

OSのソース修正に対するアドバイスをお願い

2020年7月20日
マルタカテクノ 技術部

<経緯>

弊社より Atmark 様の Partner となっている会社にて開発を委託し、Armadillo-840 のユーザープログラム開発を委託したところ、SGX エラー及び CRC エラーが発生いたしました。

これらの対策を行うべく、委託先にもご協力を戴き、対策案を検討しているのですが、以下の理由により、OS のバージョンアップに支障を来しています。

- ① 御社のオリジナル資源 (AtmarkDist、Hermit、Linux) のソースが書き換えられている。
- ② 御社のオリジナル資源 (AtmarkDist、Hermit、Linux) に対し、複数のファイルが追加適用されている。

「この内容から、OS が書き換えられていると、判断できる内容でしょうか？」
あくまで推測ですが、LCD 周りのバッファ、解像度、同期周波数等に手が加えられているものと思われませんが、この判断は、正しいでしょうか？

また、前任の方が、現在、他の都道府県に長期出張中で、過去の開発内容を確認させていただく事が出来ない為、御社のオリジナルソースに対し、明らかな修正が加えられている、次に示す 17 本のファイル修正内容をアドバイスいただけないでしょうか？

個別にご回答いただく事が難しい場合は、テレビ会議形式でのディスカッションは可能でしょうか？

対象は、以下の (01) ~ (17) のファイルです。

差分の一部抜粋は、次ページ以降に記載いたしました。

- (01) hermit-at-3.8.1¥src¥target¥armadillo8x0¥proc.c
- (02) atmark-dist-20160927¥Makefile
- (03) atmark-dist-20160927¥vendors¥AtmarkTechno¥Armadillo-840¥etc¥default¥rc.local
- (04) atmark-dist-20160927¥vendors¥AtmarkTechno¥Armadillo-840¥etc¥init.d¥pvrsrv
- (05) linux-3.4-at20¥arch¥arm¥configs¥armadillo840_defconfig
- (06) linux-3.4-at20¥arch¥arm¥mach-shmobile¥board-armadillo840.c
- (07) linux-3.4-at20¥arch¥arm¥mach-shmobile¥Kconfig.armadillo800
- (08) linux-3.4-at20¥drivers¥clocksource¥sh_cmt_simple.c
- (09) linux-3.4-at20¥drivers¥gpu¥eurasia_km¥platform-r8a7740¥config_kernel.h
- (10) linux-3.4-at20¥drivers¥gpu¥eurasia_km¥services4¥svkm¥devices¥sgx¥sgxinit.c
- (11) linux-3.4-at20¥drivers¥gpu¥eurasia_km¥services4¥svkm¥devices¥sgx¥sgxpower.c
- (12) linux-3.4-at20¥drivers¥spi¥spi-bitbang.c
- (13) linux-3.4-at20¥drivers¥spi¥spi.c
- (14) linux-3.4-at20¥drivers¥spi¥spidev.c
- (15) linux-3.4-at20¥drivers¥video¥sh_mobile_lcdcfb.c
- (16) linux-3.4-at20¥include¥video¥sh_mobile_lcdcfb.h
- (17) linux-3.4-at20¥sound¥soc¥codecs¥cs4270.c

(01) 対象ソースファイル : hermit-at-3.8.1¥src¥target¥armadillo8x0¥proc.c

ご質問 : 以下のような修正が、オリジナルのOS (Hermit) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

オリジナルソースからの任意抜粋

～前略～

```
void __init arch_init(void)
{
    unsigned int dram_id = 0;
    unsigned char val;

    /* RWDI Setting: disable RWDI but keep a WOVF */
    val = read8(0xE6020004);
    write16(0xE6020004, 0xa500 | (val & 0x10));
#if defined(CONFIG_WDT_RMA1)
    /* RWDI enable: timeout 10s */
    write16(0xE6020000, 0x5a00);
    write16(0xE6020008, 0xa50a);
    write16(0xE6020004, 0xa587 | (val & 0x10));
#endif

    /* CPG */
    write32(RMSTPCR4, 0xff800080);
    write32(SMSTPCR4, 0xff800080);
    write32(SMSTPCR1, 0xfdfdfdf);

    write32(FRQCRB, 0x00000000);
    write32(FRQCRA, 0x62030533);
    write32(FRQCRC, 0x208a354e);
    write32(FRQCRB, 0x80331050);
    while (read32(FRQCRB) & 0x80000000); /* waiting for setting complete */
}
```

～後略～

～変更部分～

```
**** [オリジナル] ****
write32(FRQCRA, 0x62030533);
write32(FRQCRC, 0x208a354e);
write32(FRQCRB, 0x80331050);

**** [修正ファイル] ****
write32(FRQCRA, 0x62030533);
//write32(FRQCRC, 0x208a354e);
//write32(FRQCRC, 0x208a355e); // 2015.12.17 1/12 (99MHz) に設定
write32(FRQCRC, 0x208a353e); // 2016.01.25 1/6 (99*2MHz) に設定
write32(FRQCRB, 0x80331050);
```

(02) 対象ソースファイル : atmark-dist-20160927¥Makefile

ご質問 : 以下のような修正が、オリジナルのOS (Atmark-Dist) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

～変更部分～

**** [オリジナル1] ****

```
CONFIG_SHELL := $(shell if [ -x "$$BASH" ]; then echo $$BASH; ¥
```

**** [修正ファイル1] ****

```
KERNELRELEASE = ${shell make ARCH=$(ARCH) CROSS_COMPILE=$(CROSS_COMPILE) -sC ¥  
$(ROOTDIR)/$(LINUXDIR) kernelrelease}
```

```
CONFIG_SHELL := $(shell if [ -x "$$BASH" ]; then echo $$BASH; ¥
```

**** [オリジナル2] ****

```
find $(ROMFSDIR)/lib/modules -type f -name "*o" | xargs -r $(STRIP) -g; ¥  
fi
```

**** [修正ファイル2] ****

```
find $(ROMFSDIR)/lib/modules -type f -name "*o" | xargs -r $(STRIP) -g; ¥  
if [ "$$CONFIG_GPU_EURASIA_SGX540" = "m" ]; then ¥  
    mkdir -p $(ROMFSDIR)/lib/modules/3.4-at1+;/ ¥  
    $(ROMFSINST) -s /lib/modules/$(KERNELRELEASE)/kernel/drivers/gpu/eurasia_km ¥  
    /lib/modules/3.4-at1+/extra; ¥  
fi ¥  
fi
```

