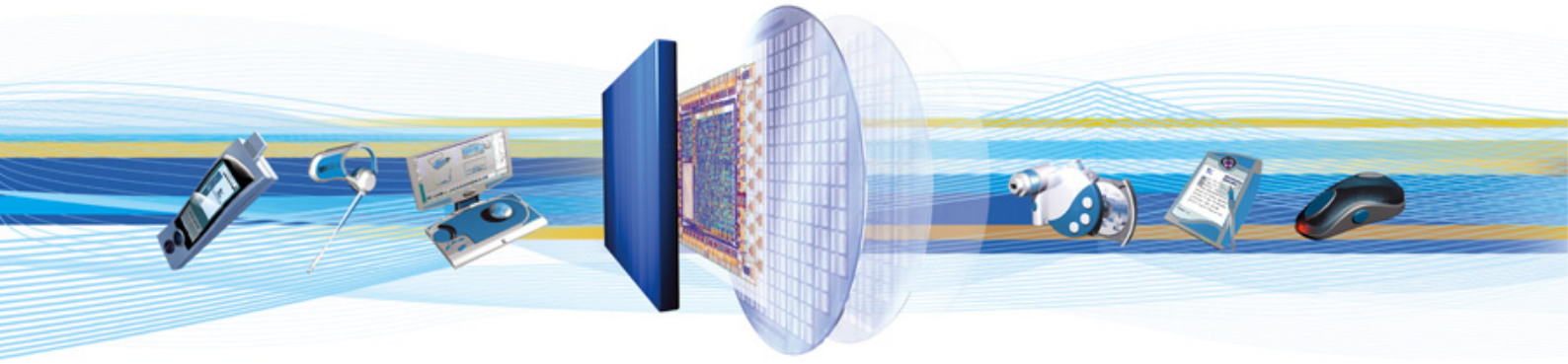
CSR Synergy Framework 3.1.2

Software Release Note

January 2012



Cambridge Silicon Radio Limited

Churchill House
Cambridge Business Park
Cowley Road
Cambridge CB4 0WZ
United Kingdom

Registered in England and Wales 3665875

Tel: +44 (0)1223 692000
Fax: +44 (0)1223 692001
www.csr.com

Contents

List of Tables

1 Introduction

CSR Synergy Framework is a software layer providing a common infrastructure for easy integration of multiple wireless technologies. The Framework provides the core services of a virtual operating system, which shields technologies from variations in specific platforms. The Framework services include scheduling mechanisms, memory management, and transport protocols for communication with the CSR wireless chipsets. Services are also included for logging and various utilities like e.g. string handling and Unicode handling.

The CSR Synergy Framework software consists of a set of components with well defined programming APIs. Some of the APIs have generic implementations provided by the Framework. These form the *Generic Support Package (GSP)* of the Framework. Other APIs are inherently platform dependent and require a platform specific implementation, also called a *port*. Such a port forms a *Board Support Package (BSP)* for the particular platform.

1.1 Certificate of Originality

All code provided in the CSR Synergy Framework is original CSR code, except for the included IP stack which is located in the: gsp\src\ip\lwip_releases folder and which is Copyrighted by the Swedish Institute of Computer Science.

2 Release Highlights

This section contains a short description of new and changed functions introduced in the Synergy Framework 3.1 version. Please note that the 3.1 version introduces changes to the generic as well as the ported api's compared to the 2.x series; more information on the specific changes from 2.x to 3.x can be found in api change document, [api_change].

The main highlights for the CSR Synergy Framework 3.1.0 are as follows.

- The IP socket API has been extended with support for IPv6, and implementations are provided for pclin and pcwin. Along with this several other improvements are added in IP related blocks, e.g. enable the DHCP Server to unicast to unconfigured clients (i.e. clients that do not yet have an IP address).
- The DSP Manager (DSPM) API is added to support newer CSR BlueCore™ chips. The DSPM API allows an application to control how audio data sources and sinks on the BlueCore™ are connected to each other, and additionally allows processing of these audio data streams.
- A CSR Trivial File Transfer Protocol (TFTP) function block is added and described in an API. The CSR TFTP is implemented as defined in RFC-1350 [RFC1350], but only the server side.
- The CSR Synergy Transport Manager (TM) API is extended with run time power up and down. This is to be used together with run time power up and down in the various CSR Synergy technologies, e.g. Bluetooth.
- Several updates, optimisations and bugfixes have been added. This includes performance optimisations in the generic scheduler which improves message passing performance.
- A new api to read current memory consumption allocated through the CSR Pmem APIs is added.
- Log function overhaul and optimisations.
- Add instrumented profiling service (IPS) support to generic scheduler.

For the full list of issues addressed in this release, see Appendix C.

3 Testing

The CSR Synergy Framework has been compiled against the latest CSR BT, Wi-fi and FM releases:

- **BT:**
  - 18.0.13
  - 18.1.8
  - 18.2.1
- **Wi-fi:**
  - 4.2.2
  - 5.0.3
- **FM:**
  - 2.0

The DSP Manager API has been tested through the BT HFG profile and the Wide Band Speech feature. This test is performed against an Elvis chip with special Gemini FW.

The CSR Synergy Framework has been tested on an ARM11 (666MHz) development board, a Linux PC and on a Windows XP PC platform.

The testing includes:

- Different host transport tests
- Automated regression testing
- Static code analysis tools (Lint)
- Runtime analysis tools (Purify used on Windows XP)

Some tests generate performance measurements. These are reproduced in the following table for reference. All SDIO measurements are at 50MHz bus frequency, and all CSPI measurements are at 25MHz bus frequency.

Test	Result
SDIO CMD52, Asynchronous	6µs / operation
SDIO CMD53, 2KiB, Asynchronous	93µs / operation
SDIO CMD52, Synchronous	7µs / operation
SDIO CMD53, 2KiB, Synchronous	99µs / operation
SDIO Interrupt Latency	3µs
CSPI Register, Asynchronous	13µs / operation
CSPI Burst, 2KiB, Asynchronous	669µs / operation
CSPI Register, Synchronous	12µs / operation

CSPI Burst, 2KiB, Synchronous	675µs / operation
CSPI Interrupt Latency	7µs

Appendix A Known limitations

None.

