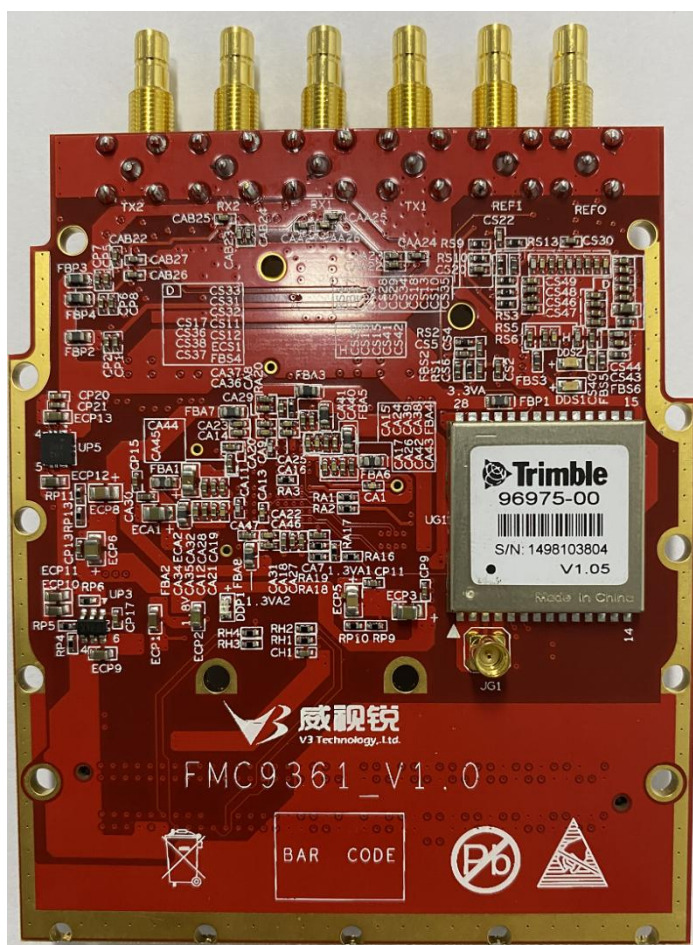


RFMC6000

FPGA 开发指南

Rev. 1.0



修订记录

版本	修订日期	修订内容
1.0	2018 年 1 月 24 日	初始版本

目 录

修订记录	2
目 录	3
1 概述	4
2 参考例程结构	4
2 寄存器控制说明	错误！未定义书签。
3 裸机应用程序	6
3.1 初始化板卡	7
3.2 运行参考例程	8
3.3 LOOPBACK 回环测试	8
3.4 射频回环测试	10
4 附录	12
自定义 IP 核的添加	12

1 概述

本文档主要介绍 RFMC6000 射频子卡的 FPGA 开发流程，包括以下几部分：

FPGA 内部生成 DDS 信号源，基于 ZYNQ 的 ARM 处理器定制，AD9361 的 SPI 配置。可以实现单音信号发送和接收的功能，并可以使用 `chipscope` 观察接收信号并导出到 `matlab` 分析。

进一步，射频子板上含有集成 VCO 的参考时钟生成 IC 选自 TI 的 CDCM6208，可以将 10MHz 基准时钟产生 AD9361 参考时钟，FPGA 开发流程可以实现用户自定义采样时钟，灵活用于各种通信场景。

通过本参考例程用户可以了解本系统的硬件构成，编程架构，对于做 FPGA 开发的用户可以基于这个 FPGA 工程做二次开发，实现 AD9361 空口传输基带数据。

威视锐提供了整体的 PS(ZYNQ 处理器部分)、PL (ZYNQ 逻辑部分)、AD9361 的配置，可以通过 FPGA DDS ipcore 输出指定频率的单音信号，并可以通过 `chipscope` 获取接收到的波形。基于这个工程，做 FPGA 基带算法开发的用户可以用于二次开发。

2 FPGA 硬件工程

本例程 PL 部分负责 AD9361 的 IQ 数据端口转换和收发数据，PS 端的 C 程序负责 AD9361 的配置。方便用户对 AD9361 进行学习。参考例程的结构框图如下：

