pads are used to interface the logic blocks and routing architecture to the external components. The I/O pads and the surrounding logic circuit form I/O cells. These cells consume a large portion of the FPGA's area. The design of I/O programmable blocks is complex, as there are great differences in the supplied voltage and referenced voltage. The selection of standards is important in an I/O architecture design. Supporting a large number of standards can increase the silicon chip area required for I/O cells [20].

With advancements in technology, the basic FPGA architecture has developed through the additions of more specialized programmable function blocks. The special functional blocks like ALUs, block RAM, multiplexers, DSP-48, and microprocessors have been added to the FPGA, due to the frequency of the need for such resources for applications [21]. FPGA have widely used in many fields, for examples wired communications. Its provide end-to-end solutions for the Reprogrammable Networking Linecard Symbol Processing, Framer/MAC, and serial backplanes, On top of that, FPGA has been used in wireless communications such as RF, base band, connectivity, transport and networking solutions for wireless equipment. Due to its performance and features, FPGA is selected as the platform to implement the digital modulation technique [22, 23].

2.3. Digital modulation implementation in FPGA

In PPM modulation, there are several research that have been implemented in FPGA. For example, In [1], the paper presents the implementation of the 16-PPM modulation on FPGA. The modulation technique is created using the Altium Nano Board 3000 (provided with Spartan 3AN FPGA) at the transmitter and receiver. A mixed design of schematic capture and a VHDL code are adapted to generate the required modulation schemes. It requires 2 clocks which are data clock and PPM clock. The simulation results show that the design perform the implemented algorithm and recorded performance a speed of 160 MHz.

Next, for optical wireless communication systems, modulation schemes such as PPM and DPPM are often used because of their power efficiency. One of the key difficulties in implementing PPM is that the receiver must be properly synchronized to align the local clock at the beginning of each symbol. Therefore, it is often implemented as Differential pulse-position modulation. As the interest in the deployment of free space optical (FSO) links gain momentum, it becomes important to investigate the implementation of the proposed higher state modulation schemes. A higher order modulation schemes seeks better results and M-ary DPPM promises significant performance enhancement. This paper gives results about an experimental VHDL based FPGA implementation of a 256-ary DPPM modem presented in [2]. Similar to a previous design, it also requires 2 clocks which are data clock and DPPM. These clocks are responsible to synchronize the data transmission. In this case, one period of data clock must contain 256 DPPM clock cycle.

A similar modulator architecture that uses dual clock system is also presented in [3] as shown in the PPM implementation in FPGA. This paper has developed an underwater laser communication in the field of deep-sea exploration which has the edges of hidden, safe, non-contact and rapid mobility. Current data exchange and transmission between underwater sensors and devices are achieved by RS-232 serial ports communication. Nevertheless, it is not able to meet the demands of the present underwater communication due to the limitations in transmitting rate and distance. Considering the effect of absorption and scattering, a series of experiments are performed for unmodulated and modulated underwater laser communication experiments under different visibility water. By comparing the bit error rate results, a whole underwater laser communication system that is compact, efficient and reliable has been created.

For D-PIM, only one resource has been found for FPGA implementation which can be found in [6]. In this paper, DPIM which has a dynamic symbol size and PPM algorithms has been created and performances of the symbol and algorithm are measured. Both modulations are implemented using XC3S400-4C PQG208 SPARTAN 3 Xilinx board. This DPIM modulator algorithm has been successfully implemented using a counter module-based design. The counter is responsible for synchronizing the data transmission according to the symbol size. Single clock architecture is used within the system. A slot clock is created to provide the respective clock interval for each symbol. The counter module synchronizes the data transmission by using a reset pin which is directly connected to a random access memory (RAM) module that only allows only a single data transmission to be done at any specific time. On top of that, it also has a latch

module within the system which handles the input bit stream. Within an FPGA system, latch is the component that is avoided during implementation. This is because latch can be very difficult for the FPGA tools to create properly. Often it adds significant routing delays and can cause a design to fail to meet the stipulated timing.

