

ADV202 Y/CbCr Merging in Multi-Chip Mode

Rev 0.1 ADI HSC Austin Design Center

The following document contains a design proposal. There are no guarantees that the described design can be implemented in this or any derivative form. Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices.

1.0 General Description

This document describes a method and hardware configuration to merge the Luminance and Chrominance compressed data byte streams from two ADV202 devices (in multi-chip mode) when running in Encode (compression) mode. This method involves capturing of attribute and compressed data from the two devices, generating a JPEG2000-compliant codestream, and end-of-field indication.

1.1 Y/CbCr Merging Method Requirements

This proposed method for the merging of compressed Y and CbCr codestreams requires the following conditions:

- Two ADV202 devices configured in multi-chip mode for separate Y/CbCr encode operation.
- Selection of Attribute Type (ATTRTYPE) 5 when initializing the ADV202 devices.
- The use of Single Transfer DREQ/DACK DMA for Attribute Data reading and Burst Transfer DREQ/DACK DMA for Code Data reading.
- 32-bit or 16-bit Host CPU.
- The use of the HDATA bus in 32-bit Host and Data mode for both ADV202 devices.

1.2 Y/CbCr Merging Method Description

The two ADV202 devices are configured in multi-chip mode using ATTRTYPE=5. Please refer to “Getting Started with the ADV202” document for more information on configuring ADV202 devices. ATTRTYPE=5 attribute format utilizes the Attribute FIFOs in each ADV202 device to provide the codestream attributes for each of the two streams (Y and CbCr) to be merged. The attributes must be read to obtain the byte counts for each codestream partition. This information is then used to generate the ADV202 header and to merge the appropriate number of bytes from each stream into a single JPEG2000-compliant stream. The data streams are read from the Code FIFOs from each device.

The following sections describe the procedure required to perform codestream merging:

1.2.1 Field Header and Partition Byte Counts

The Attribute FIFO provides a header and the number of bytes in each partition of the data for each component (Y and CbCr).

Attribute FIFOs can be read using any of the Host Interface protocols, but the single transfer External DMA protocol is recommended for this purpose. In this DMA mode, the DREQ signal will indicate when Attribute data is available for reading. For each field/frame, four 32-bit words are used as the header, followed by the number of bytes in each partition. An example of the proposed attribute file format is shown below:

TABLE 1. Attribute File Format Example (Y or CbCr)

Description	Example Data
ADV202 Header Word 1 (field delimiter)	0xFFFFFFFF0
ADV202 Header Word 2 (field index)	0x00000001
ADV202 Header Word 3 (COD Style / VFORMAT)	0x01020002
ADV202 Header Word 4 (No. of words in compressed image)	0x0000061A
Number of Partitions (Byte Length Counts) in image	0x00000019
Byte Length of Sum of Main Header, Tile Header, and Partition 1	0x00000123
Byte Length of Partition 2	0x00000456
Byte Length of Partition 3	0x00000257
Byte Length of Partition 4	0x00000034
.....	
Byte Length of Partition 25 (Zero-byte padding)	0x0000001F

