WT21

DATA SHEET

Monday, 03 August 2009

Version 1.6

bluegiga

Bluegiga Technologies assumes no responsibility for any errors which may appear in this manual. Furthermore, Bluegiga Technologies reserves the right to alter the hardware, software, and/or specifications detailed here at any time without notice and does not make any commitment to update the information contained here. Bluegiga's products are not authorized for use as critical components in life support devices or systems.

VERSION HISTORY

Version	Comment
0.1	First draft
0.2	Block diagram, descriptions added
0.3	Preliminary version
0.4	Fixed ordering codes added captions
0.5	Power control and regulation info added. Layout guide updated. Minor updates and fixes.
1.0	Electrical characteristic added. Some minor updates.
1.1	Function of the regulator enable pin corrected. Some minor updates.
1.2	New template
1.3	Pinout fixed (GND pins 1 – 3 removed, pin 23 grounded). Layout guide updated.
1.4	Improved dimensions chapter
1.41	Table 5 fixed (pad types)
1.5	Footprint added
1.6	Footprint fixed. Pin number 3 (NC) added.

TABLE OF CONTENTS

WT21 *Bluetooth*® HCI Module

DESCRIPTION

WT21 is intended for *Bluetooth* applications where a host processor is capable of running the *Bluetooth* software stack. WT21 only implements the low level *Bluetooth* Host Controller Interface (HCI) but still offers advantages of a module - easy implementation and certifications.

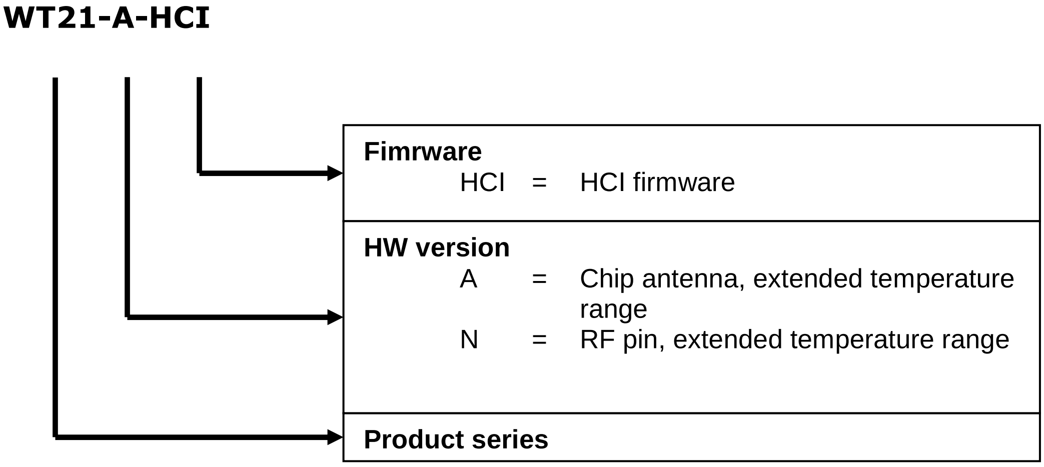
APPLICATIONS:

- PCs and laptops
- PDAs
- Embedded systems

FEATURES:

- Fully Qualified Bluetooth v2.1 + EDR System
- Piconet and Scatternet Support
- Low Power Consumption
- 1,8V to 3,6V I/O Voltage
- Integrated 1,8V Regulator
- UART to 4 Mbaud
- SDIO (Bluetooth Type A) and CSPI Host Interfaces
- Deep-Sleep SDIO Operation
- Support for 802.11 Coexistence
- RoHS Compliant
- AuriStream Baseband Codec

1 Ordering Information



WT21-A-HCI

Fimrware

HCI	=	HCI firmware

HW version

A	=	Chip antenna, extended temperature range
N	=	RF pin, extended temperature range

Product series

2 Pinout and Terminal Description

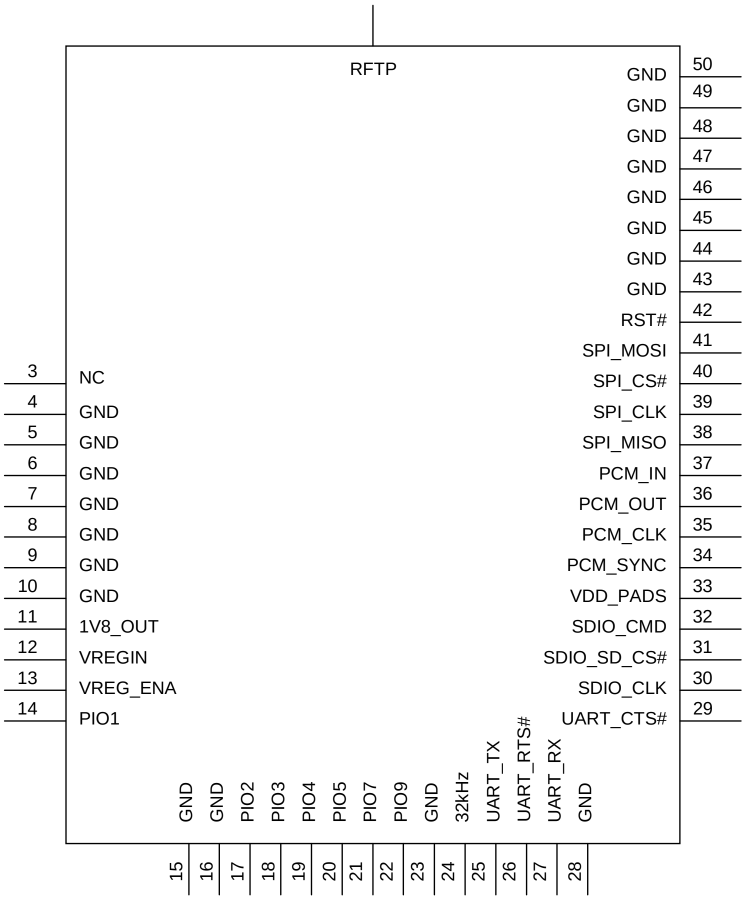


Figure 1: WT21 pin out

	PIN NUMBER	PAD TYPE	DESCRIPTION
NC	3	Not in use	Leave floating or connect to GND
RST#	42	Input, weak internal pull-up	Active low reset. Keep low for >5 ms to cause a reset
GND	23	GND	GND

Table 1: Terminal Descriptions

POWER SUPPLIES	PIN NUMBER	DESCRIPTION
VREGIN	12	Input for the internal 1,8V regulator
1v8_OUT	11	1,8V regulator output
VREG_ENA	13	Take high to enable internal voltage regulators
GND	4-10, 15-16, 28, 43-50	Ground
VDD_PADS	33	Positive supply for the digital interfaces

Table 2: Terminal Descriptions

PIO PORT	PIN NUMBER	PAD TYPE	DESCRIPTION
PIO[1]	14	Bi-directional, programmamble strength internal pull-down/pull-up	Programmamble input/output line
PIO[2]	17	Bi-directional, programmamble strength internal pull-down/pull-up	Programmamble input/output line
PIO[3]	18	Bi-directional, programmamble strength internal pull-down/pull-up	Programmamble input/output line
PIO[4]	19	Bi-directional, programmamble strength internal pull-down/pull-up	Programmamble input/output line
PIO[5]	20	Bi-directional, programmamble strength internal pull-down/pull-up	Programmamble input/output line
PIO[7]	21	Bi-directional, programmamble strength internal pull-down/pull-up	Programmamble input/output line
PIO[9]	22	Bi-directional, programmamble strength internal pull-down/pull-up	Programmamble input/output line

Table 3: Terminal Descriptions

SPI INTERFACE	PIN NUMBER	PAD TYPE	DESCRIPTION
PCM_OUT	36	Output, tri-state, weak internal pull-down	Synchronous data output
PCM_IN	37	Input, weak internal pull-down	Synchronous data input
PCM_SYNC	34	Bi-directional, weak internal pull-down	Synchronous data sync
PCM_CLK	35	Bi-directional, weak internal pull-down	Synchronous data clock

Table 4: Terminal Descriptions

SDIO/CSPI/UART Interfaces	PIN NUMBER	PAD TYPE	DESCRIPTION
SDIO_DATA[0]	25	Bi-directional, tri-state, weak internal pull-down	Synchronous data input/output
CSPI_MISO			CSPI data output
UART_TX			UART data output, active high
SDIO_DATA[1]	26	Bi-directional, weak internal pull-down	Synchronous data input/output
CSPI_INT			CSPI data input
UART_RTS#			UART request to send, active low
SDIO_DATA[2]	27	Bi-directional, weak internal pull-	Synchronous data input/output
UART_RX			UART data input, active high
SDIO_DATA[3]	29	Bi-directional, weak internal pull-down	Synchronous data input/output
CSPI_CS#			Chip select for CSR Serial Peripheral Interface, active low
UART_CTS#			UART clear to send, active low
SDIO_CLK	30	Bi-directional, weak internal pull-	SDIO clock
CSPI_CLK			CSPI clock
SDIO_SD_CS#	31	Bi-directional, weak internal pull-down	SDIO chip select to allow SDIO accessess
SDIO_CMD	32	Bi-directional, weak internal pull-	SDIO data input
CSPI_MOSI			CSPI data input

Table 5: Terminal Descriptions

SPI INTERFACE	PIN NUMBER	PAD TYPE	DESCRIPTION
SPI_MOSI	41	Weak internal pull-down	SPI data input
SPI_CS#	40	Bi-directional, weak internal pull-down	Chip select for Serial Peripheral Interface, active low
SPI_CLK	39	Bi-directional, weak internal pull-down	SPI clock
SPI_MISO	38	Output, tri-state, weak internal pull-down	SPI data output

Table 6: Terminal Descriptions

3 Microcontroller, Memory and Baseband Logic

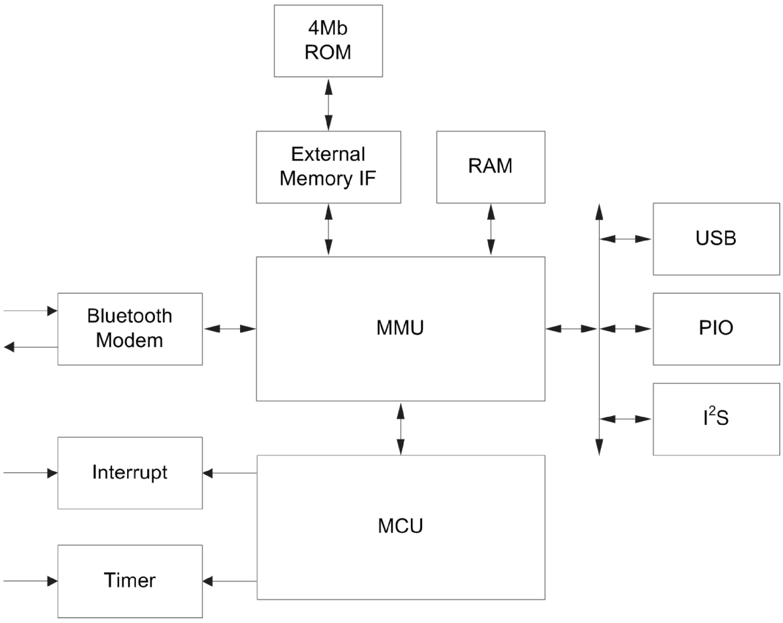


Figure 2: Baseband Digits Block Diagram

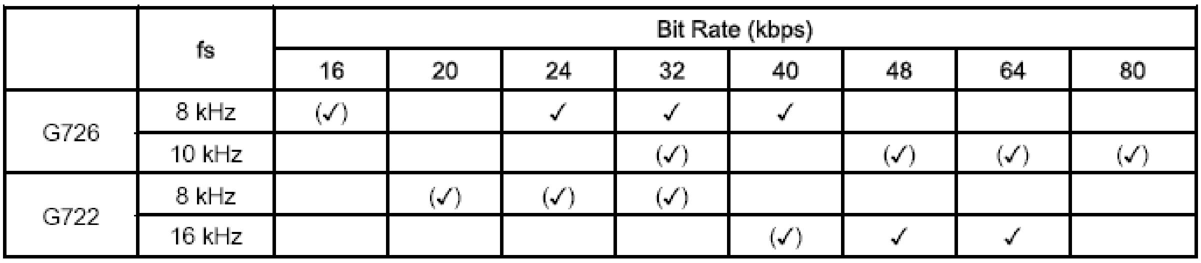
3.1 AuriStream CODEC

The AuriStream CODEC works on the principle of transmitting the delta between the actual value of the signal and a prediction rather than the signal itself. Hence, the information transmitted is reduced along with the power requirement. The quality of the output depends on the number of bits used to represent the sample.

The inclusion of AuriStream results in reduced power consumption compared to a CVSD implementation when used at both ends of the system.

3.1.1 AuriStream CODEC Requirements

AuriStream supports the following modes of operation:



	fs	Bit Rate (kbps)							
		16	20	24	32	40	48	64	80
G726	8 kHz	(✓)		✓	✓	✓			
	10 kHz				(✓)		(✓)	(✓)	(✓)
G722	8 kHz		(✓)	(✓)	(✓)				
	16 kHz					(✓)	✓	✓	

Table 7: AuriStream Supported Bitrates

Table Key:

= Standard Mode

(✓) = Optional Mode

Where possible, AuriStream shares hardware between the encoder and decoder as well as the G726 and G722 implementations of the standard. The 40kbs and 20kbs modes of the G722 codec are specific to CSR.

The AuriStream module will be required to support the 3Mbps stream transmitted by the BT radio. The worst-case scenario arises when the AuriStream block is configured as 16kbps at 8 kHz, which equates to 2 bits per sample, giving a worst-case symbol rate at the input to the AuriStream block of 1.5Msps to sustain the transmitted bit stream.

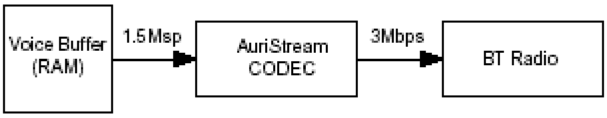


Figure 3: AuriStream CODEC and the BT Radio

3.1.2 AuriStream Hierarchy

The AuriStream CODEC is positioned in parallel with the CVSD CODEC as shown in Figure 4.

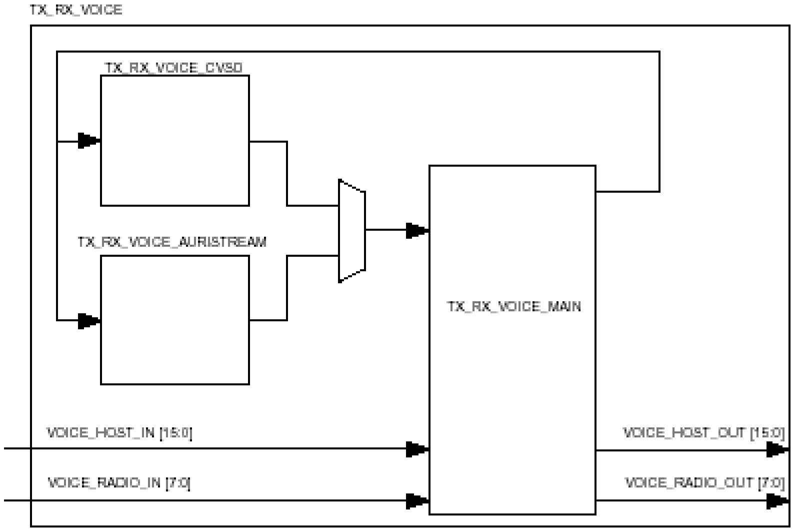


Figure 4: AuriStream CODEC and the CVSD CODEC

The AuriStream CODEC is controlled by the TX_RX_VOICEmain block and the processor. Raw data from the host is read from the MMU by the transmit block. This data is fed via the TX_RX_VOICE_MAIN module to the required CODEC, the encoded data is then fed back to the transmit block for broadcast over the Bluetooth interface. During reception, the data is sourced from the radio and applied to the required CODEC. The decoded data is then stored back to RAM by the bluetooth receiver.

3.2 Memory Managements Unit

The Memory Management Unit (MMU) provides a number of dynamically allocated ring buffers that hold the data that is in transit between the host and the air. The dynamic allocation of memory ensures efficient use of the available Random Access Memory(RAM) and is performed by a hardware MMU to minimise the overheads on the processor during data/voice transfers.

