Application Note
ZHCACU1 – 6.2023

TI RF Transceiver EVM 自动化环境搭建方法

Jason Ren

摘要

TI 的射频收发信机 Transceiver 产品族种类丰富，有以 AFE77xx 为代表的零中频架构的收发信机，和 AFE76xx, AFE79xx, AFE80xx 为代表的射频直采架构的收发信机。TI Transceiver 以其高集成度，高性能，高灵活度等优势广泛应用在 MIMO， RRU，Small Cell 等用户产品中。Transceiver 是非常复杂的器件，用户在应用前期会进行详细的系统层面评估和实际的用户验证板/TI EVM 板测试。随着器件功能越来越复杂，用户需要与 EVM 的交互项动作越来越多。同时，随着 Transceiver 的通道集成度越来越高，用户需要测试 EVM 花费的时间越来越多。为了方便用户测试，提升用户测试效率，TI EVM 开放了更多的软件接口给用户。本文会以 AFE7920 为例，介绍 TI Transceiver EVM 基础硬件环境及软件环境搭建方法， GUI 使用方法，以及如何搭建以 GUI 为 host，自动化控制外接测试仪表，TI EVM 以及和用户 matlab 中算法交互的方法。该方法也可以作为例子应用在 TI 其它种类的 RF Transceiver 评估测试过程中去。

目录

图/表

1 引言

TI Transceiver 产品族包括 2T2R, 4T4R 和 8T8R 的通道数目，高通道集成度给用户系统带来了成本，面积和热设计的独特优势。其中，AFE7920 是集成 4T4R2F 通道的射频直采架构 transceiver，广泛应用在 MIMO，RRU 和 Small Cell 产品中。针对用户评估需求，TI EVM 开发了基于 python 的 GUI 软件，支持 python script 操作，raw 寄存器操作的同时，也支持与外设仪器交互通信，matlab 侧控制。在用户进行详细评估时，搭配使用基于 pyVisa 库的接口进行自动化测试会极大程度提升用户测试 EVM 射频指标效率。搭配 matlab 侧控制会方便用户在评估前期引入用户自研 DPD 算法进行更详细的评估。

下面对基础的 TI Transceiver 的 EVM 环境搭建进行介绍，在此基础上会介绍搭建自动化测试环境及交互 matlab 侧软件的方法。

2 AFE7920 EVM 基础环境搭建

AFE7920 的 EVM 基础环境主要包括两部分：硬件部分和软件部分。需要用户按照下面流程从硬件部分->软件部分顺序搭建。

2.1 硬件部分环境搭建

硬件部分环境如图 1 所示，板子包括 capture card(TSW14J5x)和 AFE card(AFE7920 EVM)，外部连接包括电源，时钟，USB 接口。

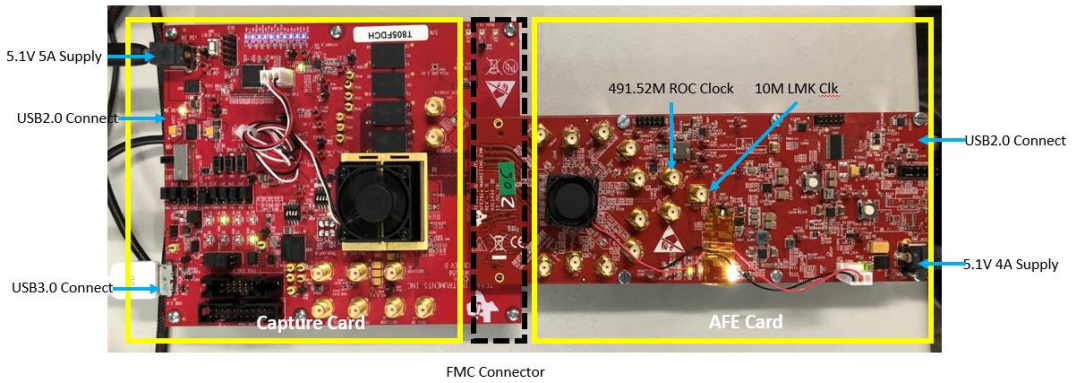


Figure 1. AFE7920 EVM 硬件连接框图

结合上图针对硬件环境搭建进行分步说明：

在所有单板下电的条件下：

1. 将 capture card 通过 FMC 接头连接到 AFE card 上，注意按紧，否则在后续的步骤中出现建链失败问题。
2. 在外供电源下电的情况下，将 5.1V 5A 限流的电源连接到 capture card 上。
3. 将 USB3.0 和 USB2.0 接口连接 capture card 到测试机上。需注意，不要用 USB HUB，HUB 会导致 PC 识别不到 EVM 板。

