IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 23, 2020, accepted April 29, 2020, date of publication May 11, 2020, date of current version May 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2993550

Self-Synchronized Encryption for Physical Layer in Gigabit Ethernet Optical Links

ADRIÁN PÉREZ-RESA, MIGUEL GARCIA-BOSQUE, CARLOS SÁNCHEZ-AZQUETA, AND SANTIAGO CELMA

Electronic and Communications Engineering Department, University of Zaragoza, 50009 Zaragoza, Spain

Corresponding author: A. Pérez-Resa (aprz@unizar.es)

This work was supported by the MINECO-FEDER under Grant TEC2017-85867-R.

ABSTRACT In this work a new self-synchronized symmetric encryption solution for high speed communication systems necessary to preserve the format of the plaintext is proposed, developed and tested. This new encryption mechanism is based on the block cipher operation mode called PSCFB (Pipelined Statistical Cipher Feedback) and the modulo operation. The confidentiality of this mode is analyzed in terms of its IND-CPA (Indistinguishability under Chosen-Plaintext Attack) advantage, concluding that it can be considered secure in the same way as traditional modes are. The encryption system has been integrated in the physical layer of a 1000Base-X Gigabit Ethernet Interface, where the 8b/10b symbol flow is encrypted at line rate. Moreover, an implementation of the proposed system has been carried out in an FPGA (Field Programmable Gate Array) device. Finally, an encrypted optical link has been tested with real Ethernet frames, getting maximum throughput and protecting the data traffic from passive eavesdroppers.

INDEX TERMS Gigabit Ethernet, physical coding sublayer, encryption, stream cipher, PSCFB (pipeline statistic cipher feedback), FPE (format preserving encryption).

I. INTRODUCTION

In recent decades, we have witnessed the rise of broadband networks, mainly thanks to the advance of communication standards in physical media such as optical fiber. Thanks to them it is possible to provide the high data bandwidth demanded by the customers and the market [1]. To achieve information confidentiality maintaining the information throughput, high speed encryption systems must be used.

Encryption methods can be implemented at different levels of a communication system, for example MACsec [2] or IPsec [3], for layer 2 and layer 3, respectively. Regarding to physical level (layer 1), different solutions have been proposed depending on the transmission medium, such as [4], [5] or [6]. In the particular case of optical networks, physical layer it is considered critical to guarantee secure communications [7], [8], [9].

There are several encryption mechanisms for optical networks at physical layer. Some of them are focused on photonics technologies [9], [10], [11], while others are based on the protocols of the optical system such as in [12], where OTN (Optical Transport Network) frame payloads are encrypted at bit level. Other examples have been shown for 1 Gbps and 10 Gbps optical Ethernet standards, where several proposals have been developed for 64b/66b [13] and 8b/10b [14] physical encodings. For the particular case of Gigabit Ethernet, the encryption is performed in the PCS (Physical Coding Sublayer) layer before the 8b/10b encoder. In this way it is possible to preserve the coding properties such as the transition density and short run length, that are necessary to facilitate the operation of the remote CDR (Clock and Data Recovery) circuits [15] at the physical layer. In addition, to preserve the coding properties it is necessary to perform a kind of encryption able to preserve the format of the plaintext, in this case the 8b/10b symbol flow.

This kind of encryption is usually called FPE (Format Preserving Encryption), and although many solutions have been proposed about it [16], the only ones approved by the NIST (National Institute of Standards and Technology) are the FF1 and FF3 modes of operation [17].

As far as the authors are concerned, there are no standardized solutions for FPE self-synchronized stream ciphers, but some proposals have been made. For example in [18], where the self-synchronized PSCFB operation mode was proposed to be used with the underlying FPE block cipher in [14]. An important issue regarded with symmetric encryption stream ciphers is that these must synchronize their keystreams before starting the encryption session. Furthermore, in case of missing the synchronization in the middle of a session,

The associate editor coordinating the review of this manuscript and approving it for publication was Mostafa M. Fouda.

an ad-hoc mechanism or protocol must be implemented to recover it. With the use of self-synchronized stream ciphers it is possible to dispense with these extra communication processes.

Unfortunately, self-synchronized solution in [18] uses internally the recommended FF3 scheme that is based on a non-binary Feistel structure, which is formed by several processing stages where an AES (Advanced Encryption Standard) block is required in each one of them. This scheme increases the hardware complexity of the resulting FPE self-synchronized solution with respect to other non-FPE ciphers.

On the other hand, the PSCFB mode has an encryption efficiency that can be inherently less than 100% [19]. If E_K is an underlying block cipher working in PSCFB mode, the encryption efficiency represents the number of ciphertext bits that can be produced in PSCFB mode relative to the number of output bits produced by E_K working in a basic CTR (Counter) mode. As consequence of having an efficiency lower than 100% it is necessary to use input and output queues in the PSCFB structure, which increases the latency introduced by the encryption system [19].

In order to reduce the hardware complexity of possible self-synchronous FPE stream ciphers, in this work a new structure able to preserve the format of the plaintext is proposed. It is based on a recommended block cipher working in an operation mode that is a synergic combination of PSCFB and CTR-MOD [20] modes. We have called it PSCFB-MOD.

Since this new proposed operation mode uses as underlying block cipher a recommended binary block cipher instead of an FPE one, as in [14] and [18], it is possible to reduce the hardware complexity introduced by the Feistel structure. In addition, with this new encryption scheme the input and output queues are not needed, which reduces the latency introduced by the encryption system.

By using a recommended underlying block cipher, for instance AES, it is possible to develop a formal security proof, in the same way as in traditional confidentiality-only operation modes, such as CTR or CBC (Cipher Block Chaining). The formal security proof consists of the IND-CPA (Indistinguishability under Chosen Plaintext Attack) advantage expression of any adversary attacking this scheme.

Although PSCFB mode has been previously implemented in Ethernet communications, as in [13], it is important to remark that it has been done for 10 Gigabit Ethernet links using the 64b/66b encoding, different to 8b/10b, which is used in this work. Therefore, the application of PSCFB-MOD in this work entails an architecture different to the used in [13], that is able to preserve the format of gigabit Ethernet symbols and needs a different IND-CPA analysis.

The paper is divided in five sections. In Section II an introduction about Gigabit Ethernet standard, CTR-MOD mode and self-synchronized encryption is given. Section III presents the security analysis and the hardware implementation of the proposed PSCFB-MOD scheme and their comparison with the traditional CTR. Subsequently, Section IV deals with the set-up and encryption results. Finally, in Section V conclusions are given.

II. IMPORTANT CONCEPTS

A. OPTICAL GIGABIT ETHERNET ENCRYPTION

Although there are no standardized solutions for FPE stream ciphers, its usage could be relevant in communications where a high encryption rate is necessary, as in physical layer 1000Base-X for optical Gigabit Ethernet systems [14].

In [14] encryption is carried out in the PCS sublayer where the 8b/10b encoding is performed. Since the 8b/10b encoding is used to provide important properties to the bitstream, it is necessary to preserve the format of the 8b/10b symbols. Therefore, the encryption of a gigabit Ethernet symbol must give as result another valid symbol that must be within the group of symbols supported by the standard, which means to perform an FPE encryption.

Encryption and decryption modules are located in the physical layer datapath as shown in Fig. 1, before the 8b/10b encoder/decoder. In this figure an Ethernet Interface is shown. It is composed of the MAC (Medium Access Control) and PHY layers. PHY module includes PCS, PMA (Physical Medium Attachment) and PMD (Physical Medium Dependant) sublayers.

Inside the ENCRYPT module, FPE encryption is performed as shown in Fig. 2. If S is the possible number of different 8b/10b symbols, using the MAP block, each symbol is mapped to an integer value in the range $\{0, \ldots, S-1\}$ to get a plaintext in radix S. The plaintext is added modulo-S to a keystream, which is also in radix S, to obtain the ciphertext.

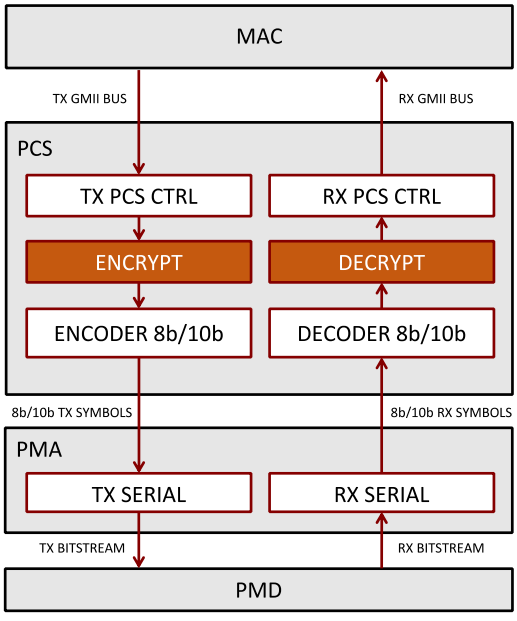

FIGURE 1. Scheme of the Ethernet Interface formed by the PHY and MAC modules. MAC layer builds the Ethernet packets transmitted to the PHY. ENCRYPT and DECRYPT modules perform the format preserving encryption/decryption of 8b/10b symbols at the PCS sublayer. TX_SERIAL and RX_SERIAL are the serializer/deserializer modules that transmit and receive the bitstream from the optical link.

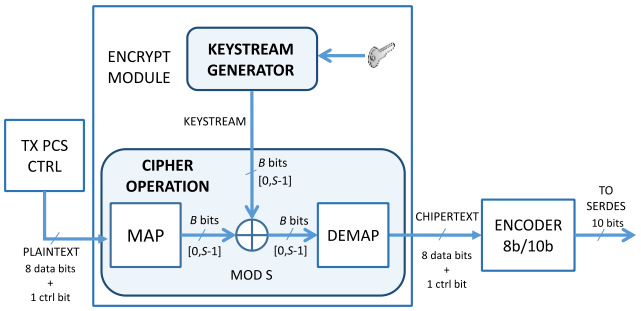

FIGURE 2. Location and generic structure of a stream cipher in a physical layer with 8b/10b line encoding. 8b/10b symbols are formed by eight data bits and one control bit. The mapped symbols are represented with $B = \lceil log_2 S \rceil$ bits.

Finally This ciphertext is reverse mapped in the DEMAP module to get the final 8b/10b encrypted flow. The ciphered 8b/10b symbols will be encoded to 10-bit values and sent to the serializer.

