RW BLE Cycling Speed and Cadence Profile Interface Specification

Interface Specification
RW-BLE-CSCP-IS
Version 8.0
2015-07-29

RivieraWaves
http://www.rivierawaves.com/

Revision History

Version	Date	Revision Description	Author
0.1	April 30th 2013	Initial draft	LT
1.0	June 24th 2013	Initial release	LT
7.0	Oct 13th 2014	Updated for BLE 4.1	CM
8.0	July 29th 2015	Updated for BLE 4.2	CM

Table of Contents

1 Overview

1.1 Document Overview

This document describes the non-standard interface of the RivieraWaves (RW) Bluetooth Low Energy (BLE) Cycling Speed and Cadence Profile (CSCP) implementation. Along this document, the interface messages will be referred to as API messages for the profile block(s).

Their description will include their utility and reason for implementation for a better understanding of the user and the developer that may one day need to interface them from a higher application.

1.2 BLE Cycling Speed and Cadence Profile Overview

The CSCP enables a collector device to connect and interact with a Cycling Speed and Cadence Sensor for use in sport and fitness applications.

This service has been implemented as a profile. Within this profile, two roles can be supported: Sensor role (CSCPS) and Collector role (CSCPC). The Collector role must support the GAP Central Role and the Sensor role, the GAP Peripheral role. The profile requires a connection to be established between the two devices for its functionality.

The various documents edited by the Bluetooth SIG present different use cases for this profile, their GATT, GAP and security, mandatory and optional requirements. The Cycling Speed and Cadence Profile specifications have been adopted by the Bluetooth SIG on August 7th 2012 ([1] and [3]). Their related Test Specifications have been released at the same time and are referenced in [2] and [4].

The profile is implemented in the RW-BLE software stack as two tasks, one for each role. Each task has an API decided after the study of the profile specifications and test specifications, and it is considered to be minimalistic and designed for a future application which would combine the profile functionality with the device connectivity and security procedures.

The structure of the Cycling Speed and Cadence service is defined in the table below:

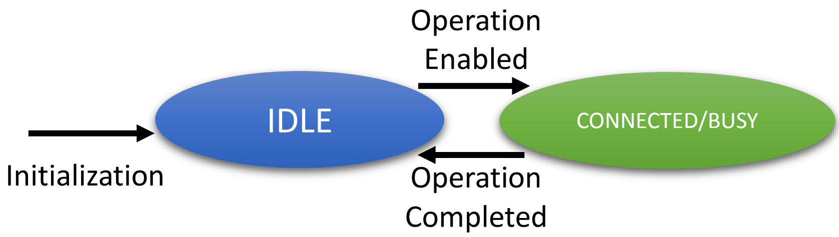
Characteristic Name	Requirements	Properties	Security	Descriptors
CSC Measurement	Mandatory	Notify	None	Client Characteristic Configuration
CSC Feature	Mandatory	Read	None	None
Sensor Location	Mandatory if the Multiple Sensor Location feature is supported, otherwise optional	Read	None	None
SC Control Point	Mandatory	Write/Indicate	None	Client Characteristic Configuration

2 CSCP Sensor Role API

2.1 Environment

Within the CSCPS task, two states are defined IDLE and CONNECTED/BUSY.

The busy state is used when a procedure is currently being processed by a connected device (read, write, …). When the state is busy, no command message sent by a higher layer can be handled, this message will be stored until the end of the procedure and handled once the procedure is over. Thus it can be considered as a connected state from an application point of view.



2.2 API Messages

2.2.1 Initialization

During the initialization phase of the Cycling Speed and Cadence Sensor, the memory for this task must be allocated using the message GAPM_PROFILE_TASK_ADD_CMD provided by the GAPM interface. Apart from the security level, the following parameters should be filled:

Parameters:

Type	Parameters	Description
uint16_t	csc_feature	CSC Feature Value - Not supposed to be modified during the lifetime of the device. This value is used to decide if the SC Control Point Characteristic is part of the Cycling Speed and Cadence service (see Table 2).
uint8_t	sensor_loc_supp	Indicate if the Sensor Location characteristic is supported. Note that if the Multiple Sensor Location feature is set has supported in the csc_feature parameter, the characteristic will be added (mandatory).
uint8_t	sensor_loc	Sensor location, used if the Sensor Location characteristic is added in the database. (see Table 3)

Description: This API message shall be used to add one instance of the Cycling Speed and Cadence Service in the database.

