AD9361 Initialization and Factory Calibration Guide

TABLE OF CONTENTS

REVISION HISTORY

4/2010 — Rev 0.1 Initial Document

4/2010 — Rev 0.2 Updated TX Secondary Filter example code

4/2010 — Rev 0.3 Added max calibration time for BBPLL VCO calibration

4/2010 — Rev 0.4 Corrections to calibration time equations

10/2010 — Rev 0.5 Added single shot RX Quadrature Calibration procedure and ADC equations.

3/2011 — Rev 1.0 Updated FDD register – moved to register 0x013[0]. RF VCO calibration updated.

4/2011 — Rev 1.1 RF VCO calibration updated to show example calibration times.

6/2011 — Rev 2.0 General updates

9/2011 — Rev 2.1 General updates before final review

10/2011 — Rev 2.2 General formatting updates

2/2012 — Rev 2.3 Removed TX Monitor DC Offset calibration (unnecessary) and updated TX Quadrature Calibration details

OVERVIEW

The AD9361 powers up into a SLEEP state for minimal power consumption. Before the AD9361 is operational, its clocks must be enabled and initial calibrations completed as shown in the example scripts generated from the AD9361 Evaluation Software. The purpose of this document is to describe in detail the operation of the different initialization calibrations as well as suggested factory calibrations. Table 1 below lists the initialization calibrations documented in this guide. Table 2 below shows the factory calibrations suggested in this guide.

#	Calibration	Run Frequency	Calibration Done Bit
1	BBPLL VCO Calibration	Once, Any time BBPLL Frequency changes	0x5E[7], 1 when locked
2	RF Synthesizer Charge Pump Calibration	Once	RX: 0x244[7], 1 when done TX: 0x284[7], 1 when done
3	RF Synthesizer VCO calibration	Occurs automatically when integer frequency word written. In TDD, occurs when TXNRX changes logic level.	RX: 0x247[1], 1 when locked TX: 0x287[1], 1 when locked
4	Baseband RX Analog Filter Tune	Once, update when BW changes	0x016[7], Self clears when done
5	Baseband TX Analog Filter Tune	Once, update when BW changes	0x016[6], Self clears when done
6	Baseband TX Secondary Filter Tune	Once - Manual Equations, update when BW changes	
7	RX TIA Calibration	Once - Manual Equations, update when BW changes	
8	RX ADC Setup	Once – Manual LUT or equations, update when ADC sampling rate changes	
9	Baseband DC Offset	Once	0x016[0], Self clears when done
10	RF DC Offset	Any time LO Frequency changes more than 100 MHz	0x016[1], Self clears when done
11	RX Quadrature Calibration	Any time LO Frequency changes more than 100 MHz	Tracking runs continuously
12	TX Quadrature Calibration	Any time LO Frequency changes more than 100 MHz	0x016[4], Self clears when done

Table 1: Initialization Calibrations Detailed in this Document

#	Factory Calibration
1	Internal DCXO (AFC tune range)
2	TX RSSI (TX Monitor)
3	RX RSSI (Absolute Power Correlation)
4	RX GM / LNA Gain Step Error
5	TX Power out Vs TX attenuation
6	TX Power out Vs Frequency

Table 2: Factory Calibrations Detailed in this Document

INITIALIZATION CALIBRATIONS

Initialization calibrations are calibrations that must be run each time the AD9361 device is powered up or hard reset using the RESETB pin. Several of the calibrations only need to run once during initialization. Others are dependent on the carrier frequency, temperature, or other parameters and need to run initially and when certain events occur (such as changing the carrier frequency more than 100 MHz). As long as power is applied to the AD9361 device, the calibration results are stored in the SPI register map, including while in the SLEEP state. Calibrations should be run in the order shown in the example scripts generated from the AD9361 Evaluation Software.

Reg	Name	D7	D6	D5	D4	D3	D2	D1	D0
0x016	Calibration Control	RX BB Tuning (self clear)	TX BB Tuning (self clear)	RX Quad cal (self clear)	TX Quad cal (self clear)	RX Gain Step Cal (self clear)	TX Monitor DC cal (self clear)	RF DC cal (self clear)	Baseband DC cal (self clear)
0x017	STATE	calibration sequence state[3:0]				ensm_state[3:0]			

Table 3: AD9361 Registers to Start/Monitor Calibrations

The six calibrations in 0x016[5:0] are part of a calibration sequence state machine and are enabled by setting the corresponding start bit in register 0x016. After a calibration completes, the corresponding bit in 0x016 will self clear. If more than one calibration is enabled in a single register write, the calibrations will progress in a set order controlled by a state machine in the AD9361. The table below shows the sequence of calibrations. When the calibration sequence state (0x017[7:4]) holds a value of 0x1, the calibrations are complete. Before a calibration is run, register settings for that calibration should be written as shown in a generated initialization script from the AD9361 Evaluation Software. Some calibrations depend on the results of previously run calibrations. The RX baseband filter and TX baseband filter calibrations in 0x016[7:6] are not part of the calibration sequence state machine, and should run only when all other calibrations are NOT running.

Calibration sequence state[3:0]	Active Calibration
0x0	Calibration WAIT State
0x1	Calibrations Done
0x2	Baseband DC Offset Calibration
0x3	RF RX DC Offset Calibration
0x4	TX1 Quadrature Calibration
0x5	TX2 Quadrature Calibration
0x6	RX1 Quadrature Calibration
0x7	RX2 Quadrature Calibration
0x8	TX Monitor Calibration (DC Offset)
0x9	RX GM \ LNA Gain Step Calibration
0xA – 0xF	Flush states

Table 4: Automatic Calibration Sequence and Calibration Status

BBPLL VCO CALIBRATION

The BBPLL VCO calibration must be run during initialization of the AD9361 device. The VCO calibration can be disabled by writing register 0x04B=0x40 (clearing 0x04B[7]). When 0x04B[7] is set, a calibration will start when 0x03F[2] is set. The calibration start bit in register 0x03F[2] does not self clear, and must be manually cleared before another calibration can be started. The maximum BBPLL VCO calibration time is shown below. Register 0x05E[7] will equal 1 when the BBPLL is locked.

$$\mathrm{BBPLL}_{\mathrm{maxVCOcalTime}} = \frac{1}{\frac{\mathrm{REF\_CLK\_IN}}{\mathrm{DivideSetting}} \times \mathrm{Scale}} \times 3 \times (\mathrm{Counter} + 128)$$

Scale is the value selected in register 0x045[1:0]. Counter is derived from the value in register 0x04B[6:5].

