

ProgrammerSought

 search

Xilinx Vitis 2020.1 Curing the MicroBlaze program with SREC SPI Bootloader, the solution that does not run after the power-on power-on

tags: [FPGA](#) [Vivado](#) [Vitis](#) [Microblaze](#) [Program curing](#)

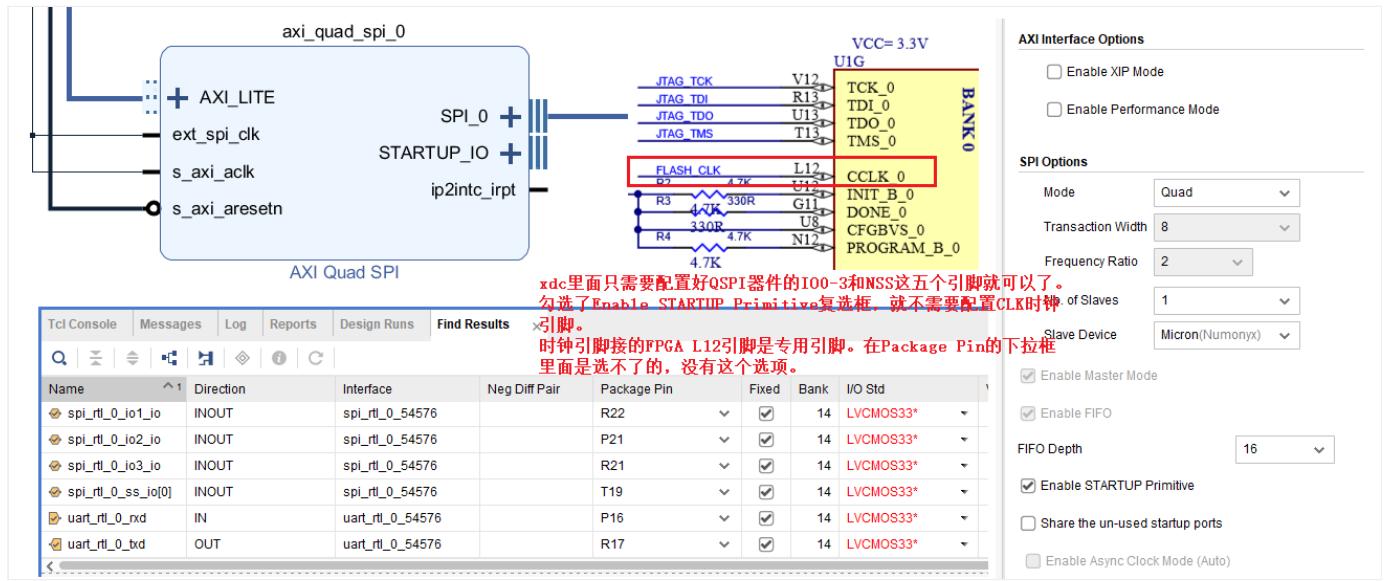
The model of FPGA is XC7A35TFFG484-2, and the development board is used by rice. The main program Hello_World runs outside DDR3 memory, and SREC SPI Bootloader runs in the FPGA tablet BRAM.

Curing the program with the method described herein:[Microblaze program curing process](#)

