Device-Tree Overlay Generation

A device tree overlay allows us to inform the Linux kernel about dynamically loaded hardware. This allows the kernel to load the registered drivers for the new hardware.

DT Generation Script

Currently, we have the `gen_pl_overlay.sh` script that does the following:

- clones the Xilinx DT spec data (if not already present in the `work` directory)
- clones a DT compiler (if not already present in the `work` directory)
- patches the Xilinx DT spec for AXI-DMA to support RDAI
- generates the DT overlay `pl.dtbo`

The `gen_pl_overlay.sh` script takes the following inputs:

- hardware design/description file (`.xsa` file)
- RDAI VLNV for the device (as provided in the property `xcel_rdai_vlnv` in `config.json`)

What RDAI info is added to the DT overlay?
The `gen_pl_overlay.sh` creates a special node in the DT for RDAI.
The node has the following schema:

```
rdai {
  compatible = "rdai,axi-dma";
  rdai,device-vlnv = RDAI_DEVICE_VLNV;
  dma-names = "dma_tx", "dma_rx";
  dmas = <&axi_dma_0 0 &axi_dma_0 1>;
};
```

- The compatible string `rdai,axi-dma` allows the RDAI DMA driver to be probed when the DT is loaded by the Linux kernel
- `RDAI_DEVICE_VLNV` is provided as an input to the script
- dma-names and dmas refers to the DMA channels present on the instantiated AXI-DMA IP

Requirements on Bitstream Naming

The generated device tree overlay creates a `firmware-name` property that provides the name of the

bitstream file associated with the overlay. The expected format of the bitstream is not the raw `.bit` file format but the packaged `.bin` format.
The name of the packaged bitstream has the following schema : `VIVADO_PROJECT_NAME.bit.bin`

The DT overlay support in the Linux kernel checks the bitstream filename against the `firmware-name` property before loading the DT overlay. These must match!

To generate the packaged bitstream, we first have to create a bootgen file:

bootgen_script.bif:

```
all:
{
  [destination_device = pl] <VIVADO_PROJECT_NAME>.bin
}
```

Then we can run:

```
$> bootgen -image bootgen_script.bif -arch zynqmp -o <VIVADO_PROJECT_NAME>.bit.bin -w
```

Once the run completes, the bootgen tool will produce the file `<VIVADO_PROJECT_NAME>.bit.bin`

Both the `<VIVADO_PROJECT_NAME>.bit.bin` and DT overlay `pl.dtbo` are used to program the FPGA device.

> Note:
> The **VIVADO_PROJECT_NAME** will come from the `name` property in the `config.json` file used to create the Vivado hardware generation script

Target Goal

At this moment, the bitstream packaging task is done manually. We would want to provide a script that does both DT overlay generation and bitstream packaging based on the same `config.json` file used for Vivado hardware generation script.