Application Report
ZHCA538 – July 2013

在 KeyStone 器件上实现高效的 LTE 上行基带前端处理

范伟，尹志新

Multicore DSP / FAE

摘要

LTE 上行基带前端处理包括为 PUSCH、PUCCH、SRS 信道执行的 FFT，以及为 PRACH 信道执行的时频转换。这些操作处理的是原始时域天线数据，数据量大，对计算资源和存储资源的需求都较高。文本针对 TI 的 KeyStone 器件给出了完整的 LTE 上行基带前端处理设计，目标是尽可能减少核的干预，消耗尽可能少的资源。本文还详细列出了该设计对各种硬件资源的需求，以及 c66x 核上任务的实测负载。

文档历史

版本	日期	作者	注释
1.0	2013 年 6 月 5 日	范伟，尹志新	初始发布版本。
1.1	2013 年 6 月 9 日	范伟，尹志新	修改错误，补充内容。
1.2	2013 年 6 月 26 日	范伟，尹志新	格式、图片调整，规范化器件型号，增加前端 FFT 的 TTI 配置函数的负载测量，调整 TDD 情况下的前端 FFT 入队 EDMA 的图示及相应描述。

目录

1 引言

LTE（Long Term Evolution）是由 3GPP 组织制定的 3G 演进标准，在物理层采用 OFDM 和 MIMO 技术。LTE 分为 FDD 和 TDD 两种双工模式。目前，LTE-FDD 在 20MHz 频谱带宽下的实际速率大约能达到下行 100Mbps、上行 50Mbps。LTE-TDD（国内通常称为 TD-LTE）的实际速率会随上、下行子帧的配比关系而变化。

[1][2][3][4]是主要的几个 LTE 物理层协议文本。[1]描述了上、下行发射机从星座点调制到基带信号上变频之间的处理步骤，通常称为符号级处理。[2]描述了星座点调制之前的处理步骤，通常称为比特级处理。[3]描述了各种物理层过程。[4]描述了各种物理层测量。

LTE 的上行物理信道信道包括用来传输数据和物理层随路控制信令的 PUSCH，专门用来传输物理层控制信令的 PUCCH，用于随机接入的 PRACH，以及用于上行信道探测的 SRS。下行物理信道包括用来传输数据的 PDSCH，用来传输各种物理层控制信令的控制信道 PCFICH、PHICH 和 PDCCH。

本文描述的是 LTE 上行基带前端处理。如图 1 所示，LTE 上行基带前端处理包括从天线接口接收时域数据，以及随后的时频变换。LTE 中所有上行信道接收机的第一步都是把信号从时域变到频域，这是上行基带前端处理最主要的任务。PRACH 采用和其它上行物理信道不同的时频结构，因此，前端时频变换需要做两次，一次用于 PRACH，一次用于其它上行物理信道。本文把前者称为“PRACH 前端时频转换”，把后者称为“上行前端 FFT”。

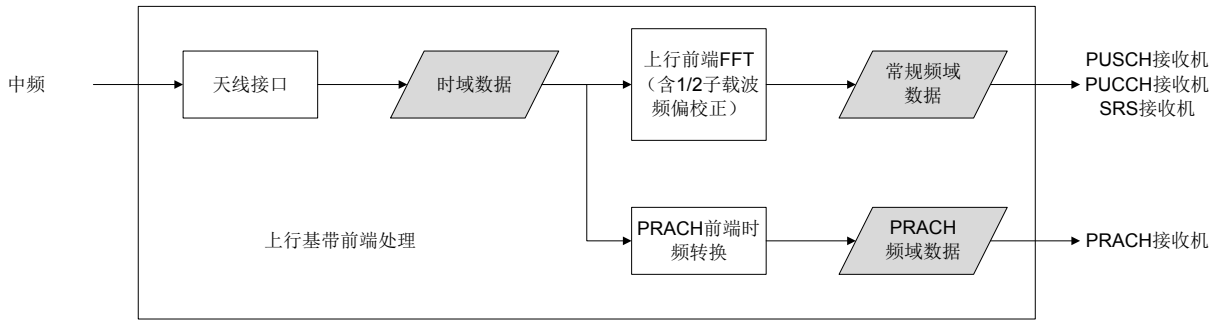


图 1: LTE 上行基带前端处理

TI 推出了一系列用于 LTE 基站基带处理的 SoC（System On Chip）。这些 SoC 基于 TI 的 KeyStone 架构，该架构目前已演进了两代——KeyStone I 和 KeyStone II。KeyStone I 家族基于 40nm 工艺，包括如下基带 SoC 器件型号：

- TCI6616，详细资料参见[5]
- TCI6618，详细资料参见[6]
- TCI6614 和 TCI6612，详细资料参见[7]和[8]
- TMS320C6670，详细资料参见[9]

KeyStone II 家族基于 28nm 工艺，包括如下基带 SoC 器件型号：

- TCI6636K2H，详细资料参见[10]

- TCI6634K2K，详细资料参见[11]
- TCI6638K2K，详细资料参见[12]
- TCI6630K2L，详细资料参见[13]

所有这些器件都具有多模能力，支持 GSM/EDGE、WCDMA、TD-SCDMA、WiMAX、LTE 的单模实现或混模实现。所有这些器件使用的 DSP 核都是 c66x，但个数不同。TCI6614 和 TCI6612 带一颗 ARM Cortex A8，TCI6636K2H 和 TCI6638K2K 带 4 颗 ARM Cortex A15，TCI6630K2L 带 2 颗 A15，它们除支持物理层以外，还支持高层（层 2，层 3）和传输处理。这些器件也可用于基于 OFDM 的无线回传（wireless backhaul），如 LTE relay 站。

本文介绍如何在上述 KeyStone 器件上实现高效的 LTE 上行基带前端处理。上述 KeyStone 器件中，只有 TCI6630K2L 包含中频处理模块，称为 DFE 模块，该模块通过 IQNet 模块和系统总线相连。图 1 中的天线接口对 TCI6630K2L 来说指的是 IQNet，对其它器件来说指的是 AIF2。IQNet 和 AIF2 都使用 PktDMA 作为对内接口，行为方式类似。本文用 AIF 统一指代 IQNet 和 AIF2，描述的前端处理方法同时适用于两者。关于 KeyStone I 和 II 的 AIF2，请参考[14]和[15]。关于 PktDMA，请参考[16]。天线接口一旦配置完毕，在实时运行过程中不需要软件干预，不产生实时负载。本文重点描述时频转换。对天线接口，仅描述其与时频转换交互的部分。

基带前端工作在基带时域采样率上，对 20MHz 载波通常为 30.72Msps，而系统中通常存在多个这样的数据流，对应多天线和/或多载波。对这样高速的数据流做处理，效率至关重要，应尽量减少对处理资源、存储资源、总线资源的占用。上行基带前端的处理效率是基带整体效率的重要组成部分。本文给出了在 KeyStone 器件上用 FFTC 硬件加速器完成尽可能多的时频转换，并将天线接口和两类时频转换用 EDMA 进行直连的方法。该方法使相关的软件负载降至最低，并且尽可能地降低对内存和总线的占用。本文还给出了由 c66x 和 FFTC 共同完成 PRACH 时频转换的方法，并给出了实测负载。该方法减少了对 FFTC、内存、总线的占用，但增加了 c66x 负载。用户可根据自身系统的资源消耗情况选择不同的 PRACH 时频转换方法。关于 EDMA，请参考[17]。关于 FFTC，请参考[18]。

本文中的“符号”默认指的是 OFDM 符号。

2 上行前端 FFT

图 2 以单片 TCI6634K2K 或 TCI6638K2K 实现 2 个 8 天线 TD-LTE 载扇为例，描述了上行前端 FFT 的设计，包括对 Tx/Rx 通道、flow、Q（本文用 Q 表示硬件队列）、Tx 和 Rx 描述符、Acc（本文用 Acc 表示 Accumulator）等物理资源的使用，以及描述符的传递路径。相关的原理和更多的细节参见以下各节。

虽然这里以单片 TCI6634K2K/TCI6638K2K 实现 8 天线 TDD 双载扇为例，本文描述的设计实际上适用于在所有 KeyStone 器件上实现各种部署场景。注意，对于 TCI6634K2K/TCI6638K2K，当不要求所有载扇同时达到最高规格（比如在所有 RB 上执行带信道估计时域内插的 MU-MIMO IRC 均衡，下行每载扇瞬时数据流量 300Mbps，每载扇 400 个激活用户等）且经过充分优化时，单片可支持 8 天线 TDD 三载扇。

该设计以一个载扇为基本设计单位。多载扇时，每个载扇使用相同的设计，但可以配置不同的硬件资源。考虑到一个 FFTC 可以在一个符号周期内完成 8 次带 1/2 子载波频偏校正的 2048 点 FFT，为简单起见，要求一个载扇的上行前端 FFT 处理由一个 FFTC 完成。

图 2 中的例子采用的物理层核间分工策略是：8 个 c66x 核分为 2 组，核 0~3 是一组，核 4~7 是另一组，每组处理一个载扇。对第一组：核 0 和核 1 处理 PUSCH，其中信道估计按天线分工，均衡按时隙分工；核 2 处理 PUCCH 和 PRACH；核 3 处理下行和 SRS，包括上行子帧最后一个符号上的 SRS，以及 UpPTS 内的 SRS。第二组内的核间分工与第一组相同。

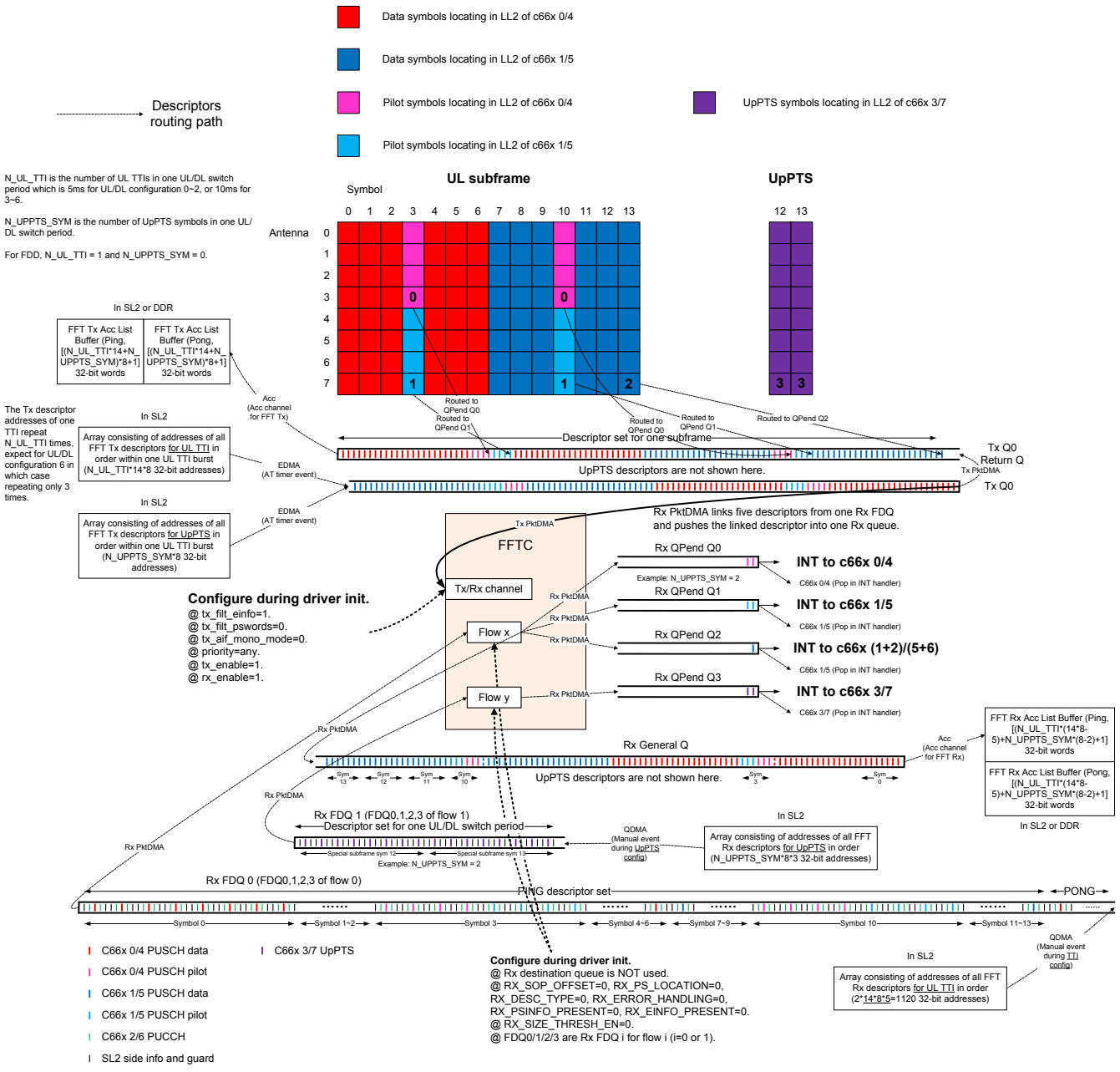


图 2: 上行前端 FFT 设计（例子：TD-LTE 8 天线）

2.1 FFTC Tx 侧设计

AIF 每收齐一根天线上的一个完整符号，就把一个包 push 到一个 RxQ 中。本设计假设一个载扇的N_A根接收天线与基带之间的延时是相同的，则对一个符号，N_A根天线对应的N_A个包几乎同时被 AIF 的 Rx PktDMA push 到 RxQ 中。为避免对 Rx 描述符的回收操作，可把 AIF 的 RxQ 和 Rx FDQ 配置成同一个 Q。除了 AT timer，其它关于 AIF 的部分没有在图 2 中画出。

AIF 和 FFTC 之间的直连不是通过 AIF 的 Rx PktDMA 把接收包直接 push 到 FFTC 的 TxQ 来实现的。实际上，AIF 可以通过内部的 AT timer 产生符号级事件，在一个符号的N_A个 Rx 包被 push 到 RxQ 之后，立即产生一个系统事件。该事件被用于触发一个 EDMA 操作，该操作把N_A个包 push 到 FFTC 的 TxQ 中。称该 EDMA 为 FFT 入队 EDMA。

对 TD-LTE，AIF 可以做到只接收上行子帧和 UpPTS 内的符号数据，但无法屏蔽其它符号对应的 AT timer 事件。FFT 入队 EDMA 接收所有的 AT timer 事件，但仅为上行符号对应的事件执行入队操作。

为了给应用提供尽可能多的灵活性，为一个子帧内的每个【符号，天线】分配一个 FFTC Tx 描述符，并使用描述符的 PS data 承载 FFTC 本地配置（Local Configuration）。记一个子帧的符号数为N_{sym}，则需要分配$N_{sym}N_A$个 Tx 描述符，每个描述符的大小是 64B。这样，应用可以在初始化时为每个【符号，天线】配置任意的参数，如 destination Q、output scaling 等。

用 Acc 对 Tx return Q 中的描述符进行自动回收，但不需要产生 Acc 中断或 EDMA 事件。为了正常工作，Acc 要求在一个乒或乓 List Buffer Page 满了之后，向 Interrupt N Count 寄存器写 1，作为应用对 Acc 的响应。因为不需要产生 Acc 事件，响应时不需要写 EOI 寄存器。

结合以上几点，FFT 入队 EDMA 的设计如图 3 和图 4 所示。

图 3 以 Normal CP 为例给出了 FDD 时的设计。此时用到了 3 个 PaRAM set：PaRAM set 0 只用一次，为系统启动后的第一个上行子帧的符号执行 FFTC 入队操作；PaRAM set 2 以子帧为单位被周期性地使用，为第一个上行子帧之后的所有上行符号执行入队操作；PaRAM set 1 用于在每个上行子帧开始时，响应前一个上行子帧的 Acc 操作。注意，为了使 Acc 操作正常，这里实际上有一个隐性前提：Acc 总能以略短于一个符号的延时处理完被 FFTC 的 Tx PktDMA push 进 Tx return Q 的描述符。该条件在通常的 Acc 负载下都能被满足。对 Extended CP，只要把 PaRAM set 0 和 2 中的 CCNT 从 14 改为 12 即可。

FFT 入队 EDMA 在处理完入队操作后，通过 chain 机制产生 PRACH 触发事件，该事件将触发 PRACH 前端时频转换。

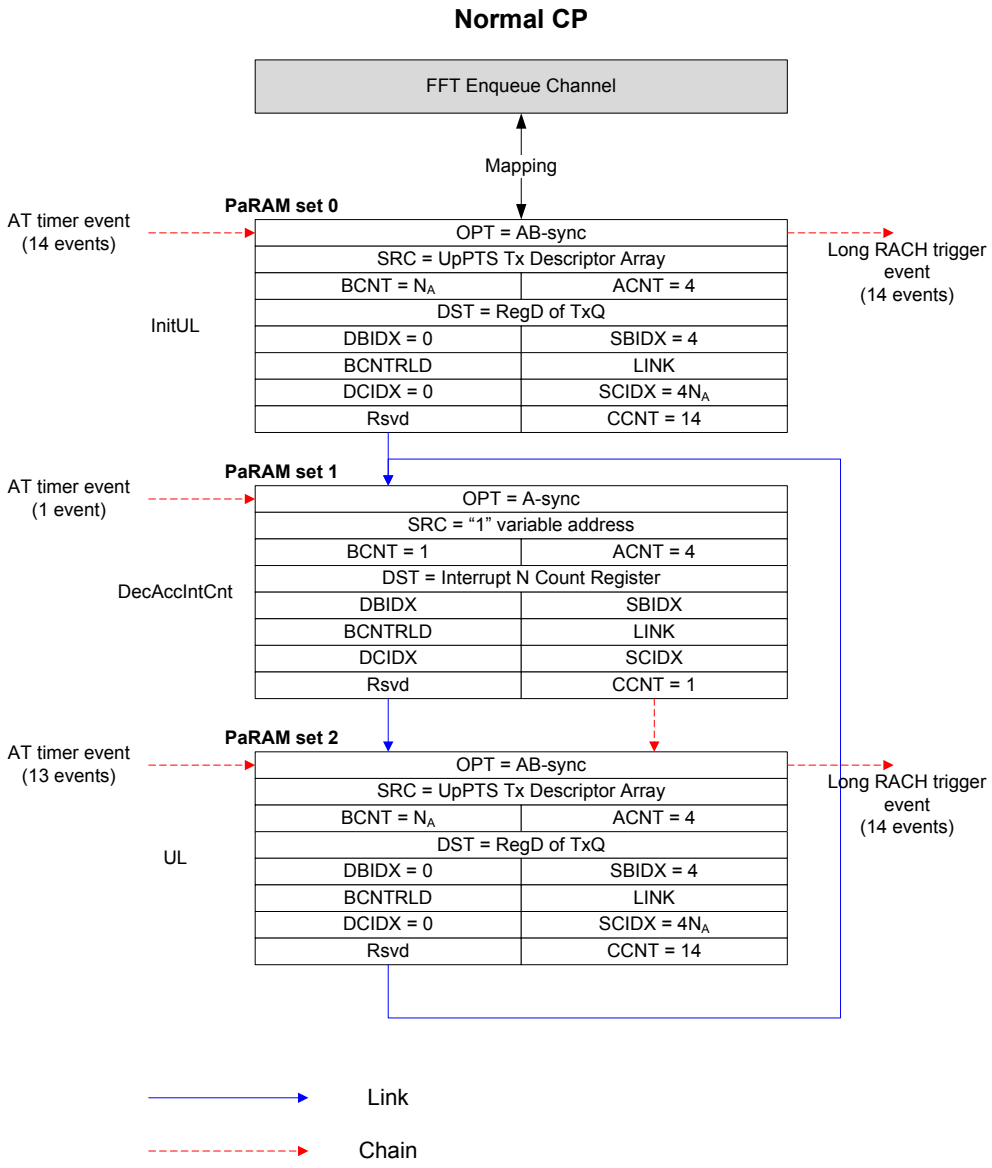


图 3: FFT 入队 EDMA 设计——FDD（例子：Normal CP）

图 4 以“UL/DL 配置 6、特殊子帧配置 5~8（UpPTS 包含 2 个符号）、Normal CP”为例给出了 TDD 时的 FFT 入队 EDMA 设计。在该系统配置下，用到了 8 个 PaRAM set：

