

PARTIAL IMPLEMENTATION OF THE PHYSICAL LAYER OF USB 3.1 RECEIVER

RONALDO ENRIQUE SERRANO PEÑA

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS  
ESCUELA DE INGENIERÍAS  
ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES  
BUCARAMANGA  
2020

PARTIAL IMPLEMENTATION OF THE PHYSICAL LAYER OF USB 3.1 RECEIVER

RONALDO ENRIQUE SERRANO PEÑA

Trabajo de Grado para optar al título de  
Ingeniero Electrónico

Director

Ckristian Ricardo Esteban Duran Blanco,  
MEng. Electrónica.

Codirector

Elkim Felipe Roa Fuentes,  
Philosophy Doctor

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS  
ESCUELA DE INGENIERÍAS  
ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES  
BUCARAMANGA  
2020

## CONTENIDO

|                                   | <b>pág.</b> |
|-----------------------------------|-------------|
| <b>INTRODUCTION</b>               | <b>10</b>   |
| <b>1. 128B/132B BLOCK ALIGNER</b> | <b>12</b>   |
| 1.1. STATE OF THE ART             | 12          |
| 1.2. PROPOSED ARCHITECTURE        | 13          |
| 1.3. RESULTS                      | 17          |
| <b>2. ELASTIC BUFFER</b>          | <b>19</b>   |
| 2.1. STATE OF THE ART             | 19          |
| 2.2. PROPOSED ARCHITECTURE        | 20          |
| 2.3. RESULTS                      | 22          |
| <b>3. DESCRAMBLING</b>            | <b>25</b>   |
| <b>4. CONCLUSIONS</b>             | <b>27</b>   |
| <b>BIBLIOGRAFÍA</b>               | <b>28</b>   |

## LISTA DE FIGURAS

|   | <b>pág.</b> |
|---|-------------|
| Figura 1. Traditional block diagram of a USB 3.1 receiver   | 10          |
| Figura 2. Traditional displacement control architecture of the 128b/132b block aligner decoder.   | 13          |
| Figura 3. Proposed architecture of the 128b/132b block aligner decoder.                           | 14          |
| Figura 4. Functional behavior of the proposed architecture of 128b/132b block aligner.            | 16          |
| Figura 5. RC corners of proposed architecture of 128b/132b block aligner decoder.                 | 17          |
| Figura 6. Layout of the proposed 128b/132b block aligner decoder in CMOS 0.18 $\mu$ m technology. | 18          |
| Figura 7. A conventional architecture of the elastic buffer.                                      | 20          |
| Figura 8. Proposed architecture of the elastic buffer.  | 21          |
| Figura 9. Functional behavior of the proposed elastic buffer.                                     | 22          |
| Figura 10. RC corners of proposed architecture of the elastic buffer.                             | 23          |
| Figura 11. Layout of the proposed elastic buffer in CMOS 0.18 $\mu$ m technology.                 | 24          |
| Figura 12. Proposed block diagram of the descrambling.  | 26          |

## LISTA DE TABLAS

|  | <b>pág.</b> |
|--|-------------|
| Tabla 1. Header control correction                               | 15          |
| Tabla 2. Summary results of the proposed 128b/132b block aligner | 18          |
| Tabla 3. Summary results of the proposed elastic buffer          | 23          |

## RESUMEN

**TÍTULO:** IMPLEMENTACIÓN PARCIAL DE LA CAPA FÍSICA DE UN RECEPTOR USB 3.1 \*

**AUTOR:** RONALDO ENRIQUE SERRANO PEÑA \*\*

**PALABRAS CLAVE:** 128b/132b alineador de bloques, bufer elastico, descrambling, frecuencia de operación, RC corners.

### DESCRIPCIÓN:

Este documento presenta la implementación de un 128b/132b alineador de bloques y un bufer elástico en una tecnología de bajo costo estándar CMOS 0.18 $\mu$ m. Se aborda el problema de la reducción del ancho de banda y de la reducción del área ocupada paralelizando los bits de entrada de la capa física digital. La arquitectura implementada en el 128b/132b alineador de bloques usa la técnica de control de concatenación, reduciendo 36 % del área ocupada y aumenta la frecuencia máxima de operación 48 %. Además, la arquitectura implementada del bufer elástico usa la técnica de ensanchamiento de canal, reduciendo 78 % del área ocupada y aumenta la frecuencia máxima de operación 42 %. La parte implementada de la capa física funciona a una frecuencia de operación de 416MHz.

Este trabajo demuestra la viabilidad de la implementación de los circuitos digitales presentes en la capa física de un receptor USB 3.1 en una tecnología de bajo costo. Al presentar los resultados de síntesis, se observa el incremento de la frecuencia de operación al comparar con las arquitecturas tradicionales presentes en el estado del arte. De este modo, es posible implementar un transceiver compatible con USB 3.1 en una tecnología con un proceso de fabricación mas grande al estándar CMOS 28 nm, lo cual reduce en gran medida el costo de un transceiver compatible con USB 3.1.

---

\* Trabajo de grado

\*\* Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: Ckristian Ricardo Esteban Durán Blanco, MEng. Electrónica. Codirector: Elkim Felipe Roa Fuentes, Philosophy Doctor

## ABSTRACT

**TITLE:** PARTIAL IMPLEMENTATION OF THE PHYSICAL LAYER OF USB 3.1 RECEIVER \*

**AUTHOR:** RONALDO ENRIQUE SERRANO PEÑA \*\*

**KEYWORDS:** 128b/132b block aligner, elastic buffer, descrambling, operation frequency, RC corners.

### DESCRIPTION:

This document presents an implementation of a 128b/132b block aligner and an elastic buffer in a CMOS standard 0.18 $\mu$ m low-cost technology. We address the problem of reducing bandwidth and occupying a smaller area by increasing the number of parallel bits in the physical structure. The architecture implemented for the 128b/132b block aligner uses the technique of concatenation control, reducing 36 % the area occupied, and the maximum operation frequency increases 48 %. Regarding the architecture implemented for the elastic buffer use the technique of channel enlargement, synthesis results demonstrate a reduction of 78 % for the occupied area, and the maximum operation frequency increases 42 %. The digital-physical layer interface supports a worst-case configurable-by-generation frequency up to 416MHz.

This work demonstrates the viability of the implementation of the digital circuits present in the physical layer of a USB 3.1 receiver in low-cost technology. We present the results of the synthesis, observe an increase in the frequency of operation compare to the traditional state of the art architectures. In this way, it will be possible to implement a transceiver compatible with USB 3.1 in technology with a manufacturing process higher than CMOS standard 28 nm, which significantly reduces the cost of a USB 3.1 compatible transceiver.

