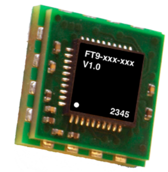
FT9-TFC
GNSS 1PPS Time to Clock Output Synchronizer

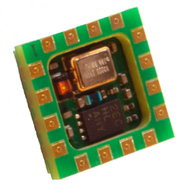
CONNOR WINFIELD

2111 Comprehensive Drive
Aurora, Illinois 60505
Phone: 630-851-4722
Email: sales@conwin.com
www.conwin.com

DATA SHEET

Overview





The FT9-TFC is a highly integrated time and frequency synchronizing module designed to receive a 1PPS reference input and generate a 1PPS output and up to two single ended clock outputs phase locked and aligned to the rising edge of the 1PPS output pulse. This high precision phase and frequency synchronization solution integrates low phase noise frequency clock translation. An external 10MHz precision OCXO or TCXO provides the system's master clock for various holdover performance options as well as the support for multiple filter bandwidth options available from <1mHz to .1 Hz. A disciplined internal VCXO provides the output characteristics for phase noise and jitter performance for the two single ended clock outputs with ultra-low jitter performance. The internal analog PLL is pre-configured in the FT9-TFC along with default NPLL settings with configuration options for different master clock (MCLK) performance types and loop bandwidth settings. The FT9-TFC has two 1PPS input pins allowing the user to switch between two 1PPS sources. The FT9-TFC allows the user access to the chip's internal phase detector to calibrate and correct for sawtooth error typically found on a 1PPS signal emanating from a GNSS receiver. The FT9-TFC also provides the user access to the internal master clock calibration feature used to calibrate and correct for calibration offsets and thermal drift found in the external supporting OCXO. Using pre-programmed MCLK OCXO modules, the FT9-TFC can automatically compensate the OCXO module's thermal drift without the need for an external micro controller.

This product can be used to support a high-stability frequency reference for use in wireless systems, IEEE 1588v2, and applications employing a 1PPS frequency source for high precision, long term time and frequency generation.

Features

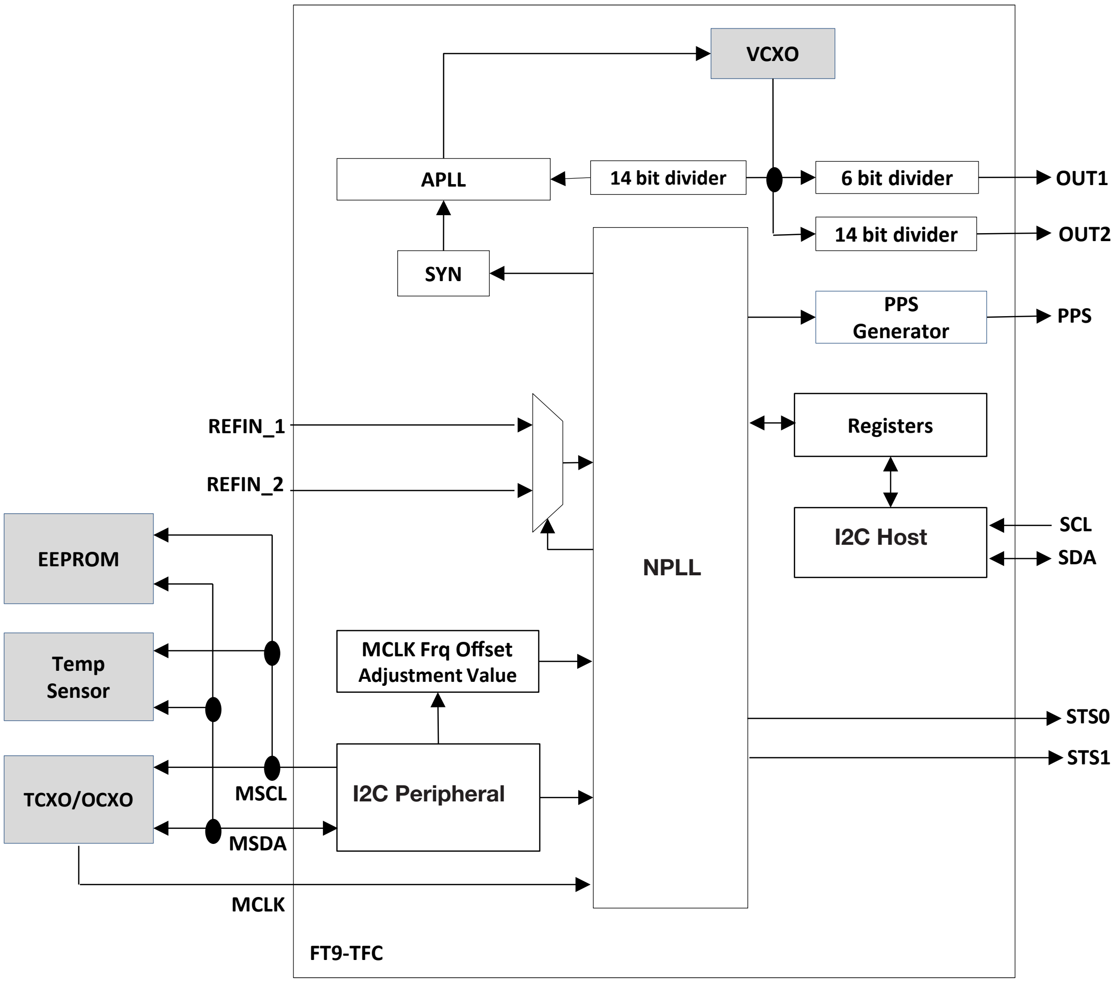
- Phase locks to one incoming 1 PPS Reference input
- Generates two (2) single ended Low Jitter Clock Outputs derived from externalVCXO
- Precision phase alignment of frequency outputs to 1 PPS phase locked output
- Pre-configured APLL and default NPLL settings for three master clock types.
- Flexible status indicators for Lock and LOS/ Holdover conditions.
- Flexible external 10MHz MCLK/Holdover options available.
- Internal VCXO supports up to 160 MHz clock output frequency range
- Programmable output dividers from VCXO frequency
- Automatic compensation of external CC correctable OCXO module
- Programmable NPLL bandwidth settings for 1PPS disciplining
- I2C Interface for system communication and programming.
- 3.3VDC Supply Voltage
- -40°C to 105°C operating temperature range
- 9.2x 9.2mm 16pin QFN surface mount package

Applications

- Primary Reference Time Clock (PRTC) [G.8272]
- Telecom Grand Master [G.8273.1]
- Telecom boundary clock [G.8273.2]
- Wireless Base Stations
- GNSS Disciplined Oscillator
- NTP Stratum 0 Standard

Bulletin	TM145
Revision	04
Date	07 Feb 2024

FT9-TFC Functional Block Diagram



Absolute Maximum Ratings

Parameter	Minimum	Nominal	Maximum	Units	Notes
Storage Temperature	-55	-	125	°C	
Supply Voltage	-0.5	-	4.5	Vdc	
Operating Supply Voltage 3.3 Vdc	3.13	3.30	3.47	Vdc	

Absolute Ratings: Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only. The functional operation of the device at those or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to conditions outside the "recommended operating conditions" for any extended period of time may adversely impact device reliability and result in failures not covered by warranty.

Operating Specifications

Parameter	Minimum	Nominal	Maximum	Units	Notes
Supply Voltage (AVDD33, DVDD33)	3.13	3.30	3.47	Vdc	
Supply Current		125	150	mA	
Operating temperature Range	-40	-	105	°C	

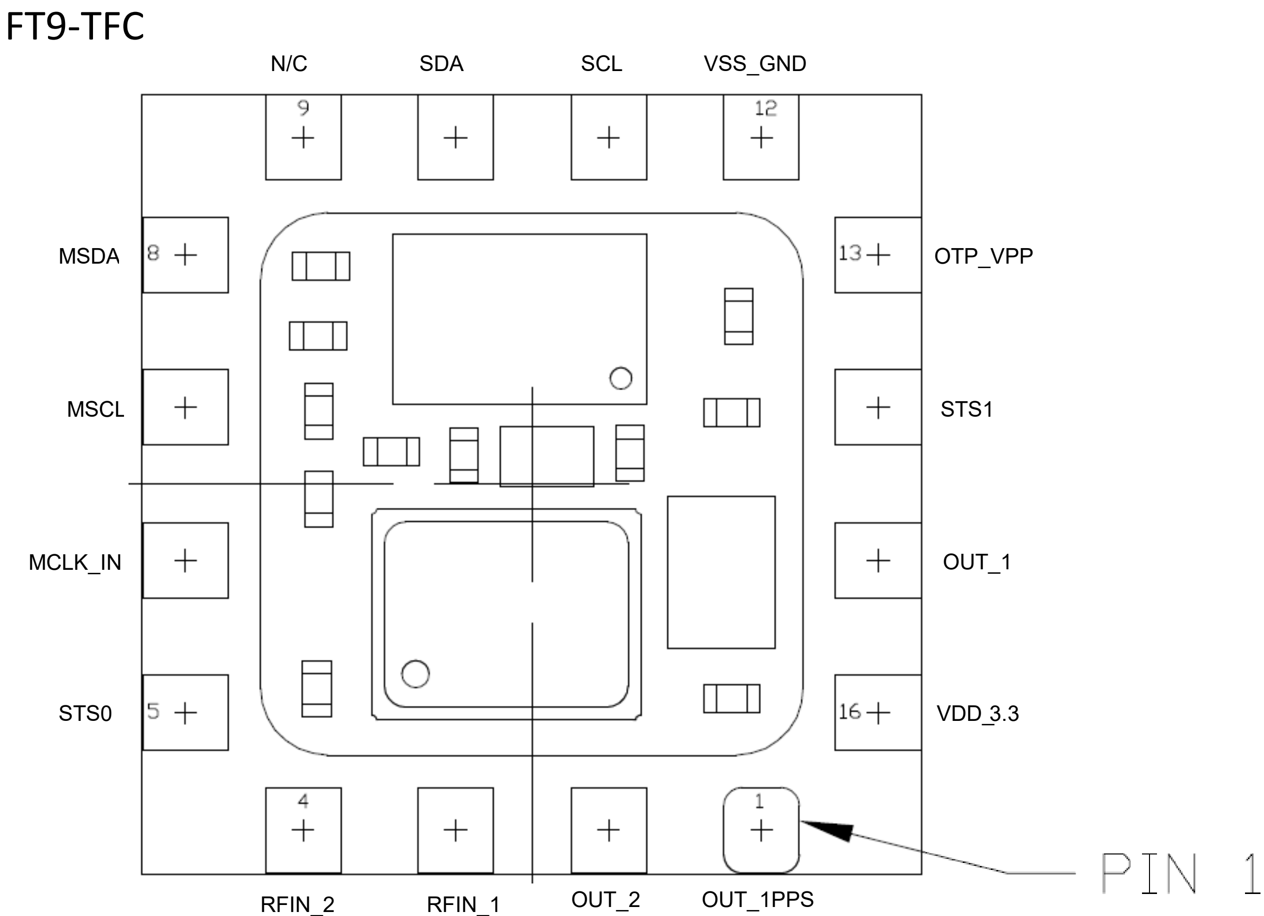
LVCMOS Output Characteristics

Parameter	Minimum	Nominal	Maximum	Units	Notes
Load	-	15	-	pF	
Output Voltage					
(High) (Voh)	3.0	-	-	V	
(Low) (Vol)	-	-	0.4		
Duty Cycle at 50% of Vcc	45	50	55	%	
Rise / Fall Time 10% to 90%	-	-	6	ns	

