

AFE79xx AGC+LNA Bypass 参数设置

Lijia Zhu

摘要

灵敏度和动态范围是无线接收机的两个重要指标，为了提高系统灵敏度，会在接收机前级加入 LNA（Low Noise Amplifier，低噪声放大器）；为了增加接收机的动态范围，可以使用 AGC（Automatic Gain Control，自动增益控制）功能。对于一定功率范围内的输入信号，通过 DSA（数控步进衰减）和 DGC（数字增益补偿）模块，在确保接收机数模总增益不变的前提下，提高系统的动态范围：信号功率超过上门限则增大 DSA 衰减模拟信号，DGC 提高数字增益进行相应的补偿；信号功率低于下门限则减小 DSA，数字增益的补偿值也相应减少。对于更大功率的输入信号，需要把前级 LNA 旁路以防止接收机被阻塞。目前业界广泛使用的 TI Transceiver 芯片有基于零中频架构的 AFE77xx 系列和基于射频采样架构的 AFE79xx 系列，它们都集成了实现 AGC 功能的所有所需模块，其中 AFE79xx 因其高集成度、支持宽带、多频等优点，广泛应用于 4G/5G RRU/AAU 等场景。本文将 AFE79xx 为例，对 AGC 功能，尤其是结合了 LNA Bypass 之后 AGC 环路的关键参数的设置进行介绍。

目录

1	引言.....	3
2	AGC 架构	3
	2.1 AGC 简介.....	3
	2.2 峰值/功率检测器.....	4
	2.2.1 Digital Peak Detectors at ADC Output.....	4
	2.2.2 Digital Power Detectors.....	5
	2.2.3 Band Detectors for Dual LNA Control.....	6
	2.2.4 RF Peak Detector.....	7
	2.3 决策状态机.....	7
	2.4 DSA 和 DGC.....	9
	2.4.1 DSA 与 DSA 校准.....	9
	2.4.2 DGC.....	10
3	AGC + LNA Bypass	12
	3.1 扩大 AGC 的范围.....	12
	3.2 LNA Bypass 参数设置.....	13
4	典型应用场景举例	15
	4.1 场景需求示例.....	15
	4.2 C 函数调用示例.....	16
5	参考文献	17

图

Figure 1.	AGC 控制逻辑示意图.....	3
Figure 2.	AGC 结构框图	4
Figure 3.	Digital Peak Detector 框图	5
Figure 4.	Band Detector 框图	6
Figure 5.	RF Peak Detector 框图	7
Figure 6.	单频场景 AGC 状态机控制流程图	8
Figure 7.	Floating Point 和 Coarse+Fine 模式对比图	10
Figure 8.	DSA 和 DGC 的关系简图	10
Figure 9.	四种可能的解码方式示意图.....	11
Figure 10.	外接 DVGA 扩大 AGC 范围的系统框图.....	12
Figure 11.	使用 LNA bypass 扩大 AGC 范围的系统框图.....	12
Figure 12.	达到 Max Atten 使能 LNA bypass 示意图.....	13
Figure 13.	达到 Early DSA 使能 LNA bypass 示意图	14
Figure 14.	典型应用场景的衰减示意图.....	16

1 引言

灵敏度定义为接收机可以检测到的符合系统性能要求的最小信号电平。在解调信噪比（SNR）一定的情况下，灵敏度由热噪声底（-174dBm/Hz）、噪声带宽（BW）和系统的噪声系数（NF）共同决定，一般而言，热噪声底和带宽不能随意改变，为了提高系统的灵敏度，选择噪声系数更低的器件或者在前级加入 LNA 以降低系统的总噪声系数。动态范围粗略的定义为接收机可以接收到的最大输入电平和可以检测到的最小输入电平（灵敏度）之比。使用 AGC 功能可以扩大接收机的动态范围：当输入端出现阻塞信号功率超过设置的门限时，降低链路的模拟增益以防止 ADC 出现饱和，同时提高数字增益以保证模数总增益恒定；当阻塞信号功率低于设置的门限时，提高模拟增益同时降低数字增益。数字增益的调节由 DGC（ALC）模块完成，模拟增益的调节由 DSA 模块和 LNA bypass 功能共同完成。

AFE79xx 是以射频采样（RF sampling）架构为基础，拥有 4 个完整的发射和接收通道和 2 个反馈通道，发射和接收/反馈通道依次包含了 JESD RX/TX, DUC/DDC, DAC/ADC 和 DSA 模块，支持单芯片的射频收发功能（Radio on chip）的高集成射频收发器（transceiver）。AGC 是接收通道重要的功能，主要包含三个功能模块：输入信号功率检测、决策状态机和模数增益控制部分。在 AGC 模式下，无需 ASIC/FPGA 的介入（除 DGC Slicer 的解码外）AFE 即可按照预设的参数完成对输入通道的自动增益控制。若只使用功率检测和增益控制部分，将逻辑判决交给外部的 ASIC/FPGA，则被称为 MGC 或开环 AGC。一般而言，使用外部 AGC 可以获得更高的灵活性，但对于系统软件的控制能力提出更高的要求。因此我们建议使用 transceiver 内部的 AGC 以简化系统的设计，本文将仅对闭环 AGC（internal AGC）的工作原理、参数的设置以及如何使用 C 函数设置参数进行介绍（正文中所有的斜体英文为对应 C 函数中的形参名）。

2 AGC 架构

2.1 AGC 简介

受无线信道的影响，信号传输过程中会发生以路径损耗为代表的慢衰落，例如当手机接近基站时，信号幅度可以到-10dBm，在地下室则信号幅度会低于-110dBm；和以多径效应为代表的快衰落，在很短的时间内，无线接收机的输入信号强度波动很大，因此要求接收机的动态范围尽量大。如图 1 所示，输入信号为大信号时（超过 Attack 门限），ADC 之前的模拟链路增益需要实时降低，以避免 ADC 的溢出。当输入阻塞信号消失（低于 Decay 门限），ADC 前的模拟链路增益回调，以追求最佳的灵敏度。ADC 之后的数字部分则根据模拟链路增益的变化对信号在数字域进行补偿以保持数模总增益的恒定。同时，为了避免乒乓效应，Attack 门限和 Decay 门限之间一般需要留有一定的回滞区间，这个区间的大小根据 Detector 的精度和系统需求而定。

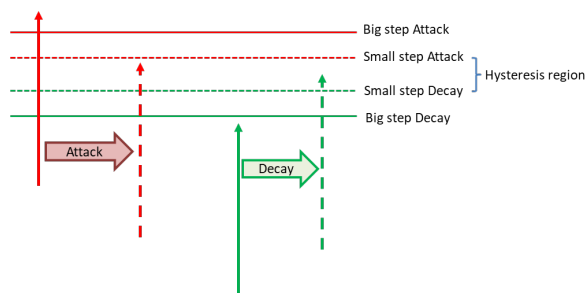


Figure 1. AGC 控制逻辑示意图

如图 2 所示，输入信号功率检测、决策状态机和模数增益控制三部分共同构成了完整的 AGC 环路，它是一个负反馈环路。其中，输入信号功率检测部分由若干个模拟或数字的峰值/功率检测器组成，在整个 AGC 链路中是观察者的角色，主要通过检测接收到的信号强度并和预设的门限值进行比较，向控制器输出检测结果；决策状态机是 AGC 的核心，根据 Detector 的检测结果来决定它的输出；模数增益控制部分由 DSA/LNA bypass 和 DGC 构成，决策状态机的输出由 DSA 执行，DGC 根据 DSA 的衰减量在数字域对信号幅度实时进行补偿。LNA bypass 功能有两种使用场景，第一种是双频场景，基于 Band Detector 的输出对相应通道的 LNA bypass 以达到调节单个频带动态范围的目的；另一种更加常用的应用场景是当 DSA 超过设定的门限值，LNA bypass 功能被触发，旁路外部 LNA 以扩大系统的总动态范围。除了 LNA bypass 外，其他功能在 AFE79xx 内部被完整集成（图 2 绿色方框内），接下来的内容会对这几个模块的工作方式和参数设置进行介绍。对于唯一需要外部器件配合使用的 LNA bypass 功能，在第三章将对如何使用 LNA Bypass 功能扩大 AGC 范围及其有关配置进行详述。

