FUJITSU TEN

How To Use Linux CAN Signal To AGL

02/08/2017
Yuichi Kusakabe
SS Engineering Group
Fujitsu TEN LIMITED

About Myself

FUJITSU TEN

- Yuichi Kusakabe (Fujitsu TEN LIMITED)

- Software Engineer of IVI about 10 years
  (for 16-bit and 32-bit architecture)

- Linux Software Engineer(2011-2013)
- Linux Software Lead Engineer(2013-Now)

- BSP Porting/Customizing
- Supporting for in-house software developers

Agenda

FUJITSU TEN

- What's CAR CAN signal
- Standard Linux CAN IF & OSS CAN Tool
- How to use CAN signal to AGL
- Demonstration & Results
- Conclusion

What’s CAR CAN Signal

What's CAR CAN Signal

FUJITSU TEN

Standard CAN Signal is Low Speed (500kbps), But High frequency (us).**

- Standard CAN Signal format(11bit).
  - Data line: D+/D-/GND(want)
  - Baud rate: 500kbps
  - CAN ID: 11bit(0x000~0x7FF)
  - Data size: 0~8byte
  - CAN Bus load: 20~75%

https://ja.wikipedia.org/wiki/Controller_Area_Network

Standard Linux CAN IF & OSS CAN Tool

Standard Linux CAN IF(SocketCAN)

FUJITSU TEN

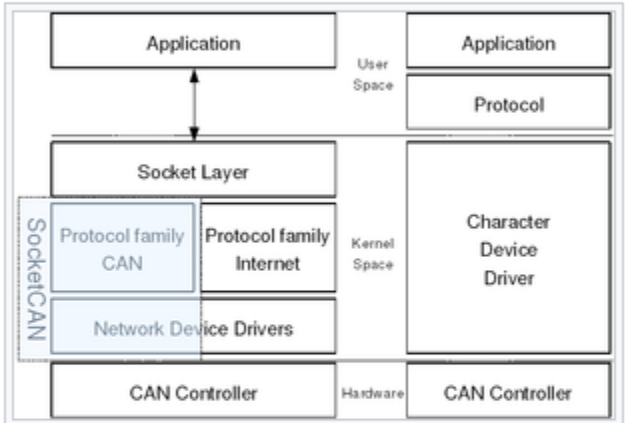
Linux kernel all ready CAN IF with Socket CAN

SocketCAN

From Wikipedia, the free encyclopedia

SocketCAN is a set of open source CAN drivers and a networking stack contributed by Volkswagen Research to the Linux kernel. Formerly known as *Low Level CAN Framework* (LLCF).

Traditional CAN drivers for Linux are based on the model of character devices. Typically they only allow sending to and receiving from the CAN controller. Conventional implementations of this class of device driver only allow a single process to access the device, which means that all other processes are blocked in the meantime. In addition, these drivers typically all differ slightly in the interface presented to the application, stifling portability. The SocketCAN concept on the other hand uses the model of network devices, which allows multiple applications to access one CAN device simultaneously. Also, a single application is able to access multiple CAN networks in parallel.



Typical CAN communication layers. With SocketCAN (left) or conventional (right).

The SocketCAN concept extends the Berkeley sockets API in Linux by introducing a new protocol family, PF_CAN, that coexists with other protocol families like PF_INET for the Internet Protocol. The communication with the CAN bus is therefore done analogously to the use of the Internet Protocol via sockets. Fundamental components of SocketCAN are the network device drivers for different CAN controllers and the implementation of the CAN protocol family. The protocol family, PF_CAN, provides the structures to enable different protocols on the bus: Raw sockets for direct CAN communication and transport protocols for point-to-point connections. Moreover the broadcast manager which is part of the CAN protocol family provides functions e.g. for sending CAN messages periodically or realize complex message filters.

