Application Note

94001481800/D
Rev. 0.7, 02/2003

MC72000 Implementation for Cellular Phones

MOTOROLA intelligence everywhere™ | digital dna™

Contents

1 Introduction

This document provides a step-by-step introduction on how to design in using the Motorola MC72000 Bluetooth solution. It clarifies that it is very simple to design in, test, and perform production with the Bluetooth Platform Solution from Motorola. Motorola has a long background in providing stable and well-proven solutions, optimized for high volume production.

This document also provides an overview for the business decision team as well as for the system architecture, design-in, test, and production teams. More specific design-in details can be found in the referenced documentation.

1.1 References

Throughout this document, references are made to the following additional documents:

- *MC72000 Integrated Bluetooth Radio Data Sheet* (document number MC72000/D)
- *Production Test, Bluetooth Platform Solution from Motorola Application Note* (document number 94001481100)
- *HCI Implementation Notes* (document number 94001480101)
- *Vendor-Specific HCI Reference* (document number 79000001800)
- *Motorola Bluetooth Embedded Protocol Stack Product Brief* (document number BTEPSPB/D)
- *Motorola Bluetooth Embedded Protocol Stack SDK* (document number 94001150000)
- *Motorola Bluetooth Solutions, Bluetooth Qualification Application Note* (document number AN2386/D)
- *MC71000/MC72000 Wakeup, Reset and Host Clock Request Sequences Application Note* (document number AN2340/D)
- *MC71000/MC72000 Non-Volatile Memory Size Selection Application Note* (document number 94001481600)
- *UART/SSI Configuration User's Guide* (document number 94001481900)
- *Bluetooth IC File System File Formats Application Note* (document number 94001481200)

- *MC71000/MC72000 Code Patching Application Note* (document number 94001481700)
- *Bluetooth Park Mode Application Note* (document number AN2337/D)
- *Motorola Bluetooth File System, Overview Application Note* (document number 94001481001)
- *Motorola Bluetooth File System, Host Based One File Application Note* (document number 94001481004)
- *Motorola Bluetooth File System, Host Based General Application Note* (document number 94001481003)
- *Motorola Bluetooth File System, Embedded File System Application Note* (document number 94001481002)
- *Examples of Bluetooth PCB Antennas Application Note* (document number 94001481400)
- *Configuration Manager User's Guide* (document number 94001480200)
- *DemoBench User's Guide* (document number 94001480300)
- *HCI Terminal User's Guide* (document number 94001480100)
- *RadioTest User's Guide* (document number 94001480800)
- *MC72000 Development Kit User's Guide* (document number 94001620000)

1.2 Mnemonics

The following mnemonics are used in this document:

ACL	Asynchronous Connection Less link
EEPROM	Electrically Erasable/Programmable Read Only Memory
HCI	Host Controller Interface/Telegram format
LSB	Least Significant Byte
MSB	Most Significant Byte
NVM	Non-Volatile Memory
RX	Receive(r)
SAT	Sector Allocation Table
SCO	Synchronous Connection Oriented link
SEEPROM	Serial Electrically Erasable/Programmable Read Only Memory
SRAM	Static Random Access Memory
TBD	To Be Defined
TRL	TRansport Layer
TX	Transmit(ter)
UART	Universal Asynchronous Receiver Transmitter

2 MC72000 Host-Based Design-In Overview

The purpose of this application note is to look into the factors involved with designing in the Motorola MC72000 Bluetooth device into a mobile phone. This document describes an application for a host-based design with a Bluetooth baseband and radio integrated as one chip. This document describes the power supply, need for different clocks, RF design to the antenna, and the communications interface to the host.

2.1 Design-In Steps

This reference design and document are based on a host-optimized system. All required supplies and system clocks are explained but are to be provided by the host system. The following steps will take you through how to interface and configure the MC72000 specific to your host and its needs.

- Section 3, "MC72000/Mobile Phone Interfaces and Configuration"
- Section 4, "Clocking Requirements and Power Saving Modes"
- Section 5, "Voltage Requirements"
- Section 6, "File System Requirements", see also *MC71000/MC72000 Code Patching Application Note* (document number 94001481700)
- Section 7, "Antenna Design/Type"
- Section 8, "Layout Guidelines and Considerations"
- Section 10, "BQB Offering from Motorola"
- Section 11, "Production and Test"

2.2 Other Useful Design-In Information

In addition to the steps mentioned above, other useful design-in information is provided in the following sections:

- Section 9, "Testing with Bluetooth Prototype Board"
- Section 12, "Bluetooth Upper Stack", including information on host applications and interfacing to the upper stack through the API
- Section 13, "Development, Test, and Evaluation Tools"

3 MC72000/Mobile Phone Interfaces and Configuration

Figure 1 shows a possible cell phone to Bluetooth interface. Note that different cell phones can and will have different implementations.

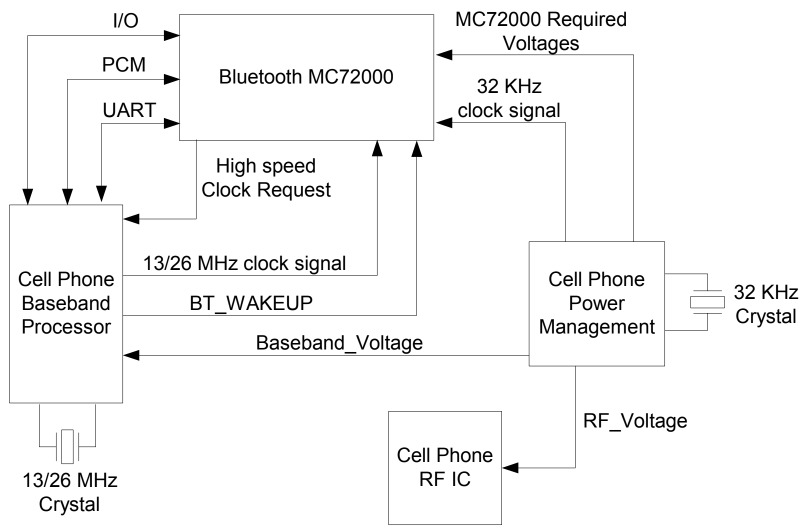


Figure 1. MC72000/Mobile Phone Interfaces and Configuration

3.1 Main Interfaces to Host Device

3.1.1 UART Interface

The host interface is a normal serial UART interface, through which all communication from application, upper stack, and file system passes. The MC72000 UART interface can be configured to be a two-, four-, or six-wire interface. Also, the default baud rate upon start up can also be configured. For more details, see *Motorola Bluetooth File System, Host Based General Application Note* (document number 94001481003) and *UART/SSI Configuration User's Guide* (document number 94001481900).

