Dynamically Reconfigurable Power Efficient Security for Internet of Things Devices

Khaled Khatib*, Mostafa Ahmed*, Ahmed Kamaleldin†, Mohamed Abdelghany‡ and Hassan Mostafa†
*Electronics Department, German University in Cairo, Egypt.
‡Electronics Department, German University in Cairo, Egypt. Integrated Electronic Systems Lab, TU Darmstadt, Germany.
†Electronics and Communications Engineering Department, Cairo University, Giza 12613, Egypt.
Emails: kt.khateb@gmail.com, sha3ban.soliman@gmail.com, ah.kamal.ahmed@gmail.com,
mohamed.abdel-ghany@guc.edu.eg, hmostafa@uwaterloo.ca

***Abstract*—Internet of Things (IoT) is rapidly gaining ground as one of the most important technologies of our time with security being one of it's biggest challenges. Two designs are proposed within this paper to tackle the IOT security problem. The first design is a power adaptive encryption solution that adapts to the available power budget by selecting one of four encryption algorithms: AES-256, AES-192, AES-128, or DES. This allows the data to be secured even under a strict power budget, which was not applicable using conventional designs. The second design implements NDES without using any more resources than those conventionally used by single DES. This provides an average decrease in resource utilization by 54% and a decrease of 63% in the power consumption in 3DES configuration. Both designs use Dynamic Partial Reconfiguration (DPR) technology to allow the switching between the configurations using minimal area and power consumption. The proposed designs were implemented on a ZedBoard development board after being synthesized using Xilinx Vivado 2015.2.**

***Index Terms*—AES, DES, IoT, power adaptive, encryption, DPR, FPGA**

I. Introduction

IoT is a novel paradigm that is expected to improve the quality of daily life by equipping all kinds of objects with the needed microcontrollers, transceivers, sensors, and software to allow these objects to communicate with each other and share useful data [1]. Additionally, IoT is expected to offer numerous solutions to improve upon many of the current manufacturing, transportation and other industrial systems [2]. This is why IoT is included by US National Intelligence Council (NIC) in the list of six "Disruptive Civil Technologies" potentially having a great impact on the future of the US. NIC foresees that "by 2025 internet nodes may reside in everyday things – food packages, furniture, paper documents, and more". It highlights the idea that in the future, the IoT will probably be as widespread as the internet is nowadays [3].

One of the most important factors about building an IoT device is implementing a low power design. The importance stems from the fact that nearly all IoT devices depend on either small batteries or renewable energy sources. IoT devices operating on limited capacity batteries are expected to keep operational for the maximum amount of time. Since the power provided by renewable energy sources usually depends on time of day, time of year, weather conditions, and the location of operation, IoT devices operating on renewable energy sources need to adapt to the amount of available power. This is why these devices need to rely on power saving modes when the available power is not enough to operate at full capacity. Providing a device with low power consumption is considered of more importance than providing a secure one. This is why in current implementations when power saving mode is activated, security and encryption modules are sacrificed. Consequently, unencrypted messages are sent instead of halting communication due to the low power available.

The objective of this paper is to propose two IoT encryption designs. The power adaptive encryption design allows data encryption, albeit using weak encryption, when the power budget is low, and provide strong encryption when a high power budget is available. The second design is the NDES encryption design, in which the number of DES implementations can be chosen to provide additional security without using any more resources than single DES.

This paper is organized as follows: Section II provides an overview about the encryption algorithms used within the proposed designs. Section III explains current DPR technologies. Section IV presents the two proposed designs and explains in detail how they were implemented. Section V presents the reached results. Sections VI concludes the paper and discusses the results.

II. Data Encryption

A. Advanced Encryption Standard

The Advanced Encryption Standard (AES) is a specification for electronic data encryption that was established by National Institute of Standards and Technology (NIST) in 2001 after selecting between fifteen competing symmetric key algorithms. It has since been adopted by the U.S. government and is now used worldwide superseding the Data Encryption Standard (DES).

