

YunSDR Y3x0

FPGA 开发指南

Rev. 1.4



威视锐旗下品牌:



修订记录

| 版本 | 修订日期 | 修订内容 |
|-----|-----------------|-------------------------|
| 1.0 | 2014 年 11 月 1 日 | 硬件用户手册初始版本 |
| 1.1 | 2015 年 1 月 10 日 | 更新模板与参考例程说明 |
| 1.2 | 2015 年 2 月 2 日 | 更新 Y310 模块说明 |
| 1.3 | 2015 年 5 月 1 日 | 更新 Y310 射频系统框图 |
| 1.4 | 2015 年 9 月 11 日 | 更新 Y310 26MHz 参考计算频点的代码 |



关于威视锐科技

北京威视锐科技有限公司是专注于软件定义无线通信（SDR）系统仿真、验证和测试平台的研发与生产，同时也提供通用高性能信号处理板卡，应用于机器视觉、生命科学和高能物理等科学计算领域。

威视锐与微软研究院联合开发的 Sora 软件无线电平台已经成为世界上知名大学和科研结构开展无线通信研究的首选平台，也是学术研究领域全球唯一的 100% 基于 x86 的宽带实时软件无线电平台，目前已经有超过 20 多个国家的 300 多个科研用户。作为全球最大的可编程器件（FPGA）公司 XILINX 的全球合作伙伴，威视锐科技提供基于 XILINX FPGA/SoC 的全方位解决方案。特别是 ZYNQ 7000 系列 SoC 产品，威视锐携手 XILINX 发布了全球第一款基于 ZYNQ SoC 的低成本开源 SoC 模块 SNOWLeo，性价比远高于国外同类产品，大大降低了 SoC 系统的开发门槛。此外，威视锐旗下的红色飓风开发板自从 2004 年发售以来已成为国内销售时间最长、知名度最高以及用户量最多的开发板产品。

多年以来，威视锐科技坚持 “Innovation for Research” 的发展理念，与国内众多知名高校建立合作关系，帮助专家、学者和研发工程师将创新的理念变成现实。对于产业界客户，威视锐提供严格验证的核心模块、智能便携的测量仪器以及定制化的设计服务来加快产品研发周期。

目 录

| | |
|---------------------------------|----|
| 修订记录..... | 2 |
| 目 录..... | 3 |
| 1 产品概述..... | 4 |
| 2 板卡硬件结构..... | 5 |
| 3 硬件资源及使用说明 | 8 |
| 3.1 供电及电源分配..... | 8 |
| 3.2 FPGA 配置方法 | 8 |
| 3.2.1 JTAG 在线调试..... | 9 |
| 3.2.2 TF 卡配置..... | 9 |
| 3.3 主时钟与复位..... | 9 |
| 3.4 板卡调试与指示设备 | 10 |
| 3.5 AD9361 射频前端相关的 FPGA 管脚..... | 11 |
| 3.6 Y310 外部参考校准原理..... | 13 |
| 4 FPGA 收发回环参考例程 | 14 |
| 4.1 参考例程结构..... | 14 |
| 4.2 VIO 控制参数说明 | 15 |
| 4.3 BBPLL 和 RF 频点计算方法 | 21 |
| 4.4 参考例程运行 | 22 |

1 产品概述

北京威视锐科技 YunSDR Y3x0 软件无线电开发套件，主要面对无线通信、频谱分析和信号处理领域应用，可作为无线通信项目开发验证、通信原理教学、无线通信学习、嵌入式开发的理想平台。套件采用 Xilinx ZYNQ 和 ADI AD9361 集成化射频方案。本文档主要介绍 YunSDR Y3x0 平台的硬件开发资源以及基于 FPGA 的参考例程使用说明。

嵌入式 SOC 主要特点如下：

- ✓ 核心控制 FPGA 采用 Xilinx ZYNQ 方案
- ✓ ARM cortex A9 双核处理器+中等规模可编程逻辑
- ✓ 10/100/1000M 高速以太网，支持标准 TCP/IP 协议栈
- ✓ 标准 USB2.0 OTG 接口可以外接 U 盘、无线模块等标准 USB 设备与主机
- ✓ Micro usb 调试接口同时可供电
- ✓ 小尺寸适合便携 160x100x30mm
- ✓ 固件保存在 TF 卡中，方便更新与维护
- ✓ 预留 4GB NAND flash
- ✓ 提供 ZYNQ 固件源代码，实现可编程逻辑与 ARM 高速通信，支持用户二次开发

射频前端主要特点：

YunSDR Y300:

- ✓ 支持 MIMO 2x2
- ✓ 最高采样率 61.44MHz
- ✓ 单通道最大带宽 56MHz
- ✓ 发射功率最大 0dBm，最佳 EVM 输出功率 -18dBm @ EVM = -36dB
- ✓ 发射功率衰减范围 0~90dB @ step 0.25dB
- ✓ 接收功率调整范围 0~73dB @ step 1dB
- ✓ 射频前端 balun 频段可调
 - 800MHz-1GHz 1.5GHz-1.7GHz 1.7GHz-2GHz 2.4GHz-2.5GHz 3.7GHz-4GHz 4.9GHz-5.9GHz
 - 全频段 70-6000MHz

YunSDR Y310:

- ✓ 板载功率放大器，支持发射功率最高 15dBm (单音)
- ✓ 通过板载不同频段的 balun 组，支持 70~6GHz 全频段
 - 发送 A 端口 3GHz-6GHz
 - 发送 B 端口 70MHz-3GHz
 - 接收 A 端口 3GHz-6GHz
 - 接收 B 端口 1.6GHz-4GHz
 - 接收 C 端口 70MHz-2.2GHz
- ✓ 板载天线开关支持 TDD 与 FDD 模式切换
- ✓ 板载鉴相器，支持外部参考时钟同步，支持内部时钟输出
- ✓ **选配** GPS 模块，支持 GPS 参考与 1PPS 脉冲同步

YunSDR Y3x0 系列开发流程

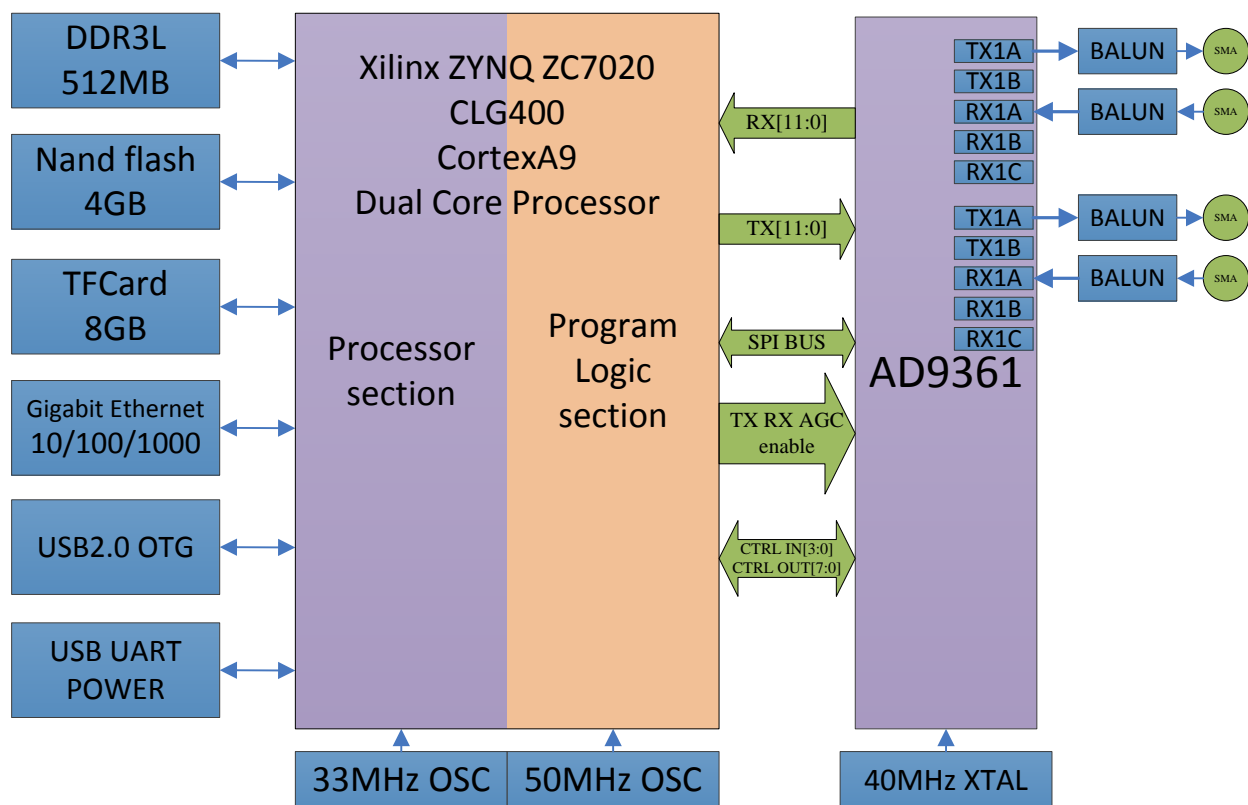
<http://www.v3t.com.cn>

- ✓ 支持 matlab 开发环境
 - 支持通过 matlab 配置射频参数
 - 发送端支持发送 matlab 产生的 IQ 数据
 - 通过 matlab 可获取射频前端采集到的 IQ 数据
 - 提供基于 matlab 版本 ieee802.11a 基带算法源程序
- ✓ 支持 gnuradio 开发环境
 - 支持通过 gnuradio 配置射频参数
 - 支持通过 gnuradio 收发基带数据
 - 提供 FM 调制方式的音频传输参考例程
- ✓ 支持二次开发
 - 提供 TCP/IP 通信协议，可实现通用平台软件无线电开发
- ✓ FPGA 底层开发支持
 - 提供基于 FPGA RTL 级控制 AD9361收发的源程序
 - 提供基于 FPGA RTL 级简化版802.11a(PHY)收发的源程序（选配组件）

