C8051F36x-DK

C8051F36x Development Kit User's Guide

1. Relevant Devices

The C8051F36x Development Kit is intended as a development platform for the microcontrollers in the C8051F36x MCU family. The members of this MCU family are C8051F360, C8051F361, C8051F362, C8051F363, C8051F364, C8051F365, C8051F366, C8051F367, C8051F368, and C8051F369.

Notes:

- The target board included in this kit is provided with a pre-soldered **C8051F360** MCU (LQFP48 package).
- Code developed on the C8051F360 can be easily ported to the other members of this MCU family.
- Refer to the C8051F36x data sheet for the differences between the members of this MCU family.

2. Kit Contents

The C8051F36x Development Kit contains the following items:

- C8051F360 Target Board
- C8051Fxxx Development Kit Quick-Start Guide
- Silicon Laboratories IDE and Product Information CD-ROM. CD content includes:
    - Silicon Laboratories Integrated Development Environment (IDE)
    - Keil 8051 Development Tools (macro assembler, linker, evaluation 'C' compiler)
    - Source code examples and register definition files
    - Documentation
    - C8051F36x Development Kit User's Guide (this document)
- AC/DC Power Adapter
- USB Debug Adapter (USB to Debug Interface)
- USB Cable

3. Getting Started

All software required to develop firmware and communicate with the target microcontroller is included in the CD-ROM. The CD-ROM also includes other useful software.

Below is the software necessary for firmware development and communication with the target microcontroller:

- Silicon Laboratories Integrated Development Environment (IDE)
- Keil 8051 Development Tools (macro assembler, linker, evaluation C compiler)

Other useful software that is provided in the CD-ROM includes the following:

- Configuration Wizard 2
- Keil uVision Drivers
- CP210x USB to UART Virtual COM Port (VCP) Drivers

3.1. Software Installation

The included CD-ROM contains the Silicon Laboratories Integrated Development Environment (IDE), Keil software 8051 tools and additional documentation. Insert the CD-ROM into your PC's CD-ROM drive. An installer will automatically launch, allowing you to install the IDE software or read documentation by clicking buttons on the Installation Panel. If the installer does not automatically start when you insert the CD-ROM, run *autorun.exe* found in the root directory of the CD-ROM. Refer to the *ReleaseNotes.txt* file on the CD-ROM for the latest information regarding known problems and restrictions. After installing the software, see the following sections for information regarding the software and running one of the demo applications.

3.2. CP210x USB to UART VCP Driver Installation

The C8051F360 target board includes a Silicon Laboratories CP2102 USB-to-UART Bridge Controller. Device drivers for the CP2102 need to be installed before PC software such as HyperTerminal can communicate with the target board over the USB connection. If the "Install CP210x Drivers" option was selected during installation, this will launch a driver "unpacker" utility.

1. Follow the steps to copy the driver files to the desired location. The default directory is *C:\SiLabs\MCU\CP210x*.
2. The final window will give an option to install the driver on the target system. Select the "Launch the CP210x VCP Driver Installer" option if you are ready to install the driver.
3. If selected, the driver installer will now launch, providing an option to specify the driver installation location. After pressing the "Install" button, the installer will search your system for copies of previously installed CP210x Virtual COM Port drivers. It will let you know when your system is up to date. The driver files included in this installation have been certified by Microsoft.
4. If the "Launch the CP210x VCP Driver Installer" option was not selected in step 3, the installer can be found in the location specified in step 2, by default *C:\SiLabs\MCU\CP210x\Windows_2K_XP_S2K3_Vista*. At this location, run *CP210xVCPInstaller.exe*.
5. To complete the installation process, connect the included USB cable between the host computer and the USB connector (P4) on the C8051F360 target board. Windows will automatically finish the driver installation. Information windows will pop up from the taskbar to show the installation progress.
6. If needed, the driver files can be uninstalled by selecting "Silicon Laboratories CP210x USB to UART Bridge (Driver Removal)" option in the "Add or Remove Programs" window.

4. Software Overview

4.1. Silicon Laboratories IDE

The Silicon Laboratories IDE integrates a source-code editor, a source-level debugger, and an in-system Flash programmer. See Section 6., "Using the Keil Software 8051 Tools with the Silicon Laboratories IDE," on page 6 for detailed information on how to use the IDE. The Keil Evaluation Toolset includes a compiler, linker, and assembler and easily integrates into the IDE. The use of third-party compilers and assemblers is also supported.

