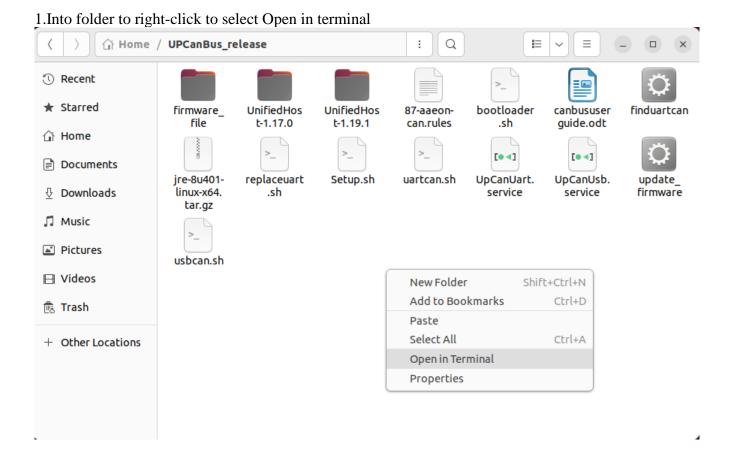
How to install



2. Type command to make sure script could be run.

Command: sudo chmod 777 *

a@a-SYSTEM-PRODUCT-NAME:~/UPCanBus_release\$ sudo chmod 777 * [sudo] password for a:

3.make sure Ethernet is connected

4. Type command to install: ./Setup.sh install

```
a@a-SYSTEM-PRODUCT-NAME:-/UPCanBus_release$ ./Setup.sh install
Host already install can-utils
find canbus uart successfully
replace uart_pid 0x9dc7 to 0x9dc7
Setup file will copy to /opt/aaeon/canbus , if need,you could go there to use
Created symlink /etc/systemd/system/multi-user.target.wants/UpCanUsb.service →/etc/systemd/system/UpCanUsb.service.
Created symlink /etc/systemd/system/multi-user.target.wants/UpCanUart.service →/etc/systemd/system/UpCanUart.service.
You need to reboot to active udev
```

5.After reboot ,you will to use CAN Bus

6.(Optional)Type command to check CAN Bus: ip a

- 4: can1: <NOARP,UP,LOWER_UP> mtu 16 qdisc pfifo_fast state UP group default qlen 10000 link/can
- 5: can0: <NOARP,UP,LOWER_UP> mtu 16 qdisc pfifo_fast state UP group default qlen 10000 link/can

How to Uninstall

1.Open terminal and type command to go to folder : cd /opt/aaeon/canbus/ a@a-SYSTEM-PRODUCT-NAME:~\$ cd /opt/aaeon/canbus/ a@a-SYSTEM-PRODUCT-NAME:/opt/aaeon/canbus\$

2. Type command to uninstall: /Setup.sh uninstall
a@a-SYSTEM-PRODUCT-NAME:/opt/aaeon/canbus\$./Setup.sh uninstall
remove aaeon UP can rules
[sudo] password for a:
Removed /etc/systemd/system/multi-user.target.wants/UpCanUsb.service.
Removed /etc/systemd/system/multi-user.target.wants/UpCanUart.service.
a@a-SYSTEM-PRODUCT-NAME:/opt/aaeon/canbus\$

How to change buadrate and can interface name

if you want to change Buadrate and infterface name, must to go to location as below to change config on uartcan.sh or usbcan.sh which you wanted, you need super user permissions to change file.