今回は、3.4-at1-at20の資源を対象としているのですが、

「/lib/modules/3.4-at1+・・・」の記述でも、このモジュールは有効となるのでしょうか？
同じ様な記述が、他のソースにも見受けられました。

或は、当初、-at16で開発していた経緯があるらしく、修正漏れでしょうか？

(03) 対象ソースファイル : atmark-dist-20160927¥vendors¥AtmarkTechno¥Armadillo-840¥etc¥default¥rc.local

ご質問 : 以下のような修正が、オリジナルのOS (Atmark-Dist) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

***** 【オリジナル】 *****

```
#
START_PHOTOVIEWER_WITH_QMLSCENE=y
if [ "${START_PHOTOVIEWER_WITH_QMLSCENE}" = "y" ]; then
```

***** 【修正ファイル】 *****

```
#
START_PHOTOVIEWER_WITH_QMLSCENE=n <= 何かを無効にしている と 推測
if [ "${START_PHOTOVIEWER_WITH_QMLSCENE}" = "y" ]; then
```

***** 【オリジナル】 *****

```
#
***** 【修正ファイル】 *****
fbset -fb /dev/fb0 -xres 800 -yres 481 -t 30100 40 88 32 11 128 2
echo 0 > /sys/class/graphics/fb1/blank
echo 0 > /sys/class/graphics/fb2/blank
export QT_QPA_EGLFS_DISPLAY=1
```

<= 詳細は不明ですが、グラフィックバッファの制御 と 推測

```
#
```

***** 【オリジナル】 *****

無し

***** 【修正ファイル】 *****

```
mount -t ext3 /dev/mmcblk0p2 /mnt <=アプリケーション起動前にSDカードをマウント と 推測
/etc/config/auto-run-app.sh & <=ユーザーアプリの起動 と 推測
```

(04) 対象ソースファイル : atmark-dist-20160927¥vendors¥AtmarkTechno¥Armadillo-840¥etc¥init.d¥pvrsrv

ご質問 : 以下のような修正が、オリジナルのOS (Atmark-Dist) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

```
**** [オリジナル] ****
echo -n "Starting PVR Server: "
pvrsrvctl --start --no-module >/dev/null 2>&1
check_status

**** [修正ファイル] ****
case "$1" in
  "stop")
    OP="--stop"
    RECOVERY_OP="--start"
    echo -n "Stopping "
    ;;
  *) # "start"
    OP="--start"
    RECOVERY_OP="--stop"
    echo -n "Starting "
    ;;
esac

echo -n "PVR Server: "

if [ -e /lib/modules/3.4-at1+/extra/bc_example.ko ]; then
  MODULE=""
  [ $# -ne 0 ] && echo -n " module: "
else
  MODULE="--no-module"
  [ $# -ne 0 ] && echo -n ": "
fi

do_pvrsrvctl ()
{
  local COUNT=0

  while [ 1 ]
  do
    COUNT=$(expr $COUNT + 1)

    pvrsrvctl $OP $MODULE > /dev/null 2>&1 && break
    echo -n "<"
    sleep 1

    # sometimes, succeeded to stop/start after try start/stop
    pvrsrvctl $RECOVERY_OP $MODULE > /dev/null 2>&1
    echo -n ">"
    sleep 1
  done
}

do_pvrsrvctl
check_status
```

(05) 対象ソースファイル: linux-3.4-at20¥arch¥arm¥configs¥armadillo840_defconfig

ご質問: 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての

修正であると推測されるでしょうか?

**** [オリジナル] ****

```
CONFIG_MACH_ARMADILLO840=y
# CONFIG_SH_TIMER_TMU is not set
```

**** [修正ファイル] ****

```
CONFIG_MACH_ARMADILLO840=y
CONFIG_ARMADILLO840_PRIMARY_FB_LCDC0=y
# CONFIG_SH_TIMER_TMU is not set
```

<= LCDの何を設定しているのでしょうか?

(06) 対象ソースファイル : linux-3.4-at20¥arch¥arm¥mach-shmobile¥board-armadillo840.c

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

**** 【オリジナル】 ****

```
#include <linux/irq.h>

#include <linux/mmc/host.h>
```

**** 【修正ファイル】 ****

```
#include <linux/irq.h>

#include <linux/leds_pwm.h>    <=    LCD の発信 PWM を修正する理由は何でしょうか？

#include <linux/mmc/host.h>
```

**** 【オリジナル】 ****

```
.tmio_flags    = TMIO_MMC_HAS_IDLE_WAIT,
.tmio_caps     = MMC_CAP_SD_HIGHSPEED | MMC_CAP_SDIO_IRQ,
.tmio_ocr_mask = MMC_VDD_165_195 | MMC_VDD_32_33 | MMC_VDD_33_34,
```

**** 【修正ファイル】 ****

```
.tmio_flags    = TMIO_MMC_HAS_IDLE_WAIT,
.tmio_caps     = MMC_CAP_SD_HIGHSPEED | MMC_CAP_SDIO_IRQ | MMC_CAP_NEEDS_POLL ,
.tmio_ocr_mask = MMC_VDD_165_195 | MMC_VDD_32_33 | MMC_VDD_33_34,
```

