

How to integrate PCA9450 driver into i.MX8M mini BSP

1. uboot-imx

1. In the "./drivers/power/pmic/" directory
 - A. copy pca9450.c and pmic_pca9450.c of uboot-pca9450.zip.
 - B. modify Kconfig to add pca9450

```
config DM_PMIC_PCA9450
    bool "Enable Driver Model for PMIC PCA9450"
    depends on DM_PMIC
    help
        This config enables implementation of driver-model pmic uclass features
        for PMIC PCA9450. The driver implements read/write operations.
```

- C. modify Makefile to add pca9450

```
obj-$(CONFIG_$(SPL_)DM_PMIC_BD71837) += bd71837.o
obj-$(CONFIG_$(SPL_)DM_PMIC_PCA9450) += pca9450.o
obj-$(CONFIG_PMIC_S2MPS11) += s2mps11.o
```

```
obj-$(CONFIG_POWER_BD71837) += pmic_bd71837.o
obj-$(CONFIG_POWER_PCA9450) += pmic_pca9450.o
obj-$(CONFIG_POWER_BF117E100) += pmic_bf117e100.o
```

2. In the "./include/power/" directory
 - A. copy pca9450.h of uboot-pca9450.zip.
3. In the "./include/configs/" directory
 - A. Modify imx8mm_*.h to remove bd71837 and add pca9450

```
#define CONFIG_POWER_I2C

#undef CONFIG_POWER_BD71837
#define CONFIG_POWER_PCA9450

#define CONFIG_SYS_I2C
#define CONFIG_SYS_I2C_MXC_I2C1 /* enable I2C bus 1 */
#define CONFIG_SYS_I2C_MXC_I2C2 /* enable I2C bus 2 */
#define CONFIG_SYS_I2C_MXC_I2C3 /* enable I2C bus 3 */
```

4. In the `./arch/arm/dts/` directory.
 - A. Change to `fsl-imx8mm-evk.dts` of `uboot-pca9450.zip`.
5. modify initial code of `spl.c` to replace `bd71837` code with `pca9450`.

```
* SPDX-License-Identifier: GPL-2.0+
*/
```

```
#define CONFIG_PCA9450

#include <common.h>
#include <spl.h>
#include <asm/io.h>
#include <errno.h>
#include <asm/io.h>
#include <asm/mach-imx/iomux-v3.h>
#include <asm/arch/imx8mm_pins.h>
#include <asm/arch/sys_proto.h>
#include <power/pmic.h>
#ifndef CONFIG_PCA9450
#include <power/pca9450.h>
#else
#include <power/bd71837.h>
#endif
#include <asm/arch/clock.h>
```

If use `imx8mm_evk` directory,

```
#undef CONFIG_PCA9450
int power_init_board(void)
{
    struct pmic *p;
    int ret;

    ret = power_pca9450_init(I2C_PMIC);
    if (ret)
        printf("power init failed");

    p = pmic_get("PCA9450");
    pmic_probe(p);

    /* BUCKxOUT_DVS0/1 control BUCK123 output */
    pmic_reg_write(p, PCA9450_BUCK123_DVS, 0x29);

    /* Buck 1 DVS control through PMIC_STBY_REQ */
    pmic_reg_write(p, PCA9450_BUCK1CTRL, 0x59);

    /* decrease RESET key long push time from the default 10s to 10ms */
    /* Ton_Deb of PCA9450 is 20ms and don't change */

    /* increase VDD_SOC to typical value 0.85v before first DRAM access */
    /* pmic_reg_write(p, PCA9450_BUCK10UT_DVS0, 0x14); */
    pmic_reg_write(p, PCA9450_BUCK10UT_DVS0, 0x14);
    pmic_reg_write(p, PCA9450_BUCK10UT_DVS1, 0x10);

    /* increase VDD_DRAM to 0.975v for 3Ghz DDR -> 0.95v instead of 0.975V, */
    /* because PCA9450 Buck3 can set 0.95V */
    /* Also, set B3_ENMODE=2 (ON by PMIC_ON_REQ=H & PMIC_STBY_REQ=L) */
    pmic_reg_write(p, PCA9450_BUCK30UT_DVS0, 0x1C);
    pmic_reg_write(p, PCA9450_BUCK3CTRL, 0x4A);

    /* set VDD_SNV5_0V8 from default 0.85V */
    pmic_reg_write(p, PCA9450_LD02CTRL, 0xC0);

#ifdef CONFIG_IMX8M_LPDDR4
    /* increase NVCC_DRAM_1V2 to 1.2v for DDR4 */
    pmic_reg_write(p, PCA9450_BUCK60UT, 0x18);
#endif

    /* set WDOG_B_CFG to 10b=Cold Reset, except LD01/2 */
    pmic_reg_write(p, PCA9450_RESET_CTRL, 0xA1);

    return 0;
}
```

