Revisiting Lightweight Block Ciphers: Review, Taxonomy and Future directions

Aaqib Bashir Dar[a,*], Mashhood Jeelani Lone[a], Nuzhat Hussain[b]

[a] *Independent Researcher, Jammu and Kashmir, India, 190015*
[b] *Department of Computer Science and Engineering, University of Kashmir, Hazratbal, Srinagar, J&K, India,190006*

Abstract

Block ciphers have been extremely predominant in the area of cryptography and due to the paradigm shift towards devices of resource constrained nature, lightweight block ciphers have totally influenced the field and has been a go-to option ever since. The growth of resource constrained devices have put forth a dire need for the security solutions that are feasible in terms of resources without taking a toll on the security that they offer. As the world is starting to move towards Internet of Things (IoT), data security and privacy in this environment is a major concern. This is due to the reason that a huge number of devices that operate in this environment are resource constrained. Because of their resource-constrained nature, advanced mainstream cryptographic ciphers and techniques do not perform as efficiently on such devices. This has led to the boom in the field of 'lightweight cryptography' which aims at developing cryptographic techniques that perform efficiently in a resource constrained environment. Over the period of past two decades or so, a bulk of lightweight block ciphers have been proposed due to the growing need and demand in lightweight cryptography. In this paper, we review the state-of-the-art lightweight block ciphers, present a comprehensive design niche, give a detailed taxonomy with multiple classifications and present future research directions.

Keywords: Lightweight Block Ciphers, Resource-Intense, Attacks, Taxonomy

1. Introduction

Ubiquitous Computing has been involved in drawing a new landscape that is free from any association of a computing device like a computer with a single device or a network of multiple devices but rather a clear picture of a physical world through a huge variety of sensors that are controlled via a rich profusion of actuators. The computing devices are believed to be omnipresent, invisibly networked and interwoven into the fabric of everyday life. It has become extremely important to be cautious about how

*Aaqib Bashir Dar

Email addresses: aaqibb13@gmail.com (Aaqib Bashir Dar), mashhood.jeelani@gmail.com (Mashhood Jeelani Lone), nuzhatbhat503@gmail.com (Nuzhat Hussain)

the data transfer takes place and account for the security of inter-connected devices without compromising the feasibility and usability and further coop with the dynamic nature of environments and ever changing resource needs. For many reasons as such, cryptography becomes an inherently crucial piece to be able to fix an important issue in this puzzle. For very obvious reasons, among the primitives present in the field of cryptography that can fit the needs of the ubiquitous computing are block ciphers, more specifically those that are lightweight in nature. Although the term lightweightness needs some careful evaluation, we shed some light on it in Section 2

Block Ciphers are widely being used as the core constituent of secure systems and many cryptographic primitives with the intention of assuring secure exchange of data through various systems while accounting for factors like high performance or throughput, high security and less energy consumption. However, its important to be sure about the security of a specific cipher, which is either assessed by its resistance to the number of attacks it can withstand which is in turn dependent on the number of attacks that has been performed on the cipher. More number of unsuccessful attack attempts on a cipher indicates its strength to thwart possibility of a successful attack. However, the possibility of existence of currently unknown attacks is always open with the advancements and rapid growth of attack strategies. Therefore, it is equally important to have a security proof of generic nature. The security model in which the security proof is presented is completely out of the context of this paper and will therefore not be discussed any further.

In cryptography, a block cipher is defined as a deterministic algorithm operating on a plaintext block along with a symmetric key and encrypts it to produce the corresponding cipher text. The plaintext block is a fixed-length group of bits and one block is serviced at a time unlike stream ciphers where encryption takes place bitwise. Two paired algorithms constitute a block cipher, an encryption and a decryption algorithm. Both of these constituent algorithms use two parameters as their inputs which are a n-bit block size and a k-bit key size, producing an output block of n bits. To decrypt the ciphertext, the decryption algorithm incorporates the inverse function of encryption. In order to provide secrecy and authentication, this deterministic algorithm must adhere to two principles of confusion and diffusion due to Shannon [1]. The confusion property ensures that the relationship among the ciphertext and the key is as synthesized as practicable. The property of diffusion states that several digits of the ciphertext should be influenced by each digit of the plaintext and several digits of the ciphertext should be influenced by each digit of the key, hiding the relationship between the ciphertext and plaintext. The objective of diffusion is to ensure no leak of information through statistical properties of the plaintext.

For security, the block cipher iterates over a predetermined number of times or rounds, with each additional round acting as a reinforcement. More rounds result in more security. That is the reason why most block cipher algorithms are classified as iterated block ciphers meaning the encryption is carried out by applying a round function which is an invertible transformation, each iteration of which is referred to as round. A key-scheduling algorithm is used to generate one sub key in each round which is derived from the main secret key. This means different keys are used in each round making the deduction process of the main secret key difficult.

While designing a block cipher, these design principles play a crucial role in the

security and performance of the block cipher. A larger block size would result in more security against code-book attacks but it slows down the cipher at the same time. Security can also be increased by using a larger key size, but it also results in exhaustive searching and makes the cipher slow. As already discussed, greater number of rounds do increase security, but more number of rounds also means more execution time. Similarly, using a simple and fast round function means more number of rounds would be required in order to achieve adequate security. However, with complex round functions, fewer rounds are generally required to achieve the security goal. Balancing the trade offs between security and performance while designing block ciphers plays a crucial role as to how good the cipher will perform. The design choices are also largely influenced by the computational power or the target hardware onto which the cipher would have to perform on. While a cipher may perform very good in hardware does not necessarily mean it would perform as efficiently in software. Similarly, a block cipher can perform very efficiently in dedicated hardware like ASIC, but may not be as good on an 8-bit microcontroller. Lightweight ciphers were employed due to their need in low-cost or resource-intense/resource-constrained environments or platforms. The platforms that the ciphers are usually employed on are ASICs, FPGA, SoCs and Microcontrollers. However, the usual applications of resource-intense nature that come into limelight and are specifically targeted/considered in most studies are Radio Frequency Identification (RFIDs) and Wireless Sensor Networks (WSNs). The design trends in block ciphers are discussed in a bit more detail in Section 3

Our Contribution:
In this paper, the knowledge and understanding in the field of lightweight block ciphers is systematized with an aim to provide a clear understanding of the block cipher design principles and a classification for a taxonomic view. Our main contribution lies in studying a total of 70 block ciphers (most, up until this point), present a clear, concise and comprehensive design niche, survey the state-of-the-art lightweight block ciphers, present a taxonomy of lightweight block ciphers, provide a brief discussion and conclude with some key insights.

We provide a thorough introduction in Section 1 followed by the comparison of our work with prior surveys and related works in this area. Section 2 provides a comprehensive overview of lightweight block ciphers and is further supported by related constructions underneath the design of block ciphers. Section 3 identifies the design trends in lightweight block ciphers and present value to important parameters involved in the design of lightweight block ciphers.

Furthermore, in Section 4 we present a taxonomy of lightweight block ciphers based on several key parameters and the underlying constructions. In Section 5, we provide a brief discussion. Section 6 gives definitive conclusions and future research perspectives in this area.

1.1. Related Work

Over the years, there have been several extensive works on several themes either directly targeting block ciphers, their hardware and/or software implementations along with other aspects where block ciphers have been an inherent part of the study.

Nayancy et al., [2] surveyed and analyzed implementations of lightweight block ciphers for resource constraint devices on the basis of block-length, key-length, number of iterated rounds, area in terms of Gate Equivalents (GEs), speed, and logic processes as metrics for comparison between the ciphers. In addition, the comprehensive survey looked at the attacks, highlighted some of the security related issues in terms of key size and identified the opportunities.

Philip et al., [3] surveyed some lightweight ciphers (both block and stream ciphers) and concluded that most of the lightweight ciphers are specific to applications, thereby limiting its wide applicability.

Hatzivasilis et al., [4] presented a survey of block ciphers along with recent advances in this field. In addition they also identified several further opportunities for future research in this area. They specifically examined lightweight implementations of block ciphers both on the software and hardware front. In essence, they evaluated a total of 52 block ciphers along with 360 implementations strictly based on their cost, performance and security. Furthermore, they classified the ciphers based on their suitability to specific applications, and looked at the various cryptanalytic efforts on these ciphers.

Mohd et al., [5] investigated lightweight block cipher implementations and presented a comprehensive review of state-of-the-art in this area and drew attention towards future directions.

Cazorla et al., [6] conducted a survey and also proposed to study block ciphers (some conventional as well as lightweight ciphers) on a dedicated platform of sensors. They furthermore described the design rationale of the ciphers chosen with a supportive summary of their security performance. In hindsight, they presented some implementation tests that were performed on their platform.

Singh et al., [7] discussed several lightweight block ciphers comprehensively along with their relevant characteristics. They categorized the ciphers based on their round function design. They considered several metrics for comparative analysis in resource constrained environments.

Prior Work	**Ciphers Covered**	**Design Rationale**	**Taxonomy**
Nayancy et al [2]	14	○	●
Philip et al [3]	10	○	◐
Hatzivasilis et al [4]	52	◐	◐
Mohd et al [5]	46	○	●
Cazorla et al [6]	18	◐	◐
Singh et al [7]	26	○	●
Our Work	**70**	●	●

2. Overview of Lightweight Block Ciphers:

The context of the term "**Lightweight**" is quite wide, often debatable and of ambiguous nature as well. Lightweight cryptography has been in the scene for a period of

more than two decades or so, very evidently by the need for algorithms targeting low-cost implementations both in hardware and software. However, over the years it has received far more attention to details than it did during its initial years. With it comes the challenge of making use of proper design choices in an algorithm in such a way, that it would perform efficiently without compromising security using low computational resources. The field of lightweight cryptography has been gaining a lot of attention in recent years due to the paradigm shift towards IoT, where it becomes essential to provide security for devices which use low computational resources. There have been ciphers addressing multiple constraints related to the niche of the design principles but not all of them. It is well understood and supported by evidence why these algorithms are needed to be in place by the likes of NIST and CRYPTREC [8] project to have algorithms to standarize. We try to give an overview of several efforts of researchers to address the term 'lightweight ciphers' through their own prisms.

Cazorla et al., [6] performed a survey and benchmarking of certain lightweight block ciphers and made a distinction between the conventional and lightweight ciphers based on certain parameters like block size, key sizes, underlying operations, number of rounds and key schedule. They focused on emphasizing block ciphers with smaller block sizes, smaller key sizes, simplified key schedules and simpler/elementary underlying operations (XOR, AND, OR) with comparatively larger number of rounds to be the right fit for the category of "**lightweight**" ciphers when compared with that of conventional block ciphers.

Some defined those ciphers as lightweight that are targeted for designs that are low-cost. Eisenbarth et al., [9] considered ciphers targeted for devices with tighter cost constraints (such as WSNs and RFIDs) as part of lightweight cryptography. However, the tighter/low cost is ambiguous due to its clear dependency upon the targeted platform and implementation in perspective (Hardware or Software).

Fan et al., [10] defined a lightweight cipher as a cryptographic primitive that is specifically targeted for a resource intense/low resource devices and should be focusing on addressing challenges that are threefold: The overhead in terms of gate counts and memory footprints should be minimal. The power consumption of the deployed solution should be low due to the low-power devices in perspective. The security solution should be relatively reasonable.