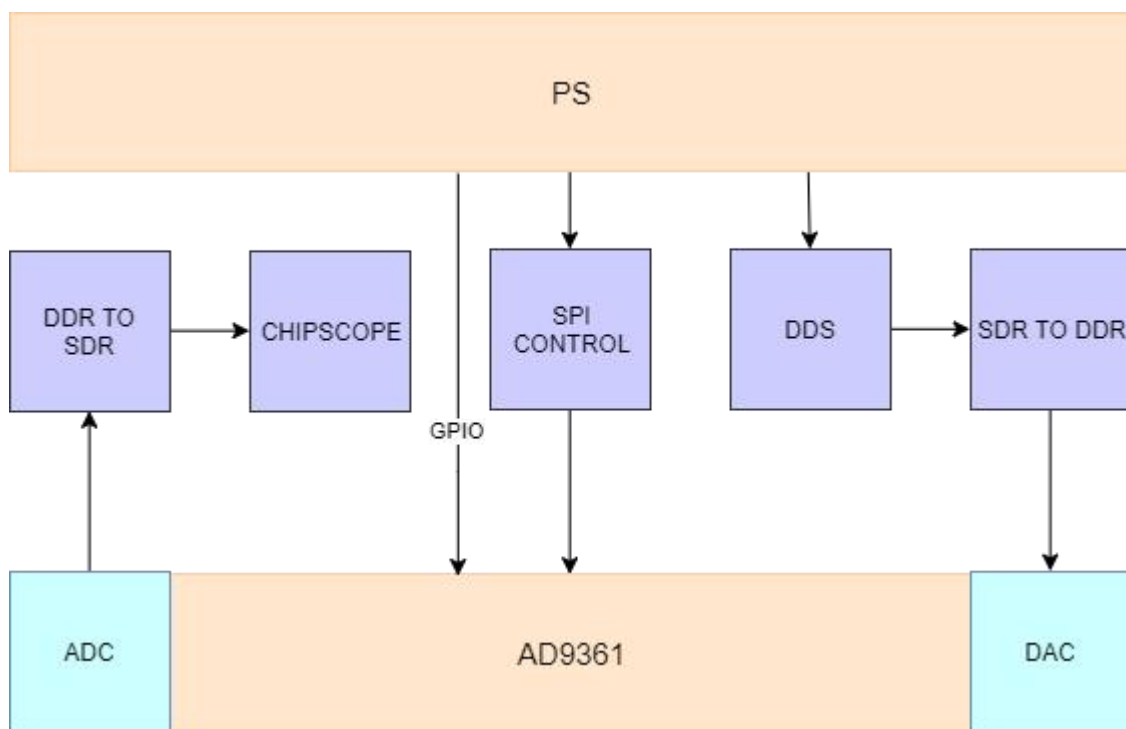


图 1 FPGA 参考程序框图

参考例程采用 verilog 语言设计，设计文件列表如下：

表格 1 文件列表

序号	文件名称	功能
1	adaloop.v	顶层文件
2	axi_ad9361_dev_if_idelay.v	AD9361 数字端口 DDR 转 SDR 匹配
3	axi_lite_drv.v	axilite 寄存器映射，PS 与 PL 的命令交互
4	ad_iobuf.v	双向缓冲模块

PS 与 PL 之间的命令交互通过 AXI_Lite 总线实现，为此自定义了 xilinx.com_user_axi_litev3_1.0 模块将 AXI 总线映射为寄存器表，映射位宽 6bit 地址空间，数据 32bit 位宽。地址总线的宽度可以通过双击 axi_litev3_1.0 模块，设定位宽实现更多的地址空间。

FPGA 工程中预定义的寄存器地址和功能如下：

表格 2 寄存器说明

参数	地址	方向	功能
sdr_info	0	RD	可以读出 SDR 模块的型号，320
mtime_start	1	WR/RD	PL 程序启动控制
pps_select	2	WR/RD	设定 pps 输入方式 0=内部，1=gps，2=外部
phase	3	WR/RD	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 = $sample_rate \div 2^{12} \times phase$ (MHz)
rf_mode	4	WR/RD	设定收发 IQ 数据方式 1: TX1/TX2 发送 DDS 序列 2: RX1 回环给 TX1, RX2 回环给 TX2 3: RX2 回环给 TX1, RX1 回环给 TX2 4: 只有 TX1 发送 DDS 序列 5: 只有 TX2 发送 DDS 序列 其他: 发送全零
clock_lpps	5	WR/RD	AD9361 rx_clk 频率，频率是 4 倍采样率
idly_en	21	WR/RD	REG21, 22 组合应用于 AD9361 RX 端 IQ 数据的 idelay 时序调整
idly_d	22	WR/RD	
rfl_ctrl_in	23	RD	AD9361GPIO 电平设定
user_io	24	WR/RD	用户 IO 设定值，或读取值
user_io_t	25	WR/RD	用户 IO 方向设定，0=输出，1=输入
adc_status	28	RD	指示 AD9361 RX 端 IQ 数据 idelay 调整状态