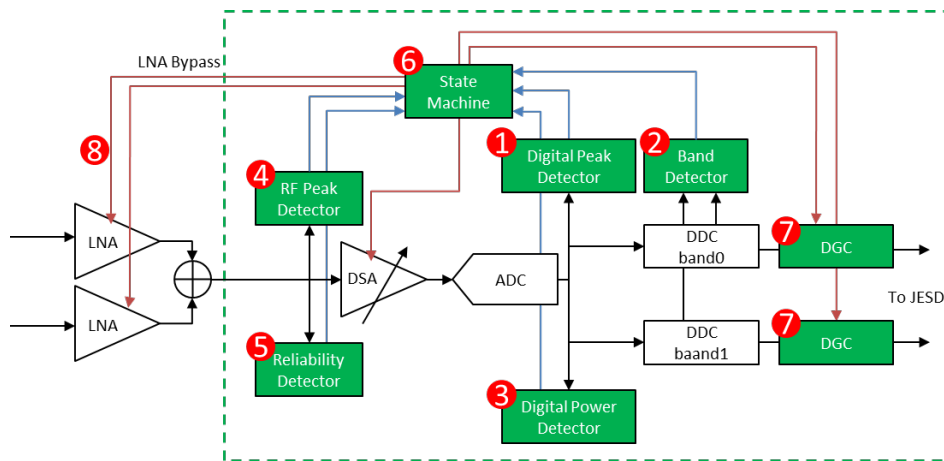


Figure 2. AGC 结构框图

2.2 峰值/功率检测器

按照不同的功能划分，AFE79xx 内部集成了四种 Detector。他们具有不同的观测带宽和独立的 attack/decay 门限设置。

2.2.1 Digital Peak Detectors at ADC Output

每一路 RX 通道都有四个独立的 Digital Peak Detectors，他们可以被独立配置：Big Step Attack Detector、Small Step Attack Detector、Big Step Decay Detector 和 Small Step Decay Detector。这些 Detector 位于 ADC 之后，DDC 之前（图 2 中红色编号 1），因此具有全频段的观测带宽和仅次于 RF Peak Detector 的时延性能。

Digital peak detector 的工作原理如图 3 所示，每 8 个数据点（ $f_s/8$ ）中的最大值和设置的门限值（ $AttkThresh/DecThresh$ ）比较，记录一定时间内（ $WinLen$ ）超过门限的次数，对于 Attack Detector，如果这段时间内超过门限的次数超过设定值（ $NumHits$ ）则输出 Attack Detector 被触发的状态给到 AGC 逻辑状态机；对于 Decay 门限，如果给定时间内超过门限的次数少于设定值，则输出 Decay Detector 被触发的状态给 AGC 状态机。

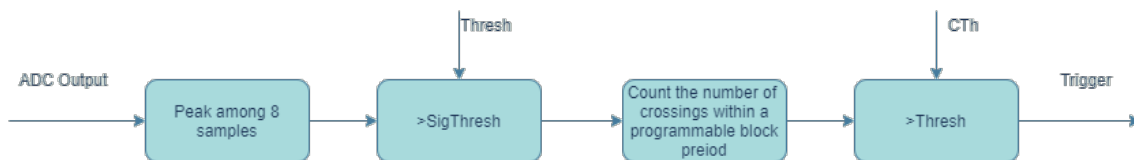


Figure 3. Digital Peak Detector 框图

Small Step Attack 适用于幅度变化相对缓慢的信号，一般 Attack 门限值可以设置在-3dBFS 左右，Decay 门限设置在-6dBFS 左右，观测窗长的设置基于“快衰慢回调”的准则，Attack 观测窗长一般为 0.1us 级别，Decay 观测窗长一般为 10ms 级别。而 Big Step Attack 适用于需要快速衰减的场景，一般设置的门限要比 Small Step Attack 要更高，观测窗长更短，同时设置的步进也会更大。实际中，为简化 AGC 的控制逻辑，仅使能 Small Step Detector 可以满足大部分系统的需求。

2.2.2 Digital Power Detectors

Digital Power Detector 和 Digital Peak Detector 具有相同的采样位置（图 2 中红色编号 3），因此具有类似的观测带宽和时延性能。不同点在于，每个 RX 端口，只有 Attack 和 Decay 两个 Power Detector。它统计信号的功率有效值而非峰值，对信号包络的突变不敏感，对于无线基站等应用场景，一般不需要使能该 Detector。

除了修改芯片的启动配置（Bringup config）外，还可以很方便的使用提供的 C 函数对 Digital Power Detector 和 Digital Peak Detector 的参数进行设置，函数原型后注释为函数功能的简单描述，以及关键参数的简要说明，完整注释请参考 C 函数源代码（C 函数会根据需要进行更新，本文使用的是 V2p5 版本的 C 函数）：

```
uint8_t agcDigDetConfig(uint8_t afeld, uint8_t chNo, uint8_t bigStepAttkEn, uint8_t
smallStepAttkEn, uint8_t bigStepDecEn, uint8_t smallStepDecEn, uint8_t powerAttkEn, uint8_t
powerDecEn, uint8_t bigStepAttkThresh, uint8_t smallStepAttkThresh, uint8_t
bigStepDecThresh, uint8_t smallStepDecThresh, uint8_t powerAttkThresh, uint8_t
powerDecThresh) // ADC Digital Detector Threshold configuration
```

```
uint8_t agcDigDetTimeConstantConfig(uint8_t afeld, uint8_t chNo, uint32_t bigStepAttkWinLen,
uint32_t miscStepAttkWinLen, uint32_t decayWinLen) // ADC Digital Detector Window
Length configuration
```

```
uint8_t agcDigDetAbsoluteNumCrossingConfig(uint8_t afeld, uint8_t chNo, uint32_t
bigStepAttkNumHits, uint32_t smallStepAttkNumHits, uint32_t bigStepDecNumHits, uint32_t
smallStepDecNumHits) // ADC Digital Detector Absolute NumHits configuration
```

```
uint8_t agcDigDetRelativeNumCrossingConfig(uint8_t afeld, uint8_t chNo, uint32_t
bigStepAttkNumHits, uint32_t smallStepAttkNumHits, uint32_t bigStepDecNumHits, uint32_t
smallStepDecNumHits) // ADC Digital Detector Relative NumHits configuration
```

