# CS2461

## 64-Point Block Based FFT/IFFT

Virtual Components for the Converging World

The CS2461 is an online programmable, block-based architecture 64-point FFT/IFFT core. This highly integrated application specific core computes the FFT/IFFT based on radix-4 algorithm in three computation passes. The CS2461 is available in both ASIC and FPGA versions that have been handcrafted by Amphion for maximum performance while minimizing power consumption and silicon area.

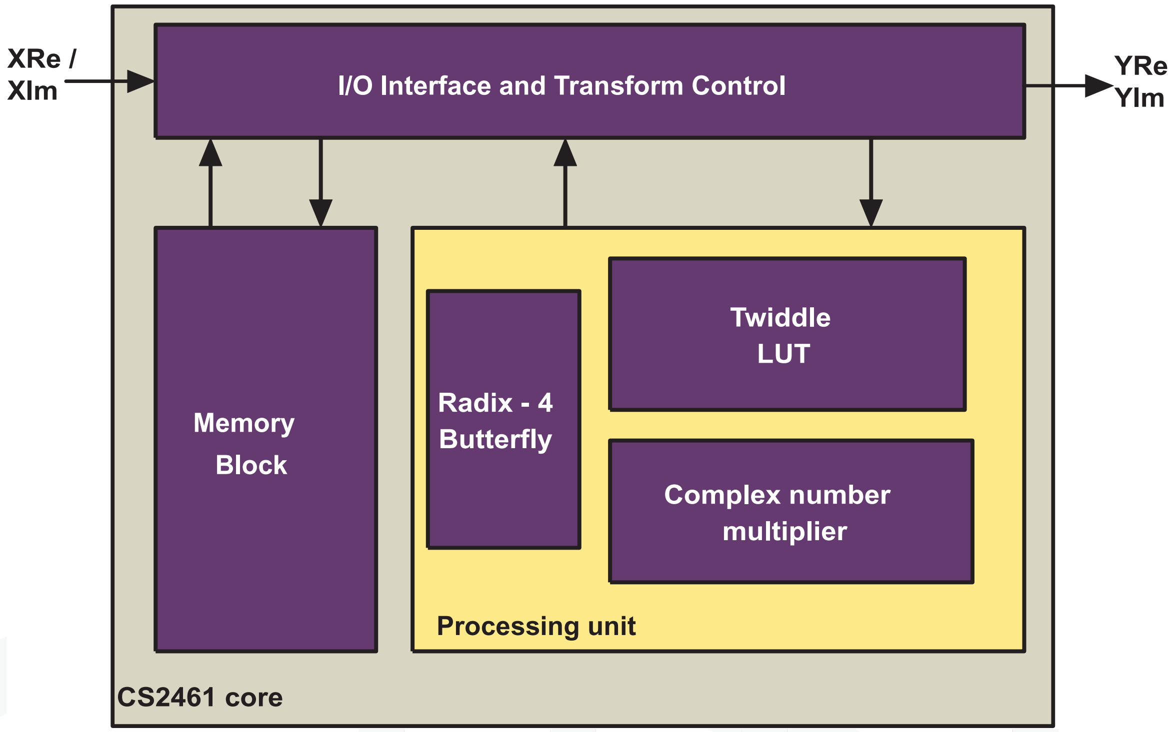


**Figure 1: CS2461 64-Point FFT/IFFT Block Diagram**

## FEATURES

- On-line programmable FFT/IFFT core
- 12-bit complex input/output in two's complement format (24-bit complex word)
- 13-bit twiddle factors generated inside the core
- 15-bit fixed-point internal arithmetic operation
- Programmable shift down control
- Radix-4 architecture
- Transform performed in three computation passes with zero-waiting
- 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
- Fully synchronous design

## APPLICATIONS

- OFDM modulation scheme for WLAN IEEE 802.11a and HiperLAN2
- Image processing
- Atmospheric imaging
- Spectral representation

## FAST FOURIER TRANSFORM

FFT (Fast Fourier Transform) and IFFT (Inverse Fast Fourier Transform) are algorithms computing $2^P$-point discrete Fourier transform and inverse discrete Fourier transform, as defined below.

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

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

Where $N=2^P$ and $W_N = e^{-j2\pi / N}$.

The computational complexity of FFT and IFFT is proportional to $N\log_R N$, where R is the radix base on which FFT/IFFT is performed. The higher the radix, the less number of multiplication is required, however the more simultaneous multiple data access is required which causes the circuits to be more complicated. The radix-4 algorithm offers a balance between the computational and circuit complexity and is often used in construction of higher radix FFT computation units when designing high performance FFT/IFFT hardware.

