

Advantech AE Technical Share Document

Date	2018 / 08 / 07	Release Note	<input type="checkbox"/> Internal <input checked="" type="checkbox"/> External
Category	<input checked="" type="checkbox"/> FAQ <input type="checkbox"/> SOP	Related OS	AdvLinuxTU
Abstract	How to test CAN function under AdvLinuxTU?		
Keyword	AdvLinuxTU / PCM-26D2CA / CANBus / CANOpen		
Related Product	PCM-26D2CA		

■ **Problem Description:**

This document describes the way to test CAN port with Advantech CAN bus communication card on AdvLinux. By connecting the wires correctly, those two CAN port testing can help user to verify the operation of CAN ports by using Advantech Linux CAN example

■ **Brief Solution:**

H/W condition --- UNO-2271G + PCM-26D2CA-AE + AdvLinuxTU 1.0.8(**already included the PCM-26D2CA driver**)

A. Connect with each CAN_L, CAN_H pin, as Figure 1:

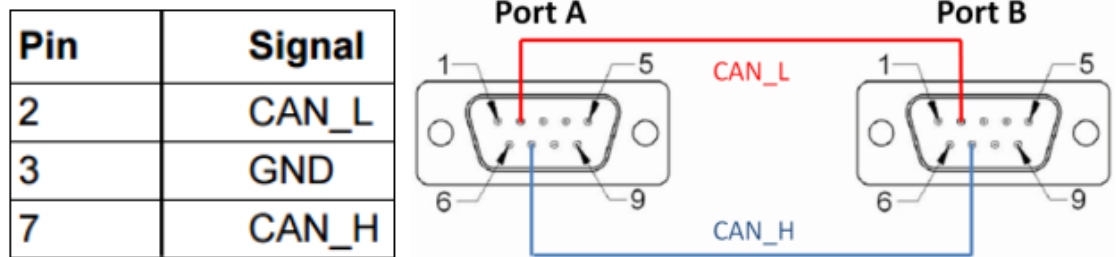


Figure 1. Connect with CAN_L & CAN_H of Port A & Port B.

- B. Press Ctrl+Alt+T to open terminal, type "su" and enter the password to change to root.
- C. Type "cd /" change to the folder
Type "cd dev" then type "ls" to list all the device.
Make sure you can find the CAN device in the list as below:

```

Terminal - localhost@localhost-UNO-2271G-E2xAE: /dev
File Edit View Terminal Tabs Help
localhost@localhost-UNO-2271G-E2xAE:~$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
localhost@localhost-UNO-2271G-E2xAE:~$ cd /
localhost@localhost-UNO-2271G-E2xAE:/ $ ls
bin  cdrom  etc  lib  lost+found  mnt  proc  run  snap  sys  usr
boot  dev  home  lib64  media  opt  root  sbin  srv  tmp  var
localhost@localhost-UNO-2271G-E2xAE:/ $ cd dev
localhost@localhost-UNO-2271G-E2xAE:/dev$ ls
adspname  hugepages  port  tty15  tty50  ttyS27
adveeprom  hwrng  ppp  tty16  tty51  ttyS28
autofs  i2c-0  psaux  tty17  tty52  ttyS29
block  i2c-1  ptmx  tty18  tty53  ttyS3
btrfs-control  i2c-2  pts  tty19  tty54  ttyS30
bus  i2c-3  ram0  tty2  tty55  ttyS31
can0  i2c-4  ram1  tty20  tty56  ttyS4
can1  i2c-5  ram10  tty21  tty57  ttyS5
can10  i2c-6  ram11  tty22  tty58  ttyS6
can11  initctl  ram12  tty23  tty59  ttyS7
can12  input  ram13  tty24  tty6  ttyS8
can13  kmsg  ram14  tty25  tty60  ttyS9
can14  kvm  ram15  tty26  tty61  uhid
can15  lightnvm  ram2  tty27  tty62  uinput
can2  log  ram3  tty28  tty63  urandom
can3  loop0  ram4  tty29  tty7  userio
can4  loop1  ram5  tty3  tty8  vcs
can5  loop2  ram6  tty30  tty9  vcs1
can6  loop3  ram7  tty31  ttyprintk  vcs2
can7  loop4  ram8  tty32  ttyS0  vcs3
can8  loop5  ram9  tty33  ttyS1  vcs4
can9  loop6  random  tty34  ttyS10  vcs5
char  loop7  rfkill  tty35  ttyS11  vcs6
console  loop-control  rtc  tty36  ttyS12  vcs7
core  mapper  rtc0  tty37  ttyS13  vcsa
  
