

1 PCIE 传输 ACM9238 数据采集

工程源码	-ACZ7015 开发板 -- pcie_riffa
相关视频课程	无
说明	如果您手头的硬件不支持本实验，您可以学习本实验的理论内容，也可以跳过本节内容，继续后续内容的学习。 进行本章实验务必先完成基于 RIFFA 框架搭建 PCie 测试环境

章节导读

本节内容将介绍 ACM9238 模块基于 PCIE 进行双通道数据采集的相关内容。PC 采用 PCIE 接口、基于 riffa 框架进行数据传输，FPGA 将数据转化成控制命令，实现对 ACM9238 模块的采样频率、数据采样个数的合理配置，同时将数据通过 PCIE 传输到电脑。用户可以在电脑上通过 riffa 驱动进行指令的下发，并以文件的形式保存接收到的数据，然后使用 MATLAB 软件进行进一步的数据处理分析。

1.1 系统整体设计

通过电脑上的 riffa 驱动，将命令帧进行发送，然后通过 ACZ7015 开发板上的 PCIE 接收，随后将接收到的数据转换成命令，从而实现对 ACM9238 模块采样频率、数据采样个数的配置。配置完成之后，ACM9238 模块开始采集数据，将采集的数据通过 PCIE 传输至电脑，电脑端将采集到的数据通过 MATLAB 进行进一步的分析。

系统的整体设计框图如图 1-1 所示。

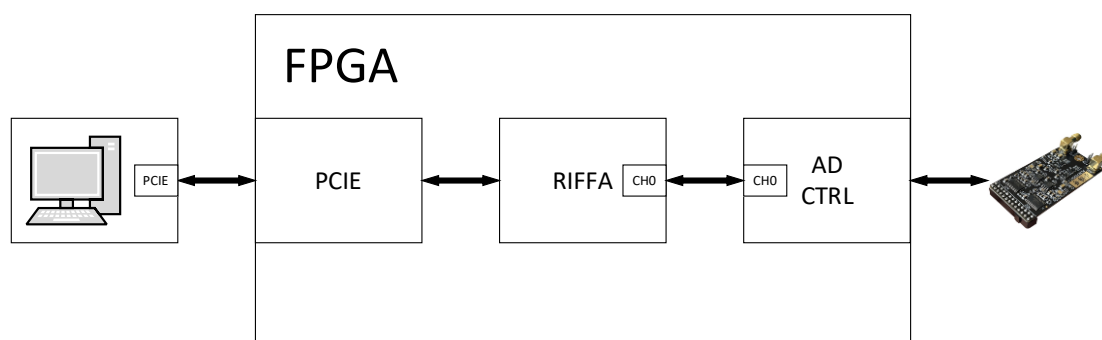


图 1-1 千兆以太网传输 9238 数据采集

1.2 ACM9238 模块简介

该模块如下图 1-2 所示。ACM9238 模块配合前端模拟信号调理电路，实现了±5V 电压范围内信号的高速采样。该模块共使用 2 路完全相同的 AD 采样和信号调理电路，构成了双通道高速 AD 采样电路。两路 ADC 电路完全独立，结构和元器件参数相同，确保了两个通道有较高的一致性。本模块与 FPGA 连接采用并行接口，每路 ADC 包括 12 位数据信号（ADC_DATA），1 位时钟信号（ADC_CLK），1 位超量程指示信号（ADC_OTR），该模块接口图如下图 1-3 所示。

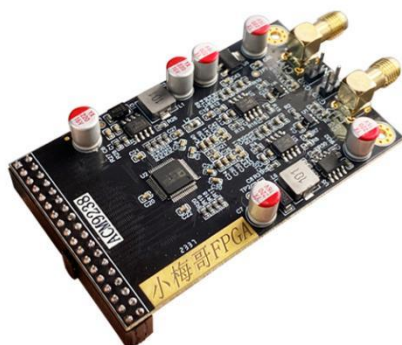


图 1-2 ACM9238 模块图

P1		
CH1 D0	1	2 CH1 CLK
CH1 D2	3	4 CH1 D1
CH1 D4	5	6 CH1 D3
CH1 D6	7	8 CH1 D5
CH1 D8	9	10 CH1 D7
	11	12
CH1 D10	13	14 CH1 D9
CH1 OTR	15	16 CH1 D11
CH2 D1	17	18 CH2 D0
CH2 D3	19	20 CH2 D2
CH2 D5	21	22 CH2 D4
CH2 D7	23	24 CH2 D6
CH2 D9	25	26 CH2 D8
CH2 D11	27	28 CH2 D10
	29	30
CH2 OTR	31	32 CH2 CLK
	33	34
	35	36
Header 18X2		

图 1-3 ACM9238 模块接口图

使用该模块时，仅需 FPGA 为每路 ADC 提供一路时钟信号，ADC 则会在每个时钟周期输出一个 12 位的采样结果。当 9238BSTZ 模拟输入端接 -5V 至 +5V 之间变化的正弦波电压信号时，其转换后的数据也是成正弦波波形变化，转换波形如下图 1-4 所示，从图中可以看出 9238BSTZ 采集到的数据是无符号数据。

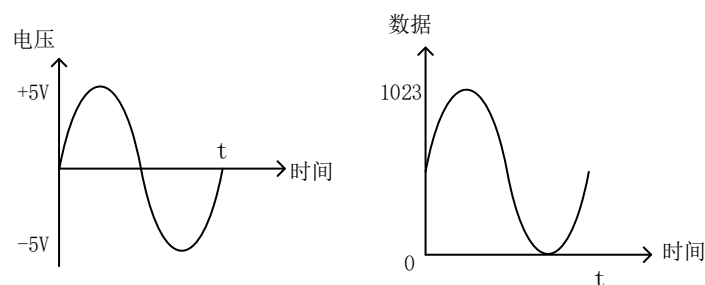


图 1-4 9238BSTZ 正弦波模拟电压值（左）、数据（右）

本模块采样率上限为 65Msps，采样率就等于 FPGA 提供给 ADC 的时钟频率。如需使用低于时钟频率的采样率，可以依旧给 ADC 提供 50MHz 的时钟信号，但在 FPGA 内部，对 50Msps 的采样结果数据进行抽取重采样的方法实现。比如期望以 1Msps 的采样速率采样，则只需要每间隔 50 个采样数据取一个结果存储或使用，其他 49 个数据直接舍弃，这样就能实现 1MSPS 的采样率了。十分不建议采用直接对提供给 ADC 芯片的时钟信号降频以实现降低采样率的效果的方法，因为时钟太低，会影响 ADC 芯片内部采样保持电路的工作情况，导致采样误差偏大。

本模块可用于小梅哥全系列 FPGA、SOC、Zynq 开发板，包括国产开发板和各核心板的评估底板。AC620、AC6102、ACX720、ACZ7015、智多晶 FPGA 开发板（AC208-SA5Z）、AC608 评估底板、AC601 评估底板、AC675 评估底板等。

