Excellent Integrated System Limited

Stocking Distributor

Click to view price, real time Inventory, Delivery & Lifecycle Information:

Maxim Integrated
DS28CZ04G-4+

For any questions, you can email us directly:
sales@integrated-circuit.com

DS28CZ04

4Kb I²C/SMBus EEPROM with Nonvolatile PIO

www.maxim-ic.com

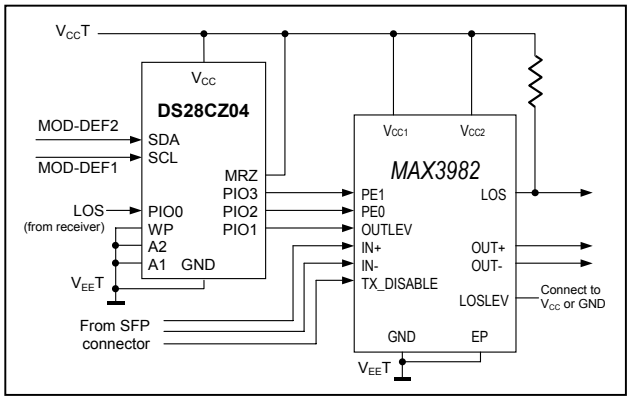
GENERAL DESCRIPTION

The DS28CZ04 combines 4Kb (512 x 8) EEPROM with 4 PIO lines. Communication with the device is accomplished with an industry standard I²C and SMBus™ interface. The memory is organized as two segments of 256 bytes with single byte and up to 16-byte block write capability. Individual PIO lines may be configured as inputs or outputs. The power-on state of PIO programmed as outputs is stored in non-volatile memory. All PIO may be reconfigured by the user through the serial interface.

APPLICATIONS

- 4G SFP Copper Modules
- SFF-8472, SFP Fiber Modules
- RAID Systems
- Servers

TYPICAL OPERATING CIRCUIT



Small Form-factor Pluggable (SFP) Circuit

SMBus is a trademark of Intel Corp.

FEATURES

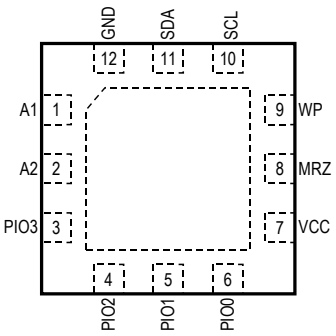
- 4Kb (512 x 8) EEPROM Organized in Two 256-Byte Blocks
- Single Byte and up to 16-Byte EEPROM Write Sequences
- Write-Protect Control Pin for the Entire EEPROM Array
- Endurance 200k Cycles per Block at 25°C; 10ms max EEPROM Write Cycle
- 4 PIO Lines
- Each PIO is Configured to Input or Output Mode on Startup by Stored Value
- All PIOs are Reconfigurable after Startup
- Serial Interface User-Programmable for I²C Bus and SMBus Compatibility
- Supports 100kHz and 400kHz I²C Communication Speeds
- Operating Range: 2.0V to 5.25V, -40°C to +85°C
- 4mm x 4mm 12-Pin TQFN Package

ORDERING INFORMATION

PART	TEMP RANGE	PIN-PACKAGE
DS28CZ04G-4+	-40°C to +85°C	TQFN12-EP* 4x4mm²
DS28CZ04G-4+T	-40°C to +85°C	TQFN12-EP* 4x4mm² Tape-and-Reel

**EP = Exposed Paddle*
+Denotes lead-free package.

PIN CONFIGURATION



Thin 12-Lead 4mm × 4mm QFN (Top View)
Package Outline Drawing 21-0139
Package Code T1244+4

Note: *Some revisions of this device may incorporate deviations from published specifications known as errata. Multiple revisions of any device may be simultaneously available through various sales channels. For information about device errata, click here: www.maxim-ic.com/errata.*

ABSOLUTE MAXIMUM RATINGS

Voltage Range on Any Pin Relative to Ground	-0.5V, +6V
Maximum Current SDA, SCL, A2, A1, WP, MRZ Pin	±20mA
Maximum Current each PIO Pin	±20mA
Maximum GND and V_{CC} Current	100mA
Operating Temperature Range	-40°C to +85°C
Junction Temperature	+150°C
Storage Temperature Range	-55°C to +125°C
Soldering Temperature	See IPC/JEDEC J-STD-020

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to the absolute maximum rating conditions for extended periods may affect device.

ELECTRICAL CHARACTERISTICS

(-40°C to +85°C, see Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Supply Voltage	V_{CC}		2.0		5.25	V
Standby Current (Note 2)	I_{CCS}	Bus idle, V_{CC} = 5.25V		1.5	4	µA
Operating Current	I_{CCA}	Bus active at 400kHz, V_{CC} = 5.25V		250	500	µA
Programming Current	I_{PROG}	V_{CC} = 5.25V		500	1000	µA
Power-up Wait Time	t_{POIP}	(Note 3)			100	µs
EEPROM						
Programming Time	t_{PROG}				10	ms
Endurance	N_{CYCLE}	At +25°C (Notes 4, 5)	200k			—
Data Retention	t_{RET}	At +85°C (Notes 5, 6)	40			years
PIO Pins, See Figures 8, 9						
LOW-Level Output Voltage	V_{OL}	1mA sink current	0		0.4	V
HIGH-Level Output Voltage	V_{OH}	500µA source current	V_{CC} - 0.5V			V
LOW-Level Input Voltage	V_{IL}		-0.3		0.3 × V_{CC}	V
HIGH-Level Input Voltage	V_{IH}		0.7 × V_{CC}		V_{CC} + 0.3V	V
Output Data Valid Time	t_{PV}				1	µs
PIO Read Setup Time	t_{PS}	(Note 5)	150			ns
PIO Read Hold Time	t_{PH}	(Note 5)	150			ns
Leakage Current	I_L	High Impedance, at V_{CCMAX}	-1		+1	µA
SCL, SDA, A2, A1, WP, MRZ Pins (Note 7), See Figure 6						
LOW Level Input Voltage	V_{IL}		-0.3		0.3 × V_{CC}	V
HIGH Level Input Voltage	V_{IH}	(Note 8)	0.7 × V_{CC}		V_{CCmax} + 0.3V	V
Hysteresis of Schmitt Trigger Inputs	V_{hys}	(Notes 5, 9)	0.05 × V_{CC}			V
LOW Level Output Voltage	V_{OL}	At 4mA Sink Current, open drain			0.4	V
Output Fall Time from V_{Ihmin} to V_{ILmax} (Notes 5, 10)	t_{of}	Bus Capacitance from 10pF to 400pF	20 + 0.1C_B		250	ns
Pulse Width of Spikes that are Suppressed by the Input Filter	t_{SP}	SDA and SCL pins only (Note 5)			50	ns

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Input Current with an Input Voltage Between $0.1V_{CC}$ and $0.9V_{CCmax}$	I_I	(Note 11)	-10		10	µA
Input Capacitance	C_I	(Notes 5, 9)			10	pF
SCL Clock Frequency	f_{SCL}	(Note 12)			400	kHz
Bus Time-Out	$t_{TIMEOUT}$	(Note 12)	25		75	ms
Hold Time (Repeated) START Condition. After this Period, the First Clock Pulse is Generated.	$t_{HD:STA}$	(Note 13)	0.6			µs
LOW Period of the SCL Clock (Note 13)	t_{LOW}	$V_{CC} \geq 2.7V$	1.3			µs
		$V_{CC} < 2.7V$	1.5			
HIGH Period of the SCL Clock	t_{HIGH}	(Note 13)	0.6			µs
Setup Time for a Repeated START Condition	$t_{SU:STA}$	(Note 13)	0.6			µs
Data Hold Time (Notes 14, 15)	$t_{HD:DAT}$	$V_{CC} \geq 2.7V$	0.3		0.9	µs
		$V_{CC} < 2.7V$	0.3		1.1	
Data Setup Time	$t_{SU:DAT}$	(Notes 13, 16)	100			ns
Setup Time for STOP Condition	$t_{SU:STO}$	(Note 13)	0.6			µs
Bus Free Time Between a STOP and START Condition	t_{BUF}	(Note 13)	1.3			µs
Capacitive Load for Each Bus Line	C_B	(Notes 5, 13)			400	pF

Note 1: Specifications at -40°C are guaranteed by design and characterization only and not production tested.
Note 2: To the first order, this current is independent of the supply voltage value.
Note 3: All PIO are tri-stated at beginning of reset prior to setting to Power-On values.
Note 4: This specification is valid for each 16-byte memory block.
Note 5: Not production tested. Guaranteed by design or characterization.
Note 6: EEPROM writes can become nonfunctional after the data-retention time is exceeded. Long-time storage at elevated temperatures is not recommended; the device can lose its write capability after 10 years at +125°C or 40 years at +85°C.
Note 7: All values are referenced to V_{IHmin} and V_{ILmax} levels.
Note 8: The maximum specification value is guaranteed by design, not production tested.
Note 9: Applies to SDA and SCL.
Note 10: C_B = total capacitance of one bus line in pF. If mixed with HS-mode devices, faster fall-times according to I²C-Bus Specification v2.1 are allowed.
Note 11: The DS28CZ04 does not obstruct the SDA and SCL lines if V_{CC} is switched off.
Note 12: The minimum SCL clock frequency is limited by the bus timeout feature. If the CM bit is 1 AND SCL stays at the same logic level or SDA stays low for this interval, the DS28CZ04 behaves as though it has sensed a STOP condition.
Note 13: System Requirement
Note 14: The DS28CZ04 provides a hold time of at least 300ns for the SDA signal (referred to the V_{IHmin} of the SCL signal) to bridge the undefined region of the falling edge of SCL.
Note 15: The maximum $t_{HD:DAT}$ has only to be met if the device does not stretch the low period (t_{LOW}) of the SCL signal.
Note 16: A Fast-mode I²C-bus device can be used in a standard-mode I²C-bus system, but the requirement $t_{SU:DAT} \geq 250ns$ must then be met. This is automatically the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_{rmax} + t_{SU:DAT} = 1000 + 250 = 1250ns$ (according to the standard-mode I²C-bus specification) before the SCL line is released.

