AND9287/D

High-Speed Serial Pixel (HiSPi) Protocol Specification

ON Semiconductor®

www.onsemi.com

APPLICATION NOTE

INTRODUCTION

The HiSPi Protocol is primarily designed to support the high-speed transmission of image sensor data to an image processor over multiple, unidirectional, serial data Lanes. HiSPi Protocol v1.60.00 has four distinct operating Modes which this document will refer to as Packetized-SP Mode, Streaming-SP Mode, Streaming-S Mode, and ActiveStart-SP8 Mode. HiSPi Protocol Receivers may support any or all of these Modes, but all Receivers are strongly encouraged to support the Packetized-SP and/or Streaming-SP Modes. Users should check with ON Semiconductor regarding the Modes supported by any particular image sensor.

The Streaming-SP and Streaming-S Modes are very similar in that each provides explicit signaling for multiple blanking lines comprising the so-called vertical blanking interval of an image frame. Aptina's image sensors supporting the HiSPi Protocol will ensure that the lengths of these blanking lines conform to the lengths of active image lines. Therefore, insofar as Aptina's image sensors are concerned, image data sequences complying with the Streaming-SP and Streaming-S Modes are guaranteed to be largely synchronous with respect to actual image frame timing.

The Packetized-SP Mode and "legacy" ActiveStart-SP8 Mode do not support explicit signaling of blanking lines; however, this specification describes a new "streaming" variant of the ActiveStart-SP8 Mode called ActiveStart-SP8+ which does support signaling of blanking lines. In particular, the Packetized-SP Mode essentially treats each active image line as a self-contained packet of information which may be transported in a manner asynchronous to actual image frame timing.

Changes Relative to the Previous Version

HiSPi Protocol Specification v1.60.00 is fully backwards-compatible with the previous specification v1.50.00 but also includes the following enhancements and clarifications:

- The definition of the ActiveStart-SP8 Protocol Mode has been expanded to include a new variant, called ActiveStart-SP8+, which supports the signaling of vertical blanking lines (see ActiveStart-SP8+ Mode Syntax).
- The definition of the Word syntax element has been expanded to include a clear statement that the first bit of a Word is always asserted on the rising edge of a HiSPi clock; this is consistent with the way all HiSPi Transmitters have been designed and can simplify the logic in HiSPi Protocol Receivers for detecting HiSPi Streaming-S Protocol Mode Sync Codes (see Protocol Words).
- The definition of the 16-bit Checksum (CRC) syntax element has been improved to include both a block diagram of the CRC polynomial shift register as well as a clear statement of the shift register's initial state, i.e. 0xFFFF (see Checksum (CRC)).
- The guidelines for detecting Streaming-S Sync Codes have been made more robust; the previous guidelines inadvertently allowed false Sync Codes to be detected under certain circumstances (see Streaming-S Mode).

Terminology

Required – Refers to a feature which must be implemented in any HiSPi Protocol Transmitter or Receiver in reference to the Mode or context in question.

Unsupported – Refers to a feature which must not be implemented in any HiSPi Protocol Transmitter in reference to the Mode or context in question.

Optional – Refers to a feature which may be implemented in a HiSPi Protocol Transmitter. The feature's presence and enablement method will generally be datasheet dependent.

Reserved (or Res) – Refers to a bit pattern whose contents may change in a future version of the HiSPi Protocol and must be ignored by a HiSPi Protocol Receiver.

HiSPi Physical Interface

The HiSPi physical interface consists of *N* PHYs, where each PHY has *K* active, serial data Lanes used for data transmission. Note that the number of active Lanes on a PHY may be less than the total number of Lanes actually available on a device. *N* is theoretically unlimited, while *K* may range from 1 to 4 for $N > 1$, and from 1 to 8 for $N = 1$. Both *N* and *K* will vary depending on the selected image sensor and HiSPi operating Mode. For the Streaming-S Mode, $K = 3, 5, 6,$ or 7 is not permitted.

HiSPi PHY electrical characteristics are described in the HiSPi Physical Layer Specification. Each PHY has exactly one Double Data Rate (DDR) clock Lane which is used to synchronize all data Lane transmissions on that PHY. DDR means that data bits are asserted on both rising and falling clock edges.

Basic Protocol Syntax

HiSPi transmits image sensor data as a sequence of one or more image frames. Each transmitted frame is composed of the basic protocol syntax elements listed in Table 1. Each Mode uses a subset of these elements, and the format of each element may also vary by Mode. Each syntax element consists of a collection of "Words" (see Protocol Words) beginning on Lane 1 of PHY 1 and ending on Lane *K* of PHY *N*.

Table 1. BASIC SYNTAX ELEMENTS OF HiSPi PROTOCOL

Syntax Element	Packetized-SP	Streaming-SP	Streaming-S	ActiveStart-SP8
SOF Sync Code: Start of Active Frame	Required	Required	Unsupported	Required
SOL Sync Code: Start of Active Line	Required	Required	Required	Required
EOF Sync Code: End of Active Frame	Required	Unsupported	Unsupported	Unsupported
EOL Sync Code: End of Active Line	Required	Unsupported	Unsupported	Unsupported
SOV Sync Code: Start of Vertical Blanking Line	Unsupported	Required	Required	Unsupported
FLR: Filler Code	Optional	Optional	Optional	Optional
AIL: Active Image Line	Required	Required	Required	Required
CRC: Checksum	Optional	Optional	Optional	Optional
IDL: Idle Period	Required	Required	Required	Required

Packetized-SP Mode Syntax

As shown in Figure 1, Packetized-SP Mode outputs an image frame as a series of *P* Active Image Lines (AILs), each of which is delimited by header and footer Sync Codes. Each AIL will normally have the same length, but this is not required by the protocol syntax.

If an AIL is the first image line of a frame, the header consists of an SOF Sync Code; otherwise, the header is an SOL Sync Code. The header Sync Code is then optionally followed by a Filler Code (FLR) which may be inserted to give the Receiver more time to process the header. If an AIL is the last image line, the footer following the AIL consists of an EOF Sync Code; otherwise, the footer is an EOL Sync Code. Each footer Sync Code is optionally followed by a Checksum (CRC).

An Idle Period (IDL) always follows the footer or CRC. The timing of the IDL may roughly coincide with an imager's horizontal or vertical blanking interval, but this is not required by the HiSPi protocol. The length of the IDL may vary from line-to-line or from frame-to-frame. As shown at the bottom of Figure 1, an IDL of undetermined length always precedes the initial frame of a frame sequence.

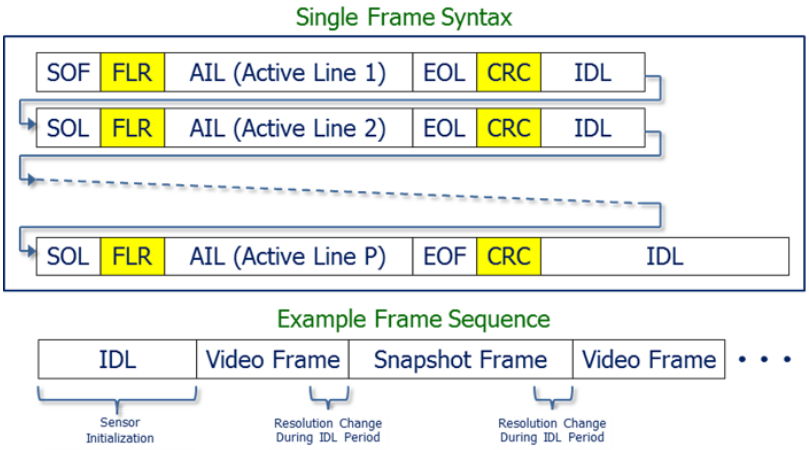


Figure 1. Packetized-SP Mode Image Frame Syntax (Yellow Indicates Optional Element)

Note that a HiSPi Protocol Receiver capable of synchronizing to every header and footer Sync Code does not need to know in advance the length of an AIL or IDL. As will be discussed later, synchronization is made possible by the special formats of the header and footer Sync Codes and restrictions placed on the values within each AIL.

Streaming Mode Syntax

The Streaming-SP and Streaming-S Modes have the same high-level frame syntax with the exception of the Sync Code used to begin the first active line of each frame (i.e. SOF vs. SOL, respectively). Each of these Modes has two sub-modes called Active-before-Blanking (ABB) and Blanking-before-Active (BBA) which differ in terms of how vertical blanking intervals are output in relation to active image lines.

Support of these sub-modes is image sensor dependent, but normally, any Aptina image sensor supporting the BBA sub-mode will also support the ABB sub-mode (however, the reverse is not true).

Active-before-Blanking Sub-Mode

As shown in Figure 2, the ABB sub-mode initially outputs an image frame as a series of *P* Active Image Lines (AILs), each of which is preceded by a Sync Code. Each AIL normally has the same length. The first AIL of an image frame is preceded by either an SOF or SOL Sync Code, respectively, depending on whether Streaming-SP or Streaming-S Mode has been selected in the image sensor. All other AILs are preceded by an SOL Sync Code.

The Sync Code is then optionally followed by a Filler Code (FLR) which may be inserted to give the Receiver more time to process the Sync Code. Unlike the Packetized-SP Mode, there is no footer Sync Code, so each AIL may be immediately followed by an optional Checksum (CRC).

An Idle Period (IDL) always follows each optional CRC. The timing of this Idle Period usually overlaps with an image sensor's horizontal blanking interval, and as such, the IDL will normally have the same length for all active image lines. However, this timing is not required by the protocol syntax. As shown in the example frame sequence at the bottom of Figure 2, an IDL of undetermined length also always precedes the initial frame of an ABB frame sequence.

As shown in Figure 2, the IDL of the last Active Image Line is followed by a series of *Q* vertical blanking lines, each of which is signaled by an SOV Sync Code. Each SOV is then followed by an IDL whose length is normally fixed by the image sensor throughout the vertical blanking interval and equal in length to the last active line; i.e. the total Word count in SOV + IDL equals the total Word count in SOL + (FLR) + AIL + (CRC) + IDL of active line *P*. The total number of vertical blanking lines is implementation-dependent and outside the scope of the HiSPi protocol.

