Si825x Multi-phase POL Reference Design User's Guide

1. Kit Contents

The non-isolated Si825x Multi-Phase Point of Load (POL) Reference Design contains the following items:

- 40 Amp Si8250-based Multi-Phase POL Target Board
- USB to SMBus™ Bridge Board
- USB Debug Adapter
- USB Cable
- Silicon Laboratories IDE and Product Information CD-ROM. CD content includes the following:
  - Silicon Laboratories Integrated Development Environment (IDE)
  - Keil 8051 Development Tools (macro assembler, linker, evaluation C compiler)
  - Source code examples and register definition files
  - Multi-Phase POL Firmware Kernel
  - Application Builder tool suite (waveform editor, compensator, system and MCU wizards)
  - SMBus™ Monitor Software
  - Documentation: Kernel flowcharts and Si825x data sheet
- Si825x Multi-Phase POL Reference Design User's Guide (this document)

Note: The full version of the Kernel when compiled is approximately 14 kB. This exceeds the 4 kB limit of the compiler that is shipped with the kit's software development tools. To avoid a compiler limit issue, either buy the full Keil compiler toolset or compile the limited version of the Kernel.

Copyright © 2006 by Silicon Laboratories

2. Hardware Overview

The Si825x Multi-Phase POL Reference Design implements a digitally-controlled POL with a DPWM (digital pulse width modulation) switching frequency of 391 kHz. The Si8250 Multi-Phase POL Target Board (Figure 1) contains system power stages and digital control circuits with debug connectors for the Si8250 digital power controller. The user can also access and control the target board using SMBus.

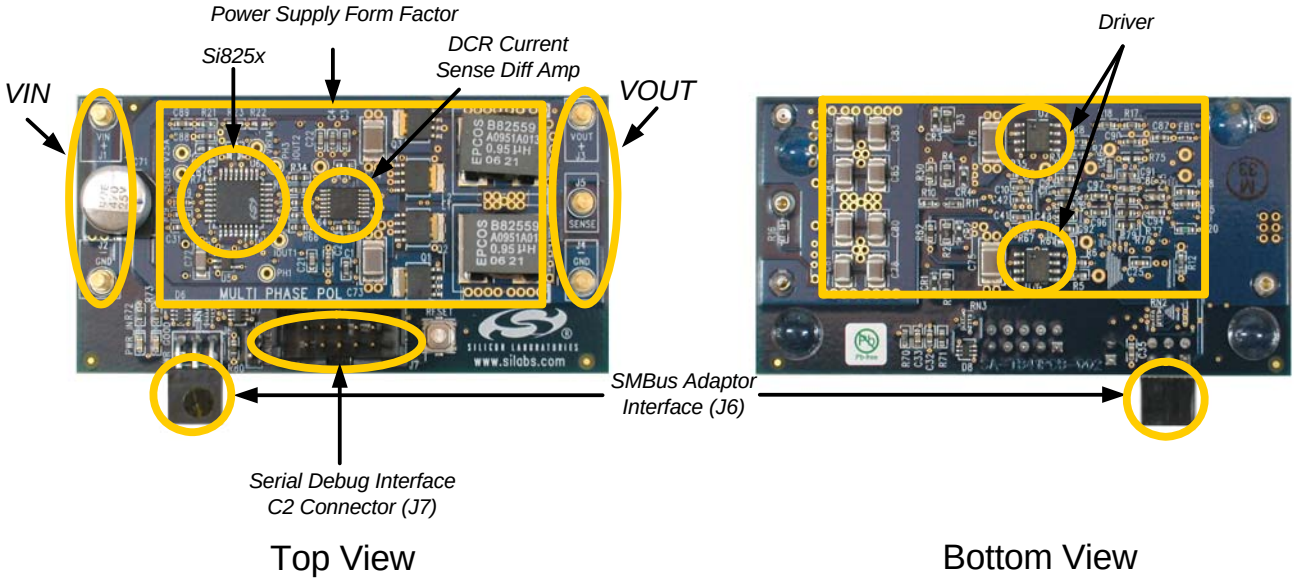


Figure 1. Si8250 Multi-Phase POL Target Board

3. Si8250 Multi Phase POL Target Board Stand-Alone Operation

The POL target board comes preloaded with firmware algorithms, and is designed to provide a 3.3 V output with up to 40 Amps of output current.

To operate the target board as a stand-alone power supply, perform the following steps:

1. Connect a V_{IN} power supply to the V_{IN} terminals (J1 and J2) as shown in Figure 2. Do not yet turn the power supply on.
2. Connect a load to the V_{OUT} terminals (J3 and J4) as shown in Figure 2. For initial testing, a 2 Ω, 15 W resistor is recommended. For higher output currents, an electronic load simulator can be used.
3. Turn the V_{IN} supply on. The converter will start up and provide 3.3 V at V_{OUT}.



Figure 2. Board Power Configuration

4. Development/Debug Operation: Initial Hardware Setup

This section describes the use of the Si8250 Multi-Phase POL Target Board with the Silicon Laboratories integrated development environment (IDE) and Application Builder tools. To configure the hardware for connection to the IDE:

1. Connect a V_{IN} power supply to the V_{IN} terminals (J1 and J2) as shown in Figure 2. Do not yet turn the power supply on.
2. Connect the USB Debug Adaptor's ribbon cable to the Si8250 Multi-Phase POL Target Board at J7 as shown in Figure 3.
3. Connect the USB cable to the USB Debug Adapter's USB input plug.
4. Connect the USB cable from the USB Debug Adapter to a USB port on the PC.
5. Turn the V_{IN} supply on.



Figure 3. Overall View of the Debug Connection (Load not Connected)

5. Development/Debug Operation: Software Setup

The Si825x Multi-Phase POL Reference Design comes with Application Builder software (detailed in "8. Si825x Application Builder" on page 12), a configurable real-time software Kernel, and a software Kernel compiled specifically for the Si8250 Multi-Phase POL Target Board.

The Kernel is royalty-free application software for the Si825x family of digital power controllers that greatly reduces application program development time, effort, and engineering risk. The Application Builder is used to customize the Kernel and create C-code source level application software for the Si825x end application. The Application Builder directly modifies the source code in the Kernel, which is then compiled and downloaded to the Si825x. For more information on the Kernel, see application note "AN271: Si8250 Real-Time Kernel Overview".

The Multi_Phase_POL directory *(SiLabs\Power\Reference_Designs\Multi_Phase_POL\Firmware)* contains the Kernel configured for the Si8250 Multi-Phase POL Target Board.

Kernel software can be loaded/reloaded to the target board using the Silicon Laboratories IDE. Note that hardware must be set up as detailed in "4. Development/Debug Operation: Initial Hardware Setup" on page 4. Follow the instructions below to configure and download the **Multi_Phase_POL** application software for the Si8250 Multi-Phase POL Target Board.

Note: A thorough understanding of the IDE is required before one can use the development/Debug Mode of the kit. The IDE is detailed in "7. Silicon Laboratories Integrated Development Environment".

1. The included CD-ROM contains the Silicon Laboratories Integrated Development Environment (IDE), Application Builder examples, the PMBus Monitor, 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 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. See "7. Silicon Laboratories Integrated Development Environment" on page 9 for further information on the development tools.
2. Open the IDE by selecting **Silicon Laboratories→Silicon Laboratories IDE** from the PC programs menu.
3. Next, the example project included with the kit is opened. Select **Project→Open Project...** from the IDE menus. In the Project Workspace window, browse to the "*SiLabs\Power\Reference_Designs\Multi_Phase_POL\Firmware\Multi_phase_POL\Source*" directory and select the ***.wsp project** file. Press **Open** to close the window and open the project.

   Note: This example will only work with the full version of the Keil compiler. If the demonstration compiler is used, use the files in "*SiLabs\Power\Reference_Designs\Multi_Phase_POL\Firmware\Multi_phase_POL_basic\Source".* This code will compile to less than 4 kB of code.
4. The Si8250 Multi-Phase POL Target Board has several connection requirements that must be specified before connecting to the board. Select **Options→Connection Options...** from the IDE menu. In the Connection Options window, select USB Debug Adapter in the Serial Adapter section. Next, select C2 in the Debug Interface section. The Si825x family of devices use the Silicon Laboratories 2-wire (C2) debug interface. Click OK to close the window.
5. Click the **Connect** button in the toolbar or select **Debug→Connect** from the menu to connect to the device.
6. Build the project by clicking on the **Build/Make Project** button in the toolbar or by 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.

7. 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. If errors occur during the build process, the IDE will not attempt the download.

8. Connect a load to the V_{OUT} terminals (J3 and J4) as shown in Figure 2 on page 3. For initial testing, a 2 Ω, 15 W resistor is recommended. For higher output currents, an electronic load simulator can be used.
9. Run the converter firmware by pressing the “Go” button in the IDE toolbar. The converter will start up and provide 3.3 V at V_{OUT}.
10. Save the project when finished with the debug session to preserve the current target build configuration, editor settings, and location of all open debug views. To save the project, select **Project→Save Project** from the menu.

6. PMBus Operation

PMBus is a connectivity solution designed for networking multiple power supplies using a single management bus. The Real-time Kernel provided with the Si825x Multi-Phase POL Reference Design includes optional support for PMBus. In addition, the Si825x POL design kit also comes with a PMBus Monitor application and USB to SMBus Bridge Board to manage the power supply through PMBus.