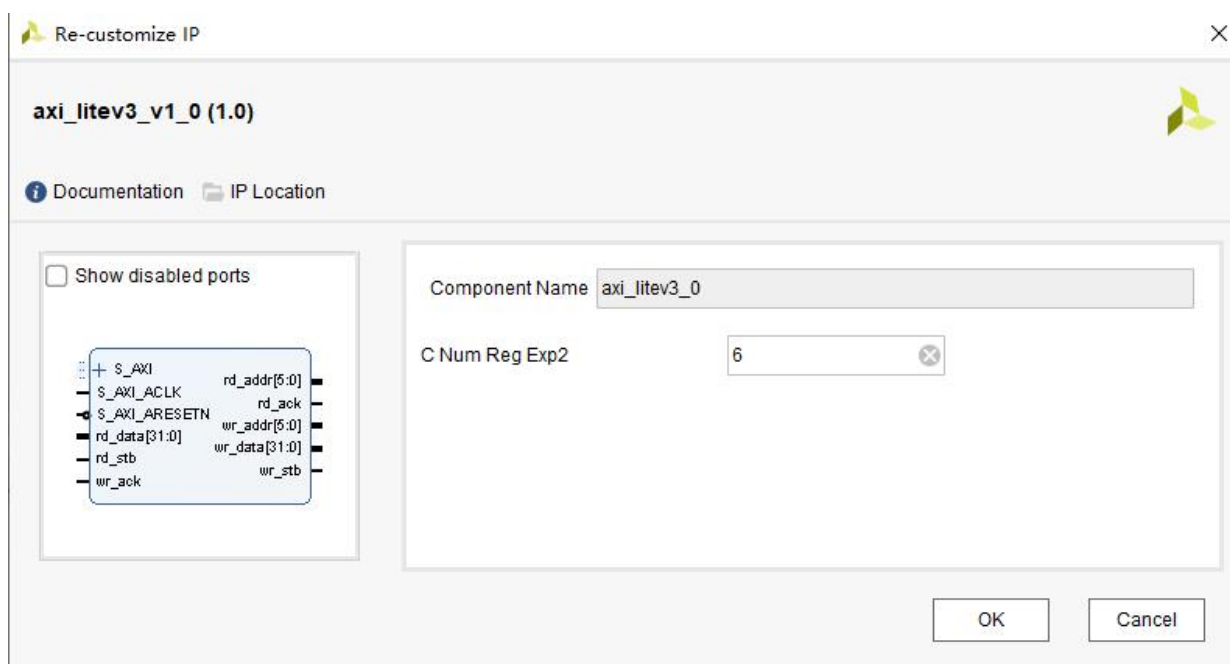


图 2 修改寄存器映射地址空间

3 裸机应用程序

本节内容包括：在裸机下验证 FPGA 硬件工程，GPIO 配置、AD9361 配置、寄存器交互。

在 Vivado 工程中选择 File-launch SDK 会自动打开工程目录下的软件 SDK。提供了如下几个工程：

adaloop_test 通过软件配置 DDS ipcore 的 phase 产生不同频率的单音信号，配置好 AD9361 后，启动单音信号发送，同时启动接收，并通过 chipscope 观察分析数据。顶层文件是 main.c。此文件架构来源于 ADI 官方提供的 NO-OS 软件程序，威视锐进行了优化和修改。

NO-OS 源程序可以参考 ADI 网站链接：

<https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms2-ebz/software/baremetal>

AD9361 输出化参数可以在 radio_set.h 文件中修改

```
uint32_t sample_rate = 61.44e6;
uint64_t tx_lo_freq = 2500e6;
uint64_t rx_lo_freq = 2500e6;
uint32_t bandwidth = 30.72e6;
int32_t gain = 10;
uint32_t txatt = 20e3;
```