3.1.2 SSI Interface

A four-wire SSI interface is typically interfaced to the host. The SSI interface is configurable to allow up to three SCO links. Also, the MC72000 can be configured to allow a choice of CODEC drivers, Network/normal mode operation, master/slave functionality, and other timing. This flexibility allows the SSI interface to be configured to suit the needs of the host without a ROM spin. For more details, see *UART/SSI Configuration User's Guide* (document number 94001481900).

3.1.3 Audio Signaling

The MC72000 provides two sources for providing digital audio signals to the host. This may be via the UART Transport Layer or SSI interface. Both interfaces support continuous slope delta modulation (CVSD), μ-Law, and A-Law coding schemes.

3.2 Host and Bluetooth Wake Up Interface

The Bluetooth and host wake up functionality can also be configured with the MC72000. Some of the functionality can even be combined with the UART interface. For more details, see *UART/SSI Configuration User's Guide* (document number 94001481900).

3.2.1 Host and Bluetooth Wake Up Signaling

Most mobile phones use temperature-controlled crystal oscillators in their design. Driving these crystals can consume a lot of current when in use, so they should be disabled whenever possible. Power consumption is always an important issue with portable devices. The MC72000 has implemented easy and configurable ways to ensure the Bluetooth device and the host are able to go into low power modes without disturbing each other. For more detailed information, including timing diagrams and sequencing information, see *MC71000/MC72000 Wakeup, Reset, and Host Clock Request Sequences Application Note* (document number AN2340/D).

3.2.1.1 Clock Request Signal

When the MC72000 needs the high-speed clock from the host, it will send a clock request signal to the host. The host can then send the appropriate clock signal to the Bluetooth device. When the clock request signal is not asserted, the host does not need to send the clock to the Bluetooth device. At this time, the MC72000 is in sleep mode, running from the provided 32-kHz clock signal. For more detailed information, including timing diagrams and sequencing information, see *MC71000/MC72000 Wakeup, Reset, and Host Clock Request Sequences Application Note* (document number AN2340/D).

3.2.1.2 Host Wake Up Signal (Optional)

Assertion of the clock request signal does not always imply that the Bluetooth device has data to send to the host. For example, in cases such as an inquiry and page scan, the Bluetooth device only needs the high-speed clock to transmit and receive, not for data transfer, although data transfer could immediately follow.

In such cases, if the host must wake up completely in order to send the clock signal to the Bluetooth device, this could become an unnecessary power drain on the host. Therefore, the MC72000 has the capability to configure and send a host wake up signal, which will assert only when the Bluetooth device has data to send to the host. Therefore, the host does not have to wake up completely to send the Bluetooth clock. Instead, the clock request signal mentioned above can be sent to request the clock signal only and the host can maintain in its own low power state, only having to wake up for the Bluetooth device when the host wake up signal is asserted. The use of this signal can be selected using the software tool. For more details, see *UART/SSI Configuration User's Guide* (document number 94001481900).

3.2.1.3 Bluetooth Wake Up Signal

The Bluetooth device will go into sleep mode whenever possible to save power consumption. If the host needs to wake up the Bluetooth device, it can assert the Bluetooth wake up signal. The host can also see if the Bluetooth device is asleep by checking the clock request signal. If the clock request signal is not asserted, the host is not sending the Bluetooth device a high speed clock signal. Therefore, the Bluetooth device is in sleep mode.

3.2.2 Power Saving Modes

This section describes the MC72000 power saving modes.

3.2.2.1 Bluetooth Specification Power Modes

The Motorola MC72000 supports the sniff and hold power savings mode features defined in the Bluetooth standard. Park mode is not supported due to interoperability problems known throughout the Bluetooth community. A good alternative to park mode already exists in the Bluetooth specification, consisting of using sniff mode with the master transmitting NULL packets. It is preferable to use this alternative solution for a lot of reasons, for example, no known interoperability problems, and a less complex system, which means lower cost. Also, the sniff mode uses the same amount of power or less compared to park mode. For more information on Motorola's alternative to park mode, see *Bluetooth Park Mode Application Note* (document number AN2337/D).

3.2.2.2 Motorola MC72000 Power Saving Modes

The Motorola MC72000 also incorporates an enhanced power saving feature called sleep mode. This is a function where the MC72000 will power down in the absence of any activity and run only on its 32 kHz clock.

For more information on the sleep mode sequence and timing, see *MC71000/MC72000 Wakeup, Reset, and Host Clock Request Sequences Application Note* (document number AN2340/D).

3.3 Other Interface Configurations Available

All configurations mentioned in Section 3 can be made via software tools. The following sections briefly describe some of the possible changes. For more details, see *UART/SSI Configuration User's Guide* (document number 94001481900).

3.3.1 Configure Radio

The MC72000 can accept a variety of different input frequencies for its high-speed clock. For more detailed information about the frequencies and signal requirements, see Section 3.3.3, "MC72000 Host-Based Reference Schematic". Using the UART/SSI configuration software tool, any of the supported input frequencies can be entered to configure the MC72000. For more information, see *UART/SSI Configuration User's Guide* (document number 94001481900).

3.3.2 Configure GPIO/SSI/UART

The software tools also contain the ability to configure the GPIOs, SSI, and UART pins. For additional information, see *UART/SSI Configuration User's Guide* (document number 94001481900).

