# MAX5868 Evaluation Kit

# Evaluates: MAX5868

## General Description

The MAX5868 evaluation kit (EV kit) contains a single MAX5868 high-performance interpolating and modulating 16-Bit, 4.96Gsps digital-to-analog converter (DAC) which can directly synthesize 500MHz of instantaneous bandwidth from DC to frequencies greater than 2GHz. The device is optimized for cable access and digital video broadcast applications and meets spectral emission requirements for a broad set of radio transmitters and modulators, including EPoC, DVB-T, DVB-T2, DVB-C2, ISDB-T, and DOCSIS 3.0/3.1. The MAX5868 EV kit provides a complete system for evaluating performance of the MAX5868 device, as well as development of a digital video solution.

The MAX5868 employs a source-synchronous 16-bit parallel LVDS data input interface. The input baseband I and Q signals are time-interleaved on a single parallel input port configured for double data rate clocking at up to 1240Mwps (620Mwps I and Q each). The device accepts data in word (16 bit), byte (8 bit), or nibble (4 bit) modes. The input data is aligned to the data clock supplied with the data. An input FIFO decouples the timing of the input interface from the DAC update clock domain. In addition, a parity input and parity flag interrupt output are available to ensure data integrity.

The MAX5868 EV kit also includes an on-board PLL-VCO (MAX2871) that provides the DAC update clock signal, CLKP/CLKN. The MAX5868EvkitSoftwareController provides all necessary controls to configure the MAX2871 for the desired DAC update rate from 100Msps to 5Gsps.

The EV kit includes Windows® 7/8/10-compatible software that provides a simple graphical user interface (GUI) for configuration of all the MAX5868 registers through the SPI interface, control of the Xilinx VC707 FPGA data source board, and temperature monitoring.

## Features

- Evaluates the MAX5868 RF DAC Performance, Capability, and Feature Set
- Single 3.3V Input Voltage Supply
- On-Board Clock Generation Module Employing MAX2871 VCO/PLL
- Direct Interface with Xilinx® VC707 Data Source Board
- Windows 7/8/10-Compatible Software
- Optional On-Board SPI Interface Control for the MAX5868
- On-Board SMBus Interface Control for the MAX6654 Temperature Sensor
- Integrated GUI Controls for VC707 Operation
- Proven 10-Layer PCB Design
- Fully Assembled and Tested

***Ordering Information*** ***appears at end of data sheet.***

*Windows is a registered trademark and registered service mark of Microsoft Corporation.*

*Xilinx is a registered trademark and registered service mark of Xilinx, Inc.*

*319-100090; Rev 0; 10/17*

maxim integrated™

## TABLE OF CONTENTS

General Description . . . 1
Features . . . 1
MAX5868EVKIT Files . . . 6
Initial Setup . . . 6
Required Equipment . . . 6
Required Software and Drivers . . . 6
Install the MAX5868EVKIT Software . . . 7
Setup and Connect the MAX5868EVKIT Hardware . . . 7
Configure the MAX5868EVKIT Graphical User Interface . . . 7
Quick Start Procedure . . . 10
Power-Up the MAX5868EVKIT Hardware . . . 10
Run the MAX5868 EV Kit Software . . . 10
Loading a Pattern Manually . . . 11
Detailed Description . . . 11
Detailed Description of Hardware . . . 11
MAX5868 EV Kit Printed Circuit Board . . . 11
Control Interface . . . 12
Interface Modules . . . 12
Power . . . 13
Temperature Monitoring . . . 15
DAC Reference . . . 15
Data Interface . . . 15
Xilinx VC707 FPGA Evaluation Board . . . 15
Programming the EEPROM . . . 16
Detailed Description of Software . . . 17
Data Interface Tab . . . 17
Results Log . . . 17
fCLK, NCO & DSP Tab . . . 19
Status Tab . . . 20
Temperature . . . 21
Device Status . . . 21
Automation Options . . . 21
Register Access Tab . . . 21
VC707 & Setup Tab . . . 23
VC707 FPGA Programming . . . 24
VC707 Pattern Control . . . 24
VC707 MicroBlaze Interface . . . 24
Test Setups . . . 24

## TABLE OF CONTENTS (CONTINUED)

Component Suppliers . . . 24
MAX5868 EV Kit Bill of Materials . . . 25
Output Module Bill of Materials . . . 28
Clock Module Bill of Materials . . . 28
MAX5868 EV Kit Schematic . . . 29
Appendix I – Software and Driver References . . . 43
Third-Party Software and Driver Installation . . . 43
Xilinx ISE 14.7 LabTools Installation . . . 43
Xilinx Drivers Installation . . . 46
Silicon Labs Driver Installation . . . 46
Removal/Uninstallation of the MAX5868 EV Kit Software . . . 46
Upgrading/Updating the MAX5868 EV Kit Software . . . 46
Appendix II – Interface Commands . . . 47
List of VC707 Control Commands . . . 47
Installing Unsigned Drivers on Windows 10 . . . 47
Description and Syntax of VC707 Control Commands . . . 48
Help . . . 48
USB Operations . . . 48
Register Operations . . . 49
I²C Operations . . . 50
Play Operations . . . 51
Ping Command . . . 51
Baudrate Command . . . 51
setIQmsbfirst . . . 51
setmode . . . 51
selmmcmclk . . . 52
spi Operations . . . 52
m2870_spi Operations . . . 52
GPIO Operations . . . 53
Init Command . . . 53
Quit Command . . . 53
Ordering Information . . . 54
Appendix III – Pattern Files . . . 54
Creating Pattern Files . . . 54
Revision History . . . 55

## LIST OF FIGURES

Figure 1. MAX5868 EV Kit and VC707 System . . . . . 5
Figure 2. Splash Screen. . . . . 7
Figure 3. MAX5868EVKIT GUI – Initial Execution Prior to FPGA Configuration . . . . . 8
Figure 4. FPGA Configuration . . . . . 9
Figure 5. Device Manager Window . . . . . 9
Figure 6. FPGA Connected . . . . . 9
Figure 7. Spectral Output – 2 Tone CW at 800MHz . . . . . 10
Figure 8. MAX5868 Evaluation System Block Diagram . . . . . 11
Figure 9. MAX5868EVKIT Jumpers. 9a – Default FPGA Pass-Through Interface; 9b – On-Board FTDI Interface . 12
Figure 10. PLLCLK Module . . . . . 12
Figure 11. MAX5868 Differential Clock Input. . . . . 13
Figure 12. MAX5868 RFDAC Output – 12a.Transformer Output Module and, 12b. Differential DAC Output. . . . . 13
Figure 13. MAX5868EVKIT LEDs . . . . . 14
Figure 14. VC707 LEDs 14a – Before FPGA is Programmed; 14b – After FPGA is Programmed . . . . . 15
Figure 15. VC707 Jumpers and Switches 15a – J44; 15b – SW11; . . . . . 16
Figure 16. MAX5868 EV Kit GUI – Data Interface After Initialization and Loading Test Setup #2. . . . . 18
Figure 17. MAX5868 EV Kit GUI – fCLK, NCO, and DSP After Initialization and Loading Test Setup #2 . . . . . 19
Figure 18. MAX5868EVKIT GUI Status After Initialization and Loading Test Setup #2 . . . . . 20
Figure 19. MAX5868EVKIT GUI Register Access After Performing a Register Read and Write Operation . . . . . 22
Figure 20. MAX5868EVKIT GUI – VC707 & Setup After Initialization and Loading Test Setup #2 . . . . . 23
Figure A1-1. Various LabTools Installation Screens. . . . . 43
Figure A1-2. LabTools Installation – Uncheck “Acquire or Manage License Keys” . . . . . 44
Figure A1-3. Various LabTools Installation screens . . . . . 45
Figure A1-4. Microsoft Visual C++ Installation and ISE Completion . . . . . 45
Figure A1-6. Driver Installation. . . . . 46

## LIST OF TABLES

Table 1. MAX5868EVKIT Jumper Settings . . . . . 7
Table 2. MAX5868 EV Kit LED Descriptions . . . . . 14
Table 3. VC707 LED Descriptions . . . . . 16

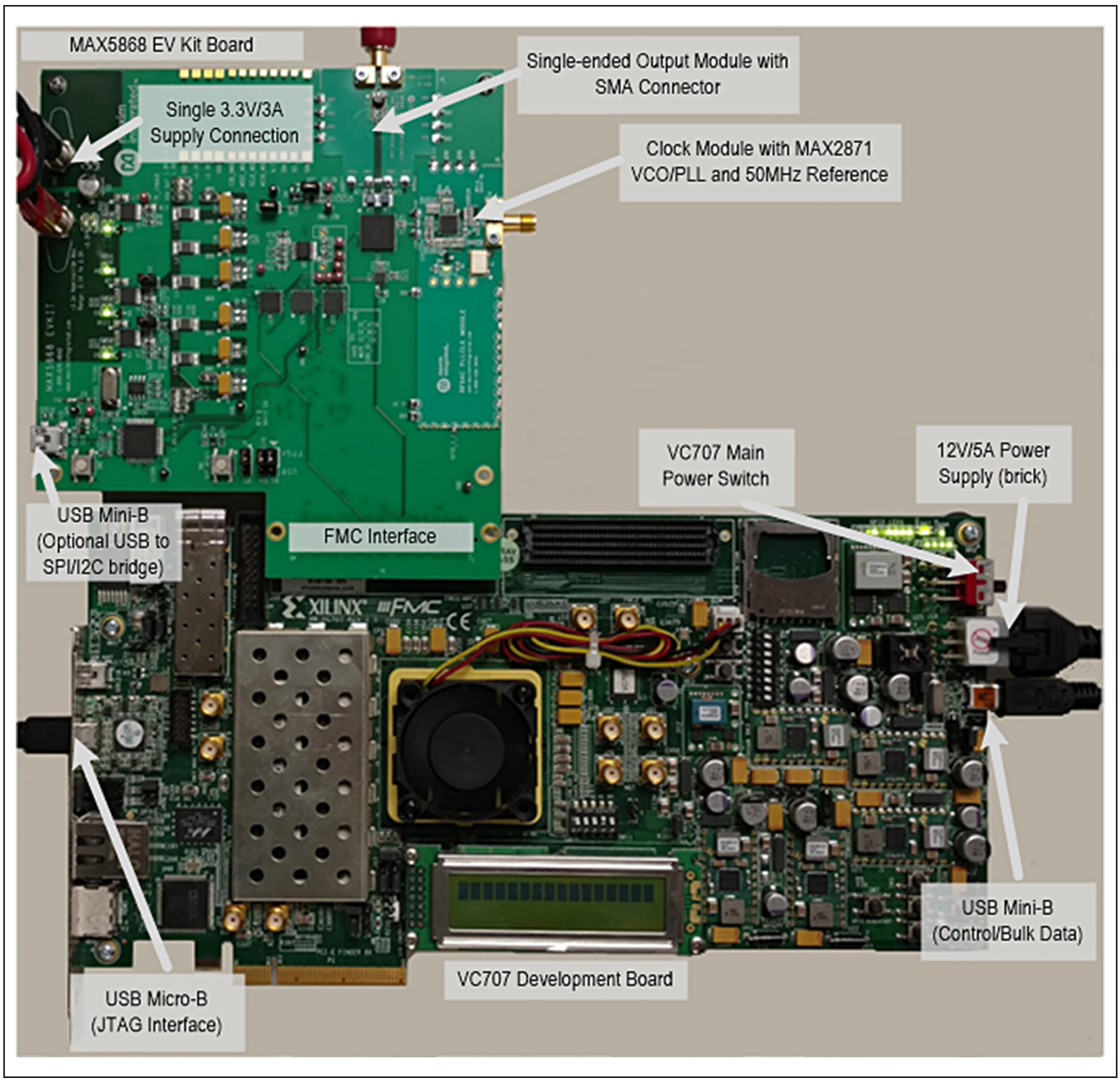



*Figure 1. MAX5868 EV Kit and VC707 System*

## MAX5868EVKIT Files

| FILE | DECRIPTION |
|---|---|
| MAX5868EVKITSoftwareController.exe | Application program |
| AppFiles Directory | Directory application support files including the USB_MS_Bulk_Transfer driver |
| DeviceScripts Directory | Directory with sample MAX5868 configuration scripts and PERL scripts for generating additional scripts |
| PatternFiles Directory | Directory with sample pattern files and Matlab routines for generating additional CW patterns |
| VC707Files Directory | Directory with FPGA programming file and supporting documentation |
| Screenshots Directory | Directory with example Spectrum Analyzer screen captures |
| Miscellaneous DLLs to include FTD2XX_NET.dll, ftd2xx.dll, LibUsbDotNet.dll, libMPSSE.dll, StatusIndicatorTest.dll and MaximStyle.dll | Supporting DLL files for software operation |

It is strongly suggested to use the default installation path (c:\MaximIntegrated\MAX5868EVKIT). If an alternate path is desired, it must NOT contain any spaces or the Xilinx LabTools will not be accessed properly. This step should take **~ 10 minutes**. The EV kit is fully assembled and tested. Follow the steps below to verify board operation:

## Initial Setup

### Required Equipment

- Windows PC (Win-7/10 recommended, Win-8 optional), with one or two available USB 2.0 ports
- Spectrum Analyzer – Agilent PXA or equivalent
- RF Signal Generator – Rohde & Schwarz SMF100A or equivalent (optional)
- 3.3V, 3A power supply for MAX5868 EVKIT
- Xilinx VC707 Evaluation Kit – user supplied
  - VC707 board
  - 12V/5A power cube
  - 1 each USB-A to Mini-B cable for interfacing with the VC707 and MAX5868
  - 1 each USB-A to Micro-B cable for programming the VC707
- Low-Loss SMA/SMA cables as needed for connections to the Spectrum Analyzer and Signal Generator
- **Included in the MAX5868EVKIT**
  - MAX5868 Evaluation Kit board
  - Two 1” stand-offs with screws
  - 2 screws for VC707/MAX5868EVKIT mating

## Required Software and Drivers

The MAX5868EVKIT Software Controller Application requires the following third-party software components and drivers to be installed. *Refer to Appendix I of this document for additional information on this installation process*. It is highly recommended that the target PC be connected to a local area network and have access to the Internet, this allows for automatic download and updates of some drivers. This process may take **30 minutes** or more to complete.

- **Xilinx ISE 14.7 LabTools**
  This is a free tool set used for programming the VC707 evaluation board, no software registration or license is required to use these tools.
- **Xilinx Drivers**
  After the LabTools have been installed on a PC, the VC707 USB interface drivers can be installed.
- **Silicon Labs Driver**
  Manual installation of this driver may not be required if the target PC is connected to the internet. It will be installed after the FPGA on the VC707 is programmed for use of the USB 2.0 bulk transfer port.

### Install the MAX5868EVKIT Software

The MAX5868EVKIT Software Controller application can be obtained from the **www.maximintegrated.com** website. Select the **DESIGN→DESIGN TOOLS** menu option in the upper-left corner of the page. A new page will appear with several additional links. Under the **Applications Software** section, select the **EV Kit Software** option. The next page will contain a listing of all EVKIT software available for download. The page provides a simple filter to ease the users search. Select the **Digital-to-Analog Converters** option if desired, then find and download the MAX5868 RF DAC EV Kit Software.

It is strongly suggested to use the default installation path (**c:\MaximIntegrated\MAX5868EVKIT**). If an alternate path is desired, it must NOT contain any spaces or the Xilinx LabTools will not be accessed properly. This step should take less than **10 minutes**.

### Setup and Connect the MAX5868EVKIT Hardware

1) Install the two 1" stand-offs included with the MAX5868 EV kit. Stand-offs should be installed on the RF DAC output side of the board.
2) Verify all jumpers on the MAX5868 EV kit board are in the default position; refer to Table 1.
3) Connect the MAX5868 EV kit board to the VC707 board using the FMC connector shown in Figure 1.
4) Connect the 3.3V/3A supply to the MAX5868 EV kit and enable the supply's output. Verify that the four green LED board supply indicators are lit.
5) Turn on the VC707 by sliding switch SW12 to the ON position. Verify ALL the LEDs on the VC707 are momentarily lit; the GPIO LEDs should then begin sequencing.
6) Make the USB connections, see Figure 1 for locations
   a. Connect the USB A–micro B cable (JTAG) from Xilinx VC707 Eval board to the PC.
   b. Connect the USB A–mini B cable (Ctrl/Bulk) from Xilinx VC707 Eval board to the PC.

*Please ensure that all the required third-party software and drivers are installed before proceeding to the next section.*

### Configure the MAX5868EVKIT Graphical User Interface

A few items need to be configured during the first execution of the GUI software.

1) Start the MAX5868 EV kit software. Double-click on the desktop icon or the **MAX5868EVKIT softwareController.exe** executable located in the **C:\MaximIntegrated\MAX5868EVKIT** folder. A splash screen will be displayed while the USB connections are established (Figure 2) followed by the main application window.

## Table 1. MAX5868EVKIT Jumper Settings

| JUMPERS | POSITION | EV KIT FUNCTION |
|---|---|---|
| JU1 | Installed* | Normal Operation |
| JU4 | Installed* | Power for U4 – MAX6161 – external reference |
| | Not Installed | MAX6161 NOT powered |
| JU5 | Installed* | MAX5868 external reference connected |
| | Not Installed | MAX5868 using internal reference |
| J4 | 1-2* | SPI control connected through FPGA |
| | 2-3 | SPI control connected to on-board USB |
| J10 | 1-2*, 4-5* | SCL, SDA connected through FPGA |
| | 2-3, 5-6 | SCL, SDA connected to on-board USB |

**Default jumper setting.*

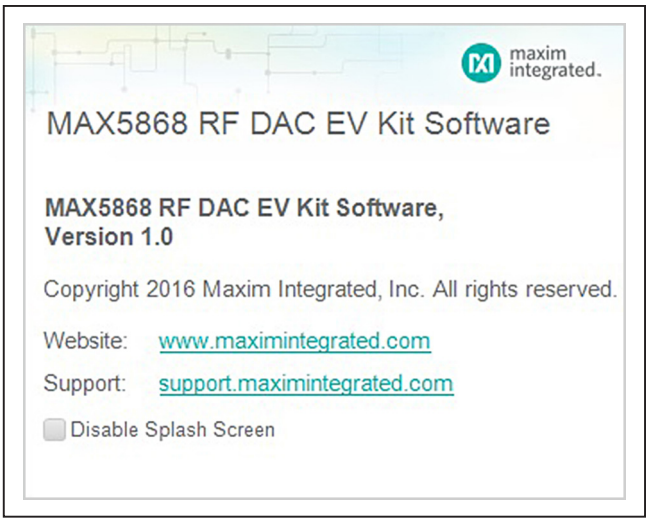



*Figure 2. Splash Screen*

2) After a brief pause during initialization, the primary application window will be loaded

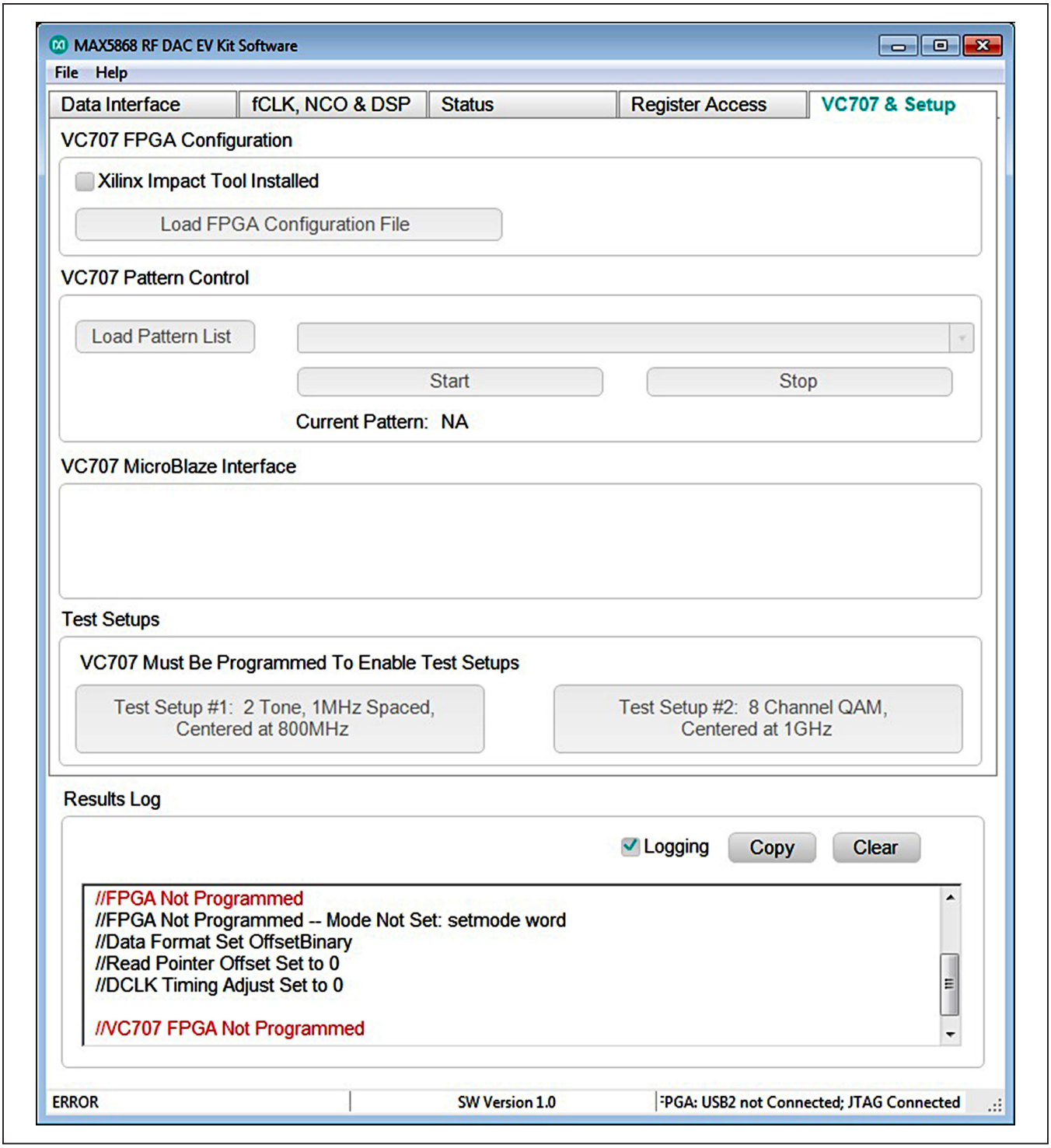



*Figure 3. MAX5868EVKIT GUI – Initial Execution Prior to FPGA Configuration*

3) Load the FPGA configuration
   a. Select the **VC707 & Setup** tab
   b. Click on the **Xilinx Impact Tool Installed** checkbox (Figure 4) and a **File Browser** window will open
      i. Browse to the directory where the impact.exe program is located. If the default installation location was used for the Xilinx Lab Tools, the path will be:
         - For a 32-bit OS – **C:\Xilinx\14.7\Lab-Tools\LabTools\bin\nt**
         - For a 64-bit OS – **C:\Xilinx\14.7\Lab-Tools\LabTools\bin\nt64**
      ii. Select the **impact.exe** file
   c. Click the **Load FPGA Configuration File** button. A file browser will open in the **C:\maximintegrated\MAX5868EVKIT\VC707Files** folder.
   d. Select the **VC707FPGA_6Apr2017.bit** file. A progress bar will display while the FPGA is configured, should take less than 2 minutes.

