



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



CLOCKBUILDERPRO™ FIELD PROGRAMMER KIT

1. Introduction and Scope

This document describes how to use the Si538x/4x ClockBuilder Field Programmer Kit (“CBPROG-DONGLE”) with [ClockBuilder Pro](#) (“CBPro”) to support four programming models. Refer to the text and table below for supported uses:

1. **In-socket NVM Programming:** NVM programming of “**base**” Si538x/4x devices (e.g., Si5341A-A-GM), or any other factory “**pre-programmed**” Si538x/4x device (e.g., Si5341A-A12345-GM) which has unused NVM banks. Silicon Labs provides 44-pin and 64-pin QFN socket adapter boards for this purpose.
2. **In-system NVM Programming:** NVM programming of Si538x/4x devices already mounted on a system PCB. Users are encouraged to include a standard 10-pin header on their PCB to allow the Si538x4x Field Programmer board and ribbon cable to easily connect to the USB to SPI/I2C adapter.
3. **In-system Volatile Register Programming:** of devices mounted on a PCB (e.g., use the Wizard and EVB GUIs to inspect status registers, make volatile configuration updates, debug system firmware, etc.).
4. **In-socket Volatile Register Programming:** of devices mounted in the socket (e.g., use the Wizard and EVB GUIs to inspect status registers, make volatile configuration updates, debug system firmware, etc.).

Location of Target Si538x/4x Device	Software Utility and Programming Model Supported	
	NVM Burn Tool	EVB GUI / Wizard
In-socket	✓ (1)	✓ (4)
In-system	✓ (2)	✓ (3)

Figure 1. Supported Programming Models

2. Kit Contents

Shown below is a diagram of how the various components in the Field Programmer kit are connected to one of the QFN socket adapter boards, or to a PCB for in-system programming.



Figure 2. Example Hardware Configuration (Using QFN Socket Board or Customer PCB)

CBPro-DONGLE UG

Figure 3 shows the kit contents for the CBPROG-DONGLE kit. Note that the 44-pin and 64-pin sockets are sold separately as part numbers “Si538X4X-44SKT” and “Si538X4X-64SKT”, respectively. The Clock Builder Pro Field Programmer resources including schematics, layout files, and BOM can be found at www.silabs.com/CBProgrammer. Note that the sockets are sold as separate kits.

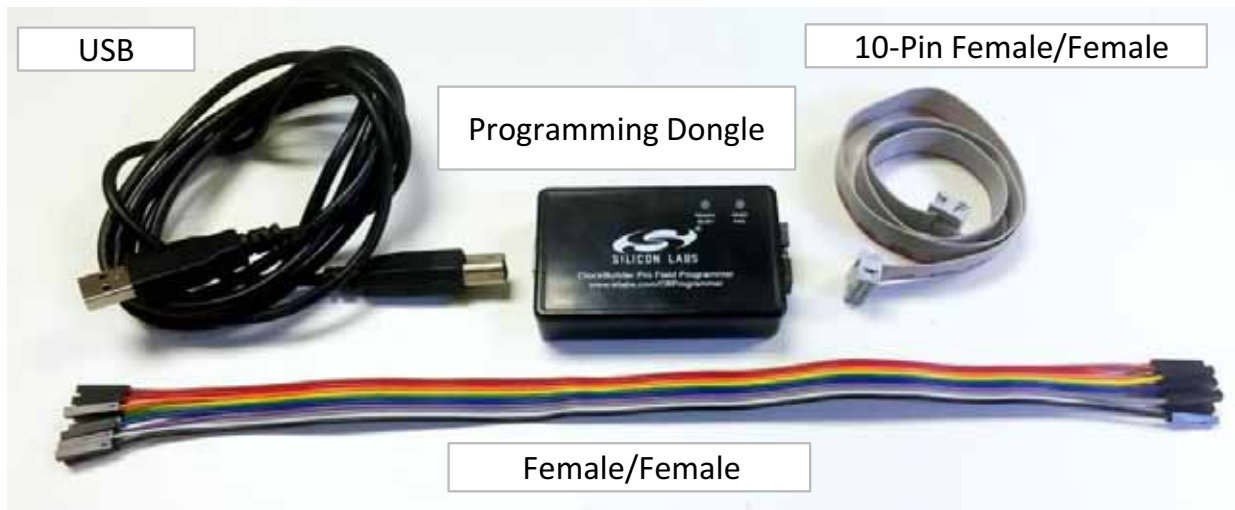


Figure 3. CBPROG-DONGLE Kit Contents



Figure 4. Si538x4x-44SKT-DK (l) and Si538x4x-64SKT-DK (r) Sockets Sold Separately

3. Software Download and Installation

To install the CBPro software on any **Windows 7** (or above) PC, go:

<http://www.silabs.com/CBPro> and download the ZIP file to install the software on your host PC.

4. Hardware Configuration

The Field Programmer Dongle acts as an interface between the CBPro GUI and the target device (any supported Si534x or Si538x IC). Connect the provided USB cable to your PC and the CBPROG-DONGLE. The CBPROG-DONGLE is then connected to the target device using the provided cables or a programming socket, depending upon the four ways you may use the programmer as detailed in “5. Ways You can Use the Programmer”.

5. Ways You can Use the Programmer

The following four sections describe four ways you can use the CBPROG-DONGLE.

5.1. In-Socket NVM Programming

This workflow describes the process of programming or “burning” the non-volatile memory (NVM) of loose devices using the Si538X4X-44SKT or Si538X4X-64SKT boards that feature a programming socket. Once completed, this flow will “burn” a complete configuration from CBPro into one of the banks of NVM on the device. Devices shipped from Silicon Labs have two NVM banks available to program (“burn”).

The steps needed to program a device’s NVM are as follows:

1. Assuming the CBPro software installs without error, connect the CBPROG-DONGLE adapter with the USB cable to the PC on which CBPro was installed. Use the USB extender cable (provided with the kit) if your host PC is located far from the CBPROG-DONGLE.



Figure 5. PC to CBPROG-DONGLE Connection

2. Insert a base or previously pre-programmed (e.g. OPN) Si538x/4x device into the QFN socket.

Socket and DUT Orientation: It is important to ensure the DUT is in the correct orientation before powering up the board. If not orientated correctly the software has a feature to auto-detect it is not able to read the part. Likely the reason is there is no part in the socket or it is oriented incorrectly. The part will not be damaged if oriented incorrectly. The DUT has two circles on the part. The smaller circle is the pin 1 indicator. Pin 1 on the socket is lined up with the U1 and dot symbol on the silk screen.

CBPro-DONGLE UG



Figure 6. Correct Orientation of a DUT in the QFN Socket

Note: Power is not applied to the socket's VDD and VDDA pins unless explicit action by you within CBPro. It is safe to:

1. Insert or remove a DUT in the socket before or after the socket has been connected to the main board.
2. Insert or remove a DUT in the socket before or after power has been applied to the main board by connecting the USB cable to your PC.

As will be discussed in the walk-through that follows, power is only applied to the DUT when you explicitly scan for a DUT or initiate a burn. Power is automatically off at all other times.

3. Connect the QFN Field Programmer Socket Board with the DUT into the CBPROG-DONGLE. Use the male-to-female ribbon cable if needed.



