Self-synchronized Encryption for Physical Layer in 10Gbps Optical Links

Adrián Pérez-Resa, Miguel Garcia-Bosque, Carlos Sánchez-Azqueta, and Santiago Celma

Abstract— in this work a new self-synchronized encryption method for 10 Gigabit optical links is proposed and developed. Necessary modifications to introduce this kind of encryption in physical layers based on 64b/66b encoding, such as 10GBase-R, have been considered. The proposed scheme encrypts directly the 64b/66b blocks by using a symmetric stream cipher based on an FPE (Format Preserving Encryption) block cipher operating in PSCFB (Pipelined Statistical Cipher Feedback) mode. One of the main novelties in this paper is the security analysis done for this mode. For the first time, an expression for the IND-CPA (Indistinguishability under Chosen-Plaintext Attack) advantage of any adversary over this scheme has been derived. Moreover, it has been concluded that this mode can be considered secure in the same way of traditional modes are. In addition, the overall system has been simulated and implemented in an FPGA (Field Programmable Gate Array). An encrypted optical link has been tested with Ethernet data frames, concluding that it is possible to cipher traffic at this level, getting maximum throughput and hiding traffic pattern from passive eavesdroppers.

Index Terms—Optical Communications, Ethernet, self-synchronous encryption, Pipeline Statistical Cipher Feedback.

1 Introduction

TODAY, high speed optical networks are a reality. Thanks to the advances in these technologies it is possible to afford the bandwidth growth that nowadays modern applications demand [1], such as cloud computing and big data. In addition, information security has become an important issue as the volume of threat events has increased over the last years [2]. Failures in security can lead to the malfunction of a service or the confidentiality loss in customer critical information.

In a layered communication system, such as OSI (Open System Interconnection) or TCP/IP (Transmission Control Protocol/Internet Protocol), passive or active attacks can be carried out at different communication levels. Depending on the communication layer, different approaches are used for getting information confidentiality. For example, standardized protocols such as MACsec [3] or IPsec [4] are usually used at layer 2 (Data link layer) and layer 3 (Network layer), respectively. In these cases, encryption is carried out in each frame individually.

For the particular case of optical networks, the threat analysis in its physical layer is also considered critical to guarantee secure communications. [5], [6]. Among the most important attacks at this level, signal splitting attacks must be taken into consideration. Nowadays thanks to low-cost tapping techniques it is possible to intercept the optical signal without the need to perceptibly interfere in communications or create visible side-effects [7].

To deal with these threats and protect data confidentiality, several physical layer mechanisms related with photonic technologies have been proposed [8], for example

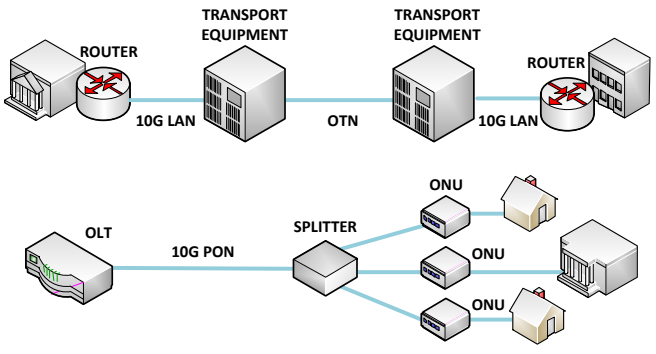


Fig. 1. Example of two different 10 Gigabit Ethernet standards in optical networks for WAN and EFM applications: 10GBase-R (upper) and 10GBase-PR (lower).

OCDM (Optical Code Division Multiplexing) [9], SCOC (Secure Communications using Optical Chaos) [10] or QKD (Quantum Key Distribution) [11]. Other techniques, related with physical layer protocols, cipher the information at bit level, for example the encryption of OTN (Optical Transport Network) frame payloads [12]. Some of the advantages claimed by these techniques are that they achieve in-flight encryption introducing null overhead and a very low latency (in the range of nanoseconds) in data packets [12]. In fact, OTN communication equipment performing encryption at line rate and getting a 100% throughput are already available on the market [13]. This contrasts with what protocols at other layers do [14], [15]. For example IPsec usually introduces latencies in the range of milliseconds. Moreover, the overhead introduced by IPsec during encryption limits the total throughput to values between 20% and 90% of the maximum achievable [16], [17].

Some applications for 10 Gigabit Ethernet standards are

• *Adrián Pérez-Resa, Miguel Garcia-Bosque, Carlos Sánchez-Azqueta and Santiago Celma are with the Group of Electronic Design, (GDE), Aragón Institute of Engineering Research (I3A) Zaragoza University, CP 50009. E-mail: aprz@unizar.es, mgbosque@unizar.es, csanaz@unizar.es, scelma@unizar.es.*

Corresponding author: A. Pérez-Resa.

xxxx-xxxx/0x/$xx.00 © 200x IEEE Published by the IEEE Computer Society

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

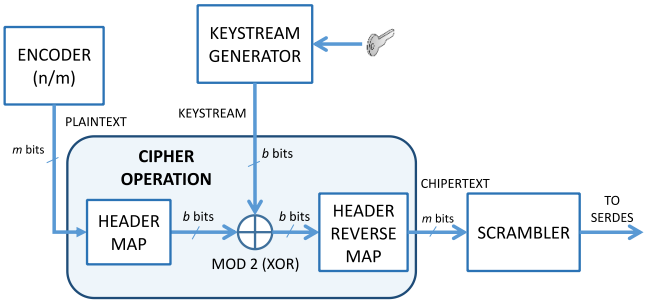


Fig. 2. Location and generic structure of a stream cipher in a physical layer with a dense block line encoding. In the case of 64b/66b encoding, parameters n, m, and b are 64, 66 and 65 bits, respectively. The output of the encoder is a 66-bit block whose 2-bit header is mapped to a value '0' or '1' and concatenated with the rest of the block in the HEADER MAP module. The result is a 65-bit block that is encrypted in a one-time pad fashion thanks to the XOR operation.

shown in Fig. 1. Nowadays one of the most used technologies for the access to optical transport networks is Ethernet, and for high data rates 10 Gigabit Ethernet is widely deployed in MAN (Metro Area Networks) and WAN (Wide Area Networks) environments. Optical 10 Gigabit Ethernet standards are also available for EFM (Ethernet in the First Mile) applications, allowing customers the access to the provider network through a PON (Passive Optical Network) infrastructure.

Regarding the physical layer security in these standards, a new mechanism was proposed in [18]. The encryption solution consisted of a symmetric chaotic stream cipher, suitable for the encryption of 64b/66b blocks. This kind of encryption could provide the mentioned advantages such as maximum throughput and low latency. However, although stream ciphers are suitable for high speed applications, their cryptanalysis and design criteria are less understood than block ciphers [19]. Indeed a stream cipher application can be implemented easily thanks to a secure block cipher such as AES working in CTR mode, considered secure thanks to its formal security proof [20]. Furthermore, in [18] the implemented stream cipher is based on a chaotic structure whose cryptanalysis could be not clear enough and it is mainly based on its randomness analysis.

On the other hand, the encryption system in [18] is not self-synchronized. To get synchronization, a mechanism was implemented based on the usage of new ordered sets in a specific 64b/66b block type, which increments the complexity of the overall system. In addition, in case of missing the synchronization in the middle of an encryption session there is a lack of a protocol to recover it.

In this work, a complete solution to overcome the mentioned disadvantages is proposed.

Regarding self-synchronization, in Ethernet data stream at PCS level there is no possibility to synchronize TX and RX stream ciphers thanks to standardized data fields or structures, as in OTN, where data stream is composed of continuous and periodic data containers.

To get synchronization in the symmetric encryption scheme, a self-synchronized operating mode called PSCFB has been analyzed and implemented. Moreover, a formal security expression for this operation mode has been deduced, concluding that it can be considered secure in the same way as other traditional modes.

The paper is divided into the following sections. Section 2 explains PCS layer encryption necessities when using 64b/66b encoding, in Section 3 an introduction to the PSCFB (Pipeline Statistical Cipher Feedback) mode of operation is made. In Section 4 IND-CPA (Indistinguishability under Chosen-Plaintext Attack) advantage expression for this operation mode is proposed. Subsequently, Section 5 deals with the practical case of 10 Gigabit Ethernet, particularly with the standard 10 GBase-R, and the overall scheme of the proposed encryption system. In Section 6, the hardware implementation of the cipher is described while in Section 7 results of the encryption are explained. Other security considerations as key distribution are taken into account in Section 8. Finally, in Section 9 conclusions are given.

2 Coding Preserving Encryption

Typically, in Ethernet standards, the physical layer is divided into three sublayers, PCS (Physical Coding Sublayer), PMD (Physical Medium Dependent) and PMA (Physical Medium Attachment). The Physical Coding Sublayer carries out functions such as link establishment, clock rate adaptation, data encoding and scrambling.

Optical Ethernet standards are high-speed communication systems where a baseband serial data transmission is carried out while clock frequency information is embedded in the serial bitstream itself. At the receiver, the clock recovery circuits must be able to extract the frequency information thanks to the bit transitions in the data stream. After that, serial data sampling can be made at the appropriate time. In order to facilitate the work of the CDR (Clock and Data Recovery) circuit, information must be encoded in such a way that a good transition density and a short run length are achieved. Also a DC-balanced serial data stream must be guaranteed, which is important for some transmission media, such as optical links.

In the case of PCS sublayers using a dense coding such as 64b/66b the mentioned properties, DC-balance and transition density, are achieved in a statistical way thanks to the scrambling of the bitstream. On the other hand, the short run length is guaranteed thanks to a synchronization header at the beginning of each 66-bit block, whose only two possible values are '10' or '01'.

