mipi®
DEVCON

Manoj Sharma Tanikella, Sr. R&D Engineer)
Amitkumar Shrichand Gound, Sr. R&D Engineer)
Synopsys

Overcoming Inter-Symbol Interference with MIPI PHYs using Training Sequences

2017
MIPI ALLIANCE
DEVELOPERS
CONFERENCE

BANGALORE, INDIA
MIPI.ORG/DEVCON

Agenda

- Training sequence requirements
- ADAPT sequence in MIPI M-PHY®
- Alternate sequence in MIPI D-PHY℠
- Calibration sequences in MIPI C-PHY℠
- Summary

© 2017 MIPI Alliance, Inc.

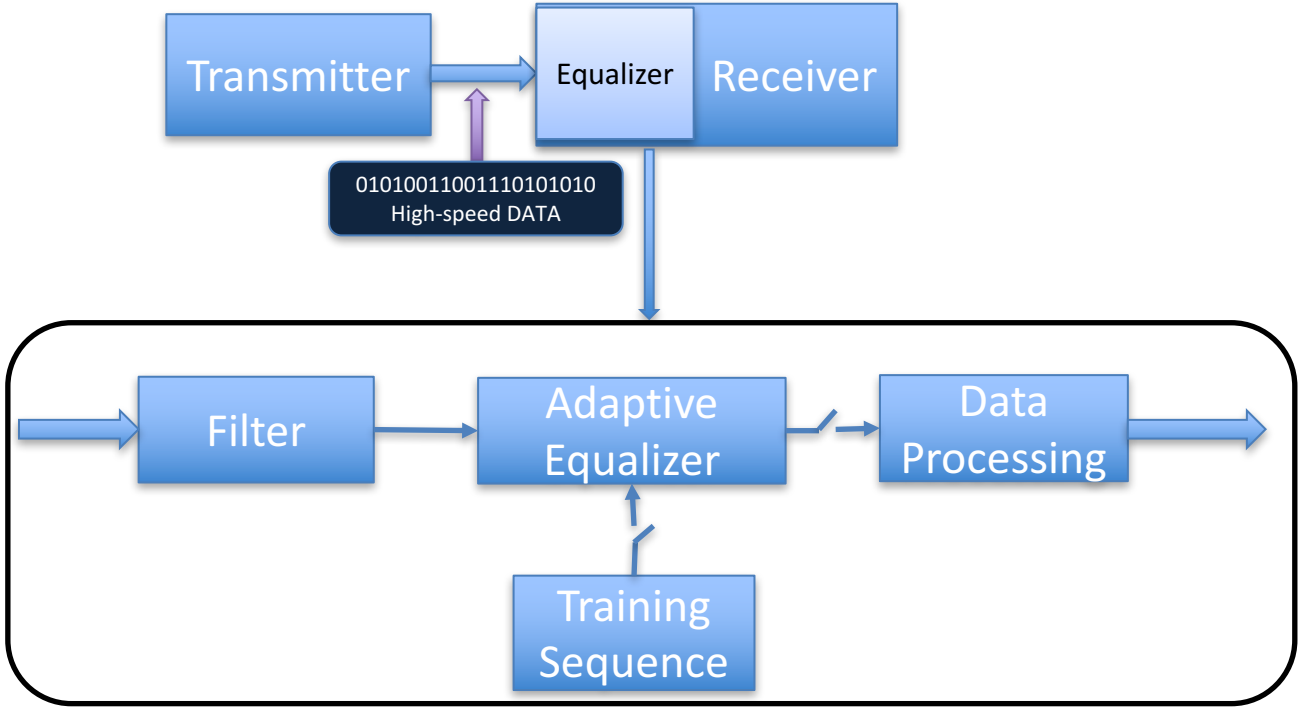
Training Sequence Requirements

- Overview on adaptive equalizer
- Purpose of training sequences

Synopsys

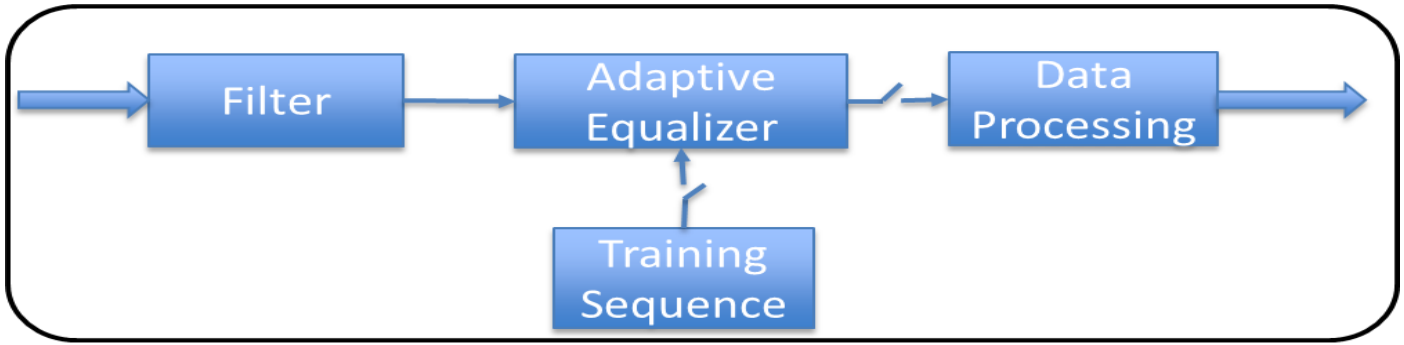
© 2017 MIPI Alliance, Inc.

Adaptive Equalizer Overview



Synopsys

Purpose of Training Sequences



- Bit patterns used in adaptive equalizer module are known as training sequences
- Receiver must realize sequences well in advance, so that receiver can compare and optimize its co-efficient for proper DATA reception
- System can send DATA once training sequence is complete
- Most commonly used training sequences are Pseudo Random Bit Sequences (PRBS)
- Example: PRBS9, PRBS32, etc.

Synopsys

© 2017 MIPI Alliance, Inc.

