

Nathan Block, Kevin Ortiz

内容

1 电源管理框架.....	2
2 低功耗的硬件设计选项.....	5
3 针对低功耗的线性调频脉冲设计优化.....	6
4 降低帧间空闲和深度睡眠状态下的功耗.....	14
5 测量功率.....	17
6 参考文献.....	18
7 修订历史记录.....	18

插图清单

图 1-1. xWRL6432 器件架构.....	2
图 1-2. 1 帧运行期间的典型应用流程.....	3
图 3-1. 线性调频脉冲中的事件序列.....	8
图 3-2. IWRL6432BOOST 虚拟天线布局.....	12
图 4-1. 可切换和不可切换存储器.....	16

表格清单

表 1-1. 每个电源域中的主要元件.....	2
表 1-2. 应用流程状态定义.....	3
表 1-3. 不同电源模式下的电源域状态.....	4
表 4-1. xWRL6432 主存储器映射.....	15
表 4-2. APPSS 和共享 RAM 存储器组.....	15

商标

所有商标均为其各自所有者的财产。

1 电源管理框架

德州仪器 (TI) 的低功耗 xWRL6432 雷达传感器通过发射和接收调频连续波 (FMCW) 来检测物体和运动。xWRL6432 专为可视门铃、安全系统、手势控制 HMI 和汽车入侵检测等低功耗应用而设计，并提供多种低功耗模式，让雷达器件能够根据用例需求更大限度地降低功耗。本应用手册详细介绍了通过有效系统设计、线性调频脉冲剖面和应用软件来降低功耗的几种方法。此外，本应用手册将介绍用于测量 xWRL6432 器件功率的技术。

电源域

xWRL6432 的低功耗架构允许雷达器件完全或部分关闭特定电源域。图 1-1 展示了 xWRL6432 电源域架构的方框图，表 1-1 详细介绍了每个电源域内的特定子块。

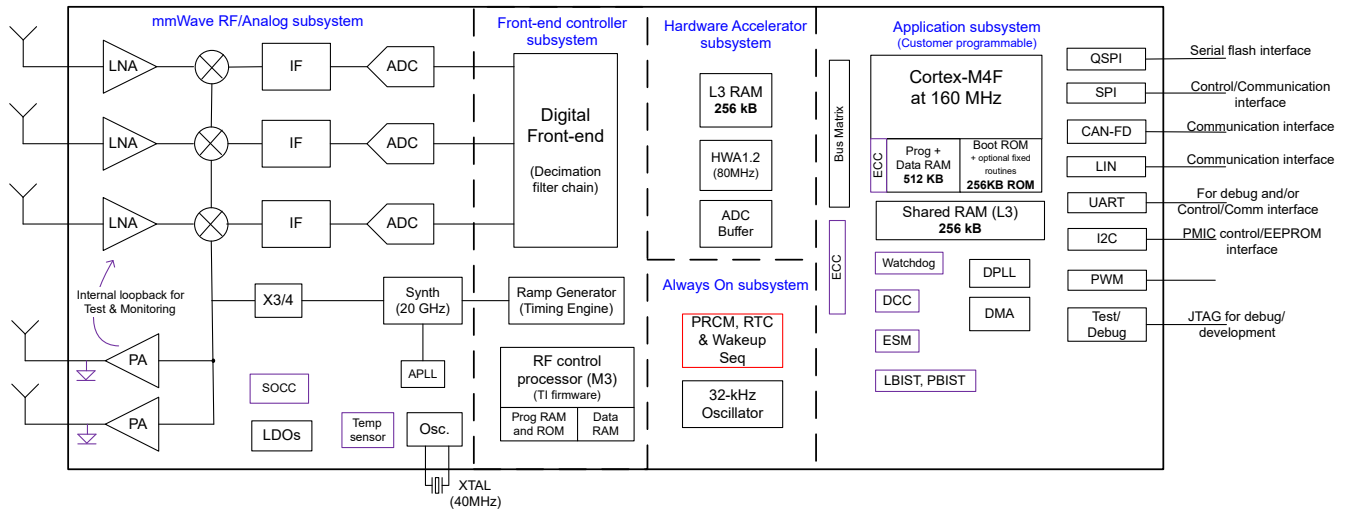


图 1-1. xWRL6432 器件架构

表 1-1. 每个电源域中的主要元件

电源域	主要元件
RF_ANA_PD (射频模拟电源域)	<ul style="list-style-type: none"> 功率放大器 (PA) 低噪声放大器 (LNA) 混音器 中频滤波器 (IF) 模数转换器 (ADC) 合成器 40MHz 晶体振荡器 (OSC)
FEC_PD (前端控制器电源域)	<ul style="list-style-type: none"> Cortex-M3 处理器 (包括存储器) 数字前端 (DFE)
APPSS_PD (应用子系统电源域)	<ul style="list-style-type: none"> Cortex-M4F 处理器 应用存储器组 通用外设 (看门狗、UART、I2C、SPI、RS232...)
HWASS_PD (硬件加速器电源域)	<ul style="list-style-type: none"> 硬件加速器 (HWA) HWA 的存储器
AON_PD (常开电源域)	<ul style="list-style-type: none"> 实时时钟 (RTC) 电源、复位和时钟管理 (PRCM) 寄存器

