Product Specification

Class2 Bluetooth v2.1 Flash Module

[Generic Version]

BT-1022

Version: 1.0

Revision History

Edition #	Reason for revision	Issue Date	Written by
1.0	Initial Document	2011/11/04	Kevin Shen

Copyright Notice

Trademarks

AtechOEM is a registered trademark of AtechOEM Technology Co., Ltd.

Disclaimer

Information in this document is subject to change without notice and does not represent a commitment on the part of AtechOEM. AtechOEM provides this document “as is,” without warranty of any kind, either expressed or implied, including, but not limited to, its particular purpose. AtechOEM reserves the right to make improvements and/or changes to this manual, or to the products and/or the programs described in this manual, at any time. Information provided in this manual is intended to be accurate and reliable. However, AtechOEM assumes no responsibility for its use, or for any infringements on the rights of third parties that may result from its use. This product might include unintentional technical or typographical errors. Changes are periodically made to the information herein to correct such errors, and these changes are incorporated into new editions of the publication.

Technical Support Contact Information

If you encounter any technical issues while using BM-1060, do not hesitate to contact us @AtechOEM. Our technical staff will help you resolve the technical issues. You can contact us by email or phone. The following is our technical contact:

- Hours: 9:30AM to 5:30PM (GMT+08:00)
- Email: wifi.support@atechtpe.com.tw
- Phone: +886.2.2377.0282

CONTENTS

1. Overview

AtechOEM BT-1022 module is highly tuned surface mount PCB module that provides fully embedded, ready to use Bluetooth wireless technology. It is also a field-proven software programmable compact **CSR BC04** external Bluetooth v2.1 module. With independent CPU and flash memory, the Bluetooth protocol stack is put on the module such as **SPP** or **HID**. The module makes adding Bluetooth connectivity easy because the host system does not need to handle the Bluetooth protocol stack. It also could be configured as a **HCI** Bluetooth module. With the HCI module, a host system can use its own Bluetooth protocol and profile to complete its own application.

■ SPP (Serial Port Profile)

SPP (Serial Port Profile) is used by devices using Bluetooth for RS232 (or similar) serial cable emulation. It is the most popular profile used in many kinds of applications. Integrators can easily implement serial cable replacement applications in their legacy wired applications.
Since all Bluetooth protocol stacks are managed by the module, Bluetooth connectivity can be easily added to devices with 8- or 16-bit processors, eliminating the need to port existing applications to a more complicated platform and resulting in fastest time to market and reduced development costs.

■ HID (Human Interface Device Profile)

HID profile shall be used by Bluetooth Human Interface Devices, such as keyboards, pointing devices, gaming devices, and remote monitoring devices. It adopts USB definition of Human Interface Device in order to leverage the existing class drivers for USB HID devices.

For users to configure and control the module from a PC or host processor, there are two options of standard firmware available. They are **AT command interface** and **PIO control interface** options. In addition to standard firmware on BT-1021, AtechOEM also offer **customized firmware services** to meet specific applications more tightly.

■ AT command interface

AT command interface provides users a fully flexible control to BT module through a UART interface. Users can use AT command instructions to create Bluetooth wireless connectivity and configure many Bluetooth parameters into the BT module. It's suitable that the BT module want to act as either a master or slave BT devices.

■ PIO control interface

PIO control interface provides an extremely easy method to create a BT connectivity between BT module and a remote BT device. Users only need to measure few GPIO's level and pull high/low them to control

the behaviour of the BT module. It's suitable that the BT module only acts as a slave Bluetooth device to listen to the connection request from a remote Bluetooth device.

■ HIC Mode

Host Controller Interface (HCI) is a standard interface that a host system can access Bluetooth's directly. BT-1022 module could be configured in HCI mode, which provides a command reference interface to the baseband controller and the link manager, and provides access to the hardware status and control registers.

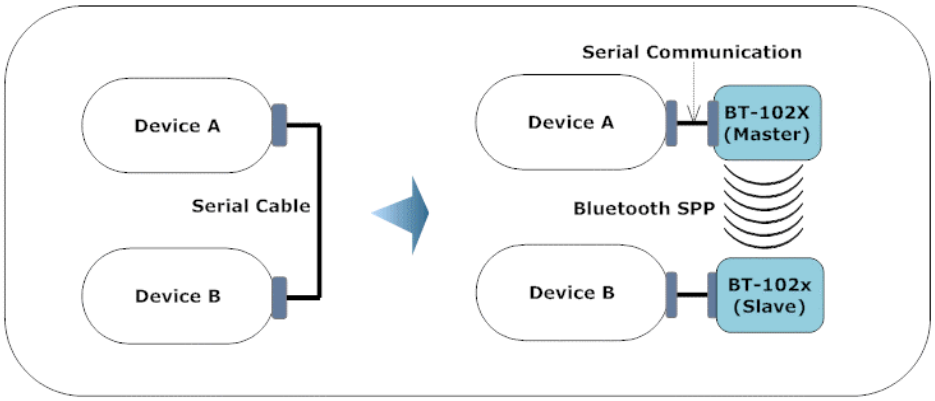
In this mode, the Bluetooth stack is no longer existed in module itself. A host system needs to use its own Bluetooth stack/driver to access the BT module. The Bluetooth module is regarded as a radio transceiver. Using this module in HCI mode allows users to use standard profiles or proprietary methods to transceiver the related data stream among Bluetooth devices.

