EMBEDDED LINUX - Kernel Device Tree

Device Tree Source configuration to add CAN protocol support and PinMux setup. Build, deploy and test on BeagleBone of custom Device Tree Binary

Resources
Device Tree Usage
Processor SDK Linux Software Developer's Guide

Contents

DEVICE TREE FOR AM335x SOC AND BEAGLEBONE

Get into the kernel sources directory.

```
cd ${DISCOPATH}/kernel/bb-kernel/KERNEL
```

Go back to the TI documentation: Processor SDK Linux Software Developer's Guide.
The section `3.2.1.5. Compiling the Sources` of the TI documentation deals with the compilation of the Device Tree Binaries.

- Which Device Tree file should we use as a source for compiling the BeagleBone Black device tree?

>

- Open this file (`arch/arm/boot/dts/am335x-boneblack.dts`) in a text editor and note the included sources files.

> #include "am33xx.dtsi"
> #include "am335x-bone-common.dtsi"
> #include "am335x-boneblack-common.dtsi"

- Open those files too and explain the link between the 4 files.

>

Full explanation of a device tree file can be found in this reference: Device Tree Usage. Have a quick look at the `Basic Concepts` section.

CAN SUPPORT IN THE AM335x DEVICE TREE

Obviously we need to add the CAN support to the device tree.
Look at the Linux documentation of the CAN driver for the TI AM335x OMAP SoC.

```
cat Documentation/devicetree/bindings/net/can/c_can.txt
```

- Which files need to be edited in order to add the CAN support to the device tree?

>

Have a look at the SoC common .dtsi file.

```
dcan1: can@481d0000 {
    compatible = "ti,am3352-d_can";
    ti,hwmods = "d_can1";
    reg = <0x481d0000 0x2000>;
    clocks = <&dcan1_fck>;
    clock-names = "fck";
    syscon-raminit = <&scm_conf 0x644 1>;
    interrupts = <55>;
    status = "disabled";
};
```

The fields values seem to be close to the example from the Linux documentation. How can you prove those values are right? Let's have a look at the AM335x Technical Reference Manual, more specifically:

- Table 2-3. Peripheral Memory Map ---> Search DCAN1 keywords
- Table 6-1. ARM Cortex-A8 Interrupts ---> Search DCAN1_INT0
- Table 9-10. CONTROL_MODULE REGISTERS ---> Search dcan_raminit
- Justify each field value, using either the TI Technical Reference Manual or the CAN Linux documentation.

>

PINMUX SUPPORT IN THE AM335x DEVICE TREE

In addition to the CAN support, we have to set the pinmux (pin multiplexing) support in the device tree. As you already know, several peripherals can be linked to a signle pin. The pin controller is used to establish which peripheral will have access to a given pin.
Observe the Linux documentation of the pin controller for the TI AM335x OMAP SoC.

```
cat Documentation/devicetree/bindings/pinctrl/pinctrl-bindings.txt
```

We have to determine on which pins the DCAN1 peripheral must be attached. Of course we can use the AM3358 datasheet but there are some user-friendly tools on the Internet:

- AM3358 datasheet
- BeagleBone Black Pin Mux Spreadsheet
- BeagleBone Pin Mux Table
- TI SysConfig
- Which pins are used for the dcan1 peripheral? Give the pins number, their address, their default name, their name when used with dcan1, the mode in which they are used by the dcan1 peripheral, and their address.

>

MODIFY DEVICE TREE SOURCES

Modify and add CAN support in the Device Tree Sources for BeagleBone.

```
nano arch/arm/boot/dts/am335x-boneblack.dts
```

Append the following configuration to the `am335x-boneblack.dts` file.

```
/* add by ensicaen students during embedded linux labs */
&am33xx_pinmux {

dcan1_pins: pinmux_dcan1_pins {
        pinctrl-single,pins = <
        0x180 0x12 /* P9.24 uart1_txd.dcan1_rx : PULLUP | OUTPUT | MODE 2 */
        0x184 0x22 /* P9.26 uart1_rxd.uart1_tx : PULLUP | INPUT | MODE 2 */
        >;
    };
};


&dcan1 {
    status = "okay";
    pinctrl-names = "default";
    pinctrl-0 = <&dcan1_pins>;
};
```

Save and verify the `am335x-boneblack.dts` file content.

```
cat arch/arm/boot/dts/am335x-boneblack.dts
```

BUILD, DEPLOY AND TEST CAN SUPPORT SOLUTION ON LINUX

Generate the `am335x-boneblack.dtb` device tree binary (this is one of the steps of the TI documentation).

```
make ARCH=arm CROSS_COMPILE=${CC} am335x-boneblack.dtb
```

- Where has been placed the generated file?

>

Copy the new DTB into our `deploy` directory.

```
cp  arch/arm/boot/dts/am335x-boneblack.dtb ${DISCOPATH}/deploy/am335x-
boneblack_can.dtb
```

Now we can finally deploy the device tree onto the SD card.

```
cd ${DISCOPATH}/deploy
sudo cp -v am335x-boneblack_can.dtb ${MEDIA}/rootfs/boot/am335x-boneblack.dtb
sync
sudo umount ${MEDIA}/rootfs
```

Start the BeagleBone Black with the updated SD card. Log in when ready and verify if the CAN network is now supported.

```
ifconfig -a
```

- Which interfaces do you see?

>

You win! You've just finished to add a new driver and network support in the Linux Kernel!

You can test your CAN peripheral using the `disco/apps/can_write.c` testing app associated to the CANBUS Cape for BeagleBone.

- CAN user space application
- CANBUS Cape