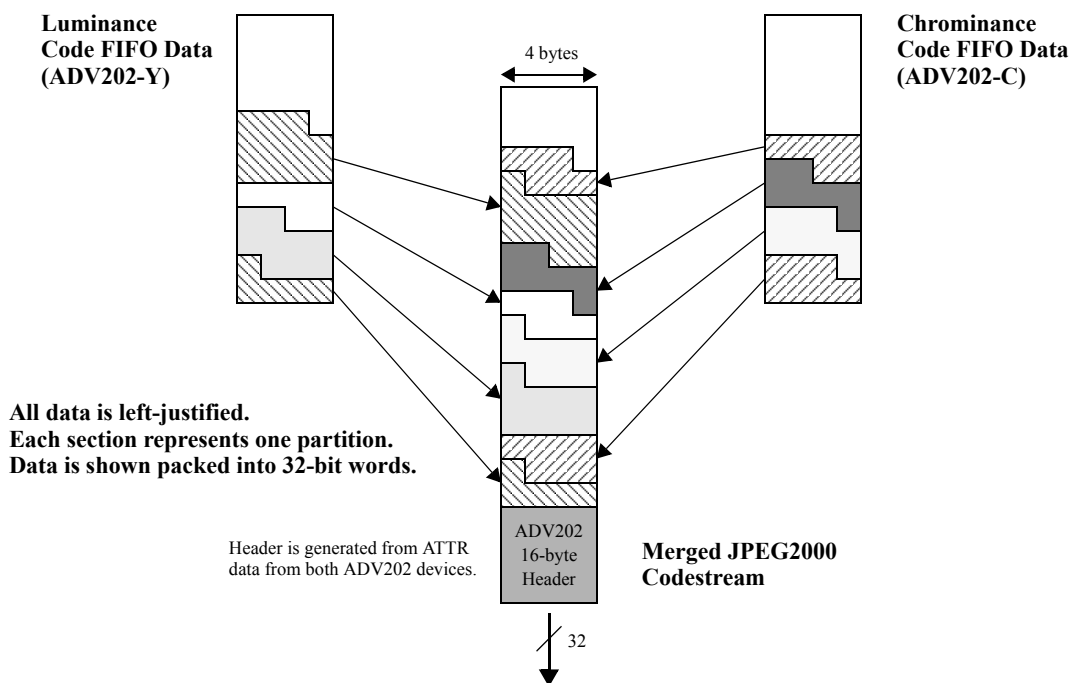
Any of the Host Interface modes can be used to read the Code FIFO, but the most efficient method is Burst Mode DREQ/DACK External DMA mode. For this reason, the last partition byte length contains the number of 0x00 bytes used for padding at the end of the compressed data stream for each field. Each ADV202 adds bytes of 0x00 at the end of each field in order to fill the last burst of data. The burst length is 8 32-bit words. Since the Code FIFOs must contain at least 8 x 32-bit words for each burst, the last partition for each ADV202 device will contain zero to 31 bytes of 0x00.

1.2.2 Merging of Compressed Data Streams

To generate the ADV202 header for the compressed stream, the first 3 words from the header in the attribute data stream for the Luminance ADV202 can be taken directly. The fourth word in the merged stream header is the sum of the fourth words from the Y and CbCr ADV202 devices. This will be the total number of compressed data words for this field/frame in the merged stream.

The partition byte count information from the attribute data is used to merge the Y/CbCr data streams. The codestream is formed by merging the data bytes from the corresponding partitions of the two devices, alternating between Y and CbCr, starting with the Y data.

FIGURE 1. Y/CbCr Data Merging/Packing Example



The last partition in each field from each ADV202 contains a number of 0x00 bytes used for padding at the end of the compressed data stream. Each ADV202 adds bytes of 0x00 at the end of each field in order to fill the last burst of data. The number of 0x00 bytes at the end of each ADV202's compressed data stream is between 0 and 31 bytes. This means that after merging, the number of 0x00 bytes will be between 0 and 62 bytes. Below are some examples of what the end of a field's data may look like after merging ("x" represents valid non-zero code data):

TABLE 2. Examples of the End of Merged Compressed Data Stream

Example 1	Example 2	Example 3	Example 4
xxxxxxxx	xxxxxxxx	FFD90000	xxxxxxxx
xxxxxxxx	xxxxxxxx	00000000	xxxxxxxx
xxxxxxxx	xxxxxxxx	00000000	xxxxxxxx
xxxxxxxx	xxxxxxFF	00000000	xxxxxxxx
xxxxxxxx	D9000000	00000000	xxxxxxxx
xxxxxxxx	00000000	00000000	xxFFD900
xxxxxxxx	00000000	00000000	00000000
xxxxxFFD9	00000000	00000000	00000000

The chrominance ADV202 provides the end-of-code (EOC) marker. The EOC marker bytes are included in the next-to-last partition byte count from the chrominance ADV202.