4.1.1. IDE System Requirements

The Silicon Laboratories IDE requirements:

- Pentium-class host PC running Microsoft Windows 2000 or newer.
- One available USB port.
- 64 MB RAM and 40 MB free HD space recommended.

4.1.2. Third Party Toolsets

The Silicon Laboratories IDE has native support for many 8051 compilers. The full list of natively supported tools is as follows:

- Keil
- IAR
- Raisonance
- Tasking
- Hi-Tech
- SDCC

The demo applications for the C8051F360 target board are written to work with the Keil and SDCC toolsets.

4.2. Keil Evaluation Toolset

4.2.1. Keil Assembler and Linker

The assembler and linker that are part of the Keil Demonstration Toolset are the same versions that are found in the full Keil Toolset. The complete assembler and linker reference manual can be found on-line under the **Help** menu in the IDE or in the "*SiLabs\MCU\hlp*" directory (A51.chm).

4.2.2. Keil Evaluation C51 C Compiler

The evaluation version of the C51 compiler is the same as the full version with the following limitation: (1) Maximum 4 kB code generation. When installed from the CD-ROM, the C51 compiler is initially limited to a code size of 2 kB, and programs start at code address 0x0800. Refer to the Application Note "AN104: Integrating Keil Tools into the Silicon Labs IDE" for instructions to change the limitation to 4 kB, and have the programs start at code address 0x0000.

4.3. Configuration Wizard 2

The Configuration Wizard 2 is a code generation tool for all of the Silicon Laboratories devices. Code is generated through the use of dialog boxes for each of the device's peripherals.

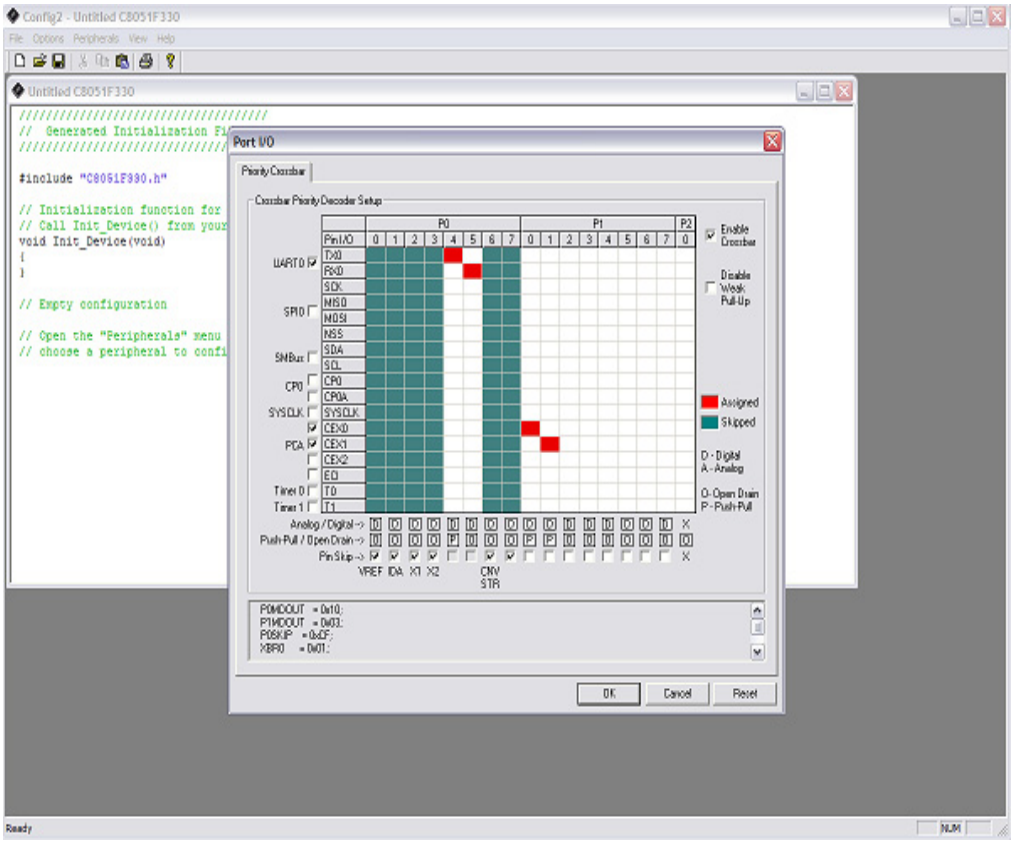

Figure 1. Configuration Wizard 2 Utility

The Configuration Wizard 2 utility helps accelerate development by automatically generating initialization source code to configure and enable the on-chip resources needed by most design projects. In just a few steps, the wizard creates complete startup code for a specific Silicon Laboratories MCU. The program is configurable to provide the output in C or assembly. For more information, refer to the Configuration Wizard 2 help available under the **Help** menu in Config Wizard 2.

