1 ADC128S102 数据采集 USB2.0 数据发送实验

1.1. 文件说明

本文档如有更新，会发布在 www.corecourse.cn 网站。如您需查看并获取更新版本，只需要在该网站以“ADC128S102”为关键词搜索，即可查找到相关内容。

版本记录：

V1.0	首次发布	2025.4.14
……	……	……

本教程配套的文件压缩包名：ACX720_V3_ADC128S102_ddr3_usb.zip，解压后可以得到配套工程文件、上位机软件和 matlab 绘图代码文件，对于压缩包中的文件说明如下：

 ADC128S102_ddr3_usb	ADC128S102_ddr3_usb：存放本次实验使用到的 vivado 工程。其中包括各个模块的源文件代码：USB 数据流接收控制模块、USB 指令接收和指令解析模块、ddr3 的含 fifo 的 2 端口封装模块、USB 数据流发送控制模块、ADC128S102 控制器模块等。
 FX2_USB 数据采集上位机工具	FX2_USB 数据采集上位机工具文件夹： cypress 上位机，用来烧写 USB 固件程序以及简单的数据发送接收测试；Win7_8_10_Drivers 文件，首次使用 USB 时，需要使用该文件夹更新驱动；slave_for_adc_clk_not.iic，烧写 USB2.0 芯片固件时需要的文件。
 小梅哥控制台For ADC采集v4.exe	小梅哥控制台 For ADC 采集 V4：使用该上位机软件采集数据时，无需设置发码指令，只需要在软件界面中填好采集信息即可，相比于通过 cypress 上位机进行数据采集，操作方式更简单直接。
 ADCdata_to_wave_v2_2.m	ADCdata_to_wave_v2_2.m：Matlab 源文件代码，对于 USB 传输过来的数据，我们得到的只是一个数据文件，如何判断采集到的数据是否正确，这里采用 Matlab 软件进行数据数据分析，用户可以通过修改我们提供的 matlab 源文件中的文件路径，进行数据分析。

1.2. 实验平台说明

1. ACX720 型 FPGA 开发板 V3：板载 USB2.0 接口物理层芯片，GPIO0 接口。
2. 数据采集模块：板载 ADC128S102 模块。
3. 支持 USB2.0 接口，操作系统为 vista，win7，win8 或 win10 的 PC 机。
4. 信号发生器：输出信号给数据采集模块 ADC128S102。

1.3. ADC 芯片概述及电路设计

1.3.1. ADC128S102 型 ADC 内部工作原理

模数转换器的种类很多，按工作原理的不同，可分成间接 ADC 和直接 ADC。

间接 ADC 是先将输入模拟电压转换成时间或频率，然后再把这些中间量转换成数字量，常用的有双积分型 ADC。直接 ADC 则直接转换成数字量，常用的有并联比较型 ADC 和逐次逼近型 ADC。

并联比较型 ADC：采用各量级同时并行比较，各位输出码也是同时并行产生，所以转换速度快。并联比较型 ADC 的缺点是成本高、功耗大。

逐次逼近型 ADC：它产生一系列比较电压 VR，但它是逐个产生比较电压，逐次与输入电压分别比较，以逐渐逼近的方式进行模数转换。它比并联比较型 ADC 的转换速度慢，比双分积型 ADC 要快得多，属于中速 ADC 器件。

双积分型 ADC：它先对输入采样电压和基准电压进行两次积分，获得与采样电压平均值成正比的时间间隔，同时用计数器对标准时钟脉冲计数。它的优点是抗干扰能力强，稳定性好；主要缺点是转换速度低。

在 XC7Z020 EDA 扩展板上使用的模数转换器为逐次逼近型的低功耗芯片 ADC128S102，其具有 8 通道以及 12 位的分辨率。电源采用独立的模拟供电以及数字供电，其中模拟电源 VA 输入范围为 2.7V~5.25V，数字电源 VD 输入范围为 2.7V~VA。其与外部通信支持多种接口如：SPI、QSPI、MICROWIRE 以及通用的 DSP 接口。转换速度在 500 kps~1 Mkps，典型情况下当 3V 供电时功耗为 2.3mW，5V 供电时为 10.7mW，如下图为该 ADC 芯片的内部结构图。

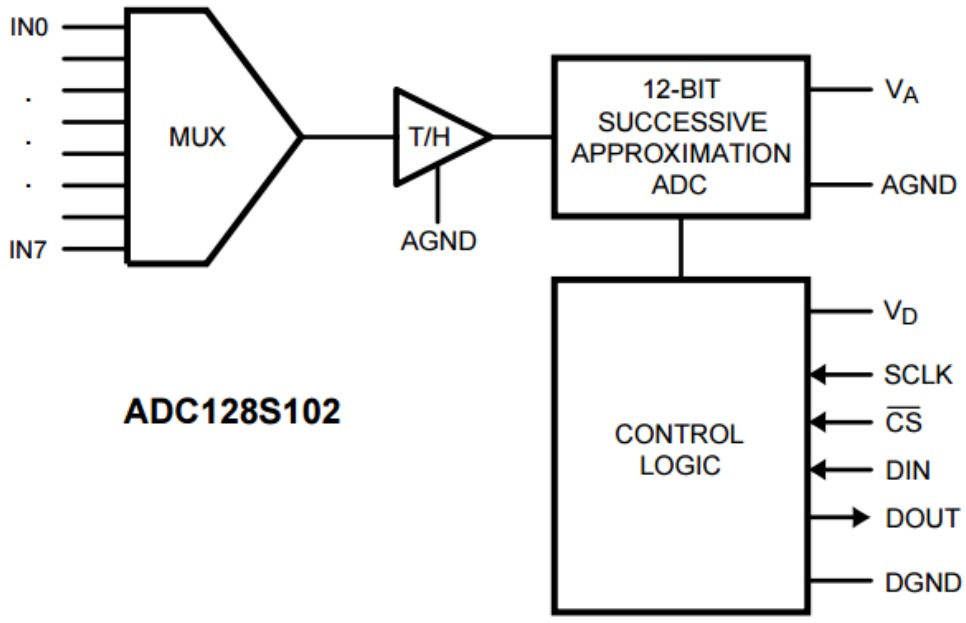


图 1-1　ADC128S102 内部功能框图

注意，图中左侧的 IN0~IN7 就是这个模数转换芯片（ADC）的模拟电压输入端，需要测量的模拟电压就通过这 8 个脚输入给该 ADC。右侧的 SCLK、CS、DIN、DOUT 是 ADC 的数字通信接口，该接口遵循 SPI 协议，ADC 完成了模数