Figure 7. System from PC to Programming CBPROG-DONGLE Board to QFN Field Programmer Socket Board

4. Start ClockBuilder Pro by locating the icon (see Figure 8) on your desktop or Windows Start Menu.



Figure 8. ClockBuilder Pro Icon

5. The ClockBuilder Pro Wizard main menu should now appear as shown in Figure 9. Select the “NVM Burn Tool” as shown. **Do not select EVB GUI.**

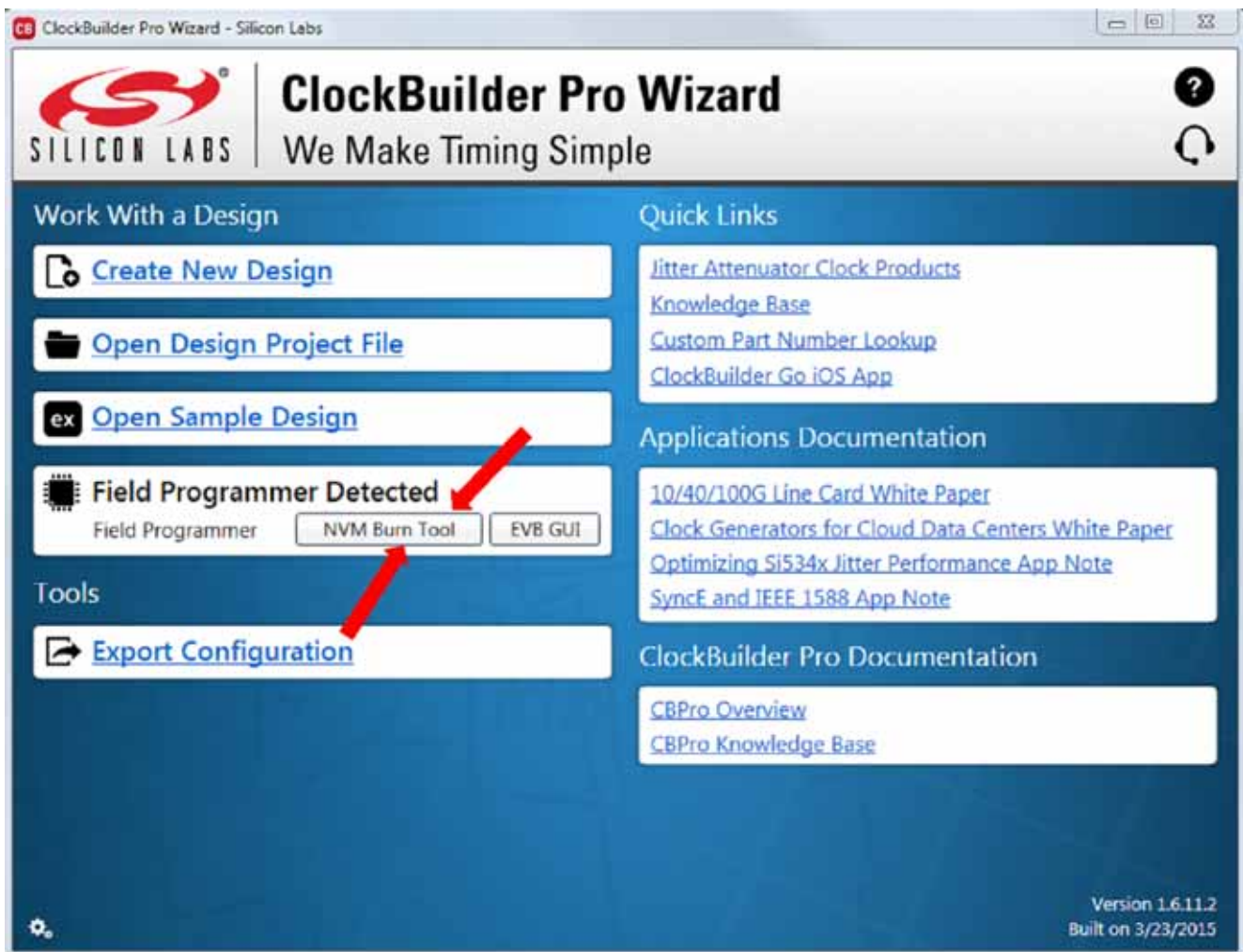


Figure 9. ClockBuilder Pro Wizard

CBPro-DONGLE UG

6. In the **Burn NVM** tool window:

a. Step 1: Make sure “**Kit Field Programmer Socket Board**” is selected.

b. Step 2: Select “**Scan**”.

If the device is properly orientated in the socket, the middle section of the window will show information about the detected device, including how many NVM banks are available to burn. In socket mode, the CBPROG-DONGLE always communicates in 4-wire SPI host interface mode.

c. Step 3: Click on “**Select Project File**”.

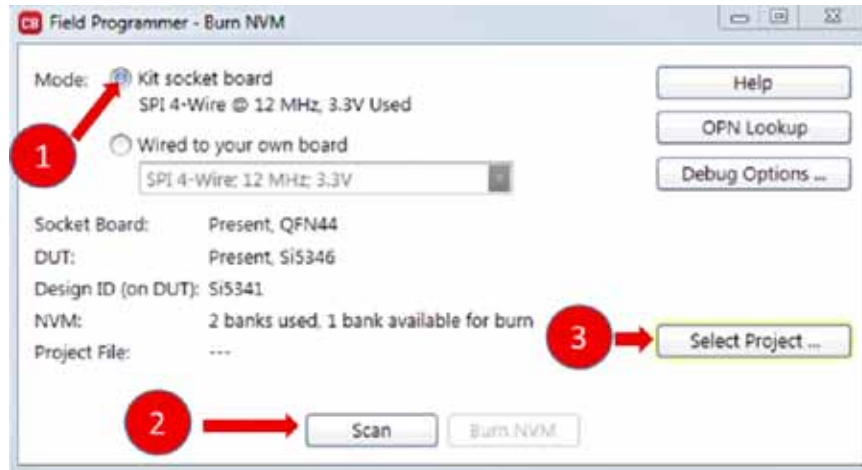


Figure 10. Field Programmer Screen

d. Step 4: Navigate to and select your CBPro project file in the Windows file browser:

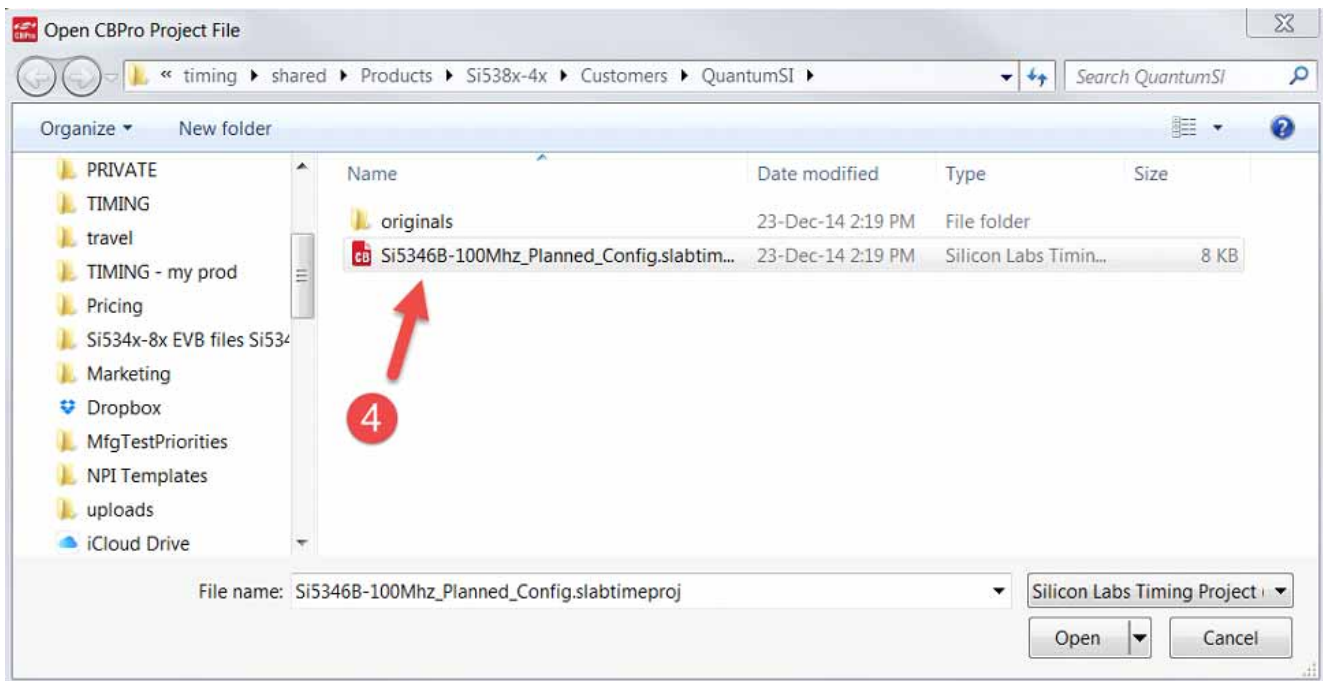


Figure 11. Navigating to CBPro Project File

e. Step 5: After selecting the project file, select “**Burn NVM**” as shown in Figure 12.

