Source Control Drawing

Part Description:	Bluetooth Low Energy Software Interface Application Manual
Customer Part Number:	
Abracon Part Number:	ABBTM-NVC-MDCS71

Customer Approval
(Please return this copy as a certification of your approval)

Approved by:	
Approval Date:	

Abracon Corporation Headquarters:

30332 Esperanza
Rancho Santa Margarita, CA-92688
Ph: (949) 546-8000
Fax: (949) 546-8001

Internal Use Only	
	500245

Revision History							
Revision	ECO	Description	Date	Prep'd By	Ck'd By	Ck'd By	Appr'd By
		Initial Release	7/15/2013	YH	SR	CB	JE
A	2811	Added iBeacon functions	1/15/2014	YH	SR	CB	JE

Table of Contents

Table of Tables

Table of Figures

1. Introduction

The Abracon Control Interface (ACCI) is a set of ASCII commands and indicators with which the user can control the Abracon's Bluetooth Module via UART interface by a host (PC, MCU, etc.).

The commands are used to control the Bluetooth Module sent by host. The indicators (or the response of the commands) are output from the Bluetooth Module to the host to indicate the status of the module.

ACCI for Bluetooth Low Energy (BLE) can be considered as a subset and/or extension of the common ACCI.

This document only describes the interface of the BLE peripheral role.

1.1. Default UART Configuration

The default configuration of UART is given below:

Baud rate: 2400
Data bits: 8
Stop bits: 1
Parity: None
Flow control: None

The configuration of UART can be changed by ACCI and the new configuration is stored and used for further communication.

2. Command and Indicator Syntax

2.1. General Syntax

The general syntax of ACCI command is shown as below:

BC:<CMD>[=Para1][,Para2][,RawData][,…]<CR><LF>

The general syntax of ACCI indicator is shown as below:

<IDC>[=Para1][,Para2][,RawData][,…]<CR><LF>

Description of each field:

BC: is the command line prefix.
CMD is the basic command. All of the commands are listed in section 3.
IDC is the basic indicator. All of the indicators are listed in section 4.
= is the separator between command/indicator and parameter. It's only needed if a parameter is presented.
Para1 is the first parameter. Not all of the commands have a parameter.
, is the separator between parameters. It's only needed if subsequent parameter is presented.
Para2 is the second parameter if available.
RawData is the raw data which will be sent by the command. Only parts of the commands have this field.
<CR><LF> is the terminator of the command line.
Notes:
1. *If a parameter is mandatory, it will be surrounded by { }. If a parameter is optional, it will be surrounded by [ ].*
2. *<CR> means Carriage Return, and <LF> means Line-Feed.*
3. *All of the parameters are composed of ASCII characters while the **RawData** field can composed of any data contents*
4. *In indicators, the module prints hex values in low case. For other places in the document we always use upper case characters for hex values.*

2.2. Examples

Here is some examples show how to use the ACCI commands and indicators.

Ex. 2.1

➔ **BC:NM=A Keyb<CR><LF>**　　　← *configure the GAP device name characteristic. The new name is "A Keyb".*

← **OK<CR><LF>**　　　← *response from the module to indicate the command is adopted.*

Notes:
1. *For the examples in this document, the command sent to the Bluetooth Module will be shown with "➔" at the beginning of the line, while the indicator output by Bluetooth Module will be shown with "←" at the beginning of the line.*
2. *For the examples in this document, the comments will start with a "←" and be written in italic.*
3. *For the examples in this document, only the characters in grey background color are the real content of a command or indicator.*

For a full example using the module with iOS devices, please check 7.1.

3. Command List

All the available ACCI commands are listed and briefly described in the tables below. The detailed description of each command is given in chapter 0.

Table 3.1 ACCI for BLE Command List

Command	Short Description
UD	Query or configure the GATT service **UUID**128 the module
AD	Query the Bluetooth **ad**dress of the module
AP	Query or configure the GAP **Ap**pearance characteristic
FT	Query or configure the **f**ea**t**ures of the Bluetooth Module.
CC	Query or configure the peripheral preferred **c**onnection parameter **c**haracteristic
NM	Query or change the GAP device **n**a**m**e characteristic.
BR	Query or change the UART **b**aud **r**ate.
FD	Configure all setting to **f**actory **d**efault values
MD	**M**ake the Bluetooth Module **d**iscoverable/connectable, in BTLE the peripheral start/stop the advertising after got this command.
DC	**D**is**c**onnect with remote BLE central.
DT	Send **dat**a packet to the connected BLE central.
RT	Query the current **r**adio s**t**atus.
RI	Query the RSSI for the current connection.
PD	**P**ower **d**own the module, put it into dormant or hibernate state, or warm reset it.
CP	**C**lear the **p**aired Bluetooth device list.
BA	Query or set the i**B**eacon m**a**jor number
BI	Query or set the i**B**eacon m**i**nor number
BP	Query or set the i**B**eacon RF **P**ower measured at one meter
BV	Query or set the i**B**econ Advertisement Inter**v**al

4. Indicator List

All the available ACCI indicators are listed and briefly described in the tables below. The detailed description of each indicator is given in chapter 0.

Table 4.1 ACCI for BLE Indicator List

Indicator	Short Description
OK	Indicates a command was adopted by the Bluetooth Module.
ER	Indicates there is an **er**ror detected in the command sent by the host.
UD	Report the GATT service **UUID**128 the module
AD	Report the Bluetooth **ad**dress of the Module
AP	Report the GAP **Ap**pearance characteristic
FT	Reports the configured **f**ea**t**ures of the Bluetooth Module.
CC	Report the peripheral preferred **c**onnection parameter
NM	Reports the device **nam**e of the Bluetooth Module.
BR	Reports the UART **b**aud **r**ate.
DT	Received **d**a**t**a packet from the connected BLE central.
RT	Report the **r**adio s**t**ate.
RI	Report the RSSI for the current connection.
BA	Report the i**B**eacon m**a**jor number
BI	Report the i**B**eacon m**i**nor number
BP	Report the i**B**eacon RF **P**ower measured at one meter
BV	Report the i**B**eacon Advertisement Inter**v**al

5. Description of ASCII Commands

5.1. General Information Commands

5.1.1. UD-GATT Service UUID128

5.1.1.1. Description:

The module works as a BLE peripheral and it implements a proprietary service which can be taken as a virtual SPP channel between the peripheral and connected central. This command can be used to set a unique UUID128 for the service. Once configured, the UUID128 will take effect when the module reset or power up for next time. The Bluetooth module stores the UUID128 in its non-volatile memory so the value won't change until be set again.

5.1.1.2. Syntax:

Synopsis:
BC:UD[=Value Index,Value]<CR><LF>

5.1.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Value Index	The UUID128 has 128bits, which is represented by 8 16-bits words. The 8 words are stored in a lowest word first order. Index 0 is the lowest word of the UUID128. Value: 0000-0007	M	
Value	The value of the word. Should be 4 hex char. The word value should be in a lowest byte first order. Refer to the following example.	O	If value is not given, the current value of the index is output as the response of this command.

