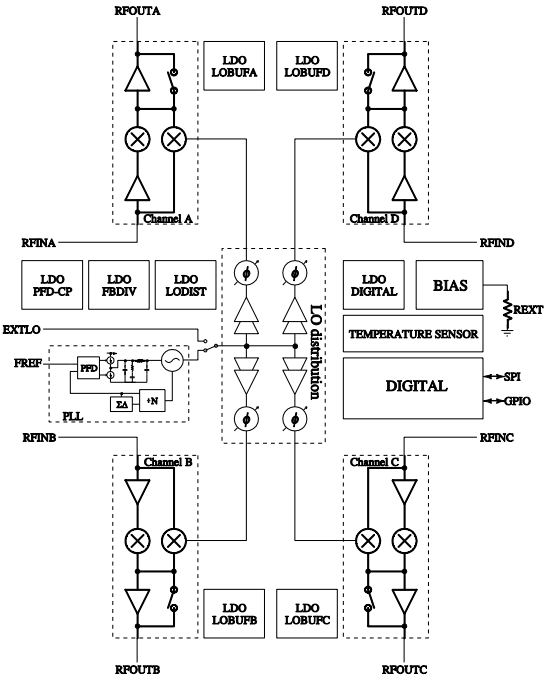
Lime Microsystems Limited

Surrey Technology Centre
Occam Road
The Surrey Research Park
Guildford, Surrey GU2 7YG
United Kingdom

Tel: +44 (0) 1483 685 063
e-mail: enquiries@limemicro.com

LMS8001
Reference Manual



Chip version:	LMS8001A, LMS8001B
Chip revision:	1
Document version:	1.0
Document revision:	0

Contents

1

Overview

LMS8001 (Figure 1.1) contains four RF channels, integrated PLL with programmable LO distribution, and auxiliary circuits, such as biasing block, integrated LDOs and a temperature sensor.

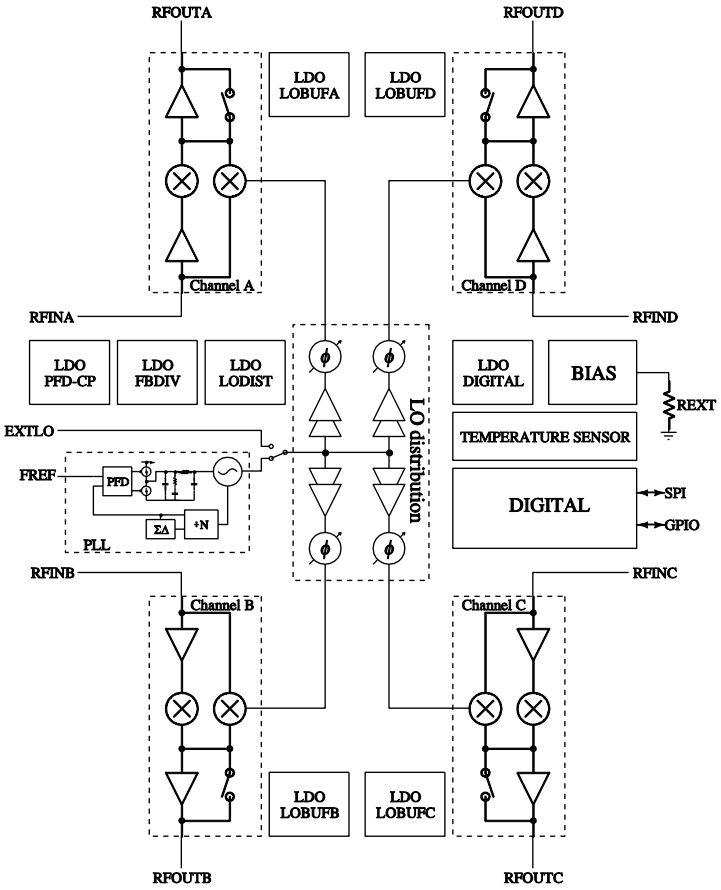


Figure 1.1: LMS8001 Block Diagram

2

Digital Logic Block Diagram

Digital logic implemented in LMS8001 is shown in Figure 2.1. It consists of SPI interface for communication, register banks, programmable GPIO, control logic for RF channels and PLL. There are four sets (profiles) per RF channel and eight sets (profiles) of PLL control signal values. Profile can be selected with GPIO pins, SPI register value, or a combination, depending on how MUXSEL macro is programmed. Additionally, each set of PLL control signals can be programmed with fast lock values, to facilitate faster frequency settling upon profile change, e.g. in a frequency hopping application.

In the following figures the signal name color indicates the following: blue – outside signal, red – signal from the SPI register, and black – internal signal.

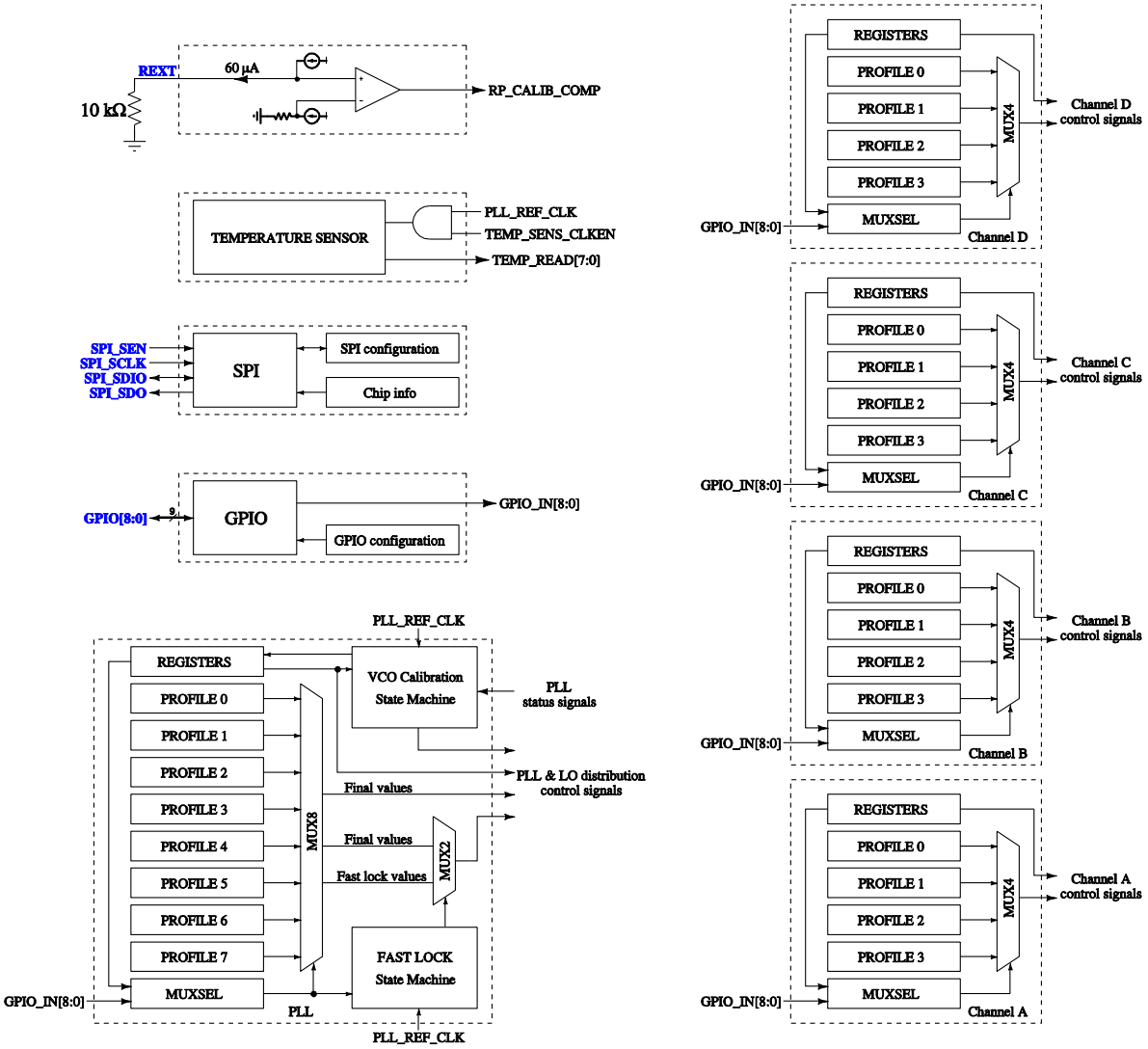


Figure 2.1: LMS8001 Digital Block Diagram

2.1 SPI interface

The functionality of LMS8001 is fully controlled by a set of internal registers which can be accessed through a serial SPI port interface. Both write and read operations are supported. The serial SPI port can be configured to run in 3 or 4 wire mode with the following pins used:

- SEN SPI serial port enable, active low, output from master;
- SCLK SPI serial clock, output from master;
- SDIO SPI serial data in/out (Master Output Slave Input (MOSI) / Master Input Slave Output (MISO)) in 3 wire mode,
  Serial data input (MOSI) in 4 wire mode;
- SDO SPI serial data out (MISO) in 4 wire mode, don't care in 3 wire mode.

SPI serial port key features:

- Operating as slave;

- Operating in SPI Mode 0 (data is captured on the clock's rising edge, while data is shifted on the clock's falling edge);
- 32 serial clock cycles are required to complete write operation;
- 32 serial clock cycles are required to complete read operation;

Multiple write/read operations are possible without toggling serial enable signal. All configuration registers are 16-bit wide. Write/read sequence consists of 16-bit instruction followed by 16-bit data to write or read. MSB of the instruction bit stream is used as SPI command where CMD = 1 for write and CMD = 0 for read. The following 15 bits are register address, followed by 16 data bits.

Write/read cycle waveforms are shown in Figure 2.2, Figure 2.3, and Figure 2.4. Note that write operation is the same for both 3-wire and 4-wire modes. Although not shown in the figures, multiple byte write/read is possible by repeating instruction/data sequence while keeping SEN low.

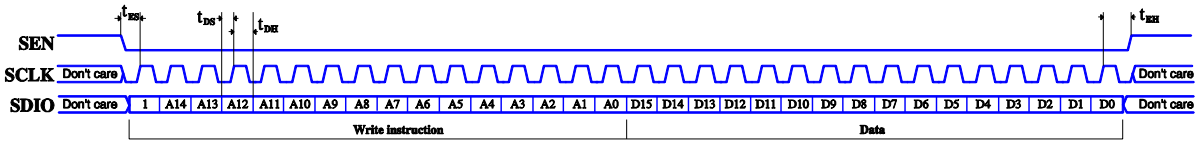


Figure 2.2: SPI write cycle, 3-wire and 4-wire modes

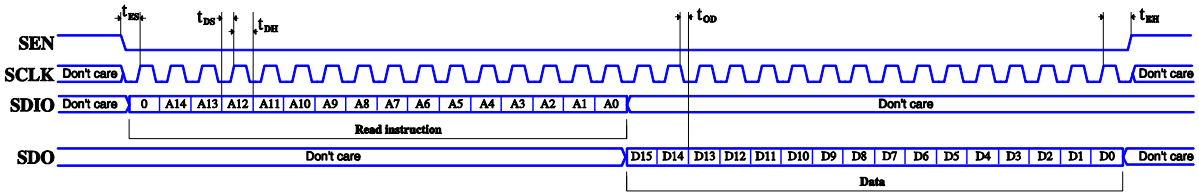


Figure 2.3: SPI read cycle, 4-wire mode (default)

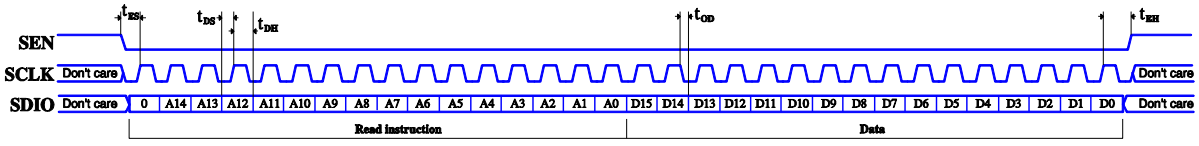


Figure 2.4: SPI read cycle, 3-wire mode

Registers relevant to SPI configuration are listed in the following table.

Register	Address	Reset value
SPIConfig	0x0000	0x001F

2.2 GPIO

LMS8001 has flexible GPIO with nine individually programmable pins, whose structure is shown in Figure 2.5. GPIO pins can be used in several ways, from basic input/output to advanced control of RF channels and PLL. GPIO pin n direction is controlled by GPIO_InO[n] bit of GPIOConfig_IO register. Pull-up resistor of GPIO pin n is controlled by

GPIO_PE[n] bit of GPIOConfig_PE register. Output driver strength is controlled by GPIO_DS[n] bit of GPIOConfig_DS register. GPIO pin n state can be read from GPIO_IN[n] bit of GPIOInData register. When used as output, GPIO pin n can be configured to output the value of GPIO_OUT_SPIO[n], or the value of internal signals PLL_LOCK, VTUNE_LOW, VTUNE_HIGH or FAST_LOCK_ACT. The output selection is configured with GPIOn_SEL[2:0] bits in registers GPIOOUT_SEL0 and GPIOOUT_SEL1.

Input signals GPIO_IN[8:0] represent the voltage level at GPIO pad, and can be used for control of RF channel configurations (profiles). When the GPIO pin n is configured as an input, the value read from GPIO_IN[n] is set by an external source, while it is a loopback signal when configured as an output. Loopback feature can be used to simultaneously trigger external event and change the RF channel and/or PLL profile.

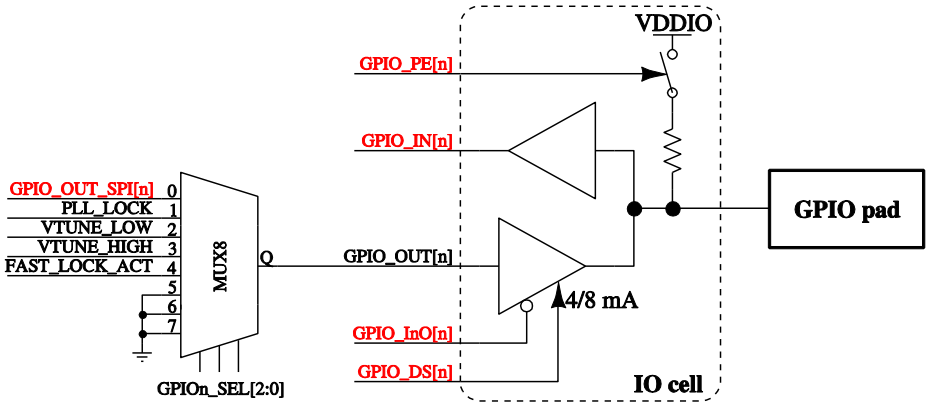


Figure 2.5: GPIO pad structure

Registers relevant to GPIO configuration are listed in the following table.

Register	Address	Reset value
GPIOOutData	0x0004	0x0000
GPIOOUT_SEL0	0x0005	0x0000
GPIOOUT_SEL1	0x0006	0x0000
GPIOInData	0x0008	
GPIOConfig_PE	0x0009	0x03FF
GPIOConfig_DS	0x000A	0x0000
GPIOConfig_IO	0x000B	0x03FF

3

Biasing & LDOs

LMS8001 biasing block, shown in Figure 3.1, generates all reference currents and voltages required for chip operation. External 10 kΩ resistor is used for calibration. Integrated LDOs allow operation from single supply voltage, and are fully programmable. LDOs can be individually controlled, allowing elaborate power management schemes.

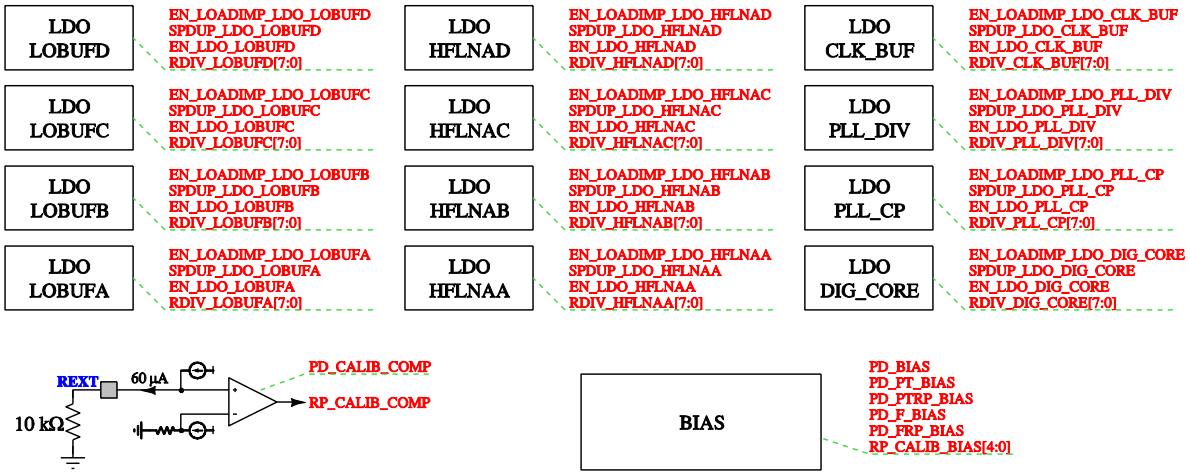


Figure 3.1: LMS8001 Biasing and LDOs

Registers relevant to biasing and LDO configuration are grouped into register bank BiasLDOConfig.

4

Temperature Sensor

Integrated temperature sensor can be used for temperature compensation of RF channels. To use the temperature sensor, bias should be enabled by setting TEMP_SENS_EN=1 and clock by setting TEMP_SENS_CLKEN=1 in register TEMP_SENS. Temperature conversion is started by writing 1 to TEMP_START_CONV bit, which is cleared when the conversion is complete. Temperature conversion result can be readout from TEMP_READ[7:0].

Chip temperature can be calculated as:

$$\text{Temperature [°C]} \approx T0 + T1 \cdot \text{TEMP\_READ} + T2 \cdot \text{TEMP\_READ}^2$$

Where the default temperature coefficient values are:

T0 = -105.45

T1 = 1.2646

T2 = -0.000548

Accuracy of the calculated temperature value is sensitive to mismatch on chip. Without any calibration the standard variation of the measurement accuracy is around 2 °C.

It is recommended that the single point calibration is performed, in which the coefficient T0 is calculated. After such calibration, the measurement accuracy should be within ±1.5 °C.

Register relevant to temperature sensor configuration is TEMP_SENS in ChipConfig register bank.

5

Channel and PLL profiles

Channel and PLL control signals are grouped in signal groups, which are multiplexed simultaneously. Each signal group has four sets of values per channel, and eight sets of values for PLL. Collection of signal groups forms a profile. Group multiplexer control signals are generated by MUXSEL macros, controlled by registers listed in table below.

