# **PMIC Programming, Power monitor and Resets**

* Place the scripts in the BOOT partition of the sd card
* Insert the sd card in the Zybo / Zed board and power on the board and wait for it to boot successfully
* After the Zed or Zybo board is booted successfully -
  + Check for **ls ../../media/**
  + If nothing present in it then run the following commands to mount the BOOT partition onto the sd card
    - i. **mkdir -p /mnt/sd**
    - ii.  **mount /dev/mmcblk0p1 /mnt/sd**
    - iii. Check for the script if present in the path **ls /mnt/sd**
    - iv. Navigate to that directory **cd /mnt/sd**
* Make the connections of the Zybo / Zed board PMOD interface with the J37 header of the validation board with proper I2C0 and I2C1 interface having a common ground
* After placing the power modules in their respective slots, power on the validation board
  + a. I2C0 - J60
  + b. I2C1 - J4

**\*\*\* Note \*\*\* Do not interchange the power modules in any case**

**\*\*\* Note \*\*\* Wait for some time to stabilise the regulators**

**\*\*\* Note \*\*\* After power on ensure that the 12V, 1.8V and 3.3V power supply regulator LEDs were turned on. If not then turn off the power immediately**

* Slaves present on the I2C0 bus -
  + 0x10, 0x11, 0x12, 0x13 - ADS7138 (power monitors)
  + 0x21 - I/O Expander for Resets control (both MKB and GFx)
  + 0x20, 0x70, 0x72 - PMIC power modules (I/O expander to control the power goods of the PMICs and MUX selection channels)
  + 0x6C - Clock buffer
  + 0x68 - 100MHz clock generator

**\*\*\* Note \*\*\* If J13 header is closed the clock generator doesn’t fall on the I2C0 bus**

* Slaves present on the I2C1 bus -
  + 0x20, 0x70, 0x72 - PMIC power modules (I/O expander to control the power goods of the PMICs and MUX selection channels)
* Run the commands **i2cdetect -y -r -a 0**  and **i2cdetect -y -r -a 1** to check if all the slave devices were present on the respective buses with respective slave addresses

**\*\*\* Note \*\*\* The I/O expander (0x21) won’t present initially on the I2C0. It can be seen after the voltage programming of the PMIC power modules as it has a dependency of VDD\_IO\_1V8 (1.8V) supply**

* To power-up the rails of J4 slot run **./J4\_PMIC\_ENABLE\_VER2.sh** script
* To power-up the rails of the J60 slot run **./J60\_PMIC\_ENABLE\_VER2.sh** script
* To monitor the power rails have been programmed successfully run **./power\_monitor.sh** script
* Ensure that all the voltage values are within the ranges

