

---

# FACE-RFSOC 高性能自适应 射频开发平台测试报告

---



文档类别	测试手册	密级	保密
文档编码	NC	作者	
更新日期	2021/11/4	文档版本	V5.0

## 目录

1. 概述.....	2
2. FACE-RFSOC-A 自适应射频开发平台.....	3
2.1 平台实物照片.....	3
2.2 平台特性与规格.....	3
3. 平台接口外设测试.....	6
3.1 测试准备.....	6
3.2 接口外设测试说明.....	9
3.2.1 PS DDR4 测试.....	9
3.2.2 PS 千兆以太网测试.....	9
3.2.3 PS USB 接口测试.....	11
3.2.4 PS DP 显示接口测试.....	12
3.2.5 PS EMMC 测试.....	13
3.2.6 PL DDR4 存储器测试.....	14
3.2.7 PL 用户拨码开关测试.....	16
3.2.8 PL 用户 J30J GPIO 测试.....	17
3.2.9 QSFP28 接口测试.....	18
3.2.10 QSFP Flash 测试.....	19
4. RF Analyzer 软件.....	23
4.1 RF Analyzer 介绍.....	23
4.2 RF Analyzer 安装.....	23
5. AD/DA 同步性能测试.....	24
5.1 测试准备.....	24
5.2 ADC 同步采集测试.....	26
5.3 DAC 同步输出测试.....	27
6. AD/DA 平坦度测试.....	30
6.1 ADC 平坦度.....	30
6.1.1 100-2000M.....	30
6.1.2 700-1300M.....	30
6.2 DAC 平坦度.....	31
7. AD/DA 信号串扰测试.....	32
7.1 ADC 串扰测试.....	32
7.2 DAC 串扰测试.....	33
8. 其他 ADC 指标测试.....	34
8.1 无杂散动态范围 SFDR.....	34
8.2 信噪比 SNR.....	36
8.3 信噪失真比 SNDR.....	36
8.4 有效位数 ENOB.....	37

# 1. 概述

FACE-RFSOC-A 自适应射频开发平台，是 FACE 系列新一代的产品。平台搭载有 16nm 工艺的 Zynq® UltraScale+™ RFSoc 系列主器件。该器件集成数千兆采样 RF 数据转换器和 ARM® Cortex®-A53 处理子系统和 UltraScale+可编程逻辑，是业界唯一单芯片自适应射频平台。

板卡是全球首款标准 MATX 版型的 RFSOC 平台，平台板载有丰富计算资源的同时还具备有非常丰富的外设接口：QSFP28、U.2 NVMe、以太网、USB3.0 等。该平台是无线电、雷达等高端射频应用场景下的理想开发/部署平台。



## FACE-RFSOC-A

高性能自适应射频开发平台

### 特点

- Zynq UltraScale+ XCZU27DR器件 (可选 XCZU28DR、XCZU43DR、XCZU47DR)
- RF部分
  - 8路 4.096GSPS 12bit ADC
  - 8路 6.554GSPS 14bit DAC
- ZU器件接口外设
  - Dual QSPI Flash
  - 4GB DDR4存储器
  - QSFP28接口2个
  - U.2 NVME SSD接口2个
  - SI5341可配置时钟源

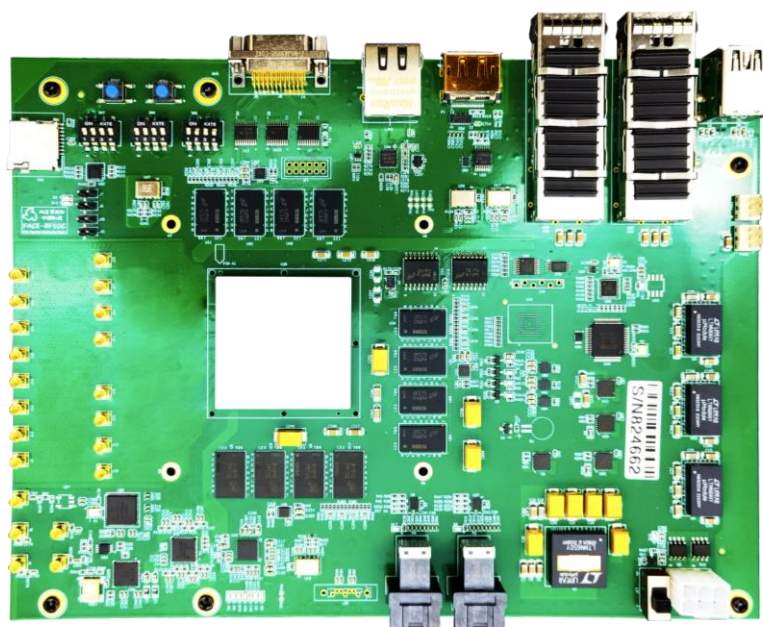


RF Analyzer 是 XILINX 提供的一套上位机软件，它为用户提供了一种简单快速的方法来评估 Zynq® UltraScale+™ RFSoc 中 RF-ADC 和 RFDAC 的性能的方法。

在本文档中，我们将会全面介绍 FACE-RFSOC-A 板卡各个接口外设的测试，并且使用 XILINX RF Analyzer 软件完成对 FACE-RFSOC-A 板卡射频部分的测试，包括板卡 ADC/DAC 配置以及 MTS 配置等，并对 ADC 和 DAC 平坦度、同步性能以及通道间的串扰情况进行测试。

## 2. FACE-RFSOC-A 自适应射频开发平台

### 2.1 平台实物照片



### 2.2 平台特性与规格

- 主芯片
- 默认装载 XCZU27DR

还支持的主器件见下图。

# 测试报告

Device Name	ZU21DR	ZU25DR	ZU27DR	ZU28DR	ZU29DR	ZU39DR	ZU42DR	ZU43DR	ZU46DR	ZU47DR	ZU48DR	ZU49DR
	Gen 1				Gen 2			Gen 3				
<b>RF Data Converter</b> Quad-core Arm® Cortex®-A53 MPCore™ up to 1.3GHz, Dual-core Arm Cortex-R5F MP Core up to 533MHz												
12-bit RF-ADC w/DDC	# of ADCs	0	8	8	8	16	16	-	-	-	-	-
	Max Rate (GSPS)	0	4.096	4.096	4.096	2.058	2.220	-	-	-	-	-
14-bit RF-ADC w/DDC	# of ADCs	-	-	-	-	-	-	8	2	4	8	4
	Max Rate (GSPS)	-	-	-	-	-	-	2.5	5.0	5.0	2.5	5.0
14-bit RF-DAC w/DUC	# of DACs	0	8	8	8	16	16	8	4	12	8	8
	Max Rate (GSPS)	0	6.554	6.554	6.554	6.554	6.554	10.0	10.0	10.0	10.0	10.0
	SD-FEC	8	0	0	8	0	0	0	0	8	0	8
	Number of DDCs per RF-ADC <sup>(1)</sup>	0	1	1	1	1	1	1	2	1	1	1
	RF Input Freq max. GHz		4			5			6			
	Decimation / Interpolation		1x, 2x, 4x, 8x			1x, 2x, 4x, 8x			1x, 2x, 3x, 4x, 5x, 6x, 8x, 10x			12x, 16x, 20x, 24x, 40x
	System Logic Cells (K)	930	678	930	930	930	930	489	930	930	930	930
	CLB LUTs (K)	425	310	425	425	425	425	224	425	425	425	425
	Max. Dist. RAM (Mb)	13.0	9.6	13.0	13.0	13.0	13.0	6.8	13.0	13.0	13.0	13.0
	Total Block RAM (Mb)	38.0	27.8	38.0	38.0	38.0	38.0	22.8	38.0	38.0	38.0	38.0
	UltraRAM (Mb)	22.5	13.5	22.5	22.5	22.5	22.5	45.0	22.5	22.5	22.5	22.5
	DSP Slices	4,272	3,145	4,272	4,272	4,272	4,272	1,872	4,272	4,272	4,272	4,272
	GTy Transceivers	16	8	16	16	16	16	8	16	16	16	16
	PCIe® Gen3 x16	2	1	2	2	2	2	-	-	-	-	-
	PCIeGen3 x16/Gen4 x8 / CCIX <sup>(2)</sup>	-	-	-	-	-	-	0	2	2	2	2
	150G Interlaken	1	1	1	1	1	1	0	1	1	1	1
	100G Ethernet MAC/PCS w/RS-FEC	2	1	2	2	2	2	0	2	2	2	2
	System Monitor	1	1	1	1	1	1	1	1	1	1	1
	Speed Grades	-1E, -1I, -1U, -2E, -2I, -2U, -2U	-1E, -1I, -1U, -2E, -2I, -2U, -2U	-1E, -1I, -1U, -2E, -2I, -2U, -2U	-1E, -1I, -1U, -2E, -2I, -2U, -2U	-1E, -1I, -1U, -2E, -2I, -2U, -2U	-2I, -2U	-1E, -1I, -1U, -2E, -2I, -2U	-1E, -1I, -1U, -2E, -2I, -2U	-1E, -1I, -1U, -2E, -2I, -2U	-1E, -1I, -1U, -2E, -2I, -2U	-1E, -1I, -1U, -2E, -2I, -2U
<b>Programmable Logic (PL)</b>												
Package Footprint	Package Dimensions	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC	PSIO, HDIO, HPFO GTR, GTY RF-ADC, RF-DAC
D1156	35x35	214, 72, 208 4, 16 0, 0										
E1156	35x35		214, 48, 104 4, 8 8, 8	214, 48, 104 4, 8 8, 8	214, 48, 104 4, 8 8, 8			214, 24, 128 4, 8 10, 8	214, 48, 104 4, 8 4, 4		214, 48, 104 4, 8 8, 8	214, 48, 104 4, 8 8, 8
G1517	40x40		214, 48, 299 4, 8 8, 8	214, 48, 299 4, 16 8, 8	214, 48, 299 4, 16 8, 8				214, 48, 299 4, 16 4, 4		214, 48, 299 4, 16 8, 8	214, 48, 299 4, 16 8, 8
F1760	42.5x42.5					214, 96, 312 4, 16 16, 16	214, 96, 312 4, 16 16, 16					214, 96, 312 4, 16 16, 16
H1760	42.5x42.5								214, 48, 312 4, 16 12, 12			

## RF 部分

- 8 路 4.096GSPS 12bit ADC
- 8 路 6.554GSPS 14bit DAC

## ZU 器件 PS 侧接口外设

- 内存: DDR4 4GB 64bit 2400MT/s
- 配置存储: Dual QSPI Flash
- SD 卡接口
- 以太网接口
- DP 显示接口
- USB 接口
- USB-UART 接口
- MIO 扩展接口

## ZU 器件 PL 侧接口外设

## 测试报告

- 内存：2 组 DDR4，每组 4GB@64bit 2666MT/s
- QSFP28 接口 2 个
- 2 个 U.2 NVME SSD 接口（PCIe Gen3x4）
- 拨码开关 x8，LEDx8
- GPIO（J30J 连接器）
- SI5341 可配置时钟源
- USB-JTAG 接口

### 3. 平台接口外设测试

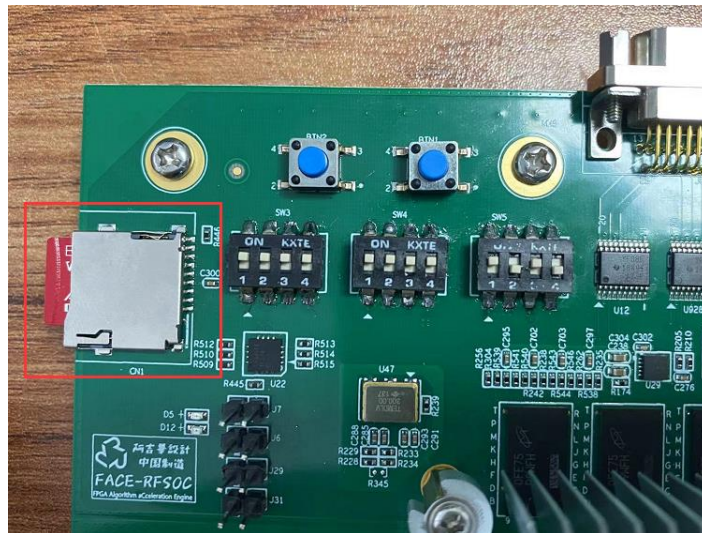
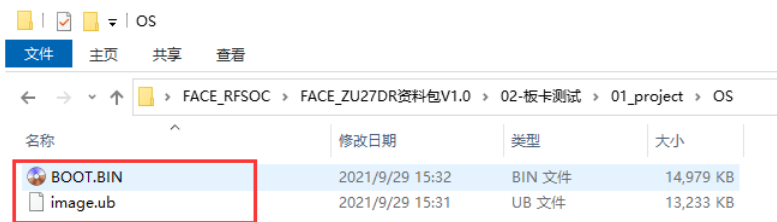
FACE-RFSOC-A 板卡外设接口外设（除 AD/DA 以外）主要包含：

1. ZU ARM PS 侧的：USB 接口、USB-UART 接口、以太网接口、SD 卡接口、DP 显示接口、QSPI Flash 存储、PS DDR4 存储等；
2. ZU PL 逻辑侧的：QSFP28 100G 高速光接口、PL DDR4 存储、J30J GPIO 扩展接口、U.2 NVME SSD 接口、拨码开关与 LED 灯等；

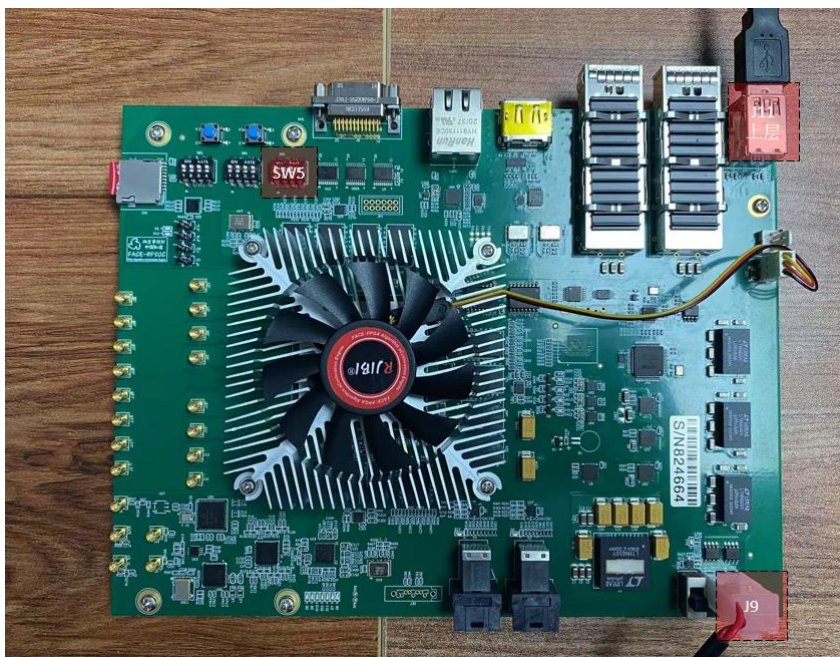
在本章节中，将通过若干 FPGA 工程来对上述接口外设进行测试，其中操作系统测试工程将覆盖上述大部分接口外设的功能。