AES is a symmetric key algorithm, which means that the same key is used for both encrypting and decrypting the data. The cipher works by repeating specific functions for a number of rounds. These functions are byte substitution, shift rows, mix columns and round key addition. The key length used specifies the number of rounds to be made. The number of rounds are 10, 12, and 14 for key sizes of 128, 192, and 256 respectively [4].

978-1-5386-4788-2/18/$31.00

B. Data Encryption Standard

DES was introduced in the early 1970s and standardized in 1977. It was a collaboration between National Security Agency (NSA) and IBM to become the encryption standard used by the US Government to encrypt sensitive data.

DES is based on a feistel network. It is a block cipher with a block size of 64 bits, an effective key length of 56 bits and is made up of 16 similar rounds. The encryption and decryption have the same steps with the only difference being the key schedule [5].

Due to its short key size, it was proven vulnerable to brute force attacks in the late 1990s. Hence, 3DES was introduced where DES is applied 3 consecutive times with 3 or 2 different keys. 3DES increases the security with an effective key size of 112 bits. 3DES was however proven to be vulnerable to linear cryptanalysis, but in order for it to be applied, an enormous amount of pairs of plaintext and cipher text need to be known to the attacker [6].

III. Dynamic Partial Reconfiguration

A. Definition

DPR is a technology that allows for a specific set of modules to be dynamically changed during runtime without the interruption of the remaining logic [7]. The DPR design flow requires partitioning the system into static and dynamic parts. The dynamic part contains a set of Reconfigurable Partition (RP) and each RP has a set of Reconfigurable Module (RM) which can be swapped during runtime [8].

Initially, a full BIT file configures the Field Programmable Gate Array (FPGA), setting the default RMs to be used when the design is first run. When it is desired to change the RM loaded on a certain RP during runtime, a partial BIT file containing the data for the desired RM can be loaded onto the FPGA without affecting the operation of other parts of the design.

B. Dynamic Partial Reconfiguration Controllers

Several DPR controller designs are proposed to reach the maximum reconfiguration throughput. [8] compared between four of these controllers and the results reached are expressed in Fig. 1. These controllers are the HWICAP, Xilinx Partial Reconfiguration Controller (PRC), ZYCAP [9], and PCAP.

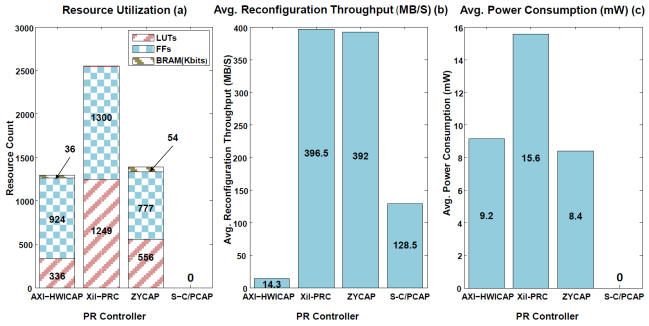

Fig. 1. Resource Utilization, Avg. Reconfiguration Throughput, and Power Consumption Comparisons between Different PR Controllers [8]

IV. Proposed Designs

A. Power Adaptive Encryption Design

The proposed design aims to provide a power adaptive encryption solution that is suitable for IoT applications. The main reason why such a solution is believed to be useful is that many IoT devices are powered using renewable energy sources such as wind or sunlight. This means that the level of power available for the device is variable according to the time of day and weather conditions. Within previously available systems, the only solution when the power is at a critical level is to shutdown the encryption circuit and communicate unencrypted data. However, in the proposed design, DPR technology allows one of four encryption levels to be chosen. The choice is done according to the amount of energy that can be spared for each encryption operation. These levels are: AES-256, AES-192, AES-128 and DES.

This would allow data transmission to still be secured even under tight power budgets.

