Application Note

The Debug Guide for Network Synchronizers (Digital and Analog Phase-Locked Loops)

Jennifer Bernal *Clocks and Timing Solutions*

ABSTRACT

Network synchronizers use a combination of a digital and analog phase-locked loops (DPLL and APLL) to provide input-to-output synchronization, output clock generation, and jitter cleaning capabilities. With the DPLL, the network synchronizer can support additional features such as hitless switching, holdover, digitally-controlled oscillators (DCO), and jitter cleaning for close-in phase noise (less than 1kHz offsets). The addition of a DPLL adds complexity and can result in more debug steps than a simple clock generator.

The purpose of the application note is to provide a step-by-step guide on how to debug network synchronizers, specifically on how to achieve DPLL phase lock, which is a common debug inquiry. The devices applicable to the application note are listed in Table 1-1. When referring to all of the devices listed, the application note uses the term *LMK device* for a generalized name. When referring to a particular device grouping, the application note uses the family name.

Table 1-1. Network Synchronizer Device Families

DEVICE FAMILY NAME	DEVICE PART NUMBERS IN FAMILY
LMK05318(1)	LMK05318
LMK05318B	LMK05318B, LMK05318B-Q1, LMK5B12204
LMK5C(1)	LMK5C33216
LMK5B	LMK5B33216, LMK5B33414, LMK5B12212
LMK5CA	LMK5C33216A, LMK5C33414A, LMK5C33216AS1, LMK5C33414AS1, LMK5C22212A, LMK5C23208A

(1) Not recommended for new designs.

Table of Contents

Trademarks

All trademarks are the property of their respective owners.

1 Start Here: Using TICS Pro for Configuration and Readback

The first step in the debug process is to make sure the register configuration settings are correct. TICS Pro Software is a software tool that contains different GUI profiles for each of the LMK devices. Register configuration files and recommended programming sequences can be generated through the tool. The exported file formats are .tcs for the TICS Pro GUI configuration file and .txt for the HEX register list or sequence. Additionally, register readback is supported to get information about the status signals. Use TICS Pro to configure the registers properly and check the status signals of the LMK device.

1.1 TICS Pro Status Page

The LMK devices contain live status registers, which are self-clearing, and interrupt (sticky) status registers, which must be cleared by the user. Enable the global interrupt register by configuring the register as shown in Table 1-1. Clear the interrupt status bit by writing a 0 to the same interrupt status bit.

Table 1-1. Global Interrupt - Register Setting

DEVICE FAMILY	TICS PRO FIELD NAME	GLOBAL INTERRUPT ENABLE REGISTER SETTING
LMK05318B, LMK05318	INT_EN	R21[0] = 1
LMK5B, LMK5CA, LMK5C	INT_EN	R49[1] = 1

The TICS Pro software provides a visual representation of the status registers in the **Status** page of the LMK device profile. The **Status** page of the LMK5B33216 profile is shown in Figure 1-1 and the LMK05318B profile in Figure 1-2. The primary status registers, which are useful for debug and discussed in the later sections, are highlighted in red.