3.3 Burst Mode Controller

During transmission the Burst Mode Controller(BMC) constructs a packet from header information previously loaded into memory-mapped registers by the software and payload data/voice taken from the appropriate ring buffer in the RAM. During reception, the BMC stores the packet header in memory-mapped registers and the payload data in the appropriate ring buffer in RAM. This architecture minimises the intervention required by the processor during transmission and reception.

3.4 Physical Layer Hardware Engine DSP

Dedicated logic is used to perform the following:

- Forward error correction
- Header error correction
- Cyclic redundancy check
- Encryption
- Data whitening
- Access code correlation
- Audio transcoding

The following voice data translations and operations are performed by firmware:

- A-law/µ-law/linear voice data (from host)
- A-law/µ-law/Continuously variable Slope Delta (CVSD) (over the air)

- Voice interpolation for lost packets
- Rate mismatches

The hardware supports all optional and mandatory features of Bluetooth v2.1 + EDR including AFH and eSCO.

3.5 WLAN Coexistence

Dedicated hardware is provided to implement a variety of coexistence schemes. Channel skipping AFH, priority signalling, channel signalling and host passing of channel instructions are all supported. The features are configured in firmware.

For more information contact Buegiga technical support.

3.6 Configurable I/O Parallel Ports

lines of programmable bi-directional input/outputs (I/O) are provided. PIO[1: 5, 7, 9] are powered from VDD_PADS.

PIO lines can be configured through software to have either weak or strong pull-ups or pull-downs. All PIO lines are configured as inputs with weak pull-downs at reset.

Any of the PIO lines can be configured as interrupt request lines or as wake-up lines from sleep modes.

Bluegiga cannot guarantee that the PIO assignments remain as described. Refer to the relevant software release note for the implementation of these PIO lines, as they are firmware build-specific.

4 Clock Generation

WT21 uses an internal 26 MHz crystal as a Bluetooth reference clock. All WT21 internal digital clocks are generated using a phase locked loop, which is locked to the 26 MHz reference clock.

Also supplied to the digits is a watchdog clock, for use in low power modes. This uses a frequency of 32.768kHz from CLK_32K, or an internally generated reference clock frequency of 1kHz, determined by PSKEY_DEEP_SLEEP_EXTERNAL_CLOCK_SOURCE.

The use of the watchdog clock is determined with respect to Bluetooth operation in low power modes.

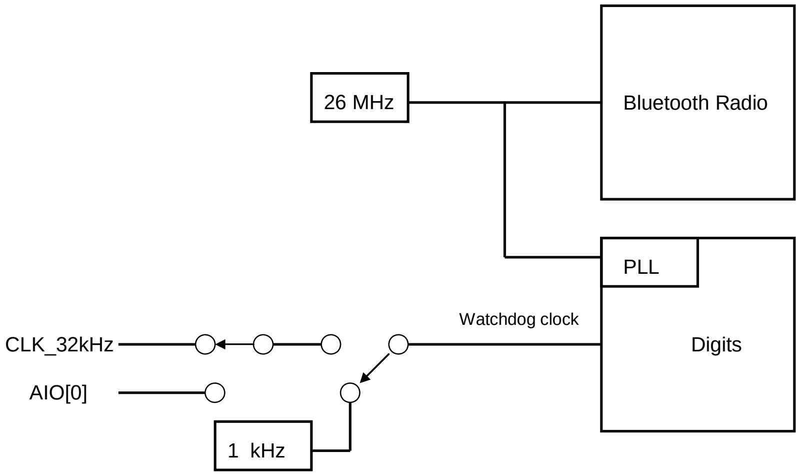


Figure 5: Clock Architecture

4.1 32kHz External Reference Clock

A 32kHz clock can be applied to CLK_32K, using PSKEY_DEEP_SLEEP_EXTERNAL_CLOCK_SOURCE.

The CLK_32K pad is in the VDD_PADS domain with all the other digital I/O pads and is driven between levels specified in Section 11.3.4.

5 Serial Peripheral Interface (SPI)

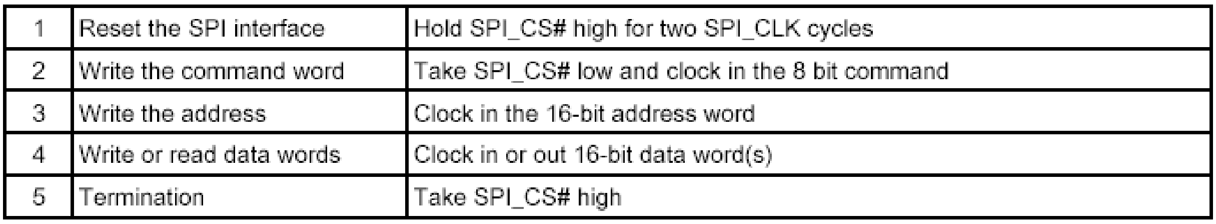
5.1 WT21 Serial Peripheral Interface (SPI)

SPI is used for debuging primarily. This section details the considerations required when interfacing to WT21 via the SPI.

Data may be written or read one word at a time or the auto increment feature may be used to access blocks.

5.2 Instruction Cycle

WT21 is the slave and receives commands on SPI_MOSI and outputs data on SPI_MISO. Table 8 shows the instruction cycle for an SPI transaction.



1	Reset the SPI interface	Hold SPI_CS# high for two SPI_CLK cycles
2	Write the command word	Take SPI_CS# low and clock in the 8 bit command
3	Write the address	Clock in the 16-bit address word
4	Write or read data words	Clock in or out 16-bit data word(s)
5	Termination	Take SPI_CS# high

Table 8: Instruction Cycle for an SPI Transaction

With the exception of reset, SPI_CS# must be held low during the transaction. Data on SPI_MOSI is clocked into the WT21 on the rising edge of the clock line SPI_CLK. When reading, WT21 replies to the master on SPI_MISO with the data changing on the falling edge of the SPI_CLK. The master provides the clock on SPI_CLK. The transaction is terminated by taking SPI_CS# high.

Sending a command word and the address of a register for every time it is to be read or written is a significant overhead, especially when large amounts of data are to be transferred. To overcome this WT21 offers increased data transfer efficiency via an auto increment operation. To invoke auto increment, SPI_CS# is kept low, which auto increments the address, while providing an extra 16 clock cycles for each extra word to be written or read.

5.2.1 Writing to the Device

To write to WT21, the 8-bit write command (00000010) is sent first (C[7:0]) followed by a 16-bit address (A[15:0]). The next 16-bits (D[15:0]) clocked in on SPI_MOSI are written to the location set by the address (A). Thereafter for each subsequent 16-bits clocked in, the address (A) is incremented and the data written to consecutive locations until the transaction terminates when SPI_CS# is taken high.

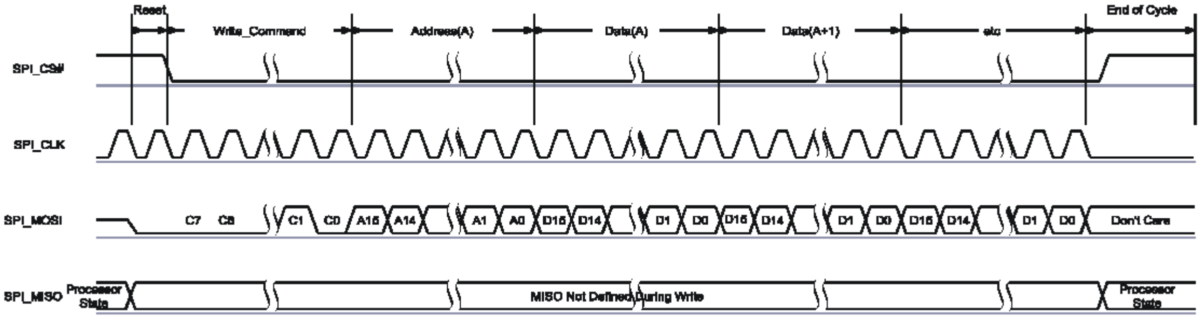


Figure 6: SPI Write Operation

5.2.2 Reading from the Device

Reading from WT21 is similar to writing to it. An 8-bit read command (00000011) is sent first (C [7:0]), followed by the address of the location to be read (A[15:0]). WT21 then outputs on SPI_MISO a check word during T[15:0] followed by the 16-bit contents of the addressed location during bits D[15:0].

The check word is composed of {command, address [15:8]}. The check word may be used to confirm a read operation to a memory location. This overcomes the problems encountered with typical serial peripheral interface slaves, whereby it is impossible to determine whether the data returned by a read operation is valid data or the result of the slave device not responding.

If SPI_CS# is kept low, data from consecutive locations is read out on SPI_MISO for each subsequent 16 clocks, until the transaction terminates when SPI_CS# is taken high.

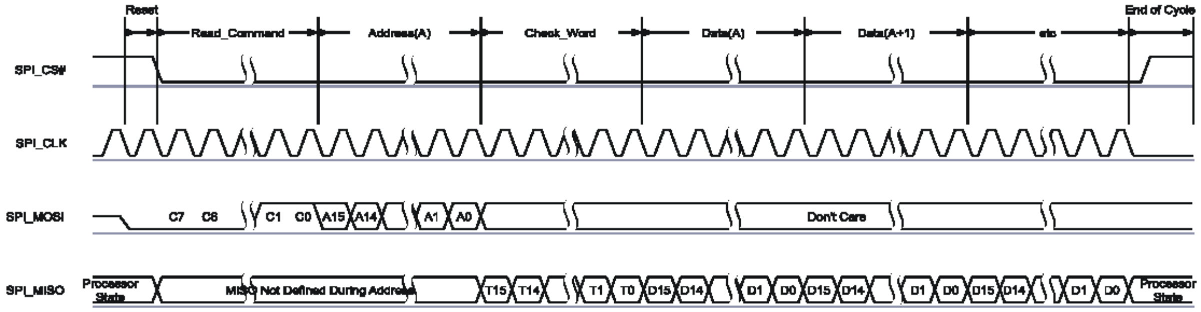


Figure 7: SPI Read Operation

5.2.3 Multi-Slave Operation

WT21 should not be connected in a multi-slave arrangement by simple parallel connection of slave MISO lines. When WT21 is deselected (SPI_CS# = 1), the SPI_MISO line does not float. Instead, WT21 outputs 0 if the processor is running or 1 if it is stopped.

6 Host Interfaces

6.1 Host Selection

The MCU selects the UART/SDIO interfaces by reading PIO[4] at boot-time. When PIO[4] is high, the SDIO interface is enabled; when PIO[4] is low, the UART is enabled.

If in UART mode, the MCU selects the UART transfer protocol automatically using the unused SDIO pins shown in Table 9

SDIO_CLK	SDIO_CMD	Protocol
0	0	bcsp
0	1	h4
1	0	h4ds
1	1	h5

Table 9: SDIO_CLK and SDIO_CMD transfer Protocols

6.2 UART Interface

This is a standard UART interface for communicating with other serial devices.

WT21 UART interface provides a simple mechanism for communicating with other serial devices using the RS232 protocol.

Note:

> WT21 uses RS232 protocol, but voltage levels are 0V to VDD_PADS (requires external RS232 transceiver chip.

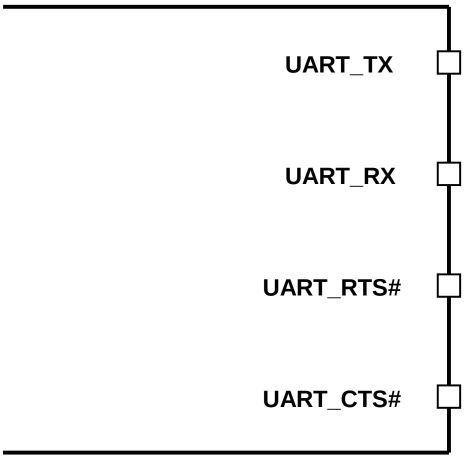


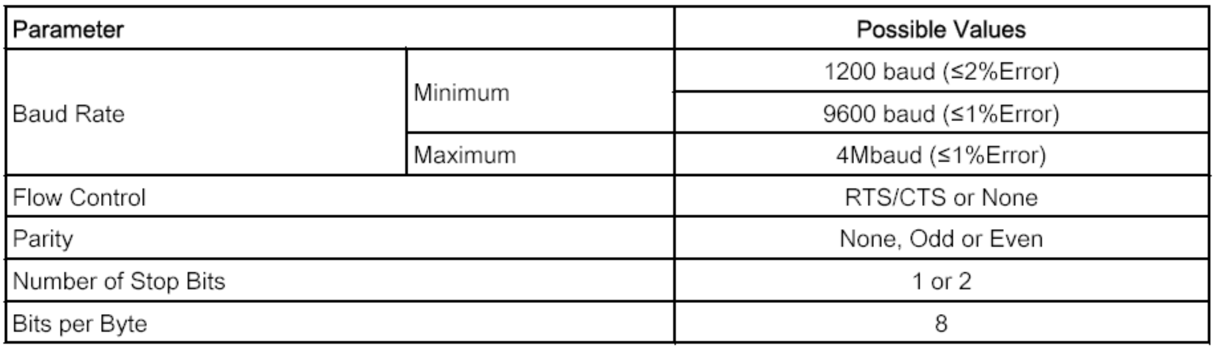
Figure 8: Universal Asynchronous Receiver

Four signals implement the UART function, as shown in Figure 8. When WT21 is connected to another digital device, UART_RX and UART_TX transfer data between the two devices. The remaining two signals, UART_CTS and UART_RTS, can be used to implement RS232 hardware flow control where both are active low indicators.

UART configuration parameters, such as baud rate and packet format, are set using WT21 firmware.

Note:

> An accelerated serial port adapter is required to communicate with the UART at maximum baud rate using a standard PC.



Parameter		Possible Values
Baud Rate	Minimum	1200 baud (≤2%Error)
		9600 baud (≤1%Error)
	Maximum	4Mbaud (≤1%Error)
Flow Control		RTS/CTS or None
Parity		None, Odd or Even
Number of Stop Bits		1 or 2
Bits per Byte		8

Table 10: Possible UART Settings

Note:

> Baud rate is the measure of symbol rate i.e. , the number of distinct symbol changes (signaling events) made to transmission medium per second in a digitally modulated signal.

The UART interface is capable of resetting WT21 on reception of a break signal. A break is identified by a continuous logic low (0V) on the UART_RX terminal, as shown in Figure 9. If tBRKis longer than the value, defined by the PSKEY_HOSTIO_UART_RESET_TIMEOUT, (0x1a4), a reset occurs. This feature allows a host to initialise the system to a known state. Also, WT21 can emit a break character that may be used to wake the host. By default this feature is disabled and it is recommended to enable it by setting PSKEY_HOSTIO_UART_RESET_TIMEOUT.