转换后，对输入的模拟电压转换得到一个量化的 12 位数字值，这个数字值需要通过该 SPI 接口传递给主控制器（FPGA、MCU、DSP）。

本款 ADC 为 12 位分辨率，因此 1bit 代表的电压值即为 VA/4096。手册中同时指出，当模拟输入电压低于 VA/8192 时，输出数据即为 0000_0000_0000b。同理由于芯片本身内部构造当输出数 0000_0000_0000b 变为 0000_0000_0001b 时，实际输入电压变化为 VA/8192 而不是 VA/4096。当输入电压大于等于 VA-1.5* VA/4096，输出数据即为 1111_1111_1111b。

1.3.2. ADC128S102 型 ADC 芯片引脚功能

ADC128S102 芯片引脚及功能描述如下表所示。

表 1-1 芯片引脚功能描述

引脚名	编号	I/O	功能描述
$\overline{CS}$	1	I	片选端，下降沿时开始测量转换，低电平状态为正在转换
VA	2	供电	模拟电源输入正，也就是参考电压。2.7V~5.25V
AGND	3	供电	模拟地
IN0~IN7	4~11	I	模拟输入脚，电压输入范围为 0~VREF
DGND	12	供电	数字地
VD	13	供电	数字电源输入正，2.7V~VA.
DIN	14	I	串行数据输入脚，SCLK 上升沿采样输入
DOUT	15	O	转换结果输出脚，SCLK 下降沿输出
SCLK	16	I	时钟输入，频率范围为 8 MHz -16 MHz。

明确了各引脚功能后，接下来介绍 ADC128S102 电路设计。

1.3.3. ADC128S102 电路设计

ADC128S102 的相关原理图如下图所示。这里外接了一个抗混叠低通滤波器，其作用是为了避免输入信号的高频成分在 ADC 的基带中引起混叠。数字电源与模拟电源电压均为 3.3V，且数字电源与模拟电源之间串联磁珠用于抑制电磁干扰。

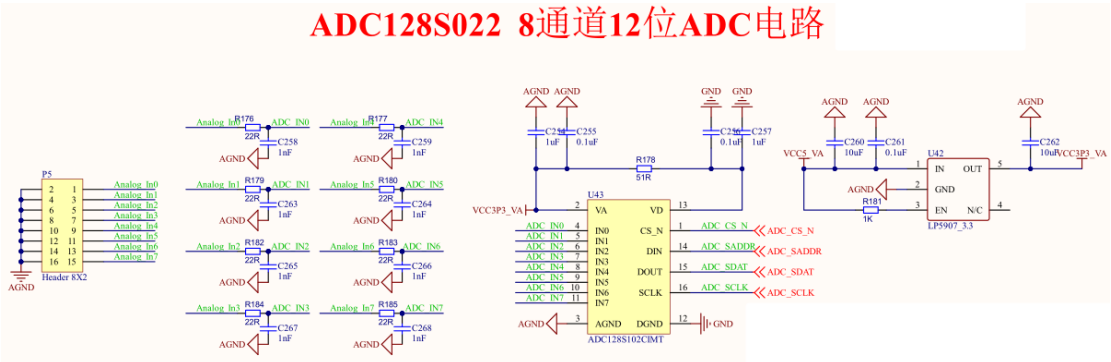


图 1-2 ADC128S102 电路图

介绍完 ADC128S102 部分电路后，接下来讲解其接口时序。

1.3.4. ADC128S102 型 ADC 接口时序

微控制器与 ADC128S102 的接口如下图所示，该接口使用标准的 SPI 接口，因此可以直接连接到微控制器的片上 SPI。对于 FPGA 来说，则可按照 SPI 时序搭建控制电路，以实现对 ADC128S102 的控制。

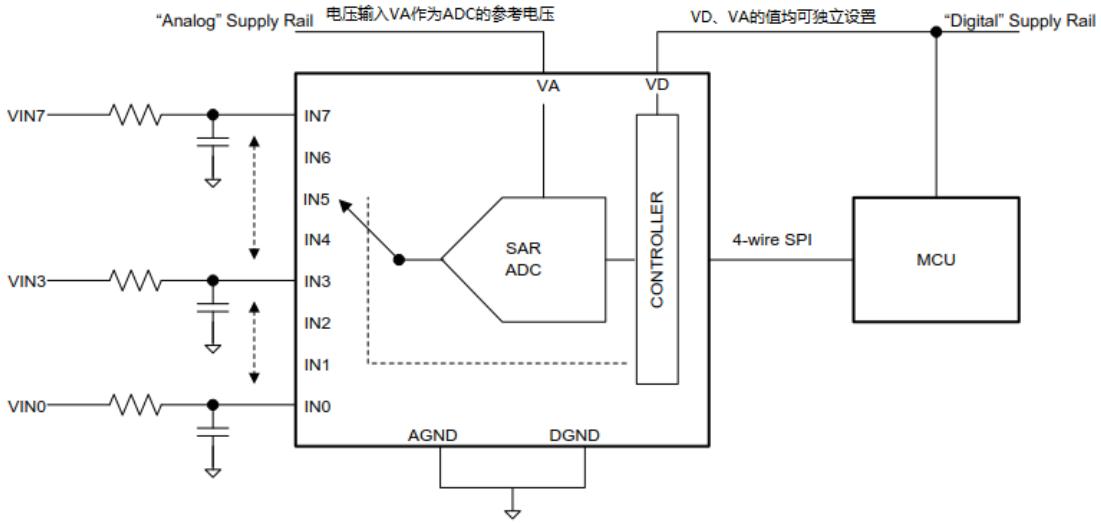


图 1-3 主机与 ADC128S102 接口示意图

ADC128S102 通过 SPI 接口与控制器进行通信的时序图如下图所示。从图中可以得出如下信息：

1. 一个串行帧开始于$\overline{CS}$的下降沿，结束于$\overline{CS}$的上升沿。

2. 一帧包含 16 个上升沿 SCLK。ADC 的 DOUT 引脚当$\overline{CS}$为高时代表空闲状态，当为低时为传输状态。也就是相当于$\overline{CS}$可以充当输出使能。在$\overline{CS}$为高时 SCLK 默认高。