3.3.2.1 GPIOs

You can select which GPIOs you want to use through the HCI interface. For UART, you can select the following:

- Default start-up baud rate
- Wake-up pin (B11, B12, or C3)

3.3.2.2 SSI Interface

Select CODEC driver: Xemics300x, Motorola, or Generic SSI. If you chose the Generic SSI driver, you can configure the following:

1. Normal or network mode.
2. If network mode is selected, the number of time slots and which one to use for RX/TX data can be configured. If normal mode is selected, the frame rate divider can be configured.
3. Slave or master mode.
4. If master mode is selected, the bit clock rate can be selected. If slave mode is selected, the master’s bit clock rate can be configured.
5. Clock polarity for RX and TX.
6. LSB or MSB first.
7. Frame sync configuration: long/short, early/long, and polarity.
8. DAC, ADC gain, and side-tone gain.

3.3.3 MC72000 Host-Based Reference Schematic

Figure 2 shows the MC72000 host-based reference schematic.

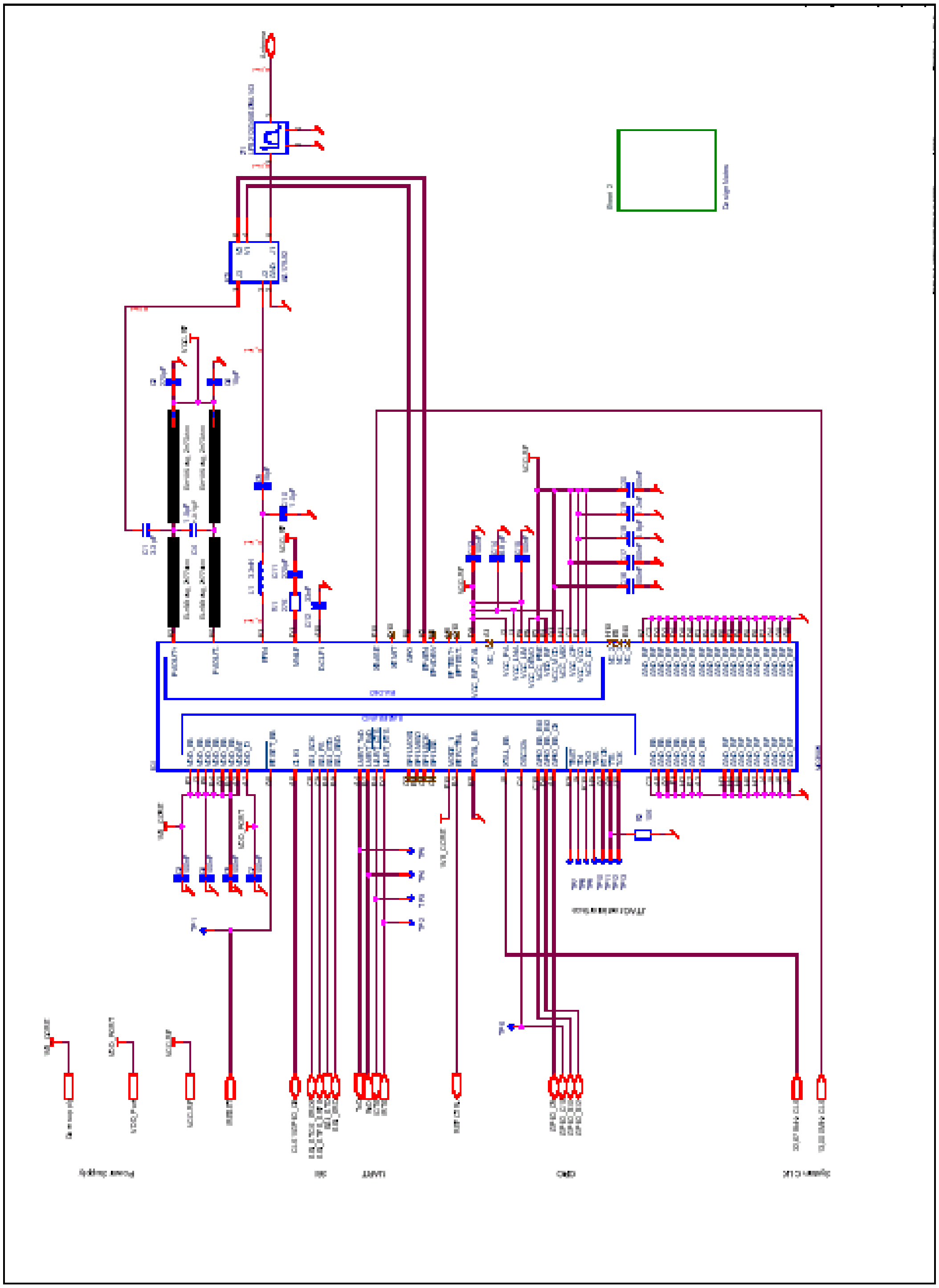


Figure 2. MC72000 Host-Based Reference Schematic

4 Clocking Requirements and Power Saving Modes

This section describes the clock requirements for the MC72000 and also the power saving modes.

4.1 Clocking Requirements for Mobile Phones

Two features are prominent in the design of a mobile phone with regard to the systems clock:

- Low cost, high accuracy oscillators
- Low level, sinusoidal signals

A 13 Mhz clock is typically used as the system clock for most dual- or tri-band cellular phones. 2.5G and 3G technologies are using higher system clocks of 26 MHz. The MC72000 is capable of a clock input across 12-26 MHz to cover both frequencies seamlessly. See Section 4.2, "Clocking Requirements for Motorola MC72000," for more information.

To ensure good performance of the mobile phone, the quality of the clock signal must be very good and the amplitude of the signal kept as low as possible. As a part of the layout, care should be taken to keep the signal free of as much noise (harmonic content) as possible. This will also minimize radiation sidebands.

4.2 Clocking Requirements for Motorola MC72000

In this section, the Motorola MC72000 describes the two clock signals that come from the host and their requirements.

