Linux CAN

开发指南

版本号：1.0
发布日期：2022.02.21

版本历史

版本号	日期	制/修订人	内容描述
1.0	2022.02.21	AWA1730	添加初版

目　录

插　图

1 前言

1.1 文档简介

介绍 CAN 模块的使用方法，方便开发人员使用。

1.2 目标读者

CAN 模块的驱动开发/维护人员。

1.3 相关术语介绍

表 1-1: 术语介绍

术语	解释说明
Sunxi	指 Allwinner 的一系列 SOC 硬件平台

2 模块配置介绍

2.1 设备树配置

CAN 模块的设备树配置位于 device/config/chips/{ic}/configs/{board}/board.dts 下面为打开 can0 的配置：

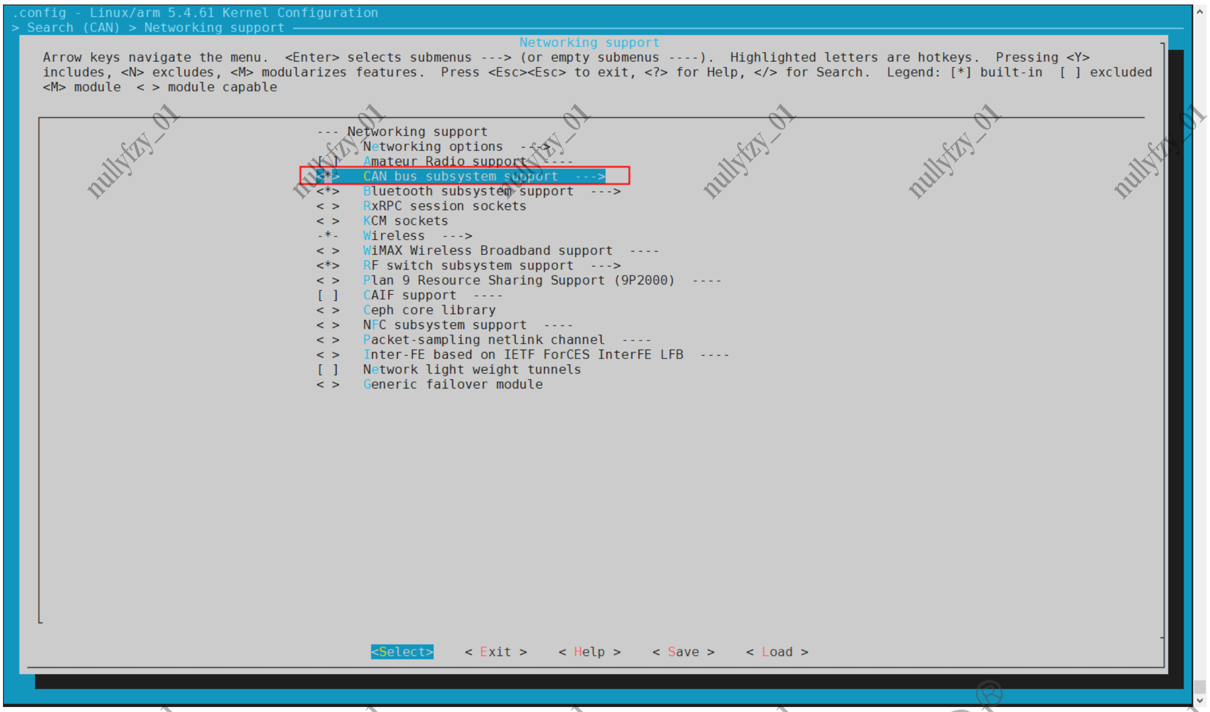
```
    can0: can@0x0 {
        #address-cells = <1>;
        #size-cells = <0>;
        compatible = "allwinner,sun8i-can";
        device_type = "can0";
        id = <0>;
        status = "disabled";
    };
```

注意：id = <0> 表示指 can0, id = <1> 表示 can1; status = "okay" 表示打开 can 模块，status = "disabled" 表示关闭 can 模块。

2.2 内核 menuconfig 配置

SDK 根目录中执行./build.sh menuconfig 打开配置项界面；./build.sh saveconfig 保存配置项选择。

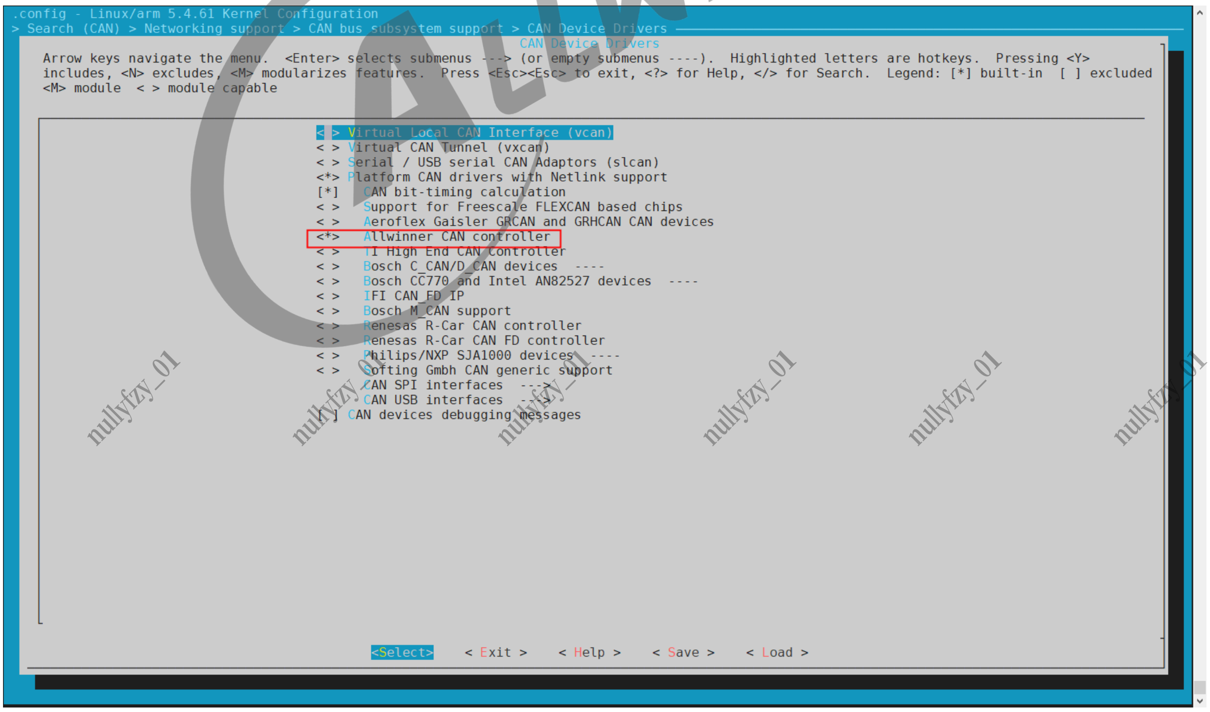
- 首先，CAN 模块配置，如下图所示：



```
.config - Linux/arm 5.4.61 Kernel Configuration
> Search (CAN) > Networking support
                                        Networking support
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y>
  includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] built-in  [ ] excluded
  <M> module  < > module capable

                                  --- Networking support
                                        Networking options  --->
                                  [ ]   Amateur Radio support  ----
                                  <*>   CAN bus subsystem support  --->
                                  <*>   Bluetooth subsystem support  --->
                                  < >   RxRPC session sockets
                                  < >   KCM sockets
                                  -*-   Wireless  --->
                                  < >   WiMAX Wireless Broadband support  ----
                                  <*>   RF switch subsystem support  --->
                                  < >   Plan 9 Resource Sharing Support (9P2000)  ----
                                  [ ]   CAIF support  ----
                                  < >   Ceph core library
                                  < >   NFC subsystem support  ----
                                  < >   Packet-sampling netlink channel  ----
                                  < >   Inter-FE based on IETF ForCES InterFE LFB  ----
                                  [ ]   Network light weight tunnels
                                  < >   Generic failover module

                   <Select>    < Exit >    < Help >    < Save >    < Load >
```