典型的雷达应用流程

图 1-2 展示了典型的雷达应用流程。表 1-2 中说明了本流程中使用的术语。

备注

此流程上有许多变化，这些变化超出了本文档的范围。

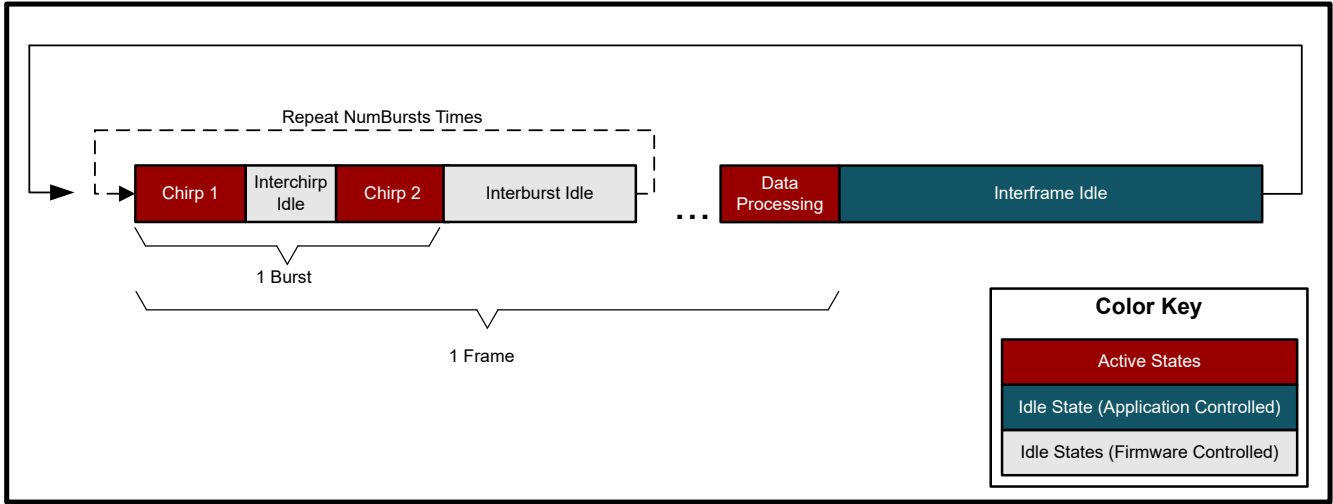


图 1-2. 1 帧运行期间的典型应用流程

表 1-2. 应用流程状态定义

线性调频脉冲	一个或多个发送器发射/接收 FMCW 波的时间段。
突发	线性调频脉冲序列。xWRL6432 器件具有两根发射天线，因此典型的突发可能在一根或两根天线上发射。
FRAME	由一系列突发后跟数据处理组成的时间段。帧在定义的间隔上是周期性的。
线性调频脉冲间空闲	线性调频脉冲之间的时间段
突发间空闲	突发之间的时间段
帧间空闲	从帧结束与下个帧开始之间的时间段
数据处理	器件处理前一帧中收集的雷达数据的时间段

如上图所示，雷达器件会定期多次发出线性调频脉冲，处理收集的所有数据，然后进入低功耗帧间空闲状态。下一节将更详细地介绍上述每个状态。

电源状态说明

运行

活动状态是指器件发出线性调频脉冲或处理线性调频脉冲数据时。在此状态下，器件可以处于数据采集子状态（即通过发出线性调频脉冲收集数据时），也可以处于数据处理子状态（即一起处理数据采集子状态中记录的样本时）。当器件处于数据采集子状态时，APPSS_PS 和 FEC_PD 都必须打开，但是，当器件处于数据处理子状态时，FEC_PD 可能会断电以减少功耗。在活动状态下的数据采集子状态中，器件具有最高功率级别。

空闲

当器件未主动发出线性调频脉冲或处理数据时，便会进入空闲状态。空闲状态有三种类型（线性调频脉冲间空闲、突发间空闲和帧间空闲）。线性调频脉冲间空闲和突发间空闲状态完全由器件固件处理。当器件在线性调频脉冲和突发之间循环时，它将自动进入这些状态。相比之下，器件在帧之间进入的帧间空闲状态可以由用户配置和修改。每个空闲状态的时间要求如下：

	说明	最短时间	相对于其他空闲模式的功率
线性调频脉冲间空闲	同一突发中两个连续线性调频脉冲之间的低功耗模式。	如果启用低功耗模式：6 μ s - TX_START_TIME 或 3.1 μ s (以较大者为准) 否则：4 μ s - TX_START_TIME 或 3.1 μ s (以较大者为准)	功耗大于突发间空闲和帧间空闲。
突发间空闲	在同一帧中两个连续突发之间的低功耗模式。	95 μ s	功耗低于线性调频脉冲间空闲，但高于帧间空闲。
帧间空闲	两帧之间的低功耗模式。	135 μ s	功耗低于线性调频脉冲间空闲和突发间空闲。

深度睡眠：帧间空闲时间的可选状态

深度睡眠状态是 xWRL6432 在已经处于帧间空闲状态时的一个应用程序驱动选项。深度睡眠是**器件中可能的最低功耗状态设计状态**，其中几乎所有器件电源域（包括应用子系统 (APPSS) 以及硬件加速器 (HWA) 和前端控制器子系统 (FECSS)）都将断电，来节省大量电量。当器件进入帧间空闲状态时，应用程序可以触发深度睡眠。深度睡眠需要一段标称时间来取消软件初始化并将硬件断电，然后在深度睡眠周期结束后为硬件加电并重新初始化软件。这种情况的典型时间约为 2 毫秒，但因所选深度睡眠选项而异。

尽管整个器件几乎完全断电，但从深度睡眠唤醒后无需重新启动。器件的内容（例如应用程序映像和线性调频脉冲剖面）会在深度睡眠周期内保存到 APPSS/FECSS 存储器中。

器件可以通过睡眠计数器、UART RX、SPI CS、GPIO 和 RTC 退出深度睡眠状态。

不同电源模式下的电源域

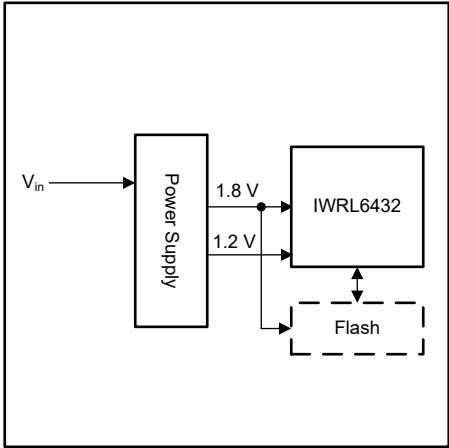
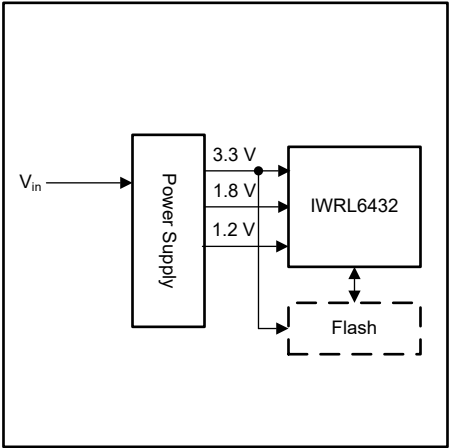
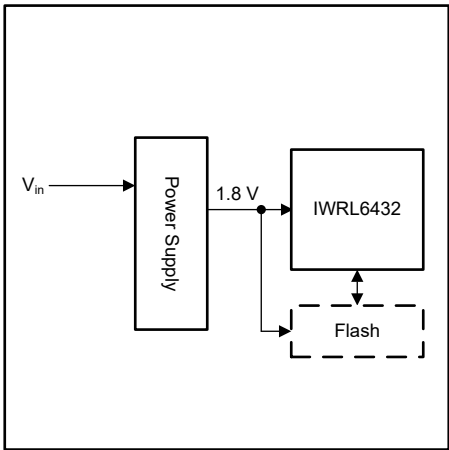
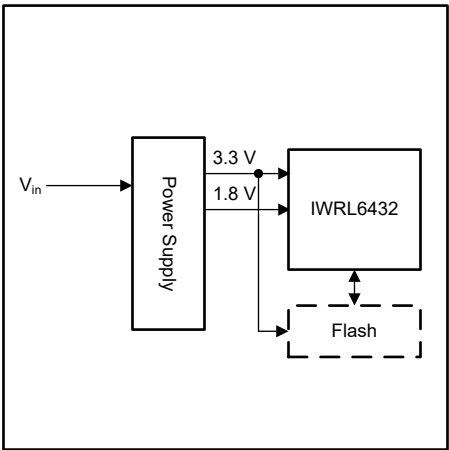
下表展示了不同模式下不同电源域的状态。请注意，其中许多可能会被修改，并且电源域的不同部分可能会由用户时钟门控或断电。