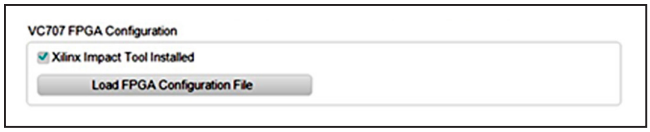



*Figure 4. FPGA Configuration*

After completing the FPGA configuration, the PC will establish a connection to the new USB2.0 port on the FPGA. It should appear as a **USB Mass Storage Device** in **Device Manager** (Figure 5).

**Note:** Before proceeding, ensure any USB flash drives have been ejected from the PC.

4) Update the USB 2.0 port driver
   a. Open the **Windows Device Manager**
      i. With Windows 7, under **Control Panel**, click on **Device Manager**.

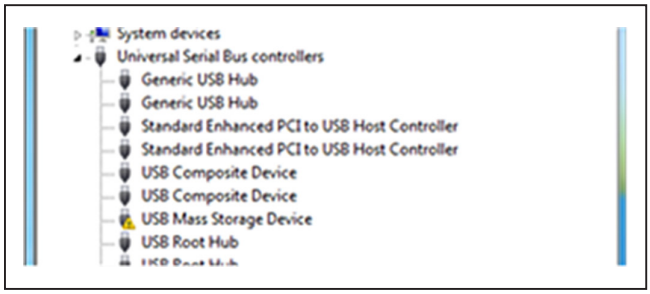



*Figure 5. Device Manager Window*

   b. Select the **USB Mass Storage Device** and right-click to select the **Update Driver Software…** option.
   c. Select **Browse my computer for driver software**.
   d. Select **Let me pick from a list of devices on my computer**.
   e. Select the **USB Mass Storage Device**, then click the **Have Disk** button.
   f. Click the **Browse** button in the **Load from Disk** pop-up window.
   g. Browse to: **C:\MaximIntegrated\MAX5868EVKIT\AppFiles\ThirdParty\USB_MS_Bulk_Transfer** and select the **USB_MS_Bulk_Transfer.inf** file. The window will now indicate "This driver is not digitally signed!"
   h. Click on the **Next** button. A final window will pop-up, stating: "Windows can't verify the publisher of this driver software".
   i. Click on the **Install this driver software anyway** selection.
   j. The MAX5868 software may show a window indicating it has encountered a problem; click on the **Close** button to continue.

5) Reboot the PC and power-cycle the FPGA and EV kit system.
   a. Turn off the VC707 by sliding switch Main Power Switch to the OFF position
   b. Reboot the PC

After the PC has booted, the drivers will be properly configured for use with the MAX5868 software. Once the FPGA is programmed, the GUI Status Bar (bottom of window) should indicate the active connections as shown in Figure 6.

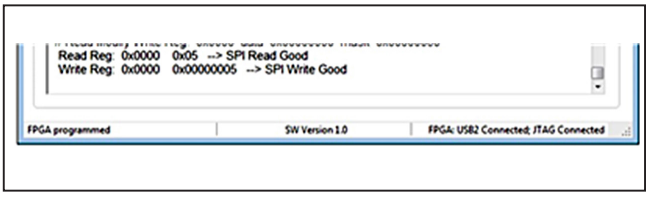



*Figure 6. FPGA Connected*

## Quick Start Procedure

### Power-Up the MAX5868EVKIT Hardware

1) Connect the DAC output to the spectrum analyzer.
2) Enable the 3.3V power supply. Verify that the four, green LED board supply indicators are lit.
3) Turn on the VC707 by sliding the Main Power Supply switch to the ON position (see Figure 1).
4) Verify all LEDs on the VC707 are lit, and the GPIO LEDs are sequencing (for an unprogrammed FPGA).

### Run the MAX5868 EV Kit Software

1) Start the MAX5868 EV kit software.
2) If the VC707 was powered cycled without a programmed EEPROM, reload the FPGA configuration.
   a. Click the **Xilinx Impact Tool Installed** check box.
   b. Click the **Load FPGA Configuration File** button.
      i. A file browser opens in the VC707 folder
      ii. Select the **VC707FPGA_6Apr2017.bit** file and click the **Open** button
3) To quickly load a default pattern…
   a. Click on the **VC707 & Setup** tab of the GUI.
   b. Click on one of the **Test Setup…** buttons.
   c. Example:
      i. Click on the **Test Setup #1: 2 Tone, 1MHz Spaced, Centered at 800MHz** button.
      ii. Set the center frequency of the spectrum a nalyzer to 800MHz.
      iii. Confirm that the spectrum displayed shows two signals of equal amplitude at 799.5 and 800.5MHz, as shown in Figure 7.

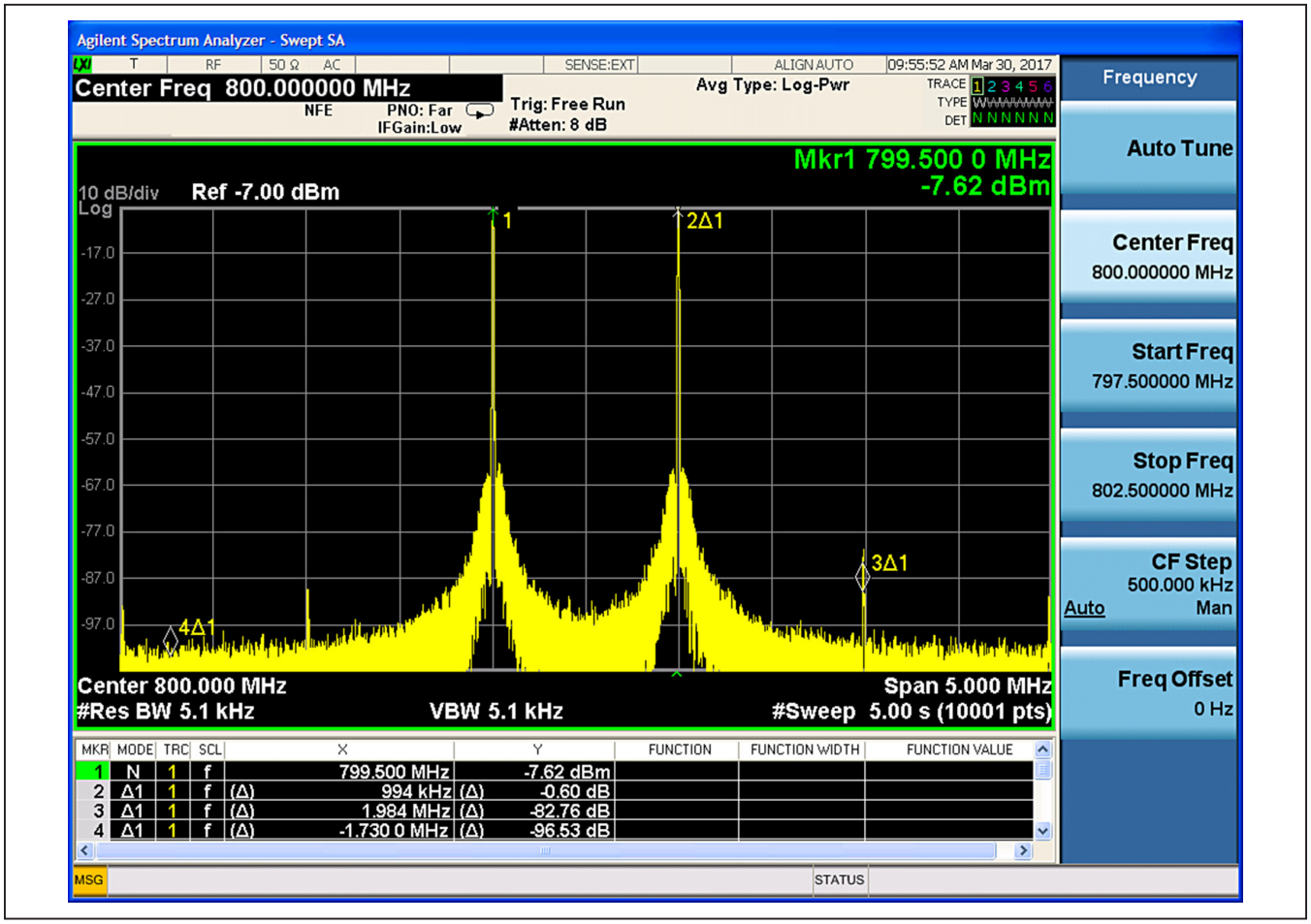



*Figure 7. Spectral Output – 2 Tone CW at 800MHz*

### Loading a Pattern Manually

1) Click on the **VC707 & Setup** tab of the GUI
2) Click on the **Load Pattern List** button
3) Select a pattern file, for example: **TestLoadPatterns.txt**.
4) Click on the pattern drop-down list.
5) Select a pattern, for example: **MAX5868_AnxB_8Ch_fCent_299.002MHz_fs_3.072E+009_-12dB**.
6) Click on the **Start Pattern** button.
7) If the pattern is not visible on the spectrum analyzer, check if the output is muted.
   a. Click on the **Setup** tab.
   b. To unmute the device, click on the **Hardware Mute** switch.

## Detailed Description

### Detailed Description of Hardware

#### MAX5868 EV Kit Printed Circuit Board

The MAX5868 EV kit PCB is manufactured on a 10-layer, 1oz copper, FR4, and Rogers 4350B dielectric stack-up PCB. Layers 2, 4, 6, and 9 are ground planes matched to controlled impedance, 50Ω differential, high-speed traces on the outer layers. All internal power planes (layers 5 and 7) and signal routing planes (layers 3 and 8) have copper ground pours in the unused areas to provide additional decoupling and to ease manufacturability.

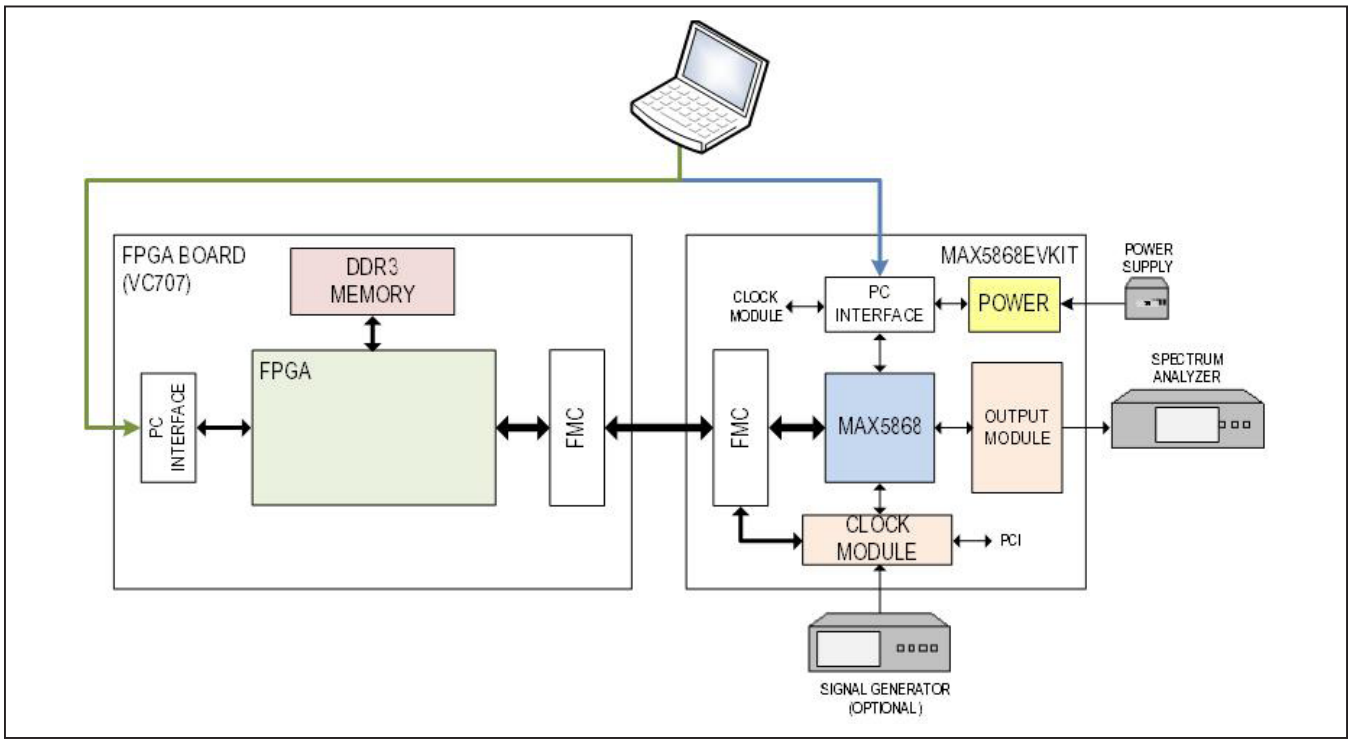



*Figure 8. MAX5868 Evaluation System Block Diagram*

## Control Interface

The MAX5868 EV kit board provides two forms of communication and control interfacing to the RF DAC and the temperature monitor: a pass-through from the FPGA system and an on-board USB Interface. The FPGA pass-through provides a Serial Port Interface (SPI) to control the MAX5868 RF DAC, and a SMBus interface to control the MAX6654 (temperature monitor). The on-board USB interface uses an FTDI4232 device which provides the SPI and I2C bus signals, as well as GPIO controls for the hardwired MUTE, INTB, and RESETB signals on the MAX5868. The FPGA pass-through and the on-board FTDI interface are selected with jumpers J4 and J10 which in-turn use CMOS switches and level translators to route the incoming control signals as needed.

The default settings of J4 and J10 are for using the FPGA pass-through interface. To use the on-board FTDI, switch the J4 and J10 jumpers, as shown in Figure 9b.

## Interface Modules

The MAX5868 EV kit employs two modules to allow for easy interfacing to Signal Generators and Spectrum Analyzers.

The Clock Input Module (MAXPLLCLKMODULE) integrates a MAX2871 23.5MHz to 6000MHz Fractional/Integer-N Synthesizer/VCO that drives the CLKP/CLKN inputs of the MAX5868. The module employs a 50MHz reference crystal for the integrated PLL of the MAX2871. Alternatively, the user can supply an external reference after proper configuration of the module. The modifications required for using an external reference are:

1) Remove R1, 0Ω resistor,
2) Install R2, 50Ω resistor,
3) Install C1, 0.01µF capacitor,
4) Connect external source, between 10MHz and 210MHz at roughly $1V_{P-P}$.



9a. 9b.

*Figure 9. MAX5868EVKIT Jumpers. 9a – Default FPGA Pass-Through Interface; 9b – On-Board FTDI Interface*

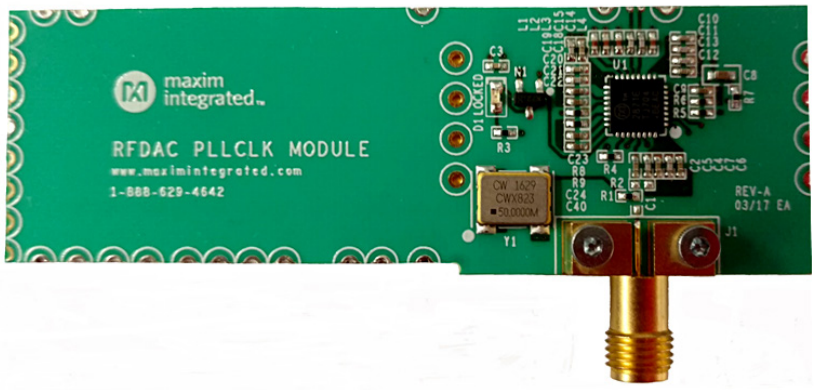



*Figure 10. PLLCLK Module*

A differential source can be connected directly to the MAX5868 by removing the PLLCLK Module and configuring the differential traces on the main EV kit board. The method for reconfiguring the clock input is the following:

1) Remove the mounting screws/backing plate from the SMA connector on the PLLCLK Module.
2) De-solder and remove the Clock Input Module from the MAX5868 EV kit.
3) Populate R61 and R62 with 0Ω resistors.
4) Mechanically connect two Rosenberger, edge-launch SMAs to the MAX5868 EV kit at J7 and J8.
5) Solder the center conductors of J7 and J8 to the differential traces.

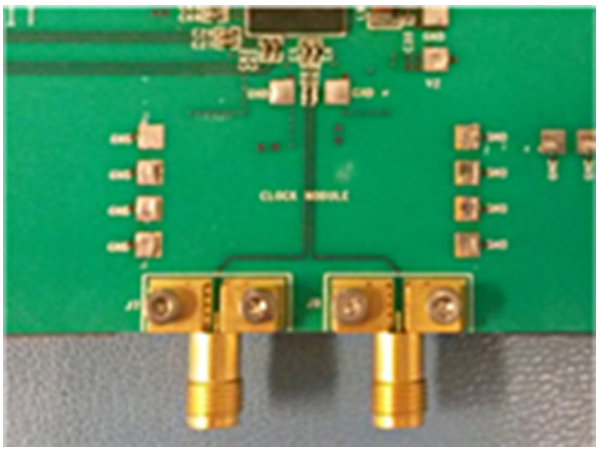


*Figure 11. MAX5868 Differential Clock Input*

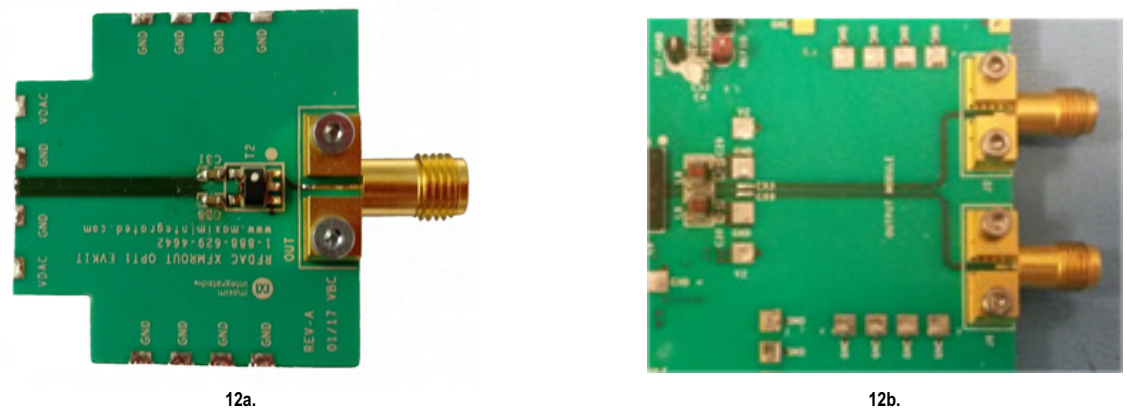
The output module (RFDAC_XFMR_OUT_MODULE) converts the differential output of the MAX5868 RF DAC to a single ended 50Ω output suitable for driving the input of a 50Ω Spectrum Analyzer. The Output Module can be removed from the MAX5868 EV kit board and a differential signal path can be substituted. The method for reconfiguring the DAC output is the following:

1) Remove the mounting screws/backing plate from the SMA connector on the Output Module.
2) De-solder and remove the Output Module from the MAX5868EVKIT.
3) Populate C93 and C99 with 0.01µF capacitors.
4) Mechanically connect two Rosenberger, edge-launch SMAs to the MAX5868 EV kit at J2 and J6.
5) Solder the center conductors of J2 and J6 to the differential traces.

## Power

The MAX5868 EV kit board requires a single +3.3V, 3A power supply connected to the board through two “banana” jacks (marked +3.3V and GND) or a set of wire loops that can be used with EZ-Hooks (also marked +3.3V and GND).

The +3.3V supply is used by the various support circuits including two MAX8527 linear regulators (LDOs) which provide the 1.8V supply rails for the MAX5868 and various support circuitry plus a MAX8556 LDO which supplies the 1.0V level to the MAX5868. One LDO is used for each supply rail, however the LDO outputs are isolated between analog and digital domains by on board filter networks. The PLL supplies for the MAX5868 are isolated from the analog domain through additional filtering.



12a. 12b.

*Figure 12. MAX5868 RFDAC Output – 12a.Transformer Output Module and, 12b. Differential DAC Output*

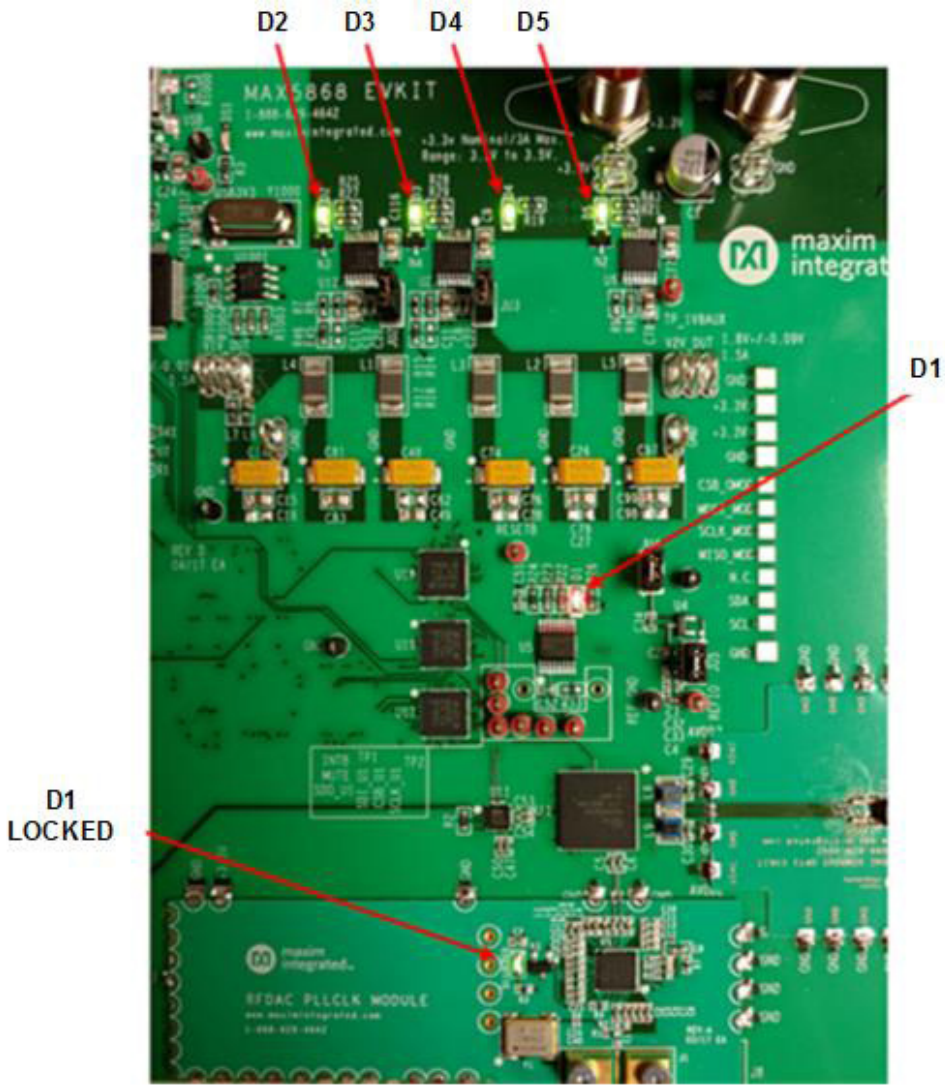



*Figure 13. MAX5868EVKIT LEDs*

## Table 2. MAX5868 EV Kit LED Descriptions

| LED | COLOR | DESCRIPTION |
|---|---|---|
| D1 | Red | Normally Off; Temperature alert based on threshold setting in GUI |
| D5 | Green | Normally On; Auxiliary 1.8V Power Indicator (U9 POK) |
| D2 | Green | Normally On; DUT 1.0V Power Indicator (U12 POK) |
| D3 | Green | Normally On; DUT 1.8V Power Indicator (U2 POK) |
| D4 | Green | Normally On; Main MAX5868EVKIT 3.3V Power Indicator |
| D1 LOCKED | Green | Normally On; PLL Locked, MAX2871 LD Output |

The operational status of each supply can be visually identified by LEDs on the board. When primary power is supplied to the 3.3V VIN on the board, D4 will light green immediately. When the 1.8V and 1.0V rails are within 10% of their nominal output voltages Power-OK lines will light their respective LED indicators.