关键参数说明：

afeld: AFE ID，用于区别系统中多片 AFE 芯片

chNo: Bit wise channel select，每个 bit 代表一个通道，选中全部四个通道为 0xf

bigStepAttkEn/smallStepAttkEn/bigStepDecEn/smallStepDecEn/powerAttkEn/powerDecEn: 对应 Detector 的使能

bigStepAttkThresh/smallStepAttkThresh/bigStepDecThresh/smallStepDecThresh/powerAttkThresh/powerDecThresh: 设置对应 Detector 的门限值，设定值= -参数值/4

bigStepAttkWinLen: Big step attack Detector 和 RF Detector 的观测窗长设置，设定值= 参数值*10ns

miscStepAttkWinLen: 其他 attack Detector 的观测窗长设置，设定值= 参数值*10ns

decayWinLen: 所有 decay Detector 的观测窗长设置，设定值= 参数值*10ns

bigStepAttkNumHits/smallStepAttkNumHits/bigStepDecNumHits/smallStepDecNumHits: 在 *agcDigDetAbsoluteNumCrossingConfig* 函数中为对应 Detector 的计数门限值（绝对量）；在 *agcDigDetRelativeNumCrossingConfig* 中为对应 Detector 的计数门限值（相对量），设定值= 参数值*观测窗长/2¹⁶

2.2.3 Band Detectors for Dual LNA Control

AFE7920/88 支持双频应用场景，RX 通道具有双路 DDC，如果系统设计中为每个 band 配置独立的 LNA，在这个应用场景下，可以根据哪一个 band 有阻塞信号来 bypass 对应 band 的 LNA 起到独立控制两个 band 动态范围的功能。Band Detectors 正是为这个需求所设计的。它位于每一路的 DDC 之后（图 2 中红色编号 2），每个 DDC 具有 Attack 和 Decay 两个 Detector，它的可观测的带宽较窄且和抽头位置有关（*tapOffPoint*，DDC 对数据进行降采样是流水线方式进行的，有若干中间节点，抽头就是在这些中间速率的节点），一般设置为对应 band 的带宽（最后一个抽头位置），也可以设置为比 band 带宽更宽（中间节点取数据）。它的结构如图 4 所示，和 Digital Peak Detector 类似。如果是单频场景，不建议使能 Band Detector。

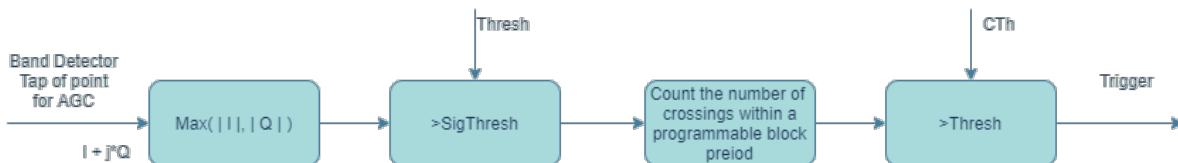


Figure 4. Band Detector 框图

Band detector 和 LNA 的有关的参数被封装在同一个 C 函数中：

```
uint8_t extLnaConfig(uint8_t afeld, uint8_t chNo, uint8_t singleDualBandMode, uint8_t
lnaGainMargin, uint8_t enBandDet, uint8_t tapOffPoint) // External LNA Configuration
```

关键参数说明：

singleDualBandMode: 0 为单频模式，1 为双频模式

enBandDet: 仅在双频场景有效，置 1 使能 Band detector

tapOffPoint: 仅在双频场景且 band detectors 使能情况下有效，0 为前级抽头，1 为末级抽头

2.2.4 RF Peak Detector

RF Peak Detector 位于 DSA 之前，是模拟 Detector（图 2 中红色编号 4），每个 RX 端口只有一个 Attack Detector，它能观测全带宽，时延最小，响应最为迅速，一般用作非常大信号的 Attack（Very Big Step Attack），门限值一般高于 Digital Peak Detector 的 Attack 门限 2-3dB，增益衰减的步进也更大。因为它是模拟的检测，精度较低，且功能上和 Digital Peak Detector 重复，作为一个可选功能，一般也不建议使用。

RF Peak Detector 的工作原理如图 5 所示，统计一定时间段内（*bigStepAttkWinLen*）超过门限值（*rfdetThreshold*）的次数，若超过设定的门限（*rfdetnumhits*），就会向 AGC 状态机送出 RF Detector 被触发的信号。与 Digital Detector 不同，它的实际门限值会根据 DSA 的值进行调节，因此等效到 DSA 之后，它的值是一个恒定的数值。我们设置门限的也正是这个等效之后的值。

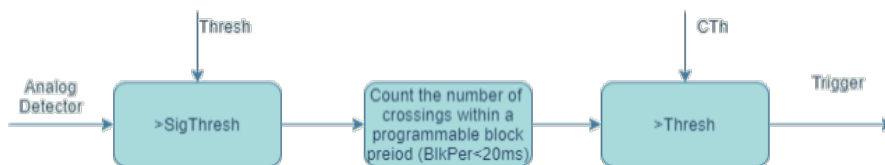


Figure 5. RF Peak Detector 框图

可以使用 C 函数配置 RF Peak Detector 的参数：

```

uint8_t rfAnalogDetConfig(uint8_t afeld, uint8_t chNo, uint8_t rfdeten, uint8_t rfdetMode, uint8_t
rfDetNumHitsMode, uint32_t rfdetnumhits, uint8_t rfdetThreshold, uint8_t rfdetstepsize)
// Analog RF Detector Configuration
  
```

关键参数说明：

rfdeten: 置 1 使能 RF Detector

rfDetMode: RF Detector 的模式，置 1 作为 very big step attack detector

rfDetNumHitsMode: 设置 *rfdetnumhits* 参数的模式，0- Absolute; 1- Relative

rfdetnumhits: Absolute 模式下，设置计数门限，支持范围 0-2³²，Relative 模式下，设置值 = floor(参数值 * bigStepAttkWinLen / 2³²)

rfdetThreshold: detector 的门限值，dBfs 为单位

rfdetstepsize: Attack 的 step size, dB 为单位

此外，还有一个不支持修改设置的 Reliability Detector（图 2 中红色编号 5），它具有固定的门限值，当该 Detector 被触发，DSA 会增加到最大衰减以保护芯片不被过大的输入功率损坏。

2.3 决策状态机

AGC 的决策状态机（图 2 中红色编号 6）根据各 Detector 来的信号和预设的 StepSize 等参数控制 DSA 和 LNA Bypass 的状态。如图 6 所示是典型的单频场景下（仅使能 small step detector）状态机的控制流程。