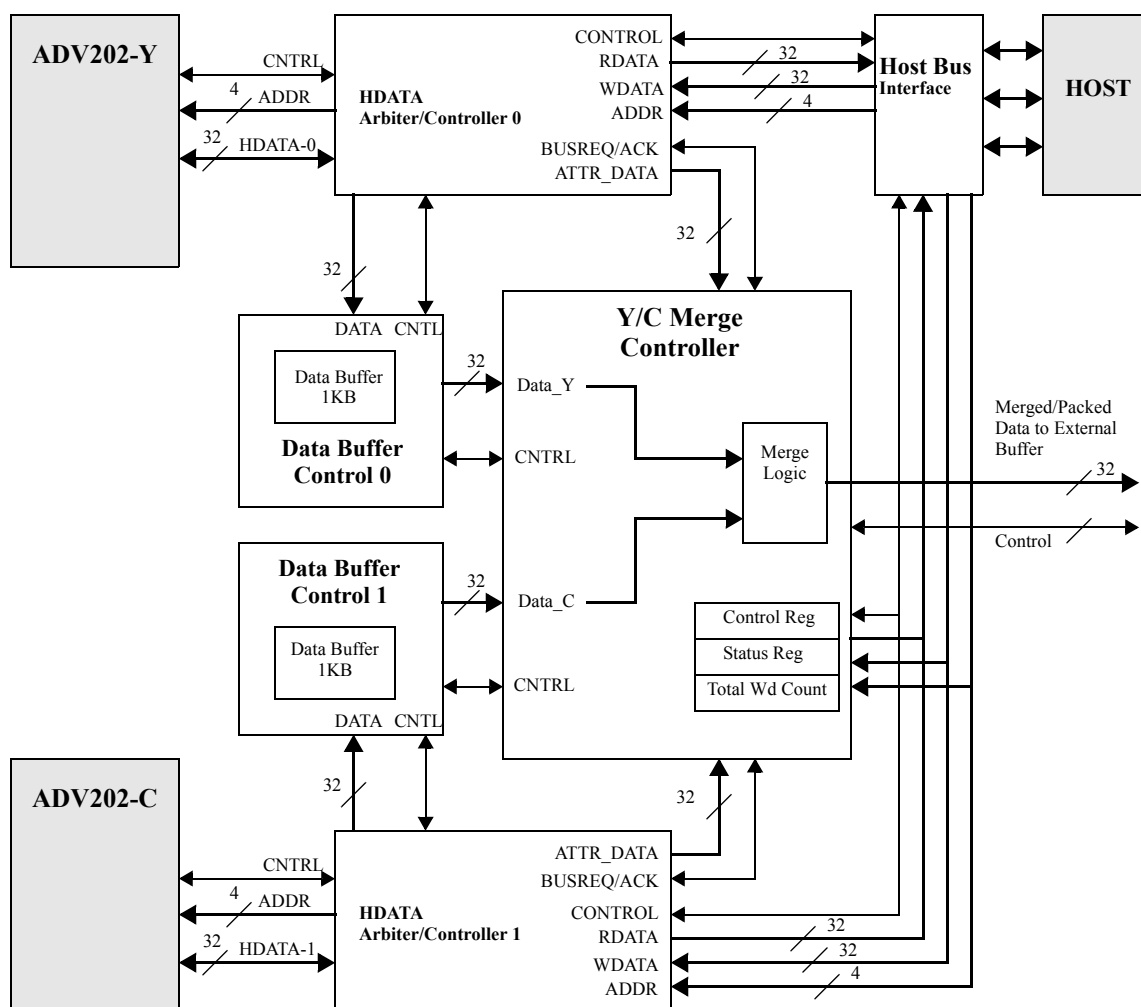
1.3 Proposed Hardware Configuration

This section describes a possible hardware implementation of this Y/CbCr codestream data merging method. This description and associated Verilog code should only be used as a reference. Analog Devices Inc. has not implemented this in hardware, and there is no guarantee that the proposed hardware implementation will function as described.