PIN DESCRIPTION

PIN	NAME	FUNCTION
1	A1	Device Address Bit 1
2	A2	Device Address Bit 2
3	PIO3	PIO line #3
4	PIO2	PIO line #2
5	PIO1	PIO line #1
6	PIO0	PIO line #0
7	V_{CC}	Power Supply Input
8	MRZ	Master Reset (active-low). Performs a reset of the serial interface and the PIOs without power-cycling the device.
9	WP	Write Protect input, to be connected to V_{CC} or GND. When connected to V_{CC}, the entire EEPROM array is write-protected. Normal read/write access when connected to GND. Changing the pin state during a write access will cause unpredictable results.
10	SCL	I²C/SMBus serial clock input; must be tied to V_{CC} through a pullup resistor.
11	SDA	I²C/SMBus bidirectional serial data line; must be tied to V_{CC} through a pullup resistor.
12	GND	Ground supply for the device.
EP	GND	Exposed Paddle. Solder evenly to the board's ground plane for proper operation. See Application Note 3273 for additional information.

OVERVIEW

The DS28CZ04 consists of a serial I²C/SMBus interface, 4Kb of EEPROM and four bidirectional PIO channels, as shown in the block diagram in Figure 1. The device communicates with a host processor through its I²C interface in standard-mode or in fast-mode; the user can switch the interface from I²C bus to SMBus mode. Two address pins allow 4 DS28CZ04 to reside on the same bus segment. A Master reset pin permits a full device reset without power cycling.

The device has a memory range of 512 bytes, organized as two segments (lower half, upper half) of 256 bytes (Figure 2). The memory map and device addressing is compatible with SFF-8472 Digital Diagnostic address assignments. The entire EEPROM can be write-protected by tying the WP pin to V_{CC}. The PIO pins can be accessed through one address (= single-address mode) or through separate addresses (= multi-address mode). PIO direct access addressing allows fast generation of data patterns and fast sampling.

The DS28CZ04 includes several EEPROM registers for the user to select whether the device powers up in SFF mode and to define the power-on default conditions for individual PIO output state (high, low, in output mode), individual PIO data direction (in, out), individual PIO output type (push-pull, open drain), individual PIO read bit inversion (true, false). Once powered up, the PIO settings can be overwritten through SRAM registers without affecting the power-on defaults.

Figure 1. Block Diagram

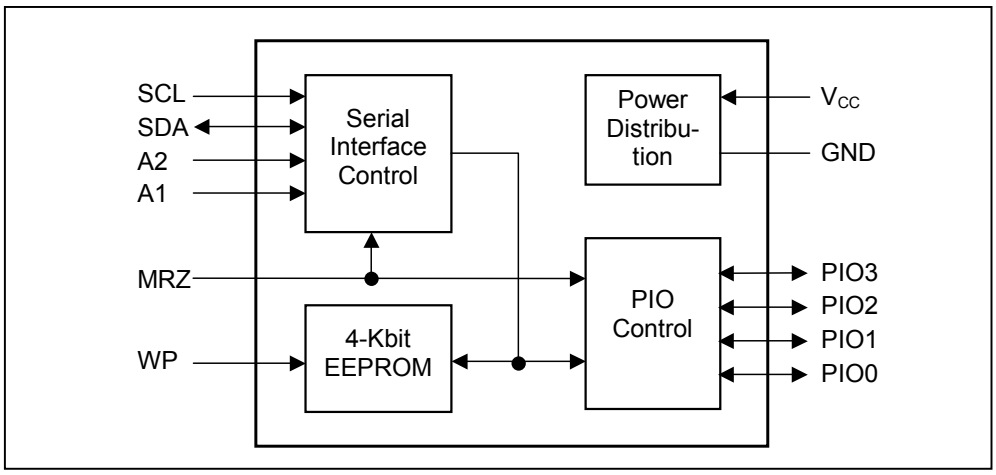


Figure 2A. Memory Map (Device Address = A0h)

ADDRESS	TYPE	ACCESS	DESCRIPTION
00h to 74h	EEPROM	R/W	User memory
75h	EEPROM	R/W	Special function/user memory; controls whether device powers-up into SFF Mode
76h	EEPROM	R/W	Power-on default for PIO output state and direction for all PIOs
77h	EEPROM	R/W	Power-on default for PIO output type and read-inversion for all PIOs
78h to 79h	—	R	Reserved (reads FFh)
7Ah	SRAM	R/W	Direction setting for all PIOs and device control/status register
7Bh	SRAM	R/W	PIO read-inversion and PIO output type for all PIOs
7Ch to 7Fh	SRAM	R/W	PIO Read/Write Access Registers
80h to FFh	EEPROM	R/W	User memory

Figure 2B. Memory Map (Device Address = A2h)

ADDRESS	TYPE	ACCESS	DESCRIPTION
00h to 6Dh	EEPROM	R/W	User memory
6Eh	EEPROM	R/W	SFF Mode off: User memory
	—	R	SFF Mode on: SFF Optional Status Register
6Fh to EFh	EEPROM	R/W	User memory
F0h to FFh	—	R	Reserved (reads FFh)

DETAILED REGISTER DESCRIPTIONS

Special Function/User Memory (Device Address = A0h)

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
75h	1	0	1	0	1	0	1	0

There is general read and write access to this address. If programmed to AAh, as shown in the bit pattern above, the SFF Mode bit at memory address 7Ah (Device Address = A0h) will be set to 1 after the next power-up, activating SFF mode with memory address 6Eh (device address A2h) functioning as the SFF Optional Status Register. **Factory-default: 00h**

Power-on Default for PIO Output State and Direction (Device Address = A0h)

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
76h	POD3	POD2	POD1	POD0	POV3	POV2	POV1	POV0

There is general read and write access to this address. **Factory-default: F0h**

BIT DESCRIPTION	BIT(S)	DEFINITION
POV0: Power-On State PIO0	b0	Power-on default output state of PIO0
POV1: Power-On State PIO1	b1	Power-on default output state of PIO1
POV2: Power-On State PIO2	b2	Power-on default output state of PIO2
POV3: Power-On State PIO3	b3	Power-on default output state of PIO3
POD0: Power-On Direction PIO0	b4	Power-on default direction of PIO0; 0 ⇒ output, 1 ⇒ input
POD1: Power-On Direction PIO1	b5	Power-on default direction of PIO1; 0 ⇒ output, 1 ⇒ input
POD2: Power-On Direction PIO2	b6	Power-on default direction of PIO2; 0 ⇒ output, 1 ⇒ input
POD3: Power-On Direction PIO3	b7	Power-on default direction of PIO3; 0 ⇒ output, 1 ⇒ input

Power-on Default for PIO Output Type and Read Inversion (Device Address = A0h)

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
77h	POT3	POT2	POT1	POT0	PIM3	PIM2	PIM1	PIM0

There is general read and write access to this address. **Factory-default: F0h**

BIT DESCRIPTION	BIT(S)	DEFINITION
PIM0: Power-On Read Inversion PIO0	b0	Power-on default state of read-inversion bit of PIO0; 0 ⇒ no inversion, 1 ⇒ inversion
PIM1: Power-On Read Inversion PIO1	b1	Power-on default state of read-inversion bit of PIO1; 0 ⇒ no inversion, 1 ⇒ inversion
PIM2: Power-On Read Inversion PIO2	b2	Power-on default state of read-inversion bit of PIO2; 0 ⇒ no inversion, 1 ⇒ inversion
PIM3: Power-On Read Inversion PIO3	b3	Power-on default state of read-inversion bit of PIO3; 0 ⇒ no inversion, 1 ⇒ inversion
POT0: Power-On Output Type PIO0	b4	Power-on default output type of PIO0; 0 ⇒ push-pull, 1 ⇒ open drain
POT1: Power-On Output Type PIO1	b5	Power-on default output type of PIO1; 0 ⇒ push-pull, 1 ⇒ open drain
POT2: Power-On Output Type PIO2	b6	Power-on default output type of PIO2; 0 ⇒ push-pull, 1 ⇒ open drain
POT3: Power-On Output Type PIO3	b7	Power-on default output type of PIO3; 0 ⇒ push-pull, 1 ⇒ open drain

Direction and Control/Status Register (Device Address = A0h)

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
7Ah	ADMD	CM	BUSY	SFF	DIR3	DIR2	DIR1	DIR0

There is general read and write access to this address. Bit 5 is read-only. The power-on default of bits 0 to 3 is copied from memory address 76h (Device Address = A0h) bits 4 to 7, respectively.

BIT DESCRIPTION	BIT(S)	DEFINITION
DIR0: Direction PIO0	b0	Direction of PIO0; 0 ⇒ output, 1 ⇒ input
DIR1: Direction PIO1	b1	Direction of PIO1; 0 ⇒ output, 1 ⇒ input
DIR2: Direction PIO2	b2	Direction of PIO2; 0 ⇒ output, 1 ⇒ input
DIR3: Direction PIO3	b3	Direction of PIO3; 0 ⇒ output, 1 ⇒ input
SFF: SFF Mode Bit	b4	SFF Mode control; 0 ⇒ SFF Mode off, 1 ⇒ SFF Mode on. See Memory Map (Device Address = A2h) and SFF Optional Status Register description for details. The SFF Mode Bit, when set to 1, does not change the direction of PIO0 and PIO1 to input.
BUSY: EEPROM Busy Indicator	b5	If this bit reads 1, an EEPROM write cycle (A0h or A2h Device Address) is in progress. (SMBus mode only; reads 0 in I²C bus mode)
CM: Communication Mode	b6	Selects mode for the serial communication interface. 0: I²C bus mode (power-on default) 1: SMBus mode
ADMD: PIO Address Mode	b7	Selects Address Mode for PIO Read/Write access. See PIO Read/Write Access Registers for details. 0: Multi-Address Mode (power-on default) 1: Single-Address Mode

PIO Read-Inversion and Output Type (Device Address = A0h)

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
7Bh	OT3	OT2	OT1	OT0	IMSK3	IMSK2	IMSK1	IMSK0

There is general read and write access to this address. The power-on default is copied from memory address 77h (Device Address = A0h).

