



Your complete choice for I²C-bus peripherals

Our invention, your solution
to everyday design challenges



Broad-based solutions from the

Benefits of the I²C-bus

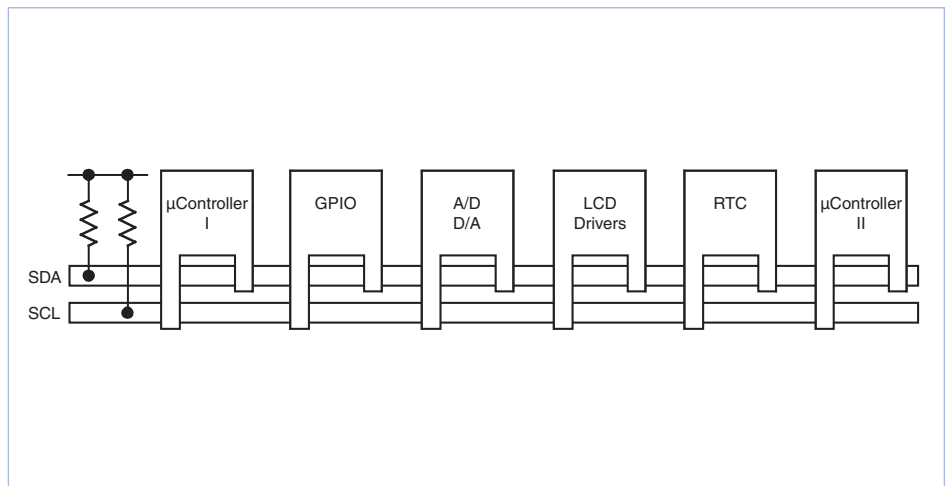
- ▶ De facto world standard
- ▶ Fast time-to-market
- ▶ Proven, robust performance
- ▶ Very wide range of functions
- ▶ Easy to design in
- ▶ Simple to debug
- ▶ Fast to test and assemble
- ▶ Reduces overall cost

NXP Semiconductors is the world's leading supplier of I²C-bus products. We invented the technology more than 20 years ago and never looked back. Since then, we've taken performance to new levels, introduced a wide variety of vital functions, and continue to offer the most flexible options for the most important applications.

We offer the industry's broadest portfolio of I²C-bus logic, providing essential components like GPIO, LED blinkers/dimmers, Hot Swap bus buffers, bus expanders, bus hubs/repeaters, DIP switches, muxes and switches, EEPROMs, data converters, LCD segment drives, and more.

To meet the needs of today's high-performance systems, our Fast-mode Plus (Fm+) devices operate at up to 1 MHz and drive up to 30 mA, so they work with long buses and support applications of up to 1,500 pF. They also use the Fast-mode software protocol (400 kHz), so you can upgrade speed without having to change your design software.

To overcome the limits of interfacing a host bus to its peripheral, we've introduced a new series of bridge ICs that manage the connection between the I²C-bus and other protocols. The bridges make it possible for legacy I²C-bus structures to connect to new systems, too, so you can retain your original design investment while extending the lifespan of older I²C-bus systems.



I²C-bus structure

inventor of the technology

Online support for the I²C-bus

I²C-bus home page : www.nxp.com/i2c
 I²C-bus logic home page : www.nxp.com/i2clogic
 Boards and samples : i2c.support@nxp.com

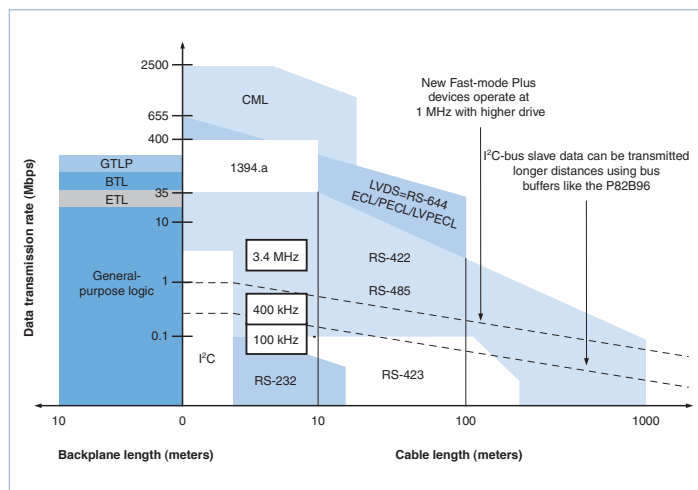
Performance, simplicity, and ease of use

There are plenty of reasons why the I²C-bus has endured for more than two decades. First, there's performance. The technology has continued to evolve and today supports several speed grades. Bus capacitance has also increased, with hubs, bus repeaters, bidirectional switches, and multiplexers that support more devices over longer distances – off the card, out of the cabinet, or even across the room.

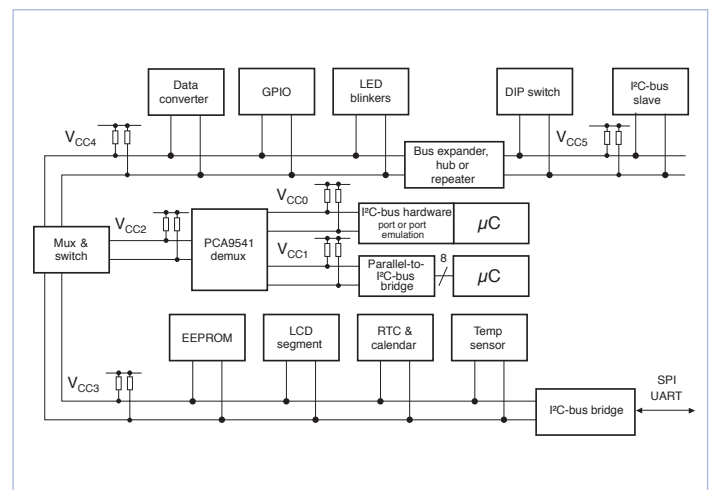
Beyond performance, though, there's ease of use. All it takes it two simple lines to connect every IC in the system. Any I²C-bus device can be attached to a common I²C-bus, and any master device can exchange information with any slave device. The software-controlled addressing scheme eliminates address-decoding hardware, and there's no need to design

and debug external control logic because it's already provided by the I²C-bus protocol. That means you can move quickly from block diagram to final hardware, simply clipping new devices and functions to an existing bus.

The I²C-bus also saves space and lowers overall cost. The two-line structure means fewer trace lines, so the PCB can be that much smaller. Having fewer trace lines simplifies debug and test, too, since there are fewer lines to check and fewer information sources to verify. What's more, as the system evolves over several generations, new I²C-bus devices can be added or removed easily, without impacting the rest of the system.