2.4. Challenges to implement dynamic packet size in FPGA

Verilog Hardware Design Language (HDL) allows one-dimensional arrays of variables all along and Verilog-2001 allows multi-dimensional ones too. SystemVerilog, for an example, classifies an array as 'packed' or 'un-packed' depending on how it is declared [21, 24]. The array upper and lower bounds are declared between the variable types and the variable names, such as;

```
reg [7:0] array; (packed)
int int_array [7:0]; (unpacked)
reg [7:0] reg_array [3:0][7:0]; (both)
```

One thing that remains common is they are all static declarations. Once an array is declared, the implementation and synthesis tools statically allocate memories for the array and there is no other way that it can change afterwards. As a result, the size of an array, for an example, cannot be changed once it has been declared.

There are occasions when it is required to declare an array size that cannot be pre-determined. For example, a temporary buffer for variable rate incoming data stream and a list that has a variable number of elements are few examples of problems that require an array size needs to be changed dynamically. Other than that, using a very large array with the assumption that it can hold the largest data size is unknown which makes the system neither safe nor efficient. The, an issue arises on how to safely implement an array size that is dynamically alterable in the FPGA [16, 25].

During the implementation of RDHPIM for an instance, a dynamic array of integer size that can be set or changed at runtime is needed. However, as mentioned earlier, in Verilog, the dimension of the array can only be set during declaration and it cannot be changed during runtime. Although this limitation has been overcome in SystemVerilog, the features have not been supported by Vivado and Quartus software [14, 20, 26, 27]. Therefore, in the modulator design, a different approach is proposed to be implemented in the design in order to achieve the desired outcome.

3. RESEARCH METHOD

In this section, the design process of proposed design is explained. The section starts with explanation of overall design and parameters that being investigated.

3.1. Overall setup

Figure 1 shows the overall experimental setup for this project which consist of a computer, Vivado HLx software, and Xilinx KCU105 Board. The computer act as a bridge to connect Vivado software and the board. While, the Vivado software is used as a platform to design the hardware in KCU105 board. Register Transfer Level (RTL) design with Verilog as the Hardware Design Language (HDL) has been chosen for this implementation. RTL implies that the Verilog code describes how data is transformed as it is passed from register to register. The transforming of the data is performed by the combinational logic that exists between the registers. This design method gives researcher a full access to control, optimize, and troubleshoot the design accurately.



Figure 1. Overall experimental setup

3.2. Proposed design

Figure 2 shows the proposed design that has been implemented in the overall architecture. The data start by feeding PRBS data into the system. The modulator encoded the data to be transmitted according

to its' symbol size. The demodulator with twice the speed of transmitter is sampling the signal through the optical channel and decoding back to its original form. It should be noted that, the parallel indicates that the operation is done simultaneously while the serial represents that the operation is repeated in a series of clock cycle. For 8-RDHPIM, in the PRBS sequence of '010', the modulator produces a modulated signal of '01111' which is transmitted in the channel in 5 clock cycle. The demodulators sample the signal '0011111111' in 10 cycles and decode it to original data of '010'.

In the 8-RDHPIM the separation of each symbol is given by the first 0 bit that indicates the stop signal for the previous transmission as well as a start signal for the current transmission. Asynchronous data transmission that can be achieved by this method reduces the needs for the synchronous clock between two systems. All the synchronization processes are done by the symbol sequences.

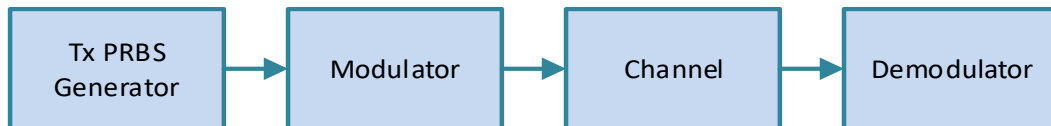


Figure 2. Proposed design

For this project, a slightly different approach is considered for the modulation design in hardware implementation. In previous study [4], the 16-PPM modulation with additional clocks is used. A different clock speed of 5 MHz and 80 MHz is used for data and modulation respectively. For 16-PPM, the modulation clock needs to be 16 times faster than the data clock speed. With these configuration, the data rate is always equal to the data clock.

Although the approach is suitable for PPM, the same design cannot be considered for 8-DPIM and 8-RDH-PIM. This is because, both modulation have a dynamic output bit size that correspond to the input bit as shown in Table 1. To achieved the same result as [4], many additional clocks speed needs to be added in the system which is quite an unproductive solution since single clocks is used for only one or two input data. Therefore, for this project, only one clock (50 MHz) is proposed to be used for both data and modulation.