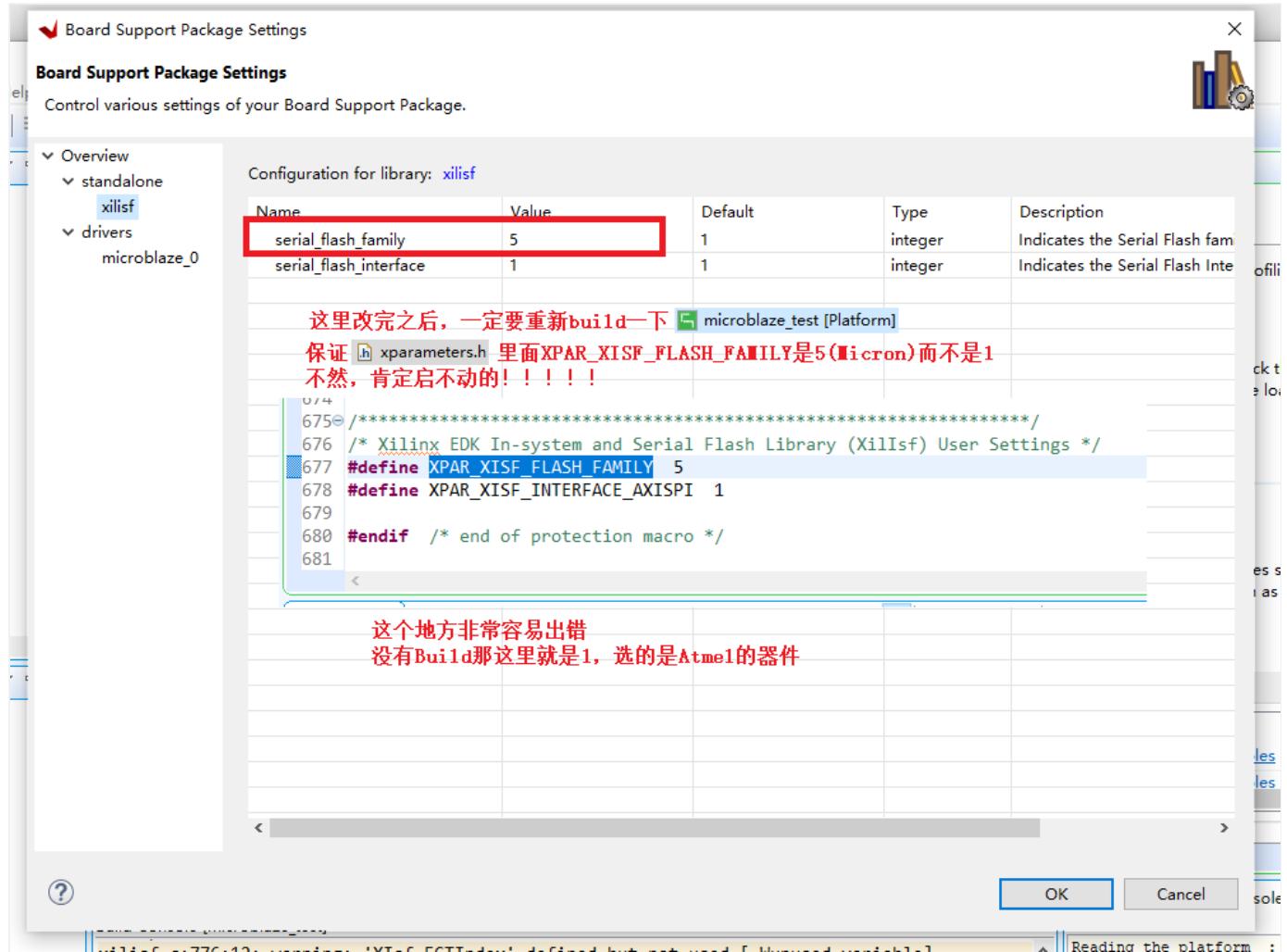
Download.BIT is generated, but it is always running off the download. Even bootloader itself can't run, and the serial port does not have any output.

First, it is suspected that the configuration problem of the AXI Quad SPI in Vivado Block Design. But with the routine of the development board, the configuration is exactly the same. In fact, there is no problem he TOP

After the Startup Primitive is checked, the IP core has lacks the QSPI_CLK pin, but does not affect communication. In the schematic, FLASH_CLK is connected to L12, which is a dedicated pin that configures FPGA. It cannot be used as a normal I / O port, and it cannot be selected in the drop-down list of package Pin. So, the clock pin is not needed, just connect 4 data pins and one NSS selection pin.



So, in the VITIS, DEBUG, the bootloader project, the serial port only outputs a SREC SPI Bootloader. Single-step debugging discovery, XISF_INITIALIZE error, returned 1, the program exits the main function directly. Continue analysis, discovery the program is ATMEL's branch, did not walk Micron's branch, open XParameters.h, XPAR_XISF_FLASH_FAMILY is 1, indicating that the set serial_flash_family does not take effect at all!