The SC Control Point characteristic will be added if at least one of the following features is supported:

- Wheel Revolution Data
- Multiple Sensor Locations

2.2.2 CSCPS_ENABLE_REQ

Source: TASK_APP

Destination: TASK_CSCPS

Required State: IDLE

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint16_t	csc_meas_ntf_cfg	CSC Measurement Characteristic - Saved Client Characteristic Configuration Descriptor Value for a bonded device. • DISABLE = PRF_CLI_STOP_NTFIND • ENABLE = PRF_CLI_START_NTF
uint16_t	sc_ctnl_pt_ntf_cfg	SC Control Point Characteristic - Saved Client Characteristic Configuration Descriptor Value for a bonded device. • DISABLE = PRF_CLI_STOP_NTFIND • ENABLE = PRF_CLI_START_IND

Response: CSCPS_ENABLE_RSP

Description: This API message shall be used after the connection with a peer device has been established in order to set the bonding data.

2.2.3 CSCPS_ENABLE_RSP

Source: TASK_CSCPS

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint8_t	status	Status of the operation

Description: This message corresponds to the response of setting bond data operation.

2.2.4 CSCPS_NTF_CSC_MEAS_REQ

Source: TASK_APP

Destination: TASK_CSCPS

Required State: CONNECTED

Parameters:

Type	Parameters	Description
uint8_t	flags	Flags
uint16_t	cumul_crank_rev	Cumulative Crank Revolution
uint16_t	last_crank_evt_time	Last Crank Event Time
uint16_t	last_wheel_evt_time	Last Wheel Event Time
int16_t	wheel_rev	Wheel Revolution since the last wheel event time.

Response: CSCPS_NTF_CSC_MEAS_RSP

Description: This API message shall be used by the application to send a CSC Measurement notification to every connected device. This profile checks whether the peer device has enable sending of notifications for the characteristic and sends them according to its value.

The wheel_rev value is added to the total wheel revolution value stored in the environment. The total value is then sent to the peer device.

2.2.5 CSCPS_NTF_CSC_MEAS_RSP

Source: TASK_ CSCPS

Destination: TASK_ APP

Parameters:

Type	Parameters	Description
uint8_t	status	Status of the operation
uint32_t	tot_wheel_rev	Cummul Wheel revolution value

Description: This API message is sent once the notification has been send to the connected devices.

2.2.6 CSCPS_SC_CTNL_PT_REQ_IND

Source: TASK_CSCPS

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint8_t	op_code	Operation Code (see Table 4)
union	value	
uint32_t	cumul_value	Cumulative Value (Total Distance)
uint8_t	sensor_location	Sensor Location (see Table 3)

Description: The message is sent to the application when the SC Control Point characteristic is written by the peer device. The application shall answer using the CSCPS_SC_CTNL_PT_CFM message.

2.2.7 CSCPS_SC_CTNL_PT_CFM

Source: TASK_CSCPS

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint8_t	status	Status (PRF_ERR_OK if the request is accepted)
union	value	
uint8_t	sensor_location	Sensor Location
uint16_t	supp_sensor_loc	Supported sensor locations
uint32_t	cumul_wheel_rev	New Cumulative Wheel revolution Value

Description: This message is sent by the application as a response to the CSCPS_SC_CTNL_PT_REQ_IND message. It contains the value requested by the profile.

In the case where this message is received while no request message had been sent, it will be automatically dropped.

2.2.8 CSCPS _CFG_NTFIND_IND

Source: TASK_CSCPS

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint8_t	char_code	Characteristic Code (CSC Measurement characteristic or SC Control Point characteristic)
uint8_t	ntf_cfg	Notification configuration new value

Description: This message is sent to the application each time a peer device successfully writes the Client Characteristic Configuration descriptor of either the CSC Measurement characteristic or the SC Control Point characteristic.

2.2.9 CSCPS_CMP_EVT

Source: TASK_CSCPS

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint8_t	operation	Operation Code: • CSCPS_CTNL_PT_CUMUL_VAL_OP_CODE • CSCPS_CTNL_PT_UPD_LOC_OP_CODE • CSCPS_CTNL_PT_SUPP_LOC_OP_CODE • CSCPS_CTNL_ERR_IND_OP_CODE
uint8_t	status	Status

Description: The API message is used by the CSCPS task to inform the sender of a command that the procedure is over and contains the status of the procedure.

3 CSCP Collector Role API

3.1 Environment

Within the CSCPC task, four states are defined: FREE, IDLE, DISCOVERYING and BUSY.

As for the server part of the profile, the connected state and the busy state will be merged together in this document for a better understanding.

