DATA SHEET

I·CODE
SL RC400

I•CODE Reader IC

Product Specification

Revision 2.0 Preliminary

November 2001

Philips
Semiconductors

PHILIPS

CONTENTS

1 GENERAL INFORMATION

1.1 Scope

This document describes the functionality of the SL RC400. It includes the functional and electrical specifications and gives details on how to design-in this device from system and hardware viewpoint.

1.2 General Description

The SL RC400 is member of a new family of highly integrated reader ICs for contactless communication at 13.56 MHz. This new reader IC family utilises an outstanding modulation and demodulation concept completely integrated for all kinds of passive contactless communication methods and protocols at 13.56 MHz.

The SL RC400 supports all layers of I•CODE1 and ISO 15693.

The internal transmitter part is able to drive an antenna designed for proximity operating distance (up to 100 mm) directly without additional active circuitry.

The receiver part provides a robust and efficient implementation of a demodulation and decoding circuitry for signals from I•CODE1 and ISO 15693 compatible transponders.

The digital part handles I•CODE1 and ISO 15693 framing and error detection (CRC).

A comfortable parallel interface which can be directly connected to any 8-bit µ-Processor gives high flexibility for the reader/terminal design.

1.3 Features

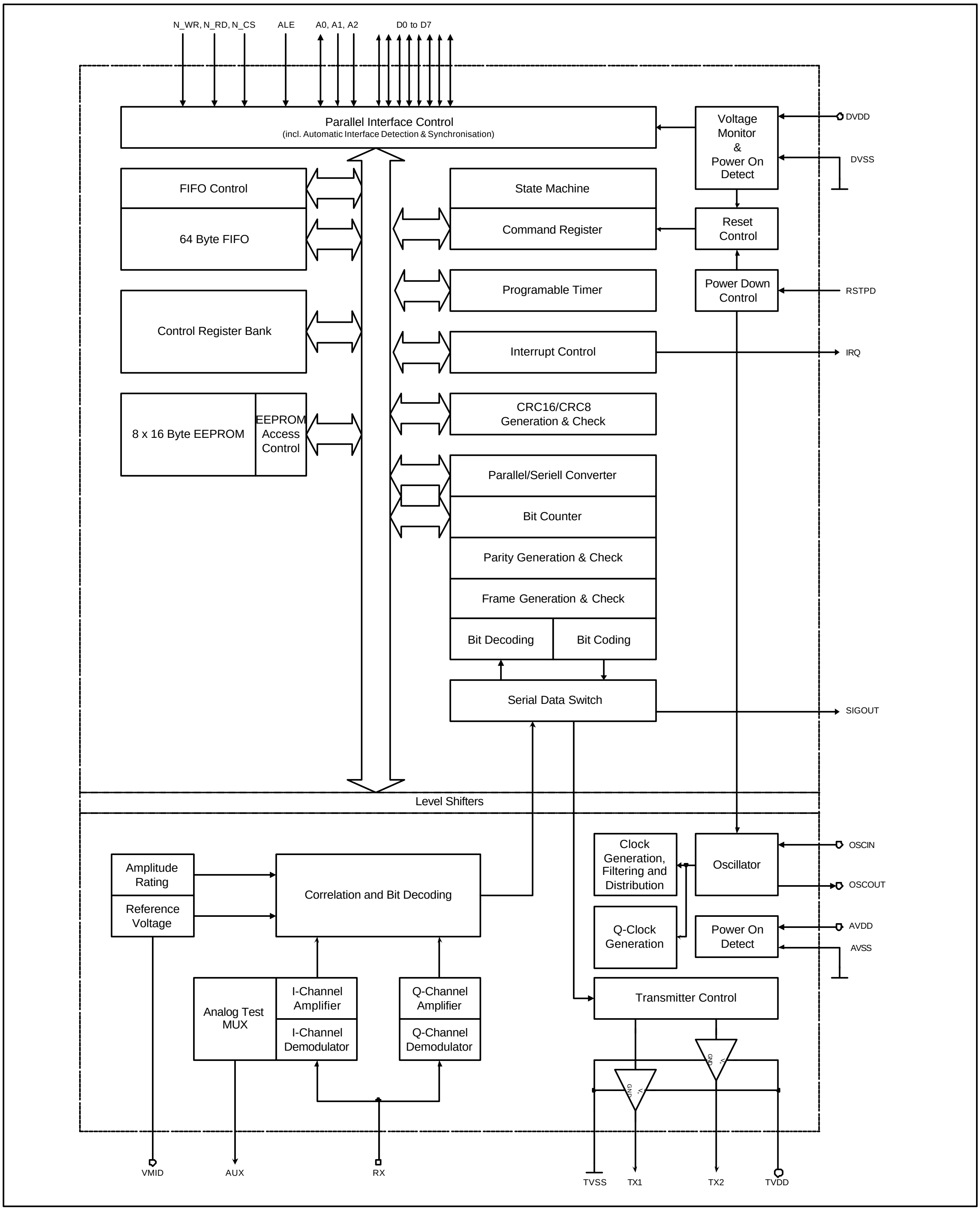
- Highly integrated analog circuitry to demodulate and decode label response
- Buffered output drivers to connect an antenna with minimum number of external components
- Proximity operating distance (up to 100 mm)
- Supports I•CODE1 and ISO 15693
- Parallel µ-Processor interface with internal address latch and IRQ line
- Flexible interrupt handling
- Automatic detection of parallel µC interface type
- Comfortable 64 byte send and receive FIFO-buffer
- Hard reset with low power function
- Power down mode per software
- Programmable timer
- Unique serial number
- User programmable start-up configuration
- Bit- and byte-oriented framing
- Independent power supply pins for digital, analog and transmitter part
- Internal oscillator buffer to connect 13.56 MHz quartz, optimised for low phase jitter
- Clock frequency filtering
- 3.3 V operation for transmitter (antenna driver) in short range applications

