Wireless Personal Communications (2019) 104:471–489
https://doi.org/10.1007/s11277-018-6031-8

Parallel-Pipelined-Memory-Based Blowfish Design with Reduced FPGA Utilization for Secure ZigBee Real-Time Transmission

Rafidah Ahmad[1] · Daniel Kho[2] · Asrulnizam Abd. Manaf[1] · Widad Ismail[3]

Published online: 15 October 2018

Abstract
Data security is currently become a serious concern in wireless communication system for both the users and providers. Without a secure medium, the data transmission is exposed to various types of wireless attacks. Therefore, this paper focuses on the development of a high performance parallel-pipelined-memory-based (P^2M) Blowfish as a security design with reduced field programmable gate array (FPGA) utilization, which is the best security design to be embedded in the mobile devices. Through FPGA platform, the performance of the proposed Blowfish shows that a throughput increases by 10.5%, with the hardware utilization and power consumption decrease by 3.5% and 21%, respectively. The P^2M Blowfish was validated in two-way communication channel by using FPGA-based radio platform together with ZigBee technology and the real-time transmission was measured in terms of bit-error-rate, received power and communication range. These characteristics have proven that the proposed P^2M Blowfish possesses the ability to replace the advanced encryption standard which is known as a complex algorithm employed by most of the wireless communication standards.

Keywords Power-throughput · Blowfish · Field programmable gate array · ZigBee · Bit-error-rate

✉ Rafidah Ahmad
rafidah.ahmad@usm.my

Daniel Kho
daniel.kho@gmail.com

Asrulnizam Abd. Manaf
eeasrulnizam@usm.my

Widad Ismail
eewidad@usm.my

[1] Collaborative Microelectronic Design Excellence Centre (CEDEC), Engineering Campus, Universiti Sains Malaysia, 14300 Nibong Tebal, Penang, Malaysia

[2] Faculty of Engineering, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia

[3] Auto-ID Laboratory, School of Electrical and Electronic Engineering, Engineering Campus, Universiti Sains Malaysia, 14300 Nibong Tebal, Penang, Malaysia

1 Introduction

There are many types of wireless communication standard such as WiFi, WiMAX, Zigbee and Bluetooth [1]. Wireless communication involves data transferring between transmitter and receiver. While data is transferred between them, there is an existence of air interface that has the potential to be attacked by the outside threats. Hence, a device level security algorithm is needed as a protection against these wireless attacks [2]. There are several researchers believed that the security algorithm via the software platform is easy to be executed with a lot of advantages in its flexibility, portability, upgrading and easy usage. However, the security key and algorithm modification via software implementation are easier to be accessed by the attackers than the security algorithm through the hardware platform. The hardware-based security algorithm is thus safer and it is faster in speed than the software-based security algorithm [3].

The advanced encryption standard (AES) algorithm has been implemented by IEEE standard as a protection on data transmission for various types of wireless communication standards. AES is a symmetric encryption. It comprises of a few operations like SubBytes, ShiftRows, MixColumns and AddRoundKey that make the algorithm to be very complicated and uses high amount of power, memory and time. This creates a concern in a research studies nowadays where the system architecture of mobile devices becoming more complex in order to support many types of applications as smart devices. The researchers also need to find a way to save the battery power for a long usage. In order to conserve the energy, the complexity of the security algorithms needs to be reduced with less processing time [1].

The Blowfish algorithm consists of various length of key within the range of 32 bits to 448 bits, block size of 64-bit [1] and two main units which are the unit of data encryption and key expansion unit. It performs the best with the lowest usage of time and power according to the outcome found by [4–10] when compared with the performance of other security algorithms such as RC6, DES, 3DES and AES. In Blowfish algorithm, the text input of 64-bit is separated into two 32-bit halves as shown in Fig. 1. This algorithm employs P-array (P_1–P_{18}) during its execution. It comprises of 16 rounds with the Feistal (F) function and 18 sub-keys of 32-bit in its unit of key expansion [1, 11]. There are four *S-boxes* of 32-bit in the block of F function with each of the box contains 256 entries. Once the last round is over which is the 16th round, the data of two halves of 32-bit are joined back together to produce the encrypted result which is the ciphertext.

This paper proposes the development of a high performance Blowfish architecture with reduced hardware utilization by using a combination of three design techniques which are parallel-pipelined-memory-based (P^2M) design techniques. The proposed P^2M Blowfish was implemented in a real-time transmission on field-programmable gate array (FPGA)-based radio platform that comprises of Zynq-7000 XC7Z020 for processing the Blowfish system and ZigBee radio frequency (RF) module as a radio communication system. The Zynq-7000 family offers the flexibility and scalability of an FPGA while providing performance, power and easy of use, where they are perfect as a prototype of mobile devices [12, 13]. Therefore, with a low cost platform, the Zynq-7000 is used in this research work only as a medium to prove the functionality of the proposed P^2M Blowfish design in the real-time transmission over the air. The performance of P^2M Blowfish as a complete security radio system was analysed in terms of four parameters which are FPGA utilization, throughput, power consumption and real-time measurements. Meanwhile, the ZigBee RF module provides cost effective wireless connectivity based on Zigbee standard for the

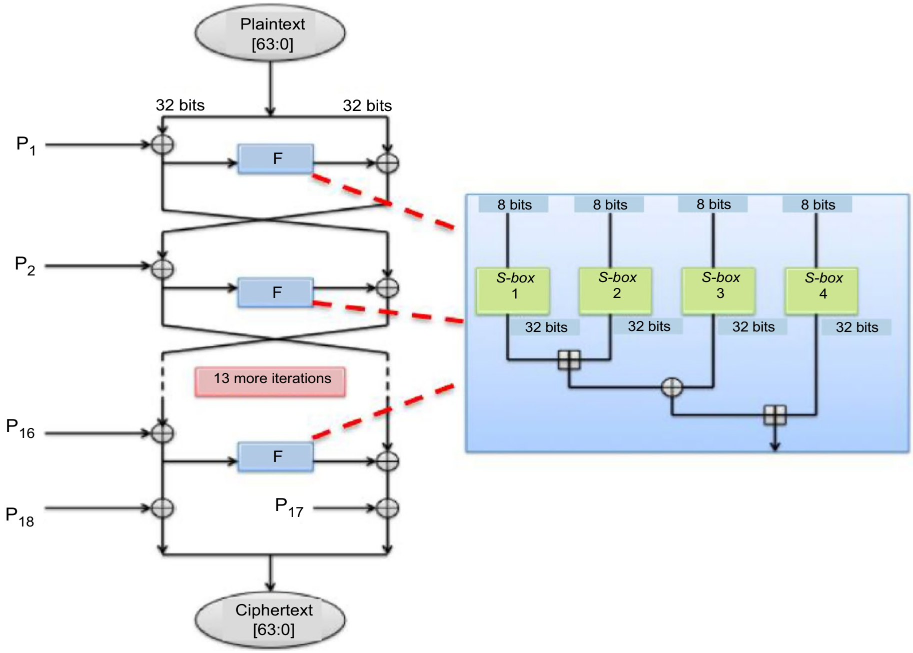


Fig. 1 Blowfish algorithm with F function [1]

electronic devices with no additional development is required [14]. This study is able to guide researchers to decide the possibility of implementing Blowfish for a secure wireless communication instead of AES.

