

Zen Cape's LED Guide

by Brian Fraser

Last update: March 9, 2023

Target Linux Kernel: 5.10

This document guides the user through:

1. Enabling the tri-colour LED #2 on the Zen cape.
2. Controlling the Red, Green, and Blue lights via the Linux command line.

Note: Use of the tri-colour LED #1 on the Zen cape is covered in the PWM guide. That LED is connected to three PWM channels and can generate mixed (analog) colours. LED #1 is connected to GPIO pins and as such each colour is either on or off.

Table of Contents

| | |
|---|---|
| 1. Zen LEDs..... | 2 |
| 2. Device Tree: Enabling the LEDs..... | 2 |
| 3. Driving the LEDs..... | 3 |
| 4. Device Tree Source (ZENCAPE_LEDS.dts)..... | 4 |

Formatting:

1. Host (desktop) commands starting with `(host)$` are Linux console commands:
`(host)$ echo "Hello world"`
2. Target (board) commands start with `(bbg)$`:
`(bbg)$ echo "On embedded board"`
3. Almost all commands are case sensitive.

Revision History:

- Nov 2, 2019: Initial version for Kernel 4.9.
- Mar 15, 2021: Updated prompts to show host/bbg.
- Mar 9, 2023: Updated .dts file to free up pins from universal cape on load (kernel 5.10).

1. Zen LEDs

There are two tri-colour LEDs on the Zen cape, both to the left of the 14-seg display:

- LED1 is connected to three PWM channels and is covered in the PWM guide.
- LED2 is connected to three GPIO pins, and its operation is described here.

See the LED guide (for just the BeagleBone's built in LEDs) for more on using the LEDs.

2. Device Tree: Enabling the LEDs

For Linux to treat the red, green, and blue components of the tri-colour LED as actual LEDs (vs unconfigured GPIO pins), we must load a device tree using the cape manage. **Do all the following on the target.**

1. Create a new folder to build the device tree overlay:

```
(bbg) $ mkdir ~/ZenCapeLedDts  
(bbg) $ cd ~/ZenCapeLedDts
```

2. Create the dts file:

```
(bbg) $ nano ZENCAPE_LEDS.dts
```

- Copy the contents of the ZENCAPE_LEDS.dts file (last section in this guide) into this file.

3. Compile the device tree source (.dts) file into a “blob overlay” file (.dtbo):

```
(bbg) $ dtc -O dtb -o ZENCAPE_LEDS-00A0.dtbo -b 0 -@ ZENCAPE_LEDS.dts
```

4. Deploy the .dtbo file to the target's /lib/firmware folder:

```
(bbg) $ sudo cp ZENCAPE_LEDS-00A0.dtbo /lib/firmware
```

5. Load the device overlay in /boot/uEnv.txt

- Copy the existing uEnv.txt:

```
(bbg) $ sudo cp /boot/uEnv.txt /boot/uEnv-BeforeZenLed.txt
```

- Edit uEnv.txt:

```
(bbg) $ sudo nano /boot/uEnv.txt
```

- Edit the Additional custom capes section.

Below shows the setup you'll have if you are already loading the audio cape. If you are not loading these, you may comment out those lines

```
###Additional custom capes  
uboot_overlay_addr4=/lib/firmware/BB-BONE-AUDI-02-00A0.dtbo  
uboot_overlay_addr6=/lib/firmware/ZENCAPE_LEDS-00A0.dtbo
```

- You can use uboot_overlay_addr0 through uboot_overlay_addr7 for any of the capes being loaded. If you use 0-3, it replaces the cape support for any automatically detected capes configured to load at that slot. If you have no physical capes connected (which are automatically detected), then you can use any of the 8 slots you like.

- You only need to load the capes you need.

6. Reboot the target

7. Troubleshooting

- See the Audio guide's last section (on the course website) to recover from a corrupted uEnv.txt. You should be able to copy back the /boot/uEnv-BeforeZenLed.txt to restore your previous working state.