5.1.1.4. Examples:

Ex. 5.1. To query the UUID128 of Bluetooth Module:

➔ **BC:UD=0000<CR><LF>**   ⬅ *query the value of word 0 in the UUID128 (lowest word).*

⬅ **UD=0000,7166<CR><LF>**   ⬅ *report the value of the word 0 in the UUID128, it's 0x6671 (in lowest byte*

first order).

Ex. 5.2. To set the UUID128 of the Bluetooth Module to "C5161D82-AAB0-4E55-8D96-C59D816E6671" (an ASCII string represents the UUID128):

➔ **BC:UD=0000,7166<CR><LF>**   ← *set the word 0 in the UUID128 to 0x6671.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

➔ **BC:UD=0001,6e81<CR><LF>**   ← *set the word 1 in the UUID128.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

➔ **BC:UD=0002,9dc5<CR><LF>**   ← *set the word 2 in the UUID128.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

➔ **BC:UD=0003,968d<CR><LF>**   ← *set the word 3 in the UUID128.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

➔ **BC:UD=0004,554e<CR><LF>**   ← *set the word 4 in the UUID128.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

➔ **BC:UD=0005,b0aa<CR><LF>**   ← *set the word 5 in the UUID128.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

➔ **BC:UD=0006,821d<CR><LF>**   ← *set the word 6 in the UUID128.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

➔ **BC:UD=0007,16c5<CR><LF>**   ← *set the word 7 in the UUID128.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

5.1.2. AD - Bluetooth Address

5.1.2.1. Description:

This command can query the Bluetooth address of local module. The response also contains the remote device address if it's bonded. Once the Bluetooth Module adopted this query request, it will report its Bluetooth address by the Indicator AD.

5.1.2.2. Syntax:

Synopsis:
BC:AD<CR><LF>

5.1.2.3. Parameter Description:

None.

5.1.2.4. Examples:

Ex. 5.3. To query the Bluetooth address of local module:

➔ **BC:AD<CR><LF>**   ← *query the Bluetooth address of local module.*

← **AD=00,189600abcd <CR><LF>**   ← *the Bluetooth address is using a public Bluetooth address,*

                                            00:18:96:00:AB:CD.

5.1.3. AP - GAP Appearance Characteristic

5.1.3.1. Description:

This command can query or configure the GAP device appearance character value of Bluetooth Module. Once configured, the configuration will take effect immediately and until the next time the module is configured by this command or the module is rebooted. It means the Bluetooth Module will not remember the configuration, and after the Bluetooth Module has been powered off, the configuration will be lost.

If the parameter is not presented, the Bluetooth Module will report current appearance by the Indicator AP.

The appearance is also used in the advertising. When updated, the value will be used in next time the module start advertising. Standard definitions of appearance could be found in Table 7.1.

5.1.3.2. Syntax:

Synopsis:
BC:AP[=Appearance value]<CR><LF>

5.1.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Appearance value	The GAP appearance characteristic of the Bluetooth Module. Value: 0000h—FFFFh as defined by Bluetooth Assigned Number, refer to Table 7.1 Default: per firmware version	O	

5.1.3.4. Examples:

Ex. 5.4. To query current appearance of Bluetooth Module:

➔ **BC:AP<CR><LF>**   ← *query current appearance value.*

← **AP=03c1<CR><LF>**   ← *report current appearance value: 03C1 as a keyboard.*

Ex. 5.5. To configure the appearance of Bluetooth Module:

➔ **BC:CD=0301<CR><LF>**   ← *configure the module appearance value: 0301.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

5.1.4. FT - Features

5.1.4.1. Description:

This command can query or configure the features of Bluetooth Module. Once configured, the configuration will take effect immediately. The Bluetooth module stores the value in its non-volatile memory so the value won't change until be set again.

If the parameter is not presented, the Bluetooth Module will report current feature configuration by the Indicator FT. If the user wants to configure the features, all of the parameters should be given together.

5.1.4.2. Syntax:

Synopsis:
FT{=Features}<CR><LF>

5.1.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Features	A 16-bit value as a bit mask, for example, 8000 means the bit 15 is 1 and all other bits is 0. The features are defined with the 16 bits mask, 1 is enable. Bit 15: automatically start advertising when power up. Bit 14: reserved, always set to 1. Bit 13: reserved, always set to 1. Bit 12: use privacy, which means the module only connects to bonded devices (need to clear the paring information before connect a new device) Bit 11-1: Not defined yet. Bit 0: Apple's iBeacon feature. Set to 1 and reset the module will put it to be an iBeacon station. Set to 0 will put it back to normal working state.	O	

Notes:

1. *The default feature configuration may be different per software version.*

5.1.4.4. Examples:

Ex. 5.6. To query current feature configuration of Bluetooth Module:

➔ **BC:FT<CR><LF>**   ⬅ *query current feature configuration.*

⬅ **FT=e000<CR><LF>**   ⬅ *report current feature configuration.*

5.1.5. CC - Preferred Connection Parameters Characteristic

5.1.5.1. Description:

This command can query or configure the GAP preferred connection parameter characteristic of Bluetooth Module. Once configured, the configuration will take effect at the next time when a Bluetooth connection is being established. The Bluetooth module stores the value in its non-volatile memory so the value won't change until be set again.

If the parameter is not presented, the Bluetooth Module will report current configuration by the Indicator CC.

5.1.5.2. Syntax:

Synopsis:
BC:CC[=Min Interval,Max Interval, Latency, Supervision Timeout]<CR><LF>

5.1.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Min interval	The minimum value for the connection interval. Hex Value: xxxx (0006 to 0C80, FFFF means not specifically defined) Default depends on firmware	O	
Max interval	The maximum value for the connection interval. Hex Value: xxxx (0006 to 0C80, FFFF means not specifically defined) Default depends on firmware	O	
Latency	The slave latency for the connection in number of connection events. Hex Value: xxxx (0000 to 03E8) Default depends on firmware	O	
Supervision Timeout	The connection supervisor timeout multiplier as a multiple of 10ms. Hex Value: xxxx (0006 to 0C80, FFFF means not specifically defined) Default depends on firmware	O	

5.1.5.4. Examples:

Ex. 5.7. To query current preferred connection parameters of the Bluetooth Module:

➔ **BC:CC<CR><LF>**   ← *query the current preferred connection parameters.*

← **CC=0010,0050,0004,0258<CR><LF>**   ← *report the current preferred connection parameters.*

Ex. 5.8. To set the preferred connection parameters of the Bluetooth Module:

➔ **BC:CC=0010,0050,0004,0258<CR><LF>**   ← *set new connection parameters.*

← **OK<CR><LF>**   ← *response from the module to indicate the command*

         is adopted.

