CS2420

2048/4096/8192 Point FFT/IFFT

AMPHION™
Virtual Components for the Converging World

The CS2420 is an online programmable 2048 - 8192-point FFT/IFFT core. It is based on the radix-4 algorithm and performs 2048-point to 8192-point FFT/IFFT computation in three computation passes. A block diagram of the core is shown in.

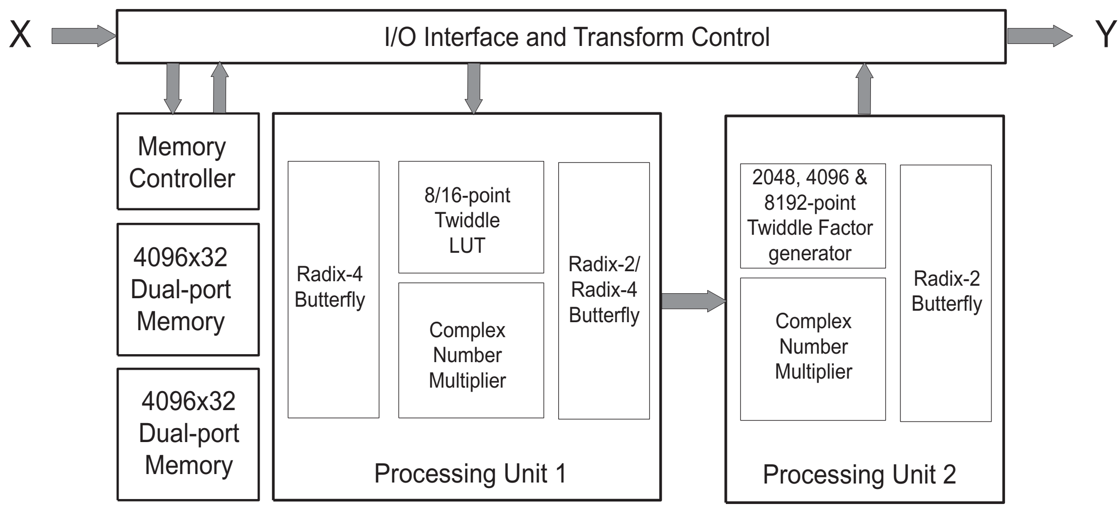

Figure 1: CS2420 Block Diagram

FEATURES

- **On-line programmable FFT/IFFT core**
- **16-bit complex input/output in two's complement format (32-bit complex word)**
- **16-bit twiddle factors generated inside the core**
- **18-bit internal accuracy**
- **Programmable shift down control**
- **Mixed radix-8/radix-16/radix-32 architecture**
- **Simultaneous loading/downloading supported**
- **Both input and output in normal order**
- **No external memory required**
- **Optimized for both ASIC and FPGA technologies with the same functionality**

KEY METRICS

- **Logic:** 59k gates
- **Memory:** <3.9mm^2
- **Total area:** <4.5mm^2

See Table 8 - 10 for more details

APPLICATIONS

- **Image processing**
- **Atmospheric imaging**
- **Spectral representation**
- **OFDM modulation scheme for DVB-T (Ref: ETS 300 744)**

CS2420 I/O DESCRIPTION

Table 1 describes the input/output ports (shown graphically in Figure 2) for the CS2420 FFT/IFFT core. Unless otherwise stated all signals are active High, and bit (0) is the least significant bit.