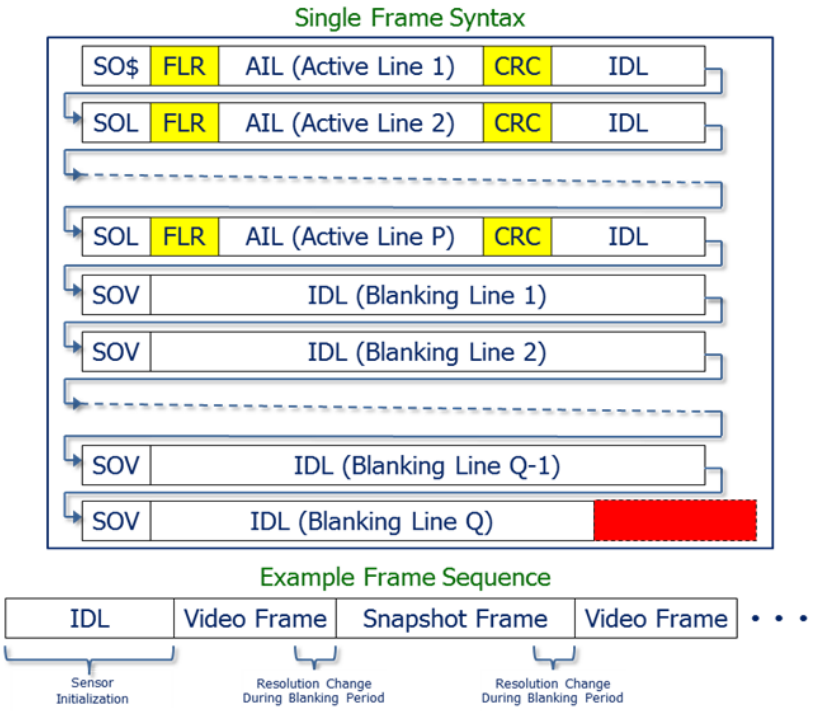


Figure 2. "Active-before-Blanking" (ABB) Frame Syntax
(SO$ is SOF for Streaming-SP and SOL for Streaming-S; Yellow Indicates Optional Element)

Note that blanking line *Q* in Figure 2, which is the last vertical blanking line before the first active line of the next image frame, may be shorter or longer than the other blanking lines if the image frame format is reset or changed during the vertical blanking interval. Actual behavior is image sensor dependent. This condition is conceptually illustrated by the **red** box in Figure 2.

Note that a HiSPi Protocol Receiver capable of synchronizing to every Sync Code does not need to know in advance the length of an AIL or IDL. However, if these lengths are known in advance, then the Receiver, in theory, does not have to resynchronize until the image frame format is reset or changed.

Blanking-before-Active Sub-Mode

The Blanking-before-Active (BBA) sub-mode differs from the ABB sub-mode in that the first active line of an image frame is always preceded by at least one complete vertical blanking line whose length matches the length of the first image line. This is shown in Figure 3, where the total Word count of SOV + IDL in blanking line *Q* equals the total Word count of SOL + (FLR) + AIL + (CRC) + IDL in the first active line of the current frame. Furthermore, if the image frame format is reset or changed during the vertical blanking interval, then blanking line *Q* may be preceded by one or more blanking lines whose lengths are shorter or longer than that of blanking line *Q*.

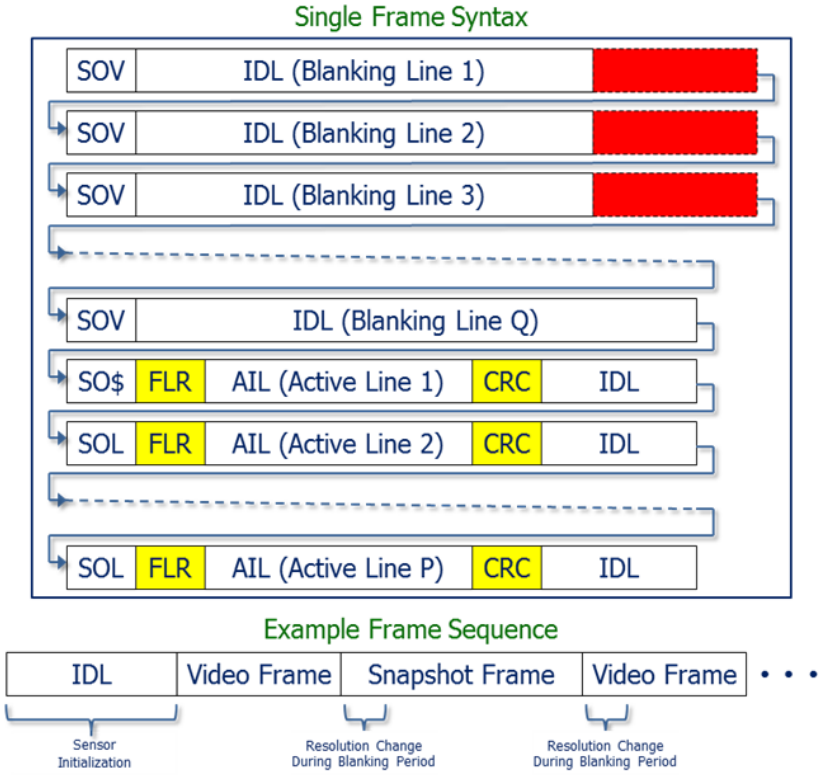


Figure 3. "Blanking-before-Active" (BBA) Frame Syntax
(SO$ is SOF for Streaming-SP and SOL for Streaming-S; Yellow Indicates Optional Element)

The latter condition is conceptually illustrated by the **red** boxes in Figure 3. During normal, uninterrupted video streaming, blanking lines 1 through *Q* will each have the same length as the first active line of the current frame. However, if the image frame format is reset or changed during the first part of the vertical blanking interval shown in Figure 3, then: (1) each blanking line preceding the reset or change will have the same length as the last active line of the previous frame; (2) the length of the blanking line during which the reset or change occurs may be shorter or longer than the preceding blanking lines; and (3) each blanking line following the reset or change will have the same length as the first active line of the current frame. See Figure 4 for further examples of conditions (1), (2), and (3). Actual behavior is image sensor dependent.

The active image portion of the Blanking-before-Active frame shown in Figure 3 has exactly the same syntax as that described in section Active-before-Blanking Sub-Mode for ABB frames.

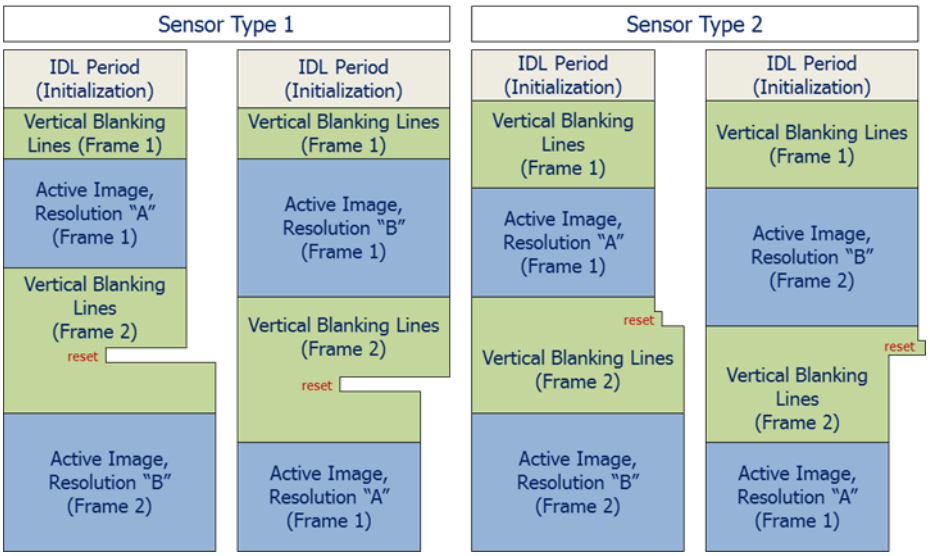


Figure 4. Examples of Blanking-before-Active Frame Resolution Changes for Two Hypothetical Image Sensor Types

ActiveStart-SP8 Mode Syntax

The ActiveStart-SP8 Protocol Mode supports both "legacy" and "streaming" frame syntaxes. The legacy syntax, known simply as "ActiveStart-SP8", is shown in Figure 5 and outputs an image frame as a series of *P* Active Image Lines (AILs), each of which is preceded by a Sync Code. Each AIL will normally have the same length, but this is not required by the protocol syntax. Note that every Sync Code is followed by an AIL plus optional FLR and CRC syntax elements.

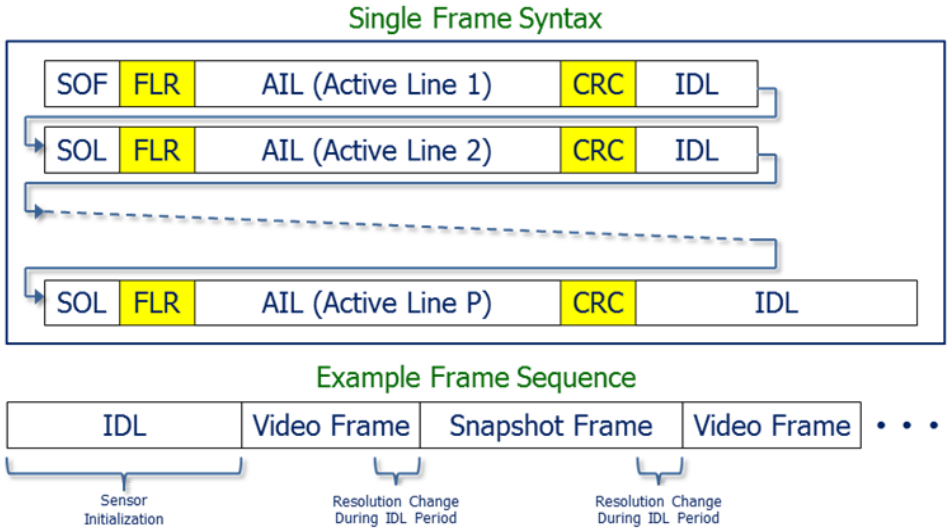


Figure 5. ActiveStart-SP8 Mode Frame Syntax (Yellow Indicates Optional Element)

An SOF Sync Code precedes the first AIL of an image frame; otherwise, an AIL is preceded by an SOL Sync Code. The Sync Code is then optionally followed by a Filler Code (FLR) which may be inserted to give the Receiver more time to process the Sync Code. Each AIL is optionally followed by a Checksum (CRC).