2 板卡硬件结构

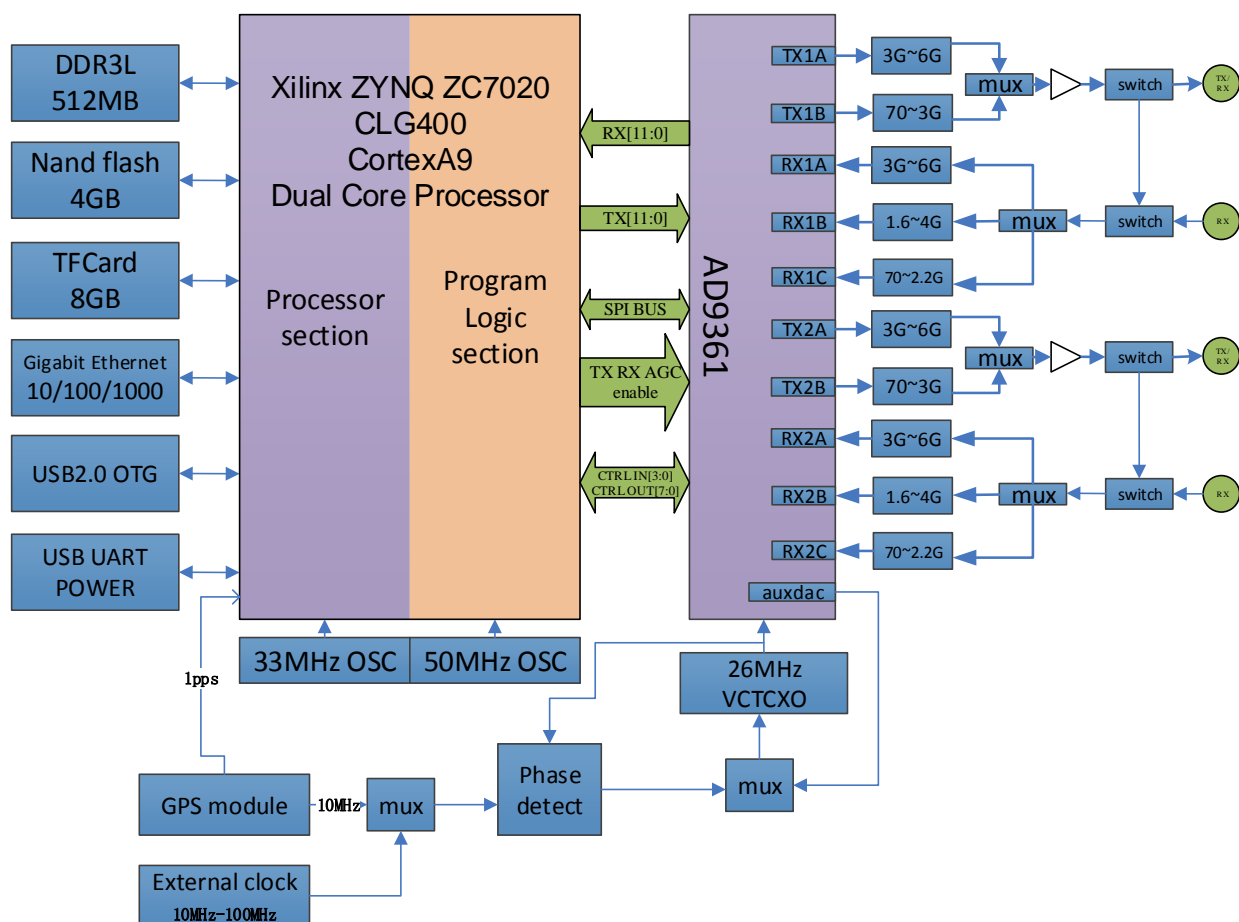
板卡由 ZYNQ 嵌入式处理器、AD9361 射频前端和电源电路构成，嵌入式处理器采用 Xilinx 新一代可编程逻辑和 CortexA9 双核处理器架构的 ZYNQ 系列 XC7Z020，外设支持 10/100/1000M 以太网、USB OTG、TFcard、UART 串口和 PIO 等，结合 ADI 公司的 AD9361 70~6000MHz 集成射频前端组成理想的软件无线电开发平台。

Y300 板卡的结构框图如下：



Y310 与 Y300 平台基带部分完全相同，Y310 采用升级的射频模块。Y310 和 Y300 在 AD9361 的数字接口是完全相同的，模拟端增加了功率放大器，天线开关，balun 等组件，提升了模块的实用性，Y310 射频部分的原理框图如下图。Y310 和 Y300 的主要区别如下：

- ✓ 板载功率放大器，支持发射功率最高15dBm(单音)
- ✓ 通过板载不同频段的 balun 组，支持70~6GHz 全频段，增加了单端控制 IO
 - 发送 A 端口 3GHz-6GHz
 - 发送 B 端口 70MHz-3GHz
 - 接收 A 端口 3GHz-6GHz
 - 接收 B 端口 1.6GHz-4GHz
 - 接收 C 端口 70MHz-2.2GHz
- ✓ 板载天线开关支持 TDD 与 FDD 模式切换，增加了单端控制 IO
- ✓ 板载鉴相器，支持外部参考时钟同步，支持内部时钟输出，增加了单端控制 IO
- ✓ 选配 GPS 模块，支持 GPS 参考与1PPS 脉冲同步



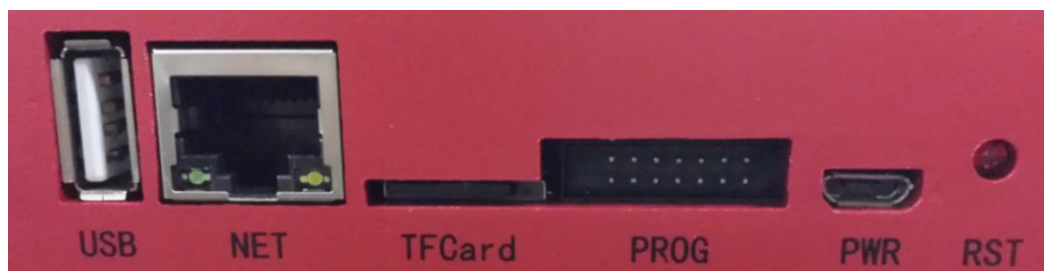
前面板功能:

从左至右分别为: 电源开关、启动模式选择开关(←TFCard; →jtag)、sma 射频连接器 TX2、RX2、RX1、TX1。在 SMA 射频连接器下方对应的是四个双色 LED 灯。



后面板功能:

从左至右分别为: USB2.0 OTG、RJ45 自适应网口、TFCard、JTAG 调试、MicroUSB 调试串口(同时供电), LED。



3 硬件资源及使用说明

3.1 供电及电源分配

板卡采用 MicroUSB 接口供电，YunSDR Y3x0 典型功耗 5W，因此可以选择 PC 的 USB3.0 接口供电(USB3.0 最大功率 10W)或 5V 2A 的 USB 电源适配器。当使用 PC 的 USB 接口供电时，MicroUSB 接口同时也可作为串口调试。

5V 系统电源通过 DCDC 变换不同的电压驱动板卡上的 FPGA 及其他电路。FPGA 的不同的区域接口电平如下：

| bank | 电平 (V) | 外设 |
|---------|--------|---------------------------|
| BANK0 | 3.3 | 配置电路 |
| BANK13 | 3.3 | 悬空 |
| BANK34 | 1.8 | AD9361 及 50MHz 晶振，LED |
| BANK35 | 1.8 | |
| BANK500 | 3.3 | NAND Flash, 33MHz 晶振 |
| BANK501 | 3.3 | USB OTG, 网络, TFCard, uart |
| BANK502 | 1.35 | DDR3L(MT41K128M16JT x2) |