2 Related Research

Many research studies have been done on Blowfish algorithm for the past few years. However, most of the researchers only simulated their designs either with very high speed integrated circuit hardware description language (VHDL), Verilog, Matlab, C program or Java. A few of them have further implemented their designs on FPGA, complex programmable logic device (CPLD) and advanced reduced-instruction-set-computer machines (ARM) as prototyping devices. However, there are very few researchers studied the implementation of Blowfish design on radio platform for real-time validation over wireless communication standard as shown in Table 1. The studies in [15–17] utilized microcontroller and Zigbee module for an integration with the Blowfish design as a radio platform. Without making a significant enhancement to the algorithm architecture, the authors claimed that the speed and power consumption of their design have been improved. The closest method of real-time transmission to this proposed work was carried out by Krishna et al. [18] that deployed Spartan3E FPGA and RF module in their work. However, their design required RF band transmission of 433 MHz. Certainly, the power consumption is reduced but with a slow data rate. Only Kalaiarasi et al. [19] has made an enhancement on the Blowfish architecture by using the EX-NOR instead of EX-XOR in F function with unified learning kit

Table 1 Design implementation of Blowfish algorithm on radio platform for real-time validation based on previous studies

Refs.	Design technique	Radio platfom	Improvement	Communication standard
[15]	Finite state machine (FSM), encryption mode only	CPLD, IRIS integrated ATMega 1281 microcontroller, Zigbee dongle	Reduce execution time and power consumption	2.4 GHz Zigbee
[16]	Standard architecture	PC, ATMega 328 microcontroller, MAX232, Zigbee module	Reduce processing time	2.4 GHz Zigbee
[17]	Standard architecture	Proteus software, PC, ATMega 8L microcontroller, Zigbee module CC2500	Improve network security	2.4 GHz Zigbee
[18]	Standard architecture	Spartan3E FPGA, RF module	Reduce power consumption	433 MHz
[19]	EX-NOR replaced the EX-XOR operation in F function	ULK with OMAP processor	Reduce processing time	2.4 GHz Zigbee

(ULK) and OMAP processor that reduced the processing time. However, all authors have not yet discussed BER result in their works. The BER test indicates the ability of the transmission system to accommodate the radio communication link characteristics.

Focusing on Blowfish architectures with 64-bit block size that were designed by using HDL and FPGA for performance comparison, Table 2 summarizes the output results performed by [3, 6, 20–23]. The results are based on FPGA utilization, throughput and power consumption. In terms of architecture, the work in [23] utilized the fewest amount of slices of only 5%. The memory-based for *S-boxes* was used in their design. The highest throughput of 1545.7 Mbps was obtained by [20] at 146.515 MHz clock frequency. However, its design slices is 61% larger than [23]. This design has deployed dynamic reconfiguration, replication, inner-loop pipelining, and loop folding techniques.

Table 2 also shows that the lowest power consumption is 66 mW by [3] with a throughput of 780 Mbps. The iterative method was used in their work to reduce the occupied area. There is only a single register required instead of a huge number of registers to give feedback to itself for each round. The Blowfish design was run with a clock frequency of 95 MHz. However, the slices used is 93%, which is the largest amount if slices compared with the other studies. Meanwhile, the highest power consumption was discovered by Dakate and Dubey [6] with a difference of 87% as compared with [3]. These findings also prove that a low throughput indicates a slow timing process and high power consumption.

3 Contribution of This Paper

This paper proposes a design of P^{2}M Blowfish with 128-bit block size since most of the research studies have shown the superiority of Blowfish algorithm over all other symmetric encryption algorithms including the AES that is practically used by IEEE standard in wireless communication today. This is necessary in order to make a fair performance comparison with the AES that also comprises of 128-bit block size. Therefore, the parallel technique is used in the Blowfish architecture to increase the throughput and reduce the power consumption.

Pipelined technique is also being implemented in the P^{2}M Blowfish design to control a few of internal signals. This is done to avoid critical paths so that the latency and power consumption can be minimized.

Another contribution of this paper is memory-based technique instead of only using registers to store the data of *S-boxes*. The block random access memory (BRAM) is used to implement large embedded storage blocks. The hardware resources can be reduced by using this method that leads to a smaller design core. The latency of the proposed Blowfish design can also be decreased.

4 Design Architecture of P^{2}M Blowfish

The P^{2}M Blowfish was designed by using HDL through Vivado 2015.2. Three design techniques as below have been used to increase the throughput and reduce the design core as well as power consumption. The maximum clock frequencies for Blowfish design during the synthesis and real-time transmission are 324 MHz and 100 MHz, respectively. This means that for software analysis, the clock frequency for the proposed design can be up to maximum 324 MHz without timing error. Meanwhile, for hardware analysis, the clock

Table 2 Performance comparison on 64-bit block size Blowfish designs based on previous studies

Refs.	Design technique	FPGA family	Design size (slices)	Throughput (Mbps)	Power consumption (mW)
[3]	Iterative	Virtex XCV50BG256-6	1608/1728 (93%)	780.0	66
[6]	Key generation	Altera Quartus II	NA	303.6	515
[20]	Dynamic reconfiguration, replication, loop pipelining, folding	Virtex2 2V500FG456-6	65%	1545.7	NA
[21]	Pipelined	Spartan3E XC3S500E – 5FG320	3222/4656 (69%)	386.1	NA
[21]	Memory-based	Virtex4 XC4VLX25-SF363	678/10752 (6%)	NA	NA
[22]	Memory-based	Virtex4 XC4VLX25-SF363	574/10752 (5%)	NA	NA

frequency is 100 MHz since this is a maximum value to avoid the timing issues in the data captured by the ChipScope which can cause the P^{2}M Blowfish design not to work at the physical level. Therefore, with a higher data rate, there is a possibility the encrypted data captured by the ChipScope during real-time transmission is missing at certain time and this can lead to a wrong data analysis.

