SPI Host-Controller Interface BRF6300 Application Note

BT-AN-0058
Revision 0.5

20 November 2006

Abstract

This document describes the SPI Host-Controller interface:

- Supported Features
- Modes of operation
- Shared SPI bus Mode

Revision Control

Author Name	Description	Revision	Date
Dana Ram	Creation	0.1	25 January 2006
Dana Ram	updates	0.2	
Dana Ram	Added Section 8	0.3	4 April 2006
Sean Block		0.4	31 October 2006
Anthony Levine	Added initial-command timing requirements Added Swapped-Bytes mode	0.5	20 November 2006

Table of Contents

1 Overview

BRF6300 provides an TI SPI (BTSPI) transport layer, in addition to the UART and the SDIO.

SPI stands for Serial Protocol Interface. The SPI interface provides a high speed data transfer with low power consumption for mobile electronic devices. The SPI bus is used as interface between a Host and several different TI devices (WLAN, BT & DTV).

The SPI bus was designed to operate on a point-to-multipoint basis by providing a separate chip select (CS) per device.

2 Supported Features of BTSPI

- Incorporates a deep-sleep protocol
- Transport layer H4
- Point-to-multipoint
- Supports BT basic, EDR2 and EDR3 data rates
- Maximum supported clock rate = 13MHz
- Supports two protocols: Standard BTSPI and Swapped-bytes BTSPI.
- BRF6300 is always SPI Slave, Host is always SPI Master.

3 BRF6300 pinouts for HCI transport Layer

The BRF6300 supports three transport layers which are multiplexed on the same IO pins as depicted in Table 1.

	UART	**SDIO 1-bit**	**SPI**
CLK (I)	**N.U**	CLK (I)	CLK (I)
RX (B)	RX (PU, I)	CMD (B)	DI (I)
CTS (I)	CTS (PU, I)	IRQ (O)	CS (PU, I)
RTS (B)	RTS (PU, O)	IRQ (O)	IRQ (O)
TX (B)	TX (PU, O)	DATA (B)	DO (O)

Table 1 - Transport Layer Pinout Multiplexing:

4 BTSPI Interface Description

In order to facilitate a broad implementation, the protocol is half duplex and does not require simultaneous operation of DO and DI.

All TI communication devices are slaves in this protocol and all transactions are initiated by the Host, as SPI Master.

The clock rate for each one of the connected devices may be different and configured per device.

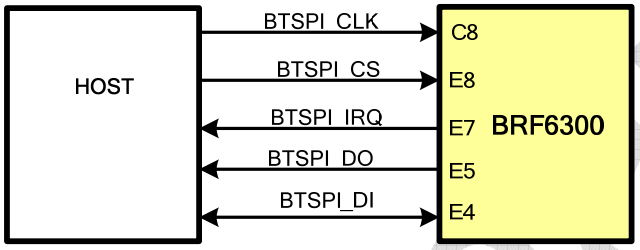

Figure 1 : SPI Interface Signals

Port name	In/Out	Description
BTSPI_CLK	**In**	**Clock (0 to 13MHz) from Host to slave**
BTSPI_DI	**In**	**Data From Host to Slave**
BTSPI_CS	**In**	**CS signal from Host to Slave**
BTSPI_IRQ	**Out**	**Interrupt from Slave to Host**
BTSPI_ DO	**Out**	**Data From Slave to Host**

Table 2 : SPI Interface Signals Description

During reset all outputs will be in high impedance mode and all pull resistors are enabled in that mode.

4.1 CS and bus sharing operation

The CS line is used to select a specific device on the shared SPI bus.

The CS line is asserted at the beginning of an SPI transaction and de-asserted on its completion.

The CS line must not be de-asserted during the transaction.

Bus sharing by multiple devices is implemented by enabling one CS signal at a time and performing transactions with a specific device.

The multiplexing of the devices is performed on a transaction basis rather than on a byte or word basis.

4.2 Sleep protocol

In order to optimize power consumption, the BRF6300 toggles between active state and low power mode (deep sleep).

When operating in low power mode the device deactivates most of its logic and is not capable of full speed transactions with the host. If the device is in sleep mode, the host has to wakeup the device from the sleep mode.

The wakeup protocol is based on assertion of the CS line.

The BT device will perform its wakeup procedure and respond by assertion of the IRQ line.

The CS has to be asserted for a minimum time of 31usec in order for a device operating in sleep mode to recognize it.