3. 在头三个 SCLK 的循环，ADC 处于采样模式（track）。在接下来的 13 个循环为保持模式（hold），转换完成并且完成数据输出。

4. 下降沿 1~4 为前导零，5~16 为输出转换结果。如果在持续测量模式中，ADC 在 N*16 个 SCLK 的上升沿会自动进入采样模式，在 N*16+4 个下降沿进入保持/转换模式。

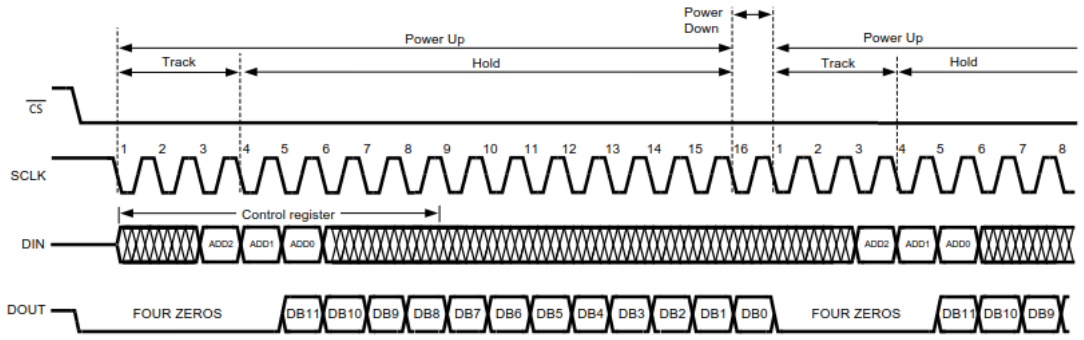


图 1-4 SPI 接口时序图

5. 到达采样模式的相应节拍后，进行采样的判据有三种描述方式，第一种当 SCLK 为高电平时$\overline{CS}$变为低，当 SCLK 第一个下降沿到来时便进入采样模式，如时序图所示；第二种当 SCLK 为低电平时$\overline{CS}$变低，自动进入采样模式，$\overline{CS}$的下降沿即视为 SCLK 的第一个下降沿；第三种 SCLK 与$\overline{CS}$一同变为低，这样对于两信号上升沿没有时序约束，但对于其下降沿有要求。

在 DIN 的 8 个控制寄存器，每一位代表的含义如下表所示。

表 1-2 DIN 控制寄存器

Bit7(MSB)	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DONTC	DONTC	ADD2	ADD1	ADD0	DONTC	DONTC	DONTC

其中 ADD[2:0]代表的输入通道选择如下表所示。

表 1-3 ADD 寄存器与输入通道选择对应表

ADD2	ADD1	ADD0	输入通道
0	0	0	IN0 (Default)
0	0	1	IN1
0	1	0	IN2
0	1	1	IN3
1	0	0	IN4
1	0	1	IN5
1	1	0	IN6
1	1	1	IN7

需要特别注意的是：ADC 虽然有 8 个通道，但无论如何同一时刻只能采样一个通道。8 个通道如果想都采到结果，那就得采样 8 次，每次选择不同的通道。

1.4. 实验内容说明

本次实验实现的内容如下所示：

1. 实验通过板载数据采集模块实现模数转换。
2. 使用 CyControl 软件，通过 USB 发送指令，可以设定需要采集的字节数，

选择采集通道号，设置采样速率，启动采集。或者直接使用我们提供的“小梅哥控制台 For ADC 采集 v4”进行数据采集，得到的波形也可以通过上位机观察到，更方便快捷。

3. 采集到的数据经过 matlab 波形分析，能够得到和输入波形一致的输出波形，无数据丢失，无杂波。后期可结合实际情况，增加对噪声评定的环节。

根据实验所要实现的功能，得到模块的整体设计如下图 1.1 所示：

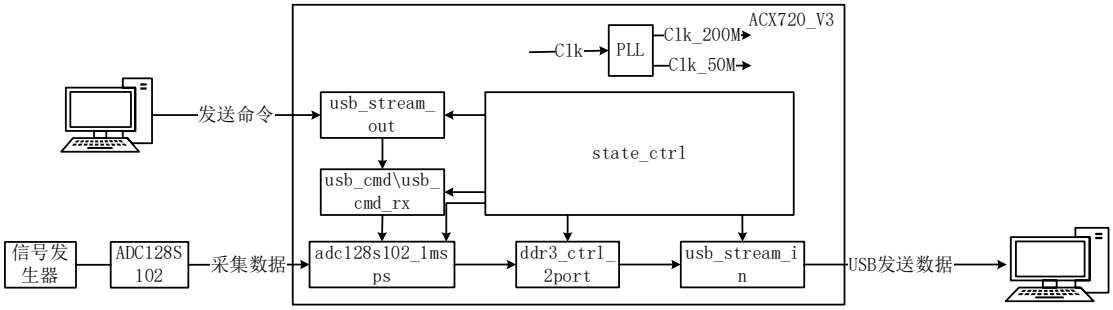


图 1.1 系统的整体设计框图

对于 ACX720 中每个模块的功能介绍如下所示，具体代码请自行查看工程中的源文件：

1. usb_stream_out：USB 接收数据模块，接收 PC 发送过来的指令。
2. usb_cmd\usb_cmd_rx：USB 指令接收和指令解析模块，将收到的指令进行翻译拆解，指令分类。
3. adc128s102_1msps：ADC128S102 控制器模块，该控制器实现了对 ADC128S102 型 8 通道 16 位 ADC 的数据转换控制并输出。使用该控制器时，用户无需关心 ADC128S102 的具体控制时序，一切都在控制器内部完成。
4. ddr3_ctrl_2port：ddr3 的含 fifo 的 2 端口封装模块，主要负责整个数据的存储功能。
5. usb_stream_in：USB 数据流发送模块，将最终采集到的数据通过 USB 发送出去。
6. state_ctrl：状态机模块，协调各个模块的信号控制，程序状态的总控制模块。
7. pll：锁相环模块，50M 时钟输入，输出 200M 的时钟给 DDR3，50M 的时钟给其他模块。由于我们在设计时，对于时钟频率没有要求的模块，都可以采用 50MHz 的频率进行设计，而 DDR3 驱动模块的工作时钟频率最低也得 200MHz 才能支持正常工作，所以我们需要使用锁相环模块