ADAPT Sequence in MIPI M-PHY

- What is ADAPT?
- Why and when ADAPT sequence is required?
- PRBS9 handling
- ADAPT operation
- Future scope

© 2017 MIPI Alliance, Inc.

What is ADAPT?

- The training sequence used to optimize the equalizer settings of the receiver, in this case M-RX, is know as ADAPT
- ADAPT was introduced in MIPI M-PHY v4.0 for higher data rate HS_G4 gear
- ADAPT consists of MK0 + Pseudo Random Bit Sequence + bit 1'b0
- PRBS with equation $X^9 + X^5 + 1$ (PRBS9) is used

© 2017 MIPI Alliance, Inc.

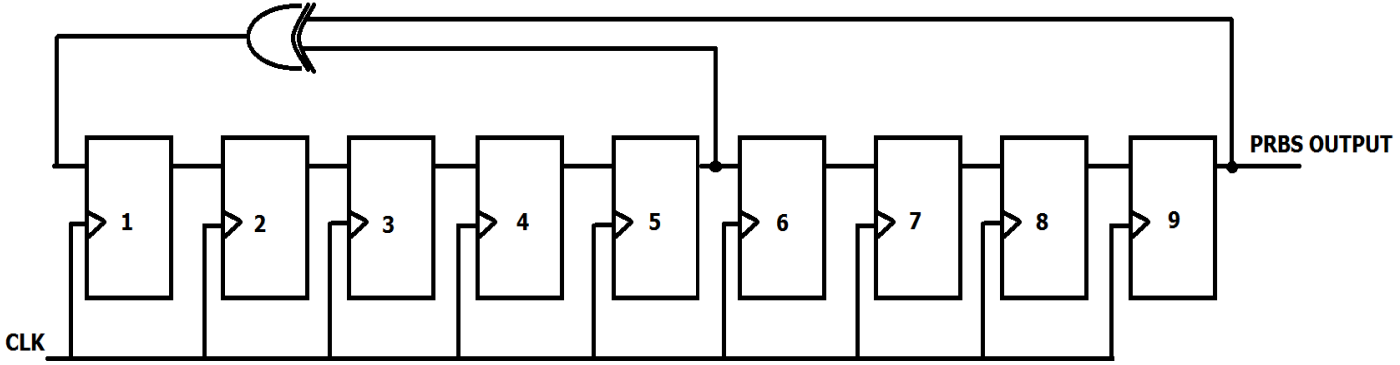
Why and When ADAPT Sequence is Required?

- As MIPI M-PHY started supporting higher speed HS_G4 gear, inter symbol interference and noise increased
- In-order to re-tune equalizer settings, ADAPT sequence was introduced as training sequence
- ADAPT sequence or ADAPT sub-state is used before starting with HS-G4 DATA
- ADAPT Sequence : PREPARE→ ADAPT PATTERN→ BURSTEND

© 2017 MIPI Alliance, Inc.

PRBS9 Handling

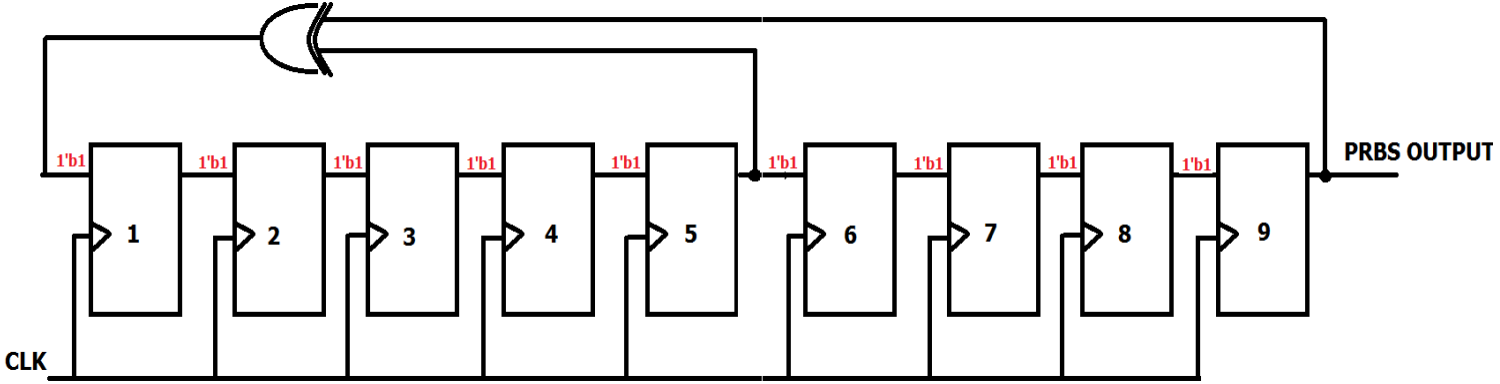
- PRBS9 is a combination of 9 linear-shift registers where output of 5th and 9th registers are xored and feedback is provided to input; Generates $2^9 - 1 = 511$ bits
- As per standard ITU-T recommendation O.150 and 0.153, pattern begins with first 1'b1 of 9 consecutive ONEs



Synopsys

© 2017 MIPI Alliance, Inc.

PRBS9 Handling (Continued…)

