目录

使用starter-kit开发RTOS应用

返回主目录 tda4系列

TI推出的 edgeai starter-kit 后称为SK板） 是一款针对边缘AI的开发板，针对Linux开发者设计，用户不需要了解底层RTOS的框架即可进行AI应用的开发。

与TI的官方EVM使用同一SDK进行开发，使得我们有可能通过简单的配置，将EVM的开发环境及开发步骤应用于SK板。

Linux SDK的修改

默认SK板的硬件都由Linux控制，包括DP显示 CSIRX的mipi输入 i2c接口等。

通过GPIO控制使能HDMI显示端口，但是在Linux中禁用HDMI.

> *您可能需要登陆才能查看下方代码*

禁用相关外设, 并配置CSIRX端口至QSH-ti cam接口，而不是 raspberry pi camera 接口。

/arch/arm64/boot/dts/ti/k3-j721e-edgeai-apps.dts.dts

```
@@ -0,0 +1,81 @@
+// SPDX-License-Identifier: GPL-2.0
+/*
+ * edgeai-apps: device-tree overlay
+ *
+ * Copyright (C) 2019 - 2021 Texas Instruments Incorporated -
http://www.ti.com/
+ */
+
+#include "k3-j721e-rtos-memory-map.dtsi"
+#include <dt-bindings/gpio/gpio.h>
+#include <dt-bindings/interrupt-controller/irq.h>
+#include <dt-bindings/pinctrl/k3.h>
+
+&main_i2c3 {
+   status = "disabled";
+};
+
+&serdes_wiz4 {
+   status = "disabled";
+};
+
+&mhdp {
+   status = "disabled";
+};
+
```

```
+&dss {
+   status = "disabled";
+};
+
+// &main_i2c1 {
+//     status = "disabled";
+// };
+
+&ti_csi2rx0 {
+   status = "disabled";
+};
+
+&ti_csi2rx1 {
+   status = "disabled";
+};
+
+
+&main_pmx0 {
+   main_rpi_cam0_reset_pins_default: main-rpi-cam0-reset-pins-default
{
+       pinctrl-single,pins = <
+           J721E_IOPAD(0x1D4, PIN_OUTPUT, 7) /* (Y3) SPI1_CS0 */
+       >;
+   };
+
+   main_rpi_cam1_reset_pins_default: main-rpi-cam1-reset-pins-default
{
+       pinctrl-single,pins = <
+           J721E_IOPAD(0x1E0, PIN_OUTPUT, 7) /* (Y5) SPI1_D0 */
+       >;
+   };
+
+   main_csi_mux_sel2_pins_default: main-csi-mux-sel2-pins-default {
+       pinctrl-single,pins = <
+           J721E_IOPAD(0x164, PIN_OUTPUT, 7) /* (V29) RGMII5_TD2 */
+       >;
+   };
+};
+
+&main_gpio0 {
+   pinctrl-names = "default";
+   pinctrl-0 = <&main_csi_mux_sel2_pins_default>;
+
+   csi-mux-hog {
+       /* CSI_MUX_SEL_2 */
+       gpio-hog;
+       gpios = <88 GPIO_ACTIVE_HIGH>;
+       output-low;
+       line-name = "CSI_MUX_SEL_2";
+   };
```

```
+
+   csi-rstz {
+       /* CSI_MUX_SEL_2 */
+       gpio-hog;
+       gpios = <78 GPIO_ACTIVE_LOW>;
+       output-high;
+       line-name = "CSI_CAM_RSTz";
+   };
+};
```

RTOS SDK的修改

使用HDMI进行显示

app_cfg_mcu2_0.h

```
#ifdef BUILD_MCU_BOARD_DEPENDENCIES

    #define ENABLE_CSI2RX
    #define ENABLE_CSI2TX

    /* IMPORANT NOTE:
     * - Only one of ENABLE_DSS_SINGLE or ENABLE_DSS_DUAL should be
defined
     * - When ENABLE_DSS_SINGLE is defined, only one of ENABLE_DSS_HDMI
or ENABLE_DSS_EDP should be defined
     * - When ENABLE_DSS_DUAL is defined, ENABLE_DSS_HDMI and
ENABLE_DSS_EDP are not used, both EDP and HDMI are enabled
unconditionally
     */
    #define ENABLE_DSS_SINGLE
    #define ENABLE_DSS_HDMI

    /* define below to enable eDP display,
       make sure to undef ENABLE_DSS_HDMI & ENABLE_DSS_DSI as well */
    //#define ENABLE_DSS_EDP
    #undef ENABLE_DSS_EDP
    /* define below to enable HDMI display,
       make sure to undef ENABLE_DSS_EDP & ENABLE_DSS_DSI as well */
    //#undef ENABLE_DSS_HDMI
    /* define below to enable DSI display, make sure to undef
ENABLE_DSS_HDMI
       & ENABLE_DSS_EDP as well */
    #undef ENABLE_DSS_DSI

    #define ENABLE_I2C
    #define ENABLE_BOARD
```

```
#else
```