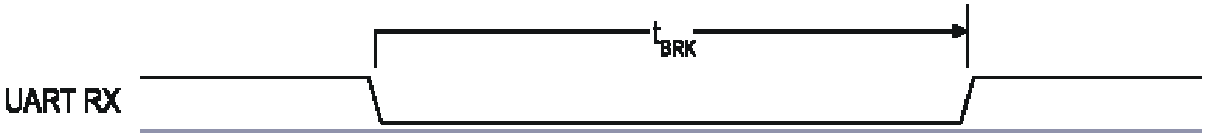
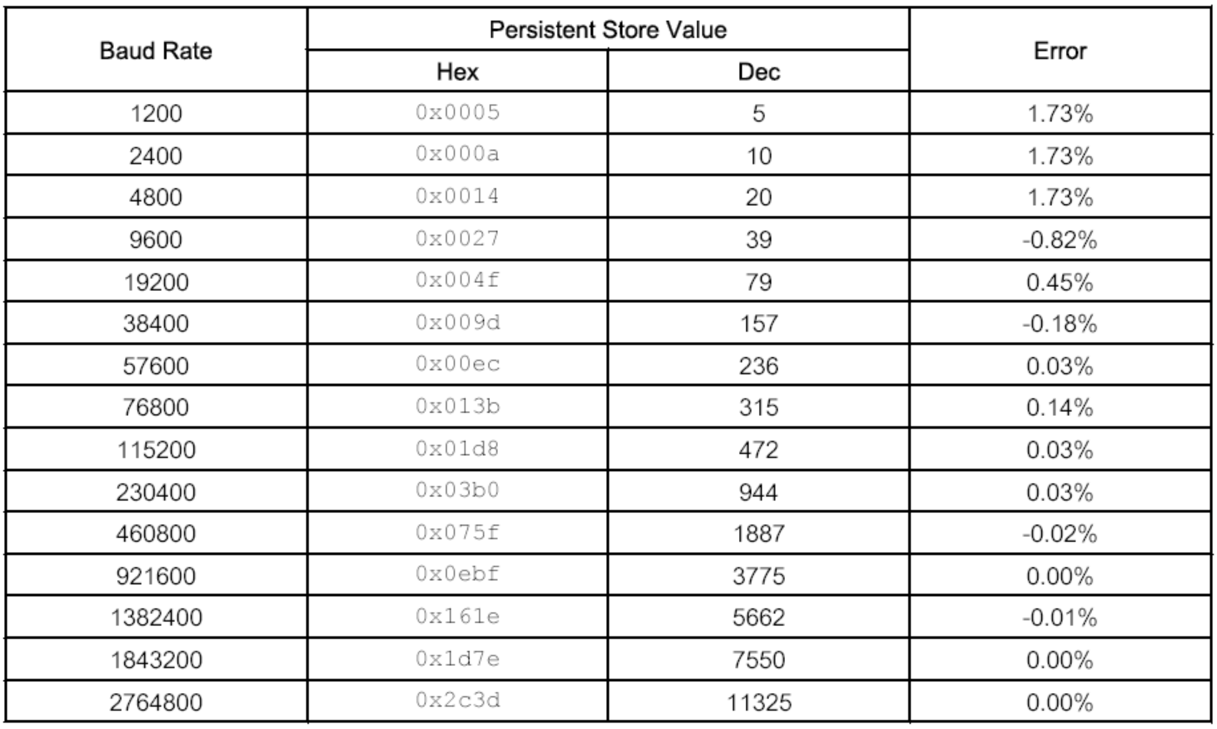


Figure 9: Break Signal

Table 11 shows a list of commonly used baud rates and their associated values for the PSKEY_UART_BAUDRATE (0x1be). There is no requirement to use these standard values. Any baud rate within the supported range can be set in the PS Key according to the formula in Equation XXX.

$$\text{Baud Rate} = \frac{\text{PSKEY\_UART\_BAUDRATE}}{0.004096}$$

Equation 1: Baud Rate



Baud Rate	Persistent Store Value		Error
	Hex	Dec	
1200	0x0005	5	1.73%
2400	0x000a	10	1.73%
4800	0x0014	20	1.73%
9600	0x0027	39	-0.82%
19200	0x004f	79	0.45%
38400	0x009d	157	-0.18%
57600	0x00ec	236	0.03%
76800	0x013b	315	0.14%
115200	0x01d8	472	0.03%
230400	0x03b0	944	0.03%
460800	0x075f	1887	-0.02%
921600	0x0ebf	3775	0.00%
1382400	0x161e	5662	-0.01%
1843200	0x1d7e	7550	0.00%
2764800	0x2c3d	11325	0.00%

Table 11: Standard Baud Rates

6.2.1 UART Configuration While Reset is Active

The UART interface for WT21 is tri-state while the chip is being held in reset. This allows the user to daisy chain devices onto the physical UART bus. The constraint on this method is that any devices connected to this bus must tri-state when WT21 reset is de-asserted and the firmware begins to run.

7 CSR Serial Peripheral Interface (CSPI)

The CSPI is a host interface which shares pins with the SDIO. It has been defined by CSR with the intention of producing a very simple interface. This has two advantages:

- It allows maximum compatibility with the possible host drivers
- It minimizes the host software effort needed to form that data to be sent (e.g., by removing the need to calculate CRCs)

This host interface allows an external host to control the Bluecore, using a CSR defined protocol built upon a 4-wire SPI bus.

Note:

The CSPI is entirely separated from the debug Serial Peripheral Interface

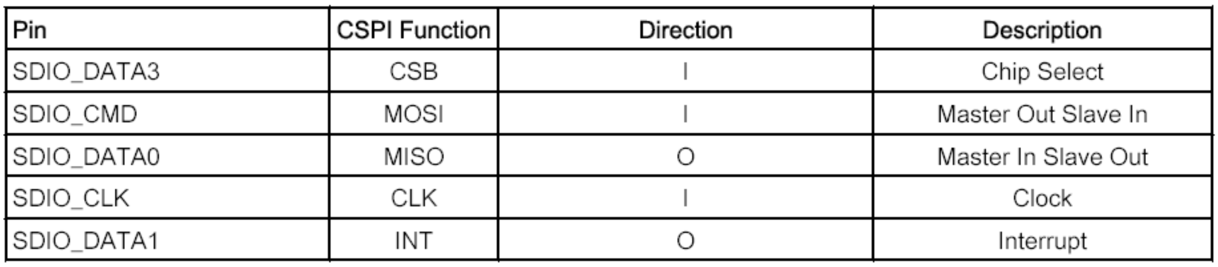
The CSPI allows access to the following:

- Function 0 registers
- Bluetooth Acceleration Registers
- MCU IO Registers
- Bluetooth MMU port

The CSPI is a third protocol available for the host to transfer data into the Bluecore and shares pins with the other SDIO protocols.

MMU buffers are accessed using burst read/writes. The command and address fields are used to select the correct buffer. The CSPI is able to generate an interrupt to the host when a memory access fails. This interrupt line is shared with the SDIO functions.

Table 12 shows the mapping of SDIO pins onto the CSPI functions when CSPI is enabled.



Pin	CSPI Function	Direction	Description
SDIO_DATA3	CSB	I	Chip Select
SDIO_CMD	MOSI	I	Master Out Slave In
SDIO_DATA0	MISO	O	Master In Slave Out
SDIO_CLK	CLK	I	Clock
SDIO_DATA1	INT	O	Interrupt

Table 12: SDIO Mapping to CSPI Functions

The CSPI Interface is an extension of the basic SPI Interface, with the access type determined by the following fields:

- 8-bit command
- 24-bit address
- 16-bit burst length (optional). Only applicable for burst transfers into or out of the MMU

7.1.1 CSPI Read/Write Cycles

Register read/write cycles are used to access Function 0, Bluetooth acceleration and MCU registers.

Burst read/write cycles are used to access the MMU.

7.1.2 CSPI Register Write Cycle

The command and address are locked into the slave, followed by 16bits of write data. An Error Byte is returned on the MISO signal indicating whether or not the transfer has been successful.

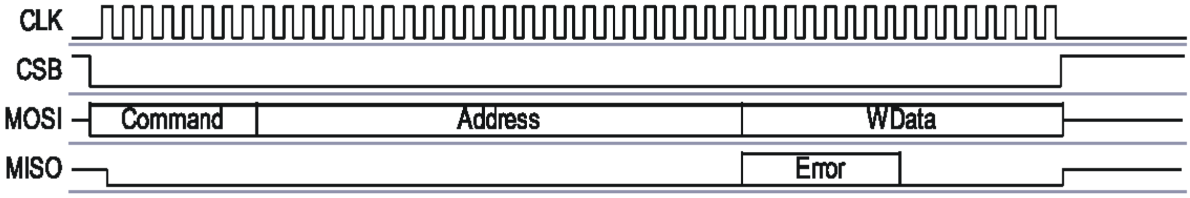


Figure 10: CSPI Register Write Cycle

7.1.3 CSPI Register Read Cycle

The command and address field are clocked into the slave, the slave then returns the following:

- Bytes of badding data (MISO held low)
- Error byte
- 16-bits of read data

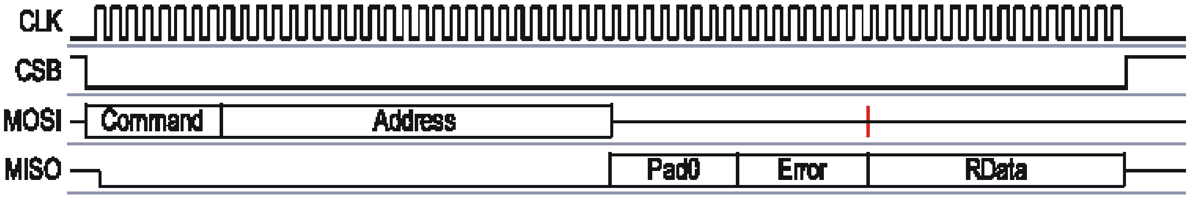


Figure 11: CSPI Register Read Cycle

7.1.4 CSPI Register Burst Write Cycle

Burst transfers are used to access the MMU buffers. They cannot be used to access registers. Burst read/write cycles are selected by setting the nRegister/Burst bit in the command field to 1.

Burst transfers are byte orientated, have a minimum length of 0 bytes and a maximum length of 64kbytes. Setting the length field to 0 results in no data being transferred to or from the MMU.

As with a register access, the command and address fields are transferred first. There is an optional length field transferred after the address. The use of the length field is controlled by the LengthFieldPresent bit in the Function 0 registers, which is cleared on reset.

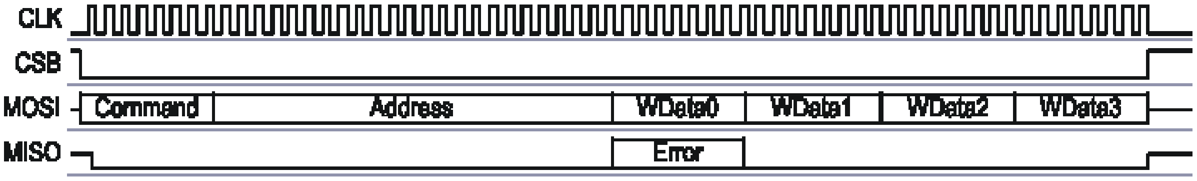


Figure 12: CSPI Burst Write Cycle

7.1.5 CSPI Register Read Cycle

Burst reads have a programmable amount of padding data that is returned by the slave. 0-15 bytes are returned as defined in the BurstPadding register. Following this the Error byte is returned followed by the data. Once the transfer has started, no further padding is needed.

A FIFO within SDIO_TOP will pre-fetch the data. The address is not retransmitted, and is auto-updated within the slave.

The length field is transmitted if LengthFieldPresentin the Function 0 registers is set. In the absence of a length field the CSB signal is used to indicate the end of the burst.

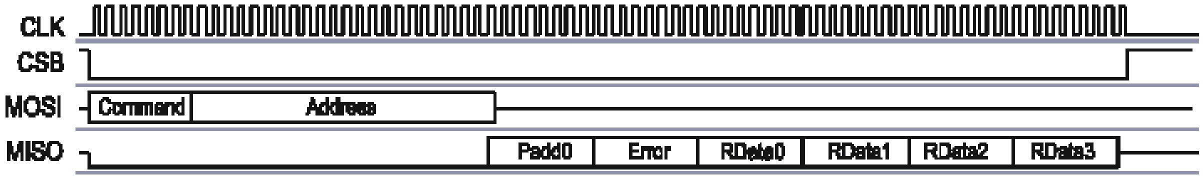


Figure 13: CSPI Burst Read Cycle

7.2 SDIO Interface

This is a host interface which allows a Secure Digital Input Output(SDIO) host to gain access to the internals of the chip. It provides all defined slave modes (SPI, SD 1bit, SD 4bit), but not SD host function.

The function provided includes generating responses to each command in hardware and implementing the state machines defined in the SDIO specification. Within the various modes of operation, it provides initialisation functions (cmds 0, 3, 5, 7, 15, 59) and two other functions:

- Function 1 provides Bluetooth type A support, and follows that specification
- Function 2 provides generic register access(cmd52 (byte read/write))

For more information, see the following specifications:

- SD Specifications Part 1 Physical Layer Specification v.1.10
- SD Specification Part E1 SDIO Specification v.1.10
- SDIO Card Part E2 Type-A Specification for Bluetooth v.1.00

7.2.1 SDIO/CSPI Deep-Sleep Control Schemes

This is the lowest power mode, where the processor, the internal reference (fast) clock, and much of the digital and analogue hardware are shut down. To support this power consumption reduction solution and to prevent any errors arising on the SDIO host interface there are two Deep-Sleep control schemes.

- Scheme 1: The host retransmits any packets that Bluecore was unable to receive as a result of being in Deep-Sleep
- Scheme 2: Introduces additional signaling to prevent the need for retransmissions

During Deep-Sleep the internal reference clock is turned off. However, the host transport protocols (SD/UART/CSPI) are driven from the SDIO clock and so continue to function during Deep-Sleep, enabling access to the function 0 interface, but not the function 1 interface.

7.2.2 Retransmission

Bluecore enters Deep-Sleep whenever it becomes idle after which time, when the host transmits a message on function 1 an illegal command error will be signaled. The activity that this initiates on the SDIO Interface provokes Bluecore into wakeup after which the host re-transmits the original message.

Bluecore will wait for a configurable period of time before re-entering Deep-Sleep, thus ensuring that the original packet is sent/received on retransmission. This control scheme is the default mode of operation.

7.2.3 Signaling

Signalling between the host and Bluecore enables host control over Bluecore Deep-Sleep mode. Consequently the host is aware of when it is appropriate to send Bluecore HCI traffic over function 1.

The signals used by this scheme are Host wakeup and Ready status interrupt select, implemented as register bit in the vendor unique area of function 0.

8 Audio Interfaces

8.1 PCM Interface

The audio Pulse Code Modulation(PCM) interface supports continuous transmission and reception of PCM encoded audio data over Bluetooth.

Pulse Code Modulation (PCM) is a standard method used to digitise audio (particularly voice) for transmission over digital communication channels. Through its PCM interface, WT21 has hardware support for continual transmission and reception of PCM data, thus reducing processor overhead for wireless headset applications. BlueCore6-ROM (QFN) offers a bi-directional digital audio interface that routes directly into the baseband layer of the on-chip firmware. It does not pass through the HCI protocol layer.

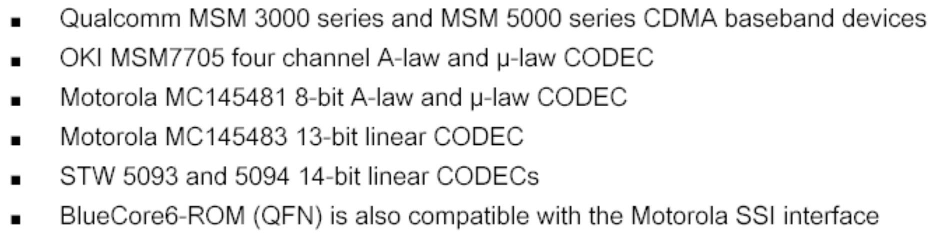
Hardware on WT21 allows the data to be sent to and received from a SCO connection.

Up to three SCO connections can be supported by the PCM interface at any one time.

WT21 can operate as the PCM interface master generating an output clock of 128, 256, 512, 1536 or 2400kHz. When configured as a PCM interface slave, it can operate with an input clock up to 2400kHz. WT21 is compatible with a variety of clock formats, including Long Frame Sync, Short Frame Sync and GCI timing environments.

It supports 13-bit or 16-bit linear, 8-bit µ-law or A-law companded sample formats at 8ksamples/s and can receive and transmit on any selection of three of the first four slots following PCM_SYNC. The PCM configuration options are enabled by setting the PSKEY_PCM_CONFIG32 (0x1b3).

WT21 interfaces directly to PCM audio devices including the following:



- Qualcomm MSM 3000 series and MSM 5000 series CDMA baseband devices
- OKI MSM7705 four channel A-law and µ-law CODEC
- Motorola MC145481 8-bit A-law and µ-law CODEC
- Motorola MC145483 13-bit linear CODEC
- STW 5093 and 5094 14-bit linear CODECs
- BlueCore6-ROM (QFN) is also compatible with the Motorola SSI interface

8.1.1 PCM Interface Master/Slave

When configured as the master of the PCM interface, BlueCore6-ROM (QFN) generates PCM_CLK and PCM_SYNC.

