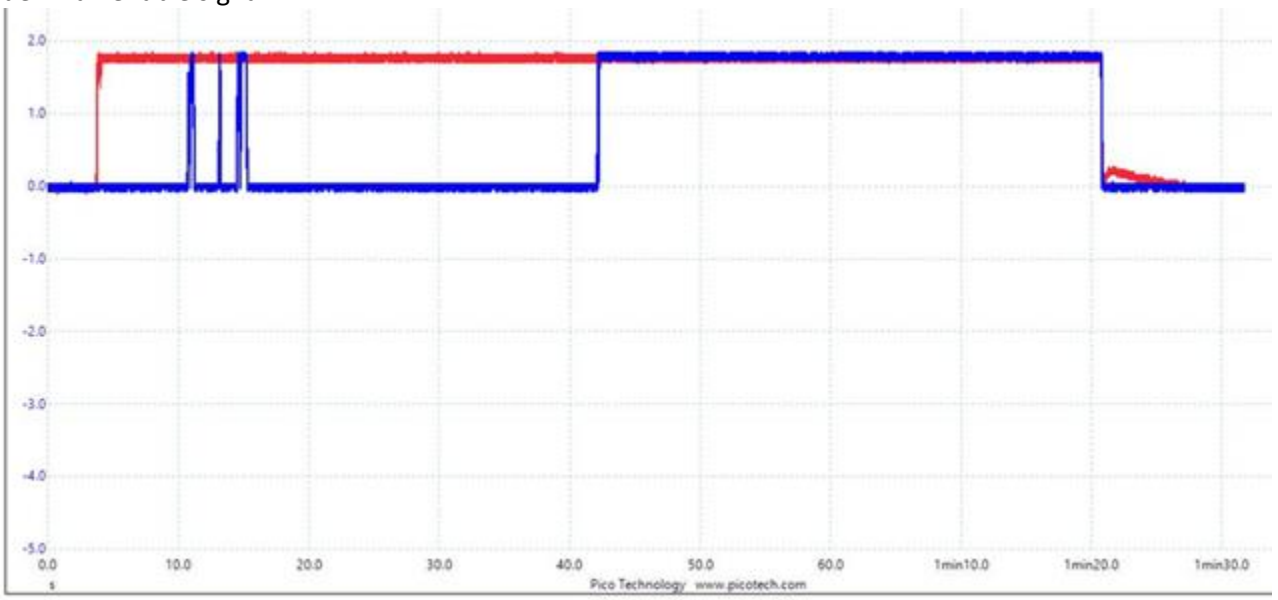
1) WLAN module initialization

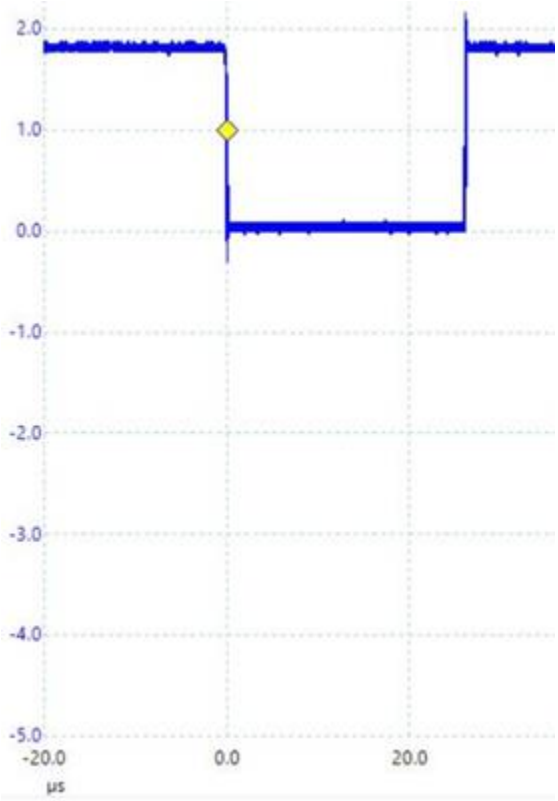
We suspect that the WLAN module start-up might could cause the problem. Module start up looks like this:

red: operating voltage

blue: wlan enable signal



Looks like for some reason during boot linux pincontrol driver enables/disables WLAN enable signal multiple times. Timing of consecutive enable/disable timing does not fulfill minimum 60us requirement



Device tree:

```
……
&mmc1 {
//      dmas = <&edma_xbar 12 0 1
//              &edma_xbar 13 0 2>;
//      dma-names = "tx", "rx";
        status = "okay";
        vmmc-supply = <&wlan_en_reg>;
        bus-width = <4>;
        ti,non-removable;
        enable-sdio-wakeup;
        cap-power-off-card;
        ti,needs-special-hs-handling;
        keep-power-in-suspend;
        pinctrl-names = "default";
        pinctrl-0 = <&mmc1_pins &wl18xx_pins>;

        #address-cells = <1>;
        #size-cells = <0>;
        wlcore: wlcore@0 {
                compatible = "ti,wl1831";
                reg = <2>;
                interrupt-parent = <&gpio2>;
                interrupts = <25 IRQ_TYPE_EDGE_RISING>;
        };
};
……
        wlan_en_reg: fixedregulator_wlan_en {
                compatible = "regulator-fixed";
                regulator-name = "wlan-en-regulator";
                regulator-min-microvolt = <1800000>;
                regulator-max-microvolt = <1800000>;
                startup-delay-us= <70000>;

                /* WL_EN */
                gpio = <&gpio2 22 0>;
                enable-active-high;
        };
……
        mmc1_pins: pinmux_mmc1_pins {
                pinctrl-single,pins = <
                        AM33XX_IOPAD(0x900, PIN_INPUT_PULLUP |
MUX_MODE0)       /* wl sdio clk */
                        AM33XX_IOPAD(0x904, PIN_INPUT_PULLUP |
MUX_MODE0)       /* wl sdio cmd */
                        AM33XX_IOPAD(0x8F0, PIN_INPUT_PULLUP |
MUX_MODE0)       /* wl sdio d3  */
                        AM33XX_IOPAD(0x8F4, PIN_INPUT_PULLUP |
MUX_MODE0)       /* wl sdio d2  */
                        AM33XX_IOPAD(0x8F8, PIN_INPUT_PULLUP |
MUX_MODE0)       /* wl sdio d1  */
                        AM33XX_IOPAD(0x8FC, PIN_INPUT_PULLUP |
MUX_MODE0)       /* wl sdio d0  */
                        AM33XX_IOPAD(0x87C, PIN_OUTPUT_PULLUP |
MUX_MODE7)      /* WLAN_LDO_EN  */
                        >;
        };

        wl18xx_pins: pinmux_wl18xx_pins {
                pinctrl-single,pins = <
                        AM33XX_IOPAD(0x8E0, PIN_OUTPUT_PULLUP  |
MUX_MODE7)     /* wlan en */
```

```
                        AM33XX_IOPAD(0x8EC, PIN_INPUT_PULLDOWN |
MUX_MODE7)      /* wl irq  */
                 >;
        };
```

This configuration configures wlan enable gpio as pull-up (as proposed by module integration document), however this causes the GPIO to high before kernel configures it as output and drives it low.
What could cause pincontrol to drive enable signal multiple time up/down?
Can WLAN module enable signaling permanently damage the module?

2) WLAN module fix
   For the WLAN modules that not communicate via SDIO, is it possible to recover them somehow e.g. via debug interface?