

PSoC4-BLE Integrated Test Code User Manual

Purpose:

The purpose of this document to explain the usage and features of the PSoC4-BLE Integrated test code (ITC)

Features:

The firmware supports the following test modes

1. MTK mode (AT like commands)
2. BLE two wire mode (Only 1 Mbps supported for now)
3. Regulatory test modes (button press based mode change)

Option 1 and 2 are recommended for regulatory tests if there is an UART port available, otherwise Option 3 can be used.

For qualification test and for testing using and standard Bluetooth testers, it is recommended to use the Direct Test Mode example project (which basically provides HCI over UART)

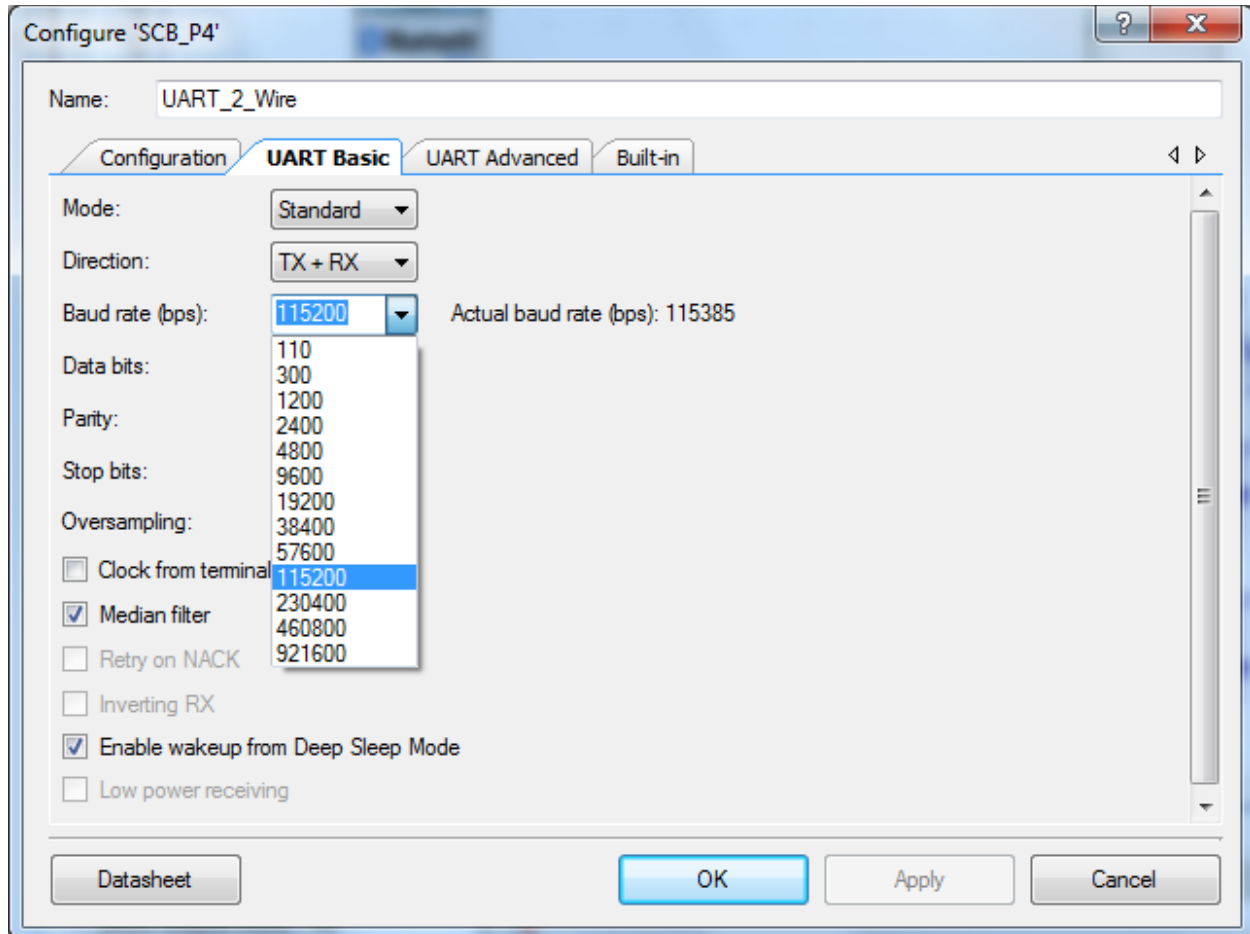
BLE two wire mode:

This mode supports the 2 wire commands defined in the BLE specifications. This mode can be used with most test equipments to test RF-PHY.

Select the port pins of the UART component to match the target hardware to use 2-wire test mode.

<input type="checkbox"/>	\UART_2_Wire:rx\ 	P6[0]	▼	26	▼	<input checked="" type="checkbox"/>
<input type="checkbox"/>	\UART_2_Wire:tx\ 	P6[1]	▼	28	▼	<input checked="" type="checkbox"/>

The baud rate is set to 115200 bps by default. It can be changed by double clicking the UART component from the TopDesign.cysch file and choosing the desired baud rate from the drop down menu in the baud rate field.