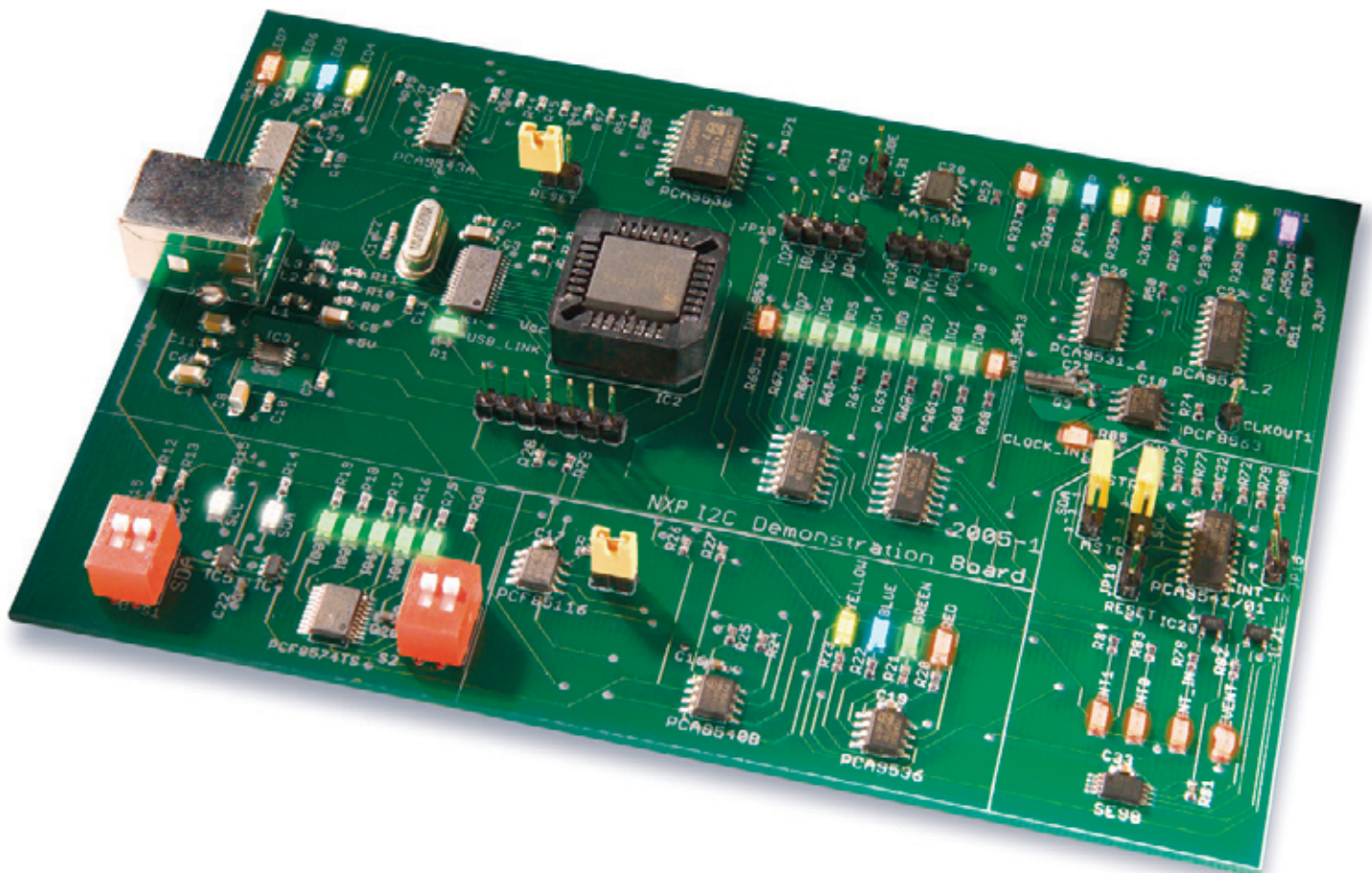


By supporting a range of performance levels, our I²C-bus devices deliver maximum design flexibility

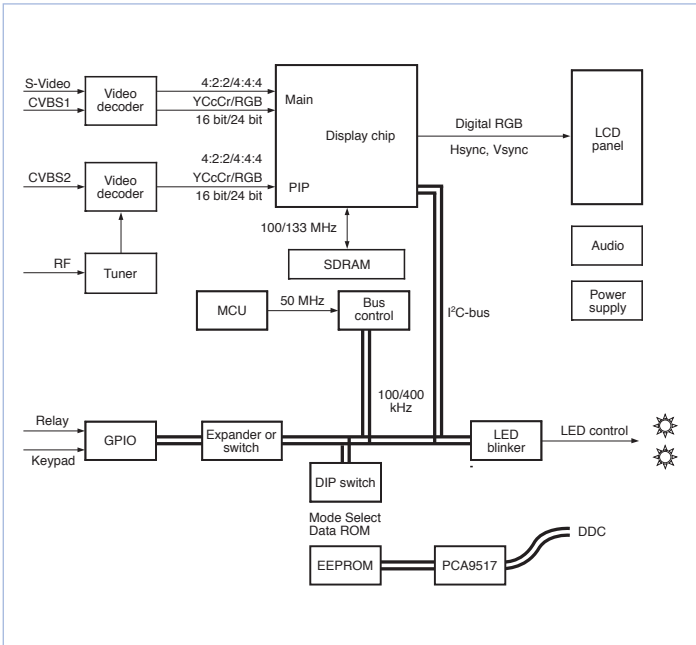
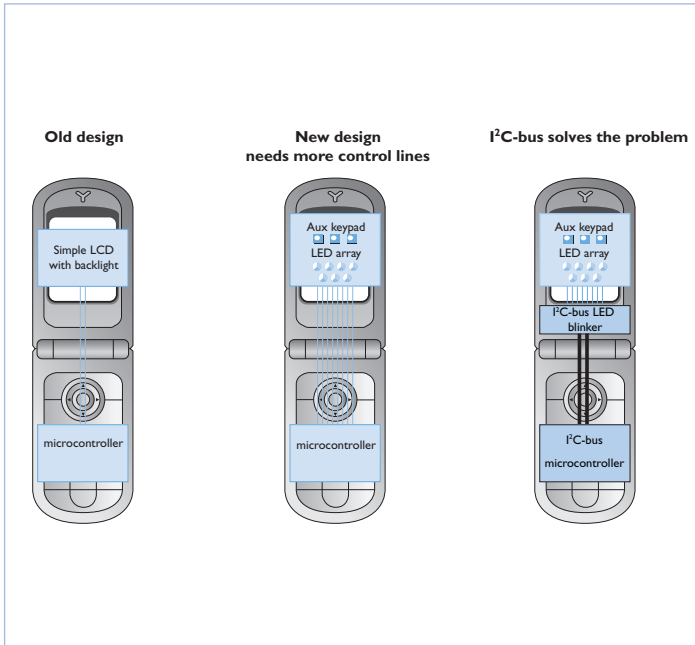