2.1. Underlying Constructions:

The underlying structure of the modern block ciphers is generally based on cryptographic frameworks with the Feistel Network and the Substitution Permutation Network (SPN or SP-Network) being the most commonly used structures. The lesser common structures include the Even-Mansour scheme and the Lai-Massey scheme. These frameworks only require the cipher designer to design a round function and plug it into the structure. This makes the round function the defining factor of the ciphers that make use of these structures. These frameworks themselves do not provide an ideal avalanche in a single round and depend on the round function to achieve it. A decent round function used in these structures does result in a very good avalanche after a few rounds.

1. **ARX based constructions:** In the field of lightweight cryptography, the primary goal of the cipher designers is to come up with secure ciphers that are as resource

efficient as possible. The emergence of ARX-based cryptography has provided these authors with strategies by the help of which they can push the limits of efficient cipher design a bit further. ARX stands for Additions(mod 2^n), Rotations and XOR, and ARX-based algorithms make use of only these three operations. These operations prove to be very cheap in both hardware and software and are relatively fast. Additionally, these operations are immune to timing attacks as they run in constant time. This is the reason why some lightweight block ciphers exclusively make use of ARX operations in their round function. These type ciphers are often known as ARX-based ciphers. SPECK [11], HIGHT [12] and LEA [13] are some of the well known lightweight block ciphers.

2. **Substitution-Permutation Networks:** The Substitution Permutation Network is Shannon's concept for a modern block cipher where the plaintext is passed through a series of sequential permutation and substitution boxes. The substitution box (S-Box) takes in a small block of input bits XORed with the round key and substitutes it to produce output bits. The mapping of the substitution process must be one-to-one which ensures invertibility for decryption. The output generated by the S-Boxes of one round is then given to the permutation box (P-Box) where the permutation of bits takes place and the permuted bits are then given as inputs to the S-boxes of the next round. A round key is used for mixing in every round which is derived from the secret key using simple operations. It should be noted that the permutation process is omitted in the last round of the scheme. The non-linear substitution stage ensures Shannon's diffusion property as in it, the mixing of key bits with plaintext bits takes place. The linear permutation stage ensures Shannon's diffusion by driving away the redundancies. The decryption process in an SPN is simply the encryption process but in a reverse manner where the inverse operations of S-Boxes and P-Boxes are carried out and the round keys are fed in reverse order.
3. **Feistel Network:** Introduced by Horst Feistel and adopted by DES [14], the Feistel Network first splits the plaintext into two or more blocks (generally equal). Then in each round, the round function is applied to one of halves of the plaintext using the round sub-key, output of which is XORed with the other half. No operation is performed on the other half. Output of the XOR operation and the other half are then swapped and given as inputs to the next round where the same procedure is followed. However, swapping halves after every round ensures that round function is applied to each half alternatively. This process continues till the last round, after which the two halves are swapped. The decryption process in the Feistel scheme follows the same structure as in encryption but the round keys are given in the reverse order. The main advantage of the Feistel scheme is that the round function need not be invertible as opposed to the Substitution Permutation Network. This provides the cipher designer extra flexibility as any operation that the designer may come up with can be used in the round function of this scheme. However, as in Feistel schemes, diffusion is applied on only half of the data block which leads to smaller round function. So in order to apply the diffusion to the un-transformed state, additional logic is required such as XOR operations which consume almost 2.5 - 3 GE per bit. So it can be stated that Feis-

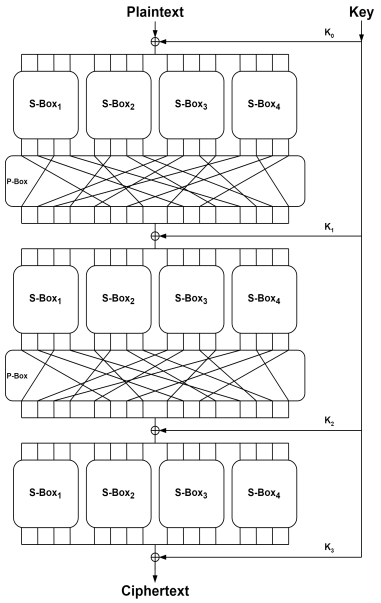

Figure 1: Description of Substitution-Permutation Network

tel Networks have an extra overhead when compared to Substitution Permutation Networks and this becomes the reason why serialized SPN ciphers are expected to achieve data-paths that are smaller [4]. Modern block ciphers quite often use one of the generalized variants of the Feistel scheme. The generalized variant is known as Generalized Feistel Network (GFN) or Generalized Feistel Structure (GFS). The most commonly used among generalized schemes of the Feistel Network are Type-I GFS, Type-II GFS and Type-III GFS. One round of each of the mentioned generalized Feistel schemes can be seen in the Fig 3. Some of the popular lightweight block ciphers that are based on the Feistel Network are μ^2 [15], CHAM [16], ITUbee [17] etc.

4. **Even-Mansour:** Named after the authors Shimon Even and Yishay Mansour who proposed it in 1991, the Even-Mansour scheme [18] used in block ciphers makes use of only one pseudorandom permutation and operates on a key which is composed of two blocks. Before the permutation is applied, one of the blocks of the key is XORed with the message block. The output produced is then XORed with the other half of the key block to produce the cryptogram block. In order to decrypt the cryptogram block, the steps performed in the encryption performed in a reverse order with the inverse of the permutation which is used in encryption. In this scheme, modification of the permutation by the key is simple. Attacks on the scheme have a negligible probability to succeed due to the pseudorandom selection of the permutation. μ^2 [15], LED [19] are examples of lightweight block ciphers that make use of this scheme.

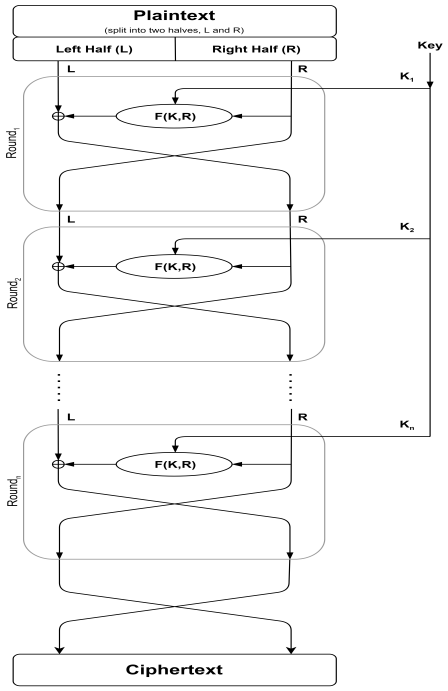

Figure 2: Description of Feistel Network

5. **Lai-Massey:** The Lai-Massey scheme is not as explored as the Feistel Network or the Substitution Permutation Network. The structure is defined in terms of finite abelian groups and behaves just like the Feistel Network in terms of security [20]. The structure was proposed in a thesis by Xuejia Lai. In the thesis, the structure was used in a cipher which later became the International Data Encryption Algorithm (IDEA) [21]. The scheme structure similar to the Feistel scheme splits the plaintext block and then performs operations on each half. Fig 5 shows the contrast between the two schemes.

2.2. Important construction parameters:

Key-Whitening and Tweaking:

Introduced by Ron Rivest in DES-X [22], Key-Whitening is a process in which additional material derived from the key is mixed with the plaintext before the first round or the additional material is mixed with the ciphertext after the last round or both of the operations are carried out. [22]. When this process is carried out using additional key bits, it results in an increase in resistance against brute force attacks due to the key size becoming larger. In contrast, when done by deriving whitening material from the primary key during key scheduling, the key size remains the same and offers no increase in resistance to brute force attacks. While being usually cheaper than round addition, whitening does increase resistance against other attacks. [22]

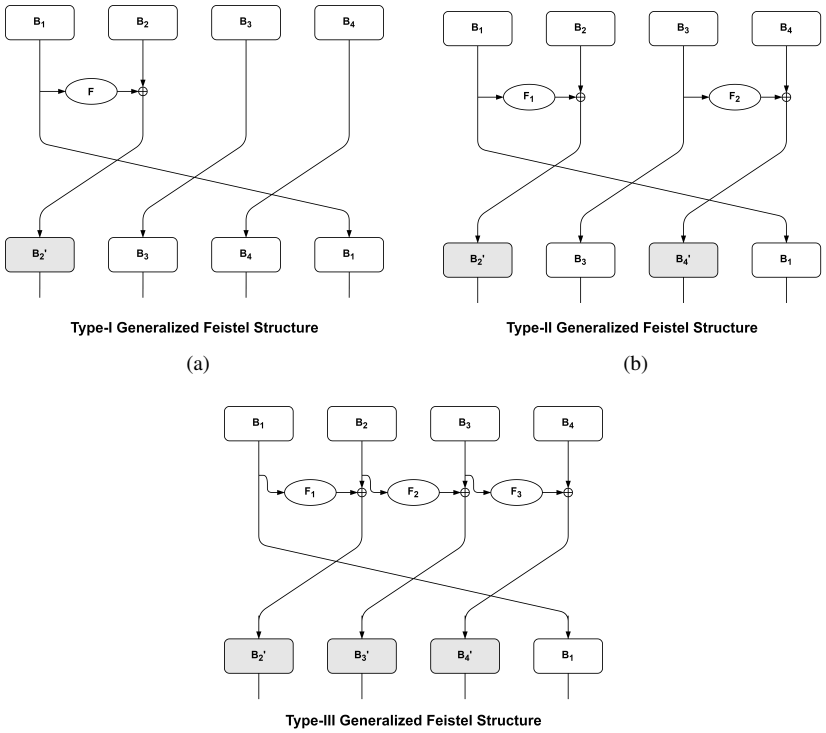

Figure 3: Types of Generalized Feistel Networks

Some of the newer block ciphers have an additional input alongside the plaintext and the key known as tweak which controls the operation of the cipher. Such ciphers are known as tweakable block ciphers. Whitening may be considered as a form of tweaking but several other ways of tweaking do exist as well. With tweaking, new modes of operations become possible if changing tweaks is lighter as compared to the key scheduling operation. The tweak is not required to be secret but rather random and should vary from block to block in some applications.

3. Design Trends in Lightweight Block Ciphers:

It is important to understand the various trends in the design of lightweight block ciphers. There are several involvements both in terms of hardware and software and a variety of metrics as well. The goal of all the primitives is to achieve a trade-off between cost, performance and security where the best that can be achieved is two-fold: a secure and fast chip will have a high cost, similarly a secure and cheap chip will be slow.

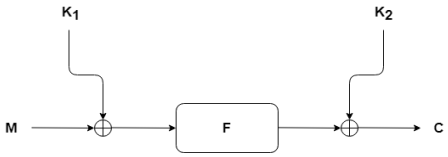

Figure 4: Description of Even-Mansour

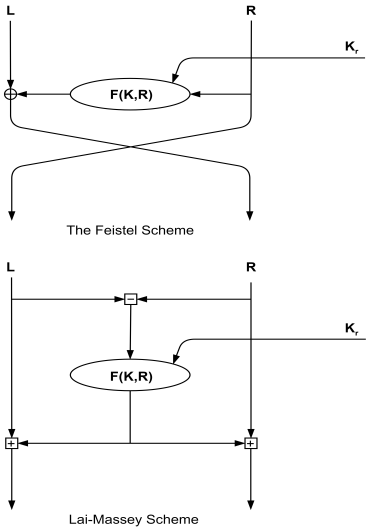

Figure 5: Description of the Lai-Massey Structure relative to the Feistel Structure

3.1. Linear Operations:

A secure cipher should be following two principles so as to foil any cryptanalytic attempts at it, according to Shannon [1]. It should be noted that the linear operations in a cipher are to provide diffusion in a cryptographic primitive indicating that the single bit change in the plaintext is well-off reflected in the ciphertext as well. Since a block cipher typically contains a number of rounds, it allows the cipher state to be dependent on each other. We look at several linear characteristics to be able to depict a clear illustration of the trends in the design choices.