---

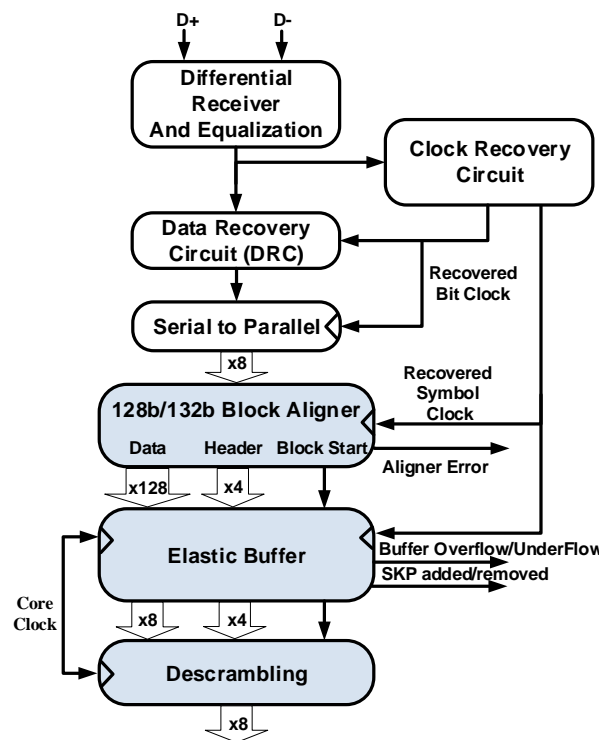
\* Bachelor Thesis

\*\* Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: Ckristian Ricardo Esteban Durán Blanco, MEng. Electrónica. Codirector: Elkim Felipe Roa Fuentes, Philosophy Doctor

## INTRODUCTION

Serial communication protocols during the last decades have significantly increased the data transfer rate. For example, the change in the USB 3.0 standard to the USB 3.1 standard increased the data rate twice (from 5Mbps to 10Mbps), generating different challenges in the circuits that composed the physical layer of the transceiver<sup>1</sup>.

Figura 1. Traditional block diagram of a USB 3.1 receiver.



The first challenge is associated to the technology high-cost, to achieve higher operation frequency. For example, in a  $0.18\mu\text{m}$  CMOS low-cost technology node, the operating frequency of a register achieves 1.15GHz (The clock conditions defined in

<sup>1</sup> HEWLETT-PACKARD COMPANY y col. *Universal Serial Bus 3.0 specification, revision 1.0*. USB-IF, November 12, 2008.

the USB 3.1 standard are: 100ps Clock Jitter and 50ps Transition time), making it unfeasible to the required 1.25GHz maximum operating frequency in USB 3.1. The second challenge is the area increasing of the storing circuits in the transceiver, as it is the case of the 128b/132b block aligner and the elastic buffer. Finally, the third challenge corresponds to the reduction of the operating frequency by conventional circuits as it is the case of the Finite State Machine (FSM), the counters and the displacement control. Each of these blocks has a critical path that limits the maximum operating frequency.

We present in Fig. 1 the physical layer architecture of the USB 3.1 standard receiver<sup>2</sup>. The differential receiver and equalization block amplifies the incoming data from the transmission channel. Next, the Clock and Data Recovery (CDR) circuit acquires from the received signal, the data and the bit-symbol clocks. Then, the incoming data is converted from serial to parallel for later data processing. The data processing is executed by the 128b/132b Block Aligner, the Elastic Buffer and the Descrambling to obtain the transmitted data without errors.

In this work, we show a partial implementation (128b/132b Block Aligner and Elastic Buffer) of the physical layer of a USB 3.1 receptor in a  $0.18\mu\text{m}$  technology node. The input of the 128b/132b Block Aligner is 24 bits long, enough to meet the requirements of both USB 3.1 and the technology constraints. The design is intended to work with a 32-bit Direct Memory Access (DMA) in a system-on-a-chip<sup>3</sup>. We reach a combined frequency of 416.66MHz, and a total area of  $190 \times 190 \mu\text{m}^2$  for the block aligner, and  $147 \times 147 \mu\text{m}^2$  for the elastic buffer.

---

<sup>2</sup> HEWLETT-PACKARD COMPANY. *Universal Serial Bus 3.1 specification, revision 1.0*. USB-IF, July 26, 2013.

<sup>3</sup> Hanssel MORALES, DURAN, Ckristian y ROA, Elkim. "A Low-Area Direct Memory Access Controller Architecture for a RISC-V Based Low-Power Microcontroller". En: *2019 IEEE 10th Latin American Symposium on Circuits Systems (LASCAS)*. 2019, págs. 97-100.

## 1. 128B/132B BLOCK ALIGNER

Decoders used in standard high-speed serial protocols involve the use of modern manufacturing technologies due to the high bandwidth required, this makes wanting to use low-cost technologies not viable. This section focuses on presenting a proposed architecture for the design of a 128b/132b decoder that reduces the critical signal path that affects the maximum operating frequency without affecting the occupied area.

### 1.1. STATE OF THE ART

A decoder 128b/132b has the function of classifying the incoming data into 128-bit output frames and 4-bit header frames. The data frame that enters the decoder has two functions that depend on the header. The first function indicates if the data is valid, and for this case the header takes the 1100'b value. For the second function, the header takes the 0011'b value, and it associates the data frame for the physical layer control to synchronise the link layer.

Figure 2 shows a traditional topology <sup>456</sup> used for the 128b/132b decoder, where the critical signal path between the Shift Register and the Displacement Control negatively impacts the maximum operating frequency. In addition, the displacement control

