

AAEON PCH I2C

Linux Driver User Guide

Most Confidential	
Confidential	
Internal	
Public	V

04/30, 2026
Version V1.1

Revision History	3
Introduction	4
Prepare and install required packages	5
Install i2c-tools	5
Access I2C slave device	5

Revision History

Revision Number	Description	Revision Date
V1.0	Initial Release	2024/06/12
V1.1	Verified for Ubuntu 26.04	2026/04/30

Introduction

This document describes how to install the driver for PCH I2C controller on aaeon motherboard.

Prepare and install required packages

Install i2c-tools

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

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

Access I2C slave device

Find out the i2c controller number:

```
$ sudo i2cdetect -l | grep Synopsys
i2c-3      i2c      Synopsys DesignWare I2C adapter      I2C adapter
i2c-1      i2c      Synopsys DesignWare I2C adapter      I2C adapter
i2c-6      i2c      Synopsys DesignWare I2C adapter      I2C adapter
i2c-4      i2c      Synopsys DesignWare I2C adapter      I2C adapter
i2c-2      i2c      Synopsys DesignWare I2C adapter      I2C adapter
i2c-7      i2c      Synopsys DesignWare I2C adapter      I2C adapter
i2c-5      i2c      Synopsys DesignWare I2C adapter      I2C adapter
```

Try to detect the i2c device number, when the address matches the slave device's address.
The I2C controller bus number is 4 of i2c-4. Remember this number, and we will use it in the following instructions.

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

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

These numbers are the I2C address where a certain device is detected. In this example, 0x20 and 0x68 are the devices we connected to.

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

```
$ sudo i2cdump -y -a 4 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: ff b6 ff ff 00 00 ff ff XX XX XX XX XX XX XX XX .....XXXXXXXXXX
10: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XXXXXXXXXXXXXXXXXXXX
20: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XXXXXXXXXXXXXXXXXXXX
30: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XXXXXXXXXXXXXXXXXXXX
```

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

```
$ sudo i2cset -y -a 4 0x20 0x04 0x01
$ sudo i2cget -y -a 4 0x20 0x04
0x01
$ sudo i2cdump -y -a 4 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: fe b6 ff ff 01 00 ff ff XX XX XX XX XX XX XX XX ?...?...XXXXXXXXXX
10: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XXXXXXXXXXXXXXXXXXXX
20: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XXXXXXXXXXXXXXXXXXXX
30: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XXXXXXXXXXXXXXXXXXXX
```

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](#).