0x045[1:0]	Scale	0x04B[6:5]	Counter	0x4E[4]	Divide Setting
00	1	00	128	0	1
01	?	01	256	1	4
10	?	10	512		
11	2	11	1024		

Table 5: BBPLL VCO Cal Time Variables

RF SYNTHESIZER CHARGE PUMP CALIBRATION

The charge pump calibration must be run once during initialization of the AD9361 device. This calibration matches the up and down currents for the RF PLL’s charge pump. This calibration must be run the first time the AD9361 device enters the ALERT state. The calibration completes after a maximum of 36864 (REF_CLK_IN * Scale) cycles. The Scale parameter is shown in the table below.

{0x2AB[0], 0x2AC[7]}	RX REFCLK Scale	0x2AC[3:2]	TX REFCLK Scale
00	1	00	1
01	?	01	?
10	?	10	?
11	2	11	2

Table 6: RX and TX Synth REFCLK Frequency Scale Settings.

With a 40MHz REF_CLK_IN, the calibration would complete in less than 921.6us. Please note the sequence of register writes in the script generated by the AD9361 Evaluation Software for a detailed order of SPI Writes. In general, follow this typical sequence:

1) Set the device up for FDD to ensure both synthesizers are powered up when in ALERT.
   a. ENSM FDD bit set in 0x013[0].
   b. Dual Synthesizer bit set in 0x015[2].
2) Move the device into the ALERT state.
3) Start the RX CP calibration for the RX synthesizer by setting 0x23D[2].
4) Wait until the CP Cal Valid bit goes high in register 0x244[7].
5) Clear the RX CP calibration enable bit in register 0x23D[2].
6) Start the TX CP calibration for the TX synthesizer by setting 0x27D[2].
7) Wait until the CP Cal valid bit goes high in register 0x284[7].
8) Clear the TX CP Calibration enable bit in register 0x27D[2].

If using TDD, finish the remaining initialization calibrations before setting the ENSM and Dual Synth bit for TDD mode to simplify the calibration process.

RF SYNTHESIZER VCO CALIBRATION

The AD9361 contains two synthesizers. When using TDD mode, the RX synthesizer is only enabled in the RX state and when in the ALERT state while TXNRX is low. The TX synthesizer is only enabled in the TX state and when in the ALERT state while TXNRX is high. During initial calibrations, it is recommended to set the AD9361 device into FDD mode to enable both synthesizers while in the ALERT state to simplify calibrations. Before running the RF synthesizer VCO calibrations, set the synthesizer and loop filter registers provided by the AD9361 Evaluation Software. The AD9361 Evaluation Software pulls these setup register values from a LUT based on the VCO frequency.

The VCO calibration starts when the integer frequency word is written for a specific synthesizer (register 0x231 for RX, register 0x271 for TX). First, set up any synthesizer setup registers, then write the fractional frequency words, followed by the integer frequency word writing 0x231 and 0x271 last. The calibration time can be traded off with calibration accuracy. It is recommended for FDD applications, to use the longest calibration for better accuracy since once in the FDD state, it may be a long time before a synthesizer VCO calibration occurs again. In TDD, the calibration time will need to be set in order to meet the TDD turnaround time, while achieving the most accurate calibration possible. In TDD, the RX VCO calibration will occur each time the receiver synthesizer is powered up (when TXRNX switches from high to low logic level). The TX VCO calibration will occur each time the transmitter synthesizer is powered up (when TXNRX switches from low to high logic level).

$$RFPLL_{maxVCOcalTime} = wait_1 + wait_2 + \frac{N_{ALC} + N_{count}}{REF\_CLK\_IN \times Scale_{table\ 6}} + wait_{ALC} \times 9$$

Where,

????? ????0?024,????????????? ????0?025,????????????? ????????? $\frac{8}{????\_??????????}$ + $\frac{18}{(??????\_??????????)}$?????????? ????????? $\frac{?1 + ?0?249[6:4], ????????? \times 40 / 0?289[6:4], ????????}{????\_???????????????????}$

???? = 6 × 2 ^ ?1+ ?0?23?[7], ???????? ; 0?27?[7], ???????? , ???? = 2 ^ ?7+ ?0?249 [3:2], ???????? ; 0?289 [3:2], ????????

0x23D[7]	0x249[6:4]	0x249[3:2]	REFCLK	Scale	wait $_1$(us)	wait $_2$(us)	wait $_{ALC}$ (us)	N $_{ALC}$	N $_{count}$	Calibration Time (us)
0	0	3	19.20	1	2	1.354	2.083	12	1024	507.729
0	0	3	30.72	1	2	0.846	1.302	12	1024	318.081
0	0	3	40.00	1	2	0.650	1.000	12	1024	244.750
0	0	3	19.20	2	2	0.885	1.042	12	1024	255.073
0	0	3	30.72	2	2	0.553	0.651	12	1024	160.171
0	0	3	40.00	2	2	0.425	0.500	12	1024	123.475

Table 7: Example Calculated VCO Calibration Times for FDD Default Settings

0x23D[7]	0x249[6:4]	0x249[3:2]	REFCLK	Scale	wait $_1$(us)	wait $_2$(us)	wait $_{ALC}$ (us)	N $_{ALC}$	N $_{count}$	Calibration Time (us)
0	0	1	19.20	1	2	1.354	2.083	12	128	87.729
0	0	1	30.72	1	2	0.846	1.302	12	128	55.581
0	0	1	40.00	1	2	0.650	1.000	12	128	43.150
0	0	1	19.20	2	2	0.885	1.042	12	128	45.073
0	0	1	30.72	2	2	0.553	0.651	12	256	47.671
0	0	1	40.00	2	2	0.425	0.500	12	256	37.075

Table 8: Example Calculated VCO Calibration Times for TDD Defaults

The ADI evaluation software default calibration settings are highlighted in blue in the tables above. Please note that the calculated calibration time does not include the synthesizer lock time after the calibration is complete. The Synth LUT used determines the VCO Kv and will dictate the N_{count} needed for an accurate RF VCO calibration.

