

# **FINTEK**

# **PCIE to UART Driver Installation Guide for Linux**

v1.09

Mar 24, 2026

## Datasheet Revision History

Date	Version	Revision History
2019/5/21	v1.0	Initial Version.
2019/7/25	v1.01	Update makefile & document.
2023/11/20	v1.02	Update makefile & document.
2024/6/27	v1.02.1	Add set_mode.c to change RS232/485/422 mode.
2024/7/12	v1.03	Fix compile issue on kernel 6.0. Update document for installation.
2024/8/2	v1.03.1	Document change.
2024/10/25	v1.04	Fix F75111 detect issue Add set_gpio.c Fix set_mode.c
2025/3/12	v1.05-20250221	Fix F81450 detect issue
2025/4/30	v1.06-20250430	Add com port disable with driver parameter
2025/5/8	v1.07-20250508	Add com port disable with H/W strap pin
2025/6/10	v1.08-20250610	Add GPIO support, controlled by reg 243h or kernel param "uart_en". Fix kernel 6.1 build issue.
2025/8/15	v1.08.1-20250815	Update set_mode/set_gpio. Driver version same with v1.08-20250610.
2026/3/24	v1.09-20260324	Fix kernel 6.17 build issue. Support EEPROM disable serial port.

## 1. Driver installation

The following installation is based on Ubuntu 22.04:

1. `sudo su`
2. `apt-get update`
3. `apt-get dist-upgrade`
4. `reboot`
5. `apt-get install gcc make build-essential`
6. Unzip the driver.zip
7. `cd driver`
8. `make clean ; make ; make install`
9. If the target system is using SecureBoot mode, Please refer the following link to sign driver:  
<https://ubuntu.com/blog/how-to-sign-things-for-secure-boot>
10. Reference section “Debian 7 / Ubuntu Configurations” for setting.
11. `reboot`
12. Check if any new port added by “`dmesg | grep ttyS`”, if you cant see any new add ports, please following the Troubleshooting A1 then `insmod` driver again.

```
[ 26.194435] 0000:02:00.0: ttyS4 at MMIO 0xf7b00000 (irq = 46, base_baud = 230400) is a 16550A
[ 26.200733] 0000:02:00.0: ttyS5 at MMIO 0xf7b00008 (irq = 47, base_baud = 230400) is a 16550A
[ 26.207132] 0000:02:00.0: ttyS6 at MMIO 0xf7b00010 (irq = 48, base_baud = 230400) is a 16550A
[ 26.227131] 0000:02:00.0: ttyS7 at MMIO 0xf7b00018 (irq = 45, base_baud = 230400) is a 16550A
```

## 2. Kernel configure

If you are using embedded system, ensure the following kernel configuration is en/disabled.

Device Drivers --->

[\*] PCI support --->

Character devices --->

[\*] Enable TTY

Serial drivers --->

<\*> 8250/16550 and compatible serial support

< > 8250/16550 PCI device support (Disable)

(48) Maximum number of 8250/16550 serial ports

(32) Number of 8250/16550 serial ports to register at runtime

### 3. Serial Port Mode Change

The Linux official RS485 API released with Kernel 3.19.0. If you are using Kernel 3.19.0 or newer kernel, It can be set by TIOCSRS485/TIOCGRS485 ioctl. Otherwise we can control the mode with our demo app “set\_mode”.

```

root@code:/home/code/ddd/hpeter/fintek/F81504-Series/change_mode# make
gcc -I.. -c -o set_mode.o set_mode.c
gcc set_mode.o -o set_mode
root@code:/home/code/ddd/hpeter/fintek/F81504-Series/change_mode# ./set_mode
./set_mode
./set_mode <port> <mode>

Uart Mode table:
1 --> eModeRS232
2 --> eModeRS485 - DE#/RE (TX Enable with RTS Low)
3 --> eModeRS485_1 - DE/RE# (TX Enable with RTS High)

root@code:/home/code/ddd/hpeter/fintek/F81504-Series/change_mode# ./set_mode ttyS4 2
Target:/sys/class/tty/ttyS4/device/config
291c411
Target:/sys/class/tty/ttyS4/port
change to mode: eModeRS485 - DE#/RE (TX Enable with RTS Low)
root@code:/home/code/ddd/hpeter/fintek/F81504-Series/change mode#
    
```

This screen-shot is based on Ubuntu 14.04.2.