BIT DESCRIPTION	BIT(S)	DEFINITION
IMSK0: Read-inversion control of PIO0	b0	0 ⇒ no inversion, 1 ⇒ data read from PIO0 is inverted
IMSK1: Read-inversion control of PIO1	b1	0 ⇒ no inversion, 1 ⇒ data read from PIO1 is inverted
IMSK2: Read-inversion control of PIO2	b2	0 ⇒ no inversion, 1 ⇒ data read from PIO2 is inverted
IMSK3: Read-inversion control of PIO3	b3	0 ⇒ no inversion, 1 ⇒ data read from PIO3 is inverted
OT0: Output Type of PIO0	b4	0: ⇒ Push-Pull, 1 ⇒ Open Drain
OT1: Output Type of PIO1	b5	0: ⇒ Push-Pull, 1 ⇒ Open Drain
OT2: Output Type of PIO2	b6	0: ⇒ Push-Pull, 1 ⇒ Open Drain
OT3: Output Type of PIO3	b7	0: ⇒ Push-Pull, 1 ⇒ Open Drain

PIO Read/Write Access Registers (Device Address = A0h)

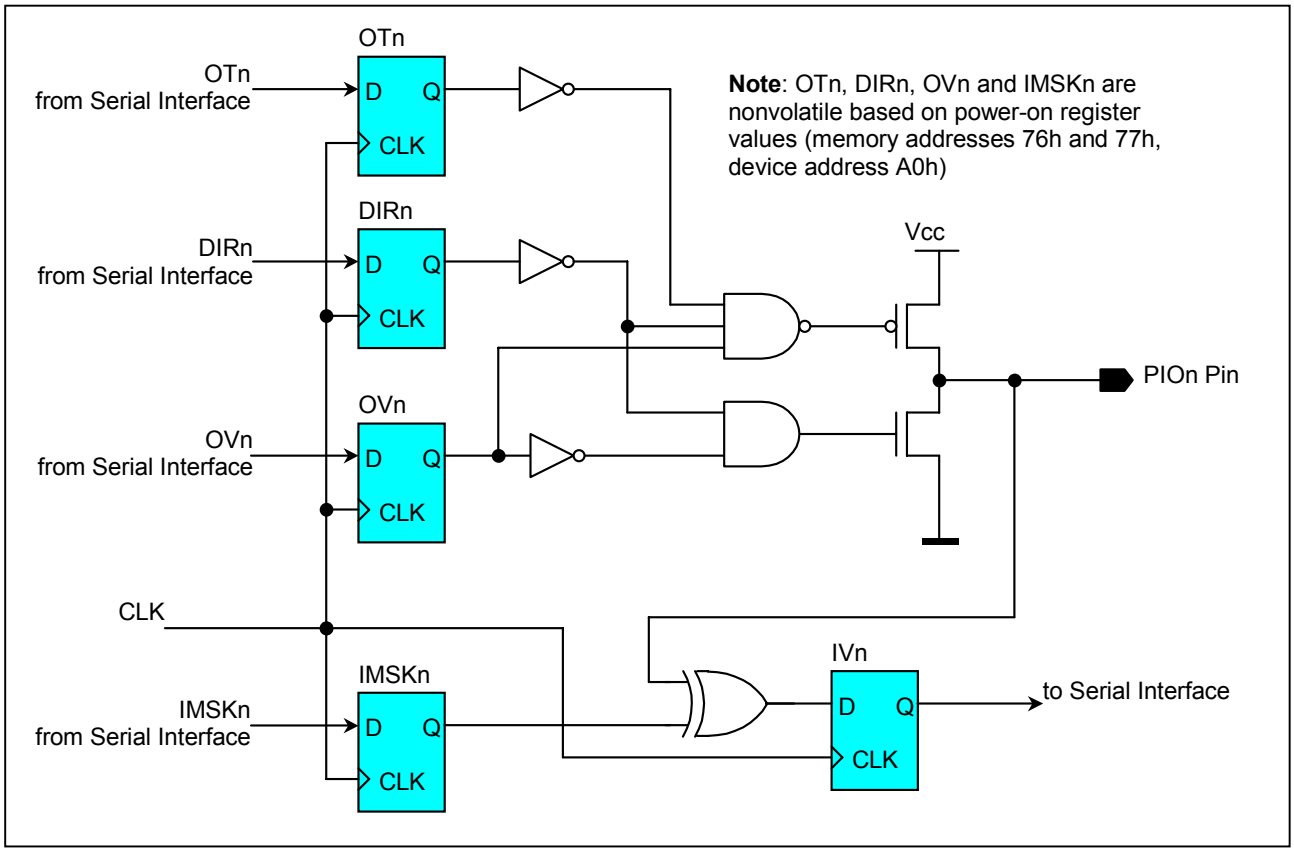
ADDR	b7	b6	b5	b4	b3	b2	b1	b0	PIO Address Mode
7Ch	IV3	IV2	IV1	IV0	OV3	OV2	OV1	OV0	Single
	1	1	1	IV0	1	1	1	OV0	Multi
7Dh	00h (no function)								Single
	1	1	1	IV1	1	1	1	OV1	Multi
7Eh	00h (no function)								Single
	1	1	1	IV2	1	1	1	OV2	Multi
7Fh	00h (no function)								Single
	1	1	1	IV3	1	1	1	OV3	Multi

There is general read and write access to these registers. Bits shown as 1 have no function; their state cannot be changed.

BIT DESCRIPTION	BIT(S)	DEFINITION
OV0: Output Value of PIO0	—	Logic output state of PIO0 if DIR0 = 0 (output)
OV1: Output Value of PIO1	—	Logic output state of PIO1 if DIR1 = 0 (output)
OV2: Output Value of PIO2	—	Logic output state of PIO2 if DIR2 = 0 (output)
OV3: Output Value of PIO3	—	Logic output state of PIO3 if DIR3 = 0 (output)
IV0: Input Value of PIO0	—	Logic state read from PIO0 XOR'ed with IMSK0
IV1: Input Value of PIO1	—	Logic state read from PIO1 XOR'ed with IMSK1
IV2: Input Value of PIO2	—	Logic state read from PIO2 XOR'ed with IMSK2
IV3: Input Value of PIO3	—	Logic state read from PIO3 XOR'ed with IMSK3

Figure 3 shows a simplified schematic of a PIO. The flip flops are accessed through the PIO R/W Access Registers and memory addresses 7Ah and 7Bh (Device Address = A0h). They are initialized at power-up or during reset according to the data stored at memory addresses 76h and 77h (Device Address = A0h). When a PIO is configured as input, the PIO output is tri-stated (high impedance). When a PIO is configured as output, the PIO input is the same as the output state XOR'ed with the corresponding read inversion bit.

Figure 3. PIO Simplified Schematic



SFF Optional Status Register (Device Address = A2h, only if SFF Mode is on)

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
6Eh	0	0	0	0	0	TXF	LOS	0

This register is read only. The functional assignments of the individual bits are explained in the table below. Bits 0 and 3 to 7 have no function; they always read 0 and cannot be set to 1.

BIT DESCRIPTION	BIT(S)	DEFINITION
LOS: Loss Of Signal	b1	Reports the logical state of PIO0; in SFF-8472 compatible modules, PIO0 is connected to the Loss Of Signal indicator
TXF: TX_FAULT	b2	Reports the logical state of PIO1; in SFF-8472 compatible modules, PIO1 is connected to the TX_FAULT indicator

DEVICE OPERATION

The typical use of the DS28CZ04 in an application involves writing to and reading from the memory and accessing the PIOs. All these activities are controlled through the I²C/SMBus serial interface. Since the DS28CZ04 has memory areas and registers of different characteristics there are several special cases to consider. See section *Read and Write* for details.

Serial Communication Interface

General Characteristics

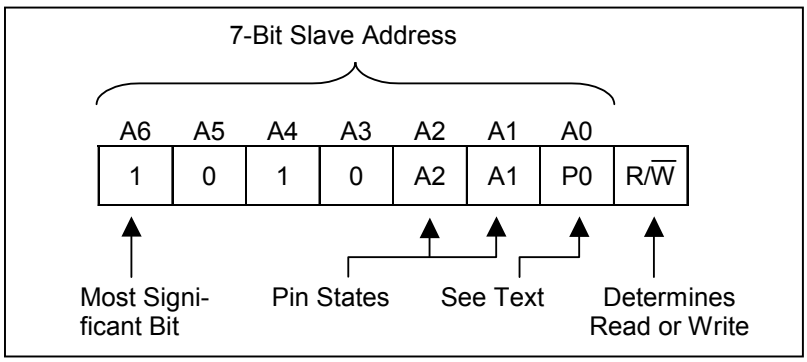
The serial interface uses a data line (SDA) plus a clock signal (SCL) for communication. Both SDA and SCL are bidirectional lines, connected to a positive supply voltage through a pullup resistor. When there is no communication, both lines are HIGH. The output stages of devices connected to the bus must have an open-drain or open-collector to perform the wired-AND function. Data can be transferred at rates of up to 100kbps in the Standard-mode, up to 400kbps in the Fast-mode. The DS28CZ04 works in both modes.

A device that sends data on the bus is defined as a transmitter, and a device receiving data as a receiver. The device that controls the communication is called a “master.” The devices that are controlled by the master are “slaves.” The DS28CZ04 is a slave device.

Slave Address/Direction Byte

To be individually accessed, each device must have a slave address that does not conflict with other devices on the bus. The slave address to which the DS28CZ04 responds is shown in Figure 4. The slave address is part of the slave-address/direction byte. The upper 4 bits of the slave address of the DS28CZ04 are set to 1010b. Bits A1 and A2 correspond to the A1 and A2 pins; to be selected the device must be addressed with A1 and A2 bits matching the logical state of the respective pins.