B. CTR-MOD OPERATION MODE

CTR-MOD mode can be considered a kind-of counter mode where the counter is in binary radix with a length of $l_{CTR-MOD}$ bits while the keystream output is a stream of symbols in radix S, which is generated thanks a function called F_K_MOD. The ciphertext is obtained thanks to the modulo-S addition between the keystream and the plaintext. This mode next to the traditional CTR are shown Fig. 3a and Fig. 3b, respectively.

F_K_MOD can be considered a PRF (Pseudo Random Function) such that F_K_MOD: $\mathcal{K} \times \{0,1\}^{l_{CTR-MOD}} \rightarrow \{0,\ldots,S-1\}$. It is configured with a random key K taken from the keyspace $\mathcal{K}$ ($K \epsilon \mathcal{K}$). F_K_MOD is built thanks to a block cipher E_K as shown in Fig. 3a. In this figure the block cipher is modelled as a PRF such that E: $\mathcal{K} \times \{0,1\}^{l_{CTR-MOD}} \rightarrow \{0,1\}^{l_{CTR-MOD}}$, where $l_{CTR-MOD}$ (the counter length in bits) is also its block size. Its least significant L output bits are taken as input of a modulo-S operation, which is the same as performing the modulo-2^L operation at the output of E_K. Finally we can represent $F_K\_MOD(x)$ in terms of E_K, such that $F_K\_MOD(x) = (E_K(x) \, mod 2^L) mod S$.

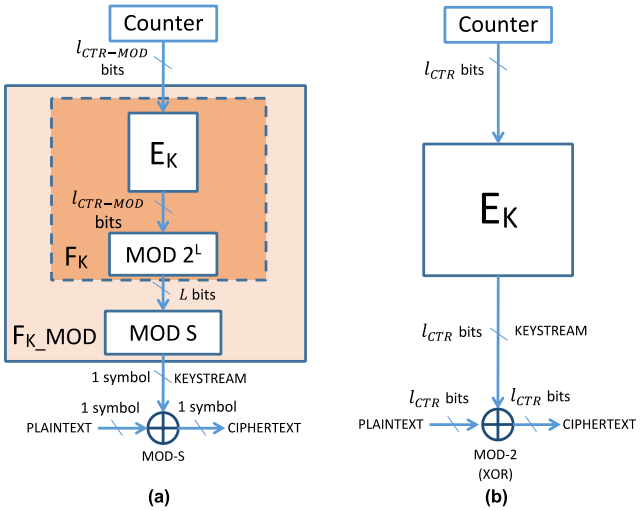

FIGURE 3. Structure of (a) CTR-MOD mode and (b) CTR traditional mode.

C. SELF-SYNCHRONIZED ENCRYPTION

Several mechanisms have been proposed for self-synchronized encryption. On the one hand, there are some proposals based on ad-hoc self-synchronized stream ciphers such as those proposed in eSTREAM project: SSS and Mosquito [21]. However, they were dismissed due to their vulnerabilities [22], [23]. On the other hand self-synchronized stream ciphers can be built thanks to block ciphers working in certain operation modes such as CFB (Cipher Feedback), OCFB (Optimized Cipher Feedback) [24], SCFB (Statistical Cipher Feedback) [25] or PSCFB (Pipeline Statistical Cipher Feedback) [19]. Among them, PSCFB can be considered the best in terms of encryption throughput, as it can reach a value near to 100%.

The PSCFB operation mode is shown in Fig. 4. It is considered a combination of two NIST recommended operation modes, CTR and CFB. The underlying block cipher E_K has a block size of l bits and its structure is a pipelined architecture with P stages. It works initially in CTR mode, while the ciphertext is scanned to find a special n-bit sequence called synchronization pattern. As it is possible to assume that the output values of E_K working in CTR mode are random and independent, the resulting ciphertext can be considered a random stream after performing the XOR operation between the keystream and the plaintext.

Due to the randomness of the ciphered values, the sync pattern is detected at a statistical random point in the ciphertext. When the pattern is detected, the underlying block cipher E_K does not increment its counter. Instead, it captures the next l ciphertext bits after the sync pattern and uses them as IV (Initialization Vector) to feed back the counter value at the block cipher input. This operation is equivalent to that made by a block cipher working in CFB mode. Moreover, as the block cipher has a pipeline architecture with P stages, it is necessary to disable the sync pattern scanning since the IV is captured until $E_K(IV)$ is available as new keystream block, P cycles later. This interval is called the blackout period and

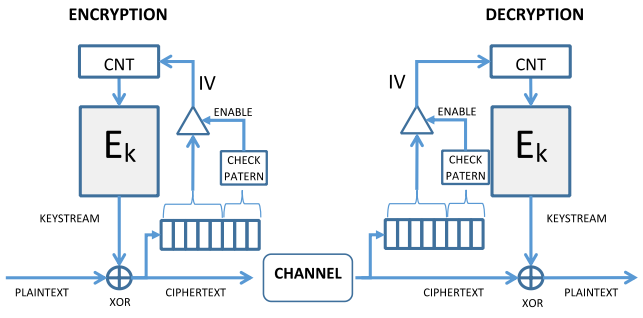

FIGURE 4. Structure of PSCFB mode for encryption and decryption. The block cipher $E_K(\cdot)$ has a block size of l bits, and is implemented using P internal stages. CNT block represents the counter of this mode.

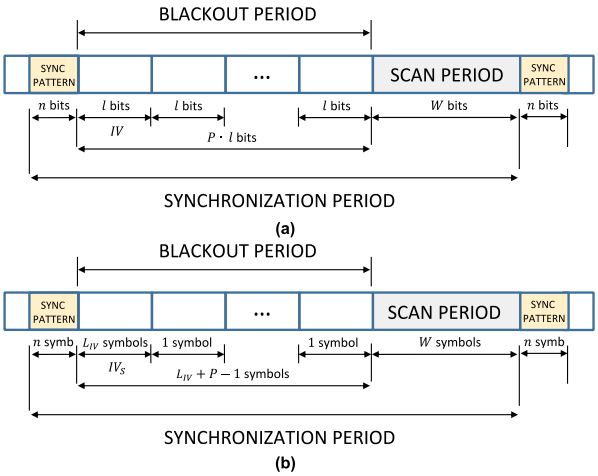

FIGURE 5. Structure of the synchronization period in (a) PSCFB mode in binary radix and (b) PSCFB-MOD with radix S.

it is formed by the IV captured after sync pattern and the next P-1 ciphered b locks of l bits. The complete synchronization cycle is shown in Fig. 5a.

After the counter CNT of Fig. 4 is momentarily loaded with the new IV, it continues with its normal operation, which means a continuous increment (as in CTR mode) until a new sync pattern is found in the ciphertext. In that case, the load operation of the IV is produced again and a new synchronization period starts.

Let us name W to the random bit-length of the scan period and L_{IV} to the length of the IV, then, as mentioned in [19], the average size in bits of a complete sync period u is formed by the length of the sync pattern, $L_{sp} = n$, the blackout period, $L_{bp} = L_{IV} + l \cdot (P-1)$, and the average of the scan period length, $L_{scan} = EW$, until next sync pattern. Therefore, u can be expressed as:

$$u = L_{sp} + L_{bp} + L_{scan}$$
$$= n + L_{IV} + l \cdot (P-1) + E\{W\} = n + l \cdot P + E\{W\} \quad (1)$$

since $L_{IV} = l$ bits.

III. PSCFB-MOD OPERATION MODE

A. PSCFB-MOD DESCRIPTION

To achieve self-synchronization in the encryption/decryption of a plaintext in radix S, the first approach could be applying PSCFB operation mode using an FPE block cipher instead of a traditional one in binary radix, such as AES. Both PSCFB structures, in binary and non-binary radix, are shown in Fig. 6a and Fig. 6b, respectively.

On the one hand, the block cipher E_K in Fig. 6a has to be substituted by an FPE block cipher F_K, such as the FF3 structure built in [14]. On the other hand, the XOR operation in Fig. 6a must be replaced by a modulo-S addition to generate the ciphertext in radix S. These modifications in Fig. 6a results in the structure of Fig. 6b, and it corresponds with the PSCFB mode in radix S that was firstly proposed in [18].

In both solutions, after the sync pattern is detected, the IV is captured and used to refresh the counter value. This IV has a length equals to the input block size l and the same radix as the ciphertext and plaintext.

In this work, to reduce the hardware complexity and latency of this first approach, PSCFB-MOD mode is proposed. Instead of using a block cipher, with the same width and radix for its input and output, a PRF F_K_MOD as described in Section II-B has been used, such that F_K_MOD: $\mathcal{K} \times \{0,1\}^l \rightarrow \{0,\ldots,S-1\}$. The resulting structure is shown in Fig. 6c.

The main difference between this mode and the previous of Fig. 6a and Fig. 6b is that the width and radix at the input of the underlying encryption function F_K_MOD is different to its output. While its input has a binary radix and an l-bit length the output is in radix S and it only has a length of one symbol. For this reason, if the captured IV in PSCFB-MOD mode has a length of L_{IV} symbols in radix S, it must be transformed to an IV with length of l-bits, which means the same size and radix as the counter and the input of F_K_MOD.

Let us name the captured Initialization Vector in radix S as IV_S, then the IV used to refresh the l-bit counter should be

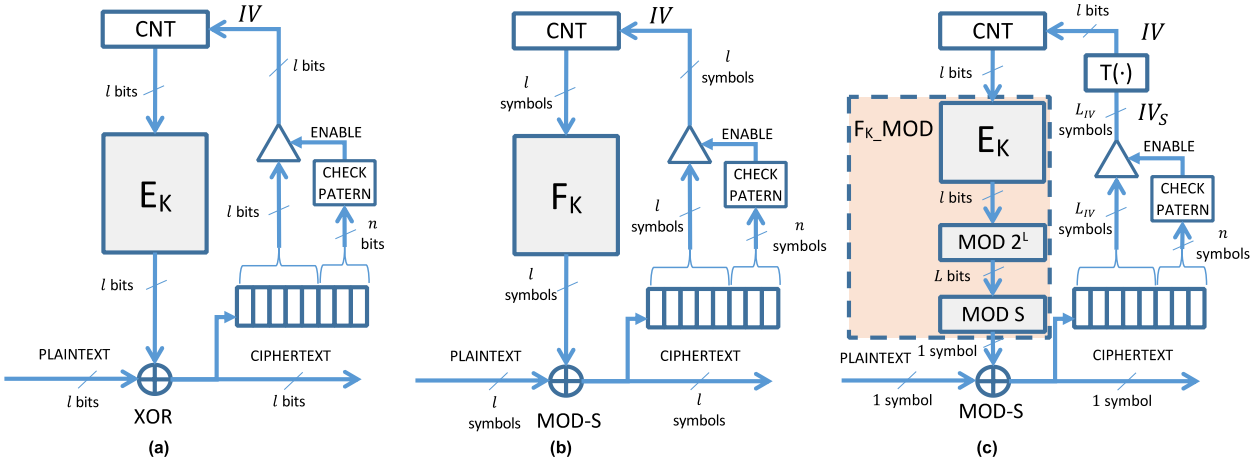

FIGURE 6. Structure of (a) PSCFB mode in binary radix, (b) PSCFB mode in radix S, and (c) PSCFB-MOD mode. E_K represents the block cipher in binary radix and a block size of l bits. F_K represents the FPE block cipher in radix S and a block size of l symbols. Symbols in cases (b) and (c) are in radix S.

obtained from IV_S thanks a function $T(\cdot)$, such that:

$$IV = T\left(IV_S\right) \tag{2}$$

The IV_S can be represented by a vector of L_{IV} symbols in radix S such that $IV_S = \left\{IV_{S0}, IV_{S1}, \ldots, IV_{SL_{IV}-1}\right\}$. The transformation function $T(\cdot)$ has consisted on two steps. Firstly, the numeral string IV_S in radix S is converted to its binary radix representation with the function $NUM(\cdot)$, secondly the result from this conversion is truncated to its least significant l bits to get the IV of length l. Therefore $T(\cdot)$ can be expressed as:

$$IV = T\left(IV_S\right) = NUM\left(IV_S\right) mod\ 2^l \tag{3}$$

where $NUM\left(IV_S\right) = \left(\sum_{i=0}^{i<L_{IV}} IV_{Si} \cdot S^i\right)$.