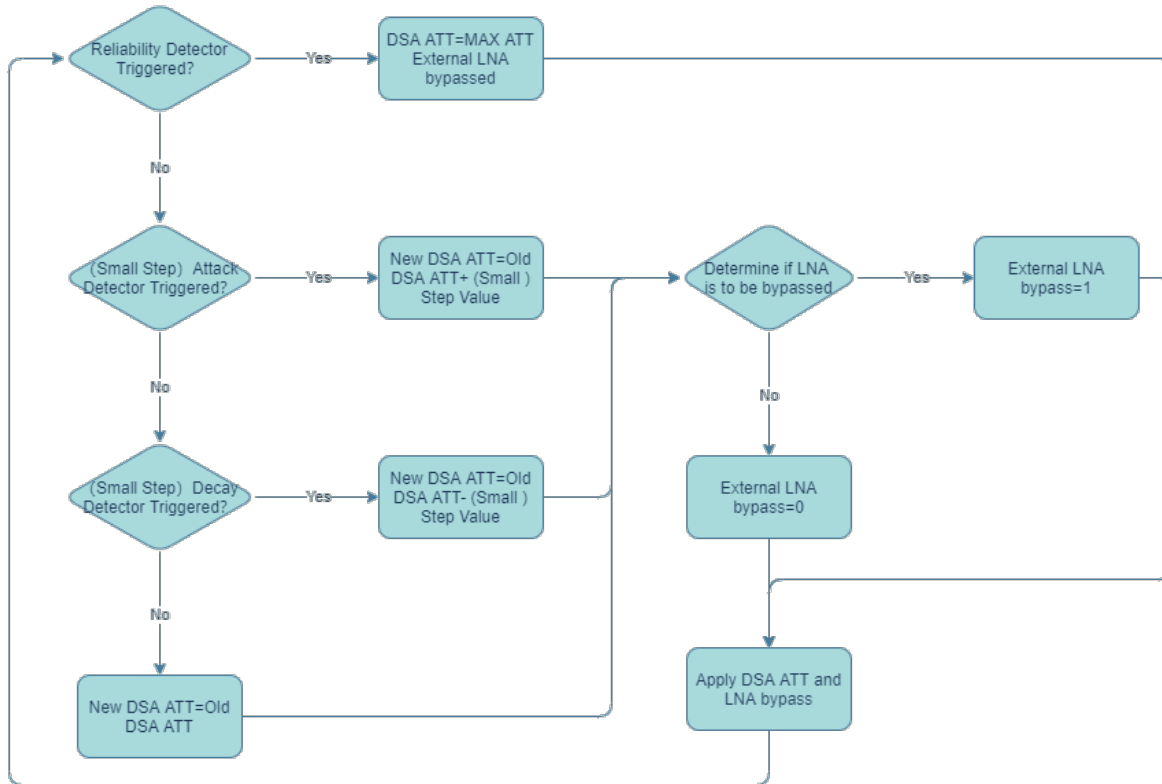


Figure 6. 单频场景 AGC 状态机控制流程图

在 TDD 场景下，Detector 观测窗长会出现比 TDD 开启周期更长的情况，状态机可以支持在 TDD off 的状态下，Freeze Detector 或者 Reset Detector，需要注意的是，只有 Decay Detector 可以被 Freeze，无论何种模式，Attack Detector 都会被 reset。此外，TDD 场景下，由于外部器件开关造成的过冲可能会误触发 AGC Attack，AGC 状态机支持从 TDD 开启时刻向后偏移的一定的 offset 开启 Detector 来规避误触发的功能。

使用 C 函数可以设置 AGC 的状态和 StepSize 等参数：

```
uint8_t agcStateControlConfig(uint8_t afeld, uint8_t chNo, uint16_t agcstate) // AGC State Control Macro
```

```
uint8_t internalAgcConfig(uint8_t afeld, uint8_t chNo, uint8_t tdd_freeze_agc, uint16_t blank_time_extcomp, uint8_t en_agcfreeze_pin, uint8_t extCompControlEn) // Internal AGC Configuration
```

```
uint8_t agcGainStepSizeConfig(uint8_t afeld, uint8_t chNo, uint8_t bigStepAttkStepSize, uint8_t smallStepAttkStepSize, uint8_t bigStepDecayStepSize, uint8_t smallStepDecayStepSize) // Internal AGC Gain-Step Configuration
```

关键参数说明：

Agcstate: 每个 bit 代表不同的 AGC 状态/模式选择, 0: Start Internal AGC with entire configuration redone; 1: Freeze the Internal AGC loop; 2: Unfreeze the Internal AGC loop (takes effect only if the loop is already in freeze); 3: Disable Internal AGC loop; 4: ALC Block enable; 5: ALC Block disable; 6: External AGC enable; 7: External AGC disable; 8: Restart the Internal AGC. (Step1: Disable Internal AGC, Step2:Enable Internal AGC); 9: Restart ALC(Step1: Disable ALC, Step2:Enable ALC); 10: Restart external AGC(Step1: Disable external AGC, Step2:Enable external AGC)。置位对应的 bit 选择该功能, 可以同时置位多个 bit, 但是部分 Bit 组合不被允许

tdd_freeze_agc: 置 1 使能该功能, 所有 Attack detector 在 TDD off 时隙被 freeze

extCompControlEn: 写 1 使能 LNA bypass 控制

bigStepAttkStepSize/smallStepAttkStepSize/bigStepDecayStepSize/smallStepDecayStepSize: 对应 detector 的 step size, 设置值=参数值*0.5, 单位为 dB

2.4 DSA 和 DGC

2.4.1 DSA 与 DSA 校准

在 AGC 模式下, RX DSA 完全受控于状态机, 如果需要手动修改 DSA 的值 (*dsaSetting*), 必须先 freeze AGC (*agcStateControlConfig* 函数) 或切换 RX DSA 的受控模式, 否则修改不会生效。需要注意的是, 寄存器的写入值和读取值是 DSA 的两倍 (寄存器每增加 1, DSA 大 0.5dB, 即 $DSA\ Value = 0.5 * Register\ Value$)

使用 C 函数可以切换 DSA 的受控模式和读写 DSA 的值:

```
uint8_t setRxDsaMode(uint8_t afeld, uint8_t topNo, uint8_t mode) // Set the RX DSA Mode
```

```
uint8_t setRxDsa(uint8_t afeld, uint8_t chNo, uint8_t dsaSetting) // Set the RX Analog DSA
```

关键参数说明:

topNo: 0 选择 A/B 通道, 1 选择 C/D 通道

mode: 写 2 为 internal AGC 模式, 写 3 为 SPI AGC 模式

dsaSetting: 设置 DSA 的值, 设置值=参数值*0.5, 单位为 dB

AFE79xx 的 RX DSA 在未校准情况下, 可以提供 0.5dB 的步进精度, 对于较大衰减情况下, 积累误差会比较大, 以至于可能大于 AGC 门限的回滞区间而导致乒乓效应, 因此我们建议对 RX DSA 的幅度和相位进行校准。DSA 校准分为两步, 第一步是生成校准数据包, 只需要在产线做一次, 取得的数据包存在板载 ROM 中; 第二步是 AFE 初始化阶段载入数据包, 该流程在每次 AFE 上电初始化阶段自动完成。进行 DSA 校准时, 可以逐个通道进行校准, 或者四个通道同时进行校准, 灌入 RX 端口的单音必须在工作频带内, 双频应用需要依次灌入两个信号, 输入单音频率偏移频带中心不超过+40%的接口速率, 且偏移量必须是 RX 接口速率/256 的整数倍, 功率大于-15dBFS, 另外要保证灌入信号的质量, 较大的相位噪声或者 spur 会导致校准精度较低或者校准失败。在某些特殊的场景, 我们还支持从 TX 输出信号灌入 RX 通道进行校准。