4. 在外供电源下电的情况下，将 5.1V 4A 限流的电源连接到 AFE card 上。
5. 将 USB2.0 接口连接 AFE card 到测试机上。
6. 将 10M 外供参考钟连接到图中的 J14 SMA connector 上。该参考钟可由信号源输出直供，或者用仪表的 ref_output（接口一般在仪表背后）输出直供。

上述步骤执行完毕后，完成了基础硬件环境搭建，在后面软件环境准备好开始 bring up EVM 时，再进行上电。

2.2 软件部分环境搭建

软件环境主要包括基础部分的 EVM 软件 Latte 和 HSDC 软件部署，扩展部分的控制库部署以及部分 matlab 接口配置。

2.2.1 基础 EVM 软件安装

- Latte 软件 AFE79xxEVM_GUI.exe. 用于控制 AFE card 的软件，集成了基础的 python 环境，多余的扩展库需要自行部署。
- HSDC 是用于控制 Capture card 的软件，可实现发数和采数功能，可受控于 Latte。部分其它的 AFE EVM（比如 AFE80xx）可以直接由 Latte 实现发数和采数功能，就不需要这个软件。

  上述两个软件可向对应的技术支持索取。软件安装过程非常简单，在此不再赘述。

2.2.2 扩展库部署(Optional)

如果用户只需要对 EVM 环境进行简单的手动测试，此部分的软件是非必要的。如果需要进行自动化测试、用户算法联调，则需要以下扩展库。

Latte 扩展库分为三大部分功能：用于控制外接测试仪表，用于交互 HSDC 软件采数和 matlab 控制。

外接仪表功能库：基础 Latte 环境已经集成了 pyVisa 库，进行仪器仪表控制。

交互 HSDC 软件采数：该 py 文件包括两部分，基础库函数定义(mhsdcparam.py)和示例控制代码(hsdcpro_cntrl.py)。放在测试机内的 C:\Users\~\Documents\Texas Instruments\Latte\lib\Afe79xxLibraries 文件夹内由 latte 进行引用。路径中"~"为系统 user id。

用户可参考示例控制代码调用基础库函数，来实现主控函数。基础库函数不需要用户进行修改，这里简介几个关键变量和控制函数，方便用户进行参考：

- hsdcparam.Boardsno:capture 单板 ID
- hsdcparam.tx.Devicename：FPGA 下载的 TX Link 配置 ini 文件
- hsdcparam.tx.Datarate：TX 数据速率

- hsdcparam.rx.Devicename：FPGA 下载的 RX Link 配置 ini 文件
- hsdcparam.tx.DACFilePath: 发射用户自定义数据的文件路径
- hsdcparam.rx.Datarate：RX 数据速率
- hsdcparam.rx.SaveCSVFilePathWithName:向用户自定义文件路径保存 RX ADC 采数数据
- hsdcparam.fb.Devicename：FPGA 下载的 FB Link 配置 ini 文件
- hsdcparam.fb.Datarate：FB 数据速率
- hsdcparam.fb.SaveCSVFilePathWithName：向用户自定义文件路径保存 FB ADC 采数数据
- confighsdcpro().fb_cs():进行反馈采数，并将采数 raw data 存储到用户指定路径中(hsdcparam.fb.SaveCSVFilePathWithName)
- confighsdcpro().rx_cs():进行接收采数，并将采数 raw data 存储到用户指定路径中(hsdcparam.rx.SaveCSVFilePathWithName)
- confighsdcpro().tx_ls()：读取用户指定路径（hsdcparam.tx.DACFilePath）的发射数据，并由 FPGA 发送给 AFE 芯片