**\*\*\* Note \*\*\* If any of the power rail noted to be recorded as zero value then immediately turn off the board**

**\*\*\* Note \*\*\* Wait for some time to stabilise the power rails and start your respective work**

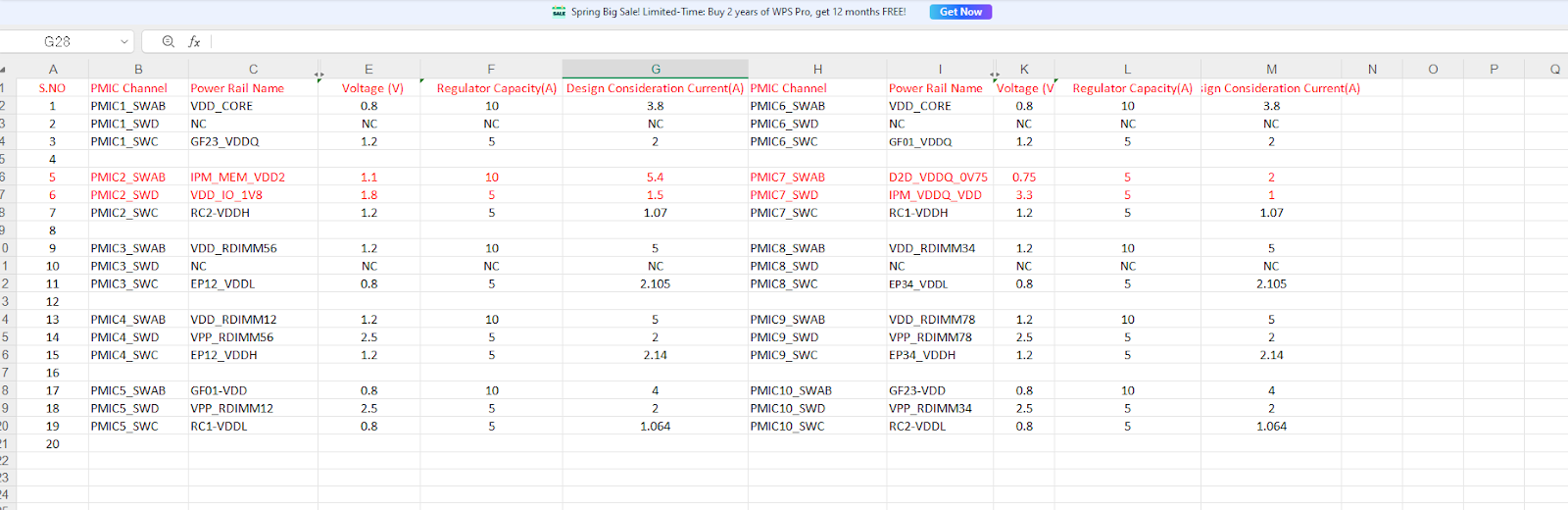
* Rerun the commands **i2cdetect -y -r -a 0**  and **i2cdetect -y -r -a 1** to check if all the slave devices were present on the respective buses with respective slave addresses

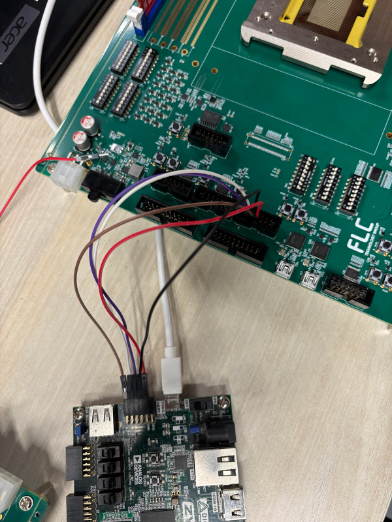
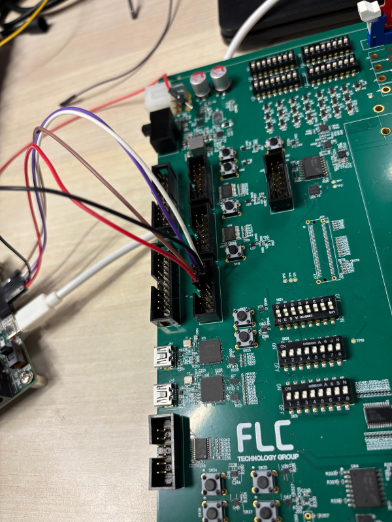
**\*\*\* Note \*\*\* Now you should be seeing the I/O expander (0x21) on the I2C0 bus**

* To check the status of the Power goods of the PMIC power modules run the following commands -
  + **i2cget -y 0 0x20 0x01** - This should return **0x1f** confirming that all the PMICs on I2C0 (J60 slot) have been programmed successfully and in a good state
  + **i2cget -y 1 0x20 0x01** - This should return **0x1f** confirming that all the PMICs on I2C0 (J60 slot) have been programmed successfully and in a good state
* To issue a reset for the MKB (using the I/O control) run **./MKB\_RESET.sh** script
* To issue a reset for the GFx (using the I/O control) run **./GF\_RESET.sh** script
* To issue all the resets at once (using the I/O control) run **./GLOBAL\_RESET.sh** script
* For the 100MHz clock generator and clock buffer to operate close the J13 header (short pins 1,2) with a jumper before power on the validation board
* To reprogram the J4 (I2C1 slot) PMIC power module if necessary run **./RESET\_J4\_I2C1\_PMIC.sh** script
* To reprogram the J60 (I2C0 slot) PMIC power module if necessary run **./RESET\_J60\_I2C0\_PMIC.sh** script

**\*\*\* Note \*\*\* Run these scripts if incase any voltage reading has exceeded its limit and a reprogramming is to be done to nullify it by disabling the power goods of the PMICs**