4.1 Parallel Technique

Figure 2 shows the proposed P^{2}M Blowfish comprises of two parallel blocks of 64-bit Blowfish algorithm that are executed concurrently as dual-core. Basically, the input data of 128-bit is divided into two 64-bit data before it is processed by the Blowfish algorithm. The *mode* is used to select the encryption or decryption process. Both parallel blocks share the same *S-box* for F function. The 64-bit input data is represented by *w_data_i[63:0]* as depicted in Fig. 3a. Each Blowfish block has 64-bit key data length which is performed by *key_data_i[63:0]*. The output data from both Blowfish cores at *r_data_o[63:0]* are concatenated to obtain 128-bit encrypted or decrypted data. Overall, the 128-bit input and output data from Blowfish cores are phrased as below according to Fig. 3b:

$$\begin{aligned} Input\,data \;:\; wData\_i \;\leq\; \{(wData\_i2),(wData\_i1)\} \\ Output\,data \;:\; rData\_o \;\leq\; \{(rData\_o2),(rData\_o1)\} \end{aligned} \tag{1}$$

4.2 Pipelined Technique

Pipelining is an important concept to ensure the timing is accurate for various communications and control applications by avoiding the critical paths so that the latency and power consumption can be minimized. The communications interfaces that are used to transmit and receive the Blowfish encrypted data are fully pipelined. The RF transmission and reception of the encrypted data are done through ZigBee and universal asynchronous receiver-transmitter (UART) modules. The UART module is the main interface between the ZigBee module and Blowfish core on the FPGA platform where a proper pipelining is implemented. One of the pipeline registers is the UART data counter represented by *uart_data_cnt* as shown in Fig. 4. The pipeline of *uart_data_cnt* is *i_uart_data_cnt* to add

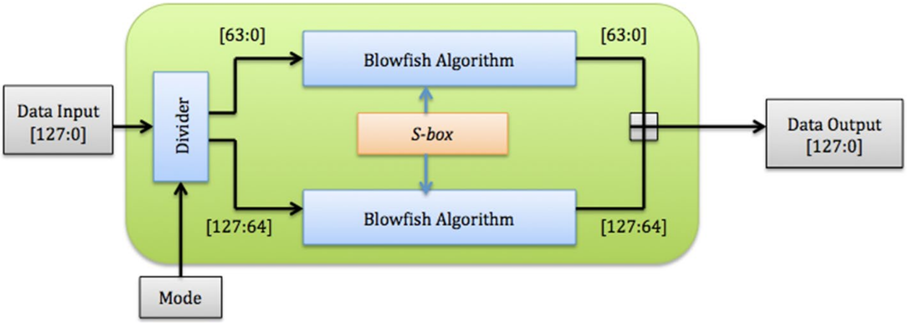
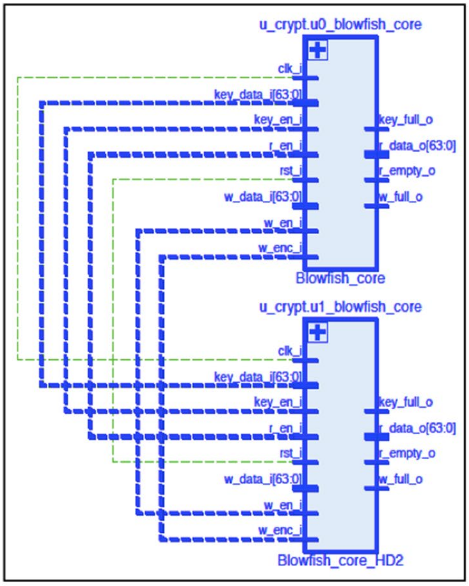
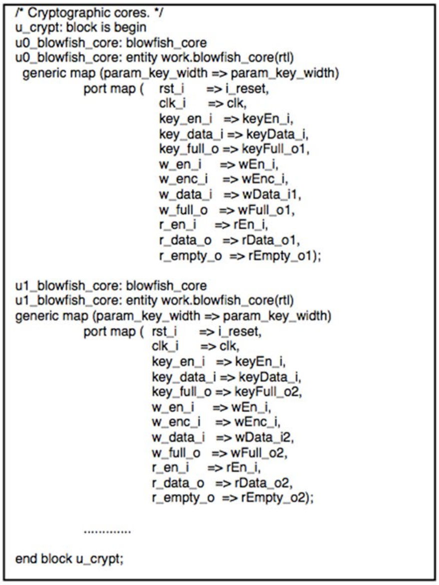


Fig. 2 Block diagram of the proposed P^{2}M Blowfish design [1]



(a)



```
/* Cryptographic cores. */
u_crypt: block is begin
u0_blowfish_core: blowfish_core
u0_blowfish_core: entity work.blowfish_core(rtl)
  generic map (param_key_width => param_key_width)
              port map (  rst_i      => i_reset,
                          clk_i      => clk,
                          key_en_i   => keyEn_i,
                          key_data_i => keyData_i,
                          key_full_o => keyFull_o1,
                          w_en_i     => wEn_i,
                          w_enc_i    => wEnc_i,
                          w_data_i   => wData_i1,
                          w_full_o   => wFull_o1,
                          r_en_i     => rEn_i,
                          r_data_o   => rData_o1,
                          r_empty_o  => rEmpty_o1);

u1_blowfish_core: blowfish_core
u1_blowfish_core: entity work.blowfish_core(rtl)
generic map (param_key_width => param_key_width)
              port map (  rst_i      => i_reset,
                        clk_i      => clk,
                        key_en_i   => keyEn_i,
                        key_data_i => keyData_i,
                        key_full_o => keyFull_o2,
                        w_en_i     => wEn_i,
                        w_enc_i    => wEnc_i,
                        w_data_i   => wData_i2,
                        w_full_o   => wFull_o2,
                        r_en_i     => rEn_i,
                        r_data_o   => rData_o2,
                        r_empty_o  => rEmpty_o2);

              ............

end block u_crypt;
```

(b)

Fig. 3 Parallel blocks of P^{2}M Blowfish: **a** schematic of 2 cores; **b** part of HDL code for 2 cores

