Flashing a MicroBlaze Program

By bkallaher (/member/bkallaher/) in Circuits (/circuits/) > Electronics (/circuits/electronics/projects/)

Now that I have finished developing a MicroBlaze project I want to be able to have the program start on boot. I looked around and could not find any comprehensive instructions for making a MicroBlaze program load on boot. In this I will be using the Arty (http://store.digilentinc.com/arty-board-artix-7-fpga-development-board-for-makers-and-hobbyists/) as a reference as that is the board I am using but I will note where your process may deviate with a different board.

Materials Needed:

- Vivado
- Xilinx FPGA board (Digilent's Arty (http://store.digilentinc.com/arty-board-artix-7-fpga-development-board-for-makers-and-hobbyists/) in my case)

Step 1: Preparing for Programming

I'm assuming that you already have a block design ready to be synthesized.

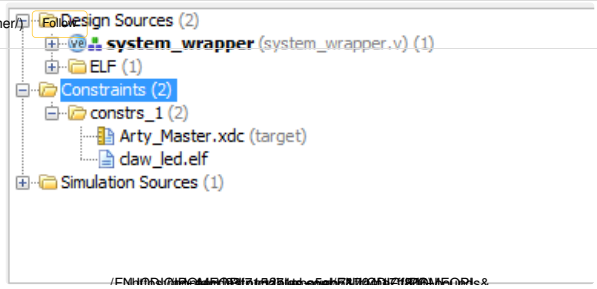
Navigate to Project Settings->Bitstream. Check the box next to -bin-file so Vivado knows to generate a bin file as well as a bit file. Run synthesis and open the synthesized design. go to Tools->Edit Device Properties. Under General Set the Enable Bitstream Compression to "TRUE". Under Configuration set the Configuration Rate (Mhz) to "33". Finally under Configuration Modes select Master SPI x4. The settings for the Configuration Rate and Mode may differ for your board so check the documentation for your board.

Click OK and save the design. It may ask to save a new constraints file or to an existing one. I personally write to my existing file as it will only add a few lines for programming. Now generate the bitstream. Do not program the board yet.

Step 2: Generating the Programming File From the SDK

Once the bitstream has finished generating export the hardware including the bitstream. Launch the SDK and create your C project as normal. Build the project to generate an .ELF file. This file will be used in the following steps to program the board.

Step 3: Adding the ELF to the Vivado Project

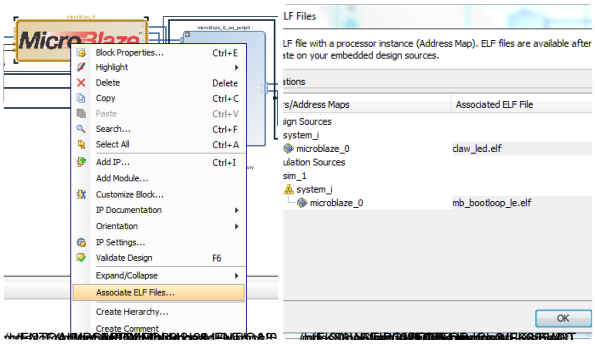


Return to Vivado and add a new constraint to the project. Locate the .ELF file generated by the SDK. The file will be located in PROJECT_ROOT/PROJECT_NAME.sdk/SDK_PROJECT_NAME/Debug/ where:

- PROJECT_ROOT is the folder where your Vivado project file is stored
- PROJECT_NAME is the name of your vivado project
- SDK_PROJECT_NAME is the name that you gave your SDK project

