

How to Bypass the **ADF4371** and **ADF4372** VCOs Autocalibration

by Kazim Peker

INTRODUCTION

The [ADF4371](#) and the [ADF4372](#) are wideband synthesizers with integrated voltage controlled oscillators (VCOs) that use autocalibration to choose the appropriate core select, band select, and bias level calibration values to lock the device. However, disabling the autocalibration feature provides faster locking times.

Autocalibration is bypassed by using Register 0x0034, Bits[7:5] (VCO_FSM_TEST_MODES) to force the VCO core, band, and bias to known values. To bypass autocalibration, lock the device for the target frequency with autocalibration enabled and read back and store the VCO core, band, and bias that has been

selected by the autocalibration feature. Next, build a table of these values for each frequency of interest. During operation, disable autocalibration and manually load the VCO values for the target frequency to force the device to select the correct frequency band and eliminate the additional time taken by the autocalibration feature.

The [Analysis, Control, Evaluation \(ACE\)](#) software plug-in programs the [EV-ADF4371SD2Z](#) and the [EV-ADF4372SD2Z](#), facilitates the readback and temporary storage of the VCO values, and manually writes these values to the device.

TABLE OF CONTENTS

Introduction	1	Using ACE to Bypass Autocalibration	4
Revision History	2	How to Access and Use the VCO CAL Bypass Tab	4
Autocalibration Bypass Procedure	3	Temperature Readback Procedure	6
VCO Calibration Data Readback	3		
Manual VCO Calibration Data Writing	3		

REVISION HISTORY

6/2019—Revision 0: Initial Version

AUTOCALIBRATION BYPASS PROCEDURE

Before operation, lock the device and generate a lookup table by reading back the VCO core, band, and bias parameters. Generate a new table for every chip because each chip is unique. During operation, the VCO parameters are manually written to the device by disabling the autocalibration feature.

VCO CALIBRATION DATA READBACK

To read back the VCO calibration data, load the required registers, let the device lock as usual using autocalibration, and read the VCO parameters for each frequency. It is important to ensure that autocalibration has completed before readback. Reading back values before autocalibration has completed will result in incorrect values being read.

The bits used for readback are labeled as follows in the [ACE](#) software:

- Register 0x0033, Bits[7:5], VCO_FSM_READBACK
- Register 0x006E, Bits[7:0], VCO_DATA_READBACK[7:0]
- Register 0x006F, Bits[7:0], VCO_DATA_READBACK[15:8]

The VCO_FSM_READBACK bits set what data is sent to the VCO_DATA_READBACK bits.

To read the VCO parameters, take the following steps:

1. Program the device to lock at the desired frequency by using the autocalibration feature. Users must wait for the device to lock.
2. Set the VCO_FSM_READBACK bits to 0x01 to allow readback of the band and the core.
3. Read Register 0x006F, Bits[3:0] to readback the current VCO core. See Table 1 to match the data with the corresponding core.
4. Read Register 0x006E, Bits[7:0] to readback the VCO band.
5. Set the VCO_FSM_READBACK bits to 0x03 to allow readback of the bias.
6. Read Register 0x006E, Bits[3:0] to readback the VCO bias.

Repeat Step 1 through Step 6 for each required frequency to build a table of these values.

Table 1. VCO Core Readback Data

Core	Register 0x006F Data
Core D	'b0001
Core C	'b0010
Core B	'b0100
Core A	'b1000

Table 2. VCO Core Write Data

Core	SI_VCO_SEL Data
Core D	0x01
Core C	0x02
Core B	0x04
Core A	0x08

MANUAL VCO CALIBRATION DATA WRITING

The VCO parameters for each required frequency force the device to the target frequency band without the use of autocalibration.

