

**NAME**

i2cget - read from I2C/SMBus chip registers

**SYNOPSIS**

```
i2cget [-f] [-y] i2cbus chip-address [data-address [mode]]  
i2cget -V
```

**DESCRIPTION**

i2cget is a small helper program to read registers visible through the I2C bus (or SMBus).

**OPTIONS**

- V Display the version and exit.
- f Force access to the device even if it is already busy. By default, i2cget will refuse to access a device which is already under the control of a kernel driver. Using this flag is dangerous, it can seriously confuse the kernel driver in question. It can also cause i2cget to return an invalid value. So use at your own risk and only if you know what you're doing.
- y Disable interactive mode. By default, i2cget will wait for a confirmation from the user before messing with the I2C bus. When this flag is used, it will perform the operation directly. This is mainly meant to be used in scripts. Use with caution.

There are two required options to i2cget. *i2cbus* indicates the number or name of the I2C bus to be scanned. This number should correspond to one of the busses listed by *i2cdetect -l*. *chip-address* specifies the address of the chip on that bus, and is an integer between 0x03 and 0x77.

*data-address* specifies the address on that chip to read from, and is an integer between 0x00 and 0xFF. If omitted, the currently active register will be read (if that makes sense for the considered chip).

The *mode* parameter, if specified, is one of the letters **b**, **w** or **c**, corresponding to a read byte data, a read word data or a write byte/read byte transaction, respectively. A **p** can also be appended to the *mode* parameter to enable PEC. If the *mode* parameter is omitted, i2cget defaults to a read byte data transaction, unless *data-address* is also omitted, in which case the default (and only valid) transaction is a single read byte.

**WARNING**

i2cget can be extremely dangerous if used improperly. I2C and SMBus are designed in such a way that an SMBus read transaction can be seen as a write transaction by certain chips. This is particularly true if setting *mode* to **cp** (write byte/read byte with PEC). Be extremely careful using this program.

**SEE ALSO**

[i2cdump\(8\)](#), [i2cset\(8\)](#)

**AUTHOR**

Jean Delvare

This manual page was strongly inspired from those written by David Z Maze for i2cset.