

# **UG286: ClockBuilder Pro Field Programmer Kit**

This user guide describes how to use the ClockBuilder Field Programmer Kit (CBPROG-DONGLE) with ClockBuilder Pro (CBPro) to support several programming models.

The ClockBuilder Pro Field Programmer supports a variety of uses:

- 1. Real Time Status Monitoring and Device Debug
- Read status and telemetry registers and write control registers on Si5332/57, Si5338/56, Si5351, and Si534x/8x/9x devices.
- Call device API commands on Si536x, Si540x, and Si55xx devices. APIs are available for device status and control.
- 2. Non-Volatile Firmware/Configuration Programming
- Select devices support burning/flashing device configuration and firmware.
- Firmware can be flashed multiple times on Si5383/4/8/9.
- Firmware can be burned to Si536x, Si540x, and Si55xx devices based on available NVM space (non-PPS PLL configurations only).
- Configuration can be burned up to two times on non-firmware-based devices (Si534x/8x/9x).
- Configuration can be flashed multiple times on firmware-based Si5383/4/8/9.
- Configuration can be burned to Si536x, Si540x, and Si55xx based on available NVM space (non-PPS PLL configurations only).
- Configuration can be burned to Si5332/7 based on available NVM space.
- 3. Volatile Firmware/Configuration Programming
- All field programmer supported devices support writing device firmware or configuration volatily.
- CBPro projects are used to calculate configuration programming data and optionally firmware image to download. CBPro handles all pre- and post-programming tasks.
- Programming data can be written via EVB GUI and command line tools, allowing customers to prototype flow their host would use to reprogram a device in-system.
- Si5332/57, Si5338/56, Si5351, and Si534x/7x/8x/9x devices can be fully reconfigured via device register scripts that can be exported from CBPro.
- Si536x, Si540x, and Si55xx device firmware and configuration can be reprogrammed via firmware and configuration boot files that can be exported from CBPro.

Most operations can be performed by either:

- 1. Wiring the Field Programmer's serial interface directly to system board for "in-system" volatile or nonvolatile programming. Users are encouraged to include a standard 10-pin header on their PCB to allow the Field Programmer board and ribbon cable to easily connect to the USB to SPI/I<sup>2</sup>C adapter.
- Placing a loose Skyworks Timing device into a dedicated field programmer socket for "in-socket" non-volatile programming (volatile is also supported but generally not useful in this scenario except for testing). Skyworks provides 32-pin, 40-pin, 44-pin, 48-pin, 64-pin, and 72-pin QFN socket adapter boards, as well as 44-pin and 64-pin LGA socket adapter boards.

# **1. Features**

- Field Programmer kit contents
- CBPro download and installation instructions
- Hardware connections
- Usage examples for the Field Programmer
- Schematics of the Field Programmer and socket board
- Bill of materials
- Troubleshooting appendix for common issues
- Supported devices include Si5332/8, Si5350/1/6/7, Si534x/6x/8x/9x, Si540x, and Si55xx part families.

# 2. Kit Contents

Table 1 below shows the kit contents for the Field Programmer Kit. Several sockets are supported for monitoring or programming a loose device. The following table summarizes available sockets:

Supported Devices	Socket Part Number	Notes
Si5332	Si5332-32SKT-DK, Si5332-40SKT-DK, Si5332-48SKT-DK	The number of pins in the socket must match the device being programmed. The Si5332E/F/G/H embedded crystal products in 40-pin LGA and 48-pin LGA packages are currently not supported.
Si5338/56	N/A	Socket boards are available as a standalone programmer, P/N Si5338/56-PROG-EVB
Si5350/1/7	N/A	No sockets are available for this part.
Si534x/8x/9x	Si538x4x-44SKT-DK, Si538x4x-56SKT-DK, Si538x4x-64SKT-DK, Si538x4x-44LGA-SKT, and Si538x4x-64LGA-SKT	For QFN packages: Si5392/94 44-pin QFN devices require the Si538x4x-44SKT-DK. Si5395 64-pin QFN devices require the Si538x4x-64SKT-DK. For LGA packages: Si5392/94 44-pin LGA devices require the Si538x4x-44LGA-SKT. Si5395 64-pin LGA devices require the Si538x4x-64LGA-SKT.
Si536x, Si540x, Si55xx	Si55xx-72SKT-DK	The Si55xx-72SKT-DK requires an external 5V power supply.

#### **Table 1. Field Programmer Supported Devices**

The 32-pin, 40-pin, 44-pin, 48-pin, 56-pin, 64-pin, and 72-pin sockets pictured in the following pages are available separately as part numbers. The ClockBuilder Pro Field Programmer resources including schematics, layout files, and BOM can be found on the Skyworks ClockBuilder Pro Field Programmer page. Note that the sockets are sold as separate kits.



### Figure 1. Field Programmer Kit Contents

The diagram below shows how the Field Programmer kit is intended to be the bridge between a computer running ClockBuilder Pro software (CBPro) and the QFN socket adapter boards/customer PCB for in-system firmware and volatile programming.



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



Figure 3. Si5332-32SKT-DK



Figure 4. Si5332-40SKT-DK



Figure 5. Si538x4x-44SKT-DK



Figure 6. Si5332-48SKT-DK



Figure 7. Si538x4x-56SKT-DK



Figure 8. Si538x4x-64SKT-DK



Figure 9. Si538x4x-44LGA-SKT



Figure 10. Si538x4x-64LGA-SKT

Skyworks Solutions, Inc. • Phone [949] 231-3000 • sales@skyworksinc.com • www.skyworksinc.com 206749A • Skyworks Proprietary Information • Products and Product Information are Subject to Change without Notice



Figure 11. Si55xx-72SKT-DK

# 3. Software Download and Installation

To install the ClockBuilder Pro Software (CBPro) on any Windows 7 (or above) PC, visit the Skyworks ClockBuilder Pro Field Programmer page and download ClockBuilder Pro (CBPro) to install the software on your host PC. ClockBuilder Pro software allows the user to easily configure many of Skyworks' jitter attenuator and clock generator products, as well as directly write the configuration to the device over the Field Programmer with the click of a button.

# 4. Hardware Configuration

The Clockbuilder Pro Field Programmer acts as an interface between the CBPro GUI (software running on a computer) and the target device. Connect the provided USB cable to your computer and the Field Programmer. The Field Programmer is then connected to the target device using the provided jumper cables or a programming socket, depending upon the desired end configuration as detailed in the following section, "5. Ways You Can Use the Programmer" on page 9.

# 5. Ways You Can Use the Programmer

The following four sections describe four ways you can use the Field Programmer.

# 5.1. In-Socket Firmware/NVM Programming

This workflow describes the process of programming loose devices using the Si5332-32/40/48SKT, Si534x/8x-56SKT, Si534x/8x-64SKT, or Si55xx-72SKT programming socket board. For non-firmware-based solutions, this flow will "burn" a complete configuration from CBPro into available NVM in the device. Si534x/8x/9x devices shipped from Skyworks have two NVM banks available to program ("burn"). Si5332 devices have a flexible NVM space. Si536x, Si540x, and Si55xx devices may have NVM space in select configurations. CBPro manages available NVM and programs ("burns") the available NVM when feasible. For Si5383/4/8/9 (firmware based) devices, this flow will flash a complete configuration from CBPro in to the device.

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

1. Assuming the CBPro software is installed, connect the Field Programmer (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 Field Programmer.



Figure 12. Computer to Field Programmer Connection

2. Insert a base or previously preprogrammed (e.g., OPN) device into the socket.

Socket and device Orientation: It is important that the device is correctly oriented before powering up the board. However, if there is no device in the socket or if the device is not correctly orientated, the software will auto-detect the fault and will not connect to the part. The part will not be damaged if oriented incorrectly. The device 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 socket board silk screen. 64-QFN and 44-QFN orientations are shown below. The same idea applies to 32-QFN, 40-QFN, 48-QFN, 56-QFN and 72-QFN package ICs.



Figure 13. Correct Orientation of a Device in the Socket

- 3. Power is only applied to the device when you perform a scan or initiate a burn. Power is off at all other times. Power is not applied to the socketed device unless explicitly triggered by the user within CBPro. It is safe to:
- Insert or remove a device in the socket before or after the socket has been connected to the main board.
- Insert or remove a device in the socket before or after power has been applied to the main board by connecting the USB cable to your PC.
- 4. Connect the QFN Field Programmer Socket Board with the device into the Field Programmer.



Field Programmer Field Programmer Socket Board

Figure 14. Connections from PC to the Target Device

5. Start ClockBuilder Pro by locating the icon on your desktop or Windows Start Menu.



Figure 15. ClockBuilder Pro Icon

6. The ClockBuilder Pro Wizard main menu should now appear, as shown in the figure below. Select the "NVM Burn Tool" as shown. *Do not select EVB GUI.* 



Figure 16. ClockBuilder Pro Wizard

7. If this is the first time you are launching the NVM Program Tool and no socket board has been detected, the tool will prompt you to select the device family you are targeting, as shown in the following figure:

B NVM Program Tool - Clock	Builder Pro v4.1	-		×
Field Programmer Mode:	Wired to Board (No Socket Detected)			
Target Device:	Select			
No field programmer kit s	ocket board detected. Attach socket now to burn NVM on loose parts.			
If you want to burn NVM o	on a device attached to the field programmer via wired serial connection, select t to the selector above.	he devic	e family	
you are programming as				

Figure 17. Select Device Family Prompt

8. Once you insert the socket in the field programmer, the tool will detect it and automatically load the appropriate programming panel:

CB NVM Program Tool - Clock	kBuilder Pro v4.1	- 🗆 X
Field Programmer Mode: Target Device:	Socket, Si534x/7x/9x-QFN44 Si534x/7x/8x/9x (not firmware based)	
Project File: Project File Created By: Project Part: Project Design ID: Project Design Check: Project File NVM Hash:	     	Select Project Clear OPN Lookup
Device Part Number: Device Design ID: Device NVM State:		Scan for Device Clear
# Valid Burns: # Burns with Error:	0 0 Program NVM	

Figure 18. Programming Panel

# 5.1.1. Programming In-Socket, Si5383/4/8/9 Firmware Based Devices

Refer to Figure 20, "Programming In-Socket, Firmware Based Devices," on page 15 below.

- 1. Configure the serial interface.
- 2. Common issues:
- If I<sup>2</sup>C is used, check that the I<sup>2</sup>C address matches what is expected (see datasheet).
- For both SPI and I<sup>2</sup>C, start with a low bus speed. There can be connection issue due to signal integrity at high bus speeds.
- 3. Select the firmware source.
- Configuration + Program from Project File

The configuration defined by the specified project + the firmware release selected in the project file will be used to generate the firmware image that will be flashed on the device. Note that different versions of CBPro may compute configuration registers differently for the same design goals as improvements are made to CBPro.

## - Configuration + Program from Firmware File

Flash a stand-alone hex or binary firmware file to the device. You must have previously exported the file in CBPro, or the file was sent to you by Skyworks. The firmware image contains both configuration and program data. This option is useful if you want to ensure the same configuration register data is flashed to the device regardless of the CBPro version this tool is running on. Firmware images can be created from the CBPro dashboard using the Export tool, selecting the stand-alone file option.

Si5383 Expor	t				-		
ntroduction	Firmware Image	Register File	Settings File	Multi-Project Register/Settings	Regma	с	
About Firm	ware Export						
This export	t will create a cus	tom firmware	image or boo	t record for the Si5383 MCU.	In either	case, th	ie
device con	figuration specifie	d in this desigi	n will be embe	dded in the firmware.	- MOU	c	
upgrade pi	ocess.	anniy Kelelen		more information regarding	ine wico	mmwa	e
A comman	id line version of	this tool is a	vailable. Type	CBProSi534x8xFirmwareExp	orthel	p from	a
command	prompt to learn m	ore.					_
Options							
Export Type	2:						
O Boot	Record						
Firmv	vare image packed	l into a boot re	ecord that is c	ompatible with the MCU bootle	oader.		
Stand Cirmu	I-Alone Firmware	urned using a	Cleanorka LICD	Dobug Adoptor (UDA)			
Export Form	nat:	unieu using a	SKYWOIKS USD	Debug Adapter (ODA).			
🔘 Binar	ý						
🔵 Intel I	Hex						
							_
			Save to File	e			

Figure 19. Stand-Alone Firmware Selection in the CBPro Export Menu for a Si5383 Device

- 4. Click the "Select ..." button and select the file to flash to the device.
- 5. Click the "Scan for Device" button (optional): Click to detect device and report on part number, firmware version, and DESIGN\_ID. This is optional. You can click 'Program NVM' without first scanning and all relevant pre-burn checks will be performed. Note a device scan is also performed after the NVM burn has been completed, regardless of whether the burn completed successfully or not.
- 6. Click the "Program NVM" button to flash device. In project file mode, CBPro will create a firmware image behind the scenes based on the project file configuration, and then flash this on the device. The firmware download is verified via read back.

CB NVM Program Tool - Cloc	kBuilder Pro v4.1 -	×	CB NVM Program Tool - Clock	Builder Pro v4.1		- 0	×
Field Programmer Mode: Target Device:	Socket, LGA56 SiS383/84 (firmware based)		Field Programmer Mode: Target Device:	Socket, LGA56 Si5383/84 (firmware	based)		
Host Interface: Firmware Source:	I2C Address 0x6C / 108d; 400 kHz; 3.3V       — 1         I2C Address       0x6C       108         7-bit address, range 1-127       •         I2C Bus Speed       400 kHz       •         I2C Bus Speed       0x6kHz       •		Host Interface: Firmware Source: 2	I2C Address 0x6C / 1 Configuration + P Project File: Creator: Part:	08d; 400 kHz; 3.3V pogram from Project File C:\Users\thumen\Desktop\Si5383-RevD-5383EVB1- Project.slabbimeproj CBPro V2.15 Si5383 Rev D	Select Clea	sar
Firmware Hash:	Design ID: Design Check: O Configuration + Program from Firmware File <b>9</b> <b>9</b>	_	2 -	Firmware Release: Design ID: Design Check: Configuration + P	1.0 (Official Release) 5383EVB1 OK Togram from Firmware File @ (CR001040968200730) (once to disboard) @		_
Device Part Number: Device Firmware: Device Design ID:	Scan for Device Clear		Device Part Number: Device Firmware: Device Design ID:		4 <b>➡</b> Sca	n for Device Clea	ar
# Valid Burns: # Burns with Error:	0 0 Program NV/M		# Valid Burns: # Burns with Error:	0	Program NVM 🗲 5		

Figure 20. Programming In-Socket, Firmware Based Devices

# 5.1.2. Programming In-Socket, Si534x/8x/9x Non-Firmware Based Devices

Refer to Figure 21 below.

- 1. Click the "Select Project" button and select the project file.
- 2. (Optional) Click the "Scan for Device" button to detect the device and report on part number, DESIGN\_ID, and NVM bank state (number of banks already burned, number available for burn). This is optional. You can click 'Program NVM' without first scanning and all relevant pre-burn checks will be performed, such as verifying there is a bank available to burn. Note a device scan is also performed after the NVM burn has been completed, regardless of whether the burn completed successfully or not.
- 3. Click the "Program NVM" button to start the programming flow:
  - a. CBPro will compute the registers to program based on the design goals entered in the project file, using the latest algorithms embedded in CBPro.
  - b. CBPro will write volatile configuration registers corresponding to the project.
  - c. CBPro will initiate a bank burn.
  - d. CBPro will force an NVM reload on the device.
  - e. CBPro will verify the bank burn by inspecting the bank pointer and read back the programmed registers.
  - f. CBPro will rescan for the device and update burn count at the bottom of the window.

