RW BLE Running Speed and Cadence Profile Interface Specification

Interface Specification

RW-BLE-RSCP-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
1.1	Oct 13th 2014	Updated for BLE 4.1	CM
7.0	Dec 1st 2014	Naming and version alignment	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) Running Speed and Cadence Profile (RSCP) 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 Running Speed and Cadence Profile Overview

The RSCP enables a collector device to connect and interact with a Running 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 (RSCPS) and Collector role (RSCPC). 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 Running 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 Running Speed and Cadence service is defined in the table below:

Characteristic Name	Requirements	Properties	Security	Descriptors
RSC Measurement	Mandatory	Notify	None	Client Characteristic Configuration
RSC 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

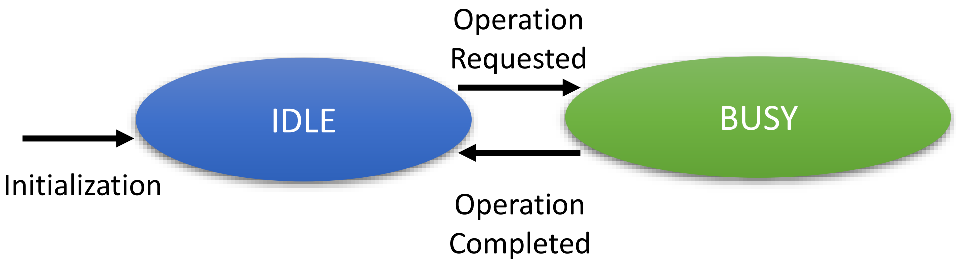
2 RSCP Sensor Role API

2.1 Environment

This role should be activated in every application that a Running Speed and Cadence Sensor is required; the provided API is capable of sending notifications such as cadence and several measurements to the collector. This FW will behave as configured by the Collector in the RSCP characteristic. Please, refer to "rscps_task.h" for implementation of this API.

Within the RSCPS 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/Database creation

During the initialization phase of the Running 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	rsc_feature	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 Running 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 supported in the rsc_feature parameter, the characteristic will be added.
uint8_t	sensor_loc	Sensor location, used if the Sensor Location characteristic is added in the database. (Table 3)

Description: This API message shall be used to add one instance of the Running 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:

- Total Distance Measurement
- Calibration Procedure
- Multiple Sensor Locations

2.2.2 RSCPS_ENABLE_REQ

Source: TASK_APP

Destination: TASK_RSCPS

Required State: IDLE

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint16_t	rsc_meas_ntf_cfg	RSC 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: RSCPS_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 RSCPS_ENABLE_RSP

Source: TASK_RSCPS

Destination: TASK_APP

Parameters:

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

Description: This API message informs the application about the enabling operation.

2.2.4 RSCPS_NTF_RSC_MEAS_REQ

Source: TASK_APP

Destination: TASK_RSCPS

Required State: CONNECTED

Parameters:

Type	Parameters	Description
uint8_t	flags	Flags. Indicate which parameters are included in the value (see Table 1)
uint8_t	inst_cad	Instantaneous Cadence Unit is in 1/minute with a resolution of 1/min.
uint8_t	inst_speed	Instantaneous Speed Unit is in m/s with a resolution of 1/256 s.
uint8_t	inst_stride_len	Instantaneous Stride Length Unit is in meter with a resolution of 1/100 m (or centimeter).

uint8_t	total_dist	Total Distance Unit is in meter with a resolution of 1/10 m (or decimeter).

Response: RSCPS_NTF_RSC_MEAS_RSP

Description: This API message shall be used by the application to send a RSC Measurement notification to every connected device.

The profile checks whether the peer device has enable sending of notifications for the characteristic.

2.2.5 RSCPS_NTF_RSC_MEAS_RSP

Source: TASK_RSCPS

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	status	Status of the operation

Description: This API message informs the application about the enabling operation.

2.2.6 RSCPS_SC_CTNL_PT_REQ_IND

Source: TASK_RSCPS

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 RSCPS_SC_CTNL_PT_CFM message.

2.2.7 RSCPS_SC_CTNL_PT_CFM

Source: TASK_RSCPS

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint8_t	status	Status
union	value	
uint32_t	cumul_value	Cumulative Value (Total Distance)
uint8_t	sensor_location	Sensor Location

Description: This message is sent by the application as a response to the RSCPS_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 RSCPS_CFG_NTFIND_IND

Source: TASK_RSCPS

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint8_t	char_code	Characteristic Code (RSC 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 RSC Measurement characteristic or the SC Control Point characteristic.