		Computer / opt	/ aaeon / canbu	IS			:	٩				=	
										Ť			
		>			>_	>_	>_	[• •]	[•]	344	>_		
nware_ file		aaeon- bootloader n.rules .sh	canbususer fi guide.odt	induartcan jre-8u401 linux-x64	- replaceuart .sh	Setup.sh	uartcan.sh	UpCanUart. service	UpCanUsb. service	update_ firmware	usbcan.s	h	
				tar.gz									
_				usbcar	.sh [Read-On	lv1							
Open ~	<				:/aaeon/canbus	10				Save		- (
#!/bin	n/bash												
#			#										
	adrate could be set	as below:	#										
# #	10K 20K		# #										
#	50K		#										
#	100K		#										
#	125K		#										
#	250K		#										
#	500K		#										
# #	800K 1000K		# #										
# #			#										
	ate="1000K"												
	an device name confl	icted,you cou	ld be chang	ge name to oth	iers								
	ne="can0"												
	(an conhesta()												
S													
	convegen()		-o AaeonUst	CanO Scanname									
{		115200 -s\$1											
{	sudo -S slcand -S sleep 0.1	115200 -s \$1											
{	<mark>sudo</mark> -S slcand -S <mark>sleep</mark> 0.1 <mark>sudo</mark> ip link set												
{	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1</pre>	\$canname up											
£	<mark>sudo</mark> -S slcand -S <mark>sleep</mark> 0.1 <mark>sudo</mark> ip link set	\$canname up	euelen 1000	90									
{ }	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set</pre>	\$canname up	euelen 1000	90									
{ } functi	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1</pre>	\$canname up	euelen 1000	90									
{ } functi	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set Lon kill() pid=\$(ps aux gr</pre>	\$canname up \$canname txqu			-v grep	awk '{pr	int \$2}')					
{ } functi	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set Lon kill() pid=\$(ps aux gr echo \$pid</pre>	\$canname up \$canname txqu ep slcand g			-v grep	awk '{pr	int \$2}';)					
{ } functi {	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set Lon kill() pid=\$(ps aux gr</pre>	\$canname up \$canname txqu ep slcand g			-v grep	awk '{pr	int \$2}')					
{ } functi { }	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set Lon kill() pid=\$(ps aux gr echo \$pid</pre>	\$canname up \$canname txqu ep slcand g			-v grep	awk '{pr	int \$2}';)					
{ } functi { }	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set Lon kill() pid=\$(ps aux gr echo \$pid</pre>	\$canname up \$canname txqu ep slcand g			-v grep	awk '{pr	int \$2}')					
{ } functi { } functi {	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid</pre>	Şcanname up Şcanname txqu ep slcand g			-v grep	awk '{pr	int \$2}')					
{ } functi { } functi {	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid lon Start() case \$Buadrate in</pre>	Şcanname up Şcanname txqu ep slcand g			-v grep	awk '{pr	int \$2}';)					
{ functi { functi { functi } functi { functi { functi } } functi { functi } functi { functi } functi { functi } functi } functi	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set Lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid Lon Start() case \$Buadrate in "10K")</pre>	Şcanname up Şcanname txqu ep slcand g			-v grep	awk '{pr	int \$2}'')					
{ functi } functi {	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set Lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid Lon Start() case \$Buadrate in "10K") canbegin</pre>	Şcanname up Şcanname txqu ep slcand g			-v grep -	awk '{pr	int \$2}';)					
{ functi { functi { functi } functi { functi { functi } } functi { functi } functi { functi functi } functi	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set Lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid Lon Start() case \$Buadrate in "10K")</pre>	Şcanname up Şcanname txqu ep slcand g			-v grep	awk '{pr	int \$2}')					
{ functi { } functi {	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid lon Start() case \$Buadrate in "10K") ;;;</pre>	Şcanname up Şcanname txqu ep slcand g 0			-v grep	awk '{pr	int \$2}')					
{ functi functi { functi f	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set ton kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid ton Start() case \$Buadrate in "10K") canbegin ;; "20K")</pre>	Şcanname up Şcanname txqu ep slcand g 0			-v grep	awk '{pr	int \$2}';)					
{ functi } functi {	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid lon Start() case \$Buadrate in "10K") canbegin ;; "20K")</pre>	Scanname up Scanname txqu ep slcand g 0 1			-v grep	awk '{pr	int \$2}'')					
{ } functi {	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid lon Start() case \$Buadrate in "10K") canbegin ;; "20K") canbegin ;; "50K")</pre>	Scanname up Scanname txqu ep slcand g 0 1			-v grep	awk '{pr	int \$2}';)					
{ functi functi } functi { functi fun	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid lon Start() case \$Buadrate in "10K") canbegin ;; "50K") canbegin ;;</pre>	Scanname up Scanname txqu ep slcand g 0 1			-v grep	awk '{pr	int \$2}';)					
{ functi { } functi {	<pre>sudo -S slcand -S sleep 0.1 sudo ip link set sleep 0.1 sudo ip link set lon kill() pid=\$(ps aux gr echo \$pid sudo kill -9 \$pid lon Start() case \$Buadrate in "10K") canbegin ;; "20K") canbegin ;; "50K")</pre>	<pre>\$canname up \$canname txqu ep slcand g 0 1 2</pre>			-v grep	awk '{pr	int \$2}';)					