1.3 模块说明

1.3.1 ad9238_ctrl 模块

ad9238_ctrl 模块将 PCIE 接收到的数据进行解析，得到控制命令，最终控制 ADC 采样数量、采样速率等，并将数据上传至上位机，该模块的基本结构框图如下图 1-5：

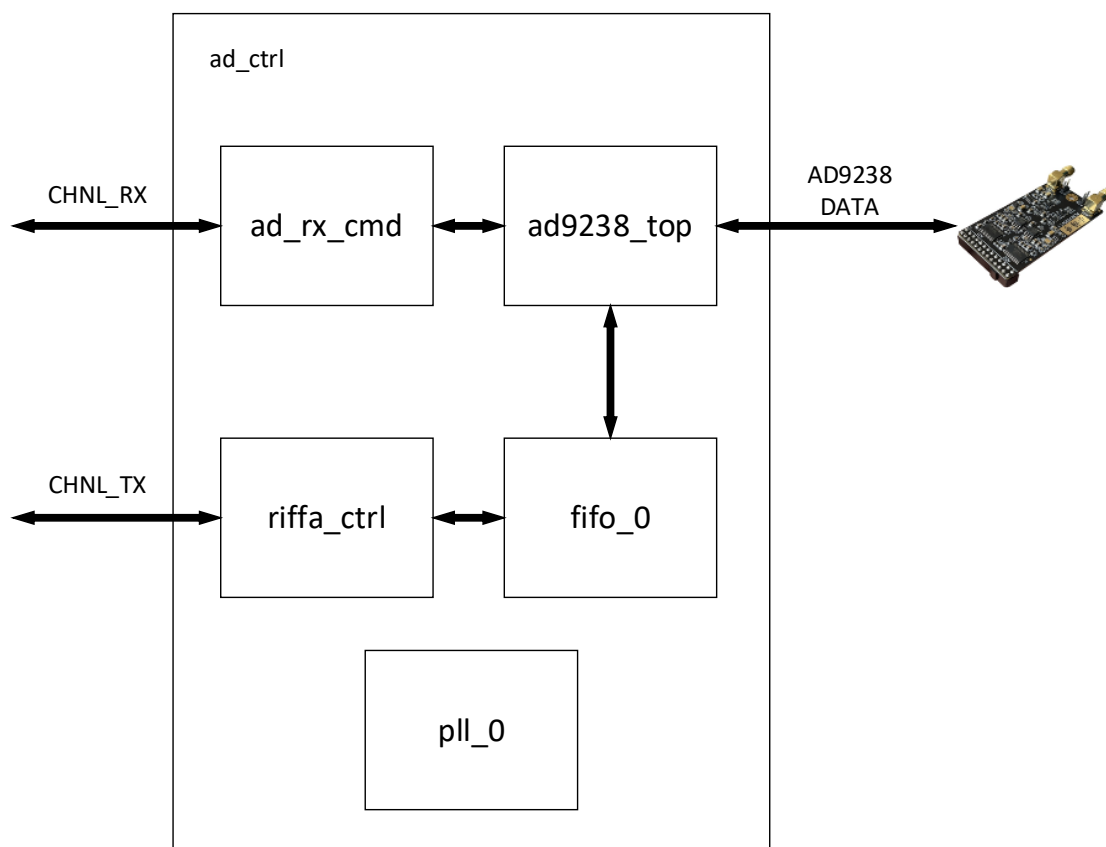


图 1-5 eth_receive_cmd 模块基本结构框图

上述图中的信号说明如下表 1-1 所示。

表 1-1 eth_receive_cmd 模块信号说明表

信号名称	I/O	信号意义
CHNL_RX	I/O	Riffa 虚拟通道 RX 接口
CHNL_TX	I/O	Riffa 虚拟通道 TX 接口
AD9238 CHNL	I/O	AD9238 数据采集通道

具体信号本节不进行说明，详情可以查阅相关模块。

下面将对本次实验需要设计需要添加的 IP 进行说明。

1.3.1.1 pll_0 模块

锁相环模块，产生 ad9238 驱动信号与数据采集信号。本次实验采取 100Mhz 采样率

锁相环 IP 配置界面如下图 1-6 所示。

Component Name clk_wiz_0

Clocking Options Output Clocks Port Renaming MMCM Settings Summary

The phase is calculated relative to the active input clock.

Output Clock	Port Name	Output Freq (MHz)		Phase (degrees)		Duty Cycle (%)		Drives
		Requested	Actual	Requested	Actual	Requested	Actual	
<input checked="" type="checkbox"/> clk_out1	clk_out1	100	100.000	0.000	0.000	50.000	50.0	BUFG
<input checked="" type="checkbox"/> clk_out2	clk_out2	100	100.000	0.000	0.000	50.000	50.0	BUFG
<input type="checkbox"/> clk_out3	clk_out3	100.000	100.000	0.000	0.000	50.000	50.0	BUFG

图 1-6 锁相环配置界面

(注：acd9238 每通道 65Msps 最高采样速率，采用 100M 采样波形会有所失真)