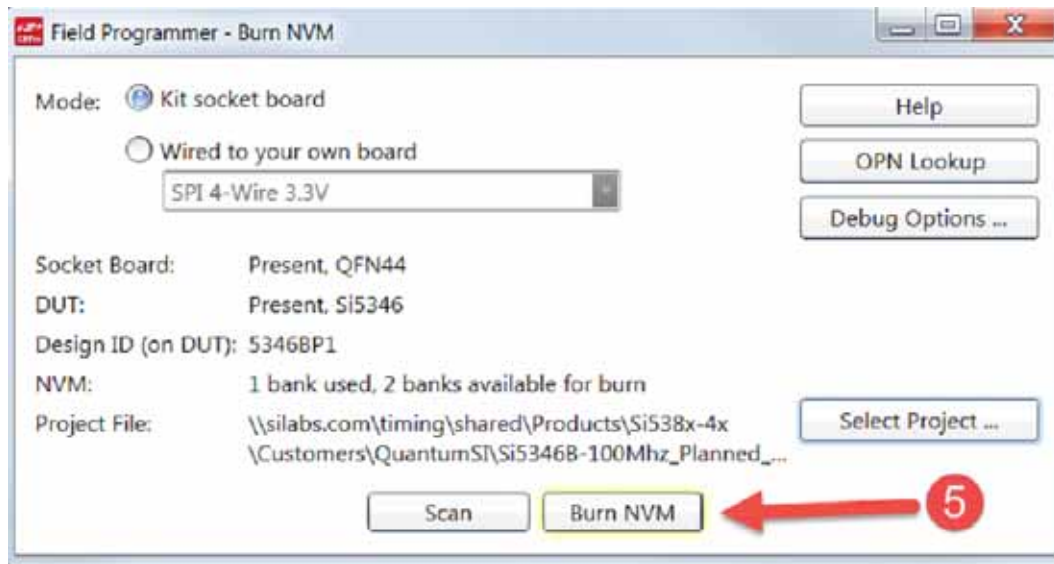


Figure 12. Click on Burn NVM

CBPro-DONGLE UG

f. A series of status windows will appear as shown if the target DUT in the socket is successfully “burned”.

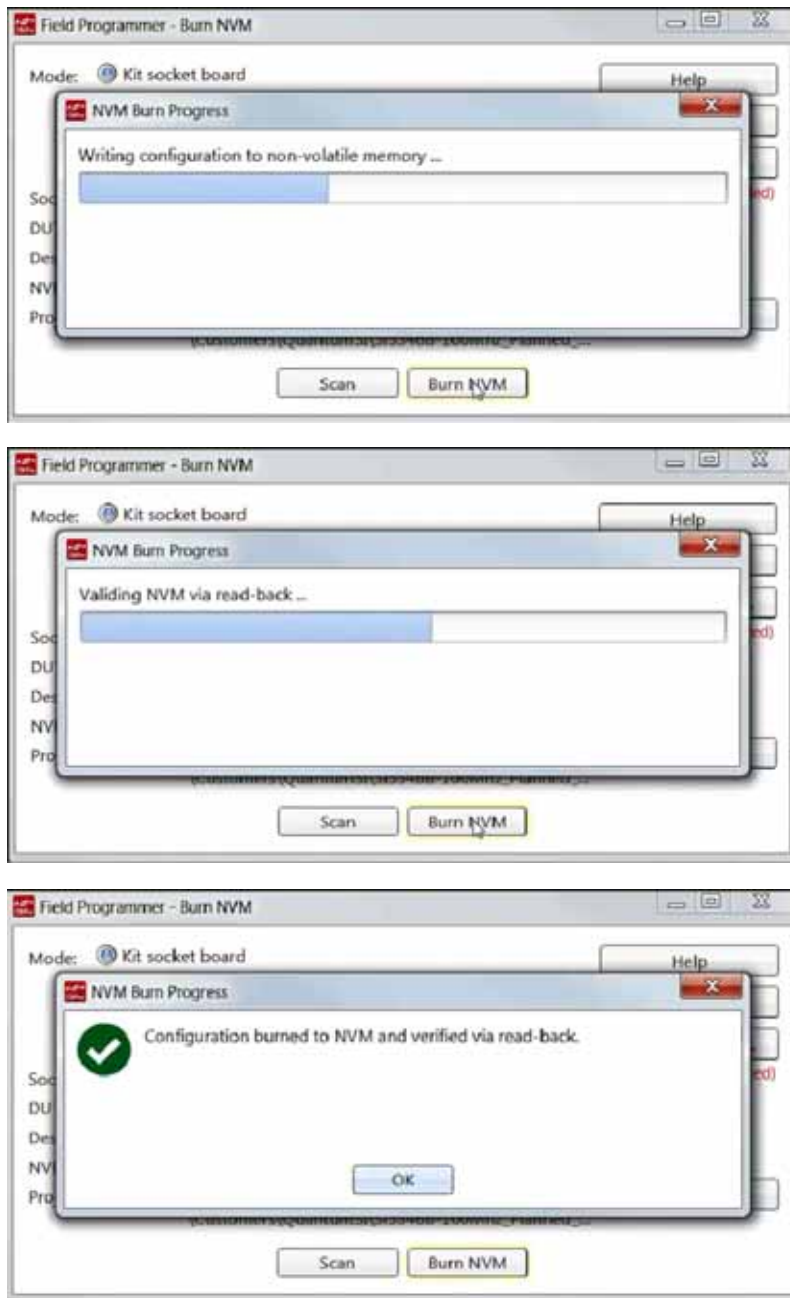


Figure 13. Status Windows

When you click OK, the DUT will be rescanned and you should see the number of banks available to burn decrease by 1.

Note that you may remove the newly programmed DUT at any time: power to the DUT is turned off immediately after scan and burn operations complete.

5.2. In-System NVM Programming

This workflow describes the process of programming or “burning” the non-volatile memory (NVMN) of a device mounted on a PCB, and your goal is to reprogram a device’s NVM with a different configuration from a CBPro project file. Once completed, this flow will “burn” a complete configuration from CBPro into one of the banks of NVM on the device, assuming an open NVM bank is available. Devices shipped from Silicon Labs always have to 2 NVM banks available to program (“burn”). If you don’t know how many banks are still open to burn on your target device, have no fear, as the kit’s software automatically detects and reports the number of remaining NVM banks.

The steps needed to program an “in-system” device’s NVM are as follows:

1. Assuming the CBPro software installs without error (see “3. Software Download and Installation” on page 3), connect the adapter (CBPROG-DONGLE) board with the USB cable to the PC on which CBPro was installed. Use the USB extender cable (provided with the kit) if your host PC is located far from the CBPROG-DONGLE.

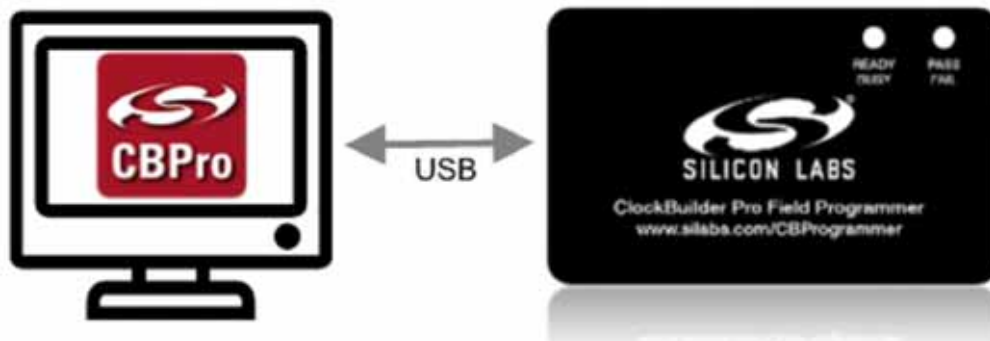


Figure 14. PC to CBPROG-DONGLE Connection

2. Lookup and verify the **host I/O mode (I2C or SPI)**, the **I2C address**, and the interface **I/O voltage level** compatibility of your host’s I/O voltage (for I2C or SPI) and the DUT’s.