Section1: Compile App

Section2: App Parameter

Section2: Demo change to RS485 Mode with DE# (We should also setup transceiver correctly)

#### Transceiver mode change (on EVB)

We designed special EVB for F81504A/Transceiver mode switch. It can be controlled by set\_mode / set\_gpio to change the mode for application (RS422/485/232/ ...).

```

root@code-ms7c82:/home/code
Fintek F81504 set gpio tools
Set mode
./set_gpio <port> <mode>
Query mode
./set_gpio <port>

Uart Mode table:
F81439(M0/M1/M2)
0: 000 - RS422
1: 001 - RS232
2: 010 - RS485 Low Active
3: 011 - RS485 High Active
4: 100 - RS422 with termination
5: 101 - Reserved
6: 110 - RS485 Low Active with termination
7: 111 - Shutdown

F81435(M0/M1/M2)
0: 000 - RS422 High Active
1: 001 - RS232
2: 010 - RS485 High Active
3: 011 - ccTALK
4: 100 - RS422 Low Active
5: 101 - RS485 Low Active (Auto Sensing)
6: 110 - RS485 Low Active
7: 111 - RS422 Low Active (Auto Sensing)

F81433(M0/M1/R_DIS)
1: 001 - RS232
2: 010 - RS422 with termination
3: 011 - RS422
4: 100 - RS485 Low Active with termination (Auto Sensing)
5: 101 - RS485 Low Active (Auto Sensing)
7: 111 - Shutdown
    
```

The set\_mode / set\_gpio must set correctly for using this EVB, the default mode is RS232. The following examples will use F81439 and ttyS7 for demo.

RS232:

```

./set_mode ttyS7 1
./set_gpio ttyS7 1
    
```

RS485:

```

./set_mode ttyS7 2
./set_gpio ttyS7 2
    
```

RS422:

```

set_mode: don't care.
./set_gpio ttyS7 0
    
```

## Query transceiver mode & type

```
./set_gpio <com port>
```

```
root@code-ms7c82:/home/code/ddd/old/hpeter/fintek/tmp/f81504/change_mode# ./set_gpio ttyS7
Transceiver: F81439
current mode: 1
root@code-ms7c82:/home/code/ddd/old/hpeter/fintek/tmp/f81504/change_mode# █
```

## H/W fix transceiver mode

Linux Driver will only detect F81504A Transceiver & H/W fix mode when driver load, reboot & S3/4 resume. The detecting flow is as following:

1. Set the M0/M1/M2 with the com port as input mode.
2. Read the three pins value.
3. If the three pins are digital 1, this com port can be controlled by set\_gpio, otherwise is H/W fix mode.