We offer the widest range I²C-bus logic

Taking I²C-bus to the next level

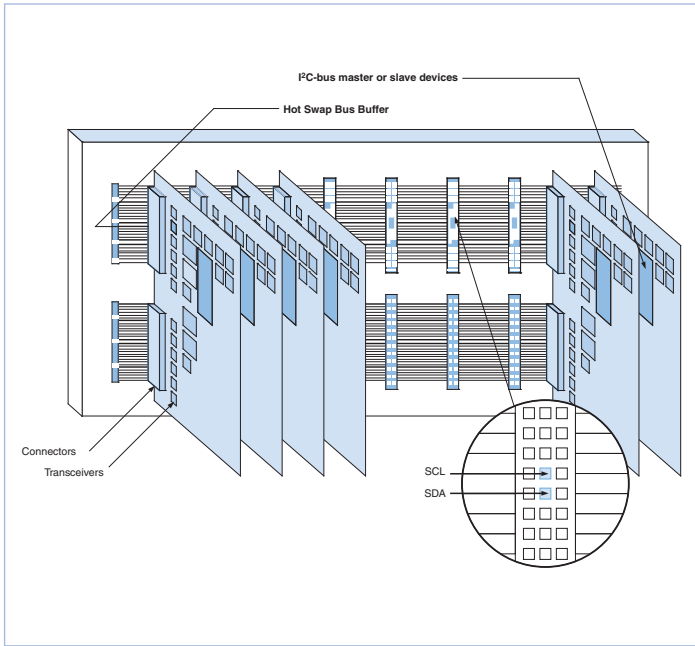


Our I2C Demonstration Board 2005-1 makes it easy to demo I²C-bus devices and learn about the I²C-bus

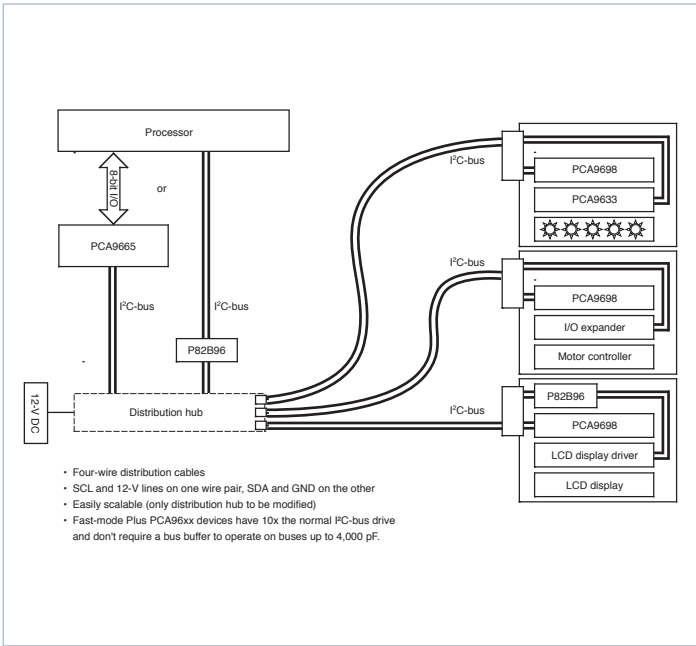


Portable device applications

LCD TV



Hot Swap applications including cPCI, VME, and advanced TCA



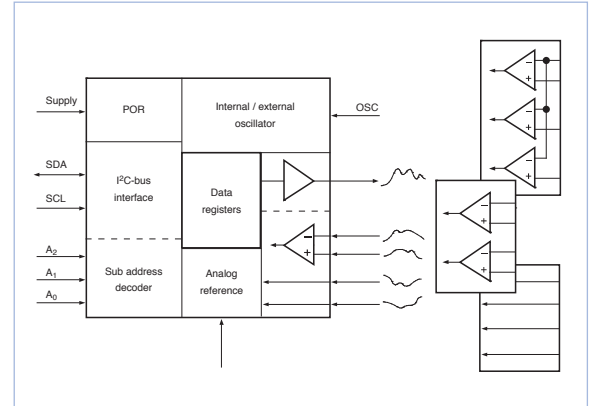
Modular and medical applications and gaming

I²C-bus product families

I²C-bus analog-to-digital converters

- ▶ Convert signals from digital to analog, analog to digital
- ▶ Internal oscillator
- ▶ Power On Reset (POR)

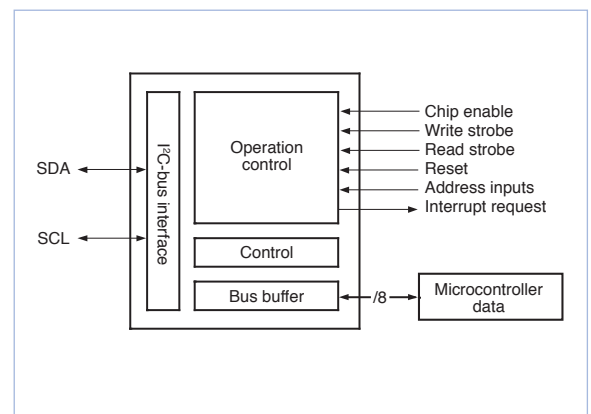
PCA9691	Fm+ 1-MHz 4-channel 8-bit A/D and D/A converter
PCF8591	100-kHz 4-channel 8-bit A/D and D/A converter
TDA8444	8-channel 6-bit D/A converter



Parallel-bus-to-I²C-bus controllers

- ▶ Interface most 8-bit parallel-bus microcontrollers and microprocessors to I²C-bus
- ▶ Parallel bus system communicates bidirectionally with I²C-bus
- ▶ Provides master and slave functions
- ▶ I²C-bus communications carried out on byte-wise basis, using interrupt or polled handshakes
- ▶ Multi-master capable
- ▶ 68-byte buffer (PCA9665)
- ▶ See page 10 for SPI or UART to I²C-bus bridges

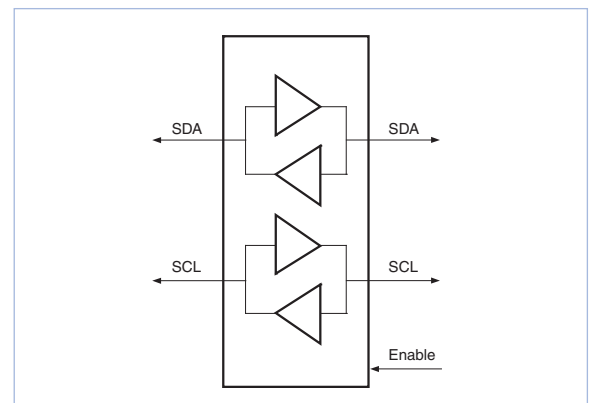
PCA9564	400-kHz 2.3-to-3.6-V I ² C-bus controller
PCA9665	Fm+ 2.3-to-3.6-V I ² C bus controller
PCF8584	100-kHz 4.5-to-5.5-V I ² C-bus controller

