

AAEON

PCH SMBus

Linux Driver User Guide

Most Confidential	
Confidential	
Internal	
Public	V

Revision History	2
Introduction	4
Supported Linux Version	4
Prepare and install required packages:	5
Install i2c-tools:	5
Check SMBus drivers (kernel module):	5
Access SMBus slave device:	5

Revision History

Revision Number	Description	Revision Date
V1.0	Initial Release	2020/04/15
V1.1	Fix wording and notice to modify the blacklist of modprobe.d	2021/05/04
V1.2	Support Ubuntu 23.10.1	2023/12/05

Introduction

This document describes how to install the driver for PCH SMBus controller on Intel x86-64 platforms.

Supported Linux Version

- Ubuntu 23.10.1, Kernel: 6.5.0-13-generic, or above.

Prepare and install required packages:

Install i2c-tools:

The user can install the i2c-tools by the instruction:

```
$ sudo apt install i2c-tools
```

Check SMBus drivers (kernel module):

i2c_i801, and i2c_smbus are the necessary drivers for SMBus controller. And these drivers are built-in Ubuntu. The user can check if the module is probed by the following instructions:

```
$ lsmod | grep i801
i2c_i801          36864 0
i2c_smbus        16384 1 i2c_i801
```

If there is no i2c_i801 driver module probed, the i2c_i801 driver module is in the blacklist. Please edit the configure /etc/modprobe.d/blacklist.conf to remove i2c_i801 from the blacklist.

Access SMBus slave device:

Find out the SMBus controller number:

```
$ sudo i2cdetect -l | grep -i smbus
i2c-1 smbus SMBus I801 adapter at efa0 SMBus adapter
```

In this case, the SMBus controller bus number is 1 of i2c-1. Remember this number, and we will use it at the following instructions.

Note: The controller number might be changed after the device rebooted.

Detect the slave devices, where 5 of the following instruction is the SMBus bus number:

```
c@c:~$ sudo i2cdetect -y -a 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00: -- -- -- -- -- -- -- -- 08 -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: 20 -- -- -- -- -- -- -- -- -- -- -- 2e --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- 44 -- -- -- -- 4a -- -- -- -- --
50: -- -- 52 -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- 68 -- -- -- -- -- 6f --
70: -- -- -- -- -- -- -- -- -- -- -- -- -- 7e --
```

These numbers are the SMBus address where a certain device is detected. In this example, 0x20 and 0x68 are the devices we connected on.

Dump the register of the slave device on the address: 0x20:

```
c@c:~$ sudo i2cdump -y -a 1 0x20
No size specified (using byte-data access)
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f      0123456789abcdef
00: 00 49 ff ff ff ff 00 04 00 04 00 70 00 00 00      .I.....??.p...
10: 15 00 01 00 80 00 28 00 e2 00 42 00 51 00 b4 00      ?..??.(?.B.Q.?.
20: 10 00 53 00 00 00 01 00 01 00 01 00 01 00 01 00      ?.S...?..?..?..
30: 01 00 01 00 01 00 64 00 00 00 41 00 26 00 18 00      ?..??.d...A.&?.
40: 36 00 06 00 21 00 00 00 00 00 00 00 00 00 30 00      6.?!......0.
50: 34 00 37 00 38 00 20 00 20 00 00 00 00 00 00 00      4.7.8. . ....
60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
80: 00 00 03 00 09 00 00 00 01 00 00 00 08 00 0f 00      ..?..?..?..?..
90: 03 00 1e 00 14 00 01 00 01 00 00 00 00 00 00 00      ?..?..?..?..
a0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
b0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
c0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
d0: 00 00 00 00 00 00 00 00 01 00 03 00 05 00 07 00      .....?..?..?..
e0: 09 00 0b 00 0d 00 0f 00 01 00 00 00 12 00 56 00      ?..?..?..?..?..V.
f0: ff 00 ff 00 ff 00 ff 00 ff 00 ff 00 ff 00 ff 00      .....
```

The register 0x04 is writable on this sample device, and we can write it by the command: i2cset, and read it by the command i2cget:

```
c@c:~$ sudo i2cset -y -a 1 0x20 0x04 0x01
c@c:~$ sudo i2cget -y -a 1 0x20 0x04
0x01
c@c:~$ sudo i2cdump -y -a 1 0x20
No size specified (using byte-data access)
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f      0123456789abcdef
00: 00 49 ff ff 01 ff ff 00 04 00 04 00 70 00 00 00      .I..?..?..?..p...
10: 15 00 01 00 80 00 28 00 e2 00 42 00 51 00 b4 00      ?..??.(?.B.Q.?.
20: 10 00 53 00 00 00 01 00 01 00 01 00 01 00 01 00      ?.S...?..?..?..
30: 01 00 01 00 01 00 64 00 00 00 41 00 26 00 18 00      ?..??.d...A.&?.
40: 36 00 06 00 21 00 00 00 00 00 00 00 00 00 30 00      6.?!......0.
50: 34 00 37 00 38 00 20 00 20 00 00 00 00 00 00 00      4.7.8. . ....
60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
80: 00 00 03 00 09 00 00 00 01 00 00 00 08 00 0f 00      ..?..?..?..?..
90: 03 00 1e 00 14 00 01 00 01 00 00 00 00 00 00 00      ?..?..?..?..
a0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
b0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
c0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00      .....
d0: 00 00 00 00 00 00 00 00 01 00 03 00 05 00 07 00      .....?..?..?..
e0: 09 00 0b 00 0d 00 0f 00 01 00 00 00 12 00 56 00      ?..?..?..?..?..V.
f0: ff 00 ff 00 ff 00 ff 00 ff 00 ff 00 ff 00 ff 00      .....
```

Note: Not all registers of the device are writable, please read the datasheet of the device. Then, you can get more information from the [i2c-tools documents](#).