Appendix B Known Issues

- Due to physical routing of lines on the 1926 board on the BDB2/3 and simultaneous set up of both CSPI and UART result in corruption of the first port used. I.e. if UART is established and the CSPI port is enabled, the UART communication is broken. This is tracked internally as D-20722.
- When running the BT throughput test group over USB, package loss are occasional seen. This is observed in combination with the Blue-suite driver version 2.4.0. Tracked through D-25744.
-

Appendix C Issues Addressed

This section describes issues that have been addressed in this release relative to previous 3.1.1. release.

ID	Description
D-23078	HCI: The receive data path has been optimized.
D-23999	pclin: An error has been fixed in which the IP ifconfig network interface state transitions were sometimes inconsistent.
D-24409	Ethernet manager: A potential NULL pointer dereference has been fixed.
D-25278	pclin: IP ifconfig has been changed to delegate all work to the IP worker thread to avoid blocking the scheduler for extended periods of time.
D-26061	pclin: A bug has been fixed in IP ifconfig where static configuration of a network interface would ignore any default gateway configuration.
D-26297	Note on originality of code added
D-26409	BDB2/3: The maximum command line argument length was constrained to 256 characters because of an undocumented limitation in the RealView ARM Standard C libraries. This limitation has been removed by implementing a custom command line chopper. The limitation is now 4096 characters (including zero termination) or 128 arguments (including final NULL argument).
D-26456	pclin: A potential deadlock has been fixed.
D-26467	Linux GCC 4.6.1 issue with "-DCMAKE_BUILD_TYPE=Release" corrected
D-26472	The declarations in the csr_hci_socket.h header file has been moved to more appropriate headers in order to better align with existing transport interfaces.
D-26607	BDB3: A buffer indexing error in the UART driver for this platform that would cause memory corruption during heavy transmit operation has been fixed.
D-26625	DSPM was unable to compile if CSR_BLUECORE_ONOFF was defined. This has been fixed.
D-26691	Interface description for CSR_FP_READ added to the Fastpipe api description, api-0010-fastpipe.pdf.
D-26693	TCP Transceiver: This example application has been improved so that it is possible to enable/disable Nagle from the UI and the server will now fall back to listening when the client disconnects instead of also closing the listening socket.
D-26724	lwIP: A memory corruption issue when closing listening TCP sockets has been fixed.
D-26878	Log: CsrLogBci prototype was incorrect. It has now been fixed.

D-27211	pclin: A synchronization issue has been fixed which could otherwise lead to deadlocks and crashes.
D-27219	pclin: An issue has been fixed in which an ether interface could read from an incorrect file descriptor and try to send whatever data as an Ethernet packet.
D-27426	When building for host on a 64bit Linux system, the default target is now pclin-2.6-x86_64 instead of pclin-2.6-x86.
D-27526	lwIP: Window scaling has been disabled by default due to bugs.

This section describes issues that have been addressed in this release relative to previous 3.1.0. release.

ID	Description
D-17305	Added new HCI Socket transport for use on Linux. The transport tunnels HCI data via a raw socket to the underlying Bluetooth HCI bus driver (in kernel). Currently the implementation supports the Bluetooth HCI USB driver.
D-21236	IP socket: Binding to TCP/UDP port zero now correctly reports the selected port in the confirmation signal.
D-23495	pclin: It is now possible to configure TLS and IP to share their worker thread.
D-24336	BDB3: A hardware glitch in the SPI controller of the Samsung S3C6410 SoC, that would cause corrupt data transfers in rare cases, has been worked around by resetting the controller hardware FIFO every time the bus clock frequency is changed.
D-24555	Add application UI lower porting example on linux as exists on other platforms.
D-24606	The DSPM task now uses the sequence numbers of the BCCMD commands to make it more debug friendly. This has no impact on the functionality.
D-24638	BlueCore: The data transmission has been changed to avoid redundant memory allocation.
D-24721	Include DSPM primitives in the wireshark log.
D-24728	lwIP: Support for TCP window scaling has been added to allow for high throughput on high-latency connections.
D-24743	Change the default thread priority of the scheduler.
D-24760	lwIP: A significant performance increase has been made for TCP transmission.
D-24771	DHCP server: A use-after-free error has been removed that could otherwise lead to undefined behavior or crashes.
D-24773	BlueCore: The H4DS receive path has been simplified for SCO data, and the general H4DS receive path has been optimised to avoid allocation. A consequence of this is a reduction of code size of the BlueCore layer.
D-25017	Optimise the sync-part of CsrSdioCspiReadAsync() function.
D-25019	pcwin: The code to handle opening of USB devices has been more robust.
D-25111	pcwin: An issue where lack of USB buffer space would lead to HCI/ACL/SCO packet loss has been fixed.
D-25128	lwIP: Incorrect return code when trying to create RAW IPv6 socket has been fixed.
D-25213	Fixed non-ANSI warning in log module
D-25228	BCSP: The default jitter value has been corrected to 10ms from 30ms.
D-25284	TLS: A race condition has been removed which could otherwise lead to undefined behavior.
D-25312	pclin: Possible memory leak during IP task deinit has been plugged.
D-25313	pclin: Possible memory leak of internal primitives during deinit has been plugged.
D-25332	pclin: A possible out-of-bounds memory access has been fixed in the IP task.
D-25367	pclin: Ether primitives were leaked during deinit.
D-25412	pclin: An error has been fixed that could lead to deadlocking and leaking of memory.

D-25415	pclin: A misleading debug statement has been corrected.
D-25416	TLS: An issue has been fixed in which connection establishment might fail when using multiple concurrent connections.
D-25417	pclin: General optimisation in IP and TLS processing leading to shorter data paths.
D-25619	Removed redundant code in scheduler main loop.
D-25620	Excluded code from scheduler main loop when compiling with shutdown support disabled.
D-25626	BDB3: The native platform UART driver has been optimised, decreasing the processing time spent accessing the hardware FIFO by ~55%.
D-25652	BDB2/3: A miscalculation in the native platform UART driver in UARTReceiveDataPointerGet function, that would cause out-of-bounds reading of the buffer in some cases, has been corrected.
D-25743	BDB: CsrEventWait() would previously return prematurely when short wait durations were requested.
D-25771	BDB: Zero event bits when waiting for events times out.

This section describes issues that have been addressed in this release relative to previous 3.0.x. release.