4.4. Keil uVision2 and uVision3 Silicon Laboratories Drivers

As an alternative to the Silicon Laboratories IDE, the uVision debug driver allows the Keil uVision IDE to communicate with Silicon Laboratories on-chip debug logic. In-system Flash memory programming integrated into the driver allows for rapidly updating target code. The uVision IDE can be used to start and stop program execution, set breakpoints, check variables, inspect and modify memory contents, and single-step through programs running on the actual target hardware.

5. Hardware Setup Using a USB Debug Adapter

The target board is connected to a PC running the Silicon Laboratories IDE via the USB Debug Adapter as shown in Figure 2.

1. Connect the USB Debug Adapter to the DEBUG connector on the target board with the 10-pin ribbon cable.
2. Connect one end of the USB cable to the USB connector on the USB Debug Adapter.
3. Connect the other end of the USB cable to a USB Port on the PC.
4. Connect the AC/DC Power Adapter to power jack P2 on the target board.

Notes:

- Use the Reset button in the IDE to reset the target when connected using a USB Debug Adapter.
- Remove power from the target board before removing the ribbon cable from the target board. Connecting or disconnecting the cable when the devices have power can damage the device and/or the USB Debug Adapter.

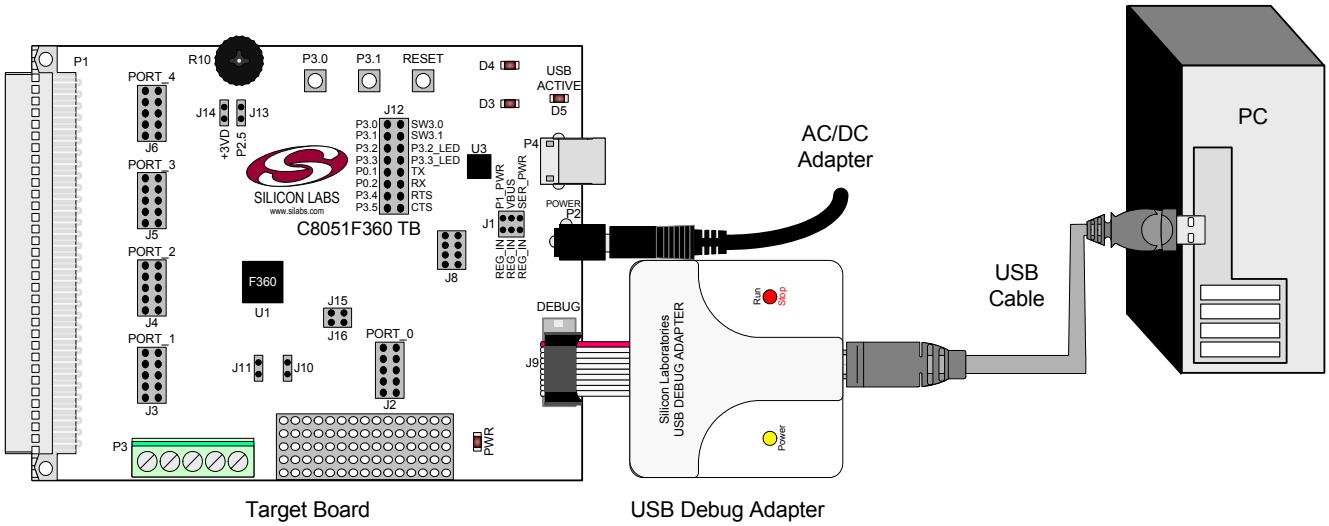

Figure 2. Hardware Setup using a USB Debug Adapter

6. Using the Keil Software 8051 Tools with the Silicon Laboratories IDE

To perform source-level debugging with the IDE, you must configure the Keil 8051 tools to generate an absolute object file in the OMF-51 format with object extensions and debug records enabled. You may build the OMF-51 absolute object file by calling the Keil 8051 tools at the command line (e.g., batch file or make file) or by using the project manager built into the IDE. The default configuration when using the Silicon Laboratories IDE project manager enables object extension and debug record generation. Refer to Application Note "AN104: Integrating Keil 8051 Tools into the Silicon Labs IDE" in the "*SiLabs\MCU\Documentation\ApplicationNotes*" directory for additional information on using the Keil 8051 tools with the Silicon Laboratories IDE.