```

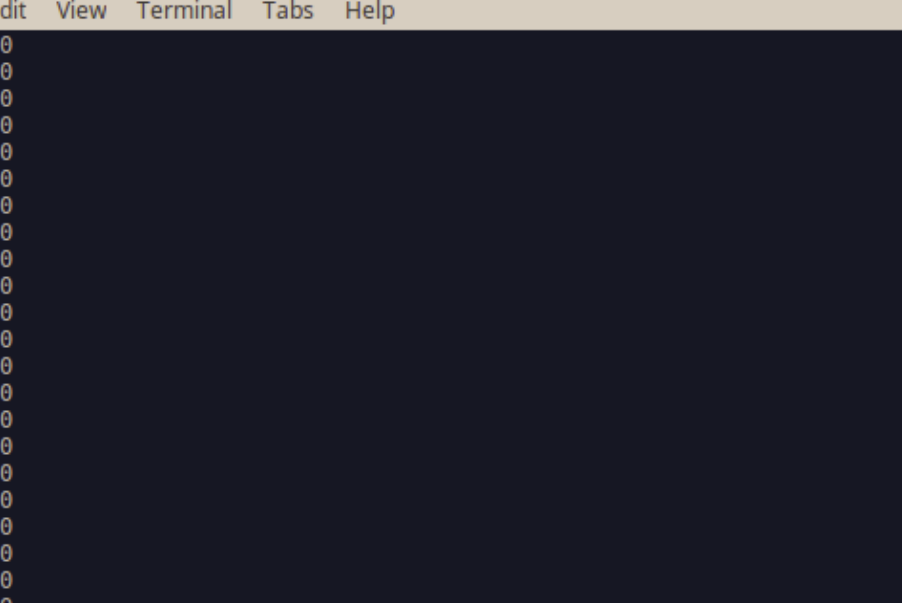
- D.** Type “cd /” again back to the main folder.
Type “cd usr/src/Advantech/advcan/examples/”

```

Terminal - root@localhost-UNO-2271G-E2xAE: /usr/src/advantech/advcan/advcan/examples - + x
File Edit View Terminal Tabs Help
localhost@localhost-UNO-2271G-E2xAE:~$ su
Password:
root@localhost-UNO-2271G-E2xAE:/home/localhost# ls
Desktop Documents Downloads Music Pictures Public Templates Videos
root@localhost-UNO-2271G-E2xAE:/home/localhost# cd /
root@localhost-UNO-2271G-E2xAE:/# cd usr/src/advantech/advcan/examples/
root@localhost-UNO-2271G-E2xAE:/usr/src/advantech/advcan/examples# ls
acceptance      receive-block      selfreception.c    transmit-block
acceptance.c    receive-block.c    send-ioctl          transmit-block.c
baud            receive-nonblock   send-ioctl.c        transmit-nonblock
baud.c          receive-nonblock.c showstat            transmit-nonblock.c
can4linux.h     receive-select     showstat.c          transmit-select
logfile.txt     receive-select.c   singlefilter        transmit-select.c
Makefile        selfreception      singlefilter.c
root@localhost-UNO-2271G-E2xAE:/usr/src/advantech/advcan/examples#

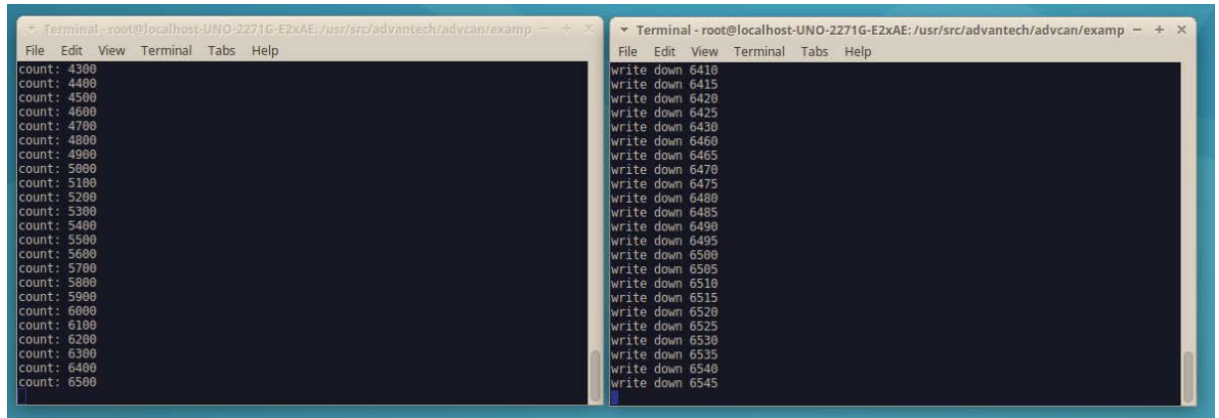
```

- E.** Type “./receive-select can0”.
The count is 0 because we have not transmitted any data yet.



The screenshot shows a terminal window with a title bar that reads "Terminal - root@localhost-UNO-2271G-E2xAE: /usr/src/advantech/advcan/examp". Below the title bar is a menu bar with the options "File", "Edit", "View", "Terminal", "Tabs", and "Help". The main area of the terminal is dark blue and contains 20 lines of text, each reading "count: 0". A blue cursor is visible at the end of the 20th line. The terminal window has a light gray border and a small vertical scrollbar on the right side.

- F. Open another terminal and use root to enter the same folder again:
usr/src/Advantech/advcan/examples/
- G. Type “./transmit-select can1”
- H. If everything works fine, two terminals will be like this:



```
Terminal - root@localhost-UNO-2271G-E2xAE: /usr/src/advantech/advcan/examp
File Edit View Terminal Tabs Help
count: 4300
count: 4400
count: 4500
count: 4600
count: 4700
count: 4800
count: 4900
count: 5000
count: 5100
count: 5200
count: 5300
count: 5400
count: 5500
count: 5600
count: 5700
count: 5800
count: 5900
count: 6000
count: 6100
count: 6200
count: 6300
count: 6400
count: 6500

Terminal - root@localhost-UNO-2271G-E2xAE: /usr/src/advantech/advcan/examp
File Edit View Terminal Tabs Help
write down 6410
write down 6415
write down 6420
write down 6425
write down 6430
write down 6460
write down 6465
write down 6470
write down 6475
write down 6480
write down 6485
write down 6490
write down 6495
write down 6500
write down 6505
write down 6510
write down 6515
write down 6520
write down 6525
write down 6530
write down 6535
write down 6540
write down 6545
```

- I. Now, you can switch the transmit port and receive port then test it again.
- J. As above test result, if both way can work well with transmit & receive example, that mean these CAN card function is fine..