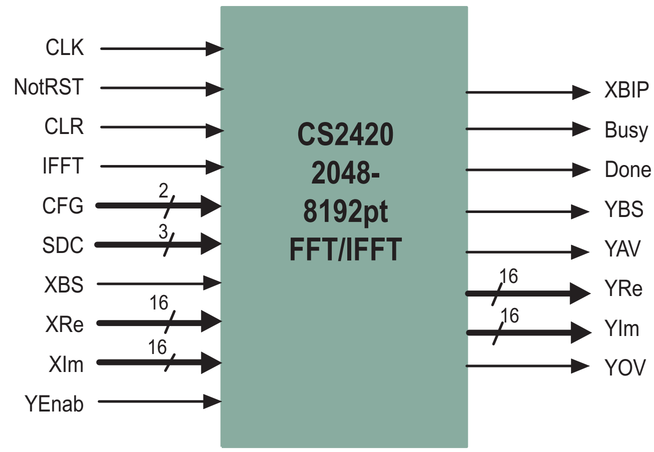

Figure 2: CS2420 Symbol

Table 1: I/O Description for the CS2420

Name	I/O	Width	Description
CLK	I	1	Clock signal, rising edge active
NotRST	I	1	Asynchronous global reset signal, active LOW
CLR	I	1	Clear (synchronous reset) and programming signal, active HIGH
IFFT	I	1	Programming signal specifying the transform type, loaded when CLR is active. 1:IFFT; 0:FFT
CFG	I	2	Programming signal specifying the transform size, loaded when CLR is active. 01:2k; 10:4k; 11:8k
SDC	I	3	Programming signal specifying the number of bits for the additional scaling down operation, loaded when CLR is active
XRe	I	16	Real component of input data X, in two's complement format
XIm	I	16	Imaginary component of input data X, in two's complement format
XBS	I	1	Input data X block start signal, active HIGH, associated with the first input data of the N-point block. The remaining N-1 data of the N-point data block are loaded into the core in the following N-1 clock cycles in the natural order.
YEnab	I	1	Output data Y enable control, active HIGH
XBIP	O	1	Output signal indicating loading X is in Progress. XBIP goes to HIGH the next clock cycle when XBS is active and returns to LOW when the last data of the N-point block is loaded into the core. XBS is ignored when it is HIGH.
Busy	O	1	Output signal indicating the transform in progress (busy). It goes to HIGH the next clock cycle when XBS is active and returns to LOW when the core is ready to accept the next input data block. XBS is ignored when it is HIGH.
Done	O	1	Output signal indicating the transform result is available. It goes to HIGH when the core is ready to output transform result and returns to LOW when YEnab is asserted to download the result.

Table 1: I/O Description for the CS2420

Name	I/O	Width	Description
YBS	O	1	Output data Y block start signal, active HIGH, asserted when the first data of the N-point transformed block is on the output port. The remaining N-1 data of the N-point transform result come out of the core in the following N-1 clock cycles in the natural order.
YAV	O	1	Output data Y available indicator, active HIGH, asserted with every data of the N-point transform result
YRe	O	16	Real component of output data Y, in two's complement format, valid only when YAV is HIGH
YIm	O	16	Imaginary component of output data Y, in two's complement format, valid only when YAV is HIGH
YOV	O	1	Output data Y overflow signal, active HIGH, asserted when overflow occurs when the transform is performed. It is reset when a new transform starts and is associated with the N-point block.

GENERAL DESCRIPTION

The CS2420 performs N-point FFT/IFFT following the equations below:

FFT: $$Y(k) = \frac{1}{2^{7+SDC}} \sum_{n=0}^{N-1} X(n) W_N^{-nk} \quad ,k=0, 1, 2,.. \qquad [1]$$

IFFT $$Y(k) = \frac{1}{2^{7+SDC}} \sum_{n=0}^{N-1} X(n) W_N^{nk} \quad ,k=0, 1, 2, \qquad [2]$$

Where N is 2048, 4096 or 8192, SDC is the scaling down control signal, X(n) is the complex input data and Y(k) the complex output data. Both the real and imaginary components of input X(n) and output Y(k) are 16-bit two's complement numbers.

In order to achieve highest data throughput rate possible, CS2420 employs fixed-point arithmetic operations and pre-scaling strategy to handle possible overflow in computation. The core has 7-bit unconditional scaling down operations and 7-bit controlled scaling down operations specified by input signal SDC, giving the user the necessary gain control means required in the application.

CS2420 employs two computation units in pipeline to perform the transform in three passes, using a mixed radix-8/radix-16 and radix-32 algorithm. Processing unit 1 consists of a radix-4 butterfly, an 8-point/16-point twiddle LUT, a complex number multiplier and a selectable radix-2/radix-4 butterfly. It performs one 16-point transform or two 8-point transforms in 16 clock cycles according to the control signals from the transform controller. Processing unit 2 consists of a 2048/4092/8192-point twiddle factor generator, a complex number multiplier and a radix-2 butterfly. In the first two passes of the computation, it takes the output of processing unit 1 and performs twiddle operation. In the last pass, it either directs the output of processing unit 1 to the controller when the core is in 2048- or 4096-point transform mode or performs 32-point twiddle and radix-2 operations when the core is in 8192-point mode.

Programming CS2420 is performed when the synchronous reset signal CLR is active. The programming signals, namely, IFFT, CFG and SDC, are loaded into the core. These set up the transform type, transform size and scaling down controls.