**** 【オリジナル】 ****

```
static struct rmob_tpu_pwm_platform_data pwm_device_data = {
    .channels[2] = {
```

**** 【修正ファイル】 ****

```
static struct rmob_tpu_pwm_platform_data pwm_device_data = {
```

```
#if IS_ENABLED(CONFIG_ARMADILLO840_TPU0T00_PORT23)
```

```
.channels[0] = {
    .polarity = 0,
    .pin_gpio = GPIO_PORT23,
    .pin_gpio_fn = GPIO_FN_TPU0T00,
},
```

```
#endif
```

```
#if IS_ENABLED(CONFIG_ARMADILLO840_TPU0T01_PORT21)
```

```
.channels[1] = {
    .polarity = 0,
    .pin_gpio = GPIO_PORT21,
    .pin_gpio_fn = GPIO_FN_TPU0T01,
},
```

```
#endif
```

```
#if IS_ENABLED(CONFIG_ARMADILLO840_TPU0T02_PORT202)
```

```
.channels[2] = {
```

```

**** [オリジナル] ****
        .pin_gpio_fn = GPIO_FN_TPU0T02_PORT202,
    }
};

**** [修正ファイル] ****
        .pin_gpio_fn = GPIO_FN_TPU0T02_PORT202,
    },
#endif

#if IS_ENABLED(CONFIG_ARMADILLO840_TPU0T03_PORT180)
    .channels[3] = {
        .polarity = 0,
        .pin_gpio = GPIO_PORT180,
        .pin_gpio_fn = GPIO_FN_TPU0T03,
    },
#endif
};

**** [オリジナル] ****
static struct platform_pwm_backlight_data pwm_backlight_data = {
    .pwm_id = RMOB_TPU_PWM_ID(0, 2),
    .lth_brightness = 50,
    .max_brightness = 255,
    .dft_brightness = 255,
    .pwm_period_ns = 33333, /* 30kHz */
};

**** [修正ファイル] ****
static struct led_pwm pwm_leds[] = {
#if IS_ENABLED(CONFIG_LEDS_PWM)
#if IS_ENABLED(CONFIG_ARMADILLO840_TPU0T00_PORT23)
    {
        .name = "TPU0T00.LED",
        .pwm_id = RMOB_TPU_PWM_ID(0, 0),
        .max_brightness = 255,
        .pwm_period_ns = 26316, /* 38kHz 1000000 / 38 = 26315.78947368421 */
    },
#endif
#endif
};

#if IS_ENABLED(CONFIG_ARMADILLO840_TPU0T01_PORT21)
    {
        .name = "TPU0T01.LED",
        .pwm_id = RMOB_TPU_PWM_ID(0, 1),
        .max_brightness = 255,
        .pwm_period_ns = 3703703, /* 270Hz 1000000000 / 270 = 3703703.703703704 */
    },
#endif
};

```

```

#endif
#if IS_ENABLED(CONFIG_ARMADILLO840_TPU0T02_PORT202)
    {
        .name = "TPU0T02.LED",
        .pwm_id = RMOB_TPU_PWM_ID(0, 2),
        .max_brightness = 255,
        .pwm_period_ns = 3703703, /* 270Hz 1000000000 / 270 = 3703703.703703704 */
        //.pwm_period_ns = 26316, /* 38kHz 1000000 / 38 = 26315.78947368421 */
    },
#endif
#if IS_ENABLED(CONFIG_ARMADILLO840_TPU0T03_PORT180)
    {
        .name = "TPU0T03.LED",
        .pwm_id = RMOB_TPU_PWM_ID(0, 3),
        .max_brightness = 255,
        .pwm_period_ns = 2000, /* 500kHz */
    },
#endif
#endif /* CONFIG_LEDS_PWM */
};

```

**** [オリジナル] ****

```

static struct platform_device pwm_backlight_device = {
    .name = "pwm-backlight",
    .dev = {
        .platform_data = &pwm_backlight_data,
    },
};

```

**** [修正ファイル] ****

```

static struct led_pwm_platform_data pwm_led_data = {
    .num_leds = ARRAY_SIZE(pwm_leds),
    .leds = pwm_leds,
};

```

**** [オリジナル] ****

/* LCDC0 */

**** [修正ファイル] ****

```

static struct platform_device pwm_led_device = {
    .name = "leds_pwm",
    .dev = {
        .platform_data = &pwm_led_data,
    },
};

```

```
};

static struct platform_pwm_backlight_data pwm_backlight_data = {
    .pwm_id = RMOB_TPU_PWM_ID(0, 2),
    .lth_brightness = 50,
    .max_brightness = 255,
    .dft_brightness = 255,
    .pwm_period_ns = 33333, /* 30kHz */
};
```

```
static struct platform_device pwm_backlight_device = {
    .name = "pwm-backlight",
    .dev = {
        .platform_data = &pwm_backlight_data,
    },
};
```

```
/* LCDC0 */
```

```
**** 【オリジナル】 ****
    .flag = FB_MODE_IS_DETAILED,
#endif
**** 【修正ファイル】 ****
    .flag = FB_MODE_IS_DETAILED,
#elif IS_ENABLED(CONFIG_ARMADILLO840_LCDC0_DEVICE_TCG070)
    .name = "KYOCERA/TCG070",
    .xres = 800,
    .yres = 480,
    .pixclock = 30100,
    .left_margin = 40, // 88,
    .right_margin = 88, // 40,
    .hsync_len = 128,
    .upper_margin = 32,
    .lower_margin = 11,
    .vsync_len = 2,
    //.refresh = 69,
    .sync = 0,
    .vmode = FB_VMODE_NONINTERLACED,
    .flag = FB_MODE_IS_DETAILED,
#endif
```

```

**** [オリジナル] ****
    .device_type = LCDC_DEV_LCD,
    .interface_type = RGB24,
};

**** [修正ファイル] ****
    .device_type = LCDC_DEV_LCD,
    .interface_type = RGB18,
};

**** [オリジナル] ****
    .chan = LCDC_CHAN_MAINLCD,
    .fourcc = V4L2_PIX_FMT_BGR32,
    .interface_type = RGB24,
    .clock_divider = 5,
    .flags = 0,

**** [修正ファイル] ****
    .chan = LCDC_CHAN_MAINLCD,
    .fourcc = V4L2_PIX_FMT_RGB565,
    .interface_type = RGB18,
    .clock_divider = 6,
    .flags = 0,

**** [オリジナル] ****
},
};

**** [修正ファイル] ****
},

#if 1
    .overlays[0] = {
        .fourcc = V4L2_PIX_FMT_BGR32,
        .pos_x = 0,
        .pos_y = 0,
        .xres = 800,
        .yres = 480,
        .max_xres = 800,
        .max_yres = 480,
    },
    .overlays[1] = {
        .fourcc = V4L2_PIX_FMT_BGR32,
        .pos_x = 0,
        .pos_y = 0,
        .xres = 800,
        .yres = 480,
        .max_xres = 800,