Package Characteristics

Package	9.2 x 9.2mm 16 pin QFN
Moisture Sensitivity Level	MSL-3

FT9-TFC Pin Assignments (Bottom View)



FT9-TFC Pin Description

Pin No.	Pin Name	I/O	Description
1	OUT_1PPS	O	1PPS Output Generator
2	OUT_2	O	3.3V LVCMOS Output
3	REF IN_1	I	Accept 3.3V LVCMOS 1PPS input
4	REF IN_2	I	Accept 3.3V LVCMOS 1PPS input
5	STS0	O	Status Pin
6	MCLK_IN	I	Master Clock input. Accepts 3.3V LVCMOS clock signal input
7	MSCL	I/O	Peripheral Series Clock Output. Communications with data from Temp Sensor
8	MSDA	I/O	Peripheral Series Data. Accepts data from Temp Sensor
9	NC		
10	SDA	I/O	HOST 12C Series Data
11	SCL	I/O	HOST 12C Series Clock Output
12	VSS_GND	Power	GROUND
13	OTP_VPP	Power	6.5V power input while programing OTP
14	STS1	O	Status Pin
15	OUT_1	O	3.3V LVCMOS Output
16	VDD_3.3	Power	3.3V Power Input

General Description

The FT9-TFC is modular design based on Connor-Winfield's NS3D02 1PPS synchronizing ASIC. In the modular format, an on-board VCXO is assembled and configured for ease of use as a plug and play module. Three NPLL setting options support three levels of master clock performance, each with its own loop bandwidth setting commensurate with the performance level of the master clock anticipated to be used to support this module's NPLL filter. This module is capable of being externally programmed by the user, except for the analog PLL which is configured based on the specific VCXO frequency in each device. While the default settings will allow the user to operate this device without programming it, all of the information for programming the module are available in this document.

The design architecture incorporates a sophisticated digital and analog PLL scheme to provide 2 low jitter phase/frequency locked clock outputs at frequencies from 8 kHz to 160MHz including a 1PPS pulse generator output. The system is clocked with an external precision 10MHz OCXO or TCXO providing the basis for various holdover and free run performance options. The FT9-TFC's NPLL (Numerical PLL) can be programmed for filter bandwidths from 100mHz to less than 1mHz for disciplining the incoming 1PPS signal.

The chip digitally synthesizes two outputs from the timing generator, one clock and a 1PPS pulse output. The 1PPS pulse is brought out directly from the NPLL synthesizer and the clock output functions as reference input to the chip's follow on APLL (Analog PLL) circuit within the chip. The analog portion of the chip consists of an independent APLL circuit with integrated charge pump and phase detector, supported by an internal VCXO, that translates the frequency and attenuates the jitter on the synthesized clock output generated in the NPLL section of the chip. Two single ended output clocks are derived from the disciplined VCXO. The internal VCXO used provides the output characteristics for phase noise and jitter performance for the 2 clock outputs. The two LVCMOS level clock output transmitters have follow on divider circuits available. Output 1 has a 6-bit divider capability, while output 2 has a 14-bit divider capability.

The FT9-TFC has functionality for creating frequency offsets in its internal master clock that allows for a compensation scheme to support stability enhancement and high precision holdover performance. A peripheral I2C connection supports direct communication with OCXO modules enabled with compatible circuitry.

Internal NPLL and Numeric Timing Generator

The kernel of the FT9-TFC is an NPLL (Numerical-based PLL). In its core, all internal modules are either digital or numerical, including the phase detectors, filters, timing generator and clock synthesizers. The pure digital design timing generator allows the FT9-TFC to become an accurate and reliable deterministic system.

The FT9-TFC's internal timing generator can be set to operate in FORCED_LOS/FREERUN, SELECT_REF1, SELECT_REF2, and FORCED_LOS/HOLDOVER mode. Operating in either SELECT_REF1 or SELECT_REF2, NPLL will discipline its clock synthesizer to phase-lock on the external timing source from the selected 1PPS reference. FT9-TFC's PLL loop bandwidth may be programmed from 100mHz down to lower than 1mHz to vary the timing generator's filtering function. If no valid 1PPS signal is present on that selected reference port, the timing generator will move to holdover mode automatically. Operating in either forced free-run or forced holdover mode, the timing generator will ignore all the 1PPS signals on both reference ports and operates in self-timing without any timing source. The difference between in free-run mode and in holdover mode is whether the timing generator shall apply its memory of the frequency offset last used while disciplined by a previous external-timing source.

An internal clock synthesizer generates an internal clock signal at any frequency from 40kHz to 1MHz in 8kHz step to act as the reference input to the follow-on APLL (Analog PLL). The APLL's clock was fanned out to two clock output ports, each has its own post divider to divide down the frequency. There is also a pulse generator to output a 1PPS pulse. Users could program to whether or how the phases were synchronized among them.

The FT9-TFC's timing generator is clocked by a fixed frequency external LVCMOS-level 10MHz as master clock (MCLK).) The MCLK input characteristics could be provided by an OCXO or a high precision TCXO that are capable of supporting both the filter bandwidth the user chooses as well as any holdover performance requirements. The external clock source not only dictates the holdover performance in LOS condition, it also plays the key role to limit the performance of the timing generator's PLL operating in low loop bandwidth. FT9-TFC also provides a series of features of numerical frequency offset calibration of less than 1 part per trillion resolution to enhance the frequency stability of the internal MCLK beyond its naked external MCLK source.

FT9-TFC Register Table

I/O Description: R = Read Only; W = Write Only; r/W = Write, but previous written value could be read back; R/W = Read and Write

ADDR	BITS	REGISTER NAME	I/O	DESCRIPTION
0x00~0x06	7 x [7:0]	Chip_ID	R	Chip ID, from byte #0 to byte #6
0x07	[7:0]	Chip_REV	R	Chip Revision of FT9-TFC_Rev_1.1
0x08	[7:0]	NPLL_FW_REV	R	NPLL's Firmware Revision of FT9-TFC_Rev_1.1
0x09	[7:0]	IC_INTR_EVENT	R/W	IC's INTERRUPT Event
0x0A	[7:0]	IC_INTR_MASK	r/W	IC's INTERRUPT Mask
0x0B	[0]	INTR_PIN_EN	r/W	To enable INTERRUPT pin
0x0C	[0]	LED_PIN_EN	r/W	To enable LED pin
0x0D	[7:0]	TEST_MODE0	r/W	TEST MODE0 (for internal use only)
0x0E	[15:0]	TEST_MODE1	r/W	TEST MODE1 (for internal use only)
0x10	[7:0]	APLL_REF_FREQ	r/W	The frequency of APLL's input reference synthesizer
0x11	[13:0]	APLL_FB_DIV	r/W	APLL's feedback divider
0x13	[0]	APLL_R0_CHOICE	r/W	APLL's R0 choice
0x14	[8:0]	APLL_R0_VALUE_L	r/W	APLL's R0_VALUE_L resistance selection
0x16	[4:0]	APLL_R0_VALUE_S	r/W	APLL's R0_VALUE_S resistance selection
0x17	[3:0]	APLL_R2_DIV	r/W	APLL's R2 resistance
0x18	[11:0]	APLL_CP_CURRENT	r/W	APLL's charge pump current
0x1A~0x1F	[47:0]	TEST_MODE2	r/W	TEST MODE2 (for internal use only)
0x20	[31:0]	NPLL_FLL_CONFIG	r/W	Configuration of NPLL in FLL_LOCKING mode
0x24	[7:0]	NPLL_PLL_DAMPING_FACTOR	r/W	Damping factor of NPLL in all PLL modes
0x25	[7:0]	NPLL_PLL_FAST_LOCKING_LBW	r/W	LBW of NPLL in PLL FAST_LOCKING mode
0x26	[31:0]	NPLL_PLL_LEAKBUCK_CONFIG	r/W	Leaking Bucket configuration of NPLL in all PLL modes
0x2A	[14:0]	NPLL_PHe_ReENTRY_TOL	r/W	NPLL ReENTRY phase error tolerance
0x2C	[14:0]	NPLL_PHe_LOL_TOL	r/W	NPLL LOL phase error tolerance
0x2E	[7:0]	NPLL_PBO_SPEED_LIMIT	r/W	Maximal PBO compensation phase shifting speed
0x2F	[7:0]	NPLL_MISC_CONFIG	r/W	NPLL Misc. Configuration
0x30	[14:0]	OUT_ALIGN_FREQ	r/W	Align Frequency to 1PPS output
0x32~0x37				~RSVD~
0x38	[7:0]	WARMUP_LIMIT_INC_INTERVAL	r/W	The interval to increase Warm-Up LBW Index Limit
0x39	[7:0]	WARMUP_LIMIT_INIT	r/W	The initial Warm-Up LBW Index Limit since NPLL being Kicked Up
0x3A	[14:0]	WARMUP_LIMIT_FINAL	r/W	The final ceiling of Warm-Up Index Limit could be increased to
0x3C	[0]	MCLK_USE_AUTO_TEMP_ADJ	r/W	Selection of using automatic MCLK temperature compensation calculation
0x3D	[1:0]	MCLK_TEMPCO_SRC	r/W	MCLK Temperature Compensation Coefficient Source Selection
0x3E	[0]	MCLK_TEMP_SENSOR_TYPE	r/W	Temperature Sensor Type for MCLK Temperature Compensation
0x3F	[0]	MCLK_TEMP_ADJ_REVERSE	r/W	To reverse MCLK Temperature Compensation Polarity
0x40				~RSVD~
0x41	[1:0]	NPLL_RT_REF_SEL	r/W	NPLL Active Reference Selection
0x42	[14:0]	NPLL_RT_PLL_TARGET_LBW	r/W	NPLL Target LBW in PLL mode
0x44	[15:0]	NPLL_RT_REF1_CALI	r/W	NPLL REF1 phase calibration
0x46	[15:0]	NPLL_RT_REF2_CALI	r/W	NPLL REF2 phase calibration
0x48	[7:0]	NPLL_RT_STS0_CRITERIA	r/W	NPLL Combo-Status STS0 Criteria Selection
0x49	[7:0]	NPLL_RT_STS1_CRITERIA	r/W	NPLL Combo-Status STS1 Criteria Selection
0x4A	[7:0]	PPS_RT_PULSE_LENGTH	r/W	PPS_OUT pulse length
0x4B	1:0]	MCLK_RT_TEMP_SENSOR_RATE	r/W	Temperature Sensor Reading Rate
0x4C	[31:0]	MCLK_RT_USER_CALI	r/W	User specified MCLK calibration
0x50~0x5C				~RSVD~
0x5D	[5:0]	OUT1_RT_POST_DIV	r/W	OUT1 Post Divider
0x5E	[13:0]	OUT2_RT_POST_DIV	r/W	OUT2 Post Divider