Regulatory Test Mode

This mode can be used for regional regulatory testing like FCC, IC, CE, KC, MIC, etc,. This mode supports putting the Bluetooth device into various transmit and receive modes needed for regulatory testing. A button trigger is used to switch between various modes. Please configure the GPIO named Button to any available button/GPIO in the target hardware.

The radio starts in idle mode and cycles through the mode defined with each button press. The suffix indicates the frequency in MHz, RX indicates reception, TX

indicates modulated packet (burst) transmission, TXC indicates carrier wave transmission mode. The device operates in 1 Mbps mode for the first cycle. After completion of all the modes, the device restarts from `DUT_RX_2402` in 2 Mbps mode.

```
typedef enum
{
    DUT_IDLE,
    DUT_RX_2402,
    DUT_RX_2440,
    DUT_RX_2480,
    DUT_TX_2402,
    DUT_TX_2440,
    DUT_TX_2480,
    DUT_TXC_2402,
    DUT_TXC_2440,
    DUT_TXC_2480,
    DUT_TX_HOPPING
}DUTStates;
```

MTK Test Mode

MTK mode allows users without sophisticated test equipment to put the put the DUT in any of the test modes simply by typing simple commands from HyperTerminal or Teraterm (or any serial port console)

MTK can be disabled by commenting `#define MTK` in Config.h

The following are the list of supported commands

Command	Parameters	Comment
TXP	Channel, Power, Number of packets	Transmits the specified number of DTM packets in the specified channel at specified output power level in dBm.
RXP	Channel	Puts the device in DTM receive mode in the specified channel
RRS		Stops ongoing DTM mode
PST		Prints the current status
SPL	Packet length	Sets the length of the DTM packet
SPT	Payload type	Sets the payload type of the DTM packet
TXC	Channel, power, duration	Transmit carrier wave in the specified channel at specified power level in dBm for the specified duration
ADV	Channel, Interval	Transmits a single DTM packet once every 'Interval' number of BT slots.
2WC	MTK_ON/OFF	This is available only when both Two Wire

		and MTK are enabled. TwoWire has the control on UART by default. This command with parameter 0 should be sent to change the control to MTK or parameter 1 to give the control to TwoWire when MTK has the control.
RXC	Channel, RX_Window	Resets receiver in the specified Channel for RX_Window number of packets and prints the number of packets. This is repeated until another MTK valid command is issued
SRP	Phy	Set the radio PHY to either 1 mbps or 2 mbps
TXM	Channel, power, duration	Transmit continuous modulated output in the specified channel at specified power level in dBm for the specified duration

The following table explains the valid values for each parameter

Parameter	Valid inputs	Default	Comments
Channel	0 to 39	0	BLR RF-PHY channel
Power	-18, -12, -6, -3, -2, -1, 0, 3	-18	Transmit power in dBm
Num. of packets	-1 to 65535	1000	-1 for indefinite transmission
Packet length	0 to 255*	37	255 bytes is supported only in devices with Bluetooth 4.2 support; otherwise, the maximum length is 37
Payload type	0 to 7	0	0 Pseudo-Random bit sequence 9 1 Pattern of alternating bits '11110000' 2 Pattern of alternating bits '10101010' 3 Pseudo-Random bit sequence 15 4 Pat-tern of All '1' bits 5 Pattern of All '0' bits 6 Pattern of alternating bits '00001111' 7 Pattern of alternating bits '0101'
Duration	-1 to 2147483647	-1	-1 for indefinite transmission Any other value is time in ms
Interval	32 to 16384	32	Specified number of BT slots (625uS) plus a random delay.
MTK_ON/OFF	0 1	1	1 – Two Wire owns the UART port 2 – MTK owns the UART port
RX_Window	100 to 65535	1600	Timer is set to receive the number of packets

			specified for each update of number of packets received
Phy	1 or 2	1	1-1 Mbps, 2-2Mbps

Examples:

TXP 19 3 1500 – transmits 1500 packets in Channel 19 at 3 dBm power level

RXP 39 - starts receiving DTM packets in Channel 39

RRS – stops DTM mode; prints the number of packets transmitted when issued after TXP; prints the number of valid packets received when issued after RXP.

PST - prints the number of packets transmitted when issued after TXP; prints the number of valid packets received when issued after RXP.

SPL 37 - sets the packet length to 37 bytes

SPT 0 - sets the packet type to PRBS9 sequence

TXC 0 3 1000 - transmits unmodulated carrier wave in channel 0, at 3 dBm for 1000 milliseconds

TXC 0 3 1000 - transmits modulated carrier in channel 0, at 3 dBm for 1000 milliseconds

SRP 2 - Sets the PHY to 2 Mbps

ADV 39 32 - transmits one DTM packet in Channel 39 every 32 BT slots

RXC 10 1600 - Starts the receiver in Channel 10, sets timer to receive 1600 packets (1600*0.625 us), stops the receiver after the timer interrupt, prints the actual number of packets received. This is repeated until the next MTK command is received.