Figure 4. DS28CZ04 Slave Address



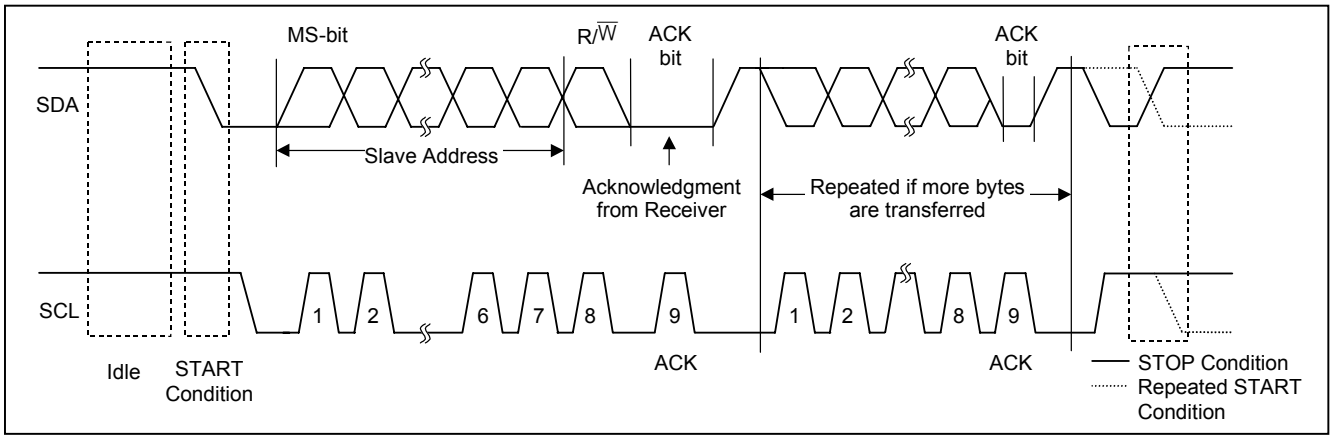
As a 512 byte memory device, the DS28CZ04 needs 9 address bits to access a memory location. The P0 bit transmitted in place of the A0 address bit specifies whether the “lower half” (0b) or the “upper half” (1b) of the memory is addressed. This causes the DS28CZ04 to occupy two logical slave addresses, one for each half of the memory. Throughout this document, the lower half of the memory is referenced as **Device Address A0h** and the upper half as **Device Address A2h**. The addresses A0h and A2h are correct if the A1 and A2 pins are tied to logic 0. For different conditions at these pins the slave address changes accordingly.

The last bit of the slave-address/direction byte (R/$\overline{W}$) defines the data direction. When set to a 0, subsequent data will flow from master to slave (write access mode); when set to a 1, data will flow from slave to master (read access mode). Although the P0 bit is also transmitted when accessing the DS28CZ04 in read mode, its value is ignored (don’t care); instead, the value transmitted in the most recent **write** access applies.

I²C/SMBus Protocol

Data transfers may be initiated only when the bus is not busy. The master generates the serial clock (SCL), controls the bus access, generates the START and STOP conditions, and determines the number of bytes transferred on the data line (SDA) between START and STOP. Data is transferred in bytes with the most significant bit being transmitted first. After each byte follows an acknowledge bit to allow synchronization between master and slave. During any data transfer, SDA must remain stable whenever the clock line is HIGH. Changes in SDA line while SCL is high will be interpreted as a START or a STOP. The protocol is illustrated in Figure 5. For detailed timing references see Figure 6.

Figure 5. I²C/SMBus Protocol Overview



Bus Idle or Not Busy
Both, SDA and SCL, are inactive, i. e., in their logic HIGH states.

START Condition
To initiate communication with a slave, the master has to generate a START condition. A START condition is defined as a change in state of SDA from HIGH to LOW while SCL remains HIGH.

STOP Condition
To end communication with a slave, the master has to generate a STOP condition. A STOP condition is defined as a change in state of SDA from LOW to HIGH while SCL remains HIGH.

Repeated START Condition
Repeated starts are commonly used for read accesses after having specified a memory address to read from in a preceding write access. The master can use a repeated START condition at the end of a data transfer to immediately initiate a new data transfer following the current one. A repeated START condition is generated the same way as a normal START condition, but without leaving the bus idle after a STOP condition.

Data Valid
With the exception of the START and STOP condition, transitions of SDA may occur only during the LOW state of SCL. The data on SDA must remain valid and unchanged during the entire high pulse of SCL plus the required setup and hold time ($t_{HD:DAT}$ after the falling edge of SCL and $t_{SU:DAT}$ before the rising edge of SCL, see Figure 6). There is one clock pulse per bit of data. Data is shifted into the receiving device during the rising edge of the SCL pulse.

When finished with writing, the master must release the SDA line for a sufficient amount of setup time (minimum $t_{SU:DAT}$ + t_R in Figure 6) before the next rising edge of SCL to start reading. The slave shifts out each data bit on SDA at the falling edge of the previous SCL pulse and the data bit is valid at the rising edge of the current SCL pulse. The master generates all SCL clock pulses, including those needed to read from a slave.

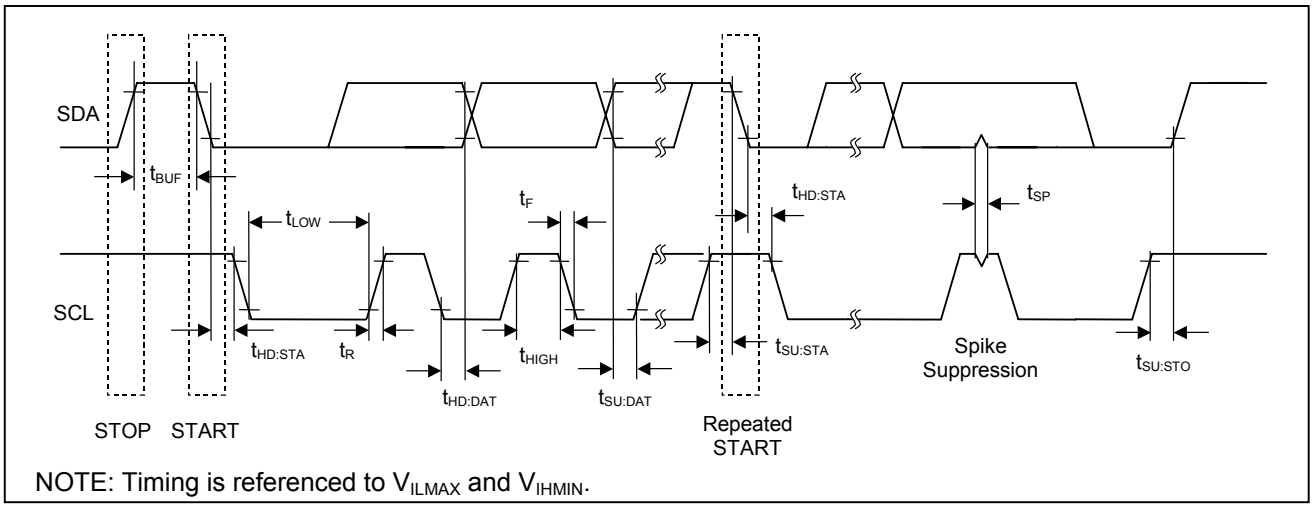
Acknowledged by Slave
Usually, a slave device, when addressed, is obliged to generate an acknowledge after the receipt of each byte. The master must generate a clock pulse that is associated with this acknowledge bit. A device that acknowledges must pull SDA LOW during the acknowledge clock pulse in such a way that SDA is stable LOW during the HIGH period of the acknowledge-related clock pulse plus the required setup and hold time ($t_{HD:DAT}$ after the falling edge of SCL and $t_{SU:DAT}$ before the rising edge of SCL).

Acknowledged by Master
To continue reading from a slave, the master is obliged to generate an acknowledge after the receipt of each byte. The master must generate the clock pulse for each acknowledge bit and, during the acknowledge clock pulse, pull SDA LOW in such a way that SDA is stable LOW during the HIGH period of the acknowledge-related clock pulse. The setup and hold time ($t_{HD:DAT}$ after the falling edge of SCL and $t_{SU:DAT}$ before the rising edge of SCL) also apply to the master.

Not Acknowledged by Slave
A slave device may be unable to receive or transmit data, e.g., because it is busy. In SMBus mode, the DS28CZ04 will always acknowledge its slave address. However, some time later the device may refuse to accept data, e.g., because of an invalid access mode or an EEPROM write cycle in progress. In this case the DS28CZ04 will not acknowledge any of the bytes that it refuses and will leave SDA HIGH during the HIGH period of the acknowledge-related clock pulse. See section *Read and Write* for a detailed list of situations where the DS28CZ04 does not acknowledge.

Not Acknowledged by Master
At some time when receiving data, the master must signal an end of data to the slave device. To achieve this, the master does not acknowledge the last byte that it has received from the slave. In response, the slave releases SDA, allowing the master to generate the STOP condition.

Figure 6. I²C/SMBus Timing Diagram



NOTE: Timing is referenced to V_{ILMAX} and V_{IHMIN}.

Read and Write

From the master's point of view, the DS28CZ04 behaves like an memory device with an address range of 512 bytes. As indicated in the Memory Map, Figure 2, the DS28CZ04 has different types of memory: SRAM, EEPROM and read-only areas. The write behavior depends on the memory type and the characteristics of the location that is addressed. The SRAM registers can be written from 1 byte to multiple bytes at a time. The EEPROM can be written from 1 byte to 16 or 8 bytes at a time, depending on the memory location.

To write to the DS28CZ04, the master must address the device in write access mode, i.e., the slave address must be sent with the direction bit set to 0. The slave address also determines which of the memory halves is accessed. The next byte sent in write access mode is the address of the memory location to be written to ("write pointer") or to start reading from ("read pointer") if the write access is terminated without sending data ("dummy write"). Additional bytes are taken as data for the addressed memory location.