The VCO calibrations can be masked (disabled) for certain cases such as the Fast Lock synthesizer mode, or when an HFDD application is desired by setting 0x230[0] for RX and 0x270[0] for TX. Using the FDD Synth LUT could be used to acquire a temperature stable lock for cases where there is not time to run the VCO calibration in TDD.

For applications where the RF frequency needs to be corrected in small frequency steps, a RF Frequency correction word can be written in register 0x24E and 0x24F for the RX synthesizer, or 0x28E and 0x23F for the TX synthesizer. Writing the correction word will not start a VCO calibration.

Calibration completion can be detected by reading the RX PLL lock bit in register 0x247[1]. The TX PLL lock bit is located in 0x287[1]. The lock bits will read logic 1 when the PLLs are locked.

BASEBAND RX ANALOG FILTER CALIBRATION

The baseband RX analog filter calibration tunes the cutoff frequency of the 3rd order Butterworth RX anti-aliasing filter. The RX filter is located just before the ADC in the RX signal path. This calibration is important for RX interferer rejection. The calibration time depends on the RX filter tune clock frequency. The RX filter tune clock frequency is calculated based on the desired baseband bandwidth. NOTE: The BBBW is half the complex bandwidth and coerced between 28 MHz to 0.20 MHz for the equations used in this filter tuning.

$$Desired\_f_{RXTuneCLK} = \frac{1.4 * BBBW * 2\pi}{\ln(2)}$$

To generate this RXTuneCLK, the BBPLL is divided down using a divide by 1 to 511 divider dedicated to the RX tuner block. Before starting the RX baseband filter tune, set this divider value using the following equation.

$$RXBBFDivide[8:0] = \min(511, ceiling(\frac{BBPLL\_Freq}{Desired\_f_{RXTuneCLK}}))$$

Typical Sequence:

1) Follow the order of events in the scripts generated by the AD9361 Evaluation Software.
2) AD9361 should be in the ALERT state and previously discussed calibrations already run. Verify that no other calibrations are currently running.
3) Set RX baseband filter divide value in register 0x1F8[7:0] and 0x1F9[0]. Be sure to retain the correct settings in 0x1F9[7:1].
4) Write the BBBW into registers 0x1FB and 0x1FC. 0x1FB holds the MHz portion, and 0x1FC holds the KHz portion. Register 0x1FB has a step size of 1MHz/LSB, while 0x1FC is written using the following equation :

$$RXTuneBBBW\_KHz[6:0] = \min(127, Round(\frac{(BBBW_{MHz} - Floor(BBBW_{MHz})) * 1000}{7.8125}))$$

Setting registers 0x1FB and 0x1FC is necessary to ensure adequate stability and linearity of the RX baseband filter.

5) Enable the RX BBF tune circuit by writing 0x1E2=0x02 and 0x1E3=0x02.
6) Start the RX Baseband Filter calibration in register 0x016[7].
7) Calibration is complete when register 0x016[7] self clears.
8) Disable the RX baseband filter tune circuit, write 0x1E2=3, 0x1E3=3.

The RX baseband analog filter calibration does not run automatically, and must be manually triggered by writing SPI register 0x016[7] in the ALERT state. Calibration completion can be monitored on a control out pin or by reading register 0x016[7] until the RX baseband filter calibration bit self clears. The calibration is a binary search algorithm with a maximum calibration time of 610 RX Tune clock cycles.

Standard	BBBW	BBPLL Frequency (MHz)	RXBBF Divider[8:0] (decimal)	Actual RX Tune Clock Frequency (MHz)	Max Calibration time (us)
WiMax 3.5 MHz	1.75	1024	58	22.069	27.641
WiBRO 4.375 MHz	2.1875	1280	47	27.234	22.398
WiMax 5 MHz	2.5	716.8	23	31.1652	19.573
WiMax 7 MHz	3.5	1024	24	42.667	14.297
WiMax 8.75 MHz	4.375	1280	24	53.333	11.438
WiMax 10 MHz	5	716.8	12	59.733	10.212
LTE 5 MHz	2.5	983.04	31	31.711	19.236
LTE 10MHz	5	983.04	16	61.44	9.928
LTE 15 MHz	7.5	737.28	8	92.16	6.619
LTE 20 MHz	10	983.04	8	122.88	4.964

Table 9: Typical RX Baseband Filter Calibration Times.

BASEBAND TX ANALOG FILTER CALIBRATION

The baseband TX analog filter calibration tunes the cutoff frequency of the 3rd order Butterworth TX anti-imaging filter. The TX filter is located just after the DAC in the TX signal path. The calibration time depends on the TX filter tune clock frequency. The TX filter tune clock frequency is calculated based on the desired baseband bandwidth. NOTE: The BBBW is half the complex bandwidth and coerced between 20 MHz to 0.625 MHz for the equations used in this filter tuning.

$$Desired\_f_{TXTuneCLK} = \frac{1.6 * BBBW * 2\pi}{\ln(2)}$$

To generate this TXTuneCLK, the BBPLL is divided down using a divide by 1 to 511 divider dedicated to the TX tuner block. Before starting the TX baseband filter tune, set this divider value using the following equation into registers 0x0D6 and 0x0D7.

$$TXBBFDivide[8:0] = \min(511, \mathrm{ceiling}(\frac{BBPLL\_Freq}{Desired\_f_{TXTuneCLK}}))$$

Typical Sequence:

1) Follow the order of events in the scripts generated by the AD9361 Evaluation Software.
2) AD9361 should be in the ALERT state and previously discussed calibrations already run. Verify that no other calibrations are currently running.
3) Set TX baseband filter divide value in register 0x0D6[7:0] and 0x0D7[0]. Be sure to retain the correct settings in 0x0D7[7:1].
4) Enable the TX baseband filter tune circuit by setting 0x0CA=0x22.
5) Start the TX Baseband Filter calibration in register 0x016[6].
6) Calibration is complete when register 0x016[6] self clears.
7) Disable the TX baseband filter tune circuit by writing 0x0CA=0x26.