Usually stream ciphers are implemented by carrying out the XOR operation between the plaintext and a keystream obtained from a secure pseudorandom generator. In case of using physical layer encryption in the PCS sublayer, it is necessary to preserve the properties of the block line encoder, therefore the location of the XOR operation in the datapath must be taken into account. For 64b/66b encoding where firstly the blocks of bits are formatted and finally processed by a scrambler, the stream cipher should implement the XOR operation before the scrambler to guarantee that the scrambler transfers its statistical properties to the resulting bitstream before being transmitted.

As mentioned before, the synchronization header of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

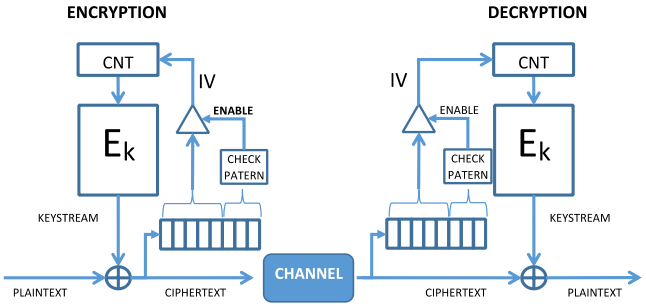


Fig. 3. Structure of PSCFB mode for encryption and decryption. The block cipher $E_K(\cdot)$ has a block size of L bits, and is implemented using P internal stages. The difference between the encryption and decryption is that the sync pattern scan is performed after the XOR operation in the transmitter and before the XOR in the receiver.

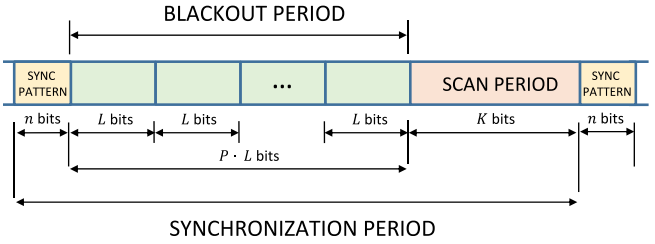


Fig. 4. Structure of the synchronization period in PSCFB mode. A complete synchronization period is formed by the sync pattern, blackout period and scan period. As the cipher has P stages, the encrypted value of the IV, $E_K(IV)$ is used as keystream P blocks later, when it has been fed back as a new counter value.

each 66-bit block is composed of a pair of bits with two possible values '01' or '10'. On the one hand, it is necessary for avoiding long runs of zeros or ones, which limits the run length to a maximum of 66 bits. On the other hand, it is used to detect the 66-bit block boundary in the receiver and perform the decoding process properly. For this reason, this 2-bit header is kept untouched and is not scrambled as the rest of the 66 bit block.

To preserve the two-bit transition of the sync header, the XOR operation must be carried out as shown in Fig. 2. The 2-bit header must be mapped to values '0' or '1' depending on whether it is equal to '01' or '10', respectively. Then the mapped value is concatenated to the 64-bit block payload. These two operations are performed in the HEADER_MAP block. The resulting output is a 65-bit word that is XORed with the keystream, giving a new 65-bit word. The first bit of this word will be reverse mapped giving the new 2-bit header while the subsequent 64 bits will be directly the new payload. The concatenation of both data fields will result on the new ciphered 66-bit block. These last operations are performed in the HEADER_REVERSE_MAP module.

3 Self-Synchronized Encryption

3.1 Self-synchronized Stream Ciphers

Self-synchronized stream ciphers usually generate the keystream as a function of the key and the preceding ciphered bitstream. In spite of its self-synchronizing properties, this kind of ciphers are less understood and their security analysis is more difficult than typical synchronized stream ciphers. Indeed, there are few proposals of these algorithms. For example, only two of the proposed stream ciphers in eSTREAM project, SSS and Mosquito were self-synchronized [21]. However, they were dismissed owing to their vulnerabilities [22], [23].

On the other hand, stream ciphers can also be based on different operating modes of block ciphers, such as OFB (Output Feedback), CFB (Cipher Feedback) or CTR (Counter) modes [24]. For self-synchronized purposes CFB is the only one recommended by the NIST (National Institute of Standards and Technology). However, to achieve synchronization with a loss of an arbitrary number of bits, CFB mode must only feedback one ciphertext bit for every block cipher operation. For this reason, if the block size of the underlying block cipher working in CFB mode is L, the CFB resulting throughput would be $1/L$ of the underlying block cipher.

To solve this throughput limitation, SCFB (Statistical Cipher Feedback) [25] and OCFB (Optimized Cipher Feedback) [26] modes were proposed. Particularly, SCFB was deeply analyzed in [27], [28], and in [29] was compared with OCFB, resulting in better properties for high-speed physical layer security. In spite of this advantage, it is recommended that an implementation of conventional SCFB be constrained to 50% of the throughput of its underlying block cipher. This constraint is necessary to ensure that no bits are lost due to queue overflow in the SCFB system.

To overcome this limitation, PSCFB (Pipelined Statistical Cipher Feedback) was proposed [30]. This mode allows an efficient utilization of a block cipher using a pipeline architecture, which results in implementations with a throughput near to 100%.

3.2 PSCFB Mode of Operation

The PSCFB mode of operation is essentially a hybrid of CFB and CTR modes, where the underlying block cipher is implemented with a pipelined architecture of P stages and a block size of L bits. Let us assume that the cipher is configured with a key K and it has an encryption function $E_K(\cdot)$. The cipher operates in conventional CTR mode while scanning the ciphertext looking for a special n-bit length synchronization pattern. Assuming that the underlying block cipher is a good PRP (Pseudo Random Permutation), each value of the keystream block cipher output can be seen as randomly and independently chosen, giving also random and independent values of ciphertext blocks after performing the XOR. Therefore, the sync pattern will be observed at a statistically random point in the ciphertext stream. When this pattern is detected, the next L bits are captured and used as an initialization vector (IV) that feeds back the counter value at the block cipher input. Therefore, it can be considered that the block cipher temporarily works in CFB mode. On the other hand, it is necessary to disable the sync pattern scanning since the IV is captured until $E_K(IV)$ is available as new keystream block. This interval is called the blackout period.

In Fig. 3 and Fig. 4 the structure of PSCFB mode and the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

complete synchronization cycle are shown, respectively.

4 Security considerations

4.1 Background of PSCFB security

Regarding to the security of SCFB and PSCFB modes, the probability of generating repeated keystream blocks has been analyzed in [22] and [27]. If the counter reaches some value already used in previous synchronization cycles, the keystream will be repeated until next sync pattern detection. This issue would compromise the security of these modes.

The conclusion of these analyses is that the probability of repeated keystream is very low for both modes with typical size parameters, such as 128-bit block size. However, although PSCFB mode is built from two secure modes, CTR and CFB, with formal security proofs [20] [26], no formal security proof exists for PSCFB, and it has been let as an open problem [30].

In this section we discuss the security for PSCFB mode. This mode can be considered similar to other one called CTR$, for which a security proof exists. Based on this proof it is possible to derive a formal security expression for PSCFB for the first time and determine under which conditions it can be considered, at least, as secure as other traditional modes, e.g. CTR.

In the next subsection, CTR$ mode and its formal security proof are introduced. Authors consider that this explanation is convenient for the understanding of subsection 4.3, where the IND-CPA advantage for PSCFB is derived.

4.2 CTRC and CTR$ Modes

Concrete security analysis for CTRC and CTR$ were originally established in [20]. The CTRC scheme is a stateful (counter based and deterministic) while the CTR$ is a stateless (randomized) variant of CTRC.

Let us consider a family of functions F such that: $F: K \times \{0,1\}^l \rightarrow \{0,1\}^L$ where L is the block size and K the keyspace. Given a plaintext M formed by m L-bit blocks $\{M_0, M_1, \dots, M_{m-1}\}$, then the m blocks of the output ciphertext C_i are obtained in each $S\mathcal{E}$ (encryption scheme) CTRC and CTR$, applying their encryption functions. Encryption function for CTR$ is shown in Algorithm 1.

Algorithm 1. Function $\mathcal{E}-CTR\$^F(M)$

$R \xleftarrow{\$} \{0,1\}^l$
$CNT_0 = R + 1$
$K_i = F_K[CNT_i]\ for\ i = 0,1,\dots,m-1$
$CNT_{i+1} = CNT_i + 1\ for\ i = 0,1,\dots,m-2$
$C_i = (M_i \oplus K_i)\ for\ i = 0,1,\dots,m-1$
Return $\{C_0, C_1, \dots, C_{m-1}\}$

In this algorithm CNT_i and K_i are the values of the counter and keystream block in each encryption step. F_K is the underlying encryption function and R is an l-bit random value. In CTR$ the counter is set to a random value R at the beginning of each message encryption.

Usually the security of these modes is studied in the sense of IND-CPA (Indistinguishability under Chosen-Plaintext Attack) security [31]. An advantage expression is obtained thanks to a game between an active adversary A and an encryption oracle performing the encryption scheme $S\mathcal{E}$, configured with a key K and an experiment bit b.

It is demonstrated in [32] that an adversary B attacking the PRF security of F_K can be built thanks to the adversary A and their advantages are related as follows:

$$ADV^{IND-CPA}_{S\mathcal{E}(F)}(A) = 2 \cdot ADV^{PRF}_{F}(B) + ADV^{IND-CPA}_{S\mathcal{E}(Func)}(A) \quad (1)$$

where $ADV^{IND-CPA}_{SE(F)}(A)$ is the advantage of A attacking $S\mathcal{E}$ when its underlying encryption function is a PRF F_K, $ADV^{IND-CPA}_{SE(Func)}(A)$ is the advantage of A over $S\mathcal{E}$ when the underlying encryption function is a random function $Func(l, L)$ and $ADV^{PRF}_{F}(B)$ is the prf-advantage of B as defined in [31].

In the formal security proofs of CTRC and CTR$ modes the term $ADV^{IND-CPA}_{SE(Func)}(A)$ is obtained [32], then allowing to reach the final advantage expression. It is proven that:

$$ADV^{IND-CPA}_{S\mathcal{E}(Func)}(A) \leq \Pr(col) \quad (2)$$

where $\Pr(col)$ is the probability of a collision among the counter values used during the game.