Important Note: The TASK_CSCPC task is multi-instantiated, one instance is created for each connection for which the profile will be enabled and each of these instances will have a different task ID. Thus, it is very important for the application to keep the source task ID of the first received CSCPC_CMP_EVT message to be able to communicate with the peer device linked to this task ID once it has been enabled.

The term TASK_CSCPC_IDX will be used in the rest of the document to refer to any instance of the Cycling Speed and Cadence profile Collector Role Task. The term TASK_CSCPC will refer to the first instance of this task.

3.2 API Messages

3.2.1 Initialization

During the initialization phase of the Cycling Speed and Cadence Collector, the memory for this task must be allocated using the message GAPM_PROFILE_TASK_ADD_CMD provided by the GAPM interface.

3.2.2 CSCPC_ENABLE_REQ

Source: TASK_APP

Destination: TASK_CSCPC

Required State: FREE

Parameters:

Type	Parameters	Description
uint8_t	con_type	Connection Type
struct cscpc_cscs_content	cscs	Service structure previously discovered in the database of the peer device.

Response: CSCPC_ENABLE_RSP and CSCPC_CMP_EVT

Description: This API message is used for enabling the Collector role of the CSCP. This Application message contains BLE Connection index, the connection type and the previously saved discovered CSCS details on peer.

The connection type may be PRF_CON_DISCOVERY (0x00) for discovery/initial configuration or PRF_CON_NORMAL (0x01) for a normal connection with a bonded device. Application shall save this information to reuse them for other connections. During normal connection, previously discovered device information can be reused.

For a normal connection, the response to this request is sent right away after saving the CSCS content in the environment and registering CSCPC in GATT to receive the notifications for the known attribute handles in CSCS that would be notified.

For a discovery connection, discovery of the peer CSCS is started and the response will be sent at the end of the discovery with the discovered attribute details.

3.2.3 CSCPC_ENABLE_RSP

Source: TASK_APP

Destination: TASK_CSCPC

Parameters:

Type	Parameters	Description
uint8_t	status	Status of the operation
struct cscpc_cscs_content	cscs	Service structure previously discovered in the database of the peer device.

This message contains the service structure if the discovery operation was performed correctly.

3.2.4 CSCPC_READ_CMD

Source: TASK_APP

Destination: TASK_CSCPC_IDX

Parameters:

Type	Parameters	Description
uint8_t	operation	Operation Code, will be set by the profile task.
uint8_t	read_code	Read Code: • CSCPC_RD_CSC_FEAT • CSCPC_RD_SENSOR_LOC • CSCPC_RD_WR_CSC_MEAS_CFG • CSCPC_RD_WR_SC_CTNL_PT_CFG

Response: CSCPC_VALUE_IND and CSCPC_CMP_EVT

Description: The API message shall be used to read the value of an attribute in the peer device database.

3.2.5 CSCPC_CFG_NTFIND_CMD

Source: TASK_APP

Destination: TASK_CSCPC_IDX

Required State: IDLE

Parameters:

Type	Parameters	Description
uint8_t	operation	Operation Code, fill by the profile
uint8_t	desc_code	Descriptor Code • CSCPC_RD_WR_CSC_MEAS_CFG • CSCPC_RD_WR_SC_CTNL_PT_CFG
uint16_t	ntfind_cfg	NTF/IND Configuration

Response: CSCPC_CMP_EVT

Description: This API message is used to configure sending of notification/indication in the peer device database.

3.2.6 CSCPC_CTNL_PT_CFG_REQ

Source: TASK_CSCPC_IDX

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	operation	Operation code, fill by the profile.
struct cscp_sc_ctnl_pt_req	sc_ctnl_pt	SC Control Point Request

Response: CSCPC_CTNL_PT_RSP

Description: This API message allows writing the value of the SC Control Point characteristic.

If the SC Control Point characteristic has not been found in the peer device database during the discovery procedure, a CSCPC_CMP_EVT message is sent back to the requester with a PRF_ERR_INVALID_HDL error status.

3.2.7 CSCPC_CTNL_PT_RSP

Source: TASK_APP

Destination: TASK_CSCPC_IDX

Parameters:

Type	Parameters	Description
struct cscp_sc_ctnl_pt_rsp	sc_ctnl_pt_rsp	SC Control Point Response

Description: Response from the peer device containing the parameters of the control point.