3.2 FPGA 配置方法

YunSDR Y3x0 支持多种 FPGA 配置方法，出厂默认设置 SD 卡启动模式，出厂 SD 卡中已经存储了 YunSDR 软件无线电应用的镜像。此外板卡预留 JTAG 接口用于 FPGA 调试。上电启动模式由套件前面板左下方的拨码开关选择，左侧是 TF 卡启动，右侧是 JTAG 启动。**请注意：当处于 TF 启动模式时，在 FPGA 配置成功后可以使用 JTAG 下载电缆进**

行在线调试，比如通过 **chipscope** 采集观测 **FPGA** 内部信号波形，但是不可以下载 **FPGA** 固件，只有拨码开关处在 **JTAG** 模式时才可以下载 **FPGA** 的固件。

3.2.1 JTAG 在线调试

在板卡的前面板接口处的 PTOG 是 **FPGA** 的 **JTAG** 下载接口，采用 2.0mm 间距 14pin 连接器，请使用威视锐公司的 redcable-xilinx 下载电缆，当使用第三方下载电缆时，请注意下载电缆的接口定义，防止接线错误。



接口定义如下：

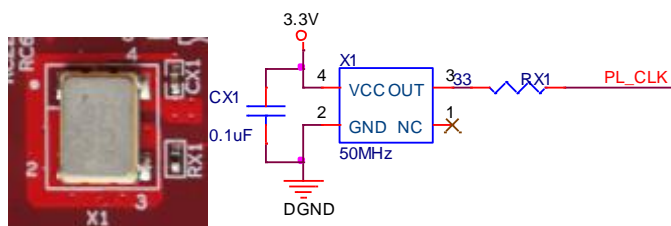
| PIN | NAME | PIN | NAME |
|-----|------|-----|------|
| 1 | GND | 2 | 3.3V |
| 3 | GND | 4 | TMS |
| 5 | GND | 6 | TCK |
| 7 | GND | 8 | TDO |
| 9 | GND | 10 | TDI |
| 11 | GND | 12 | NC |
| 13 | GND | 14 | NC |

3.2.2 TF 卡配置

ZYNQ 的 TF 卡启动需要制作 boot.bin 镜像，必须由 CPU 参与进行加载。

3.3 主时钟与复位

主板只提供单晶振时钟源，默认焊接 50MHz 晶振，封装 5x7mm。



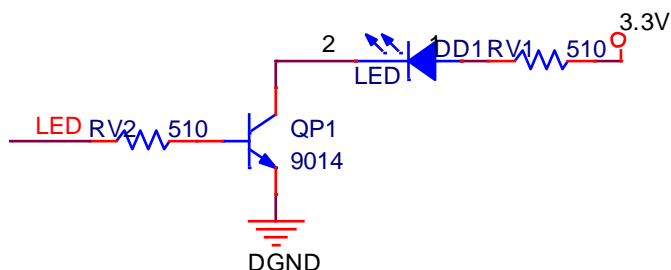
| 信号名称 | 连接器管脚 | FPGA 管脚 |
|--------|-------|---------|
| PL_CLK | X1 | H16 |

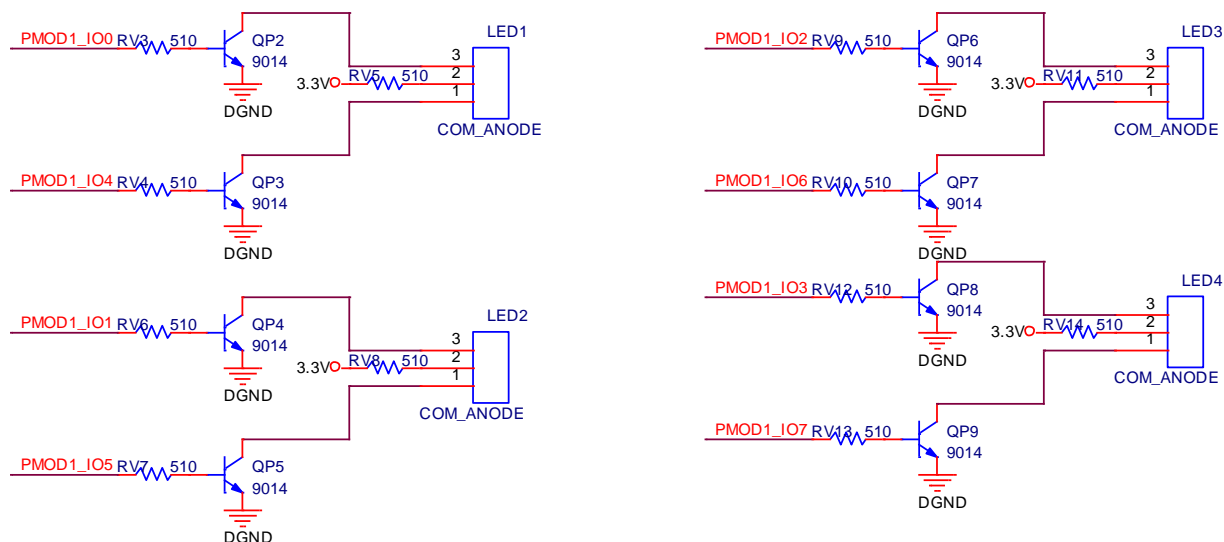
3.4 板卡调试与指示设备

主板提供一个 LED 指示灯伸出前面板供用户输出调试状态，高电平点亮。板卡的后面板提供四个双色 LED 在每个 SMA 接口的下方，可用于指示 AD9361 射频通道工作的状态或自定义功能。

连接到 FPGA 的调试 IO:

| 信号名称 | 设备位号 | FPGA 管脚 | 功能 |
|-----------|------|---------|---------------------|
| LEDG0 | DD1 | V18 | 高电平点亮指示 AD9361 配置完成 |
| PMOD1_I00 | LED1 | K14 | 高电平红色亮指示 TX2 通道关闭 |
| PMOD1_I04 | | J14 | 高电平绿色亮指示 TX2 通道打开 |
| PMOD1_I01 | LED2 | M15 | 高电平红色亮指示 RX2 通道关闭 |
| PMOD1_I05 | | H17 | 高电平绿色亮指示 RX2 通道打开 |
| PMOD1_I02 | LED3 | L15 | 高电平红色亮指示 RX1 通道关闭 |
| PMOD1_I06 | | M14 | 高电平绿色亮指示 RX1 通道打开 |
| PMOD1_I03 | LED4 | U17 | 高电平红色亮指示 TX1 通道关闭 |
| PMOD1_I07 | | P15 | 高电平绿色亮指示 TX1 通道打开 |





3.5 AD9361 射频前端相关的 FPGA 管脚

| 信号名称 | FPGA 管脚 | 信号名称 | FPGA 管脚 |
|-----------------|---------|------------------|---------|
| rx_clk_in_n | U19 | tx_clk_out_n | W16 |
| rx_clk_in_p | U18 | tx_clk_out_p | V16 |
| rx_data_in_n[0] | W13 | tx_data_out_n[0] | R18 |
| rx_data_in_n[1] | V13 | tx_data_out_n[1] | T15 |
| rx_data_in_n[2] | W15 | tx_data_out_n[2] | Y17 |
| rx_data_in_n[3] | W20 | tx_data_out_n[3] | U20 |
| rx_data_in_n[4] | W19 | tx_data_out_n[4] | Y19 |
| rx_data_in_n[5] | U12 | tx_data_out_n[5] | R14 |
| rx_data_in_p[0] | V12 | tx_data_out_p[0] | T17 |
| rx_data_in_p[1] | U13 | tx_data_out_p[1] | T14 |
| rx_data_in_p[2] | V15 | tx_data_out_p[2] | Y16 |
| rx_data_in_p[3] | V20 | tx_data_out_p[3] | T20 |
| rx_data_in_p[4] | W18 | tx_data_out_p[4] | Y18 |
| rx_data_in_p[5] | T12 | tx_data_out_p[5] | P14 |
| rx_frame_in_n | P19 | tx_frame_out_n | Y14 |
| rx_frame_in_p | N18 | tx_frame_out_p | W14 |

| | | | |
|-------------|-----|----------|-----|
| ctrl_in[0] | E18 | en_agc | R16 |
| ctrl_in[1] | E19 | enable | N17 |
| ctrl_in[2] | K16 | resetb | K19 |
| ctrl_in[3] | J16 | txnrx | P18 |
| ctrl_out[0] | N15 | sync_in | R17 |
| ctrl_out[1] | N16 | spi_clk | F20 |
| ctrl_out[2] | H15 | spi_csn | F19 |
| ctrl_out[3] | G15 | spi_miso | H20 |
| ctrl_out[4] | M17 | spi_mosi | J20 |
| ctrl_out[5] | M18 | pg_fmc | V17 |
| ctrl_out[6] | M19 | | |
| ctrl_out[7] | M20 | | |