As explained in section III, the basic design has a static and a dynamic part. The static design contains the top module that controls the three RPs (key logic, key memory and encryption round) and gives each of them signals to start working. Each of the three RPs has a number of RMs that can be configured on it depending on the triggers given to the DPR controller. The three RPs and their RMs are shown in Fig. 2 and are the following:

Key Logic is responsible for performing the key schedule and producing the sub-keys used in each round. It has five RMs that can be configured on it which are: KL-256, KL-192, KL-128, KL-DES and Blank

- **KL-256** performs the key schedule for AES-256.
- **KL-192** performs the key schedule for AES-192.
- **KL-128** performs the key schedule for AES-128.
- **KL-DES** performs the key schedule for DES.
- **Blank** is a blanking configuration used when the key schedule is already completed to reduce power consumption.

Key Memory is responsible for storing the sub-keys produced by the key logic RP. It has two RMs which are: KM1 and KM2

- **KM1** stores sub-keys for the AES-128 and the DES key schedule.
- **KM2** stores sub-keys for the AES-256 and the AES-192 key schedule.

Encryption Round is responsible for performing the encryption rounds for the given algorithm. The two RMs that can be configured on it are: AES and DES

- **AES** is responsible for performing encryption using AES algorithm of either of the three key sizes (128, 192 or 256 bits).
- **DES** is responsible for performing encryption using DES algorithm.

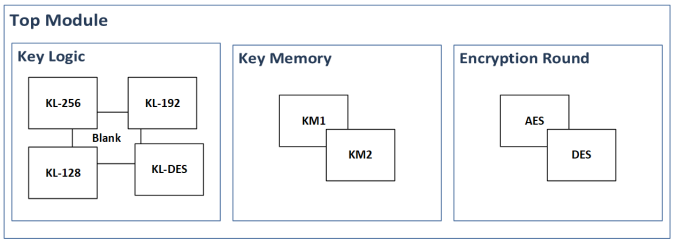

Fig. 2. Power adaptive encryption design blocks

At the beginning of the operation, key logic RP starts execution based on the RM configured on it. It produces the sub-keys required to complete the encryption process and saves them to the key memory RP. Once the required sub-keys are stored in the key memory RP, there is no need to reactivate the key logic RP and thus the blank RM is loaded on it. This is assuming that the same input key is used for all executions of an algorithm which is the case most of the time. A new key can however be used for the same encryption algorithm by resetting the system using the reset input.

The top module is responsible for starting the key schedule and the encryption rounds at the correct time either during the first execution of the algorithm or any following execution. The key memory was used instead of on the fly calculation of sub-keys every time the algorithm is run to allow the key schedule RP to be blanked and thus decrease the consumed power. The choice of two key memory RMs with each RM containing two algorithms came after deep consideration of the normal usage schemes. The key memory RP design originally had four RMs with each saving sub-keys of one algorithm, but that was found to be very power inefficient and would have additionally leaded to a more complex DPR trigger controller.

Resource reutilization was used to be able to implement the design utilizing minimal area and thus using minimal power. This is why the encryption round components were only implemented once as a combinational circuit, and the output was re-fed into the input for the required number of rounds for each algorithm. This was implemented in both the AES and DES rounds.

B. NDES Encryption Design

The proposed design has higher security than 3DES and consumes less power. That is achieved by combining NDES and DPR technology. The main role of DPR is to alternate the mode of operation between encryption and decryption. This means that instead of having N different modules that consume lots of power and area, only one module is needed and its mode is changed between encryption and decryption. Normally, DES has two modules for key scheduling; one for encryption and the other for decryption with a MUX choosing the needed configuration thus increasing the power and area consumption. Since DPR is being used to change the mode between encryption and decryption, only one module is needed to be working at a single instant, decreasing the power.