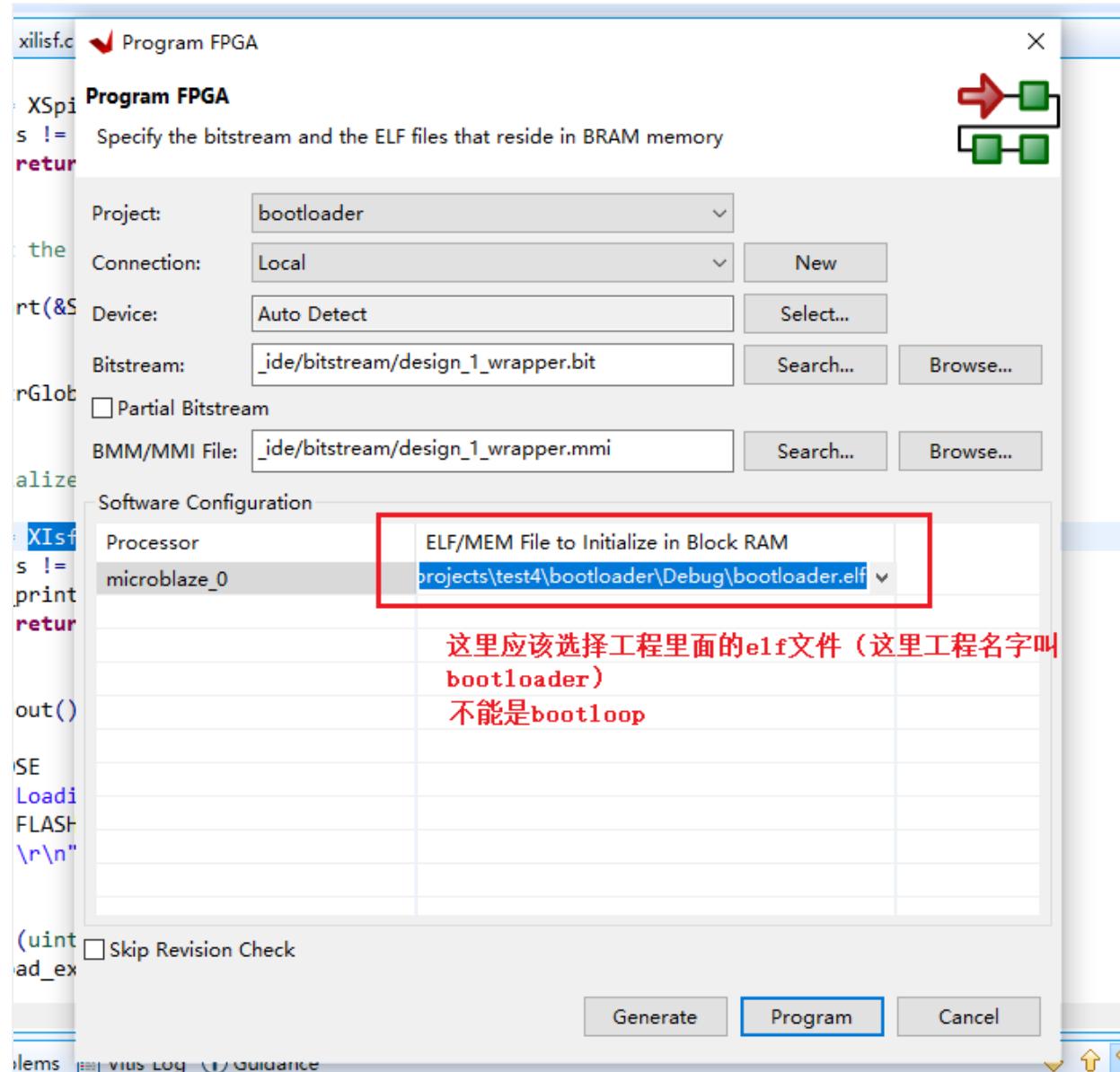
After modifying `serial_flash_family`, you must re-build Platform project, `XPAR_XISF_FLASH_FAMILY` inside `XParameters.h` will be updated.

The second place is that I have to open the Generate Linker Script. When the bootloader is configured to run in BRAM, whether the main program is running in DDR memory. If not, you have to change it manually.