#### 3.1 测试准备

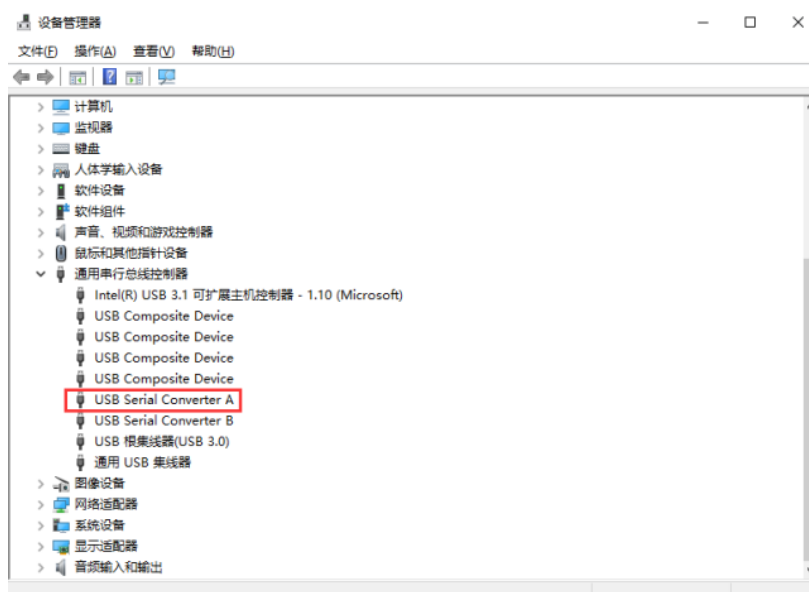
将阿吉毕科技提供的资料包 OS 文件夹中的文件复制到文件系统为 FAT32 的 SD 卡中，然后将 SD 卡插入板卡 SD 卡插槽，如下图。



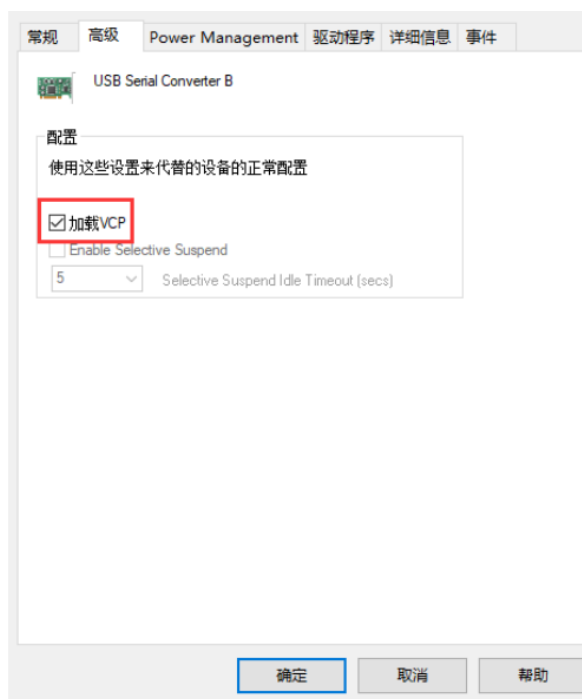
使用阿吉毕科技提供的 12V 电源适配器连接板卡 J9 电源连接器，使用 USB-TypeA 线连接板卡 J11 双层 USB 连接器的上层接口和 PC 上位机的 USB 接口。通过拨码开关 SW5 来设置板卡启动方式为 SD 卡启动（ON 为 0，OFF 为 1。SW5=4'b1010）。最后打开板卡 J9 电源连接器旁边的电源开关对板卡进行上电。



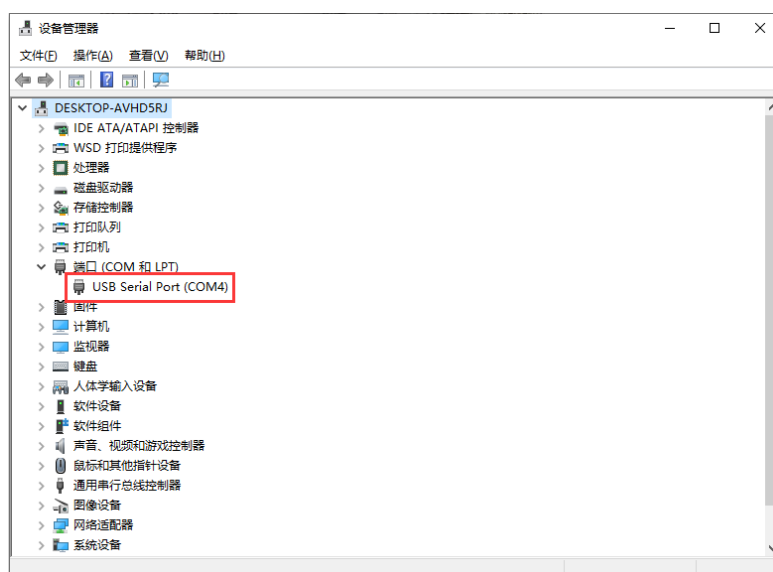
为了确保板板的 USB-JTAG/UART 驱动可以被 Windows 操作系统正确识别，请在上位机打开设备管理器中选中 USB Serial Converter A，右键属性，高级界面中勾选 VCP 后点击确定。对 USB Serial Converter B 也进行同样操作。



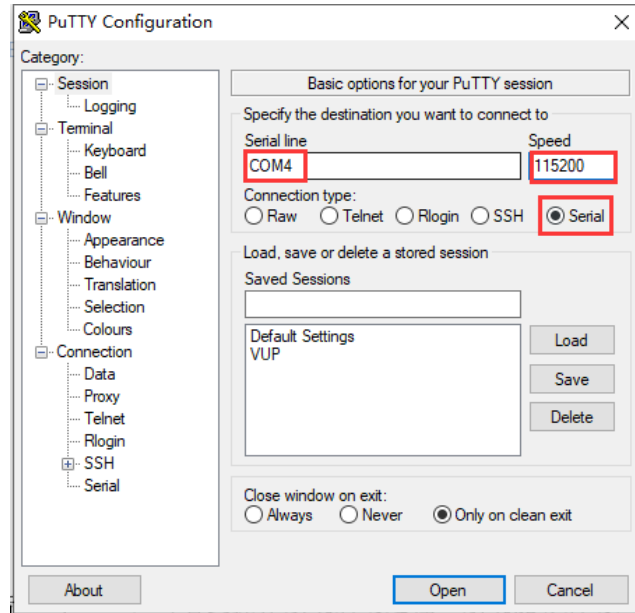




随后将 USB 线缆重新插拔或板卡重新上电，设备管理器即可正常搜索到串口。  
如，在设备管理器查串口号为 COM4（此时若看到系统识别到有两个 COM 号，  
UART 串口为数字大的那一个）。



打开 PUTTY 或其他第三方串口工具，设置波特率为 115200 如下图所示。



点击“Open”，即可打开板卡串口终端。

若此时正确安装了 SD 卡，则可看到板卡操作系统启动过程。

## 3.2 接口外设测试说明

### 3.2.1 PS DDR4 测试

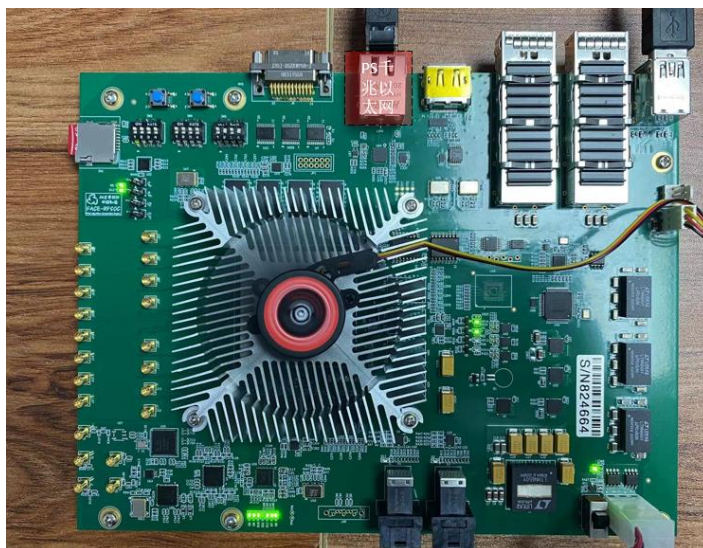
在串口终端进入操作系统后，输入登录用户名：**root**，密码：**root**。

操作系统的正常启动即可说明板卡 PS DDR4 的功能正常。

### 3.2.2 PS 千兆以太网测试

系统启动后使用网线连接 RFSOC 板卡 PS 千兆网口与 PC 或路由器/交换机，在串口终端输入 `ifconfig` 命令，查看被分配到的 IP。

## 测试报告



```
COM4 - PuTTY
root@face_zul9eg:~# ifconfig[ 183.954101] macb ff0e0000.ethernet eth0: link up
(1000/Full)
[ 183.959774] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready

eth0      Link encap:Ethernet  HWaddr 00:0A:35:00:22:01
          inet addr:192.168.1.144  Bcast:0.0.0.0  Mask:255.255.255.0
          inet6 addr: fe80::20a:35ff:fe00:2201/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:276 (276.0 B)  TX bytes:976 (976.0 B)
          Interrupt:30

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

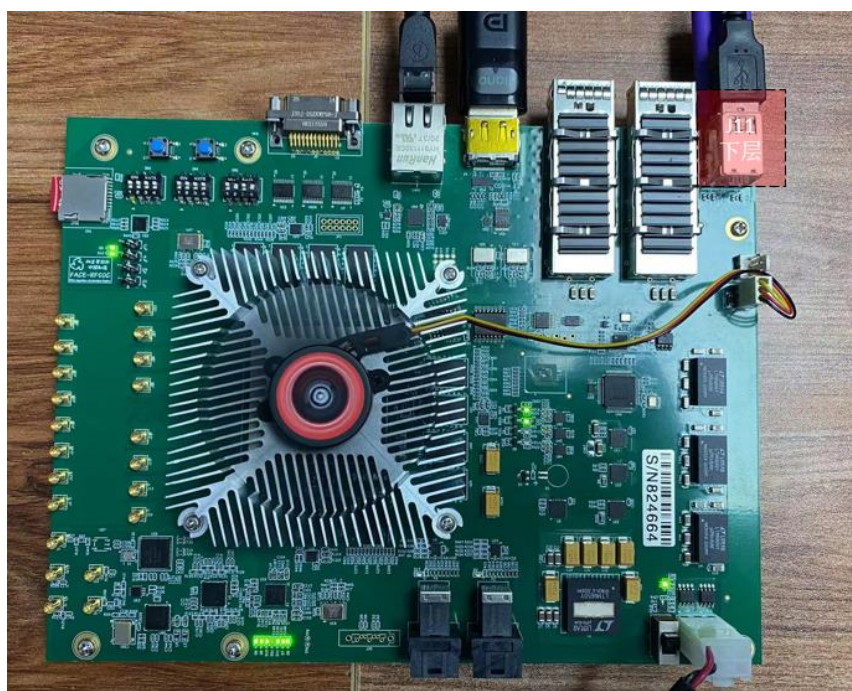
root@face_zul9eg:~#
```

在串口终端使用 PING 命令测试网络连通性。若结果如下图所示。表示 PS 千兆以太网口接口测试 OK。

```
COM4 - PuTTY
root@face_zul9eg:~# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1): 56 data bytes
64 bytes from 192.168.1.1: seq=0 ttl=64 time=0.686 ms
64 bytes from 192.168.1.1: seq=1 ttl=64 time=0.428 ms
64 bytes from 192.168.1.1: seq=2 ttl=64 time=0.422 ms
64 bytes from 192.168.1.1: seq=3 ttl=64 time=0.422 ms
64 bytes from 192.168.1.1: seq=4 ttl=64 time=0.417 ms
64 bytes from 192.168.1.1: seq=5 ttl=64 time=0.434 ms
64 bytes from 192.168.1.1: seq=6 ttl=64 time=0.417 ms
64 bytes from 192.168.1.1: seq=7 ttl=64 time=0.420 ms
64 bytes from 192.168.1.1: seq=8 ttl=64 time=0.445 ms
64 bytes from 192.168.1.1: seq=9 ttl=64 time=0.414 ms
64 bytes from 192.168.1.1: seq=10 ttl=64 time=0.445 ms
64 bytes from 192.168.1.1: seq=11 ttl=64 time=0.400 ms
64 bytes from 192.168.1.1: seq=12 ttl=64 time=0.420 ms
64 bytes from 192.168.1.1: seq=13 ttl=64 time=0.445 ms
64 bytes from 192.168.1.1: seq=14 ttl=64 time=0.389 ms
64 bytes from 192.168.1.1: seq=15 ttl=64 time=0.408 ms
64 bytes from 192.168.1.1: seq=16 ttl=64 time=0.413 ms
64 bytes from 192.168.1.1: seq=17 ttl=64 time=0.407 ms
64 bytes from 192.168.1.1: seq=18 ttl=64 time=0.433 ms
64 bytes from 192.168.1.1: seq=19 ttl=64 time=0.416 ms
^Z
[1]+  Stopped(SIGTSTP)                  ping 192.168.1.1
```

### 3.2.3 PS USB 接口测试

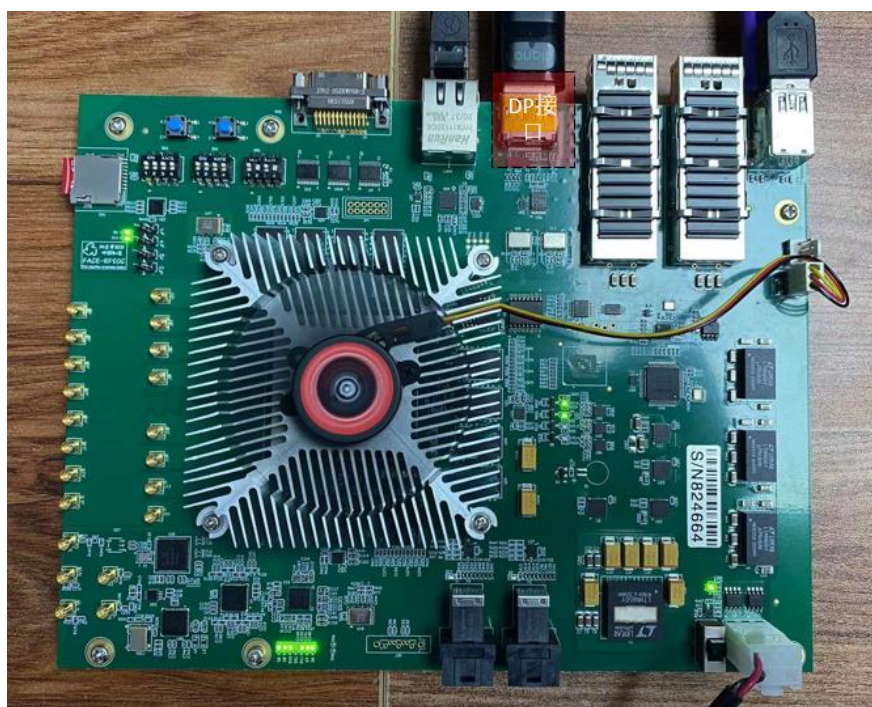
将 U 盘插入板卡 USB 口（J11 下层），可在串口终端内看到识别 U 盘的输出信息。表示 USB 接口测试 OK。



```
root@Z7020TEST:~# usb 1-1.1: new high-speed USB device number 3 using ci_hdrc
usb-storage 1-1.1:1.0: USB Mass Storage device detected
scsi host0: usb-storage 1-1.1:1.0
scsi 0:0:0:0: Direct-Access          USB DISK 3.0      DL17 PQ: 0 ANSI: 6
sd 0:0:0:0: Attached scsi generic sg0 type 0
sd 0:0:0:0: [sda] 60532992 512-byte logical blocks: (31.0 GB/28.9 GiB)
sd 0:0:0:0: [sda] Write Protect is off
sd 0:0:0:0: [sda] Write cache: disabled, read cache: enabled, doesn't support D
0 or FUA
sda: sda1
sd 0:0:0:0: [sda] Write Protect is on
sd 0:0:0:0: [sda] Attached SCSI removable disk
```