CB NVM Program Tool - Clock	KBuilder Pro v4.1		_		$\times$
Field Programmer Mode:	Socket, Si534x/7x/9x-QFN44	1			
Target Device:	Si534x/7x/8x/9x (not firmware based)		<b>,</b>		
Project File:	C:\Users\ngk\Downloads\Si5342-RevD-Project.slabtimeproj	Select I	Project	Cle	ear
Project File Created By:	CBPro v4.1			un .	
Project Part:	Si5342 Rev D	<b>`</b>		, up	
Project Design ID:	5342EVB3				
Project Design Check:	ОК				
Project File NVM Hash:	0xE54F502D1214F7B1D2D0D693A0EBEEE5 (copy to clipboard)				
Device Part Number:		Scan fo	r Device	Cle	ear
Device Design ID:					
Device NVM State:		1	ſ		
# Valid Burns:	0	2	2		
# Burns with Error:	0				
	Program NVM 3				

Figure 21. Programming In-Socket, Non-Firmware Based Devices

# 5.1.3. In-Socket Programming Status

During the programming process and if the programming is successful, you should see the following windows.

CB NVM Program Progress	×	CB NVM Program Success	×
Writing configuration to non-volatile memory		Configuration burned to NVM and verified via read-back.	
		ОК	



# 5.2. In-System Firmware/NVM Programming

This workflow describes the process of programming a device mounted on a PCB. For Si534x/8x/9x (not firmware based) devices, 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 Skyworks always have two NVM banks available to program ("burn"). If you don't know how many banks are still open to burn on your target device, CBPro can detect and report the number of remaining NVM banks. For Si5383/84 (firmware based) devices, this flow will flash a complete configuration from CBPro into the device. NVM Programming is supported for select configurations for the Si536x/Si540x/Si55xx devices. Check the device reference manuals for details.

The steps needed to program an "in-system" device's NVM are as follows:

- 1. Install the CBPro software if you have not already done so.
- 2. Connect the adapter (Field Programmer) board with the USB cable to the PC on which CBPro was installed.



Computer Running CBPro

Figure 23. PC to Field Programmer Connection

3. Look up and verify the host I/O mode ( $I^2C$  or SPI), the  $I^2C$  address, and the interface I/O voltage level compatibility of your host's I/O voltage (for  $I^2C$  or SPI) and the device.

On Si534x/8x/9x devices, the value set at the device register address of 0x0943 determines how the I/O supply voltages must be configured to communicate reliably with the Field Programmer. You can look up your device host I/O voltage using the "OPN Lookup" option in the NVM Burn tool, as shown below.

Field Programmer Mode:	Socket, Si534x/7x/9x-QFN44	
Target Device:	Si534x/7x/8x/9x (not firmware based)	
Project File:		Select Project Clear
Project File Created By:		
Project Part:		ОРН Свокир
Project Design ID:		
Project Design Check:		
Project File NVM Hash:	0	
Device Part Number:		Scan for Device Clear
Device Design ID:		
Device NVM State:		
# Valid Burns:	0	
# Burns with Error:	0	

Figure 24. OPN Lookup Option

If you have a custom OPN mounted on your board (a part number with a five-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 0x0943) by selecting the OPN Lookup option. A browser will open, and you will then enter your custom OPN, as shown below.

- a. Enter in your full ordering part number (OPN). E.g., Si5342D-D14249-GM.
- b. Click Search.
- c. Click the addendum link to download the device addendum.

SKYWORKS Look Up an Oscillator or Clock @ Part Number Ex: 530, 570BBA000653DG, Si5332DD13997-GM1 Si5342D-D14249-GM Search Mark Code Ex: 100209 a b Existing Custom Parts | 1 result Part Number: Si5342D-D14249-GM Request Date 10/18/2021 **Order Part Number or Samples** Part Number Revision 0 **Contact Sales** Product Si5342D Modify a Custom Part Number Data Sheet Addendum Addendum C Configure new part from Si5342D-D14249-GM Data Sheet

#### Figure 25. OPN Lookup

- 4. Verify the I/O Power Supply setting of your device in the Data Sheet Addendum.
- "VDD (Core)" indicates the I/O supply for the Si534x/8x/9x I<sup>2</sup>C/SPI interface will operate from a 1.8 V supply.
- "VDDA (3.3 V)" indicates the I/O supply for the Si534x/8x/9x I<sup>2</sup>C/SPI interface will operate from a 3.3 V supply.
- "VDDD" indicates the I/O supply for the Si5332 I<sup>2</sup>C interface. "VDDIO" indicates the I/O supply for the Si536x, Si540x, and Si55xx devices.

The image below 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 26. Finding VDDA Value

Table 2. Support	ted Serial Protoc	ols for the Si53	4x/8x/9x Devices
------------------	-------------------	------------------	------------------

	1.8 V	2.5 V	3.3 V
4-wire SPI			
3-wire SPI	Supported		
I <sup>2</sup> C			

For Si536x, Si540x, and Si55xx devices, I/O supply voltage is listed under "VDDIO".

### Figure 27. Finding the VDDIO Setting

The list of supported SPI modes and voltages for the Si536x, Si540x, and Si55xx are listed in Table 3.

#### Table 3. Supported Serial Protocols for the Si536x, Si540x, and Si55xx Devices

	1.8V 2.5V		3.3V		
4-wire SPI	Supported				
3-wire SPI	Not Supported Not Supported Supported				
l <sup>2</sup> C	Supported				

5. Connect/wire the pins of the Field Programmer to your host system with the target device. Use the femaleto-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 on your PCB. Note the pinout diagram and descriptions in the table below.



Figure 28. Interface Pins on Header (Front View of the Field Programmer)

Pin #	Description	Wire to Your PCB?	l <sup>2</sup> C	4-wire SPI	3-wire SPI		
1	GND	Always	GND				
2	ID	Never	Select programming Field Prog the board. For in-system progra signal.	rammer Socket Boards provide a amming, this pin should be grou	a voltage on this pin to identify inded or not connected to any		
3	SCLK	Always	Serial clock signal for I <sup>2</sup> C transactions.	Serial clock signal for SPI transa	actions.		
4	VDDA_VDDS (Si536x, Si540x, and Si55xx devices do not use this pin)	Never	Supplies the VDDA and VDDS v Programmer Socket Board. Do	upplies the VDDA and VDDS voltages to the device when using a programming Field ogrammer Socket Board. Do not use this pin for in-system programming.			
5	A1_SDO (applies only for Si534x/8x/9x, Si55xx, Si540x, Si536x devices)	4-Wire SPI Only	For Si534x/8x/9x devices, this pin can be used to set I <sup>2</sup> C address bit A1 high or low. Routed to A1 device pin on the programming Field Programmer Socket Boards. For Si536x, Si540x, Si55xx devices, this pin is used to received data from the device in 4-wire SPI mode.	Serial data from device for 4- wire SPI transactions (MISO).	Not used		
6	I2C_SEL2 (applies only for Si534x/8x/9x device)	Never	Used to set I2C_SEL signal high to set the device for I <sup>2</sup> 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. (Refe to specific part pinout and the programming Field Programmer Socket Board to determine whether to use I2C_SEL1 or I2C_SEL2)			
7	SDA_SDIO	Always	Serial data signal for I <sup>2</sup> C transactions.	Serial data out to device for 4- wire SPI transactions (MOSI).	Bidirectional Serial data for 3- wire SPI transactions (SDIO).		
8	I2C_SEL1 (applies only for Si534x/8x/9x device)	Never	Used to set I2C_SEL signal high to set the device for I <sup>2</sup> C communication. (Refer to specific part pinout and the programmer Socket Board to determine whether to use I2C_SEL1 or I2C_SEL2)	Used to put I2C_SEL signal low for SPI communication. (Ref to specific part pinout and the programming Field Programmer Socket Board to determine whether to use I2C_SEL1 or I2C_SEL2)			
9	A0_CSB (applies only for Si534x/6x/8x/9x, Si55xx, Si540x, device)	3- or 4-Wire SPI	Can be used to set I <sup>2</sup> C address bit A0 high or low. Routed to A0 device pin on the programming Field Programmer Socket Boards.	; Drives the chip select signal during SPI transactions			
10	VDD	Never	Supplies the Core VDD voltage Socket Board. Do not use this p	plies the Core VDD voltage to the device when using a programming Field Programmer ket Board. Do not use this pin for in-system programming.			

#### Table 4. Interface Pin Connections from the Field Programmer

# 5.2.1. I<sup>2</sup>C Hardware Configuration

For I<sup>2</sup>C Communication connecting to an external device board, the following pins should be used from the:

Field Programmer

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

Si534x/8x/9x Devices:

- A0/CS: Drive this pin high or low to set the I<sup>2</sup>C Address.
- A1/SDO: Drive this pin high or low to set the I<sup>2</sup>C Address.
- I2C\_SEL: Drive this pin high to select I<sup>2</sup>C communication.

Si5332/7, Si536x, Si540x, Si55xx Devices:

- The device must be explicitly configured for I<sup>2</sup>C from the Host Interface page of ClockBuilder Pro. The field programmer supports I<sup>2</sup>C at 1.8V, 2.5V, and 3.3V.
- A0/CSb: Drive this pin high or low to set the I<sup>2</sup>C Address.



Figure 29. Example I<sup>2</sup>C 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. Use short wires when possible.

# 5.2.2. SPI 3-Wire Hardware Configuration

For 3-wire SPI communication, when connecting to an external device board, the following pins should be used from:

Field Programmer

- Pin 3: Serial Clock SCLK
- Pin 7: Serial Data SDIO for Data In and Out
- Pin 9: A0\_CSB for Chip Select
- Pin 1: Ground

Si534x/8x/9x device

• I2C\_SEL: Drive this pin low to select SPI communication.

Si536x, Si540x, Si55xx device

• The device must be explicitly configured for 3-wire SPI from the Host Interface page of ClockBuilder Pro. The field programmer only supports 3.3V communications with these devices.

# 5.2.3. SPI 4-Wire Hardware Configuration

For 4-wire SPI communication, when connecting to an external device board, the following pins should be used from:

**Field Programmer** 

- Pin 3: Serial Clock SCLK
- Pin 7: Serial Data SDIO for Data In to device (MOSI)
- Pin 5: A1\_SDO for Data Out of device (MISO)
- Pin 9: A0\_CSB for Chip Select
- Pin 1: Ground

Si534x/8x/9x Device

• I2C\_SEL: Drive this pin low to select SPI communication.

Si536x, Si540x, Si55xx Device

• The device must be explicitly configured for 4-wire SPI from the Host Interface page of ClockBuilder Pro. The field programmer supports 3.3V, 2.5V, and 1.8V 4-wire SPI communications with these devices.

## 5.2.4. Programming In-system, Si5383/4/8/9 Firmware Based Devices

Refer to Figure 30. Programming In-system, Firmware Based Devices below.

After verifying the CBPro Field Programmer to device connections, execute the following steps. This example assumes a device is configured with an  $I^2C$  address of 0x6C, and an  $I^2C$  bus speed of 400 kHz.

- 1. Select "Si5383/4/8/9 (firmware based)" in the Target Device drop down.
- 2. Click the Host Interface drop down:
  - a. Enter the I<sup>2</sup>C address of the device.
  - b. Select the communication bus speed.
- 3. Select the firmware source.
- Configuration + Program from Project File
- The configuration defined by the specified project + the firmware release selected in the project file will be used to generate the firmware image that will be flashed on the device. Note that different versions of CBPro may compute configuration registers differently for the same design goals as improvements are made to CBPro.
- Configuration + Program from Firmware File
   Flash a stand-alone hex or binary firmware file to the device. You must have previously exported the file in
   CBPro, or the file was sent to you by Skyworks. The firmware image contains both configuration and
   program data. This option is useful if you want to ensure the same configuration register data is flashed to
   the device regardless of the CBPro version this tool is running on. Firmware images can be created from the
   CBPro dashboard using the Export tool, selecting the stand-alone file option.
- 4. Click the "Select Project ..." button and select the project file to be written to the device.
- 5. (Optional) Click the "Scan for Device" button to detect device and report on part number, firmware version, and DESIGN\_ID. This is optional. You can click Program NVM' without first scanning and all relevant preprogram checks will be performed. Note a device scan is also performed after the NVM programming has been completed, regardless of whether the programming completed successfully or not.
- 6. Click the "Program NVM" button to flash device. In project file mode, CBPro will create a firmware image behind the scenes based on the project file configuration, and then flash this on the device. The firmware download is verified via read back.

and the base of the
based)
08d; 400 kHz; 3.3V
4E04D49B6BE0273D (copy to clipboard)
Scan for Device Clear



# 5.2.5. Programming In-system, Si534x/8x/9x Non-firmware Based Devices

# Refer to Figure 31. Programming In-system, Non-firmware Based Devices below.

After verifying the CBPro Field Programmer to device connections, execute the following steps. This example assumes a device is configured with the host  $I^2C$  interface operating in 3.3 V I/O mode with an  $I^2C$  address of 0x68, and an  $I^2C$  bus speed of 400 kHz. For Si536x, Si540x, and si55xx devices, this section does not apply because the device does not have any available NVM banks. The example below shows the workflow for a Si534x/8x/9x device.

- 1. Select "Si534x/8x/9x (not firmware based) in the Target Device drop down.
- 2. Click the Host Interface drop down (Review: host I/O mode (I<sup>2</sup>C or SPI), the I<sup>2</sup>C address,
  - and I/O voltage level to determine these settings):
  - a. Select communication protocol for the device.
  - b. Select the I/O voltage for the device
  - c. For  $I^2C$ , enter the address of the device.
  - d. Select the communication bus speed.
- 3. Click the "Select Project ..." button and select the project file to be written to the device.
- 4. (Optional) Click the "Scan for Device" button to detect the device and report on part number, DESIGN\_ID, and NVM bank state (number of banks already burned, number available for burn). This is optional. You can click Program NVM' without first scanning and all relevant pre-programming checks will be performed, such as verifying there is a bank available to burn. Note a device scan is also performed after the NVM burn has been completed, regardless of whether the burn completed successfully or not.
- 5. Click the "Program NVM" button to start the programming flow:
  - a. CBPro will compute the registers to program based on the design goals entered in the project file, using the latest algorithms embedded in CBPro.
  - b. CBPro will write volatile configuration registers corresponding to the project.
  - c. CBPro will initiate a bank burn.
  - d. CBPro will force an NVM reload on the device.
  - e. CBPro will verify the bank burn by inspecting the bank pointer and read back the programmed registers.
  - f. CBPro will rescan for the device and update burn count at the bottom of the window.