Profile	Signal Group	Control Registers	Address
CHx	CHx_PD	CHx_PD_SEL0	CHx_BASE+0x10
		CHx_PD_SEL1	CHx_BASE+0x11
	CHx_LNA	CHx_LNA_SEL0	CHx_BASE+0x12
		CHx_LNA_SEL1	CHx_BASE+0x13
	CHx_PD	CHx_PD_SEL0	CHx_BASE+0x14
		CHx_PD_SEL1	CHx_BASE+0x15
	Internal value	CHx_INT_SEL	CHx_BASE+0x16
HLMIXx	HLMIXx_CONF	HLMIXx_CONF_SEL0	HLMIXx_BASE+0x8
		HLMIXx_CONF_SEL1	HLMIXx_BASE+0x9
	HLMIXx_LOSS	HLMIXx_LOSS_SEL0	HLMIXx_BASE+0xA
		HLMIXx_LOSS_SEL1	HLMIXx_BASE+0xB
	Internal value	HLMIXx_INT_SEL	HLMIXx_BASE+0xC
PLL	PLL	PLL_CFG_SEL0	0x4008
		PLL_CFG_SEL1	0x4009
		PLL_CFG_SEL2	0x400A
	Internal value	PLL_CFG_SEL	0x400B

6

MUXSEL Macro

RF channel and PLL profiles are selected by multiplexer control signals. Each bit of multiplexer control signals is generated by MUXSEL macro, shown in Figure 6.1. Control signal can be generated from GPIO inputs (INTERNAL=0) or from internal register (INTERNAL=1). GPIO_IN[8:0] signals are masked (logical AND) by GPIO_MASK[8:0] and the individual terms are ORed to form the control signal value. MUXSEL output is inverted when control signal INVERT=1 or passed through when INVERT=0, which can be used to control mutually exclusive configurations with a single GPIO pin.

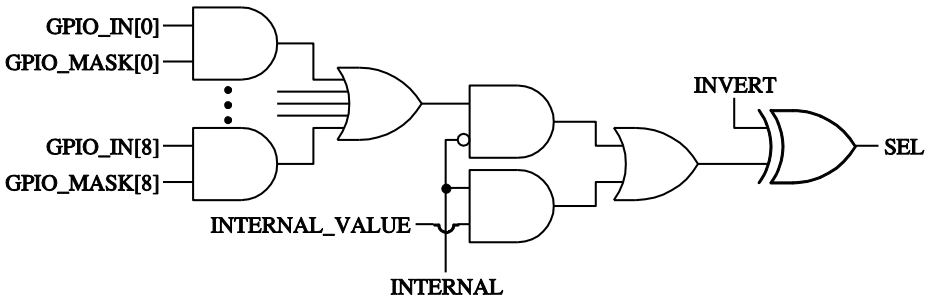


Figure 6.1: MUXSEL macro

MUXSEL control signals GPIO_MASK, INTERNAL and INVERT are packed in a register with structure given in table below. There are two or three registers per signal group, depending on whether there are four or eight profiles. INTERNAL_VALUE bits for all signal groups are in a separate register, allowing the update of control signals with a single SPI write command.

Address		Register Name (Reset Value)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	Group_SELn_INTERNAL	RW	**Group** control signals multiplexer SELn signal is generated 0 – from GPIO & **Group**_SELn_MASK, 1 – from **Group**_INT_SEL<n>

Address		Register Name (Reset Value)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	Group_SELn_INVERT	RW	Invert the SELn signal of **Group** control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	Group_SELn_MASK<8:0>	RW	GPIO mask for SELn signal of **Group** control signals multiplexer.

7

Channel Control Logic

There are two options of LMS8001. The difference is in the architecture of the RF channels. Option LMS8001A contains RF channels with LNA, two mixers and power amplifier (Figure 7.1), whereas LMS8001B channel is comprised of high-linearity mixer (HLMIX) only (Figure 7.3).

7.1 RF Channel Control Logic – LMS8001A

Structure of Channel_**x** (**x**=A,B,C,D) is shown in Figure 7.1. Input RF signal can be amplified by LNA and mixed by MIXA or can be directly fed to mixer MIXB. At mixer output there is a programmable 50 Ω termination resistor, which can be switched on or off. Power amplifier (PA) can be used to further amplify the signal, or can be powered down and bypassed.

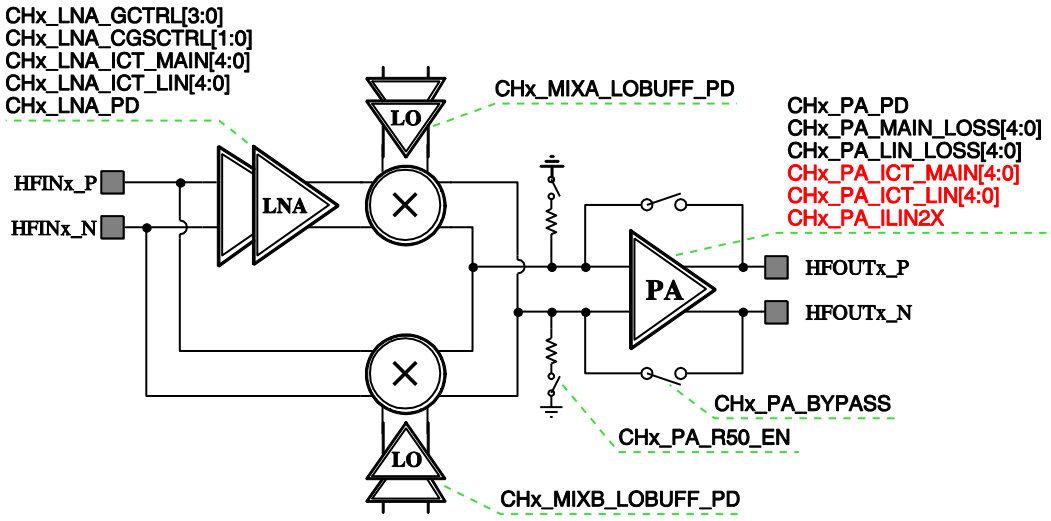


Figure 7.1: RF Channel – LMS8001A

Channel control signals are generated by digital circuit shown in Figure 7.2. Control signals are divided into three groups, each with four sets of values, which results in total of 64

possible configurations. Signal group multiplexer control signals are generated by MUXSEL macro, allowing the control via GPIO pins, SPI registers, or combination of them.

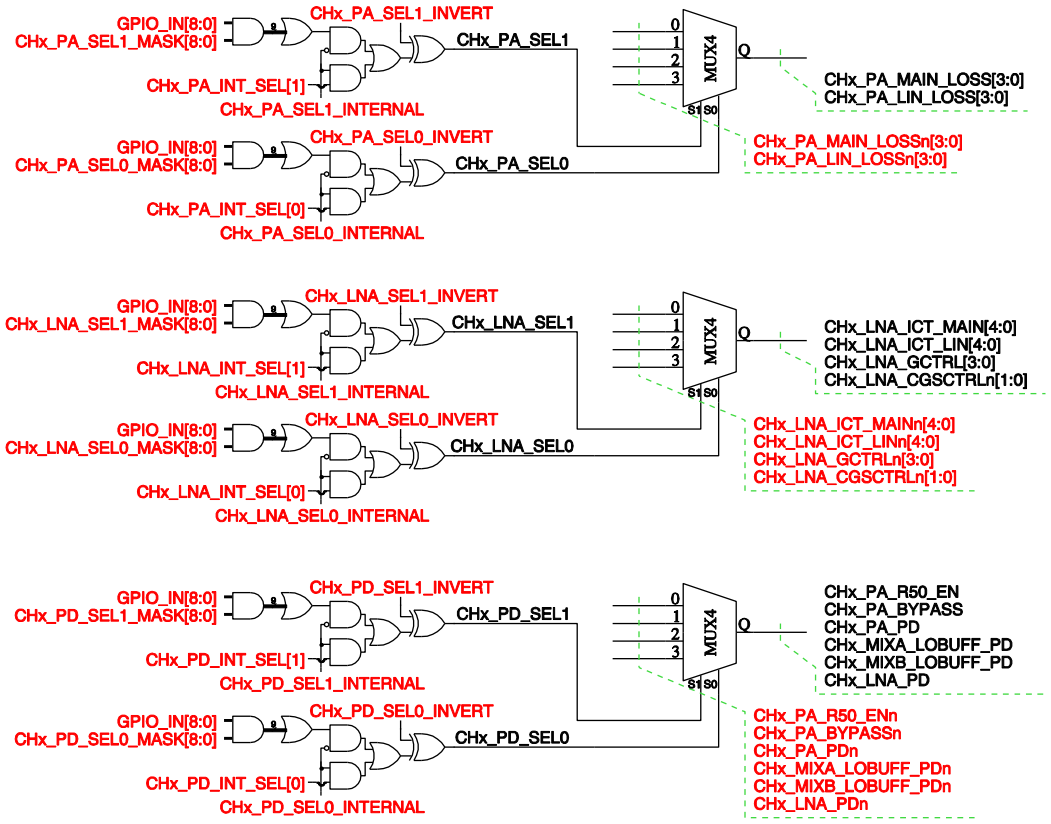


Figure 7.2: RF Channel control signal multiplexing

Registers relevant to RF Channel configuration are grouped into register banks Channel_x (x=A,B,C,D).

7.2 High-Linearity Mixer (HLMIX) Control Logic – LMS8001B

Structure of HLMIX**x** (**x**=A,B,C,D) is shown in Figure 7.3.

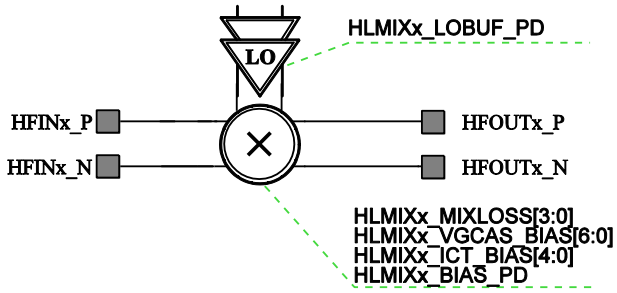


Figure 7.3: High-Linearity Mixer (HLMIX) – LMS8001B

Channel control signals are generated by digital circuit shown in Figure 7.4. Signal group multiplexer control signals are generated by MUXSEL macro, allowing the control via GPIO pins, SPI registers, or combination of them.

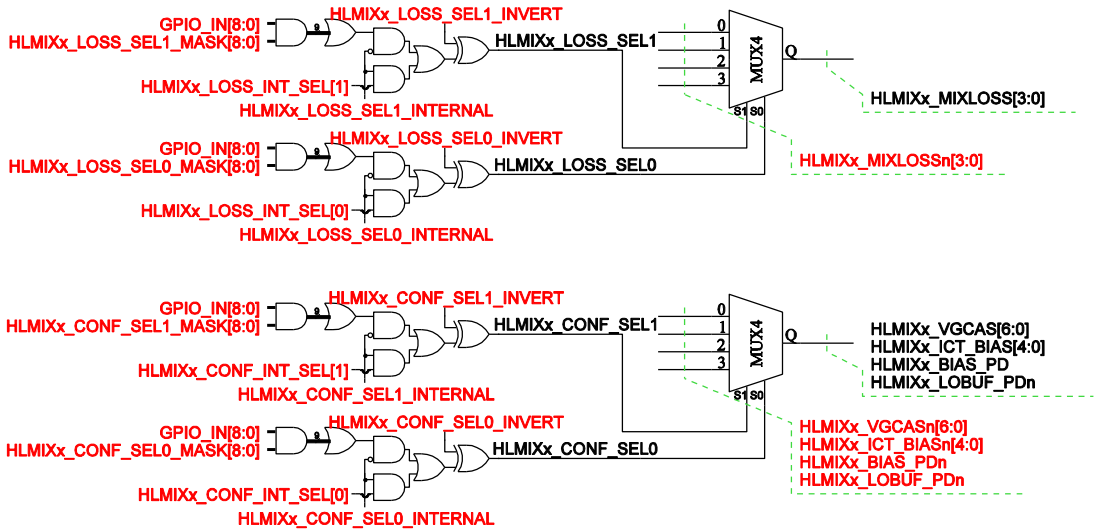


Figure 7.4: HLMIX control signal multiplexing

Registers relevant to HLMIX configuration are grouped into register banks HLMIXx (x=A,B,C,D).

8

HFPLL

8.1 Overview

The LMS8001 up/down frequency shifter IC contains one HFPLL frequency synthesizer for the generation of the LO signal required in four integrated RF paths. The HFPLL frequency synthesizer uses fractional-N PLL architecture with completely integrated VCOs and loop filter. It is fully self-contained synthesizer and requires no external parts to cover the full specified frequency range of the device using convenient reference frequency values between 10 and 50 MHz. Besides, LMS8001 IC can use external LO signal from the dedicated input pins.

The fundamental frequency of integrated HFPLL VCO cores ranges from 4.8 to 9.6 GHz. Local oscillator frequencies can take values from 300 MHz to 9.6 GHz.

8.2 Architecture

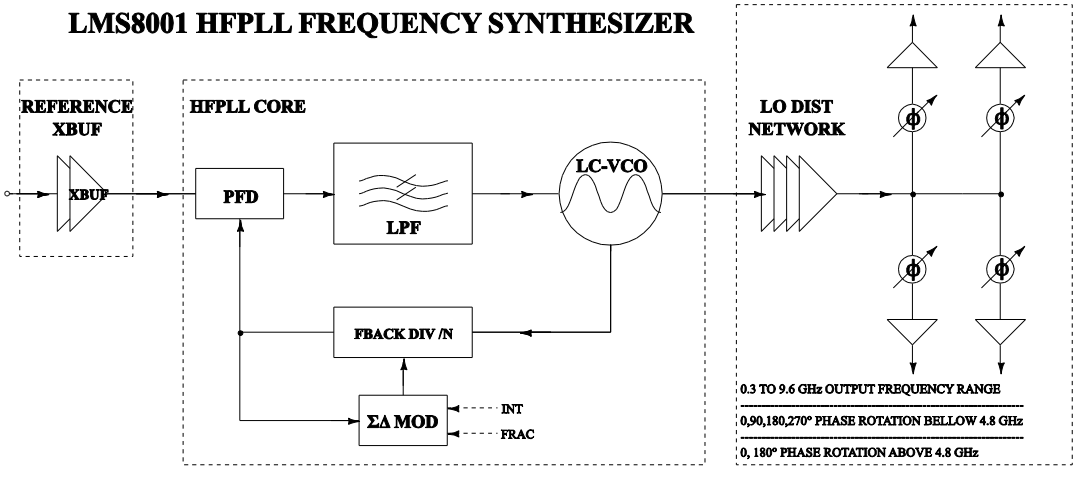


Figure 8.1: LMS8001 Frequency Synthesizer

Block diagram of the complete frequency synthesizer of the LMS8001 IC is shown in the Figure 8.1. Three main sub-parts are indicated: reference clock buffer, HFPLL core and LO distribution network. Details about mentioned sub-parts and its containing circuits will be given in the following sections.

8.3 Reference

The external clock source (as TCXO for example) should be provided to the reference input of the LMS8001 device. The LMS8001 can accept clipped sine-wave as well as CMOS levels for the HFPLL reference clock. Diagram of the reference input circuit (XBUF) with dedicated control signals is presented in Figure 8.2.

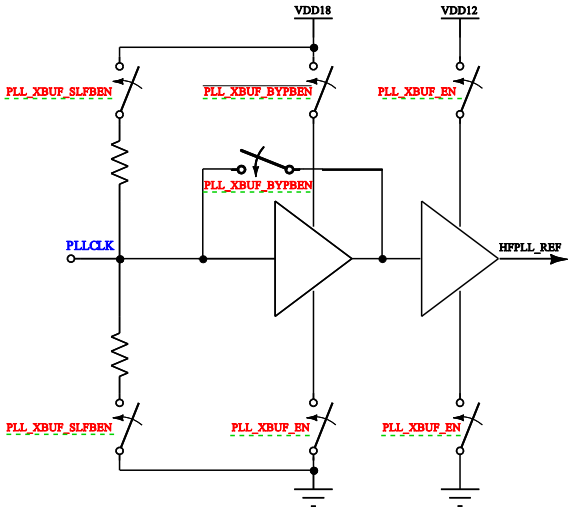
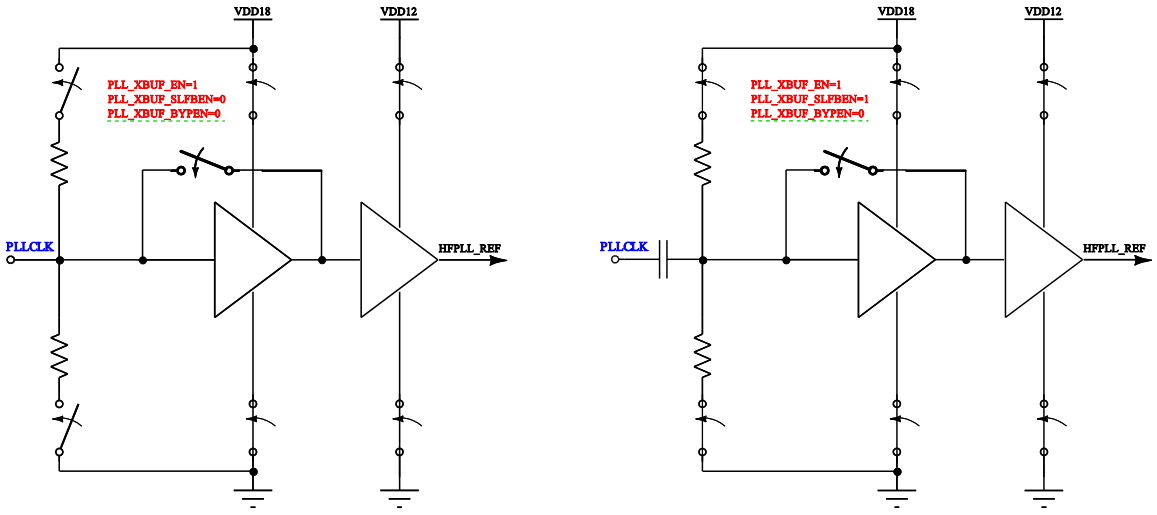


Figure 8.2: HFPLL Reference Clock Input Buffer

Both, DC and AC coupling topologies are supported as shown in Figure 8.3. Reference buffer self-biasing option should be enabled for AC-coupling configuration. For continuous LO frequency range coverage, reference frequency input should be kept between 10 and 50 MHz.



(a) DC Coupling Mode

(b) AC Coupling Mode

Figure 8.3: HFPLL Reference Clock Input Buffer Configurations

8.4 HFPLL CORE

The HFPLL core block diagram is shown in Figure 8.4. It is classic type-II, fourth-order PLL core which uses fractional-N technique with 3rd order noise shaping to synthesize and fine adjust the LO frequency value. HFPLL core can also be operated in integer-N mode, which gives the best phase noise performance at the expense of reduced frequency resolution of the synthesizer.