1. The PMBus Monitor software is installed during the initial software setup. (See "5. Development/Debug Operation: Software Setup" on page 5.)
2. Connect the board as shown in "3. Si8250 Multi Phase POL Target Board Stand-Alone Operation" on page 3. Note that the PMBus Monitor may also be operated with the USB Debug Adaptor. If this is desired, connect the target board as shown in "4. Development/Debug Operation: Initial Hardware Setup" on page 4.
3. Drivers must be installed to allow the PMBus Monitor to communicate with the USB to SMBus Bridge Board. The driver files are located by default in the "*Silabs\Power\Si825x AppBuilder\PMBus Monitor\USB-SMBus Bridge Board Drivers*" directory. Run the *PreInstaller.exe* application. This program will copy the driver files to the PC's "Program Files" directory and then register the driver files so the board will be recognized when it is connected. Windows Logo testing warnings may appear. Press the **Continue Anyway** button.
4. Connect the USB to SMBus Bridge Board to an available USB slot on your PC with a USB cable.
5. Windows will open a **Found New Hardware Wizard** window. Press **Next** after selecting the **(Recommended)** option. Windows Logo testing warnings may appear. Press the **Continue Anyway** button. Press "Finish" to finish installing the USB to SMBus Bridge Board.
6. Connect the Si8250 Multi-Phase POL Target Board to the USB to SMBus Bridge Board as shown in Figure 4.



Figure 4. USB to SMBus Bridge Board Connection (Load not Connected)

7. Open the Application Builder by selecting **Silicon Laboratories→Si825x Application Builder** from the PC programs menu.
8. Run the PMBus Monitor application by selecting **Options→Launch PMBus Monitor Tool** from the Application Builder. The window shown in Figure 5 will appear. The PMBus Monitor can be used to control and configure the target board. The target can be enabled/disabled through the monitor. The PMBus Monitor allows parameters, such as fault thresholds, to be changed. It also reports operating conditions and problems.

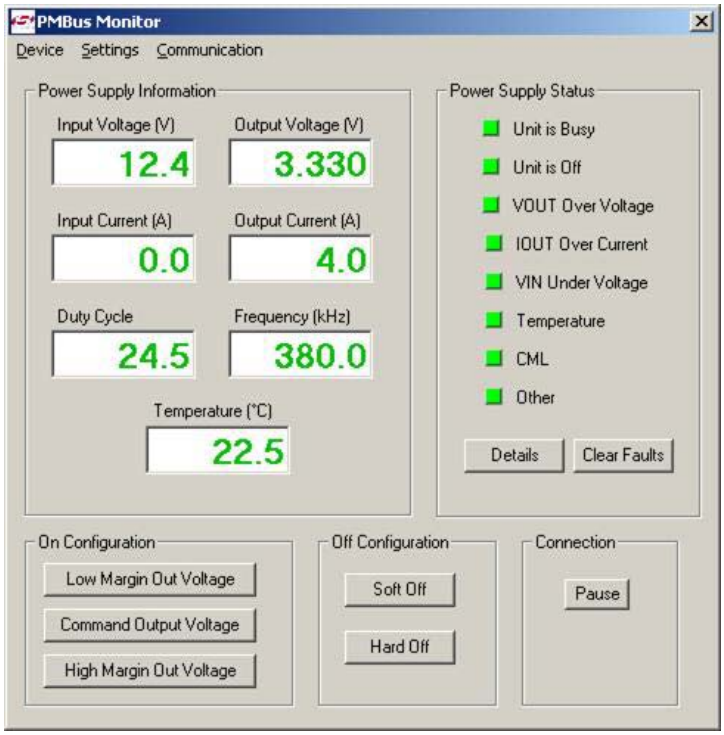


Figure 5. View of the PMBus Monitor

7. Silicon Laboratories Integrated Development Environment

The Silicon Laboratories IDE combines an editor, project manager, code development tools, and a debugger into a single intuitive environment for code development and in-system debugging. No additional target RAM, program memory, or communications channels are required. The use of third-party compilers and assemblers is also supported. This development kit includes the Keil Software A51 macro assembler, BL51 linker, and evaluation version C51 C compiler. These tools can be used from within the Silicon Laboratories IDE. Figure 6 shows the IDE.

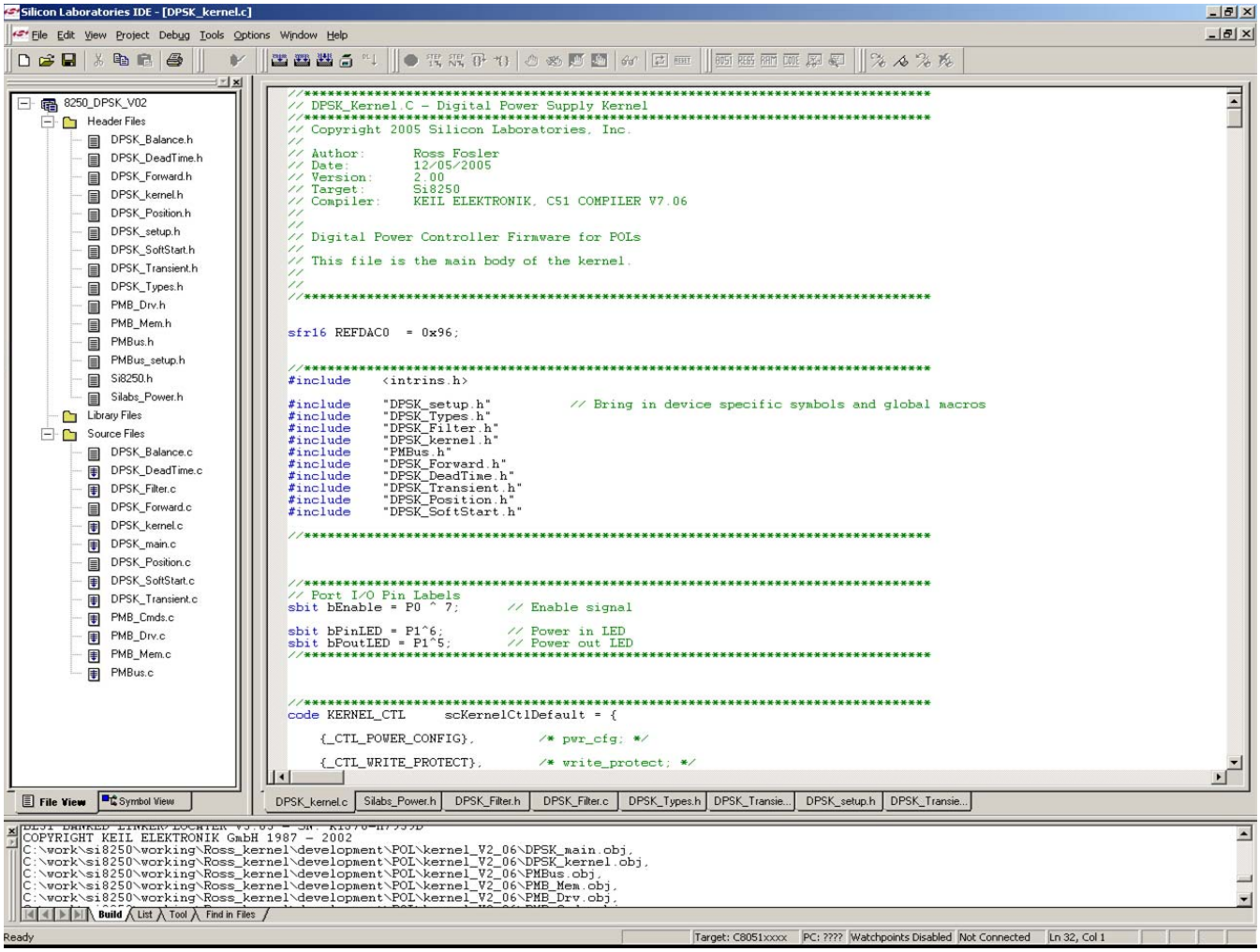


Figure 6. IDE

7.1. System Requirements

Silicon Laboratories IDE requirements are as follows:

- Pentium-class host PC running Microsoft Windows 95 or later or Microsoft Windows NT or later.
- One available USB port.
- 64 MB RAM and 40 MB free HD space recommended.

7.2. Assembler and Linker

A full-version Keil A51 macro assembler and BL51 banking linker are included with the development kit and are installed during IDE installation. The complete assembler and linker reference manual can be found online under the **Help** menu in the IDE or in the "*SiLabs\MCU\hlp*" directory (A51.pdf).

7.3. Evaluation C51 C Compiler

An evaluation version of the Keil C51 C compiler is included with the development kit and is installed during IDE installation. The evaluation version of the C51 compiler is the same as the full professional version except, that code size is limited to 4 kB, and the floating point library is not included. The C51 compiler reference manual can be found under the **Help** menu in the IDE or in the "*SiLabs\MCU\hlp*" directory (C51.pdf).

7.4. 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" 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.)

7.4.1. 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 them. Right-click on **New Project** in the **Project Window**. Select **Add Groups to project**. Add predefined 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.

7.4.2. 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 adapter that was included with the kit in the "Serial Adapter" section. Next, the correct "Debug Interface" must be selected. Si825x family devices use the Silicon Laboratories 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. If errors occur during the build process, the IDE will not attempt the download.