### 3.2.4 PS DP 显示接口测试

使用 DP 线连接板卡 DP 口和显示器。在终端进入 root 权限后输入如下命令。

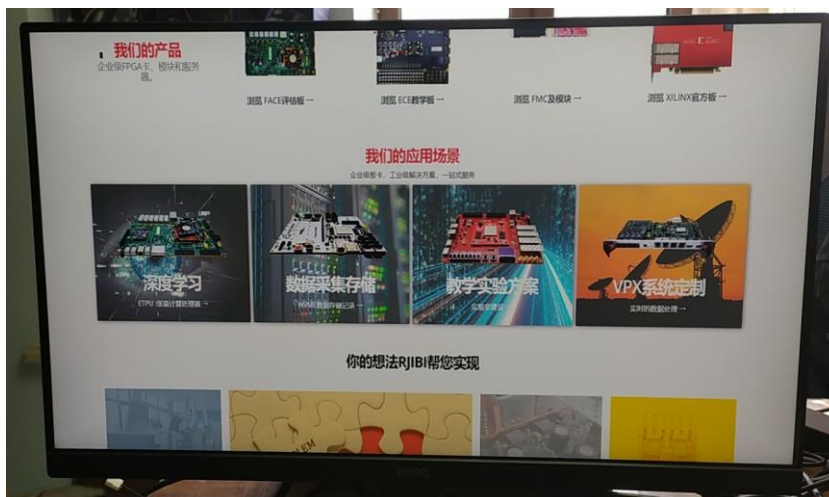


1. mount /dev/mmcblk0p1 /mnt
2. cd /mnt
3. chmod 777 test.elf
4. ./test.elf face.bmp

```
COM4 - PuTTY
PetaLinux 2019.2 zul9egdeskvl /dev/ttyPS0

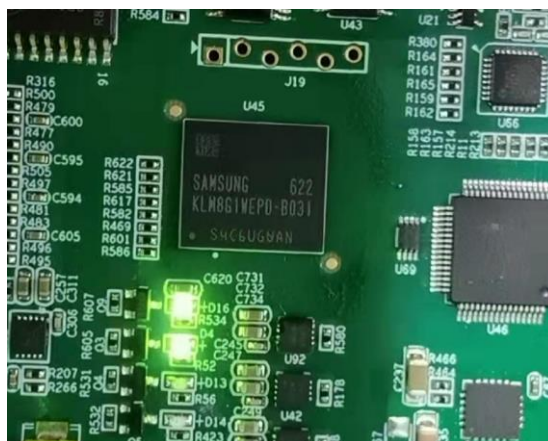
zul9egdeskvl login: root
Password:
root@zul9egdeskvl:~# mount /dev/mmcblk0p1 /mnt/
root@zul9egdeskvl:~# cd /mnt/
root@zul9egdeskvl:/mnt# chmod 777 test.elf
root@zul9egdeskvl:/mnt# ./test.elf face.bmp
img name = face.bmp
fb info x[1920] y[1080] x_v[1920] y_v[2160] xoffset[0] yoffset[0] bpp[16] line_1
ength[3840] size[8294400]
fb info red off[11] len[5] msb[0]
fb info green off[5] len[6] msb[0]
fb info blue off[0] len[5] msb[0]
img info w[1920] h[1080] bpp[24] size[6220854] offset[54]
img len_one_line = 5760
fb_len_one_line = 3840
root@zul9egdeskvl:/mnt#
```

该预置的测试软件将会在 DP 显示屏上输出 SD 卡中的图片，显示器有对应的图像输出则表示 DP 接口测试 OK;

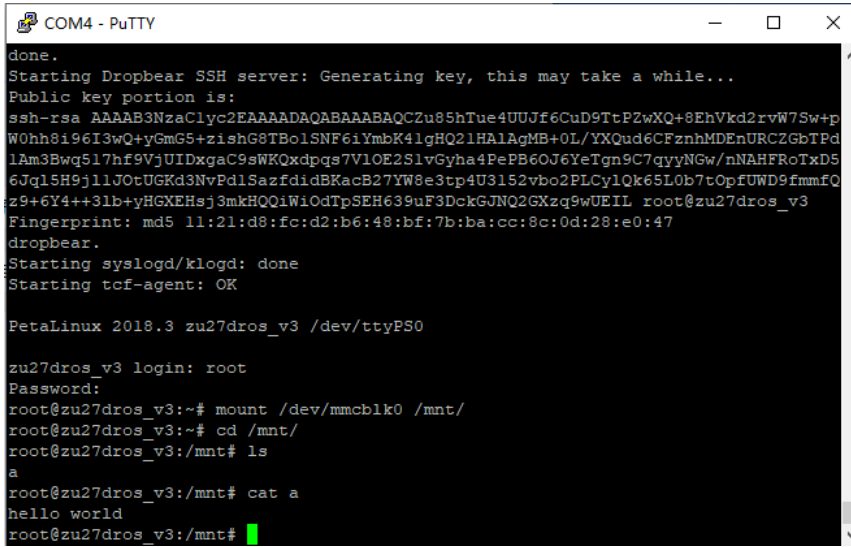


### 3.2.5 PS EMMC 测试

板卡板载 8GB EMMC 存储器件，



进入操作系统后，可按 Linux 命令对 EMMC 存储进行格式化等相应操作，可以在 EMMC 存储区域创建文件后关机断电重启后打开文件进行验证。



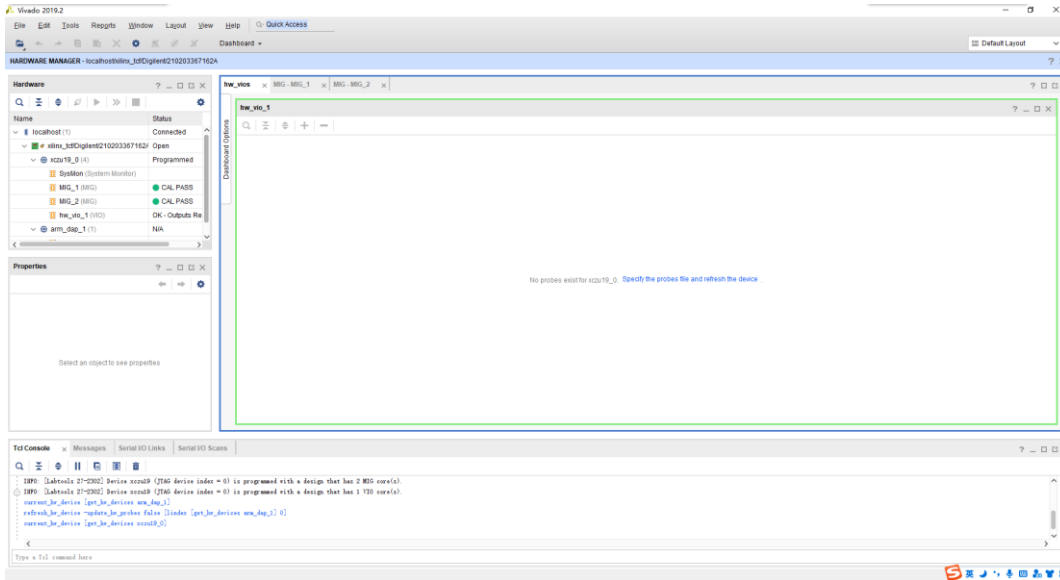
```
COM4 - PuTTY
done.
Starting Dropbear SSH server: Generating key, this may take a while...
Public key portion is:
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCAZu85hTue4UUJf6CuD9TtPZwXQ+8EhVkd2rvW7Sw+p
W0hh8i96I3wQ+yGmG5+ziShG8TBo1SNF6iYmbK41gHQ21HAlAgMB+0L/YXQuD6CFznhMDEnURCZGbtPd
lAm3Bwq517hf9VjUIDXgaC9sWKQxdpqs7V1OE2S1vGyha4PePB6OJ6YeTgn9C7qyyNGw/nNAHFRoTxD5
6Jq15H9j1lJ0tUGKd3NvPd1SazfdidBKacB27YW8e3tp4U3152vbo2PLCy1Qk65L0b7tOpfUWD9fmmfQ
z9+6Y4++31b+yHGXEhsj3mkHQQiWiOdTpSEH639uF3DckGJNQ2GXzq9wUEIL root@zu27dros_v3
Fingerprint: md5 11:21:d8:fc:d2:b6:48:bf:7b:ba:cc:8c:0d:28:e0:47
dropbear.
Starting syslogd/klogd: done
Starting tcf-agent: OK

PetaLinux 2018.3 zu27dros_v3 /dev/ttyPS0

zu27dros_v3 login: root
Password:
root@zu27dros_v3:~# mount /dev/mmcblk0 /mnt/
root@zu27dros_v3:~# cd /mnt/
root@zu27dros_v3:/mnt# ls
a
root@zu27dros_v3:/mnt# cat a
hello world
root@zu27dros_v3:/mnt#
```

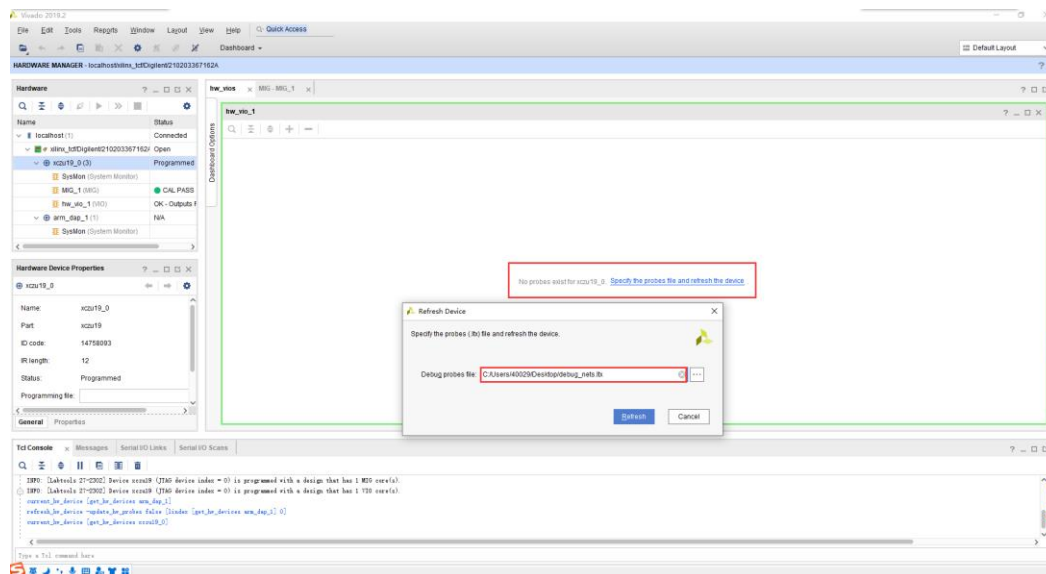
### 3.2.6 PL DDR4 存储器测试

打开 VIVADO 2018.3，选择 Open hardware manager，点击 open target→auto connect。

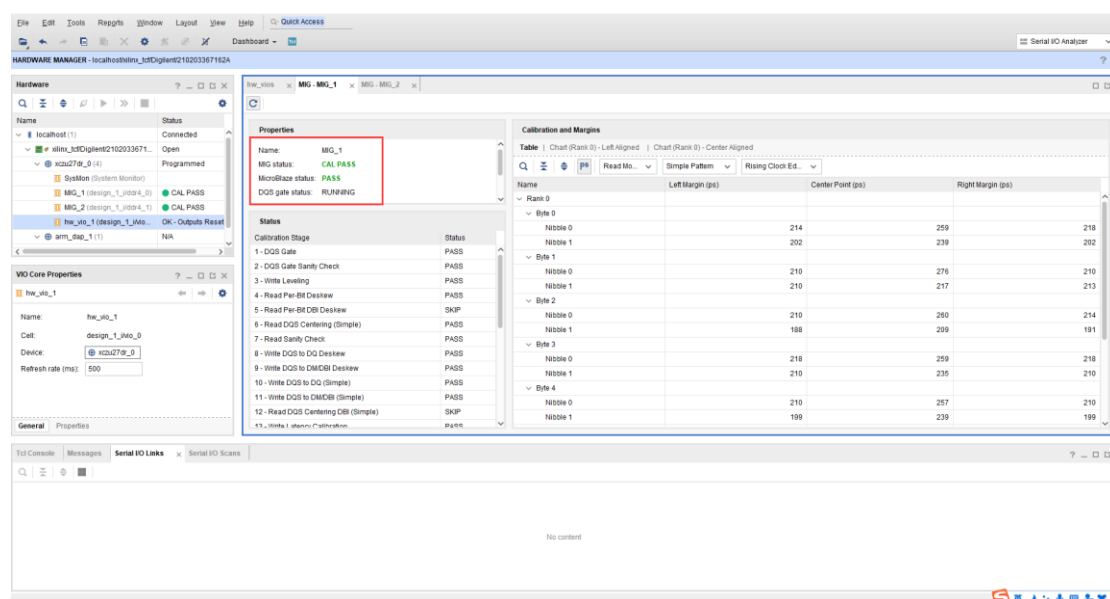


点击 Specify the probes file and refresh the device，浏览并找到阿吉毕科技提供的资料包中的 design\_1\_wrapper.ltx 文件。