Note: If a device is in sleep mode, transactions (read or write) should not be started until the IRQ line is asserted.

The SPI driver within the Host does not have to be aware of the sleep state of the device. It is the responsibility of the BT stack to report to the SPI driver what state is the device is in.

The procedure described above is also utilized as a method for flow control. The BT stack will always signal the SPI driver that the device is in sleep mode. In response to CS assertion the device will assert the IRQ line only when it has enough free buffer space to receive a transaction.

Note:
The delay from assertion of CS to the assertion of the IRQ line may vary depending on the actual state of the device and the delay in activation of the system clock. it is guaranteed that the delay will not exceed 2msec at worst case in order to prevent bus blocking. Please note that in normal operation the delay until assertion of the IRQ line will be significantly shorter and polling of the IRQ line may be beneficial.

4.2.1 Enabling the BTSPI Deep Sleep Protocol

The default setting at power up is that Deep Sleep operation is disabled. In order to enable the BRF Deep Sleep feature and make the Deep Sleep protocol active, the Host must first send an HCI_VS_Sleep_Mode_Configurations command.

Send_HCI_VS_Sleep_Mode_Configurations 0xFD0C, 1, 1, 0x06, 0xFF, 0xFF, 0xFF, 0, 100
Wait_HCI_Command_Complete_VS_Sleep_Mode_Configurations_Event 5000, any, HCI_VS_Sleep_Mode_Configurations, 0x00.

Refer to Section 10.1 for detailed description of this command.

5 BTSPI Transactions

BTSPI has two modes of operation:

- Standard
- Swapped-Bytes

The following describes common requirements and Write/Read operations for each mode.

5.1 Operation requirements common to both Standard and Swapped-Bytes modes

5.1.1 Initial command

The BRF6300 defaults to Standard BTSPI mode after wakeup. Therefore the first command (even before the init script) must be the HCI_VS_TI_SPI_Configuration command, given in Standard format.

To remain in Standard mode:

Send_HCI_VS_TI_SPI_Configuration 0xFD41, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
Wait_HCI_Command_Complete_VS_TI_SPI_Configuration_Event 5000, any, HCI_VS_TI_SPI_Configuration, 0x00

Outgoing hex dump:
01 00 15 00 00 01 41 fd 11 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00

The first five bytes are the SPI header. The last 21 bytes are the HCI_VS_SPI_Configuration command bytes.

To change to Swapped-Bytes mode:

Send_HCI_VS_TI_SPI_Configuration 0xFD41, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0
Wait_HCI_Command_Complete_VS_TI_SPI_Configuration_Event 5000, any, HCI_VS_TI_SPI_Configuration, 0x00

Outgoing hex dump:
01 00 15 00 00 01 41 fd 11 00 01 00 00 00 00 00 00 00 00 00 00 01 00 00 00

Special note:

The normal Host write sequence for SPI is CS low (Host --> BRF), followed by IRQ low (BRF --> Host), indicating that the BRF is ready to accept data. However after power up, IRQ will go low automatically to indicate power-up complete - it is the same line as RTS.
Therefore for this first command only, the Host cannot rely on IRQ low and must wait a delay of > 55uS before sending the SPI packet.

In addition, during this command the BRF6300 performs its internal processing to switch to the required SPI mode. This requires an additional small amount of processing time.

Therefore also for this first command only, a short delay is required after the first four bytes and the following 22 bytes. This delay also > 55uS.

5.1.2 Write sequence - Host to BRF6300 data transfer

The Host must assert the CS line (drive the signal to low) before starting to send data on the DI line.

After completing its wakeup sequence (Wakeup sequence period depends whether the external fast clock is available and it is configurable by the *HCI_VS_Fast_Clock_Configuration* command- *refer to the BRF3000 HCI Vendor Specific Command document (BT-SW-0029).*

The BT device asserts the IRQ line to inform the host that it is ready. In case the Bluetooth device is awake the IRQ line will be asserted immediately (within <250nsec).

When the BT device detects the HCI packet header it de-asserts its IRQ line.

When the host completes the SPI transaction it must de-assert its CS line.

Note: *The host can wake-up the Bluetooth device by generating a pulse on the CS line or performing a write transaction*

5.1.3 Read sequence - BRF6300 to Host data transfer

The BT device signals to the host its desire to transfer data by asserting the IRQ line.

The host will assert its CS line and perform a read transaction.