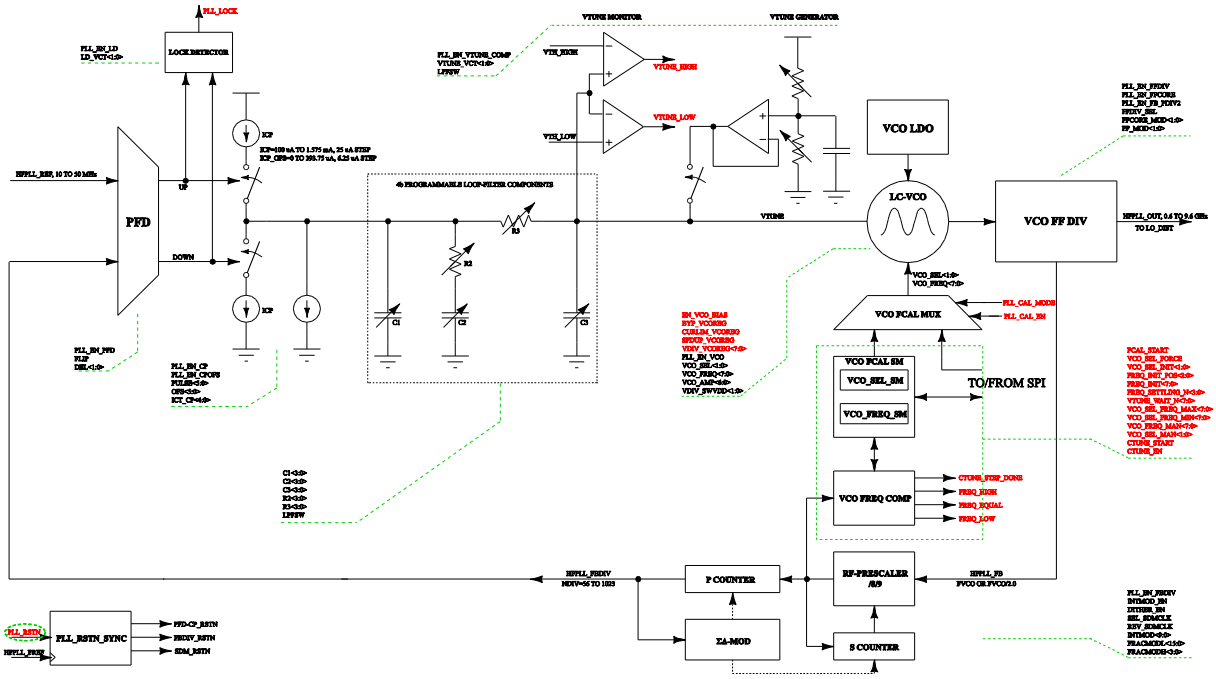


Figure 8.4: HFPLL Core

Fundamental VCO core frequency covers full octave between 4.8 and 9.6 GHz. Lower frequency sub-bands down to 300 MHz, can be synthesized by using the feed-forward divider stages implemented in FF-DIV circuit and LO Distribution Network.

Charge pump current is programmable, as well as the loop filter components, in order to enable PLL performance optimization for various applications.

PLL configuration for a desired LO frequency is obtained by calculating the required divider values and VCO settings. User can get the optimal VCO configuration for a desired carrier frequency by using the internal digital state-machine that implements the VCO frequency calibration process (or coarse frequency tuning) in the open loop mode. On the other hand, this state machine can be bypassed and the user can implement its own algorithm, through the use of SPI interface and feedback information from the HFPLL core about the PLL lock status (PLL_LOCK, VTUNE_HIGH, VTUNE_LOW) and/or results of comparing the targeted HFPLL frequency value and VCO oscillation frequency (FREQ_HIGH, FREQ_EQUAL and FREQ_LOW). More details will be given in the next sections.

8.4.1 Charge-Pump

The charge-pump circuit has 6b programmable pulse and 6b programmable offset currents. The pulse current value varies from 25 µA to 1.575 mA in 25 µA steps. Offset current value varies from 0 to 393.75 µA in 6.25 µA steps. Offset current should be used when HFPLL core operates in fractional-N mode to improve phase-noise performance. It should not exceed 25% of programmed pulse current value.

8.4.2 Loop Filter

The HFPLL loop-filter is fully integrated on-chip. It is a standard type II, third-order loop-filter topology (Figure 8.5). All components are programmable with 4b control words.

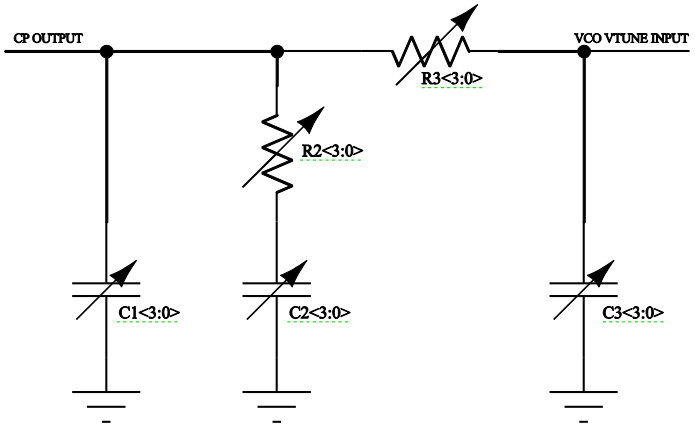


Figure 8.5: HFPLL Loop Filter

C1 capacitor value can be programmed from 0 to 18 pF with 1.2 pF steps.

C2 capacitor value can be programmed from 150 pF to 300 pF in 10 pF steps.

C3 capacitor value can be programmed from 5 pF to 23 pF with 1.2 pF steps.

R2 resistor value is given by the equation 24.6 kΩ/N where N is decimal value of 4b control word for R2.

R3 resistor value is given by the equation 14.9 kΩ/N where N is decimal value of 4b control word for R3.

8.4.3 Lock-Detection

Analog Lock Detector is implemented in the HFPLL core. This circuit monitors the phase relationship between UP and DOWN pulses at the PFD output, and produces the PLL_LOCK indicator bit to signalize whether or not HFPLL core has achieved phase-locked condition. PLL_LOCK high logic value means that HFPLL has achieved phase-locked state.

LMS8001 HFPLL core also contains two comparators which can be used to detect if the VTUNE control voltage is inside the recommended limits. All-zero output of the VTUNE monitoring circuits means that VCO VTUNE voltage is properly centered. If any output has high logic value, VTUNE voltage is outside the recommended range.

8.4.4 VCO

The LC-VCO of LMS8001 IC's HFPLL consists of three VCO cores. These cores together with the integrated feed-forward divider stages cover the LO frequency range from 300 MHz to 9.6 GHz. Each VCO core has 256 overlapping tuning curves simultaneously achieving low tuning sensitivity and good phase noise performance across very wide frequency tuning range.

Schematic block diagram of the VCO is given in Figure 8.6.

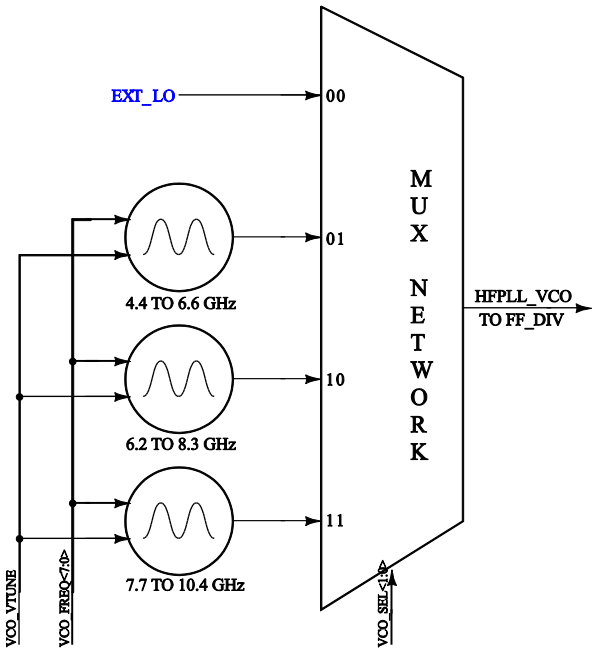


Figure 8.6: LC-VCO

VCO_SEL<1:0> control word defines the active LC-VCO core. When external LO input is selected to propagate to the output of the VCO multiplexing network, all three on-chip LC-VCO cores will be automatically powered-down.

VCO_FREQ<7:0> word controls the configuration of the binary-weighted switched-capacitor bank for the chosen VCO core. All-zero value corresponds to the lowest frequency tuning curve. All-ones value corresponds to the highest frequency tuning curve.

VCO_VTUNE analog voltage coming from the feedback action of the PLL loop, fine-tunes the VCO oscillation frequency to the desired value.

VCO Frequency Calibration

Optimal VCO core and its cap bank configuration for the desired LO frequency, will vary from sample to sample, due to the inevitable process variations. The process of searching for optimal VCO core and cap bank code for a targeted carrier frequency is often known as VCO frequency calibration or VCO coarse frequency tuning. It can be performed in open-loop or in closed-loop mode, which depends on the status of the PLL feedback loop during VCO frequency calibration process. Usually, open-loop methods are more complex but lead to the faster calibration times.

LMS8001 provides to the user several options to perform the coarse frequency tuning of the VCO inside the HFPLL_CORE block.

Before performing any kind of VCO frequency calibration, the user should carefully program the static registers inside the PLL_CONFIGURATION register bank and PLL_PROFILE_n he/she intends to use. Among other settings, it's always necessary to define the right values for integer and fractional modulus words (INTMOD_**n**<9:0>, FRACMODH_**n**<3:0> FRACMODL_**n**<15:0>) even when the PLL loop is opened during the calibration process. With n is denoted the active PLL profile selected by the user.

Two basic modes for VCO frequency calibration are provided to the user of LMS8001 IC: automatic and manual calibration mode. The mode is chosen by setting the PLL_CALIBRATION_MODE bit to either 0b0 (automatic, default) or 0b1 (manual). Either calibration mode should be first enabled by setting the PLL_CALIBRATION_EN bit to high logic value. These two bits are located inside the PLL_CONFIGURATION register bank.

Automatic Mode

In this mode, internal digital state-machine is enabled and runs the coarse-frequency tuning algorithm based on the binary-search process. The whole process is initiated by setting the sticky-bit FCAL_START to the logic-high value. State-machine clears FCAL_START bit when the VCO frequency calibration is completed.

During the VCO calibration, HFPLL feedback loop is broken. VCO tuning voltage is fixed. Default value is 0.6 V but it can be changed by choosing the value for VTUNE_VCT_**n**<1:0> control word inside the PLL_PROFILE_**n** register bank.

State-machine uses the internal VCO frequency estimator/comparator circuit (marked as VCO FREQ COMP in Figure 8.4) to drive the coarse frequency tuning process in the right direction. This circuit provides the information to the state machine about whether the VCO oscillation frequency for a given configuration is higher (FREQ_HIGH=0b1), lower (FREQ_LOW=0b1) or equal (FREQ_EQUAL=0b1) to the targeted value. The accuracy of the VCO frequency estimator/comparator circuit is determined by the value of the CTUNE_RES<1:0> control word inside the PLL_CONFIGURATION register bank. Higher value improves the resolution of the VCO coarse frequency tuning at the cost of the longer time duration of the whole process.

User can customize and configure many steps in the algorithm.

The complete process consists of several steps.

First, after starting the VCO calibration, by writing logic one to FCAL_START bit, state machine opens the HFPLL feedback loop and waits for the VCO tuning voltage to settle-down to its final value (0.6 V default) which will be used during the coarse tuning. The duration of this time can be programmed using dedicated fields inside the PLL_CONFIGURATION register bank (VTUNE_WAIT_N<7:0>) and it's reflected in a number of reference clock cycles during which VTUNE voltage is expected to settle.

After that, VCO core auto-select process starts. The purpose of this process is to find the optimal VCO core for synthesizing the targeted HFPLL frequency. Starting value for VCO_SEL<1:0> is 0b10(2), which corresponds to the medium frequency VCO core. If the desired frequency value is found to be between the oscillation frequencies of the medium VCO core for low and high frequency cap bank configurations, final value for VCO_SEL<1:0> control word is set to be 0b10(2). If the targeted frequency is below the oscillation frequency of medium VCO core for low-frequency cap bank configuration, final value for VCO_SEL<1:0> control word is set to be 0b01(1). If the targeted frequency is above the oscillation frequency of medium VCO core for high frequency cap bank configuration, state-machine sets the final value for VCO_SEL<1:0> word to be 0b11(3). Low and high frequency configuration codes for switched capacitor bank of medium-frequency VCO core are configurable and defined by the VCO_SEL_FREQ_MIN<7:0> and VCO_SEL_FREQ_MAX<7:0> fields inside the PLL_CONFIGURATION register bank.

This step in the algorithm can be skipped by forcing the fixed value for VCO_SEL<1:0> word defined by the user. This can be accomplished by filling the VCO_SEL_INIT<1:0> and VCO_SEL_FORCE fields. For example, if it's known that the desired carrier frequency can be synthesized when the high-frequency VCO core is active, user can set VCO_SEL_INIT<1:0> value to 0b11(3) and set VCO_SEL_FORCE bit to 0b1. In that case, internal state-machine skips the VCO core auto-search process, enables high-frequency VCO core and moves to the next step.

Final step in the whole process is binary-search based algorithm whose function is to find the optimal capacitor bank configuration for the selected VCO core. This process is also referred to as a coarse frequency tuning process. The center of the VCO tuning curve determined by

the final cap bank configuration should be very close to the targeted HFPLL frequency. Binary search, by default, starts from the MSB bit of the cap bank configuration code, VCO_FREQ<7:0>, and determines the final result bit by bit down to LSB. Calibration time per bit of the cap bank code mainly depends on the CTUNE_RES<1:0> word programmed by the user. The user has the freedom to speed-up the whole process by forcing several MSB bits of VCO_FREQ<7:0> control word to take fixed values from the FREQ_INIT<7:0> register and let the remaining LSB portion of the cap bank configuration code to be determined by the VCO coarse frequency tuning process. The starting bit position in the cap bank code is defined by the FREQ_INIT_POS<2:0> control word. The default value is 0b111(7) which means that the algorithm starts from the MSB bit and searches for the optimal code over whole VCO_FREQ<7:0> space, from 0 to 255.

For example, if the optimal capacitor bank code for the targeted carrier frequency is located somewhere between 160 and 191 values (32 potential solutions, 5 bits) for VCO_FREQ<7:0>, the user can optimize and shorten the calibration time in the following way. First, only the last five bits of the capacitor bank code should be determined by the coarse tuning algorithm. That can be achieved by setting the FREQ_INIT<2:0> value to 0b101(5). In that case, VCO frequency calibration state-machine will take the first three MSB bits of the capacitor bank code from the FREQ_INIT<7:0> value. Therefore, FREQ_INIT<7:0> register should be programmed to be 0b10100000(160).

After each sub-step in coarse frequency tuning algorithm, when switched capacitor bank code of active VCO core is updated, state machine waits some predefined number of reference clock cycles in order for VCO oscillation frequency to reach its final value for a given configuration. This waiting time user can program using dedicated FREQ_SETTLING_N<3:0> bits. As in the previous case for VTUNE settling time, it's expressed in the number of reference clock cycles.

After the calibration is finished, the final results and the appropriate validity bits will be available within the dedicated read-only registers (VCO_SEL_FINAL<1:0>, VCO_SEL_FINAL_VAL, VCO_FREQ_FINAL<7:0>, VCO_FREQ_FINAL_VAL). The user should not forget to copy these final values to the appropriate fields inside the desired PLL_PROFILE_**n** register bank (VCO_FREQ_**n**<7:0> and VCO_SEL_**n**<1:0>).

Finally, a brief calculation of the time required for the VCO frequency calibration in automatic calibration mode will be presented. First of all, the duration of one coarse tuning step in the implemented algorithm is given by the following equation:

$$T_{STEP} = \left(6 + 2^{CTUNE\_RES<1:0>+5}\right) / F_{REF} .$$

Initial VTUNE wait time is given with:

$$T_{VTUNE\_WAIT} = VTUNE\_WAIT\_N<7:0> / F_{REF} .$$

Wait time after each capacitor bank code update is given with:

$$T_{FREQ\_WAIT} = FREQ\_SETTLING\_N<3:0> / F_{REF} .$$

The duration of VCO core auto-search process can be calculated using:

$$T_{VCO\_SEL\_SM} = (1 - VCO\_SEL\_FORCE) \times \left(N \times T_{STEP} + N \times T_{FREQ\_WAIT} + 4 / F_{REF}\right).$$

Parameter N from previous equation can take value 1 or 2 and depends on the final results for VCO_SEL<1:0> word.

The duration of optimal capacitor bank code search process is given with the following equation:

$$T_{VCO\_FREQ\_SM} = FREQ\_INIT\_POS<2:0> \times \left(T_{STEP} + 2 / F_{REF} + T_{FREQ\_WAIT}\right) + T_{STEP} + 2 / F_{REF} .$$

The total calibration time in the automatic mode is given with following formula:

$$T_{VCO\_AUTO\_CAL} = 3/F_{REF} + T_{VTUNE\_WAIT} + T_{VCO\_SEL\_SM} + T_{VCO\_FREQ\_SM}.$$

Basic steps for configuring the LMS8001 IC when running the automatic VCO frequency calibration are presented in Figure 8.7.

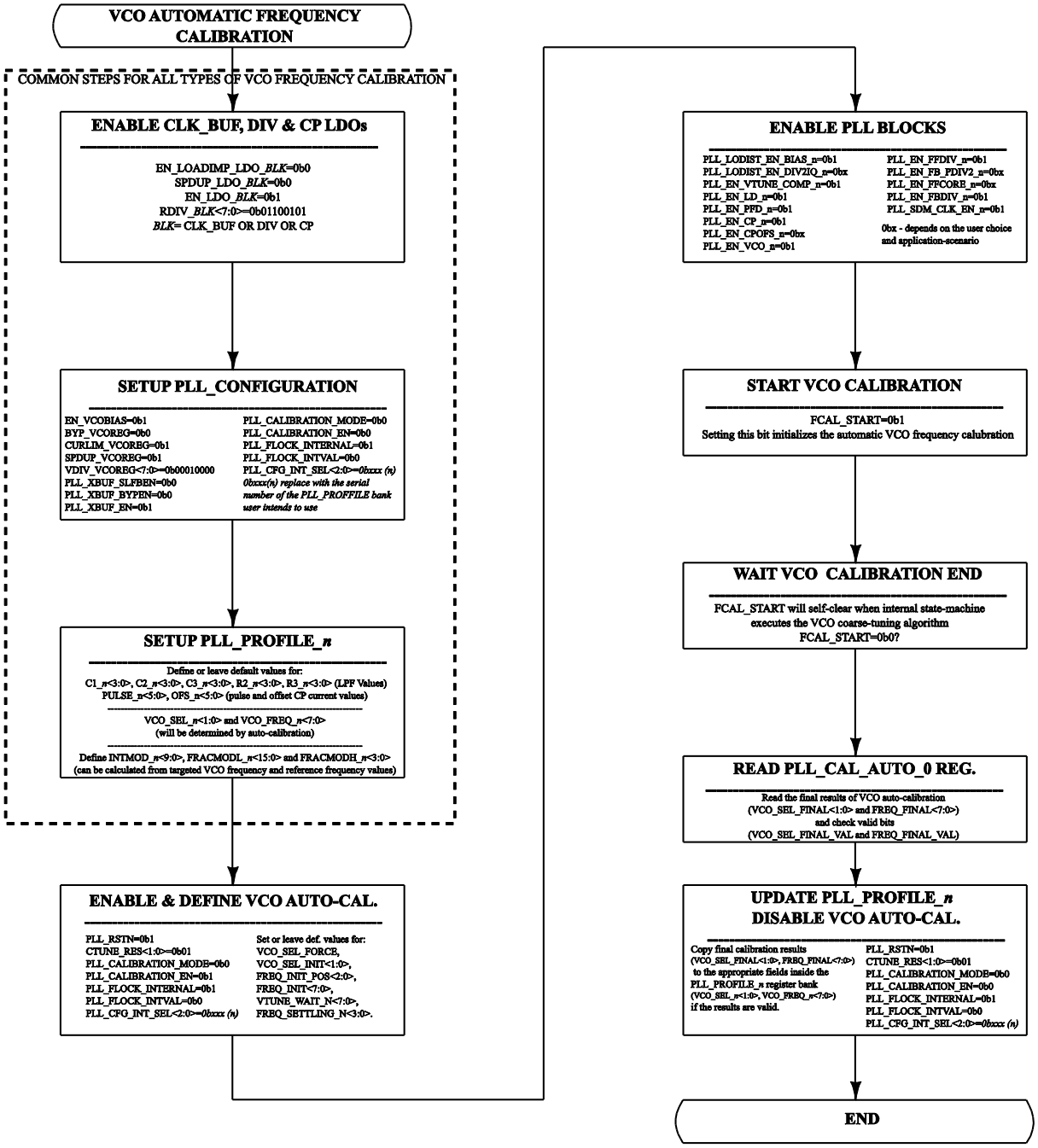


Figure 8.7: Automatic VCO Frequency Calibration

Manual Mode

In the manual calibration mode internal state-machine is disabled (PLL_CALIBRATION_EN=0b1, PLL_CALIBRATION_MODE=0b1). User can implement custom algorithm by manually configuring the PLL sub-circuits through the SPI and

dedicated register banks. Feedback information from several sub-circuits inside the HFPLL core is useful for making the decisions in the user-defined coarse-tuning algorithm.