ID	Description
D-6821	GSP: A library has been added that can be used to perform memory usage over time.
D-10888	USB: The driver now detects write failures.
D-12418	[TFTP] A TFTP server has been added for use by the application. Furthermore a service TFTP_FILE has been added on top of TFTP for mapping all request to the file system using FSAL.
D-12919	BSP: Memory allocation APIs have been extended with optional interfaces that can be used for memory debugging.
D-13093	Log: A new interface for configuring the filtering of log text based on names has been added. See CsrLogLevelTextSet in csr_log_configure.h.
D-15668	IP: Updated the code base for the generic IP stack to the latest lwIP release (1.3.2). The internal pool allocator of lwIP has been disabled and is now mapped directly to CsrPmemAlloc.
D-15711	Doc: User guide has been amended with details about the build process.
D-16054	Log: The default cleartext formatter output has been adjusted to fix a few issues and make the output easier to parse.
D-17290	Added API document for the Bluecore Transport Manager.
D-17293	Added a template for the porting of the CSR IP Ether, Socket and Ifconfig interfaces. See bsp/templates directory.
D-17905	SDIO: Preliminary support for SDIO on Windows has been implemented.
D-18013	Example Application: A new Bootstrap implementation has been integrated which supports more recent BlueCore devices.
D-18381	lwIP: Support for port NAT of UDP and TCP connections has been added.
D-18512	DSPM: A new task has been added to provide support for the new DSP Manager interface exposed by certain BlueCore devices, enabling configuration of audio routing and control of advanced DSP functionality.
D-18765	The example application key input handling has been improved to support all keys in a consistent way across all supported platforms. All platforms now support direct input of ASCII characters from a connected keyboard, avoiding the tedious and time consuming task of using the multi tap input method.

D-18933	pclin: A number of fixes concerning the use of the pthreads library have been made that in rare circumstances could cause crashes and erratic behavior.
D-19392	Removed the internal and unused CSR_EXT_PRIM define from csr_prim_defs.h
D-19465	HCI: Several calls to CsrPanic() have been removed because they could not be reached.
D-19515	Log: Fixed an indexing issue in the cleartext formatter.
D-19611	Example application: Dead code has been removed from the example bootstrap implementation.
D-19863	Rename conflicting parameter name (CsrSchedTid renamed to tid).
D-19884	A 'getting started'.html page is included for easy reference to all documents.
D-19938	Add instrumented profiling service (IPS) support to generic scheduler
D-19946	lwIP: A number of issues have been fixed that would cause instability and erratic behavior.
D-20023	AM: Reduced the code size by removing unused sections of code in the Audio Manager task (AM).
D-20076	HQ: The HQ interface has been extended to allow the application to (optionally) provide a response to an HQ command, instead of automatically responding to any incoming commands.
D-20210	Naming in api change document correct for CSR_IP_ETHER_IF_FLOW_CONTROL_PAUSE_REQ and
D-20304	BCSP: It is now possible to let BCSP pick the retransmission timer period at runtime based on the current baud rate.
D-20327	IP socket: The API has been extended with support for IPv6, and implementations are provided for pclin and pcwin.
D-20335	AM: Fixed a number of issues in the Audio Manager deinitialisation code that would cause memory leaks under certain circumstances.
D-20336	AM: Fixed an issue that would cause the Audio Manager to not clean its own queue during deinitialisation.
D-20342	TM: Added missing declaration of CsrTmBluecoreFreeUpstreamMessageContents to csr_tm_bluecore_lib.h.
D-20512	Android: Compilation and linking fixes.
D-20607	Log: The BTSnoop log formatter could misbehave in out of memory cases.
D-20639	Log: File and line has been added to the output of CSR_LOG_TEXT_UNHANDLED_PRIMITIVE.
D-20650	Changed CsrEventSet implementation for linux to signalling the condition variable with the mutex locked.
D-20666	[DHCP_SERVER] The use of offsetof has been changed to use CsrOffsetOf instead.
D-20672	Added support in framework extensions for kernel 2.6.37 which lacks init_MUTEX.
D-20756	Added missing source files to build system when compiling with CHIP_MANAGER_ENABLE.
D-20866	pclin: It is now possible to specify a prefix for the compiler on all (user space) targets using the new variable COMPILER_PREFIX. In addition, a generic target has been added to facilitate cross compiling to various targets: pclin-2.6-generic.
D-20901	[DHCP_SERVER] The defines CSR_DHCP_SERVER_INIT, CSR_DHCP_SERVER_HANDLER, and CSR_DHCP_SERVER_DEINIT have been added to the file csr_dhcp_server_task.h.
D-21004	[DHCP Server] The DHCP Server was handling the IP address of one less than the broadcast address as invalid.
D-21006	[DHCP Server] Removed memory leaks in the DHCP server.
D-21038	pclin: the example app now correctly reacts to the backspace key if it is configured as CTRL-H in the terminal. In addition, the escape sequence for the delete key has been

	mapped to CSR_UI_KEY_DEL.
D-21051	SDIO: Fix an issue in the generic CSR SDIO driver which would cause the interrupt processing to stall.
D-21072	[DHCP Server] Lease in CSR_DHCP_SERVER_LEASES_SET_REQ was not handled correct by the server.
D-21102	pclin: An IP ifconfig bug which would prevent multiple up/down mode changes of a network interface when using DHCP has been fixed.
D-21211	pclin: A use-after-free error has been corrected in the IP socket code.
D-21238	pcwin: A livelock has been fixed in the IP socket component that could prevent the task from shutting down or failing to handle socket communication.
D-21262	Log: The default value of CSR_LOG_PRIM_SIZE_UPPER_LIMIT has been increased to 64 bytes.
D-21343	Generic scheduler: Timers scheduled to run after time wraparound no longer trigger too late.
D-21376	Generic scheduler: A rounding error caused by integer division could previously cause the scheduler to spin for up to 1ms when a timer was close to triggering.
D-21474	BCSP: BCSP no longer uses the deprecated CSR_LOG_TEXT functionality.
D-21523	lwIP: Raw sockets no longer consume packets.
D-21574	BDB3: Fix an issue in the SDIO Master driver that would sometimes cause a transfer to never finish.
D-21591	BCSP: Dead code has been removed in BCSP.
D-21684	pclin: A number of stability issues have been fixed in the TLS tasks.
D-21744	pclin: A bookkeeping bug has been fixed that would prevent creating new sessions.
D-21826	BT was not able to build with the target file target-pclin-android2.6-armv5.mk. The target file now depends on an external script like agcc to set up the environment for building Android.
D-21835	pclin: The TLS component was incorrectly using CSR IP socket definitions internally.
D-21853	pclin: The IP task now supports the CSR IP ether interface which makes it possible to run network drivers in userland.
D-21882	IP: Fixed an issue in the generic IP stack that would cause memory trashing when fragmenting outgoing datagrams.
D-22000	SDIO: Fix an issue in the interrupt handling that would cause the interrupt processing to stall, if an interrupt occurs when no Function Driver has interrupts enabled.
D-22013	pclin: When compiling for Android, CSR IP ifconfig will now use dhcpcd for DHCP configuration of network interfaces instead of dhclient.
D-22026	BDB3: Fix an issue in the hardware timer control that would cause scheduler timers to trigger too early.
D-22042	pclin: The IP ifconfig interface failed to bring the interface up (IFF_UP) when doing static configuration.
D-22123	SDIO Probe: This example application now sets the bus clock frequency before starting any test, to make sure that it is running at the maximum performance.
D-22130	BDB2/3: Added support for compiling with RVCT4.1.
D-22136	SDIO: Fixed a race condition that would occur in scenarios with multiple devices on the same bus.
D-22234	A IPS Manager has been added which is an example application of how Instrumented Profiling Service can be used
D-22479	pclin: When using CSR IP ifconfig to configure interfaces using DHCP, DHCP clients are no longer lingering after the task has shut down.
D-22536	IP: Queue incoming TCP data up to a maximum of the window size in the wrapper,