2.2.9 RSCPS_CMP_EVT

Source: TASK_RSCPS

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	conidx	Connection index
uint8_t	operation	Operation Code: • RSCPS_SEND_RSC_MEAS_OP_CODE • RSCPS_CTNL_PT_CUMUL_VAL_OP_CODE • RSCPS_CTNL_PT_START_CAL_OP_CODE • RSCPS_CTNL_PT_UPD_LOC_OP_CODE • RSCPS_CTNL_PT_SUPP_LOC_OP_CODE • RSCPS_CTNL_ERR_IND_OP_CODE
uint8_t	status	Status

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

3 RSCP Collector Role API

3.1 Environment

Within the RSCPC task, three 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_RSCPC 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 RSCPC_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_RSCPC_IDX will be used in the rest of the document to refer to any instance of the Running Speed and Cadence profile Collector Role Task. The term TASK_RSCPC will refer to the first instance of this task.

3.2 API Messages

3.2.1 Initialization

During the initialization phase of the Location and Navigation 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 RSCPC_ENABLE_REQ

Source: TASK_APP

Destination: TASK_RSCPC

Required State: IDLE

Parameters:

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

Response: RSCPC_ENABLE_RSP

Description: This API message is used for enabling the Collector role of the RSCP. This Application message contains BLE Connection index, the connection type and the previously saved discovered RSCS 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 those 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 RSCS content in the environment and registering RSCPC in GATT to receive the notifications for the known attribute handles in RSCS that would be notified.

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

Description: This API message is used by the RSCP Collector role to inform the application of a correct disable after a disconnection.

3.2.3 RSCPC_ENABLE_RSP

Source: TASK_RSCPC

Destination: TASK_APP

Parameters:

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

Description: This API message informs the application about the status of the operation.

3.2.4 RSCPC_READ_CMD

Source: TASK_APP

Destination: TASK_RSCPC_IDX

Parameters:

Type	Parameters	Description
uint8_t	operation	Operation Code will be set by the profile task.
uint8_t	read_code	Read Code: • RSCPC_RD_RSC_FEAT • RSCPC_RD_SENSOR_LOC • RSCPC_RD_WR_RSC_MEAS_CFG • RSCPC_RD_WR_SC_CTNL_PT_CFG

Response: RSCPC_VALUE_IND and RSCPC_CMP_EVT

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

3.2.5 RSCPC_CFG_NTFIND_CMD

Source: TASK_APP

Destination: TASK_RSCPC_IDX

Required State: IDLE

Parameters:

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

Response: RSCPC_CMP_EVT

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

3.2.6 RSCPC_CTNL_PT_CFG_REQ

Source: TASK_RSCPC_IDX

Destination: TASK_APP

Parameters:

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

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 RSCPC_CMP_EVT message is sent back to the requester with a PRF_ERR_INVALID_HDL error status.

3.2.7 RSCPC_CTNL_PT_CFG_RSP

Source: TASK_RSCPC_IDX

Destination: TASK_APP

Parameters:

Type	Parameters	Description
struct rscp_sc_ctnl_pt_rsp	ctnl_pt_rsp	SC Control Point Response

Description: This API message informs the application about the status of the control point.

3.2.8 RSCPC_VALUE_IND

Source: TASK_RSCPC_IDX

Destination: TASK_APP

Parameters:

Type	Parameters	Description
uint8_t	att_code	Attribute Code • RSCPC_NTF_RSC_MEAS • RSCPC_RD_RSC_FEAT • RSCPC_RD_SENSOR_LOC • RSCPC_IND_SC_CTNL_PT • RSCPC_RD_WR_RSC_MEAS_CFG • RSCPC_RD_WR_SC_CTNL_PT_CFG
union	value	
struct rscp_rsc_meas	rsc_meas	RSC Measurement
uint16_t	sensor_feat	RSC Feature
uint8_t	sensor_loc	Sensor Location
struct rscp_sc_ctnl_pt_rsp	ctnl_pt_rsp	SC Control Point Response
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 RSCPC_CMP_EVT