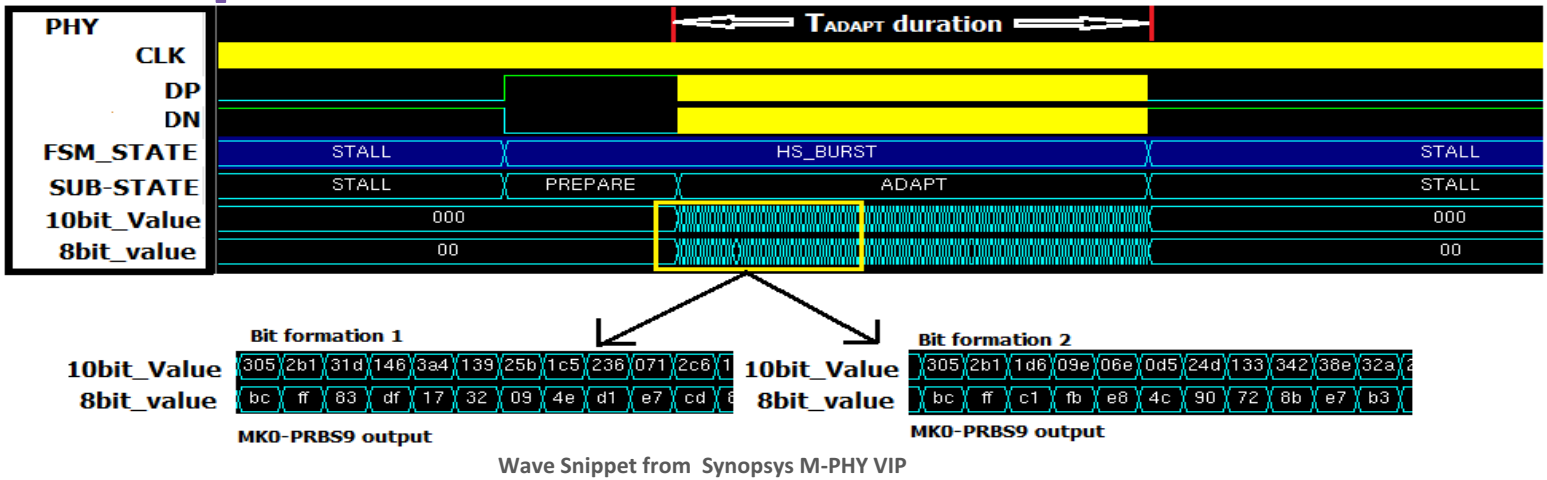


- If PRBS9 is initialized with all 1's (1_1111_1111) on every 8 clock, output will be:
- Bit formation1 : 11111111_10000011_11011111_00110010_00001001 i.e., 'hFF → 'h83 → 'hDF → 'h17 → 'h32 → 'h09
- Bit formation2 : 11111111_11000001_11111011_01001100_10010000 i.e., 'hFF → 'hC1 → 'hFB → 'h4C → 'h90

Synopsys

© 2017 MIPI Alliance, Inc.

ADAPT Operation



Wave Snippet from Synopsys M-PHY VIP

- ADAPT operation requirements:
  - Link should be in HS_G4 gear
  - M-RX attribute RX_ADAPT_Control should be programmed to 2'bX0
    - Where X can be 0 = Initial ADAPT length or 1 = Refresh ADAPT length
- On receiving ADAPTSTART.req from protocol, PHY moves to ADAPT sub-state after PREPARE and starts driving ADAPT sequence for T_{ADAPT} length; ADAPT sub-state is ended by BURSTEND SAP

Synopsys

© 2017 MIPI Alliance, Inc.

ADAPT Operation

- Duration for which complete ADAPT sequence takes is known as T_{ADAPT}
- M-RX has two 8-bit capability attribute for ADAPT length
  - RX_HS_ADAPT_REFRESH_Capability
  - RX_HS_ADAPT_INITIAL_Capability
- M-TX TX_HS_ADAPT_Length should be greater or equal to above capability values
- As per MIPI M-PHY specification T_{ADAPT} length is calculated as the following:

Attribute or Parameter	Value	Units
T_{ADAPT}	IF (ADAPT_type = FINE) T_{ADAPT} = 650 * (ADAPT_length + 1) ELSE (IF ADAPT_type = COARSE) T_{ADAPT} = 650 * 2^{ADAPT_length}, where ADAPT_length < 18 END	bit

Source: MIPI Alliance M-PHY v4.1

© 2017 MIPI Alliance, Inc.

Future Scope

- Improve performance by skipping SAVE state between ADAPT and DATA burst
  - i.e., PREAPRE → ADAPT → SYNC → DATA

© 2017 MIPI Alliance, Inc.

Alternate Sequence in MIPI D-PHY

- What is alternate sequence?
- Why and when alternate sequence is required?
- PRBS9 handling
- Alternate operation
- State flow
- Future scope

© 2017 MIPI Alliance, Inc.

What is Alternate Sequence?

- Training sequence for optimization of receiver equalizer settings in MIPI D-PHY using PRBS pattern is known as alternate sequence
- Alternate sequence was introduced in MIPI D-PHY v2.1
- PRBS with equation $X^9 + X^5 + 1$ (PRBS9) is used
- Used in combination with initial skew calibration

© 2017 MIPI Alliance, Inc.

Why and When Alternate Sequence is Required?

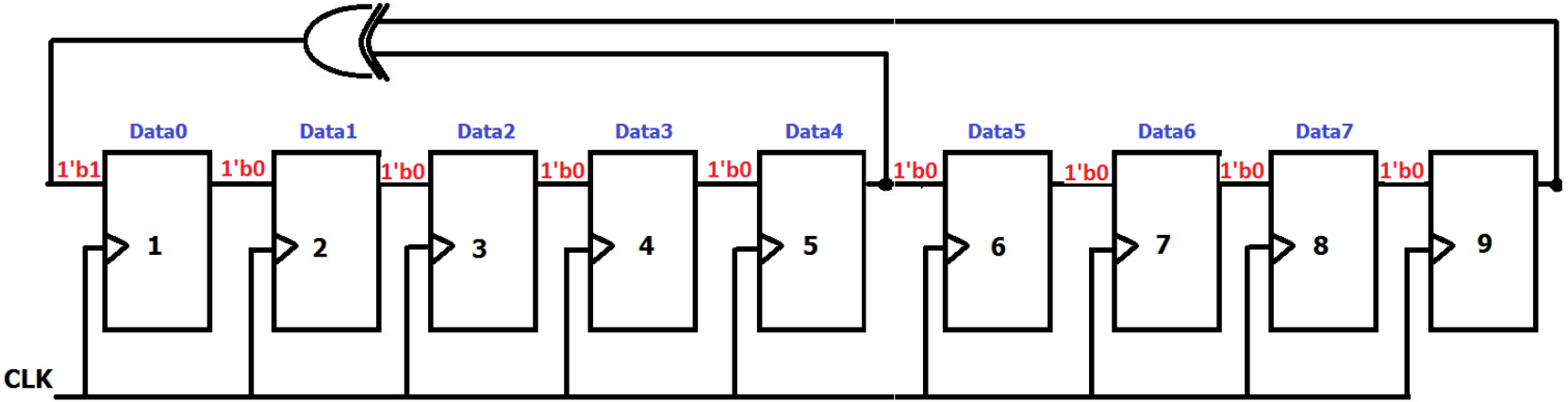
- Similar to MIPI M-PHY, MIPI D-PHY started supporting higher DATA rate 2.5Gbps or above; This leads to same inter symbol inference and noise
- In order to re-tune equalizer setting, alternate sequence is introduced as training sequence
- Its mandatory for DATA transmission above 2.5 Gbps; designer may choose to support alternate sequence at or below 2.5 Gbps
- Used at link power up and/or on link re-initialization
- Alternate sequence divided into three part HS0 → Calibration Sync → Calibration Pattern (PRBS9)