3.2.8 CSCPC_VALUE_IND

Source: TASK_CSCPC_IDX

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	att_code	Attribute Code • CSCPC_NTF_CSC_MEAS • CSCPC_RD_CSC_FEAT • CSCPC_RD_SENSOR_LOC • CSCPC_RD_WR_CSC_MEAS_CFG
union	value	
struct cscp_csc_meas	csc_meas	CSC Measurement
uint16_t	sensor_feat	CSC Feature
uint8_t	sensor_loc	Sensor Location
uint16_t	ntf_cfg	Client Characteristic Configuration Descriptor Value

Description: This API message is sent to the application when a new value is received from the peer device within a read response, an indication, or a notification.

3.2.9 CSCPC_CMP_EVT

Source: TASK_CSCPC_IDX

Destination: TASK_APP

Parameters:

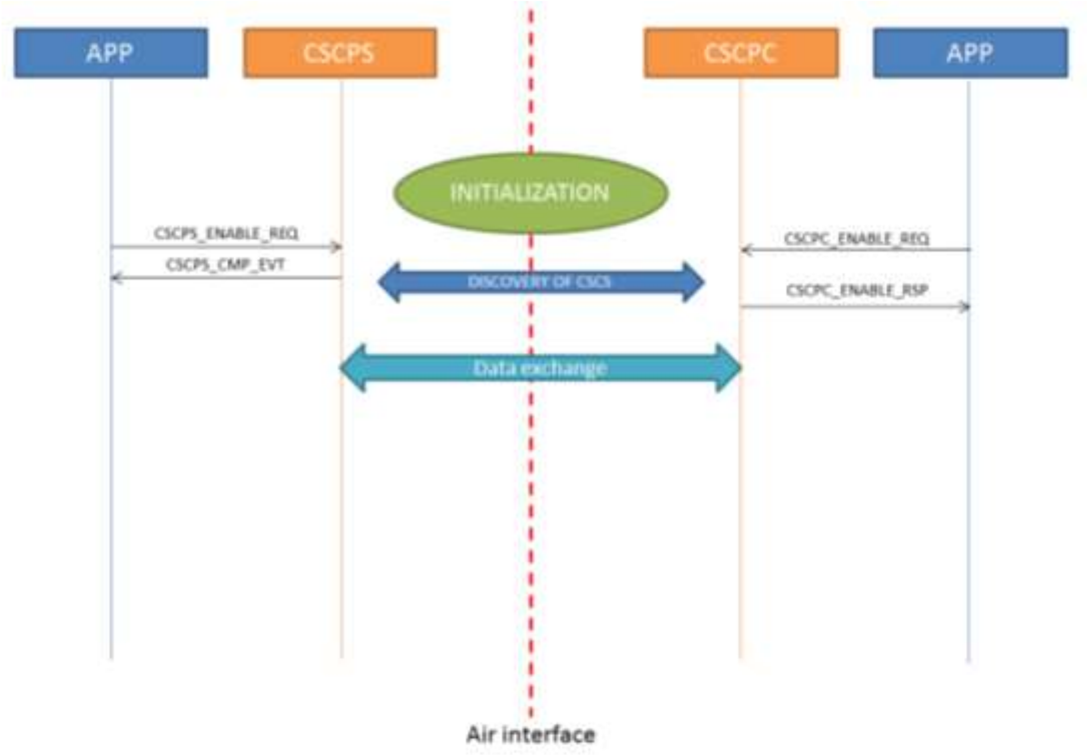
Type	Parameters	Description
uint16_t	conidx	Connection index
uint8_t	operation	Operation Code: • CSCPC_READ_OP_CODE • CSCPC_CFG_NTF_IND_OP_CODE • CSCPC_CTNL_PT_CFG_WR_OP_CODE • CSCPC_CTNL_PT_CFG_IND_OP_CODE
uint8_t	status	Status

Description: The API message is used by the CSCPC task to inform the sender of a command that the procedure is over and contains the status of the procedure.