Field Programmer Mode:	Wired to Board (	No Socket Detected)		
Target Device:	Si534x/7x/8x/9x	(not firmware based)	<b>4</b> 1	
Host Interface:	I2C Address 0x6	8 / 104d; 400 kHz; 3.3V	<b>4</b> 2	
Project File:	Protocol	O SPI 4-Wire	3 📥	Select Project
Project File Created By:		O SPI 3-Wire		ODNU selsus
Project Part:		I2C		OPIN LOOKUP
Project Design ID:				
Project Design Check:	I/O Voltage	3.3 V		
Project File NVM Hash:				
Device Part Number:	I2C Address	68 104		
Device Design ID:		7-bit address, range 1-127	4 🖛	Scan for Device Clear
Device NVM State:	I2C Bus Speed	400 kHz		
# Valid Burns:	0		,	
# Burns with Error:	0			

Figure 31. Programming In-system, Non-firmware Based Devices

# 5.2.6. Programming Status

During the programming process and if the programming is successful, you should see the following windows:

CB NVM Program Progress ×	CB NVM Program Success X
Writing configuration to non-volatile memory	Configuration burned to NVM and verified via read-back.
	ОК



# 5.3. In-System Volatile Register Programming and Register Debug

This workflow guides users through the full CBPro configuration Wizard to make volatile changes to a device's configuration, and also walks the user through the CBPro EVB GUI application to inspect the state of various status registers and other information via the Device APIs (where supported) in real-time. There are two ways you can interact with your PCB-based device using the field programmer:

- Use CBPro Design Dashboard to edit your device configuration, and write out changes directly to your device.
- Launch the EVB GUI, to inspect registers and execute Device API commands where supported.

All of the relevant CBPro features available when working with a Skyworks 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 device 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 device (these registers are skipped when writing a design to a Skyworks Si534x/8x/9x evaluation boards, but are written to the Si536x, Si540x, and Si55xx evaluation boards. If using a Si536x, Si540x, or Si55xx evaluation board, check the board schematic and make the required board modifications).

# 5.3.1. Using the CBPro Design Dashboard

When you launch CBPro, instead of clicking the NVM Burn Tool, open your existing project file or a sample file to open the design dashboard window as shown in Figure 33.



Figure 33. Open Design Project File, and see Field Programmer Detected

# 5.3.2. Overview of CBPro Configuration Wizard and the Field Programmer

When you open a ClockBuilder Pro project file, you are taken to the design dashboard. This is a gateway to perform activities against your design, including writing your project's configuration to a device using the CBPro Field Programmer. For example, in the figure below, a Si5397 project has been opened and the CBPro Field Programmer has been detected, and no socket is present:

CB Open Sample Project - ClockBuilder Pro	– 🗆 X
ClockBuilder Pro 🖽 🗫	SKYWORKS
Design Dashboard 🔻	Configuring Si5397A Rev A
Loaded Si5397 sample design. You can review or edit the configurate below or using the pulldown step menu above.	ion by clicking a link in the "Edit Configuration with Wizard" section
Edit Configuration with Wizard Design ID & Notes · Host Interface · Reference · Free Run · Inputs · Input Select · Outputs · DCO · Hittess Switching Assistant · DSPLL · LOS · OOF · LOL · INTR         Save Design to Project File Your configuration is stored to a project file, which can be opened in ClockBuilder Pro at a later time.         Design Report & Datasheet Addendum You can view a design report (text) or create a draft datasheet addendum (PDF) for your design.	Field Programmer Detected       Image: Wired to PCB (serial)         Interface:       SPI 4-Wire; 1 MHz; 3.3V         Write Design to DUT       Open EVB GUI         Image: Export       You can export your configuration to a format suitable for in-system programming.         Image: Documentation       Si5397/96 Reference Manual Si5397/96 Data Sheet         Si5397/96 Data Sheet       Si5397/96 Laser's Guide
<b>Skyworks Cloud Services</b> You can <u>create a custom part number</u> for your design, which can be used to order factory pre-programmed devices. Or <u>request a phase noise report</u> for this design.	Ask for Help Have a question about your design? Click here to get assistance.
🕒 Frequency Plan Valid 🧭 Design OK 😗 Typical Pd 1.32 W, Tj 4	B°C Home Close

Figure 34. Overview of CBPro Configuration Wizard and the Field Programmer

With a click of the "Write Design to DUT" button, you can reconfigure the Si5397 in-system to test changes to your design. The "Open EVB GUI" button can be used to launch the EVB GUI. The EVB GUI presents a real-time view of the status of the clock device, and provides an interface to interact with the device during runtime. For Si534x/8x/ 9x devices, the EVB GUI allows you to peek/poke registers, and watch realtime lock flags and alarms on the insystem device. For Si536x, Si540x, and Si55xx devices, the EVB GUI allows you to view the realtime PLL lock status of the device, view alarms, and interact with the device through the device API. See "5.3.4. Using the EVB GUI with In-System Devices" on page 32 to learn more.

## 5.3.2.1. Using the Dashboard with In-system Devices

If the CBPro Field Programmer is connected via USB and detected by CBPro, you will see will see a pulldown to configure the host interface between the Field Programmer and your PCB, as shown in the figure below. Refer to "5.2. In-System Firmware/NVM Programming" on page 17 for information on connecting the CBPro Field Programmer to your hardware.

Field Programmer Detected Target: Wired to PCB (serial)						
	Interface: I2C Address 0x6C / 108d; 100 kHz; 3.3V					
	Write	Design to DUT Open EVB GUI				
	Figu	re 35. Field Programmer Detected				

Once detected, select the appropriate settings via the dropdown menus.

		Field Pr Target:	<b>OGRAMMER [</b> Wired to PCB (see	Detected rial)	
		Interface:	I2C Address 0x6	C / 108d; 100 kHz; 3.3V	•
 Field Programmer Detected		Write	Protocol	O SPI 4-Wire O SPI 3-Wire	1
 Target: Wired to PCB (serial)	┢	Export		12C	ala far
Interface: I2C Address 0x6C / 108d; 100 kHz; 3.3V		in-system	I/O Voltage	3.3 V	Die for
Write 12C Address 0x6C 108		-	I2C Address	0x6C 108	
You can e I2C Bus Speed 100 kHz ble for in-system programming.		Docum <u>Si5341/4</u> <u>Si5341/4</u>	I2C Bus Speed	100 kHz	

#### Figure 36. Communication Interface Selection

Once configured, you can write out your design to the device by clicking the Write Design to DUT button:

	Field Programmer Detected								
	- <del></del> -	Target: Wired to PCB (serial)							
	Interface: I2C Address 0x6C / 108d; 100 kHz; 3.3V						•		
		Write	Desig	n to DUT		Open EVB GUI			
1					_				

Figure 37. Write Design to DUT

Or on any configuration page in the wizard:

Write to FP	< Back	Next >	Finish	Cancel

#### Figure 38. Write to FP

When you initiate a project write to the device, CBPro will first try to verify the device is present via the communication interface you have configured. This is normally accomplished by trying to read device identification register on the device, such as PN\_BASE on Si534x/8x/9x devices, or by trying the SIO\_TEST API command for Si536x, Si540x, and Si55xx devices.

If it cannot read these registers, the device write will be aborted and you will see an error message like the example shown in the figure below:



#### Figure 39. Error Message

Si536x, Si540x, and Si55xx devices will generate an error based on a failed API command read/write:



Figure 40. Si536x, Si540x, and Si55xx (API based) Write Error

## 5.3.2.2. Using the CBPro Dashboard with In-socket Devices

In the design dashboard, you will see a dropdown menu to configure the host interface between the CBPro Field Programmer and the socket. If the connected socket is not compatible with the selected CBPro project file, an error message will be displayed and the interface configuration pulldown will be disabled, as shown in Figure 41.

Field Programmer Detected	Field Programmer Detected			
 Target: Socket, LGA56 Off		Target:	Socket, QFN44 (not compatible with Si5383)	
Interface: I2C Address 0x6C / 108d; 100 kHz; 3.3V		Interface:	I2C Address 0x6C / 108d; 100 kHz; 3.3V	
Write Design to DUT Open EVB GUI		Write	Design to DUT Open EVB GUI	



**Note:** Manually powering up the socket is an optional step. If you click the "Write Design to DUT" button, CBPro will automatically power up the socket (and you will see it switch from Off to the On state). Socket power refers to VDD and VDDA power on the device.

	Field Pr Target:	ogrammer [ Socket, QFN44	Detected	Ţ
	Interface:	I2C Address 0x60	C / 108d; 100 kHz; 3.3V	
Field Programmer Detected       Target:     Socket, LGA56	Write	Protocol	<ul> <li>SPI 4-Wire</li> <li>SPI 3-Wire</li> </ul>	
 Interface: I2C Address 0x6C / 108d; 100 kHz; 3.3V	Export		I2C	
Write II2C Address 0x6C 108	You can e in-system	I/O Voltage	3.3 V	ble for
Export     7-bit address, range 1-127       You can e     100 kHz		I2C Address	0x6C 108	
in-system programming.	Docum Si5345/44	I2C Bus Speed	100 kHz	
	<u>Si5345/44</u> Si5344 Re	VA2 Rev D Datas V D EVB User's (	s <u>heet</u> Guide	,



Once configured, you can write out your design to the device by clicking the "Write Design to DUT" button:

Field Programmer Detected	
Target: Socket, LGA56 On	
Interface: I2C Address 0x6C / 108d; 100 kHz; 3.3V	
Write Design to DUT Open EVB GUI	

Figure 43. Write Design to DUT

Or on any configuration page in the wizard:



Figure 44. Write Design to FP

## 5.3.3. Launching the CBPro EVB GUI

From the CBPro Wizard screen, click the EVB GUI button to open the EVB GUI screen.

Work With a Design
Create New Project
Dpen Project
Convert Existing Project/NVM File
ex Open Sample Project
Field Programmer Detected

Figure 45. Open EVB GUI Screen

If this is the first time launching the EVB GUI and no socket board is detected, the tool will prompt user to select the device family they are targeting:

- 🗆 X
Field Programmer
Family: Si55xx Target: Wired to PCB (serial) Interface: SPI 4-Wire; 1 MHz; 3.3V ROM Part Number CBPro CBPro Config Scan
5xx PI 4-Wire PI 3-Wire
v 🔽
AHz V
3

Figure 46. Select Device Family Prompt

If a socket is connected, the family is auto selected based on the socket. The tool polls for socket state every 500 milliseconds and will detect if a socket is present or has been changed.

CB Fie	Id Programmer - Clock	Builder Pro					-		×
File H	lelp								
Info	DUT Settings Editor	DUT Register Editor	Status		Field	Program	mer		
Info v	vill be available once a	a device has been succ	essfully	scanned by the field programmer.	Family Targe Socke Interfa ROM Part N Applia CBPric	y: t: ace: Number cation	Si55xx Socket, 9 QFN72 SPI 4-Wi 3.3V	fi ff re; 1 MH	55хх- z;
Log Filtere Times	ed 💽 Auto Scrol stamp Source	l: On 🔽 Insert Ma Message	rker	Clear Copy to Clipboard	Devic	Cor e Contro ce contro	fig S	ican	once
VB Firr	mware 0.84    Field Prog	grammer: Socket, Si54xx	/Si55xx-C	2FN72; Si55xx; SPI 4-Wire; 1 MHz; 3.3V		ClockB	ilder Pro v	4.1 [2021	-09-2

Figure 47. Socket Detected, Auto-Selected Family Prompt

## 5.3.4. Using the EVB GUI with In-System Devices

Connect the CBPro Field Programmer to the PCB mounted device. Refer to "5.2. In-System Firmware/NVM Programming" on page 17 for information on connecting the CBPro Field Programmer to your hardware.

- 1. Check that the appropriate voltage rails of the device are active. Attempting to communicate with a powered-off device will result in errors.
- 2. Click the Config button and click the Device Family pulldown to select the appropriate target device.
- 3. Configure the host interface to match what is being used on the device. In the example below, the Field Programmer is connected to a Si5518 device over 3-wire SPI.
- 4. I/O voltage should be set to match the voltage of the device. If you do not know the I/O voltage setting of the device, follow the steps in "5.2. In-System Firmware/NVM Programming" on page 17 to find out what I/O voltage is being used on your device.
- 5. After the configuration is complete, click the Scan button.



#### Figure 48. Configuring an In-System Device

The Part Number and Design ID fields should update with the device information under the Info tab. For Si534x/8x/9x devices, the DUT Register Editor tab can be used to make volatile register value changes to the device and the Status Registers tab can be used to monitor the status of the device. For Si536x/540x/55xx devices, the Device API tab can be used to interact with the device. Screen shots that follow walk through some features of the EVB GUI on Si55xx devices. Some features are shared between part families, while others may be unique.

B Field Programmer - C	lockBuilder Pro	- 🗆 ×
ile Help		
Info DUT Settings Ed	tor DUT Register Editor Device API Device API (Low-Level) Status	Field Programmer
Field Programmer Ider Serial Number:	tification: 00-00-2E-45-FD-79	Family: Si55xx Target: Wired to PCB (serial) Interface: SPI 3-Wire; 500 kHz; 3.3
DUT Identification:		ROM 2.0.0 SVN: 0
DEVICE_INFO API	Part: SI5518A-B-GM ROM: 2.00 SVN: 0 HW: 0.0 Variant: 0 Part grade. revision information. Empty until firmware has been loaded.	Part NumberSi5518A-B-GM Application 1.0.0.5894 CBPro 4.1.1.100 Design ID FP-DEMO
APP_INFO API	CBPro: v4.1.1.100 Firmware: v1.0.0.5894 Planner: v1.0.0.5895 Design ID: FP-DEMO Firmware and frequency planner revision information. Endpy until firmware has been loaded.	Device Control / Misc Fixed-Step DCO FINC FDEC
		SYSREF Pulser Pulse Phase Readout
		Die Temperature
.og		°C Read Poll
Filtered 📘 Auto :	croll: On 📱 Insert Marker Clear Copy to Clipboard Pause	
Timestamp Source	Message	
11:12:00.723 EVB	Finished Read_DUT_Bytes(address=0x2000, num_bytes=260) =>	
/B Firmware 0.84    Field	Programmer: Wired to PCB (serial); Si55xx; SPI 3-Wire; 500 kHz; 3.3V	ClockBuilder Pro v4.1.3.100 [2021-11

Figure 49. Device info is Displayed after a Successful Scan Operation (Si5518 Device)

To view the device status in real-time, go to the "Status API" tab (you may have to use the small arrows to scroll over). In the example below, the real-time display shows that there is an issue with the reference clock input, and all the PLLs are not locked due to this issue.

Refresh	All Clear Flags	Copy State to Clip	board	Auto	Poll Sta	atus AP	1	Last Update: 11:27:
Keywords:		Clear	) AND		2 Lev	vel: All		
Key: Not	Asserted Info Valid	Varn Error						
Level	Description		Flag		-			
Primary	(API)		1055.0					
Drimony	Pof Out Of Frequency				ENCV			
e i i i i i i i i i i i i i i i i i i i	Rel Out-OI-Frequency							Reference Invalid
Primary	Phase Monitor		PHASE		IR_PHAS	SE_EKK	Эк	flag shows that
Secondary	/ Early Phase Monitor		PHASE		R_SIGN	AL_EAF	RLY	there is a problem
Secondary	/ Late Phase Monitor		PHASE		R_SIGN	AL_LAT	E	with the reference
Secondary	/ Reference Clock Status	(	REFER	ENCE_CLC	CK_STA	tus in	VLD	clock input.
Inputs (Al	PI)							
Primary	Input Loss-of-Signal		IN0 I	N1 IN2	IN2b	IN3	IN3b	
Primary	Input Out-Of-Frequency		IN0 I	N1 IN2	IN2b	IN3	IN3b	
Primary	Input Phase Monitor		IN0 I	N1 IN2	IN2b	IN3	IN3b	
Secondary	/ Input Clock Status		IN0 IN	VLD IN	INVLD	IN2	NVLD	IN2b INVLD IN3 INVLD
			IN3b II	NVLD				
Secondary	/ Early Phase Detection		IN0 I	N1 IN2	IN2b	IN3	IN3b	
Secondary	/ Late Phase Detection		IN0 I	N1 IN2	IN2b	IN3	IN3b	
PLLs (API)	I							
Primary	PLL in Holdover		PLLR	PLLA	LLB P	LL_PPS		
Primary	Holdover History Valid		PLLR	PLLA P	LLB	LL_PPS		ad arror flags show that
Primary	PLL Initial Lock		PLLR	PLLA P	LLB	LL_PPS		Led error hags show that PLLs are not locked.
Primary	PLL Loss-of-Lock		PLLR	PLLA P	LLB P	LL_PPS		

Figure 50. EVB GUI Reference Error Example, Si5518

To generate the scenario above, the CBPro frequency plan that was loaded onto this device had an incorrect reference frequency of 12.345MHz. On a customer board, a more likely error is that the external clock is powered on but at the incorrect frequency. Correcting the reference frequency to what is actually fed to the reference input (54MHz) and overwriting the plan onto the Si5518 device clears the reference clock status invalid error.