Synopsys

© 2017 MIPI Alliance, Inc.

PRBS9 Handling

- MIPI D-PHY was second PHY to introduce PRBS9 for calibration
- PRBS9 is initialized with 9-bit value 000000001
- For 8-bit data width on every 8 shifts, output is formed; As shown in figure Data[7:0] is the output from shift register 8 to 1

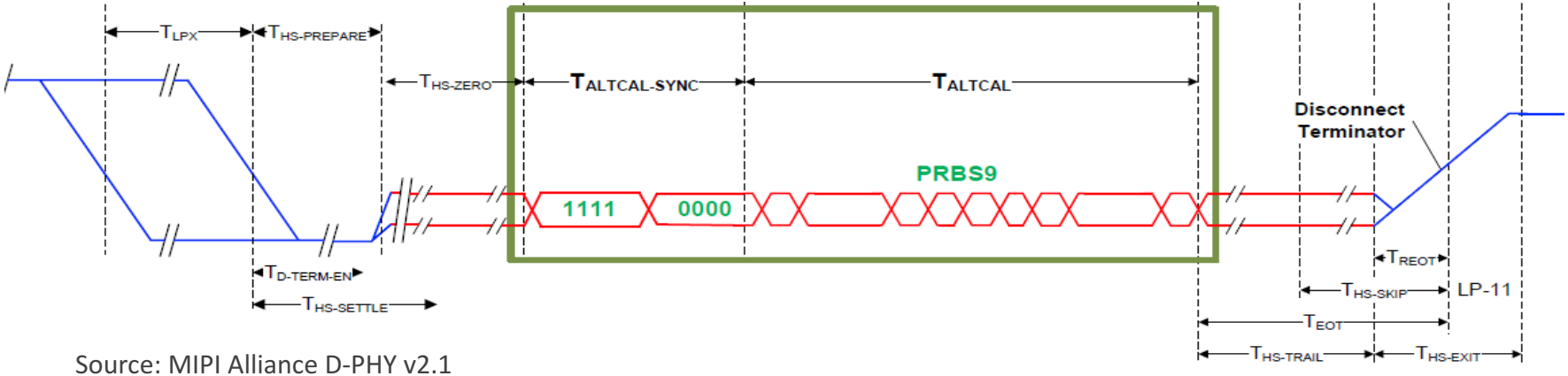


- For 16/32bit data width, same PRBS9 will be shifted 16/32 times to form Data[15:0]/Data[31:0] output respectively

© 2017 MIPI Alliance, Inc.

Alternate Operation

- Initial skew calibration is followed by alternate calibration when link is above 2.5Gbps speed
- After Initial skew links stop state LP11, followed by HS-request LP01 → LP-00 → HS0
- Alternate calibration consists of calibration sync and PRBS9 pattern

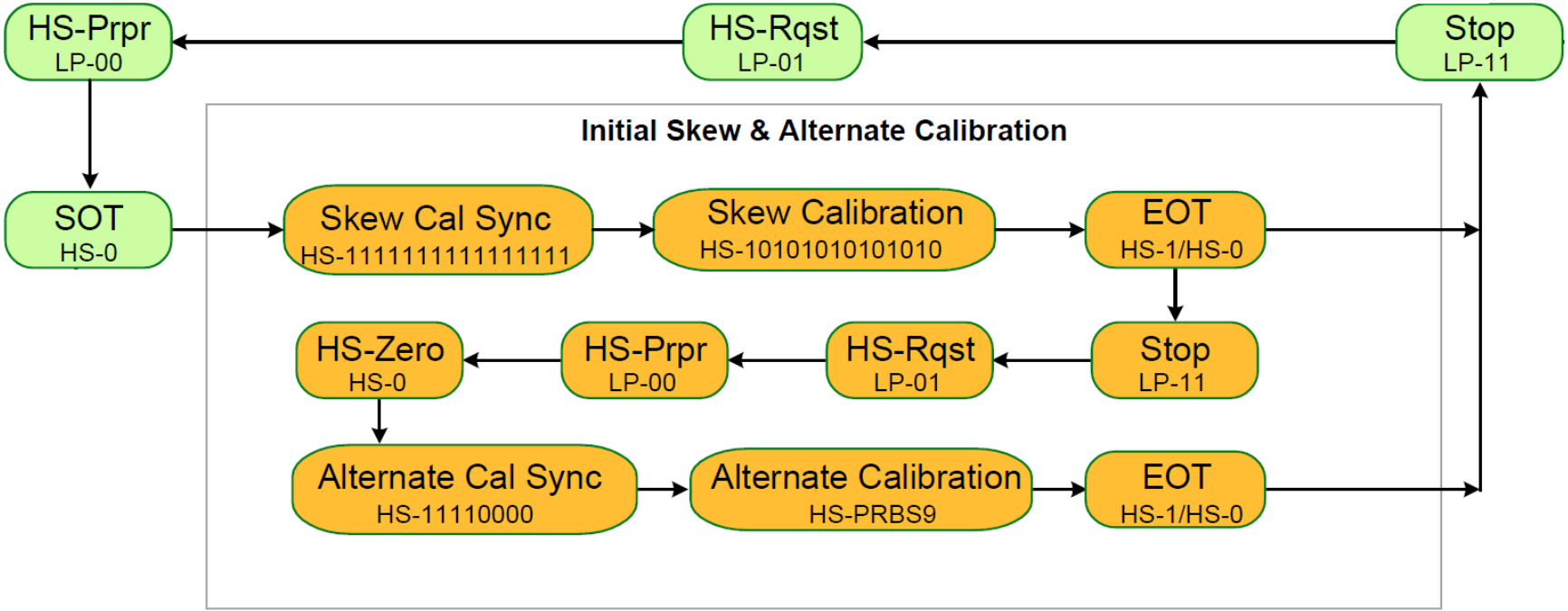


Source: MIPI Alliance D-PHY v2.1

Synopsys

© 2017 MIPI Alliance, Inc.

