Device Information Service (DIS)

Application Programming Interface Reference Manual

Profile Version: 1.0

Release: 4.0.1
May 13, 2014

Louisville, KY www.stonestreetone.com

Table of Contents

1. Introduction

Bluetopia®+LE is Stonestreet One's Bluetooth protocol stack that supports the adopted Bluetooth low energy specification. Stonestreet One's upper level protocol stack that supports Single Mode devices is Bluetopia®+LE Single. More specifically, this stack is a software solution that resides above the Physical HCI (Host Controller Interface) Transport Layer and extends through the L2CAP (Logical Link Control and Adaptation Protocol), ATT (Attribute Protocol) Link Layers, the GAP (Generic Attribute Profile) Layer and the Genetic Attribute Protocol (GATT) Layer. In addition to basic functionality of these layers, the Bluetooth Protocol Stack by Stonestreet One provides implementations of the Device Information Service (DIS), and several of the Bluetooth Profiles. Program access to these layers, services, and profiles is handled via Application Programming Interface (API) calls.

The remainder of this chapter has sections on the scope of this document, other documents applicable to this document, and a listing of acronyms and abbreviations. Chapter 2 is the API reference that contains a description of all programming interfaces for the Device Information Service Profile Stack provided by Bluetopia®+LE Single. And, Chapter 3 contains the header file name list for the Device Information Service library.

1.1 Scope

This reference manual provides information on the DIS API. This API is available on the full range of platforms supported by Stonestreet One:

- Windows
- Linux
- Windows Mobile
- QNX
- Windows CE
- Other Embedded OS

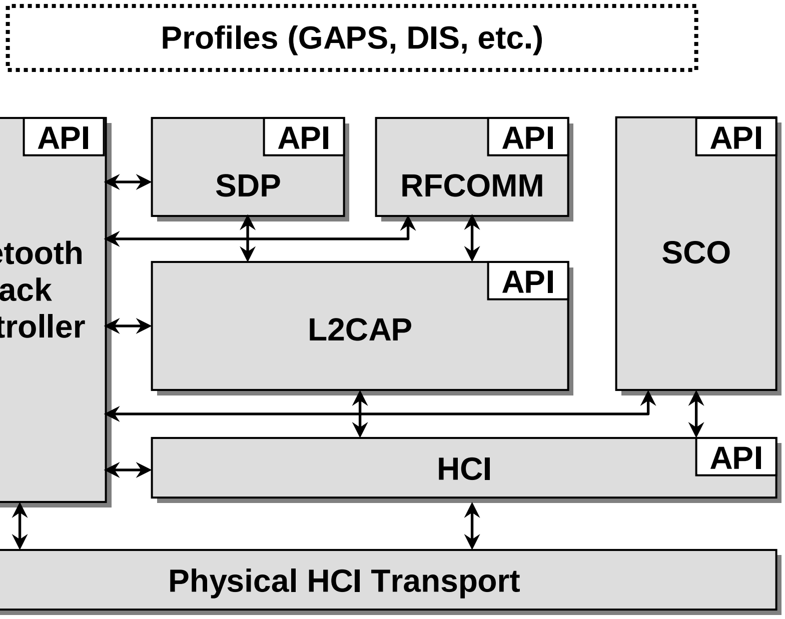

Figure 1-1 The Stonestreet One Bluetooth Protocol Stack

1.2 Applicable Documents

The following documents may be used for additional background and technical depth regarding the Bluetooth technology.

1. *Specification of the Bluetooth System, Volume 1, Architecture and Terminology Overview*, version 4.0, June 30, 2010.
2. *Specification of the Bluetooth System, Volume 6, Core System Package [Low Energy Controller Volume]*, version 4.0, June 30, 2010.
3. *Bluetopia® Protocol Stack, Application Programming Interface Reference Manual*, version 4.0.1, May 13, 2014.
4. *Bluetooth Device Information Service Specification*, version v10r00, April 3, 2012.

Possible error returns are listed for each API function call. These are the *most likely* errors, but in fact programmers should allow for the possibility of any error listed in the BTErrors.h header file to occur as the value of a function return.

1.3 Acronyms and Abbreviations

Acronyms and abbreviations used in this document and other Bluetooth specifications are listed in the table below.

Term	Meaning
API	Application Programming Interface
ATT	Attribute Protocol
BD_ADDR	Bluetooth Device Address
BT	Bluetooth
DIS	Device Information Service
GAPS	Generic Access Profile Service
GATT	Generic Attribute Protocol
HCI	Host Controller Interface
HS	High Speed
L2CAP	Logical Link Control and Adaptation Protocol
LE	Low Energy
LSB	Least Significant Bit
MSB	Most Significant Bit

2. DIS Programming Interface

The Device Information Service, DIS, programming interface defines the protocols and procedures to be used to implement DIS capabilities for both Server and Client services. The DIS commands are listed in section 2.1, the event callback prototypes are described in section 2.2, the DIS events are itemized in section 2.3. The actual prototypes and constants outlines in this section can be found in the **DISAPI.h** header file in the Bluetopia distribution.