## CS2461 SYMBOL AND PIN DESCRIPTION

Figure 2 and Table 1 provide the CS2461 block based 64-point FFT/IFFT core symbol, and the I/O interface descriptions respectively. Unless otherwise stated, all signals are active high and bit(0) is the least significant bit.

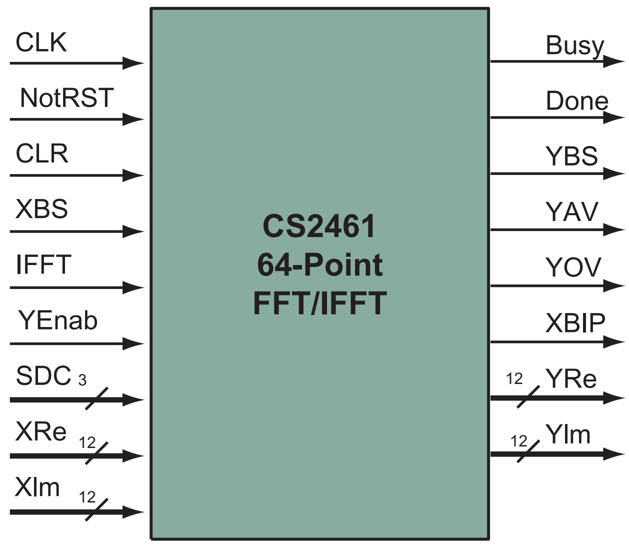


**Figure 2: CS2461 Symbol**

**Table 1: CS2461 - 64-Point FFT/IFFT Interface Signal Definitions**

| Name | I/O | Width | Description |
|---|---|---|---|
| CLK | 1 | 1 | Clock signal, rising edge active |
| NotRST | 1 | 1 | Asynchronous global reset signal, active LOW |
| CLR | 1 | 1 | Clear (synchronous reset) and programming signal, active HIGH |
| IFFT | 1 | 1 | Programming signal specifying the transform type, loaded when CLR is active |
| SDC | 1 | 3 | Programming signal specifying the number of bits for the scaling down operation, loaded when CLR is active |
| XRe | 1 | 12 | Real component of input data X, in two's complement format |
| XIm | 1 | 12 | Imaginary component of input data X, in two's complement format |
| XBS | 1 | 1 | Input data X block start signal, active HIGH, associated with the first input data of the 64-point block. The remaining data of the 64-point data block is loaded into the core in the following clock cycles in the natural order. |
| YEnab | 1 | 1 | Output data Y enable control, active HIGH |
| XBIP | 0 | 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 64-point block is loaded into the core. XBS is ignored when it is HIGH. |
| Busy | 0 | 1 | Output signal indicating the transform in progress (busy). It goes to HIGH the next clock cycle when the last data of the 64-point block is loaded into the core and returns to LOW when the core is ready to accept the next input data block. XBS is ignored when it is HIGH. |

Table 1: CS2461 - 64-Point FFT/IFFT Interface Signal Definitions

| Name | I/O | Width | Description |
|---|---|---|---|
| Done | 0 | 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. |
| YBS | 0 | 1 | Output data Y block start signal, active HIGH, asserted when the first data of the 64-point transformed block is available on the output port. The remaining data of the 64-point transform result is available at the output of the core in the following clock cycles in natural order. |
| YAV | 0 | 1 | Output data Y available indicator, active HIGH, asserted with valid data of the 64-point transform result |
| YRe | 0 | 12 | Real component of output data Y, in two's complement format, valid only when YAV is HIGH |
| YIm | 0 | 12 | Imaginary component of output data Y, in two's complement format, valid only when YAV is HIGH |
| YOV | 0 | 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 64-point block. |

# FUNCTIONAL DESCRIPTION

The CS2461 core performs a decimation in frequency (DIF), radix-4, forward or inverse Fast Fourier Transforms on a 64-point complex data block. The transform is scheduled in three computation passes and the data is loaded into the core in normal sequential (natural) order. The transform result is outputted from the core also in the natural order. The core is on-line programmable on the transform type and scaling down control. It's input/output data and the twiddle factor wordlengths are selected such that it can be used in a wide range of applications.