Patches about CAN were added in the 2.6.25 Linux kernel. Meanwhile some controller drivers were added and work is going on to add drivers for a variety of controllers.

https://en.wikipedia.org/wiki/SocketCAN

Standard Linux CAN IF(CAN Driver)

FUJITSU TEN

Linux kernel all ready CAN IF with Socket CAN

```
Readme file for the Controller Area Network Protocol Family (aka SocketCAN)

This file contains

  1 Overview / What is SocketCAN

  2 Motivation / Why using the socket API

  3 SocketCAN concept
    3.1 receive lists
    3.2 local loopback of sent frames
    3.3 network problem notifications

  4 How to use SocketCAN
    4.1 RAW protocol sockets with can_filters (SOCK_RAW)
      4.1.1 RAW socket option CAN_RAW_FILTER
      4.1.2 RAW socket option CAN_RAW_ERR_FILTER
      4.1.3 RAW socket option CAN_RAW_LOOPBACK
      4.1.4 RAW socket option CAN_RAW_RECV_OWN_MSGS
      4.1.5 RAW socket option CAN_RAW_FD_FRAMES
      4.1.6 RAW socket option CAN_RAW_JOIN_FILTERS
      4.1.7 RAW socket returned message flags
    4.2 Broadcast Manager protocol sockets (SOCK_DGRAM)
      4.2.1 Broadcast Manager operations
      4.2.2 Broadcast Manager message flags
      4.2.3 Broadcast Manager transmission timers
      4.2.4 Broadcast Manager message sequence transmission
      4.2.5 Broadcast Manager receive filter timers
      4.2.6 Broadcast Manager multiplex message receive filter
      4.2.7 Broadcast Manager CAN FD support
    4.3 connected transport protocols (SOCK_SEQPACKET)
    4.4 unconnected transport protocols (SOCK_DGRAM)

  5 SocketCAN core module
    5.1 can.ko module params
    5.2 procfs content
    5.3 writing own CAN protocol modules
```

```
CONFIG_CAN=y
CONFIG_CAN_RAW=y
CONFIG_CAN_BCM=y
CONFIG_CAN_GW=y
CONFIG_CAN_RCAR=y
```

https://www.kernel.org/doc/Documentation/networking/can.txt

OSS CAN Tool(Powerful software)

FUJITSU TEN

can-utils easy to debug CAN Signal (read/write/play)

linux-can / can-utils

Unwatch 60 | Star 229 | Fork 110

Code | Issues 1 | Pull requests 2 | Projects 0 | Wiki | Pulse | Graphs

Linux-CAN / SocketCAN user space applications

300 commits | 2 branches | 0 releases | 18 contributors

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

jihochu committed with hartkopp bcmserver: allow CAN netdevice names greater than 6 characters ... Latest commit 99f1664 19 days ago

config/m4	add autotools infrastructure	7 years ago
include/linux	bcm: add support for CAN FD frames	4 months ago
.gitignore	gitignore: add isotpperf	2 years ago
Android.mk	can-utils: added isotpperf tool for performance measurements	2 years ago
GNUmakefile.am	configure: switch to new libtool-2.0 macro	2 years ago
Makefile	can-utils: added isotpperf tool for performance measurements	2 years ago
README.md	Create README.md	a year ago
asc2log.c	janitorial: asc2log: properly close infile	2 years ago
autogen.sh	do not use --symlink for autoreconf	3 years ago

https://github.com/linux-can/can-utils

OSS CAN Tool(Powerful software)

FUJITSU TEN

CAN data send (cansend)

```
ID=333(11bit), DATA=33 send=can0
# cansend can0 333#33

ID=00004444(24bit), DATA=44 send=can0
# cansend can0 00004444#44
```

CAN data recv (candump)

```
recv=can
# candump can0 -ta
root@porter:~# candump can0 -ta
 (1478869757.430017)  can0  344   [8]  FF EE 00 00 00 00 EE AA
 (1478869757.431290)  can0  226   [8]  E4 00 00 EE 00 EE EE 00

recv=all
# candump any -ta
```

https://github.com/linux-can/can-utils

How to use CAN signal to AGL

AMB(Automotive Message Broker)

FUJITSU TEN

This time AGL provide AMB, but AGL remake new CAN Signal handing FW.

otcshare / automotive-message-broker

Watch 25 | Star 33 | Fork 34

Code | Issues 3 | Pull requests 0 | Projects 0 | Wiki | Pulse | Graphs

No description, website, or topics provided.