Source: TASK_RSCPC_IDX

Destination: TASK_APP

Parameters:

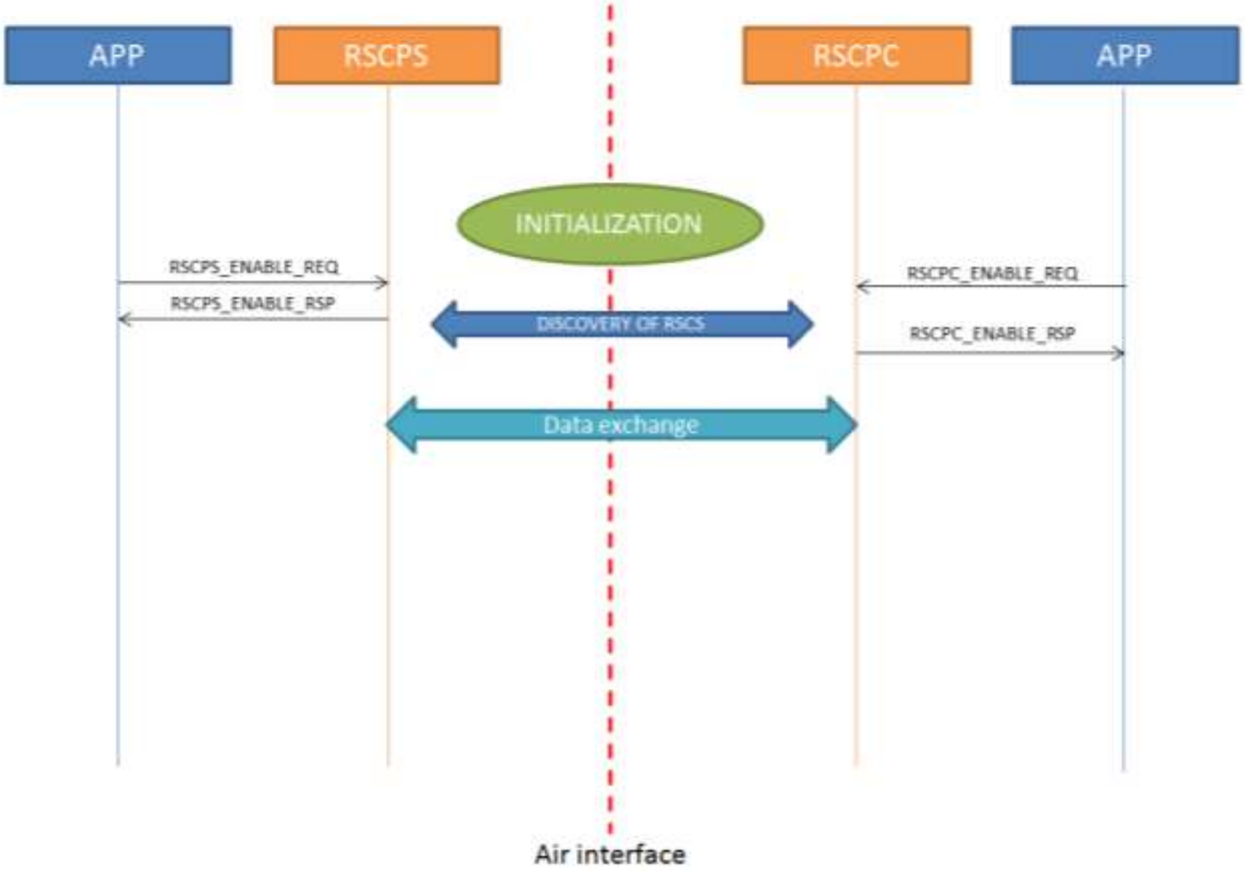
Type	Parameters	Description
uint8_t	operation	Operation Code: • RSCPC_ENABLE_OP_CODE • RSCPC_READ_OP_CODE • RSCPC_CFG_NTF_IND_OP_CODE • RSCPC_CTNL_PT_CFG_WR_OP_CODE • RSCPC_CTNL_PT_CFG_IND_OP_CODE
uint8_t	status	Status

Description: The API message is used by the RSCPC 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)