(Note: restore process not working on kernel 5.10, Mar 2023)

3. Driving the LEDs

1. List all files in the `/sys/class/leds/` directory:

```
(bbg)$ ls /sys/class/leds/
```

```
beaglebone:green:usr0  beaglebone:green:usr2  zencape:blue  zencape:red  
beaglebone:green:usr1  beaglebone:green:usr3  zencape:green
```

2. Change to the Zen cape's blue LED folder:

```
(bbg)$ cd /sys/class/leds/zencape\:blue
```

- Note that you cannot type ':' in a path, you must escape it.

3. Change the trigger to a heartbeat:

```
(bbg)$ echo heartbeat | sudo tee trigger
```

4. Manually turn on the blue LED:

```
(bbg)$ echo none | sudo tee trigger
```

```
(bbg)$ echo 1 | sudo tee brightness
```

5. Troubleshooting

- If the zencape LEDs are not displayed, double check the `uEnv.txt` file correctly configures the cape, and ensure you removed the '#' from the front of the line.

4. Device Tree Source (ZENCAPE_LEDS.dts)

```
/*
 * Configure the three signals for the tri-colour LED on the Zen cape to be an LED in Linux
 * Based on: https://github.com/nomel/beaglebone/blob/master/led-header/generated/led-P9.12-00A0.dts
 * Written by Brian Fraser; released under GPL and BSD.
 *
 * Compile with:
 * dtc -O dtb -o ZENCAPE_LEDS-00A0.dtbo -b 0 -@ ZENCAPE_LEDS.dts
 * Copy file:
 * sudo cp ZENCAPE_LEDS-00A0.dtbo /lib/firmware
 * Load by editing /boot/uEnv.txt
 * uboot_overlay_addr4=/lib/firmware/ZENCAPE_LEDS-00A0.dtbo
 * Use like any Linux LEDs via /sys/class/leds/zencape:red/...
 * Modified Oct 2019: Works with Kernel 4.9.
 * Feb 2023 for Kernel 5.10+ (Bullseye): Free up the pins used by the universal cape
 */
/dts-v1/;
/plugin/;
/ {
    compatible = "ti,beaglebone", "ti,beaglebone-black", "ti,beaglebone-green";

    /* identification */
    part-number = "ZENCAPE_LEDS";
    /* version = "00A0"; */

    /* state the resources this cape uses */
    exclusive-use =
        /* the pin header uses */
        "P9.12", /* Red */
        "P9.13", /* Green */
        "P9.11", /* Blue */
        /* the hardware IP uses */
        "gpio1_28",
        "gpio0_31",
        "gpio0_30";

    /* Free up the pins used by the cape from the pinmux helpers. */
    fragment0 {
        target = <&ocp>;
        __overlay__ {
            P9_12_pinmux { status = "disabled"; }; /* P9_12: Red LED */
            P9_13_pinmux { status = "disabled"; }; /* P9_13: Green LED */
            P9_11_pinmux { status = "disabled"; }; /* P9_11: Blue LED */
        };
    };

    /* rxDisable_pullNone state */
    fragment_1 {
        target = <&am33xx_pinmux>;
        __overlay__ {
            configure_pins_rxDisable_pullNone: pinmux_gpio_rxDisable_pullNone {
                pinctrl-single,pins = <
                    0x78 0xf /* P9.12 Output Mode 7, no pull */
                    0x74 0xf /* P9.13 Output Mode 7, no pull */
                    0x70 0xf /* P9.11 Output Mode 7, no pull */
                >;
            };
        };
    };

    fragment_2 {
        target = <&ocp>;
        __overlay__ {
            zencape_led_helper {
                compatible = "gpio-leds";
                pinctrl-names = "default";
                pinctrl-0 = <&configure_pins_rxDisable_pullNone>;

                zencape-red {
                    label = "zencape:red";
                    gpios = <&gpio1_28 0>;
                    linux,default-trigger = "heartbeat";
                    default-state = "off";
                };

                zencape-green {
                    label = "zencape:green";
                    gpios = <&gpio0_31 0>;
                    linux,default-trigger = "heartbeat";
                    default-state = "off";
                };

                zencape-blue {
                    label = "zencape:blue";
                    gpios = <&gpio0_30 0>;
                    linux,default-trigger = "heartbeat";
                    default-state = "on";
                };
            };
        };
    };
};
```