完成整个工程的频率配置，以实现在工程中引入两种时钟频率的设计需求。

1.5. Vivado 和 cypress 工作环境的搭建

Vivado 下载环境搭建，详细内容可以参考小梅哥 xilinx FPGA 自学教程第一章相关内容，简单概况为：软件解压，设置安装组件，设置非中文安装路径，安装，破解。

本次实验 ACX720 开发板完成 USB 采样的连线方法如下图 1. 2 所示：

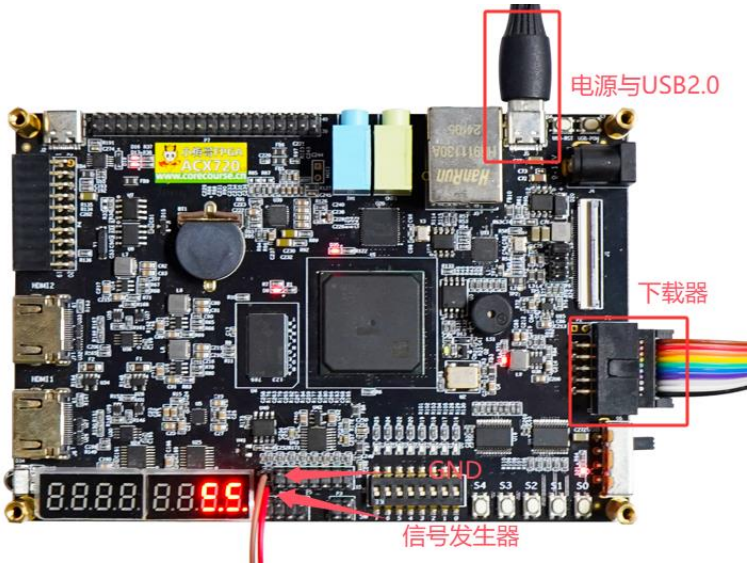


图 1. 2 ACX720 开发板完成 USB 采样硬件连接图

1.5.1. USB 驱动安装

USB 通信线连接到 PC 机，直接 USB 连线即可，无需任何端口号或接口号设置。但首次使用时，需在设备管理器中安装驱动程序。其步骤如下所示：

1. 使用 USB 线连接开发板 USB 接口和电脑的 USB 接口，此时，系统右下角应该会弹出发现新硬件，提示驱动安装失败，如下图 1. 3 所示。



图 1. 3 驱动安装失败示意图

2. 此时查看设备管理器中，应该能看到如下所示的一个未知设备，如下图 1. 4 所示。

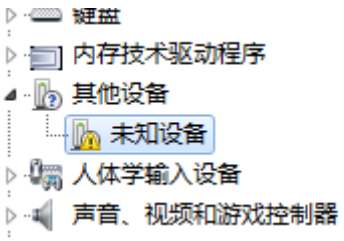


图 1.4 设备管理器显示图

3. 我们选中该设备，点击鼠标右键，即可查看其详细信息，显示如下图 1.5 所示。在详细信息这一栏我，我们在属性的下拉表中选择“硬件 ID”，可以看到，其值为 USB\VID_040B&PID_8613，这就是我们的 USB2.0 模块的硬件 ID。