表 1-3. 不同电源模式下的电源域状态

电源域	活动 (发出线性调频脉冲)	活动 (数据处理)	线性调频脉冲间空闲	突发间空闲	帧间空闲	深度睡眠
RF_ANA_PD	打开	关闭 (晶体振荡器开启)	开启 (PA、LNA 关闭)	关闭 (晶体振荡器开启)	关闭 (晶体振荡器开启)	关闭
FEC_PD	打开	关闭	开启 (数字前端时钟选通)	开启 (数字前端时钟选通)	开启 (整个子系统时钟选通)	关闭
APPSS_PD	开启 (PLL 关闭)	开启 (在数字 PLL 上运行)	开启 (PLL 关闭)	开启 (PLL 关闭)	开启 (在晶体时钟上运行)	关闭
HWASS_PD	开启 (HWA 时钟选通)	打开	打开	开启 (HWA 时钟选通)	关闭	关闭
AON_PD	打开	打开	打开	打开	打开	打开

2 低功耗的硬件设计选项

xWRL6432 器件设计为由电源管理集成电路 (PMIC) 或分立式电源解决方案供电。可以为 xWRL6432 提供 1-3 个不同的电源电压轨。这种灵活性使系统能够进行定制，从而降低功耗或 BOM 成本。启动时，该器件会检测提供的输入电压数量，并相应地调整其内部电路。下表 (图片来自 [1]) 展示了可用于为 xWRL6432 提供电源轨的四种拓扑。

	1.8V IO 模式	3.3V IO 模式
功耗优化	 <p>所需的电源轨数量 : 2 最低功耗拓扑</p>	 <p>所需的电源轨数量 : 3</p>
BOM 优化	 <p>所需的电源轨数量 : 1 成本最低的拓扑</p>	 <p>所需的电源轨数量 : 2</p>

这四种拓扑在两个方面有所不同：IO 电压和 1.2V 电源的来源。根据应用要求，器件上的 IO 可以在 3.3V 逻辑或 1.8V 逻辑上运行。1.2V 电源可以通过低压降稳压器 (LDO) 从 1.8V 电源生成 (BOM 优化模式)，也通过外部电压源提供给器件 (功耗优化模式)。

3 针对低功耗的线性调频脉冲设计优化

在 xWRL6432 可能进入的所有不同状态中，活动模式（特别是活动模式期间的线性调频脉冲频率扫描行为）是功耗最大的。由于线性调频脉冲频率扫描行为会打开高功率射频放大器来发送和接收雷达波，因此这种行为的功耗最大。既如此，用户便可通过减少频率扫描所花费的时间和/或降低频率扫描时的功耗来降低总体功耗。以下非详尽列表列出了在线性调频脉冲频率扫描期间设计降低功耗的最有效方法。

功率降低技术	性能影响
增加帧之间的时间	更高的延迟
减少线性调频脉冲空闲和突发间空闲中花费的时间	更高的最大速度。更粗的速度分辨率
缩短线性调频脉冲频率扫描时间	范围缩小，在视场边缘更明显
减少发送器/接收器的数量	更低的角分辨率。 范围缩小，在视场边缘更明显
降低发射功率	范围缩小，在视场边缘更明显

这些技术许多都涉及修改配置文件，因此为了方便起见，下面展示了 MMWAVE-L-SDK 版本 05.01.00.04 中修改后的配置文件。不同的 MMWAVE-L-SDK 版本可能会以不同的方式对讨论的参数进行命名或排序，但不同版本的 SDK 之间的基本概念不应发生变化。此外，每个参数的详细说明可以在 [2] 中找到，它位于 MMWAVE-L-SDK 的 docs/ 文件夹中。

备注

在线提供的毫米波传感估算器工具为用户提供了一个用于计算线性调频脉冲参数和估算功耗的直观界面。

```

% *****
% MotionDetect: Chirp configuration and Processing chain are
% designed to detect moving objects, including estimation of
% range, velocity and angle of the objects. It is typically
% useful for detecting, localizing and tracking objects in
% indoor or outdoor settings.
% *****
channelCfg 7 3 0
chirpComnCf 8 0 0 256 4 28 0
chirpTimingCf 6 63 0 75 60
frameCfg 2 0 200 64 1000 0
guiMonitor 2 1 0 0 0 1
sigProcChainCfg 16 8 1 0 4 4
cfarCfg 2 8 4 3 0 12.0 0 0.5 0 1 1 1
aoaFovCfg -60 60 -40 40
rangeSelCfg 0.1 12.0
clutterRemoval 1
compRangeBiasAndRxChanPhase 0.0 1.00000 0.00000 -1.00000 0.00000 1.00000 0.00000 -1.00000 0.00000
1.00000 0.00000 -1.00000 0.00000
adcDataSource 0 C:/ti/mmwave_lp_sdk/examples/datapath/common/testBench/major_motion/
adc_data_0001_CtestAdc6Ant.bin
adcLogging 0
lowPowerCfg 1
factoryCalibCfg 1 0 40 3 0x1ff000
mpdBoundingBox 1 -1.5 1.5 0 3.8
sensorStart 0 0 0 0

```

技术 1 - 增加帧之间的时间

说明

帧之间的时间越长，器件处于帧间空闲模式（特别是深度睡眠状态）的时间就越长。由于帧间空闲/深度睡眠状态是器件的最低功率级别，因此增加器件处于深度睡眠的时间会降低总体功耗，这是通过降低较高功耗模式（如活动模式、线性调频脉冲空闲状态和突发间空闲状态）的功耗影响。这使得平均功耗更接近帧间空闲/深度睡眠中的功耗。下面的三个公式以数学方式描述了这一点。

$$PoWer_{Average} = (PoWer_{Interframe Idle} \times Time_{Interframe Idle} + PoWer_{Interburst Idle} \times Time_{Interburst Idle} + PoWer_{Interchirp Idle} \times Time_{Interchirp Idle} + PoWer_{Active} \times Time_{Active}) \div (Time_{Interframe Idle} + Time_{Interburst Idle} + Time_{Interchirp Idle} + Time_{Active}) \quad (1)$$

$$PoWer_{Interframe Idle} < PoWer_{Interburst Idle}, PoWer_{Interchirp Idle}, PoWer_{Active} \quad (2)$$

$$As Time_{Interframe Idle} \gg \gg, PoWer_{Average} \rightarrow PoWer_{Interframe Idle} \quad (3)$$

