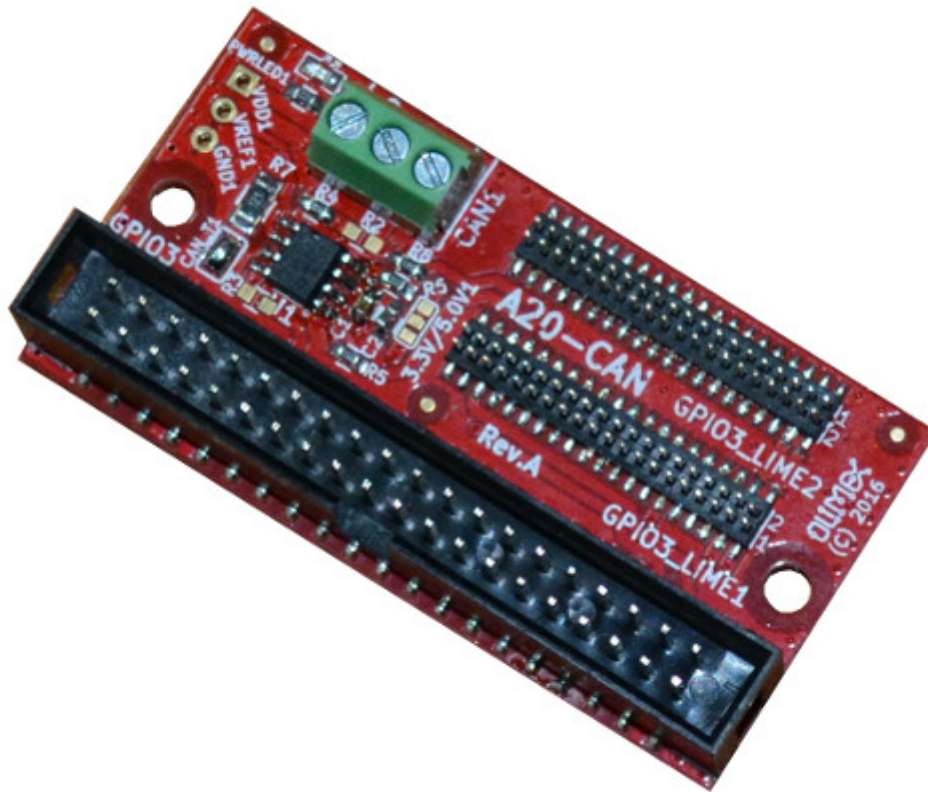


# A20-CAN



A20-CAN is CAN driver board for A20 Allwinner SOCs. A20-CAN features are:

- compatible with A20-OLinuXino-LIME, A20-OLinuXino-LIME2, A20-OLinuXino-MICRO, A20-SOM-EVB
- connects to GPIO3 connector
- allow CAN networking of A20 boards running Linux
- Power LED
- PCB dimensions: 60x30mm
- supported in Olimage Linux



## Contents

---

- 1\_Enabling A20-CAN in Olimage Linux
- 2\_Building Linux with CAN support
  - 2.1\_Get sources
  - 2.2\_For Linux-Sunxi kernel (3.4)
  - 2.3\_For Mainline kernel
  - 2.4\_CAN bus usage

## Enabling A20-CAN in Olimage Linux

---

Run the built-in script:

```
o linuxino-overlay
```

Navigate down and enable with "space" key either "sun7i-a20-can-pa.dtbo" or "sun7i-a20-can.dtbo" depending on which port the A20-CAN is connected:

```
[ ] sun7i-a20-can-pa.dtbo          Enable CAN controller on PA
[*] sun7i-a20-can.dtbo           Enable CAN controller
```

sun7i-a20-can-pa.dtbo sets pins "PA16", "PA17"

sun7i-a20-can.dtbo sets pins "PH20", "PH21"

Navigate to <Ok> with "tab" key and save changes with "enter" key. Reboot when prompted.

When done refer further below to [CAN bus usage](#)

## Building Linux with CAN support

---

### Get sources

---

Download kernel tree from:

```
git clone https://github.com/linux-sunxi/linux-sunxi.git
```

### For Linux-Sunxi kernel (3.4)

---

The can driver is not in the kernel tree. This is done with the following steps:

Download the can driver:

```
clone https://github.com/btolfa/sunxi-can-driver.git
```

Copy file to kernel tree:

```
cp -v sunxi-can-driver/sun7i_can.* linux-sunxi/drivers/net/can
```

Add Kconfig selection:

```
vi linux-sunxi/drivers/net/can/Kconfig
```

Add the following:

```
config CAN_SUN7I
tristate "Sun7i CAN bus controller"
default n
help
This is the Sun7i CAN BUS driver for android system by peter chen.
```

Modify Makefile:

```
vi linux-sunxi/drivers/net/can/Makefile
```

Add:

```
obj-$(CONFIG_CAN_SUN7I) += sun7i_can.o
```

**NOTE:** The example build is with default sun7i\_defconfig. You can use your own. Now your device tree is ready for build. However the new module is not selected.

Modify kernel:

```
cd linux-sunxi
make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- -j4 sun7i_defconfig
make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- -j4 menuconfig
```

Select from:

```
[*] Networking support --->
    <*> CAN bus subsystem support --->
        <*> CAN Device Drivers --->
            [M] Sun7i CAN bus controller
```

Save current configuration and build:

```
make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- -j4 uImage modules
```

Modify can section of script.bin file:

```
[can_para]
can_used = 1
can_tx = port:PH20<4><default><default><default>
can_rx = port:PH21<4><default><default><default>
```

Copy kernel and modules to SD-CARD and boot the system.

## For Mainline kernel

---

In the mainline kernel there is support for CAN bus. However we need to declare that it's gonna be used.

First of all checkout \_\_sunxi\_next\_\_ branch:

```
cd linux-sunxi
git checkout -b sunxi-next origin/sunxi-next
```

Select \_\_sunxi\_defconfig\_\_:

```
make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- -j4 sunxi_defconfig
```

**NOTE:** By default can module is build as part of the kernel.  
**NOTE:** In the example we are gonna use A20-OLinuxino-MICRO, but all other A20 based boards can be used.  
**NOTE:** See device-tree binding in:

```
/compile/A20/linux-sunxi/Documentation/devicetree/bindings/net/can/sun4i_can.txt
```

Modify common device-tree sources:

```
vi arch/arm/boot/dts/sun7i-a20.dtsi
```

Append at line 1216:

```
can0_pins_a: can0@0 {
    allwinner,pins = "PH20", "PH21";
    allwinner,function = "can";
    allwinner,drive = <0>;
    allwinner,pull = <0>;
};
```

Append at line 1442:

```
can0: can@01c2bc00 {
    compatible = "allwinner,sun4i-a10-can";
    reg = <0x01c2bc00 0x400>;
    interrupts = <0 26 4>;
    clocks = <&apb1_gates 4>;
    status = "disabled";
};
```

Open arch/arm/boot/dts/sun7i-a20-olinuxino-micro.dts:

```
vi arch/arm/boot/dts/sun7i-a20-olinuxino-micro.dts
```

Append at the bottom:

```
&can0 {
    pinctrl-names = "default";
    pinctrl-0 = <&can0_pins_a>;
    status = "okay";
};
```