An Idle Period (IDL) always follows the AIL or CRC. The timing of the IDL may roughly correspond to an imager's horizontal or vertical blanking interval, but this is not required by the HiSPi protocol. As shown at the bottom of Figure 5, an IDL of undetermined length always precedes the initial frame of a frame sequence.

ActiveStart-SP8+ Mode Syntax

The ActiveStart-SP8 streaming frame syntax, pictured in Figure 6, is known as “ActiveStart-SP8+”. Unlike the legacy frame syntax, the streaming syntax permits Active Image Lines to be preceded and/or followed by a series of vertical blanking lines, each of which begins with an SOL Sync Code.

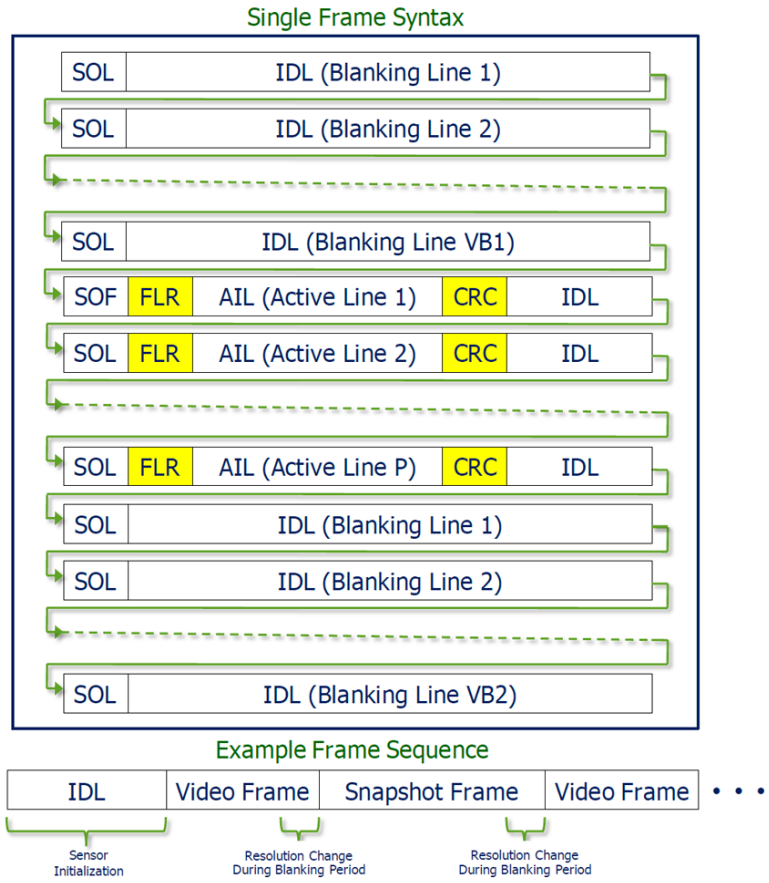


Figure 6. ActiveStart-SP8+ Mode Frame Syntax (Yellow Indicates Optional Element)

The values of VB1 and VB2 in Figure 6 are implementation-dependent. Note that if VB1 = VB2 = 0, the streaming frame syntax matches the legacy frame syntax. Also, since both Active Image Lines and vertical blanking lines use the same SOL Sync Code, it is critical that the Receiver know the value of P (i.e. the number of Active Image Lines) in advance.

Note that a HiSPi Protocol Receiver capable of synchronizing to every ActiveStart-SP8 Sync Code does not need to know in advance the length of an AIL or IDL Period if the IDL Word consists of either all 0- or 1-bits. In this case, the Receiver can unambiguously detect the beginning of an IDL Period following an AIL (or CRC, if present) since an AIL or CRC is not permitted to contain Words consisting of all 0- or 1-bits. However, if these lengths *are* known in advance, then the Receiver, in theory, does not have to resynchronize until the output image format is reset or changed.

SYNTAX ELEMENT DESCRIPTIONS

Protocol Words

Each of the syntax elements in Table 1 is transmitted as one or more L-bit Words, where L = 8, 10, 12, 14, or 16. The Word size must be understood by both the HiSPi Transmitter and Receiver and must remain fixed throughout the transmission of an image frame. Generally, L will match the active pixel precision, but this is not required by the HiSPi Protocol.

A Protocol Word is serially transmitted as a complete unit on a PHY data Lane; i.e. the bits making up a Word are never split between Lanes. Protocol Words are assigned to Lanes using two methods, Lane striping and Lane duplication, which may also be used in combination. Lane striping sequentially assigns each Word of a syntax element to a different data Lane, whereas Lane duplication copies a Word to multiple Lanes. The methods used vary by both syntax element and Mode.

A HiSPi PHY always synchronizes serial transmission of the first data bit of a Word with a rising clock edge of the associated clock Lane.

Sync Codes (SO*/EO*)

For the Packetized-SP, Streaming-SP, and Streaming-S Modes, each of the five Sync Codes listed in Table 1 consists of four L-bit Words. The first three Words of each Sync Code always consist of a single Word of all 1-bits followed by two Words of all 0-bits. The contents of the fourth Word depend on the selected HiSPi Mode and how the Sync Code is used.

However, for the ActiveStart-SP8 Mode, each of the SOF and SOL Sync Codes listed in Table 1 consists of eight L-bit Words.

SP Modes

For the Packetized- and Streaming-SP Modes, the fourth Sync Code Word is defined as shown in Table 2. Bit 0 is always transmitted first. Note that the “E” bit in the SOF and SOL Sync Codes signals whether the AIL following the Sync Code contains pixels or embedded data; this option is discussed later in section Active Image Line (AIL). All Packetized- and Streaming-SP Sync Codes are physically transmitted using Lane duplication; i.e. each four-Word Sync Code is duplicated and sequentially transmitted on each of the *N***K* Lanes. See Table 3 for examples.

Table 2. WORD 4 OF PACKETIZED- AND STREAMING-SP SYNC CODES (Note 1)

Sync Code Description	Which SP	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5: (L−1)
SOF: Start of Active Frame; E = 0 Implies Pixel Data, E = 1 Implies Embedded Data	Both	1	1	0	0	E	Reserved
SOL: Start of Active Line; E = 0 Implies Pixel Data, E = 1 Implies Embedded Data	Both	1	0	0	0	E	Reserved
EOF: End of Active Frame	Packetized	1	1	1	Res	Res	Reserved
EOL: End of Active Line	Packetized	1	0	1	Res	Res	Reserved
SOV: Start of Vertical Blanking Line	Streaming	1	0	0	1	Res	Reserved

1. Sync Code Word 1 always consists of all 1s, whereas each of Sync Code Words 2 and 3 always consists of all 0s.

Table 3. SYNC CODE TRANSMISSION FOR PACKETIZED- AND STREAMING-SP SYNC CODES (N ≥ 1, K ≤ 8)

PHY	Lane	Cycle 1	Cycle 2	Cycle 3	Cycle 4
1	1	111…1	000…0	000…0	Sync Code, Word 4
1	2	111…1	000…0	000…0	Sync Code, Word 4
↓	↓	↓	↓	↓	↓
1	K	111…1	000…0	000…0	Sync Code, Word 4
2	1	111…1	000…0	000…0	Sync Code, Word 4
2	2	111…1	000…0	000…0	Sync Code, Word 4
↓	↓	↓	↓	↓	↓
2	K	111…1	000…0	000…0	Sync Code, Word 4
…	…	…	…	…	…
N	1	111…1	000…0	000…0	Sync Code, Word 4
N	2	111…1	000…0	000…0	Sync Code, Word 4
↓	↓	↓	↓	↓	↓
N	K	111…1	000…0	000…0	Sync Code, Word 4

Streaming-S Mode

For the Streaming-S Mode, the fourth Sync Code Word is defined as shown in Table 4. Bit 0 is always transmitted first. All Streaming-S Sync Codes are physically transmitted using a combination of Lane duplication and Lane striping; in this case, a given four Word Sync Code is duplicated for each of the *N* PHYs, but the Words making up each Sync Code are striped across the *K* Lanes of each PHY. This means that the m-th Sync Code Word (for m = 1, 2, 3, or 4) is transmitted on Lane [(m−1) mod *K*] + 1 for each of PHYs 1 through *N*. Note that for the Streaming-S Mode, *K* must equal 1, 2, 4, or 8 (with *K* = 8 only being permitted for *N* = 1), so Word 4 will always be transmitted on Lane *K* of every PHY. See Table 5 for examples.

