

**Understand iMX8QX Hardware Partitioning  
By Making M4 Hello world Running Correctly  
(Hardware Partitioning Practitce)**

1/9/2020

Biyong Sun

# Preface

Suppose you have knowledges of i.MX8QXP building flash bin, scfw, uboot, linux Kernel and M4 program.

# Running M4 Hello\_world without any change

If you have tried running M4 hello\_world.bin on i.MX8QXP even with fsl-imx8qxp-mek-rpmsg.dtb, You will find M4 Hello\_world can run, but A35 side, uboot and linux can not run correctly.

But rpmsg\_lite\_str\_echo\_rtos\_imxcm4.bin, which has rpmsg on M4 side, has no problem.

## What makes this difference?

U-Boot 2018.03 (Jan 08 2020 - 17:46:55  
+0800)

CPU: Freescale i.MX8QXP revB A35 at 1200  
MHz at 23C

Model: Freescale i.MX8QXP MEK

Board: iMX8QXP MEK

Boot: SD1

DRAM: 2.8 GiB

**imx\_vservice\_connect: Timeout to  
connect peer, -110**

**VService: Connection is failed, ret -110**

**virt\_i2c: Faild to setup vservice**

Linux

[ 109.394251] imx-drm display-subsystem: master  
bind failed: -517

[ 109.423462] [drm] Supports vblank timestamp  
caching Rev 2 (21.10.2013).

[ 109.430139] [drm] No driver support for vblank  
timestamp query.

[ 109.436574] imx-drm display-subsystem: bound  
imx-drm-dpu-bliteng.2 (ops dpu\_bliteng\_ops)

# BSP design for Hardware partitioning

# M4 assigned to P3

imx-mkimage/iMX8QX/soc.mak:

flash\_regression\_linux\_m4

**-flags 0x00200000 -p3**

scfw/platform/board/mx8qx\_mek:

**sc\_err\_t board\_system\_config(sc\_bool\_t early, sc\_rm\_pt\_t pt\_boot)**

**if (alt\_config != SC\_FALSE)**

System Controller Firmware Porting Guide(sc\_fw\_port.pdf)

Flag	Bit	Meaning
SC_BD_FLAGS_NOT_SECURE	16	Initial boot partition is not secure
SC_BD_FLAGS_NOT_ISOLATED	17	Initial boot partition is not isolated
SC_BD_FLAGS_RESTRICTED	18	Initial boot partition is restricted
SC_BD_FLAGS_GRANT	19	Initial boot partition grants access to the SCFW
SC_BD_FLAGS_NOT_COHERENT	20	Initial boot partition is not coherent
SC_BD_FLAGS_ALT_CONFIG	21	Alternate SCFW config (passed to board.c)
SC_BD_FLAGS_EARLY_CPU_START	22	Start some CPUs early
SC_BD_FLAGS_DDRTEST	23	Config for DDR stress test
SC_BD_FLAGS_NO_AP	24	Don't boot AP even if requested by ROM

imx-mkimage/iMX8QX/soc.mak:

**-flags 0x00200000**

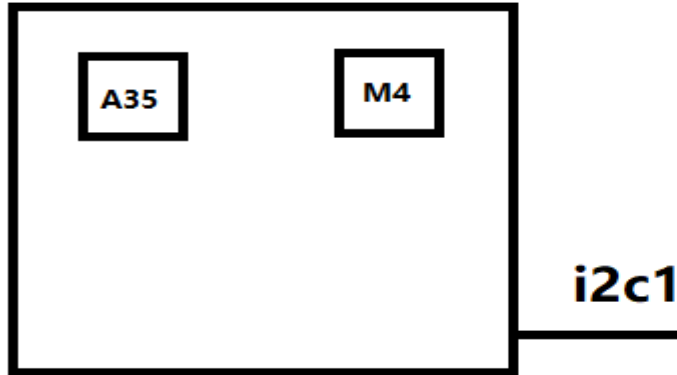
scfw/platform/board/mx8qx\_mek:

**sc\_err\_t board\_system\_config(sc\_bool\_t early, sc\_rm\_pt\_t pt\_boot)**

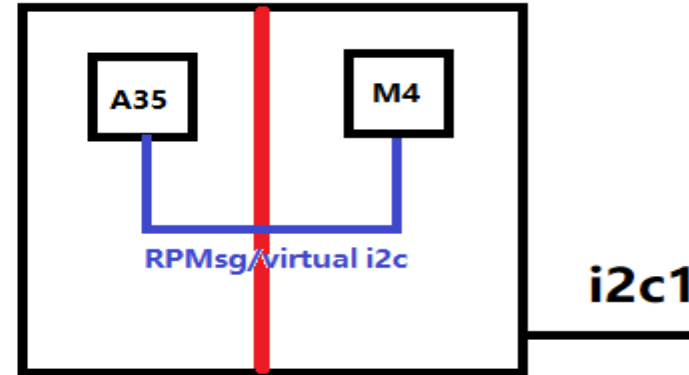
**if (alt\_config != SC\_FALSE)**

# Virtual i2c with Hardware Partiton Enabled

Without Hardware Partiton Enabled



With Hardware Partiton Enabled



With hardware partition enabled, the A35 will check the i2c1 owner and use virtual i2c.

```
u-boot/drivers/i2c/imx_virt_i2c.c:int __weak board_imx_virt_i2c_bind(struct udevice *dev)
```

```
u-boot/arch/arm/mach-imx/imx8/cpu.c:int board_imx_virt_i2c_bind(struct udevice *dev)
```

```
int board_imx_virt_i2c_bind(struct udevice *dev)
```

```
{
```