The proposed hardware configuration has the following features:

- Independent HDATA Arbiter/Controller units handle register, attribute data and code data accesses.
- The host interface in each ADV202 is used in 32-bit mode. Access to all ADV202 internal registers is performed through the 32-bit interface, and the HDATA Arbiter/Controller data ports are also 32-bits wide.
- The Host Bus accesses follow the Direct Register access protocol described in the ADV202 datasheet. The attribute and data accesses will make use of the External DMA channels (using DREQ and DACK signals in single and burst modes).
- Maximum data transfer rate from ADV202 devices to the Y/C Merge logic is 100MBytes/sec using a system (Y/C Merge) clock of 50MHz. Using this system clock, the HDATA burst accesses can occur at 25MHz. The Y/C Merge block is capable of sustaining an average throughput (merged stream output) of over 65MBytes/sec (50Mhz system clock).
- Using the ADV202 External DMA DREQ/DACK protocol, indication of code/attribute data availability is observable via DREQx signals.
- Separate 1KB code data FIFO buffers provide decoupling between the Y/C merging logic and the DMA transfer logic. 8-word deep attribute FIFOs decouple the merging control logic from the Single DMA bus accesses.
- Interface to an external FIFO buffer for capturing of the merged compressed data stream. Capability to stall the merging process via “external buffer full” indication.

FIGURE 2. Y/C Merge Hardware Configuration



Note: Shaded blocks are not part of the design described in this document.

1.4 General Functional Requirements

1.4.1 ADV202 and Y/C Merge Logic Operating Frequency

For High-Definition video data compression, the internal frequency of the ADV202 must be set to 148~150Mhz. For highest reliability, it is strongly recommended that the MCLK input is the same 74.25Mhz clock used for VCLK. By setting the value of the ADV202 PLL_LO register to 0x84, the required internal frequency of 148.5Mhz can be generated.

The Y/C Merge block operating frequency will depend on the logic implementation. Using a reference clock of 50MHz, the Y/C Merge logic can support an average throughput of over 65Mbytes/sec. By setting the HI_SPD_EN0/1 bits in the ycm_ctl_reg, the speed of the bus is one

half of the reference clock. With a 50Mhz reference clock, the HDATA bus frequency would be 25MHz, which would support a maximum throughput of up to 100MBytes/sec from each ADV202 device.

1.4.2 Luminance/Chrominance Compressed Data Capture

The Data Buffer Controllers (DBUF0/DBUF1) capture the two separate compressed data streams from the ADV202 devices. They must capture the data at a rate which is adequate for the compression ratio chosen for the application's data rates. Using the burst mode DMA protocol to output compressed data, *each* ADV202 can support peak transfer speeds of up to 200MBytes/sec. However, a maximum transfer rate of 100MBytes/sec (25MHz HDATA Bus) will satisfy performance requirements for this application.

1.4.3 Merged Codestream Output

The merged JPEG2000 codestream is available from the output data port (CODEDATA) as packed 32-bit words. It must be captured by an external buffer. Buffer write enable and first/last word indication is provided. The capability to stall the merging process is also available by using the EXTBUF_FULL signal.

1.5 Detailed Block Implementation (Verilog HDL Reference)

The following sections describe the functions of each hardware block in the Y/C Merge reference Verilog HDL code. The Verilog HDL model names are provided in parentheses.

The Verilog HDL models and this description are provided for reference only. Analog Devices Inc. does not guarantee operation.

1.5.1 Host Bus Interface (HostBus_if.v)

This block interfaces the Host Bus Controller signal protocol to the YC_Merge internal protocol. It provides access to the YC_Merge control registers and the ADV202 registers.