---

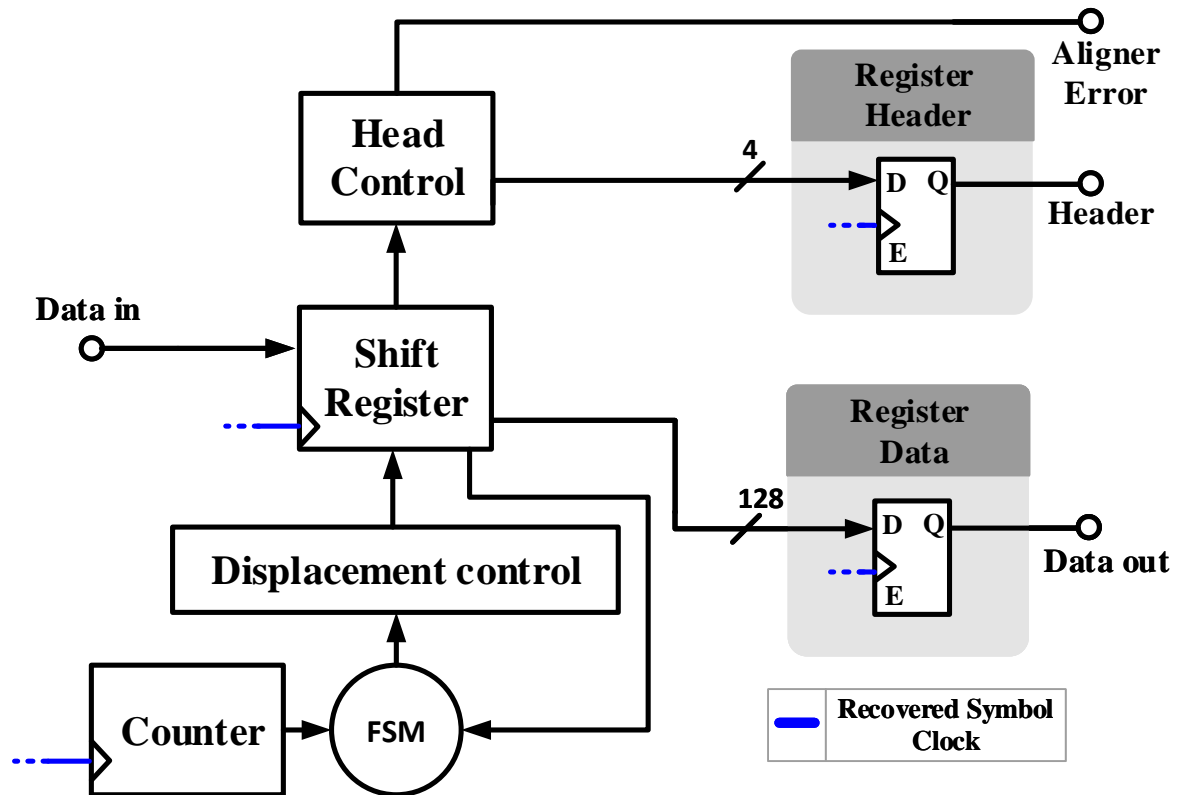
<sup>4</sup> Rakeshroshan MALDAL y PHILEMON, Daniel. "A Study to Implement Superspeed Plus USB 3.1 Physical Layer." En: *Proceedings of ASAR-IJIEEE International Conference*. 2014.

<sup>5</sup> Shah MONIKA y SHRUTHI, G. "Study and Implementation of Physical Layer Coding Used in Super Speed USB". En: *International Journal of Scientific Engineering and Research*. 2014.

<sup>6</sup> Sergio Andrés. TOVAR DURÁN. "DECODIFICADOR 132b/128b COMPATIBLE CON USB3.1 EN TECNOLOGÍA CMOS 180NM". En: *Universidad Industrial de Santander*. 2017.

logic occupies a large area compared to the proposed architecture as will be presented later.

Figura 2. Traditional displacement control architecture of the 128b/132b block aligner decoder.

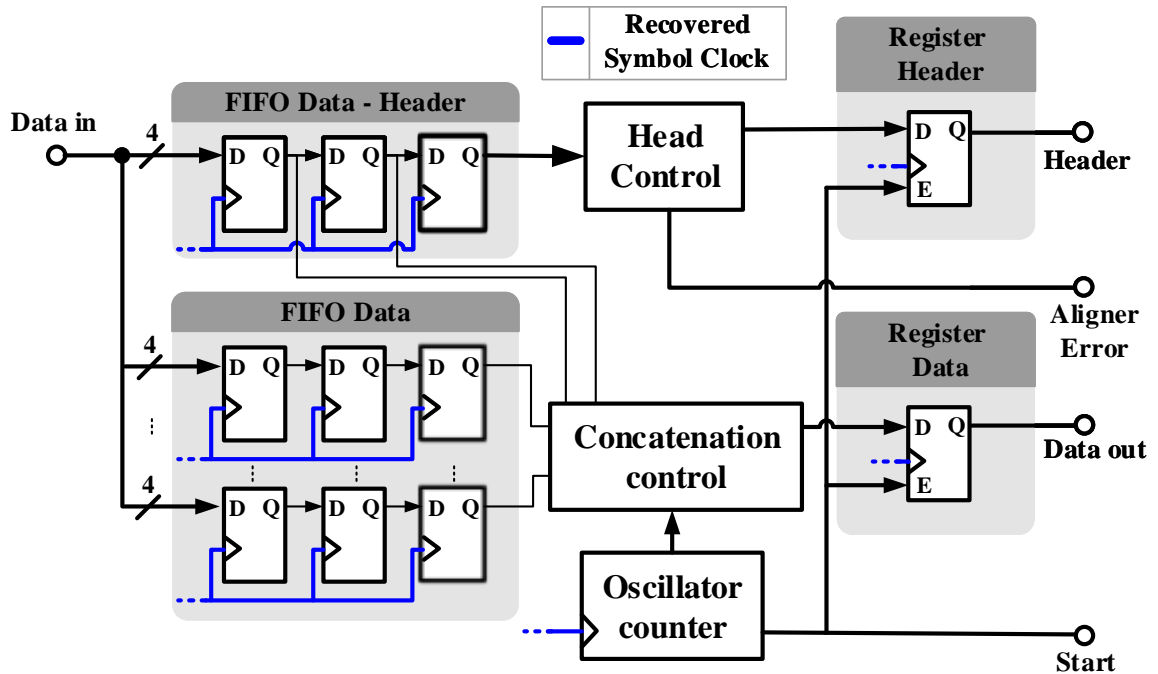


## 1.2. PROPOSED ARCHITECTURE

The approach is based on reducing the critical signal path to increase the maximum operating frequency in a low-cost standard CMOS technology  $0.18\mu\text{m}$ . The new architecture uses a concatenation control logic instead of the displacement control. The concatenation control logic adopts two storage channels composed by FIFOs to distinguish between the data and the header. The first channel, FIFO Data-Header, has

the function of storing the header and part of the data frame (28 bits). The second channel stores 100 bits in 4-bit sub-channels. The number of input bits supported by the proposed architecture is 8 and 24 bits.

Figura 3. Proposed architecture of the 128b/132b block aligner decoder.



The concatenation control needs a signal that indicates the position of the current information packet header, which is generated via an oscillating counter. The header may arrive with errors, which affect the utility of the data frame that the decoder delivers to the elastic buffer. To solve this problem, the decoder's header control has the possibility to correct a 1-bit error.

Table 1 shows all the possible cases of the header control and the corresponding correction for the 1-bit error header frames. The 2-bit error header frames are not possible to correct due to the uncertainty of the correct value. For the proposed architecture, it is remarkable that the critical signal path of the base architecture is between FIFO Data-Header and the output of the Register Header, as illustrated in

Fig. 3.