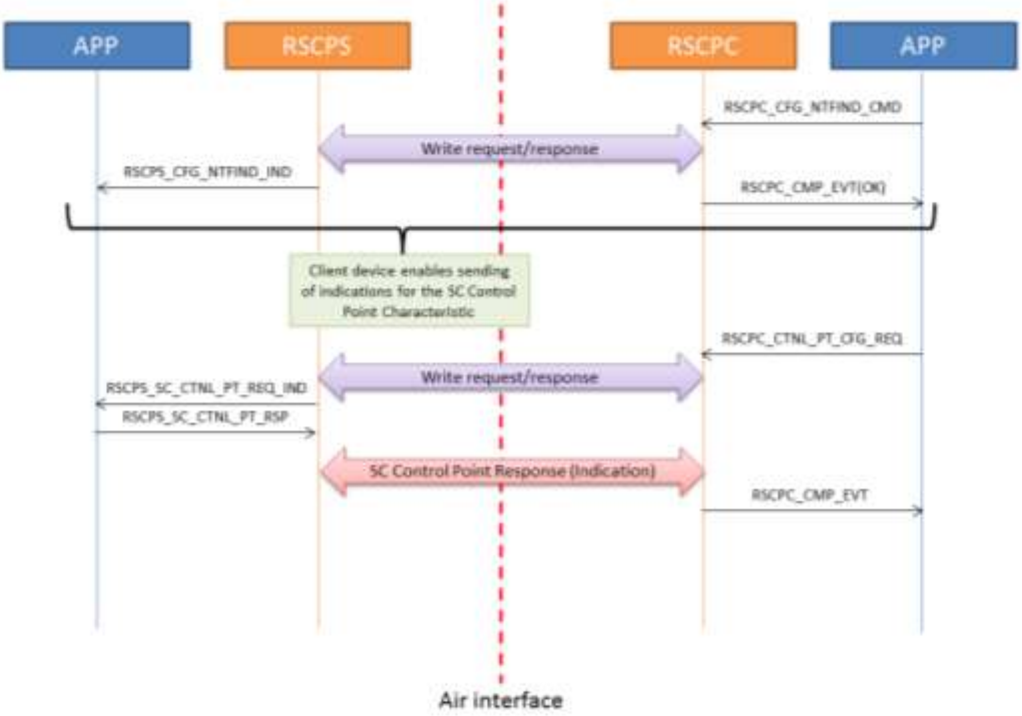
This part describes the different procedure that can be used within the Running Speed and Cadence profile. In these MSCs, it is supposed that two RW stacks (one with the server role of the profile and one with the client role) are connected together.

4.1 Device Initialization / Connection / Disconnection



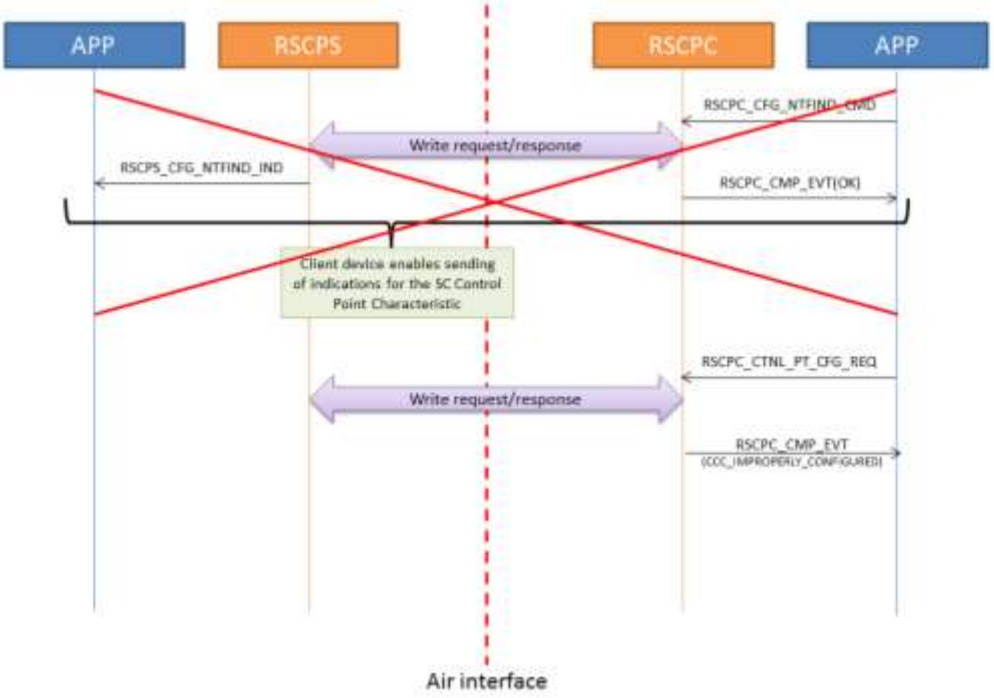
4.2 SC Control Point Characteristic usage