图 2-1: 内核配置 1

- 然后，选择 allwinner CAN 选项，如下图所示：

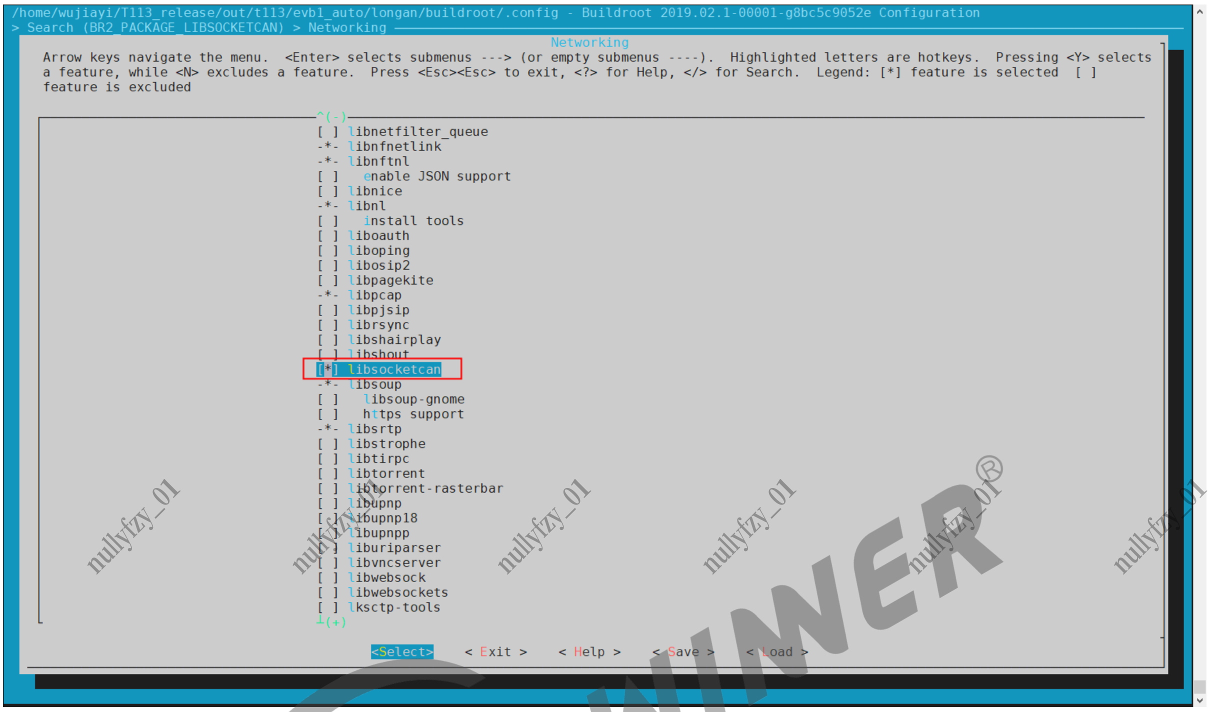


```
.config - Linux/arm 5.4.61 Kernel Configuration
> Search (CAN) > Networking support > CAN bus subsystem support > CAN Device Drivers
                                        CAN Device Drivers
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y>
  includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] built-in  [ ] excluded
  <M> module  < > module capable

                                  < > Virtual Local CAN Interface (vcan)
                                  < > Virtual CAN Tunnel (vxcan)
                                  < > Serial / USB serial CAN Adaptors (slcan)
                                  <*> Platform CAN drivers with Netlink support
                                  [*]   CAN bit-timing calculation
                                  < >   Support for Freescale FLEXCAN based chips
                                  < >   Aeroflex Gaisler GRCAN and GRHCAN CAN devices
                                  <*>   Allwinner CAN controller
                                  < >   TI High End CAN Controller
                                  < >   Bosch C_CAN/D_CAN devices  ----
                                  < >   Bosch CC770 and Intel AN82527 devices  ----
                                  < >   IFI CAN_FD IP
                                  < >   Bosch M_CAN support
                                  < >   Renesas R-Car CAN controller
                                  < >   Renesas R-Car CAN FD controller
                                  < >   Philips/NXP SJA1000 devices  ----
                                  < >   Softing Gmbh CAN generic support
                                        CAN SPI interfaces  --->
                                        CAN USB interfaces  --->
                                  [ ] CAN devices debugging messages

                   <Select>    < Exit >    < Help >    < Save >    < Load >
```