- PaRAM set 0 只用一次，用于消耗掉系统启动后的第一个 UpPTS 之前的所有 AT timer 事件，不执行有实际意义的数据搬移操作。
- PaRAM set 2 和 5 以无线帧为单位被周期性地使用，分别为一个无线帧的前、后半帧中的所有上行子帧的符号执行 FFTC 入队操作。
- PaRAM set 1 和 4 分别为一个无线帧的前、后半帧中的 UpPTS 符号执行 FFTC 入队操作。
- PaRAM set 3 和 6 负责消耗掉非上行符号对应的 AT timer 事件。

- PaRAM set 7 响应前一个操作周期的 Acc 操作，这里的操作周期指的是从一个无线帧的第一个 UpPTS 开始的一个 10ms。

对 UL/DL 配置 6，需要为前、后半帧的上行子帧集合、UpPTS、其它符号集合的 AT timer 事件响应使用不同的 PaRAM set，这是因为两个半帧包含的上行子帧数是不一样的。对其它所有 UL/DL 配置，只需要 5 个 PaRAM set，对应的操作周期等于上下行切换周期（UL/DL 配置 0~2 是 5ms，3~5 是 10ms）。

不论是 FDD 还是 TDD，服务于上行子帧或 UpPTS 的 PaRAM set 使用 AB-sync，每次事件触发当前上行符号的N_A个 Tx 描述符的 FFTC 入队操作。CCNT 等于该上行子帧集合或 UpPTS 包含的符号数。

上行子帧集合和 UpPTS 对应的 PaRAM set 使用的源 buffer 的大小，以及 Acc 的 List Buffer Page 的大小，参见图 2。

对 TDD，如果当前载扇使用的是长 RACH（PRACH preamble 格式 0~3），上行子帧中的 FFT 入队 EDMA 在处理完入队操作后，还需通过 chain 机制产生长 RACH 触发事件，该事件将触发针对长 RACH 的前端时频转换；如果当前扇区使用的是短 RACH（PRACH preamble 格式 4），UpPTS 中的 FFT 入队 EDMA 通过 chain 机制产生短 RACH 触发事件，触发针对短 RACH 的前端时频转换。一个 TDD 小区不会同时使用长 RACH 和短 RACH，因此这两类 PRACH 触发事件不需要同时配置。注意，AIF 无法屏蔽非上行符号对应的 AT timer 事件，但是，经过 FFT 入队 EDMA 的过滤，PRACH 前端时频转换功能模块只会收到上行子帧对应的符号级事件。

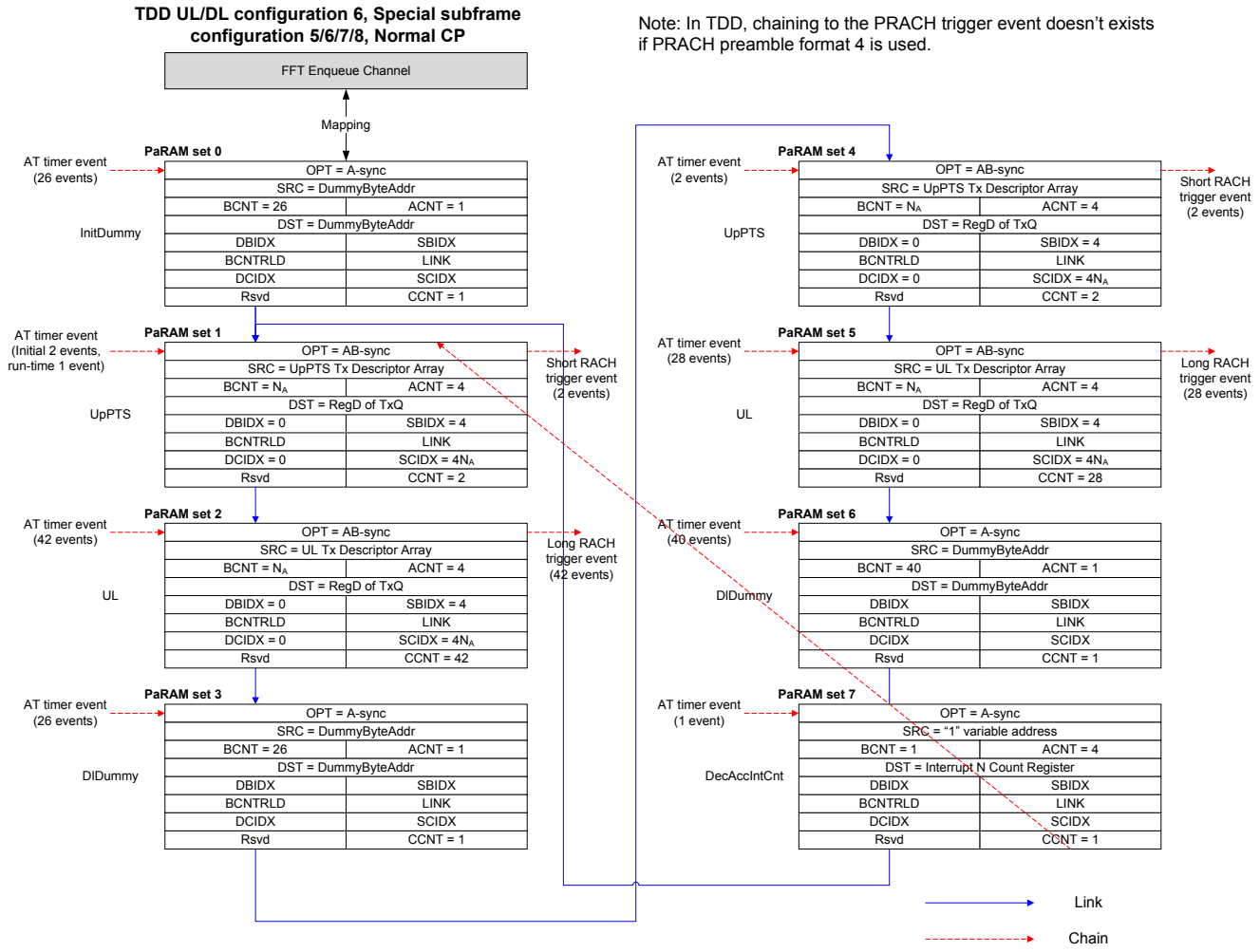


图 4: FFT 入队 EDMA 设计——TDD（例子：UL/DL 配置 6，特殊子帧配置 5~8，Normal CP）

图 5 以 TDD 8 天线、normal CP 为例描述了 Tx 描述符的配置。所有的配置域都是静态的，也就是说，只需要在小区建立或重配时配置，实时运行过程中无需修改。大部分配置域的取值对上行前端 FFT 应用来说是确定的，也有一些域的配置具有一定程度的灵活度，可根据算法、软件框架、资源分配等方面的考虑灵活配置，这些域在图 5 中用下划线标识，并说明如下。

- Return Q 和 flow 根据应用的硬件资源分配进行配置。
- 为了尽可能节省内存，这个例子为 AIF 接收数据使用符号级乒乓缓存。挂在 Tx 描述符上的 buffer 指针根据该描述符对应的【符号，天线】指向 16 个 buffer 中的一个。AIF 使用的是 monolithic 描述符，这种描述符的净荷直接跟在描述符头的后面。为了避免数据搬移，FFTC Tx 描述符的 buffer 指针直接指向 AIF Rx monolithic 描述符中净荷的起始位置。注意，即使不按符号乒乓设计 AIF 接收数据缓存，其它方面的设计无需变化，只影响 Tx 描述符中 buffer 指针的配置。
- 这个例子使用了 FFTC 的动态 scaling 功能，使 FFT 计算过程完全不发生饱和截位，同时每步蝶形运算的输入数据都有最大有效位宽，因而获得最高的计算精度。此时，每个【符号，天线】

的 FFT 输出数据块的定标各不相同，应用应使能 FFTC 的 side information 输出，该信息包括 FFTC 内部实际执行的各级 scaling 的总和，应用在随后的信道估计、测量、均衡处理中应考虑到各【符号，天线】上的 scaling 的不同，执行必要的定标对齐。应用也可以不使用动态 scaling，直接配置各级蝶形运算之前的移位量。无论使用动态还是静态 scaling，都需要配置静态的 output scaling factor 参数。该参数限制了 FFT 输出 I/Q 分量的最大幅度，应用可根据随后的信道估计和均衡所执行的计算，分别限制导频符号和数据符号的 FFT 输出最大幅度，使随后的计算在不出现饱和或溢出的情况下具有尽可能高的有效数据位宽。这个例子把 output scaling factor 配置成了 0x40，这使得输出数据的 I/Q 分量最大幅度是 Q15 定标下的 1/2，也就是说，输出复样本的最大幅度是 Q15 定标下的 1。不使用动态 scaling 时，通常不需要使能 side information 输出。

- 上行前端 FFT 通过将 Rx 描述符 push 进相应的 QPend RxQ 来通知 c66x 核，从而启动信道估计或均衡。在这个例子中，一个核处理天线 0~3 的信道估计，另一个核处理天线 4~7 的信道估计，因此：每个导频符号上的天线 3 的 Rx 包被 push 进一个 QPend Q，触发前一个核上的信道估计；每个导频符号上的天线 7 的 Rx 包被 push 进另一个 QPend Q，触发后一个核上的信道估计。另外，子帧中最后一个符号上的天线 7 的 Rx 包被 push 进第 3 个 QPend Q，该 QPend Q 触发均衡。剩下的所有【符号，天线】对应的 Rx 包进一个 General Purpose Q。对不同的核间分工，触发 c66x 核的细节也会不同，但只影响 Tx 描述符中的 destination Q 的配置，其它方面的设计不变。
    - 本设计用 FFTC 提供的 destination Q 本地配置域指定 RxQ。其实还可以把 FFTC 本地配置中的 destination Q 配置成无效值（0x3FFF），然后配置多个 flow，为每个 flow 配置不同的 destination Q，并为每个【符号，天线】的 Tx 描述符指定相应的 flow。
    - 如果一个上行子帧的最后一个符号是 SRS 符号，当 PUSCH 和 SRS 由不同的核处理时，收到 QPend 中断的 PUSCH 核应向 SRS 核发送核间中断，触发后者的 SRS 处理。
- UpPTS 使用和上行子帧不同的 flow。UpPTS 符号统一由一个核来处理，每个 UpPTS 符号的最后一根天线对应的 Rx 包触发对该符号的处理。UpPTS 也采用符号级乒乓缓存，且和上行子帧共用缓存，但需要注意，当 UpPTS 仅含一个符号时，该符号使用的是乓（而非乒）缓存。

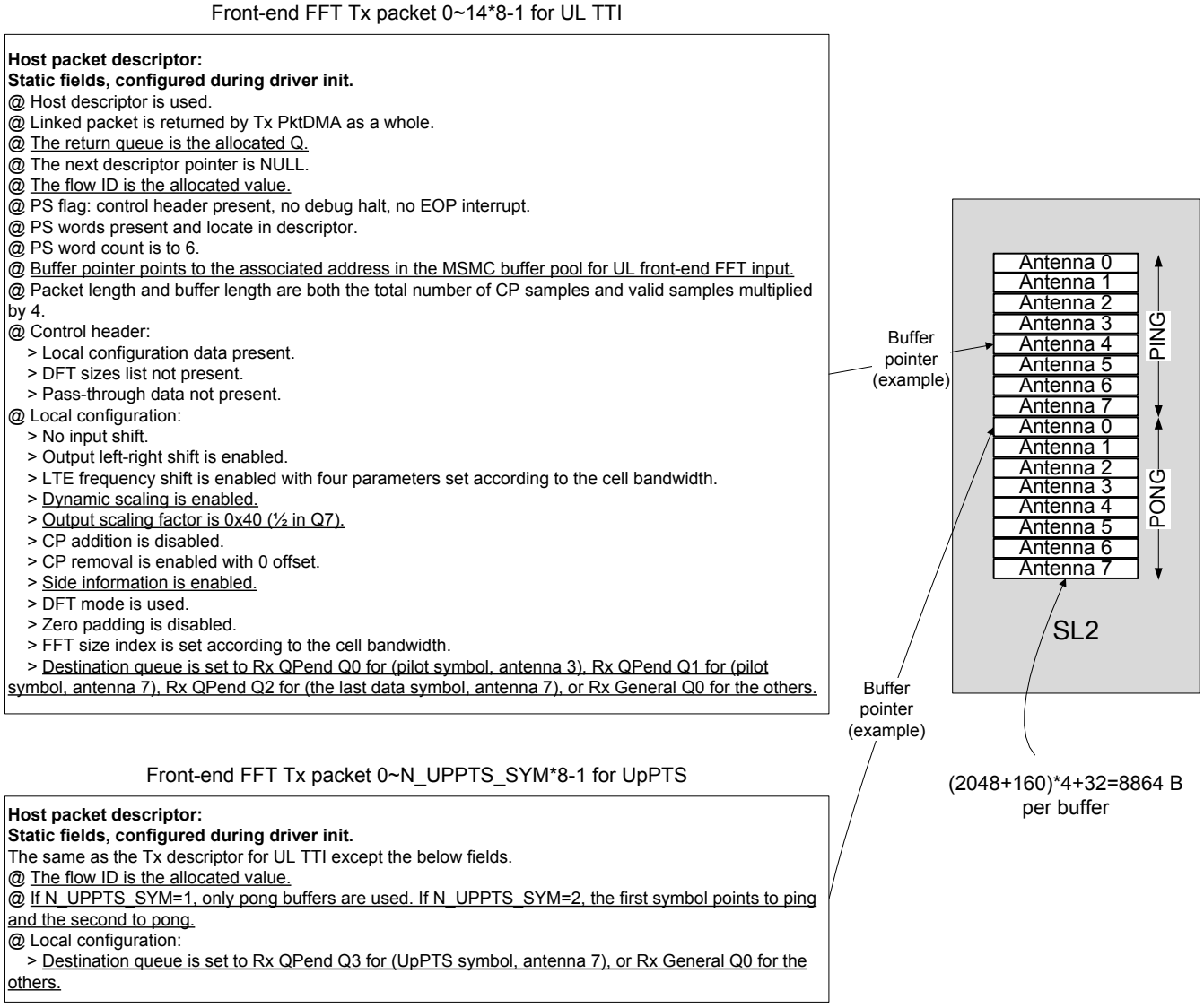


图 5: 上行前端 FFT Tx 描述符配置（例子：同图 2）

2.2 FFTC Rx 侧设计

在 Tx 侧，在一个符号内接收到的时域天线数据可以由 FFTC 在下一个符号内处理完毕，因此可以使用符号级乒乓缓存。在 Rx 侧，FFTC 输出的频域数据需要被 PUSCH、PUCCH、SRS 接收机处理，其中 PUSCH 和 PUCCH 处理不是符号级的，需要对整个子帧的数据做综合处理，且处理时间较长，理论上只要在该子帧全部收齐后 1ms 内处理完即可。因此，Rx 侧数据的生存期较长，需要使用子帧级乒乓缓存。

如前所述，在图 2 所示的例子中，上行前端 FFT 之后，PUSCH 导频符号按天线在双核间分工，PUSCH 数据符号按时隙分工。数据符号按时隙分工的优点是：当使用 IRC 均衡算法时，每个 RB 的所有 RE 共用一个噪声干扰空间相关阵R_n，因此，把一个 RB 的所有 RE 的处理集中在一个核上，有利于核尽可能多地复用加载到寄存器中的R_n元素，执行效率更高；当不做时域信道估计内插时，

一个子载波在一个时隙内的 6 个数据 RE 共用一个信道估计矩阵H，因此，把一个子载波在一个时隙内的所有 RE 的处理集中在一个核上，有利于核尽可能多地复用加载到寄存器中的H元素，执行效率更高。对不同的核间分工，基本设计不变，只是 Rx 描述符中的 buffer 指针会有不同的配置，宗旨是尽量把 FFT 输出数据直接送到将要处理该数据的那个核的 LL2，以使随后的处理有尽可能高的执行效率。

和 Tx 侧类似，在 Rx 侧，用 Acc 对 Rx General Q 中的描述符进行自动回收，但不需要产生 Acc 中断或 EDMA 事件。为了正常工作，Acc 要求在一个乒或乓 List Buffer Page 满了之后，向 Interrupt N Count 寄存器写 1，作为应用对 Acc 的响应。因为不需要产生 Acc 事件，响应时不需要写 EOI 寄存器。Tx 侧的响应由 EDMA 完成，Rx 侧的响应由核完成。对 FDD，Rx 侧的 Acc 响应周期是一个子帧；对 TDD UL/DL 配置 0~5，响应周期是 UL/DL 切换周期；对 TDD UL/DL 配置 6，响应周期是 10ms。对 FDD，Rx 侧的 Acc 响应在“TTI 配置函数”中实现；对 TDD，该响应在“UpPTS 配置函数”中实现。“TTI 配置函数”和“UpPTS 配置函数”的主要任务是修改 Rx 描述符中的动态域，并完成 Rx FDQ 入队，参见后面关于 Rx 描述符配置的部分。注意，Acc 不处理进入 QPend Q 的 Rx 描述符——如图 2 所示，这些 Rx 描述符的回收由中断处理函数执行。

为了尽量减少实时运行时修改 Rx 描述符所引入的开销，为两个子帧内的每个【符号，天线】分配一组 FFTC Rx 描述符。当软件架构要求 FFT 输出的 PUSCH 和 PUCCH 数据分开存放时，每个 Rx 包由 5 个描述符链接而成；当不要求这样的 PUSCH/PUCCH 数据分离时，每个 Rx 包由 3 个描述符链接而成。因此，总共需要分配的 Rx 描述符个数是$2 \cdot N_{\mathrm{sym}} \cdot N_{\mathrm{A}} \cdot 5$或$2 \cdot N_{\mathrm{sym}} \cdot N_{\mathrm{A}} \cdot 3$。每个 Rx 描述符的大小是 32B。

一次 FFT 的输出数据块格式如图 6 所示——先是 16 字节的 side information（如果使能的话），然后是低频侧保护子载波，接着是 PUCCH 低频侧 RB、PUSCH RB、PUCCH 高频侧 RB，最后是高频侧保护子载波。FFTC 的 Rx PktDMA 不能自动去除两边的保护子载波。为了减少内存占用，可以让所有 FFT 输出数据块的左侧和右侧保护段都输出到同一块内存中。

