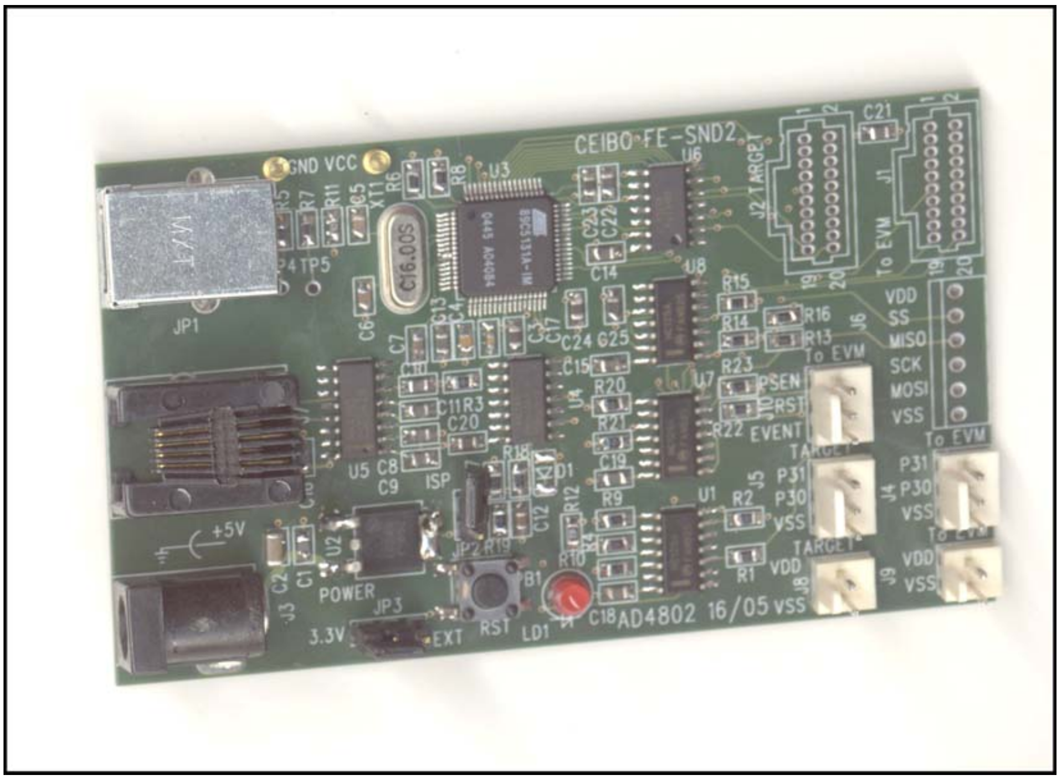
CEIBO FE-51SND2 Development System



Development System for Atmel AT89C51SND2 Microcontrollers

FEATURES

- **Emulates Atmel AT89C51SND2 Microcontrollers**
- **62K Code Memory**
- **Real-Time Emulation**
- **Frequency up to fmax**
- **ISP and X2 Mode Support**
- **Windows Debugger For C, C++ and Assembler**
- **USB or RS-232 Linked to PC**
- **C Compiler and Assembler**

General Description

Ceibo FE-SND2 is a development system that supports ATMEL AT89C51SND2C microcontrollers with 6/12 clocks/cycle at any frequency allowed by the devices.

It is serially linked to a PC or compatible systems by means of USB or RS-232 and can emulate the microcontrollers directly from the target board.

Loading the system with the user software and an embedded monitor program carries out emulation. FE-SND2 locates the monitor in the upper 2K of the code memory space of the ATMEL AT89C51SND2C.

Two working modes are available: real-time and simulator.

In the real-time mode the user software is executed transparently and without interfering with the microcontroller speed. Breakpoints can be added to stop program execution at a specific address.

The ***simulation mode does not implement all the chip options*** and it is intended only for software debugging of the basic 8051 functions. ***Use the system in emulation mode.*** The simulation mode is used to debug the software without any hardware. FE-SND2 may be disconnected while using the simulation mode.

The software includes C and Assembler Source Level Debugger, On-line Assembler and Disassembler, Trace, Conditional Breakpoints and many other features. Also a free C-Compiler and Assembler not limited in code size is supplied with the system. C++ is optional.

The system is supplied with Windows debugger software, RS-232 cable and a power supply.

This chapter describes the parts of the system.

Applications

The main applications of FE-SND2 Development System are:

- Emulation of MP3 microcontrollers
- Development of microprocessor based systems
- Hardware and software debugging purposes

Specifications

System Memory

FE-SND2 provides 64K of code memory. However, only 62K are available for user's programs while emulating because the system comes with an embedded monitor program that uses 2K of the memory space, from E800h to EFFFh.