Other issue that has to be taken into account is the length of the sync pattern. In the case of PSCFB in binary radix in Fig. 6a, this pattern is n-bit length. However, in PSCFB schemes for Fig. 6b and Fig. 6c, as the radix is S, the sync pattern will have a length of n symbols. This length will influence in the desired SRD (Synchronization Recovery Delay) for the PSCFB encryption scheme [26], as we discuss in Subsection III-F.

The structure of a synchronization period of PSCFB-MOD is shown in Fig. 5b. As the output block size of F_K_MOD is one symbol the blackout period will be formed by the L_{IV} symbols captured after sync pattern and the next P-1 ciphered symbols. According to this, the average length in symbols of a complete sync period u is shown in (4) in a similar way as in (1).

$$u = L_{sp} + L_{bp} + L_{scan} = n + L_{IV} + P - 1 + E\left\{W\right\} \tag{4}$$

where W is the random length in symbols of the scan period.

B. IND-CPA SECURITY IN CTR AND PSCFB MODES

Usually the security of the confidentiality-only operation modes for block ciphers is studied in the sense of IND-CPA (Indistinguishability under Chosen-Plaintext Attack) security [27]. A metric called adversary advantage is obtained thanks to a game between an active adversary A and an encryption oracle performing the target encryption scheme Sε. This encryption scheme Sε is configured with a key and an experiment bit b, which the adversary tries to guess.

During the game the adversary sends to the oracle a sequence of p pair of messages $(M_1^0, M_1^1), \ldots, (M_p^0, M_p^1)$. Both messages in each pair (M_i^0, M_i^1) have the same length m_i. For each query from the adversary, the oracle responds with the ciphertext C_i corresponding to the message M_i^b. Finally the adversary will try to guess if the oracle encrypted $(M_1^0, \ldots, M_p^0)$ or $(M_1^1, \ldots, M_p^1)$, which is the same as guessing the value of b after the p queries.

To measure the success of the adversary in breaking a symmetric encryption scheme Sε, the adversary advantage is defined in [28] as in the following equation:

$$ADV_{S\varepsilon}^{IND-CPA}\left(A\right) = 2 \cdot Pr\left(\hat{b} = b\right) - 1 \tag{5}$$

where the $ADV_{S\varepsilon}^{IND-CPA}\left(A\right)$ is the IND-CPA advantage of the adversary A over the encryption scheme $S\varepsilon$, and $Pr\left(\hat{b} = b\right)$ is the probability of the adversary A of guessing the correct value of configuration bit b. The advantage of A can be understood as the excess of this probability over 1/2. When the 'guess' probability is almost 1/2 and then the adversary advantage is negligible, the encryption scheme Sε can be considered secure.

It is demonstrated in [29] that an adversary B attacking the PRF security of the underlying block cipher E_K of $S\varepsilon$ can be built thanks to the adversary A, and their advantages are related as follows:

$$ADV_{S\varepsilon}^{IND-CPA}\left(A\right) = 2 \cdot ADV_{E_K}^{PRF}\left(B\right) + ADV_{S\varepsilon(Func)}^{IND-CPA}\left(A\right) \tag{6}$$

where $ADV_{S\varepsilon}^{IND-CPA}\left(A\right)$ is the advantage of A attacking $S\varepsilon$ when its underlying encryption function is E_K, $ADV_{SE(Func)}^{IND-CPA}\left(A\right)$ is the advantage of A over SE when the underlying encryption function is a random function $Func(l, l)$ such that $Func(l, l) : \{0, 1\}^l \rightarrow \{0, 1\}^l$, and $ADV_{E_K}^{PRF}\left(B\right)$ is the prf-advantage of any adversary B over E_K as defined in [27].

In the formal security proofs of CTR and PSCFB modes, expression in (6) can be particularized changing SE term by CTR and PSCB names, respectively:

$$ADV_{CTR}^{IND-CPA}\left(A\right) = 2 \cdot ADV_{E_K}^{PRF}\left(B\right) + ADV_{CTR(Func)}^{IND-CPA}\left(A\right) \tag{7}$$

$$ADV_{PSCFB}^{IND-CPA}\left(A\right) = 2 \cdot ADV_{E_K}^{PRF}\left(B\right) + ADV_{PSCFB(Func)}^{IND-CPA}\left(A\right) \tag{8}$$

where $ADV_{CTR}^{IND-CPA}$ and $ADV_{PSCFB}^{IND-CPA}$ are the IND-CPA advantages expressions of any adversary A against CTR and PSCFB encryption modes, respectively, and $ADV_{CTR(Func)}^{IND-CPA}$ and $ADV_{PSCFB(Func)}^{IND-CPA}$ are the advantages over each encryption scheme when the underlying encryption function is a random function $Func(l, l)$. In both security proofs the terms $ADV_{CTR(Func)}^{IND-CPA}\left(A\right)$ and $ADV_{PSCFB(Func)}^{IND-CPA}\left(A\right)$ are obtained [29], [13]. It allows to reach the final advantage expression of both operation modes.

In [29] it is proven that:

$$ADV_{S\varepsilon(Func)}^{IND-CPA}\left(A\right) \leq P_{S\varepsilon}(col) \tag{9}$$

where $P_{S\varepsilon}(col)$ is the probability of a collision among the counter values used during the adversary game, it means the probability of a counter value repetition. According to this, (7) and (8) can be rewritten as:

$$ADV_{CTR}^{IND-CPA}\left(A\right) \leq 2 \cdot ADV_{E_K}^{PRF}\left(B\right) + P_{CTR}(col) \tag{10}$$

$$ADV_{PSCFB}^{IND-CPA}\left(A\right) \leq 2 \cdot ADV_{E_K}^{PRF}\left(B\right) + P_{PSCFB}(col) \tag{11}$$

where $P_{CTR}(col)$ and $P_{PSCFB}(col)$ are the counter collision probability for CTR and PSCFB, respectively.

Since in CTR mode the counter is incremented in each encryption step and it is never repeated, the probability of collision is zero: $P_{CTR}(col) = 0$. However, during the

IND-CPA game for PSCFB mode it is possible for the counter to be fed with a new IV value when the sync pattern is detected randomly at some point of the ciphertext. As this new counter value is a random one, this and the subsequent values of the counter during the next synchronization cycle could produce a collision with the values of the counter in previous cycles, then $P_{PSCFB}(col) \neq 0$. According to this, in [29] and [13] $P_{\mathrm{SE}}(col)$ is obtained, giving rise to the following expressions:

$$ADV_{CTR}^{IND-CPA}(A) \leq 2 \cdot ADV_{E_K}^{PRF}(B) \tag{12}$$

$$ADV_{PSCFB}^{IND-CPA}(A) \leq 2 \cdot ADV_{E_K}^{PRF}(B) + \frac{\mu^2}{l^2 2^l} \cdot \frac{P+1}{P^2} \tag{13}$$

For the specific case of PSCFB the second term $P_{PSCFB}(col)$ is not zero. This term depends on some parameters that define the architecture of the PSCFB scheme, such as the block size l and the number of pipeline stages P of the block cipher E_K, and also on the quantity of information bits μ encrypted by the oracle during the adversary game. From (12) and (13) it is possible to conclude that CTR mode, although it does not have the self-synchronous property, it is inherently more secure than PSCFB when using the same underlying block cipher E_K. Although the first term $ADV_{E_K}^{PRF}$ in both expressions is the same, the second one in CTR is zero, which makes IND-CPA advantage be lower in CTR than in PSCFB mode.

In the same way as in PSCFB, it is possible to obtain an IND-CPA advantage expression for PSCFB-MOD, in terms of some parameters of its encryption scheme. The idea in this work is to stablish the bounds for these parameters, under which the resulting structure of PSCFB-MOD is better in terms of IND-CPA advantage than an operation mode taken as reference. Particularly, the mode used as reference has been CTR, since, in general, it can be considered the best to achieve confidentiality-only encryption [30].

The expression of the IND-CPA security for PSCFB-MOD mode is given in Subsection III-C, while its proof is shown in Appendix A. The comparison analysis between PSCFB-MOD and CTR is explained in Subsection III-D.

C. IND-CPA SECURITY IN PSCFB-MOD MODE

It is possible to express the IND-CPA security of PSCFB-MOD mode according to the following theorem:

Theorem 1: Let F_K_MOD: $\mathcal{K} \times \{0,1\}^l \rightarrow \{0,\ldots,S-1\}$ be the underlying function of the encryption scheme SE that corresponds with PSCFB-MOD symmetric encryption mode. Let A be an adversary attacking the IND-CPA security of SE that asks at most p queries formed each one for a pair of messages (M_i^0, M_i^1) with a length of m_i symbols of radix S. The p message queries will produce a total number of q encrypted symbols, which means that $q = \sum_{i=1}^{i=p} m_i$.