## GPIO control

The Fintek F81504A/508A/512A can convert serial ports to GPIOs, controlled via register 0x243 or the kernel parameter "uart\_en". We can control it using Linux "gpiod" tools. For example, connect GPIO03 (SOUT3) and GPIO04 (SIN3), then use "gpioset gpiochip0 3=1 / 3=0" to toggle GPIO03, and "gpioget 0 4" to read GPIO04.

```
root@code-ms7592:/home/code
3 modules
#make -C /DataDisk/hpeter/DMA-210UII/samsung_android_kernel_3.0 M=/home/code/ddd/old/hpeter/fintek/F81504-Series/U3 modules
#make -C /DataDisk/freescale/android_x86-6.0-r2/out/target/product/x86/obj/kernel M=/home/code/ddd/old/hpeter/fintek/F81504-Series/U3 modules
root@code-ms7592:/home/code/ddd/old/hpeter/fintek/F81504-Series/U3# insmod f81504a_u3.ko uart_en=-1

root@code-ms7592:/sys/class/gpio/gpio523#
root@code-ms7592:/sys/class/gpio/gpio523#
root@code-ms7592:/sys/class/gpio/gpio523#
root@code-ms7592:/sys/class/gpio/gpio523#
root@code-ms7592:/sys/class/gpio/gpio523# gpioset gpiochip0 3=1
root@code-ms7592:/sys/class/gpio/gpio523# gpioset gpiochip0 3=0

root@code-ms7592:/sys/class/gpio#
root@code-ms7592:/sys/class/gpio#
root@code-ms7592:/sys/class/gpio# gpioget 0 4
1
root@code-ms7592:/sys/class/gpio# gpioget 0 4
0
root@code-ms7592:/sys/class/gpio#

[ 3960.807120] f81504a_u3-4-8-12 0000:01:00.0: found GPIO IC(9c), set_gpio supported
[ 3960.824069] f81504a_u3-4-8-12 0000:01:00.0: We had 16 MSI allocated
[ 3960.825025] f81504a_u3-4-8-12 0000:01:00.0: IRQ start from: 71
[ 3960.845421] 0000:01:00.0: ttyS26 at MMIO 0xdff00000 (irq = 72, base_baud = 230400) is a 16550A
[ 3960.865896] 0000:01:00.0: ttyS27 at MMIO 0xdff00008 (irq = 73, base_baud = 230400) is a 16550A
[ 3960.866145] f81504a_u3-4-8-12 0000:01:00.0: f81504a_probe: serial port: 2 change to gpio set: 0
[ 3960.866270] f81504a_u3-4-8-12 0000:01:00.0: f81504a_probe: serial port: 3 change to gpio set: 1

[0] 0: bash* 1: bash- 2: bash "code-ms7592" 15:11 10-6-11-25
```

# Appendix:

## Prepare Tools:

### If You are using Debian:

1. nano /etc/apt/sources.list
2. add '#' in the front of "deb cdrom:[Debian GNU/Linux 7.8.0 \_Wheezy\_ - Official i386 DVD Binary-1 20150110-13:32]/ wheezy contrib main"

```
# deb cdrom:[Debian GNU/Linux 7.8.0 _Wheezy_ - Official i386 DVD Binary-1 20150110-13:32]/ wheezy contrib main
#deb cdrom:[Debian GNU/Linux 7.8.0 _Wheezy_ - Official i386 DVD Binary-1 20150110-13:32]/ wheezy contrib main
deb http://ftp.tw.debian.org/debian/ wheezy main
deb-src http://ftp.tw.debian.org/debian/ wheezy main
```

3. CTRL-X & save
4. apt-get update
5. apt-get dist-upgrade
6. reboot
7. apt-get install gcc make linux-headers-`uname -r` build-essential

### If You are using Ubuntu:

1. apt-get update
2. apt-get dist-upgrade
3. reboot
4. apt-get install gcc make build-essential

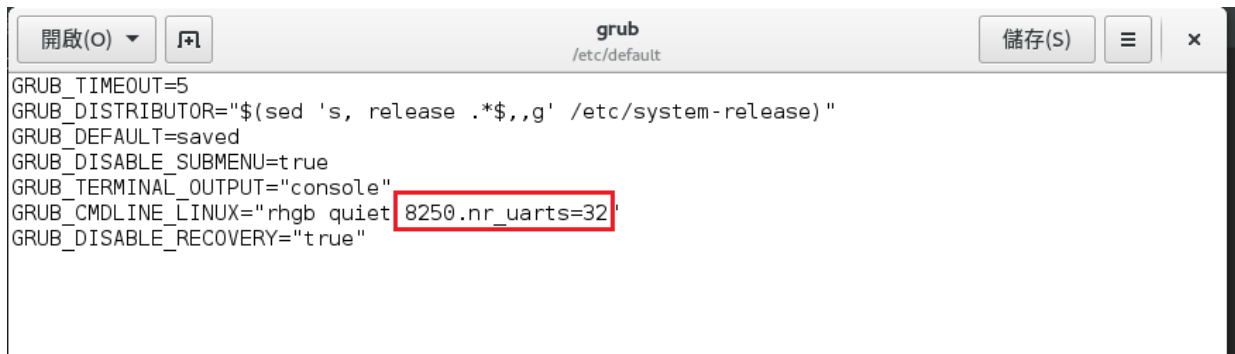
### If You are using Fedora series (e.g. CentOS / Fedora 21):