4.2.1 High Speed Clock Requirements

Referring to the reference schematic in Section 3.3.3, "MC72000 Host-Based Reference Schematic", the signal has to be AC-coupled into MC72000, pin E10, XBASE. Pin D10, Xemit, is left unconnected.

If using a 13 Mhz reference, the internal trim capacitance should be set to 0 pF to obtain the smallest load.

Changing the reference frequency to a clock other than 13 Mhz requires small changes to the loop filter resistor and capacitor. Also, a value stored in non-volatile memory will need to be changed and configured using the software tools. For more information on needed component changes, see *MC72000 Integrated Bluetooth Radio Data Sheet* (document number MC72000/D). Table 1 shows the high speed clock requirements.

Table 1. High Speed Clock Requirements

Clock	Input to Bluetooth Device	Requirements
Frequency (1)	12 – 26 MHz clock, in an integral multiple of 20 kHz	+/-20 ppm over temperature and aging (Bluetooth spec). The signal may be sine- or square wave.
Voltage Swing	0.2 to 1Vpp	The DC blocking capacitor may be used as a capacitive voltage divider together with the input and PCB capacitance, in case of too much voltage swing.
Duty Cycle	TBD	
Edge Jitter	TBD	
Logic High Level	TBD	
Logic Low Level	TBD	

Table 1. High Speed Clock Requirements (Continued)

Clock	Input to Bluetooth Device	Requirements
Rise Time	TBD	
Fall Time	TBD	

4.2.2 Low Speed Clock Requirements

The reference should be DC-coupled to pin K8, EXTAL_BB. Pin J8, XTAL_BB, should be left unconnected.

Minimum AC input voltage (Vpp) should be: 200 mVpp, when AC coupled. Table 2 shows the low speed clock requirements.

Table 2. Low Speed Clock Requirements

Clock	Input to Bluetooth Device	Requirements/Comments
Frequency (1)	32.768 kHz	+/-250 ppm over temperature and aging (Bluetooth spec)
Voltage Swing	0 to VCCcore, (min/max specification: 0.8*VCCcore to VCCcore)	A voltage divider may be used in the case of a reference with higher voltage swing. In case of a lower swing reference, the internal oscillator inverter may be biased into linear mode by connecting 10 Mohm between K8 and J8, and AC-coupling the signal into pin K8.
Duty Cycle	TBD	
Edge Jitter	TBD	
Logic High Level	TBD	
Logic Low Level	TBD	
Rise Time	TBD	
Fall Time	TBD	

5 Voltage Requirements

This section outlines the voltage requirements for the MC72000. All the supplies used in this design are assumed to be stabilized. If this host cannot provide the needed voltages, a discrete LDO can be used. This section will make reference to ports and pins that can also be located on the reference schematic in Section 3.3.3, “MC72000 Host-Based Reference Schematic”.

Table 3 shows the required voltages and other requirements. For more detailed information, see *MC72000 Integrated Bluetooth Radio Data Sheet* (document number MC72000/D).

5.1 Baseband Voltages

The GPIO and interfacing pads for the UART, SSI, SPI, and GPIOs of the Motorola MC72000 are all powered using the same port, “VDD_Port”. With regard to other interface signals:

- BT_WAKEUP: The BT_WakeUp signal is configured as a GPIO. Therefore, this signal should be interfaced at the host with the same voltage as that applied to VDD_Port.
- HOST WAKEUP (Optional): The Host_WakeUp signal is configured as a GPIO. Therefore, this signal should be interfaced at the host with the same voltage as that applied to VDD_Port.
- REFCTRL (Clock Request): The REFCTRL signal is also used as the host clock request signal. The clock request signal is part of the Bluetooth interface, which is powered at the core supply voltage.
- RESET: The RESET signal is used to reset the Bluetooth device. This signal is tied to the same voltage as the baseband core and Bluetooth interface.

5.2 RF Voltages

The pin labeled “VCC-RF” supplies the radio part of the MC72000. This voltage needs to be smoothed with a low noise level since it could affect the TX phase noise at the RF output. Also, VCC-RF should not be shared with other digital circuits.

If the noise mainly consists of a single frequency, as is typical with SMPS power supplies, and so on, Figure 3 may be used to estimate reasonable noise levels, showing maximum nominal levels versus frequency. Noise is measured at the MC72000 pins. For burst noise, a sine wave with the same peak value and slope may be used to simulate the burst.

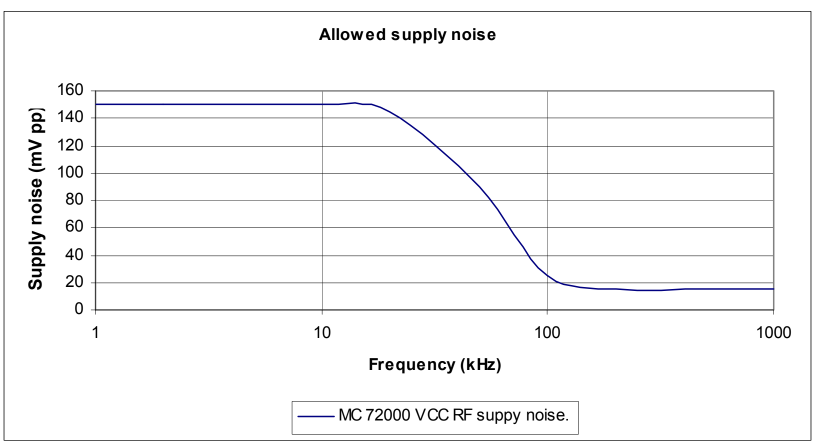


Figure 3. Reasonable Noise Levels for Radio Voltage Supply

Table 3. Voltage Supply Requirements

Voltage Supply	Labeled on Reference Design	Host Requirements	Comments
Baseband (1)	Core Supply	1.65-1.95V(2)	
Radio	VCC-RF	2.5-3.10V	Smoothed with low noise level, can not be shared
UART, SSI, SPI, GPIOs(3)	VDD_Port	1.8-3.60V(2)	Typical voltages of 1.8 V, 2.7 V, 3.0 V or 3.3 V.