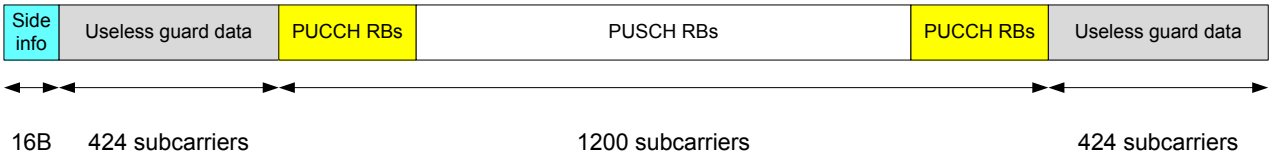


图 6: 上行前端 FFT Rx buffer 结构（例子：20M，动态 scaling）

对 PUSCH/PUCCH 数据分离的情况，图 7 以动态 scaling 为例描述了上行子帧 Rx 描述符的配置。Rx PktDMA 为每个 FFT 输出块从 Rx FDQ 中 pop 出 5 个描述符，将这 5 个描述符链接成一个 Rx 包，按顺序将输出数据块（包括 side information）写入这 5 个描述符提供的 5 个 buffer，再将这个包的 SOP 描述符（也就是构成这个包的第一个描述符）push 到 RxQ 中。因为每个上行子帧中实际承载 PUCCH 的 RB 数是动态变化的（实际上，频段外侧用于 2 类格式的 RB 数是半静态配置的，内侧用于 1 类格式的 RB 数动态变化的），所以，用于 PUSCH 和 PUCCH 的 Rx 描述符中的 original buffer length 域需要根据当前上行子帧实际包含的 PUCCH RB 数进行修改。另外，当存在位于上行子帧的 SRS 符号，且软件架构要求 SRS 使用和 PUSCH 最后一个数据符号不同的 buffer 时，上行子帧中最后一个符号对应的用于 PUSCH 的 Rx 描述符中的 original buffer pointer 域也需要根据该符号属于 PUSCH 还是 SRS 动态更新。注意，SRS 符号上的 PUCCH RB 数和非 SRS 符号可能不同（SRS 可能使 PUCCH 1 类格式采用协议中定义的 shortened 格式）。上行子帧 Rx 描述符动态域的修改以及它们的 Rx FDQ 入队操作都在“TTI 配置函数”中完成，该函数必

须在相应上行子帧开始之前、在该上行子帧之前使用相同的乒或乓描述符集合的最近上行子帧结束之后调用。为降低核负载，TTI 配置函数用 EDMA 完成 Rx 描述符的修改和入队，启动后无需等待传输结束，最多需要 3 个 EDMA 通道，分别对应非最后一个符号的 Rx 描述符修改、最后一个符号的 Rx 描述符修改、Rx 描述符入 Rx FDQ。

在图 7 中，对乒或乓，高频侧保护段的输出地址是固定的，低频侧保护段的输出地址按照 FFTC 实际处理的【符号，天线】顺序从左向右递进，每次增加 16B，目的是保存一个子帧中每个【符号，天线】对应的 side information。Side information 区域必须是子帧级乒乓的，导致保护段区域也不得不按乒乓分配。

当不要求 PUSCH 和 PUCCH 数据分离时，每个 Rx 包由 3 个描述符链接而成，PUCCH 位于每个 buffer 的两侧。

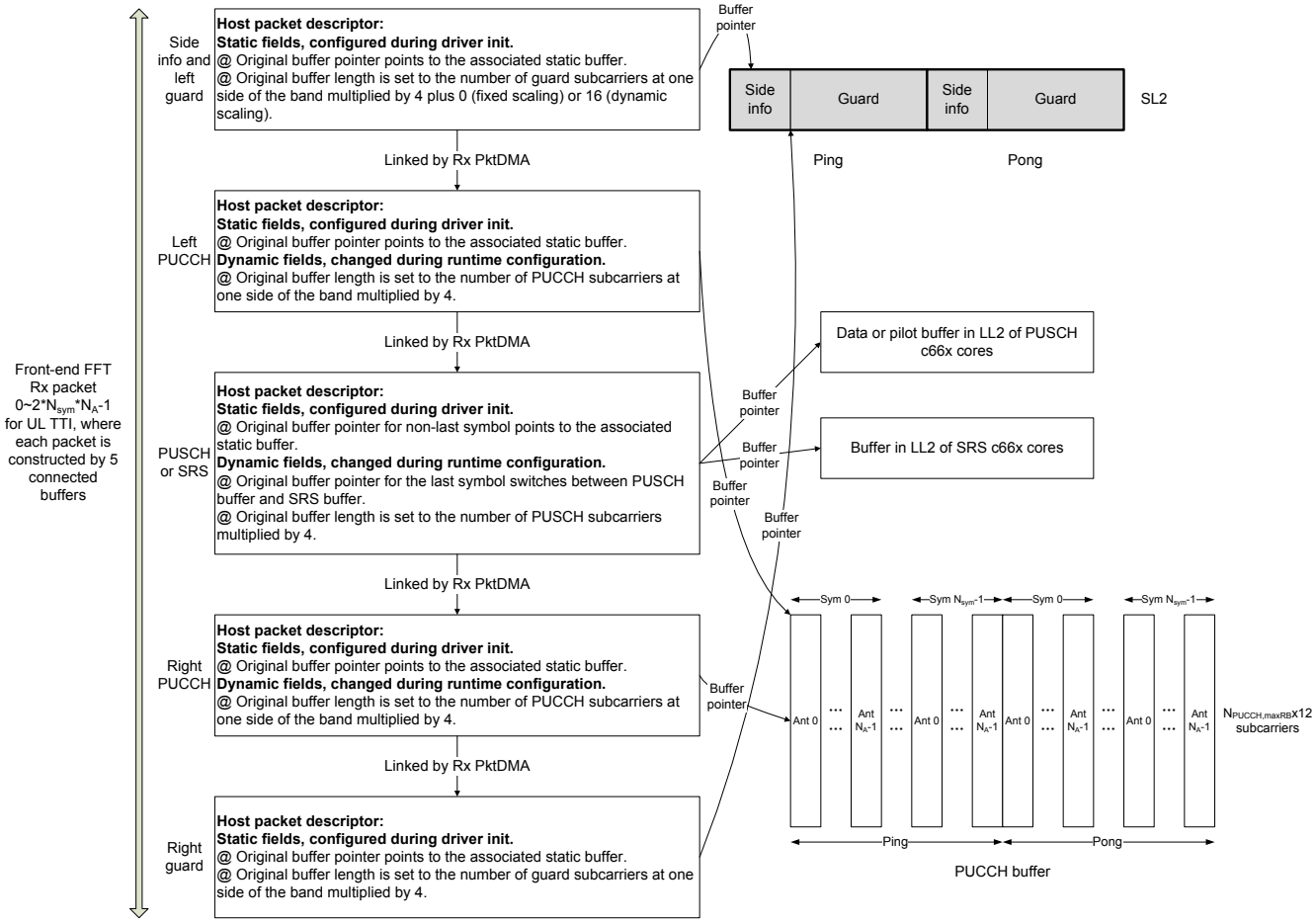


图 7: 前端 FFT 的上行子帧 Rx 描述符配置（例子：动态 scaling）

图 8 以动态 scaling 为例描述了 UpPTS Rx 描述符的配置。和上行子帧相比，UpPTS 没有 PUSCH/PUCCH 分离的问题，所以一个 Rx 包总是由 3 个描述符链接而成，而且描述符的所有域都是静态域，不需要动态修改。UpPTS Rx 描述符的 Rx FDQ 入队操作在“UpPTS 配置函数”中完成，该函数必须在相应 UpPTS 开始之前、前一个 UpPTS 结束之后调用。为降低核负载，用

EDMA 完成描述符入队。注意，对 TDD UL/DL 配置 6，“UpPTS 配置函数”每半帧调一次，但 Rx 侧 Acc 响应每 10ms 只需执行一次。

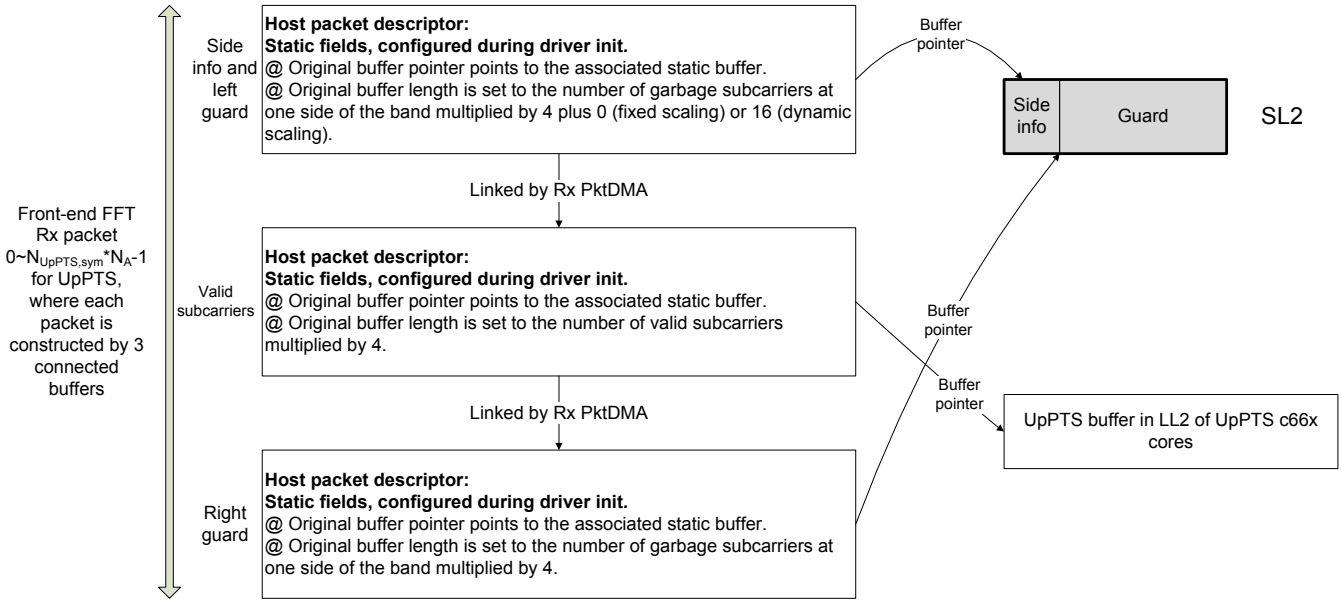


图 8: 前端 FFT 的 UpPTS Rx 描述符配置（例子：动态 scaling）

图 9 以单片 TCI6634K2K/TCI6638K2K 实现 3 载扇为例描述了从 QPend 事件到核中断的传递路径实例。在这个例子中：

- 分配给每载扇的 3 个 QPend Q 的 Q 编号分别是 652/653/654、655/656/657、658/659/660。
- 每个核的 QPend 中断索引都是 10。
- 核 0/1、2/3、4/5 分别处理三个载扇的 PUSCH。PUSCH 导频符号按天线分工，数据符号按时隙分工。每对核中的第一个核只需接收导频符号上第$N_A/2-1$（从 0 开始编号）根天线对应的 QPend 中断（触发信道估计，而均衡可以在第二个时隙的信道估计完成后立即执行）；第二个核既要接收导频符号上最后一根天线对应的 QPend 中断（触发信道估计），还要接收最后一个数据符号上最后一根天线对应的 QPend 中断（触发均衡）。
- 核 6 处理前载扇 0 和 2 的 PUCCH，核 7 处理载扇 1 的 PUCCH。因此，每个 QPend Q 654/657/660 事件要传递到 2 个核（载扇 0 是核 1 和核 6，载扇 1 是核 3 和核 7，载扇 2 是核 5 和核 6）。

关于 TCI6634K2K/TCI6638K2K 的内部事件路由，参见[19]。关于 CIC（Chip Interrupt Controller），参见[20]。

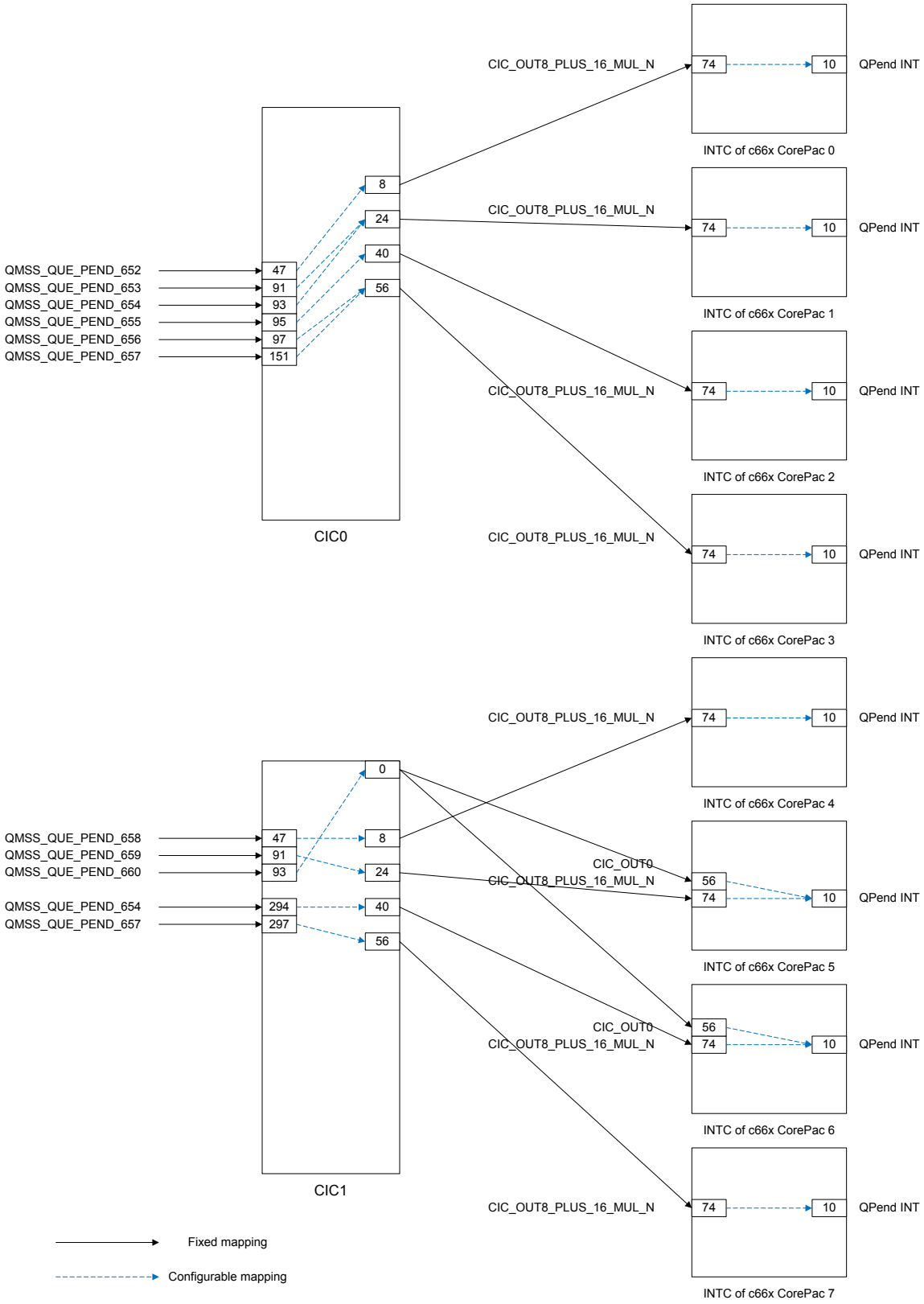


图 9: QPend 事件传递路径举例（单片 TCI6634K2K/TCI6638K2K 实现 3 载扇）

2.3 硬件资源需求

对每个载扇，上行前端 FFT 所需的硬件资源如下：

- 1 个 FFTC 实例。
- 该 FFTC 实例上的 1 个 Tx/Rx 通道（和该 FFTC 实例的一个 TxQ 绑定）。
- 1 个用于上行子帧的 flow。TDD 时，还需 1 个用于 UpPTS 的 flow。
- 1 个 Tx return Q，1 个用于上行子帧的 Rx FDQ，1 个 Rx General Q，若干个 Rx QPend Q（具体个数由核间分工确定，比如：当 PUSCH 双核处理，PUCCH 在另一个核上处理时，需要 3 个；当单核处理单载扇时，只需 1 个）。TDD 时，还需 1 个用于 UpPTS 的 Rx FDQ。
- $N_{\mathrm{sym}}N_{\mathrm{A}}$个用于上行子帧的 Tx 描述符。TDD 时，还需$N_{\mathrm{UpPTS,sym}}N_{\mathrm{A}}$个用于 UpPTS 的 Tx 描述符。每个 Tx 描述符的大小是 64B。
- $2\cdot N_{\mathrm{sym}}\cdot N_{\mathrm{A}}\cdot 5$（要求 PUSCH/PUCCH 数据分离时）或$2\cdot N_{\mathrm{sym}}\cdot N_{\mathrm{A}}\cdot 3$（不要求数据分离时）个用于上行子帧的 Rx 描述符。TDD 时，还需$N_{\mathrm{UpPTS,sym}}\cdot N_{\mathrm{A}}\cdot 3$个用于 UpPTS 的 Rx 描述符。每个 Rx 描述符的大小是 32B。
- 用于 Tx 和 Rx 描述符回收的 Acc 所使用的 PDSP 实例。
- 用于 Tx 描述符回收的 Accumulator 通道，用于 Rx 描述符回收的 Accumulator 通道。
- Tx 侧 TxQ 入队 EDMA 用到的 1 个 CC 实例，1 个 TC 实例，1 个 EDMA 通道，3（FDD）、5（TDD UL/DL 配置 0~5）或 8（TDD UL/DL 配置 6）个 PaRAM set。
- Rx 侧为上行子帧用到了 3 个 EDMA 通道，分别对应非最后一个符号的 Rx 描述符修改、最后一个符号的 Rx 描述符修改、Rx FDQ 入队。每类操作涉及 1 个 CC 实例，1 个 TC 实例，1 个 EDMA 通道，1 个 PaRAM set。Rx FDQ 入队也可以改用 QDMA，其它两类操作涉及 3D 搬移，只能用 EDMA。TDD 时，只要在使用时间上不冲突，针对上行子帧和针对 UpPTS 的操作可以合用一套资源。
- Acc 内存资源：
  - 用于 Tx 描述符回收的 List Buffer Page：
    - FDD 时，以子帧为单位乒乓，总大小为$2\cdot\left(N_{\mathrm{sym}}N_{\mathrm{A}}+1\right)\cdot 4$字节。
    - TDD UL/DL 配置 0~5 时，以 UL/DL 切换周期为单位乒乓，总大小为$2\cdot\left(\left(N_{\mathrm{sym}}N_{\mathrm{TTI}}+N_{\mathrm{UpPTS,sym}}\right)N_{\mathrm{A}}+1\right)\cdot 4$字节，其中$N_{\mathrm{TTI}}$表示一个 UL/DL 切换周期包含的上行子帧数。
    - TDD UL/DL 配置 6 时，以 10ms 为单位乒乓，总大小为$2\cdot\left(\left(5N_{\mathrm{sym}}+2N$UpPTS,sym$NA+1\right)\cdot 4$ 字节。
  - 用于 Rx 描述符回收的 List Buffer Page：

- FDD 时，以子帧为单位乒乓，总大小为$2\cdot\left(N_{\mathrm{sym}}N_{\mathrm{A}}-N_{\mathrm{TTI,QPend}}+1\right)\cdot 4$字节，其中$N_{\mathrm{TTI,QPend}}$表示一个上行子帧中进入 QPend Q 的 Rx 包个数。
- TDD UL/DL 配置 0~5 时，以 UL/DL 切换周期为单位乒乓，总大小为$2\cdot\left(\left(N_{\mathrm{sym}}\cdot N_{\mathrm{A}}-N_{\mathrm{TTI,QPend}}\right)N_{\mathrm{TTI}}+\left(N_{\mathrm{UpPTS,sym}}\cdot N_{\mathrm{A}}-N_{\mathrm{UpPTS,QPend}}\right)+1\right)\cdot 4$字节，其中$N_{\mathrm{UpPTS,QPend}}$表示一个 UpPTS 中进入 QPend Q 的 Rx 包个数。
- TDD UL/DL 配置 6 时，以 10ms 为单位乒乓，总大小为$2\cdot\left(\left(N_{\mathrm{sym}}\cdot N_{\mathrm{A}}-\right.\right.$ NTTI,QPend·5+NUpPTS,sym·NA−NUpPTS,QPend·2+1·4 字节。