Upon completion of the read transaction, the Host must de-assert the CS line.

The Bluetooth device will de-assert its IRQ line immediately as a response (within <250nsec).

Note: *The host should be set to trigger on High to Low edge.*

5.1.4 16 Bit Alignment

All the data transacted over the BT SPI interface is 16 bits aligned.

5.2 Standard BTSPI mode

5.2.1 Standard mode write operation - Host to BRF6300 data transfer

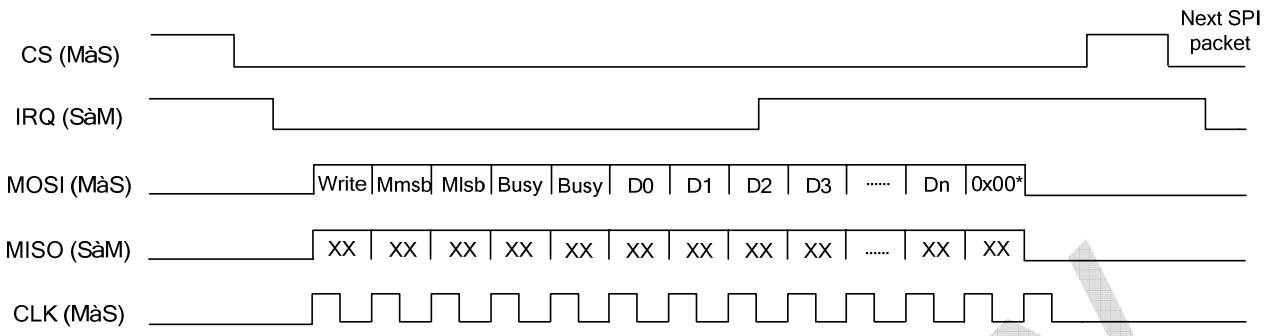

Figure 2 - BTSPI Standard mode write transaction

* Padding byte for 16 bits alignment, if required.

SPI Header

MOSI = BRF6300 DI
MISO = BRF6300 DO
Write – Opcode for write is 0x01
Mmsb & Mlsb – 16 bit data payload length (including the alignment byte)
Busy – Busy byte (0x00)

SPI Payload (= HCI command + padding byte)

D0 ... Dn + 0x00 (depending of number of bytes in SPI payload)

XX – Should be ignored by master

In order for the total SPI packet (SPI transaction) to be 16 bit aligned, the HCI command must be padded with an additional 0x00 byte if the HCI packet is even size.

SPI Header	SPI Payload (= HCI command)	Padding byte
5 bytes	Odd	None
5 bytes	Even	0x00

Table 3 - BT SPI Read/Write Transaction - 16 bits alignment

The write transaction is performed according to the following:

- A single BTSPI write transaction includes a full HCI packet
- The number of bytes for each SPI transaction will always be even
- The padding byte is added at the end of the HCI packet, but is not reflected in the HCI header length parameter (H4 packet length will ignore this byte).

- The BRF6300 will ignore the additional byte

5.2.2 Standard mode read operation - BRF6300 to Host data transfer

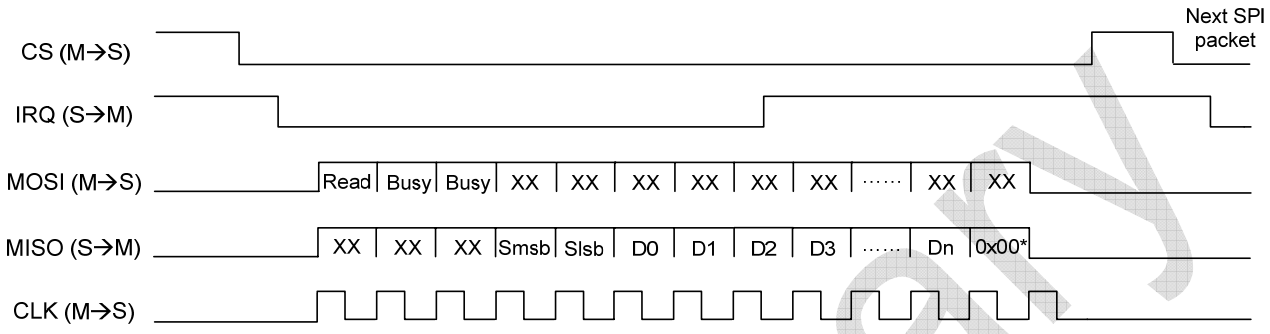

Figure 3 - BTSPI Read Transaction

* Padding byte for 16 bits alignment, if required.