1.3.1.2 ad_rx_cmd 模块

FIFO IP 核，接受上位机通过 PCIE 下发指令。

配置如图 1-7 所示：

Basic Native Ports Status Flags Data Counts Summary

Interface Type

☒ Native ☐ AXI Memory Mapped ☐ AXI Stream

Fifo Implementation Independent Clocks Block RAM

Synchronization Stages 2

图 1-7 双端口 FIFO

然后配置读写位宽为 64 位，写深度为 512，如图 1-8 所示：

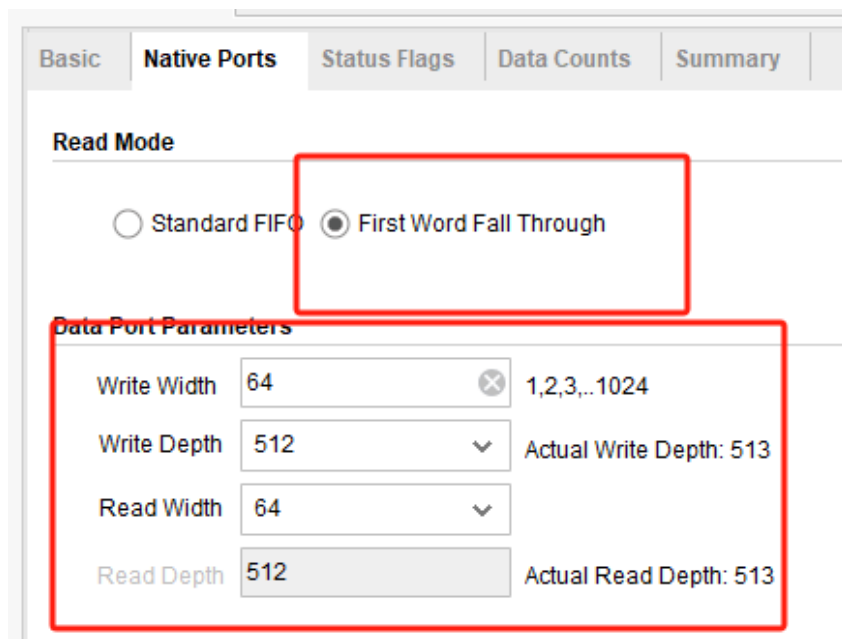


图 1-8 配置 FIFO 的读写位宽及深度

1.3.1.3 fifo_0 模块

FIFO IP 核，缓存将通过 PCIE 发送至上位机 AD 数据。

配置如图 1-9 所示：

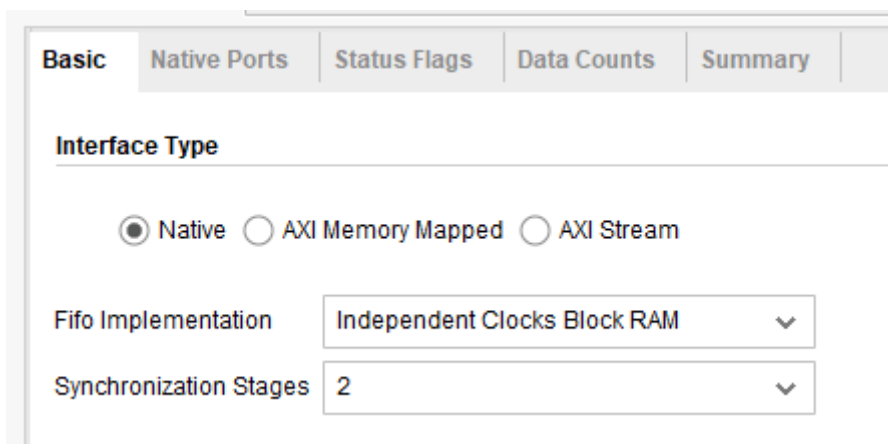


图 1-9 双端口 FIFO

然后配置读写位宽分别为 64 位、32 位，写深度为 1024，如图 1-8 所示：

The screenshot shows the 'Native Ports' configuration tab. Under 'Read Mode', the 'First Word Fall Through' option is selected. Under 'Data Port Parameters', the 'Write Width' is set to 32, 'Write Depth' to 1024, 'Read Width' to 64, and 'Read Depth' to 512. The 'Actual Write Depth' is 1027 and the 'Actual Read Depth' is 513. Red boxes highlight the 'First Word Fall Through' option and the 'Data Port Parameters' section.

Read Mode	
<input type="radio"/> Standard FIFO	<input checked="" type="radio"/> First Word Fall Through

Data Port Parameters	
Write Width	32
Write Depth	1024
Read Width	64
Read Depth	512

图 1-10 配置 FIFO 的读写位宽及深度

1.3.2 ad9238_top 模块

ACM9238 控制器模块（ad9238_top）的功能是控制 ADC 的采样，并且将 ADC 采集到的 12 位数据转换成双通道 32 位的数据，以便于我们的电脑进行数据存储，该模块如图 1-11 所示：

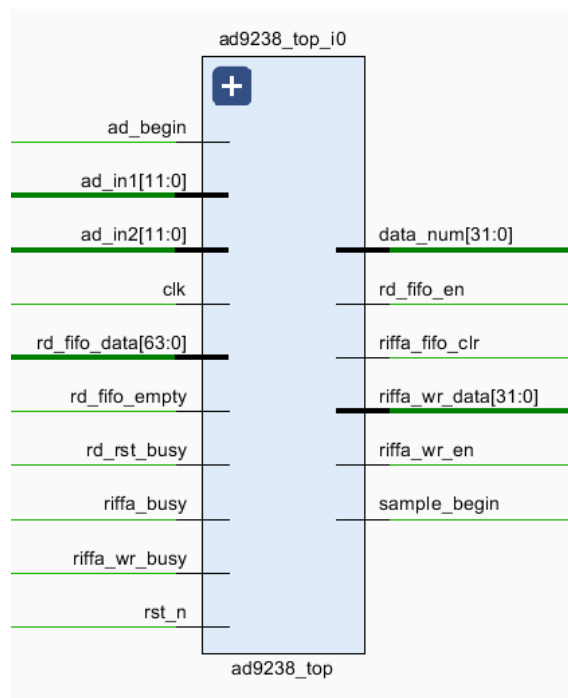


图 1-11 ad9238_top 模块基本结构框图

具体功能可以查阅相关模块。

1.3.3 riffa_ctrl 模块

riffa 控制模块的功能是向 riffa 框架提供的接口写数据，该模块如图 1-12 所示。

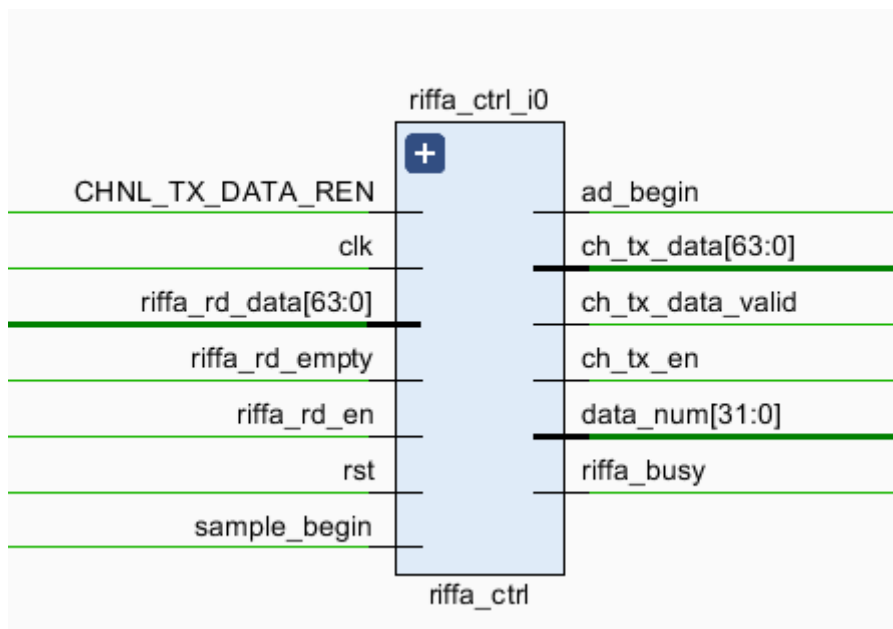


图 1-12 state_ctrl 模块结构框图

具体功能可以查阅相关模块。

1.4 板级验证

本次实验的板级验证环节，主要验证：通过电脑将命令帧进行发送，然后通过 ACZ7015 开发板上的 PCIE 接收，随后将接收到的数据转换命令，最终实现对 ACM9238 模块的采样频率、数据采样个数的配置。配置完成之后，ACM9238 模块开始采集数据，并将 ACM9238 模块采集的数据通过 PCIE 传输到电脑。电脑端将接收到的数据进行保存，然后通过 MATLAB 进行进一步的分析。针对本次实验，我们也提供有专门的上位机软件，用户只需要在软件界面进行参数配置，便可以实时观察到实时的数据波形变化，使用起来非常方便。