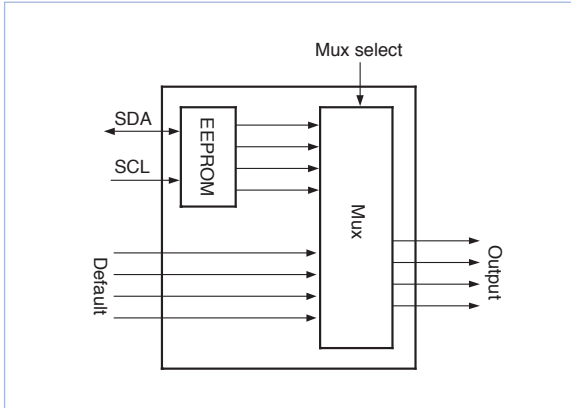


I²C-bus repeaters, hubs, and extenders

- ▶ Bidirectional I²C-bus drivers isolate I²C-bus capacitance, accommodating more I²C-bus devices and longer bus length
- ▶ Up to 400-pF load on each segment, 4000-pF for P82B devices and PCA9600
- ▶ Different voltages (3.3 or 5 V) on each segment
- ▶ Devices transparent to bus arbitration and contention protocols in multi-master environment
- ▶ Twisted-pair differential transmission or opto-electrical isolation of I²C-bus (P82B96 or PCA9600)

P82B715	I ² C-bus extender
P82B96	Dual bidirectional I ² C-bus buffer
P82B485	Differential dual bidirectional I ² C-bus buffer
P82B486	Differential triple bidirectional I ² C-bus buffer
PCA9507	I ² C-bus repeater with rise time accellerator
PCA9508	Level translating I ² C-bus repeater for cable applications
PCA9509	Low-voltage I ² C-bus repeater
PCA9510A/11A/12A/13A/14A	I ² C-bus Hot Swap bus buffers
PCA9515/15A	I ² C-bus repeater
PCA9516/16A	5-channel I ² C-bus hub
PCA9517/17A	Low-voltage I ² C-bus repeater
PCA9518A	Expandable 5-channel I ² C-bus hub
PCA9519	4-channel low-voltage I ² C-bus repeater
PCA9600	Fm+ dual bidirectional I ² C-bus buffer

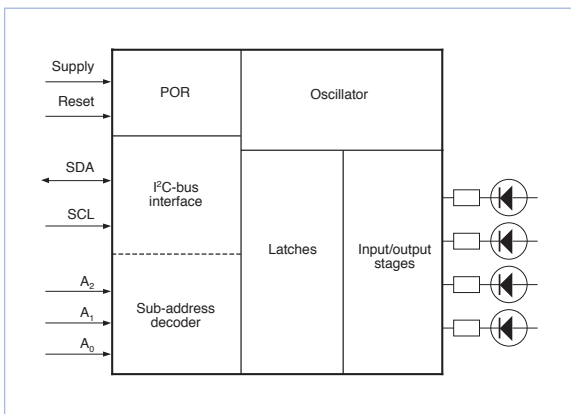




I²C-bus EEPROM-based DIP switches

- ▶ Replace jumpers and DIP switches
- ▶ Hands-free manipulation via integrated, I²C-bus-controlled EEPROM and multiplexer
- ▶ Program and store settings in I²C-bus-controlled EEPROM register
- ▶ Multiplex between default values and stored settings
- ▶ Non-volatile memory retains register values, even on power-down

PCA8550	4-bit multiplexed / 1-bit latched 5-bit I ² C-bus EEPROM
PCA9559	5-bit multiplexed / 1-bit latched 6-bit I ² C-bus EEPROM
PCA9560	Dual 5-bit multiplexed / 1-bit latched I ² C-bus EEPROM
PCA9561	Quad 6-bit multiplexed I ² C-bus EEPROM



I²C-bus LED dimmers/blinkers

- ▶ General-purpose I/O and LED control
- ▶ No external components required: internal oscillator provides two frequency and duty cycles
- ▶ Two programmable blink rates
- ▶ On/off/blink or dim LED
- ▶ LED brightness controlled by setting frequency to 152 Hz and changing duty cycle to vary LED's average current
- ▶ Extra pins can be used as inputs or outputs
- ▶ Hardware reset pin for state machine
- ▶ Open-drain outputs capable of 25-mA sink

LED dimmers (Freq. range: 152 Hz to 1.69 sec.)

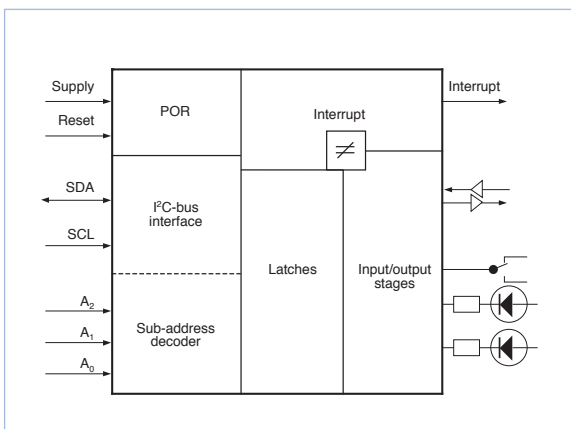
PCA9530	2-bit LED dimmer
PCA9531	8-bit LED dimmer
PCA9532	16-bit LED dimmer
PCA9533	4-bit LED dimmer
PCA9632/33	Fm+ 4-bit 5-PWM LED controller
PCA9634	Fm+ 8-bit 9-PWM LED controller
PCA9635	Fm+ 16-bit 17-PWM LED controller
PCA9685	Fm+ 16-bit 4Kstep LED controller

LED blinkers (Freq. range: 152 Hz to 1.69 sec.)

PCA9550	2-bit LED blinker
PCA9551	8-bit LED blinker
PCA9552	16-bit LED blinker
PCA9553	4-bit LED blinker
PCA9625	Fm+ LED controller (24 V, 100 mA)
SAA1064	4 x 7 I ² C-bus LED driver with decimal

I²C-bus general-purpose I/O (GPIO) expanders

- ▶ Connect parallel I/O to serial I²C-bus or SMBus and provide I/O expansion
- ▶ Quasi output: upper transistor is on for half a clock cycle and then held up by a weak current source (25-mA sink)
- ▶ Extend GPIO from I²C-bus or SMBus
- ▶ True Output: configurable as input or output with polarity inversion (push/ pull with 25-mA sink and 10-mA source)



Quasi output (weak current source)

PCA8574/74A	8-bit with interrupt
PCA8575	16-bit with interrupt
PCA9500	8-bit with 2-kbit EEPROM
PCA9501	8-bit with 2-kbit EEPROM, interrupt
PCA9558	8-bit w/ 5-bit DIP, 2-kbit EEPROM
PCA9670	Fm+ 8-bit with reset
PCA9671	Fm+ 16-bit with reset
PCA9672	Fm+ 8-bit with interrupt, reset
PCA9673	Fm+ 16-bit with interrupt, reset
PCA9674/74A	Fm+ 8-bit with interrupt
PCA9675	Fm+ 16-bit with interrupt
PCF8574/74A	8-bit with interrupt
PCF8575/75C	16-bit with interrupt