CS2420 performs the three computation passes continuously in a pipelined manner without wasting any clock cycle, due to the fixed-point arithmetic and pre-scaling strategy used. The core can perform the transform and loading input data/downloading transform result with a 4x clock. For example, an 8192-point transform with data/IO can be performed with 32768 clock cycles.

The scaling down operation is spread into various computing passes and computation units. The two processing units use 18-bit arithmetic operations and detect the possible overflow in computation. When overflow occurs, the processing units flag it to the controller and saturate the overflow results on the fly.

The core has separate I/O indicator and control signals to support simultaneous or separate loading input data and downloading the transform result. The input data is burst in to and the transformed result is burst out from CS2420 on block-by-block basis.

FUNCTIONAL DESCRIPTION

GENERAL

CS2420 performs a mixed decimation in frequency (DIF), radix-8, radix-16 and radix-32, forward or inverse Fast Fourier Transform on 2048-point, 4096-point or 8192-point complex data block. The transform is scheduled in three computation passes. Data is loaded into the core in normal sequential (natural) order. The transform result comes out from the core also in the natural order.

The core is on-line programmable on the transform type, transform size and scaling down control. The input and output data and the twiddle factor wordlengths are selected such that it can be used in a wide range of applications.

The core computes the transform using fixed-point arithmetic with programmable shift down control on each computation pass to handle the possible wordlength growth and overflow in the transform. This achieves the maximal accuracy possible while maintaining the desired dynamic range for the output.

The internal 8K 32-bit word dual port memory is organised in two banks with 4K words each. In 2048-point and 4096-point transform mode, only one bank is enabled. This is to improve power consumption of the core when it is operating for the smaller transform size.

The core is a synchronous design with all the flip-flops being triggered at the rising edge of the clock signal CLK.

PROGRAMMING THE CORE

Programming CS2420 is performed when the core is synchronously reset. This is done through asserting signal CLR and applying appropriate values to input ports CFG, IFFT and SDC.

Port CFG and IFFT specify the transform size and transform type. Table 2 lists the CFG and IFFT value for programming the core to different transform sizes and types.

The core performs 7-bit unconditional shifting down on the internal data during the transform. However, theoretically the 2048-point, 4096-point and 8192-point FFT may have up to 12, 13 and 14 bit word growth in total, respectively. The CS2420 core can perform up to 7 bits controlled shift down operation to avoid possible overflow and to allow the transform gain to be controlled. This is programmed through port SDC. The total number of shift down bits decides the transform scaling down factor. Table 3 lists the SDC values for programming the scaling factor.

After the global asynchronous reset signal RST is applied, the core is reset to the default mode: 2048-point FFT without the additional shifting operation. Programming the core can be performed at any time subsequently. The programming signals are valid only when CLR is HIGH. This is illustrated in Figure 3. It is noted that when CLR is applied the core is reset as well.

Table 2: Programming Transform Type and Size

Port CFG	Port IFFT	Transform Type	Transform Size
00	0	Reserved	Reserved
00	1	Reserved	Reserved
01	0	FFT	2048-point
01	1	IFFT	2048-point
10	0	FFT	4096-point
10	1	IFFT	4096-point
11	0	FFT	8192-point
11	1	IFFT	8192-point

Table 3: Programming Scaling Factor

Port SDC	Fixed Shifting (bits)	Additional Shifting (bits)	Scaling Factor ($2^{-(7+SDC)}$)
000	7	0	1/128
001	7	1	1/256
010	7	2	1/512
011	7	3	1/1024
100	7	4	1/2048
101	7	5	1/4096
110	7	6	1/8192
111	7	7	1/16384

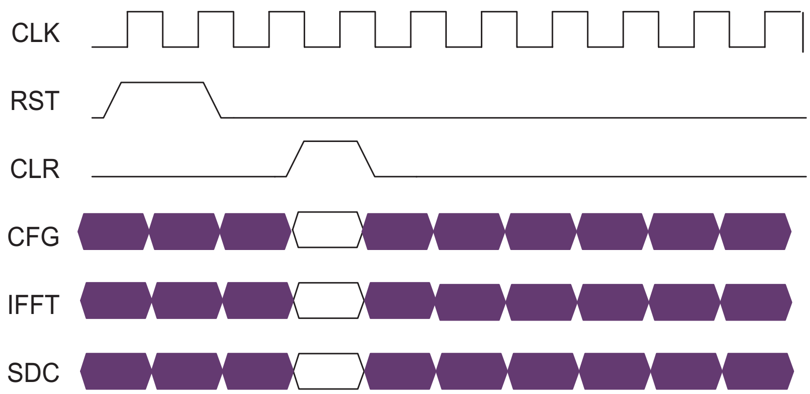

Figure 3: Configuration Timing

INPUT AND OUTPUT DATA FORMAT

The input complex number data is represented by 16-bit real and imaginary components, namely XRe and XIm, in the two's complement format.