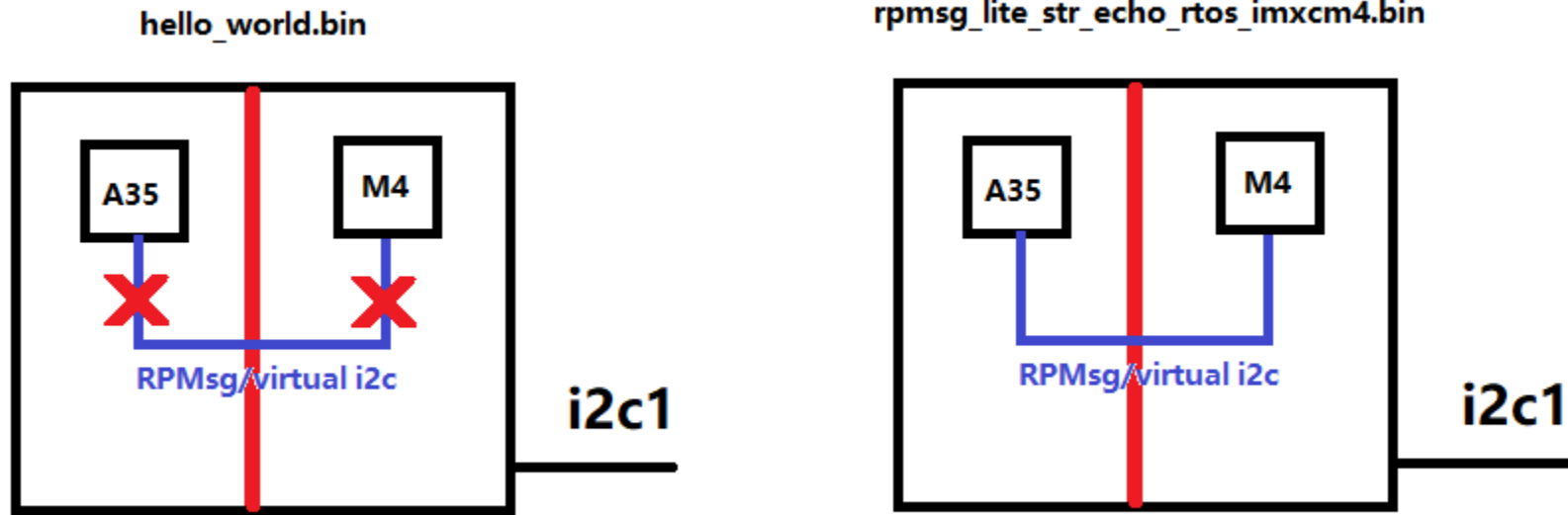
```
    if (check_owned_udevice(dev))
```

```
        return -ENODEV;
```

```
    return 0;
```

```
}
```

# hello\_world.bin VS rpmsg\_lite\_str\_echo\_rtos\_imxcm4.bin



Hello\_world\_bin has no rpmsg up for virtual i2c, that makes the error print out.

And A35 side linux also loses virtual i2c in `fsl-imx8qxp-mek-rpmsg.dts`

Uboot:

`imx_vservice_connect: Timeout to connect peer, -110`

`VService: Connection is failed, ret -110`

**Make hello\_world.bin running correctly**



# Solution 1: With NO Hardware Partition Enabled

No hardware partition still can run M4 program as i.MX6SOLO/i.MX8MM does.

```
imx-mkimage/iMX8QX/soc.mak
```

```
flash_regression_linux_m4
```

```
flash_regression_linux_m4: $(MKIMG) mx8qx-ahab-container.img scfw_tcm.bin u-boot-atf.bin m4_image.bin  
./$(MKIMG) -soc QX -rev B0 -append mx8qx-ahab-container.img -c -flags 0x00200000 -scfw scfw_tcm.bin -ap u-boot-atf.bin a35 0x80000000 -p3 -m4 m4_image.bin 0 0x34FE0000 -out flash.bin
```

## Change to

```
flash_regression_linux_m4: $(MKIMG) mx8qx-ahab-container.img scfw_tcm.bin u-boot-atf.bin m4_image.bin  
./$(MKIMG) -soc QX -rev B0 -append mx8qx-ahab-container.img -c -scfw scfw_tcm.bin -ap u-boot-atf.bin a35 0x80000000 -m4 m4_image.bin 0 0x34FE0000 -out flash.bin
```