4 Message Sequence Charts (MSCs)

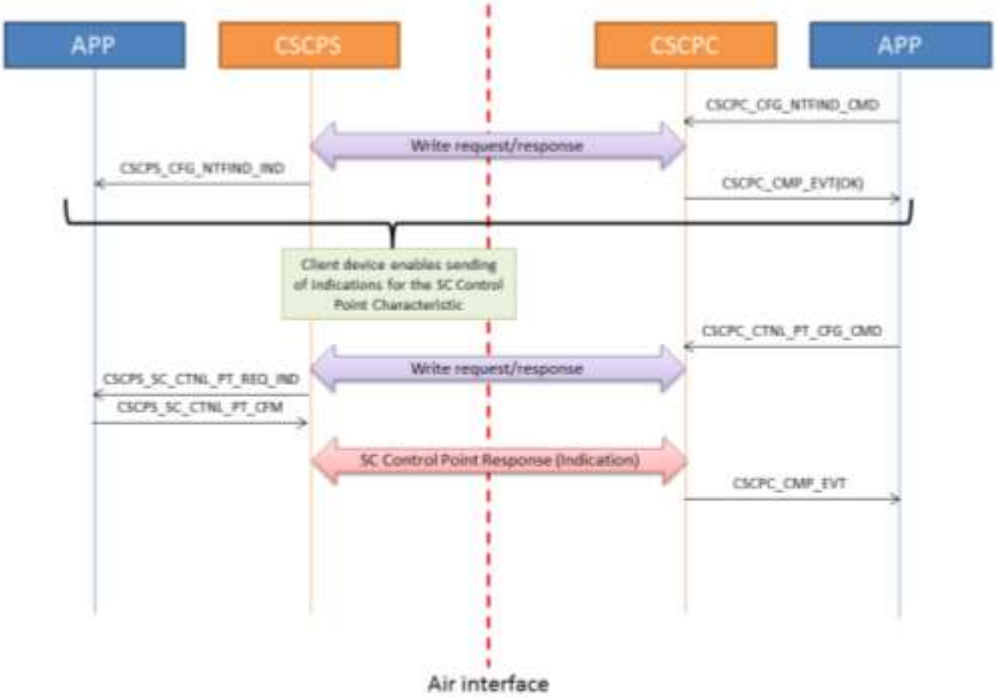
This part describes the different procedure that can be used within the Cycling Speed and Cadence profile. In these MSCs, it is supposed that two RW stacks (one with the server role of the profile and one with the client role) are connected together.