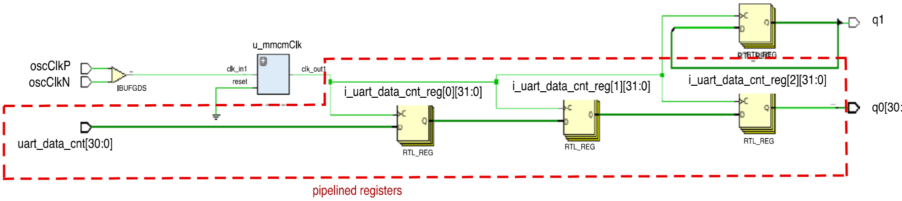
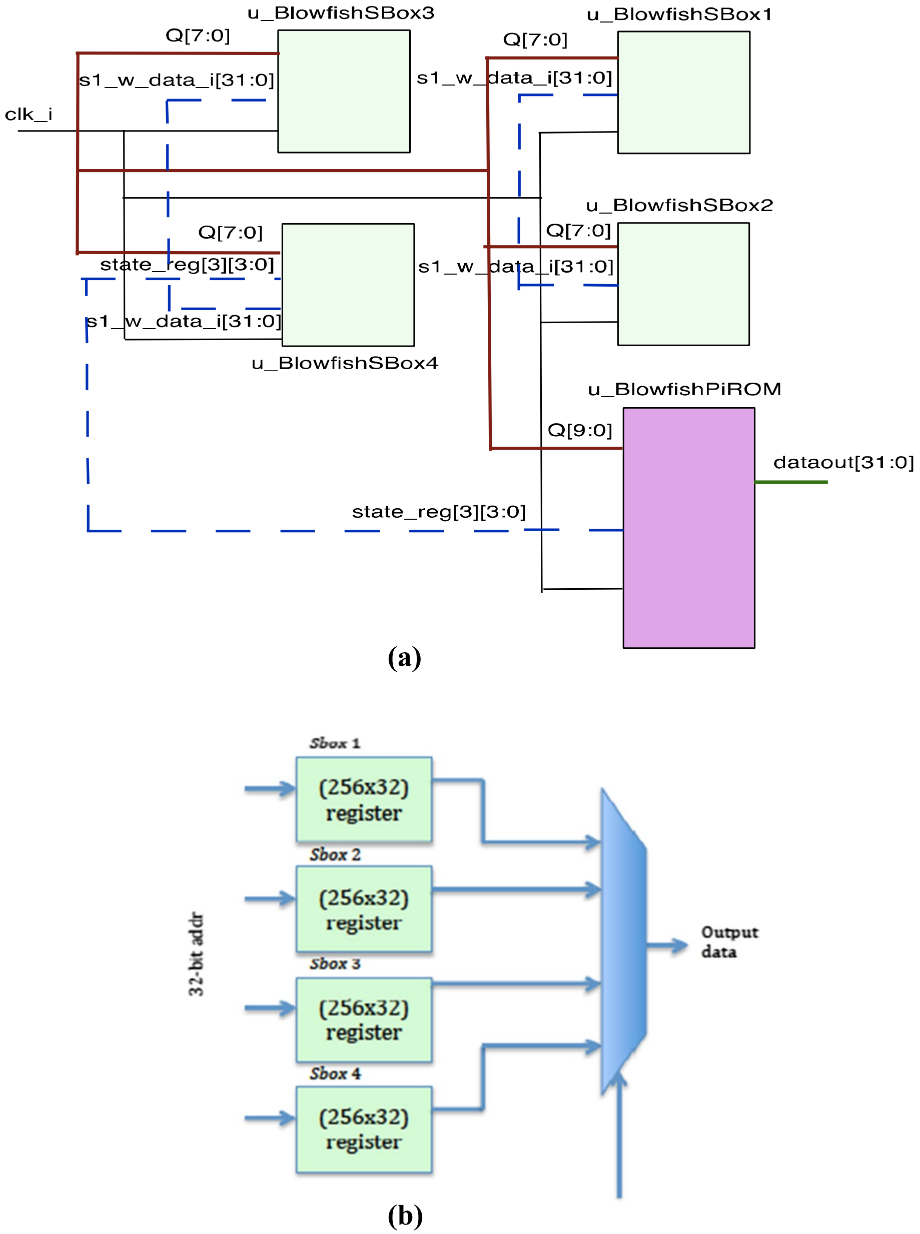


Fig. 4 UART data counter with pipelined technique in P^{2}M Blowfish design

a delay in the design which is important for a data synchronization between UART and external circuit from the ZigBee module as every logic block has its own delay. In this figure, *i_uart_data_cnt_reg[0]* is delayed by 1 clock cycle, then *i_uart_data_cnt_reg [1]* is delayed by 2 clock cycles and so forth. This pipelined technique is needed to obtain correct encrypted or decrypted data.

4.3 Memory-Based Technique

It is already shown in Fig. 2 that the parallel Blowfish blocks share the same four *S-boxes* known as S1, S2, S3 and S4 that contain 32-bit data with 256 entries each. The *S-box* is initialized with values derived from the hexadecimal digits of *π* which are D1310BA6, 98DFB5AC, 2FFD72DB and etc. The data from four *S-boxes* with a total of 1024 entries is stored into a block memory represented by *u_BlowfishPiROM*. There is only 2.5% BRAM utilized by these *S-boxes* and *u_BlowfishPiROM* during implementation. As shown in Fig. 5a with red line, the signals *Q[7:0]* from *u_BlowfishSBox1*, *u_BlowfishSBox2*, *u_BlowfishSBox3* and *u_BlowfishSBox4* are run concurrently with the signal *Q[9:0]* from the same *u_BlowfishPiROM* for the storage process of π data. Then, the 32-bit binary output data are read from *u_BlowfishPiROM* at a positive clock edge. It is found that by



(a)

(b)

Fig. 5 The structure of different techniques for S-boxes: **a** proposed memory-based technique; **b** conventional design technique

implementing the memory-based technique, the P^{2}M Blowfish design does not require a look-up table (LUT) as memory and the total of slices used could be reduced. This factor therefore leads to a faster encryption or decryption process and reduces the design core size. In Zynq-7000, a slice contains a set number of LUTs, flip-flops (FFs) and multiplexers that are used to perform logic, arithmetic and memory functions [24].

The proposed technique is quite different from the conventional technique as illustrated in Fig. 5b. The 256 × 32-bit registers are used for each *S-box*, which shows the involvement of many groups of FFs that store a bit pattern. A single register has a clock, input and output data. It also enables a signal port. The 32-bit input data of *addr* are latched and stored internally for every clock cycles. This method slows down the speed of the Blowfish performance as each register has its own timing delay.

5 Hardware Architecture

In order to evaluate the performance of the proposed P^{2}M Blowfish on the hardware implementation, two Zynq-7000 FPGA platforms are used as a transmitter and receiver. Since the Zynq-7000 integrates the ARM core and Xilinx programmable logic in a single device, this feature allows the proposed design to be enhanced with very high data storage. This is because an extra on-chip memory and external memory interfaces through the ARM core can support to store this data. Furthermore, the ARM core also supports for the USB as IO peripheral for computer connection to enable the data analysis. The hardware requirement can be determined through the FPGA implementation in terms of slices, FF, LUT, BRAM and input output (IO) block. Besides that, the power consumption can also be configured based on the hardware utilization. Meanwhile, the throughput can be analyzed based on the maximum clock frequency performed by the proposed design. In this work, the maximum clock frequency during the synthesis is 324 MHz implying that the proposed Blowfish encrypts and decrypts at a very fast speed.

To test and validate the proposed P^{2}M Blowfish in the real-time transmission, the experimental set-up using FPGA-based radio platform is depicted in Fig. 6. In this work, the UART module is used as a communication between the FPGA platform and ZigBee RF

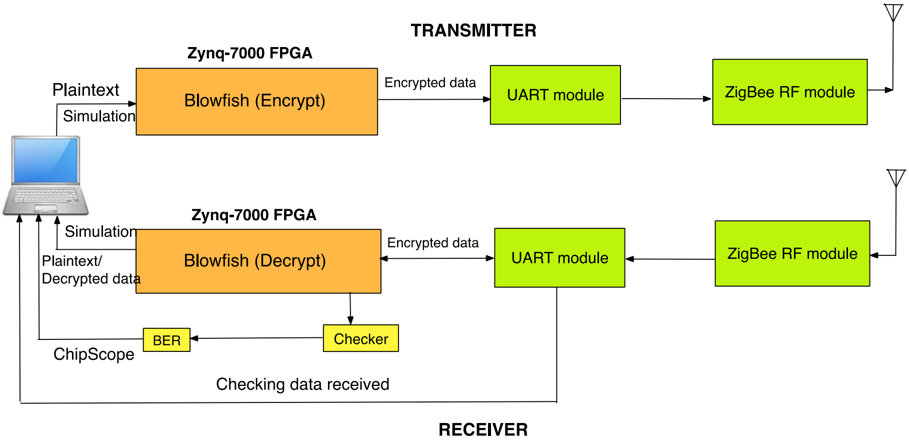


Fig. 6 Experimental set-up of the proposed P^{2}M Blowfish architecture using FPGA-based radio platform

module. This ZigBee module provides a cost effective wireless connectivity to the electronic devices [14]. It can be integrated with the FPGA through UART with a data rate of 250 Kbps at 2.4 GHz frequency band. As shown in Fig. 6, the encrypted data of 128-bit is transmitted for every 8-bit packet from the FPGA platform to the ZigBee module. The FPGA platform in the transmitter block stores the default data known as the test vectors data to activate the BER checker. Meanwhile, the FPGA platform in the receiver block stores the incoming data from the transmitter. This data are later checked against the data stored in the transmitter as reference data. If there is any occurrence of mismatching in the data, an error counter will be incremented.