Table 4. WORD 4 OF STREAMING-S SYNC CODES (Note 2)

Sync Code Description	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8: (L−1)
SOL: Start of Active Line	1	0	0	0	0	0	0	0	000…0
SOV: Start of Vertical Blanking Line	1	0	1	0	1	0	1	1	000…0

2. Sync Code Word 1 always consists of all 1s, whereas each of Sync Code Words 2 and 3 always consists of all 0s.

Table 5. STREAMING-S SYNC CODE TRANSMISSION EXAMPLES

PHY	Lane	Cycle 1	Cycle 2	Cycle 3	Cycle 4
2 PHYS WITH 4 ACTIVE LANES PER PHY					
1	1	111…1	Pixels or Vertical Blanking Words (as Appropriate)		
1	2	000…0			
1	3	000…0			
1	4	Sync Code, Word 4			
2	1	111…1			
2	2	000…0			
2	3	000…0			
2	4	Sync Code, Word 4			
2 PHYS WITH 2 ACTIVE LANES PER PHY					
1	1	111…1	000…0	Pixels or Vertical Blanking Words (as Appropriate)	
1	2	000…0	Sync Code, Word 4		
2	1	111…1	000…0		
2	2	000…0	Sync Code, Word 4		
2 PHYS WITH 1 ACTIVE LANE PER PHY					
1	1	111…1	000…0	000…0	Sync Code, Word 4
2	1	111…1	000…0	000…0	Sync Code, Word 4
1 PHY WITH 8 ACTIVE LANES					
1	1	111…1	Pixels or Vertical Blanking Words (as Appropriate)		
1	2	000…0			
1	3	000…0			
1	4	Sync Code, Word 4			
1	5	111…1			
1	6	000…0			
1	7	000…0			
1	8	Sync Code, Word 4			
1 PHY WITH 4 ACTIVE LANES					
1	1	111…1	Pixels or Vertical Blanking Words (as Appropriate)		
1	2	000…0			
1	3	000…0			
1	4	Sync Code, Word 4			

Table 5. STREAMING-S SYNC CODE TRANSMISSION EXAMPLES (continued)

PHY	Lane	Cycle 1	Cycle 2	Cycle 3	Cycle 4
1 PHY WITH 2 ACTIVE LANES					
1	1	111…1	000…0	Pixels or Vertical Blanking Words (as Appropriate)	
1	2	000…0	Sync Code, Word 4		
1 PHY WITH 1 ACTIVE LANE					
1	1	111…1	000…0	000…0	Sync Code, Word 4

ActiveStart-SP8 Mode

Unlike the Packetized-SP, Streaming-SP, and Streaming-S Modes, the ActiveStart-SP8 Mode does not mandate any specific Sync Code values; rather, these values may be user-defined within all ON Semiconductor sensors supporting the ActiveStart-SP8 Mode. Such sensors may have default (i.e. reset) Sync Code values, but these defaults are not required to be supported by a HiSPi Protocol Receiver. The general guidelines for defining ActiveStart-SP8 Sync Codes are as follows:

1. An SOF or SOL Sync Code must consist of an ordered sequence of eight L-bit Words, each of which consists entirely of 1- or 0-bits.
2. SOF and SOL Sync Codes must not match.
3. SOF and SOL Sync Code patterns must not appear as a sequence of eight IDL Words.
4. SOF and SOL Sync Code patterns should be chosen such that they cannot naturally occur within sequences of pixels or embedded data whose values are constrained as described in section Active Image Line (AIL), implying that each 8*L-bit Sync Code should contain at least one run of consecutive 1- or 0-bits which is at least 2*L – 1 bits long. Sync Codes may violate this guideline, but their detection must then be enabled only during IDL Periods when no pixel or embedded data are expected.

Table 6 shows the default ActiveStart-SP8 Sync Codes for the 8 megapixel MT9E501 sensor. Note that the fourth guideline above is violated by the MT9E501's default SOL Sync Code; see ActiveStart-SP8 Mode for further discussion.

Table 6. DEFAULT ACTIVESTART-SP8 SYNC CODES FOR MT9E501 SENSOR

Sync Code Description	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
SOF: Start of Active Frame	111…1	000…0	111…1	000…0	111…1	111…1	111…1	111…1
SOL: Start of Active Line	111…1	000…0	111…1	000…0	111…1	000…0	111…1	000…0

Like the Packetized- and Streaming-SP Sync Codes, the ActiveStart-SP8 Sync Codes are physically transmitted using Lane duplication; as shown in Table 7, each eight-Word Sync Code is duplicated and sequentially transmitted on each of the *N***K* Lanes starting with Word 1.

Table 7. ACTIVESTART-SP8 SYNC CODE TRANSMISSION (N ≥ 1, K ≤ 8)

PHY	Lane	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5	Cycle 6	Cycle 7	Cycle 8
1	1	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
1	2	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
1	K	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
2	1	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
2	2	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
2	K	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
…	…	…	…	…	…	…	…	…	…

Table 7. ACTIVESTART-SP8 SYNC CODE TRANSMISSION (N ≥ 1, K ≤ 8) (continued)

PHY	Lane	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5	Cycle 6	Cycle 7	Cycle 8
N	1	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
N	2	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
N	K	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8

Filler Code (FLR)

The optional Filler Code (FLR) consists of a single Word formatted as shown in Table 8 and transmitted once (i.e. duplicated) on each of the *N***K* Lanes. Bit 0 is always transmitted first.

Table 8. FILTER CODE WORD

Bits 0: (L−2)	Bits (L−1)
000…0	1

Active Image Line (AIL)

Active image lines (AILs) hold the pixels or embedded data making up the active portion of each image frame. More than one complete pixel may be packed within each Word of an AIL; for example, two 7-bit pixels may be packed into a single 14-bit Word. A single pixel may also be split between two or more Words; for example, a 20-bit pixel may be transmitted using two 10-bit Words. The rules governing how multiple pixels may be packed into a single Word or how a single pixel may be split between multiple Words are product-specific.

Note that only the SOF and SOL Sync Codes as used by the Packetized- and Streaming-SP Modes support explicit signaling of embedded data on a line-by-line basis. Normally, an image sensor may optionally output one or more embedded data lines before and/or after all image pixel lines; however, the number, location, and contents of embedded data lines are beyond the scope of this specification.

A sequence of *M* pixel or embedded data Words making up an AIL is always physically transmitted using Lane-striping. This means that the m-th Word of an AIL (for m = 1, 2, …) is always transmitted on Lane ((m−1) *mod K*) + 1 of PHY *ceil* [(m−1)/*N*] (where *mod* is the modulo operator, and *ceil* is the "smallest integer greater than or equal to" ceiling operator). Note that *M* must be an integer multiple of *N***K*, thereby ensuring that the first Word in an AIL is always transmitted on Lane 1 of PHY 1, and the last Word of an AIL is always transmitted on Lane *K* of PHY *N*. Compliance with this rule may result in cases where the number of pixels in an active image line is less than the number of Words in an AIL. For such cases, a HiSPi Protocol Receiver must be aware of the number of trailing "padding" Words in an AIL so they may be discarded later. The contents of padding Words are not specified. See Table 9 for examples.

Unlike Sync Codes, the bit transmission order for AIL Words may be either LSB or MSB first. This order must be understood by both the HiSPi Protocol Transmitter and Receiver, since it is not signaled as part of the HiSPi protocol.

In order to avoid the creation of false Sync Codes, AILs created for the Streaming-S and ActiveStart-SP8 Modes may not contain any Words consisting entirely of 1- or 0-bits. A HiSPi Protocol Transmitter generally enforces this rule by clipping L-bit pixel values to the range $[1, 2^L - 2]$. However, for the Packetized-SP and Streaming-SP Modes, AILs are simply forbidden from containing Words consisting entirely of 0-bits. To enforce this rule, a HiSPi Protocol Transmitter generally clips L-bit pixel values to the range $[1, 2^L - 1]$.

Table 9. EXAMPLES OF AIL WORD STRIPING

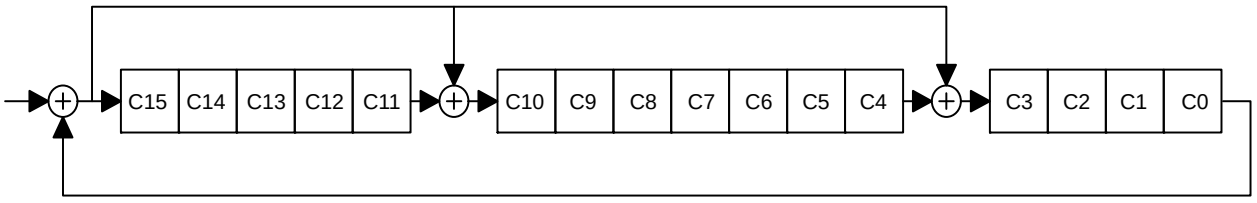
PHY	Lane	Cycle 1	Cycle 2	Cycle 3	Cycle 4
2 PHYS WITH 4 ACTIVE LANES PER PHY (AIL = 27 TOTAL PIXELS + 5 PADDING WORDS)					
1	1	Word 1	Word 9	Word 17	Word 25
1	2	Word 2	Word 10	Word 18	Word 26
1	3	Word 3	Word 11	Word 19	Word 27
1	4	Word 4	Word 12	Word 20	Padding Word
2	1	Word 5	Word 13	Word 21	Padding Word
2	2	Word 6	Word 14	Word 22	Padding Word
2	3	Word 7	Word 15	Word 23	Padding Word
2	4	Word 8	Word 16	Word 24	Padding Word

Table 9. EXAMPLES OF AIL WORD STRIPING (continued)

PHY	Lane	Cycle 1	Cycle 2	Cycle 3	Cycle 4
2 PHYS WITH 3 ACTIVE LANES PER PHY					
1	1	Word 1	Word 7	Word 13	Word 19
1	2	Word 2	Word 8	Word 14	Word 20
1	3	Word 3	Word 9	Word 15	Word 21
2	1	Word 4	Word 10	Word 16	Word 22
2	2	Word 5	Word 11	Word 17	Word 23
2	3	Word 6	Word 12	Word 18	Word 24
2 PHYS WITH 2 ACTIVE LANES PER PHY					
1	1	Word 1	Word 5	Word 9	Word 13
1	2	Word 2	Word 6	Word 10	Word 14
2	1	Word 3	Word 7	Word 11	Word 15
2	2	Word 4	Word 8	Word 12	Word 16
2 PHYS WITH 1 ACTIVE LANE PER PHY					
1	1	Word 1	Word 3	Word 5	Word 7
2	1	Word 2	Word 4	Word 6	Word 8
1 PHY WITH 1 ACTIVE LANE					
1	1	Word 1	Word 2	Word 3	Word 4

Checksum (CRC)

The optional Checksum is computed as a 16-bit CRC with generator polynomial $x^{16} + x^{12} + x^5 + 1$. A serial shift register representation of this generator polynomial is shown in Figure 7; the register is initialized to 0xFFFF at the start of each AIL. During the transmission of each AIL, a CRC is computed over each Lane's optional FLR and AIL Words plus, for the Packetized-SP Mode, the four footer Sync Code Words (i.e. EOL or EOF); bits are input to each Lane's CRC shift register in the same order they are transmitted. In essence, the Checksum is a vector of *N*K* CRCs whose values are striped across the *N*K* data Lanes. Each 16-bit CRC value, i.e. C[15:0], is output on its respective Lane as two successive Words formatted as shown in Table 10. The Checksum bits are serially transmitted on each Lane following the same bit order selected for pixels and embedded data. If "LSB first" is selected, then Bit 0 in Table 10 is transmitted first for each Word; otherwise, Bit (L−1) is transmitted first. See Table 11 for a simple Checksum transmission example.