2.4.2 DGC

DGC 又称 ALC（图 2 中红色编号 7），内部 AGC 模式时，它必须被使能，而外部 AGC / MGC 模式时，根据具体应用场景，它可以被 disable。DGC 的工作原理是根据 DSA 的衰减值得信号在数字域进行补偿，但是对 DDC 输出的定点数直接进行补偿会导致数字溢出（16bit 二进制补码，可表达 $-2^{15} \sim 2^{15}$ 范围内的整数），因此需要使用某种方式去扩展原先数据的范围，最简单的方式是使用浮点数输出形式，若使用 2bit 阶码，可以扩大 12dB 的范围，可以看到，每 bit 阶码代表了 6dB，传递增益的精度不高，而且，I 和 Q 数据一般具有类似的幅度，因此 IQ 的阶码是一样的，因此我们可以考虑删去重复的部分，并通过人为的方式定义每一个 bit 的权重（Slicer Step），如图 7 所示，我们得到了 Coarse+Fine 模式的雏形。

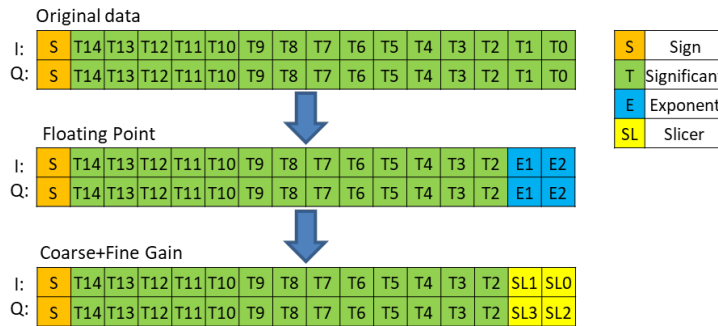


Figure 7. Floating Point 和 Coarse+Fine 模式对比图

如图 8 所示，Coarse+Fine 模式下，DGC 将 DSA 的值分成两部分，一部分称为 Coarse Gain，其 Index 取值等于 $DSA / \text{Slicer Step}$ （商取整），使用 GPIO 或者插入数据流的 LSB 传递给 ASIC/FPGA，若 Slicer Step (stepSize) 为 3dB，4bit 的 Slicer 可以表示 48dB 的动态范围；剩下的部分被称为 Fine Gain，其值等于 $DSA \% \text{Slicer Step}$ （取余），直接在定点数格式的数据上进行补偿，为了避免数字溢出，在进行 Fine Gain 补偿之前，需要对数据进行大小为 Slicer Step-1 的数字域回退。因此接收端通过 $\text{Slicer Index} * \text{Slicer Step}$ 得到 Coarse Gain，结合 Fine Gain 得到实际的信号幅度。

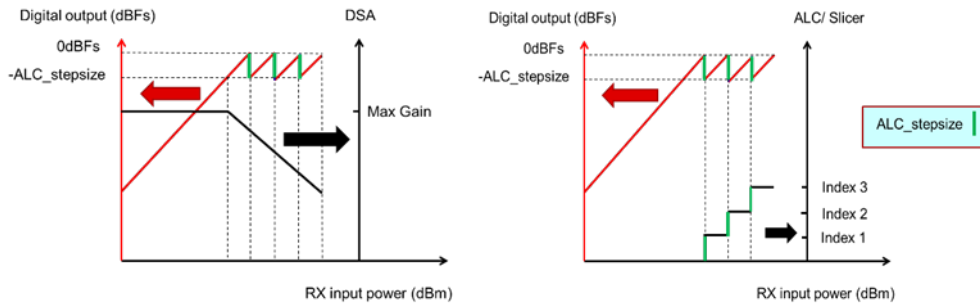


Figure 8. DSA 和 DGC 的关系简图

因为插入 Slicer bits 会占据数据的 LSB，影响原始数据。一个常见的错误是接收端解码时未注意 AFE 侧使能 DGC 与否，如图 9 所示，上方是未启用 DGC 的数据帧，下方所示是插入 Slicer Bits 之后的数据帧，如果如右上方那样对原始 16bit 数据做截断，或者如左下方那样对整个数据帧（包含 Slicer bits）进行解码，都会造成在 DC 处观察到一个 -85dBfs 左右的 spur。

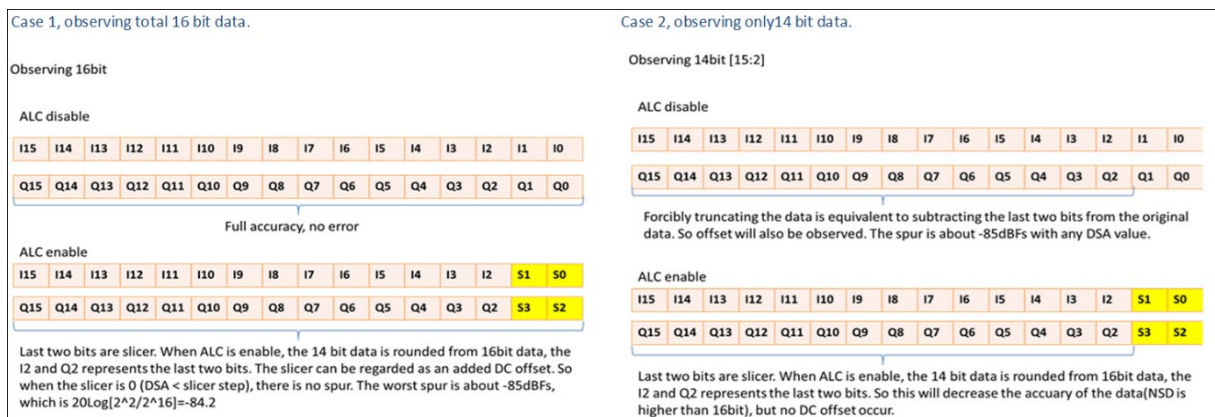


Figure 9. 四种可能的解码方式示意图

可以使用 C 函数设置 DGC 的状态和相关参数：

```
uint8_t alcConfig(uint8_t afeld, uint8_t chNo, uint8_t alcMode, uint8_t totalGainRange, uint8_t minAttnAlc, uint8_t useMinAttnAgc) // ALC Configuration
```

```
uint8_t coarseFineConfig(uint8_t afeld, uint8_t chNo, uint8_t stepSize, uint8_t nBitIndex, uint8_t indexInvert, uint8_t indexSwapIQ, uint8_t sigBackOff, uint8_t gainChangeIndEn) // Coarse-Fine Mode Configuration
```

关键参数说明：

alcMode: 写 3 使能 coarse fine 模式

totalGainRange: ALC 补偿的最大 DSA，一般设置成 maxDsaAttn

minAttnAlc: ALC 补偿的最小 DSA，当 DSA 值小于该值时，ALC 不对增益进行补偿，一般设置为 0

useMinAttnAgc: 写 1 则 ALC 最小补偿电平按 minAttnAlc 的设置，写 0 则 ALC 最小补偿电平跟随 minDsaAttn

stepSize: slicer step 的大小，0x00 → 0 dB；0x01 → 1 dB；0x02 → 2 dB；0x03 → 3 dB；0x04 → 4 dB；0x05 → 5 dB；0x06 → 6 dB；0x08 → 8 dB

nBitIndex: 设置 slicer 的 bit 数，支持 0，2，3，4

indexInvert: 为 0 输出原始值，为 1 输出 15-slicer index

indexSwapIQ: 0: LSB on I, MSB on Q；1: MSB on I, LSB on Q

sigBackOff: 仅在 coarse fine gain 模式使用，当 ALC 的补偿范围小于 maxDsaAttn 时，设置值 = totalGainRange - coarse fine range

gainChangeIndEn: 仅当 slicer bit 为 3 时有效，DSA 值的改变时 slicer 的 MSB 置 1

3 AGC + LNA Bypass

3.1 扩大 AGC 的范围

AFE79xx 的 RX 通道 DSA 能够提供 25dB 的衰减，精度优于 0.5dB（校准后精度优于 0.1dB）。可以取 1dB 为 Small Step Attack/Decay 的 DSA 步进，即 AFE 可以提供 25 个档位。某些情况下，25 个档位是不够用的，系统设计中需要想办法扩大系统的衰减档位，主要有外加 DVGA 和 LNA BYPASS 两种方式。

