ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

IEEE *Draft* P802.3cg/D3.0 Delimiter Randomization

Proposal relating to comment i-284

MICK MCCARTHY, ANALOG DEVICES
STEFFEN GRABER, PEPPERL+FUCHS

22 MAY 2019

SSD4, ESD4 and ESD_ERR4

- In IEEE P802.3cg™/D3.0 the frame delimiters are defined in Table 146-3 as follows:

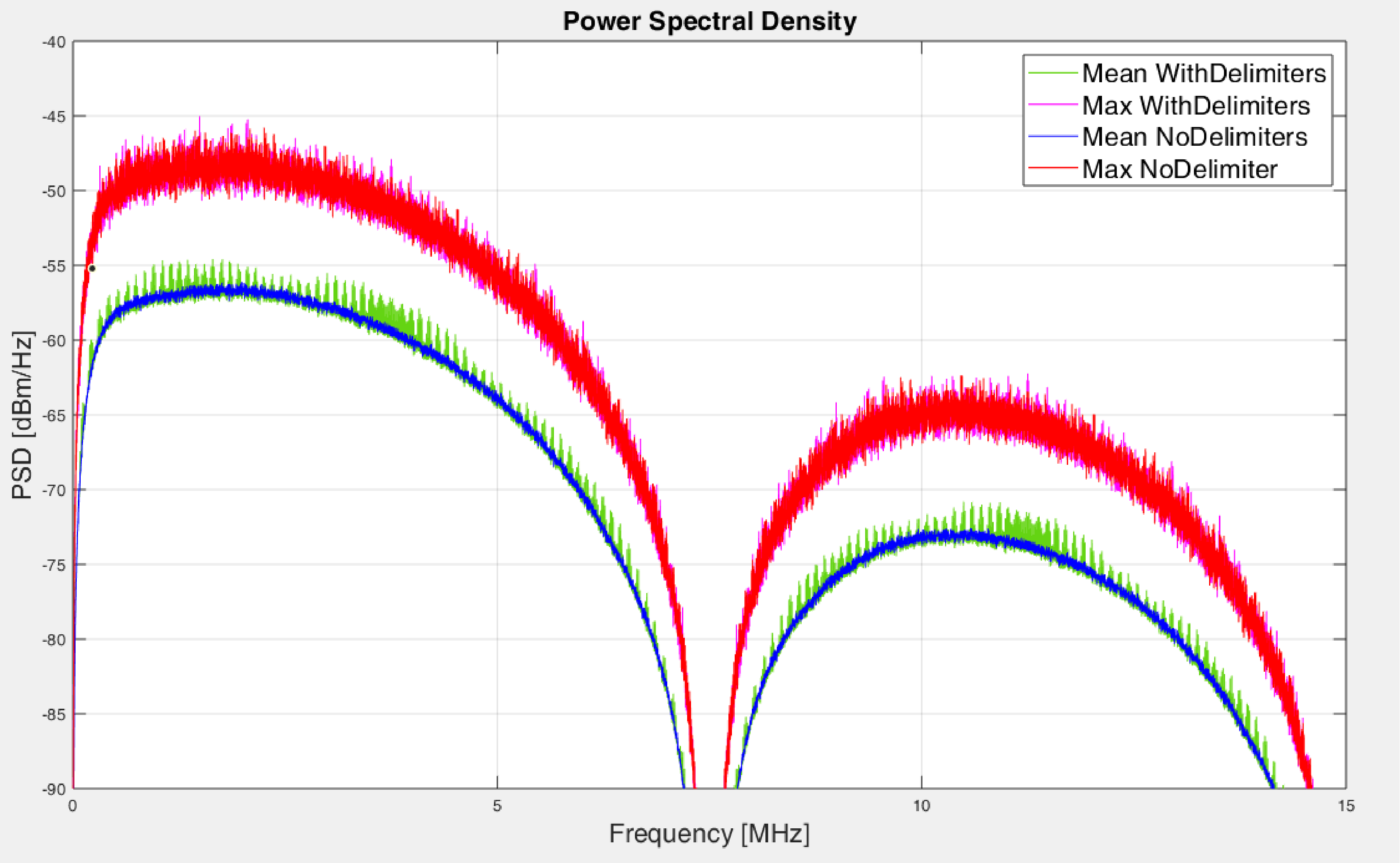
Delimiter	(TA_n, TB_n, TC_n)
SSD4	(1, 1,-1)
ESD4	(1,-1, 1)
ESD_ERR4	(-1, 1, 1)