3.1 初始化板卡

首先执行 platform_init(void) 进行 GPIO 设置方向并初始化。之后将以下 IO 设定指定的电平。

```
gpio_init(GPIO_DEVICE_ID);
gpio_direction(CDCM_SYNC      ,1);
gpio_direction(RF1_TXNRX      ,1);
gpio_direction(SYNC_IN        ,1);
gpio_direction(RF1_EN_AGC      ,1);
gpio_direction(REF_SELECT2     ,1);
gpio_direction(REF_SELECT      ,1);
gpio_direction(FDD_TDD_SEL     ,1);
gpio_direction(TRX_SW          ,1);
gpio_direction(RF1_ENABLE      ,1);
gpio_direction(RF1_RESETB      ,1);
mdelay(100);
gpio_set_value(RF1_RESETB      ,0);
gpio_set_value(RF1_ENABLE      ,1);
gpio_set_value(TRX_SW          ,1);
gpio_set_value(FDD_TDD_SEL     ,1);
gpio_set_value(REF_SELECT      ,1);
gpio_set_value(REF_SELECT2     ,0);
gpio_set_value(RF1_EN_AGC      ,0);
gpio_set_value(SYNC_IN        ,1);
gpio_set_value(RF1_TXNRX      ,1);
gpio_set_value(CDCM_SYNC      ,0);
```

第二步进行 SPI 初始化并初始化 CDCM6208 时钟芯片。

```
spi_init(SPI_DEVICE_ID1, 0, 0);
cdcm6208_init(&cdcm6208_spi);
cdcm6208_y5_internal(&cdcm6208_spi);
uint32_t lock_rf1=-1;
while (lock_rf1 != 0) {
    lock_rf1 = cdc6208_spi_read(&cdcm6208_spi, 21) & 0xF;
    printf("read from CDCM6208_RF1 reg21=0x%1x\n", lock_rf1);
}
printf("read from CDCM6208_RF1 reg21=0x%1x\n", lock_rf1);
mdelay(10);
gpio_set_value(CDCM_SYNC, 1);
mdelay(10);
gpio_set_value(CDCM_SYNC, 0);
mdelay(10);
gpio_set_value(CDCM_SYNC, 1);
mdelay(500);
```

3.2 运行参考例程

AD9361 提供测试一些测试模式和正常模式，用户第一次应用参考例程可以首先尝试 AD9361 内部回环模式，此时发送的基带信号直接在 AD9361 的 TX RX 数字接口回环，并不经过 AD/DA 和射频链路。下文分别介绍内部回环测试和射频回环测试。

3.3 LoopBack 回环测试

使用 LoopBack 回环需要在 ad9361_config.c 文件下，使能以下代码，之后选择 build project。

```
ad9361_spi_write(ad9361_phy->spi, 0x3F5, 0x41);
```

将线缆连接及 JTAG 线缆插好，选择 JTAG 模式。板卡上电，run 应用程序。

程序运行起来后可以在串口调试终端看到 AD9361 的配置过程。串口调试终端推荐使用 putty、mobaxterm 等工具。波特率 115200，8 数据位，1 开始，1 停止。
串口打印信息如下：


```

*****
*****DeepRED trxloop standalone begin*****
*****
cdcm6208_reg[40]=102
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x4
read from CDCM6208_RF1 reg21=0x0
read from CDCM6208_RF1 reg21=0x0
ad9361_init : AD936x Rev 2 successfully initialized
tx_fir_config:coef_size=96,int=2
rx_fir_config:coef_size=96,dec=2
sample_freq=40000000Hz
sample_freq=40000000Hz
ad9361_phy,REG_PRODUCT_ID=10
register addr=0x0 read out = 420
The FPGA firmware version=V1.0
The Software firmware version=V1.0
The lasted build at 20201111_PM1
The lasted build at 16:46:37 Sep 9 2021
Xadc Setup PASSED
CDCM6208 use internal 10MHz VCTCXO lock successful!
DeepRED initial successful!
fpga temperature=39.013519
rfl_ad9361 temp=95,