"125K"**)**

Starred

49

sh ~ Tab Width: 8 ~ Ln 1, Col 1 INS \sim

Open	✓ [F]	uartcan.sh [Read-Only] /opt/aaeon/canbus	Save =	- (×
#!/b	in/bash				
2					
#		#			
1 # Bi	uadrate could be set as below:				
5 #	10K				
5 #	20K				
7 #	50K				
в #	100K				
9 #	125K				
9 <mark>#</mark>	250K				
1 #	500K				
2 #	800K				
3 #	1000K				
4 #					
	rate="1000K"				
		could be change name to others			
7 canna	ame="can1"				
В					
	tion canbegin()				
9 {					
1		-s\$1 -o AaeonUartCan \$canname			
2	sleep 0.1				
3	sudo ip link set Șcanname	up			
4	sleep 0.1				
5	sudo ip link set \$canname	txqueuelen 10000			
5 }					
	tion kill()				
в {					
9		grep AaeonUartCan grep -v grep awk '{print \$2}')			
Э	echo \$pid				
1	sudo kill -9 \$pid				
2 }					
3					
	tion Start()				
5 {					
5	case \$Buadrate in				
7	"10K")				
В	canbegin 0				
9	;;				
9	"20K")				
1	canbegin 1				
2	;; "50K")				
5 1					
4 5	canbegin 2				
5	;; "100K")				
7 B	canbegin 3				
9	;; "125K")				
9	IZOK				
		sh ∽ Tab Width: 8 ∽	Ln 3, Col 1	1 ~	IN

Finished change,need to type command to restart service. If change USB CAN Bus , type command:

sudo systemctl stop UpCanUsb.service sudo systemctl start UpCanUsb.service

If change UART CAB Bus, type command:

sudo systemctl stop UpCanUart.service sudo systemctl start UpCanUart.service

How to update firmware

Into firmware update mode have two approach:

1.SW

(1)On the folder, Right-click to select Open in Terminal

	🕝 Computer / opt / aae	eon / canbus			: Q		
firmware_UnifiedHos_UnifiedHos_8 file_t-1.17.0 t-1.19.1 c	7-aaeon- bootloader car	nbususer uide.odt	jre-8u401- linux-x64sh tar.gz	euart Setup.sh	uartcan.sh UpCanUart. service	UpCanUsb. service	ateusbcan.sh
		New Folder	Shift+Ctrl+N				
		Add to Bookmark	s Ctrl+D				
		Paste Select All	Ctrl+A				
		Open in Terminal					
		Properties					

(2)Type command to run script : sudo ./bootloader.sh

a@a-SYSTEM-PRODUCT-NAME:/opt/aaeon/canbus\$ sudo ./bootloader.sh

2.EE

Swtich the jump to into bootloader mode



Update tool:

Type command to run script : sudo ./bootloader.sh

a@a-SYSTEM-PRODUCT-NAME:/opt/aaeon/canbus\$ sudo ./bootloader.sh

	Unified Bootloader Host Application v1.17.0	- • ×				
File Settings Tools Help						
Device Architecture: Selected Hex File:	< No Hex File Currently Selected >					
 Bootloader Host 						