图 2-2: 内核配置 2

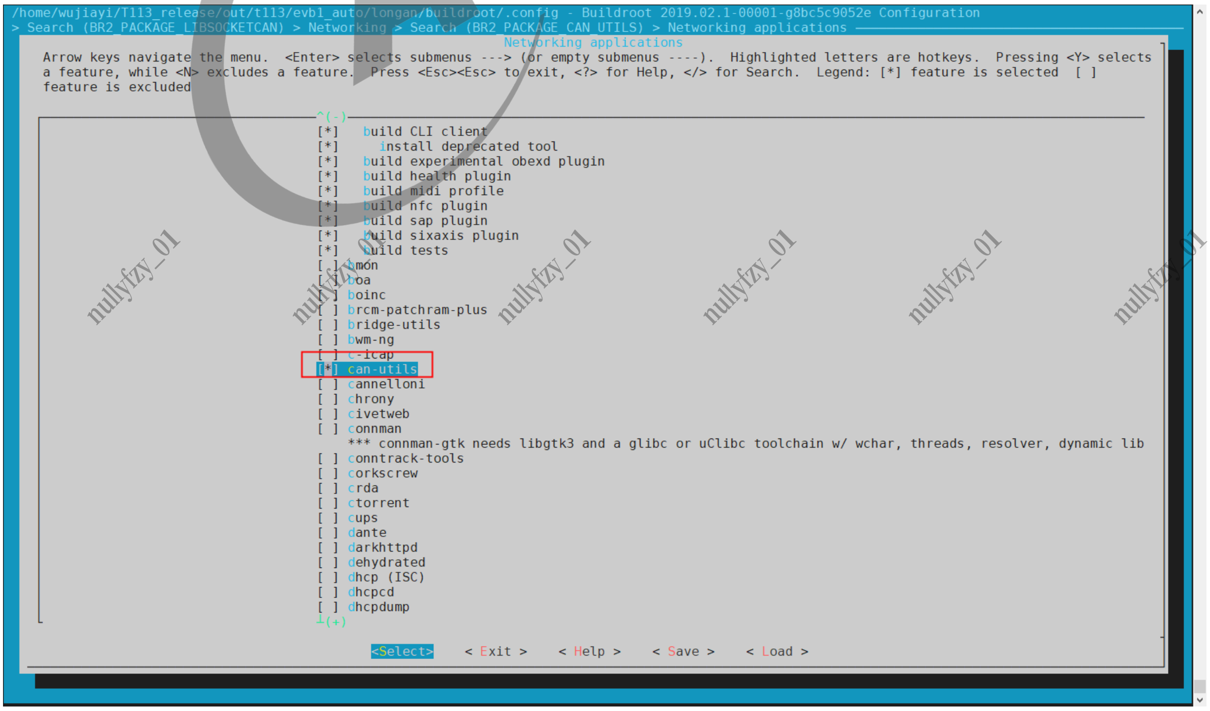
2.3 buildroot 的 menuconfig 配置

- 打开对应 CAN 工具



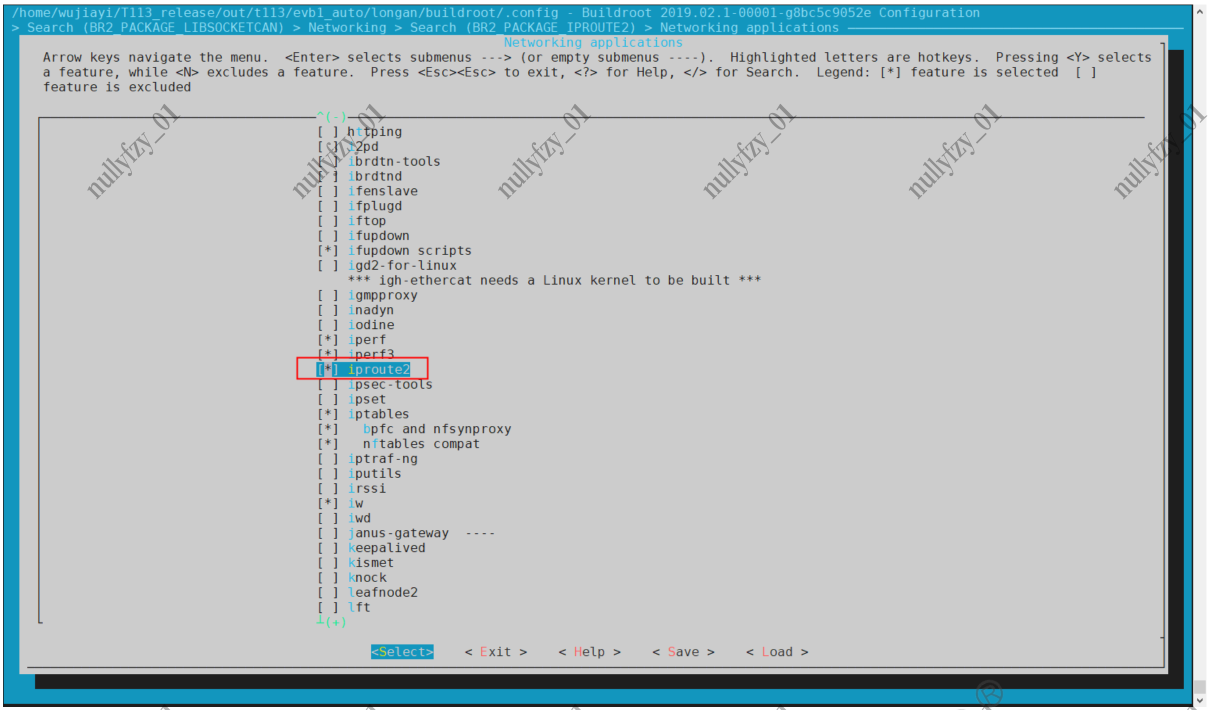
```
/home/wujiayi/T113_release/out/t113/evb1_auto/longan/buildroot/.config - Buildroot 2019.02.1-00001-g8bc5c9052e Configuration
> Search (BR2_PACKAGE_LIBSOCKETCAN) > Networking ─────────────────────────────────────────
                                          Networking
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects
  a feature, while <N> excludes a feature.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ]
  feature is excluded

                                  ^(-)
                                  [ ] libnetfilter_queue
                                  -*- libnfnetlink
                                  -*- libnftnl
                                  [ ]   enable JSON support
                                  [ ] libnice
                                  -*- libnl
                                  [ ]   install tools
                                  [ ] liboauth
                                  [ ] liboping
                                  [ ] libosip2
                                  [ ] libpagekite
                                  -*- libpcap
                                  [ ] libpjsip
                                  [ ] librsync
                                  [ ] libshairplay
                                  [ ] libshout
                                  [*] libsocketcan
                                  -*- libsoup
                                  [ ]   libsoup-gnome
                                  [ ]   https support
                                  -*- libsrtp
                                  [ ] libstrophe
                                  [ ] libtirpc
                                  [ ] libtorrent
                                  [ ] libtorrent-rasterbar
                                  [ ] libupnp
                                  [ ] libupnp18
                                  [ ] libupnpp
                                  [ ] liburiparser
                                  [ ] libvncserver
                                  [ ] libwebsock
                                  [ ] libwebsockets
                                  [ ] lksctp-tools
                                  ┴(+)

                        <Select>    < Exit >    < Help >    < Save >    < Load >
```

图 2-3: buildroot 配置 1



```
/home/wujiayi/T113_release/out/t113/evb1_auto/longan/buildroot/.config - Buildroot 2019.02.1-00001-g8bc5c9052e Configuration
> Search (BR2_PACKAGE_LIBSOCKETCAN) > Networking > Search (BR2_PACKAGE_CAN_UTILS) > Networking applications ──────────────
                                    Networking applications
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects
  a feature, while <N> excludes a feature.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ]
  feature is excluded

                                  ^(-)
                                  [*]   build CLI client
                                  [*]     install deprecated tool
                                  [*]   build experimental obexd plugin
                                  [*]   build health plugin
                                  [*]   build midi profile
                                  [*]   build nfc plugin
                                  [*]   build sap plugin
                                  [*]   build sixaxis plugin
                                  [*]   build tests
                                  [ ] bmon
                                  [ ] boa
                                  [ ] boinc
                                  [ ] brcm-patchram-plus
                                  [ ] bridge-utils
                                  [ ] bwm-ng
                                  [ ] c-icap
                                  [*] can-utils
                                  [ ] cannelloni
                                  [ ] chrony
                                  [ ] civetweb
                                  [ ] connman
                                      *** connman-gtk needs libgtk3 and a glibc or uClibc toolchain w/ wchar, threads, resolver, dynamic lib
                                  [ ] conntrack-tools
                                  [ ] corkscrew
                                  [ ] crda
                                  [ ] ctorrent
                                  [ ] cups
                                  [ ] dante
                                  [ ] darkhttpd
                                  [ ] dehydrated
                                  [ ] dhcp (ISC)
                                  [ ] dhcpcd
                                  [ ] dhcpdump
                                  ┴(+)

                        <Select>    < Exit >    < Help >    < Save >    < Load >
```