The proposed modulator design starts the cycle by providing an input data to the design. The data are then transformed into their corresponding symbol size. The symbol is then transmitted until the last bits before it cycles to the next input data. The key challenges for both designs are to implement the dynamic array size which cannot be applied using conventional methods. Meanwhile, demodulator's proposed design starts by feeding a demodulated signal to the sampling module. The sampling process starts upon receiving the start bit (logic 0) until the next start bit (logic 0). After detecting the next start bit, the sampled data is analysed and transformed into the original data. At the same time, the sampling process is continued by sampling module. For this project, the parameters that being analyze are timing diagram, timing summary, hardware utilization, power summary, average transmission rate, and bandwidth efficiency.

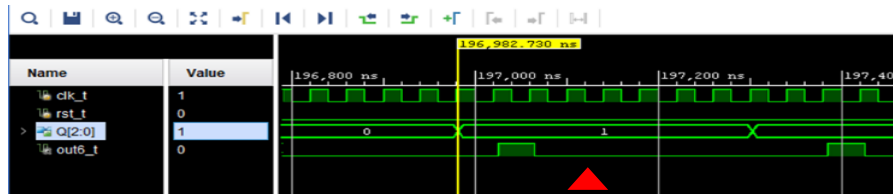
4. RESULTS AND ANALYSIS

In this section, the overall result and analysis is presented.

4.1. Modulation timing diagram

Figure 3(a), (b), and (c) shows the timing diagram for 8-PPM, 8-DPIM, and 8-RDH-PIM respectively. The module consists of clock (clk_t) signal, reset signal (rst_t), 3 bit input data (bin_in_t), and 3-9 bit output data (out6_t) that has different array size depending on the modulation scheme. The yellow line and triangle shape shown indicates the data transmission starting period. The output pin transmit data based on the data that given previously. There are two important parameters that can be seen by the waveform. Firstly, the operation is executed during positive edge the clock. Secondly, the output data holds the operation until the last bit transmitted before proceed to the next data transmission.

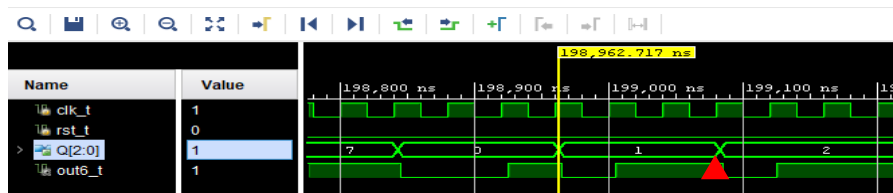
From the timing diagram, the transmission rate can also be determined. The highest transmission rate is recorded by 8-RDH-PIM with average speed of 11.11 Msymbol/s or 33.33 Mbps. Which then followed by 8-DPIM with 9.09 Msymbol/s or 27.27 Mbps and lastly by 8-PPM with 6.25 Msymbol/s or 18.75 Mbps. On top of that, bandwidth efficiency of 66.66%, 54.54% and 37.5% was calculated with respect to 50 MHz original data clock speed that leads by 8-RDH-PIM, 8-DPIM, and 8-PPM respectively.



(a)



(b)



(c)

Figure 3. Timing diagram, (a) 8-PPM, (b) 8-DPIM, (c) 8-RDH-PIM

4.2. Power and hardware utilization

Table 2 shows the hardware utilization in actual FPGA's. Ideally, lower hardware utilization is desired as to lower the power consumption to drive the component and saving the hardware components for other applications. In this case, 8-DPIM has the lowest hardware utilization with 66 components. Followed by 8-RDH-PIM with 77 components and 83 components by 8-PPM. The explanation about each component and how its operates can be found in [22]. For power consumption, basically all modulations have a total of 0.479 W On-chip powers which 0.001 W is allocated for dynamic design. There are no significant different in the power allocation since the design have relatively small number of hardware utilization.