```

```

        .max_yres      = 480,
    },
#endif
};

**** [オリジナル] ****
/* pwm for backlight */
if (IS_ENABLED(CONFIG_ARMADILL0840_TPU0T02_PORT202)) {
    gpio_reserve(GPIO_PORT202);
**** [修正ファイル] ****
/* pwm for backlight */
if (IS_ENABLED(CONFIG_ARMADILL0840_TPU0T02_PORT202))
    gpio_reserve(GPIO_PORT202);

**** [オリジナル] ****
    platform_device_register(&pwm_device);
}
/* backlight */
**** [修正ファイル] ****
if (IS_ENABLED(CONFIG_ARMADILL0840_TPU0T00_PORT203))
    gpio_reserve(GPIO_PORT203);

if (IS_ENABLED(CONFIG_ARMADILL0840_TPU0T01_PORT201))
    gpio_reserve(GPIO_PORT201);

if (IS_ENABLED(CONFIG_ARMADILL0840_TPU0T03_PORT180))
    gpio_reserve(GPIO_PORT180);

    platform_device_register(&pwm_device);
/* backlight */

**** [オリジナル] ****
    platform_device_register(&pwm_backlight_device);
}
**** [修正ファイル] ****
    platform_device_register(&pwm_backlight_device);

if (IS_ENABLED(CONFIG_LEDS_PWM))
    platform_device_register(&pwm_led_device);
}

**** [オリジナル] ****
static struct platform_device lcdc1_device = {
    .name      = "sh_mobile_lcdc_fb",

```

```

**** [修正ファイル] ****
static struct platform_device __maybe_unused lcdc1_device = {
    .name          = "sh_mobile_lcdcb_fb",

**** [オリジナル] ****
static struct gpio hdmi_gpios[] = {
    { GPIO_FN_HDMI_HPD,          0,          "HPD" },

**** [修正ファイル] ****
static struct gpio __maybe_unused hdmi_gpios[] = {
    { GPIO_FN_HDMI_HPD,          0,          "HPD" },

**** [オリジナル] ****
{
    lcdc1_info.ch[0].caps = LCDC_CAP_HDMI;

**** [修正ファイル] ****
{
#if 0
    lcdc1_info.ch[0].caps = LCDC_CAP_HDMI;

**** [オリジナル] ****
platform_device_register(&lcdc1_device);
}

**** [修正ファイル] ****
platform_device_register(&lcdc1_device);
#endif
}

**** [オリジナル] ****
#if IS_ENABLED(CONFIG_ARMADILLO840_GPIO_KEY_POWER_PORT97)
    GPIO_KEY(KEY_POWER,          GPIO_PORT97,          "SW1", .wakeup = 1),
#endif

**** [修正ファイル] ****
#if IS_ENABLED(CONFIG_ARMADILLO840_GPIO_KEY_POWER_PORT97)
    GPIO_KEY(KEY_1, GPIO_PORT97,          "SW1", .wakeup = 1),
#endif

**** [オリジナル] ****
#if IS_ENABLED(CONFIG_ARMADILLO840_GPIO_KEY_BACK_PORT98)
    GPIO_KEY(KEY_BACK,          GPIO_PORT98,          "SW2"),
#endif

**** [修正ファイル] ****
#if IS_ENABLED(CONFIG_ARMADILLO840_GPIO_KEY_BACK_PORT98)
    GPIO_KEY(KEY_2, GPIO_PORT98,          "SW2"),
#endif

**** [オリジナル] ****

```

```

#if IS_ENABLED(CONFIG_ARMADILLO840_GPIO_KEY_MENU_PORT99)
    GPIO_KEY(KEY_MENU, GPIO_PORT99, "SW3"),
#endif

***** 【修正ファイル】 *****
#if IS_ENABLED(CONFIG_ARMADILLO840_GPIO_KEY_MENU_PORT99)
    GPIO_KEY(KEY_3, GPIO_PORT99, "SW3"),
#endif

***** 【オリジナル】 *****
#if IS_ENABLED(CONFIG_ARMADILLO840_GPIO_KEY_HOME_PORT100)
    GPIO_KEY(KEY_HOME, GPIO_PORT100, "SW4"),
#endif

***** 【修正ファイル】 *****
#if IS_ENABLED(CONFIG_ARMADILLO840_GPIO_KEY_HOME_PORT100)
    GPIO_KEY(KEY_4, GPIO_PORT100, "SW4"),
#endif

***** 【オリジナル】 *****
static void __init a840_clock_init(void)

***** 【修正ファイル】 *****
#define HSPRARCR12 0xffa00038

static void a840_hsprarc12_init(void)
{
    void __iomem *reg;

    reg = ioremap_nocache(HSPRARCR12, 4);
    if (reg) {
        /* increase acnum of ICB bus */
        iowrite32(0x00000001, reg);
        iounmap(reg);
    }
}

static void __init a840_clock_init(void)

***** 【オリジナル】 *****
r8a7740_sgx540_workaround();

***** 【修正ファイル】 *****
r8a7740_sgx540_workaround();
a840_hsprarc12_init();

```

(07) 対象ソースファイル : linux-3.4-at20¥arch¥arm¥mach-shmobile¥Kconfig.armadillo800

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか?

```
**** [オリジナル] ****
endchoice

**** [修正ファイル] ****
config ARMADILLO840_LCDCO_DEVICE_TCG070
    bool "LCD: TCG070"
endchoice
```

```
**** [オリジナル] ****
default n

depends on !(ARMADILLO840_SCIFA4_12_13)
depends on !(ARMADILLO840_CON8EB_KBCRICO1VG)
```

```
**** [修正ファイル] ****
default n

depends on !(ARMADILLO840_CON8EB_KBCRICO1VG)
```

```
**** [オリジナル] ****
config ARMADILLO840_TPUOT02_PORT202
```

```
**** [修正ファイル] ****
config ARMADILLO840_TPUOT00_PORT23
    bool "use TPU0 Output0 [PORT23]"
    default n
    depends on !(ARMADILLO840_CEU1)
    depends on !(ARMADILLO840_SCIFA1)
```

```
config ARMADILLO840_TPUOT01_PORT21
    bool "use TPU0 Output1 [PORT21]"
    default n
    depends on !(ARMADILLO840_CEU1)
    depends on !(ARMADILLO840_SCIFA1)
```

```
config ARMADILLO840_TPUOT02_PORT202
```

```
**** [オリジナル] ****
choice
```

```
**** [修正ファイル] ****
config ARMADILLO840_TPUOT03_PORT180
    bool "use TPU0 Output3 [PORT180]"
    default n
    depends on !(ARMADILLO840_CEU1)
choice
```