To build an absolute object file using the Silicon Laboratories IDE project manager, you must first create a project. A project consists of a set of files, IDE configuration, debug views, and a target build configuration (list of files and tool configurations used as input to the assembler, compiler, and linker when building an output object file).

The following sections illustrate the steps necessary to manually create a project with one or more source files, build a program, and download the program to the target in preparation for debugging. (The IDE will automatically create a single-file project using the currently open and active source file if you select **Build/Make Project** before a project is defined.)

6.1. Creating a New Project

1. Select **Project→New Project** to open a new project and reset all configuration settings to default.
2. Select **File→New File** to open an editor window. Create your source file(s) and save the file(s) with a recognized extension, such as .c, .h, or .asm, to enable color syntax highlighting.
3. Right-click on "New Project" in the **Project Window**. Select **Add files to project**. Select files in the file browser and click Open. Continue adding files until all project files have been added.
4. For each of the files in the **Project Window** that you want assembled, compiled and linked into the target build, right-click on the file name and select **Add file to build**. Each file will be assembled or compiled as appropriate (based on file extension) and linked into the build of the absolute object file.

   Note: If a project contains a large number of files, the "Group" feature of the IDE can be used to organize. Right-click on "New Project" in the **Project Window**. Select **Add Groups to project**. Add pre-defined groups or add customized groups. Right-click on the group name and choose **Add file to group**. Select files to be added. Continue adding files until all project files have been added.

6.2. Building and Downloading the Program for Debugging

1. Once all source files have been added to the target build, build the project by clicking on the **Build/Make Project** button in the toolbar or selecting **Project→Build/Make Project** from the menu.

   Note: After the project has been built the first time, the **Build/Make Project** command will only build the files that have been changed since the previous build. To rebuild all files and project dependencies, click on the **Rebuild All** button in the toolbar or select **Project→Rebuild All** from the menu.
2. Before connecting to the target device, several connection options may need to be set. Open the **Connection Options** window by selecting **Options→Connection Options . . .** in the IDE menu. First, select the appropriate adapter in the "Serial Adapter" section. Next, the correct "Debug Interface" must be selected. C8051F36x family devices use the Silicon Labs 2-wire (C2) debug interface. Once all the selections are made, click the OK button to close the window.
3. Click the **Connect** button in the toolbar or select **Debug→Connect** from the menu to connect to the device.
4. Download the project to the target by clicking the **Download Code** button in the toolbar.

   Note: To enable automatic downloading if the program build is successful, select **Enable automatic connect/ download after build** in the **Project→Target Build Configuration** dialog box. If errors occur during the build process, the IDE will not attempt the download.
5. Save the project when finished with the debug session to preserve the current target build configuration, editor settings, and the location of all open debug views. To save the project, select **Project→Save Project As . . .** from the menu. Create a new name for the project and click on **Save**.

7. Example Source Code

Example source code and register definition files are provided in the "*SiLabs\MCU\Examples\C8051F36x*" directory during IDE installation. These files may be used as a template for code development. Example applications include a blinking LED example which configures the green LED on the target board to blink at a fixed rate.

7.1. Register Definition Files

Register definition files *C8051F360.inc and C8051F360.h* define all SFR registers and bit-addressable control/status bits. They are installed into the "*SiLabs\MCU\Examples\C8051F36x*" directory during IDE installation. The register and bit names are identical to those used in the C8051F36x data sheet. Both register definition files are also installed in the default search path used by the Keil Software 8051 tools. Therefore, when using the Keil 8051 tools included with the development kit (A51, C51), it is not necessary to copy a register definition file to each project's file directory.

7.2. Blinking LED Example

The example source files, *blink.asm* and *blinky.c,* show examples of several basic C8051F36x functions. These include: disabling the watchdog timer (WDT), configuring the Port I/O crossbar, configuring a timer for an interrupt routine, initializing the system clock, and configuring a GPIO port. When compiled/assembled and linked, this program flashes the green LED on the C8051F360 target board about five times a second using the interrupt handler with a timer.

8. Target Board

The C8051F36x Development Kit includes a target board with a C8051F360 device pre-installed for evaluation and preliminary software development. Numerous input/output (I/O) connections are provided to facilitate prototyping using the target board. Refer to Figure 3 for the locations of the various I/O connectors.