1,277 commits | 22 branches | 352 releases | 12 contributors

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

tripzero committed on GitHub Merge pull request #86 from again4you/fix_warning_v2 ... Latest commit 8af1cc9 on 4 Jan

ambd	fixed the warning message when enabling -Wall option	a month ago
docs	[amb.fidl] fix Ignition type and some whitespace issues	6 months ago
examples	Added chrony sink plugin	2 years ago
lib	fixed the warning message when enabling -Wall option	a month ago
packaging.in	[PACKAGING] add unpackaged files into rpm package	6 months ago
plugins	fixed the warning message when enabling -Wall option	a month ago
tests	Merge pull request #49 from tripzero/master	2 years ago
tools	fixed the warning message when enabling -Wall option	a month ago
xwalk	[bluemonkey] websocket server implemented	2 years ago
CMakeLists.txt	fixed the warning message when enabling -Wall option	a month ago
COPYING	added COPYING file with license	5 years ago

https://github.com/otcshare/automotive-message-broker

AMB Plugins

FUJITSU TEN

AMB provide simple plugin only, default is not use SocketCAN

Branch: master ▾ automotive-message-broker / plugins / Create new file | Upload files | Find file | History

again4you fixed the warning message when en

..	
bluemonkey	ble r
bluetooth	[AMB
cangenplugin	Impl
cansimplugin	fixed
chrony	Adde
common	fixed
database	[Data
dbus	cmak
demosink	rever
gpsnmea	fixed
murphyplugin	rever
obd2plugin	[PAC
opencvlux	[amb
openxc	mult
testplugin	fixed
websocket	[AMB
wheel	fixed

Branch: master ▾ automotive-message-broker / plugins / common / Create new file | Upload files | Find file | History

again4you fixed the warning message when enabling -Wall option ... Latest commit 5b9c641 on 22 Dec 2016

..		
CMakeLists.txt	Merge pull request #48 from CogentEmbedded/master	2 years ago
abstractdbusinterface.cpp	fixed the warning message when enabling -Wall option	a month ago
abstractdbusinterface.h	override org.freedesktop.DBus.Properties so we can handle errors prop...	2 years ago
abstractio.hpp	[AMBClient] - fixed json stream handling	2 years ago
amb-plugins-common.pc.in	cmake: fix the error in pkgconfig file	4 months ago
bluetooth.hpp	fixed the warning message when enabling -Wall option	a month ago
bluetooth5.cpp	fixed the warning message when enabling -Wall option	a month ago
bluetooth5.h	[Packaging][Deb] - More packaging	2 years ago
bluetoothadapterproxy.c	added common library for plugins	4 years ago
bluetoothadapterproxy.h	added common library for plugins	4 years ago
bluetoothmanagerproxy.c	added common library for plugins	4 years ago
bluetoothmanagerproxy.h	added common library for plugins	4 years ago
bluetoothserialproxy.c	added common library for plugins	4 years ago
bluetoothserialproxy.h	added common library for plugins	4 years ago
canadapter.cpp	added cansim, cangen plugins	3 years ago
canadapter.h	Updated comments and fixed tabbing	2 years ago
canbus.h	Updated comments and fixed tabbing	2 years ago
canbusimpl.cpp	Implement CAN_BCM support (SocketCAN)	2 years ago
canbusimpl.h	Updated comments and fixed tabbing	2 years ago

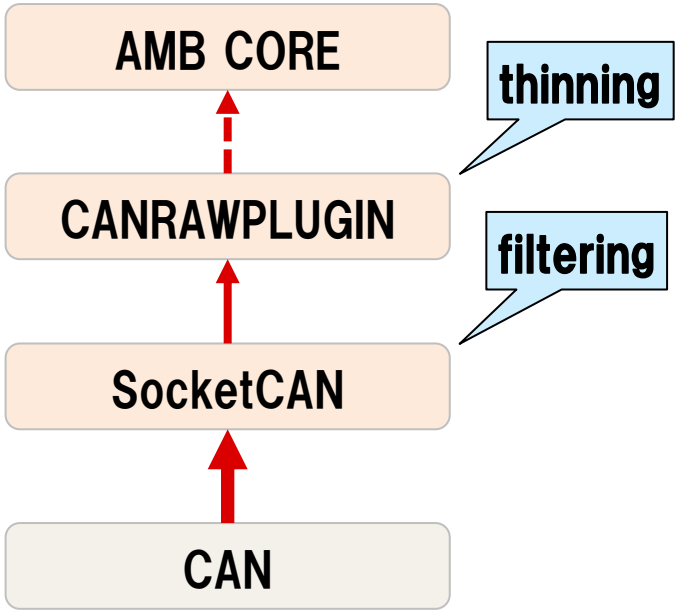
https://github.com/otcshare/automotive-message-broker

