

# EVM User's Guide: AFE7900EVM AFE7920EVM TSW14J58EVM

## AFE79xx 评估模块



### 说明

AFE79xx 评估模块 (EVM) 是用于评估德州仪器 (TI) 的 AFE79xx 系列集成式射频采样收发器的评估模块。

AFE79xx 器件支持多达四个发送通道、四个接收通道和两个反馈通道 (4T4R2F)，并集成了锁相环 (PLL) 和压控振荡器 (VCO)，用于生成数据转换器时钟。

AFE79xx 器件集成了八个符合 JESD204B/JESD204C 标准的串行器/解串器 (SerDes) 收发器，它们能够以高达 29.5Gbps 的速率运行，从而通过板载 FPGA 夹层卡 (FMC) 连接器发送和接收数字数据。

EVM 包括 LMK04828 时钟发生器，用于为模拟前端 (AFE) 和采集卡 (现场可编程门阵列, FPGA) 提供参考时钟和 SYSREF。该评估模块 (EVM) 具有单个 5.5V 输入，而且包含完整的电源管理电路。外部时钟选项包括对馈送参考时钟 (用于片上 PLL) 提供支持。该设计可与 TI 图形和采集卡解决方案 (TSW14J58) 以及许多 FPGA 开发套件相连接。

### 特性

- 板载 FPGA 夹层卡 (FMC) 连接器
- 包括完整的电源管理电路
- 板载时钟发生器用于提供参考时钟和 SYSREF
- 用于生成 DAC/ADC 时钟的内部 PLL/VCO
- DAC 或 ADC 速率下的可选外部 CLK
- 串行器/解串器数据接口：
  - 符合 JESD204B 和 JESD204C 标准
  - 8 个高达 29.5Gbps 的串行器/解串器收发器
  - 8b/10b 和 64b/66b 编码
  - 12 位、16 位、24 位和 32 位分辨率
  - 子类 1 多器件同步

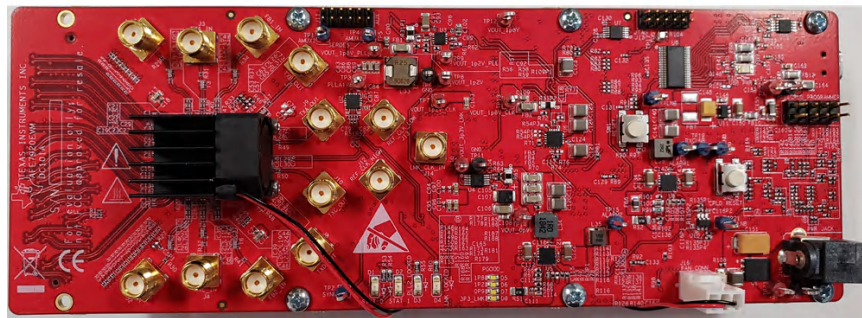


图 1-1. AFE79xxEVM 顶视图

## 1 EVM 概述

### 1.1 注意事项和警告

	<p><b>警告</b></p> <p>活动器件。在运行过程中，请勿触摸旋转盘或电机。</p>
	<p><b>警告</b></p> <p>表面高温。接触会导致烫伤。请勿触摸。</p>

### 1.2 简介

AFE79xxEVM 包括一个时钟和电源方法，并采用 5.5V 单电源供电。如图 1-1 所示，使用微型 A 版 (SMA) 连接器的射频输入和输出位于 EVM 顶部。将参考时钟（例如 10MHz）提供给名为 LMK CLKIN (SMA J19) 的连接器，以通过 LMK04828 的 PLL-1 锁定板载压控晶体振荡器 (VCXO)。<sup>1</sup>

图 1-1 显示了 AFE79xxEVM 的底视图。

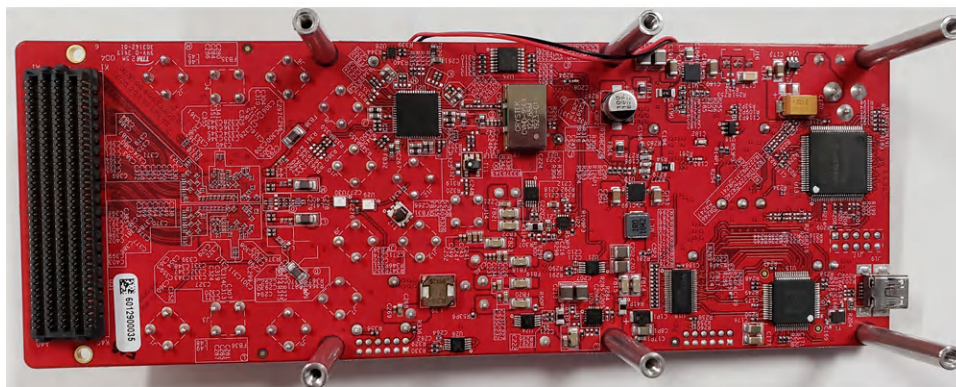


图 1-1. AFE79xxEVM 底视图

使用 SMA J12 (REF\_CLK\_HIGH) 或 SMA J13 (REF\_CLK\_LOW) 提供外部参考时钟以锁定 AFE79xx 中的 PLL。USB 连接器和 5.5V 连接器位于电路板右侧。

将 TSW14J58 采集卡与 AFE79xx EVM 结合使用。TSW14J58 支持高达 29.5Gbps 的串行器/解串行器速度。有关 TSW14J58 EVM 和 AFE79xx EVM 之间的典型连接，请参阅图 1-2。

<sup>1</sup> 许多典型的实验室设备具有 10MHz 振荡器输出，可用于同步多个实验室系统。板载 LMK04828 可以接受来自外部实验室设备的 10MHz 信号，以验证数据采集和生成与 AFE79xx EVM 的同步性和一致性。