修改makefile编译SK版的mcu2.0固件

```
# Inter-core virtual ethernet support
# Supported Values: yes | no
ifneq (,$(filter yes,$(BUILD_CPU_MCU2_0)))
ifeq ($(BUILD_QNX_A72),yes)
ETHFW_INTERCORE_ETH_SUPPORT?=no
else
ETHFW_INTERCORE_ETH_SUPPORT?=yes
endif
endif

#do

BUILD_EDGEAI?=yes

# If set to no, then MCU core firmware will be built with NO board
dependencies
# (such as I2C, board specific PINMUX, DSS, HDMI, I2C, ETHFW, CSIRX, CSITX).
Most of
# the packaged vision_apps require these interfaces on the MCU for the EVM,
but
# when porting to a board other than an EVM, or using applications which
control
# these interfaces from the HLOS on A72 (such as EdgeAI kit), then this
should be set
# to 'no'.
BUILD_MCU_BOARD_DEPENDENCIES?=yes

ifeq ($(BUILD_EDGEAI), yes)
#BUILD_MCU_BOARD_DEPENDENCIES=yes   #vivozhang
FIRMWARE_SUBFOLDER=vision_apps_eaik
UENV_NAME=uEnv_$(SOC)_edgeai_apps.txt
endif

ifeq ($(BUILD_MCU_BOARD_DEPENDENCIES),no)
BUILD_ENABLE_ETHFW=no
endif

BUILD_ENABLE_ETHFW=no

# Need to export this variable so that the following xdc .cfg file can pick
this up from the env:
# ${PSDK_PATH}/vision_apps/platform/$(SOC)/rtos/mcu2_0/mcu2_0.cfg
export BUILD_ENABLE_ETHFW

# A72 OS specific Build flag
```

```
BUILD_LINUX_A72?=yes
BUILD_QNX_A72?=no
```

增加gpio的pinmux配置

vision_apps/utils/misc/src/app_pinmux.c

```
static pinmuxPerCfg_t gGpio0PinCfg[] =
{
    /* GpioCamCtrl -> GPIO0_64 -> AD28 */
    {
        PIN_PRG0_PRU1_GPO1, PIN_MODE(7) | \
        ((PIN_PULL_DISABLE | PIN_INPUT_ENABLE) & (~PIN_PULL_DIRECTION))
    },
    /* GpioCamCtrl -> GPIO0_74 -> AG26 */
    {
        PIN_PRG0_PRU1_GPO11, PIN_MODE(7) | \
        ((PIN_PULL_DISABLE | PIN_INPUT_ENABLE) & (~PIN_PULL_DIRECTION))
    },
    /* GpioCamCtrl -> GPIO0_75 -> AF27 */
    {
        PIN_PRG0_PRU1_GPO12, PIN_MODE(7) | \
        ((PIN_PULL_DISABLE | PIN_INPUT_ENABLE) & (~PIN_PULL_DIRECTION))
    },
    /* GpioCamCtrl -> GPIO0_76 -> AF26 */
    {
        PIN_PRG0_PRU1_GPO13, PIN_MODE(7) | \
        ((PIN_PULL_DISABLE | PIN_INPUT_ENABLE) & (~PIN_PULL_DIRECTION))
    },
    /* GpioCamCtrl -> GPIO0_77 -> AE25 */
    {
        PIN_PRG0_PRU1_GPO14, PIN_MODE(7) | \
        ((PIN_PULL_DISABLE | PIN_INPUT_ENABLE) & (~PIN_PULL_DIRECTION))
    },
    /* GpioCamCtrl -> GPIO0_78 -> AF29 */
    {
        PIN_PRG0_PRU1_GPO15, PIN_MODE(7) | \
        ((PIN_PULL_DISABLE | PIN_INPUT_ENABLE) & (~PIN_PULL_DIRECTION))
    },
    /* GpioCamCtrl -> GPIO0_79 -> AG29 */
    {
        PIN_PRG0_PRU1_GPO16, PIN_MODE(7) | \
        ((PIN_PULL_DISABLE | PIN_INPUT_ENABLE) & (~PIN_PULL_DIRECTION))
    },
    {PINMUX_END}
};

static pinmuxModuleCfg_t gGpioPinCfg[] =
{
    {0, TRUE, gGpio0PinCfg},
    {PINMUX_END}
```

```
};



static pinmuxPerCfg_t gI2c3PinCfg[] =
{
    /* camCtrl -> I2C3_SCL -> T26 */
    {
        PIN_MMC2_CLK, PIN_MODE(4) | \
        ((PIN_PULL_DIRECTION | PIN_INPUT_ENABLE) & (~PIN_PULL_DISABLE))
    },
    /* camCtrl -> I2C3_SDA -> T25 */
    {
        PIN_MMC2_CMD, PIN_MODE(4) | \
        ((PIN_PULL_DIRECTION | PIN_INPUT_ENABLE) & (~PIN_PULL_DISABLE))
    },
    {PINMUX_END}
};


static pinmuxModuleCfg_t gI2cPinCfg[] =
{
    {3, TRUE, gI2c3PinCfg},
    {PINMUX_END}
};


static pinmuxBoardCfg_t gBasicDemoPinmuxDataInfo[] =
{
    {0, gDispPinCfg},
    {1, gCaptPinCfg},
    // {2, gGpioPinCfg},
    {2, gI2cPinCfg},
    {PINMUX_END}
};
```

From:
https://wiki.yytech.cc/ - **YYT Docs**

Permanent link:
https://wiki.yytech.cc/doku.php?id=tda4%E7%B3%BB%E5%88%97:%E5%88%A9%E7%94%A8starter-kit%E5%BC%80%E5%8F%91%E6%9D%BF%E5%BC%80%E5%8F%91rtos%E5%BA%94%E7%94%A8

Last update: **2023/10/09 11:45**