DataExchanger-AT

User Guide

Version 1.3

Issue 1

[2017-03-26]

Change History

Issue	Date	Changes	Author
1	2017-03-26	1. Based on version 1.2.1 2. Added DataExchanger in Central Role. 3. Synced with DataExchanger AT Command Reference 1.3	ML

© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics' non-disclosure agreement.

New In This Release

New Features

- Introduce DataExchanger in Central Role examples

Changes

Command / Notification	Change	Notes
AT+EC	New response parsing mode	To support AT response parsing
AT+I2CRW	New timestamp parameter in the response	Running clock from MCU recorded when the I2C command is issued on the driver.
AT+LSND	Send to specific connection AT+LSND=<connIdx>,<bytesToSend>	Since Central Role supports multiple connections, the new connIdx parameters will allow users to send data to a specific connection.
+LRCV	Indicate which connection the data is coming from +LRCV:<connIdx>,bytesSent>	Since Central Role supports multiple connections, the new connIdx parameters will indicate users from which specific connection the data are received.
AT+LRL?	Deprecated	
AT+DL?	Deprecated	

Supported Firmware Release

- R444.

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

Table of Contents

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

Introduction

DataExchanger-AT is an embedded application running on CC264BPA modules that enables customers to incorporate Bluetooth connectivity for their applications and products. DataExchanger-AT features a serial communication interface to connect to customers' host MCU for data transfer and command line. Through its command line, DataExchanger-AT also can control local peripherals via other interfaces where sensors, actuators, and lights, etc, can be hooked up to. The key advantage of using DataExchanger-AT is its substantial saving in design effort that allows fast time-to-market but without adding much cost. DataExchanger-AT is ideal to be used in applications and products that require connection to smartphones, PC, standalone wireless console, and IoT gateways.

Key Features

- Single or Dual channels communication to support different usage scenarios:
  - Dedicated channel for data bridging with host MCU via UART interface, and dedicated channel for CLI remotely accessed to control local module functions.
  - CLI locally accessed by host MCU with inband data send and receive capabilities
  - Simple and Just-Work data bridging – i.e. no CLI involved
- CLI (remotely or locally accessed) to setup and control peripheral functions including gpio, i2c, and pwm.
- Simple AT command set supporting machine optimized commands and responses.
- Data bridging support hardware flow control or command based flow control (for host hardware not supporting UART hardware flow control)
- Startup auto-connect and data bridging as the default configuration.
- Power saving control

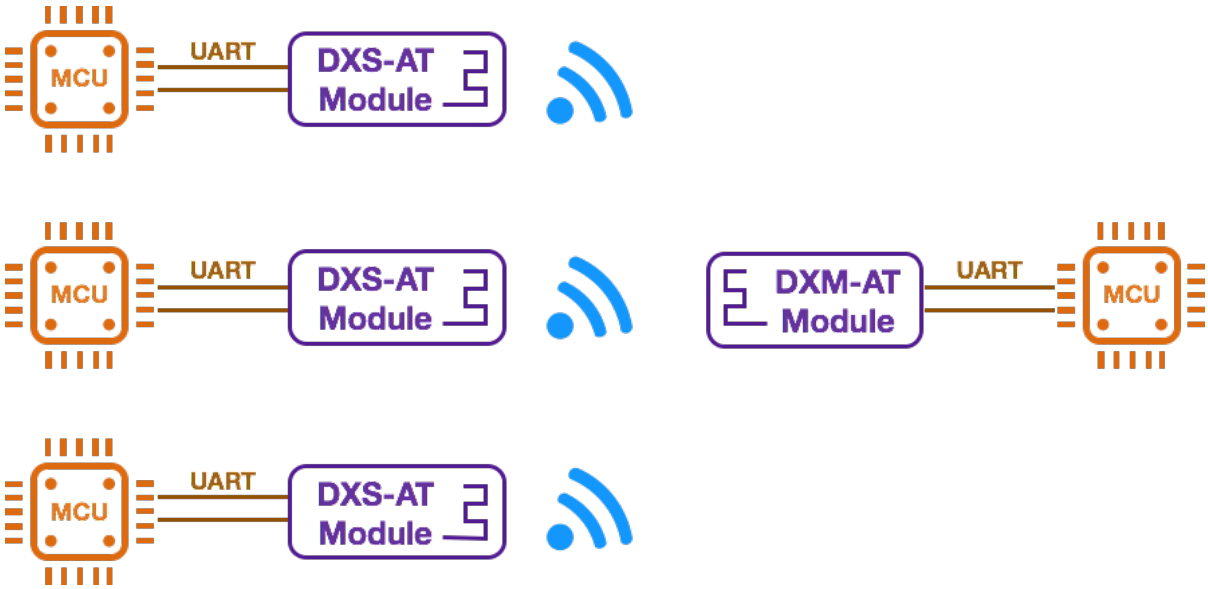
Usage Scenarios

The following diagrams illustrate some possible usage scenarios for DataExchanger-AT modules in real world.

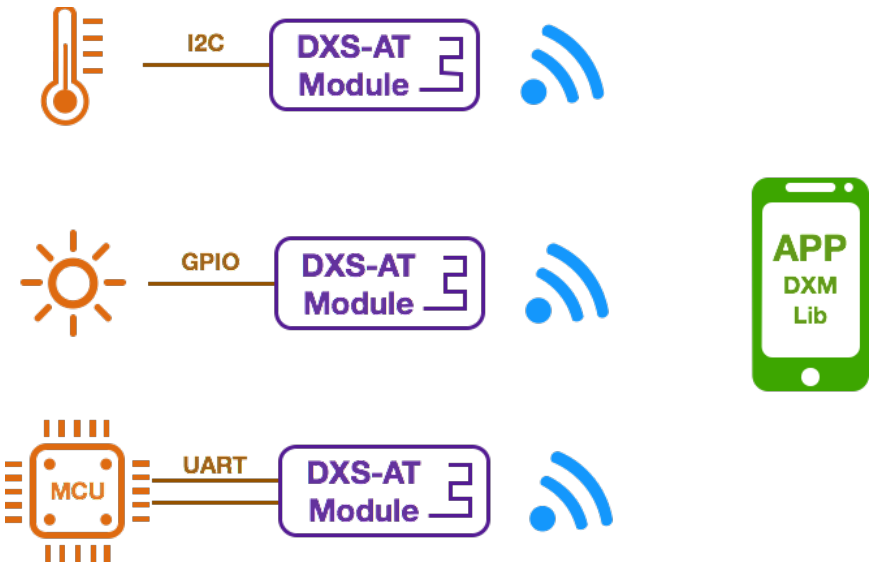
© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