配置文件位置

参数	命令行参数名称	命令行参数位置	单位
framePeriodicity	frameCfg	5	毫秒

性能影响

增加帧之间的时间会增加延迟并可能提高跟踪性能。如果帧之间的时间很长，则可能无法跟踪快速移动的物体。

示例

在以下示例中，帧周期设置为 1000，这意味着每 1000 毫秒开始一个新帧。

```
frameCfg 2 0 200 64 1000 0
```

技术 2 - 减少线性调频脉冲间空闲和突发间空闲中花费的时间

说明

减少线性调频脉冲间空闲和突发间空闲中花费的时间会增加帧间空闲模式中花费的时间。尽管雷达器件在线性调频脉冲和突发之间会进入低功耗状态，但雷达可以在帧之间进入帧间空闲/深度睡眠模式，这些模式下的功耗要少得多。因此，通过延长处于深度睡眠模式的时间，并缩短处于线性调频脉冲间空闲和突发间空闲的时间，可以降低整体功耗。

配置文件位置

参数	命令行参数名称	命令行参数位置	单位
burstPeriodicity	frameCfg	3	微秒
chirpIdleTime	chirpTimingCfg	1	微秒
chirpRampEndTime	chirpComnCfg	6	微秒
numOfChirpsInBurst	frameCfg	1	无

性能影响

如果线性调频脉冲使用不同的发送器，强烈建议减少线性调频脉冲之间的时间长度。在时分多路复用 (TDM) 和二进制相位调制 (BPM) 方案 (详见 [3]) 中，有效角度计算假设两个发送器大致同时发射。但实际上，它们一次只有一个发射。通过减少使用不同发送器发射的线性调频脉冲之间的时间长度，可以减少此假设产生的误差。

但是，缩短突发之间的时间将影响雷达器件可以检测和识别的速度范围¹。这是因为速度是通过同一发送器/接收器对上的雷达回波差异来测量的。当突发连续发生得更快时，雷达可以检测到的最小速度将会减小。通过微小运动模式仍可检测到小幅运动，该模式会在帧之间查找一些点，但可能无法准确估算这些微小运动点的速度。

备注

突发间空闲时间和线性调频脉冲间空闲时间必须满足第 1 节中讨论的最低要求。要检测更快的速度并将功耗降低到突发方案允许的功耗以下，用户可以考虑使用每帧单次突发和每次突发多个线性调频脉冲，从而消除最小 95 μs 的突发间空闲时间要求。

示例

¹ 这假定使用突发模式而不是正常模式。[2] 中详细介绍了这两种情况。

在下面的示例中，突发周期为 200 微秒，线性调频脉冲空闲时间为 6 微秒，单次突发中的线性调频脉冲数量为 2，线性调频脉冲斜坡结束时间为 28 微秒。因此，线性调频脉冲间空闲时间为 6 微秒，而突发间空闲时间为 132 微秒。

$$\text{Interburst Idle Time} = \text{Burst Periodicity} - (\text{Number of Chirps in Burst} \times (\text{Chirp Idle Time} + \text{Chirp Ramp End Time})) \quad (4)$$

```
chirpComncfg 8 0 0 256 4 28 0
chirpTimingCfg 6 63 0 75 60
frameCfg 2 0 200 64 1000 0
```

技术 3 - 缩短线性调频脉冲频率扫描时间

说明

减少线性调频脉冲频率扫描时间可以通过强制缩短器件进行线性调频脉冲频率扫描的时间来降低功耗。高效的线性调频脉冲设计将确保快速采集所有样本，并且在采集样本前后进行线性调频脉冲频率扫描所用的时间非常少。下面介绍了线性调频脉冲频率扫描之前、期间和之后发生的事件序列。器件进行线性调频脉冲频率扫描所用的总时间称为斜坡结束时间，计算方法如下。

$$\text{Ramp End Time} = \text{ADC Start Time} + \text{Chirp Time} + \text{Excess Time} \quad (5)$$

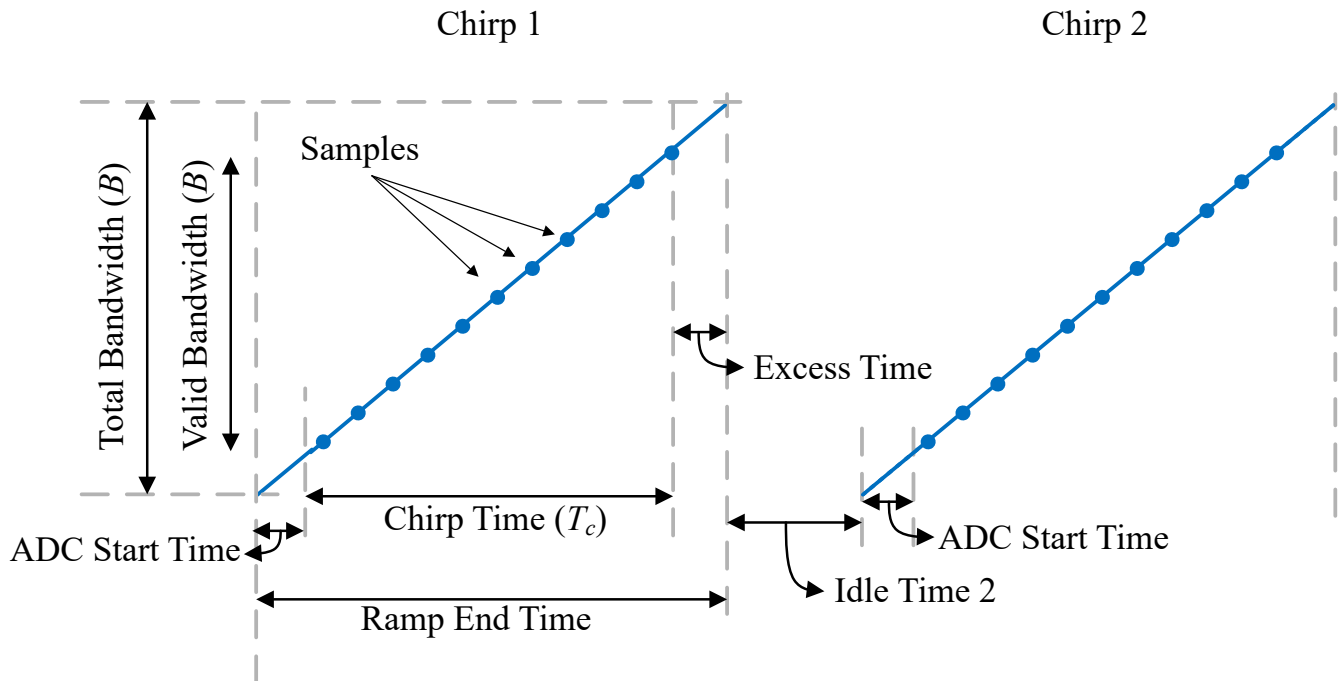


图 3-1. 线性调频脉冲中的事件序列

设置适当斜坡结束时间的过程如下：

1. 计算线性调频脉冲时间，这是收集所有 ADC 样本所需的时间。

$$\text{Chirp Time} = \text{Sampling Rate} \times \text{Number of Samples} \quad (6)$$