Then, it is possible to express the IND-CPA advantage of A over the PSCFB-MOD scheme in terms of the prf-advantage of any adversary over the block cipher E_K that is part of the F_K_MOD function, such that:

$$ADV_{PSCFB-MOD}^{IND-CPA}(A) \leq 2 \cdot ADV_{E_K}^{PRF}(B) + \\ + \frac{q}{2^{I-1}} + P_{max_{r_i}} \cdot \frac{q^2}{P-1+L_{IV}} \tag{14}$$

where $ADV_{E_K}^{PRF}(B)$ is the prf-advantage of any adversary B over E_K, E_K corresponds to the block cipher that is part of the F_K_MOD function, P is the total number of pipeline stages in F_K_MOD, L_{IV} is the length of the IV_S used in PSCFB-MOD scheme and I is the difference between L (the input bit length of modulo-S operation in F_K_MOD) and T, with $T = log_2 S$.

As soon as $L \geq 128 + \log_2(1+2S)$ the term $P_{max_{r_i}}$ can be expressed as:

$$P_{max_{r_i}} \leq 1/2^l + 1/S^{L_{IV}} \tag{15}$$

where l is the block size of E_K.

The proof of Theorem 1 is developed in Appendix A.

D. SECURITY ANALYSIS: PSCFB-MOD VS CTR

Although usually block ciphers are analyzed as PRFs, PRPs (Pseudo Random Permutations) are what best models them. Thanks to the PRF-PRP switching lemma [31] it is possible to relate the PRF and PRP advantages of an adversary against a block cipher as shown in (16).

$$ADV_{E_K}^{PRF}(A) \leq ADV_{E_K}^{PRP}(A) + \frac{q^2}{2^{l+1}} \tag{16}$$

where $ADV_{E_K}^{PRF}(A)$ and $ADV_{E_K}^{PRP}(A)$ are the prf-advantage and prp-advantage of adversary A against block cipher E_K, respectively. The block size is l and the number of encryption queries performed by the adversary during the prf-advantage game is q.

According to (16), IND-CPA advantages for CTR and PSCFB-MOD in (12) and (14) can be rewritten as:

$$ADV_{CTR}^{IND-CPA}(A) \leq 2 \cdot ADV_{E_K}^{PRP}(B) + \frac{q_{CTR}^2}{2^{l_{CTR}}} \tag{17}$$

$$ADV_{PSCFB-MOD}^{IND-CPA}(A) \leq 2 \cdot ADV_{E_K}^{PRP}(B) + \frac{q_{PSCFB-MOD}^2}{2^{l_{PSCFB-MOD}}} \\ + \frac{q_{PSCFB-MOD}}{2^{I-1}} + P_{max_{r_i}} \cdot \frac{q_{PSCFB-MOD}^2}{P-1+L_{IV}} \tag{18}$$

where E_K is the underlying block cipher, l_{CTR} and $l_{PSCFB-MOD}$ are the block sizes of E_K in CTR and PSCFB-MOD, respectively, and q_{CTR} and $q_{PSCFB-MOD}$ are the number of encrypted blocks and symbols during the IND-CPA games of each mode. For each encrypted block, CTR mode encrypts l_{CTR} information bits, while for each encrypted symbol PSCFB-MOD encrypts T information bits ($T = log_2 S$). Therefore, the total number of encrypted bits in each mode during the IND-CPA game are $\mu_{CTR} = q_{CTR} \cdot l_{CTR}$ for CTR mode and $\mu_{PSCFB-MOD} = q_{PSCFB-MOD} \cdot T$ in PSCFB-MOD mode, respectively.

TABLE 1. IND-CPA advantage comparison of different modes.

Encryption Mode $\mathcal{SE}(E)$	IND-CPA Advantage expression[1] $ADV^{IND-CPA}_{\mathcal{SE}(E)}(A)$
PSCFB-MOD[3]	$2 \cdot ADV^{PRP}_{E}(B) + \frac{\mu^2}{T^2} \cdot f + \frac{\mu}{T \cdot 2^{I-1}}$
CTR	$2 \cdot ADV^{PRP}_{E}(B) + \frac{\mu^2}{l^2 2^l}$
CTR$	$2 \cdot ADV^{PRP}_{E}(B) + \frac{2\mu^2}{l^2 2^l}$
CBC	$2 \cdot ADV^{PRP}_{E}(B) + \frac{2\mu^2}{l^2 2^l}$
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, μ the number of encrypted bits and T the bits mapped per symbol.

[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 block size l and $select(\cdot)$ is a function that outputs m fixed bits from its input.

[3]The term f is : $f = 1/2^l + P_{max_{r_i}}/(P - 1 + L_{IV})$.

According to this, it is possible to express the advantages of (17) and (18) in terms of the encrypted bits and compare them with the IND-CPA advantages of other well-known operation modes, as shown in Table 1.

We want to parametrize the structure of PSCFB-MOD to get a security at least better than the CTR mode when encrypting the same amount of information. It means to obtain the constraints needed to meet condition (19).

$$ADV^{IND-CPA}_{PSCFB-MOD}(A) \leq ADV^{IND-CPA}_{CTR}(A) \tag{19}$$

We assume that in both modes it is used a secure underlying block cipher E_K that can be considered a good PRP, which means that the term $ADV^{PRP}_{E_K}$ is negligible in both expressions, (17) and (18). According to this, it is only necessary to compare the second term of both advantage expressions to meet (19), as shown in (20).

$$\frac{q_{PSCFB-MOD}}{2^{I-1}} + q^2_{PSCFB-MOD} \cdot \left(\frac{1}{2^{l_{PSCFB-MOD}}} + \frac{P_{max_{r_i}}}{P - 1 + L_{IV}} \right) \leq \frac{q^2_{CTR}}{2^{l_{CTR}}} \tag{20}$$

Let us name $l_{PSCFB-MOD}$ as l. Since $q_{CTR} = \mu_{CTR}/l_{CTR}$, $q_{PSCFB-MOD} = \mu_{PSCFB-MOD}/T$ and $\mu_{CTR} = \mu_{PSCFB-MOD} = \mu$, expression in (20) can be rewritten as:

$$\frac{\mu}{T \cdot 2^{I-1}} + \frac{\mu^2}{T^2} \cdot \left(\frac{1}{2^l} + \frac{P_{max_{r_i}}}{P - 1 + L_{IV}} \right) \leq \frac{\mu^2}{l^2_{CTR} \cdot 2^{l_{CTR}}} \tag{21}$$

where $I = L - T$.

Assuming what is mentioned in Theorem 1, as soon as $L \geq 128 + \log_2(1 + 2S)$, condition $P_{max_{r_i}} \leq 1/2^l + 1/S^{L_{IV}}$ is met. Then by substituting $P_{max_{r_i}}$ expression in (21) and doing some transformations we get (22) and (23).

$$\frac{1}{T^2} \cdot \left(\frac{1}{2^l} + \frac{1/2^l + 1/S^{L_{IV}}}{P - 1 + L_{IV}} \right) \leq \frac{1}{l^2_{CTR} \cdot 2^{l_{CTR}}} - \frac{1/\mu}{T \cdot 2^{I-1}} \tag{22}$$

$$\frac{1/2^l + 1/S^{L_{IV}}}{P - 1 + L_{IV}} \leq \frac{T^2}{l^2_{CTR} \cdot 2^{l_{CTR}}} - \frac{T/\mu}{2^{I-1}} - \frac{1}{2^l} \tag{23}$$

As $\mu \geq 1$ bit and $P \geq 1$, if (23) is met for $\mu = 1$ it is met for every μ, and the same happens with P. Then (23) can be rewritten giving rise to the final condition:

$$\frac{1/2^l + 1/S^{L_{IV}}}{L_{IV}} \leq \frac{T^2}{l^2_{CTR} \cdot 2^{l_{CTR}}} - \frac{T}{2^{I-1}} - \frac{1}{2^l}$$
$$L \leq 128 + \log_2(1 + 2S) \tag{24}$$

Equation (24) is the final condition necessary to fix the values of the PSCFB-MOD parameters such as l, L and L_{IV}. Note that also condition mentioned in Theorem 1 must be met, then it has also been included in (24).

According to this, we can conclude that if the underlying block ciphers used in CTR and PSCFB-MOD modes are good PRPs and have a negligible $ADV^{PRP}_{E_K}$ term, it is possible to obtain a PSCFB-MOD configuration to achieve the same or better IND-CPA security than CTR when encrypting the same amount of information. The parameters of PSCFB-MOD scheme l, L, and L_{IV} will depend on the block size l_{CTR} of the CTR mode taken as reference and the radix S of the plaintext, as $T = log_2 S$. Expression in (24) will be the constraint necessary to achieve condition (19).

E. PSCFB-MOD HARDWARE IMPLEMENTATION FOR 1000BASE-X

As we have mentioned in Section I the PSCFB-MOD encryption system has been adapted to work in the 1000Base-X physical layer for Gigabit Ethernet standard. In this standard only 267 possible symbols are valid in the 8b/10b encoding, which means that $S = 267$.

On the other hand, we assume that we want to achieve at least a better IND-CPA security than a standard and recommended block cipher, such as AES, working in CTR mode with a standard block size of $l_{CTR} = 128$ bits. According to this, (24) can be rewritten as:

$$\frac{1/2^l + 1/S^{L_{IV}}}{L_{IV}} \leq \frac{T^2}{2^{142}} - \frac{T}{2^{I-1}} - \frac{1}{2^l}$$
$$L \geq 128 + \log_2(1 + 2 \cdot 267) \rightarrow L \geq 138 \tag{25}$$

where $T = log_2 S \cong 8.06$ and $I = L - T$.

In this work, we have used the same structure for F_K_MOD than in [20], letting parameters l and L as fixed values such that $l = 192$ and $L = 149$. Therefore to accomplish with (25) it is also necessary that $L_{IV} \geq 17$.

If we substitute the keystream generator of Fig. 6c in the generic structure of Fig. 2, and set the PSCFB-MOD parameters as $l = 192$, $L = 149$ and $L_{IV} = 17$ then it is possible to get the final encryption scheme for PCS sublayer in 1000Base-X standard as shown in Fig.7.