交互 matlab 控制：该.m 文件包括两部分，基础交互 socket 函数(socket_binary.m)定义和示例控制代码(main_script.m)。放在测试机内的 matlab 工程文件路径中。

这三部分可向对应的 TI 技术支持索取。

2.3 EVM 基础初始化

本节介绍 EVM 初始化流程，器件参数配置方法，动态修改器件参数流程。

2.3.1 初始化流程

初始化流程中需要操作两个 GUI: HSDC PRO 和 Latte。按如下步骤执行：

1. 两块单板上电，确保 USB 和时钟源连接正确，工作正常。
2. 打开 HSDC 软件，在弹出的对话框内点击 ok，确认连接的 capture card 型号正确。

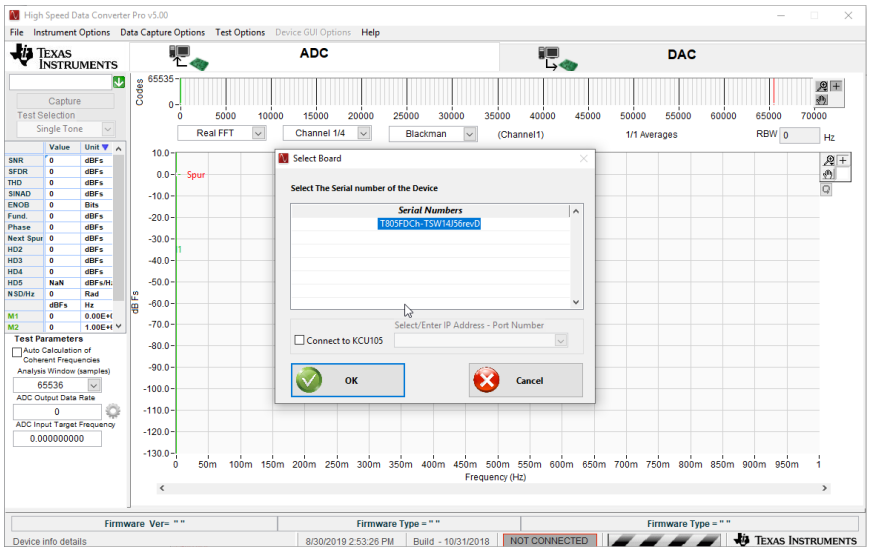


Figure 2. HSDC 软件 GUI

3. 打开 latte 软件，执行 AFE79xx > Automation > AFE79xx_TSW14J56_Mode1.py 文件，自动初始化并建链器件。

2.3.2 器件参数配置方法

器件参数功能分为两部分：静态配置部分和动态配置部分。静态部分是在 EVM 环境初始化前进行的配置，从器件维度分为两部分：(1) AFE 侧的静态配置，以 bring up 文档的形式存在，可由用户进行维护，用来配置 AFE 器件的射频参数，数字参数，JESD/SERDES 参数等。(2)FPGA 侧的静态配置，是以 ini 文件形式存在，不需要由用户进行维护，主要配置 JESD/SERDES 相关参数。

TI 提供了这两个软件的交互功能，这两个软件交互的过程是由 Latte 作为 host 进行控制，该 host 是 2.3.1 节中提到的 AFE79xx_TSW14J56_Modex.py。它会控制 Latte 进行 AFE 器件静态配置和动态控制，也会控制 HSDC 对 FPGA 进行初始化和动态采数/发数。

针对 Latte 内静态配置，可由用户静态修改 bring up 文件，再由 host 脚本对控制 Latte 执行 bring up 文件。Host 脚本访问 AFE 器件静态配置的命令为：

mainWindow.runFile(r"文件路径\AFE79xx_TSW14J56_Modex.py")。

上面提到了访问 AFE 静态配置文件，该静态文件可由用户进行维护，配置 AFE 工作参数，关键参数解释如下：

- sysParams.FRef：AFE 工作参考钟
- sysParams.FadcRx：RX ADC 采样率
- sysParams.FadcFb ：FB ADC 采样率
- sysParams.Fdac： TX DAC 采样率