Polynomial: $x^{16} + x^{12} + x^5 + x^0$

Note: C15 represents x^0, C0 represents x^{15}

Figure 7. CRC Generator Shift Register

Note that the HiSPi Protocol does not permit Checksum transmission for L = 8 (i.e. 8-bit Words), nor does it specify the action a HiSPi Protocol Receiver should take if it detects a Checksum error in a received image line.

Table 10. CHECKSUM FORMAT

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8	Bit 9:(L−1)
CRC Word 1	C[8]	C[9]	C[10]	C[11]	C[12]	C[13]	C[14]	C[15]	1	000…0
CRC Word 2	C[0]	C[1]	C[2]	C[3]	C[4]	C[5]	C[6]	C[7]	1	000…0

Table 11. CHECKSUM TRANSMISSION EXAMPLE: SINGLE 4-LANE PHY

PHY	Lane	Cycle 1	Cycle 2
1	1	Word 1 of Lane 1 CRC	Word 2 of Lane 1 CRC
1	2	Word 1 of Lane 2 CRC	Word 2 of Lane 2 CRC
1	3	Word 1 of Lane 3 CRC	Word 2 of Lane 3 CRC
1	4	Word 1 of Lane 4 CRC	Word 2 of Lane 4 CRC

Idle Period (IDL)

The Idle Period (IDL) syntax element is used by all Modes but is formatted differently for the Streaming-S Mode. The number of Words in an IDL sequence must be an integer multiple of $N*K$ to ensure that the next Sync Code starts on Lane 1 of PHY 1. If an IDL is being used to create a vertical blanking line for the Streaming-SP or Streaming-S Mode as shown in Figure 2 and Figure 3, its length (in the absence of sensor resets) is normally fixed throughout the vertical blanking interval and equal to the active line length (i.e. the total Word count in SOV + IDL equals the total Word count in SOF/SOL + (FLR) + AIL + (CRC) + IDL).

For the Packetized-SP, Streaming-SP, and ActiveStart-SP8 Modes, IDL consists of a sequence of identical Words. This Word may be user-defined, but the default value is 2^L-1 (i.e. all 1-bits) for each of the SP Modes and 0 (i.e. all 0-bits) for the ActiveStart-SP8 Mode.

The Streaming-S Mode defines the IDL sequence as two alternating, complementary cycles of $N*K$ identical Words, where each Word in each cycle is defined as shown in Table 12; Bit 0 is always transmitted first.

Table 12. STREAMING-S IDL EXAMPLES

	Bit 0	Bit 1: (L−1)
L-bit Word for Cycles 1, 3, 5, …	0	111…1
L-bit Word for Cycles 2, 4, 6, …	1	000…0

Note that while the total number of Words in a Streaming-S IDL must be an integer multiple of $N*K$, it does not have to be an even multiple; i.e. the IDL may end with either Word value. During transmission, the IDL is striped across the $N*K$ data Lanes, effectively resulting in the two complementary Word patterns of Table 12 alternating on each Lane. See Table 13 for examples.

Table 13. STREAMING-S IDL EXAMPLES

PHY	Lane	Cycle 1	Cycle 2	Cycle 3	Cycle 4
1 PHYS WITH 4 ACTIVE LANES; IDL = 4 WORDS/CYCLE, 12 TOTAL WORDS					
1	1	011…1	100…0	011…1	
1	2	011…1	100…0	011…1	
1	3	011…1	100…0	011…1	
1	4	011…1	100…0	011…1	
2 PHYS WITH 2 ACTIVE LANES PER PHY; IDL = 4 WORDS/CYCLE, 16 TOTAL WORDS					
1	1	011…1	100…0	011…1	100…0
1	2	011…1	100…0	011…1	100…0
2	1	011…1	100…0	011…1	100…0
2	2	011…1	100…0	011…1	100…0
2 PHYS WITH 1 ACTIVE LANE PER PHY; IDL = 2 WORDS/CYCLE, 8 TOTAL WORDS					
1	1	011…1	100…0	011…1	100…0
2	1	011…1	100…0	011…1	100…0

PROTOCOL EXAMPLES

The protocol examples in this section exhibit the Word-level structure of each HiSPi frame format for a four-Lane PHY. Each box in each of the following four figures below corresponds to one Word within a protocol syntax element, with the box label indicating the syntax element, and the subscript of each label indicating the Word number within the Word sequence comprising the element. For example, SOF4 is Word 4 of the four-Word SOF Sync Code, and FLR1 is the first (and only) Word comprising the Filler Code. Moreover, the superscript of each CRC Word indicates the Lane number the CRC is computed over.

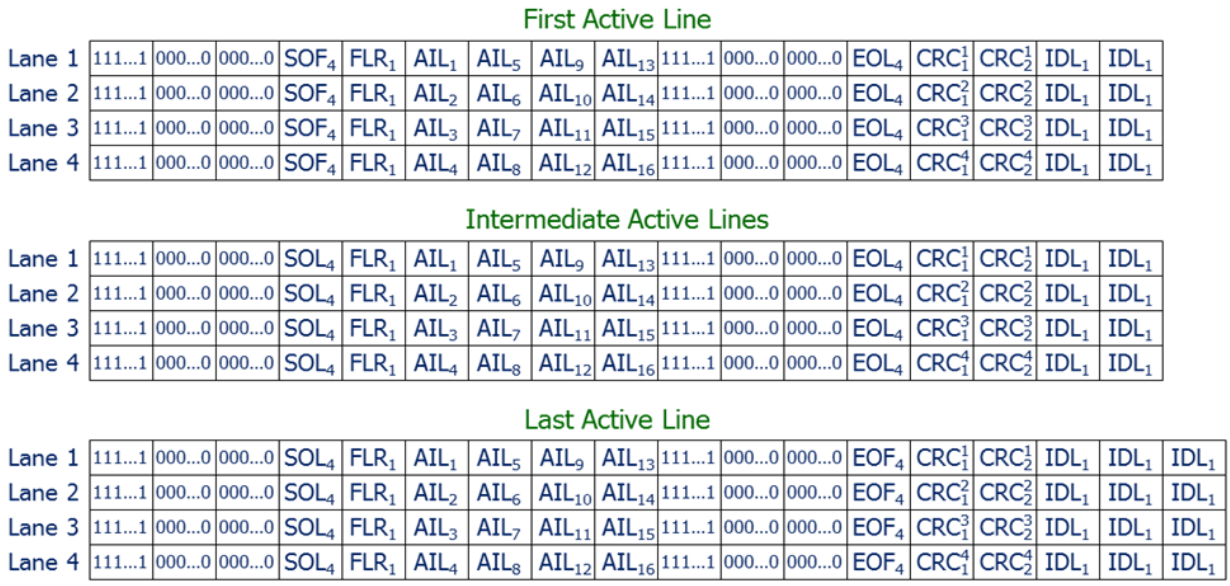


Figure 8. Four-Lane Example of Packetized-SP Mode Frame Format (16 Pixels/Line)

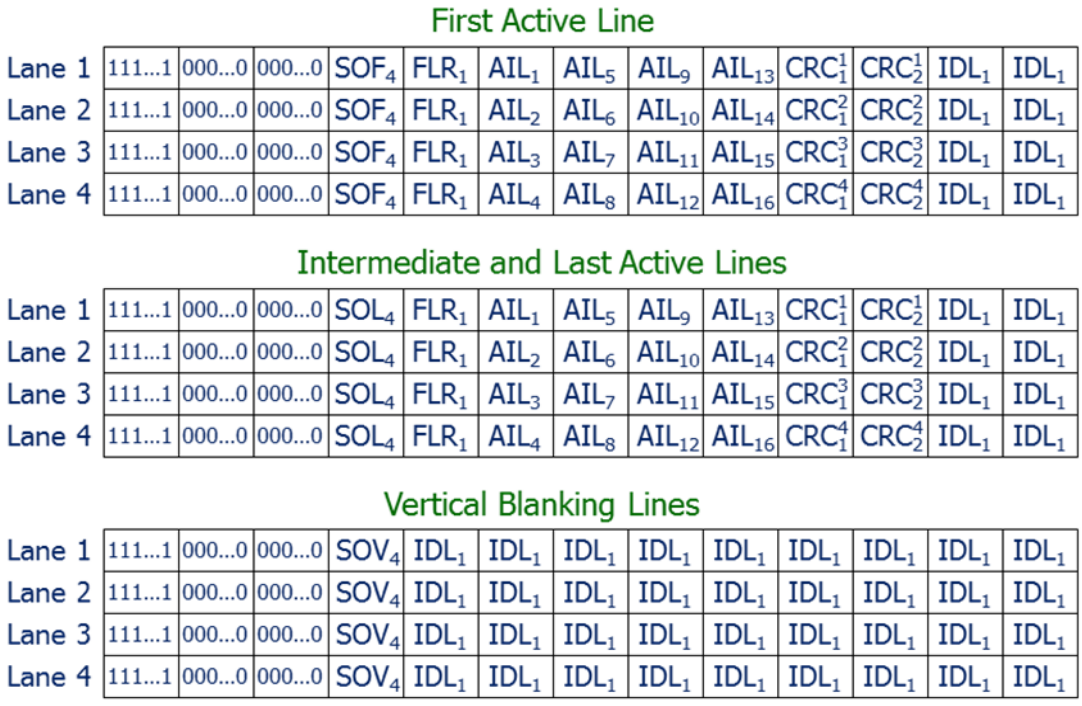


Figure 9. Four-Lane Example of Streaming-SP Mode Frame Format (16 Pixels/Line)

Active Lines

Lane 1	111...1	FLR_1	AIL_1	AIL_5	AIL_9	AIL_{13}	CRC_1^1	CRC_2^1	IDL_1	IDL_2
Lane 2	000...0	FLR_1	AIL_2	AIL_6	AIL_{10}	AIL_{14}	CRC_1^2	CRC_2^2	IDL_1	IDL_2
Lane 3	000...0	FLR_1	AIL_3	AIL_7	AIL_{11}	AIL_{15}	CRC_1^3	CRC_2^3	IDL_1	IDL_2
Lane 4	SOL_4	FLR_1	AIL_4	AIL_8	AIL_{12}	AIL_{16}	CRC_1^4	CRC_2^4	IDL_1	IDL_2