The CS2461 computes the transform using fixed-point arithmetic with programmable shift down control on each computation passes 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. This 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 CS2461 is performed when the core is synchronously reset. This is done through asserting signal CLR and applying the appropriate signals to the input ports IFFT and SDC, where port IFFT specifies the transform type i.e. FFT/IFFT. Table 2 lists the FFT/IFFT value for programming the core to appropriate transform type.

Table 2: Programming Transform Type

| Port IFFT | Transform Type |
|---|---|
| 0 | FFT |
| 1 | IFFT |

The core performs conditional shifting down on the internal data during the 64-Point transform. Theoretically the 64-Point FFT may have up to a total of 7-bits word growth. The CS2461 core can perform up to 7-bit controlled shifting down operation to avoid possible overflow and also 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.

Table 3: Programming Scaling Factor

| Port SDC | Controlled Shifting (Bits) | Scaling Factor ($2^{-(SDC)}$) |
|---|---|---|
| 000 | 0 | 1 |
| 001 | 1 | 1/2 |
| 010 | 2 | 1/4 |
| 011 | 3 | 1/8 |
| 100 | 4 | 1/16 |
| 101 | 5 | 1/32 |
| 110 | 6 | 1/64 |
| 111 | 7 | 1/128 |

After the global asynchronous reset signal RST is applied, the core is reset to the default mode: 64-point FFT with a 7-bit shifting operation. Programming the core can be performed at any time subsequently. The programming signals are valid only when CLR is asserted. This is illustrated in Figure 3. It is noted that when CLR is applied the core is reset as well.

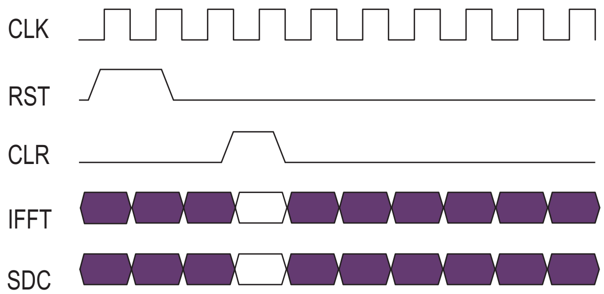


**Figure 3: Configuration Timing**

## INPUT AND OUTPUT DATA FORMAT

The input complex number data for the CS2461 is represented by 12-bit real and imaginary components, namely XRe and XIm, in the two's complement format. The input data is loaded 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), etc. In total it takes 64 clock cycles for a data block to enter the core for FFT/IFFT processing. The transformed data is represented by complex numbers which consist of a 12-bit real component YRe and a 12-bit imaginary component YIm both 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), etc in consecutive clock cycles.

## 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 processing unit, fetches the computation results from the processing unit and writes the result back to memory for the next pass or for the output. The CS2461 employs a Cooley-Tukey radix-4 decimation-in-frequency (DIF) to compute the FFT/IFFT. This algorithm requires the calculation of radix-4 butterflies and twiddle multiplications in multiple passes. Theoretically the intermediate result value of a radix-4 butterfly with twiddle operation may grow by a factor of up to 5.657. This represents up to three-bit wordlength growth.

The rounding technique is employed to achieve the maximum possibe computation accuracy. When the intermediate value is derived from the twiddle multiplication result, or the input to the butterfly is scaled down, round-to-the-nearest operation is performed. This gives the maximal computation accuracy possible for the given wordlength.

The CS2461 performs scaling down operation by right shifting the intermediate result in the four passes, according to the scaling down control programmed. Table 4 lists the relationship between the programming input signal SDC and the number of scaling down bits performed in the four passes. It is noted that there is no overflow in the computation when the total number of shifting bits is equal to 7 bits.

.

**Table 4: Number of Right Shifting Bits in Each Pass**

| SDC | Pass 1 | Pass 2 | Pass 3 | Total |
|---|---|---|---|---|
| 000 | 0 | 0 | 0 | 0 |
| 001 | 1 | 0 | 0 | 1 |
| 010 | 1 | 1 | 0 | 2 |
| 011 | 2 | 1 | 0 | 3 |
| 100 | 2 | 1 | 1 | 4 |
| 101 | 2 | 2 | 1 | 5 |
| 110 | 3 | 2 | 1 | 6 |
| 111 | 3 | 2 | 2 | 7 |

## FIXED WORD LENGTH AND ACCURACY

The CS2461 core uses fixed-point arithmetic to perform the transform. The twiddle factors (Sine and Cosine values), which are generated by the core internally, have 13-bit accuracy. At the end of each computation pass, the result is rounded to 12 bits. Figure 4 illustrates the word lengths at various computation stages in the CS2461 core.