SPI Header

Read – Opcode for read is 0x03
Busy – Busy byte (0x00)
Smsb & Slsb – 16 bit data payload length (including the alignment byte)

SPI Payload (= HCI event + padding byte)

D(0) .. D(n) + 0x00 (depending of number of bytes in SPI payload)

In order for the total SPI packet (SPI transaction) to be 16 bit aligned, the HCI event sent from the BT device will be padded with an additional 0x00 byte if required.

SPI Header	SPI Payload (= HCI event)	Padding byte
5 bytes	Odd	None
5 bytes	Even	0x00

Table 4 - BT SPI Read/Write Transaction - 16 bits alignment

The read transaction will be performed according to the following:

- A single BTSPI read transaction includes a full HCI packet
- The number of bytes for each SPI transaction will always be even.

- The padded byte will be added at the end of the HCI packet, but will not be reflected in the HCI header length parameter (H4 packet length will ignore this byte).
- The host should read a full SPI packet (including the alignment byte) according to the SPI packet length (Smsb & Slsb)
- ***The Host must ignore the additional byte according to the HCI packet length***

5.3 Swapped-Bytes BTSPI mode

5.3.1 Swapped-Bytes write operation - Host to BRF6300 data transfer

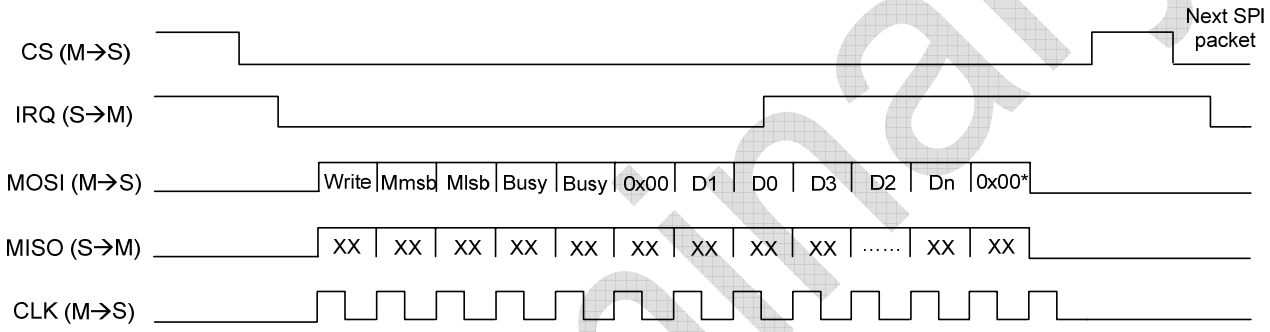

Figure 4 - BTSPI Swapped-Bytes mode Write transaction

* Padding byte for 16 bits alignment, if required.

SPI Header

MOSI = BRF6300 DI
MISO = BRF6300 DO
Write – Opcode for write is 0x01
Mmsb & Mlsb – 16 bit data payload length (including the alignment byte)
Busy – Busy byte (0x00)

SPI Payload (= HCI command + padding byte)

D(0) .. D(n) + 0x00 (depending of number of bytes in SPI payload)

XX – Should be ignored by master

In order for the total SPI packet (SPI transaction) to be 16 bit aligned, the HCI command must be padded with an additional 0x00 byte if the HCI packet is odd size.

SPI Header	SPI Payload (= HCI command)	Padding byte
6 bytes	Odd	0x00
6 bytes	Even	None

Table 5 - BT SPI Read/Write Transaction - 16 bits alignment

The write transaction is performed according to the following:

- A single BTSPI write transaction includes a full HCI packet
- The number of bytes for each SPI transaction will always be even
- The padding byte is added at the end of the HCI packet, but is not reflected in the HCI header length parameter (H4 packet length will ignore this byte).
- The BRF6300 will ignore the additional byte

