SIEMENS

Microcomputer Components

Standalone Full-CAN Controller

SAE 81C90/91

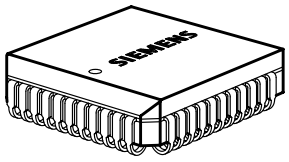
Data Sheet 01.97 Preliminary

SIEMENS

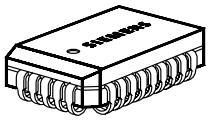
Stand Alone
Full CAN Controller

SAE 81C90/91

- Full CAN controller for data rate up to 1 Mbaud
- Complies with CAN specification V2.0 part A (part B passive)
- Up to 16 messages simultaneous (each with maximum data length)
- Message identifier reprogrammable “on the fly”
- Several transmit jobs can be sent with a single command
- Transmit check
- Basic CAN feature
- Time stamp for eight messages
- Two host interfaces (parallel and serial)
- User-configurable outputs for different bus concepts
- Programmable clock output
- Two 8 bit I/O-Port extension (P-LCC-44-1 package only)



P-LCC-44-1



P-LCC-28-1

The device comes in two versions:

SAE 81C90 in a P-LCC-44-1 package with two 8-bit I/O ports, and

SAE 81C91 in a P-LCC-28-1 package without I/O ports.

SAE 81C90/91 **Revision History:**	**Version 01.97**
Previous Releases:	06.95 05.94 (Copy version)

Page	**Subjects**
1056	Figure 1 corrected.
1057	Figure 2 corrected.
1059	Notes updated.
1065 - 1084	Register description and arrangement improved.
1066, 1069	New register maps.
1093	t_{AVLL}, t_{LLAX}, t_{DVWH} changed to 10 ns.
1093	t_{WHDX} changed to 5 ns.

Controller Area Network (CAN): License of Robert Bosch GmbH

Introduction

The Siemens Stand Alone Full CAN (SFCAN) circuit incorporates all the parts for completely autonomous transmission and reception of messages using the CAN protocol. The flexible, programmable interface allows hookup to different implementations of the physical layer. The link to a host controller can be made either by a multiplexed 8-bit address/data bus or by a high-speed, serial synchronous interface.

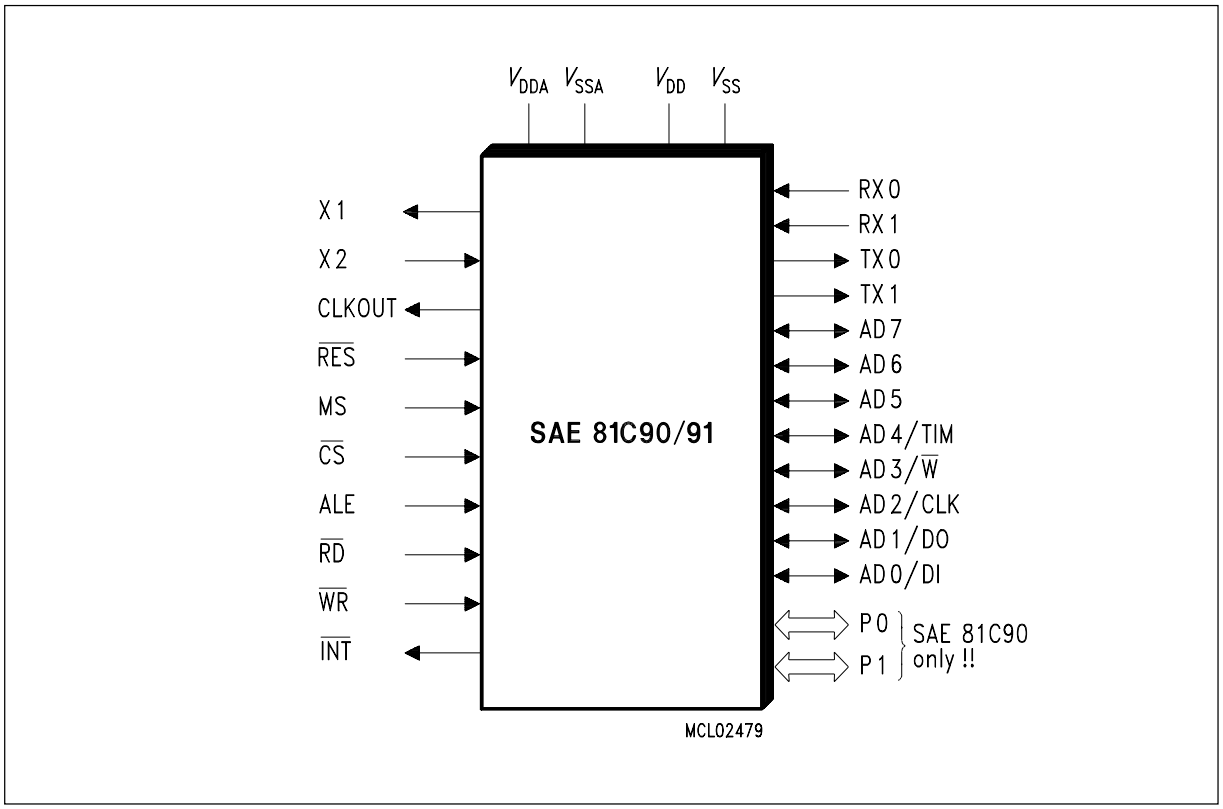


Figure 1
Logic Symbol

Ordering Information

Type	Ordering Code	Package	Function
SAE 81C91	Q67121-F0001	P-LCC-28-1	Stand Alone Full CAN Controller Temperature range – 40 to + 110 ˚C
SAE 81C90	Q67121-H9038	P-LCC-44-1	Stand Alone Full CAN Controller Temperature range – 40 to + 110 ˚C

Pin Configurations
(top view)

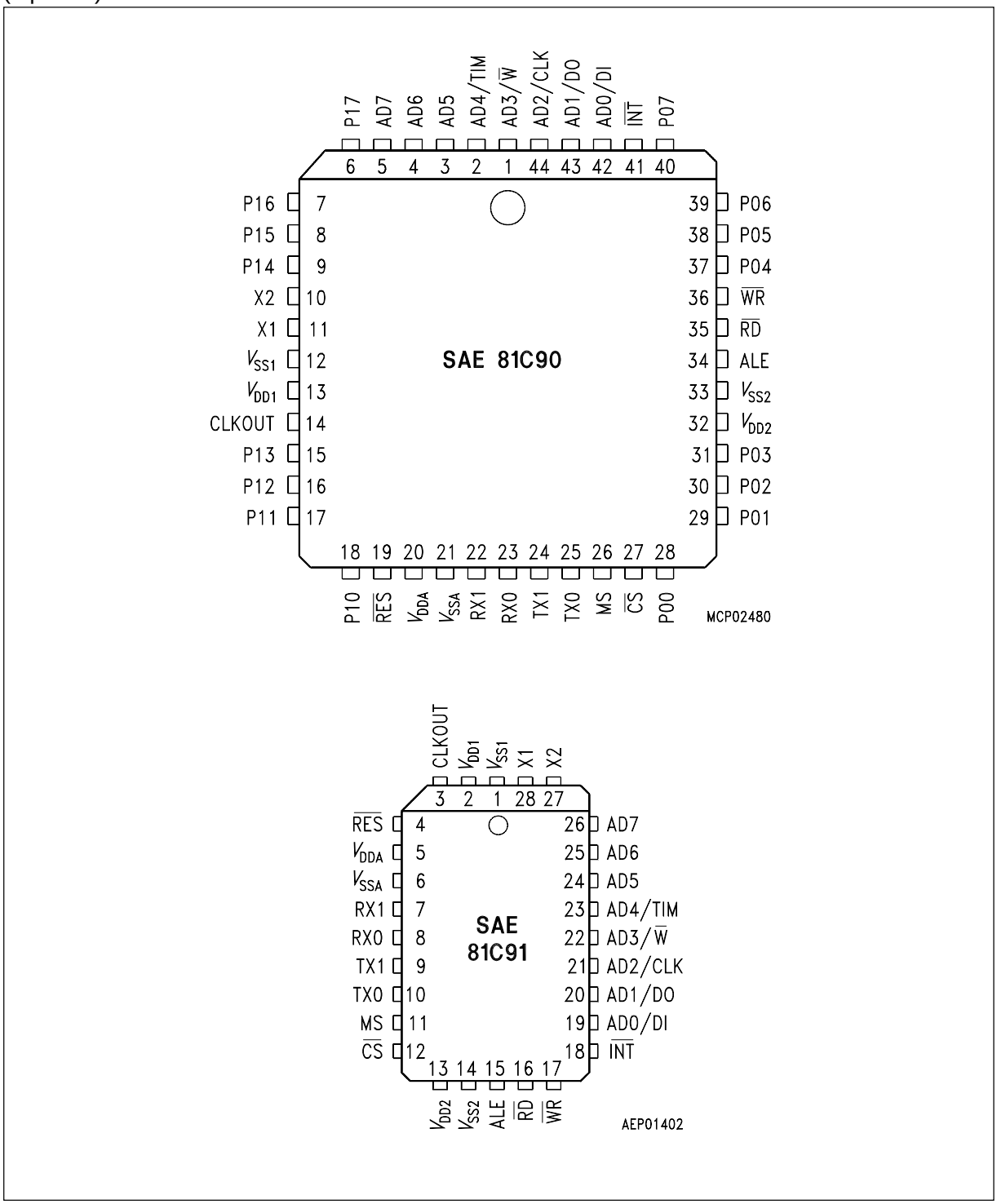


Figure 2

Pin Definitions and Functions

Symbol	Pin Number		Input (I) Output (O)	Function
	PLCC-48	PLCC-28		
X1 [1]	11	28	O	Crystal oscillator output. Must be unconnected for external clock input.
X2 [1]	10	27	I	Crystal oscillator input. Used for external clock input.
CLKOUT [1]	14	3	O	Clock output
$\overline{RES}$	19	4	I	Reset. (Schmitt trigger characteristic)
AD0/DI	42	19	I/O	PI: Address / Data bus / SI: Data input
AD1/DO	43	20	I/O	PI: Address / Data bus / SI: Data output
AD2/CLK	44	21	I/O	PI: Address / Data bus / SI: Clock input
AD3/$\overline{W}$	1	22	I/O	PI: Address / Data bus / SI: Write select
AD4/TIM	2	23	I/O	PI: Address / Data bus / SI: TIM = 0: Timing A; TIM = 1: Timing B
AD5	3	24	I/O	PI: Address/Data bus
AD6	4	25	I/O	PI: Address/Data bus
AD7	5	26	I/O	PI: Address/Data bus
$\overline{RD}$	35	16	I	PI: Read / SI: no Function
$\overline{WR}$	36	17	I	PI: Write / SI: no Function
ALE	34	15	I	PI: Address Latch Enable / SI:no Function
$\overline{CS}$	27	12	I	Chip Select
$\overline{INT}$	41	18	O	Interrupt
MS	26	11	I	Mode Select (PI $\leftrightarrow$ SI)
P00 … P03, P04 … P07	28, 29, 30, 31, 37, 38, 39, 40	– –	I/O I/O	Port 0 These pins provide internal pullup resistors of about 10...200 kΩ.
P10 … P13, P14 … P17	18, 17, 16, 15, 9, 8, 7, 6	– –	I/O I/O	Port 1 These pins provide internal pullup resistors of about 10...200 kΩ.
TX0	25	10	O	Transmitter output 0
TX1	24	9	O	Transmitter output 1
RX0	23	8	I	Comparator input 0 / Digital input [2]
RX1	22	7	I	Comparator input 1 [2]
V_{DDA}	20	5	I	Analog power supply for comparator (may be unconnected using the digital mode)

Pin Definitions and Functions (cont'd)

Symbol	Pin Number		Input (I) Output (O)	Function
	PLCC-48	PLCC-28		
V_{SSA}	21	6	I	Analog power ground for comparator (must always be connected)
V_{DD1}	13	2	I	Digital power supply 3)
V_{DD2}	32	13	I	Digital power supply
V_{SS1}	12	1	I	Digital power ground 3)
V_{SS2}	33	14	I	Digital power ground

1) For best results keep the crystal circuitry connections as short as possible and keep the CLKOUT line away from it.