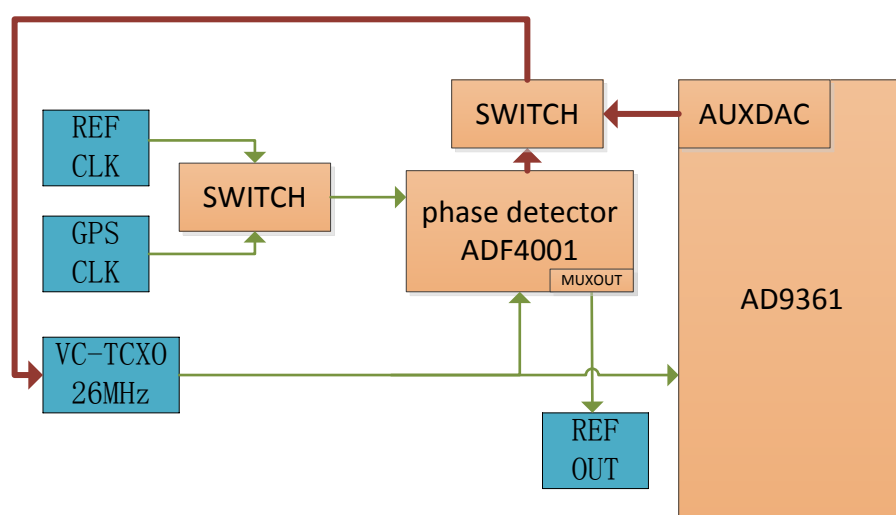
Y310 新增功能管脚

| 信号名称 | FPGA 管脚 | 功能 |
|----------------|---------|---|
| ref_select | K17 | 1=外部参考时钟 0=内部参考(GPS 模块) |
| vco_cal_select | K18 | 晶振压控端选择 1=AD9361 AUXDAC1 0=ADF4001 鉴相器 |
| pps_1s | L17 | GPS 模块 1pps 输出 |
| rx1_band_sel_a | G17 | RX1_BAND_SEL_A,B=10 RX1A 3G~6G |
| rx1_band_sel_b | G18 | RX1_BAND_SEL_A,B=11 RX1B 1.6~4G RX1_BAND_SEL_A,B=01 RX1C 70~2.2G |
| rx2_band_sel_a | B19 | RX2_BAND_SEL_A,B=11 RX1A 3G~6G |
| rx2_band_sel_b | A20 | RX2_BAND_SEL_A,B=10 RX1B 1.6~4G RX2_BAND_SEL_A,B=01 RX1C 70~2.2G |
| fdd_tdd_sel | D19 | 1=FDD 模式, TXsma 发, RXsma 收; 0=TDD 模式, TXsma 收发, RXsma 不用 |
| trx_sw | D20 | FDD 模式时=1; TDD 模式时 1=TX, 0=RX |
| tx_band_sel | F16 | 1=3G~6G TX1/2A;0=70~3G TX1/2B |

| | | |
|--------------|-----|------------------|
| ref_pll_clk | F17 | ADF4001 SPI 配置时钟 |
| ref_pll_data | E17 | ADF4001 SPI 配置数据 |
| ref_pll_le | D18 | ADF4001 SPI 配置使能 |
| clk_out | N20 | 参考时钟输出 |

3.6 Y310 外部参考校准原理

Y310 增加了外部参考校准的功能，可以通过板载 GPS 模块、外部参考时钟对板卡上的 VC-TCXO(温补压控振荡器)进行校准，使板载时钟的基准频率与参考时钟一致。原理框图如下：



外部参考时钟 reference 和 vctcxo 时钟 vc-clock 分别进行分频，分频后鉴相，将 CP 电压输出给 VC-TCXO 的压控端，当 ADF4001 锁定后 VC-TCXO 与外部参考的基准时钟严格同频。假设外部参考 10MHz，本地晶振 26MHz，rcount 表示参考频频系数，ncount 表示本地时钟分频系数，另 rcount=10，ncount=26 可以完成 ADF4001 的鉴相功能。

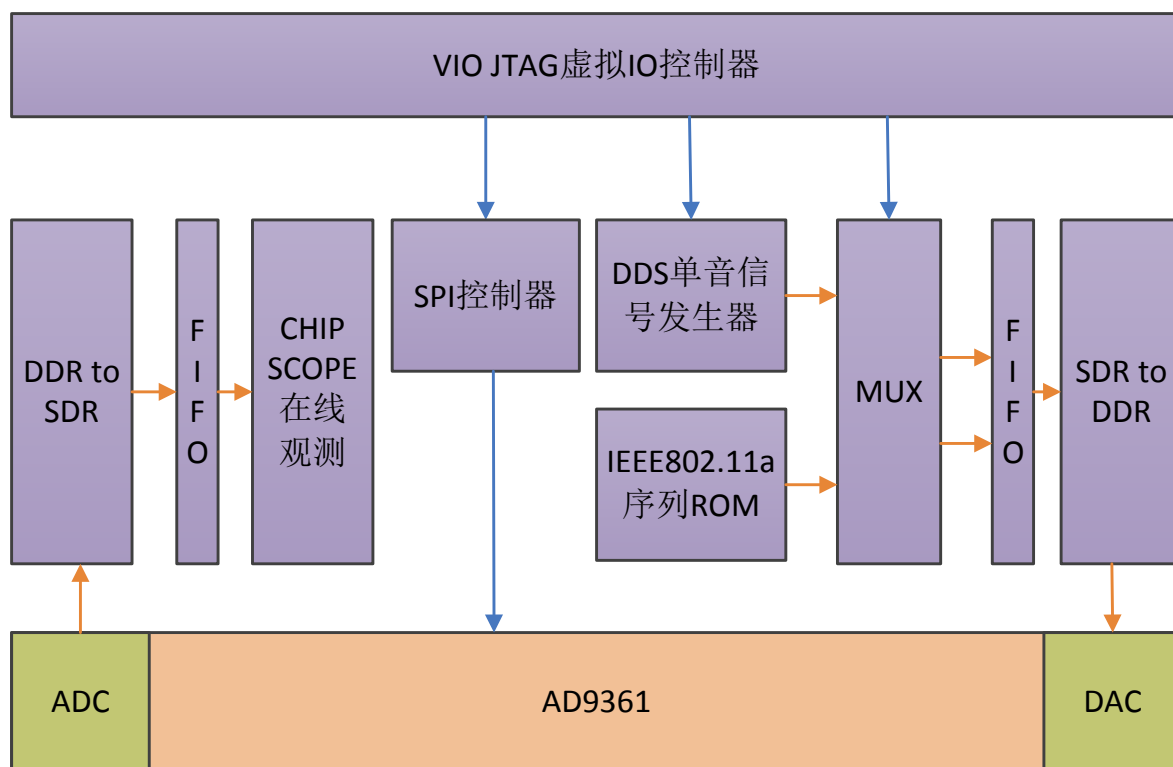
此外 ADF4001 还有另外一个功能，完成双板卡级联同步。当没有 GPS 或者外部的标准参考时，可以通过将一块板卡的参考输出连接到第二块板卡的参考输入上完成两块板卡的同步。ADF4001 的 muxout 可以选择输入 ncount 分频后的时钟，将第一个板卡的 ncount=1，则参考输出 26MHz 时钟，这时第二块板卡设置 ncount=1，rcount=1，将第二块板卡的时钟锁定到第一路的时钟上，完成同步。

4 FPGA 收发回环参考例程

本例程是使用 FPGA 单独控制 AD9361 进行收发数据回环的参考例程。参考例程包括 AD9361 的 SPI 配置功能，FPGA 内部生成 DDS 单音信号源，FPGA 内部 ROM 存储标准 IEEE80.211a 36Mbps 码率的 IQ 序列 AD9361 DDR 转 SDR 接口模块，ADC 采样信号的 chipscope 波形显示，AD93561 工作状态参数的 VIO 配置。

4.1 参考例程结构

本例程只使用 zynq 的 PL 部分进行 AD9361 的操作，方便用户对 AD9361 进行学习。参考例程的结构框图如下：



参考例程采用 verilog 语言设计，设计文件列表如下：

| 序号 | 文件名称 | 功能 |
|----|---------------------|--------------------------|
| 1 | adaloop.v | 顶层文件 |
| 2 | axi_ad9361_dev_if.v | AD9361 数字端口 DDR 转 SDR 匹配 |
| 3 | ad9361_spi_drv.v | AD9361 spi 配置控制 |
| 4 | ad9361_spi_if.v | AD9361 spi 配置驱动 |
| 5 | ad9361_lut.v | AD9361 spi 配置初始化查找表 |

| | | |
|----|------------------|---|
| 6 | clk_wiz_v3_6.xco | Xilinx ipcore pll |
| 7 | dds1.xco | Xilinx ipcore dds |
| 8 | fifo_128x64.xco | Xilinx ipcore fifo 128depth *64width |
| 9 | frame36_4096.xco | Xilinx ipcore rom ieee802.11a 36Mbps 两倍插值 IQ 数据 |
| 10 | icon1.xco | Xilinx ipcore 在线调试控制器 |
| 11 | ila1.xco | Xilinx ipcore 在线调试 chipscope |
| 12 | viol.xco | Xilinx ipcore 在线调试虚拟 io |
| 13 | adf4001_spi.v | Y310 新增配置 ADF4001 鉴相器的模块 |