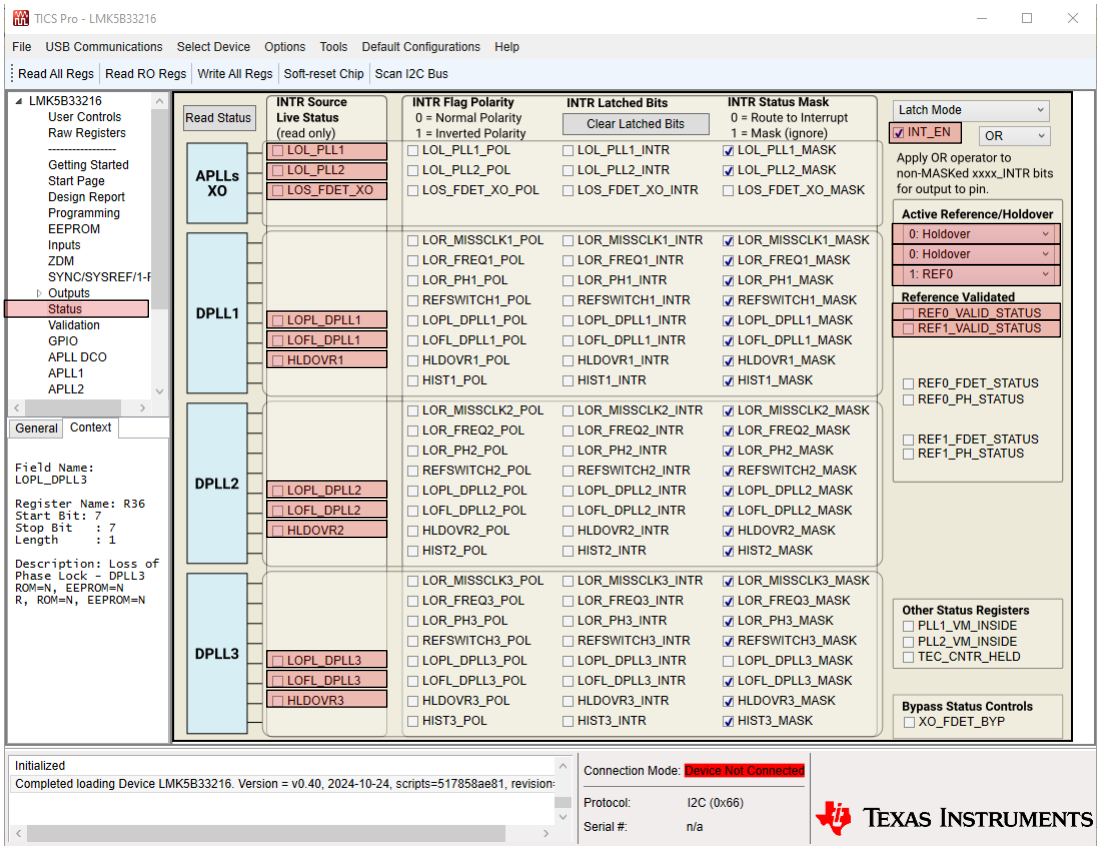

Figure 1-1. LMK5B33216 TICS Pro Status Page

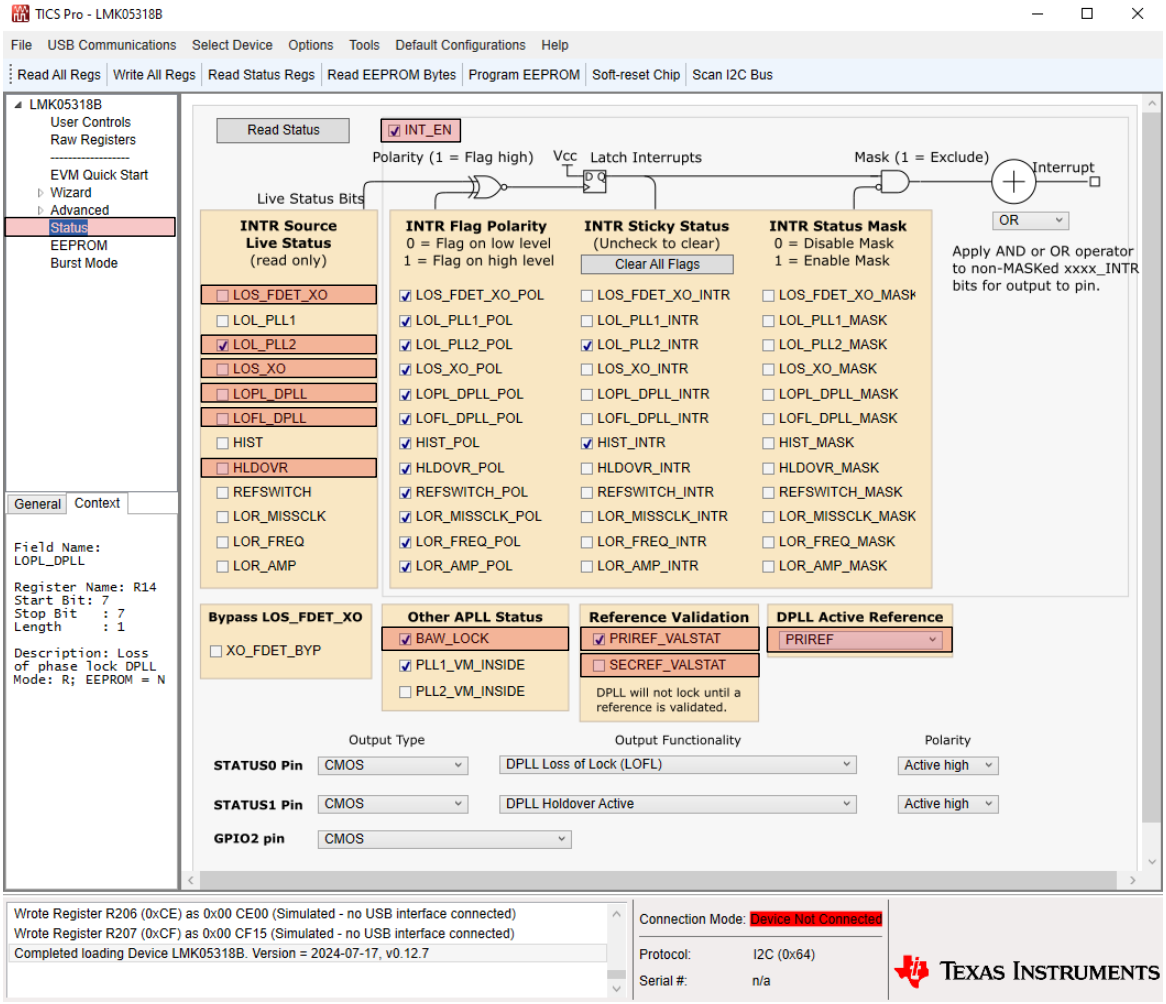

Figure 1-2. LMK05318B TICS Pro Status Page

1.2 TICS Pro Default Setting

The best practice when generating a new TICS Pro configuration file is to first load from a default configuration file which has been previously tested by TI. The default setting is found in the *Default Configurations* tab of each profile. Figure 1-3 shows the location of the default setting for the LMK5B33216 profile. Loading from the default settings allows the user to start in a known working state of register settings. After loading the default setting, complete each step in either the **Start Page** for LMK5B33216 or the **Wizard** for the LMK05318B as required for the user's application.

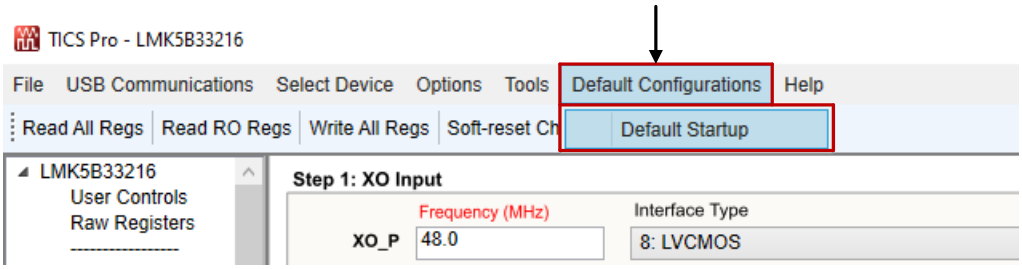

Figure 1-3. LMK5B33216 TICS Pro Default Configurations

Make sure to calculate the DPLL loop filter register settings by clicking on the *Run Script* button found at the bottom of the **Start Page** for LMK5B33216 (see Figure 1-4) or the end of the **Wizard** for LMK05318B (see Figure 1-5). The MATLAB Runtime R2015b v9.0 64-bit is required and the version must match exactly to use the script.

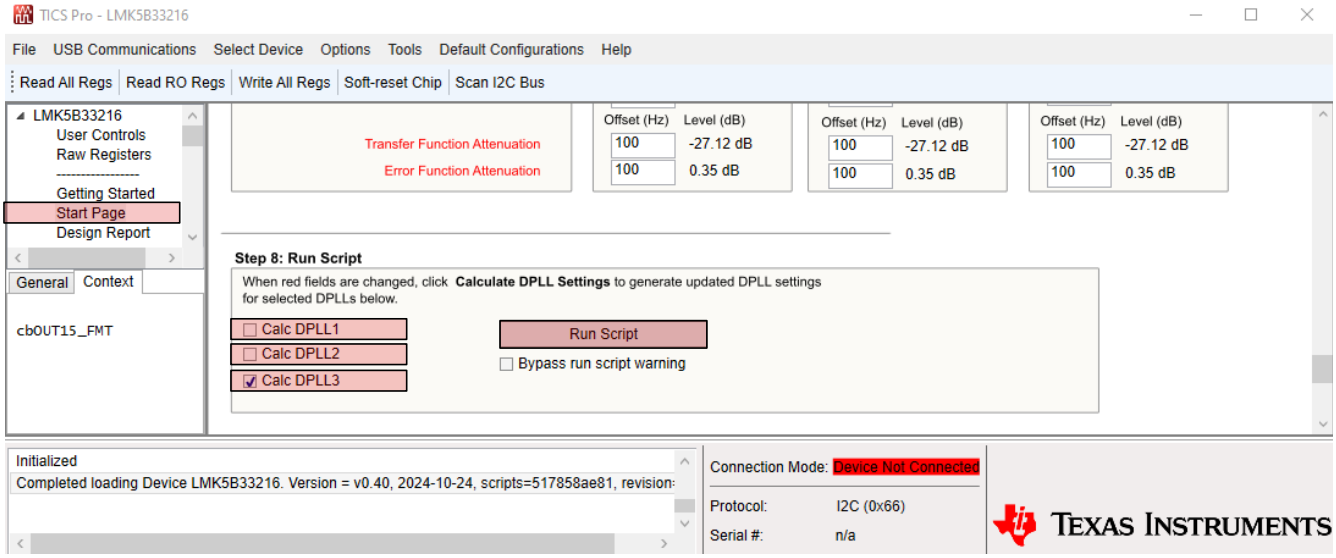

Figure 1-4. LMK5B33216 TICS Pro Run Script

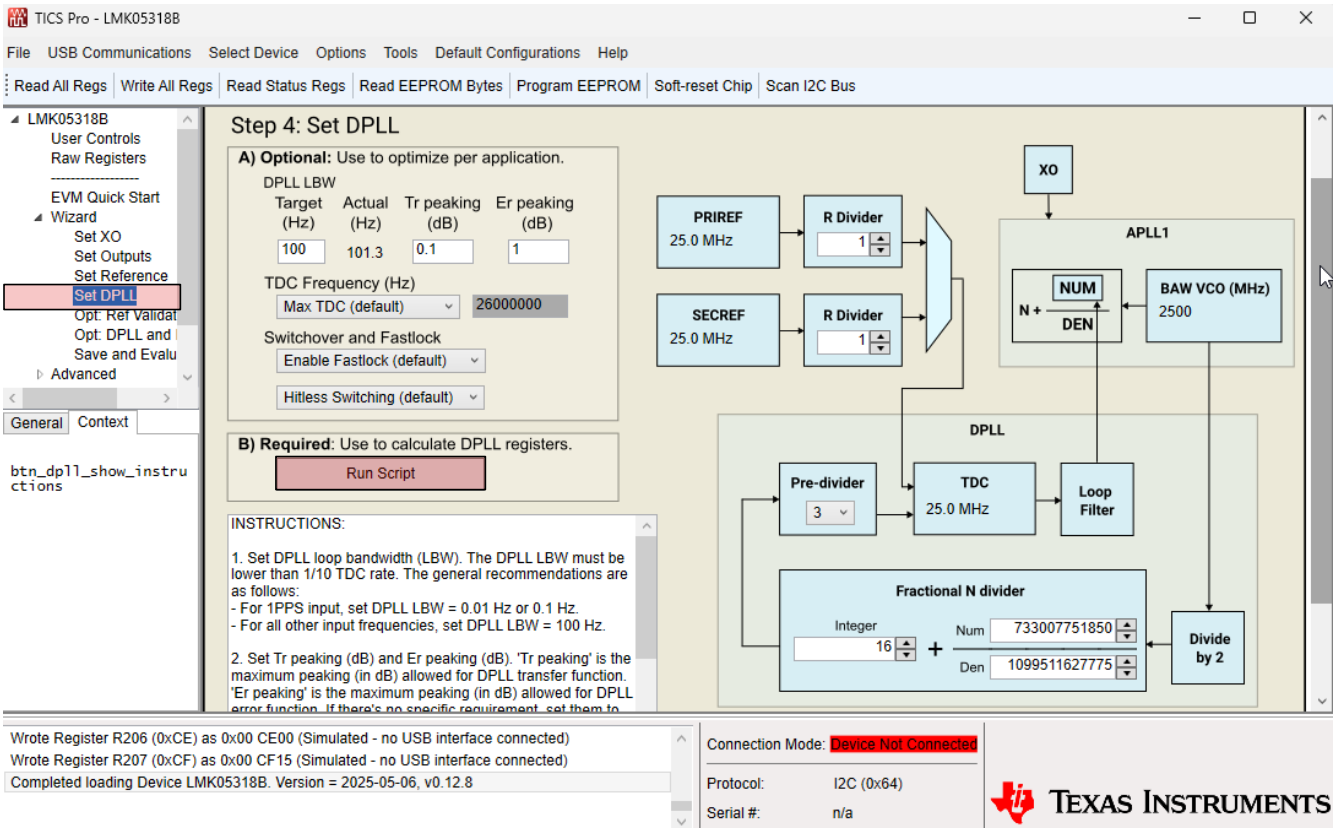

Figure 1-5. LMK05318B TICS Pro Run Script

For assistance creating a register configuration file for the LMK5B or LMK5CA family, refer to the walkthrough videos, TICS Pro LMK5Bxxxxx and LMK5CxxxxxxA profile overview and TICS Pro LMK5B/LMK5C start page tutorial. Additional videos are found in the *Videos* section of the TICS Pro Software website.