Given the experiment bit b, during the attacking game the adversary performs q queries of a pair of messages. For each pair (M_i^0, M_i^1) it receives from the oracle the ciphertext C_i corresponding to the message M_i^b. Assuming that each encrypted message M_i^b has a length of m_i blocks, the counters used during the game session can be represented as in the following table:

$$\begin{matrix} r_1 + 1,\ r_1 + 2, \dots, r_1 + m_1 \\ r_2 + 1,\ r_2 + 1, \dots, r_2 + m_2 \\ \dots \quad\quad \dots \quad\quad \dots \\ r_q + 1,\ r_q + 1, \dots, r_q + m_q \end{matrix} \quad (3)$$

where r_i is the randomized counter value loaded at the beginning of the message M_i^b with length m_i. The subsequent counters from r_i in advance will be incremented up to $r_i + m_i$. According to (3), the probability of collision of every counter value, $\Pr(col)$, can be bounded with the following expression:

$$\Pr(col) \leq \frac{(q-1) \cdot \sum_{i=1}^{i=q} m_i}{2^l} \quad (4)$$

Thanks to (4) it is possible to derive an upper bound for $ADV^{IND-CPA}_{S\mathcal{E}(Func)}(A)$, and by replacing it in (1), the final $ADV^{IND-CPA}_{S\mathcal{E}(F)}(A)$ expression of CTR$ mode.

4.3 PSCFB vs CTR$

In PSCFB, the sync pattern will be observed at a statistically random point in the keystream. On the other hand, if

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

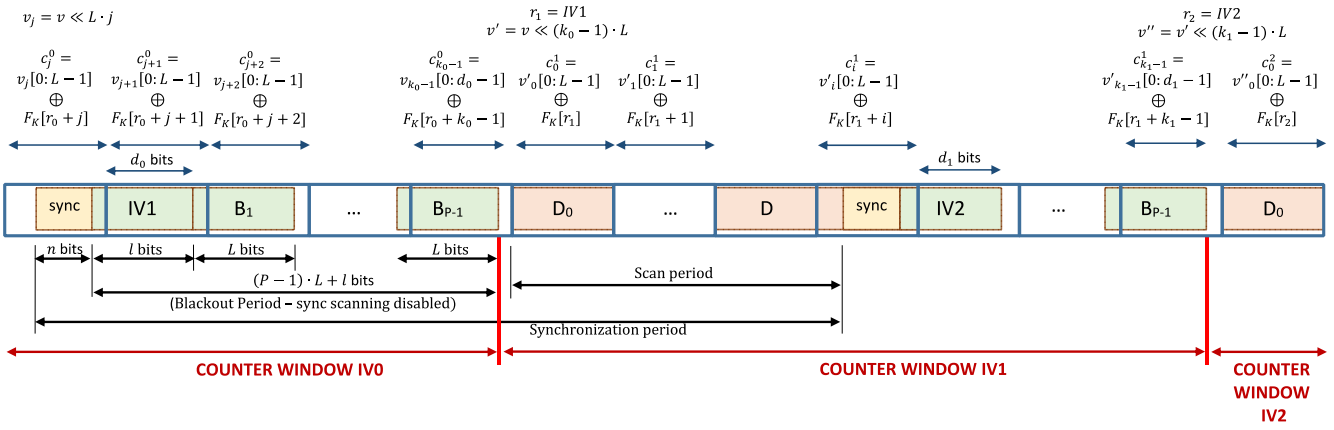


Fig. 5. Counter windows in PSCFB mode. Each IV used in each counter window is captured in the previous one. For example, IV_1 is captured at the beginning of the blackout period in counter window IV_0. Then its encrypted value $F_K(IV_1)$ is used as the first keystream block in next scan period (in counter window IV_1). In counter window IV_i the counter r_i takes values from IV_i to $IV_i + k_i - 1$ where $k_i = \lceil \mu_i / L \rceil$, μ_i is the length in bits of this counter window and L is the block size.

the block cipher is considered a good PRF and there are no collisions among the counter values, ciphertext blocks can be considered random and independent. Therefore, we can consider the new IV as a random value. As we consider the block cipher F such that: $F: K \times \{0,1\}^l \rightarrow \{0,1\}^L$, in this section we let the IV be an l-bit word and the blackout period will have a length of $(P-1) \cdot L + l$ bits.

Let us assume that the adversary tries a game over an oracle performing a PSCFB encryption scheme. The adversary sends q queries to the oracle, but now, unlike the CTR$ mode, the counter of the PSCFB scheme is not reinitiated randomly at the beginning of each message. The counter could be reseeded at a random point of each message. Depending on the length of the messages this initialization could happen more or fewer times. For example if the length of a message is shorter than the mean synchronization period, possibly in that message only a new sync pattern is received and then the counter is reinitiated only once to a random IV. However, for longer messages this could happen more times.

In general, in CTRC, CTR$ and PSCFB we can understand that the counter behaves in a cyclic fashion during the encryption session. We call this type of cycle a counter window. Inside each window, the counter is incremented and not repeated. In the case of CTRC there is only one counter window, which means that the counter is initialized only once, at the beginning of the first message, and never repeated. In CTR$ there are so many windows as encrypted messages because the counter is reinitialized randomly at the beginning of each message. In the same way in PSCFB there are so many windows as synchronization cycles are produced during the whole session, as the counter is reseeded at the beginning of each blackout period. The counter window will start at the beginning of a scan period and will finish at the end of the next blackout period, as shown in Fig. 5.

In addition, the counter window length is different in each mode. In CTRC and CTR$ the length is a number of bits that is multiple of the block size, while in PSCFB it is multiple of the block size plus a random number of bits between 0 and the block size. This fact depends on when the end of the blackout period happens, as it could be not aligned with the end of an encrypted block of bits. An example is shown in Fig. 5, where the counter windows IV_0 and IV_1 do not finish exactly in block boundaries.

Starting from (1), which can be considered generic for any counter encryption scheme we will have:

$$ADV_{PSCFB(F)}^{IND-CPA}(A) = 2 \cdot ADV_F^{PRF}(B) + ADV_{PSCFB(Func)}^{IND-CPA}(A) \quad (5)$$

As we can consider PSCFB a counter mode, it is possible to make the same assumptions as in [32] to reach equation (6), that is, if the block cipher is a good PRF and there are no repeated counter values during the game session, then the adversary has zero advantage in winning the game and the encryption scheme behaves as a one-time pad. Then next condition is fulfilled:

$$ADV_{PSCFB(Func)}^{IND-CPA}(A) \leq \Pr(col) \quad (6)$$

During the game session the adversary sends q messages of length m_i blocks. The block size is L bits and the total number of bits in the session is μ. Let us assume that in the case of PSCFB N sync cycles happen during the whole session. Therefore we can consider that the counter table (3) for PSCFB can be represented as:

$$\begin{matrix} r_1,\ r_1+1, \ldots, r_1+k_1-1 \\ r_2,\ r_2+1, \ldots, r_2+k_2-1 \\ \ldots \quad\quad \ldots \quad\quad \ldots \\ r_N,\ r_N+1, \ldots, r_N+k_N-1 \end{matrix} \quad (7)$$

where k_i is the length in data blocks for the i-th counter window. As the bit length of a counter window could not be a multiple of the block size, the length k_i will be: $k_i = \lceil \mu_i / L \rceil$, where μ_i is its length in bits and the operator $\lceil \cdot \rceil$ means its rounded up value.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

As in (4), $\Pr(col)$ for PSCFB case is obtained:

$$\Pr(col) \leq \frac{(N-1)\cdot\sum_{i=1}^{i=N} k_i}{2^l} \tag{8}$$

Since $k_i = \lceil \mu_i/L \rceil \leq \mu_i/L + 1$, then:

$$\Pr(col) \leq \frac{(N-1)\cdot\sum_{i=1}^{i=N}(\mu_i/L+1)}{2^l} \leq \frac{N\cdot\left(\frac{\mu}{L}+N\right)}{2^l} \tag{9}$$

Let us consider N_{max} the maximum number of sync cycles in μ bits and C_{min} the minimun posible size of a sync cycle. Therefore $N_{max} = \mu/C_{min}$. Let $C = (P-1)\cdot L + l$, since $C_{min} = (P-1)\cdot L + l + n$, then $N_{max} = \mu/C_{min} \leq \mu/C$. Therefore, it is possible to rewrite equation (9) as:

$$\Pr(col) \leq \frac{\frac{\mu}{C}\cdot\left(\frac{\mu}{L}+\frac{\mu}{C}\right)}{2^l} = \frac{\mu^2}{2^l}\cdot\frac{C+L}{C^2\cdot L} \tag{10}$$

Finally, according to (5), the IND-CPA advantage of adversary A against PSCFB can be expressed as:

$$ADV^{IND-CPA}_{PSCFB(F)}(A) \leq 2\cdot ADV^{PRF}_{F}(B) + \frac{\mu^2}{2^l}\cdot\frac{C+L}{C^2\cdot L} \tag{11}$$

Although block ciphers are analyzed as PRFs, their input and output size are equal, therefore if we consider that $L = l$, $C = P\cdot L$, then the advantage result is:

$$ADV^{IND-CPA}_{PSCFB(F)}(A) \leq 2\cdot ADV^{PRF}_{F}(B) + \frac{\mu^2}{L^2 2^L}\cdot\frac{P+1}{P^2} \tag{12}$$

Supposing that the term $ADV^{PRF}_{F}(B)$ is negligible, $ADV^{PRF}_{PSCFB}(A)$ will be negligible when the following condition is accomplished:

$$L^2 2^L \cdot \left(\frac{P^2}{P+1}\right) \gg \mu^2 \tag{13}$$

We can conclude that the PSCFB mode can be considered secure under certain conditions, in the same way that happens with CTR\$ mode. According to (12), the advantage of an adversary over PSCFB will be reduced if the size of the blackout period P is lengthened, which is directly related with the number of pipeline stages with which the block cipher has been implemented. It is a coherent result, because the longer the sync period the fewer random counter initializations will be produced, consequently reducing the probability of a collision and increasing the security. However, as proved in [30] longer P means worst values of SRD (Synchronization Recovery Delay) and EPF (Error Propagation Factor) in the PSCFB system. It is possible to conclude that there is a compromise between security and inherent properties of PSCFB measured by SRD and EPF.