During a real-time transmission, the encrypted data can only be captured by a ChipScope at a maximum clock frequency of 100 MHz in order to avoid timing error. The ChipScope tool integrates key logic analyzer as well as other test and measurement of the hardware components with a target design inside the supported Xilinx FPGA devices [25]. This tool communicates with these components and provides the designer with a robust logic analyzer solution [25].

6 Performance Comparison

In this section, the performance of P^{2}M Blowfish design is compared with the other Blowfish designs from previous studies. This section is categorized into four parts which are the FPGA utilization, throughput, power consumption and real-time measurements.

6.1 FPGA Utilization

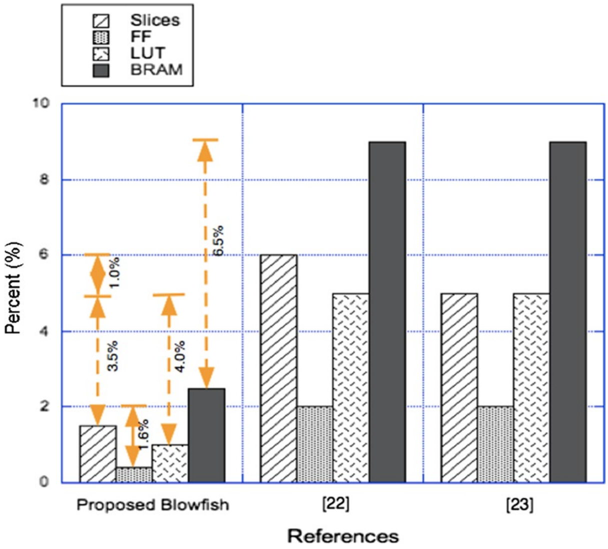
Table 3 presents the FPGA utilization of P^{2}M Blowfish architecture based on a post-implementation report from Vivado. The dual-core part of this Blowfish requires only 1.5% slices which is 3.5% less than the work in [23]. For Zynq-7000, four LUTs and eight FFs as well as multiplexers and arithmetic carrying logic form a slice where two slices form a configurable logic block [24]. One FF per LUT can optionally be configured as latches [24]. This Blowfish also uses 1% LUTs that is used for random logic implementation or distributed memory. The usage of FF is 0.4% and there is no input output (IO) block used since the dual-core architecture only involves internal signal for the data flow. The utilization of BRAM for *S-boxes* and *u_BlowfishPiROM* is about 2.5%. The comparison of hardware utilization between the dual-core of P^{2}M Blowfish design and previous research works is shown in Fig. 7. The outcome of this comparison shows that the proposed Blowfish design has the smallest design size with the least number of hardware requirements.

Since the proposed P^{2}M Blowfish design functions as a security radio system for real-time validation based on ZigBee standard, the total of slices and FF used increases to 14.3% and

Table 3 Architectural characteristic of the proposed P^{2}M Blowfish design

Architecture	Slices	FF	LUT	BRAM	IO
Dual-core	196/13300 (1.5%)	401/106400 (0.4%)	517/53200 (1%)	3.5/140 (2.5%)	0/200 (0%)
P^{2}M	1906/13300 (14.3%)	4820/106400 (4.5%)	4098/53200 (7.7%)	65.5/140 (46.8%)	18/200 (9%)

Fig. 7 Comparison of hardware utilization between the dual-core part of P^{2}M Blowfish design and previous research works



4.5% respectively as observed in Table 3. This is followed by the usage of 7.7% LUT. The utilization of these parameters increases due to the requirement of additional logic blocks needed for the UART, frequency divider and test vector modules. The usage of BRAM has also seen to be increased by 44.3% for the integrated logic analyzer (ILA) purpose. ILA requires the block memory to store a data when a certain condition is met. The size of this memory depends on the width and depth of data as IO which has been implemented in the design. The usage of IO blocks is measured to be totally 9%.

6.2 Throughput

In this paper, throughput is directed towards evaluating the characteristic and performance during synthesis of the proposed Blowfish architecture. Throughput is calculated as Eq. (2) based on [26].

$$\text{Throughput (Gbps)} = \frac{\text{128bits} * \text{Maximum Clock Frequency (MHz)}}{\text{Latency}} \tag{2}$$

Latency is the time interval between the starting of encrypted/decrypted per block data and the starting of output data, which is calculated in clock cycles [27]. In reliable two-way communication systems, latency limits the maximum rate that information can be transmitted. If a security algorithm is directed toward a device that wakes up, captures data, encrypts data, transmits data, and reverts to sleep mode, latency can become an issue because the longer the system needs to be awake, the more power is required [27]. Hence, latency should be as small as possible to achieve a power saving system [28]. Furthermore, longer battery lifetime is necessary, particularly for mobile devices.

The comparison of throughput per slice can be determined with the size and transmission speed. This procedure is the most objective method of comparing different security architectures [29]. The equation for throughput per slice is shown as below.

Table 4 Performance data of the proposed P^{2}M Blowfish design

Max. clock frequency (MHz)	Latency (cycle)	Throughput (Gbps)	Throughput/slice (Mbps/slice)
324	24.0	1.728	0.907

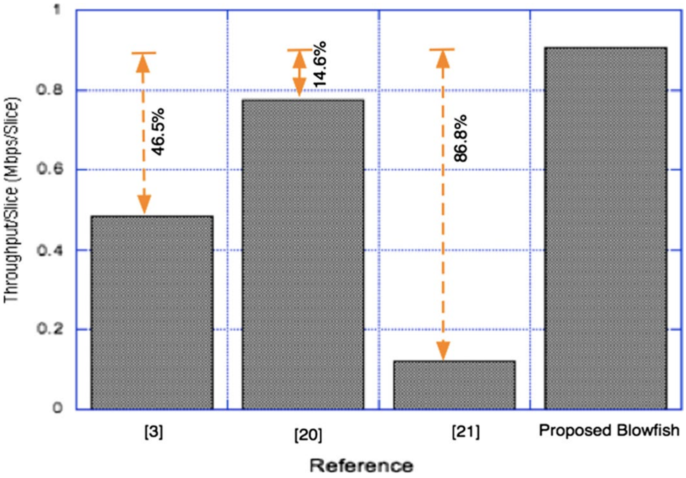


Fig. 8 Comparison of throughput per slice of the proposed P^{2}M Blowfish design with reference works

$$\text{Throughput/slice} = \frac{\text{Throughput (Gbps)}}{\text{No.of slices used}} \tag{3}$$

In Table 4, the proposed P^{2}M Blowfish has the smallest latency and the highest clock frequency if compared to the Blowfish designs of the previous works. The throughput and throughput per slice of the proposed Blowfish are 10.5% and 14.6% higher than [20], respectively. The highest throughput achieved by the proposed architecture indicates that it has the highest encryption speed and the best performance. Meanwhile, Fig. 8 shows that the throughput per slice for P^{2}M Blowfish design is 0.907 Mbps/slice, which is still higher than the previous research works.