```

打开 ILA 在线逻辑分析仪，可以观察发送与接收到的 IQ 数据如下图，可以看到内部回环收发信号仅存在延时，内容是相同的。



图 3 内部回环测试

3.4 射频回环测试

将上一测试代码改回原来状态。可以用回环射频缆将 TX 和 RX 回环，可以看到通过射频回环的结果



图 4 射频回环测试

可以选中波形导出使用第三方软件进行分析

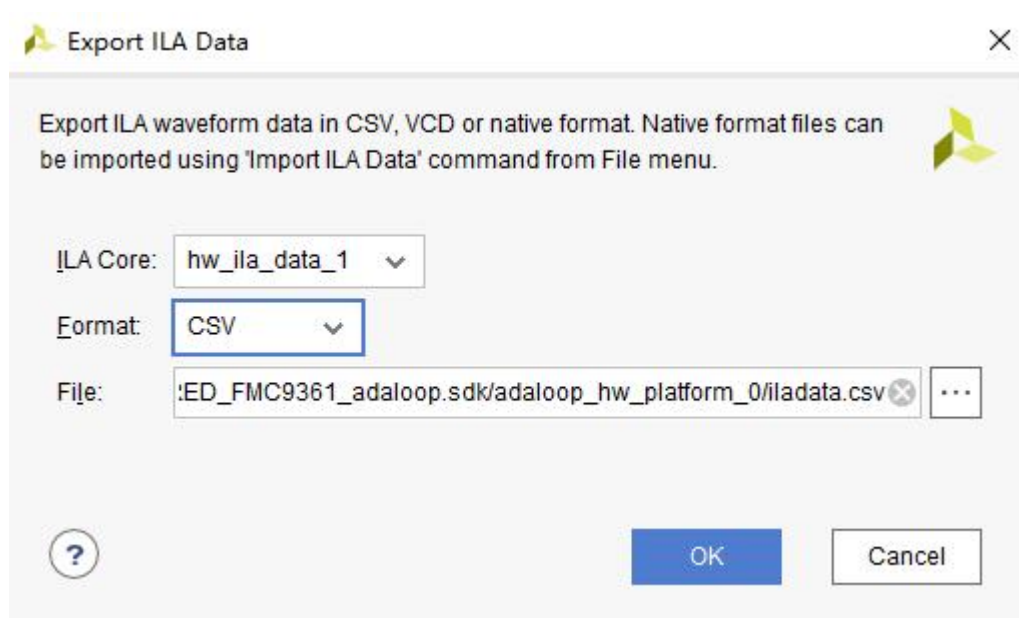


图 5 ILA 数据导出

比如通过 matlab 进行分析：

```

clc;clear;close all;
m = csvread('iladata.csv', 1, 0);
data(:,1)=m(1:4:end, 11)+1i*m(1:4:end, 10);
data(:,2)=m(1:4:end, 13)+1i*m(1:4:end, 12);
for i=1:size(data,2)
    subplot(size(data,2),3,3*(i-1)+1);plot(real(data(:,1)));hold
on;plot(imag(data(:,1)));
    subplot(size(data,2),3,3*(i-1)+2);plot(data(:,1));axis equal;

    subplot(size(data,2),3,3*(i-1)+3);pwelch(data(:,1),[],[],[],40e6,'centered','psd');
end