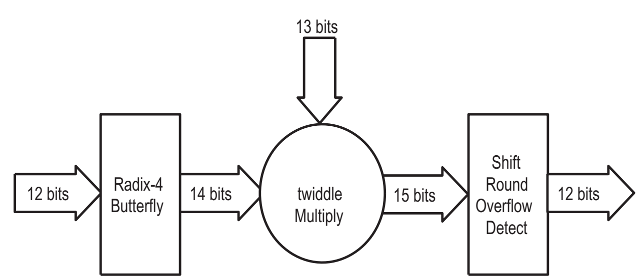


**Figure 4: : Wordlength in Arithmetic Operations**

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 5 gives the simulation results on the transform accuracy of CS2461 core. The results are obtained by applying 64 blocks of 12-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 maximized while no overflow occurs. The 12-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.

.

**Table 5: Simulation Results of Transform Accuracy**

| | |
|---|---|
| **Transform Size** | 64-point |
| **SDC setting** | 3 |
| **Scaling Factor** | 1/(2^7) |
| **Number of complex data samples compared** | 64K |
| **Maximal output Magnitude** | 2624 |
| **Maximal Error (Re)** | 7 |
| **Maximal Error (Imag)** | 7 |
| **Average Absolute Output** | 472.134 |
| **Average Absolute Error** | 0.851654 |
| **Mean Square Error** | 1.3474 |
| **Average SNR** | 54.876 dB |

## 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 BUSY are de-asserted. It indicates the first data of the 64-point data block. The data is clocked in, on the rising edge of the CLK signal. The remaining data of the 64-point data block is loaded successively on the rising edge of the clock in natural order. When the core starts to load an 64-point data block, signals XBIP and BUSY get asserted to indicate that loading a data block is in progress. The signal XBS will be ignored if XBIP is HIGH. When the last data of the block is loaded into the core, signal XBIP returns is de-asserted and signal Busy remains asserted to indicate the transform computation is in progress. Signal XBS is still ignored in this case until Busy returns to LOW.

The CS2461 core starts the transform prior to the completion of loading the 64-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 7*64 clock cycles.

In order to minimize the size of the core, the complex multiplier has been implemented using only two normal multipliers. This means that each full complex multiplication requires two clock cycles. Therefore, each of the three computation passes requires 2*64 clock cycles. The consequences of this are further explained in Processing Time and Latency section.

Signal Done is asserted after 433 cycles when the transform result is available. Downloading of the transform result is started by asserting the input signal YEnab when Done is asserted. The signal Done returns to LOW when downloading is started and the first sample of the transform result is outputted from the core in the natural order two clock cycles later after the assertion of theYEnab signal. Output signal YAV is asserted when the data on port YRe and YIm are valid and output signal YBS is asserted if the first sample of the 64-point result is output from the core. The output data is burst out from the core in 64 clock cycles.

Downloading the result can be overlapped with the 3rd computation pass to achieve 7*64 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 is de-asserted.

Figure 5 shows the functional timing diagram for the 7*64 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.