只要处理能力足够，多个载扇可以共用一个 FFTC 实例、一个 CC 甚至 TC 实例。多扇区通常共用一个 Acc PDSP 实例。所有用于通知某个核或某组核的 QPend 事件可以来自同一个 QPend Q，软件通过从中 pop 出的描述符地址辨别事件类型。其它资源通常不适合多载扇共用，需要为每个载扇单独分配。

2.4 C66x 核负载测量

采用本设计，大部分配置工作在小区建立或重配时完成，需要在实时运行过程中调用的函数仅限于 TTI 配置函数和 UpPTS 配置函数。表 1 给出了 TTI 配置函数的实测 cycle 数。TTI 配置函数的任务主要是响应 Acc 以及配置/启动用于 Rx 描述符修改和入队的 3 个 EDMA，因为无需等待 EDMA 传输结束，所以该函数的负载和天线数无关。UpPTS 配置函数的负载比 TTI 配置函数更低，此处从略。本测试使用 TCI6638K2K EVM；L1P/L1D cache 初始状态都为 cold，也就是说，函数开始执行时，代码和数据都不在 L1 cache 中。

表 1: TTI 配置函数负载（cold cache）

	Cycle 数
PUSCH/PUCCH 分离	788
PUSCH/PUCCH 不分离	351

3 PRACH 前端时频转换

图 10 给出了 PRACH 接收机的流程图，本章描述的 PRACH 前端时频转换包括去 CP 和转换至频域这两个步骤。

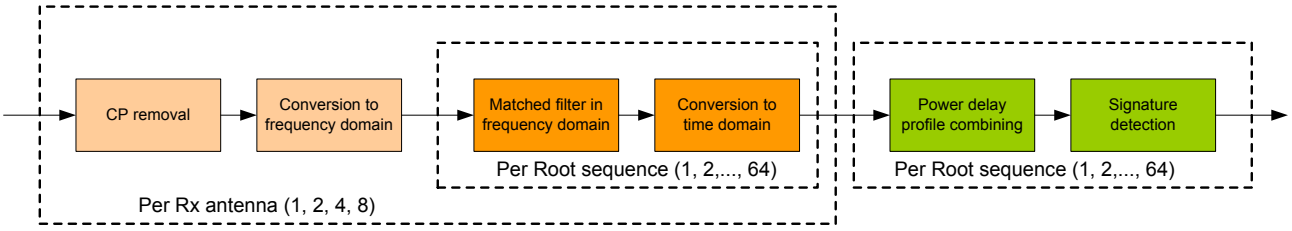


图 10: PRACH 接收机流程图

本文用短 RACH 指代 PRACH preamble 格式 4，用长 RACH 指代其它格式。表 2 给出了各种小区带宽情况下，长、短 RACH 的前端时频转换所需的 DFT 原始长度。下面分情况描述时频转换方案设计：

- 长 RACH：
  - 对 20、15、10MHz 小区，FFTC 不支持 DFT 原始长度，需要采用间接的时频转换法，具体分为两种：
    - 大点 DFT 法。该方法主要使用 FFTC，辅以少量的核处理，直接完成一个大点数的 DFT。参见 3.1 节。
    - 混合法。该方法使用核完成下变频和滤波降采样，然后用 FFTC 在降低后的采样率上执行小点 DFT。参见 3.2 节。
  - 对更窄带宽的小区，可以直接用 FFTC 执行 DFT。参见 3.3 节。
- 短 RACH：
  - 总是可以直接用 FFTC 执行 DFT。参见 3.4 节。

表 2: PRACH 前端时频转换 DFT 原始长度

带宽（MHz）	长 RACH	短 RACH
20	24576	4096
15	18432	3072
10	12288	2048
5	6144	1024
3	3072	512
1.4	1536	256

3.1 长 RACH——大点 DFT 法

本节先描述原理，再给出实现方案。

3.1.1 原理

一个长度为 N 的序列$x(n)$的 N 点 DFT 定义为

$$\mathrm{X}(k)=\sum_{n=0}^{\mathrm{N}-1}x(n)\mathrm{W}_{\mathrm{N}}^{nk};\ k=0,1,\cdots,\mathrm{N}-1$$

其中n是时间索引，k是频率索引，$\mathrm{W}_{\mathrm{N}}=e^{-j2\pi/\mathrm{N}}$。大点 DFT 法直接实现一个长度超过 FFTC 支持范围的 FFT。用于实现快速 DFT 的 FFT 算法通常可以分为时域抽取法（DIT）和频域抽取法（DIF）两类。当计算完整的 DFT 时，DIT 和 DIF 的计算复杂度是一样的。但是，PRACH 前端时频转换实际上只需要计算一部分$\mathrm{X}(k)$，比如，对 20MHz 小区、频域仅 1 个 opportunity 的情况，N=24576，但真正需要的$\mathrm{X}(k)$只有 839 点。通过对比分析可知，只有 DIT 可以利用到这个应用特性降低核负载。因此，我们选择采用 DIT 法。

DIT 法将一个 N 点 DFT 分解成 P 个 N/P 点 DFT 的方式为

$$\mathrm{X}(k)=\sum_{m=0}^{\mathrm{N/P}-1}x(\mathrm{P}\cdot m)\mathrm{W}_{\mathrm{N}}^{\mathrm{P}mk}+\mathrm{W}_{\mathrm{N}}^{k}\sum_{m=0}^{\mathrm{N/P}-1}x(\mathrm{P}\cdot m+1)\mathrm{W}_{\mathrm{N}}^{\mathrm{P}mk}+\cdots+\mathrm{W}_{\mathrm{N}}^{(\mathrm{P}-1)k}\sum_{m=0}^{\mathrm{N/P}-1}x(\mathrm{P}\cdot m+\mathrm{P}-1)\mathrm{W}_{\mathrm{N}}^{\mathrm{P}mk}$$

定义$f_p(m) = x(\mathrm{P}\cdot m + p); m = 0,1,\cdots,\mathrm{N/P}-1$，下标$p = 0,1,\cdots,\mathrm{P}-1$，定义$\mathrm{F}_p(k)$是$f_p(m)$的 N/P 点 DFT，并注意到$\mathrm{W_N^P} = \mathrm{W_{N/P}}$，可得

$$\mathrm{X}(k) = \mathrm{F}_0(k) + \mathrm{W_N^k F}_1(k) + \cdots + \mathrm{W_N^{(P-1)k} F_{P-1}}(k)$$

进一步地，定义$k = k' + p'\cdot \mathrm{N/P}$，$k' = 0,1,\cdots,\mathrm{N/P}-1$，$p' = 0,1,\cdots,\mathrm{P}-1$，因为$\mathrm{F}_p(k)$以 N/P 为周期，且$\mathrm{W_N^k} = \mathrm{W_N^{k'+p'\cdot N/P}} = \mathrm{W_P^{p'} W_N^{k'}}$，$\mathrm{X}(k)$可以拆分成如下的多个表达式

$$\mathrm{X}(k') = \mathrm{F}_0(k') + \mathrm{W_P^0 W_N^{k'} F}_1(k') + \cdots + \mathrm{W_P^{0(P-1)} W_N^{(P-1)k'} F_{P-1}}(k')$$

$$\mathrm{X}\left(k' + \frac{\mathrm{N}}{\mathrm{P}}\right) = \mathrm{F}_0(k') + \mathrm{W_P^1 W_N^{k'} F}_1(k') + \cdots + \mathrm{W_P^{1(P-1)} W_N^{(P-1)k'} F_{P-1}}(k')$$

……

$$\mathrm{X}\left(k' + (\mathrm{P}-1)\frac{\mathrm{N}}{\mathrm{P}}\right) = \mathrm{F}_0(k') + \mathrm{W_P^{P-1} W_N^{k'} F}_1(k') + \cdots + \mathrm{W_P^{(P-1)(P-1)} W_N^{(P-1)k'} F_{P-1}}(k')$$

其中$k' = 0,1,\cdots,\mathrm{N/P}-1$。这组表达式可以用图 11 表示，并描述如下：

1. 对输入 N 点序列做 P 点抽取，获得 P 个子序列，每个子序列的长度是 N/P。
2. 分别对 P 个子序列做 N/P 点 DFT 运算，得到$\mathrm{F}_p(k)$，$p = 0,1,\cdots,\mathrm{P}-1$，$k = 0,1,\cdots,\mathrm{N/P}-1$。
3. 第 *p* 个子序列$\mathrm{F}_p(k)$点乘$\mathrm{W_N^{pk}}$。
4. 对第 3 步输出的 P 个子序列，执行 P 点“块 DFT”，相当于每次取 P 个子序列相同位置上的点，构成一个 P 点序列，然后对该 P 点序列执行 P 点 DFT，第 *m* 次 P 点 DFT 输出的 P 个结果对应$\mathrm{X}(m), \mathrm{X}(m+\mathrm{N/P}), \cdots, \mathrm{X}(m+(\mathrm{P}-1)\cdot \mathrm{N/P})$。

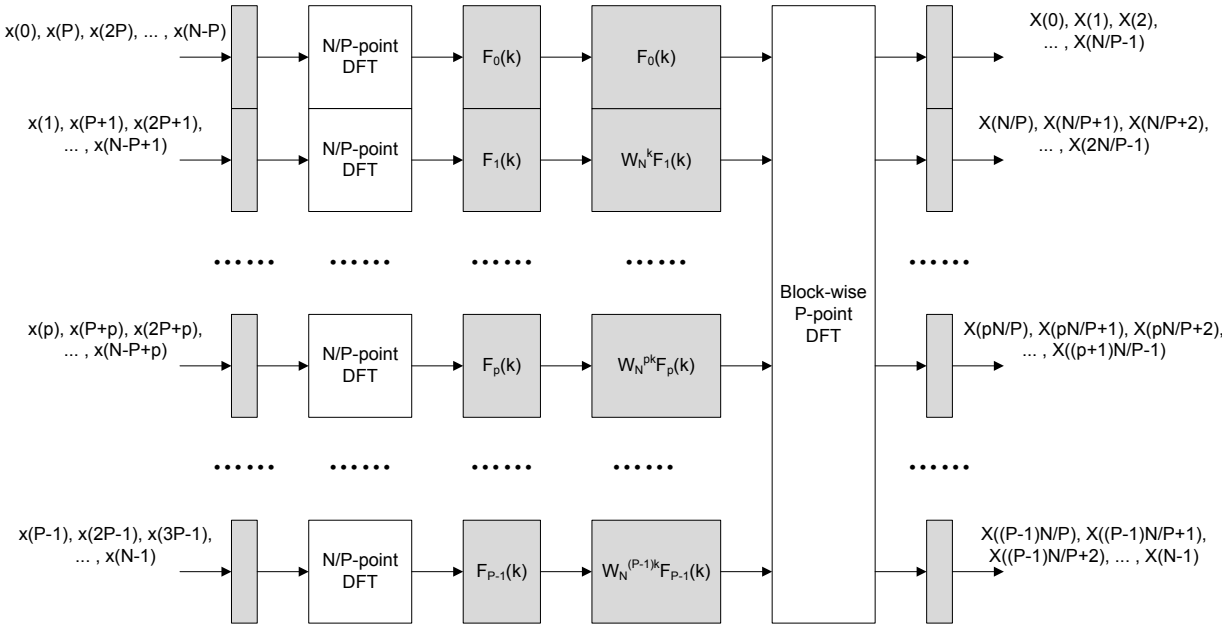


图 11: DIT 大点 DFT 原理

对 20MHz 小区，按照 FFTC 支持的 DFT 长度对整个 24576 点序列做分割，有 24576 = 8192*3 = 6144*4 = 4096*6 = 3072*8 = 2048*12 = ……，对应的抽取因子 P 分别是 3、4、6、8、12 等。对 P 的选取应综合考虑以下因素：

- C66x 核负载。P 越大，步骤 3 和 4 涉及的计算量就越大。
- 抽取 EDMA 的配置复杂度。如前一章所述，AIF 收到的上行时域天线数据是按符号乒乓缓存的，为了用 EDMA 完成 P 点抽取，最直接的方法是在 FFT 入队 EDMA 完成后，触发另一个 EDMA 把当前符号的时域数据以 P 点抽取的方式搬到别的地方去。为了 EDMA 配置的简单性，一个符号内的样本数最好是 P 的倍数。对 20M 小区、normal CP 场景，为了满足此需要，P 只能等于 1、2、4、8、16。
- FFTC 执行效率。根据[21]，1GHz KeyStone 器件上的一个 FFTC 的流量在 DFT 长度为 8192、6144、4096、3072、2048 时分别为 378、360、437、413、431 Msps。不同长度对应的流量有较为可观的差异，比如，4096 点时的流量比 6144 点时高 21.4%。

综合考虑以上因素，本文推荐使用 P=4 作为 20MHz 小区的抽取因子，对应 6144 点 DFT。对 15 和 10MHz 小区，假设采样率按带宽等比下降，则 PRACH 序列长度从 20M 时的 24576 降为 18432 和 12288。对这两类小区带宽，分别使用 P=3 和 2，而保持 6144 作为 DFT 点数。这样的选择偏向更轻的核负载和 EDMA 配置简单性，牺牲了一些 FFTC 执行效率。

3.1.2 实现

如上所述，基于 DIT 的大点 DFT 有 4 个处理步骤。实现时，步骤 1（P 点抽取）由 EDMA 完成，步骤 2 由 FFTC 完成，步骤 3 和 4 由 c66x 核完成。

抽取 EDMA 需要用到 2 个 EDMA 通道，分别称为 trigger 通道和 working 通道。图 12 以“TDD UL/DL 配置 1，PRACH 配置 10，20MHz，Normal CP”为例给出了抽取 EDMA 的配置细节。

根据附录 A，在该系统配置下，trigger 通道需要 7 个 EDMA PaRAM set 来响应上行前端 FFT 产生的长 RACH 触发事件。第 1 个 PaRAM set 只用一次，用于响应系统初始化后最初的若干上行子帧符号对应的事件。之后的 6 个 PaRAM set 以 10ms 为周期重复使用，用于处理该系统配置下一个无线帧包含的 3 个 opportunity，以及相邻 opportunity 之间不承载 PRACH 数据的上行子帧符号。对不承载 PRACH 的上行子帧符号，相应的长 RACH 触发事件不触发实际操作，本文称这样不承载实际数据搬移任务的 EDMA 为 dummy 类型的 EDMA。对承载 PRACH 的上行子帧符号，相应的长 RACH 触发事件通过 trigger 通道的 chain 机制间接触发 working 通道，并且在触发 working 通道之前先用 trigger 通道修改 working 通道将要用到的 PaRAM set 中的符号级动态域。

Working 通道需要$N_A + 3$个 PaRAM set。第 1 个 PaRAM set 只用到 1 次，属于 dummy 类型的。后面的$N_A + 2$个 PaRAM set 以 link 加 self-chain 的方式，以上行子帧符号为周期重复使用。对每个上行子帧符号，先执行 1 个 dummy 类型的 PaRAM set，用于确保 trigger 通道对接下来的$N_A + 1$个 PaRAM set 的动态域的修改已经完成。接下来的N_A个 PaRAM set 用于完成N_A根天线的 P 点抽取。最后 1 个 PaRAM set 用于在整个序列的最后一个符号处执行 FFTC TxQ 入队操作，非序列最后一个符号对应的该 PaRAM set 属于 dummy 类型。

Working 通道的后$N_A + 1$个 PaRAM set 中的动态域包括 SRC、DST 和 BCNT。Trigger 通道为每个上行子帧符号执行一次对这$N_A + 1$个 working 通道 PaRAM set 的动态更新，数据源位于内存中，称为 PaRAM set LUT，其内容在小区初始化或重配时由驱动软件配置。

对图 12 示例之外的其它系统配置，基本原理是类似的，但细节上有一些差别。

- Trigger 通道用到的 PaRAM set 个数等于附录 A 表格中“EDMA 分段”列给出的值加 1。
- Trigger 通道的每个 PaRAM set 负责响应的上行子帧符号数等于附录 A 表格中“初始 Dummy 符号数”列或“EDMA 段内符号数”列给出的值。对 dummy 类型的 PaRAM set，这个值配置到 BCNT；对其它 PaRAM set，配置到 CCNT。
- 对 PRACH 格式 0 和 1，一个抽取目的 buffer 包含 1 个 PRACH 序列，仅需为最后一个符号执行 FFTC 入队操作。对 PRACH 格式 2 和 3，一个抽取目的 buffer 包含 2 个序列。此时，为了在一个序列收齐之后立刻启动 FFTC，需要为每个序列的最后一个符号执行 FFTC 入队操作。这些都通过配置 PaRAM set LUT 实现。
- 对 PRACH 格式 0，如果两个相邻的上行子帧都承载了 PRACH，需要分配两个抽取目的 buffer，以子帧为单位乒乓使用。此时，每个 buffer 有自己的 PaRAM set LUT。对其它 PRACH 格式，一个 buffer 就够了，但格式 2 和 3 的一个 buffer 实际上包含 2 个序列。
- 对 PRACH 格式 2 和 3，不能把一个 opportunity 的两个序列的抽取输出连续排列，这是因为 DCIDX 域最大只有 16 位（包括符号位），连续排列的话正好超出了表达范围。
- 对 PRACH 格式 2 和 3，为了让两个序列各自放在连续的内存中，需要对两个序列交接处的那个符号的抽取 EDMA 做特殊处理。为此，对 working 通道，在原有的$N_A + 3$个 PaRAM set 的基础上增加N_A个，这些新增的 PaRAM set 不需要动态更新，负责处理交接符号中属于后一个序列的部分的抽取，而原有的用于抽取的N_A个 PaRAM set 现在只处理属于前一个序列的部分。这两组 PaRAM set 以相互交织的方式 link 和 self-chain，为此，LINK 必须成为动态更新域，以便动态改变 link 目标。