True output (configurable push/pull)

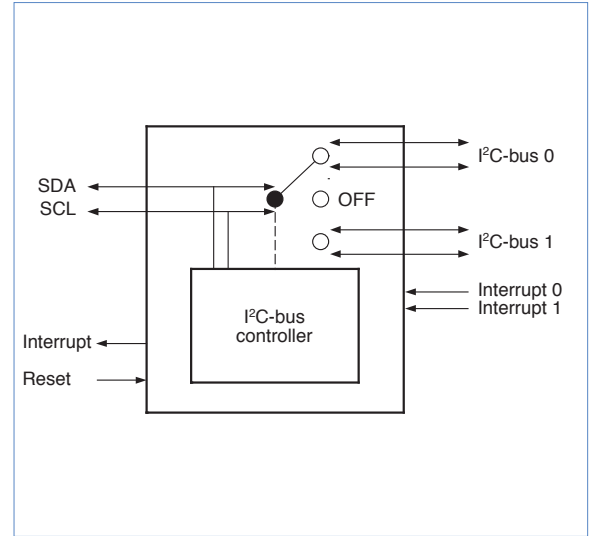
PCA9502	8-bit SPI/I ² C with interrupt, reset
PCA9505/06	40-bit with interrupt, reset, OE
PCA9534/54/54A	8-bit with interrupt
PCA9535/35C/55	16-bit with interrupt
PCA9536	4-bit
PCA9537	4-bit with interrupt, reset
PCA9538	8-bit with interrupt, reset
PCA9539	16-bit with interrupt, reset
PCA9557	8-bit with reset
PCA9698	Fm+ 40-bit with interrupt, reset, OE

I²C-bus product families

I²C-bus multiplexers and switches

- ▶ Fans I²C-bus out to multiple I²C-buses
- ▶ Buses selected via I²C-bus commands from master
- ▶ Multiplexers and switches select one downstream I²C-bus at a time
- ▶ Switches can, in broadcast mode, turn on all channels simultaneously
- ▶ Interrupts are collected and sent to master
- ▶ Reset pin deselects all channels, so master can regain control if bus hangs up
- ▶ Designed for address conflict resolution, I²C-bus sub-branch isolation, I²C-bus level shifting, etc.

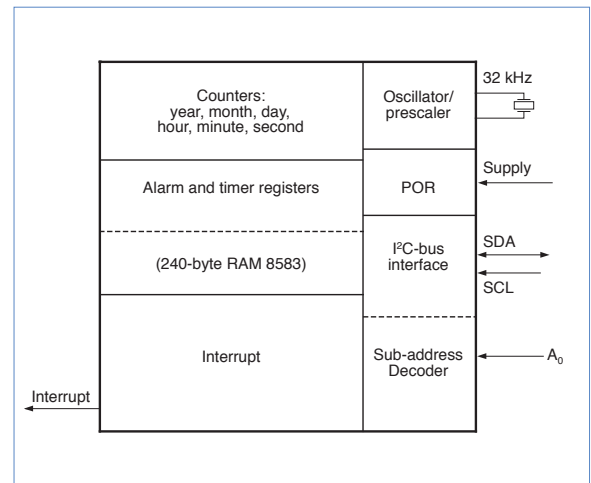
PCA9540B	1-to-2 multiplexer	PCA9545	1-to-4 switch w/interrupt, reset A/B/C
PCA9541	2-to-1 master select w/int, reset	PCA9546A	1-to-8 switch with reset
PCA9542A	1-to-2 multiplexer with interrupt	PCA9547	1-to-8 multiplexer with reset
PCA9543	1-to-2 switch w/interrupt, reset A/B/C	PCA9548A	1-to-8 switch with reset
PCA9544A	1-to-4 multiplexer w/interrupt	PCA9549	8-bit bus switch with reset



I²C-bus real-time clocks

- ▶ 32.768-kHz quartz oscillator
- ▶ Year, month, day, hour, minute, second (1/100-sec resolution)
- ▶ Alarm and timer functions
- ▶ Wide operating supply voltage range (1.0 to 5.5 V)
- ▶ Open drain interrupt pin
- ▶ Low back-up current (0.25 μ A at 3 V)
- ▶ Low-voltage detection
- ▶ Temperature compensation, integrated quartz

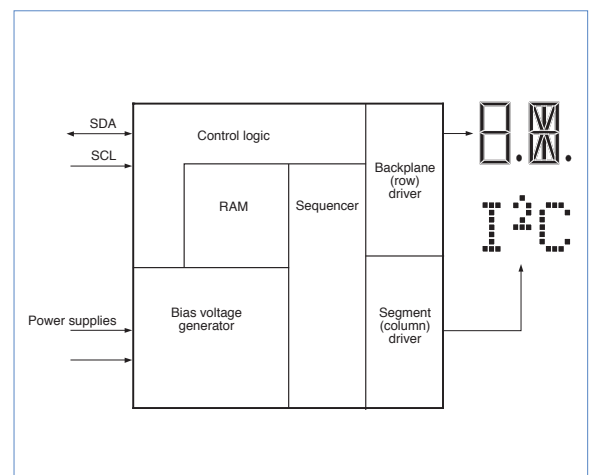
PCA8565	High-temperature (125 °C) I ² C-bus real-time clock
PCF8563	Low-power I ² C-bus real-time clock
PCF8583	I ² C real-time clock with 240-byte RAM
PCF8593	I ² C real-time clock with 1/100-second resolution
PCF2128	RTC with integrated temperature compensated quartz oscillator

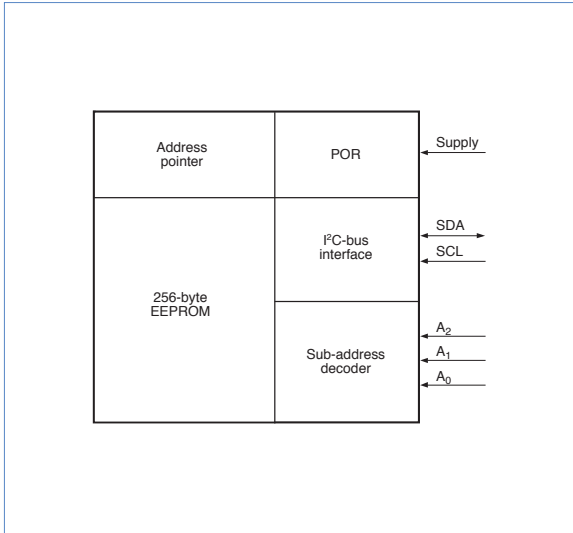


I²C-bus LCD drivers

- ▶ Low power
- ▶ On-chip bias
- ▶ Temperature compensation
- ▶ Blinking
- ▶ No external parts required