What's CANRAWPLUGIN

FUJITSU TEN

CANRAWPLUGIN is Simple SocketCAN AMB Plugins

- CAN signal filtering(setting SocketCAN)
  - CAN ID xxx -> xx
- CAN signal thinning out
  - CAN cycle xx ms -> xxx ms
- CAN data convert AMB format

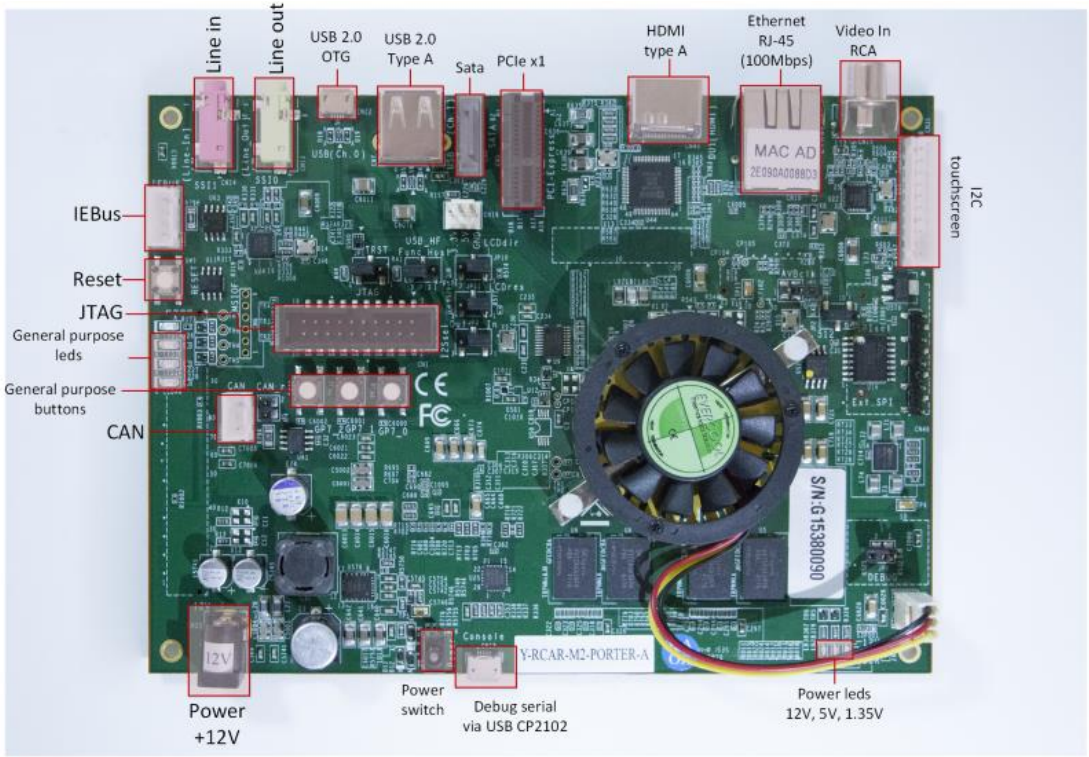


Demonstration & Results

Target Hardware spec

FUJITSU TEN

AGL reference Hardware to Renesas R-CAR M2 Porter board



http://elinux.org/R-Car/Boards/Porter

32GB microSDHC

http://panasonic.jp/sd/p-db/RP-SMGB32GJK.html

- R-Car M2 SoC
  - ARM®Cortex-A15 Dual Core 1.5GHz
  - Multimedia Engine SH4A 780 MHz
  - GPU
    - PowerVR SGX544MP2 (3D)
    - Renesas graphics processor (2D)
- 2 GB DDR3 memory (dual channel)
- Two flash memory chips
  - 4 MB SPI
  - 64 MB SPI
- Debug Ethernet (100 Mbps)
- Storage connection
  - one SATA rev. 3.1 port
  - one SD card slot
  - one microSD card slot
- Analog Video In: ADV7180 Video Decoder
  - RCA jack
  - NTSC/PAL/SECAM autodetection
- Audio codec: AK4643EN
  - Line In 3.5 mm jack
  - LineOut 3.5 mm jack
- Two USB 2.0 ports
  - microUSB port supports host, device and OTG modes
- PCI Express x1 slot
- CAN transceiver