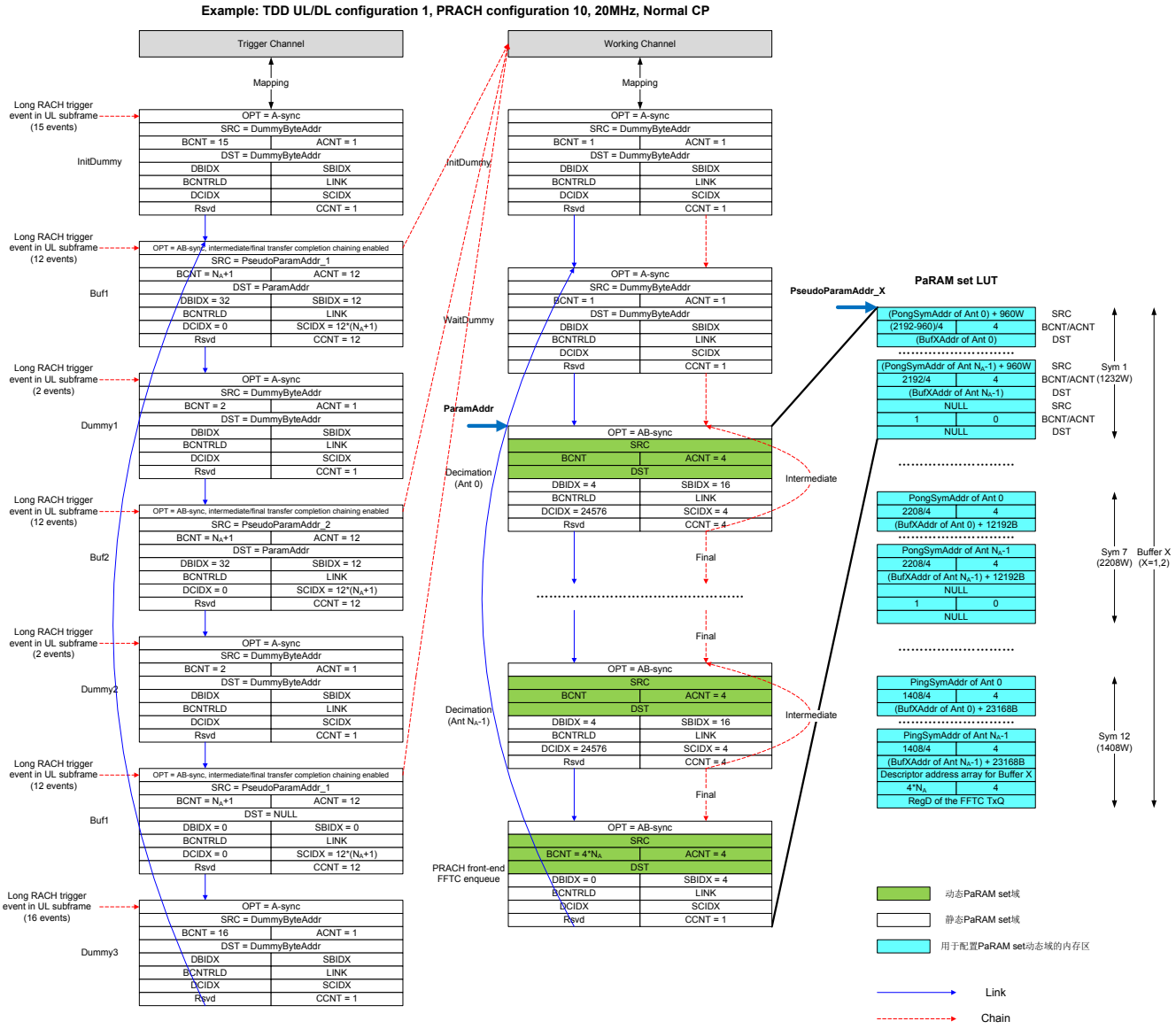


图 12: 抽取 EDMA 配置举例

步骤 2，也就是 6144 点 DFT，由 FFTC 完成。这一步采用原位操作以节省内存。推荐采用动态 scaling，此时每个 6144 点的 buffer 需要在头部多留出 16B 以容纳 FFTC 输出的 side information。

步骤 3 和 4 由核实现。一个 PRACH opportunity 在频域占用 6 个 RB，但并不是整个 6 RB 带宽都被 PRACH 占用，因为两侧有相同宽度的保护频带。图 13 给出了一个例子，其中，PRACH 导频子载波被标记为红色。

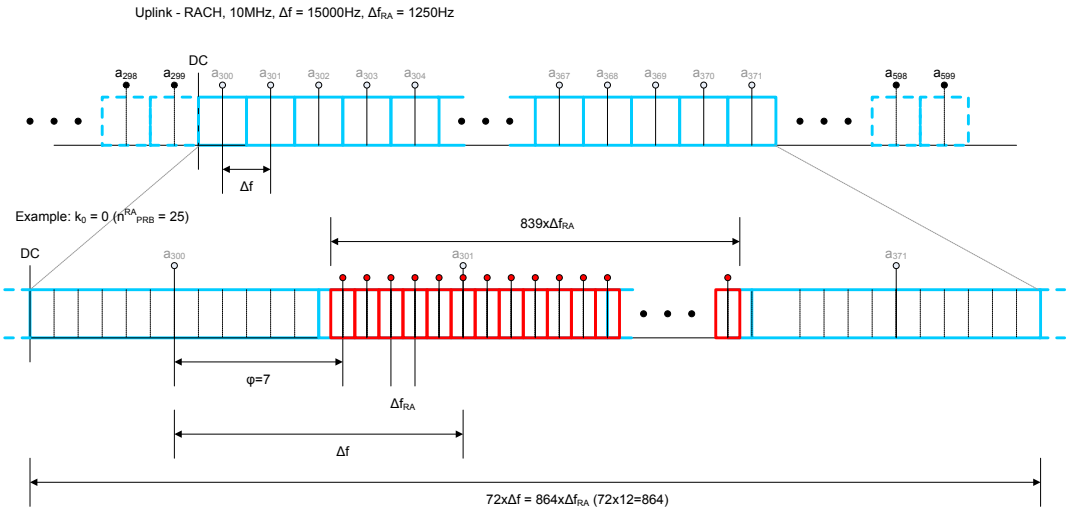


图 13: PRACH 导频的频域位置

对 FDD，每小区在一个特定时刻最多只有 1 个 opportunity，其起始物理 RB 索引是$n_{\mathrm{PRB}}^{\mathrm{RA}} = n_{\mathrm{PRB\,offset}}^{\mathrm{RA}}$，其中，$n_{\mathrm{PRB\,offset}}^{\mathrm{RA}}$表示半静态的 PRACH 频偏参数。

对 TDD，一个小区在一个特定时刻可以有多个 opportunity，其中第f_{RA}个 opportunity 的起始物理 RB 索引是$n_{\mathrm{PRB}}^{\mathrm{RA}} = \begin{cases} n_{\mathrm{PRB\,offset}}^{\mathrm{RA}} + 6\left\lfloor \frac{f_{\mathrm{RA}}}{2} \right\rfloor, \text{if } f_{\mathrm{RA}} \bmod 2 = 0 \\ \mathrm{N}_{\mathrm{RB}}^{\mathrm{UL}} - 6 - n_{\mathrm{PRB\,offset}}^{\mathrm{RA}} - 6\left\lfloor \frac{f_{\mathrm{RA}}}{2} \right\rfloor, \text{otherwise} \end{cases}$，其中$\mathrm{N}_{\mathrm{RB}}^{\mathrm{UL}}$表示小区上行 RB 数。

假设一个 PRACH opportunity 的起始物理 RB 索引是$n_{\mathrm{PRB}}^{\mathrm{RA}}$，以 20MHz 小区为例，它的第一个子载波在整个 24576 点 DFT 输出中的位置是

$$i_{\mathrm{start}} = \left(\left((2048-1200)/2 + 12 \cdot n_{\mathrm{PRB}}^{\mathrm{RA}}\right) \cdot 12 + 13\right) + 12288\Big) \bmod 24576 = \left(144 \cdot n_{\mathrm{PRB}}^{\mathrm{RA}} + 17389\right) \bmod 24576$$

参考图 11，假设该起始位置落在第$b_0 \in [0, \mathrm{P}-1]$个大点 DFT 输出块内，在该块内的内部偏移是$o_0$，则$b_0 = \lfloor i_{\mathrm{start}}/6144 \rfloor$，$o_0 = i_{\mathrm{start}} \bmod 6144$。如果$o_0 \le 6144 - 839 = 5305$，这个 opportunity 位于一个大点 DFT 输出块内；否则，这个 opportunity 跨越 2 个大点 DFT 输出块，其最后$o_0 - 5305$个点位于块$b_1 = (b_0 + 1) \bmod 6$的起始处。

如果该 opportunity 位于一个输出块内，核执行下面的计算

$$\mathrm{Y}(k) = \mathrm{X}_{b_0}(o_0 + k) = \sum_{p=0}^{3} \mathrm{W}_4^{pb_0} \mathrm{W}_{24576}^{p(o_0+k)} \mathrm{F}_p(o_0 + k); k = 0,1,\cdots,838$$

如果跨越 2 个块，计算

$$\mathrm{Y}(k) = \mathrm{X}_{b_0}(o_0 + k) = \sum_{p=0}^{3} \mathrm{W}_4^{pb_0} \mathrm{W}_{24576}^{p(o_0+k)} \mathrm{F}_p(o_0 + k); k = 0,1,\cdots,6143 - o_0$$

$$\mathrm{Y}(k)=\mathrm{X}_{b_1}(k-6144+o_0)=\sum_{p=0}^{3}\mathrm{W}_4^{pb_1}\mathrm{W}_{24576}^{p(k-6144+o_0)}\,\mathrm{F}_p(k-6144+o_0);k=6144-o_0,\cdots,838$$

当 FFTC 为步骤 2 执行动态 scaling 时，核在处理步骤 3 和 4 时，除了上述公式描述的计算，还要执行定标对齐。

3.1.3 硬件资源需求

对每个载扇，PRACH 前端时频转换所需的硬件资源如下：

- 1 个 FFTC 实例。
- 该 FFTC 实例上的 1 个 Tx/Rx 通道（和该 FFTC 实例的一个 TxQ 绑定）。
- 1 个 flow。
- 1 个 Tx return Q，1 个 Rx FDQ，1 个 Rx QPend Q。
- $N_{\mathrm{X}}\cdot N_{\mathrm{S}}\cdot N_{\mathrm{A}}\cdot \mathrm{P}$个 Tx 描述符，其中$N_{\mathrm{S}}$表示一个 opportunity 中的序列个数，$N_{\mathrm{X}}$表示前面所说的抽取目的 buffer 个数（1 或 2，仅格式 0 在部分配置下取 2）。每个 Tx 描述符的大小是 32B。
- Rx 描述符的个数和 Tx 描述符一样。每个 Rx 描述符的大小是 32B。
- 抽取 EDMA 用到的 1 个 CC 实例，1 个 TC 实例，2 个 EDMA 通道。Trigger 通道需要的 PaRAM set 个数等于附录 A 表格的“EDMA 分段”列取值加 1，最大值等于 11，发生在 TDD UL/DL 配置 0、PRACH 配置为 15/16/17 时。对 PRACH 格式 0 或 1，working 通道需要 $\mathrm{N_A}+3$个 PaRAM set；对格式 2 或 3，需要$2\mathrm{N_A}+3$个。
- 用于控制的内存资源：
  - PaRAM set LUT，PRACH 格式 0 或 1 时的大小是$N_{\mathrm{X}}\cdot l\cdot(N_{\mathrm{A}}+1)\cdot 12$字节，格式 2 或 3 时的大小是$l\cdot(N_{\mathrm{A}}+1)\cdot 20$字节，其中$l$表示一个 opportunity 的序列部分占用的符号数，如附录 A 中的表 4 所示。
  - 描述符地址数组（用于 PRACH 前端 FFTC 入队的源 buffer），大小是 Tx 描述符个数乘以 4 字节。

只要处理能力足够，多个载扇可以共用一个 FFTC 实例、一个 CC 甚至 TC 实例。所有用于通知某个核或某组核的 QPend 事件可以来自同一个 QPend Q，软件通过从中 pop 出的描述符地址辨别事件类型。其它资源通常不适合多载扇共用，需要为每个载扇单独分配。

3.1.4 负载测量

步骤 3 和 4 在 c66x 核上的硬件实测负载如表 3 所示，对应一个 opportunity。如果在频域有多个 opportunity，则总 cycle 数要乘以 opportunity 个数。本测试使用 TMS320C6670 EVM，内存配置是：输入、输出 buffer 都位于 LL2；L1P/L1D cache 初始状态都为 cold，也就是说，函数开始执行时，代码和数据都不在 L1 cache 中。

表 3: PRACH 前端处理步骤 3 和 4 的硬件实测 cycle 数

	P=2	P=4	P=6	P=8
2 天线	6,270	9,433	14,234	20,942
4 天线	12,420	18,578	27,977	41,491
8 天线	24,705	36,666	55,579	82,683

3.2 长 RACH——混合法

混合法的原理如图 14 所示。C66x 核负责下变频和低通滤波抽取，并启动 FFTC 完成降采样序列的 DFT。当频域有多个 opportunity 时，为每个 opportunity 分别执行整个处理流程。降采样后的序列长度等于 1536，20、15、10MHz 小区分别对应 16、12、8 倍降采样。

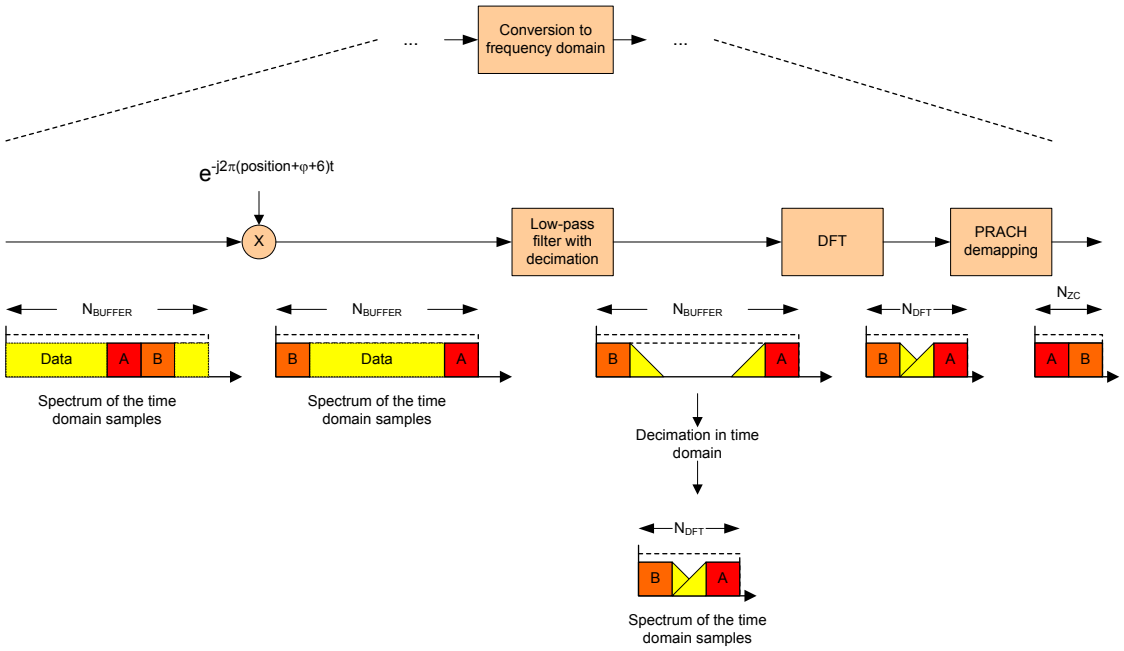


图 14: 混合法原理

C66x 核处理任务可以每符号执行一次，也可以在一个 PRACH 序列收集完毕后执行一次。前者的优点是节省内存；缺点是由多任务相互打断而导致的 cache 开销和任务切换开销，由“重叠相加”或“重叠保留”分段滤波带来的额外计算量，以及更频繁的中断开销。反之，后者的优点是核开销较低，缺点是需要更多的内存。对后者，如果因片内内存不够而需要把 PRACH 序列收集 buffer 放到 DDR，则核开销方面的优点可能会被更低的内存访问效率抵消。

采用逐符号处理时，PRACH 前端 EDMA 配置和图 12 之间的区别是：

- 只用到了 trigger 通道，working 通道及其 PaRAM set 不再需要。
- 原本用于触发 working 通道的事件现在用于触发核中断，相应的 PaRAM set 仅做 dummy 数据搬运。

采用逐序列处理时，PRACH 前端 EDMA 配置和图 12 之间的区别是：

- PaRAM set LUT 中的配置参数对应连续内存搬移，而非抽取操作。
- Working 通道用于 FFTC 入队的 PaRAM set 不再需要，而它之前的那个 PaRAM set 的 final 事件不仅触发 self-chain 事件，还产生核中断。

下变频和低通滤波抽取在 c66x 核上的硬件实测负载为：一个 opportunity、一个序列、一根天线上的 cycle 数是 55,958。本测试使用 TMS320C6670 EVM，系统参数和内存配置是：20MHz 小区，32 阶滤波器；分为 12 段按重叠相加法处理；输入、输出 buffer 都位于 LL2；L1P/L1D cache 初始状态都为 cold。注意，这里测量的负载没有包含多任务互相打断而导致的额外开销。

3.3 长 RACH——直接 DFT

5、3、 1.4MHz 小区中的长 RACH 使用直接 DFT，因为根据表 2，这些小区带宽下的 DFT 原始长度可以被 FFTC 直接处理。此时，PRACH 前端 EDMA 配置类似图 12，唯一的区别是 PaRAM set LUT 中的配置参数对应连续内存搬移，而非抽取操作。

直接 DFT 没有实时 c66x 核开销。

3.4 短 RACH

UpPTS 的长度可以是 1 或 2 个符号，而短 RACH 仅在 UpPTS 长度是 2 个符号时才允许存在。短 RACH 在 UE 侧 UpPTS 结束点之前 4832Ts 处发射，CP 长度是 448Ts，序列部分从第 1 个 UpPTS 符号的 0Ts（normal CP）或 736Ts（extended CP）开始，长度是 4096Ts。

如图 4 所示，前端 FFT 入队 EDMA 可以为 UpPTS 中的每个符号产生一个短 RACH 事件，该事件触发短 RACH 的 working 通道，完成数据搬移和 FFTC 入队。短 RACH 的 working 通道使用 4 个 PaRAM set，第一个用于初始 dummy，接下来的两个分别用于两个符号的数据搬移，最后一个用于 FFTC 入队。

对每个载扇，短 RACH 前端时频转换所需的硬件资源如下：

- 1 个 FFTC 实例。
- 该 FFTC 实例上的 1 个 Tx/Rx 通道（和该 FFTC 实例的一个 TxQ 绑定）。
- 1 个 flow。
- 1 个 Tx return Q，1 个 Rx FDQ，1 个 Rx QPend Q。
- N_A个 Tx 描述符。每个 Tx 描述符的大小是 32B。
- Rx 描述符的个数和 Tx 描述符一样。每个 Rx 描述符的大小是 32B。
- 数据搬移 EDMA 用到的 1 个 CC 实例，1 个 TC 实例，1 个 EDMA 通道，4 个 PaRAM set。
- 用于控制的内存资源：
    - 描述符地址数组（用于短 RACH 前端 FFTC 入队的源 buffer），大小是 Tx 描述符个数乘以 4 字节。

只要处理能力足够，多个载扇可以共用一个 FFTC 实例、一个 CC 甚至 TC 实例。所有用于通知某个核或某组核的 QPend 事件可以来自同一个 QPend Q，软件通过从中 pop 出的描述符地址辨别事件类型。其它资源通常不适合多载扇共用，需要为每个载扇单独分配。

4 小结

本文详细描述了在 TI 的 KeyStone 器件上实现高效的 LTE 上行基带前端处理的方法，包括对常规前端 FFT 的处理和对 PRACH 前端时频转换的处理，并给出了实测的 c66x 核负载。

从本文的描述可见，KeyStone 架构提供的 EDMA 和 Navigator 机制非常灵活，可以把数据搬运、加速器启动、核触发等操作步骤串联成一个可自动执行的整体流程，极大降低了对核实时干预的需求。

当输入数据位于 DDR 时，FFTC 的执行效率较低。因此，对 PRACH 前端处理，建议把从符号级乒乓缓存搬移出来的数据放在片内。

对 PRACH 前端处理，大点 DFT 法比混合法节省 10 倍以上的核负载。当 FFTC 和片内存储资源足够时（比如对小基站），大点 DFT 较为合适。在其它情况下，哪种方法合适需要根据系统对 c66x 核、FFTC、内存的使用情况选择。