4.2.1 Normal procedure



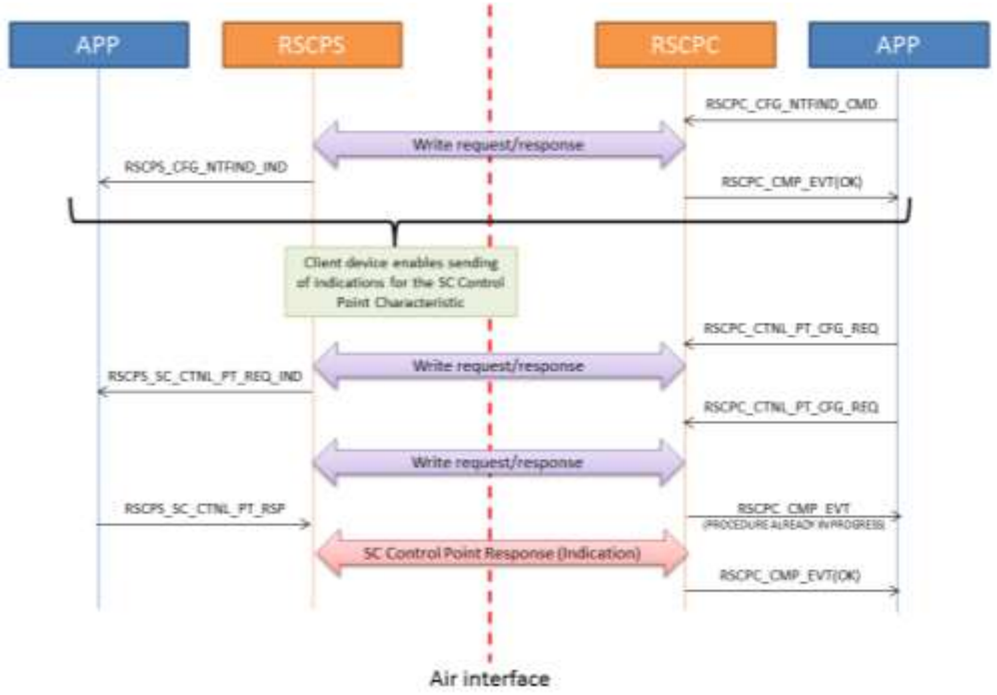
4.2.2 CCC improperly configured Error

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



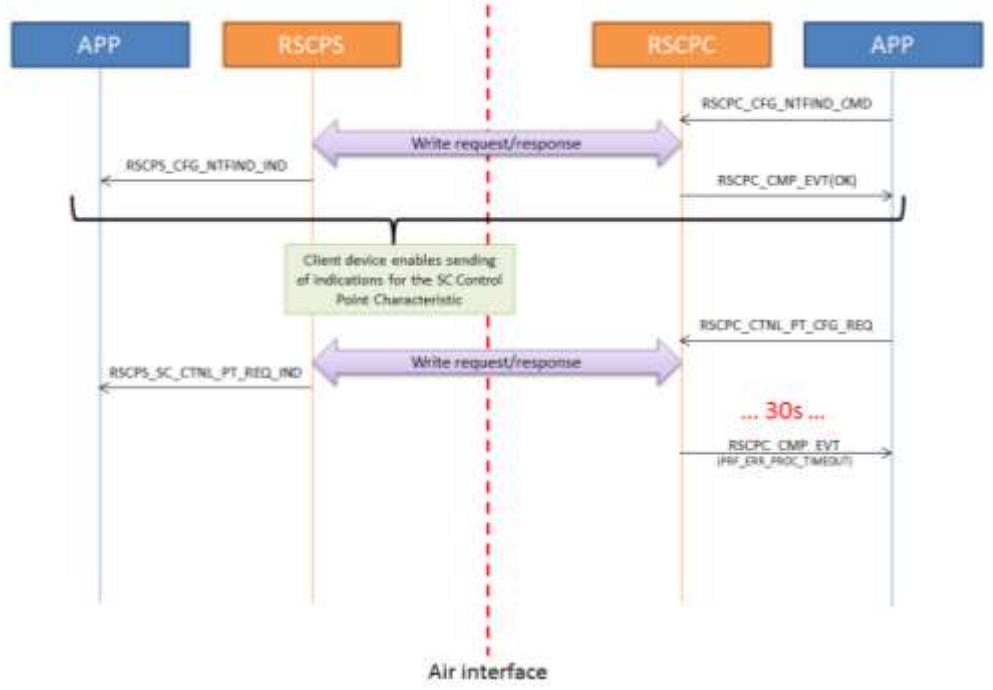
4.2.3 Procedure Already in Progress Error