Target Hardware spec

FUJITSU TEN

CAN simulator running to AGL reference Hardware to Renesas R-CAR M3

R-Car M2

Cortex-A15 (VFP, NEON)	Cortex-A15 (VFP, NEON)
SH-4A (CPU, FPU)	
Snoop Controller	DDR3 I/F(800MHz) (32bit bus)(2ch)
L2 Cache	Display Out (2ch)
Video Accelerator (H.264, MPEG2/4, VC-1)	3D Graphics Processor (SGX544MP2)
Audio DSP (AAC, MP3, WMA)	Renesas 2D Graphics Processor
Sound Routing Unit (10ch x SSI, SRC)	TS-IF
Video Capture (3ch)	USB 2.0 Host (2ch)
Distortion Compensation Module	USB 3.0 Host
PCI-Express	SD Card Host I/F (3ch)
Serial ATA	MMC I/F
GPS B/B	High Speed Serial I/F (3ch)
CAN (2ch)	Ethernet AVB
MOST I/F	
DARC	

http://elinux.org/R-Car/Boards/

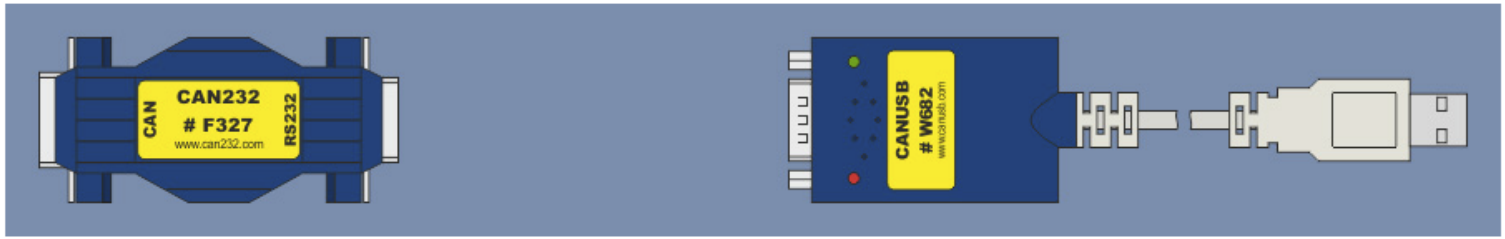
Target Hardware spec

FUJITSU TEN

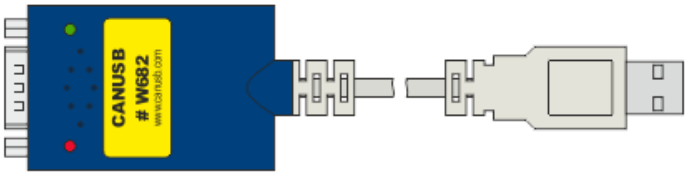
R-CAR M3 not include CAN IF, CANUSB easy connect CANIF

CAN Tools

Home News Products Projects Downloads Where to Buy? About



CANUSB



General Information:

CANUSB is a very small dongle that plugs into any PC USB Port and gives an instant CAN connectivity. This means it can be treated by software as a standard COM Port (virtual serial RS232 port) with the FTDI USB drivers which eliminates the need for any extra drivers (DLL) or by installing a direct driver DLL (D2XX) together with our CANUSB DLL for faster communications and higher CAN bus loads. Sending and receiving can be done in standard ASCII format.

Menu

- Home
- News
- Products
  - CAN232
  - CANUSB
- Projects
- Downloads
  - CAN232 Download
  - CANUSB Download
- Where to Buy?
- About