1.5.2 HDATA Bus Arbiter/Controllers (hdata_cntl.v)

Independent HDATA Bus Arbiter/Controllers are required for each ADV202 device. These blocks (hdata_cntl.v) perform the following functions:

1.5.2.1 Host Bus Interface and Y/C Merge Controller (YC_Merge) Bus Request Arbitration

This logic provides arbitration for accesses from these two masters. The Host Bus Interface requests read/write accesses to ADV202 control registers. The YC_Merge requests are read-only (attribute and codestream data). The Host Bus Interface requests are direct register accesses using the Acknowledge (ACKn) signal. The YC_Merge requests are external DMA requests using the Data Request / Data Acknowledge (DREQn/DACKn) protocol. Attribute data requests use Single Transfer External DMA mode. The compressed data requests use Burst Transfer DMA External mode. Each burst transfers eight 32-bit words.

1.5.2.2 HDATA Control

This logic provides all signals to perform direct and external DMA accesses for each ADV202 device.

Attribute FIFO accesses will be handled using the Single Transfer External DMA mode in the ADV202. Code FIFO accesses will use Burst Transfer External DMA mode with a Burst Length of 8 x 32-bit words.

The operating frequency of the HDATA bus can be set to half the system clock frequency by setting the HI_SPD_EN0/1 bits in the ycm_ctl_reg register. By default the HDATA bus speed is one fourth of the system clock frequency.

1.5.3 Data Buffer Controllers (dbuf.v)

These blocks serve as data buffers between the ADV202 devices and YC_Merge. The DBUF0/DBUF1 blocks acquire code data from the HDATA arbiter/controllers and store it in a 32-bit wide FIFO. The depth of these FIFOs is 256 32-bit words. The data from the two FIFOs (luminance and chrominance) is read by YC_Merge for codestream merging. Each DBUF block will continuously request data from its corresponding ADV202 when there is space available in the FIFOs.

1.5.4 Y/CbCr Merging Block (ycm.v)

The Y/CbCr Merging Block provides the following functions:

1.5.4.1 Attribute Read Logic (ycm_attr.v)

The YC_Merge Attribute Read modules read the attribute data from their corresponding ADV202 and capture it into the Attribute FIFOs (attr_fifo0/attr_fifo1). These FIFOs are currently 8 entries deep, but this may be changed as necessary. These FIFOs are implemented as 8 x 32-bit register files.

1.5.4.2 Attribute Parsing, Byte Packing, and Merging Logic (ycm_merge.v)

The Y/C Merge block reads attribute data from the attribute FIFOs (attr_fifo0/attr_fifo1) to determine the luminance and chrominance byte counts for each partition. The field delimiters and indexes for each field are read and the luminance and chrominance values are compared. If the comparisons fail, the merging state machine will enter an error condition and stop processing data. The number of words in the field are read from each ADV202's attribute data and added to obtain the total number of words (luminance+chrominance). The first three words from the luminance ADV202 attribute FIFO are used for the ADV202 header. The total number of words is inserted as the fourth word in the ADV202 header.

The number of partitions and the partition byte lengths are loaded in counters. These counters are updated as the data from the two streams is packed/merged.

The Y/CbCr code data from the ADV202s is alternatively merged into a single stream. The bytes will be packed into 32-bit words. For each field, the merged stream will end at an 8-word 32-bit boundary. Additional bytes filled with 0x00 are used to fill the last burst.

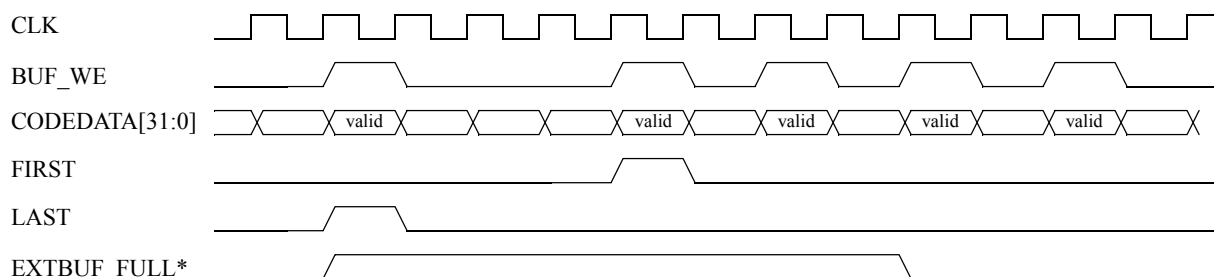
1.5.4.3 External Buffer Interface

The following signals are provided to interface with an external buffer to capture the merged compressed data stream.