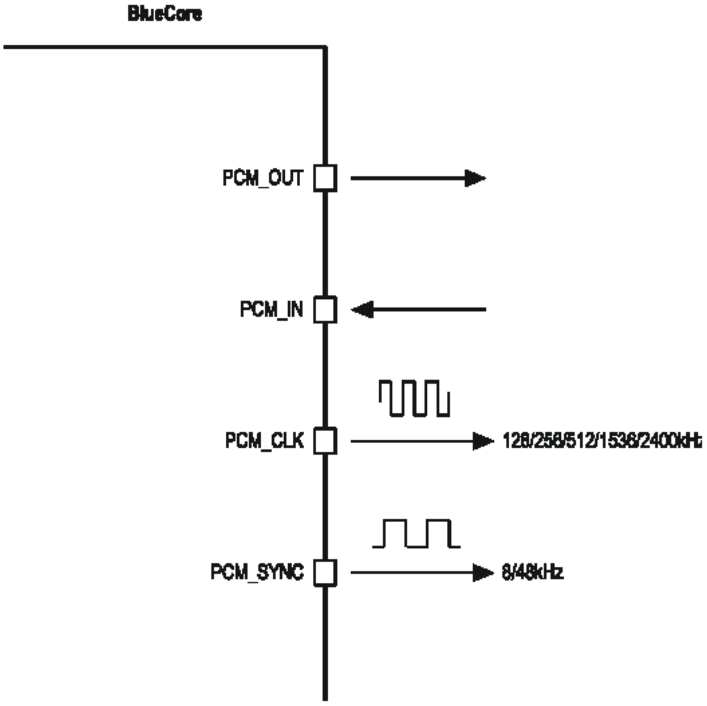


Figure 14: WT21 as a PCM Interface Master

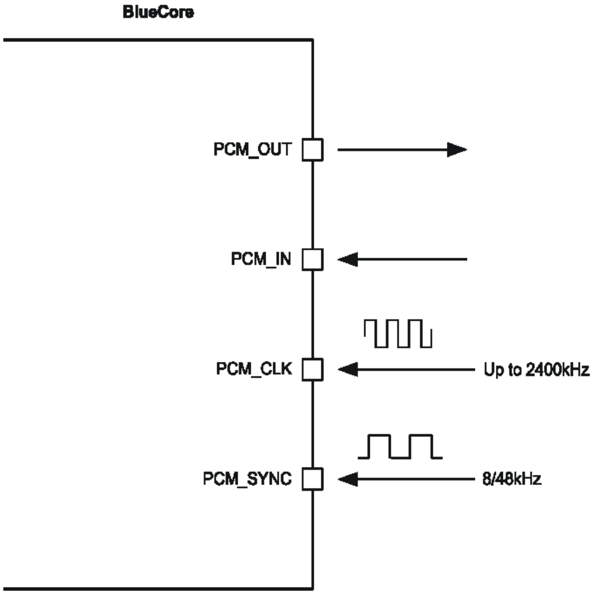


Figure 15: WT21 as a PCM Interface Slave

8.1.2 Long Frame Sync

Long Frame Sync is the name given to a clocking format that controls the transfer of PCM data words or samples. In Long Frame Sync, the rising edge of PCM_SYNC indicates the start of the PCM word. When WT21 is configured as PCM master, generating PCM_SYNC and PCM_CLK, then PCM_SYNC is 8-bits long. When WT21 is configured as PCM Slave, PCM_SYNC may be from two consecutive falling edges of PCM_CLK to half the PCM_SYNC rate, i.e., 62.5µs long.

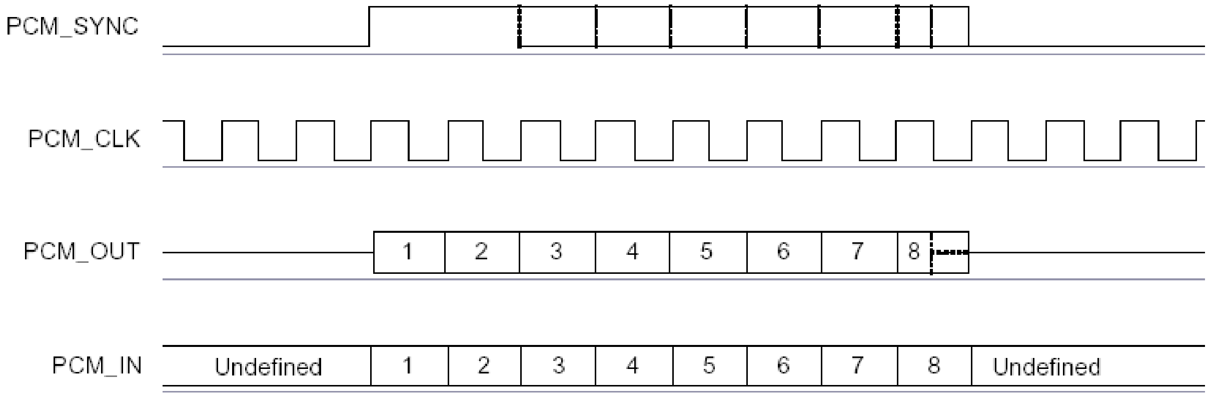


Figure 16: Long Frame Sync (Shown with 8-bit Companded Sample)

BlueCore6-ROM (QFN) samples PCM_IN on the falling edge of PCM_CLK and transmits PCM_OUT on the rising edge. PCM_OUT may be configured to be high impedance on the falling edge of PCM_CLK in the LSB position or on the rising edge.

8.1.3 Short Frame Sync

In Short Frame Sync, the falling edge of PCM_SYNC indicates the start of the PCM word. PCM_SYNC is always one clock cycle long.

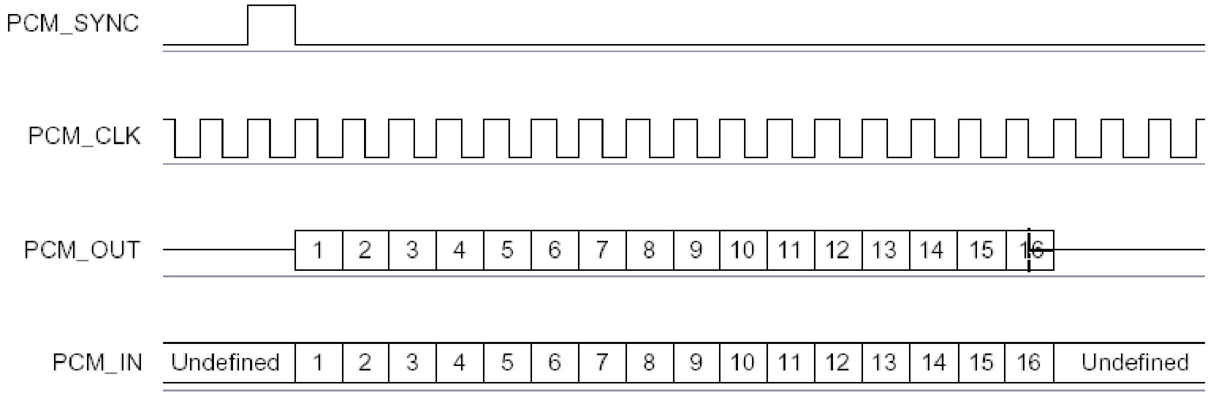


Figure 17: Short Frame Sync (Shown with 16-bit Sample)

As with Long Frame Sync, BlueCore6-ROM (QFN) samples PCM_IN on the falling edge of PCM_CLK and transmits PCM_OUT on the rising edge. PCM_OUT may be configured to be high impedance on the falling edge of PCM_CLK in the LSB position or on the rising edge.

8.1.4 Multi-Slot Operation

More than one SCO connection over the PCM interface is supported using multiple slots. Up to three SCO connections can be carried over any of the first four slots.

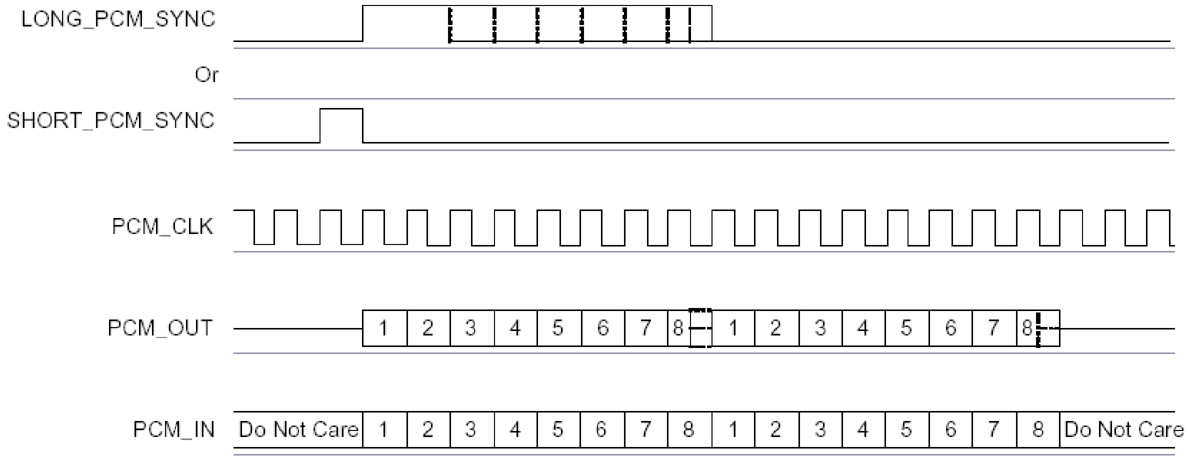


Figure 18: Multi-Slot Operation with Two Slots and 8-bit Companded Samples

8.1.5 GCI Interface

WT21 is compatible with the General Circuit Interface (GCI), a standard synchronous 2B+D ISDN timing interface. The two 64kbps B channels can be accessed when this mode is configured.

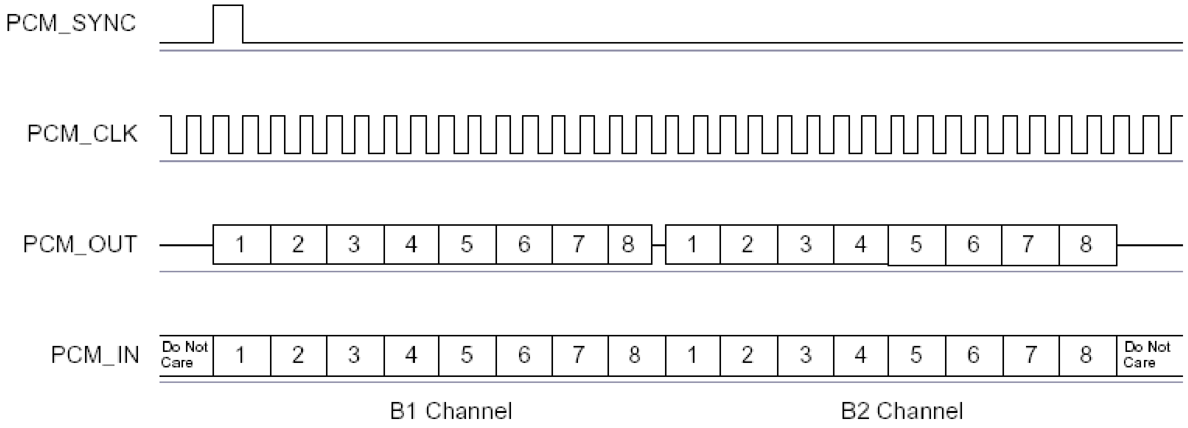


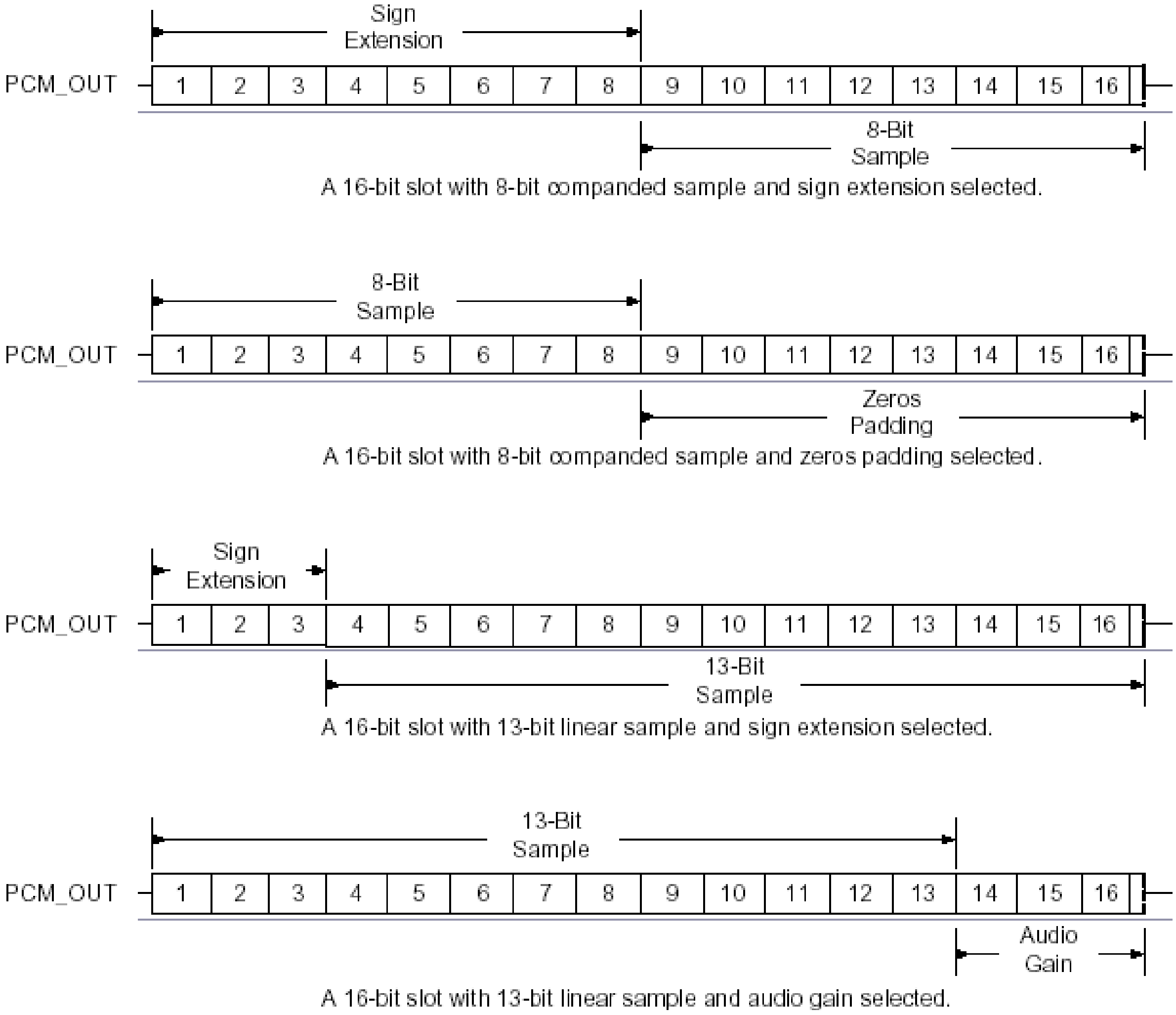
Figure 19: GCI Interface

The start of frame is indicated by the rising edge of PCM_SYNC and runs at 8kHz. With WT21 in Slave mode, the frequency of PCM_CLK can be up to 4.096MHz.

8.1.6 Slots and Sample Formats

WT21 can receive and transmit on any selection of the first four slots following each sync pulse. Slot durations can be either 8 or 16 clock cycles. Durations of 8 clock cycles may only be used with 8-bit sample formats. Durations of 16 clocks may be used with 8-bit, 13-bit or 16-bit sample formats.

WT21 supports 13-bit linear, 16-bit linear and 8-bit μ-law or A-law sample formats. The sample rate is 8ksamples/s. The bit order may be little or big endian. When 16-bit slots are used, the 3 or 8 unused bits in each slot may be filled with sign extension, padded with zeros or a programmable 3-bit audio attenuation compatible with some Motorola CODECs.



A 16-bit slot with 8-bit companded sample and sign extension selected.

A 16-bit slot with 8-bit companded sample and zeros padding selected.

A 16-bit slot with 13-bit linear sample and sign extension selected.