The input data is burst into the core in the normal order, i.e., X(0) enters the core first, followed immediately in the next clock cycle by X(1), and then X(2), and so on so forth. It takes 2048, 4096 and 8192 clock cycles for a data block to enter the core for transforms of 2048-point, 4096-point and 8192-point, respectively.

The transform result is also complex numbers. They are represented by 16-bit real components YRe and imaginary components YIm in the two's complement format.

The output data is burst out from the core when the transform has been performed to the stage that allows the result to be output and the output port is enabled. The result from the core is also in the normal order, i.e., Y(0) first, followed by Y(1), Y(2) and so on so forth.

TRANSFORM COMPUTATION

The transform is scheduled to complete in three passes. In each pass the controller fetches the intermediate data from the internal dual port memory, sends it to the two processing units, collects the computation results from the processing units and writes them back to the memory for the next pass or for the output.

In the first two passes, Processing Unit 1 performs 16-point FFT on the intermediate data from the memory, using a Cooley-Tukey radix-4 decimation-in-frequency (DIF) algorithm. This involves two radix-4 butterflies and a 16-point twiddle operation. The intermediate result value may grow by a factor of up to 4*5.657, representing 4 to 5 bits word length growth. Processing Unit 2 performs twiddle operations on the

16-point FFT result from Processing Unit 1 for the programmed transform size.

In the third pass, Processing Unit 1 performs 16-point FFT when the transform size is 4096-point or 8192-point, using the same algorithm as that used in the first two passes. It performs 8-point FFT when the transform size is 2048-point, using a mixed radix-4 and radix-2 DIF algorithm. For 8192-point transform, Processing Unit 2 performs 32-point twiddle operation and a further radix-2 operation on the result from Processing Unit 1. This, together with the operations of Processing Unit 1, effectively forms a radix-32 operation. For 2048-point and 4096-point transforms, Processing Unit 2 performs no operation in the third pass. The transform operation performed in each pass is summarised in Table 4.

CS2420 performs scaling down operation by right shifting the intermediate result in the three passes, according to the scaling down control programmed. Table 5 lists the relationship between the programming input signal SDC and the number of scaling down bits performed in the three passes. It is noted that for 2048-point, 4096-point and 8192-point transform, there is no overflow in the computation when the total number of shifting bits is equal to or more than 12, 13, and 14 bits, respectively.

Table 4: Transform Operation in Each Pass

Transform Size	Pass 1	Pass 2	Pass3
2048-point	Radix-16	Radix-16	Radix-8
4096-point	Radix-16	Radix-16	Radix-16
8192-point	Radix-16	Radix-16	Radix-32

Table 5: Number of Right Shifting Bits in Each Pass

SDC	Pass 1	Pass 2	Pass 3	Total
000	3	3	1	7
001	4	6	1	8
010	4	3	2	9
011	5	3	2	10
100	5	4	3	11
101	5	4	4	12
110	5	4	4	13
111	5	4	5	14

FIXED WORD LENGTH AND ACCURACY

CS2420 uses fixed-point arithmetic to perform the transform. All the arithmetic operations involved have 16 bits or higher accuracy. The twiddle factors (sine and cosine values), which are generated by the core internally, have 16-bit accuracy. At the end of each computation pass, the result is rounded to 16 bits. Figure 4 illustrates the word lengths at various computation stages in the CS2420 core.

The rounding technique is employed to achieve the maximal computation accuracy possible for the given word lengths. When the intermediate value is derived from the twiddle multiplication result, the output from the butterflies is scaled down, or the intermediate result is right shifted, the core performs the round-to-the-nearest operation to keep the loss of accuracy minimal.

Table 6 gives the simulation results on the transform accuracy of the CS2420 core. These results are obtained by applying100 blocks of 16-bit random input data to the core and the scaling down control is set such that there is just no overflow in the computation, i.e., the output magnitude is maximised while no overflow occurs. The 16-bit output data from the core is compared with the result of double precision FFT model. The error is measured in terms of the output LSB weight. It is noted that when overflow occurs the transform accuracy will be decreased severely.