NDES is the application of DES algorithm N consecutive times to the data block of plaintext, alternating between encryption and decryption modes similar to the alternation in 3DES (encryption $\rightarrow$ decryption $\rightarrow$ encryption). The number of times NDES can be applied is only limited by the time that is allowed for encryption or decryption. Every time DES is applied when NDES is used, it can be applied in either encryption or decryption mode. This creates a sequence of encryption and decryption operations that can differ from one block to another. This sequence can be produced using Pseudo Random Number Generator (PRNG).

Using the same seed for the PRNG at the receiving end allows the same sequence to be generated to decrypt the data properly. Linear Feedback Shift Registers (LFSR) with N registers are used to implement the PRNG. The output of the PRNG is the sequence that will be used. This sequence consists of N bits with each bit representing either encryption or decryption configuration.

The top module consists of two parts: a dynamic and a static part. The static part is made of the DES round and the PRNG. The dynamic part is the key scheduling RP, which is made of two RMs (encryption and decryption). This is shown in fig. 3.

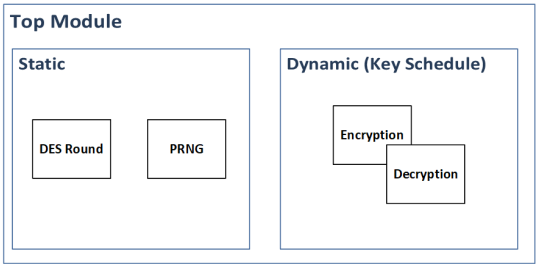

Fig. 3. NDES encryption design blocks

A minimum of two different keys need to be used: one for the encryption mode and the other for the decryption mode. If only one key is used, each consecutive encryption and decryption will cancel each other out. NDES applies an extra layer of security since each of the N iterations acts as if it adds a bit to the effective key. Assuming nk keys are used, the effective key length for NDES will be (56*nk)+N bits provided that N is greater than 2 to protect the design from the meet in the middle attack. For the rest of the paper, it is assumed that NDES is implemented using 2 different keys. NDES provides an extra layer of security since the attacker needs to figure out the sequence in which the N times DES has been applied, otherwise every possible sequence needs to be tried until the attacker finds the right one. Since the sequence changes with every block, the interception of a single sequence by the attacker would not affect the integrity of the rest of the data.

N should be chosen depending on the required throughput of the device. Although the proposed design takes longer encryption time than conventional designs, it uses less power which is more important than the time delay for IoT applications.

The following equation describes the relation between available time for encrypting a 64 bit data block (t) and the number of DES implementations (N). This equation assumes that the

device alternates between encryption and decryption for every DES implementation, and this is the worst case scenario.

$$N = \frac{t}{17 * 10^{-7} + 2 * 10^{-3}}$$

V. Results and Discussion

A. Resource Utilization

Table I shows the resource utilization of the power adaptive encryption design for each configuration. It includes the four main configurations: AES-256, AES-192, AES-128 and DES. It also contains configurations with the key scheduling module set as blank. The resource utilization decreases by 77% and 7.5% when using DES and AES-128 respectively as compared to AES-256.

TABLE I
RESOURCE UTILIZATION FOR POWER ADAPTIVE ENCRYPTION DESIGN

Algorithm	Slice LUTs	Slice registers	F7 muxes	F8 muxes	Block RAM tiles
AES-256	1216	607	115	46	3
AES-192	1170	542	6	3	3
AES-128	1126	476	6	3	3
DES	276	372	0	0	2
AES (blank key scheduling)	962	267	134	46	3
DES (blank key scheduling)	142	197	0	0	2

As for the NDES encryption design, 513 Slice LUTs and 584 Slice registers were used as compared to 1178 Slice LUTs and 1197 Slice registers when conventional 3DES was implemented. This means that using DPR allowed for a 56.5% decrease in Slice LUTs and a 51.2% decrease in Slice registers compared to the conventional design.