Table 2. Hardware utilization

Parameter	8-PPM	8-DPIM	8-RDH-PIM
CLB LUTs	16	11	14
CLB Registers	27	24	26
CLB	4	4	4
LUT as Logic	16	11	14
LUT Flip Flop Pairs	14	10	13
Bonded IOB	3	3	3
HPIOB	3	3	3
Total	83	66	77

4.3. Timing summary

Timing summary is shown in Table 3. As we can see all the timing requirement has been meet. For 8-RDH-PIM, setup, hold and pulse width slack time record a positive value of 19.215 ns, 0.048 ns, and 9.725 ns respectively which indicates that the system work perfectly. Setup slack time is the marginal time recorded of signal from one module to the other modules. Higher slack time gives developer an option to use higher clock speed. As mention earlier, 50 MHz or 20 ns clock speed is used in this design. Since we have 19ns setup time margin, a maximum of 2 ns or 500 MHz clock speed can be implemented. However, for this design, 50 MHz clock is enough to show the operation of this systems.

Table 3. Timing summary

Parameter	8-PPM	8-DPIM	8-RDH-PIM
	Time (ns)	Time (ns)	Time (ns)
Worst Negative Slack (WNS)	19.030	19.249	19.215
Worst Hold Slack (WHS)	0.058	0.061	0.048
Worst Pulse Width Slack (WPWS)	9.725	9.725	9.725

4.4. Result summary

Table 4 shows the comparison summary of analyzed parameters. From all the results, 8-RDH-PIM is selected as the best modulation when implemented in hardware design as compared to 8-PPM and 8-DPIM. In terms of power consumption, hardware utilization and timing summary, most of them perform in almost similar nature. However, as transmission rate and bandwidth efficiency is concerns, 8-RDH-PIM outperforms both of them with 33.33 Mbps data transfer rate and 66.66% efficiency for 50 MHz input data.

Table 4. Result summary

Parameter	8-PPM	8-DPIM	8-RDH-PIM
Symbol transmission rate (Msymbol/s)	6.25	9.09	11.11
Data transmission rate (Mbps)	18.75	27.27	33.33
Bandwidth efficiency (5%)	37.5	54.54	66.66
Power allocation (W)	0.001	0.001	0.001
Hardware Utilization	83	66	77
Worst Negative Slack (WNS) (ns)	19.030	19.249	19.215

5. CONCLUSION

In this work, an 8-PPM, 8-DPIM, and 8-RDH-PIM modulator system has been implemented in Verilog hardware design language. The results indicate that the behavior of the algorithms follows the desired output for each modulation. All modulation basically has the same timing summary, hardware utilization and power consumption. 8-RDH-PIM however is chosen as the best modulation in this research since it can provide higher transmission rate and bandwidth efficiency as compared to 8-PPM and 8-DPIM. We believe, further development on RDHPIM on FPGA design can be done as to improve the overall design and performance especially for higher level modulation.

ACKNOWLEDGEMENTS

The authors would like to thank the Ministry of Education Malaysia for supporting this research under Fundamental Research Grant Scheme Vot No. FRGS/1/2018/ICT06/UTHM/02/1 and partially sponsored by Universiti Tun Hussein Onn Malaysia.

REFERENCES

- [1] R. M. Hagem, "FPGA Based Implementation of Pulse Position Modulation for Underwater Optical Wireless Communication," *Int. J. Eng. Innov. Technol.*, vol. 6, no. 5, pp. 47-50, 2016.
- [2] S. Chouhn and V. Kumar, "VHDL based FPGA Implementation of 256- ARYDPPM for," *Int. J. Electr. Electron. Data Commun.*, vol. 3, no. 2, pp. 42-45, 2015.
- [3] J. Liu, Z. Gu, B. Zheng, L. Zhao, and Z. Gong, "A Design of Underwater Wireless Laser Communication System based on PPM Modulating Method," *Ocean.-Shanghai*, pp. 1-6, 2016.
- [4] M. F. A. Ilyas, M. Othman, and S. Shah, "Enabling Toy Vehicles Interaction with Visible Light Communication (VLC)," *Adv. Sci. Technol. Eng. Syst. J.*, vol. 2, no. 3, pp. 210-216, 2017.
- [5] M. A. Ilyas, M. B. Othman and M. F. M. Ali, "Two toys vehicles interactions using communication protocol for visible light communication," *2016 IEEE Student Conference on Research and Development (SCoReD)*, Kuala Lumpur, 2016, pp. 1-6.
- [6] M. H. G. Ayagh and A. A. Varzanesh, "Implementation of DPIM Modem on FPGA and Comparison of it Functionality with PPM Modem," *Passiv. Def. Sci. Tech.*, vol. 39, no. 5, pp. 39-47, 2012.
- [7] G. Z. Yang, "A New Modulation Scheme of Visible Light Communication," *Optoelectron. Lett.*, vol. 10, no. 4, pp. 273-276, 2014.
- [8] N. Lourenço, D. Terra, N. Kumar, L. N. Alves, and R. L. Aguiar, "Visible Light Communication System for Outdoor Applications," *Proc. 2012 8th Int. Symp. Commun. Syst. Networks Digit. Signal Process. CSNDSP*, 2012.