BT-1022 provides two HCI interfaces to a host processor – UART and USB. **HCI-over-UART** allows a host processor to access BT module through an UART interface. An USB interface is used between BT-1022 and a host processor. In **HCI-over-USB**, the BT module is the USB slave and the host processor is the USB host.

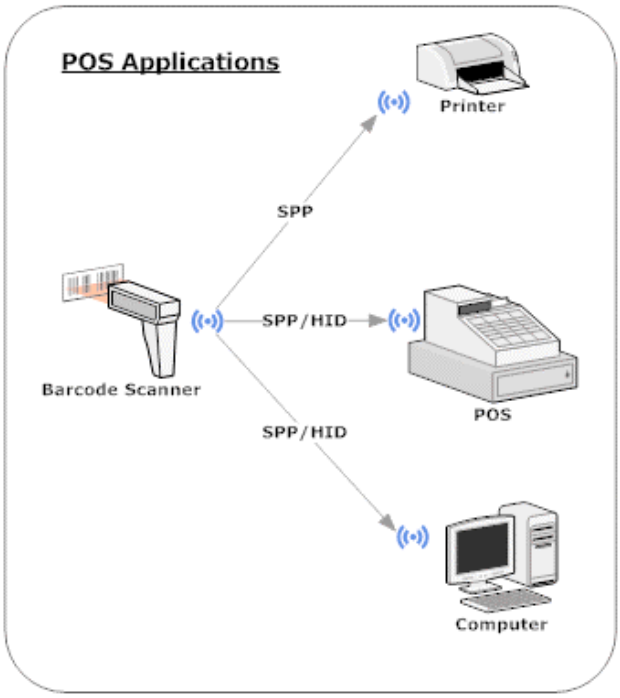
2. Typical Applications

- Asset tracking
- Measurement/ Remote Equipment Monitoring
- Scanners
- Sensors & Controls
- Industrial devices
- Point-of-Sale systems
- Automotive Diagnostics Units
- Handheld terminals

■ Cable Replacement Logical Diagram



■ Barcode Scanner & POS Applications Diagram



3. Product Information

■ Product Number : **BM-1022**

■ Product Description: **Bluetooth Class2 v2.1+EDR Flash Module**

■ Product Features:

Chip	CSR BlueCore-04 External
Standard	Bluetooth v2.1 + EDR
Flash	8MB
RF band	2.4~2.4835GHz ISM band
Host Interface	UART / USB
Digital Interface	PCM / GPIOs
RF Output Power	Class II (<4dBm)
Sensitivity	<-80dBm
Antenna	External antenna
Power voltage	3.3V
Dimension	21.8mm x 15mm x 3mm

■ Product photo :

4. Hardware

BT-1022 class 2 is a surface-mount module designed to be integrated to a system board as a wireless subsystem or standalone system. The interface is grouped into the following functional blocks:

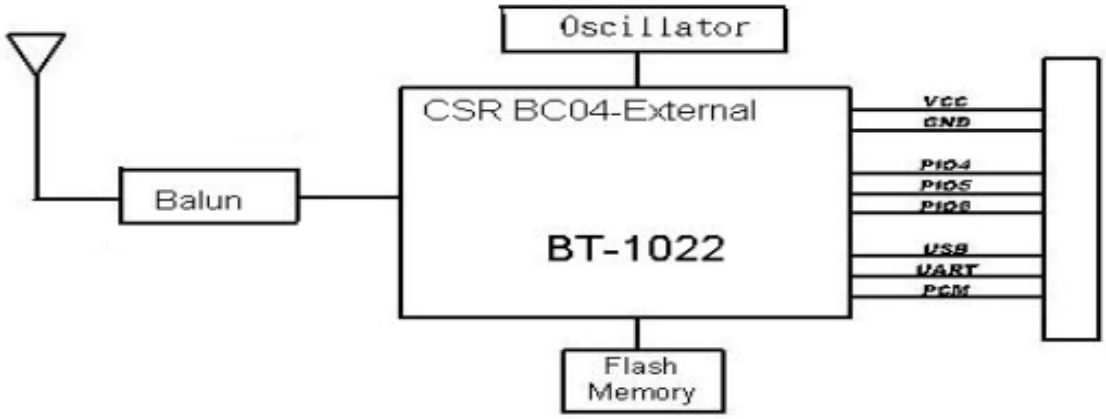
- Power supplies: VCC line should be connected to 3.3VDC.
- GPIOs: The PIO lines can be configured as general purpose 3.3V level digital inputs or outputs.
- UART: UART_TX, UART_RX, UART_CTS and UART_RTS are UART communication interface signals. UART_CTS and UART_RTS are not necessary to be connected if hardware flow control is not used.
- USB: USB interface is used for HCI mode only.
- PCM: PCM interface is used for audio codec. Special software programming support is required.
- SPI: SPI interface can be set to master or slave. This is software programmable. Special software programming support is required.
- RF: The RF_OUT should be connected to the antenna.