- Note that the delimiters are constant
  - When frames of fixed length are continuously transmitted using a fixed interframe gap harmonics appear in the transmitted power spectrum
  - Adaptive processes may align with the constant ternary symbols of the delimiters potentially causing filter coefficients to become misadjusted

Simulation of Transmitted Power Spectrum

- Simulated pattern with delimiters consists of 16 frames with 64 bytes of data and 12 bytes of interframe gap
- Simulated pattern without delimiters uses IDLE encoding
- Pattern is 8x oversampled and PSD is estimated using Welch method
- Process is repeated 500 times using different seed values for the random number generator
- Harmonics appear in the computed PSD at about 2dB above the level without delimiters
- The harmonics in the PSD are associated with periodic non-zero values in the auto-correlation sequence which may cause adaptive filter misadjustment over time
  - For example, the optimal set of echo canceler coefficients depends on the auto-correlation matrix of the transmitted symbol stream
  - Data dependent artefacts in the auto-correlation sequence may cause the echo canceler coefficients to move

Computed PSD Using Existing Delimiters



Overview of Proposed Solution

- Propose to generate an additional random bit $Sy_n[4]$ in 146.3.3.2.2 as follows:

$$Sy_n[4] = g^4(Scr_n[0]) = Scr_n[12] \wedge Scr_n[32]$$

- When generating the delimiter use $Sy_{n-1}[4]$ to randomly determine its sign
- When generating DISPRESET3 use $Sy_n[4]$ to determine whether to bring the disparity after the transmission of the delimiter to 2 or to 3
  - By doing this can achieve symmetry amongst the DISPRESET3 ternary triplets

Proposed Delimiter Encoding

- Propose to replace Table 146-3 with the following:

	Delimiter	(TA_n, TB_n, TC_n)
$Sy_{n-1}[4] = 0$	SSD4	(1, 1, -1)
	ESD4	(1, -1, 1)
	ESD_ERR4	(-1, 1, 1)
$Sy_{n-1}[4] = 1$	SSD4	(-1, -1, 1)
	ESD4	(-1, 1, -1)
	ESD_ERR4	(1, -1, -1)

- Ternary triplets as in current draft when $Sy_{n-1}[4] = 0$
- Ternary triplets negated when $Sy_{n-1}[4] = 1$