5. When finished with the debug session, save the project 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**.

7.5. Si825x Debug Mode

The IDE contained in the Si825x Multi-Phase POL Reference Design has an online debug feature that optionally enables the user to inspect or update special function registers (SFRs) in the Si825x while it is operating. For example, filter coefficients can be optimized by simply typing in new coefficient values while the supply is connected to a network analyzer and running. The IDE can also be operated in Standard mode where SFR inspect and update is allowed only when the Si825x is not running. As shown in Figure 7, the IDE can be set for Online Debug Mode or Standard Debug mode by clicking on the circled mode switch.

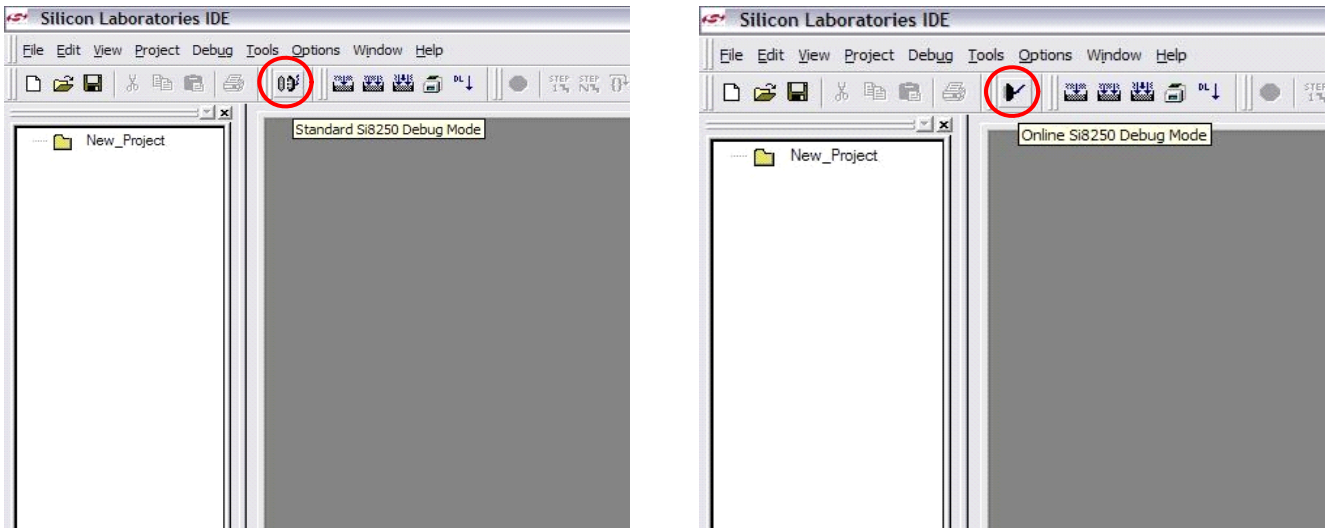


Figure 7. Si825x IDE Debug Modes

8. Si825x Application Builder

In addition to the IDE, the Si825x family is supported with an intuitive toolset that leverages traditional power supply control design methods minimizing the digital supply design learning curve. The toolset consists of a real-time firmware Kernel (C-language source code), and the **Application Builder** (see Figure 8). The Application Builder includes device peripheral configuration options with the **Peripheral Configuration Wizard**. Additionally, the Application Builder tools are used to modify Kernel operations. Three key development tools within the Application Builder are the **DPWM Timing Diagram Editor**, the **System Parameter Programmer**, and the **Compensation Editor**. These tools are detailed in the following sections. The flow diagrams for the Kernel are included on the CD.

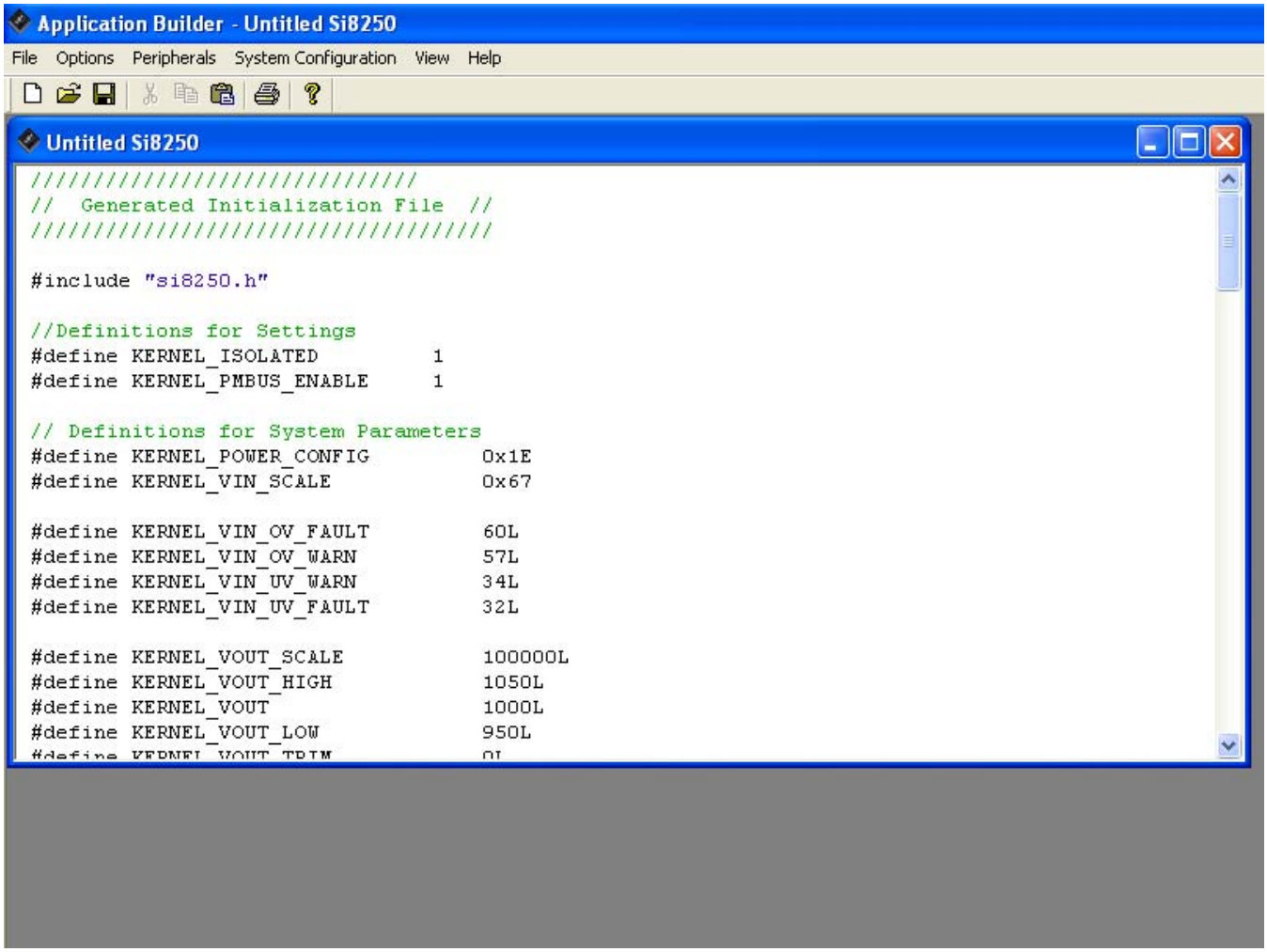


Figure 8. Si825x Application Builder

8.1. DPWM Timing Diagram Editor

The **DPWM Timing Diagram Editor** permits designers to generate DPWM initialization code by simply drawing the timing for their end system. The wizard accommodates up to six output phases and can be used to establish positive or negative dead-times, relative edges, absolute edges, and other timing required by the end system. Refer to the Si825x data sheet for a description of the different edges.

This example illustrates how to use the **DPWM Timing Diagram Editor** to create and simulate the timing for a single-phase POL as well as generate the initialization code in the Kernel.

1. To open the **DPWM Timing Diagram Editor** window, open the Application Builder and select **System Configuration→DPWM Timing Diagram Editor** from the menu.
2. To create an absolute edge on Phase 1, hold the mouse above the Phase 1 zero timing line (default) at time tick 10. Then, either double click with the left mouse button or right-click at that point and select **Absolute Edge**. An absolute edge at time tick 10 will be created (see Figure 9).
3. To finish the timing for Phase 1, specify hardware modulation using **(Cu0)**. This event edge will be modulated relative to its absolute edge at time tick 10. To create this edge, hold the mouse above the Phase 1 timing line to the right of time tick 10 at time tick 60. Either double-click or right-click at that point and select **Event (Cu0) Edge**. Next, select the edge to reference by clicking on edge 1 in Phase 1. A relative falling (Cu0) edge will be created at time tick 60 since u(n) defaults to 50 (see Figure 9).