1.4.1 系统所需硬件

本次设计所需硬件如下，相关模块资料可以点击超链接查看：

1. [ACZ7015 开发板](#)一块
2. [ACM9238 模块](#)一个
3. 具有 PCIE 插槽 PC 设备一台
4. Type-C 下载线一根
5. 信号发生器一台

1.4.2 硬件连接

本次设计系统硬件连接如下图 1-14 所示：

1. 使用 Type-C 线连接开发板调试接口和电脑 USB 口
2. ACM9238 模块连接至 GPIO0 的 40 pin 的排针上，靠右连接，1 脚和 1 脚对应
3. 在电脑断电的情况下将开发板 pcie 金手指与主板 pcie 插槽进行连接（或者选择延长线接法，**一定要在主机关机的状态下才能拔插 PCIe 开发板，否则会导致硬件的损坏**）。
4. 将开发板的电源开关打开，然后启动电脑。本实验直接使用主机对 FPGA 开发板进行供电，不需要外加电源对开发板进行供电。电脑开机后将比特流下载到开发板中。



图 1-13 硬件连接图



图 1-14 硬件连接图（延长线接法）

1.4.3 驱动的安装

本章实验为基于 RIFFA 框架搭建 PCIe 测试环境修改，不对驱动的安装进行讲解，驱动安装完成如图所示



图 1-15 修改电脑 IP 地址

1.4.4 功能验证

我们通过信号发生器给 ACM9238 的通道一输入 100Khz，vpp=5V 的正弦波、通道二输入 200KHZ，vpp=5V 的正弦波。

本次实验所用到的执行文件都在 sample 文件夹下，测速执行文件可以在 DOS 界面中执行，图 1-16 为 sample 文件夹所在路径。

名称	修改日期	类型	大小
pcie_riffa	2024/10/18 14:14	文件夹	
riffa-master	2024/10/18 12:40	文件夹	
sample_app	2024/10/18 12:52	文件夹	
RIFFA 2.1 A Reusable Integration Fra...	2020/4/29 9:56	PDF 文件	1,412 KB
riffa-master.7z	2020/6/12 14:25	WinZip 文件	19,512 KB
setup_dbg.exe	2016/8/15 10:21	应用程序	5,248 KB

图 1-16 sample_app 文件夹所在路径

同时按住键盘上的 Win+R 快捷键，然后在弹出的界面中输入 cmd，然后再点击确定。

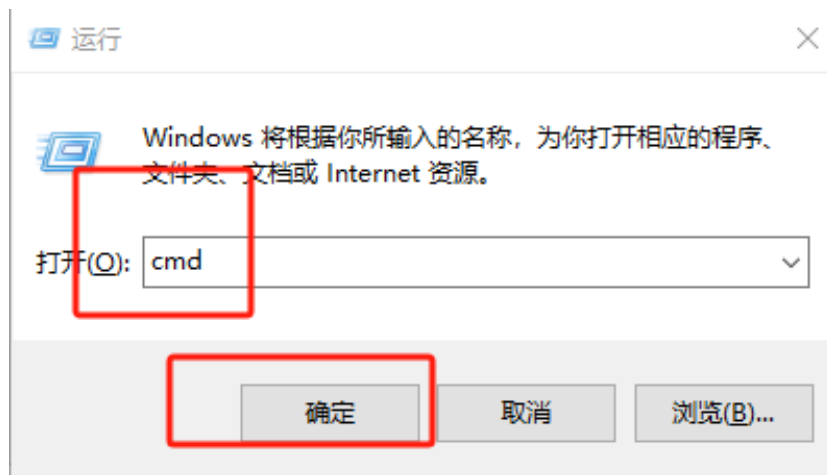


图 1-17 DOS 界面进入方式

进入 DOS 界面，首先将可执行文件所在的电脑盘确定。本次实验 sample 文件夹放在 E 盘，所以在 DOS 界面中输入如下指令（如果将 sample 文件夹放在 D 盘就应该输入指令 D:）

E:

指令输入好之后，我们点击回车 DOS 界面如下所示。

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.19045.5011]
(c) Microsoft Corporation. 保留所有权利。
C:\Users\Mr_Zhou>E:
E:\>
```

图 1-18 DOS 界面 1

然后再输入测速执行文件所在的路径，指令为 cd+具体路径，本次实验的路径为 E:\Learn\others\riffa_ad9238\sample_app，所以输入如下指令（如果将 sample 文件夹放在 D 盘就应该输入指令 cd D:\pcie_riffa\sample_app）

cd E:\Learn\others\riffa_ad9238\sample_app

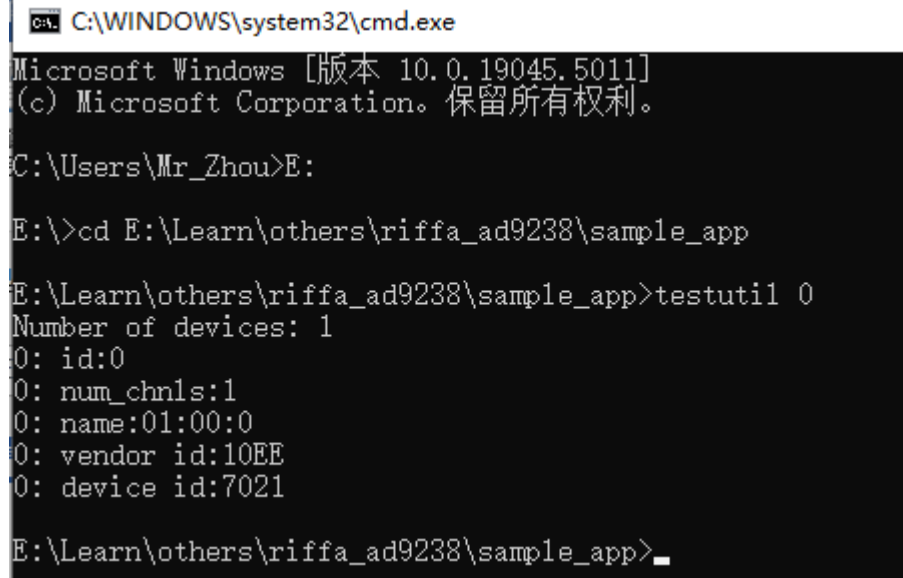
```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.19045.5011]
(c) Microsoft Corporation. 保留所有权利。
C:\Users\Mr_Zhou>E:
E:\>cd E:\Learn\others\riffa_ad9238\sample_app
E:\Learn\others\riffa_ad9238\sample_app>_
```

图 1-19 DOS 界面 2

然后输入 testutil 0，可以查看当前的 PCIE 设备信息。

testutil 0

指令输入好之后，我们点击回车 DOS 界面如下所示。



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.19045.5011]
(c) Microsoft Corporation。保留所有权利。

C:\Users\Mr_Zhou>E:

E:\>cd E:\Learn\others\riffa_ad9238\sample_app

E:\Learn\others\riffa_ad9238\sample_app>testutil 0
Number of devices: 1
0: id:0
0: num_chnls:1
0: name:01:00:0
0: vendor id:10EE
0: device id:7021

E:\Learn\others\riffa_ad9238\sample_app>_
```