## Temperature Monitoring

As described in the *Detailed Description of Software–Status Tab* section, an alarm threshold can be set for the MAX5868 device temperature. When this threshold temperature is exceeded, the ALERT output of the MAX6654 Temperature Monitor is asserted (active-low) and D1 is lit as a visual warning. This is a latched output so the alert needs to be cleared manually with the GUI software.

### DAC Reference

The MAX5868 EV kit includes a MAX6120 precision reference for use as an external voltage level for the RF DAC. Power for the MAX6120 is supplied through jumper JU4, while JU5 connects the MAX6120 output to the MAX5868 VREF input.

### Data Interface

The MAX5868 EV kit directly connects to the VC707 FPGA board through the HPC-1 FMC connector, providing a high-quality interconnect which provides the LVDS data interface at DDR rates up to 625MHz. The MAX5868 EV kit incorporates an external frequency divider which divides the REFCLK output to the frequency required by the FPGA.

Schematic and layout files for the MAX5868 EV kit board are included with the software installation and can be found in the **MAX5868\EVKIT** Info folder

## Xilinx VC707 FPGA Evaluation Board

The Xilinx VC707 board acts as the data source for the MAX5868, allowing for user-defined signal generation. Test patterns, generated externally, are stored in the VC707's on-board DDR memory and subsequently transmitted to the MAX5868. A total of 1GB of pattern(s) can be stored, allowing for the use of very long patterns, or multiple patterns consecutively. Multiple patterns allow the user to easily change patterns without repetitive upload commands. The USB2.0 (BULK) interface minimizes the time requirement for uploading the test patterns. Integrated commands allow the VC707 to properly drive all interpolation rates and bus configurations supported by the MAX5868.

The MAX5868 EV kit GUI software also provides a simple interface for controlling the VC707 board. Use the **VC707 & Setup** tab in the GUI to upload the firmware file which configures the on-board Virtex7 FPGA. The firmware design incorporates the MicroBlaze microcontroller function in the FPGA, which is used to manipulate the operation of the FPGA as well as pass-through commands for the MAX5868 EV kit. The supported set of MicroBlaze commands are listed in Appendix II for reference, however all required commands for normal operation are incorporated into specific controls in the GUI software.

When the VC707 board is first powered up, the INIT, DONE, and Supply LEDs will be solidly lit green while the GPIO LEDS will flash ON, cycling from 0 through 7 (see Figure 14a). The GPIO LEDS can be used to identify various states of the FPGA-to-MAX5868 interface. Table 3 describes these states




**14a** **14b**

*Figure 14. VC707 LEDs 14a – Before FPGA is Programmed; 14b – After FPGA is Programmed*

## Table 3. VC707 LED Descriptions

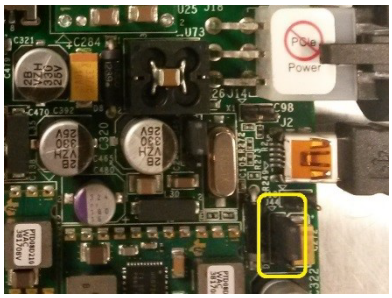
| LED | COLOR | STANDARD OPERATION | DESCRIPTION |
|---|---|---|---|
| 0 | Green | On | REFCLK MMCM locked |
| 1 | Green | On | FPGA system clock MMCM locked |
| 2 | Green | On | FPGA reset, active-low |
| 3 | Green | On | SPISEL - 1 when FPGA controlled |
| 4 | Green | On | DAC MUTE signal |
| 5 | Green | On | DAC RESETB signal, active-low |
| 6 | Green | Off | Serial FPGA clock alive |
| 7 | Green | On | FPGA core clock alive |

All jumpers and switches on VC707 should be used in its default configuration for normal operation of the MAX5868EVKIT software. Occasionally jumpers may have been changed during use with other systems, so it is recommended the user confirm jumper J44 (near the USB 2.0 port) be connected 1-2, as in Figure 15a. Likewise, the user should confirm that Master BPI Programming switch bank, SW11 (near the FPGA and LCD display) be set to 00010 as in Figure 15b.

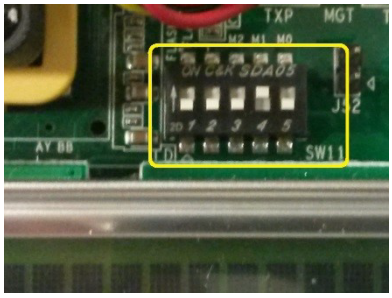
### Programming the EEPROM

Rather than programming the FPGA after each power cycle of the VC707, the on-board flash memory can be used to store the default RTL for the MAX5868 evaluation system. Once the EEPROM has been programmed, the USB cable connecting to the JTAG port (USB micro-B) will no longer be necessary.

For more information on programming the VC707 EEPROM, see the *VC707 FPGA Programming* section in the *Detailed Description of Software*.



15a




15b

*Figure 15. VC707 Jumpers and Switches 15a – J44; 15b – SW11;*

## Detailed Description of Software

The MAX5868 EV kit Controller GUI Software is designed to control the MAX5868 evaluation kit board and the VC707 board as shown in Figure 8. The software includes USB controls that provide SPI and SMBus communication to the MAX5868 and the MAX6654 interfaces. The GUI also controls the VC707 through the USB 2.0 control and bulk transfer port on the VC707 board.

The MAX5868 EV kit Software Controller GUI features four window tabs for configuration and control of the MAX5868 and the VC707. The specific tabs are:

- Data Interface
  - Configure Data Interface and Interpolation Rate
  - Hardware and software MUTE control
  - Various reset functions
- fCLK, NCO, and DSP
  - DAC Clock Configuration (MAX2871)
  - NCO Settings
  - Modulation Settings
- Status
  - Temperature readings and control of the MAX6654 Temperature Sensor IC
  - Status of the MAX5868 EV kit and the Device Under Test
  - Automation support through TCP/IP port
- Register Access
  - User access to read/write MAX5868 configuration and status registers
- VC707 & Setup
  - FPGA Configuration
  - Pattern file loading and control
  - MicroBlaze command line
  - Automated example setups

When the software starts up, the GUI begins on the **VC707 & Setup** tab, as shown in Figure 3.

### Data Interface Tab

The **Data Interface** tab, Figure 16, allows the user to configure the data interface between the FPGA and the MAX5868. Data bus controls are provided for the Width (Word, Byte, Nibble), Interpolation Rate, Data Format (Offset Binary, Two's Complement) and DCLK timing. Additional controls for MUTE and RESET functionality as well as FIFO configuration are also provided.

#### Results Log

Independent of the tab selection, the Results Log block displays active interactions between the GUI software, the MAX5868 EV kit board, the MAX5868 DAC, and the VC707 FPGA system. Logging of most commands can be turned on or off by clicking on the **Logging** check box. The user can manually enter information into the text box and the whole log can be copied to the Windows clipboard or cleared by clicking on the **Copy** or **Clear** buttons, respectively.

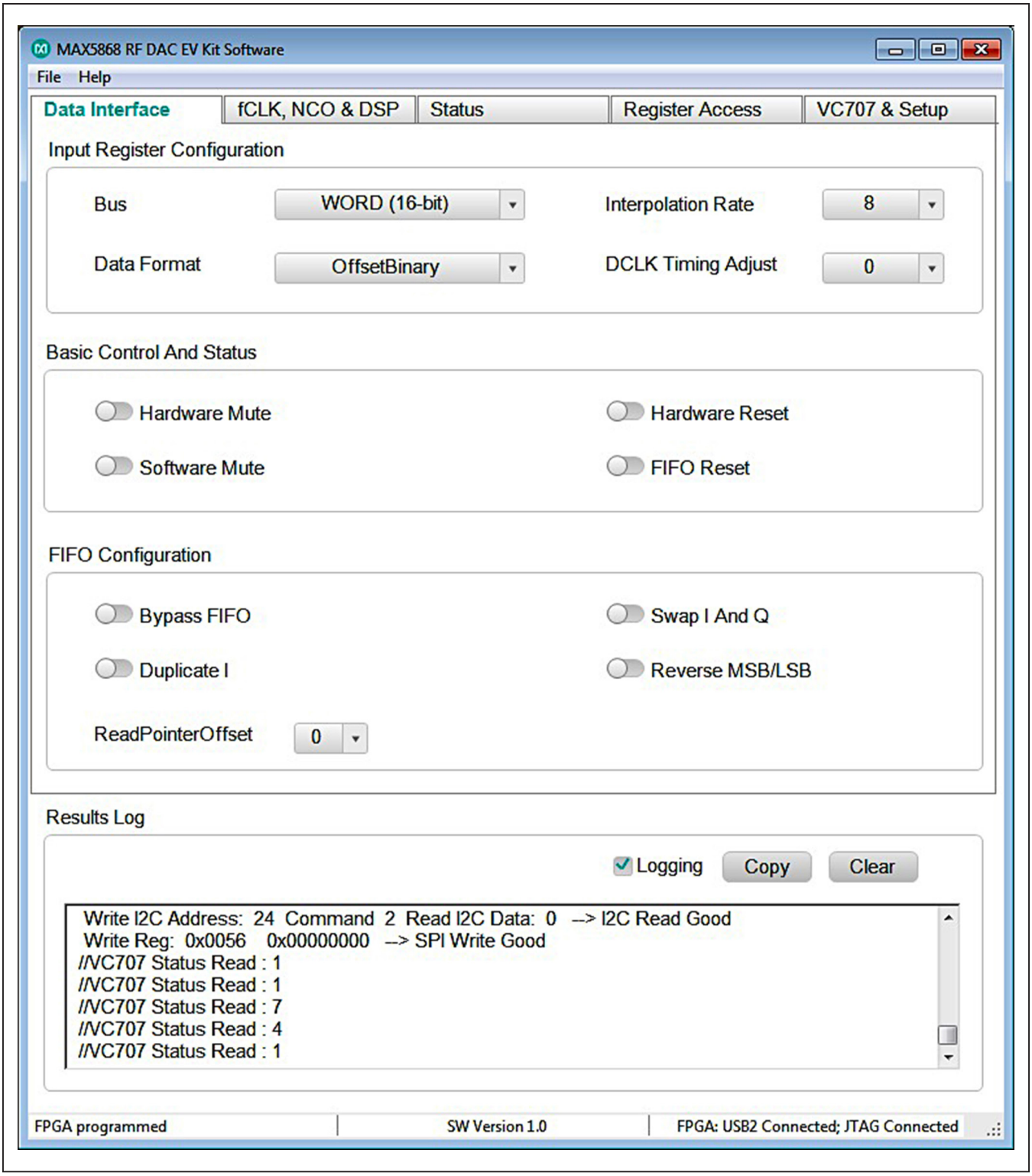



*Figure 16. MAX5868 EV Kit GUI – Data Interface After Initialization and Loading Test Setup #2*

### fCLK, NCO & DSP Tab

The **fCLK, NCO & DSP** tab (Figure 17) provides all the necessary controls to configure the Input Clock Module (MAX2871 configuration) as well as the NCO within the MAX5868.

The **PLL Reference** and **Target DAC Clock Frequency** are initialized to 50MHz and 4915.2MHz, respectively. However, these values can be altered by the user and subsequent executions of the GUI application will start with the last stored settings for these two values.

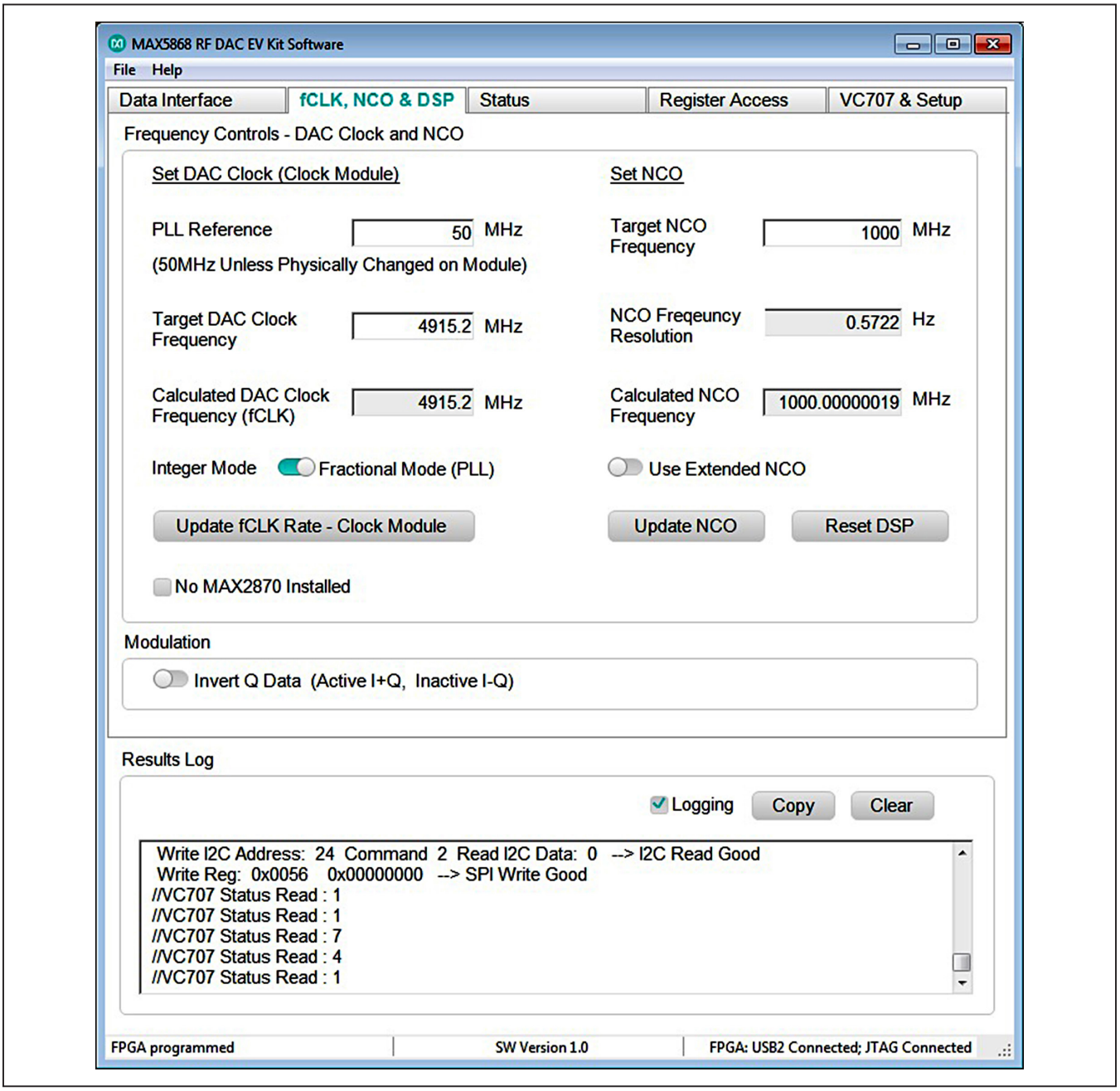



*Figure 17. MAX5868 EV Kit GUI – fCLK, NCO, and DSP After Initialization and Loading Test Setup #2*

### Status Tab

The **Status** tab (Figure 18) is used to monitor the MAX5868 internal status registers as well as certain board functions and the device temperature. It also provides a means for automating measurements using a TCP/IP connection.

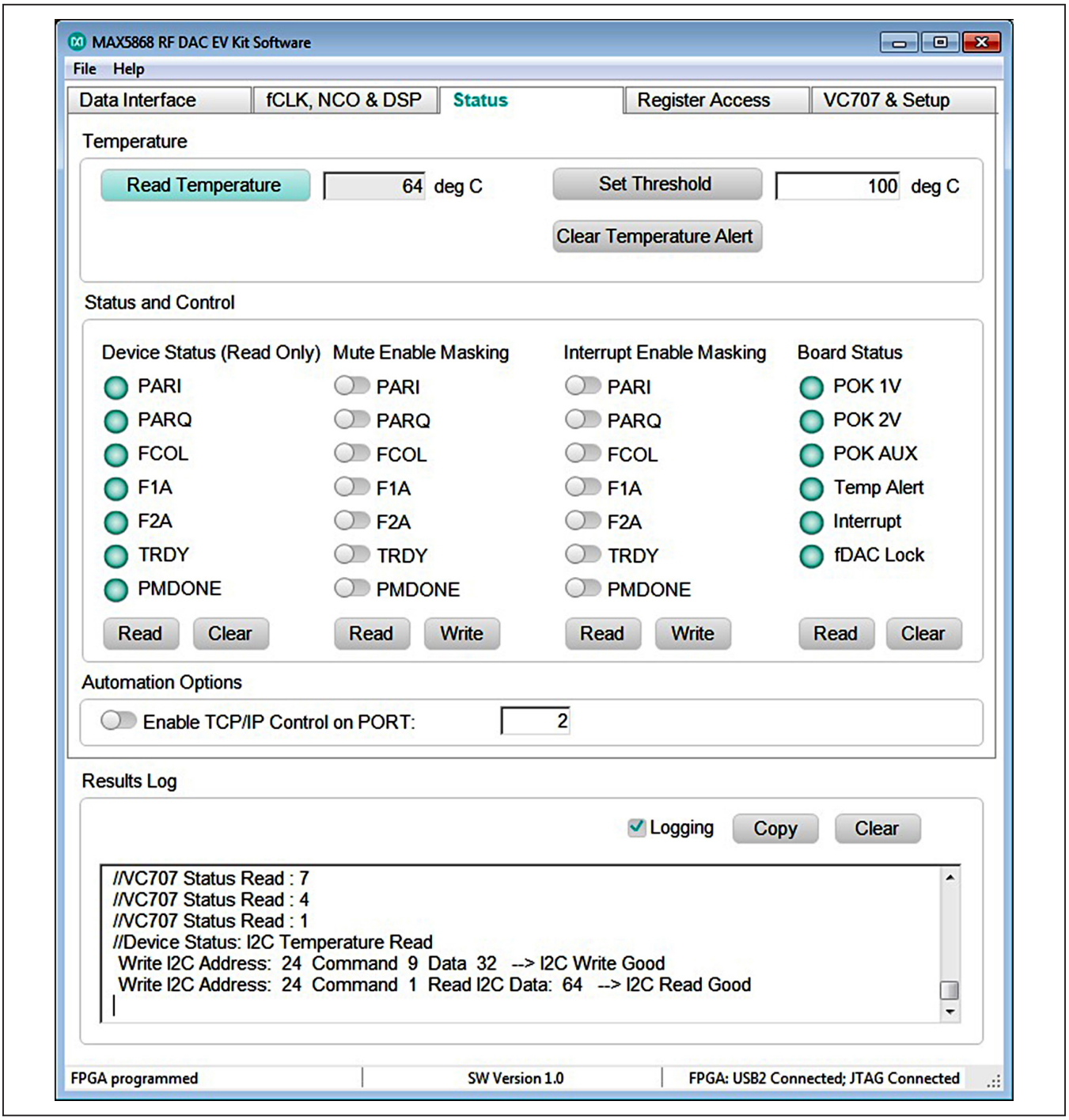



*Figure 18. MAX5868EVKIT GUI Status After Initialization and Loading Test Setup #2*

### Temperature

The **Temperature** block displays the MAX5868 DAC temperature on request. To read the temperature from the MAX6654 through the I2C interface, click on the **Read Temperature** button. If the user would like a visual indicator of an overtemperature fault (ALERT), enter the threshold temperature into the text box and click the **Set Threshold** button. This will cause the MAX6654 to set (active-low) the ALERT output pin when the threshold temperature is exceeded. This will light the red, D1 LED on the MAX5868 EV kit board as a visual warning. This is a latched output so the alert needs to be cleared manually with the GUI software. To clear the temperature fault, click on the **Clear Temperature Alert** button.

### Device Status

The **Device Status** block shows various operating states of the MAX5868. To update the device status, click on the **Get Status** button. The GUI software will read out the flag bits/registers of the MAX5868. These are all displayed in the easy-to-read Device Status panel. If the user wishes to check the individual status bits, the *Register Access Tab* section describes how to interface directly with the MAX5868's internal registers.

### Automation Options

The Automation block contains the **Enable TCP/IP Control** switch and a text box for entering a port number. This utility feature allows the user to run many of the GUI-based operations without the actual GUI. This TCP/IP command option turns the MAX5868 evaluation system into a valuable tool for automated evaluation or characterization of different DAC configurations typically used in automated bench testing.

See *Appendix II* for a list of supported TCP/IP, remote-control commands.

## Register Access Tab

The **Register Access** tab (Figure 19) provides the controls needed to write and/or read individual registers in the MAX5868. Read, Write, or Register Description options are provided for the transaction.

The following process shows the steps required to access the MAX5868 registers.

1) Select the text box next to the **Register 0x** label
   a) Enter the address of the register to access. The register address must be a hexadecimal number ("0x" prefix is not necessary).
2) In the **Select Command** section, choose a radio button to **Read**, **Write** or get **Description** of the previously selected register.
3) In the **Select Data** section, if **Write** was selected then enter the intended value in this text box as a hexadecimal number ("0x" prefix is not necessary)
4) In the **Execute Command** section, after the desired **write** command has been assembled, the user simply clicks on the **Execute** button to write the information to the selected register
5) In the **Results** section, the GUI will provide any feedback received from the MAX5868 through the FPGA interface. If only a **write** operation was executed, a read-back will be reflected in this text box. If a **read** operation was executed then the value is displayed in this text box. Also note that the command that was written and the resultant read-back will be logged in the **Results Log** text in the lower portion of the GUI window.