- [9] J. Fridlander, C. Lee, J. S. Speck, S. P. DenBaars, J. Klamkin, and J. Klamkin, "Direct Pulse Position Modulation of a 410 nm Semipolar GaN Laser Diode for Space Optical Communications," *Conf. Lasers Electro-Optics*, vol. 1, no. c, p. STu4Q.3, 2018.
- [10] W. Xia *et al.*, "An Optical Pulse Position Modulation based on Double-Circle Iteration," *ICOON 2016 - 2016 15th Int. Conf. Opt. Commun. Networks*, pp. 15-17, 2017.
- [11] N. M. Aldibbiat, Z. F. Ghassemlooy, and R. McLaughlin, "Performance of Dual Header-Pulse Interval Modulation (DH-PIM) for Optical Wireless Communication Systems Nawras," *SPIE 4214, Opt. Wirel. Commun. III*, 144, vol. 44, no. 0, pp. 144-152, 2001.
- [12] N. M. Aldibbiat, Z. Ghassemlooy and R. McLaughlin, "Error performance of dual header pulse interval modulation (DH-PIM) in optical wireless communications," in *IEE Proceedings - Optoelectronics*, vol. 148, no. 2, pp. 91-96, April 2001.
- [13] T. Bostelmann and S. Sawitzki, "Towards a guided design flow for heterogeneous reconfigurable architectures," *2015 25th International Conference on Field Programmable Logic and Applications (FPL)*, London, 2015, pp. 1-2.
- [14] J. Roth and H. Charles, "Digital systems design using Verilog," Boston: Cengage Learning, 2016.
- [15] E. A. Bezerra and D. V. Lettnin, "Synthesizable VHDL design for FPGAs," vol. 9783319025. Cham: Springer, 2014.
- [16] P. Athanas, "Embedded systems design with FPGAS," New York: Springer, 2013.
- [17] J. Zhang, Y. Lin, Y. Lyu and G. Qu, "A PUF-FSM Binding Scheme for FPGA IP Protection and Pay-Per-Device Licensing," in *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 6, pp. 1137-1150, June 2015.
- [18] V. A. Pedroni, "Circuit design and simulation with VHDL," Cambridge, Mass.: MIT Press, 2010.
- [19] S. Brown, *Fundamentals of digital logic with Verilog design*. Dubuque: McGraw-Hill Higher Education, 2014.
- [20] J. M. P. Cardoso, "Reconfigurable computing : from FPGAs to hardware/software codesign," New York: Springer, 2011.
- [21] G. R. Smith, "FPGAs 101 : everything you need to know to get started," Amsterdam: Elsevier, 2010.
- [22] Xilinx, "UltraScale Architecture Configurable Logic Block," vol. 574, 2017.
- [23] M. A. L. Sarker and M. H. Lee, "Synthesis of VHDL Code for FPGA Design Flow using Xilinx PlanAhead Tool," *2012 Int. Conf. Educ. e-Learning Innov. ICEELI 2012*, 2012.
- [24] R. Sass, "Embedded systems design with platform FPGAs : principles and practices," Boston: Morgan Kaufmann, 2010.
- [25] J. Teubner, "Data processing on FPGAs," San Rafael: Morgan & Claypool, 2013.
- [26] P. Wilson, "Design recipes for FPGAs : using Verilog and VHDL," Amsterdam: Newnes, Elsevier, 2016.
- [27] W. Huang and L. Jia, "FPGA-based Matrix Keyboard Common IP Core Design and the Implementation Using Verilog HDL," *Proc.-2017 2nd Int. Conf. Mech. Control Comput. Eng. ICMCCE 2017*, vol. 2018-January, pp. 161-164, 2018.