参考文献

[1] *3GPP TS-36.211*
[2] *3GPP TS-36.212*
[3] *3GPP TS-36.213*
[4] *3GPP TS-36.214*
[5] http://www.ti.com/product/tms320tci6616
[6] http://www.ti.com/product/tms320tci6618
[7] http://www.ti.com/product/tms320tci6614
[8] http://www.ti.com/product/tms320tci6612
[9] http://www.ti.com/product/tms320c6670
[10] http://www.ti.com/product/tci6636k2h
[11] http://www.ti.com/product/tci6634k2k
[12] http://www.ti.com/product/tci6638k2k
[13] http://www.ti.com/product/tci6630k2l
[14] *KeyStone Architecture Antenna Interface 2 (AIF2) User Guide* (SPRUGV7)
[15] *KeyStone II Architecture Antenna Interface 2 (AIF2) User Guide* (SPRUHL2)
[16] *Multicore Navigator User Guide* (SPRUGR9)
[17] *Enhanced Direct Memory Access (EDMA3) Controller User Guide* (SPRUGS5)
[18] *Fast Fourier Transform Coprocessor (FFTC) User Guide* (SPRUGS2)
[19] *TCI6638K2K Data Manual* (SPRS836)
[20] *KeyStone Architecture Chip Interrupt Controller (CIC) User Guide* (SPRUGW4)
[21] *Throughput Performance Guide for TCI66x KeyStone Devices* (SPRABH2)

附录 A：PRACH 分布与 EDMA 参数

表 5 到表 12 分别给出了 FDD 以及 7 种 TDD UL/DL 配置下的 PRACH 分布描述表。解释如下。

- “缓存数”一列描述每种配置下需要的缓存个数，等于 1 或 2，2 表示需要乒乓缓存。对于 PRACH 格式 2 和 3，每个 opportunity 包含 2 个序列，1 个缓存对应一个 opportunity，而非一个序列。格式 2 和 3 时，缓存数总是等于 1，为了强调该缓存实际上要容纳 2 个序列，在表中用“1’”标识。当缓存数等于 1 时，这个唯一的缓存用 1 编号；当缓存数等于 2 时，两个缓存分别用 1 和 2 编号。确定缓存数时，假设系统能够在“下下个”序列到达前处理完当前序列的时频转换。

- 假设系统启动后总是从一个无线帧的第一个子帧开始接收天线数据，则一开始会有若干个上行子帧符号不承载 PRACH 数据，这些上行子帧符号的个数在“初始 Dummy 符号数”一列给出。注意，如前所述，经过 FFT 入队 EDMA 的过滤，PRACH 前端时频转换功能模块只会收到上行子帧对应的符号级事件。

- 经过初始阶段的不承载 PRACH 数据的若干个符号后，随后的 PRACH 时域承载模式按周期重复，每个周期包含若干个 EDMA 分段，在“EDMA 分段”一列给出。1 表示该分段内的符号数据被 EDMA 搬运到缓存 1，2 表示该分段内的符号数据被 EDMA 搬运到缓存 2，0 表示在该分段内不执行实际的天线数据搬运，只是消耗长 RACH 触发事件。“EDMA 段内符号数”一列给出每个 EDMA 分段中的上行子帧符号数。

n表示一个上行子帧内的符号数，l_x（$x = 0, 1, 2, 3$）表示格式x的一个 opportunity 中的 1 或 2 个序列跨越的总符号数，h_x表示格式x的一个 opportunity 从起始子帧的头部到序列开始前不承载 PRACH 的符号数（字母 h 表示 head），t_x表示格式x的一个 opportunity 从序列结束后到结束子帧的末尾的不承载 PRACH 的符号数（字母 t 表示 tail）。

- 表 4 给出两种 CP 模式下，这些变量的取值。

表 4: 表 5 到表 12 中的变量取值

格式		0			1			2			3		
变量	n	h_0	l_0	t_0	h_1	l_1	t_1	h_2	l_2	t_2	h_3	l_3	t_3
Normal CP	14	1	12	1	9	12	7	2	24	2	9	23	10
Extended CP	12	1	10	1	8	10	6	2	20	2	8	20	8

表 5: PRACH 分布与 EDMA 参数（FDD）

PRACH 配置	格式	系统帧	子帧	缓存数	初始Dummy 符号数	EDMA分段	EDMA段内 符号数	偶数无线帧/无线帧 0 1 2 3 4 5 6 7 8 9	奇数无线帧 0 1 2 3 4 5 6 7 8 9
0	0	Even	1	1	n+h0	1,0	l0, t0+19n+h0		
1	0	Even	4	1	4n+h0	1,0	l0, t0+19n+h0		
2	0	Even	7	1	7n+h0	1,0	l0, t0+19n+h0		
3	0	Any	1	1	n+h0	1,0	l0, t0+9n+h0		
4	0	Any	4	1	4n+h0	1,0	l0, t0+9n+h0		
5	0	Any	7	1	7n+h0	1,0	l0, t0+9n+h0		
6	0	Any	1, 6	1	n+h0	1,0	l0, t0+4n+h0		
7	0	Any	2 ,7	1	2n+h0	1,0	l0, t0+4n+h0		
8	0	Any	3, 8	1	3n+h0	1,0	l0, t0+4n+h0		
9	0	Any	1, 4, 7	1	n+h0	1,0,1,0,1,0	l0, t0+2n+h0, l0, t0+2n+h0, l0, t0+3n+h0		
10	0	Any	2, 5, 8	1	2n+h0	1,0,1,0,1,0	l0, t0+2n+h0, l0, t0+2n+h0, l0, t0+3n+h0		
11	0	Any	3, 6, 9	1	3n+h0	1,0,1,0,1,0	l0, t0+2n+h0, l0, t0+2n+h0, l0, t0+3n+h0		
12	0	Any	0, 2, 4, 6, 8	1	h0	1,0	l0, t0+n+h0		
13	0	Any	1, 3, 5, 7, 9	1	n+h0	1,0	l0, t0+n+h0		
14	0	Any	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	2	h0	1,0,2,0	l0, t0+h0, l0, t0+h0		
15	0	Even	9	1	9n+h0	1,0	l0, t0+19n+h0		
16	1	Even	1	1	n+h1	1,0	l1, t1+18n+h1		
17	1	Even	4	1	4n+h1	1,0	l1, t1+18n+h1		
18	1	Even	7	1	7n+h1	1,0	l1, t1+18n+h1		
19	1	Any	1	1	n+h1	1,0	l1, t1+8n+h1		
20	1	Any	4	1	4n+h1	1,0	l1, t1+8n+h1		
21	1	Any	7	1	7n+h1	1,0	l1, t1+8n+h1		
22	1	Any	1, 6	1	n+h1	1,0	l1, t1+3n+h1		
23	1	Any	2 ,7	1	2n+h1	1,0	l1, t1+3n+h1		
24	1	Any	3, 8	1	3n+h1	1,0	l1, t1+3n+h1		
25	1	Any	1, 4, 7	1	n+h1	1,0,1,0,1,0	l1, t1+n+h1, l1, t1+n+h1, l1, t1+2n+h1		
26	1	Any	2, 5, 8	1	2n+h1	1,0,1,0,1,0	l1, t1+n+h1, l1, t1+n+h1, l1, t1+2n+h1		
27	1	Any	3, 6, 9	1	3n+h1	1,0,1,0,1,0	l1, t1+n+h1, l1, t1+n+h1, l1, t1+2n+h1		
28	1	Any	0, 2, 4, 6, 8	1	h1	1,0	l1, t1+h1		
29	1	Any	1, 3, 5, 7, 9	1	n+h1	1,0	l1, t1+h1		
30	N/A	N/A	N/A						
31	1	Even	9	1	9n+h1	1,0	l1, t1+18n+h1		
32	2	Even	1	1'	n+h2	1',0	l2, t2+18n+h2		
33	2	Even	4	1'	4n+h2	1',0	l2, t2+18n+h2		
34	2	Even	7	1'	7n+h2	1',0	l2, t2+18n+h2		
35	2	Any	1	1'	n+h2	1',0	l2, t2+8n+h2		
36	2	Any	4	1'	4n+h2	1',0	l2, t2+8n+h2		
37	2	Any	7	1'	7n+h2	1',0	l2, t2+8n+h2		
38	2	Any	1, 6	1'	n+h2	1',0	l2, t2+3n+h2		
39	2	Any	2 ,7	1'	2n+h2	1',0	l2, t2+3n+h2		
40	2	Any	3, 8	1'	3n+h2	1',0	l2, t2+3n+h2		
41	2	Any	1, 4, 7	1'	n+h2	1',0,1',0,1',0	l2, t2+n+h2, l2, t2+n+h2, l2, t2+2n+h2		
42	2	Any	2, 5, 8	1'	2n+h2	1',0,1',0,1',0	l2, t2+n+h2, l2, t2+n+h2, l2, t2+2n+h2		
43	2	Any	3, 6, 9	1'	3n+h2	1',0,1',0,1',0	l2, t2+n+h2, l2, t2+n+h2, l2, t2+2n+h2		
44	2	Any	0, 2, 4, 6, 8	1'	h2	1',0	l2, t2+h2		
45	2	Any	1, 3, 5, 7, 9	1'	n+h2	1',0	l2, t2+h2		
46	N/A	N/A	N/A						
47	2	Even	9	1'	9n+h2	1',0	l2, t2+18n+h2		
48	3	Even	1	1'	n+h3	1',0	l3, t3+17n+h3		
49	3	Even	4	1'	4n+h3	1',0	l3, t3+17n+h3		
50	3	Even	7	1'	7n+h3	1',0	l3, t3+17n+h3		
51	3	Any	1	1'	n+h3	1',0	l3, t3+7n+h3		
52	3	Any	4	1'	4n+h3	1',0	l3, t3+7n+h3		
53	3	Any	7	1'	7n+h3	1',0	l3, t3+7n+h3		
54	3	Any	1, 6	1'	n+h3	1',0	l3, t3+2n+h3		
55	3	Any	2 ,7	1'	2n+h3	1',0	l3, t3+2n+h3		
56	3	Any	3, 8	1'	3n+h3	1',0	l3, t3+2n+h3		
57	3	Any	1, 4, 7	1'	n+h3	1',0,1',0,1',0	l3, t3+h3, l3, t3+h3, l3, t3+n+h3		
58	3	Any	2, 5, 8	1'	2n+h3	1',0,1',0,1',0	l3, t3+h3, l3, t3+h3, l3, t3+n+h3		
59	3	Any	3, 6, 9	1'	3n+h3	1',0,1',0,1',0	l3, t3+h3, l3, t3+h3, l3, t3+n+h3		
60	N/A	N/A	N/A						
61	N/A	N/A	N/A						
62	N/A	N/A	N/A						
63	3	Even	9	1'	9n+h3	1',0	l3, t3+17n+h3		

表 6: PRACH 分布与 EDMA 参数（TDD UL/DL 配置 0）

PRACH 配置	格式	密度	版本	TDD配置0 时频分布	缓存数	初始Dummy 符号数	EDMA分段	EDMA段内 符号数
0	0	0.5	0	(0,1,0,2)	1	2n+h0	1,0	I0, t0+11n+h0
1	0	0.5	1	(0,2,0,2)	1	8n+h0	1,0	I0, t0+11n+h0
2	0	0.5	2	(0,1,1,2)	1	5n+h0	1,0	I0, t0+11n+h0
3	0	1	0	(0,0,0,2)	1	2n+h0	1,0	I0, t0+5n+h0
4	0	1	1	(0,0,1,2)	1	5n+h0	1,0	I0, t0+5n+h0
5	0	1	2	(0,0,0,1)	1	n+h0	1,0	I0, t0+5n+h0
6	0	2	0	(0,0,0,2)(0,0,1,2)	1	2n+h0	1,0	I0, t0+2n+h0
7	0	2	1	(0,0,0,1)(0,0,1,1)	1	n+h0	1,0	I0, t0+2n+h0
8	0	2	2	(0,0,0,0)(0,0,1,0)	1	h0	1,0	I0, t0+2n+h0
9	0	3	0	(0,0,0,1)(0,0,0,2)(0,0,1,2)	2	n+h0	1,0,2,0, 1,0	I0, t0+h0, I0, t0+2n+h0, I0, t0+n+h0
10	0	3	1	(0,0,0,0)(0,0,1,0) (0,0,1,1)	2	h0	1,0,2,0, 1,0	I0, t0+2n+h0, I0, t0+h0, I0, t0+n+h0
11	0	3	2	N/A				
12	0	4	0	(0,0,0,1)(0,0,0,2)(0,0,1,1)(0,0,1,2)	2	n+h0	1,0,2,0	I0, t0+h0, I0, t0+n+h0
13	0	4	1	(0,0,0,0)(0,0,0,2)(0,0,1,0)(0,0,1,2)	1	h0	1,0,2,0	I0, t0+n+h0, I0, t0+h0
14	0	4	2	(0,0,0,0)(0,0,0,1)(0,0,1,0)(0,0,1,1)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+n+h0
15	0	5	0	(0,0,0,0)(0,0,0,1)(0,0,0,2)(0,0,1,1)(0,0,1,2)	2	h0	1,0,2,0, 1,0,2,0, 1,0	I0, t0+h0, I0, t0+h0, I0, t0+n+h0, I0, t0+h0, I0, t0+h0
16	0	5	1	(0,0,0,1)(0,0,0,2)(0,0,1,0)(0,0,1,1)(0,0,1,2)	2	n+h0	1,0,2,0, 1,0,2,0, 1,0	I0, t0+h0, I0, t0+h0, I0, t0+h0, I0, t0+h0, I0, t0+n+h0
17	0	5	2	(0,0,0,0)(0,0,0,1)(0,0,0,2)(0,0,1,0)(0,0,1,2)	2	h0	1,0,2,0, 1,0,2,0, 1,0	I0, t0+h0, I0, t0+h0, I0, t0+h0, I0, t0+n+h0, I0, t0+h0
18	0	6	0	(0,0,0,0)(0,0,0,1)(0,0,0,2)(0,0,1,0)(0,0,1,1)(0,0,1,2)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0
19	0	6	1	N/A				
20	1	0.5	0	(0,1,0,1)	1	n+h1	1,0	I1, t1+10n+h1
21	1	0.5	1	(0,2,0,1)	1	7n+h1	1,0	I1, t1+10n+h1
22	1	0.5	2	(0,1,1,1)	1	4n+h1	1,0	I1, t1+10n+h1
23	1	1	0	(0,0,0,1)	1	n+h1	1,0	I1, t1+4n+h1
24	1	1	1	(0,0,1,1)	1	4n+h1	1,0	I1, t1+4n+h1
25	1	2	0	(0,0,0,1)(0,0,1,1)	1	n+h1	1,0	I1, t1+n+h1
26	1	3	0	(0,0,0,1)(0,0,1,1)(1,0,0,1)	1	n+h1	1,0	I1, t1+n+h1
27	1	4	0	(0,0,0,1)(0,0,1,1)(1,0,0,1)(1,0,1,1)	1	n+h1	1,0	I1, t1+n+h1
28	1	5	0	(0,0,0,1)(0,0,1,1)(1,0,0,1)(1,0,1,1)(2,0,0,1)	1	n+h1	1,0	I1, t1+n+h1
29	1	6	0	(0,0,0,1)(0,0,1,1)(1,0,0,1)(1,0,1,1)(2,0,0,1)(2,0,1,1)	1	n+h1	1,0	I1, t1+n+h1
30	2	0.5	0	(0,1,0,1)	1'	n+h2	1',0	I2, t2+10n+h2
31	2	0.5	1	(0,2,0,1)	1'	7n+h2	1',0	I2, t2+10n+h2
32	2	0.5	2	(0,1,1,1)	1'	4n+h2	1',0	I2, t2+10n+h2
33	2	1	0	(0,0,0,1)	1'	n+h2	1',0	I2, t2+4n+h2
34	2	1	1	(0,0,1,1)	1'	4n+h2	1',0	I2, t2+4n+h2
35	2	2	0	(0,0,0,1)(0,0,1,1)	1'	n+h2	1',0	I2, t2+n+h2
36	2	3	0	(0,0,0,1)(0,0,1,1)(1,0,0,1)	1'	n+h2	1',0	I2, t2+n+h2
37	2	4	0	(0,0,0,1)(0,0,1,1)(1,0,0,1)(1,0,1,1)	1'	n+h2	1',0	I2, t2+n+h2
38	2	5	0	(0,0,0,1)(0,0,1,1)(1,0,0,1)(1,0,1,1)(2,0,0,1)	1'	n+h2	1',0	I2, t2+n+h2
39	2	6	0	(0,0,0,1)(0,0,1,1)(1,0,0,1)(1,0,1,1)(2,0,0,1)(2,0,1,1)	1'	n+h2	1',0	I2, t2+n+h2
40	3	0.5	0	(0,1,0,0)	1'	h3	1',0	I3, t3+9n+h3
41	3	0.5	1	(0,2,0,0)	1'	6n+h3	1',0	I3, t3+9n+h3
42	3	0.5	2	(0,1,1,0)	1'	3n+h3	1',0	I3, t3+9n+h3
43	3	1	0	(0,0,0,0)	1'	h3	1',0	I3, t3+3n+h3
44	3	1	1	(0,0,1,0)	1'	3n+h3	1',0	I3, t3+3n+h3
45	3	2	0	(0,0,0,0)(0,0,1,0)	1'	h3	1',0	I3, t3+h3
46	3	3	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)	1'	h3	1',0	I3, t3+h3
47	3	4	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)	1'	h3	1',0	I3, t3+h3
48	4	0.5	0	(0,1,0,*)				
49	4	0.5	1	(0,2,0,*)				
50	4	0.5	2	(0,1,1,*)				
51	4	1	0	(0,0,0,*)				
52	4	1	1	(0,0,1,*)				
53	4	2	0	(0,0,0,*)(0,0,1,*)				
54	4	3	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)				
55	4	4	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)				
56	4	5	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)(2,0,0,*)				
57	4	6	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)(2,0,0,*)(2,0,1,*)				
58	N/A	N/A	N/A	N/A				
59	N/A	N/A	N/A	N/A				
60	N/A	N/A	N/A	N/A				
61	N/A	N/A	N/A	N/A				
62	N/A	N/A	N/A	N/A				
63	N/A	N/A	N/A	N/A				