图 2-4: buildroot 配置 2



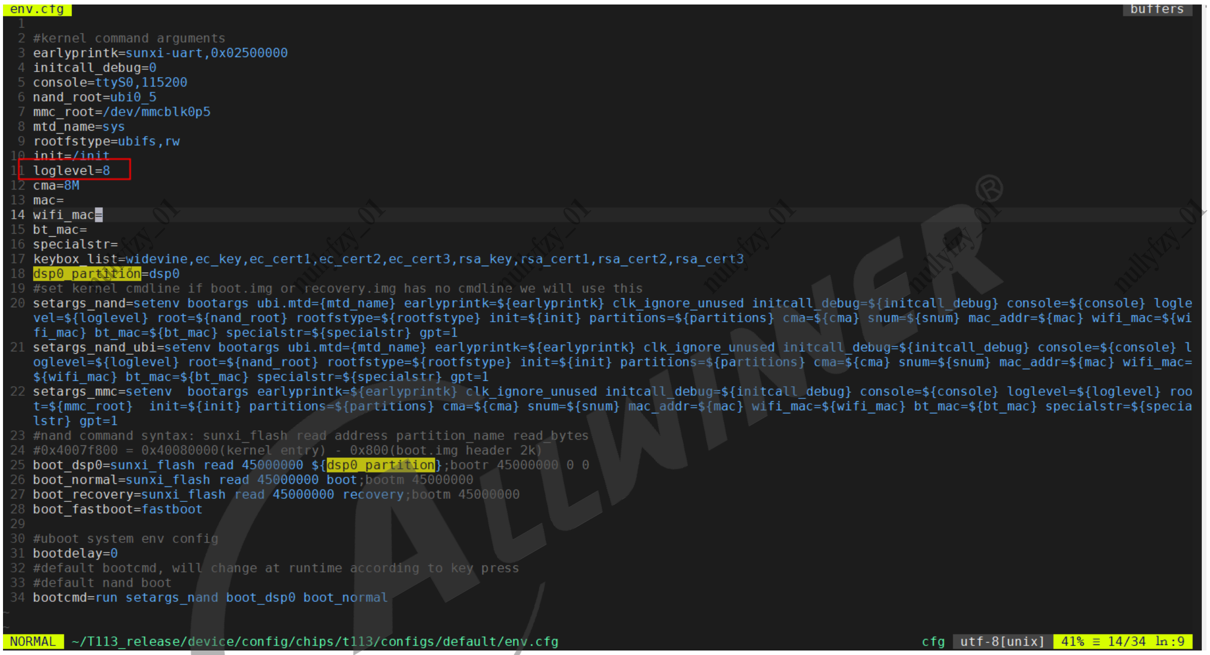
```
/home/wujiayi/T113_release/out/t113/evb1_auto/longan/buildroot/.config - Buildroot 2019.02.1-00001-g8bc5c9052e Configuration
> Search (BR2_PACKAGE_LIBSOCKETCAN) > Networking > Search (BR2_PACKAGE_IPROUTE2) > Networking applications
                                    Networking applications
   Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects
   a feature, while <N> excludes a feature.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ]
   feature is excluded

                                      ^(-)
                                      [ ] httping
                                      [ ] i2pd
                                      [ ] ibrdtn-tools
                                      [ ] ibrdtnd
                                      [ ] ifenslave
                                      [ ] ifplugd
                                      [ ] iftop
                                      [ ] ifupdown
                                      [*] ifupdown scripts
                                      [ ] igd2-for-linux
                                          *** igh-ethercat needs a Linux kernel to be built ***
                                      [ ] igmpproxy
                                      [ ] inadyn
                                      [ ] iodine
                                      [*] iperf
                                      [*] iperf3
                                      [*] iproute2
                                      [ ] ipsec-tools
                                      [ ] ipset
                                      [*] iptables
                                      [*]   bpfc and nfsynproxy
                                      [*]   nftables compat
                                      [ ] iptraf-ng
                                      [ ] iputils
                                      [ ] irssi
                                      [*] iw
                                      [ ] iwd
                                      [ ] janus-gateway  ----
                                      [ ] keepalived
                                      [ ] kismet
                                      [ ] knock
                                      [ ] leafnode2
                                      [ ] lft
                                      ┴(+)

                                          <Select>    < Exit >    < Help >    < Save >    < Load >
```