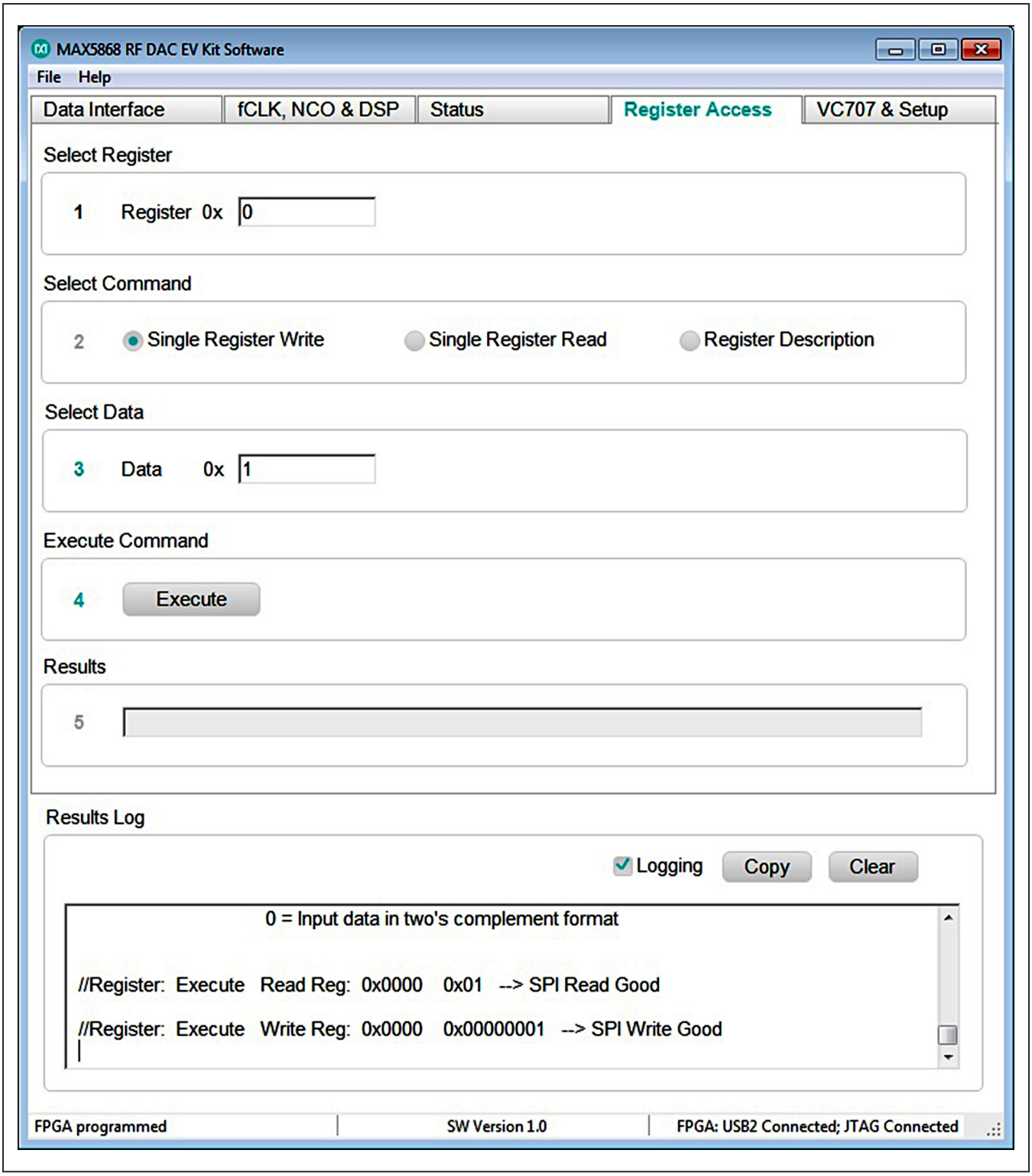



*Figure 19. MAX5868EVKIT GUI Register Access After Performing a Register Read and Write Operation*

### VC707 & Setup Tab

The **VC707 & Setup** tab, Figure 20, allows the user to configure and control the Xilinx VC707 FPGA system.

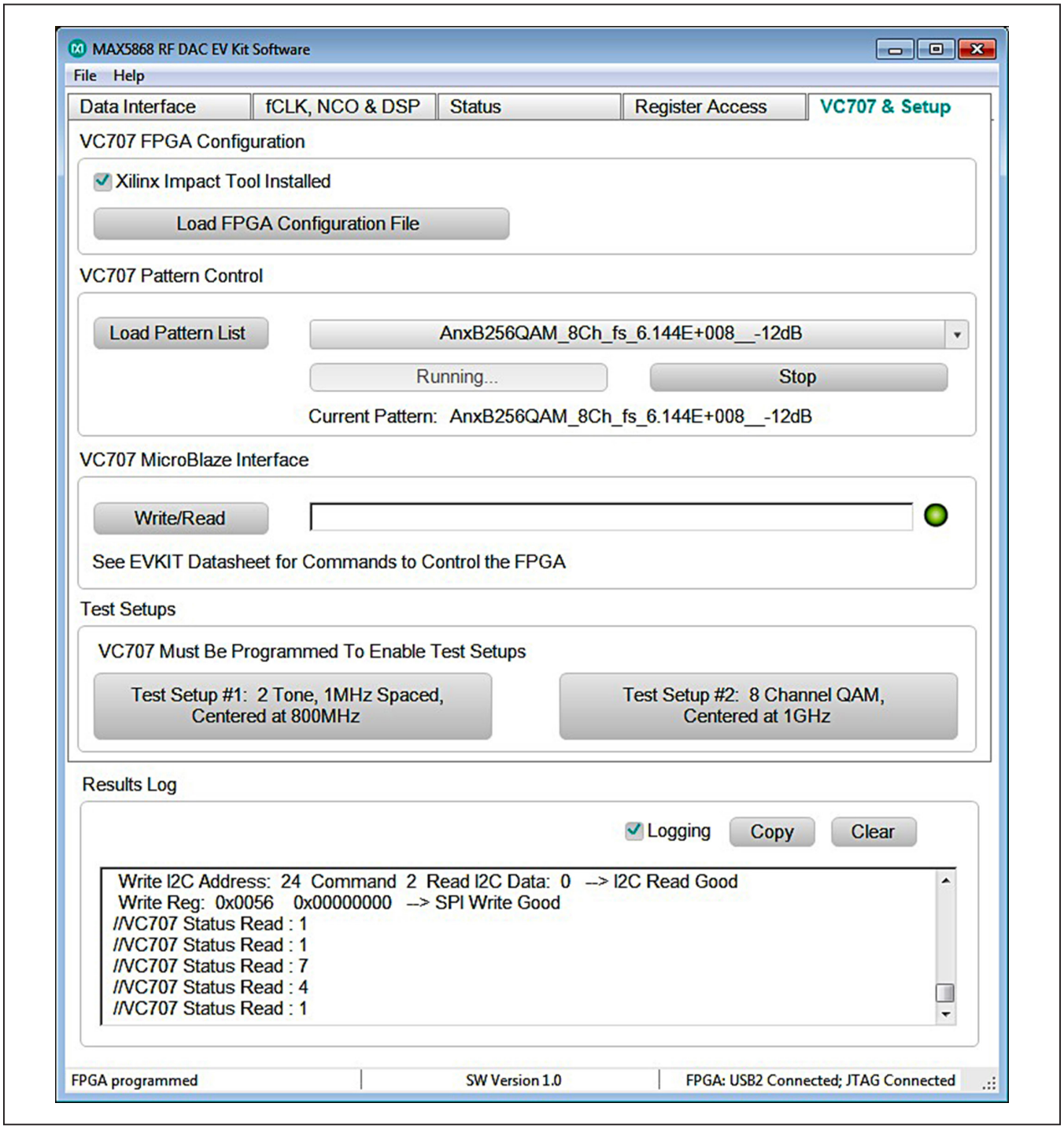



*Figure 20. MAX5868EVKIT GUI – VC707 & Setup After Initialization and Loading Test Setup #2*

### VC707 FPGA Programming

The **FPGA Programming** block allows the user to interact with the Xilinx FPGA at a basic, bit-file level. As already discussed in the *Configure the MAX5868EVKIT Graphical User Interface* section, the VC707 is initially programed through this interface. The FPGA fabric provided with this evaluation system configures the FPGA to control and communicate with the MAX5868. The Virtex-7 FPGA also acts as a data source system, loading pattern data into memory and transferring that data across the high-speed LVDS interface to the DAC.

### VC707 Pattern Control

The **Pattern Control** block allows the user to load different patterns by first opening a Pattern List file. The software utilizes these list files for loading test patterns into the VC707 memory, such as the example **TestLoadPatterns.txt** located in the **MAX5868\ PatternFiles** folder. The list file simply contains an unordered list of names of test pattern files, including extensions. The format is simple ASCII text with one pattern file name on each line. Any line within the list that contains a '#' character will cause it to be skipped when the list is loaded by the GUI. The list can contain multiple patterns with up to 1MB in total pattern length, but only one *pattern list* can be loaded at a time; loading a new list will cause the previously loaded patterns to be overwritten. The MAX5868 EV kit software includes two pattern list files: **TestLoadPatterns.txt** and test **QAM_PatList.txt**.

Pattern files contain the raw waveform data used by the RF DAC to generate an analog RF output. A collection of example patterns used to drive the MAX5868 RF DAC have been provided with the MAX5868 EV kit software. The user may have different types of patterns they wish to test with the evaluation system. To generate other pattern files, it is recommended the user have access to MathWorks MATLAB software.

For additional information on the Pattern File format used with the VC707 FPGA and the MAX5868 software, please refer to *Appendix III*.

### VC707 MicroBlaze Interface

The MicroBlaze Control block provides the user with a means to control and communicate with both the VC707 system and the MAX5868 using the pass-through SPI interface. The user is able to type commands into the text box, and can directly execute those commands by clicking on the <Write/Read> button. Any results that are returned such as values or simply an **ACK#** response will be displayed in the text box. The "LED" indicator to the right of the block will be green when an **ACK#** response is received and will turn red, when a **NAK#** response is received.

### Test Setups

The GUI provides two convenient example setups, with test patterns, to allow the user to easily verify operation of the evaluation system. The two test setups configure the system as follows:

- fDAC: 4915.2MHz
- Interpolation Rate: 8x (614.4MHz Data interface)
- Bus Mode: Word

Each setting has a unique pattern and NCO frequency associated with them, these are also loaded and configured based on the setup selected.

## Component Suppliers

| SUPPLIER | WEBSITE |
|---|---|
| Fairchild Semiconductor | www.fairchildsemi.com |
| Hong Kong X'tals Ltd. | www.hongkongcrystal.com |
| Murata Electronics North America, Inc. | www.murata-northamerica.com |
| Panasonic Corp. | www.panasonic.com |
| Taiyo Yuden | www.t-yuden.com |
| TDK Corp. | www.component.tdk.com |

***Note:*** *Indicate that you are using the MAX5868 when contacting these component suppliers.*

## MAX5868 EV Kit Bill of Materials

| PART | QTY | DESCRIPTION |
|---|---|---|
| GND, +3.3V | 2 | CONNECTOR; MALE; PANELMOUNT; BANANA JACK; STRAIGHT; 1PIN |
| C1 | 1 | CAPACITOR; SMT (CASE_D); ALUMINUM-ELECTROLYTIC; 150µF; 10V; TOL = 20%; MODEL = FK SERIES |
| C3, C4, C23, C29, C30 | 5 | CAPACITOR; SMT (0402); CERAMIC CHIP; 1µF; 6.3V; TOL = 10%; TG = -55°C TO +85°C; TC = X5R; |
| C5, C6, C21, C22 | 4 | CAPACITOR; SMT (0402); CERAMIC CHIP; 100PF; 50V; TOL = 5%; TG = -55°C TO +125°C; TC = C0G |
| C8, C13, C47, C50, C51 | 5 | CAPACITOR; SMT (0402); CERAMIC CHIP; 0.01µF; 10V; TOL = 10%; MODEL = GRM SERIES; TG = -55°C TO +85°C; TC = X5R |
| C9, C11, C15, C24, C49, C76-C79, C83, C99, C116, C117 | 13 | CAPACITOR; SMT (0805); CERAMIC CHIP; 10µF; 6.3V; TOL = 20%; TG = -55°C TO +85°C; TC = X5R |
| C10, C25, C31, C63, C84, C100 | 6 | CAPACITOR; SMT (0201); CERAMIC CHIP; 0.22µF; 6.3V; TOL = 20%; MODEL=; TG = -25°C TO +85°C; TC = X5R |
| C12, C17-C20, C35-C39, C42, C45, C46, C67, C69, C72, C73, C95 | 18 | CAPACITOR; SMT (0402); CERAMIC CHIP; 0.1µF; 10V; TOL = 10%; TG = -55°C TO +85°C; TC = X5R |
| C14, C26, C48, C74, C81, C97 | 6 | CAPACITOR; SMT (6032); TANTALUM CHIP; 47µF; 16V; TOL = 20%; MODEL = TPS SERIES |
| C34, C40, C41, C43, C44, C85-C88, C90, C92, C1002-C1010 | 20 | CAPACITOR; SMT (0402); CERAMIC CHIP; 0.1µF; 10V; TOL = 10%; MODEL = GRM SERIES; TG = -55°C TO +125°C; TC = X7R; |
| C52 | 1 | CAPACITOR; SMT (0402); CERAMIC CHIP; 2200PF; 50V; TOL=10%; TG = -55°C TO +125°C; TC = X7R |
| C53, C54 | 2 | CAPACITOR; SMT (0402); CERAMIC CHIP; 0.1µF; 25V; TOL = 10%; MODEL = GRM SERIES; TG = -55°C TO +125°C; TC = X7R |
| C1000, C1001 | 2 | CAPACITOR; SMT (3528); TANTALUM CHIP; 4.7µF; 16V; TOL = 20% |
| C1011, C1012 | 2 | CAPACITOR; SMT (0402); CERAMIC CHIP; 8PF; 50V; TOL = ±0.25PF; MODEL = C0G; TG = -55°C TO +125°C; TC |
| C1013 | 1 | CAPACITOR; SMT (0402); CERAMIC CHIP; 3.3µF; 6.3V; TOL = 20%; MODEL = C SERIES; TG = -55°C TO +85°C; TC=X5R |
| INTB, MUTE, REFIO, CSB_U1, RESETB, SDI_U1, SDO_U1, USB3V3, SCLK_U1, TP_1V8AUX | 10 | TEST POINT; PIN DIA = 0.1IN; TOTAL LENGTH = 0.3IN; BOARD HOLE = 0.04IN; RED; PHOSPHOR BRONZE WIRE SILVER PLATE FINISH |
| D1 | 1 | DIODE; LED; STANDARD; RED; SMT (0603); PIV = 2V; IF = 0.02A |
| D2-D5, DS1 | 5 | DIODE; LED; WATER CLEAR GREEN; SMT (0603); VF = 2.1V; IF = 0.03A; -55°C TO +85°C |
| GND1-GND5, GND10, REF-GND | 7 | TEST POINT; PIN DIA = 0.1IN; TOTAL LENGTH = 0.3IN; BOARD HOLE = 0.04IN; BLACK; PHOSPHOR BRONZE WIRE SILVER PLATE FINISH |
| J4 | 1 | CONNECTOR; MALE; THROUGH HOLE; BREAKAWAY; STRAIGHT; 3PINS |

## MAX5868 EV Kit Bill of Materials (continued)

| PART | QTY | DESCRIPTION |
|---|---|---|
| J5 | 1 | CONNECTOR; MALE; SMT; HIGH SPEED/HIGH DENSITY OPEN PIN FIELD TERMINAL ARRAY; STRAIGHT; 400PINS |
| J10 | 1 | CONNECTOR; MALE; THROUGH HOLE; BREAKAWAY; STRAIGHT; 6PINS |
| JU2-JU5 | 4 | CONNECTOR; MALE; THROUGH HOLE; BREAKAWAY; STRAIGHT; 2PINS |
| L1-L5 | 5 | INDUCTOR; SMT (1812); FERRITE-BEAD; 120; TOL = ±25%; 3A |
| L6, L7, L1000, L1001 | 4 | INDUCTOR; SMT (0603); FERRITE-BEAD; 28; TOL = ±25%; 4A |
| L8, L9 | 2 | INDUCTOR; SMT (1008); CERAMIC CHIP; 2.2µH; TOL = ±5%; 0.28A; -40°C TO +125°C |
| MODULE1 | 1 | EV KIT MODULE, MAXPLLCLK_ MODULE PACKOUT PART |
| MODULE2 | 1 | EVKIT PART-MODULE; MAXXFMROUT_OPT1#; RFDAC_XFMROUT_OPT1_EVKIT_A; PACKOUT PART |
| N2-N4 | 3 | TRAN; ; NCH; SOT-23; PD-(0.33W); IC-(0.5A); VCEO-(60V) |
| R1 | 1 | RESISTOR; 0603; 499Ω; 1%; 100PPM; 0.10W; THICK FILM |
| R2, R3, R19, R25, R26, R28, R43 | 7 | RESISTOR; 0603; 100Ω; 1%; 100PPM; 0.10W; THICK FILM |
| R4, R5, R21-R23, R27, R29, R34-R36, R41, R1003-R1005 | 14 | RESISTOR; 0603; 10K OHM; 5%; 200PPM; 0.10W; THICK FILM |
| R6, R7, R9, R15 | 4 | RESISTOR; 0603; 4.02K; 1%; 100PPM; 0.10W; THICK FILM |
| R8, R14 | 2 | RESISTOR; 0603; 10.5KΩ; 1%; 100PPM; 0.063W; THICK FILM |
| R17, R33, R46 | 3 | RESISTOR; 0603; 0Ω; 5%; JUMPER; 0.10W; THICK FILM |
| R20 | 1 | RESISTOR; 0603; 1.27KΩ; 0.1%; 25PPM; 0.10W; THIN FILM |
| R24 | 1 | RESISTOR; 0603; 47Ω; 5%; 200PPM; 0.10W; THICK FILM |
| R1001 | 1 | RESISTOR; 0603; 1KΩ; 5%; 200PPM; 0.10W; THICK FILM |
| R1002 | 1 | RESISTOR, 0603, 12KΩ, 1%, 100PPM, 0.10W, THICK FILM |
| R1006 | 1 | RESISTOR; 0603; 2.2KΩ; 5%; 200PPM; 0.10W; THICK FILM |
| SU2-SU5, SU10, SU12 | 6 | TEST POINT; JUMPER; STR; TOTAL LENGTH = 0.256IN; BLACK; INSULATION = PBT CONTACT = PHOSPHOR BRONZE; COPPER PLATED TIN OVERALL |
| SW1, SW2 | 2 | SWITCH; SPST; SMT; 24V; 0.05A; NORMALLY OPEN-SURFACE MOUNT TACTILE SWITCH; RCOIL = Ω |
| U1 | 1 | IC; RFDAC; 16-BIT; 5GSPS INTERPOLATING AND MODULATING RF DAC; CSBGA144 10X10 |
| U2, U9, U12 | 3 | IC; VREG; 0.2V DROPOUT LDO REGULATOR; TSSOP14-EP |
| U3, U6, U8, U14, U16 | 5 | IC; TXRX; 4-BIT DUAL-SUPPLY BUS TRANSCEIVER WITH CONFIGURABLE VOLTAGE TRANSLATION AND 3-STATE OUTPUT; TSSOP16 |
| U4 | 1 | IC; VREF; LOW-COST; MICROPOWER; PRECISION; 3-TERMINAL; 1.2V VOLTAGE-REFERENCE; SOT23-3 |

## MAX5868 EV Kit Bill of Materials (continued)

| PART | QTY | DESCRIPTION |
|---|---|---|
| U5 | 1 | IC; SNSR; ACCURATE TEMPERATURE SENSOR WITH SMBUS SERIAL INTERFACE; QSOP16 |
| U7 | 1 | IC; VSUP; LOW-POWER TRIPLE-VOLTAGE uP SUPERVISORY CIRCUIT; SC70-5 |
| U10 | 1 | IC; VREG; ULTRA-LOW-NOISE, HIGH PSRR, LOW-DROPOUT, LINEAR REGULATOR; SC70-5 ; -40°C TO +85°C |
| U11 | 1 | IC; DIV; DIFFERENTIAL IN-TO-LVDS PROGRAMMABLE CLOCK DIVIDER AND 1:2 FANOUT BUFFER WUTH INTERNAL TERMINATION; MLF16 |
| U17-U19 | 3 | IC; ASW; HIGH-BANDWIDTH; QUAD DPDT SWITCH; TQFN36-EP |
| U1000 | 1 | IC; USB; QUAD HIGH SPEED USB TO MULTIPURPOSE UART/ MPSSE IC; LQFP64 12X12 |
| U1001 | 1 | IC; EEPROM; 2K; 16-BIT MICROWIRE COMPATIBLE SERIAL EEPROM; NSOIC8 150MIL |
| USB | 1 | CONNECTOR; FEMALE; SMT; USB MINI B-TYPE SMT CONNECTOR WITH DOWEL PINS; RIGHT ANGLE; 9PINS |
| GND6, GND7, VIN1, VIN2, V1V_ DUT, V2V_DUT, X2, X6, X5, X8, X7, X9 | 12 | EVK KIT PARTS; MAXIM PAD; WIRE; NATURAL; SOLID; WEICO WIRE; SOFT DRAWN BUS TYPE-S; 20AWG |
| Y1000 | 1 | CRYSTAL; SMT NO DATA; 18PF; 12MHZ; ±30PPM; ±50PPM |
| **PART** | **QTY** | **DESCRIPTION** |
| PCB | 1 | PCB:MAX5868 |
| X20, X28 | 2 | STANDOFF; FEMALE-THREADED; HEX; 4-40IN; 1IN; NYLON |
| X22, X23 | 2 | MACHINE SCREW; PHILLIPS; PAN; 4-40; 3/4IN; 18-8 STAINLESS STEEL |
| X24, X25 | 2 | MACHINE SCREW; PHILLIPS; PAN; 4-40; 3/8IN; 18-8 STAINLESS STEEL |
| X26, X27 | 2 | NUT; HEX; 4-40; #4; STAINLESS STEEL |
| C2, C7, C32, C33, C82 | 0 | PACKAGE OUTLINE 0402 NON-POLAR CAPACITOR |
| C16, C27, C28, C62, C98 | 0 | PACKAGE OUTLINE 0805 NON-POLAR CAPACITOR |
| C55, C93 | 0 | CAPACITOR; SMT (0402); OPEN; FORMFACTOR |
| J2, J6-J8 | 0 | CONNECTOR; FEMALE; SMT; SMA JACK PCB; RIGHT ANGLE; 2PINS; FOR NELCO 4000-13SI BOARD MATERIAL |
| R16, R45 | 0 | PACKAGE OUTLINE 0603 RESISTOR |
| R61, R62 | 0 | RESISTOR; 0402; OPEN; FORMFACTOR |
| R1000 | 0 | RESISTOR; 0603; 0Ω; 5%; JUMPER; 0.10W; THICK FILM |
| TP1 | 0 | TEST POINT; PIN DIA = 0.1IN; TOTAL LENGTH = 0.3IN; BOARD HOLE = 0.04IN; RED; PHOSPHOR BRONZE WIRE SILVER PLATE FINISH |
| TP2 | 0 | TEST POINT; PIN DIA = 0.1IN; TOTAL LENGTH = 0.3IN; BOARD HOLE = 0.04IN; BLACK; PHOSPHOR BRONZE WIRE SILVER PLATE FINISH |