C	CB Open Sample Project - ClockBuilder Pro											
С	lockBuilder	Pro	v4.1 🍫 (no overrid	es)	1							
St	ep 4 of 23 - Referer	nce 🔻										
	Reference Mode Reference Type Reference Format	Dual R XO (RI	Ref - JA		In <b>single refere</b> determines botl holdover stabili <sup>-</sup> jitter XO or VCX for best jitter pe							
	Reference Frequency	123.45			TCXO may be of performance w mode is recom							
		Freque	54M 123.45 MHz Frequency Range: 30 ency entry examples: 19.2M 19.2 MHz 19.2 MHz 1N0 + 5ppb 2*IN0 10e9*4*255 / (236* 5 MHz + 25 ppm	0.72 M 64)	Hz to 250 MHz							

Figure 51. Correcting the Reference Frequency

After correcting the frequency, the reference invalid flag turns green. However, all the input flags are red, and the PLLs are still unlocked.

Keywords:		Clear	◎ AND ○ OR ②   Level: All
Level	Description		Flag
Reference	(API)		
Primary	Loss-of-Signal	?	LOSS_OF_SIGNAL
Primary	Ref Out-Of-Frequency	?	OUT_OF_FREQUENCY
Primary	Phase Monitor	?	PHASE_MONITOR_PHASE_ERROR
Secondary	Early Phase Monitor	?	PHASE_MONITOR_SIGNAL_EARLY
Secondary	Late Phase Monitor	?	PHASE_MONITOR_SIGNAL_LATE
Secondary	Reference Clock Status	9	REFERENCE_CLOCK_STATUS VLD
Inputs (AF	비)		
Primary	Input Loss-of-Signal	?	INO IN1 IN2 IN2b IN3 IN3b
Primary	Input Out-Of-Frequency	?	INO INI IN2 IN2b IN3 IN3b
Primary	Input Phase Monitor	?	INO IN1 IN2 IN2b IN3 IN3b
Secondary	Input Clock Status	?	IN0 INVLD IN1 INVLD IN2 INVLD IN2b INVLD IN3 INVLD IN3b VLD
Secondary	Early Phase Detection	?	INO IN1 IN2 IN2b IN3 IN3b
Secondary	Late Phase Detection	?	INO INI IN2 IN2b IN3 IN3b
PLLs (API)			
Primary	PLL in Holdover	?	PLLR PLLA PLLB PLLPPS
Primary	Holdover History Valid	?	PLLR PLLA PLLB PLLPPS
Primary	PLL Initial Lock	2	PLLR PLLA PLLB PLLPPS
		•	

Figure 52. External Reference Valid but Inputs are Invalid and PLLs Not Locked

The input Loss-of-Signal errors, as well as all the other errors under the Inputs (API) section, are caused by the input clocks not being present. Connecting valid inputs will clear those errors, and PLLs should start locking.

On the Si536x, Si540x, and Si55xx devices, the Device API tab can be used to further interact with the clock device. In the image below, the Device API is used to read the die temperature of the Si5518.

CB Fi	eld Pro	ogrammer - ClockBuilder Pro															×
File	Help	-															
Info	Info DUT Settings Editor DUT Register Editor Device API Device API (Low-Level) Status												▼ Field Programmer				
Dev	Device API Documentation: View · Save to Folder · Salt to Zip											Family:	Si55xx				
TEN	TEMPERATURE_READOUT (0x19)												-	Target:	Wired t	o PCB (s	erial)
													2 Selec	t	e: SPI 3-V	/ire; 500	KHZ; 3.3V
Ar	Arguments												comma	ROM and Part Nu	2.0.0 S\ mberSi5518	/N: 0	
	ndex	Name		7	6	5	4	3	2	1	0	Actual		Applicat	tion 1.0.0.58	94	
(	0x00	CMD					CMD	(0x19)				0x19		CBPro	4.1.1.10	0	
	0x01	Х		Х	Х	Х	X	Х	Х	Х	Х	0x00	]	Design I	ID FP-DEN	10	
		Send Command						Cor	w to Clin	board					Config	Scan	
									y to chp	bound				Device (	Control / Mis	SC .	
Re	sponse	e 3 Send command	ł		4_0	lear-to	o-send is	checke orted	d, no						Fixed-Ste	ep DCO	
	ndex	Name	7	6		5	4	3	2	1	0	Actual			FIN	С	
			CTS	HW	FRR	APIERR	EWERR						1		FDE	C	
	0x00	STATUS			1			х	Х	Х	Х	0x80			CYCDEE	Dulan	
	)x01			-	-							0x62	-		Pulser		
	)x02					L 4'						0x3D		_	T dibe		
	)x03	TEMPERATURE_READOUT				44		055002)				0x03			eadout		
	)x04					5 Re	ad Resul	t				0x15			Die Temr	erature	
														43			Poll
																	1 Oli
Log																	
EVB F	irmwar	e 0.84    Field Programmer: Wire	d to PCE	3 (serial)	): Si55x	x: SPI 3-V	Vire: 500 kH	z: 3.3V						ClockB	uilder Pro v4.	1.1.100 [2	021-10-15



The field programmer supports reading and writing individual registers to in-system Si5338/51 family devices. The EVB GUI interface can be used to write exported setting or project files to these devices.
				_		
Field Programmer - ClockBuilder Pro				-		Х
File Help						
Info DUT Settings Editor DUT Register Editor Status	Ŧ	Field Pro	ogrammer			
Register Peek/Poke         Hex       Decimal         Address:       0x0001       1         # Bytes:       1       Read       Write         Hex:       0x10       16       16         Unsigned Int:       16       2       1       0         Binary:       0       0       1       0       0       0         (binary edit is only supported with 16 bits or less)       10       10       10       10		Family: Target: Interfactor	Si5351 Wired to 2: I2C Addr kHz; 1.8V Config	PCB ( ress 0x / sc r Si53!	(serial) (60 / 96 (7) (60 / 96 (7) (60 / 96 (7) (7) (7) (7) (7) (7) (7) (7) (7) (7)	id; 40

Figure 54. Registers Can be Read/Written on the Si5332/8 and Si5350/1/7 Devices

Field Program	ımer - ClockBuilder Pro	)		- 0 )
ile Help				
Info DUT Setti	ings Editor DUT Reg	ister Editor Status		<ul> <li>Field Programmer</li> </ul>
Refresh All	Clear Flag/Stic	ky Bits 🗌 Auto Poll Reg	isters 🕽	Family: Si5338/56 Target: Wired to PCB (serial) Interface: I2C Address 0x70 / 112d;
	Inputs	PLL	Other	400 kHz; 3.3V
Non-Sticky	LOS_CLKIN	PLL_LOL	SYS_CAL	Part Si5338
I	LOS_FDBK			Config Scan
Sticky	LOS_CLKIN	LOL_PLL	SYS_CAL	Device Control / Misc
I	LOS_FDBK			N/A for Si5338/56
oq				

Figure 55. Reading Live Status Bits on the Si5332/8 and Si5350/1/7 Devices

# 5.3.5. Using the EVB GUI with In-Socket Devices

CBPro will detect the connected socket when the EVB GUI is started. Click the Config button to configure the communication protocol, address (I<sup>2</sup>C), bus speed, and the I/O voltage (non-firmware based devices), as shown in the figure below. For Si536x, Si540x, and Si55xx devices, remember to plug in DC power adapter to the DC jack on the Field Programmer socket board.

**Note:** For firmware based devices the communication protocol available is  $I^2C$  with a 3.3 volt I/O voltage. For non-firmware based devices, there is a selection of SPI 4-wire, SPI 3-wire, or  $I^2C$  and the I/O voltage must be selected.

			_					
			F	ield Program	mer			
			F	amily:	Si55xx			
			Т	arget:	Socket, Si54xx/Si55xx-			
					QFN72		<ul> <li>Field Program</li> </ul>	nmer
			S	ocket Power:	Off		Family:	Si55xx
			H.	terface:	SPI 4-Wire: 1 MHz		Target:	Socket, Si54xx/Si55xx-
					3.3V		C L I D	QFN72
			-				Socket Powe	r: Off
			F	MO			Interface:	12C Address 0x68 /
			P	art Number				1040, 400 KH2, 3.3V
Field Program	imer	1	A	pplication			ROM	
- "				BPro			Part Number	·
Family:	SI55XX			esign ID			Application	
Target:	Socket, Si54xx/Si55xx-			Con	fig Scan		Design ID	
	QFN72	-			ing Scan			
Socket Power:	Off		C	Protocol	SPI 4-Wire		Co	nfig Scan
Interface:	SPI 4-Wire: 1 MHz:				O SPI 3-Wire		Protocol	SPI 4-Wire
	3.3V			1	0 I2C		(	SPI 3-Wire
								) I2C
ROM				I/O Voltag	e 3.3 V 🔽			
Part Number							I/O Voltage	3.3 V
Application				SPI Bus Sp	eed 1 MHz			0
CBPro							I2C Address	7-bit address, range 1-127
Design ID								-bit address, range 1-127
							I2C Bus Speed	400 kHz
Con	nfig Scan			Claub D	ilder Dre v.4.4 (2024, 02, 22)			
				ClockBu	ilider Pro v4.1 [2021-09-22]	kHz;	3.3V ClockE	Builder Pro v4.1 [2021-09-22]
The "Config	" button brings up th	е					0	

host interface configuration menu.

SPI Devices Configuration

I<sup>2</sup>C Devices Configuration

Figure 56. Configuring an In-Socket Device

After the configuration is complete, click the Socket Power slider and the Scan button. The Part Number and Design ID fields should update with the device information along with the Info tab field. On Si534x/8x/9x parts, the DUT Register Editor tab can be used to make volatile register value changes to the device and the Status Registers tab can be used to monitor the status of the device. On Si536x/540x/55xx parts, the Device API tab can be used to monitor the status of the device.

Program	mer	1					
mily:	Si55xx						
rget:	Socket, Si54xx/Si55xx-	CB Fie	eld Programmer - Cl	ockBuilder Pro			
	QFN72	File	Help				
ocket Power:	On 🛛 🔶	Info	DUT Settings Edit	or DUT Register Editor	Device API	Device API (Low-Level)	S
nterface <sup>.</sup>	SPI 4-Wire: 1 MHz	Field	Programmer Iden	tification:			
incernace.	3.3V	S	erial Number:	00-00-2F-45-FD-7	79	]	
ROM		DUT	Identification:				
Part Number		C	EVICE_INFO API	Part: Si5518B-B-71M			
Application				ROM: 2.0.0 SVN: 0 HW: 0.0 Variant: 0			
CBPro				Part grade, revision informatio	on.		
Design ID	🔰		1	oaded.	n		
Con	fig Scan	A	PP_INFO API				
				irmware and frequency plan	ner		

Figure 57. In-Socket Scan Prompt and DUT Info Tab

# 5.4. Firmware Update

Firmware update of the field programmer is now automatically done if CBPro detects an old firmware version on a connected Field Programmer.

To manually update the firmware at the direction of Skyworks Support, follow the instructions below. The firmware of the field programmer must be updated to be used with the Si536x, Si540x, and Si55xx devices.

## 5.4.1. Procedure for Updating the Field Programmer Firmware

- 1. Plug in the Field Programmer to a PC, with the latest version of ClockBuilder Pro installed.
- 2. Search for and open the "EVB, Field Programmer Flash Utility". This utility is bundled with the regular ClockBuilder Pro installation.

All Apps Documents Web More	<b>,</b>
Best match	
EVB, Field Programmer Flash Utility App	CBPro
Search work and web	EVB, Field Programmer Flash Utility
flash flood watch - See work and web results	Арр >
	> C Open
	> Run as administrator
♀ flash	> Open file location
♀ flash <b>score</b>	> Pin to Start
	Pin to taskbar
0 flachlight	>

Figure 58. Searching for the Flash Utility

can For Connected Devices	Compare Firmware O Choose Files	
	Compare	
v		
one Firmware	Download Firmware	
Save Device Firmware to File	Choose File	
rite EEPROM (Select EVBs Only)	O Standard Build	
write EEPROM	)	Download Fil

Figure 59. Flash Utility

3. With the Field Programmer plugged in, the Flash Utility open, click Scan. The connected Field Programmer should be displayed. If the scan did not detect a field programmer, make sure you have all other open CBPro windows closed, disconnect and reconnect the Field Programmer USB connection, then try scanning again.

EVB / Field Programmer Flash Utility - ClockBuilder Pro v4.1 [2021-09-22]		-		×
This utility should only be used at the direction of a Skyworks represent Scan For Connected Devices Scan	Compare Firmware 👔			
EV8s in program mode (1): \\? \usb#vid_10c4&pid_8a0c#si5380fpb_0084_00002f45fd79# (3c5e1462-5695-4e18-876b-f3f3d08aaf18) All EVBs (1): Name: Silicon Labs ClockBuilder Programmer, SN: si5380fpb_0084_00002f45fd79, Dev Info: Not in BL mode USB Debuggers (0):	Compare			*
Using USBXpress adapter si5380fpb_0084_00002f45fd79 for firmware operations				*
Clone Firmware	Download Firmware			
Save Device Firmware to File	Custom Choose File			
Write EEPROM (Select EVBs Only) Board ID: Vrite EEPROM	🔿 Standard Build 🔞	Dowr	nload Fi	ile
Status Finished discover				

Figure 60. Field Programmer Successful Scan

4. Select the Standard Build option, then find the latest firmware image to flash to the Field Programmer. Select the latest "cbpro\_fp\_v#.##.hex", where "v#.##" is the version number. In this example, the version number is v0.85.

can For Connected Devices	Compare Firmware 👔
Scan	Choose Files
EVBs in program mode (1): (\? \u2) \u2015&Wid 1.04&&pid .8a0c#si5380frpb_0084_00002f45fd79# (3c5e1462-5695-4e18-876b-f3f3d08aaf18) All EVBs (1): Name: Silicon Labs ClockBuilder Programmer, SN: si5380fpb_0084_00002f45fd79, Dev Info: Not in BL mode	Compare
USB Debuggers (0). Jsing USBXpress adapter si5380fpb_0084_00002f45fd79 for irmware operations	
lone Firmware	Download Firmware
Save Device Firmware to File	O Custom Choose File
Vrite EEPROM (Select EVBs Only) Ioard ID: Write EEPROM	Standard Build 2 cbpro_fp_v0.85.hex

Figure 61. Select the Latest Field Programmer Firmware Image

5. Click "Download File" and wait for the operation to complete.

# 5.5. Obtaining a Debug Log for Skyworks Support

In case Skyworks support requests a debug log of a device, follow these steps to obtain the log. The log contains a snapshot of the device volatile and non-volatile states.

