ClearSpeed

Software Overview
Release Rev: 3.0

1 Overview of ClearSpeed software

The ClearSpeed Advance accelerators are provided with a package of runtime software. A software development kit (SDK) is also available for programmers who need to write code for the CSX600 processor. The overall structure of the software is shown in *Figure 1*.

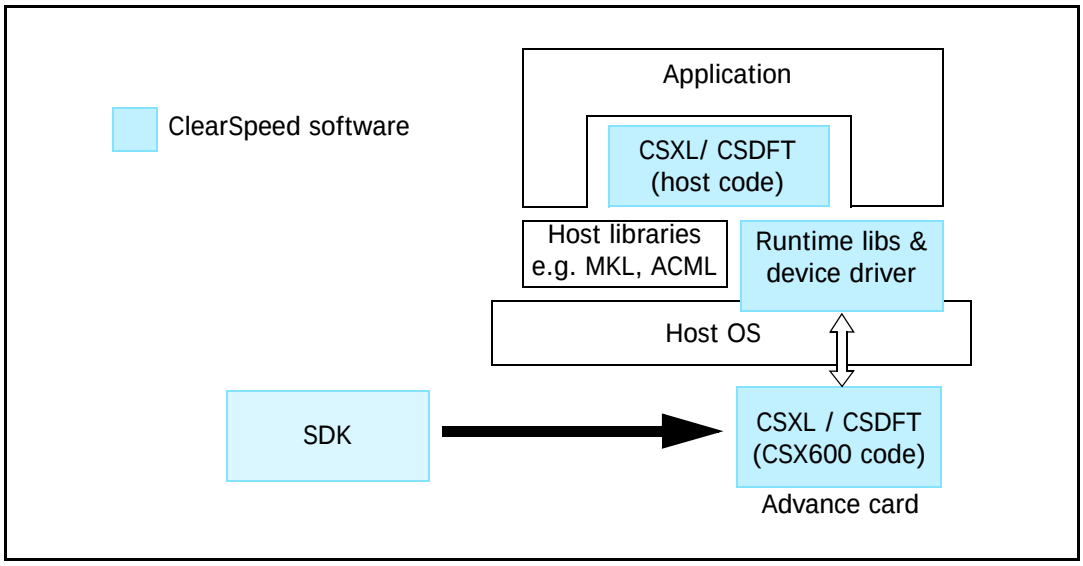

Figure 1. Software architecture

2 Runtime software

The runtime software includes:

- Device driver to interface between the host and the Advance card
- CSXL application acceleration library
- Libraries for communication between host code and software running on the card

The runtime software is delivered on CD-ROM with the Advance cards and is also available for download from the ClearSpeed customer support site:

http://support.clearspeed.com/downloads/.

2.1 Compatibility

See the ClearSpeed web site for the latest details of supported applications and operating systems.

http://www.clearspeed.com/products/applicationsupport/

2.2 CSXL library

The CSXL library implements a number of BLAS and LAPACK functions found in host math libraries such as MKL and ACML. When an application uses CSXL as the host library, appropriate function calls will be offloaded to the Advance card for acceleration. This allows applications which use functions such as `DGEMM` in these libraries to be accelerated transparently. CSXL uses heuristics based on problem size, for example, to determine how to best share processing between the host CPU and the Advance card.

2.2.1 Functions supported

The following Level 3 BLAS and LAPACK functions are directly accelerated by this version of the CSXL library:

- `DGEMM`, the BLAS level-3 routine that multiplies a real double precision matrix by a real double precision matrix.
- `ZGEMM`, the BLAS level-3 routine that multiplies a complex double precision matrix by a complex double precision matrix.
- `ZGEMM3M`, an implementation of `ZGEMM` that requires three real matrix multiplications and five real matrix additions to compute the complex matrix product; `ZGEMM` uses four real matrix multiplications and two real matrix additions. See the *CSXL User Guide* for additional information. This may be faster than the standard implementation under certain circumstance.
- `DTRSM`, the BLAS level-3 routine that solves a real double precision triangular system of equations with multiple right-hand-sides. This routine constitutes a large percentage of the computation done in the LAPACK routines that factor and solve a general system of linear equations, respectively `DGETRF` and `DGETRS`.
- `DGETRF` and `DGETRS`, the LAPACK routines that factor and solve a real double precision general system of linear equations using the LU method.
- `DPOTRF` and `DPOTRS`, the LAPACK routines that factor and solve a real double precision symmetric positive definite system of linear equations using the Cholesky method.
- `DORGQR`, the LAPACK routine that generates all or part of the orthogonal matrix Q from a QR factorization computed by `DGEQRF`.
- `DORMQR`, the LAPACK routine that multiplies a matrix by the orthogonal matrix Q from a QR factorization computed by `DGEQRF`.