If the LMK device is not behaving as expected, even after using the settings generated by TICS Pro, then proceed to Debug Procedure: From Device Start-Up to Locked Output State.

2 Debug Procedure: From Device Start-Up to Locked Output State

The remaining sections provide step-by-step debugging guidance to reach DPLL lock. Before starting the debug procedure, review Start Here: Using TICS Pro for Configuration and Readback and confirm that the LMK supply pin voltage meets the data sheet specification.

Perform the debug steps in sequential order starting with Is the APLL Reference Valid?. Do not skip a section unless specified. If the issue persists after debugging, make a post on the TI public e2e forum, providing detailed debug findings to get further assistance.

2.1 Is the APLL Reference Valid?

The APLL cannot lock without a valid APLL reference and the DPLL cannot lock without a locked APLL. Therefore, the best practice is to check the APLL reference input clock. The APLL reference is commonly sourced from an external XO, TCXO, or OCXO. However, the LMK device also supports casaded APLL operation, using one APLL output as the APLL reference of another APLL. Figure 2-1 provides an example cascaded scheme. APLLx is the upstream APLL (cascaded source) and the APLL reference is the XO input. APLLy is the downstream APLL (cascaded APLL) and the APLL reference is the cascaded source.

Throughout the application note, the term "XO input" refers to any **external** source used as the APLL reference and the term "cascaded source" refers to any **internal** APLL source used as the APLL reference.

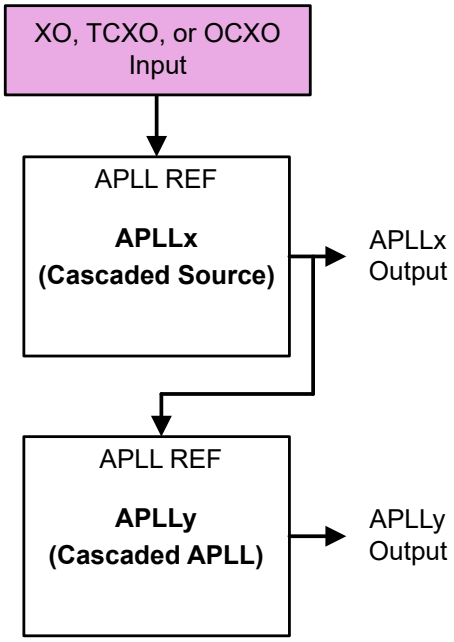

Figure 2-1. Cascaded APLL Reference Example

Proceed to Is the APLL Locked? if the APLL reference is considered valid but the APLL and DPLL are not locked. Otherwise, refer to Check Status of APLL Reference and Debug APLL Reference.

2.1.1 Check Status of APLL Reference

The APLL reference can be sourced externally from an XO input or internally from a cascaded source. For externally-sourced APLL references, the LMK device has an XO input detector that checks the validity of the XO input signal. The detector reports the validity of the signal through the "loss of frequency detection XO" bit. Refer to Table 2-1 for the bit definition. Table 2-2 shows the XO input frequency range and status registers for the LMK device families. For internally-sourced APLL references, use the APLL lock detector described in Is the APLL Locked? to check the validity of the cascaded source. The cascaded source is considered valid when the upstream APLL is locked.

Table 2-1. Loss of Frequency Detection XO - Status Bit Definition

BIT STATE	MEANING	DESCRIPTION
0	Desired result	The LMK device detects an XO input that is **within** the data sheet frequency range (with some applied margin).
1	Undesired result	The LMK device detects an XO input that is **outside** the data sheet frequency range (with some applied margin).

Table 2-2. Loss of Frequency Detection XO - Status Register

DEVICE FAMILY	XO FREQUENCY RANGE [MHz]	TICS PRO FIELD NAME	LOSS OF FREQUENCY DETECTION XO REGISTER
LMK05318B, LMK05318	10 to 100	LOS_FDET_XO	R13[4]
LMK5C	10 to 100	LOS_FDET_XO	R33[0]
LMK5B, LMK5CA	10 to 156.25	LOS_FDET_XO	R33[0]

2.1.2 Debug APLL Reference

If the status is undesired, run through the following debug steps:

1. **Check the external APLL reference by confirming the XO input termination.**

   Confirm that the correct input termination scheme is used for the XO input. Refer to the LMK device data sheet or the TICS Pro Software for guidance.

2. **Check the external APLL reference by probing the XO input pins.**

   Confirm the measured frequency and voltage swing meet the XO input requirements from the *Electrical Characteristics* table in the LMK device data sheet.

   Confirm the measured frequency and voltage swing are operating within the data sheet specification of the XO input. For example, if a 48MHz TCXO with a ± 1ppm frequency accuracy is used, then the probed frequency must be within 48MHz ± 1ppm. Use a frequency counter or phase noise analyzer to measure the frequency accurately instead of an oscilliscope which has inherent noise that causes inaccurate readings. Oscilliscopes can be used to check if the APLL is unlocked. However, due to the inaccuracy, oscilliscopes cannot be used to check if the APLL is properly frequency locked.

3. **Check the external APLL reference by buffering the XO input to an output.**

   For certain LMK devices, the XO input clock can be buffered out to an output channel to provide insight on what the APLL "sees" as the APLL reference. The buffered output clock is useful for checking the XO input frequency when the XO input pins are not readily available for probing. Measure the buffered output clock with a phase noise analyzer to observe any abnormal degradation in phase noise or jitter.

4. **Check the internal APLL reference by confirming the cascaded APLL settings.**

   Check that the cascaded source is configured properly and locked. Refer to Is the APLL Locked? for further APLL debug.

2.2 Is the APLL Locked?

At this stage in the debug, the XO input (or cascaded source) must be considered valid. The APLL automatically attempts to lock to the APLL reference after a power-on reset (POR), hardware reset, or software reset. The APLL reference provides the phase and frequency reference for the phase frequency detector (PFD). The PFD compares the APLL reference against the APLL VCO feedback clock to get a phase error. The charge pump converts the phase error into a correction current. The APLL loop filter then converts the correction current to a correction voltage, which tunes the VCO output frequency and keeps the VCO output synchronized to the APLL reference. The APLL reference, PFD, charge pump, APLL loop filter, and VCO are key contributors to getting a locked APLL.

Additionally, the path to DPLL lock must be considered and can be simplified into the four sequential states (listed below) with each state impacting the APLL output behavior. Refer to the LMK data sheet for more details on the DPLL lock states.

1. **Holdover: Free-Run**

   After APLL initialization, the free-run state is entered and the initial frequency accuracy of the APLL outputs is determined by the free-run tuning word register. The APLL reference determines the frequency accuracy and stability of the APLL outputs. If the free-run tuning word is not 0, then there is a frequency offset applied to the APLL output.

2. **DPLL Lock Acquisition**

   The DPLL continuously steers and updates the APLL numerator to achieve DPLL frequency and phase lock. The APLL output clocks transition from locking to the APLL reference to locking to the DPLL reference.

3. **DPLL Lock**

   During the DPLL lock state, the DPLL reference determines the frequency accuracy and stability of the APLL outputs, predominately below the DPLL loop bandwidth (LBW). DCO adjustments can impact the accuracy and stability depending on the DCO adjustment value and rate. In the DPLL lock state, the LMK device averages the DPLL reference frequency to accumulate history data (if the tuning word history is enabled).

4. **Holdover: History, Free-Run, Last Control Word**

   The holdover state is entered when the DPLL reference becomes invalid. Upon entering holdover, the frequency accuracy of the APLL outputs are determined by the tuning word history (averaged DPLL input frequency), the DPLL free-run register, or the last numerator word. The LMK device can successfully enter holdover with history if enough data is accumulated from the digital-loop filter (DLF) output during the programmed history averaging time. After an extended period of holdover, the frequency accuracy of the APLL outputs are determined by the APLL reference. During holdover, the stability of the APL output clocks changes because the DPLL reference no longer impacts below the DPLL LBW. Instead, the APLL reference alone determines the performance below the DPLL LBW.