Unified Bootloader Host Application v1.17.0 – 🗆 🗙							
File Settings Tools Help							
Device Architecture: Selected Hex File: Bootloader Host	PIC32MK\PIC32MX\PIC32MZ\SAM MCUs ▼ PIC10\PIC12\PIC16\PIC18 MCUs ATtiny\ATmega\AVR128D MCUs PIC24 MCUs \ dsPIC33 DSCs PIC32MK\PIC32MX\PIC32MZ\SAM MCUs						
Protocol: UA Configuration: Not	ART Settings: Configure Configured Program Device mmunication Method: UART						

Unified Bootloader Host Application v1.17.0 – 🗆 🗙								
File Settings Tools Help								
Device Architecture: PIC32MK\PIC32MX\PIC32MZ\SAM MCUs Selected Hex File: < No Hex File Currently Selected >								
▼ Bootloader Host								
32-Bit Bootloader Configuration Window								
Protocol: USB - Settings: Configure								
Configuration: UART ed								
USB UDP Program Device								
Status: Communication Method: USB								

Click Configure and Select "3C" to Apply

	Unified Bootloader Host Application v1.17.0 – ×						
File Settir	ngs Tools Help						
Device Archit Selected H							
 Bootload 	der Host						
	32-Bit Bootloader Configuration Window						
Prot(Configura	USB Devices: 3C - C Program Device						
Sta	Apply						

Open to Console

	Unified Bootloader Host Application v1.17	.0	- (×			
File Settings	Tools Help						
Console 2MK\PIC32MX\PIC32MZ\SAM MCUs Selected Hex File: < No Hex File Currently Selected >							
 Bootloader Ho 	ost						
	32-Bit Bootloader Configuration Window						
Protocol:	USB 🔻 Settings: Configu	ire					
Configuration:	HID device PID: 3C Connected.						
	Program	n Device					
Status:	Communication Method: USB						

Select bin file to record

			Unifie	d Bootl	oader H	ost Applicat	tion v1.	17.0	-	. (×
File	Settings	Tools	Help								
Open/Load File (*.hex) Close Selected Hex File: < No Hex File Currently Selected >											
▼ Bo	otloader Ho	ost									
	Protocol: guration: Status:	USB HID de	• evice PI	D: 3C C	Configur onnected nod: USB	ation Window Settings: d.	Config	gure am Device			

Click Program Device to record

File Settings T	Unified Bootloader Host Application v1.	17.0 – • ×	.ld process "dbu	
The Settings I			Console	- • ×
Device Architecture:	PIC32MK\PIC32MX\PIC32MZ\SAM MCUs			
Selected Hex File:	canbus_v1_0_1.hex loaded			
 Bootloader Host 				
	32-Bit Bootloader Configuration Window			
Protocol:	USB 🔻 Settings: Confi	gure		
Configuration: H	ID device PID: 3C Connected.			
	Progr	am Device		
Status: C	communication Method: USB			

Unified Bootloader Host Application v1.17.0	– 🗆 × .ld process "dbu
File Settings Tools Help	Console – 🗆 🗙
Device Architecture: PIC32MK\PIC32MX\PIC32MZ\SAM MCUs ▼ Selected Hex File: canbus_v1_0_1.hex loaded	22:38:11.268 > Device: 3C Bootloading started 22:38:11.321 > Hex File (re)loaded 22:38:11.323 > Reading Version 22:38:11.339 > Bootloader Version Read Successful 22:38:11.350 > Erasing Device
Bootloader Host 32-Bit Bootloader Configuration Window	22:38:19.252 > Erase Successful 22:38:19.252 > Programming Flash 22:38:22.697 > Flashed Programmed
Protocol: USB Settings: Configuration: HID device PID: 3C Connected. Status: Disconnected after Programming was Successful.	22:38:22.698 > Resetting Device 22:38:22.698 > Device Reset 22:38:22.698 > Device: Bootloaded Successful