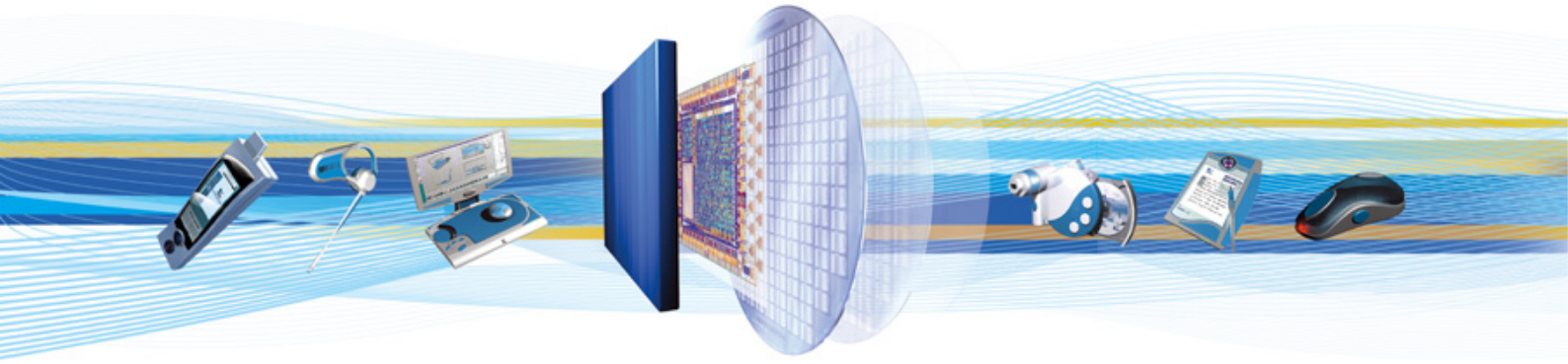
CSR Synergy Framework 3.1.0

CSR IP Architecture

August 2011



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 Figures

1 Introduction

The CSR Synergy Framework defines APIs for interfacing to a TCP/IP stack collectively known as CSR IP. The interfaces concern the following three aspects of a TCP/IP stack:

1. IP stack configuration (CSR_IP_IFCONFIG) – [SYN-FRW-IFCONFIG-API]
2. Application-level communication (CSR_IP_SOCKET) – [SYN-FRW-SOCKET-API]
3. Exchanging IP-datagrams with other IP stacks (CSR_IP_ETHER) – [SYN-FRW-ETHER-API]

These aspects are orthogonal and thus are divided into three different APIs such that an implementation for a given platform only needs to handle the functionality that is required by the CSR Synergy software used.

2 CSR IP Architecture

The figure below outlines the overall architecture of CSR IP component:

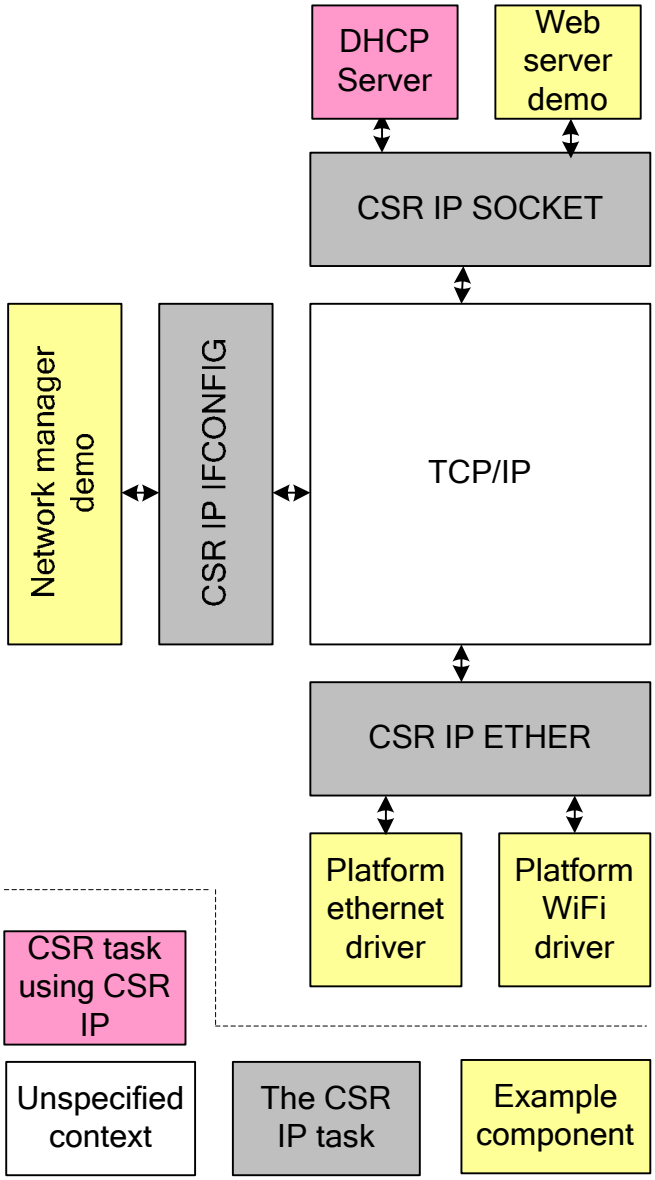


Figure 1: CSR IP architecture and example uses

As can be seen, there are three interfaces to the TCP/IP stack for CSR Synergy components. These handle the fundamental aspects of an IP stack from an API and are described in detail in the following. It is very important to note that a platform does not have to provide all three interfaces; they have no API dependencies between them, a platform only needs to provide the interfaces that CSR Synergy components depend on. In this regard, it is important to note that while the three interfaces are completely orthogonal, CSR Synergy code assumes they *are implemented in a single task*.

CSR IP SOCKET is the interface used by CSR Synergy components that need to communicate over TCP or UDP. This interface supports creating TCP, UDP, and raw sockets and to transmit and receive data over them.

CSR IP IFCONFIG is used to configure the lower part of the IP stack that handles transmission and reception of IP datagrams over network interfaces. A network interface in the CSR IP ETHER context means any medium,

physical or logical, which is capable of exchanging IP datagrams with peers. This interface is used to register network interfaces in the network stack such that they can be used for communication.

An IP stack must be configured in order to operate. Such configuration involves setting addresses for network interfaces, picking DNS-servers, and setting up routing. CSR IP IFCONFIG is used to provide a simplified interface to this aspect of an IP stack. This interface is used by CSR Synergy components that need to control the configuration.

The three interfaces are used by CSR Synergy components via the standard CSR Synergy message-passing interface described in [SYN-FRW-SCHED-API]. However, it is important to notice that there is *no requirement* that the IP stack itself runs in the scheduler. CSR Synergy Framework provides examples of both the IP stack running outside the scheduler (in the pcwin and pclin ports) as well as running the IP stack inside (the bdb2 and bdb3 ports).

3 Generic Ethernet Manager

Figure 1 shows a demo network manager configuring the IP stack through the CSR IP task. The framework ships with a generic ethernet manager task that can be used in applications that run on platforms where the operating system does not handle configuration of network interfaces. The task is fully autonomous and does not require (or support for that matter) any configuration.

The generic ethernet manager task uses the CSR IP IFCONFIG interface to listen for changes in the current network interface configuration. Whenver it is notified of a network interface in the down state, it will attempt to configure it using DHCP.

As mentioned above, the task is fully autonomous, but it does have limited user interaction. The ethernet manager task uses the CSR APP and CSR UI interfaces to create a menu (Ethernet | Manager) in which currently known network interfaces can be seen along with the current state (no cable, down, or the currently configured IP address.

Using the task is as simple as registering a task with the queue and functions declared in the `gsp/inc/csr_ethernet_handler_task.h` header:

```
CsrSchedRegisterTask(&CSR_ETHERNET_MANAGER_IFACEQUEUE,

    CsrEthernetManagerInit,

    CsrEthernetManagerDeinit,

    CsrEthernetManagerHandler,

    "CSR_EXAMPLE_ETHMGR",

    data, 0);
```

4 DHCP Server

The DHCP Server shown in Figure 1 is a full featured DHCP Server shipped with the framework. A description of the DHCP Server can be found in [DHCP-SERVER-API]. The DHCP Server is a generic task running in the scheduler. A demo application using the DHCP Server called DHCP Server Manager is also shipped with the framework. The DHCP Server Manager task uses the CSR APP and CSR UI interfaces to create a menu in which is possible to create, start and stop the server. Furthermore, it is also possible to list the current leases of the server.