The value set at the DUT’s register address of 0x09043 determines how the I/O supply voltages must be configured to communicate reliably with the CBPROG-DONGLE. You can look up your DUT’s host I/O voltage using the “**OPN Lookup**” option in the NVM Burn tool as shown in Figure 15.

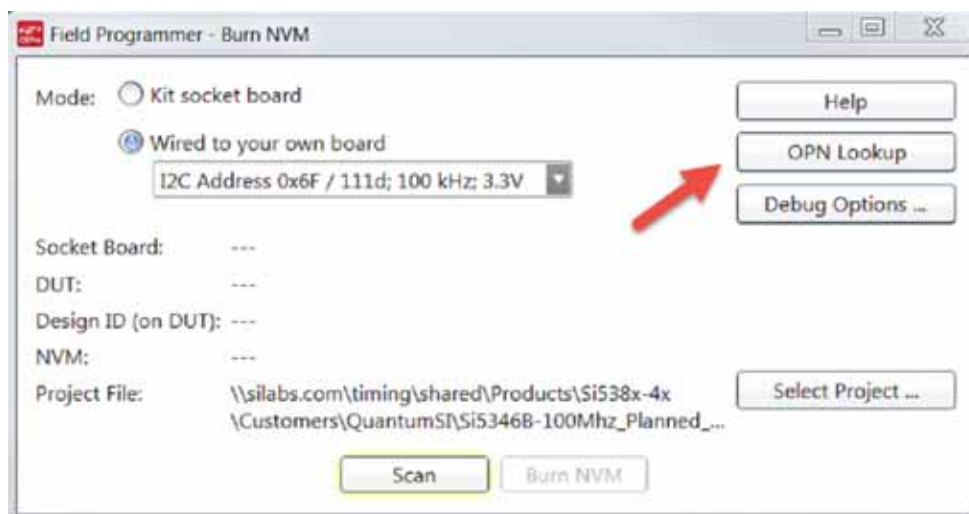


Figure 15. OPN Lookup Option

CBPro-DONGLE UG

If you have a custom OPN mounted on your board (a part number with a 5 digit code in the middle of the part number, such as Si5346B-A03260-GM), you should look up the host I/O setting (located at address of 0x09043) by selecting the **OPN Lookup** option. A browser will open and you will then enter in your custom OPN, as shown below.

- a. Select “**Clock or Buffer**”.
- b. Enter in your full ordering part number (OPN). E.g., Si5346B-A03260-GM.
- c. Click the blue arrow to lookup your OPN to verify the host I/O voltage setting of your DUT.

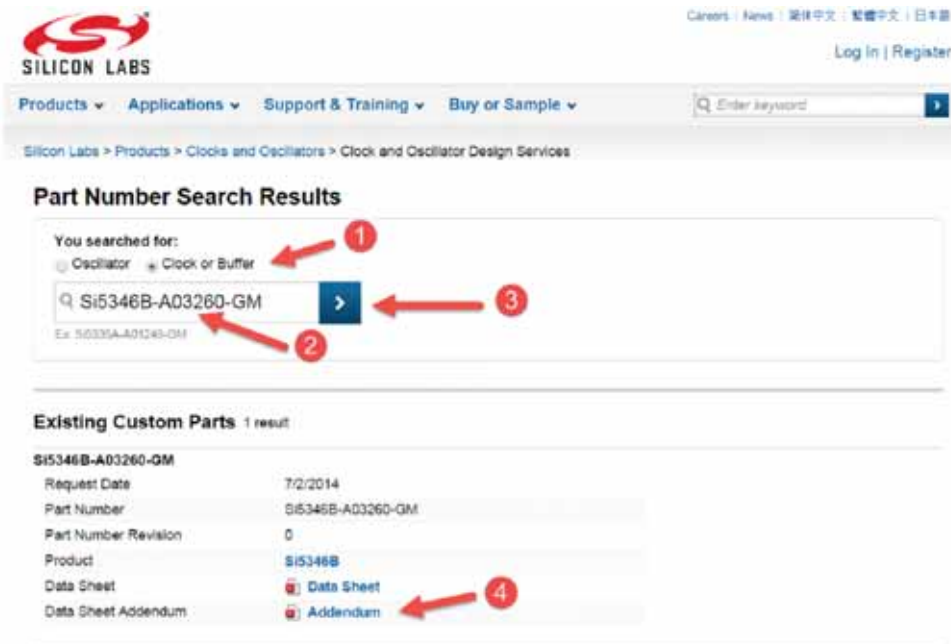



Figure 16. OPN Lookup

- When the utility displays the OPN's files, click on Addendum to verify the I/O Power Supply setting of your DUT in the Data Sheet Addendum.
"VDD Core" indicates the I/O supply for the I2C/SPI interface will operate from a 1.8 V supply.
"VDDA 3.3 V" indicates the I/O supply for the I2C/SPI interface will operate from a 3.3 V supply.



Si5346B Datasheet Addendum
Device Configuration Summary for Si5346B-A03260-GM
Page 1 of 2

4-Output Dual DSPLL Any-Frequency Jitter Attenuating Clock Multiplier

Overview

Part: Si5346
Created By: ClockBuilder Pro v0.100 [2014-07-01]
Timestamp: 02-Jul-14 5:46 PM

Design Rule Check

Errors:
- No errors

Warnings:
- OUT0 is disabled; consider setting to Unused to conserve power

Device Grade

Maximum output frequency: 25 MHz
Frequency synthesis mode: Fractional
Minimum Base OPN: Si5346B
Actual Base OPN: Si5346B

Base OPN Grade	Output Clock Frequency Range
Si5346A	100 Hz to 800 MHz
Si5346B	100 Hz to 350 MHz

Design

Host Interface:
I/O Power Supply: VDD (Core)
SPI Mode: 4-Wire
I2C Address Range: 108d to 111d / 0x6C to 0x6F (selected via A0/A1 pins)




Figure 17. Finding the I/O Power Supply Type

Figure 18 shows an example data sheet addendum showing VDDA (3.3 V).

Design

Host Interface:
I/O Power Supply: VDDA (3.3V)
SPI Mode: 4-Wire
I2C Address Range: 116d to 119d / 0x74 to 0x77 (selected via A0/A1 pins)




Figure 18. Finding VDDA Value

- Connect/wire the pins of the CBPROG-DONGLE to your host system with the target Si538x4x device. Use the male-to-female ribbon cable to connect to your host board fitted with a standard 10-pin header. This assumes you included the 10-pin header on your PCB and followed the recommended pinout and connections to the target Si438x/4x on your PCB. Note the pinout diagram and descriptions in the table below.