4.2 VIO 控制参数说明

VIO 是 xilinx 在线调试工具，可以虚拟输入输出 IO，本例程采用虚拟输出 IO 方式，通过在 pc 端的 analyzer 控制器输入参数，可以通过 jtag 将配置参数传递到 FPGA 中控制程序的工作参数，配置参数如下表：

| 参数 | 位宽 | 功能 |
|-----------|----|---|
| sel | 4 | [1:0] 与[3:2]分别控制 TX1 与 TX2 通道输出信号。b01 单音;b10 ieee802.11a 其他 零序列 |
| phase | 12 | DDS 累加相位递增选择，值越大单音信号输出频率越大，输出单音信号频率= $40 \div 2^{12} \times phase(MHz)$ |
| rx1_index | 8 | RX1 通道增益序号，d0~77 |
| rx2_index | 8 | RX2 通道增益序号，d0~77 |
| tx1_att | 8 | TX1 通道衰减，d0~255，衰减值= $0.25 \times tx1att (dB)$ |
| tx2_att | 8 | TX2 通道衰减，d0~255，衰减值= $0.25 \times tx2att (dB)$ |
| tx_chan | 2 | 保留 |
| rx_chan | 2 | 保留 |
| lo_start | 1 | 频率调整使能，当修改以下频率参数后，将 lo_start 先拉高在拉低会将以下频率参数整体配置到 ad9361 中 |
| lo_txint | 16 | 发送端 VCO 整数部分 |

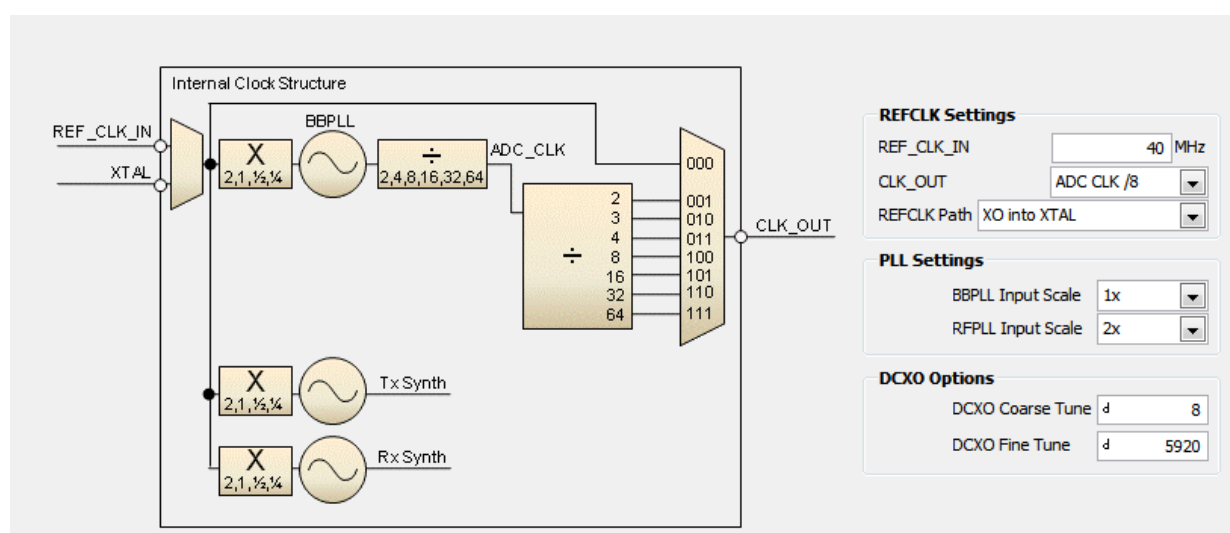
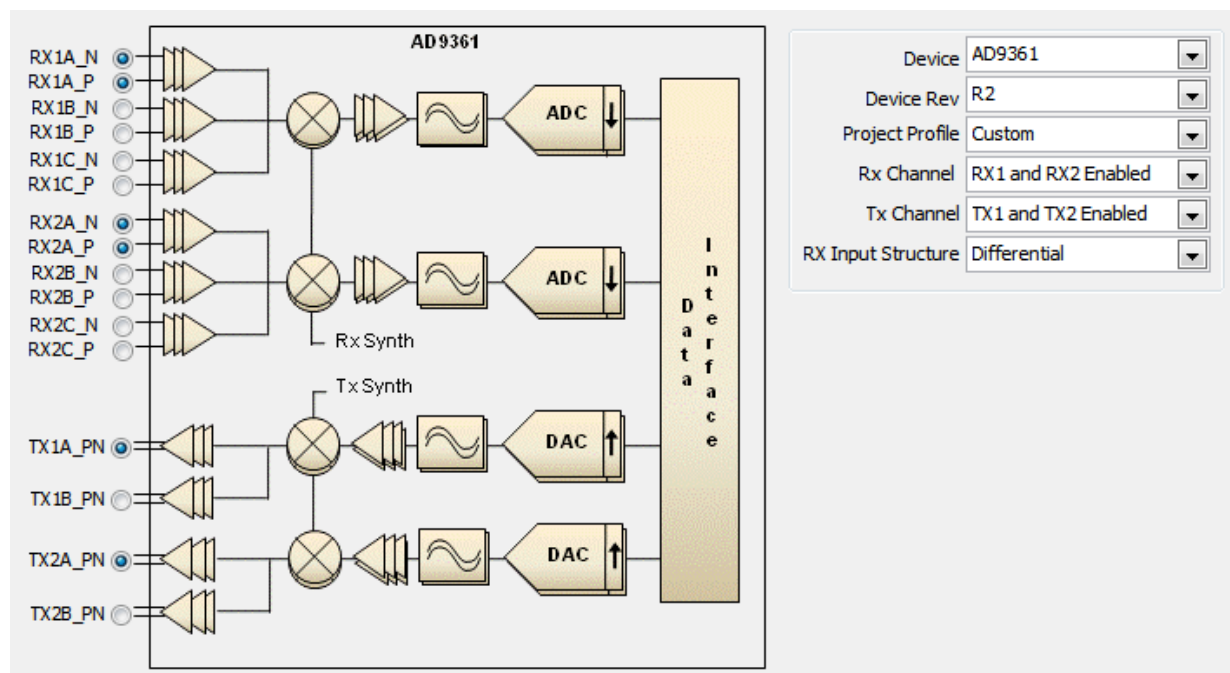
| | | |
|-----------|----|--------------|
| lo_txfrac | 24 | 发送端 VCO 小数部分 |
| lo_txdiv | 4 | 发送端 VCO 输出分频 |
| lo_rxint | 16 | 接收端 VCO 整数部分 |
| lo_rxfrac | 24 | 接收端 VCO 小数部分 |
| lo_rxdiv | 4 | 接收端 VCO 输出分频 |
| rst_vio | 1 | 保留 |
| bb_frac | 24 | BBPLL 小数部分 |
| bb_int | 8 | BBPLL 整数部分 |
| bb_div | 3 | BBPLL 输出分频 |