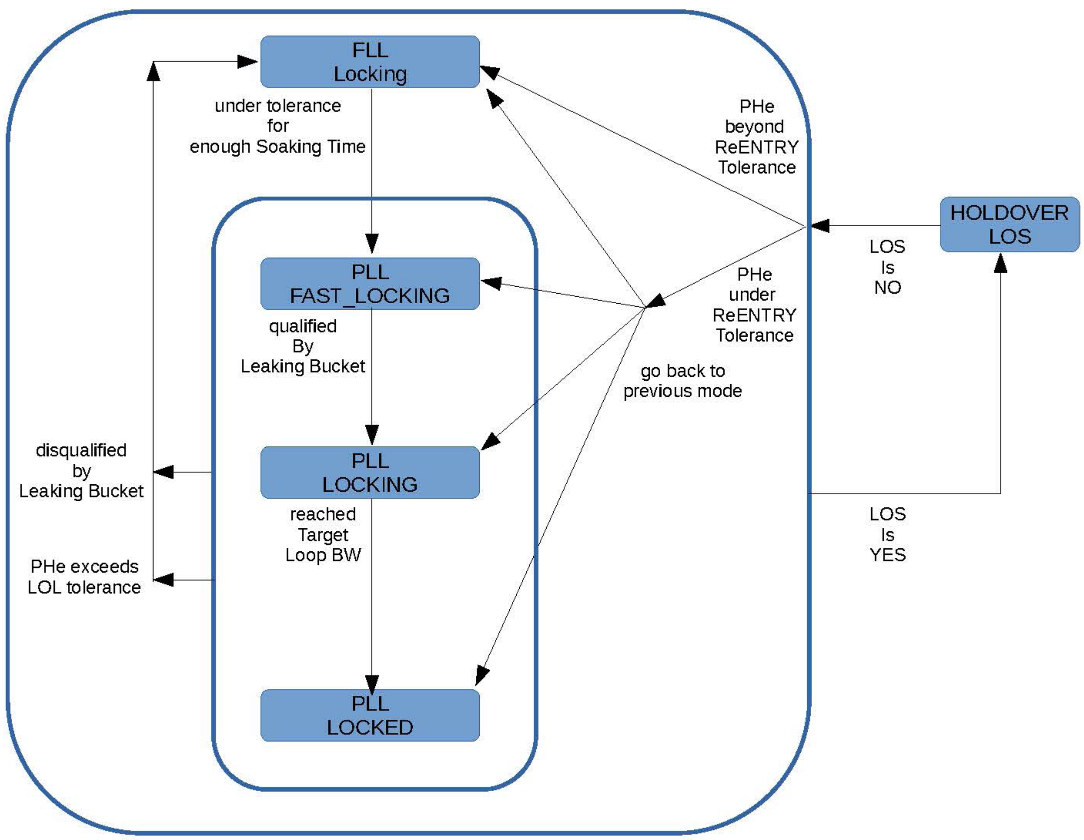
FT9-TFC Register Table continued

ADDR	BITS	REGISTER NAME	I/O	DESCRIPTION
0x60	[7:0]	NPLL_KICKUP	r/W	NPLL Kick-Up
0X61	[7:0]	NPLL_INFO	R	NPLL Status Information
0x62	[15:0]	NPLL_INFO_EX	R	NPLL Extra Information
0x64	[15:0]	NPLL_1PPS_Rx_COUNT	R	NPLL PPS Reference Receiving Count
0x66	[31:0]	NPLL_PHe_CALI	R	NPLL PPS_INPUT Phase Error Calibration
0x6A	[31:0]	NPLL_PBO_remain	R	NPLL PBO Compensation Remaining
0x6E	[15:0]	NPLL_PLL_LBW_NOW	R	NPLL Current LBW in all PLL modes
0x70	[31:0]	NPLL_OUT_FFO	R	NPLL Clock Output Fractional Frequency Offset
0x74	[31:0]	NPLL_HOLDOVER_HISTORY	R	NPLL Accumuated Holdover History
0x78~0x7B				~RSVD~
0x7C	[14:0]	WARMUP_LIMIT_NOW	R	The current Warm-Up LBW Index Limit
0x7E~0x88				~RSVD~
0x89	[7:0]	MCLK_TEMPCO_PAGE_IDX	R	EEPROM Page Index of founded MCLK Temperature Coefficient Table
0x8A	[15:0]	MCLK_TEMP_SENSOR_VALUE	R	MCLK Temperature Sensor Raw Reading
0x8C	[31:0]	MCLK_AUTO_ADJ_RESULT	R	MCLK Temperature Compensation Calculation Result
0x90~0x9F				~RSVD~
0xA0	[5:0]	LOAD_STATUS	R	OTP/EEPROM loading status
0xA1	[7:0]	EE/OTP_PAGE_IDX	r/W	Target EEPROM/OTP read/write page index
0xA2	[0]	EEPROM_CMD	W	EEPROM command
	[2:0]	EEPROM_STS	R	EEPROM status
0xA3	[7:0]	SOFT_RESET	W	Module Soft Reset Command
0xA4	[1:0]	OTP_CMD	r/W	OTP command
	[1:0]	OTP_STS	R	OTP STATUS
0xA5	[11:0]	OTP_PWE_TIMER	r/W	OTP PWE Timer
0xA7	[0]	PERIPHERAL_I2C_BUS_STS	R	Peripheral I2C bus Master Controller status
0xA8~0xBF				~RSVD~
0xC0~0xFF	64 x [7:0]	PAGE_BUFFER	R/W	EEPROM/OTP read/write page buffer, from byte#0 to byte#63

Basic Operation: 1PPS Locking and Phase Synchronization Solution

When the NPLL is "kicked up", the module starts up in LOS_FREERUN/HOLDOVER mode by default. All the clock/pulse outputs are generated based on the characteristics of the external MCLK source, or in conjunction with any user specified frequency offset calibration instructions set in registers. When the module is commanded to lock on a valid 1PPS input, it first calculates the real phase position of those input pulses, adjusting for any user specified phase detector calibration, and then measured every period of this 1PPS input. The user can choose to output a 1PPS pulse at all times after the NPLL is kicked up or only after the first-time phase tracking has begun on the incoming 1PPS signal.

The process of locking to the incoming 1PPS input signal is designed to be accomplished in incremental locking stages. These locking stages must be configured using the NPLL configuration registers prior to kicking up the NPLL. Each locking stage can be configured with a loop bandwidth setting consistent with the user's configuration and setup, incrementally reducing the filter's bandwidth as the system stabilizes in locking to the incoming 1PPS input pulse and reducing the phase error between the incoming 1PPS pulse and the output 1PPS pulse to a minimal level. This process begins with a frequency locking stage. In this operation mode, the module adjusts its clock/pulse output frequency attempting to "frequency lock" to the 1PPS input. The system will claim frequency lock is established once the frequency offset fluctuation on the output 1PPS keeps under a threshold value for longer than a given soaking time. The phase locking mode then begins. Users are allowed to specify the frequency offset fluctuation threshold and the soaking time to optimize the frequency lock regulation in respect to the noise characteristics of the incoming 1PPS reference.



While moving from frequency lock to phase lock mode, a phase build-out and the intermediate fast PLL locking mode provides a smooth and safe transient until he FT9-TFC satisfies full locking status. The design incorporates a "leaking bucket" mechanism that allows the system to gauge the statistical condition of the phase error fluctuation between the incoming 1PPS and the output 1PPS. With a given phase error threshold, the violation accumulation and leaking level of phase error monitoring presents the wellness of the phase locking progress. At any point, a hard phase noise tolerance violation will disqualify the phase lock condition immediately. With the endorsement from the leaking bucket, the PLL moves from the fast-locking mode to normal locking mode and gradually lowers its loop bandwidth from fast locking mode's high loop bandwidth to the targeted low loop bandwidth. However, the FT9-TFC may need to roll back to its frequency locking mode whenever it violates the PLL's phase lock condition from the leaking bucket and/or the hard toleration monitoring.

The design takes into consideration the various MCLK source options available to users. Due to the differing warm up characteristics of OCXOs and TCXOs, the user can set various delay options to accommodate for low bandwidth settings with the time necessary for the MCLK to settle to a level to support those bandwidth settings.

Basic Operation: 1PPS Locking and Phase Synchronization Solution continued

When the system suffers a loss of signal condition (LOS), the module will enter the LOS/HOLDOVER mode. In this mode, the FT9-TFC freezes its PLL and holds the frequency offset between its MCLK and the output clocks. Under this condition, the PLL loses its discipline source and the clock outputs operate in the behavior of its stand-alone external MCLK oscillator. Even with the MCLK calibration enhancement, staying in LOS/HOLDOVER mode too long will cause the output clocks' phase to drift away from other clocks disciplined by a valid reference source.

Once a valid incoming 1PPS signal returns, the system resumes back the original locking stage before the LOS event. Depending upon how long the holdover period was and how much phase drift occurred, it may roll back all the way to the frequency locking mode if the phase error exceeds a certain re-entrance phase error threshold identified by the user.

The FT9-TFC supports phase alignment between the output 1PPS and the two output clocks divided down from APLL's VCXO output. However, the output clock's frequency must be divisible by 8kHz for phase alignment to be possible.

Certain OCXO models provisioned for external compensation are capable of having a micro-controller to project their frequency offset due to the inclusion of temperature sensing circuitry and pre-programmed coefficient values added by the manufacturer. The FT9-TFC's internal design can take advantage of the frequency offset projection to improve the overall stability performance of the MCLK oscillator. The FT9-TFC can automatically take temperature readings from the external temperature sensor and execute a frequency offset correction to compensate for the OCXO's instability due to temperature fluctuations. In addition, the design enables further MCLK calibration commands from host controllers directly in runtime for even more sophisticated frequency offset projection compensation.

The FT9-TFC design provides manipulation to compensate its reference input phase detector. The user can make calibration adjustments to address cable length propagation delay as well as dynamically applying phase offset adjustments to compensate for the inherent sawtooth error in typical GPS receivers where quantization error messaging is supported.

In configuring the registers for the system, registers that are not identified as "run time" registers (_RT_) must be set prior to kicking up the NPLL. Any register that is not defined as a run time register will not react to any register changes after the NPLL is kicked up. Run time registers are designed to be used during operation of the NPLL and can be adjusted dynamically before and after NPLL kick up.

Detailed Desciption

Incoming 1PPS signal on pin REFIN_1 and REFIN_2

FT9-TFC has two reference input ports REFIN_1 and REFIN_2. While only one port can be selected at a time to be used as the input to the NPLL, both pins, 14 and 15, can be receiving a 1PPS reference. This enables the user to dynamically switch between two 1PPS input sources in real time. The design allows the user to have individual phase control/calibration to each pin to compensate the latency delay of each one's signal path and the possible quantization error of individual 1PPS source through register **NPLL_RT_REF1_CALI and NPLL_RT_REF2_CALI**, to have virtual hit-less switching between two reference inputs.

The value chosen in register **NPLL_RT_REF_SEL** determines which input port the module will recognize as its 1PPS incoming signal. There is no functionality for reference qualification and automatically switching between two input references in the event of LOS on one pin.

The design allows the user to manually and dynamically switch between two signal sources in both configuration and run-time, via register. The FT9-TFC monitors and reports only on the reference actively being used.