If the APLL is locked but the DPLL is unlocked, proceed to Is the DPLL Reference Valid?. Otherwise, refer to Check Status of APLL Lock and Debug APLL Lock.

2.2.1 Check Status of APLL Lock

The LMK device has an APLLx lock detector that can determine the APLLx lock state through the "loss of lock APLLx" bit. The APLLx lock detector determines the lock state by a voltage threshold. This means the APLL lock status bit, as described in Table 2-3, can be cleared even though the APLL is not *frequency* locked. Refer toTable 2-4 for the loss of lock APLL status registers in the LMK device families.

Table 2-3. Loss of APLL Lock - Status Bit Definition

BIT STATE	MEANING	DESCRIPTION
0	Desired result	The APLLx tuning voltage is **within** the internally defined tuning voltage threshold.
1	Undesired result	The APLLx tuning voltage is **outside** the internally defined tuning voltage threshold.

Table 2-4. Loss of APLL Lock - Status Register

DEVICE FAMILY	TICS PRO FIELD NAME	LOSS OF LOCK APLL REGISTER		
		BAW APLL(1)	APLL2	APLL1
LMK05318B, LMK05318	LOL_PLLx	N/A	R13[3]	N/A
LMK5B, LMK5CA, LMK5C	LOL_PLLx	N/A	R33[2]	R33[3](2)

(1) The loss of APLL lock status bit is not available and/or not recommended for use.
(2) Column is not applicable for LMK5B12212 or LMK5C22212A.

For APLLs that have a BAW VCO (VCBO), use the BAW frequency lock detector to check the APLL lock status. The BAW frequency lock detector determines APLL lock by a user-programmed frequency threshold (relative to the XO input frequency). Therefore, the BAW lock detector is more precise than the voltage-based APLL lock detector. Note that the BAW detector can report unlock when the DPLL is locked if the frequency error between the DPLL reference and APLL reference exceeds the BAW lock threshold. Refer to Table 2-5 for the status bit definition of the BAW detector and Table 2-6 for the registers to configure.

Table 2-5. BAW Frequency Lock Detect - Status Bit Definition

BIT STATE	MEANING	DESCRIPTION
0	Undesired result	The BAW APLL reports frequency lock if the frequency error between the VCBO output and the XO input is **within** the BAW Frequency Lock Detect **lock threshold**.
1	Desired result	The BAW APLL reports unlock while the frequency error **exceeds** the **lock threshold** and becomes unlocked when the frequency error **exceeds** the **unlock threshold**.

Table 2-6. BAW Frequency Lock Detect - Status and Threshold Registers

DEVICE FAMILY	TICS PRO FIELD NAME	BAW LOCK STATUS REGISTER	BAW LOCK ENABLE REGISTER	BAW LOCK - LOCK THRESHOLD REGISTER(1)	BAW LOCK - UNLOCK THRESHOLD REGISTER(1)
LMK05318B, LMK05318	BAW_LOCK	R80[7]	R79[4] = 1	R80[6:0] to R81[7:0], R82 to R85, R86 to R89	R90 to R91, R92 to R95, R96 to R99
LMK5B, LMK5CA, LMK5C	LOFL_DPLL3 (APLL DLD)	R36[6]	R22[5] = 1 , R561[7] = 1(2), R528[7] = 1	R540 to R543, R528[6:0] to R529	R530 to R531

(1) Use TICS Pro Software to configure the lock and unlock threshold register settings.
(2) When R561[7] = 1, the R36[6] register reports the status of the BAW APLL digital lock detect (DLD), which is the BAW APLL frequency lock detector. When R561[7] = 0, the R36[6] register reports the status of the BAW DPLL DLD, which is the BAW DPLL loss of frequency lock detector. The APLL DLD is independent from the DPLL DLD.

2.2.2 Debug APLL Lock

If the status is undesired, run through the following debug steps.

1. **Disable the DPLL to confirm APLL lock.**

   Confirm the APLL is configured properly and can achieve lock by disabling the DPLL. If the DPLL is enabled, the DPLL can rail the APLL which can cause the frequency to unlock and make the debug more complicated. Re-enable the DPLL once APLL lock is validated.
2. **Recalibrate the APLL by issuing a software reset.**

   Each APLL VCO must be calibrated to achieve APLL lock and deliver the best phase noise performance. The VCO calibration establishes the best operating point within the VCO tuning range. After POR, hardware reset, or software reset, the VCO calibration is executed automatically. The APLL reference must be stable and considered valid before the start of calibration to provide a successful calibration and APLL lock. The output drivers are held in a muted state until the VCO calibration completes and the APLL locks. If you are not getting an APLL lock, try issuing a software reset after loading the register configuration. The best practice is to recalibrate the VCO whenever the PFD, charge pump, APLL loop filter, or APLL N divider registers are modified.
3. **Check the PFD inputs by probing the GPIO pins.**

   The PFD has two inputs: the APLL R divider path (from the XO input) and the APLL N divider path (from the VCO), as shown in Figure 2-2. The GPIO pins of the LMK devices can be configured as "APLL R divided by 2" and "APLL N divided by 2" signals which applies a divide by 2 to each path to reduce the GPIO output frequency. Configure the two GPIO pins per Table 2-7 and Table 2-8. Note that the APLL R divider is unavailable when R=1 (bypassed); in this case, probe the external APLL reference. Probe the two signals using an oscilliscope. Confirm that the two signals are present, have the same frequency, and are frequency-locked to each other. During a locked APLL state, the two signals are triggered to each other with no drifting observed on the oscilliscope.

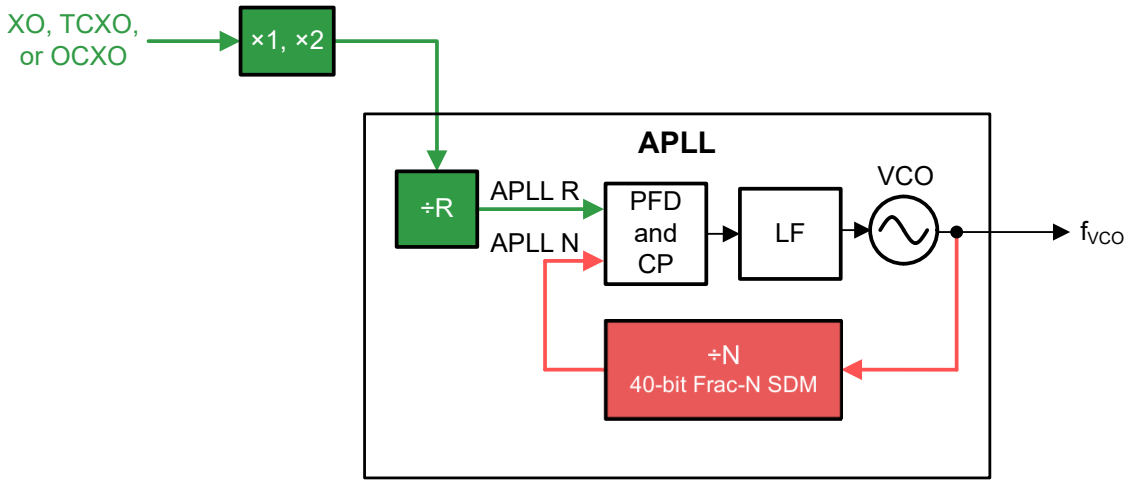

Figure 2-2. APLL R and APLL N Divider Paths

Table 2-7. APLL R Divided by 2 - Register Setting

DEVICE FAMILY	TICS PRO FIELD NAME	PIN NAME	APLL R DIVIDED BY 2 REGISTER SETTING		
			BAW APLL	APLL2	APLL1[2]
LMK05318B, LMK05318	PLLx R Divider, div-by-2	STATUS0	N/A[1]	R48[6:0] = 0x0F	N/A
		STATUS1	N/A[1]	R49[6:0] = 0x0F	N/A
LMK5B or LMK5CA	PLLx R-Divider Divided By 2	Required for any GPIOx	R70[0] = 1 R872[3] = 1	R70[0] = 1 R808[3] = 1	R70[0] = 1 R736[3] = 1
		GPIO0	R57[6:0] = 0x55	R57[6:0] = 0x54	R57[6:0] = 0x53
		GPIO1	R58[6:0] = 0x55	R58[6:0] = 0x54	R58[6:0] = 0x53
		GPIO2	R59[6:0] = 0x55	R59[6:0] = 0x54	R59[6:0] = 0x53

(1) APLL R divider signal is not available. Probe the external APLL reference instead and trigger the oscilliscope on APLL N-divider path (the lower frequency).
(2) Column is not applicable for LMK5B12212 or LMK5C22212A.