5.1.6. NM - GAP Device Name Character Value

5.1.6.1. Description:

This command can query or change the GAP device name character value of the Bluetooth Module. It takes effect immediately. The Bluetooth Module stores the value in its non-volatile memory so the value won't change until be set again.

If the parameter is not presented, the Bluetooth Module will report current friendly name by the Indicator NM.

5.1.6.2. Syntax:

Synopsis:
BC:NM[=Name]<CR><LF>

5.1.6.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	The new device name of the Bluetooth Module. Length: 1—19 characters Default: Per software version.	O	

5.1.6.4. Examples:

Ex. 5.9. To query current local friendly name of Bluetooth Module:

➔ **BC:NM<CR><LF>**   ← *query current GAP device name character value.*

← **NM=NVC_iGATE<CR><LF>**   ← *report current GAP device name character value, it's "NVC_iGATE".*

Ex. 5.10. To change the local friendly name of Bluetooth Module:

➔ **BC:NM=NVC_BLE<CR><LF>**   ← *change the GAP device name character value to "NVC_BLE"*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

5.1.7. BR—UART Baud Rate

5.1.7.1. Description:

This command can query or change the UART baud rate of Bluetooth Module. Once changed, the new baud rate will take effect immediately. The Bluetooth module stores the value in its non-volatile memory so the value won't change until be set again.

If the parameter is not presented, the Bluetooth Module will report current baud rate by the Indicator BR.

5.1.7.2. Syntax:

Synopsis:
BC:BR[=Baud Rate]<CR><LF>

5.1.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Baud Rate	The new baud rate of the Bluetooth Module. Value: 00h—0Ch 00: 2400 01: 9600 02: 19200 03: 38400 04: 57600 05: 115200 06: 230400 07: 460800 08: 921600 09: 1382400 0A: 1843200 0B: 2764800 0C: 3686400 Default: 00 (2400)	O	The default baud rate may not be 2400 per software version. **There's an advantage using default 2400 baud rate – no need to assert (pull up to high logic level) the WAKE pin to wake up the module from deep sleep before transmitting to it.**

Warning:
1. *Please do NOT try to change to a new baud rate if you don't have a host which can work in that baud rate, for there is no other way to reset it except for UART port.*
2. *If baud rate other than 2400 is used, please keep pulling up the WAKE pin during sending data to the module.*

5.1.7.4. Examples:

Ex. 5.11. To query the baud rate of Bluetooth Module:

➔ **BC:BR\<CR>\<LF>**   ← *query the baud rate.*

← **BR=00\<CR>\<LF>**   ← *report the baud rate, it's 2400.*

Ex. 5.12. To change the baud rate of Bluetooth Module:

➔ **BC:BR=01\<CR>\<LF>**   ← *change the baud rate to 9600.*

←   ← *There's no response for the set command. The new baud rate is effective immediately after receive the command. And the following commands and indicators are in new baud rate.*

5.1.8. FD—Configure All Setting to Factory Default Values

5.1.8.1. Description:

This command can be used to configure all settings to the factory default values.

5.1.8.2. Syntax:

Synopsis:
BC:FD\<CR>\<LF>

5.1.8.3. Parameter Description:

None.

5.1.8.4. Examples:

Ex. 5.13. To reset all settings of the Bluetooth Module:

➔ **BC:FD\<CR>\<LF>**   ← *Configure all settings to factory default values.*

←   ← *There's no response of the command because the UART baud rate is also reset to default.*

5.2. Connection and Data Transfer Commands

5.2.1. MD - Discoverable

5.2.1.1. Description:

This command can start or stop the BLE advertising. The module is discoverable/connectable only when the module is advertising.

5.2.1.2. Syntax:

Synopsis:
BC:MD{=Flag}\<CR>\<LF>

5.2.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Flag	Switch on/off advertising. Value: 00h or 01h 00: Stop advertising. 01: Start advertising.	M	

5.2.1.4. Examples:

Ex. 5.14. To make Bluetooth Module discoverable:

➔ **BC:MD=01<CR><LF>**   ⬅ *make Bluetooth Module discoverable.*

⬅ **OK<CR><LF>**   ⬅ *response from the module to indicate the command is adopted.*

5.2.2. DC—Disconnect with remote BLE Central

5.2.2.1. Description:

This command disconnects the Bluetooth Module from the connected Bluetooth BLE central.

5.2.2.2. Syntax:

Synopsis:
BC:DC<CR><LF>

5.2.2.3. Parameter Description:

None.

5.2.2.4. Examples:

Ex. 5.15. To disconnect with the connected devices:

➔ **BC:DC<CR><LF>**   ⬅ *disconnect with current connected devices.*

⬅ **RT=00<CR><LF>**   ⬅ *the Bluetooth Module is now disconnected.*

5.2.3. DT—Send data packet to BLE central

5.2.3.1. Description:

This command is used to send a data packet to the connected BLE central.

5.2.3.2. Syntax:

Synopsis:
BC:DT{=Data string}<CR><LF>

5.2.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Data String	The data string to be send. **(keep the length of string not more than 20, not including the <CR><LF>)**	M	

Notes:
  1. *The <CR><LF> sequence is treated as the terminator of a command line. If the user wants to send these special characters to remote device, precede them with a backslash character "\". (So send "\\" for "\").*

5.2.3.4. Examples:

Ex. 5.16. To send data:

➔ **BC:DT=1234567890<CR><LF>**   ← *send a data packet.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

5.2.4. RT—Radio Status

5.2.4.1. Description:

This command is used to query the status of BLE radio.

5.2.4.2. Syntax:

Synopsis:
BC:RT<CR><LF>

5.2.4.3. Parameter Description:

None.

5.2.4.4. Examples:

Ex. 5.17. To query the state of BTLE radio:

➔ **BC:RT<CR><LF>**   ← *query the state of BTLE connection.*

← **RT=05,01,00189600abcd<CR><LF>**   ← *the Bluetooth Module is now connecting to the BLE central device*
                                          whose random address is 00:18:96:00:AB:CD.

5.2.5. RI—RSSI

5.2.5.1. Description:

This command is used to query the RSSI (radio signal strength indication) of the current connection.

5.2.5.2. Syntax:

Synopsis:
BC:RI<CR><LF>

5.2.5.3. Parameter Description:

None.

5.2.5.4. Examples:

Ex. 5.18. To query the RSSI of the current connection:

➔ **BC:RI<CR><LF>**   ← *query the RSSI of the current connection.*

← **RI=bb<CR><LF>**   ← *the RSSI of the current connection is -69dBm.*

5.3. Power Mode Commands

5.3.1. PD—Power Down the Module

5.3.1.1. Description:

This command can power down the module into a dormant or hibernate state. In dormant state the module can only be wake up by the pull high the WAKE pin. In hibernate state it will automatically wake up after a given timeout. This command can also be used to warm reset the module.

There's no indicator or response from the UART when the module is power down.