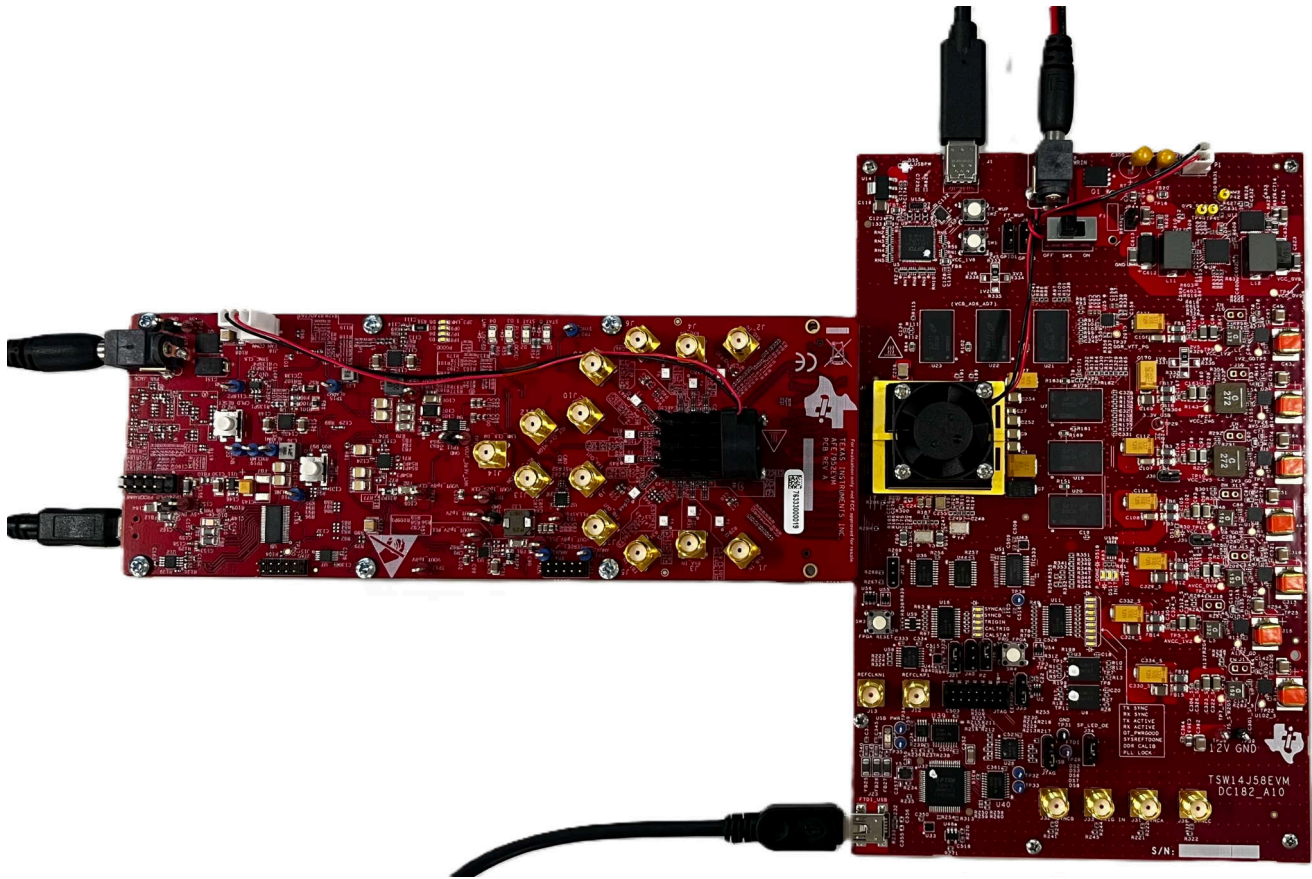


图 1-2. AFE79xxEVM 和 TSW14J58EVM 修订版 A10

### 1.3 套件内容

表 1-1 列出了 EVM 套件的元件。如果缺少任何元件，请与德州仪器 (TI) 产品信息中心联系。TI 强烈建议用户访问 [Ti.com](http://Ti.com)，以确认是否使用了相关软件的最新版本。

表 1-1. 套件内容

条目	数量
AFE79xx EVM	1
Mini USB 电缆	1
电源线	1

### 1.4 规格

[单击此处](#)了解器件规格。

### 1.5 器件信息

AFE79xx 是一系列高性能、高带宽、多通道收发器，集成了四个射频采样发送器链、四个射频采样接收器链和多达两个射频采样数字化辅助链（反馈路径）。发送器链和接收器链的高动态范围使器件可以从无线基站生成和接收 3G、4G 和 5G 信号，而 AFE79xx 器件的高带宽能力适用于多频带 4G 和 5G 基站。

每个接收链均包含一个 25dB 范围的 DSA（数字步进衰减器），后跟一个 3GSPS ADC（模数转换器）。每个接收器通道都有一个模拟峰值功耗检测器和各种数字功耗检测器，可辅助外部或内部自主的自动增益控制器，以及射频过载检测器，用于提供器件可靠性保护。单通道或双通道数字下变频器 (DDC) 提供了高达 600MHz 的组合信号带宽。在 TDD 模式下，接收器通道经过配置可在流量接收器 (TDD RX) 和宽带反馈接收器 (TDD FB) 间动态切换，能够重复使用同一模拟输入来实现两种用途。

每个发送器链均包含一个单通道或双通道数字上变频器 (DUC)，支持最高 1200MHz 的组合信号带宽。DUC 的输出驱动 12GSPS DAC (数模转换器)，通过混合模式输出选项增强在第二或第三奈奎斯特区域的运行。DAC 输出包括一个具有 40dB 范围以及 1dB 模拟和 0.125dB 数字步进的可变增益放大器 (TX DSA)。