At the beginning it should be noted that in the manual VCO calibration mode, LC-VCO does not respond to the values for VCO_SEL_**n**<1:0> and VCO_FREQ_**n**<7:0> defined in the active PLL_PROFILE_**n** register bank. Instead, separate fields are provided for the manual calibration mode and should be used for that purpose. These are VCO_SEL_MAN<1:0> and VCO_FREQ_MAN<7:0> words from PLL_CAL_MAN register inside the PLL_CONFIGURATION register bank.

Both open- and closed-loop coarse tuning algorithms can be developed in the manual calibration mode. A brief overview of both possibilities is given bellow.

Open-loop manual VCO calibration

In this case the user should use the internal VCO frequency estimator/comparator circuit to drive the coarse tuning algorithm in the right direction. In the automatic mode, internal state-machine controls this circuit and uses its output values for making the decisions.

This circuit has 2 inputs and 4 outputs which are available in the PLL_CAL_MAN register. These are CTUNE_EN(input), CTUNE_START(input), CTUNE_STEP_DONE(output), FREQ_HIGH(output), FREQ_EQUAL(output), FREQ_LOW(output). The purpose of each bit will be briefly described here.

CTUNE_EN input bit enables the VCO frequency estimator/comparator circuits. When it has logic high value, circuit is enabled and HFPLL core loop is automatically opened. VCO VTUNE voltage will be fixed to the value defined by VCO_VTUNE_**n**<1:0> word from the active PLL profile register bank. Typical settling time for the VCO tuning voltage is around 1 us.

CTUNE_START input is used for starting the process of estimating the VCO oscillation frequency for a given configuration and comparing it to the desired HFPLL frequency whose value is defined by the integer and fractional divider modulus words from the selected PLL profile register bank. Setting this bit to 0b1 starts one coarse-tuning step at the upcoming rising edge of the reference clock. It should be always set back to 0b0 after the current step is finished to reset the estimator/comparator circuit before the new calibration step is triggered again by setting this bit back to 0b1.

CTUNE_STEP_DONE output signalizes that triggered VCO calibration step is finished and the outputs are valid and ready for reading (0b1 value means step finished.).

Logic high value of FREQ_HIGH output means that the VCO oscillation frequency is higher than the targeted value.

Logic high value of FREQ_EQUAL output means that the VCO oscillation frequency is equal to the targeted value. In fact, this output can only be active when the measured difference between the real VCO oscillation frequency and targeted one is below the resolution of the estimator circuit.

Logic high value of FREQ_LOW output means that the VCO oscillation frequency is below the targeted value.

Using the described circuits and functionality, the user of LMS8001 IC is free to develop custom algorithm for VCO coarse-frequency tuning in the open-loop PLL configuration.

One basic example, with indicated basic steps and important register values that need to be defined, is presented in Figure 8.8.

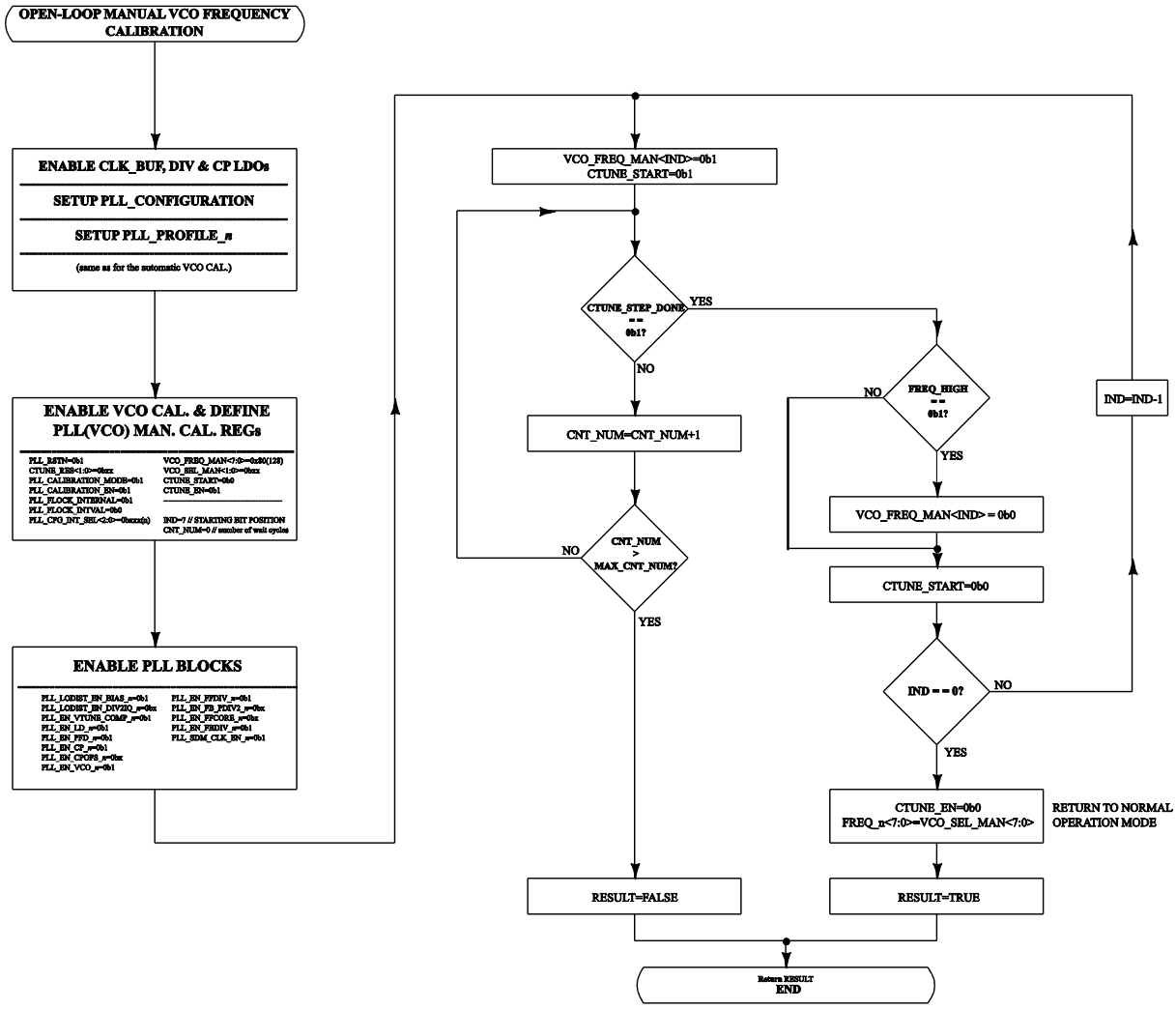


Figure 8.8: Manual Open-Loop VCO Frequency Calibration Algorithm Example

Closed-loop manual VCO calibration

In this type of VCO calibration process, PLL core loop is closed. In the case of LMS8001 IC, the HFPLL core loop will be closed when all the blocks inside the loop are powered up, CTUNE_EN and LPFSW_n bits are set to 0b0, and PLL_RSTN is equal to 0b1.

The correctness of the programmed VCO_SEL_MAN<1:0> and VCO_FREQ_MAN<7:0> settings, for the targeted LO frequency, in the closed-loop manual VCO coarse tuning can be checked using the PLL lock indicators. These are the two outputs of VTUNE comparators, VTUNE_HIGH and VTUNE_LOW bits. Besides, the output of analog lock detector, PLL_LOCK bit, can be used to double check the PLL lock status.

In order to use the VTUNE comparators, those should be enabled first. This can be achieved by setting the PLL_EN_VTUNE_COMP_n bit to 0b1. The threshold voltage levels are set to VTH_LOW=0.3 V and VTH_HIGH=0.9 V. The state of the comparator outputs are given in Table 1.

These signals can be used to perform VCO frequency calibration in closed-loop PLL mode. Assuming for now that the right choice for VCO_SEL_MAN<1:0> value is known, one can choose the appropriate value for VCO_FREQ_MAN<7:0> word in the following way. The user should find the minimum value for VCO_FREQ_MAN<7:0> word when VTUNE comparators would change the state from 0b10 to 0b00. Then the user should find the maximum value of the same word, when VCO comparator state is changed from 0b00 to

0b01. Optimum VCO capacitor bank code can be taken to be the average of these two values. More complicated algorithms are also possible including the process of selecting the right VCO core.

Table 1: VTUNE Comparator Truth Table

VTUNE_HIGH	VTUNE_LOW	STATUS
0	0	OK, VTUNE is in the recommended range.
0	1	VTUNE is too low, VTUNE<0.3 V
1	0	VTUNE is too high, VTUNE>0.9 V
1	1	When powered-down, otherwise not possible. Check SPI connection.

As in previous cases, user should take care that the final results of the implemented coarse tuning algorithm are written in the appropriate registers of the active PLL profile.

One basic example, with indicated basic steps and important register values that need to be defined, is presented in Figure 8.9.

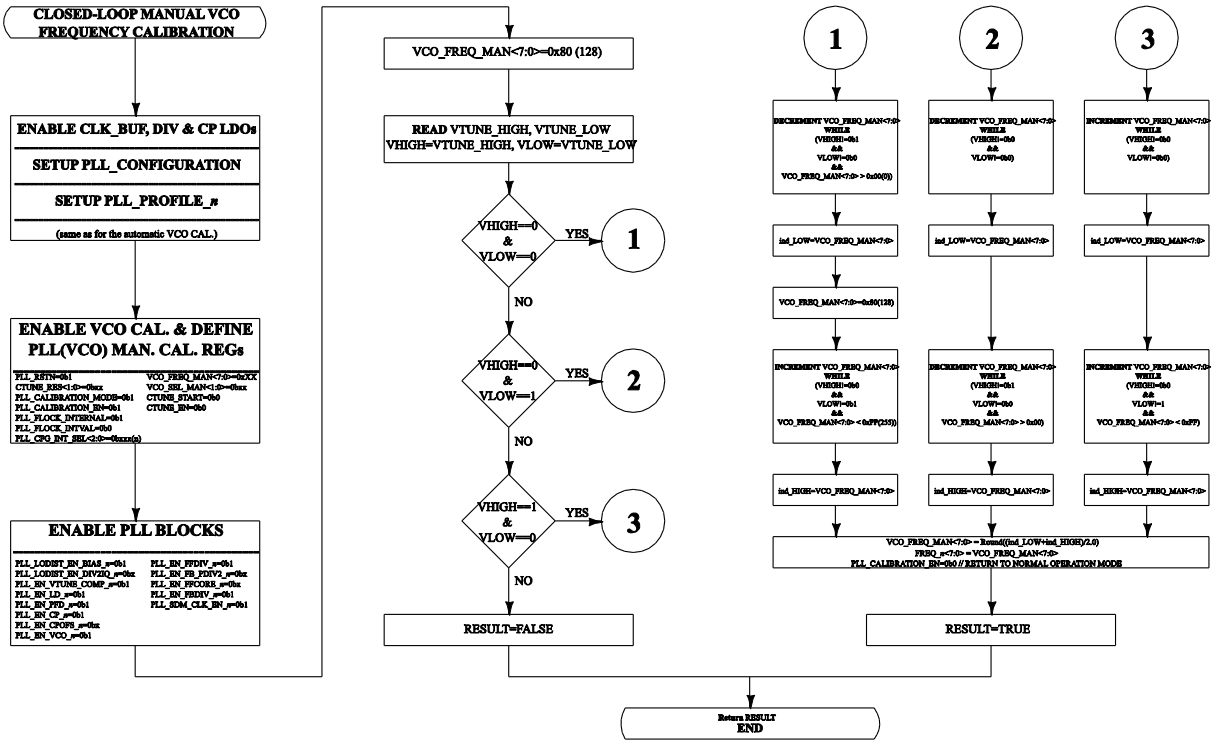


Figure 8.9: Manual Closed-Loop VCO Frequency Calibration Algorithm Example

8.4.5 FF-DIV

Block diagram of the VCO Feed-Forward Divider is given in Figure 8.10. It has two outputs, one connected to the input of the LO distribution network and the other connected to the input of RF dual-modulus frequency prescaler in the feedback divider of the HFPLL core.

VCO fundamental frequency can be divided by 2, 4 or 8 or just fed through to the input of the LO distribution network.

Input frequency of the feedback divider can be equal to the VCO fundamental frequency or scaled down by two before entering the dual-modulus prescaler stage in the feedback path.

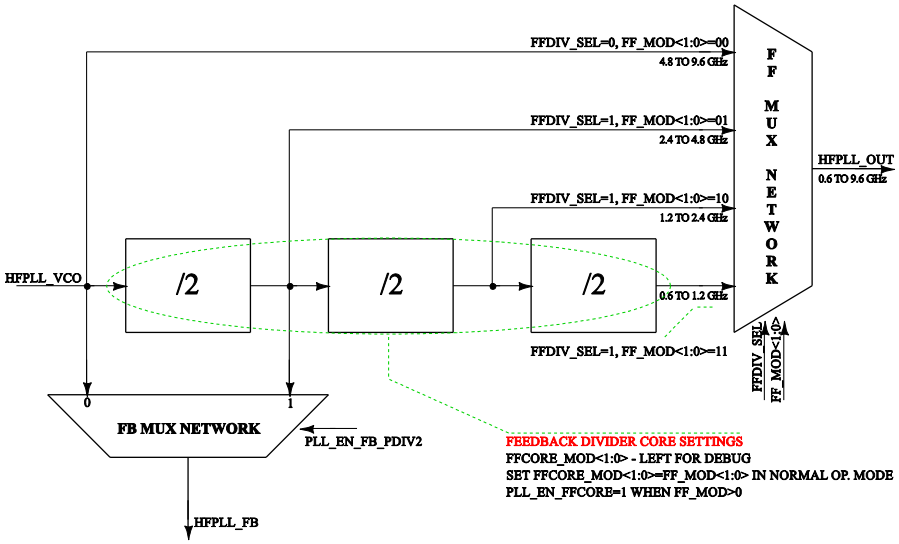


Figure 8.10: Feed-Forward Divider Block Diagram

8.5 LO Distribution Network

LO Distribution network of the LMS8001 IC is presented in Figure 8.11. It provides four LO outputs for four different RF channels integrated on-chip.

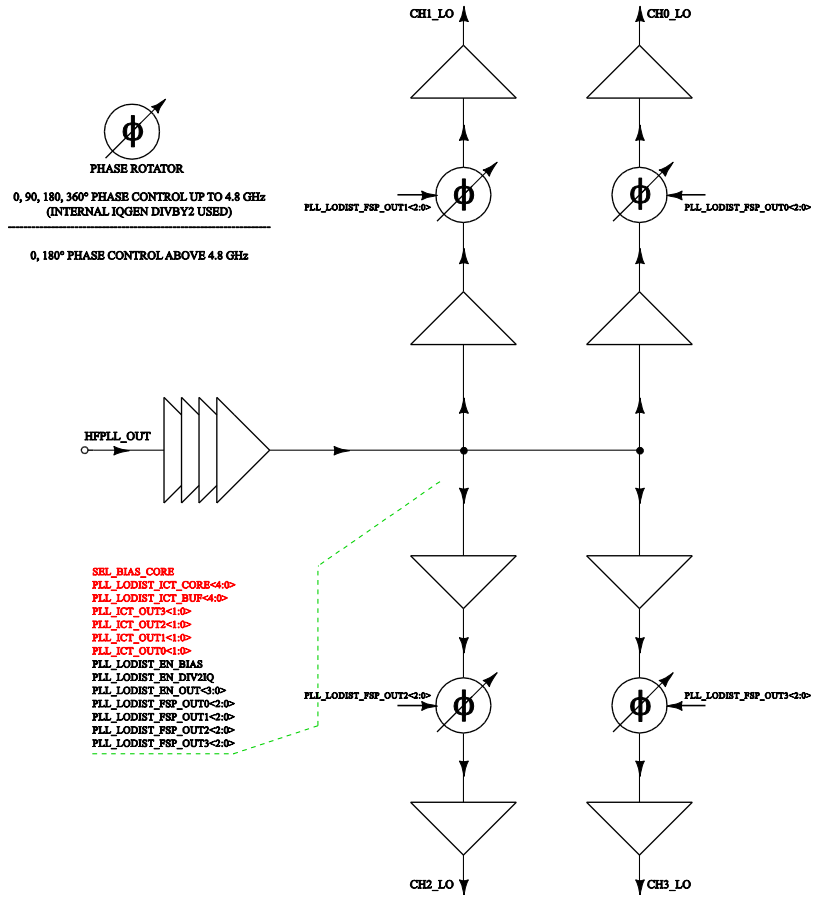


Figure 8.11: LO Distribution

For LO frequencies up to 4.8 GHz, internal IQ Divide By 2 generator and phase rotators can provide programmable LO phase shift of 0°, 90°, 180° and 270°. In this case the frequency of the signal at the particular LO output is half of the LO distribution network input signal frequency.

Above 4.8 GHz, LO phase can only be programmed to be either 0° or 180°.

LO phase shift for each RF channel can be configured independently through the SPI.

8.6 External LO

Although, the LMS8001 IC has fully self-contained frequency synthesizer, external LO signal can be applied through the dedicated input pins (differential interface) of the device. External LO signal is applied to the input of the VCO MUX network and it passes through the feed-forward divider and LO distribution network before reaching the RF channels on the chip.

8.7 HFPLL Configuration

8.7.1 Digital Control Logic

Top level structure of the digital control logic implemented in LMS8001 for configuration and control of HFPLL frequency synthesizer is shown in Figure 8.12. There are eight sets (profiles) of PLL control signal values. Profile can be selected with GPIO pins, SPI register value, or a combination, depending on how MUXSEL macro is programmed. Additionally, each set of PLL control signals can be programmed with fast lock values, to facilitate faster frequency settling upon profile change (e.g. frequency hopping applications).