	avoiding unnecessarily dropping data, which lowers the receive performance considerably.
D-22609	BDB2/3: Added fflush implementation to the csr_console component.
D-22612	Exceptionhandling: The deprecated exception handler module is no longer used.
D-22623	BDB2/3: The csr_console component now checks the return value of UARTTransmit, and retries until the character has been delivered to the UART buffer.
D-22635	Doc: Ether API updated with correct references.
D-22711	BDB2/3: Increased maximum command line argument string length from 256 to 4096 (including zero termination).
D-22726	Added dependency file generation to the BSP specification tests.
D-22761	IP: The DNS fields of the CSR_IP_IFCONFIG_SUBSCRIBE_CFM message was not filled out properly, causing uninitialised data read errors. This has been fixed.
D-22767	Added support for enabling/disabling the entire BlueCore stack and device communication at runtime. This functionality must be enabled by the option CSR_BLUECORE_ONOFF.
D-22771	lwIP: Raw sockets now provide payload from all fragments for a given packet.
D-22816	GSP: No longer copies static data.
D-22879	Linux: Removed gcc compiler warnings due to not checking return value from certain system functions.
D-22885	Add cast to avoid warnings on 64bit systems.
D-22907	pclin: DHCP client path is now configurable.
D-23077	Log: Added a new dynamically assigned handle based log text interface (csr_log_text_2.h) that improves the efficiency of the log system by avoiding excessive traversal of linked lists to look up context information, by storing the information directly in the handle.
D-23185	IP: Added support for multicast forwarding and NAT to lwIP. Multicast datagrams will be forwarded to all other interfaces. If NAT is enabled on a particular interface, the forwarded datagram will have its source ip rewritten on output on that interface, and datagrams with destination 224.0.0/24 and 239/8 will not be delivered through the NAT (will be dropped) as specified in RFC5135.
D-23222	pclin: In TLS, enable one more cypher.
D-23223	Prevent passing uninitialised pointer to CsrPmemFree
D-23242	DHCP Server: When broadcasting DHCP messages, the DHCP Server would previously use the universal all-ones IP address as destination, which would cause some IP stacks to discard the datagram or pass it to the default route (which may not be the subnet the DHCP Server is operating on). This has been changed to use the network broadcast address of the particular subnet the DHCP Server is operating on, to ensure that the DHCP messages are delivered on the correct subnet.
D-23267	lwIP: NAT was previously done only on the IP output path and thus was not done on the IP forwarding path.
D-23308	Corrected misplaced endif in csr_tm_bluecore_lib.h to prevent compilation fail on C++ compilers if the header is included more than once.
D-23391	Ethernet manager: Illegal memory accesses fixed.
D-23440	Handler, init and deinit defines have been added to all task header files, and the example application is now able to build with ENABLE_SHUTDOWN=0.
D-23471	ifconfig: An interface for adding ARP entries has been added.
D-23582	TFTP: A memory leak has been plugged.
D-23630	pclin: -N and -Y are no longer passed as arguments when using dhcpcd as the DHCP client.

D-23677	pclin: A file descriptor was sometimes incorrectly closed which could cause the IP ifconfig interface to malfunction.
D-23711	pclin: When an interface configured using DHCP loses link, the DHCP client was previously not terminated.
D-23984	gsched: The message and timer paths have been optimised.
D-23992	gsched: Unused internal type removed.
D-24023	gsched: A memory leak in the deinit path has been plugged.
D-24065	pcwin: Memory leaks have been plugged in the IP task.
D-24070	Fixed some issues with the pcwin-nt-x86_64 target scripts that would cause some of the compiler options to not be applied. Also added the /D_CRT_SECURE_NO_DEPRECATE and /D_CRT_NONSTDC_NO_DEPRECATE options. Removed all compiler warnings generated when building for this target.
D-24089	pclin: Some issues, static overflow and use of an insecure API, found by Coverity have been fixed.
D-24096	pclin: Unsafe use of execve(2) has been rewritten.
D-24104	pclin: Dead code has been removed.
D-24120	pclin: A number of potential use-after-free issues have been fixed.
D-24127	pclin: csr_file now compiles cleanly for Android.
D-24159	gsched: The background interrupt handling path has been improved slightly.
D-24191	UI: Fixed a crash in multi tap input handling when the graphics renderer is disabled, and added proper redraw for text renderer in same situation.
D-24241	pclin: Thread removed from userspace UART driver.
D-24287	DHCP server: Memory leaks at deinit time plugged.
D-24394	All components that previously used csr_log_text.h has now been modified to use the new and more efficient interface in csr_log_text_2.h. Compatibility with existing formatters and filtering functions is retained.
D-24442	The example application has a new command line option that utilises the new name based log text filtering functionality to make it easy to setup the log text filtering.
D-24461	Log: The log component now uses a linked list instead of a fixed size list to store registered formatters to improve the efficiency of the log system. The argument passed to CsrLogInit is now ignored, as the linked list can grow dynamically as formatters are registered.
D-24468	Log: The formatter interface (csr_log_formats.h) has been extended with three new callbacks: ltextregister2, ltextprint2, ltextbufprint2. These are equivalent to the existing functions by the same name without the trailing 2. If these callbacks are non-NULL, events pertaining to components that use the new csr_log_text_2.h interface will be delivered to the formatter (only) through these new callbacks, whereas events pertaining to components that use the previous interface will unconditionally be delivered through the existing functions. If these new callbacks are NULL, all events (regardless of interface used by the component) will be delivered through the existing callbacks (for backward compatibility). The cleartext formatter has been updated to take advantage of this to avoid storing and retrieving information about components that use the new csr_log_text_2.h interface.