The TX baseband analog filter calibration does not run automatically, and must be manually triggered by writing SPI register 0x016[6] in the ALERT state. Calibration completion can be monitored on a control out pin or by reading register 0x016[6] until the TX baseband filter calibration bit self clears. The calibration is a binary search algorithm with a maximum calibration time of 355 TX Tune clock cycles.

Standard	BBBW	BBPLL Frequency (MHz)	TXBBF Divider[8:0]	Actual TX Tune Clock Frequency (MHz)	Max Calibration time (us)
WiMax 3.5 MHz	1.75	1024	41	24.9756	14.2139
WiBRO 4.375 MHz	2.1875	1280	41	31.2195	11.3711
WiMax 5 MHz	2.5	716.8	20	35.84	9.9051
WiMax 7 MHz	3.5	1024	21	48.7619	7.2803
WiMax 8.75 MHz	4.375	1280	21	60.9524	5.8242
WiMax 10 MHz	5	716.8	10	71.68	4.9526
LTE 5 MHz	2.5	983.04	28	35.1086	10.1115
LTE 10MHz	5	983.04	14	70.2171	5.0558
LTE 15 MHz	7.5	737.28	7	105.326	3.3705
LTE 20 MHz	10	983.04	7	140.434	2.5278

Table 10: Typical TX Baseband Filter Calibration Times.

BASEBAND TX SECONDARY FILTER

The baseband TX secondary filter is a tunable single pole filter after the baseband TX analog filter. The TX secondary filter corner is ideally set to 5 times the baseband bandwidth to help filter out of band TX noise emissions. No calibration is required for this filter other than using equations to properly set the corner frequency. The registers setting the corner frequency for this filter only need to be set once. They only need to be updated if the baseband bandwidth changes. NOTE: The baseband bandwidth (BBBW) is half the complex bandwidth and coerced between 20 MHz to 0.53 MHz for the equations used in this filter tuning.

Equation theory

Loop through the filter ’s possible resistor values and calculate a valid capacitor value. There are four possible resistor values 200 ohms, 400 ohms, and 800 ohms. After the resistor and capacitor values are determined, set the other filter settings based on the RC determined.

Example TX Secondary Filter Calculation Code

```
res = 100;  //start with a resistor of 100ohms
corner = 5*BBBW_MHz*2π;
for(i=0;i<=3;i++){
    cap = (round((1/((corner*res)*1e6)) * 1e12) ) -12 ;  //where 12pf is parasitic cap when cap register=0
    if(cap<=63) {break;} //if capacitor value is valid exit the loop and use the current res and cap setting.
    res=res*2;
}
if (cap > 63) {cap = 63;}

if((BBBW_MHz*2) <= 9) {reg0D0 = 0x59;}
else if ((BBBW_MHz*2) > 9 & (BBBW_MHz*2) <=24){reg0D0 = 0x56;}
else if ((BBBW_MHz*2) > 24) {reg0D0=0x57;}

if(res==100) {reg0D1=0x0C;}
else if (res == 200) {reg0D1=0x04}
else if (res == 400) {reg0D1=0x03}
else if (res == 800) {reg0D1=0x01}

reg0D2 = cap; //6-bit cap word.

SPIWrite(0x0D0, reg0D0);
SPIWrite(0x0D1, reg0D1);
SPIWrite(0x0D2, reg0D2);
```

Typically, register 0x0D3 is set to its default value of 0x60 for best performance.

RX TIA CALIBRATION EQUATIONS

The RX trans-impedance amplifier (TIA) is located between the mixer and RX baseband analog filter. The TIA has two gain settings (0dB gain and -6dB gain), and applies a single pole filter with a corner at 2.5 times the baseband bandwidth. There is no actual calibration in the AD9361 device. Equations calculate the capacitance to set the corner frequency of the TIA for the 0dB gain case. When the gain index in the AD9361 RX gain table changes the RX TIA gain setting, the AD9361 device will automatically scale the capacitance to maintain the same corner frequency. The RX TIA registers should be updated anytime the baseband BW changes. The necessary registers to write are shown below.

Register Address	Name	D7	D6	D5	D4	D3	D2	D1	D0	Default	R/W
1DB	RX TIA Config	TIA Sel CC<2:0>			Open	TIA2 Override C	TIA2 Override R	TIA1 Override C	TIA1 Override R	60h	R/W
1DC	TIA1 C LSB	TIA1 RF CNT<1:0>		TIA1 C LSB<5:0>						03h	R/W
1DD	TIA1 C MSB	Open	TIA1 C MSB<6:0>							0Bh	R/W
1DE	TIA2 C LSB	TIA2 RF CNT<1:0>		TIA2 C LSB<5:0>						03h	R/W
1DF	TIA2 C MSB	Open	TIA2 C MSB<6:0>							0Bh	R/W

Table 11: RX TIA Setup Registers

NOTE: for normal operation, the override bits should be set to zeros. The TIA RF CNT[1:0] bits are controlled by the RX gain table to scale the gain of the TIA. Set the TIA Sel CC[2:0] using the lookup table below. Set the four registers containing the TIA capacitance as shown in the following equations. The RX TIA depends on the RX Baseband filter calibration results. Make sure the RX Baseband filter calibration successfully completed before programming the TIA registers.