### 5.5.1. Procedure for Updating the Debug Log through the Field Programmer

- 1. Follow Steps 1–5 in "5.3.4. Using the EVB GUI with In-System Devices" on page 32 to connect to the device under test.
- 2. After a successful scan, click File, then "Create DUT Dump for Skyworks Support" and wait for the log collection to complete.
- 3. Save the resulting file on your system, then send it to Skyworks Support.

B Field Progra	ammer - C	lockBuilder Pro		- 🗆 X
ile 🔫 🕇	1			
Write Proje	ect File to	Device Device API Device API (Low-Level) Status	~	Field Programmer
Write Boot	file to De	vice		Family: Si55xx
Create DU	T Dump F	ile for Skyworks Support 🗲 🔁		Target: Wired to PCB (serial)
Preference	s			Interface: SPI 3-Wire; 1 MHz; 3.3
Exit				ROM 2.3.0 SVN: 2204
		Devel CEE40A D. CM		Part NumberSi5518A-B-GM
DEVICE_IN	NFU API	ROM: 2.3.0 SVN: 2204		Application 0.10.1.5352
		HW: 1.0 Variant: 0		CBPro 4.1.0.0
		Part grade, revision information.		Design ID (empty)
		Empty until firmware has been loaded.	U	Config Scan
APP_INFO	API	CBPro: v4.1.0.0 Firmware: v0.10.1.5352		Device Control / Misc
		Planner: v0.10.1.5353		Fixed-Step DCO
og				FINC
-iltered	Auto :	Scroll: On Minsert Marker Clear Copy to Clipboard Pause		EDEC
			<b>A</b>	
2:08:36.466	EVB	Starting Set_MCU_Signal(signal_id=VDDIU_33VB, state=Logicu)		SYSREF Pulser
12:08:36.482	EVB	Starting Set_MCU_Signal(signal_id=VDDIO_ENB, state=Logic0)		Pulse
12:08:36.519	EVB	Starting Set_SPI_Speed(1 MHz)		
12:08:36.482	EVB	Starting Set_Voltage_Regulator_Enable(regulator=VDDIO, enabled=True)		Phase Readout
12:08:36.466	EVB	Starting Set_Voltage_Regulator_Level(regulator=VDDIO, voltage=V3P30)		
	Other	Starting status register polling		Die Temperature
12:24:02.983				

Figure 62. Accessing the DUT Dump Tool

CB Create DUT Dump File	_	×
Reading DUT Contents		
Reading status registers		
Cancel		

Figure 63. Waiting for the Log Collection to be Complete

# 6. Appendix A. Troubleshooting

# 6.1. Why Can't I Communicate with the Device on My Hardware Using the CBPro Field Programmer?

There are multiple windows in the CBPro software that use or provide communication to the device connected to the CBPro Field Programmer. The examples below show the windows and type of errors you may encounter. All of these situations can be resolved using the following steps.

## 6.1.1. General Steps for Resolving Communication Issues

- 1. Verify which communication protocol your hardware is using SPI or I<sup>2</sup>C. If using SPI, check whether it is 3wire or 4-wire SPI.
- 2. For Si534x/8x/9x devices if using I<sup>2</sup>C, use a multimeter and measure the voltage on the I2C\_SEL control pin on the DUT. The voltage should be logic low (0 V) if your communication protocol is SPI. This level should be logic high (1.8 V or 3.3 V refer step 3 below) if your communication protocol is I<sup>2</sup>C. I2C\_SEL is not a pin on the Si536x/Si540x/Si55xx devices, so this point can be ignored for those devices.
- 3. For Si534x/8x/9x devices, verify the value of the IO\_VDD\_SEL bit (Register 0x0943[0]) for the DUT. If IO\_VDD\_SEL is 0, the I/O Voltage setting should be 1.8V. If IO\_VDD\_SEL is 1, the I/O Voltage setting should 3.3V. If you do not know this value, you can try both voltages to determine which voltage level will work successfully.
- 4. For Si536x/Si540x/Si55xx devices, check that the communication protocol selected under the HOST INTERFACE page in CBPro matches the protocol the host (master) device is using.
- 5. For Si534x/8x/9x devices, if the communication protocol is I<sup>2</sup>C, verify the I<sup>2</sup>C address setting (Register 0x000B) for the device. You may also need to verify the voltage level on the A0/CSb and A1/SDO pins if they are not connected to the field programmer. The level on these pins set bit 1 and bit 0 in the I<sup>2</sup>C address. If these are connected to the CBPro Field Programmer, they are both driven low.
- 6. For Si536x/Si540x/Si55xx devices, make sure the regulators used to power the device can supply enough current. On the evaluation boards, make sure the external 5VDC adapter is plugged in.

## 6.1.2. General Steps for Resolving Communication Issues (Si5383/4/8/9)

- 1. Verify the I<sup>2</sup>C address for the device.
- 2. Verify the voltage level on the A0/CSb and A1/SDO pins if they are not connected to the field programmer. The level on these pins set bit1 and bit 0 in the l<sup>2</sup>C address. If these are connected to the CBPro Field Programmer, they are both driven low.

### 6.1.3. Communication Error Using the Design Dashboard Window

If the design dashboard experiences an error communicating the device, the following error window will appear. Errors communicating to the device can be due to the device not being powered on, or the selected communication protocol does not match what is expected by the device.



Figure 64. Communication Error Using Design Dashboard

This example window shows how to adjust the communication settings of the dashboard to resolve communication error.



Figure 65. Design Dashboard Communication Error Solution

## 6.1.4. USB Communication Error

You may encounter a USB related error. To solve this, close out all cbpro instances, unplug the USB cable, and reopen CBPro and plug in the USB cable once CBPro is loaded.



Figure 66. USB Error Message

# 6.1.5. Communication Error Using the Burn NVM Window

The following window shows a communication error in the NVM Burn window. This error can appear after the Scan for Device button is pressed if the incorrect communication protocol was selected.

B NVM Program Tool - Cloc	kBuilder Pro v4.1		_		×		
Field Programmer Mode:	Wired to Board (No Socket Detected)						
Target Device:	Si534x/7x/8x/9x (not firmware based)						
Host Interface:	I2C Address 0x68 / 104d; 400 kHz; 3.3V						
Project File:	ClockBuilder Pro v4.1 –	×	Project .	C	lear		
Project File Created E Project Part:	Scan Error		OPN Lo	okup			
Project Design ID:	There was an error trying to scan field programmer state:						
Project Design Check	Set_GPO(port_index=4, pin_index=4, state=Logic1): timeout afte 1500 msec	r					
Project File NVM Has	ОК		_				
Device Part Number:		Jun	-or Device	e Cl	ear		
Device Design ID:							
Device NVM State:							
# Valid Burns:	0						
# Burns with Error:	0						
	Program NVM						

Figure 67. Burn NVM Error Message

To address the error, select the correct communication protocol the device is configured for. If  $I^2C$  is used, check all the addresses between 0x68 and 0x71 (the allowable range of addresses set via hardware pin). It is also possible that the  $I^2C$  address was programmed to a number outside the normal range using the  $I^2C$  Address burn tool. The following window shows the dropdown used to adjust the communication settings to resolve communication error.

CB NVM Program Tool - Clock	kBuilder Pro v4.1		_		×
Field Programmer Mode:	Wired to Board (	No Socket Detected)			
Target Device:	Si534x/7x/8x/9x	(not firmware based)			
Host Interface:	I2C Address 0x6	8 / 104d; 400 kHz; 3.3V 🛛			
Project File: Project File Created By: Project Part:	Protocol	SPI 4-Wire SPI 3-Wire I2C	Select Project	ct C	lear
Project Design ID: Project Design Check: Project File NVM Hash:	I/O Voltage	3.3 V			
Device Part Number: Device Design ID: Device NVM State:	I2C Address	7-bit address, range 1-127 400 kHz	Scan for Dev	vice	ear
# Valid Burns: # Burns with Error:	0				
		Program NVM			

Figure 68. Burn NVM Error Message Solution

# 6.1.6. Communication Error Using the EVB GUI Window

The following window shows an example of the error produced when the EVB GUI experiences an I<sup>2</sup>C error.

CB Field Programmer - ClockBuilder Pro			-		×
File Help					
Info DUT Settings Editor DUT Register Editor Status		Field Progra	mmer		
Unable to identify the device. Check field programmer options and connection.		Family: Target: Interface:	Si534x/7 firmware Wired to I2C Addro 104d; 400	k/8x/9x (r based) PCB (seri ess 0x68 0 kHz; 3.3	ial) / 3V
		Design ID	-ERR-		
		C	onfig	Scan	
		Device Cont	trol / Misc		
		Unable to i	identify the	device. (	Check f
Log					
Filtered         Auto Scroll: On         Insert Marker         Clear         Copy to Clipboard         Pause					
Timestamp Source Message					
10.33.43.095 EVD Starting Read_DOT_byte(address=0x0000)					
10:55:45.731 EVB error Read_DUT_Byte(address=0x0000) => I2C_Write(i2c_bus=1, i2c_slave_address=0x68, data=0x0100): operation failed on MCU; error code 0xF (general failure)	A				
10:55:45.731 EVB Starting Read_DUT_Byte(address=0x026B)					
10:55:45.762 EVB error Read_DUT_Byte(address=0x0268) => I2C_Write(i2c_bus=1, i2c_slave_address=0x68, data=0x0102): operation failed on MCU; error code 0xF (general failure)	A 💡	1			ŀ
EVB Firmware 0.84    Field Programmer: Wired to PCB (serial); Si534x/7x/8x/9x (not firmware based); I2C Address 0x68	/ 104d; 40	00 kHz; 3.3V			

Figure 69. EVB GUI I<sup>2</sup>C Error

The following window shows an example of the error produced when the EVB GUI experiences an SPI error. An error in communicating through SPI can occur due to many things, including:

- Incorrect connection of SPI wires.
- SPI host is communicating at a voltage level not expected by the clock device.
- I<sup>2</sup>C pullup resistors being present on a custom board.
- Long traces or heavy capacitive loads on a custom board leading to errors in high-frequency SPI communications.
- If the Field Programmer is being used with an evaluation board, double check the schematic for any missing (or extra) 0  $\Omega$  resistors in the path. On some newer devices, it is possible for a SPI pin to double as a GPIO pin.

CB Field Progra	mmer - Clock	Builder Pro		-		×
File Help						
Info DUT Se	ttings Editor	DUT Register Editor Status	Field Progra	mmer		
Unable to ide	ntify the devi	ce. Check field programmer options and connection.	Family: Target: Interface: Part Numbe Design ID Co Device Cont Unable to in	Si534x/7x,/ firmware I Wired to F SPI 3-Wire r -ERR- ??????? onfig S rol / Misc dentify the e	/8x/9x (r pased) PCB (seri e; 1 MHz can device. C	iot al) ; 3.3V
Log						
Filtered	Auto Scrol	l: On 💽 Insert Marker Clear Copy to Clipboard Pause				
Timestamp	Source	Message				
10:52:04.757	EVB	Finished Kead_DUT_Byte(address=UXU26F) => UXFF				
10:52:04.757	EVB	Starting Read_DUT_Byte(address=0x0270)				
10:52:04.773	EVB	Finished Read_DUT_Byte(address=0x0270) => 0xFF				
10:52:04.773	EVB	Starting Read_DUT_Byte(address=0x0271)				
10:52:04.820	EVB	Finished Read_DUT_Byte(address=0x0271) => 0xFF				
10:52:04.820	EVB	Starting Read_DUT_Byte(address=0x0272)				
10:52:04.842	EVB	Finished Read_DUT_Byte(address=0x0272) => 0xFF	•			Þ
EVB Firmware 0.8	84    Field Prog	grammer: Wired to PCB (serial); Si534x/7x/8x/9x (not firmware based); SPI 3-Wire; 1 MHz; 3.3V	Clock	Builder Pro v	4.1 [2021	-09-22]

Figure 70. EVB GUI SPI Error

The following window shows how to change the communication settings using the EVB GUI window. Selecting the correct communication protocol, and double checking the signal at the SPI pins with an oscilloscope should resolve the -ERR- message.

CB Field Progra	mmer - Clocki	Builder Pro									_		×
File Help													
Info DUT Se	ttings Editor	DUT Register Editor	Status					~	Field Prog	grammer			
Unable to ide	ntify the devi	ce. Check field progra	immer op	otions and co	onnection.				Family: Target:	Si534 firmw Wired	x/7x/ vare b d to F	/8x/9x (i based) PCB (ser	not ial)
									Part Num Design ID	ber	- vvire	e; 1 MH2	2; 3.3V
										Config	S	can	
							Device Family	Si	534x/7x/8x	/9x (not f	irmw	are base	ed) 🔽
							Protocol	0	SPI 4-Wire SPI 3-Wire				
Log								$\bigcirc$	12C				
Filtered	Auto Scrol	l: On 🔽 🛛 Insert Ma	arker	Clear	Copy to Clip	oboard	I/O Voltage	3.	3 V				
Timestamp	Source	Message											
10:52:04.757	EVB	Starting Read_DUI_E	syte(addr	ess=0x026F)	) => 0xFF		SPI Bus Speed	1	MHz				
10:52:04.737		Einiched Read DUT	yte(addr	acc=0x0270)							_		
10.52.04.775		Starting Read DUT R	yte(addr	ess = 0x0270	) = > 0xFF								
10.52.04.775		Starting Read_DUT_B	yte(auun	255-0x0271)									
10.52.04.820	EVB	Starting Read DUT B	vte(addr	ess=0x0271)	/ = > 0AFF								
10:52:04:820	EVB	Einished Read DUT E	Byte(addr	ass=0x0272)				Ļ					
10.32.04.042	LVD	Thisney Read_DOT_E	sytelauui	C33-0X0272)	/ = > 0AFF			•	•				+
EVB Firmware 0.8	84 Field Proc	grammer: Wired to PCB	(serial); Si	534x/7x/8x/9	x (not firmware	based); SPI 4-	Wire; 1 MHz; 3.3V		Clo	ockBuilder	Pro v	4.1 [202	1-09-22

Figure 71. EVB GUI Solution

# 6.2. Why do I Have a Communication Error when I Write My New Project to the Si534x/8x/9x Device?

# 6.2.1. Description of What Happens when a New Plan Changes the IO\_VDD\_SEL Bit (Register 0x0943[0]) Value

In order for the CBPro Field Programmer to communicate with the device correctly, the Field Programmer's IO voltage needs to match the IO\_VDD\_SEL bit in the device. CBPro may force this bit when writing a new plan to a device. If the plan changes this bit during the writing process, communication can fail. To determine if the new plan is changing this bit, perform the following steps:

- Read the current IO\_VDD\_SEL value in the device by using the DUT Register Editor tab in the EVB GUI window.
- Check if IO\_VDD\_SEL read above is different from the settings in the Host Interface tab in the Design Dashboard of the new project.
  - If VDD (Core) radio button selected and 0x943 = 0, no change from new plan,

Else VDD (Core) radio button selected and 0x943 = 1, new plan is changing IO\_VDD\_SEL refer to "6.3. How do I write a project file to the device that changes the I/O Power Supply setting in Si534x/8x/9x devices (IO\_VDD\_SEL bit)?" on page 52.

- If VDDA (3.3 V) radio button selected and 0x943 = 1, no change from new plan,

Else VDDA (3.3 V) radio button selected and 0x943 = 0, new plan is changing IO\_VDD\_SEL refer to "6.3. How do I write a project file to the device that changes the I/O Power Supply setting in Si534x/8x/9x devices (IO\_VDD\_SEL bit)?" on page 52.