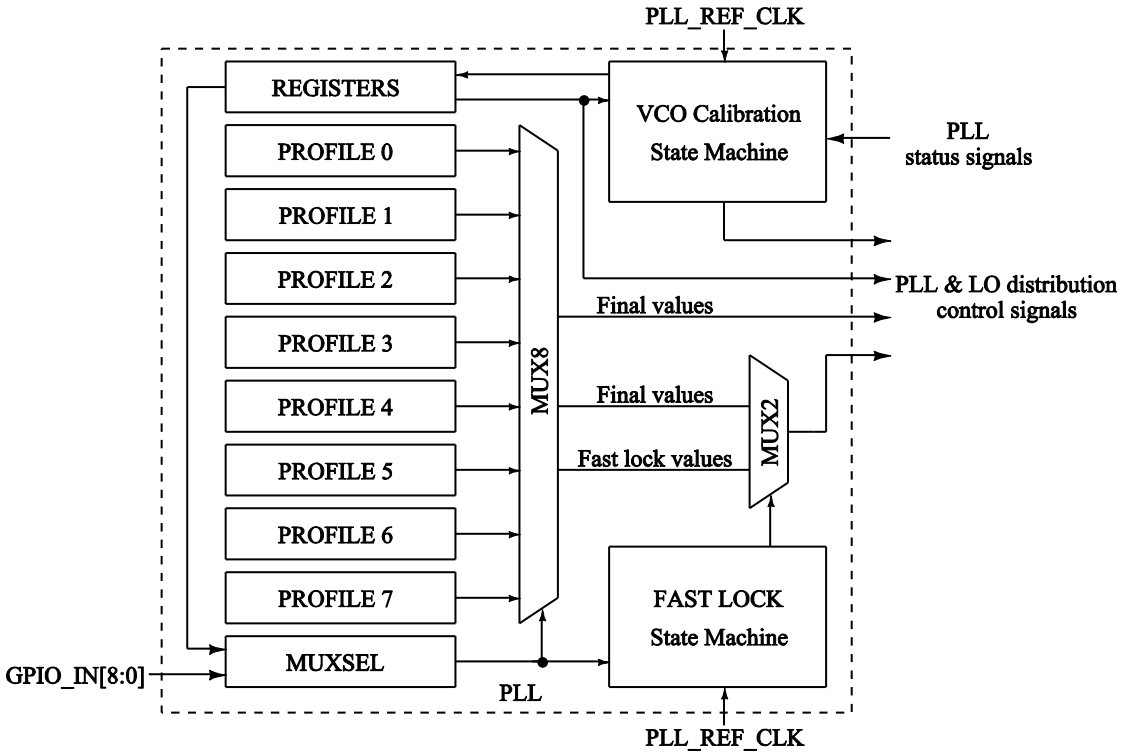


Figure 8.12: HFPLL Digital Control Logic Block Diagram

Control signals are selected from one of eight PLL profiles, stored in register banks PLL_PROFILE_**n** (**n**=0..7). Multiplexer control signals are generated by MUXSEL macros, enabling the profile selection by GPIO pins, value of SPI register, or a combination of them. A subset of control signals has two values per profile: fast lock and final values. Fast lock control logic, shown in Figure 8.13, consists of fast lock state machine, selection logic and a two input multiplexer. Fast lock state machine is triggered by a change in PLL_CFG_SEL[2:0] signals, caused by either change on GPIO pin or a write to SPI bitfield PLL_CFG_INT_SEL[2:0] depending on the MUXSEL configuration. When the fast lock machine is triggered, it outputs FAST_LOCK_SEL=1 during the next FLOCK_N[9:0] cycles of PLL reference frequency. If PLL_FLOCK_INTERNAL=0, the value of FAST_LOCK_SEL is passed to multiplexer and the fast lock values are used as control values for FLOCK_N clocks, as shown in timing diagram in Figure 8.14. Fast lock functionality can be manually controlled, or disabled, by setting PLL_FLOCK_INTERNAL=1, when the bit PLL_FLOCK_INTVAL directly controls the fast lock multiplexer.

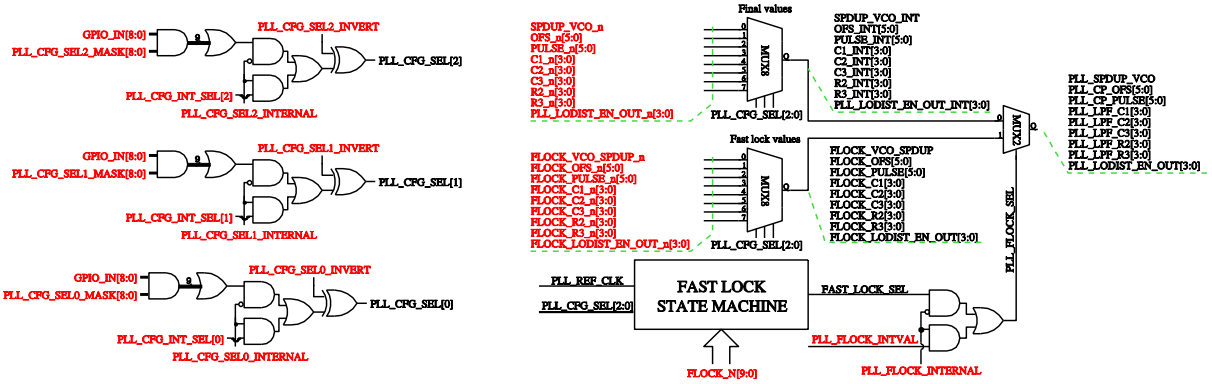


Figure 8.13: Fast lock control logic

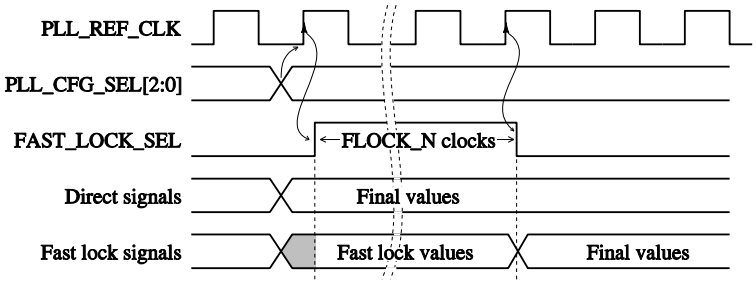


Figure 8.14: Fast lock timing diagram

Table 2: LMS8001 Memory Space for HFPLL Configuration

Register Bank	Start Address	End Address
PLL_CONFIGURATION	0x4000	0x401F
PLL_PROFILE_0	0x4100	0x410F
PLL_PROFILE_1	0x4110	0x411F
PLL_PROFILE_2	0x4120	0x412F
PLL_PROFILE_3	0x4130	0x413F
PLL_PROFILE_4	0x4140	0x414F
PLL_PROFILE_5	0x4150	0x415F
PLL_PROFILE_6	0x4160	0x416F
PLL_PROFILE_7	0x4170	0x417F

Registers relevant to PLL & LO distribution are grouped in PLL_CONFIGURATION and PLL_PROFILE_**n** (**n**=0..7) register banks (Table 2).

8.7.2 HFPLL Fast-Lock Mode

Overview

The LMS8001 IC provides to the user the option to optimize its system and achieve faster frequency settling times. There are eight different PLL_PROFILE registers in the memory space of the device that store sets of the HFPLL programming information. Only one PLL_PROFILE is active at the time and its settings are multiplexed to the HFPLL control inputs. The PLL_PROFILE registers can eliminate the most of LMS8001 HFPLL programming by providing 8 full PLL configurations for different channel frequencies to be stored in the chip memory and to easily switch between, for achieving faster frequency changes. Choosing the active PLL profile number can be done through the SPI or by using the GPIO pins of the device.

Particular profile must be configured correctly before the use. The new PLL profile can be configured or just updated at any convenient time for particular application. It's recommended to configure at least one PLL profile at the power-up.

Usual sequence when defining one PLL profile is PLL configuration, VCO frequency calibration and locking of the HFPLL to the particular frequency. The user alone should ensure copying of final calibration results from dedicated registers to the memory space reserved for the desired PLL profile. If automatic VCO frequency calibration process is used, it's recommended to use the highest accuracy setting for the VCO frequency calibration (CTUNE_RES<1:0>=0b11). This setting leads to the longest calibration time but the final configuration should be very robust to the environmental variations and appropriate for the longer use.

Additionally, the user of LMS8001 IC also has the possibility to define the initial values for charge pump current and loop filter components during the programmable amount of time for each PLL profile. This option can be used to increase the loop bandwidth of the PLL when locking process starts and further reduce the total settling time of the frequency synthesizer. These bit fields are also available in the PLL profile register banks.

Initial Wide Loop BW Option

The possibility to switch between two predefined PLL profiles without the need to run VCO coarse-tuning process might not lead to sufficient reduction of PLL lock time for some applications. Therefore, possibility to increase the loop BW for some programmable amount of time by setting the initial values for charge-pump current and loop-filter components is provided. This option should further improve the locking speed of the synthesizer. After the defined time interval expires, charge-pump current and LPF components are set back to the default values (Figure 8.14).

Assuming that user wants to initially expand the PLL loop bandwidth N times, then, in order to maintain the same level of loop stability for both wide and normal bandwidth modes, the following relations should apply:

$FLOCK\_PULSE = PULSE \times N^2$, CP pulse current value during fast-lock

$FLOCK\_R2 = R2/N$, resistor R2 value during fast-lock

$FLOCK\_R3 = R3/N$, resistor R3 value during fast-lock

FLOCK prefix in previous formulas are used to represent the “fast-lock” values, i.e. initial values for charge-pump current and loop-filter resistors for wide loop bandwidth PLL configuration. The same prefix is used to mark the dedicated registers in each PLL profile register bank.

In some cases, it is possible that some values will max out or min out before it could actually take the calculated values using the previous formulas. Even then, the improvements in synthesizer's settling time are possible. The capacitor values are also available to take different values during the fast-lock, wide bandwidth, and normal, narrow bandwidth, operation modes. The LMS8001 IC user will need to optimize the fast-lock configurations during product development.

Fast-Lock Mode Configuration

The Fast-Lock registers are available in each of the eight PLL profile banks. The names of the registers are PLL_FLOCK_CFG1_**n**, PLL_FLOCK_CFG2_**n** and PLL_FLOCK_CFG3_**n**, where n can be from 0 to 7 and represents the PLL profile serial number.

The basic procedure for configuring the PLL_PROFILE_n for using it in fast-lock mode is described below:

1. Determine the fast-lock mode time duration and write the value to FLOCK_N_**n**<9:0> field in PLL_FLOCK_CFG3_n register. This time is expressed as the number of reference clock cycles. The default value is 0b0110010000 (400), which equals 10 μs for the 40 MHz reference frequency value. If not using fast-lock mode, write all zeros to FLOCK_N_**n**<9:0> word.
2. Configure the parameters for normal-mode in PLL_PROFILE_**n** register bank:
   - Enable and configure all the relevant sub-blocks
   - Run VCO Frequency Calibration
   - Copy final results of VCO Frequency Calibration to the dedicated fields inside the PLL_PROFILE_**n** register bank
3. Configure the parameters in fast-lock registers. The parameters which can be programmed are loop-filter component values (FLOCK_R3_**n**<3:0>, FLOCK_R2_**n**<3:0>, FLOCK_C3_**n**<3:0>, FLOCK_C2_**n**<3:0> and FLOCK_C1_**n**<3:0>), charge-pump pulse and offset current values (FLOCK_PULSE_**n**<5:0> and FLOCK_OFS_**n**<5:0>), VCO speed-up bit (FLOCK_VCO_SPDUP_**n**) and enable signals for output buffers of the LO Distribution Network (FLOCK_LODIST_EN_OUT_**n**<3:0>).

   The option to power-down LO buffers during the fast-lock mode can be useful to further reduce the level of unwanted emission during frequency acquisition process in HFPLL core.

8.7.3 HFPLL Frequency Calculation

The LC-VCO inside the HFPLL of LMS8001 IC covers frequency range from 4.8 up to 9.6 GHz. Lower frequency bands are synthesized using feed-forward divider stages. Output of the VCO drives the feedback divider or it can be first scaled down in frequency two times as illustrated in Figure 8.10.

There are four divide by 2 output stages in total. Three of them are placed inside the feed-forward divider circuit as shown in Figure 8.10. The last divide by 2 stage is inside the LO distribution network and it can also be used for quadrature generation.

Therefore, the output VCO frequency, when HFPLL core is in lock, can be calculated using the following equation:

$$F_{VCO} = 2^{PLL\_EN\_PDIV2\_n} \times \left(N_{INT\_n} + N_{FRAC\_n} / 2^{20}\right) \times F_{REF} .$$

The HFPLL core output frequency is given with:

$$F_{HFPLL} = \frac{F_{VCO}}{2^{FF\_MOD\_n<2:0>}} .$$

The final value of LO frequency for channel A (numbered as 0) can be calculated using:

$$F_{LO} = \frac{F_{HFPLL}}{2^{(1-PLL\_LO\_DIST\_FSP\_OUT0\_n<2>)}} .$$

When LO frequency for a RF channel is known together with the feed-forward divider configuration, the integer and fractional modulus words for feedback divider can be calculated using following equations:

$$N_{INT\_n} = Floor\left(\frac{F_{VCO}}{2^{PLL\_EN\_FB\_PDIV2} \times F_{REF}}\right),$$

$$F_{FRAC\_n} = Round\left(2^{20} \times \left(\frac{F_{VCO}}{2^{PLL\_EN\_FB\_PDIV2\_n} \times F_{REF}} - N_{INT\_n}\right)\right).$$

Frequency step size of the complete LMS8001 frequency synthesizer is given with:

$$F_{STEP} = \frac{2^{PLL\_EN\_FB\_PDIV2\_n} \times F_{REF}}{2^{20} \times 2^{FF\_MOD\_n<2:0>} \times 2^{(1-PLL\_LO\_DIST\_FSP\_OUT0\_n<2>)}} .$$

In previous equations, with suffix "_n" is marked the serial number of the active PLL profile register bank.

9

Register banks

LMS8001 address space is divided into register banks, listed in Table 1.

Table 3: LMS8001 Memory Map

Register Bank	Start Address	End Address	Comment
ChipConfig	0x0000	0x000F	
BiasLDOConfig	0x0010	0x001F	
Channel_A	0x1000	0x101F	Relevant only for LMS8001A
Channel_B	0x1020	0x103F	Relevant only for LMS8001A
Channel_C	0x1040	0x105F	Relevant only for LMS8001A
Channel_D	0x1060	0x107F	Relevant only for LMS8001A
HLMIXA	0x2000	0x200F	Relevant only for LMS8001B
HLMIXB	0x2010	0x201F	Relevant only for LMS8001B
HLMIXC	0x2020	0x202F	Relevant only for LMS8001B
HLMIXD	0x2030	0x203F	Relevant only for LMS8001B
PLL_CONFIGURATION	0x4000	0x401F	
PLL_PROFILE_0	0x4100	0x410F	
PLL_PROFILE_1	0x4110	0x411F	
PLL_PROFILE_2	0x4120	0x412F	
PLL_PROFILE_3	0x4130	0x413F	
PLL_PROFILE_4	0x4140	0x414F	
PLL_PROFILE_5	0x4150	0x415F	
PLL_PROFILE_6	0x4160	0x416F	
PLL_PROFILE_7	0x4170	0x417F	

9.1 Register bank ChipConfig(0x0000 – 0x001F)

Register bank ChipConfig contains registers related to chip configuration.

0x0000		SPIConfig (0x001F)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
6	0	SPI_SDIO_DS	RW	Driver strength of SPI_SDIO pad. 0 – Driver strength is 4mA (default) 1 – Driver strength is 8mA
5	0	SPI_SDO_DS	RW	Driver strength of SPI_SDO pad. 0 – Driver strength is 4mA (default) 1 – Driver strength is 8mA
4	1	SPI_SDIO_PE	RW	Pull up control of SPI_SDIO pad. 0 – Pull up disengaged 1 – Pull up engaged (default)
3	1	SPI_SDO_PE	RW	Pull up control of SPI_SDO pad. 0 – Pull up disengaged 1 – Pull up engaged (default)
2	1	SPI_SCLK_PE	RW	Pull up control of SPI_SCLK pad. 0 – Pull up disengaged 1 – Pull up engaged (default)
1	1	SPI_SEN_PE	RW	Pull up control of SPI_SEN pad. 0 – Pull up disengaged 1 – Pull up engaged (default)
0	1	SPIMODE	RW	SPI communication mode. 0 – 3 wire mode 1 – 4 wire mode (default)

0x0004		GPIOOutData (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
8:0	000000000	GPIO_OUT_SPI<8:0>	RW	Output data for GPIO pads from SPI

0x0005		GPIOOUT_SEL0 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
14:12	000	GPIO4_SEL<2:0>	RW	GPIO4 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved
11:9	000	GPIO3_SEL<2:0>	RW	GPIO3 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved
8:6	000	GPIO2_SEL<2:0>	RW	GPIO2 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved

0x0005		GPIOOUT_SEL0 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
5:3	000	GPIO1_SEL<2:0>	RW	GPIO1 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved
2:0	000	GPIO0_SEL<2:0>	RW	GPIO0 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved

0x0006		GPIOOUT_SEL1 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
14:12	000	GPIO9_SEL<2:0>	RW	GPIO9 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved
11:9	000	GPIO8_SEL<2:0>	RW	GPIO8 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved
8:6	000	GPIO7_SEL<2:0>	RW	GPIO7 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved
5:3	000	GPIO6_SEL<2:0>	RW	GPIO6 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved
2:0	000	GPIO5_SEL<2:0>	RW	GPIO5 source select 000 – from SPI 001 – PLL_LOCK 010 – VTUNE_LOW 011 – VTUNE_HIGH 100 – Fast lock active others – reserved

0x0008		GPIOInData (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
8:0	000000000	GPIO_IN<8:0>	R	Data read from GPIO pads.

0x0009		GPIOConfig_PE (0x03FF)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
8:0	111111111	GPIO_PE<8:0>	RW	GPIO pull up control 0 – Pull up disengaged 1 – Pull up engaged (default)

0x000A		GPIOConfig_DS (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
8:0	000000000	GPIO_DS<8:0>	RW	GPIO drive strength 0 – Driver strength is 4mA (default) 1 – Driver strength is 8mA

0x000B		GPIOConfig_IO (0x03FF)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
8:0	111111111	GPIO_InO<8:0>	RW	GPIO input/output control 0 – Pin is output 1 – Pin is input (default)

0x000C		TEMP_SENS (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	TEMP_SENS_EN	RW	Enable the temperature sensor biasing.
9	0	TEMP_SENS_CLKEN	RW	Temperature sensor clock enable
8	0	TEMP_START_CONV	RW	Start the temperature conversion. Bit is cleared when the conversion is complete.
7:0	00000000	TEMP_READ<7:0>	R	Readout of temperature sensor

0x000F		ChipInfo (0x4040)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:11	01000	VER<4:0>	R	Chip version. 01000 – Chip version is 8.
10:6	00001	REV<4:0>	R	Chip revision. 00001 – Chip revision is 1.
5:0	000000	MASK<5:0>	R	Chip mask. 000000 – Chip mask is 0.

9.2 Register bank BiasLDOConfig (0x0010 – 0x001F)

Register bank BiasLDOConfig contains control registers for main bias and LDOs.