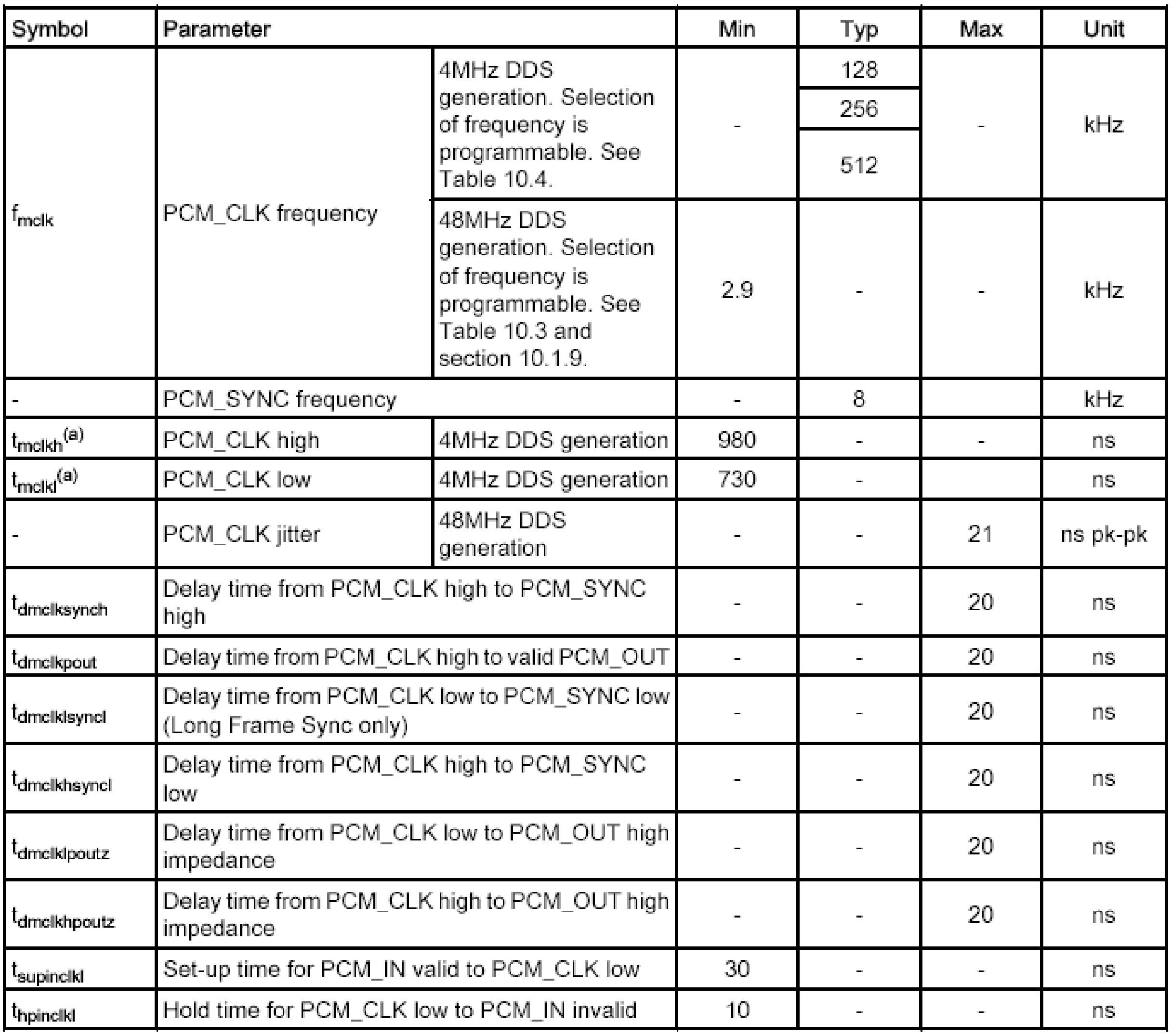
A 16-bit slot with 13-bit linear sample and audio gain selected.

Figure 20: 16-Bit Slot Length and Sample Formats

8.1.7 Additional Features

WT21 has a mute facility that forces PCM_OUT to be 0. In master mode, PCM_SYNC may also be forced to 0 while keeping PCM_CLK running which some CODECS use to control power down.

8.1.8 PCM Timing Information



Symbol	Parameter		Min	Typ	Max	Unit
f_{mclk}	PCM_CLK frequency	4MHz DDS generation. Selection of frequency is programmable. See Table 10.4.	-	128 256 512	-	kHz
		48MHz DDS generation. Selection of frequency is programmable. See Table 10.3 and section 10.1.9.	2.9	-	-	kHz
-	PCM_SYNC frequency		-	8		kHz
t_{mclkh}[(a)]	PCM_CLK high	4MHz DDS generation	980	-	-	ns
t_{mclkl}[(a)]	PCM_CLK low	4MHz DDS generation	730	-		ns
-	PCM_CLK jitter	48MHz DDS generation	-	-	21	ns pk-pk
$t_{dmclksynch}$	Delay time from PCM_CLK high to PCM_SYNC high		-	-	20	ns
$t_{dmclkpout}$	Delay time from PCM_CLK high to valid PCM_OUT		-	-	20	ns
$t_{dmclklsyncl}$	Delay time from PCM_CLK low to PCM_SYNC low (Long Frame Sync only)		-	-	20	ns
$t_{dmclkhsyncl}$	Delay time from PCM_CLK high to PCM_SYNC low		-	-	20	ns
$t_{dmclklpoutz}$	Delay time from PCM_CLK low to PCM_OUT high impedance		-	-	20	ns
$t_{dmclkhpoutz}$	Delay time from PCM_CLK high to PCM_OUT high impedance		-	-	20	ns
$t_{supinclkl}$	Set-up time for PCM_IN valid to PCM_CLK low		30	-	-	ns
$t_{hpinclkl}$	Hold time for PCM_CLK low to PCM_IN invalid		10	-	-	ns

Figure 21: PCM Master Timing

[a)] Assumes normal system clock operation. Figures will vary during low power modes, when system clock speeds are reduced.

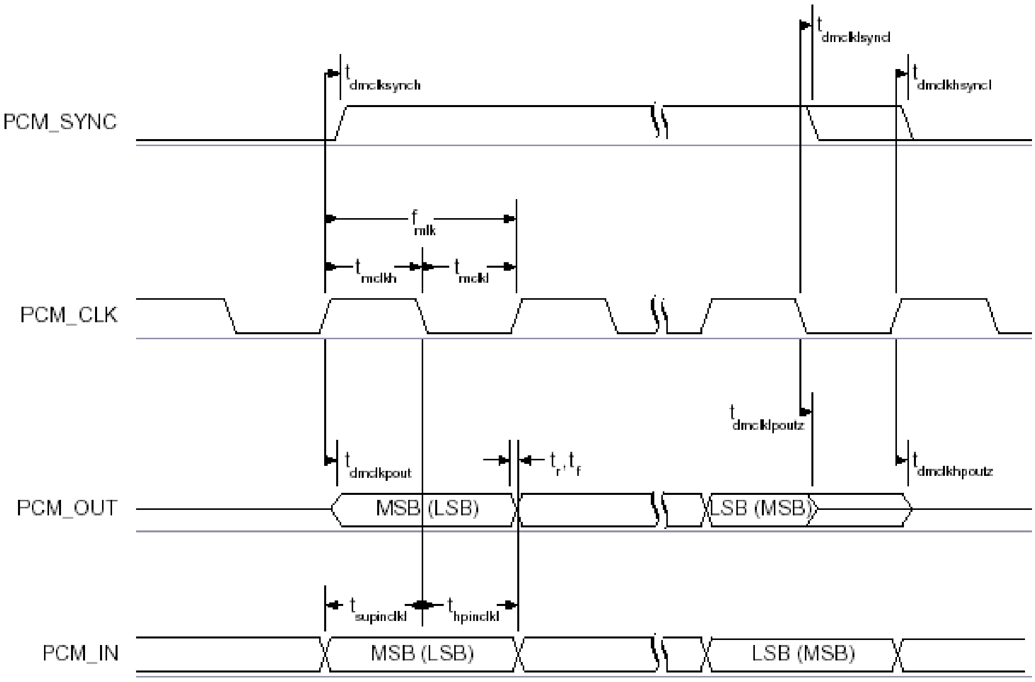


Figure 22: PCM Master Timing Long Frame Sync

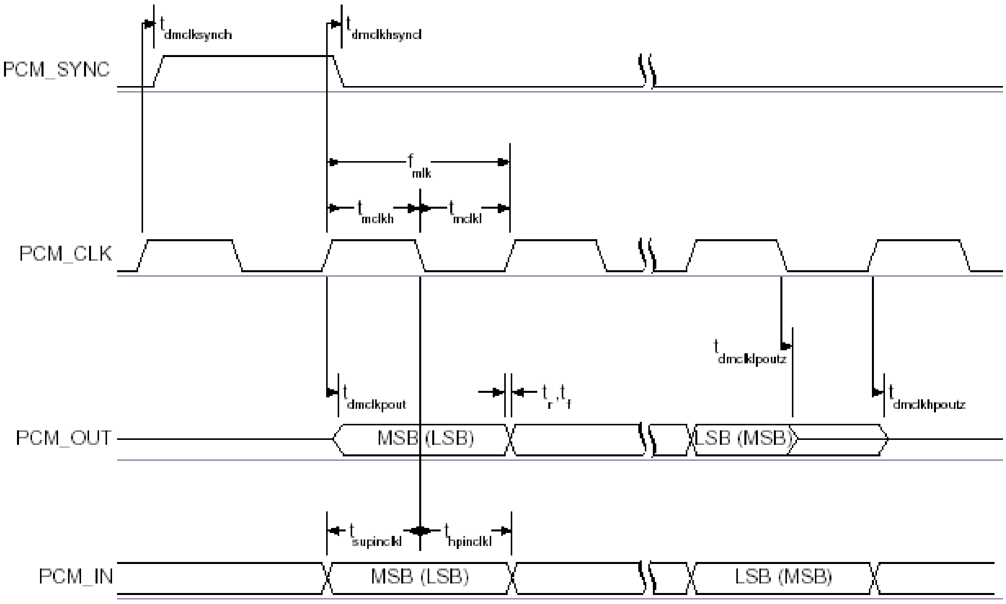


Figure 23: PCM Master Timing Short Frame Sync

Symbol	Parameter	Min	Typ	Max	Unit
f_{sclk}	PCM clock frequency (Slave mode: input)	64	-	2048	kHz
f_{sclk}	PCM clock frequency (GCI mode)	128	-	4096	kHz
t_{sclkl}	PCM_CLK low time	200	-	-	ns
t_{sclkh}	PCM_CLK high time	200	-	-	ns
$t_{hsclksynch}$	Hold time from PCM_CLK low to PCM_SYNC high	30	-	-	ns
$t_{susclksynch}$	Set-up time for PCM_SYNC high to PCM_CLK low	30	-	-	ns
t_{dpout}	Delay time from PCM_SYNC or PCM_CLK whichever is later, to valid PCM_OUT data (Long Frame Sync only)	-	-	20	ns
$t_{dsclkhpout}$	Delay time from CLK high to PCM_OUT valid data	-	-	20	ns
t_{dpoutz}	Delay time from PCM_SYNC or PCM_CLK low, whichever is later, to PCM_OUT data line high impedance	-	-	20	ns
$t_{supinsclkl}$	Set-up time for PCM_IN valid to CLK low	30	-	-	ns
$t_{hpinsclkl}$	Hold time for PCM_CLK low to PCM_IN invalid	30	-	-	ns

Table 13: PCM Slave Timing

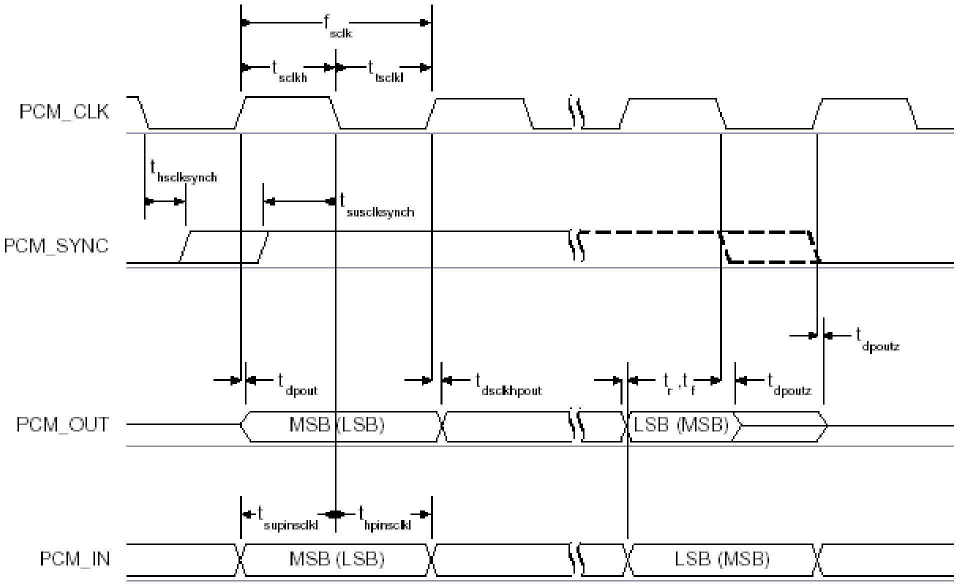


Figure 24: PCM Slave Timing Long Frame Sync

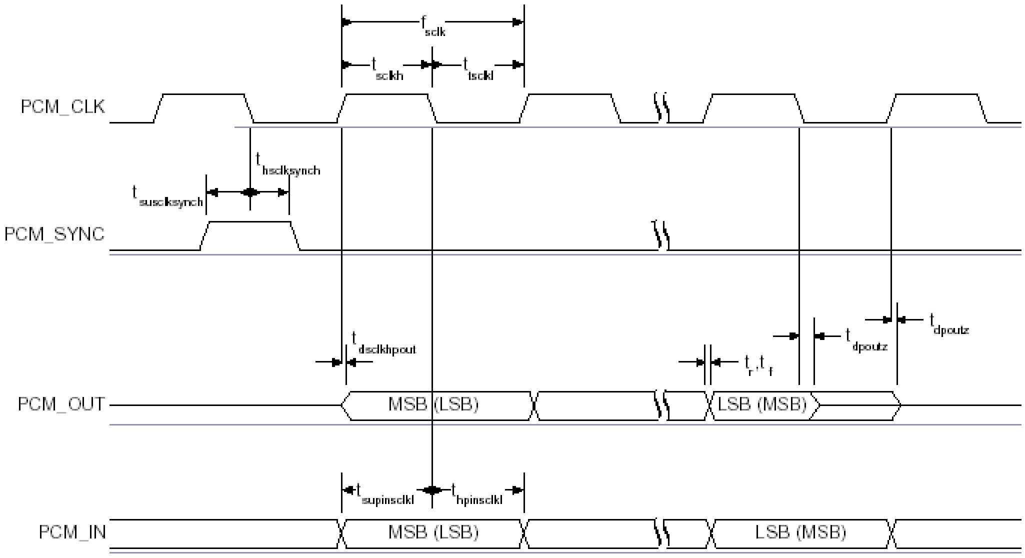


Figure 25: PCM Slave Timing Short Frame Sync

8.1.9 PCM_CLK and PCM_SYNC Generation

WT21 has two methods of generating PCM_CLK and PCM_SYNC in master mode. The first is generating these signals by Direct Digital Synthesis(DDS) from WT21 internal 4MHz clock. Using this mode limits PCM_CLK to 128, 256 or 512kHz and PCM_SYNC to 8kHz. The second is generating PCM_CLK and PCM_SYNC by DDS from an internal 48MHz clock (which allows a greater range of frequencies to be generated with low jitter but consumes more power). This second method is selected by setting bit 48M_PCM_CLK_GEN_EN in PSKEY_PCM_CONFIG32. When in this mode and with long frame sync, the length of PCM_SYNC can be either 8 or 16 cycles of PCM_CLK, determined by LONG_LENGTH_SYNC_EN in PSKEY_PCM_CONFIG32.