B. Energy Utilization

The energy utilization is calculated for each configuration of the power adaptive encryption design and is shown in table II . During the first run, the key logic RP is not blanked and consumes power as well as time to perform the key schedule algorithm. However, during subsequent runs the sub-key values are already calculated and ready and thus the complete system takes less clock cycles and less energy to encrypt the same block of data. Energy per bit is the consumed energy for encrypting a single bit while energy per block is the consumed energy to encrypt a complete block (block size is 128 bits for AES and 64 bits for DES).

TABLE II
ENERGY UTILIZATION FOR POWER ADAPTIVE ENCRYPTION DESIGN

	Energy per bit first run (nJ/bit)	Energy per bit subsequent runs (nJ/bit)	Energy per block (nJ/block)
AES-256	0.3374	0.052	6.656
AES-192	0.2816	0.0467	5.978
AES-128	0.2252	0.0395	5.056
DES	0.0748	0.0314	2.01

The NDES encryption design consumes 2.25uJ/Byte when 3DES configuration is used while [10] used 6.04uJ/Byte for the 3DES algorithm, leading to a 63% reduction in energy needed to encrypt one Byte of data when compared to [10]. The NDES encryption design can also increase the security without any increase in either the power consumption or the resource utilization.

C. Reconfiguration Time

The reconfiguration time of RPs depends on the bit size assigned to the RP. The reconfiguration times for the power adaptive encryption design are 0.0943, 0.1092, 0.6726 ms for the key logic, key memory and round RPs respectively. The NDES encryption design has one RP which is the key schedule and it needed 2 ms to reconfigure.

VI. Conclusion

This paper introduced two encryption solutions for IoT devices. The power adaptive encryption design is able to adapt to the available power budget by selecting one of four encryption algorithms. This is done using DPR and it allows the data to be secured even under a strict power budget, something that was not applicable using conventional designs. The slice utilization ranged from 1216 LUTs and 607 registers to 142 LUTs and 197 registers depending on the configuration. The NDES encryption design utilizes minimal power and resources by using DPR. It provided an average decrease in resource utilization by 54% and a decrease of 63% in the power consumption in the 3DES configuration.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Springer Journal on Computer Networks*, vol. 54, pp. 2787–2805, Oct. 2010.

[2] L. D. Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, pp. 2233 – 2243, Jan. 2014.

[3] National Intelligence Council, *Disruptive Civil Technologies: Six Technologies with Potential Impacts on US Interests out to 2025*, Nov. 2011.

[4] A. Kumar and N. Tiwari, "Aes security enhancement by using double s-box," *International Journal of Computer Science and Information Technologies*, vol. 3, pp. 3980 – 3984, 2012.

[5] G. Singh and Supriya, "A study of encryption algorithms (rsa, des, 3des, and aes) for information security," *International Journal of Computer Applications*, vol. 67, Apr. 2013.

[6] M. A. Bahnasawi, K. Ibrahim, A. Mohamed, M. K. Mohamed, A. Moustafa, K. Abdelmonem, Y. Ismail, and H. Mostafa, "Asic-oriented comparative review of hardware security algorithms for internet of things applications," *IEEE International Conference on Microelectronics*, pp. 285 – 288, Dec. 2016.

[7] Xilinx Inc., "Vivado design suite partial reconfiguration user guide ug909," Apr. 2017.

[8] A. Kamaleldin, A. Mohamed, A. Nagy, Y. Gamal, A. Shalash, and H. Mostafa, "Design guidelines for the high-speed dynamic partial reconfiguration based software defined radio implementations on xilinx zynq fpga," *IEEE International Symposium on Circuits and Systems*, May 2017.

[9] K. Vipin and S. A. Fahmy, "Zycap: Efficient partial reconfiguration management on the xilinx zynq," *IEEE Embedded Systems Letters*, vol. 6, pp. 41 – 44, Mar. 2014.

[10] N. Potlapally, S. Ravi, A. Raghunathan, and N. Jha, "A study of the energy consumption characteristics of cryptographic algorithms and security protocols," *IEEE Transactions on Mobile Computing*, vol. 5, pp. 128 – 143, Feb. 2006.