反馈路径包括一个用于驱动 3GSPS 射频采样 ADC 的 25dB 范围 DSA，后跟一个带宽高达 1200MHz 的 DDC。

## 2 硬件

需要两个台式电源来为 AFE79xxEVM 和 TSW14J58 EVM 供电。需要一台 PC 来对 EVM 和采集卡进行编程。有关更多信息，请参阅节 2.1。所有实验室设备要求（例如信号源、信号分析仪等）由用户自行决定。

### 备注

通常，额定电压为 5.5V 的台式电源用于为 AFE79xx EVM 供电。标称 EVM 电源电压为 5V。因为 AFE79xx EVM 和 TSW14J58 配置为完全工作模式以适应与电源线损耗相关的压降，因此增加了额外的 0.5V 开销来补偿电源线损耗。

### 2.1 建议的测试环境

- 最大电压和电流分别为 5.5V 和 5A 的电源，用于为 AFE79xx EVM 供电。
- 最大电压和电流分别为 5.5V 和 5A 的电源，用于为 TSW14J58 EVM 供电。
- 支持 USB 3.0 的 Windows PC™，用于通过 ADC 采集和 DAC 图形加载实现文件的快速传输。
- 高质量的射频信号发生器，支持用于评估的相关射频频率。示例设置使用 Keysight™ PSG 系列信号发生器。
- 高质量的射频频谱分析仪，支持用于评估的相关射频频率。示例设置使用 Rohde & Schwarz™ FSQ@26 系列频谱分析仪。

### 2.2 所需硬件

- AFE79xx EVM
- TSW14J58 EVM
- 1 × USB 3.0 Micro-B 电缆（最高 TSW14J58 修订版 A8）/ USB Type-C® 3.0 电缆（TSW14J58 修订版 A9+）
- 2 × USB Mini-B 电缆
- 2 根电源线

### 2.3 硬件设置

#### 2.3.1 AFE79xx EVM 和 TSW14J58 EVM 连接

1. 将 AFE79xx EVM 的 FMC 连接器 U31 连接到 TSW14J58 EVM 的 FMC 连接器 J3。
2. 在电源处于断电模式时，将最大电压和电流分别为 5.5V 和 5A 的电源连接到 TSW14J58EVM 的 J2 +5V IN 连接器。
3. 将 USB 3.0 电缆从 PC 连接到 TSW14J58 EVM 的 J1 连接器。
4. 将 USB Mini-B 电缆从 PC 连接到 TSW14J58 EVM 的 FTDI\_USB 连接器。
5. 在电源处于断电模式时，将最大电压和电流分别为 5.5V 和 5A 的电源连接到 AFE79xx EVM 的 J18 连接器。
6. 将 USB Mini-B 电缆从 PC 连接到 AFE79xx EVM 的 J19 连接器。
7. 可选：将 10MHz 参考实验室设备连接到 J14、LMK\_CLK\_IN 连接器。
8. 可选：通过更改 J35 跳线位置来选择 FPGA 位文件版本。
  - a. 默认值：对于位文件版本 204B，请检查跳线 J35 的物理位置是否覆盖引脚 2 和 3（两个引脚位于距离 FPGA 风扇最远的位置）。
  - b. 对于位文件版本 204C，请检查跳线 J35 的物理位置是否覆盖引脚 1 和 2（两个引脚位于距离 FPGA 风扇最近的位置）。
  - c. 本指南中的配置步骤假定用户使用的是位文件版本 204B。

有关 TSW14J58 EVM 和 AFE79xx EVM 之间的典型连接，请参阅图 1-2。TI 建议将 USB 直接连接到 PC，而不是 USB 集线器，并从 PC 上断开不必要的 USB 设备。其他 USB 设备可能会干扰 PC 识别 EVM USB 句柄的能力。

完成这些步骤后，打开电源。

#### 2.3.2 电源设置

1. 检查 D9 (PWR) LED 灯是否亮起。电源应消耗约 550mA 至 650mA 电流。
2. 检查 D10 (USB\_PWR) LED 灯是否亮起。LED 灯的亮度取决于 USB 电缆的长度。如果 LED 未亮起，请使用较短的 USB 电缆。TI 在此设置中测试了三英尺长的 USB 电缆。

3. AFE79xx EVM 上的电源序列发生器显示了每个电源轨的电源状态。如果电源正常输出 (PGOOD) 处于逻辑高电平，则相应的电源轨已正确供电。相应的 LED 亮起。检查以下 LED 以确认指示灯亮起。
- D5 (1P8) LED
  - D6 (1P2) LED
  - D7 (0p9) LED
  - D8 (3p3\_LMK) LED

## 3 软件

### 3.1 所需软件

用于配置 AFE79xxEVM 的软件称为 **Latte**。最新版本的 Latte 可从 [TI 的安全资源网站](#) 下载。安装程序文件名为 **TI-AFE79xx-Latte\_V2p5.exe**。

#### 3.1.1 软件安装顺序

1. 安装 **TI-AFE79xx-Latte\_V2p5.exe** 或最新版本。安装软件后，从 AFE79xx 启动窗口中的“启动”菜单启动该软件，或选择桌面上的 AFE79xx 图标。初始化软件最长需要 2 分钟。
2. 记下 Latte 软件的安装目录。

