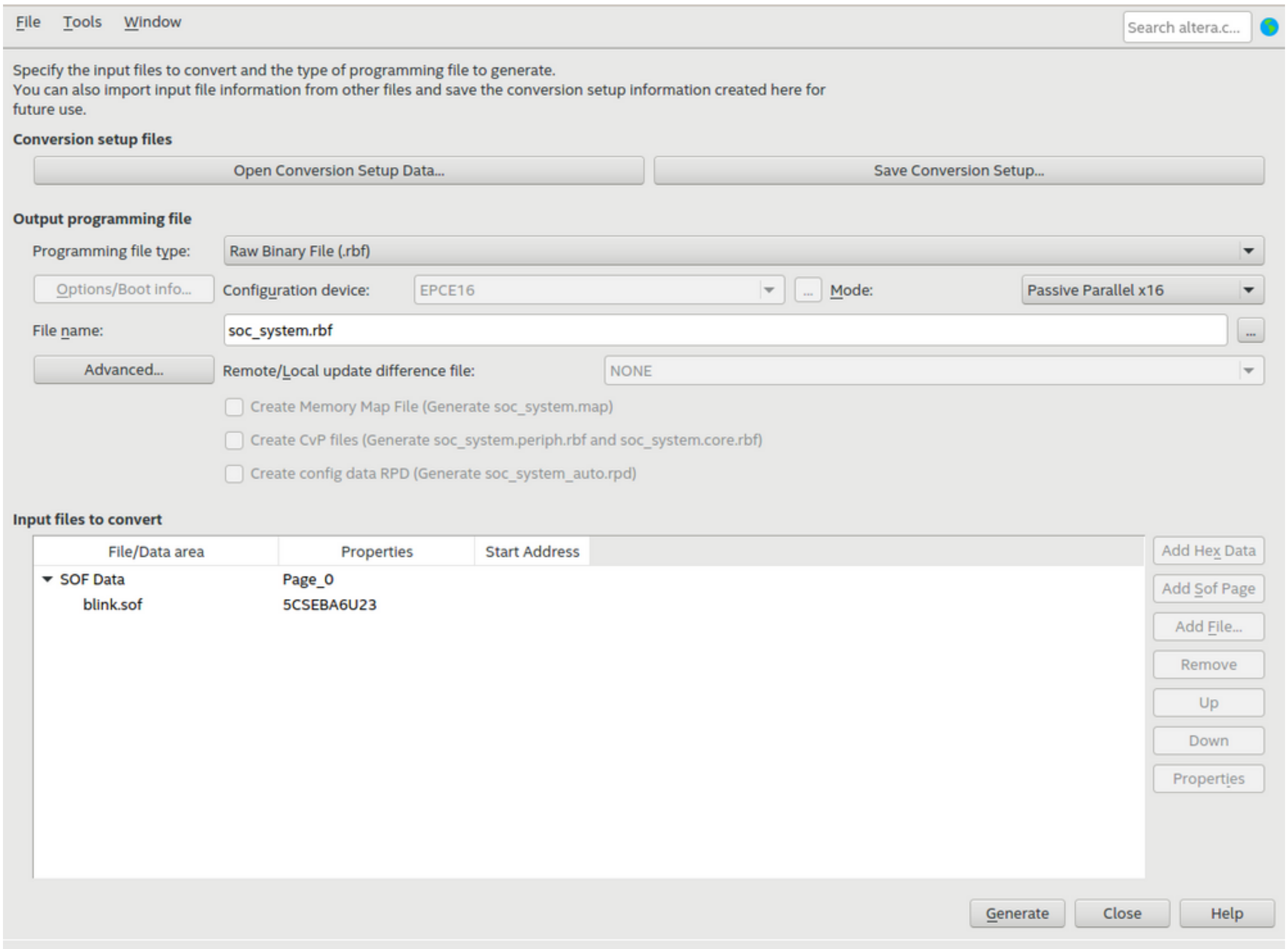
## HOW TO PROGRAM THE FPGA FABRIG FROM LINUX

Create a simple FPGA only design in Quartus Prime. I used Quartus Prime Lite Edition 20.1. E.g. “blink”.

1. Assign the pins using the pin planner or importing a proper assignment file (as the DE10_nano.qsf).
2. Run the Assembler in Quartus. It will generate the *sof* in the output_files folder.
3. Convert the *sof* to *rbf* using File > Convert Programming File. This will open a new window.
4. Under 'Programming File Type', select 'Raw Binary (.rbf)'
5. Under 'Mode', select 'Passive Parallel x16'.
6. Name the output file `**soc_system.rbf**`.
7. In the Input files section, click on SOF Data and click on 'Add File'. Select the *sof* file. In my case it was at `output_files/blink.sof`.
8. Click Generate and close the window.

Here is a screenshot:



Specify the input files to convert and the type of programming file to generate.
You can also import input file information from other files and save the conversion setup information created here for future use.

**Conversion setup files**

**Output programming file**

Programming file type: Raw Binary File (.rbf)

Configuration device: EPCE16

Mode: Passive Parallel x16

File name: soc_system.rbf

Remote/Local update difference file: NONE

☐ Create Memory Map File (Generate soc_system.map)

☐ Create CvP files (Generate soc_system.periph.rbf and soc_system.core.rbf)

☐ Create config data RPD (Generate soc_system_auto.rpd)

**Input files to convert**

| File/Data area | Properties | Start Address |
|---|---|---|
| SOF Data | Page_0 | |
| blink.sof | 5CSEBA6U23 | |

Set the MSEL bits on the DE10-Nano all to **ON**. This is needed for the "Passive Parallel x16" mode that we used: now when you insert the sd card and power it on, the design gets flashed and the FPGA fabric now implements your design.

To change the design to boot from Linux, you can scp the new design to the de10-nano, mount the fat partition (which should be /mnt/mmcblk0p1) and replace the **soc_system.rbf** file there. After that reboot it and it should flash the new design.

```
ssh root@<ipaddress>

mkdir -p fat
mount /dev/mmcblk0p1 fat

# Our bootloader expects the file to be named soc_system.rbf.
# Rename blink.rbf with whatever your file is called.
cp blink.rbf fat/soc_system.rbf

# Unmount the fat partition.
umount fat

reboot
```

Refernce

https://github.com/zangman/de10-nano/blob/master/docs/Flash-FPGA-On-Boot-Up.md

## Adding Programmer to the VM:

Fist add device “Altera”

USB bus is initially read only. Changed it: chmod 666 /dev/bus/usb/001/002

The bus and device numbers in path above found with lsusb inside VM.

Then add device “Altera DE SoC”

USB bus is initially read only. Changed it: chmod 666 /dev/bus/usb/001/002

The bus and device numbers in path above found with lsusb inside VM.