Or if use "imx8mm_val" directory,

```
int power_init_board(void)
{
    struct pmic *p;
    int ret;

    ret = power_pca9450_init(I2C_PMIC);
    if (ret)
        printf("power init failed");

    p = pmic_get("PCA9450");
    pmic_probe(p);

    /* BUCKxOUT_DVS0/1 control BUCK123 output */
    pmic_reg_write(p, PCA9450_BUCK123_DVS, 0x29);

    /* Buck 1 DEV control through PMIC_STBY_REQ */
    pmic_reg_write(p, PCA9450_BUCK1CTRL, 0x59);

    /* decrease RESET key long push time from the default 10s to 10ms */
    /* Ton_Deb of PCA9450 is 20ms and don't change */

    /* increase VDD_SOC to typical value 0.85v before first DRAM access */
    /* pmic_reg_write(p, PCA9450_BUCK10OUT_DVS0, 0x14); */
    pmic_reg_write(p, PCA9450_BUCK10OUT_DVS1, 0x10);

    /* increase VDD_DRAM to 0.975v for 3Ghz DDR -> 0.95V instead of 0.975V, */
    /* because PCA9450 Buck3 can set 0.95V */
    /* Also, set B3_ENMODE=2 (ON by PMIC_ON_REQ=H & PMIC_STBY_REQ=L) */
    pmic_reg_write(p, PCA9450_BUCK3OUT_DVS0, 0x1C);
    pmic_reg_write(p, PCA9450_BUCK3CTRL, 0x4A);

    /* set VDD_SNV5_0V8 from default 0.85V */
    pmic_reg_write(p, PCA9450_LDO2CTRL, 0xC0);

#ifdef CONFIG_TARGET_IMX8MM_DDR4_VAL
    /* increase NVCC_DRAM_1V2 to 1.2v for DDR4 */
    pmic_reg_write(p, PCA9450_BUCK6OUT, 0x18);
#elif defined(CONFIG_TARGET_IMX8MM_DDR3L_VAL)
    /* increase NVCC_DRAM_1V35 to 1.35v for DDR3L */
    pmic_reg_write(p, PCA9450_BUCK6OUT, 0x1E);
#endif

    /* set WDOG_B_CFG to 10b(Cold Reset, except LDO1/2 */
    pmic_reg_write(p, PCA9450_RESET_CTRL, 0xA1);

    return 0;
} ? end power_init_board ?
```

2. linux-imx

1. In the "./drivers/regulator/" directory
 - A. copy pca9450-regulator.c of linux-imx8mm-pca9450.zip.
 - B. modify Kconfig to add pca9450

```
config REGULATOR_PCA9450
    tristate "NXP PCA9450 Power Regulator"
    depends on MFD_PCA9450
    help
        This driver supports PCA9450 voltage regulator chips.
```

- C. modify Makefile to add pca9450

```
obj-$(CONFIG_REGULATOR_WM8994) += wm8994-regulator.o
obj-$(CONFIG_REGULATOR_BD71837) += bd71837-regulator.o
obj-$(CONFIG_REGULATOR_PCA9450) += pca9450-regulator.o
```

```
ccflags-$(CONFIG_REGULATOR_DEBUG) += -DDEBUG
```

2. In the "./drivers/mfd/" directory

- A. copy pca9450.c of linux-imx8mm-pca9450.zip.
B. modify Kconfig to add pca9450

```
config MFD_PCA9450
    bool "PCA9450 Power Management chip"
    depends on I2C=y
    select MFD_CORE
    help
        if you say yes here you get support for the PCA9450
        Power Management chips.
```

- C. modify Makefile to add pca9450

```
obj-$(CONFIG_MFD_MXS_LRADC) += mxs-lradc.o
obj-$(CONFIG_MFD_BD71837) += bd71837.o
obj-$(CONFIG_MFD_PCA9450) += pca9450.o
```

3. In the "./include/linux/mfd/" directory

- A. copy pca9450.h of linux-imx

4. In the "./arch/arm64/configs/" directory

- A. Modify defconfig to remove bd71837 and add pca9450

```
CONFIG_MFD_SEC_CORE=y
# remove for PCA9450 CONFIG_MFD_BD71837=y
# add PCA9450 feature
CONFIG_MFD_PCA9450=y
CONFIG_REGULATOR_FIXED_VOLTAGE=y
CONFIG_REGULATOR_AXP20X=y
CONFIG_REGULATOR_FAN53555=y
CONFIG_REGULATOR_GPIO=y
CONFIG_REGULATOR_HI6421V530=y
CONFIG_REGULATOR_HI655X=y
CONFIG_REGULATOR_MAX77620=y
CONFIG_REGULATOR_PFUZE100=y
CONFIG_REGULATOR_PWM=y
CONFIG_REGULATOR_QCOM_SMD_RPM=y
CONFIG_REGULATOR_QCOM_SPMI=y
CONFIG_REGULATOR_RK808=y
CONFIG_REGULATOR_S2MPS11=y
CONFIG_REGULATOR_BD71837=y
# add PCA9450 feature
CONFIG_REGULATOR_PCA9450=y
CONFIG_RC_DEVICES=y
.....
```

5. In the `./arch/arm64/boot/dts/freescale/` directory.
 - A. Change to `fsl-imx8mm-evk.dts` of `linux-imx8mm-pca9450.zip`.