#### 备注

默认 Latte 软件目录位于：

**C:\Users\“User ID”\Documents\Texas Instruments\Latte**

将 “User ID” 替换为相应的 Windows 登录 ID。

### 3.2 Latte 概述

1. 完全初始化软件后，从桌面快捷方式启动 Latte GUI，也可以选择所有程序，然后选择 **Texas Instruments**。

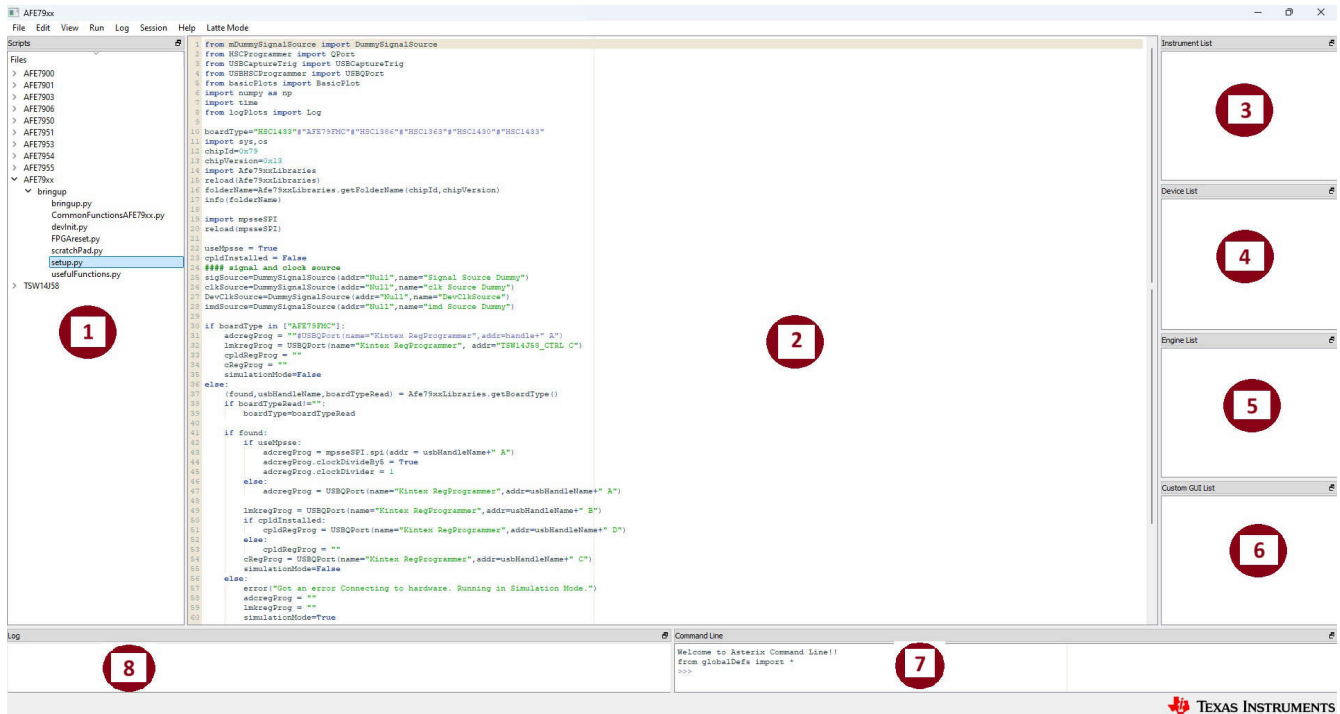


图 3-1. Latte GUI 概述

#### 3.2.1 Latte 用户界面

Latte UI 分为八个窗口（标记为 1 到 8），具有以下功能：

- 窗口 1：

此窗口（也称为 **Scripts**）显示了可用的 Python 脚本列表，这些脚本可生成用于配置 AFE79xxEVM 的寄存器命令。显示的脚本文件位于 **..\Documents\Texas Instruments\AFE79xx\Latte\projects\AFE79xx\bringup** 文件夹中。根据需要修改和创建新脚本。修改后的和新的脚本会在 Latte 重新启动时出现在窗口 1 的子窗口中。

- 窗口 2：

此窗口 ( 也称为 *编辑器* ) 显示当前选定脚本中的代码。根据需要, 使用窗口 2 修改并保存代码。

- **窗口 3 至 6 :**

当脚本运行以配置 AFE79xxEVM 时, 这些窗口会更新, 主要用于提供信息。

- **窗口 7 :**

此窗口 ( 也被称为 *Command Line* ) 用于输入和运行各个命令。此类命令的示例包括更改 TX、RX 和 FBRX DSA、NCO 等。

- **窗口 8 :**

此窗口 ( 也被称为 *Log* ) 会在脚本执行期间显示消息以显示当前状态。此窗口也用于故障排除。

### 3.2.2 有用的 Latte 快捷方式

**运行脚本文件:** 运行脚本文件的方法是, 首先在 “Scripts” 窗口中选择文件, 然后按下 F5 ( 或依次选择菜单栏中的 *Run* 和 *Buffer* ) 。

**运行部分脚本:** 运行部分脚本文件的方法是, 在 “Editor” 窗口中选择相关行, 然后按下 F7 ( 或依次选择菜单栏中的 *Run* 和 *Run Selection* ) 。

**停止执行:** 停止当前执行的方法是按下 F10 ( 或依次选择菜单栏中的 *Run* 和 *Stop* ) 。

**清除会话:** 按 Ctrl-T ( 或依次选择菜单栏中的 *Session* 和 *Clear Session* ) , 清除当前会话以将 Latte UI 重置为初始状态。此过程相当于重新启动, 可用于在不关闭 GUI 的情况下重新启动会话。

## 4 实现结果

### 4.1 AFE79xxEVM 配置

本节将指导用户完成启动 AFE79xxEVM 的步骤序列。本节对自动设置过程进行了细分，以使用户了解该过程并进行相应的修改。

#### 4.1.1 将 Latte 连接到板

执行此步骤会在运行 Latte 的 PC 和 AFE79xxEVM 之间建立连接。

1. 在脚本窗口中，选择 **AFE79xx** 下拉菜单，然后选择 **启动** 下拉菜单。选择 **setup.py**。在第 13 行，验证芯片版本是否为 0x13。按 F5 运行程序。
2. 检查日志窗口以确保没有错误。检查以下行是否显示两次：*Kintex RegProgrammer - USB Instrument created*。
3. AFE79xxEVM 中 FT4232H 芯片的驱动程序缺失或过时是常见的错误源。在 Latte 中，使用 **视图** 选项卡，然后使用 **USB 句柄** 来检查与 Latte 的 USB 句柄连接。TSW14J58 有四个连接，EVM 有四个连接。也可以利用设备管理器，通过检查 USB 实例来验证 PC 和 EVM 之间的连接。
4. 如有必要，利用适当的驱动程序更新您的 PC。

```
Log
#=====
#Executing .. AFE79xx/bringup/setup.py
#Start Time 2025-04-03 16:00:10.889000
AFE79xxLibraryPG1p0
spi - USB Instrument created.
resetDevice
Kintex RegProgrammer - USB Instrument created.
Kintex RegProgrammer - USB Instrument created.
FPGA reset - USB Instrument created.
#Done executing .. AFE79xx/bringup/setup.py
#End Time 2025-04-03 16:00:17.812000
#Execution Time = 6.92300009727 s
#===== ERRORS:0, WARNINGS:0 =====#
```