2) If the bus lines work according to the ISO specification, additional circuitry is necessary for interconnection of the input comparator to the bus lines.
The digital mode is enabled by setting bit DI in register BL2. When using the digital mode pin RX1 should be on V_{SS}.

3) It is recommended to decouple these supply pins close to the device using a 10 pF capacitor in addition to the standard 100 nF capacitor.

Functional Description

The Siemens stand-alone Full-CAN (SFCAN) circuit is a large-scale-integrated peripheral device that executes the entire protocol of an automobile or industrial network.

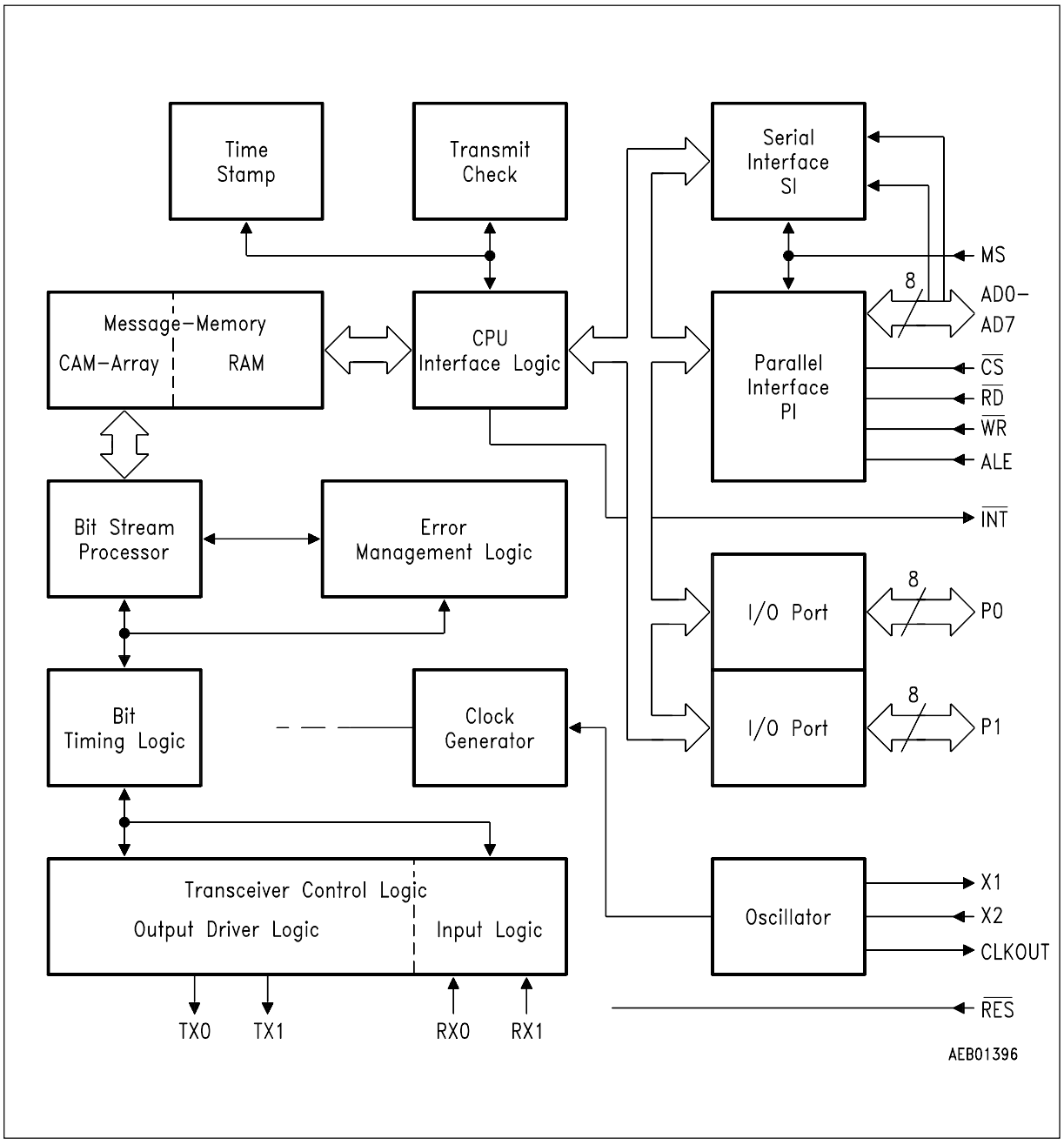


Figure 3
Block Diagram

Bus communication is based on the controller-area-network (CAN) protocol. With features like short message length, guaranteed reaction time for messages of appropriate priority, which is defined by the message identifiers. Also included are powerful error detection and treatment capabilities plus ease of operation. The CAN protocol is especially designed for the requirements of automobile and industrial electronic networks.

The SFCAN circuit incorporates all the parts for completely independent transmission and reception of messages using the CAN protocol. The flexible, programmable interface allows connection to different implementations of the physical layer. The link to a host controller can be made either by a multiplexed 8-bit address/data bus or by a high-speed, serial synchronous interface.

Message Memory

The SFCAN circuit filters incoming messages with an associative memory (CAM = content-addressable memory). For this the identifier and RTR bits of the required message must be written to the appropriate memory location.

The identifier of each incoming message is compared with the identifiers stored in the CAM. Upon a match the received data bytes are written into the RAM buffer of the matching message. At the same time the corresponding receive-ready bit is set and a receive interrupt is generated, if it is enabled. If no match is detected, the received message is rejected.

Identifiers can be reprogrammed at any time, although it is possible that data of the old or new identifier may be lost during reprogramming.

An incoming transmit request will only be satisfied automatically by the hardware if the RTR bit of the particular identifier is set in CAM.

To ensure data consistency when reading or writing several data bytes of a specific message the message objects are not accessed directly but via a 64-bit shadow register (see figure below).
This shadow register stores the complete data field of a certain message object for both reading and writing.

For read accesses the message's data field is copied to the shadow register...
...with the 1st read access to the respective data field (e.g. 80_H ... 87_H for message 0), or
...with any read access to byte 7 of the respective data field (e.g. 87_H for message 0).
This ensures that all bytes read via the shadow register belong to the same message, even though a new one might have been received in the meantime.

For write accesses the shadow register is copied to the respective message data field...
...with any write access to byte 0 of the respective data field (e.g. 80_H for message 0).
This ensures that only completely updated message are transmitted.

It is therefore recommended to begin all read and write accesses with the most-significant data byte of a message and end with data byte 0. This ensures operations on consistent data and correct transfers between the shadow register and the message RAM.

Note: For these reasons it is absolutely essential to ensure that the writing of data is not interrupted by a read operation and vice versa, a read operation should not be interrupted by a write.

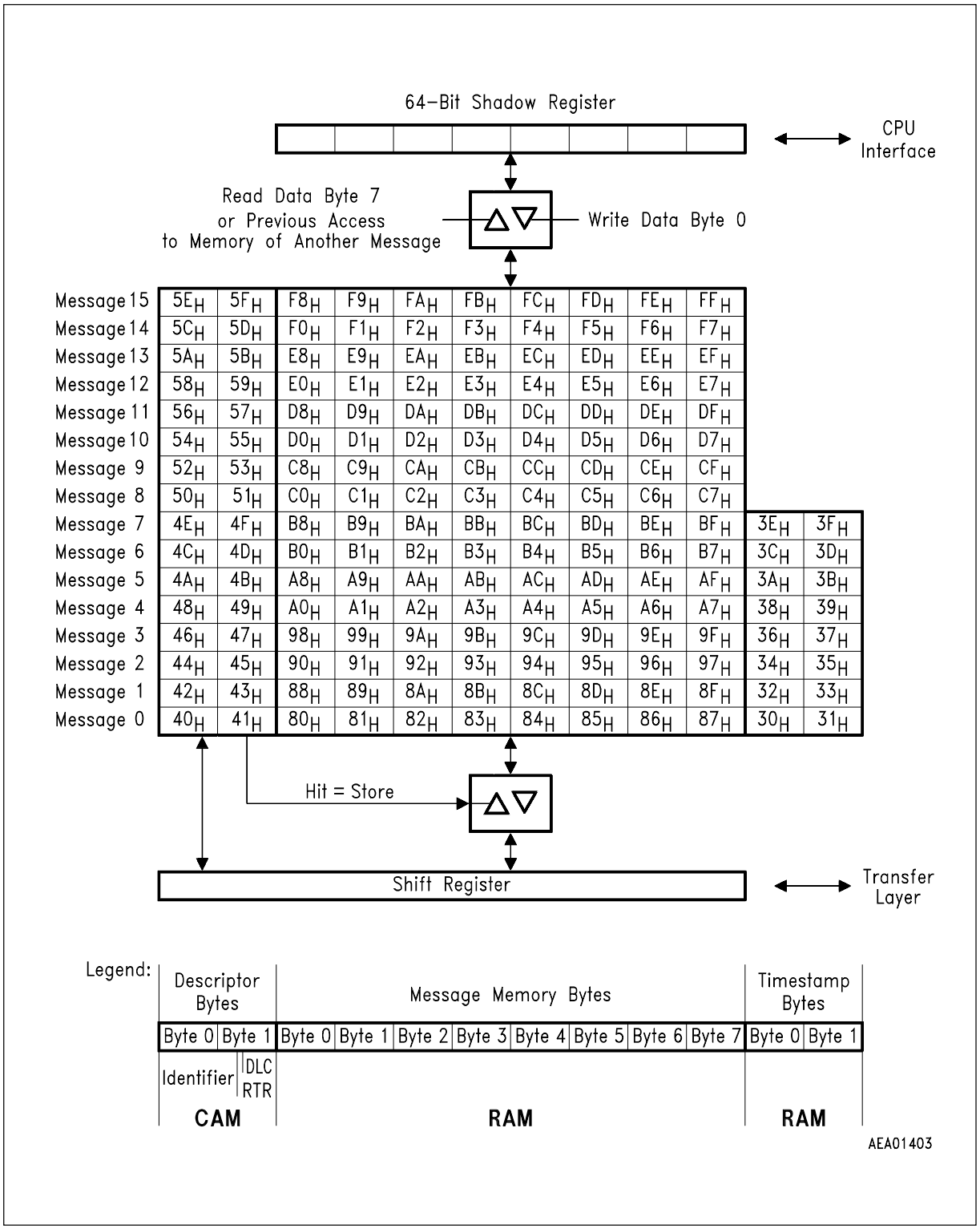


Figure 4
CAM, Message Memory and Time-Stamp Registers

Bit Stream Processor (BSP)

The bit-stream processor controls the entire protocol, differentiates between the frames types and detects frame errors.

Error Management Logic (EML)

The error-management logic receives error messages from the BSP and, in turn, sends back information about error state to the BSP and CIL.

Bit Timing Logic (BTL)

The bit-timing logic determines the timing of the bits and synchronizes with the edges of the bit stream on the CAN bus.

Transceiver Control Logic (TCL)

The transceiver-control logic consists of programmable output driver, input comparator and input multiplexer.

Clock Generator (CG)

The clock generator consists of an oscillator and a programmable divider. The oscillator can be fed from an external quartz crystal, ceramic resonator or an external timing source. The permissible crystal frequency is 1 to 20 MHz, and the external clock may be between 0 and 20 MHz. A programmable frequency, dependent on the crystal clock, is available with the CLKOUT pin, e.g. for the clocking of a host controller.

CPU Interface Logic (CIL)

The CPU interface logic controls the access of the host via the parallel or serial interface, interprets the commands and outputs status and interrupt information.

Transmit Check

The CAN protocol ensures a very high integrity for the data transferred over the bus. The on-chip path from the data stored in parallel to the serial bit stream is not protected by the protocol. To eliminate any possible uncertainties at this point too, the SFCAN circuit incorporates a transmit-check unit. This unit reads back a transmitted message via the normal receive path from the bus interface and compares the data with those written into the message memory by the host controller. If any inconsistency of the data is detected, the current message will be invalidated by an error frame.