NPLL FLL Configuration

To lock on a first-seen 1PPS incoming reference, the timing generator starts from the FLL (frequency-lock loop) mode in the beginning. In this frequency locking stage, the module adjusts its clock/pulse output frequency attempting to "frequency lock" to the selected 1PPS input. Users shall program the register **NPLL_FLL_CONFIG** to configure the minimum soaking time for this process to take and the frequency offset fluctuation tolerance threshold for the frequency lock loop. The system will claim frequency lock is established once the peak-to-peak frequency offset fluctuation on the output 1PPS keeps under the threshold value for longer than the soaking time.

NPLL PLL Configuration

Just like in the FLL stage, many parameters need to be programmed to configure the operating in PLL stage. First, the basic parameters of a PLL includes the damping factor and both loop bandwidths for the initial fast locking mode and the final locked mode. Second, the leaking bucket's phase error threshold and its bucket size, the phase error tolerances for LOL (loss of lock) and re-entry from holdover are needed. Besides these, the phase shifting speed limit of phase built-out compensation and the loop bandwidth shifting rate from fast locking to final locked have to be programmed. In addition, there are also various other miscellaneous registers that are required to be set, such as when to generate the first 1PPS pulse, the criteria to accumulate the frequency output history for holdover mode once the incoming signal is lost, and also how the phase alignment is to be set up among the 1PPS output and those two clock outputs divided down from VCXO's clock. All of these configurations could be done by programming the following registers.

- NPLL_PLL_DAMPING_FACTOR, NPLL_PLL_FAST_LOCKING_LBW, and NPLL_RT_PLL_TARGET_LBW
- NPLL_PLL_LEAKBUCK_CONFIG, NPLL_PBO_SPEED_LIMIT, NPLL_PHe_LOL_TOL, and NPLL_PHe_ReENTRY_TOL
- NPLL_MISC_CONFIG and OUT_ALIGN_FREQ

FT9-TFC's NPLL was designed to support 4 different damping factors in PLL mode, including 0.7, 1.4, 2.0, and 3.5, by programming register **NPLL_PLL_DAMPING_FACTOR**, to cover most of applications. The loop bandwidth in the initial PLL fast locking mode could be configured from (1/10) Hz down to (1/255) Hz, and the loop bandwidth of the final PLL locked mode could even go down to 0.03 mHz.

When NPLL claims that FLL has been achieved, and frequency lock mode moves to fast-locking PLL mode, the NPLL will do three things: (1) initialize the PLL loop bandwidth to the bandwidth defined in register **NPLL_PLL_FAST_LOCKING_LBW**, (2) synchronize the output phase and build out the phase error to wave off the initial phase hit, and (3) reset/start the leaking bucket.

The NPLL synchronizes the output phase with a phase jump to its 1PPS output to the same location as the 1PPS incoming reference, such that there is no initial phase hit entering the fast-locking PLL mode. However, if the phase alignment is required between its 1PPS output and the other two clock outputs as called for by setting register **OUT_ALIGN_FREQ**, the 1PPS pulse output's new location is limited to the phase locations of those two clock outputs divided down from the internal VCXO (also see register **OUT_ALIGN_FREQ**), causing an initial phase error between the 1PPS incoming reference and the 1PPS output to exist. To eliminate the initial phase hit entering the fast-locking PLL, FT9-TFC issues a one time phase build-out. However, the phase error being built-out needs to be compensated back to finally have the phase locked and aligned between the 1PPS incoming reference and the 1PPS output. The design has the phase compensation process in a smooth way to avoid causing the frequency hit on the output. The maximal frequency swing to compensate the phase is configured in register **NPLL_PBO_SPEED_LIMIT**.

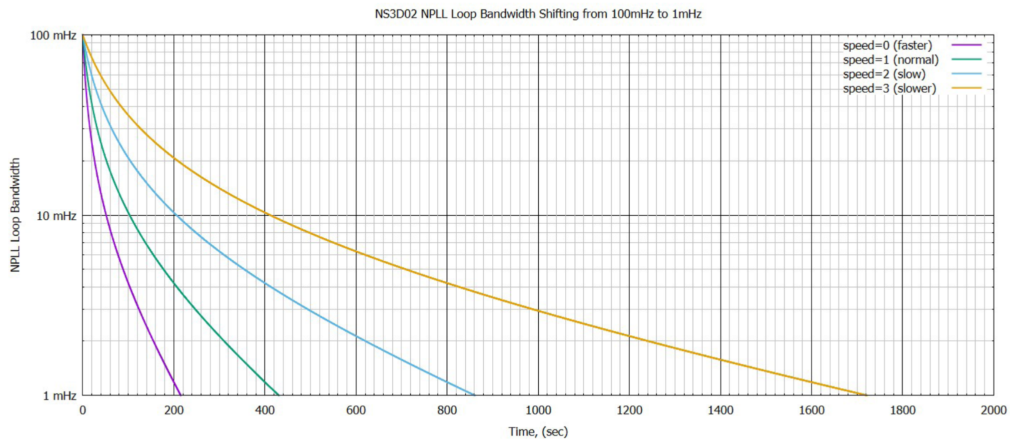
The leaking bucket's phase error threshold could be configured from 0 to 1023nS. The bucket size could be selected from 1 to 65,535. The initial bucket level always starts from half of the bucket size every time a phase position of the incoming 1PPS reference was detected in the PLL mode. Once the phase error between the position and 1PPS output after phase build-out exceeds the threshold, the leaking bucket level will be raised by 1, otherwise, it will be reduced by 1. The floor of the bucket level is 0 and the ceiling is the bucket size. If the level reaches the ceiling, a statistic LOL will be recognized and triggers the NPLL to roll back to the FLL mode. Of course, if the phase error exceeds the hard limit defined by register **NPLL_PHe_LOL_THRESHOLD**, NPLL will go back to FLL mode right away.

Detailed Description continued

NPLL PLL Configuration continued

NPLL will initialize the PLL's loop bandwidth to the bandwidth defined in register **NPLL_PLL_FAST_LOCKING_LBW**, starting from fast-locking mode. The leaking bucket will be reset and start to monitor the long-term statistic phase lock condition. Once the leaking bucket's level reaches 0, a statistic phase lock is recognized and moves the PLL to normal PLL LOCKING mode. The loop bandwidth of the PLL will start to shift from the fast locking's higher loop bandwidth toward the lower final locked bandwidth defined in register **NPLL_RT_PLL_TARGET_LBW**. The bandwidth shifting rate is configured in register **NPLL_MISC_CONFIG**. Once the loop bandwidth becomes no higher than the target loop bandwidth, the PLL will claim to be in PLL LOCKED mode.

LBW Shifting from 100mHz down to 1mHz



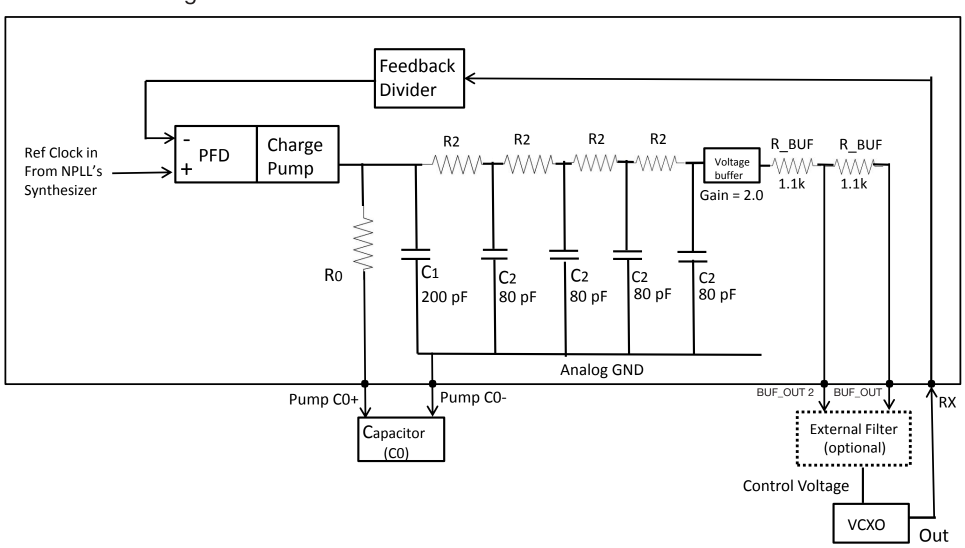
The NPLL may not always have a valid reference to lock to. The NPLL will move to LOS/HOLDOVER/FREERUN mode under two conditions: 1, LOL was detected on the selected reference input port, or 2,the user forced a move to FREERUN or HOLDOVER by programming register **NPLL_RT_REF_SEL**. When neither of these two conditions exist, the NPLL will try to re-enter to FLL/PLL mode from the freerun/holdover mode. The user can define a phase error threshold by using the register **NPLL_Phe_ReENTRY_TOL**. When re-entering from freerun/holdover mode, if the measured phase error exceeds the threshold, NPLL will roll back to operate in FLL mode, otherwise, it will resume back to its previous mode and condition just before the LOS or forced LOS happened.

APLL Configuration

The analog PLL (APLL)'s function in FT9-TFC is to discipline the internal VCXO to output up to two divided down clocks. The APLL will take a clock synthesized by the NPLL as its reference input, translate the frequency to higher frequency and also attenuate the jitter generated by NPLL's digital clock synthesizer.

FT9-TFC's APLL circuit contains a phase frequency detector (PFD), a programmable charge pump, an uncompleted programmable passive low-pass filter (LPF), a fixed-gain voltage buffer, and a programmable clock feedback divider. With some extra external capacitors to complete the LPF, the APLL can operate to cover loop bandwidth ranging from 10Hz to 200 Hz easily.

APLL Circuit Diagram



Detailed Description continued

APLL Configuration continued

The APLL's input reference is synthesized from the NPLL and can be programmed to have frequency from 40kHz to 1MHz, in 8kHz step, by configuring the register APLL_REF_FREQ. The charging pump could be programmed to have current ranging from 0.3125uA to 1,280uA by configuring the register **APLL_CP_CURRENT**. Users must add an external capacitor C0 (connecting to pin PUMP_C0+ and pin PUMP_CO-) to complete the passive LPF. This passive LPF includes a programmable resistor R0, an external capacitor C0, a fixed capacitor C1, 4 programmable resistors R2 (R21~4), and 4 fixed capacitors C2 (C21~4). Those programmable resistors could be configured by register **APLL_R0_CHOICE, APLL_R0_VALUE_L, APLL_R0_VALUE_S** and **APLL_R2_DIV**.

The voltage buffer has a fixed-gain of x2. The buffer also provides the impedance isolation between the internal LPF and the VCXO's voltage control pin. The output of the buffer goes through two internal serial resistors, each has 1.1k ohm resistance. The end of these two resistors connects to pin BUF_OUT, and the center-tap of these two resistors connects to pin BUF_OUT2. Users could simply connect from either one to VCXO's control voltage pin directly or build a more sophisticated post filter utilizing these two existing internal resistors by adding some external capacitors. The last part of the APLL circuit is the internal programmable non-fractional feedback divider, configured by the 14-bit register **APLL_FB_DIV**.

Clock/Pulse Outputs

The FT9-TFC generates up to two 3.3V LVCMOS level clock outputs on two output transmitter ports plus a 1PPS pulse output.

Output transmitter ports OUT1 and OUT2 are driven by clocks divided down from APLL's internal VCXO clock. The output ports consist of a single ended 3.3Vdc CMOS level transmitter. Output transmitters can be disabled if not in use. The LVCMOS output transmitter driving capability is 12 mA.