Wireless Serial-to-Serial Communication Bridge

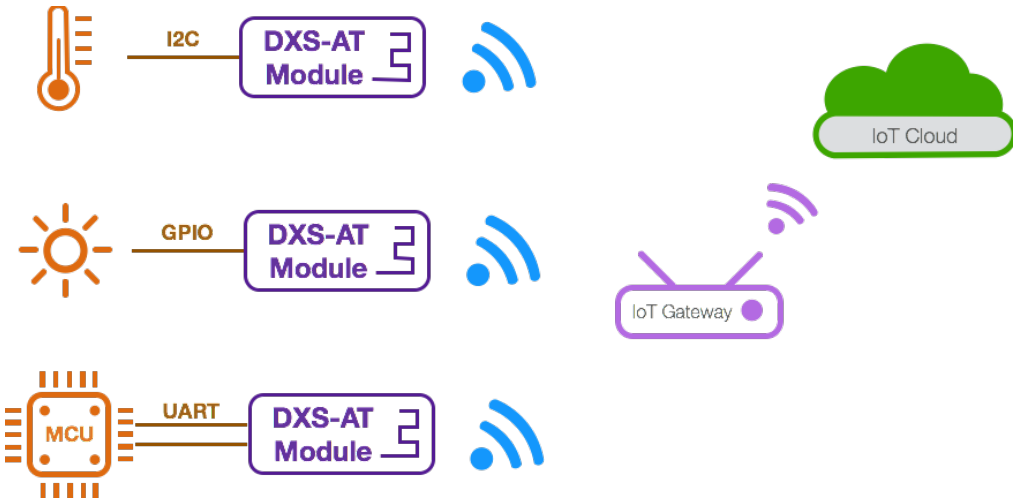


Accessory Device Connected to Smartphone



© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics' non-disclosure agreement.

Accessory Device Connected to Cloud via IoT Gateway



Serial Interface for Host MCU Communication

UART is used to connect host MCU and DataExchanger-AT. Host MCU uses this interface to send data stream or commands depending on the mode operated.

Control Interface for Peripherals

Peripherals are the hardware devices that the DataExchanger-AT modules are directly connected – e.g. sensors, light, servo, etc. Currently DataExchanger-AT supports GPIO, I2C, and PWM interface for peripheral control. ADC, SPI, and 1-wire interface will be added in the future.

AT Command Reference

Please refer to the latest issue of *DataExchanger-AT Command Reference v1.3* documentation.

Supported Hardware

Please refer to the latest issue of *DataExchanger-AT Firmware R444 Release Note* for details.

Central/Peripheral Roles

DataExchanger supports *central role* and *peripheral role*. While the DataExchanger app is always a central role device, a CC264BPA module can run a firmware that supports either central role or peripheral role. This guides focuses on the firmware which supports peripheral role.

© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics' non-disclosure agreement.

Getting Start

The easiest way to demonstrate DataExchanger-AT function is to use CC26xBPA-TIEM evaluation module with DataExchanger-AT firmware preinstalled. The CC26xBPA-TIEM evaluation module can be powered standalone or via Texas Instruments' SmartRF06 evaluation board. Please refer to the latest *CC26xBPA-TIEM User Guide* for details.

Other equipment needed for this demo are:

- A PC with a USB port and a terminal software to emulate the host MCU
- A USB-to-UART dongle or cable that supports TTL 3.3v
- An iPhone, iPad, or iPodTouch running iOS7 and up.
- An iTune account, which can be used to download free app from Apple's App Store.

Connect The Wires

Connect the wires between CC26xBPA-TIEM evaluation module and USB-to-UART dongle according to the following diagram.

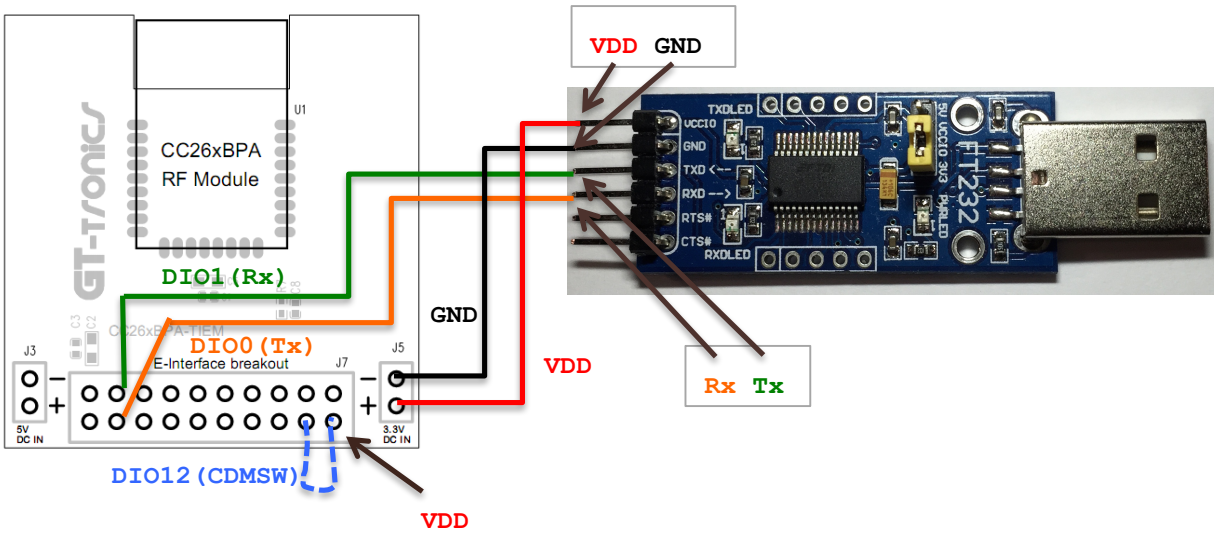


Figure 1 Connect The Wire Diagram

Power Up

Follow these steps to power up the CC26xBPA-TIEM evaluation module and connect to a PC.