To read from the DS28CZ04, the master must address the device in read access mode, i.e., the slave address must be sent with the direction bit set to 1. The read pointer determines the location from which the master starts reading. To set the pointer, the DS28CZ04 must be addressed in write access mode, as described above.

Write Access

Due to the different memory types, special function registers, PIO access registers and address modes, there are several cases to be distinguished:

- Normal EEPROM — EEPROM block of 16 bytes
- Short EEPROM — EEPROM block of 8 bytes
- Special EEPROM — EEPROM block of 16 bytes with one or more non-writeable bytes
- Reserved — Block of 16 non-writeable bytes
- SRAM Write — SRAM bytes including PIO Read/Write Access Registers
- PIO direct — PIO Read/Write Access Registers only

Table 1A maps the various cases to the applicable memory addresses and explains the device behavior in detail. All EEPROM writes depend on the state of WP pin. Only when the EEPROM is not write-protected (WP pin state = 0) is data accepted and transferred to the EEPROM. When writing to PIO Read/Write Registers, either by running into their address range or by addressing them directly, one needs to further distinguish between PIO Multi-Address Mode and PIO Single Address Mode. The address mode is selected through the ADMD bit of the *Direction and Control/Status Register (Device Address A0h)* at address 7Ah. In Multi-Address Mode, each PIO occupies one memory address whereas in Single-Address Mode all PIOs share a single address. See the PIO Read/Write Access Registers description for details. The PIO address mode does not affect the device behavior when writing to the EEPROM sections.

Writing to EEPROM Locations

If the DS28CZ04 is addressed in write access mode, any data bytes that follow the address are written to a 16-byte buffer, beginning at an offset that is determined by the 4 least significant bits of the target address. This buffer is initialized (pre-loaded) with data from the addressed 16-byte EEPROM block. Incoming data replaces pre-loaded data. With every byte received, the buffer's write pointer as well as the read pointer is incremented. If the buffer's write pointer has reached its maximum value of 1111b (normal EEPROM and special EEPROM) or 0111b (short EEPROM) and additional data is received, the pointer wraps around (rolls over) and the incoming data is written to the beginning of the EEPROM write buffer and continuing. The same wrap-around applies to the 4 least-significant bits of the read pointer. This way the read pointer maintains the last address accessed during a write operation, incremented by one. The transfer from the buffer to the EEPROM begins when the master generates a STOP condition. Until the write cycle is completed, the DS28CZ04 is **busy** for the duration of t_{PROG}.

Table 1A. Write Access

WRITING WHILE DEVICE IS NOT BUSY		
PIO Mode	**Starting Address**	**SMBus or I²C Bus Mode**
Multi-Address	Device address = **A0h,** any 16-byte block **except 70h to 7Fh**; Device address = **A2h,** any 16-byte block **except 60h to 6Fh, F0h to FFh** (normal EEPROM)	**If WP pin is tied to GND**: Slave address is acknowledged; memory address is acknowledged, data is acknowledged; **write** pointer increments and wraps around from end of block to beginning of block, **read** pointer = write pointer +1. **If WP pin is tied to V_{CC}**: data is **not** acknowledged, no EEPROM write cycle takes place; everything else remains the same.
	Device address = **A0h,** memory address from **70h to 77h** (short EEPROM)	Same as “normal EEPROM” **except** that **write** pointer wraps around from **77h to 70h**.
	Device address = **A2h,** memory address from **60h to 6Fh** (special EEPROM)	**SFF** mode **off**: same as “normal EEPROM”. **SFF** mode **on**: data for address **6Eh** is **not** acknowledged; everything else is the same as with “normal EEPROM”.
	Device address = **A2h,** memory address from **F0h to FFh** (reserved)	Same as “normal EEPROM” **except** that data is **not** acknowledged.
	Device address = **A0h,** memory address from **78h to 7Bh** (SRAM write)	Slave address is acknowledged; memory address is acknowledged, data for address **78h and 79h** is **not** acknowledged; **write** pointer increments and wraps around from **7Fh to 7Ah**, **read** pointer = write pointer +1.
	Device address = **A0h,** memory address from **7Ch to 7Fh** (PIO direct)	Slave address is acknowledged; memory address is acknowledged, data is acknowledged; **write** pointer increments and wraps around from **7Fh to 7Ch**, **read** pointer = write pointer +1.
Single-Address	Device address = **A0h**, memory address from **78h to 7Fh excluding 7Ch** (SRAM write)	Slave address is acknowledged; memory address is acknowledged, data for addresses **7Dh to 7Fh and 78h to 79h** is **not** acknowledged; **write** pointer increments and wraps around from **7Fh to 7Ah**, **read** pointer = write pointer +1.
	Device address = **A0h**, memory address = **7Ch** (PIO direct)	Slave address is acknowledged; memory address is acknowledged, data is acknowledged; **write** pointer **stays at 7Ch**; **read** pointer **stays at 7Ch**.
	All other cases	Same as in PIO Multi-Address Mode.

Busy Polling
While busy, the behavior of the DS28CZ04 depends on the communication mode, which is selected through the CM bit of the *Direction and Control/Status Register (Device Address A0h)* at address 7Ah. Tables 1B and 2B show details. The PIO address mode does not affect the device behavior when busy.

In **I²C bus mode**, when busy the DS28CZ04 does not acknowledge its slave address until the write cycle is completed. The master can access the device by transmitting the Slave Address/Direction Byte and testing whether the address is acknowledged. As soon as the DS28CZ04 acknowledges, the master knows that the device is ready for further activities.

In **SMBus mode**, the DS28CZ04 always acknowledges its slave address. The only way for the master to detect the completion of the write cycle is through the BUSY bit in the *Direction and Control/Status Register (Device Address A0h)* at. To get to this bit the master must first address the DS28CZ04 in write access mode, Device Address A0h, and set the memory address to 7Ah (see Table 1B). Now the master can address the DS28CZ04 in read access

mode and generate pulses on SCL to read data, one byte after another without issuing a STOP (see Table 2B). Eventually the BUSY bit changes from 1 to 0 indicating the end of the write cycle. The BUSY bit is sampled during the transmission of the byte before it is read out; consequently, **the state read out reflects the state at sample time and not the actual state.** To get the actual state of the busy bit the master can a) read at the maximum data rate, b) read two bytes in sequence without delay in between and use the BUSY bit in the second byte or c) in a loop: read one byte, issue a STOP, wait, reposition the read pointer, address the DS28CZ04 in read mode to get another status byte.

Table 1B. Prepare For Busy Polling

WRITING WHILE DEVICE IS BUSY			
PIO Mode	**Starting Address**	**SMBus Mode**	**I²C Bus Mode**
Either Address Mode	Device Address = **A0h,** memory address = **7Ah**	Slave address is acknowledged; memory address is acknowledged; data is not acknowledged; **write** pointer keeps its last position; **read** pointer = **7Ah**.	Slave address is NOT acknowledged; memory address is not acknowledged; data is not acknowledged; **write** pointer keeps its last position; **read** pointer = write pointer +1.
	Device Address = **A0h,** any memory address **except 7Ah**	Slave address is acknowledged; memory address is not acknowledged; data is not acknowledged; **write** pointer keeps its last position; **read** pointer = write pointer +1.	
	Device Address = **A2h,** any memory address		

Table 2B. Busy Polling

READING WHILE DEVICE IS BUSY			
PIO Mode	**Read Pointer**	**SMBus Mode**	**I²C Bus Mode**
Either Address Mode	Device Address = **A0h**, memory address = **7Ah**	Slave address is acknowledged; data is delivered; **read** pointer **stays at 7Ah**.	Slave address is NOT acknowledged; no data is delivered; **read** pointer stays as is.
	Device Address = **A0h**, **excluding** memory address **7Ah**	Slave address is acknowledged; no data is delivered; **read** pointer = last write pointer +1.	
	Device Address = **A2h**, any memory address		

Writing to SRAM and PIO Locations

If the DS28CZ04 is addressed in write access mode, any data bytes that follow the address are directly written to their respective memory location. The PIO address mode controls the device behavior when writing to the PIO Read/Write access registers. Depending on whether one runs into the PIO address range (SRAM write) or whether one starts at a PIO address (PIO direct) the pointer and data acknowledge behavior is different. Table 1A shows the details. The PIO Address Mode is another parameter that affects the pointer behavior. Figure 7 illustrates the possible cases and the sequence in which the addresses are accessed.

The common characteristic in both SRAM write cases is a starting address in the SRAM block (address range 78h to 7Fh) **excluding any address used for PIO access**. Data for writeable registers (7Ah, 7Bh and valid addresses for PIO Read/Write access) is acknowledged; the write pointer increments and after address 7Fh rolls over to 7Ah.

