Escape Encoder

Induction Training

Version 1.0

Block Owner

Si-Vision

Authors

Amira Khaled

This Page is left Blank Intentionally

1 Table of Contents

2 Revision History

Version	Date	Author(s)	Revision Notes	Owner Approval
1.0				

3 Overview

The Esc_Encoder module is responsible for generating Escape (ESC) sequences for the MIPI C-PHY transmitter during low-power or control signaling modes for escape mode. It encodes binary control or data bits (EscBit) into line states across the C-PHY lanes A, B, and C in compliance with the ESC signaling protocol. This encoder operates under the control of an ESC domain clock (TxClkEsc) and is enabled via EscEncodeEn.

4 Operation and Description

4.1 Digital Interface

4.1.1 Parameters Names

Parameter Name	Default	Description
None	-	-

4.1.2 Ports Names

Port Name	Port Width	Port Type	Description
TxClkEsc	1	Input	ESC domain clock
RST	1	Input	Active-low reset
EscEncodeEn	1	Input	Escape encoding enables
EscBit	1	Input	Data bit to be encoded
DataValid	1	Input	Indicates when EscBit is valid and should be encoded
A	1	Output	Output line A for ESC sequence
B	1	Output	Output line B for ESC sequence
C	1	Output	Output line C for ESC sequence

4.1.3 CDC Table

CDC signal	Source Domain	Destination Domain	Synchronization method
EscEncodeEn	TxWordClk	TxClkEsc	2-stage flip-flop synchronizer
DataValid	TxWordClk	TxClkEsc	2-stage flip-flop synchronizer

4.2 Functional Description

This block provides Return-to-Zero encoding in which data is valid on the positive half of the clock and return to zero at the second half of the clock period so the ESC encoder works based on a dual-edge control scheme involving two key internal enable signals:

- Enable1: Toggles on the positive edge of TxClkEsc when encoding is enabled.
- Enable0: Toggles on the negative edge of TxClkEsc when encoding is enabled.

The XOR of these two signals (Enable1 ^ Enable0) determines when the encoded data is valid and should be driven to the outputs.

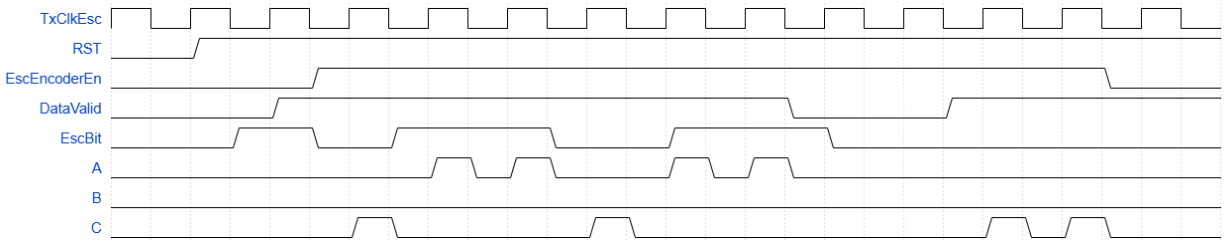
When the RST = 0, the encoder outputs will be 0 even if the enable is asserted.

After the RST = 1, Escape Encoder starts to work under these conditions:

1- If EscEncoderEn = 0, the output will be kept 0 and ignore any change on the EscBit
2- If EscEncoderEn = 1 and DataValid = 1 at the same time, The EscBit will be sampled at the positive edge of the TxClkEsc and stored.
3- If EscEncoderEn = 1 and DataValid = 0 at the same time, The EscBit will be ignored and the stored value will be 0 (This case indicates sending space on the 3 lines A = 0, B = 0 and C = 0 as the low power data isn't available at this moment but we still in the low power mode waiting for the data or the stop condition)

When Enable1 ^ Enable0 is true then the output line states are driven with the stored data captured at the positive edge else the output line states are 0.

4.3 Timing Diagram



4.4 Verification Requirements

- **Reset behavior**
  Ensures that all internal registers (data_A, data_C, Enable0, Enable1) and outputs (A, B, C) are properly reset when RST is asserted low.
- **Valid Data Transmission**
  Sends valid EscBit with EscEncodeEn and DataValid high, and verifies that A = EscBit, C = ~EscBit, and B = 0 when Enable1 ^ Enable0 is true.
- **Space Transmission (Idle)**
  Sends data with EscEncodeEn high and DataValid low, and verifies that all outputs (A, B, C) are 0 when Enable1 ^ Enable0 is true.
- **No Output When Disabled**
  Verifies that outputs remain zero when EscEncodeEn is low, regardless of DataValid or EscBit.
- **Data Integrity Check**
  Ensures that C is always the inverse of A when data is valid (DataValid = 1), confirming proper encoding.