The bits used for writing to the VCO parameters are the following in the [ACE](#) software:

- Register 0x0034, Bits[7:5], VCO_FSM_TEST_MODES
- Register 0x0037, Bits[7:0], SI_BAND_SEL
- Register 0x0038, Bits[7:4], SI_VCO_SEL
- Register 0x0038, Bits[3:0], SI_VCO_BIAS_CODE

To write the VCO parameters to the device, take the following steps:

1. At power-up, set up the serial port interface and initialize the device as explained in the [ADF4371](#) and [ADF4372](#) data sheets. Ensure that EN_AUTOCAL is set to 'b0 to disable autocalibration and that Register 0x0034, Bits[7:5] (VCO_FSM_TEST_MODES) is set to 'b1 to overwrite the VCO core, band, and bias.
2. Program the registers, except Register 0x0010, as required for the target frequency. This step is frequency dependent. It may be necessary to program the integer, fractional, and modulus words, bleed current, and tracking filter coefficients, and to enable or disable integer mode, auxiliary modulus, output divider, doubler, and quadrupler.
3. Set the core by using the SI_VCO_SEL data listed in Table 2.
4. Set the band by using the SI_BAND_SEL bits. The data can be written to these bits in the same format collected from the readback.
5. Set the bias by using the SI_VCO_BIAS_CODE bits. Again, the data can be written to these bits in the same format collected from the readback.
6. Write to Register 0x0010. When this register is written to, the device locks to the new frequency.

Repeat Step 2 through Step 6 for any frequencies the VCO calibration data collected.

USING ACE TO BYPASS AUTOCALIBRATION

HOW TO ACCESS AND USE THE VCO CAL BYPASS TAB

1. Open the ACE software with the EV-ADF4371SD2Z or the EV-ADF4372SD2Z connected, as explained in the respective user guides. The ACE main control window is shown in Figure 1.

2. In the ACE main control window, click **Other Controls** to open the VCO Cal Bypass pane (see Figure 2). The VCO Cal Bypass pane contains the controls to make a slow frequency sweep and to collect and write the calibration data to the device.