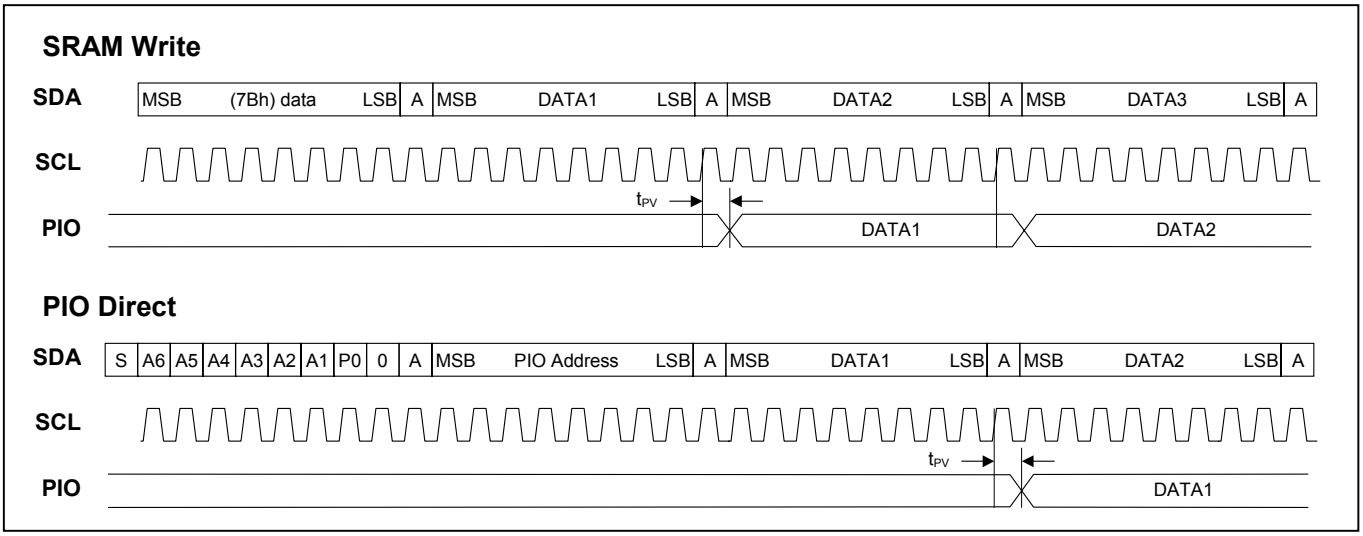
The common characteristic in both PIO direct cases is a starting address **within the address range used for PIO access**. In PIO Multi-Address Mode, there are four such addresses (7Ch to 7Fh); each PIO occupies its own address. Data is always acknowledged; the write pointer increments to the next PIO and eventually wraps around to 7Ch. In PIO Single-Address Mode, there is exactly one address (7Ch) that is shared by all PIOs. Data is always acknowledged; the write pointer stays at 7Ch.

Figure 7. SRAM and PIO Writing

	Memory Location		PIO Multi-Address Mode		PIO Single-Address Mode	
	Address	Function	SRAM Write	PIO Direct	SRAM Write	PIO Direct
Lower Half	00h to 77h	Memory				
	78h	Reserved				
	79h	Reserved				
	7Ah	Register				
	7Bh	Register				
	7Ch	PIO R/W				
	7Dh	PIO R/W				
	7Eh	PIO R/W				
	7Fh	PIO R/W				
	80h to FFh	Memory				
Upper Half	00h to FFh	Memory				

When writing to a PIO, as shown in Figure 8, any state change is triggered by the SCL pulse that the master generates for the acknowledge bit of byte written to the PIO Read/Write Access Register. After the output transition time t_{PV} is expired, the state change is completed. In PIO Single-Address mode all PIOs change their state approximately at the same time; in this mode the fastest rate for a PIO to change its state is $f_{SCL}/9$. In PIO Multi-Address Mode each PIO is accessed individually; in this mode when writing in an endless loop the fastest rate for a PIO to change its state is $f_{SCL}/36$. Transfer of data can be stopped at any moment by a STOP condition. When this occurs, data present at the last acknowledged phase is valid.

Figure 8. PIO Write Access Timing



Reading Memory and PIOs

If the DS28CZ04 is addressed in read access mode, the read pointer determines the location from which the master will start reading. The read pointer is set when the DS28CZ04 is accessed in write access mode, either for

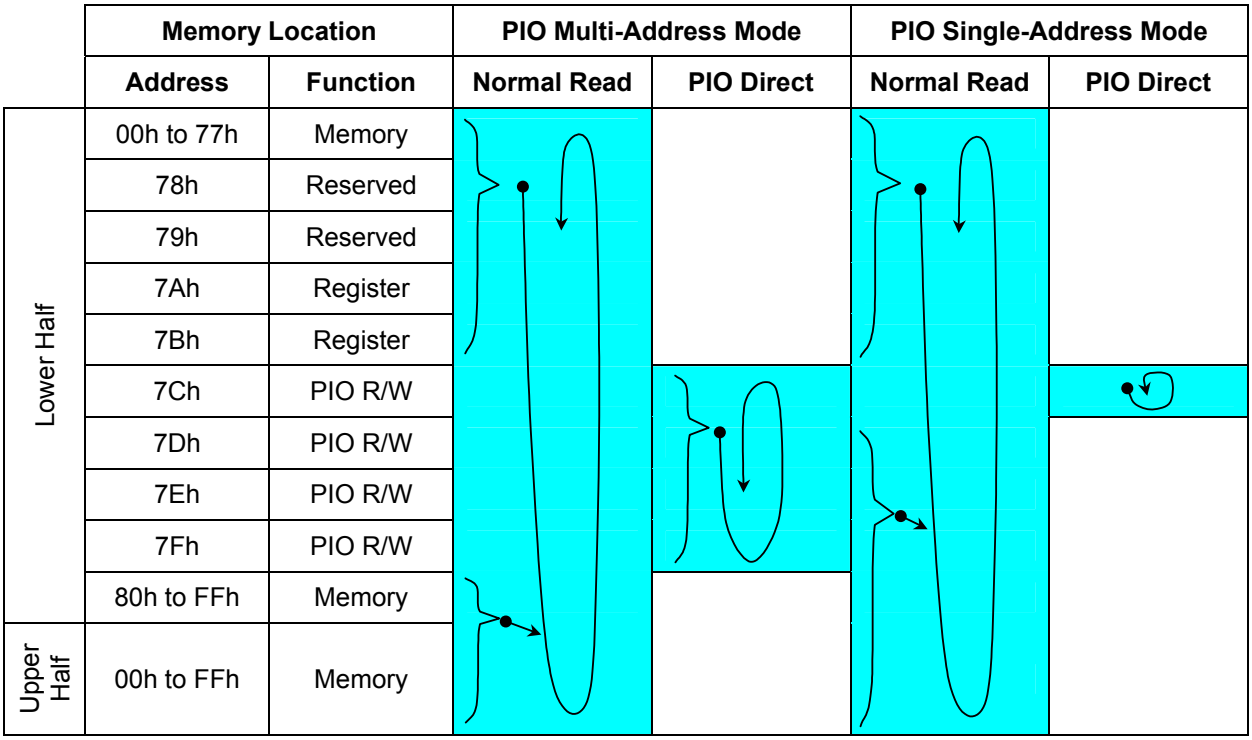
writing data or through a dummy write. At power-on the read pointer is reset to address 00h of the lower half of the memory. A description on how the read pointer is affected during write accesses is included in Table 1A. In contrast to write accesses where the memory is updated in small blocks of 8 or 16 bytes, all 512 bytes are readable in a single read access. Only two cases need to be distinguished: normal read and PIO direct. Table 2A explains the cases in detail.

Table 2A. Read Access

READING WHILE DEVICE IS NOT BUSY		
PIO Mode	**Read Pointer**	**SMBus or I²C Bus Mode**
Multi-Address	Anywhere **excluding** device address = **A0h**, memory address from **7Ch to 7Fh** (normal read)	Slave address is acknowledged; data is delivered; **read** pointer increments, eventually crossing from lower half to upper half of the memory, **and wraps around from upper half FFh to lower half 00h**.
	Device address = **A0h**, memory address from **7Ch to 7Fh** (PIO direct)	Slave address is acknowledged; data is delivered; **read** pointer increments **and wraps around from 7Fh to 7Ch, staying in the lower half of memory**.
Single-Address	Anywhere **excluding** device address = **A0h**, memory address = **7Ch** (normal read)	Slave address is acknowledged; data is delivered; **read** pointer increments, eventually crossing from lower half to upper half of the memory, **and wraps around from upper half FFh to lower half 00h**.
	Device address = **A0h**, memory address = **7Ch** (PIO direct)	Slave address is acknowledged; data is delivered; **read** pointer **stays at 7Ch**.

The PIO Address Mode in conjunction with the initial read pointer position determines the sequence in which the addresses are accessed. Figure 9 illustrates the possible cases.

Figure 9. Memory and PIO Reading

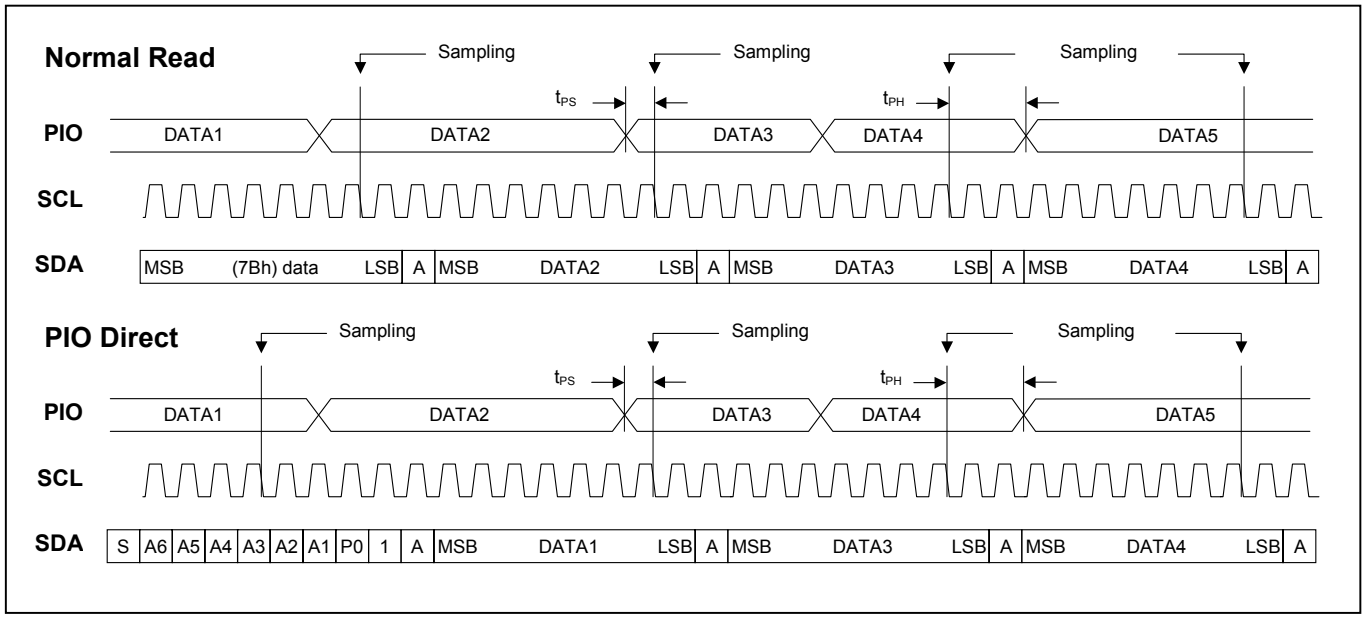


The common characteristic in both normal read cases is a starting address anywhere in the memory **excluding any address used for PIO access**. The read pointer increments after every byte read. This way a series of read accesses reveals memory data of consecutive addresses, without any duplications or gaps. When reading from reserved areas the master receives FFh bytes. When the end of the upper half of the memory is reached (device address A2h, address FFh) the read pointer wraps around to the start of the lower half of the memory (device address A0h, address 00h). When the end of the lower half of the memory is reached, the read pointer continues at the start of the upper half of the memory. To change the read address, the master has to address the DS28CZ04 in write access mode and specify a new memory address.