6.3 Power Consumption

The power requirement for the proposed P^{2}M Blowfish architectures is discussed in this section. The Vivado Power tool was used to analyze the power consumption. In this paper, the dynamic power is analyzed. Dynamic power is associated with design activity and switching events in the core or IO of the device [30]. It is determined in terms of clock trees, logic, signals, BRAM, and IO power. This analysis introduces a very efficient method of locating the blocks or parts of the design that are the most deprived in the power aspect, thereby providing an easy path to power optimization [30].

Table 5 shows the comparison of power consumption performed between the P^{2}M Blowfish design and previous research works. In the proposed Blowfish design, there is

Table 5 Comparison of power consumption of the proposed P^{2}M Blowfish design with previous studies

Ref.	Throughput (Mbps)	Power consumption (mW)
[3]	780.0	66
[6]	303.6	515
P^{2}M Blowfish	1728.0	52

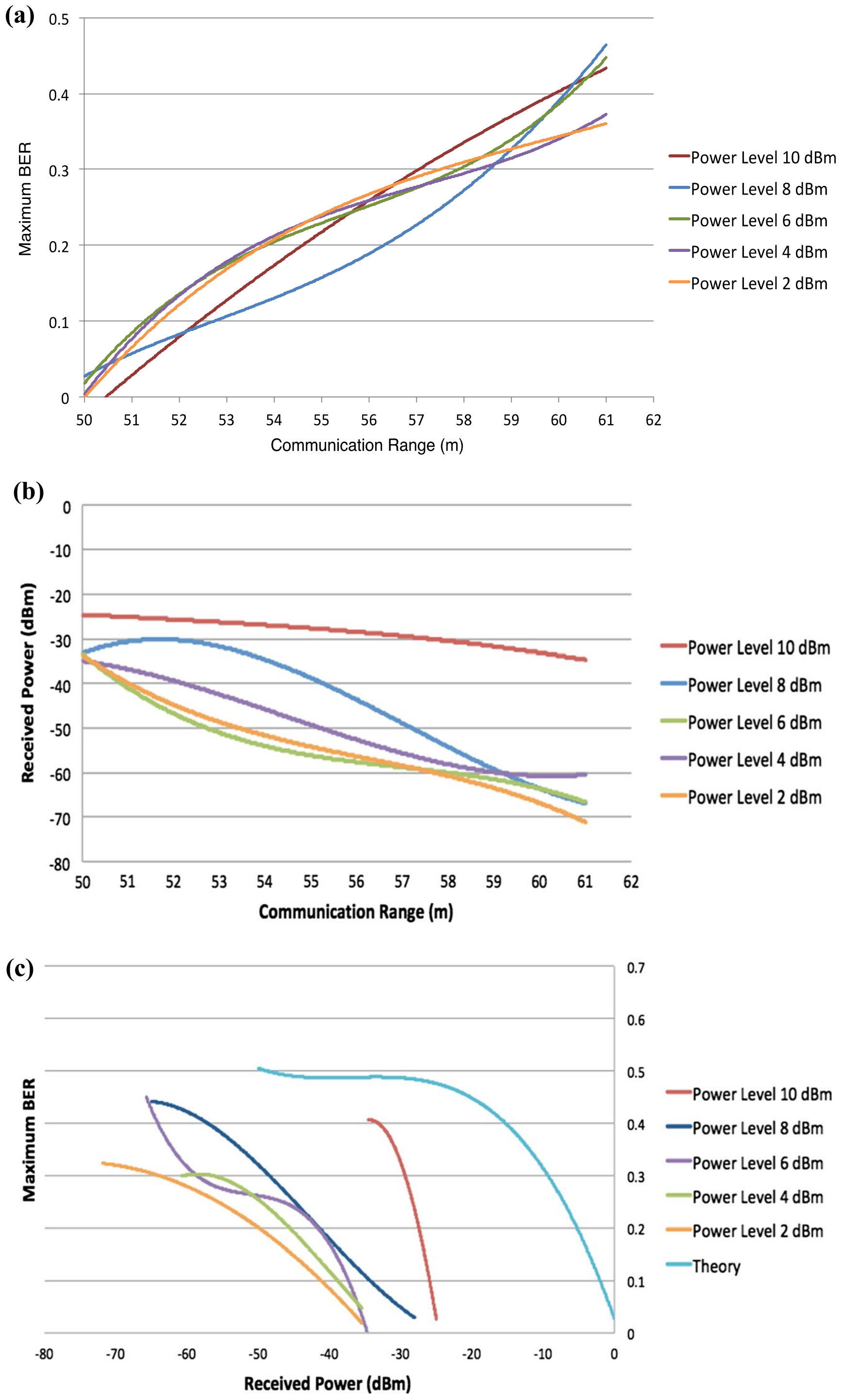
about 71% of the total power consumption is obtained from the IO blocks. However, there is no power consumed by BRAM although this design has utilized 46.8% BRAM. The power consumed by clock, logic and signal are 4 mW, 5 mW and 6 mW respectively. Overall, the total power consumed by the proposed Blowfish is 21% lower than the work in [3] with the highest throughput among the others. These findings indicate that the power-throughput of the proposed Blowfish design could be improved through the combination of parallel, pipelined and memory-based techniques.

6.4 Real-Time Measurements

The results for real-time measurements based on BER, received power and communication range in indoor environment are discussed in this section. Each data frame contains of (2 × 64)-bit with 30 addresses for data transmission through ZigBee module. The encrypted data is transmitted and received by FPGA platform through UART ports at 9600 baud rate with one stop bit, no parity and 8-bit data. The RF power for ZigBee module in transmitter part is set at five levels which are 2 dBm, 4 dBm, 6 dBm, 8 dBm and 10 dBm. Meanwhile, the RF power for Zigbee module in receiver part remains 10 dBm. The measurement of the proposed work is done up to maximum of 61 m in distance gap between the transmitter and receiver because there is no signal received after this distance as depicted in Fig. 9a. The data transmission with 10 dBm power level is discovered to be the best transmission since data error only start to occur at the distance of 52 m when it is compared to the other transmissions with lower power levels. It is also showed that even though the lowest BER at 61 m is only 33.9% with 2 dBm power level, the received power is the lowest among the others with −71.77 dBm as illustrated in Fig. 9b. The highest received power is −24.98 dBm with 10 dBm RF power at 50 m distance. Meanwhile, Fig. 9c shows that the maximum BER is 56.2% with a low received power of −63.99 dBm.

Generally, the increment of the communication range will affect the BER obtained to be higher and influence the received power to be lower as indicated in Fig. 9. This measurement also shows that the P^{2}M Blowfish design has a good performance for the indoor transmission since its BER is < 60% at 61 m with the received power of < −75 dBm.