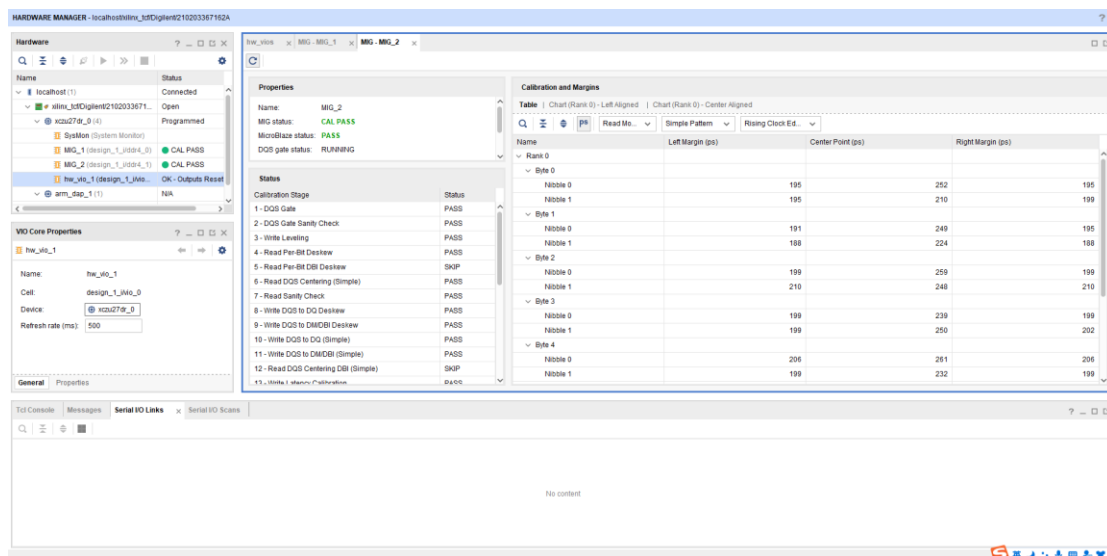
## 测试报告



在 Vivado HardwareManager 中连接 RFSOC 器件，会弹出 MIG 校验界面。校验通过表示两组 PL DDR4 硬件功能 OK。

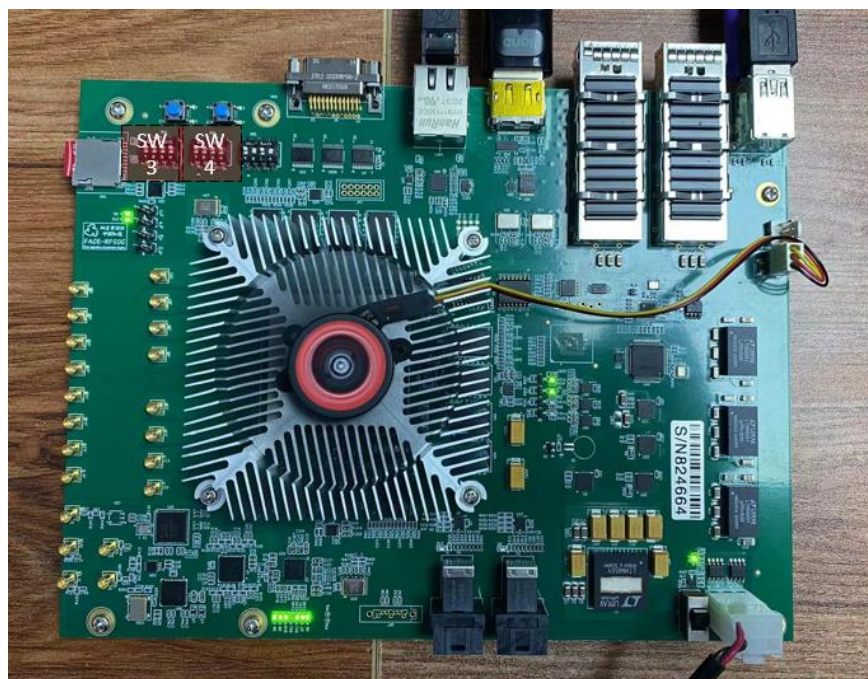






### 3.2.7 PL 用户拨码开关测试

打开 VIVADO 硬件管理器的 VIO 接口，在板卡上拨动拨码开关 SW3、SW4，观察 in1 变化。若 VIO 界面显示与拨码开关拨动状态一致，则拨码开关功能测试 OK。

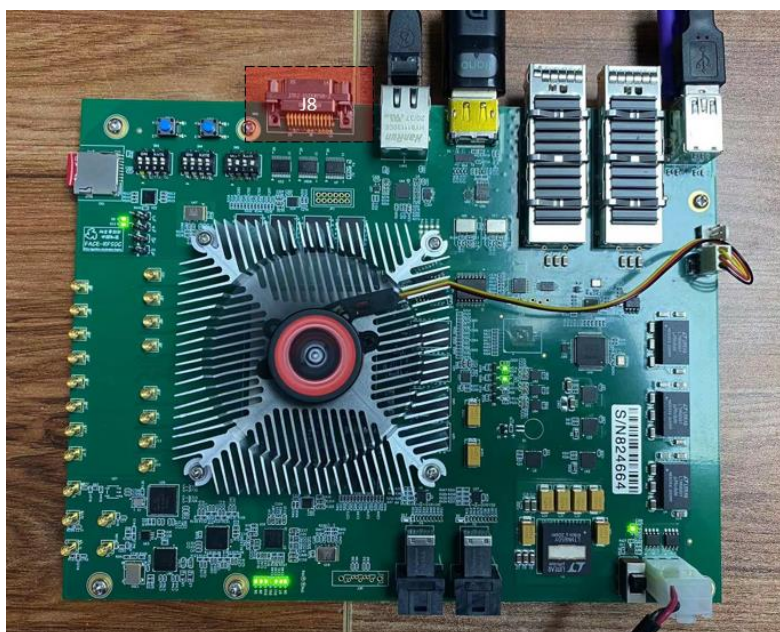


hw\_vio\_1

Name	Value	Acti...	Directi...	VIO
> design_1_i/labtools_fmter_0_F0[31:0]	[H] 0000_0041		Input	hw_vio_1
✓ design_1_i/probe_in1_0_1[7:0]	[H] FF		Input	hw_vio_1
└ design_1_i/probe_in1_0_1[7]	1		Input	hw_vio_1
└ design_1_i/probe_in1_0_1[6]	1		Input	hw_vio_1
└ design_1_i/probe_in1_0_1[5]	1		Input	hw_vio_1
└ design_1_i/probe_in1_0_1[4]	1		Input	hw_vio_1
└ design_1_i/probe_in1_0_1[3]	1		Input	hw_vio_1
└ design_1_i/probe_in1_0_1[2]	1		Input	hw_vio_1
└ design_1_i/probe_in1_0_1[1]	1		Input	hw_vio_1
└ design_1_i/probe_in1_0_1[0]	1		Input	hw_vio_1
> design_1_i/vio_0_probe_out0[7:0]	[H] 00	▼	Output	hw_vio_1
> design_1_i/vio_0_probe_out1[19:0]	[H] 0_0000	▼	Output	hw_vio_1

### 3.2.8 PL 用户 J30J GPIO 测试

在 VIO 界面中分别设置 out1 为 20hFFFFFF 或者 16'h00000。测试 J30J GPIO 连接器的 20 路 GPIO。若对应接口电压为 3.3V 或者 0V，表示对应 GPIO 信号测试 OK。

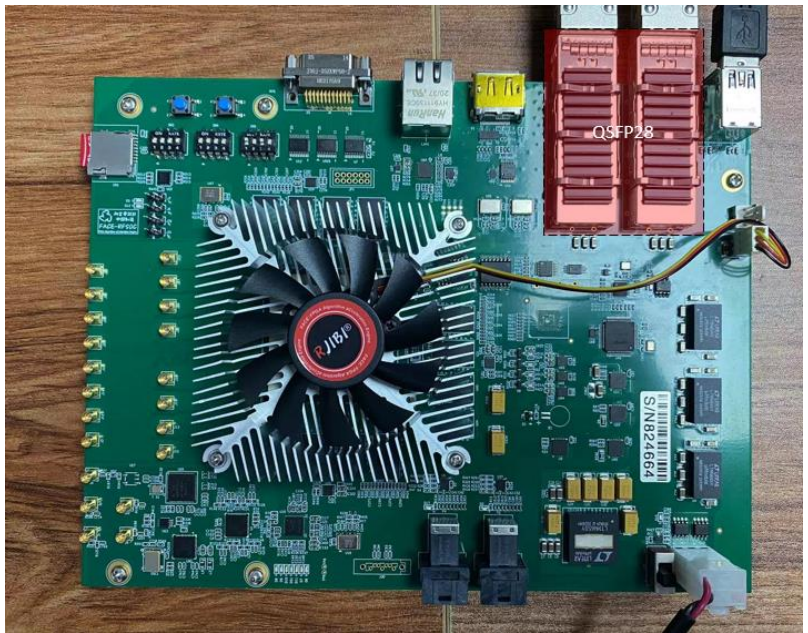


Name	Value	Acti...	Direct...	VIO
design_1_iMio_0_probe_out0[6]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out0[5]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out0[4]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out0[3]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out0[2]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out0[1]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out0[0]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[19:0]	[H] 0_0000	Output	hw_vio_1	
design_1_iMio_0_probe_out1[19]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[18]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[17]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[16]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[15]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[14]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[13]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[12]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[11]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[10]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[9]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[8]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[7]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[6]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[5]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[4]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[3]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[2]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[1]	0	Output	hw_vio_1	
design_1_iMio_0_probe_out1[0]	0	Output	hw_vio_1	

### 3.2.9 QSFP28 接口测试

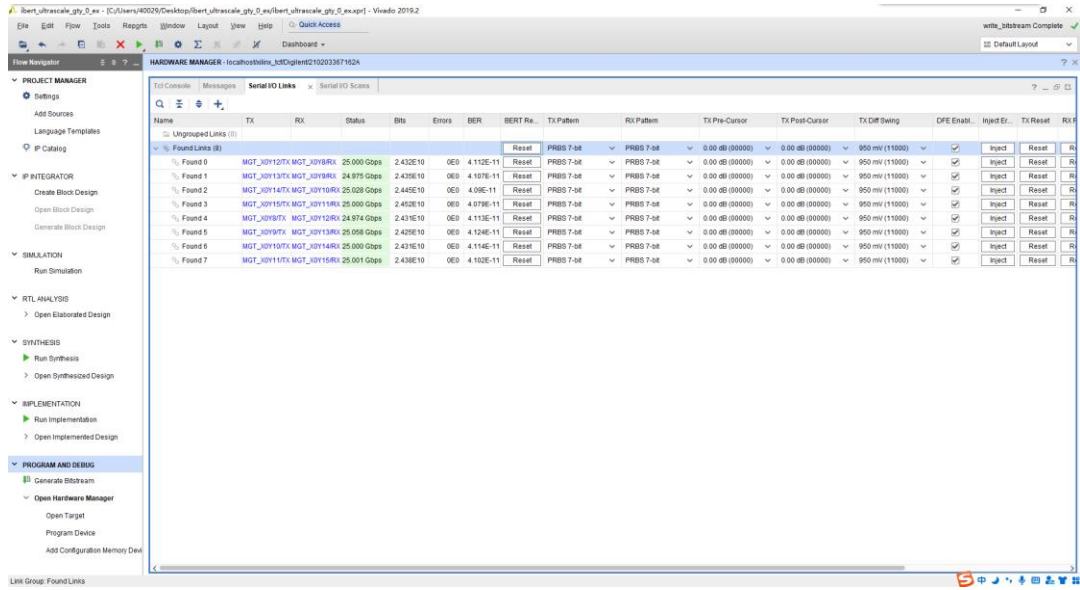
QSFP28 接口的 IBERT 测试需要使用阿吉毕科技提供的资料包中的 IBERT 测试工程来进行测试。

首先将 QSFP 线缆两端分别插入两路 QSFP28 口，如下图所示。然后给板卡上电；



解压并使用 vivado 打开 ibert\_ultrascale\_gty\_0\_ex.xpr.zip 工程，烧写 bitstream 到 FPGA。在 Hardware Manager 中查看各 MGT 通道 ibert 测试情况；

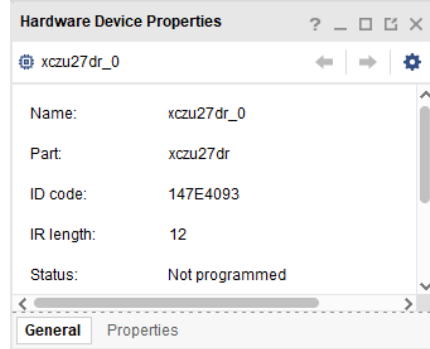
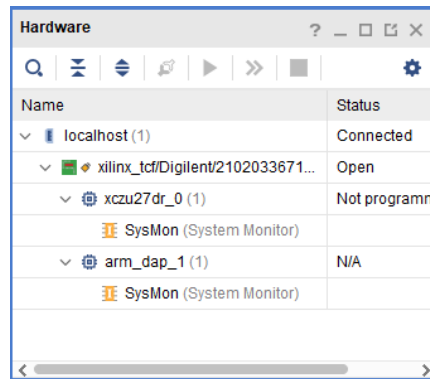
## 测试报告



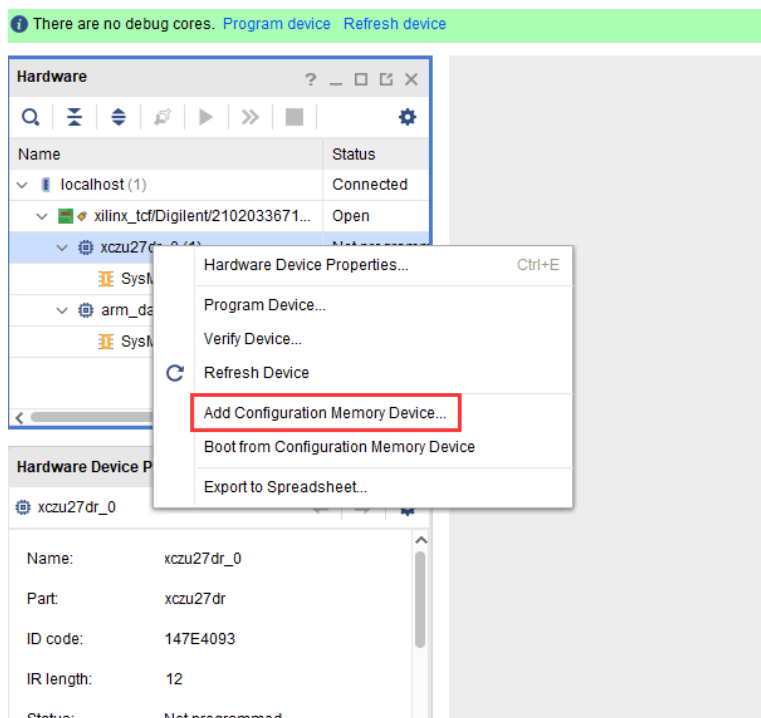
该测试工程为 lane 速率 25Gbps 的 IBERT 测试，若在界面中观测数据链路稳定，传输数据无丢包误码，则 QSFP28 接口测试 OK。