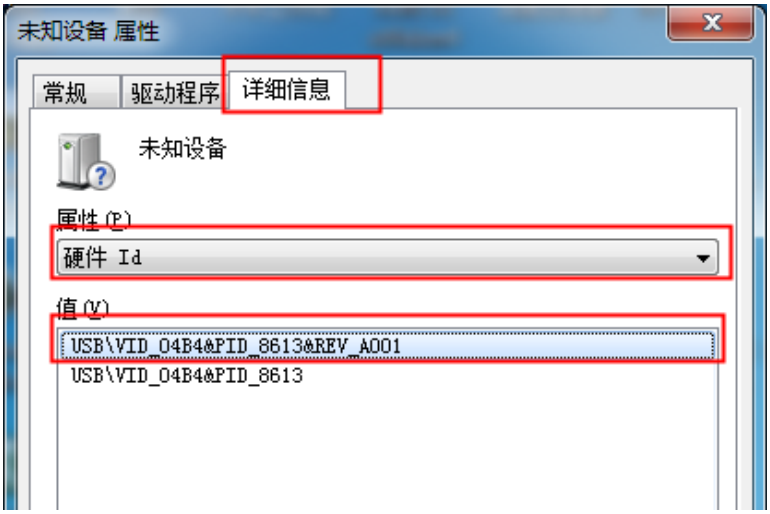


图 1.5 设备详细信息示意图

4. 在属性选项卡中，切换到驱动程序选项卡，选择“更新驱动程序”，如下图 1.6 所示。

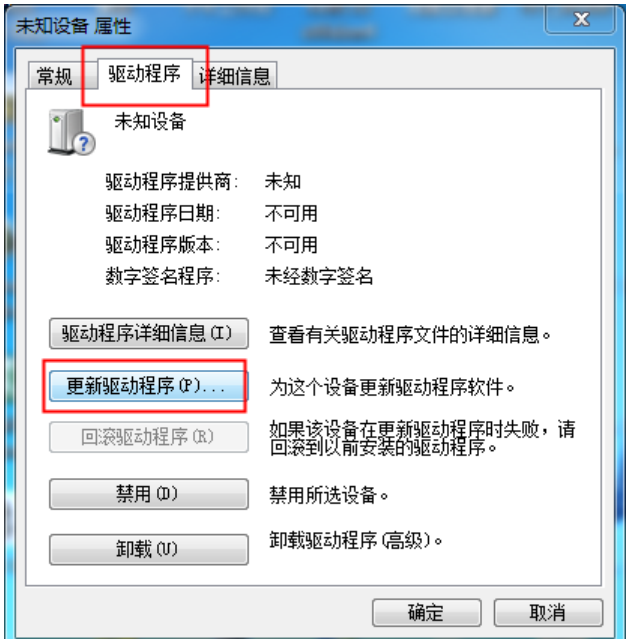


图 1.6 更新驱动设备

5. 在弹出的界面中，选择“浏览计算机以查找驱动程序软件”，路径定位到我们提供的 USB 开发包的驱动程序安装路径下，然后点击确定，如图 1.7 所示。

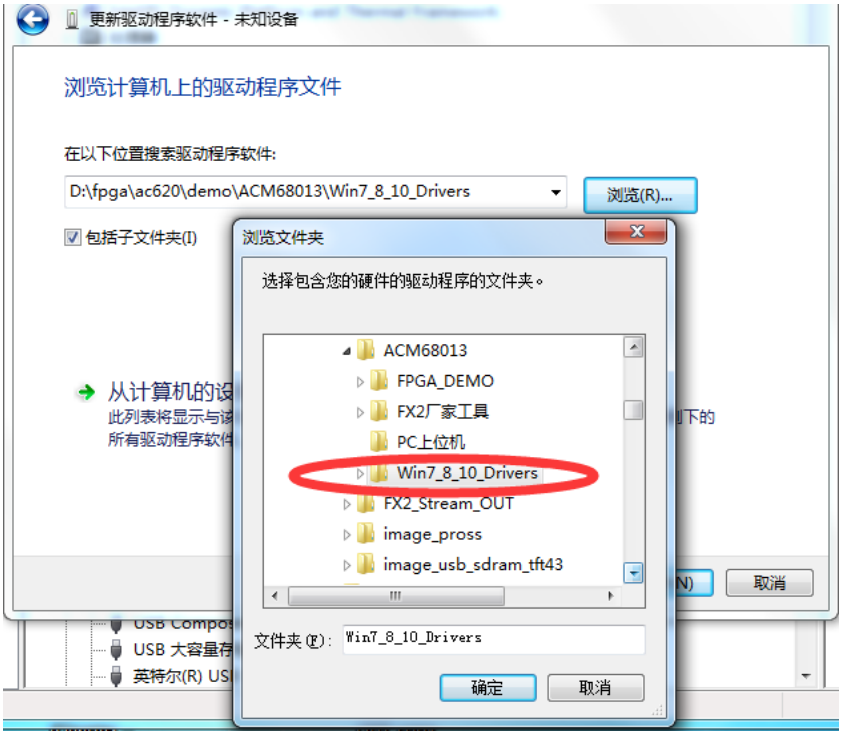


图 1.7 安装驱动设备

6. 接着点击下一步，就会开始驱动的安装，安装完成后，弹出如下图 1.8 所示的界面，表明驱动安装完成，点击关闭按钮完成安装。

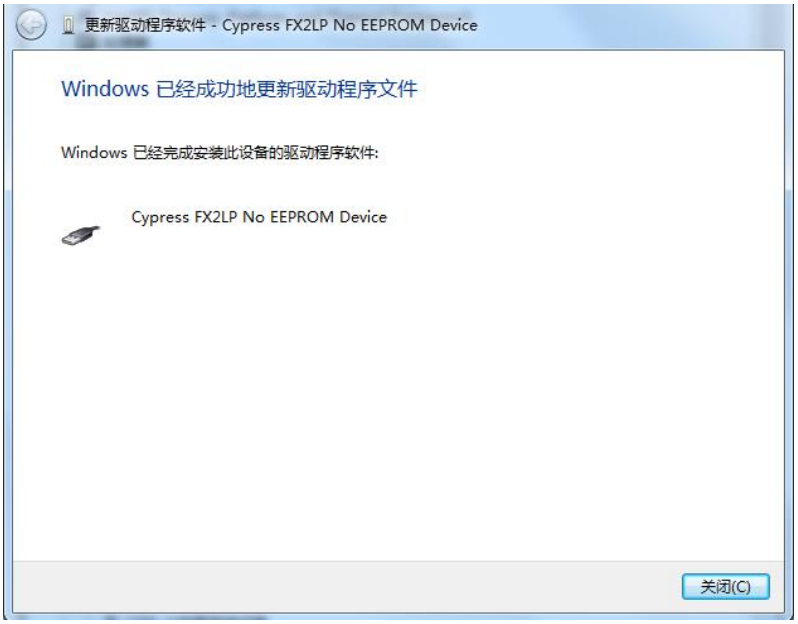


图 1. 8 驱动安装成功示意图

通过上述步骤就完成了 USB 驱动的安装，接着我们回到设备管理器中，在通用串行总线控制器下，可以看到如下图 1. 9 所示的设备，且无叹号提示异常，则表明驱动程序安装正确，安装成功之后，也可能显示的是“Cypress FX3 USB StreamExample Device”，这是因为我们在出厂固件中就已经烧写了对应的固件程序，这样用户就不用再烧写 USB2.0 芯片固件。



图 1. 9 驱动安装成功之后设备管理器显示

1.5.2. 烧写 USB2.0 芯片固件

在我们提供的文件夹中，有一个名为“FX2_USB 数据采集上位机工具”的文件夹，该件夹中包含了一个名为 CyControl.exe 的软件和 CyUSB.dll 的动态链

接库，如下图 1. 10 所示。这个 CyControl.exe 就是 Control Center 软件，我们直接使用这个软件，就可以完成 FX2 芯片的固件烧写和基本的数据传输测试。

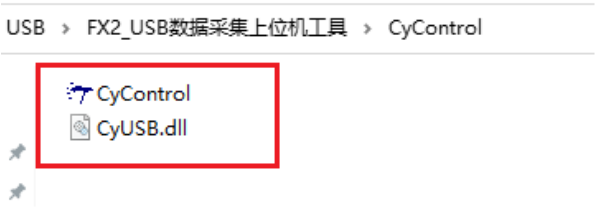


图 1. 10 烧写 USB 固件对应软件

双击 CyConsole.exe 打开软件，如果已经使用 USB 线连接开发板 USB 接口和电脑的 USB 接口，则会显示出一个名为 Cypress FX2LP No EEPROM Device 的设备，如下图 1. 11 所示。（如未识别可按下 USB 接口旁的两个 RST 和 PGM 键）