Fig. 9 The results of real-time transmission based on BER, received power and communication range. **a** ▶ Maximum BER versus communication range (m); **b** received power (dBm) versus communication range (m); and **c** maximum BER versus received power (dBm)


(a)
Maximum BER
0.5
0.4
0.3
0.2
0.1
0
50
51
52
53
54
55
56
57
58
59
60
61
62
Communication Range (m)
Power Level 10 dBm
Power Level 8 dBm
Power Level 6 dBm
Power Level 4 dBm
Power Level 2 dBm
(b)
Received Power (dBm)
0
-10
-20
-30
-40
-50
-60
-70
-80
50
51
52
53
54
55
56
57
58
59
60
61
62
Communication Range (m)
Power Level 10 dBm
Power Level 8 dBm
Power Level 6 dBm
Power Level 4 dBm
Power Level 2 dBm
(c)
Maximum BER
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-80
-70
-60
-50
-40
-30
-20
-10
0
Received Power (dBm)
Power Level 10 dBm
Power Level 8 dBm
Power Level 6 dBm
Power Level 4 dBm
Power Level 2 dBm
Theory

7 Conclusion

The P^{2}M Blowfish design is proposed in this paper with real-time transmission over ZigBee standard using FPGA-based radio platform. This Blowfish was developed with three design techniques which are parallel, pipelined and memory-based. The proposed design was analyzed and compared with the other previous studies in terms of FPGA utilization, throughput, power consumption and real-time measurements. The results comparison indicates that the proposed Blowfish has reduced 3.5% FPGA utilization and 21% power consumption. This Blowfish also has a throughput of 10.5% and throughput per slice of 14.6% higher than other previous studies. Based on these findings, the P^{2}M Blowfish design possesses the best performance among the others. These parameters are important to fulfill the requirement of current research trend that is leading towards a compact mobile system with higher data transmission speed and longer battery usage than the existing one in the market. The results of the real-time transmission in terms of BER, received power and communication range were also analyzed to determine the quality of a digital transmission system of the proposed Blowfish. The results show that the encrypted data can be transmitted without any error with a distance up to 50 m in an indoor environment.

Acknowledgements This research was financially supported by the Ministry of Science, Technology, and Innovation (MOSTI) ScienceFund Research Grant (Project No. 06-01-05-SF0640) and the Universiti Sains Malaysia Research University Grant for Individuals (RUI) (Project No. 1001/PELECT/814241).

References

1. Ahmad, R., Manaf, A. A., & Ismail, W. (2016). Implementation of a high-performance Blowfish for secure wireless communication. *Journal of Telecommunication, Electronic and Computer Engineering, 8*(6), 147–151.
2. Ahmad, R., & Ismail, W. (2013). A survey of high performance cryptography algorithms for WiMAX applications using SDR. In A. Al-Dulaimi, J. Cosmas, & A. Mohammed (Eds.), *Self-organization and green applications in cognitive radio networks. Chapter 11* (pp. 231–246). Hershey: IGI-Global.
3. Karthigaikumar, P., & Baskaran, K. (2010). Partially pipelined VLSI implementation of blowfish encryption/decryption algorithm. *International Journal of Image and Graphics, 10*(3), 327–341.
4. Elminaam, D. S. A., Kader, H. M. A., & Hadhoud, M. M. (2010). Evaluating the performance of symmetric encryption algorithms. *International Journal of Network Security, 10*(3), 213–219.
5. Thakur, J., & Kumar, N. (2011). DES, AES and blowfish: Symmetric key cryptography algorithms simulation based performance analysis. *International Journal of Emerging Technology and Advanced Engineering, 1*(2), 6–12.
6. Dakate, D. K., & Dubey, P. (2012). Performance comparison of symmetric data encryption techniques. *International Journal of Advanced Research in Computer Engineering & Technology, 1*(4), 163–166.
7. Kumar, A., & Karthikeyan, S. (2012). Investigating the efficiency of blowfish and rejindael (AES) algorithms. *International Journal Computer Network and Information Security, 2,* 22–28.
8. Mandal, P. C. (2012). Superiority of blowfish algorithm. *International Journal of Advanced Research in Computer Science and Software Engineering, 2*(9), 196–201.
9. Haldankar, C., & Kuwelkar, S. (2014). Implementation of AES and blowfish algorithm. *International Journal of Research in Engineering and Technology., 3*(3), 143–146.
10. Harinath, D., Murthy, M. V. R., & Chitra, B. (2015). Cryptographic methods and performance analysis of data encryption algorithms in network security. *International Journal of Advanced Research in Computer Science and Software Engineering, 5*(7), 680–688.
11. Schneier, B., & Whiting, D. (1997). Fast software encryption: Designing encryption for optimal speed on the Intel Pentium processor. In *Proceedings of 4th international workshop on fast software encryption. LNCS* (pp. 242–259). Berlin: Springer.

12. Ahmad, R., & Ismail, W. (2014). Implementation of high performance advanced encryption standard-128 for WiMAX application on FPGA. In *Proceedings of the 2nd IEEE international symposium on telecommunication technologies* (*ISTT2014*) (pp. 326–331), Langkawi.
13. Xilinx, Inc. (2015). Zynq-7000 all programmable SoC overview. Product Specification, DS190, v(1.8).
14. Digi International, Inc. (2015). *XBee and XBee-PRO Zigbee: Embedded Zigbee modules provide OEMs with a simple way to integrate mesh technology into their application* (pp. 1–3). Minnetonka: Digi International.
15. Mplemenos, G. G., Papadopoulos, K., & Papaefstathiou, I. (2010). Using reconfigurable hardware devices in WSNs for reducing the energy consumption of routing and security tasks. In *Proceedings of IEEE global telecommunications conference* (*Globecom 2010*) (pp. 1–5).
16. Arur, P. C., Chandrasekhar, M. S., Sreeram, S. S., Ramkishore, K., & Gopal, C. V. (2014). Secure data transmission using Blowfish algorithm. *International Journal of Innovative Research in Science and Engineering*, *2*(5), 2347–3207.
17. Priyadharshini, S. P., Arumuagam, N., & Ananthamani, K. S. (2014). Implementation of security in wireless sensor network using blowfish algorithm. In *Proceedings of International Conference on Innovations in Information, Embedded and Communication* (pp. 33–37). ICIIECS.
18. Krishna, B. M., Varshini, J. S., Murthy, A. N., Santosh, N. A., Kumar, G. S. P., & Rao, B. S. V. (2015). RF module based wireless secured home automation system using FPGA. *Journal of Theoretical and Applied Information Technology, 77*(2), 273–279.
19. Kalaiarasi, D., & Prathipa, R. (2016). Zigbee based secured wireless transmission using advanced Blowfish cryptographic algorithm. *Australian Journal of Basic and Applied Sciences, 10*(1), 332–336.
20. Sudarshan, T. S. B., Mir, R. A., & Vijayalakshmi, S. (2005). *DRIL-A flexible architecture for Blowfish algorithm encryption using dynamic reconfiguration, replication, inner-loop pipelining, loop folding techniques. Lecture notes in computer science* (*including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics*) (pp. 625–639). Berlin: Springer.
21. Chatterjee, S. R., Majumder, S., & Pramanik, B. (2014). FPGA implementation of pipelined blowfish algorithm. In *Proceedings of 5th international symposium electronic system design* (pp. 208–209).
22. Prasetyo, K. N., Purwanto, Y., & Darlis, D. (2014). An implementation of data encryption for internet of things using blowfish algorithm on FPGA. In *Proceedings of international conference on information and communication technology* (*ICoICT 2014*) (pp. 75–79).
23. Bansal, V. P., & Jassal, P. S. (2016). Synthesis and analysis of 64-bit blowfish algorithm using VHDL. *International Journal of Engineering Sciences, 17*(1), 316–322.
24. Xilinx, Inc. (2012). Virtex6 family overview. Product specification, DS150, v2.4, USA.
25. Xilinx, Inc. (2012). ChipsScope Pro software and cores: User guide, UG029, v14.1, USA.
26. Elbirt, A. J., Yip, W. Chetwynd, B. & Paar, C. (2000). An FPGA implementation and performance evaluation of the AES block cipher candidate algorithm finalists. In *Proceedings of the third advanced encryption standard candidate conference* (pp. 13–27). New York: National Institute of Standards and Technology (NIST).
27. Dyken, J., & Delgado-Frias, J. G. (2010). FPGA schemes for minimizing the power-throughput trade-off in executing the Advanced Encryption Standard algorithm. *Journal of Systems Architecture, 56,* 116–123.
28. Ahmad R., & Ismail W. (2016). Performance comparison of the improved power-throughput AES and Blowfish algorithms on FPGA. In *Proceedings of the 9th international conference on robotics, vision, signal processing & power applications* (*ROVISP2016*) (pp. 75–82), Penang.
29. Ahmad, R., & Ismail, W. (2016). Performance comparison of advanced encryption standard-128 algorithms for WiMAX application with improved power-throughput. *Journal of Engineering Science and Technology (JESTEC), 11*(12), 1–17.
30. Xilinx, Inc. (2011). *Power methodology guide, UG786, v13.1* (pp. 8–9), USA.