1. Connect the USB-to-UART dongle (or cable) to the PC.
2. Install driver for the USB-to-UART Bridge if needed. Depending on the OS, some OSs will automatically install the driver.
3. Start a terminal program in the PC, then select the com port and choose the following UART setting
    - Baud rate: 115200
    - Data Length: 8 bits

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

- Stop Bit: 1 bit
- Parity: None
- Flow Control: None

After all the steps above the DataExchanger-AT firmware will be functioning. The next step will see end-to-end data transfer.

Data Transfer

After power up, DataExchanger-AT is ready to connect and bridge data. No command is necessary to prepare for data transfer.

The iOS DataExchanger app from Apple App Store will be used to connect to the CC26xBPA-TIEM evaluation module. GT-tronics also provides the DataExchangerDemo app source code that can be used to build custom app to work with this demo.

Follow these steps to see data transfer:

1. Download the DataExchanger app from Apple App Store. Use “DataExchanger” keyword to locate the app. Make sure the publisher is GT-tronics.
2. Make sure your Bluetooth is turned on.
3. Open the DataExchanger app and bring the CC26xBPA-TIEM evaluation module very close to your iDevice to make connection.
4. Select the Data Channel tab on the app.
5. Change display to text mode by tapping the “Hex” button on the navigation bar..
6. Type some characters in the PC terminal. Those characters will appear on the app’s receiver console. To send something in the other direction, tap the input text field in the app to bring up the keyboard. Type some characters and hit return. Those characters will appear on the PC terminal screen.
7. See Figure 2 for some screen shots.

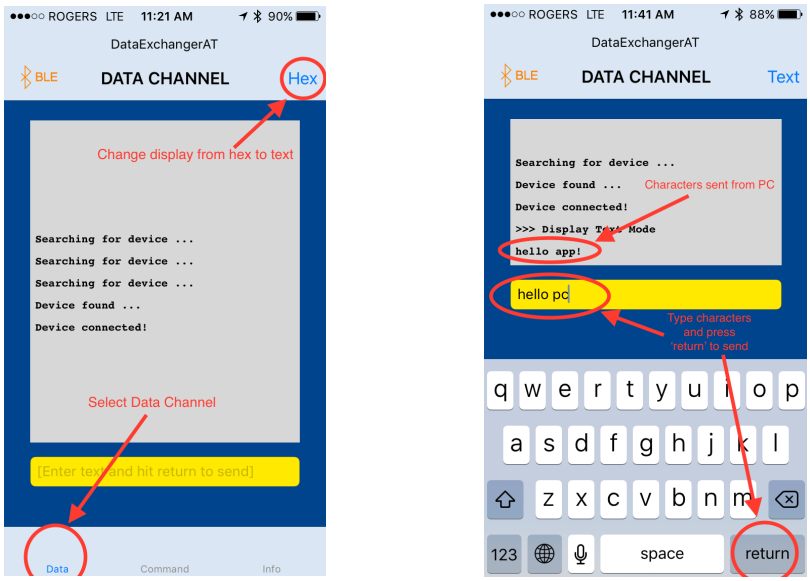


Figure 2

© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics’ non-disclosure agreement.

Remote Command Line

Congratulation! You have made the successful data transfer using a DataExchanger-AT module and DataExchanger app. The next step is to check out the remote command line using the app.

The remote command line feature in the app allows you to execute some command functions in the module locally. For example, you can check the DataExchanger-AT firmware version, change the UART settings, setup and control some GPIO pins. The support command list can be found in *DataExchanger-AT Firmware R400 Release Note*, and the details of the command usage are documented in *DataExchanger-AT Command Reference v1.2*.

Follow these steps to use the remote command line to check out the firmware version.

1. Tap the receive console once to hide the keyboard.
2. Select the Command Channel tab on the app.
3. Type VS? (after AT+) in the input text field and press return
4. The receive console will show the response from the AT+VS? command.
5. See Figure 3 for the screen shots.

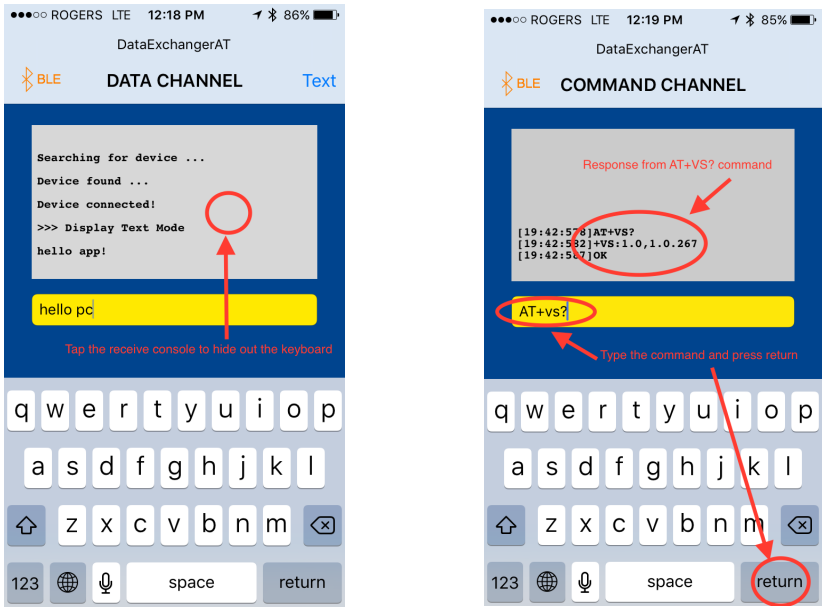


Figure 3

Different Operation Modes

So far, you have seen how data transfer is possible between the app and UART, and how the app can access the command line in the module remotely. There are other modes of operation that DataExchanger-AT supports. For example, one of the modes that is used often in supporting external MCU is to have the command line interface be facing to the UART always, no matter the module is connected or not.