State Flow



Source: MIPI Alliance D-PHY v2.1

Synopsys

© 2017 MIPI Alliance, Inc.

Future Scope

- Alternate sequence can start as soon as initial skew pattern ends, instead of going to STOP state

Reference: MIPI Alliance DPHY Version 2.1

© 2017 MIPI Alliance, Inc.

Calibration Sequences in MIPI C-PHY

- What is calibration sequence?
- Types of calibration sequence
- Preamble handling and operation
- Alternate calibration handling and operation
- User-define calibration handling and operation
- State flow
- Future scope

Synopsys

© 2017 MIPI Alliance, Inc.

What is Calibration Sequence?

- Similar to MIPI M-PHY and MIPI D-PHY, in MIPI C-PHY, training sequence used to optimize receiver equalizer settings is known as Calibration sequence
- Calibration sequences were introduced in MIPI C-PHY v1.2
- There are three different calibration sequences that were introduced in MIPI C-PHY
- Calibration sequence will be transmitted along with DATA burst
- Calibration sequences are used when system operates above 3.0 Gsps
- System may support calibration below 3.0 Gsps DATA transmission

Synopsys

© 2017 MIPI Alliance, Inc.

Types of Calibration Sequence

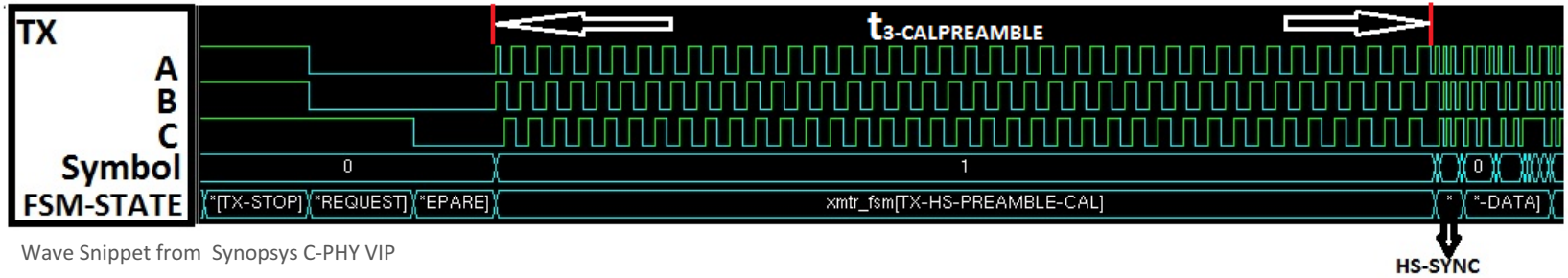
- MIPI C-PHY introduced three type of calibration sequence
  - Preamble calibration
  - Alternate calibration
  - User-define calibration
- Preamble calibration
  - series of 1's symbol
  - It's different from normal preamble pattern – all 3's – so receiver can easly identify difference between normal and calibration preamble
- Alternate Calibration
  - Starts right after preamble calibration
  - Begins with alternate ID followed by alternate pattern (PRBS9) similar as other two PHYs
- User-define Calibration
  - Starts right after preamble calibration
  - Begins with user-define ID followed by user-define pattern

Synopsys

© 2017 MIPI Alliance, Inc.

Preamble Sequence Handling & Operation

- This calibration type only consists of preamble sequence which is sequence of 1s symbol
- Sequence of 1 symbol transmission duration of $t_{3\text{-CALPREAMBLE}}$
- $t_{3\text{-CALPREAMBLE}}$ should be minimum of one group (7UI) to a maximum of 256 groups (7 X 256 = 1792 UI)
- Once preamble sequence completes, followed by SYNC and DATA



Wave Snippet from Synopsys C-PHY VIP

- $t_{3\text{-CALPREAMBLE}}$ For transmitter ranges from 7UI to 1792 UI
- Receiver should specify minimum value by which it can detect sequence of 1 symbol
- Max for receiver is 1792UI

Synopsys

© 2017 MIPI Alliance, Inc.

Alternate Sequence Handling

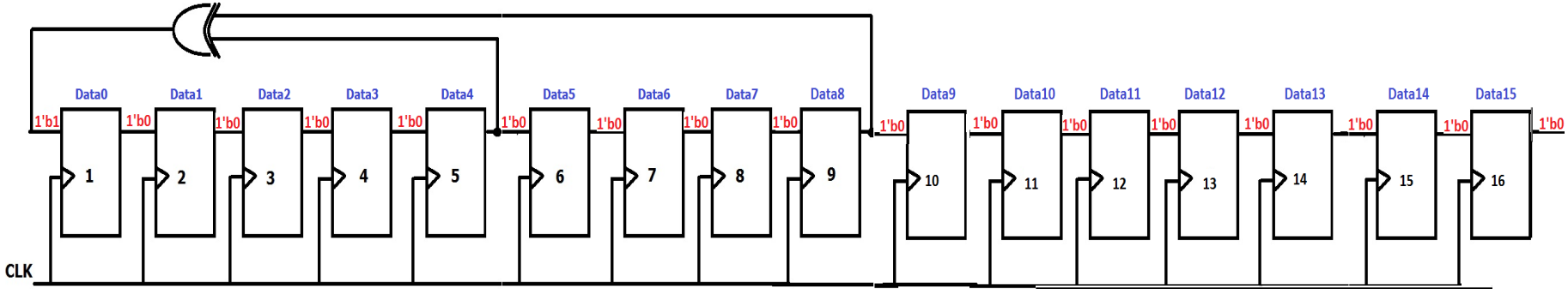
- Alternate calibration is divided into three parts
  - Preamble calibration
  - Alternate ID
  - Alternate sequence (PRBS9)
- Preamble Calibration
  - Alternate calibration first starts with preamble calibration
  - Preamble calibration (symbol of 1) is transmitted for $t_{3\text{-}CALPREAMBLE}$ duration
- Alternate ID
  - In-order to identify which type of calibration receiver is receiving, after preamble calibration seven “3” symbol is transmitted to indicate start of alternate calibration for $t_{3\text{-}ASID}$ duration which is fixed at 7UI
- Alternate sequence
  - Similar to MIPI M-PHY and MIPI D-PHY, MIPI C-PHY also uses PRBS9 sequence
  - Sequence is encoded in symbols using the MIPI C-PHY 3-phase mapper and encoder
  - Transmitted for $t_{3\text{-}CALALTSEQ}$ duration with minimum of one group (7UI) to 2048 groups (7 X 2048 = 14336 UI)

