MISRA Compliant TCP/IP Stack

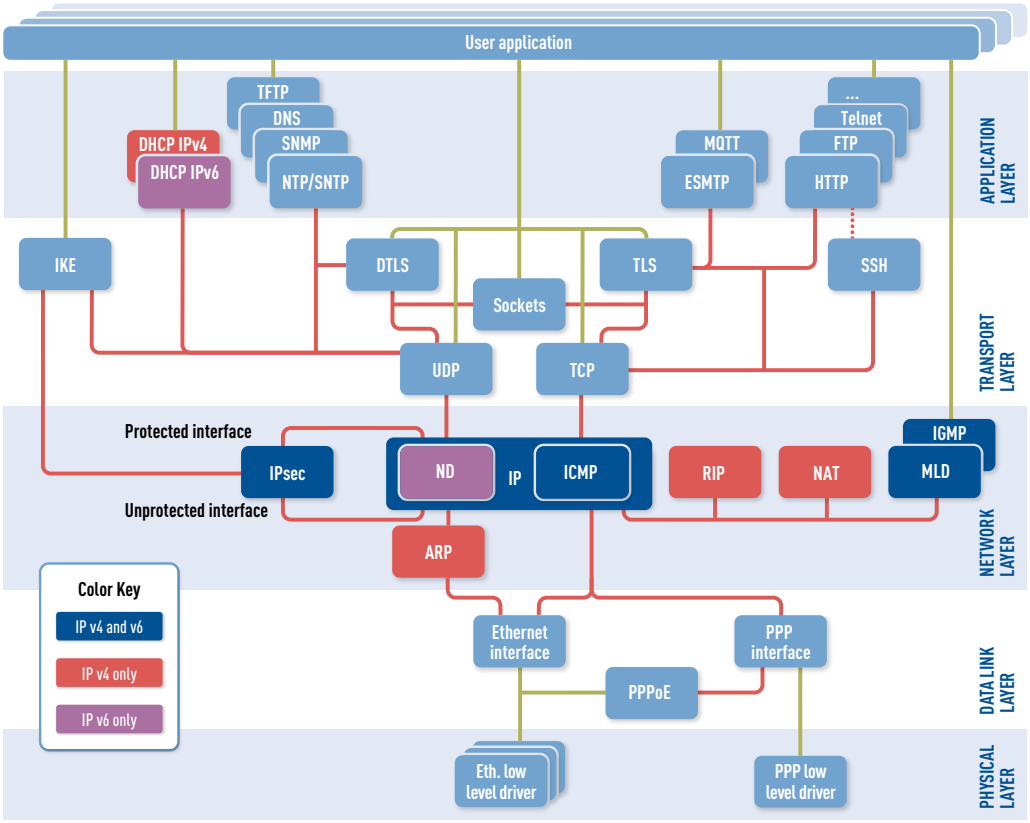
Network quality and security issues are not usually caused by problems with the requirements or security protocols, but with their implementation. Traditional freestyle 'code-then-test' methods are not sufficient to guarantee correctness and security. However there are formal development methods used in many industries which are proven to minimise the risk of errors.

HCC's TCP/IP stack was developed using a 'V model' methodology. It uses a strongly typed subset of the 'C' language based on a strict adherence to MISRA[1] compliance and is supplied with detailed static and dynamic analysis reports. It is also supplied with test suites that verify interoperability and code integrity. HCC's networking stack provides a significant range of protocols as well as support for both IPv4 and IPv6, providing flexibility and long-term network compatibility.

[1] „MISRA" is a registered trademark of MIRA Ltd, held on behalf of the MISRA Consortium. No endorsement by MISRA is claimed or implied for any product.

Supported Protocols

The following protocols are available for HCC's MISRA-compliant TCP/IP Stack



High Quality TCP/IPv4, IPv6, Dual IPv4/v6 Stacks

The key to a successful embedded application is to use high-quality software that is verifiably developed and ensures a stable, low-risk development platform. HCC's TCP/IP stack was developed with a rigorous approach to quality using a strongly typed subset of the 'C' language. All stacks are available with an extensive set of applications and protocols and can be supplied tightly integrated with HCC's other storage and communications solutions if required.

All stacks are provided with optimized Ethernet drivers and will integrate easily with any RTOS. The software was designed to provide high performance on embedded microcontrollers. There are no unnecessary copies, static memory management is carefully thought out, and it fully exploits dedicated memory areas and cache. HCC's networking solution provides a significant range of protocols as well as support for both IPv4 and IPv6, providing reliability and long-term network compatibility.