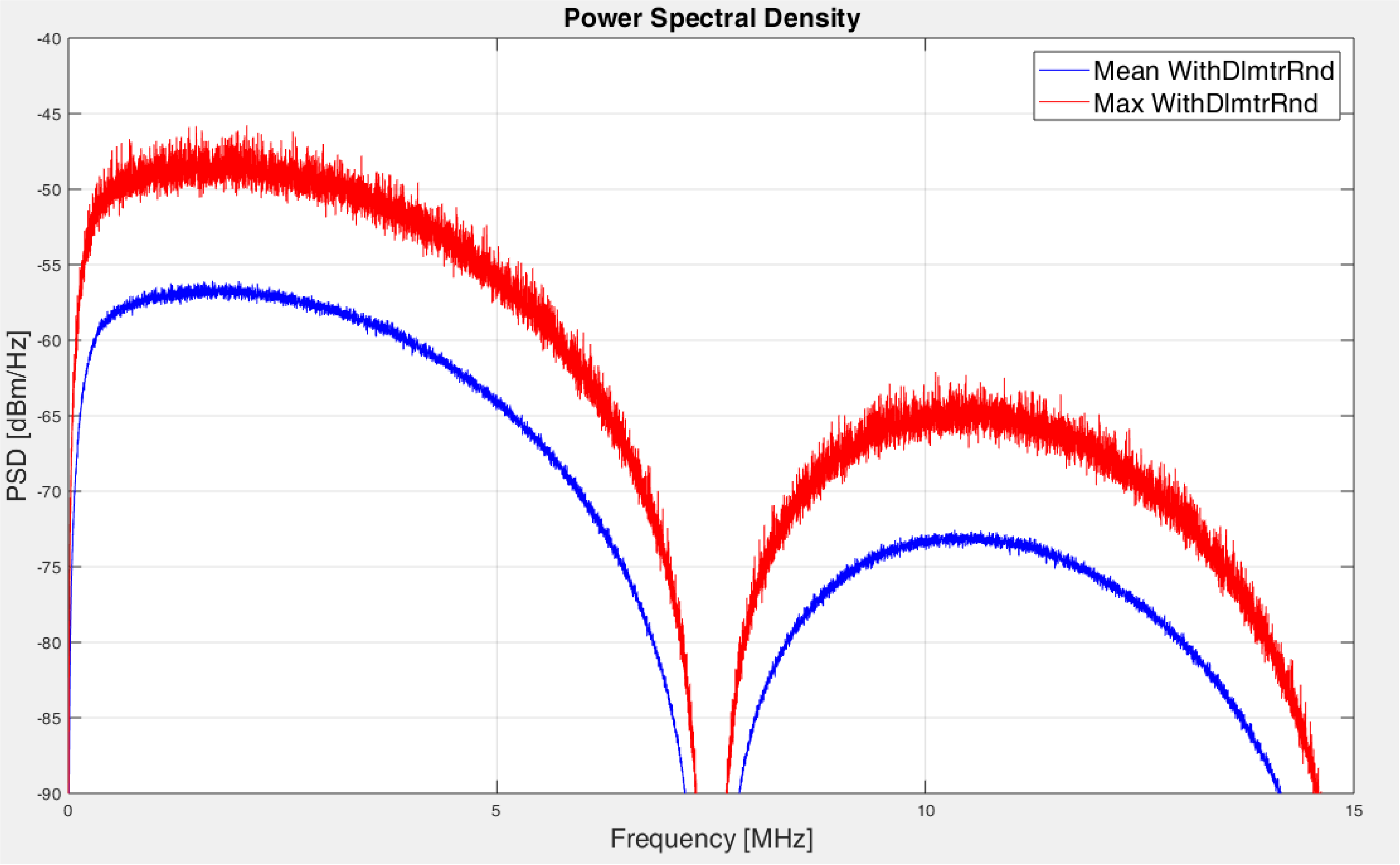
Proposed Disparity Reset Encoding

- Propose to replace Table 146-2 with the following:

DISPRESET3	disparity = 1	disparity = 2	disparity = 3	disparity = 4
$Sy_n[4] = 0$	(-1, 0, 1)	(-1, 0, 0)	(-1, 0, -1)	(-1, -1, -1)
$Sy_n[4] = 1$	(1, 1, 1)	(1, 0, 1)	(1, 0, 0)	(1, 0, -1)

- Disparity brought to 2 after transmission of delimiter when $Sy_n[4] = 0$
- Disparity brought to 3 after transmission of delimiter when $Sy_n[4] = 1$
- Ternary triplets for disparity 1 and disparity 4 are symmetrical
- Ternary triplets for disparity 2 and disparity 3 are symmetrical

Computed PSD Using Randomized Delimiters



ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

Detailed Changes to 146.3.3.1.1 Variables

- Add new variable $Sy_n[4:0]$ just before $Sd_n[3:0]$ as follows:

  $Sy_n[4:0]$

  The $Sy_n[4:0]$ bits from the scrambler as defined in 146.3.3.2.2.

Detailed Changes to 146.3.3.1.2 Functions

- Change DISPRES function definition as highlighted:

The function DISPRES returns one of the eight possible DISPRESET3 triple ternary symbols (see Table 146–2), depending on the values of $Sy_n[4]$ and tx_disparity:

$$tx\_symb\_triplet = table_{DISPRESET3}(Sy_n[4],\ tx\_disparity)$$

Detailed Changes to 146.3.3.1.2 Functions

- Add following new function:

RND_SSD4

The function RND_SSD4 takes $Sy_{n-1}[4]$ as its argument and returns the corresponding tx_symb_triplet as well as the updated tx_disparity.

$(tx\_symb\_triplet\ ,\ tx\_disparity) = RND\_SSD4(Sy_{n-1}[4])$

The returned tx_symb_triplet corresponds to one of the two possible SSD4 triple ternary symbols (see Table 146–3), depending on the value of $Sy_{n-1}[4]$:

$tx\_symb\_triplet = table_{SSD4}(Sy_{n-1}[4])$

The returned tx_disparity also depends on the value of $Sy_{n-1}[4]$ as follows:

$$tx\_disparity = \begin{cases} 2 & \text{if } Sy_{n-1}[4] = 0 \\ 3 & \text{else} \end{cases}$$

Detailed Changes to 146.3.3.1.2 Functions

- Add following new function:

RND_ESD4

The function RND_ESD4 takes $Sy_{n-1}[4]$ as its argument and returns the corresponding tx_symb_triplet as well as the updated tx_disparity.