DataExchanger-AT supports four different modes of operations:

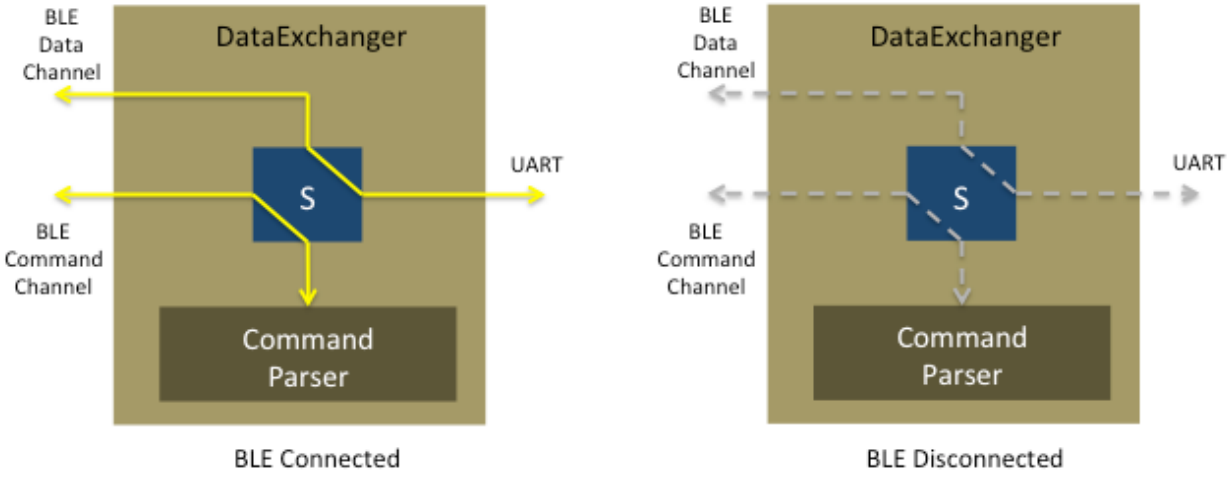
1. Dual BLE channels without local command line
2. Dual BLE channels with local command line

© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics' non-disclosure agreement.

3. Single BLE channel without local command line
4. Single BLE channel with local command line

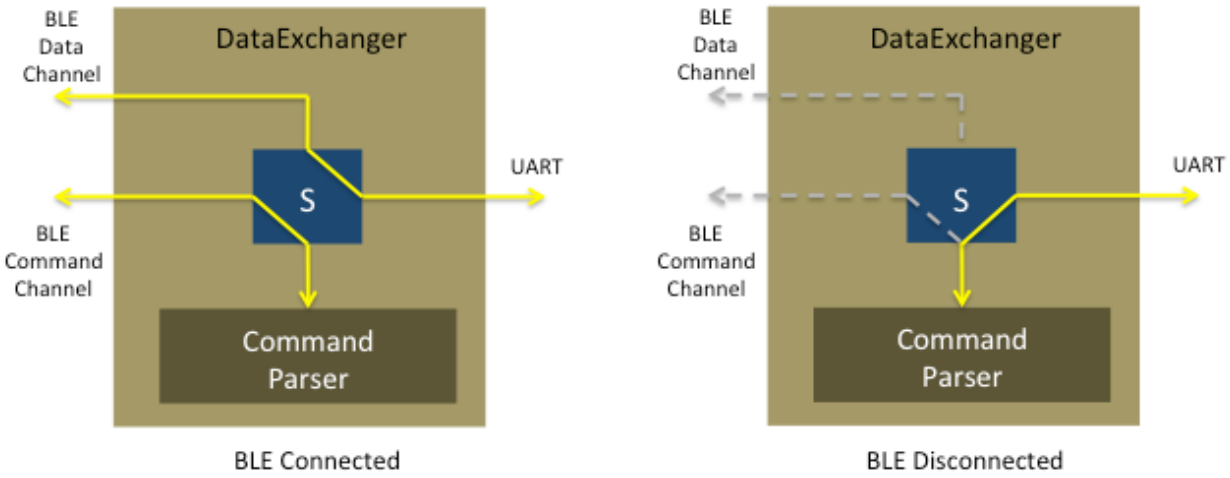
The following explains the different operation modes.

Dual BLE Channels Without Local Command Line



This is the mode that previous demo above runs in. The app is the sole source of control.