CBPro-DONGLE UG



Figure 19. Interface Pins on Header (Front View of CBPROG-DONGLE)

Table 1. Interface Pin Connections from CBPROG-DONGLE

Pin #	Description	Wire to Your PCB?	I ² C	4-wire SPI	3-wire SPI
9	A0_CS	3- or 4-Wire SPI	Can be used to set I ² C address bit A0 high or low. Routed to A0 DUT pin on the programming Field Programmer Socket Boards.	Drives the chip select signal during SPI transactions	Drives the chip select signal during SPI transactions
10	VDD	Never	Supplies the Core VDD voltage to the DUT when using a programming Field Programmer Socket Board. Do not use this pin for in-system programming.	Supplies the Core VDD voltage to the DUT when using a programming Field Programmer Socket Board. Do not use this pin for in-system programming.	Supplies the core VDD voltage to the DUT when using a programming Field Programmer Socket Board. Do not use this pin for in-system programming.
7	SDA_SDIO	Always	Serial data signal for I ² C transactions.	Serial data out to DUT for 4-wire SPI transactions (MOSI).	Bidirectional Serial data for 3-wire SPI transactions (SDIO).
8	I2C_SEL1	Never	Used to set I2C_SEL signal high to set the DUT for I ² C communication. (Refer to specific part pinout and the programming Field Programmer Socket Board to determine whether to use I2C_SEL1 or I2C_SEL2)	Used to put I2C_-SEL signal low for SPI communication. (Refer to specific part pinout and the programming Field Programmer Socket Board to determine whether to use I2C_-SEL1 or I2C_SEL2)	Used to put I2C_-SEL signal low for SPI communication. (Refer to specific part pinout and the programming Field Programmer Socket Board to determine whether to use I2C_-SEL1 or I2C_SEL2)

Table 1. Interface Pin Connections from CBPROG-DONGLE (Continued)

Pin #	Description	Wire to Your PCB?	I ² C	4-wire SPI	3-wire SPI
5	A1_SDO	4-Wire SPI Only	Can be used to set I2C address bit A1 high or low. Routed to A1 DUT pin on the programming Field Programmer Socket Boards.	Serial data from DUT for 4-wire SPI transactions (MISO).	Not used
6	I2C_SEL2	Never	Used to set I2C_SEL signal high to set the DUT for I2C communication. (Refer to specific part pinout and the programming Field Programmer Socket Board to determine whether to use I2C_SEL1 or I2C_SEL2)	Used to put I2C_SEL signal low for SPI communication. (Refer to specific part pinout and the programming Field Programmer Socket Board to determine whether to use I2C_SEL1 or I2C_SEL2)	Used to put I2C_SEL signal low for SPI communication. (Refer to specific part pinout and the programming Field Programmer Socket Board to determine whether to use I2C_SEL1 or I2C_SEL2)
3	SCLK	Always	Serial clock signal for I2C transactions.	Serial clock signal for SPI transactions.	Serial clock signal for SPI transactions.
4	VDDA_VDDS	Never	Supplies the VDDA and VDDS voltages to the DUT when using a programming Field Programmer Socket Board. Do not use this pin for in-system programming.	Supplies the VDDA and VDDS voltages to the DUT when using a programming Field Programmer Socket Board. Do not use this pin for in-system programming.	Supplies the VDDA and VDDS voltages to the DUT when using a programming Field Programmer Socket Board. Do not use this pin for in-system programming.
1	GND	Always	GND	GND	GND
2	ID	Never	The programming Field Programmer Socket Boards provide a voltage on this pin to identify the board. For in-system programming, this pin should be grounded or not connected to any signal.	The programming Field Programmer Socket Boards provide a voltage on this pin to identify the board. For in-system programming, this pin should be grounded or not connected to any signal.	The programming Field Programmer Socket Boards provide a voltage on this pin to identify the board. For in-system programming, this pin should be grounded or not connected to any signal.

CBPro-DONGLE UG

5.2.1. I²C Hardware Configuration

For I²C Communication connecting to an external DUT board, the following pins should be used from the:

CBPROG-DONGLE pins

- Pin 3: Serial Clock SCLK
- Pin 7: Serial Data SDA
- Pin 1: Ground

DUT pins

- A0/CS: Drive this pin high or low to set the I²C Address.
- A1/SDO: Drive this pin high or low to set the I²C Address.
- VDDA: The DUT should be externally powered. VDDA must be set to 3.3 V.
- VDD: The core voltage must be set to 1.8 V.

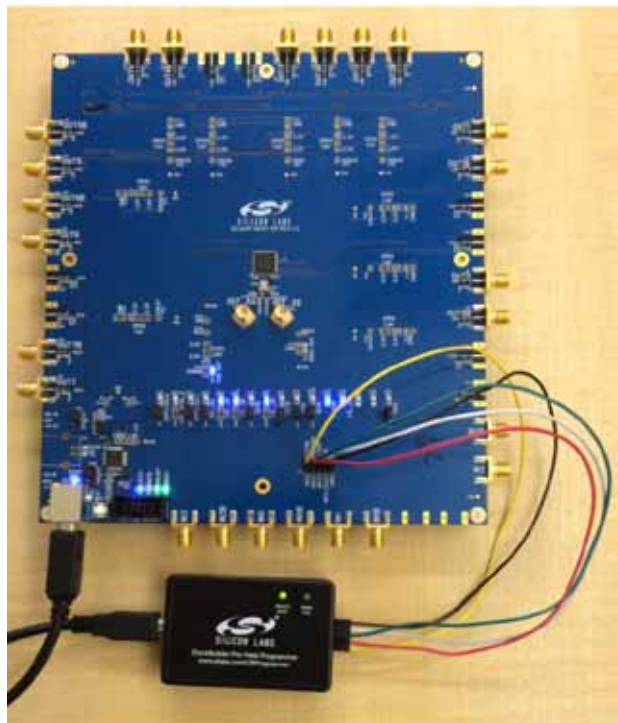


Figure 20. Example I2C Connection to External System Target Board Using Jumper Wires (Si5346-EVB)

When using SPI Communication with long wires as shown above it is advisable to use 6 Mb/s bus speed or less.

5.2.2. SPI 3-Wire Hardware Configuration

For 3-wire SPI communication, when connecting to an external DUT board, the following pins should be used from the MCU CBPROG-DONGLE.

CBPROG-DONGLE pins

- Pin 3: Serial Clock SCLK
- Pin 7: Serial Data SDIO for Data In and Out
- Pin 9: A0_CS for Chip Select
- Pin 1: Ground

DUT PINS

- VDDA: The DUT should be externally powered. VDDA must be set to 3.3 V.
- VDD: The core voltage must be set to 1.8 V.

5.2.3. SPI 4-Wire Hardware Configuration

For 4-wire SPI communication, when connecting to an external DUT board, the following pins should be used from the MCU CBPROG-DONGLE.

CBPROG-DONGLE pins

- Pin 3: Serial Clock SCLK
- Pin 7: Serial Data SDIO for Data In to DUT (MOSI)
- Pin 5: A1_SDO for Data Out of DUT (MISO)
- Pin 9: A0_CS for Chip Select
- Pin 1: Ground

DUT PINS

- VDDA: The DUT should be externally powered. VDDA must be set to 3.3 V.
- VDD: The core voltage must be set to 1.8 V.

5. After verifying the CBPROG-DONGLE to DUT connections as noted above, execute the following steps in the “**Burn NVM**” menu. This example assumes a DUT is configured with the host I²C interface operating in 3.3 V I/O mode with an I²C address of 0x6F, and an I²C bus speed of 100 kHz.

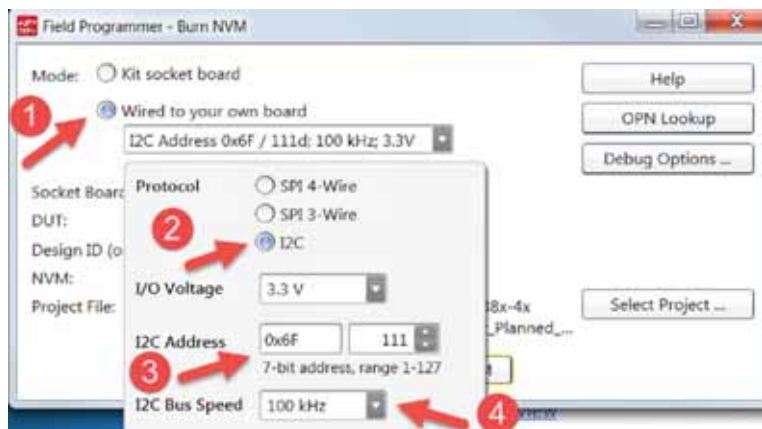


Figure 21. Selecting the I²C Configuration Options

Note: 100 kHz I²C bus speed is recommended to avoid programming errors. The CBPro burn tool will always read back the programmed NVM to verify the DUT was programmed correctly.