**\*\*\* Note \*\*\* If the Power goods were downed then for the next power on of the validation board the PMICs of J4 (I2C1) and J60 (I2C0) slots should be programmed with their respective voltage values and to placed in their respective slots**



| **Powers {I2C1 - J60 (PMIC1), I2C2 - J4 (PMIC2)}** | | | |
| --- | --- | --- | --- |
|  |  |  |  |
| **MKB** | | | |
| **SI NO** | **Power Rail Name** | **Voltage** | **Description** |
| 1 | VDD\_CORE | 0.8 V | Core Power (MKB / GF) |
| 2 | VDD\_CORE | 0.8 V |
| 3 | VDD\_IO\_1V8 | 1.8 V | I/O Power |
| 4 | IPM\_MEM\_VDD2 | 1.1 V | IPM |
| 5 | IPM\_VDDQ\_VDD | 3.3 V |
| 6 | RC1-VDDH | 1.2 V | RC |
| 7 | RC1-VDDL | 0.8 V |
| 8 | RC2-VDDH | 1.2 V |
| 9 | RC2-VDDL | 0.8 V |
| 10 | EP12-VDDH | 1.2 V | EP |
| 11 | EP12-VDDL | 0.8 V |
| 12 | EP34-VDDH | 1.2 V |
| 13 | EP34-VDDL | 0.8 V |
| 14 | D2D\_VDDQ\_0V75 | 0.75 V | D2D |
|  |  |  |  |
| **GF** | | | |
| **SI NO** | **Power Rail Name** | **Voltage** | **Description** |
| 1 | GF01-VDD | 0.8 V | GF0 / GF1 (combined) |
| 2 | GF01-VDDQ | 1.2 V |
| 3 | GF23-VDD | 0.8 V | GF2 / GF3 (combined) |
| 4 | GF23-VDDQ | 1.2 V |
| 5 | VDD\_RDIMM12 | 1.2 V | DDR4 Memory |
| 6 | VDD\_RDIMM34 | 1.2 V |
| 7 | VDD\_RDIMM56 | 1.2 V |
| 8 | VDD\_RDIMM78 | 1.2 V |
| 9 | VPP\_RDIMM12 | 2.5 V |
| 10 | VPP\_RDIMM34 | 2.5 V |
| 11 | VPP\_RDIMM56 | 2.5 V |
| 12 | VPP\_RDIMM78 | 2.5 V |
|  |  |  |  |
| **Resets** | | | |
| **MKB and GF** | | | |
| **SI NO** | **Rest Name** | **Switch** | **Test Point** |
| 1 | CB\_nPOR | SW8 | R231, R96 |
| 2 | CB\_nRST | SW8, SW12 | R230, R111 |
| 3 | CORERSTn | SW8, SW12, SW9 | R229, R102 |
| 4 | GF0\_CB\_nPOR | SW15 | R228, R136 |
| 5 | GF0\_CB\_nRST | SW15, SW25 | R227, R196 |
| 6 | GF0\_CORERSTn | SW15, SW25, SW23 | R229, R194 |
| 7 | GF1\_CB\_nPOR | SW19 | R225, R155 |
| 8 | GF1\_CB\_nRST | SW19, SW35 | R224, R281 |
| 9 | GF1\_CORERSTn | SW19, SW35, SW34 | R223, R280 |
| 10 | GF2\_CB\_nPOR | SW37 | R252, R327 |
| 11 | GF2\_CB\_nRST | SW37, SW36 | R251, R326 |
| 12 | GF2\_CORERSTn | SW37, SW36, SW41 | R250, R383 |
| 13 | GF3\_CB\_nPOR | SW40 | R249, R380 |
| 14 | GF3\_CB\_nRST | SW40, SW39 | R248, R366 |
| 15 | GF3\_CORERSTn | SW40, SW39, SW38 | R247, R365 |
|  |  |  |  |
| **Clocks** | | | |
| **SI NO** | **Reference CLK** | **CLK / CLK Buffer** | **Test Point** |
| 1 | 25MHz | Oscillator - X1 | R261 |
| 2 | OSCCLK\_0 | R269 |
| 3 | GF0\_OSCCLK | R260 |