In the same way as in [20], to get a block size $l = 192$, we have used as underlying block cipher E_K a Rijndael structure configured with 192 bit block size and 128 bit key length. For modulo operation functions, the block MOD_2^{149} of Fig. 7 is simply to take the 149 least significant bits

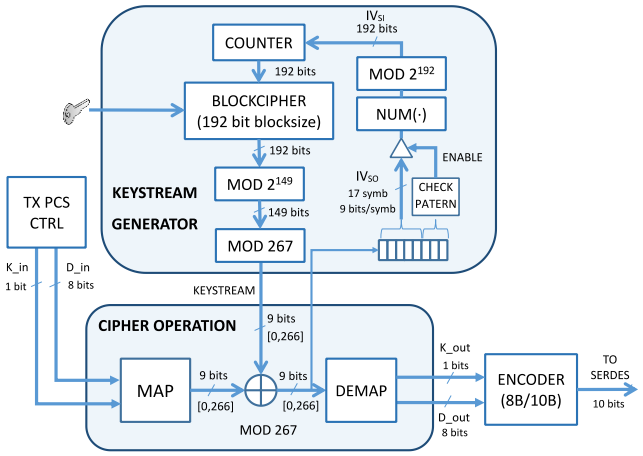

FIGURE 7. Overall structure for PSCFB-MOD mode in a physical layer with 8b/10b encoding. Decryption will be as encryption but using a modulo-267 subtraction instead of an addition.

of the Rijndael output, while MOD_267 module uses more resources as 267 is not a power of two. Its implementation has been based on [32], which presents a high-speed hardware structure for a generic operation '*x mod z*'.

The PSCFB-MOD structure shown in Fig. 7 has been synthetized in a Xilinx Virtex 7 FPGA (Field Programmable Gate Array) and the hardware resources used by this solution are shown in Table 2 in terms of LUTs (Look-Up Tables), registers and block RAMs. In this table PSCFB-MOD implementation is compared with other FPE implementations [33], [20], showing that this work entails a good *Encryption_Rate/resource* ratio. Although in this work the mode PSCFB-MOD entails more hardware resources than CTR-MOD [20], it is at expense of adding the self-synchronization property.

Regarding the comparison with the other PSCFB solution in radix S [18], another benefit of this work is that it does not need input and output queues, which makes it better in terms of encryption latency. While in [18] latency introduced in the PCS datapath is 648 ns, in this work it is only 48 ns.

TABLE 2. Comparison with other solutions.

	FF1 [33]	FF3 [33]	CTR-MOD [20]	PSCFB 10G [13]	This Work
Slice Registers	11285	5592	8807	19154	10317
Slice LUTs	7426	3587	10974	17599	12261
18K Block RAMs	343	170	78	153	78
Slices[1]	3268	1596	3844	6794	4355
Encryption Rate (Mbps)	41.1	109.6	1000	10000	1000
Encryption Rate/Slice[2] (Kbps/Slice)	12.57	68.7	260.1	1471.8	229.6
Self-synchronization	No	No	No	Yes	Yes
Plaintext Format Preservation	Yes	Yes	Yes	No	Yes

[1]Slices are estimated from the number of register and LUTs, assuming they are not packed together.
[2]Althouth Encryption Rate/Slice in [13] is better than in other solutions, this PSCFB schemme is only valid for plaintext in binary radix. It is not able to preserve the format as the PSCFB structure presented in this work does.

As mentioned in the Introduction, in the original PSCFB specification [19] input and output queues are needed 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. It makes PSCFB mode inherently to have an encryption efficiency that can be less than 100%, which makes necessary the usage of input and output encryption queues increasing its overall latency as in [18].

However, in PSCFB-MOD the sync pattern (formed by n symbols) is always aligned with the output block size of F_K_MOD as it is only one symbol length. It means that no queue is needed in this encryption scheme, which makes the encryption throughput be 100% and reduces substantially the overall latency.

The test set-up for the complete Gigabit Ethernet interface and the encryption results are shown in Section IV.

F. SYNCHRONIZATION RECOVERY DELAY DISCUSSION

The SRD (Synchronization Recovery Delay) is the metric used to examine the resynchronization properties of SCFB and PSCFB modes [26]. It is defined as the expected number of bits following a sync loss before synchronization is reestablished. According to [19], upper and lower bounds for SRD depend on the block size of the underlying block cipher, the number of pipelines P and the size of the synchronization pattern n.

As shown in Fig. 5, a sync cycle is composed of fixed length data such as the sync pattern and blackout period, and a variable length data formed by the scan period with a length of W bits or symbols. In [25] and [19] it is shown that W follows a probability distribution that can be approximated by a geometric distribution such as:

$$P(W) = (1-p)^W \cdot p \tag{26}$$

where n is the length of the sync pattern in bits and $p = 1/2^n$. When the radix of the symbols is S instead of 2, as in this work or in [18], it is possible to reach the same conclusion, but using $p = 1/S^n$. According to this, the expressions for the first and second moment of the distribution of W will be as in [25] and [19], but using S as radix:

$$\begin{aligned} E\{W\} &= S^n - 1 \\ E\{W^2\} &= 2{\cdot}S^{2n} - 3 \cdot S^n + 1 \end{aligned} \tag{27}$$

As in (1), we can express, in a generic way, the average sync cycle size in symbols instead of bits. It is the sum of the sync pattern, blackout period and scan period:

$$u = n + L_{IV} + l \cdot (P-1) + E\{W\} \tag{28}$$

where u is the average sync cycle size in symbols, l is the output block size of the underlying encryption function in

symbols, L_{IV} is the size of the *IV* after the sync pattern and P is the pipeline stages.

The same reasoning in [19] relative to the bit slips and misalignments can be made for [18] and this work, but using symbols in radix S instead of bits. Thank to this, the lower bound of SRD can be express in the same way as in [19]:

$$SRD \geq \frac{3}{2}\left(n + L_{IV} + l \cdot (P-1)\right) + \\ + \frac{1}{2u}\left(\left(n + L_{IV} + l \cdot (P-1)\right) E\{W\} + E\{W^2\}\right) \quad (29)$$

If we call $m = n + L_{IV} + l \cdot (P-1)$, then, as in [19], but using radix S instead of binary radix, it is possible to express the upper bound as:

$$SRD \leq \frac{3}{2}m + \frac{1}{2u}m \cdot E\{W\} + \frac{1}{2u}E\{W^2\} + \frac{n}{S^n}m \cdot \lambda \quad (30)$$

where $\lambda = (1 - 1/S^n)^{-m}$.

With both equations, (29) and (30), we can get the theoretical upper and lower bounds for SRD in PSCFB modes with radix S.

As in [25] and [19], with small values of n, better SRD are obtained. Indeed, with the parameters used in this work for S, P, L and L_{IV} the values for SRD become huge when $n > 1$. Then, for a feasible solution n has been set to the minimum $n = 1$.

In the case of PSCFB in radix S [18], $S = 267$, $P = 41$, $l = 22$, $L_{IV} = 22$ and $n = 1$, which makes the maximum SRD bound be $SRD_{max} = 1618$ symbols.

In this work, PSCFB-MOD, the parameters are $S = 267$, $P = 84$, $l = 1$, $L_{IV} = 17$, $n = 1$, then the maximum SRD bound is $SRD_{max} = 382$ symbols.

As the symbol cycle lasts 8 ns in 1000Base-X standard, the upper bound for SRD will be approximately 12.94 μs and 3.56 μs for PSCFB in [18] and PSCFB-MOD in this work, respectively.

IV. TEST SET-UP AND ENCRYPTION RESULTS

A. ENCRYPTION SET-UP

The KEYSTREAM GENERATOR and CIPHER OPERATION modules in Fig. 7 have been integrated in the ENCRYPT and DECRYPT modules of the Ethernet Interface described in Fig. 1, which has been implemented in a Xilinx Virtex 7 FPGA. On the one hand, the Ethernet Interface is connected to an SFP (Small Form-Factor Pluggable) module capable of transmitting at a rate of 1.25 Gbps. On the other hand, it is attached to an Ethernet Frame Generator able to generate and analyze real Ethernet traffic flows composed by data frames. The test set-up is shown in Fig. 8 and Fig. 9, where two FPGA eval boards are faced to test an encrypted optical link.

B. ENCRYPTION THROUTHPUT AND LATENCY

Several conclusions can be drawn from simulation and hardware debugging. On the one hand, it is possible to conclude that encryption/decryption is performed correctly and synchronously achieving the 100% of the encryption throughput.

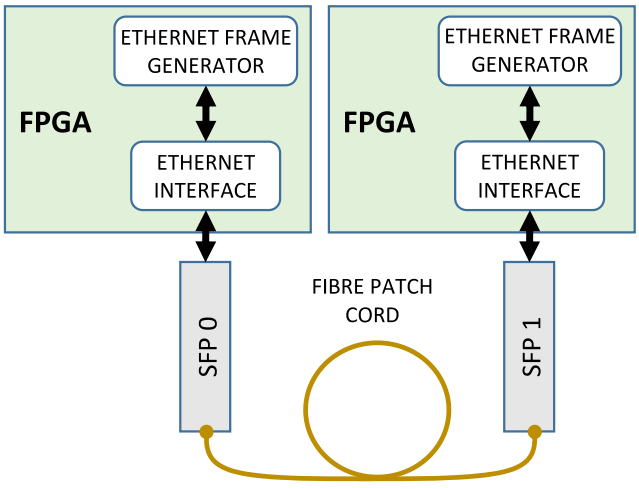

FIGURE 8. Test set-up scheme.

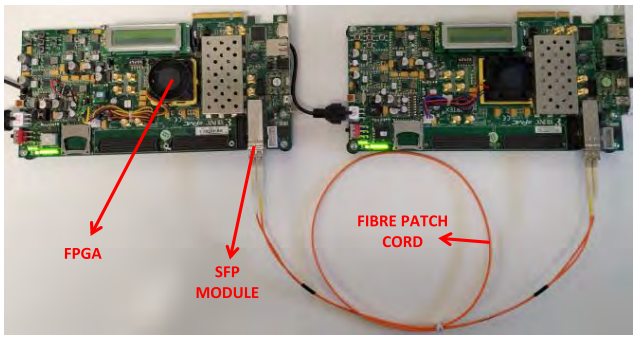

FIGURE 9. Test set-up photo.

In particular, bursts of 1024-byte length frames were tested with a duration of 10^7 frames transmitted using the maximum possible bandwidth of 98%, according to the minimum standard IFG (Inter Frame Gap). Another advantage is that the proposed encryption system only introduces a total extra latency of 48 ns in the PCS datapath, which includes the overall latency of PSCFB-MOD mode without queues.

These advantages of performing encryption at physical layer contrasts with encryption techniques at other layers [34] [35], which introduces overhead on data frames and reduce the overall data bandwidth. For example, in IPsec the encryption overhead can reduce the overall throughput between 20% and 90% of the maximum achievable [36]. In addition, regarding to the previously mentioned latency we can conclude that this work can perform encryption with comparable latency figures to those achieved by other physical layer techniques such as in OTN encryption [12], where latency is in the range of hundreds of nanoseconds [37].

C. TRAFFIC PATTERN ENCRYPTION

One of the benefits of this kind of encryption, as mentioned in [14], is that it is possible to mask data patterns and hide the presence of transmitted frames, for example by

TABLE 3. Measured shannon entropy.