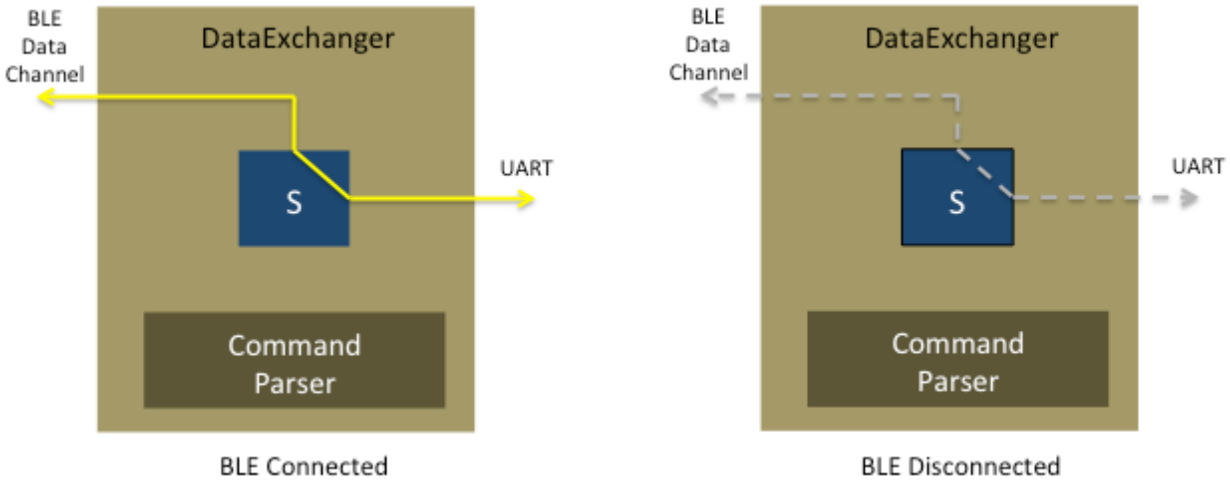
Dual BLE Channels With Local Command Line



This is the mode that the app and the terminal (PC or another MCU) shares the access of the command parser. However, the app will get the full access of the command parser once connected.

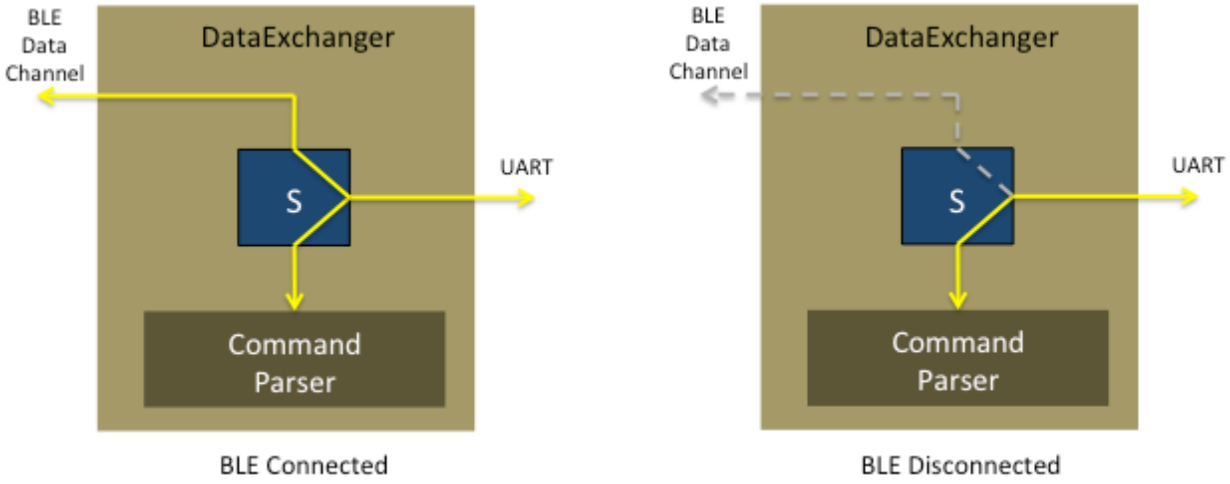
© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics' non-disclosure agreement.

Single BLE Channels Without Local Command Line



This is the mode that the command parser is completely ignored by the app or the terminal on UART. Applications that just want the UART data transfer will be running in this mode.

Single Channels With Local Command Line



This is the mode that the command parser is always facing to the UART, whether the module is connected or not. The terminal (PC or MCU) on UART will always have the parser control. This is the only mode that allows the terminal to cut the connection at any time.

Setting In Different Operating Modes

There are two switches to determine which mode the DataExchanger-AT module will run into. One switch is the CDMSW pin on the module, and the other one is the Command Channel Enable switch in the DataExchanger app. The following table determines the modes:

Operating Mode	CDMSW Pin	Command Channel Enable
Dual Channel with Command Line	High	On
Dual Channel without Command Line	Low	On

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

Operating Mode	CDMSW Pin	Command Channel Enable
Single Channel with Command Line	High	Off
Single Channel without Command Line	Low	Off

To change the Command Channel Enable switch setting in the DataExchanger app, follow the steps below:

1. Go to the Info tab.
2. Select Micro Settings.
3. Toggle the switch besides "Command Channel".

In the next section, we will discuss Local Command Line in the Single Channel with Command Line mode – how to set CDMSW pin for local command line and how the terminal on UART be able to send and receive in-band data within command line.

Local Command Line

The Blue wire in Figure 1 is used to toggle the CDMSW pin. When `DIO12` is connected to VDD, it run in CLI mode. When `DIO12` is opened, it runs in data mode. There should be notification of "+CMDM" and "+DATM" popped up on the terminal while toggle the blue wire.

The default mode is Data Streaming. To switch to Command Line mode, do the following steps:

1. Connect the blue wire according to Figure 1
2. Observe for a "+CMDM" notification in the terminal
3. Make sure the terminal use <CR><LF> for line termination for both transmit and receive.
4. Type "AT" and return. DataExchanger-AT will response with "OK".

In-Band Sending Data

One of unique feature in DataExchanger-AT is ability to send (and receive) data in Command Line mode. To send data in Command Line mode, follows the example below:

AT Command and Response	Explanation
`>AT+LSND=0,10`	User sends 10 bytes of user data
`OK`	DataExchanger-AT responses "OK"
`>0123456789`	User types: `0123456789`
`+LSND:0,10`	Upon the last character ('9')was input, DataExchanger-AT will response with "+LSND" notification.

In-Band Receiving Data

There is no command needed to receive data in Command Line mode. However, user application can pick a different data presentation mode for the received data. The following table shows how to display received data in Compact Hex format.

AT Command and Response	Explanation
`>AT+DTP?`	User queries Data Presentation mode

© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics' non-disclosure agreement.