## Output Module Bill of Materials

| PART | QTY | DESCRIPTION |
|---|---|---|
| C28, C31 | 2 | CAPACITOR; SMT (0402); CERAMIC CHIP; 0.01UF; 16V; TOL = 10%; MODEL = ; TG = -55°C TO +125°C; TC = X7R |

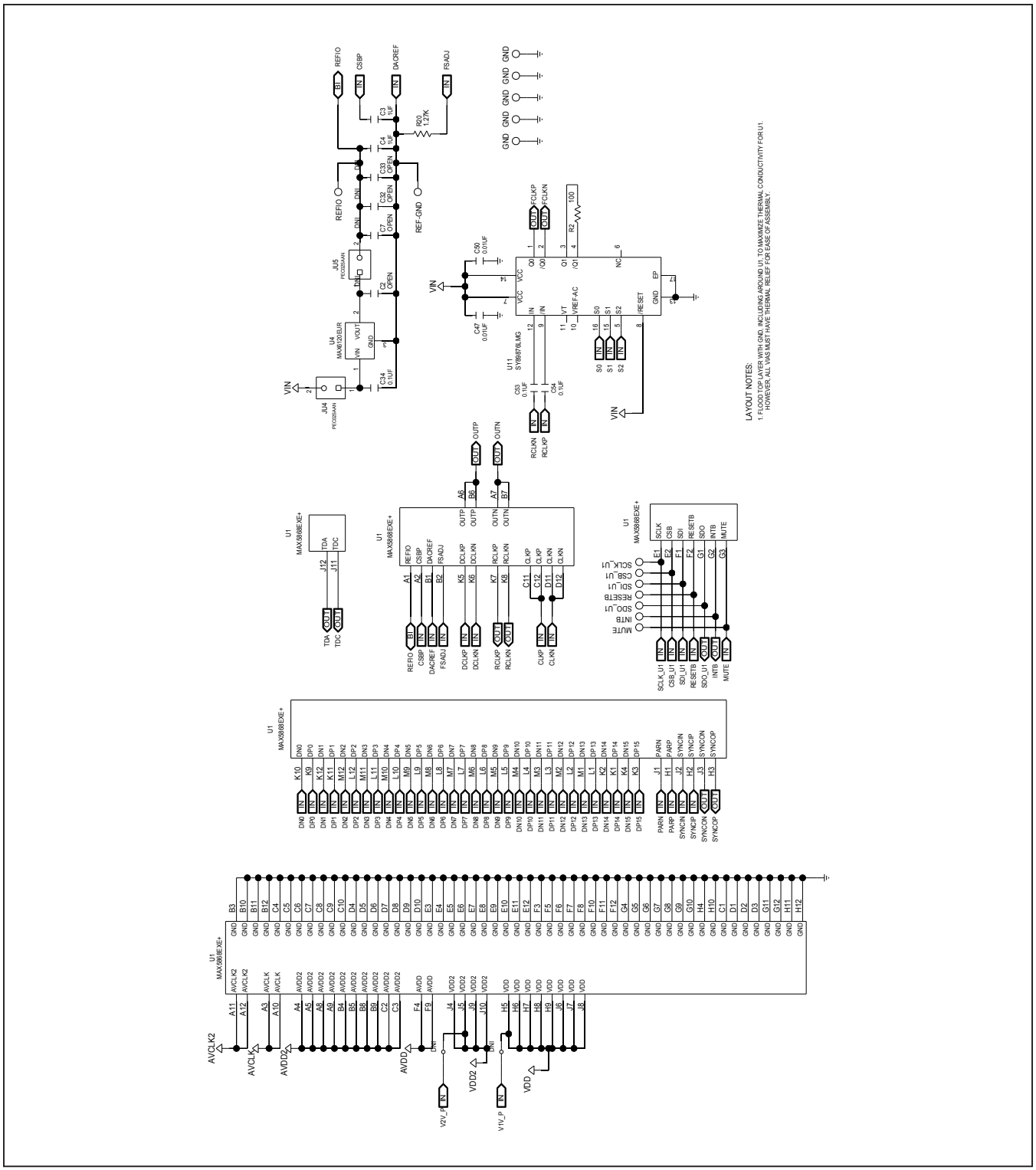
| PART | QTY | DESCRIPTION |
|---|---|---|
| OUT | 1 | CONNECTOR; FEMALE; SMT; SMA JACK PCB; RIGHT ANGLE; 2PINS |
| T2 | 1 | TRANSFORMER; SMT; 4.5-3000 MHZ; RF TRANSFORMER |
| PCB | 1 | PCB: MAXRFDACXFMROUTOPT1 |

## Clock Module Bill of Materials

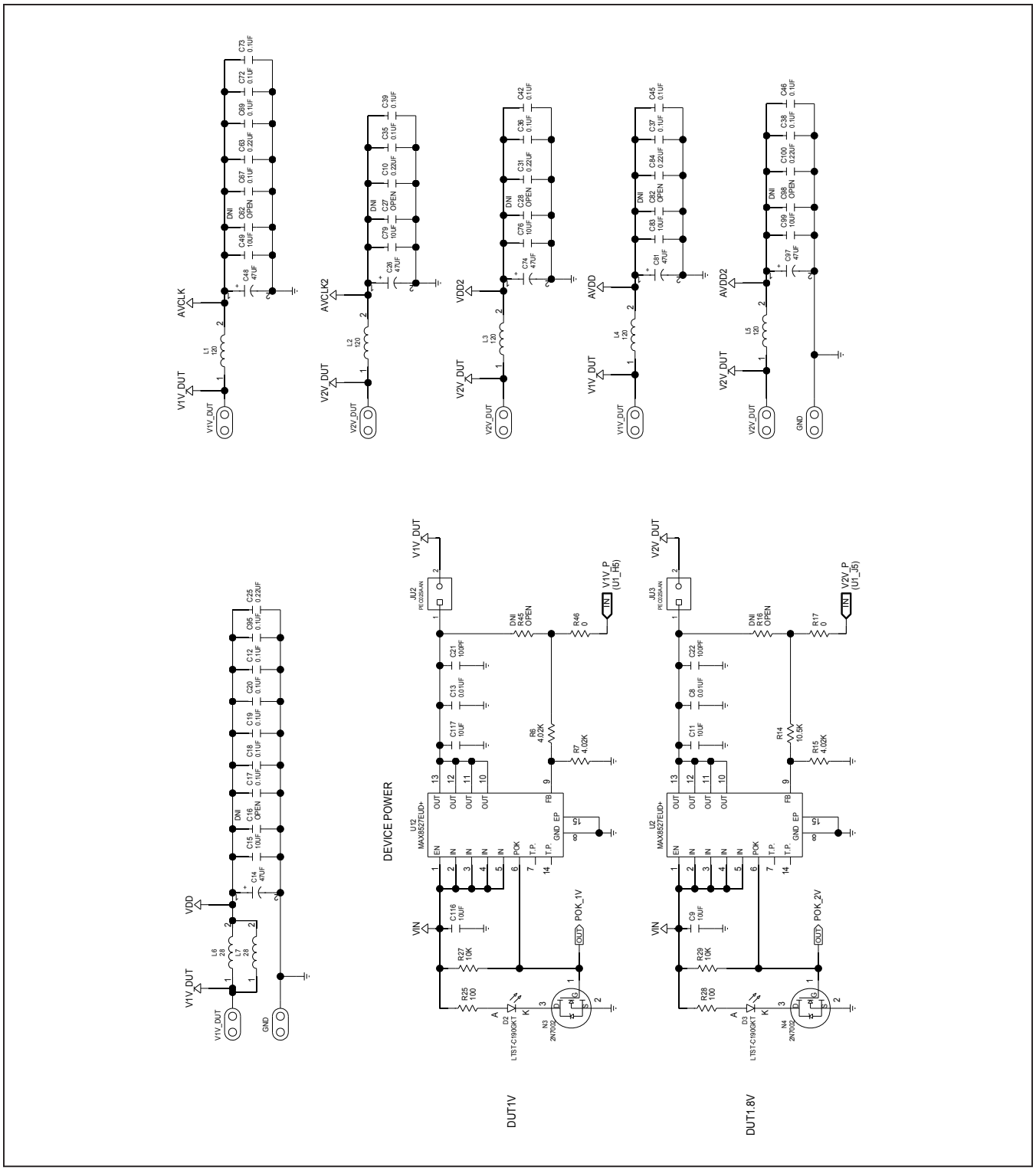
| PART | QTY | DESCRIPTION |
|---|---|---|
| C2-C4, C6, C10, C12, C18, C20 | 8 | CAPACITOR; SMT (0402); CERAMIC CHIP; 0.01µF; 16V; TOL = 10%; MODEL = ULTRA-BROADBAND; TG = -55°C TO +125 °C; TC = X7R |
| C5, C7, C11, C13-C15, C19, C21 | 8 | CAPACITOR; SMT (0402); CERAMIC CHIP; 100PF; 25V; TOL = 2%; MODEL = GRM SERIES; TG=-55°C TO +125°C; TC = C0G |
| C8 | 1 | CAPACITOR; SMT (0603); CERAMIC CHIP; 0.1µF; 16V; TOL = 5%; MODEL = X7R; TG = -55°C TO +85°C; TC=±; |
| C9 | 1 | CAPACITOR; SMT; 0402; CERAMIC; 0.012µF; 16V; 10%; X7R; -55°C to +125°C; 0 ±15% °C |
| C22, C24, C40 | 3 | CAPACITOR; SMT; 0402; CERAMIC;1µF; 10V; 10%; X7R; -55°C to + 125°C |
| C23 | 1 | CAPACITOR; SMT; 0402; CERAMIC; 820pF; 50V; 10%; X7R; -55°C to +125°C; 0 ±15% °C MAX. |
| C1 | 0 | CAPACITOR; SMT (0402); OPEN |
| D1 | 1 | DIODE; LED; WATER CLEAR GREEN; SMT (0603); VF = 2.1V; IF = 0.03A; -55°C TO +85°C |
| J1 | 1 | CONNECTOR; FEMALE; SMT; SMA JACK PCB; RIGHT ANGLE; 2PINS |

| PART | QTY | DESCRIPTION |
|---|---|---|
| L1-L4 | 4 | INDUCTOR; SMT (0402); CERAMIC CHIP; 27NH; TOL = ±5%; 0.4A; -40°C TO +125°C |
| N1 | 1 | TRAN; ; NCH; SOT-23; PD-(0.33W); IC-(0.5A); VCEO-(60V); -55 DEGC TO +150 DEGC |
| R1 | 1 | RESISTOR; 0402; 0Ω; 0%; JUMPER; 0.10W; THICK FILM |
| R3 | 1 | RESISTOR; 0402; 499Ω; 1%; 100PPM; 0.0625W; THICK FILM |
| R4, R5 | 2 | RESISTOR; 0402; 10K; 1%; 100PPM; 0.0625W; THICK FILM |
| R6 | 1 | RESISTOR, 0402, 90.9Ω, 1%, 100PPM, 0.0625W, THICK FILM |
| R7 | 1 | RESISTOR; 0402; 30.1Ω; 1%; 100PPM; 0.0625W; THICK FILM |
| R8 | 1 | RESISTOR; 0402; 243R; 1%; 100PPM; 0.0625W; THICK FILM |
| R9 | 1 | RESISTOR; 0402; 5.11KΩ; 1%; 100PPM; 0.0625W; THICK FILM |
| R2 | 0 | RESISTOR; 0402; OPEN |
| U1 | 1 | IC; SYNTH; MAX2871, 23.5MHZ TO 6000MHZ FRACTIONAL/ INTEGER-N SYNTHESIZER/VCO; TQFN32-EP |
| Y1 | 1 | OSCILLATOR; SMT 5X7; 15PF; 50MHZ; 3.3V; ±50PPM |
| PCB | 1 | PCB: MAXRFDACPLLCLKMODULE |

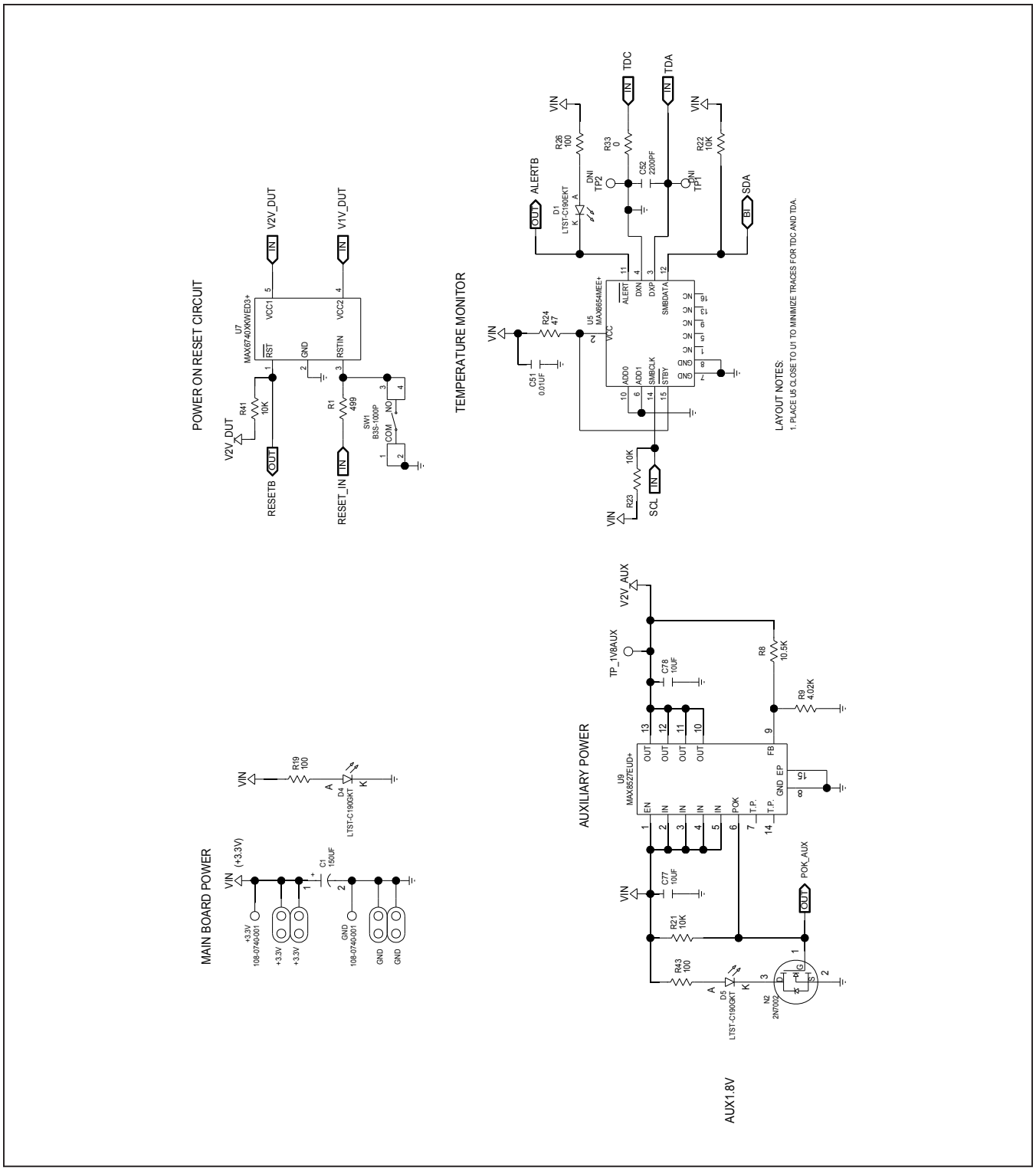
## MAX5868 EV Kit Schematic



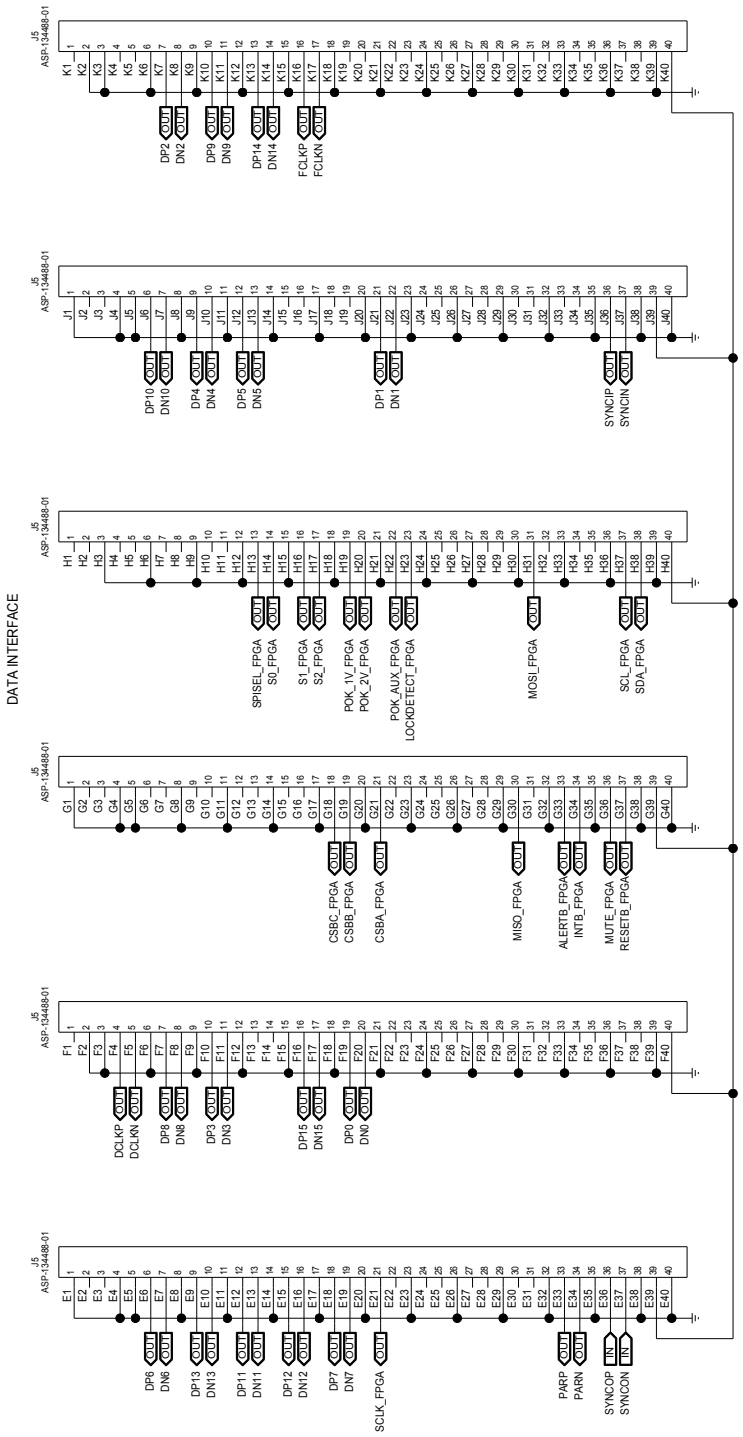
## MAX5868 EV Kit Schematic (continued)



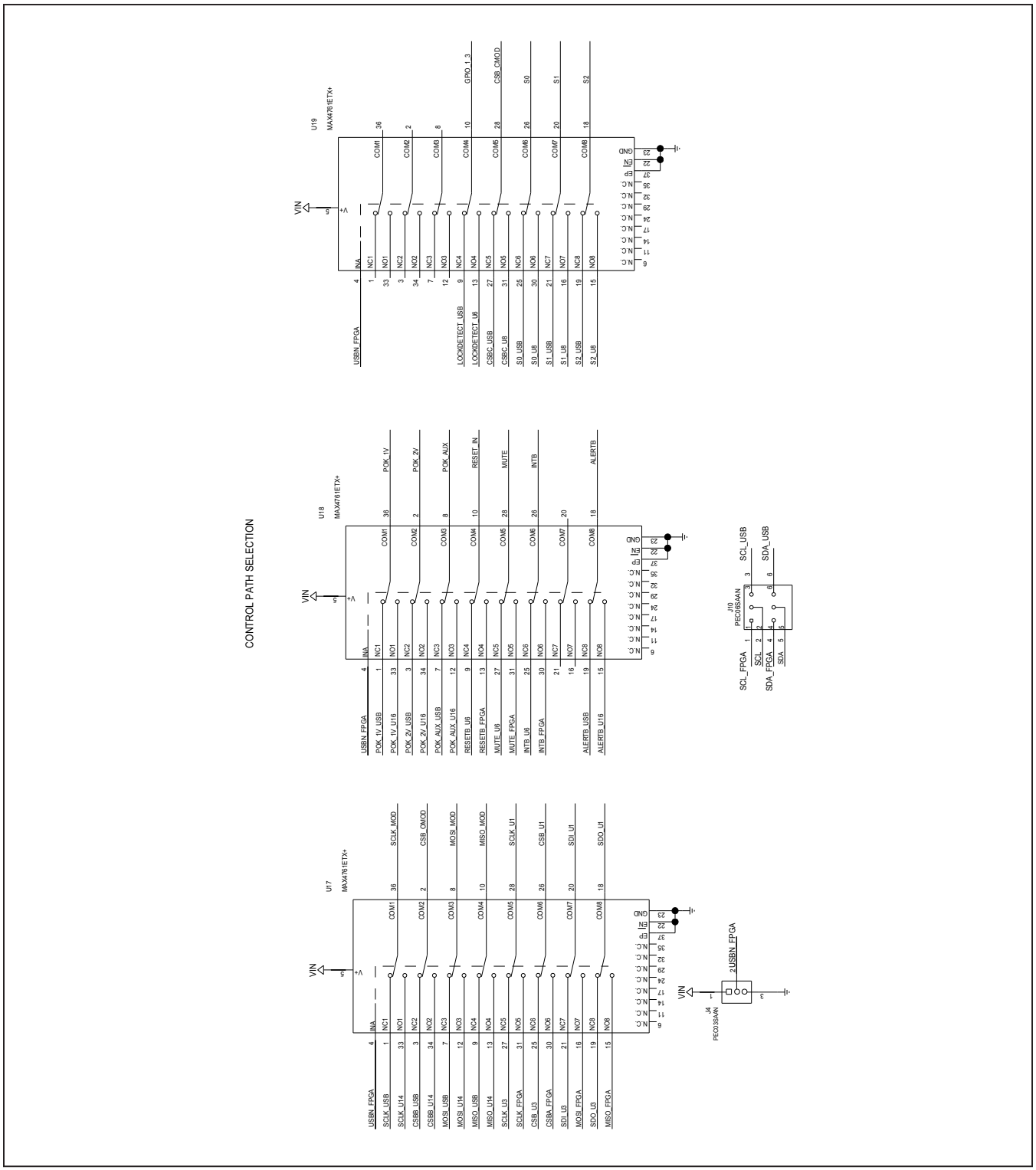
## MAX5868 EV Kit Schematic (continued)