TABLE 1
IND-CPA ADVANTAGE COMPARISON OF DIFFERENT MODES

Encryption Mode $\mathcal{SE}(F)$	IND-CPA Advantage expression[1] $ADV^{IND-CPA}_{\mathcal{SE}(F)}(A)$
PSCFB	$2\cdot ADV^{PRP}_{F}(B) + \frac{\mu^2}{L^2 2^L}\cdot\left(1+\frac{1}{P}+\frac{1}{P^2}\right)$
CTRC	$2\cdot ADV^{PRP}_{F}(B) + \frac{\mu^2}{L^2 2^L}$
CTR\$	$2\cdot ADV^{PRP}_{F}(B) + \frac{2\mu^2}{L^2 2^L}$
CBC	$2\cdot ADV^{PRP}_{F}(B) + \frac{2\mu^2}{L^2 2^L}$
CFB[2]	$2\cdot ADV^{PRF}_{F}(B) + \frac{\mu^2}{m^2 2^{L+1}}$

[1] *In each expression L is the block size and μ the number of encrypted bits.*
[2] *The term ADV^{PRF}_{F} refers to the prf-advantage where F_K is the function $select(E_K(\cdot)).E_K(\cdot)$ is the block cipher with blocksize L and $select(\cdot)$ is a function that outputs m fixed bits from its input.*

4.4 PSCFB vs CTR

As mentioned before, usually block ciphers are analyzed as PRFs, however it is well known that the PRPs (Pseudo Random Permutation) are what best models them. It is known that the prp-security and prf-security of a block cipher are related thanks to the PRF-PRP switching lemma [20]. Due to this, the overall advantage of the adversary A over the CTRC scheme is degraded by an amount given by the birthday attack, it is $\mu^2/L^2 2^{L+1}$, which means that its advantage can be written as:

$$ADV^{IND-CPA}_{CTRC(F)}(A) \leq 2\cdot ADV^{PRP}_{F}(B) + \frac{\mu^2}{L^2 2^L} \tag{14}$$

The same consideration can be taken for PSCFB, for which advantage can also be rewritten as:

$$ADV^{IND-CPA}_{PSCFB(F)}(A) \leq 2\cdot ADV^{PRP}_{F}(B) + \frac{\mu^2}{L^2 2^L} + \\ + \frac{\mu^2}{L^2 2^L}\cdot\left(\frac{1}{P}+\frac{1}{P^2}\right) \tag{15}$$

which means that:

$$ADV^{IND-CPA}_{PSCFB(F)}(A) \leq 2\cdot ADV^{PRP}_{F}(B) + \\ + \frac{\mu^2}{L^2 2^L}\cdot\left(1+\frac{1}{P}+\frac{1}{P^2}\right) \tag{16}$$

In Table 1, a comparison between the derived advantage of PSCFB and other operation modes is shown. As mentioned in [33] CTRC can be considered the best and most modern way to achieve privacy-only encryption. For this reason it is useful to make a comparison between the IND-CPA advantages between this mode and PSCFB.

Let us suppose that the underlying block ciphers for the two modes, CTRC and PSCFB, are good PRPs and have the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

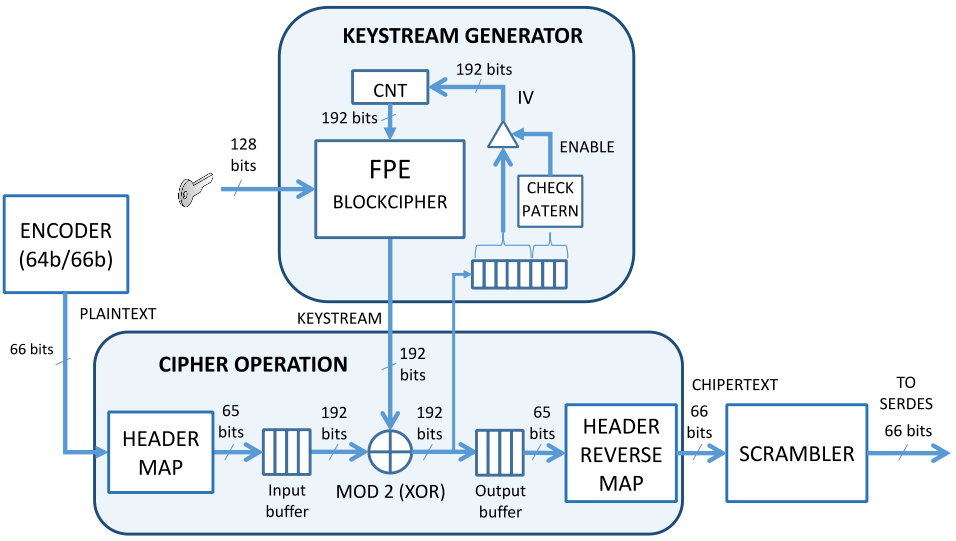


Fig. 6. Overall structure of the PSCFB system in the encryption module. In the decryption side the scheme is the same, however the check pattern input is taken directly from the input buffer output, before performing the XOR operation.

same prp-security, then we can stablish under what values of block size and pipelining the block cipher in PSCFB mode has at least the same security as that in CTRC when encrypting the same amount of information. For this purpose we should compare the second terms of (14) and (15) as follows:

$$\frac{\mu^2}{L_{PSCFB}^2 \cdot 2^{L_{PSCFB}}} \cdot \left(1 + \frac{1}{P} + \frac{1}{P^2}\right) \leq \frac{\mu^2}{L_{CTR}^2 \cdot 2^{L_{CTR}}} \qquad (17)$$

where L_{PSCFB} and L_{CTR} are the block sizes of the underlying block ciphers in PSCFB and CTRC modes respectively. According to (17), we can deduce the block size that a secure pipelined block cipher should have in case of being used in PSCFB mode if we want to provide the same security as another one with the same prp-security working in CTRC mode.

For example, let us assume that we have an AES (Advance Encryption Standard) block cipher working in CTRC mode, then $L_{CTR} = 128$ bits. We can stablish the value of L_{PSCFB} provided that:

$$\begin{aligned} L_{PSCFB}^2 \cdot 2^{L_{PSCFB}} &\geq L_{CTR}^2 \cdot 2^{L_{CTR}} \left(1 + \frac{1}{P} + \frac{1}{P^2}\right) \\ &= 2^{142} \cdot \left(1 + \frac{1}{P} + \frac{1}{P^2}\right) \end{aligned} \qquad (18)$$

As $P \geq 1$ the right term will be maximum with P=1, then

$$L_{PSCFB}^2 \cdot 2^{L_{PSCFB}} \geq 2^{142} \cdot 3 \qquad (19)$$

This inequality is fulfilled with $L_{PSCFB} \geq 130$ bits. For $P > 1$ the condition is fulfilled with $L_{PSCFB} > 128$.

Therefore, we can conclude that given two block ciphers with the same prp-security that work in two different modes, CTRC and PSCFB, if the block size of the one working in CTRC is 128 bits, then the other must have a block size larger than 128 to get the same or better IND-CPA security when encrypting the same amount of data.

In case of setting the same block size for both block ciphers, to get better IND-CPA security in PSCFB mode, less amount of data could be encrypted per key session.

5 Application Case: Ethernet 10GBase-R

In this paper we have focused on the case of high-speed communications, particularly in the 10GBase-R standard used in 10 Gigabit Ethernet optical links. In this standard, the PCS level is responsible for generating, encoding and scrambling the control and data blocks that are transmitted to the optical line. As block line coding is 64b/66b, the purpose of the encryption will be to cipher the complete 64b/66b block flow as shown in Fig. 2.

Because no extra data fields are added to the packets when they are encrypted, then no overhead is introduced and no throughput loss is produced. Moreover it not only encrypts the contents of the packets but also the activity or data traffic pattern. This is because by encrypting at 64b/66b block level, control blocks are also encrypted, such as packet start and end blocks or control blocks full of idle characters when no traffic is transmitted or during the IFGs (Inter Frame Gaps). Thanks to this last property, security could be improved, as passive eavesdroppers would be prevented from performing traffic analysis attacks. It would be useful in scenarios where traffic pattern analysis could reveal sensitive information about the behavior of a critical infrastructure or facility.