In front of each output port, a programmable divider is available to divide down the clock from VCXO. The divider at OUT1 is a 6-bit divider, capable of dividing from 1 to 63 or shut-off. OUT2 has a 14-bit divider, capable of dividing from 2 to 16,383 or shut-off. Please pay attention that OUT2 doesn't support to output the same frequency of the VCXO. These two dividers could be programmed by register **OUT1_RT_POST_DIV and OUT2_RT_POST_DIV.** Both of them are adjustable in run-time.

The pulse length of the 1PPS OUTPUT is also programmable, by programming register **PPS_RT_PULSE_LENGTH**, in step of mini sec. This is also adjustable in run-time.

Clock Phase Alignment to 1PPS OUTPUT

FT9-TFC provides limited capability of phase alignment between the 1PPS OUTPUT and the two clock outputs that are divided down from the internal VCXO's clock. The limitation is that only clock output frequencies divisible by 8kHz can be configured to phase align to the 1PPS OUTPUT. Configuring register **OUT_ALIGN_FREQ** with the frequency of the clock output to be aligned with the 1PPS OUTPUT enables this alignment feature. If both clock outputs OUT1 and OUT2 are required to phase align to the 1PPS OUTPUT, the frequency to program into this register has to be the 8kHz-base common divisor of both output's frequencies. Choosing the GCD (greatest common divisor) of them will reduce the phase error caused by the phase built-out while moving from FLL mode to fast-locking PLL mode. Please see the previous section **NPLL PLL Configuration**.

STS0/STS1 Status Pins

Customized STS0 and STS1 run-time monitoring statuses

FT9-TFC has two internal, user-customized run-time monitoring status pins, STS0 and STS1. Each could be customized to report a variety of the system's operation status as to whether it meet its own criteria by programming register NPLL_RT_STS0_CRITERIA and NPLL_RT_STS1_CRITERIA. The supported criteria covers a broad range of the combination of NPLL locking status, APLL locking status, phase build-out compensation status, and MCLK warm-up status.

Pin STS0/INTERRUPT

FT9-TFC has no dedicated INTERRUPT pin, however, Pin STS0 can be programmed to function as an INTERRUPT notification pin by programming register INTR_PIN_EN. FT9-TFC's major events are latched on register IC_INTR_EVENT. After correlating to the interrupt mask (register IC_INTR_MASK), if there are any events still present, the interrupt pin will be asserted high, until all the presented events have been erased. Users can read and erase those interrupt events via register IC_INTR_EVENT. If not working as an interrupt pin, pin STS0 will present the internal customized monitoring status STS0.

Pin STS1/LED

Just like pin STS0, pin STS1 (pin 22) can be used for an alternate feature. FT9-TFC can have pin STS1 function as an LED indicator pin. When register LED_PIN_EN bit [0] is 1, pin STS1 becomes pin LED. This feature can be used in conjunction with LED driving circuits to manipulate the LED driving circuits to present NPLL_MODE indications. This pin will drive high while NPLL is in LOS/HOLDOVER mode, continue to blink continuously at a 4Hz rate in FLL_LOCKING mode, 3 rapid blinks per second if in PLL_FAST_LOCKING mode, 2 rapid blinks per second if in PLL_LOCKING mode and finally one blink per second if in PLL_LOCKED mode. If not working as a LED pin, pin STS1 will present the internal customized monitoring status STS1.

STS0/STS1 Status Pins continued

MCLK Warm-Up

For highly stable and high precision OCXOs and TCXOs, their specifications generally meet their serious performance specification only after being powered for an extended period of time. After a long cool-off periods, upon power up, it takes time to settle down their instabilities and thus, need ample time to warm up. If measuring their stability performance without enough warm up time, they could easily violate their own performance specification.

This warm-up issue becomes a challenge when defining the NPLL system to have very low loop bandwidth settings. Without enough warm up time for the MCLK oscillator, the stability of the oscillator is not fit to work for a low loop bandwidth PLL. The FT9-TFC provides features to meet this challenge.

Option 1

FT9-TFC allows the users to change in run-time its target PLL loop bandwidth. With a micro-controller, users could set a higher target loop bandwidth after power up and lower it down to its desired filtering bandwidth after passing the needed warm up time by programming register **NPLL_RT_PLL_TARGET_LBW** in run-time. FT9-TFC's NPLL will transient from a previous higher filtering bandwidth to the lower desired target bandwidth gradually, depending on the loop bandwidth shifting rate defined in register **NPLL_MISC_CONFIG**.

Option 2

The FT9-TFC warm up feature uses a simple model to roughly predict the usable filtering loop bandwidth of an oscillator during its warm up time by assuming the usable bandwidth is proportional to power up time's reciprocal.

By programming register **MCLK_WARMUP_LIMIT_INIT, MCLK_WARMUP_LIMIT_INC_INTERVAL** and **MCLK_WARMUP_LIMIT_FINAL**, FT9-TFC could predict the allowed filtering bandwidth limit of the given MCLK oscillator. FT9-TFC then applies this warm up constraint to its PLL loop bandwidth not allowing its loop bandwidth lower than this limit. Using this feature, an external micro-controller is not needed, and the warm up loop bandwidth transient speed is defined by those MCLK_WARMUP_LIMIT_**xxxx** registers instead of the register **NPLL_MISC_CONFIG**. This MCLK warm up feature will be turned off if register **MCLK_WARMUP_LIMIT_INC_INTERVAL** was set to zero.

Register Group Introduction

FT9-TFC's registers are separated into following eight groups:

- ID and Revision
  has read-only product identity registers, including IC's ID, hardware revision, and firmware revision
- System
  has all the system hardware configuration registers, such like interrupt event/mask and shared pin configuration
- Configuration of APLL
  contains all the registers to configure the APLL
- Configuration of NPLL, Alignment, and MCLK
  contains most of NPLL's configuration registers, plus registers for Alignment and MCLK automatic calibration's configuration
- Run-Time Control
  has all the registers for run-time control, such like
  REFIN_1's phase error calibration
  REFIN_2's phase error calibration
  NPLL's reference selection
  NPLL's target loop bandwidth
  STS0's criteria
  STS1's criteria
  1SSP_OUT's pulse length
  OUT1's post divisor
  OUT2's post divisor
  MCLK's user calibration value
  Rate of temperature sensor's automatic reading and frequency offset projection calculation
- NPLL Kick-Up
  has only one register NPLL_KICKUP
- Run-Time Information
  contains all the registers to present the run-time status of this IC
- EEPROM/OTP, Loader and Soft Reset
  contains all the registers to soft reset the whole module and to access the content of the internal OTP and the external EEPROM on the Peripheral I2C bus

NPLL Kick-Up and the Register Programming Sequence

The special register NPLL_KICKUP was used to kick up NPLL's state machine by setting it to non-zero value. Once NPLL's state machine has started, only power cycle or the Soft Reset can reset it. All the registers in group (Configuration of APLL) and group (Configuration of NPLL, Alignment, and MCLK) have to be set before NPLL kick up. Any value changing of these registers after NPLL kick-up will either be ignored or cause serious circuit and firmware error.

"Do NOT write to any register of group (Configuration of APLL) and group (Configuration of NPLL, Alignment, and MCLK) after kicking up NPLL"

The following is the recommended register program sequence after reset or power on

1. programming all necessary registers in group (System)
2. programming all necessary registers in group (Configuration of APLL)
3. programming all necessary registers in group (Configuration of NPLL, Alignment, and MCLK)
4. programming all necessary registers in group (Run-Time Control)
5. write a non-zero value to register NPLL_KICKUP

MCLK Frequency Offset Compensation

The FT9-TFC takes its MCLK clock from an external fixed frequency 10MHz MCLK oscillator input on the MCLK_IN pin. However, free running clock oscillators generally have a calibration error as well as a temperature-caused frequency offset away from its nominal frequency. FT9-TFC provides users an ability to make adjustments to compensate the frequency offset. While receiving the external MCLK clock, FT9-TFC could apply an additional frequency offset adjustment to the it while converting into the internal MCLK system clock by simply programming register **MCLK_RT_USER_CALI**. As a run-time control register, this feature is suitable to not only compensate its original calibration error but also the run-time frequency offset, such as thermal-caused frequency offset, during system operation.

In addition, the FT9-TFC is capable of automatically calculating the thermal frequency offset of the external MCLK oscillator by itself using a stateless polynomial function to convert from the ambient temperature to projected frequency offset. With an external temperature sensor and a set of temperature coefficients of the polynomial function for the oscillator thermal stability characteristics, the FT9-TFC can calculate the external oscillator's projected frequency offset and apply a compensating offset adjustment to its own MCLK calibration automatically without the need of an external micro-controller.

MCLK Automatic Frequency Offset Projection

The FT9-TFC uses a stateless 5th order polynomial function to project its external MCLK oscillator's frequency offset. The coefficients of this polynomial function could either be extracted from some certain OCXO directly (e.g., Connor-Winfield OH320-CC), or be stored in an external EEPROM or in the FT9-TFC's internal MTP/OTP non-volatile memory. The FT9-TFC was designed to automatically probe certain models of temperature sensors to gather the temperature reading by itself and then calculate the projected frequency offset by applying each temperature reading to the polynomial function. The coefficient source and the temperature sensor type could be configured in register **MCLK_TEMPCO_SRC** and register **MCLK_TEMP_SENSOR_TYPE**. The temperature reading and the calculated projected frequency offset could be read back from register **MCLK_TEMP_SENSOR_VALUE** and register **MCLK_AUTO_ADJ_RESULT**. Users could set FT9-TFC's temperature reading rate by programming register **MCLK_RT_TEMP_SENSOR_RATE**, to either holding the reading, or be 1, 2, or 4 times/sec. FT9-TFC support two different temperature sensor model, either Texas Instruments' TMP116/117 or AMS's AS621X. The temperature coefficients could be stored on the same EEPROM used for field upgrade. The EEPROM has to be ATMEL's AT24C128C, or other compatible parts. Both the temperature sensor and the EEPROM have to be located on the Peripheral I2C bus. Users don't need to program the temperature sensor. FT9-TFC will initialize the temperature sensor by itself. The I2C addresses of the temperature sensor and the EEPROM will be defined in later I2C bus document section. Be aware that the temperature reading rate could be programmed in run-time. The format used to store the temperature coefficients on the internal MTP/OTP and the external EEPROM will be illustrated later on another sector.

MCLK Compensation, Manual plus Automatic

The FT9-TFC could take the MCLK calibration values from both register MCLK_RT_USER_CALI combined with the automatic projected frequency offset calculation result. Register **MCLK_USE_AUTO_TEMP_ADJ** could be configured to define whether the MCLK calibration taken on only the manual value from register MCLK_RT_USER_CAL, or both the manual value plus the run-time calculation result. Manufacturer's temperature coefficients could be either for projecting its thermal frequency offset or the needed adjustment to compensate this offset. Register **MCLK_TEMP_ADJ_REVERSE** could be configured to define the polarity of how the calculation result combines to the manual user calibration values. Be aware of that both **MCLK_USE_AUTO_TEMP_ADJ** and **MCLK_TEMP_ADJ_REVERSE** are configuration registers. This means that they could not be changed in run-time. The boolean of register MCLK_USE_AUTO_TEMP_ADJ will not affect the temperature reading and the calculation result. This also means that even if MCLK_USE_AUTO_TEMP_ADJ was set to FALSE, users could still utilize FT9-TFC's automatic temperature reading and frequency offset projection calculation.