Baseband (Real) Bandwidth (BBBW)	TIA Sel CC<2:0>
BBBW≤3MHz	7
3MHz < BBBW ≤10MHz	3
BBBW > 10MHz	1

Table 12: RX TIA Sel CC Register Setting

Using the results from the RX baseband filter calibration in registers 0x1EB, 0x1EC and 0x1E6, calculate the desired CTIA.

$$C_{rxbbf} = (reg1EB * 160\ fF) + (reg1EC * 10\ fF) + 140\ fF\ ,$$

$$R_{2346} = (reg1E6[2:0]) * 18300ohm),$$

$$C_{TIA\_fF} = \frac{(C_{rxbbf} * R_{2346} * 0.56)}{3500ohms}$$

CTIA	TIA C MSB<6:0>	TIA C LSB<5:0>
>2920 fF	$\frac{C_{TIA\_fF} - 400\ fF}{320\ fF}$	0
<2920 fF	0	$\frac{C_{TIA\_fF} - 400\ fF}{40\ fF}$

Table 13: RX TIA C MSB/LSB Register Setting

Example RX TIA Calculation Code

```
reg1EB = SPIRead(0x1EB);
reg1EC = SPIRead(0x1EC);
reg1E6 = SPIRead(0x1E6);

if (BBBW_MHz > 20) { BBBW_MHz = 20}; //coerce in this case
else if (BBBW_MHz < 0.2) {   BBBW_MHz=0.2};

Cbbf = (reg1EB * 160) + (reg1EC * 10) + 140; //fF
R2346 = 18300 * (reg1E6 & 7); //valid values of reg1E6 are 1, 2, or 4 (18300,36600,73200ohms)
```

```
 CTIA_fF = (Cbbf * R2346 * 0.56) / 3500;

 if (Ceil(BBBW_MHz) <= 3) { reg1DB = 0xE0;}
 else if (Ceil(BBBW_MHz) > 3 And Ceil(BBBW_MHz) <= 10) {reg1DB = 0x60;}
 else if (Ceil(BBBW_MHz) > 10){  reg1DB = 0x20;}

 if (CTIA_fF > 2920) {
             reg1DC = 0x40;
             reg1DE = 0x40;
             temp = min(127,Round((CTIA_fF - 400) / 320));
            reg1DD = temp;
            reg1DF = temp;
}
else {
             temp = Round((CTIA_fF - 400) / 40) + 0x40;
            reg1DC = temp;
            reg1DE = temp;
            reg1DD = 0;
             reg1DF = 0;
 }

 SPIWrite(0x1DB, reg1DB);
 SPIWrite(0x1DC, reg1DC);
 SPIWrite(0x1DD, reg1DD);
 SPIWrite(0x1DE, reg1DE);
 SPIWrite(0x1DF, reg1DF);
```

RX ADC SETUP

The receive ADCs are 3rd order continuous time delta sigma modulators and are highly programmable. The values in many of the ADC registers change with sampling clock frequency while others do not change but the correct initial settings are critical for proper operation. Due to process variation, it is important to use the equations below to scale the ADC registers based on the results of the RX BBF tune calibration. The ADC registers should be updated anytime the ADC Clock or Baseband BW change. In the equations below, the following functions are used:

Function	Description
floor(x)	Rounds a fraction down to the next lower integer.
int(x)	Rounds a fraction to the nearest integer.
min(x,y)	Selects the minimum of the 2 parameters
max(x,y)	Selects the maximum of the 2 parameters
**	Exponential (10**x = $10^{????}$)

Table 14: Functions used in the RX ADC Equations

Inputs	Description
FsADC	The ADC clock rate
BBBW_MHz	The base band BW, half the RF bandwidth. Valid range 0.2MHz to 28MHz. For more accurate equation calculation, derive this value from the nominal RXBBF tune corner using this equation. ????????????? ?????????????????????? ? × ln ?(2) ? (0.2???) ??????????????????????????????????? ? 1.4 × 2??? ? ?
rxbbf_c3_msb	The RX Baseband Filter calibration MSB result for the C3 capacitor. This is the value read from SPI Register 0x1EB[5:0].
rxbbf_c3_lsb	The RX Baseband Filter calibration LSB result for the C3 capacitor. This is the value read from SPI Register 0x1EC[6:0].
rxbbf_r2346	The RX Baseband Filter calibration result for the resistor. This is the value read from SPI Register 0x1E6[2:0].

Table 15: Variable Names used in the RX ADC Equations

Equations to calculate ADC settings in SPI registers 0x200 – 0x227

```
rxbbf_c3_msb = SPIRead(0x1EB);
rxbbf_c3_lsb = SPIRead(0x1EC);
rxbbf_r2346 = SPIRead(0x1E6);

if (BBBW_MHz > 28) { BBBW_MHz = 28}; //coerce in this case
else if (BBBW_MHz < 0.2) {   BBBW_MHz=0.2};

scale_snr_dB = (FsADC < 80)?(0):(2);
if (BBBW_MHz < 18)
        rc_timeConst = 1/((1.4*2*pi)*(18300*rxbbf_r2346)*(160e-15*rxbbf_c3_msb + 10e-15*rxbbf_c3_lsb + 140e-15)*(BBBW_MHz * 1e6));
else
        rc_timeConst = 1/((1.4*2*pi)*(18300*rxbbf_r2346)*(160e-15*rxbbf_c3_msb + 10e-15*rxbbf_c3_lsb + 140e-15)*(BBBW_MHz *
                1e6)*(1+.01*(BBBW_MHz-18)));
scale_res = sqrt(1/rc_timeConst);
scale_cap = sqrt(1/rc_timeConst);
scale_snr = 10**(scale_snr_dB/10); //** is exponentiation, where the base is 10 and the exponent is (scale_snr_dB/10)
maxsnr = 640/160;

data[0] = 0;  //address x200
data[1] = 0;  //address x201
data[2] = 0;  //address x202
data[3] = 0x24;  //address x203
```

```
data[4] = 0x24;  //address x204
data[5] = 0; //addr x205
data[6] = 0; //address x206
data[7] = min(124,floor(-0.5+80*scale_snr*scale_res*min(1,sqrt(maxsnr*FsADC/640)))); //addr x207
data[8] = min(255,floor(0.5+20*(640/FsADC)*(data[7]/80)/(scale_res*scale_cap)));  //addr x208
data[10] = min(127,floor(-0.5+77*scale_res*min(1,sqrt(maxsnr*FsADC/640))));  //addr x20A
data[9] = min(127,floor(0.8*data[10]));  //addr x209
data[11] = min(255,floor(0.5+20*(640/FsADC)* (data[10]/77)/(scale_res*scale_cap))); //addr x20B
data[12] = min(127,floor(-0.5+80*scale_res*min(1,sqrt(maxsnr*FsADC/640)))); //addr x20C
data[13] = min(255,floor(-1.5+20*(640/FsADC)*(data[12]/80)/(scale_res*scale_cap))); //addr x20D
data[14] = 21*floor(0.1*640/FsADC); //addrx20E
data[15] = min(127,int(1.025*data[7])); //addr x20F
data[16] = min(127,floor(data[15]*(0.98+0.02*max(1,(640/FsADC)/maxsnr)))); //addr x210
data[17] = data[15];  //addr x211
data[18] = min(127,int(0.975*data[10])); //addr x212
data[19] = min(127,floor(data[18]*(0.98+0.02*max(1,(640/FsADC)/maxsnr)))); //addr x213
data[20] = data[18];  //addr x214
data[21] = min(127,int(0.975*data[12])); //addr x215
data[22] = min(127,floor(data[21]*(0.98+0.02*max(1,(640/FsADC)/maxsnr))));  //addr x216
data[23] = data[21]; //addr x217
data[24] = 0x2E; //addr x218
data[25] = floor(128+min(63,63*(FsADC/640))); //addr x219
data[26] = floor(0+min(63,63*(FsADC/640)*(0.92+0.08*( 640/FsADC)))) ; //addr x21A
data[27] = floor(0+min(63,32*sqrt(FsADC/640))); //addr x21B
data[28] = floor(128+min(63,63*(FsADC/640))); //addr x21C
data[29] = floor(0+min(63,63*(FsADC/640)*(0.92+0.08*(640/FsADC)))); //addr x21D
data[30] = floor(0+min(63,32*sqrt(FsADC/640))); //addr x21E
data[31] = floor(128+min(63,63*(FsADC/640))); //addr x21F
data[32] = floor(0+min(63,63*(FsADC/640)*(0.92+0.08*(640/FsADC)))) ; //addr x220
data[33] = floor(0+min(63,63*sqrt(FsADC/640))); //addr x221
data[34] = min(127,floor(64*sqrt(FsADC/640))); //addr x222
data[35] = 0x40; //addr x223
data[36] = 0x40; //addr x224
data[37] = 0x2C; //addr x225
data[38] = 0x00; //addr x226
data[39] = 0x00; //addr x227
```

