

Flash LED handling under Linux

Some LED devices provide two modes - torch and flash. In the LED subsystem those modes are supported by LED class (see Documentation/leds/leds-class.rst) and LED Flash class respectively. The torch mode related features are enabled by default and the flash ones only if a driver declares it by setting LED_DEV_CAP_FLASH flag.

In order to enable the support for flash LEDs CONFIG_LEDS_CLASS_FLASH symbol must be defined in the kernel config. A LED Flash class driver must be registered in the LED subsystem with led_classdev_flash_register function.

Following sysfs attributes are exposed for controlling flash LED devices: (see Documentation/ABI/testing/sysfs-class-led-flash)

- flash_brightness
- max_flash_brightness
- flash_timeout
- max_flash_timeout
- flash_strobe
- flash_fault

V4L2 flash wrapper for flash LEDs

A LED subsystem driver can be controlled also from the level of VideoForLinux2 subsystem. In order to enable this CONFIG_V4L2_FLASH_LED_CLASS symbol has to be defined in the kernel config.

The driver must call the v4l2_flash_init function to get registered in the V4L2 subsystem. The function takes six arguments:

- dev:
flash device, e.g. an I2C device
- of_node:
of_node of the LED, may be NULL if the same as device's
- fled_cdev:
LED flash class device to wrap
- iled_cdev:
LED flash class device representing indicator LED associated with fled_cdev, may be NULL
- ops:
V4L2 specific ops
 - external_strobe_set
defines the source of the flash LED strobe - V4L2_CID_FLASH_STROBE control or external source, typically a sensor, which makes it possible to synchronise the flash strobe start with exposure start,
 - intensity_to_led_brightness and led_brightness_to_intensity
perform enum led_brightness <-> V4L2 intensity conversion in a device specific manner - they can be used for devices with non-linear LED current scale.
- config:
configuration for V4L2 Flash sub-device
 - dev_name
the name of the media entity, unique in the system,
 - flash_faults
bitmask of flash faults that the LED flash class device can report; corresponding LED_FAULT* bit definitions are available in <linux/led-class-flash.h>,
 - torch_intensity
constraints for the LED in TORCH mode in microamperes,
 - indicator_intensity
constraints for the indicator LED in microamperes,
 - has_external_strobe
determines whether the flash strobe source can be switched to external,

On remove the v4l2_flash_release function has to be called, which takes one argument - struct v4l2_flash pointer returned previously by v4l2_flash_init. This function can be safely called with NULL or error pointer argument.

Please refer to drivers/leds/leds-max77693.c for an exemplary usage of the v4l2 flash wrapper.

Once the V4L2 sub-device is registered by the driver which created the Media controller device, the sub-device node acts just as a node of a native V4L2 flash API device would. The calls are simply routed to the LED flash API.

Opening the V4L2 flash sub-device makes the LED subsystem sysfs interface unavailable. The interface is re-enabled after the V4L2 flash sub-device is closed.