如图 10 所示，外加的 DVGA（可变增益放大器）受 MCU/ASIC/FPGA 的控制改变增益，当需要的衰减档位小于 25dB（或其他设定值）的时候，AFE 内部 DSA 提供衰减量，超过这一范围时由外置 DVGA 提供额外的衰减，相当于使 MGC 环路中嵌套了一个 AGC 环路，其中 AGC 环路只提供一定范围内的动态范围，因此系统实现相对比较复杂，且 DVGA 的成本也更高。

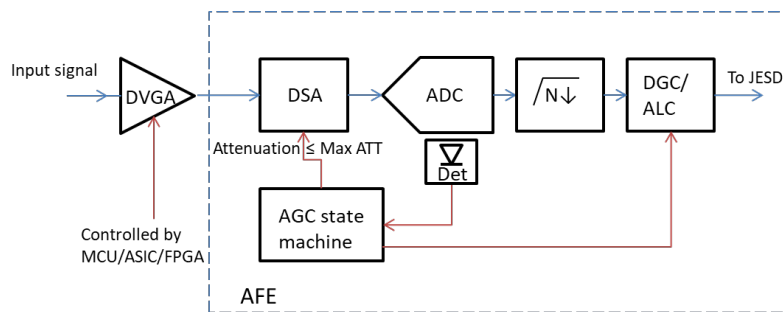


Figure 10. 外接 DVGA 扩大 AGC 范围的系统框图

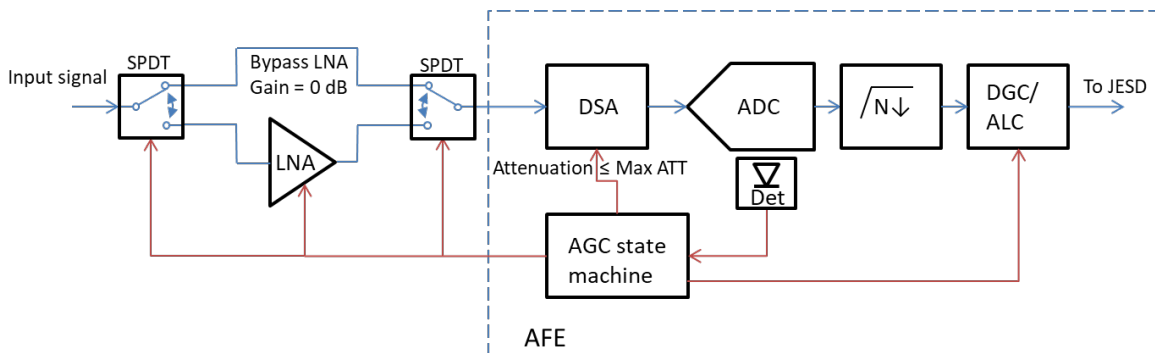


Figure 11. 使用 LNA bypass 扩大 AGC 范围的系统框图

另一种方式是使用 LNA bypass 功能，如图 11 所示，只需要在外部使用 LNA 和开关（或带 bypass 功能的 LNA），开关受 LNA Bypass 的控制，当 DSA 值达到设定的门限时，AFE 使能 LNA bypass 即可提供额外的动态范围。

3.2 LNA Bypass 参数设置

AFE79xx 提供两种 LNA bypass 的模式，第一种模式仅需要配置较少的参数，适用于外部 LNA 增益大于或等于系统的总衰减量一半的场景。下面的例子假设系统的总衰减是 30dB 且 LNA 增益恰好等于总衰减的一半，因此需要设定一个门限 Max ATT (*maxDsaAttn*) 为 15dB，应当注意，这里的 Max ATT 指的是逻辑意义上的 DSA 最大衰减值，小于等于 79xx RX 支持的最大衰减 25dB，在 LNA bypass 未使能情况下，DSA 逐渐增大并超过门限 (*maxDsaAttn*) 时，LNA bypass 被使能，外部 LNA 被 Bypass，提供 15dB 的衰减，同时 DSA 释放 14dB 衰减，LNA Bypass 之后的总衰减为 16dB (15dB LNA +1dB DSA)，当信号进一步增大，DSA 进一步增大以提供更大的衰减量；同理，在 LNA bypass 使能的情况下，当 DSA 不断减小并回到 0dB 时，LNA 重新使能释放 15dB 衰减，DSA 增加到 15dB。图 12 用图形化的方式画出了这一过程，每一行表示不同的总衰减量情况，蓝色方块指代 DSA 的衰减量，每格为 0.5dB，绿色方块代表 LNA bypass 提供的衰减量，固定为 15dB，红线为设置的 Max ATT 值，可以看到，DSA 提供的衰减量不超过红线所在的 15dB (Max ATT) 位置。