MDS Matrices, Permutations, Rotations and XOR Operation: The design of block ciphers have seen the use of MDS matrices and these matrices are specifically used in order to ensure resistance against linear [23] and differential cryptanalysis [24]. An example of this is seen in AES [25]. AES was followed by several other block ciphers like CLEFIA [26], PHOTON [27] etc. A detailed survey of MDS matrices that are significant in cryptography can be found in [28].

Permutations are simple bit level or word level manipulations. The prominent block ciphers that made use of bit permutations are DES and PRESENT [29]. The permutations however are inefficient in software due to their inconsistent nature. The bit per-

mutations are used in several lightweight block ciphers like Eight-sided fortress [30], OLBCA [31], RECTANGLE [32], SIMON AND SPECK [11] etc.

Rotations are simple bit permutations that are extremely cheap in hardware. The usual rotations that are used are word-wise and are then followed by word-wise XOR operation in ARX based designs specifically. The relevant block ciphers are HIGHT [12], mCRYPTON [33], OLBCA [31] and others.

3.2. Non-Linear Operations:

Non-linear operations are equally as important as linear operations and are attained by the use of S-boxes or arithmetic operations. S-boxes are either based on bit-slicing or Look-Up-tables (LUTs). The primitives using arithmetic operations are usually the ones based on ARX architectural foundation therefore using modular addition and XOR operation.

ARX and Bit-sliced Algorithms, S-Boxes: The representative primitives belonging to the class of ARX based designs are XTEA [34], HIGHT [12], Speck [11] etc. Modular addition along with rotations and XOR operation provide non-linearity and diffusion respectively. The modular addition operation is inexpensive in software due to its simplified nature without any further computational overheads.

Bit-slicing is a technique that allows faster and constant time implementations of secure primitives with immunity to several attacks related to cache and timing with just single bit logical operations. Bit-slicing term was coined by Mathew Kwan [35], although the same technique was used by Biham earlier [36]. In the hardware scenario, bit-slicing allows programming the Look-Up-tables (LUTs) rather than the usual implementation of boolean gates. When bit-slicing is employed within the S-boxes, it becomes an ideal design choice. Some primitives making use of this are RECTANGLE [32] and GIFT [37].

S-Boxes are inherently a part of several cryptographic primitives and are implemented in software using Look-Up-tables (LUTs). They offer nearly optimal properties to cryptographic primitives, however they require storing of all values (outputs), for example in case of AES [25, 38], an 8-bit S-Box has a significant cost in implementations. It is also important to note that looking up LUT values leak the most information [39]. However, LUTs allow efficient hardware implementations. S-Boxes (4-bit) are used by PRINCE [40], PRESENT [29] or Piccolo [41].

Countermeasures on Side-Channel Analysis: It is the non-linear layer that is responsible and accountable for relieving the implementation, thus allowing countermeasures against side-channel attacks (SCA). The complexity of the countermeasure against SCA and the performance depends on the structure of the non-linear layer. It is important to attain a clear balance between resistance against cryptanalysis versus resistance against side-channel analysis.

The lightweight block ciphers using an ARX structure can possibly use the structure of the modular addition to mask the operation directly. The aim should be to ensure security guarantees against all forms of side-channel analysis with a minimal logic and efficient masking as possible. The bit-sliced S-Boxes allow better side-channel resilient

implementations other than being efficient in performance in the non-protected case. The designers of LS-Designs intended the same in [42]

3.3. Key-Schedule:

The Key-schedule is an important and defining factor of a block cipher because it explains the generation and manipulation of a key. Another importance of a key schedule is directly influenced by the Kerchoff's principle [43] which emphasizes in strictly focusing on the key in its entirety. The key-schedules vary based on the underlying construction of a block cipher. The underlying constructions are discussed in a detailed manner in Section 2.1. The key-schedules in lightweight primitives are much simpler than the conventional block ciphers which have complex key-schedules most due to the resource-intense nature of lightweight primitives and the associated costs incurred through the use of complex key-schedules and the storing of subkeys.

Related-Key Attacks: While there are some lightweight block ciphers that explicitly provide resilience against related-key attacks right where the niche of the design strategy is discussed. Some of the lightweight block ciphers being vulnerable to these related key attacks are frivolous. Examples of lightweight block ciphers that have simple related key distinguishers are PRINCE [40].

Security against related-key attacks is an advantage in any case, however it can be attained at the cost of complex key schedules or more number of rounds which in turn is disastrous to the performance specifically in the lightweight scenario.

4. Taxonomy of Lightweight Block Ciphers

There has not been a clear and well-put out taxonomy of lightweight block ciphers that strictly classifies them into suitable categories due to unclear involvement of ciphers that does not fall into the space of lightweight category of block ciphers. However, our effort is the first of its kind to present several classifications of lightweight block ciphers strictly adhering to the lightweight norms. For our survey, we have considered block ciphers that explicitly mentioned the word lightweight in them or explicitly targeted resource constrained devices/environments. There are several reasons why we opted to narrow down this area, the biggest of which is an effort to settle-down the debate of the word "lightweight". Since, it is burdensome to account for all the considerations, we classify the lightweight block ciphers based on their parametric evaluation, and furthermore based on the trends in underlying constructions of design rationale/design niche over the years. In Table 1, we present a parametric evaluation of the studied lightweight ciphers. Table 2 classifies the lightweight block ciphers based on their underlying constructions.

Table 1: Lightweight block cipher parametric evaluation based on underlying principles

Reference	CIPHER NAME	BLOCK SIZE (in bits)	KEY SIZE (in bits)	NUMBER OF ROUNDS
[44]	ANU	64	80, 128	25
[45]	ANU-II	64	80, 128	25
[46]	BORON	64	80, 128	25
[16]	CHAM	64, 128, 128	128, 128, 256	80, 80, 96
[26]	CLEFIA	128	128, 192, 256	18, 22, 26
[47]	CRAFT	64	128	32
[48]	DESL	64	54	16
[48]	DESXL	64	184	16
[49]	DoT	64	80, 128	31
[30]	Eight-Sided Fortress	64	80	32
[50]	EPCBC	48, 96	96	32
[51]	FeW	64	80, 128	32
[52]	FLY	64	128	20
[37]	GIFT	64, 128	128	28, 40
[53]	GRANULE	64	80, 128	32
[54]	Halka	64	80	24
[55]	HERMES	64	128	30
[12]	HIGHT	64	128	32
[56]	HISEC	64	80	15
[57]	I-PRESENT	64	80, 128	30
[17]	ITUbee	80	80	20
[58]	KATAN	32, 48, 64	80	254
[59]	Khudra	64	80	18
[60]	KLEIN	64	64, 80, 96	12, 16 , 20
[58]	KTANTAN	32, 48, 64	80	254

[61]	LBlock	64	80	32
[13]	LEA	128	128. 192, 256	24. 28, 32
[19]	LED	64	64, 96, 128	32, 48, 48
[62]	LiARX	64	128	
[63]	LiCi	64	128	31
[64]	LILLIPUT	64	80	30
[65]	Lilliput-AE	128	128, 192, 256	32, 36, 42
[66]	Loong	64	64, 80, 128	16, 20, 32
[67]	LRBC	16	16	24
[68]	MAES	128	128, 192, 256	10, 12, 14
[69]	MANTIS	64	128	10, 14
[70]	MANTRA	64	80, 128	32
[71]	MARVIN	256	256	16
[33]	mCrypton	64	64, 96, 128	12
[72]	MIBS	64	64, 80	32
[73]	Midori	64, 128	128	16, 20
[74]	NOEKEON	128	128	16
[75]	NVLC	64	80, 128	20
[31]	OLBCA	64	80	22
[41]	Piccolo	64	80, 128	25, 31
[76]	PICO	64	128	32
[29]	PRESENT	64	80	31
[77]	PRIDE	64	128	20
[40]	PRINCE	64	128	12
[78]	PRINTcipher	48, 06	80, 160	48, 96
[79]	PriPresent	64, 80	80, 128	31
[80]	PUFFIN	64	128	32

[81]	QARMA	64, 128	128, 256	3
[82]	QTL	64	64, 128	16, 20
[32]	RECTANGLE	64	80, 128	25
[83]	RoadRunneR	64	80, 128	10, 12
[84]	SAT_Jo	64	80	31
[85]	SFN	64	96 (64-bit round key + 32-bit control key)	32
[86]	Simeck	32, 48, 64	64, 96, 128	32, 36, 44
[11]	SIMON	32, 48, 64, 96, 128	(64), (72/96), (96/128), (96/144), (128/192/256)	(32), (36/36), (42/44), (52/54), (68/69/72)
[87]	SIT	64	64	5
[69]	SKINNY			
	SKINNY-64	n = 64	n / 2n / 3n	32 for n, 36 for 2n, 40 for 3n
	SKINNY-128	n = 128	n / 2n / 3n	40 for n, 48 for 2n, 56 for 3n
[88]	SLIM	32	80	32
[11]	SPECK	32, 48, 64, 96, 128	(64), (72/96), (96/128), (96/144), (128/192/256)	(22), (22/23), (26/27), (28/29), (32/33/34)
[89]	TWINE	64	80, 128	36
[90]	TWIS	128	128	10
[91]	VH	64	64, 80, 96, 112, 128	10, 11, 12, 13, 14
[34]	XTEA	64	128	64
[92]	Zorro	128	128	24
[15]	μ^2	64	80	15

Table 2: Lightweight block cipher taxonomy based on underlying principles

Reference	CIPHER NAME	UNDERLYING CONSTRUCTION
[44]	ANU	Feistel Structure
[45]	ANU-II	Feistel Structure
[46]	BORON	SPN
[16]	CHAM	Type-III GFS
[26]	CLEFIA	Type-II GFS
[47]	CRAFT	SPN
[48]	DESL	Feistel Structure
[48]	DESXL	Feistel Structure
[49]	DoT	SPN
[30]	Eight-Sided Fortress	2-Branch Feistel Structure
[50]	EPCBC	Feistel Structure
[51]	FeW	Feistel-M
[52]	FLY	Lai-Massey Scheme
[37]	GIFT	SPN
[53]	GRANULE	Feistel Structure
[54]	Halka	SPN
[55]	HERMES	SPN Inspired
[12]	HIGHT	ARX
[56]	HISEC	Modifeid Feistel Structure
[57]	I-PRESENT	SPN
[17]	ITUbee	Feistel Structure
[58]	KATAN	Non-linear Funcions
[59]	Khudra	Type-II GFS
[60]	KLEIN	SPN
[58]	KTANTAN	Non-linear Funcions
[61]	LBlock	Feistel Structure
[13]	LEA	ARX
[19]	LED	Special case of Even-Mansour
[62]	LiARX	Long Trail Strategy (LTS)
[63]	LiCi	Feistel Structure
[64]	LILLIPUT	Extended GFN
[65]	Lilliput-AE	Extended Generalized Feistel Network
[66]	Loong	SPN
[67]	LRBC	Feistel-SPN Hybrid
[68]	MAES	SPN
[69]	MANTIS	SPN-ARX Hybrid
[70]	MANTRA	Feistel Structure
[71]	MARVIN	LS-Design
[33]	mCrypton	SPN
[72]	MIBS	Feistel Structure
[73]	Midori	SPN