Tabla 1. Header control correction

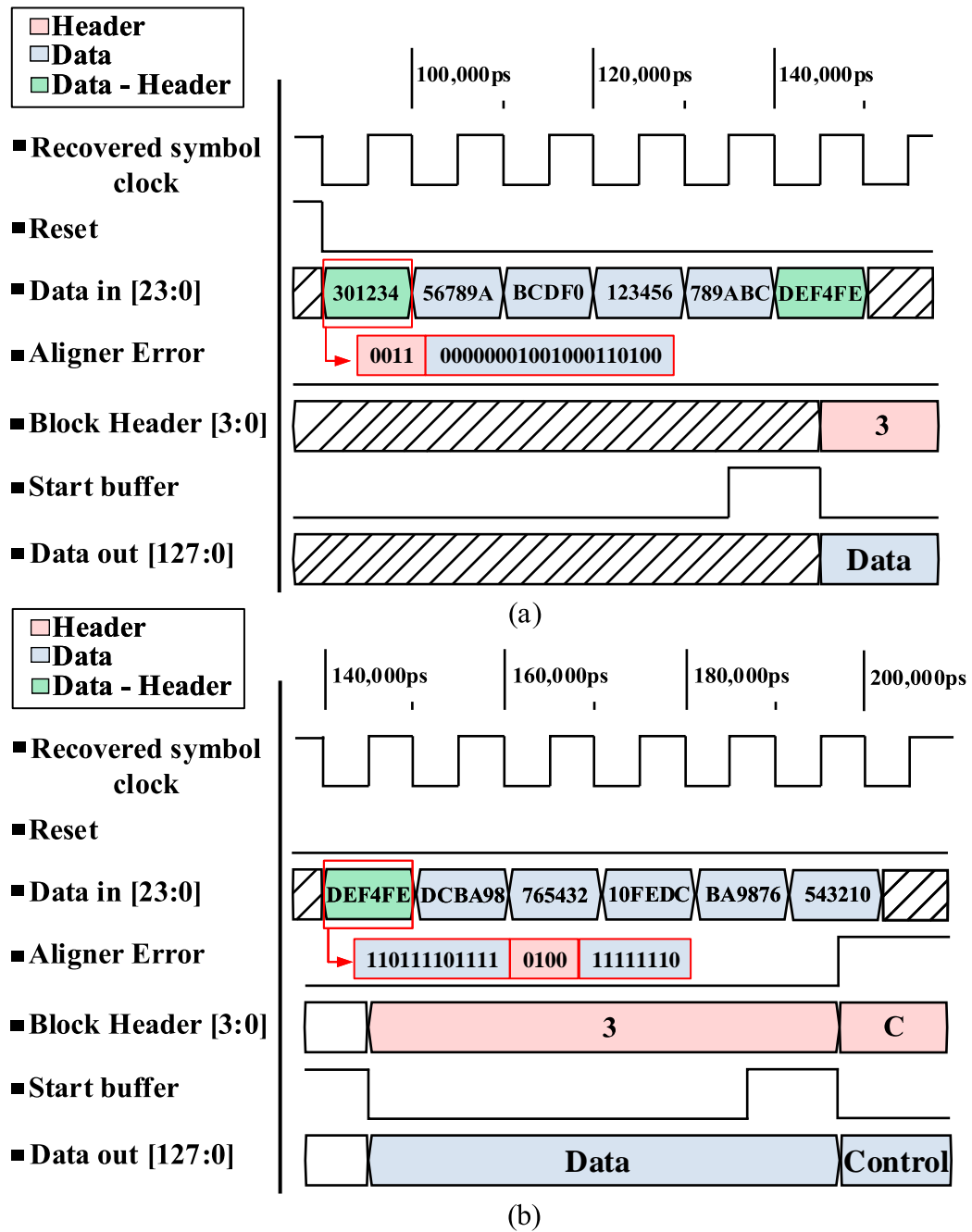
| <b>Header</b> | <b>Correction</b> | <b>Aligner Error</b> |
|---------------|-------------------|----------------------|
| 0000          | —                 | 2-bit                |
| 0001          | 0011              | 1-bit                |
| 0010          | 0011              | 1-bit                |
| 0011          | N/A               | 0-bit                |
| 0100          | 1100              | 1-bit                |
| 0101          | —                 | 2-bit                |
| 0110          | —                 | 2-bit                |
| 0111          | 0011              | 1-bit                |
| 1000          | 1100              | 1-bit                |
| 1001          | —                 | 2-bit                |
| 1010          | —                 | 2-bit                |
| 1011          | 0011              | 1-bit                |
| 1100          | N/A               | 0-bit                |
| 1101          | 1100              | 1-bit                |
| 1110          | 1100              | 1-bit                |
| 1111          | —                 | 2-bit                |

The functional behavior of the 128b/132b block aligner is based on two sequential states, as depicted in Fig 4. The first state occurs when the header of the information packet is at the beginning of the incoming bit frame (a). The second state takes place when the header of the information packet is crossed with the information of another information packet (b). In both cases, the information is extracted correctly, and the data and the header are stored separately.

Finally, the recovered symbol clock reaches the decoder at a frequency of As mentioned before, the 1.25GHz is not achievable by the low-cost CMOS standard 0.18 $\mu$ m technology. The proposed architecture uses a 416.66MHz (worst-case) recovery symbol clock for a 24-input bits, instead of the 1.25GHz frequency clock associa-

ted to the USB 3.1 standard, for the 8bits input case. The latter sets the maximum operating frequency for the elastic buffer and the descrambling blocks.

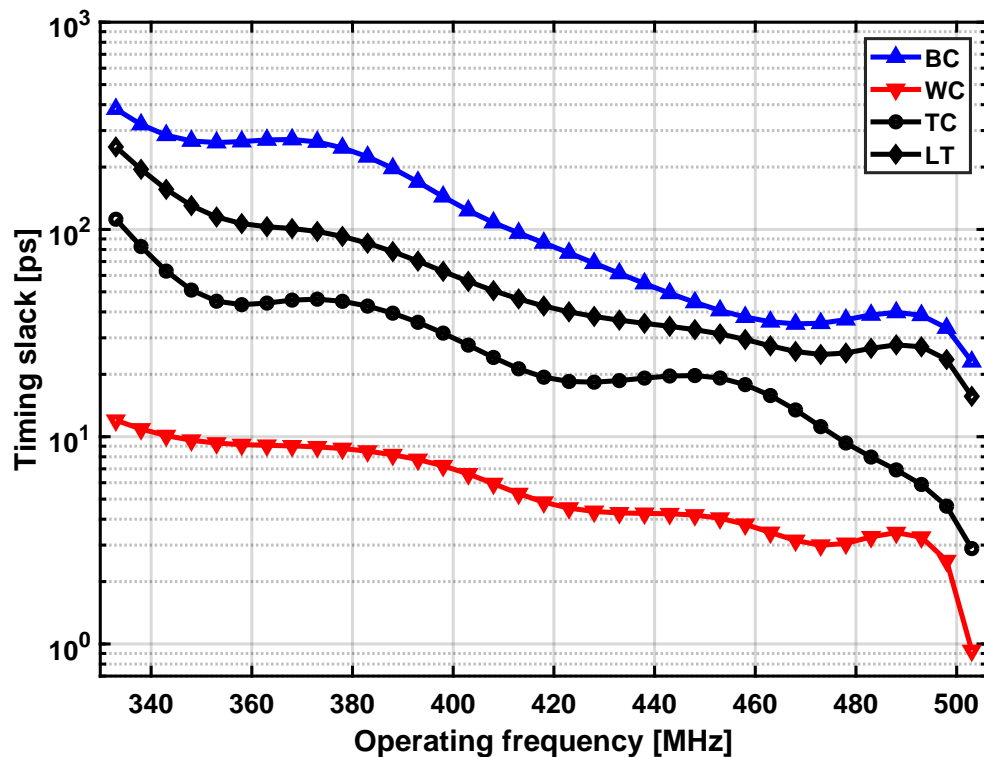
Figura 4. Functional behavior of the proposed architecture of 128b/132b block aligner.