After calculating the values for the data array, write the array into the AD9361 SPI port starting at SPI register 0x200 and ending at 0x227.

BASEBAND DC OFFSET CALIBRATION

It is recommended to run the Baseband DC offset calibration once during device initialization in the ALERT state. Since the baseband signal path does not change with different wireless standards or clock frequencies, it should never need to be run again. The baseband DC offset correction values are stored for all of the RX analog baseband filter gain steps. The correction words are applied as the RX gain changes based on the current RX gain table index. The AD9361 Evaluation Software generates scripts with suggested register settings for the initial one shot baseband DC offset calibration. A baseband DC offset calibration must be started manually by writing register 0x016[0]. The calibration is complete when the calibration start bit self clears.

The one-shot baseband DC offset calibration will complete in a finite time depending on the settings in the SPI registers. The following equation will calculate the duration of the baseband DC offset calibration. If both RX chains are enabled, the two receive chains calibrate in parallel.

DCOffsetcount = 0x193[7:0]

CountSize = (190[7]==1)?(1024):(256);

BB DC Offset Cal Time$_{\text{(CLKRF Clock Cycles)}}$ = ((DCOffsetcount * CountSize) + 0x185) * (25 RXBBF Gain Indexes)

Baseband DC offset Tracking

Baseband DC offset tracking should be used in conjunction with the RF DC offset tracking option. The AD9361 Evaluation Software enables Baseband DC offset tracking and RF DC offset tracking by default. A high pass filter loop is utilized to track DC changes caused by the RF DC offset and RX quadrature correction block. The speed and accuracy of the BB DC tracking loop can be set in SPI registers

0x190 through 0x192. Also, an option exists for the BB DC offset tracking to attack/settle quickly after a gain change and then after a certain time switch to slower DC offset tracking updates. When using the tracking mode, first run an initial one-shot baseband DC offset calibration to minimize any static DC offsets in the signal chain. Then enable tracking along with the desired tracking loop settings. Baseband DC offset tracking is enabled in register 0x18B[5]. By default, baseband DC offset tracking is disabled.

RF DC OFFSET CALIBRATION

The RF DC Offset calibration should be run once during initialization, or when moving to a new carrier frequency that is more than 100 MHz away from the previous carrier frequency. This calibration should be run in the ALERT state while the RX synthesizer is enabled. Write register 0x016[1] to start the calibration. The calibration has finished when 0x016[1] self clears. The internal calibration results LUT stores separate results for the RF RX A input. If using the RF RX B or C inputs along with the RF RX A input, you should run the calibration twice, once with each input band selected. After calibrating each band, switching between from the A input to the B or C input should not require another calibration unless a large frequency change is made. Since the B and C inputs use the same calibration results, switching from input B to input C may require running the RF DC offset calibration.

When using the full RX gain table, the RF DC offset calibration only calibrates at gain indexes that are designated to calibrate in the RX gain table. This is because several consecutive gain steps may leave the front end gain at the same setting, while only changing baseband gain settings. The RF DC offset correction is only designed to remove DC offset due to the RF parts of the signal chain. By only running the calibration at gain indexes that actually change the front end gain, the calibration time is reduced. If the LUT does not hold a DC correction value for the current RX gain index, it will use the DC offset correction for the next higher gain index that was calibrated. In the case of a split RX gain table, the calibration runs at each front end gain LUT index.

The RF DC offset tracking is enabled by default. The tracking triggers an RF DC offset update based on three events: RX gain change, no energy detected, or when the ENSM exits the RX state. A register setting allows enabling or disabling any combination of these events. Disabling RF DC offset tracking would use the initially calibrated RF DC offset and never update the correction words.

Depending on the settings in registers 0x185 through 0x18D, the calibration time and accuracy will vary according to the equations below. If both receive chains are enabled, both RX chains are calibrated in parallel.