2.1 Device Information Service Commands

The available DIS command functions are listed in the table below and are described in the text that follows.

Server Commands	
Function	**Description**
DIS_Initialize_Service	Opens a DIS Server.
DIS_Cleanup_Service	Closes an opened DIS Server.
DIS_Set_Manufacturer_Name	Sets the Manufacturer Name characteristic on the specified Device Information Service instance.
DIS_Query_Manufacturer_Name	Queries the current Manufacturer Name characteristic value on the specified DIS instance.
DIS_Set_Model_Number	Sets the Model Number characteristic on the specified Device Information Service instance.
DIS_Query_Model_Number	Queries the current Model Number characteristic value on the specified DIS instance.
DIS_Set_Serial_Number	Sets the Serial Number characteristic on the specifided Device Inofrmation Service instance.
DIS_Query_Serial_Number	Queries the current Serial Number characteristic value on the specifided DIS instance.
DIS_Set_Hardware_Revision	Sets the Hardware Revision characteristic on the specified Device Information Service instance.
DIS_Query_Hardware_Revision	Queries the current Hardware Revision characteristic value on the specified

	Device Information Service instance.
DIS_Set_Firmware_Revision	Sets the Firmware revision characteristic on the specified Device Information Service instance.
DIS_Query_Firmware_Revision	Queries the current Firmware Revision characteristic value on the specified DIS instance.
DIS_Set_Software_Revision	Sets the Software Revision characteristic on the specified Device Information Service instance.
DIS_Query_Software_Revision	Queries the current Software Revision characteristic value on the specified DIS instance.
DIS_Set_System_ID	Sets the System ID characteristic on the specified Device Inforamtion Service instance.
DIS_Query_System_ID	Queries the current System ID characteristic value on the specified DIS instance.
DIS_Set_IEEE_Certification_Data	Sets the IEEE Certification Data characteristic on the specified Device Information Service instance.
DIS_Query_IEEE_Certification_Data	Queries the current IEEE Certification Data characteristic value on the specified DIS instance.
DIS_Set_PNP_ID	Sets the Plug and Play (PnP) ID characteristic on the specified Device Information Service instance.
DIS_Query_PNP_ID	Queries the current PnP ID Data characteristic value on the specified DIS instance.
DIS_Decode_PNP_ID	Parses a PnP ID value received from a remote DIS Server.

DIS_Initialize_Service

This function opens a DIS Server on a specified Bluetooth Stack.

Notes:

Only one DIS Server, per Bluetooth Stack ID, may be open at a time.

Prototype:

int BTPSAPI **DIS_Initialize_Service**(unsigned int BluetoothStackID,
unsigned int *ServiceID);

Parameters:

BluetoothStackID[1] Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.

ServiceID Unique GATT Service ID of the registered DIS service returned from GATT_Register_Service API.

Return:

Positive non-zero if successful. The return value will be the Service ID of DIS Server that was successfully opened on the specified Bluetooth Stack ID. This is the value that should be used in all subsequent function calls that require Instance ID.

Negative if an error occurred. Possible values are:

DIS_ERROR_INSUFFICIENT_RESOURCES
DIS_ERROR_SERVICE_ALREADY_REGISTERED
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_INVALID_SERVICE_TABLE_FORMAT
BTGATT_ERROR_INSUFFICIENT_RESOURCES
BTGATT_ERROR_INVALID_PARAMETER
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_NOT_INITIALIZED

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Cleanup_Service

This function is responsible for cleaning up and freeing all resources associated with a Device Information Service Instance. After this function is called, no other Device Information Service function can be called until after a successful call to the DIS_Initialize_Service() function is performed.

Prototype:

int BTPSAPI **DIS_Cleanup_Service**(unsigned int BluetoothStackID,
unsigned int InstanceID);

Parameters:

BluetoothStackID[1] Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.

InstanceID The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_INVALID_PARAMETER

DIS_ERROR_INVALID_INSTANCE_ID

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Set_ Manufacturer_Name

This function is responsible for setting the Manufacturer Name characteristic on the specified Device Information Service instance.

Prototype:

int BTPSAPI **DIS_Set_Manufacturer_Name**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *ManufacturerName);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
ManufacturerName	The value to be entered as the Manufacturer Name for the specified DIS Instance..

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_INVALID_INSTANCE_ID

DIS_ERROR_INVALID_PARAMETER

BTGATT_ERROR_NOT_INITIALIZED

BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID

BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Query_ Manufacturer_Name

This function is responsible for querying the current Manufacturer Name characteristic value on the specified DIS instance.