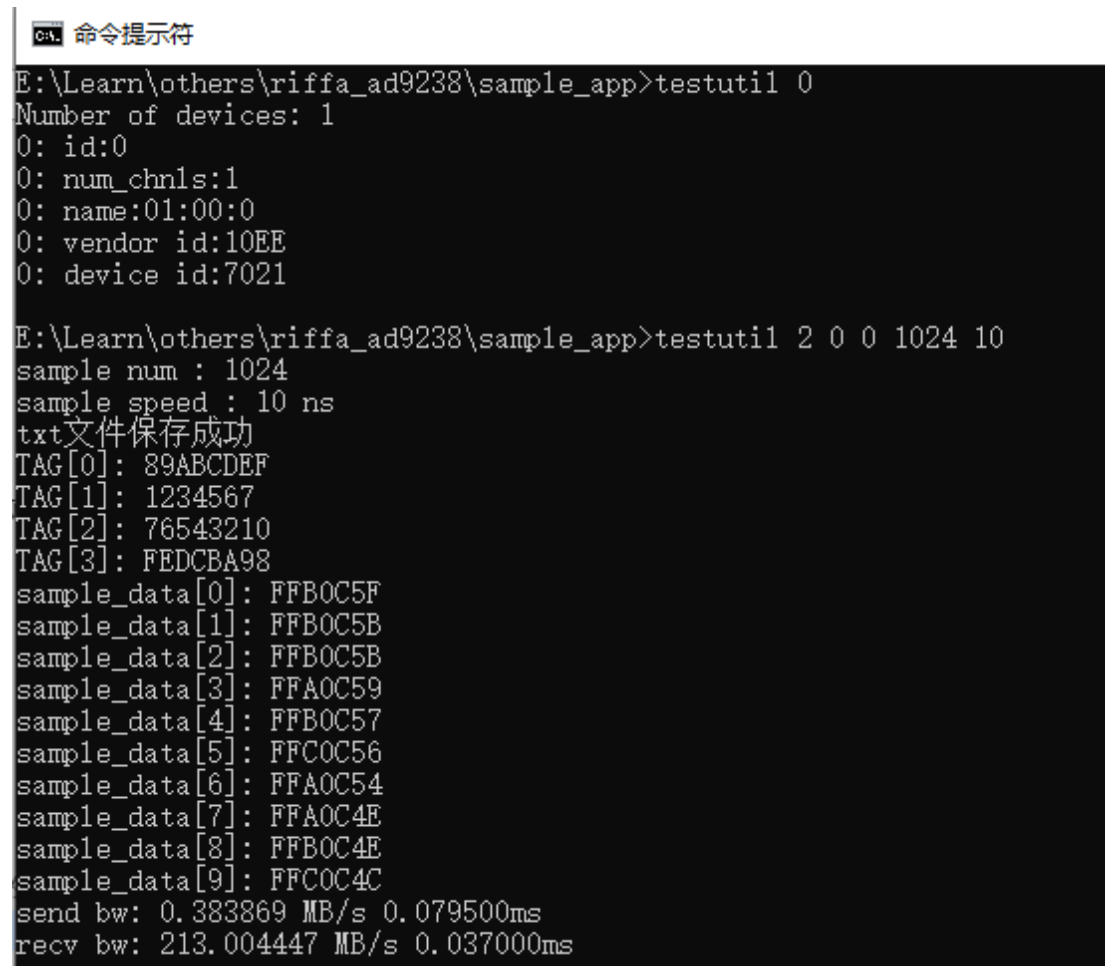
图 1-20 DOS 界面 3

vendor id 选项：是厂家信息，默认为 10EE，这是 Xilinx 的供应商标识；
device id 选项：是设备信息，默认为 7021，7021 中的 7 表示 7 系列器件，7021 中的 2 对应的是 PCIe2.x 版本，7021 中的 1 是通信链路的数量，这些信息均为 7 Series Integrated Block for PCI Express IP 设置一致。

接下来我们输入指令 `testutil 2 0 0 1024 10`，该命令是通过 `pcie` 下发指令使 `ad9238` 以 `100mhz` 的速率采样 `1024` 个数据

```
testutil 2 0 0 1024 10
```

指令输入好之后，我们点击回车 DOS 界面如下所示。



```
04. 命令提示符
E:\Learn\others\riffa_ad9238\sample_app>testutil 0
Number of devices: 1
0: id:0
0: num_chnls:1
0: name:01:00:0
0: vendor id:10EE
0: device id:7021


E:\Learn\others\riffa_ad9238\sample_app>testutil 2 0 0 1024 10
sample num : 1024
sample speed : 10 ns
txt文件保存成功
TAG[0]: 89ABCDEF
TAG[1]: 1234567
TAG[2]: 76543210
TAG[3]: FEDCBA98
sample_data[0]: FFB0C5F
sample_data[1]: FFB0C5B
sample_data[2]: FFB0C5B
sample_data[3]: FFA0C59
sample_data[4]: FFB0C57
sample_data[5]: FFC0C56
sample_data[6]: FFA0C54
sample_data[7]: FFA0C4E
sample_data[8]: FFB0C4E
sample_data[9]: FFC0C4C
send bw: 0.383869 MB/s 0.079500ms
recv bw: 213.004447 MB/s 0.037000ms
```

图 1-21 DOS 界面 4

其中 `TAG` 为可定义标识符，`sample_data` 为部分展示采样数据。

接下来我们再输入指令 `testutil 2 0 0 1024000 10`，采样 1024000 个数据。

```
testutil 2 0 0 1024000 10
```