图 4-1. setup.py 成功后的 Latte 日志

#### 4.1.2 编译库

在这一步中，将编译与 Latte UI 一同打包的脚本库，大约需要运行 30 秒。

1. 在脚本窗口中选择 **devInit.py**。
2. 选择 F5 运行程序。
3. 在日志窗口中查看状态和错误。

```
Log
#=====
#Executing .. AFE79xx/bringup/devInit.py
#Start Time 2025-04-03 16:02:34.001000
Power Card - USB Instrument created.
Version : 0x104204b
Connected to Capture Card
Loaded Libraries
#Done executing .. AFE79xx/bringup/devInit.py
#End Time 2025-04-03 16:02:50.159000
#Execution Time = 16.1579999924 s
#===== ERRORS:0, WARNINGS:0 =====#
```

图 4-2. devInit.py 成功后的 Latte 日志

#### 4.1.3 对 AFE79xx EVM 进行编程

按照以下说明在 AFE79xxEVM 上对 LMK04828 和 AFE79xx 进行编程。

1. 点击名为 **bringup.py** 的脚本，然后按下 F5。预计不会出现错误；忽略有关 SPI 控制、撤回或重置属性的任何警告。此步骤需要几分钟。
2. 检查“Log”窗口以监控任何错误。此步骤会完成 AFE79xxEVM 配置。AFE79xxEVM 中的电流消耗约为 3A。
3. FPGA 位文件版本不匹配错误表明固件与配置文件不匹配。对于位文件版本 204B，请检查跳线 J35 的物理位置是否覆盖引脚 2 和 3 (两个引脚位于距离 FPGA 风扇最远的位置)。

4. LOS 错误表示 SerDes RX 电气空闲，并且 TX 输出不正常。通过重新发送数据 ( DAC 图形 ) 并重新运行 *bringup.py* 重新配置 AFE79xxEVM 来解决此错误。
5. GPIO 警告或 sysref 错误通常表示电源电压或电流限制。验证 AFE79xxEVM 的电源并确保使用 5.5V 电源电压和 5A 电流限制。重新启动 Latte UI 并重新运行脚本。

```
Log
AFE AGC configured.
AFE GPIO configured.
Sysref Read as expected
#####Device DAC JESD-RX 0 Link Status#####
CS State TX0: 0b10101010 . It is expected to be 0b10101010
FS State TX0: 0b01010101 . It is expected to be 0b01010101
Could get the link up for device RX: 0
#####Device DAC JESD-RX 1 Link Status#####
CS State TX0: 0b10101010 . It is expected to be 0b10101010
FS State TX0: 0b01010101 . It is expected to be 0b01010101
Could get the link up for device RX: 1
#####
AFE Configuration Complete
#Done executing ., AFE79xx/bringup/bringup.py
#End Time 2025-04-03 16:05:57.313000
#Execution Time = 104.355000019 s
#===== ERRORS:0, WARNINGS:1 =====#
```

图 4-3. bringup.py 成功后的 Latte 日志

#### 4.1.4 TXDAC 评估

1. 将信号分析仪连接到 J9 (TXA)、J7 (TXB)、J10 (TXC) 或 J8 (TXD) 以监控 TXDAC 输出。
2. 突出显示适用代码部分，然后按 F7 运行。参阅图 4-4 以了解典型的性能测量。

```
### TX Tone Out

## Send 10MHz Tone to TX AB
amp1A = -10 #band 0amp in dBfs
freqA = 10 #band 0 freq in MHz
amp1B = -10 #band 1amp
freqB = 10 #band 1 freq
AFE.selectCH(2,0)
AFE.FPGA.sendSingleTone(0,freqA,amp1A,freqB,amp1B)

## Send 10MHz Tone to TX CD
amp1A = -10 #band 0amp
freqA = 10 #band 0 freq
amp1B = -10 #band 1amp
freqB = 10 #band 1 freq
AFE.selectCH(2,2)
AFE.FPGA.sendSingleTone(0,freqA,amp1A,freqB,amp1B)
```