2.2.2 Card side functions

In addition to the functions that can be called from an application running on the host, CSXL includes a version of DGEMM which can be called directly by code written to run on the Advance card.

2.3 Host interface library

The API available to user programs to communicate with a CSX600 device is defined in the header file `csapi.h`. For details of the CSAPI functions see the runtime documentation. The CSAPI functions are grouped into the following functional areas.

- CSAPI create and destroy functions: to manage card-state data structures
- Card control functions: provides an API for exclusive access to a CSX600 card (connect, reset)
- Program setup functions: to load and unload programs (register application, load, unload)
- Processor control functions: controlling the state of the CSX600 processors (run, halt, start, wait, get return value)
- CSX600 register access functions
- CSX600 memory access functions
- Thread functions
- Semaphore functions: synchronization with CSX600 events
- Callback functions: to allow applications to register responses to CSX600 events
- Memory allocation functions: allowing applications to manage CSX600 memory
- Utility functions: miscellaneous operations (set parameters, number of cards/processors/PEs, get error)

3 Software development kit

The SDK provides a full suite of professional software development tools built around an optimizing, ANSI C-based cross compiler. This allows developers to write and debug code for the CSX600.

3.1 Tools

The SDK contains the following tools:

- Command line tool driver providing a standard interface to the other compilation tools
- ANSI C-based optimizing cross compiler for the CSX600
- Industry-standard GDB source-level debugger
- Visual profiling tool
- Macro assembler
- Linker
- Dynamic loader
- Object file tools (archiver/librarian, dump/disassemble, etc.)
- Instruction-set simulator

The ClearSpeed C^n language is based on ANSI C with extensions to support the data-parallel architecture of the CSX600. The main addition to standard C is the addition of keywords for defining *mono* (scalar) and *poly* (parallel) data types.

The C^n compiler supports inline assembler, pragmas for controlling data layout, and vector instructions.

3.2 Libraries

The SDK includes the following standard C libraries. Most libraries include support for mono and poly data. These libraries are based on the Newlib open-source library.

- Standard utility functions (`stdlib.h`)
- Character type functions and macros (`ctype.h`)
- Standard input and output (`stdio.h`)
- String and memory functions (`string.h`)
- Wide character strings (`wchar.h`)
- Locale functions (`locale.h`)
- Math functions (`math.h`)

Note that not all functions in these libraries are implemented as some do not make sense for a coprocessor or embedded processor.

In addition, a set of extra libraries are included which support architecture-specific features.

- Vector math functions (`vmathp.h`)
- Random number generators (`rngp.h`)
- Reduction functions (`reduction.h`)
- Debug printing functions (`dprint.h`)
- Memory transfer functions (`string_ext.h`, `async_string.h`)
- Inter-PE communication functions (`swazzle.h`)
- Thread support (`semaphore.h`)
- Other functions (`misc.h`)

The C^n compiler also supports a set of intrinsics for operating on vector data to allow more efficient exploitation of the floating-point pipeline.

3.3 CSDFT library

The CSDFT library implements a number of common FFT-related functions. The library can be used directly by a host application calling functions which then pass the data to be processed to the Advance card and return the results. Because many applications perform an FFT then some other processing such as a convolution and then an inverse FFT, the CSDFT library can also be integrated with user code running on the CSX600. In this way, all of the processing can be performed on the accelerator card providing greater acceleration and minimizing the amount of data transfer between the host and the Advance card.

3.3.1 Functions supported

- Functions
  - FFT
  - Inverse FFT
  - Convolution

See the *CSDFT User Guide* for details of the functions implemented.

- 1D DFT
  - Power of two sizes from 128 to 8192
  - Single and double precision
  - Complex types
  - Complex to complex (forward and backward)
  - Interleaved only
  - Natural and optimal order (only optimal order supported by CSDFT card-side API)
- 2D DFT
  - Power of two sizes from 128 to 2048
  - Single and double precision
  - Real and complex types
  - Complex to complex (forward and backward)
  - Real to complex (forward)
  - Complex to real (backward)
  - Interleaved and split array (real and imaginary in separate arrays)
  - Natural and optimal order

3.4 Licensing

The SDK is sold with a single-user floating license. This is managed by the FLEXlm license manager. To allow multiple concurrent users, the appropriate number of licenses must be purchased. Time limited evaluation licenses are also available. No license is required, and no royalties payable, on any software developed with the SDK. Note that the C^n standard libraries are licensed under the terms of the GNU Lesser General Public License (LGPL) or similar terms and any software linked with those libraries must comply with those license terms.