图 2-5: buildroot 配置 3

3 FAQ

3.1 调试等级

调低打印等级 (8 为最低，0 为最高)，保证 CAN 驱动调试信息能够输出

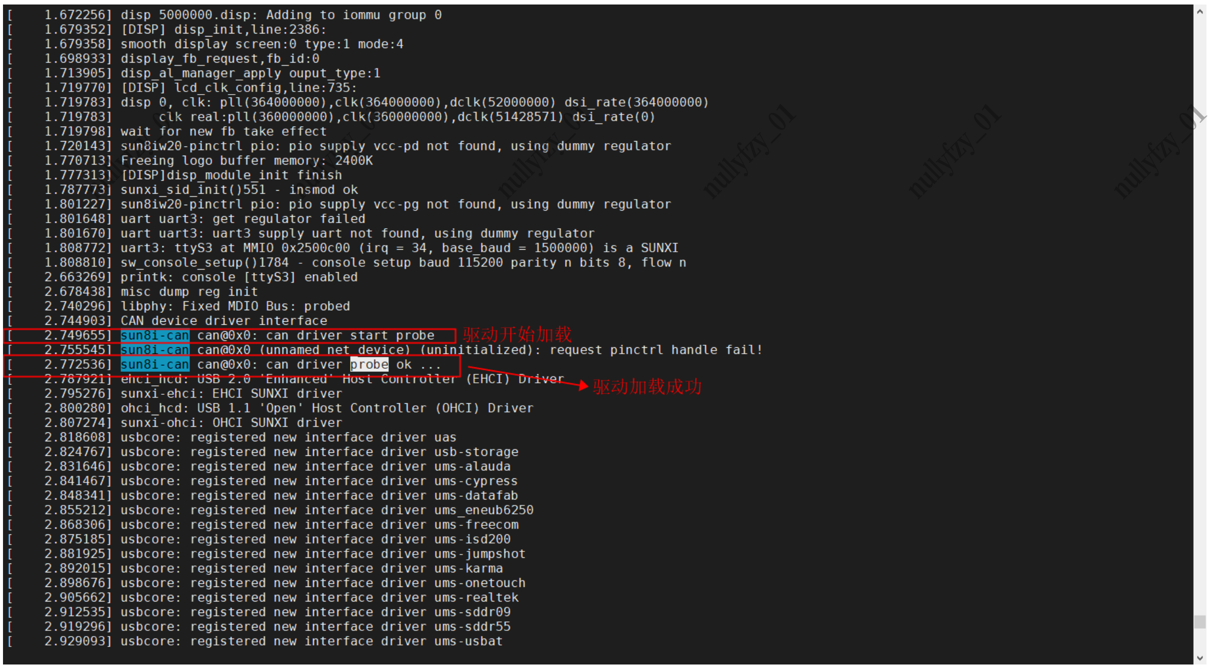


```

#kernel command arguments
earlyprintk=sunxi-uart,0x02500000
initcall_debug=0
console=ttyS0,115200
nand_root=ubi0_5
mmc_root=/dev/mmcblk0p5
mtd_name=sys
rootfstype=ubifs,rw
init=/init
loglevel=8
cma=8M
mac=
wifi_mac=
bt_mac=
specialstr=
keybox_list=widevine,ec_key,ec_cert1,ec_cert2,ec_cert3,rsa_key,rsa_cert1,rsa_cert2,rsa_cert3
dsp0_partition=dsp0
#set kernel cmdline if boot.img or recovery.img has no cmdline we will use this
setargs_nand=setenv bootargs ubi.mtd={mtd_name} earlyprintk=${earlyprintk} clk_ignore_unused initcall_debug=${initcall_debug} console=${console} loglevel=${loglevel} root=${nand_root} rootfstype=${rootfstype} init=${init} partitions=${partitions} cma=${cma} snum=${snum} mac_addr=${mac} wifi_mac=${wifi_mac} bt_mac=${bt_mac} specialstr=${specialstr} gpt=1
setargs_nand_ubi=setenv bootargs ubi.mtd={mtd_name} earlyprintk=${earlyprintk} clk_ignore_unused initcall_debug=${initcall_debug} console=${console} loglevel=${loglevel} root=${nand_root} rootfstype=${rootfstype} init=${init} partitions=${partitions} cma=${cma} snum=${snum} mac_addr=${mac} wifi_mac=${wifi_mac} bt_mac=${bt_mac} specialstr=${specialstr} gpt=1
setargs_mmc=setenv  bootargs earlyprintk=${earlyprintk} clk_ignore_unused initcall_debug=${initcall_debug} console=${console} loglevel=${loglevel} root=${mmc_root}  init=${init} partitions=${partitions} cma=${cma} snum=${snum} mac_addr=${mac} wifi_mac=${wifi_mac} bt_mac=${bt_mac} specialstr=${specialstr} gpt=1
#nand command syntax: sunxi_flash read address partition_name read_bytes
#0x4007f800 = 0x40080000(kernel entry) - 0x800(boot.img header 2k)
boot_dsp0=sunxi_flash read 45000000 ${dsp0_partition};bootr 45000000 0 0
boot_normal=sunxi_flash read 45000000 boot;bootm 45000000
boot_recovery=sunxi_flash read 45000000 recovery;bootm 45000000
boot_fastboot=fastboot

#uboot system env config
bootdelay=0
#default bootcmd, will change at runtime according to key press
#default nand boot
bootcmd=run setargs_nand boot_dsp0 boot_normal
~
~
NORMAL  ~/T113_release/device/config/chips/t113/configs/default/env.cfg          cfg  utf-8[unix]  41% ≡ 14/34 ln:9
```

图 3-1: 调试等级

3.2 驱动加载

板子上电检查 log



```
[    1.672256] disp 5000000.disp: Adding to iommu group 0
[    1.679352] [DISP] disp_init,line:2386:
[    1.679358] smooth display screen:0 type:1 mode:4
[    1.698933] display_fb_request,fb_id:0
[    1.713905] disp_al_manager_apply ouput_type:1
[    1.719770] [DISP] lcd_clk_config,line:735:
[    1.719783] disp 0, clk: pll(364000000),clk(364000000),dclk(52000000) dsi_rate(364000000)
[    1.719783]      clk real:pll(360000000),clk(360000000),dclk(51428571) dsi_rate(0)
[    1.719798] wait for new fb take effect
[    1.720143] sun8iw20-pinctrl pio: pio supply vcc-pd not found, using dummy regulator
[    1.770713] Freeing logo buffer memory: 2400K
[    1.777313] [DISP]disp_module_init finish
[    1.787773] sunxi_sid_init()551 - insmod ok
[    1.801227] sun8iw20-pinctrl pio: pio supply vcc-pg not found, using dummy regulator
[    1.801648] uart uart3: get regulator failed
[    1.801670] uart uart3: uart3 supply uart not found, using dummy regulator
[    1.808772] uart3: ttyS3 at MMIO 0x2500c00 (irq = 34, base_baud = 1500000) is a SUNXI
[    1.808810] sw_console_setup()1784 - console setup baud 115200 parity n bits 8, flow n
[    2.663269] printk: console [ttyS3] enabled
[    2.678438] misc dump reg init
[    2.740296] libphy: Fixed MDIO Bus: probed
[    2.744903] CAN device driver interface
[    2.749655] sun8i-can can@0x0: can driver start probe
[    2.755545] sun8i-can can@0x0 (unnamed net device) (uninitialized): request pinctrl handle fail!
[    2.772536] sun8i-can can@0x0: can driver probe ok ...
[    2.787921] ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver
[    2.795276] sunxi-ehci: EHCI SUNXI driver
[    2.800280] ohci_hcd: USB 1.1 'Open' Host Controller (OHCI) Driver
[    2.807274] sunxi-ohci: OHCI SUNXI driver
[    2.818608] usbcore: registered new interface driver uas
[    2.824767] usbcore: registered new interface driver usb-storage
[    2.831646] usbcore: registered new interface driver ums-alauda
[    2.841467] usbcore: registered new interface driver ums-cypress
[    2.848341] usbcore: registered new interface driver ums-datafab
[    2.855212] usbcore: registered new interface driver ums_eneub6250
[    2.868306] usbcore: registered new interface driver ums-freecom
[    2.875185] usbcore: registered new interface driver ums-isd200
[    2.881925] usbcore: registered new interface driver ums-jumpshot
[    2.892015] usbcore: registered new interface driver ums-karma
[    2.898676] usbcore: registered new interface driver ums-onetouch
[    2.905662] usbcore: registered new interface driver ums-realtek
[    2.912535] usbcore: registered new interface driver ums-sddr09
[    2.919296] usbcore: registered new interface driver ums-sddr55
[    2.929093] usbcore: registered new interface driver ums-usbat
```

驱动开始加载

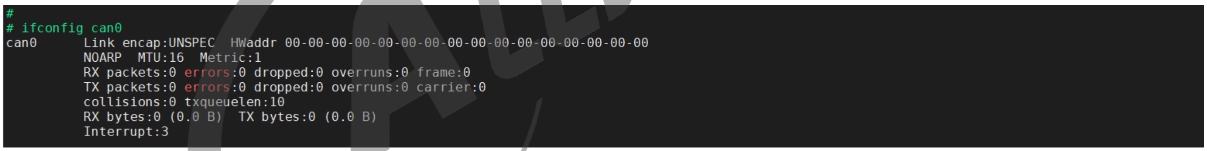
驱动加载成功

图 3-2: 上电 log0

进入终端后，检查 CAN 是否存在，输入命令

```
ifconfig can0
```

可以看到输出 log，证明已经存在 can0



```
#
# ifconfig can0
can0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          NOARP  MTU:16  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:10
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
          Interrupt:3
```