$(tx\_symb\_triplet , tx\_disparity) = RND\_ESD4(Sy_{n-1}[4])$

The returned tx_symb_triplet corresponds to one of the two possible ESD4 triple ternary symbols (see Table 146–3), depending on the value of $Sy_{n-1}[4]$:

$tx\_symb\_triplet = table_{ESD4}(Sy_{n-1}[4])$

The returned tx_disparity also depends on the value of $Sy_{n-1}[4]$ as follows:

$$tx\_disparity \quad = 2 \quad \text{if } Sy_{n-1}[4] = 0$$
$$= 3 \quad \text{else}$$

Detailed Changes to 146.3.3.1.2 Functions

- Add following new function:

RND_ESD_ERR4

The function RND_ESD_ERR4 takes $Sy_{n-1}[4]$ as its argument and returns the corresponding tx_symb_triplet as well as the updated tx_disparity.

$(tx\_symb\_triplet , tx\_disparity) = RND\_ESD\_ERR4(Sy_{n-1}[4])$

The returned tx_symb_triplet corresponds to one of the two possible ESD_ERR4 triple ternary symbols (see Table 146–3), depending on the value of $Sy_{n-1}[4]$:

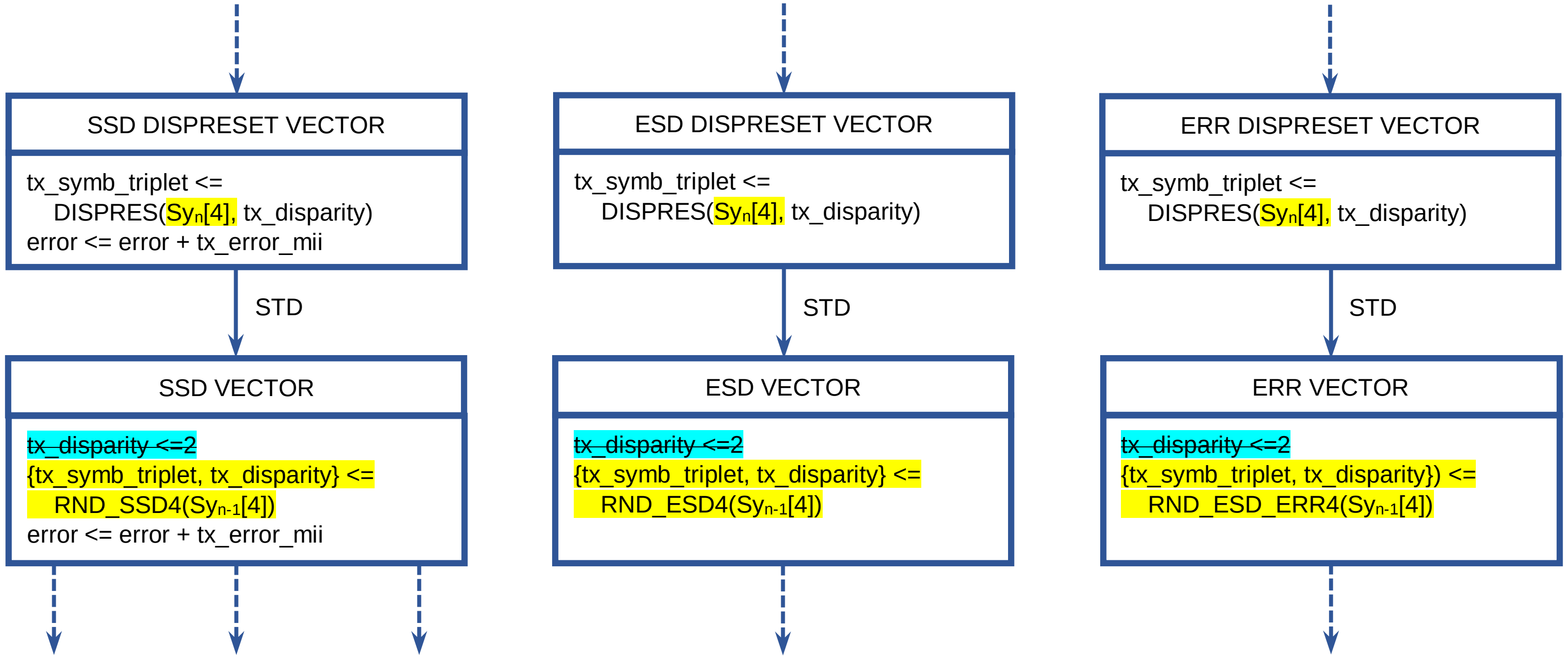
$tx\_symb\_triplet = table_{ESD\_ERR4}(Sy_{n-1}[4])$