TABLE 3. External Buffer Interface Signals

Name	Direction	Description
BUF_WE	Output	External Buffer Write Enable. Active high signal indicates that the data value on CODEDATA is valid and will be written to the external device.
CODEDATA[31:0]	Output	Merged Compressed Codestream Data. 32-bit merged/packed code-stream. Valid in cycle when BUF_WE is asserted.
FIRST	Output	Indicates First Word of Field. Asserted in same cycle with corresponding CODEDATA.
LAST	Output	Indicates Last Word of Field. Asserted in same cycle with corresponding CODEDATA.
EXTBUF_FULL	Input	Asserted when external device is nearing full capacity. This will stall the merging process. The YC Merge logic may write up to 7 additional 32-bit words after the detection of EXTBUF_FULL. External device must be designed to avoid an overflow condition.

FIGURE 3. External Output Buffer Interface Signal Timing



* Up to 7 additional words may be written by the YC_Merge block after detection of EXTBUF_FULL assertion.

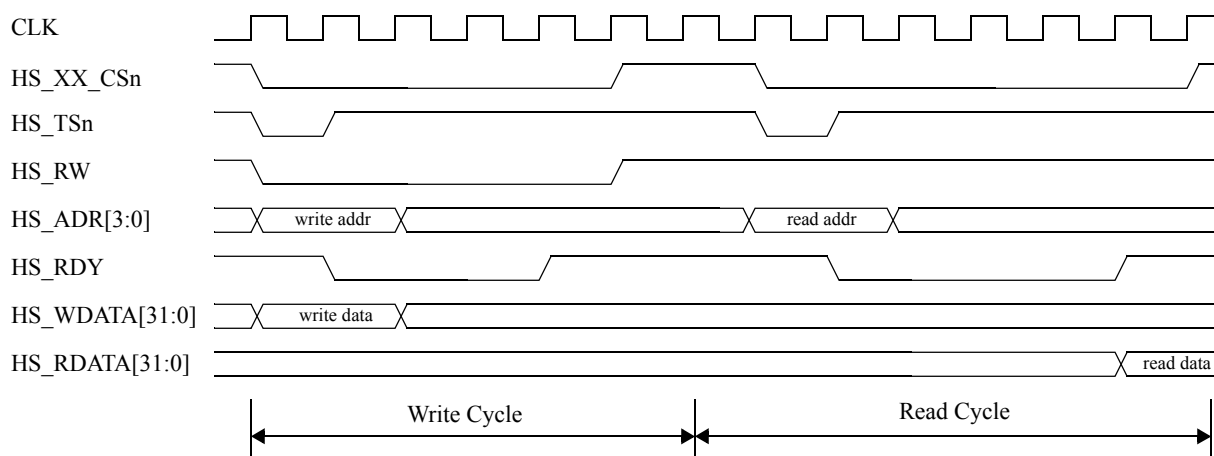
1.5.4.4 Host Bus Interface

The Host Bus interface signal names and timing are provided below. This provides access to control registers in the ADV202 devices and the YC_Merge Controller.

TABLE 4. Host Bus Interface Signals

Name	Direction	Description
HS_H0_CS _n HS_H1_CS _n HS_YC_CS _n	Input	Chip Selects for ADV202-0/1 or YC_Merge. Must remain asserted until HS_RDY (Ready) is asserted.
HS_TS _n	Input	Host Access Transfer Start.
HS_ADR[3:0]	Input	Host Access Address for ADV202-0/1 and YC_Merge registers.
HS_RW	Input	Host Access Read/Write indication.
HS_RDY	Output	Ready. Indicates transfer is complete.
HS_WDATA[31:0]	Input	Host Access Write Data.
HS_RDATA[31:0]	Output	Host Access Read Data.