In this work the keystream generator of Fig. 2 has been implemented thanks to a block cipher working in PSCFB mode. As concluded in Section 4.4, to provide the same security level as a 128-bit block cipher working in CTRC mode when transmitting the same amount of data, it is necessary to use a block cipher with larger block size. In particular, for $P > 1$ this block size must be larger than 128

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

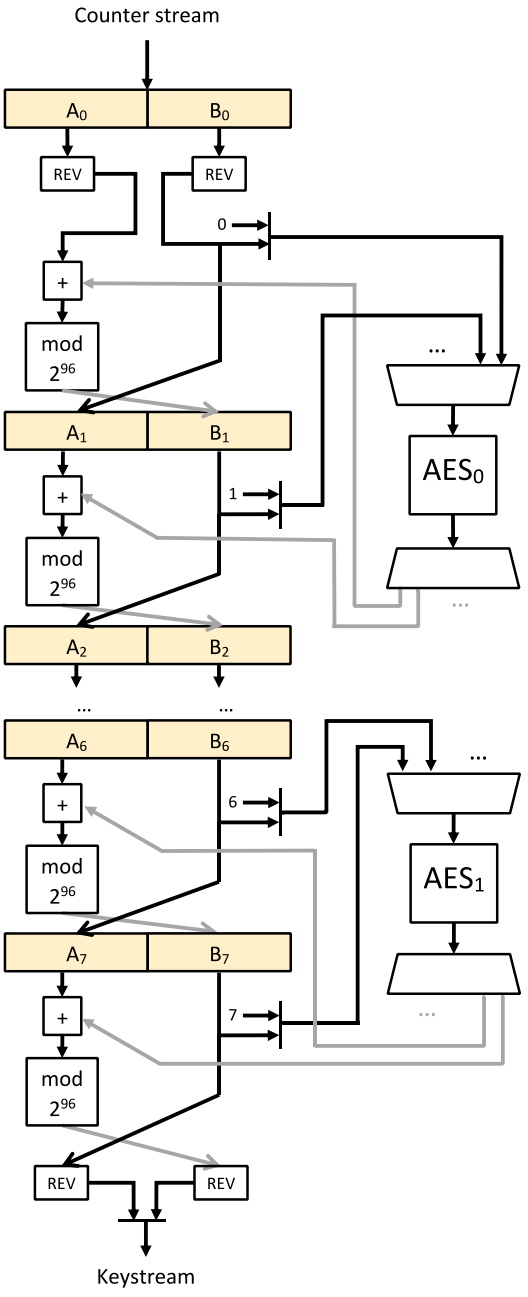


Fig. 7. Structure of the Feistel network for the 192-bit FPE cipher.

bits.

A possible block cipher candidate could be the well-known cipher Rijndael [34]. The main difference between Rijndael and AES is the range of configuration values for the block size and key length. Particularly, AES is a subset of Rijndael that uses a fixed block size of 128 bits and a key length of 128, 192 or 256 bits. On the contrary, the original specification of Rijndael also included 192 and 256 bits as possible block sizes. However, only the subset corresponding to the current version of AES was standardized by NIST, and as far as the authors know, there is no recommended block cipher with a block size greater than 128 bits.

Other solution for building a suitable and standardized block cipher with more than 128-bits block size is the use of the recent FPE (Format Preserving Encryption) modes approved by NIST [35]. FPE modes encrypt plaintext in a ciphertext preserving its original format and length. Typical applications of this modes are the encryption of PANs (Primary Account Numbers) or SSNs (Social Security Numbers) where a standard block cipher would not preserve their format.

Currently, two FPE modes are recommended by NIST, FF1 and FF3 [35]. Both modes are based on a non-binary Feistel structure, whose underlying round function consists of an AES block cipher. These are considered AES modes allowing to configure the block size and data radix of the resulting FPE block cipher.

Between the two NIST recommendations, we have selected FF3, as it is built with less rounds in its Feistel network and the cost in hardware resources is lower. In this mode the block size is limited according to the radix used, as shown in (20):

$$\begin{aligned} R &\in [2 \dots 2^{16}] \\ R^{minlen} &\geq 100 \\ 2 \leq minlen &\leq maxlen \leq 2\lfloor log_R(2^{96}) \rfloor \end{aligned} \tag{20}$$

where R is the radix and $minlen$ and $maxlen$ the bounds for the block size. With $R = 2$, the block size is between 2 and 192.

In this work the selected value for block size is the maximum, $L = 192$. With this size it is possible to get a major margin of cycles available per stage in a possible pipelined architecture. This fact allows a better reuse of the hardware resources.

According to this parameter, the final structure of the self-synchronized encryption system is shown in Fig. 6. It is similar to Fig. 2, but the keystream generator has been replaced with the final self-synchronous PSCFB structure based on AES in FPE mode. Also, in this figure it is shown that buffers are required at the input and output sides of the encryption structure. They are necessary to achieve 100% efficiency of the final encryption scheme.

6 System Implementation

6.1 PSCFB System Implementation

The underlying block cipher of the implemented PSCFB mode has been built using an FF3 structure. FF3 algorithm is described in [35]. In this work the cipher tweak value has been set to zero, while only the key is configurable. Taking into account the selected parameters for our FF3 implementation, $R = 2$ and $L = 192$, its structure is shown in Fig. 7 where one AES core is used each four stages of the Feistel network. The final latency introduced in its pipelined structure is 58 cycles.

Regarding to the system throughput, the efficiency η of an encryption scheme represents the amount of ciphered information that it can produce relative to the number of bits generated by its underlying block cipher. In the case of PSCFB, as the sync cycle length can be different to a multiple of the block size in bits, then not all blocks produced by the block cipher will be used completely for encryption. Indeed, the end of the blackout period could not be aligned

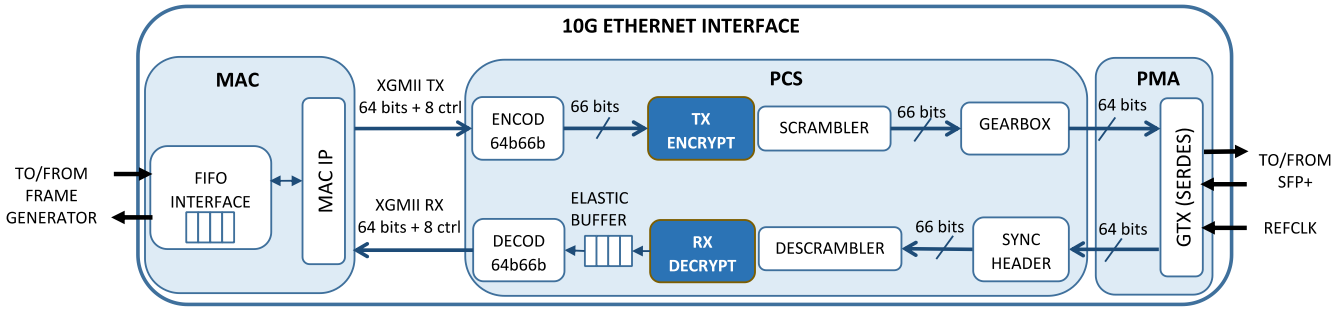


Fig. 8. Structure of the 10G Ethernet interface including the encryption function. It is composed by the MAC module, the PCS and the SERDES. In the PCS layer, TX_ENCRYPT and RX_DECRYPT are the encryption/decryption modules. Both include the CIPHER_OPERATION and KEYSTREAM_GENERATOR modules shown in Fig. 8.

TABLE 2
COMPARISON WITH OTHER SOLUTIONS

	FF1 [36]	FF3 [36]	FF3 [37]	This work
Slice Registers	11285	5592	11127	19154
Slice LUTs	7426	3587	16978	17599
18K Block RAMs[1]	343	170	77	153
Slices[2]	3268	1596	5636	6794
Operation Freq. (MHz)	279.6	283.5	125	217
Cycles/Encryption	707	269	1	1
Bytes/Encryption	13	13	1	192
Encryption Rate (Mbps)	41.1	109.6	1000	10000
Encryption Rate/Slice (Kbps/Slice)	12.57	68.7	177.4	1471.8

[1]*The 153 Block RAMs used in this work are due to the AES cores.*

[2]*Slices are estimated from the number of register and LUTs, assuming they are not packed together.*

with a kesytream block ending. It means that in the last block cipher operation of the blackout period the block cipher output could be used partially producing less ciphertext bits than the block size.

As explained in [30], it can be shown that the efficiency can be lower bounded by $\eta = P/(P+1)$, which means that for T bits produced by the block cipher output at least $T \cdot P/(P+1)$ ciphertext bits of PSCFB are generated. Therefore, to generate the ciphertext stream at the 100% throughput rate of a 10 Gbps Ethernet system, it is necessary to overclock the PSCFB subsystem with respect to the PCS sublayer. For this reason the two buffers shown in Fig. 6 have been introduced, to isolate the different clock domains of PSCFB and PCS.

On the other hand, these buffers also work as the input/output queues defined in the original specification of PSCFB, necessary to store information temporarily during periods of resynchronization, where partial block cipher outputs are used to encrypt data due to the fact that the sync pattern is not aligned with the end of the block cipher output.

Taking into account that the bus widths for PSCFB and PCS are 192 and 65 bits respectively, and equating the PCS 64b/66b bit throughput with that of the PSCFB, the following condition must be achieved:

$$T_{PSCFB} \cdot 192 = \eta \cdot T_{FPE} \cdot 192 = T_{PCS} \cdot 65 \qquad (21)$$

where T_{PSCFB} , T_{FPE} and T_{PCS} are the word throughputs of PSCFB, FPE and PCS domains, respectively, measured in words per second.

As in our FF3 implementation we are using two AES cores that have to attend to four stages each one, the total throughput of each AES core, T_{AES}, will be four times that of the FF3. Then (21) can be rewritten as:

$$\frac{P}{P+1} \cdot \frac{T_{AES}}{4} \cdot 192 = T_{PCS} \cdot 65 \qquad (22)$$

As P=58 and the PCS word rate is 156.25 MHz, the resulting clock frequency at which AES cores should work for getting a 100% throughput is 215.24 MHz. In this work the total frequency used for PSCFB system has been set to 217 MHz.

The described system in Fig. 6 has been implemented in a Xilinx Virtex 7 FPGA (Field Programmable Gate Array). In Table 2, the hardware resources of the PSCFB system are shown. It includes the resources used by the main modules in Fig. 6, KEYSTREAM_GENERATOR and CIPHER_OPERATION. Moreover, in Table 2, a comparison in terms of LUTs (Look-Up Tables), registers and BRAMs (Block RAMs) is made between this work and other implementations [36], [37]. Particularly, in [36] the FPGA used is a Virtex-6 device, different model than in this work, however the CLB structure in both devices is similar in terms of LUTs and registers, with four six-input LUTs and eight registers per slice. Although the implementation in this work entails more hardware resources, the ratio *Encryption_Rate/Slice* is clearly superior.