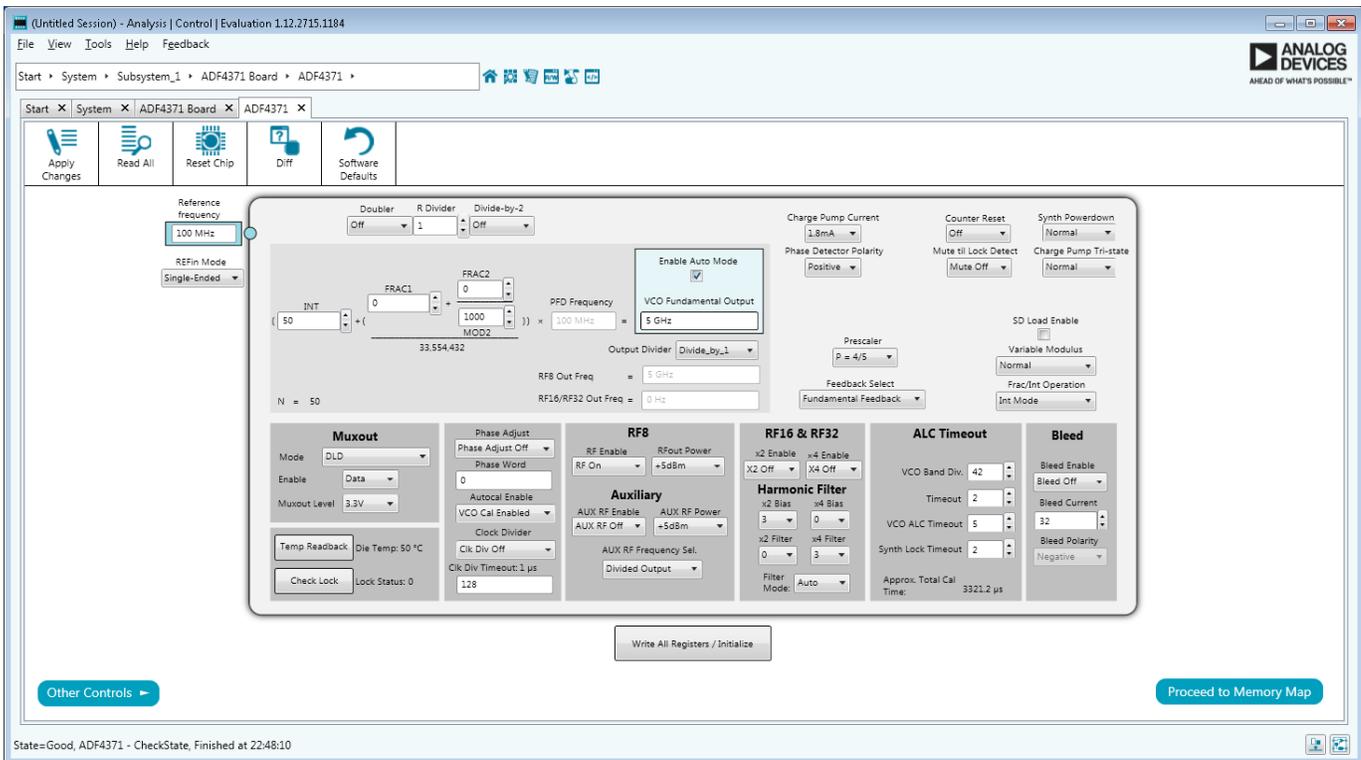


Figure 1. ACE Main Control Window

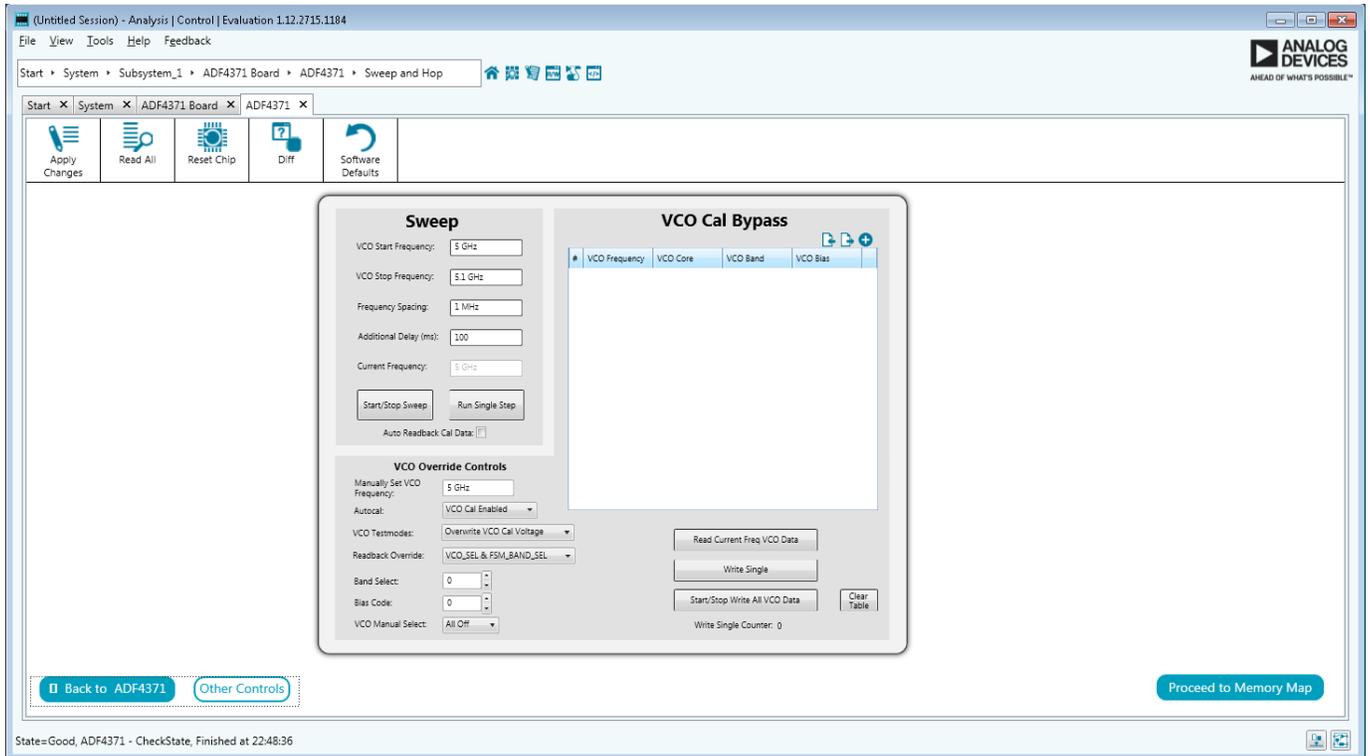


Figure 2. VCO Cal Bypass Pane

- During the frequency sweep, calibration data is automatically collected by using the following steps:
 - Program the device to the required frequency.
 - Under **Sweep**, check off the **Auto Readback Cal Data** check box.
 - Set up the required sweep parameters for the calibration data by filling in the **VCO Start Frequency**, **VCO Stop Frequency**, **Frequency Spacing**, and **Additional Delay** boxes.
 - Click **Start/Stop Sweep** for the software to automatically run through the sweep and to measure the calibration data.