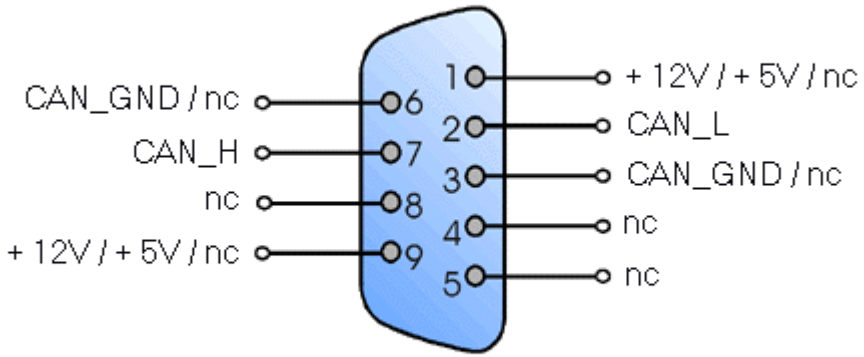
http://www.can232.com/?page_id=16

Target Hardware spec

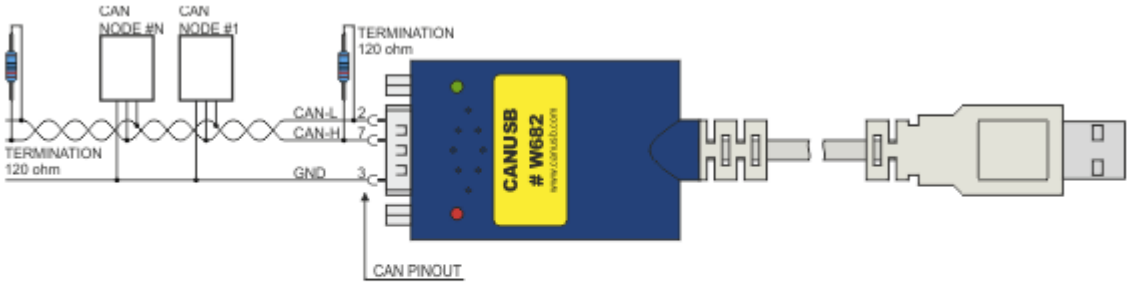
FUJITSU TEN

CANUSB connected CAN IF simple Hardware

CAN Pin assignement:



Pin assignement according to CiA recommendations DS102-1.
The CANUSB is powered from USB port, so no need to connect external power on pin 9. Use only CAN_L (pin2), CAN_H (Pin7) and CAN_GND (pin3).



The picture above shows how to connect the CANUSB (click here for a larger view). No external power is needed, the CANUSB uses 5VDC/100mA from USB.

http://www.can232.com/?page_id=16

Software setup

FUJITSU TEN

Add Kernel defconfig CAN driver and CANUSB

```
CONFIG_CAN=y
CONFIG_CAN_VCAN=y
CONFIG_CAN_RCAR=y  <- Renesas Porter board only
CONFIG_CAN_SLCAN=y
CONFIG_USB_SERIAL=y
CONFIG_USB_SERIAL_FTDI_SIO=y
```

Add rootfs “can-utils” and “iproute2”

```
yocto local.conf
IMAGE_INSTALL_append = “ can-utils iproute2”
```

Setup CAN and CANUSB

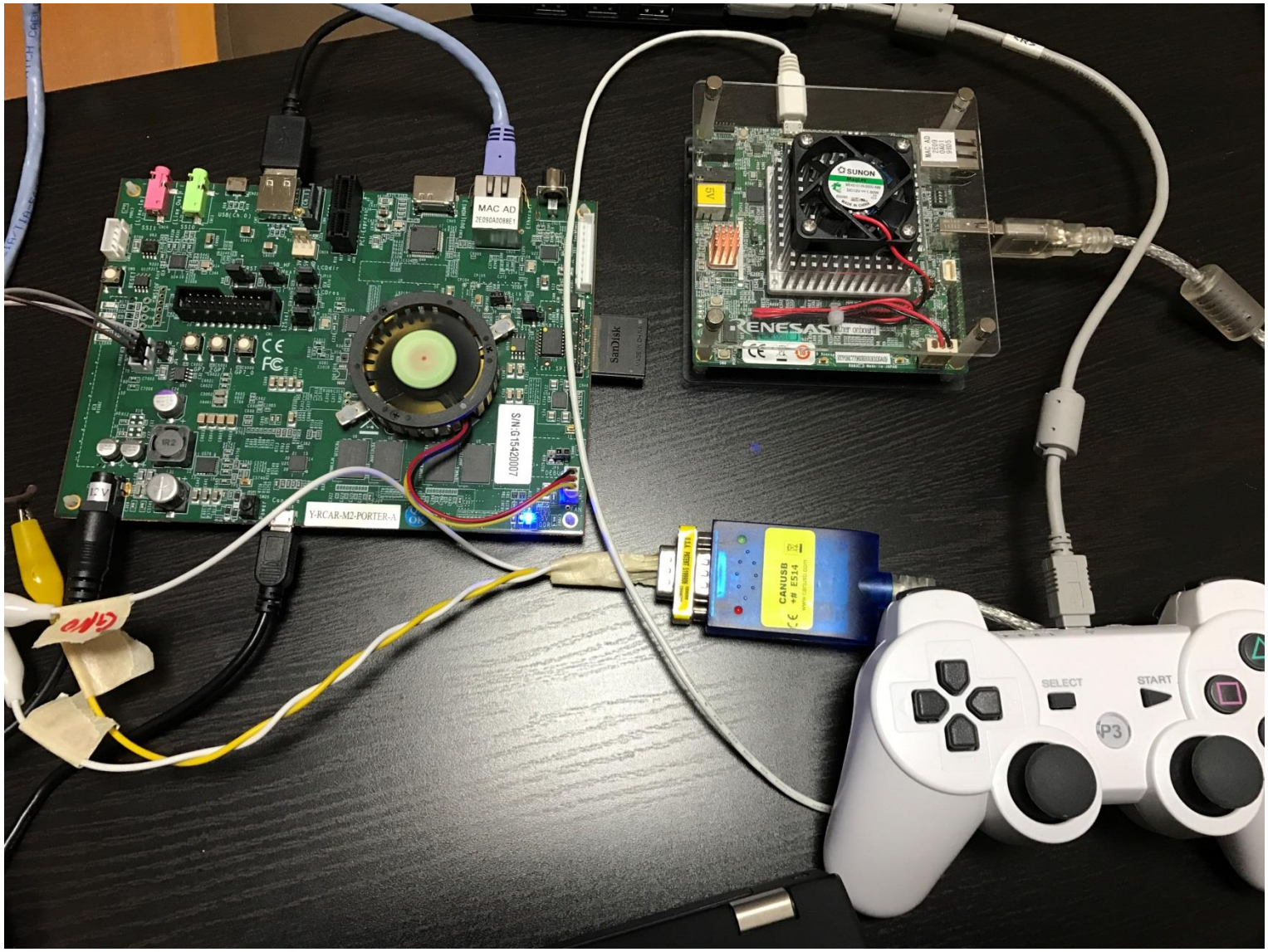
```
CAN0
ip link set can0 type can bitrate 500000
ip link set can0 up
```

```
CANUSB
slcand -o -s 6 -t hw /dev/ttyUSB*
ip link set slcan0 up
```