Patterns[1]	SE (t=1)		SE (t=2)		SE (t=3)	
	N-E	E	N-E	E	N-E	E
A	1	8.06	0	8.059	0.33	8.059
B	2.19	8.06	1.05	8.058	1.26	8.059
C	5.48	8.06	4.46	8.058	4.44	8.058
D	7.79	8.06	7.45	8.059	7.42	8.058
E	7.98	8.06	7.82	8.058	7.81	8.059

[1]Pattern A corresponds with the case of no frame transmission, where only idle sets 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 91% of the maximum Gigabit line rate. Pattern E corresponds to continuous frame transmision of random length and minimium IFG. SE is calculated for each pattern without encryption (N-E) and after encryption (E).

making unrecognizable their beginning and end, and also their complete headers, therefore hiding statistical traffic features, which could improve the overall security. As in [14], to prove this capability SE (Shannon Entropy) has been measured as shown in (31) for different encrypted and non-encrypted data traffic patterns. The 8b/10b symbol stream for each traffic pattern, mapped between 0 and $S-1$, has been grouped in tuples of t symbols called β_t, and the probability for each tuple, $P(\beta_t)$, has been calculated. Particularly, SE has been measured for values of t from 1 to 3.

$$SE = -\frac{1}{t} \cdot \sum_{0 \le \beta_t < S^t} P(\beta_t) \cdot \log_2 P(\beta_t) \tag{31}$$

SE result is given in bits/symbol. Ideally, if every t-tuple (β_t) is equally likely with probability $P(\beta_t) = p = S^{-t}$ the value of Shannon Entropy for every t should be as in (32).

$$-\frac{1}{t} \cdot S^t \cdot p \cdot \log_2 p = \log_2 S = \log_2 267 \cong 8.0606 \tag{32}$$

Owing to the limited memory in FPGA hardware resources, measurements for each pattern have been calculated at simulation stage. The results for SE values has been shown in Table 3. In this table it is possible to note that encrypted flows (values in column 'E') achieve the maximum entropy value while non encrypted flows (values in column 'NE') achieve a value different for each traffic pattern and lower than the maximum in (32). This fact shows that the different traffic patterns are indistinguishable when encryption is active, which proves the masking property of the proposed scheme.

V. CONCLUSIONS

In this work the authors have presented a new self-synchronous symmetric encryption scheme able to cipher data preserving its format. The security analysis has been carried out concluding that it is possible to get at least the same or better security than classical CTR mode structures using 128-bit block size ciphers. In addition, as the underlying block cipher in the proposed mode can be a recommended one in binary radix, the hardware complexity is reduced in regards to the typical FPE modes, such as FF1 or FF3.

The proposed solution has been parametrized to work in an optical Ethernet Interface with encryption capabilities. It has been tested with real Ethernet traffic proving its masking property at physical layer. The implementation results also give a better *Encryption_Rate/Slice* ratio than other existing implementations for FPE solutions. Moreover, the proposed self-synchronous encryption structure does not need input and output queues, which contrasts with typical PSCFB schemes, reducing the latency introduced in the encryption datapath.

APPENDIX A
PROOF OF THEOREM 1

By taking into account that the underlying encryption function of PSCFB-MOD is F_K_MOD instead of E_K, it is possible to express directly its IND-CPA advantage expression in the same way as with PSCFB in (11), such that:

$$ADV^{IND-CPA}_{PSCFB-MOD}(A) \le 2 \cdot ADV^{PRF}_{F_K\_MOD}(B) + P_{PSCFB-MOD}(col) \tag{33}$$

where $ADV^{IND-CPA}_{PSCFB-MOD}$ is the advantage of A attacking the PSCFB-MOD scheme when its underlying function is the PRF F_K_MOD, $ADV^{PRF}_{F_K\_MOD}$ is the prf-advantage over F_K_MOD and $P_{PSCFB-MOD}(col)$ is the collision probability of the counter during the IND-CPA game for PSCFB-MOD.

According to the following two lemmas it is possible to proof Theorem 1.

Lemma 1: Let F_K_MOD: $\mathcal{K} \times \{0,1\}^l \rightarrow \{0,\ldots,S-1\}$ be a PRF, such that $F_K\_MOD(x) = \left(E_K(x) \bmod 2^L\right) \bmod S$, as defined in Section II-B, where S is not a power of two, E_K is a block cipher with block size l and L is an integer number smaller than l. Then, according to [20], any adversary A making q oracle queries when attacking the prf-security of F_K_MOD will obtain an advantage bounded by the following expression:

$$ADV^{PRF}_{F_K\_MOD}(A) \le ADV^{PRF}_{E_K}(B) + \frac{q}{2^I} \tag{34}$$

where $I = L - T$, $T = log_2 S$ and $ADV^{PRF}_{E_K}(B)$ is the prf-advantage of any adversary over E_K. Note that in the definition of the prf-advantage $ADV^{PRF}_{F_K}$ of any adversary over a generic PRF F_K [27], the adversary sends q blocks of data to an oracle able to perform that PRF. Then the oracle answers to the adversary with the q encrypted blocks. In the case of the advantage over F_K_MOD, as it has an input size of l-bits and an output size of one symbol in radix S, the q queries performed by the adversary will correspond with q encrypted symbols instead of q blocks of l-bit width.

Lemma 2: Let F_K_MOD: $\mathcal{K} \times \{0,1\}^l \rightarrow \{0,\ldots,S-1\}$ be a PRF, such that $F_K\_MOD(x) = \left(E_K(x) \bmod 2^L\right) \bmod S$, where S is not a power of two, E_K is a block cipher with block size l and L is an integer number smaller than l. Then the term $ADV^{PRF}_{F_K\_MOD}$ can be considered lower than the prf-advantage

over a generic block cipher with block size l_{REF} bits as soon as $L \geq l_{REF} + \log_2(1 + 2S)$.

Lemma 3: Let SE be a PSCFB-MOD scheme with an underlying encryption function F_K_MOD such that F_K_MOD: $\mathcal{K} \times \{0,1\}^l \rightarrow \{0,\ldots,S-1\}$, where L_{IV} is the length of the IV_S used in the PSCFB-MOD scheme, P is the number of pipeline stages for F_K_MOD and q is the number of encrypted symbols during the p message queries performed in the IND-CPA adversary game against this encryption mode. Then the probability of collision $P_{SE}(col)$ of any counter value during the p queries is bounded by the following expression:

$$P_{PSCFB-MOD}(col) \leq P_{max_{r_i}} \cdot q^2/(P - 1 + L_{IV}) \quad (35)$$

where $P_{max_{r_i}} \leq 1/2^l + 1/S^{L_{IV}}$ as soon as the parametrization of F_K_MOD makes it be a good PRF when comparing it with a recommended block cipher with a 128 bit block size, as specified in Lemma 2. It means that L must meet the condition $L \geq 128 + \log_2(1 + 2S)$.

By substituting the terms $ADV^{PRF}_{F_K\_MOD}(A)$ and $P_{PSCFB-MOD}(col)$ of equations (34) and (35) in (33) it is possible to get as result equation (14) which proves Theorem 1.

Expression of Lemma 1 is already proven in [20]. For clarity purposes proofs of Lemma 2 and Lemma 3 are included in Appendices B and C, respectively.

APPENDIX B
PROOF OF LEMMA 2

In the proof of Lemma 2 it is necessary to take into account that the underlying encryption function in PSCFB must be a good PRF. In this way it is possible to consider that its output values are random and independent. As F_K_MOD is the underlying function in PSCFB-MOD then it is necessary to know under what condition we can consider it as a good PRF.

According to [27] when a random function works as the underlying function in a counter scheme, successive values of the counter makes the output values of the function be random and independent. When the underlying function is not a random one, but a PRF, then if the prf-advantage of any adversary over this function is low enough we can consider that the PRF is undistinguishable from a random function and its output values also can be considered random and independent.

In general, recommended block ciphers can be considered good enough to generate random values as keystream in counter schemes such as CTR or PSCFB. The idea of this proof is to compare the prf-advantages of a recommended block cipher and the function F_K_MOD. Let be E_{K_REF} the block cipher used as reference with a block size of l_{REF} bits. Let be F_K_MOD a PRF such that $F_K\_MOD(x) = \left(E_K(x) \bmod 2^L\right) \bmod S$, where E_K is a block cipher with block size l and L is an integer number smaller than l. Then their prf-advantages should be related as in (36).

$$ADV^{PRF}_{F_K\_MOD}(A) \leq ADV^{PRF}_{E_{K\_REF}}(A) \quad (36)$$

According to Lemma 1 [20] prf-advantage of F_K_MOD is bouded by the following expression:

$$ADV^{PRF}_{F_K\_MOD}(A) \leq ADV^{PRF}_{E_K}(B) + \frac{q}{2^I} \quad (37)$$

where q is the number of queries performed during the prf-advantage game and $I = L - T$.

On the other hand, thanks to PRF-PRP switching lemma expression (16) it is possible to rewrite prf-advantages of F_K_MOD and E_{K_REF} as:

$$ADV^{PRF}_{F_K\_MOD}(A) \leq ADV^{PRP}_{E_K}(B) + \frac{q^2}{2^{l+1}} + \frac{q}{2^I}$$
$$ADV^{PRF}_{E_{K\_REF}}(A) \leq ADV^{PRP}_{E_{K\_REF}}(A) + \frac{q^2_{REF}}{2^{l_{REF}+1}} \quad (38)$$

where q_{REF} are the number of queries performed during the prf-advantage game of E_{K_REF}.

By taking into account (38) and assuming that we want to get a better bound for the prf-advantage with F_K_MOD than with E_{K_REF} when performing the same number of queries in the prf-game, which means $q_{REF} = q$, then (36) can be expressed as:

$$ADV^{PRP}_{E_K}(B) + \frac{q^2}{2^{l+1}} + \frac{q}{2^I} \leq ADV^{PRP}_{E_{K\_REF}}(A) + \frac{q^2}{2^{l_{REF}+1}} \quad (39)$$

We assume that in both cases secure underlying block cipher E_K and E_{K_REF} are used. As block ciphers can be considered good PRPs it means that the term $ADV^{PRP}_{E_K}$ is negligible in both sides of the condition in (39). According to this, it is only necessary to compare the second term of both advantage expressions to meet (39).

$$\frac{1}{2^{l+1}} + \frac{1}{q \cdot 2^I} \leq \frac{1}{2^{l_{REF}+1}} \quad (40)$$

As $q \geq 1$ the left side of (40) is maximum with $q = 1$. Then if (40) meets for $q = 1$ it will met for every q. Therefore:

$$\frac{1}{2^l} + \frac{1}{2^{I-1}} \leq \frac{1}{2^{l_{REF}}} \quad (41)$$

As $L \leq l$, let be $d = l - L$. Since $I = L - T$, then:

$$\frac{1}{2^{d+L}} + \frac{1}{2^{L-T-1}} \leq \frac{1}{2^{l_{REF}}} \quad (42)$$

By doing some transformations in (42) we get:

$$2^L \geq 2^{l_{REF}}\left(2^{T+1} + 1/2^d\right) \quad (43)$$

As d ≥ 0 the right side in (43) will be maximum with $d = 0$. In addition $T = log_2 S$. Therefore (43) can be rewritten as:

$$L \geq l_{REF} + log_2(1 + 2S) \quad (44)$$

Equation (44) will be the condition to meet (36) and consider that F_K_MOD is a better PRF than a block cipher with block size l_{REF}, which proves Lemma 2.