■ Hardware Integration

The module requires only 5 connections to power up and create a Bluetooth wireless connection. In the simplest configuration, the hardware only needs the following connections: VCC, RF_OUT, UART-TX, UART-RX, and GND. VCC and GND pins connect to a regulated 3.3VDC power supply. The RF_OUT connects to a 2.4GHz antenna. The user connects UART-TX, UART-RX and ground pins to the host controller or a PC. The host controller or a PC can control the module and set up Bluetooth connection with the commands and settings defined in this document.

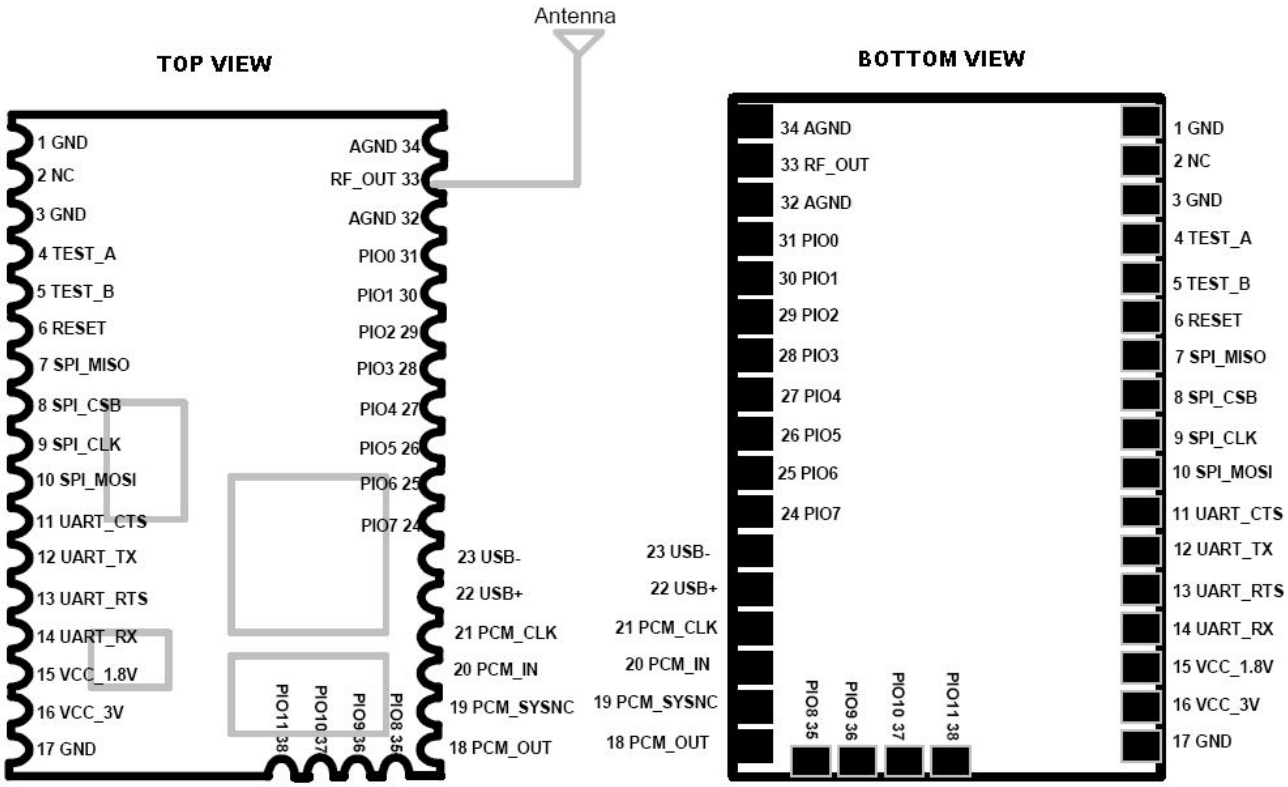
The user can also use other functional blocks like GPIO, USB, SPI and PCM interfaces. Special software support is required to use these interfaces.

4.1. Block Diagram



4.2 Pin Assignment:

The following picture shows pinouts of BT-1022 from the top and bottom of the module.