This section describes issues that have been addressed in release 3.0.0.

ID	Description
D-3377	BCSP: Prevent LE timers from triggering when unnecessary.
D-4405	Added support for compiling with Microsoft Visual C++ 2010. Use COMPILER=VC10 to enable.
D-6300	Log: an ouf of bound read would occur when piping log to the FTS Virtual Sniffer. This has now been fixed.
D-6926	BCSP: The retransmission timer can now be jittered to avoid retransmission while the chip is waking up which could the retransmission to be lost.
D-7989	Enabled bind check for UDP
D-9741	Removed assertions on rtc time results
B-91891	pclin: IP ifconfig now reports link up for wireless interfaces in host AP (master) mode.
D-10500	BCSP: Cancel timer on ACK reception.
D-10643	Log: A document describing how to set up the CSR Synergy logging services has been added.
D-11635	The default thread priority for the BDB2/3 schedulers have been changed to match the equivalent default priority used in the framework extensions.
D-11705	IP: Zero-initialization of certain header fields has been added to silence false positives from Purify.
D-11783	IP ether: An unused type has been removed.
D-11815	Added mechanism to retrieve latest panic and fault arguments from chip when booting. Applies when chip manager is enabled.
D-12151	A new cryptographic strength generic random number generator has been implemented which replaces CsrRandom from the BSP header csr_util.h.
D-12202	Added chip manager component to monitor bluecore chip status. When enabled the chip manager will detect and act on unexpected chip resets. Having detected an unexpected reset, the chip manager will rerun the bootstrap and inform subscribers of the event.
D-12305	Deprecated: A CsrSprintf() prototype was missing from csr_deprecated.h.
D-12471	Buildsystem default target now depends on host OS
D-12769	GSP: Compatibility code for Synergy Framework 1.x has been removed.
D-12894	csr_msgconv.h: All serialisation and deserialisation code now uses CsrSize instead of CsrUint16 to represent the sizes, to remove the limitation on the size of the serialized primitives.
D-13140	The CsrLogRegisterPlatform and CsrLogPlatformInformation functions and the corresponding lregplatform formatter callback has been removed and these settings are now automatically detected by the PCAP formatter.
D-13190	Changed type of configMethod to reflect the default of the framework. Both still being Uint8 at the low level.
D-13208	MBLK reference counting fixed such that CsrMblkDestroy() decreases the reference count. Duplicate MBLKs must have their reference counters increased and the CsrMblkIncRefcount() function behave like the duplicate functions and bump the parent(s) counters.
D-13244	Added a depmap tool making it possible to easily reveal dependencies between components/libraries. See global_rules.mk for instructions.
D-13304	Changed convertAttribToDirMode to distinguish between directories and files.
D-13364	csr_msgconv.h: The serialised format of CsrUint16 and CsrUint32 has changed from big endian to little endian, and several unused serialisation/deserialisation functions have been removed.
D-13424	Type A: Unused debugging macros have been removed.
D-13440	pcwin: the implementation of csr_time has been modified to use GetSystemTime instead of the deprecated ftime, to increase compatibility with WinCE.
D-13448	pcwin: Removed unused private header file.

D-13458	Added common define for number of threads created by test cases
D-13522	Wireshark: The relationship between activate <-> deactivate task is now properly tracked.
D-13596	The source files csr_eh_panic.c and csr_eh_warn.c have been merged to one csr_exceptionhandler.c. And the check for the EXCEPTION_PANIC conditions has been moved into the source file. This change is for the pcwin and pclin ports - they now match the structure of the BDB2/3 port.
D-13758	Added target scripts to build the pclin port for Android.
D-13760	Added optional floating point types CsrFloat and CsrDouble to csr_types.h. Added hardware floating point support to BDB3 example port.
D-13839	csr_ip_ether.h: Make the context typing/naming more explicit to reduce the confusion about which context is being handled.
D-13871	csr_sdio.h: Added priv field to the CsrSdioFunctionDriver structure for use by the implementation of the SDIO driver.
D-13873	csr_ip_ifconfig.h: The CSR_IP_IFCONFIG_DHCP_RENEW_REQ and CSR_IP_IFCONFIG_DHCP_RENEW_CFM primitives have been removed as they are unused.
D-13880	CSR_PARAM_UNUSED has been removed, as it serves the same purpose as CSR_UNUSED.
D-13927	csr_ip_ether.h: Added individual priority flow control capability to the API.
D-13996	Fixed a problem with CsrUtf8StrNCpy and CsrUtf8StrNCpyZero causing it to truncate the string in a way that would leave invalid UTF-8 characters.
D-14004	The drawing routines of the application framework port for Windows has been optimised to eliminate the input lag experienced on some PC's.
D-14046	Bccmd: A new interface has been added for building bccmds.
D-14050	csr_sdio.h: Added device field to CsrSdioFunction to allow the Function Drivers to detect whether any given set of Device Functions reside on the same device.
D-14055	Removed all HOUSE_CLEANING primitives from public interfaces.
D-14122	Log: several redundant lookups have been eliminated during logging.
D-14212	CSPI: A superfluous check has been removed during initialisation.
D-14229	SDIO: The implementation of CsrSdioHardReset now conforms to the API and no longer returns CSR_SDIO_RESULT_INVALID_VALUE.
D-14242	SDIO/SSPI: The blocksize is no longer set explictly during initialisation. This should be done by the function driver.
D-14258	GSP: Documentation for the csr_formatted_io API has been added.
D-14288	Type-A: Now sets SDIO block size during initialization.
D-14298	Some panic calls in the Type-A driver has been changed into messages send to the transport manager. This affects panics that originates from being unable to communicate with the chip. The transport manager will handle the messages as an unexpected loss of chip communication. If the chip manager component is enabled it will try to recover. If the chip manager is not enabled the transport manager will panic.
D-14343	pclin: Attempt to detect wireless interfaces in IP ifconfig.
D-14350	BSP: The native schedulers for the pclin and pcwin BSPs have been removed since they were single-instance schedulers and gsched can be used instead.
D-14374	pclin: The peer_com library has been removed since it depended on the native pclin scheduler and was broken and no longer used.
D-14376	pcwin, pclin, gsched: Align memory to an 8 byte boundary when compiling with memory profiling enabled.
D-14401	Fix a large number of code style conformance/consistency issues.
D-14424	Safely handle of socket return value. (Removed warnings)
D-14432	APP/UI: CsrLogDeinit is now called from CsrAppMain on shutdown, to make sure the log subsystem is deinitialised.
D-14493	IP: Clarified the flow control requirements when using the IP Socket interface in the documentation.
D-14525	pclin: IP ifconfig correctly terminates netlink parsing.