In general, although the inherent structure of an AES in FPE mode consumes more resources than the AES core, authors have considered this option to grant that the security of the PSCFB system is not degraded in respect a typical

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

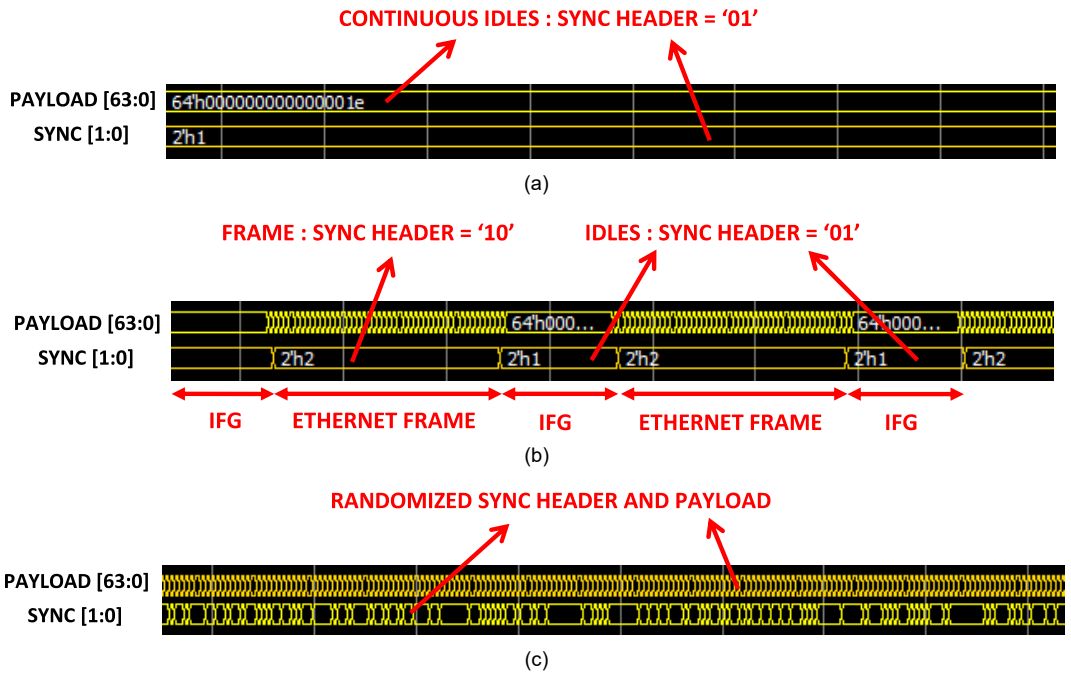


Fig. 10. (a) SYNC header pattern without encryption when no Ethernet frame is transmitted; (b) SYNC header pattern without encryption when transmitting an Ethernet frame burst; (c) SYNC header pattern after encryption regardless of the transmission or non-transmission of Ethernet frames.

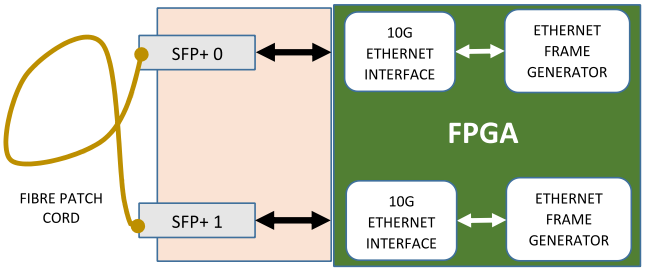


Fig. 9. Test setup scheme with SFP+ modules working at 10Gbps rate.

CTRC implementation of AES. On the other hand, although implementations of a 192-bit block cipher such as Rijndael could have been possible, they are not recommended by NIST, as the FPE option is.

6.2 Test Setup

To carry out the test of the system, the proposed encryption scheme has been integrated in a 10GBase-R Ethernet interface. In Fig. 8, the complete PCS structure is shown. Apart from the 64b/66b encoding and scrambling functions it contains the encryption and decryption modules.

Moreover, a traffic generator has been also implemented and linked to the Ethernet interface to test it with real data frames. Two chains composed each one by the 10GBase-R Ethernet interface and a frame generator have been implemented over the Xilinx Virtex 7 FPGA. Both Ethernet interfaces have been connected to two SFP+ (Small Form-factor Pluggable) modules configured to work at 10 Gbps speed and linked between them with a multimode fiber patch cord. Some of the parameters of the transmitter/receiver and the fiber link are shown in Table 3.

TABLE 3
OPTICAL LINK PARAMETERS

	Parameter	Value
Transmitter	Average Launch Power (dBm)	-1
	Optical Wavelength (nm)	850
	RMS Spectral Width (dB)	0.45
	Optical Extinction Ratio (dB)	5.5
Receiver	Receiver Sensitivity (dBm)	-11.1
	RX Wavelength Range (nm)	840-860
Fiber Link	Link Length (m)	1
	Fiber diameter - core/cladding (nm/µm)	62.5/125
	Modal Bandwidth (MHz x km)	200
	Attenuation (dB/km)	3

A scheme of the setup for test is shown in Fig. 9. Thanks to the Ethernet frame generators, it has been possible to check the encrypted link with real data packets without producing any frame loss or CRC (Cyclic Redundancy Check) errors.

The extra hardware resources introduced in the PCS datapath generate an extra latency of 266 ns for both transmission and reception. It is noteworthy that the proposed encryption system gets values comparable to those achieved by OTN equipment [12]. Moreover, no overhead is introduced in the encryption process allowing data be encrypted at line rate.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

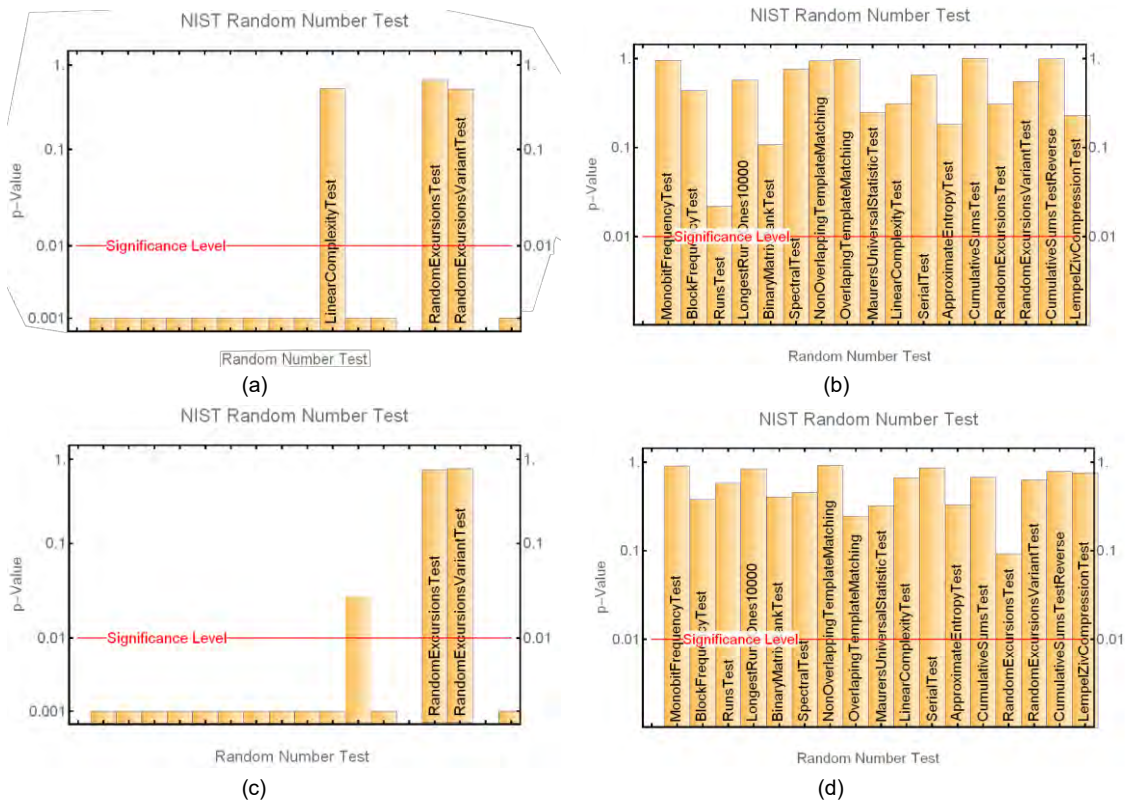


Fig. 11. NIST test results for the 64b/66b block payload of D pattern, (a) before encryption and (b) after it. NIST test results for the 64b/66b block payload of C pattern, (c) before encryption and (d) after it.

This is a clear advantage over other encryption mechanisms whose inherent overhead limits the total throughput to values lower than 100% [16].

7 Encryption Results

7.1 Encryption Properties

The encrypted link has been tested with different Ethernet traffic flows. Thanks to the Ethernet frame generators four different traffic patterns have been encrypted. These have been named A, B, C and D. Pattern A corresponds with the case of no frame transmission, where only 64b/66b control blocks full of idle characters are transmitted over the link. Patterns B, C and D correspond to continuous frame transmission of 1024-bytes length at rates of 10.2%, 50% and 98% of the maximum 10 Gigabit line rate, respectively, and with random payloads.

Two conclusions arise from simulation and hardware debugging. On the one hand, encryption and decryption work correctly and synchronously without harming data traffic or link establishment between 10G Ethernet interfaces. No CRC errors are produced when transmitting the mentioned traffic patterns. On the other hand encrypted traffic patterns are masked, which can improve the overall security against passive eavesdroppers.

Regarding this capability, it is interesting to monitor signal waveforms after the encryption module, at the input of the scrambler. The 64b/66b data bus is formed by the 2-bit sync header and the 64-bit block payload. If encryption is disabled, the synchronization header takes the value '10' when frames are transmitted and '01' during the IFG between frames. In t-he case of pattern A, as no frames are transmitted, sync header is always equal to '01'. However when encryption is enabled, the sync header takes random values between '10' and '01'. Also, the 64-bit block payload is randomized. In this way the traffic pattern is indistinguishable, independently of whether frames are being transmitted or not. In Fig. 10 waveforms of an encrypted pattern are shown.