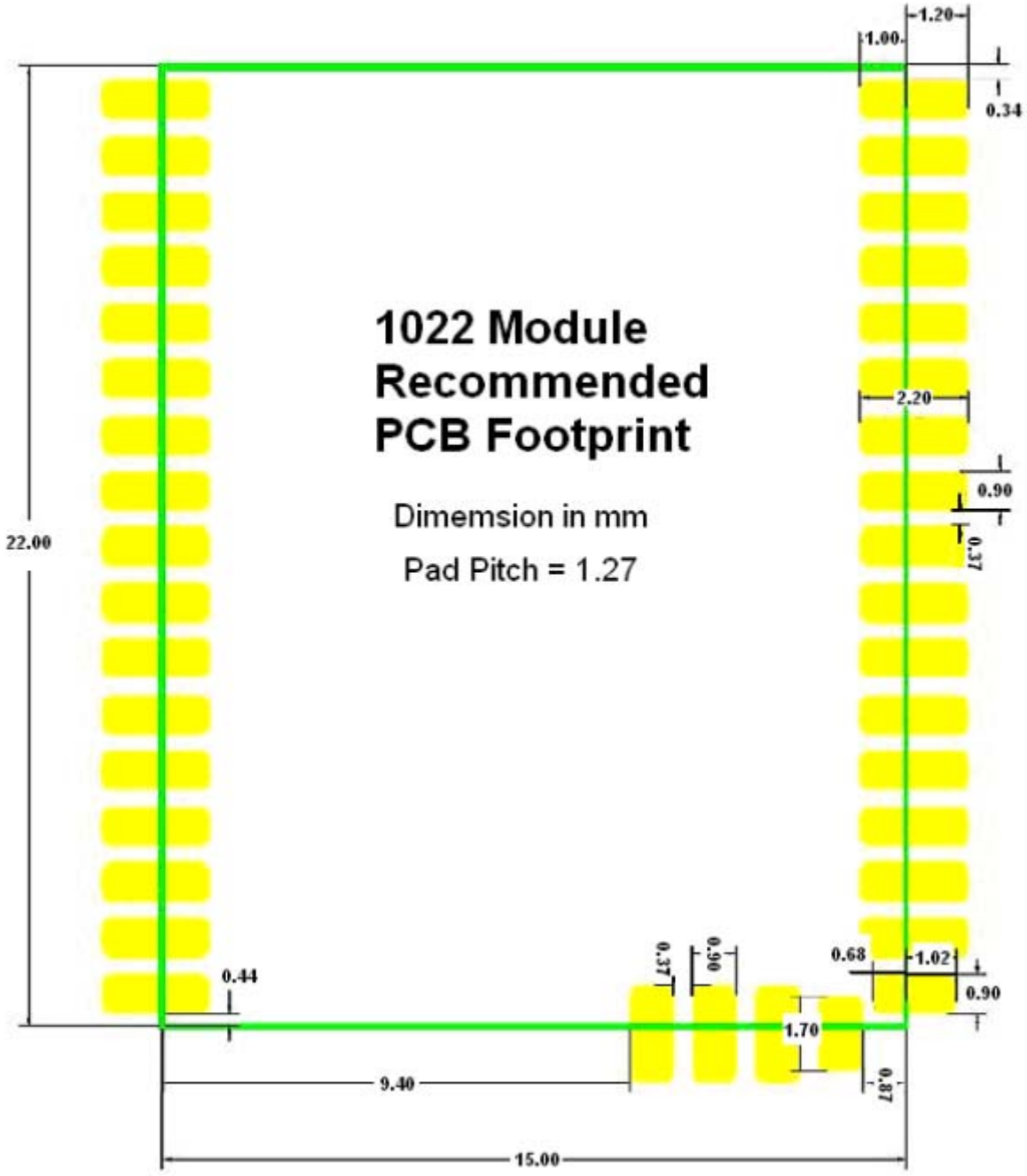


Pin	Name	Type	Note
1	GND		
2	NC		
3	GND		
4	TEST_A		Internal Testing Use
5	TEST_B		Internal Testing Use
6	RESET	In	1. It should be pulled low for normal operation (Active High) 2. A Reset will be performed after PIN6 being active for 1.5~4.0 ms
7	SPI_MISO	In	Internal Testing Use
8	SPI_CSB	In	Internal Testing Use
9	SPI_CLK	In/Out	Internal Testing Use
10	SPI_MOSI	Out	Internal Testing Use
11	UART_CTS	In	
12	UART_TX	Out	Connect to host processor UART RX pin
13	UART_RTS	Out	
14	UART_RX	In	Connect to host processor UART TX pin
15	VCC_1.8V	Out	
16	VCC_3.3V	In	Main power supply
17	GND		
18	PCM_OUT	Out	Codec Interface
19	PCM_SYNC	In/Out	Codec Interface
20	PCM_IN	In	Codec Interface
21	PCM_CLK	In/Out	Codec Interface
22	USB+	In/Out	
23	USB-	In/Out	
24	PIO7	In/Out	General Purpose I/O
25	PIO6	In/Out	General Purpose I/O
26	PIO5	In/Out	General Purpose I/O
27	PIO4	In/Out	General Purpose I/O
28	PIO3	In/Out	General Purpose I/O
29	PIO2	In/Out	General Purpose I/O
30	PIO1	In/Out	Internal Used
31	PIO0	In/Out	Internal Used

32	AGND		Analog GND
33	RF_OUT		Connect to antenna
34	AGND		Analog GND
35	PIO8	In/Out	General Purpose I/O
36	PIO9	In/Out	General Purpose I/O
37	PIO10	In/Out	General Purpose I/O
38	PIO11	In/Out	General Purpose I/O

4.3. Dimensions

Use the dimensions shown in the picture below to do the PCB layout for the host system or carrier board that will use BT-1022.