Equation XXX describes PCM_CLK frequency when being generated using the internal 48MHz clock:

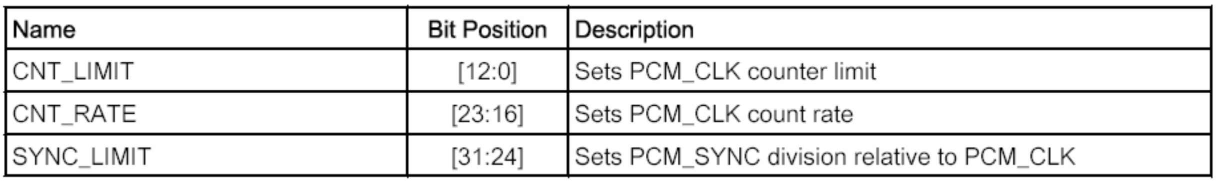
$$f = \frac{CNT\_RATE}{CNT\_LIMIT} \times 24MHz$$

Equation 2: PCM_SYNC Frequency Relative to PCM_CLK

CNT_RATE, CNT_LIMIT and SYNC_LIMIT are set using PSKEY_PCM_LOW_JITTER_CONFIG. As an example, to generate PCM_CLK at 512kHz with PCM_SYNC at 8kHz, set PSKEY_PCM_LOW_JITTER_CONFIG to 0x08080177.

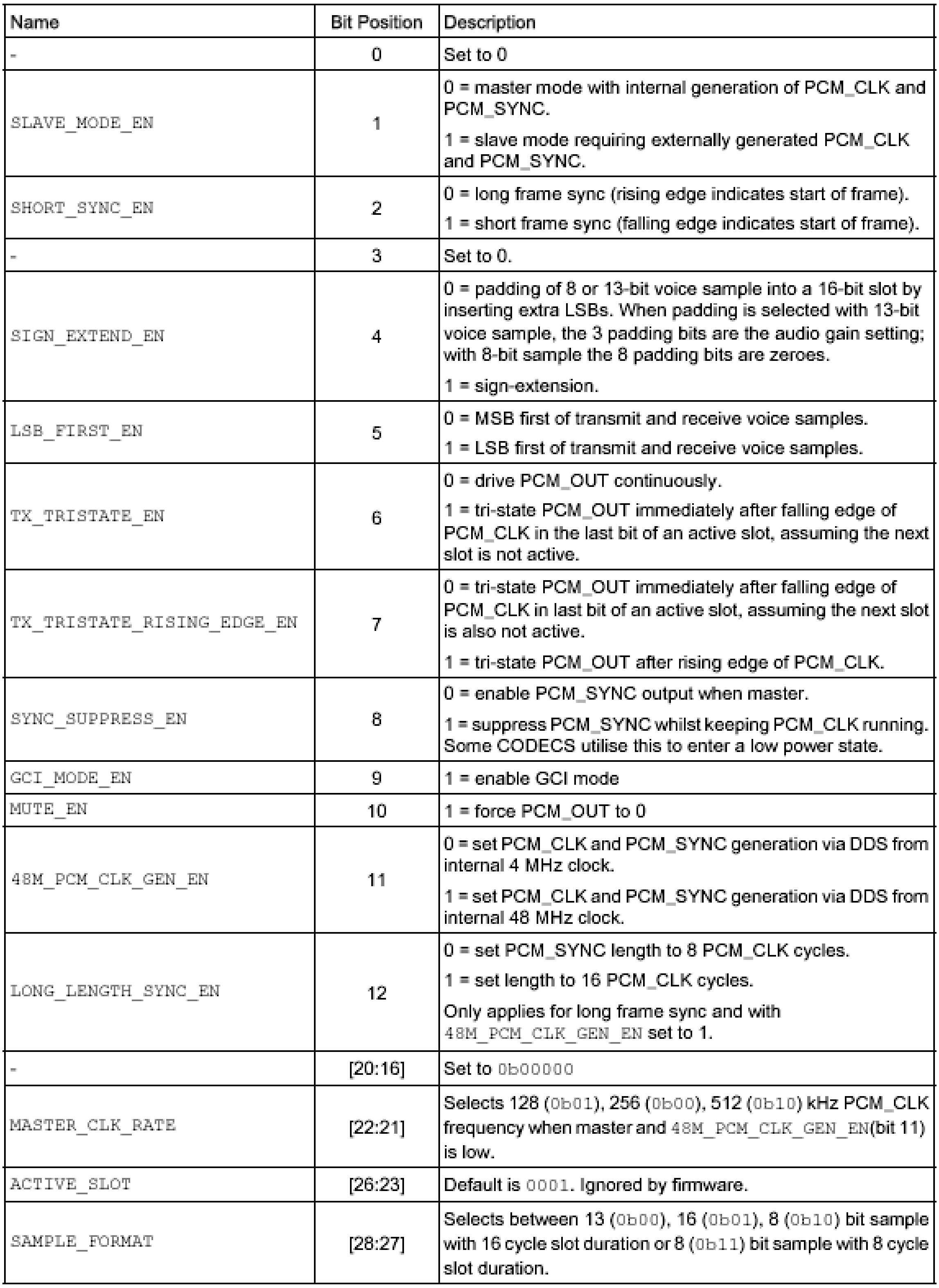
8.1.10 PCM Configuration

The PCM configuration is set using the PS Keys, PSKEY_PCM_CONFIG32 described in Table 14, PSKEY_PCM_LOW_JITTER_CONFIG in Table 13, and PSKEY_PCM_SYNC_MULT in Table 15. The default for PSKEY_PCM_CONFIG32is 0x00800000, i.e., first slot following sync is active, 13-bit linear voice format, long frame sync and interface master generating 256kHz PCM_CLK from 4MHz internal clock with no tri-state of PCM_OUT.



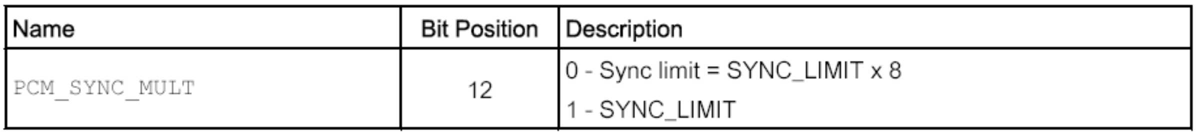
Name	Bit Position	Description
CNT_LIMIT	[12:0]	Sets PCM_CLK counter limit
CNT_RATE	[23:16]	Sets PCM_CLK count rate
SYNC_LIMIT	[31:24]	Sets PCM_SYNC division relative to PCM_CLK

Table 14: PSKEY_PCM_LOW_JITTER_CONFIG Description



Name	Bit Position	Description
-	0	Set to 0
SLAVE_MODE_EN	1	0 = master mode with internal generation of PCM_CLK and PCM_SYNC. 1 = slave mode requiring externally generated PCM_CLK and PCM_SYNC.
SHORT_SYNC_EN	2	0 = long frame sync (rising edge indicates start of frame). 1 = short frame sync (falling edge indicates start of frame).
-	3	Set to 0.
SIGN_EXTEND_EN	4	0 = padding of 8 or 13-bit voice sample into a 16-bit slot by inserting extra LSBs. When padding is selected with 13-bit voice sample, the 3 padding bits are the audio gain setting; with 8-bit sample the 8 padding bits are zeroes. 1 = sign-extension.
LSB_FIRST_EN	5	0 = MSB first of transmit and receive voice samples. 1 = LSB first of transmit and receive voice samples.
TX_TRISTATE_EN	6	0 = drive PCM_OUT continuously. 1 = tri-state PCM_OUT immediately after falling edge of PCM_CLK in the last bit of an active slot, assuming the next slot is not active.
TX_TRISTATE_RISING_EDGE_EN	7	0 = tri-state PCM_OUT immediately after falling edge of PCM_CLK in last bit of an active slot, assuming the next slot is also not active. 1 = tri-state PCM_OUT after rising edge of PCM_CLK.
SYNC_SUPPRESS_EN	8	0 = enable PCM_SYNC output when master. 1 = suppress PCM_SYNC whilst keeping PCM_CLK running. Some CODECS utilise this to enter a low power state.
GCI_MODE_EN	9	1 = enable GCI mode
MUTE_EN	10	1 = force PCM_OUT to 0
48M_PCM_CLK_GEN_EN	11	0 = set PCM_CLK and PCM_SYNC generation via DDS from internal 4 MHz clock. 1 = set PCM_CLK and PCM_SYNC generation via DDS from internal 48 MHz clock.
LONG_LENGTH_SYNC_EN	12	0 = set PCM_SYNC length to 8 PCM_CLK cycles. 1 = set length to 16 PCM_CLK cycles. Only applies for long frame sync and with 48M_PCM_CLK_GEN_EN set to 1.
-	[20:16]	Set to 0b00000
MASTER_CLK_RATE	[22:21]	Selects 128 (0b01), 256 (0b00), 512 (0b10) kHz PCM_CLK frequency when master and 48M_PCM_CLK_GEN_EN(bit 11) is low.
ACTIVE_SLOT	[26:23]	Default is 0001. Ignored by firmware.
SAMPLE_FORMAT	[28:27]	Selects between 13 (0b00), 16 (0b01), 8 (0b10) bit sample with 16 cycle slot duration or 8 (0b11) bit sample with 8 cycle slot duration.

Table 15: PSKEY_PCM_CONFIG32 Description

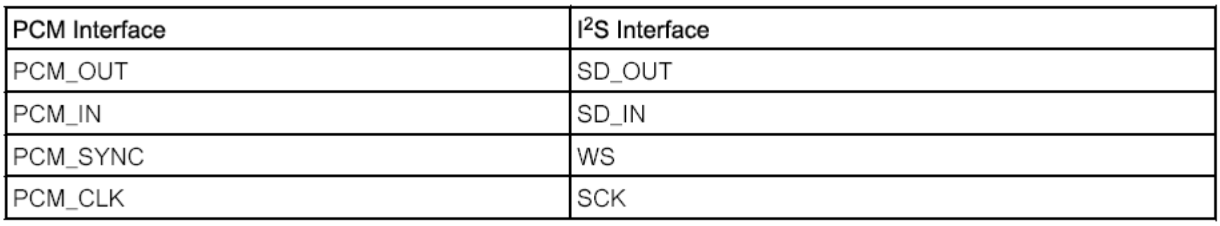


Name	Bit Position	Description
PCM_SYNC_MULT	12	0 - Sync limit = SYNC_LIMIT x 8 1 - SYNC_LIMIT

Table 16: PSKEY_PCM_SYNC_MULT Description

8.2 Digital Audio Interface (I2S)

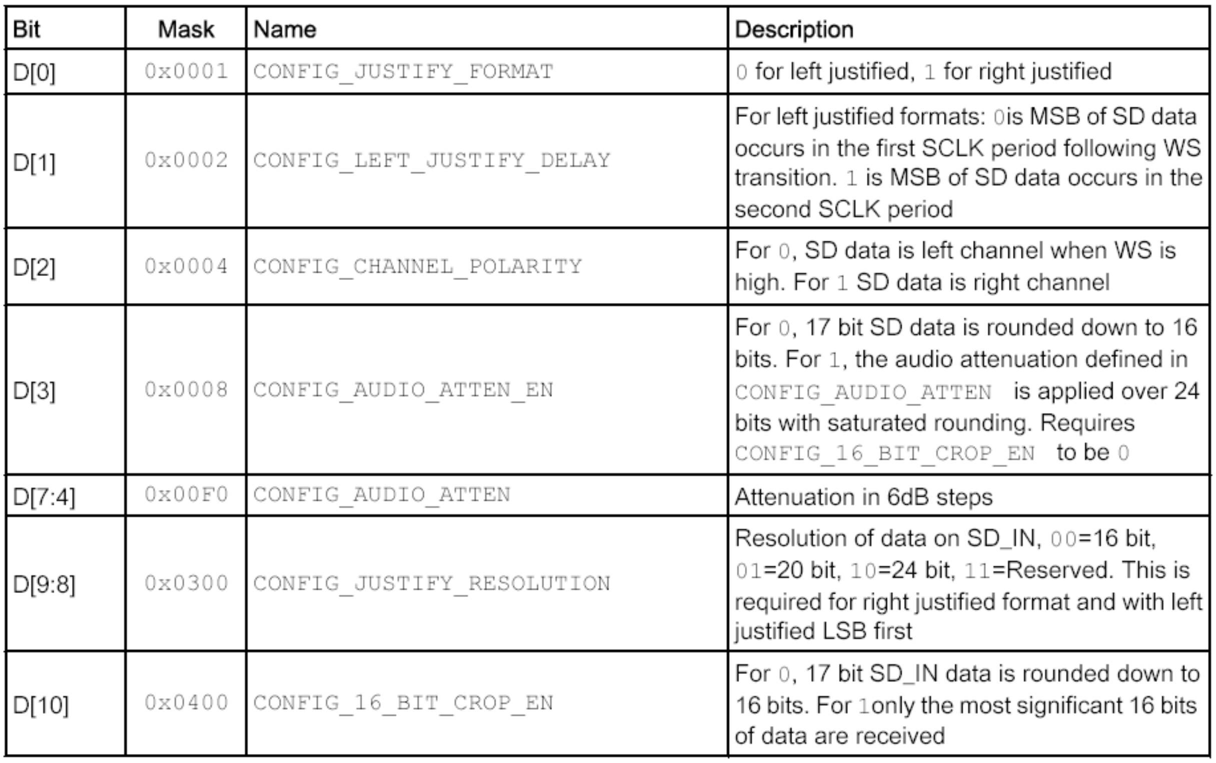
The digital audio interface supports the industry standard formats for I2S, left-justified (LJ) or right-justified(RJ). The interface shares the same pins as the PCM interface, which means each audio bus is mutually exclusive in its usage. Table 17 lists these alternative functions. Figure 26 shows the timing diagram.



PCM Interface	I^2S Interface
PCM_OUT	SD_OUT
PCM_IN	SD_IN
PCM_SYNC	WS
PCM_CLK	SCK

Table 17: Alternative Functions of the Digital Audio Bus Interface on the PCM Interface.

Table 18 describes the values for the PS Key (PSKEY_DIGITAL_AUDIO_CONFIG) that is used to set-up the digital audio interface. For example, to configure an I2S interface with 16-bit SD data set PSKEY_DIGITAL_CONFIG to 0x0406.



Bit	Mask	Name	Description
D[0]	0x0001	CONFIG_JUSTIFY_FORMAT	0 for left justified, 1 for right justified
D[1]	0x0002	CONFIG_LEFT_JUSTIFY_DELAY	For left justified formats: 0is MSB of SD data occurs in the first SCLK period following WS transition. 1 is MSB of SD data occurs in the second SCLK period
D[2]	0x0004	CONFIG_CHANNEL_POLARITY	For 0, SD data is left channel when WS is high. For 1 SD data is right channel
D[3]	0x0008	CONFIG_AUDIO_ATTEN_EN	For 0, 17 bit SD data is rounded down to 16 bits. For 1, the audio attenuation defined in CONFIG_AUDIO_ATTEN is applied over 24 bits with saturated rounding. Requires CONFIG_16_BIT_CROP_EN to be 0
D[7:4]	0x00F0	CONFIG_AUDIO_ATTEN	Attenuation in 6dB steps
D[9:8]	0x0300	CONFIG_JUSTIFY_RESOLUTION	Resolution of data on SD_IN, 00=16 bit, 01=20 bit, 10=24 bit, 11=Reserved. This is required for right justified format and with left justified LSB first
D[10]	0x0400	CONFIG_16_BIT_CROP_EN	For 0, 17 bit SD_IN data is rounded down to 16 bits. For 1only the most significant 16 bits of data are received