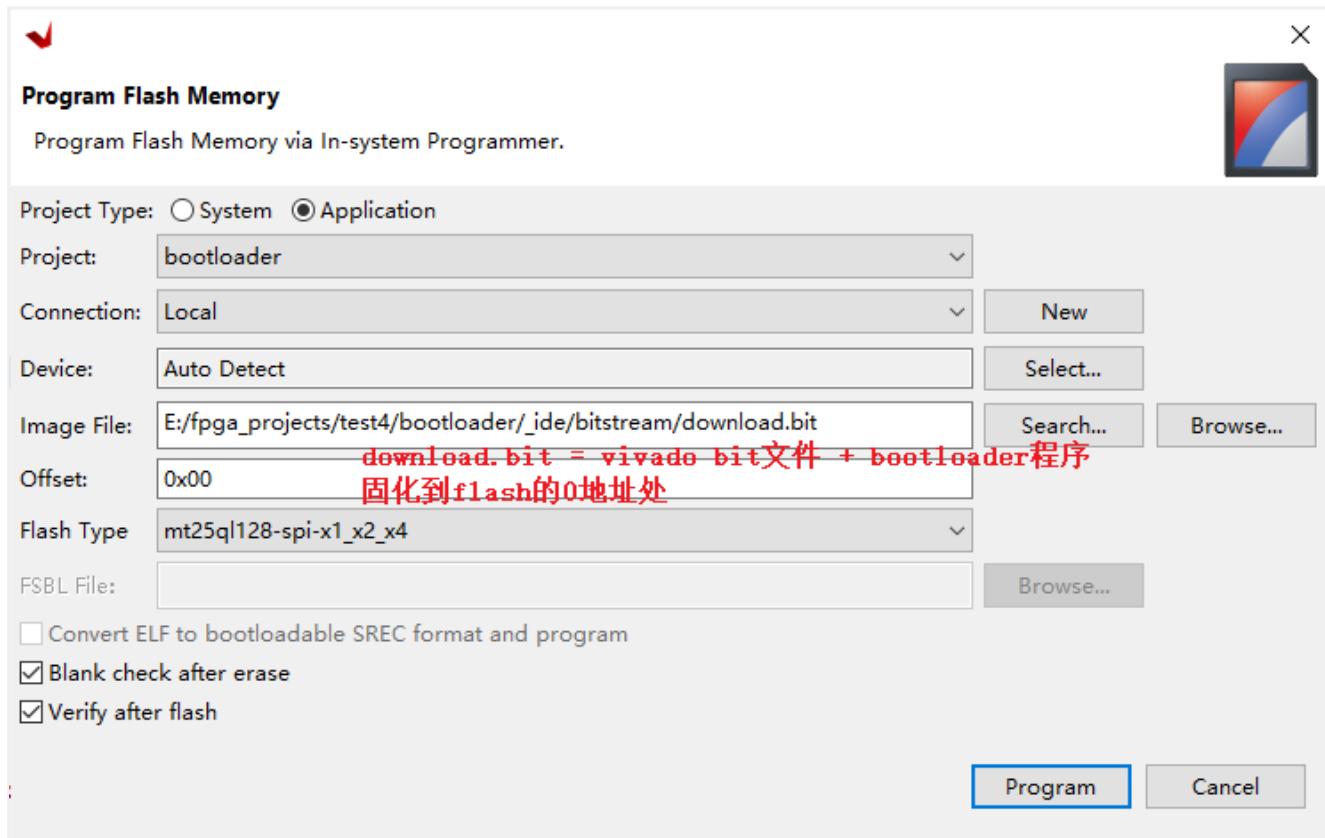
The screenshot shows the 'Generate linker script' dialog for a 'bootloader' project. The 'Output Script' field is set to 'E:\fpga_projects\test4\bootloader\src\script.ld'. The 'Basic' tab is selected, showing memory settings for the Microblaze local memory and mig_7series_0_memaddr. The 'Place Code Sections in:' dropdown is set to 'roblaze_0_local_memory_dlmb_bram_if_cntlr_Mem'. A red box highlights this dropdown. The 'Place Data Sections in:' and 'Place Heap and Stack in:' dropdowns are also highlighted with red boxes and show the same value. Below these, 'Heap Size:' and 'Stack Size:' are both set to '1 KB' at address '0x00000400'. A red annotation on the right says 'bootloader要选择在BRAPI里面运行' (Bootloader should be chosen in BRAM).