2. 设置适当的 ADC 启动时间。典型值约为 5 微秒。ADC 启动时间越短，线性调频脉冲斜率线性度越小。

备注

ADC 启动时间由 ADC 跳过样本数和采样率决定。建议的最小 ADC 跳过样本数为 24 或 $3\mu\text{s} \times \text{采样率}$ (以 MHz 为单位)，并以较大者为准。

3.根据 [2]，对于短滤波器 ($dfeFirSel = 1$)，斜坡超时时间至少应为 7 个样本；对于长滤波器 ($dfeFirSel = 0$)，斜坡超时时间至少应为 10 个样本。

4.将斜坡时间设置为线性调频脉冲时间、ADC 启动时间和斜坡超时时间之和。

$$Ramp\ Time = Chirp\ Time + ADC\ Start\ Time + Ramp\ Excess\ Time \quad (7)$$

缩短斜坡时间需要更大限度地减小其全部三个组成部分。ADC 启动时间和斜坡超时时间可以设置为其允许的最小值，而不会影响范围和速度测量。更快的采样率也可以缩短线性调频脉冲时间。但是，如果线性调频脉冲数和频率斜率保持恒定，减少线性调频脉冲时间会导致带宽降低且距离库大小加粗（尽管有助于增加最大距离）。有效带宽 (B) 和距离分辨率的计算方式如下，其中 c 等于光速 (3E8m/s)：

$$B = ADC\ Sampling\ Time \times Frequency\ Slope \quad (8)$$

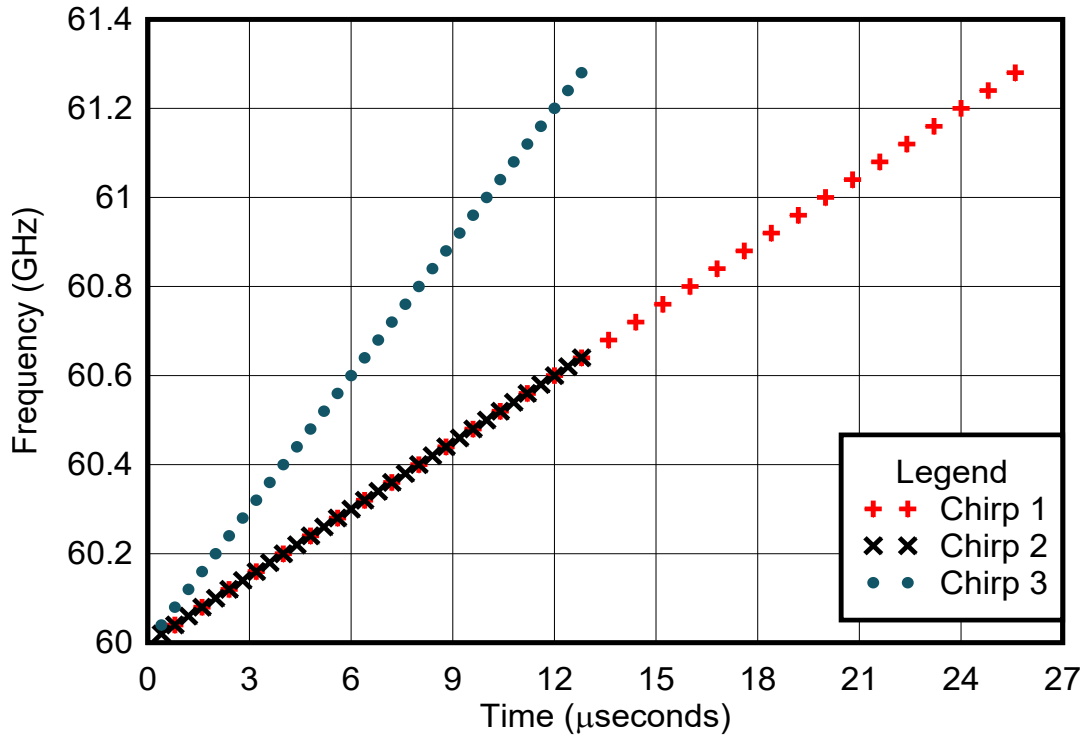
$$Range\ Resolution = \frac{c}{2B} \quad (9)$$

下面的线性调频脉冲 2 说明了这种增加采样率的策略，该线性调频脉冲具有与线性调频脉冲 1 完全相同的参数，但采样速度更快，从而导致距离分辨率相对较粗。但是，如果器件以相同的系数增加线性调频脉冲的采样率和频率斜率，则可以在不降低带宽的情况下缩短活动时间，如下面的线性调频脉冲 3 所示。通过以更快的速率在相同带宽上收集相同的数据，该器件可以缩短线性调频脉冲频率扫描时间并增加处于较低功耗模式下的时间，从而降低总体功耗。

备注

线性调频脉冲 1 至 3 旨在说明权衡取舍，而不是显示最佳性能。

	线性调频脉冲 1 (采样率较慢)	线性调频脉冲 2 (采样率较快)	线性调频脉冲 3 (采样率较快且频率斜率较大)
ADC 采样频率	1.25MHz	2.5MHz	2.5MHz
ADC 样本数	32	32	32
线性调频脉冲时间	25.6μs	12.8μs	12.8μs
频率斜率	50MHz/μs	50MHz/μs	100MHz/μs
有效带宽	1280 MHz	640 Mhz	1280 MHz
距离分辨率	0.117 m	0.234 m	0.117 m
最大距离	1.5m	3 m	1.5m



配置文件位置

参数	命令行参数名称	命令行参数位置	单位
digOutputSampRate (ADC 采样率)	chirpComnCfg	1	抽取值。采样率由 100MHz/digOutputSampRate 设置
chirpRfFreqSlope	chirpTimingCfg	4	MHz/µs
chirpRampEndTime	chirpComnCfg	6	微秒
chirpADCSkipSamples	chirpTimingCfg	2	采样数

性能影响

缩短线性调频脉冲时间将减小雷达可检测到人员和物体的最大距离。有关更多信息，请参阅 [4] 中的雷达距离公式。

示例

```
chirpTimingCfg 20 63 0 30 60
chirpComnCfg 8 0 0 256 4 70 0
```

在上述配置示例中，通过将 100MHz 除以设为 20 的 digOutputSampRate 来获得采样率，从而得到 5MHz。

$$\text{Sampling Rate} = \frac{100 \text{ MHz}}{\text{digOutputSampRate}} = \frac{100 \text{ MHz}}{20} = 5 \text{ MHz} \quad (10)$$

ADC 跳过样本数设置为 63，这意味着采样仅在 12.6 µs 后开始

$$\text{Sample Starting Time} = \frac{63 \text{ Samples}}{5 \text{ MHz}} = 12.6 \text{ µsec} \quad (11)$$