5.3.2 Swapped-Bytes read operation - BRF6300 to Host data transfer

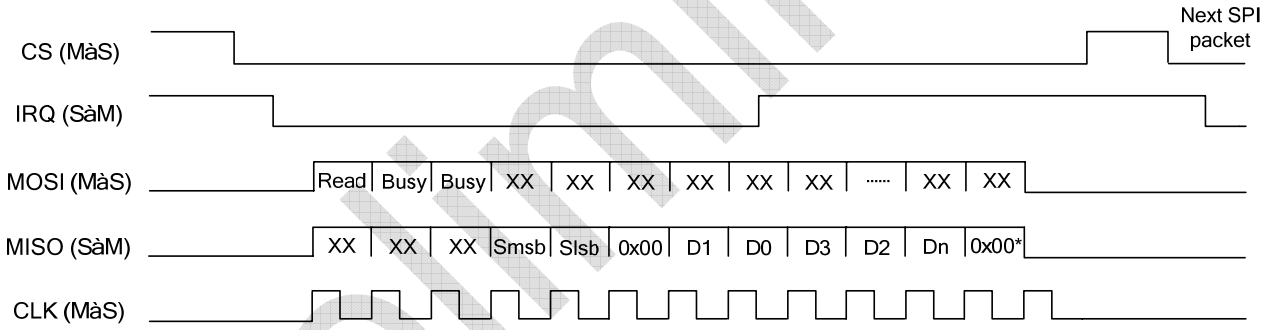

Figure 5 - BTSPI Read Transaction

* Padding byte for 16 bits alignment, if required.

SPI Header

Read – Opcode for read is 0x03
Busy – Busy byte (0x00)
Smsb & Slsb – 16 bit data payload length (including the alignment byte)

SPI Payload (= HCI event + padding byte)

D0 ... Dn + 0x00 (depending of number of bytes in SPI payload)

In order for the total SPI packet (SPI transaction) to be 16 bit aligned, the HCI event sent from the BT device will be padded with an additional 0x00 byte if required.

SPI Header	SPI Payload (= HCI event)	Padding byte
6 bytes	Odd	0x00
6 bytes	Even	None

Table 6 - BT SPI Read/Write Transaction - 16 bits alignment

The read transaction will be performed according to the following:

- A single BTSPI read transaction includes a full HCI packet
- The number of bytes for each SPI transaction will always be even.
- The padded byte will be added at the end of the HCI packet, but will not be reflected in the HCI header length parameter (H4 packet length will ignore this byte).
- The host should read a full SPI packet (including the alignment byte) according to the SPI packet length (Smsb & Slsb)
- ***The Host must ignore the additional byte according to the HCI packet length***

6 Clock Polarity

The polarity of the clock used by the SPI protocol can be switched by initiating a vendor specific command through the SPI interface- *HCI_VS_BTSPI_Configuration* (Refer to Section 10.2 for detailed description of this command).

The switching command will have to be sent using the default clock polarity as described in the diagrams above, but the polarity can be switched for subsequent transactions.

7 Shared SPI Bus Mode

This section describes TI solution for a system in which the SPI Host Controller interfaces with Bluetooth (BRF6300) and WLAN (WL1251) devices.

The topology is of one SPI master and an SPI bus shared by several slaves.

The bus topology has a single master (HOST) and multiple slaves (WLAN and Bluetooth devices).

The following lines are common to all SPI devices in the system:

- CLK
- DI
- DO – when this line is shared between the different devices, it must be set to tri-state when the CS is de-asserted via the VS command *HCI_VS_BTSPI_Configuration (refer to Section 9.2)*