P1	96-pin female connector
P2	Power connector (Accepts input from 7 to 15 VDC unregulated power adapter.)
P3	Analog I/O terminal block
P4	USB connector (for CP2102 USB-to-UART bridge)
J1	Power supply header (Selects power from the USB Debug Adapter, P1 Power Adapter, or USB power if P4 is connected. Only one power option should be selected at one time.)
J2	Port 0 header
J3	Port 1 header
J4	Port 2 header
J5	Port 3 header
J6	Port 4 header
J7	Connects the +3 V supply net to the VDD supply net
J8	Supply signal header
J9	Debug connector for debug adapter interface
J10, J11	External crystal port pin enable connectors
J12	Port I/O jumper configuration block
J13	Jumper connection for potentiometer to pin 2.5
J14	Jumper connection for potentiometer source to +3 V
J15	Jumper connection for pin 0.3 to capacitors (used when VREF is internally generated)
J16	Jumper connection for pin 0.4 to resistor/capacitor (used to convert IDAC output to a voltage)
J18	Connects the +3 V supply net to the AV+ supply net

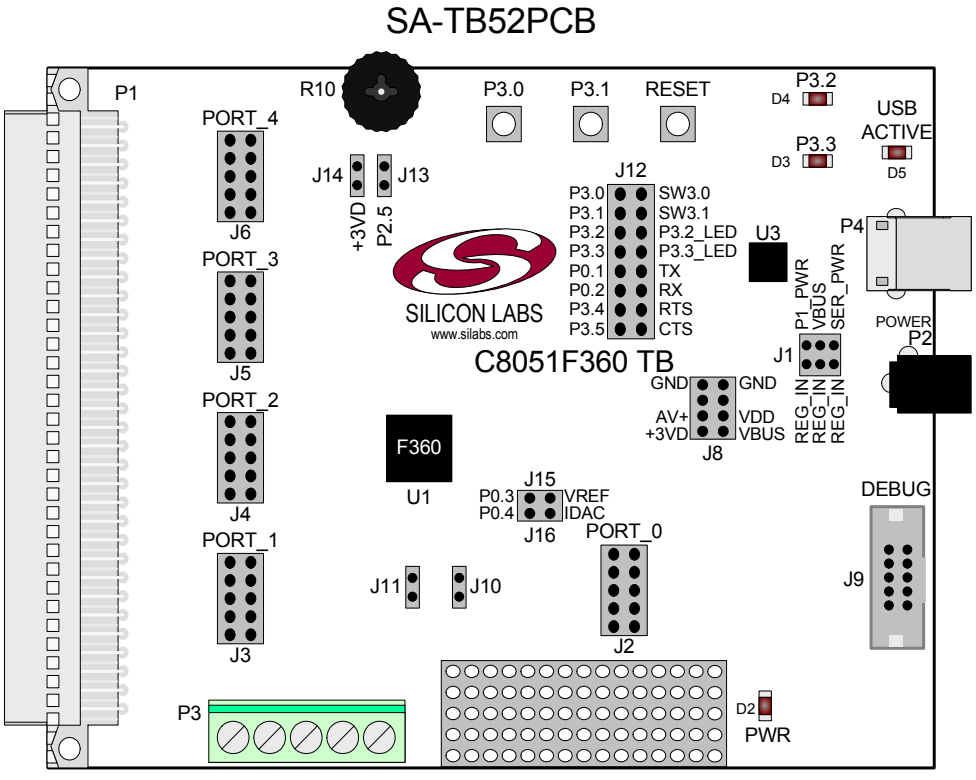

Figure 3. C8051F360 Target Board

8.1. System Clock Sources

The C8051F360 device installed on the target board features a calibrated programmable internal oscillator which is enabled as the system clock source on reset. After reset, the internal oscillator operates at a frequency of 3.0625 MHz (±1.5%) by default but may be configured by software to operate at other frequencies. Therefore, in many applications an external oscillator is not required. However, if you wish to operate the C8051F360 device at a frequency not available with the internal oscillator, an external crystal may be used. Refer to the C8051F36x data sheet for more information on configuring the system clock source.

The target board is designed to facilitate the installation of an external crystal. Remove shorting blocks at headers J10 and J11 and install the crystal at the pads marked Y1. Install a 10 MΩ resistor at R1 and install capacitors at C20 and C19 using values appropriate for the crystal you select. Refer to the C8051F36x data sheet for more information on the use of external oscillators.