The following window shows how to read the IO\_VDD\_SEL bit from the device.

CB Field Programmer - ClockBuilder Pro		-		×
File Help 1				
Info DUT Settings Editor DUT Register Editor Status	Field Program	imer		
Register Peek/Poke	Family:	Si534x/7x/ firmware b	'8x/9x ( ased)	not
2 Hex Decimal Address: 0x0943 2.371	Target:	Socket, Si5 QFN44	34x/7x	(/9x-
	Socket Power	: On		
# Bytes: 1 Read Write	Interface:	I2C Addres 104d; 400	ss 0x68 kHz; 3.	3V
Hey: Ov01	Part Number	Si5342H-D	-GM	
	Design ID	5342EVB3		
Unsigned Int:				
	Cor	ifig Sca	an	
	Device Contro	ol / Misc		
Log	Soft Rese	t and Calib	ration	â
Filtered 💽 Auto Scroll: On 💽 Insert Marker Clear Copy to Clipboard Pause	SO	FT RST ALL		
Timestamp Source Message				5
12:32:14:009 EVB Finished Kead_DUT_Byte(address=0X0008) => 0X08		OFT_RST		
12:32:58.596 EVB Starting Read_DUT_Bytes(address=0x0943, num_bytes=1)	Hard Peset	Sync & Pow	er Do	
12:32:58.632 EVB Finished Read_DUT_Bytes(address=0x0943, num_bytes=1) => 0x00	Hard Reset,	sync, a row		
12:34:19.905 EVB Starting Read_DUT_Bytes(address=0x0943, num_bytes=1)	н	ARD_RST		
12:34:19.925 EVB Finished Read_DUT_Bytes(address=0x0943, num_bytes=1) => 0x00		SYNC		
12:34:20.669 EVB Starting Read_DUT_Bytes(address=0x0943, num_bytes=1)	PD	N: 0		
12:34:20.703 EVB Finished Read_DUT_Bytes(address=0x0943, num_bytes=1) => 0x00	]			_
EVB Firmware 0.84    Field Programmer: Socket, Si534x/7x/9x-QFN44; Si534x/7x/8x/9x (not firmware based); I2C Address 0x68 / 10-	04d; 400 kHz; 3.3V			

Figure 72. Read IO\_VDD\_SEL Bit from Device

The following window shows how to determine the value of the IO\_VDD\_SEL bit that will be written to the device from the project file.



Figure 73. Determine the Value of IO\_VDD\_SEL Bit Written to Device

# 6.2.2. Fix: Changing the IO\_VDD\_SEL Bit

If it is suspected that IO\_VDD\_SEL has changed after writing a new plan to the device, you will not be able to communicate with the device. However, the device will still respond to a register write request to change the IO\_VDD\_SEL bit even when communicating with the incorrect IO\_VDD. To do this, follow the steps below:

- 1. Power down and power on the clock device to ensure that the next write operation is the first bus transaction.
- 2. Write a 9 (0x9) to the page address register (address=0x1)
- 3. Write a value of 0x00 to register 67 (0x43). Steps 2 and 3 combined will write a 0 value to register address=0x0943.
- 4. Read back some registers to check if communications is now restored. A suggestion is to read registers 0x02 and 0x03. These registers should be non-zero and correspond to the device part number.

# 6.3. How do I write a project file to the device that changes the I/O Power Supply setting in Si534x/8x/9x devices (IO\_VDD\_SEL bit)?

# 6.3.1. General Steps to Change I/O Power Supply Setting with a Project File

In order for the field programmer to communicate with the device correctly, the field programmer's IO voltage needs to match the IO\_VDD\_SEL bit in the device and use the correct serial communication protocol to match the I2C\_SEL pin on the device. This is not automatically detected by the GUI or the CLI command.

If the new project changes the IO\_VDD\_SEL bit, the following summarized steps need to be performed. The flow chart and figures that follow provide the details for each of these steps. There are detailed steps using CBPro Graphical User Interface and detailed steps using the CBPro Command Line interface.

- 1. Establish communication with the device to be programmed and determine the current value of the IO\_VDD\_SEL (0x0943[0]) bit.
- 2. The current value of the IO\_VDD\_SEL bit matches the value of the new plan to be written to the device?
- Yes Proceed to step 3.
- No Change the IO\_VDD\_SEL bit to match the value in the new plan. Re-establish communication with the device after changing the IO\_VDD\_SEL value (change the field programmer I/O Voltage to match new value for IO\_VDD\_SEL).
- 3. Write the new plan to the device.



Figure 74. General Steps to Change I/O Power Supply Setting with a Project File

## 6.3.2. Steps Using CBPro Graphical User Interface

1. Select the "EVB GUI" button on the home screen as shown to attempt communication with the device.



Figure 75. EVB GUI Button

- a. Select the "DUT Register Editor" tab.
- b. Determine the correct device communication protocol and setup CBPro accordingly as shown. For an Insocket device, click the Socket Power slider to power up the device. For In-system devices, click the Device Family pulldown and select the appropriate device family.
- c. Click the Scan button to verify communication with the device.
- d. If communication is successful, the device part number and design ID will be updated. If communication is not successful, the part number field will display -ERR- and the DUT register tab will be disabled.
   Configuring communication settings are shown in "Figure 76. Configuring Communication Settings" on page 55.

Field Brogram	mor
Field Program	imer
Family:	Si534x/7x/8x/9x (not firmware based)
Target:	Socket, Si534x/7x/9x-
	QFN44
Socket Power	: Off
Interface:	SPI 4-Wire; 1 MHz;
	1.8V
Part Number	
Design ID	
Cor	nfig Scan
D Protocol	SPI 4-Wire
	O SPI 3-Wire
	O 12C
ų .	
I/O Voltag	je 1.8 V 🔽
-	
F SPI Bus Sp	beed 1 MHz
F	IARD_RST
	SVNC
	STINC
PD	N: 0

Figure 76. Configuring Communication Settings

## **USER GUIDE**

Examples of a Communication failure for  $I^2C$  and SPI:

CB Field Programmer - Cloc	kBuilder Pro								-		$\times$
File Help											
Info DUT Settings Editor	DUT Register Editor	Status					~	Field Prog	rammer		
Unable to identify the dev	ice. Check field progr	ammer op	otions and	connection	1.			Family: Target: Interface: Part Numb Design ID Device Co Unable to	Si534x/ firmwai Wired 1 I2C Adu 104d; 4 Der -ERR- -ERR- Config	7x/8x/9x ( re based) o PCB (se dress 0x68 00 kHz; 3. Scan	not ial) / 3V
log											
Filtered     Auto Scrophic       Timestamp     Source       10:55:45.731     EVB	II: On Tinsert M Message Starting Read_DOT_ error Read_DUT_Byt i2c_slave_address=0 (general failure)	larker	Clear ess=0x0000 =0x00000) = =0x0100): (	Copy to 00) => I2C_Writ operation fa	to Clipboard ite(i2c_bus=1, failed on MCL	Pause	xFA				
10:55:45.731 EVB	Starting Read_DUT_I	Byte(addr	ess=0x026	δB)							
10:55:45.762 EVB	error Read_DUT_Byt i2c_slave_address=0	e(address 1x68, data:	=0x026B) =0x0102):	=> I2C_Wri	ite(i2c_bus=1	l' error code 0	xFA				
EVB Firmware 0.84    Field Pro	grammer: Wired to PCE	B (serial); Si	i534x/7x/8x	x/9x (not firm	nware based);	I2C Address 0x6	▼ 58 / 104d; 40	4 ⊂ 0 kHz; 3.3V			•
EVB Firmware 0.84    Field Pro	ogrammer: Wired to PCE	B (serial); Si	i534x/7x/8x	x/9x (not firm	nware based);	I2C Address 0x6	▼ 58 / 104d; 40	0 kHz; 3.3V			,
EVB Firmware 0.84    Field Pro	egrammer: Wired to PCE	8 (serial); Si	i534x/7x/8x	x/9x (not firm	nware based);	I2C Address 0x6	▼ 58 / 104d; 40	40 kHz; 3.3V	_		, ×
EVB Firmware 0.84    Field Pro B Field Programmer - Clock File Help Info DUT Settings Editor	general failule) ogrammer: Wired to PCE Builder Pro DUT Register Editor	B (serial); Si Status	i534x/7x/8x	x/9x (not firm	nware based);	I2C Address 0x6	58 / 104d; 40	0 kHz; 3.3V	- ammer		×
EVB Firmware 0.84    Field Pro B Field Programmer - Clock File Help Info DUT Settings Editor Unable to identify the devi	grammer: Wired to PCE Builder Pro	8 (serial); Si Status ammer op	i534x/7x/8x	x/9x (not firm	nware based);	12C Address 0xd	58 / 104d; 40	Field Progr Family: Target: Interface: Part Numb Design ID C Device Cor Unable to	ammer Si534x// firmwar Wired to SPI 3-W er -ERR- ???????? Config Config trol / Misc identify th	/x/8x/9x (r e based) > PCB (ser ire; 1 MHz Scan	, x oot al) 3.3V
EVB Firmware 0.84    Field Pro CB Field Programmer - Clock File Help Info DUT Settings Editor Unable to identify the devi	graemen landre) Igrammer: Wired to PCE Builder Pro DUT Register Editor ce. Check field progra	8 (serial); S Status ammer op	i534x/7x/8x	connection.	nware based);	12C Address 0xd	58 / 104d; 40	Field Progr Family: Target: Interface: Part Numb Design ID C Device Cor Unable to	ammer Si534x/7 firmwart SPI 3-W er -ERR- ???????? Config	C (x/8x/9x (r = based) > PCB (ser ire; 1 MHz Scan	x oot al) 3.3V
EVB Firmware 0.84   Field Pro EVB Field Programmer - Clock File Help Info DUT Settings Editor Unable to identify the devi Duable to identify the devi Filtered Auto Scro Timestamp Source	Igneral Island() Igrammer: Wired to PCE Builder Pro DUT Register Editor ce. Check field progra ce. Check field progra Island () Island () Insert M. Message	8 (serial); S Status ammer op	tions and o	connection.	nware based); o Clipboard	I2C Address 0xd	v 104d; 40	Field Progr Family: Target: Interface: Part Numb Design ID	ammer Si534x// firmwar Wired ta SPI 3-W er -ERR- ???????? Config	Cx/8x/9x (r = based) o PCB (ser ire; 1 MHz Scan	× ot al) i 3.3V
EVB Firmware 0.84    Field Pro GB Field Programmer - Clock File Help Info DUT Settings Editor Unable to identify the devi Diable to identify the devi Filtered P Filtered Auto Scro Timestamp Source	I: On I Insert M. Message	a (serial); Si Status mmer op	Clear	x/9x (not firm connection.	nware based);	I2C Address 0xd	× 104d; 40	Field Progr Family: Target: Interface: Part Numb Design ID	ammer Si534x/3 firmwar Wired tt SPI 3-W er -ERR- ???????? Config	v/8x/9x (9x (9x (9x (9x (9x (9x (9x (9x (9x (	v ot al) : 3.3V
EVB Firmware 0.84   Field Pro B Field Programmer - Clock File Help Unable to identify the devi Unable to identify the devi Filtered Auto Scro Timestamp Source 10:52:04.757 EVB 10:52:04.757 EVB	I: On V Insert M. Message Finished Read_UUT_E Starting Read_UUT_E	8 (serial); S: Status Status arker op	Clear دور میں کوئی کی کوئی کوئی کوئی کوئی کوئی کوئی ک	<pre>x/9x (not firm connection.</pre>	nware based);	22C Address 0xd	58 / 104d; 40	Field Progr Family: Target: Interface: Part Numb Design ID	ammer SIS34x// firmwar SPI 3-W er -ERR- 22222222 Config Config Config trtrd / Misc	l (x/8x/9x (t e based) PCB (ser PCB (ser PCB (ser Scan	intervention of the second sec
Log Filed Programmer - Clock File Help Unable to identify the devi Info DUT Settings Editor Unable to identify the devi Filtered Auto Scroo Timestamp Source T0:52:04.757 EVB 10:52:24.773 EVB	I: On V Insert M. Message Finished Read_DUT_8 Starting Read_DUT_8 St	8 (serial); S: Status arker ( syte(addre syte(addre yte(addre yte(addre	Lis34x/7x/8x Lis34x Lis34x/7x/8x Lis34x/7x	<pre>x/9x (not firm connection. ) Copy to r) =&gt; 0xFF 0) 0) =&gt; 0xFF 1)</pre>	o Clipboard	22C Address 0xd	58 / 104d; 40	Field Progr Family: Target: Interface: Part Numb Design ID C Device Cor Unable to	ammer SIS34x/17 firmwar SPI 3-W SPI 3-	c x/8x/9x (t based) PCB (ser PCB (ser Scan	iot al) : 3.3V
Log File Help Unable to identify the devi Filestered Auto Scroo Timestamp Source 10:52:04.737 EVB 10:52:04.773 EVB 10:52:04.773 EVB	Igneral land(c) Igneral land(	8 (serial); S; Status immer op arker ( ayte(addre yte(addre yte(addre yte(addre yte(addre	E534x/7x/8x tions and c Clear ess=0x0270 ess=0x0277	<pre>x/9x (not firm connection. ) Copy to F) =&gt; 0xFF 0) 0) =&gt; 0xFF 1) 1) =&gt; 0xFF</pre>	nware based);	22C Address 0xt	\$8 / 104d; 40	Field Progr Family: Target: Interface: Part Numb Design ID C Device Cor Unable to	ammer Si534x Wired tt SPI 3-W ZP272727 Config	'x/8x/9x (k b based) > PCB (ser ire: 1 MHz Scan	× al) 3.3V
EVB Firmware 0.84   Field Pro Field Programmer - Clock File Help Info DUT Settings Editor Unable to identify the devi Piltered Auto Scro Tinestamp Serve T0:52:04.757 EVB 10:52:04.757 EVB 10:52:04.757 EVB 10:52:04.757 EVB 10:52:04.757 EVB	I: On IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	8 (serial); S; Status immer op syte(addre byte(addre byte(addre byte(addre	534x/7x/8x tions and c clear ss=0x027c ss=0x027c ss=0x027c ss=0x027c	<pre>x/9x (not firm connection. r) =&gt; 0xrF 0) 0) =&gt; 0xFF 1) 1) =&gt; 0xFF 2)</pre>	o Clipboard	I2C Address 0xf	58 / 104d; 40	Field Progr Family: Target: Interface: Part Numb Design ID C Device Cor Unable to	ammer SiS34X, Wired ta SPI 3-W Vired ta SPI 3-W VI SPI 3-W SPI 3-W VI SPI 3-W VI SPI 3-W VI SPI 3-W VI	'x/8x/9x (k b based) > PCB (ser rire: 1 MH2 Scan	x oot al) 3.3V

Figure 77. I<sup>2</sup>C and SPI Communication Failure Examples

- 2. Match the IO\_VDD\_SEL bit to the value in the plan that will be written to the device.
  - a. If the IO\_VDD\_SEL bit already matches the value in the plan to be written, skip to Step 3.
  - b. If the IO\_VDD\_SEL bit is not correct, change the value and write the new value to the device (see the figure below).
  - c. Reconfigure the communication settings of the field programmer to reestablish communication to the device.