4.4. Electrical Characteristics

	Min	Typ.	Max.	Unit
Supply Voltage	3.0	3.3	3.6	V
RX Supply Current	-	44	-	mA
TX Supply Current(Normal)	-	39	-	mA
TX Supply Current(Continuous)	-	50	-	mA
Sniff Mode Current	-	6	-	mA

■ Operating Conditions

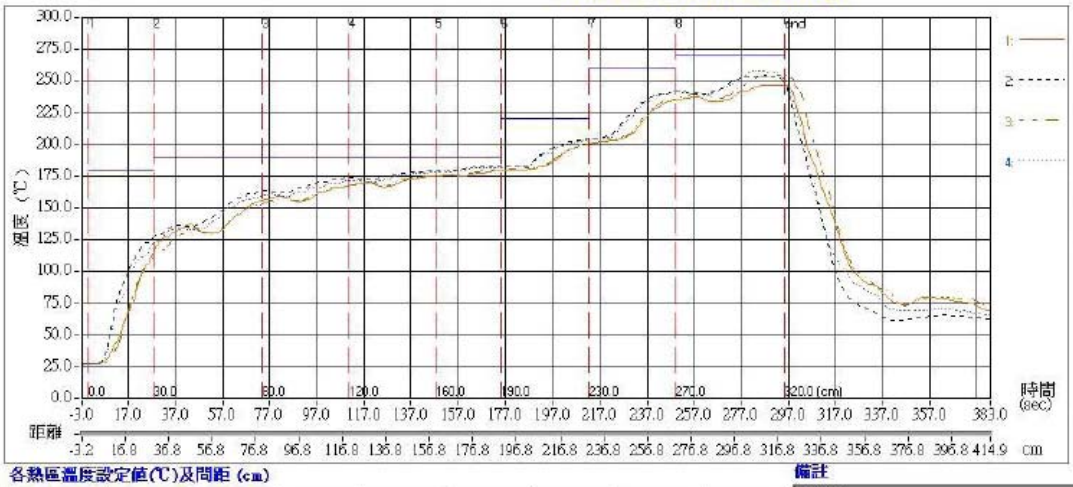
Voltage Range	3.3V±0.3V
Operating Temperature Range	-20 °C ~ 70 °C
Storage Temperature Range	-40 °C ~ 85 °C
Relative Humidity (Operating)	≤90%
Relative Humidity (Storage)	≤90%

4.5. Radio Characteristics

	Frequency (GHz)	Min	Typ	Max	BT Spec.	Unit
Sensitivity at 0.1%BER	2.402	-	-86	-80	≤ -70	dBm
	2.441	-	-86	-80		dBm
	2.480	-	-86	-80		dBm
RF Transmit Power	2.402	-6	0	4	≤4	dBm
	2.441	-6	0	4		dBm
	2.480	-6	0	4		dBm
Initial Carrier Frequency Tolerance	2.402	-	5	75	75	kHz
	2.441	-	5	75		kHz
	2.480	-	5	75		kHz
20dB bandwidth for modulated carrier		-	900	1000	≤ 1000	kHz
Drift (Five slots packet)		-	15	-	40	kHz
Drift Rate		-	13	-	20	kHz
$\Delta f1_{avg}$ "Maximum Modulation"	2.402GHz	140	165	175	$140 < \Delta f1_{avg}$	kHz
	2.441GHz	140	165	175		kHz
	2.480GHz	140	165	175		kHz
$\Delta f2_{max}$ "Minimum Modulation"	2.402GHz	115	190	-	115	kHz
	2.441GHz	115	190	-		kHz
	2.480GHz	115	190	-		kHz

4.6. Reflow Profile

THERMOTRACKER 測溫報告



日期(日/月/年): 14/12/2007

公司名稱: Atech

產品名稱: BT1022

速度設定值: 65.00 cm/Min

下載資訊

取樣速率(分:秒): 00:01.0

日期(日/月/年): 12/12/2007

時間(時:分:秒): 13:10:42

資料檔名稱: BT-1022_961212.pwd

各熱區溫度設定值(℃)及間距 (cm)

熱區	1	2	3	4	5	6	7	8
TOP	180	190	190	190	190	220	260	270
BOTTOM	180	190	190	190	190	220	260	270
間距	30.0	50.0	40.0	40.0	30.0	40.0	40.0	50.0

備註

ROHS

最高溫度及時間分析

測溫點名稱	最高溫度 (℃)	位於 (秒)	高於150.0℃ 的時間(秒)	高於190.0℃ 的時間(秒)	高於220.0℃ 的時間(秒)
n2	247.3	298.00	248.00	107.00	67.00
U1	254.3	292.00	253.00	114.00	72.00
n2	254.6	295.00	248.00	111.00	70.00
U1	258.0	287.00	251.00	114.00	72.00

測溫點位置 X mm , Y mm

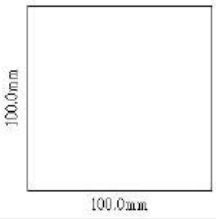
1: ------ , ------

2: ------ , ------

3: ------ , ------

4: ------ , ------

工件 移動方向



5. Software & Operation Mode

The special firmware in BT-1022 supports two different operation modes: **PIO mod**e and **AT-Command mode**. Both modes support SPP (Serial Port Profile) as its application connection. These two modes operate separately based on different firmware provided by AtechOEM.