Verifiable TLS/DTLS

TLS/SSL is a highly optimized software module designed to provide secure network communications for embedded devices. It is delivered with a full MISRA compliance report. The importance of using a strong development process and source code control has been emphasized by a number of high-profile security problems caused by source code errors. Network security requires a high degree of quality and traditional methods of 'freestyle coding' and test do not provide sufficient guarantees of correctness.

HCC's verifiable Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS) provides a framework for secure communication in networks based on the TCP/IP or UDP protocols. The module implements TLS 1.2 and, optionally, downrev versions (including SSL 3.0).

- Low memory footprint - typically around 20KB of ROM or 8KB of RAM.
- Typically uses a standard Sockets interface, allowing easy integration.
- TLS 1.0, 1.1 and 1.2 (RFC 5246) and SSL 3.0 and is verifiable.
- DTLS version 1.2 (RFC 6347) and version 1.0 (RFC 4347).
- Heartbeat extensions (RFC 6520).
- HTTP over TLS (RFC 2818).
- HTTPS Secure Server, HTTPS Secure Client and FTPS support
- HCC's Embedded Encryption Manager (EEM) providess full certificate management.
- Supports wide range of cipher suites including the following algorithms:
  DH/DHE/DSS/ECDHE/RSA, AES/RC4/3DES, SHA/MD5

Verifiable IPSec & IKE

HCC's IPv4 & IPv6 stack is supported by an extensive set of protocols and applications and, thanks to the development of a 'clean' stack developed for embedded systems, provides unrivalled performance and security. All software components are created using a strong development process and are supplied with quality verification including a full MISRA compliant static analysis report. Other 'v-model' life cycle artifacts are available on request.

IPSec provides VPN security in embedded applications such as cars, 'point-of-sale' terminals, medical devices, industrial equipment and many others. It ensures integrity, confidentiality and authentication between two devices in a network, providing strong defense against threats such as 'man in the middle' attacks and packet sniffers.

MQTT

MQTT is a small, low-bandwidth networking protocol ideally suited for connecting the growing number of embedded applications that are remotely monitored through an Internet connection. HCC's MQTT implementation runs on its trusted TCP/IP stack and uses verifiable TLS for secure connections.

- Complete implementation of all MQTT features.
- Client can be publisher and/or subscriber to a configurable range of topics on multiple MQTT Brokers.
- All QOS (Quality of Service) levels supported.
- Full MISRA compliance report available.
- Tested with many well known brokers in secure & non-secure mode.
- Can be used with HCCs verifiable TLS to ensure completely secure IoT cloud connections.

When a secure connection is required HCC provides a verifiable TLS module to handle encryption independently of MQTT. Additionally a client can provide a user name and password so that the broker can authenticate the client. When MQTT operates over a TLS connection, both the client and the server can authenticate each other using x.509 certificates.

SNMP

SNMP is a protocol developed to give devices connected on a network a consistent and reliable way to share information. HCC provides a high quality SNMPv2 and SNMPv3 implementation to provide embedded devices with secure network management capability. Using SNMP engineers can monitor device operation, usage, detect network faults or inappropriate access and configure remote devices. SNMP is designed to be robust and used on a large number of network devices with minimal impact on the managed nodes, low transport overheads and to keep working even when other applications fail.

- SNMP agent supporting v1, v2c, v3
- SNMP manager capability to query remote agents
- MIB compiler for easy integration of any MIB
- Can be integrated with HCC's network stack using the standard UDP interface

HTTP/HTTPS

HCC provides a highly flexible web-server solution for embedded systems, allowing the creation of dynamic content within a highly secure environment. 'HTTP-Secure' (HTTPS) provides secure communication over computer networks. It operates as a request-response protocol in the client/server model. The secure client may be a web browser, while an application hosting a website may be the secure server.
HTTPS resources are identified and located on the network using Uniform Resource Identifiers (URIs). HTTPS secure operation relies on using HCC's Transport Layer Security (TLS) module. TLS provides security by encrypting the whole HTTPS message, including the header and the request/response content.

- Compliant with RFC 2818.
- Designed for integration with both RTOS and non-RTOS based systems. Can be configured to use BSD sockets.

- Supports all standard HTTP methods: GET, PUT, POST, and DELETE. Supports HTTP Secure (HTTPS) connections
- Handles a configurable number of simultaneous connections.
- Handles static ROMed pages.
- Can be connected to any file system and process pages received from it.
- Pages may contain dynamic content that can be created by user-specified functions. Supports dynamic variables from tags in HTML.
- Supports optional user authentication based on user name and IP address (as a sample).