5 Example implementations

In this section, a number of possible scenarios are described along with porting requirements.

5.1 PC Windows

Windows has its own IP stack running inside the operating system kernel, which applications interface with using system calls. The CSR Synergy Framework port of CSR IP for PC Windows consists of a thin wrapper layer around the system call interface to provide the CSR IP SOCKET interface as shown in the figure below:

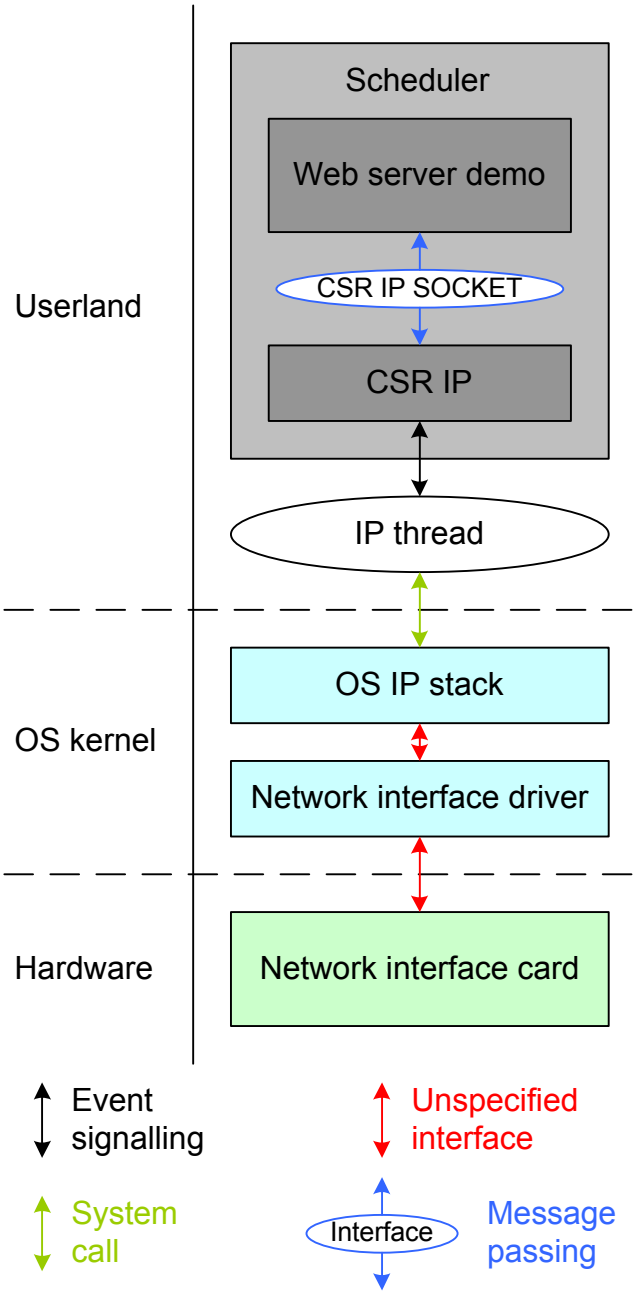


Figure 2: PC Windows implementation

The operating system handles configuration of the IP stack as well as the communication path between network interfaces and the IP stack, so these two aspects are not provided in the PC Windows port.

As can be seen in the figure, a thread is used to decouple the asynchronous CSR IP interface from the synchronous system call interface used in the operating system. This thread communicates with the CSR IP task using shared memory and events and communicates with the operating system using events and system calls.

5.2 Generic IP implementation

The CSR Synergy Framework also contains a generic implementation of CSR IP with an IP stack, lwIP, running inside the scheduler. This can be used on platforms where there the operating system (e.g. Nucleus) or a different component does not provide an IP stack and there are CSR Synergy components that require CSR IP.