Table 18: PSKEY_DIGITAL_AUDIO_CONFIG

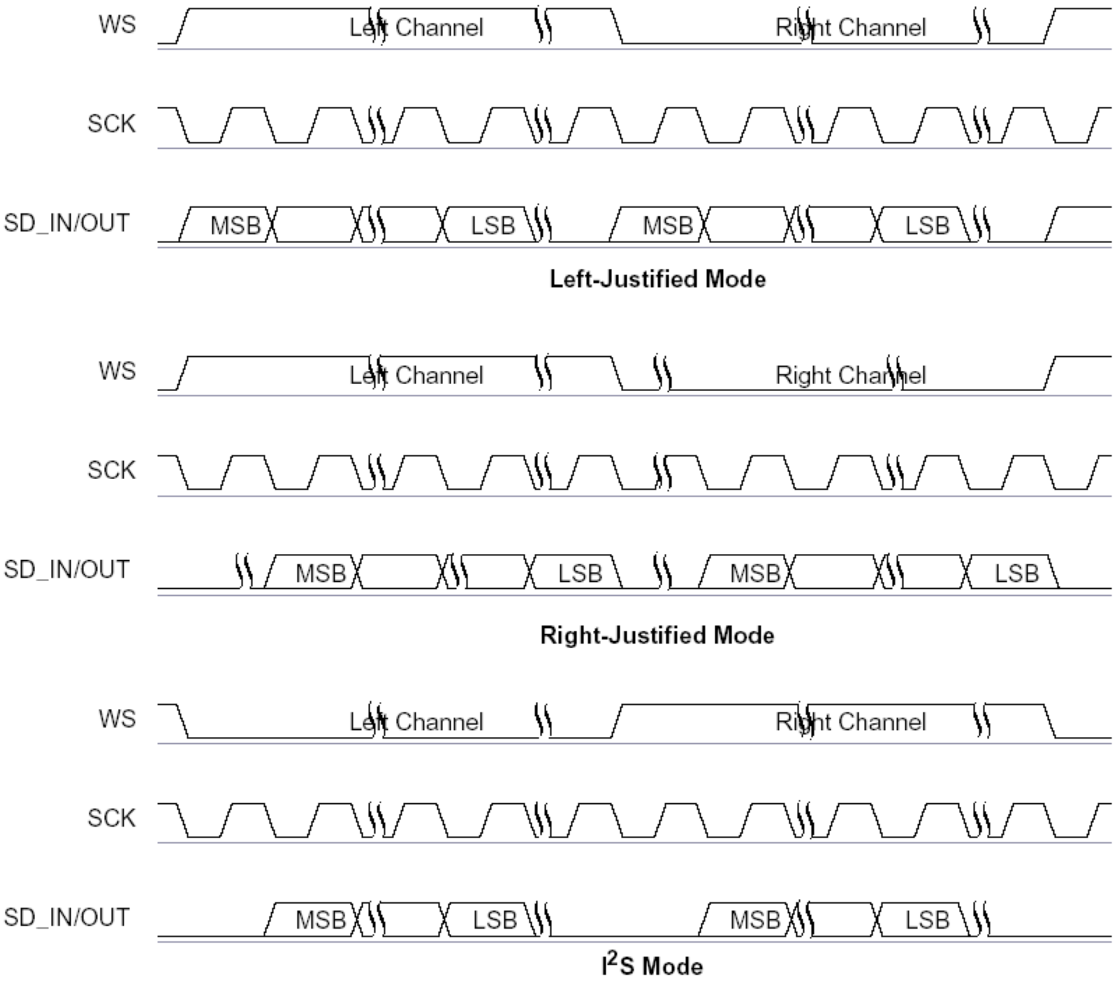


Figure 26: Digital Audio Interface Modes

The internal representation of audio samples within BlueCore6-ROM (QFN) is 16-bit and data on SD_OUT is limited to 16-bit per channel.

Symbol	Parameter	Min	Typ	Max	Unit
-	SCK Frequency	-	-	6.2	MHz
-	WS Frequency	-	-	96	kHz
t_{ch}	SCK high time	80	-	-	ns
t_{cl}	SCK low time	80	-	-	ns
t_{opd}	SCK to SD_OUT delay	-	-	20	ns
t_{ssu}	WS to SCK set-up time	20	-	-	ns
t_{sh}	WS to SCK hold time	20	-	-	ns
t_{isu}	SD_IN to SCK set-up time	20	-	-	ns
t_{ih}	SD_IN to SCK hold time	20	-	-	ns

Table 19: Digital Audio Interface Slave Timing

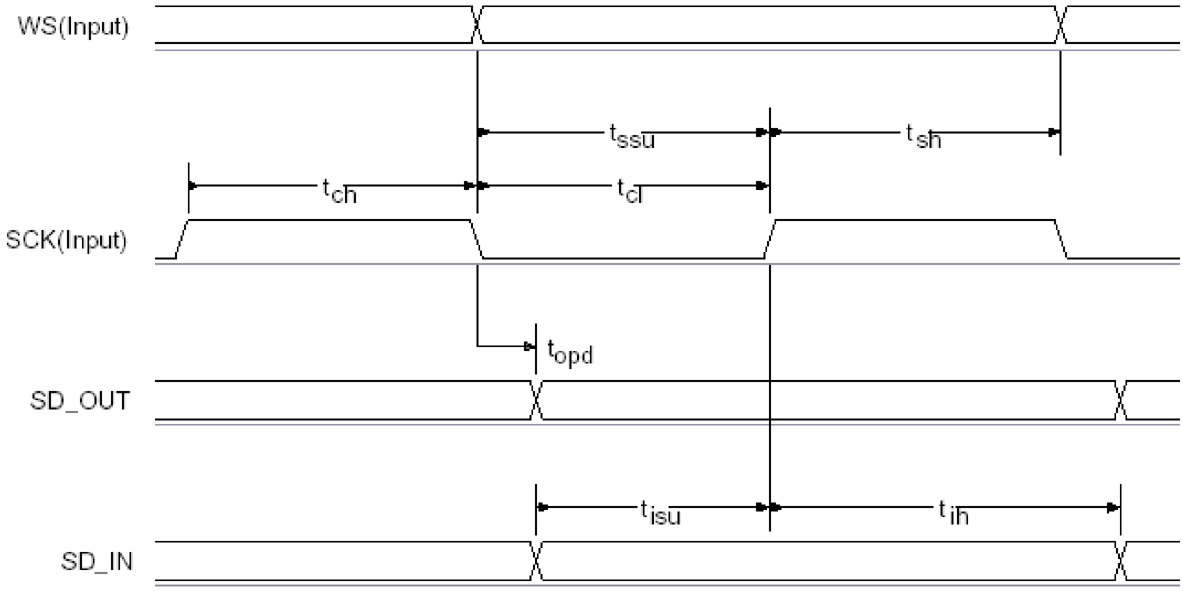


Figure 27: Digital Audio Interface Slave Timing

Symbol	Parameter	Min	Typ	Max	Unit
-	SCK Frequency	-	-	6.2	MHz
-	WS Frequency	-	-	96	kHz
t_{opd}	SCK to SD_OUT delay	-	-	20	ns
t_{spd}	SCK to WS delay	-	-	-	ns
t_{isu}	SD_IN to SCK set-up time	20	-	-	ns
t_{ih}	SD_IN to SCK hold time	10	-	-	ns

Table 20: Digital Audio Interface Master Timing

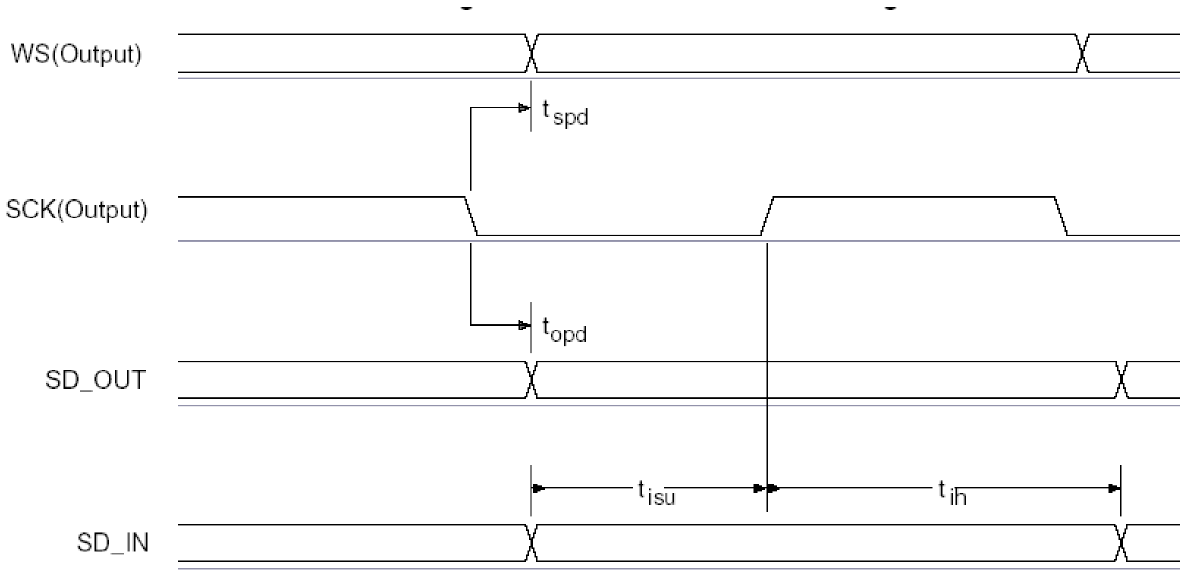


Figure 28: Digital Audio Interface Master Timing

9 Power Control and Regulation

9.1 Power Control and Regulation

WT21 contains two linear regulators.

- A high voltage regulator to generate 1,8 V rail for the module I/Os
- A low voltage regulator to supply the 1,5 V core from the 1,8 V rail

The module can be powered from a high-voltage rail through both regulators and the output of the high-voltage regulator can be used as a supply voltage for the digital interfaces of the module (VDD_PADS). Alternatively VDD_PADS can be supplied by an external voltage source. if the I/O supply VDD_PADS is powered before the 1.5V supplies the digital pads default to their No Core Voltage Reset state.

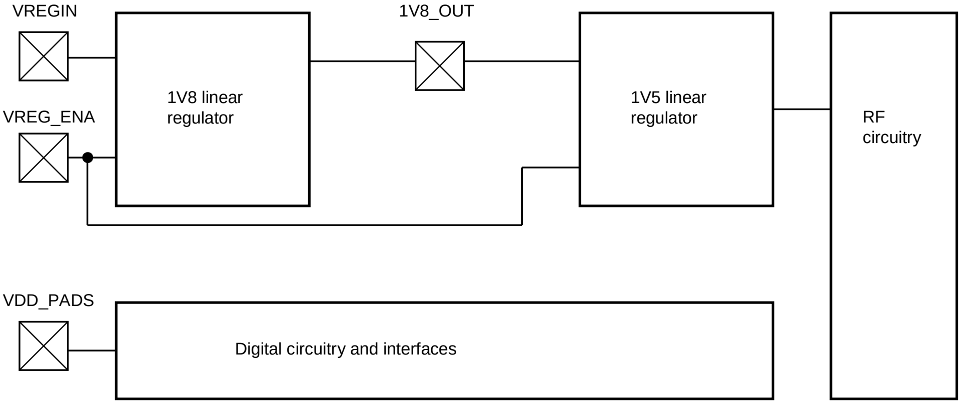


Figure 29: Voltage Regulator Configuration

9.2 VREG_ENABLE

The regulator enable pin VREGENABLE is used to enable the WT21. VREGENABLE enables both the high voltage regulator and the low voltage regulator.

The pin is active high, with a logic threshold of around 1V, and has a weak pull-down. VREGENABLE can tolerate voltages up to 4.9V, so may be connected directly to a battery to enable the device.

9.3 RST#

WT21 may be reset from several sources: RST# pin, power on reset, a UART break character or via a software configured watchdog timer.

The RST# pin is an active low reset and is internally filtered using the internal low frequency clock oscillator. A reset is performed between 1.5 and 4.0ms following RST# being active. It is recommended that RST# be applied for a period greater than 5ms.

The power on reset occurs when the core supply falls below typically 1.24V and is released when core voltage rises above typically 1.31V. At reset the digital I/O pins are set to inputs for bi-directional pins and outputs are tri-state. The pull-down state is shown in Table 21. Following a reset, WT21 assumes the maximum XTAL frequency, which ensures that the internal clocks run at a safe (low) frequency until WT21 is configured for the actual XTAL frequency.

9.4 Digital Pin States on Reset

The digital I/O interfaces on the WT21 device are optimised for minimum power consumption after initialisation of digital interfaces.

Table 21 shows the pin states of WT21 on reset. Pull-up (PU) and pull-down (PD) default to weak values unless specified otherwise.

Pin Name / Group	**I/O Type**	**No Core Voltage Reset**		**Full Chip Reset**	
		Pull R	**I/O**	**Pull R**	**I/O**
Reset / Control					
RST#	Digital Input	PU	Input	PU	Input
Digital Interfaces - SDIO					
SDIO_DATA[3]	Digital bi-directional	PD	Input	PU	Input
SDIO_DATA[2]	Digital bi-directional	PD	Input	PU	Input
SDIO_DATA[1]	Digital bi-directional	PD	Input	PU	Input
SDIO_DATA[0]	Digital bi-directional	PD	Input	PU	Input
SDIO_SD_CS#	Digital bi-directional	PD	Input	PU	Input
SDIO_CMD	Digital bi-directional	PD	Input	PU	Input
SDIO_CLK	Digital bi-directional	PD	Input	PU	Input
PCM Interface					
PCM_IN	Digital Input	PD	Input	PD	Input
PCM_OUT	Digital tri-state output	PD	High impedance	PD	High impedance
PCM_CLK	Digital bidirectional	PD	Input	PD	Input
PCM_SYNC	Digital bidirectional	PD	Input	PD	Input
SPI Interface					
SPI_MOSI	Digital input	PD	Input	PD	Input
SPI_CLK	Digital input	PD	Input	PD	Input
SPI_CS#	Digital input	PU	PU	PU	Input
SPI_MISO	Digital tri-state output	PD	PD	PD	High impedance
PIOs					
PIO[0]	Digital bi-directional	PD	Input	PD	Input
PIO[1]	Digital bi-directional	PD	Input	PD	Input
PIO[2]	Digital bi-directional	PD	Input	PD	Input
PIO[3]	Digital bi-directional	PD	Input	PD	Input
PIO[4]	Digital bi-directional	PD	Input	PD	Input
PIO[5]	Digital bi-directional	PD	Input	PD	Input
PIO[7]	Digital bi-directional	PD	Input	PD	Input
PIO[9]	Digital bi-directional	PD	Input	PD	Input
Clock					
CLK_32K	Digital input	PD	Input	PD	Input

Table 21: Pin States of WT21 on Reset

10 Bluetooth Radio

10.1 Bluetooth Receiver

The receiver features a near-zero Intermediate Frequency (IF) architecture that allows the channel filters to be integrated onto the die. Sufficient out-of-band blocking specification at the Low Noise Amplifier (LNA) input allows the receiver to be used in close proximity to Global System for Mobile Communications(GSM) and Wideband Code Division Multiple Access (W-CDMA) cellular phone transmitters without being desensitised. The use of a digital Frequency Shift Keying(FSK) discriminator means that no discriminator tank is needed and its excellent performance in the presence of noise allows WT21 to exceed the Bluetooth requirements for co-channel and adjacent channel rejection.

For EDR, the Demodulator contains an ADC which is used to digitise the IF received signal. This information is then passed to the EDR modem.

10.1.1 RSSI Analogue to Digital Converter

The Analogue to Digital Converter (ADC) implements fast Automatic Gain Control (AGC). The ADC samples the Received Signal Strength Indicator (RSSI) voltage on a slot-by-slot basis. The front-end LNA gain is changed according to the measured RSSI value, keeping the first mixer input signal within a limited range. This improves the dynamic range of the receiver, improving performance in interference limited environments.

10.2 Bluetooth Transmitter

The transmitter features a direct IQ modulator to minimise the frequency drift during a transmit timeslot, which results in a controlled modulation index. Digital baseband transmit circuitry provides the required spectral shaping.

The internal Power Amplifier (PA) has a maximum output power of +6dBm.