AT Command and Response	Explanation
+DTP:0 Binary OK	DataExchanger-AT responses with the current Data Presentation mode. The default is Binary (0). Other supported modes are Compact Hex (1), and Formatted Hex (2).
>AT+DTP=1	User sets Data Presentation mode to Compact Hex (1)
+DTP:1 Compact Hex OK	DataExchanger-AT responses with the current Data Presentation mode. Please note DTP mode is not persistent and will be reset (to binary) after each system reset.
+LRCV:0,10 30 31 32 33 34 35 36 37 38 39	User types "0123456789" on the Tx window in the DataExchanger App, then hit return. DataExchanger-AT will response with data in the left. The 1st line is the "+LRCV" notification to indicate that there are 10 bytes of user data received. The 2nd line is the received data presented in hex format.

Restart Connection

The following command and response sequence table illustrate a connection restart.

AT Command and Response	Explanation
>AT+LSTOP	User disconnects current connection (or stop connecting)
OK	DataExchanger-AT response "OK"
+LCONN:3,0,01:02:03:04:05:06	DataExchanger-AT sends "+LCONN" notification to indicate change of connection status – in this case 3 means disconnect.
>AT+LCONN=10000	User request to connect with timeout equal to 10000ms.
+LCONN:1,0,01:02:03:04:05:06	DataExchanger-AT indicates "connecting".
+LCONN:2,0,01:02:03:04:05:06	If the connection with the app is successful, DataExchanger-AT indicates "connected".

The AT+LCONN command can be sent without the timeout parameter. If timeout parameter is skipped, the advertising pattern (i.e. how DataExchanger-AT would send connection invitation message) will follow what "AT+LADVP" command has set. Please see the next section for more details.

Using Other Supported Hardware

DataExchanger-AT can also run on other hardware besides CC26xBPA-TIEM. First, locate the proper firmware that supports the target hardware. Then use a JTAG emulator to download the firmware. Like the CC26xBPA-TIEM evaluation module, the firmware that runs on other supported hardware will auto-connect and stream data without typing any command. Simply follow the steps in the Power Up and the Data Transfer sections above to see data transfer over BLE link.

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

Advanced Control and Customization

Although it is zero configuration in bringing up DataExchanger-AT for BLE data transfer, the current behavior may not be what users want in their applications. For example, some users may want to decide themselves when DataExchanger-AT should start connecting rather than connecting at power up by default, or they need a way to power down the module for near-zero power consumption.

The following controls and customizations will be covered in this section:

- System I/O Pin Configuration
- Command Line and Data Streaming Modes
- Connection Management
- In-Band Data Transfer
- System Reset
- Power Saving Control
- UART Setting and Flow Control
- GPIO Setting and Control
- Other Supporting Commands

System I/O Pin Configuration

System I/O pins are the pins that host MCU will use to communicate with DataExchanger-AT. The following table describes the pin's name, pin function, default pin id, pin direction, pin reassignment, and pin's usage.

<table>
<tr><th rowspan="2">Pin Name</th><th colspan="3">CC26xBPA-TIEM</th><th colspan="2" rowspan="2">Usage</th></tr>
<tr><th>Default Pin ID</th><th>Direction</th><th>Assign Pin ID</th></tr>
<tr><td>UART_TX</td><td>DIO0</td><td>Output</td><td rowspan="4">Fixed</td><td colspan="2" rowspan="4">Serial Communication Interface (UART)</td></tr>
<tr><td>UART_RX</td><td>DIO1</td><td>Input</td></tr>
<tr><td>UART_RT</td><td>DIO3</td><td>Output</td></tr>
<tr><td>UART_CT</td><td>DIO2</td><td>Input</td></tr>
<tr><td>CDMSW</td><td>DIO12</td><td>Input (Pulled Low)</td><td rowspan="3">Changeable</td><td>Cmd Line (CL) / Data Stream (DS) Mode Switch</td><td>Set High=CL
Set Low=DS</td></tr>
<tr><td>CONNST</td><td>DIO6</td><td>Output</td><td>Connection Status</td><td>High=Connected
Low=Not Connected</td></tr>
<tr><td>SWKUP</td><td>Not Assigned</td><td>Input (Pulled High)</td><td>Sleep Wakeup</td><td>Negative Edge Triggered</td></tr>
</table>

Please note the default Pin IDs listed are specific to the DataExchanger-AT firmware built for CC26xBPA-TIEM. Other supported hardware may require a different version of DataExchanger-AT firmware that may have different default Pin IDs. Please refer to the latest issue of DataExchanger-AT Firmware R219 Release Note for more details.

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

Serial Communication Interface (UART) Pins

These pins cannot be reassigned to different pin id. However, the two hardware flow control pins (UART_RT and UART_CT) can be used for other functions, e.g. GPIO, when hardware flow control is disabled.

Command Line/Data Stream Mode Switch Pin

This pin is used to switch between command line mode and data streaming mode. This pin is pulled low internally. Therefore, if the pin were left opened, the operating mode would be data streaming. This pin can be reassigned but cannot be left unassigned.

Connection Status Pin

This pin is used to indicate the BLE connection status. This pin can be reassigned and can be left unassigned.

Sleep Wakeup Pin

This pin is used to wake up DataExchanger-AT after a standby sleep or a power down sleep. This pin is pulled high internally, and it needs a negative edge to trigger the wake up. This pin is unassigned by default and can be assigned with any available pin ID.

Pin ID Assignment

Pin ID assignment is a feature that allows the pin function to be assigned with different hardware pin other than the default. The CDMSW pin, CONNST pin, and SWKUP pin support pin ID assignment. To assign pin ID for these system I/O pin, please refer to "`AT+SYSP`" command in DataExchanger-AT Command Reference.

Command Line and Data Streaming Modes

DataExchanger-AT can be switched between command line mode and data streaming mode on the UART interface using the CDMSW hardware pin. In command line mode, the DataExchanger-AT command parser will listen for AT commands from the host MCU. In data streaming mode, DataExchanger-AT will bypass the parser and transfer the data from host MCU to the BT radio directly.