### 3.2.10 QSFP Flash 测试

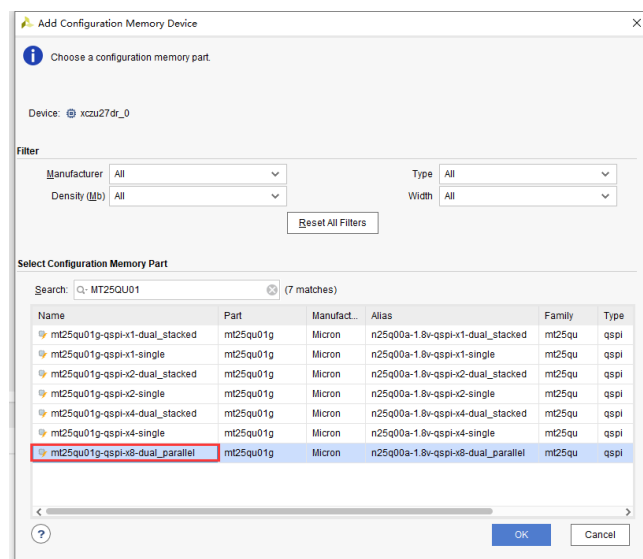
设置板卡启动方式为 JTAG 启动（SW5=4'b0000）。打开 VIVADO2018.3，选择 Open Hardware Manager→Open Target→Auto Connect 连接板卡。



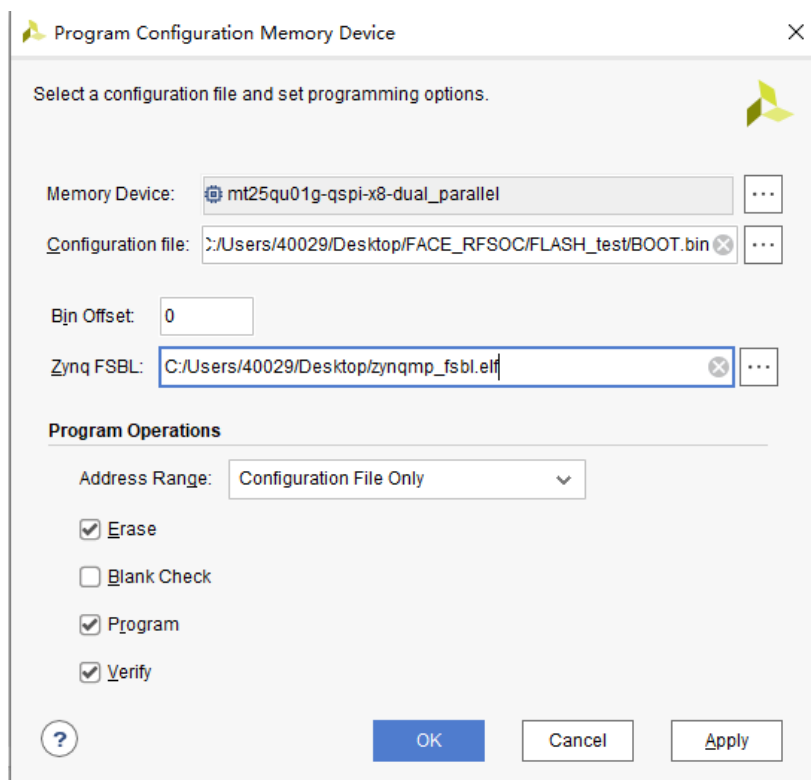
右键 XCZU27DR 并选择 Add Configuration Memory Device。



选择 mt25qu01g-qspi-x8-dual\_parallel 作为配置内存。



点击 OK 编程配置内存。在 Programming Configuration Memory Device 窗口中进入 Configuration File 输入框，选择阿吉毕科技配套资料包中的 BOOT.BIN 文件。转到 ZYNQ FSBL 输入框，浏览并选择 fsbl.elf 文件。检查 Program Configuration Memory Device 窗口中显示的所有其他设置。单击 OK 开始编程配置内存。



配置好内存后，关闭电源，设置板卡启动方式为 QSPI 启动（SW5=4'b0100）。然后打开电源开关。



打开 PUTTY 工具，设置波特率为 115200，可看到终端一直打印“hello world”，表示 QSPI FLASH 测试 OK。。

# 测试报告



The image shows a screenshot of a PuTTY terminal window. The window title is "COM4 - PuTTY". The terminal output consists of 20 lines of "Hello World" text. The text is white on a black background. The window has standard Windows window controls (minimize, maximize, close) in the top right corner. The bottom of the terminal shows the start of a new line with "Hel" and a green cursor.

```
COM4 - PuTTY
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hel
```

## 4. RF Analyzer 软件

### 4.1 RF Analyzer 介绍

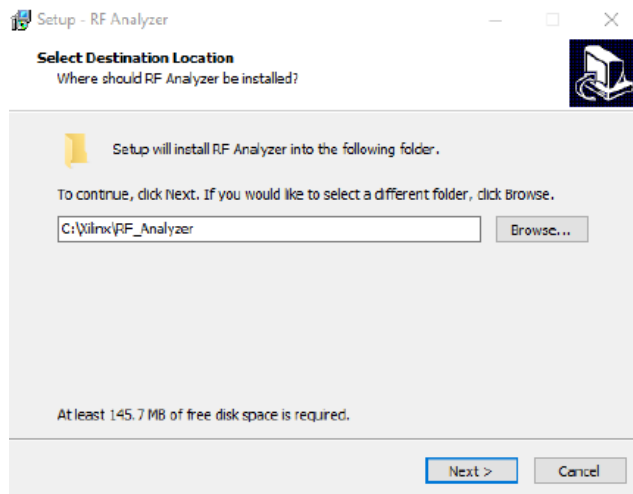
RF Analyzer 提供了一种简单快速的方法来评估 Zynq® UltraScale+™ RFSoc 中 RF-ADC 和 RFDAC 的性能。

RF Analyzer 要求主机上安装 2020.1 硬件服务器（或更高版本）或 Vivado® Design Suite（2020.1 或更高版本）。下载 BIT 流前请根据实际硬件设计确保外部时钟稳定。如果转换器 IP 启动时没有可用的有效时钟，则相应的转换器图块可能会显示错误。

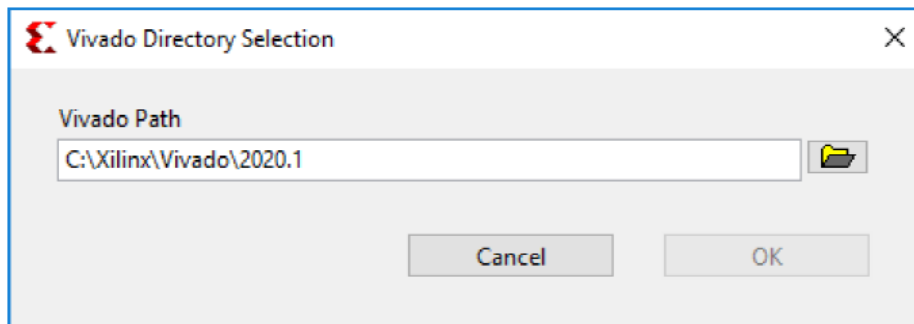
### 4.2 RF Analyzer 安装

双击阿吉毕科技提供的软件安装包：

选择合适的安装路径：



安装后，需要指定 VIVADO 安装路径。

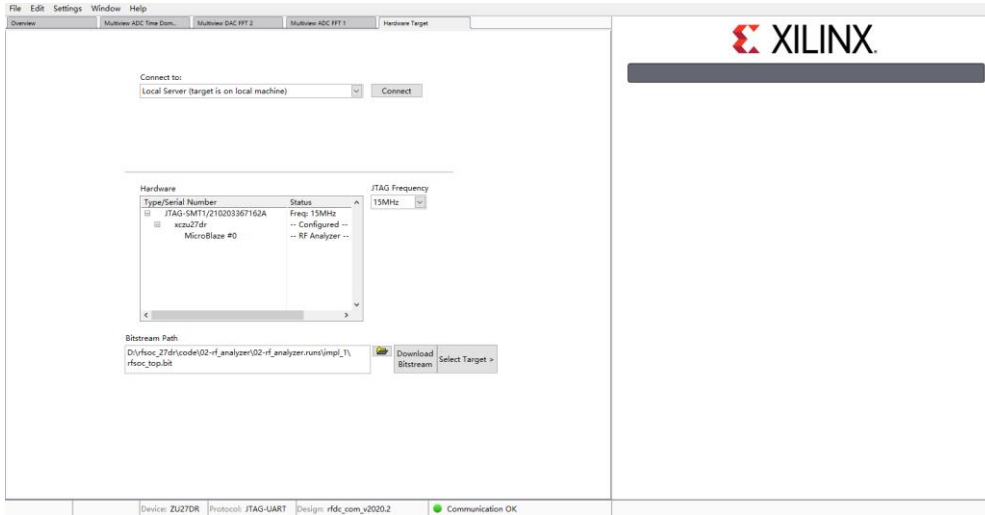




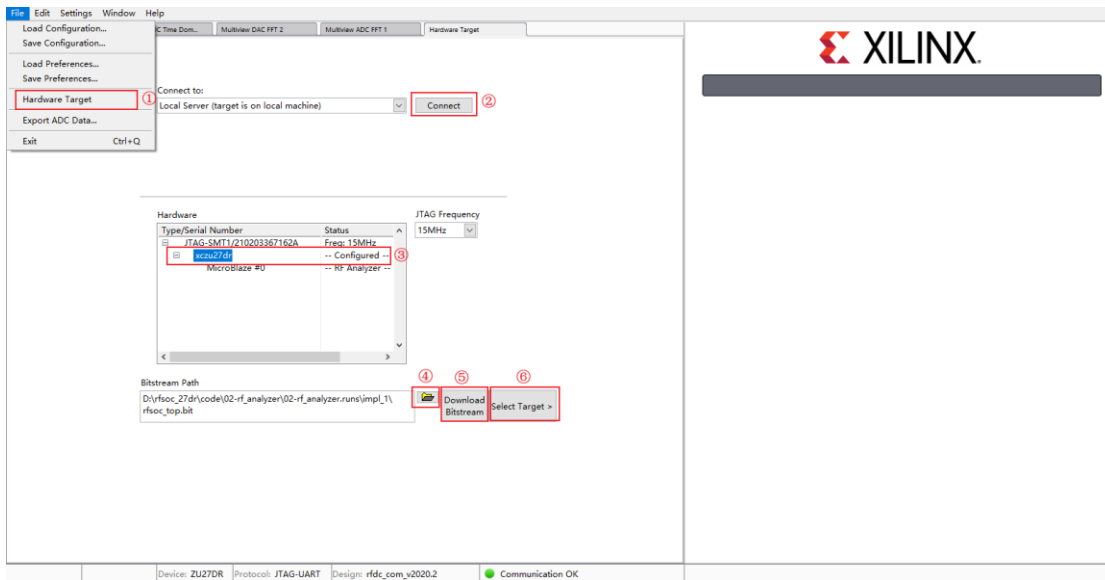
## 5. AD/DA 同步性能测试

### 5.1 测试准备

软件安装完成后，PC 连接板卡 JTAG，打开软件后点击链接，默认会看到下图窗口。

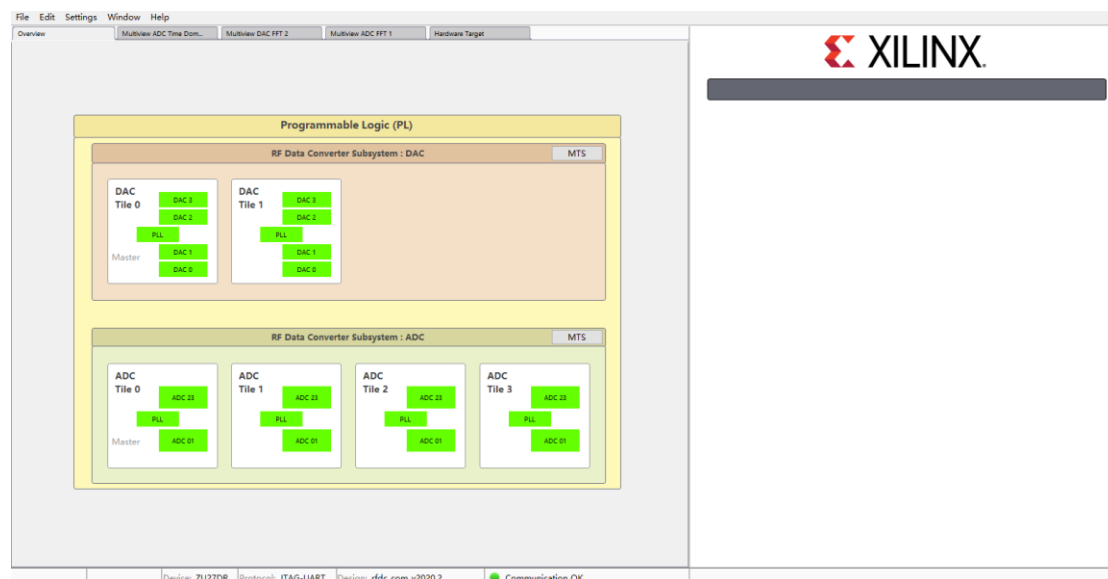


如果没有上图的 HardWare 窗口，点击①HardWare Target 打开 Hardware Target 窗口。

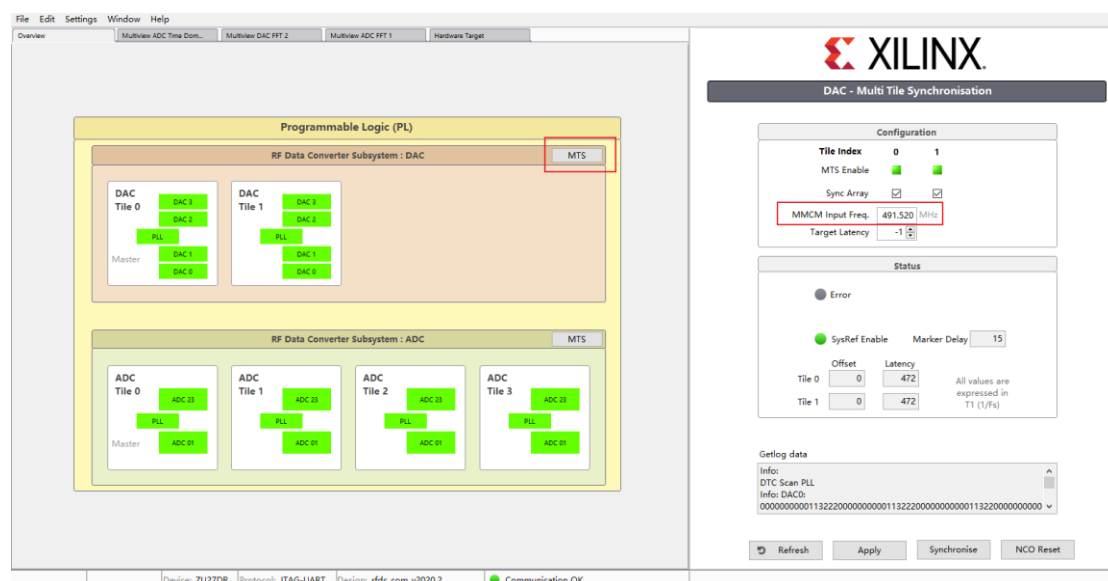


点击②Connect 扫描硬件，点击③选中 FPGA，点击④选择 bit 文件（选择 Rjibi 阿吉毕科技提供的 bit 文件，工程为 rf\_analyzer.xpr.zip），点击⑤下载 bit 文件，点击⑥打开 Overview 窗口。