5.3.1.2. Syntax:

Synopsis:
BC:PD[=Mode, Timeout Value]<CR><LF>

5.3.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
State	The power down mode. Value: 00: Dormant, the module power down immediately. 01: Hibernate, the module wake up after the timeout 02: warm reset	M	
Timeout value	7 hex value, the time out unit is 2^20 microseconds, about 1second. So the valid range is 1s to 8.9 years.	Should not be used with Dormant or Warm reset. M for hibernate	

Notes:
  1. When in dormant, the module can only wake up by pull WAKE pin to logic 1 (high logical voltage).

5.3.1.4. Examples:

Ex. 5.19. To put the module into hibernate and make it automatically wake up after 16 seconds:

➔ **BC:PD=01,0000020<CR><LF>**   ← *Put the module into hibernate, and ask it automatically wake up*

                                                          after 32 seconds.

Ex. 5.20. To put the module into dormant state:

➔ **BC:PD=00<CR><LF>**   ← *Put the module into dormant, can only wake it up by WAKE pin.*

Ex. 5.21. To warm reset the module:

➔ **BC:PD=02<CR><LF>**   ← *Reset the Bluetooth Module.*

5.4. Security Commands

5.4.1. CP—Clear the paired Bluetooth device Information

5.4.1.1. Description:

This command can clear the paired device information stored in the Bluetooth Module. All paired devices are stored in the module as a list. This command will clear all the information in the list.

5.4.1.2. Syntax:

Synopsis:
BC:CP<CR><LF>

5.4.1.3. Parameter Description:

None.

5.4.1.4. Examples:

Ex. 5.22. To clear the paired device list:

➔ **BC:CP<CR><LF>**   ⬅ *clear the paired device list.*

⬅ **OK<CR><LF>**   ⬅ *response from the module to indicate the command is adopted.*

5.5. iBeacon Commands

5.5.1. BA—Query or Set the iBeacon Major Number

5.5.1.1. Description:

This command can be used to query or set the iBeacon major number.

5.5.1.2. Syntax:

Synopsis:
BC:BA[=iBeacon Major Number]<CR><LF>

5.5.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
iBeacon Major Number	The new iBeacon major number in four-char hex format. Value: 0000h—FFFFh	O	If no parameter is given, the current number is echoed.

5.5.1.4. Examples:

Ex. 5.23. To query the current iBeacon major number:

➔ **BC:BA<CR><LF>**   ⬅ *Query current iBeacon major number.*

⬅ **BA=ff01<CR><LF>**   ⬅ *The current number is 0xff01.*

5.5.2. BI—Query or Set the iBeacon Minor Number

5.5.2.1. Description:

This command can be used to query or set the iBeacon minor number.

5.5.2.2. Syntax:

Synopsis:
BC:BI[=iBeacon Minor Number]<CR><LF>

5.5.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
iBeacon Minor Number	The new iBeacon major number in four-char hex format. Value: 0000h—FFFFh	O	If no parameter is given, the current number is echoed.

5.5.2.4. Examples:

Ex. 5.24. To set the iBeacon minor number to 0xff01:

➔ **BC:BI=ff01<CR><LF>**   ⬅ *set the iBeacon minor number to 0xff01.*

⬅ **OK<CR><LF>**   ⬅ *the processing is successfully done.*

5.5.3. BP—Query or Set the iBeacon RF Power Level Measured at One Meter

5.5.3.1. Description:

This command can be used to query or set the iBeacon transmission power at 1 meter distance. Normally the power is set by Abracon to indicate the BLE RF power level measured at 1 meter distance from the module. Customer can fine tune it to match with their own environment.

5.5.3.2. Syntax:

Synopsis:
BC:BP[=iBeacon Power Level]<CR><LF>

5.5.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
iBeacon RF power level measured at one meter	The new power level in four-char hex format. The unit is dBm and it is normally a minus number. The default value is 0xffc5 (-59dBm). A valid range should be within -30dBm to -80dBm.	O	If no parameter is given, the current power level is echoed.

5.5.3.4. Examples:

Ex. 5.25. To query the current power level:

➔ **BC:BP<CR><LF>**   ⬅ *Query current iBeacon RF power level measured at one meter.*

⬅ **BP=ffc5<CR><LF>**   ⬅ *The current power level is 0xffc5 (-59dBm).*

	Bluetooth Low Energy Software Interface Application Manual	
Date of Issue: Jan 15th, 2014	**ABBTM-NVC-MDCS71**	**ESD Sensitive**
Page (19) of (33)	**Abracon Drawing # 453732**	Revision #: A

ABRACON CORPORATION — The Power of Linking Together

RoHS Compliant

5.5.4. BV—Query or Set the iBeacon Advertisement Interval

5.5.4.1. Description:

This command can be used to query or set the iBeacon advertisement interval. The default interval is 1 second. Too small interval (so advertisement is sent too frequently) will increase the average power consumption of the BLE module.

5.5.4.2. Syntax:

Synopsis:
BC:BV[=iBeacon Advertisement Interval]<CR><LF>

5.5.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
iBeacon advertisement interval	The interval in four-char hex format. The unit is milliseconds. Value: 0000h—FFFFh	O	If no parameter is given, the current interval is echoed.

5.5.4.4. Examples:

Ex. 5.26. To set the current iBeacon advertisement interval:

➔ **BC:BV=03E8<CR><LF>**   ← *Set current iBeacon advertisement interval to 1second (1000ms).*

← **OK<CR><LF>**   ← *The processing is successfully done.*

6. Description of ASCII Indicators

6.1. General Indicators

6.1.1. OK - Command was adopted by the Bluetooth Module

6.1.1.1. Description:

This indicator indicates a command was adopted by the Bluetooth Module successfully.

6.1.1.2. Syntax:

Synopsis:
OK<CR><LF>

6.1.1.3. Parameter Description:

None.

6.1.1.4. Examples:

Ex. 6.1. To make Bluetooth Module discoverable:

➔ **BC:MD=01<CR><LF>**   ← *make Bluetooth Module discoverable.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

6.1.2. ER - Error

6.1.2.1. Description:

This indicator indicates there is an error detected in the command sent by the host.

6.1.2.2. Syntax:

Synopsis:
ER{=Error Code}<CR><LF>

6.1.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Error Code	The error code to give the reason of an error. Value: 01h—03h 01: The command is not allowed in current state. 02: The command is not given in proper format. 03: The command cannot be recognized.	M	

6.1.2.4. Examples:

Ex. 6.2.

➔ **RT=01**   ← *The module is advertising now.*

➔ **BC:MD=01<CR><LF>**   ← *Send command to make Bluetooth Module discoverable.*

⬅ **ER=01<CR><LF>**   ← *response from the module to indicate the command is not allowed in current state since the Bluetooth Module is already discoverable(advertising).*

Ex. 6.3.

➔ **BC:XX<CR><LF>**   ← *Send a command "XX" which is not in the supported command list.*

⬅ **ER=03<CR><LF>**   ← *response from the module to indicate the command cannot be recognized.*

6.1.3. UD—GATT Service UUID128

6.1.3.1. Description:

This indicator reports the GATT service UUID128 of the Module.