The common characteristic in both PIO direct cases is a starting address **within the address range used for PIO access**. In PIO Multi-Address Mode, there are four such addresses (7Ch to 7Fh); each PIO occupies its own address. After a byte is sent to the master, the read pointer increments to the next PIO and eventually wraps around to 7Ch. In PIO Single-Address Mode, there is exactly one address (7Ch) that is shared by all PIOs. Consequently, the master can continue reading, but the read pointer stays at 7Ch.

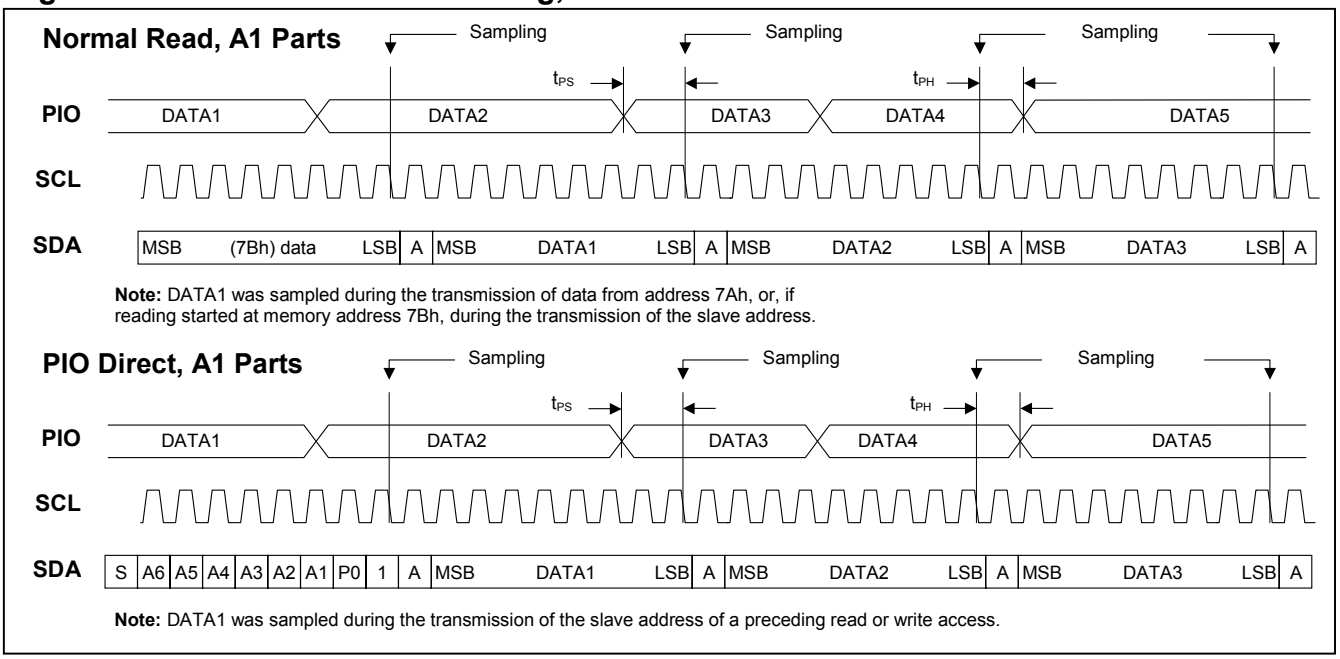
When reading from a PIO, as shown in Figure 10, the sampling takes place on the falling SCL edge of the 2nd-last bit before the acknowledge bit. With PIO direct mode, the first sample is taken 3 SCL cycles earlier, i. e., during the transmission of the A3 bit of the slave address. To be correctly assessed, the PIO state must not changed during the t_{PS} and t_{PH} interval. In PIO Single-Address mode all PIOs are sampled simultaneously; in this mode with PIO direct access the fastest sample rate for a PIO is $f_{SCL}/9$. In PIO Multi-Address Mode each PIO is sampled individually; in this mode with PIO direct access the fastest sample rate for a PIO is $f_{SCL}/36$. Transfer of data can be stopped at any moment by a STOP condition. When this occurs, data from the last sampling instance is lost.

Figure 10. PIO Read Access Timing



With revision A1 devices, the sampling always takes place on the falling SCL edge of the last bit before the acknowledge bit. The sampled data, however, is reported to the master one byte late, as shown in Figure 10A. The first sample of PIO data that the master receives in PIO direct access should be discarded since its timing relative to the transmission of the slave address is undefined. Any application firmware developed for revision A1 devices is fully compatible to newer devices.

Figure 10A. PIO Read Access Timing, A1 devices



I²C/SMBus Communication—Legend

SYMBOL	DESCRIPTION
S	START Condition
ADL,0	Select for Write Access to lower half
ADH,0	Select for Write Access to upper half
ADX,1	Select for Read Access
ADX,0	Select for Write access
A	Acknowledged

SYMBOL	DESCRIPTION
xx0xx1xxb	Byte that defines specific bits only
P	STOP Condition
A\	Not Acknowledged
<byte>	Transfer of 1 Byte
AMA	Any 8-bit Memory Address
Sr	Repeated START Condition

Command-Specific Communication—Color-Codes

Master-to-Slave	Slave-to-Master	Programming

Communication Examples

Set I²C mode, write 3 bytes starting at address 25h, lower half of the memory, test for end of cycle

S	ADL,0	A	7Ah	A	x0xxxxxxb	A	P

Set I²C bus mode; optional step; I²C bus mode is the power-on default.

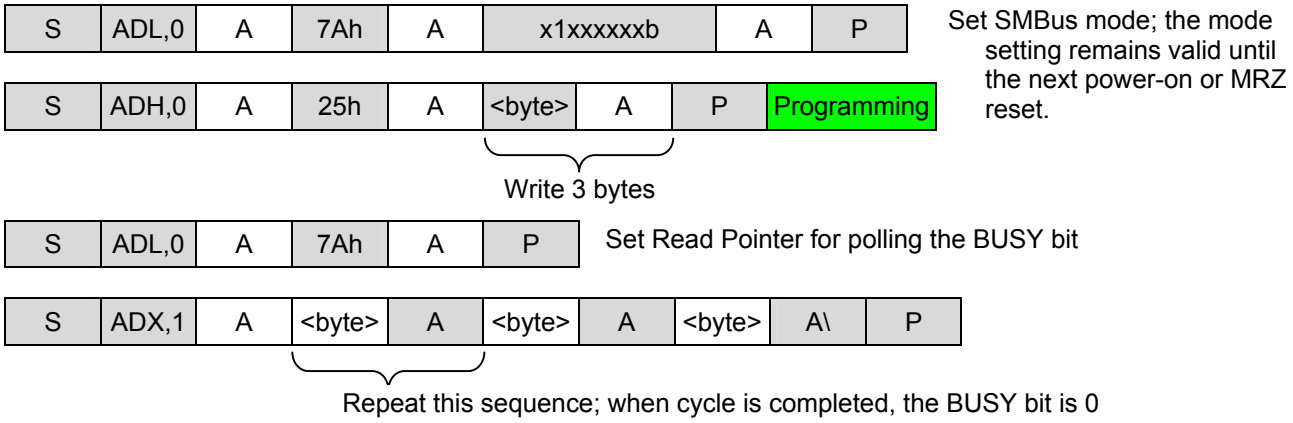
S	ADL,0	A	25h	A	<byte>	A	P	Programming

Write 3 bytes

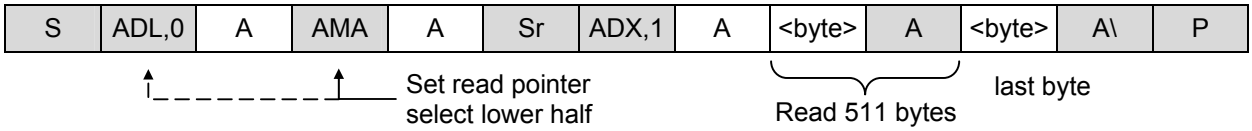
S	ADX,0	A\	Sr	ADX,0	A\	Sr	ADX,0	A	P

Repeat this sequence; when cycle is completed, the DS28CZ04 will acknowledge.