1. yum update
2. reboot
3. yum install gcc make kernel-devel rpm-build

# Boot Configuration

## Fedora 21 Configurations / CentOS 7

1. sudo su
2. gedit /etc/default/grub
3. add command "8250.nr\_uaarts=32 initcall\_blacklist=serial\_pci\_driver\_init" to "GRUB\_CMDLINE\_LINUX" like:



```

GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="rhgb quiet 8250.nr_uaarts=32"
GRUB_DISABLE_RECOVERY="true"
    
```

4. grub2-mkconfig > /boot/grub2/grub.cfg
5. reboot
6. dmesg | grep ttyS

```

[code@localhost ~]$ dmesg | grep ttyS
[ 0.445584] 00:06: ttyS0 at I/O 0x3f8 (irq = 4, base_baud = 115200) is a 16550A
[ 0.467713] 0000:02:00.0: ttyS4 at I/O 0xd000 (irq = 16, base_baud = 115200) is a 16550A
[ 0.488809] 0000:02:00.0: ttyS5 at I/O 0xd008 (irq = 16, base_baud = 115200) is a 16550A
[ 0.509899] 0000:02:00.0: ttyS6 at I/O 0xd010 (irq = 16, base_baud = 115200) is a 16550A
[ 0.530986] 0000:02:00.0: ttyS7 at I/O 0xd018 (irq = 16, base_baud = 115200) is a 16550A
[ 0.552177] 0000:06:00.0: ttyS8 at I/O 0xb000 (irq = 18, base_baud = 115200) is a 16550A
[ 0.573278] 0000:06:00.0: ttyS9 at I/O 0xb008 (irq = 18, base_baud = 115200) is a 16550A
[ 0.594377] 0000:06:00.0: ttyS10 at I/O 0xb010 (irq = 18, base_baud = 115200) is a 16550A
[ 0.615463] 0000:06:00.0: ttyS11 at I/O 0xb018 (irq = 18, base_baud = 115200) is a 16550A
[ 0.636550] 0000:06:00.0: ttyS12 at I/O 0xb020 (irq = 18, base_baud = 115200) is a 16550A
[ 0.657641] 0000:06:00.0: ttyS13 at I/O 0xb028 (irq = 18, base_baud = 115200) is a 16550A
[ 0.678728] 0000:06:00.0: ttyS14 at I/O 0xb030 (irq = 18, base_baud = 115200) is a 16550A
[ 0.699833] 0000:06:00.0: ttyS15 at I/O 0xb038 (irq = 18, base_baud = 115200) is a 16550A
[ 0.720936] 0000:06:00.0: ttyS16 at I/O 0xb040 (irq = 18, base_baud = 115200) is a 16550A
[ 0.742037] 0000:06:00.0: ttyS17 at I/O 0xb048 (irq = 18, base_baud = 115200) is a 16550A
[ 0.763133] 0000:06:00.0: ttyS18 at I/O 0xb050 (irq = 18, base_baud = 115200) is a 16550A
[ 0.784234] 0000:06:00.0: ttyS19 at I/O 0xb058 (irq = 18, base_baud = 115200) is a 16550A
[code@localhost ~]$ sudo su
[sudo] password for code:
[root@localhost code]# uname -a
Linux localhost.localdomain 3.19.7-200.fc21.x86_64 #1 SMP Thu May 7 22:00:21 UTC 2015 x86_64 x86_64 x86_64 GNU/Linux
    
```