The transmit-check error counter TCEC is then incremented by 1. If this counter reaches 4 an error interrupt (bit TCI in the INT register) is generated, provided that this has not been masked (bit ETCI in the IMSK register). This count will also produce the Bus Off status.

The TCEC is set to 0 after a reset and can be read and also written for test purposes at any time.

Note: The transmit-check is an additional feature of the Siemens Full CAN Chip and is not part of the CAN protocol.

Time Stamp

It is impossible to determine from the received data in the message memory when they were received. So the host controller is unable to derive any information about the actuality or the repetition rate of the data.

To enable an indication of the time of reception for at least some of the messages, a 16-bit timer is implemented on the SAE 81C90/91. The content of this gets written into the time-stamp registers of the particular message when it is received (for the messages 0 through 7). There are two time-stamp bytes for each of the messages 0 through 7, and these hold the value of the 16-bit timer.

The actuality of a message is determined by subtracting the old time-stamp of a message, stored in the host controller, from the new one, with respect to the timer overflow bit.

Overflow of the timer can be detected by bit TSOV in the CTRL register. This bit does not trigger an interrupt and has to be reset by the host controller. Depending on the setting of bitfield TSP in register CTRL, the counter is fed with 1/32, 1/64, 1/128 or 1/256 of the bus clock. The momentary timer status can be read and set at any time. The timer starts at 0 after a reset and cannot be stopped.

I/O-Ports

There are two parallel I/O ports in the SAE 81C90, each with eight pins. These ports are configured pin by pin as input or output, depending on the contents of the port-direction register.

The output data for the port pins can be written (latched) into the port-latch register. Reading this register reproduces the contents of the latch. The levels on the port pins can be read from the port-pin register.

For the SAE 81C91 in its P-LCC-28-1 package, the pads of the I/O ports are not bonded and therefore unavailable to the user.

Note: Registers PxPDR and PxPL may be used for general purpose storage if the ports are not used.

Device Control and Registers

The operation of the SAE 81C90/91 is controlled via a number of registers. These registers allow initialization and function control, provide status information and configure the message objects. The upper part of the address space provides access to the data buffers of the message objects. The data buffers are ordered sequentially as shown in the table below.

The register map on the next page summarizes the other registers (i.e. except the data registers) ordered by their address, while the following pages describe these registers in more detail from a functional point of view.

Note: Locations marked "*Reserved*" in the register map must not be written in initialization mode. This also applies to locations 60_H through $7F_H$.

Data Registers

Address	Function	
80_H	Byte 0	Message 0
81_H	Byte 1	
82_H	Byte 2	
83_H	Byte 3	
84_H	Byte 4	
85_H	Byte 5	
86_H	Byte 6	
87_H	Byte 7	
88_H	Byte 0	Message 1
89_H	Byte 1	
:	:	:
$F6_H$	Byte 6	Message 14
$F7_H$	Byte 7	
$F8_H$	Byte 0	Message 15
$F9_H$	Byte 1	
FA_H	Byte 2	
FB_H	Byte 3	
FC_H	Byte 4	
FD_H	Byte 5	
FE_H	Byte 6	
FF_H	Byte 7	

Register Map (ordered by address)

Addr.	Reg. Name	Reset	Addr.	Reg. Name	Reset	Addr.	Reg. Name	Reset
00_H	BL1	00_H	20_H	*Reserved*	---	40_H	DR0H	UU_H
01_H	BL2	00_H	21_H	*Reserved*	---	41_H	DR0L	UU_H
02_H	OC	00_H	22_H	*Reserved*	---	42_H	DR1H	UU_H
03_H	BRP	00_H	23_H	*Reserved*	---	43_H	DR1L	UU_H
04_H	RRR1	00_H	24_H	*Reserved*	---	44_H	DR2H	UU_H
05_H	RRR2	00_H	25_H	*Reserved*	---	45_H	DR2L	UU_H
06_H	RIMR1	00_H	26_H	*Reserved*	---	46_H	DR3H	UU_H
07_H	RIMR2	00_H	27_H	*Reserved*	---	47_H	DR3L	UU_H
08_H	TRS1	00_H	28_H	P0PDR	00_H	48_H	DR4H	UU_H
09_H	TRS2	00_H	29_H	P0PR	XX_H	49_H	DR4L	UU_H
$0A_H$	IMSK	00_H	$2A_H$	P0LR	00_H	$4A_H$	DR5H	UU_H
$0B_H$	*Reserved*	---	$2B_H$	*Reserved*	---	$4B_H$	DR5L	UU_H
$0C_H$	*Reserved*	---	$2C_H$	P1PDR	00_H	$4C_H$	DR6H	UU_H
$0D_H$	*Reserved*	---	$2D_H$	P1PR	XX_H	$4D_H$	DR6L	UU_H
$0E_H$	*Reserved*	---	$2E_H$	P1LR	00_H	$4E_H$	DR7H	UU_H
$0F_H$	*Reserved*	---	$2F_H$	*Reserved*	---	$4F_H$	DR7L	UU_H
10_H	MOD	00_H	30_H	TSR0H	UU_H	50_H	DR8H	UU_H
11_H	INT	00_H	31_H	TSR0L	UU_H	51_H	DR8L	UU_H
12_H	CTRL	00_H	32_H	TSR1H	UU_H	52_H	DR9H	UU_H
13_H	*Reserved*	---	33_H	TSR1L	UU_H	53_H	DR9L	UU_H
14_H	CC	01_H	34_H	TSR2H	UU_H	54_H	DR10H	UU_H
15_H	TCEC	00_H	35_H	TSR2L	UU_H	55_H	DR10L	UU_H
16_H	TCD	XX_H	36_H	TSR3H	UU_H	56_H	DR11H	UU_H
17_H	*Reserved*	---	37_H	TSR3L	UU_H	57_H	DR11L	UU_H
18_H	TRR1	00_H	38_H	TSR4H	UU_H	58_H	DR12H	UU_H
19_H	TRR2	00_H	39_H	TSR4L	UU_H	59_H	DR12L	UU_H
$1A_H$	RRP1	00_H	$3A_H$	TSR5H	UU_H	$5A_H$	DR13H	UU_H
$1B_H$	RRP2	00_H	$3B_H$	TSR5L	UU_H	$5B_H$	DR13L	UU_H
$1C_H$	TSCH	00_H	$3C_H$	TSR6H	UU_H	$5C_H$	DR14H	UU_H
$1D_H$	TSCL	00_H	$3D_H$	TSR6L	UU_H	$5D_H$	DR14L	UU_H
$1E_H$	*Reserved*	---	$3E_H$	TSR7H	UU_H	$5E_H$	DR15H	UU_H
$1F_H$	*Reserved*	---	$3F_H$	TSR7L	UU_H	$5F_H$	DR15L	UU_H

Note: The locations marked "UU_H" are not changed upon a reset.
After a power on reset they are undefined (XX_H), of course.

Descriptor Registers

A descriptor register is available for each message object and contains the eleven bits of the message identifier (ID.0 through ID.10), the remote-transmission-request bit (RTR) and the data length code (DLC) of a message.

DRnH	7	6	5	4	3	2	1	0
Address: XX_H	**ID.10**	**ID.9**	**ID.8**	**ID.7**	**ID.6**	**ID.5**	**ID.4**	**ID.3**
Reset Value: UU_H	rw	rw	rw	rw	rw	rw	rw	rw

DRnL	7	6	5	4	3	2	1	0
Address: XX_H	**ID.2**	**ID.1**	**ID.0**	**RTR**	**DLC**			
Reset Value: UU_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	**Function**
DLC	**Data Length Code** Defines the number of data bytes of message n. Defined values are 0000...1000, i.e. 0...8 bytes. Other values are not permitted.
RTR	**Remote Transmission Request Bit** '0': This message operates as a data frame. '1': This message operates as a remote frame. **Note:** See description and table below.
ID.10-0	**Identifier** Identifier associated with message n, controls the acceptance of received frames and is inserted into transmitted frames.

n = 0...15

Bit RTR determines the function of the corresponding message object when it is transmitted, and its reaction on a received data frame or remote frame.
The table below summarizes the message object's behaviour in the different cases.

RTR bit	**Object is transmitted**	**Matching Data Frame received**	**Matching Remote Frame received**
'0' (data frame)	The message object is transmitted as a standard data frame.	The data frame is stored in the message object.	The remote frame is ignored.
'1' (remote frame)	The message object is transmitted as a remote frame (i.e. a request).	The data frame is ignored and not stored.	The message object is sent as a data frame.

Note: For the transmission of remote frames (RTR = 1) the data-length-code should be set to '0'.

Descriptor Register Arrangement

Address	Function	
40_H	High Byte	Descriptor Register for Message Object 0
41_H	Low Byte	
42_H	High Byte	Descriptor Register for Message Object 1
43_H	Low Byte	
:	:	:
$5C_H$	High Byte	Descriptor Register for Message Object 14
$5D_H$	Low Byte	
$5E_H$	High Byte	Descriptor Register for Message Object 15
$5F_H$	Low Byte	

Control Register Summary

Register Name	Address	Function	Reset Value	Read Write [1]
OC	02_H	Output-control register	00_H	r/w, I
CC	14_H	Clock-control register	01_H	wo
CTRL	12_H	Control register	00_H	r/w
MOD	10_H	Mode/status register	00_H	r/w
INT	11_H	Interrupt register	00_H	r/w
IMSK	$0A_H$	Interrupt-mask register	00_H	r/w
BL1	00_H	Bit-length register 1	00_H	r/w, I
BL2	01_H	Bit-length register 2	00_H	r/w, I
BRP	03_H	Baud-rate prescaler	00_H	wo, I
RRR1	04_H	Receive-ready register 1	00_H	r/w
RRR2	05_H	Receive-ready register 2	00_H	r/w
RIMR1	06_H	Receive-interrupt-mask register 1	00_H	r/w
RIMR2	07_H	Receive-interrupt-mask register 2	00_H	r/w
TRSR1	08_H	Transmit-request-set register 1	00_H	r/w
TRSR2	09_H	Transmit-request-set register 2	00_H	r/w
TRRR1	18_H	Transmit-request-reset register 1	00_H	wo
TRRR2	19_H	Transmit-request-reset register 2	00_H	wo
RRPR1	$1A_H$	Remote-request-pending register 1	00_H	ro
RRPR2	$1B_H$	Remote-request-pending register 2	00_H	ro
TSCH	$1C_H$	Time-Stamp counter high byte	00_H	r/w
TSCL	$1D_H$	Time-Stamp counter low byte	00_H	r/w
TCEC	15_H	Transmit-check error counter	00_H	r/w
TCD	16_H	Transmit-check data register	XX	ro
P0PDR	28_H	Port 0 port-direction register	00_H	r/w
P1PDR	$2C_H$	Port 1 port-direction register	00_H	r/w
P0LR	$2A_H$	Port 0 latch register	00_H	r/w
P1LR	$2E_H$	Port 1 latch register	00_H	r/w
P0PR	29_H	Port 0 pin register	XX_H	ro
P1PR	$2D_H$	Port 1 pin register	XX_H	ro

Note: [1] **ro:** read only, **r/w:** read and write access, **wo:** write only, **I:** access only with bit IM set.

Output-Control Register

The output drivers of the SAE 81C90/91's transmit pins (TXn) can be individually configured. Thus they can be adapted to the requirements of the external bs system.

