RFDev: Linux Kernel Driver for Routing Fabrics in AXIOM Beta Main Board

GSoC 2020 - Phase 1 - Mid

Swaraj Hota

Contents

Claim PIC addresses

- Firstly, basic i2c driver code was added
- The driver claims i2c addresses 0x40-0x4f, which PIC responds to
- A device tree entry was added for this (inside the entry for i2cbus-2):

```
pic@40 {
compatible = "apertus,pic-rf-interface";
reg = <0x40 0x10 0x60 0x10>;
#address-cells = <1>;
#size-cells = <1>;
};
```

- Only 0x40 claimed by default, others claimed through "dummy clients"

Create a Sysfs interface

- A group of attributes for the driver found in /sys/devices/rfdev/
- Currently, the group contains only “idcode” attribute
- Reading out idcode is the first successful test for communicating with the routing fabrics through the PIC
- Other useful attributes can be added in this group (which are not provided by FPGA Manager)

Integrate FPGA Manager

- A Linux Driver Framework which will allow us to easily upload firmware images into the MachXO2's SRAM
- Once registered, provides a sysfs interface inside /sys/class/fpga_manager/fpga#/
- "firmware" attribute will allow us to do something like:

  ```
  echo firmware.bin >
  /sys/class/fpga_manager/fpga1/firmware
  ```

- API functions: write_init, write and write_complete, do the uploading
- "state" attribute will allow us to read the current state of the FPGA

Upcoming

- To fill in the API functions by FPGA Manager with appropriate read/write commands to the PIC, and get a firmware uploaded successfully
- To research and find a suitable way to implement a JTAG interface
- JTAG interface will allow software like OpenOCD to debug the routing fabrics as well as upload/flash new firmware
- To have a checksum/hash reading attribute that helps us to know which firmware is uploaded

Code

Link to Github repository:
`https://github.com/Swaraj1998/axiom-beta-rfdev.git`