[74]	NOEKEON	SPN
[75]	NVLC	SPN
[31]	OLBCA	-
[41]	Piccolo	Variant of GFS
[76]	PICO	SPN
[29]	PRESENT	SPN
[77]	PRIDE	SPN
[40]	PRINCE	SPN
[78]	PRINTcipher	SPN
[79]	PriPresent	SPN
[80]	PUFFIN	SPN
[81]	QARMA	Even-Mansour
[82]	QTL	Feistel Structure
[32]	RECTANGLE	SPN
[83]	RoadRunneR	Feistel Structure
[84]	SAT_Jo	SPN
[85]	SFN	Feistel-SPN Hybrid
[86]	Simeck	Feistel Structure
[11]	SIMON	Feistel Structure
[87]	SIT	Feistel-SPN Hybrid
[69]	SKINNY SKINNY-64 SKINNY-128	SPN
[88]	SLIM	Feistel Structure
[11]	SPECK	ARX
[89]	TWINE	Type-II GFS
[90]	TWIS	2-branch GFS
[91]	VH	SPN
[34]	XTEA	Feistel Structure
[92]	Zorro	Generalized Even-Mansour
[15]	μ^2	Type-II GFS

5. Discussion:

While studying the field of Lightweight Cryptography, one of the main things we found was that there is no clear and established line drawn as to when we call a cipher 'lightweight' or 'ultra-lightweight' in terms of metrics and performance. As per our knowledge, there is not a single agreed upon universal definition of when a cipher should be called lightweight, the reason for that is the involvement of several parameters. However, with growing demand and ciphers pouring in, it is equally important to settle this question.

While AES [25] may be called a lightweight cipher when compared to other mainstream block ciphers such as Blowfish [93] or Twofish [94], the case would not be

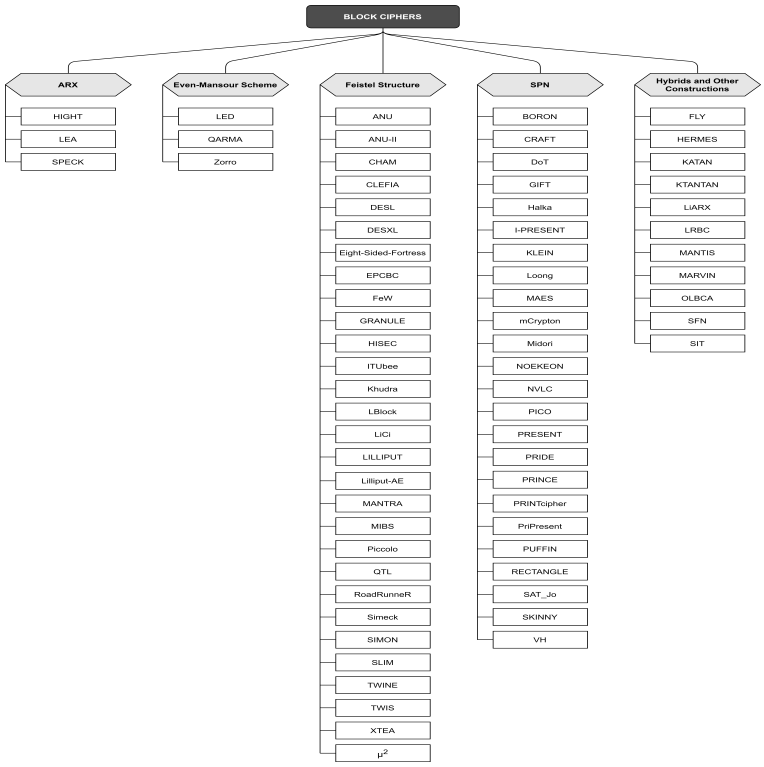

Figure 6: A Taxonomy of lightweight block ciphers based upon their underlying structure

the same when AES is compared to the mainstream lightweight block ciphers like PRESENT or SIMON. This is because AES was not proposed to perform in resource constrained environments. We feel that the tags 'lightweight' or 'ultra-lightweight' need some form of standardization in the field of lightweight cryptography so as to eradicate the confusion that if the term is being used relatively or if the proposed cryptographic technique or cipher is meant for resource constrained platforms. This is the reason why we in our study, we chose only those ciphers that were proposed to perform in resource constrained environments. Another factor we think is important is that a chunk of lightweight block ciphers are inspired from the block cipher PRESENT and a large number of them have chosen similar parameters as that of PRESENT.

Another important aspect that we want to shed some light on is the security reduction of cryptographic primitives which has become a de facto standard property of any proposed modern primitive. It is observed throughout, that the security of a lightweight block is mainly observed through the computational complexity of various number of attacks. However, cryptography is having a paradigm shift towards provably secure primitives, we believe it is equally important to have a generic (or common) security model for lightweight primitives that can be used to prove the security of lightweight block ciphers as well.

6. Conclusions and Future directions:

Lightweight block ciphers have been in tremendous demand due to the immense rise of IoT applications and need for secure primitives in applications that have limited resources like that of wireless sensor networks, RFID etc. We look at the state-of-the-art lightweight block ciphers that have been proposed over the years both in academia and by government authorities. We discuss the trends in their design strategies, classify the lightweight block ciphers on several parameters and provide a taxonomy of the parametric evaluation and discuss the findings. There are several open research directions in the area of lightweight cryptography in general. A security model will be a bliss when it comes to primitives in the field of cryptography. Since the notions of security proofs in secure models like that of CCA, CPA and their variants are already present to propose primitives that are provably secure. It is seen that the security of block ciphers has been considered through the computational complexity of attacks that are performed on a cipher, however this measure is not a concrete one due to the advent of various attack strategies with the steep technological growth and expansion of extreme processing power. There is serious need to pay attention to this area of research for finding further prospects. Also, with the prediction of a fully scalable quantum within a period of this decade, it is important to have security primitives that are quantum safe. For this area, a separate field called 'Post Quantum Cryptography' is established for the sake of building primitives that are safe from quantum attacks. With IoT, how we interact with devices is going to change immensely. Considering the fact that lightweight cryptographic techniques are the backbone of security in this environment, the supply of new ideas and techniques targeting post-quantum security would go up by quite a margin to meet the growing demand. The focus in designing new lightweight cryptographic primitives should now be to ensure post-quantum security while enabling scalability in resource constrained devices like IoT.

7. Declaration of Competing Interest:

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] C. E. Shannon, Communication theory of secrecy systems, The Bell system technical journal 28 (4) (1949) 656–715.

[2] Nayancy, S. Dutta, S. Chakraborty, A survey on implementation of lightweight block ciphers for resource constraints devices, Journal of Discrete Mathematical Sciences and Cryptography (2020) 1–22.

[3] M. A. Philip, et al., A survey on lightweight ciphers for iot devices, in: 2017 International Conference on Technological Advancements in Power and Energy (TAP Energy), IEEE, 2017, pp. 1–4.

[4] G. Hatzivasilis, K. Fysarakis, I. Papaefstathiou, C. Manifavas, A review of lightweight block ciphers, Journal of cryptographic Engineering 8 (2) (2018) 141–184.

[5] B. J. Mohd, T. Hayajneh, A. V. Vasilakos, A survey on lightweight block ciphers for low-resource devices: Comparative study and open issues, Journal of Network and Computer Applications 58 (2015) 73–93.

[6] M. Cazorla, K. Marquet, M. Minier, Survey and benchmark of lightweight block ciphers for wireless sensor networks, in: 2013 international conference on security and cryptography (SECRYPT), IEEE, 2013, pp. 1–6.

[7] P. Singh, B. Acharya, R. K. Chaurasiya, A comparative survey on lightweight block ciphers for resource constrained applications, International Journal of High Performance Systems Architecture 8 (4) (2019) 250–270.

[8] H. Imai, A. Yamagishi, Cryptrecproject, AdvancesinCryptology-ASIACRYPT2000 LNCS (1976) 399–400.

[9] T. Eisenbarth, S. Kumar, C. Paar, A. Poschmann, L. Uhsadel, A survey of lightweight-cryptography implementations, IEEE Design & Test of Computers 24 (6) (2007) 522–533.

[10] X. Fan, K. Mandal, G. Gong, Wg-8: A lightweight stream cipher for resource-constrained smart devices, in: International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness, Springer, 2013, pp. 617–632.

[11] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, L. Wingers, The simon and speck lightweight block ciphers, in: Proceedings of the 52nd Annual Design Automation Conference, 2015, pp. 1–6.

[12] D. Hong, J. Sung, S. Hong, J. Lim, S. Lee, B.-S. Koo, C. Lee, D. Chang, J. Lee, K. Jeong, et al., Hight: A new block cipher suitable for low-resource device, in: International Workshop on Cryptographic Hardware and Embedded Systems, Springer, 2006, pp. 46–59.

[13] D. Hong, J.-K. Lee, D.-C. Kim, D. Kwon, K. H. Ryu, D.-G. Lee, Lea: A 128-bit block cipher for fast encryption on common processors, in: International Workshop on Information Security Applications, Springer, 2013, pp. 3–27.

[14] D. E. Standard, et al., Data encryption standard, Federal Information Processing Standards Publication (1999) 112.

[15] W.-Z. Yeoh, J. S. Teh, M. I. S. B. M. Sazali, μ 2: A lightweight block cipher, in: Computational Science and Technology, Springer, 2020, pp. 281–290.

[16] B. Koo, D. Roh, H. Kim, Y. Jung, D.-G. Lee, D. Kwon, Cham: a family of lightweight block ciphers for resource-constrained devices, in: International Conference on Information Security and Cryptology, Springer, 2017, pp. 3–25.

[17] F. Karakoç, H. Demirci, A. E. Harmancı, Itubee: a software oriented lightweight block cipher, in: International Workshop on Lightweight Cryptography for Security and Privacy, Springer, 2013, pp. 16–27.

[18] S. Even, Y. Mansour, A construction of a cipher from a single pseudorandom permutation, Journal of cryptology 10 (3) (1997) 151–161.

[19] J. Guo, T. Peyrin, A. Poschmann, M. Robshaw, The led block cipher, in: International workshop on cryptographic hardware and embedded systems, Springer, 2011, pp. 326–341.

[20] A. Yun, J. H. Park, J. Lee, Lai-massey scheme and quasi-feistel networks., IACR Cryptol. ePrint Arch. 2007 (2007) 347.

[21] H. Lipmaa, Idea: A cipher for multimedia architectures?, in: International Workshop on Selected Areas in Cryptography, Springer, 1998, pp. 248–263.

[22] R. Rivest, personal communication (1995, 1996).

[23] A. Tardy-Corfdir, H. Gilbert, A known plaintext attack of feal-4 and feal-6, in: Annual International Cryptology Conference, Springer, 1991, pp. 172–182.