FIGURE 4. Host Bus Interface Signal Timing



1.5.4.5 YC_Merge Register Definitions and Programming Sequence

The following table provides the YC_Merge register bit assignments and functionality.

TABLE 5. YC_Merge Registers - All Registers set to zero on RESET.

Addr	Register	Description	Bits	Function
0x0	ycm_ctl_reg	YC_Merge Control	[0]	DB_EN0 - Set to Enable Data Buffer Controller data reading from ADV202-0
			[1]	DB_EN1 - Set to Enable Data Buffer Controller data reading from ADV202-1
			[4]	ATTR_EN0 - Set to Enable Attribute reading from ADV202-0
			[5]	ATTR_EN1 - Set to Enable Attribute reading from ADV202-1
			[8]	HDAT_EN0 - Set to Enable HDATA Controller for ADV202-0
			[9]	HDAT_EN1 - Set to Enable HDATA Controller for ADV202-1
			[10]	HI_SPD_EN0 - Set to Enable High-Speed HDATA Accesses for ADV202-0.
			[11]	HI_SPD_EN1 - Set to Enable High-Speed HDATA Accesses for ADV202-1.
			[12]	MERGE_EN - Set to Enable YC_Merge State Machine
0x1	ycm_stat_reg	YC_Merge Status	[0]	External Buffer Full Detected. Write “1” to clear.
			[4]	Field Header Comparison Error detected. Write “1” to clear.
			[5]	Field Index Comparison Error detected. Write “1” to clear.
			[6]	Partition Count Comparison Error detected. Write “1” to clear.
0x2	TOTAL_WDCNT	Total Word Count	[31:0]	Total word count for current field/frame.

The following sequence for programming the YC_Merge/ADV202s should be followed:

- Enable the two HDATA Bus Controllers by setting the ycm_ctl_reg to 0x0000_0300. This must be done to enable all accesses to the ADV202 devices.
- Program the ADV202 EDMOD0 (Burst Transfer Mode - Length 8) and EDMOD1 (Single Transfer Mode) registers for both ADV202 devices. Do not enable the channels at this time.
- Enable the Data Buffer Controllers (DBUF0/DBUF1) and the Attribute Acquisition logic by writing the ycm_ctl_reg with 0x0000_0333.
- Enable the YC_Merge Controller by writing the ycm_ctl_reg with 0x0000_1333. If high-speed HDATA bus operation is desired, write ycm_ctl_reg with 0x0000_1F33.
- Enable the four DMA channels in the ADV202 devices and start the encode firmware.

- The HDATA controllers will begin to read data when the corresponding DREQx signals are asserted. On the Output Buffer Interface, the merged codestream should become available within a few cycles after the first Attribute headers and the first words of code data are read from both ADV202 devices.

1.5.5 Verilog Testbench Blocks

The following blocks are part of the Verilog testbench to allow simulation and verification of the Y/C Merge Controller. These blocks are *not* part of the hardware implementation and are *not* synthesizable.