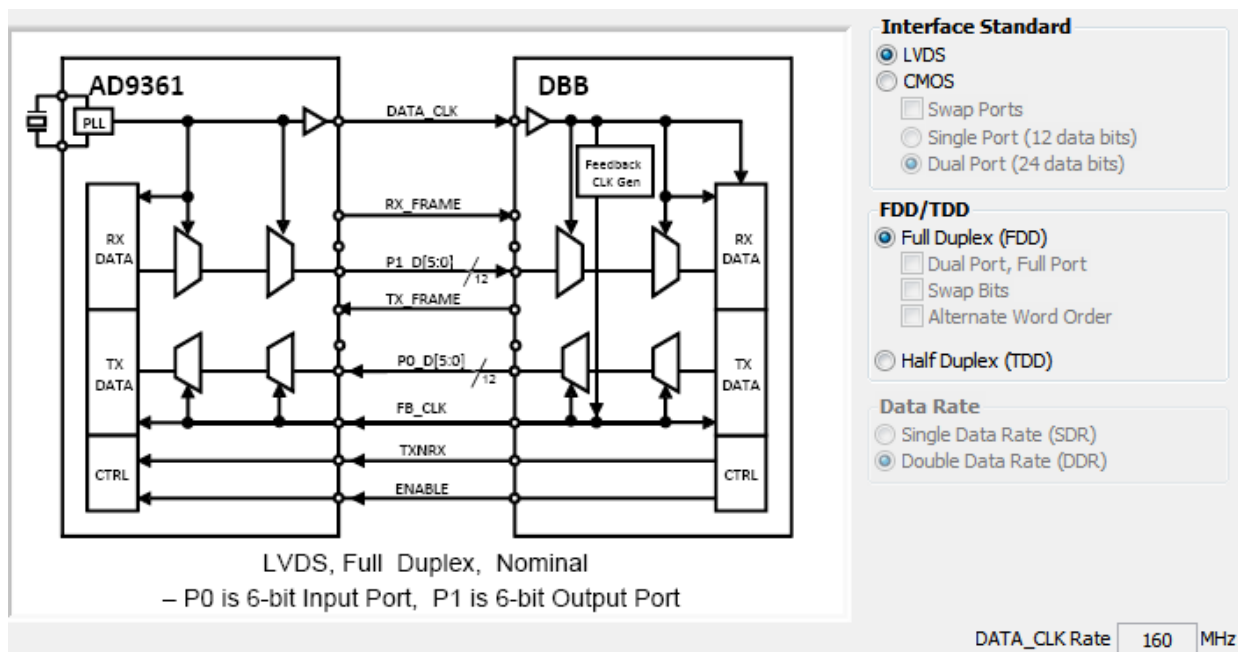
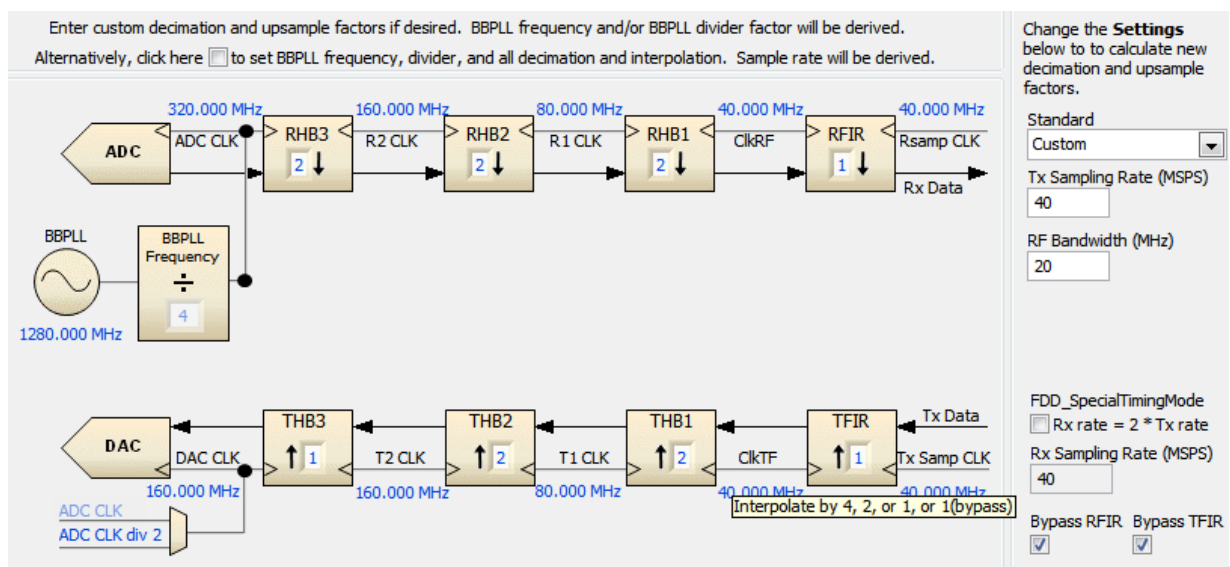
Y310 新增配置参数

| 参数 | 位宽 | 功能 |
|----------------|----|--|
| ref_select | 1 | 1=外部参考时钟 0=内部参考(GPS 模块) |
| vco_cal_select | 1 | 晶振压控端选择 1=AD9361 AUXDAC1 0=ADF4001 鉴相器 |
| rx1_band_sel_a | 1 | RX1_BAND_SEL_A,B=10 RX1A 3G~6G |
| | | RX1_BAND_SEL_A,B=11 RX1B 1.6~4G |
| rx1_band_sel_b | 1 | RX1_BAND_SEL_A,B=01 RX1C 70~2.2G |
| rx2_band_sel_a | 1 | RX2_BAND_SEL_A,B=11 RX1A 3G~6G |
| | | RX2_BAND_SEL_A,B=10 RX1B 1.6~4G |
| rx2_band_sel_b | 1 | RX2_BAND_SEL_A,B=01 RX1C 70~2.2G |
| fdd_tdd_sel | 1 | 1=FDD 模式, TXsma 发, RXsma 收; 0=TDD 模式, TXsma 收发, RXsma 不用 |
| trx_sw | 1 | FDD 模式时=1; TDD 模式时 1=TX, 0=RX |
| tx_band_sel | 1 | 1=3G~6G TX1/2A;0=70~3G TX1/2B |
| rf_chan | 8 | RF 通道选择, 需要根据 rx_band_sel 和 tx_bans_sel 进行配置 [7]=0,[6]=0 TX1/2A;[6]=1 TX1/2B [5:0]=000011 RX1/2A [5:0]=001100 RX1/2B [5:0]=110000 RX1/2C |

| | | |
|---------|----|---|
| test_en | 1 | 1=测试模式，ADC 可以采集到标准单音信号 0=正常工作模式 |
| auxdac1 | 10 | AD9361 集成的辅助 DAC1 输出数字量 0~1023 |
| auxvref | 2 | 选择辅助 DAC 参考电源 00=1V;01=1.5V;10=2V;11=2.5V |

FPGA 配置完成后默认收发频点 2422MHz，采样率 40MHz，发送衰减 20dB，接收增益 0dB。其他配置工作状态如下图：





Data Interface Pins
VDD_INTERFACE
LVDS 1.8v-2.5v

- DATACLK (1 Pair)
- RX FRAME (1 Pair)
- P0 DATA[5:0] (6 Pair)
- FBCLK (1 Pair)
- TX FRAME (1 Pair)
- P1 DATA[5:0] (6 Pair)

Control Interface Pins
VDD_INTERFACE
1.8v-2.5v

- ENABLE
- TXNRX
- ENAGC
- CTRLOUT[7:0]
- CTRLIN[3:0]
- SPI_CLK
- SPI_ENB
- SPI_DI
- SPI_DO
- RESETB

General Purpose Outputs
VDD_GPO
2.5v-3.3v

- GPO[3:0]

Clock Pins
VDD_INTERFACE
1.8v-2.5v

- REFCLKIN (1.3v Only)
- REFCLKOUT

LVDS Settings

In LVDS mode, the PN pair can be swapped by enabling an inverter on the single ended side of the LVDS buffer inside the AD936x device. Default is the unchecked state.

- ☐ FBCLK
- ☐ Tx Frame
- ☐ DATACLK
- ☐ Rx Frame
- ☐ TX 5 (P0[11:10])
- ☐ TX 4 (P0[9:8])
- ☐ TX 3 (P0[7:6])
- ☐ TX 2 (P0[5:4])
- ☐ TX 1 (P0[3:2])
- ☐ TX 0 (P0[1:0])
- ☐ RX 5 (P1[11:10])
- ☐ RX 4 (P1[9:8])
- ☐ RX 3 (P1[7:6])
- ☐ RX 2 (P1[5:4])
- ☐ RX 1 (P1[3:2])
- ☐ RX 0 (P1[1:0])

☒ Enable LVDS On chip termination

150 mV RX LVDS amplitude

General Settings

☒ Always use 2R2T Timing

☐ Invert DATACLK

☐ Invert Bus P0[11:0] <--> P0[0:11]

Tx Data Settings

☒ Tx Swap IQ

☐ Tx Swap Channel

Rx Data Settings

☒ Rx Swap IQ

☐ Rx Swap Channel

0 Rx Data delay (1 sample)

Rx Frame Settings

☒ Rx Frame Pulse Mode

☐ Invert Rx Frame

Delay Cell Control

0 DATACLK delay

4 Rx Data/Rx Frame delay

7 FBCLK delay

0 Tx Data/Tx Frame delay

DATA_CLK Rate 160 MHz

Manual Enable
Auto Enable
Reg 0x026[7]

Manual Enable
Auto Enable
Reg 0x026[7]

Volts

DAC Code

AuxDAC Settings

AuxDAC 1 Value × 200

Vref 1v

Step Factor 2mV

AuxDAC 2 Value × 200

Vref 1v

Step Factor 2mV

☒ Enable Manual AuxDAC Control

☐ Enable Automatic AuxDAC Control

☒ Enable AuxDAC 1

☒ Enable AuxDAC 2

☐ Enable AuxDAC 1 in ALERT

☐ Enable AuxDAC 1 in RX

☐ Enable AuxDAC 1 in TX

☐ Enable AuxDAC 2 in ALERT

☐ Enable AuxDAC 2 in RX

☐ Enable AuxDAC 2 in TX

AuxDAC 1 RX Delay (us) × 0

AuxDAC 1 TX Delay (us) × 0

AuxDAC 2 RX Delay (us) × 0

AuxDAC 2 TX Delay (us) × 0

In FDD, the TX Enable and TX delays apply, RX Enable and RX delays are ignored.