12.6 µs 的时间远远足够让线性调频脉冲斜坡斜率变为可靠的线性。将其缩短至 5µs 可以节省功耗。这对应的样本数为 25，比最小值 24 大，因此可以接受，从而生成以下配置文件。

$$5 \mu\text{sec} \times 5 \text{ MHz} = 25 \text{ samples} \quad (12)$$

```
chirpTimingCfg 20 25 0 30 60
chirpComncCfg 8 0 0 256 4 70 0
```

现在，ADC 启动时间为 5 微秒。

线性调频脉冲时间等于样本数除以采样率。

$$\text{Chirping Time} = \frac{256 \text{ Samples}}{5 \text{ MHz}} = 51.2 \mu\text{sec} \quad (13)$$

由于 dfeFirSel 等于 0，因此建议按照 [2] 中的说明对斜坡结束时间（2 微秒）进行 10 次采样。

$$10 \text{ samples} \times \frac{1}{5 \text{ MHz}} = 2 \mu\text{sec} \quad (14)$$

总体而言，此线性调频脉冲需要 51.2（线性调频脉冲时间）+ 5（ADC 启动时间）+ 2（斜坡超时时间）= 58.2 微秒。因此，斜坡结束时间可以从 70 微秒减少到 59 微秒，从而在高功率线性调频脉冲频率扫描状态下节省 11 微秒的时间。

```
chirpTimingCfg 20 25 0 30 60
chirpComncCfg 8 0 0 256 4 59 0
```

现在，线性调频脉冲已经过优化来减少 ADC 数据采样之前和之后的线性调频脉冲频率扫描时间，下面我们可以开始减少采集数据所花费的时间。目前，线性调频脉冲的有效带宽可以按如下方式计算：

$$B = \text{Frequency Slope} \times \text{Chirp Time} = 30 \frac{\text{MHz}}{\mu\text{sec}} \times 51.2 \mu\text{sec} = 1536 \text{ MHz} \quad (15)$$

通过将 chirpRfFreqSlope 和采样率增加相同的因子 2，可以缩短此线性调频脉冲的活动时间。因此，采样率变为 10MHz（相当于 digOutputSampRate 为 10），而线性调频脉冲频率斜率 chirpRfFreqSlope 变为 60。

```
chirpTimingCfg 10 25 0 60 60
chirpComncCfg 8 0 0 256 4 54 0
```

现在采样率等于 100/10 = 10MHz，ADC 跳过样本数应增加到 50，以便将 ADC 启动时间保持在恒定的 5 微秒。

```
chirpTimingCfg 10 50 0 60 60
chirpComncCfg 8 0 0 256 4 54 0
```

但是，线性调频脉冲频率扫描时间现在等于 25.6 微秒。因此，ADC 启动时间和线性调频脉冲时间之和现在等于 30.6 微秒。为斜坡超时时间再分配 10 个样本需要 1 微秒，这时斜坡结束时间至少为 31.6 微秒。因此，斜坡结束时间可以设置为 32 微秒，与最初的 70 微秒相比，线性调频脉冲频率扫描时间缩短了 54%。

```
chirpTimingCfg 10 50 0 60 60
chirpComncCfg 8 0 0 256 4 32 0
```

技术 4 - 减少发送器/接收器的数量

说明

减少发送器或接收器的数量将会减少活动模式下的功耗。通过减少发送器和接收器的数量，可以减少任何特定线性调频脉冲行为中为发送器和接收器电路供电所需的功率。如果设备通过 TDM（请参阅 [3]）使用多个发射天线，则减少发送器数量也会减少设备线性调频脉冲频率扫描所花费的时间，从而降低功耗。发送器的功耗要高于接收器，因此 TI 建议先减少发送器数量，再减少接收器数量。

配置文件位置

参数	命令行参数名称	命令行参数位置	单位
rxChCtrlBitMask	channelCfg	1	指示哪些接收器处于打开/关闭状态的位掩码。
txChCtrlBitMask	channelCfg	2	指示哪些接收器处于打开/关闭状态的位掩码。

性能影响

减少发送/接收通道的数量将直接降低雷达器件的角分辨率。雷达的角分辨率定义为区分同一距离库中的两个物体所需的最小间距，即 $2/N$ 弧度，其中 N 是该角度下的发送/接收通道数（方位角与仰角）。例如，IWRL6432BOOST 评估板在方位角有 4 个通道，在仰角有 2 个通道。因此，方位角的角分辨率为 $2/4$ 弧度 = 28.6 度，仰角的角分辨率为 $2/2 = 1$ 弧度 = 57 度。移除发送器会将可用通道数量从 6 个减少到 3 个，这将降低方位角和/或仰角的角分辨率，具体取决于所选的天线布局。

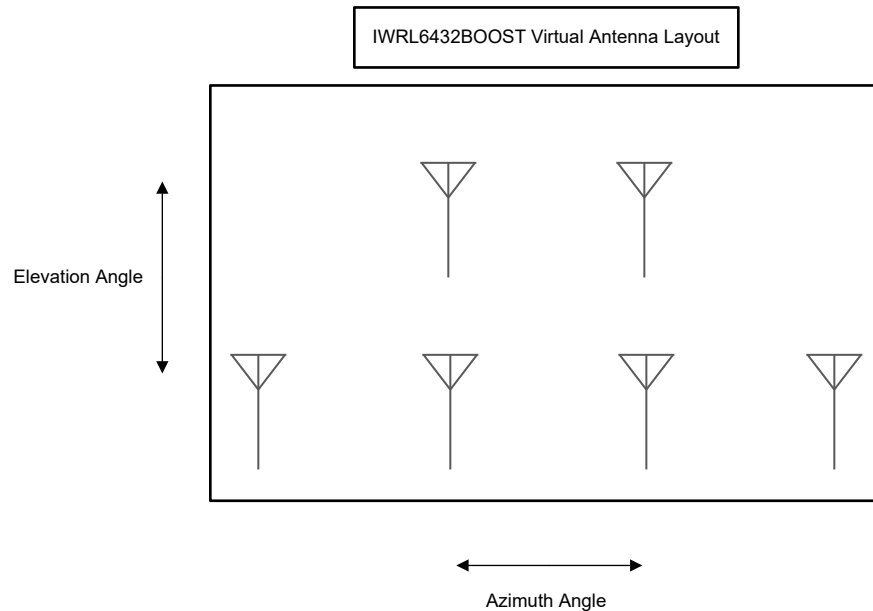


图 3-2. IWRL6432BOOST 虚拟天线布局

此外，移除发送/接收对将减小测量的 SNR，从而减小雷达可检测到给定目标的最大距离。发送/接收通道越多，对同一突发的观察就越多，因此可以提高给定突发的精度。

示例

rxChCtrlBitMask 可以设置为 0-7 之间的值。当转换为二进制时，数字集表示哪些接收器已上电（由 1 表示）以及哪些接收器已关断（由 0 表示）。在以下示例中，值 6 = 0b110 时会打开第二个和第三个接收器，同时关闭第一个接收器。