图 3-3: 上电 log1

3.3 通讯测试

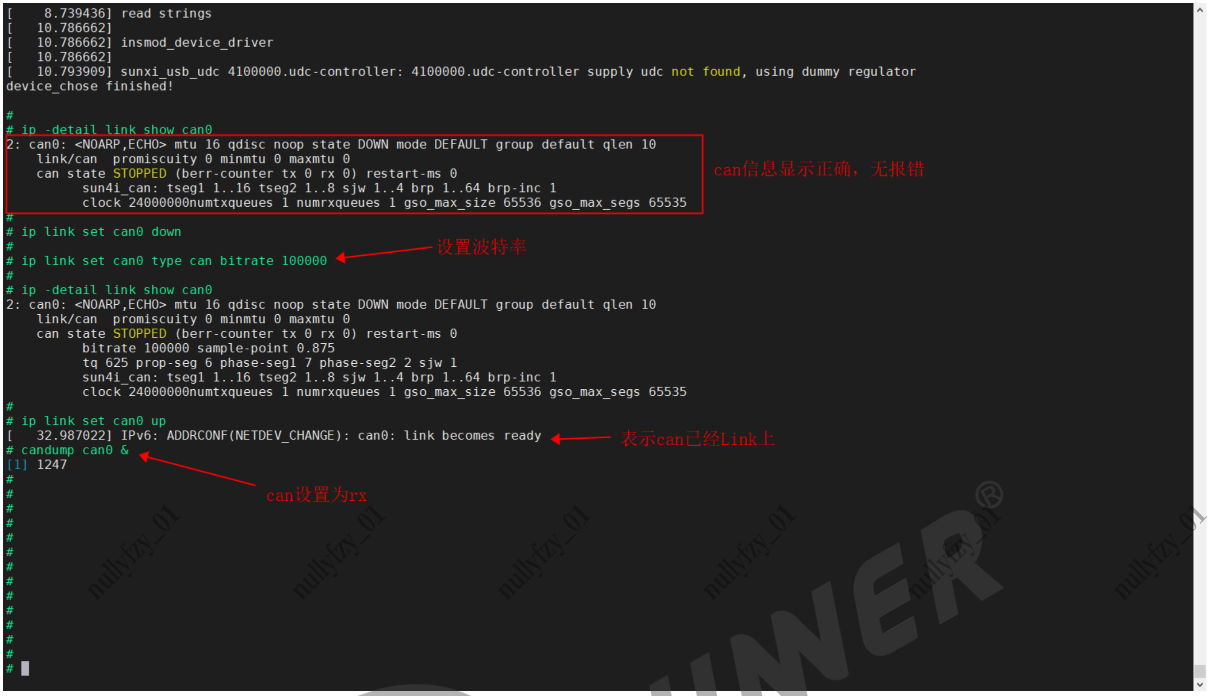
准备两块板子 A 和 B，A 号版子作为 TX，B 号板子作为 RX。

A 号板子 CAN_H 连接到 B 号板子 CAN_H，A 号板子 CAN_L 连接到 B 号板子 CAN_L。

在 B 号板子输入命令：

```
ip -detail link show can0

ip link set can0 down

ip link set can0 type can bitrate 100000

ip -detail link show can0
ip link set can0 up

candump can0 &
```

对应 log 如下



```
[    8.739436] read strings
[   10.786662]
[   10.786662] insmod_device_driver
[   10.786662]
[   10.793909] sunxi_usb_udc 4100000.udc-controller: 4100000.udc-controller supply udc not found, using dummy regulator
device_chose finished!

#
# ip -detail link show can0
2: can0: <NOARP,ECHO> mtu 16 qdisc noop state DOWN mode DEFAULT group default qlen 10
    link/can  promiscuity 0 minmtu 0 maxmtu 0
    can state STOPPED (berr-counter tx 0 rx 0) restart-ms 0
          sun4i_can: tseg1 1..16 tseg2 1..8 sjw 1..4 brp 1..64 brp-inc 1
          clock 24000000numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max_segs 65535
#
# ip link set can0 down
#
# ip link set can0 type can bitrate 100000
#
# ip -detail link show can0
2: can0: <NOARP,ECHO> mtu 16 qdisc noop state DOWN mode DEFAULT group default qlen 10
    link/can  promiscuity 0 minmtu 0 maxmtu 0
    can state STOPPED (berr-counter tx 0 rx 0) restart-ms 0
          bitrate 100000 sample-point 0.875
          tq 625 prop-seg 6 phase-seg1 7 phase-seg2 2 sjw 1
          sun4i_can: tseg1 1..16 tseg2 1..8 sjw 1..4 brp 1..64 brp-inc 1
          clock 24000000numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max_segs 65535
#
# ip link set can0 up
[   32.987022] IPv6: ADDRCONF(NETDEV_CHANGE): can0: link becomes ready
# candump can0 &
[1] 1247
#
#
#
#
#
#
#
#
#
#
#
#
#
# 
```

can信息显示正确，无报错

设置波特率

表示can已经Link上

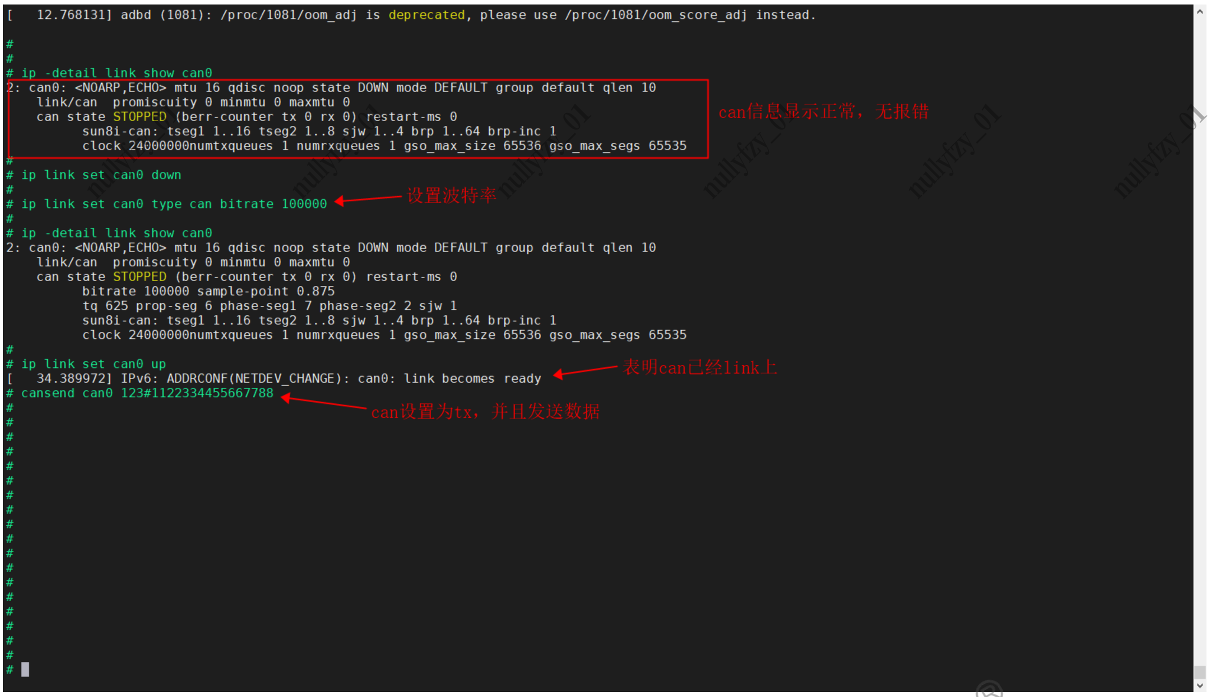
can设置为rx

图 3-4: RX 的 log

在 A 号板子输入命令：

```
ip -detail link show can0

ip link set can0 down

ip link set can0 type can bitrate 100000

ip -detail link show can0

ip link set can0 up

cansend can0 123#1122334455667788
```