Breakpoints

Breakpoints allow real-time program execution until an opcode is executed at a specified address.

SoftwareTrace

Trace can be used to display the last executed instructions in real time. Trace is variable in depth and shows backward all the sequential instructions until the last branch instruction is found (LJMP, ACALL, DJNZ, etc.).

Windows Debugger

The FE-SND2 software includes a source level debugger for Assembler and high-level languages C and others with the capability of executing lines of the program while displaying the state of any variable. The debugger uses symbols contained in the absolute file generated by the most commonly used Assemblers and High Level Language Compilers. The CEIBO Windows Debugger runs only under Windows 98 or later (NT, XP, 2000, etc).

Keil μVision2 Debugger

This debugger may also be used to operate the emulator. Some files have to be added to the original Keil's package and those can be downloaded from **www.ceibo.com**.

Supported Microcontrollers

The system supports ATMEL AT89C51SND2 microcontrollers and other derivatives that may be announced in the future. The standard supported package for emulation is BGA.

Frequency

FE-SND2 runs from the clock source supplied by the user hardware. The minimum and maximum frequencies are determined by the emulated chip characteristics, while the emulator is not limited in frequency.

Host Characteristics

PC or compatible systems with 8 MByte of RAM, one COM port or USB, Windows 98 or later.

Input Power

5V, 1.5A power supply supplied.

Mechanical Dimensions

5cm x 9cm.

Items Supplied as Standard

Development system including emulator, Windows software with source level debugger, on-line assembler and disassembler, power supply, C-Compiler and Assembler, user's manual and RS-232 cable.

Hardware Description - FE-SND2 Emulator

The first step required for working with the FE-SND2 board is to be able to identify its different parts and to understand how the electronics function. This will help you to take full advantage of all the FE-SND2 capabilities.

On the left side of the FE-SND2 you will see a phone jack. It is used to serially link the FE-SND1 to your computer COM port and to utilize the software in emulation mode instead of just the simulator. The emulation mode is used to interact with the hardware while the simulator is independent of any hardware connection. The RS-232 cable connections are given separately.

Above the RS-232 connector there is a USB connector to link the emulator to your PC, instead of the COM port.

Below the RS-232 there is Power jack to connect the power supply.

On the bottom side you will see an LED indicating whether or not the power is applied to the system.

U3 is the microcontroller used to control the emulation of the target AT89C51SND2. It is a QFP device programmed with the emulator Firmware. JP2 is the jumper used to enter that microcontroller in ISP mode and update the Firmware in case that future releases will be available. Make sure that the JP2 cap is not placed on the jumper to work properly.

On the right side of the board there are several connectors used to connect the emulator to a target board, thus providing optional sharing resources: RS-232, I2C, SPI.

JP3 is a jumper that selects the power source. FE-SND2 can use the target power supply or the Ceibo power supply, according to the jumper setting. Ceibo Power supply is 5VDC, but a voltage converter drops it to 3.3V for the emulator.