Demonstration

FUJITSU TEN

Porter board (running AGL CES DEMO), M3 (running CAN simulator) and CANUSB



Results

FUJITSU TEN

- CAN data detail
  - Filtering CAN ID:124 -> 40
  - Thinning out time: **ms -> 100ms

No	CAN Bus load	can id/sec	CPU load (AMB + d-bus)
1	0%	0	6.7%
2	29%	1,447	35.88%
3	43%	2,063	86.29%
4	50%	Unmeasurable ->data lost	Unmeasurable ->data lost

AMB and d-bus is heavy, and small CAN data handing difficult

Conclusion

FUJITSU TEN

- Linux Kernel all ready use to CAN
- OSS CAN Tool “can utils” is good software
- CAN Signal handing resource is difficult
- Next step
  - Define AGL public CAR CAN data format
  - AGL standard CAN simulator
  - New CAN handing FW support SocketCAN

Extra(Japanese How to Use Linux CAN)

FUJITSU TEN

Qiita キーワードを入力 Hot Markdownによる情報共有サ... 投稿する ストック一覧 8

Fujitsu extended Advent Calendar 2016 | 1日目

LinuxでCAN(車載通信)を動作させてみる

Linux 5310 can 14

22 いいね 0 コメント

yuichi-kusakabe 2016年12月22日に更新 2 投稿を編集 989views ストック

この記事は、Fujitsu Advent Calender 2枚目の1日目の記事です。

Tweet B! 1 G+1 0 Like 12 Pocket 5

はじめに

組み込みLinux評価ボードでCAN(車載通信)を使用する方法について調べてみました。
CANには色々な規格がありますが、今回は一般的な500kビット/秒の標準フォーマット(11bit)を使用してみました(CANの詳細はwikipedia等を参照して下さい)。

1.ターゲットボードの用意

手頃な価格でCANを使用出来る組み込み評価ボードが無いか調べてみましたが、SoCがCANに対応していても評価ボードにCAN IFが搭載されていないボードが大半です。今回はAutomotiveGradeLinuxの標準リファレンスボードに指定されているRenesas社のPorterボードを使用してみます。
ボードのCAN IFについては下記に詳しく説明が記載されていました。

yuichi-kusakabe
22 Contribution

人気の投稿
- LinuxでCAN(車載通信)を動作させてみる

http://qiita.com/yuichi-kusakabe/items/e5b50aa3edb712bb6916

Thank you!!!

yuichi.kusakabe@jp.fujitsu.com