



## CSR Synergy Bluetooth 18.2.0

### DUN Gateway

### Demo Description

November 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](http://www.csr.com)



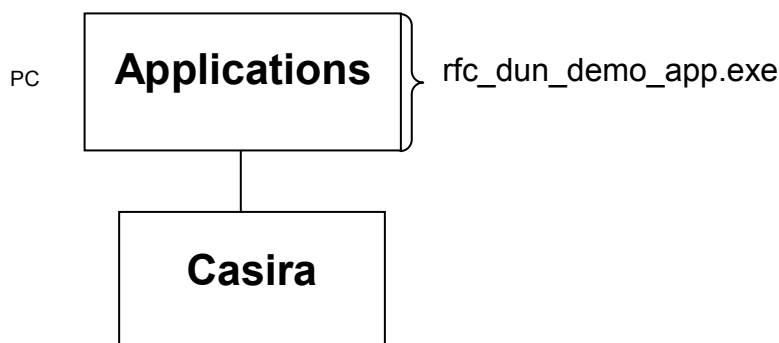
## Contents

<b>1</b>	<b>Dial-Up Networking (DUN) Profile</b>	<b>3</b>
1.1	Generally	3
1.2	Use of Program dun_demo_app.exe	3
	Applications	3
	Casira	3
<b>2</b>	<b>Linux</b>	<b>5</b>
	Terms and Definitions	6
	Document History	6
	TradeMarks, Patents and Licences	7
	Life Support Policy and Use in Safety-critical Compliance	7
	Performance and Conformance	7

# 1 Dial-Up Networking (DUN) Profile

## 1.1 Generally

The DUN provides the GW functionality of the DUN profile. This demo is running with a CASIRA with RFCOMM-build firmware.



The Dial-up networking Gateway (DG) profile defines the protocols and procedures being used by devices implementing the usage model called 'internet bridge'.

The scenarios covered are the following:

- Usage of a cellular phone or modem by a computer as a wireless modem for connecting to a Dial-Up Internet access server, or using other Dial-Up services
- Usage of a cellular phone or modem by a computer to receive data calls

The following roles are defined:

**Gateway (GW):** This is the device that provides access to the public network. Typical devices acting as gateways are cellular phones and modems.

**Data Terminal (DT):** Device that uses the Dial-Up services of the gateway. Typical devices acting as data terminals are laptops and desktop PCs.

The application has been made to run on Windows and Linux and may be connected to the Casira using either a serial connection using BCSP (`rfc_dun_demo_app.exe`), a serial connection using H4DS (`rfc_dun_demo_app_h4ds.exe`) or an USB connection (`rfc_dun_demo_app_usb.exe`).

On Windows the DG demo application and the underlying layers all run in the same process while on Linux it is possible to:

- 1) have the DG demo application running together with the underlying layers in one process.
- 2) have the DG demo application running in one process and the rest of the protocol stack in another process, communicating between the two processes by means of TCP/IP.
- 3) have the DG demo application running in user space and the rest in kernel space.
- 4) have the DG demo application running in user space and the rest in kernel space as a standard Linux serial port (TTY) device.

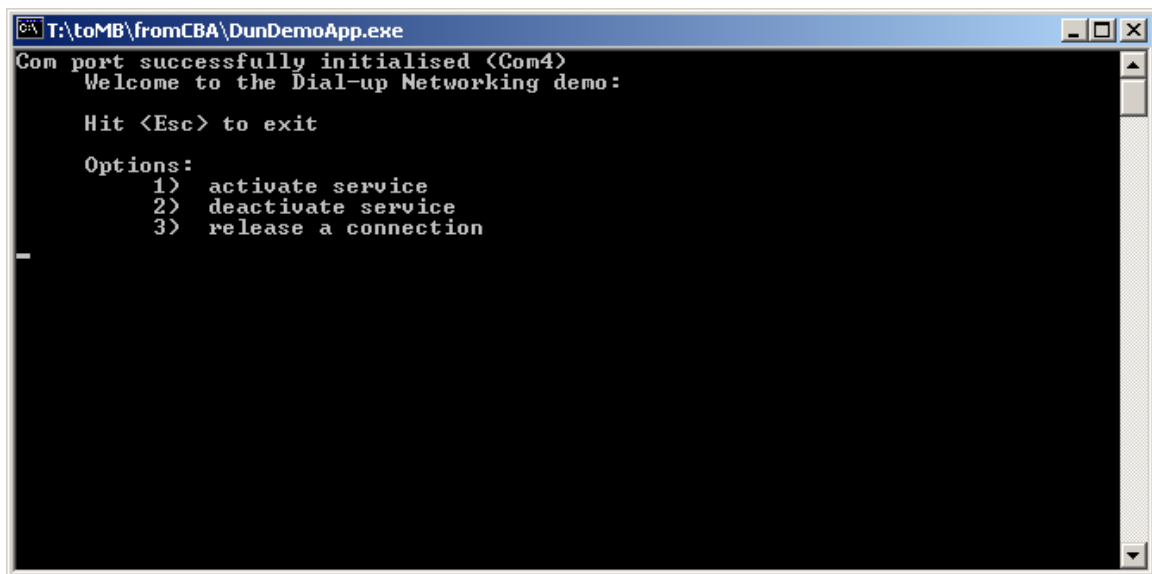
## 1.2 Use of Program `dun_demo_app.exe`