(08) 対象ソースファイル : linux-3.4-at20¥drivers¥clocksource¥sh_cmt_simple.c

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

```
**** [オリジナル] ****
static void sh_cmt_set_next(struct sh_cmt_priv *p, unsigned long delta)
{
**** [修正ファイル] ****
static int sh_cmt_set_next(struct sh_cmt_priv *p, unsigned long delta)
{
**** [オリジナル] ****
    unsigned long flags;
**** [修正ファイル] ****
    unsigned long flags;

    int ret;

**** [オリジナル] ****
    sh_cmt_write(p, CMCOR, sh_cmt_read(p, CMCNT) + delta);
    spin_unlock_irqrestore(&p->lock, flags);
}
**** [修正ファイル] ****
    sh_cmt_write(p, CMCOR, sh_cmt_read(p, CMCNT) + delta);
    ret = (sh_cmt_read(p, CMCNT) > sh_cmt_read(p, CMCOR)) ? -ETIME : 0;
    spin_unlock_irqrestore(&p->lock, flags);

    return ret;
}

**** [オリジナル] ****
    sh_cmt_set_next(p, delta);

    return 0;
}
**** [修正ファイル] ****
    return sh_cmt_set_next(p, delta);
}
```

(09) 対象ソースファイル : linux-3.4-at20¥drivers¥gpu¥eurasia_k¥platform-r8a7740¥config_kernel.h

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

**** [オリジナル] ****
config_kernel.h

**** [修正ファイル] ****
#define PVR_LDM_PLATFORM_PRE_REGISTERED
#define SYS_SGX_DEV_NAME "pvrsrvkm"

(10) 対象ソースファイル : linux-3.4-at20¥drivers¥gpu¥eurasia_km¥services4¥srvkm¥devices¥sgx¥sgxinit.c

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

**** [オリジナル] ****
sgxinit.c

```
PVR_LOG(("HWRecoveryResetSGX: SGX Hardware Recovery triggered"));
```

**** [修正ファイル] ****
SGXINIT.C

```
PVR_LOG(("HWRecoveryResetSGX: SGX Hardware Recovery triggered"));  
dump_stack();
```

**** [オリジナル] ****
sgxinit.c

```
    ui32LockupCounter++;  
    if (ui32LockupCounter == 3)  
    {
```

**** [修正ファイル] ****
SGXINIT.C

```
    ui32LockupCounter++;  
    if (ui32LockupCounter == 30)  
    {
```

(11) 対象ソースファイル : linux-3.4-at20¥drivers¥gpu¥eurasia_km¥services4¥srvkm¥devices¥sgx¥sgxpower.c

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

**** [オリジナル] ****
sgxpower.c

```
                psSGXTimingInfo->ui32uKernelFreq / psSGXTimingInfo->ui32HWRecoveryFreq;  
            }
```

**** [修正ファイル] ****
SGXPOWER.C

```
                psSGXTimingInfo->ui32uKernelFreq / psSGXTimingInfo->ui32HWRecoveryFreq;  
            /* psDevInfo->psSGXHostCtl->ui32HWRecoverySampleRate *= 10; */  
            /* psDevInfo->psSGXHostCtl->ui32HWRecoverySampleRate /= 10; */  
            }
```

(12) 対象ソースファイル: linux-3.4-at20¥drivers¥spi¥spi-bitbang.c

ご質問: 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか?

オリジナルソースからの任意抜粋

**** [オリジナル] ****

```
dev_dbg(&spi->dev, "%s, %u nsec/bit¥n", __func__, 2 * cs->nsecs);
```

**** [修正ファイル] ****

```
//dev_dbg(&spi->dev, "%s, %u nsec/bit¥n", __func__, 2 * cs->nsecs);
```

(13) 対象ソースファイル : linux-3.4-at20¥drivers¥spi¥spi.c

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

```
***** [オリジナル] *****
dev_dbg(&spi->dev, "setup mode %d, %s%s%s%s"
        "%u bits/w, %u Hz max --> %d¥n",
***** [修正ファイル] *****
/*dev_dbg(&spi->dev, "setup mode %d, %s%s%s%s"
        "%u bits/w, %u Hz max --> %d¥n",
***** [オリジナル] *****
        status);
***** [修正ファイル] *****
        status);
*/
```

(14) 対象ソースファイル: linux-3.4-at20¥drivers¥spi¥spidev.c

ご質問: 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか?

```
**** [オリジナル] ****
        spi->mode = save;
        else
        dev_dbg(&spi->dev, "spi mode %02x¥n", tmp);
    }
**** [修正ファイル] ****
        spi->mode = save;
        //else
        //dev_dbg(&spi->dev, "spi mode %02x¥n", tmp);
    }
```

(15) 対象ソースファイル : linux-3.4-at20¥drivers¥video¥sh_mobile_lcdcfb.c

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

FG(ファイルコンペアコマンド) にて、以下のメッセージが出るほど、ファイルの内容が大きく異なります。
「再同期に失敗しました。ファイルが著しく異なります。」

```
***** 【オリジナル】 *****
#include <linux/console.h>
#include <linux/dma-mapping.h>

***** 【修正ファイル】 *****
#include <linux/console.h>
#include <linux/ctype.h>
#include <linux/dma-mapping.h>

***** 【オリジナル】 *****
#define SIDE_B_OFFSET 0x1000
#define MIRROR_OFFSET 0x2000

#define MAX_XRES 1920
#define MAX_YRES 1080

struct sh_mobile_lcdc_priv {
    void __iomem *base;
    int irq;
    atomic_t hw_usecnt;
    struct device *dev;
    struct clk *dot_clk;
    unsigned long lddckr;
    struct sh_mobile_lcdc_chan ch[2];
    struct notifier_block notifier;
    int started;
    int forced_fourcc; /* 2 channel LCDC must share fourcc setting */
    struct sh_mobile_meram_info *meram_dev;
};

/* -----
 * Registers access
 */

***** 【修正ファイル】 *****
/* -----
 * Overlay register definitions
 */
```