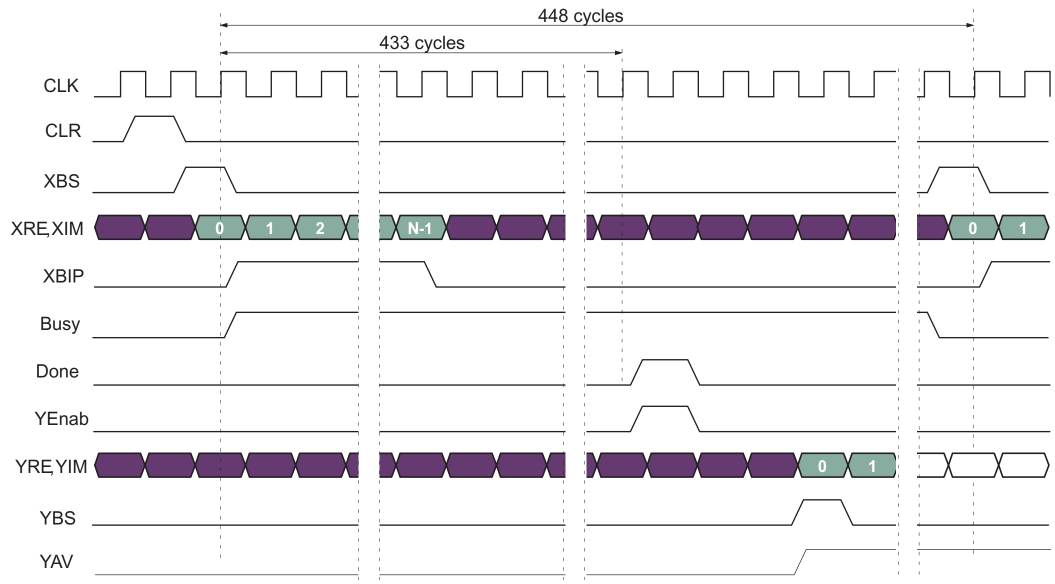


**Figure 5: 7*64 Cycle I/O and Transform Timing**

It is noted that the core waits for YEnab being asserted with the assertion of signal Done to start the downloading process, to allow the user to control the transform data flow. The system clock rate is not restricted to the 7*64 cycles and can be any rate higher than 7X the data rate. In this case if the downloading result has been completed but loading the next block is not started, signal Done will be re-asserted to indicate that the transform result in the internal memory is still available and can be downloaded again. This feature can be utilized in C-OFDM modulation systems to perform the guard interval insertion.

Figure 6 shows the operating flowchart for the CS2461 core.

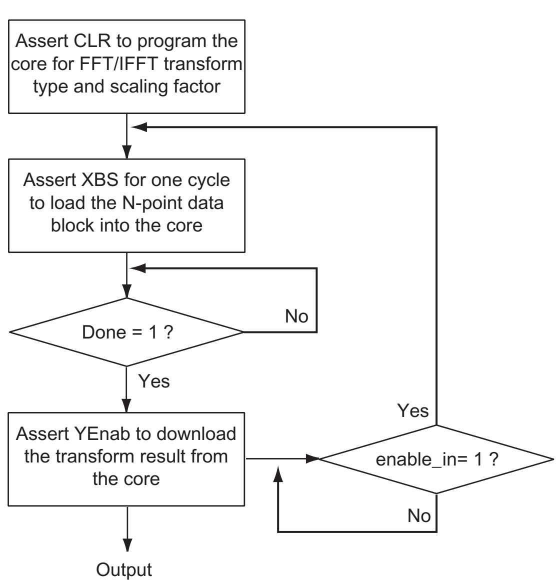


**Figure 6: CS2461 Operating Flowchart**

## OVERFLOW HANDLING

The CS2461 keeps track of 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 and 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 7*N cycle operating mode. An overflow occurring on the last few computations may not be indicated until the computation has been completed which 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 is asserted to when it is de-asserted 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 is asserted and by starting the next data block as soon as output signal Busy is de-asserted.

Table 6: CS2461 Processing Time and Transform Period

| Transform Size | Processing Time (Clock cycles) | Minimum Transform Period (Clock cycles) |
|---|---|---|
| 64-point | 384 | 448 |

The explanations for these times lie in the fact that a two cycle complex multiplier is employed, thus each internal computational pass requires 2*64 clock cycles. It is noted that the I/O process still requires only 64 cycles for a 64 point data block.

## AVAILABILITY AND IMPLEMENTATION INFORMATION

Amphion offers the CS2461 core in ASIC and programmable logic versions. Consult your local Amphion representative for product specific performance information, current availability of individual products, and lead times on ASIC or different programmable logic core porting.

The implementation information provided in Table 7 has been obtained for a stand-alone design on an Altera Apex EP20K60EQC208-2 device and a QuickLogic QL7100 device.

Table 7: Programmable Logic Core

| Product ID | Silicon Vendor | Programmable Logic Product | Maximum Frequency (MHz) | Device Resources Used (Logic) | Device Resources Used (Memory) | Availability |
|---|---|---|---|---|---|---|
| CS2461AA | Altera Apex* | EP20K60EQC208-2 | 59 | 2287 LEs | 4 ESBs | Now |
| CS2461QL | Quicklogic* | QL7100 | 58 | 10 ECUs | 4 BRAMS | Now |

* The implemenation information on ASIC or other programmable devices is available upon request.

## ABOUT AMPHION

Amphion (formerly Integrated Silicon Systems) is the leading supplier of speech coding, video/ image processing and channel coding application specific silicon cores for system-on-a-chip (SoC) solutions in the broadband, wireless, and mulitmedia 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

**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

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

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

**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
1 Rue Holiday
Tour Est, Suite 501
Point Claire, Quebec
Canada H9R 5N3

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 P ark II
Singapore 117628

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