```

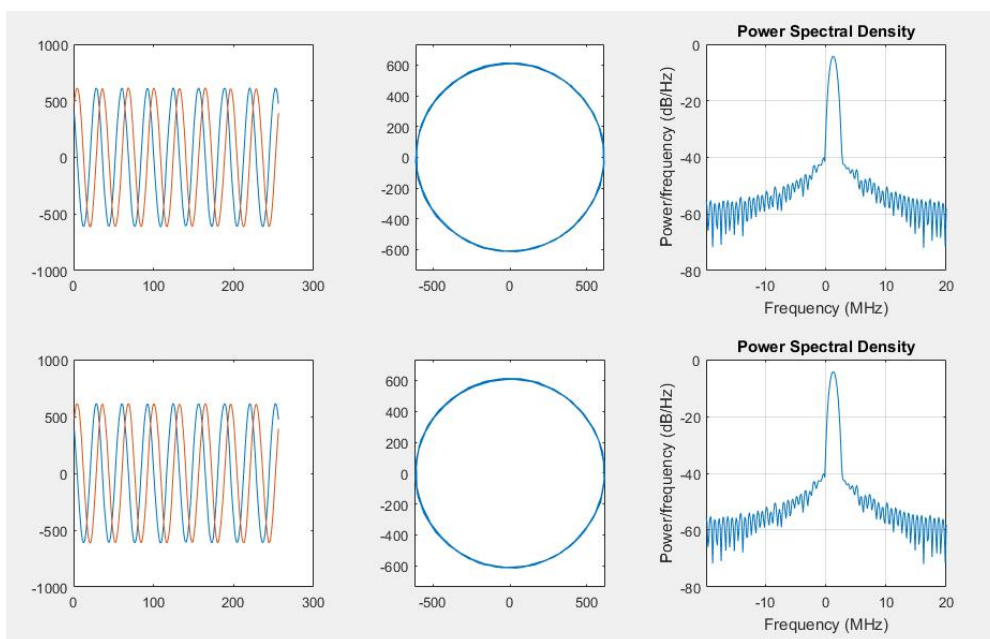


图 6 matlab 数据分析结果

附录 A

自定义 IP 核的添加

首先在工程目录创建名为 DeepRED_FMC9361_adaloop.ipdef 的文件夹，再将威视锐提供的自定义 AXI 接口 IP 核文件 xilinx.com_user_axi_litev3_1.0 复制，添加到这个文件夹中；

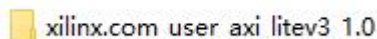


图 7 自定义 AXI 接口 IP 核文件夹

接着在 Block Design 窗口空白处右击，选择 IP Settings，进入 IP 设置界面，在左侧 Project Settings 的 IP 下拉栏里选择 Repository，然后在右侧点击加号；