- sysParams.rxNco0： RX NCO0 频点，长度为 4 的数组分别对应每个通道的 NCO 频点
- sysParams.ddcFactorRx： RX 抽值 DDC 系数，长度为 4 的数组分别对应每个通道系数
- sysParams.fbNco0： FB NCO0 频点，长度为 2 的数组分别对应每个通道的 NCO 频点
- sysParams.ddcFactorFb： FB 抽值 DDC 系数，长度为 2 的数组分别对应每个通道系数
- sysParams.txNco0： TX NCO0 频点，长度为 4 的数组分别对应每个通道的 NCO 频点
- sysParams.ducFactorTx： TX 插值 DUC 系数，长度为 4 的数组分别对应每个通道系数
- sysParams.LMFSHdRx：RX 的 JESD 链路 LMFS 组帧格式
- sysParams.LMFSHdFb：FB 的 JESD 链路 LMFS 组帧格式
- sysParams.LMFSHdTx：TX 的 JESD 链路 LMFS 组帧格式

2.3.3 动态参数配置方法

器件初始化后，可在线进行 AFE 器件的动态控制，在部分器件版本的 GUI 中，可由 device.gui.show()呼出器件控制图形化 GUI。更全面地，我们可以在 Latte 地 Command Line 中使用 python script 进行器件控制，控制 API 指令集在 C:\User\Documents\Texas Instruments\Latte\lib\Afe79xxLibraries\AFE79xxLibraryPG1p0\resourceFiles\docs 文件夹中可以找到。这里只简单举几个常用脚本进行解释说明：

AFE.TOP.overrideTdd(TX,FB,RX)，射频通道开关控制，按 argument 的 bit 进行控制

AFE.DSA.setRxDsa(CH,Value)，设置 RX 通道 DSA，CH-通道号，Value-DSA 值

AFE.DSA.setTxDsa(CH,Value) ，设置 TX 通道 DSA，CH-通道号，Value-DSA 值

AFE.DSA.setFbDsa(CH,Value) ，设置 FB 通道 DSA，CH-通道号，Value-DSA 值

AFE.updateTxNco(CH, mixerFreq, BandNo, NcoNo) ：设置 TX NCO 频点，CH-通道号，MixerFreq-配置频点，BandNo-Band 号，单 band 应用选择 0，NcoNo:Nco number，常规应用选 0

AFE.updateRxNco(CH, mixerFreq, BandNo, NcoNo) ：设置 RX NCO 频点，CH-通道号，MixerFreq-配置频点，BandNo-Band 号，单 band 应用选择 0，NcoNo:Nco number，常规应用选 0

AFE.updateFbNco(CH, mixerFreq, BandNo, NcoNo) ：设置 Fb NCO 频点，CH-通道号，MixerFreq-配置频点，BandNo-Band 号，单 band 应用选择 0，NcoNo:Nco number，常规应用选 0

3 AFE7920 EVM 自动化测试环境搭建

基于上一章所讲，基础的 EVM 软硬件环境已经搭建完毕，EVM 已经初始化并可简单控制，本章在上一章的基础上，讲解如何搭建自动化测试环境。

自动化测试环境主要思路是利用 Latte 作为 host，控制 EVM 做 AFE 的射频控制，控制外部频谱仪做 EVM 射频指标的测试和读取，以及控制信号源输出，并输出测试报告到测试机中，连接框图如图 3 所示。

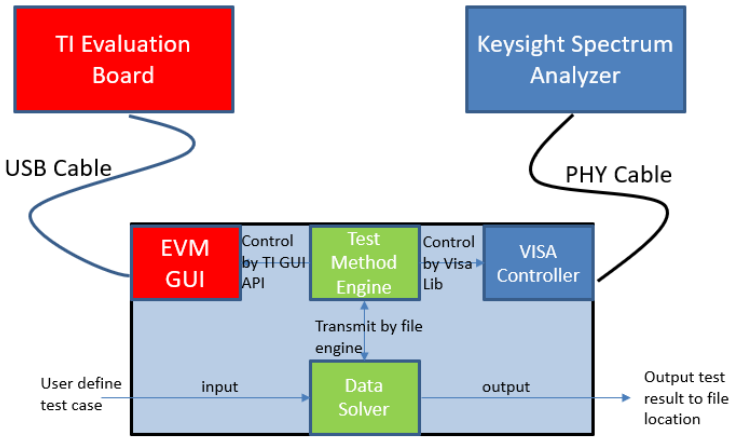


Figure 3.　自动化测试环境框图

控制 EVM 是由图中的 EVM GUI 实现，利用 Latte 的动态脚本对 AFE 器件的动作进行控制，可利用 2.3.3 节中提到的相关命令。在 host 函数中，可直接调用 2.3.3 节中提到的相关脚本进行控制。