**** 【オリジナル】 ****

```
static unsigned long lcdc_offs_mainlcd[NR_CH_REGS] = {  
    [LDDCKPAT1R] = 0x400,  
    [LDDCKPAT2R] = 0x404,  
    [LDMT1R] = 0x418,  
    [LDMT2R] = 0x41c,  
    [LDMT3R] = 0x420,  
    [LDDFR] = 0x424,  
    [LDSM1R] = 0x428,  
    [LDSM2R] = 0x42c,  
    [LDSA1R] = 0x430,  
    [LDSA2R] = 0x434,  
    [LDMLSR] = 0x438,  
    [LDHGNR] = 0x448,  
    [LDHSYNR] = 0x44c,  
    [LDVLNR] = 0x450,  
    [LDVSYNR] = 0x454,  
    [LDPMR] = 0x460,  
    [LDHAJR] = 0x4a0,  
};
```

**** 【修正ファイル】 ****

```
#define LDBCR 0xb00  
#define LDBCR_UPC(n) (1 << ((n) + 16))  
#define LDBCR_UPF(n) (1 << ((n) + 8))  
#define LDBCR_UPD(n) (1 << ((n) + 0))  
#define LDBnBSIFR(n) (0xb20 + (n) * 0x20 + 0x00)  
#define LDBBSIFR_EN (1 << 31)  
#define LDBBSIFR_VS (1 << 29)  
#define LDBBSIFR_BRSEL (1 << 28)  
#define LDBBSIFR_MX (1 << 27)  
#define LDBBSIFR_MY (1 << 26)  
#define LDBBSIFR_CV3 (3 << 24)  
#define LDBBSIFR_CV2 (2 << 24)  
#define LDBBSIFR_CV1 (1 << 24)  
#define LDBBSIFR_CV0 (0 << 24)  
#define LDBBSIFR_CV_MASK (3 << 24)  
#define LDBBSIFR_LAY_MASK (0xff << 16)  
#define LDBBSIFR_LAY_SHIFT 16  
#define LDBBSIFR_ROP3_MASK (0xff << 16)  
#define LDBBSIFR_ROP3_SHIFT 16  
#define LDBBSIFR_AL_PL8 (3 << 14)  
#define LDBBSIFR_AL_PL1 (2 << 14)
```

```

#define LDBBSIFR_AL_PK          (1 << 14)
#define LDBBSIFR_AL_1          (0 << 14)
#define LDBBSIFR_AL_MASK      (3 << 14)
#define LDBBSIFR_SWPL         (1 << 10)
#define LDBBSIFR_SWPW         (1 << 9)
#define LDBBSIFR_SWPB         (1 << 8)
#define LDBBSIFR_RY           (1 << 7)
#define LDBBSIFR_CHRR_420     (2 << 0)
#define LDBBSIFR_CHRR_422     (1 << 0)
#define LDBBSIFR_CHRR_444     (0 << 0)
#define LDBBSIFR_RPKF_ARGB32  (0x00 << 0)
#define LDBBSIFR_RPKF_RGB16   (0x03 << 0)
#define LDBBSIFR_RPKF_RGB24   (0x0b << 0)
#define LDBBSIFR_RPKF_MASK    (0x1f << 0)
#define LDBnBSSZR(n)          (0xb20 + (n) * 0x20 + 0x04)
#define LDBBSSZR_BVSS_MASK    (0xffff << 16)
#define LDBBSSZR_BVSS_SHIFT   16
#define LDBBSSZR_BHSS_MASK    (0xffff << 0)
#define LDBBSSZR_BHSS_SHIFT   0
#define LDBnBLOCR(n)          (0xb20 + (n) * 0x20 + 0x08)
#define LDBBLOCR_CVLC_MASK    (0xffff << 16)
#define LDBBLOCR_CVLC_SHIFT   16
#define LDBBLOCR_CHLC_MASK    (0xffff << 0)
#define LDBBLOCR_CHLC_SHIFT   0
#define LDBnBSMWR(n)          (0xb20 + (n) * 0x20 + 0x0c)
#define LDBBSMWR_BSMWA_MASK   (0xffff << 16)
#define LDBBSMWR_BSMWA_SHIFT  16
#define LDBBSMWR_BSMW_MASK    (0xffff << 0)
#define LDBBSMWR_BSMW_SHIFT   0
#define LDBnBSAYR(n)          (0xb20 + (n) * 0x20 + 0x10)
#define LDBBSAYR_FG1A_MASK    (0xff << 24)
#define LDBBSAYR_FG1A_SHIFT   24
#define LDBBSAYR_FG1R_MASK    (0xff << 16)
#define LDBBSAYR_FG1R_SHIFT   16
#define LDBBSAYR_FG1G_MASK    (0xff << 8)
#define LDBBSAYR_FG1G_SHIFT   8
#define LDBBSAYR_FG1B_MASK    (0xff << 0)
#define LDBBSAYR_FG1B_SHIFT   0
#define LDBnBSACR(n)          (0xb20 + (n) * 0x20 + 0x14)
#define LDBBSACR_FG2A_MASK    (0xff << 24)
#define LDBBSACR_FG2A_SHIFT   24

```

```

#define LDBBSACR_FG2R_MASK      (0xff << 16)
#define LDBBSACR_FG2R_SHIFT    16
#define LDBBSACR_FG2G_MASK      (0xff << 8)
#define LDBBSACR_FG2G_SHIFT    8
#define LDBBSACR_FG2B_MASK      (0xff << 0)
#define LDBBSACR_FG2B_SHIFT    0
#define LDBnBSAAR(n)           (0xb20 + (n) * 0x20 + 0x18)
#define LDBBSAAR_AP_MASK        (0xff << 24)
#define LDBBSAAR_AP_SHIFT      24
#define LDBBSAAR_R_MASK         (0xff << 16)
#define LDBBSAAR_R_SHIFT       16
#define LDBBSAAR_GY_MASK        (0xff << 8)
#define LDBBSAAR_GY_SHIFT      8
#define LDBBSAAR_B_MASK         (0xff << 0)
#define LDBBSAAR_B_SHIFT       0
#define LDBnBPPCR(n)           (0xb20 + (n) * 0x20 + 0x1c)
#define LDBBPPCR_AP_MASK        (0xff << 24)
#define LDBBPPCR_AP_SHIFT      24
#define LDBBPPCR_R_MASK         (0xff << 16)
#define LDBBPPCR_R_SHIFT       16
#define LDBBPPCR_GY_MASK        (0xff << 8)
#define LDBBPPCR_GY_SHIFT      8
#define LDBBPPCR_B_MASK         (0xff << 0)
#define LDBBPPCR_B_SHIFT       0
#define LDBnBBGCL(n)           (0xb10 + (n) * 0x04)
#define LDBBBGCL_BGA_MASK       (0xff << 24)
#define LDBBBGCL_BGA_SHIFT     24
#define LDBBBGCL_BGR_MASK       (0xff << 16)
#define LDBBBGCL_BGR_SHIFT     16
#define LDBBBGCL_BGG_MASK       (0xff << 8)
#define LDBBBGCL_BGG_SHIFT     8
#define LDBBBGCL_BGB_MASK       (0xff << 0)
#define LDBBBGCL_BGB_SHIFT     0