Vertical Blanking Lines

Lane 1	111...1	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1
Lane 2	000...0	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1
Lane 3	000...0	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1
Lane 4	SOV_4	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1	IDL_2	IDL_1

Figure 10. Four-Lane Example of Streaming-S Mode Frame Format (16 Pixels/Line)

First Active Line

Lane 1	SOF_1	SOF_2	SOF_3	SOF_4	SOF_5	SOF_6	SOF_7	SOF_8	FLR_1	AIL_1	AIL_5	AIL_9	AIL_{13}	CRC_1^1	CRC_2^1	IDL_1	IDL_1
Lane 2	SOF_1	SOF_2	SOF_3	SOF_4	SOF_5	SOF_6	SOF_7	SOF_8	FLR_1	AIL_2	AIL_6	AIL_{10}	AIL_{14}	CRC_1^2	CRC_2^2	IDL_1	IDL_1
Lane 3	SOF_1	SOF_2	SOF_3	SOF_4	SOF_5	SOF_6	SOF_7	SOF_8	FLR_1	AIL_3	AIL_7	AIL_{11}	AIL_{15}	CRC_1^3	CRC_2^3	IDL_1	IDL_1
Lane 4	SOF_1	SOF_2	SOF_3	SOF_4	SOF_5	SOF_6	SOF_7	SOF_8	FLR_1	AIL_4	AIL_8	AIL_{12}	AIL_{16}	CRC_1^4	CRC_2^4	IDL_1	IDL_1

Intermediate Active Lines

Lane 1	SOL_1	SOL_2	SOL_3	SOL_4	SOL_5	SOL_6	SOL_7	SOL_8	FLR_1	AIL_1	AIL_5	AIL_9	AIL_{13}	CRC_1^1	CRC_2^1	IDL_1	IDL_1
Lane 2	SOL_1	SOL_2	SOL_3	SOL_4	SOL_5	SOL_6	SOL_7	SOL_8	FLR_1	AIL_2	AIL_6	AIL_{10}	AIL_{14}	CRC_1^2	CRC_2^2	IDL_1	IDL_1
Lane 3	SOL_1	SOL_2	SOL_3	SOL_4	SOL_5	SOL_6	SOL_7	SOL_8	FLR_1	AIL_3	AIL_7	AIL_{11}	AIL_{15}	CRC_1^3	CRC_2^3	IDL_1	IDL_1
Lane 4	SOL_1	SOL_2	SOL_3	SOL_4	SOL_5	SOL_6	SOL_7	SOL_8	FLR_1	AIL_4	AIL_8	AIL_{12}	AIL_{16}	CRC_1^4	CRC_2^4	IDL_1	IDL_1

Last Active Line

Lane 1	SOL_1	SOL_2	SOL_3	SOL_4	SOL_5	SOL_6	SOL_7	SOL_8	FLR_1	AIL_1	AIL_5	AIL_9	AIL_{13}	CRC_1^1	CRC_2^1	IDL_1	IDL_1	IDL_1
Lane 2	SOL_1	SOL_2	SOL_3	SOL_4	SOL_5	SOL_6	SOL_7	SOL_8	FLR_1	AIL_2	AIL_6	AIL_{10}	AIL_{14}	CRC_1^2	CRC_2^2	IDL_1	IDL_1	IDL_1
Lane 3	SOL_1	SOL_2	SOL_3	SOL_4	SOL_5	SOL_6	SOL_7	SOL_8	FLR_1	AIL_3	AIL_7	AIL_{11}	AIL_{15}	CRC_1^3	CRC_2^3	IDL_1	IDL_1	IDL_1
Lane 4	SOL_1	SOL_2	SOL_3	SOL_4	SOL_5	SOL_6	SOL_7	SOL_8	FLR_1	AIL_4	AIL_8	AIL_{12}	AIL_{16}	CRC_1^4	CRC_2^4	IDL_1	IDL_1	IDL_1

Figure 11. Four-Lane Example of ActiveStart-SP8 Mode Frame Format (16 Pixels/Line)

RECEIVER SYNC CODE DETECTION

HiSPi Protocol Sync Codes serve two primary purposes: (1) they enable HiSPi Receivers to identify and synchronize to Word boundaries within serial bit streams, and (2) depending upon the selected Protocol Mode, they enable HiSPi Receivers to discriminate between active and blanking lines as well as determine the length and content of active lines.

Note that each HiSPi Receiver PHY/deserializer receives a separate clock signal from the Transmitter which may have a different phase relative to the clocks received by other PHYs. Therefore, the Words received by one PHY will eventually require timing resynchronization with the Words received by other PHYs. Furthermore, Sync Codes may conceivably be detected using either bit-serial or Word-parallel logic associated with each PHY.

Packetized- and Streaming-SP Modes

Sync Codes for the Packetized-SP and Streaming-SP Modes are detected using the same procedure. As described in section SP Modes, each Sync Code is four L-bit Words long and identically transmitted on each data Lane of every PHY. As such, a Sync Code may be detected using bits received from only a single data Lane of a given HiSPi Receiver PHY.

A HiSPi Packetized- or Streaming-SP Transmitter using the default IDL Word of all 1-bits will never output an L-bit Word of all 0-bits except during a Sync Code. However, as shown in Figure 12, runs of up to 2*L – 2 consecutive 0-bits may normally occur across Word boundaries in a variety of ways. Therefore, a Sync Code may be robustly detected on any given data Lane by observing a stream of 2*L – 1 or 2*L consecutive 0-bits (i.e. if 2*L – 1 consecutive 0-bits are observed, then the protocol guarantees that the next bit will also be 0). Furthermore, as shown in Figure 12, Word 4 of the Sync Code consists of the next L-bits to follow the 2*L consecutive 0-bits.

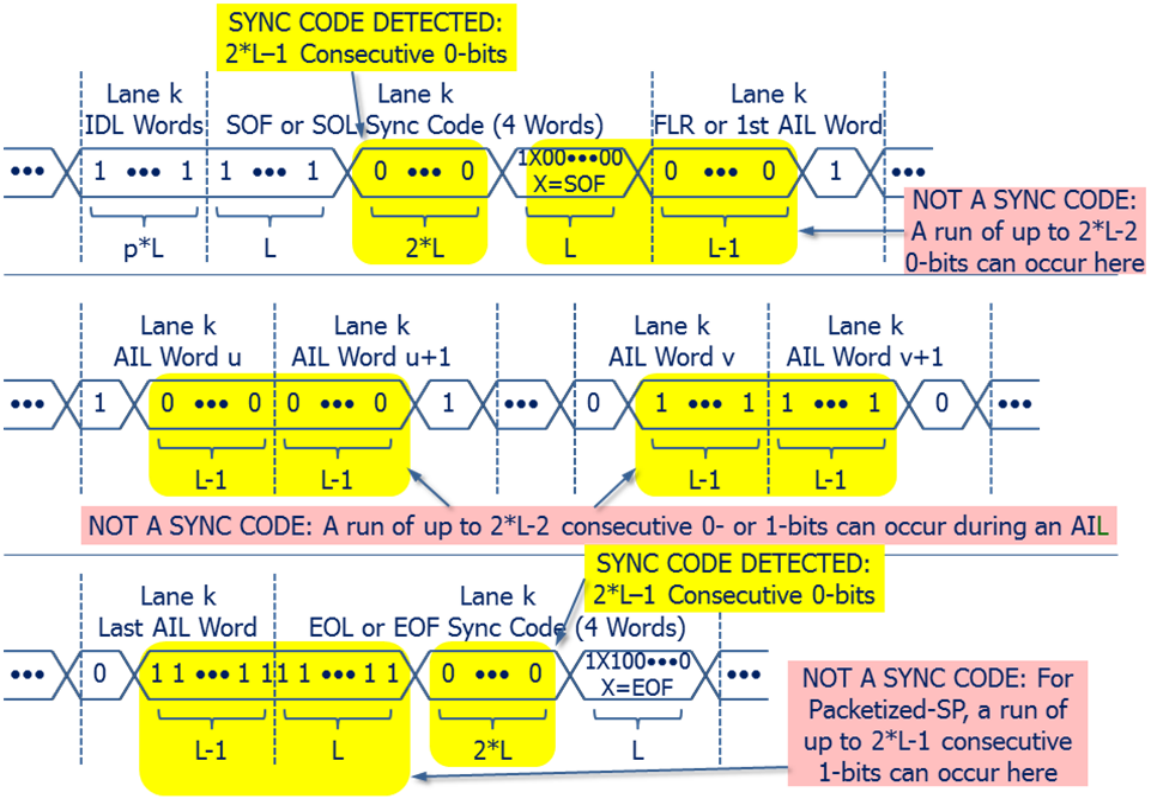


Figure 12. Sync Code Detection for Packetized- and Streaming-SP Modes

Streaming-S Mode

For applications with only a single active data Lane, the Streaming-S Sync Code detection procedure is the same as that discussed previously for the Packetized- and Streaming-SP Modes. However, the procedure is a bit more complicated for Streaming-S applications with two, four, or eight active data Lanes per PHY because the four-Word Sync Code is striped across the data Lanes within each PHY as described in section Streaming-S Mode.

As shown in Figure 2 and Figure 3, a Streaming-S Sync Code is always immediately preceded by one or more IDL Words. As such, the Sync Code detection procedure must accommodate the two different formats a Streaming-S IDL Word can take just prior to the occurrence of a Sync Code.

These formats are shown in Table 12 and alternate between odd- and even-numbered cycles during an IDL Word sequence. In one case, a Sync Code is immediately preceded by an odd-cycle IDL Word; in the other, a Sync Code is immediately preceded by an even-cycle IDL Word.