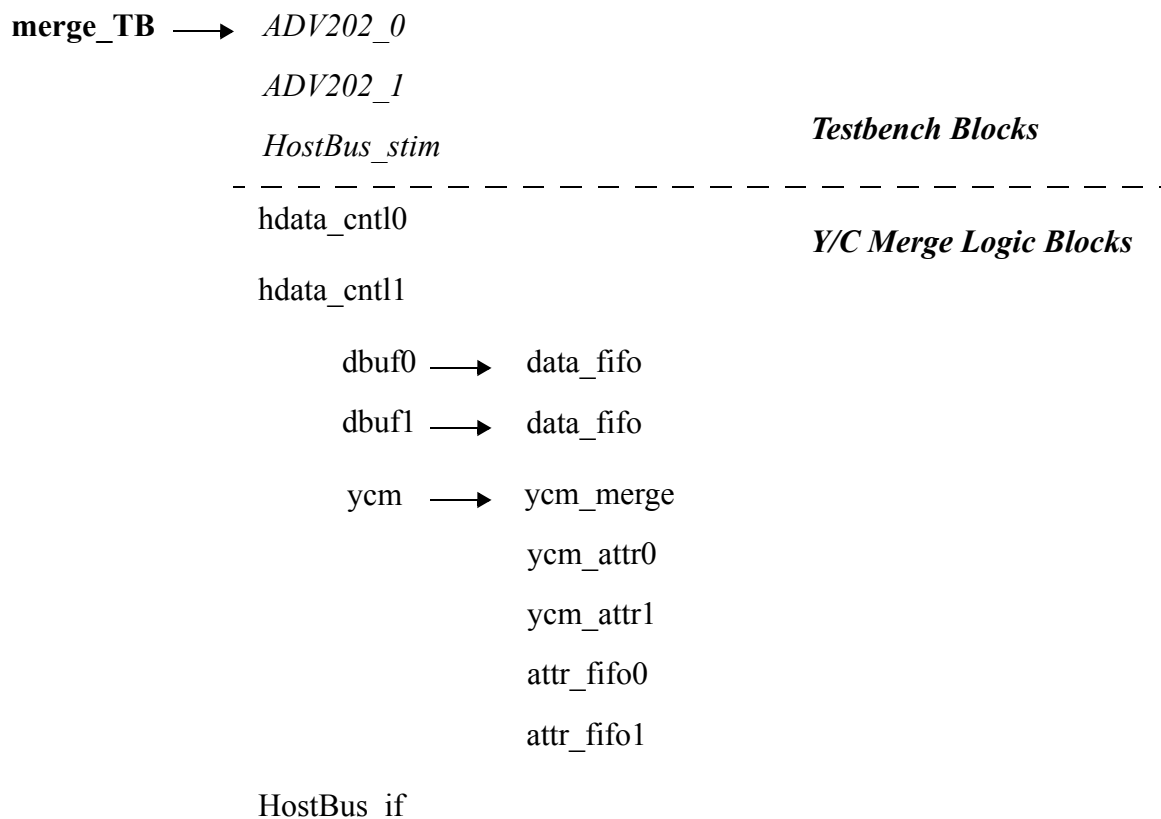
1.5.5.1 Host Bus Stimulus (*HostBus_stim.v*)

This non-synthesizable module provides stimulus to simulate Host Bus Interface accesses to the two ADV202 devices and YC_Merge registers. Verilog “tasks” are used to program accesses. Random accesses to the ADV202 devices are simulated during Y/C codestream merging.

1.5.5.2 ADV202 Emulation (*ADV202_emu0.v & ADV202_emu1.v*)

These non-synthesizable modules simulate the behavior of the two ADV202 devices for this application. The data streams are contained in the files: luma.attr, luma.data, chroma.attr, chroma.data. Device 0 handles luminance, and Device 1 handles chrominance data. For simulation only, 16 direct registers are provided. These registers only provide read/write functionality to verify the correct operation of the HDATA Controllers/Arbiters.

1.5.6 YC_Merge Verilog HDL Testbench Hierarchy



1.6 Other Design Issues

1.6.1 Code Data Buffer Sizes

In this reference design, a FIFO depth of 256 32-bit words (1KB) has been selected for the Code Data FIFO buffers (DBUF0/DBUF1). Specific requirements for a given application may require adjustment of the depth of these FIFOs.

1.6.2 Merging Throughput

Using a 50MHz clock and the HDATA buses in high-speed mode, the YC_Merge block can output the merged stream at a rate greater than 65MBytes/sec. However, actual throughput will be limited by the encode speed of the ADV202 devices, Host accesses of ADV202 registers, and External Buffer stall requests.

1.6.3 ADV202 Deadlock Condition

A watchdog timer may be required for Host Bus accesses to avoid a deadlock condition. This condition could occur if an abnormal register access (i.e. invalid address) is attempted to an ADV202. If this condition occurs, all accesses to the ADV202 device will be halted.