0x0010		BiasConfig (0x1400)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
12	1	PD_CALIB_COMP	RW	Calibration comparator power down. 0 – Enabled 1 – Powered down (default)
11	0	RP_CALIB_COMP	R	Comparator output. Used in rppolywo calibration algorithm.
10:6	10000	RP_CALIB_BIAS<4:0>	RW	Calibration code for rppolywo. This code is set by calibration algorithm. Default value : 10000 (16)
4	0	PD_FRP_BIAS	RW	Power down signal for Fix/RP 0 – Enabled (default) 1 – Powered down
3	0	PD_F_BIAS	RW	Power down signal for Fix 0 – Enabled (default) 1 – Powered down
2	0	PD_PTRP_BIAS	RW	Power down signal for PTAT/RP block 0 – Enabled (default) 1 – Powered down
1	0	PD_PT_BIAS	RW	Power down signal for PTAT block 0 – Enabled (default) 1 – Powered down

0x0010		BiasConfig (0x1400)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
0	0	PD_BIAS	RW	Enable signal for central bias block 0 – Sub blocks may be selectively powered down (default) 1 – Powers down all BIAS blocks

0x0011		LOBUFA_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_LOBUFA	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_LOBUFA	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_LOBUFA	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_LOBUFA<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x0012		LOBUFB_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_LOBUFB	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_LOBUFB	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_LOBUFB	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_LOBUFB<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x0013		LOBUFC_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_LOBUFC	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_LOBUFC	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_LOBUFC	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_LOBUFC<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x0014		LOBUFD_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_LOBUFD	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_LOBUFD	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_LOBUFD	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_LOBUFD<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x0015		HFLNAA_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_HFLNAA	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_HFLNAA	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_HFLNAA	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_HFLNAA<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x0016		HFLNAB_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_HFLNAB	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_HFLNAB	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_HFLNAB	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_HFLNAB<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x0017		HFLNAC_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_HFLNAC	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_HFLNAC	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_HFLNAC	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled

0x0017		HFLNAC_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7:0	01100101	RDIV_HFLNAC<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x0018		HFLNAD_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_HFLNAD	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_HFLNAD	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_HFLNAD	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_HFLNAD<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x001A		CLK_BUF_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_CLK_BUF	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_CLK_BUF	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_CLK_BUF	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_CLK_BUF<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x001B		PLL_DIV_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_PLL_DIV	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_PLL_DIV	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_PLL_DIV	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_PLL_DIV<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x001C		PLL_CP_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_PLL_CP	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_PLL_CP	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	EN_LDO_PLL_CP	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_PLL_CP<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

0x001F		DIG_CORE_LDO_Config (0x0065)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	EN_LOADIMP_LDO_DIG_CORE	RW	Enables the load dependent bias to optimize the load regulation 0 – Constant bias (default) 1 – Load dependant bias
9	0	SPDUP_LDO_DIG_CORE	RW	Short the noise filter resistor to speed up the settling time 0 – Noise filter resistor in place (default) 1 – Noise filter resistor bypassed
8	0	PD_LDO_DIG_CORE	RW	Enables the LO buffer LDO 0 – LDO powered down (default) 1 – LDO enabled
7:0	01100101	RDIV_DIG_CORE<7:0>	RW	Controls the output voltage of the LO buffer LDO by setting the resistive voltage divider ratio. Vout = 860 mV + 3.92 mV * RDIV Default : 01100101 (101) Vout = 1.25 V

9.3 Register bank Channel_x

Note: This register bank is relevant only for LMS8001A.

Register bank Channel_x (x=A, B, C, D) contains control registers for RF channel. Register addresses are given relative to channel base address. Channel base addresses are given in the following table.

Channel	Base Address
Channel_A	0x1000
Channel_B	0x1020
Channel_C	0x1040
Channel_D	0x1060

+0x00		CHx_MIX_ICT (0x0210)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
9:5	10000	/	RW	Reserved
4:0	10000	/	RW	Reserved

+0x01		CHx_HFPAD_ICT (0x0210)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	CHx_PA_ILIN2X	RW	Double the linearization bias current 0 – Ilin * 1 (default) 1 – Ilin * 2
9:5	10000	CHx_PA_ICT_LIN<4:0>	RW	Controls the bias current of linearization section of HFPAD I = Inom * CHx_PA_ICT_LIN/16 Default : 16
4:0	10000	CHx_PA_ICT_MAIN<4:0>	RW	Controls the bias current of main gm section of HFPAD I = Inom * CHx_MIXA_ICT/16 Default : 16

+0x04		CHx_PD0 (0x00BF)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7	1	CHx_PA_R50_EN0	RW	Controls the switch in series with 50 Ω resistor to ground at HFPAD input. 0 – Switch is open 1 – Switch is closed
6	0	CHx_PA_BYPASS0	RW	Controls the HFPAD bypass switches. 0 – HFPAD in not bypassed 1 – HFPAD is bypassed Note : HFPAD must be manually disabled when bypassed.
5	1	CHx_PA_PD0	RW	Power down for HFPAD 0 – Enabled 1 – Powered down
4	1	CHx_MIXB_LOBUFF_PD0	RW	Power down for MIXB LO buffer 0 – Enabled 1 – Powered down
3	1	/	RW	Reserved
2	1	CHx_MIXA_LOBUFF_PD0	RW	Power down for MIXA LO buffer 0 – Enabled 1 – Powered down
1	1	/	RW	Reserved
0	1	CHx_LNA_PD0	RW	Power down for LNA 0 – Enabled 1 – Powered down

+0x05		CHx_PD1 (0x00BF)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7	1	CHx_PA_R50_EN1	RW	Controls the switch in series with 50 Ω resistor to ground at HFPAD input. 0 – Switch is open 1 – Switch is closed
6	0	CHx_PA_BYPASS1	RW	Controls the HFPAD bypass switches. 0 – HFPAD in not bypassed 1 – HFPAD is bypassed Note : HFPAD must be manually disabled when bypassed.
5	1	CHx_PA_PD1	RW	Power down for HFPAD 0 – Enabled 1 – Powered down
4	1	CHx_MIXB_LOBUFF_PD1	RW	Power down for MIXB LO buffer 0 – Enabled 1 – Powered down
3	1	/	RW	Reserved
2	1	CHx_MIXA_LOBUFF_PD1	RW	Power down for MIXA LO buffer 0 – Enabled 1 – Powered down
1	1	/	RW	Reserved
0	1	CHx_LNA_PD1	RW	Power down for LNA 0 – Enabled 1 – Powered down

+0x06		CHx_PD2 (0x00BF)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7	1	CHx_PA_R50_EN2	RW	Controls the switch in series with 50 Ω resistor to ground at HFPAD input. 0 – Switch is open 1 – Switch is closed
6	0	CHx_PA_BYPASS2	RW	Controls the HFPAD bypass switches. 0 – HFPAD in not bypassed 1 – HFPAD is bypassed Note : HFPAD must be manually disabled when bypassed.
5	1	CHx_PA_PD2	RW	Power down for HFPAD 0 – Enabled 1 – Powered down
4	1	CHx_MIXB_LOBUFF_PD2	RW	Power down for MIXB LO buffer 0 – Enabled 1 – Powered down
3	1	/	RW	Reserved
2	1	CHx_MIXA_LOBUFF_PD2	RW	Power down for MIXA LO buffer 0 – Enabled 1 – Powered down
1	1	/	RW	Reserved
0	1	CHx_LNA_PD2	RW	Power down for LNA 0 – Enabled 1 – Powered down

+0x07		CHx_PD3 (0x00BF)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7	1	CHx_PA_R50_EN3	RW	Controls the switch in series with 50 Ω resistor to ground at HFPAD input. 0 – Switch is open 1 – Switch is closed
6	0	CHx_PA_BYPASS3	RW	Controls the HFPAD bypass switches. 0 – HFPAD in not bypassed 1 – HFPAD is bypassed Note : HFPAD must be manually disabled when bypassed.
5	1	CHx_PA_PD3	RW	Power down for HFPAD 0 – Enabled 1 – Powered down
4	1	CHx_MIXB_LOBUFF_PD3	RW	Power down for MIXB LO buffer 0 – Enabled 1 – Powered down
3	1	/	RW	Reserved
2	1	CHx_MIXA_LOBUFF_PD3	RW	Power down for MIXA LO buffer 0 – Enabled 1 – Powered down
1	1	/	RW	Reserved
0	1	CHx_LNA_PD3	RW	Power down for LNA 0 – Enabled 1 – Powered down

+0x08		CHx_LNA_CTRL0 (0x8428)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:11	10000	CHx_LNA_ICT_LIN0<4:0>	RW	Controls the bias current of linearization section of LNA I = Inom * CHx_LNA_ICT_LIN/16 Default : 16
10:6	10000	CHx_LNA_ICT_MAIN0<4:0>	RW	Controls the bias current of main gm section of LNA I = Inom * CHx_LNA_ICT_MAIN/16 Default : 16
5:4	10	CHx_LNA_CGSCTRL0<1:0>	RW	Controls the additional LNA input device gate-source capacitance to control the Q of the matching circuit and enable the trade-off between the gain, NF and the linearity. Increase of this capacitance lowers the Q, lowers the gain, increases the NF, and provides better linearity.

+0x08		CHx_LNA_CTRL0 (0x8428)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
3:0	1000	CHx_LNA_GCTRL0<3:0>	RW	Controls the LNA gain Gain = Gain_max – CHx_LNA_GCTRL * (approx.) 0.5 dB

+0x09		CHx_LNA_CTRL1 (0x8428)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:11	10000	CHx_LNA_ICT_LIN1<4:0>	RW	Controls the bias current of linearization section of LNA I = Inom * CHx_LNA_ICT_LIN/16 Default : 16
10:6	10000	CHx_LNA_ICT_MAIN1<4:0>	RW	Controls the bias current of main gm section of LNA I = Inom * CHx_LNA_ICT_MAIN/16 Default : 16
5:4	10	CHx_LNA_CGSCTRL1<1:0>	RW	Controls the additional LNA input device gate-source capacitance to control the Q of the matching circuit and enable the trade-off between the gain, NF and the linearity. Increase of this capacitance lowers the Q, lowers the gain, increases the NF, and provides better linearity.
3:0	1000	CHx_LNA_GCTRL1<3:0>	RW	Controls the LNA gain Gain = Gain_max – CHx_LNA_GCTRL * (approx.) 0.5 dB

+0x0A		CHx_LNA_CTRL2 (0x8428)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:11	10000	CHx_LNA_ICT_LIN2<4:0>	RW	Controls the bias current of linearization section of LNA I = Inom * CHx_LNA_ICT_LIN/16 Default : 16
10:6	10000	CHx_LNA_ICT_MAIN2<4:0>	RW	Controls the bias current of main gm section of LNA I = Inom * CHx_LNA_ICT_MAIN/16 Default : 16
5:4	10	CHx_LNA_CGSCTRL2<1:0>	RW	Controls the additional LNA input device gate-source capacitance to control the Q of the matching circuit and enable the trade-off between the gain, NF and the linearity. Increase of this capacitance lowers the Q, lowers the gain, increases the NF, and provides better linearity.
3:0	1000	CHx_LNA_GCTRL2<3:0>	RW	Controls the LNA gain Gain = Gain_max – CHx_LNA_GCTRL * (approx.) 0.5 dB

+0x0B		CHx_LNA_CTRL3 (0x8428)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:11	10000	CHx_LNA_ICT_LIN3<4:0>	RW	Controls the bias current of linearization section of LNA I = Inom * CHx_LNA_ICT_LIN/16 Default : 16
10:6	10000	CHx_LNA_ICT_MAIN3<4:0>	RW	Controls the bias current of main gm section of LNA I = Inom * CHx_LNA_ICT_MAIN/16 Default : 16
5:4	10	CHx_LNA_CGSCTRL3<1:0>	RW	Controls the additional LNA input device gate-source capacitance to control the Q of the matching circuit and enable the trade-off between the gain, NF and the linearity. Increase of this capacitance lowers the Q, lowers the gain, increases the NF, and provides better linearity.
3:0	1000	CHx_LNA_GCTRL3<3:0>	RW	Controls the LNA gain Gain = Gain_max – CHx_LNA_GCTRL * (approx.) 0.5 dB

+0x0C		CHx_PA_CTRL0 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7:4	0000	CHx_PA_LIN_LOSS0<3:0>	RW	Controls the gain of HFPAD linearizing section Pout = Pout_max – Loss
3:0	0000	CHx_PA_MAIN_LOSS0<3:0>	RW	Controls the gain of HFPAD main section Pout = Pout_max – Loss

+0x0D		CHx_PA_CTRL1 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7:4	0000	CHx_PA_LIN_LOSS1<3:0>	RW	Controls the gain of HFPAD linearizing section Pout = Pout_max – Loss
3:0	0000	CHx_PA_MAIN_LOSS1<3:0>	RW	Controls the gain of HFPAD main section Pout = Pout_max – Loss

+0x0E		CHx_PA_CTRL2 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7:4	0000	CHx_PA_LIN_LOSS2<3:0>	RW	Controls the gain of HFPAD linearizing section Pout = Pout_max – Loss
3:0	0000	CHx_PA_MAIN_LOSS2<3:0>	RW	Controls the gain of HFPAD main section Pout = Pout_max – Loss

+0x0F		CHx_PA_CTRL3 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7:4	0000	CHx_PA_LIN_LOSS3<3:0>	RW	Controls the gain of HFPAD linearizing section Pout = Pout_max – Loss
3:0	0000	CHx_PA_MAIN_LOSS3<3:0>	RW	Controls the gain of HFPAD main section Pout = Pout_max – Loss

+0x10		CHx_PD_SEL0 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	CHx_PD_SEL0_INTERNAL	RW	CHx PD control signals multiplexer SEL0 signal is generated 0 – from GPIO & CHx_PD_SEL0_MASK, 1 – from CHx_PA_INT_SEL<0>
10	0	CHx_PD_SEL0_INVERT	RW	Invert the SEL0 signal of CHx PD control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	CHx_PD_SEL0_MASK<8:0>	RW	GPIO mask for SEL0 signal of CHx PD control signals multiplexer.

+0x11		CHx_PD_SEL1 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	CHx_PD_SEL1_INTERNAL	RW	CHx PD control signals multiplexer SEL0 signal is generated 0 – from GPIO & CHx_PD_SEL0_MASK, 1 – from CHx_PA_INT_SEL<0>
10	0	CHx_PD_SEL1_INVERT	RW	Invert the SEL1 signal of CHx PD control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	CHx_PD_SEL1_MASK<8:0>	RW	GPIO mask for SEL1 signal of CHx PD control signals multiplexer.

+0x12		CHx_LNA_SEL0 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	CHx_LNA_SEL0_INTERNAL	RW	CHx LNA control signals multiplexer SEL0 signal is generated 0 – from GPIO & CHx_LNA_SEL0_MASK, 1 – from CHx_LNA_INT_SEL<0>
10	0	CHx_LNA_SEL0_INVERT	RW	Invert the SEL0 signal of CHx LNA control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	CHx_LNA_SEL0_MASK<8:0>	RW	GPIO mask for SEL0 signal of CHx LNA control signals multiplexer.

+0x13		CHx_LNA_SEL1 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	CHx_LNA_SEL1_INTERNAL	RW	CHx LNA control signals multiplexer SEL0 signal is generated 0 – from GPIO & CHx_LNA_SEL0_MASK, 1 – from CHx_LNA_INT_SEL<0>
10	0	CHx_LNA_SEL1_INVERT	RW	Invert the SEL1 signal of CHx LNA control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	CHx_LNA_SEL1_MASK<8:0>	RW	GPIO mask for SEL1 signal of CHx LNA control signals multiplexer.

+0x14		CHx_PA_SEL0 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	CHx_PA_SEL0_INTERNAL	RW	CHx PA control signals multiplexer SEL0 signal is generated 0 – from GPIO & CHx_PA_SEL0_MASK, 1 – from CHx_PA_INT_SEL<0>
10	0	CHx_PA_SEL0_INVERT	RW	Invert the SEL0 signal of CHx PA control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	CHx_PA_SEL0_MASK<8:0>	RW	GPIO mask for SEL0 signal of CHx PA control signals multiplexer.

+0x15		CHx_PA_SEL1 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	CHx_PA_SEL1_INTERNAL	RW	CHx PA control signals multiplexer SEL0 signal is generated 0 – from GPIO & CHx_PA_SEL0_MASK, 1 – from CHx_PA_INT_SEL<0>
10	0	CHx_PA_SEL1_INVERT	RW	Invert the SEL1 signal of CHx PA control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	CHx_PA_SEL1_MASK<8:0>	RW	GPIO mask for SEL0 signal of CHx PA control signals multiplexer.

+0x16		CHx_INT_SEL (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
5:4	00	CHx_PA_INT_SEL<1:0>	RW	Internal value of CHx PA control signals multiplexer selection signals.
3:2	00	CHx_LNA_INT_SEL<1:0>	RW	Internal value of CHx LNA control signals multiplexer selection signals.
1:0	00	CHx_PD_INT_SEL<1:0>	RW	Internal value of CHx PD control signals multiplexer selection signals.

+0x1D		CHx_PD_RB		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7		CHx_PA_R50_EN_RB	R	Readback the actual controlling value
6		CHx_PA_BYPASS_RB	R	Readback the actual controlling value
5		CHx_PA_PD_RB	R	Readback the actual controlling value
4		CHx_MIXB_LOBUFF_PD_RB	R	Readback the actual controlling value
3		/	R	Reserved
2		CHx_MIXA_LOBUFF_PD_RB	R	Readback the actual controlling value
1		/	R	Reserved
0		CHx_LNA_PD_RB	R	Readback the actual controlling value

+0x1E		CHx_LNA_CTRL_RB		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:11		CHx_LNA_ICT_LIN_RB<4:0>	R	Readback the actual controlling value
10:6		CHx_LNA_ICT_MAIN_RB<4:0>	R	Readback the actual controlling value
5:4		CHx_LNA_CGSCTRL_RB<1:0>	R	Readback the actual controlling value
3:0		CHx_LNA_GCTRL_RB<3:0>	R	Readback the actual controlling value

+0x1F		CHx_PA_CTRL_RB		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7:4		CHx_PA_LIN_LOSS_RB<3:0>	R	Readback the actual controlling value
3:0		CHx_PA_MAIN_LOSS_RB<3:0>	R	Readback the actual controlling value

9.4 Register bank HLMIXx

Note: This register bank is relevant only for LMS8001B.

Register bank HLMIX**x** (**x**=A, B, C, D) contains control registers for RF channel. Register addresses are given relative to base address. Base addresses are given in the following table.

Channel	Base Address
HLMIXA	0x2000
HLMIXB	0x2010
HLMIXC	0x2020
HLMIXD	0x2030

+0x0		HLMIXx_CONFIG0 (0x2043)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
13:7	1000000	HLMIXx_VGCAS0<6:0>	RW	HLMIXx LO bias voltage. HLMIXx_VGCAS<6:5> = 00 – Rb = 20 kΩ HLMIXx_VGCAS<6:5> = 01, 10 – Rb = 15 kΩ HLMIXx_VGCAS<6:5> = 11 – Rb = 10 kΩ Ib = Inom * HLMIXx_VGCAS<4:0>/16 Vb = VDD – Rb * Ib
6:2	10000	HLMIXx_ICT_BIAS0<4:0>	RW	HLMIXx core bias current control. Ibias = Inom * HLMIXx_ICT_BIAS/16
1	1	HLMIXx_BIAS_PD0	RW	HLMIXx core bias control 0 – HLMIXx core is biased, 1 – HLMIXx core is powered down.
0	1	HLMIXx_LOBUF_PD0	RW	HLMIXx LO buffer power down 0 – LO buffer is enabled, 1 – LO buffer is powered down.