6.1.3.2. Syntax:

Synopsis:
UD{=Value Index,Value}<CR><LF>

6.1.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Value Index	The UUID128 has 128bits, which is represented by 8 16-bits words. The 8 words are stored in a lowest word first order. Index 0 is the lowest word of the UUID128. Value: 0000-0007	M	
Value	The value of the word. The word value is print in a lowest byte first order. Refer to the example in 5.1.1.4.	M	

6.1.3.4. Examples:

Refer to the examples in section 5.1.1.4.

6.1.4. AD - Bluetooth address of the Module

6.1.4.1. Description:

This indicator reports the Bluetooth address of the Module.

6.1.4.2. Syntax:

Synopsis:
AD{=Bluetooth Address Type,Bluetooth Addrress}<CR><LF>

6.1.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Bluetooth Address Type	The type of the connected device Bluetooth address. Value: 00h-01h 00: Public device address 01: Random device address	M	Only available when there's a connection existed (state = 04)
Bluetooth Addrress	The Bluetooth address of the connected BLE central	M	

6.1.4.4. Examples:

Refer to the examples in section 5.1.2.4.

6.1.5. AP - GAP Appearance Characteristic

6.1.5.1. Description:

This indicator indicates the GAP appearance characteristic.

6.1.5.2. Syntax:

Synopsis:
AP{=Appearance value}<CR><LF>

6.1.5.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Appearance value	The GAP appearance characteristic of the Bluetooth Module. Value: 0000h—FFFFh as defined by Bluetooth Assigned Number, refer to Table 7.1 Default: per firmware version	M	

6.1.5.4. Examples:

Refer to 5.1.3.4.

6.1.6. FT - Feature configuration

6.1.6.1. Description:

This indicator will report current feature configuration of the Bluetooth Module.

6.1.6.2. Syntax:

Synopsis:
FT{=Features}<CR><LF>

6.1.6.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Features	Refer to 5.1.4.3	M	

Notes:
1. *The default feature configuration may be different per software version.*

6.1.6.4. Examples:

Refer to the examples in section 5.1.4.4.

6.1.7. CC - Preferred Connection Parameter Characteristic

6.1.7.1. Description:

This indicator report the GAP preferred connection parameter characteristic of Bluetooth Module.

6.1.7.2. Syntax:

Synopsis:
CC{=Min Interval,Max Interval, Latency, Supervision Timeout}<CR><LF>

6.1.7.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Min interval	The minimum value for the connection interval. Hex Value: xxxx (0006 to 0C80, FFFF means not specifically defined) Default depends on firmware	M	
Max interval	The maximum value for the connection interval. Hex Value: xxxx (0006 to 0C80, FFFF means not specifically defined) Default depends on firmware	M	
Latency	The slave latency for the connection in number of connection events. Hex Value: xxxx (0000 to 03E8) Default depends on firmware	M	
Supervision timeout	The connection supervisor timeout nultiplier as a nultple of 10ms. Hex Value: xxxx (0006 to 0C80, FFFF means not specifically defined) Default depends on firmware	M	

6.1.7.4. Examples:

Refer to the examples in section 5.1.5.4.

6.1.8. NM - GAP Device Name Characteristic

6.1.8.1. Description:

This indicator will report current GAP device name characteristic of the Bluetooth Module.

6.1.8.2. Syntax:

Synopsis:
NM{=Name}<CR><LF>

6.1.8.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Name	Refer to 5.1.6.3	M	

6.1.8.4. Examples:

Refer to the examples in section 5.1.6.4.

6.1.9. BR - UART baud rate

6.1.9.1. Description:

This indicator will report current UART baud of the Bluetooth Module.

6.1.9.2. Syntax:

Synopsis:
BR{=Baud Rate}<CR><LF>

6.1.9.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Baud Rate	Refer to 5.1.7.3	M	

	Bluetooth Low Energy Software Interface Application Manual	RoHS Compliant
Date of Issue: Jan 15th, 2014	**ABBTM-NVC-MDCS71**	**ESD Sensitive**
Page (24) of (33)	**Abracon Drawing # 453732**	Revision #: A

ABRACON CORPORATION
The Power of Linking Together

6.1.9.4. Examples:

Refer to the examples in section 0.

6.2. Connection and Data Packet Indicators

6.2.1. RT - Radio State

6.2.1.1. Description:

This indicator reports the radio state.

6.2.1.2. Syntax:

Synopsis:
RT{=Radio State},[Remote Bluetooth Address Type],[Remote Bluetooth Address]<CR><LF>

6.2.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Radio State	The state of radio connection. State Value: 00h-05h 00: idle and no connection exists. 01: the module is in fast advertising. 02: the module is in slow advertising. (connecting speed is slower, but power consumption is much lower than fast advertising) 03: the module is connected with a BLE central. 04: the module is trying to disconnect, this is a temporary state. 05: a security channel is made with the connected BLE central	M	
Remote Bluetooth Address Type	The type of the connected device BLE central address. Value: 00h-01h 00: Public device address 01: Random device address	O	Only available when there's a connection existed (state =05).
Remote Bluetooth Address	The Bluetooth address of the connected BLE central.	O	

6.2.1.4. Examples:

Refer to the examples in section 5.2.4.4.

6.2.2. DT—Data packet received from remote device

6.2.2.1. Description:

This indicator indicates there is a data packet received from a remote device.

6.2.2.2. Syntax:

Synopsis:
DT{=Data String}<CR><LF>

6.2.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
Data String	The received data. The raw data is output as what they are encoded in the remote sender. So the parser should handle the case that <CR> and <LF> is mixed in the data string.	M	

6.2.2.4. Examples:

Ex. 6.4. A data packet is received:

← DT=1234567890<CR><LF>   ← *a data string received from remote device.*

6.2.3. RI - RSSI

6.2.3.1. Description:

This indicator reports the RSSI of the current connection.

6.2.3.2. Syntax:

Synopsis:
RI{=RSSI}<CR><LF>

6.2.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
RSSI	The RSSI of the current connection. The value is in dBm and the accuracy is 6dB. The value is in two's complement signed format. So the 2 hex characters have a range of +127 to -128.	M	

6.2.3.4. Examples:

Refer to the examples in section 5.2.5.4.

6.3. Power Mode Indicators

The module goes directly to the power mode which "PD" command requests. It doesn't send any indicator when power mode changes.

6.4. Security Indicators

Currently no security indicator is supported. The module uses "Just work" simple pairing to bond with a BLE central.

6.5. iBeacon Indicators

6.5.1. BA – iBeacon Major Number

6.5.1.1. Description:

This indicator reports the iBeacon major number.