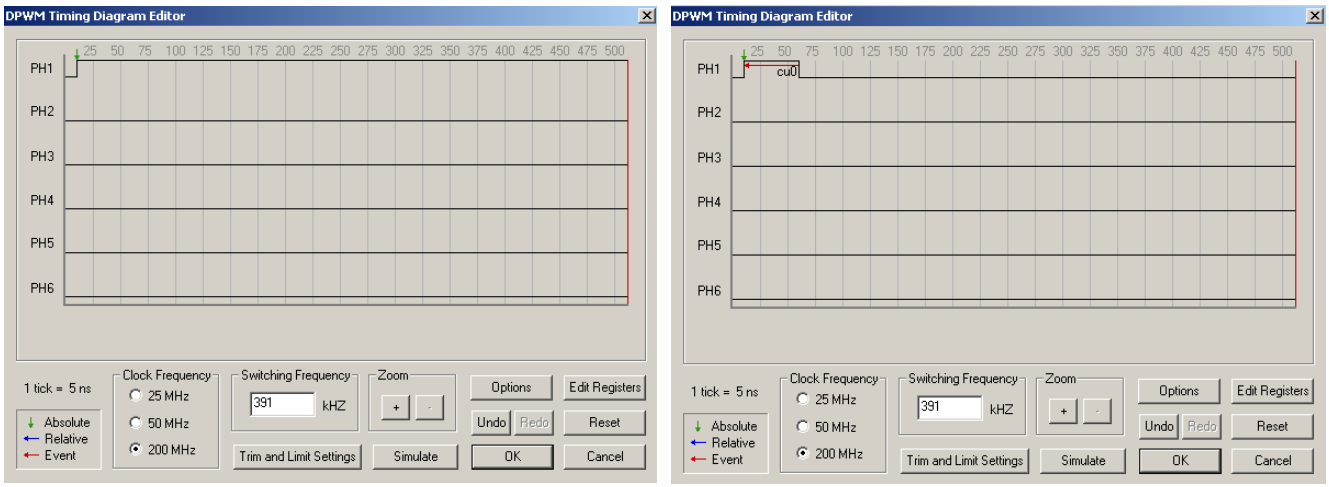


Figure 9. DPWM Timing Diagram Editor—Phase 1

1. Now, create the timing for Phase 2. For Phase 2, the goal is to create a relative rising edge relative to the falling edge of Phase 1 and an absolute falling edge on Phase2 at time tick 505. To create the relative rising edge, hold the mouse above the Phase 2 zero timing line (default) at time tick 100. Then, either double-click with the left mouse button or right-click at that point and select **Relative Edge**. Next, select the edge to reference by clicking on Phase 1's falling edge. A relative edge at approximately time tick 90 will be created (see Figure 10).
2. To finish the timing for Phase 2, create an absolute falling at time tick 505. To create this edge, hold the mouse above the Phase 2 timing line at time tick 505. Then, either double-click with the left mouse button or right-click at that point and select **Absolute Edge**. An absolute edge at time tick 505 will be created (see Figure 10).

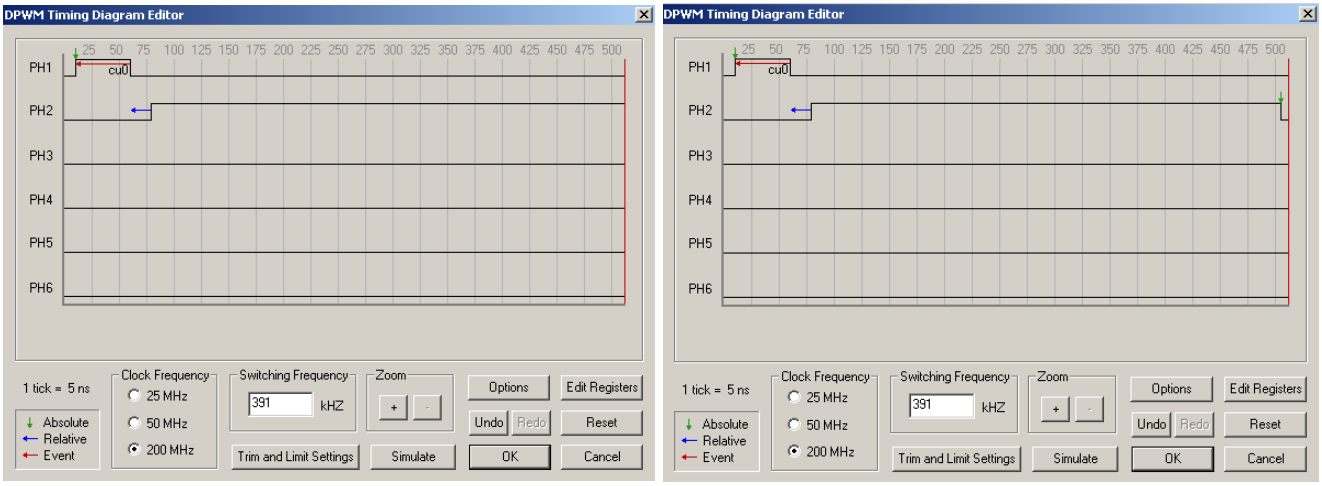


Figure 10. DPWM Timing Diagram Editor—Phase 2

3. Click the **Simulate** button to display the Simulate Window (see Figure 11). Use the arrows to increase and decrease the value of u(n). Notice that Phase 1's edge should modulate with its absolute edge starting at 10 ticks. Phase 2 should also modulate. However, its rising edge starts relative to Phase 1's falling edge.

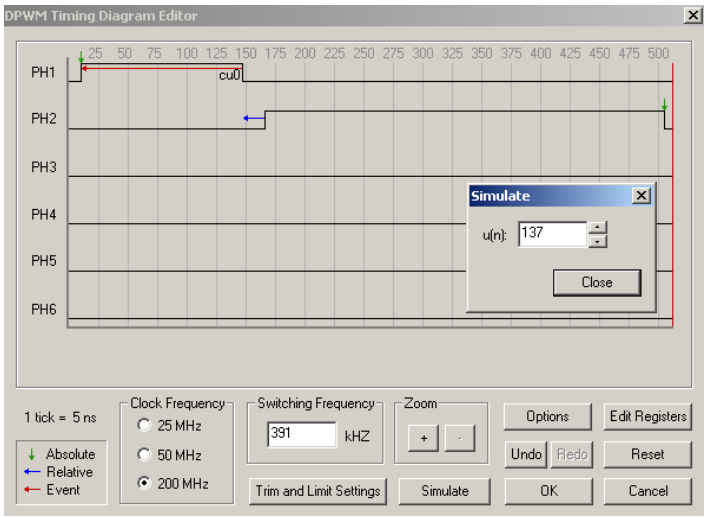


Figure 11. DPWM Timing Diagram Editor—Timing Simulator

4. Now that the desired timing for the two-phase system has been created, click on **OK**. The Application Builder will automatically extract the correct DPWM timing initialization data for the Kernel. These coefficients can be saved to a new project file if desired for later use. Click on **File→Save Project** to save this project. To generate an IDE project, select **File→Build IDE Project...** and select the directory for project generation.

8.2. Compensation Editor

The **Compensation Editor** is a loop simulation and coefficient generator tool for frequency compensating the system. To open the **Compensation Editor** window, open the Application Builder and select **System Configuration→Compensation Editor** from the menu. The Compensation Editor for a buck regulator is shown in Figure 12. As shown, this tool provides fields for the user to enter power stage parameters, such as the output filter component and parasitic values; controller parameters, such as PWM frequency and pole/zero locations; and Si825x-specific data, such as ADC sample frequency. The simulator comes pre-populated with default values for the **Si8250 Half-Bridge Target Board**. These model parameters can be changed as desired to accommodate other buck topologies. To view the gain and phase plots with the default parameters, click on the View Graphs button, and their plots will be generated a short time later. Moreover, clicking on the View Graphs button, the user can view different responses of the buck regulator. Sample graphs are shown in Figure 12.

Table 1 shows two frequency responses for the multi-phase POL: one for steady-state operation and one for operation during a transient. During transients, the Si8250 automatically extends loop bandwidth by writing faster coefficients to the loop compensation filter (DSP filter engine). This nonlinear control response improves system transient response, reducing both the magnitude and duration of the output transient.

Table 1. Si8250 Multi-Phase POL Target Board's Frequency Response

	Steady-State Response	Transient Response
Loop Gain Bandwidth	45 kHz	95 kHz
Phase Margin	54º	25º

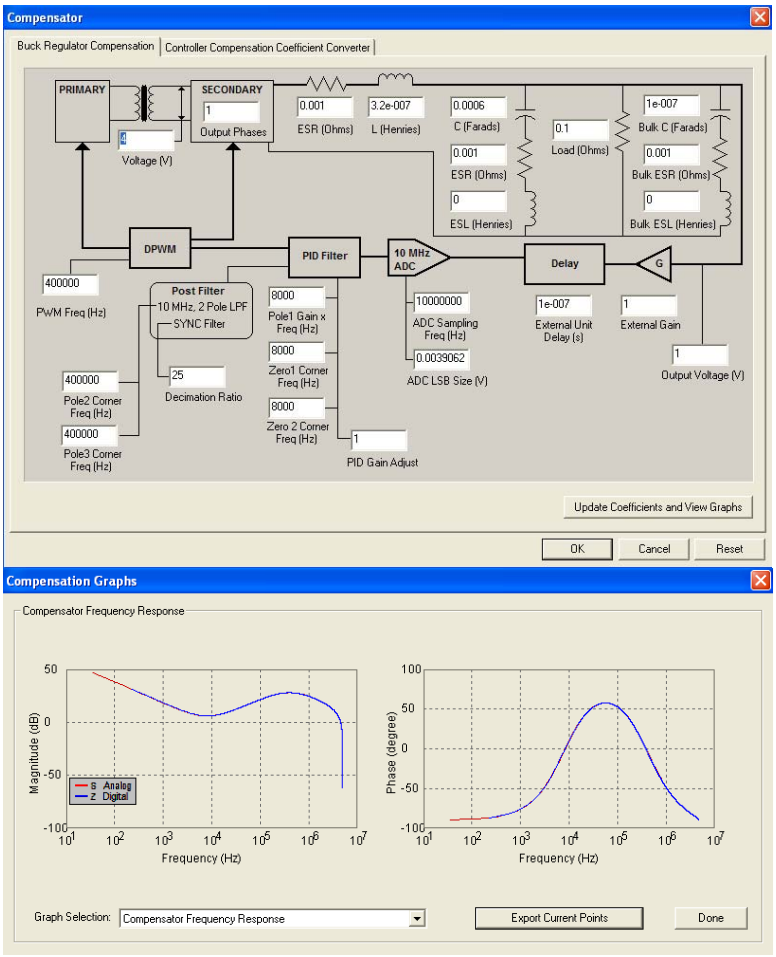


Figure 12. Compensation Editor, Input/Output Windows

8.3. System Parameter Programmer

The **System Setting Programmer** allows the designer to input all system settings (UVLO, OV, OCP, etc.) and then converts these parameters to HEX and populates the resulting initialization code in the Kernel. To open the **System Settings** window, open the Application Builder and select **System Configuration→System Settings** from the menu (see Figure 13).