 命令提示符

```
sample_data[5]: FFC0C56
sample_data[6]: FFA0C54
sample_data[7]: FFA0C4E
sample_data[8]: FFB0C4E
sample_data[9]: FFC0C4C
send bw: 0.383869 MB/s 0.079500ms
recv bw: 213.004447 MB/s 0.037000ms

E:\Learn\others\riffa_ad9238\sample_app>testutil 2 0 0 1024000 10
sample num : 1024000
sample speed : 10 ns
txt文件保存成功
TAG[0]: 89ABCDEF
TAG[1]: 1234567
TAG[2]: 76543210
TAG[3]: FEDCBA98
sample_data[0]: FFA0C23
sample_data[1]: FFC0C24
sample_data[2]: FFA0C25
sample_data[3]: FFA0C22
sample_data[4]: FFC0C25
sample_data[5]: FFB0C25
sample_data[6]: FFB0C25
sample_data[7]: FFC0C25
sample_data[8]: FFB0C26
sample_data[9]: FFA0C26
send bw: 0.864521 MB/s 0.035300ms
recv bw: 376.080801 MB/s 10.397300ms
```

图 1-22 DOS 界面 5

接下来我们再输入指令 `testutil 2 0 0 10240000 10`，采样 1024000 个数据。

```
testutil 2 0 0 10240000 10
```

命令提示符

```
sample_data[5]: FFB0C25
sample_data[6]: FFB0C25
sample_data[7]: FFC0C25
sample_data[8]: FFB0C26
sample_data[9]: FFA0C26
send bw: 0.864521 MB/s 0.035300ms
recv bw: 376.080801 MB/s 10.397300ms

E:\Learn\others\riffa_ad9238\sample_app>testutil 2 0 0 10240000 10
sample num : 10240000
sample speed : 10 ns
txt文件保存成功
TAG[0]: 89ABCDEF
TAG[1]: 1234567
TAG[2]: 76543210
TAG[3]: FEDCBA98
sample_data[0]: FFA01EA
sample_data[1]: FFA01F0
sample_data[2]: FFB01F6
sample_data[3]: FFA01FC
sample_data[4]: FFB01FF
sample_data[5]: FFC0205
sample_data[6]: FFB020A
sample_data[7]: FFB020E
sample_data[8]: FFC0215
sample_data[9]: FFA021A
send bw: 0.201836 MB/s 0.151200ms
recv bw: 376.734974 MB/s 103.697500ms
```

图 1-23 DOS 界面 6

接着打开 `sample_app` 文件夹，可以查看采样数据文件

此电脑 > DATA (E:) > Learn > others > riffa_ad9238 > sample_app				
名称	修改日期	类型	大小	
example.bin	2024/10/18 15:34	BIN 文件	40,000 KB	
README.txt	2024/10/16 15:43	文本文档	1 KB	
riffa.h	2020/4/26 17:11	C/C++ Header	6 KB	
riffa.lib	2020/4/26 17:11	Object File Library	3 KB	
testutil.c	2024/10/18 15:21	C Source	5 KB	
testutil.exe	2024/10/18 15:21	应用程序	45 KB	
timer.h	2020/4/26 17:11	C/C++ Header	1 KB	
untitled.m	2024/10/18 15:43	M 文件	3 KB	

图 1-24 采样文件路径

1.4.4.1 MATLAB 图像绘制

前面通过 pcie 得到了 ADC 采集到的数据文件，然后我们就需要对采集到的数据进行分析，本次实验使用 MATLAB 软件进行分析。使用 MATLAB 软件需要读者电脑安装了 MATLAB，如果已经安装好了 MATLAB 软件，则可以双击我们提供的 untitled.m 文件，在打开方式里选择以 MATLAB 打开，该文件位于本次实验的工程压缩包下，将压缩包解压便可以看到该文件。文件打开之后，读者需要将代码中文件路径修改为你保存的数据文件路径，随后点击运行便可以直观的看到数据是否正确，MATLAB 操作如下图 1-25 所示：

```
%基于7015，加载两个7606C模块，每个模块同时采集相同通道，后续可改不同通道
%
%=====
%设置ADC采集数据有效位宽
ADC_DATA_WIDTH = 12;
%ADC采集电压范围±VOL_RANGE (V)
VOL_RANGE = 5;
%采样率(Hz)
SAMPLE_RATE = 100000000;
%ADC采集电压分辨率 LSB
LSB = VOL_RANGE*2/(2^ADC_DATA_WIDTH);
%=====
% ... (前面的代码保持不变)

% 读取文件数据，转换成波形显示
fileID = fopen('E:\Learn\others\riffa_ad9238\sample_app\example.bin');
src_data = fread(fileID);
fclose(fileID);
src_data_hex = src_data;

DATA_NUM = length(src_data_hex);
if (DATA_NUM > 1024*64)
    DATA_NUM = 1024*64;
end

% 假设两个通道交替存储，每个通道的采样数为DATA_NUM/2
voltage_code = 1:DATA_NUM/2;
voltage_code = voltage_code';

for i=1:DATA_NUM/2
    if src_data_hex(2*i-1)+src_data_hex(i*2)*256 > 2^(ADC_DATA_WIDTH-1)
        voltage_code(i) = src_data_hex(2*i-1)+src_data_hex(i*2)*256-2^ADC_DATA_WIDTH;
    else
        voltage_code(i) = src_data_hex(2*i-1)+src_data_hex(i*2)*256;
    end
end

% 二进制ADC数据转换为电压值
voltage = voltage_code*LSB - LSB/2;
```