OC	7	6	5	4	3	2	1	0
Address: 02_H	**OCTP1**	**OCTN1**	**OCP1**	**OCTP0**	**OCTN0**	**OCP0**	**OCM**	
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	Function
OCM	**Output Mode** '0X': **Normal Mode**: TX0 = Bit Sequence, TX1 = Bit Sequence. '10': **Test Mode**: TX0 = Bit Sequence, TX1 = RX0. '11': **Clock Mode**: TX0 = Bit Sequence, TX1 = Bit Clock.
OCPn	**Output Polarity** '0': Output is driven directly with CAN data. '1': Output is driven with inverted CAN data.
OCTNn	**Negative Output Transistor Control** '0': The low side output transistor TnN is disabled. '1': The low side output transistor TnN drives the pin according to data.
OCTPn	**Positive Output Transistor Control** '0': The high side output transistor TnP is disabled. '1': The high side output transistor TnP drives the pin according to data.

n = 0, 1

Note: This register can only be written when bit IM (MOD.0) is set.

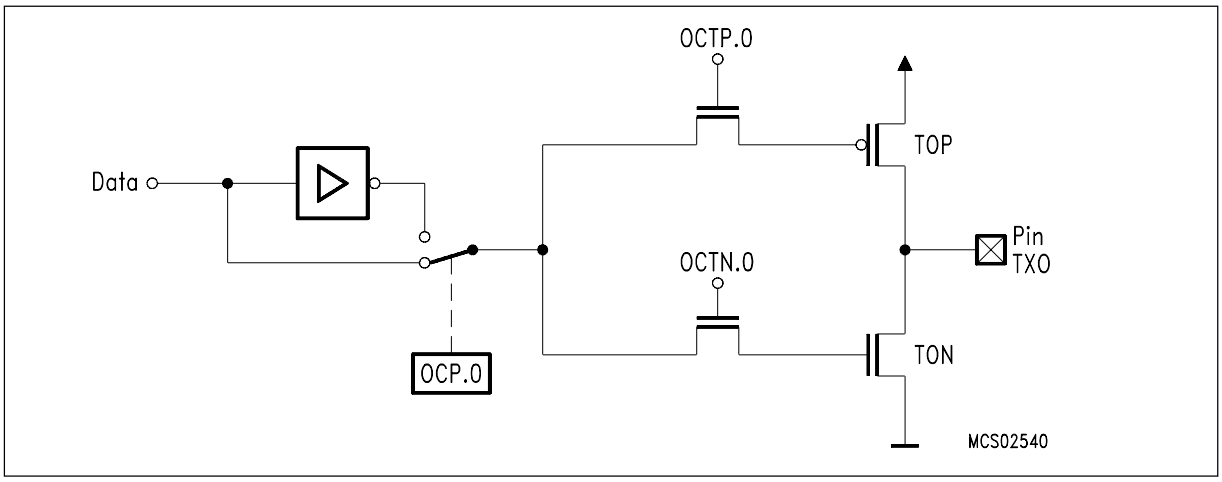


Figure 5
Output Control Circuitry

Output Programming

OCTP.n	OCTN.n	OCP.n	Data	TnP	TnN	TXn-Level
0	0	0	0 = dominant	OFF	OFF	float
0	0	0	1 = recessive	OFF	OFF	float
0	0	1	0	OFF	OFF	float
0	0	1	1	OFF	OFF	float
0	1	0	0	OFF	ON	LOW
0	1	0	1	OFF	OFF	float
0	1	1	0	OFF	OFF	float
0	1	1	1	OFF	ON	LOW
1	0	0	0	OFF	OFF	float
1	0	0	1	ON	OFF	HIGH
1	0	1	0	ON	OFF	HIGH
1	0	1	1	OFF	OFF	float
1	1	0	0	OFF	ON	LOW
1	1	0	1	ON	OFF	HIGH
1	1	1	0	ON	OFF	HIGH
1	1	1	1	OFF	ON	LOW

TnP is the output transistor switching to V_{DD}, TnN switches to V_{SS}.
TXn is the output level at the transmit pin.

Clock Control Register

The Clock Control Register determines the output frequency at pin CLKOUT which is derived from the oscillator frequency.

CCR	7	6	5	4	3	2	1	0
Address: 14_H	**-**	**-**	**-**	**-**	**CC**			
Reset Value: 01_H	-	-	-	-	w	w	w	w

Bit(field)	Function
CC	**Clock Output Control** '0000': $f_{CLKOUT} = f_{OSC}$ '0001': $f_{CLKOUT} = f_{OSC} / 2$ '0010': $f_{CLKOUT} = f_{OSC} / 4$ '0011': $f_{CLKOUT} = f_{OSC} / 6$ '0100': $f_{CLKOUT} = f_{OSC} / 8$ '0101': $f_{CLKOUT} = f_{OSC} / 10$ '0110': $f_{CLKOUT} = f_{OSC} / 12$ '0111': $f_{CLKOUT} = f_{OSC} / 14$ '1XXX': f_{CLKOUT} = LOW (clock output is switched off)

The Clock Control Register requires a special protocol for writing in order to prevent the clock output from being changed inadvertently:

- Step 1: Write 80_H to CC
- Step 2: Write desired value to CC (bits 7...4 must be '0000')

Note: Not defined bit positions must be '0' for write accesses.

Control Register

CTRL	7	6	5	4	3	2	1	0
Address: 12_H	**RX**	**TST**	**TSP**		**TSOV**	**SME**	**TCE**	**MM**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	Function
MM	**Monitor Mode** '0': Message object 0 operates like all other objects. '1': Message object 0 receives all identifiers that are not accepted by other objects (corresponds to a Basic CAN receive register).
TCE	**Transmit Check Enable** '0': If the transmit check detects an error, there is no intervention. '1': If the transmit check detects an error, the message is invalidated by an error frame and the error counter TCEC is incremented by 1. If the counter reaches 4, the Bus Off status is initiated and, if enabled, an interrupt (TCI) is generated.
SME	**Sleep Mode Enable** '0': Normal operation. '1': The sleep mode is enabled: the crystal oscillator is deactivated, all other activities are inhibited. The wake up is done by a reset signal or by an active signal at the $\overline{CS}$ pin or by an input edge going from recessive to dominant at pin Rx0 or Rx1.
TSOV	**Time Stamp Overflow** '0': There has been no overflow '1': There was at least one overflow of the time-stamp timer.
TSP	**Time Stamp Prescaler** (Defines the input clock of the time-stamp timer) '00': f_{BL} / 32 '01': f_{BL} / 64 '10': f_{BL} / 128 '11': f_{BL} / 256 (For f_{BL} see baud-rate prescaler BRP).
TST	**Time Stamp Test** '0': The prescaler is activated. '1': The time-stamp prescaler is deactivated. (Only for testing purposes, bit IM = MOD.0 must be set to '1').
RX	**Input Monitor RX** This bit monitors the actual state of the digital input pin RX0.

Mode/Status-Register

MOD	7	6	5	4	3	2	1	0
Address: 10_H	**ADE**	**RS**	**TC**	**TWL**	**RWL**	**BS**	**RES**	**IM**
Reset Value: 00_H	rw	r	r	r	r	r	rw	rw

Bit(field)	Function
IM	**Init Mode** '0': Normal mode. '1': Initialization mode: write access to the configuration registers BL1, BL2, OC, BRP is enabled. If the bit stays set, the chip enters the normal mode, with enabled access to the configuration registers. If this bit is set in conjunction with bit RES a hard software reset is activated.
RES	**Reset Request** '0': Normal mode. '1': The chip enters the reset state: – if bit IM = '0' a soft software reset takes place. – if bit IM = '1' a hard software reset takes place. Further details see below.
BS	**Bus State (read only)** '0': Normal mode. '1': Bus Off state, the IC does not participate in bus activities.
RWL	**Receiver Warning Level (read only)** '0': Receive-error counter below 96. '1': Receive-error counter equal or above 96.
TWL	**Transmit Warning Level (read only)** '0': Transmit-error counter below 96. '1': Transmit-error counter equal or above 96.
TC	**Transmission Complete (read only)** '0': The last transmission request is not yet executed successfully. '1': The last transmission request was executed successfully.
RS	**Receive State (read only)** '0': No reception active. '1': Currently the SAE 81C90/91 is in receive mode.
ADE	**Auto Decrement Enable** '0': No automatic address decrement. '1': With every read or write access using the serial synchronous interface SI the address is automatically decremented by one. So data can be accessed sequentially without the need of writing a new address.

Notes on Bit TC

Scanning this bit is particularly useful if only one transmission is active. If there are several transmission jobs at the same time, it is better to scan the transmit-request register, because bit TC may possibly only be set very briefly between acknowledgment of the previous message and the start of the next one.

Notes on Bit RES and IM and reset modes

There are three different reset modes implemented in the SAE 81C90/91:

hardware reset (activated by low level on pin $\overline{\text{RES}}$)

hard software reset (activated by setting both bits RES and IM to 1)

soft software reset (activated by setting bit RES to 1 and bit IM to 0)

The only difference between hardware and hard software reset affect bits RES and IM, that are not changed by software reset.

With soft software reset the registers RRR1, RRR2, TRSR1, TRSR2, RRPR1 and RRPR2 are cleared, all bus activities are stopped, the error counters are not cleared, the Bus Off state is cancelled only after 128 idle phases (according to the CAN protocol 1 idle phase = 11 recessive bits in sequence). Simply spoken a soft software reset interrupts and cancels all bus activities and - if necessary - recovers from Bus Off state.

Notes on Bit RS

Bit RS directly reflects the internal status.
RS is ’0’ during transmission or when the SAE 81C90/91 is idle.
RS is ’1’ during reception or during the synchronization after a reset.

Interrupt Register

INT	7	6	5	4	3	2	1	0
Address: 11_H	**TCI**	**EPI**	**BOI**	**WUPI**	**RFI**	**WLI**	**TI**	**RI**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	Function
RI	**Receive Interrupt** After a valid message has been received and filed, this bit is set and an interrupt generated. This bit will remain set until all bits of the registers RRR1 and RRR2 are reset.
TI	**Transmit Interrupt** This bit is set and an interrupt generated as soon as a transmit request has been processed.
WLI	**Warning Level Interrupt** If at least one of the two error counters is greater than or equals 96, this bit is set and an interrupt generated.
RFI	**Remote Frame Interrupt** This interrupt is generated after reception of a remote frame.
WUPI	**Wake Up Interrupt** After a wake-up this bit is set and an interrupt generated.
BOI	**Bus Off Interrupt** This bit is set and an interrupt generated when the Bus Off status is entered.
EPI	**Error Passive Interrupt** If at least one of the two error counters is greater than or equals 128, this bit is set and an interrupt generated.
TCI	**Transmit Check Interrupt** If the transmit-check error counter reaches 4, this bit is set and an interrupt generated.

Note: All bits of this register must be reset by software. This is done by writing ’0’ to the respective bit location, writing ’1’ has no effect.
An interrupt is only generated if the respective IMSK bit is set. The bits in this register are set independent of register IMSK (see below).
The interrupt output is active for at least one bit time. The interrupt output is deactivated when all enabled request bits are cleared. A request bit is enabled by setting its corresponding mask bit. Masked request bits do not activate the interrupt output.

Interrupt-Mask Register

These mask bits determine if an event activates the INT pin. They do not influence the INT register.

IMSK	7	6	5	4	3	2	1	0
Address: $0A_H$	**ETCI**	**EEPI**	**EBOI**	**EWUPI**	**ERFI**	**EWLI**	**ETI**	**ERI**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	Function
ERI	**Enable Receive Interrupt** '0': No receive interrupt enabled. '1': Receive interrupts are enabled.
ETI	**Enable Transmit Interrupt** '0': No transmit interrupt enabled. '1': Completed transmit jobs generate interrupts.
EWLI	**Enable Warning Level Interrupt** '0': No warning level interrupt enabled. '1': There is an interrupt when the warning level is reached.
ERFI	**Enable Remote Frame Interrupt** '0': No remote frame interrupt enabled. '1': A receive interrupt is generated after receiving a remote frame
EWUPI	**Enable Wake Up Interrupt** '0': No wake up interrupt enabled. '1': Wake-up interrupt is enabled.
EBOI	**Enable Bus Off Interrupt** '0': No bus off interrupt enabled. '1': Bus off interrupt is enabled.
EEPI	**Enable Error Passive Interrupt** '0': No error passive interrupt enabled. '1': Error passive interrupt is enabled.
ETCI	**Enable Transmit Check Error Interrupt** '0': No transmit check interrupt enabled. '1': Transmit-check error interrupt is enabled.