### 1.3. RESULTS

The proposed architecture is synthesized in a CMOS standard  $0.18\mu\text{m}$  technology. The clock conditions taken into account for the synthesis process are: 100ps of jitter, 50ps of transition time and a 500pF load capacitance. To validate the design, the time slack of the proposed architecture is measured to identify the maximum operation frequency. The time slack corresponds to the excess time that a circuit has before the maximum operation frequency is achieved. Figure 5 shows the Parasitic-Interconnect Corner (RC) cases. The maximum operating frequency is 505MHz for the worst-case.

Figura 5. RC corners of proposed architecture of 128b/132b block aligner decoder.



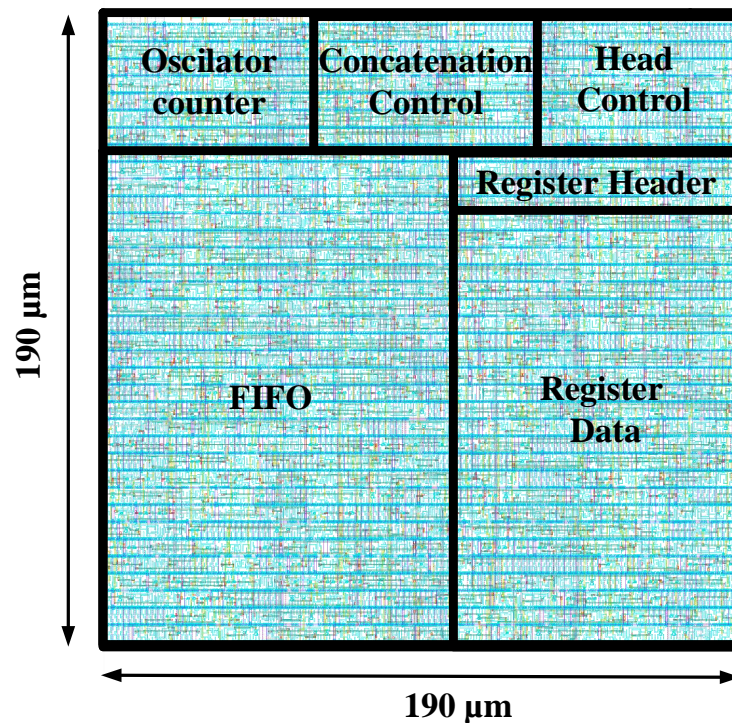
Additionally, results of the synthesis process are shown in Table II to compare the

proposed architecture with recent work architectures. A 36 % area reduction is achieved and the maximum operation frequency increases 48 %. However, the concatenation control technique increases the dynamic power consumption in 28 % due to a higher number of output registers. Also, Figure 6 shows the layout of the decoder that has an area of  $190 \times 190 \mu m^2$ .

Tabla 2. Summary results of the proposed 128b/132b block aligner

|                                    | <b>Tovar</b>    | <b>This work</b>                 |
|------------------------------------|-----------------|----------------------------------|
| <b>Area [<math>\mu m^2</math>]</b> | 260x260         | 190x190                          |
| <b>Technology [nm]</b>             | 180             | 180                              |
| <b>Power [mW]</b>                  | 77.26@ 312.5MHz | 99.12 @312.5MHz<br>128.6 @416MHz |
| <b>Required frequency [MHz]</b>    | 312.5           | 416                              |
| <b>Maximum frequency [MHz]</b>     | 341             | 505                              |
| <b># Cells</b>                     | 1560            | 620                              |
| <b># Bits input data</b>           | 32              | 24                               |

Figura 6. Layout of the proposed 128b/132b block aligner decoder in CMOS  $0.18 \mu m$  technology.



## **2. ELASTIC BUFFER**

The elastic buffer receives 128 data bits, and four header bits of the 128b/132b block aligner decoder with a sequence of 5-6 clock cycles. The main function of this module is to synchronize the data frame in both clock domains (Recovery Symbol Clock and Core Clock). This section focuses on presenting the new proposed architecture for the elastic buffer, taking into account the restrictions of the 128b/132b decoder's architecture.