The returned tx_disparity also depends on the value of $Sy_{n-1}[4]$ as follows:

$$tx\_disparity = \begin{cases} 2 & \text{if } Sy_{n-1}[4] = 0 \\ 3 & \text{else} \end{cases}$$

Detailed Changes to Figure 146-5 - PCS Transmit State Diagram

- Dashed arrows come from and go to unchanged parts of the state diagram
- Modified state diagram uses the modified DISPRES function and the new RND_SSD4, RND_ESD4 and RND_ESD_ERR4 functions



Detailed Changes to 146.3.3.2.2 Generation of $Sy_n[3:0]$

- Modify as highlighted:

PCS Transmit encoding rules are based on the generation, at time n, of the five bits $Sy_n[4:0]$. The four bits $Sy_n[3:0]$ are used for de-correlating the MII data word TXD<3:0> during data transmission and for generating the idle symbols. The bit $Sy_n[4]$ is used to randomize the frame delimiters. These five bits are generated as described below, using the auxiliary generating polynomial, g(x) defined in Equation (146–3):

$$g(x) = x^3 \wedge x^8 \quad (146–3)$$

The five bits $Sy_n[4:0]$ shall be generated using the bit $Scr_n[0]$ and $g(x)$ as in the following equations:

$$Sy_n[0] = Scr_n[0]$$
$$Sy_n[1] = g(Scr_n[0]) = Scr_n[3] \wedge Scr_n[8]$$
$$Sy_n[2] = g^2(Scr_n[0]) = Scr_n[6] \wedge Scr_n[16]$$
$$Sy_n[3] = g^3(Scr_n[0]) = Scrn[9] \wedge Scr_n[14] \wedge Scr_n[19] \wedge Scr_n[24]$$
$$Sy_n[4] = g^4(Scr_n[0]) = Scr_n[12] \wedge Scr_n[32]$$

By construction, the five bits $Sy_n[4:0]$ are derived from elements of the same maximum-length shift register sequence of length $2^{33}–1$ as $Scr_n[0]$, but shifted in time by varying delays. The associated delays are all large and different so that there is no apparent correlation among the bits.

Detailed Changes to 146.3.3.2.4 Generation of Ternary Triplet in Mode SEND_N and SEND_I

- Change the third and fourth paragraphs as highlighted below:

The DISPRESET3 triplet, together with the following fourth symbol group ~~(which always has a disparity of 1),~~ shall be used to bring back the running disparity to a defined value of either 2 or 3, depending on the value of the bit $Sy_n[4]$ from the scrambler. The coding shown in Table 146-2 shall be used for the DISPRESET3 symbol triplet.

The fourth symbol group (SSD4/ESD4/ESD_ERR4) shall be encoded as shown in Table 146–3. ~~(all have disparity of +1):~~

Detailed Changes to 146.3.3.2.6 Generation of Symbol Sequence

- Replace Table 146-2 with the following table:

DISPRESET3	Disparity = 1	Disparity = 2	Disparity = 3	Disparity = 4
$Sy_n[4] = 0$	(-1, 0, 1)	(-1, 0, 0)	(-1, 0, -1)	(-1, -1, -1)
$Sy_n[4] = 1$	(1, 1, 1)	(1, 0, 1)	(1, 0, 0)	(1, 0, -1)

Detailed Changes to 146.3.3.2.6 Generation of Symbol Sequence

- Replace Table 146-3 with the following table:

	Delimiter	(TA_n,TB_n,TC_n)
$Sy_{n-1}[4] = 0$	SSD4	(+1,+1,-1)
	ESD4	(+1,-1,+1)
	ESD_ERR4	(-1,+1,+1)
$Sy_{n-1}[4] = 1$	SSD4	(-1,-1,+1)
	ESD4	(-1,+1,-1)
	ESD_ERR4	(+1,-1,-1)

Detailed Changes to 146.3.4.1.2 Functions

- Modify valid_dispreset function definition as highlighted:

valid_dispreset

Determines if the rx_symb_triplet is one of the DISPRESET3 triplets as specified in 146.3.3.2.4. It returns a Boolean value indicating whether or not one of the eight possible DISPRESET3 triplets has been received.

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

Detailed Changes to 146.3.4.1.2 Functions

- Add the following new functions:

valid_ssd4

Determines if the rx_symb_triplet is one of the SSD4 triplets as specified in 146.3.3.2.4. It returns a Boolean value indicating whether or not one of the two possible SSD4 triplets has been received.

valid_esd4

Determines if the rx_symb_triplet is one of the ESD4 triplets as specified in 146.3.3.2.4. It returns a Boolean value indicating whether or not one of the two possible ESD4 triplets has been received.

valid_esd_err4

Determines if the rx_symb_triplet is one of the ESD_ERR4 triplets as specified in 146.3.3.2.4. It returns a Boolean value indicating whether or not one of the two possible ESD_ERR4 triplets has been received.

Detailed Changes to 146.3.4.1.2 Functions

- Add the following new function:

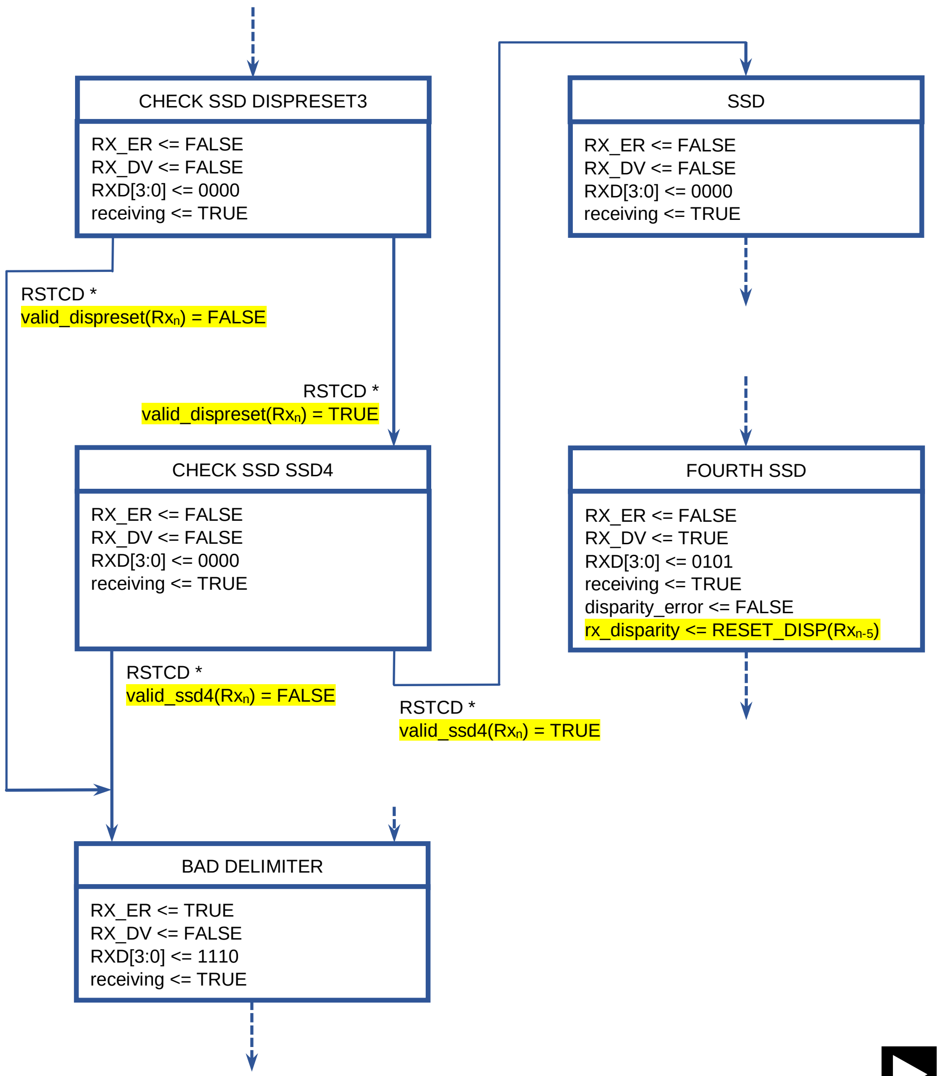
RESET_DISP

This function takes as its argument the value of Rx_n, corresponding to a valid SSD4 triplet, and returns the updated rx_disparity as follows:

$$rx\_disparity = \begin{cases} 2 & \text{if } Rx_n = (1, 1, -1) \\ 3 & \text{else} \end{cases}$$

Detailed Changes to Figure 146-8 - PCS Receive State Diagram (part a)

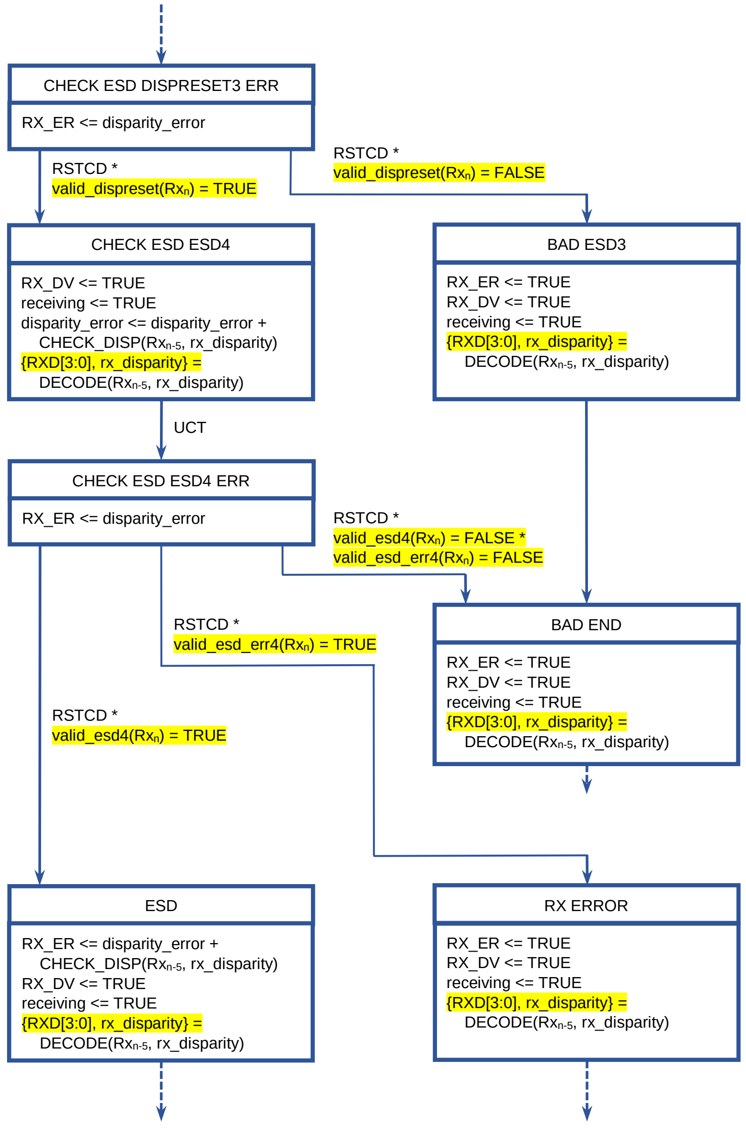
- Dashed arrows come from and go to unchanged parts of the state diagram
- Calls to function valid_dispreset modified to pass Rx_n as an argument
- State diagram modified to use the new valid_ssd4 and RESET_DISP functions
- Checking of Boolean return values modified to follow convention



ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

Detailed Changes to Figure 146-9 - PCS Receive State Diagram (part b)

- Calls to function valid_dispreset modified to pass Rx_n as an argument
- State diagram modified to use the new valid_esd4 and valid_esd_err4 functions
- Checking of Boolean return values modified to follow convention
- DECODE function calls modified to explicitly show assignment to variables



Thank you