Bit-Length Registers

BL1	7	6	5	4	3	2	1	0
Address: 00_H	**SAM**	**TS2**			**TS1**			
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	Function
TS1	**Length of Timing Segment 1** (TSeg1). t_{TSeg1} = (TS1 + 1) × t_{SCL}. For t_{SCL} see baud-rate prescaler BRP.
TS2	**Length of Timing Segment 2** (TSeg2). t_{TSeg2} = (TS2 + 1) × t_{SCL}. For t_{SCL} see baud-rate prescaler BRP.
SAM	**Sample Rate** '0': Input signal is sampled once per bit. '1': Input signal is sampled three times per bit. **Note:** Bit SAM should only be set to '1' using very low baud rates.

BL2	7	6	5	4	3	2	1	0
Address: 01_H	**IPOL**	**DI**	**–**	**–**	**–**	**SM**	**SJW**	
Reset Value: 00_H	rw	rw	-	-	-	rw	rw	rw

Bit(field)	Function
SJW	**Maximum Synchronization Jump Width.** $t_{SJWidth}$ = (SJW + 1) × t_{SCL}. For t_{SCL} see baud-rate prescaler BRP.
SM	**Speed Mode** (Defines edge used for synchronization) '0': Recessive to dominant is used. '1': Both edges are used. **Note:** According to the CAN specification this bit should not be set to '1'.
DI	**Digital Input** '0': The input signal is applied to the input comparator. [1)] '1': The input signal on pin RX0 is evaluated digitally. The input comparator is inactive. Pin RX1 should be on V_{SS}.
IPOL	**Input Polarity** '0': The input level remains unaltered. '1': The input level is inverted.

Note: Not defined bit positions must be '0' for write accesses.
The Bit Length Registers BL1 and BL2 can only written while bit IM (MOD.0) is set.
[1)] If the bus lines work according to the ISO specification, additional circuitry is necessary for interconnection of the input comparator to the bus lines.

Baud Rate Prescaler Register

The register is not readable and can only be written when bit IM (MOD.0) is set.

BRPR	7	6	5	4	3	2	1	0
Address: 03_H	–	–	BRP					
Reset Value: 00_H	-	-	w	w	w	w	w	w

Bit(field)	Function
BRP	**Baud Rate Prescaler** This prescaler determines the period of the system clock: $t_{SCL} = (BRP + 1) \times 2 \times t_{OSC}$, where $t_{OSC} = 1 / f_{crystal}$.

Note: Not defined bit positions must be ’0’ for write accesses.

The bit length t_{BL} is computed as follows:

$t_{BL} = t_{TSeg1} + t_{TSeg2} + 1\ t_{SCL}$

The baudrate BR can be computed with the following formula:

$BR = f_{crystal} / (2 \times (BRP + 1) \times (TS1 + TS2 + 3))$

Note: BRP see Baud Rate Prescaler Register
TS1 see Bit Length Register 1
TS2 see Bit Length Register 1

Receive-Ready Registers

RRR2	7	6	5	4	3	2	1	0
Address: 05_H	**RR15**	**RR14**	**RR13**	**RR12**	**RR11**	**RR10**	**RR9**	**RR8**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

RRR1	7	6	5	4	3	2	1	0
Address: 04_H	**RR7**	**RR6**	**RR5**	**RR4**	**RR3**	**RR2**	**RR1**	**RR0**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	Function
RRn	**Receive Ready Bit** '0': No new message received in object n. '1': A new message has been received and stored in object n.

These register bits can be reset by writing '0' to the respective bit, writing '1' has no effect. Bit RRn is set when a message has arrived and been written into the memory location of message n. Setting this bit by hardware can generate a receive interrupt, which can be blocked by bit RIMn in the receive-interrupt-mask register.
Bits RRn must be reset by software.

Receive-Interrupt-Mask Registers

Setting bit RIMn enables a receive interrupt to be generated if the receive-ready bit RRn has been set, i.e. a message has arrived and was written into the memory location of message n.

RIMR2	7	6	5	4	3	2	1	0
Address: 07_H	**RIM15**	**RIM14**	**RIM13**	**RIM12**	**RIM11**	**RIM10**	**RIM9**	**RIM8**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

RIMR1	7	6	5	4	3	2	1	0
Address: 06_H	**RIM7**	**RIM6**	**RIM5**	**RIM4**	**RIM3**	**RIM2**	**RIM1**	**RIM0**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	Function
RIMn	**Receive Interrupt Mask Bit** '0': No interrupt upon reception of object n. '1': When a new message is stored in object n an interrupt is generated.

Note: Bit ERI in the interrupt-mask register IM blocks all receive interrupts, even if bits RIMn are set.

Transmit Request Registers

The Transmit Request Set Registers provide a transmission request bit (TRSn) for each message object. Setting a transmission request bit causes the respective message x to be transmitted. The bit is cleared by hardware after transmission. Several bits can be set simultaneously. In this way all messages whose request bits are set are transmitted in turn, starting with the memory location with the highest number.

Note: A transmission request bit is set by writing '1' to the respective bit location (TRSn). Writing '0' has no effect.

TRSR2	7	6	5	4	3	2	1	0
Address: 09_H	**TRS15**	**TRS14**	**TRS13**	**TRS12**	**TRS11**	**TRS10**	**TRS9**	**TRS8**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

TRSR1	7	6	5	4	3	2	1	0
Address: 08_H	**TRS7**	**TRS6**	**TRS5**	**TRS4**	**TRS3**	**TRS2**	**TRS1**	**TRS0**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	Function
TRSn	**Transmit Request Set Bit** '0': No change of the respective transmit request bit. '1': The respective transmit request bit is cleared.

n = 0...15

The Transmit Request Reset Registers provide a transmit request reset bit (TRRn) for each transmit request bit TRSn, i.e. for each message object.

Writing '1' to a TRRn bit clears the corresponding transmission request bit TRSn. This causes a transmission request, initiated by the corresponding bit TRSn, to be cancelled, provided that it is not currently processed.

This scheme avoids conflicts when writing to register bits while they are cleared by hardware because of a completed transmission.

Note: Registers TRRRx cannot be read.

TRRR2	7	6	5	4	3	2	1	0
Address: 19_H	**TRR15**	**TRR14**	**TRR13**	**TRR12**	**TRR11**	**TRR10**	**TRR9**	**TRR8**
Reset Value: 00_H	w	w	w	w	w	w	w	w

TRRR1	7	6	5	4	3	2	1	0
Address: 18_H	**TRR7**	**TRR6**	**TRR5**	**TRR4**	**TRR3**	**TRR2**	**TRR1**	**TRR0**
Reset Value: 00_H	w	w	w	w	w	w	w	w

Bit(field)	Function
TRRx	**Transmit Request Reset Bit** '0': No change of the respective transmit request bit. '1': The respective transmit request bit is cleared.

Remote-Request-Pending Registers

RRPR2	7	6	5	4	3	2	1	0
Address: $1B_H$	**RRP15**	**RRP14**	**RRP13**	**RRP12**	**RRP11**	**RRP10**	**RRP9**	**RRP8**
Reset Value: 00_H	r	r	r	r	r	r	r	r

RRPR1	7	6	5	4	3	2	1	0
Address: $1A_H$	**RRP7**	**RRP6**	**RRP5**	**RRP4**	**RRP3**	**RRP2**	**RRP1**	**RRP0**
Reset Value: 00_H	r	r	r	r	r	r	r	r

Bit(field)	Function
RRPn	**Remote Request Pending Bit** '0': No remote request pending. '1': A remote request (remote frame) for message n was received but is not yet answered by the transmission of the corresponding data frame.

n = 0...15

Message Time Stamp

This mechanism stores the time at which a specific message was received, i.e. it assigns a time stamp to that message. For this purpose the contents of the free-running time stamp counter TSC is copied to the time stamp register TSRn of the respective message object upon reception of this message.

The Time Stamp Counter Registers provide access to the free-running time stamp counter.

TSCH	7	6	5	4	3	2	1	0
Address: $1C_H$	**TSC.15**	**TSC.14**	**TSC.13**	**TSC.12**	**TSC.11**	**TSC.10**	**TSC.9**	**TSC.8**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

TSCL	7	6	5	4	3	2	1	0
Address: $1D_H$	**TSC.7**	**TSC.6**	**TSC.5**	**TSC.4**	**TSC.3**	**TSC.2**	**TSC.1**	**TSC.0**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

Bit(field)	Function
TSC	**Time Stamp Counter** Current contents of the free running time stamp counter.

The Time-Stamp Registers are available for each of message objects 0...7 (see table below) and contain the time-stamp of the corresponding message. These registers can only be read.

TSRnH	7	6	5	4	3	2	1	0
Address: $3X_H$	**TSn.15**	**TSn.14**	**TSn.13**	**TSn.12**	**TSn.11**	**TSn.10**	**TSn.9**	**TSn.8**
Reset Value: UU_H	r	r	r	r	r	r	r	r

TSRnL	7	6	5	4	3	2	1	0
Address: $3X_H$	**TSn.7**	**TSn.6**	**TSn.5**	**TSn.4**	**TSn.3**	**TSn.2**	**TSn.1**	**TSn.0**
Reset Value: UU_H	r	r	r	r	r	r	r	r

Bit(field)	Function
TSn	**Time Stamp n** A 16-bit timer value to identify the time of reception of message n.

n = 0...7

Time Stamp Register Table

Address	Function	
30_H	High Byte	Time-Stamp 0
31_H	Low Byte	
32_H	High Byte	Time-Stamp 1
33_H	Low Byte	
:	:	:
$3C_H$	High Byte	Time-Stamp 6
$3D_H$	Low Byte	
$3E_H$	High Byte	Time-Stamp 7
$3F_H$	Low Byte	

Transmit Check Error Counter

TCEC	7	6	5	4	3	2	1	0
Address: 15_H	**-**	**-**	**-**	**-**	**-**	**TCECV**		
Reset Value: 00_H	-	-	-	-	-	rw	rw	rw

Bit(field)	Function
TCECV	**Transmit Check Error Counter Value** Number of errors detected by the transmit check unit. When a count of 4 is reached an interrupt is generated if enabled. If bit TCE (CTRL.1) is set to ’1’ the Bus Off status is entered in this case.

Note: Not defined bit positions must be ’0’ for write accesses.

Transmit Check Data Register

This register supports an error analysis when a transmit check error is encountered. Reading TCD provides the byte which was actually being sent when the error occurred.

TCD	7	6	5	4	3	2	1	0
Address: 16_H	**Data Byte**							
Reset Value: XX_H	r	r	r	r	r	r	r	r

Bit(field)	Function
Data Byte	The data byte which was attempted to be sent while a transmit check error was encountered.

Port Control Registers

These registers control the parallel ports P0 and P1 which are provided in the SAE 81C90.

The Port Direction Registers PxPDR select each port pin separately for input (PxPDR.n='0') or output (PxPDR.n='1'). After reset the ports are switched as inputs.

P1PDR	7	6	5	4	3	2	1	0
Address: $2C_H$	**P1PD.7**	**P1PD.6**	**P1PD.5**	**P1PD.4**	**P1PD.3**	**P1PD.2**	**P1PD.1**	**P1PD.0**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

P0PDR	7	6	5	4	3	2	1	0
Address: 28_H	**P0PD.7**	**P0PD.6**	**P0PD.5**	**P0PD.4**	**P0PD.3**	**P0PD.2**	**P0PD.1**	**P0PD.0**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

The Port Latch Registers PxLR store the output data for those port pins that are switched to output.

P1LR	7	6	5	4	3	2	1	0
Address: $2E_H$	**P1L.7**	**P1L.6**	**P1L.5**	**P1L.4**	**P1L.3**	**P1L.2**	**P1L.1**	**P1L.0**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

P0LR	7	6	5	4	3	2	1	0
Address: $2A_H$	**P0L.7**	**P0L.6**	**P0L.5**	**P0L.4**	**P0L.3**	**P0L.2**	**P0L.1**	**P0L.0**
Reset Value: 00_H	rw	rw	rw	rw	rw	rw	rw	rw

The Port Pin Registers PxPR provide the current level of the port pins. These registers can only be read.