The screenshot shows the 'Generate linker script' dialog for a 'hello_world' project. The 'Output Script' field is set to 'E:\fpga_projects\test4\hello_world\src\script.ld'. The 'Basic' tab is selected, showing memory settings for the Microblaze local memory and mig_7series_0_memaddr. The 'Place Code Sections in:' dropdown is set to 'mig_7series_0_memaddr'. A red box highlights this dropdown. The 'Place Data Sections in:' and 'Place Heap and Stack in:' dropdowns are also highlighted with red boxes and show the same value. Below these, 'Heap Size:' and 'Stack Size:' are both set to '1 KB' at address '0x00000400'. A red annotation on the right says '主程序要选择在DDR内存里面运行' (Main program should be chosen in DDR memory).

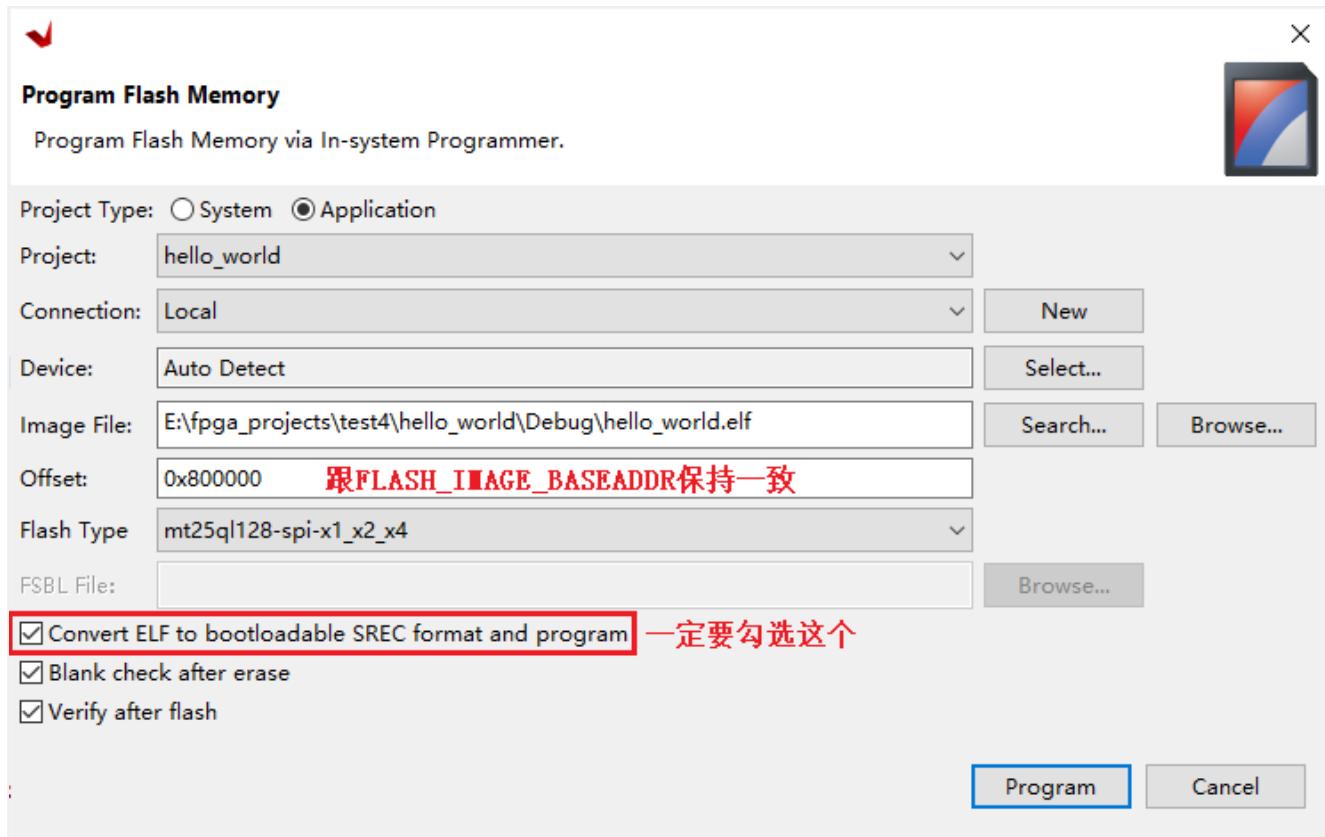
The third place has been found for a long time to find that the default BRAM content in the Program FPGA dialog box is bootloop, and the difference in the article is here! **You should choose bootloader.elf!** After selecting it, click the Program button to merge Vivado's BIT file, and the ELF file of the VITIS Bootloader project to generate download.bit.