Synopsys

© 2017 MIPI Alliance, Inc.

Alternate Sequence Handling (Continued…)

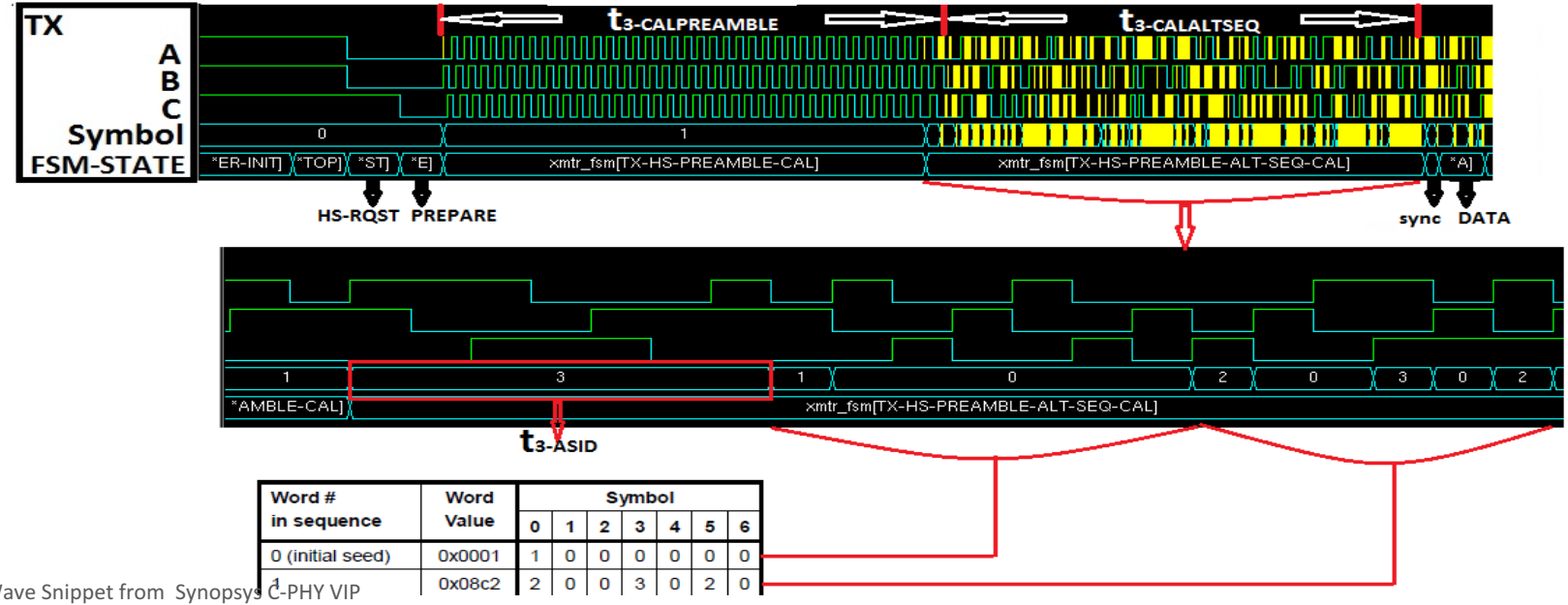
- Data0 to Data15 represent 16-bit output data Data[15:0]; data output is collected on every 16 clock shift
- There is no fix value for initial seed; below graphic shows when PRBS9 is initialized with seed value 0x0001



Word # in sequence	Word Value	Symbol						
		0	1	2	3	4	5	6
0 (initial seed)	0x0001	1	0	0	0	0	0	0
1	0x08c2	2	0	0	3	0	2	0

Synopsys

Alternate Calibration Operation



Word # in sequence	Word Value	Symbol						
		0	1	2	3	4	5	6
0 (initial seed)	0x0001	1	0	0	0	0	0	0
1	0x08c2	2	0	0	3	0	2	0

Wave Snippet from Synopsys C-PHY VIP

$t_{3\text{-CALALTSEQ}}$ For transmitter it ranges from 7UI to 14336 UI; receiver should specify required minimum; max for receiver is 14336UI

Synopsys

© 2017 MIPI Alliance, Inc.