5.1. PIO Operation Mode

BT-1022 in PIO operation mode is controlled by hardware GPIOs. PIO control interface provides an extremely easy method to create a BT connectivity between BT module and a remote BT device. Users only need to measure few GPIO's level and pull high/low them to control the behaviour of the BT module. It's suitable that the BT module only acts as a slave Bluetooth device to listen to the connection request from a remote Bluetooth device.

Two sub-modes existed in PIO mode, which are **configuration** and **operation** mode. The configuration mode is used to set the UART setting. In the operation mode, BT module can be connected by a remote BT device. Baud rate of the UART is also configured in operation mode. **PIO7** is used to switch these two modes while powering/resetting the BT module.

■ **PIO Definition in PIO mode:**

Pins	Type	Function	Descriptions
PIO 2	(In)	Configure Baudrate or UART settings	Described below
PIO 3	(In)	Configure Baudrate or UART settings	Described below
PIO 4	(In)	Configure Baudrate or UART settings	Described below
PIO 5	(In)	Configure Hardware Flow Control or DTE/DCE	Described below
PIO 6	(Out)	LED	Blinks once per second: BT is in **discoverable/disconnected** status Blinks twice per second: BT is in **connected** status If in **configuration** mode, the LED is always ON.
PIO 7	(In)	Reset Button	1. To disconnect the connection from peer device; Drive high for at least 30 ms to enable the disconnecting procedure. 2. To switch configuration/operation modes while power on reset
PIO 8	(In/Out)	DTR	Input for DTE / Output for DCE
PIO9	(In/Out)	DSR	Output for DTE / Input for DCE
PIO10	(In/Out)	DCD	Output for DTE / Input for DCE
PIO 11	(In/Out)	RI	Output for DTE / Input for DCE

■ PIO Definition in "Configuration mode": (PIO7 is high while power-on)

The following table is used to configure UART setting.

Voltage Level / PIO	PIO2	PIO3	PIO4	PIO5
Low	One stop bit	No parity bit	Odd parity	DTE
High	Two stop bits	One parity bit	Even parity	DCE

■ PIO Definition in "Operation mode": (PIO7 is low while power-on)

The following table is used to configure the baud rate of UART.

Voltage Level of PIO			
PIO4	**PIO3**	**PIO2**	**Baud Rate of UART (bps)**
Low	Low	Low	115200
Low	Low	High	57600
Low	High	Low	38400
Low	High	High	19200
High	Low	Low	9600
High	Low	High	4800
High	High	Low	2400
High	High	High	1200

The following table is used to configure the hardware flow control of UART

Voltage Level / PIO5	Hardware Flow Control
Low	On
High	Off

■ Default Configurations:

Bluetooth		UART	
Mode	Slave	**Baud Rate**	115200 bps
Profile	SPP	**Data Bits**	8
Device Name	BT SPP + BD ADDR (Lower)	**Parity**	No
PIN Code	"1234"	**Stop Bits**	1

5.2. AT-Command Operation Mode

BT-1022 in AT-Command operation mode is controlled by instruction method – AT Command. AT command interface provides users a fully flexible control to BT module through a UART interface. Users can use AT command instructions to create Bluetooth wireless connectivity and configure many Bluetooth parameters into the BT module. It's suitable that the BT module want to act as either a master or slave BT devices.

Two sub-modes existed in AT-Command mode, which are **Command** and **Data** mode. When BT module is not connected to any peer BT devices, BT module is in command mode. In command mode, users can operate and configure BT module through AT-commands. After SPP connection created, BT module could start to transceiver data from the UART interface to/from the SPP connection (radio interface). There are few simple settings could be configured by hardware PIOs refer to the below table.