PCF2113/2116/2119	I ² C-bus LCD character display
PCF8531/8535/8811	I ² C-bus LCD graphic drivers
PCF8533	80-to-320 segment I ² C-bus LCD segment driver
PCF8534A	60-to-240 segment I ² C-bus LCD segment driver
PCF8562	32-to-128 segment I ² C-bus LCD segment driver
PCF8566	24-to-96 segment I ² C-bus LCD segment driver
PCF8576D	40-to-160 segment I ² C-bus LCD segment driver
PCF8577C	32-to-64 segment I ² C-bus LCD segment driver
PCF8578/79	>384 segment I ² C-bus LCD segment/dot driver

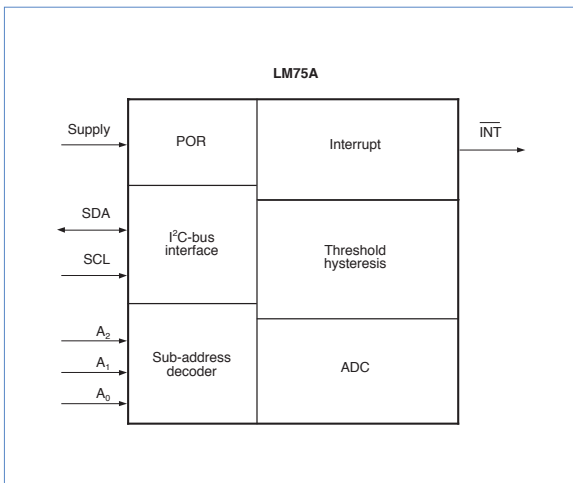




I²C-bus Serial EEPROMs

- ▶ Wide voltage range minimizes EEPROM inventory
- ▶ I²C-bus reads/writes information to/from memory
- ▶ Wide voltage range (2.5 to 5.5 V)
- ▶ 1,000,000 read/write cycles
- ▶ 10-year data retention

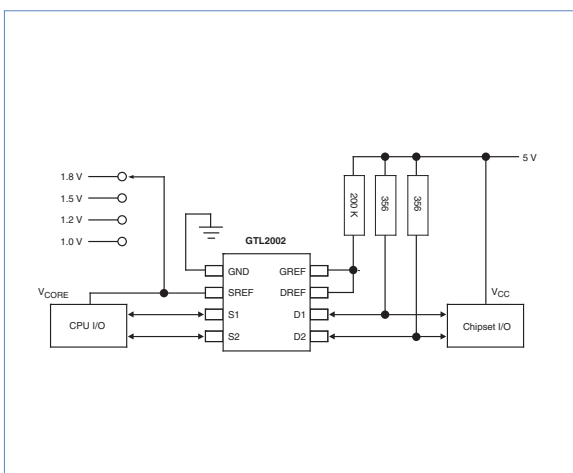
PCA24S08	1024 x 8-bit (8-kbit) I²C-bus serial EEPROM with security features
PCF8570	256 x 8-bit (2-kbit) I²C-bus static low-voltage RAM
PCF8581(C)	128 x 8-bit (1-kbit) I²C-bus serial EEPROM
PCF8582C-2	256 x 8-bit (2-kbit) I²C-bus serial EEPROM
PCF8594C-2	512 x 8-bit (4-kbit) I²C-bus serial EEPROM
PCF8598C-2	1024 x 8-bit (8-kbit) I²C-bus serial EEPROM
PCF85102C-2	256 x 8-bit (2-kbit) I²C-bus serial EEPROM
PCF85103C-2	256 x 8-bit (2-kbit) I²C-bus serial EEPROM with alternate I²C address
PCF85116-3	2048 x 8-bit (16-kbit) I²C-bus serial EEPROM



I²C-bus temperature and voltage sensors

- ▶ Hardware monitors use I²C-bus to report temperature and/or voltage
- ▶ Remote sensor is more accurate than package-mounted sensors (analog input pins let external transistors and diodes – in a processor, for example – perform very precise temperature sensing)
- ▶ SE97/98 per JEDEC specification for RDIMM and SO-DIMM applications

LM75A/B	± 2 °C local I²C-bus temperature sensor and thermal watchdog™
NE1617A	± 2 °C local / remote I²C-bus temperature monitor
NE1619	± 2 °C local / remote I²C-bus temperature and voltage monitor
SA56004	± 1 °C local / remote digital temp sensor with over-temp alarms
SE95	± 1 °C high accuracy local I²C-bus temperature sensor and thermal Watchdog
SE97	± 1 °C local I²C-bus temperature sensor with 2-Kbit EEPROM SPD
SE98	± 1 °C local I²C-bus temperature sensor



I²C-bus voltage-level translators

- ▶ Voltage translation between any voltage from 1.0 to 5.0 V
- ▶ Open drain on both sides with no drive
- ▶ Bidirectional voltage translation with no direction pin
- ▶ Reference voltage clamps input voltage with low propagation delay
- ▶ Supports bidirectional translation of normal 3.3- and/or 5.0-V I²C-bus signals to low-voltage I²C-bus processor ports
- ▶ BiCMOS process for excellent ESD performance (except PCA9306)

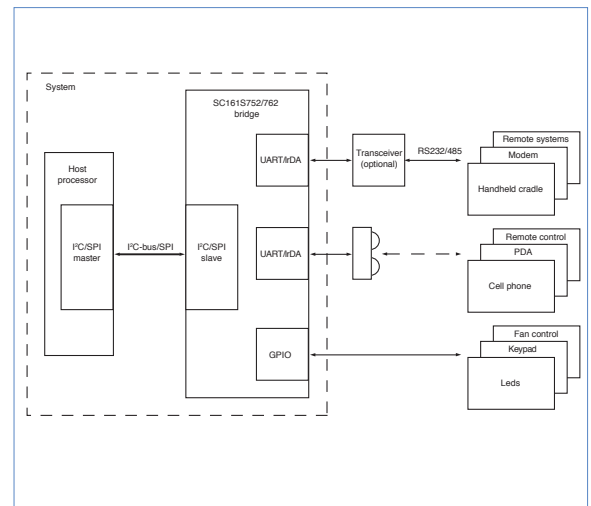
GTL2000	22-bit bidirectional I²C-bus voltage-level translator
GTL2002	2-bit bidirectional I²C-bus voltage-level translator
GTL2003	8-bit bidirectional I²C-bus voltage-level translator
GTL2010	10-bit bidirectional I²C-bus voltage-level translator
PCA9306	Bidirectional I²C-bus and SMBus voltage-level translator

I²C-bus product families

High-speed I²C-bus / SPI-slave bridge to UART / IrDA / GPIO

- ▶ Communication between SPI / I²C-bus slave and high-speed UART
- ▶ Low power, low voltage
- ▶ Minimal host connection
- ▶ Broadband UART (5 Mbps)
- ▶ Enable communication between I²C-bus or SPI and RS232 or RS485
- ▶ IrDA interface for wireless links
- ▶ Configurable GPIO
- ▶ SPI speed: 4 to 15 Mbps
- ▶ I²C-bus speed: 400 kbps
- ▶ High data throughput and integrity

