

Preparing Kri-App Firmwares for KV260 Board

For running the “Prophesee-Event-VitisAI” Kria Application we have to create the firmware files which includes the compiled device tree (DTBO), hardware design binary file (BIN) , Acceleration file (XCLBIN) and shell.json file.

After creating the Vitis Hardware Platform from VIVADO with “MIPI camera pipeline and PS” we can get the XSA or hardware file.

By using this XSA we have to create the device tree using XSCT.

The default generated device tree is not complete , so for completing it we have to perform following “device tree customization” step:

Device Tree Customizations:

In generated (default) device tree or pl.dtsi we have to update as following.

In mipi-csi-tx node update compatible to :

```

smart_tracker_out: endpoint {
    remote-endpoint = <packetizer_in>;
};

};

mipi_csi2_rx_subsys0: mipi_csi2_rx_subsystem@a0010000 {
    clock-names = "lite_aclk", "dphy_clk_200M", "video_aclk";
    clocks = <zynqmp_clk 71>, <mtsc_clk 0>, <zynqmp_clk 71>;
    compatible = "psee,mipi-csi2-rx-subsystem-5.2", "psee,mipi-csi2-rx-subsystem-5.0";
    interrupt-names = "csirxss_csi_irq";
    interrupt-parent = <gic>;
    interrupts = <0 92 4>;
    reg = <0x0 0xa0010000 0x0 0x2000>;
    xlnx,axis-tdata-width = <64>;
    xlnx,csi-pxl-format = <0x2a>;
    xlnx,dphy-present ;
    xlnx,en-active-lanes ;
    xlnx,max-lanes = <2>;
    xlnx,ppc = <1>;
    xlnx,vc = <4>;
    mipi_csi_ports@mipi_csi2_rx_subsys0: ports {
        #address-cells = <1>;
        #size-cells = <0>;
        mipi_csi_port@mipi_csi2_rx_subsys0: port@1 {
            /* Fill cfa-pattern=rggb for raw data types, other fields video-format and video-width user needs to fill */
            reg = <1>;
            xlnx,cfa-pattern = "rggb";
            xlnx,video-format = <12>;
            xlnx,video-width = <8>;
            mipi_csi_rx_out@mipi_csi2_rx_subsys0: endpoint {
                remote-endpoint = <axis_tkeep_handler_0@mipi_csi2_rx_subsys0>;
            };
        };
        mipi_csi_port@mipi_csi2_rx_subsys0: port@0 {
            /* Fill cfa-pattern=rggb for raw data types, other fields video-format,video-width user needs to fill */
            /* User need to add something like remote-endpoint=<out> under the node csiss_in:endpoint */
            reg = <0>;
            xlnx,cfa-pattern = "rggb";
            xlnx,video-format = <12>;
            xlnx,video-width = <8>;
            mipi_csi_in@mipi_csi2_rx_subsys0: endpoint {
                data-lanes = <1 2>;
                remote-endpoint = <inmx636>;
            };
        };
    };
};

ps_host_if_0: ps_host_if@a0030000 {
    /* This is a place holder node for a custom IP, user may need to update the entries */
    clock-names = "aclk";
    clocks = <zynqmp_clk 71>;

```

For any Queries, please contact us at : info@logictronix.com

After device tree customized, it is now ready for compiling and getting DTBO file.

Device Tree compilation:

You can use the DTC in Ubuntu or Linux machine to create the DTBO from DTSI file.

Firmware ready for Kria

Now you have the:

1. DTBO
2. BIN
3. XCLBIN
4. Shell.json

After you copy these firmware files, it is now ready to copy it in Kria board and load using xmutil loadapp command.