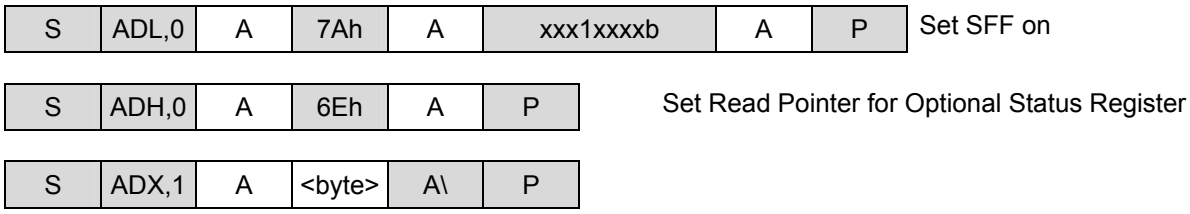
Set SMBus mode, write 3 bytes starting at address 25h, upper half of the memory, test for end of cycle



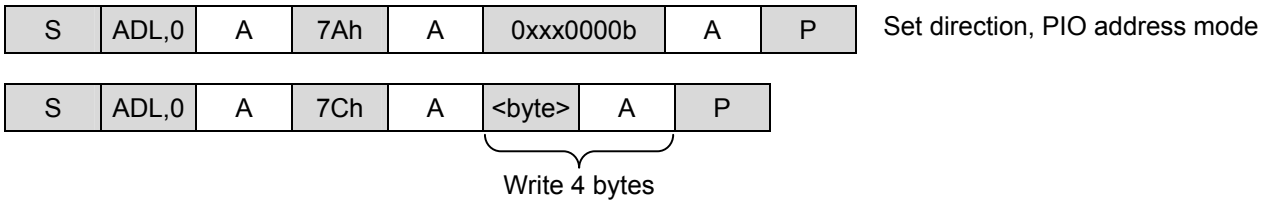
Read all memory, starting at the lower half of memory



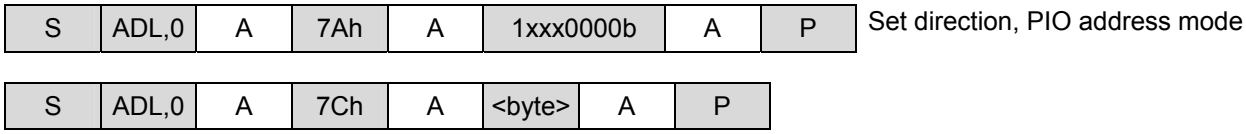
Set SFF Mode on, read SFF Optional Status Register



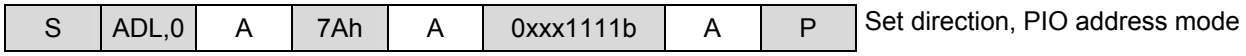
Write to all four PIOs in Multi-Address Mode, starting at PIO0



Write to all four PIOs in Single-Address Mode



Read from all four PIOs in Multi-Address Mode, starting at PIO1



S	ADL,0	A	7Dh	A	P

Set Read Pointer for PIO Access Register

S	ADX,1	A	<byte>	A	<byte>	A\	P

Read 3 bytes

Read from all four PIOs in Single-Address Mode

S	ADL,0	A	7Ah	A	1xxx1111b	A	P

Set direction, PIO address mode

S	ADL,0	A	7Ch	A	P

Set Read Pointer for PIO Access Register

S	ADX,1	A	<byte>	A\	P

Application Information

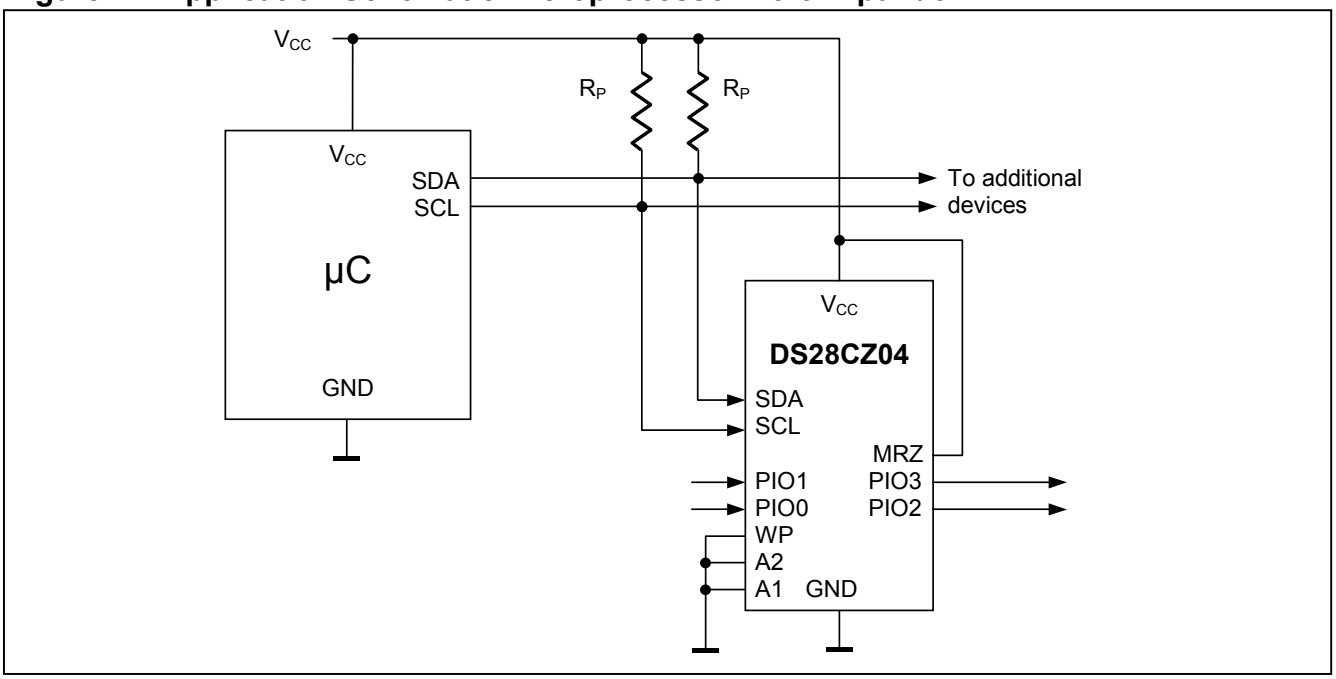
SDA and SCL Pullup Resistors

SDA is an open-drain output on the DS28CZ04 that requires a pullup resistor (Figure 11) to realize high logic levels. Because the DS28CZ04 uses SCL only as input (no clock stretching) the master can drive SCL either through an open-drain/collector output with a pullup resistor or a push-pull output.

Pullup Resistor R_P Sizing

According to the I²C specification, a slave device must be able to sink at least 3mA at a V_{OL} of 0.4V. The SMBus specification requires a current sink capability of 4mA at 0.4V. The DS28CZ04 can sink at least 4mA at 0.4V V_{OL} over its entire operating voltage range. This DC characteristic determines the minimum value of the pullup resistor: **$R_{PMIN} = (V_{CC} - 0.4V)/4mA$.** With a maximum operating voltage of 5.25V, the minimum value for the pullup resistor is 1.2kΩ. The "Minimum R_P" line in Figure 12 shows how the minimum pullup resistor changes with the operating (pullup) voltage.

Figure 11. Application Schematic Microprocessor Port Expander

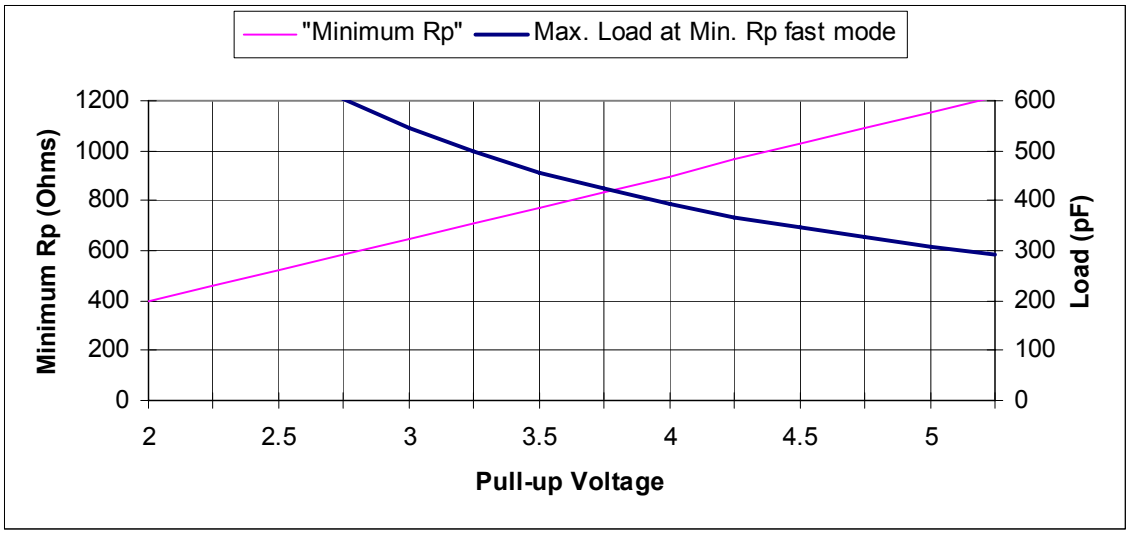


For I²C systems, the rise time and fall time are measured from 30% to 70% of the pullup voltage. The maximum bus capacitance C_B is 400pF. The maximum rise time must not exceed 300ns. Assuming maximum rise time, the maximum resistor value at any given capacitance C_B is calculated as: $\mathbf{R_{PMAX} = 300ns/(C_B \cdot ln(7/3))}$. For a bus capacitance of 400pF the maximum pullup resistor would be 885Ω.

Since an 885Ω pullup resistor, as would be required to meet the rise time specification and 400pF bus capacitance, is lower than R_{PMIN} at 5.25V, a different approach is necessary. The "Max. Load…" line in Figure 12 is generated by first calculating the minimum pullup resistor at any given operating voltage ("Minimum R_P" line) and then calculating the respective bus capacitance that yields a rise time of 300ns.

Only for pullup voltages of 4V and lower can the maximum permissible bus capacitance of 400pF be maintained. A reduced bus capacitance of 300pF is acceptable for the entire operating voltage range. The corresponding pullup resistor value at the voltage is indicated by the "Minimum R_P" line.

Figure 12. I²C Fast Speed Pullup Resistor Selection Chart



PACKAGE INFORMATION

(The package drawing(s) in this data sheet may not reflect the most current specifications. For the latest package outline information, go to **www.maxim-ic.com/DallasPackInfo**.)