8.2. Switches and LEDs

Three switches are provided on the target board. Switch RESET is connected to the RESET pin of the C8051F360. Pressing RESET puts the device into its hardware-reset state. Switches P3.0 and P3.1 are connected to the C8051F360's general purpose I/O (GPIO) pins through headers. Pressing P3.0 or P3.1 generates a logic low signal on the port pin. Remove the shorting blocks from the J12 header to disconnect Switch P3.0 and Switch P3.1 from the port pins. See Table 1 for the port pins and headers corresponding to each switch.

Four LEDs are also provided on the target board. The red LED labeled PWR is used to indicate a power connection to the target board. The green surface-mount LEDs labeled with port pin names are connected to the C8051F360's GPIO pins through headers. Remove the shorting blocks from the header to disconnect the LEDs from the port pin. The USB ACTIVE red LED indicates when the CP210x USB-to-UART bridge (U3) on the board is receiving power from the USB bus and is properly enumerated (i.e. drivers are installed and a USB cable is connected to P4). See Table 1 for the port pins and headers corresponding to each LED.

Also included on the C8051F360 target board is a 10 KΩ thumb-wheel rotary potentiometer, part number R10. The potentiometer is connected to the C8051F360's P2.5 pin through the J13 header. Remove the shorting block from the header to disconnect the potentiometer from the port pin. See Table 1 for the port pin and header corresponding to the potentiometer.

Table 1. Target Board I/O Descriptions

Description	I/O	Header
SW1	Reset	none
SW2	P3.0	J12[1–2]
SW3	P3.1	J12[3–4]
Green LED	P3.2	J12[5–6]
Green LED	P3.3	J12[7–8]
Red LED	PWR	none
Red LED	USB ACTIVE	none
Potentiometer	P2.5	J13

8.3. PORT I/O Connectors (J2 - J6)

In addition to all port I/O signals being routed to the 96-pin expansion connector, each of the five parallel ports of the C8051F360 has its own 10-pin header connector. Each connector provides a pin for the corresponding port pins 0–7, +3.3 VDC and digital ground. Table 3 defines the pins for the port connectors, where Pn represents P0 through P4. The same pin-out order is used for all of the port connectors.

Table 2. J12–J19 Port Connector Pin Descriptions

Pin #	Description
1	Pn.0
2	Pn.1
3	Pn.2
4	Pn.3
5	Pn.4
6	Pn.5
7	Pn.6
8	Pn.7 (not connected for J6)
9	+3 VD (+3.3 VDC)
10	GND (Ground)

8.4. Target Board DEBUG Interface (J9)

The DEBUG connector (J9) provides access to the DEBUG (C2) pins of the C8051F360. It is used to connect the Serial Adapter or the USB Debug Adapter to the target board for in-circuit debugging and Flash programming. Table 3 shows the DEBUG pin definitions.

Table 3. DEBUG Connector Pin Descriptions

Pin #	Description
1	+3 VD (+3.3 VDC)
2, 3, 9	GND (Ground)
4	C2D
5	/RST (Reset)
6	P4.6
7	C2CK
8	Not Connected
10	USB Power (from USB Debug Adapter)

8.5. USB to Serial Connector (P1)

A USB-to-Serial bridge interface is provided. A USB B-type connector (P1), a Silicon Laboratories CP2102 USB-to-UART Bridge, and related circuits are provided to facilitate the serial connection between a PC and the C8051F360 microcontroller on the target board. The RX, TX, CTS and RTS signals of the UART side of the Bridge (CP2102) may be connected to the microcontroller by installing shorting blocks on J12 as follows:

Table 4. UART Connections (J12)

Connection	Signals
J12[9–10]	P0.1 to TX_MC
J12[11–12]	P0.2 to RX_MC
J12[13–14]	P3.4 to RTS
J12[15–16]	P3.5 to CTS

8.6. Analog I/O (P2)

Several of the C8051F360 target device's port pins are connected to the P3 terminal block. Refer to Table 5 for the P3 terminal block connections.

Table 5. J6 Terminal Block Pin Descriptions

Pin #	Description
1	P2.3/AIN2.3/CP0+
2	P2.4/AIN2.4/CP0–
3	GND (Ground)
4	P0.3/VREF (Voltage Reference)
5	P0.4/IDAC

8.7. Power Connector (J1)

The Target Board can be powered from three different sources: 1) The regulator input from the P2 9 V DC Power Adapter, 2) The 5 V VBUS signal if P4 is connected to a USB bus, and 3) The 5 V USB bus if a USB Debug Adapter is connected to the Debug Header (J9). Place a shorting block at header J1[REG_IN-P1_PWR] to power the board directly from an AC/DC Power Adapter. Place a shorting block at header J1[REG_IN-VBUS] to power the board from the USB bus connected to P4. Place a shorting block at header J1[REG_IN-SER_PWR] to power the board from the USB Debug Adapter. Please note that the second option is not supported if a USB bus is not connected to P4 and the third option is not supported with either the EC1 or EC2 Serial Adapters.