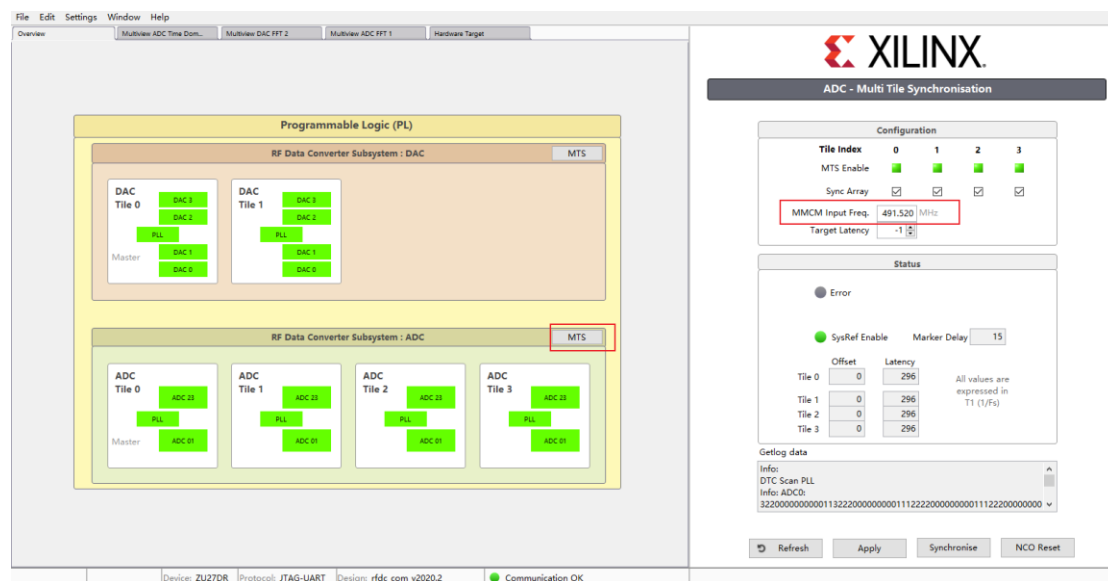
## 测试报告



使用 8 路 ADC/DAC 同步工作需要在软件上进行 MTS 设置。点击下图 MTS 打开 DAC MTS 设置 MMCM Input Freq 输出 491.52。点击 Apply，再点击 Synchronise。Error 如果为红色说明同步失败。



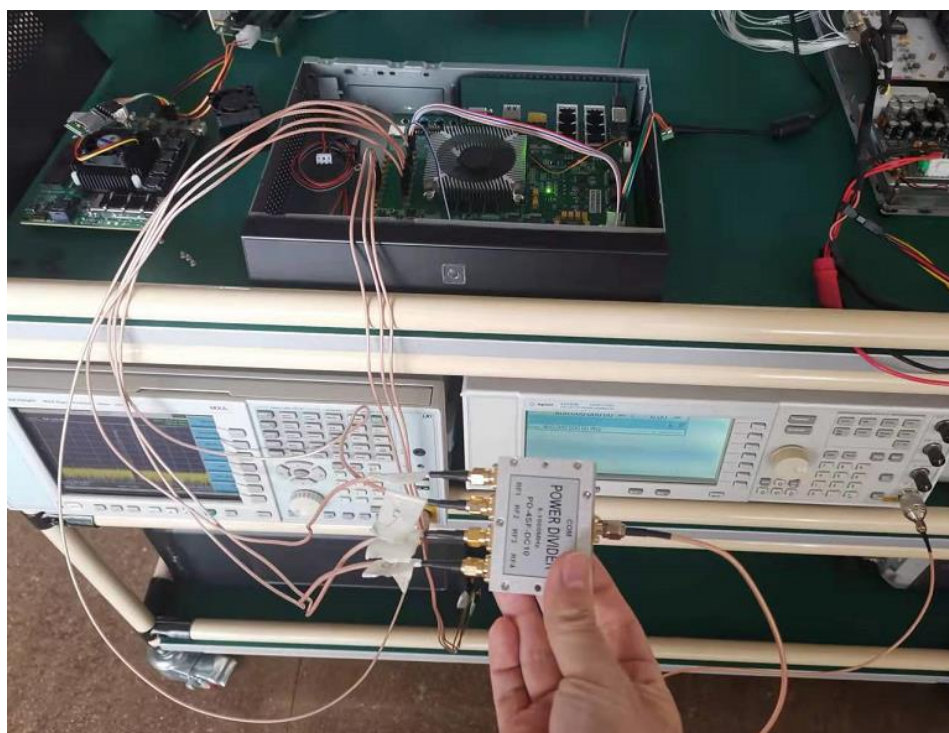
ADC MTS 设置和 DAC MTS 设置操作步骤相同。



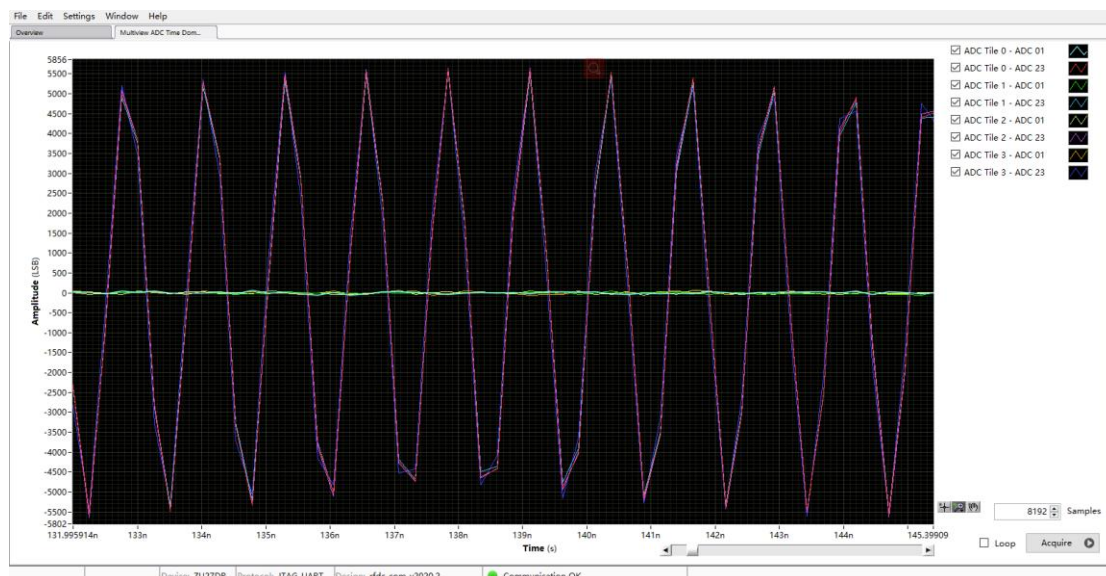
设置完毕后，ADC 和 DAC 则可以同步工作。

## 5.2 ADC 同步采集测试

信号源输出 800Mhz 正弦波，经过功分器转为 4 路，分别输入至 RFSOC 4 个不同的 TILE 中的 ADC 上。

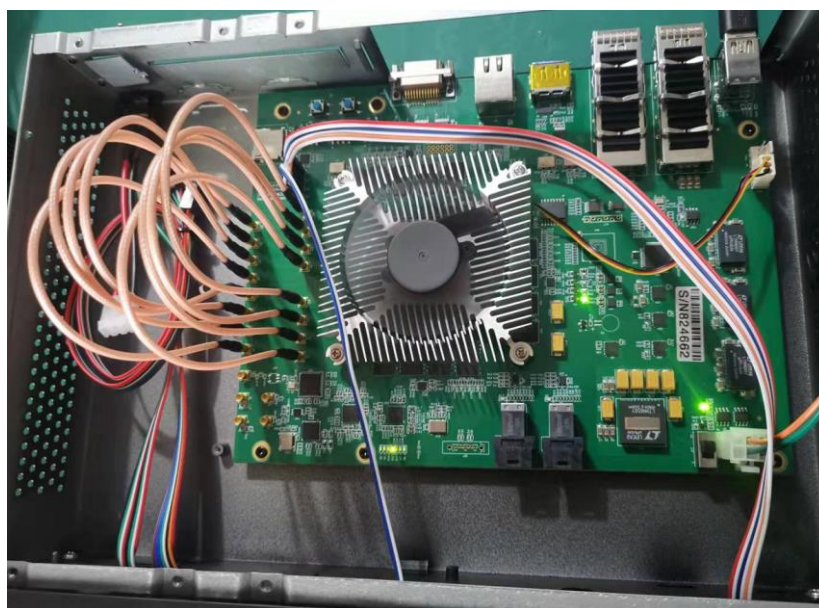


在窗口中可以将 8 路 ADC 的采集波形同时绘出，以便观察信号采集的同步性能。



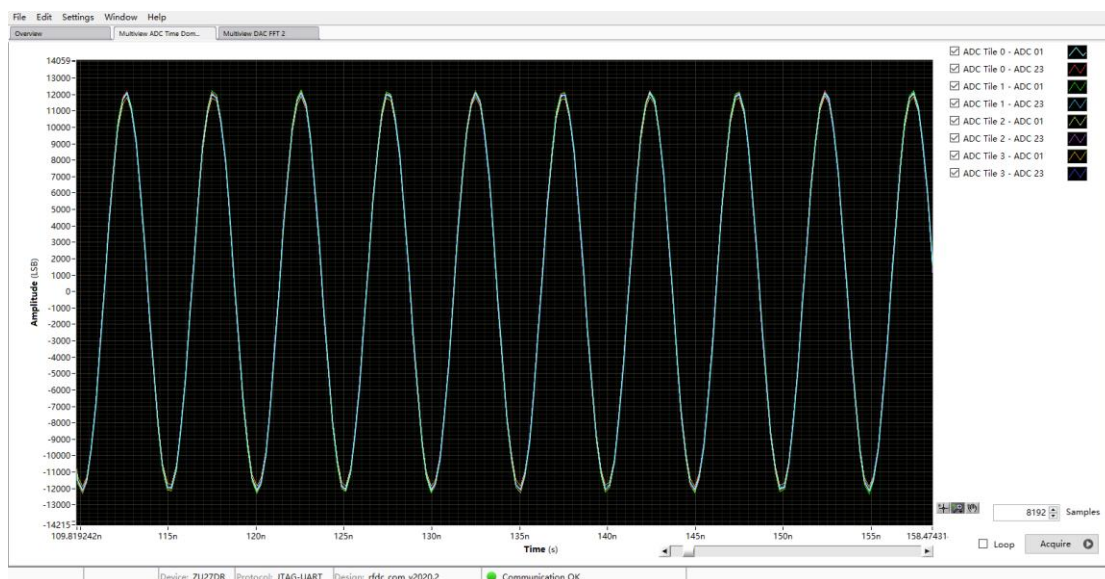
## 5.3 DAC 同步输出测试

RFSOC 设置为 DAC 同步输出 200M 单音正弦波信号，板卡进行回环连接线至 8 路 ADC 进行同步采集。



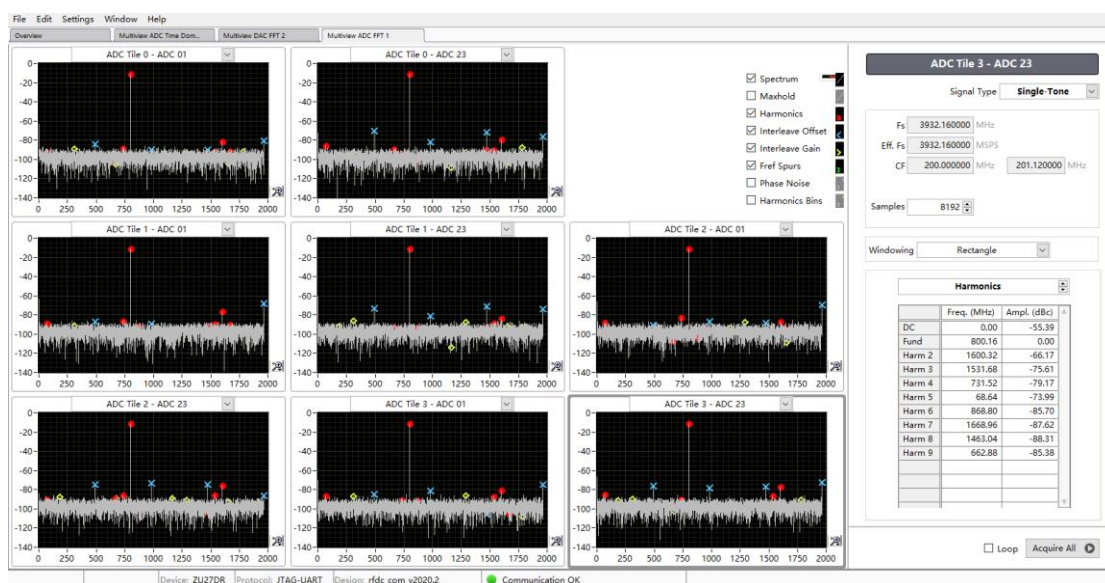
在窗口中可以将 8 路 ADC 的采集波形同时绘出，以便观察信号采集的同步性能。

## 测试报告



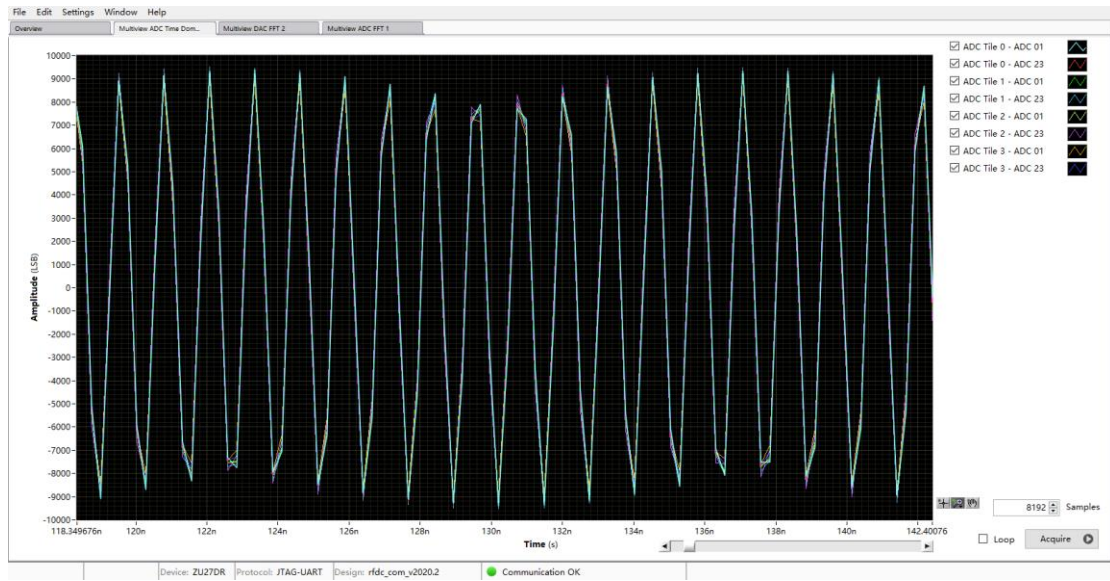
提高 DAC 输出单音正弦波信号至 800MHz，板卡进行回环连接线至 8 路 ADC 进行同步采集。