对应 log 如下



```
[   12.768131] adbd (1081): /proc/1081/oom_adj is deprecated, please use /proc/1081/oom_score_adj instead.

#
#
# ip -detail link show can0
2: can0: <NOARP,ECHO> mtu 16 qdisc noop state DOWN mode DEFAULT group default qlen 10
    link/can  promiscuity 0 minmtu 0 maxmtu 0
    can state STOPPED (berr-counter tx 0 rx 0) restart-ms 0
          sun8i-can: tseg1 1..16 tseg2 1..8 sjw 1..4 brp 1..64 brp-inc 1
          clock 24000000numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max_segs 65535
#
# ip link set can0 down
#
# ip link set can0 type can bitrate 100000
#
# ip -detail link show can0
2: can0: <NOARP,ECHO> mtu 16 qdisc noop state DOWN mode DEFAULT group default qlen 10
    link/can  promiscuity 0 minmtu 0 maxmtu 0
    can state STOPPED (berr-counter tx 0 rx 0) restart-ms 0
          bitrate 100000 sample-point 0.875
          tq 625 prop-seg 6 phase-seg1 7 phase-seg2 2 sjw 1
          sun8i-can: tseg1 1..16 tseg2 1..8 sjw 1..4 brp 1..64 brp-inc 1
          clock 24000000numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max_segs 65535
#
# ip link set can0 up
[   34.389972] IPv6: ADDRCONF(NETDEV_CHANGE): can0: link becomes ready
# cansend can0 123#1122334455667788
#
#
#
#
#
#
#
#
#
#
#
#
#
#
#
#
#
#
# ▮
```

can信息显示正常，无报错

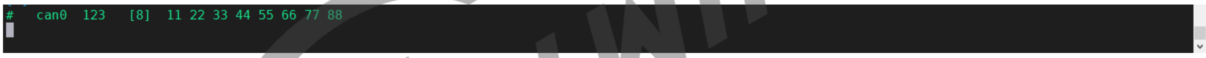
设置波特率

表明can已经link上

can设置为tx，并且发送数据

图 3-5: TX 的 log

然后我们可以在 B 号板子看到收到信息，和 A 板子发送的数据一样



```
#   can0  123   [8]  11 22 33 44 55 66 77 88
▮
```

图 3-6: Device Drivers

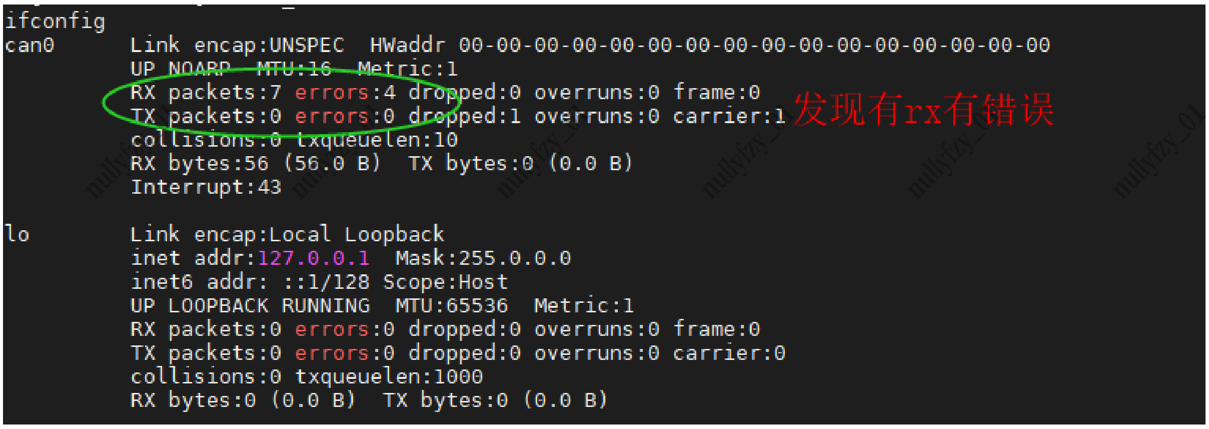
3.4 常见问题

3.4.1 情况一

CAN 发送端发送数据，发现对端设备无法接收到数据，查看 CAN 发送端 log，发现总线提示报错：

```
sun8i-can 0x0.can can0: bus-off
```

然后查看 CAN 发送端状态，发现发送数据后，收到应答信息，但是判断为错误信息



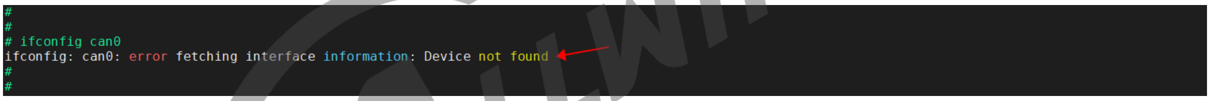
```
ifconfig
can0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          UP NOARP  MTU:16  Metric:1
          RX packets:7 errors:4 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:1 overruns:0 carrier:1
          collisions:0 txqueuelen:10
          RX bytes:56 (56.0 B)  TX bytes:0 (0.0 B)
          Interrupt:43

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

图 3-7: bus-off

根据以上情况一般为板子上 CAN 收发器，或者线路上阻抗匹配存在问题。后由硬件同事确认为 CAN 收发器问题，修改硬件电路后解决。

3.4.2 情况二

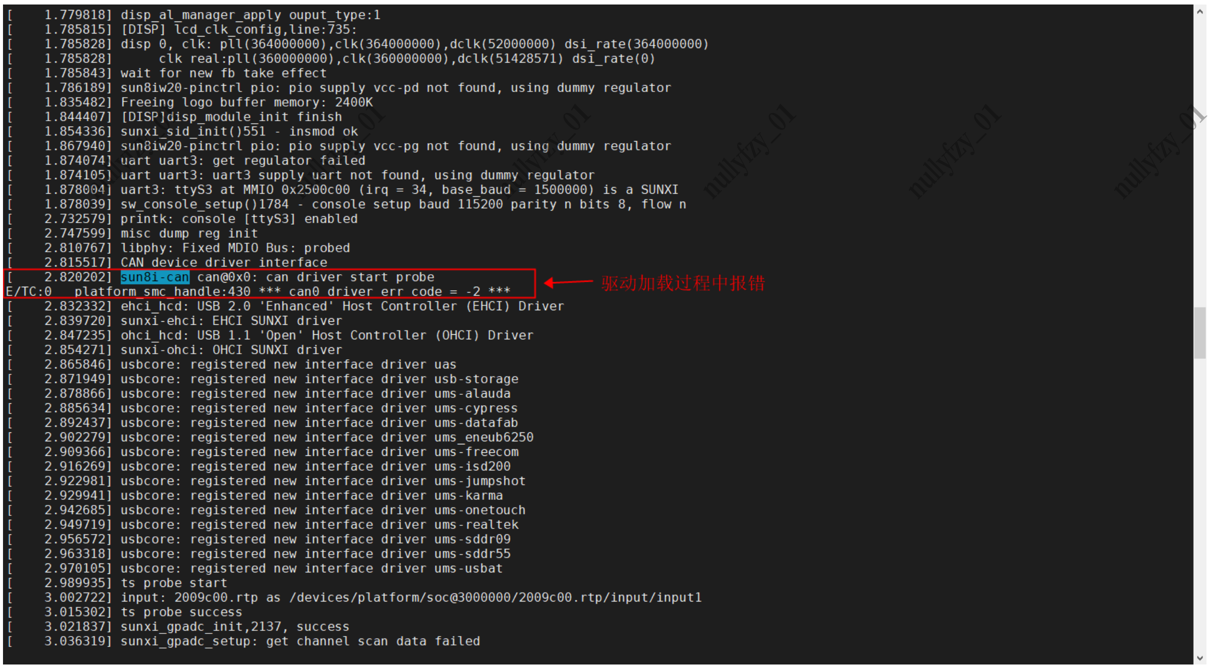
终端查看 can0，发现无法找到 can0，此时有可能出现 2 种情况