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.



5 Miscellaneous

Name	Value	Description
RSCP_MEAS_INST_STRIDE_LEN_PRESENT	0x01	Instantaneous Stride Length Present
RSCP_MEAS_TOTAL_DST_MEAS_PRESENT	0x02	Total Distance Present
RSCP_MEAS_WALK_RUN_STATUS	0x04	Walking or Running
RSCP_MEAS_ALL_PRESENT	0x07	All parameters are present

Table 1 – RSC Measurement Present Parameters bit flags

Name	Value	Description
RSCP_FEAT_INST_STRIDE_LEN_SUPP	0x0001	Instantaneous Stride Length Measurement Supported
RSCP_FEAT_TOTAL_DST_MEAS_SUPP	0x0002	Total Distance Measurement Supported
RSCP_FEAT_WALK_RUN_STATUS_SUPP	0x0004	Walking or Running Status Supported
RSCP_FEAT_CALIB_PROC_SUPP	0x0010	Calibration Procedure Supported
RSCP_FEAT_MULT_SENSOR_LOC_SUPP	0x0020	Multiple Sensor Locations Supported
RSCP_FEAT_ALL_SUPP	0x003F	All features are supported

Table 2 – RSC Feature bit flags

Name	Value	Description
RSCP_LOC_OTHER	0	Other
RSCP_LOC_TOP_SHOE	1	Top of shoe
RSCP_LOC_IN_SHOE	2	In shoe
RSCP_LOC_HIP	3	Hip
RSCP_LOC_CHEST	14	Chest

Table 3 – Sensor Location Keys

Name	Value	Description
RSCP_CTNL_PT_OP_SET_CUMUL_VAL	1	Set Cumulative Value
RSCP_CTNL_PT_OP_START_CALIB	2	Start Sensor Calibration
RSCP_CTNL_PT_OP_UPD_LOC	3	Update Sensor Location
RSCP_CTNL_PT_OP_REQ_SUPP_LOC	4	Request Supported Sensor Locations
RSCP_CTNL_PT_RSP_CODE	16	Response Code

Table 4 – SC Control Point Operation Code Keys

Name	Value	Description
RSCP_CTNL_PT_RESP_SUCCESS	1	Success
RSCP_CTNL_PT_RESP_NOT_SUPP	2	Operation Code Not Supported
RSCP_CTNL_PT_RESP_INV_PARAM	3	Invalid Parameter
RSCP_CTNL_PT_RESP_FAILED	4	Operation Failed

Table 5 – SC Control Point Response Value Keys

6 Abbreviations

Abbreviation	Original Terminology
API	Application Programming Interface
BLE	Bluetooth Low Energy
GAP	Generic Access Profile
GATT	Generic Attribute Profile
MSC	Message Sequence Chart
RSCP	Running Speed and Cadence Profile
RSCPS	Running Speed and Cadence Server Role
RSCPC	Running Speed and Cadence Client Role
RSCS	Running Speed and Cadence Service
RW	RivieraWaves

7 References

[1]	**Title**	RUNNING SPEED AND CADENCE PROFILE SPECIFICATION		
	Reference	RSCP_SPEC_V10		
	Version	V10r00	**Date**	2012-08-07
	Source	Bluetooth SIG		

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

[3]	**Title**	RUNNING SPEED AND CADENCE SERVICE SPECIFICATION		
	Reference	RSCS_SPEC_V10		
	Version	V10r00	**Date**	2012-08-07
	Source	Bluetooth SIG		

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