Configuring EZ-USB Without a USB Connection

Introduction

Certain USB peripherals must become operational before they connect to USB. For example, an electronic camera might use the integrated 8051 microprocessor in the EZ-USB chip for camera operation, and later connect to USB to download pictures. Another example would be a device that already incorporates a standard serial port, which runs on its own before connecting to USB.

The EZ-USB chip family is "soft," receiving device configuration and program code via the USB connection. Thus a natural question arises; "How do I load code into EZ-USB if the USB connection is not initially there?"

The EZ-USB family provides two alternate methods (other than USB download) for loading code into internal program RAM and commencing 8051 operation:

1. Loading all code from EEPROM.
2. Loading code via a serial port.

The first method is straightforward, and described in detail in the EZ-USB Technical Reference Manual. Briefly, an EEPROM is attached to the EZ-USB I^2C bus, and the 8051 program code (which includes all USB configuration information) is automatically loaded into RAM and executed at power-on.

The second method is less obvious, and is the subject of this note.

Details

Figure 1 is a block diagram of an AN2121 system that is capable of being loaded with 8051 code from two sources:

- A standard serial port
- USB

This implements a "dual-boot" system, whereby the AN2121 can be loaded either from the USB or serial ports, at any time.

Implementation Notes

1. Since the system supports loading code via the serial port without being connected to USB, it must be self-powered with a local 3.3-volt power supply. This supply provides power to the AN2121, the EEPROM, and the RS-232 level shifter.
2. The MAX-232 or equivalent chip shifts the logic-level AN2121 inputs and outputs to RS-232 signaling levels. It must operate with a 3.3-volt supply.
3. The EEPROM contains boot code that is loaded into AN2121 RAM at power-up. The EEPROM shown (24LC02) contains 256 bytes. Larger EEPROMS may be used depending on required boot code size. The AN2121 performs a "B2-load" from this EEPROM, which loads the EEPROM data into AN2121 RAM and then executes it from RAM. (Refer to the EZ-USB TRM Section 5.8 for details.)
4. The 8051 boot code (loaded from the EEPROM at power-on) continuously checks for data on both the USB and serial ports. A simple way to detect USB attachment is to

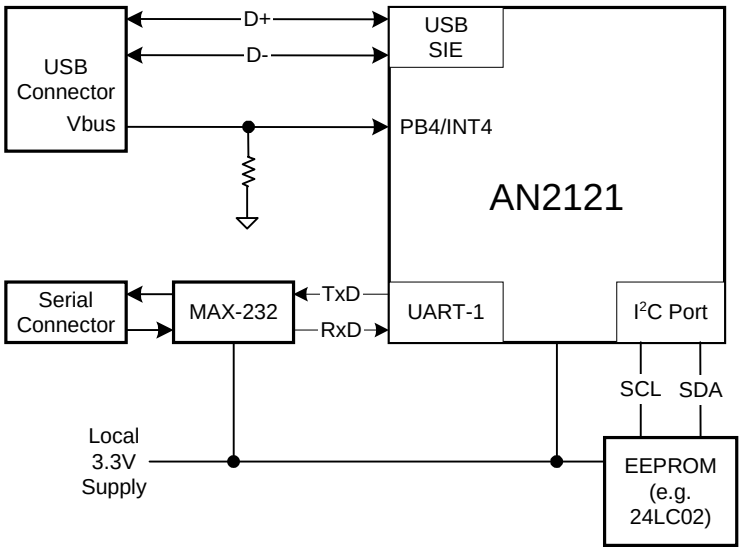

Figure 1. Dual-boot AN2121 Configuration

connect the Vbus power line to an AN2121 IO port, which is pulled down with a resistor (as shown). The PB4 pin can also be used as INT4, which allows the 8051 to detect the USB connection as an interrupt. Even though the AN2121 is powered from a 3.3-volt supply, its IO pins are tolerant to 5-volt signals, such as the Vbus power line. Therefore the Vbus line may be safely connected to PB4.

5. Once the boot code detects traffic on one of the ports, it downloads target code into RAM, and then executes it.
6. When loading code from the USB port, the boot loader can be simplified by utilizing the “Anchor Download” command that is built into the AN2121 core. The 8051 takes no part in an Anchor Download—it is handled entirely by the AN2121 core.
7. Although a convenient protocol is built into the AN2121 for an Anchor-Download of 8051 code via the USB port, the user will need to design a protocol for loading code via the serial port. One simple method would be to use the industry standard “Intel hex” format. After completion of the code download, the boot loader would transfer control to the code in RAM.