PROOF OF LEMMA 3

The generic expression for the collision probability $P_{\mathrm{SE}}(col)$ of the counter values during the IND-CPA game against counter based encryption schemes SE is given in [29]. Particularly, traditional CTR and CTR\$ modes are analyzed. In CTR mode the counter values are never repeated while in CTR\$ the counter is reset to a random value at the beginning of a message query, which means that some repetition can happen between the counter values used to encrypt different messages.

As mentioned in Section III-B each message query in the IND-CPA game is formed by a pair of messages (M_i^0, M_i^1) with a length of m_i blocks of data. The p message queries will produce a total number of q encrypted blocks, which means that $q = \sum_{i=1}^{i=p} m_i$. These blocks of data have the same length and radix as the output of the underlying encryption function. In the case of PSCFB-MOD these blocks are formed by one symbol in radix S, as F_K_MOD output.

For the case of CTR\$, the underlying encryption function is the block cipher E_K, with an input and output that are l-bit length. The l-bit length counter values along the game are represented in the following table:

$$\begin{array}{ccc} r_1+1, r_1+2, \ldots, r_1+m_1 \\ r_2+1, r_2+2, \ldots, r_2+m_2 \\ \ldots \quad\quad \ldots \quad\quad \ldots \\ r_p+1, r_p+2, \ldots, r_p+m_p \end{array} \tag{45}$$

In (45) each i-th row correspond with the counters used in the encryption of the pair (M_i^0, M_i^1) with length m_i. At the beginning of each query a random counter value r_i is generated. As the counter is incremented in each encryption step and the message has a length of m_i data blocks the last value of the counter in each row will be $r_i + m_i$.

In [29] it is demonstrated that the collision probability between any counter value in the i-th row and the previous row counters is limited by:

$$\Pr(col_i | no\_col_{i-1}) \leq \frac{Nchoices(r_i)}{2^l} \tag{46}$$

where col_i is the event where there is a collision in the first i rows of the table, and no_col_{i-1} the event where there is no collision in the first i-1 rows. $Nchoices(r_i)$ are the number of choices of r_i that can produce a collision and $1/2^l$ is the probability for r_i to be equal to any of these choices. We call this probability $P_{r_i}(x) = P(r_i = x)$, with x any l-bit length value in $Nchoices(r_i)$. As r_i is generated randomly we can consider it has a uniform random distribution such that $P_{r_i}(y) = P_{r_i} = 1/2^l$ for any i and $y \in [0, 2^l - 1]$. It means that $P_{r_i}(x)$ is equal to $1/2^l$ for any $x \in Nchoices(r_i)$.

Finally, given (46), in [29] it is shown that $P_{\mathrm{SE}}(col)$ for CTR\$, the collision probability along the counter table in (45), can be expressed as:

$$\begin{aligned} P_{CTR\$}(col) &\leq \sum_{i=2}^{p} \Pr(col_i \mid no\_col_{i-1}) \leq \frac{\sum_{i=2}^{i=p} Nchoices(r_i)}{2^l} \\ &\leq \frac{(p-1)\cdot\sum_{i=1}^{i=p} m_i}{2^l} = P_{r_i}\cdot(p-1)\cdot\sum_{i=1}^{i=p} m_i \end{aligned} \tag{47}$$

For the particular case of PSCFB, in [13] it is shown that the encryption session can be divided into different sync cycles as the shown in Fig. 5. In each sync cycle the counter is reset to a random IV value as mentioned in Section II-C. Let us assume that N sync cycles happen during the IND-CPA game between the adversary and the oracle. Then, the counter values during the session can be represented as in the following table:

$$\begin{array}{ccc} r_1, r_1+1, \ldots, r_1+k_1-1 \\ r_2, r_2+1, \ldots, r_2+k_2-1 \\ \ldots \quad\quad \ldots \quad\quad \ldots \\ r_N, r_N+1, \ldots, r_N+k_N-1 \end{array} \tag{48}$$

where k_i is the length in blocks of data of the i-th sync cycle. The subsequent counters from r_i in advance will be incremented up to $r_i + k_i - 1$. According to [13], by using the same reasoning than in [29], $P_{PSCFB}(col)$ can be expressed as in (47), in terms of P_{r_i} :

$$P_{PSCFB}(col) \leq P_{r_i}\cdot(N-1)\cdot\sum_{i=1}^{i=N} k_i \tag{49}$$

In PSCFB mode, as in CTR\$, the counter is an l-bit length value. It is reset to the IV at the beginning of each sync cycle and it can be considered random and uniformly distributed. Therefore, in (47) $P_{r_i} = 1/2^l$ for any i and $x \in [0, 2^l - 1]$.

In the case of PSCFB-MOD mode, as the encryption game can be also divided in N sync cycles the same expression as (49) could be applied to obtain $P_{PSCFB-MOD}(col)$.

However, as explained in Section III-A, the l-bit length IV is not taken directly from the ciphertext, but its value is obtained from the L_{IV}-symbol length value IV_S, such that $IV = T(IV_S)$. As the ciphertext, in radix S, can be considered random and uniformly distributed, the same happens with IV_S, as it is taken from the L_{IV} symbols after the sync pattern detection. Therefore, IV_S can be considered a uniform random value in the range $[0, S^{L_{IV}} - 1]$. Particularly the transformation $T(\cdot)$ as been chosen such that $IV = T(IV_S) = NUM(IV_S) \bmod 2^l$ as shown in (3).

IV_S and its binary representation, $NUM(IV_S)$, can be considered random and uniformly distributed as soon as F_K_MOD output is considered also random, which means that parameter L in F_K_MOD must meet condition in Lemma 2. However, IV is obtained by truncating $NUM(IV_S)$ to its least l significant bits. Therefore, IV cannot be considered random and uniformly distributed as a bias is introduced due to modulo-2^l operation. Owing to this, r_i counter values at the beginning of each sync cycle will not be uniformly distributed and special considerations must be taken into account to obtain $P_{r_i}(x)$ as it will not be constant and equal to $1/2^l$.

Let assume $P_{max_{r_i}}$ as the maximum probability in the probability density function $P_{r_i}(x)$, then if we particularize (49) to PSCFB-MOD mode we can get an upper bound for the following collision probability such that:

$$P_{PSCFB-MOD}(col) \leq P_{max_{r_i}}\cdot(N-1)\cdot\sum_{i=1}^{i=N} k_i \tag{50}$$

In addition, as the output block size in PSCFB-MOD is one symbol with radix S, then $k_i = q_i$, where q_i is the number of symbols with radix S encrypted in the i-th sync cycle, then:

$$P_{PSCFB-MOD}(col) \leq P_{max_{r_i}} \cdot (N-1) \cdot \sum_{i=1}^{i=N} q_i = P_{maxr_i} \cdot (N-1) \cdot q \leq P_{max_{r_i}} \cdot N \cdot q \leq P_{max_{r_i}} \cdot N_{max} \cdot q \quad (51)$$

where q is the total number of symbols encrypted during the IND-CPA game and N_{max} the maximum possible number of sync cycles produced during the game. It is possible to express this value as $N_{max} = q/C_{min}$, where C_{min} is the minimum size of any sync cycle in symbols. As shown in Fig. 5b the generic sync cycle length in PSCFB-MOD can be expressed as $u = n + L_{IV} + P - 1 + W$ symbols, where n, L_{IV} and P are constant, while W is variable. The minimum value for this cycle length could be considered with $W = 0$, then $u_{min} = n + L_{IV} + P - 1$, where P is the pipeline size of the underlying block cipher and n the sync pattern size. If we call $w = P - 1 + L_{IV}$ then $N_{max} = q/u_{min} \leq q/w$, therefore:

$$P_{PSCFB-MOD}(col) \leq P_{max_{r_i}} \cdot \frac{q^2}{w} = P_{max_{r_i}} \cdot \frac{q^2}{(P-1+L_{IV})} \quad (52)$$

which is the same as equation (35) in Lemma 3.

The last step to prove Lemma 3 is to demonstrate that $P_{max_{r_i}} \leq 1/2^l + 1/S^{L_{IV}}$, which is proved in the following paragraphs.

Let us name $NUM(IV_S)$ as v_i and IV as r_i. Then v_i and r_i will be two random variables such that $r_i = v_i \ mod \ 2^l$, where $v_i \in [0, M-1]$, $r_i \in [0, 2^l - 1]$ and M is not a multiple of 2^l. If F_K_MOD meets Lemma 2 its captured output v_i will be random and uniformly distributed, then $P(v_i = x) = 1/M$, where $P(v_i = x)$ is the probability that v_i is equal to x and $x \in [0, M-1]$. In Fig. 10 the way of mapping values from the domain $\{0, \ldots, M-1\}$ to $\{0, \ldots, 2^l - 1\}$ thanks to modulo-2^l operation is shown. The remainder R of the division between M and 2^l can be written as:

$$R = M - d \cdot 2^l \quad (53)$$

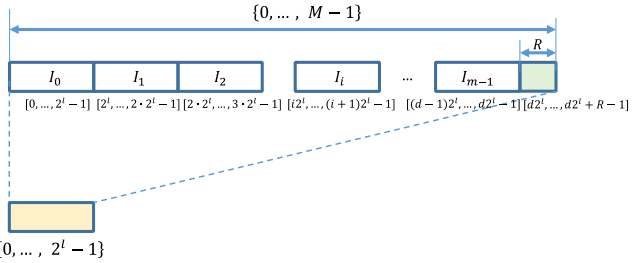

FIGURE 10. Domain $\{0, \ldots, M-1\}$ is mapped into interval $\{0, \ldots, 2^l-1\}$. After modulo-2^l operation, each range I_i is equally distributed in the destination range $\{0, \ldots, 2^l-1\}$, however the last range $\{d2^l, \ldots, d2^l+R-1\}$, only has R values and they are mapped only in the first R values of $\{0, \ldots, 2^l-1\}$. It makes each value in the destination range $\{0, \ldots, R-1\}$ has one more occurrence than in $\{R, \ldots, 2^l-1\}$.