Four (or Eight) Active Data Lanes per PHY

Figure 13 depicts the case of a Streaming-S Sync Code which has been striped over four data Lanes and is immediately preceded by an odd-cycle IDL Word on each Lane. For this case, a Sync Code may robustly be detected within a PHY by observing a run of 2*L – 1 consecutive 1-bits on Lane 1 immediately following a 0-bit occurring at an even-numbered bit position which, for purposes of this discussion, means a bit asserted on a rising edge of the HiSPi clock. Figure 13 also illustrates how as many as 3*L – 2 consecutive 1-bits can occur on Lane 1 when the first AIL Word is 0xFFFE. However, by using the previous rule, none of these subsequent runs of 2*L – 1 consecutive 1-bits will signal a Sync Code because none of them is preceded by a 0-bit.

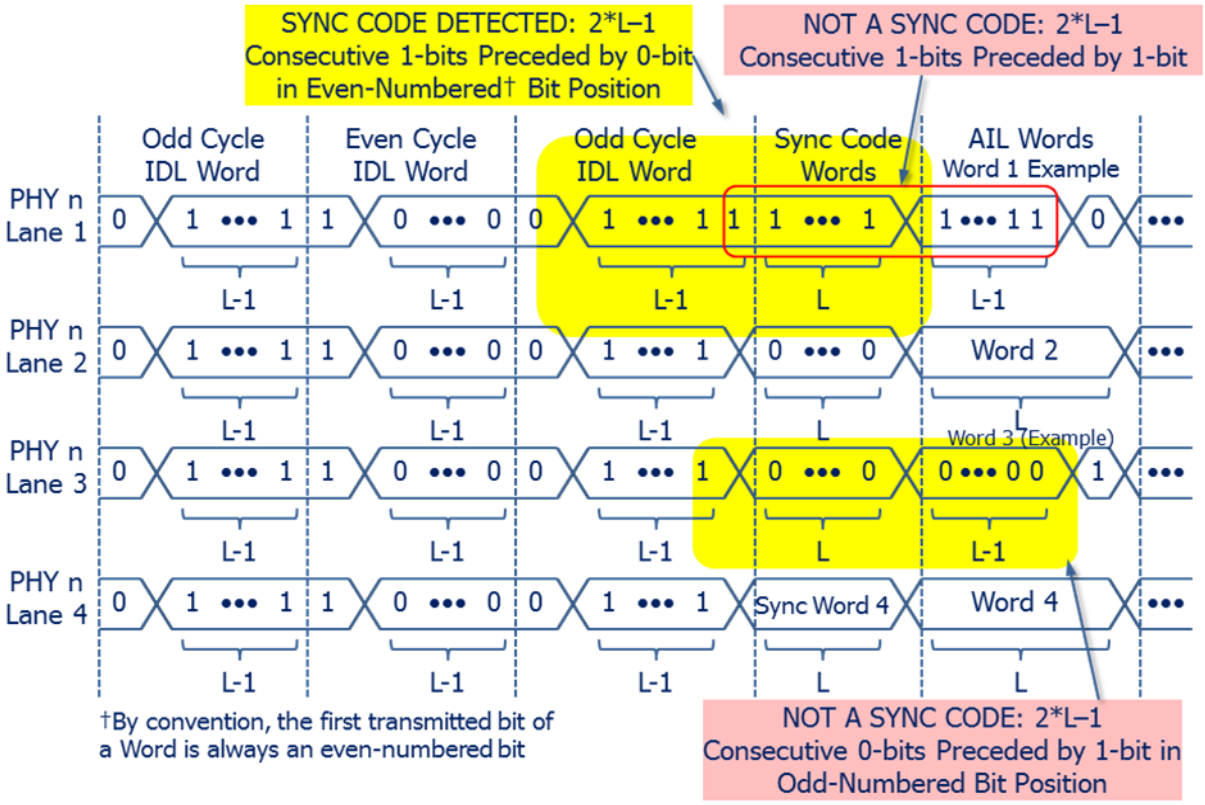


Figure 13. Four Lane Streaming-S Sync Code Immediately Preceded by an Odd-Cycle IDL Word

Figure 14 depicts the alternate case where the Streaming-S Sync Code is immediately preceded by an even-cycle IDL Word. For this case, a Sync Code may be robustly detected by observing 2*L – 1 consecutive 0-bits on Lanes 2 or 3 which immediately follow a 1-bit occurring in an even-numbered bit position. Note that a run of 2*L – 1 consecutive 0-bits following a 0-bit does not cause a Sync Code to be detected.

Figure 14 further illustrates how this case can result in a 0-bit being followed by 2*L – 1 consecutive 1-bits on Lane 1 when the first AIL Word has the value 0xFFFE. However, this will not cause a Sync Code to be detected on Lane 1 because the 0-bit preceding the 2*L – 1 consecutive 1-bits occurs at an odd-numbered bit position (corresponding to a falling clock edge) rather than at an even-numbered position.

Note that the previous Figure 13 also illustrates how, for the case when the Sync Code is immediately preceded by an odd-cycle IDL Word, a run of 2*L – 1 consecutive 0-bits can occur on Lanes 2 or 3 immediately following a 1-bit occurring in an odd-numbered bit position, again resulting in no Sync Code being detected.

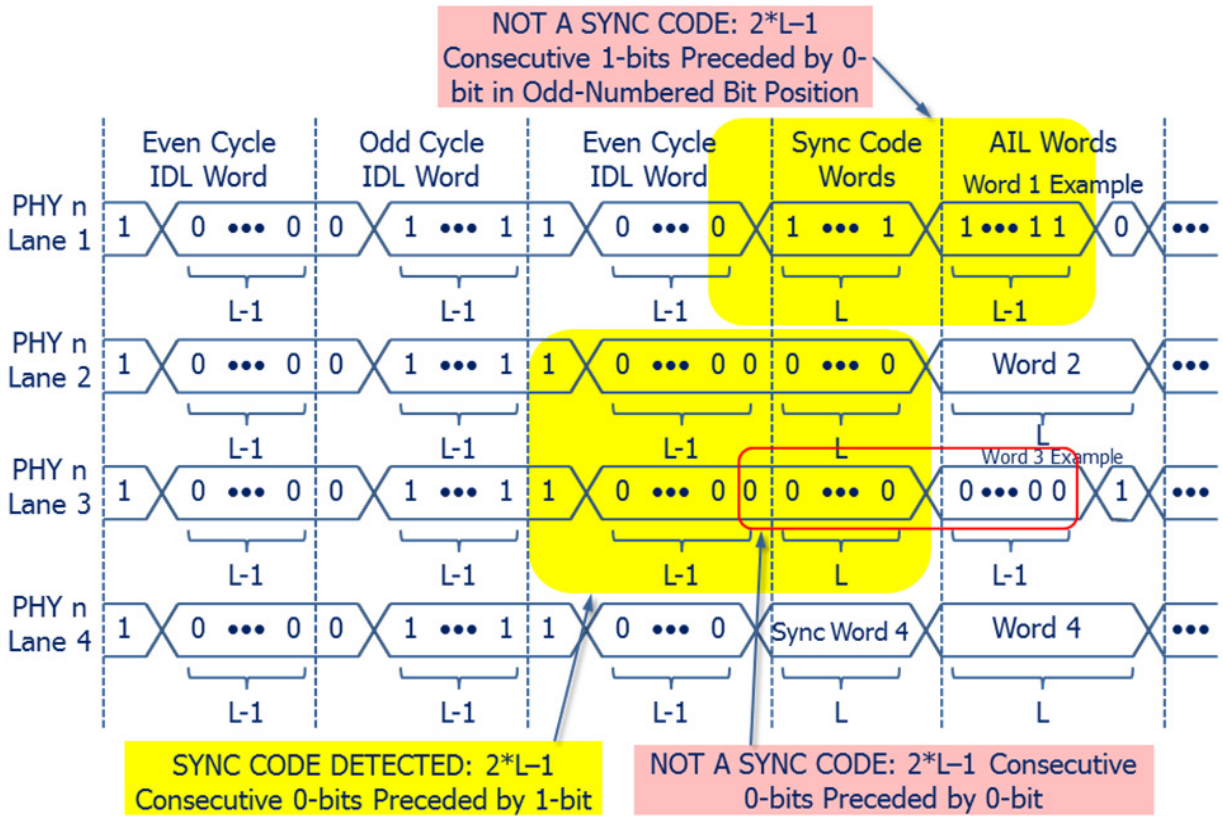


Figure 14. Four Lane Streaming-S Sync Code Immediately Preceded by an Even-Cycle IDL Word and Immediately Followed by an AIL Word of 0xFFFE

In both cases (i.e. Sync Code preceded by either an odd- or even-cycle IDL Word), the final L-bits received on Lane 4 when a Sync Code is detected correspond to Word 4 of the Sync Code. Note that in general, a Streaming-S Receiver cannot anticipate the length of an IDL Word sequence, so it must simultaneously monitor both Lanes 1 and 2 (or alternatively Lanes 1 and 3) for Sync Code occurrences.

The cases described above for a 4-Lane PHY also apply to an 8-Lane PHY, since only the latter's first four data Lanes need be used for Sync Code detection.

Two Active Data lanes per PHY

Figure 15 depicts the case of a Streaming-S Sync Code which has been striped over two data Lanes and is immediately preceded by an odd-cycle IDL Word on each Lane. In this case, a Sync Code may be detected within a PHY by observing 2*L – 1 consecutive 1-bits on Lane 1. Figure 16 depicts the alternate case where the same Sync Code is immediately preceded on each Lane by an even-cycle IDL Word, in which case a Sync Code may be detected by observing 2*L – 1 consecutive 0-bits on Lane 2. Note that for two active Lanes per PHY, it doesn't matter what bit value precedes the run of 2*L – 1 consecutive 1- or 0-bits.

The next L-bits observed on Lane 2 after a Sync Code is detected correspond to Word 4 of the Sync Code. As noted for four active Lanes per PHY, a Streaming-S Receiver with two active Lanes per PHY generally cannot anticipate the length of an IDL Word sequence, so it must simultaneously monitor both Lanes 1 and 2 for Sync Code occurrences.