| 4 | GF1\_OSCCLK | R243 |
| 5 | GF2\_OSCCLK | R242 |
| 6 | GF3\_OSCCLK | R267 |
| 7 | 133.33MHz | Oscillator - Y3 | R239 |
| 8 | REF\_IPM\_0 | R236 |
| 9 | REF\_IPM\_1 | R238 |
| 10 | REF\_IPM\_2 | R235 |
| 11 | REF\_IPM\_3 | R234 |
| 12 | 100MHz | Reference clock for Clock Generator  CLOCK\_GEN\_1\_P / CLOCK\_GEN\_1\_N | C7, C8 |
| 13 | Clock Generator Output  GEN\_REF\_CLK\_100MHz\_P / GEN\_REF\_CLK\_100MHz\_N | C13, C14 |
| 14 | Clock Buffer Input  PCIE\_REF\_CLK\_P / PCIE\_REF\_CLK\_N | R19, R20 |
| 15 | 100MHz differential clock for D2D PLL  REF\_100M\_P[0] / REF\_100M\_N[0] | C118, C119 |
| 16 | 100Mz differential clock for D2D PLL  REF\_100M\_P[1] / REF\_100M\_N[1] | C124, C125 |
| 17 | PCIe/CXL RC PHY reference clock  RC\_X8\_REF\_CLK0\_P / RC\_X8\_REF\_CLK0\_N | C130, C131 |
| 18 | PCIe/CXL RC PHY reference clock  RC\_X8\_REF\_CLK1\_P / RC\_X8\_REF\_CLK1\_N | C136, C137 |
| 19 | PCIe/CXL endpoint PHY reference clock  EP0\_REF\_CLK\_100MHz\_P / EP0\_REF\_CLK\_100MHz\_N | C144, C145 |
| 20 | PCIe/CXL endpoint PHY reference clock  EP1\_REF\_CLK\_100MHz\_P / EP1\_REF\_CLK\_100MHz\_N | C149, C150 |
| 21 | PCIe/CXL endpoint PHY reference clock  EP2\_REF\_CLK\_100MHz\_P / EP2\_REF\_CLK\_100MHz\_N | C153, C154 |
| 22 | PCIe/CXL endpoint PHY reference clock  EP3\_REF\_CLK\_100MHz\_P / EP3\_REF\_CLK\_100MHz\_N | C160, C161 |
| 23 | PCIe/CXL RC reference clock  CLK\_CXL\_CONN\_P / CLK\_CXL\_CONN\_N | C162, C163 |
| 24 | PCIe/CXL RC PHY reference clock  RC\_CXL\_REF1\_P / RC\_CXL\_REF1\_N | C164, C165 |
| 25 | 100MHz differential clock for DDR  GF0\_REF\_100M\_P / GF0\_REF\_100M\_N | C166, C167 |
| 26 | 100MHz differential clock for DDR  GF1\_REF\_100M\_P / GF1\_REF\_100M\_N | C168, C169 |
| 27 | 100MHz differential clock for DDR  GF2\_REF\_100M\_P / GF2\_REF\_100M\_N | C155, C52 |
| 28 | 100MHz differential clock for DDR  GF3\_REF\_100M\_P / GF3\_REF\_100M\_N | C151, C147 |
| 29 | NVMe0 reference clock  PCIE\_RC\_NVME0\_REF\_CLK\_P / PCIE\_RC\_NVME0\_REF\_CLK\_N | C146, C139 |
| 30 | NVMe1 reference clock  PCIE\_RC\_NVME1\_REF\_CLK\_P / PCIE\_RC\_NVME1\_REF\_CLK\_N | C138, C133 |
| 31 | SSD1 reference clock  PCIE\_RC\_SSD1\_REF\_CLK\_P / PCIE\_RC\_SSD1\_REF\_CLK\_N | C132, C127 |
| 32 | SSD0 reference clock  PCIE\_RC\_SSD\_REF\_CLK\_P / PCIE\_RC\_SSD\_REF\_CLK\_N | C126, C121 |
| 33 | PCIE\_CLK\_BUFFER\_18\_P / PCIE\_CLK\_BUFFER\_18\_N | C120, C116 |