Prototype:

int BTPSAPI **DIS_Query_Manufacturer_Name**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *ManufacturerName) ;

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
ManufacturerName	A pointer to return the current Manufacturer Name for the specified DIS Instance.

Return:

Zero if successful.

An error code if negative; one of the following values:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Set_ Model_Number

This function is responsible for setting the Model Number characteristic on the specified Device Information Service instance.

Prototype:

int BTPSAPI **DIS_Set_Model_Number**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *ModelNumber) ;

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().

ModelNumber	The value to be entered as the Model Number for the specified DIS Instance.

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Query_ Model_Number

This function is responsible for querying the current Model Number characteristic value on the specified DIS instance.

Prototype:

int BTPSAPI **DIS_Query_Model_Number**(unsigned int BluetoothStackID,
    unsigned int InstanceID, char *ModelNumber);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
ModelNumber	A pointer to return the current Model Number for the specified DIS Instance.

Return:

Zero if successful.

An error code if negative; one of the following values:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Set_ Serial_Number

This function is responsible for setting the Serial Number characteristic on the specified Device Information Service instance.

Prototype:

int BTPSAPI **DIS_Set_Serial_Number**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *SerialNumber);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
SerialNumber	The value to be entered as the Serial Number for the specified DIS Instance.

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Query_ Serial_Number

This function is responsible for querying the current Serial Number characteristic value on the specified DIS instance.

Prototype:

int BTPSAPI DIS_Query_Serial_Number(unsigned int BluetoothStackID, unsigned int
InstanceID, char *SerialNumber);

Parameters:

BluetoothStackID[1] Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.

InstanceID The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().

SerialNumber A pointer to return the current Serial Number for the specified DIS Instance.

Return:

Zero if successful.

An error code if negative; one of the following values:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Set_ Hardware_Revision

This function is responsible for setting the Hardware Revision characteristic on the specified Device Information Service instance.

Prototype:

int BTPSAPI **DIS_Set_Hardware_Revision**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *Hardware_Revision);

Parameters:

BluetoothStackID[1] Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.

InstanceID The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().

Hardware_Revision The value to be entered as the Hardware Revision for the specified DIS Instance.

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_INVALID_INSTANCE_ID

DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Query_ Hardware_Revision

This function is responsible for querying the current Hardware Revision characteristic value on the specified DIS instance.

Prototype:

int BTPSAPI **DIS_Query_Hardware_Revision**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *Hardware_Revision);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
Hardware_Revision	A pointer to return the current Hardware Revision for the specified DIS Instance.

Return:

Zero if successful.

An error code if negative; one of the following values:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Set_ Firmware_Revision

This function is responsible for setting the Firmware Revision characteristic on the specified Device Information Service instance.

Prototype:

int BTPSAPI **DIS_Set_Firmware_Revision**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *FirmwareRevision);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
FirmwareRevision	The value to be entered as the Firmware Revision for the specified DIS Instance.

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Query_ Firmware_Revision

This function is responsible for querying the current Firmware Revision characteristic value on the specified DIS instance.

Prototype:

int BTPSAPI **DIS_Query_Firmware_Revision**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *FirmwareRevision);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
FirmwareRevision	A pointer to return the current Firmware Revision for the specified DIS Instance.

Return:

Zero if successful.

An error code if negative; one of the following values:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Set_ Software_Revision

This function is responsible for setting the Software Revison characteristic on the specified Device Information Service instance.

Prototype:

int BTPSAPI **DIS_Set_Software_Revision**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *SoftwareRevision);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
SoftwareRevision	The value to be entered as the Software Revision for the specified DIS Instance.

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Query_ Software_Revision

This function is responsible for querying the current Software Revision characteristic value on the specified DIS instance.

Prototype:

int BTPSAPI **DIS_Query_Software_Revision**(unsigned int BluetoothStackID,
unsigned int InstanceID, char *SoftwareRevision);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
SoftwareRevision	A pointer to return the current Software Revision for the specified DIS Instance.

Return:

Zero if successful.

An error code if negative; one of the following values:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Set_ System_ID

This function is responsible for setting the System ID characteristic on the specified Device Information Service instance.

Prototype:

int BTPSAPI **DIS_Set_System_ID**(unsigned int BluetoothStackID,
unsigned int InstanceID, DIS_System_ID_Value_t *SystemID);

Parameters:

BluetoothStackID[1] Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.

InstanceID The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().

SystemID The value to be entered as the System ID for the specified DIS Instance.

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Query_ System_ID

This function is responsible for querying the current System ID characteristic value on the specified DIS instance.

Prototype:

int BTPSAPI **DIS_Query_System_ID**(unsigned int BluetoothStackID,
  unsigned int InstanceID, DIS_System_ID_Value_t *SystemID) ;

Parameters:

BluetoothStackID[1] Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.