P1PR	7	6	5	4	3	2	1	0
Address: $2D_H$	**P1P.7**	**P1P.6**	**P1P.5**	**P1P.4**	**P1P.3**	**P1P.2**	**P1P.1**	**P1P.0**
Reset Value: 00_H	r	r	r	r	r	r	r	r

P0PR	7	6	5	4	3	2	1	0
Address: 29_H	**P0P.7**	**P0P.6**	**P0P.5**	**P0P.4**	**P0P.3**	**P0P.2**	**P0P.1**	**P0P.0**
Reset Value: 00_H	r	r	r	r	r	r	r	r

In parallel to the standard CMOS structure there are additional internal pullup devices of about 10...200 kΩ at each port pin.

Note: Registers PxPDR and PxPL may be used for general purpose storage if the ports are not used.

Bit Timing

A regular bit period is composed of the following three segments:

- synchronization segment
- timing segment 1
- timing segment 2.

The sampling point is between timing segment 1 and timing segment 2.

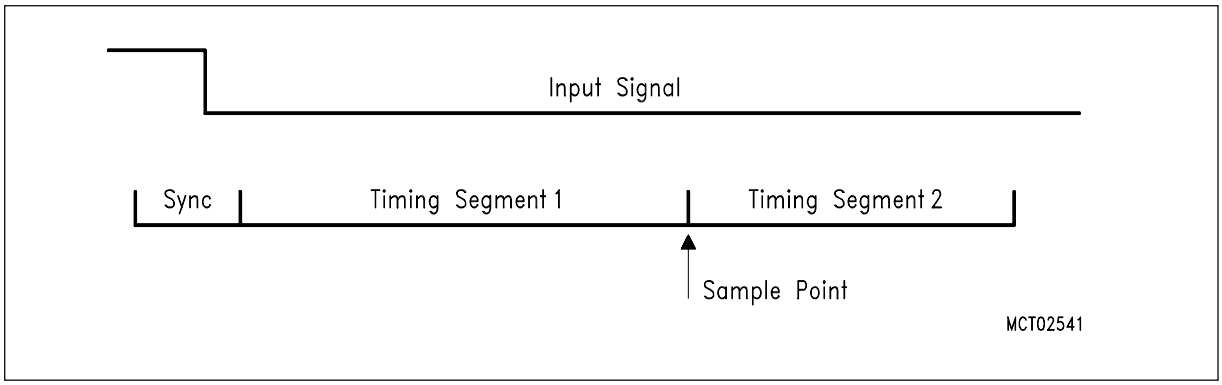


Figure 6
Bit Time Segments

Synchronization

The edge of the input signal is expected during the sync segment (duration = 1 system clock cycle = 1 t_{SCL}).

Timing Segment 1 (TSeg1)

Timing segment 1 determines the sampling point within a bit period. This point is always at the end of segment 1. The segment is programmable from 1 to 16 t_{SCL} (see bit-length register BL1).

Timing Segment 2 (TSeg2)

Timing segment 2 provides extra time for internal processing after the sampling point. The segment is programmable from 1 to 8 t_{SCL} (see bit-length register BL1).

Synchronization Jump Width (SJW)

To compensate for phase shifts between the oscillator frequencies of the different bus stations, each CAN controller must be able to synchronize to the relevant signal edge of the incoming signal. The synchronization jump width (SJW) determines the maximum number of system clock pulses by which the bit period can be lengthened or shortened for resynchronization. The synchronization jump width is programmable from 1 to 4 t_{SCL} (see bit-length register BL2).

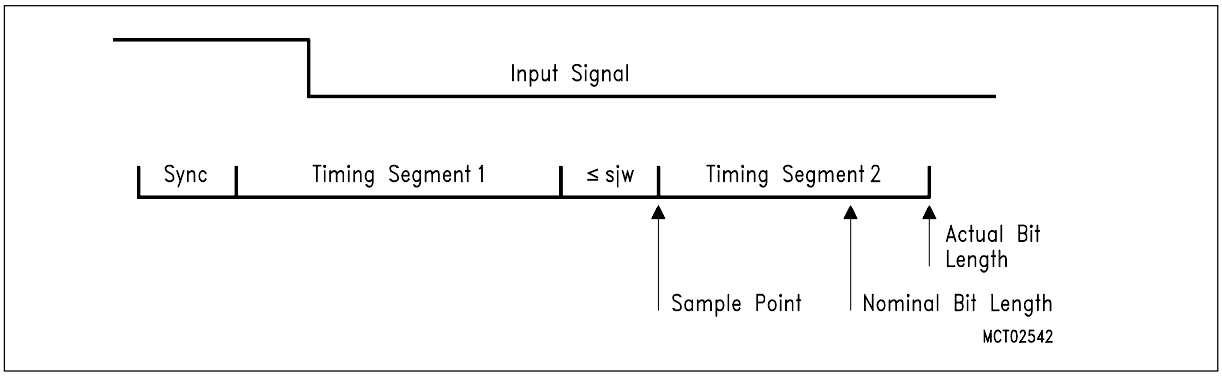


Figure 7
Lengthening a Bit Period

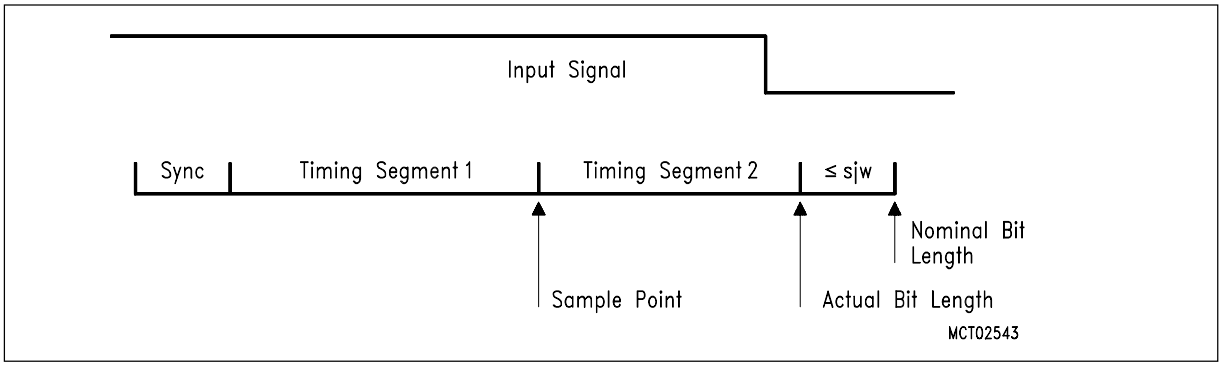


Figure 8
Shortening a Bit Period

Delay Times

The total delay is calculated from the following single delays:

- 2 × physical bus t_{Bus} (max. 100 ns acc. to CAN specification)
- 2 × input comparator t_{Comp} (depends on application circuit)
- 2 × output driver t_{Driver} (depends on application circuit)
- 1 × input to output of CAN controller t_{InOut} (max. 1 t_{SCL} + 80 ns)

$$t_{Delay} = 2 \times (t_{Bus} + t_{Comp} + t_{Driver}) + t_{InOut}$$

Recommendations

On the premise of the stated conditions, there are the following essential requirements to be maintained:

$t_{TSeg1} \geq t_{Seg2}$
$t_{TSeg1} \geq t_{Delay}$
$t_{TSeg2} > t_{SJW}$
$t_{TSeg2} \geq 3 \times t_{SCL} + t_{SJW}$ if bit SAM = 1 (otherwise bit recognition does not work).

Host Interfaces

There are two different host interfaces implemented in the SAE 81C90/91.

Data and addresses on a multiplexed 8-bit bus, compatible with Siemens microcontrollers (C5xx, C16x), can be transferred via the parallel interface (PI). Using the serial synchronous interface (SI), any host controller with a serial three-lead interface can be connected with.

The interface is selected by hardware through the wiring of the MS (Mode Select) pin. This pin may not be switched during operation. If there is a High level on the MS pin, the SI and thus pins DI, DO, CLK, $\overline{W}$ and TIM are activated, while pins AD5 through AD7, $\overline{RD}$, $\overline{WR}$ and ALE are inactive. A Low level on the MS pin switches to the PI and thus activates pins AD0 through AD7, $\overline{RD}$, $\overline{WR}$ and ALE.

Parallel Interface PI

The parallel interface uses a multiplexed 8-bit address/data bus. First the address of the required register is applied to the pins AD0 through AD7. A falling edge on pin ALE means that this address is transferred to an on-chip latch. After this, data can either be written into the selected register (pin $\overline{WR}$ = 0) or read from it (pin $\overline{RD}$ = 0) via the address/data bus. Pin $\overline{CS}$ must be 0 for the entire duration of the $\overline{RD}$/$\overline{WR}$ active time so that the circuit is activated.

Serial Synchronous Interface SI

If the SI is used the unused pins of PI must be set to inactive levels ($\overline{RD}$, $\overline{WR}$ to V_{DD} and ALE, AD5, AD6, AD7 to V_{SS}).

Communication on the SI is accomplished according to the following procedure:

Each access to the stand-alone Full-CAN circuit has to be started by activating the device ($\overline{CS}$ = 0). After the beginning of access, an address must be written first and then data can be read or written. The required function is determined by pin $\overline{W}$ ($\overline{W}$ = 1: read; $\overline{W}$ = 0: write). If the automatic decrementing of the address is activated (bit ADE in the MOD register), any number of data bytes can be accessed in succession. Finally the device has to be deactivated.

Procedure:

- Activate device ($\overline{CS}$ = 0)
- Set pin $\overline{W}$ to 1 for read, to 0 for write
- Write in address of first data byte
- Read out/write in one or more data bytes
- Deactivate device ($\overline{CS}$ = 1)

The most-significant bit is always output as the first bit of an address or a data byte.
Data from pin DI are transferred into the internal shift register with the **rising** edge of the clock. The active clock edge of pin DO is selectable via the pin TIM. If this pin is 0 the data are output from the shift register to pin DO with the **rising** clock edge (Timing A). If the pin TIM is 1, the output of data is done with the **falling** edge (Timing B).

The timing for reading and writing of two data bytes with automatic decrementing activated is illustrated below.