+0x1		HLMIXx_CONFIG1 (0x2043)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
13:7	1000000	HLMIXx_VGCAS1<6:0>	RW	HLMIXx LO bias voltage. HLMIXx_VGCAS<6:5> = 00 – Rb = 20 kΩ HLMIXx_VGCAS<6:5> = 01, 10 – Rb = 15 kΩ HLMIXx_VGCAS<6:5> = 11 – Rb = 10 kΩ Ib = Inom * HLMIXx_VGCAS<4:0>/16 Vb = VDD – Rb * Ib
6:2	10000	HLMIXx_ICT_BIAS1<4:0>	RW	HLMIXx core bias current control. Ibias = Inom * HLMIXx_ICT_BIAS/16
1	1	HLMIXx_BIAS_PD1	RW	HLMIXx core bias control 0 – HLMIXx core is biased, 1 – HLMIXx core is powered down.
0	1	HLMIXx_LOBUF_PD1	RW	HLMIXx LO buffer power down 0 – LO buffer is enabled, 1 – LO buffer is powered down.

+0x2		HLMIXx_CONFIG2 (0x2043)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
13:7	1000000	HLMIXx_VGCAS2<6:0>	RW	HLMIXx LO bias voltage. HLMIXx_VGCAS<6:5> = 00 – Rb = 20 kΩ HLMIXx_VGCAS<6:5> = 01, 10 – Rb = 15 kΩ HLMIXx_VGCAS<6:5> = 11 – Rb = 10 kΩ Ib = Inom * HLMIXx_VGCAS<4:0>/16 Vb = VDD – Rb * Ib
6:2	10000	HLMIXx_ICT_BIAS2<4:0>	RW	HLMIXx core bias current control. Ibias = Inom * HLMIXx_ICT_BIAS/16
1	1	HLMIXx_BIAS_PD2	RW	HLMIXx core bias control 0 – HLMIXx core is biased, 1 – HLMIXx core is powered down.
0	1	HLMIXx_LOBUF_PD2	RW	HLMIXx LO buffer power down 0 – LO buffer is enabled, 1 – LO buffer is powered down.

+0x3		HLMIXx_CONFIG3 (0x2043)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
13:7	1000000	HLMIXx_VGCAS3<6:0>	RW	HLMIXx LO bias voltage. HLMIXx_VGCAS<6:5> = 00 – Rb = 20 kΩ HLMIXx_VGCAS<6:5> = 01, 10 – Rb = 15 kΩ HLMIXx_VGCAS<6:5> = 11 – Rb = 10 kΩ Ib = Inom * HLMIXx_VGCAS<4:0>/16 Vb = VDD – Rb * Ib
6:2	10000	HLMIXx_ICT_BIAS3<4:0>	RW	HLMIXx core bias current control. Ibias = Inom * HLMIXx_ICT_BIAS/16
1	1	HLMIXx_BIAS_PD3	RW	HLMIXx core bias control 0 – HLMIXx core is biased, 1 – HLMIXx core is powered down.
0	1	HLMIXx_LOBUF_PD3	RW	HLMIXx LO buffer power down 0 – LO buffer is enabled, 1 – LO buffer is powered down.

+0x4		HLMIXx_LOSS0 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
5:2	0000	HLMIXx_MIXLOSS0<3:0>	RW	Pout = Pout_max – Loss
1:0	00	/	RW	Reserved

+0x5		HLMIXx_LOSS1 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
5:2	0000	HLMIXx_MIXLOSS1<3:0>	RW	Pout = Pout_max – Loss
1:0	00	/	RW	Reserved

+0x6		HLMIXx_LOSS2 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
5:2	0000	HLMIXx_MIXLOSS2<3:0>	RW	Pout = Pout_max – Loss
1:0	00	/	RW	Reserved

+0x7		HLMIXx_LOSS3 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
5:2	0000	HLMIXx_MIXLOSS3<3:0>	RW	Pout = Pout_max – Loss
1:0	00	/	RW	Reserved

+0x8		HLMIXx_CONF_SEL0 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	HLMIXx_CONF_SEL0_INTERNAL	RW	HLMIXx bias control signals multiplexer SEL0 signal is generated 0 – from GPIO & HLMIXx_CONF_SEL0_MASK, 1 – from HLMIXx_CONF_INT_SEL<0>
10	0	HLMIXx_CONF_SEL0_INVERT	RW	Invert the SEL0 signal of HLMIXx bias control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	HLMIXx_CONF_SEL0_MASK<8:0>	RW	GPIO mask for SEL0 signal of HLMIXx bias control signals multiplexer.

+0x9		HLMIXx_CONF_SEL1_INTERNAL (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	HLMIXx_CONF_SEL1_INTERNAL	RW	HLMIXx bias control signals multiplexer SEL1 signal is generated 0 – from GPIO & HLMIXx_CONF_SEL1_MASK, 1 – from HLMIXx_CONF_INT_SEL<1>
10	0	HLMIXx_CONF_SEL1_INVERT	RW	Invert the SEL1 signal of HLMIXx bias control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	HLMIXx_CONF_SEL1_MASK<8:0>	RW	GPIO mask for SEL1 signal of HLMIXx bias control signals multiplexer.

+0xA		HLMIXx_LOSS_SEL0 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	HLMIXx_LOSS_SEL0_INTERNAL	RW	HLMIXx loss control signals multiplexer SEL0 signal is generated from 0 – from GPIO & HLMIXx_LOSS_SEL0_MASK, 1 – from HLMIXx_LOSS_INT_SEL<0>
10	0	HLMIXx_LOSS_SEL0_INVERT	RW	Invert the SEL0 signal of HLMIXx loss control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	HLMIXx_LOSS_SEL0_MASK<8:0>	RW	GPIO mask for SEL0 signal of HLMIXx loss control signals multiplexer.

+0xB		HLMIXx_LOSS_SEL1 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	HLMIXx_LOSS_SEL1_INTERNAL	RW	HLMIXx loss control signals multiplexer SEL0 signal is generated from 0 – from GPIO & HLMIXx_LOSS_SEL0_MASK, 1 – from HLMIXx_LOSS_INT_SEL<0>
10	0	HLMIXx_LOSS_SEL1_INVERT	RW	Invert the SEL0 signal of HLMIXx loss control signals. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	HLMIXx_LOSS_SEL1_MASK<8:0>	RW	GPIO mask for SEL1 signal of HLMIXx loss control signals multiplexer.

+0xC		HLMIXx_INT_SEL (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
3:2	00	HLMIXx_LOSS_INT_SEL<1:0>	RW	Internal value of HLMIXx loss control signals multiplexer selection signals.
1:0	00	HLMIXx_CONF_INT_SEL<1:0>	RW	Internal value of HLMIXx control signals multiplexer selection signals.

+0xE		HLMIXx_CONFIG_RB		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
13:7		HLMIXx_VGCAS_RB<6:0>	R	Readback the actual controlling value
6:2		HLMIXx_ICT_BIAS_RB<4:0>	R	Readback the actual controlling value
1		HLMIXx_BIAS_PD_RB	R	Readback the actual controlling value
0		HLMIXx_LOBUF_PD_RB	R	Readback the actual controlling value

+0xF		HLMIXx_LOSS_RB		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
5:2		HLMIXx_MIXLOSS_RB<3:0>	R	Readback the actual controlling value
1:0		/	R	Reserved

9.5 Register bank PLL_CONFIGURATION (0x4000 – 0x401F)

+0x4000		PLL_VREG (0x0210)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	0	EN_VCOBIAS	RW	Enables VCO LDO and Bias Circuits 0 – Powered down (default) 1 – Enabled
10	0	BYP_VCOREG	RW	Bypasses VCO LDO 0 – LDO Active (default) 1 – LDO Bypassed
9	1	CURLIM_VCOREG	RW	Enables output current limitation in the VCO LDO 0 – Current limit disabled 1 – Current limit enabled (default)
8	0	SPDUP_VCOREG	RW	Shorts the noise filter resistor in the VCO LDO for fast settling time. It should be connected to 1 us pulse. 0 – Noise filter resistor in place(default) 1 – Noise filter resistor bypassed
7:0	00010000	VDIV_VCOREG<7:0>	RW	Controls the VCO LDO Output Voltage. VOUT=1.8 V x (257-RDIV)/(265-RDIV) Default: 00010000(32) VOUT=1.7 V

+0x4001		PLL_CFG_XBUF (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
2	0	PLL_XBUF_SLFBEN	RW	Enables self-biasing at the input of the XBUF 0 – Self-biasing is disabled. Input signal needs to be DC-coupled (default) 1 – Self-biasing is enabled. Input signals needs to be AC-coupled
1	0	PLL_XBUF_BYPEN	RW	Shorts the input stage buffer in XBUF. The final 1.2 V stages are active. In bypass-mode input should be 1.2 V level full-scale CMOS signal. 0 – Bypass not active 1 – Bypass active

+0x4001		PLL_CFG_XBUF (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
0	0	PLL_XBUF_EN	RW	Enables XBUF 0 – Powered down (default) 1 – Enabled

+0x4002		PLL_CAL_AUTO0 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
12	0	FCAL_START	RW	Starts the automatic VCO frequency calibration algorithm (sticky-bit). Writing 1 starts the calibration, automatically cleared when calibration is finished.
11	0	VCO_SEL_FINAL_VAL	R	Valid bit for VCO_SEL_FINAL<1:0> result of automatic VCO frequency calibration process 0 – Data not valid 1 – Data valid
10:9	00	VCO_SEL_FINAL<1:0>	R	Defines the optimal VCO core for synthesizing the targeted LO frequency.
8	0	FREQ_FINAL_VAL	R	Valid bit for VCO_FREQ_FINAL<7:0>result of automatic VCO frequency calibration process 0 – Data not valid 1 – Data valid
7:0	00000000	FREQ_FINAL<7:0>	R	Defines the optimal cap bank configuration of the active LC-VCO core for synthesizing the targeted LO frequency.

+0x4003		PLL_CAL_AUTO1 (0x1700)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
13	0	VCO_SEL_FORCE	RW	Forces the user-defined VCO_SEL_INIT<1:0> word to select the active LC-VCO core and skips the VCO auto-select process during automatic VCO frequency calibration 0 – Runs VCO auto-select process (default) 1 – Skips VCO auto-select process and forces VCO_SEL_INIT<1:0>
12:11	10	VCO_SEL_INIT<1:0>	RW	Defines active LC-VCO core when skipping the VCO auto-select process
10:8	111	FREQ_INIT_POS<2:0>	RW	Defines the starting bit-position for optimal cap-bank binary search process for the active LC-VCO core. 111(7) – Algorithm starts from MSB and determines the complete word (default). 110(6) – Algorithm determines bits from (MSB-1)-th position down to LSB. … 000(0) – Algorithm determines only the LSB bit of the cap-bank code.
7:0	00000000	FREQ_INIT<7:0>	RW	Initial cap bank configuration for binary search process during the automatic VCO frequency calibration. The MSB bits that will not be determined by the state-machine will be taken from this word.

+0x4004		PLL_CAL_AUTO2 (0x0440)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11:8	0100	FREQ_SETTLING_N<3:0>	RW	VCO oscillation frequency settling-time during auto-calibration after updating cap bank configuration. Expressed as the number of reference clock cycles. Default: 0100(4), 100ns for 40 MHz reference frequency
7:0	01000000	VTUNE_WAIT_N<7:0>	RW	VCO tuning-voltage settling time at the beginning of the auto-calibration after opening the PLL loop. Expressed as the number of reference clock cycles. Default: 01000000(64), 1.6us for 40 MHz reference frequency

+0x4005		PLL_CAL_AUTO3 (0xFA05)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:8	11111010	VCO_SEL_FREQ_MAX<7:0>	RW	High-frequency cap-bank configuration used during VCO auto-select process Default: 11111010 (250)
7:0	00000101	VCO_SEL_FREQ_MIN<7:0>	RW	Low-frequency cap-bank configuration used during VCO auto-select process Default: 00000101 (5)

+0x4006		PLL_CAL_MAN (0x8080)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:8	10000000	VCO_FREQ_MAN<7:0>	RW	Cap Bank configuration multiplexed to the input of active VCO core when manual VCO calibration mode is selected. Default: 10000000 (128)
7:6	10	VCO_SEL_MAN<1:0>	RW	VCO select word for choosing the active VCO core when manual VCO calibration mode is selected. Default: 10 (2)
5	0	FREQ_HIGH	R	VCO Frequency Estimator Output 1 – VCO Oscillation Frequency higher than targeted frequency 0 – VCO Oscillation Frequency not-higher than targeted frequency
4	0	FREQ_EQUAL	R	VCO Frequency Estimator Output 1 – VCO Oscillation Frequency equal to the targeted frequency 0 – VCO Oscillation Frequency not-equal to the targeted frequency
3	0	FREQ_LOW	R	VCO Frequency Estimator Output 1 – VCO Oscillation Frequency lower than targeted frequency 0 – VCO Oscillation Frequency not-lower than targeted frequency
2	0	CTUNE_STEP_DONE	R	VCO Frequency Estimator Output Signalizes the end of VCO coarse-tuning step. 1 – Step Done, FREQ_HIGH, FREQ_EQUAL, FREQ_LOW outputs are valid
1	0	CTUNE_START	RW	Starts the process of estimating the VCO oscillation frequency. 1 – Starts coarse-tuning step 0 – Use to reset the VCO Frequency Estimator Circuit before starting the new step of VCO coarse tuning.
0	0	CTUNE_EN	RW	Enables the VCO Frequency Estimator Circuit 0 – Powered down (default) 1 – Enabled

+0x4008		PLL_CFG_SEL0 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	PLL_CFG_SEL0_INTERNAL	RW	PLL profile multiplexer SEL0 signal is generated from 0 – from GPIO & PLL_CFG_SEL0_MASK, 1 – from PLL_CFG_INT_SEL<0>
10	0	PLL_CFG_SEL0_INVERT	RW	Invert the SEL0 signal of PLL profile multiplexer. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	PLL_CFG_SEL0_MASK<8:0>	RW	GPIO mask for SEL0 signal of PLL profile multiplexer.

+0x4009		PLL_CFG_SEL1 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	PLL_CFG_SEL1_INTERNAL	RW	PLL profile multiplexer SEL1 signal is generated from 0 – from GPIO & PLL_CFG_SEL1_MASK, 1 – from PLL_CFG_INT_SEL<1>

+0x4009		PLL_CFG_SEL1 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	PLL_CFG_SEL1_INVERT	RW	Invert the SEL1 signal of PLL profile multiplexer. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	PLL_CFG_SEL1_MASK<8:0>	RW	GPIO mask for SEL1 signal of PLL profile multiplexer.

+0x400A		PLL_CFG_SEL2 (0x0800)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
11	1	PLL_CFG_SEL2_INTERNAL	RW	PLL profile multiplexer SEL2 signal is generated from 0 – from GPIO & PLL_CFG_SEL2_MASK, 1 – from PLL_CFG_INT_SEL<2>
10	0	PLL_CFG_SEL2_INVERT	RW	Invert the SEL2 signal of PLL profile multiplexer. 0 – No inversion, 1 – Signal is inverted.
8:0	000000000	PLL_CFG_SEL2_MASK<8:0>	RW	GPIO mask for SEL2 signal of PLL profile multiplexer.

+0x400B		PLL_CFG (0x0080)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
9	0	PLL_RSTN	RW	PLL reset, active low.
8:7	01	CTUNE_RES<1:0>	RW	Automatic VCO Frequency Calibration Resolution. Default: 01 (2) 11 – Highest Accuracy, Lowest Speed 00 – Lowest Accuracy, Highest Speed
6	0	PLL_CALIBRATION_MODE	RW	PLL calibration mode. 0 – Automatic calibration (default) 1 – Manual calibration
5	0	PLL_CALIBRATION_EN	RW	Activate PLL calibration. 0 – Normal mode (default) 1 – Calibration mode
4	0	PLL_FLOCK_INTERNAL	RW	Fast lock control. 0 – Normal operation. Fast lock select signal comes from fast lock state machine. (default) 1 – Debug mode. Fast lock select signal is forced by PLL_FLOCK_INTVAL
3	0	PLL_FLOCK_INTVAL	RW	Fast lock control internal select value.
2:0	000	PLL_CFG_INT_SEL<2:0>	RW	Internal PLL profile control.

+0x400C		PLL_CFG_STATUS (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
2	0	VTUNE_HIGH	R	VCO Tuning voltage high. 1 – VCO Tuning Voltage Above Recommended Upper Limit
1	0	VTUNE_LOW	R	VCO Tuning voltage low. 1 – VCO Tuning Voltage Bellow Recommended Lower Limit
0	0	PLL_LOCK	R	PLL lock detect. 1 – PLL Locked

+0x400E		PLL_LODIST_CFG1 (0x0210)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
10	0	SEL_BIAS_CORE	RW	Selects the bias for the core of LO Distribution Network. 0 – PVT Compensated Bias (default) 1 – IP20FRP Bias
9:5	10000	PLL_LODIST_ICT_CORE<4:0>	RW	Controls the IP20FRP bias current value when SEL_BIAS_CORE=1. Ibias = Inom * PLL_LODIST_ICT_CORE / 16

+0x400E		PLL_LODIST_CFG1 (0x0210)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
4:0	10000	PLL_LODIST_ICT_BUF<4:0>	RW	Controls the input bias current value for LO Distribution Network output buffers. Ibias = Inom * PLL_LODIST_ICT_BUF / 16

+0x400F		PLL_LODIST_CFG2 (0x00AA)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7:6	10	PLL_ICT_OUT3<1:0>	RW	Controls the current drive-strength of the LO Distribution Network Output Buffer Stage for CHD. Iout=Inom*(1+PLL_ICT_OUT3/4)
5:4	10	PLL_ICT_OUT2<1:0>	RW	Controls the current drive-strength of the LO Distribution Network Output Buffer Stage for CHC. Iout=Inom*(1+PLL_ICT_OUT2/4)
3:2	10	PLL_ICT_OUT1<1:0>	RW	Controls the current drive-strength of the LO Distribution Network Output Buffer Stage for CHB. Iout=Inom*(1+PLL_ICT_OUT1/4)
1:0	10	PLL_ICT_OUT0<1:0>	RW	Controls the current drive-strength of the LO Distribution Network Output Buffer Stage for CHA. Iout=Inom*(1+PLL_ICT_OUT0/4)

+0x4010		PLL_SDM_BIST1 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:9	0000000	BSIGL<6:0>	R	BIST signature. Read only.
8	0	BSTATE	R	BIST state indicator 0 – BIST not running (default) 1 – BIST running
4	0	EN_SDM_TSTO	RW	Enable test buffer output
1	0	BEN	RW	Enable BIST
0	0	BSTART	RW	Starts BIST

+0x4011		PLL_SDM_BIST2 (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:0	0000000000000000	BSIGH<15:0>	R	PLL_SDM_BIST Output

9.6 Register bank PLL_PROFILE_n

Register bank PLL_PROFILE_**n** (**n**=0..7) contains control registers for PLL profile. Register addresses are given relative to channel base address. Profile base addresses are given in the following table.