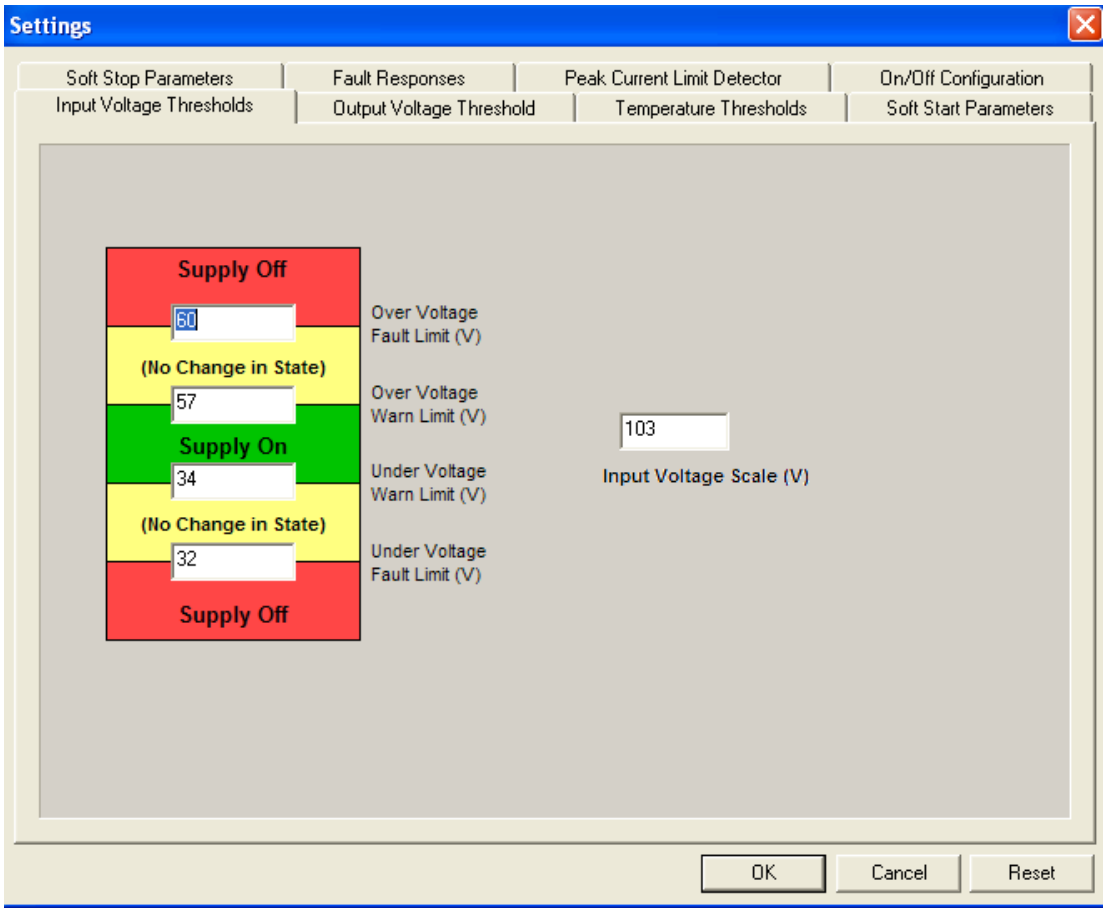


Figure 13. System Parameters Window

8.4. Peripheral Configuration Wizard

The **Peripheral Configuration Wizard** can be used to automatically generate initialization code for the Si825x's on-chip peripherals (ADC2, comparator, UART, SMBus™, etc.). The peripheral windows can be accessed by clicking on the **Peripherals** menu in the Application Builder. Figure 14 illustrates the **Port I/O** window. For more details on using this wizard, consult the "Help" file by clicking on **Help→Help...**.

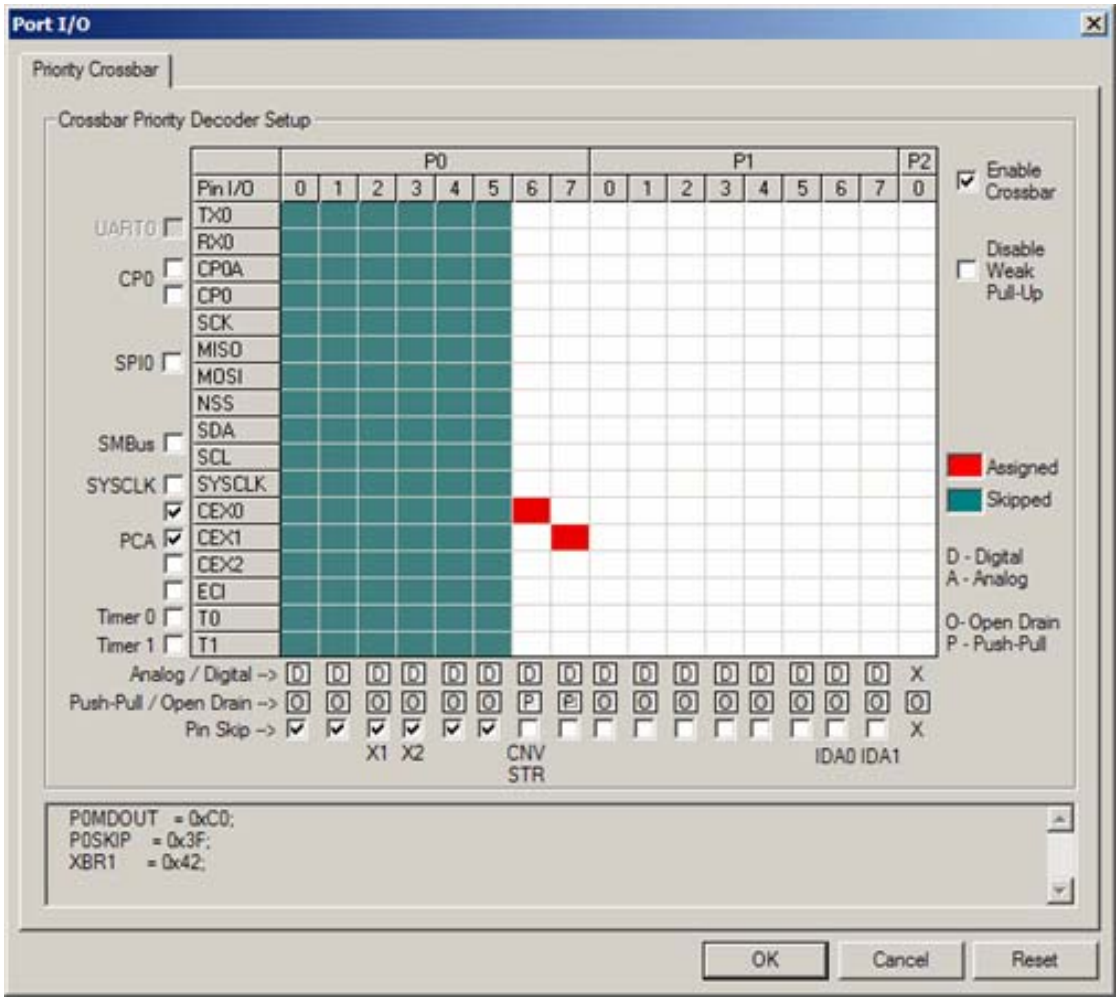


Figure 14. Peripheral Configuration Wizard—Port I/O Window

9. Restoring Factory Defaults

The Si825x Multi-Phase POL Reference Design includes hex files created for the multi-phase POL application. Downloading these hex files to the Si8250 Multi-Phase POL Target Board will restore the board to its factory defaults.

