

드라이버 User's 가이드

nx_gpio_ctrl driver (Nexell Gpios Control Driver) / misc



misc folder ->

- This misc driver is for control the Nexell Gpios Base Address and ALT function by DTS file.
- > misc 드라이버는 DTS 파일로 Nexell Gpios Base Address 및 ALT 기능을 제어하기 위한 드라이버입니다.
- ioctl operations are Gpio direction set, direction set check, read and write value setting.
- > ioctl 동작은 Gpio 방향 설정, 방향 설정 확인, 읽기 및 쓰기 값 설정입니다.
- According to the direction, when read operation get the current gpio value and write operation set the gpio value, direction also set.
- > 방향에 따라 읽기 연산이 현재 gpio 값을 가져오고 쓰기 연산이 gpio 값을 설정할 때 방향도 설정됩니다.

MISC 디바이스 드라이버 사용을 위해 MISC 형식으로 드라이버를 등록을 해야하며 Makefile, Board config에 해당 driver 등록 해야한다.

- How to add a Misc driver in Misk **Kconfig** → drivers/misc/Kconfig

```
config NX_GPIO_CTRL
    bool "Enable gpio test driver for training."
    depends on MISC
    help
        Enable a nexell gpio control driver.
```

MISC 드라이버 임을 명시한다.

- How to add a Misc driver in Misk **Makefile** → drivers/misc/Makefile

```
obj-$(CONFIG_NX_GPIO_CTRL) += nx_gpio_ctrl.o
```

- How to add a driver config in the **board config** → configs/s5p4418_daudio_ref_defconfig

```
CONFIG_NX_GPIO_CTRL=y
```

- **nx_gpio_ctrl.h**

```
#ifndef __NX_GPIO_CTRL_H
#define __NX_GPIO_CTRL_H

#define NX_GPIO_IOCTL_DIRECTION_IN  0x0000
#define NX_GPIO_IOCTL_DIRECTION_OUT 0x0001
#define NX_GPIO_DIR_IN              0x0000
#define NX_GPIO_DIR_OUT              0x0001

static int nx_gpio_ctrl_read(struct udevice *dev,
                             int offset, void *buf, int size);
static int nx_gpio_ctrl_write(struct udevice *dev,
                              int offset, void *buf, int size);
static int nx_gpio_ctrl_ioctl(struct udevice *dev,
                              unsigned long request, void *buf);
static int nx_gpio_ctrl_probe(struct udevice *dev);
static int nx_gpio_ctrl_ofdata_to_platdata(struct udevice *dev);

#endif
```

- 방향설정을 위한 매크로, 방향설정 확인을 위한 매크로와, read, write, ioctl, probe, platdata 함수 구현

Driver properties, dts setting

- Required properties:

- compatible: Should be "nexell,nx_gpio_ctrl"

- gpio: GPIO specification for write or read, it contains information of gpio in the same order as described by gpio-names

- pad_func: This value is function of pad for nx_gpio_set_pad_function(), this value obtained from device tree info.

- How to add a Misc dts in the **dts** → dts/s5p4418-daudio-ref.dts

```
// Nexell's LCD ON/OFF botton - Alt0

nx_gpio_ctrl@0{
    compatible = "nexell,nx_gpio_ctrl";
    gpio = <&gpio_alv 0 0>;
    pad_func = <1>;
};

// Nexell's Android Back Key botton - Alt1
```

```
nx_gpio_ctrl@1{  
    compatible = "nexell,nx_gpio_ctrl";  
    gpio = <&gpio_alv 1 0>;  
    pad_func = <1>;  
};
```