Embedded Encryption Manager

HCC's Embedded Encryption Manager (EEM) allows developers to secure embedded systems using multiple encryption or hash algorithms through a uniform interface. Using a well-defined interface shortens development time as developers can now simply drop-in the EEM and encrypt data stored on flash or transmitted across a network. Such security is necessary to block potential hackers looking for a backdoor to access embedded system data.

Developed using a formal process, the EEM undergoes verification to ensure stability and enhanced integrity. It is delivered with a full MISRA compliance report. This level of verifiable quality in the area of security and encryption stands in direct contrast with the widely used 'code-then-test' methods, which have resulted in serious security breaches, such as Heartbleed.

Available algorithms include AES, 3DES, DSS, ECC, EDH, MD5, RSA, SHA and Tiger.

Small Footprint, High Throughput, Low CPU Cycle Operation

An innovative approach to design has resulted in an extremely high-speed data transfer rate, with minimal system resource requirements.

Tests have shown that HCC's packet processing runs faster than comparable embedded stacks, while using around 14kB of ROM, in a typical application scenario[2]. RAM requirements can vary widely depending on application needs but are typically as low as 12kB. It is possible, with a minimum configuration UDP application, to use less than 5kB of ROM and a few hundred bytes of RAM (plus network buffers). Supported features include;

- No dynamic memory allocation (no malloc/free)
- Standard BSD sockets interface
- Zero copy
- 85Mb/s bi-directional throughput on STM32F457with 27% CPU idle time
- Small footprint (RAM/ROM)
- High speed data transfer
- Low power consumption due to low CPU overhead
- Verified compatibility with most popular embedded RTOSes
- Efficient operation without an RTOS

[2] Based on measurements taken using the LPC2468 MCU

Broad Range of Target Processors & Tools

HCC's MISRA-compliant TCP/IP can operate efficiently on a broad range of target processors. Designed so that it can be ported easily and quickly to new architectures, the stack is available with drivers for a range of leading processors.

RTOS Abstractions

RTOS abstractions are available for the following systems: CMX RTX, eCOS, emBOS, EUROS, FreeRTOS, Keil RTX, Nucleus, Quadros RTXC, ThreadX, µ-velOSity, µC/OS-II, and many others. Importantly, for custom schedulers and superloops, HCC offers an abstraction for 'No RTOS'. We also offer our own eTaskSync, a small cooperative scheduler, which is designed to handle all processing and interface requirements of HCC middleware. This means that developers can choose our robust quality and outstanding performance irrespective of their legacy software.

Extensive Compiler Support

Eclipse/GCC, IAR Embedded Workbench, Keil ARM Compiler, Freescale CodeWarrior, Atmel AVR Studio, Green Hills Multi, Microchip MPLAB, Renesas HEW, TI Code Composer Studio, Mentor CodeSourcery, Atollic True Studio and many more.

Microcontrollers

ARM Cortex-M0/M1/M3/M4/R4/A8, ARM7/9/11; **Atmel** AVR32, SAM3/4/7/9; **Freescale** ColdFire, Kinetis, PowerPC, i.MX, Vybrid, QorIQ; **Infineon** C164, XMC1000, XMC4000; **Microchip** PIC24, PIC32; **NXP** LPC1300/1700/1800/2000/3000/4000; **Renesas** SuperH, RX, RL, 78k; **SiliconLabs** EFM32, SIM3; **Spansion** FM0/FM3/FM4; **STMicroelectronics** STM32; **Texas Instruments** MSP430, Stellaris, C2000, Hercules, DaVinci, Sitara, Tiva; **Toshiba** TMP M0/M3; **Xilinx** Zynq;

Licensing & Purchasing

All HCC reusable software components are royalty-free and distributed in source form with support and maintenance included for one year with all purchases. We deliver sample projects tailored to an environment agreed with customers to ensure the quickest possible start. Visit HCC's website to find a sample license and to obtain the contact details of your local sales representative. Or, simply send an email to info@hcc-embedded.com and we will send all the details you require.

HCC embedded

US sales office: +1 310 265-1304
European sales office:
1133 Budapest, Váci út 76., Hungary • **Tel.:** +36 1 450 1302
info@hcc-embedded.com • sales@hcc-embedded.com • **www.hcc-embedded.com**

07032017