[24] E. Biham, A. Shamir, Differential cryptanalysis of des-like cryptosystems, Journal of CRYPTOLOGY 4 (1) (1991) 3–72.

[25] V. Rijmen, J. Daemen, Advanced encryption standard, Proceedings of Federal Information Processing Standards Publications, National Institute of Standards and Technology (2001) 19–22.

[26] T. Shirai, K. Shibutani, T. Akishita, S. Moriai, T. Iwata, The 128-bit blockcipher clefia, in: International workshop on fast software encryption, Springer, 2007, pp. 181–195.

[27] J. Guo, T. Peyrin, A. Poschmann, The photon family of lightweight hash functions, in: Annual Cryptology Conference, Springer, 2011, pp. 222–239.

[28] K. C. Gupta, S. K. Pandey, I. G. Ray, S. Samanta, Cryptographically significant mds matrices over finite fields: A brief survey and some generalized results, Advances in Mathematics of Communications 13 (4) (2019) 779.

[29] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. Robshaw, Y. Seurin, C. Vikkelsoe, Present: An ultra-lightweight block cipher, in: International workshop on cryptographic hardware and embedded systems, Springer, 2007, pp. 450–466.

[30] L. Xuan, W.-y. ZHANG, X.-z. LIU, L. Feng, Eight-sided fortress: a lightweight block cipher, The Journal of China Universities of Posts and Telecommunications 21 (1) (2014) 104–128.

[31] S. S. M. Aldabbagh, I. F. T. Al Shaikhli, Olbca: A new lightweight block cipher algorithm, in: 2014 3rd International Conference on Advanced Computer Science Applications and Technologies, IEEE, 2014, pp. 15–20.

[32] W. Zhang, Z. Bao, D. Lin, V. Rijmen, B. Yang, I. Verbauwhede, Rectangle: a bit-slice lightweight block cipher suitable for multiple platforms, Science China Information Sciences 58 (12) (2015) 1–15.

[33] C. H. Lim, T. Korkishko, mcrypton–a lightweight block cipher for security of low-cost rfid tags and sensors, in: International workshop on information security applications, Springer, 2005, pp. 243–258.

[34] M. Roger, D. Wheeler, Tea extensions, Tech. rep., Technical Report, Computer Laboratory, University of Cambridge (1997).

[35] M. Kwan, Reducing the gate count of bitslice des., IACR Cryptol. ePrint Arch. 2000 (2000) 51.

[36] E. Biham, A fast new des implementation in software, in: International Workshop on Fast Software Encryption, Springer, 1997, pp. 260–272.

[37] S. Banik, S. K. Pandey, T. Peyrin, Y. Sasaki, S. M. Sim, Y. Todo, Gift: a small present, in: International Conference on Cryptographic Hardware and Embedded Systems, Springer, 2017, pp. 321–345.

[38] J. Daemen, V. Rijmen, The design of Rijndael, Vol. 2, Springer, 2002.

[39] A. Biryukov, D. Dinu, J. Großschädl, Correlation power analysis of lightweight block ciphers: From theory to practice, in: International Conference on Applied Cryptography and Network Security, Springer, 2016, pp. 537–557.

[40] J. Borghoff, A. Canteaut, T. Güneysu, E. B. Kavun, M. Knezevic, L. R. Knudsen, G. Leander, V. Nikov, C. Paar, C. Rechberger, et al., Prince–a low-latency block cipher for pervasive computing applications, in: International conference on the theory and application of cryptology and information security, Springer, 2012, pp. 208–225.

[41] K. Shibutani, T. Isobe, H. Hiwatari, A. Mitsuda, T. Akishita, T. Shirai, Piccolo: an ultra-lightweight blockcipher, in: International workshop on cryptographic hardware and embedded systems, Springer, 2011, pp. 342–357.

[42] V. Grosso, G. Leurent, F.-X. Standaert, K. Varıcı, Ls-designs: Bitslice encryption for efficient masked software implementations, in: International Workshop on Fast Software Encryption, Springer, 2014, pp. 18–37.

[43] N. Ferguson, B. Schneier, Practical cryptography, Vol. 141, Wiley New York, 2003.

[44] G. Bansod, A. Patil, S. Sutar, N. Pisharoty, Anu: an ultra lightweight cipher design for security in iot, Security and Communication Networks 9 (18) (2016) 5238–5251.

[45] V. Dahiphale, G. Bansod, J. Patil, Anu-ii: A fast and efficient lightweight encryption design for security in iot, in: 2017 International Conference on Big Data, IoT and Data Science (BID), IEEE, 2017, pp. 130–137.

[46] G. Bansod, N. Pisharoty, A. Patil, Boron: an ultra-lightweight and low power encryption design for pervasive computing, Frontiers of Information Technology & Electronic Engineering 18 (3) (2017) 317–331.

[47] C. Beierle, G. Leander, A. Moradi, S. Rasoolzadeh, Craft: lightweight tweakable block cipher with efficient protection against dfa attacks, IACR Transactions on Symmetric Cryptology 2019 (1) (2019) 5–45.

[48] G. Leander, C. Paar, A. Poschmann, K. Schramm, New lightweight des variants, in: International Workshop on Fast Software Encryption, Springer, 2007, pp. 196–210.

[49] J. Patil, G. Bansod, K. S. Kant, Dot: A new ultra-lightweight sp network encryption design for resource-constrained environment, in: Proceedings of the 2nd International Conference on Data Engineering and Communication Technology, Springer, 2019, pp. 249–257.

[50] H. Yap, K. Khoo, A. Poschmann, M. Henricksen, Epcbc-a block cipher suitable for electronic product code encryption, in: International Conference on Cryptology and Network Security, Springer, 2011, pp. 76–97.

[51] M. Kumar, P. Sk, A. Panigrahi, Few: a lightweight block cipher, Turkish Journal of Mathematics and Computer Science 11 (2) (2014) 58–73.

[52] P. Karpman, B. Grégoire, The littlun s-box and the fly block cipher, in: Lightweight Cryptography Workshop, 2016, pp. 17–18.

[53] G. Bansod, A. Patil, N. Pisharoty, Granule: An ultra lightweight cipher design for embedded security., IACR Cryptol. ePrint Arch. 2018 (2018) 600.

[54] S. Das, Halka: A lightweight, software friendly block cipher using ultra-lightweight 8-bit s-box., IACR Cryptol. ePrint Arch. 2014 (2014) 110.

[55] S. B. Măluţan, I. R. Dragomir, M. Lazăr, D. Vitan, Hermes, a proposed lightweight block cipher used for limited resource devices, in: 2019 International Conference on Speech Technology and Human-Computer Dialogue (SpeD), IEEE, 2019, pp. 1–6.

[56] S. S. M. AlDabbagh, I. F. T. Al Shaikhli, M. A. Alahmad, Hisec: A new lightweight block cipher algorithm, in: Proceedings of the 7th International Conference on Security of Information and Networks, 2014, pp. 151–156.

[57] M. R. Z'aba, N. Jamil, M. E. Rusli, M. Z. Jamaludin, A. A. M. Yasir, I-present tm: An involutive lightweight block cipher, Journal of Information Security 2014.

[58] C. De Canniere, O. Dunkelman, M. Knežević, Katan and ktantan—a family of small and efficient hardware-oriented block ciphers, in: International Workshop on Cryptographic Hardware and Embedded Systems, Springer, 2009, pp. 272–288.

[59] S. Kolay, D. Mukhopadhyay, Khudra: a new lightweight block cipher for fpgas, in: International Conference on Security, Privacy, and Applied Cryptography Engineering, Springer, 2014, pp. 126–145.

[60] Z. Gong, S. Nikova, Y. W. Law, Klein: a new family of lightweight block ciphers, in: International Workshop on Radio Frequency Identification: Security and Privacy Issues, Springer, 2011, pp. 1–18.

[61] W. Wu, L. Zhang, Lblock: a lightweight block cipher, in: International conference on applied cryptography and network security, Springer, 2011, pp. 327–344.

[62] S. Mishra, D. Sadhya, Liarx: A lightweight cipher based on the lts design strategy of arx, in: International Conference on Information Systems Security, Springer, 2020, pp. 185–197.

[63] J. Patil, G. Bansod, K. S. Kant, Lici: A new ultra-lightweight block cipher, in: 2017 International Conference on Emerging Trends & Innovation in ICT (ICEI), IEEE, 2017, pp. 40–45.

[64] T. P. Berger, J. Francq, M. Minier, G. Thomas, Extended generalized feistel networks using matrix representation to propose a new lightweight block cipher: Lilliput, IEEE Transactions on Computers 65 (7) (2015) 2074–2089.

[65] A. Adomnicai, T. P. Berger, C. Clavier, J. Francq, P. Huynh, V. Lallemand, K. Le Gouguec, M. Minier, L. Reynaud, G. Thomas, Lilliput-ae: a new lightweight tweakable block cipher for authenticated encryption with associated data, Submitted to NIST Lightweight Project.

[66] B.-T. Liu, L. Li, R.-X. Wu, M.-M. Xie, Q. P. Li, Loong: A family of involutional lightweight block cipher based on spn structure, IEEE Access 7 (2019) 136023–136035.

[67] A. Biswas, A. Majumdar, S. Nath, A. Dutta, K. Baishnab, Lrbc: a lightweight block cipher design for resource constrained iot devices, Journal of Ambient Intelligence and Humanized Computing (2020) 1–15.

[68] A. R. Chowdhury, J. Mahmud, A. R. M. Kamal, M. A. Hamid, Maes: modified advanced encryption standard for resource constraint environments, in: 2018 IEEE Sensors Applications Symposium (SAS), IEEE, 2018, pp. 1–6.

[69] C. Beierle, J. Jean, S. Kölbl, G. Leander, A. Moradi, T. Peyrin, Y. Sasaki, P. Sasdrich, S. M. Sim, The skinny family of block ciphers and its low-latency variant mantis, in: Annual International Cryptology Conference, Springer, 2016, pp. 123–153.

[70] G. Bansod, N. Pisharoty, A. Patil, Mantra: an ultra lightweight cipher design for ubiquitous computing, International Journal of Ad Hoc and Ubiquitous Computing 28 (1) (2018) 13–26.

[71] S. Saha, K. Rarhi, A. Bhattacharya, P. Mukherjee, An involutive lightweight block cipher for 256-bit block size, in: 2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT), IEEE, 2018, pp. 251–256.

[72] M. Izadi, B. Sadeghiyan, S. S. Sadeghian, H. A. Khanooki, Mibs: a new lightweight block cipher, in: International Conference on Cryptology and Network Security, Springer, 2009, pp. 334–348.

[73] S. Banik, A. Bogdanov, T. Isobe, K. Shibutani, H. Hiwatari, T. Akishita, F. Regazzoni, Midori: A block cipher for low energy, in: International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2015, pp. 411–436.

[74] J. Daemen, M. Peeters, G. Van Assche, V. Rijmen, Nessie proposal: Noekeon, in: First Open NESSIE Workshop, 2000, pp. 213–230.

[75] S. Abd Al-Rahman, A. Sagheer, O. Dawood, Nvlc: New variant lightweight cryptography algorithm for internet of things, in: 2018 1st Annual International Conference on Information and Sciences (AiCIS), IEEE, 2018, pp. 176–181.