The Conflict between MCLK Automatic Temperature Reading and Other Operations

The operation of the automatic temperature sensor reading could not coexist with some other operations, including any other peripheral I2C bus activity, operating by this FT9-TFC or others, and the MTP/OTP memory read and write. The automatic temperature reading has to be put into hold by programming register **MCLK_RT_TEMP_SENSOR_RATE** in advance of these operations. For example, the reading needs to be turned off before accessing (reading/writing) to either the external EEPROM or the internal MTP/OTP.

For safety, it is recommended to make the peripheral I2C bus a module's private local I2C bus containing only the temperature sensor and the EEPROM, or the compatible OCXO such as Connor-Winfield's OH20TSE.

I2C Buses

FT9-TFC has two I2C controllers, one master controller for its peripheral I2C interface and one slave controller for its host I2C interface. Both I2C interfaces are compliant to the multi-device I2C bus standard. A user's host controller could manipulate the FT9-TFC by accessing the internal registers using the host I2C interface via pin SDA and SCL.

The FT9-TFC provides access to some certain external peripheral parts like MCLK oscillator modules that have the requisite internal circuitry to help projecting its frequency offset for auto calibration, external EEPROMs, and temperature sensor components using the peripheral I2C interface via pin MSDA and MSCL.

All the peripheral parts OCXO-CC, temperature sensor, and the external EEPROM for automatic MCLK frequency offset compensation, field upgrade, and parameter initialization shall be placed on the Peripheral I2C bus interfacing via pin MSDA and MSCL. To access FT9-TFC, user's host controller shall use the Host I2C bus interfacing via pin SDA and SCL.

For I2C bus, parts on the same bus cannot share I2C ID addresses. The FT9-TFC could be configured up to 4 different ID addresses to support multiple IC's existence on the same I2C without address conflict. However, the I2C ID addresses of all the peripheral parts for FT9-TFC are defined. Users have to configure the ID addresses of those parts in the PCB design stage.

I2C	Bus Device Name	I2C Device Address	Remark
Host	NS3D0s	0b 010.0100 0b 010.0101 0b 010.0110 0b 010.0111	Configurable
Peripheral	EEPROM (Atmel AT24C128C or compatible)	0b 101.0000	Must be fixed
	EEPROM (inside provisioned MCLK OCXO-CC module P/N)	0b 101.0100	Must be fixed
	Temperature Sensor (TI TMP116/117) Temperature Sensor (AMS AS621X) Temperature Sensor (inside provisioned MCLK OCXO-CC module P/N)	0b 100.1000	Must be fixed

FT9-TFC Host I2C Interface, the I2C Frame and Data Transfer Format

The user host controller (micro-controller or FPGA) could have register reading and writing to access and manipulate FT9-TFC. It supports a 7-bit I2C ID address. The format is MSB-bit (most significant bit) leading. The format uses only one byte for 8-bit register address. For read/write multiple bytes in burst mode, the register address will be increased by one automatically for each data byte.

Abbreviations

- $\underline{A}$: Acknowledge
- A: No acknowledge
- S: Start
- P: Stop
- R: Read
- W: Write
- Sr: Repeated Start

Host I2C Frame Format

Write Format

Single Byte

S	ID[6:0]	W	A	ADR[7:0]	A	WDATA[7:0]		A	P

Multiple Bytes

S	ID[6:0]	W	A	ADR[7:0]	A	WDATA1[7:0]	A	WDATA2[7:0]		A	P

Read Format 1

Single Byte

S	ID[6:0]	W	A	ADR[7:0]	A	Sr	ID[6:0]	R	A	RDATA[7:0]	$\underline{A}$	P

Multiple Bytes

S	ID[6:0]	W	A	ADR[7:0]	A	Sr	ID{6:0}	R	A	RDATA1[7:0]	A	RDATA2[7:0]	$\underline{A}$	P

Read Format 2

Single Byte

S	ID[6:0]	W	A	ADR[7:0]	A	P
S	ID[6:0]	R	A	RDATA1[7:0]	A	P

Multiple Bytes

S	ID[6:0]	W	A	ADR[7:0]			A	P
S	ID[6:0]	R	A	RDATA1[7:0]	A	RDATA2[7:0]	A	P

Multiple-Byte Register Read/Write Data Format

FT9-TFC's registers are either single-byte or multiple-byte registers. For multiple-byte registers, the reading/writing sequence is to start from the LSB (least significant byte) byte to the MSB (most significant byte) byte, without any other read/write interrupt. The LSB byte of a multiple-byte registers always have the lower address. The timeout of each multiple-byte reading is around 100 mini seconds. The writing of the multiple bytes will take affect after the MSB byte writing.

Internal OTP/MTP and External EEPROM

The Internal OTP/MTP

FT9-TFC has an internal OTP of 13,696 byte size. The first 13,312 bytes store the firmware image used by FT9-TFC's proprietary embedded ALU. This image will be pre-programmed by the manufacturer before shipping. The other 384 bytes could be used as multiple-time programming storage, split into six 64-byte segment of MTPs, to carry the TEMPCO (temperature coefficients) table for the automatic thermal frequency offset projection. The format of the TEMPCO table will be illustrated in a later section.

The External EEPROM

FT9-TFC can work with an external field upgrade EEPROM to carry the update firmware and the customized register initial values. It could also piggyback carry the TEMPCO Set for automatic thermal frequency offset projection. This EEPROM, being either ATMEL's AT24C128C or its compatible products with I2C ID address of 0b101.0000, has to be placed on the Peripheral I2C bus via pins MSDA and MSCL.

The first 13,504 bytes on the EEPROM will store the update firmware and the customized register initial values. Its first 13,312 bytes are for the update firmware and the following 160 bytes are the initial values of the registers of address from 0x00 to 0x9F. Only write-able registers' values will be initialized by the content here. The next 28 bytes are not used, followed by two byte of magic key 0x55 and 0xAA, and then the last two bytes for CRC16 checksum. The CRC16 checksum covers the previous 13,502-byte content, using MSB-bit-leading CRC16 checksum algorithm with CRC16 polynomial (X16) + X15 + X2 + X0.

Then the first 64 bytes of the leftover 2880 bytes could be used to piggyback store the TEMPCO Set for automatic thermal frequency offset prediction.

Every time FT9-TFC was booted up from soft reset (see register SOFT_RESET) or from power on, FT9-TFC will load from its internal OTP first, then it will check the existence of the external EEPROM. If the EEPROM exists and the content of its bytes 13,500 and 13,501 match the magic keys, the update firmware and the register initial values will be downloaded to override the content from OTP.

FT9-TFC Register Detailed Description Table

ID and Revision

ADDR	BITS	REGISTER NAME	I/O		VALUE	DESCRIPTION
0x00	[7:0]	Chip_ID Byte 0	R		0x4E	Chip ID, byte 0: ASCII code of 'N'
0x01	[7:0]	Chip_ID Byte 1	R		0X53	Chip ID, byte 1: ASCII code of 'S'
0x02	[7:0]	Chip_ID Byte 2	R		0x33	Chip ID, byte 2: ASCII code of '3'
0x03	[7:0]	Chip_ID Byte 3	R		0x44	Chip ID, byte 3: ASCII code of 'D'
0x04	[7:0]	Chip_ID Byte 4	R		0x30	Chip ID, byte 4: ASCII code of '0'
0x05	[7:0]	Chip_ID Byte 5	R		0x32	Chip ID, byte 5: ASCII code of '2'
0x06	[7:0]	Chip_ID Byte 6	R		0x00	Chip ID, byte 6: ASCII code of '\0'
0x07	[7:0]	Chip_REV	R		1	Chip Revision of NS-3D02_Rev_1.1
0x08	[7:0]	NPLL_FW_REV	R		1	NPLL's Firmware Revision of NS-3D02_Rev_1.1

System

ADDR	BITS	REGISTER NAME	I/O	DEFAULT	DESCRIPTION
0x09	[7:0]	IC_INTR_EVENT	R/W	0	IC_INTR_EVENT, each bit presents a special event bit[0] REG(NPLL_MODE) value changed bit[1] REG(NPLL_INFO) value changed bit[2] 1PPS_REF received bit[3] 1PPS_OUT sent bit[7:4] ~rsvd~ READ: to query the status of interrupt events WRITE: to cancel the dedicated interrupt events <ps> Bit-write '1' to each associated bit will erase that event bit individually (write-to-erase, not write-to-replace).
0x0A	[7:0]	IC_INTR_MASK	r/W	0	The passing mask of previous REG(IC_INTR_EVENT) to assert the pin (INTR). Each mask-bit being 0 will block its associated interrupt event to assert the interrupt pin.
0x0B	[0]	INTR_PIN_EN	r/W	0	To enable the interrupt feature which replaces the STS0 indication on pin(STS0) 0: disable 1: enable
0x0C	[0]	LED_PIN_EN	r/W	0	To enable the LED indication which replaces the STS1 indication on pin(STS1). 0: disable 1: enable This LED pin can be used to output to LED driving circuit to indicate the NPLL_MODE status. LOS/HOLDOVER signal level high FLL_LOCKING continue blinking (4Hz) PLL_FAST_LOCKING 3 rapid blinks per second PLL_LOCKING 2 rapid blinks per second PLL_LOCKED 1 blink per second
0x0D	[7:0]	TEST_MODE0	r/W	4	Test Mode0 (for internal use only). VALUE MUST BE 4
0x0E	[15:0]	TEST_MODE1	r/W	1	Test Mode1 (for internal use only). VALUE MUST BE 1

Configuration of APLL

ADDR	BITS	REGISTER NAME	I/O	DEFAULT	DESCRIPTION
x10	[7:0]	APLL_REF_FREQ	r/W	0	The frequency of APLL's reference clock synthesized from NPLL. FREQ = reg_value x 8kHz **reg_value must be in the range from 5 to 125**
0x11	[13:0]	APLL_FB_DIV	r/W	0	APLL's 14-bit feedback divider; reg_value==0 means disable
0x13	[0]	APLL_R0_CHOICE	r/W	0	APLL's R0 choice, 0 Selection of R0 using register APLL_R0_VALUE_L 1 Selection of R0 using register APLL_R0_VALUE_S
0x14	[8:0]	APLL_R0_VALUE_L	r/W	0	APLL's R0_VALUE_L resistance selection. Unit in ohm. bit[0] RL[0]; 0: 1k, 1: 10k bit[1] RL[1]; 0: 1k, 1: 20k bit[2] RL[2]; 0: 1k, 1: 40k bit[3] RL[3]; 0: 1k, 1: 80k bit[4] RL[4]; 0: 1k, 1: 160k bit[5] RL[5]; 0: 1k, 1: 320k bit[6] RL[6]; 0: 1k, 1: 640k bit[7] RL[7]; 0: 1k, 1: 1280k bit[8] RL[8]; 0: 1k, 1: 2560k R0_VALUE_L = RL[0] + RL[1] + … RL[7] + RL[8]
0x16	[4:0]	APLL_R0_VALUE_S	r/W	0	APLL's R0_VALUE_S resistance selection. Unit in ohm. bit[0] Rs[0]; 0: 0.4k, 1: 2.5k bit[1] Rs[1]; 0: 0.4k, 1: 5k bit[2] Rs[2]; 0: 0.4k, 1: 10k bit[3] Rs[3]; 0: 0.4k, 1: 20k bit[4] Rs[4]; 0: 0.4k, 1: 40k R0_VALUE_S = Rs[0] + Rs[1] + Rs[2] + Rs[3] + Rs[4]
0x17	[3:0]	APLL_R2_DIV	r/W	0	APLL's R2 resistance = (160k ohm) / reg_value; **This value must NOT be 0**.
0x18	[11:0]	APLL_CP_CURRENT	r/W	0	APLL's charge pump current = 0.3125uA * reg_value
0x1A~0x1F	[47:0]	TEST_MODE2	r/W	0	Test Mode2 (for internal use only). Don't change the value.

