

# I<sup>2</sup>C and SMBus Subsystem

I<sup>2</sup>C (or without fancy typography, "I2C") is an acronym for the "Inter-IC" bus, a simple bus protocol which is widely used where low data rate communications suffice. Since it's also a licensed trademark, some vendors use another name (such as "Two-Wire Interface", TWI) for the same bus. I2C only needs two signals (SCL for clock, SDA for data), conserving board real estate and minimizing signal quality issues. Most I2C devices use seven bit addresses, and bus speeds of up to 400 kHz; there's a high speed extension (3.4 MHz) that's not yet found wide use. I2C is a multi-master bus; open drain signaling is used to arbitrate between masters, as well as to handshake and to synchronize clocks from slower clients.

The Linux I2C programming interfaces support the master side of bus interactions and the slave side. The programming interface is structured around two kinds of driver, and two kinds of device. An I2C "Adapter Driver" abstracts the controller hardware; it binds to a physical device (perhaps a PCI device or platform\_device) and exposes a `:c:type:'struct i2c_adapter <i2c_adapter>'` representing each I2C bus segment it manages. On each I2C bus segment will be I2C devices represented by a `:c:type:'struct i2c_client <i2c_client>'`. Those devices will be bound to a `:c:type:'struct i2c_driver <i2c_driver>'`, which should follow the standard Linux driver model. There are functions to perform various I2C protocol operations; at this writing all such functions are usable only from task context.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\driver-api\ (linux-master) (Documentation) (driver-api) i2c.rst, line 16);**  
[backlink](#)

Unknown interpreted text role "c:type".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\driver-api\ (linux-master) (Documentation) (driver-api) i2c.rst, line 16);**  
[backlink](#)

Unknown interpreted text role "c:type".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\driver-api\ (linux-master) (Documentation) (driver-api) i2c.rst, line 16);**  
[backlink](#)

Unknown interpreted text role "c:type".

The System Management Bus (SMBus) is a sibling protocol. Most SMBus systems are also I2C conformant. The electrical constraints are tighter for SMBus, and it standardizes particular protocol messages and idioms. Controllers that support I2C can also support most SMBus operations, but SMBus controllers don't support all the protocol options that an I2C controller will. There are functions to perform various SMBus protocol operations, either using I2C primitives or by issuing SMBus commands to i2c\_adapter devices which don't support those I2C operations.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\driver-api\ (linux-master) (Documentation) (driver-api) i2c.rst, line 38)**

Unknown directive type "kernel-doc".

```
.. kernel-doc:: include/linux/i2c.h
   :internal:
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\driver-api\ (linux-master) (Documentation) (driver-api) i2c.rst, line 41)**

Unknown directive type "kernel-doc".

```
.. kernel-doc:: drivers/i2c/i2c-boardinfo.c
   :functions: i2c_register_board_info
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\driver-api\ (linux-master) (Documentation) (driver-api) i2c.rst, line 44)**

Unknown directive type "kernel-doc".

```
.. kernel-doc:: drivers/i2c/i2c-core-base.c
   :export:
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\driver-api\linux-master) (Documentation) (driver-api) i2c.rst, line 47)**

Unknown directive type "kernel-doc".

```
.. kernel-doc:: drivers/i2c/i2c-core-smbus.c
   :export:
```