Download Download.bit files to the 0 address of the SPI Flash:



Download the master program's ELF file to the non-zero address of the SPI Flash, pay attention to check the check box for Convert Elf to Srec:



After the development board, after the power is started, the serial port finally output, and the bootloader runs, but only one SREC SPI Bootloader is output.

Add XIL_PRINTF printing in the XIOTLOADER program, compile the bootloader program and run:

(Note After opening the file where the XISF_INITIALIZE function is located, the window will automatically TOP

select the Platform project, and the direct button is compiled. So first select the bootloader engineering on the left, then click Build Compilation)

```

1  /*
2   * Check for Intel/STM/Winbond/Spansion Serial Flash.
3   */
4 Status = IntelStmFlashInitialize(InstancePtr, ReadBuf);
5 xil_printf("ReadBuf: ");
6 dump_data(ReadBuf, sizeof(ReadBuf));
7 xil_printf("IntelStmFlashInitialize=%d\r\n", Status);

```

It was found that although XISF_Initialize did correctly entered Micron's branch, the IntelStmflashInitialize function failed, returned Status = 1, and then the main function returned to XST_Failure exited. Print the readbuf's content with your own dump_data function, will find that the content is not right, and there is no data to be read at all.

At this time, press the reset button on the development board (the reset pin of the normal I / O port, connect to the reset signal of all modules in Verilog), and successfully launched the program.

Power out again, execute the IntelStmflashInitialize function after power-on start, return value or status = 1

Press the custom reset button on the development board and load success and start the program.

.....

If the reset key is pressed when the program starts successfully, the program will stop running, and the bootloader is not running, and the serial port does not have any output.

Power off, again power, bootload can run.

In estimated: When energizing, the initial value of BRAM is the program content of the bootloader, and the bootloader program is running, just QSPI initialization failed. At this time, press the reset button, reinitialize the QSPI is successful, and can start the program.

After the program is started, the content of the BRAM is overwritten, and the bootloader program is gone. Since the reset key is a normal I / O port, press the post after the module is reset, but the content of BRAM cannot be restored, and Bootloader is of course unable to run.

At the time of power-on, the IntelStmflashInitialize function failed, which may be due to insufficient power-on time, Flash device is not ready.

We can change the code of the main function in Bootloader, remove the code of the exiting the main function, plus the Do-While loop, and initialize the delay after the delay is re-initialized:

(The header file where the delay uses the header "is" Sleep.H>)

```

1  /*
2   * Initialize the Serial Flash Library.
3   */
4 do
5 {
6     Status = XIIsf_Initialize(&Isf, &Spi, ISF_SPI_SELECT, IsfwriteBuffer);
7     if (Status != XST_SUCCESS)
8     {
9         xil_printf("XIIsf_Initialize failed! Status=%d\r\n", Status);
10        Usleep (100000); // Time 100MS

```

TOP

```

11 |     }
12 | } while (Status != XST_SUCCESS);
13 | xil_printf("XIsf_Initialize OK!\r\n");

```

```

165 Status = XSpi_Initialize(&Spi, SPI_DEVICE_ID);
166 if(Status != XST_SUCCESS) {
167     return XST_FAILURE;
168 }
169 /*
170 * Start the SPI driver so that interrupts and the device are enabled.
171 */
172 XSpi_Start(&Spi);
173
174 XSpi_IntrGlobalDisable(&Spi);
175
176 /*
177 * Initialize the Serial Flash Library.
178 */
179 do
180 {
181     Status = XIsf_Initialize(&Isf, &Spi, ISF_SPI_SELECT, IsfWriteBuffer);
182     if (Status != XST_SUCCESS)
183     {
184         xil_printf("XIsf_Initialize failed! Status=%d\r\n", Status);
185         usleep(100000); // 延时100ms
186     }
187 } while (Status != XST_SUCCESS);
188 xil_printf("XIsf_Initialize OK!\r\n");
189
190 ...