Configuration of NPLL, Alignment and MCLK

ADDR	BITS	REGISTER NAME	I/O	DEFAULT	DESCRIPTION
0x20	[31:0]	NPLL_FLL_CONFIG	r/W	0	The configuration of NPLL in FLL_LOCKING mode bit[15:0] Soaking Time, unit in second bit[23:16] Pk-Pk output FFO tolerance, unit in ppb bit[31:24] ~rsvd~
0x24	[7:0]	NPLL_PLL_DAMPING_FACTOR	r/W	0	The damping factor of NPLL in all PLL modes 0: 0.7 1: 1.4 2: 2.0 3: 3.5 4~255 ~rsvd~
0x25	[7:0]	NPLL_PLL_FAST_LOCKING_LBW	r/W	0	The LBW of NPLL in PLL FAST_LOCKING mode LBW = 1 / reg_value, unit in Hz, Valid value range is from 10 to 255
0x26	[31:0]	NPLL_PLL_LEAKBUCK_CONFIG	r/W	0	The Leaking Bucket configuration of NPLL in all PLL modes bit[9:0] phase error threshold, unit in nS bit[15:10] ~rsvd~ bit[31:16] bucket size
0x2A	[14:0]	NPLL_PHe_ReENTRY_TOL	r/W	0	NPLL's phase error tolerance of ReENTRY, unit in nS.
0x2C	[14:0]	NPLL_PHe_LOL_TOL	r/W	0	NPLL's phase error tolerance to claim LOL in all PLL modes, unit in nS.
0x2E	[7:0]	NPLL_PBO_SPEED_LIMIT	r/W	0	The maximal phase shifting speed to compensate a previous PBO (phase build-out) phase error. bit[5:0] index=0~63 bit[7:6] band=0~3 maximal shift speed = index x (16band) x (10 pS/S)
0x2F	[7:0]	NPLL_MISC_CONFIG	r/W	0	NPLL's Misc. Configuration bit[1:0] LBW shifting speed 0: FAST 1: NORMAL 2: SLOW 3: SLOWER bit[3:2] Holdover History Update Criteria 0: FLL_LOCKING and up 1: PLL_FAST_LOCKING and up 2: PLL_LOCKING and up 3: PLL_LOCKED bit[4] Default PPS Output 0: No default 1PPS OUT There will be no 1PPS OUT since NPLL being kicked up or forced into Freerun mode until first time phase tracking on 1PPS reference input. 1: Always has 1PPS OUT There will always be 1PPS OUT since NPLL was kicked up. bit[7:4] ~rsvd~
0x30	[14:0]	OUT_ALIGN_FREQ	r/W	0	The frequency of output to align to 1PPS output. FREQ = 8kHz x reg_value Reg_value 0 means no need of phase alignment (phase error will be still be fixed but arbitrary).

Configuration of NPLL, Alignment and MCLK continued

ADDR	BITS	REGISTER NAME	I/O	DEFAULT	DESCRIPTION
0x32~0x37					~RSVD~
0x38	[7:0]	WARMUP_LIMIT_INC_INTERVAL	r/W	0	The interval to increase Warm-Up LBW Index Limit. • unit(sec) • The Warm-Up LBW Index Limit will be increased by one for every specified interval since NPLL has being kicked up, until it reaches the final target limit. • The LBW index will approach to the target LBW index, but will be limited by the warm-up LBW index limit. • This warm-up LBW Index limitation feature will be disabled if this interval value was set to zero
0x39	[7:0]	WARMUP_LIMIT_INIT	r/W	0	The initial Warm-Up LBW Index Limit since NPLL was Kicked Up
0x3A	[14:0]	WARMUP_LIMIT_FINAL	r/W	0	The final target LBW of Warm-Up Index Limit being increased to • if this warm-up LBW index limit feature is on, the final target LBW must be lower than initial limit LBW = 1 / reg_value, unit in Hz
0x3C	[0]	MCLK_USE_AUTO_TEMP_ADJ	r/W	0	To specify whether using automatic MCLK temperature compensation calculation result 0: No 1: Yes
0x3D	[1:0]	MCLK_TEMPCO_SRC	r/W	0	The source selection of the Temperature Coefficient for automatic MCLK temperature compensation calculation 0: internal OTP The image content format will be illustrated by other document. 1: external I2C EEPROM (ATMEL AT24C128C or compatible) The image content format will be illustrated by other document. 2,3: external I2C EEPROM inside CW OH20TSE The image content format was illustrated by CW's document.
0x3E	[0]	MCLK_TEMP_SENSOR_TYPE	r/W	0	The type of the temperature sensor used for automatic MCLK temperature compensation 0: TI TMP116/117, or compatible 1: AMS AS621X, or compatible
0x3F	[0]	MCLK_TEMP_ADJ_REVERSE	r/W	0	The reverse the positive/negative polarity of calculation result of the automatic MCLK temperature compensation. 0: No reverse 1: to reverse
0x40					~RSVD~

Run-Time Controls

ADDR	BITS	REGISTER NAME	I/O	DEFAULT	DESCRIPTION
0x41	[1:0]	NPLL_RT_REF_SEL	r/W	0	To specify the active reference selection of NPLL 0: forced LOS/FREERUN and reset HOLDOVER_HISTORY 1: select REF1 2: select REF2 3: forced LOS/HOLDOVER
0x42	[14:0]	NPLL_RT_PLL_TARGET_LBW	r/W	0	The target LBW of NPLL in PLL mode LBW = 1 / reg_value, unit in Hz
0x44	[15:0]	NPLL_RT_REF1_CALI	r/W	0	The NPLL phase calibration of REF1, unit in 0.01 nS, 2's comp
0x46	[15:0]	NPLL_RT_REF2_CALI	r/W	0	The NPLL phase calibration of REF2, unit in 0.01 nS, 2's comp
0x48	[7:0]	NPLL_RT_STS0_CRITERIA	r/W	0	The criteria of NPLL's STS0 status bit[2:0] LOGIC_NPLL 0: TRUE 1: LOS 2: in FLL_LOCKING and up 3: in PLL_FAST_LOCKING and up 4: in PLL_LOCKING and up 5: in PLL_LOCKED 6,7: ~rsvd~ bit[3] LOGIC_WARMUP 0: TRUE 1: MCLK warm-up enough for NPLL The warm-up will be enough if (MCLK_WARMUP_LIMIT_NOW ≥ NPLL_RT_TARGET_LBW). bit[5:4] LOGIC_APLL 0: TRUE 1: APLL is not LOCKED 2: APLL is LOCKED 3: ~rsvd~ bit[6] LOGIC_PBO 0: TRUE 1: PBO compensation is completed bit[7] invert 0: no invert, 1: invert the logic result The final logic result of STS0 is (invert) (LOG_NPLL, LOGIC_APLL, LOGIC_WARMUP and LOGIC_PBO)
0x49	[7:0]	NPLL_RT_STS1_CRITERIA	r/W	0	The criteria of NPLL's STS1 status same format/mechanism as REG(NPLL_RT_STS0_CRITERIA)
0x4A	[7:0]	PPS_RT_PULSE_LENGTH	r/W	0	The PPS output's pulse length, unit in mS.
0x4B	[1:0]	MCLK_RT_TEMP_SENSOR_RATE	r/W	0	Temperature Sensor Reading Rate for automatic MCLK temperature compensation 0: OFF 1: 1 time/sec 2: 2 times/sec 3: 4 times/sec FT9-TFC will communicate with the external temperature sensor to change its conversion cycle time and the average configuration automatically. Users do not need to communicate to the temperature sensor directly.
0x4C	[31:0]	MCLK_RT_USER_CALI	r/W	0	The MCLK extra user calibration, unit in (ppb/1024), 2's comp
0x50~0x5C					~RSVD~
0x5D	[5:0]	OUT1_RT_POST_DIV	r/W	1	The post divider of OUT1, 6-bit; 0: disable
0x5E	[13:0]	OUT2_RT_POST_DIV	r/W	1	The post divider of OUT2, 14-bit; 0,1: disable. Requires a minimum value of 2

NPLL Kick-Up

ADDR	BITS	REGISTER NAME	I/O	DEFAULT	DESCRIPTION
0x60	[7:0]	NPLL_KICKUP	r/W	0	NPLL Kick-Up. Write any non-zero value to kick-up NPLL. Once NPLL was kicked up, avoid to change the value of any register addressed from 0x10 to 0x40.

Run Time Information

ADDR	BITS	REGISTER NAME	I/O	VALUE	DESCRIPTION
0X61	[7:0]	NPLL_INFO	R		NPLL's status information bit[3:0] NPLL_MODE 0: LOS_FREERUN/HOLDOVER 1: FLL_LOCKING 2: PLL_FAST_LOCKING 3: PLL_LOCKING 4: PLL_LOCKED 5~15: ~rsvd~ bit[4] APLL_LOCKED; 0:NOT_LOCKED, 1:LOCKED bit[5] PBO_COMPENSATION_COMPLETED; 0:NOT_DONE, 1:DONE bit[6] NPLL status STS0; 0:FALSE, 1:TRUE bit[7] NPLL status STS1; 0:FALSE, 1:TRUE

0x62	[15:0]	NPLL_INFO_EX	R		NPLL's extra information
					if (bit[15]==0)
					bit[14:0] presents FLL_SOAKING_TIME's countdown
					else
					bit[14:0] presents LEAKING BUCKET's level