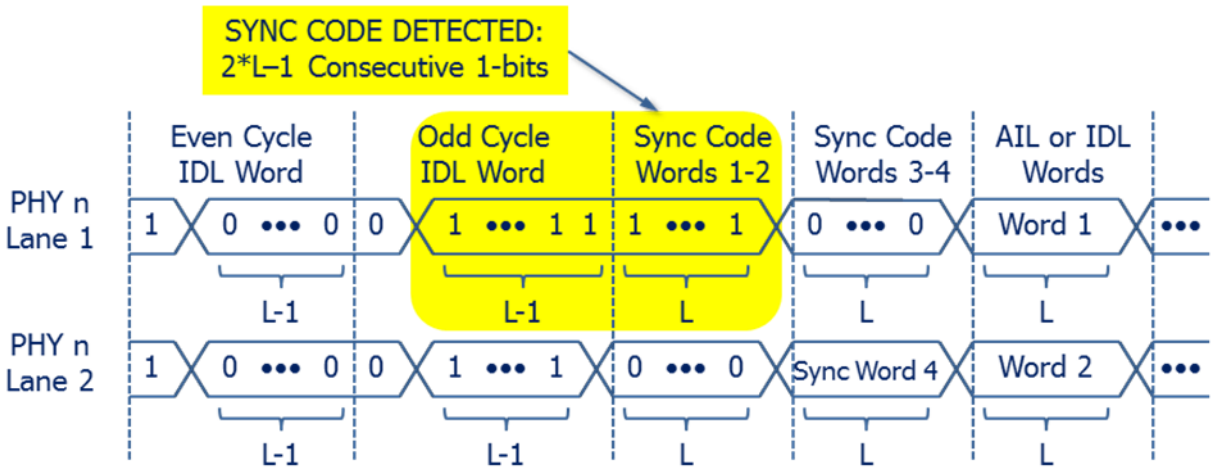


Figure 15. Two Lane Streaming-S Sync Code Immediately Preceded by an Odd-Cycle IDL Word

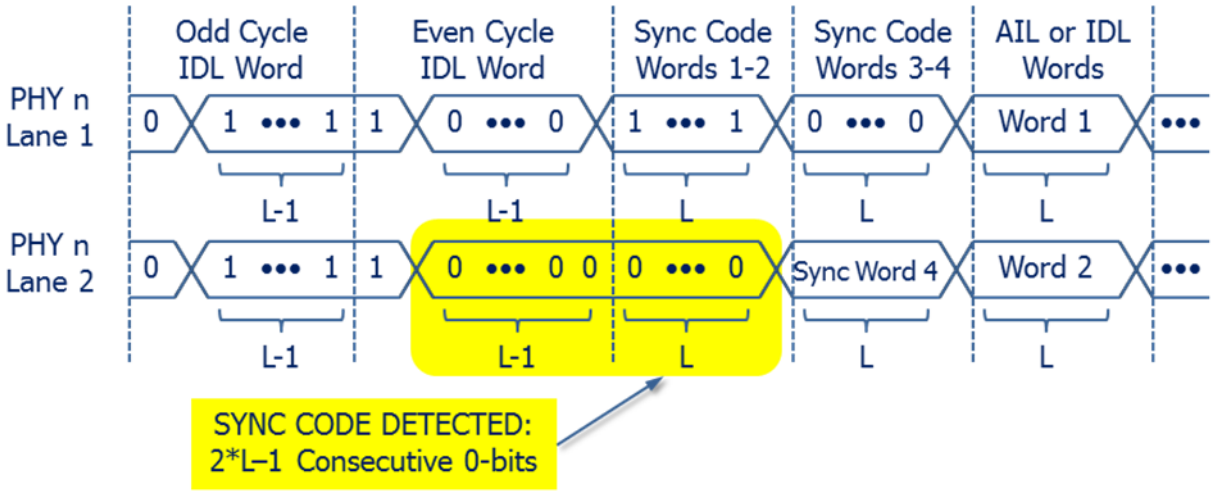


Figure 16. Two Lane Streaming-S Sync Code Immediately Preceded by an Even-Cycle IDL Word

One Active Data Lane per PHY

Finally, as was previously stated, Sync Codes for the single-Lane case may be detected using the same procedure employed for the Packetized- and Streaming-SP Modes. That is, a Sync Code may be detected by observing a stream of 2*L – 1 or 2*L consecutive 0-bits (i.e. if 2*L – 1 consecutive 0-bits are observed, then the protocol guarantees that the next bit will also be 0), and Word 4 of the Sync Code consists of the next L-bits following the 2*L consecutive 0-bits.

ActiveStart-SP8 Mode

As described in section ActiveStart-SP8 Mode, each ActiveStart-SP8 Mode Sync Code is eight L-bit Words long and identically transmitted on each data Lane of every PHY. Similar to SP Mode, an ActiveStart-SP8 Sync Code may be detected using bits received from only a single data Lane within a given HiSPi PHY.

An ActiveStart-SP8 Mode Transmitter using the default IDL Word of all 0-bits will never output an L-bit Word of all 0- or 1-bits during an active line except during an SOF or SOL Sync Code. In particular, as shown in Figure 12, the maximum run length of consecutive 0- or 1-bits which can occur across Word boundaries in an AIL is 2*L – 2. This guarantees that an 8*L-bit sequence matching any of the default ActiveStart-SP8 Sync Codes shown in Table 6, with the exception of the default SOL Sync Code for the MT9E501, cannot occur during an Active Image Line because each of these Sync Codes contains a run of consecutive 0- or 1-bits longer than 2*L – 2. Therefore, these Sync Codes may be robustly detected at any time by observing a sequence of at least 2*L – 1 consecutive 1-bits.

However, there is no guarantee that an 8*L-bit sequence matching the MT9E501 default SOL Sync Code shown in Table 6 will never appear somewhere within an Active Image Line. This is illustrated by the example in Figure 17 which shows how a false SOL Sync Code could theoretically appear in an AIL as a simple bit-wise shift of the 8*L-bit SOL Sync Code pattern. A solution to this problem is to

employ a state machine to detect the specific SOL bit-pattern shown in Table 6, but to enable the state machine only when a Sync Code is expected. (Note that in the interest of uniformity, the same state machine may also be used to detect the MT9E501 default SOF Sync Code, although it is not strictly necessary, as outlined in the previous paragraph.)

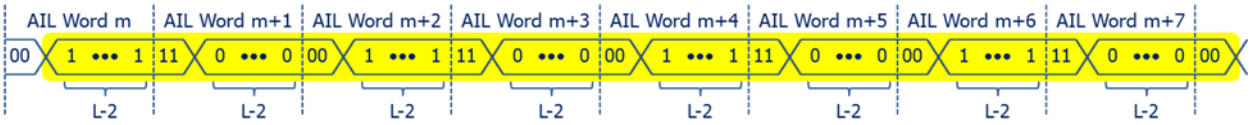


Figure 17. Example Occurrence of False ActiveStart-SP8 Default SOL Sync Code for MT9E501

An ActiveStart-SP8 Sync Code always occurs at the conclusion of an IDL, so MT9E501 default Sync Code detection should be enabled whenever it can be reliably determined that an IDL Word sequence is in-progress. Assuming an IDL consists of at least two Words, a default IDL Word sequence may, in turn, be detected by observing at least 2*L – 1 consecutive 0-bits (i.e. one more than the 2*L – 2 consecutive 0-bits which may occur across Word boundaries in an AIL).

Figure 18 presents another example of legal SOL and SOF Sync Codes requiring special consideration during detection. The SOF and SOL Sync Codes are, respectively, $\{2^L-1, 0, 2^L-1, 0, 2^L-1, 0, 0, 0\}$ and $\{0, 2^L-1, 0, 2^L-1, 0, 2^L-1, 0, 0\}$. These Sync Codes are distinguished in that the first Word of the SOL Sync Code matches the default IDL Word, and the last seven Words of the SOL Sync Code match the first seven Words of the SOF Sync Code (i.e. SOL and SOF "overlap"). The 10-Word sequence shown in Figure 18 depicts an IDL Period followed by the SOF Sync Code; however, this sequence could also be confused with an IDL Period followed by the SOL Sync Code if Word 10 is not also examined.

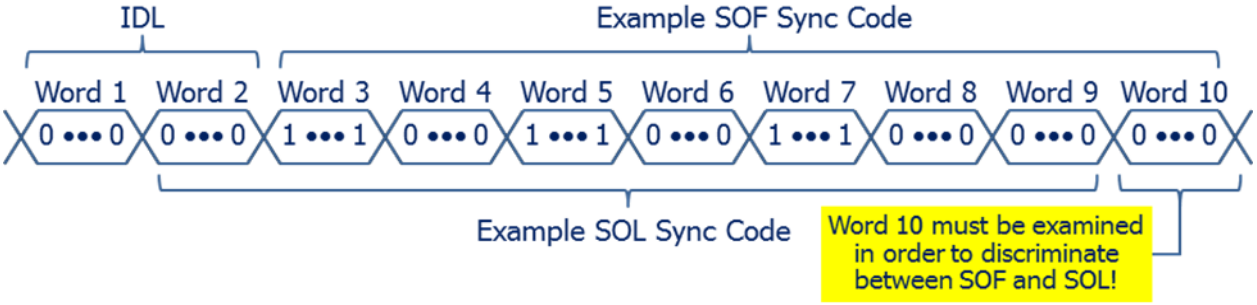


Figure 18. Example of Overlapping ActiveStart-SP8 SOF and SOL Sync Codes

For proper Sync Code detection, the general rule underlying this example is that if Words 1 through k (for $k \le 7$) of an SOF or SOL Sync Code match the IDL Word, and, furthermore, the remaining Words k + 1 through 8 of that Sync Code match Words 1 through 8 − k of the other Sync Code, then the former 8-Word, SOF or SOL Sync Code must be detected by observing a total of nine Words in order to guarantee that the ninth Word is not really Word 8 − k + 1 of the other Sync Code.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:
Literature Distribution Center for ON Semiconductor
19521 E. 32nd Pkwy, Aurora, Colorado 80011 USA
Phone: 303−675−2175 or 800−344−3860 Toll Free USA/Canada
Fax: 303−675−2176 or 800−344−3867 Toll Free USA/Canada
Email: orderlit@onsemi.com

N. American Technical Support: 800−282−9855 Toll Free USA/Canada
Europe, Middle East and Africa Technical Support:
Phone: 421 33 790 2910
Japan Customer Focus Center
Phone: 81−3−5817−1050

ON Semiconductor Website: **www.onsemi.com**

Order Literature: http://www.onsemi.com/orderlit

For additional information, please contact your local Sales Representative