TxChCtrlBitMask 可以设置为 0-3 之间的值。在与 rxChCtrlBitMask 相同的过程中，当转换为二进制时，数字集表示哪个发送器已上电（由 1 表示）以及哪些发送器已关断（由 0 表示）。在以下示例中，值 3 = 0b11 时，两个发送器都会打开。

```
channelCfg 6 3 0
```

技术 5 - 降低发送功率

说明

要降低线性调频脉冲期间的功耗，最简单的方法之一是减少从雷达器件广播的功率大小。用户可以设置 TX 回退功率，这是要从传输出功率最大值 11dBm 降低的功率大小。

$$Total\ Output\ PoWer = Maximum\ Output\ PoWer - TX\ Backoff\ Value \quad (16)$$

配置文件位置

参数	命令行参数名称	命令行参数位置	单位
txBackoff	factoryCalibCfg	4	dB

性能影响

降低输出功率可能会缩小器件可检测的最大距离。器件广播的功率大小将决定器件可接收的功率大小。一旦接收到的功率降至器件的本底噪声以下，器件将无法再准确检测目标。输出功率的下降可以在一定程度上通过 **CFAR** 阈值范围内的等效下降得到补偿，但不能无限度地进行补偿。一旦 **CFAR** 阈值范围变得过低，误报检测就会掩盖真实检测。

示例

在以下示例中，由于 **factoryCalibCfg** 行的第四个值设置为 3，因此总输出功率将等于 $11-3 = 8\text{dBm}$ 。

```
factoryCalibCfg 1 0 40 3 0x1ff000
```

4 降低帧间空闲和深度睡眠状态下的功耗

降低空闲状态下的功耗

帧间空闲状态是器件中的一种软件可编程状态。在此模式下，器件要么等待来自外部主机的命令，要么通过 SPI 或 CAN 接口传输捕获的样本。器件中未在进行雷达处理或数据采集，同时器件的振荡器电路和可能的 PLL（取决于所使用的主机接口外设）已启动并正在运行。在帧间空闲状态下，用户可以将器件的一些部分断电或进行时钟选通，从而降低功耗。

xWRL6432 允许使用如下所示的序列关闭以下外设。

备注

MMWAVE-L-SDK 中的存在和运动检测演示示例已在不使用时关闭以下外设。

要关断的子系统	关断方法
硬件加速器	HWA_close(HWA_Handle handle); HWA_deinit()
前端控制器子系统	MMWave_stop(gMmwMssMCB.ctrlHandle,&err); MMWave_close(gMmwMssMCB.ctrlHandle,&err); MMWave_deinit(gMmwMssMCB.ctrlHandle,&err);
UART	Drivers_uartClose() * 作为 sysconfig 的一部分包含在 Drivers_close() 中。
I2C	Drivers_i2cClose() * 作为 sysconfig 的一部分包含在 Drivers_close() 中。
EDMA	Drivers_edmaClose() * 作为 sysconfig 的一部分包含在 Drivers_close() 中。
QSPI	Drivers_qspiClose() * 作为 sysconfig 的一部分包含在 Drivers_close() 中。

计时

用户可以选择通过更改外设的时钟选项来降低其功耗，而不是完全关闭外设。实现这一目标的最常见方法如下：

- *更改输入时钟源*

许多外设可以由多个不同的时钟源提供时钟。通常，时钟源越快，外设的运行期间的功耗就越高。[\[5\]](#) 列举了可用于更改外设输入时钟的时钟控制 (CLKCTL) 寄存器。

- *通过时钟分频器进一步降低时钟*

设置外设的时钟源后，可以对其进行分频以降低其频率，从而降低其功耗。通过上面 [\[5\]](#) 中提到的相同时钟控制 (CLKCTL) 寄存器，用户可以将输入时钟向下分频至所需频率。

- *时钟门控*

时钟门控可以阻止向外设提供输入时钟。这意味着外设将不会运行，直到它不再被时钟门控。与完全关闭外设相比，时钟门控功耗更高，但从关闭状态恢复上电所需的时间和能耗也更少，这有利于更快地进行响应。上述 [\[5\]](#) 中所述的相同时钟控制 (CLKCTL) 寄存器允许用户对外设的输入时钟进行门控。

降低深度睡眠模式下的功耗

深度睡眠是器件可能的最低硬件功耗状态。在深度睡眠模式中，几乎整个器件都处于关断状态，可节省大量功耗。深度睡眠模式下只会保留器件上的重要内容（如应用程序映像和线性调频脉冲剖面）。这些会保留器件的上下文，使其无需重新启动即可恢复正常工作。

在存在检测等低占空比应用中，深度睡眠模式的优势最为突出，在这些应用中，器件仅在短时间内处于活动状态。在其余时间内，当器件不需要处于活动状态时，系统可以进入深度睡眠模式，从而为电池供电应用节省功耗并延长使用寿命。

深度睡眠状态下的功耗主要取决于两个因素：深度睡眠期间保留的存储器容量和 IO 引脚的状态。

深度睡眠中 xWRL6432 上的存储器保持

xWRL6432 具有 1MB 的存储器，这些存储器在 APPSS、FECSS 和 HWASS 电源域之间分配。要保留内容的存储器越大，深度睡眠模式下的功耗越高。

表 4-1. xWRL6432 主存储器映射

器件型号	APPSS_RAM	RADAR_DATA_CUBE_RAM	SHARED_RAM ²	FECSS_RAM
xWRL6432	512KB	256KB	256KB	32KB

APPSS_RAM 和 SHARED_RAM 进一步分为不同的存储器组，如下所示。

表 4-2. APPSS 和共享 RAM 存储器组

存储器组	存储器位置	存储器空间 (KB)
组 1	APPSS	256
组 2	APPSS	128
组 3	APPSS	128
Shared_RAM_1	共享 RAM	128
Shared_RAM_2	共享 RAM	128

xWRL6432 器件可以通过减少深度睡眠模式中保留的存储器容量来节省功耗。在 xWRL6432 上，存储器实例在关断保持模式下的功耗约为 45 μ W/64kB。如果不保留集群内的存储器内容，则功耗约为 20 μ W/64kB。为了在不保留存储器内容的情况下进一步节省功耗，xWRL6432 上的一些存储器集群组合在一起，并且电源开关部署在相应

² SHARED_RAM 由 APPSS_RAM 和 RADAR_DATA_CUBE_RAM 共享

的 SRAM 电源轨上。如果某个组中的所有集群都不用在深度睡眠期间保留内容，则该组的开关会断开，从而完全关断该组，使得存储器完全关闭（不保留）且功耗为 0 μA/KB。