控制外部仪器是由图中的 VISA Controller 实现，以频谱仪为例，频谱仪与测试机用网线连接通信，利用 Latte 引入的 pyVisa 库，通过 SCPI 控制频谱仪进行读数。这里需要注意，不同的仪器仪表厂家的 SCPI 命令有一定的差异，用户需要参考测试环境中实际使用的仪器仪表的程控手册进行编程，我们的脚本只是一个例子。

输出测试报告是由图中的 Data Solver 实现，Data Solver 将从频谱仪中读取到的测试结果以及测试机本地的采数功率统计结果，输出到指定路径 excel 文件中。

下行和上行自动化测试软件流程图如下图所示：

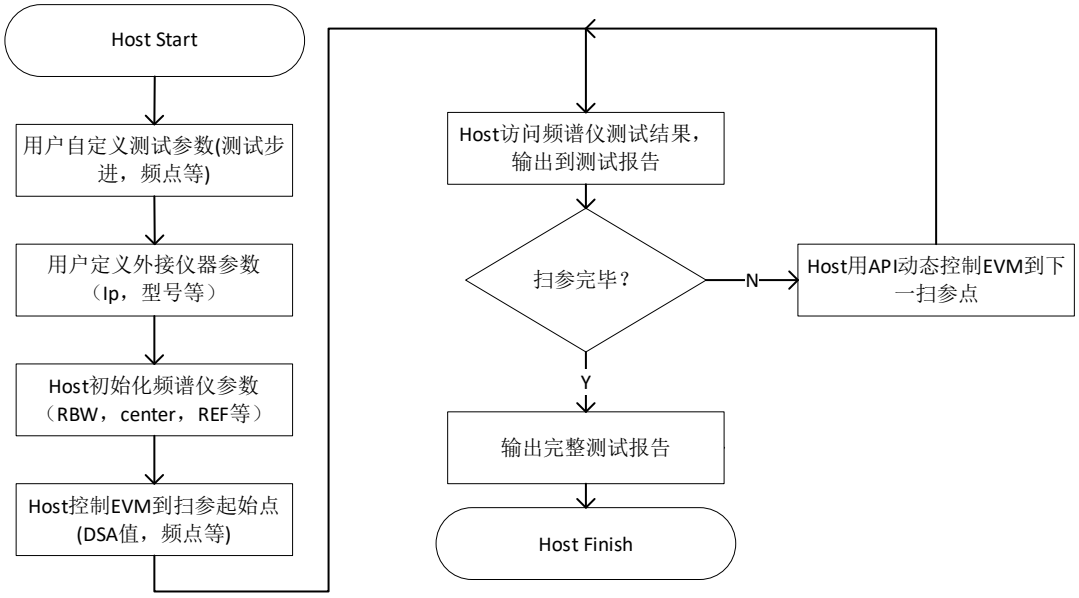


Figure 4.　下行平坦度/DSA 精度自动化测试 Host 软件流程