■ **PIO Definition in PIO mode:**

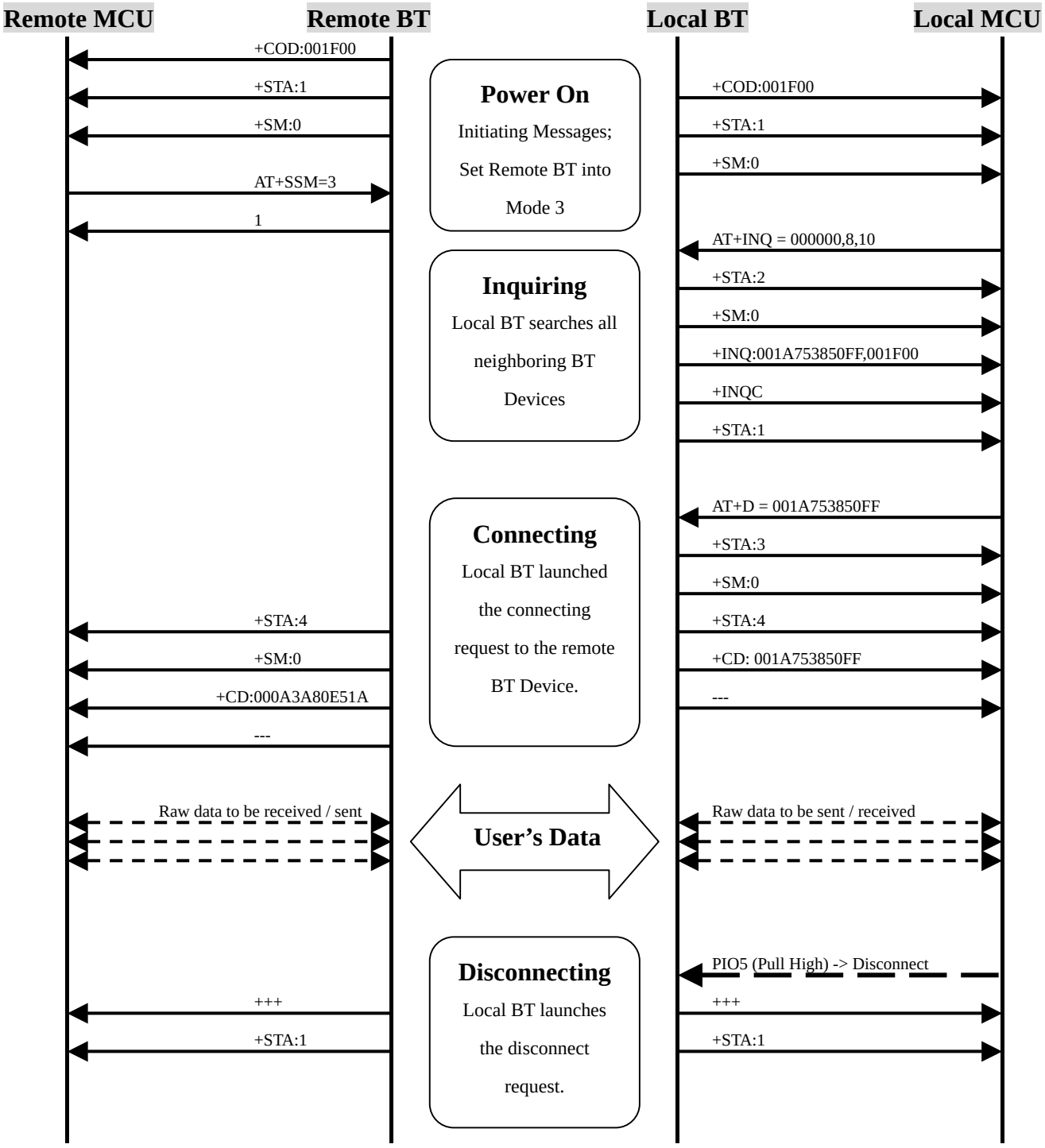
Pins	Type	Function	Descriptions
PIO 2	(Out)	CONNECTION_IND	Connected: High, otherwise: Low
PIO 3	(In)	SWITCH_HW_FC	Connected to VDD to enable CTS/RTS FC, otherwise left not connected
PIO 4	(NA)	--	--
PIO 5	(In)	BUTTON_DISCONNECT	Pulled high to disconnect SPP connection
PIO 6	(In)	BAUD_RATE_115K2	Pulled high to reset baud rate to 115200 bps
PIO 7	(In)	BAUD_RATE_9K6	Pulled high to reset baud rate to 9600 bps
PIO 8	(NA)	--	--
PIO9	(NA)	--	--
PIO10	(NA)	--	--
PIO 11	(NA)	--	--

■ **Default Configurations:**

Bluetooth		**UART**	
Mode	Master/Slave	**Baud Rate**	9600 bps
Profile	SPP	**Data Bits**	8
Device Name	BT SPP	**Parity**	No
PIN Code	"0000"	**Stop Bits**	1

5.3. SPP Connection Example in AT-command mode

The following example describes simple initial, inquiring, connecting, data-sending/receiving and disconnecting process by using AT Command set in BT-1022.



Appendix A: AT Command

The modem-like AT Command set of BT-1022 is based on Command/Response handshake. The following sections describe syntax and commands used in command interface for the user to configure and control BT-1021.