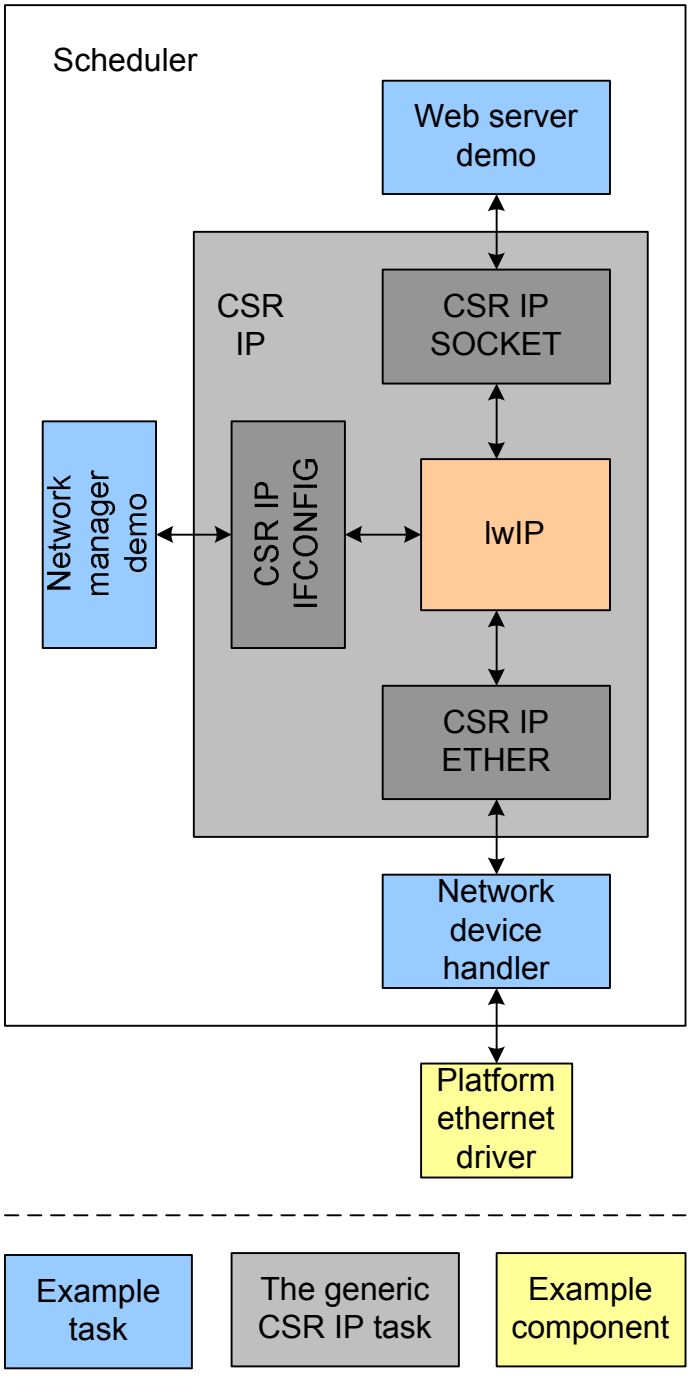


Figure 3: Generic CSR IP implementation

Because the operating system does not provide an IP stack, it also does not handle configuration of it, and thus a network manager task must be used to control network interfaces in order for network communication to be

possible. Network interface drivers must implement the CSR IP ETHER interface to be used by the generic CSR IP implementation.

6 Document References

[SYN-FRW-SCHED-API]	Synergy Framework Scheduler API. Doc. api-0004-sched
[SYN-FRW-ETHER-API]	Synergy Framework Ether API. Doc. api-0017-ether
[SYN-FRW-IFCONFIG-API]	Synergy Framework Ether API. Doc. api-0019-ifconfig
[SYN-FRW-SOCKET-API]	Synergy Framework Ether API. Doc. api-0018-socket
[DHCP-SERVER-API]	Synergy Framework DHCP Server API. Doc. api-0032-dhcp_server

Terms and Definitions

Abbreviation	Explanation
CSR	Cambridge Silicon Radio

Document History

Revision	Date	History
1	2010-08-20	First draft
2	2010-08-23	Second draft
3	2010-08-25	Document ethernet manager task
4	DEC 10	Ready for release 3.0.0
5	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.