在软件上可以使用频域显示 8 路 ADC 采集信号对应的 FFT 结果。



在时域窗口中可以将 8 路 ADC 的采集波形同时绘出，以便观察信号采集的同步性能。

# 测试报告

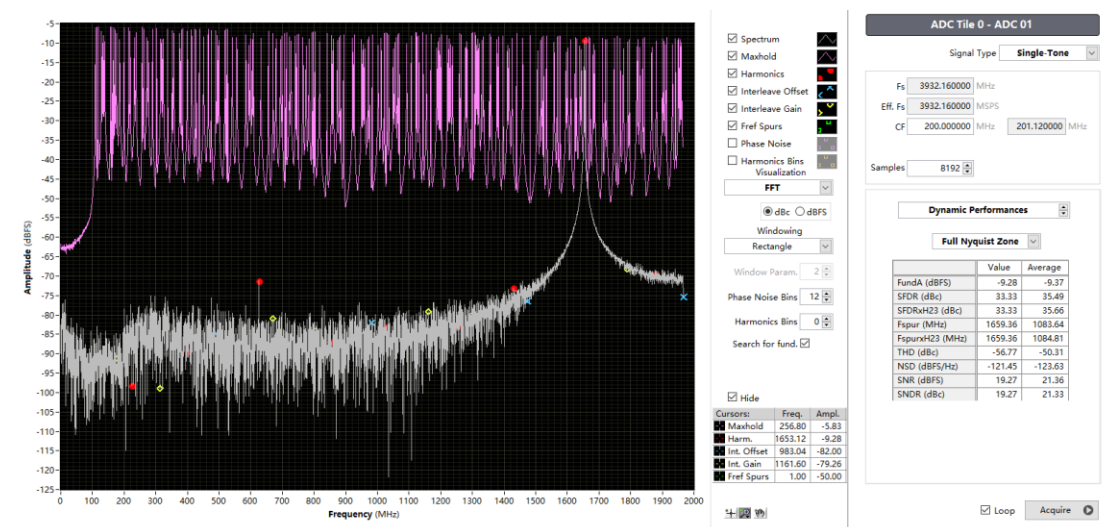


## 6. AD/DA 平坦度测试

### 6.1 ADC 平坦度

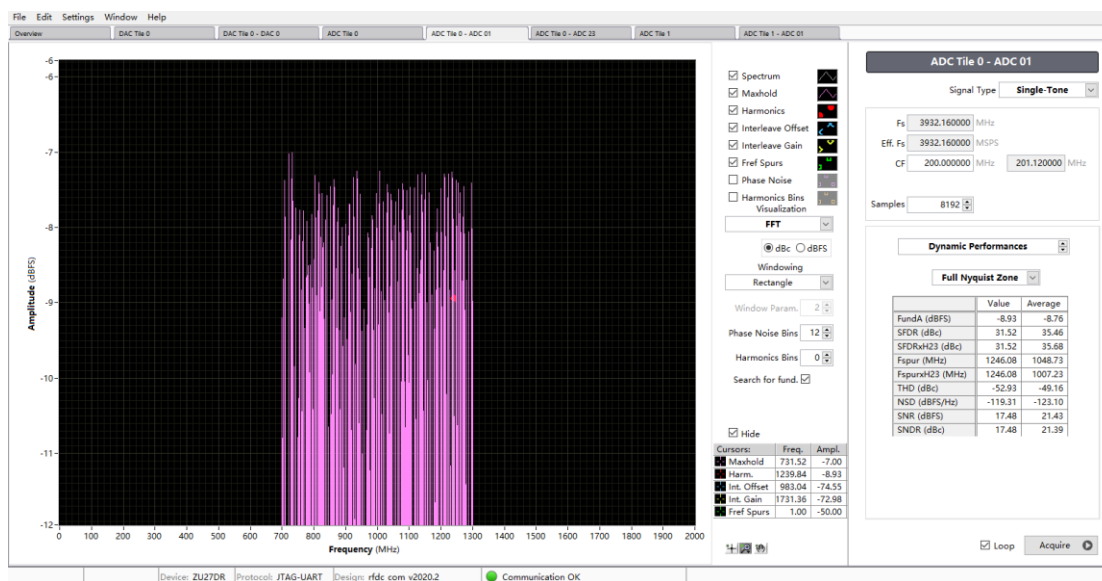
#### 6.1.1 100-2000M

外部信号源输出 100-2000Mhz 扫频信号，ADC 进行采集，使用 RF Analyzer 进行观察；可见 100-2000Mhz 频率范围内，ADC 采集信号的幅度差在大约 3-5dbm 之内。



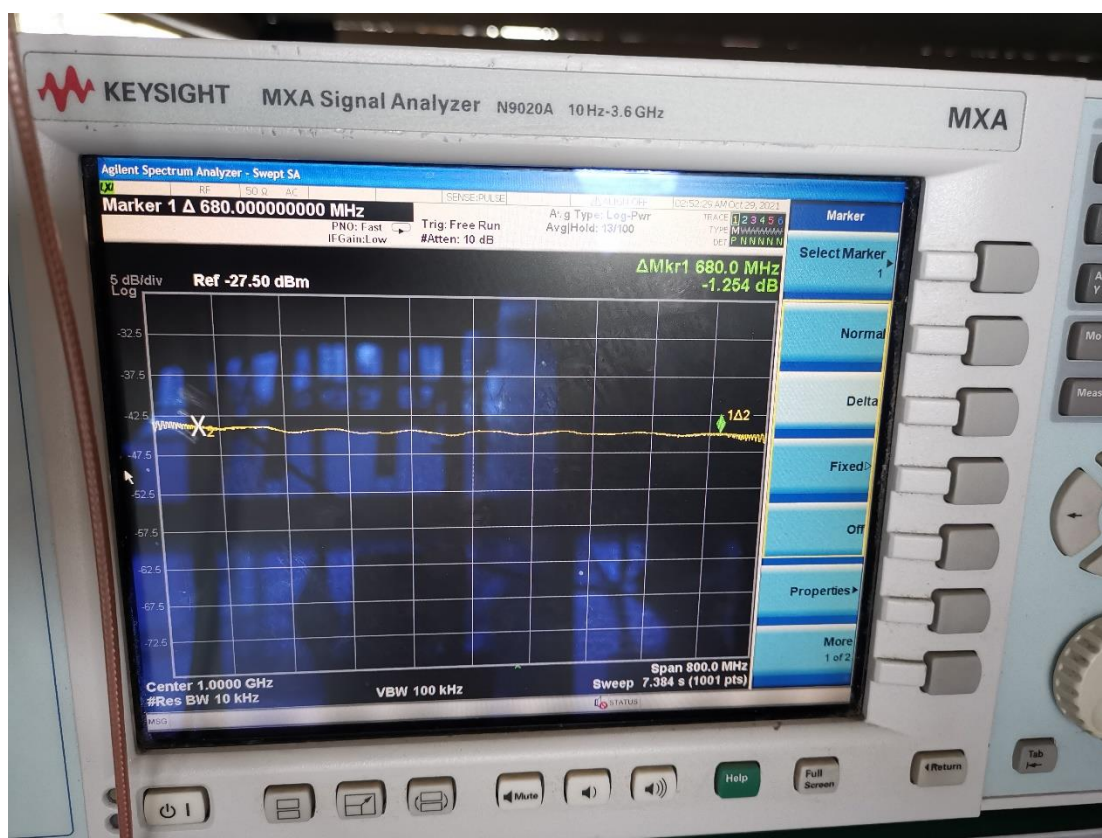
#### 6.1.2 700-1300M

外部信号源输出 700-1300Mhz 扫频信号，ADC 进行采集，使用 RF Analyzer 进行观察；可见 700-1300Mhz 频率范围内，ADC 采集信号的幅度差在大约 1-2dbm 之内。



## 6.2 DAC 平坦度

使用 RF Analyzer 输出 800M 带宽扫频信号，由 DAC 输出至频谱仪进行观察；可见输出信号幅度差在 2dbm 之内。

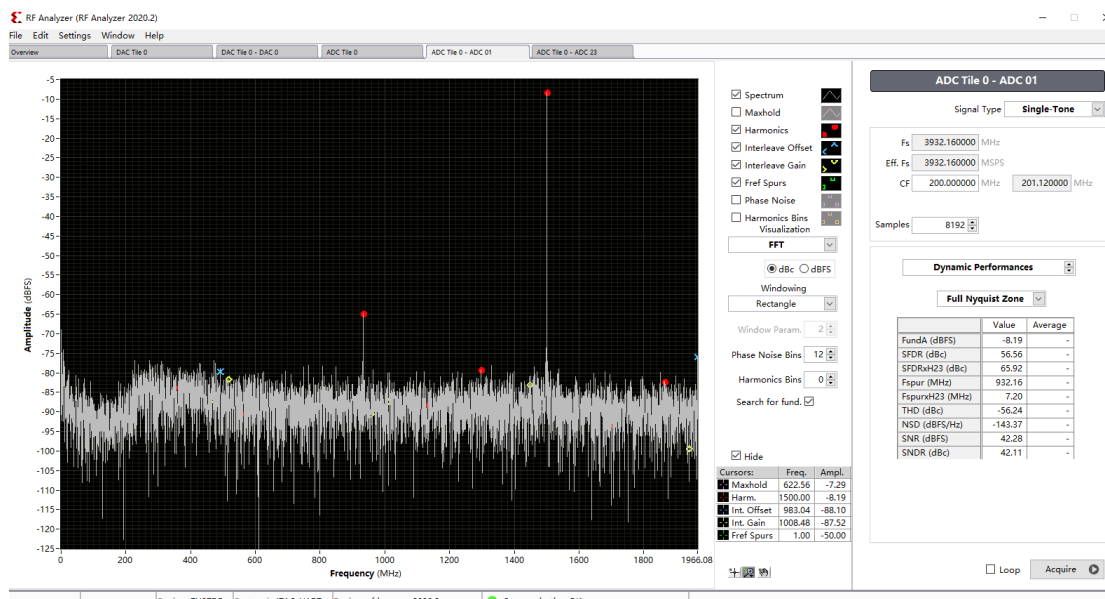




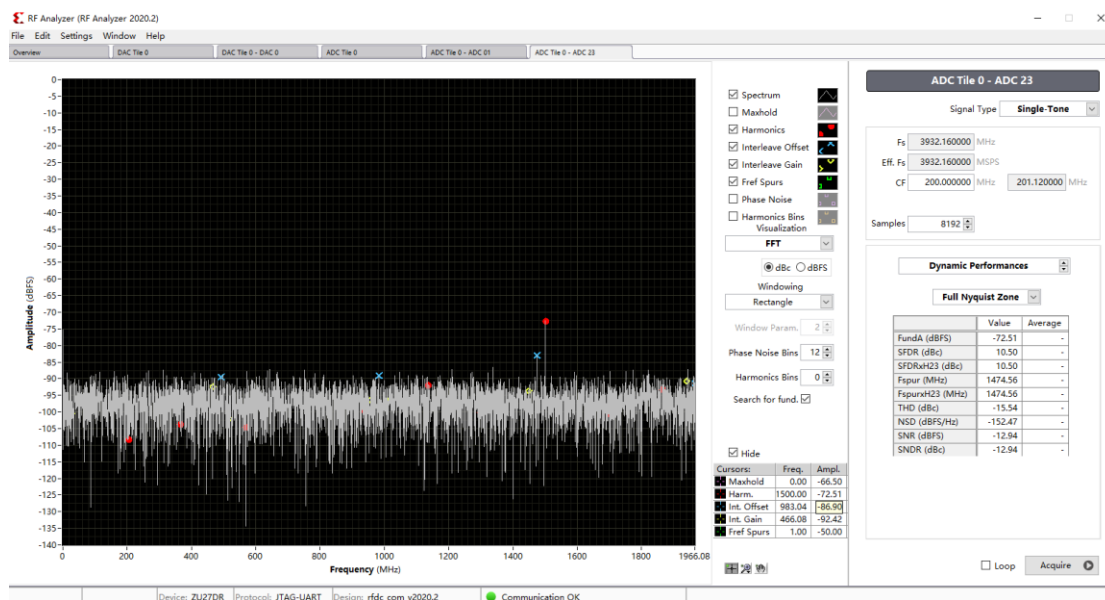
## 7. AD/DA 信号串扰测试

### 7.1 ADC 串扰测试

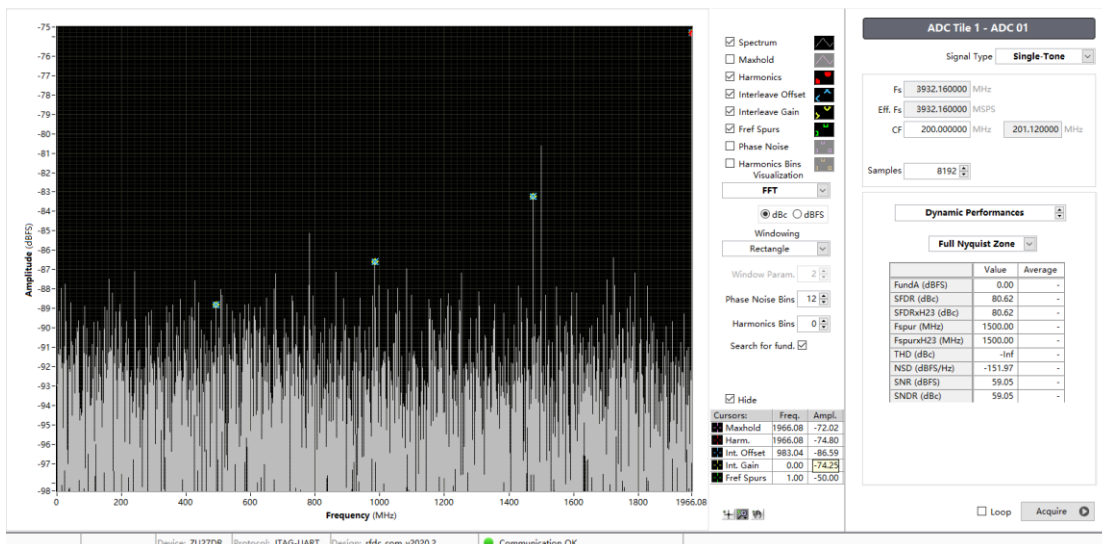
外部信号源输出 1500Mhz 单音信号，ADC 进行采集，使用 RF Analyzer 进行观察：