Table 2-8. APLL N Divided by 2 - Register Setting

DEVICE FAMILY	TICS PRO FIELD NAME	PIN NAME	APLL N DIVIDED BY 2 REGISTER SETTING		
			BAW APLL	APLL2	APLL1[1]
LMK05318B, LMK05318	PLLx N Divider, div-by-2	STATUS0	R48[6:0] = 0x05	R48[6:0] = 0x08	N/A
		STATUS1	R48[6:0] = 0x05	R49[6:0] = 0x08	N/A
LMK5B or LMK5CA	PLLx N-Divider, Divided By 2	Required for any GPIOx	R70[0] = 1 R872[3] = 1	R70[0] = 1 R808[3] = 1	R70[0] = 1 R736[3] = 1
		GPIO0	R57[6:0] = 0x52	R57[6:0] = 0x51	R57[6:0] = 0x50
		GPIO1	R58[6:0] = 0x52	R58[6:0] = 0x51	R58[6:0] = 0x50
		GPIO2	R59[6:0] = 0x52	R59[6:0] = 0x51	R59[6:0] = 0x50

(1) Column is not applicable for LMK5B12212 or LMK5C22212A.

4. **Check the APLL charge pump settings.**

   Try using the charge pump register settings provided by the Default Configuration tab in TICS Pro Software. Too much charge pump current can oversaturate the APLL resulting in unstable APLL outputs. Too little charge pump current can cause the APLL to unlock and become an open-loop.

5. **Check the loop filter settings.**

   Try using the loop filter register settings provided by the Default Configuration tab in TICS Pro Software. Additionally, confirm the external capacitor applied to the LFx pin is per data sheet recommendation. The external capacitor is the "C2" for the respective APLLx loop filter.

6. **Check the APLL tuning voltage by probing the LFx pin.**

   Measure the voltage of the LFx pin over time to observe the trend. During normal operation, the voltage varies over time due to variation in the APLL reference frequency (when the APLL is locked), DPLL reference frequency (when the DPLL is locked), and temperature. The typical tuning voltage is somewhere in the middle. Too low or too high can mean the APLL has railed to an incorrect frequency. If the DPLL is enabled, the tuning voltage can be railed (< 0.1V or > 2.4V) because the DPLL drives the APLL to rail off. A reading of 0V is not expected behavior and can be due to a damaged or improperly soldered device. If the voltage is not changing, but is not 0V, contact TI by posting on the public TI E2E forum for further assistance.

2.3 Is the DPLL Reference Valid?

The DPLL cannot lock unless at least one DPLL reference is considered valid and selected by the DPLL. The DPLL reference clock is referred to as the xxxREF, REFx, or INx inputs. In some LMK devices, a cascaded source can be routed to a DPLL input to serve as the DPLL reference.

If the DPLL is considered valid but the DPLL is unlocked, proceed to Is the DPLL Selecting a Reference?. Otherwise, refer to Check Status of DPLL Reference Validation and Debug DPLL Reference Validation.

2.3.1 Check Status of DPLL Reference Validation

The LMK device has a DPLL reference validation detector that checks the validity of the **external** DPLL reference input clock. The validity of the external DPLL reference is determined by the enabled detectors, outlined in Table 2-9. The status is reported by the "DPLL reference validation" bit, described in Table 2-10. Refer to Table 2-11 for the DPLL reference validation status registers in the LMK device families.

Note that the reference validation detectors do not apply to DPLL inputs that are sourced internally from a cascaded source. When using a cascaded source as the input, the reference validation bit is always 1 unless the upstream APLL is disabled or unlocked.

Table 2-9. Reference Validation Settings

DEVICE FAMILY	REFERENCE VALIDATION SETTING	DESCRIPTION
LMK05318B, LMK05318, LMK5B, LMK5CA, LMK5C	Validation timer	The setting is true when the DPLL reference remains valid (across all enabled detectors) for the time defined by the validation timer.
LMK05318B, LMK05318	Amplitude detector	The setting is true when the voltage swing of the DPLL reference is higher than the defined amplitude threshold.
LMK05318B, LMK05318, LMK5B, LMK5CA, LMK5C	Frequency detector	The setting is true when the frequency error between DPLL reference and the XO input is within the valid threshold. If the frequency exceeds the invalid threshold, then the setting is false. Disable for 1PPS inputs.
LMK05318B, LMK05318, LMK5B, LMK5CA, LMK5C	Early (runt pulse) and late (missing pulse) window detectors	The setting is true when the period of the DPLL reference is less than the missing clock threshold and greater than the early clock threshold. Disable for 1PPS inputs.
LMK05318B, LMK05318, LMK5B, LMK5CA, LMK5C	Phase valid monitor (1PPS phase detector)	The setting is true when the next clock edge of the DPLL reference is within the expected clock edge defined by the detector threshold. Enable only for 1PPS inputs.

Table 2-10. DPLL Reference Validation - Status Bit Definition

BIT STATE	MEANING	DESCRIPTION
0	Undesired result	The DPLL reference valid bit is cleared when the DPLL reference **does not meet** the requirements of the enabled reference validation settings. At least one of the enabled reference detectors is failing (false).
1	Desired result	The DPLL reference valid bit is set when the DPLL reference **meets** the requirements of the enabled reference validation settings. All of the enabled reference detectors must be passing (true).

Table 2-11. DPLL Reference Validation - Status Register

DEVICE FAMILY	TICS PRO FIELD NAME	DPLL REFERENCE VALID REGISTER
LMK05318B, LMK05318	PRIREF_VALSTAT	R411[2]
	SECREF_VALSTAT	R411[3]
LMK5B, LMK5CA, LMK5C	REF0_VALID_STATUS	R50[0]
	REF1_VALID_STATUS	R50[1]
	REF2_VALID_STATUS(1)	R50[2]
	REF3_VALID_STATUS(1)	R50[3]

(1) Only available on the 4-input devices.

2.3.2 Debug DPLL Reference Validation

If the status is undesired, run through the following debug steps.

1. **Check the DPLL reference input termination.**

   Confirm the correct input termination scheme is used (whether external or internal to the device). Test with different internal termination register settings. Refer to the LMK device data sheet or TICS Pro for guidance.

2. **Check the DPLL input signals by probing the external DPLL input pins.**

   Confirm the measured frequency and voltage swing meet the DPLL reference input requirements from the *Electrical Characteristics* table in the LMK device data sheet. Measure the frequency using a frequency counter or phase noise analyzer. Avoid using an oscilliscope for frequency measurements as the inherent oscilliscope noise does not provide accurate readings. Measure the voltage swing using an oscilliscope.

3. **Check the DPLL inputs by probing the GPIO pins.**

   Use the GPIO pins to view the DPLL inputs when probing the DPLL input pins is not easily possible. Configure the GPIOs as the "Reference Monitor Output Divided by 2" to apply a divide by 2 to the DPLL input path and reduce the GPIO output frequency. Refer to Table 2-12.

Table 2-12. Reference Monitor R Divided by 2 - Register Setting

DEVICE	TICS PRO FIELD NAME	PIN NAME	REFERENCE MONITOR OUTPUT DIVIDED BY 2 REGISTER SETTING			
			REF0 or PRIREF	REF1 or SECREF	REF2(2)	REF3(2)
LMK05318B	STATx_SEL (xxxREF Monitor Divider Output, div-by-2)	STATUS0	R48[6:0] = 0x0D	R48[6:0] = 0x0E	N/A	N/A
		STATUS1	R49[6:0] = 0x0D	R49[6:0] = 0x0E	N/A	N/A
LMK5B or LMK5CA(1)	GPIOx_SEL (REFx Monitor Divider Output Divided by 2)	Required for any GPIOx	R70[0] = 1			
		GPIO0	R57[6:0] = 0x59	R57[6:0] = 0x5A	R57[6:0] = 0x5B	R57[6:0] = 0x5C
		GPIO1	R58[6:0] = 0x59	R58[6:0] = 0x5A	R58[6:0] = 0x5B	R58[6:0] = 0x5C
		GPIO2	R59[6:0] = 0x59	R59[6:0] = 0x5A	R59[6:0] = 0x5B	R59[6:0] = 0x5C

(1) To use the REF monitor on the GPIO pin, the DPLLx_REFy_AUTO_PRTY register for the respective REFy must be greater than 0. This is true for any input mux option selected (Auto Revertive, Auto Non-Revertive, Manual with Holdover, or Manual with Holdover).
(2) Only available on the 4-input devices.