In order to check this masking property, the randomness test suite proposed by NIST [38] has been used to evaluate encrypted patterns. Also SE (Shannon Entropy) has been measured and compared among the mentioned non-encrypted patterns and the encrypted signal.

Owing to the limited memory in FPGA hardware resources, these tests have been performed at simulation stage, but this fact does not invalidate experimental results.

Regarding the NIST tests, sync header and block payload have been evaluated separately with these tests concluding that both can be considered random sequences after encrypting any of the mentioned patterns. As an example, results for NIST test applied to the patterns D and C before and after encryption are shown in Fig. 11. It is possible to conclude that although they are very different patterns, after encryption they are transformed into a sequence that passes these tests, which makes them indistinguishable from a random stream.

As for SE measurement, it has also been calculated for

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2018.2890259, IEEE Transactions on Computers

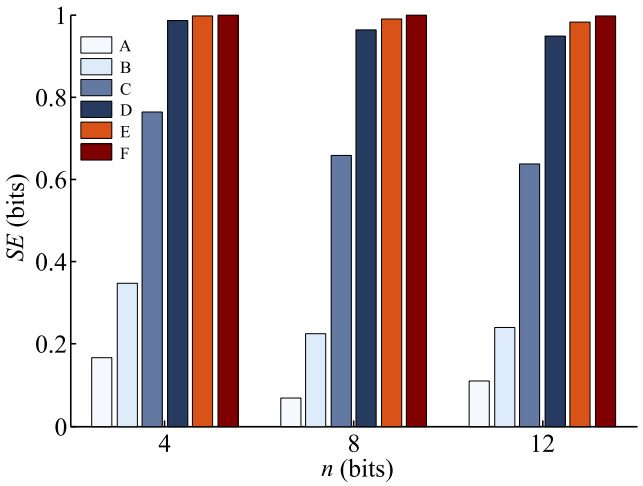


Fig. 12. Shannon Entropy of block payloads measured with n equal to 4, 8 and 12 in Ethernet traffic patterns from A to F.

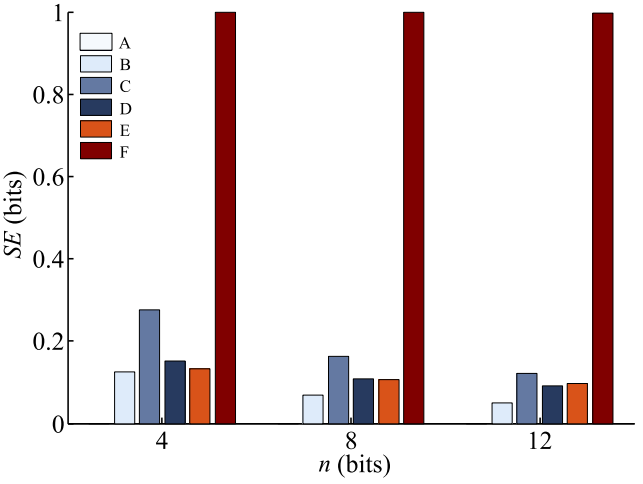


Fig. 13. Shannon Entropy of mapped sync header measured with n equal to 4, 8 and 12 in Ethernet traffic patterns from A to F. In the case of A pattern, sync header has a continuous value, which means that its entropy is zero. Because of this, A bar is zero in this figure.

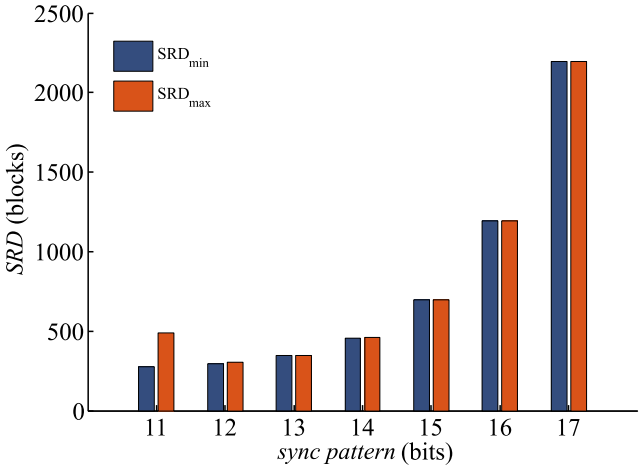


Fig. 14. Theoretical maximum and minimum SRD values for different sync pattern sizes.

the mapped sync header and block payload separately as defined in (23).

$$SE = -\frac{1}{n} \cdot \sum_{\beta_n \epsilon 2^n} P(\beta_n) \cdot \log_2 P(\beta_n) \qquad (23)$$

In both cases the bit stream has been grouped in tuples of n bits called β_n and the probability for each tuple, $P(\beta_n)$, has been calculated. SE has been measured for values of n equal to 4, 8 and 12 bits in each of the mentioned non-encrypted patterns A, B, C and D, all of them with fixed length frame and random payloads. In addition two more patterns have been added: E and F. Pattern E corresponds to continuous frame transmission with random payloads and random length between 64 and 1516 bytes while pattern F corresponds to the randomized signal after encryption of pattern A, which can be considered the worst case in terms of entropy. In Figs. 12 and 13 the comparison among the measured SE is shown. It is possible to notice that SE in pattern F is as expected, almost the ideal value of 1 in both cases, block payload and sync header, as it is encrypted. In the rest of non-encrypted patterns SE of block payloads decreases as the transmission rate decreases from E to A. This effect is owing to the ratio of the bandwidth that is used by the IFGs (Inter Frame Gaps). As when lower bandwidth is used, the IFGs full of idle sets takes more bandwidth percentage versus the random payloads of the transmitted frames, resulting in a lower SE. In the case of sync header entropy, clearly all non-encrypted patterns achieve a very low value.

Thank to this result it is possible to conclude that encryption makes indistinguishable data traffic pattern, as the maximum value for SE is obtained and NIST tests are passed successfully when traffic is encrypted.

7.2 Self-synchronization

The SRD (Synchronization Recovery Delay) is the metric used to examine the resynchronization properties of SCFB and PSCFB modes [27]. It is defined as the expected number of bits following a sync loss before synchronization is reestablished. According to [30], upper and lower bounds for SRD depend on the block size L, number of pipelines P and size of the synchronization pattern n. Taking into account that in this system $L=192$ and $P=58$, upper and lower SRD bounds calculated with different values of n are shown in Fig. 14. As for n below 11 the upper bound grows, those values have not been shown. In this work $n=12$ has been used, limiting the upper bound of SRD to 304 encrypted 64b/66b blocks.

Experimental results and simulation show that self-synchronization works correctly. By removing the optical fiber with encryption activated between Ethernet interfaces, both link status and encryption synchronization are lost. However, when restoring the optical fiber encryption synchronization is always recovered after PCS link status is achieved. Traffic bursts were correctly tested after synchronization recovering to check the integrity of the link.

8 Other Security Considerations

Although in this paper physical layer encryption mechanism is proposed and developed, other security considerations must be taken into account, especially regarding the key configuration and refreshing. One possible solution could be the use of classical public key encryption schemes, where the interchange of the symmetric master key is performed thanks to algorithms such as RSA used by protocols at higher communication layers. As an example, in the layer 2 encryption protocol MACsec, the master key negotiation is outside of its specification and is carried out by IEEE 802.1X standard. Therefore, the same idea could be considered for this work.

Other possible solutions for key distribution are optical mechanisms such as QKD (Quantum Key Distribution) that is based on the laws of physics rather than classic asymmetric cryptography algorithms. In spite of the crosstalk that classical signals can introduce to QKD channels, successful experiments where QKD channels coexist with classical data traffic are a reality [39], [40]. Thanks to WDM (Wavelength-Division-Multiplexing) techniques it is possible to reduce the crosstalk noise to a tolerable level, which means that QKD could also be considered for its use with the encryption mechanism presented in this work.

9 Conclusion

To the authors best knowledge, this is the first time that a self-synchronous encryption method is proposed for ciphering physical layer communications based on 64b/66b encoding. The new encryption system consists of a self-synchronous symmetric ciphering of the complete 64b/66b block stream. The encryption is based on the PSCFB mode, and it has been simulated and implemented over an FPGA. Also security considerations for this mode have been taken into account, deriving a formal security expression similar to that known for other operation modes.

This new mechanism is able to perform encryption at a line-rate introducing a latency in the range of nanoseconds, while the complete data traffic pattern is masked, improving the overall security.

Although this mechanism is proposed for 10Gbps Ethernet links, 64b/66b encoding is used in other standards at higher rates, as 100 Gigabit Ethernet. It means that the same encryption scheme could be applied not only to the access networks but also to long-haul optical links in transport networks.

In addition to this, by preserving coding properties such as short run length and transition density, physical layer encryption is achieved without making changes in the subsequent circuitry. For example, commercial SFP+ modules or SERDES at 10 Gbps rate are compatible with the proposed encryption scheme.

Acknowledgment

This work has been supported in part by the MINECO-FEDER under Grant TEC2014-52840-R and Grant TEC2017-85867-R and in part by the FPU fellowship program to M. García-Bosque under Grant FPU14/03523.

References

[1] Cisco Systems, "Cisco Global Cloud Index: Forecast and Methodology. 2015-2020," 2016.

[2] Cisco Systems, "Cisco 2018 Annual Cybersecurity Report," Feb. 2018.

[3] "IEEE Standard for Local and Metropolitan Area Networks: Media Access Control (MAC) Security, IEEE Std 802.1AE, 2006".

[4] S. Kent and K. Seo, "RFC 4301: Security Architecture for the Internet Protocol," 2005. [Online]. Available: https://tools.ietf.org/html/rfc4301.

[5] N. Skorin-Kapov, M. Furdek, S. Zsigmond and L. Wosinska, "Physical-Layer security in evolving optical networks," *IEEE Commun Mag.*, vol. 54, no. 8, pp. 110-117, Aug. 2016.