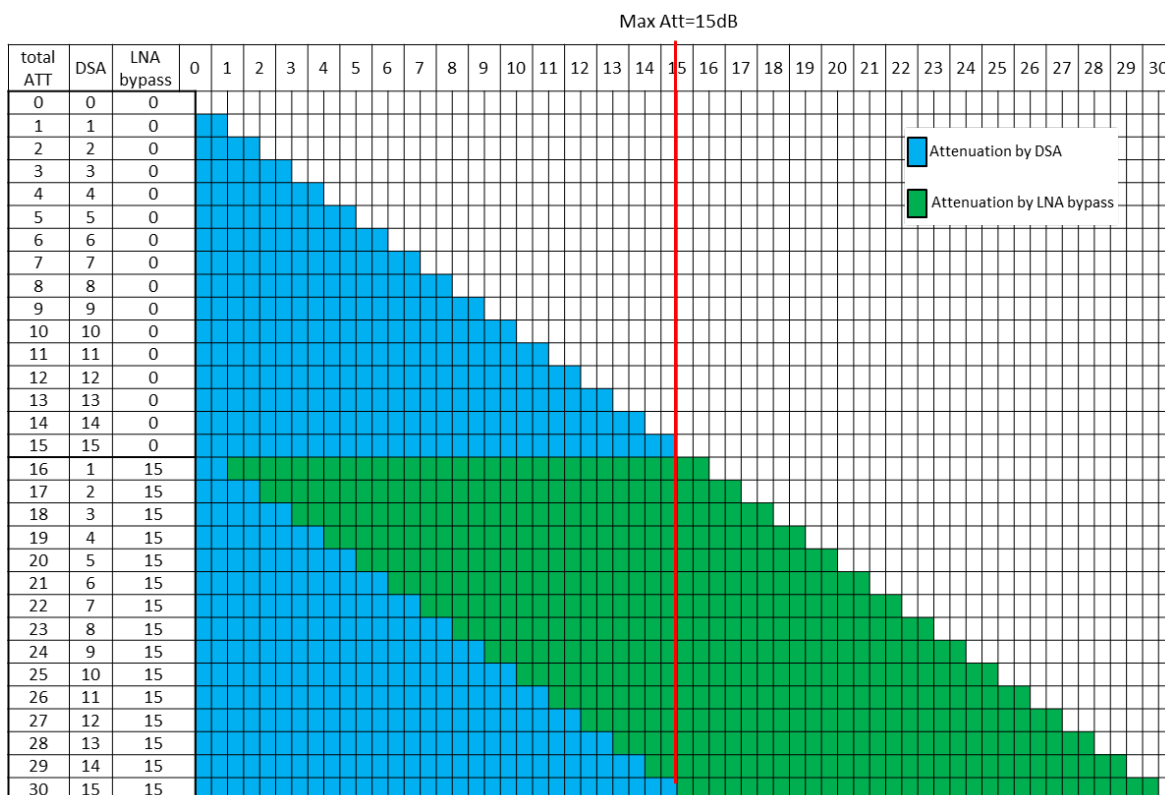


Figure 12. 达到 Max Atten 使能 LNA bypass 示意图

第二种模式可以支持 DSA 未达到 Max ATT 的情况下提前 Bypass LNA，相较于第一种模式，需要额外配置 Gain Margin (*lnaGainMargin*) 和 Early DSA (*earlyDsaBypass*) 参数，它们之间的关系如下：

Early DSA must 0.5dB bigger than LNA Gain

Gain Margin >= MAX ATT – Early DSA (typically Gain Margin =MAX ATT. – Early DSA)

re-enable DSA = Max ATT - LNA Gain - Gain Margin

这种模式适用于 LNA 的增益小于系统总衰减一半的情况。下面的例子中，假设 LNA 增益为 8dB，系统需要的总衰减为 23dB，当 DSA 达到 8dB 时触发 LNA bypass。如图 13 所示，计算得到，Early DSA 设置为 8.5dB，Gain Margin 设置为 6.5dB，则 re-enable DSA 为 0.5dB。LNA bypass 未使能情况下，DSA 超过 8.5dB 时，触发 LNA bypass，LNA 使能情况下，DSA 小于 0.5dB，LNA bypass 释放。可以看到释放 LNA bypass 的前提是 DSA < re-enable DSA，而 DSA 不可以小于 0dB，因此必须要求 Early DSA 必须比 LNA Gain 大 0.5dB。

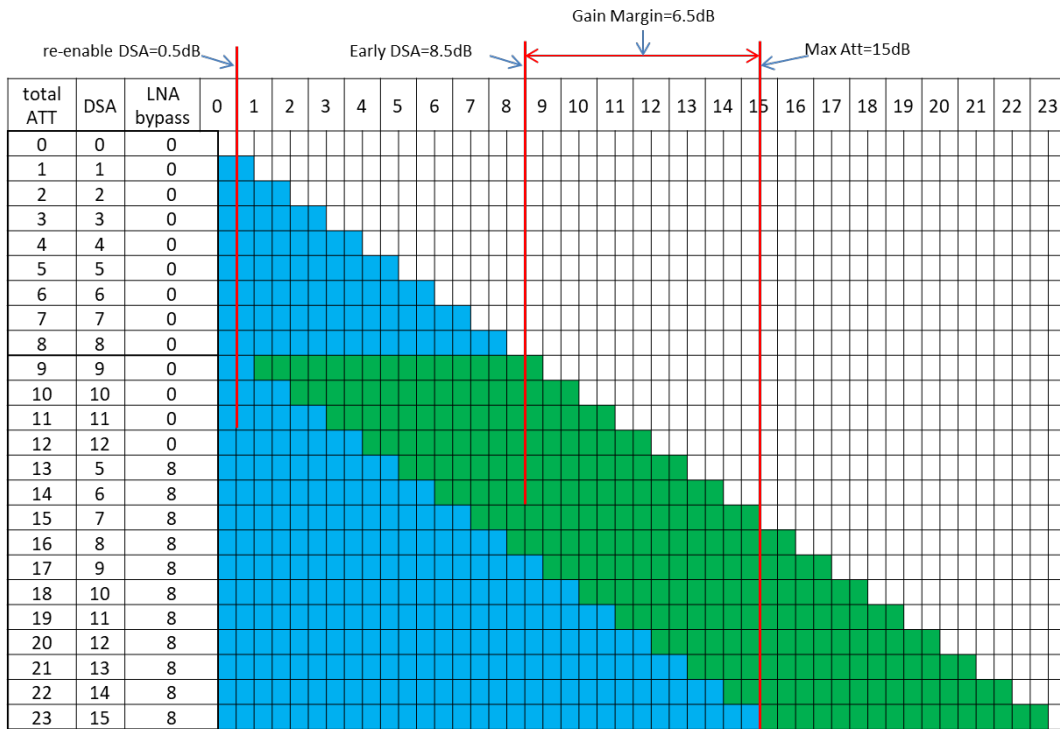


Figure 13. 达到 Early DSA 使能 LNA bypass 示意图

使用 C 函数可以设置 LNA Bypass 的参数，DSA 相关寄存器值需等于 2*DSA Value:

```
uint8_t minMaxDsaAttnConfig(uint8_t afeld, uint8_t chNo, uint8_t minDsaAttn, uint8_t maxDsaAttn) // Internal AGC Min-Max Attenuation Configuration
```

```
uint8_t extLnaConfig(uint8_t afeld, uint8_t chNo, uint8_t singleDualBandMode, uint8_t lnaGainMargin, uint8_t enBandDet, uint8_t tapOffPoint) // External LNA Configuration
```

```
uint8_t earlyDsaConfig(uint8_t afeld, uint8_t chNo, uint8_t earlyDsaBypassEn, uint8_t earlyDsaBypass) // Early Dsa Configuration
```

关键参数说明:

minDsaAttn: 最小 DSA 值，一般设置为 0，设置值=传参*0.5dB

maxDsaAttn: 最大 DSA 值，按需求设置，设置值=传参*0.5dB

lnaGainMargin: 设置 Gain Margin 的值, 设置值=传参*0.5dB

earlyDsaBypassEn: 写 1 使能 Early DSA 功能, DSA 超过 earlyDsaBypass bypass 外部 LNA; 写 0 关闭 Early DSA 功能, DSA 超过 maxDsaAttn bypass 外部 LNA

earlyDsaBypass: 设置 Early DSA 的值, 设置值=传参*0.5dB

4 典型应用场景举例

4.1 场景需求示例

下面以一个 F_ADC=2949.12M, 仅使能 Small Step Attack Detector 和 Small Step Decay Detector, 外部 LNA 为 8dB, 使能 LNA bypass 且 LNA bypass 需要在 DSA=12dB 的位置触发的单频 TDD 应用场景为例举例 C 函数的调用及参数设置。图 14 所示, 我们需要设置 Early DSA 为 12dB, Gain Margin 设置为 3dB, 则 re-enable DSA 为 4dB。所有参数可以在 Bringup 脚本中被配置, 也可以使用 C 函数在 debug 阶段快速修改进行验证实验。建议在使用 C 函数进行调试完毕之后再参数固定到 Bringup 脚本里面。