1 The baseband core supply voltage also power the RESET and JTAG and Bluetooth interface ports.

2 A 5% (pp) noise voltage on this supply is allowable, as long as the supply voltage including the noise at all times stay within the stated limits.

3 GPIOs included in this section include GPIO_B10, GPIO_B12, GPIO_C9 which can be used for BT_WAKEUP and host wake up signaling.

5.3 Other Signaling Interface Information

REFCTRL is part of the Bluetooth interface port (which is supplied between 1.65–1.95 V). This signal can be used as a host wake up signal and it should be ensured that the host provides corresponding voltage.

Unconnected pins: As seen in the reference design, some pins are left unconnected, such as the SPI port. If any pins are not used in an application, internal pull up resistors will be used to ensure they are always in a known state. This ensures that external resistors are not needed as well.

6 File System Requirements

A small amount of non-volatile memory (NVM) is needed for Bluetooth devices to store variables such as the Bluetooth address, which is unique for every Bluetooth device. For more details on the amount of NVM required, see *MC71000/MC72000 Non-Volatile Memory Size Selection Application Note* (document number 94001481600). In a host-based system, this small amount of non-volatile memory will be located on the host device, typically as flash attached to the host. An existing or new file system, placed on the host, can be used for storing the file(s) to be downloaded. The MC72000 offers two options for downloading the file system into own RAM.

For more information, see the file system documentation:

- *Motorola Bluetooth File System, Overview Application Note* (document number 94001481001)
- *Motorola Bluetooth File System, Host-Based General Application Note* (document number 94001481003)
- *Motorola Bluetooth File System, Host-Based One File Application Note* (document number 94001481004)
- *Motorola Bluetooth File System, Embedded File System Application Note* (document number 94001481002)

See also the following documents:

- *HCI Implementation Notes* (document number 94001480101)
- *Motorola Bluetooth Solutions, Bluetooth Qualification Application Note* (document number AN2386/D)

6.1 One-File File System

A single file can be downloaded from host to the MC72000 RAM immediately after a reset. The host will send only one file, containing all the files for the file system to be used on the MC72000 IC. This single file contains the files that store all information about patches, OEM code, radio parameters, BD address, etc., needed by the MC72000 IC. Only a small piece of software needs to be implemented on the host for the host initialization sequence. All requests and data regarding this file are sent over the serial UART interface as vendor-specific HCI telegrams. For more information, see *Motorola Bluetooth File System, Host-Based One File Application Note* (document number 94001481004).

6.2 Complete File System

A complete file system, with all files downloaded from the host can be also be implemented. The difference between the complete and one-file file system is that the host will need to implement in software the ability to send, receive, and sort a number of additional commands corresponding to this file system. The host will then have the capability to read and write different Bluetooth parameters stored in this file system. All requests and data regarding this file are sent over the serial UART interface as vendor-specific HCI telegrams. For more details, see the file system documentation listed above in Section 6, "File System Requirements".

7 Antenna Design/Type

The following section provides an overview of the antenna guidelines and layout for the MC72000 IC. For more information than what is provided here, see *Examples of Bluetooth PCB Antennas Application Note* (document number 94001481400).

7.1 General Antenna Needs

The antenna should have a nominal impedance of 50 ohm and a return loss better than -10 dB across the 2400-2480 MHz band (VSWR better than 2:1). A worse return loss can be tolerated but should be avoided for performance reasons.

The antenna may be of either a ceramic type or a PCB structure (F antenna, patch, loop, etc.).

The ceramic types may initially seem smaller, but consider the recommended keep-out area around the antenna, which may be several times larger than the antenna itself.

The PCB antennas require more board space, but once designed, they are almost "free of charge". They often exhibit better efficiency than the chip types.

There is a size/performance trade-off in all antennas. Even when downscaling antennas where only a small PCB area is concerned, efficiency drops off. Some small chip antennas may have efficiencies as low as 20% (-7 dBi average gain). This 7 dB loss works on both RX and TX, giving marginal performance.

For example, take two units fitted with a reasonably good antenna, with -1 dBi average gain (80% efficiency). The antennas are replaced by a -7 dBi (20%) antenna. As a result, the useful range between these two modules drops to 25% of the original range with the good antennas. This example clearly shows that high antenna performance is most important in obtaining good range.

7.2 Selecting Antenna

Antennas are very sensitive to the surroundings. Depending on PCB layout and size, surrounding plastic and metallic objects, and so on, retuning will almost always be required in the final product to assure high performance.

Hints for selecting a good antenna:

- Good efficiency, with average gain approach 0 dBi (can never exceed 0 dBi).
- Reasonable matching, return loss less than -10 dB.
- Smooth radiation pattern without too many sharp dips.

NOTE:

The RF matching components shown in the reference schematic (see Section 3.3.3, "MC72000 Host-Based Reference Schematic") are preliminary but show typical values to be used. As usual with RF, fine-tuning will probably be needed, depending on actual layout and PCB build-up. The wires marked 50_ohmx are tracks with characteristics impedance at 50 ohms.

8 Layout Guidelines and Considerations

This section discusses the layout and considerations which should be taken into account during the layout according to the reference design provided in this document (see Section 3.3.3, "MC72000 Host-Based Reference Schematic").

8.1 General PCB Layout Techniques

All tracks drawn at a PCB board should be as short as possible, and loops should be avoided. Circuits generating noise should be kept away from sensitive circuits. Use one of the layers in the PCB build as a ground plane as this will give a short way from the GND pins or decoupling capacitors to GND. A good place to start is to place the vital components first, then components with a special function. A CODEC, for instance, should be placed near the transducers.

The next step is the decoupling capacitors, which should be placed as closely as possibly to the pins where they are needed.

8.2 Radio

"C5" in the reference schematic (see Section 3.3.3, "MC72000 Host-Based Reference Schematic") should preferably be a low tolerance unit. The usual tolerance of ±0.25 pF may cause 1-2 dB tolerance on the output power, but this may be entirely acceptable in some applications.

"R1" and "R2" in the reference schematic attenuate spurious resonance in the balun structure. If FR4 board material is used, the losses will normally provide the attenuation necessary so that these resistors can be left out.