CBPro-DONGLE UG

6. After entering in the fields noted above, select “**Scan**” to initiate detection of your target DUT.
7. Assuming your DUT was properly detected, the **Burn NVM** window will appear again, showing the presence of the DUT, its DESIGN_ID value, and the # of NVM banks available (in this case, there are no more NVM banks available to burn).

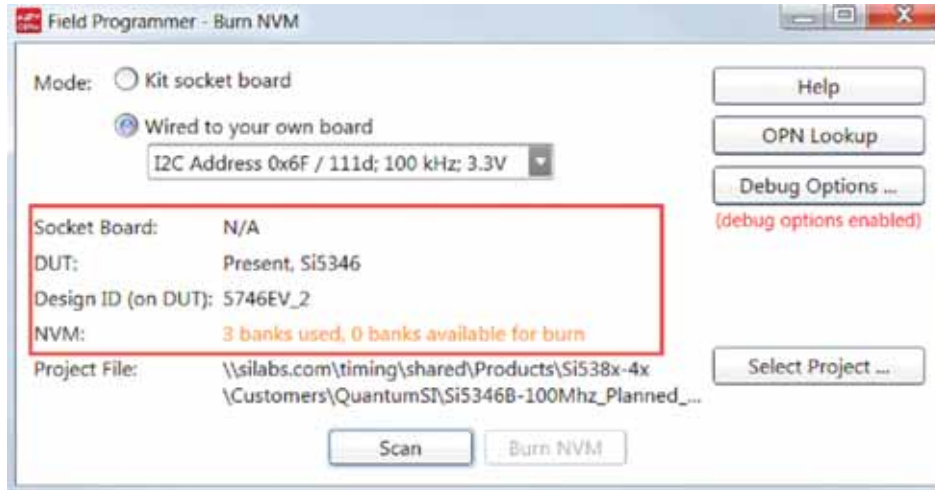


Figure 22. Board Information and NVM Burn Banks Available

8. Navigate to and select your CBPro project file in the Windows file browser.

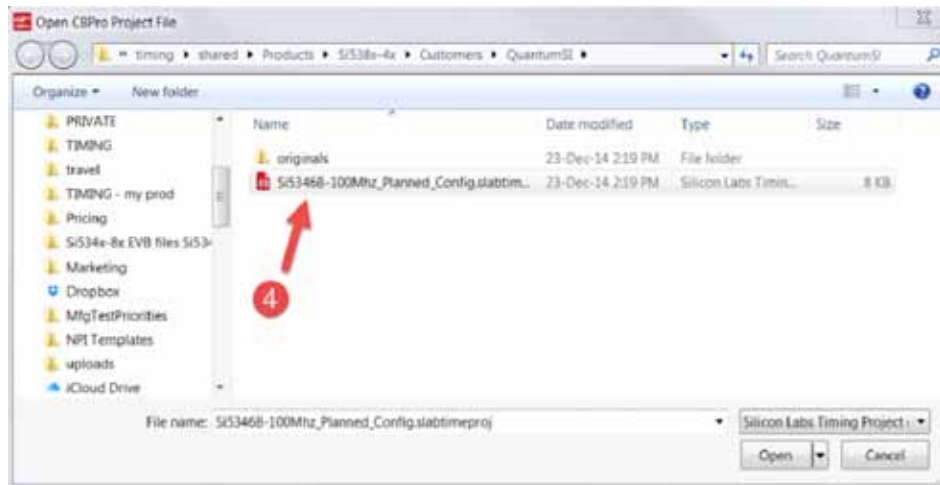


Figure 23. Navigating to CBPro Project File

9. After selecting the project file, select “**Burn NVM**” as shown:



Figure 24. Select Burn NVM if Fewer than 3 Banks Burned

5.3. In-System Volatile Register Programming and Register Debug

This workflow allows users to use the full CBPro configuration Wizard and EVB GUI to make volatile changes to a device's configuration and inspect the state of various status registers. There are two ways you can interact with your PCB-based DUT using the field programmer:

- Use CBPro Wizard to edit your device configuration, and write out changes directly to your DUT.
- Launch the EVB GUI, to inspect registers

All of the relevant CBPro features available when working with a Silicon Labs EVB will be available to you, with these exceptions:

- There is no voltage regulator control or voltage/current readings of any kind
- You must configure the host interface settings so that CBPro can use the DUT correct communication scheme/wire out.
- If you write out your design/project file, all registers configured via the “Host Interface” section of the wizard are written to the DUT (these registers are skipped when writing a design to a Silicon Labs EVB).

5.3.1. Using the CBPro Wizard with Your PCB-Based DUT

When you launch the CBPro wizard, instead of clicking the NVM Burn Tool, open your existing project file:

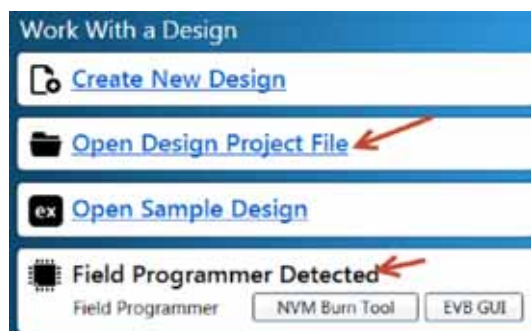


Figure 25. Open Design Project File, and see Field Programmer Detected

CBPro-DONGLE UG

In the design dashboard, you will see a pulldown to configure the host interface between the CBPROG-DONGLE and your PCB:



Figure 26. Pull Down Menu for Communication Options

Click the pulldown and configure the interface:

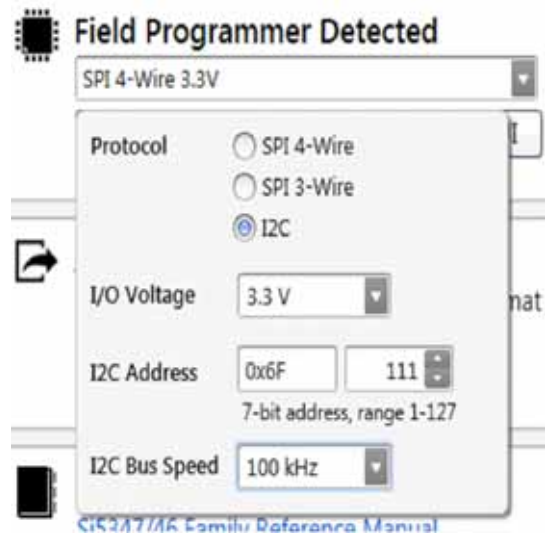


Figure 27. Communication Configuration Window

Once configured, you can write out your design to DUT using the button on the dashboard:



Or on any configuration page in the wizard:



5.3.2. Using the EVB GUI with your PCB-Based DUT

You launch the EVB GUI that same way you would when working with a Silicon Labs EVB:

1. From the CBPro wizard home page:



2. From the CBPro wizard design dashboard (you have opened a project file):



3. From the EVB GUI shortcut on the desktop or Windows Start Menu.

Similar to the dashboard and burn tool, you must configure the host interface options. These are available in a panel in the upper right hand corner of the EVB GUI window.