To manually read the calibration data, take the followings steps:

- Take a reading of the current calibration data when the device is locked. Click **Read Current Freq VCO Data** to read the VCO core, band, and bias and to display the data under the **VCO Cal Bypass** pane.

Note that this process can be repeated for other frequencies and set ups. To change the frequency manually, use the **Manually Set VCO Frequency** text box.

- After the required data is collected and displayed in the **VCO Cal Bypass** pane, click either **Write Single** or **Start/Stop Write All VCO Data** to write to the device. Both of these command buttons automatically disable the autocalibration feature and manually write the core, band, and bias parameters to the device for each corresponding frequency. The **Write Single** and **Start/Stop Write All VCO Data** commands take the following steps:

- For every click, **Write Single** steps through one of the rows in the table. The current row is displayed next to the **Write Single Counter** in the bottom right.
- Start/Stop Write All VCO Data** runs through and writes each row of data automatically. Until this button is clicked again, the writing process repeats.

TEMPERATURE READBACK PROCEDURE

Chip temperature can provide information regarding system temperature. Also, the chip temperature can be used for temperature compensation.

The [ADF4371](#) and the [ADF4372](#) include an analog-to-digital converter (ADC) that enables reading the chip temperature. The ADC clock (ADC_CLK) is generated from the phase frequency detector clock (f_{PFD}) with the following equations:

$$ADC_CLK = \frac{f_{PFD}}{((ADC_CLK_DIV \times 4) + 2)}$$

where:

ADC_CLK_DIV is stored in Register 0x0035.

$$f_{PFD} = REF_{IN} \times \left(\frac{(1 + D)}{(R \times (1 + T))} \right)$$

where:

REF_{IN} is the reference frequency input.

D is the REF_{IN} doubler bit.

R is the reference division factor.

T is the reference divide by 2 bit (0 or 1).

A valid reference signal is required to complete a conversion. Target 100 kHz for ADC_CLK and calculate ADC_CLK_DIV with the following equation:

$$ADC_CLK_DIV = \text{ceiling} \left(\frac{\left(\left(\frac{f_{PFD}}{100,000} \right) - 2 \right)}{4} \right)$$

If ADC_CLK_DIV is greater than 255, set it to 255.

The bits used for temperature readback are the following in the [ACE](#) software:

- Register 0x0032, Bit 2, ADC_ENABLE
- Register 0x0032, Bit 3, ADC_CONVERSION
- Register 0x0033, Bits[7:5], VCO_FSM_READBACK
- Register 0x006E, Bits[7:0], VCO_DATA_READBACK[7:0]
- Register 0x0073, Bit 2, ADC_CLK_DISABLE

To read back the temperature, take the following steps:

1. Ensure that the ADC_ENABLE bit is enabled.
2. Enable the ADC_CONVERSION bit by setting Register 0x0032, Bit 3 to b1.
3. Wait 16 ADC_CLK cycles.
4. Set the VCO_FSM_READBACK bits to 0x05 (skip this step if it is already set).
5. Read the VCO_DATA_READBACK[7:0] bits to readback the raw ADC output that corresponds to the chip temperature (RAW_TEMP).
6. Disable the ADC_CONVERSION bit by setting Register 0x0032, Bit 3 to b0.
7. Disable the ADC_ENABLE bit by setting Register 0x0032, Bit 2 to b0, which prevents any spurs generated by the ADC clock. Similarly, the ADC_CLK_DISABLE bit can disable the ADC clock.

Perform Step 1 and Step 2 separately. However, Step 6 and Step 7 can be done together.

Use the following equation to calculate the actual chip temperature in Celsius (°C):

$$\text{Chip Temperature} = -83.5^{\circ}\text{C} + RAW_TEMP$$