图 1-25 修改文件路径并运行

得到的波形图如下图 1-26 所示：

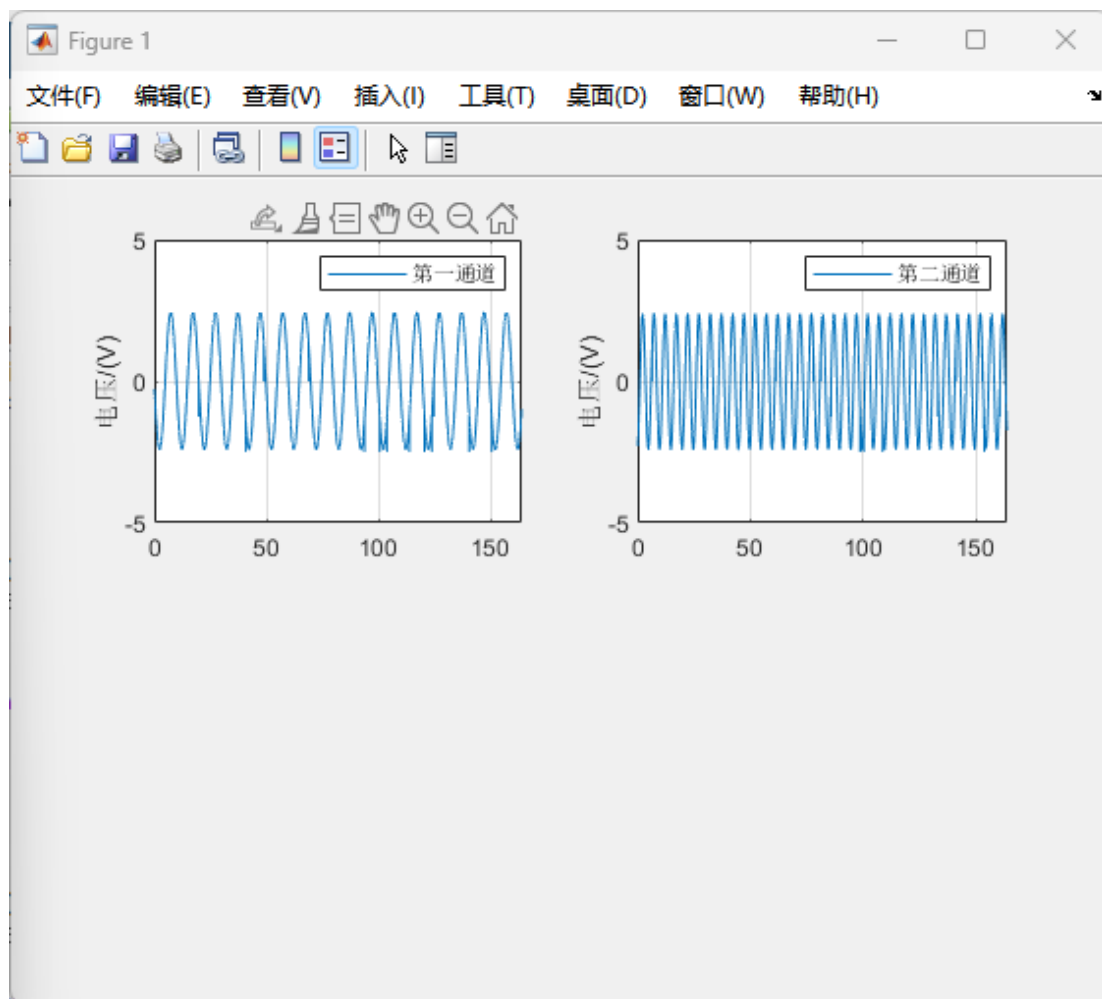


图 1-26 MATLAB 分析波形图

前面我们提到过本次实验提供的信号源为通道一输入 100Khz， $v_{pp}=5V$ 的正弦波、通道二输入 200KHZ， $v_{pp}=5V$ 的正弦波，与 MATLAB 分析出来的波形频率一致。若未安装 matlab，可通过小梅哥数据采集仪绘制波形。