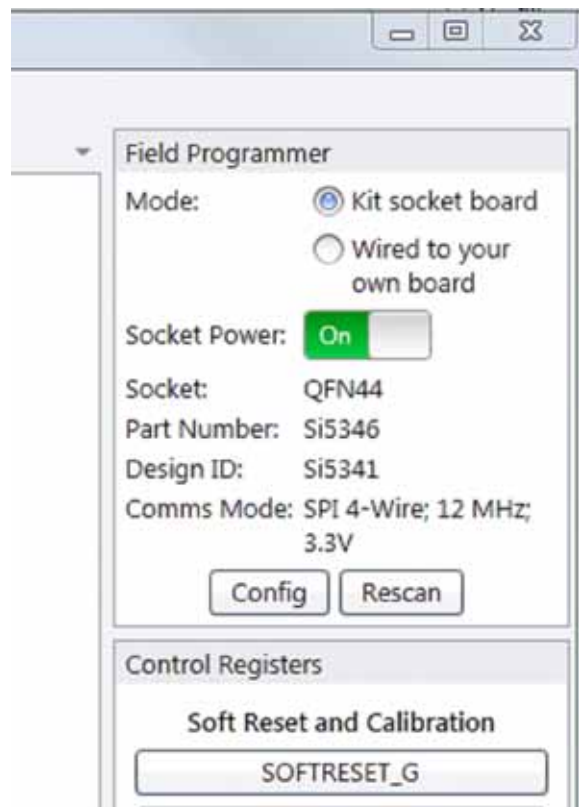


Figure 28. Socket or In-System Programming Options

CBPro-DONGLE UG

First select either the Kit socket board if the socket is attached. Otherwise select “Wired to your own board”.

Click the Config button to modify the host interface options.

Once configured, you can then turn on the socket power to do the following:

- Write a project file to the device (File menu)
- Inspect DUT status registers
- Peek/poke arbitrary DUT registers

Click the “Rescan” button to update the configuration options to the last command selected.

5.3.3. Socket Detection in EVB GUI

Putting the device into the socket in the wrong orientation will not ruin the device, and the software does detect an orientation error. The first two images on the left show wrong orientation. The image on the right shows the proper orientation. The software automatically keeps the power off if it detects that the device is in an orientation that could damage the chip and won't allow the socket power to be turned on.

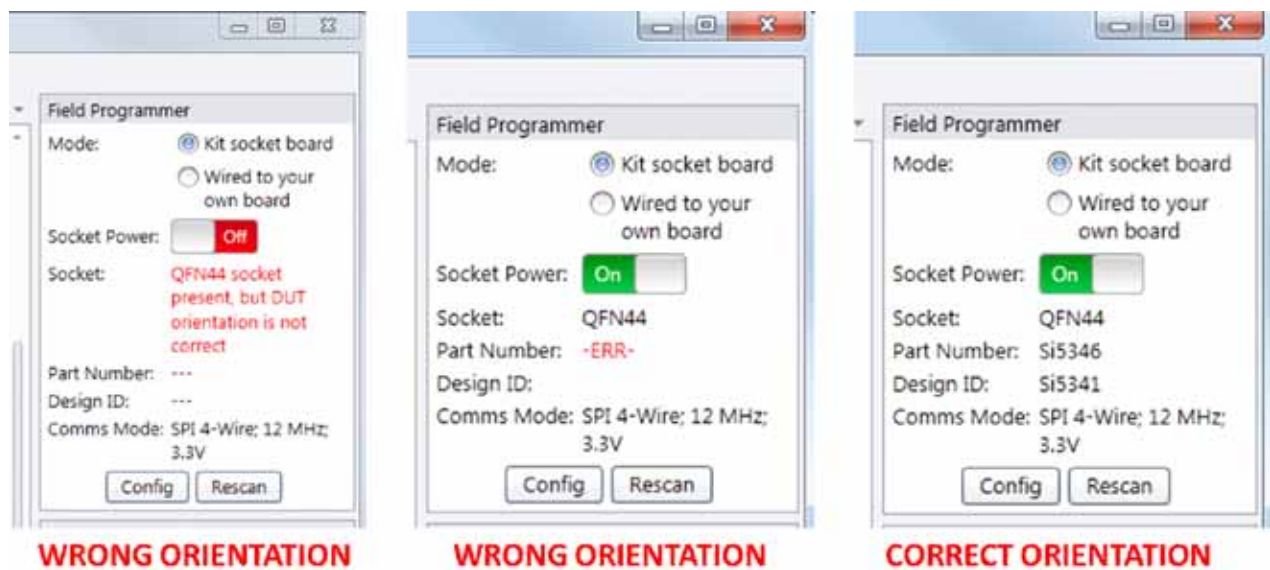


Figure 29. Socket Detection Results with Part in Different Orientations

It is advised to review the socket orientation and chip to ensure everything is correct. Make sure that the smaller dot on the chip is lined up in the corner where the U1 is located on the socket as shown below.



Figure 30. Correct Orientation of Device in Socket

5.4. In-Socket Volatile Register Programming and Register Debug

This workflow allows users to use the full CBPro configuration Wizard and EVB GUI to make volatile changes to a device's configuration and inspect the state of various status registers. There are two ways you can interact with your PCB-based DUT using the field programmer:

- Use CBPro Wizard to edit your device configuration, and write out changes directly to your DUT.
- Launch the EVB GUI, to inspect registers

All of the relevant CBPro features available when working with a Silicon Labs EVB will be available to you, without any exceptions. The details of this workflow are the same as above. The IO voltage is set to 3.3 V within the hardware appropriately, and does not require any external configuration.

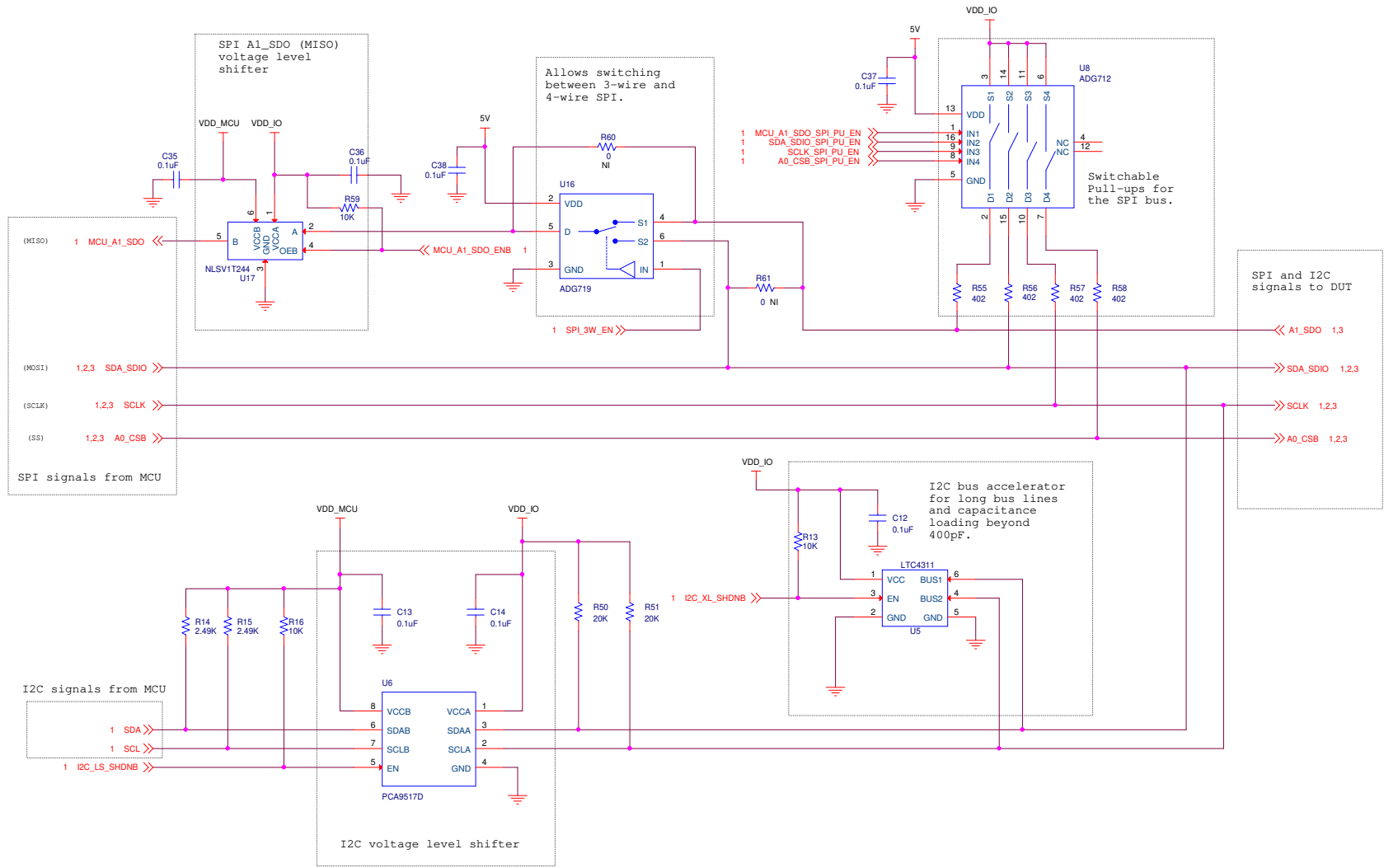


Figure 32. CBPROG-DONGLE Schematic (2 of 3)