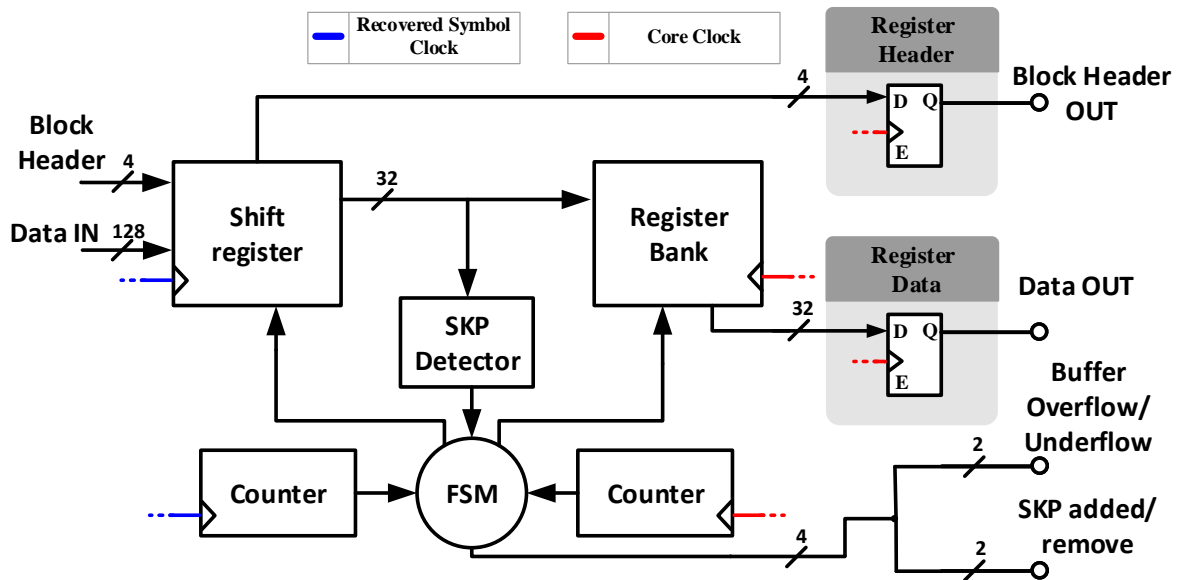
### **2.1. STATE OF THE ART**

An elastic buffer supports the frequency variations of the two clock domains presented in the physical layer of the USB 3.1 receiver. The accuracy of each clock domain reference is required to be within  $\pm 300\text{ppm}$ . Thus, the total magnitude variation of the frequency can range from -5300 to 300ppm. This frequency variation is managed by an elastic buffer that consumes or inserts SKP ordered sets to avoid data corruption.

In the literature, a conventional architecture of the elastic buffer, as presented in Fig. 7, is mainly conformed by a Shift register, an FSM and a Register Bank. The Shift register is in charge of serializing the input bits in smaller information packets. These packets are synchronized in the recovered symbol clock domain and they go through an SKP detector to report the presence of SKP symbols in the information packet. Meanwhile, in the Core Clock domain, an FSM indicates to an FIFO if an SKP symbol

must be added or removed in the information packet <sup>789</sup>. Also, the FSM measures the phase difference of the clock domains through counters.

Figura 7. A conventional architecture of the elastic buffer.



## 2.2. PROPOSED ARCHITECTURE

The approach of the proposed architecture of the elastic buffer is to reduce the area occupied and to maximize the operation frequency with respect to the 128b/132b block aligner decoder. As mentioned before, the data frames arrive at the input of the

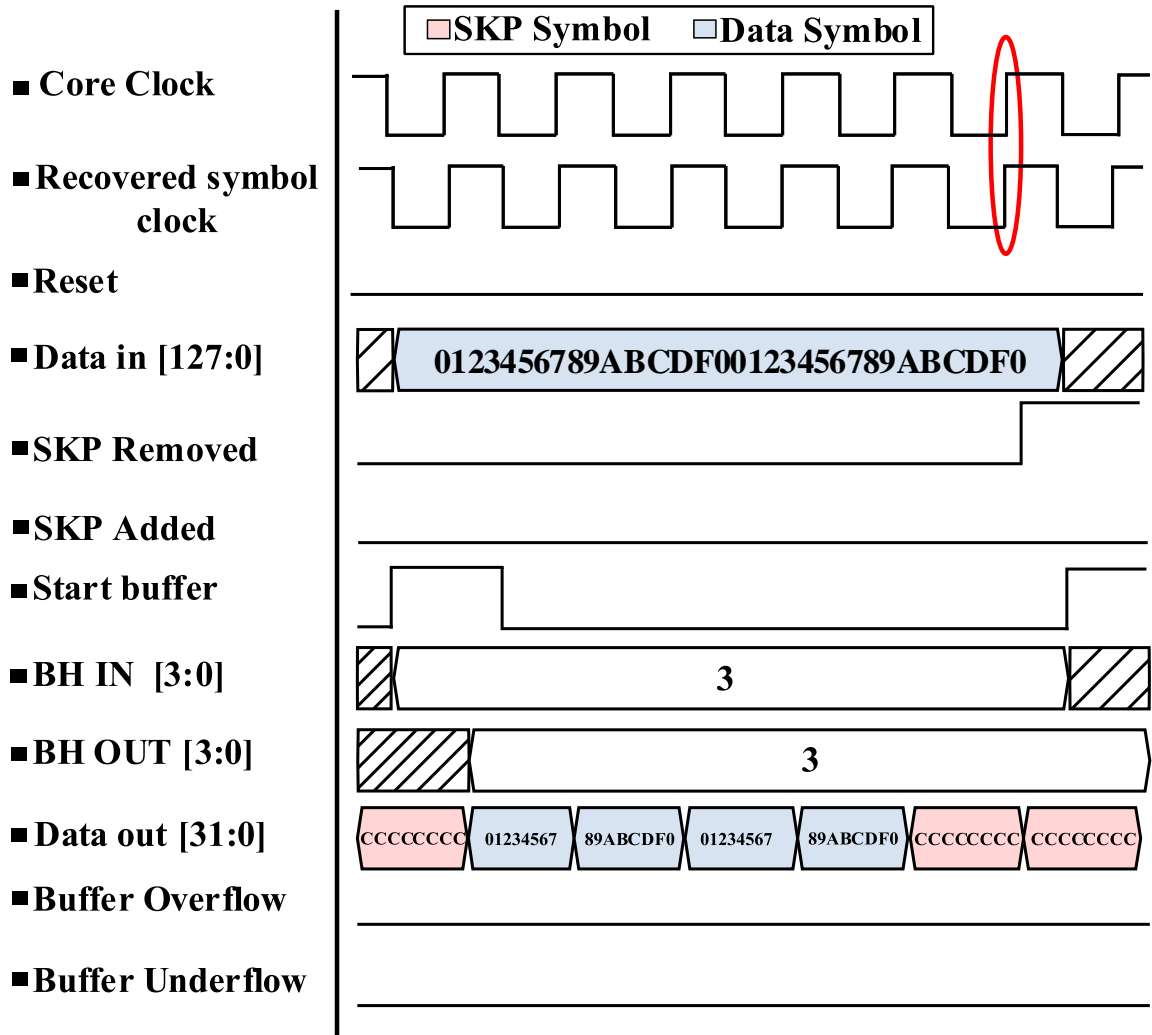
<sup>7</sup> Ioannis SEITANIDIS y col. "ElastiStore: Flexible Elastic Buffering for Virtual-Channel-Based Networks on Chip". En: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 23.12 (2015), págs. 3015-3028.

<sup>8</sup> George MICHELOGIANNAKIS y DALLY, William. "Elastic Buffer Flow Control for On-Chip Networks". En: *IEEE Transactions on Computers* 62.2 (2013), págs. 295-309.

<sup>9</sup> Sergio Andrés. TOVAR DURÁN. "DECODIFICADOR 132b/128b COMPATIBLE CON USB3.1 EN TECNOLOGÍA CMOS 180NM". En: *Universidad Industrial de Santander*. 2017.



Figura 9. Functional behavior of the proposed elastic buffer.