```
#
#
# ifconfig can0
ifconfig: can0: error fetching interface information: Device not found
#
#
```

图 3-8: 查找 can0 失败

- 第一种情况，查看启动 log，假如出现以下报错，**表明改芯片不支持 CAN 功能。**



```
[    1.779818] disp_al_manager_apply ouput_type:1
[    1.785815] [DISP] lcd_clk_config,line:735:
[    1.785828] disp 0, clk: pll(364000000),clk(364000000),dclk(52000000) dsi_rate(364000000)
[    1.785828]      clk real:pll(360000000),clk(360000000),dclk(51428571) dsi_rate(0)
[    1.785843] wait for new fb take effect
[    1.786189] sun8iw20-pinctrl pio: pio supply vcc-pd not found, using dummy regulator
[    1.835482] Freeing logo buffer memory: 2400K
[    1.844407] [DISP]disp_module_init finish
[    1.854336] sunxi_sid_init()551 - insmod ok
[    1.867940] sun8iw20-pinctrl pio: pio supply vcc-pg not found, using dummy regulator
[    1.874074] uart uart3: get regulator failed
[    1.874105] uart uart3: uart3 supply uart not found, using dummy regulator
[    1.878004] uart3: ttyS3 at MMIO 0x2500c00 (irq = 34, base_baud = 1500000) is a SUNXI
[    1.878039] sw_console_setup()1784 - console setup baud 115200 parity n bits 8, flow n
[    2.732579] printk: console [ttyS3] enabled
[    2.747599] misc dump reg init
[    2.810767] libphy: Fixed MDIO Bus: probed
[    2.815517] CAN device driver interface
[    2.820202] sun8i-can can@0x0: can driver start probe
E/TC:0   platform_smc_handle:430 *** can0 driver err code = -2 ***
[    2.832332] ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver
[    2.839720] sunxi-ehci: EHCI SUNXI driver
[    2.847235] ohci_hcd: USB 1.1 'Open' Host Controller (OHCI) Driver
[    2.854271] sunxi-ohci: OHCI SUNXI driver
[    2.865846] usbcore: registered new interface driver uas
[    2.871949] usbcore: registered new interface driver usb-storage
[    2.878866] usbcore: registered new interface driver ums-alauda
[    2.885634] usbcore: registered new interface driver ums-cypress
[    2.892437] usbcore: registered new interface driver ums-datafab
[    2.902279] usbcore: registered new interface driver ums_eneub6250
[    2.909366] usbcore: registered new interface driver ums-freecom
[    2.916269] usbcore: registered new interface driver ums-isd200
[    2.922981] usbcore: registered new interface driver ums-jumpshot
[    2.929941] usbcore: registered new interface driver ums-karma
[    2.942685] usbcore: registered new interface driver ums-onetouch
[    2.949719] usbcore: registered new interface driver ums-realtek
[    2.956572] usbcore: registered new interface driver ums-sddr09
[    2.963318] usbcore: registered new interface driver ums-sddr55
[    2.970105] usbcore: registered new interface driver ums-usbat
[    2.989935] ts probe start
[    3.002722] input: 2009c00.rtp as /devices/platform/soc@3000000/2009c00.rtp/input/input1
[    3.015302] ts probe success
[    3.021837] sunxi_gpadc_init,2137, success
[    3.036319] sunxi_gpadc_setup: get channel scan data failed
```

图 3-9: 驱动加载报错

- 第二种情况，查看启动 log，假如无任何 can 驱动加载信息，表明驱动没有加载，**请查看《模块配置介绍》章节，核对配置是否正确。**

3.4.3 情况三

打开 can0 时候，出现以下报错:

```
# ip link set can0 type can bitrate 1000000
ip: either "dev" is duplicate, or "type" is garbage
# ifconfig can0 up
[ 1647.050606] sun4i_can 1c2bc00.can can0: bit-timing not yet defined
ifconfig: SIOCSIFFLAGS: Invalid argument
<3>[ 1647.050606] sun4i_can 1c2bc00.can can0: bit-timing not yet defined
```

ip 工具相关依赖配置没有选择正确，请查看《模块配置介绍》章节，核对配置是否正确。

著作权声明

本文档及内容受著作权法保护，其著作权由珠海全志科技股份有限公司（“全志”）拥有并保留一切权利。

本文档是全志的原创作品和版权财产，未经全志书面许可，任何单位和个人不得擅自摘抄、复制、修改、发表或传播本文档内容的部分或全部，且不得以任何形式传播。

商标声明

、、、（不完全列举）均为珠海全志科技股份有限公司的商标或者注册商标。在本文档描述的产品中出现的其它商标，产品名称，和服务名称，均由其各自所有人拥有。

免责声明

您购买的产品、服务或特性应受您与珠海全志科技股份有限公司（“全志”）之间签署的商业合同和条款的约束。本文档中描述的全部或部分产品、服务或特性可能不在您所购买或使用的范围内。使用前请认真阅读合同条款和相关说明，并严格遵循本文档的使用说明。您将自行承担任何不当使用行为（包括但不限于如超压，超频，超温使用）造成的不利后果，全志概不负责。

本文档作为使用指导仅供参考。由于产品版本升级或其他原因，本文档内容有可能修改，如有变更，恕不另行通知。全志尽全力在本文档中提供准确的信息，但并不确保内容完全没有错误，因使用本文档而发生损害（包括但不限于间接的、偶然的、特殊的损失）或发生侵犯第三方权利事件，全志概不负责。本文档中的所有陈述、信息和建议并不构成任何明示或暗示的保证或承诺。

本文档未以明示或暗示或其他方式授予全志的任何专利或知识产权。在您实施方案或使用产品的过程中，可能需要获得第三方的权利许可。请您自行向第三方权利人获取相关的许可。全志不承担也不代为支付任何关于获取第三方许可的许可费或版税（专利税）。全志不对您所使用的第三方许可技术做出任何保证、赔偿或承担其他义务。