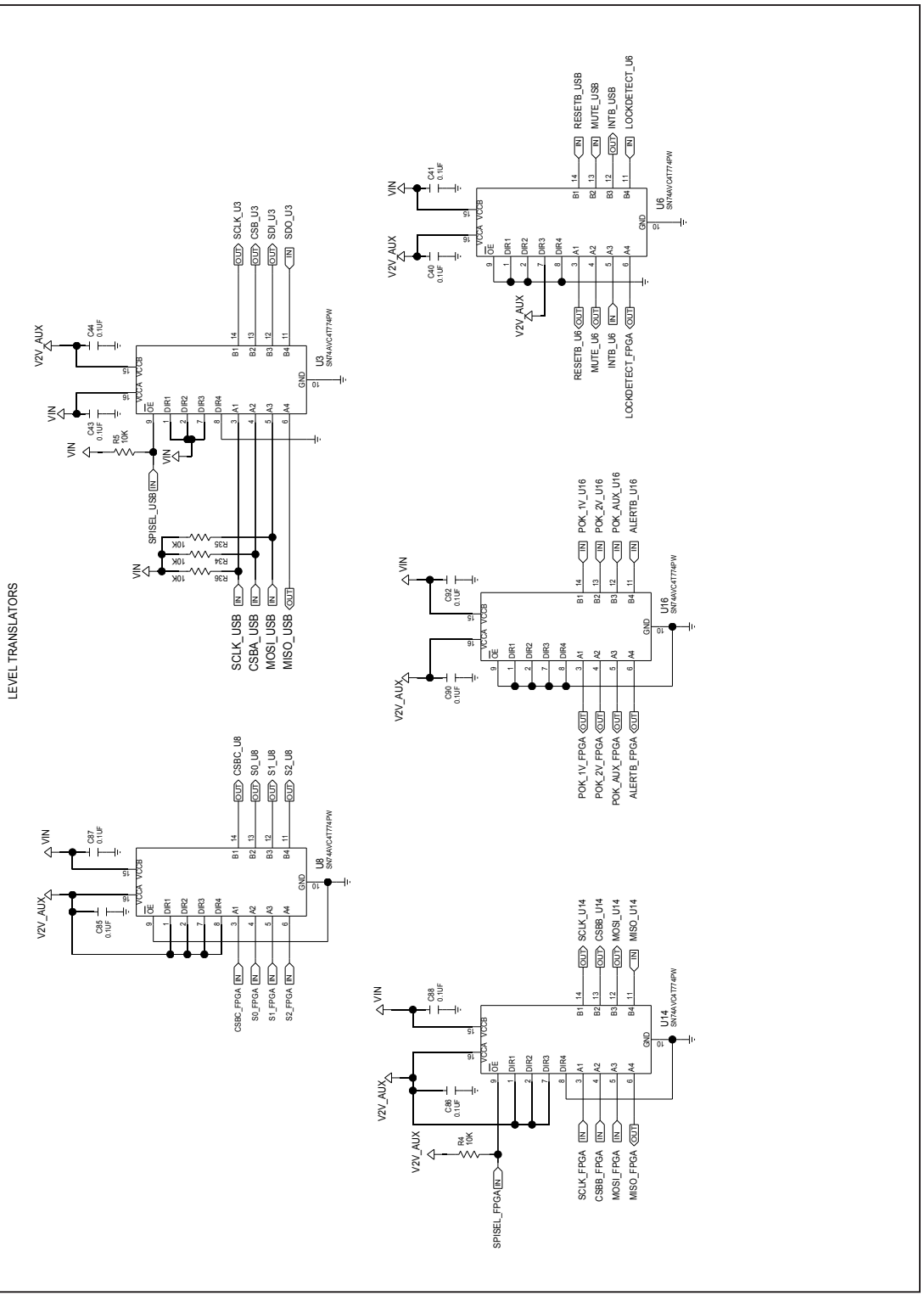
## MAX5868 EV Kit Schematic (continued)



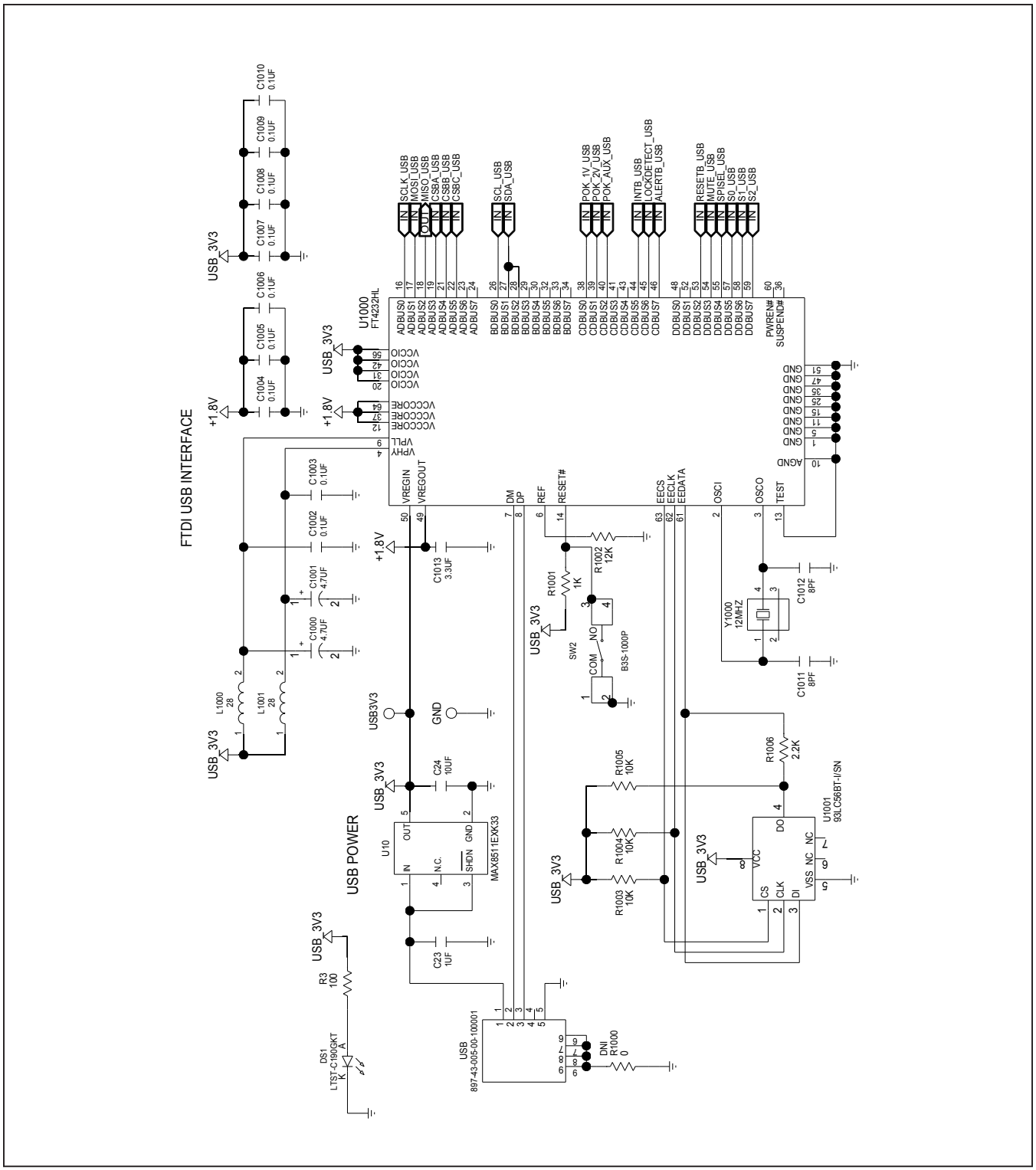
## MAX5868 EV Kit Schematic (continued)



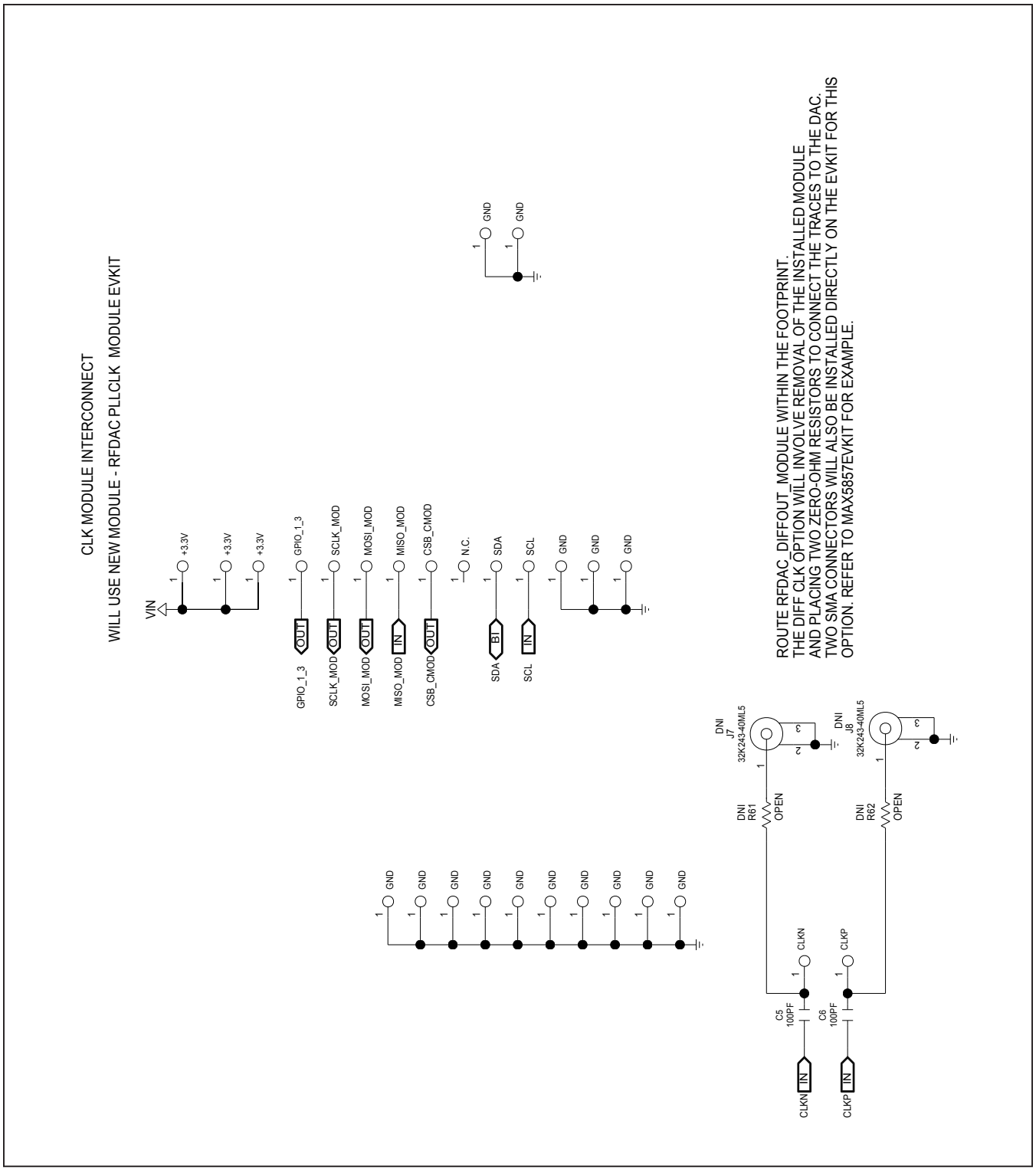
## MAX5868 EV Kit Schematic (continued)



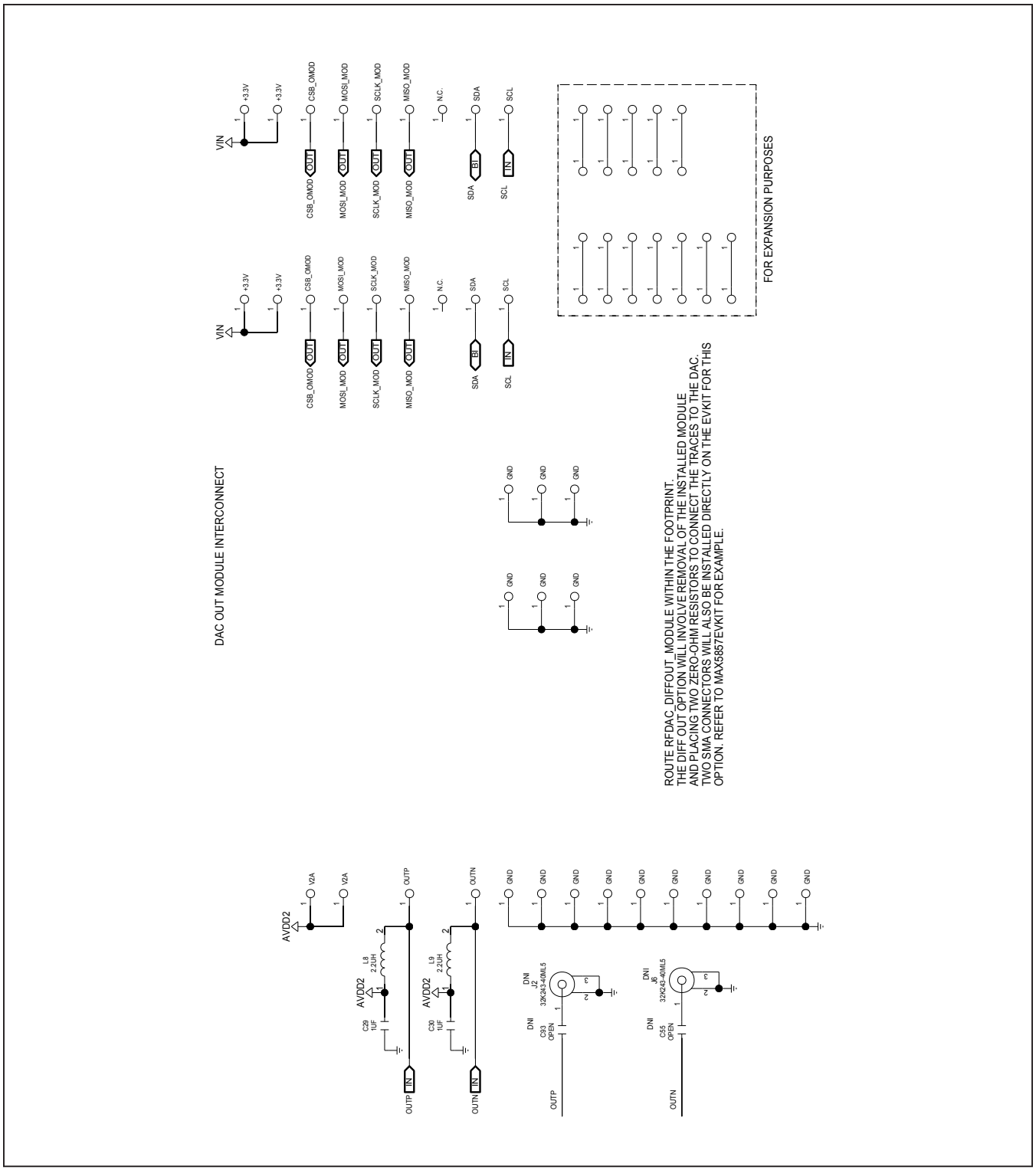
## MAX5868 EV Kit Schematic (continued)



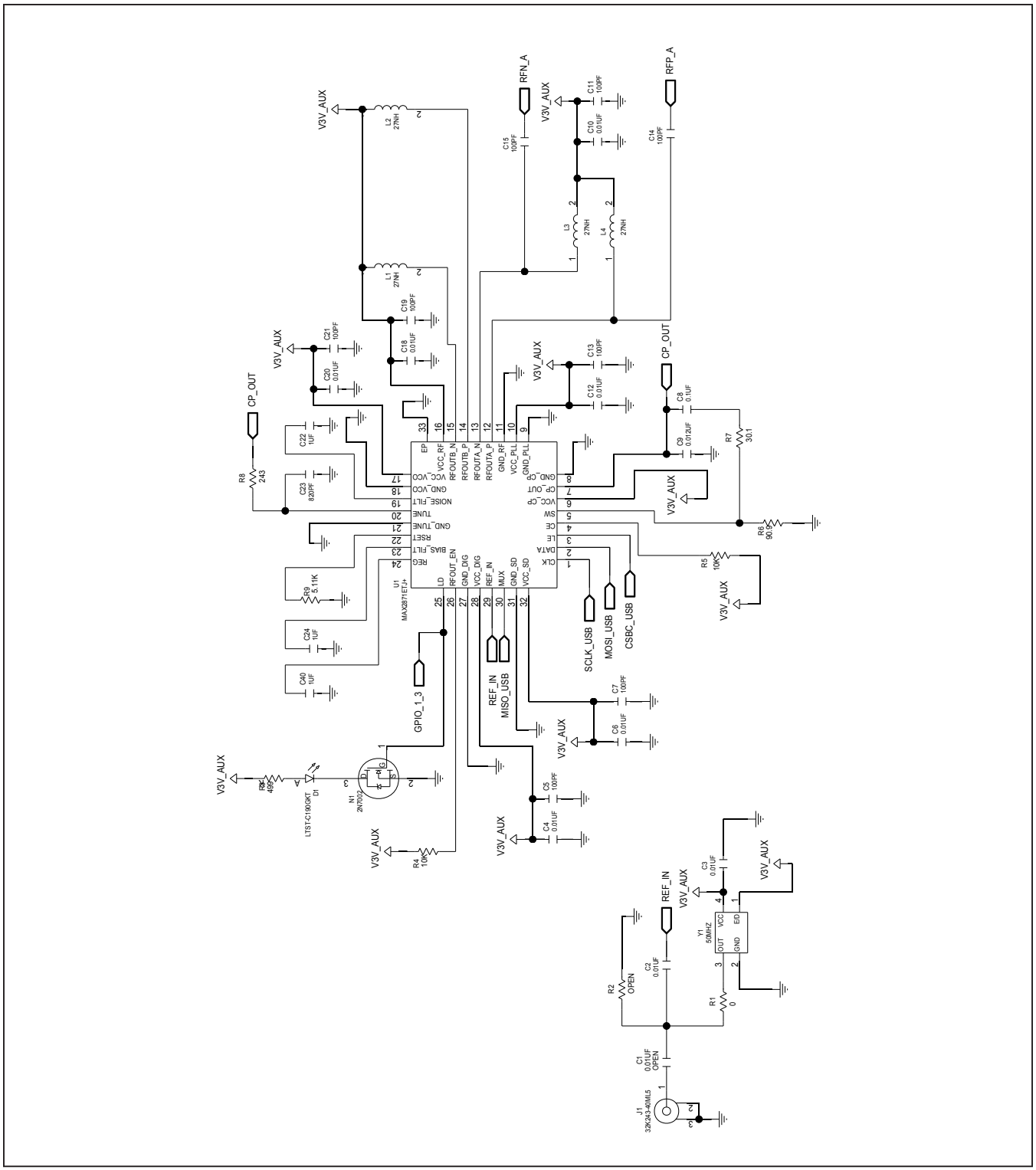
## MAX5868 EV Kit Schematic (continued)



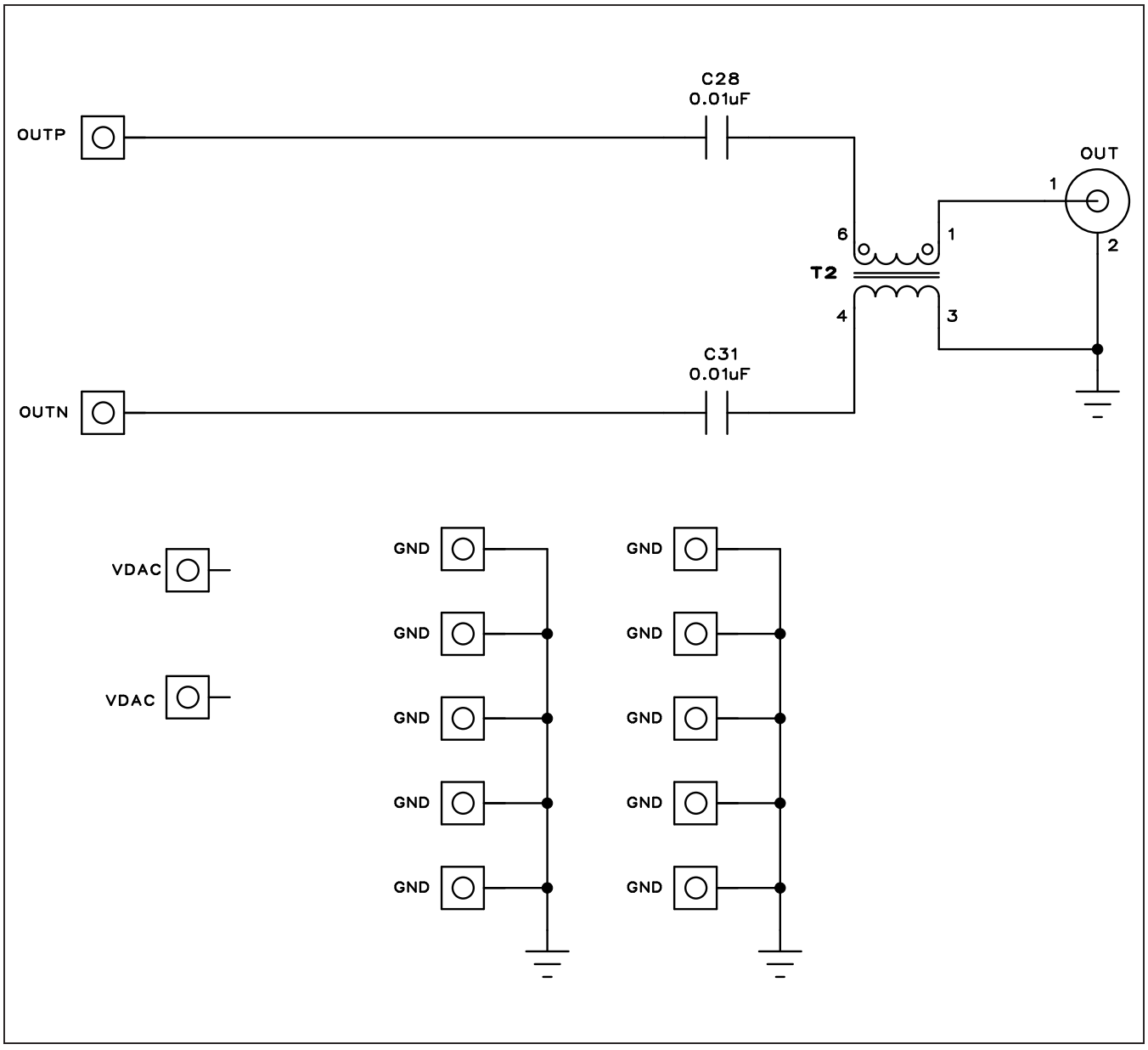
## MAX5868 EV Kit Schematic (continued)