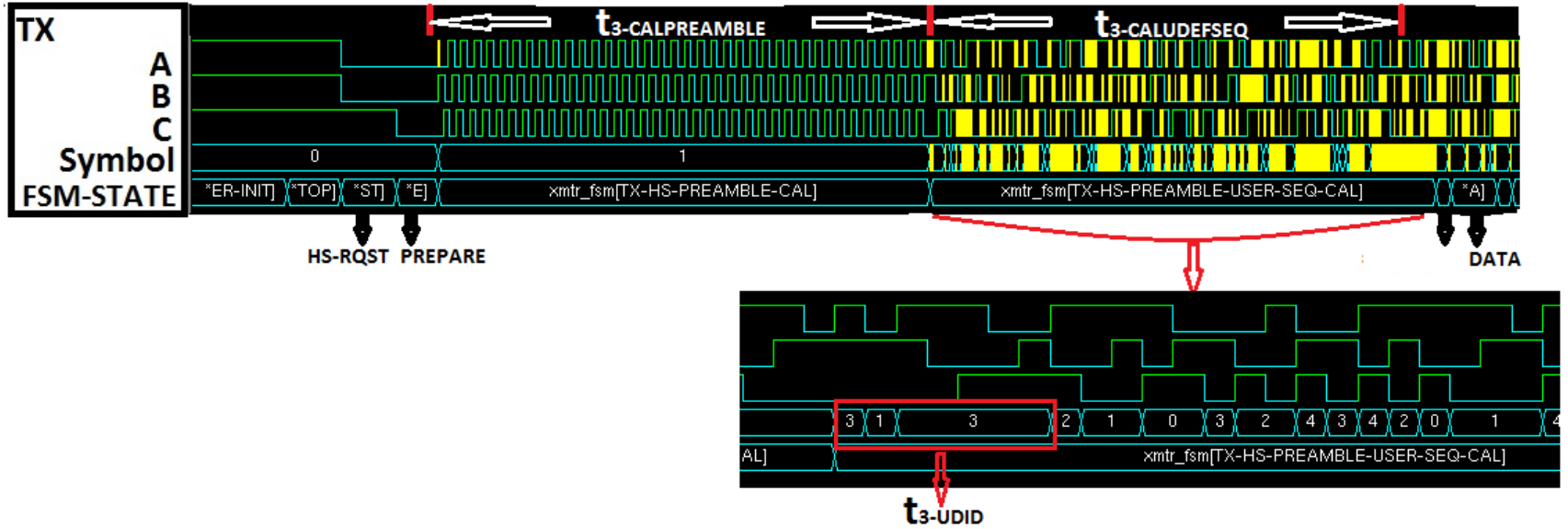
User-Defined Sequence Handling

- User-defined calibration is divided into three parts:
  - Preamble calibration
  - User-defined ID
  - User-defined sequence
- Preamble calibration
  - User-defined calibration also first starts with preamble calibration as discussed last
  - Preamble calibration (symbol of 1) is transmitted for $t_{3\text{-CALPREAMBLE}}$ duration
- User-defined ID
  - In-order to identify which type of calibration receiver receiving, after preamble calibration “3333313” symbol is transmitted to indicate start of user-defined calibration for $t_{3\text{-UDID}}$ duration which is fixed at 7UI
- User-defined sequence
  - As name suggests it consists of user define sequence
  - Driven for $t_{3\text{-CALUDEFSEQ}}$ duration which can be minimum of one group (7UI) to 2048 group (14336 UI)

Synopsys

© 2017 MIPI Alliance, Inc.

User-Defined Calibration Operation



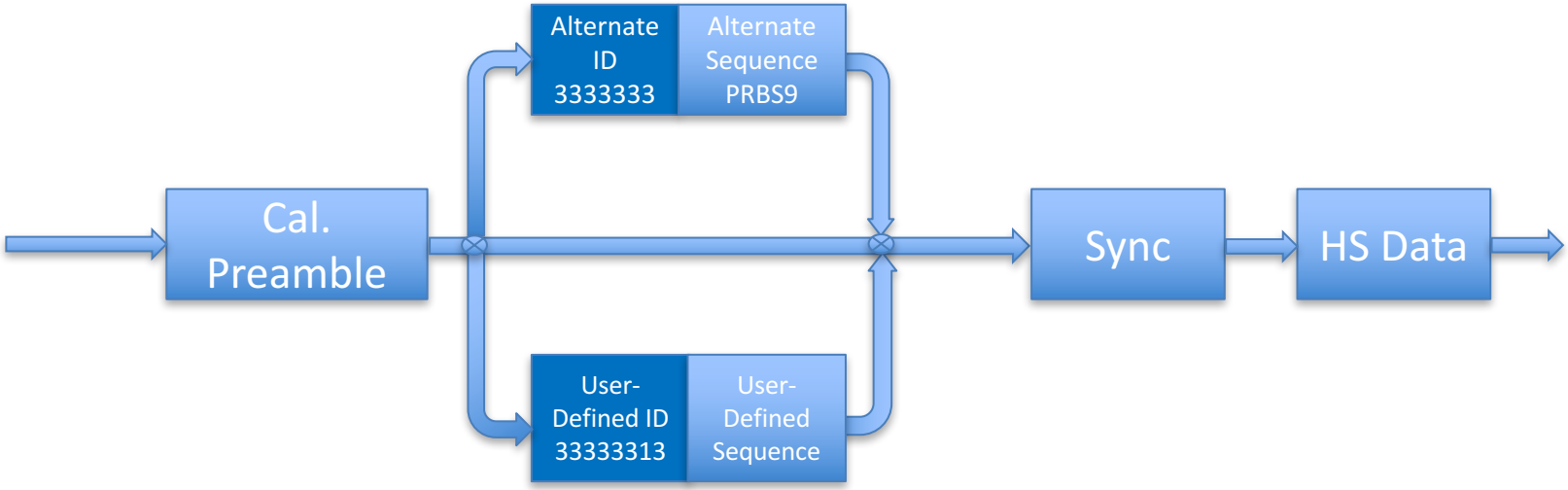
Wave Snippet from Synopsys C-PHY VIP

$t_{3\text{-CALUDEFSEQ}}$ For transmitter ranges from 7UI to 14336 UI; receiver should specify required minimum; max for receiver is 14336UI

Synopsys

© 2017 MIPI Alliance, Inc.

State Flow



Synopsys

© 2017 MIPI Alliance, Inc.

Future Scope

- To reduce clock cycle instead of sending calibration in each burst, we can send calibration in first burst (when driving above 3Gsps)

Reference: MIPI Alliance DPHY Version 2.1

Synopsys

© 2017 MIPI Alliance, Inc.