[76] G. Bansod, N. Pisharoty, A. Patil, Pico: An ultra lightweight and low power encryption design for ubiquitous computing., Defence Science Journal 66 (3).

[77] M. R. Albrecht, B. Driessen, E. B. Kavun, G. Leander, C. Paar, T. Yalçın, Block ciphers–focus on the linear layer (feat. pride), in: Annual Cryptology Conference, Springer, 2014, pp. 57–76.

[78] L. Knudsen, G. Leander, A. Poschmann, M. J. Robshaw, Printcipher: a block cipher for ic-printing, in: International Workshop on Cryptographic Hardware and Embedded Systems, Springer, 2010, pp. 16–32.

[79] M. Girija, P. Manickam, M. Ramaswami, Pripresent: an embedded prime lightweight block cipher for smart devices, Peer-to-Peer Networking and Applications (2020) 1–11.

[80] H. Cheng, H. M. Heys, C. Wang, Puffin: A novel compact block cipher targeted to embedded digital systems, in: 2008 11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools, IEEE, 2008, pp. 383–390.

[81] R. Avanzi, The qarma block cipher family. almost mds matrices over rings with zero divisors, nearly symmetric even-mansour constructions with non-involutory central rounds, and search heuristics for low-latency s-boxes, IACR Transactions on Symmetric Cryptology (2017) 4–44.

[82] L. Li, B. Liu, H. Wang, Qtl: a new ultra-lightweight block cipher, Microprocessors and Microsystems 45 (2016) 45–55.

[83] A. Baysal, S. Şahin, Roadrunner: A small and fast bitslice block cipher for low cost 8-bit processors, in: Lightweight Cryptography for Security and Privacy, Springer, 2015, pp. 58–76.

[84] M. J. R. Shantha, L. Arockiam, Sat_jo: An enhanced lightweight block cipher for the internet of things, in: 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), IEEE, 2018, pp. 1146–1150.

[85] L. Li, B. Liu, Y. Zhou, Y. Zou, Sfn: A new lightweight block cipher, Microprocessors and Microsystems 60 (2018) 138–150.

[86] G. Yang, B. Zhu, V. Suder, M. D. Aagaard, G. Gong, The simeck family of lightweight block ciphers, in: International workshop on cryptographic hardware and embedded systems, Springer, 2015, pp. 307–329.

[87] M. Usman, I. Ahmed, M. I. Aslam, S. Khan, U. A. Shah, Sit: a lightweight encryption algorithm for secure internet of things, arXiv preprint arXiv:1704.08688.

[88] B. Aboushosha, R. A. Ramadan, A. D. Dwivedi, A. El-Sayed, M. M. Dessouky, Slim: A lightweight block cipher for internet of health things, IEEE Access 8 (2020) 203747–203757.

[89] T. Suzaki, K. Minematsu, S. Morioka, E. Kobayashi, Twine: A lightweight block cipher for multiple platforms, in: International Conference on Selected Areas in Cryptography, Springer, 2012, pp. 339–354.

[90] S. K. Ojha, N. Kumar, K. Jain, et al., Twis–a lightweight block cipher, in: International Conference on Information Systems Security, Springer, 2009, pp. 280–291.

[91] X. Dai, Y. Huang, L. Chen, T. Lu, F. Su, Vh: a lightweight block cipher based on dual pseudo-random transformation, in: International Conference on Cloud Computing and Security, Springer, 2015, pp. 3–13.

[92] B. Gérard, V. Grosso, M. Naya-Plasencia, F.-X. Standaert, Block ciphers that are easier to mask: How far can we go?, in: International Conference on Cryptographic Hardware and Embedded Systems, Springer, 2013, pp. 383–399.

[93] B. Schneier, Description of a new variable-length key, 64-bit block cipher (blowfish), in: International Workshop on Fast Software Encryption, Springer, 1993, pp. 191–204.

[94] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson, The Twofish encryption algorithm: a 128-bit block cipher, John Wiley & Sons, Inc., 1999.

Table 3: Summarizing Key-findings of Lightweight Block Ciphers

Block Cipher	Important Findings
μ^2 [15]	μ^2 is a 64-bit lightweight block cipher offering a key-size of 80-bits, consisting of 15 rounds and is based on the Generalized Feistel Structure (GFS) with a round function, which is a 4-round and 16-bit ultra-lightweight SPN based block cipher and Even-Mansour construction. For encryption and key generation μ^2 uses 4-bit S-Box. The cipher performs better than PRESENT and the security evaluation shows that it resists well-known attacks
CHAM [16]	CHAM is a block cipher family that has three variants: CHAM64/128 with 80 rounds, CHAM128/128 with 80 rounds and CHAM 128/256 with 96 rounds which were designed using a four-branch Type-III Generalized Feistel Structure based on ARX operations. It has a key schedule that is stateless-on-the-fly, implemented thereby improving the hardware implementations. CHAM defends against simple rotational attacks and slide attacks. It is similar to that of SPECK in 8-bit 16-bit software. In terms of hardware, it can use only 73% area as that of SIMON.
ITUbee [17]	ITUbee is a lightweight block cipher that is software oriented based on Feistel structure that has key whitening layers at top and bottom of the cipher. The block size and key length is 80-bits. The number of rounds it has is 20 with no key-schedule thus allowing low energy consumption but making the cipher prone to related-key differential attacks. The cipher offers 80-bit security in related-key and single-key attack models
KATAN KTANTAN [58] **Family**	KATAN and KTANTAN are hardware oriented lightweight block ciphers which are extremely productive in hardware along with a decent level of security and a minimalistic number of gates. Both the families have three variants: a 32, 48 and 64-bit variant. They use an 80-bit long key with 254 rounds. The two families differ only in their key-schedule.

Khudra [59]	Khudra is a 4-branch, Type-II, Generalized Feistel Structure based block cipher designed for FPGAs operating on 64-bit blocks using 80-bit key. The cipher is based on a different design strategy designed specifically for FPGAs that is presented in the paper itself. The security analysis shows that the cipher is resilient against conventional attacks, MIME attacks and related-key differential attacks. The implementation results show that it outperforms the likes of PRESENT and Piccolo.
KLEIN [60] **Family**	KLEIN is a family of SPN based lightweight block ciphers designed primarily for software implementations. The family has three variants with a fixed block size of 64-bits. They differ in the security they provide with key-lengths of 64, 80 and 96-bits respectively. The 64-bit variant iterates over 12 rounds, 80-bit variant over 16 rounds and 96-bit variant over 20 rounds. The security analysis shows it to be resistant against algebraic attack, integral attack, linear and differential attacks, key-schedule attacks and side-channel attack.
LBlock [61]	LBlock is a lightweight block cipher that has a block size of 64-bits and a key-length of 80-bits. It is based on Feistel network iterating over 32 rounds. A 32-bit round key derived from the master key is used in each round of encryption and decryption procedure. The hardware implementation of this cipher takes up 1350GE on 0.18μm technology and 866.3GE in area optimized implementation with additional RAM. The security analysis shows that it provides security against known attacks.
XTEA [34]	XTEA is an extended version of the lightweight block cipher, Tiny encryption algorithm (TEA) family of block ciphers. It was developed to address to overcome the discovered weakness in the original TEA. XTEA consists of a 64-bit block size and a key-length of 128-bits utilizing a 64 round Feistel network. XTEA employs ARX operations.

LEA [13]	LEA is a 128-bit lightweight block cipher supporting three key-sizes of 128-bits, 192-bits and 256-bits employing 24, 28 and 32 rounds respectively. It makes use of ARX operations allowing fast software encryption, small size code and provides resistance against attacks. The round key function is used to process round keys for encryption. Its hardware implementations shows that it has decent throughput per area. Low overheads results in less power consumption, thereby making it energy efficient.
mCrypton [33]	The lightweight block cipher was designed to be efficient in resource consumption. It has a block-size of 64-bits with a 12 round SPN structure, and offers three key-sizes of 64, 96 and 128-bits. It uses different key-schedules for encryption and decryption. The prototype implementation on 1 cycle/round architecture shows that the cipher requires about 3500 to 4100 gates for encryption and decryption schemes and requires about 2400 to 3000 gates for 0.1μm CMOS technology. The cipher is secure against known attacks.
MIBS [72]	MIBS is a very compact 32 round Feistel cipher that has a 64-bit block size and supports key-lengths of 64 and 80-bits. The key-schedule of 64-bit variant of MIBS is designed following the key-schedule of PRESENT. The cipher provides adequate security margins along with being very efficient in hardware with the 64-bit variant only requiring 1400 gates on 0.18μm technology.
Midori [73]	Midori was designed to keep the energy consumption as low as possible. It was based on SPN and has two variants: Midori64 and Midori128, iterating over 16 and 20 rounds respectively using 128-bit key. The factors that were considered are clock frequency, architecture, loop unrolling etc. Both the variants provide a decent level of security against boomerang attack, differential and linear attacks, impossible differential attacks, meet-in-the-middle attack and others.

PRINCE [40]	PRINCE is a lightweight 64-bit block cipher that uses a 128-bit key and is based on a variant of FX-construction. Its key is made up of two 64-bit elements which is extended to 192-bits. The core cipher $PRINCE_{core}$ is an AES like block cipher and is based on SPN which iterates over 12 rounds. The authors did not claim to provide 128-bit security and resistant to related-key attacks
LED [19]	LED is a lightweight block cipher that operates on 64-bit block size and supports key-sizes ranging from 64-bits (LED-64) to 128-bits (LED-128). It is based on a special case of Even-Mansour scheme. LED-64 iterates over 32 rounds and LED-128 iterates over 48 rounds. LED proves to be secure than AES and very compact in hardware and maintains a competitive performance in hardware as well.
Simeck Family [86]	Simeck family was designed by combining the design features of SIMON and SPECK families of block ciphers. The Simeck family has three variants: Simeck32/64 uses a block size of 32-bits and a 64-bit key, Simeck-48/96 uses a 48-bit block and 96-bit key, Simeck-64/128 uses a 64-bit block and a 128-bit key. The round function of the key schedule is based on Feistel structure and its three variants iterates over 32, 36 and 44 rounds respectively. It uses ARX type operations. Simeck shows better performance than hardware-optimized SIMON in terms of area and power requirements in both CMOS 130nm and 65nm technologies. The security analysis shows that it is resistant to know attacks.
TWINE [89]	Twine is a 64-bit lightweight block cipher based on the Generalized Feistel Structure of Type-II. The 36 round cipher uses key lengths of 80-bits or 128-bits. This is claimed to be the first attempt to combine 4-bit S-box and 4-bit-wise permutations along with XOR. Twine performs well in hardware and software as well due to the implementation of GFS Type-II with highly diffusive block shuffle. Security analysis shows Twine providing good level of security, particularly against differential and saturation attacks.