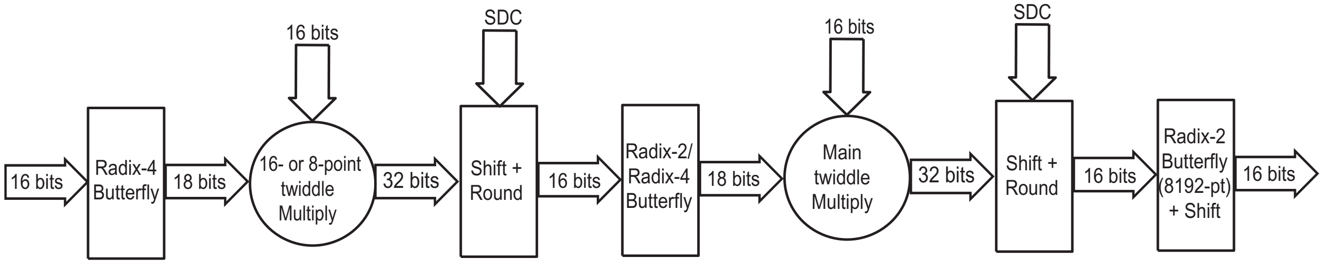

Figure 4: Word Length in Arithmetic Operations

Table 6: Simulation Results of Transform Accuracy

Transform Size	2048-point	4096-point	8192-point
SCD setting	001	001	010
Scaling Factor	1/256	1/256	1/512
Number of complex data samples compared	204800	409600	819200
Maximal output Magnitude	16884	23234	16651
Maximal Error	5	9	10
Average Absolute Output	2268.0	3773.7	2668.0
Average Absolute Error	0.527	0.681	0.589
Mean Square Error	0.610	0.932	0.730
Average SNR	74.1dB	74.8dB	73.1dB

LOADING INPUT AND DOWNLOADING RESULT

Loading the input data is performed under the control of signal XBS. Signal XBS should be asserted when the output signal XBIP and BSY are LOW. It indicates the first data of the N-point data block and the data is clocked in on the clock rising edge. The remaining N-1 point data are loaded in the successive N-1 clock cycles in the natural order.

When the core starts to load an N-point data block, signal XBIP goes to HIGH to indicate that loading a data block is in progress. Signal XBS will be ignored when XBIP is HIGH. When the last data of the block is loaded into the core, signal XBIP returns to LOW and signal Busy remains HIGH to indicate the transform computation is in progress. Signal XBS is still ignored in this case until Busy returns to LOW.

The CS2420 core starts the transform prior to the completion of loading the N-point data block when the required data has been loaded, i.e., the input data loading is overlapped with the first computation pass. This compensates for the latency introduced by the pipelined computation units so that the input data loading and the three computation passes can be completed in 4*N clock cycles.

Signal Done goes to HIGH when the transform result is available. Downloading of the transform result is started by asserting the input signal YEnab when Done is HIGH. Signal Done returns to LOW when downloading is started. The first sample of the transform result comes out from the core in the natural order two clock cycles later after YEnab is asserted. Output signal YAV is asserted when the data on port YRe and YIm are valid and output signal YBS is asserted when the first sample of the N-point result is on the output port. The output data is burst out from the core in N clock cycles.

Downloading the result can be overlapped with the third computation pass to achieve 4*N clock cycles operation, if input signal YEnab is asserted as soon as the output signal Done goes to HIGH. Loading the next data block can be started as soon as output signal Busy returns to LOW.

Figure 5 shows the functional timing for the 4*N clock cycle I/O and transform operation. It is noted that the input signal YEnab can be constantly asserted and if so the transform result will be automatically downloaded when it is available.

It is noted that the core waits for YEnab being asserted when signal Done goes to HIGH to start the downloading process, allowing the user to control the transform data flow. The system clock rate is not restricted to the 4*N cycles and can be any rate higher than 4X the data rate. In this case if the downloading result has been completed but loading the next block is not started, signal Done will go to HIGH again to indicate the transform result in the internal memory is still available and can be downloaded again. This feature can be utilised in C-OFDM modulation systems to perform the guard interval insertion.

Figure 6 shows the operating flowchart for the CS2420 core.