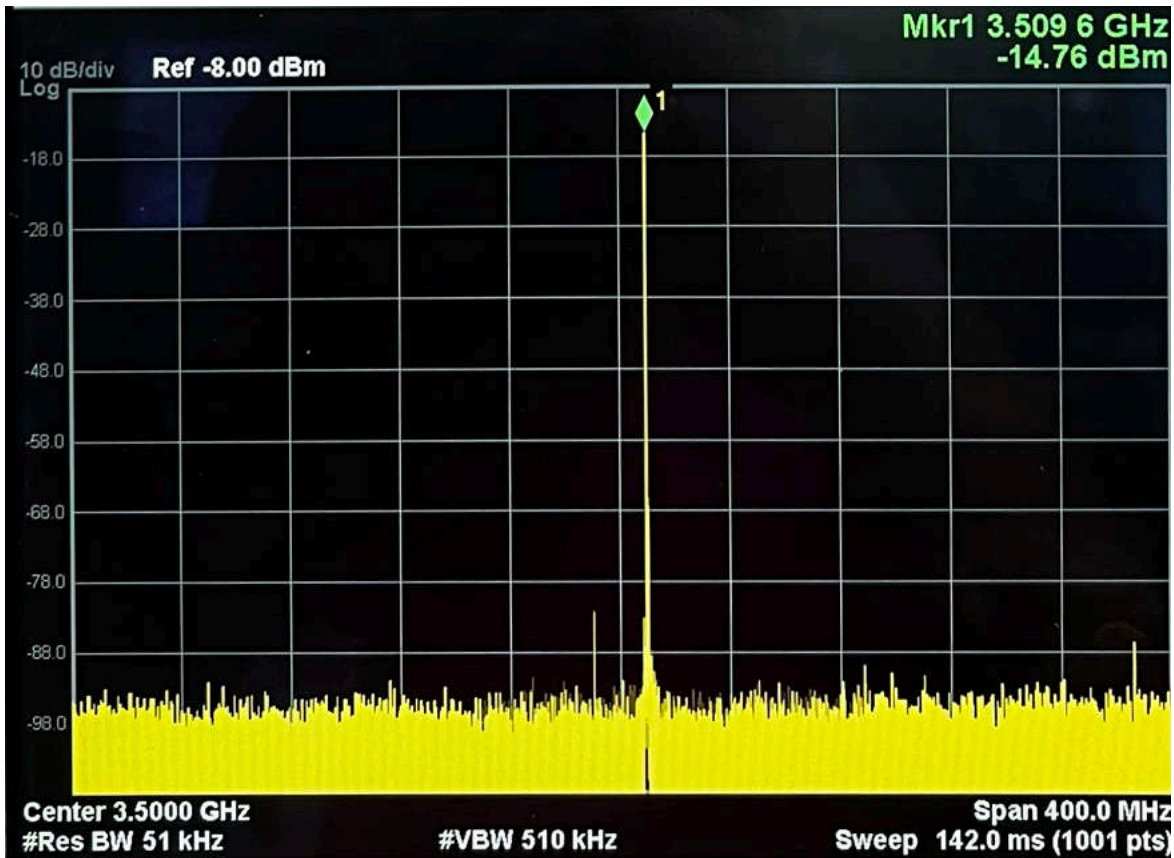


图 4-4. 在 3.5GHz NCO 且振幅为 -10dBFS 时的典型 TXDAC 10MHz 单频性能

#### 4.1.5 RXADC 和 FBADC 评估

1. 将信号发生器连接到 J3 (RXA)、J1 (RXB)、J4 (RXC) 或 J2 (RXD)，以在定义的 NCO + 10MHz 上提供 RXADC 输入。将信号发生器连接到 J5 (FBAB) 或 J6 (FBCD)，以在定义的 NCO + 10MHz 处提供 FBADC 输入。
2. 突出显示以下适用代码部分，然后选择 F7 运行。在节 3.2.1 的 Latte 窗口 6 中，选择 CGui\_CGui\_Custom\_Gui 以查看采集结果。
3. 在采集窗口中，选择有限样本采集类型。在复杂选项卡中查看采集结果。

```
### RX/FB Tone In
AFE.selectCh(0,0,0) # select RX A, band 0
AFE.selectCh(0,0,1) # select RX A, band 1
AFE.selectCh(0,1,0) # select RX B, band 0
AFE.selectCh(0,1,1) # select RX B, band 1
AFE.selectCh(0,2,0) # select RX C, band 0
AFE.selectCh(0,2,1) # select RX C, band 1
AFE.selectCh(0,3,0) # select RX D, band 0
AFE.selectCh(0,3,1) # select RX D, band 1

AFE.selectCh(1,0) #select FBAB
AFE.selectCh(1,1) #select FBCD
```