9.1. Restoring the Si8250

To download the factory default Si8250 hex file to the target board, perform the following steps:

1. Connect the USB Debug Adaptor's ribbon cable to the Si8250 Multi-Phase POL Target Board at J7 as shown in Figure 3 on page 4.
2. Open the IDE by selecting **Silicon Laboratories→Silicon Laboratories IDE** from the PC programs menu.
3. The Si8250 Multi-Phase POL Target Board has several connection requirements that need to be specified before connecting to the board. Select **Options→Connection Options...** from the IDE menu. In the **Connection Options** window, select **USB Debug Adapter** in the Serial Adapter section. Next, select **C2** in the Debug Interface section. The Si825x family of devices use the Silicon Laboratories 2-wire (C2) debug interface. Press OK to close the window.
4. Click the **Connect** button in the toolbar or select **Debug→Connect** from the menu to connect to the device.
5. Select **Debug→Download Object File...** from the IDE menus to open the download window.
6. Press the **Browse** button to open the **Download Filename...** window.
7. In the **List files of type:** drop down box, select the **Intel-Hex** option.
8. Browse to the "*SiLabs\Power\Reference_Designs\Multi_Phase_POL\Firmware\multi_phase_POL\hex*" directory and select the ***.hex** file. Press OK to close the window.
9. Press the **Download** button to download the file.
10. Click the **Disconnect** button in the toolbar, or select **Debug→Disconnect** from the menu to disconnect from the device.
11. Power cycle the device to run the downloaded program.

10. Si8250 Multi-Phase POL Target Board

The Si8250 Multi-Phase POL Target Board has a Si8250-IQ installed. Refer to Figure 15 for the locations of the various I/O connectors and major components.

- J1, J2 VIN, Supply Input power connection 10-15 V, 10 A
- J3, J4 VOUT, Supply output connection for load simulator
- J5, J4 Alternate VOUT monitoring node
- J6 SMBus adapter
- J7 Si8250 Debug Interface
- JP1 Si8250 ENABLE Polarity Select
- LED VIN Power Good Indicator
- LED VOUT Power Good indicator
- RESET Si8250 and Target Board Reset

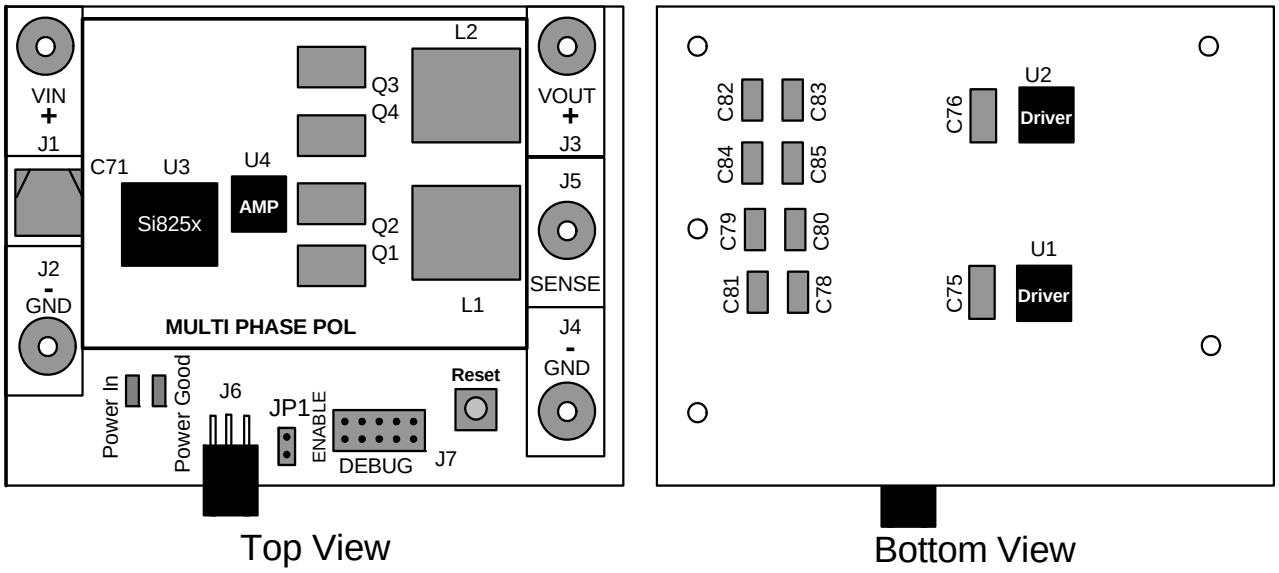


Figure 15. Si8250 Multi-Phase POL Target Board

10.1. System Clock Sources

The Si8250-IQ device installed on the target board features a calibrated programmable internal oscillator that is enabled as the system clock source on reset. After reset, the device operates at a frequency of 80 kHz by default using the internal low-frequency oscillator but may be configured by software to operate at other frequencies. Refer to the Si825x family data sheet for more information on configuring the system clock source.

10.2. Switches and LEDs

One switch is provided on the target board. Switch S1 is connected to the RESET pin of the Si8250. Pressing S1 puts the Si8250 device into its hardware-reset state.

Two LEDs are also provided on the target board. The POWER IN LED is used to indicate that the POL target board is properly powered. The POWER GOOD LED is used to indicate that VOUT power is being regulated by the Si8250. See Table 2 for a description of each LED and on board switches.

Table 2. Target Board Switch and LED Descriptions

Label	Reference	Description
RST	S1	Si8250 Reset Switch
POWERIN	LED	Indicates that V_{IN} is operating properly
POWERGOOD	LED	Indicates that V_{OUT} is regulated properly

10.3. V_{IN} (J1, J2)

The user-provided power supply should be connected to connectors J1 and J2 where J2 is the reference. The power source must be from 10–15 V with at least 10 A maximum output.

Table 3. J1, J2 Pin Descriptions

Jumper #	Description
J1	10–15 V, 10 A
J2	GND

10.4. V_{OUT} (J3, J4)

Connectors J3 and J4 are the dc output from the Si8250 Multi-Phase POL Target Board. A load (preferably electronic) should be connected here.

Table 4. J3, J4 Pin Descriptions

Jumper #	Description
J3	3.3 V (Nominal), 40 A max
J4	GND

10.5. V_{OUT} (J5, J4)

Connectors J5 and J4 provide an alternate output monitoring note that minimizes output loading while V_{OUT} is being monitored.

Table 5. J5, J4 Pin Descriptions

Jumper #	Description
J5	Alternate V_{OUT} Sense, minimal loading
J4	GND

10.6. SMBus Connector (J6)

The J6 connector is the SMBus interface connector for the Si8250 Multi-Phase POL Target Board. Table 6 shows the J6 pin definitions.

Table 6. J6 Pin Descriptions

Pin #	Description
1	SCL
2	DGND
3	SDA
4	DGND
5	SMBA
6	DGND

10.7. Si8250 DEBUG Interface (J7)

The Si8250 DEBUG connector (J7) provides access to the DEBUG (C2) pins of the Si8250 device on the target board. It is used to connect the USB Debug Adapter to the target board for in-circuit debugging and programming. Table 7 shows the J7 Si8250 DEBUG pin definitions.

Table 7. P1 Si8250 DEBUG Connector Pin Descriptions

Pin #	Description
1	+2.5 V
2, 3, 9	DGND
4	C2D
7	RST/C2CK
5, 6, 8, 10	Not Connected

10.8. Si8250 ENABLE Polarity Select (JP1)

The JP1 jumper selects the polarity of the reset line for the Si8250. When installed, the reset signal is active high; otherwise, it is active low.

10.9. Voltage and Current Sense Test Points

The Si8250 Multi-Phase POL Target Board has several test points for V_{REF}, $V_{INSENSE}$, I_{IN}, V_{SENSE}, all PHn, I_{PK}, and more. These test points correspond to the respective pins on the Si8250-IQ integrated circuit as well as other useful inspection points. See “12. Schematics” on page 23.

11. USB Debug Adapter

The USB Debug Adapter provides the interface between the PC's USB port and the Si825x's in-system debug/programming circuitry. The attached 10-pin DEBUG ribbon cable connects the adapter to the target board and the target device's debug interface signals. (The USB Debug Adapter supports both Silicon Laboratories JTAG and C2 debug interfaces.) Power is provided to the adapter from the USB connection to the PC. The USB Debug Adapter is capable of providing power to a circuit board via pin 10 of the DEBUG connector. The Si8250 Multi-Phase POL Target Board is not designed to be powered from this source. Table 8 shows the pin definitions for the DEBUG ribbon cable connector.

Note: The USB Debug Adapter requires a target system clock of 32 kHz or greater.
With the default settings, the USB Debug Adapter can supply up to 100 mA to a target system.

Table 8. USB Debug Adapter DEBUG Connector Pin Descriptions

Pin #	Description
1, 8	Not Connected
2, 3, 9	GND (Ground)
4	TCK (C2D)
5	TMS
6	TDO
7	TDI (C2CK)
10	USB Power