Type	Request	Configure	Comments
Mode	agc_Mode	Internal AGC	启用内部 AGC
	tdd_freeze_agc	Freeze	TDD off 期间 freeze AGC 状态机, DSA 值保持到下一个时隙
Attn	minDsaAttn	0dB	最小 DSA 值
	maxDsaAttn	15dB	最大 DSA 值
	totalGainRange	23dB	DSA 15dB, LNA Gain = 8dB
ALC	alcMode	coarsefine	Coarse Fine Gain 模式
	alc_coarse_step	3dB	Slicer Step 为 3dB
	alc_num_bits_coarse_index	4bit	4bit Slicer (I 和 Q 各 2bit)
EN	alcEn	Enable	使能 DGC
	lnaEn	Enable	使能 LNA Bypass
	agc_small_step_decay_en	Enable	仅使能 Small Step Attack/Decay Detector
	agc_small_step_attack_en	Enable	
PARAM	agc_small_step_attack_step_size	1dB	Attack 步进 1dB
	agc_small_step_decay_step_size	1dB	Decay 步进 1dB
	agc_small_step_attack_sig_th	-3dBFS	Attack 门限-3dBFS
	agc_small_step_decay_sig_th	-6dBFS	Decay 门限-6dBFS
	agc_small_step_attack_win_len	0.1us	Attack 检测窗长 0.1us
	agc_small_step_decay_win_len	10ms	Decay 检测窗长 10ms
	agc_small_step_attack_num_hits	10ms	10ms 内的数据都低于门限
	agc_small_step_decay_num_hits	0.1us*10%	0.1us 内 10%的数据高于门限

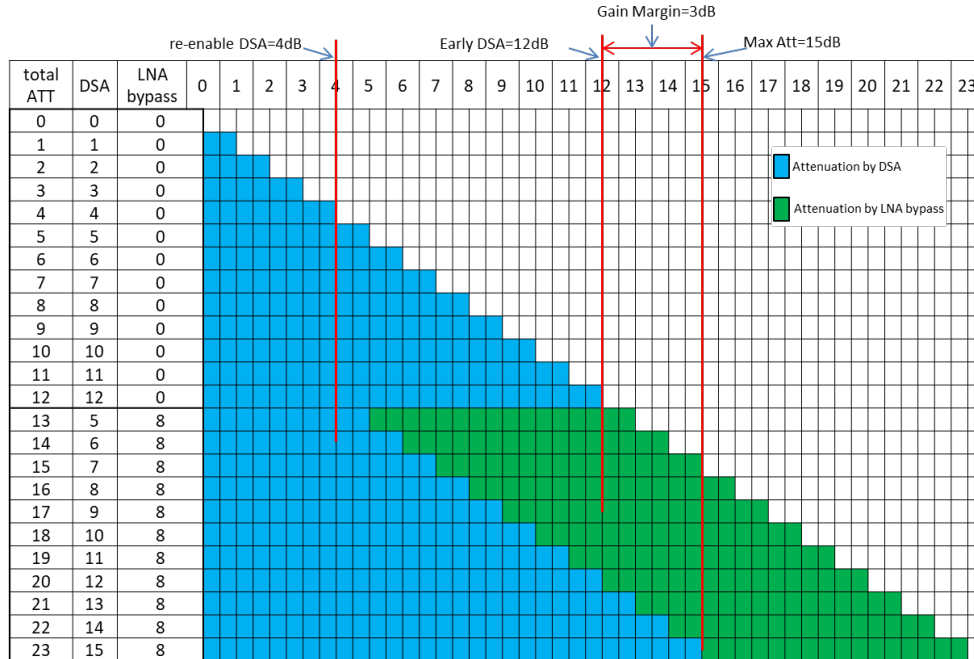


Figure 14. 典型应用场景的衰减示意图

4.2 C 函数调用示例

为了方便阅读，函数调用的关键参数位置写成 parameter name = value 的形式，实际使用中，需要省略 “parameter name = ”。

1. 设置使能的 Detector 及其对应的参数

agcDigDetConfig(afeld = 0, chNo = 0xf, 0, 1, 0, 1, 0, 0,0, smallStepAttkThresh = 12, 0, smallStepDecThresh = 24, 0, 0)//仅使能 Small Step Attack Detector 和 Big Step Attack Detector 并分别设置门限为 3dB 和 6dB

agcDigDetTimeConstantConfig(0, 0xf, 0, miscStepAttkWinLen = 10, decayWinLen = 1000000)//设置 Attack 观测窗长为 100ns，Decay 观测窗长为 10ms

agcDigDetRelativeNumCrossingConfig(0, 0xf, 0, smallStepAttkNumHits = 0x199A, 0, smallStepDecNumHits = 0xffff)//使用 relative 方式，设置 attack number of hits 为 100ns 的 10%，decay number of hits 为全部的 100%的 10ms

agcGainStepSizeConfig(0, 0xf, 0, smallStepAttkStepSize = 2, 0, smallStepDecayStepSize = 2)//设置 attack 和 decay 的步进都为 1dB

2. 设置 DGC 有关参数

alcConfig(0, 0xf, alcMode = 3, totalGainRange = 25, 0, useMinAttnAgc = 1)//设置 DGC 模式为 Coarse+Fine，最大 DSA 为 25dB，最小补偿值跟随 AGC 的设置


```
coarseFineConfig(0, 0xf, stepSize = 3, nBitIndex = 3, 0, 0, 0, 0)//设置 Slicer Step 为 3dB, 3 bit Slicer
```

3. 设置 LNA Bypass 有关参数

```
minMaxDsaAttnConfig(0, 0xf, minDsaAttn = 0, maxDsaAttn = 30)//设置 DSA 最大为 15dB
```

```
extLnaConfig(0, 0xf, singleDualBandMode = 0, lnaGainMargin = 5, enBandDet = 0, tapOffPoint = 0)//设置 LNA 为单频模式, 设置 Gain Margin 为 2.5dB
```

```
extLnaGainConfig(0, 0xf, lnaGainB0 = 256, lnaPhaseB0 = 0, 0, 0)//设置 external LNA Gain 为 8dB
```

```
earlyDsaConfig(0, 0xf, earlyDsaBypassEn = 1, earlyDsaBypass = 25)//使能 Early DSA 并设置 Early DSA 为 12.5dB
```

4. 设置 TDD 模式下的参数

```
internalAgcConfig(0, 0xf, tdd_freeze_agc = 1, 0, en_agcfreeze_pin = 0, extCompControlEn = 1)//TDD off 状态下 Decay detector 被 freeze
```

5. 设置 DSA 的受控模式 (若静态配置已经配置为 AGC 模式则可以跳过)

```
setRxDsaMode(0, topNo = 0, mode = 2)//RXA&B 设置为 internal AGC 模式
```

```
setRxDsaMode(0, topNo = 1, mode = 2)//RXC&D 设置为 internal AGC 模式
```

6. 设置 AGC 模式为内部 AGC 并使能 AGC 和 DGC

```
agcStateControlConfig(0, 0xf, agcstate = 0x11)//启动 AGC, 必须最后调用该函数
```

5 参考文献

1. Datasheet “AFE79xx Quad-Channel RF Transceiver With Feedback Path”
2. Application Report “AFE79xx Configuration Guide”

重要声明和免责声明

TI 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安全、安保或其他要求。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 (<https://www.ti.com.cn/zh-cn/legal/termsofsale.html>) 或 [ti.com.cn](https://www.ti.com.cn) 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。

邮寄地址：上海市浦东新区世纪大道 1568 号中建大厦 32 楼，邮政编码：200122

Copyright © 2021 德州仪器半导体技术（上海）有限公司