6.5.1.2. Syntax:

Synopsis:
BA{=iBeacon Major Number}<CR><LF>

6.5.1.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
iBeacon Major Number	The new iBeacon major number in four-char hex format. Value: 0000h—FFFFh	M	

6.5.1.4. Examples:

Ex. 6.5. To query the current iBeacon major number:

➔ **BC:BA<CR><LF>** ⬅ *Query current iBeacon major number.*

⬅ **BA=ff01<CR><LF>** ⬅ *The current number is 0xff01.*

6.5.2. BI—iBeacon Minor Number

6.5.2.1. Description:

This indicator reports the iBeacon minor number.

6.5.2.2. Syntax:

Synopsis:
BI{=iBeacon Minor Number}<CR><LF>

6.5.2.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
iBeacon Minor Number	The new iBeacon major number in four-char hex format. Value: 0000h—FFFFh	M	

6.5.2.4. Examples:

Ex. 6.6. To query the iBeacon minor number:

➔ **BC:BI<CR><LF>** ⬅ *Query the current iBeacon minor number.*

⬅ **BI=ff01<CR><LF>** ⬅ *the current iBeacon minor number is 0xff01.*

6.5.3. BP—iBeacon RF Power Level Measured at One Meter

6.5.3.1. Description:

This indicator reports the iBeacon transmission power at 1 meter distance. Normally the power is set by Abracon to indicate the BLE RF power level measured at 1 meter distance from the module. Customer can fine tune it to match with their own environment.

6.5.3.2. Syntax:

Synopsis:
BP{=iBeacon Power Level}<CR><LF>

6.5.3.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
iBeacon RF power level measured at one meter	The new power level in four-char hex format. The unit is dBm and it is normally a minus number. The default value is 0xffc5 (-59dBm). A valid range should be within -30dBm to -80dBm.	M	

6.5.3.4. Examples:

Ex. 6.7. To query the current power level:

➔ **BC:BP<CR><LF>** ⬅ *Query current iBeacon RF power level measured at one meter.*

⬅ **BP=ffc5<CR><LF>** ⬅ *The current power level is 0xffc5 (-59dBm).*

6.5.4. BV—iBeacon Advertisement Interval

6.5.4.1. Description:

This indicator reports the iBeacon advertisement interval.

6.5.4.2. Syntax:

Synopsis:
BV{=iBeacon Advertisement Interval}<CR><LF>

6.5.4.3. Parameter Description:

Parameter	Description	Mandatory or Optional	Comments
iBeacon advertisement interval	The interval in four-char hex format. The unit is milliseconds. Value: 0000h—FFFFh	M	

6.5.4.4. Examples:

Ex. 6.8. To query the current iBeacon advertisement interval:

➔ **BC:BV<CR><LF>** ⬅ *Query the current iBeacon advertisement interval.*

⬅ **BV=03e8<CR><LF>** ⬅ *The current advertisement interval is 0x03e8 (1000ms or 1s).*

7. Appendix A: Examples

7.1. Work with the iOS Devices

To simplify the integration of the module into an iOS accessory, Abracon also supplies library on iOS to help customer's apps communicating with the Abracon BLE module. Abracon has another user manual for the iOS library.

Two examples of the common commands and indicators log when work with iOS devices at the module side are listed below. **The iOS app should use the same UUID128 as described in 5.1.1 to make the connection.**

Ex. 7.1, automatically advertising after power up:

➔ **iGate V1.2<CR><LF>**   ← *The module powers up and print its version number 1.2.*

➔ **RT=01<CR><LF>**   ← *The module start advertising automatically after power up.*

← **RT=03<CR><LF>**   ← *A BLE central is connecting with the module.*

← **RT=05,01,64200c2c462a<CR><LF>**   ← *Indicate that the module has been connected with an iOS device whose random address is 64:20:0C:2C:46:2A.*

➔ **BC:DT=Hello iOS central<CR><LF>**   ← *Send "Hello iOS central" to connected iOS app.*

← **DT=Hello NovaComm BLE<CR><LF>**   ← *Got a message "Hello NovaComm BLE" from iOS app.*

……

Ex. 7.2, start advertising by command:

➔ **BC:MD=01<CR><LF>**   ← *When the module is in "RT=00" status.*

← **OK<CR><LF>**   ← *response from the module to indicate the command is adopted.*

← **RT=03<CR><LF>**   ← *A BLE central is connecting with the module.*

← **RT=05,01,64200c2c462a<CR><LF>**   ← *Indicate that the module has been connected with an iOS device whose random address is 64:20:0C:2C:46:2A.*

……

7.2. Code of GAP Appearance

Table 7.1 Code of GAP Appearance

Value(DEC)	Device Type	Catagory
0	Unknown	None
64	Generic Phone	Generic category
128	Generic Computer	Generic category
192	Generic Watch	Generic category
193	Watch: Sports Watch	Watch subtype
256	Generic Clock	Generic category
320	Generic Display	Generic category
384	Generic Remote Control	Generic category
448	Generic Eye-glasses	Generic category
512	Generic Tag	Generic category
576	Generic Keyring	Generic category
640	Generic Media Player	Generic category
704	Generic Barcode Scanner	Generic category
768	Generic Thermometer	Generic category
769	Thermometer: Ear	Thermometer subtype
832	Generic Heart rate Sensor	Generic category
833	Heart Rate Sensor: Heart Rate Belt	Heart Rate Sensor subtype
896	Generic Blood Pressure	Generic category
897	Blood Pressure: Arm	Blood Pressure subtype
898	Blood Pressure: Wrist	Blood Pressure subtype
960	Human Interface Device (HID)	HID Generic
961	Keyboard	HID subtype
962	Mouse	HID subtype
963	Joystick	HID subtype
964	Gamepad	HID subtype
965	Digitizer Tablet	HID subtype
966	Card Reader	HID subtype
967	Digital Pen	HID subtype
968	Barcode Scanner	HID subtype
1024	Generic Glucose Meter	Generic category
1088	Generic: Running Walking Sensor	Generic category
1089	Running Walking Sensor: In-Shoe	Running Walking Sensor subtype
1090	Running Walking Sensor: On-Shoe	Running Walking Sensor subtype
1091	Running Walking Sensor: On-Hip	Running Walking Sensor subtype
1152	Generic: Cycling	Generic category
1153	Cycling: Cycling Computer	Cycling subtype
1154	Cycling: Speed Sensor	Cycling subtype
1155	Cycling: Cadence Sensor	Cycling subtype
1156	Cycling: Power Sensor	Cycling subtype
1157	Cycling: Speed and Cadence Sensor	Cycling subtype

7.3. Control Packet Format for iLight (PWM) Extention

The ABBTM-NVC-MDCS71 Bluetooth module has four PWM output IOs (PIO3, PIO4, PIO9, PIO10). With proper firmware, special control packet can be send from the connected iOS device to the Bluetooth Module to control the four PWM ports.