### 2.3. RESULTS

The results of the proposed architecture are synthesized in a standard CMOS 0.18 $\mu$ m technology. Figure 10 shows the RC corners, indicating that the maximum operating frequency is 454MHz. The clock characteristics for the simulation are the same presented in the 128b/132b block aligner results.

Figura 10. RC corners of proposed architecture of the elastic buffer.

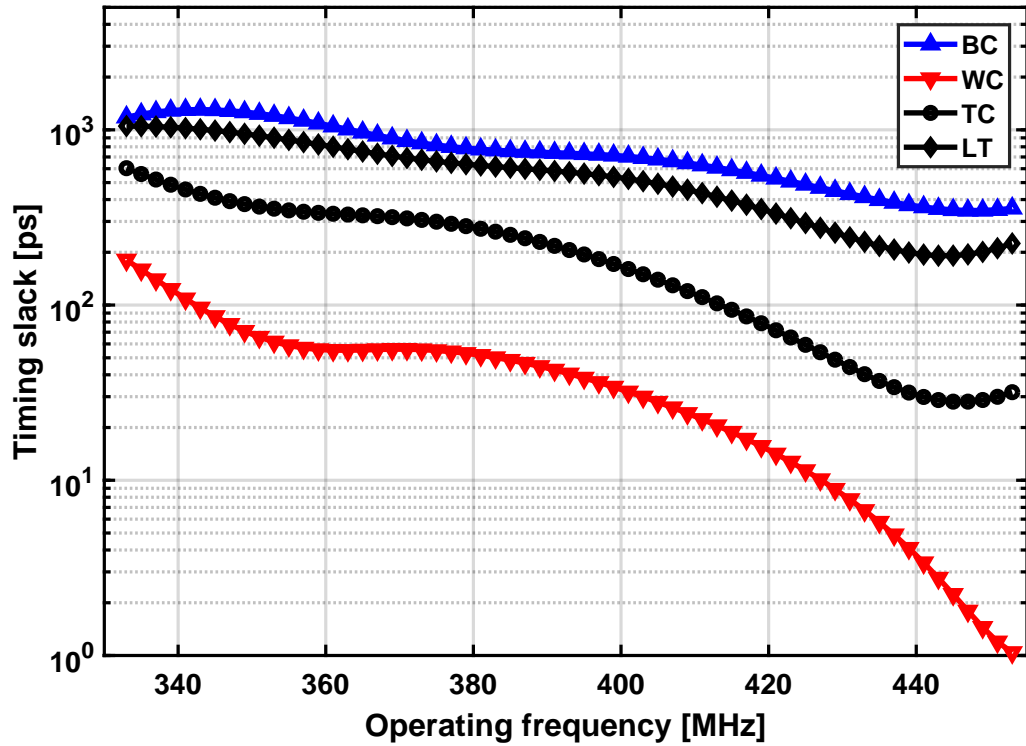


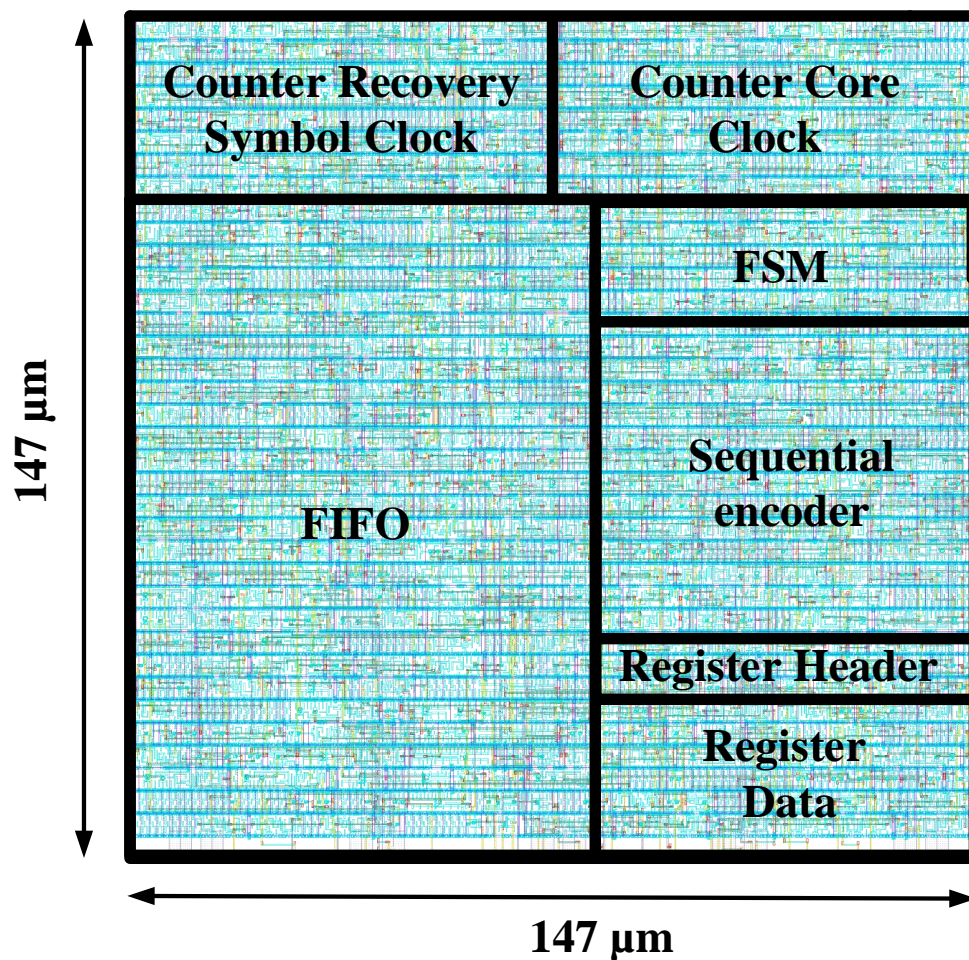
Tabla 3. Summary results of the proposed elastic buffer

|                                    | <b>Tovar</b>    | <b>This work</b>                 |
|------------------------------------|-----------------|----------------------------------|
| <b>Area [<math>\mu m^2</math>]</b> | 262x262         | 147x147                          |
| <b>Technology [nm]</b>             | 180             | 180                              |
| <b>Power [mW]</b>                  | 89.09@ 312.5MHz | 27.98 @312.5MHz<br>37.93 @416MHz |
| <b>Required frequency [MHz]</b>    | 312.5           | 416                              |
| <b>Maximum frequency [MHz]</b>     | 318             | 454                              |
| <b># Cells</b>                     | 1282            | 372                              |
| <b># Bits output data</b>          | 32              | 32                               |

Figure 11 shows the elastic buffer layout, which occupies an area of  $147 \times 147 \mu\text{m}^2$ . Finally, Table 3 shows the results of the area, power, and operating frequency of the conventional architecture presented and the proposed architecture.