4.1 Device Initialization

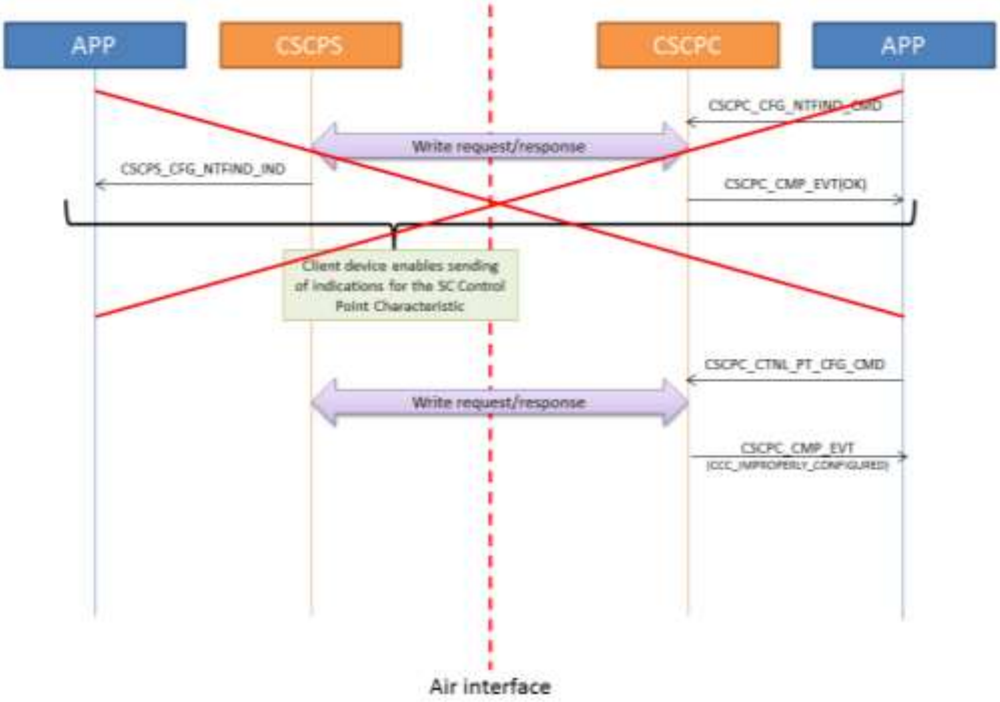


4.2 SC Control Point Characteristic usage

4.2.1 Normal procedure

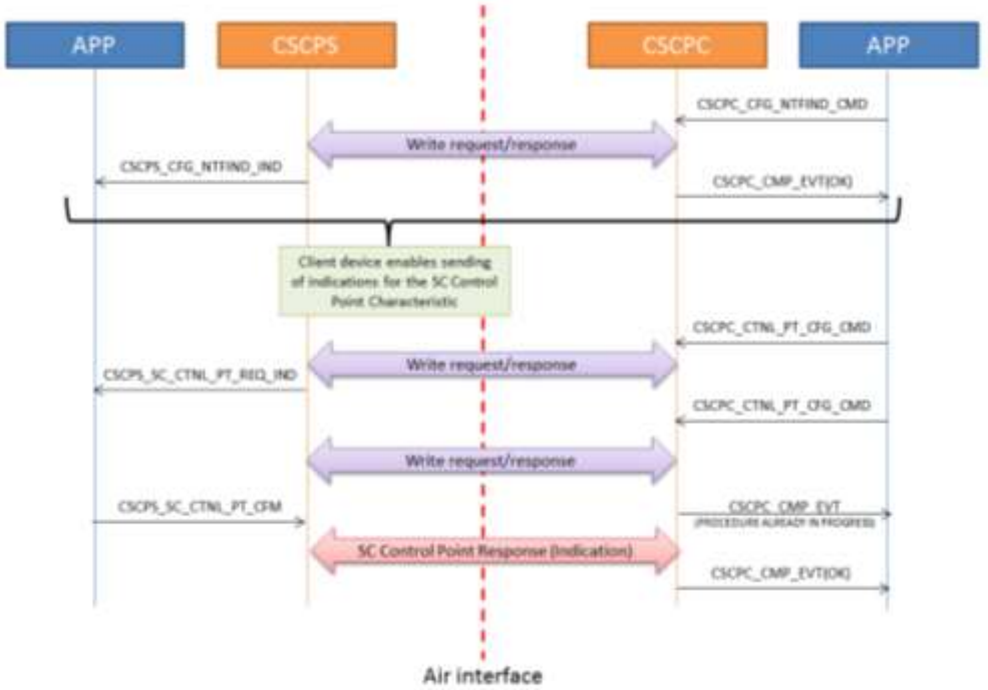


4.2.2 CCC improperly configured Error



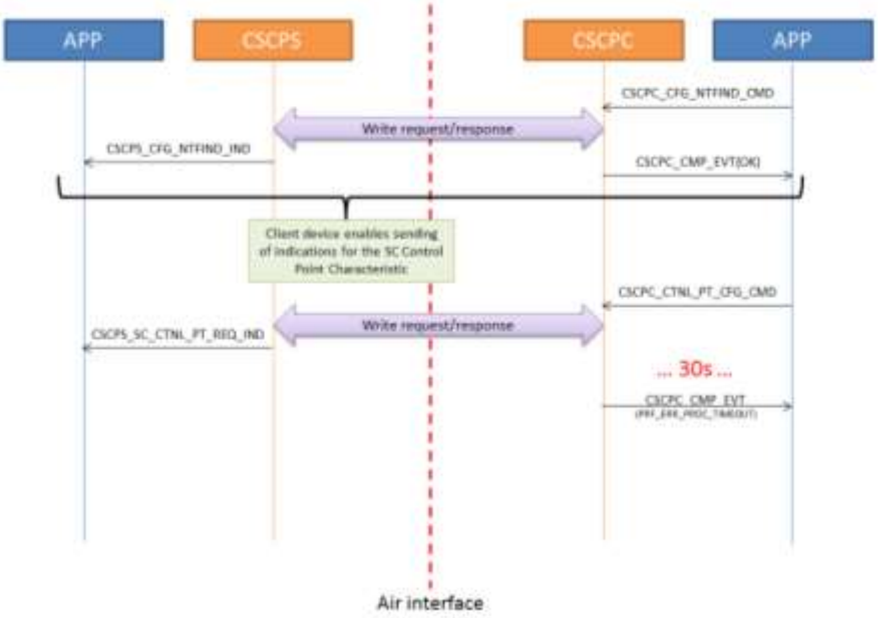
If the client device has not enabled sending of indications, the server device will answer with a CCC_IMPROPERLY_CONFIGURED error.

4.2.3 Procedure Already in Progress Error



If the client device writes the SC Control Point characteristic while the previous procedure is not over, the server will answer with a PROCEDURE_ALREADY_IN_PROGRESS error.