D-14556	Log: HCI packets are no longer embedded in Synergy layers.
D-14577	Update API documentation with missing REQ for CSR_IP_SOCKET_TCP_NEW
D-14584	IP: Clarified the use of the CrIpIfConfigUpReq and CsrIpIfConfigDownReq in the documentation, and improved the code to enforce proper use, and report wrong use more clearly.
D-14608	pcwin: A race condition has been fixed in the IP socket interface where an application could sometimes request a data transfer too soon after connecting, leading to a deadlock.
D-14622	FW: Redundant use of `extern' has been removed
D-14640	The following unused log level flags have been removed: CSR_LOG_LEVEL_TASK_MEMORY_ALLOC, CSR_LOG_LEVEL_TASK_MEMORY_ALLOC_LOC, CSR_LOG_LEVEL_TASK_MEMORY_FREE, CSR_LOG_LEVEL_TASK_MEMORY_FREE_LOC, CSR_LOG_LEVEL_ENVIRONMENT_LIST and CSR_LOG_LEVEL_ENVIRONMENT_LIST_SEARCH.
D-14678	BDB2/3: The example application framework will now automatically detect how much memory is available for the heap, and will fail during initialisation if this size is smaller than a set lower limit.
D-14685	Implemented a Black Screen of Death (BSOD) for BDB3 to make it easier to detect fatal execution anomalies.
D-14686	csr_ip_socket.h: Use CsrUint16 instead of CsrUint32 for the dataLength field in the CSR_IP_SOCKET_TCP_DATA_REQ primitive.
D-14751	SDIO: Debug code has been changed to use the log API properly.
D-14875	Logtransport: A superfluous <windows.h> include has been removed from csr_logtransport_init.h.
D-14916	pclin: Some uses of gettimeofday and do_gettimeofday has been replaced by clock_gettime and current_kernel_time respectively, to ensure a monotonically increasing time base.
D-14920	BDB2/3: Switched to -O2 -Otime for RVCT3.0 to work around a known and confirmed code generation error in the compiler.
D-14939	Buildsystem: The GENERATE_SERIALIZATION makefile variable has been moved to global_rules.mk to ease porting.
D-15075	BDB2/3: Fixed an issue with interrupt handling in the wired ethernet driver.
D-15234	BDB2/3: Fixed an issue with flooding the scheduler with background interrupts due to wired ethernet activity.
D-15517	Log: Several superfluous "csr_log_version.h" includes have been removed.
D-15595	Added clean_exec to target scripts to allow complete cleaning of all output files generated by the create_exec macro.
D-15599	IP socket: TCP output rewritten to improve stability and reliability.
D-15600	APP/UI: CSR_APP_BACKLOG_REQ now also results in a log text entry.
D-15612	Fixed the setting of network mask and broadcast address in the pclin port of the ip/ifconfig interface.
D-15625	FRW ext: A size_t typo has been corrected to CsrSize.
D-15633	BDB3: Improve the global interrupt masking/unmasking efficiency by taking advantage of the CPS instruction.
D-15652	Fixed wrong IP address conversion when resolving name via DNS when the name is already in the cache.
D-15656	Throughout all interfaces, the const qualifier has been added to parameters as appropriate.
D-15674	pclin: The target scripts have been extended to make building applications easier.
D-15686	HCI: Use of C native types has been replaced by correct CSR Synergy types.
D-15691	BCSP: Use of C native types has been replaced by correct CSR Synergy types.
D-15693	H4DS: Use of C native types has been replaced by correct CSR Synergy types.
D-15696	Type-A: An unused bookkeeping counter has been removed.
D-15732	IP socket: The generic implementation would in some cases fail to respond to DNS lookup requests and leak memory while handling them.
D-15742	Buildsystem: $(FW_LIB) and $(BSP_LIB) is now used consistently throughout the makefiles

D-15761	Log: It is now possible to configure the cleartext log formatter to output logging of scheduler events.
D-15762	pclin: A file descriptor leak during deinitialization of the IP task has been fixed.
D-15771	Type-A: The SDIO bus frequency is no longer adjusted during device wakeup.
D-15778	Ethernet manager: A generic task has been added that performs IP stack configuration using the CSR IP IFCONFIG interface.
D-15801	pcwin: The IP socket implementation now handles temporary errors gracefully.
D-15805	IP ifconfig: Unused data types have been removed.
D-15846	Log: Wireshark DLL's were deleted as part of 'make veryclean'. This has now been fixed.
D-15849	BDB3: Fixed a data corruption issue in the SPI driver.
D-15878	pclin: A race condition triggered by large data requests and connection terminations has been removed that could otherwise cause a panic.
D-15880	pclin: The IP socket handlers now correctly access shared data.
D-15883	pclin: The IP socket implementation would use a system call incorrectly.
D-15891	pclin: IP socket no longer sends a CsrIpSocketUdpNewCfm in response to handling an CsrIpSocketTcpNewReq.
D-15892	pclin: File descriptor leak plugged in IP socket error path.
D-15893	pclin: Correctly track IP sockets.
D-15896	pclin: The IP socket code now defers setting the socket type until necessary.
D-15909	pclin: Plug memory leak in IP socket.
D-15914	pclin: Plug file descriptor leak in IP socket.
D-15925	pcwin: Plug IP socket memory leak in connection failure path.
D-15928	Buildsystem: BSP_TOP is no longer locally defined in each makefile. Instead the globally defined BSP_ROOT is used.
D-15929	pclin: Plug file descriptor leak in IP socket.
D-15931	IP: The socket instance data would be leaked in some cases.
D-15933	BDB2/3: Fixed an issue with the interrupt handler that could cause a masked interrupt to be signalled.
D-15941	IP: The socket cleanup has now been consolidated into one function.
D-15946	pclin: The IP task deinitialization routine has been corrected to avoid leaking memory.
D-15947	pclin: A memory leak has been fixed that could occur when the peer closed a TCP connection and the local side had pending data to transmit.
D-15948	BDB3: Fixed an issue with the DMA driver which would lead to cache inconsistency.
D-15960	IP socket: The generic implementation would in some cases pass an invalid pointer to the application that could result in heap corruption.
D-15962	IP socket: A memory leak during task deinitialization has been fixed.
D-15964	pcwin: The deinitialization routine would reference undefined memory in some cases.
D-15965	Fixed csr_app_main_transport_bluecore.o not removed by clean target.
D-15966	IP: The IP task would in some cases reference memory that was no longer valid.
D-15969	pclin: Fix use of uninitialized memory in IP socket.
D-15984	IP: An error in the handling of buffering of incoming data has been fixed that would otherwise lead to undefined behavior.
D-15992	pclin: Correctly check IP socket failures using getsockopt().
D-16011	GSP: The generic scheduler library no longer contains an implementation of csr_pmem.
D-16026	The legacy application framework has been removed, as it has been replaced by a new application framework.
D-16029	TM: Replace use of char pointers with CSR data types.
D-16032	Build system: Certain targets had incorrect dependencies which have been fixed.