SC16IS750	I ² C-bus / SPI-slave bridge to single UART with 64-byte FIFOs, 115.2-kbps IrDA SIR, 8 GPIO
SC16IS752	I ² C-bus / SPI-slave bridge to dual UARTs with 64-byte FIFOs, 115.2-kbps IrDA SIR, 8 GPIO
SC16IS760	I ² C-bus / SPI-slave bridge to single UART with 64-byte FIFOs, 1.152-Mbps IrDA SIR, 8 GPIO
SC16IS762	I ² C-bus / SPI-slave bridge to dual UARTs with 64-byte FIFOs, 1.152-Mbps IrDA SIR, 8 GPIO
SC16IS740	I ² C-bus / SPI-slave bridge to single UART with 64-byte FIFOs, 115.2-kbps IrDA SIR
SC16IS850	I ² C-bus / SPI-slave bridge to single UART with 128-byte FIFOs, 115.2-kbps IrDA SIR
SC16IS852	I ² C-bus / SPI-slave bridge to dual UART with 64-byte FIFOs, 115.2-kbps IrDA SIR

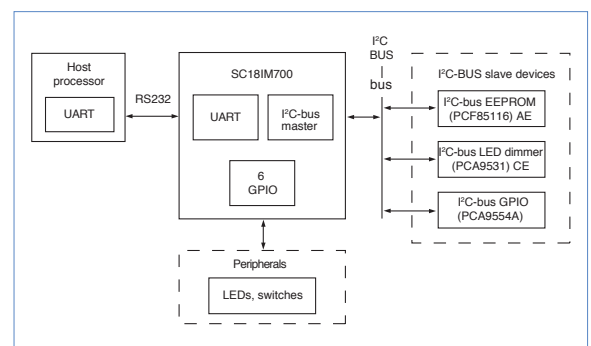


SC16is7xx usage scenarios

UART bridge to I²C-bus master with GPIO

- ▶ Control connection between UART host and I²C-bus
- ▶ Low power, low voltage
- ▶ Fast-mode 400 kHz I²C-bus device interface
- ▶ Programmable GPIO and baud rate
- ▶ Enables UART connectivity to the I²C-bus
- ▶ High data throughput and integrity
- ▶ Multi master capable

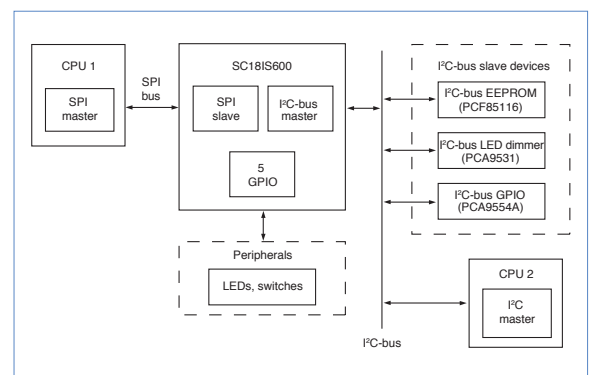
SC18IM700	UART bridge to I ² C-bus master with GPIO
-----------	--



SPI-slave bridge to I²C-bus master / GPIO

- ▶ Control connection between SPI host and I²C-bus
- ▶ Low power, low voltage
- ▶ Full-duplex high-speed SPI interface
- ▶ SPI speed: 1 to 3 Mbps
- ▶ Fast I²C-bus device interface
- ▶ Enables communication between SPI and I²C-bus
- ▶ Programmable GPIO
- ▶ High data throughput and integrity

SC18IS600	SPI-slave bridge to I ² C master with GPIO, internal clock
SC18IS601	SPI-slave bridge to I ² C master with GPIO, external clock
SC18IS602	I ² C-bus-slave bridge to SPI master with GPIO, external clock



Development kits for bridge ICs

I ² C-bus / SPI to UART (SC16IS7xx)	Easy interface between an I ² C-bus / SPI host and IrDA, RS232 / RS485, and GPIO devices. The bus interface is selectable for I ² C-bus or SPI. Kit includes sample code with device initialization (UART, RS485) and IrDA communications, plus user manual.
UART to I ² C-bus (SC18IM700)	Easy interface between a UART host and various I ² C-bus and GPIO devices. Kit includes sample code with SC18IM700 UART messages and commands for NXP I ² C-bus slave devices, plus user manual.
SPI to I ² C-bus (SC18IS60x)	Easy interface between an SPI host and various I ² C-bus and GPIO devices. The board also includes an I ² C-bus EEPROM and an LED dimmer. Kit includes sample code with SC18IS80x SPI messages and commands for NXP I ² C-bus slave, plus user manual.

I²C-bus FAQs

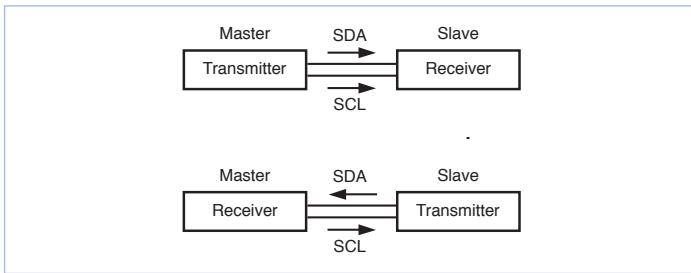
Problem	Solution	Suggested parts
Putting too many of the same devices on the bus causes an address conflict.	Use a multiplexer to break address space into sub-branches. Master selects which sub-branch to address under I ² C-BUS control. Two to eight sub-branches are available per multiplexer.	PCA9540B, 41, 42A, 43A/B/C, 44A, 45A/B/C, 46A, 47, 48A, 49
I need more devices than the 400-pF lead allows.	Use a multiplexer to break up the capacitance by sub-branch. Sub-branches are addressed individually.	PCA9540B, 41, 42A, 43A/B/C, 44A, 45A/B/C, 46A, 47, 48A,49
	Use a repeater or hub to address all devices at the same time (be careful of address conflicts).	PCA9507, 08, 09, 15, 15A, 16, 16A, 17, 17A, 18A, 19
	Use Fast-mode Plus devices.	PCA9632, 33, 34, 35, 70, 71, 72, 73, 74, 74A, 75, 98, with PCA9600 from master or PCA9665
I have to send I ² C-bus signals through a really long cable.	Use an I ² C-bus extender to add high drive to any I ² C-bus. Guidelines (80 pF per meter) for 400 kHz are at least 20 m and for 30 kHz are at least 1000 m.	P82B715, P82B96, and PCA9600 with 30-mA drive PCA9507, 08, 17, 17A with 6-mA drive
I have a mixed-voltage environment that causes conflicts on the open-drain I ² C-bus.	Use a switch to dynamically split the bus by pulling downstream channels to different voltages.	PCA9543A/B/C, 45A/B/C, 46A, 48A GTL2002, 2010, 2000, PCA9306 PCA9507, 08, 09, 12A, 17, 17A, 19
I have slow (100-kHz) and fast (400-kHz) parts on the same bus. How can I isolate them to get the most performance from the 400-kHz devices?	Use a bus repeater and have the master disable the 100-kHz segment during 400-kHz communication.	PCA9508, 15, 15A, 16, 16A, 17, 17A, 18A
		PCA9540B, 42A, 43A/B/C
I have two masters in my fault-tolerant system, one active at a time. How do I select one master over the other?	Use an I ² C-bus master selector to switch from a failing master to its backup. The selector also provides isolation.	PCA9541
I support Hot Swap in my fault-tolerant system. How can I keep I ² C-bus traffic undisturbed during a swap?	Use an I ² C-bus Hot Swap bus buffer to detect bus-idle condition, isolate capacitance, and prevent bus glitches.	PCA9510A, 11A, 12A, 13A, 14A PCA9508
I need to add a new I ² C-bus port to a host.	Use a bus controller to create a new I ² C-bus port. You may not need a new port. A multiplexer or a switch may give you the features you need.	PCA9564, PCF8584, PCA9665 PCA954x
The I ² C-bus is "multi-drop" and any device can hang the bus. How can I make this bus more fault-tolerant?	Break the bus into different branches to create a "star" configuration. Multi-cast is allowed. The master sets the configuration via I ² C-bus.	PCA9543A/B/C, 45A/B/C, 46A, 48A Simple discrete circuitry with any buffer can detect and isolate failed sections.