```

After the change, recompile the program and then solidify it to the development board. The development board is broken, and then open, you can start successfully, no need to press the reset button again.

The problem finally solved.

```

COM21 - Tera Term VT
File Edit Setup Control Window Help

SREC SPI Bootloader
ReadBuf: 000000000000
IntelStmFlashInitialize=1
XIsf_Initialize failed! Status=1
ReadBuf: FF20BA181000
IntelStmFlashInitialize=0
XIsf_Initialize OK!
Loading SREC image from flash @ address: 00800000
Bootloader: Processed (0x)000012d1 S-records
Executing program starting at address: 00000000
Hello World
Successfully ran Hello World application
&i=0x80013498, &f=0x80013494, data=0x8001349c
sizeof(int)=4, sizeof(float)=4
i=0, f=0.0
i=1, f=3.1
i=2, f=6.3
i=3, f=9.4
i=4, f=12.6
i=5, f=15.7
i=6, f=18.8
i=7, f=22.0
i=8, f=25.1

```

上电后第一次初始化SPI Flash失败
ReadBuf里面没有读出来器件ID
但第二次初始化就成功了，器件ID:0x20也读出来了

In fact, MicroBlaze does not require an external DDR3 memory can also run independently, running memory with BRAM, at which time MicroBlaze does not enable Instruction and Data Cache. No external reset pin is required, do MicroBlaze's reset signal with the Locked signal output by Clock Wizard, and the reset signal is valid when lockd = 0 is valid, which is a low level reset. The Clock Wizard itself can remove the RESET signal in the IP core configuration window.

In this way, the ELF files compiled by VITIS can be integrated directly with Vivado's bit files in the Program FPGA dialog, generate download.bit, then burn the download.bit file to the 0 address of the SPI Flash in Program Flash Memory, You can run the power on the power. No SREC SPI bootloader is required.

In this case, the program code and program variables are saved inside the BRAM.

[Copyright Complaint](#) [Spam Report](#)

Powered By **VDO.AI**

Zabbix disk performance monitoring

Intelligent Recommendation



The uninterrupted power-on program can still run after CCS stops debugging

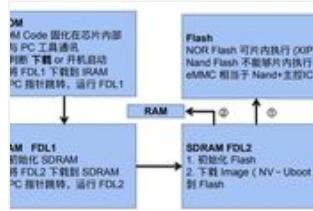
Recently I developed a project and used CCS to compile and debug. I found a problem with debugging, which would lead us to misjudge, such as the problem I encountered: description: I was debugging the...



The first entry helloworld program Xilinx Vitis IDE

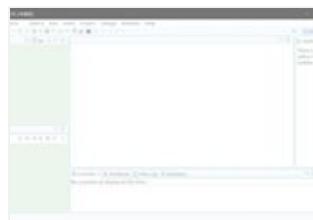
Operating System: Ubuntu 18.04.4 LTS Lsb_release -a command
Installation Vitis

<https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/vitis.html>
Online installation sp...



[Boot] Power on the hardware to the Bootloader

Overview This series briefly introduces the Android boot process to understand the overall Android boot process. Further provide theoretical support for boot optimization, system tailoring, related ...



Xilinx vitis unsweefence solution when starting

When starting the Xilinx Vitis 2021.1, whether it is started from the Launch Vitis from the Xilinx Vivado interface or directly, it will show that the main interface will not be responded after the ma...



Xilinx Vitis

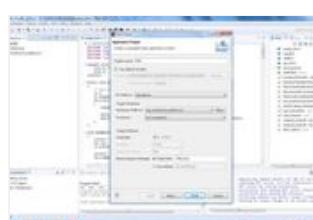
Host Application To set he environment properly, the host application needs to initialize the standard OpenCL structures: target platform, devices, context, command queue, and program. Once the OpenCL...

Save Money At The Airport With These Thrifty Budget Ideas

Lasetrips.com

tm

More Recommendation



Xilinx Zynq program Curing Summary Record

Xilinx Zynq Series Program Curing Process Summary Record The following will summarize the Xilinx Zynq Series Program Curing Process. Step 1: Create FSBL Create an Application Project in the FILE bar t...