11 Electrical Characteristics

11.1 Absolute Maximum Ratings

Rating		Min	Max	Unit
Storage Temperature		-40	85	℃
IO Voltage	VDD_PADS	-0.4	3.7	V
Supply Voltage	VREG_IN, VREG_ENA	-0.4	4.9	V
Other Terminal Voltages		VSS-0.4	VDD+0.4	V

Table 22: Absolute Maximum Ratings

11.2 Recommended Operating Conditions

Rating		Min	Max	Unit
Operating Temperature Range		TBD	TBD	℃
IO Voltage	VDD_PADS	1.7	3.7	V

Table 23: Recommended Operating Conditions

11.3 Input/Output Terminal Characteristics

11.3.1 Linear Voltage Regulator

Normal Operation	Min	Typ	Max	Unit
Input voltage	2.7	-	4.9	V
Output voltage (I_{load} = 70 mA / VREG_IN = 3.0 V)	1.7	1.8	1.9	V
Temperature coefficient	-250	0	250	ppm/℃
Output noise	-	-	1	mV rms
Load regulation (I_{load} < 70 mA)	-	-	50	mV/A
Settling time	-	-	50	µs
Maximum output current	70	-	-	mA
Minimum load current	5	-	-	µA
Drop-out voltage (I_{load} = 70 mA)	-	-	600	mV
Quiescent current (excluding load, I_{load} < 100µA)	30	40	60	µA
Low Power Mode				
Quiescent current (excluding load, I_{load} < 100µA)	10	13	21	µA
Standby Mode				
Quiescent current (excluding load, I_{load} < 100µA)	1.5	2.5	3.3	µA

Table 24: Recommended Operating Conditions

11.3.2 Digital

Digital Terminals	Min	Typ	Max	Unit
Input Voltage Levels				
V_{IL} input logic level low 1.7V ≤ VDD ≤ 3.6V	-0.4	-	0.25xVDD	V
V_{IH} input logic level high 1.7V ≤ VDD ≤ 3.6V	0.7VDD	-	VDD+0.3	V
Output voltage levels				
V_{OL} output logic level low 1.7V ≤ VDD ≤ 3.6V, (I_o = 4.0 mA)	-	-	0.125	V
V_{OH} output logic level high 1.7V ≤ VDD ≤ 3.6V, (I_o = -4.0 mA)	VDD-0.4	-	VDD	V
Input Tri-state Current with:				
Strong pull-up	-100	-40	-10	µA
Strong pull-down	10	40	100	µA
Weak pull-up	-5	-1	-0.2	µA
Weak pull-down	0.2	1	5	µA
I/O pad leakage curren	-1	0	1	µA
Cl input capacitance	1	-	5	pF

Table 25: Digital terminal electrical characteristics

11.3.3 Reset

Power-on Reset	Min	Typ	Max	Unit
VDD_CORE [a] falling threshold	1.13	1.24	1.3	V
VDD_CORE [a] rising threshold	1.2	1.31	1.35	V
Hysteresis	0.05	0.07	0.15	V

[a] VDD_CORE is a core voltage supplied by the internal 1.5 V voltage regulator.

Table 26: Power on reset characteristics

11.3.4 32 kHz External Reference Clock

Parameter	Conditions/Notes	Specifications			Units
		Min	Nom	Max	
Frequency		32748	32768	32788	Hz
Frequency deviation	@25℃	-	-	20	+/- ppm
Frequency deviation	-25℃ to 85℃	-	-	150	+/- ppm
Input high level	Square wave	0.625xVDD_PADS	-	-	V
Input low level	Square wave	-	-	0.425xVDD_PADS	V
Duty cycle	Square wave	30	-	70	%
Rise and fall time		-	-	50	ns
Integrated frequency jitter	Integrated over the band 200 Hz to 15 kHz	-	-	-	Hz (rms)

Table 27: External Reference Clock

11.4 Power Consumption

Operation Mode	Connection Type	Average	Unit
Page scan, time interval 1.28s	-	0.4	mA
Inquiry and page scan, time interval1,28s	-	0.8	mA
ACL no trafic	Master	4	mA
ACL with file transfer	Master	9	mA
ACL 40ms sniff	Master	2	mA
ACL 1,28s sniff	Master	0.2	mA
eSCO EV5	Master	12	mA
eSCO EV3	Master	18	mA
eSCO EV3 - hands-free - setting S1	Master	18.5	mA
SCO HV1	Master	37	mA
SCO HV3	Master	17	mA
SCO HV3 30ms sniff	Master	17	mA
ACL no traffic	Slave	14	mA
ACL with file transfer	Slave	17	mA
ACL 40ms sniff	Slave	1.6	mA
ACL 1.28s sniff	Slave	0.2	mA
eSCO EV5	Slave	19	mA
eSCO EV3	Slave	23	mA
eSCO EV3 - hands-free - setting S1	Slave	23	mA
SCO HV1	Slave	37	mA
SCO HV3	Slave	23	mA
SCO HV3 30ms sniff	Slave	16	mA
Standby host connection (Deep-Sleep)	-	40	µA
Reset (active low)	-	39	µA

Note:
Conditions 20℃
VREG_IN 3.15V
VDD_PADS 3.15V
UART BAUD rate 115.2 kbps

Table 28: Power Consumption

12 Physical Dimensions

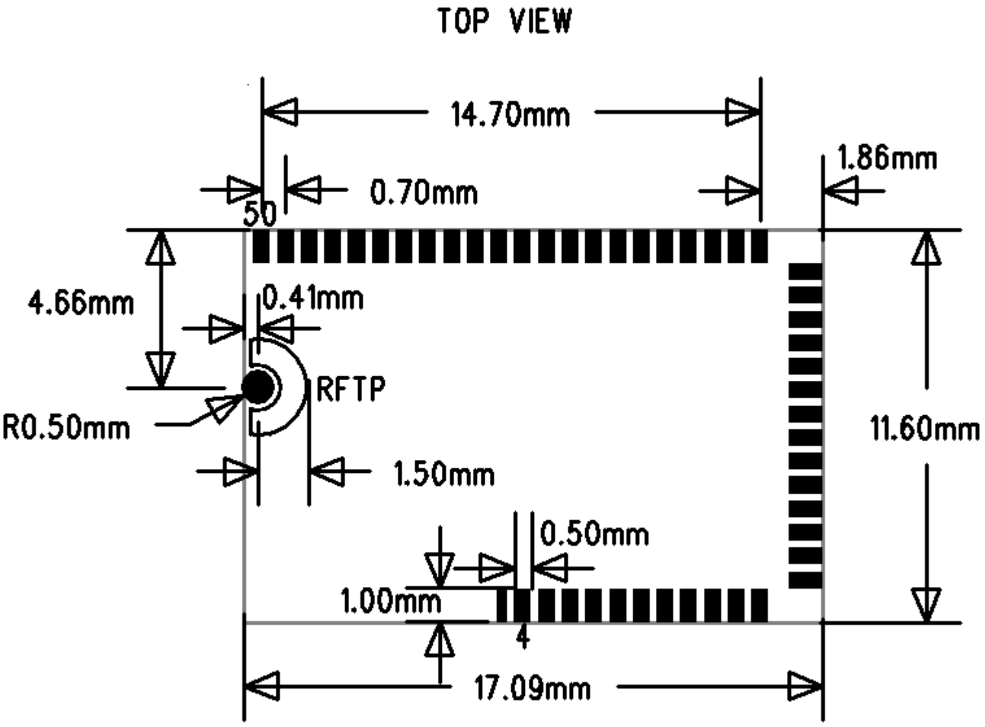


Figure 30: Physical dimensions

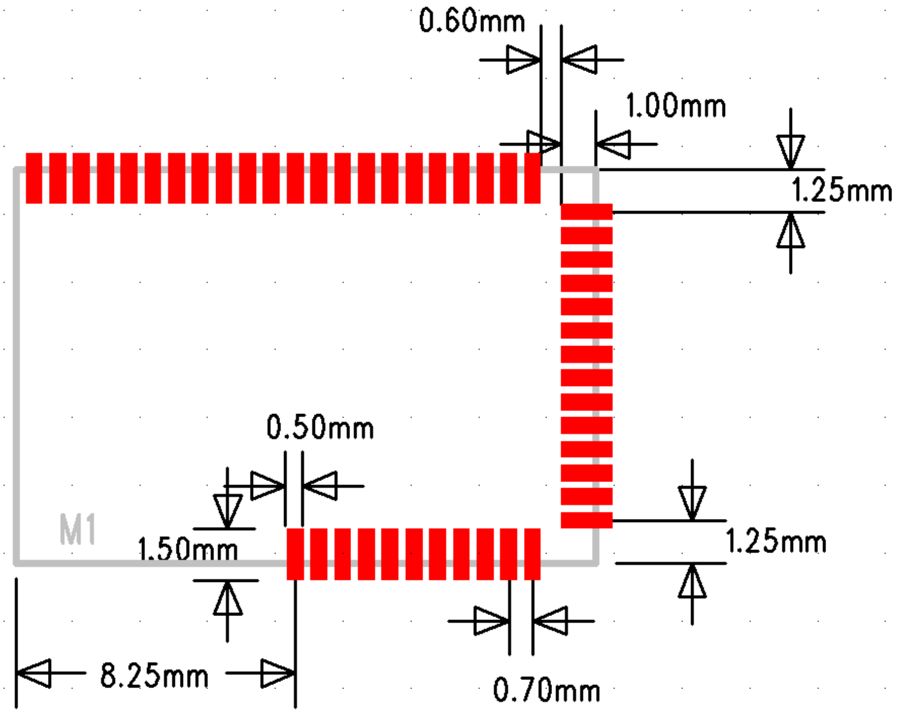


Figure 31: WT21 recommended land pattern

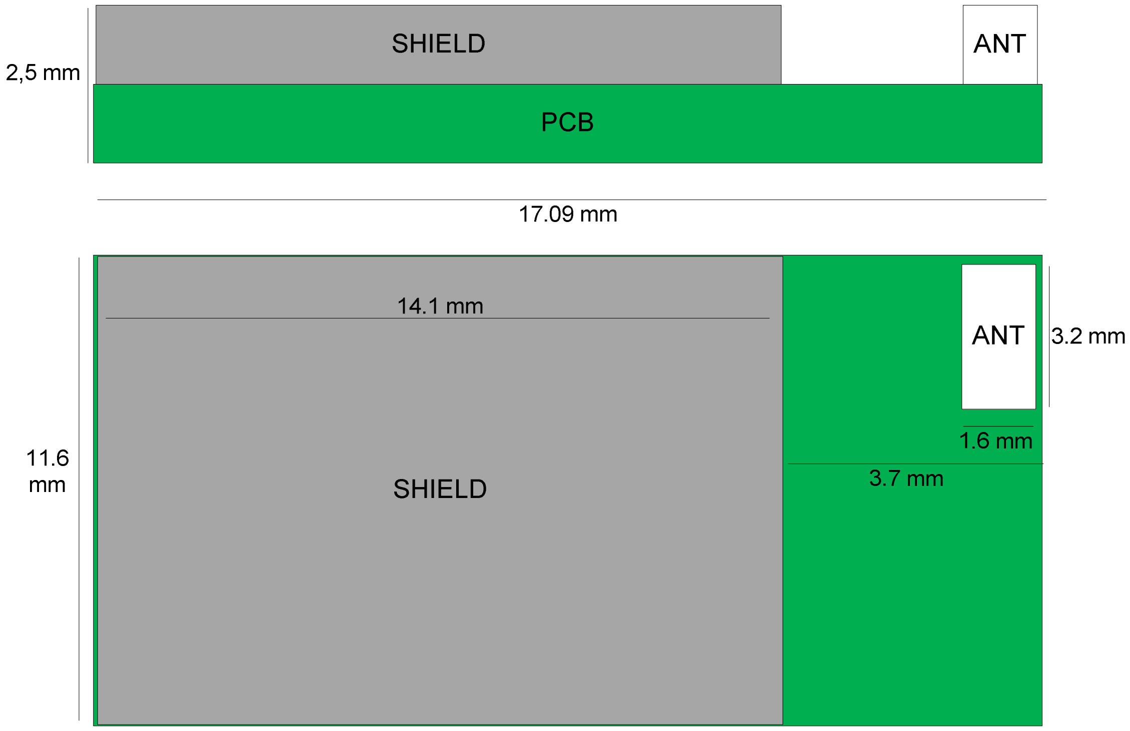


Figure 32: Detailed dimensions

13 Layout Guidelines

13.1 WT21-N

RF output can be taken directly from the RF test point (RFTP) of the module. RFTP has a signal pin surrounded by a ground. Dimensions for the RFTP are shown in the figure below. Use 50 ohm trace to route RF from RFTP. With WT21-A leave RFTP floating and do not place copper directly under RFTP.

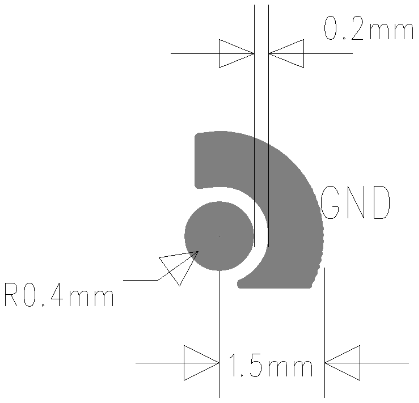


Figure 33: Dimensions of the RFTP

13.2 WT21-A

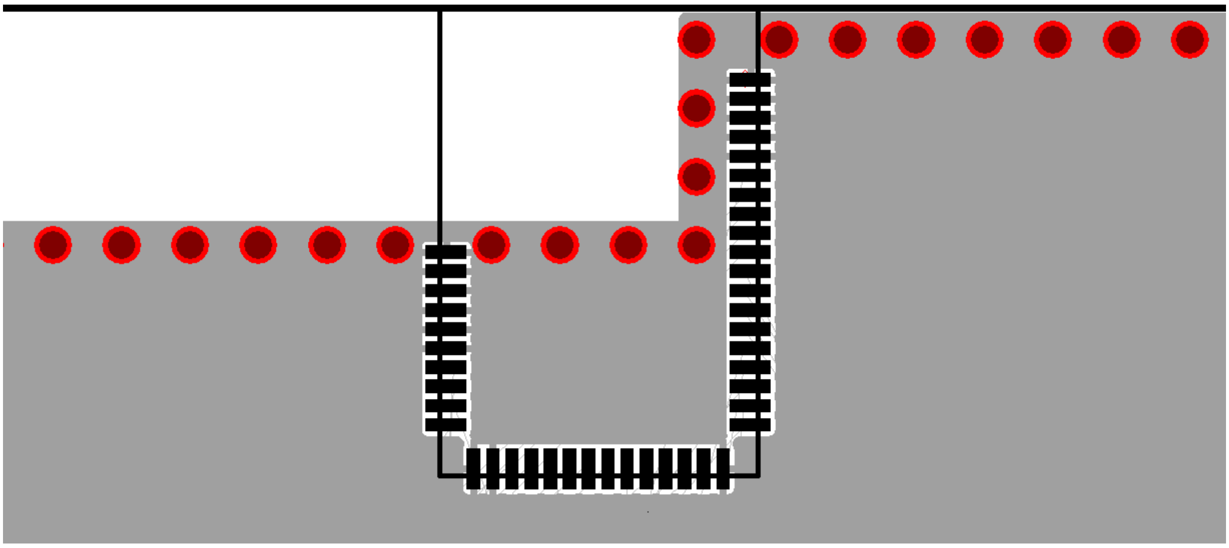


Figure 34: Example layout

The impedance matching of the antenna is design for the evaluation board of WT21. For an optimal performance of the antenna the layout should strictly follow the layout example shown in figure 31 and the thickness of FR4 should be 1,6 mm. Any dielectric material close to the antenna will change the resonant frequency and it is recommended not to place a plastic case or any other dielectric closer than 5 mm from the antenna. If this is not possible, or if using other thickness of FR4 than 1,6 mm, then the antenna can be retuned by removing extra FR4 under the antenna. Please, contact Bluegiga for the details.

Any metal in close proximity of the antenna will prevent the antenna to radiate freely. It is recommended not to place any metal closer than 20 mm from the antenna.

Following recommendations helps to avoid EMC problems arising in the design. Note that each design is unique and the following list do not consider all basic design rules such as avoiding capacitive coupling between signal lines. Following list is aimed to avoid EMC problems caused by RF part of the module.

- Do not remove copper from the PCB more than needed. Use ground filling as much as possible. However remove small floating islands after copper pour.
- Do not place a ground plane underneath the antenna. The grounding areas under the module should be designed as shown in Figure 31.
- When using overlapping ground areas use conductive vias separated max. 3 mm apart at the edge of the ground areas. This prevents RF to penetrate inside the PCB. Use ground vias extensively all over the PCB. All the traces in (and on) the PCB are potential antennas.
- Avoid loops.
- Ensure that signal lines have return paths as short as possible. With sensitive analog signals, such as analog audio, use solid ground plane and make sure that the return path for the signal lines is low impedance and follows the signal lines all the way.

14 Contact Information

Sales: sales@bluegiga.com

Technical support: support@bluegiga.com
http://www.bluegiga.com/techforum/

Orders: orders@bluegiga.com

Head Office / Finland:

Phone: +358-9-4355 060
Fax: +358-9-4355 0660

Street Address:

Sinikalliontie 5A
02630 ESPOO
FINLAND

Postal address:

P.O. BOX 120
02631 ESPOO
FINLAND

Sales Office / USA:

Phone: (781) 556-1039
Bluegiga Technologies, Inc.
99 Derby Street, Suite 200 Hingham, MA 02043