A.1. AT Command Format

The AT command format is:

AT<cmd><cr> or **AT<cmd>=<value><cr>**

******* **<cmd>** represents ***specified command;*** **<cr>** represents **\r**

Format	Example	Example
AT+<cmd><cr>	AT + GLN \r	Host send Get local name command
AT+<cmd>=<value><cr>	AT + PIN = 12345678 \r	Host send Set pin code command

A.2. AT Command Response Format

The AT command response format is:

<cr><lf><Response Code><cr><lf>

*** **<cr>** represents **\r**; **<lf>** represents **\n**

There are two categories of Response Code:

■ **Category 1 Response Code:**

Example	Meaning	Condition & Notes
\r \n 1 \r \n	OK	
\r \n 2 \r \n	ERROR	ie, AT+INQ = 000000,**15**,10 \r (1 <= maxResponses <= **10**)
\r \n 3 \r \n	UNKNOWN	ie, AT+**ABC** \r (command ABC is not available)

\r \n 4 \r \n	**FAILED**	ie, command/response successfully finished; result to be no match

■ Category 2 Response Code:

Example	Meaning	Condition & Notes
\r \n +STA:state \r \n	**STATE:** **to indicate current status**	**state:** 1 : sppDevReady 2 : sppDevInquiring 3 : sppDevConnecting 4 : sppDevConnected
\r \n +SM:scanMode \r \n	**SCANMODE:** **to indicate current scan mode**	**scanMode:** 0 : hci_scan_enable_off 1 : hci_scan_enable_inq 2 : hci_scan_enable_page 3 : hci_scan_enable_inq_and_page
\r \n +INQ:address, classOfDevice \r \n	**INQUIRY result**	**address:** 12 hexadecimal **classOfDevice:** 6 hexadecimal
\r \n +INQC \r \n	**INQUIRYCOMPLETE**	
\r \n +COD: classOfDevice \r \n	to indicate class of device (local device)	**classOfDevice:** 6 hexadecimal
\r \n +LA:address \r \n	LocalAddress	address: 12 hexadecimal
\r \n +LN:name \r \n	Local Name: Return code of get local name command	User friendly name of the local device
\r \n +RN:address,name \r \n	Remote Name: Return code of get remote name command	**address:** 12 hexadecimal **name:** user friendly name of the remote device
\r \n +SSPP:address,result \r \n	SEARCHSPP : Return code of search SPP profile command	**address:** 12 hexadecimal **Result:** 0 : SPP profile not found 1 : SPP profile found
\r \n +CD:address \r \n	CONNECTED : Indication of SPP connection	address: 12 hexadecimal
\r \n --- \r \n	Connected : The last result code before entering data mode	

\r \n +++ \r \n	Disconnected : The first result code after returning to command mode	

A.3. Set Commands

The Set commands are for the user to set the BT-1021 configuration.

SET Commands	Command Set	Conditions & Notes
Set inquiry scan and page scan mode command	**AT + SSM = scanMode \r**	**scanMode:** 0 : hci_scan_enable_off 1 : hci_scan_enable_inq 2 : hci_scan_enable_page 3 : hci_scan_enable_inq_and_page
Set pin code command	**AT + PIN = pinCode \r**	**pinCode:** default: 0000 4< pincode length <16
Set local name command	**AT + SLN = localName \r**	1 <= local name length <= 16
Set Baud rate command	**AT + SBR = baudRate \r**	**baudRate:** **Default = 9k6 bps** UART Baud rate =baudRate/0.004096 Some common values are: 9k6 baud - 39 19k2 baud - 79 38k4 baud - 157 57k6 baud - 236 115k2 baud - 472 230k4 baud - 944 460k8 baud - 1887 921k6 baud - 3775 1382k4 baud - 5662
Set class of device command	**AT + SCOD = classOfDevice \r**	**classOfDevice:** 6 hexadecimal class of device

Delete all authorized devices command	**AT + DA \r**	

A.4. Get Commands

GET Commands	Command Set	Conditions & Notes
Get inquiry scan and page scan mode command	**AT + GSM \r**	
Get local name command	**AT + GLN \r**	
Get class of device command	**AT + GCOD \r**	
Get local address command	**AT + GLA \r**	
Get remote name command	**AT + GRN = address \r**	**address:** 12 hexadecimal Bluetooth Device Address
Get state command	**AT + STA \r**	

A.5. Action Commands

The actions commands are for the users to control the BT-1021 to do certain actions. The defined actions are: Inquiry, Connect, Reset and Search SPP profile.

ACTION Commands	Command Set	Conditions & Notes
Inquiry command	**AT + INQ = CLASSOFDEVICE, MAXRESPONSES, TIMEOUT \R**	**classOfDevice:** 6 hexadecimal class of device filter set 000000 -- notified of all devices within range regardless of their class of device. audio/ video to be 000400 ; headset to be 200404 **maxResponses:** 1 <= maxResponses <= 10 **timeout:**

		1(1.28 seconds) <= timeout <= 48(61.44 seconds)
Connect command	**AT + D = address \r**	**address:** 12 hexadecimal Bluetooth Device Address
Reset command	**AT + RST \r**	
Search SPP profile command	**AT + SSPP = address \r**	**address:** 12 hexadecimal Bluetooth Device Address