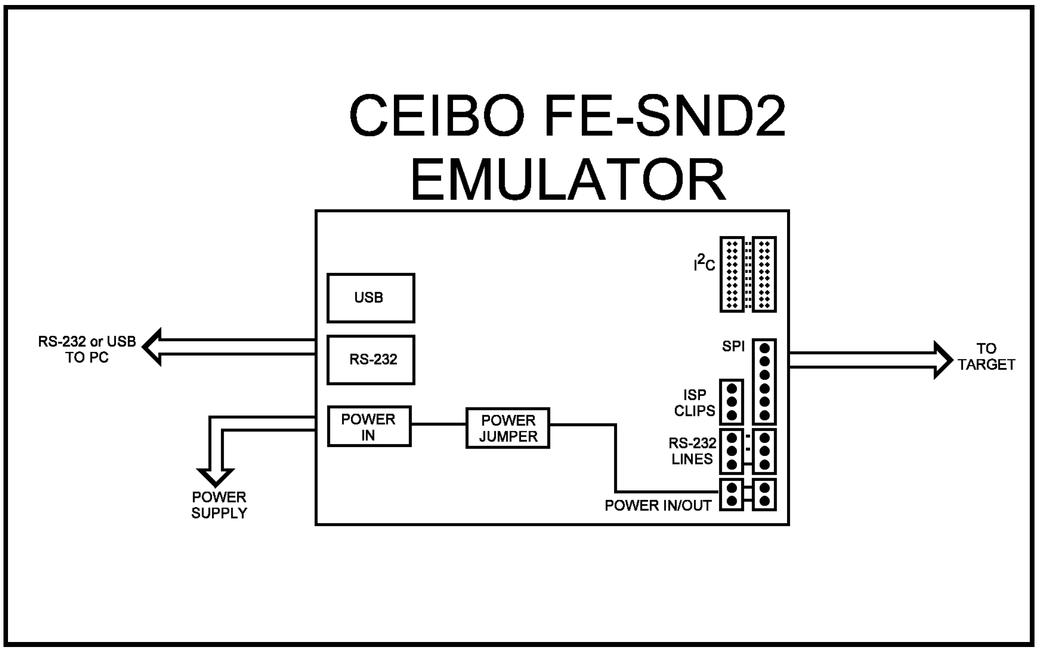


FIGURE: ***FE-SND2 Emulator Connections***

Connecting FE-SND2 to a Target

The user target needs to be prepared with a few testpoints or soldering pads that may be not used in production units. Those will not add components to your final product and will have the advantage to provide easy debugging at any time.

The Ceibo recommended soldering pads on the target are:

1. RST and ISP (PSEN) - these will be needed to enter the ISP mode and update the Flash memory of the BGA chip.

2. RS-232 lines - these are TxD, RxD and GND. The emulator will share these lines and not reserve them. That means, the user application will not be affected by these line sharing. If your target connects on the board these lines to RS-232 line drivers (MAX232, etc.), then you can use Ceibo MP-51RD2 or similar board to reconvert the signals to the logic levels needed by the emulator.

Having these testpoints available make the connections as simple as shown in the following figure.

FE-SND2 emulator has two RS-232 connectors for the target, so in case your application needs your target connector, there are chained in the emulator. In such a case, you connect the target connector to one of the two emulator RS-232 connectors and your application to the additional connector on the emulator.

The same chaining is done with the power connectors if you want to apply to the emulator the target power source.

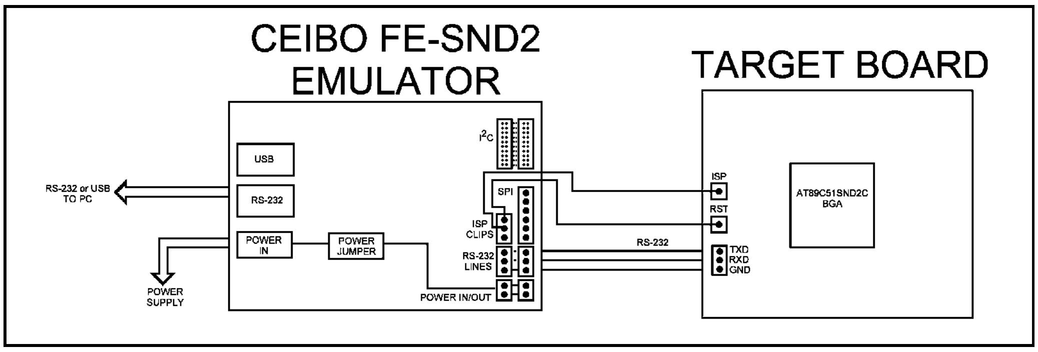


FIGURE: ***FE-SND2 Emulator Connections***

The emulator has also other connecting capabilities that avoid any RS-232 line sharing. In that case, you may connect the emulator to the AT89C51SND2 BGA chip by means of SPI or I2C. These possibilities are described in the connections to Atmel Demo Board.

Connecting FE-SND2 to Atmel Demo Board

Atmel AT89RFD-08 Demo Board can be connected to Ceibo FE-SND1 in different ways. The first connection is sharing RS-232 lines. In this case, the 3-pin connector on the emulator is directly connected to AT89RFD-08. The emulator has a second similar connector to chain the signals in case your application connects them to the demo board; in that case, you need to connect the signals to the emulator and you take them back from the additional emulator connector to your application.

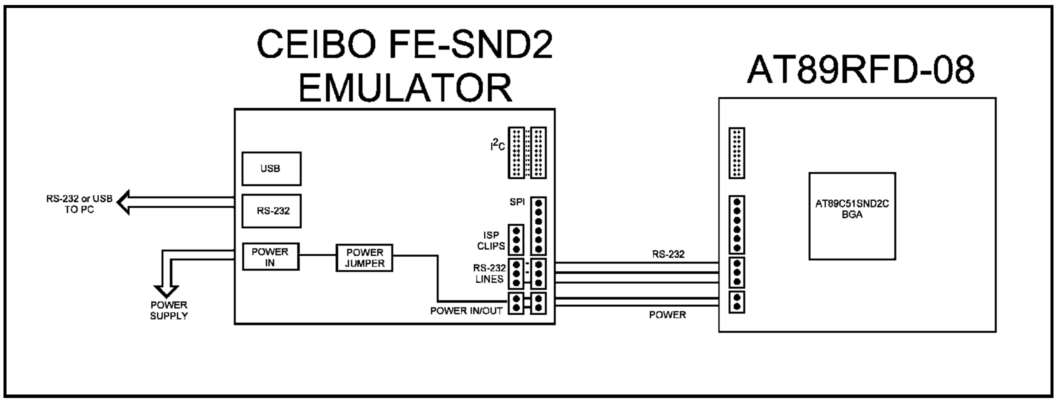


FIGURE: ***FE-SND2 and AT89RFD-08 RS-232 Connections***

If this connection is not suitable to your application, then you can connect the 20-pin connector on the board to the emulator. These lines include I2C and they provide an alternate way to connect to the BGA chip. The emulator will gain control on these lines and disconnect them from your application, which may chain the additional lines of that connector from the additional 20-pin connector on the emulator.