Whenever a mode switch happens, DataExchanger-AT will generate a notification. The following table describes the mode change and notification associations.

Mode Change	Notification
Command Line To Data Stream	`+DATM`
Data Stream To Command Line	`+CMDM`

Host MCU should wait for the change mode notification before carrying on further operation.

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

Connection Management

DataExchanger-AT supports the following connection management functions:

- Connect Control
- Connection and Advertising Parameter Settings
- Connect Policy

Connect Control

The following table captures the control functions.

Control Function	AT Command
Start Connecting	AT+LCONN
Stop Connecting or Disconnect	AT+LSTOP

Whenever a connection state change happens, a "+LCONN" notification with status will be generated. The following table shows the list of connection status notifications.

Connection Status	Notification
Idle	+LCONN:0,0,01:02:03:04:05:06
Connecting	+LCONN:1,0,01:02:03:04:05:06
Connected	+LCONN:2,0,01:02:03:04:05:06
Disconnected	+LCONN:5,0,01:02:03:04:05:06
Disconnected By Remote	+LCONN:6,0,01:02:03:04:05:06

Host MCU can always query the current connection status using "AT+LCONN?" command. The *Disconnect* connection status is transitional. Therefore, when the connection status is queried after receiving "+LCONN:5,0,01:02:03:04:05:06" is received, the response would be "+LCONN:0,0,00:00:00:00:00:00"

Please refer to DataExchanger-AT Command Reference document for details.

Connection and Advertising Parameter Settings

Connection parameters are used when connection is accepted by the other party and it is about to finalize the terms about interval and timeout. The defaults of those parameters are captured in the following table.

Connection Parameter	Default Value	Range
Connection Interval	20ms	7ms to 4000ms
Slave Latency	0	0 to 4
Connection Timeout	6000ms	100ms to 32000ms

© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics' non-disclosure agreement.

Connection Parameter	Default Value	Range
Parameter Update	1 (Yes)	0 or 1

Connection parameters are persistent and stored in NV flash. To update the parameters, please refer to "AT+LCONP" command in DataExchanger-AT Command Reference.

Advertising parameters are used during connection establishment with the other party. Advertising parameters determine how often the advertising packet should be sent and how long should it be kept sending – advertising packets is an "invite for connection" message to the third party devices. The following table captures the defaults of those parameters.

Advertising Parameter	Default Value	Range
Idle Between Fast Cycle and Slow Cycle	0ms	0 – 66535ms
Interval in Fast Cycle	30ms	7ms to 4000ms
Interval in Slow Cycle	100ms	7ms to 4000ms
Duration in Fast Cycle	30000ms	0 to 65535
Duration in Slow Cycle	65535 (Inf)	0 to 65535

Advertising is done in one or two cycles depending on the setting. The first cycle is called the fast cycle and the second cycle is called the slow cycle. Usually the advertising interval of fast cycle is higher than the one in slow cycle and the duration is just the opposite. However it is not a rule and users can choose whatever they want as long as the parameters are within range.

When DataExchanger-AT starts connecting – either received "AT+LCONN" command or auto-connect at power up, it will attempt to connect with fast cycle advertising first. Once fast cycle is completed, it starts connecting with slow cycle. If slow cycle is done, it becomes idle. If users want only one cycle, they can set either cycle duration to zero. Also any cycle can run infinitely by setting the duration to 65535 (0xFFFF).

Advertising parameters are persistent and stored in NV flash. To change the parameters, please refer to "AT+LAVP" command in DataExchanger-AT Command Reference for details.

Connect Policy

The current supported connect policy is auto-connect at power up, and connect-after-remote-disconnect which can be set using "AT+LCM" command. Please refer to DataExchanger-AT Command Reference for details. The default for both auto-connect at power up and connect-after-remote-disconnect is on.

System Reset

DataExchanger-AT supports normal system reset and factory setting recovery reset. Please refer to "AT+RST" command and "AT+ORGL" command in DataExchanger-AT Command Reference.

Power Saving Control

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

DataExchanger-AT supports Standby Sleep and Power-Down Sleep using "AT+SLP=2" command and "AT+SLP=1" command, respectively. It also can wakeup from Standby Sleep using "AT+SLP=0" command. For the details of those commands, please refer to DataExchanger-AT Command Reference.

The differences between the three commands are captured in the following table:

Sleep State	AT Command	Consequence
Wakeup	AT+SLP=0	1. Wake up from Standby Sleep state. 2. Can only be issued by the app.
Power Down Sleep	AT+SLP=1	1. Goes into power down state. 2. Current consumption is ~0.1uA for CC26xBPA standalone module 3. RAM content has no retention 4. No "OK" response. 5. After SWKUP pin triggered, it will follow through a system reset and a "+RDY" notification will be sent.
Standby Sleep	AT+SLP=2	1. Goes into standby state. 2. Current consumption is ~300uA for CC26xBPA standalone module (depending on CI and AI settings) 3. RAM content has retention 4. An "OK" will be responded if no error. 5. After SWKUP pin triggered or AT+SLP=0 command, "+WKUP" notification will be sent.

UART Settings and Flow Control

The following table shows the default UART settings programmed at factory.

UART Parameter	Default
Baud Rate	115200
Data Length	8
Stop Bit	1
Parity	None
Flow Control	No

UART settings are persistent and stored in NV flash. To change UART setting, please refer to "AT+UART" command in DataExchanger-AT Command Reference. Change of setting is required a system reset to take into effect. Please refer to the System Reset section for command line reset.

Hardware Flow control is turned off by default. If host MCU doesn't support hardware flow control, it can use command flow control to ensure no data overflow from host MCU to DataExchanger direction. Host MCU should send a "AT+RBUF?" command and read the response which contains the current free buffer

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

size. Host MCU should send bytes not exceeding the free buffer size. "AT+RBUF?" should be called before each sent attempt.