How the I²C-bus works

I²C-bus terminology

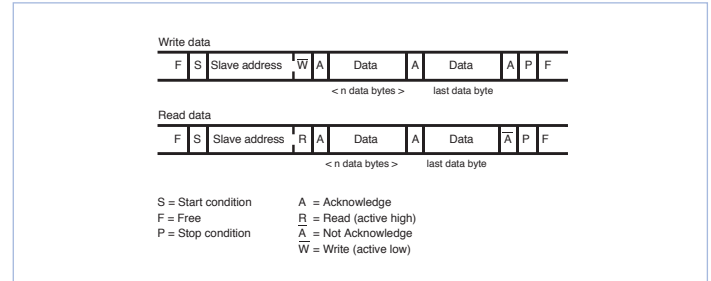
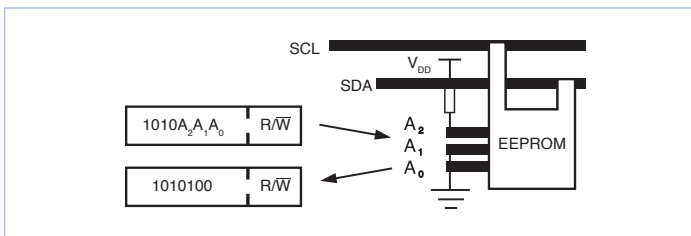
All the ICs along an I²C-bus are either masters or slaves. A master is an IC that initiates a data transfer, provides the serial clock signal (SCL) during the transfer, and then terminates the transfer. There can be more than one master on the bus at a time – the I²C software protocol uses arbitration and synchronization to prevent collisions and data loss. A slave is any device addressed by the master.

While the master or slave is sending data over the serial data line (SDA), it is referred to as a transmitter. While accepting data from the SDA data line, it's called a receiver. A master transmitter can put data onto the SDA data line at any time; a slave transmitter can do so only by request.



Device addresses

Each device along the I²C-bus has a unique, 7-bit I²C-bus address. The first four bits in the address are typically fixed, indicating device category (e.g. 1010 is assigned to EEPROMs). The last three bits (A₂, A₁, A₀) are set by hardware address pins on the IC package. In this case, there are up to eight different address combinations available for each device, so up to eight identical devices can operate on the same I²C-bus. The address pins are held high (1) to V_{CC} or held low (0) to GND.



The data-transfer bitstream

"F" indicates the bus is Free, or available for use. During the Free state the serial data line (SDA) is transferring no data and both the SDA data line and the SCL clock line are high.

To take control of the bus and initiate a transaction, the master sends out a Start (S) condition. If the master already controls the bus and wants to initiate an additional transaction, it sends out a Repeat Start (Sr) condition. For the start condition, the SDA data line goes from high to low while the SCL clock remains high.

The Start condition is followed by the desired slave address and the instruction to write (\bar{W}) or read (R) data.

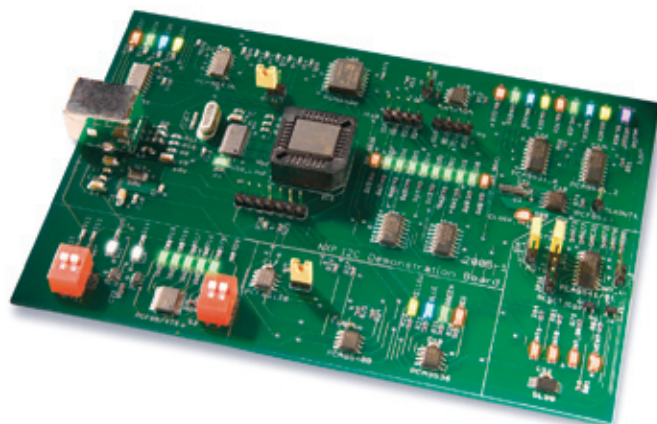
The slave responds to the master by sending an Acknowledge (A) bit and the transfer begins. During data transfer, the SCL clock line is brought and held low; the SDA data line can be either high or low. The SDA data line only changes state when the SCL clock line is low.

After each byte is sent, the receiver will Acknowledge (A) the transmitter. When the master is the receiver, it will Not Acknowledge (\bar{A}) the last byte so it can regain control of the bus to terminate the communication. When the master has completed the transmission, it sends the Stop (P) condition to free the bus. For the Stop condition, the SDA data line goes from low to high while the SCL clock line remains high.

With the data transfer complete, the bus returns to the Free state, with both lines high.

I²C vs. SMBus

Developed by Intel in the mid-1990s, the System Management Bus, also as the SMBus, is a popular derivative of the I²C-bus that is, in most cases, compatible with I²C-bus. Both buses use a two-wire communication scheme and have addressable slaves. The SMBus is limited to a maximum data transfer rate of only 100 kbps, requiring special handling in systems that also use the higher transfer rates available with I²C-bus. Other differences include timeout and minimum clock speed, voltage levels, pull-up resistor values, and current levels.



Visit the I²C-bus homepage at
www.nxp.com/i2clogic

Order evaluation kits, get technical
information, or request samples by
contacting us at
i2c.support@nxp.com



www.nxp.com

© 2007 NXP B.V.

All rights reserved. Reproduction in whole or in part is prohibited without the prior written consent of the copyright owner. The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice. No liability will be accepted by the publisher for any consequence of its use. Publication thereof does not convey nor imply any license under patent- or other industrial or intellectual property rights.

Date of release: December 2007

Document order number: 9397 750 15669

Printed in the Netherlands