CB Field Programmer - ClockBuilder Pro			_		×
File Help 1					
Info DUT Settings Editor DUT Register Editor Status	-	Field Program	mer		
Register Peek/Poke	Î	Family:	Si534x/7x firmware	(/8x/9x ( based)	not
4 Hex Decimal		Target:	Socket, S QFN44	i534x/7x	(/9x-
		Socket Power:	On	2	
* Bytes: Read Write 6		Interface:	SPI 4-Wir 3.3V	e; 1 MH	Z;
Hex: 0x01		Part Number	Si5342H-	D-GM	
Unsigned Int: 1		Design ID	5342EVB.	3	
7     6     5     4     3     2     1     0       Binary:     0     0     0     0     0     1     5		Con	fig S	an	3
	-	Device Contro	l / Misc		
Log Filtered 🔹 Auto Scroll: On 💽 Insert Marker Clear Copy to Clipboard Pause		Soft Rese	t and Calil	oration L	
Timestamp Source Message					
12:59:00.964 EVB Finished Kead_DUT_Byte(address=0x0006) => 0x00		5	OFT_KST		
12:59:00.964 EVB Starting Read_DUT_Byte(address=0x0007)		Hard Reset	Sync & Po	wer Dov	
12:59:01.027 EVB Finished Read_DUT_Byte(address=0x0007) => 0x01					
12:59:01.027 EVB Starting Read_DUT_Byte(address=0x0008)		H.	ARD_KST		
			SYNC		
12:59:01.080 EVB Finished Read_DUT_Byte(address=0x0008) => 0x08				_	
12:59:01.080         EVB         Finished Read_DUT_Byte(address=0x0008) => 0x08           12:59:09.734         EVB         Starting Read_DUT_Bytes(address=0x0943, num_bytes=1)		PDN	N: 0		

Figure 78. Reconfiguring Communication Settings of the Field Programmer

3. Write your new plan to the device.

CB	Field Programmer - ClockBuilder Pro									
File	Help									
	Write Project File to Device Status									
	Write Settings File to Device									
	Write Register File to Device									
	Create DUT Dump File for Skyworks Support									
	Preferences									
	Exit									
7	Kead Write									
ł	Hex: 0x01									

Figure 79. Write New Plan to Device

### 6.3.3. Steps Using CBPro Command Line Interface

1. Attempt to communicate with the Si534x/8x/9x device and determine the current value of the IO\_VDD\_SEL bit.

SPI communication Examples:

CBProDeviceRead.exe --io-voltage 1.8 --mode spi4wire --speed 1M --family si538x4x --registers 0x0943 CBProDeviceRead.exe --io-voltage 3.3 --mode spi4wire --speed 1M --family si538x4x --registers 0x0943

# **Note:** The commands above are examples. Refer to the document and help for the CBPro CLI for your specific configuration.

### I<sup>2</sup>C communication Examples:

```
CBProDeviceRead.exe --io-voltage 1.8 --mode i2c --speed 100k --i2c-address 0x68 --family si538x4x --
registers 0x0943
```

```
CBProDeviceRead.exe --io-voltage 3.3 --mode i2c --speed 100k --i2c-address 0x68 --family si538x4x --
registers 0x0943
```

- **Note:** The commands above are examples. Refer to the document and help for the CBPro CLI for your specific configuration.
  - 2. Match the IO\_VDD\_SEL bit to the value in the plan that will be written to the device.
  - a. A simple text file will need to be created that will write register 0x943 to 0x00 or 0x01.
    - To write 0x01 to 0x0943, the text file should contain the following single line of text:

#### 0x0943,0x01

To write 0x00 to 0x0943, the text file should contain the following single line of text:

#### 0x0943,0x00

b. Run the CLI command below to change the IO\_VDD\_SEL bit.

#### SPI Example:

```
CBProDeviceWrite.exe --mode spi4wire --speed 4M --io-voltage 3.3 --family si538x4x --registers simple text file.txt
```

### I<sup>2</sup>C Example:

```
CBProDeviceWrite.exe --mode i2c -i2c-address 0x68 --speed 400K --io-voltage 3.3 --family si538x4x --
registers simple_text_file.txt
```

# **Note:** The commands above are examples. Refer to the document and help for the CBPro CLI for your specific configuration.

### 3. Write the new plan to the part.

### SPI Example:

```
CBProDeviceWrite.exe --mode spi4wire --speed 4M --io-voltage 3.3 --family si538x4x --project your_plan_name.slabtimeproj
```

### I<sup>2</sup>C Example:

CBProDeviceWrite.exe --mode i2c -i2c-address 0x68 --speed 400K --io-voltage 3.3 --family si538x4x -project your\_plan\_name.slabtimeproj

# **Note:** The commands above are examples. Refer to the document and help for the CBPro CLI for your specific configuration.

# 6.4. I Burned a Project File to my Device with a New Base I<sup>2</sup>C Address, but the Base Address in the Device was Not Changed after the Burn Process was Complete

The I<sup>2</sup>C address will not be changed during the burn process. Changes to the base I<sup>2</sup>C address in the CBPro Configuration Wizard will be included in exports and the project file used to create orderable part numbers. However, this change is not burned to the device using the NVM Burn Tool. See the note highlighted in Figure 80.

Base I2C Address	
the A0 and A1 pins on t	2C address are 2-bits are controlled using he Si5342.
Address:	
6 5 4 3	2 1 0
1 1 0 1	0 A1 A0
Address Range: 104d to	o 107d / 0x68 to 0x6B
Host interface re- - IO_VDD_SEL, SP are not written to	gisters defined by this page PI_3WIRE, and I2C_ADDR – > EVBs. They are included in
exports and orde	rable part number non-
Manual for more	information.

Figure 80. Base I<sup>2</sup>C Address

To permanently change the  $I^2C$  base address on your device, use the  $I^2C$  Address Burn Tool. See Figure 81 below.

- 🗆 ×



CB Si538x/4x I2C Address Burn Tool

Figure 81. I<sup>2</sup>C Address Burn Tool



# 7. Appendix B. Field Programmer Schematic

Figure 82. CBPROG-DONGLE Schematic (1 of 3)



Figure 83. CBPROG-DONGLE Schematic (2 of 3)



Figure 84. CBPROG-DONGLE Schematic (3 of 3)



Figure 85. 64-Pin Socket Board Schematic



Figure 86. 56-Pin Socket Board Schematic



Figure 87. 44-Pin Socket Board Schematic





Figure 88. Si5332 48-Pin Socket Board Schematic





Figure 89. Si5332 40-Pin Socket Board Schematic





Figure 90. Si5332 32-Pin Socket Board Schematic



Figure 91. Si55xx 72 pin Socket Board Schematic

# 8. Appendix C. Bill of Materials

# 8.1. Field Programmer Bill of Materials

NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB Footprint	ManufacturerPN	Manufacturer
		C2 C3 C8 C11 C26								
	c	C28 C30 C31 C32	1uF		16V	+10%	X7R	C0603	C0603X7R160-105K	Venkel
	-									· cinton
		C24 C27 C20	0.01.5		104	120%	V70	00000	COCO2Y701C0 10204	Verlat
	c	3 C24 C27 C29	0.010F		101	±20%	X/R	00603	C0603X7R160-103W	venkel
	1	L C39	33pF		25V	±10%	CUG	C0402	C0402C0G250-330K	Venkel
		C4 C6 C12 C13								
		C14 C35 C36 C37								
	9	C38	0.1uF		10V	±10%	X7R	C0402   C0402L	C0402X7R100-104K	Venkel
	2	C5 C25	4.7uF		10V	±20%	X7R	C1206	C1206X7R100-475M	Venkel
	1	(9	10uE		10V	+20%	X7R	C1206	C1206X7R100-106M	Venkel
	-	D14D1E		20-4	200	12070	~~~			UTE ON TECHNICLOCY CORD
	4	014015	choroop Aut	201174	2011		71.00			LITE-ON TECHNOLOGY CORP
	1	L D5	SP0503BAH I	300mW	200		105	SOT143-AKKK SOT143	SP0503BAH IG	Littletuse
		D6 D7 D8 D9 D10								
	7	7 D11 D12	GL05T-E3-08	5A	11V		Dual Common Anode	SOT23-123	GL05T-E3-08	Vishay
	1	FB1	22 Ohm	6000mA			SMT	L0805	BLM21PG220SN1	MuRata
	1	J2	USB Type B				USB	CONN-USB-B	61729-0010BLF	FCI
	1	13	ENCLOSURE					Ν/Δ	Emulator7045	Shanahai Zhongxingda Electronics
	-	D1 D11 D12 D65	11/	1/161		+1%	ThickFilm	R0402	CP0402 16W/ 1001E	Vonkol
		AT ATT ATZ AUD	IN	1/10/		11/0	THICKFIIII	R0402	CR0402-10W-1001F	VEIIKEI
		R13 R16 R26 R27								
		R40 R41 R42 R43								
	10	R59 R67	10K	1/16W		±1%	ThickFilm	R0402 R0402L	CR0402-16W-1002F	Venkel
	4	R14 R15 R24 R25	2 49K	1/16W		+1%	ThickFilm	R0402	CR0402-16W-2491E	Venkel
	1	R78	1.0	1/16W		+1%	ThickFilm	R0402	CR0402-16W/-1R00E	Venkel
		n20	1.0	1/10/		11/0	THICKFIIIII	R0402	CR0402-1000-1R00F	VEIIKEI
		R29 R30 R31 R32								
	8	8 R35 R37 R38 R68	0	1A			ThickFilm	R0402 R0402L	CR0402-16W-000	Venkel
	1	L R3	100	1/16W		±1%	ThickFilm	R0402	CR0402-16W-1000F	Venkel
	1	L R33	4.42K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-4421F	Venkel
	1	R34	9.53K	1/16W		+1%	ThickFilm	R0402	CR0402-16W-9531F	Venkel
	1	R36	5.9K	1/16W		+1%	ThickFilm	R0402	CR0402-16W/-5901E	Venkel
		D D D D C C	5.5K	2/414/		+10/	Thiekfilm	R0402	CRC402-1000-55011	Venker Visbau Dala
	4	K35 K00	1.0	5/400		1170	INICKFIIM	R1210	CREWIZIOIROOFKEAHP	visnay bare
	2	2 R44 R47	8.06K	1/16W		±0.1%	±25PPM	R0402	TFCR0402-16W-E-8061B	Venkel
	2	2 R45 R48	17.4K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-1742F	Venkel
	2	R46 R49	21.5K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-2152F	Venkel
	2	R50 R51	20K	1/10W		±1%	ThickFilm	R0603	CR0603-10W-2002F	Venkel
	4	R55 856 857 858	402	1/16W		+1%	ThickFilm	R0402	CR0402-16W-4020E	Venkel
	1	P62	102	1/16W		+1%	ThickFilm	R04021 R04021	CR0402-16W/-4990E	Venkel
	-		455	1/1000		11/0	Thiskfilm	R0402 R0402E	CR0402-10W-43501	Venkel
	4	K03 K04	470	1/16//		15%	INICKFIIM	R0402	CR0402-16VV-4713	venkei
	2	2 R9 R10	10.0K	1/10W		±0.1%	±25PPM	R0603	ERA-3AEB103V	Panasonic
	2	2 SF1 SF2	BUMPER					RUBBER_FOOT_0.250"	SJ5382	3M
						_				
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
		2 SF3 SF4	SPACER					N/A	7363	Keystone Electronics
		1 U1	DS2431					SOJ6N4.45P1.27	DS2431P+	Maxim
		1 U12	TPS76201	100mA			LDO	SOT5N2.8P0.95	TPS76201DBV	ті
		2 11131114	TPS79501	500mA			LDO	DEN8N3 OP0 65E2 4X1 65	TPS79501DBBT	TI
		1  114	ADG719	SoomA			190	SOTEN2 800.05	ADG7198877	Analog Devices
		1 016	ADG/19		0.4.514		n ((	5016N2.8P0.95	ADG/19BR12	Analog Devices
		1 017	NLSV11244		.9-4.5V		Buffer	UDFN6N1P0.4	NLSV11244WUTBG	Un Semi
		1 U2	C8051F380				MCU	QFP48N9X9P0.5	CF380P1104AGQ	SiLabs
		2 U3 U8	ADG712					TSSOP16N6.4P0.65	ADG712BRU	Analog Devices
		1 U5	LTC4311		5.5V			SC70-6N2.1P0.65	LTC4311CSC6#TRMPBF	Linear Technology
		1 U6	PCA9517D				12C	SO8N6.0P1.27	PCA9517D	NXP
		1 X1	2X5 Header				Shrouded	CONN2X5-RA-SBH11	SBH11-PBPC-D05-RA-BK	Sullins Connector Solutions
Not Installed Co	mnonerte	- ^1					511000000		52.122   DI C DOD IN DI	salaria connector porutions
NOT INSTAILED CO	Quert'	Defe	Value	Detin	Valer	Televini	Tume	DCD Factoriat	Manufacture Phi	Manufacture
NI	Quantity	Keterence	value	Kating	voltage	Iolerance	Type	PCB_FOOTPrint	WanutacturerPN	Manufacturer
N		2 C10 C40	1uF	_	16V	±10%	X7R	C0603	CU603X7R160-105K	Venkel
NI		1 C7	0.1uF		10V	±10%	X7R	C0402 C0402L	C0402X7R100-104K	Venkel
NI		1 JP1	HEADER 4X1				Header	CONN-1X4	TSW-104-07-T-S	Samtec
NI		2 R60 R61	0	1A			ThickFilm	R0603	CR0603-16W-000	Venkel
		TP1 TP2 TP3 TP9								
NI		5 TP10	RED				Loon	TESTROINT	151-207-RC	Kohiconn
NU		1 TD7	BUUE	-	-		Loop	TESTFOINT	151-207-RC	Kobicom
INI NU		1 12/	BLUE	-		-	Loop	TESTPOINT	151-205-RC	Kubiconn
NI		1 TP8	BLACK			_	Loop	TESTPOINT	151-203-RC	Kobiconn
NI		1 U4	AD8628		5V		OPAMP	SOT23-5N	AD8628AUJ-R2	Analog Devices

# 8.2. Si538x4x-64SKT-DK Socket Board BOM

	0	Deferrer	Mathem	D-12-	24-12	<b>*</b> -1	-	DOD Bernarden	54	20
NI	Quantity	Reference	value	Rating	voitage	Tolerance	туре	PCB_FOOTPrint	WanutacturerPN	Manufacturer
		5 C1 C2 C3 C4 C5 C6	0.1uF		10V	±10%	X7R	C0402 C0402L	C0402X7R100-104K	Venkel
	:	2 C7 C8	10uF		10V	±20%	X7R	C1206	C1206X7R100-106M	Venkel
		1 R2	0	1A			ThickFilm	R0402 R0402L	CR0402-16W-000	Venkel
		4 SF1 SF2 SF3 SF4	BUMPER					RUBBER_FOOT_SMALL	SJ61A6	3M
	:	1 U1	64QFN-SKT, 9x9x0.5mm				QFN	QFN64N9X9P0.5-SKT-WELLS-CTI	790-42064-101G	Sensata
	:	1 U2	DS2431					SOJ6N4.45P1.27	DS2431P+	Maxim
		1 X1	2X5 FEMALE				CONN	CONN2X5-FRA-SFH11	SFH11-PBPC-D05-RA-BK	Sullins Connector Solutions
Not Installed Com	ponents									
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Type	PCB_Footprint	ManufacturerPN	Manufacturer
NI		1 R1	1K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-1001F	Venkel
NI		1 R3	20K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-2002F	Venkel
NI	;	3 TP1 TP2 TP3	RED				Loop	TESTPOINT	151-207-RC	Kobiconn
NI		2 TP4 TP5	BLUE				Loop	TESTPOINT	151-205-RC	Kobiconn
NI		1 TP6	BLACK				Loop	TESTPOINT	151-203-RC	Kobiconn