```

```
#define SIDE_B_OFFSET 0x1000
```

```
#define MIRROR_OFFSET 0x2000
```

**** **【オリジナル】** ****

```
static unsigned long lcdc_offs_sublcd[NR_CH_REGS] = {
    [LDDCKPAT1R] = 0x408,
    [LDDCKPAT2R] = 0x40c,
```

```

[LDMT1R] = 0x600,
[LDMT2R] = 0x604,
[LDMT3R] = 0x608,
[LDDFR] = 0x60c,
[LDSM1R] = 0x610,
[LDSM2R] = 0x614,
[LDSA1R] = 0x618,
[LDMLSR] = 0x620,
[LDHCNR] = 0x624,
[LDHSYNR] = 0x628,
[LDVLR] = 0x62c,
[LDVSYNR] = 0x630,
[LDPMR] = 0x63c,
};

```

****** 【修正ファイル】 ******

```
#define MAX_XRES 1920
```

```
#define MAX_YRES 1080
```

```

enum sh_mobile_lcdc_overlay_mode {
    LGDC_OVERLAY_BLEND,
    LGDC_OVERLAY_ROP3,
};

```

****** 【オリジナル】 ******

```
static bool banked(int reg_nr)
```

```

{
    switch (reg_nr) {
        case LDMT1R:
        case LDMT2R:
        case LDMT3R:
        case LDDFR:
        case LDSM1R:
        case LDSA1R:
        case LDSA2R:
        case LDMLSR:
        case LDHCNR:
        case LDHSYNR:
        case LDVLR:
        case LDVSYNR:
        case LDHAJR:
            return true;
    }
}

```

```

        return false;
    }

static int lcdc_chan_is_sublcd(struct sh_mobile_lcdc_chan *chan)
{
    return chan->cfg->chan == LCDC_CHAN_SUBLCD;
}

static void lcdc_write_chan(struct sh_mobile_lcdc_chan *chan,
                           int reg_nr, unsigned long data)
{
    iowrite32(data, chan->lcdc->base + chan->reg_offs[reg_nr]);
    if (banked(reg_nr))
        iowrite32(data, chan->lcdc->base + chan->reg_offs[reg_nr] +
                  SIDE_B_OFFSET);
}

static void lcdc_write_chan_a(struct sh_mobile_lcdc_chan *chan,
                              int reg_nr, unsigned long data)
{
    iowrite32(data, chan->lcdc->base + chan->reg_offs[reg_nr]);
}

static void lcdc_write_chan_b(struct sh_mobile_lcdc_chan *chan,
                              int reg_nr, unsigned long data)
{
    iowrite32(data, chan->lcdc->base + chan->reg_offs[reg_nr] +
              SIDE_B_OFFSET);
}

static unsigned long lcdc_read_chan(struct sh_mobile_lcdc_chan *chan,
                                    int reg_nr)
{
    return ioread32(chan->lcdc->base + chan->reg_offs[reg_nr]);
}

static void lcdc_write(struct sh_mobile_lcdc_priv *priv,
                      unsigned long reg_offs, unsigned long data)
{
    iowrite32(data, priv->base + reg_offs);
}

```

```

static unsigned long lcdc_read(struct sh_mobile_lcdc_priv *priv,
                             unsigned long reg_offs)
{
    return ioread32(priv->base + reg_offs);
}

static void lcdc_wait_bit(struct sh_mobile_lcdc_priv *priv,
                         unsigned long reg_offs,
                         unsigned long mask, unsigned long until)
{
    while ((lcdc_read(priv, reg_offs) & mask) != until)
        cpu_relax();
}

static const struct sh_videomode_hint *sh_get_videomode_hint(struct fb_info *info)
{
    struct sh_mobile_lcdc_chan *ch = info->par;
    const struct sh_videomode_hint *hints = ch->lcd_mode_hints;
    struct fb_var_screeninfo *var;
    struct fb_videomode mode;
    const struct fb_videomode *modedb;
    unsigned int modedb_len;
    unsigned int i;

    if (!hints)
        return NULL;

    var = &info->var;
    modedb = ch->cfg->lcd_modes;
    modedb_len = ch->cfg->num_modes;

    fb_var_to_videomode(&mode, var);

    for (i = 0; i < modedb_len; i++) {
        if (fb_mode_is_equal(&mode, &modedb[i]))
            return &hints[i];
    }

    return NULL;
}

```

**** 【修正ファイル】 ****

```

/*
 * struct sh_mobile_lcdc_overlay - LCDC display overlay
 *
 * @channel: LCDC channel this overlay belongs to
 * @cfg: Overlay configuration
 * @info: Frame buffer device
 * @index: Overlay index (0-3)
 * @base: Overlay registers base address
 * @enabled: True if the overlay is enabled
 * @mode: Overlay blending mode (alpha blend or ROP3)
 * @alpha: Global alpha blending value (0-255, for alpha blending mode)
 * @rop3: Raster operation (for ROP3 mode)
 * @fb_mem: Frame buffer virtual memory address
 * @fb_size: Frame buffer size in bytes
 * @dma_handle: Frame buffer DMA address
 * @base_addr_y: Overlay base address (RGB or luma component)
 * @base_addr_c: Overlay base address (chroma component)
 * @pan_offset: Current pan offset in bytes
 * @format: Current pixel format
 * @xres: Horizontal visible resolution
 * @xres_virtual: Horizontal total resolution
 * @yres: Vertical visible resolution
 * @yres_virtual: Vertical total resolution
 * @pitch: Overlay line pitch
 * @pos_x: Horizontal overlay position
 * @pos_y: Vertical overlay position
 */
struct sh_mobile_lcdc_overlay {
    struct sh_mobile_lcdc_chan *channel;

    const struct sh_mobile_lcdc_overlay_cfg *cfg;
    struct fb_info *info;

    unsigned int index;
    unsigned long base;

    bool enabled;
    enum sh_mobile_lcdc_overlay_mode mode;
    unsigned int alpha;
    unsigned int rop3;
}