The AuxADC input mux is automatically set. When a temperature measurement is enabled, either manually or periodically, the input mux is set to the temperature sensor input. When the temperature measurement is complete, the mux is set back to the AuxADC pin.

Suggest Settings

Temperature Sensor Settings

Temp Sensor Decimation: 4096

Temp Sensor Offset[7:0]: d 206

Temp Sensor Window[3:0]: d 0

Periodic Temperature Measurement? ☒

Measurement Interval: d 2

Measurement Interval Time: 0.839 s

AuxADC Settings

AuxADC Clock Divide: d 24

AuxADC Decimation: 256

AuxADC Clock: 53.333 MHz

AuxADC Conversion Time: 4.800 us

Delays are only applied while moving into the RX or TX/FDD states. When moving back into the ALERT state, the GPO holds its value through the RX and TX FLUSH states.

GPO Settings(Manual/auto)

☐ Manual GPO Control

| GPO | 3 | 2 | 1 | 0 |
|--------|--------------------------|--------------------------|--------------------------|--------------------------|
| Enable | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

☒ Auto (ENSM) GPO Control

| GPO | 3 | 2 | 1 | 0 | 3 | 2 | 1 | 0 |
|------------------|--------------------------|--------------------------|-------------------------------------|-------------------------------------|--------------|---|---|---|
| Enable in ALERT | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | RX Delay(us) | 0 | 0 | 0 |
| Enable in RX | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | TX Delay(us) | 0 | 0 | 0 |
| Enable in TX/FDD | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | | | | |

In FDD, the TX Enable and TX delays apply, RX Enable and RX delays are ignored.

Gain Table Overview

Receiver Gain Control moves an index that points to a Gain Table. The Table sets the Gain in the Receiver Blocks shown below. The pointer moves up and down in "steps". Moving one step changes the pointer position by one row index.

Gain Index (Pointer) →

| LNA Index | Mixer Index | TIA Index | LPF Index | Digital Index |
|-----------|-------------|-----------|-----------|---------------|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Automatic vs. Manual Gain

Automatic/Manual Gain Control

☐ Automatic Gain Control (AGC)

☒ Manual Gain Control (MGC)

Automatic Gain Control Options

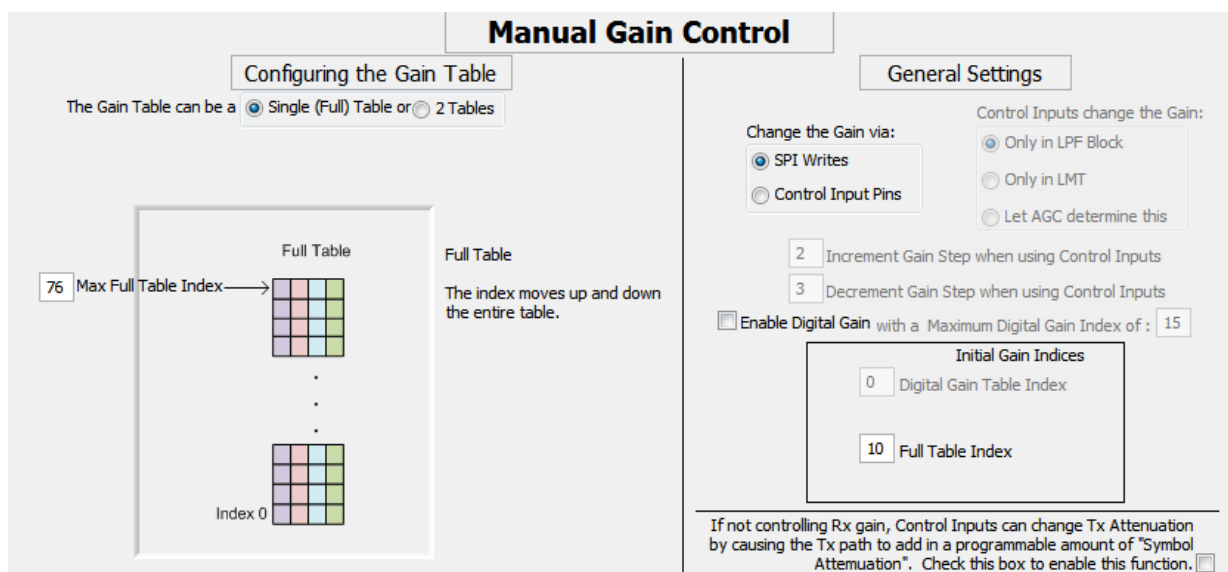
☒ Fast Attack (for "Bursty" Signals)

☐ Slow Attack (for Slowly Changing Signals)

Press to load values that work for most applications. Then, press "Finish" if no changes are needed.

Wait 10 microsecond(s) after the receivers are enabled to start the AGC.

Note: Mouse over any checkbox or numerical input box to temporarily see register(s) affected, and (for numerical inputs) units, resolution, and range. To open a window with this information, right click over a checkbox or numerical input box and select "Description and Tip".



4.3 BBPLL 和 RF 频点计算方法

Y300 参考时钟 40MHz，经过时钟前端电路，BBPLL 参考时钟 40MHz(允许范围 35~70MHz)，RF 参考时钟 80MHz(允许范围 5~320MHz)。射频 VCO 的频率范围是 6~12GHz，输出分频范围 1~128。BBPLL 的频率范围 715~1430MHz，输出分频 1~128。

选择合适的输出分频通过以下 matlab 脚本就可以计算锁相环的参数，比如选择收发频点 2422MHz，采样率 1280MHz。

```
refclk=40e6;
bbpll=1280e6;
rfpll=2422e6*4;
bbpll_int=dec2hex(floor(bbpll/refclk),2)
bbpll_frac=dec2hex(round((bbpll/refclk-floor(bbpll/refclk))*2088960),6)
rfpll_int=dec2hex(floor(rfpll/refclk/2),4)
rfpll_frac=dec2hex(round((rfpll/refclk/2-floor(rfpll/refclk/2))*8388593),6)
```

运算结果如下：

```
bbpll_int =0x20
bbpll_frac =0x000000
rfpll_int =0x0079
rfpll_frac =0x0CCCCB
```

辅助 DAC 计算方法：

```
dac=1023;
vref=1.5;
auxdac=0.97*vref+(0.000738+9e-6*(vref*1.6-2))*dac*2-0.3572*2+0.05;
```

结果 2.3079V

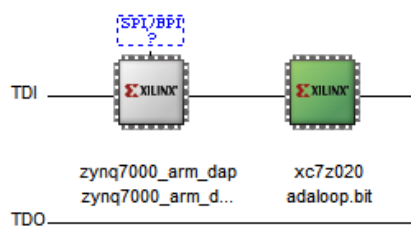
Y310 参考时钟 26MHz，经过时钟前端电路，BBPLL 参考时钟 52MHz(允许范围 35~70MHz)，RF 参考时钟 52MHz(允许范围 5~320MHz)。射频 VCO 的频率范围是 6~12GHz，输出分频范围 1~128。BBPLL 的频率范围 715~1430MHz，输出分频 1~128。

选择合适的输出分频通过以下 matlab 脚本就可以计算锁相环的参数，比如选择收发频点 2422MHz，采样率 1280MHz。

```
refclk=26e6;  
bbpll=1280e6;  
rfpll=2422e6*4;  
dac=1023;  
vref=1.5;  
bbpll_int=dec2hex(floor(bbpll/refclk/2),2)  
bbpll_frac=dec2hex(round((bbpll/refclk/2-floor(bbpll/refclk/2))*2088960),6)  
auxdac=0.97*vref+(0.000738+9e-6*(vref*1.6-2))*dac*2-0.3572*2+0.05;  
rfpll_int=dec2hex(floor(rfpll/refclk/2),4)  
rfpll_frac=dec2hex(round((rfpll/refclk/2-floor(rfpll/refclk/2))*8388593),6)  
运行结果:  
bbpll_int =0x18  
bbpll_frac =0x 139D8A  
rfpll_int =0x 00BA  
rfpll_frac =0x 276272
```