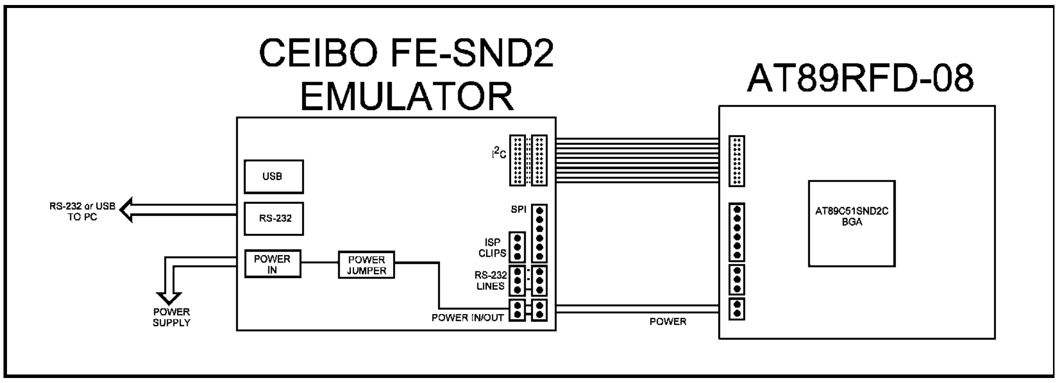


FIGURE: ***FE-SND2 and AT89RFD-08 I2C Connections***

Another possibility is to connect the emulator to the target using SPI. Connections are given in the following figure.

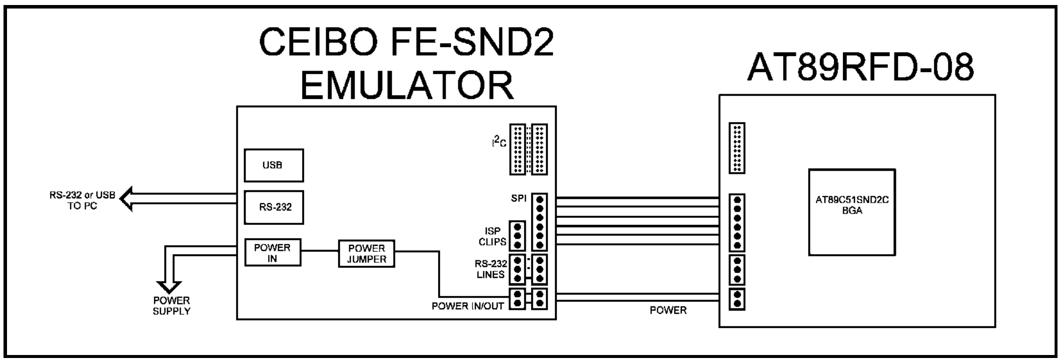


FIGURE: ***FE-SND2 and AT89RFD-08 SPI Connections***

Emulation Restrictions

The following restrictions are valid for FE-SND2:

1. FE-SND2 Monitor Program shares 2 KByte of the 64K memory code space. Therefore, only 62K are available for user's programs while emulating because the system comes with an embedded monitor program that uses these 2K of the memory space, from E800h to EFFFh.

2. Code memory cannot be mapped external and always belongs to the emulator system.

3. The program also uses 4 Bytes of the internal stack memory.

4. The stack pointer may not be defined below address 7.

5. The first instruction (address 0000h) must be 3-bytes long. For example: use LJMP and not SJMP or AJMP as the first instruction.

6. The UART is shared with the system and interrupts should not be disabled. Also the related timer to the serial port must not be stopped. The Halt Mechanism in the Options Menu offers other solutions in case you need these resources in your application.

7. TxD line can be used as Output Port or UART line. It cannot be used as Input Port.

8. RxD line can be used only as UART line. It cannot be used as generic I/O Port.

9. Breakpoints cannot be set to the Reset Vector (address 0000h-0002h), Serial Interrupt (address 0023h-0025h) or to the 4K of the reserved memory space (address F000h-FFFFh). Any other location is allowed, included inside interrupt service routines.