GPIO Setting and Control

Another important feature that DataExchanger-AT supports is GPIO control. CC26xBPA modules (as well as CC26x Runner) has total of 15 I/O pins. Each I/O pins can be assigned with different peripheral function – GPIO is one of the peripheral function, UART is another one, SPI, I2C, PWM, 1-wire are also the possible peripheral functions. When DataExchanger-AT powers up, it will always allocate the pins for UART first. The rest of the pins are free for user application to grab by executing the corresponding peripheral setting function.

DataExchanger-AT provides the following commands to set and control GPIO pins:

GPIO Function	AT Command
GPIO Allocation and Remove	AT+GIOC
GPIO Direction and Interrupt Configuration	AT+GIOD
GPIO Input Query	AT+GIOI
GPIO Output Set and Query	AT+GIOO

Please refer to DataExchanger-AT Command Reference for details.

GPIO Input Interrupt

DataExchanger-AT allows user application to set up input pins with interrupt enabled (or disabled). Interrupt can be specified with positive edge trigger, negative edge trigger or both. When an interrupt is detected, DataExchanger-AT will generate a "+GIO" notification with its pin index. Host MCU can watch for this notification to react further.

GPIO input interrupt cannot be setup together with wake up function. If user application wants sleep and wakeup capable interrupt, use SWKUP hardware pin instead. To setup SWKUP pin, please refer to System I/O Pin Configuration section.

I2C Setting and Control

DataExchanger-AT provides the following commands to set and control I2C peripheral interface:

I2C Function	AT Command
Enable (or disable) I2C port	AT+I2CC
Read and/or write to I2C port	AT+I2CRW

Please refer to DataExchanger-AT Command Reference for details.

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.

Examples

The following table shows some examples how an I2C EEPROM can be accessed.

Examples	AT Command Sequences
Create I2C port using default the default SDA and SCL pins.	`AT+I2CC=0,1` `>OK`

Find out which pins I2C port is created on.
• Port 0 is enabled and its SDA is on pin 4 and its SCL pin is on pin 5
	`AT+I2CC?`
	`>+I2CC:0,1,4,5`
	`>OK`

EEPROM AT24Cxx from Atmel as example
• I2C chip address is 0x50
• Read 4 bytes from location 0xABCD
• AT24Cxx will increment the current address pointer by 4 bytes automatically (0xABD1).
	`AT+I2CRW=0,0x50,4,ABCD`
	`>+I2CRW:0x7F6BA,EF 00 11 22`
	`>OK`

Continue the example from above
• Read 4 bytes from the current location (0xABD1)
	`AT+I2CRW=0,0x50,4`
	`>+I2CRW:0x475454,3F FF FF FF`
	`>OK`

Continue the example from above
• Write 0x12 0x34 0x56 0x78 at location 0xABCD
• Read 4 bytes from location 0xABCD
	`AT+I2CRW=0,0x50,0,ABCD12345678`
	`>OK`
	`AT+I2CRW=0,0x50,4,ABCD`
	`>+I2CRW:0xF6AB67,12 34 56 78`
	`>OK`

Notes

1. I2C port is disabled by default and its configuration is not persistent.
2. Check out `AT+DTP` command for different display formats.

PWM Setting and Control

DataExchanger-AT provides the following commands to set and control PWM:

PWM Function	AT Command
Enable (or disable) PWM port	`AT+PWMC`
Set PWM period and duty cycle	`AT+PWM`

Please refer to DataExchanger-AT Command Reference for details.

Examples

The following table shows some examples how PWM can be set.

Examples	AT Command Sequences

© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics' non-disclosure agreement.

Examples	AT Command Sequences
Create PWM Port 0 using default pin, period and duty cycle.	AT+PWMC=0,1 >OK
Find out which pins PWM ports is created on. • Port 0 is enabled and is on pin 7 • Port 1 is not enabled	AT+PWMC? >+PWMC:0,1,7;1,0,0 >OK
Change PWM Port 0 period to 1KHz and duty cycle = 30%	AT+PWM=0,1000,0x4CCCCCC >OK
Create PWM Port 1 on pin 6 with 10KHz and 50% duty cycle	AT+PWM=0,0x10000,0x7FFFFFFF >OK AT+PWMC=1,1,6 >OK

Notes

1. PWM ports are disabled by default and their configurations are not persistent.
2. To create a PWM port with certain period and duty cycle values, use `AT+PWM` command to set the period and duty cycle before enabling the pwm port.

iBeacon Programming

iBeacon and other BLE beacons can be easily programmed using `AT+LAVP` command. The following table shows how to program the module to some well known iBeacon in the market.

iBeacon Vendor	AT+LAVP Command
Kontakt	AT+LAVP=100,100,30000,65535,0,0201061AFF4C0002158889A8CA0F7E45658D1974C20C4F940000010001C5
Estimote	AT+LAVP=100,100,30000,65535,0,0201061AFF4C000215B9407F30F5F8466EAFF925556B57FE6D0049000AC5

Once iBeacon is set, it cannot be connected with the DataExchanger app. To revert the settings, use `AT+ORGL` command to perform a factory reset.

Other Supporting Commands

The following table shows the list of other support functions:

AT Command	Type	
AT+EC/AT+EC?	Set/Query	Echo (or not echo) any input character to the output interface.
AT+VS?	Query	Show hardware and software versions
AT+NM?	Query	Report the local info of DataExchanger-AT device.
AT+LADR?	Query	Report the local Bluetooth address
AT+DTP/AT+DTP?	Set/Query	Select the data presentation mode for '`+LRCV`' notification. Support presentation modes are: binary, compact hex, and formatted hex.

Please refer to DataExchanger-AT Command Reference for details.

© GT-tronics (H.K.) Ltd 2017
This material is subject to GT-tronics' non-disclosure agreement.

© GT-tronics (H.K.) Ltd 2017

This material is subject to GT-tronics' non-disclosure agreement.