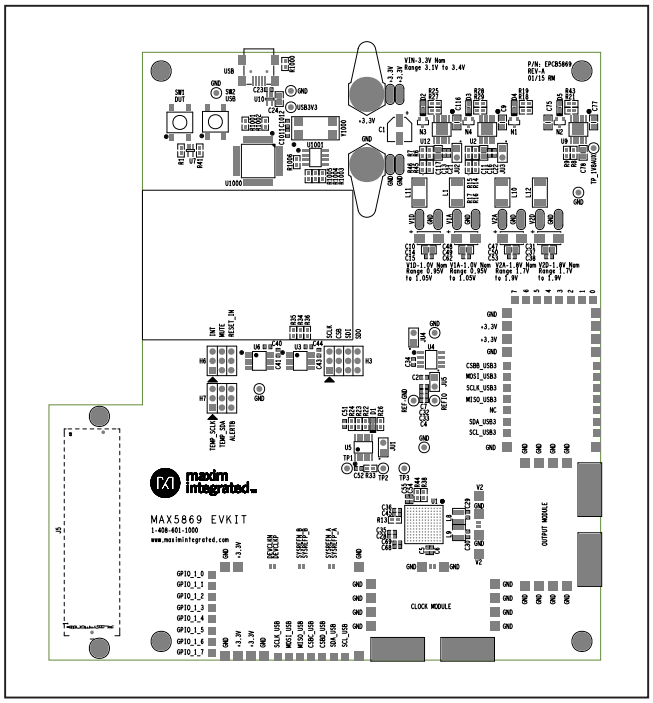
## MAX5868 EV Kit Schematic (continued)



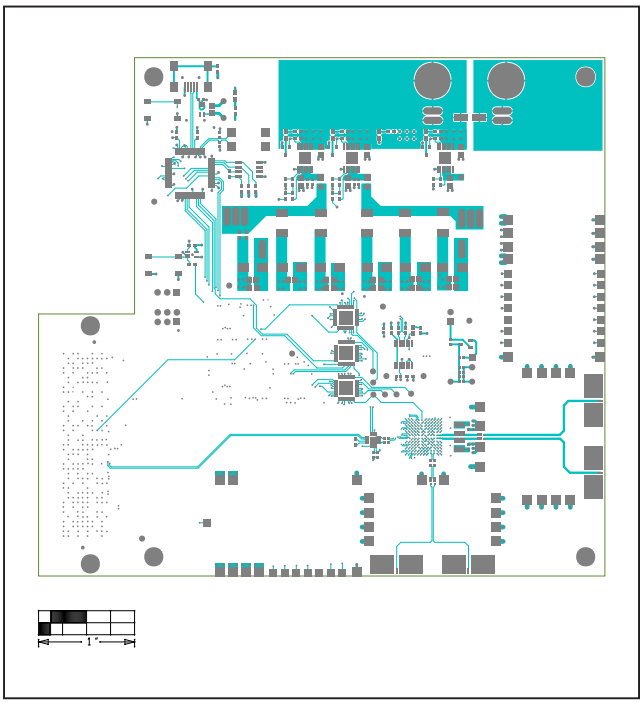
## MAX5868 EV Kit Schematic (continued)



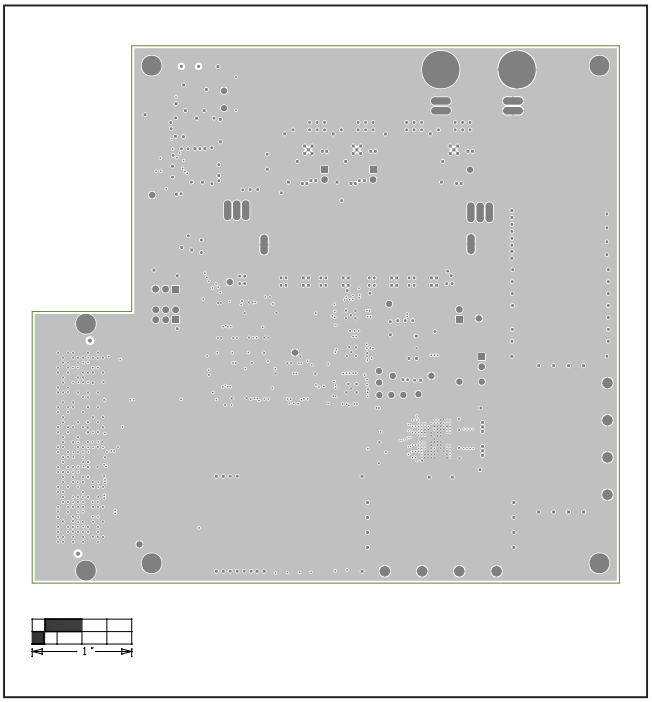
## MAX5868 EV Kit PCB Layout Diagrams



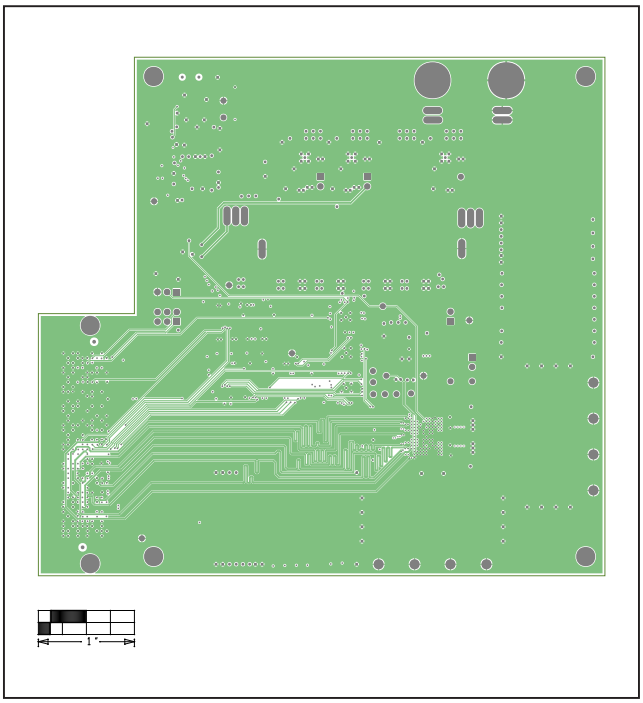

*MAX5868 EV Kit—Top Silkscreen*



*MAX5868 EV Kit—Top*

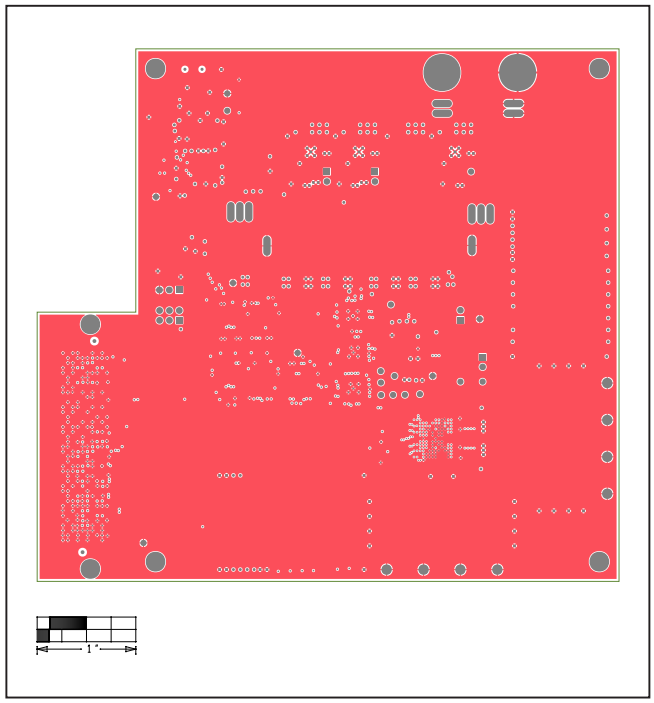


*MAX5868 EV Kit—Level 2 GND*

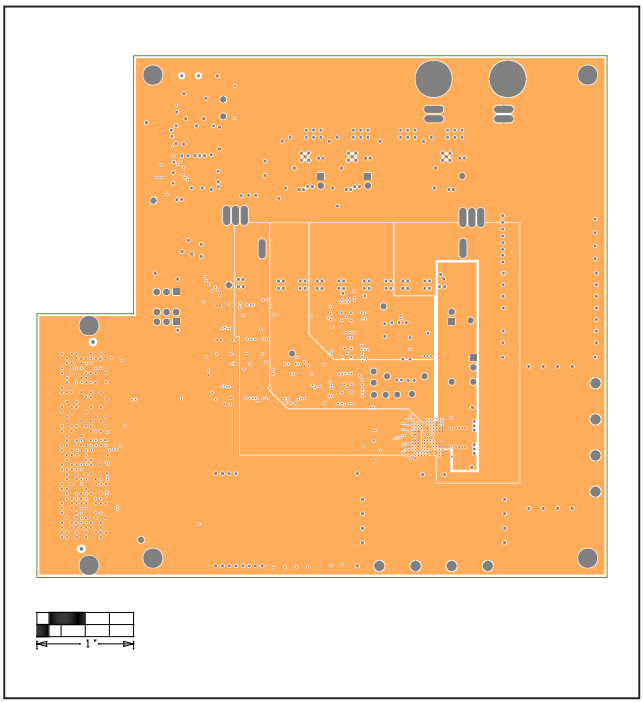


*MAX5868 EV Kit—Level 3 Signal*

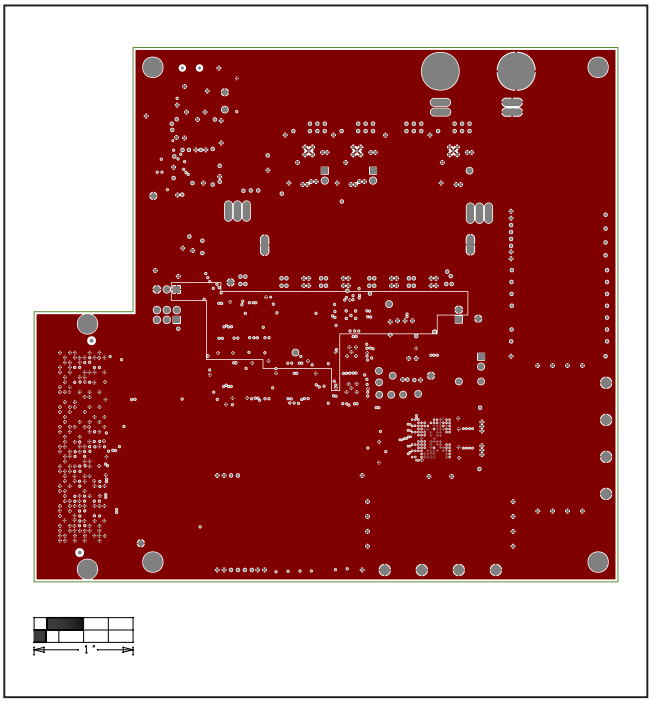
## MAX5868 EV Kit PCB Layout Diagrams (continued)



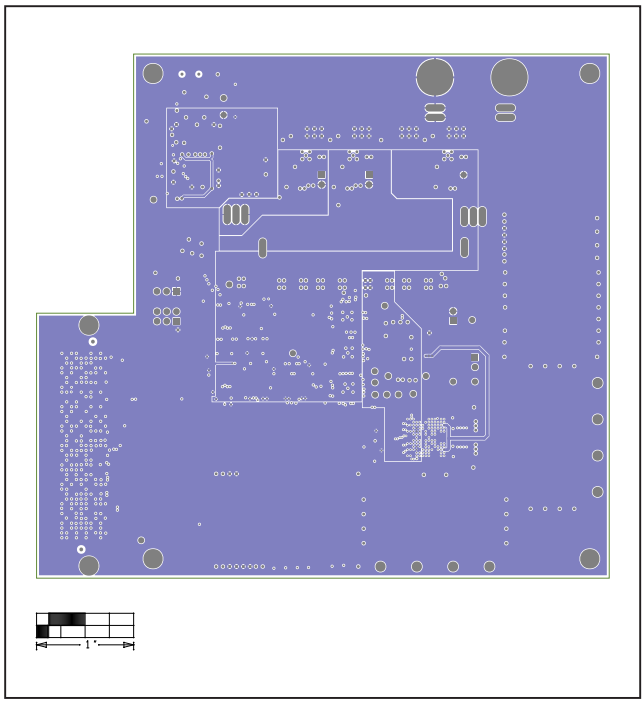
*MAX5868 EV Kit—Level 4 GND*



*MAX5868 EV Kit—Level 5 Power*

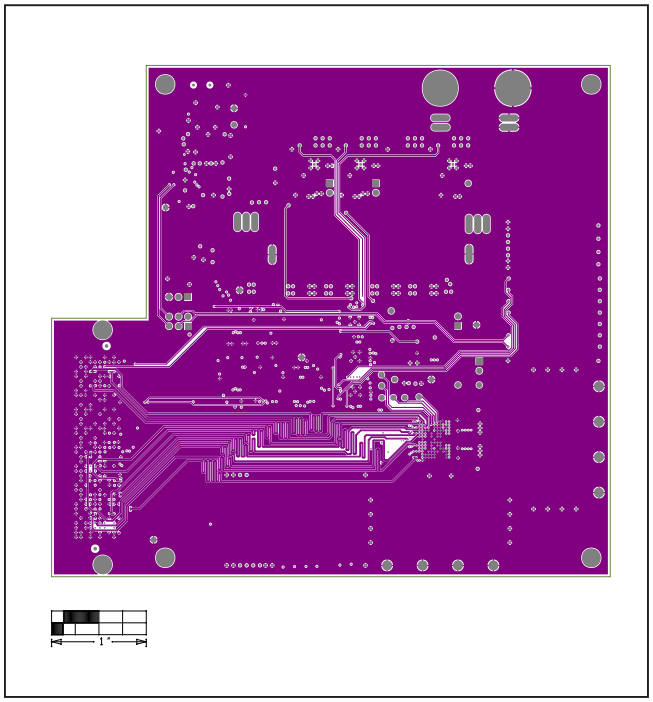


*MAX5868 EV Kit—Level 6 GND*

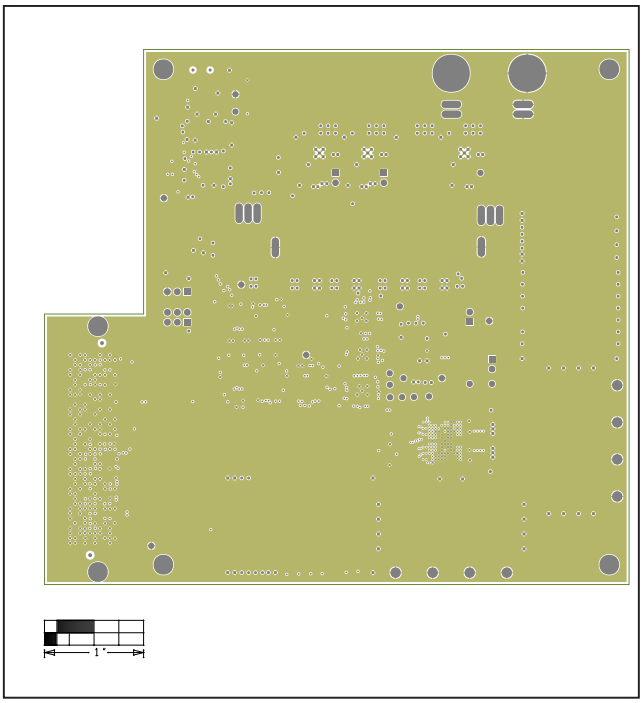


*MAX5868 EV Kit—Level 7 Power*

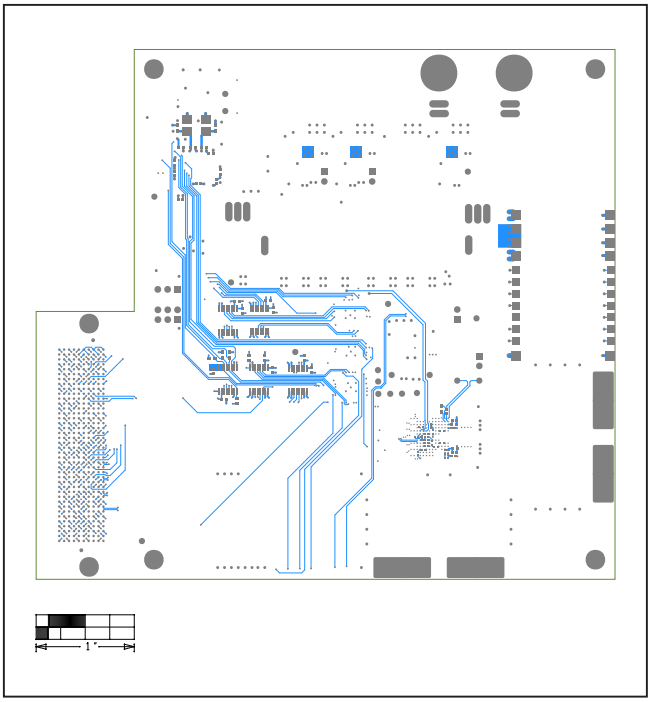
## MAX5868 EV Kit PCB Layout Diagrams (continued)



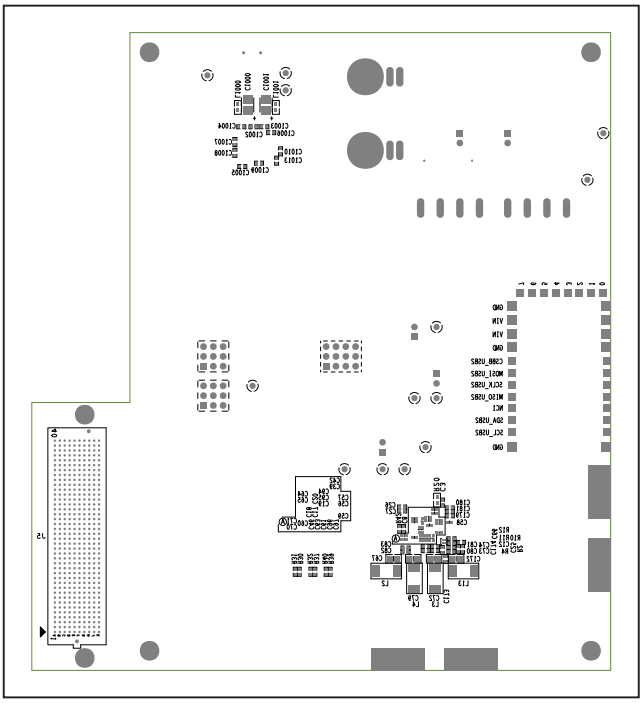
*MAX5868 EV Kit—Level 8 Signal 2*



*MAX5868 EV Kit—Level 9 GND*



*MAX5868 EV Kit—Bottom*



*MAX5868 EV Kit—Bottom Silkscreen*

# Appendix I – Software and Driver References

## Third-Party Software and Driver Installation

For the MAX5868 EV kit software to fully operate with the Xilinx VC707 FPGA platform, software and drivers need to be installed on the target PC. It is highly recommended that the PC be connected to a local area network and have access to the Internet, this will allow for automatic download and updates of some drivers. This process may take 30 minutes or more to complete.

## Xilinx ISE 14.7 LabTools Installation

This is a free tool set used for programming the VC707 Evaluation Board, no software registration or license is required. Xilinx ISE LabTools can be downloaded directly from the web at www.xilinx.com; search for “LabTools 14.7 Download”. Launch the **xsetup.exe** program to install the LabTools on the target PC.

1) Double-click on the **xsetup.exe** program to start installation of the Xilinx ISE LabTools software.
2) Windows may prompt to allow the fallowing program to make changes, click to **Allow Changes**.
3) The ISE Design Suite splash screen will appear, then the Welcome Page, click **Next >** to continue.
4) Two **Accept License Agreement** pages appear, check the boxes and click **Next >** to continue.
5) The **Select Products to Install** page appears, the **Lab Tools – Standalone Installation** should be selected, click **Next >** to continue.

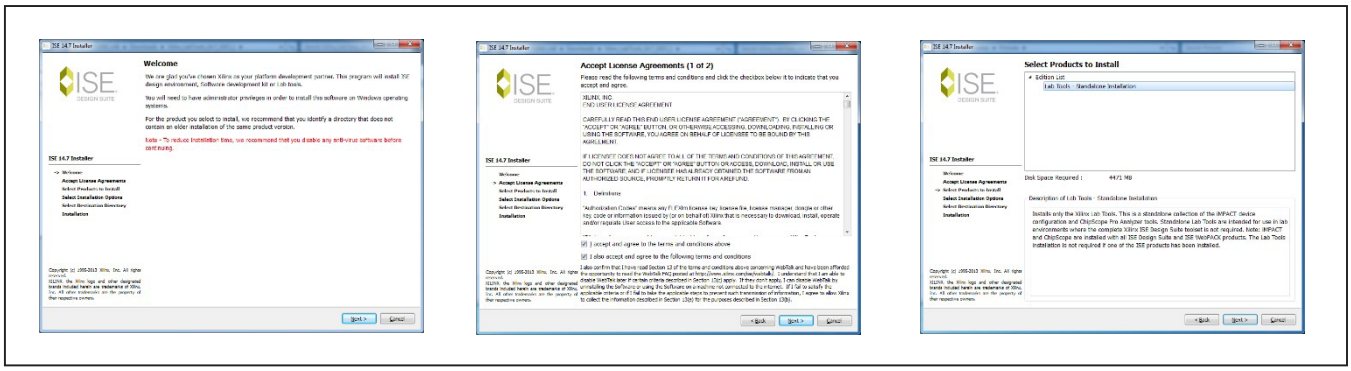



*Figure A1-1. Various LabTools Installation Screens*

6) The **Select Installation Options** page will appear (Figure A1-2), uncheck the **Acquire or Manage a License Key**" option and click **Next >** to continue

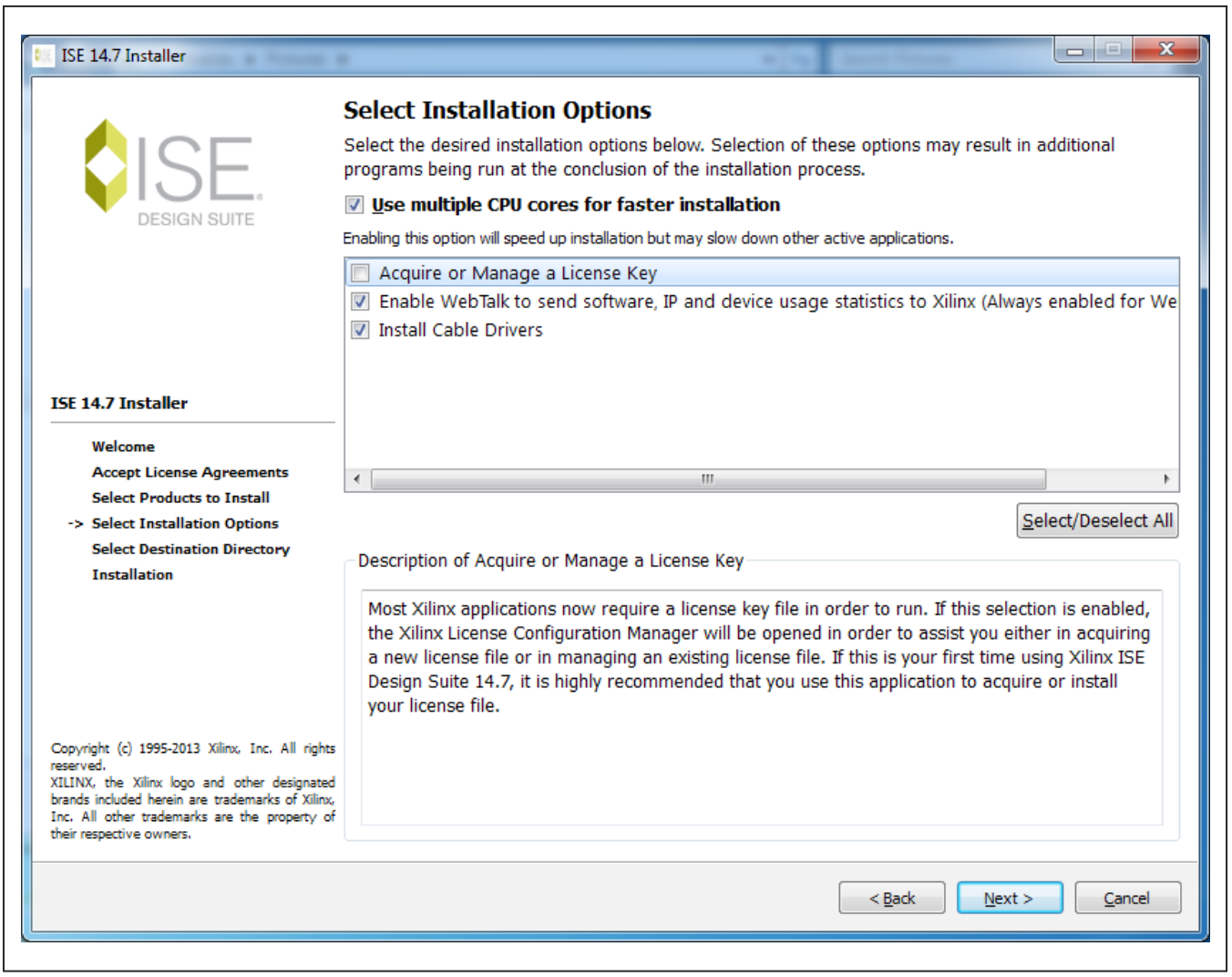



*Figure A1-2. LabTools Installation – Uncheck "Acquire or Manage License Keys"*

7) The **Select Destination Directory** page will appear. *It is highly recommended the default directory be used for the Xilinx LabTools*. Click **Next >** to continue.