D-16033	UART: Use of C native types has been replaced by correct CSR Synergy types.
D-16143	Buildsystem: The multi-build configurator no longer specify an absolute path to itself when building a component.
D-16196	IP socket: Support for setting socket options has been added.
D-16232	pcwin: A more fair scheduling mechanism has been introduced in the IP socket interface to prevent a socket from livelocking other sockets by doing large writes.
D-16255	Added support for 64bit compilation in pclin and pcwin target scripts.
D-16274	pcwin: A potential livelock caused by continuous data reception has been fixed.
D-16305	The following unsupported target scripts have been removed: target-pclin-2.4-x86.mk, target-pclin-2.6-armv6.mk, target-pclin-2.6-ppc.mk, target-pcwin-ntcyg-x86.mk.
D-16309	Generic tasks: Remove use of CsrSchedTaskQueueGet().
D-16317	App: Plug memory leaks that would occur during deinit.
D-16324	The pclin kernel port implementation of CsrThreadSleep() changed to use uninterruptible sleep.
D-16378	GSP: Typecasts have been removed from the default configuration file to make it possible to perform compile-time sanity checks on configuration values using the preprocessor.
D-16406	BDB3: Unsupported compiler options have been removed.
D-16423	IP socket: Support for multicast has been added.
D-16435	APP/UI: Changed the --log-fts-file parameter to --log-btsnoop-file and changed the behaviour to generate usable files.
D-16479	Replace all uses of dos2unix in the build system with sed, to reduce the number of external dependencies.
D-16499	csr_ip_ether.h: The appHandle member has been removed from all CSR_IP_ETHER_IF_MULTICAST_*_RES primitives, and the order of the remaining members have changed to match the other primitives in this interface. In addition the CSR_IP_ETHER_IF_MULTICAST_ADDR_GET_RES and CSR_IP_ETHER_IF_MULTICAST_ADDR_GET_IND primitives have been removed as they are unnused.
D-16507	Type A: Notify upper layers on initialization error.
D-16544	Converters: EXCLUDE flags were not used consistently.
D-16628	CsrApp: Redundant code has been removed.
D-16818	FW: Memory leaks plugged.
D-16866	Log: Fix incorrect filter checks.
D-16869	platform/csr_serial_init.h: This platform dependent (and optional) interface has been replaced with a more general interface that allows the individual serial port parameters to be set and the handle is now opaque and allocated by the driver.
D-16872	Removed redundant -I in include directives which caused issues with some compilers.
D-16887	Schedulers: Correctly set task context when executing background interrupt handlers.
D-16906	Fixed bug in the CsrEventWait implementation for the Linux port. The timespec structure given to pthread_cond_timedwait was incorrect.
D-16917	pcwin and pclin: The wrong background interrupt was logged when unregistering bgints.
D-16927	gsched: Wrong bgint values were logged.
D-16929	BDB: Update compiler settings to be more strict.
D-16972	Application framework: add PSR file and string command line options and corresponding functions for retrieving the information from the application.
D-16973	HCI: Registration of SCO handles has been fixed.
D-16975	USB: Correctly namespaced global symbols.
D-16991	platform/csr_framework_ext_init.h: The CsrGlobalMutexCreate and CsrGlobalMutexDestroy functions have been removed. In addition the return value from CsrGlobalMutexLock and CsrGlobalMutexUnlock has been removed as they cannot fail.
D-16993	Fixed the scheduler tests to build also with the generic scheduler and on linux.
D-17014	BDB: On RVCT 4.0, a new method for treating warnings as errors is being used.