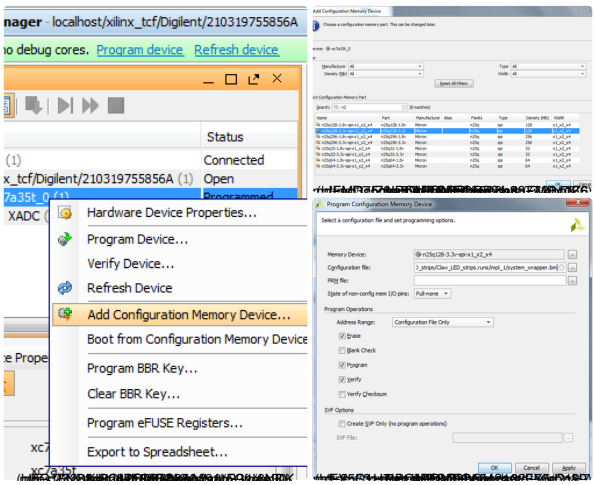
Click OK until you return to the main Vivado window.

Step 4: Associate the ELF With the MicroBlaze



Open the block design and right click on the MicroBlaze block. Select the "Associate ELF files" option. Find the Design Sources->system_i->microblaze_0 and click the "..." button on the right of the window. Navigate to the .ELF file as in the last step, select it and click OK until you are back to the main window of Vivado.

Step 5: Program the Board



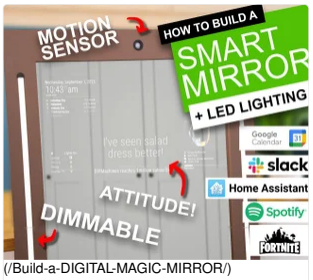
Regenerate the bitstream, this will build the c program into the binary. This step may differ from board to board due to memory differences.

Make sure the board is in Quad SPI programming mode. For the Arty this means that JP1 is populated with a jumper. Open the Hardware

Manager and open the device. Right click on the device and click Add Configuration Memory Device. The next window will ask for the memory chip that is on your board in the case of the Arty the chip is a Micron n25q128-3.3v part. Find and select your part and click OK. A dialog will ask if you want to program the device now. Click OK to do so. Select the bin file in the PROJECT_NAME.runs/impl_1 directory as the Configuration file and click OK. Vivado will now erase and reprogram the memory on the board. To run the project, power cycle the board.

If you are interested in the Arty (http://store.digilentinc.com/arty-board-artix-7-fpga-development-board-for-makers-and-hobbyists/), more information can be found here (http://reference.digilentinc.com/arty).

Recommendations



(/Build-a-DIGITAL-MAGIC-MIRROR/)

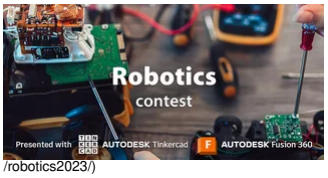
Build a DIGITAL MAGIC MIRROR (/Build-a-DIGITAL-MAGIC-MIRROR/) by DIY Machines (/member/DIY+Machines/) in 3D

32 1.8K

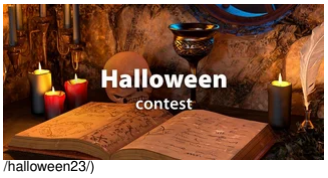
(/Sharp-C1-Famicom-Inspired-NES-TV/)

Sharp C1 Famicom Inspired NES T.V. (/Sharp-C1-Famicom-Inspired-NES-TV/) by insomniacfactory (/member/insomniacfactory/)

10 2.1K



(/contest/robotics2023/)



(/contest/halloween23/)

2 Comments

(/member/adhhrww/) adhhrww (/member/adhhrww/) 2 years ago Reply Upvote

Hi,
Thanks for the clear instructions.
I want to execute my program directly from the internal BRAM, I'm using Arty-7 board too.
My questions are:
Do I need to write a bootloader?
Can I just associate my application elf file?
Thanks.

(/member/kb5pgy/) kb5pgy (/member/kb5pgy/) 3 years ago Reply Upvote

Thanks for posting. I was having a beasty of a time programming the Arty S7 as a stand-alone. I was interested in the board for expendable data loggers since I deal with destructive testing at work, and I didn't want to trash $5K+ single-board computers.

One constructive criticism -- Which version of Vivado were you using for the example?

Post Comment