4. **Identify the failing reference validation detector.**

   The DPLL reference validation can fail when at least one detector is not properly configured. To identify the failing detector, disable all of the reference valiation settings. Enable a validation setting one-by-one, checking the status bit after each enable until the failing detector is identified. Refer to the steps below for an example debug scenario. Additionally, try minimizing the validation timer to determine if the issue is intermittent and/or because the DPLL reference fails a longer validation window.

a. From readback, the DPLL reference valid bit = 0.
b. Disable all reference validation settings.
c. Enable only the validation timer first.
d. From readback, the DPLL reference valid bit = 1. This means the validation timer is not the issue.
e. Enable also the early and late window detectors.
f. From readback, the DPLL reference valid bit = 1. This means the early and late window detectors are not the issue.
g. Enable also the frequency detector.
h. From readback, the DPLL reference valid bit = 0. This means the frequency detector is the issue. The frequency thresholds need to be adjusted to account for the reference and XO frequency error.
i. Increase the frequency detector valid and invalid thresholds.
j. From readback, the DPLL reference valid bit = 1.
k. The DPLL reference is valid with all the desired validation settings enabled.

2.4 Is the DPLL Selecting a Reference?

The DPLL reference selection depends on two factors:

1. The presence of a valid input.
2. The reference input mux selection registers.

At this stage in the debug, Factor (1) must be true. Factor (2) comprises of four mux options: Auto Revertive, Auto Non-Revertive, Manual with Holdover, and Manual with Holdover. Each mux option determines how each DPLL reference is selected. Refer to the LMK data sheet for more details on the DPLL reference selection mux.

If the DPLL is selecting a reference but the DPLL is unlocked, proceed to Is the DPLL Frequency Locked?. Otherwise, see Check Status of DPLL Reference Selection and Debug DPLL Reference Selection.

2.4.1 Check Status of DPLL Reference Selection

The LMK device has a DPLL reference selector that chooses one valid DPLL reference to use for DPLL lock. The selected DPLL input is reported through the "DPLL Reference Selection" status bits as described in Table 2-13. Table 2-11 shows the DPLL reference selection status registers for the LMK device families.

Table 2-13. DPLL Reference Selection - Status Bits Definition

BIT[n:0] STATE	MEANING	DESCRIPTION
0	Undesired result	The DPLL reference valid bit is cleared when the DPLL is in holdover. There is not one valid reference input and/or the DPLL register settings are incorrect.
> 0	Desired result	The DPLL reference valid bit is greater than 0 when the DPLL is locked to the input clock specified by the status value.(1) There is at least one valid reference input.

(1) Refer to the LMK device programming manual for the full list of options.

Table 2-14. DPLL Reference Selection - Status Register

DEVICE FAMILY	TICS PRO FIELD NAME	DPLL REFERENCE SELECTION REGISTER		
		BAW DPLL	DPLL2	DPLL1(2)
LMK05318B, LMK05318	DPLL_REFSEL_STAT	R167[1:0]	N/A	N/A
LMK5B, LMK5CA, LMK5C	DPLLx_REFSEL_STAT	R527[5:0]	R377[5:0]	R227[5:0]

2.4.2 Debug DPLL Reference Selection

If the status is undesired, run through the following debug steps.

1. **Check the DPLL reference input mux setting.**

   Confirm the mux option is configured properly. For example, for Auto-Revertive, Auto Non-Revertive, and Manual Fallback modes, make sure that at least one input is assigned a priority through the *DPLLx_REFy_AUTO_PRTY* register.

   Try forcing a selection by manually selecting one input using the Manual Holdover mode. Make sure to specify the selected input through either register or pin setting. If the DPLL remains in Holdover even though the reference is valid, contact TI by posting on the public TI E2E forum for further assistance.

2. **Create a new configuration.**

   The DPLL input registers can be incorrect. Try generating a new configuration using TICS Pro.

2.5 Is the DPLL Frequency Locked?

At this stage in the debug, the APLL and the DPLL references are valid input signals and the APLL is locked. The next step is to check the frequency lock state of the DPLL. If the DPLL registers are misconfigured, then the DPLL can rail the APLL causing the output clocks to be off in frequency (with a frequency offset applied that is within the VCO frequency pulling range). When the DPLL is frequency locked, the VCO output and output clocks are frequency locked to the selected DPLL input. The locked output clocks slew at a rate defined by the DLF until the desired output frequency is reached (near 0ppm error).

The DPLL frequency lock state can be identified by checking the respective status bit or debugging the time-to-digital converter (TDC). The TDC compares the phase error between the DPLL reference clock after the R divider (DPLL R path) and the respective VCO output clock after the feedback divider (DPLL FB path). The TDC phase error is generated as a digital correction word and is filtered by the DLF. The DLF output adjusts the APLL N divider numerator to pull the VCO frequency into lock with the DPLL reference. Figure 2-3 shows the block diagram view of the DPLL R and DPLL FB paths to the TDC.

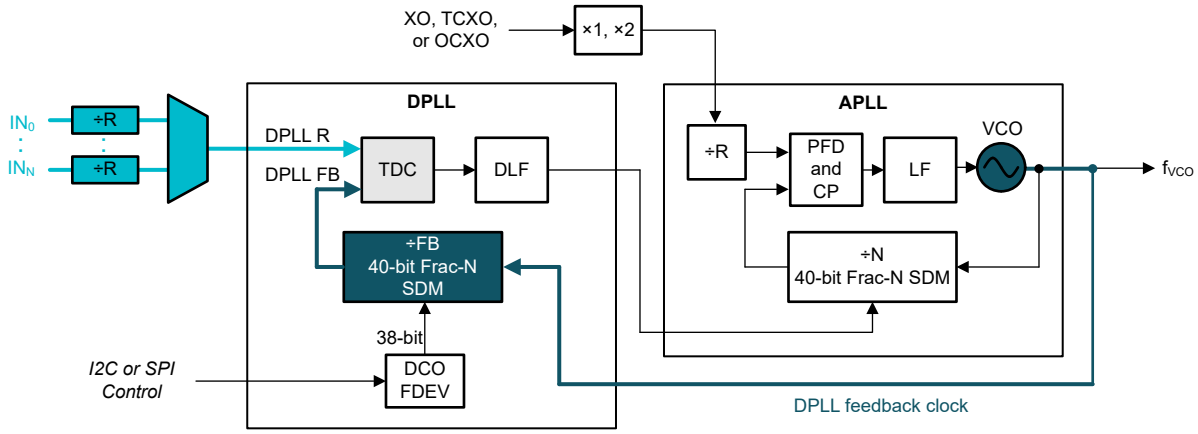

Figure 2-3. DPLL R and DPLL FB Paths

If the DPLL is frequency locked but not phase locked to the DPLL input, proceed to Is the DPLL Phase Locked?. Otherwise, refer to Check Status of DPLL Frequency Lock and Debug DPLL Frequency Lock.

2.5.1 Check Status of DPLL Frequency Lock

The LMK device has a DPLL frequency lock detector that provides the lock state through the "loss of DPLL frequency lock" or "LOFL" bit. The DPLL LOFL bit definition is described in Table 2-15 and the status register is listed in Table 2-16. The registers to configure DPLL frequency lock detector are specified in Table 2-17.

Table 2-15. Loss of DPLL Frequency Lock - Status Bit Definition

BIT STATE	MEANING	DESCRIPTION
0	Desired result	The DPLL reports frequency lock if the frequency error between the VCO output and the DPLL input is **within** the LOFL **lock threshold**.
1	Undesired result	The DPLL reports unlocked while the frequency error **exceeds** the LOFL **lock threshold** and becomes unlocked when the frequency error **exceeds** the LOPL **unlock threshold**.

Table 2-16. Loss of DPLL Frequency Lock - Status Register

DEVICE FAMILY	TICS PRO FIELD NAME	LOSS OF DPLL FREQUENCY LOCK REGISTER		
		BAW DPLL	DPLL2	DPLL1
LMK05318B, LMK05318	LOFL_DPLL	R80[7]	N/A	N/A
LMK5B, LMK5CA, LMK5C	LOFL_DPLLx	R36[6]	R35[6]	R34[6](1)

(1) Column is not applicable for LMK5B12212 or LMK5C22212A.