```

```

void *fb_mem;
unsigned long fb_size;

dma_addr_t dma_handle;
unsigned long base_addr_y;
unsigned long base_addr_c;
unsigned long pan_offset;

const struct sh_mobile_lcdc_format_info *format;
unsigned int xres;
unsigned int xres_virtual;
unsigned int yres;
unsigned int yres_virtual;
unsigned int pitch;
int pos_x;
int pos_y;

struct dmabuf_priv dmabuf_priv[SH_MOBILE_LCDC_SCREEN_NUM];
};

struct sh_mobile_lcdc_priv {
    void __iomem *base;
    int irq;
    atomic_t hw_usecnt;
    struct device *dev;
    struct clk *dot_clk;
    unsigned long lddckr;

    struct sh_mobile_lcdc_chan ch[2];
    struct sh_mobile_lcdc_overlay overlays[4];

    struct notifier_block notifier;
    int started;
    int forced_fourcc; /* 2 channel LCDC must share fourcc setting */
    struct sh_mobile_meram_info *meram_dev;
};

/* -----
* Registers access
*/

```

```
static unsigned long lcdc_offs_mainlcd[NR_CH_REGS] = {
    [LDDCKPAT1R] = 0x400,
    [LDDCKPAT2R] = 0x404,
    [LDMT1R] = 0x418,
    [LDMT2R] = 0x41c,
    [LDMT3R] = 0x420,
    [LDDFR] = 0x424,
    [LDSM1R] = 0x428,
    [LDSM2R] = 0x42c,
    [LDSA1R] = 0x430,
    [LDSA2R] = 0x434,
    [LDMLSR] = 0x438,
    [LDHCNR] = 0x448,
    [LDHSYNR] = 0x44c,
    [LDVLR] = 0x450,
    [LDVSYNR] = 0x454,
    [LDPMR] = 0x460,
```

(16) 対象ソースファイル : linux-3.4-at20¥include¥video¥sh_mobile_lcdc.h

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

```
**** [オリジナル] ****
struct sh_mobile_lcdc_chan_cfg {

**** [修正ファイル] ****
struct sh_mobile_lcdc_overlay_cfg {
    int fourcc;
    unsigned int pos_x;
    unsigned int pos_y;
    unsigned int xres;
    unsigned int yres;
    unsigned int max_xres;
    unsigned int max_yres;
};

struct sh_mobile_lcdc_chan_cfg {

**** [オリジナル] ****
    struct sh_mobile_lcdc_chan_cfg ch[2];
    struct sh_mobile_meram_info *meram_dev;

**** [修正ファイル] ****
    struct sh_mobile_lcdc_chan_cfg ch[2];
    struct sh_mobile_lcdc_overlay_cfg overlays[4];
    struct sh_mobile_meram_info *meram_dev;
```

(17) 対象ソースファイル : linux-3.4-at20¥sound¥soc¥codecs¥cs4270.c

ご質問 : 以下のような修正が、オリジナルのOS (linux) に加えられていますが、どのような効果を期待しての修正であると推測されるでしょうか？

```
**** [オリジナル] ****
int ret;

**** [修正ファイル] ****
int ret;

int count;

**** [オリジナル] ****
ret = i2c_smbus_read_byte_data(i2c_client, CS4270_CHIPID);
if (ret < 0) {
    dev_err(&i2c_client->dev, "failed to read i2c at addr %X¥n",
            i2c_client->addr);
    return ret;
}

/* The top four bits of the chip ID should be 1100. */
if ((ret & 0xF0) != 0xC0) {
    dev_err(&i2c_client->dev, "device at addr %X is not a CS4270¥n",
            i2c_client->addr);
    return -ENODEV;
}

**** [修正ファイル] ****
ret = i2c_smbus_read_byte_data(i2c_client, CS4270_CHIPID);

count = 0;
while(ret < 0 && count < 5) {
    dev_err(&i2c_client->dev, "failed to read i2c at addr %X and retry wait 1000msec¥n",
            i2c_client->addr);
    /* Read ReTry. */
    ret = i2c_smbus_read_byte_data(i2c_client, CS4270_CHIPID);
    mdelay(1000);
    count++;
}

**** [オリジナル] ****
dev_info(&i2c_client->dev, "found device at i2c address %X¥n",
        i2c_client->addr);
dev_info(&i2c_client->dev, "hardware revision %X¥n", ret & 0xF);

cs4270 = devm_kzalloc(&i2c_client->dev, sizeof(struct cs4270_private),
                    GFP_KERNEL);

if (!cs4270) {
```

```

        dev_err(&i2c_client->dev, "could not allocate codec¥n");
        return -ENOMEM;
    }
**** [修正ファイル] ****
    if (ret < 0) {
        dev_err(&i2c_client->dev, "failed to read i2c at addr %X¥n",
                i2c_client->addr);
        return ret;
    }

**** [オリジナル] ****
    i2c_set_clientdata(i2c_client, cs4270);
**** [修正ファイル] ****
    /* The top four bits of the chip ID should be 1100. */
    if ((ret & 0xF0) != 0xC0) {
        dev_err(&i2c_client->dev, "device at addr %X is not a CS4270¥n",
                i2c_client->addr);
        return -ENODEV;
    }

    dev_info(&i2c_client->dev, "found device at i2c address %X¥n",
            i2c_client->addr);
    dev_info(&i2c_client->dev, "hardware revision %X¥n", ret & 0xF);

    cs4270 = devm_kzalloc(&i2c_client->dev, sizeof(struct cs4270_private),
                        GFP_KERNEL);
    if (!cs4270) {
        dev_err(&i2c_client->dev, "could not allocate codec¥n");
        return -ENOMEM;
    }

    i2c_set_clientdata(i2c_client, cs4270);

```