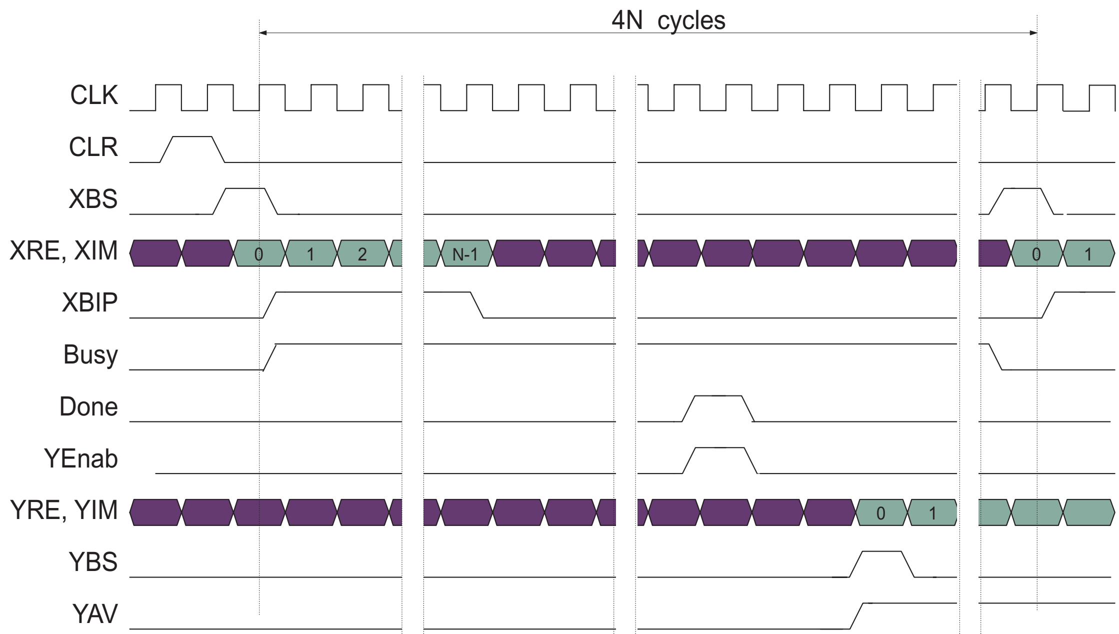

Figure 5: 4N Clock Cycle I/O and Transform Timing

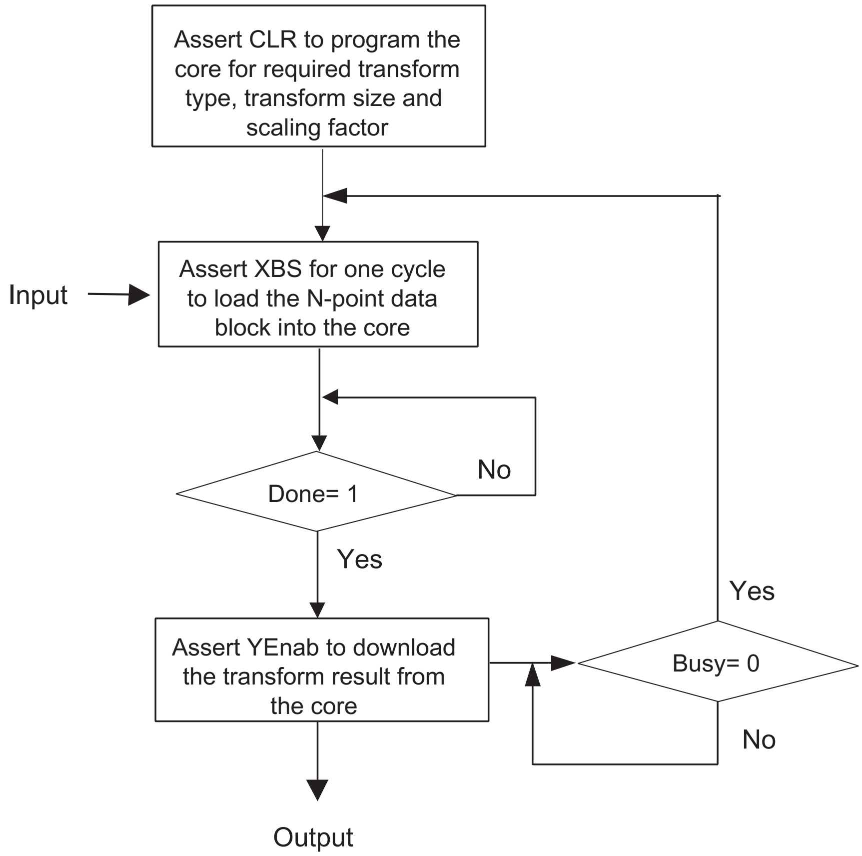

Figure 6: CS2420 Operating Flowchart

OVERFLOW HANDLING

CS2420 keeps tracking the numeric values during the transform computation. If overflow occurs, due to the insufficient number of shifting down bits programmed for the given input data, the overflow value is saturated and the overflow flag signal (YOV) is asserted to alert the application system.