[6] M. Furdek, N. Skorin-Kapov, S. Zsigmond and L. Wosinska, "Vulnerabilities and Security Issues in Optical Networks," in *International Conference on Transparent Optical Networks (ITCON)*, Graz, Austria, 2014.

[7] M. Zafar, H. Fathallah and N. Belhadj, "Optical fiber tapping: Methods and precautions," in *High Capacity Optical Networks and Enabling Technologies (HONET)*, Rihad, 2011.

[8] M. P. Fok, Z. Wang, Y. Deng and P. R. Prucnal, "Optical Layer Security in Fiber-Optic Networks," *IEEE Trans. Inf. Forensics Security,* vol. 6, no. 3, pp. 725-736, 2011.

[9] J. Ji, G. Zhang, W. Li, L. Sun, K. Wang and M. Xu, "Performance Analysis of Physical-Layer Security in an OCDMA-Based Wiretap Channel," *J. Opt. Commun. Netw.*, vol. 9, no. 10, pp. 813-818, 2017.

[10] J. Hizanidis, S. Deligiannidis, A. Bogris and D. Syvridis, "Enhancement of Chaos Encryption Potential by Combining All-Optical and Electrooptical Chaos Generators," *IEEE Journal of Quantum Electronics,* vol. 46, no. 11, pp. 1642-, Nov. 2010.

[11] D. Elkouss, J. Martinez-Mateo, A. Ciurana and V. Martin, "Secure Optical Networks Based on Quantum Key Distribution and Weakly Trusted Repeaters," *J. Opt. Commun. Netw,* vol. 5, no. 4, pp. 316-328, 2013.

[12] K. Guan, J. Kakande and J. Cho, "On Deploying Encryption Solutions to Provide Secure Transport-as-a-Service (TaaS) in Core and Metro Networks," in *European Conference and Exhibition on Optical Communications*, Düsseldorf, September 18 – 22, 2016.

[13] MicroSemi, "In-flight Encryption in Service Provider Networks, No: PMC-2150716, Issue 2," 2016.

[14] S. Salehi, Y. Rico Cao and H. Chen, "Bandwidth-IPSec security trade-off in IPv4 and IPv6 in Windows 7 environment," in *International Conference on Future Generation Communication Technology (FGCT)*, London, UK, Nov. 2013.

[15] C. Xenakis, N. Laoutaris, L. Merakos and I. Stavrakakis, "A generic characterization of the overheads imposed by IPsec and associated cryptographic algorithms," *Computer Networks,* no. 50, pp. 3225-3241, 2006.

[16] L. Troell, J. Burns, K. Chapman, D. Goddard, M. Soderlund and C. Ward, "Converged vs. Dedicated IPSec Encryption Testing in Gigabit Ethernet Networks, Technical Report," 2005. [Online]. Available: http://scholarworks.rit.edu/article/1743.

[17] S. Salehi, Y. Rico Cao and H. Chen, "Bandwidth-IPSec security trade-off in IPv4 and IPv6 in Windows 7 environment," in *International Conference on Future Generation Communication Technology (FGCT)*, London, UK, Nov. 2013.

[18] A. Pérez-Resa, M. Garcia-Bosque, C. Sanchez-Azqueta y S. Celma, «Chaotic Encryption for 10-Gb Ethernet Optical Links,» *IEEE Tansactions On Circuits and Systems–I: Regular Papers,* p. doi: 10.1109/TCSI.2018.2867918, 2018.

[19] A. Klein, Stream Ciphers, London: Springer-Verlag, 2013.

[20] M. Bellware, A. Desai, E. Jokipii and P. Rogaway, "A concrete security treatement of symmetric encryption: Analysis of the DES modes of operation," in *Symposium on Foundations of Computer Science (FOCS)*, 1997.

[21] M. Robshaw and O. Billet, New Stream Cipher Designs: The eSTREAM Finalists, Springer, 2008.

[22] J. L. J. P. B. Daemen, "Chosen ciphertext attack on SSS," 2005. [Online]. Available: http://www.ecrypt.eu.org/stream/papersdir/044.pdf (p. 235).

[23] A. Joux and F. Muller, "Chosen-ciphertext attacks against MOSQUITO," in *Robshaw, M. (ed.) Fast Software Encryption*, Berlin, Springer, 2006, pp. 390-404.

[24] M. Dworkin, Recommendation for Block Cipher Modes of Operation: Methods and Techniques, Gaithersburg, Maryland: NIST Special Publication 800-38G, 2001.

[25] O. Jung and C. Ruland, "Encryption with Statistical Self-Synchronization in Synchronous Broadband Networks," in *Proc. Conf. Cryptographic Hardware and Embedded Systems (CHES '99)*, 1999.

[26] A. Alkassar, A. Geraldy, B. Pfitzmann and A.-R. Sadeghi, "Optimized Self-Synchronizing Mode of Operation," in *Proc. Conf. Fast Software Encryption (FSE '01)*, Apr. 2001.

[27] H. M. Heys, "Analysis of the Statistical Cipher Feedback Mode of Block Ciphers," *IEEE Transactions on Computers,* vol. 52, no. 1, pp. 77-92, 2003.

[28] H. M. Heys, "An Analysis of the Statistical Self-Synchronization of Stream Ciphers," in *Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society*, Anchorage, USA, 2001.

[29] F. Yang and H. M. Heys, "Comparison of Two Self-Synchronizing Cipher Modes," in *Queen's 22nd Biennial Symposium on Communications*, 2004.

[30] H. M. Heys and L. Zhang, "Pipelined Statistical Cipher Feedback: A New Mode for High-Speed Self-Synchronizing Stream Encryption," *IEEE Transactions on Computers,* vol. 60, no. 11, pp. 1581-1595, 2011.

[31] M. Bellware and P. Rogaway, "Introduction to modern cryptography," May 2005. [Online]. Available: http://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf.

[32] M. Bellware and P. Rogaway, "Chapter 5.7 Security of CTR Modes," in *Introduction to modern cryptography*, May 2005.

[33] P. Rogaway, "Evaluation of some blockcipher modes of operation.," in *Technical report, Cryptography Research and Evaluation Committees (CRYPTREC) for the Goverment of Japan*, Feb. 2011.

[34] J. Daemen y V. Rijmen, The Design of Rijndael, AES - The Advanced Encryption Standard, Berlin Heidelberg New York: Springer-Verlag , 2002.

[35] M. Dworkin, Recommendation for Block Cipher Modes of Operation: Methods for Format-Preserving Encryption, Gaithersburg, Maryland: NIST Special Publication 800-38G, 2016.

[36] R. Agbeyibor, J. Butts, M. Grimaila and R. Mills, "Evaluation of format preserving encryption algorithms for critical infrastructure protection," in *Critical Infrastructure Protection VIII*, Springer, 2014, pp. 245-261.

[37] A. Pérez-Resa, M. Garcia-Bosque, C. Sánchez-Azqueta and S. Celma, "Physical Layer Encryption for Industrial Ethernet in Gigabit Optical Links," *IEEE Transactions on Industrial Electronics,* June 2018.

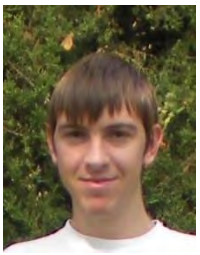
[38] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks and A. Heckert, A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications, NIST—Information Technology: Gaithersburg, MD, USA: NIST Special Publication 800-22 Rev.1a, 2010.

[39] Y. e. a. Mao, «Integrating quantum key distribution with classical communications in backbone fiber network,» *Optics Express,* vol. 26, nº 5, 2018.

[40] A. e. a. Slavisa, «Perspectives and limitations of QKD integration in metropolitan area networks,» *Optical Express,* vol. 23, nº 8, pp. 10359-10373, 2015.

Adrián Pérez-Resa was born in San Sebastián, Spain. He received the M.Sc degree in telecommunications engineering from the University of Zaragoza, Zaragoza, Spain, in 2005. Currently he is working toward the Ph.D degree from the Group of Electronic Design, Aragón Institute of Engineering Research, University of Zaragoza.

He was an R&D Engineer with the Telecommunications Industry for more than ten years. He is currently a member of the Group of Electronic Design, Aragón Institute of Engineering Research, University of Zaragoza. His research interests include high speed communications and cryptography applications.



Miguel Garcia-Bosque was born in Zaragoza, Spain. He received the B.Sc and M.Sc in Physics from the University of Zaragoza, Zaragoza, Spain in 2014 and 2015 respectively.
He is currently a member of the Group of Electronic Design, Aragón Institute of Engineering Research, University of Zaragoza. His research interests include chaos theory and cryptography algorithms.

Carlos Sánchez-Azqueta was born in Zaragoza, Spain. He received the B.Sc., M.Sc., and Ph.D. degrees from the University of Zaragoza, Zaragoza, Spain, in 2006, 2010, and 2012, respectively, all in Physics, and the Dipl.-Ing. Degree in electronic engineering from the Complutense University of Madrid, Madrid, Spain in 2009.
He is currently a member of the Group of Electronic Design, Aragón Institute of Engineering Research, University of Zaragoza. His research interests include mixed-signal integrated circuits, high-frequency analog communications, and cryptography applications.

Santiago Celma was born in Zaragoza, Spain. He received the B.Sc., M.Sc., and Ph.D. degrees in physics from the University of Zaragoza, Zaragoza, Spain, in 1987, 1989, and 1993, respectively.
He is currently a Full Professor in the Group of Electronic Design, Aragon Institute of Engineering Research, University of Zaragoza. He has coauthored more than 100 technical papers and 300 international conference contributions. He is coauthor of four technical books and the holder of four patents. He appears as principal investigator in more than 30 national and international research projects. His research interests include circuit theory, mixed-signal integrated circuits, high-frequency communication circuits, wireless sensor networks and cryptography for secure communications.