Win 10 after the power off, the power light does not stop flashing solution

After the Win 10 system was shut down, it was found that the power suppl...

TOP



```

    .org reset
        nmi_clock, freq period, reset 200ns / 1 us = 40,000 clock
    unprepared
    {25 : 0}
    sys_rst_cnt >= RST_TH;
    sys_rst_cnt <= RST_TH;
    if sys_rst_cnt >= RST_TH then
        sys_rst_out <= sys_rst_out + 1'0;
    end
    sys_rst_out <= sys_rst_out - 1'0;
    when flag update

```

Xilinx power-on initial value

When xillix is powered on, if the variable is not reset, it is not necessarily the initial value of 0; After viewing the comprehensive schematic, you can find that the initial value of sys_rst_cnt

i...



I] Xilinx FPGA's power budget

Recently opened a new project or four days did not update a bit of a mess, and today the first stroke it about power budget. Xilinx common power budget in two ways: 1, pre-design for the power budget:...

Description	Min
Internal supply voltage	+0.0
Auxiliary supply voltage	+0.0
Supply voltage for the block RAM memories	+0.0
Output driver supply voltage for I/O banks	+0.0
Input reference voltage	+0.0
I/O input voltage	+0.0
I/O input voltage (when $V_{DDQ} = 3.3V$) for V _{DQ} and differential I/O standards	+0.4
Reference clock absolute input voltage	+0.0
Key memory battery backup supply	+0.0
Analog supply voltage for the GTP transmitter and receiver circuits	+0.0
Analog supply voltage for the GTP transmitter and receiver termination circuits	+0.0
Reference clock absolute input voltage	+0.0
Receiver (RxFP) and Transmitter (TxFP) absolute input voltage	+0.0

Xilinx artix7 power rail

For artix7 series, we usually need to use the following power supply rails: Reference Document: DS181 (v1.22) April 13, 2017 Artix-7 FPGAs Data Sheet: DC and AC Switching Characteristics 1, each of th...



Related Posts

- Xilinx Vitis 2020.1 cannot create SREC SPI Bootloader project, prompting the solution of lack of Xilisf library

TOP

- [Tutorial] Xilinx Vivado / Vitis 2020.1 Create the simplest MicroBlaze project to run the Hello World C language program (do not use external DDR3 memory), and solidify to SPI Flash
- [Tutorial] Xilinx Vivado / Vitis 2020.1 Create a MicroBlaze project to run the Hello World C language program (using external DDR3 memory)
- Questions about the Sleep function card of the MicroBlaze soft core in Xilinx VITIS 2020.1
- Microblaze program curing process
- Use the VITIS curing program
- I can't find MicroBlaze_o when running a C program in Xilinx Vitis 2020.1
- The power-on program caused by NVIC_Init in STM32 does not run
- Xilinx Vitis 2020.1 method to modify the size of BRAM memory occupied by the project
- Microblaze Bootloader

Advertisement



Popular Posts

- Verilog design work
- Ios UIView common animation effects
- Scala study notes (2)-basic grammar
- Java Mac computers need to pay attention to learning Java
- H5 uses input type = file to implement a direct call camera
- Numpy - Basic knowledge
- The difference between fair lock and unfair lock in AQS
- IIC control of VC707 development board
- Supplementary conversion, operation
- centOS 7 Firewalls are frequently used settings

TOP



Recommended Posts

- Enter two sequences of integers. The first sequence indicates the push order of the stack.
Please determine if the second sequence is the pop-up order of the stack. Assume that all numbers pushed onto the stack are not equal.
- Algorithm and Data Structure Analysis (1)
- Java internal class and exception class
- Overview of JavaBean
- [Kuangbin] four thematic shortest exercises Silver Cow Party POJ - 3268 [dijkstra]
- Federal study notes (2)
- LeetCode - 50.Pow(x,n)
- [MyBatis] Architecture layering and main objects
- Page directory entry and page table entry
- Netease mailbox front-end Javascript coding specification: basic specification



**GET YOUR
TeeJet[®]
TECHNOLOGIES**
**NOZZLES AT
DULTMEIER SALES**

SHOP NOW

tm

Related Tags

FPGA

Xilinx

Vitis

Vivado

SREC

MicroBlaze

Solidify

DDR3

MIG

sleep