Length of the packet (10, not include itself)	Command ID (0x00 for LED control)	LED index (0x00 ~ 0x03)	LED switch (00: off 01: on 02: dim)	LED dull on	LED dull off	LED dull hold	LED bright on	LED bright off	LED bright hold	LED ramp rate

— 11 bytes total —

LED index: 00~03 map to pio 3,4,9,10 of the module
LED switch:
00: always off
01: full on
02: dim pattern defined as following
dull_on_time the amount of time, in units of ~30us, for which the LED should be on during the dullest part of the flash sequence
dull_off_time the amount of time, in units of ~30us, for which the LED should be off during the dullest part of the flash sequence
dull_hold_time the amount of time, in units of ~16ms, for which the LED should be held in the dullest part of the flash sequence
bright_on_time the amount of time, in units of ~30us, for which the LED should be on during the brightest part of the flash sequence
bright_off_time the amount of time, in units of ~30us, for which the LED should be off during the brightest part of the flash sequence
bright_hold_time the amount of time, in units of ~16ms, for which the LED should be held in the brightest part of the flash sequence
ramp_rate the ramp rate for ramping between brightness levels, in units of ~30us per step with 0 being instantaneous(no ramp).

Figure 7.1 Packet Format of iLight (PWM)

And here's an example code in Xcode to send a iLight command to the connected Bluetooth BLE module.

```
- (IBAction)lightLum:(id)sender
{
  if([self.swLight isOn])
  {
    NSMutableData *valData;
    Byte cmd[11];
    cmd[0]=10;
    cmd[1]=00; // cmd id
    cmd[2]=02; // led index
    cmd[3]=02; //02; // switch off / on / breathing
    cmd[4]=(Byte)(self.slideLight.value*2); // 0; // dull on (unit 30us)
    cmd[5]=(Byte)((100-self.slideLight.value)*2); //200; // dull off (unit 30us)
    cmd[6]=62; // dull hold  (unit 16ms)
    cmd[7]=(Byte)(self.slideLight.value*2); // bright on (unit 30us)
    cmd[8]=(Byte)((100-self.slideLight.value)*2); // bright off (unit 30us)
    cmd[9]=62; // bright hold  (unit 16ms)
    cmd[10]=166; // ramp rate (unit 30us)
    valData = [NSData dataWithBytes:cmd length:11];
    [iGate sendData:valData];
  }
}
```

8. Notes

i)   The parts are manufactured in accordance with this specification. If other conditions and specifications which are required for this specification, please contact ABRACON for more information.

ii) ABRACON will supply the parts in accordance with this specification unless we receive a written request to modify prior to an order placement.

iii) In no case shall ABRACON be liable for any product failure from in appropriate handling or operation of the item beyond the scope of this specification.

iv) When changing your production process, please notify ABRACON immediately.

v)   ABRACON Corporation's products are COTS – Commercial-Off-The-Shelf products; suitable for Commercial, Industrial and, where designated, Automotive Applications. ABRACON's products are not specifically designed for Military, Aviation, Aerospace, Life-dependant Medical applications or any application requiring high reliability where component failure could result in loss of life and/or property. For applications requiring high reliability and/or presenting an extreme operating environment, written consent and authorization from ABRACON Corporation is required. Please contact ABRACON Corporation for more information.

vi) All specifications and Marking will be subject to change without notice.

9. ABRACON CORPORATION – TERMS & CONDITIONS OF SALE

The following are the terms and conditions under which Abracon Corporation ("AB") agrees to sell, to the entity named on the face hereof ("Buyer"), the products specified on the face hereof (the "Products"). Notwithstanding Buyer's desire to use standardized RFQs, purchase order forms, order forms, acknowledgment forms and other documents which may contain terms in addition to or at variance with these terms, it is expressly understood and agreed that other forms shall neither add to, nor vary, these terms whether or not these terms are referenced therein. Buyer may assent to these terms by written acknowledgment, implication and/or by acceptance or payment of goods ordered any of which will constitute assent.

1. **Prices**: Prices shown on the face hereof are in US dollars, with delivery terms specified herein and are exclusive of any other charges including, without limitation, fees for export, special packaging, freight, insurance and similar charges. AB reserves the right to increase the price of Products by written notice to Buyer at least thirty (30) days prior to the original date of shipment. When quantity price discounts are quoted by AB, the discounts are computed separately for each type of product to be sold and are based upon the quantity of each type and each size ordered at any one time. If any discounted order is reduced by Buyer with AB's consent, the prices shall be adjusted to the higher prices, if applicable, for the remaining order.

2. **Taxes**: Unless otherwise specified in the quotation, the prices do not include any taxes, import or export duties, tariffs, customs charges or any such other levies. Buyer agrees to reimburse AB the amount of any federal, state, county, municipal, or other taxes, duties, tariffs, or custom charges AB is required to pay. If Buyer is exempt from any such charges, Buyer must provide AB with appropriate documentation.

3. **Payment Terms**: For each shipment, AB will invoice Buyer for the price of the Products plus all applicable taxes, packaging, transportation, insurance and other charges. Unless otherwise stated in a separate agreement or in AB's quotation, payments are due within thirty (30) days from the date of invoice, subject to AB's approval of Buyer's credit application. All invoicing disputes must be submitted in writing to AB within ten (10) days of the receipt of the invoice accompanied by a reasonably detailed explanation of the dispute. Payment of the undisputed amounts shall be made timely. AB reserves the right to require payment in advance or C.O.D. and otherwise modified credit terms. When partial shipments are made, payments for such shipments shall become due in accordance with the above terms upon submission of invoices. If, at the request of Buyer, shipment is postponed for more than thirty (30) days, payment will become due thirty days after notice to Buyer that Products are ready for shipment. Any unpaid due amounts will be subject to interest at one decimal five percent (1.5%) per month, or, if less, the maximum rate allowed by law.

4. **Delivery and Shipment**: Shipment dates are estimates only. Failure to deliver by a specified date shall neither entitle Buyer to any compensation nor impose any liability on AB. AB reserves the right to ship and bill ten percent more or less than the exact quantity specified on the face hereof. All shipments will be made Ex Works as per Incoterms 2000 from AB's place of shipment. In the absence of specific instructions, AB will select the carrier. Claims against AB for shortages must be made in writing within ten (10) days after the arrival of the shipment. AB is not required to notify Buyer of the shipment. Buyer shall pay all freight charges, insurance and other shipping expenses.

Freight charges, insurance and other shipping expenses itemized in advance of actual shipment, if any, are estimates only that are calculated on the basis of standard tariffs and may not reflect actual costs. Buyer must pay actual costs.

5. **Purchase Order Changes and Cancellations**: Purchase orders for standard AB Products may not be canceled within sixty (60) days of the original shipping date. Purchase orders for non-standard AB Products are non-cancelable and non-returnable. All schedule changes must be requested at least thirty (30) days prior to original shipping date. Maximum schedule change "push-out" shall be no more than thirty (30) days from original shipping date. AB may terminate or cancel this order, in whole or in part, at any time prior to the completion of performance by written notice to Buyer without incurring any liability to Buyer for breach of contract or otherwise. AB reserves the right to allocate Products in its sole discretion among Buyer and other potential buyers, or defer or delay the shipment of any Product, which is in short supply due to any reason.