InstanceID The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().

SystemID A pointer to return the current System ID for the specified DIS Instance.

Return:

Zero if successful.

An error code if negative; one of the following values:

DIS_ERROR_INVALID_INSTANCE_ID

DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Set_ IEEE_Certification_Data

This function is responsible for setting the IEEE Certificastion Data characteristic on the specified Device Information Service instance.

Prototype:

int BTPSAPI **DIS_Set_IEEE_Certification_Data**(unsigned int BluetoothStackID,
unsigned int InstanceID, unsigned int Length, Byte_t *IEEE_Certification_Data);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
IEEE_Certification_Data	The value to be entered as the IEEE Certification data for the specified DIS Instance.

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Query_ IEEE_Certification_Data

This function is responsible for querying the current IEEE Certification Data characterisitv value on the specified DIS instance.

Prototype:

int BTPSAPI **DIS_Query_IEEE_Certification_Data**(unsigned int BluetoothStackID, unsigned int InstanceID, Byte_t *IEEE_Certification_Data) ;

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
IEEE_Certification_Data	A pointer to return the current IEEE Certification for the specified DIS Instance.

Return:

Zero if successful.

An error code if negative; one of the following values:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Set_ PNP_ID

This function is responsible for setting the Plue and Play (PnP) ID characteristic on the specified Device Information Service instance.

Prototype:

int BTPSAPI **DIS_Set_PNP_ID**(unsigned int BluetoothStackID, unsigned int InstanceID, DIS_PNP_ID_Data_t *PNP_ID_Data);

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
PNP_ID_Data	The value to be entered as the PNP ID Data for the specified DIS Instance. The PnP ID Data structure is as follows:

```
typedef struct
{
     Byte_t     VendorID_Source;
     Word_t     VendorID;
     Word_t     ProductID;
     Word_t     ProductVersion;
} DIS_PNP_ID_Data_t;
```

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

```
DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER
```

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Query_ PNP_ID

This function is responsible for querying the current PnP ID Data charctertisic value on the specified DIS instance.

Prototype:

```
int BTPSAPI DIS_Query_PNP_ID(unsigned int BluetoothStackID,
     unsigned int InstanceID, DIS_PNP_ID_Data_t *PNP_ID_Data) ;
```

Parameters:

BluetoothStackID[1]	Unique identifier assigned to this Bluetooth Protocol Stack via a call to BSC_Initialize.
InstanceID	The Service Instance ID to close. This InstanceID was returned from the DIS_Initialize_Service().
PNP_ID_Data	A pointer to return the current PNP ID Data for the specified DIS Instance. The PnP ID Data structure is as follows:

```
typedef struct
{
     Byte_t     VendorID_Source;
     Word_t     VendorID;
     Word_t     ProductID;
     Word_t     ProductVersion;
} DIS_PNP_ID_Data_t;
```

Return:

Zero if successful.

An error code if negative; one of the following values:

DIS_ERROR_INVALID_INSTANCE_ID
DIS_ERROR_INVALID_PARAMETER
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

Notes:

1. The BluetoothStackID parameter is not included in versions of Bluetopia that have been optimized to only control a single Bluetooth device, such as some embedded versions of Bluetopia. Please refer to the appropriate header file to determine if this parameter is part of the function call or not.

DIS_Decode_ PNP_ID

The following function is responsible for parsing a PnP ID value received from a remote DIS server.

Prototype:

int BTPSAPI **DIS_Decode_PNP_ID**(unsigned int ValueLength, Byte_t *Value,
DIS_PNP_ID_Data_t *PNP_ID_Data);

Parameters:

ValueLength
Specifies the length of the Device Information value returned by the remote DIS Server.

Value
Value is a pointer to the Device Information data returned by the remote DIS Server.

PNP_ID_Data
A pointer to store the parsed PNP ID Data for the specified DIS Instance. The PnP ID Data structure is as follows:

```
typedef struct
{
   Byte_t     VendorID_Source;
   Word_t     VendorID;
   Word_t     ProductID;
   Word_t     ProductVersion;
} DIS_PNP_ID_Data_t;
```

Return:

Zero if successful.

Negative if an error occurred. Possible values are:

DIS_ERROR_MALFORMATTED_DATA
BTGATT_ERROR_NOT_INITIALIZED
BTGATT_ERROR_INVALID_BLUETOOTH_STACK_ID
BTGATT_ERROR_INVALID_PARAMETER

3. File Distributions

The header files that are distributed with the Bluetooth Device Information Service Library are listed in the table below

File	Contents/Description
DISAPI.h	Bluetooth Device Information Service (GATT based) API Type Definitions, Constants, and Prototypes.
DISTypes.h	Bluetooth Device Information Service Types.
SS1BTDIS.h	Bluetooth Device Information Service Include file