4 Multicard support

The driver supports multiple Advance cards in a system. It provides access control on a per-process basis. A host process calling the CSXL or CSDFT libraries can only use a single Advance Accelerator Card (each process can be running on up to 16 host cores) as shown in *Figure 2*. In a system with, say, four Advance cards, each can be used by a separate process. These processes could be part of a single application, or could be independent application programs.

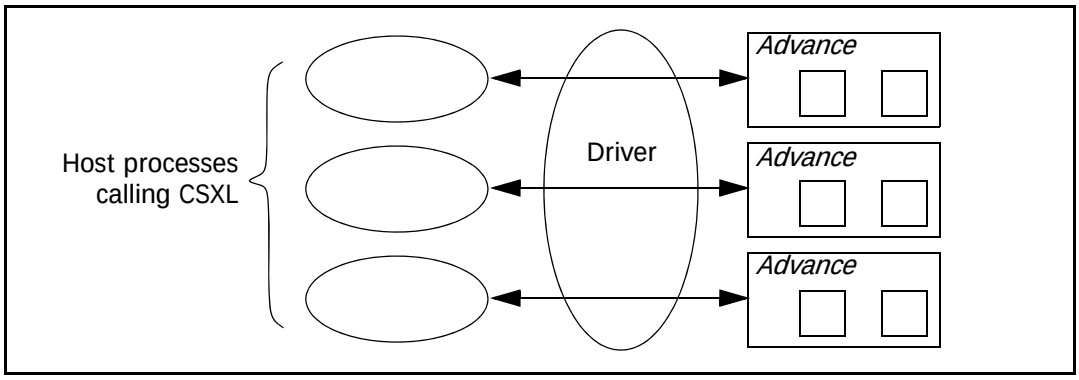

Figure 2. Using multiple Advance cards in a system

Custom host code, which does not use ClearSpeed's application acceleration libraries, can access any number of cards from a single process.

5 Availability and installation

The software is available in two downloadable packages: Base and Developer. The contents of the packages are as follows:

Base package	**Developer package**
● Runtime and drivers ● CSXL library ● Release notes ● Installation instructions ● Other documentation	● SDK ● Simulator ● Visual profiler ● CSDFT library ● FLEXlm license server ● Release notes ● Installation instructions ● Other documentation

Table 1

The packages are available as tarballs for Linux or as zip files for Microsoft Windows. These can be downloaded from the ClearSpeed customer support web site: http://support.clearspeed.com/downloads/.

Only developers who have purchased the SDK can download the developer package: you will be provided with a login account on the ClearSpeed support site in order to download the SDK.

Once you have downloaded the installation packages, you should use `tar` on Linux, or a suitable unzip program on Windows, to unpack the contents. The installation instructions will be found in the top level directory and other documents in the `docs` directory. Each of the individual components can then be installed as described in the appropriate installation documents.

6 Open source software

Some tools and libraries are open source. The main ones are:

- The kernel module driver for Linux and Microsoft Windows
- csgdb, the port of the GDB debugger to the CSX600
- The C standard libraries, based on the Newlib source

The source code for all open source software is provided as part of the appropriate package or is available on request from ClearSpeed.

7 Revision history

Date	Revision	Changes
26-07-07	2.0	Updated for release 2.51
02-08-07	2.1	Minor corrections after external review
24-10-07	3.0	Updated for software release 3.0
30-01-08	3.A	New corporate template applied

ClearSpeed Technology, Inc.
3031 Tisch Way, Suite 200
San Jose, CA 95128
United States of America

Tel: +1 408 557 2067
Fax: +1 408 557 9054

Email: info@clearspeed.com

Web: http://www.clearspeed.com

Support: http://support.clearspeed.com

ClearSpeed Technology plc
3110 Great Western Court
Hunts Ground Road
Bristol BS34 8HP
United Kingdom

Tel: +44 (0)117 317 2000
Fax: +44 (0)117 317 2002

Acknowledgments:

Windows is a registered trademark of Microsoft Corporation in the United States and other countries

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.