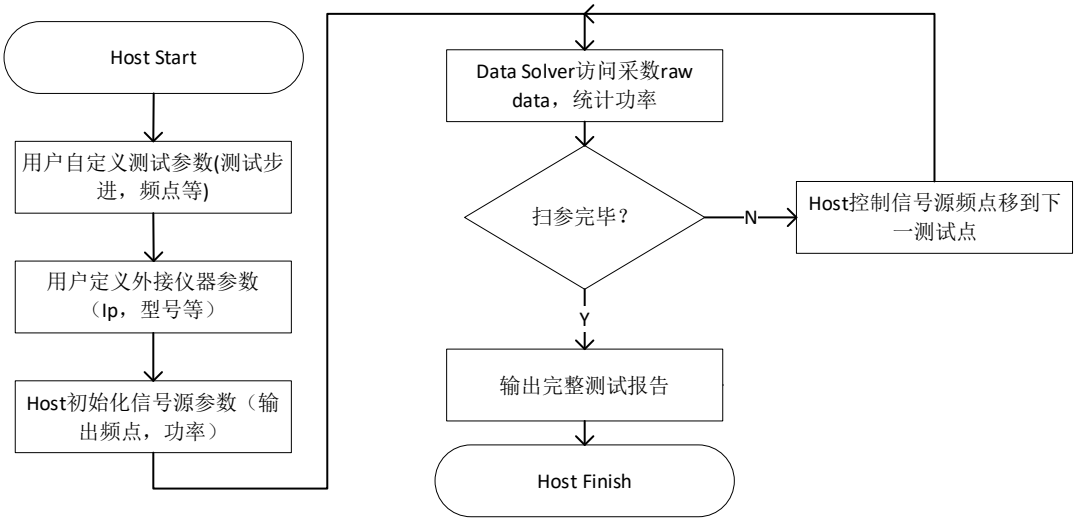


Figure 5.　上行平坦度自动化测试 Host 软件流程

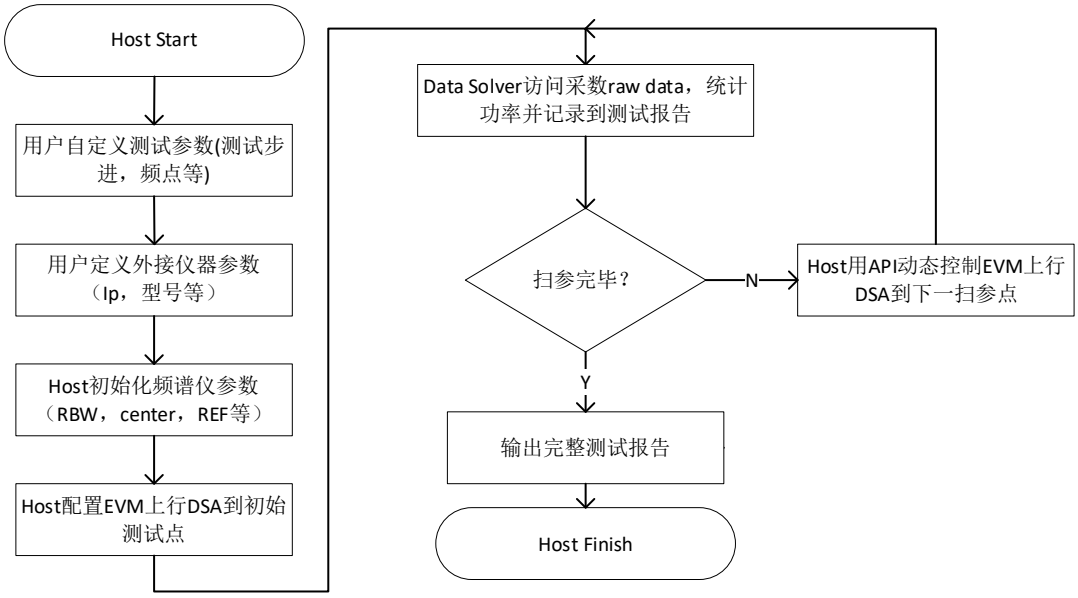


Figure 6. 上行 DSA 精度自动化测试 Host 软件流程

该部分完整 python 代码可向技术支持索取。

4 AFE7920 EVM 和 matlab 联调功能实现

AFE 器件在基站射频应用中，往往涉及了较为复杂的 DPD 算法，这部分算法一般由用户实现。在前期基于 EVM 的测试过程中，难以集成到 capture card 的 FPGA 上。因此 TI transceiver EVM 提供了 matlab 与 Latte 交互选项。