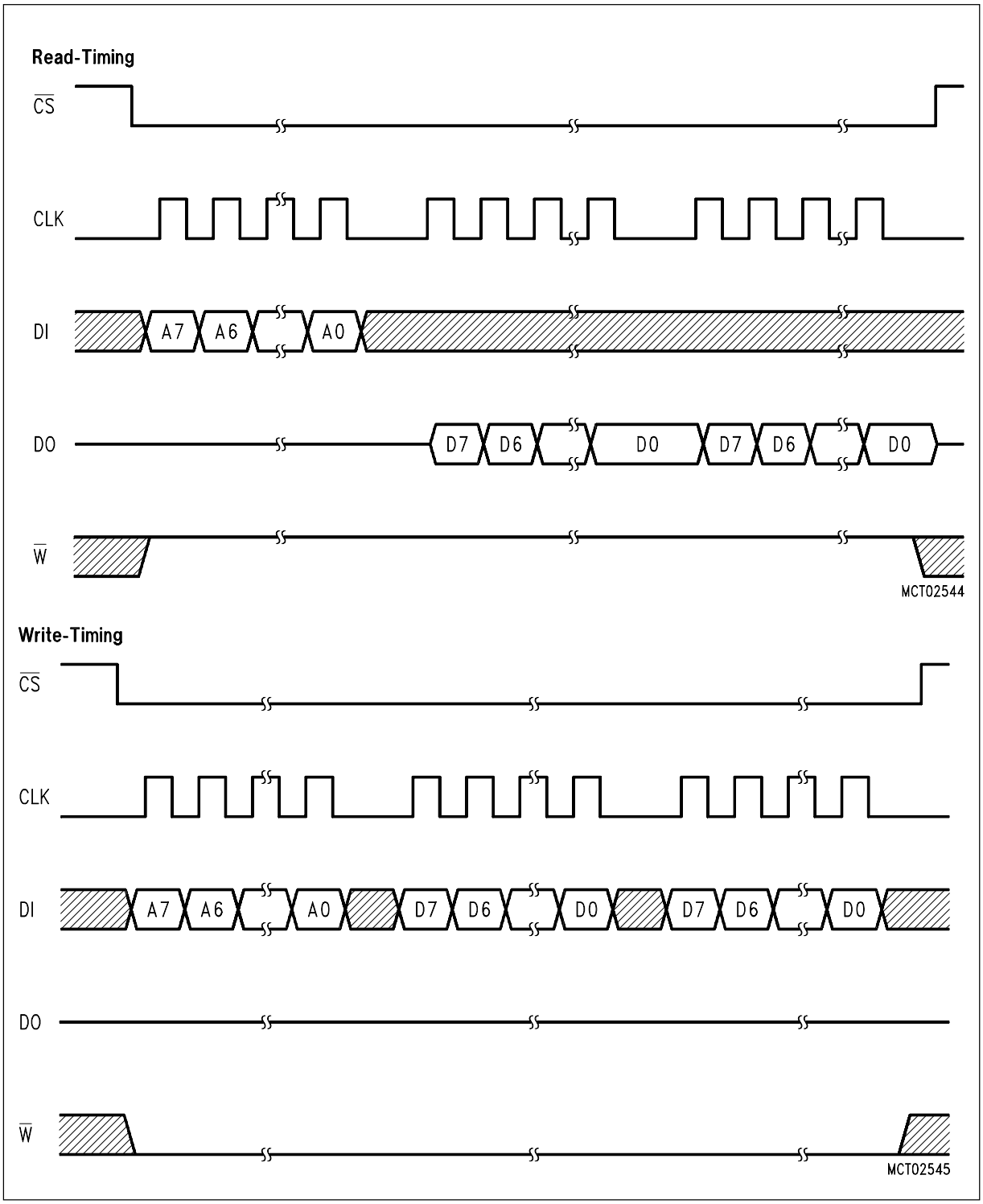


Figure 9
Serial Interface Timing (for 2 Data Bytes)

Absolute Maximum Ratings

Ambient temperature under bias (T_A): ..... – 40 to + 110 ˚C
Storage temperature (T_{ST})..... – 50 to + 150 ˚C
Voltage on V_{CC} pins with respect to ground (V_{SS}) ..... – 0.5 to + 6.0 V
Voltage on any pin with respect to ground (V_{SS}) .....– 0.5 to V_{CC} + 0.5 V
Input current on any pin during overload condition..... – 10 to + 10 mA
Absolute sum of all input currents during overload condition ..... |100 mA|
Power dissipation..... 0.5 W

Note: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. During overload conditions ($V_{IN} > V_{CC}$ or $V_{IN} < V_{SS}$) the voltage on pins with respect to ground (V_{SS}) must not exceed the values defined by the Absolute Maximum Ratings.

Parameter Interpretation

The parameters listed in the following partly represent the characteristics of the SAE 81C90/91 and partly its demands on the system. To aid in interpreting the parameters right, when evaluating them for a design, they are marked in column “Symbol”:

DC (**D**evice **C**haracteristics):
The logic of the SAE 81C90/91 will provide signals with the respective timing characteristics.

SR (**S**ystem **R**equirement):
The external system must provide signals with the respective timing characteristics to the SAE 81C90/91.

DC Characteristics

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = – 40 to + 110 °C

Parameter	Symbol		Limit Values		Unit	Test Condition
			min.	max.		
Input low voltage (all except XTAL1 and XTAL2)	V_{IL}	SR	0	0.3 V_{CC}	V	–
Input low voltage (XTAL1 and XTAL2)	V_{ILX}	SR	0	0.5	V	–
Input high voltage (all except XTAL1 and XTAL2)	V_{IH}	SR	0.7 V_{CC}	V_{CC}	V	–
Input high voltage (XTAL1 and XTAL2)	V_{IHX}	SR	V_{CC} – 1.0	V_{CC}	V	–
Comparator input voltage [1]	V_{CI}	SR	0.5	V_{CC} + 0.5	V	–
Common mode voltage [2]	V_{ICOM}	SR	1.5	V_{CC} – 1.5	V	–
Hysteresis [2]	V_{HYS}	DC	–	100 [3]	mV	–
Offset voltage [2]	V_{OFF}	DC	–	100 [3]	mV	–
Output low voltage (all except CLKOUT, TX0, TX1)	V_{OL}	DC	–	0.2 V_{CC}	V	I_{OL} = 1.6 mA
Output low voltage (CLKOUT)	V_{OLC}	DC	–	0.4	V	I_{OL1} = 10 mA
Output high voltage (all except CLKOUT, TX0, TX1)	V_{OH}	DC	0.8 V_{CC}	V_{CC}	V	I_{OH} = – 1.6 mA
Output high voltage (CLKOUT)	V_{OHC}	DC	V_{CC} – 0.8	V_{CC}	V	I_{OH} = – 10 mA
Input leakage current	I_I	DC	–	±1	µA	0 V < V_{IN} < V_{CC} [4]
Source output current (TX0, TX1)	I_{SRC}	DC	5	–	mA	V_O = V_{CC} – 1 V
Sink output current (TX0, TX1)	I_{SNK}	DC	5	–	mA	V_O = 1 V
Low end capacitance [5]	C_L	DC	6.8	12	pF	
Pin capacitance [2]	C_I	DC	–	10	pF	f = 1 MHz T_A = 25 °C
Power supply current	I_{CC}		–	30	mA	

Notes

1) If the bus lines work according to the ISO specification, additional circuitry is necessary for interconnection of the input comparator to the bus lines.
2) Not 100% tested, guaranteed by design characterization.
3) This value is a typical value!
4) This specification does not apply to the port pins (P00...P07, P10...P17) due to the implemented pullups!
5) In oscillator mode the size of the low-end capacitance must correspond to the specification of the crystal manufacturer. The optimum values depend on the selected crystal, the intended frequency and the actual application hardware (stray capacitances). 10 pF are recommended for C_L.
For best results keep the crystal circuitry connections as short as possible and keep the CLKOUT line away from it. If the CLKOUT signal is not required by the system it should be switched off.

AC Characteristics (General Timing)

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = – 40 to + 110 ˚C

Parameter	Symbol		Limit Values		Unit	Test Conditions
			min.	max.		
Oscillator period	t_{OSC}	SR	50	-	ns	
Clock input high time	t_H	SR	23.5	–	ns	
Clock input low time	t_L	SR	23.5	–	ns	
Reset pulse width	t_{RES}	SR	2	–	t_{OSC}	
Output rise time [1)]	t_{QR}	DC	–	40	ns	C_L = 70 pF
Output fall time [1)]	t_{QF}	DC	–	40	ns	C_L = 70 pF
CLKOUT rise time [1)]	t_{QRC}	DC	–	20	ns	C_L = 50 pF
CLKOUT fall time [1)]	t_{QFC}	DC	–	20	ns	C_L = 50 pF

1) Not 100% tested, guaranteed by design characterization.

AC Characteristics (SI Timing)

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = – 40 to + 110 ˚C; C_L = 50 pF

Parameter	Symbol		Limit Values		Unit	Test Conditions
			min.	max.		
Chip Select Setup	t_{CSS}	SR	10		ns	
Clock High Time	t_{CH}	SR	1.5 t_{OSC} + 10		ns	
Clock Low Time	t_{CL}	SR	1.5 t_{OSC} + 10		ns	
Clock Period	t_C	SR	4 t_{OSC}		ns	
DI Setup	t_{DIS}	SR	10		ns	
DI Hold	t_{DIH}	SR	0		ns	
Address to Data Out	t_{ADO}	DC	3 t_{OSC}		ns	
Output Delay	t_{OD}	DC		25	ns	
Data Float after $\overline{CS}$ high	t_{DF}	DC		25	ns	
Chip Select Hold	t_{CSH}	SR	1 t_{OSC}		ns	
Write to Clock	t_{WC}		0		ns	
$\overline{W}$ to $\overline{CS}$ high	t_{WCS}	SR	0		ns	
Address to Data In	t_{ADI}	DC	0		ns	

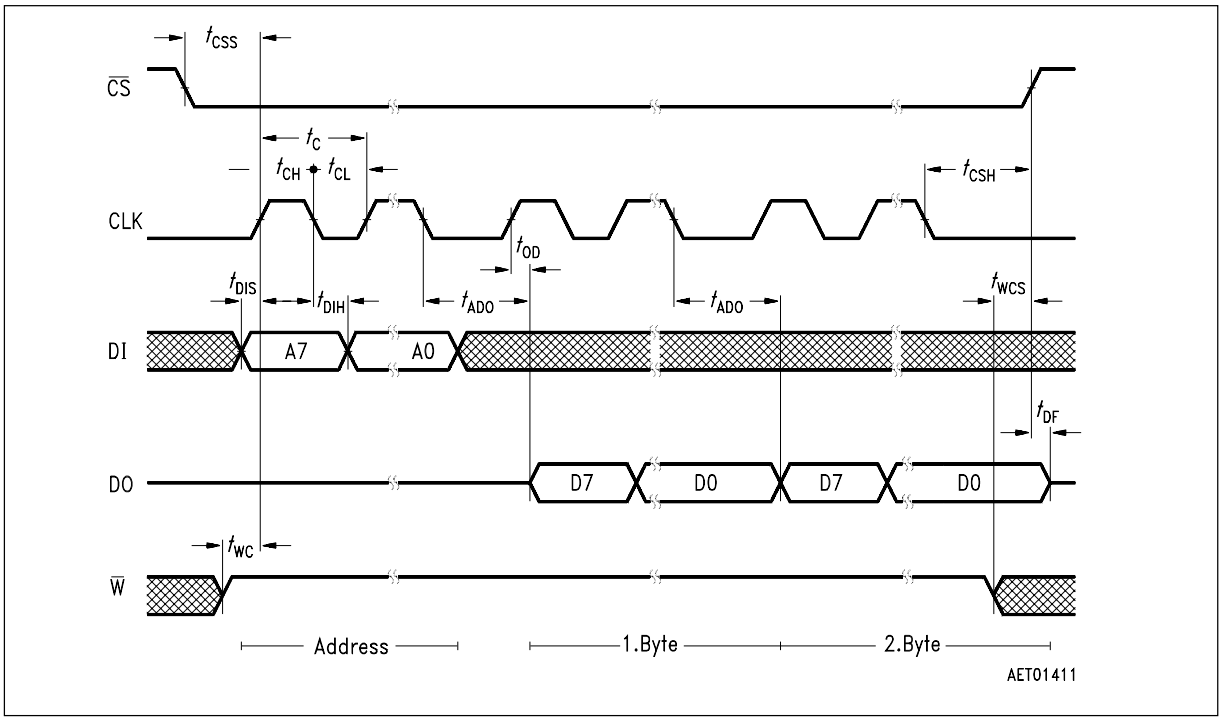


Figure 10
SI-Read-Timing (Timing A: Pin TIM = 0)