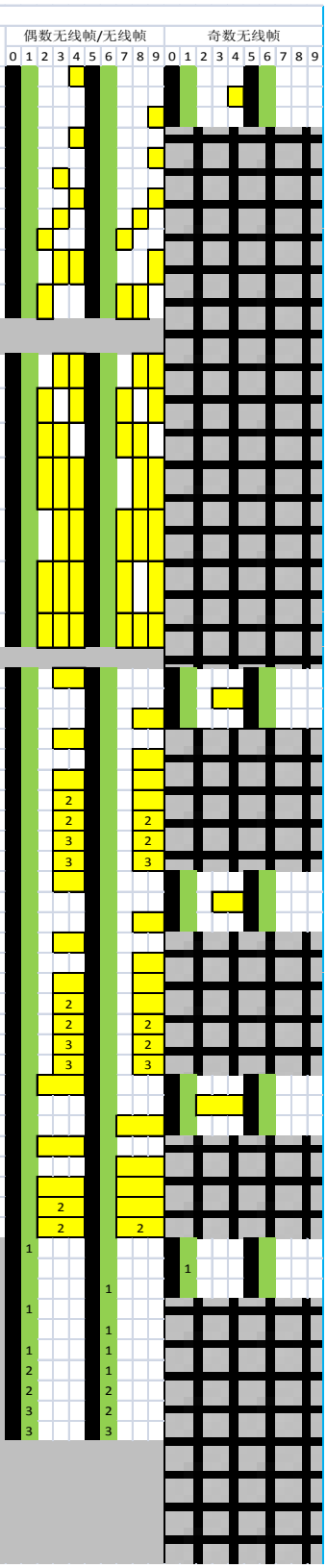


表 7: PRACH 分布与 EDMA 参数（TDD UL/DL 配置 1）

PRACH 配置	格式	密度	版本	TDD配置1 时频分布	缓存数	初始Dummy 符号数	EDMA分段	EDMA段内 符号数	偶数无线帧/无线帧 0 1 2 3 4 5 6 7 8 9	奇数无线帧 0 1 2 3 4 5 6 7 8 9
0	0	0.5	0	(0,1,0,1)	1	n+h0	1,0	l0, t0+7n+h0		
1	0	0.5	1	(0,2,0,1)	1	5n+h0	1,0	l0, t0+7n+h0		
2	0	0.5	2	(0,1,1,1)	1	3n+h0	1,0	l0, t0+7n+h0		
3	0	1	0	(0,0,0,1)	1	n+h0	1,0	l0, t0+3n+h0		
4	0	1	1	(0,0,1,1)	1	3n+h0	1,0	l0, t0+3n+h0		
5	0	1	2	(0,0,0,0)	1	h0	1,0	l0, t0+3n+h0		
6	0	2	0	(0,0,0,1)(0,0,1,1)	1	n+h0	1,0	l0, t0+n+h0		
7	0	2	1	(0,0,0,0)(0,0,1,0)	1	h0	1,0	l0, t0+n+h0		
8	0	2	2	N/A						
9	0	3	0	(0,0,0,0)(0,0,0,1)(0,0,1,1)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+n+h0, l0, t0+h0		
10	0	3	1	(0,0,0,1)(0,0,1,0) (0,0,1,1)	2	n+h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+n+h0		
11	0	3	2	(0,0,0,0) (0,0,0,1)(0,0,1,0)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+n+h0		
12	0	4	0	(0,0,0,0)(0,0,0,1)(0,0,1,0)(0,0,1,1)	2	h0	1,0,2,0	l0, t0+h0, l0, t0+h0		
13	0	4	1	N/A						
14	0	4	2	N/A						
15	0	5	0	(0,0,0,0)(0,0,0,1)(0,0,1,0)(0,0,1,1)(1,0,0,1)	2	h0	1,0,2,0	l0, t0+h0, l0, t0+h0	2	
16	0	5	1	(0,0,0,0)(0,0,0,1)(0,0,1,0)(0,0,1,1)(1,0,1,1)	2	h0	1,0,2,0	l0, t0+h0, l0, t0+h0		2
17	0	5	2	(0,0,0,0)(0,0,0,1)(0,0,1,0)(0,0,1,1)(1,0,0,0)	2	h0	1,0,2,0	l0, t0+h0, l0, t0+h0	2	
18	0	6	0	(0,0,0,0)(0,0,0,1)(0,0,1,0)(0,0,1,1)(1,0,0,1)(1,0,1,1)	2	h0	1,0,2,0	l0, t0+h0, l0, t0+h0	2	2
19	0	6	1	(0,0,0,0)(0,0,0,1)(0,0,1,0)(0,0,1,1)(1,0,0,0)(1,0,1,0)	2	h0	1,0,2,0	l0, t0+h0, l0, t0+h0	2	2
20	1	0.5	0	(0,1,0,0)	1	h1	1,0	l1, t1+6n+h1		
21	1	0.5	1	(0,2,0,0)	1	4n+h1	1,0	l1, t1+6n+h1		
22	1	0.5	2	(0,1,1,0)	1	2n+h1	1,0	l1, t1+6n+h1		
23	1	1	0	(0,0,0,0)	1	h1	1,0	l1, t1+2n+h1		
24	1	1	1	(0,0,1,0)	1	2n+h1	1,0	l1, t1+2n+h1		
25	1	2	0	(0,0,0,0)(0,0,1,0)	1	h1	1,0	l1, t1+h1		
26	1	3	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)	1	h1	1,0	l1, t1+h1		
27	1	4	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)	1	h1	1,0	l1, t1+h1		
28	1	5	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)(2,0,0,0)	1	h1	1,0	l1, t1+h1		
29	1	6	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)(2,0,0,0)(2,0,1,0)	1	h1	1,0	l1, t1+h1		
30	2	0.5	0	(0,1,0,0)	1'	h2	1',0	l2, t2+6n+h2		
31	2	0.5	1	(0,2,0,0)	1'	4n+h2	1',0	l2, t2+6n+h2		
32	2	0.5	2	(0,1,1,0)	1'	2n+h2	1',0	l2, t2+6n+h2		
33	2	1	0	(0,0,0,0)	1'	h2	1',0	l2, t2+2n+h2		
34	2	1	1	(0,0,1,0)	1'	2n+h2	1',0	l2, t2+2n+h2		
35	2	2	0	(0,0,0,0)(0,0,1,0)	1'	h2	1',0	l2, t2+h2		
36	2	3	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)	1'	h2	1',0	l2, t2+h2	2	
37	2	4	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)	1'	h2	1',0	l2, t2+h2	2	2
38	2	5	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)(2,0,0,0)	1'	h2	1',0	l2, t2+h2	3	2
39	2	6	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)(2,0,0,0)(2,0,1,0)	1'	h2	1',0	l2, t2+h2	3	3
40	3	0.5	0	N/A						
41	3	0.5	1	N/A						
42	3	0.5	2	N/A						
43	3	1	0	N/A						
44	3	1	1	N/A						
45	3	2	0	N/A						
46	3	3	0	N/A						
47	3	4	0	N/A						
48	4	0.5	0	(0,1,0,*)					1	
49	4	0.5	1	(0,2,0,*)						1
50	4	0.5	2	(0,1,1,*)						
51	4	1	0	(0,0,0,*)					1	
52	4	1	1	(0,0,1,*)						
53	4	2	0	(0,0,0,*)(0,0,1,*)					1	
54	4	3	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)					2	
55	4	4	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)					2	
56	4	5	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)(2,0,0,*)					3	
57	4	6	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)(2,0,0,*)(2,0,1,*)					3	
58	N/A	N/A	N/A	N/A						
59	N/A	N/A	N/A	N/A						
60	N/A	N/A	N/A	N/A						
61	N/A	N/A	N/A	N/A						
62	N/A	N/A	N/A	N/A						
63	N/A	N/A	N/A	N/A						

表 8: PRACH 分布与 EDMA 参数（TDD UL/DL 配置 2）

PRACH 配置	格式	密度	版本	TDD配置2 时频分布	缓存数	初始Dummy 符号数	EDMA分段	EDMA段内 符号数	偶数无线帧/无线帧 0 1 2 3 4 5 6 7 8 9	奇数无线帧 0 1 2 3 4 5 6 7 8 9
0	0	0.5	0	(0,1,0,0)	1	h0	1,0	l0, t0+3n+h0		
1	0	0.5	1	(0,2,0,0)	1	2n+h0	1,0	l0, t0+3n+h0		
2	0	0.5	2	(0,1,1,0)	1	n+h0	1,0	l0, t0+3n+h0		
3	0	1	0	(0,0,0,0)	1	h0	1,0	l0, t0+n+h0		
4	0	1	1	(0,0,1,0)	1	n+h0	1,0	l0, t0+n+h0		
5	0	1	2	N/A						
6	0	2	0	(0,0,0,0)(0,0,1,0)	1	h0	1,0	l0, t0+h0		
7	0	2	1	N/A						
8	0	2	2	N/A						
9	0	3	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)	1	h0	1,0	l0, t0+h0	2	
10	0	3	1	(0,0,0,0)(0,0,1,0) (1,0,1,0)	1	h0	1,0	l0, t0+h0		2
11	0	3	2	N/A						
12	0	4	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)	1	h0	1,0	l0, t0+h0	2	2
13	0	4	1	N/A						
14	0	4	2	N/A						
15	0	5	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)(2,0,0,0)	1	h0	1,0	l0, t0+h0	3	2
16	0	5	1	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)(2,0,1,0)	1	h0	1,0	l0, t0+h0	2	3
17	0	5	2	N/A						
18	0	6	0	(0,0,0,0)(0,0,1,0)(1,0,0,0)(1,0,1,0)(2,0,0,0)(2,0,1,0)	1	h0	1,0	l0, t0+h0	3	3
19	0	6	1	N/A						
20	1	0.5	0	N/A						
21	1	0.5	1	N/A						
22	1	0.5	2	N/A						
23	1	1	0	N/A						
24	1	1	1	N/A						
25	1	2	0	N/A						
26	1	3	0	N/A						
27	1	4	0	N/A						
28	1	5	0	N/A						
29	1	6	0	N/A						
30	2	0.5	0	N/A						
31	2	0.5	1	N/A						
32	2	0.5	2	N/A						
33	2	1	0	N/A						
34	2	1	1	N/A						
35	2	2	0	N/A						
36	2	3	0	N/A						
37	2	4	0	N/A						
38	2	5	0	N/A						
39	2	6	0	N/A						
40	3	0.5	0	N/A						
41	3	0.5	1	N/A						
42	3	0.5	2	N/A						
43	3	1	0	N/A						
44	3	1	1	N/A						
45	3	2	0	N/A						
46	3	3	0	N/A						
47	3	4	0	N/A						
48	4	0.5	0	(0,1,0,*)					1	1
49	4	0.5	1	(0,2,0,*)						
50	4	0.5	2	(0,1,1,*)					1	1
51	4	1	0	(0,0,0,*)					1	
52	4	1	1	(0,0,1,*)						1
53	4	2	0	(0,0,0,*)(0,0,1,*)					1	1
54	4	3	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)					2	1
55	4	4	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)					2	2
56	4	5	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)(2,0,0,*)					3	2
57	4	6	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)(2,0,0,*)(2,0,1,*)					3	3
58	N/A	N/A	N/A	N/A						
59	N/A	N/A	N/A	N/A						
60	N/A	N/A	N/A	N/A						
61	N/A	N/A	N/A	N/A						
62	N/A	N/A	N/A	N/A						
63	N/A	N/A	N/A	N/A						

表 9: PRACH 分布与 EDMA 参数（TDD UL/DL 配置 3）

PRACH 配置	格式	密度	版本	TDD配置3 时频分布	缓存数	初始Dummy 符号数	EDMA分段	EDMA段内 符号数	偶数无线帧/无线帧 0 1 2 3 4 5 6 7 8 9	奇数无线帧 0 1 2 3 4 5 6 7 8 9
0	0	0.5	0	(0,1,0,2)	1	2n+h0	1,0	l0, t0+5n+h0		
1	0	0.5	1	(0,2,0,2)	1	5n+h0	1,0	l0, t0+5n+h0		
2	0	0.5	2	(0,1,0,1)	1	n+h0	1,0	l0, t0+5n+h0		
3	0	1	0	(0,0,0,2)	1	2n+h0	1,0	l0, t0+2n+h0		
4	0	1	1	(0,0,0,1)	1	n+h0	1,0	l0, t0+2n+h0		
5	0	1	2	(0,0,0,0)	1	h0	1,0	l0, t0+2n+h0		
6	0	2	0	(0,0,0,1)(0,0,0,2)	2	n+h0	1,0,2,0	l0, t0+h0, l0, t0+n+h0		
7	0	2	1	(0,0,0,0)(0,0,0,2)	1	h0	1,0,1,0	l0, t0+n+h0, l0, t0+h0		
8	0	2	2	(0,0,0,0)(0,0,0,1)	2	h0	1,0,2,0	l0, t0+h0, l0, t0+n+h0		
9	0	3	0	(0,0,0,0)(0,0,0,1)(0,0,0,2)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+h0		
10	0	3	1	N/A						
11	0	3	2	N/A						
12	0	4	0	(0,0,0,0)(0,0,0,1)(0,0,0,2)(1,0,0,2)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+h0	2	
13	0	4	1	(0,0,0,0)(0,0,0,1)(0,0,0,2)(1,0,0,1)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+h0	2	
14	0	4	2	(0,0,0,0)(0,0,0,1)(0,0,0,2)(1,0,0,0)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+h0	2	
15	0	5	0	(0,0,0,0)(0,0,0,1)(0,0,0,2)(1,0,0,1)(1,0,0,2)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+h0	2 2	
16	0	5	1	(0,0,0,0)(0,0,0,1)(0,0,0,2)(1,0,0,0)(1,0,0,2)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+h0	2 2	
17	0	5	2	(0,0,0,0)(0,0,0,1)(0,0,0,2)(1,0,0,0)(1,0,0,1)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+h0	2 2	
18	0	6	0	(0,0,0,0)(0,0,0,1)(0,0,0,2)(1,0,0,0)(1,0,0,1)(1,0,0,2)	2	h0	1,0,2,0, 1,0	l0, t0+h0, l0, t0+h0, l0, t0+h0	2 2 2	
19	0	6	1	N/A						
20	1	0.5	0	(0,1,0,1)	1	n+h1	1,0	l1, t1+4n+h1		
21	1	0.5	1	(0,2,0,1)	1	4n+h1	1,0	l1, t1+4n+h1		
22	1	0.5	2	N/A						
23	1	1	0	(0,0,0,1)	1	n+h1	1,0	l1, t1+n+h1		
24	1	1	1	N/A						
25	1	2	0	(0,0,0,1)(1,0,0,1)	1	n+h1	1,0	l1, t1+n+h1	2	
26	1	3	0	(0,0,0,1)(1,0,0,1)(2,0,0,1)	1	n+h1	1,0	l1, t1+n+h1	3	
27	1	4	0	(0,0,0,1)(1,0,0,1)(2,0,0,1)(3,0,0,1)	1	n+h1	1,0	l1, t1+n+h1	4	
28	1	5	0	(0,0,0,1)(1,0,0,1)(2,0,0,1)(3,0,0,1)(4,0,0,1)	1	n+h1	1,0	l1, t1+n+h1	5	
29	1	6	0	(0,0,0,1)(1,0,0,1)(2,0,0,1)(3,0,0,1)(4,0,0,1)(5,0,0,1)	1	n+h1	1,0	l1, t1+n+h1	6	
30	2	0.5	0	(0,1,0,1)	1'	n+h2	1,0	l2, t2+4n+h2		
31	2	0.5	1	(0,2,0,1)	1'	4n+h2	1,0	l2, t2+4n+h2		
32	2	0.5	2	N/A						
33	2	1	0	(0,0,0,1)	1'	n+h2	1,0	l2, t2+n+h2		
34	2	1	1	N/A						
35	2	2	0	(0,0,0,1)(1,0,0,1)	1'	n+h2	1,0	l2, t2+n+h2	2	
36	2	3	0	(0,0,0,1)(1,0,0,1)(2,0,0,1)	1'	n+h2	1,0	l2, t2+n+h2	3	
37	2	4	0	(0,0,0,1)(1,0,0,1)(2,0,0,1)(3,0,0,1)	1'	n+h2	1,0	l2, t2+n+h2	4	
38	2	5	0	(0,0,0,1)(1,0,0,1)(2,0,0,1)(3,0,0,1)(4,0,0,1)	1'	n+h2	1,0	l2, t2+n+h2	5	
39	2	6	0	(0,0,0,1)(1,0,0,1)(2,0,0,1)(3,0,0,1)(4,0,0,1)(5,0,0,1)	1'	n+h2	1,0	l2, t2+n+h2	6	
40	3	0.5	0	(0,1,0,0)	1'	h3	1,0	l3, t3+3n+h3		
41	3	0.5	1	(0,2,0,0)	1'	3n+h3	1,0	l3, t3+3n+h3		
42	3	0.5	2	N/A						
43	3	1	0	(0,0,0,0)	1'	h3	1,0	l3, t3+h3		
44	3	1	1	N/A						
45	3	2	0	(0,0,0,0)(1,0,0,0)	1'	h3	1,0	l3, t3+h3	2	
46	3	3	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)	1'	h3	1,0	l3, t3+h3	3	
47	3	4	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)	1'	h3	1,0	l3, t3+h3	4	
48	4	0.5	0	(0,1,0,*)					1	
49	4	0.5	1	(0,2,0,*)						1
50	4	0.5	2	N/A						
51	4	1	0	(0,0,0,*)					1	
52	4	1	1	N/A						
53	4	2	0	(0,0,0,*)(1,0,0,*)					2	
54	4	3	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)					3	
55	4	4	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)(3,0,0,*)					4	
56	4	5	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)(3,0,0,*)(4,0,0,*)					5	
57	4	6	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)(3,0,0,*)(4,0,0,*)(5,0,0,*)					6	
58	N/A	N/A	N/A	N/A						
59	N/A	N/A	N/A	N/A						
60	N/A	N/A	N/A	N/A						
61	N/A	N/A	N/A	N/A						
62	N/A	N/A	N/A	N/A						
63	N/A	N/A	N/A	N/A						