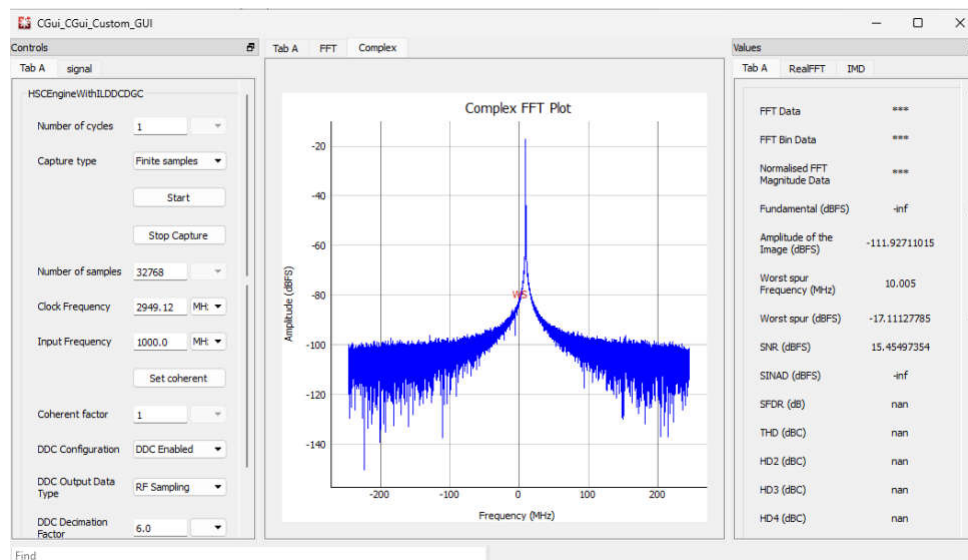


图 4-5. 3.5GHz NCO 时的典型 RXADC 10MHz 单频性能

## 4.2 AFE79xxEVM 配置修改

所提供的脚本使用 Latte 脚本中的默认设置配置 AFE79xx。您可以通过修改一组参数来更改设置。

本节介绍通过 python 脚本修改 AFE79xxEVM 启动的步骤。示例展示的是默认 AFE79xxEVM。表 4-1 提供了默认模式配置的概述。

表 4-1. AFE79xx EVM 默认引脚概述

模式	默认编程
TX ( 发送器 )	启用 4 个 TXDAC，DSA = 0，LMFSHd_2TX = 44210，24 次内插，491.52MSPS 数据速率
RX ( 接收器 )	启用 4 个 RXADC，DSA = 0，LMFSHd_2RX = 12410，12 次内插，245.76MSPS 数据速率
FBRX ( 反馈接收器 )	启用 2 个 FBADC，DSA = 0，LMFSHd_1FB = 24410，6 次内插，数据速率为 491.52MSPS
串行器/解串器	八个通道在 9830.4Mbps 下运行
数据转换器时钟速率	F <sub>RXADC</sub> = 2949.12MSPS，F <sub>FBADC</sub> = 2949.12MSPS，F <sub>TXDAC</sub> = 11796.48MSPS。
状态	RX AGC 已禁用，RX、TX DSA 阶跃损伤未校正，DAC 处于交错模式。

### 4.2.1 数据转换器时钟设置

该参数用于配置数据转换器时钟和时钟分配路径。

```
#Configures the reference input frequency to the on-chip PLL of the AFE79xx.
sysParams.FRef = 491.52
#Configures the RXADC converter sample rate.
sysParams.FadcRx = 2949.12
#Configures the FBADC converter sample rate.
sysParams.FadcFb = 2949.12
#Configures the TXDAC converter sample rate.
sysParams.Fdac = 2949.12*4
#Sets the clock source for the RXADC converters. The source is now from the on-chip PLL.
sysParams.externalClockRx = False
#Sets the clock source for the TXDAC converters. The source is now from the on-chip PLL.
sysParams.externalClockTx = False
```

#### 4.2.2 数据速率和 JESD 参数

信号链中的数据速率通常与 JESD 模式 (LMFS、SerDes 速率) 相连。器件数据表提供了兼容模式的列表。使用 `bringup.py` 脚本中的以下参数来修改配置。为以下参数分配新值后重新运行脚本。

```
## In below parameters, each element sets the particular LMFS-Hd for the particular channels.
# JESD and Serdes Parameters
sysParams.LMFSHdRx = ["12410","12410","12410","12410"]
sysParams.LMFSHdFb = ["24410","24410"]
sysParams.LMFSHdTx = ["44210","44210","44210","44210"]
# Decimation and interpolation parameters for the data converter signal chains.
sysParams.ddcFactorRx = [12,12,12,12]
sysParams.ddcFactorFb = [6,6]
sysParams.ducFactorTx = [24,24,24,24]
```

#### 4.2.3 修改 NCO 的步骤

- 默认 AFE79xx EVM 具有以下射频频率匹配网络：
  - RXA、RXB、FBAB、TXA 和 TXB = 3500MHz
  - RXC、RXD、FBCD、TXC 和 TXD = 2600MHz
- 下面所示为更改 NCO 以匹配默认射频频率匹配网络的示例脚本：

```
## Update RX NCO
afeInst = 0 #AFE Instance of AFE79_INST_TYPE type. If using the EVM this should be 0.
rxChSel = 0 #Value to select the RX chain.Value 0 for RXA to 3 for RXD.
bandNo = 0 #Band number. 0-band0, 1-band1.
ncoNo = 0 #NCO number. 0-NCO0, 1-NCO1.
ncoFreq = 7200#NCO frequency to set the NCO to in MHz.

if sysParams.ncoFreqMode == '1KHz':
    mixer = ncoFreq*1e3 #Should pass value in KHz in 1KHz          ncoFreqMode and the frequency word
    value in FCW mode.

elif sysParams.ncoFreqMode == 'FCW':
    mixer = int(round(2**32*ncoFreq/sysParams.FadcRx)) #Should pass value in KHz in 1KHz
    ncoFreqMode and the frequency word value in FCW mode.

CAFE.updateRXNco(afeInst,rxChSel,mixer,bandNo,ncoNo)
engine.DDCNCOFreqWord=((ncoFreq)%sysParams.FadcRx)/(sysParams.FadcRx)*2**32 #Updating NCO word in
capture window

## Update TX NCO
afeInst = 0 #AFE Instance of AFE79_INST_TYPE type. If using the EVM this should be 0.
txChSel = 0 #Value to select the TX chain.Value 0 for TXA to 3 for TXD.
ncoNo = 0 #NCO number. 0-NCO0, 1-NCO1.
band0NCO0 = 2600000 #NCO frequency to set the band 0 NCO0 to in KHz.
band1NCO0 = 2600000 #NCO frequency to set the band 1 NCO0 to in KHz.
band0NCO1 = 3500000 #NCO frequency to set the band 0 NCO1 to in KHz.
band1NCO1 = 3500000 #NCO frequency to set the band 1 NCO1 to in KHz.
CAFE.updateTXNcoDb(afeInst,txChSel,ncoNo,band0NCO0,band1NCO0,band0NCO1,band1NCO1)
```

#### 4.2.4 修改 DSA 的步骤

启用 DAC 上的数据传输功能和 ADC 上的数据采集功能。此时调整 RXNCO、FBNCO 和 TXNCO。

此时调整 RXDSA、FBDSA 和 TXDSA。

```
## Set RX DSA
afeInst = 0 #AFE Instance of AFE79_INST_TYPE type. If using the EVM this should be 0.
rxChSel = 0 #Value to select the RX chain.Value 0 for RXA to 3 for RXD.
dsaSetting = 20 #Analog DSA Index. Attenuation applied is dsaSetting*0.5dB
CAFE.setRXDsa(afeInst,rxChSel,dsaSetting)

## Set TX DSA
afeInst = 0 #AFE Instance of AFE79_INST_TYPE type. If using the EVM this should be 0.
txChSel = 0 #Value to select the TX chain.Value 0 for TXA to 3 for TXD.
dsaSetting = 20 #Analog DSA Index. Attenuation applied is dsaSetting*1dB
CAFE.setTXDsa(afeInst,txChSel,dsaSetting)
```