where d is the quotient of the division, which means that $d = \lfloor M/2^l \rfloor$. After modulo-$2^l$ operation, as R is different to zero, a bias is introduced in the resulting distribution of values of r_i in the range $\{0, \ldots, 2^l - 1\}$. v_i values that are in the range $\{d2^l, \ldots, d2^l + R - 1\}$ will generate one more occurrence in the resulting range $\{0, \ldots, R-1\}$ after modulo operation. Therefore, the probability distribution after modulo-2^l will be:

$$P(r_i = x) = \begin{cases} \frac{d+1}{M} & for \ 0 \leq x \leq R-1 \\ \frac{d}{M} & for \ R \leq x \leq 2^l - 1 \end{cases} \quad (54)$$

The maximum value for the probability of $P(r_i = x)$ will be:

$$P_{max_{r_i}} = P(r_i = x | 0 \leq x \leq R-1) = \frac{d+1}{M} \quad (55)$$

As $d = \lfloor M/2^l \rfloor$ then:

$$P_{max_{r_i}} = \frac{\lfloor M/2^l \rfloor}{M} + \frac{1}{M} \leq \frac{M/2^l}{M} + \frac{1}{M} = \frac{1}{2^l} + \frac{1}{M} \quad (56)$$

As we have called $v_i = NUM(IV_S)$ then $v_i \in [0, M-1] \rightarrow v_i \in [0, S^{L_{IV}} - 1]$, which means that $M = S^{L_{IV}}$ and $P_{max_{r_i}}$ can be expressed as:

$$P_{max_{r_i}} \leq 1/2^l + 1/S^{L_{IV}} \quad (57)$$

The final expression for $P_{PSCFB-MOD}(col)$ in (52) next to the condition in (57) proves Lemma 3.

REFERENCES

[1] Cisco Systems, "Cisco global cloud index: Forecast and methodology. 2015–2020," Cisco, San Jose, CA, USA, White Paper, 2016. Accessed: Aug. 17, 2017. [Online]. Available: https://www.cisco.com/c/dam/m/en_us/service-provider/ciscoknowledgenetwork/files/622_11_15-16-Cisco_GCI_CKN_2015-2020_AMER_EMEAR_NOV2016.pdf

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

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

[4] H. Rahbari and M. Krunz, "Full frame encryption and modulation obfuscation using channel-independent preamble identifier," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 12, pp. 2732–2747, Dec. 2016.

[5] Y. Zhao, X. Zou, Z. Lu, and Z. Liu, "Chaotic encrypted polar coding scheme for general wiretap channel," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 12, pp. 3331–3340, Dec. 2017.

[6] W. Li, D. Mclernon, J. Lei, M. Ghogho, S. A. R. Zaidi, and H. Hui, "Cryptographic primitives and design frameworks of physical layer encryption for wireless communications," *IEEE Access*, vol. 7, pp. 63660–63673, 2019.

[7] 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.

[8] M. Furdek, N. Skorin-Kapov, S. Zsigmond, and L. Wosinska, "Vulnerabilities and security issues in optical networks," in *Proc. 16th Int. Conf. Transparent Opt. Netw. (ICTON)*, Graz, Austria, Jul. 2014.

[9] 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, Sep. 2011.

[10] 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, Oct. 2017.

[11] K. Cui, J. Wang, H.-F. Zhang, C.-L. Luo, G. Jin, and T.-Y. Chen, "A real-time design based on FPGA for expeditious error reconciliation in QKD system," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 1, pp. 184–190, Jan. 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 *Proc. 42nd Eur. Conf. Opt. Commun.*, Düsseldorf, Germany, Sep. 2016, pp. 1–3.
[13] A. Perez-Resa, M. Garcia-Bosque, C. Sanchez-Azqueta, and S. Celma, "Self-synchronized encryption for physical layer in 10Gbps optical links," *IEEE Trans. Comput.*, vol. 68, no. 6, pp. 899–911, Jun. 2019.
[14] A. Perez-Resa, M. Garcia-Bosque, C. Sanchez-Azqueta, and S. Celma, "Physical layer encryption for industrial Ethernet in gigabit optical links," *IEEE Trans. Ind. Electron.*, vol. 66, no. 4, pp. 3287–3295, Apr. 2019.
[15] J. Han, H. Won, and H.-M. Bae, "0.6–2.7-Gb/s referenceless parallel CDR with a stochastic dispersion-tolerant frequency acquisition technique," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 6, pp. 1219–1225, Jun. 2014.
[16] P. Rogaway, "A synopsis of format-preserving encryption," Voltage Secur., Cupertino, CA, USA, Tech. Note, 2013. [Online]. Available: https://web.cs.ucdavis.edu/~rogaway/papers/synopsis.pdf
[17] M. Dworkin, "Recommendation for block cipher modes of operation: Methods for format-preserving encryption," NIST, Gaithersburg, MD, USA, NIST Special Publication 800-38G, 2016.
[18] A. Perez-Resa, M. Garcia-Bosque, C. Sanchez-Azqueta, and S. Celma, "Self-synchronized encryption using an FPE block cipher for gigabit Ethernet," in *Proc. 15th Conf. Ph.D Res. Microelectron. Electron. (PRIME)*, Lausanne, Switzerland, Jul. 2019, pp. 81–84.
[19] H. M. Heys and L. Zhang, "Pipelined statistical cipher feedback: A new mode for high-speed self-synchronizing stream encryption," *IEEE Trans. Comput.*, vol. 60, no. 11, pp. 1581–1595, Nov. 2011.
[20] A. Perez-Resa, M. Garcia-Bosque, C. Sanchez-Azqueta, and S. Celma, "A new method for format preserving encryption in high-data rate communications," *IEEE Access*, vol. 8, pp. 21003–21016, 2020.
[21] M. Robshaw and O. Billet, *New Stream Cipher Designs: The eSTREAM Finalists*. Berlin, Germany: Springer, 2008.
[22] J. Daemen, J. Lano, and B. Preneel, "Chosen ciphertext attack on SSS," Ecrypt, Tech. Rep., 2005, p. 235. [Online]. Available: http://www.ecrypt.eu.org/stream/papersdir/044.pdf
[23] A. Joux and F. Muller, "Chosen-ciphertext attacks against MOSQUITO," in *Fast Software Encryption*, M. Robshaw, Ed. Berlin, Germany: Springer, 2006, pp. 390–404.
[24] A. Alkassar, A. Geraldy, B. Pfitzmann, and A.-R. Sadeghi, "Optimized self-synchronizing mode of operation," in *Proc. Conf. Fast Softw. Encryption (FSE)*, Apr. 2001, pp. 78–91.
[25] O. Jung and C. Ruland, "Encryption with statistical self-synchronization in synchronous broadband networks," in *Proc. Conf. Cryptograph. Hardw. Embedded Syst. (CHES)*, 1999, pp. 340–352.
[26] H. M. Heys, "Analysis of the statistical cipher feedback mode of block ciphers," *IEEE Trans. Comput.*, vol. 52, no. 1, pp. 77–92, Jan. 2003.
[27] M. Bellware and P. Rogaway. (May 2005). *Introduction to Modern Cryptography*. [Online]. Available: http://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf
[28] M. Bellare and P. Rogaway, "Chapter 5.4 indistinguishably under chosen-plaintext attack," in *Introduction to Modern Cryptography*. May 2005. [Online]. Available: http://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf
[29] M. Bellare and P. Rogaway, "Chapter 5.7 security of CTR modes," in *Introduction to Modern Cryptography*. May 2005. [Online]. Available: http://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf
[30] P. Rogaway, "Evaluation of some blockcipher modes of operation," Cryptogr. Res. Eval. Committees (CRYPTREC) Goverment Japan, Tokyo, Japan, Tech. Rep., Feb. 2011. [Online]. Available: https://web.cs.ucdavis.edu/~rogaway/papers/modes.pdf
[31] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway, "A concrete security treatement of symmetric encryption: Analysis of the DES modes of operation," in *Proc. Symp. Found. Comput. Sci. (FOCS)*, 1997, pp. 394–403.
[32] J. T. Butler and T. Sasao, "Fast hardware computation of X mod Z," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. Workshops Phd Forum (IPDPSW)*, Shanghai, China, May 2011, pp. 294–297.
[33] R. Agbeyibor, J. Butts, M. Grimaila, and R. Mills, "Evaluation of format-preserving encryption algorithms for critical infrastructure protection," in *Critical Infrastructure Protection VIII*. Berlin, Germany: Springer, 2014, pp. 245–261.
[34] S. S. Kolahi, Y. R. Cao, and H. Chen, "Bandwidth-IPSec security trade-off in IPv4 and IPv6 in windows 7 environment," in *Proc. 2nd Int. Conf. Future Gener. Commun. Technol. (FGCT)*, London, U.K., Nov. 2013, pp. 148–152.
[35] C. Xenakis, N. Laoutaris, L. Merakos, and I. Stavrakakis, "A generic characterization of the overheads imposed by IPsec and associated cryptographic algorithms," *Comput. Netw.*, vol. 50, no. 17, pp. 3225–3241, Dec. 2006.
[36] L. Troell, J. Burns, K. Chapman, D. Goddard, M. Soderlund and C. Ward, "Converged vs. dedicated IPSec encryption testing in gigabit Ethernet networks," Rochester Inst. Technol., Rochester, NY, USA, Tech. Rep. 1743, 2005. [Online]. Available: http://scholarworks.rit.edu/article/1743
[37] MicroSemi. (2016). *In-Flight Encryption in Service Provider Networks, No: PMC-2150716, Issue 2*. Accessed: Aug. 17, 2017. [Online]. Available: https://pmcs.com/cgi-bin/document.pl?docnum=2150716

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. He is currently pursuing the Ph.D. degree from the Group of Electronic Design, Aragon Institute of Engineering Research, University of Zaragoza.

He was an Research and Development Engineer with telecommunications industry for more than ten years. He is also 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. degrees in physics from the University of Zaragoza, Zaragoza, Spain, in 2014 and 2015, respectively. He is currently pursuing the Ph.D. degree from the Group of Electronic Design, Aragon Institute of Engineering Research, University of Zaragoza.

He is also 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 also 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 with the Group of Electronic Design, Aragon Institute of Engineering Research, University of Zaragoza. He has coauthored more than 100 technical articles and 300 international conference contributions. He has coauthored four technical books and the holder of four patents. He appears as a 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.