Table 2-17. Loss of DPLL Frequency Lock - Enable Setting

DEVICE FAMILY	LOFL ENABLE REGISTER SETTING		
	BAW DPLL	DPLL2	DPLL1
LMK05318B, LMK05318	R79[4] = 1	N/A	N/A
LMK5B, LMK5CA, LMK5C	R22[5] = 1, R528[7] = 1, R561[7] = 0(1)	R22[3] = 1, R378[7] = 1	R22[1] = 1, R228[7] = 1(2)

(1) When R561[7] = 1, the R36[6] register reports the status of the BAW APLL digital lock detect (DLD), which is the BAW APLL frequency lock detector. When R561[7] = 0, the R36[6] register reports the status of the BAW DPLL DLD, which is the BAW DPLL loss of frequency lock detector. The APLL DLD is independent from the DPLL DLD.
(2) Column is applicable for LMK5B12212 or LMK5C22212A.

Table 2-18. Loss of DPLL Frequency Lock - Threshold Registers

DEVICE FAMILY	LOFL - LOCK THRESHOLD REGISTER(1)			LOFL - UNLOCK THRESHOLD REGISTER(1)		
	BAW DPLL	DPLL2	DPLL1	BAW DPLL	DPLL2	DPLL1
LMK05318B, LMK05318	R80[6:0] to R81, R82 to R85, R86 to R89	N/A	N/A	R90 to R91, R92 to R95, R96 to R99	N/A	N/A
LMK5B, LMK5CA, LMK5C	R536 to R539, R540 to R543, R528[6:0] to R529	R386 to R389, R390 to R393, R378[6:0] to R379	R236 to R239, R240 to R243, R228[6:0] to R229(2)	R530[6:0] to R531	R380[6:0] to R381	R230[6:0] to R231(2)

(1) Use TICS Pro Software to configure the lock and unlock threshold register settings.
(2) Column is not applicable for LMK5B12212 or LMK5C22212A.

2.5.2 Debug DPLL Frequency Lock

1. **Calculate the expected PFD, TDC, and VCO frequencies based on the DPLL and APLL registers.**

   Confirm the numerator, denominator, N divider, pre-divider and post-divider settings are correct for the DPLL and APLL. Refer to the LMK data sheet for the PFD, TDC, and VCO equations.
2. **Check the LOFL lock and unlock thresholds.**

   The LOFL status bit can falsely report an unlocked state due to improper threshold settings. Try widening the frequency lock threshold (see Table 2-18) by 10ppm to see if the LOFL status bit clears. Use TICS Pro to configure the threshold settings.
3. **Debug the TDC inputs by adding a frequency offset.**

   Confirm the DPLL is actually locked to the DPLL reference and not misreporting LOFL=0 by applying a small frequency offset (such as 1ppm to 5ppm) to **either** the APLL reference **or** the DPLL reference. Observe the impact on the output frequency. When the DPLL is locked and properly configured, the outputs have the same frequency accuracy as the DPLL reference. When the DPLL is not locked, the outputs have the same frequency accuracy as the APLL reference. Therefore, a frequency offset applied to the DPLL reference results in a frequency offset applied to the output clocks by the same amount when the DPLL is locked.

   The following steps provide an example debug scenario with the results summarized in Table 2-19. In the example, the XO input is used as the APLL reference and the INx input is used as the DPLL reference. The "programmed frequency" refers to the configured register settings and the "expected output frequency" is the output frequency that is expected when measured.

   a. Use TICS Pro to configure the register settings for XO = 48MHz, INx = 156.25MHz, OUTx=156.25MHz.
   b. From readback, DPLL LOPL = 1.
   c. Route the output clock to a frequency counter or phase noise analyzer. Record the output frequency.
   d. Apply a 5ppm offset to the DPLL reference input by setting the frequency to 156.25MHz + 5ppm.
   e. Record the output frequency.
      i. Is there a 10ppm offset added to the output frequency? If yes, the DPLL is locked and the LOPL thresholds need to be adjusted. If no, contact TI for further support.

Table 2-19. Applied Frequency Offset Debug Example

PROGRAMMED FREQUENCY [MHz]			APPLIED FREQUENCY [MHz]		EXPECTED OUTPUT FREQUENCY [MHz]	
XO	INx	OUTx	XO	INx	OUTx, WHEN DPLL IS LOCKED	OUTx, WHEN DPLL IS UNLOCKED
48	25	156.25	48	25	156.25	156.25
48	25	156.25	48.00024	25	156.25	156.2507
48	25	156.25	48	25.000125	156.2507	156.25

4. **Debug the TDC inputs by reading the APLL Numerator.**

   The LMK device offers a status register, as seen in Table 2-20, to monitor the live APLL numerator value. When the DPLL is locked, the DPLL makes updates to the APLL numerator (at a rate defined by the TDC) until a 0ppm error between the DPLL reference and VCO output is attained. Since the DPLL reference has noise and jitter, the DPLL continuously makes corrections to the APLL numerator trying to achive a 0ppm error. Checking the live APLL numerator status for DPLL lock is similar to checking the tuning voltage (LFx pin) for APLL lock.

Use the following pseudocode to get the frequency error (ppb) between the expected and measured VCO frequency, where APLL_NUM_STAT is the APLL numerator status register readback value and APLL_N_DIV is the APLL N divider register readback value. The APLL N divider register address is listed in Table 2-21. Graph the results over the time to get a time interval error (TIE) plot. In a stable DPLL locked state, the frequency error is near 0ppb and does not sporadically drift in an uncontrolled manner.

```
get_APLL_NUM_STAT_as_ppb (phase_detect_freq, expected_vco_freq){
    numStatVal = Read_Reg("APLL_NUM_STAT")
    fract = numStatVal / (2^40)
    n_div = Read_Reg("APLL_N_DIV")
    meas_vco_freq = phase_detect_freq * (n_div + fract)
    ppb_error = 1e9 * ((meas_vco_freq - expected_vco_freq) / vco_freq)
    return ppb_error
}
```

Table 2-20. Live APLL Numerator - Status Register

DEVICE FAMILY	TICS PRO FIELD NAME	LIVE APLL NUMERATOR REGISTER		
		BAW APLL	APLL2	APLL1
LMK05318B, LMK05318	PLL1_NUM_STAT	R127 to R123	N/A	N/A
LMK5B, LMK5CA, LMK5C	APLLx_NUM_STAT	R862 to R858	R799 to R795	R729 to R725(1)

(1) Column is not applicable for LMK5B12212 or LMK5C22212A.

Table 2-21. APLL N Divider - Register

DEVICE FAMILY	TICS PRO FIELD NAME	APLL N DIVIDER REGISTER		
		BAW APLL	APLL2	APLL1
LMK05318B	PLL1_NDIV	R109 to R108	N/A	N/A
LMK5B or LMK5CA	PLLx_NDIV	R850 to R849[0]	R787 to R786[0]	R717 to R716[0](1)

(1) Column is not applicable for LMK5B12212 or LMK5C22212A.

2.6 Is the DPLL Phase Locked?

Once the DPLL is frequency locked, the DPLL reports phase lock when the phase error between the VCO output and DPLL input is

Once the DPLL is frequency locked, the DPLL attempts to phase lock at a rate defined by the DPLL loop bandwidth until phase error is reached between the VCO output and DPLL input. While the phase error drifts, there is also a minor frequency error, but the frequency error is not significant enough to cause the DPLL frequency to unlock.

If the DPLL is phase locked and the device performs as expected, then the network synchronizer has been debugged. Otherwise, see Check Status of DPLL Phase Lock and Debug DPLL Phase Lock.

2.6.1 Check Status of DPLL Phase Lock

The LMK device has a DPLL phase lock detector that determines the lock state through the "loss of DPLL phase lock" or "LOPL" bit. The DPLL LOPL bit is a status flag that is used to monitor the phase lock between the input and outputs and is described in Table 2-22. The LOPL status register is listed in Table 2-23.

Table 2-22. Loss of DPLL Phase Lock - Status Bit Definition

BIT STATE	MEANING	DESCRIPTION
0	Undesired result	The DPLL reports phase lock if the phase error between the VCO output and the DPLL input is **within** the LOPL **lock threshold**.
1	Desired result	The DPLL reports unlock while the phase error **exceeds** the LOPL **lock threshold** and becomes unlocked when the phase error **exceeds** the LOPL **unlock threshold**.