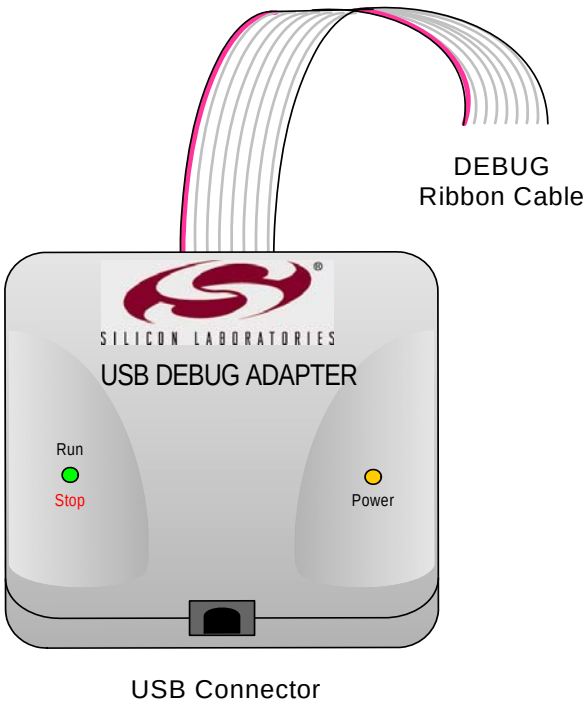


Figure 16. USB Debug Adapter

12. Schematics

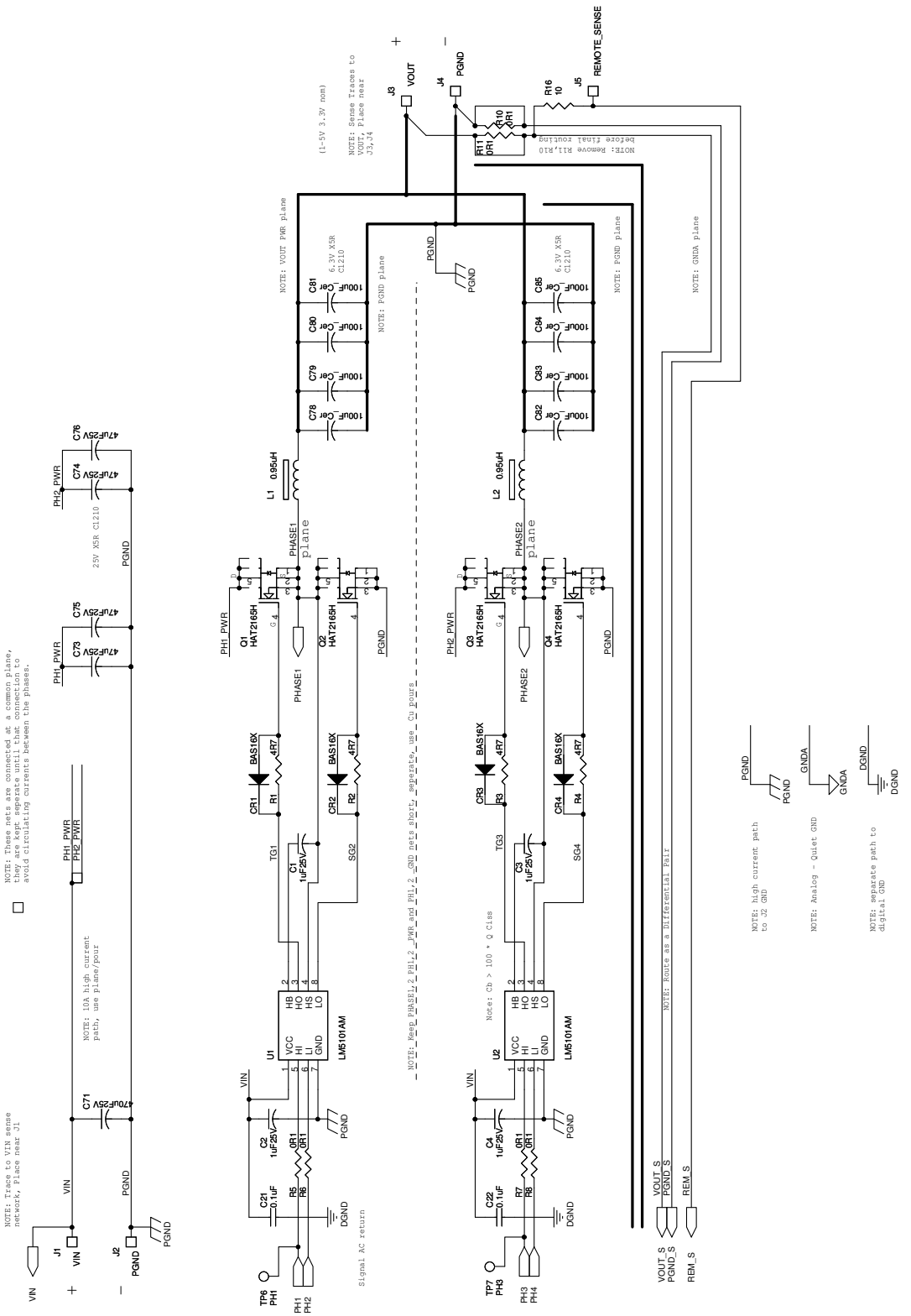


Figure 17. Si8250 Multi-Phase POL Target Board Schematic (Power)

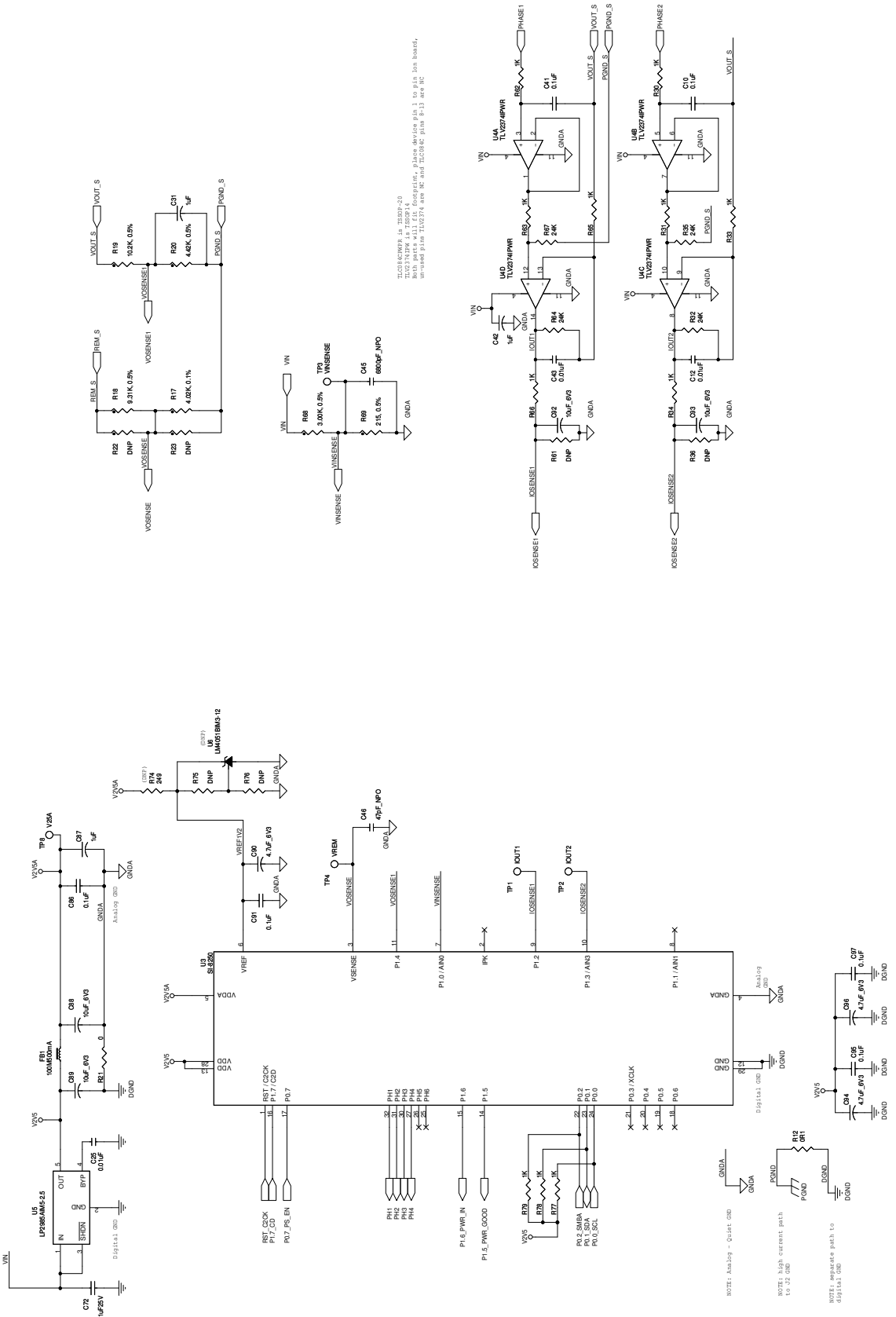


Figure 18. Si8250 Multi-Phase POL Target Board Schematic (Control)