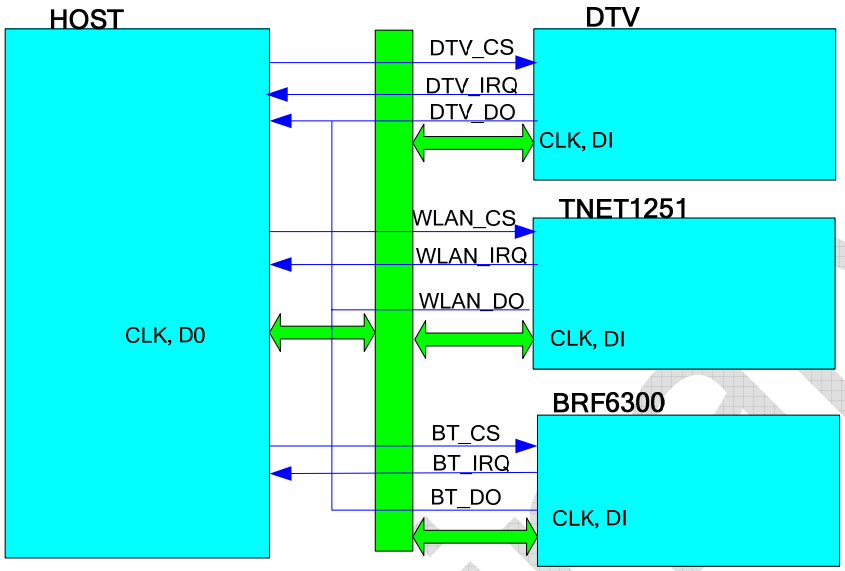

Figure 6 : Shared SPI Bus Topology

8 Initialization Commands For BTSPI

In order for the BRF6300 to be working in BTSPI protocol following commands must be sent to the device and they must be sent in the specified order:

1) When working in non-shared mode:
   Send_HCI_VS_TI_SPI_Configuration 0xFD41, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0
   Wait_HCI_Command_Complete_VS_TI_SPI_Configuration_Event 5000, any, HCI_VS_TI_SPI_Configuration, 0x00

   When working in shared mode:
   Send_HCI_VS_TI_SPI_Configuration 0xFD41, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0
   Wait_HCI_Command_Complete_VS_TI_SPI_Configuration_Event 5000, any, HCI_VS_TI_SPI_Configuration, 0x00

2) Enabling the deep-sleep protocol
   Send_HCI_VS_Sleep_Mode_Configurations 0xFD0C, 1, 1, 0x06, 0xFF, 0xFF, 0xFF, 0, 100
   Wait_HCI_Command_Complete_VS_Sleep_Mode_Configurations_Event 5000, any, HCI_VS_Sleep_Mode_Configurations, 0x00

3) All the commands of the most updated init script
4) Other platform-related commands

9 Related Commands

The related commands are enclosed here for the convenience of the user. However for the most updated version of the commands, please refer to the BRF6300 HCI Vendor Specific Command (BT-SW-0029) document.

9.1 The HCI_VS_Sleep_Mode_Configurations Command

Command	Opcode	Command Parameters	Return Parameters
HCI_VS_Sleep_Mode_Configurations	0xFD0C	Big Sleep Enable Deep Sleep Enable Deep Sleep protocol mode Output IO select (Reserved) Output pull enable Input pull enable Input IO select Reserved De-assertion Timeout	status

Description:

This command configures the sleep mode to be used.

Note that prior to sending this command deep sleep is disabled.

Default values:

Big Sleep is enabled by default

Deep Sleep is disabled by default.

Command Parameters:

Big sleep enable	*Size: 1 Byte*
Value	**Parameter Description**
0x0	Big sleep is disabled
0x1	Big sleep is enabled

Deep sleep enable	*Size: 1 Byte*
Value	**Parameter Description**
0x0	Deep sleep is disabled
0x1	Deep sleep is enabled

Deep sleep protocol mode	*Size: 1 Byte*
Value	**Parameter Description**
0x0	HCILL

0x1	Reserved
0x2	Reserved
0x3	Reserved
0x4	Reserved
0x5	Reserved
0x6	SPI
0x7	SDIO Protocol
0x8	Reserved

Reseved	*Size: 1 Byte*
Value	**Parameter Description**
Reserved	*Value must be set to 0xFF, Reserved for future usage.*

Output pull Enable	*Size: 1 Bytes*
Value	**Parameter Description**
0x0	Output pull is disabled
0x1	Output pull is enabled
0xFF	Don't change

Input pull Enable	*Size: 1 Bytes*
Value	**Parameter Description**
0x0	Input pull is disabled
0x1	Input pull is enabled
0xFF	Don't change

Reserved	*Size: 1 Byte*
Value	**Parameter Description**
Reserved	*Value must be set to 0x00*, Reserved for future usage.

Host_Wake deassertion timer	*Size: 2 Bytes*
Value	**Parameter Description**
0x00	Reserved
0x0001 - 0xFFFF	Reserved

Return Parameters:

Status:	Size: 1 Byte
Value	Parameter Description
0x00	Command Succeeded.

0x01-0xFF	Command failed. See Appendix B for HCI error codes.