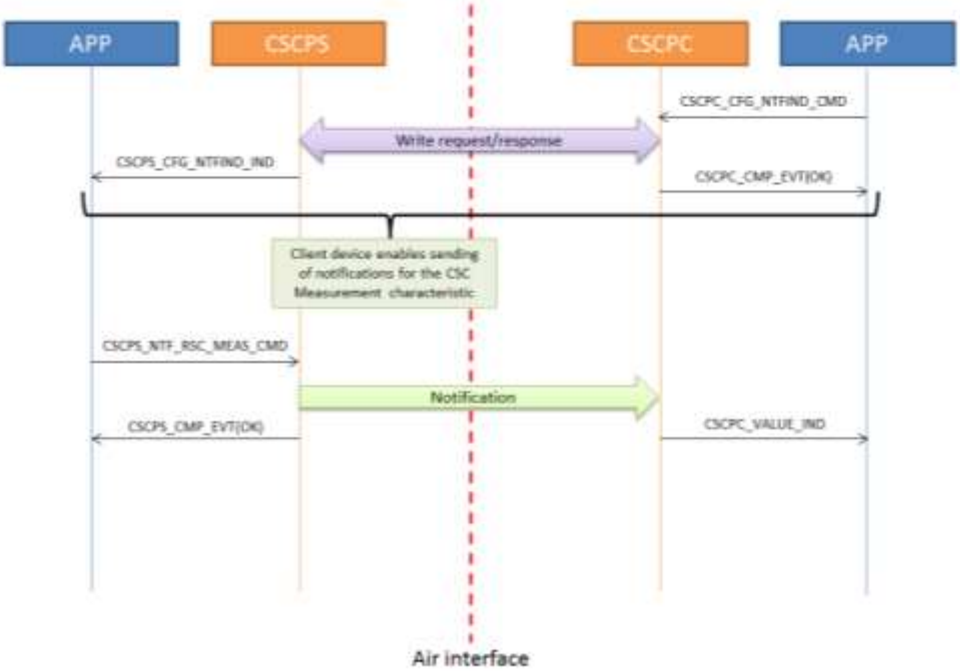
4.2.4 Procedure Timeout



If the client device does not receive a SC Control Point response within 30s after reception of the write response, a procedure timeout error will be raised.

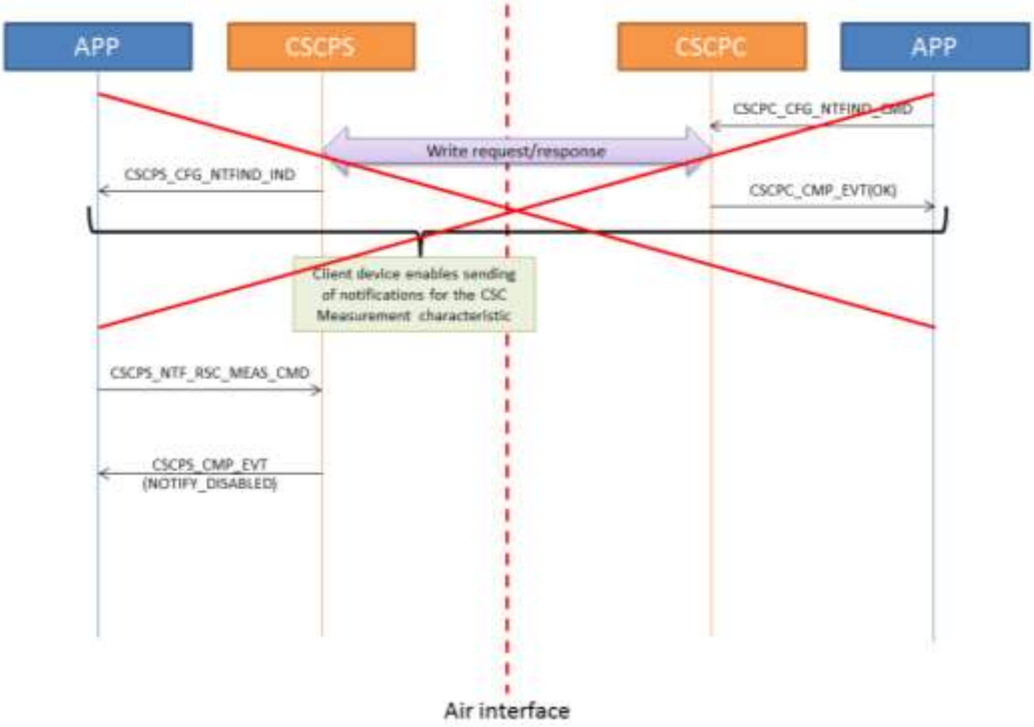
4.3 Sending / Reception of CSC Measurements

4.3.1 Normal procedure



4.3.2 Sending of notifications disabled

If sending of notifications has not been enabled by the collector device, the server device won't be able to send measurements, a PRF_ERR_NOTIFY_DISABLED error will be sent back to the application.