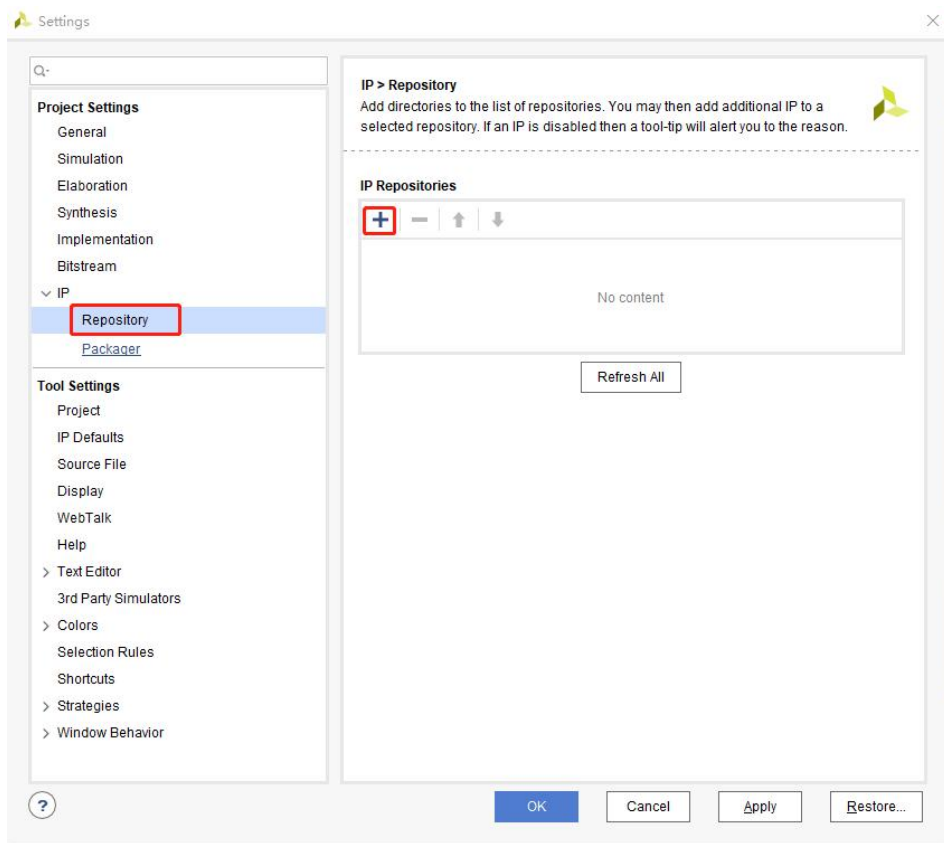


图 8 IP Settings 界面

在弹出的 IP Repository 设置窗口栏中，选择上一步保存的 IP 核文件；

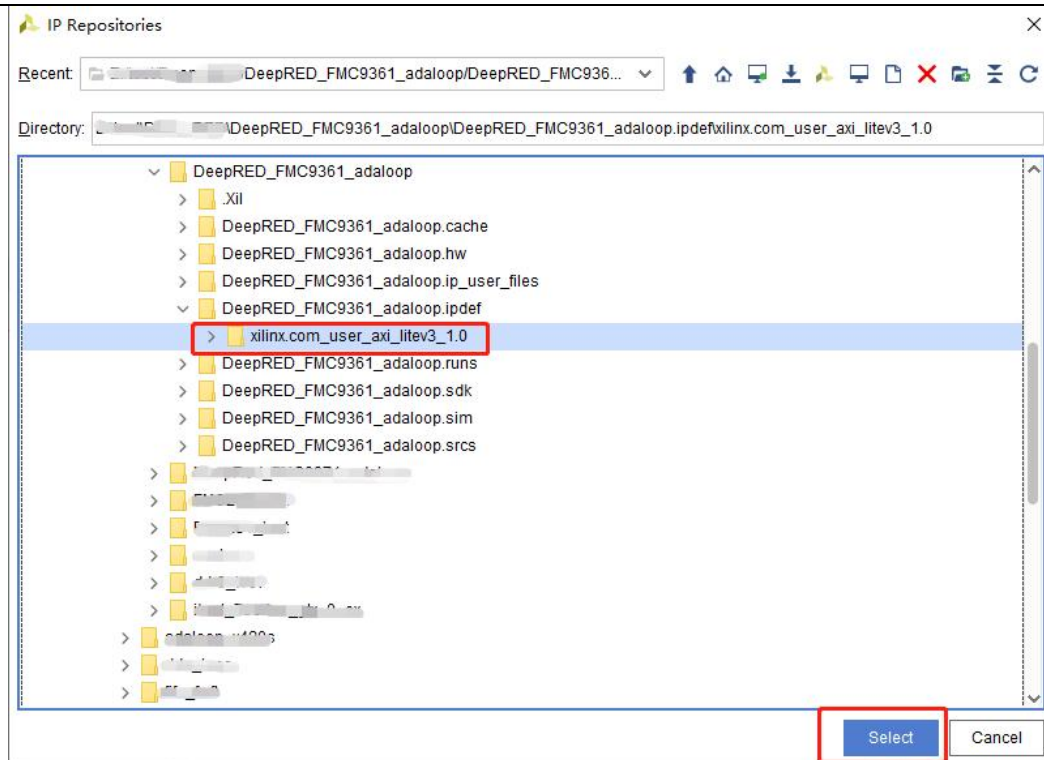


图 9 IP Repository 设置窗口栏

添加成功之后如图所示，这时已经将自定义的 IP 核添加到这个工程的 IP 库中，点击 OK 回到 Block Design 界面；



图 10 IP 库中的自定义 IP 核

接着在 Block Design 窗口上方点击加号，添加 IP 核，在弹出的窗口中搜索 IP 核的名称 axi_litev3_1_0，双击确认；