TWIS [90]	TWIS is a 128-bit block-cipher influenced directly by CLEFIA and is based on 2-branch Generalized Feistel Structure performing key-whitening at the start and end of the algorithm. It uses a key-length of 128-bits and iterates over 10 rounds. TWIS has two main parts: Key-scheduling and data processing part. The cipher shows to be more efficient than CLEFIA while providing the same security level as CLEFIA.
VH [91]	VH (Vertical Horizontal) is a lightweight block cipher relying on a transformation called dual pseudo-random transformation. It employs SPN with a block size of 64-bits, supporting key-lengths of 64, 80, 96, 112 and 128-bits. VH is secure against known attacks and efficient in software as well as hardware implementations using about 3182GE on 0.18μm technology and requires about44.47Mb/s to encrypt a plaintext block on an 8-bit microcontroller.
LRBC [67]	LRBC is a 20 round lightweight block cipher that leverages both Feistel and SPN structures for faster diffusion. The block-size and the key-size of this cipher is 16-bits. The operations used are XOR, XNOR, concatenation etc. The cipher proves to be very secure with the use of both SPN and Feistel structures along with linear box. It is resistant against linear and differential cryptanalysis. Its key-consideration scheme allows it to consume less power and memory while maintaining a high-level of security.
PriPresent [79]	PriPresent (Prime Number based PRESENT) is a lightweight cipher utilizing SPN iterating over 31 rounds primarily focused on data integrity and confidentiality. It uses a block size of 64/80 bits and a key-size of 80/128bits. PriPresent incorporates theSHA-256 hash algorithm for maintaining data integrity. In terms of encryption time, the cipher is slightly slower than PRESENT due to the overhead of ensuring data integrity. However, it produces better throughput than PRESENT.

RECTANGLE [32]	RECTANGLE is a lightweight block cipher that uses bit-slice technique in a lightweight manner to perform with decent efficiency in both hardware and software. It is based on a SPN structure which has a block-size of 64-bits iterating over 25 rounds with a key-length of 80 or 128-bits. Its name RECTANGLE is conceptualized based on the 4$\times$16 array. The cipher is secure against known attacks.
ANU [44]	ANU is an ultra-lightweight block cipher leveraging Feistel network, iterating over 25 rounds operating on 64-bit block size using two key options of 80 and 128-bits. The key-schedule is inspired by the key-schedule of PRESENT for being one of the most secure key-scheduling scheme. ANU's design is resistant to linear, differential, biclique, zero-correlation, impossible differential, MIMT, and algebraic attack as well. ANU is resource and energy efficient as well.
ANU-II [45]	ANU-II is an enhanced version of the lightweight block cipher ANU. It is a 25 round Feistel structure that uses 64-bit block size supporting the key size of 80 and 128-bits. ANU-II is resistant to basic and advanced attacks on lightweight ciphers using P-layer.It uses a minimum number of rounds resulting in less-power consumption. The cipher is proven to be secure against linear attacks with 24 rounds and differential attacks with attack complexity of $2{\^{}4}$ as the defined limit and also against biclique and avalanche attack. ANU-II uses 20\% less power than ANU. It has faster throughput and execution time than ANU and requires lesser power than PRESENT and ANU.
HERMES [55]	HERMES is a lightweight block cipher that uses a block size of 64-bits, a key-size of 128-bits and iterates over 30 rounds. Its main focus was on software implementations with an attempt to keep the code size and memory consumption minimal. The cipher when tested using the NIST battery shows that it passes all the tests which include component tests, statistical tests and diffusion tests.

Loong [66]	Loong is a lightweight block cipher that is involutional that uses a new SPN structure which is well suited for lightweight cryptography as its involutional property enables it to use the same procedure for both encryption and decryption. It has a block size of 64 bits and has three key options of 64 bits, 80 bits and 128 bits. Corresponding to the key size, it iterates over 16, 20 and 32 rounds respectively. It proves to be very efficient in hardware as well as in software as it enables reusability of the code. The cipher performs very well against differential and linear attacks and is also resistant to meet-in-the middle attack, related-key attacks and other known attacks.
MAES [68]	Modified AES or MAES is a lightweight version of the original Advanced Encryption Algorithm (AES) in which a new S-box (1-dimensional) is presented by the development of a new equation for the construction of a square matrix in the affine transformation phase of MAES. Besides the new S-box, the cipher has the same parameters as AES. The arithmetic operations in MAES are carried out over Galois Field 24. This new novel design results in increased efficiency and also helps in reducing the latency of the original AES which makes it suitable for lightweight applications.
NVLC [75]	NVLC is a lightweight encryption algorithm that uses a block size of 64 bits and has two key options of 80 and 128 bits. It is based on an SPN structure and iterates over 20 rounds. It uses key whitening schemes at the beginning and end of the encryption. The cipher is designed to resist differential and linear attacks and possesses full diffusion for shortcut attack accumulation resistance. Its adoption of PRESENT's key schedule also helps it to defend against various other known attacks as well.
OLBCA [31]	OLBCA is a lightweight block cipher that has a block size of 64-bits and a key length 80 bits. Iterating over 22 rounds, the cipher is based on a SPN structure. Upon careful analysis, it can be observed that OLBCA has more than twice the number of active S-Boxes than PRESENT at the twenty round mark. The proposed cipher also proves to be better than PRESENT in terms of resistance against differential and integral attacks. The cipher is also faster and is also more cost efficient when compared to PRESENT.

PUFFIN [80]	PUFFIN is an involutional lightweight block cipher that operates on a 64-bit block and uses a 128-bit key. The cipher is based on an involutional SPN structure. Due to its involutional structure, the cipher proves to be very efficient in hardware and provides sufficient resistance against differential and linear attacks. Furthermore, due to the cipher's non-regularity in the key schedule, it is resistant to related-key attacks and no weak keys exist in the cipher.
SAT_Jo [84]	SAT_Jo is a lightweight block cipher that combines advantages of the block ciphers DES and PRESENT. It operates on a 64-bit block and uses a key length of 80 bits. It is based on the SPN structure that iterates over 31 rounds. The security analysis of this cipher shows it to be resistant to differential, linear and algebraic attacks.
SIMON and SPECK [11]	SIMON and SPECK family of lightweight block ciphers were born with both operating on variable block sizes of 32, 48, 64, 96 and 128 bits offering key-lengths of 64-bits, 72 or 96-bits, 96 or 128-bits, 96 or 144-bits and 128, 192 or 256-bits respectively. The SIMON and SPECK lightweight cipher does not use S-Boxes. SIMON uses XOR and two rotations. While as SPECK uses ARX operations. SIMON and SPECK any may be iterated as many times as necessary for desired security.
SFN [85]	SFN (SPN Feistel Network) is a lightweight block cipher that incorporates both SPN and Feistel network structure. The block size on which the cipher operates is 64 bits and the key length of this cipher is 96 bits iterating over 32 rounds. Due to high confusion and diffusion, weak keys in this cipher are non-existent making its key expansion method resistant to related key attacks. The cipher's 32-bit control keys provide enough security for it to be resistant against impossible differential attack and integral attack, meet-in-the-middle attack, and differential linear attacks.

Eight-Sided Fortress [30]	Eight-Sided Fortress is a lightweight block cipher, designed using the inspiration from PRESENT and LBlock. It uses a block size of 64-bits and a key length of 80 bits. The cipher's key schedule design is borrowed from PRESENT's key schedule design. Adopting tested designs from PRESENT and LBlock, ESF's design is resistant to all known attacks and the cipher performs efficiently in hardware and software implementations
QTL [82]	QTL, an ultra-lightweight block cipher uses a block size of 64-bits and supports 64 and 128-bit keys with the 64-bit key variant iterating over 16 rounds and the 128-bit key variant iterating over 20 rounds. Their proposed method improves security by enabling a differential characteristic probability to a low and the best linear characteristic approximation thereby making it secure against differential linear cryptanalysis. Not using a key-schedule allows it to reduce the cost of energy consumption in hardware implementation and makes it secure against related-key attacks
Piccolo [41]	Piccolo is a 64-bit lightweight block cipher that uses a variant of GFN (Generalized Feistel Network) as its underlying structure. It supports two key options of 80 (Piccolo-80) and 128 bits (Piccolo-128) iterating over 25 and 31 rounds respectively. The permutation in the key scheduling is carefully chosen in order for the cipher to be secure in opposition to related-key differential attacks and MITM attacks. The cipher is also resistant to differential and linear attack, boomerang attacks and impossible differential attack and very efficient in hardware.
PRESENT [29]	PRESENT is a 31-round, SPN based ultralight block cipher that uses a block size of 64-bits and supports two key sizes of 80 (PRESENT-80) and 128 (PRESENT-128) bits. It holds the status of being one among the recommended lightweight block ciphers by ISO/IEC standard. PRESENT is one of the most popular lightweight ciphers with it taking only 1570 GE in hardware. The cipher is resistant to known attacks such as algebraic attacks, key schedule attacks, differential linear attacks.

HIGHT [12]	HIGHT is a block cipher designed for resource-intense devices that operates on a block size of 64-bits and uses a 128-bit key. The cipher is based on a variant of GFS (Generalized Feistel Structure) that iterates over 32 rounds. On 0.25m technology which is not optimized for area, the cipher takes 3048 GE with a throughput of 150 mbps at 80 MHz. The security evaluation of HIGHT also shows that the cipher possesses enough security to defend against all known attacks.
GIFT [37]	GIFT is a Substitution Permutation Network (SPN) based lightweight block cipher that uses two block size options of 64 bits (GIFT-64) and 128 bits (GIFT-128) bits, with both of the variants using a 128-bit key iterating over 28 and 40 rounds respectively. The designers of the cipher do not claim any security regarding related-key attacks but enough security against linear and differential attacks, impossible differential attacks, algebraic attacks, meet-in-the-middle attacks, nonlinear invariant attacks, invariant subspace attacks and integral attacks.
SKINNY Family [69]	SKINNY lightweight block ciphers are a family that is based on SPN structure and follows a TWEAKEY framework taking a tweakey input. It uses a block size of 64 or 128 bits. Corresponding to the block size, it uses three key-length options of sizes n-bits, 2n-bits or 3n bits, n being the size of the block. The 64-bit block version of this cipher iterates over 32 rounds for n bit key size, 36 times for key-length 2n, and 40 times for key-length 3n. Similarly, the 128-bit block variant of this cipher iterates over 40 rounds for n bit key length, 48 times for key-length 2n, and 56 times for the key-length 3n. SKINNY outperforms SIMON in terms of both area and throughput when round-based implementation is used.
MANTIS [69]	MANTIS is a tweakable lightweight block cipher that is based on the α-reflective structure of PRINCE block cipher and uses the round function of the cipher Midori. All the variants of MANTIS use a 64-bit block-size using a key length of 128 bits with a 64-bit tweak and the only difference is the number of rounds they iterate over. It is based on FX construction, the whitening keys are applied before and after utilization of its core components. The security claims of the authors of MANTIS are the same as that of PRINCE except they also claim the related-tweak security.

BORON [46]	BORON is an ultra-lightweight block cipher leveraging Substitution Permutation Network (SPN) that iterates over 25 times. It has two key-length options of 80-bits and 128-bits. BORON's implementations in both hardware and software are very efficient, with it taking only 1929 GE for the 128-bit key variant and 1626 GE for the 80-bit key variant. The cipher also displays decent amount of resistance against known attacks like linear attacks, differential attacks, algebraic attack, related key and slide attacks.
LiCi [63]	LiCi is a feistel based ultra-lightweight block cipher that consists of 31 rounds. It operates on a block size of 64 bits and uses a key length of size 128 bits. The design of LiCi shows to perform very well in both hardware and software platforms. The cipher requires just 1153 GE and its memory and power requirements are also low. The cipher is shown to resist known attacks like linear attacks, differential attacks and advanced attacks like Biclique and Zero correlation attack.
DoT [49]	DoT is a Substitution Permutation Network (SPN) based ultra-lightweight block cipher that iterates over 31 rounds. It uses 64-bit block size and key-length options of 80 and 128 bits. The key scheduling algorithm of this cipher is inspired by the key scheduling technique of the lightweight block cipher PRESENT due to it being very secure. The cipher shows good resistance to attacks like linear attack, differential attack and biclique attack and the cipher consumes 993 GE in hardware implementation and 2464 bytes of flash memory only. The cipher also shows to be 250 times faster than PRESENT when it comes to throughput.
Zorro [92]	Zorro is an AES-like lightweight block cipher that is based on generalized Even-Mansour construction that iterates over 24 rounds. Zorro follows no complex key schedule as the master key is added simply to the state in a bitwise manner, just like in the lightweight block cipher LED. The authors show that the cipher is resistant to various known attacks except for related-key attacks for which they claim no security.