BIOGRAPHIES OF AUTHORS



Mohammad Arif Bin Ilyas (M'26) was born in Malaysia in 1991. He received the B.E. degree from Universiti Tun Hussein Onn Malaysia in 2015. Currently he is completing his PhD in Electrical Engineering at the same University. His main areas of research interest are Optical wireless communication especially on visible light communication. Currently, his research involving around several digital modulation design using FPGA.



Maisara Othman received the B.Eng. (Hons.) Degree in computer system and communication engineering and M.Sc. Degree in communication network engineering from Universiti Putra Malaysia (UPM) in 2001 and 2005, respectively. She received her PhD. Degree in metro-access and short range system from DTU Fotonik, Technical University of Denmark in 2012. Currently, she is an Associate Professor and Deputy Dean (Student Affairs and Alumni) at Faculty of Electrical & Electronic Engineering (FKEE), UTHM. She is a member of IEEE. Her research interests include advanced modulation formats, photonic wireless integration and access and in-home network technologies.



Rahmat Bin Talib received the B.Eng. (Hons.) Degree in electric, electronic and system from Universiti Kebangsaan Malaysia (UKM) and M.Sc. Degree in electric-electronic and telecommunication from Universiti Teknologi Malaysia (UTM) in 1996 and 2004, respectively. He received his PhD. Degree in electrical engineering from Universiti Tun Hussein Onn Malaysia in 2015. Currently, He is lecturer at Faculty of Electrical & Electronic Engineering (FKEE), UTHM. His research interests mainly on optical and photonic systems.



Roshayati Binti Yahya @ Atan received the B.Eng. (Hons.) Degree in engineering (electrical-telecommunication) from Kolej Universiti Teknologi Tun Hussein Onn and M.Eng. Degree in engineering (electrical-electronic telecommunication) from Universiti Teknologi Malaysia (UTM) in 2006 and 2010, respectively. She received her PhD. degree in electrical engineering from Universiti Teknologi Malaysia (UTM) in 2016. Currently, she is lecturer at Faculty of Electrical & Electronic Engineering (FKEE), UTHM. Her research interests include Textile and flexible antenna, RF and microwave, Specific Absorption Rate (SAR), Microwave Imaging.



Maslina Yaacob received the B.Eng. (Hons.) Degree in engineering (electrical-telecommunication) and M.Eng. Degree in electrical engineering from Universiti Teknologi Malaysia (UTM) in 2007 and 2010, respectively. She received her PhD. degree in fiber optic sensing system from Universiti Teknologi Malaysia (UTM) in 2016. Currently, she is lecturer at Faculty of Electrical & Electronic Engineering (FKEE), UTHM. Her research interests include optical waveguide, optical device, optical sensor and spectroscopy.



Saizalmursidi Bin Md Mustam received the B.Eng. (Hons.) Degree in electronic communication from Kolej Universiti Teknologi Tun Hussein Onn and M.Sc. Degree in electrical engineering from Universiti Tun Hussein Onn Malaysia (UTHM) in 2006 and 2011, respectively. He received his PhD. Degree in electrical engineering from Universiti Teknologi Malaysia (UTM) in 2016. Currently, He is lecturer at Faculty of Electrical & Electronic Engineering (FKEE), UTHM. His research interests include Communication Engineering, Electromagnetic Compatibility, and Molecular Communication.



Marliana Jaafar received the B.Eng. (Hons.) Degree in electronic engineering and M.Eng. Degree in electrical engineering in 2014 and 2017 respectively from Universiti Tun Hussein Onn Malaysia (UTHM). She is currently pursuing the Ph.D. degree in electrical engineering with the UTHM. She is a member of IEEE and BEM. Her research interests include photonic system, modulation format, optical sensor and spectroscopy.



Mohd Rashidi Bin Che Beson received the B.Eng. (Hons.) Degree in Communication Engineering and M.Sc. Degree in Communication Engineering from Universiti Malaysia Perlis (UniMAP) in 2007 and 2011, respectively. He received his PhD. Degree in electrical engineering from Universiti Malaysia Perlis (UniMAP) in 2014. Currently, He is lecturer at Advanced Communication Engineering Centre (ACE), School of Computer and Communication Engineering, UniMAP. His research interests mainly on Optical CDMA Systems.