如下图所示，为 matlab 与 Latte 联调时的软件框图。

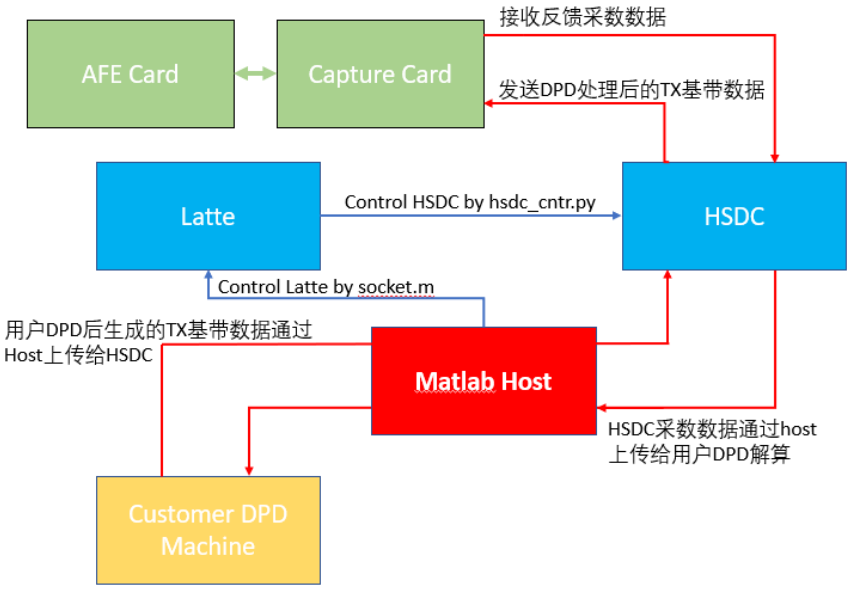


Figure 7. Matlab 和 Latte 软件联调框图

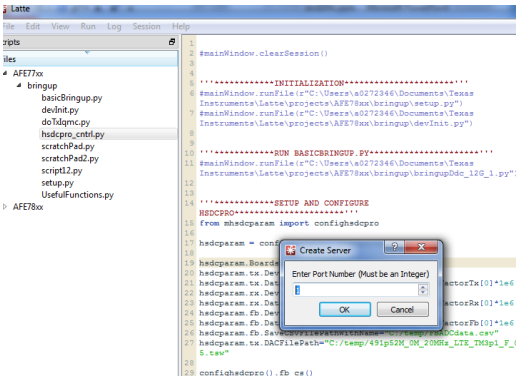
上图中，核心思想是利用 Matlab Main 函数作为主控函数，分别控制 TI 侧和用户侧。

TI 侧主要思想是利用 matlab 控制 Latte，进一步控制 HSDC。从发射链路来看，matlab host 获取用户 DPD 解算后需要发送到下行的基带数据，利用 socket 函数通知 Latte 来控制 HSDC 从测试机中下载该基带数据，并上传到 capture card 中；从反馈链路来看，matlab host 通过 socket 函数通知 Latte 来控制 HSDC 进行反馈采数，并将该数据包存储到测试机中，再通知用户 DPD 模块读取该数据并进行解算。

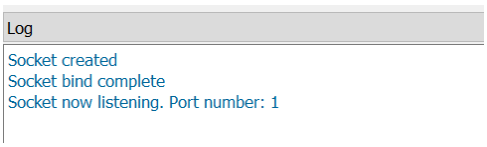
其中，利用 Latte 控制 HSDC 的 python 脚本在 2.3.2 中已经详细描述，在此不进行赘述。针对 matlab host 对 Latte 的控制，进行以下重点应用声明：

在配置前期，需要指定 latte 通信端口，如下图所示例子：

1. 在 latte 中，点击 RUN>Create Server，将 port 端口配置为 1



2. 在 latte 的 log 窗口确认 server 端口创建成功



3. 确认 main 函数中定义的 port 接口为 1

配置好通信端口后，基于图 5 介绍的软件结构进行 main 函数编写，这里介绍关键函数：

socket_binary(skt,'mainWindow.TimeOut=600'); 定义 server 超时时间，保持默认 600

socket_binary(skt,'mainWindow.clearSession()');初始化 Latte 软件参数

socket_binary(skt,['mainWindow.runFile(r"',fpath_setup,'")']);运行指定路径的.py 文件

用户侧主要是利用 host 与 DPD 模块进行交互，由用户维护代码，在此不再赘述。

5 参考文献

AFE79xx 器件手册：AFE7920_88_89

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出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