CLEFIA [26]	CLEFIA is a lightweight block cipher that uses a block size of 128 bits and uses key-length options of 128, 192 and 256 bits. It iterates over 18, 22 and 26 times for 128, 192 and 256-bit keys respectively. CLEFIA's implementation of Diffusion Switching Mechanism (DFS) ensures resistance against major attacks. CLEFIA holds the position of being on the list of CRYPTREC which is a standardization body of the Japanese government. It is also one of the lightweight block ciphers among the only two recommended by the ISO/IEC standard.
RoadRunneR [83]	RoadRunneR is a lightweight bitslice block cipher that is based on the Feistel structure and uses a block size of 64 bits. It uses two key-length options of 80 and 128 bits iterating over 10 and 12 rounds respectively. Key whitening procedure is carried out. Implementation results of RoadRunneR show only SPECK and PRIDE to perform better than RoadRunner in some areas, but when it comes to security, RoadRunner provides better security than SPECK and PRIDE, particularly in resisting differential and linear attacks.
LILLIPUT [64]	LILLIPUT is a lightweight block cipher based on Extended Generalized Feistel Network (EGFS) which is a TWINE-like GFS using an additional linear layer, operating on a block size of 64 bits and using a key size of 80-bits. Due to its involutive nature, the cipher design is compact and it takes about 1055 GE when implemented in a serialized manner. In the round-wise implementations, it takes around 1581 GE. Integral attack or Impossible differential attack reach fewer rounds for LILLIPUT than for TWINE due to its better diffusion mechanism.
LiARX [62]	LiARX is a lightweight block cipher based on Long Trail Strategy (LTS). It uses a block size of 64-bits and a key-length of size 128 bits. The S-Box in LTS is replaced with an ARX box. In terms of performance, the cipher is compared with SPECK and LED utilizing FELICS. The cipher due to its round function being larger than that of SPECK does not perform as good as SPECK, but due its ARX-based design, it does outperform LED.

PRIDE [77]	PRIDE is a 64-bit lightweight block cipher utilizing the Substitution Permutation Network (SPN) that iterates over 20 rounds. The cipher utilizes the FX construction. The cipher is shown to outperform most of the existing ciphers at the time in terms of both code size and cycle count.
Lilliput-AE [65]	Lilliput-AE is a lightweight block cipher that uses a tweakable block cipher Lilliput-TBC which is based on Extended Generalized Feistel Network (EGFN). Lilliput-TBC uses three key-length options of 128, 192 or 256 bits tweak-length options of 128 or 192-bits iterating over 32, 36 and 42 rounds respectively. These parameters give rise to six variants of the cipher which are grouped into two sub groups of Lilliput-TBC-I and Lilliput-TBC-II. Lilliput-TBC-I uses all the three key-length options with 192-bit tweak and Lilliput-TBC-II uses the three key-length options with 128-bit tweak. Various attacks like differential and linear attacks, related tweakey boomerang attacks, structural attacks, subspace attacks, algebraic attacks, are carried out which show the cipher to be very secure and does resist most of the attacks by quite a margin
NOEKEON [74]	NOEKEON is an iterated lightweight block cipher that uses a block key size of 128-bits with its design inspired from its predecessors using a self-inverse bit-slice design without using any key-schedule. The authors of the cipher show it to be secure against all known attacks using differential linear attacks, truncated differentials, interpolation attacks and related-key attacks. The cipher's serf-inverse bit slice design also helps it to be compact in hardware.

PRINTcipher [78]	PRINTcipher is a Substitution Permutation Network (SPN) based lightweight block cipher that takes its inspiration from the block cipher PRESENT with the main difference between PRESENT and PRINTcipher being the absence of a key schedule and the S-Boxes. PRINTcipher has two variants which are PRINTcipher-48 and PRINTcipher-96. PRINTcipher-48 uses a block size of 48 bits, a key-length of 80 bits and iterates over 48 rounds while PRINTcipher-96 uses a block size of 96 bits, a key length of 160 bits and iterates over 96 rounds. The security evaluation of the cipher using related-key attacks, differential and linear attacks, algebraic attacks, and statistical saturation attacks show the cipher is secure even after the absence of a key schedule.
FeW [51]	FeW is a 32-round lightweight block cipher that is based on Feistel-M structure which is a blend of Feistel 4-branch Generalized Feistel Structure (GFN). It operates on a block size of 64-bits and uses two key length options of 80 known as FeW-80 and 128 bits known as FeW-128. The security analysis of FeW shows that the cipher does possess enough margins of security towards known attacks. The software implementation shows it to perform significantly better than PRESENT and RECTANGLE block ciphers.
Halka [54]	Halka is a Substitution Permutation Network (SPN) based block cipher that uses a block size of 64-bits and a key-length of 80 bits and it iterates over 24 rounds. The 8-bit multiplicative inverse S-Box provides the cipher with enough resistance to defend against differential and linear attacks. The cipher also stays strong against related key attacks, side attacks, and other attacks like structural attacks, algebraic attacks, cube attack and side channel attack. The cipher is shown to be almost as efficient as PRESENT in terms of hardware implementations while maintaining a higher level of security than PRESENT.

CRAFT [47]	CRAFT (with effiCient pRotection Against differential Fault analysis aTtacks) is a lightweight tweakable block cipher whose design is based on efficient implementation that allows protection to Differential Fault Analysis (DFA) while providing strong security bounds in the related tweak model. CRAFT has a 64-bit plaintext/ciphertext and 128-bit key-length and supports a 64-bit tweak adding little overhead to the corresponding implementations and has an SPN-structure. A 128-bit security is claimed in the related tweak model.
FLY [52]	FLY is a bit-sliced cipher specifically targeting 8-bit microcontrollers, with blocks of size 64-bits and 128-bit key length iterating over 20 rounds. It uses Lai-Massey as its underlying structure and bit-sliced S-boxes. Its round function is optimized for 8-bit microcontrollers. It allows two key-schedules, one that is elementary and the other that is Related-Key Attack resistant. MIMT, algebraic, impossible differential and integral attacks are a lesser concern.
GRANULE [53]	GRANULE is an ultra-lightweight feistel based cipher that encrypts 64-bits of data with 80/128-bits of key iterating over 32 rounds that allows it to be resistant to all possible kinds of attacks. GRANULE needs 1288GEs and 1577GEs for 80-bit and 128-bit key sizes respectively. GRANULE requires low area footprints and allows a robust and secure design thus providing resistance to biclique attack, collision attack, key schedule attack, meet in the middle attack, and key zero correlation attack.
QARMA [81]	QARMA is a tweakable block cipher family specifically targeting applications relating to tag generation (short-tags) for prevention of software exploitation assisted with hardware, for memory encryption and construction of keyed hash functions. QARMA is inspired by the block ciphers PRINCE and MANTIS. QARMA allows a block size of 64 and 128-bits and equal tweak size as that of block size with key-sizes twice as long as the block size.
MARVIN [71]	Marvin follows the extended LS-design criteria where the key is of the same dimensions as the input state. The cipher is designed such that it suits a 256-bit block size along with a 256-bit key as well, iterating over 16 rounds. Linear and differential characteristics do not exist after two rounds, no impossible differential characteristic, truncated differentials and integral characteristics exist after four rounds.

MANTRA [70]	MANTRA is an ultra-lightweight cipher which is a feistel based cipher operating on 64-bits of plaintext supporting key-sizes of 80/128-bits iterating over 32 rounds. The cipher needs a small footprint area thus consuming only 1662GE and 1374GE for key-sizes 128-bits and 80-bits respectively. MANTRA has shown good resistance against linear, differential and algebraic attacks.
DESL **DESXL** [48]	DESL is based on classical DES design, but the difference is in the redundant use of S-boxes in this variant. This variant has a brute-force resistance of 256. The DESXL variant is yielded so as to strengthen the cipher through key-whitening procedure. DESL is secure against differential cryptanalysis and algebraic attacks. The major difference between DES and DESL lies in the f-function which uses a single cryptographically strong S-box repeated 8-times.
SLIM [88]	SLIM is a 32-bit ultra-lightweight block cipher utilizing the Feistel structure targeted for RFID systems. SLIM focuses on security and simplicity as its core design principles. The cipher achieves resistance against exhaustive search by using 80-bit key length which is a NIST recommendation. It operates on 32-bit plaintext iterating over 32 rounds. The cipher is resistant to differential and linear cryptanalysis and provides a decent security margin.
EPCBC [50]	EPCBC is a lightweight feistel based cipher structure that has a block size of 48/96-bit along with a 96-bit key size. It is based on a generalized PRESENT and a customized key schedule allowing it to be resistant in case of related-key attacks. It comes in two variants: EPCBC(48,96) having a block size of 48-bits and a key-size of 96-bits and EPCBC(96,96) with a block key-size of 96-bits. The full version of EPCBC is immune to higher order differential attacks, integral attacks and statistical saturation.

I-PRESENT^{TM} [57]	I-PRESENT^{TM} is an involution lightweight block cipher for environments of resource constraint nature. The PRESENT cipher was used as an inspiration for the design of this cipher. The cipher operating on a 64-bit plaintext allows key-lengths of 80 128-bits thus giving rise to two variants: I-PRESENT-80 and I-PRESENT-128. The cipher iterates over 30 rounds. The involution part of the cipher is inspired by the lightweight block cipher PRINCE.
HISEC [56]	HISEC is a lightweight block cipher adopting characteristics from PRESENT cipher with a different bit permutations. It makes use of a modified feistel structure. HISEC operates on a 64-bit plaintext and a 80-bit key iterating over 15 rounds. The cipher is resislient against differential boomerang attacks and can go until four rounds with a complexity of 2^{55} in integral attack to recover the 32-bits of key.
PICO [76]	PICO is a SPN based ultra-lightweight cipher operating on a block size of 64-bits and a key-size of 128-bits iterating over 32 rounds. Its different design allows generation of S-boxes in fewer rounds thus thwarting linear and differential cryptanalysis. The cipher is promising in both hardware and software.
SIT [87]	SIT is a lightweight block cipher with a block size and key length of 64 bits. It uses a hybrid algorithm that is based on a combination of Feistel and Substitution Permutation Network (SPN). This cipher requires only 5 rounds to achieve desired security levels. On the ATmega 328 platform, the execution time of this cipher is shown to be 0.188 milliseconds. For the encryption and decryption process, the cipher takes 0.087 milliseconds. In terms of security, the cipher is shown to be resistant to linear and differential attacks, related key attacks, interpolation attacks and square attacks.