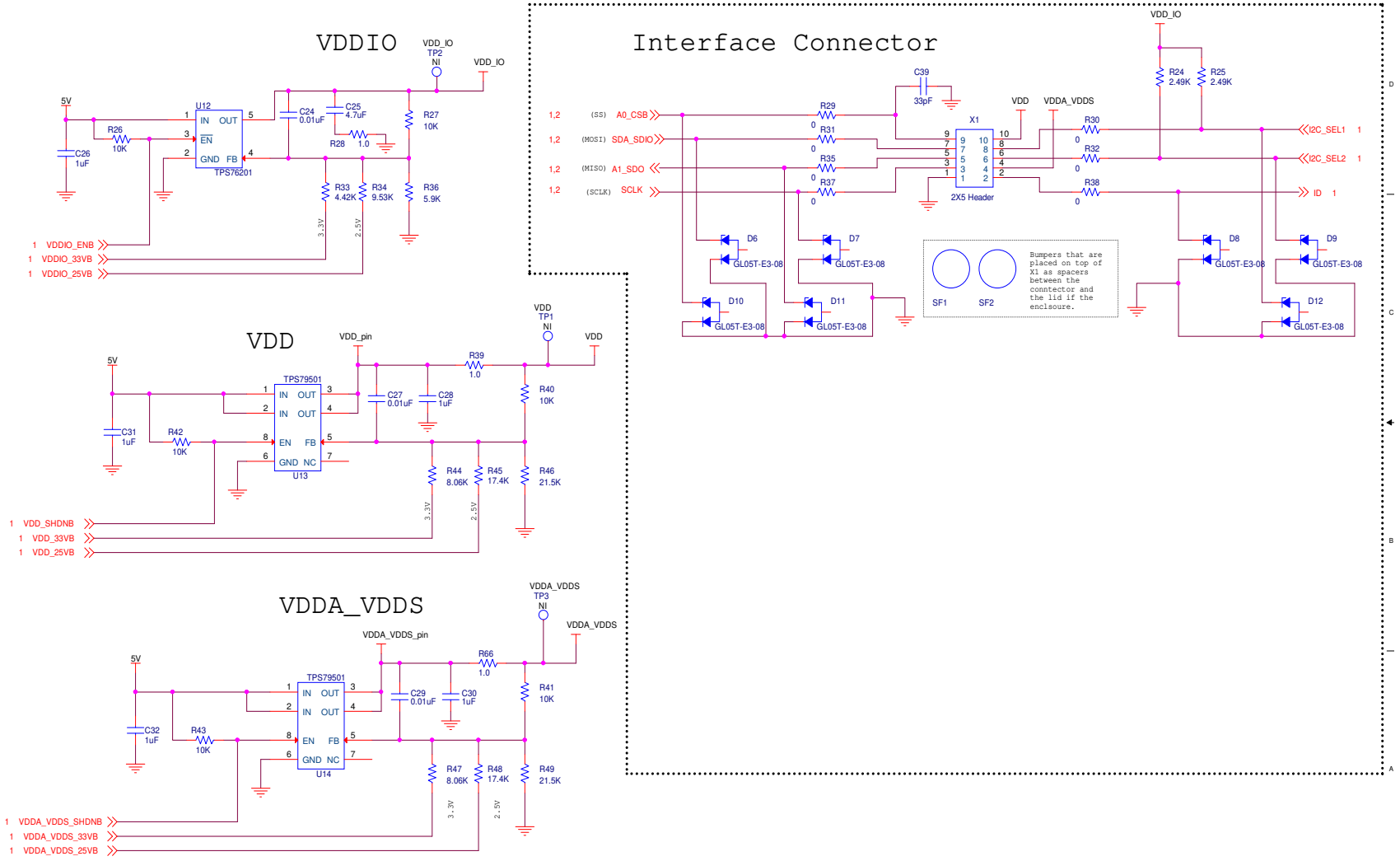


Figure 33. CBPROG-DONGLE Schematic (3 of 3)

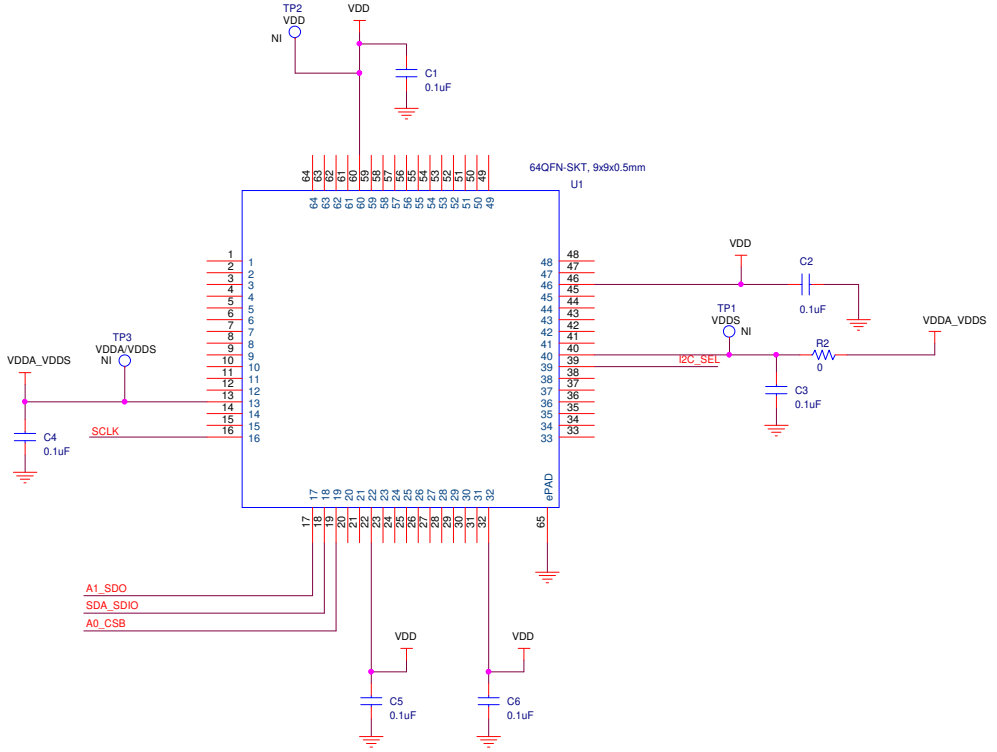
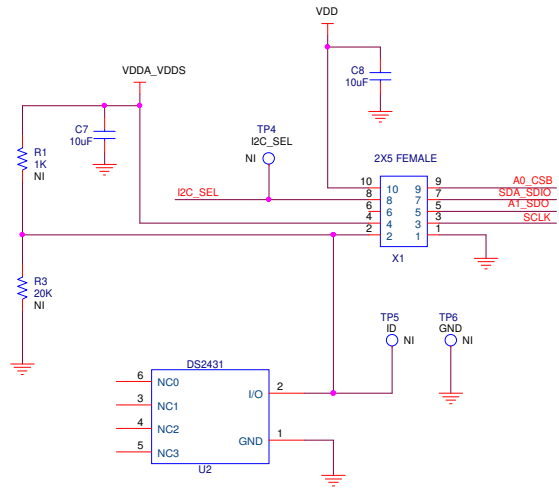


Figure 34. 64-Pin Socket Board Schematic