The overflow signal is flagged on-the-fly when the computation is in progress. It is automatically reset when a new transform is started. It should be noted that as there is an overlap between the third computation pass and the downloading transform result in the 4*N cycle operating mode; if the overflow occurs on the last few computations it may not be indicated until the computation has been completed. This is very unlikely to happen in practical applications.

PROCESSING TIME AND LATENCY

The processing time, defined from when the last data of a data block is loaded into the core to when the transform has been completed, is a function of the transform size. It is equivalent to the time interval from when output signal Busy goes to HIGH to when it returns to LOW and is measured in number of clock cycles listed. The real transform time depends on the clock frequency.

The transform period includes the transform time and the data I/O time. It indicates the number of clock cycles required for the core to perform one transform with input data loading and transform result downloading. The minimum transform period is obtained by asserting input signal YEnab as soon as the output signal Done goes to HIGH and by starting the next data block as soon as output signal Busy returns to LOW.

Table 7 lists the transform time and minimum period for different transform size.

Table 7: CS2420 Processing Time and Transform Period

Transform Size	Processing Time (Clock cycles)	Minimum Transform Period (Clock cycles)
2048-point	6144	8192
4096-point	12288	16384
8192-point	24576	32768

DESIGN METHODOLOGY SUPPORT

The Amphion ASVCs support industry standard design flows. The process for integrating the CS2420 into a design flow is shown in the following diagram. Contact Amphion for information on compatibility of the deliverables with specific EDA tools.

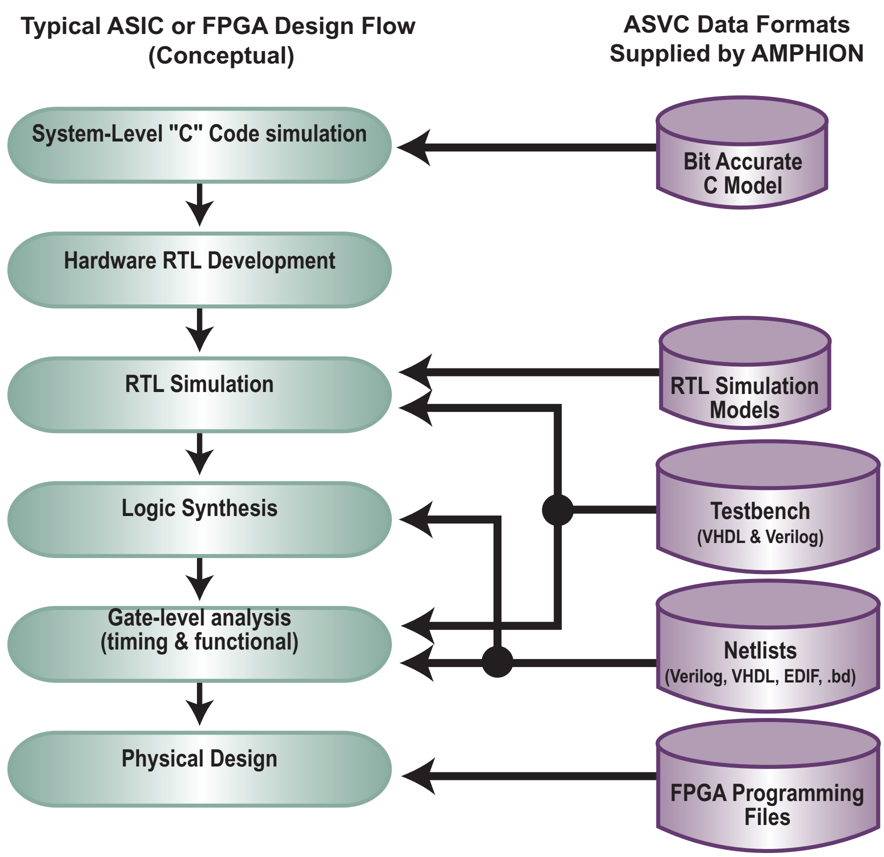

Figure 7: ASVC Design Data Formats Supplied by Amphion

PERFORMANCE AND SIZE

Performance and size of CS2420 depend on the target technology and a wide range of process technologies are supported. In this datasheet the CS2420 has been targeted to three different technologies, namely, the TSMC 180nm ASIC process (CS2420TK), the Xilinx Virtex device(CS2420XV) and the Altera Apex20K device (CS2420AA). All the three have the same functional behaviour and timing. Their performance and size are summarised below. These are subject to synthesis settings and the actual target device. They are therefore provided for information only.