8) The **Installation** summary page will appear, click on the Install button to continue.

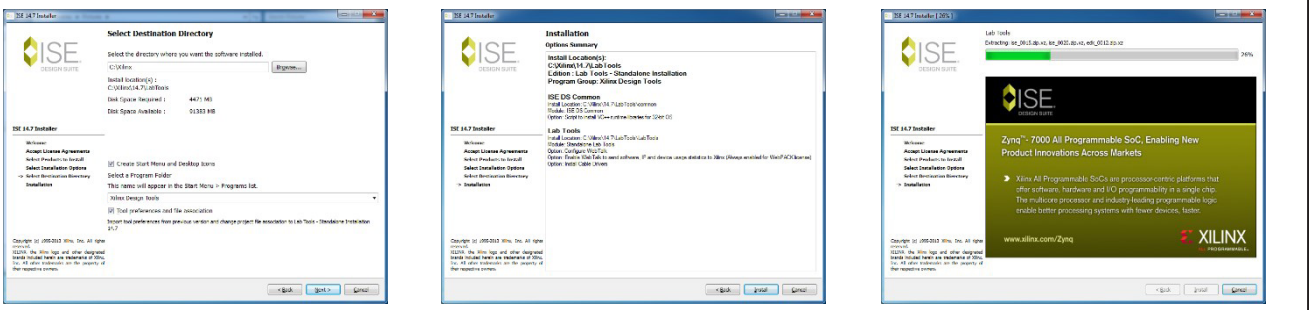



*Figure A1-3. Various LabTools Installation screens*

9) The ISE installation may require Microsoft Visual C++ 2008 Redistribution files, install these during the process if necessary.
   a. On the **…Redistribution Setup** page, click **Next >** to continue.
   b. The **License Terms** page will appear, check the box and click the **Install** button to continue.
   c. When the **Setup Complete** page appears, click the **Finish** button to continue.

10) The installation may also require Jungo software to be installed separately, if prompted by Windows Security, click on the **Install** button to continue.

11) The installer will also be prompted by Windows Security to install **Xilinx** software, click on the **Install** button to continue.

12) A final **Install Completed** page will appear. Click on **Finish** to complete the ISE LabTools installation process.

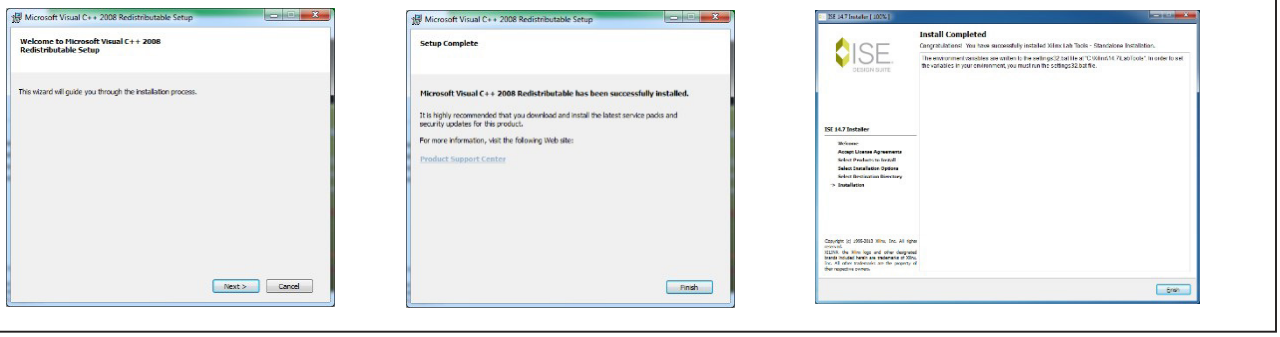



*Figure A1-4. Microsoft Visual C++ Installation and ISE Completion*

## Xilinx Drivers Installation

After the LabTools have been installed on a PC, the VC707 USB interface drivers need to be installed to access the JTAG port.

1) Browse to the Xilinx folder created during the installation of LabTools: **C:\Xilinx\14.7\LabTools\LabTools\bin\nt** (or \nt64)
2) Execute the **install_drivers.exe** application

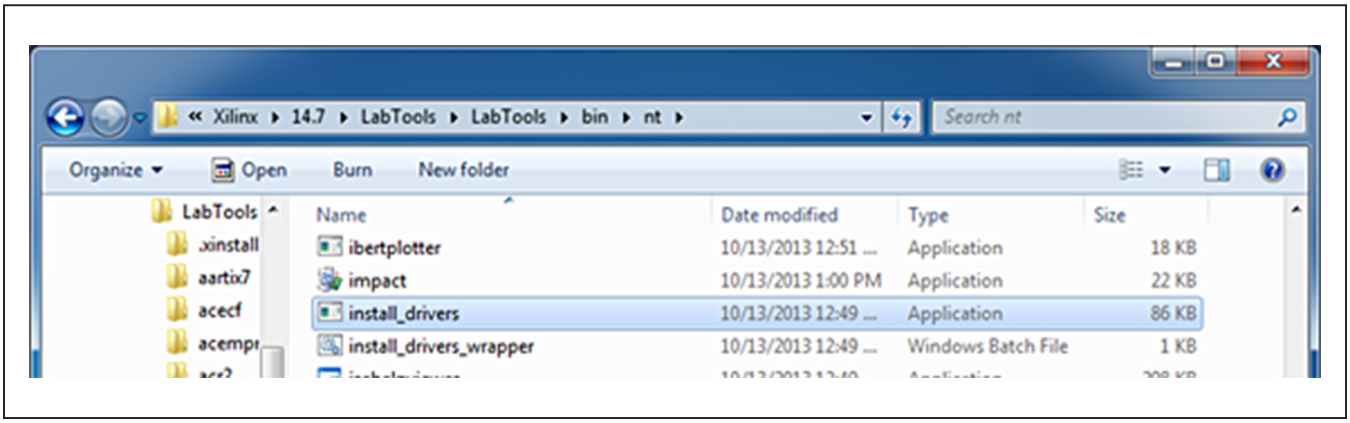



*Figure A1-5. Driver Installation*

3) A Command Prompt window may briefly flash, and a message may appear stating "This program might not have installed correctly", click on **This program installed correctly** to continue.

After these drivers are installed the JTAG port on the VC707 will be registered to the PC's Device Manager.

## Silicon Labs Driver Installation

This driver is needed to properly interface to the USB 2.0 bulk port on the VC707. This port provides fast, bulk data transfers stored in the DDR and used to generate DAC data patterns. The USB 2.0 port is also used to control the FPGA and as a pass-through control port for the MAX5868 SPI.

A **\USB_MS_Bulk_Transfer** folder can be found on the flash drive and is also created during the MAX5868 EV kit software installation. This contains the **libusb-win32 devices** driver for the bulk port. Since the VC707 does not inherently use the USB 2.0 (ULPI) port, this driver can only be installed after the **.bit** file has been programmed into the Virtex FPGA.

Alternatively, the driver and instructions for installation can be downloaded from the SiLabs web site: www.silabs.com. When extracting the files from a downloaded copy, it is recommended the user "extract all files" in order to properly install the driver required for operating the VC707.

## Removal/Uninstallation of the MAX5868 EV Kit Software

For the MAX5868 EV kit software to fully operate with the Xilinx VC707 FPGA platform, software and drivers need to be installed on the target PC. It is highly recommended that the PC be connected to a local area network and have access to the Internet, this will allow for automatic download and updates of some drivers. This process may take **30 minutes** or more to complete.

## Upgrading/Updating the MAX5868 EV Kit Software

Occasionally upgrades or updates may be available for the MAX5868EVKIT GUI software. Be sure to check the Maxim website from time-to-time for information on the MAX5868 evaluation system. Applying upgraded software may require the removal of previous versions.

This document was written based on Revision 1.0.1 (Jul 2017) of the **MAX5868 RF DAC EV Kit Software**.

### Installing Unsigned Drivers on Windows 10

1) Press the **Win + C** and click on **PC settings**.
2) Switch over to the **Update & recovery** section.
3) Click on the **Recovery** option on the left-hand side.
4) Once selected, an advanced startup section appears on the right-hand side. Click on the **Restart now** button.
5) Once the computer has rebooted, choose the **Troubleshoot** option.
6) Go to **Advanced** options.
7) Then **Startup Settings**.
8) Since boot time configuration settings are being modified, restart the computer.
9) Here a list of modifiable startup settings is provided. The applicable option is **Disable driver signature enforcement**. To choose the setting, press the **F7** key.
10) The Windows operating system will now boot and the user will be able to install the unsigned driver. **NOTE:** Driver Signature Enforcement will be reactivated upon the next boot cycle.

## Appendix II – Interface Commands

### List of VC707 Control Commands

| COMMAND | DESCRIPTION |
|---|---|
| help | Prints help. Use -a flag for all commands. Try “help -a” now! |
| usb | Read or write DDR memory with USB bulk transfer |
| reg | Perform a reg read or reg write operation |
| i2c | Perform and I2C Read or I2C write |
| play | Manage play DMA channel |
| ping | Should return #ACK |
| baudrate | Set the baud rate (obsolete?) |
| setIQmsbfirst | Select MSB or LSB first |
| setmode | Select Word, Byte or Nibble data mode |
| selrefclk | Select the reference clock source |
| Setmmcmclk | Configure MMCM based on fDAC and interpolation rate |
| spi | Perform an SPI read or SPI write operation |
| m2870_spi | Perform a SPI read or write operation with the MAX2871 |
| gpio | Perform an GPIO read or GPIO write operation |
| init | Initialization |
| quit | Quits this program |

## Description and Syntax of VC707 Control Commands

### Help

Online help is provided for all commands. Short and long versions are available.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **help** | <command1>,…,<commandN>,<-flags> | help mem<br>help -all |

| ARGUMENTS/FLAGS | DESCRIPTION |
|---|---|
| **<command>** | Name of command, or command and subcommand to request help information about |
| **-all**<br>**-a** | Print help for all the commands |

## USB Operations

The usb commands are for reading and writing the DDR memory used for test data. Read a specific number of bytes from a given address of DDR memory or write a specific number of bytes to a given address of DDR memory. The data are transferred using a USB BULK OUT or USB BULK IN format.

After the proper number of bytes has been received or sent, an ACK will be returned.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **usb read** | [adr] [num bytes] | usb read 0x100 8 |
| **usb write** | [adr] [num bytes] [word1] … [wordN] | usb write 0x100 8 0x64636261 0x68676665 |

| ARGUMENTS / FLAGS | DESCRIPTION |
|---|---|
| **[adr]** | Address to read from or write to, in any format that strtoul will parse |
| **[num bytes]** | Number of bytes to read or write in multiples of 4 |
| **[word]** | 32 bit values in any format that strtoul will parse |

### Register Operations

The reg commands are for reading and writing registers in the MAX5868. Read and write transactions use a 32-bit word. The list operation returns a list of the register or register spaces and when used with the set operation, it allows the user to write to control register fields by name. Only the bits of that field within the control register are modified.

After the proper number of bytes has been received or sent, an ACK will be returned.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **reg read** | [adr] | reg read 0x800 |
| **reg write** | [adr] [word] | reg write 0x800 0xA002007F |
| **reg list** | <reg space> | reg list |
| **reg set** | <reg space> <reg name> [value] | reg set |

| ARGUMENTS / FLAGS | DESCRIPTION |
|---|---|
| **[adr]** | Address to read from or write to, in any format that strtoul will parse |
| **[word]** | 32 bit values in any format that strtoul will parse |
| **<reg space>** | List the hierarchical register spaces, or the registers in a register space. For the set command, <reg space> is optional |
| **<reg name>** | name of the control register field to set |
| **[value]** | value to write to the control register field |

Valid register names are shown in the GUI software under the **Register Access Tab**.

## I²C Operations

The I²C commands are for reading and writing to the I²C bus which only interfaces to the MAX6654 temperature monitor. The read operation will receive one byte from a given register address whereas the read multiple bytes will receive N bytes from the registers starting at the base address provided. The write operation will transmit a single byte to a given register address whereas the write multiple bytes operation will transmit N bytes to the registers starting at the address provided.

After the proper number of bytes has been received or sent, an ACK will be returned.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **i2c read** | [slave-adr] [reg-adr] [Num Bytes] | i2c read 0x08 0x00 8 (read 8 bytes from device address 0x80, starting at register 0) |
| **i2c write** | [slave-adr] [reg-adr] [byte 1] [byte 2]… [byte N] | i2c write 0x08 0x00 0x10 0x20(write 2 bytes (0x10 and 0x20) to device address 0x80, starting at register 0) |
| **i2c recv** | [slave-adr] [reg-adr] [Num Bytes] | i2c recv 0x08 0x00 24 (receive 24 bytes from device address 0x80, starting at register 0) |
| **i2c send** | [slave-adr] [reg-adr] [command-byte] | i2c send 0x80 0x00 (write the byte command 0x00 to device address 0x80) |

| ARGUMENTS / FLAGS | DESCRIPTION |
|---|---|
| **[slave-adr]** | I2C Device Address to read from or write to, in any format that strtoul will parse |
| **[reg-adr]** | Device Register Address to read from or write to, in any format that strtoul will parse |
| **[N bytes]** | Number of bytes to read or write in multiples of 4 |
| **[byte]** | 8 bit value in an format that strtoul will parse |

### Play Operations

The play commands are for configuring, starting, and stopping the play/TX DMA channel.

The buffer operation allows the user to configure the base address or starting address, and length of the play buffer. The base address must start on a 64-byte alignment and be a multiple of 512 bytes. The TX DMA engine must be stopped before using this command. The dumpregs operation will print the play channel registers, effectively dumping the contents of the TX DMA engine.

The start operation will begin the play channel or start the TX DMA engine. The play buffer command must be used first to define the buffer. The stop operation will close the play channel or stop the TX DMA engine.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **play buffer** | [adr] [num bytes] | play buffer 0x400 512 |
| **play start** | | play start |
| **play stop** | | play stop |
| **play reset** | | play reset |

| ARGUMENTS / FLAGS | DESCRIPTION |
|---|---|
| **[adr]** | Address to read from or write to, in any format that strtoul will parse |
| **[num bytes]** | Number of bytes to read or write in multiples of 512 |

### Ping Command

The ping command is a non-operation which simply returns an ACK from the system.

### Baudrate Command

The baudrate command sets the communication rate for the UART interface.

### setIQmsbfirst

The setIQmsbfirst command will configure the MAX5868 to internally operate in MSB first mode.

After the proper number of bytes has been received or sent, an ACK will be returned.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **setIQmsbfirst** | [-y/-n] | -y : set MSB first<br>-n : set MSB last |

### setmode

The setmode command configures the bus width for the data interface between the MAX5868 and the VC707. Both the MAX5868 and the VC707 settings will be modified when changing between modes.

After the proper number of bytes has been received or sent, an ACK will be returned.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **setmode** | [word\|byte\|nibble] | setmode word : use 16-bit bus<br>setmode byte : use 8-bit bus<br>setmode nibble : use 4-bit bus |

### selmmcmclk

The selmmcmclk command configures the bus width for the data interface between the MAX5868 and the VC707. Both the MAX5868 and the VC707 settings will be modified when changing between modes.

After the proper number of bytes has been received or sent, an ACK will be returned.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **selmmcmclk** | [fDAC-in-MHz] [interpol_ratio] [mode] | Selmmcmclk 4915.2 8 word |

| ARGUMENTS / FLAGS | DESCRIPTION |
|---|---|
| **[fDAC-in-MHz]** | The frequency of the DAC clock |
| **[Interpol_ratio]** | The interpolation rate set in the MAX5868 |
| **[mode]** | The bus width mode setting – word, byte or nibble |

### spi Operations

The spi commands are for reading and writing to the SPI bus which interfaces to the MAX5868. The read operation will receive one byte from a given register address. Similarly, the write operation will transmit a single byte to a given register address.

After the proper number of bytes has been received or sent, an ACK will be returned.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **spi read** | [address] | spi read 0x00<br>read the contents of the MAX5868 register 0) |
| **i2c write** | [address] [byte] | spi write 0x00 0x28<br>(write the value 0x28 to register 0 of the MAX5868 |

| ARGUMENTS / FLAGS | DESCRIPTION |
|---|---|
| **[address]** | Register Address to read from or write to, in any format that strtoul will parse |
| **[byte]** | 8 bit value in an format that strtoul will parse |

### m2870_spi Operations

Just like the spi command, m2870_spi commands are for reading and writing to the SPI bus, but target communications with the MAX2871 installed on the Input Clock Module. Command syntax is identical to the spi command.

## GPIO Operations

The gpio commands are for controlling or reading the status of the general purpose I/O lines which are used to drive or are connected to pins on the MAX5868 evaluation board. The resetb operation is active-low signal, using the -y flag will reset the DAC, using the -n flag will disable reset. The mute operation is an active-high signal, using the -y flag will mute the DAC, using the -n flag will unmute the DAC. The intb line is connected to the active-low interrupt pin of the DAC, using the intb operation will read the status of the INTB pin. The altb line is connected to the alarm (alert) pin of the MAX6654, using the altb operation will read the status of the ALARMB pin of the temperature sensor.

| COMMAND | SYNTAX | EXAMPLES |
|---|---|---|
| **gpio resetb** | [-y\|-n] | gpio restb -y |
| **gpio mute** | [-y\|-n] | gpio mute -n |
| **gpio intb** | | gpio intb |
| **gpio altb** | | gpio altb |
| **gpio spisel** | [-y\|-n\|read] | gpio spisel -y |
| **gpio extdiv** | [-y\|-n\|write\|read] [value] | gpio extdiv write 7 |
| **gpio pok** | [-y\|-n\|read] | gpio pok read |

| ARGUMENTS / FLAGS | DESCRIPTION |
|---|---|
| **[-y\|-n]** | Set (-y) or clear (-n) the appropriate line |
| **[read]** | Read the current state of the control |
| **[write]** | Write a new value |
| **[value]** | extdiv is stored as a 3-bit value, value must be in the range of 0 to 7 decimal |

## Init Command

The init command is used to initiate communication with the FPGA.

## Quit Command

The quite command closes communication with the FPGA.

# Appendix III – Pattern Files

## Creating Pattern Files

The MAX5868 EVKIT Software Controller is provided with a limited number of sample patterns. The provided patterns allow the user to generate a two-tone, CW signal with a 1MHz spacing between the tones. Typically, the user will want to generate signals with other properties, including modulated signals.

A \matlab folder is included in the MAX5868\PatternFiles folder, which contains sample MATLAB routines (.m files) which allow a user with MATLAB installed to create additional continuous wave, sinusoidal test patterns. The makesignal.m routine will perform the CW generation, providing a complex vector with a range of ±1.0. The set_datascale_avg.m routine is used to convert the complex vector into integer representations of the offset binary values for the I and Q data paths. Finally, the scaled pattern is stored to a file using the MAX5868_PatFile.m routine which formats the pattern appropriately for use with the VC707. Please refer to the comments in these files for additional information. Lines that begin with % following the Copy Right statement provide information about the input arguments used by these routines.

The pattern file can use any extension, such as csv, txt, etc., as long as the contents are simple text and conform to the format expected by the MAX5868 EV kit Software Controller.

The specific format is as follows:

- The first line contains the total number of I/Q data points, N, in the pattern.
- The total number of lines in the pattern file will be N + 1.
- The second through N lines contain four, comma separated integer values. These values represent, in order:
  - the I data Least Significant BYTE (LS BYTE)
  - the I data Most Significant BYTE (MS BYTE)
  - the Q data LS BYTE
  - the Q data MS BYTE.

**Note:** I/Q data is in decimal format, offset binary with the 14-bits of data being LSB-justified. That is, the MS BYTE values can range from 0 to 63 and the LS BYTE values can range from 0 to 255.

For example, the first few lines of the file could look something like this:

```
65536
19, 62, 253, 37
19, 62, 250, 37
17, 62, 248, 37
15, 62, 245, 37
13, 62, 242, 37
10, 62, 239, 37
7, 62, 236, 37
3, 62, 234, 37
```

# Ordering Information

| PART | TYPE |
|---|---|
| MAX5868EVKIT# | EV Kit |
| EK-V7-V707-G | Xilinx Virtex 7 FPGA Board* |

*#Denotes RoHS compliant.*

**Order from Xilinx or authorized distributor.*

## Revision History

| REVISION NUMBER | REVISION DATE | DESCRIPTION | PAGES CHANGED |
|---|---|---|---|
| 0 | 10/17 | Initial release | — |

For pricing, delivery, and ordering information, please contact Maxim Direct at 1-888-629-4642, or visit Maxim Integrated's website at www.maximintegrated.com.

*Maxim Integrated cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim Integrated product. No circuit patent licenses are implied. Maxim Integrated reserves the right to change the circuitry and specifications without notice at any time.*