此时相同 Tile 内其他 ADC 采集到的信号为：

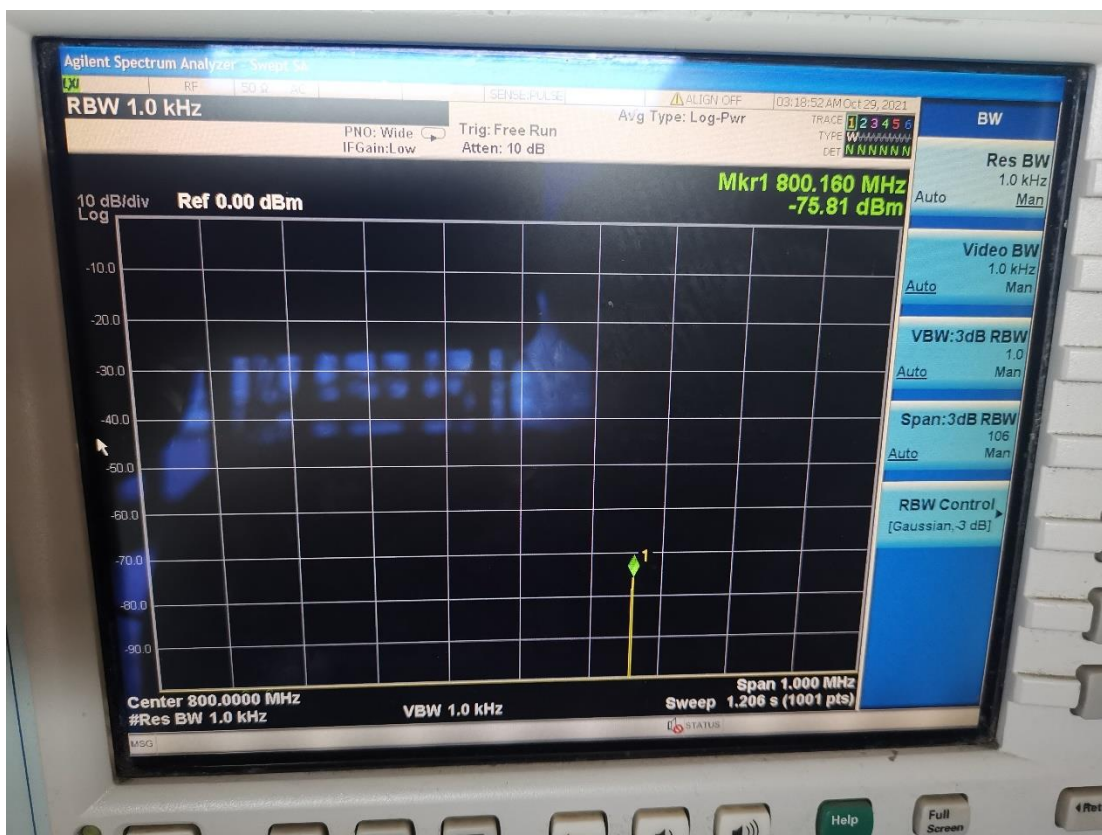


此时其他不同 Tile 内 ADC 采集到的信号为：



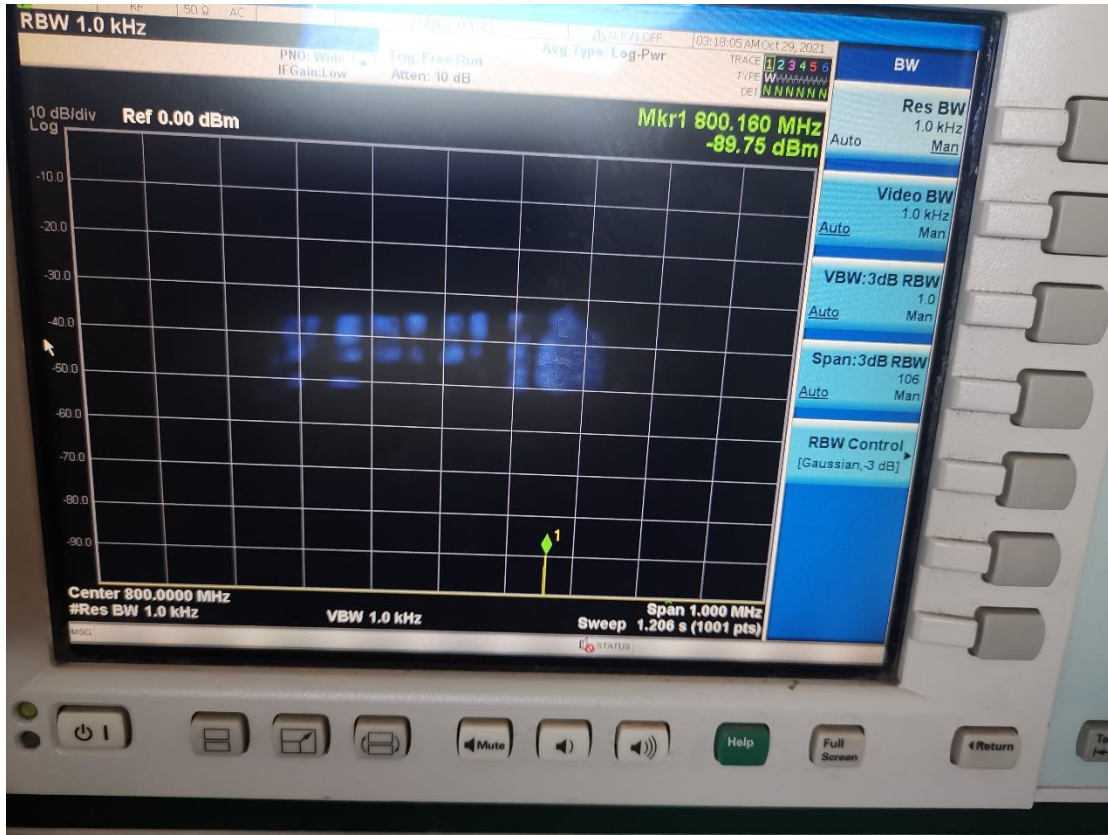
## 7.2 DAC 串扰测试

使用 RF Analyzer 输出 800M 单音信号，连接相同 Tile 内其他 DAC 至频谱仪进行观察；



## 测试报告

使用 RF Analyzer 输出 800M 单音信号，连接其他 Tile 内 DAC 至频谱仪进行观察：



## 8. 其他 ADC 指标测试

### 8.1 无杂散动态范围 SFDR

无杂散动态范围 (SFDR) 常用于衡量数据转换器在杂散分量干扰基本信号或导致基本信号失真之前可用的动态范围。SFDR 的定义是基本正弦波信号均方根 (RMS) 值与从 0Hz (DC) 到二分之一数据转换器采样速率 (如  $f_s/2$ ) 范围内测得的输出峰值杂散信号均方根值之比。峰值杂散分量可以是谐波关系，也可以是非谐波关系。

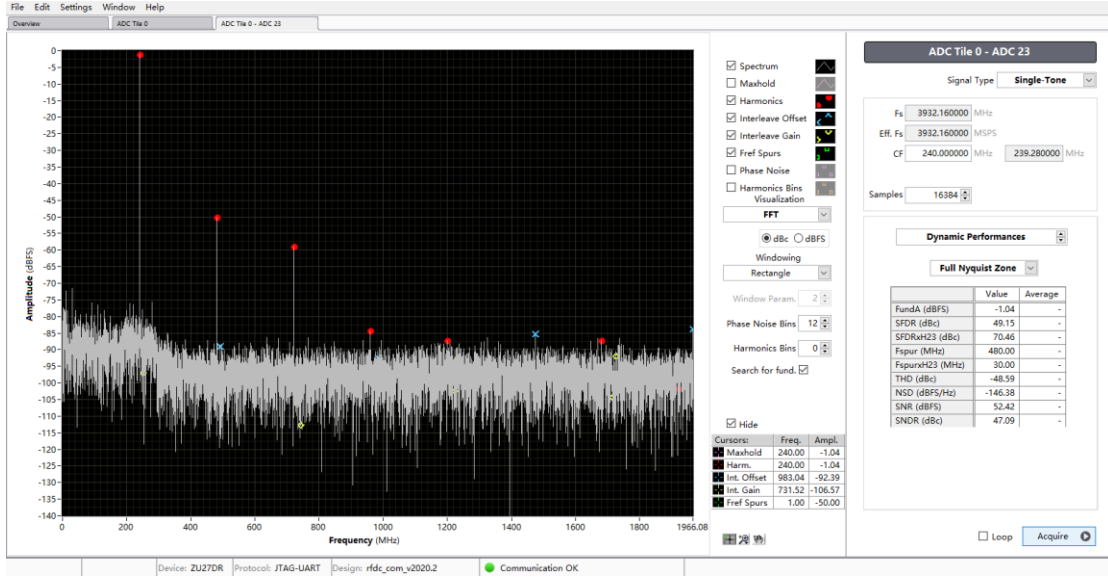
无杂散动态范围 (SFDR) 常用于衡量数据转换器在杂散分量干扰基本信号或导致基本信号失真之前可用的动态范围。SFDR 的定义是基本正弦波信号均方根 (RMS) 值与从 0Hz (DC) 到二分之一数据转换器采样速率 (如  $f_s/2$ ) 范围内测得的输出峰值杂散信号均方根值之比。峰值杂散分量可以是谐波关系，也可以是非谐波关系。

SFDR 可以使用下列方程计算：

$$SFDR_{(dBc)} = 20 \log \left( \frac{\text{Fundamental Amplitude (RMS)}}{\text{Largest Spur Amplitude (RMS)}} \right)$$

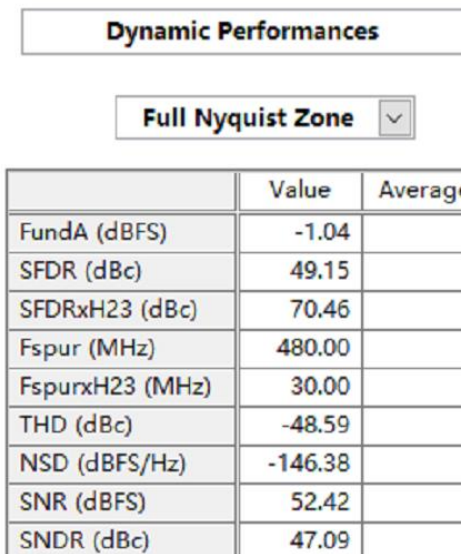
或者,  $SFDR_{(dBc)} = \text{基本信号幅度} - \text{最大杂散幅度}$

下图所示的是 Zynq UltraScale+ RFSoc 的 RF-ADC 的 SFDR 性能和谐波性能, 在 240MHz 输入幅度为-1dBFS 时, 测得  $SFDR = 49.15\text{dBc}$ 。



数据转换器的 SFDR 常常受输入信号的二次或三次谐波限制, 但通过精心设计滤波器和优化频率分配, 一般可避免二次谐波 (HD2) 和/或三次谐波 (HD3), 大幅提高 SFDR。在排除 HD2 和 HD3 后, 图中的 SFDRxH23 是 70.46dBc。

上图中工具也会同时分析出 ADC 相关指标, 如下。



## 8.2 信噪比 SNR

信噪比 (SNR) 是一般用来量化数据转换器内噪声的参数。它是输入信号功率与噪声功率的比值，一般使用 dB 作为单位。类似地，SNR 也能使用信号幅度和噪声幅度的 RMS 值衡量，如方程。

$$\begin{aligned} SNR &= \frac{Power_{signal}}{Power_{noise}} \\ &= \left( \frac{Amplitude_{signal(RMS)}}{Amplitude_{noise(RMS)}} \right)^2 \\ &= 20 \log \left( \frac{V_{in(RMS)}}{V_Q(RMS)} \right) \end{aligned}$$

由于采样抖动，信噪比在较高频率下一般会劣化。噪声来自于三个源头：

1. 量化噪声
2. ADC 热噪声
3. 抖动或采样不确定噪声

在满刻度正弦波输入条件下，ADC 的理论最高 SNR 从量化噪声推导而得。在奈奎斯特带宽上，信噪比还有另一个表达式：

$$SNR=6.02N+1.76db$$

这里 N 是理想 ADC 的位数。该公式体现的是对于理想的 N 位数据转换器（不考虑谐波失真）的正弦波输入，整个奈奎斯特带宽上能达到的最佳 SNR。此外，数据转换器的 SNR 也受到自身热噪声和采样时钟相位噪声的限制。当输入信号带宽低于奈奎斯特速率时，SNR 可以得到改善。

## 8.3 信噪失真比 SNDR

信噪失真比（也称为 SINAD）指输入正弦波时，RMS 信号功率与 (a) 总噪声功率和 (b) 输出端（不含 DC）的所有其他频率分量功率加上所有其他谐波分量功率的 RMS 和的比值。

SNDR 是用于衡量数据转换器的动态性能的关键参数之一，因为 SNDR 包含奈奎斯特带宽上的所有噪声和杂散。SNDR 说明的是输入信号的质量；SNDR 越大，输入功率中的噪声和杂散比率越小。SNDR 的表达式为：

$$SNDR = 10 \log_{10} \left( \frac{P_{Signal}}{P_{Noise} + P_{Distortion}} \right)$$

其中，信号功率是有用信号、噪声和失真分量的平均功率。SNDR 一般使用的单位有分贝 (dB)、相对于载波分贝 (dBc) 或满刻度分贝 (dBFS)。

SNDR 也有另一个表达式：

$$SNDR = 20 \log_{10} \sqrt{10^{\frac{-SNR}{10}} + 10^{\frac{THD}{10}}}$$

SNDR 是 SNR 规格和 THD 规格的综合，因此，SNDR 将所有不良频率分量与输入频率做比较，从而从总体上衡量数据转换器的动态性能。

## 8.4 有效位数 ENOB

有效位数 (ENOB) 是用于衡量数据转换器相对于输入信号在奈奎斯特带宽上的转换质量 (以位为单位) 的参数。ENOB 假定转换器是拥有理论上完美性能的转换器。完美数据转换器绝对不发生失真，唯一产生的噪声是量化噪声，所以 SNR 等于方程 3 中的 SNDR，即  $SNR_{(dBFS)} = 6.02N + 1.76$ 。因此，ENOB 也是指定 SNDR 的另一种表达方式：

$$ENOB(N) = \frac{SNDR_{(dBFS)} - 1.76}{6.02}$$

其中 SNDR(dBFS) 假定满刻度输入信号。

然而，对于非理想数据转换器而言，SNDR 和 ENOB 会发生劣化，包含噪声和其他缺陷，例如器件热噪声、输出代码缺失、谐波、AC/DC 非线性、增益/偏移误差和孔径时钟相噪或抖动。外部偏置基准源和电源轨上的噪声也会降低 ENOB。

此外，类似于 THD 因非线性原因随输入频率增加而劣化，ENOB 值也会随频率加大而劣化。ENOB 来自于 SNDR，而 SNDR 又与 THD 以及 SNR 相关联。要了解数据转换器的准确 ENOB，需阅读数据手册中的详细规格和规定的条件。

鉴于上述标准，大部分模拟数据转换器 IC 厂商一般倾向于推广理想条件下的 ENOB，尤其是数据手册标题所载的 ENOB 值。然而，大量系统工程师和采购经理仍然好奇的是，为什么测量出的 ENOB 值和数据手册所载的理想值不同。有关 ENOB 的一些理解要点：

## 测试报告

- 一般数据转换器数据手册标题显示的“位数”（12 位或 14 位）指的是数字位或电压分辨率。这与 ENOB 无关。
- ENOB 主要与噪声、非线性和输入频率存在函数关系。
- ENOB 会因多种外部不确定性因素（例如时钟源、电源等）而劣化。
- ENOB 是在整个奈奎斯特带宽上（DC 到  $f_s/2$ ）计算的。
- ENOB 并非分析 SDR 等直接 RF 系统的理想指标。