Summary

- As discussed, training sequences are introduced to re-tune the receiver equalizer circuits for proper data reception at high speed
- These patterns are used before actual data transmission so that receiver can self tune itself

	MIPI M-PHY	MIPI D-PHY	MIPI C-PHY
Used at SPEED	HS-G4 gear	Speed above 2.5 Gbps	Speed above 3.0 Gsps
Lower speed support	NO	May support	May support
Patterns used	PRBS9	PRBS9	Preamble, PRBS9 and User-define
Calibration flow	Before actual Data	Along with Initial skew before actual Data	Along with Data burst itself

Synopsys

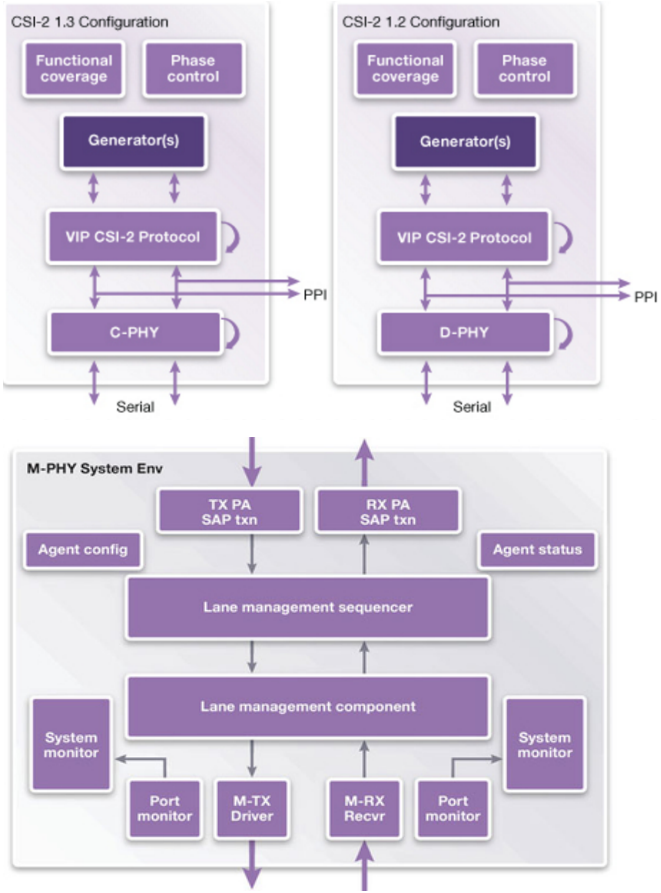
© 2017 MIPI Alliance, Inc.

VC VIP & TestSuite for MIPI

- MIPI CSI-2
  - CSI-2 1.1, 1.2
  - D-PHY 1.1, 1.2
  - CSI-2 1.3 compliant with C-PHY 1.0 and DPHY 1.2
  - CSI-2 2.0 compliant with C-PHY 1.1 and DPHY 2.0
- MIPI CSI-3 v1.0
- MIPI DSI
  - DSI 1.1, 1.2 compliant with D-PHY 1.1
  - DSI 1.3 compliant with D-PHY 1.2
  - DSI 2.0 compliant with C-PHY 1.0 and DPHY 2.0
- MIPI DigRF v4 1.10, 1.00 and 0.64

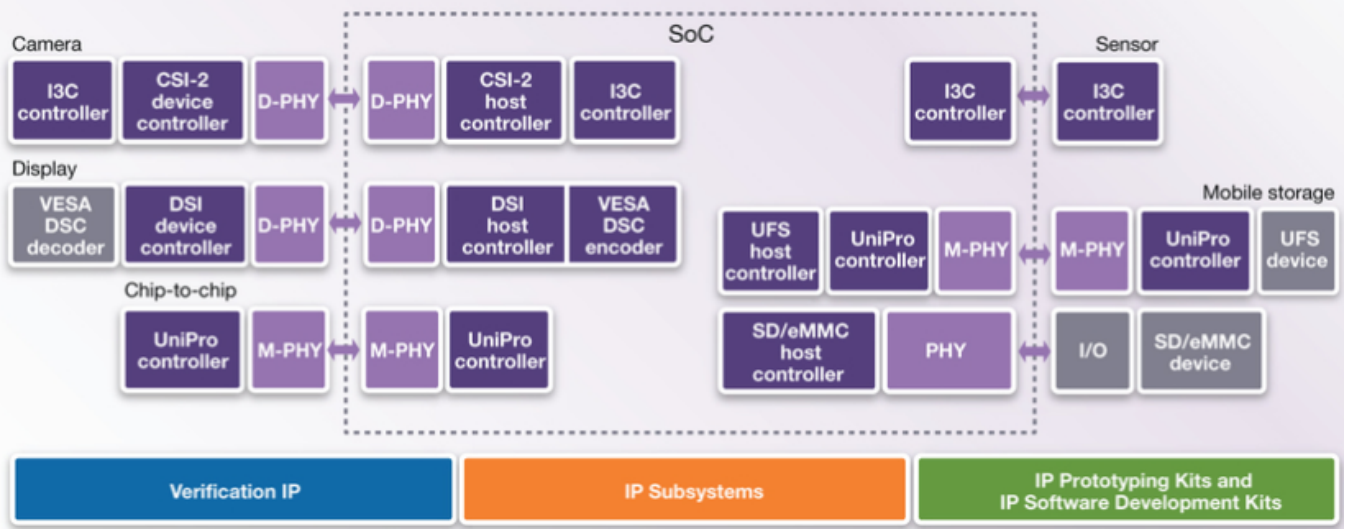
- MIPI HSI
- MIPI DBI 2.0 Specification
- MIPI DPI 2.0 Specification
- MIPI UniPro
- MIPI I3C
- MIPI RFFE
- MIPI SoundWire
- MIPI SPMI 2.0
- MIPI M-PHY v4.1

Synopsys



© 2017 MIPI Alliance, Inc.

Synopsys® DesignWare® MIPI IP Portfolio



Synopsys

© 2017 MIPI Alliance, Inc.

mipi®
DEVCON

THANK YOU

2017
MIPI ALLIANCE
DEVELOPERS
CONFERENCE

BANGALORE, INDIA

MIPI.ORG/DEVCON