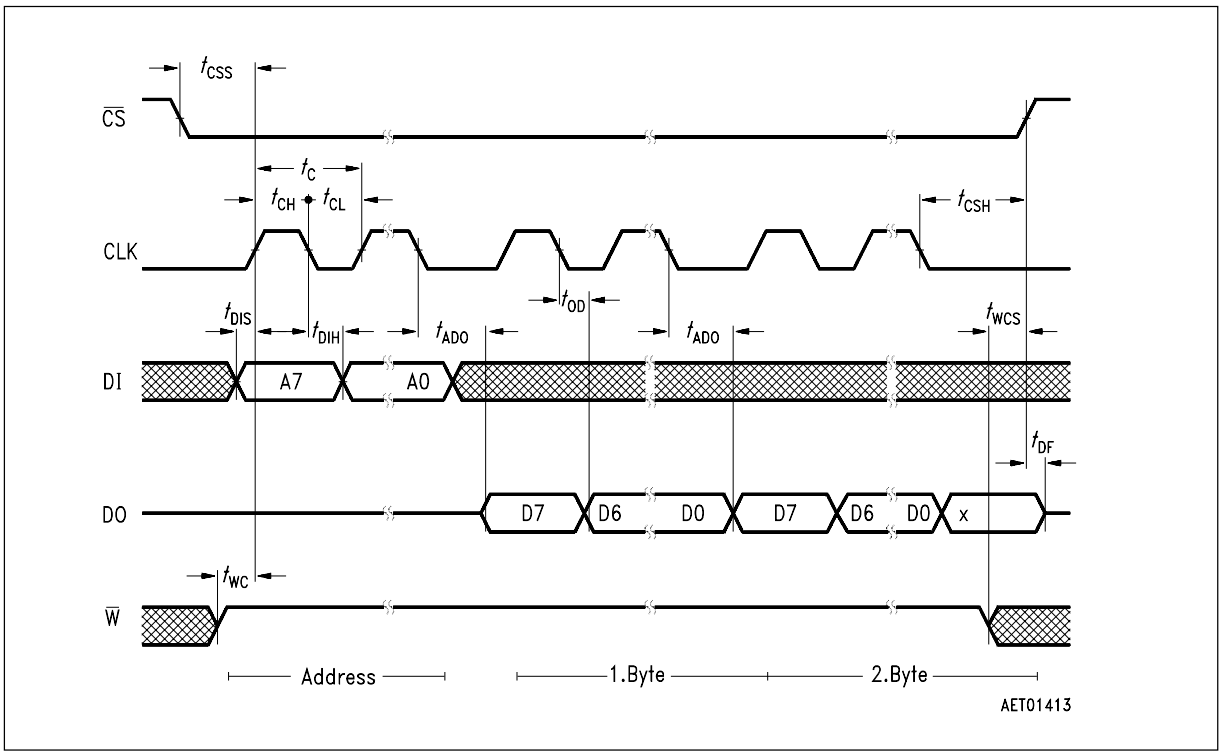


Figure 11
SI-Read-Timing (Timing B: Pin TIM =1)

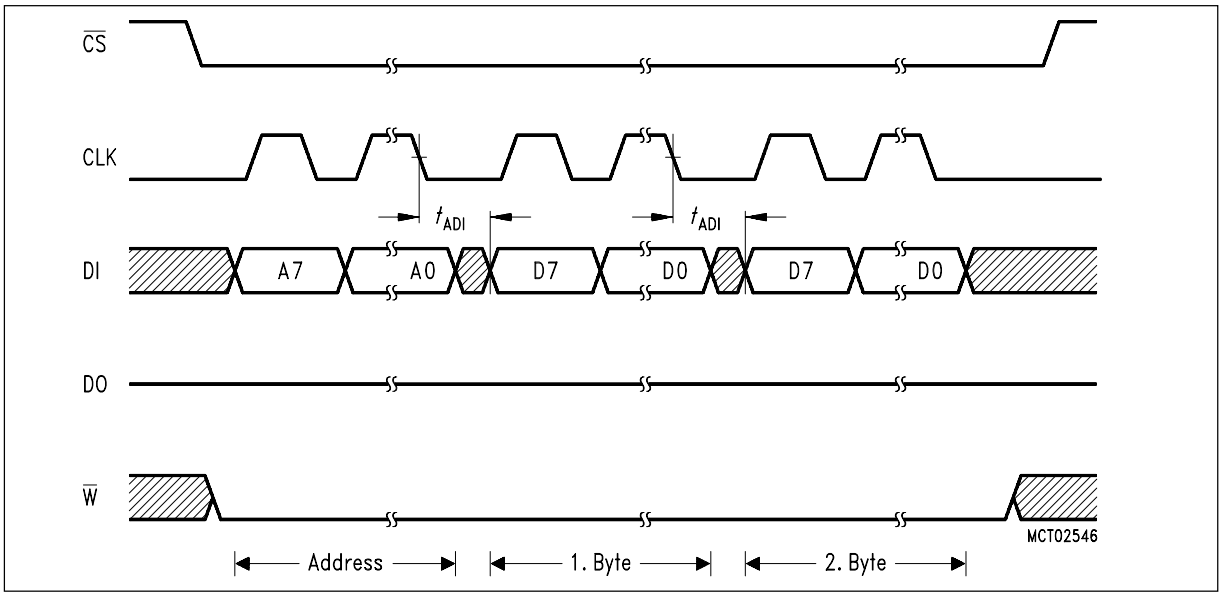


Figure 12
SI-Write-Timing

AC Characteristics (PI Timing)
V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = – 40 to + 110 °C; C_L = 50 pF

Parameter	Symbol		Limit Values		Unit	Test Condition
			min.	max.		
Read-Cycle time	t_{CYR}	DC	4 t_{OSC}		ns	
Write-Cycle time	t_{CYW}	DC	4 t_{OSC}		ns	
ALE pulse width	t_{LHLL}	DC	30		ns	
Address setup to ALE low	t_{AVLL}	SR	10		ns	
Address hold after ALE low	t_{LLAX}	SR	10		ns	
$\overline{RD}$ pulse width	t_{RLRH}	SR	2 t_{OSC} + 30		ns	
$\overline{WR}$ pulse width	t_{WLWH}	SR	2 t_{OSC} + 30		ns	
ALE low to $\overline{WR}$ active	t_{LLWL}	SR	20		ns	
ALE low to $\overline{RD}$ active	t_{LLRL}	SR	20		ns	
Data float after $\overline{RD}$ high	t_{RFDX}	DC	0	20	ns	
$\overline{RD}$ low to data valid	t_{RLDV}	DC		2 t_{OSC} + 20	ns	
Data setup before $\overline{WR}$ high	t_{DVWH}	SR	10		ns	
Data hold after $\overline{WR}$ high	t_{WHDX}	SR	5		ns	
$\overline{CS}$ low to $\overline{RD}$ low	t_{CLRL} /	SR	0		ns	
$\overline{CS}$ low to $\overline{WR}$ low	t_{CLWL}	SR	0		ns	
$\overline{WR}$ high to next ALE low	t_{WHLL}	SR	1.5 t_{OSC}		ns	

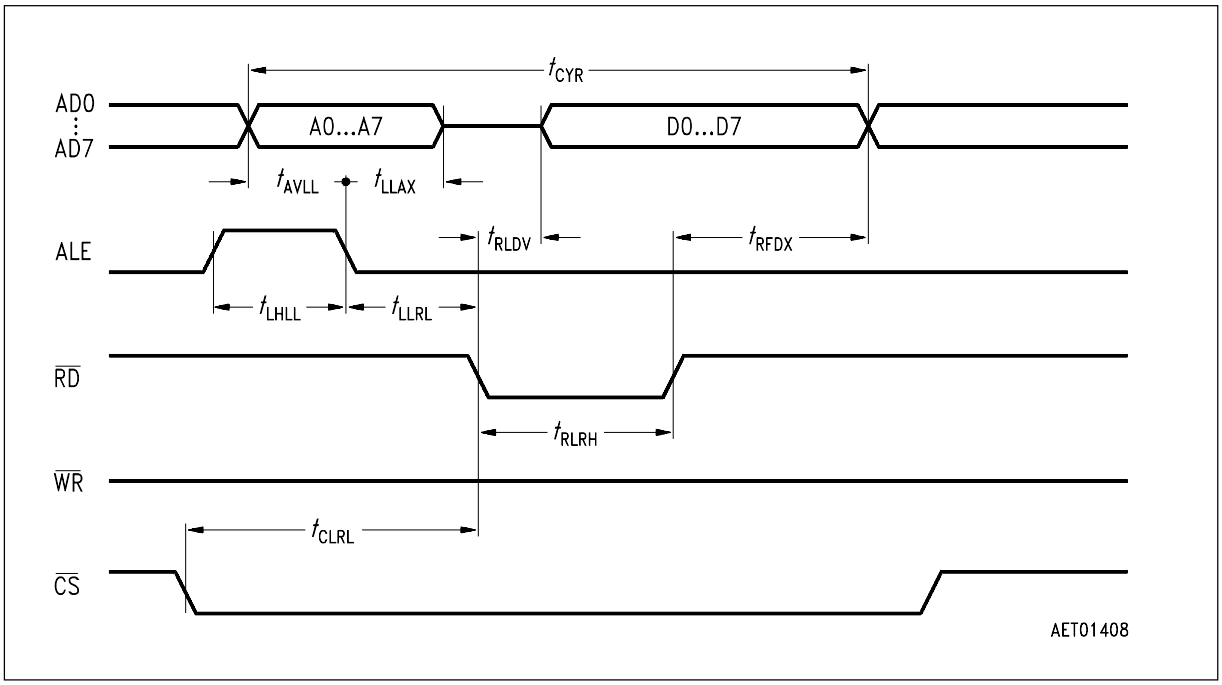


Figure 13
PI Timing: Read-Cycle-Timing

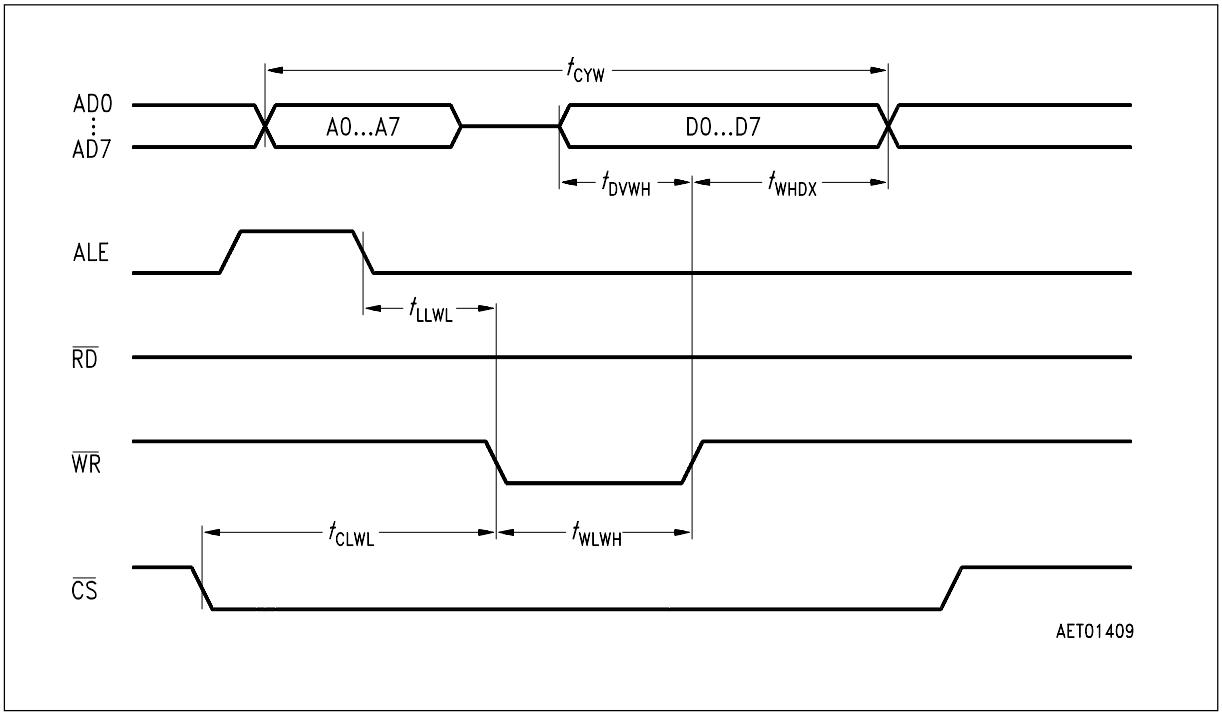


Figure 14
PI Timing: Write-Cycle-Timing