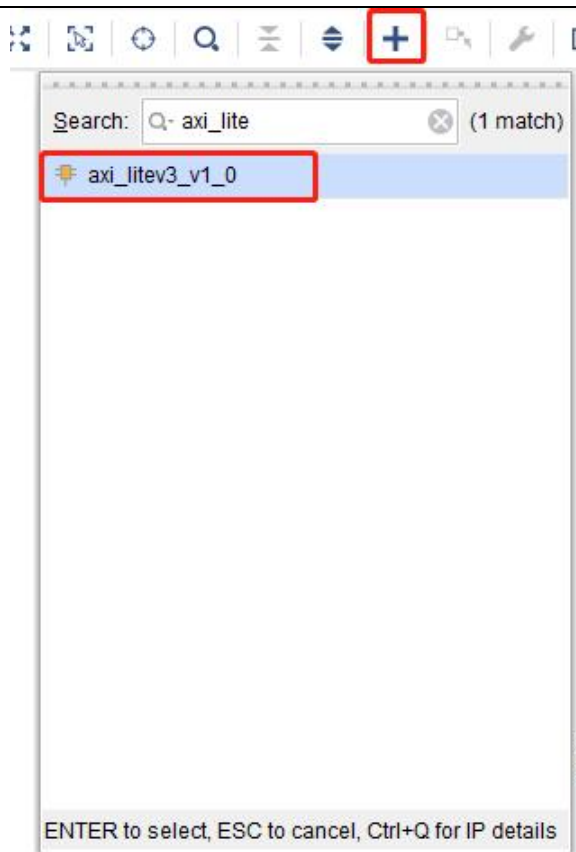


图 11 在 IP 库中搜索自定义的 IP 核

在 Block Design 中正确添加自定义 IP 核后如图所示：

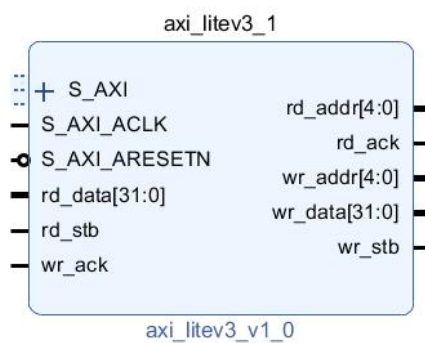


图 12 axi_litev3_1_0 IP 核

双击该 IP 核可进入设置页面，可以看到仅有一个参数，是配置读写地址位宽，本实验例程中配置位宽位 6 位；

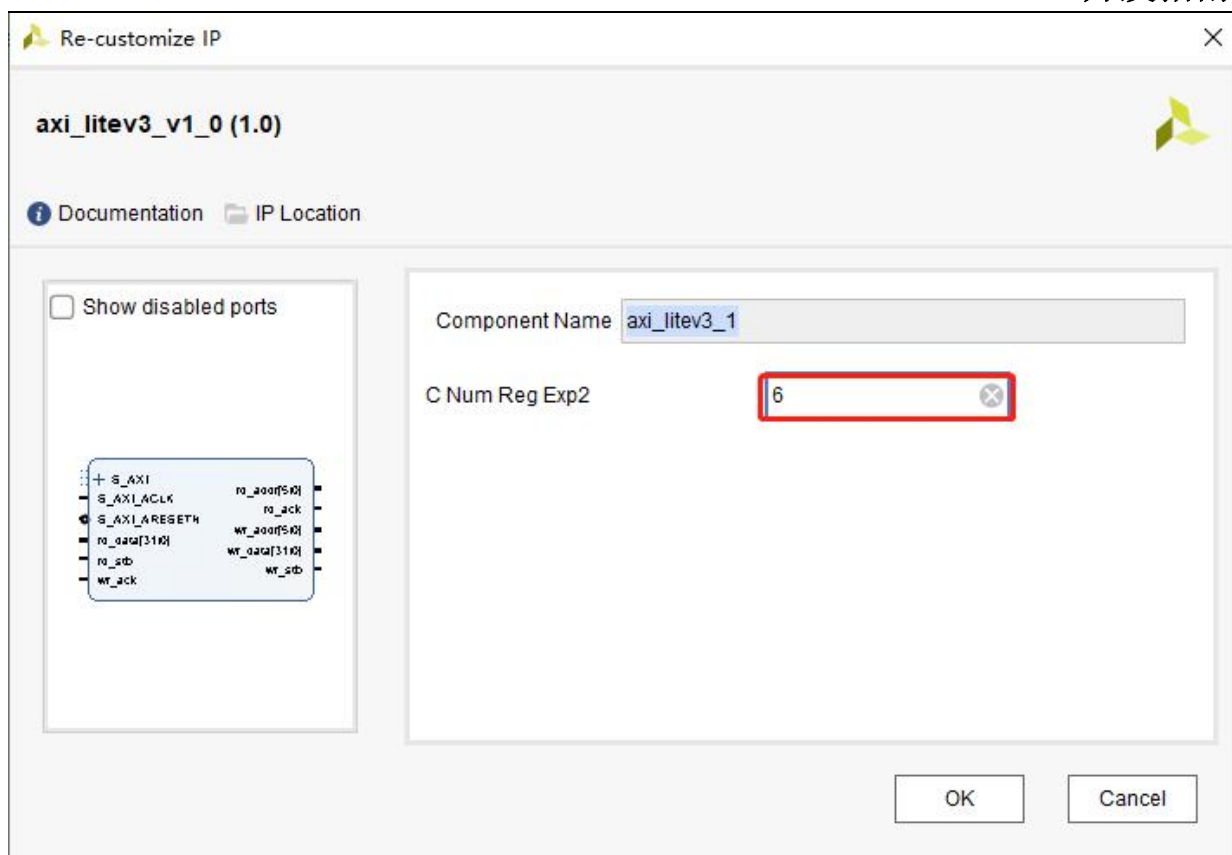


图 13 配置 axi_litev3_1_0 IP 核