# 8.3. Si538x4x-56SKT-DK Socket Board Bill of Materials

NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
		C1 C2 C5 C6 C7 C8								
	7	C9	0.1uF		10V	±10%	X7R	C0402 C0402L	C0402X7R100-104K	Venkel
	2	C3 C4	10uF		10V	±20%	X7R	C1206	C1206X7R100-106M	Venkel
		R1 R2 R7 R8 R9								
	7	R10 R11	1K	1/16W		±1%	ThickFilm	R0603	CR0603-16W-1001F	Venkel
	2	R12 R13	0	1A			ThickFilm	R0603   R0603L	CR0603-16W-000	Venkel
	4	R3 R4 R5 R6	4.7K	1/10W		±1%	ThickFilm	R0603	CR0603-10W-4701F	Venkel
	4	SF1 SF2 SF3 SF4	BUMPER					RUBBER_FOOT_SMALL	SJ61A6	3M
	1	U1	56LGA-SKT, 8x8x0.5mm				LGA	QFN56N8X8P0.5-SKT-WELLS-CTI	790-42056-101G	Sensata
	1	U2	DS2431					SOJ6N4.45P1.27	DS2431P+	Maxim
	1	X1	2X5 FEMALE				CONN	CONN2X5-FRA-SFH11	SFH11-PBPC-D05-RA-BK	Sullins Connector Solutions
Not Installed Com	ponents									
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
NI	2	TP1 TP2	RED				Loop	TESTPOINT	151-207-RC	Kobiconn
NI	3	TP3 TP5 TP6	BLUE				Loop	TESTPOINT	151-205-RC	Kobiconn
NI	1	TP4	BLACK				Loop	TESTPOINT	151-203-RC	Kobiconn

# 8.4. Si538x4x-44SKT-DK Socket Board Bill of Materials

NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
	5	C1 C2 C3 C4 C5	0.1uF		10V	±10%	X7R	C0402 C0402L	C0402X7R100-104K	Venkel
	2	C6 C7	10uF		10V	±20%	X7R	C1206	C1206X7R100-106M	Venkel
	1	R2	0	1A			ThickFilm	R0402 R0402L	CR0402-16W-000	Venkel
	4	SF1 SF2 SF3 SF4	BUMPER					RUBBER_FOOT_SMALL	SJ61A6	3M
	1	U1	44QFN-SKT, 7x7x0.5mm				QFN	QFN44N7X7P0.5-SKT-WELLS-CTI	790-41044-101G	Sensata
	1	U2	DS2431					SOJ6N4.45P1.27	DS2431P+	Maxim
	1	X2	2X5 FEMALE				CONN	CONN2X5-FRA-SFH11	SFH11-PBPC-D05-RA-BK	Sullins Connector Solutions
Not Installed Components										
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
NI	1	R1	1K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-1001F	Venkel
NI	1	R3	182	1/16W		±1%	ThickFilm	R0402	CR0402-16W-1820F	Venkel
NI	3	TP1 TP2 TP3	RED				Loop	TESTPOINT	151-207-RC	Kobiconn
NI	3	TP4 TP5 TP6	BLUE				Loop	TESTPOINT	151-205-RC	Kobiconn
NI	1	TP7	BLACK				Loop	TESTPOINT	151-203-RC	Kobiconn
## 8.5. Si5332-48SKT-DK Socket Board Bill of Materials

Eval Boan	d Name	SI5332-485KT	1							
Revision		1.0	1							
1101101011		110	1							
CreationDate	Proto Rev	Released								
5/18/2017	1.0	o c								
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
		2 C3 C8	0.1uF		10V	±10%	X7R	C0402 C0402L	C0402X7R100-104K	Venkel
		2 C6 C7	10uF		10V	±20%	X7R	C1206	C1206X7R100-106M	Venkel
		4 SF1 SF2 SF3 SF4	BUMPER					RUBBER_FOOT_SMALL	SJ61A6	3M
		1 U1	48QFN-SKT, 6x6x0.4mm				QFN		790-62048-101G	Sensata
		1 U2	DS2431					SOJ6N4.45P1.27	DS2431P+	Maxim
		1 X1	2X5 FEMALE				CONN	CONN2X5-FRA-SFH11	SFH11-PBPC-D05-RA-BK	Sullins Connector Solutions
Not Installed Components										
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
NI		4 TP1 TP2 TP8 TP9	RED				Loop	TESTPOINT	151-207-RC	Kobiconn
NI		1 TP6	BLUE				Loop	TESTPOINT	151-205-RC	Kobiconn
NI		1 TP7	BLACK				Loop	TESTPOINT	151-203-RC	Kobiconn

## 8.6. Si5332-40SKT-DK Socket Board Bill of Materials

		-	-							
Eval Board Name		SI5332-40SKT								
Revision		1.0								
			_							
CreationDate	Proto Rev	Released								
5/18/20:	1.00	) C								
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
	2	C3 C8	0.1uF		10V	±10%	X7R	C0402 C0402L	C0402X7R100-104K	Venkel
	2	C6 C7	10uF		10V	±20%	X7R	C1206	C1206X7R100-106M	Venkel
	4	SF1 SF2 SF3 SF4	BUMPER					RUBBER_FOOT_SMALL	SJ61A6	зм
	1	. U1	40QFN-SKT, 6x6x0.5mm				QFN		790-42040-101G	Sensata
	1	. U2	DS2431					SOJ6N4.45P1.27	DS2431P+	Maxim
	1	. X1	2X5 FEMALE				CONN	CONN2X5-FRA-SFH11	SFH11-PBPC-D05-RA-BK	Sullins Connector Solutions
Not Installed Components										
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
NI	4	TP1 TP2 TP8 TP9	RED				Loop	TESTPOINT	151-207-RC	Kobiconn
NI	1	. TP6	BLUE				Loop	TESTPOINT	151-205-RC	Kobiconn
NI	1	. TP7	BLACK				Loop	TESTPOINT	151-203-RC	Kobiconn

Eval Board Name

Revision

## 8.7. Si5332-32SKT-DK Socket Board Bill of Materials

<b></b>	1.81		7							
Eval Board Name		SI5332-325K1	4							
Revision		1.0								
CreationDate	Proto Rev	Released								
5/18/2017	1.00	0								
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
	2	C3 C8	0.1uF		10V	±10%	X7R	C0402 C0402L	C0402X7R100-104K	Venkel
	2	C6 C7	10uF		10V	±20%	X7R	C1206	C1206X7R100-106M	Venkel
	4	SF1 SF2 SF3 SF4	BUMPER					RUBBER_FOOT_SMALL	SJ61A6	3M
	1	U1	32QFN-SKT, 5x5x0.5mm				QFN		790-42032-101G	Sensata
	1	U2	DS2431					SOJ6N4.45P1.27	DS2431P+	Maxim
	1	X1	2X5 FEMALE				CONN	CONN2X5-FRA-SFH11	SFH11-PBPC-D05-RA-BK	Sullins Connector Solutions
Not Installed Components										
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB_Footprint	ManufacturerPN	Manufacturer
NI	4	TP1 TP2 TP8 TP9	RED				Loop	TESTPOINT	151-207-RC	Kobiconn
NI	1	TP6	BLUE				Loop	TESTPOINT	151-205-RC	Kobiconn
NI	1	TP7	BLACK				Loop	TESTPOINT	151-203-RC	Kobiconn

## 8.8. Si55xx-72SKT-DK Socket Board Bill of Materials

SI55XX-72SKT 1.0

CreationDate	Proto Rev	Released	<b>.</b>							
8/20/2020	12.00	0 0	End product must b	e RoHS d	compliant					
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB Footprint	ManufacturerPN	Manufacturer
	1	L C1	220uF		6.3V	±20%	X5R	C1210	C1210X5R6R3-227MNE	Venkel
	1	C10	2.2uF		10V	±10%	X7R	C0603	C0603X7R100-225K	Venkel
	2	C12 C22	22uF		6.3V	±20%	X5R	C0603	C1608X5R0J226M080AC	TDK
	1	L C2	10uF		25V	±20%	TANT	C6032	T491C106M025ZT	Kemet
	8	C3 C13 C14 C15 C16 C17 C18 C19	1.0uF		6.3V	±10%	X5R	C0402 C0402L	C0402X5R6R3-105K	Venkel
	4	4 C4 C5 C7 C11	0.1uF		16V	±10%	X7R	C0402	C0402X7R160-104K	Venkel
	4	C6 C9 C20 C21	10uF		6.3V	±20%	X5R	C0603 C0603L	C0603X5R6R3-106M	Venkel
	1	L C8	0.01uF		25V	±10%	X7R	C0402	C0402X7R250-103K	Venkel
	1	L D1	10V	500mW	10V	5%	Zener	SOD-123	MMSZ4697T1G	On Semi
	3	D2 D3 D4	Green	20mA	3.4V		SMT, ChipLED	LED-HSMX-C170	HSMQ-C170	Avago Technologies
	1	L J1	CONN TRBLK 2	24A			TERM BLK	CONN-TB-1711026	1711026	PHOENIX CONTACT
	1	L J2	Power Jack	2.5A			BARREL	CONN-3-PWR PJ-002A	PJ-002A	CUI
	1	L J3	2X5 FEMALE				CONN	CONN2X5-FRA-SFH11	SFH11-PBPC-D05-RA-BK	Sullins Connector Solutions
	1	JP1	HEADER 1x3				Header	CONN-1X3	TSW-103-07-T-S	Samtec
	1	L JS1	Jumper Shunt				Shunt	N/A SHUNT	SNT-100-BK-T	Samtec
	3	8 R1 R13 R14	17.4K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-1742F	Venkel
	1	L R10	0	1A			ThickFilm	R0603   R0603L	CR0603-16W-000	Venkel
	1	R12	2.49K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-2491F	Venkel
	2	2 R2 R5	0	1W			ThickFilm	R2512	RMCF2512ZT0R00	Stackpole Electronics Inc
	1	L R3	1.2K	1/16W		±5%	ThickFilm	R0402	CR0402-16W-122J	Venkel
	1	L R4	604	1/16W		±1%	ThickFilm	R0402	CR0402-16W-6040F	Venkel
	1	R6	3.57K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-3571FT	Venkel
	1	L R7	1.15K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-1151FT	Venkel
	1	S1	SW PUSHBUTTON	50mA	12Vdc		Tactile	SW4N10P4.5	2-1437565-8	Tyco Electronics
	4	SF1 SF2 SF3 SF4	BUMPER				FOOT	RUBBER FOOT SMALL	SJ61A6	3M
	1	L U1	72 QFN-SKT, 10x10x0.5mm				QFN	QFN72N10X10P0.5-SKT-WELLS-CTI	790-42072-101G	Sensata
	1	L U2	DS2431					SOJ6N4.45P1.27	DS2431P+	Maxim
	1	L U3	TPS7A88	1A			LDO	QFN20N4X4P0.5E2.7	TPS7A88	TI
	1	L U30	WSU050-3000	15W	5V		MISC		WSU050-3000	TRIAD
	1	L U4	MAX15102EWL+	2A			LDO	WLBGA15N1.6X2.7P0.5	MAX15102EWL+	Maxim
Not Installed Con	ponents									
NI	Quantity	Reference	Value	Rating	Voltage	Tolerance	Туре	PCB Footprint	ManufacturerPN	Manufacturer
NI	1	R11	0	1A			ThickFilm	R0402 R0402L	CR0402-16W-000	Venkel
NI	1	L R8	1K	1/16W		±1%	ThickFilm	R0402 R0402L	CR0402-16W-1001F	Venkel
NI	1	L R9	20K	1/16W		±1%	ThickFilm	R0402	CR0402-16W-2002F	Venkel
NI		TP1 TP3 TP7 TP8 TP9 TP10 TP11	RED				Loop	TESTPOINT	151-207-RC	Kobiconn
NI		TP2 TP5	BLACK				Loop	TESTPOINT	151-203-RC	Kobiconn
NI		TP4 TP6	BLUE				Loop	TESTPOINT	151-205-RC	Kobiconn

Copyright © 2023, Skyworks Solutions, Inc. All Rights Reserved.

Information in this document is provided in connection with Skyworks Solutions, Inc. ("Skyworks") products or services. These materials, including the information contained herein, are provided by Skyworks as a service to its customers and may be used for informational purposes only by the customer. Skyworks assumes no responsibility for errors or omissions in these materials or the information contained herein. Skyworks may change its documentation, products, services, specifications or product descriptions at any time, without notice. Skyworks makes no commitment to update the materials or information and shall have no responsibility whatsoever for conflicts, incompatibilities, or other difficulties arising from any future changes.

No license, whether express, implied, by estoppel or otherwise, is granted to any intellectual property rights by this document. Skyworks assumes no liability for any materials, products or information provided hereunder, including the sale, distribution, reproduction or use of Skyworks products, information or materials, except as may be provided in Skyworks' Terms and Conditions of Sale.

THE MATERIALS, PRODUCTS AND INFORMATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, INCLUDING FITNESS FOR A PARTICULAR PURPOSE OR USE, MERCHANTABILITY, PERFORMANCE, QUALITY OR NON-INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT; ALL SUCH WARRANTIES ARE HEREBY EXPRESSLY DISCLAIMED. SKYWORKS DOES NOT WARRANT THE ACCURACY OR COM-PLETENESS OF THE INFORMATION, TEXT, GRAPHICS OR OTHER ITEMS CONTAINED WITHIN THESE MATERIALS. SKYWORKS SHALL NOT BE LIABLE FOR ANY DAMAGES, INCLUDING BUT NOT LIMITED TO ANY SPECIAL, INDIRECT, INCIDENTAL, STATUTORY, OR CONSEQUENTIAL DAMAGES, INCLUDING WITHOUT LIM-ITATION, LOST REVENUES OR LOST PROFITS THAT MAY RESULT FROM THE USE OF THE MATERIALS OR INFORMATION, WHETHER OR NOT THE RECIPIENT OF MATERIALS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Skyworks products are not intended for use in medical, lifesaving or life-sustaining applications, or other equipment in which the failure of the Skyworks products could lead to personal injury, death, physical or environmental damage. Skyworks customers using or selling Skyworks products for use in such applications do so at their own risk and agree to fully indemnify Skyworks for any damages resulting from such improper use or sale.

Customers are responsible for their products and applications using Skyworks products, which may deviate from published specifications as a result of design defects, errors, or operation of products outside of published parameters or design specifications. Customers should include design and operating safeguards to minimize these and other risks. Skyworks assumes no liability for applications assistance, customer product design, or damage to any equipment resulting from the use of Skyworks products outside of Skyworks' published specifications or parameters.

Skyworks, the Skyworks symbol, Sky5<sup>®</sup>, SkyOne<sup>®</sup>, SkyBlue<sup>™</sup>, Skyworks Green<sup>™</sup>, ClockBuilder<sup>®</sup>, DSPLL<sup>®</sup>, ISOmodem<sup>®</sup>, ProSLIC<sup>®</sup>, SiPHY<sup>®</sup>, and RFelC<sup>®</sup> are trademarks or registered trademarks of Skyworks Solutions, Inc. or its subsidiaries in the United States and other countries. Third-party brands and names are for identification purposes only and are the property of their respective owners. Additional information, including relevant terms and conditions, posted at www.skyworksinc.com, are incorporated by reference.