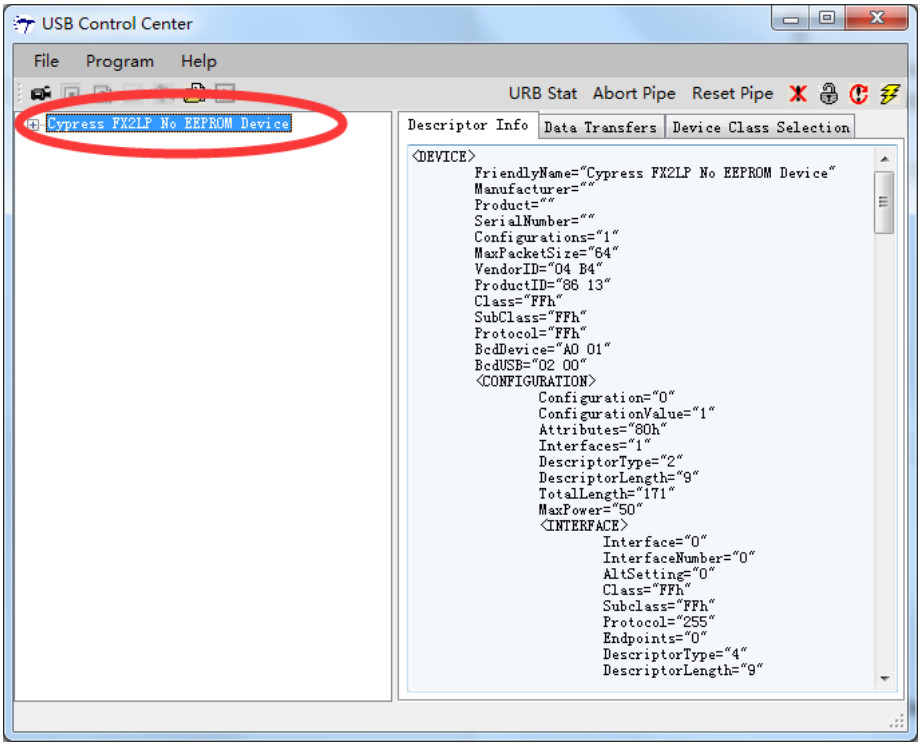


图 1. 11 USB Control Center 显示界面

鼠标选中该设备，点击 Program -> FX2 -> 64KB EEPROM 如下图 1. 12 所示。

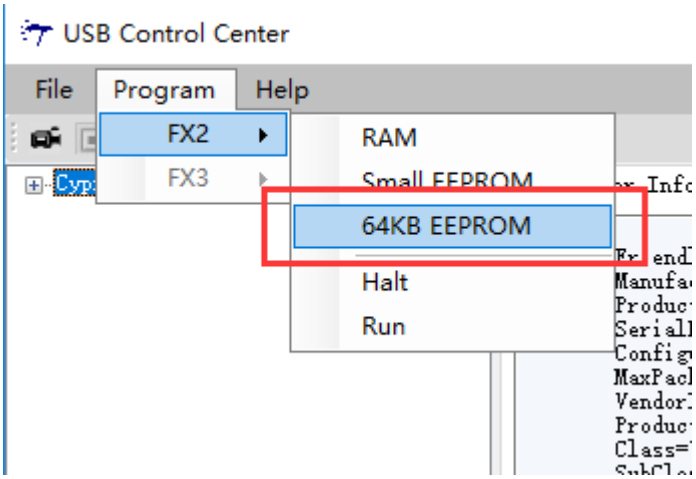


图 1. 12 选择烧写文件存储位置

选择 64KB EEPROM 之后，就会弹出一个选择文件的对话框，在该对话框中，定位路径到工程文件包路径下，烧写 slave_for_adc_clk_not.iic 文件（位于“FX2_USB 数据采集上位机工具”文件夹中）到 USB 芯片的 EEPROM 中。下次 FX2 接入电脑，或者按下开发板背面的 USB-RST 按键，FX2 就会自动从 EEPROM 中读取程序并加载到芯片的 RAM 中运行，可以实现掉电不丢失，如下图 1. 13 所示。

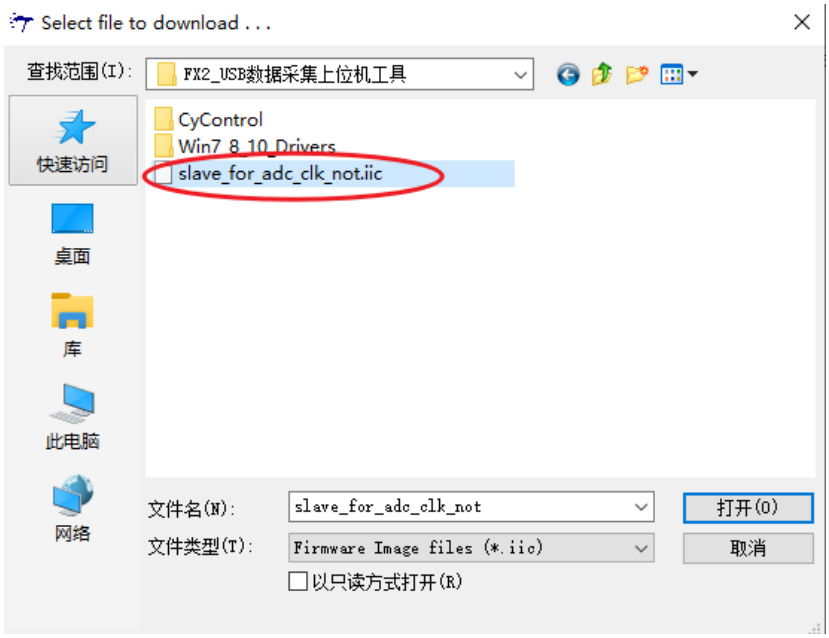


图 1. 13 烧写 IIC 文件

这样，我们就完成了开发板上 USB 芯片的程序固件烧录工作。然后在 Control Center 软件中可以看到，该设备的识别名称已经变为了 Cypress FX3 USB StreamExample Device，之所以识别为 FX3，是因为 FX2 和 FX3 的这个例子使用了同一个 PID 和 VID，所以被识别为了 FX3，但是这并不影响我们的正常使用。