表 10: PRACH 分布与 EDMA 参数（TDD UL/DL 配置 4）

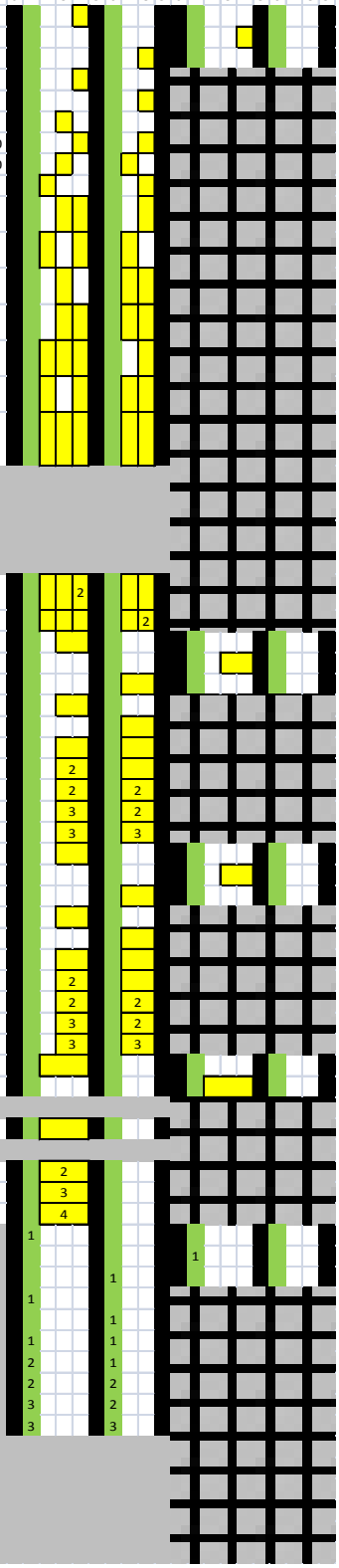
PRACH 配置	格式	密度	版本	TDD配置4 时频分布	缓存数	初始Dummy 符号数	EDMA分段	EDMA段内 符号数	偶数无线帧/无线帧 0 1 2 3 4 5 6 7 8 9	奇数无线帧 0 1 2 3 4 5 6 7 8 9
0	0	0.5	0	(0,1,0,1)	1	n+h0	1,0	I0, t0+3n+h0		
1	0	0.5	1	(0,2,0,1)	1	3n+h0	1,0	I0, t0+3n+h0		
2	0	0.5	2	(0,1,0,0)	1	h0	1,0	I0, t0+3n+h0		
3	0	1	0	(0,0,0,1)	1	n+h0	1,0	I0, t0+n+h0		
4	0	1	1	(0,0,0,0)	1	h0	1,0	I0, t0+n+h0		
5	0	1	2	N/A						
6	0	2	0	(0,0,0,0)(0,0,0,1)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0		
7	0	2	1	N/A						
8	0	2	2	N/A						
9	0	3	0	(0,0,0,0)(0,0,0,1)(1,0,0,1)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0	2	
10	0	3	1	(0,0,0,0)(0,0,0,1)(1,0,0,0)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0	2	
11	0	3	2	N/A						
12	0	4	0	(0,0,0,0)(0,0,0,1)(1,0,0,0)(1,0,0,1)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0	2 2	
13	0	4	1	N/A						
14	0	4	2	N/A						
15	0	5	0	(0,0,0,0)(0,0,0,1)(1,0,0,0)(1,0,0,1)(2,0,0,1)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0	2 3	
16	0	5	1	(0,0,0,0)(0,0,0,1)(1,0,0,0)(1,0,0,1)(2,0,0,0)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0	3 2	
17	0	5	2	N/A						
18	0	6	0	(0,0,0,0)(0,0,0,1)(1,0,0,0)(1,0,0,1)(2,0,0,0)(2,0,0,1)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0	3 3	
19	0	6	1	N/A						
20	1	0.5	0	(0,1,0,0)	1	h1	1,0	I1, t1+2n+h1		
21	1	0.5	1	(0,2,0,0)	1	2n+h1	1,0	I1, t1+2n+h1		
22	1	0.5	2	N/A						
23	1	1	0	(0,0,0,0)	1	h1	1,0	I1, t1+h1		
24	1	1	1	N/A						
25	1	2	0	(0,0,0,0)(1,0,0,0)	1	h1	1,0	I1, t1+h1	2	
26	1	3	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)	1	h1	1,0	I1, t1+h1	3	
27	1	4	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)	1	h1	1,0	I1, t1+h1	4	
28	1	5	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)(4,0,0,0)	1	h1	1,0	I1, t1+h1	5	
29	1	6	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)(4,0,0,0)(5,0,0,0)	1	h1	1,0	I1, t1+h1	6	
30	2	0.5	0	(0,1,0,0)	1'	h2	1,0	I2, t2+2n+h2		
31	2	0.5	1	(0,2,0,0)	1'	2n+h2	1,0	I2, t2+2n+h2		
32	2	0.5	2	N/A						
33	2	1	0	(0,0,0,0)	1'	h2	1,0	I2, t2+h2		
34	2	1	1	N/A						
35	2	2	0	(0,0,0,0)(1,0,0,0)	1'	h2	1,0	I2, t2+h2	2	
36	2	3	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)	1'	h2	1,0	I2, t2+h2	3	
37	2	4	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)	1'	h2	1,0	I2, t2+h2	4	
38	2	5	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)(4,0,0,0)	1'	h2	1,0	I2, t2+h2	5	
39	2	6	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)(4,0,0,0)(5,0,0,0)	1'	h2	1,0	I2, t2+h2	6	
40	3	0.5	0	N/A						
41	3	0.5	1	N/A						
42	3	0.5	2	N/A						
43	3	1	0	N/A						
44	3	1	1	N/A						
45	3	2	0	N/A						
46	3	3	0	N/A						
47	3	4	0	N/A						
48	4	0.5	0	(0,1,0,*)					1	1
49	4	0.5	1	(0,2,0,*)						
50	4	0.5	2	N/A						
51	4	1	0	(0,0,0,*)					1	
52	4	1	1	N/A						
53	4	2	0	(0,0,0,*)(1,0,0,*)					2	
54	4	3	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)					3	
55	4	4	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)(3,0,0,*)					4	
56	4	5	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)(3,0,0,*)(4,0,0,*)					5	
57	4	6	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)(3,0,0,*)(4,0,0,*)(5,0,0,*)					6	
58	N/A	N/A	N/A	N/A						
59	N/A	N/A	N/A	N/A						
60	N/A	N/A	N/A	N/A						
61	N/A	N/A	N/A	N/A						
62	N/A	N/A	N/A	N/A						
63	N/A	N/A	N/A	N/A						

表 11: PRACH 分布与 EDMA 参数（TDD UL/DL 配置 5）

PRACH 配置	格式	密度	版本	TDD配置5 时频分布	缓存数	初始Dummy 符号数	EDMA分段	EDMA段内 符号数	偶数无线帧/无线帧 0 1 2 3 4 5 6 7 8 9	奇数无线帧 0 1 2 3 4 5 6 7 8 9
0	0	0.5	0	(0,1,0,0)	1	h0	1,0	I0, t0+n+h0		
1	0	0.5	1	(0,2,0,0)	1	n+h0	1,0	I0, t0+n+h0		
2	0	0.5	2	N/A						
3	0	1	0	(0,0,0,0)	1	h0	1,0	I0, t0+h0		
4	0	1	1	N/A						
5	0	1	2	N/A						
6	0	2	0	(0,0,0,0)(1,0,0,0)	1	h0	1,0	I0, t0+h0		
7	0	2	1	N/A						
8	0	2	2	N/A						
9	0	3	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)	1	h0	1,0	I0, t0+h0		
10	0	3	1	N/A						
11	0	3	2	N/A						
12	0	4	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)	1	h0	1,0	I0, t0+h0		
13	0	4	1	N/A						
14	0	4	2	N/A						
15	0	5	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)(4,0,0,0)	1	h0	1,0	I0, t0+h0		
16	0	5	1	N/A						
17	0	5	2	N/A						
18	0	6	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)(4,0,0,0)(5,0,0,0)	1	h0	1,0	I0, t0+h0		
19	0	6	1	N/A						
20	1	0.5	0	N/A						
21	1	0.5	1	N/A						
22	1	0.5	2	N/A						
23	1	1	0	N/A						
24	1	1	1	N/A						
25	1	2	0	N/A						
26	1	3	0	N/A						
27	1	4	0	N/A						
28	1	5	0	N/A						
29	1	6	0	N/A						
30	2	0.5	0	N/A						
31	2	0.5	1	N/A						
32	2	0.5	2	N/A						
33	2	1	0	N/A						
34	2	1	1	N/A						
35	2	2	0	N/A						
36	2	3	0	N/A						
37	2	4	0	N/A						
38	2	5	0	N/A						
39	2	6	0	N/A						
40	3	0.5	0	N/A						
41	3	0.5	1	N/A						
42	3	0.5	2	N/A						
43	3	1	0	N/A						
44	3	1	1	N/A						
45	3	2	0	N/A						
46	3	3	0	N/A						
47	3	4	0	N/A						
48	4	0.5	0	(0,1,0,*)						
49	4	0.5	1	(0,2,0,*)						
50	4	0.5	2	N/A						
51	4	1	0	(0,0,0,*)						
52	4	1	1	N/A						
53	4	2	0	(0,0,0,*)(1,0,0,*)						
54	4	3	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)						
55	4	4	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)(3,0,0,*)						
56	4	5	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)(3,0,0,*)(4,0,0,*)						
57	4	6	0	(0,0,0,*)(1,0,0,*)(2,0,0,*)(3,0,0,*)(4,0,0,*)(5,0,0,*)						
58	N/A	N/A	N/A	N/A						
59	N/A	N/A	N/A	N/A						
60	N/A	N/A	N/A	N/A						
61	N/A	N/A	N/A	N/A						
62	N/A	N/A	N/A	N/A						
63	N/A	N/A	N/A	N/A						

表 12: PRACH 分布与 EDMA 参数（TDD UL/DL 配置 6）

PRACH 配置	格式	密度	版本	TDD配置6 时频分布	缓存数	初始Dummy 符号数	EDMA分段	EDMA段内 符号数
0	0	0.5	0	(0,1,0,2)	1	2n+h0	1,0	I0, t0+9n+h0
1	0	0.5	1	(0,2,0,2)	1	7n+h0	1,0	I0, t0+9n+h0
2	0	0.5	2	(0,1,1,1)	1	4n+h0	1,0	I0, t0+9n+h0
3	0	1	0	(0,0,0,2)	1	2n+h0	1,0	I0, t0+4n+h0
4	0	1	1	(0,0,1,1)	1	4n+h0	1,0	I0, t0+4n+h0
5	0	1	2	(0,0,0,1)	1	n+h0	1,0	I0, t0+4n+h0
6	0	2	0	(0,0,0,2)(0,0,1,1)	1	2n+h0	1,0,1,0	I0, t0+n+h0, I0, t0+2n+h0
7	0	2	1	(0,0,0,1)(0,0,1,0)	1	n+h0	1,0,1,0	I0, t0+n+h0, I0, t0+2n+h0
8	0	2	2	(0,0,0,0)(0,0,1,1)	1	h0	1,0,1,0	I0, t0+3n+h0, I0, t0+h0
9	0	3	0	(0,0,0,1)(0,0,0,2)(0,0,1,1)	2	n+h0	1,0,2,0, 1,0	I0, t0+h0, I0, t0+n+h0, I0, t0+n+h0
10	0	3	1	(0,0,0,0)(0,0,0,2)(0,0,1,0)	1	h0	1,0,1,0, 1,0	I0, t0+n+h0, I0, t0+h0, I0, t0+n+h0
11	0	3	2	(0,0,0,1)(0,0,1,0)(0,0,1,1)	2	n+h0	1,0,2,0, 1,0	I0, t0+n+h0, I0, t0+h0, I0, t0+n+h0
12	0	4	0	(0,0,0,1)(0,0,0,2)(0,0,1,0)(0,0,1,1)	2	n+h0	1,0,2,0, 1,0,2,0	I0, t0+h0, I0, t0+h0, I0, t0+h0, I0, t0+n+h0
13	0	4	1	(0,0,0,0)(0,0,0,1)(0,0,0,2)(0,0,1,1)	2	h0	1,0,2,0, 1,0,2,0	I0, t0+h0, I0, t0+h0, I0, t0+n+h0, I0, t0+h0
14	0	4	2	(0,0,0,0)(0,0,0,2)(0,0,1,0)(0,0,1,1)	2	h0	1,0,2,0, 1,0,2,0	I0, t0+n+h0, I0, t0+h0, I0, t0+h0, I0, t0+h0
15	0	5	0	(0,0,0,0)(0,0,0,1)(0,0,0,2)(0,0,1,0)(0,0,1,1)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0
16	0	5	1	N/A				
17	0	5	2	N/A				
18	0	6	0	(0,0,0,0)(0,0,0,1)(0,0,0,2)(0,0,1,0)(0,0,1,1)(1,0,0,2)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0
19	0	6	1	(0,0,0,0)(0,0,0,1)(0,0,0,2)(0,0,1,0)(0,0,1,1)(1,0,1,1)	2	h0	1,0,2,0	I0, t0+h0, I0, t0+h0
20	1	0.5	0	(0,1,0,1)	1	n+h1	1,0	I1, t1+8n+h1
21	1	0.5	1	(0,2,0,1)	1	6n+h1	1,0	I1, t1+8n+h1
22	1	0.5	2	(0,1,1,0)	1	3n+h1	1,0	I1, t1+8n+h1
23	1	1	0	(0,0,0,1)	1	n+h1	1,0	I1, t1+3n+h1
24	1	1	1	(0,0,1,0)	1	3n+h1	1,0	I1, t1+3n+h1
25	1	2	0	(0,0,0,1)(0,0,1,0)	1	n+h1	1,0,1,0	I1, t1+h1, I1, t1+n+h1
26	1	3	0	(0,0,0,1)(0,0,1,0)(1,0,0,1)	1	n+h1	1,0,1,0	I1, t1+h1, I1, t1+n+h1
27	1	4	0	(0,0,0,1)(0,0,1,0)(1,0,0,1)(1,0,1,0)	1	n+h1	1,0,1,0	I1, t1+h1, I1, t1+n+h1
28	1	5	0	(0,0,0,1)(0,0,1,0)(1,0,0,1)(1,0,1,0)(2,0,0,1)	1	n+h1	1,0,1,0	I1, t1+h1, I1, t1+n+h1
29	1	6	0	(0,0,0,1)(0,0,1,0)(1,0,0,1)(1,0,1,0)(2,0,0,1)(2,0,1,0)	1	n+h1	1,0,1,0	I1, t1+h1, I1, t1+n+h1
30	2	0.5	0	(0,1,0,1)	1'	n+h2	1',0	I2, t2+8n+h2
31	2	0.5	1	(0,2,0,1)	1'	6n+h2	1',0	I2, t2+8n+h2
32	2	0.5	2	(0,1,1,0)	1'	3n+h2	1',0	I2, t2+8n+h2
33	2	1	0	(0,0,0,1)	1'	n+h2	1',0	I2, t2+3n+h2
34	2	1	1	(0,0,1,0)	1'	3n+h2	1',0	I2, t2+3n+h2
35	2	2	0	(0,0,0,1)(0,0,1,0)	1'	n+h2	1',0,1',0	I2, t2+h2, I2, t2+n+h2
36	2	3	0	(0,0,0,1)(0,0,1,0)(1,0,0,1)	1'	n+h2	1',0,1',0	I2, t2+h2, I2, t2+n+h2
37	2	4	0	(0,0,0,1)(0,0,1,0)(1,0,0,1)(1,0,1,0)	1'	n+h2	1',0,1',0	I2, t2+h2, I2, t2+n+h2
38	2	5	0	(0,0,0,1)(0,0,1,0)(1,0,0,1)(1,0,1,0)(2,0,0,1)	1'	n+h2	1',0,1',0	I2, t2+h2, I2, t2+n+h2
39	2	6	0	(0,0,0,1)(0,0,1,0)(1,0,0,1)(1,0,1,0)(2,0,0,1)(2,0,1,0)	1'	n+h2	1',0,1',0	I2, t2+h2, I2, t2+n+h2
40	3	0.5	0	(0,1,0,0)	1'	h3	1',0	I3, t3+7n+h3
41	3	0.5	1	(0,2,0,0)	1'	5n+h3	1',0	I3, t3+7n+h3
42	3	0.5	2	N/A				
43	3	1	0	(0,0,0,0)	1'	h3	1',0	I3, t3+2n+h3
44	3	1	1	N/A				
45	3	2	0	(0,0,0,0)(1,0,0,0)	1'	h3	1',0	I3, t3+2n+h3
46	3	3	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)	1'	h3	1',0	I3, t3+2n+h3
47	3	4	0	(0,0,0,0)(1,0,0,0)(2,0,0,0)(3,0,0,0)	1'	h3	1',0	I3, t3+2n+h3
48	4	0.5	0	(0,1,0,*)				
49	4	0.5	1	(0,2,0,*)				
50	4	0.5	2	(0,1,1,*)				
51	4	1	0	(0,0,0,*)				
52	4	1	1	(0,0,1,*)				
53	4	2	0	(0,0,0,*)(0,0,1,*)				
54	4	3	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)				
55	4	4	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)				
56	4	5	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)(2,0,0,*)				
57	4	6	0	(0,0,0,*)(0,0,1,*)(1,0,0,*)(1,0,1,*)(2,0,0,*)(2,0,1,*)				
58	N/A	N/A	N/A	N/A				
59	N/A	N/A	N/A	N/A				
60	N/A	N/A	N/A	N/A				
61	N/A	N/A	N/A	N/A				
62	N/A	N/A	N/A	N/A				
63	N/A	N/A	N/A	N/A				



重要声明

德州仪器(TI) 及其下属子公司有权根据 JESD46 最新标准, 对所提供的产品和服务进行更正、修改、增强、改进或其它更改， 并有权根据 JESD48 最新标准中止提供任何产品和服务。客户在下订单前应获取最新的相关信息, 并验证这些信息是否完整且是最新的。所有产品的销售都遵循在订单确认时所提供的TI 销售条款与条件。

TI 保证其所销售的组件的性能符合产品销售时 TI 半导体产品销售条件与条款的适用规范。仅在 TI 保证的范围内，且 TI 认为 有必要时才会使用测试或其它质量控制技术。除非适用法律做出了硬性规定，否则没有必要对每种组件的所有参数进行测试。

TI 对应用帮助或客户产品设计不承担任何义务。客户应对其使用 TI 组件的产品和应用自行负责。为尽量减小与客户产品和应 用相关的风险，客户应提供充分的设计与操作安全措施。

TI 不对任何 TI 专利权、版权、屏蔽作品权或其它与使用了 TI 组件或服务的组合设备、机器或流程相关的 TI 知识产权中授予 的直接或隐含权限作出任何保证或解释。TI 所发布的与第三方产品或服务有关的信息，不能构成从 TI 获得使用这些产品或服 务的许可、授权、或认可。使用此类信息可能需要获得第三方的专利权或其它知识产权方面的许可，或是 TI 的专利权或其它 知识产权方面的许可。

对于 TI 的产品手册或数据表中 TI 信息的重要部分，仅在没有对内容进行任何篡改且带有相关授权、条件、限制和声明的情况 下才允许进行复制。TI 对此类篡改过的文件不承担任何责任或义务。复制第三方的信息可能需要服从额外的限制条件。

在转售 TI 组件或服务时，如果对该组件或服务参数的陈述与 TI 标明的参数相比存在差异或虚假成分，则会失去相关 TI 组件 或服务的所有明示或暗示授权，且这是不正当的、欺诈性商业行为。TI 对任何此类虚假陈述均不承担任何责任或义务。

客户认可并同意，尽管任何应用相关信息或支持仍可能由 TI 提供，但他们将独力负责满足与其产品及在其应用中使用 TI 产品 相关的所有法律、法规和安全相关要求。客户声明并同意，他们具备制定与实施安全措施所需的全部专业技术和知识，可预见 故障的危险后果、监测故障及其后果、降低有可能造成人身伤害的故障的发生机率并采取适当的补救措施。客户将全额赔偿因 在此类安全关键应用中使用任何 TI 组件而对 TI 及其代理造成的任何损失。

在某些场合中，为了推进安全相关应用有可能对 TI 组件进行特别的促销。TI 的目标是利用此类组件帮助客户设计和创立其特 有的可满足适用的功能安全性标准和要求的终端产品解决方案。尽管如此，此类组件仍然服从这些条款。

TI 组件未获得用于 FDA Class III（或类似的生命攸关医疗设备）的授权许可，除非各方授权官员已经达成了专门管控此类使 用的特别协议。

只有那些 TI 特别注明属于军用等级或“增强型塑料”的 TI 组件才是设计或专门用于军事/航空应用或环境的。购买者认可并同 意，对并非指定面向军事或航空航天用途的 TI 组件进行军事或航空航天方面的应用，其风险由客户单独承担，并且由客户独 力负责满足与此类使用相关的所有法律和法规要求。

TI 已明确指定符合 ISO/TS16949 要求的产品，这些产品主要用于汽车。在任何情况下，因使用非指定产品而无法达到 ISO/TS16949 要求，TI不承担任何责任。

	产品		应用
数字音频	www.ti.com.cn/audio	通信与电信	www.ti.com.cn/telecom
放大器和线性器件	www.ti.com.cn/amplifiers	计算机及周边	www.ti.com.cn/computer
数据转换器	www.ti.com.cn/dataconverters	消费电子	www.ti.com/consumer-apps
DLP® 产品	www.dlp.com	能源	www.ti.com/energy
DSP - 数字信号处理器	www.ti.com.cn/dsp	工业应用	www.ti.com.cn/industrial
时钟和计时器	www.ti.com.cn/clockandtimers	医疗电子	www.ti.com.cn/medical
接口	www.ti.com.cn/interface	安防应用	www.ti.com.cn/security
逻辑	www.ti.com.cn/logic	汽车电子	www.ti.com.cn/automotive
电源管理	www.ti.com.cn/power	视频和影像	www.ti.com.cn/video
微控制器 (MCU)	www.ti.com.cn/microcontrollers		
RFID 系统	www.ti.com.cn/rfidsys		
OMAP应用处理器	www.ti.com/omap		
无线连通性	www.ti.com.cn/wirelessconnectivity	德州仪器在线技术支持社区	www.deyisupport.com

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