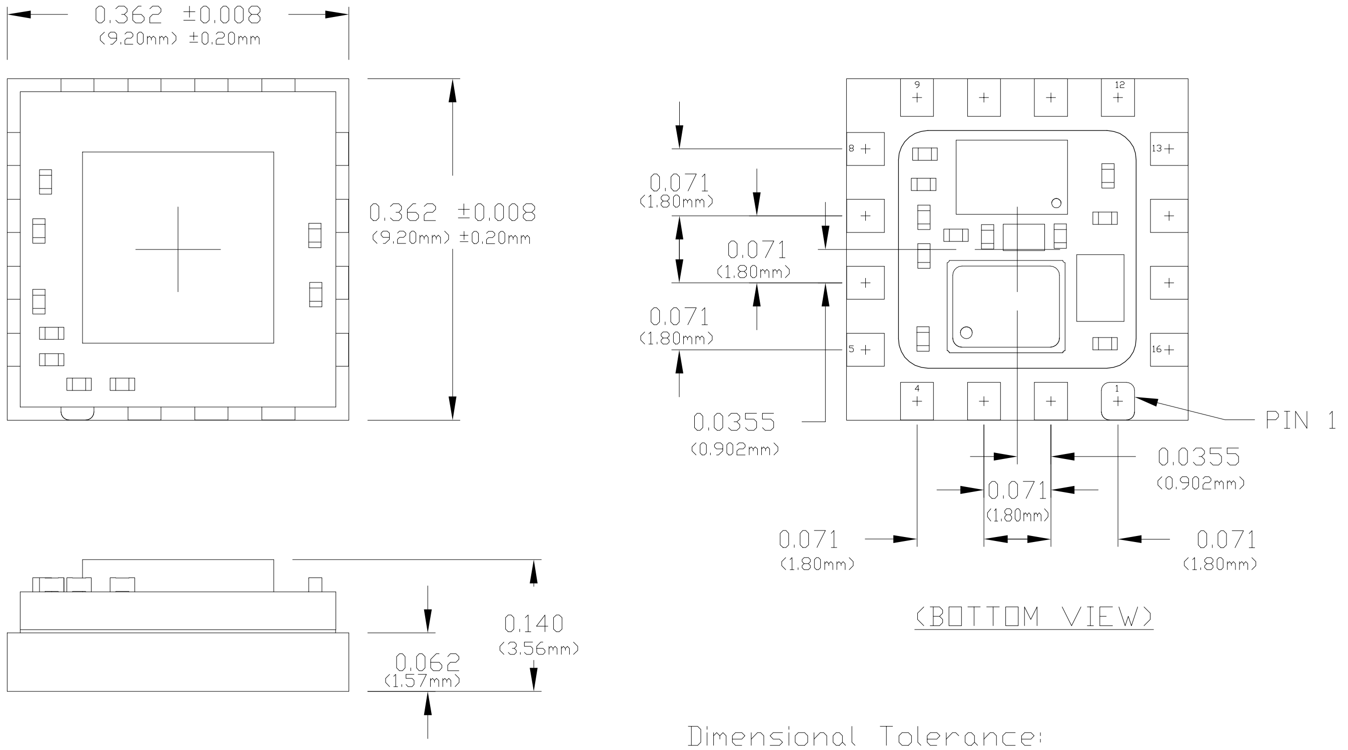
0x64	[5:0]	NPLL_1PPS_Rx_COUNT	R		NPLL's receiving count of valid 1PPS input; Carry-over bit of overflowed values will be truncated.
0x66	[31:0]	NPLL_PHe_CALI	R		NPLL's detected phase error between its 1PPS_OUT and 1PPS reference input after user applied phase calibration; unit in 0.01 nS, 2's comp.
0x6A	[31:0]	NPLL_PBO_remain	R		NPLL's compensation remain of previous PBO; unit in 0.01 nS, 2's comp.
0x6E	[15:0]	NPLL_PLL_LBW_NOW	R		NPLL's current LBW in all PLL modes LBW = (1 / reg_value), unit in Hz
0x70	[31:0]	NPLL_OUT_FFO	R		NPLL's clock output FFO (fractional frequency offset) away from calibrated MCLK, unit in (ppb/1024), 2's comp.
0x74	[31:0]	NPLL_HOLDOVER_HISTORY	R		NPLL's accumulated holdover history as FFO away from calibrated MCLK, unit in (ppb/1024), 2's comp.
0x78~0x7B					~RSVD~
0x7C	[14:0]	WARMUP_LIMIT_NOW	R		The current Warm-Up LBW Index Limit
0x7E~0x88					
0x89	[7:0]	MCLK_TEMPCO_PAGE_IDX	R		The EEPROM page index of the found Temperature Coefficient. The valid page index is from 214 to 255. Value other than these indicates no valid data found.
0x8A	[15:0]	MCLK_TEMP_SENSOR_VALUE	R		The raw value read from the temperature sensor for the automatic MCLK Temperature compensation
0x8C	[31:0]	MCLK_AUTO_ADJ_RESULT	R		The calculation result of the automatic MCLK temperature compensation, unit in (ppb/1024), 2's comp. <PS> Whether this calculation result will be applied to MCLK calibration depends on register MCLK_USE_AUTO_ADJ.
0x90~0x9F					~RSVD~

Run time Information continued

ADDR	BITS	REGISTER NAME	I/O	VALUE	DESCRIPTION
0xA0	[5:0]	LOAD_STATUS	R		LOAD status bit[0] load complete; 0: not completed, 1: completed bit[1] OTP content; 0: invalid, 1: valid bit[2] OTP content checksum; 0: FAILED, 1: SUCC bit[3] EEPROM existence; 0: non-detected, 1: detected bit[4] EEPROM content; 0: invalid, 1: valid bit[5] EEPROM content checksum; 0: FAILED, 1: SUCC
0xA1	[7:0]	EE/OTP_PAGE_IDX	r/W		To specify the page index of EEPROM/OTP to read from or write to
0xA2	[0]	EEPROM_CMD	W		EEPROM command to start the page reading or writing of a 64-byte page data • Write 0 to initiate the 64-byte writing from the REGS(PAGE_BUFFER) to the EEPROM page specified by REG(EE/OTP_PAGE_IDX). • Write 1 to initiate the 64-byte reading from the EEPROM page specified by REG(EE/OTP_PAGE_IDX) to the REGS(PAGE_BUFFER).
	[2:0]	EEPROM_STS	R		The status of the EEPROM bit[0] 0: WRITE, 1: READ bit[1] 0: ready, 1: not ready bit[2] 0: EEPROM exists, 1: EEPROM not exists
0xA3	[7:0]	SOFT_RESET	W		Module Soft Reset write value 0xA5 to reset the IC
0xA4	[0]	OTP_CMD	W		OTP command to start the page reading of writing of a 64-byte page data • Write 0 to initiate the 64-byte writing from the REGS(PAGE_BUFFER) to the OTP page specified by REG(EE/OTP_PAGE_IDX). • Write 1 to initiate the 64-byte reading from the OTP page specified by REG(EE/OTP_PAGE_IDX) to the REGS(PAGE_BUFFER).
	[1.0]	OTP_STS	R		The status of the OTP bit[0] 0: WRITE, 1: READ bit[1] 0: ready, 1: not ready
0xA5	[11:0]	OTP_PWE_TIMER	r/W		The time to program one byte on OTP, unit in internal clock cycle; For FT9-TFC, set the value to be 2333
0xA7	[0]	PERIPHERAL_I2C_BUS_STS	R		The status of Master I2C bus 0: IDLE 1: BUSY <ps>
0xA8~0xBF					~RSVD~
0xC0~0xFF	8 x 64	PAGE_BUFFER	R/W		The 64-byte page buffer for OTP/EEPROM page read/write operation

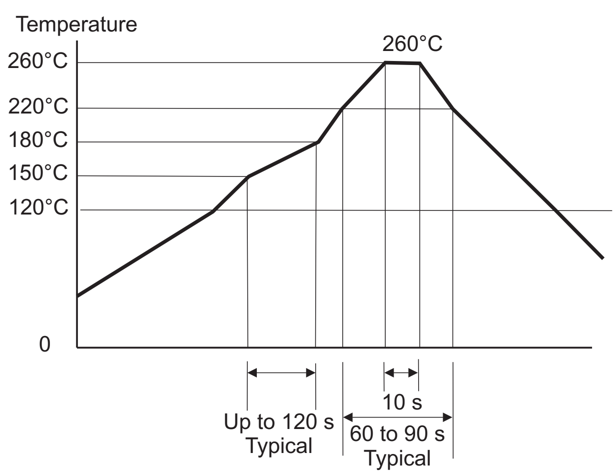
FT9-TFC

FT9-TFC Mechanical Drawing



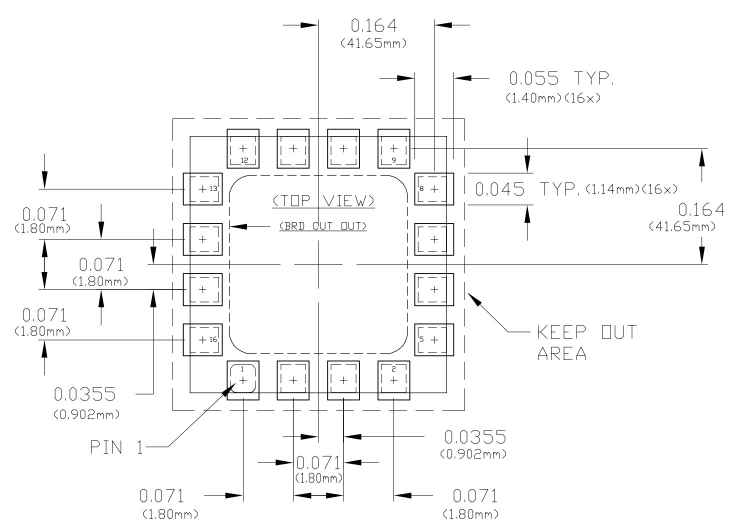
Dimensional Tolerance:
±0.005 (±0.127mm) unless shown otherwise

Solder Profile



Meets IPC/JEDEC J-STD-020C

Suggested Pad Layout



KEEP OUT AREA
UNDER THE PCBOARD IS A KEEP OUT AREA,
DO NOT PLACE ANY PARTS IN THIS AREA.

Standard Frequencies

10.00 MHz	A		
100.00 MHz	B	65.5360 MHz	T
122.88 MHz	C	32.768 MHz	W
125.00 MHz	D	64.0 MHz	X
156.25 MHz	E	38.880 MHz	Y
80 MHz	F	51.84 MHz	Z
98.304 MHz	G		
61.44 MHz	H	16.384 MHz	0
77.76 MHz	J	30.72 MHz	1
155.520 MHz	K	20.48 MHz	2
50 MHz	L	5.0 MHz	3
64 MHz	M	12.8 MHz	4
40 MHz	N	49.152 MHz	5
20.0 MHz	O	16.0 MHz	6
25.0 MHz	P	19.440 MHz	7
96.0 MHz	R	10.24 MHz	8
81.92 MHz	S	8 kHz	9

Ordering Information

FT9-TFC - X X X X X X X

Custom Option

MCLK TYPE
1 TCXO
2 Standard OCXO
3 HI Precision OCXO

*** Output Frequency 2**

NPLL Final LBW Setting
1 50mHz
2. 25mHz
2 10mHz
3 1mHz

Number of Output Frequencies
0: No Clock Output
1: 1 Output Frequency
2: 2 Output Frequencies

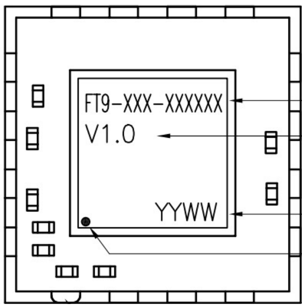
*** OUTPUT Frequency 1**

*** VCXO Frequency**

** See Standard Frequencies chart above. If the desired frequency is not listed, please contact a sales representative for availability of additional frequencies.*

2111 Comprehensive Drive
Aurora, Illinois 60505
Phone: 630-851-4722
Fax: 630-851-5040
www.conwin.com

Marking Configuration



Revision History

Revision	Date	Note
00	04/18/23	New Release
01	06/27/23	Updated Ordering Information
02	09/07/23	Updated Ordering Information and Electrical Specifications
03	12/21/23	Updated supply current, product photo, part # and marking configurations
04	02/07/24	Updated with Digi-Key information