The balun is made by the four "thick black" wires. The value stated at the black wire shows the electrical length and the characteristics impedance. The physical length and width of the track depends on the actual PCB build/materials. The wires marked 50_ohmx are tracks with a characteristics impedance at 50 ohms.

8.3 Power Supply

When designing the layout for the power supply, the power track into the LDOs should be as wide as possible to minimize the impedance. The decoupling capacitors should be placed close to the input and output pins at the LDOs. GND Vias need to be placed close to the GND pins and connected to the ground plane.

Power supplies to audio and radio circuits need to be separated and designed with two individual LDOs. Power can also be distributed as planes like the ground plane, but you should be aware that power planes should be kept in areas where they are supposed to be used—there should be no “loose ends”.

8.4 Baseband

The MC72000 IC needs decoupling capacitors. These capacitors must be placed close to the power pins.

9 Testing with Bluetooth Prototype Board

This section provides an overview of how to verify and test your own Motorola Bluetooth Platform implementation. It provides some useful information that should be considered before the initial prototype board layout is made. The test overview covers hardware and software from HCI and below.

There are two options when testing your Bluetooth prototype board.

- The first requires you to make a break-in between the UART signaling from the Bluetooth device and host, and enable these signals to be connected to a PC on which Motorola’s software tools are running and from which the application software can be downloaded to the Bluetooth RAM.
- The second option requires you to program your host device to download software into the Bluetooth RAM.

9.1 Option One: Using PC as Host

9.1.1 Test Points and Setup

To connect the prototype board to your PC, ensure that on the hardware, the four UART wires are available as test points. Using an external level shifter, you can use these signals (plus ground) to communicate with a PC. For this option, the RxD, TxD, CTS, and RTS signals would need to be level shifted to correspond to RS-232 voltage settings. For more information, including schematics from the level shifter used on the development kit, refer to the development kit schematics found in the user guides for the 71000 or 72000 Bluetooth Development Kits from Motorola. After attaching these five signals to a 9-pin standard female connector, a standard serial cable can be used to connect to a computer communication port.

Before attempting to download any software to the Bluetooth device, ensure the proper voltages and frequencies are provided to the Bluetooth device. For more information on which voltages and/or frequencies need to be provided, see Section 5, “Voltage Requirements”.

9.1.2 Hardware Test

After setting up the correct voltage and frequencies, the Bluetooth hardware should be tested for functionality. More information including an overview of what kind of equipment and test cases are required to ensure the functionality of the Bluetooth device can be found in *Production Test, Bluetooth Platform Solution from Motorola Application Note* (document number 94001481100).

9.1.3 Software Download

Using Motorola's software tools on the PC, you can configure and download code into the internal RAM or onto an EEPROM if your Bluetooth prototype has one on board. This is all done using the Configuration Manager software. To generate the files to download, refer to the *Configuration Manager User's Guide* (document number 94001480200), which includes screen shots and step-by-step instructions.

After the software has been successfully downloaded to the Bluetooth platform, you can test your Bluetooth hardware using the provided software tools. For more information on how to send and receive HCI commands as well as files to other Bluetooth devices, see *HCI Terminal User's Guide* (document number 94001480100) and *DemoBench User's Guide* (document number 94001480300).

9.2 Option Two: Using Host IC

9.2.1 Test Points and Setup

To use the host IC to test the Bluetooth device, the host must be connecting to the Bluetooth device using the four UART signals. More pins, including GPIOs and PCM port, might be interfaced as well to the host.

Before attempting to download any software to the Bluetooth device, ensure the proper voltages and frequencies are provided to the Bluetooth device. For more information on which voltages and/or frequencies need to be provided, see Section 5, "Voltage Requirements".

9.2.2 Software Download

To download the software into the Bluetooth RAM, non-volatile memory is needed, which is typically connected to the host. This option requires that you understand the file system needed to download software to the Bluetooth RAM. For more information, refer to the following documents:

- *Motorola Bluetooth File System, Overview Application Note* (document number 94001481001)
- *Motorola Bluetooth File System, Host Based One File Application Note* (document number 94001481004)
- *Motorola Bluetooth File System, Host Based General Application Note* (document number 94001481003)
- *Motorola Bluetooth File System, Embedded File System Application Note* (document number 94001481002)

The easiest file system to implement is the "one-file file system". The necessary software files can be configured and saved using the Configuration Manager Software. For more information, refer to the *Configuration Manager User's Guide* (document number 94001480200).

9.2.3 Hardware Test

After setting up the correct voltage and frequencies, the Bluetooth hardware should be tested for functionality. More information including an overview of what kind of equipment and test cases are required to ensure the functionality of the Bluetooth device can be found in *Production Test, Bluetooth Platform Solution from Motorola Application Note* (document number 94001481100).

10 BQB Offering from Motorola

For information on the Bluetooth qualification offering from Motorola, see *Motorola Bluetooth Solutions, Bluetooth Qualification Application Note* (document number AN2386/D)

11 Production and Test

This section briefly describes a number of test philosophies, each representing a different level of testing. These range from minimal testing to the most comprehensive solution. For more detailed information, see *Production Test, Bluetooth Platform Solution from Motorola Application Note* (document number 94001481100).

11.1 No Test

When using a pre-tested and pre-programmed daughter board in the product, a measurement of power consumption and a connect to another reference board may solve the testing of the Bluetooth device. The BD address and the crystal trim of the installed daughter board are programmed by the manufacturer of the daughter board.

11.2 Minimum Testing

When producing equipment that includes Bluetooth chips, there will always be a minimum of parameters that need to be adjusted/programmed in the equipment produced. The simplest way to test if the radio works, is to make a connection to another reference Bluetooth device.

11.3 Host Connect

When the production test is of a more complex nature, special test software needs to be placed in the RAM of the Bluetooth IC to handle the test requirements. The Bluetooth IC will search for a host connect command at the serial port at 9600 baud. The connected host, for example a PC, should then simulate the file system that the Bluetooth IC needs. The connected host then delivers the RAM portion of the software needed to run the various software on the Bluetooth IC.