**Remove `-flags 0x00200000`, `-p3` to make all resources in only one partition.**

# Solution 1: With NO Hardware Partition Enabled(Cont.)

If you need to run rpmsg like i.MX6SOLOX/i.MX8M add rpmsg node

```
&rpmsg{
    /*
     * 64K for one rpmsg instance:
     */
    vdev-nums = <2>;
    reg = <0x0 0x90000000 0x0 0x20000>;
    status = "okay";
};
```

## Solution 2: Move i2c with pads to A35 side

```
scfw/platform/board/mx8qx_mek/board.c
```

```
sc_err_t board_system_config(sc_bool_t early, sc_rm_pt_t pt_boot)
```

```
672     BRD_ERR(rm_set_resource_movable(pt_boot, SC_R_CAN_0,  
673         SC_R_CAN_2, SC_TRUE));  
674 //     BRD_ERR(rm_set_resource_movable(pt_boot, SC_R_I2C_1,  
675 //         SC_R_I2C_1, SC_TRUE));  
676     BRD_ERR(rm_set_resource_movable(pt_boot, SC_R_FSPI_0,  
677         SC_R_FSPI_0, SC_TRUE));  
  
680     BRD_ERR(rm_set_pad_movable(pt_boot, SC_P_FLEXCAN0_RX,  
681         SC_P_FLEXCAN2_TX, SC_TRUE));  
682 //     BRD_ERR(rm_set_pad_movable(pt_boot, SC_P_USB_SS3_TC1,  
683 //         SC_P_USB_SS3_TC1, SC_TRUE));  
684 //     BRD_ERR(rm_set_pad_movable(pt_boot, SC_P_USB_SS3_TC3,  
685 //         SC_P_USB_SS3_TC3, SC_TRUE));  
686     BRD_ERR(rm_set_pad_movable(pt_boot, SC_P_QSPI0A_DATA0,  
687         SC_P_COMP_CTL_GPIO_1V8_3V3_QSPI0B, SC_TRUE));
```

**Note: Just a practice for understanding hardware partitioning.  
i2c1 move to A35 will make M4 core no access to i2c1.**

## Solution 2: Move i2c to A35 side(Cont.)

Check the changes by scfw **Debug Monitor**.

System Controller Firmware Porting Guide(sc\_fw\_port.pdf)

md.w address [count]	display count words at address
md[.l] address [count]	display count long-words at address
mm.b address value	modify byte at address
mm.w address value	modify word at address
mm[.l] address value	modify long-word at address
ai.r ss sel addr	read analog interface (AI) data
ai.w ss sel addr data	write analog interface (AI) data
fuse.r word	read OTP fuse word
fuse.w word value	write value to OTP fuse word
dump rm	dump all the resource manager (RM) info
dump rm part [part]	dump all partition info for part (default = all)
dump rm rsrc [part]	dump all resource info for part (default = all)
dump rm mem [part]	dump all memory info for part (default = all)
dump rm pad [part]	dump all pad info for part (default = all)
power.r [resource]	read/get power mode of resource (default = all)
power.w resource mode	write/set power mode of resource to mode (off, stby, lp, on)
info	display SCFW/SoC info like unique ID, etc.
seco lifecycle change	send SECO lifecycle update command (change) to SECO

```

COM8115200baud - Tera Term VT
File Edit Setup Control Window KanjiCode Help
Nameservice sent, ready for incoming messages...
Get Message From Master Side : "hello world!" [len : 12]
Get Message From Master Side : "abc" [len : 3]
Get New Line From Master Side
RPMSG String Echo FreeRTOS RTOS API Demo...
Nameservice sent, ready for incoming messages...
Hello from SCU (Build 3307, Commit f83a2bed, Mar 22 2019 10:39:10)
DDR frequency = 1200000000
ROM boot time = 220797 usec
SCFW boot time = 21783 usec
Banner = 6 usec
Init = 5400 usec
Config = 2513 usec
DDR = 52 usec
SConfig = 3330 usec
Prep = 6233 usec
*** Debug Monitor ***
>$
  
```

## Solution 2: Move i2c to A35 side(Cont.)

Check the changes by scfw **Debug Monitor command dump rm**  
System Controller Firmware Porting Guide(sc\_fw\_port.pdf)

M4 hold i2c1 in its partition **3**

```
dump rm
```

```
*** Resources ****
```

```
Partition: 3
```

```
I2C_1
```

```
*** Pads ***
```

```
Partition: 3
```

```
USB_SS3_TC1
```

```
USB_SS3_TC3
```

Move i2c to partition **5** for A35

```
dump rm
```

```
*** Resources ****
```

```
Partition: 5
```

```
I2C_1
```

```
*** Pads ***
```

```
Partition: 5
```

```
USB_SS3_TC1
```

```
USB_SS3_TC3
```

**Now, A35 side has i2c1 control now.**

**But M4 has no i2c1.**