Profile	Base Address
PLL_PROFILE_0	0x4100
PLL_PROFILE_1	0x4110
PLL_PROFILE_2	0x4120
PLL_PROFILE_3	0x4130
PLL_PROFILE_4	0x4140
PLL_PROFILE_5	0x4150
PLL_PROFILE_6	0x4160
PLL_PROFILE_7	0x4170

+0x0		PLL_ENABLE_n (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
12	0	PLL_LODIST_EN_BIAS_n	RW	Enable for LO distribution bias.
11	0	PLL_LODIST_EN_DIV2IQ_n	RW	Enable for IQ generator in LO distribution. 0 – Clock is not divided by 2 1 – Clock is divided by 2, I and Q are generated
10	0	PLL_EN_VTUNE_COMP_n	RW	Enable for tuning voltage comparator in PLL.
9	0	PLL_EN_LD_n	RW	Lock detector enable.
8	0	PLL_EN_PFD_n	RW	Enable for PFD in PLL.
7	0	PLL_EN_CP_n	RW	Enable for charge pump in PLL.
6	0	PLL_EN_CPOFS_n	RW	Enable for offset (bleeding) current in charge pump.
5	0	PLL_EN_VCO_n	RW	Enable for VCO.
4	0	PLL_EN_FFDIV_n	RW	Enable for feed-forward divider in PLL. 0 – Output clock is not divided
3	0	PLL_EN_FB_PDIV2_n	RW	Enable for feedback pre-divider. 0 – Output clock is directly fed to feedback divider
2	0	PLL_EN_FFCORE_n	RW	Enable for feed-forward divider core
1	0	PLL_EN_FBDIV_n	RW	Enable for feedback divider core
0	0	PLL_SDM_CLK_EN_n	RW	Enable for sigma-delta modulator

+0x1		PLL_LPF_CFG1_n (0x1188)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:12	0001	R3_n<3:0>	RW	Control word for loop filter. R3_val =14.9 kΩ/R3<3:0> When fast lock mode is enabled, this is the final value.
11:8	0001	R2_n<3:0>	RW	Control word for loop filter. R2_val = 24.6 kΩ/R2<3:0> When fast lock mode is enabled, this is the final value.
7:4	1000	C2_n<3:0>	RW	Control word for C2 in PLL loop filter. C2_val = 150 pF+10 pF * C2<3:0> When fast lock mode is enabled, this is the final value.
3:0	1000	C1_n<3:0>	RW	Control word for C1 in PLL loop filter. C1_val = 1.2 pF*C1<3:0> When fast lock mode is enabled, this is the final value.

+0x2		PLL_LPF_CFG2_n (0x0028)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
6:5	01	VTUNE_VCT_n<1:0>	RW	Tuning voltage control word during coarse tuning (LPFSW=1). 00 – 300 mV, 01 – 600 mV, 10 – 750 mV, 11 – 900 mV.
4	0	LPFSW_n	RW	Loop filter control. 0 – PLL loop is closed, 1 – PLL loop is open and tuning voltage is set to value specified by VTUNE_VCT<1:0>. When LFPSW=1 PLL is in open-loop configuration for coarse tuning.
3:0	1000	C3_n<3:0>	RW	Control word for C3 in PLL loop filter. C3_val = 5 pF + 1.2 pF * C3<3:0> When fast lock mode is enabled, this is the final value.

+0x3		PLL_CP_CFG0_n (0x0100)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
14	0	FLIP_n	RW	Flip for PFD inputs 0 – Normal operation, 1 – Inputs are interchanged
13:12	00	DEL_n<1:0>	RW	Reset path delay
11:6	000100	PULSE_n<5:0>	RW	Charge pump pulse current I = 25 µA * PULSE<5:0>
5:0	000000	OFS_n<5:0>	RW	Charge pump offset (bleeding) current I = 6.25 µA * OFS<5:0>

+0x4		PLL_CP_CFG1_n (0x0050)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
6:5	10	LD_VCT_n<1:0>	RW	Threshold voltage for lock detector 00 – 600 mV, 01 – 700 mV, 10 – 800 mV, 11 – 900 mV.
4:0	10000	ICT_CP_n<4:0>	RW	Charge pump bias current. ICP_BIAS = ICP_BIAS_NOM * ICT_CP<4:0>/16

+0x5		PLL_VCO_FREQ_n (0x0080)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
7:0	10000000	VCO_FREQ_n<7:0>	RW	VCO cap bank code. 00000000 – lowest frequency 11111111 – highest frequency

+0x6		PLL_VCO_CFG_n (0x0D81)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
12	0	SPDUP_VCO_n	RW	Speed-up VCO core by bypassing the noise filter
11	1	VCO_AAC_EN_n	RW	Enable for automatic VCO amplitude control.
10:9	10	VDIV_SWVDD_n<1:0>	RW	Capacitor bank switches bias voltage 00 – 600 mV, 01 – 800 mV, 10 – 1000 mV, 11 – 1200 mV.
8:7	11	VCO_SEL_n<1:0>	RW	VCO core selection 00 – External VCO, 01 – Low-frequency band VCO (4 – 6 GHz), 10 – Mid-frequency band VCO (6 – 8 GHz), 11 – High-frequency band VCO (8 – 10 GHz).
6:0	0000001	VCO_AMP_n<6:0>	RW	VCO amplitude control word. 0000000 – minimum amplitude Lowest two bits control the VCO core current. Other bits are used for fine amplitude control, automatically determined when VCO_AAC_EN=1

+0x7		PLL_FF_CFG_n (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
4	0	FFDIV_SEL_n	RW	Feed-forward divider multiplexer select bit 0 – No division, 1 – Input frequency is divided
3:2	00	FFCORE_MOD_n<1:0>	RW	Feed-forward divider core modulus 00 – No division 01 – Div by 2 10 – Div by 4 11 – Div by 8
1:0	00	FF_MOD_n<1:0>	RW	Multiplexer for divider outputs. In normal operation FF_MOD should be equal to FFCORE_MOD.

+0x8		PLL_SDM_CFG_n (0x00D8)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
14	0	INTMOD_EN_n	RW	Integer mode enable
13	0	DITHER_EN_n	RW	Enable dithering in SDM 0 – Disabled 1 – Enabled
12	0	SEL_SDMCLK_n	RW	Selects between the feedback divider output and FREF for SDM 0 – CLK CLK_DIV 1 – CLK CLK_REF
11	0	REV_SDMCLK_n	RW	Reverses the SDM clock 0 – Normal 1 – Reversed (after INV)
9:0	0011011000	INTMOD_n<9:0>	RW	Integer section of division ratio.

+0x9		PLL_FRACMODL_n (0x5730)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:0	0101011100110000	FRACMODL_n<15:0>	RW	Fractional control of the division ratio LSB

+0xA		PLL_FRACMODH_n (0x0005)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
3:0	0101	FRACMODH_n<3:0>	RW	Fractional control of the division ratio MSB

+0xB		PLL_LODIST_CFG_n (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:12	0000	PLL_LODIST_EN_OUT_n<3:0>	RW	LO distribution enable signals. Each bit is an enable for individual channel.
11:9	000	PLL_LODIST_FSP_OUT3_n<2:0>	RW	LO distribution channel D frequency, sign and phase control. FSP_OUT<2> - Frequency division control 0 – LO is divided by 2, 1 – LO is not divided. FSP_OUT<1> - LO sign 0 – LO is not inverted 1 – LO is inverted FSP_OUT<0> - LO phase 0 – LO phase 0 deg (I) 1 – LO phase 90 deg (Q)
8:6	000	PLL_LODIST_FSP_OUT2_n<2:0>	RW	LO distribution channel C frequency, sign and phase control. FSP_OUT<2> - Frequency division control 0 – LO is divided by 2, 1 – LO is not divided. FSP_OUT<1> - LO sign 0 – LO is not inverted 1 – LO is inverted FSP_OUT<0> - LO phase 0 – LO phase 0 deg (I) 1 – LO phase 90 deg (Q)
5:3	000	PLL_LODIST_FSP_OUT1_n<2:0>	RW	LO distribution channel B frequency, sign and phase control. FSP_OUT<2> - Frequency division control 0 – LO is divided by 2, 1 – LO is not divided. FSP_OUT<1> - LO sign 0 – LO is not inverted 1 – LO is inverted FSP_OUT<0> - LO phase 0 – LO phase 0 deg (I) 1 – LO phase 90 deg (Q)

+0xB		PLL_LODIST_CFG_n (0x0000)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
2:0	000	PLL_LODIST_FSP_OUT0_n<2:0>	RW	LO distribution channel A frequency, sign and phase control. FSP_OUT<2> - Frequency division control 0 – LO is divided by 2, 1 – LO is not divided. FSP_OUT<1> - LO sign 0 – LO is not inverted 1 – LO is inverted FSP_OUT<0> - LO phase 0 – LO phase 0 deg (I) 1 – LO phase 90 deg (Q)

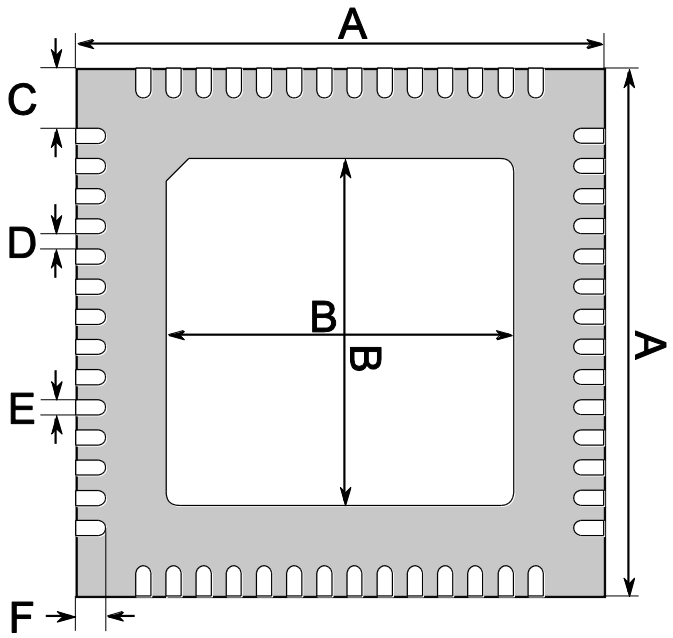
+0xC		PLL_FLOCK_CFG1_n (0x4488)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:12	0100	FLOCK_R3_n<3:0>	RW	Loop filter R3 used during fact lock.
11:8	0100	FLOCK_R2_n<3:0>	RW	Loop filter R2 used during fast lock.
7:4	1000	FLOCK_C2_n<3:0>	RW	Loop filter C2 used during fast lock.
3:0	1000	FLOCK_C1_n<3:0>	RW	Loop filter C1 used during fast lock.

+0xD		PLL_FLOCK_CFG2_n (0x8FC0)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
15:12	1000	FLOCK_C3_n<3:0>	RW	Loop filter C3 used during fast lock.
11:6	111111	FLOCK_PULSE_n<5:0>	RW	Charge pump pulse current used during fast lock.
5:0	000000	FLOCK_OFS_n<5:0>	RW	Charge pump offset (bleeding) current used during fast lock.

+0xE		PLL_FLOCK_CFG3_n (0x0190)		
Bit	**Default**	**Bitfield Name**	**Mode**	**Description**
14:11	0000	FLOCK_LODIST_EN_OUT_n<3:0>	RW	LO distribution enable signals used during fast lock
10	0	FLOCK_VCO_SPDUP_n	RW	VCO speedup used during fast lock
9:0	0110010000	FLOCK_N_n<9:0>	RW	Duration of fast lock in PLL reference frequency clock cycles.

10

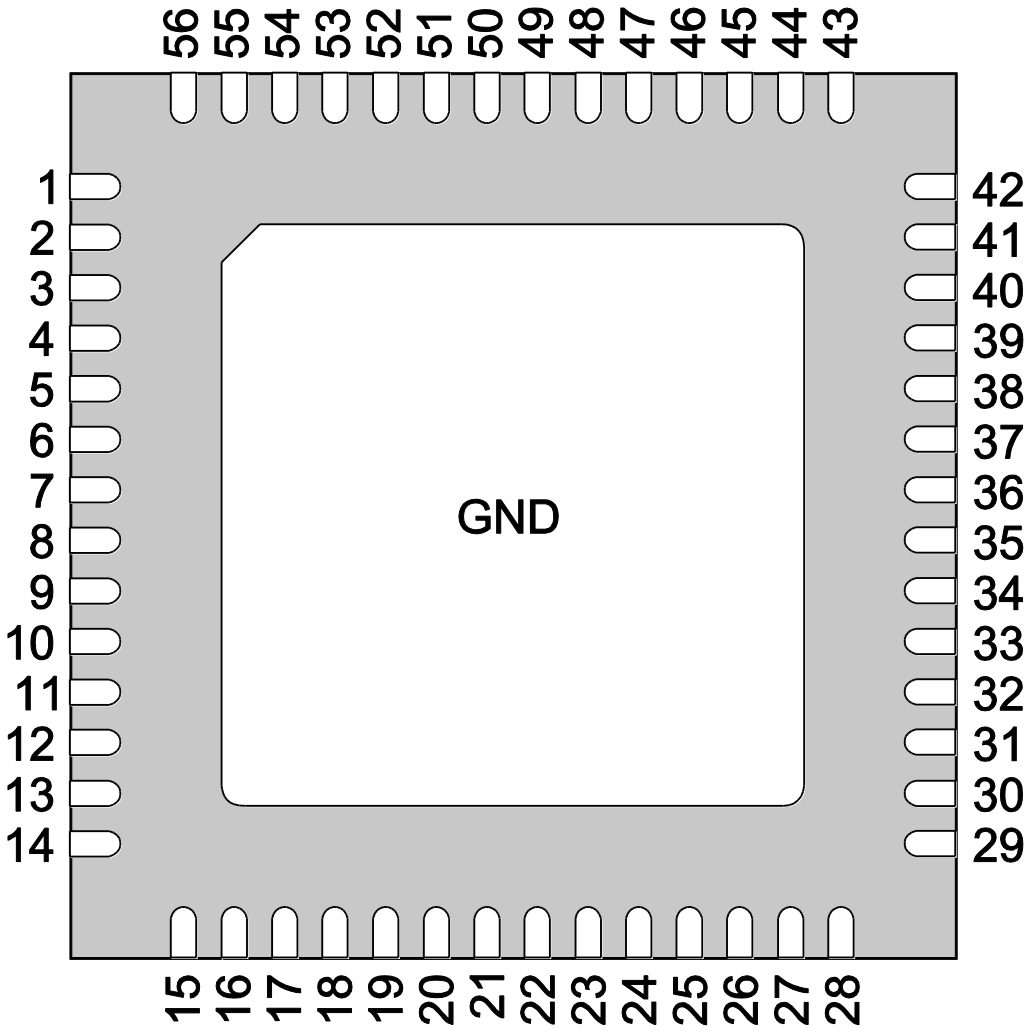
LMS8001 Package Drawing



A	7.0 mm
B	4.6 mm
C	0.8 mm
D	0.2 mm
E	0.2 mm
F	0.4 mm

11

LMS8001 Pinout



Pin	Name	Description
GND	GND	Ground. Connect paddle to good ground with as many as possible vias to ensure proper RF grounding.
1	NC	Not connected
2	EXTLO_P	External LO +
3	VDD18_BIAS	1.8 V supply voltage for bias.
4	HFINA_N HFINA_HLMIX_P	Channel A RF input -
5	HFINA_P HFINA_HLMIX_N	Channel A RF input +
6	VDD_HFINA	Normal bonding: Supply voltage for Channel A LNA. HLMIX bonding: Output of 120 mA LDO.
7	VDD_HFINAB	1.8 V supply voltage for 2x 120 mA LDO.
8	VDD_HFINB	Normal bonding: Supply voltage for Channel B LNA. HLMIX bonding: Output of 120 mA LDO.
9	HFINB_P HFINB_HLMIX_N	Channel B RF input +
10	HFINB_N HFINB_HLMIX_P	Channel B RF input -
11	SPI_SEN	SPI SEN
12	SPI_SDIO	SPI SDIO
13	SPI_SCLK	SPI SCLK
14	SPI_SDO	SPI SDO
15	GPIO8	GPIO 8 pad
16	VDD_DIG_IO	Supply voltage for digital IO output drivers
17	VDD12_DIG_CORE	Supply voltage for digital core. Generated by on-chip LDO.
18	RESETN	Chip reset. Active low.
19	HFOUTB_P HFOUTB_HLMIX_N	Channel B RF output +
20	HFOUTB_N HFOUTB_HLMIX_P	Channel B RF output -
21	VDD12_LOBUFB	Supply voltage for channel B LO buffer. Generated by on-chip LDO.
22	VDD18_LOBUFBC	Supply voltage for channel B & C LO buffer LDOs.
23	VDD12_LOBUFC	Supply voltage for channel C LO buffer. Generated by on-chip LDO.
24	HFOUTC_N HFOUTC_HLMIX_P	Channel C RF output -
25	HFOUTC_P HFOUTC_HLMIX_N	Channel C RF output +
26	GPIO0	GPIO 0
27	GPIO1	GPIO 1
28	GPIO3	GPIO 3
29	GPIO2	GPIO 2
30	GPIO5	GPIO5
31	GPIO4	GPIO4
32	GPIO6	GPIO6
33	GPIO7	GPIO7
34	HFINC_N HFINC_HLMIX_P	Channel C RF input -
35	HFINC_P HFINC_HLMIX_N	Channel C RF input +
36	VDD_HFINC	Normal bonding: Supply voltage for Channel C LNA. HLMIX bonding: Output of 120 mA LDO.
37	VDD_HFIN_CD	1.8 V supply for 2 x 120 mA LDOs.

Pin	Name	Description
38	VDD_HFIND	Normal bonding: Supply voltage for Channel D LNA. HLMIX bonding: Output of 120 mA LDO.
39	HFIND_P HFIND_HLMIX_N	Channel D RF input +
40	HFIND_N HFIND_HLMIX_P	Channel D RF input -
41	VDD18_VCO	1.8 V supply for VCO.
42	VDD18_PLL	1.8 V supply for three on-chip PLL LDOs.
43	VDD12_PLL_CP	1.2 V supply for PLL charge pump. Generated by on-chip LDO.
44	VDD12_PLL_DIV	1.2 V supply for PLL divider. Generated by on-chip LDO.
45	VDD12_PLL_CLK_BUF	1.2 V supply for LO distribution. Generated by on-chip LDO.
46	HFOUTD_P HFOUTD_HLMIX_N	Channel D RF output +
47	HFOUTD_N HFOUTD_HLMIX_P	Channel D RF output -
48	VDD12_LOBUFD	Supply voltage for channel D LO buffer. Generated by on-chip LDO.
49	VDD18_LOBUFAD	Supply voltage for channel A & D LO LDOs.
50	VDD12_LOBUFA	Supply voltage for channel A LO buffer. Generated by on-chip LDO.
51	HFOUTA_N HFOUTA_HLMIX_P	Channel A RF output -
52	HFOUTA_P HFOUTA_HLMIX_N	Channel A RF output +
53	PLLCLK	PLL reference clock.
54	REXT	External 10 kΩ resistor.
55	GLOBAL_PD	Global power down.
56	EXTLO_N	External LO -