11.4 JTAG

In some cases, the production test needs to download some RAM-based test software, but the host connect version is too slow or the serial connection on the Bluetooth IC is used for something else and, therefore, cannot be used by the test system. In such cases, the JTAG connector of the Bluetooth IC can be used as testing interface.

It is possible to make a complete RF test and boundary-scan test using standard JTAG and RF equipment.

12 Bluetooth Upper Stack

This section describes the Bluetooth upper stack.

12.1 Embedded Upper Stack Development

The Motorola Bluetooth embedded core stack is written to be hardware/CPU and operating system independently. However, a few modules need to be modified so that the stack can interface correctly to the hardware and operating system that has been chosen by the user. All dependencies are put into a few files that surround the core stack and those must be modified by the person doing the port to the new platform. These interfaces are simple and small and require only a minimum of effort to adapt to a new platform. Basically, the stack consists of the following more or less independent modules as shown in Figure 4.

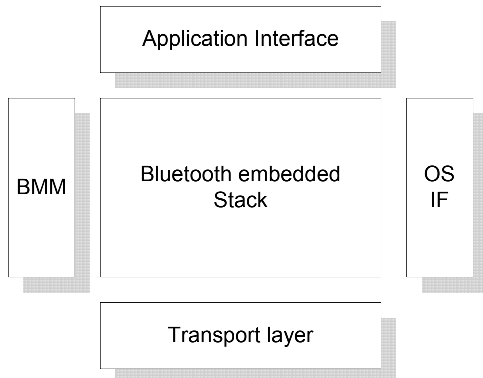


Figure 4. Bluetooth Embedded Stack Modules

For more information on each of the modules, see *Motorola Bluetooth Embedded Protocol Stack Product Brief (document number BTEPSPB/D)*.

12.2 Supported Profiles

The current stack supports the following profiles, but Motorola is constantly developing new profiles, working closely with the Bluetooth SIG.

- Generic Access Profile (GAP)
- Service Discovery Application Profile (SDAP)
- Serial Port Profile (SPP)
- Headset Profile (Headset and Audio Gateway)
- Dial-Up Networking Profile (DUN)
- Fax Profile
- Cordless Telephony Profile
- Intercom Profile
- File Transfer Profile (FTP)
- Object Push Profile (OPP)
- Synchronization Profile
- Personal Area Networking Profile (PAN)

For detailed instructions on how to port the protocol stack, see *Motorola Bluetooth Embedded Protocol Stack SDK* (document number 94001150000).

13 Development, Test, and Evaluation Tools

The Motorola Bluetooth development kits include a number of software tools:

- Configuration Manager
- RadioTest
- DemoBench
- Bluetooth HCI Terminal
- UART Download Tool
- UART/SSI Configuration
- MMI Headset
- Audio Gateway

The following sections briefly describe each of the software tools.

13.1 Configuration Manager

The configuration manager is one of the program tools accompanying the Bluetooth Development Kit from Motorola. The configuration manager is accessible from the Development Kit CD. Using the configuration manager, one can configure and create the files needed to download to the Motorola 72000 at boot time. For more information, see *Configuration Manager User's Guide* (document number 94001480200).

13.2 RadioTest

The RadioTest application allows you to test all aspects of your Bluetooth hardware. This application lets you control your hardware so as to carry out any test required for development purposes and when preparing for production. Testing is both fast and simple; you can do all your testing with one and the same program. In addition, you can create your own test system as desired without losing any of the benefits of the RadioTest application. Finally, the application allows for simultaneous testing of several units using the same equipment. The RadioTest application is exclusively based on RAM. For more information, see *RadioTest User's Guide* (document number 94001480800).

13.3 DemoBench

The DemoBench is one of the program tools included with the Bluetooth Development Kit from Motorola. The DemoBench, which is accessible from the Development Kit CD, offers the following main functions:

- Monitoring of link and packet statistics
- Audio
- Chat
- File transfer

The DemoBench includes a number of monitors: HCI Command/Event monitor, HCI Data Statistics monitor, Current Links monitor, Packet Error monitor, and Packet Statistics monitor.

For more information, see *DemoBench User's Guide* (document number 94001480300).

13.4 HCI Terminal

With the Bluetooth HCI Terminal you can interact with your Bluetooth hardware. The interface is similar to that of an AT terminal application when communicating with a modem.

The Bluetooth HCI Terminal makes it easy to send HCI commands from a computer to a Bluetooth device. Likewise, it is easy to receive HCI responses from a Bluetooth device. Consequently, you can get hands-on experience with the HCI. For more information, see *HCI Terminal User's Guide* (document number 94001480100).

13.5 UART Download

With the UART Download tool, you can download Bluetooth firmware to a blank EEPROM. The type of EEPROM can be changed. If using a Motorola Bluetooth firmware version 1.18 or higher, the feature "Disable SEEPROM automatically" by software may be used.

The BD Address and the X-Tal trim value are parts of the parameters when downloading an application.

13.6 UART/SSI Configuration

The UART/SSI Configuration tool allows you to change the UART/GPIO and SSI settings of a selected configuration.

Parameters to be changed in connection with UART/GPIO settings are baud rate and the GPIO pins on port B (0..13) and port C (0..12) used for HCI commands. The wake-up pin can be set.

SSI settings include: CODEC, Network, Frame Sync, Gain, and Clock.

On finishing the configuration, the settings can be stored

13.7 Headset MMI Configuration

With the Headset MMI Configuration tool, you can customize a headset MMI. The button functions, for example "button pressed" and "button hold", can be modified with up to 6 system states.

LED flash sequences can be defined together with audio indicator sequences.

The configuration setup can be stored.

13.8 Audio Gateway

The Development Kit Audio Gateway application was developed for demonstration purposes, primarily as an easy-to-use means of evaluating the Development Kit embedded headset.