Note: Only one power option should be selected at one time.

9. Schematics

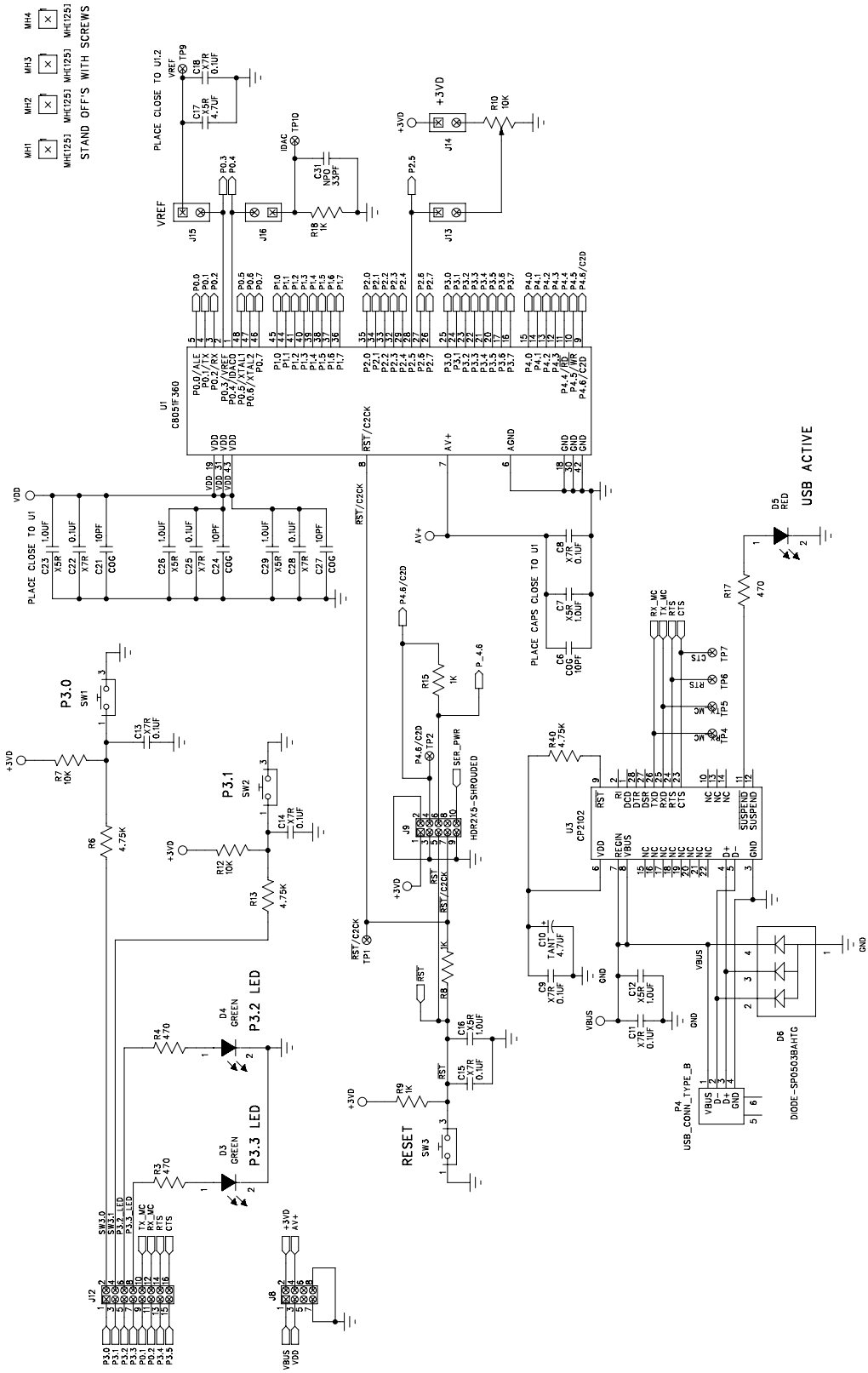

Figure 4. C8051F360 Target Board Schematic (Page 1 of 2)