4.4 参考例程运行

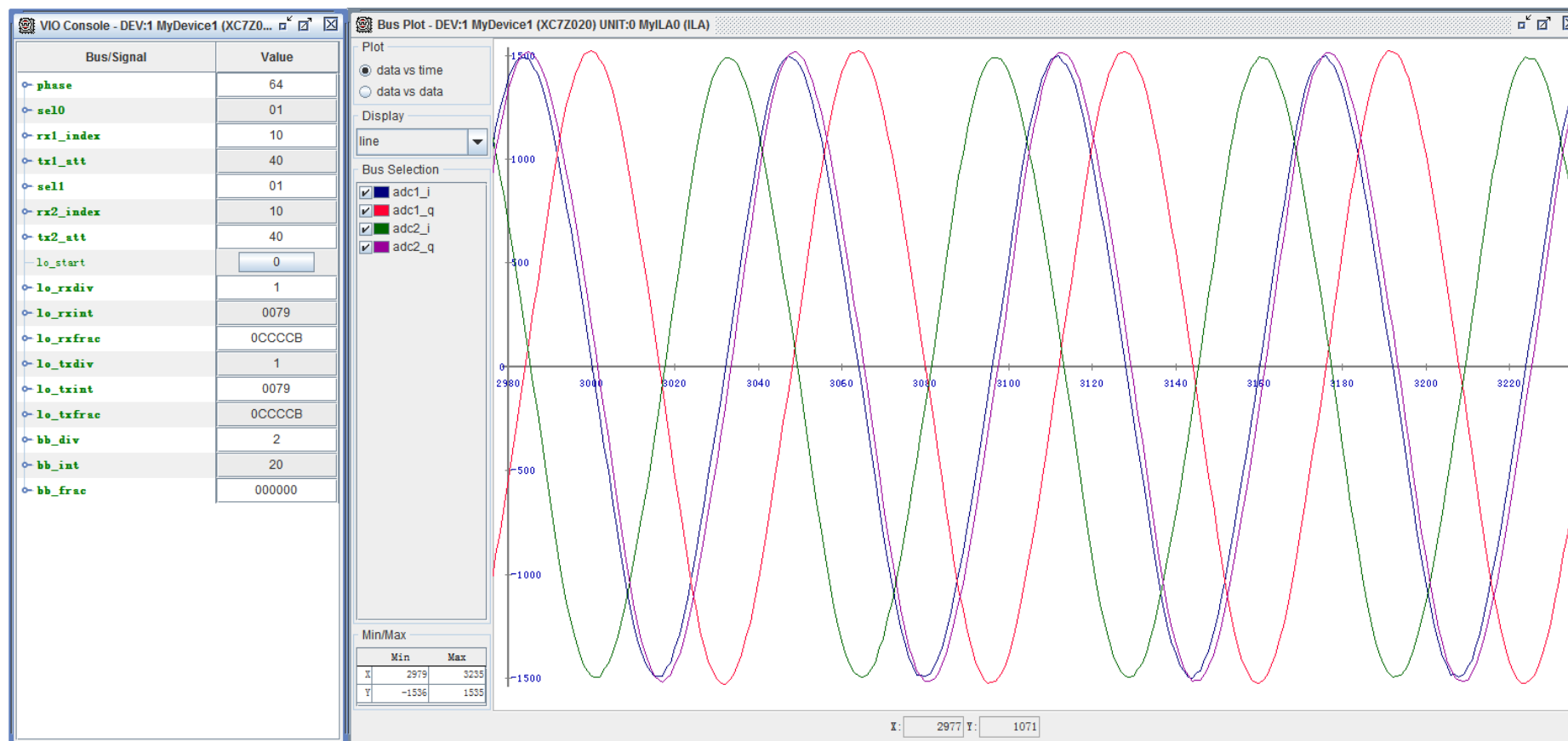
- 1 将射频的发送通过射频线缆直接连接到接收
- 2 板卡上电
- 3 下载 FPGA 配置文件



Program Succeeded

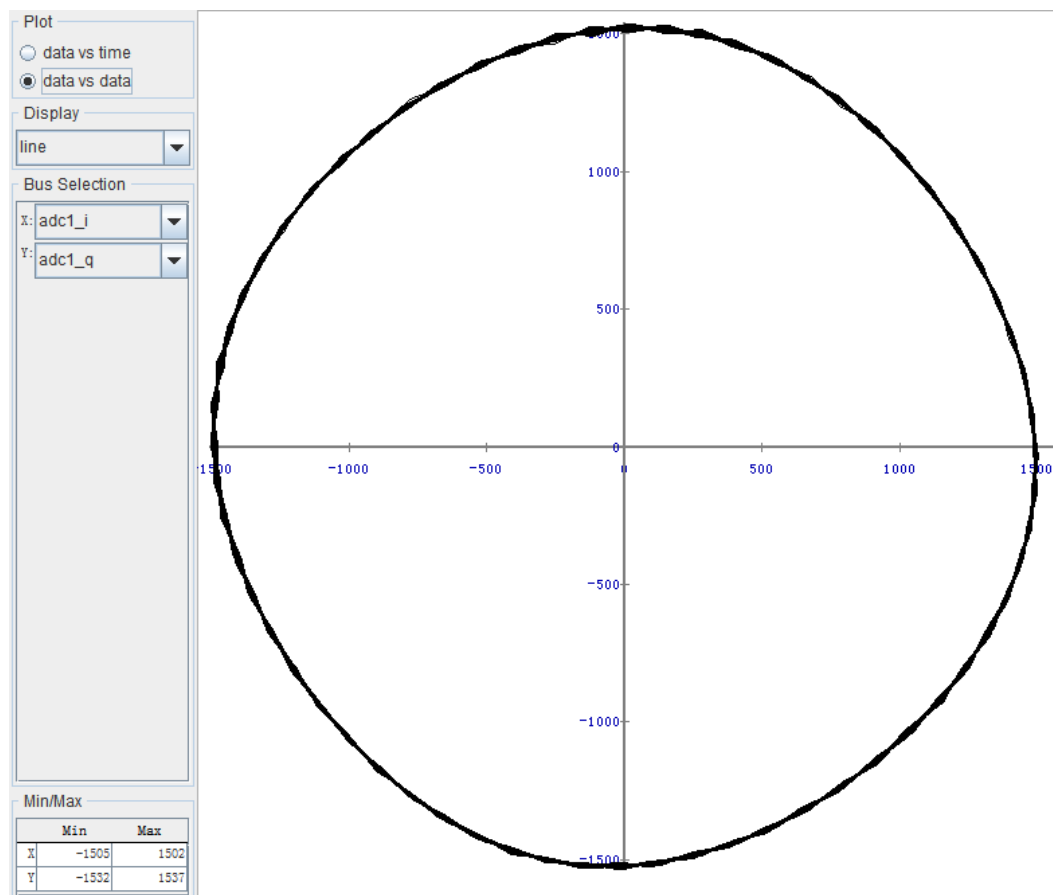
4 打开 analyzer，读入 adaloop.cpj

5 输入 VIO 数值，运行就可以看到 ADC 采集到的波形

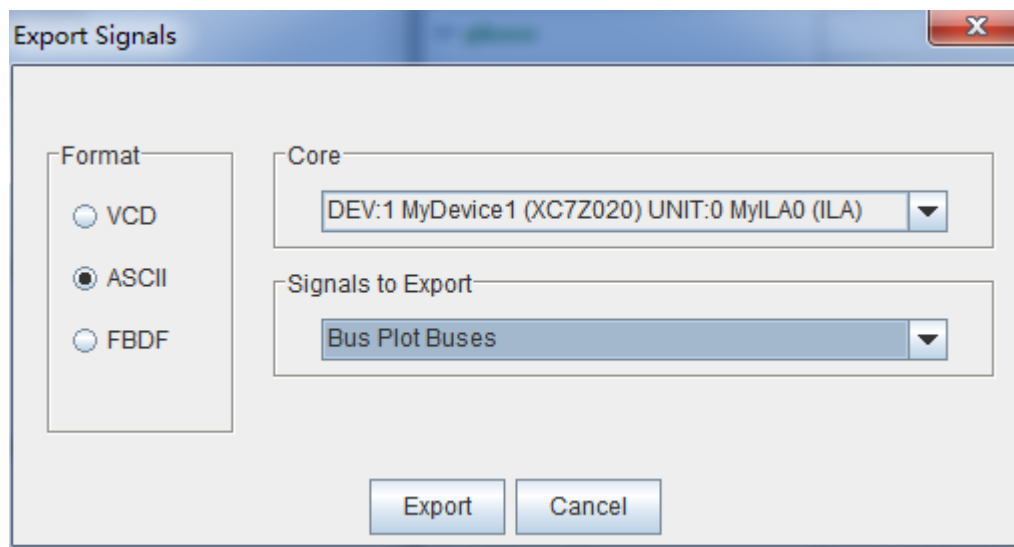


注意：VIO 的默认值是全零，所以在下载程序后需要把每一个数值输入一遍。

6 XY 轴显示



7 chipscope 波形导出, file-export



8 在 matlab 中进行数据分析

```
%% read .prn
[a1,a2,a3,a4,a5,a6]=textread('E:\code\yunsdrr\data\1.prn','%f%f%f%f%f','headerlines',1);
subplot(221)
plot(a3,a4);
axis equal;
subplot(222)
pwelch(a3+1i*a4,[],[],[],40e6,'centered','psd');
subplot(223)
plot(a5,a6);
```

```
axis equal;  
subplot(224)  
pwelch(a5+1i*a6,[],[],[],40e6,'centered','psd');
```