Table 2-23. Loss of DPLL Phase Lock - Status Register

DEVICE FAMILY	TICS PRO FIELD NAME	LOSS OF DPLL PHASE LOCK REGISTER		
		BAW DPLL	DPLL2	DPLL1
LMK05318B, LMK05318	LOPL_DPLL	R14[7]	N/A	N/A
LMK5B, LMK5CA, LMK5C	LOPL_DPLLx	R36[7]	R35[7]	R34[7](1)

Table 2-24. Loss of DPLL Phase Lock - Threshold Registers

DEVICE FAMILY	LOPL - LOCK THRESHOLD REGISTER(1)			LOPL - UNLOCK THRESHOLD REGISTER(1)		
	BAW DPLL	DPLL2	DPLL1	BAW DPLL	DPLL2	DPLL1
LMK05318B, LMK05318	R301[5:0]	N/A	N/A	R302[5:0]	N/A	N/A
LMK5B, LMK5CA, LMK5C	R603[5:0]	R453[5:0]	R303[5:0](2)	R604[5:0]	R454[5:0]	R304[5:0](2)

(1) Use TICS Pro Software to configure the lock and unlock threshold register settings.
(2) Column is not applicable for LMK5B12212 or LMK5C22212A.

2.6.2 Debug DPLL Phase Lock

1. **Confirm lock by phase *synchronization*, not phase offset.**

   The DPLL is considered phase locked when the APLL output clocks track the phase of the selected DPLL input. This is also known as phase synchronization. When phase-locked, the input and output clocks can be triggered to each other on an oscilliscope. There can be an offset present between the locked outputs and the selected DPLL input. The offset is programmable depending on the LMK device. Typically, the offset is small and subject to change after POR, software reset, hardware reset, or a hitless switch. Use zero-delay mode when a deterministic phase offset is required.
2. **Check the LOPL lock and unlock thresholds.**

   The LOPL status bit can falsely report an unlocked state due to improper threshold settings. A noisy DPLL reference or APLL reference with a narrow DPLL loop bandwidth can cause large phase variations. To account for such variations, widen the phase lock thresholds to allow the device to be considered phase locked. Use TICS Pro to configure the threshold settings.
3. **Debug the TDC inputs by reading the TDC registers.**

   The miminum and maximum TDC debug registers can be readback as listed in Table 2-25. Configure the TDC debug register as specified in Table 2-26.

Table 2-25. TDC Minimum and TDC Maximum - Status Register

DEVICE FAMILY	TDC MIN REGISTER	TDC MAX REGISTER
LMK5B, LMK5CA	R191 to R198	R199 to R206

Table 2-26. TDC Minimum and TDC Maximum - Register Setting

DEVICE FAMILY	TDC MIN AND MAX REGISTER SETTING		
	BAW DPLL	DPLL2	DPLL1
LMK5B, LMK5CA	R207[1:0] = 0x2, R207[6] = 1, R208[3:0] = 0x1	R207[1:0] = 0x1, R207[6] = 1, R208[3:0] = 0x1	R207[1:0] = 0x0, R207[6] = 1, R208[3:0] = 0x1(1)

(1) Column is not applicable for LMK5B12212 or LMK5C22212A.

Use the pseudocode below to capture the phase error measured by the TDC of the device. Note that the readback value is a digital number (unitless). The readback rate of the minimum and maximum TDC registers must be no faster than the configured TDC rate. If the readback rate is faster than the TDC rate, a 0 value is readback. Graph the results over the time. In a stable DPLL locked state, the TDC reaches a near 0 value and does not sporadically change in an uncontrolled manner.

```
dpll_config_R207 = # DEFINED BY ENABLE SETTING TABLE

def setup_tdc_readback():
    WriteReg(R208,0x01)
    WriteReg(R207, dpll_config_R207)

def get_tdc_min_max():
    # Read MSB to LSB.  Min/max reset after R206 is read.
    rawTDC_MIN = ReadReg(R191, R198)
    rawTDC_MAX = ReadReg(R199, R206)

    val = sum([(2**(56-(8*i)))* rawTDC_MIN[i] for i in range(8)])
    if  rawTDC_MIN[0] > 127:
        val ^= 0xffffffffffffffff
        val &= 0xffffffffffffffff
        val += 1
        val *= -1
    TDC_MIN = val

    val = sum([(2**(56-(8*i)))*rawTDC_MAX[i] for i in range(8)])
    if rawTDC_MAX[0] > 127:
        val ^= 0xffffffffffffffff
        val &= 0xffffffffffffffff
        val += 1
        val *= -1
    TDC_MAX = val

    return (TDC_MIN, TDC_MAX)
```

4. **Debug the TDC inputs by DPLL R and DPLL FB dividers.**

One way to check the inputs to the TDC is by configuring the GPIO status pins as the "DPLL R divided by 2" and "DPLL FB divided by 2" signals. The two signals represent the DPLL R divider and DPLL FB divider paths with a divide by 2 on each signal. Route the two TDC input signals to a scope to determine the DPLL lock state. If the DPLL FB clock is a drifting clock compared to the DPLL R clock, then the DPLL settings (such as for the TDC and DLF) is possibly not configured properly. Contact TI as the next step by posting on the public TI E2E forum, include the TICS Pro configuration file (.tcs) with oscilloscope screenshots capturing the FB and R waveforms. Otherwise, if the DPLL FB clock is triggered to the DPLL R clock and not drifting, then the DPLL is locked and properly configured. The LOPL flag can be false reporting due to improper LOPL threshold settings. Increase the lock and unlock thresholds until the LOPL flag clears.

The equations for each signal are represented by Equation 1 and Equation 2.

$$f_{DPLL\ R\ divided\ by\ 2} = \frac{f_{INx}}{2 \times DIV_{DPLL,\ R}} \quad (1)$$

$$f_{DPLL\ FB\ divided\ by\ 2} = \frac{f_{INx}}{2 \times DIV_{DPLL,\ FB}} \quad (2)$$

For example, the "DPLL R divide by 2" signal frequency is 0.625MHz for a DPLL reference frequency of 156.25MHz and an R divider value of 125 as shown in Equation 3.

$$f_{DPLL\ R\ divided\ by\ 2} = \frac{156.25MHz}{2 \times 125} = 0.625MHz \quad (3)$$

The registers for the DPLL R divided by 2 and FB divided by 2 signals are listed in Table 2-27 and Table 2-28.

Table 2-27. DPLL R Divided by 2 - Register Setting

DEVICE FAMILY	TICS PRO FIELD NAME	PIN NAME	DPLL R DIVIDED BY 2 REGISTER SETTING		
			BAW DPLL	DPLL2	DPLL1(1)
LMK05318B, LMK05318	DPLL R Divider, div-by-2	STATUS0	R48 = 0x40	N/A	N/A
		STATUS1	R49 = 0x40	N/A	N/A
LMK5B, LMK5CA, LMK5C	TDCx R-Divider Divided By 2	Required for any GPIOx	R70[0] = 1		
		GPIO0	R57[6:0] = 0x64	R57[6:0] = 0x61	R57[6:0] = 0x5E
		GPIO1	R58[6:0] = 0x64	R58[6:0] = 0x61	R58[6:0] = 0x5E
		GPIO2	R59[6:0] = 0x64	R59[6:0] = 0x61	R59[6:0] = 0x5E

(1) Column is not applicable for LMK5B12212 or LMK5C22212A.

Table 2-28. DPLL FB Divided by 2 - Register Setting

DEVICE FAMILY	TICS PRO FIELD NAME	PIN NAME	DPLL FB DIVIDED BY 2 REGISTER SETTING		
			BAW DPLL	DPLL2	DPLL1(1)
LMK05318B, LMK05318	DPLL FB Divider, div-by-2	STATUS0	R48 = 0x41	N/A	N/A
		STATUS1	R49 = 0x41	N/A	N/A
LMK5B, LMK5CA, LMK5C	TDCx FB-Divider Divided By 2	Required for any GPIOx	R70[0] = 1		
		GPIO0	R57[6:0] = 0x66	R57[6:0] = 0x63	R57[6:0] = 0x60
		GPIO1	R58[6:0] = 0x66	R58[6:0] = 0x63	R58[6:0] = 0x60
		GPIO2	R59[6:0] = 0x66	R59[6:0] = 0x63	R59[6:0] = 0x60

(1) Column is not applicable for LMK5B12212 or LMK5C22212A.

3 Summary

As the need for more features increases, so does the complexity of the device which can cause debug uncertainty. However, debugging clocking devices can be made easy once there is a better understanding of the cause. In general, the recommended debug flow is to sequentially check: the power supply, register configuration, APLL reference clock, and DPLL reference clock. See this debug guide next time the APLL or DPLL is unlocked.

4 References

- Texas Instruments, TICS Pro Software, tool page.
- Texas Instruments, TICS Pro LMK5Bxxxxx and LMK5CxxxxxxA profile overview, product page.
- Texas Instruments, TICS Pro LMK5B/LMK5C start page tutorial, product page.
- Texas Instruments, E2E, forum page.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale, TI’s General Quality Guidelines, or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Last updated 10/2025