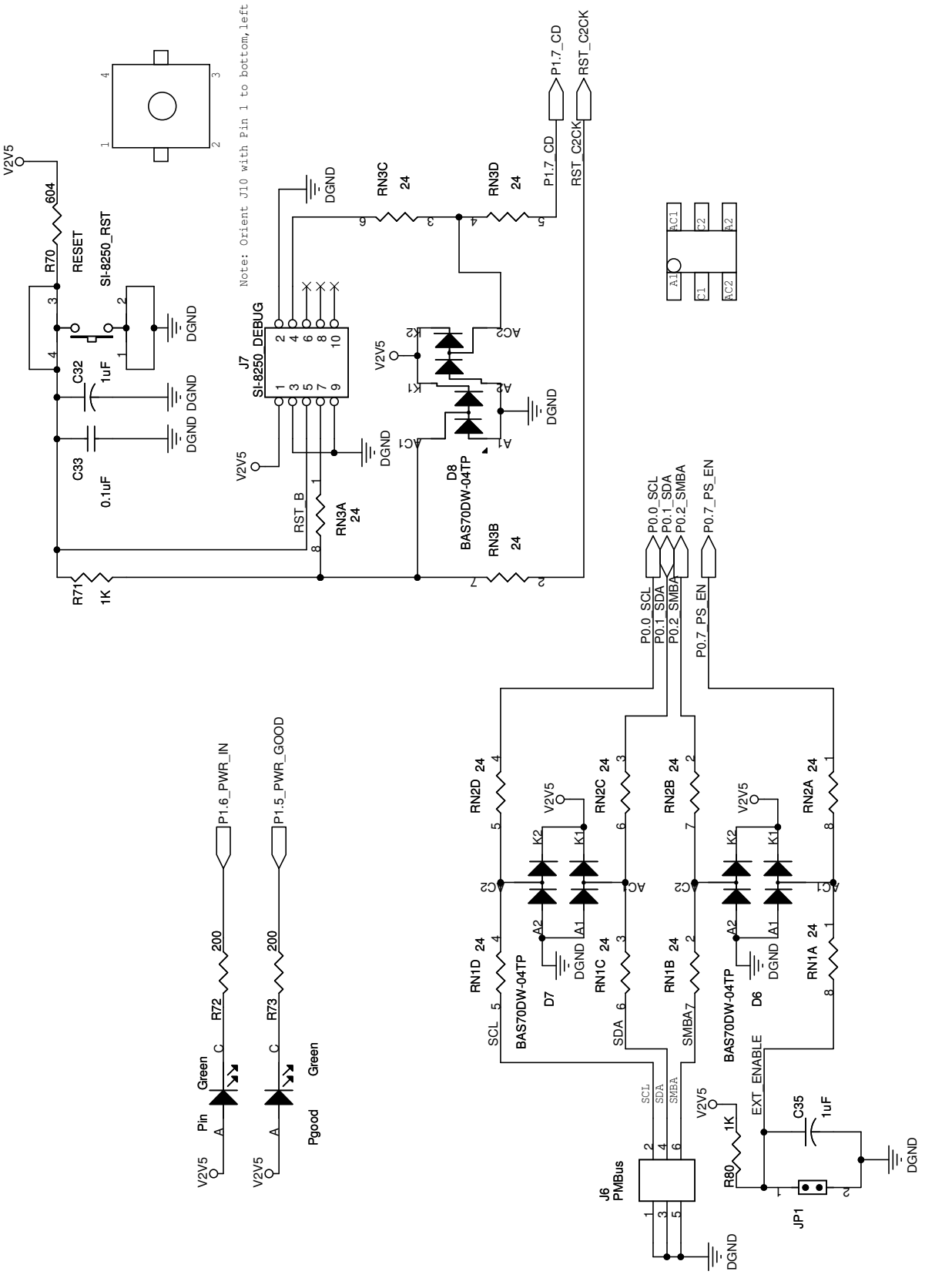


Figure 19. Si8250 Multi-Phase POL Target Board Schematic (Debug)

13. Bill of Materials

Table 9. Si825x Multi-Phase POL Reference Design Bill of Materials

Item	Qty	Reference	Part Number	Manufacturer	Description
1	4	CR1,CR2,CR3,CR4	BAS16XV2T1GOSCT-ND	Digi-Key	BAS16X
2	4	C1,C2,C3,C4	587-1248-1-ND	Digi-Key	1 uF, 25 V
3	9	C10,C21,C22,C33,C41,C86, C91,C95,C97	PCC2277CT-ND	Digi-Key	0.1 uF, 25 V
4	3	C12,C25,C43	478-1227-1-ND	Digi-Key	0.01 uF, 50 V
5	5	C31,C32,C35,C42,C87	PCC1915CT-ND	Digi-Key	1 uF, 6.3 V
6	1	C45		Venkel	6800 pF_NPO
7	1	C46	311-1065-1-ND	Digi-Key	47 pF_NPO
8	1	C71	565-2224-1-ND	Digi-Key	470 uF, 25 V
9	1	C72	PCC1893CT-ND	Digi-Key	1 uF, 25 V
10	4	C73,C74,C75,C76	587-1413-2-ND	Digi-Key	47 uF, 25 V
11	8	C78,C79,C80,C81,C82,C83, C84,C85	587-1387-2-ND	Digi-Key	100 uF_Cer, 6.3 V
12	4	C88,C89,C92,C93	PCC2395CT-ND	Digi-Key	10 uF, 6.3 V
13	3	C90,C94,C96	PCC2318CT-ND	Digi-Key	4.7 uF, 6.3 V
14	3	D6,D7,D8	BAS70DW-04-DICT-ND	Digi-Key	BAS70DW-04TP
15	1	FB1	240-2362-1-ND	Digi-Key	100M500 mA
16	1	JP1	WM4000-ND	Digi-Key	JUMPER1x2
17	1	J1	2551-2-00-44-00-00-07-0	Bisco	VIN
18	2	J2,J4	2551-2-00-44-00-00-07-0	Bisco	PGND
19	1	J3	2551-2-00-44-00-00-07-0	Bisco	VOUT
20	1	J5	2551-2-00-44-00-00-07-0	Bisco	REMOTE_SENSE
21	1	J6	S4416-ND	Digi-Key	PMBus
22	1	J7	MHB10K-ND	Digi-Key	SI-8250_DEBUG
23	2	L2, L1	871-B82559A951A13	Mouser	0.95 uH
24	2	Pin, Pgood	67-1549-1-ND	Digi-Key	Green
25	4	Q1,Q2,Q3,Q4		SiLabs	HAT2165H
26	1	RESET	P8046SCT-ND	Digi-Key	SI-8250_RST
27	3	RN1,RN2,RN3	Y7240CT-ND	Digi-Key	24, 5%
28	4	R1,R2,R3,R4	P4.7ACT-ND	Digi-Key	4R7, 5%
29	5	R5,R6,R7,R8,R12	P.10AHCT-ND	Digi-Key	0R1, 5%

Table 9. Si825x Multi-Phase POL Reference Design Bill of Materials (Continued)

Item	Qty	Reference	Part Number	Manufacturer	Description
30	2	R11,R10	P.10AHCT-ND	Digi-Key	0R1, 5%
31	1	R16	P10GCT-ND	Digi-Key	10, 5%
32	1	R17	RR08P4.02KBCT-ND	Digi-Key	4.02 K, 0.1%
33	1	R18	RR08P9.31KDCT-ND	Digi-Key	9.31 K, 0.5%
34	1	R19	RR08P10.2KDCT-ND	Digi-Key	10.2 K, 0.5%
35	1	R20	RN731JTTDK4421B25	Garrett	4.42 K, 0.5%
36	1	R21	P0.0GCT-ND	Digi-Key	0
37	2	R22,R23		Venkel	DNP
38	13	R30,R31,R33,R34,R62,R63, R65,R66,R71,R77,R78,R79, R80	P1.00KHCT-ND	Digi-Key	1 K, 1%
39	4	R32,R35,R64,R67	P24KGCT-ND	Digi-Key	24 K, 5%
40	3	R36,R61,R75			DNP
41	1	R68	RR08P3.01KDCT-ND	Digi-Key	3.00 K, 0.5%
42	1	R69		Digi-Key	215, 0.5%
43	1	R70	P604HCT-ND	Digi-Key	604, 1%
44	2	R73,R72	P200HCT-ND	Digi-Key	200, 1%
45	1	R74	P249HCT-ND	(DNP) Digi-Key	249, 5%
46	1	R76			DNP
47	1	TP1	5002K-ND	Digi-Key	IOUT1
48	1	TP2	5002K-ND	Digi-Key	IOUT2
49	1	TP3	5002K-ND	Digi-Key	VINSENSE
50	1	TP4	5002K-ND	Digi-Key	VREM
51	1	TP6	5002K-ND	Digi-Key	PH1
52	1	TP7	5002K-ND	Digi-Key	PH3
53	1	TP8	5002K-ND	Digi-Key	V25A
54	2	U1,U2	LM5101M-ND	Digi-Key	LM5101AM
55	1	U3		SiLabs	Si-8250
56	1	U4	595-TLV2374IPWR	Mouser	TLV2374IPWR
57	1	U5	LP2985AIM5-2.5CT-ND	Digikey	LP2985AIM5-2.5
58	1	U6	LM405	(DNP) Digi-Key	LM4051BIM3-12

DOCUMENT CHANGE LIST

Revision 0.1 to Revision 0.2

- Updated board photos in Figure 1, “Si8250 Multi-Phase POL Target Board,” on page 2 and Figure 2, “Board Power Configuration,” on page 3.
- Updated PMBus part number (Item 21) in Section "13. Bill of Materials” on page 26.

NOTES:

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

The information in this document is believed to be accurate in all respects at the time of publication but is subject to change without notice. Silicon Laboratories assumes no responsibility for errors and omissions, and disclaims responsibility for any consequences resulting from the use of information included herein. Additionally, Silicon Laboratories assumes no responsibility for the functioning of undescribed features or parameters. Silicon Laboratories reserves the right to make changes without further notice. Silicon Laboratories makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Silicon Laboratories assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. Silicon Laboratories products are not designed, intended, or authorized for use in applications intended to support or sustain life, or for any other application in which the failure of the Silicon Laboratories product could create a situation where personal injury or death may occur. Should Buyer purchase or use Silicon Laboratories products for any such unintended or unauthorized application, Buyer shall indemnify and hold Silicon Laboratories harmless against all claims and damages.
The sale of this product contains no licenses to Power-One's intellectual property. Contact Power-One, Inc. for appropriate licenses.

Silicon Laboratories and Silicon Labs are trademarks of Silicon Laboratories Inc.

Other products or brandnames mentioned herein are trademarks or registered trademarks of their respective holders.