Unlike the headset, the Audio Gateway is (typically) not a self-contained, embedded product, but requires serial interfacing to a PC running a console application (at least for initial setup). This allows a much more open, tester-friendly interface than would be feasible for a stand-alone product. Additionally, status information about inquiries, pairing, existing link, link quality, etc., is made available to the tester.

NOTES

HOW TO REACH US:

USA/EUROPE/LOCATIONS NOT LISTED:

Motorola Literature Distribution;
P.O. Box 5405, Denver, Colorado 80217
1-303-675-2140 or 1-800-441-2447

JAPAN:

Motorola Japan Ltd.; SPS, Technical Information Center,
3-20-1, Minami-Azabu Minato-ku, Tokyo 106-8573 Japan
81-3-3440-3569

ASIA/PACIFIC:

Motorola Semiconductors H.K. Ltd.; Silicon Harbour
Centre, 2 Dai King Street, Tai Po Industrial Estate,
Tai Po, N.T., Hong Kong
852-26668334

TECHNICAL INFORMATION CENTER:

1-800-521-6274

HOME PAGE:

http://www.motorola.com/semiconductors

MOTOROLA

94001481800/D

Preliminary

射 频 和 天 线 设 计 培 训 课 程 推 荐

易迪拓培训(www.edatop.com)由数名来自于研发第一线的资深工程师发起成立，致力并专注于微波、射频、天线设计研发人才的培养；我们于 2006 年整合合并微波 EDA 网(www.mweda.com)，现已发展成为国内最大的微波射频和天线设计人才培养基地，成功推出多套微波射频以及天线设计经典培训课程和 ADS、HFSS 等专业软件使用培训课程，广受客户好评；并先后与人民邮电出版社、电子工业出版社合作出版了多本专业图书，帮助数万名工程师提升了专业技术能力。客户遍布中兴通讯、研通高频、埃威航电、国人通信等多家国内知名公司，以及台湾工业技术研究院、永业科技、全一电子等多家台湾地区企业。

易迪拓培训推荐课程列表： http://www.edatop.com/peixun/tuijian/

射频工程师养成培训课程套装

该套装精选了射频专业基础培训课程、射频仿真设计培训课程和射频电路测量培训课程三个类别共 30 门视频培训课程和 3 本图书教材；旨在引领学员全面学习一个射频工程师需要熟悉、理解和掌握的专业知识和研发设计能力。通过套装的学习，能够让学员完全达到和胜任一个合格的射频工程师的要求…

课程网址：http://www.edatop.com/peixun/rfe/110.html

手机天线设计培训视频课程

该套课程全面讲授了当前手机天线相关设计技术，内容涵盖了早期的外置螺旋手机天线设计，最常用的几种手机内置天线类型——如 monopole 天线、PIFA 天线、Loop 天线和 FICA 天线的设计，以及当前高端智能手机中较常用的金属边框和全金属外壳手机天线的设计；通过该套课程的学习，可以帮助您快速、全面、系统地学习、了解和掌握各种类型的手机天线设计，以及天线及其匹配电路的设计和调试...

课程网址： http://www.edatop.com/peixun/antenna/133.html

WiFi 和蓝牙天线设计培训课程

该套课程是李明洋老师应邀给惠普 (HP)公司工程师讲授的 3 天员工内训课程录像，课程内容是李明洋老师十多年工作经验积累和总结，主要讲解了 WiFi 天线设计、HFSS 天线设计软件的使用，匹配电路设计调试、矢量网络分析仪的使用操作、WiFi 射频电路和 PCB Layout 知识，以及 EMC 问题的分析解决思路等内容。对于正在从事射频设计和天线设计领域工作的您，绝对值得拥有和学习！…

课程网址：http://www.edatop.com/peixun/antenna/134.html

CST 学习培训课程套装

该培训套装由易迪拓培训联合微波 EDA 网共同推出，是最全面、系统、专业的 CST 微波工作室培训课程套装，所有课程都由经验丰富的专家授课，视频教学，可以帮助您从零开始，全面系统地学习 CST 微波工作的各项功能及其在微波射频、天线设计等领域的设计应用。且购买该套装，还可超值赠送 3 个月免费学习答疑…

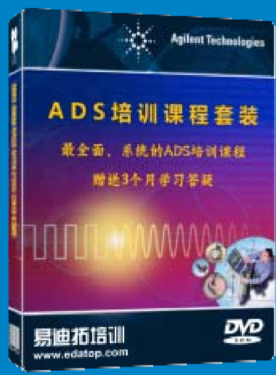
课程网址：http://www.edatop.com/peixun/cst/24.html

HFSS 学习培训课程套装

该套课程套装包含了本站全部 HFSS 培训课程，是迄今国内最全面、最专业的 HFSS 培训教程套装，可以帮助您从零开始，全面深入学习 HFSS 的各项功能和在多个方面的工程应用。购买套装，更可超值赠送 3 个月免费学习答疑，随时解答您学习过程中遇到的棘手问题，让您的 HFSS 学习更加轻松顺畅…

课程网址：http://www.edatop.com/peixun/hfss/11.html

ADS 学习培训课程套装



该套装是迄今国内最全面、最权威的 ADS 培训教程，共包含 10 门 ADS 学习培训课程。课程是由具有多年 ADS 使用经验的微波射频与通信系统设计领域资深专家讲解，并多结合设计实例，由浅入深、详细而又全面地讲解了 ADS 在微波射频电路设计、通信系统设计和电磁仿真设计方面的内容。能让您在最短的时间内学会使用 ADS，迅速提升个人技术能力，把 ADS 真正应用到实际研发工作中去，成为 ADS 设计专家...

课程网址： http://www.edatop.com/peixun/ads/13.html

我们的课程优势：

※ 成立于 2004 年，10 多年丰富的行业经验，

※ 一直致力并专注于微波射频和天线设计工程师的培养，更了解该行业对人才的要求

※ 经验丰富的一线资深工程师讲授，结合实际工程案例，直观、实用、易学

联系我们：

※ 易迪拓培训官网：http://www.edatop.com

※ 微波 EDA 网：http://www.mweda.com

※ 官方淘宝店：http://shop36920890.taobao.com