CS2420TK

CS2420TK is the implementation of CS2420 on TSMC 180nm 2.5V standard cell library. When synthesising, the worst case operating conditions are used. The actual gate counts depend on the timing constraints used and if scan-insertion is enabled. The following tables list the performance, size and transform time.

Table 8: Performance and Size of CS2420TK

Timing Constraints (Clock Period)	Logic Area	Equivalent Gates	Memory Area 2 x (32 x 4096 dual port)	Total Area
10ns (100MHz) without scan-insertion	589,665 μm^2	58.97K	3,849,981 μm^2	4,439,646 μm^2
6.5ns (153MHz) without scan insertion	603,749 μm^2	60.38K	3,849,981 μm^2	4,453,730 μm^2
6.5ns (153MHz) with scan-insertion	649,534 μm^2	64.96K	3,849,981 μm^2	4,499,515 μm^2

CS2420XV

CS2420XV is the implementation of CS2420 on the Xilinx Virtex device. The following tables list its performance, size and transform time. These figures may vary if a different device is used.

Table 9: Performance and Size of CS2420XV

Device	Number of 4-input LUTs	Number of slices	Number of Block RAMs	Maximal Clock Frequency
XCV600E-7	5,814	3,758	66	50.0MHz

CS2420AA

CS2420AA is the implementation of CS2420 on the Altera Apex device. The following tables list its performance, size and transform time. These figures may vary if a different device is used.

Table 10: Performance and Size of CS2420AA

Device	Number of Logic Cells	Number of ESBs	Maximal Clock Frequency
EP20K600E-1	8,583	134	43.9 MHz

ABOUT AMPHION

Amphion (formerly Integrated Silicon Systems) is the leading supplier of speech coding, video/ image processing and channel coding ASVCs for system-on-a-chip (SoC) solutions in the telecommunications/ Internet, consumer / communications and wireless markets.

Web: www.amphion.com

Email: info@amphion.com

CORPORATE HEADQUARTERS
Amphion Semiconductor Ltd
50 Malone Road
Belfast BT9 5BS
Northern Ireland, UK

Tel: +44 28 9050 4000
Fax: +44 28 9050 4001

WORLDWIDE SALES & MARKETING
Amphion Semiconductor, Inc
2001 Gateway Place, Suite 130W
San Jose, CA 95110

Tel: (408) 441 1248
Fax: (408) 441 1239

EUROPEAN SALES
Amphion Semiconductor Ltd
CBXII, West Wing
382-390 Midsummer Boulevard
Central Milton Keynes
MK9 2RG England, UK

Tel: +44 1908 847109
Fax: +44 1908 847580

CANADA & EAST COAST US SALES
Amphion Semiconductor, Inc
Montreal
Quebec
Canada

Tel: (450) 455 5544
Fax: (450) 455 5543

SALES AGENTS

Voyageur Technical Sales Inc
6205 Airport Road
Building A, Suite 300
Toronto, Ontario
Canada L4V1E1

Tel: (905) 672 0361
Fax: (905) 677 4986

Phoenix Technologies Ltd
3 Gavish Street
Kfar-Saba, 44424
Israel

Tel: +972 9 7644 800
Fax: +972 9 7644 801

SPINNAKER SYSTEMS INC
Hatchobori SF Bldg. 5F 3-12-8
Hatchobori, Chuo-ku
Tokyo 104-0033 Japan

Tel: +81 3 3551 2275
Fax: +81 3 3351 2614

JASONTECH, INC
Hansang Building, Suite 300
Bangyidong 181-3, Songpaku
Seoul Korea 138-050

Tel: +82 2 420 6700
Fax: +82 2 420 8600

SPS-DA PTE LTD
21 Science Park Rd
#03-19 The Aquarius
Singapore Science Park II
Singapore 117628

Tel: +65 774 9070
Fax: +65 774 9071

04/02 Publication #: DS2420 v1.2