5 Miscellaneous

Name	Value	Description
CSCP_MEAS_WHEEL_REV_DATA_PRESENT	0x01	Wheel Revolution Data Present
CSCP_MEAS_CRANK_REV_DATA_PRESENT	0x02	Crank Revolution Data Present
CSCP_MEAS_ALL_PRESENT	0x03	All parameters are present

Table 1 – CSC Measurement Present Parameters bit flags

Name	Value	Description
CSCP_FEAT_WHEEL_REV_DATA_SUPP	0x0001	Wheel Revolution Data Supported
CSCP_FEAT_CRANK_REV_DATA_SUPP	0x0002	Crank Revolution Data Supported
CSCP_FEAT_MULT_SENSOR_LOC_SUPP	0x0004	Multiple Sensor Location Supported
CSCP_FEAT_ALL_SUPP	0x0007	All features are supported

Table 2 – CSC Feature bit flags

Name	Value	Description
CSCP_LOC_OTHER	0	Other
CSCP_LOC_FRONT_WHEEL	4	Front Wheel
CSCP_LOC_LEFT_CRANK	5	Left Crank
CSCP_LOC_RIGHT_CRANK	6	Right Crank
CSCP_LOC_LEFT_PEDAL	7	Left Pedal
CSCP_LOC_RIGHT_PEDAL	8	Right Pedal
CSCP_LOC_REAR_DROPOUT	9	Rear dropout
CSCP_LOC_CHAINSTAY	10	Chainstay
CSCP_LOC_FRONT_HUB	11	Front Hub
CSCP_LOC_REAR_WHEEL	12	Rear wheel
CSCP_LOC_REAR_HUB	13	Rear hub

Table 3 – Sensor Location Keys

Name	Value	Description
CSCP_CTNL_PT_OP_SET_CUMUL_VAL	1	Set Cumulative Value
CSCP_CTNL_PT_OP_START_CALIB	2	Start Sensor Calibration
CSCP_CTNL_PT_OP_UPD_LOC	3	Update Sensor Location
CSCP_CTNL_PT_OP_REQ_SUPP_LOC	4	Request Supported Sensor Locations
CSCP_CTNL_PT_RSP_CODE	16	Response Code

Table 4 – SC Control Point Operation Code Keys

Name	Value	Description
CSCP_CTNL_PT_RESP_SUCCESS	1	Success
CSCP_CTNL_PT_RESP_NOT_SUPP	2	Operation Code Not Supported
CSCP_CTNL_PT_RESP_INV_PARAM	3	Invalid Parameter
CSCP_CTNL_PT_RESP_FAILED	4	Operation Failed

Table 5 – SC Control Point Response Value Keys

Type	Parameters	Description
uint8_t	flags	Flags
uint16_t	cumul_crank_rev	Cumulative Crank Revolution
uint16_t	last_crank_evt_time	Last Crank Event Time Unit has a resolution of 1/1024s.
uint16_t	last_wheel_evt_time	Last Wheel Event Time Unit has a resolution of 1/1024s.
uint32_t	cumul_wheel_rev	Cumulative Wheel Revolution

Table 6 – CSC Measurement Structure (struct cscps_ntf_csc_meas_cmd)

6 Abbreviations

Abbreviation	Original Terminology
API	Application Programming Interface
BLE	Bluetooth Low Energy
GAP	Generic Access Profile
GATT	Generic Attribute Profile
CSCP	Cycling Speed and Cadence Profile
CSCPS	Cycling Speed and Cadence Server Role
CSCPC	Cycling Speed and Cadence Client Role
CSCS	Cycling Speed and Cadence Service
MSC	Message Sequence Chart
RW	RivieraWaves

7 References

[1]	**Title**	CYCLING SPEED AND CADENCE PROFILE SPECIFICATION		
	Reference	CSCP_SPEC_V10		
	Version	V10r00	**Date**	2012-08-07
	Source	Bluetooth SIG		

[2]	**Title**	CYCLING SPEED AND CADENCE PROFILE TEST SPECIFICATION		
	Reference	CSCP.TS.1.0.0		
	Version	1.0.0	**Date**	2012-08-07
	Source	Bluetooth SIG		

[3]	**Title**	CYCLING SPEED AND CADENCE SERVICE SPECIFICATION		
	Reference	CSCS_SPEC_V10		
	Version	V10r00	**Date**	2012-08-07
	Source	Bluetooth SIG		

[4]	**Title**	CYCLING SPEED AND CADENCE SERVICE TEST SPECIFICATION		
	Reference	CSCS.TS.1.0.0		
	Version	1.0.0	**Date**	2012-08-07
	Source	Bluetooth SIG		