WaitTime = 0x185[7:0]
DCcount = 0x186[7:0]
DCcalTime = 0x187[3:0]
Length Of Each Average = [{(DCcount*256)+255} + WaitTime]

numberOfAverages = ((DCcalTime + 1) * 4) + 1 //number of times to repeat the measurement
Calibration Time for One RX Gain Index = Length Of Each Average * numberOfAverages
Total Calibration time $_{\text{(CLKRF Clock Cycles)}}$ = Calibration Time for One RX Gain Index * (Number of Enabled Gain indexes + 1)

The calibration begins at min gain (index = 0) and moves toward max gain. The RF DC offset correction for each enabled RX gain index is stored in a LUT and applied when the RX gain index is used. If the full RX gain table is used, the RF DC offset is calibrated only at gain indexes specified to calibrate in the RX gain table. If the RX gain table is split, the Number of Enabled Gain indexes in the equation above equals 41 gain steps.

RX QUADRATURE CALIBRATION

The RX quadrature calibration minimizes the phase and gain error in the receive path. Three options exist to calibrate the RX quadrature:

1) Single shot RX quadrature calibration
2) RX quadrature continuous tracking
3) Single shot calibration + tracking

The AD9361 Evaluation Software uses option 2 (RX quadrature continuous tracking). Option 2 may require that the device be placed in the RX state for some amount of time to track out the initial quadrature error before the error is minimized. If the quadrature error minimization is required for the first RX data, then option 3 may be a better choice.

Single Shot RX Quadrature Calibration

This calibration is manually triggered by writing register 0x016[5] in the ALERT state. The single shot calibration has completed when 0x016[5] self clears. Run this calibration during initialization and any time the carrier frequency changes more than 100 MHz. In one-shot mode, an internal calibration signal is enabled to minimize the quadrature error. The max calibration time is $\text{RX}_{\text{QuadcalTime}}$ CLKRF clock cycles. ClockRF is the clock rate at the input of the RX FIR filter (before RX decimation). If both receivers are enabled, RX1 will calibrate, followed by RX2.

$$RX_{QuadcalTime} = \#\,RxChannels \times \left(\left[Decimation \times \left(Settle + cal\right)\right] + SettleDC\right)$$, where cal=0x16C, Settle = 0x16D

$$Decimation = \begin{cases} 2048 & \text{if } 0x169[2] = 0 \\ 2^{(4+m)},\quad 0 \le m \le 5 & 0x169[2] = 1 \end{cases}$$

$$SettleDC = \begin{cases} (0x186 \times 64 + 63) \times \left(\left[(0x187[3:0] + 1) \times 4\right] + 1\right) & 0x169[5:4] = 2'bx0 \\ 0x16D + 1 & 0x169[5:4] = 2'b11 \\ 0 & 0x169[5:4] = 2'b01 \end{cases}$$

The RX quadrature calibration stores a separate set of calibration results for the RF RX_A input path. The RX_B and RX_C input paths share the same calibration results. If using both the RX_A input and the RX_B/RX_C input, run the RX quadrature calibration twice, once with each input path selected. When using the RX_B and RX_C inputs, if the desired LO frequency between the two inputs is greater than 100MHz, a refresh calibration may be needed when switching between the inputs.

The single shot quadrature calibration is only accurate for LO frequencies below 3GHz. When using frequencies 3GHz and above, it is recommended not to run the single shot calibration. Instead, use the RX quadrature tracking option.

To run the single shot RX quadrature calibration, perform the following steps:

1) Refer to the AD9361 Evaluation Software generated script for setup values for registers 0x186, 0x187, 0x168
2) Move the AD9361 into the ALERT state
3) Ensure both RF synthesizers are enabled (FDD mode). 0x013=0x01(ENSM FDD) and 0x015[2] (Dual synth mode) is set.
4) Set 0x169=0xC0 //Verify free run mode is disabled in 0x169[3].
5) Set 0x057=0x33 //Power down TX mixer. This improves the calibration result.
6) Set TX LO frequency to the RX LO frequency + BBBW/2. This places the TX LO in the passband of the RX spectrum. If the TX LO integer frequency word was written, allow time for the TX VCO calibration to complete.
7) Set 0x016=0x20 //Start the RX Quadrature Calibration
8) Wait for calibration to complete (when 0x016 = 0x00).
9) Set TX LO back to its original frequency.
10) Set 0x057=0x30 //Re-enable TX mixer for normal operation.
11) Return to TDD operation if desired by setting register 0x013=0x00 and clear 0x015[2] (Dual synth mode).

The AD9361 Evaluation Software does not use the single shot RX quadrature calibration. It only uses the tracking mode. Only using the tracking mode may require a short time the first time in the RX state to allow the quadrature error to track and improve.

RX Quadrature Tracking Calibration

In the continuous running mode, the calibration will use RX data to minimize the quadrature error continuously. During initialization in the ALERT state, follow the script generated by the AD9361 Evaluation Software to enable RX Quadrature tracking. The same register settings for 0x168 – 0x16F are used from the single shot calibration method. After setting the registers properly, the tracking mode is enabled by setting 0x169 to 0xCF. As soon as the AD9361 ENSM enters the RX or FDD state, the tracking will be to minimize the quadrature error.

TX QUADRATURE CALIBRATION

The TX quadrature calibration uses a calibration signal internally to minimize the TX DC offset, gain, and phase errors to improve the performance of the transmit chain. This calibration is triggered by writing register 0x016[4] during initialization in the ALERT state. Completion of the calibration can be monitored by reading SPI register 0x016[4] until the bit self clears. When changing the carrier frequency, a much faster refresh calibration can be initiated if desired to update the TX offset, gain and phase error corrections. It is also recommended to refresh the calibration results if the device temperature changes dramatically. The AuxADC can be used to measure the device temperature using the internal temperature sensor, and to know when to refresh the TX Quadrature calibration. To improve the calibration time, set registers 0x0A4 and 0x0A9 to smaller values. For the refresh calibration, setting the TX Quadcal calibration settings(m, settle, or cal_count) too small can result in bad calibration results.

The TX quadrature calibration is a convergence algorithm, but has a maximum calibration time described below. The equation calculates the number of CLKRF clock cycles used for maximum calibration time. CLKRF is the clock rate at the output of the TX FIR filter (after TX FIR interpolation). If both transmit chains are enabled, TX1 will calibration first, followed by TX2.

$$TX_{QuadcalTime(CLKRF\ clockcycles)} = \#TxChannels \times \left(2^{(4+m)} \times \left(2 \times Settle + cal\_count\right)\right),$$

m=0x0A1[1:0], Settle=0x0A4[7:0]+1, cal_count=0x0A9[7:0]+1

The TX quadrature calibration stores a separate set of calibration results for the RF TX_A output path. If using both, the TX_A output and the TX_B output, run the calibration twice, once with each output path selected.

