PETALINUX GUIDE FOR NEXYS 4 DDR / NEXYS VIDEO (PURE FPGA)
***THIS IS FOR VIVADO 2023.2 AND PETALINUX 2023.2

1. Download and install the PetaLinux tool:
   https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/embedded-design-tools.html . PetaLinux, in fact, can only be used on Linux. You also need to have Vivado on Linux in order to have access to full functionality of PetaLinux.

2. To create a project, use "petalinux-create -t project --template microblaze --name myproject"

3. Grab the XSA file from Vivado. This is done via *File -> Export -> Export Hardware*. Include bitstream in the XSA.

4. Now go in the project directory just created. Run the command "petalinux-config --get-hw-description [xsa_file.xsa]". This will open a config file. You can check the various configs that have been auto set using the XSA in order to understand more about the design.

5. Run "petalinux-build". This will populate the necessary files in the project directory for the first time.

6. There will be a file in the path
   "[project_dir]/project-spec/meta-user/recipes-bsp/device-tree/files/system-user.dtsi".
   This is the user defined device tree additions you are allowed to modify.
   The generated device tree files are in directory
   "[project_dir]/components/plnx_workspace/device-tree/device-tree/include/dt-bindings/".
   Read the .dtsi and .dts files to understand how device trees work. Not all of your blocks might have been fully defined, and you will notice that those functionalities are missing when you boot PetaLinux.

7. For use of an SD card through SPI (having connected it to an AXI QSPI block in Vivado), you would have to define the block in system-user.dtsi:

```
&axi_quad_spi_0 {
  #address-cells = <1>;
  #size-cells = <0>;
  mmc-slot@0 {
    compatible = "mmc-spi-slot";
    reg = <0>;
    voltage-ranges = <3300 3300>;
    spi-max-frequency = <50000000>;
  };
};
```

8. In order to enable kernel modules that weren't automatically detected from your XSA file, you need to run "petalinux-config -c kernel".
In this config UI, you can enable device drivers and other compatibility modules. For example, in order to add SD card over SPI support, go to *Device Drivers -> MMC/SD/SDIO support* and enable *MMC/SD over SPI.*

9. In order to add libraries to your root filesystem, run "petalinux-config -c rootfs". There will be a similar config to earlier ones where you can enable libraries.
    ● For custom libraries, refer to **PetaLinux documentation (UG1144)** on AMD/Xilinx website.

10.   Run "petalinux-build" in order to save your changes to any part of the design. The build will take shorter as it only builds changes.

11. In order to boot petalinux over JTAG, run "petalinux-boot --jtag --fpga --kernel". This might need root access, in which case run this on "sudo bash".

12. If you want to boot petalinux from QSPI flash, you can create an MCS file by running "petalinux-package --boot --force --fsbl ./images/linux/fs-boot.elf --fpga ./images/linux/system.bit --u-boot --kernel --flash-size 32 --flash-intf SPIx4".
The referenced files are generated after running "petalinux-build". The resulting file boot.mcs can be used to configure the onboard flash on Vivado. Make sure you read the generated boot.scr file on how the boot script works, in case the addresses are misaligned. These can be defined in "petalinux-config".

13. If you want to use your SD card as the root filesystem, in "petalinux-config", go to *Image Packaging Configuration -> INITRAMFS/INITRD Image name* , and change it to "petalinux-initramfs-image".

    Then, go to *DTG Settings -> Kernel Bootargs -> Auto generated bootargs* , and change it "console=ttyUL0,115200 earlycon root=/dev/mmcblk0p1 rw rootwait rootfstype=ext4".
    This will cause the kernel to look for an SD card to mount on when petalinux is launched. You can unzip the rootfs already generated in "[project_dir]/images/linux/rootfs.tar.gz" into your SD card to use as root filesystem.

14. This is all the knowledge I can provide on PetaLinux. There exists documentation at:
    ● **PetaLinux documentation (UG1144)**:
      https://docs.amd.com/r/en-US/ug1144-petalinux-tools-reference-guide
    Xilinx forums might have people that have similar problems to you:
    https://support.xilinx.com/s/topiccatalog?language=en_US