Events Generated:
Command Complete Event

9.2 The HCI_VS_TI_SPI_Configuration Command

Command	Opcode	Command Parameters	Return Parameters
HCI_VS_BTSPI_Configuration		BTSPI Bits Direction (LSB First Enable) BTSPI Clock Polarity (Data Changed on rising enable) BTSPI Clock Frequency BTSPI DO 3's Enable BTSPI Fast Wakeup Enable BTSPI Mode BTSPI DO Pull Enable BTSPI DI Pull Enable BTSPI Clock Pull Enable BTSPI IRQ Pull Enable BTSPI CS Pull Enable BTSPI Mode select Spare [3 Bytes]	status

Description:
This command configures the BRF6300 SPI configuration. Please note that when using SPI this command is required to be sent to device directly after device initialization.

Default values:

Bits Direction:	MSB First
Clock Polarity:	Output changed on rising edge, Sample on Falling edge
Clock Frequency:	6.5MHz
DO 3's Enable:	Disable
Fast Wakeup Enable:	Disable
BT SPI Mode:	H4 with SPI flow control
DO Pull Enable:	Enable
DI Pull Enable:	Enable
Clock Pull Enable:	Enable
IRQ Pull Enable:	Enable
CS Pull Enable:	Enable

Command Parameters:

*Bits Direction (LSB First Enable)**	*Size: 1 Byte*
Value	**Parameter Description**

0x0	MSB First on SPI data
0x1	LSB First on SPI data
0xFF	Don't change

Clock Polarity (Data Changed on rising enable)	*Size: 1 Byte*
Value	**Parameter Description**
0x0	Output changed on falling edge, Sampling on rising edge
0x1	Output changed on rising edge, Sampling on falling edge
0xFF	Don't change

Clock Frequency [KHz]	*Size: 2 Bytes*
Value	**Parameter Description**
0x0 – 0xFFFE	BTSPI Clock input frequency [KHz] (0-65534KHz)
0xFFFF	Don't change

BTSPI DO tri-state Enable	*Size: 1 Byte*
Value	**Parameter Description**
0x0	BTSPI DO is always driven – for shared mode explain…
0x1	BTSPI DO is in tri-state mode when the CS is de-asserted (not active). This is relevant when shared SPI mode is used.
0xFF	Don't change

BTSPI Fast Wakeup Enable	*Size: 1 Byte*
Value	**Parameter Description**
0x0	BTSPI Fast wakeup from sleep is disabled
0x1	BTSPI Fast wakeup from sleep is enabled – relevant for cases where the fast clock is already available when the BRF is woken up
0xFF	Don't change

BTSPI Mode	*Size: 1 Byte*
Value	**Parameter Description**
0x0	Reserved
0x1	BTSPI
0xFF	Don't change

BTSPI DO Pull Enable	*Size: 1 Byte*
Value	**Parameter Description**
0x0	Disable BTSPI DO pull up resistor
0x1	Enable BTSPI DO pull up resistor
0xFF	Don't change

BTSPI DI Pull Enable	*Size: 1 Byte*
Value	**Parameter Description**

0x0	Disable BTSPI DI pull up resistor
0x1	Enable BTSPI DI pull up resistor
0xFF	Don't change

BTSPI Clock Pull Enable	*Size: 1 Byte*
Value	**Parameter Description**
0x0	Disable BTSPI Clock pull up resistor
0x1	Enable BTSPI Clock pull up resistor
0xFF	Don't change

BTSPI IRQ Pull Enable	*Size: 1 Byte*
Value	**Parameter Description**
0x0	Disable BTSPI IRQ pull up resistor
0x1	Enable BTSPI IRQ pull up resistor
0xFF	Don't change

BTSPI CS Pull Enable	*Size: 1 Byte*
Value	**Parameter Description**
0x0	Disable BTSPI CS pull up resistor
0x1	Enable BTSPI CS pull up resistor
0xFF	Don't change

BTSP Mode select	*Size: 1 Byte*
Value	**Parameter Description**
0x0	Standard BTSPI mode
0x1	Swapped-Bytes BTSPI mode
0xFF	Don't change
Spare	*Size: 3 Byte*
Value	**Parameter Description**
Reserved	0x00

Return Parameters:

Status:	*Size: 1 Bytes*
Value	**Parameter Description**
0x00 0x01-0xFF	Command Succeeded. Command failed.

Events Generated:

Command Complete Event

* First command to be sent on SPI must be MSB mode.

10 Reference Documents

Document	Reference
BRF6300 Product Review	BT-DS-0023
BRF3000 HCI Vendor Specific Command,	BT-SW-0029

Important Notice

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE–SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, the customer to minimize inherent or procedural hazards must provide adequate design and operating safeguards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.