See the scripts generated by the AD9361 Evaluation Software for suggested register settings for registers 0x0A0 through 0x0AE. The TX quadrature calibration settings change based on the baseband bandwidth and which digital filters are enabled in the RX data path. If using a custom AD9361 setup, ensure the TX Quadcal NCO frequency is within the RX baseband filter bandwidth. As the enabled digital filters change, the RX NCO Phase offset in register 0xA0[4:0] may need to be adjusted to find the optimal setting.

The TX quadrature calibration settings were modified in version 2.1.0 of the AD9361 Evaluation Software. With these changes, if using a custom RX Gain table, verify that the TX quadrature calibration gain index in register 0x0AA points to an index with the TIA index=1 and LPF index=0. When using a split gain table, 0x0AA should point to a gain index with TIA Index=1. For a split gain table, set the LPF index in register 0x0AE to a value of 0x00.

FACTORY CALIBRATIONS

Factory calibrations are necessary to limit the amount of variation seen across a large quantity of circuit boards. Some calibrations are used to increase the accuracy of the AD9361 device, while others are needed to calibrate non-linearities of external components in the RF front-end. The factory calibrations described below are suggested calibrations. The number of calibration points for a specific design is dependent on the temperature stability and linearity of the RF components. The actual factory calibrations necessary might vary from the list below depending on the accuracy and performance of the desired system.

#	Factory Calibration
1	Internal DCXO (AFC tune range)
2	TX RSSI (TX Monitor)
3	RX RSSI
4	RX GM / LNA Gain Step Error
5	TX Power out Vs TX attenuation
6	TX Power out Vs Frequency

Table 16: Suggested Factory Calibrations

INTERNAL DCXO

(If using an external clock into the REF_CLK_IN pin, ignore this calibration)

The AD9361 device can use a crystal oscillator (XO) to clock the chip. In this mode, the AD9361 has an internal DCXO (digitally control oscillator) that can pull the XO frequency. This is necessary to cancel out carrier frequency offset and Doppler shift due to mobility. Carrier frequency offset can also occur due to the XO changing temperature.

The DCXO tune registers consist of a coarse tune value and a fine tune value. During factory calibration, the coarse tune setting should be found such that the fine tune range is centered evenly around the desired XO frequency. This would ensure the widest usable DCXO range.

If necessary by the system, the factory calibration could also sweep the DCXO fine tune codes to produce a LUT of frequency error vs. Fine Tune code to allow quick updates of the XO tuning. The temperature sensor could also be read to aid in correcting for the frequency variation across temperature of the XO.

TX RSSI (TX MONITOR)

If the power detector is used, at minimum, a single point TX RSSI measurement must be made to correlate the absolute output power after the PA to the TX RSSI value reported by the AD9361 device. Typically, a coupler is used to sample the power after the PA back into the TX Monitor input of the AD9361 device. During this measurement, transmit a typical burst. A signal with constant power (such as a preamble) would give the most accurate results. Once a single point measurement has been measured, the TX RSSI for other TX power out levels can be calculated with the assumption that the assumption that the TX path is linear, the power detector path is linear and all external RF components in the path are operating in their linear regions. If the system requires a higher precision of power control, multiple points across the AD9361 TX attenuation range and carrier frequency range can be measured during this factory calibration to generate a matrix of TX RSSI correction values.

The factory calibration should be completed using the corresponding transmitter on the AD9361 device to transmit a known reference signal while the TX RSSI is read across the SPI port for each desired output power point. Refer to the AD9361 TX Power Control Guide for detailed information on this process and the different power control modes.

RX RSSI

The receive signal strength indicator (RSSI) measurement occurs after the RX gain is set (manual gain)/ frozen(AGC). For the power measurement to be meaningful to the BBIC, the RSSI code must be related to an absolute RX input level (dBm) during factory calibration. Depending on the RSSI accuracy desired, this calibration can be a signal point calibration at a single frequency and input power level, or can consist of a matrix of frequencies and input power levels. The calibration signal should be a typical burst of the associated standard. For each receiver, there are two different RSSI read back registers. One register stores the first power measurement (RSSI Preamble register). The second register updates the power measurement at the symbol rate (RSSI Symbol register). If a calibration signal has a preamble or symbol with constant power, the RSSI will be more accurate and repeatable. For more detail, please read the description of the RX RSSI registers 0x1A7 through 0x1AC. Set the RSSI measurement configuration in SPI registers 0x150 through 0x15D. For a particular RX gain control setup, the AD9361 Evaluation Software provides a setup wizard to help set these registers correctly. The RSSI measurement of the AD9361 device automatically takes into account the RX gain step used and factors it into the RSSI value. For improved RSSI accuracy over frequency, run the following calibration (RX GM/LNA gain step calibration) before this factory calibration.

RX GM / LNA GAIN STEP CALIBRATION

The gain of the LNA and mixer stages inside the AD9361 varies over temperature and frequency. This calibration is a onetime factory calibration that measures the gain step error for the LNA and mixer to reduce the error of the RX RSSI measurement. In the field, the BBIC would write the error-corrected expected gain steps into the AD9361 for the current frequency used. This process will improve RX RSSI accuracy and linearity. This procedure should be completed before making the absolute RX RSSI correlation mentioned above. See the AD9361 Gain Control and RSSI User Guide for a detailed procedure for this calibration. This is only necessary if using the RX RSSI feature.

TX POWER OUT VS TX ATTENUATION AND TX POWER OUT VS CARRIER FREQUENCY

A factory calibration should be completed such that the output power of the system is known. Depending on the linearity of the RF components chosen, a single point calibration may be sufficient. If higher accuracy is desired, a matrix of carrier frequencies and AD9361 TX attenuation points may be evaluated to measure the TX power output. This factory calibration would prevent the output power from exceeding regulatory limits. This calibration should also check for transmit mask compliance and transmit emission limits (ACLR).