## CentOS 6 Configurations

1. su
2. gedit /boot/grub/grub.conf
3. Add text "8250.nr\_uarts=32 initcall\_blacklist=serial\_pci\_driver\_init" to kernel option.

```

*grub.conf x
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE: You have a /boot partition. This means that
#         all kernel and initrd paths are relative to /boot/, eg.
#         root (hd0,0)
#         kernel /vmlinuz-version ro root=/dev/mapper/vg_livedvd-lv_root
#         initrd /initrd-[generic-]version.img
#boot=/dev/sda
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title CentOS (2.6.32-504.16.2.el6.i686)
    root (hd0,0)
    kernel /vmlinuz-2.6.32-504.16.2.el6.i686 ro root=/dev/mapper/vg_livedvd-lv_root rd_NO_LUKS LANG=en_US.UTF-8
rd_LVM LV=vg_livedvd/lv_root rd_NO_MD SYSFONT=lataarcyrheb-sun16 crashkernel=auto rd_LVM_LV=vg_livedvd/lv_swap
KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM rhgb quiet 8250.nr_uarts=32
    initrd /initramfs-2.6.32-504.16.2.el6.i686.img
title CentOS (2.6.32-279.el6.i686)
    root (hd0,0)
    kernel /vmlinuz-2.6.32-279.el6.i686 ro root=/dev/mapper/vg_livedvd-lv_root rd_NO_LUKS LANG=en_US.UTF-8
rd_LVM LV=vg_livedvd/lv_root rd_NO_MD SYSFONT=lataarcyrheb-sun16 crashkernel=auto rd_LVM_LV=vg_livedvd/lv_swap
KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM rhgb quiet 8250.nr_uarts=32
    initrd /initramfs-2.6.32-279.el6.i686.img
    
```

4. Save
5. cp /usr/share/dbus-1/system-services/org.freedesktop.ModemManager.service /usr/share/dbus-1/system-services/org.freedesktop.ModemManager.service.bak
6. Reboot

## Debian 7 / Ubuntu Configurations

1. su  
or sudo su
2. nano /etc/default/grub
3. add  
8250.nr\_uares=32 initcall\_blacklist=serial\_pci\_driver\_init  
to "GRUB\_CMDLINE\_LINUX\_DEFAULT"

```
GRUB_DEFAULT=0
GRUB_HIDDEN_TIMEOUT=10
GRUB_HIDDEN_TIMEOUT_QUIET=false
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash initcall_blacklist=serial_pci_driver_init 8250.nr_uares=32"
GRUB_CMDLINE_LINUX=""
```

4. update-grub
5. reboot

## Disable com port with driver parameter

We can disable serial port with driver parameter. The following instruction will disable serial port 2 & 3 (started with serial port 0) in Ubuntu 24.04.

1. Install Driver.
2. Edit file in “/etc/modprobe.d/f81504a.conf” and add content as following:

```
options f81504a_u3 disable_port=0x0c
```

The hexadecimal 0x0c can be mapping to binary 00001100, we only disable bit 2 & 3.

3. It'll apply when reboot.

```
[ 1.973931] f81504a_u3: loading out-of-tree module taints kernel.
[ 1.973935] f81504a_u3: module verification failed: signature and/or required key missing - tainting kernel
[ 1.974196] f81504a_u3-4-8-12 0000:01:00.0: f81504a_probe: version: v1.06-20250430
[ 1.975453] f81504a_u3-4-8-12 0000:01:00.0: Cant found any GPIO IC, no set_gpio support (1)
[ 1.975460] f81504a_u3-4-8-12 0000:01:00.0: enabling device (0000 -> 0003)
[ 1.975708] f81504a_u3-4-8-12 0000:01:00.0: We had 16 MSI allocated
[ 1.975763] f81504a_u3-4-8-12 0000:01:00.0: IRQ start from: 151
[ 1.975890] 0000:01:00.0: ttyS6 at MMIO 0x80a00000 (irq = 152, base_baud = 230400) is a 16550A
[ 1.977966] 0000:01:00.0: ttyS7 at MMIO 0x80a00008 (irq = 153, base_baud = 230400) is a 16550A
```