6. **Title and Risk of Loss**: AB's responsibility for any loss or damage ends, and title passes, when Products are delivered Ex Works as per Incoterms 2000 at AB's designated shipping location to carrier, to Buyer or to Buyer's agent, whichever occurs first.

7. **Packing**: Packaging shall be AB's standard shipping materials or as specified on the face hereof. Any cost of non-standard packaging and handling requested by Buyer shall be abided by AB provided Buyer gives reasonable prior notice and agrees in writing to pay additional costs.

8. **Security Interest**: Buyer hereby grants AB a purchase money security interest in the Products sold and in the proceeds of resale of such Products until such time as Buyer has paid all charges. AB retains all right and remedies available to AB under the Uniform Commercial Code.

9. **Specifications**: Specifications for each Product are the specifications specified in the published datasheets of such Product, as of the date of AB's quotation (the "Specifications"). Except as otherwise agreed, AB reserves the right to modify the Specifications at any time without adversely affecting the functionality.

10. **Acceptance**: Unless Buyer notifies AB in writing within ten (10) days from the date of receipt of Products that the Products fail to conform to the Specifications, the Products will be deemed accepted by Buyer. No such claim of non-conformity shall be valid if (i) the Products have been altered, modified or damaged by Buyer, (ii) the rejection notice fails to explain the non-conformance in reasonable detail and is not accompanied by a test report evidencing the non-conformity, or (iii) rejected Products are not returned to AB within thirty (30) days of rejection; provided, that no Product returns may be made without a return material authorization issued by AB.

11. **Limited Warranties and Disclaimers**: AB warrants to Buyer that each Product, for a period of twelve (12) months from shipment date thereof, will conform to the Specifications and be free from defects in materials and workmanship. AB's sole liability and Buyer's exclusive remedy for Products that fail to conform to this limited warranty ("Defective Products") is limited to repair or replacement of such Defective Products, or issue a credit or rebate of no more than the purchase price of such Defective Products, at AB's sole option and election. This warranty shall not apply: (i) if Products have been damaged or submitted to abnormal conditions (mechanical, electrical, or thermal) during transit, storage, installation, or use; or (ii) if Products are subject to Improper Use (as defined below); or (iii) if the non-conformance of Products results from misuse, neglect, improper testing, storage, installation, unauthorized repair, alteration, or excess usage at or beyond the maximum values (temperature limit, maximum voltage, and other Specification limits) defined by AB; (iv) to any other default not attributable to AB; or (v) removal, alteration, or tampering of the original AB product labeling. This warranty does not extend to Products or components purchased from entities other than AB or AB's authorized distributors or to third-party software or documentation that may be supplied with any Product. In the event no defect or breach of warranty is discovered by AB upon receipt of any returned Product, such Product will be returned to Buyer at Buyer's expense and Buyer will reimburse AB for the transportation charges, labor, and associated charges incurred in testing the allegedly Defective Product. The above warranty is for Buyer's benefit only, and is non-transferable. OTHER THAN THE LIMITED WARRANTY SET FORTH ABOVE, AB MAKES NO WARRANTIES, EXPRESS, STATUTORY, IMPLIED, OR OTHERWISE AND SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT, TO THE MAXIMUM EXTENT PERMITTED BY LAW. WITHOUT LIMITING THE GENERALITY OF THE FOREGOING DISCLAIMERS, AB INCORPORATES BY REFERENCE ANY PRODUCT-SPECIFIC WARRANTY DISCLAIMERS SET FORTH IN THE PUBLISHED PRODUCT DATASHEETS.

12. **Limitation of Liability**: AB SHALL HAVE NO LIABILITY FOR LOSS ARISING FROM ANY CLAIM MADE AGAINST BUYER, OR FOR SPECIAL, INDIRECT, RELIANCE, INCIDENTAL, CONSEQUENTIAL, OR PUNITIVE DAMAGES INCLUDING, WITHOUT LIMITATION, LOSS OF USE, PROFITS, REVENUES, OR COST OF PROCUREMENT OF SUBSTITUTE GOODS BASED ON ANY BREACH OR DEFAULT OF AB, HOWEVER CAUSED, AND UNDER ANY THEORY OF LIABILITY. BUYER'S SOLE REMEDY AND AB'S SOLE AND TOTAL LIABILITY FOR ANY CAUSE OF ACTION, WHETHER IN CONTRACT (INCLUDING BREACH OF WARRANTY) OR TORT (INCLUDING NEGLIGENCE OR MISREPRESENTATION) OR UNDER STATUTE OR OTHERWISE SHALL BE LIMITED TO AND SHALL NOT EXCEED THE AGGREGATE AMOUNTS PAID BY BUYER TO AB FOR PRODUCTS WHICH GIVE RISE TO CLAIMS. BUYER SHALL ALWAYS INFORM AB OF ANY BREACH AND AFFORD AB REASONABLE OPPORTUNITY TO CORRECT ANY BREACH. THE FOREGOING LIMITATIONS SHALL APPLY REGARDLESS OF WHETHER AB HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES AND NOTWITHSTANDING THE FAILURE OF ESSENTIAL PURPOSE OF ANY LIMITED REMEDY.

13. **Improper Use**: Buyer agrees and covenants that, without AB's prior written approval, Products will not be used in life support systems, human implantation, nuclear facilities or systems or any other application where Product failure could lead to loss of life or catastrophic property damage (each such use being an "Improper Use"). Buyer will indemnify and hold AB harmless from any loss, cost, or damage resulting from Improper Use of the Products.

14. **Miscellaneous**: In the event of any insolvency or inability to pay debts as they become due by Buyer, or voluntary or involuntary bankruptcy proceeding by or against Buyer, or appointment of a receiver or assignee for the benefit of creditors of Buyer, AB may elect to cancel any unfulfilled obligations. No Products or underlying information or technology may be exported or re-exported, directly or indirectly, contrary to US law or US Government export controls. AB will be excused from any obligation to the extent performance thereof is caused by, or arises in connection with, acts of God, fire, flood, riots, material shortages, strikes, governmental acts, disasters, earthquakes, inability to obtain labor or materials through its regular sources, delay in delivery by AB's supplies or any other reason beyond the reasonable control of AB. In the event any one or more of the provisions contained herein shall for any reason be held to be invalid, illegal, or unenforceable in any respect, such invalidity, illegality, or unenforceability shall not affect any other provision hereof and these terms shall be construed as if such invalid, illegal, or unenforceable provision had never been contained herein. A waiver of a breach or default under these terms shall not be a waiver of any subsequent default. Failure of AB to enforce compliance with any of these terms shall not constitute a waiver of such terms. These terms are governed by the laws of the State of California without reference to conflict of law principles. The federal and state courts located within the State of California will have exclusive jurisdiction to adjudicate any dispute arising out of these terms.