此时打开设备管理器，也可以看到设备名称变成了 FX3，如下图 1. 14 所示。



图 1. 14 烧写成功之后，设备管理器显示界面

1.6. 实验功能测试

1.6.1. 硬件连接

要验证本次实验，首先第一步就是对本次实验的整个系统硬件进行连接，本次工程硬件具体的连接如下图 1. 15 所示。

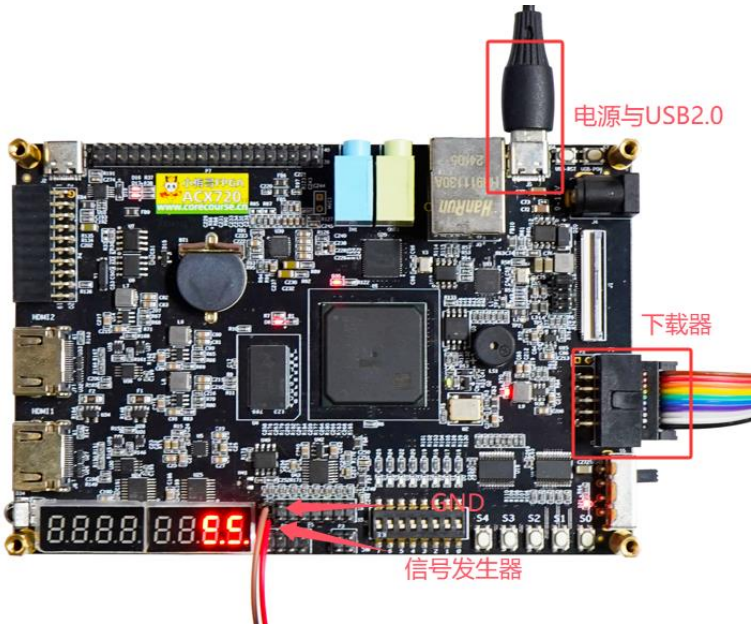


图 1. 15 硬件连接图

本次使用的信号源是通过信号发生器产生的，一端接 A1，一端接地。注意 USB 接口必须连接在 ACX720 板子的右侧接口，电源线可直接使用 USB 接口，也可以单独连接电源线，下载器连接如上图所示，在硬件连接好后打开开关就可

以下载程序。

1.6.2. 下载 bit 文件

打开本次实验的 vivado 工程，下载本次实验的 bit 文件，用户可以直接将我们提供的 bit 文件夹下的文件烧写进板中，也可以打开工程重新生成 bit 下载。

程序烧写完成后，开发板的 led0 和 led1 的灯会点亮，此时表明：锁相环工作正常，ddr3 初始化正常。

1.6.3. 发码指令设置

我们发码指令参数，总共分为以下类别：

1. 设置采样通道（Channel 寄存器，地址为 1）：如 55 A5 01 00 00 00 01 F0，对通道 1 进行采样。通道设置寄存器，共 8 位，对应 8 个通道的数据存储开关，如果某通道对应的设置位为 1，则该通道的采样结果就会被存入 DDR 并通过 USB 发送。
2. 设置采样个数（DataNum 寄存器，地址为 2）：如 0x55 0xA5 0x02 0x00 0x00 0x3E 0x80 0xF0，设置的采样个数为 16000 个（0x3E80）。注意，该寄存器设置的总共采集的数据个数，假设设置采集 100 个数据，而且设置 ChannelSel 为 0000_0011，则实际每个通道采样的次数就是 50，2 个通道的数据加起来就是 100 个。假设设置采集 100 个数据，而且设置了 ChannelSel 为 0011_0011，则实际每个通道采样次数就是 25，4 个通道加起来采集 100 个数据。
3. 设置采样速率（ADC_Speed_Set 寄存器，地址为 3）：如 0x55 0xA5 0x03 0x00 0x00 0x01 0xF3 0xF0，设置采样速率为 100K。该寄存器用来设置 ADC 每多久执行一次转换，由于 ADC 的最大采样速率为 125Ksps，所以可以通过设置该寄存器的值来让 ADC 的采样速率在 1~125Ksps 范围内调整，以适应不同的应用场景。该寄存器的值与采样速率关系为：ADC_Speed_Set=1000000000/20/speed-1，其中 speed 就是实际要设置的采样速率。（本次实验实际采样速率不可修改）
4. 启动传输（RestartReq 寄存器，地址为 0），重新开始采集请求寄存器，向该寄存器写入任意值即可启动新一轮的采样存储传输。

1.6.4. 使用 cypress 上位机的操作流程

下面将演示一次完整的使用 cypress 软件的操作流程，同时指出使用注意事项。

1. 使用 ADC128S102 模块的通道 1 进行采集。我们设置信号发生器的发生频率为 200hz，高电平为 3V 得到正弦波，准备采样 2048 个 16bit 数据，

那么就需要通过 cypress 上位机接收 4096 个 8bit 的数据.

2. 打开我们提供的资料包中的 cypress 软件（acx720_ADC128S102_ddr3_usb\ FX2_USB 数据采集上位机工具\ CyControl），如果 USB 连接正常并且软件按前叙步骤正确安装驱动，则可以看到列表框有 FX3 芯片或者 FX2 芯片被识别。确认固件已经正确烧写后，找到 bulk out endpoint（0x02），输入指令串，比如按前述：我们使用通道 1，采样率设置为 200k，采样 2048 个 16bit 数据，我们可以设置指令串如下图 1. 16 所示：

   55 A5 02 00 00 08 00 F0 55 A5 01 00 00 00 01 F0 55 A5 03 00 00 00 F9 F0 55 A5 00 00 00 00 00 F0

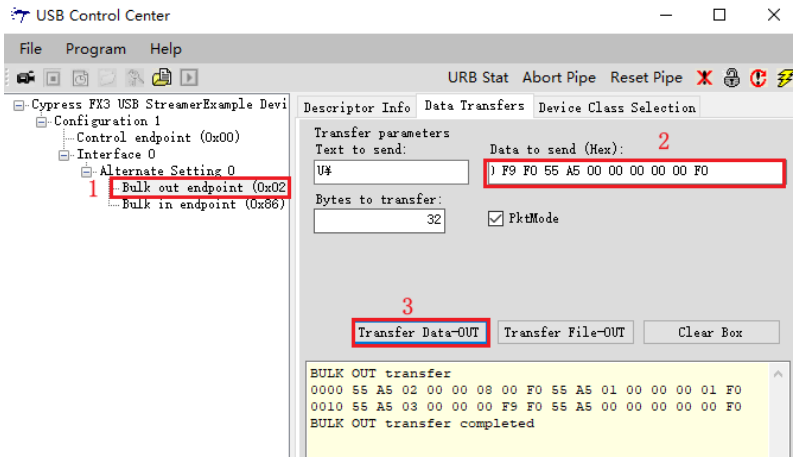


图 1. 16 发送指令软件示意图

3. 切换到 bulk in endpoint，确认 PktMode 被勾选，同时按照自己的采样设置需求，设定接收数据量，操作步骤如下图 1. 17 所示。完成设定后，点击 Transfer File-IN，就可以接收到需要采样的数据，设置保存的文件名及路径，如下图 1. 18 所示。