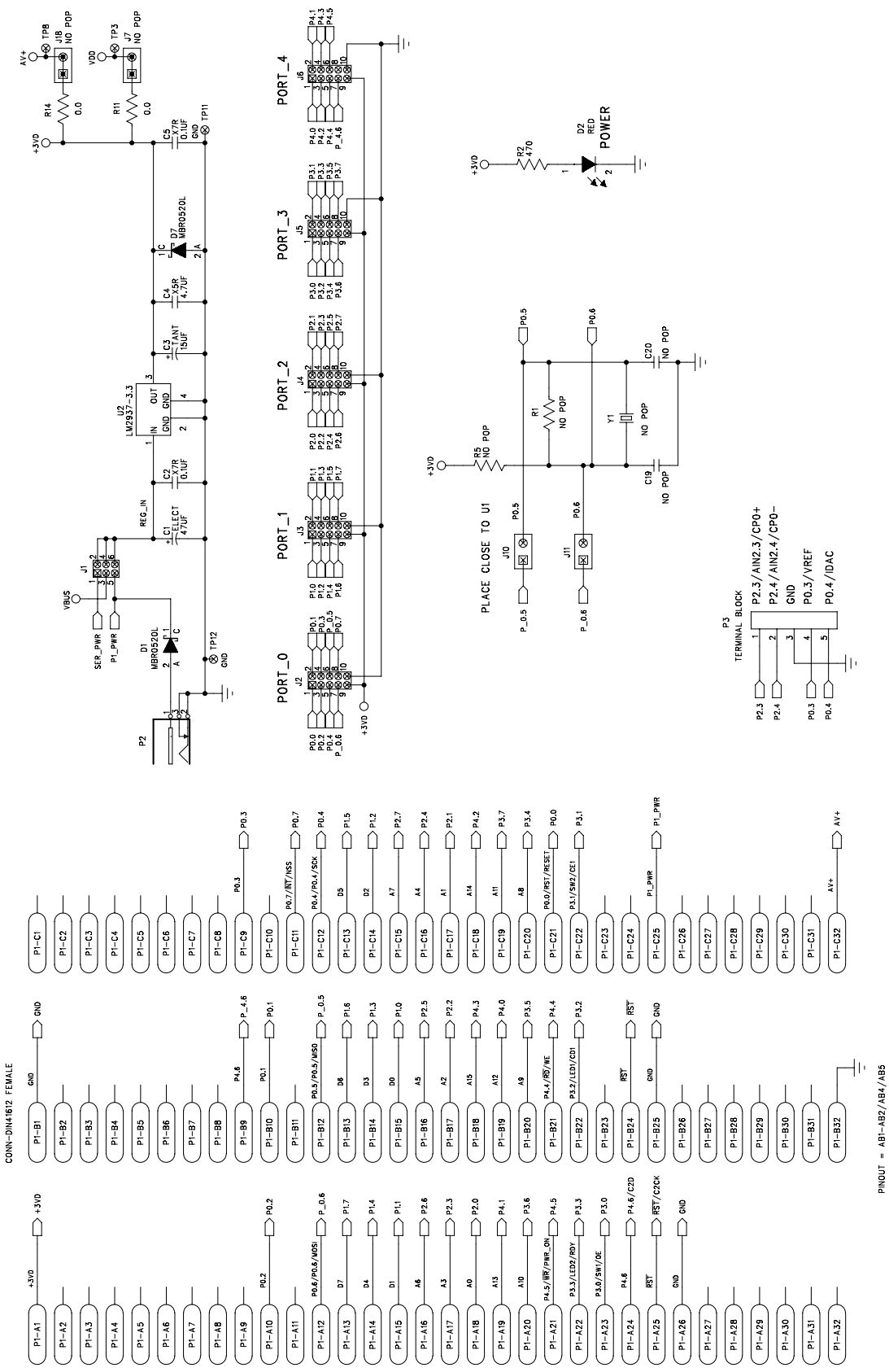

Figure 5. C8051F360 Target Board Schematic (Page 2 of 2)

NOTES:

DOCUMENT CHANGE LIST

Revision 0.1 to Revision 0.2

- Added Relevant Devices section.
- Section 2 moved to Section 5.
- Change section 3 to "Getting Started."
- Updated section 3 to include latest VCP driver installation instructions.
- Changed section 4 to "Software Overview."
- Updated Evaluation Compiler restrictions in section 4.2.2.
- Added overview of Configuration Wizard 2 and Keil uVision Drivers to section 4.
- Created new section 6.

CONTACT INFORMATION

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
Tel: 1+(512) 416-8500
Fax: 1+(512) 416-9669
Toll Free: 1+(877) 444-3032

Email: MCUinfo@silabs.com
Internet: www.silabs.com