Rafidah Ahmad received the B.Eng. degree and M.S. degree in Electrical and Electronic Engineering from Universiti Sains Malaysia (USM), Penang, Malaysia, in 2001 and 2005, respectively. She is currently pursuing the Ph.D. degree in Electrical and Electronic Engineering at USM. Currently, she is a Senior Research Officer with Collaborative Microelectronic Design Excellence Centre (CEDEC), USM since 2005. Her research interest includes the development of cryptography for wireless communication and digital signal processing with ASIC and FPGA. To date she has produces more than 20 international publications as the first author including book chapter, journal papers, conference papers and lecture note.

Daniel C. K. Kho holds a B.Eng. from Northumbria University, U.K., and is also currently working on hardware-based image processing as part of his on-going Master's research at Multimedia University, Malaysia. He is a registered Certified Professional Trainer and a Certified International Professional Manager of the IPMA, United Kingdom. He spent close to 15 years working on microelectronics design and test. Throughout his career, he has designed and verified safety- and mission-critical digital signal processing (DSP) and reduced-instruction-set computer (RISC) processor subsystems, some of which have either been deployed in some of the world's best-selling systems, or used to enable the rapid deployment of such systems. He has deep experience designing complex SoC subsystems and bus interfaces, and pioneered hardware-based transaction-level modeling. He is experienced in both FPGA and ASIC design. He also serves as the CTO for Tauhop Solutions, working on several new DSP and imaging projects. He trains and provides consulting to multinational companies in the area of digital design and simulations, SoC bus architecture design, and DSP. He enjoys sharing his experience in making complex designs and testbenches easy and fun. He has also contributed to the IEEE P1076-2008 VHDL standard, and continues working towards advancing the microelectronics industry.

Asrulnizam Abd. Manaf received the Bachelor Engineering in Electrical and Electronic Engineering from Toyohashi University of Technology, Japan in 2001. Then, he worked as Electrical Engineer at Toyo-Memory Technology Sdn. Bhd at Kulim High Tech Park, MALAYSIA before he further his master degree at Toyohashi University of Technology, Japan. He received Master Engineering in Electrical and Electronic Engineering in 2005. He pursued his Ph.D study in Keio University, Japan in 2006. He received Ph.D in Engineering from Department of Applied Physic and Physico Informatics, School of Fundamental Science and Technology, Keio University Japan in 2009. Since 2009, he joined the school of Electrical and Electronic Engineering, Universiti Sains Malaysia as a senior lecturer. Then, promoted to Associate Professor in 2015. He has graduated 7 Ph.D students and 16 MSc students. Currently, 3 Ph.D students and 2 MSc under supervision. He has authored and co-authored 60 international technical journal or conference papers. He won several innovation award at national and International Invention competition. Currently he has 2 patent filings for DNA sensor and CMOS based tuneable inductor. His current research interest includes development of microfluidic-based DNA sensor integrated with CMOS circuitry, miniaturized of fluidic-based inclination sensor, bio inspired based microfluidic acoustic, pressure and flow sensor for underwater system, micro fluidic based memristor, micro fluidic based tuneable inductor, micro fluidic Thermoelectric Generator (mTEG)-based energy harvesting, Graphene-based transistor and micro 3-dimension fabrication technique by using grayscale Technology. From 1st January 2016, he transferred to Collaborative Microelectronic Design Excellence Center (CEDEC), Universiti Sains Malaysia.

Widad Ismail is a Professor and the Project Coordinator for the Auto-ID Laboratory (AIDL), Universiti Sains Malaysia (USM), Penang, Malaysia. She received her B.Eng (H) First Class Honors in Electronics and Communication Engineering from The University of Huddersfield, United Kingdom in 1999. By 2004, she completed her Ph.D. in Electronics and Communication Engineering specializing in Active Integrated Antenna (AIA) with Image Rejection from the University of Birmingham, United Kingdom. Since year 2000, she served as a Postgraduate Teaching Assistant at the university for three years. Once graduated, she started her career at USM as a lecturer until today. She was appointed as professor in the year 2014 at the School Electrical & Electronics Engineering, USM. Her main areas of research are wireless sensor and system design, RFID (Radio Frequency Identification), active integrated antennas (AIA) and RF and microwave systems engineering. Her research and scientific outputs have been translated to numbers of awards, publications and patents. To date, she is a Principal Investigator for 26 research grants. These research works have produced 8 filed patents, 10 international awards, 4 commercialized main research products and more than 150 publications including the international journal papers, conference/seminars and other publications. Furthermore, several incomes are received to the University mainly from the Commercialization of research innovative products and also the services as a principle consultant. In addition, there are more than 35 consultations and collaborations that have been established with various agencies and institutions which bridging the gap between the academicians to the industrialists. Currently, she is the main supervisor of 13 PhD students (active candidature) and 4 Master's by research students and she has graduated a total of 12 PhD and 12 Master's students under her supervision and guidance. On top of these, she is a member of Wireless World Research Forum (WWRF).