**Note:** This description is for CSR Synergy Bluetooth RFCOMM. The functionality of the application for the HCI build is identical. The only difference is the naming: `rfc_dun_demo_app.exe` versus `hci_dun_demo_app.exe`.

The DUN scenario has two sides: one gateway side (the `dun.exe` program) and one data terminal side (e.g. a standard laptop).

Start the program `rfc_dun_demo_app.exe`:

1. Choose the COM port on which the modem is connected, by giving parameters to the program, e.g. using a 'command prompt'. An example could be `rfc_dun_demo_app.exe -D COM4`. At start up COM4 is selected as default.
2. Choose the COM port on which the Bluetooth module is connected, by specifying the `-C` parameter to the program, e.g. `rfc_dun_demo_app.exe -C COM1`. At start up COM1 is selected as default.
3. Choose baudrate for the COM port on which the Bluetooth module is connected, by specifying the `-B` parameter to the program, e.g. `rfc_dun_demo_app.exe -B 115200`. If no parameter is specified the default is 115200.
4. Choose a specific device for default connection by specifying the `-A` parameter to the program, e.g. `rfc_dun_demo_app.exe -a 0002:5b:AEAE20`. If no address is specified it is necessary to perform a search for servers in order to establish a connection.
5. DUN valid commands are listed on the screen, see illustration below.



```

T:\toMB\fromCBA\DunDemoApp.exe
Com port successfully initialised (Com4)
Welcome to the Dial-up Networking demo:

Hit <Esc> to exit

Options:
1> activate service
2> deactivate service
3> release a connection
  
```

1. Choose e.g. '1' – accept that PC/client connects to the GW
2. Start the client side by searching for devices in the vicinity and connect
3. Once connected normal network services are available
4. To stop scanning press '2'
5. To disconnect an established connection press '3'

## 2 Linux

This section describes how to build and run the DG demo application on Linux. Prior to compiling any of the Linux demos the CSR Synergy Bluetooth libraries must be compiled from the root of the CSR Synergy Bluetooth directory tree. The User Guide describes how this is done.

The basic DG demo application (pure user space), located in `./applications/dun_gateway/`, may be compiled on Linux by means of:

```
> make clean all TARGET_ARCH=Linux-2.6-x86
```

This will output six files: `hci_dun_demo_app`, `hci_dun_demo_app_h4ds` and `hci_dun_demo_app_usb`, for serial BCSP communication, serial H4DS communication and USB communication using a HCI split and `rfc_dun_demo_app`, `rfc_dun_demo_app_h4ds` and `rfc_dun_demo_app_usb`, for serial BCSP communication, serial H4DS communication and USB communication respectively using a RFC split.

The demo applications are used like described above for Windows.

Besides the basic DG demo application, where everything is running in user space, three additional SG demo application set-ups may be run, namely:

- 1) DG demo application and CSR Synergy Bluetooth protocol stack split, communication over TCP/IP (located in `./applications/dun_gateway/projects/linux/usr2usr`)
- 2) DG demo application and CSR Synergy Bluetooth protocol stack split, between user and kernel space (located in `./applications/dun_gateway/projects/linux/usr2kernel`)
- 3) DG demo application and CSR Synergy Bluetooth protocol stack split between user and kernel space utilizing the TTY kernel driver (located in `./applications/dun_gateway/projects/linux/usr2kernel`)

## 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

## Document History

Revision	Date	History
1	26 SEP 11	Ready for release 18.2.0

## TradeMarks, Patents and Licences

Unless otherwise stated, words and logos marked with <sup>™</sup> or <sup>®</sup> are trademarks registered or owned by CSR plc or its affiliates. Bluetooth<sup>®</sup> 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.

## 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](http://www.csrsupport.com) for compliance and conformance to standards information.