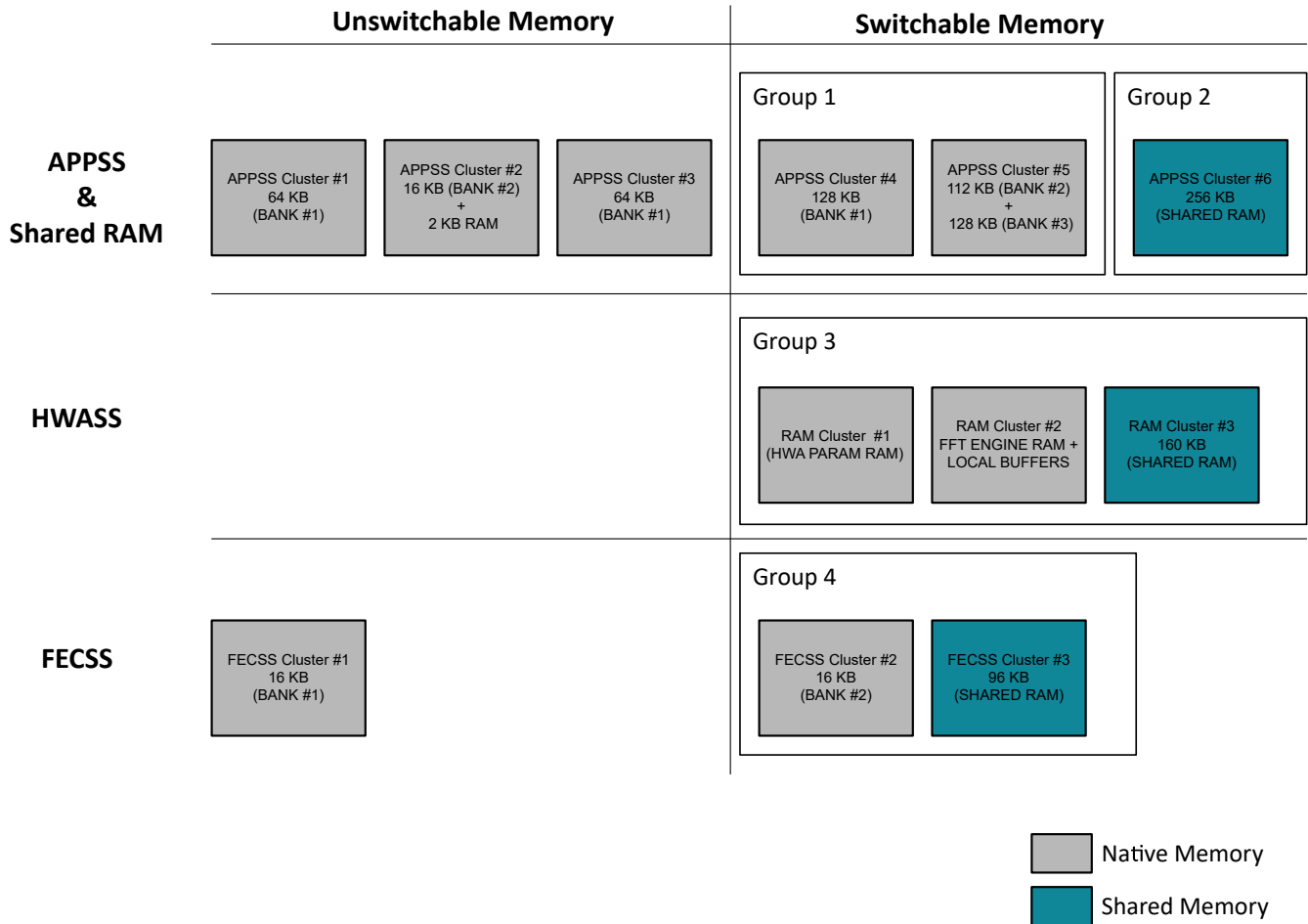


图 4-1. 可切换和不可切换存储器

CAUTION

FECSS 集群 #1 和 #2 完全由 DFP 处理。如果不保留这些集群上的存储器内容，可能会导致意外和错误的行为。

可以使用 [SysConfig](#) (可用于配置许多 TI 器件的 TI 工具) 来关断特定的存储器组。禁用 RAM 保持时，请确保后续帧中操作所需的数据 (如器件代码) 都不会存储在未保留的 RAM 中。如果发生这种情况，器件将无法正常运行，因为器件运行所需的信息将被擦除。

数字 IO 电源的影响

xWRL6432 中存在各种数字外设 IO (例如 SPI、UART、CAN-FD、I2C 或 LIN 等)，用于传输数据、下载映像或与外部 MCU 建立连接。由于深度睡眠状态下的功耗非常低，因此任何悬空节点都可能导致严重的电流泄漏。因此，当 xWRL6432 器件处于深度睡眠模式时，必须适当地将这些 IO 置于其相关的上拉/下拉或高阻抗状态。同样，这可以使用 [SysConfig](#) 来实现。

为了更大限度地降低功耗，TI 建议将所有可用引脚置于深度睡眠模式。用作 xWRL6432 唤醒源的引脚可以保持其默认输入状态。此外，如果从 xWRL6432 输出信号的引脚需要在器件处于深度睡眠状态时保持其状态 (例如，到另一个器件或 LED 灯的复位线路)，也可以保持其默认输出状态。

5 测量功率

IWRL6432BOOST 上的功率测量

利用 IWRL6432BOOST EVM，用户可以通过板载 INA 电流检测放大器测量 xWRL6432 器件的功耗。xWRL6432 SDK 中提供的运动和存在检测演示包含一个输出，该输出会显示电源轨上提供给器件的平均电流测量值。此外，当无法在 xWRL6432 器件上记录输出（例如测量深度睡眠功率）时，也可以从外部微控制器通过 I2C 接口读取 INA 电流检测放大器。有关如何启用和解释功率测量数据包的详细信息，请参阅 [节 6\[2\]](#)。

CAUTION

由于板载 INA226 器件存在的限制，Rev A 和 Rev B IWRL6432BOOST 无法精确测量功率。Rev A 和 Rev B IWRL6432BOOST 需要进行以下更改才能正确测量功率。

参考标识符	初始器件	替换器件
R200、R192、R193	PMR18EZPFV2L00 2m Ω 电阻器	WSL1206R0400FEB 40m Ω 电阻器
R134	PMR18EZPFV2L00 2m Ω 电阻器	WSL1206R0200FEB 20m Ω 电阻器
U6、U7、U8、U25	INA226AIDGST 电流监控器	INA228AIDGST 电流监控器

6 参考文献

1. [IWR6432 单芯片 57GHz 至 64GHz 工业雷达传感器数据表](#)
2. [运动和存在检测演示调优指南](#)
3. [MIMO 雷达](#)
4. [《对 TI 雷达器件中的啁啾参数进行编程》](#)
5. [IWR6432 技术参考手册](#)

7 修订历史记录

日期	修订版本	说明
2023 年 4 月	*	初始发行版

重要声明和免责声明

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、某特定用途方面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功能安全、信息安全、监管或其他要求。

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 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/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。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2023，德州仪器 (TI) 公司