## 5 硬件设计文件

### 5.1 原理图

[单击此处](#)查看器件原理图。

### 5.2 PCB 布局

[点击此处](#)了解 PCB 布局。

### 5.3 物料清单 (BOM)

[单击此处](#)查看物料清单。

## 6 其他信息

### 6.1 状态检查和故障排除指南

本节提供有关 AFE79xx EVM 状态指示器的通用指南以及相应的故障排除指南。

#### 6.1.1 AFE79xx EVM 状态指示器

此时绿色 LED 灯 D3 亮起。D3 表示 LMK04828 的 PLL 环路 2 已锁定。(可选) LED D4 指示 LMK04828 的 PLL 环路 1 已锁定。如果有外部设备为 LMK04828 提供 10MHz 参考时钟,以进行实验室设备同步,则 LED 灯 D4 必须亮起。EVM 在没有运行 PLL 环路 1 的情况下仍然可以工作,但需要 PLL 环路 2 才能成功启动。

- 如果 PLL 环路 1 未运行,请检查 10MHz 参考时钟。10MHz 参考时钟对于通过信号发生器和频谱分析仪实现信号一致性非常必要。
- 如果 PLL 环路 2 未锁定,请联系 TI 应用以获得更多支持。

#### 6.1.2 TSW14J58EVM 状态指示器

在 LED 灯条上,TX\_Sync 和 RX\_Sync 未亮起。TX\_Active 和 RX\_Active 亮起。亮起规范是 JESD204B 收发器模式工作的必要要求。TX\_Active 闪烁表示 TXDAC JESD204B 链路已建立,而 RX\_Active 闪烁表示 RXADC 或 FBADC JESD204B 链路已建立。

#### 商标

Windows PC™ is a trademark of Microsoft Corporation.

Keysight™ is a trademark of Keysight Technologies, Inc.

Rohde & Schwarz™ is a trademark of Rohde. & Schwarz GmbH. & Co.

USB Type-C® is a registered trademark of USB Implementers Forum.

所有商标均为其各自所有者的财产。

## 7 修订历史记录

注：以前版本的页码可能与当前版本的页码不同

Changes from Revision A (April 2023) to Revision B (September 2025)	Page
• 更新了文档以反映当前的 EVM 用户指南格式.....	1
• 通篇添加了 TSW14J58EVM 和 TSW14J58 采集卡信息.....	1
• 添加了 特性 章节.....	1
• 添加了 套件内容 章节.....	3
• 添加了 规格 章节.....	3
• 添加了 器件信息 部分.....	3
• 更新了 硬件设置 章节.....	5
• 更新了 AFE79xx EVM 和 TSW14J58 EVM 连接 章节.....	5
• 更新了 所需软件 章节.....	7
• 将 所需软件 从 硬件 移至 软件 .....	7
• 更新了 软件安装序列 章节.....	7
• 删除了 软件安装检查 章节.....	7
• 更新了 Latte 概述 章节.....	7
• 将 Latte 概述 移至 软件 章节.....	7
• 删除了 AFE79xx EVM 自动配置 章节.....	9
• 添加了 实现结果 章节.....	9
• 从 AFE79xxEVM 手动配置更新到了：AFE79xxEVM 配置.....	9
• 将 AFE79xxEVM 配置 移至 实现结果 .....	9
• 添加了 setup.py 成功后的 Latte 日志 图.....	9
• 添加了 devlnit.py 成功后的 Latte 日志 图.....	9
• 添加了 bringup.py 成功后的 Latte 日志 图.....	9
• 删除了 3.5GHz 时的典型 TXDAC 单频性能 图.....	10
• 添加了 ### TX 输出单频 代码块.....	10
• 添加了 在 3.5GHz NCO 且振幅为 -10dBFS 时的典型 TXDAC 10MHz 单频性能 图.....	10
• 更新了 RXADC 和 FBADC 评估 章节.....	11
• 删除了 HSDC PRO ADC 性能 FFT 频段配置 图.....	11
• 删除了 RXADC 的 HSDC PRO ADC 数据采集和典型单频 3.5GHz 性能 图.....	11
• 删除了 FBADC 的典型单频 3.5GHz 性能 图.....	11
• 添加了 ### RX/FB 输入单频 代码块.....	11
• 添加了 3.5GHz NCO 时的典型 RXADC 10MHz 单频性能 代码块.....	11
• 从 AFE79xx EVM 自动配置 更新为了：AFE79xxEVM 配置修改 .....	12
• 更新了 AFE79xx EVM 自动配置 章节.....	12
• 添加了 修改 NCO 的步骤 章节.....	13
• 删除了 AFE79xxEVM 手动配置 章节.....	13
• 删除了 使用 HSDC Pro 设置 TSW14J5x 章节.....	13
• 删除了 ##RXDSA 调整 代码块.....	13
• 添加了 ## 设置 RX DSA 代码块.....	13
• 添加了 硬件设计文件 章节.....	14
• 添加了 更多信息 章节.....	15
• 将 状态检查和故障排除指南 移到了 更多信息 章节.....	15

Changes from Revision * (October 2019) to Revision A (April 2023)	Page
• 添加了 注意事项和警告 部分.....	2

## 重要通知和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、与某特定用途的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他安全、安保法规或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的相关应用。严禁以其他方式对这些资源进行复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。对于因您对这些资源的使用而对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，您将全额赔偿，TI 对此概不负责。

TI 提供的产品受 [TI 销售条款](#)、[TI 通用质量指南](#) 或 [ti.com](#) 上其他适用条款或 TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。除非德州仪器 (TI) 明确将某产品指定为定制产品或客户特定产品，否则其产品均为按确定价格收入目录的标准通用器件。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

版权所有 © 2025，德州仪器 (TI) 公司

最后更新日期：2025 年 10 月