As observed in Table 3, reductions of 78 % in occupied area, 31 % in power consumption and an increase of 42 % in the operation frequency are achieved.

Figura 11. Layout of the proposed elastic buffer in CMOS  $0.18 \mu\text{m}$  technology.



### 3. DESCRAMBLING

The transmitter pseudo-randomizes the data to generate enough changes for clock recovering in the receiver. Thus, the descrambling is in charge of reorganize the transmitted data . Then, it sends the descrambled data to the DMA. To design the descrambling block, the standard USB 3.1 proposes a Linear Feedback Shift Register (LFSR) to implement the descrambling. The behavior of this block is defined by a characteristic polynomial  $G(x)$ , as presented in equation 1. The seed used to initialize the circuit is 1DBFBC'h.

$$G(x) = x^{23} + x^{21} + x^{16} + x^8 + x^5 + x^2 + 1 \quad (1)$$

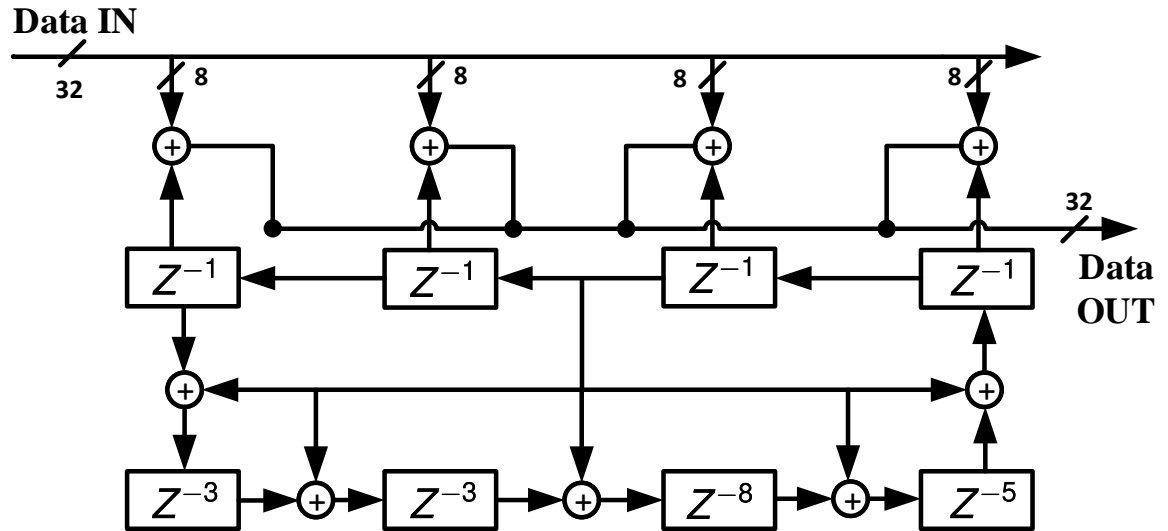
The descrambling has three modes of operation, as presented in<sup>10</sup>. In the first mode, the input data is XORed with the sequence coming from the LFSR. In the second mode, the input data is not XORed and the sequence from the LFSR advances to the future state. Finally, in the third mode, the input data is not XORed and the sequence from the LFSR does not advance.

As mentioned before, the USB 3.1 standard defines the LFSR as a sequential block that only processes 1 byte per cycle. However, based on Fig. 12, the descrambling requires an operation frequency 4 times larger than the operation frequency of the physical layer to reorganize the 32 input bits. Thus, achieving this larger frequency is not feasible with a low-cost technology. Therefore, this block is out of scope of this work.

---

<sup>10</sup> HEWLETT-PACKARD COMPANY. *Universal Serial Bus 3.1 specification, revision 1.0*. USB-IF, July 26, 2013.

Figura 12. Proposed block diagram of the descrambling.



A possible implementation using the schematic proposed from the USB 3.1 standard is to use more LFSRs in parallel for creating the sequence of the characteristic polynomial. However, instead of achieving 4 future states from 4 present states, only 1 present state evolves to a future state and the 3 remaining present states does not evolve due to the sequential logic. To solve this issue, the LFSR should be implemented in a ladder architecture.

## 4. CONCLUSIONS

- In this paper, we presented a partial digital-physical layer compatible with USB 3.1 receiver synthesized in a CMOS standard  $0.18\mu\text{m}$  technology with an operation frequency of 416.66MHz. Compare to a conventional architecture, the proposed architecture of the 128b/132b block aligner reduces 36% the occupied area, the operation frequency increases 48%, and the power consumption increases 28%. For the elastic buffer architecture, the occupied area and the power consumption are reduced 78% and 31%, respectively. Additionally, the operation frequency is increased 42%. Based on simulation results, the descrambling lies outside the scope of this work due to the necessity of using a larger operation frequency than the maximum one achieved by the low-cost technology.

## BIBLIOGRAFÍA

HEWLETT-PACKARD COMPANY. *Universal Serial Bus 3.1 specification, revision 1.0*. USB-IF, July 26, 2013 (vid. págs. 11, 25).

HEWLETT-PACKARD COMPANY y col. *Universal Serial Bus 3.0 specification, revision 1.0*. USB-IF, November 12, 2008 (vid. pág. 10).

MALDAL, Rakeshroshan y PHILEMON, Daniel. "A Study to Implement Superspeed Plus USB 3.1 Physical Layer." En: *Proceedings of ASAR-IJIEEE International Conference*. 2014 (vid. pág. 12).

MICHELOGIANNAKIS, George y DALLY, William. "Elastic Buffer Flow Control for On-Chip Networks". En: *IEEE Transactions on Computers* 62.2 (2013), págs. 295-309 (vid. pág. 20).

MONIKA, Shah y SHRUTHI, G. "Study and Implementation of Physical Layer Coding Used in Super Speed USB". En: *International Journal of Scientific Engineering and Research*. 2014 (vid. pág. 12).

MORALES, Hanssel, DURAN, Ckristian y ROA, Elkim. "A Low-Area Direct Memory Access Controller Architecture for a RISC-V Based Low-Power Microcontroller". En: *2019 IEEE 10th Latin American Symposium on Circuits Systems (LASCAS)*. 2019, págs. 97-100 (vid. pág. 11).

SEITANIDIS, Ioannis y col. "ElastiStore: Flexible Elastic Buffering for Virtual-Channel-Based Networks on Chip". En: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 23.12 (2015), págs. 3015-3028 (vid. pág. 20).

TOVAR DURÁN, Sergio Andrés. "DECODIFICADOR 132b/128b COMPATIBLE CON USB3.1 EN TECNOLOGÍA CMOS 180NM". En: *Universidad Industrial de Santander*. 2017 (vid. págs. 12, 20).