（注：acd9238 每通道 65Msps 最高采样速率，采用 100M 采样波形会有所失真）

1.4.4.2 小梅哥数据采集仪通信

上位机暂未添加基于 pcie 采样功能，后续待更新。

数据采集仪界面设备选择 ACM9238，勾选双通道：

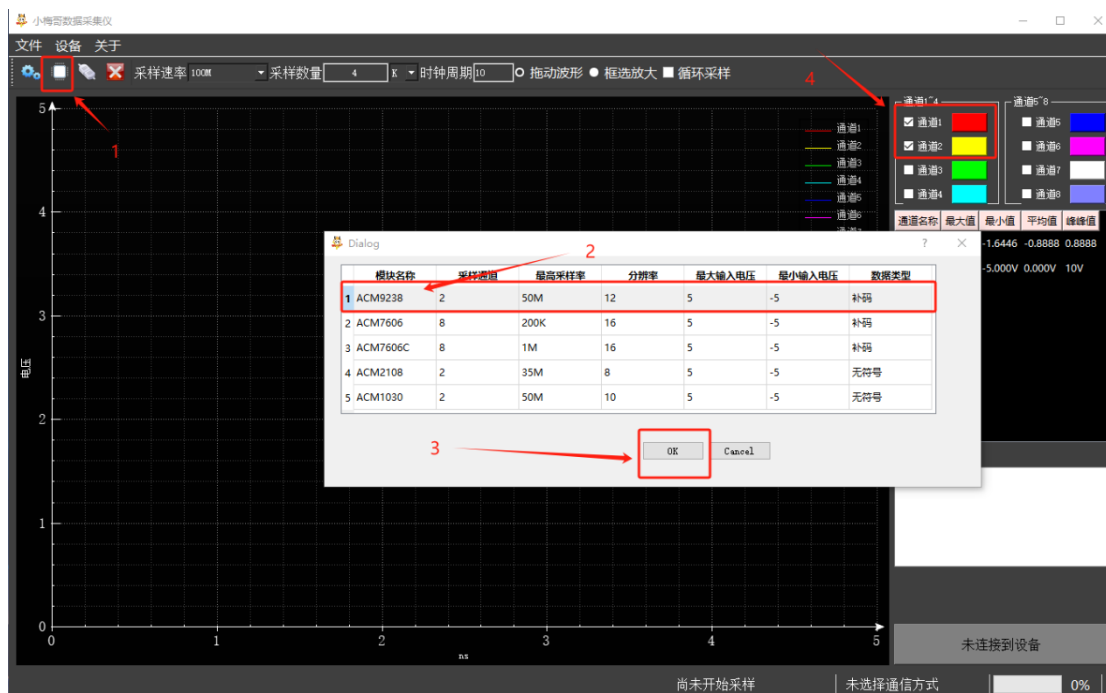


图 1-27 小梅哥数据采集仪界面

点击文件选择导入数据文件

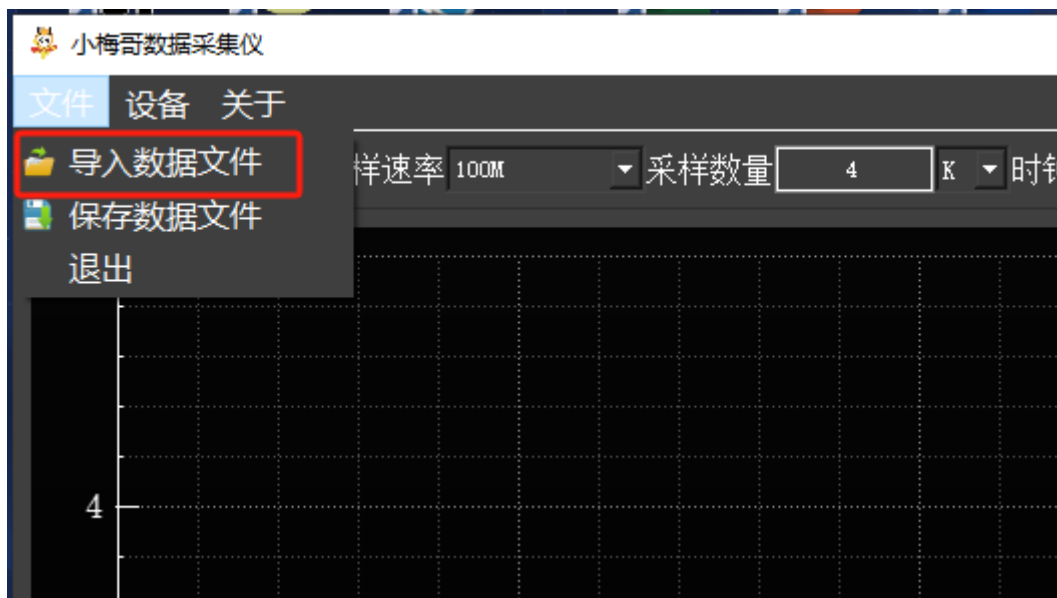


图 1-28 小梅哥数据采集仪界面

选中需要导入的数据文件并单击打开：

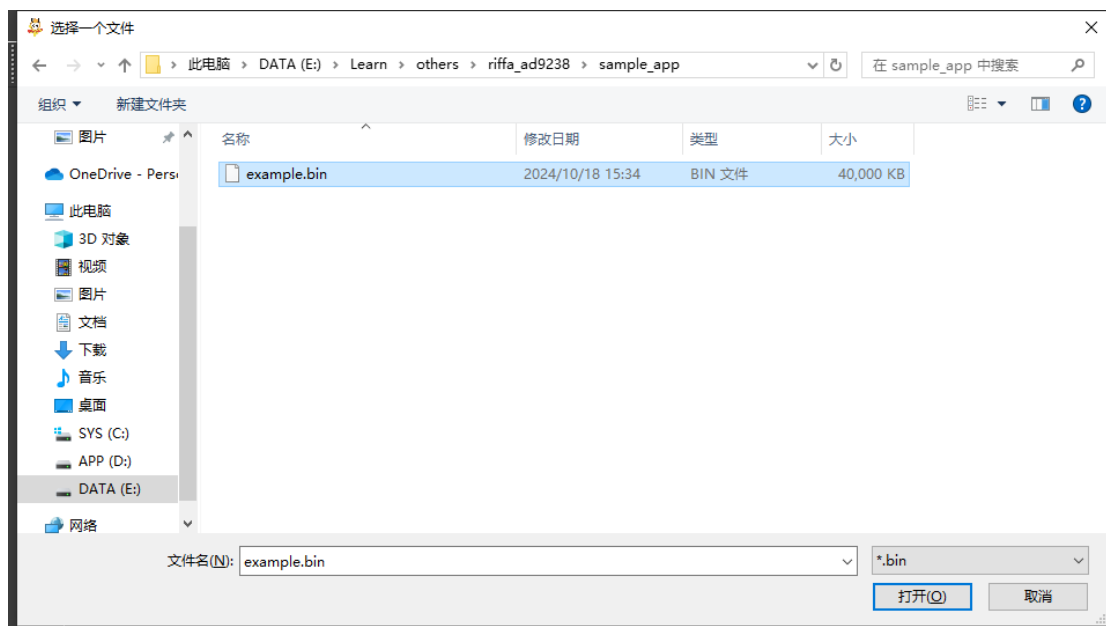


图 1-29 选择采集数据界面



图 1-30 小梅哥数据采集仪界面

(注：acd9238 每通道 65Msps 最高采样速率，采用 100M 采样波形会有所失真)

1.4.4.3 波形频率测量

波形频率测量基于小梅哥数据采集仪而来，matlab 分析请读者自行进行测试。

右键单击小梅哥数据采集仪波形界面，选择设置光标：



图 1-31 小梅哥数据采集仪界面

勾选光标 1、设置通道 1、左键单击确定



图 1-32 小梅哥数据采集仪光标设置界面

选择想要的测量点 1 并且左键单击：

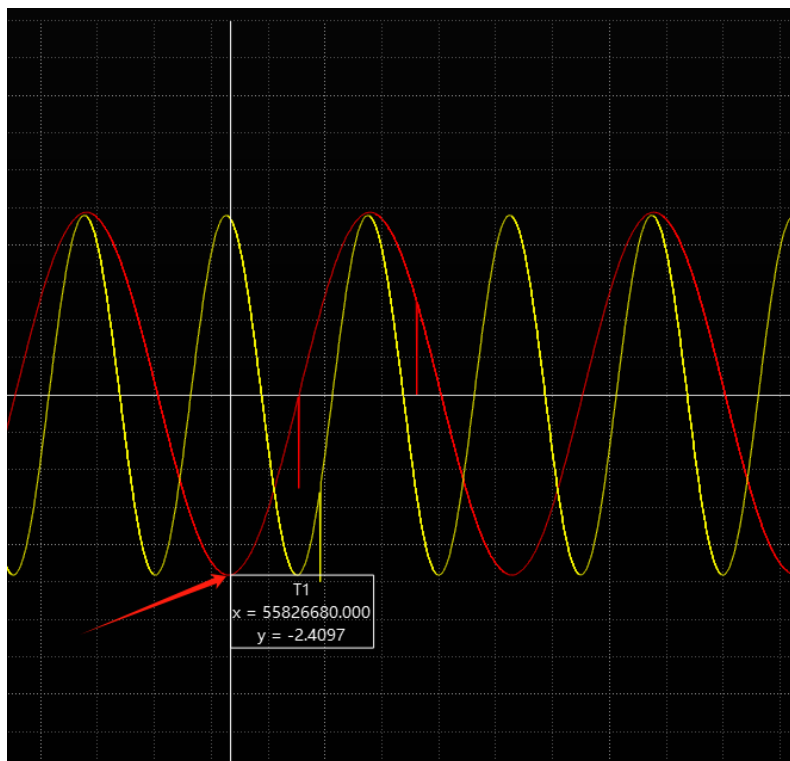


图 1-33 小梅哥数据采集仪界面

同理添加光标 2：



图 1-34 小梅哥数据采集仪光标设置界面

添加完成后选择测量点 2:



图 1-35 小梅哥数据采集仪界面

如上图所示当前通道 1 测量频率为 100_000hz，与输入相符。

通道 2 同理可得：



图 1-36 小梅哥数据采集仪界面

1.5 思考与总结

本次实验介绍了采用 PCIE 接口、基于 riffa 框架的 ACM9238 的数据采集，用户通过 pcie 向开发板发送指令数据配置控制 ACM9238 模块的寄存器，以此控制 ADC 进行采样，并将数据缓存后再通过 pcie 发送至电脑，借由 matlab 工具对数据进行查看。如果使用我们提供的上位机软件，则不需要自己通过 matlab 对数据进行绘制，只需要在界面上导入波形文件，便可以在波形显示界面观察到波形变化。

本次实验涉及时钟域的转换，以及采集数据位宽的转换，完成这些转换需要对 FIFO 有一定的了解，想要对 FIFO 进一步了解可以参看逻辑教程中“IP 核使用之 FIFO”一节的内容。