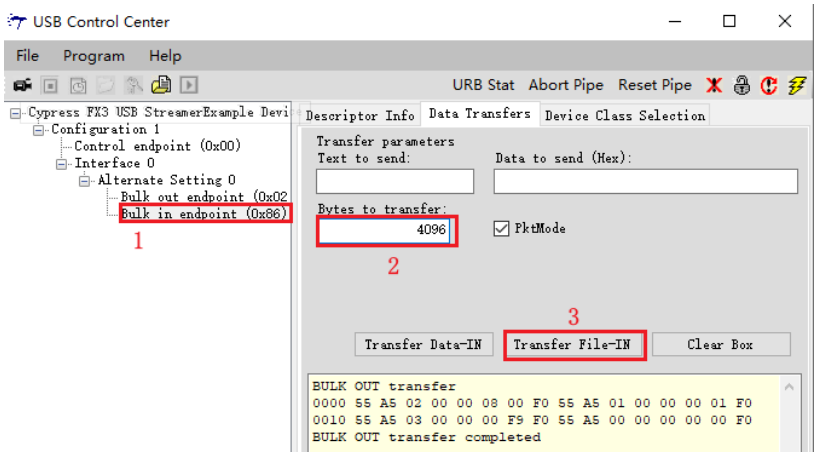


图 1. 17 采样数据



图 1. 18 保存接收到的数据

这里，有几点需要再次强调：

1. 上位机接收数据前，一定确认 pktmode 勾选。
2. 采样数量必须为 256 个 16bit 数的整数倍，即接收的 bytes to transfer，一定是 512 个 8bit 数据的整数倍。如果操作失误要求读数据的数量大于指令码设置的写入数据故障，会出现输出框为 997 的故障码。这时候建议将开发板复位，fx2 芯片复位以杜绝上次不正常读写的数据残留。开发板复位，按下 S4 按钮即可，fx2 芯片复位，在背面按下 USB-RST 按钮。
3. 如果是逐条发送指令，则必须确保启动指令在设置通道和设置采样数量指令之后，否则一旦发送采样启动指令，通道和采样数量设置将不会生效。
4. 目前最大采样数量经过测试支持 128M 的 16bit 数，即参数设置为 55 A5 02 08 00 00 00 F0。此时 cypress 软件最大读出量 stream in 为 268435456 的 8bit 数。（最大读出量 stream in 读出时，cypress 软件必须勾选 PktMode，且读出时间较长，不稳定。建议分批次读出，每次 stream in 读出小于等于 16777216 个数）。

1.6.5. 使用小梅哥控制台 For ADC 采集上位机的操作流程

如果小梅哥控制台 For ADC 采集上位机，则要简单很多，可以无需设置发码指令，只需要在软件界面中填好采集信息即可，如下图 1. 19 所示，可以看到界面右边也可以看到采样电压波形图，通过改变信号源，也可以看到波形图发生变化。

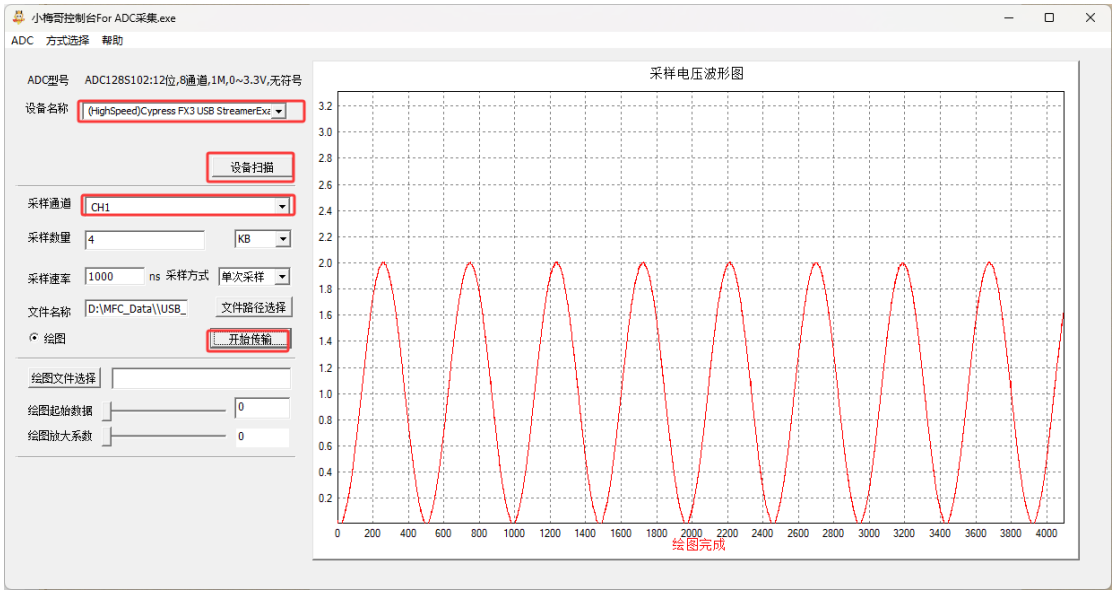


图 1. 19 通过上位机软件采集数据

初次使用，有可能需要进行扫描设备操作。这里我们使用的是通道 1（不支持 CH0 测试功能），采样速率不支持修改。

采样得到的数据，默认保存在 d 盘根目录下方。

采样数量可以设置最大为 4KB。如果采样数量设置过大，则有可能会占用较长的采样时间，具体可参考 USB 通信协议数据传输速率。

设定好以后，可以点击“开始采样”。

完成后点击“停止采样”，就可以得到采样数据的 bin 文件。