D-17038	Fixed an issue with the log text implementation that would cause memory corruption if multiple formatters where used.
D-17100	pcwin-ntkernel-x86 target now respects NO_TREAT_WARN_AS_ERROR=1
D-17102	Changed an incoming private primitive to the HCI task from an indication to a request. This also makes for the proper handling in autogenerated free functions.
D-17141	BDB: Set up thread pool with extra threads instead of just what the scheduler needs.
D-17142	IP ifconfig: An API for controlling NAT and IP forwarding has been added.
D-17172	pclin: Kernel framework extensions now compile under Linux 2.6.35.
D-17211	pclin: The code is now 64-bit clean.
D-17229	A DHCP Server has been added which can be used by the application.
D-17309	pclin: Correctly report link state in IP ifconfig.
D-17323	BDB2/3: The target scripts for these ports have been enhanced to allow more fine-grained control of optimisation level and heuristic as well as instructions set. The preset configurations CODESIZE_BUILD=1 and PERFORMANCE_BUILD=1 are used for reference measurements of code size and performance measurements respectively.
D-17340	csr_types.h: Added CsrPtrdiff, CsrUintptr and CsrIntptr types.
D-17378	Build: BSP_LIB no longer contains csr_sched.
D-17559	Added support for compiling on VC8/VS2005.
D-17636	The application framework now allows the individual application to append to the usage information printed when --help is supplied on the command line.
D-17640	csr_time.h: The header documentation of CsrTime, CsrTimeUtc, CsrTimeGet and CsrTimeUtcGet has been improved to clarify the properties of the two different timers.
D-17656	BDB: The ethernet driver has had support for controlling multicast groups added.
D-17657	BDB: The ethernet control task now handles CsrIpMulticast indications.
D-17667	gsched: The deinitialisation sequence has been made more reliable. Sometimes scheduler threads would continue to run after scheduler shutdown, and sometimes the deinitialisation handlers for tasks would not be called.
D-17684	The UI thread of the application framework is now run in it's own scheduler thread by default to avoid the redrawing of the screen affecting the execution of other tasks.
D-17813	Framework extension document updated. Non-existing interfaces removed.
D-17830	A bug in the implementation of CsrUtf16StringDes causing the deserialisation to emit the entire string on the same byte has been fixed.
D-17832	In Windows it was possible that the same data was read multiple times from the chip. In both Linux and Windows the calculation of how many bytes to read was incorrect.
D-17888	pclin: A crash in IP ifconfig has been fixed.
D-17910	A new example application task using the csr_ip_socket.h API that allows transmission and reception of TCP data has been added.
D-17941	pclin: A crash due to a double free in IP ifconfig has been fixed.
D-18073	csr_sdio: Use csr_sdio as library name for all variants of csr_sdio. Use CSR_SDIO_USE_SDIO/CSR_SDIO_USE_CSPI/CSR_SDIO_USE_SSPI in the config file to choose the variant. The BSP_SUPPORT_TRANSPORT_* and similar defines in the the target scripts have been removed in the process, as they are no longer needed.
D-18082	csr_framework_ext.h: The header description of the CsrGlobalMutex functions has been improved to clarify the purpose of this functionality.
D-18086	CSR TLS now depends on sys/socket.h
D-18172	The CSR_LOGTRANSPORT_HAVE_* variables have been removed from the target scripts as they are unused (note that the compiler flags are set in the header files).
D-18226	The CsrThreadCreate implementation in the pcwin bsp now sets thread priority.
D-18241	pclin: A deadlock in the IP ifconfig implementation has been fixed.
D-18282	Rules for parsing Chip Manager build configuration has been removed from global to local rules.
D-18293	The MBLK_DEBUG compiler flag is no longer set based on the DEBUG make variable. The

	MBLK_DEBUG make variable has been added to the configuration file to control this compiler flag.
D-18302	The tools for auto generating free functions now has ability to call handcoded free functions.
D-18321	pclin: The code has been made 64-bit clean.
D-18341	Ethernet manager: An error has been fixed which could previously cause writes out of bounds or crashes.
D-18343	pclin: The IP ifconfig subscriber list is no longer leaked during deinit.
D-18357	SDIO: The callback inhibition control API has been removed.
D-18380	lwIP: Only change default interface when we don't have one, and only select one that has a default route on it.
D-18431	BDB2/3: The UART drivers of these example ports are now able to perform the UART routing required on these platforms. The routing can optionally be appended to the device name (example: "uart1:debug", selects uart1 as device and sets up the routing to the debug uart port).
D-18517	pclin: Corrected message handler to use the proper message type for IP socket TCP abort requests.
D-18548	pclin: Correctly reset request handler events in IP socket.
D-18552	The sleepTimeInMs parameter of CsrThreadSleep has changed from CsrUint32 to CsrUint16 to match the timeoutInMs parameter of CsrEventWait.
D-18646	IP socket: The CSR_IP_SOCKET_RESULT_INVALID_HANDLE return value defined was not meaningful for this API and has been removed.
D-18664	csr_log_text.h: CSR_LOG_TEXT_CONDITIONAL_* macros have been added to allow more convenient conditional logging.
D-18673	CsrApp: A memory leak has been fixed in the deinitialization handler.
D-18677	csr_serial_com.h: The header documentation has been improved to more clearly describe the individual functions of the interface.
D-18691	csr_serial_com.h/csr_usb_com.h: The CsrUartDrvReset and CsrUsbDrvReset functions have been removed from this interface as they are not needed.

Table 1: Issues Addressed

4 Document References

Ref	Title
[api_change]	API Changes between 2.x and 3.x. Document number: GU-0003-API_CHANGE

Terms and Definitions

BlueCore®	Group term for CSR's range of Bluetooth wireless technology chips
Bluetooth®	Set of technologies providing audio and data transfer over short-range radio connections
CSR	Cambridge Silicon Radio
UniFi™	Group term for CSR's range of chips designed to meet IEEE 802.11 standards
API	Application Programming Interface

Document History

Revision	Date	History
1	19 DEC 08	Ready for first Engineering Release
2	16 JAN 09	Ready for second Engineering Release
3	09 FEB 09	Ready for release 1.0.0
4	26 MAY 09	Ready for release 1.1.0
5	30 NOV 09	Ready for release 2.0.0
6	20 APR 10	Ready for release 2.1.0
7	DEC 10	Ready for release 3.0.0
8	Aug 11	Ready for release 3.1.0

TradeMarks, Patents and Licences

Unless otherwise stated, words and logos marked with ™ or ® are trademarks registered or owned by CSR plc or its affiliates. Bluetooth® and the Bluetooth logos are trademarks owned by Bluetooth SIG, Inc. and licensed to CSR. Other products, services and names used in this document may have been trademarked by their respective owners.

The publication of this information does not imply that any licence is granted under any patent or other rights owned by CSR plc.

CSR reserves the right to make technical changes to its products as part of its development programme.

While every care has been taken to ensure the accuracy of the contents of this document, CSR cannot accept responsibility for any errors.

No statements or representations in this document are to be construed as advertising, marketing, or offering for sale in the United States imported covered products subject to the Cease and Desist Order issued by the U.S. International Trade Commission in its Investigation No. 337-TA-602. Such products include SiRFstarIII™ chips that operate with SiRF software that supports SiRFInstantFix™, and/or SiRFLoc® servers, or contains SyncFreeNav functionality.

Life Support Policy and Use in Safety-critical Compliance

CSR's products are not authorised for use in life-support or safety-critical applications. Use in such applications is done at the sole discretion of the customer. CSR will not warrant the use of its devices in such applications.

Performance and Conformance

Refer to www.csrsupport.com for compliance and conformance to standards information.