1.4 Ordering Information

Type Number	Package	
	Name	Description
SL RC400 01T	SO32	Small Outline Package; 32 leads

Table 1-1: SL RC400 Ordering Information

2 BLOCK DIAGRAM



3 PINNING INFORMATION

3.1 Pin Configuration

Pins denoted by bold letters are supplied by AVDD and AVSS. Pins drawn with bold lines are supplied by TVSS and TVDD. All other pins are supplied by DVDD and DVSS.

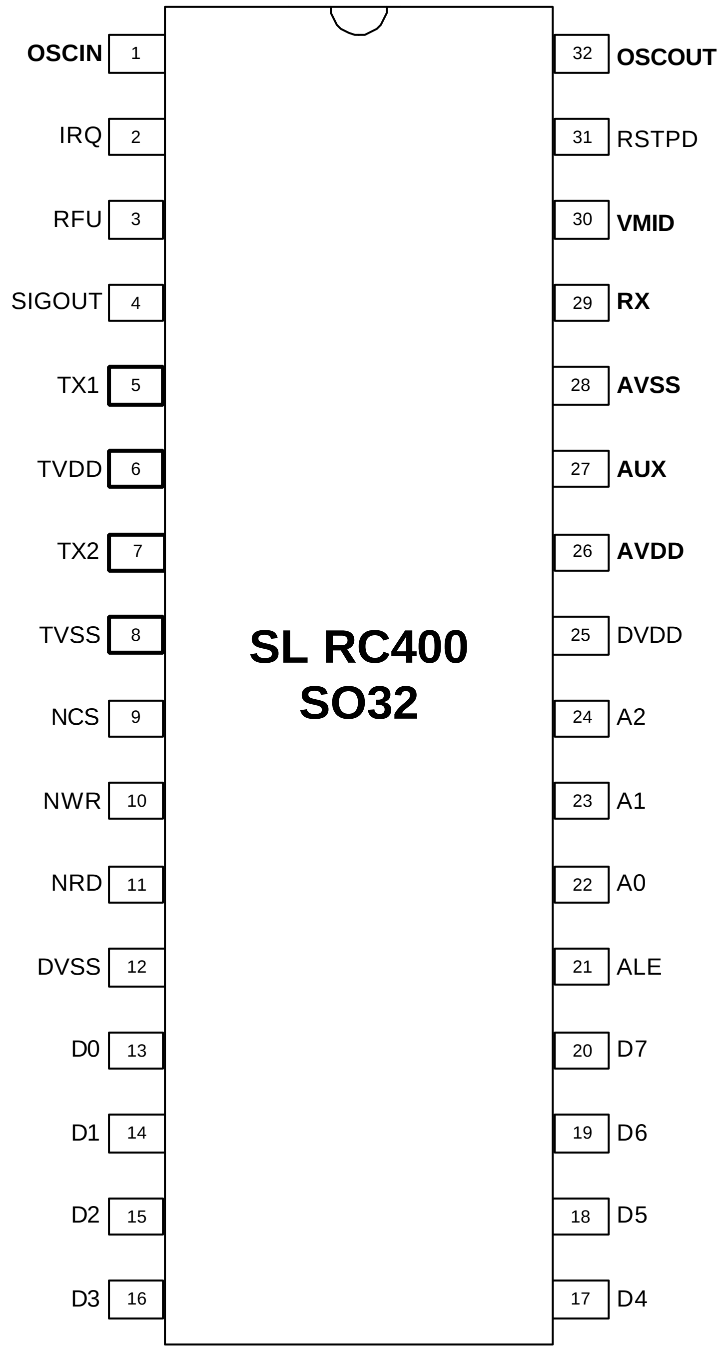


Figure 3-1: SL RC400 Pin Configuration for SO32 package

3.2 Pin Description

Pin Types: I...Input; O...Output; PWR...Power

PIN	SYMBOL	TYPE	DESCRIPTION
1	OSCIN	I	**Crystal Oscillator Input**: input to the inverting amplifier of the oscillator. This pin is also the input for an externally generated clock (f_{osc} = 13.56 MHz).
2	IRQ	O	**Interrupt Request**: output to signal an interrupt event
3	RFU	I	This Pin should be connected to Ground
4	SIGOUT	O	**I·CODE Interface Output**: delivers a serial data stream according to I•CODE1 and ISO 15693
5	TX1	O	**Transmitter 1**: delivers the modulated 13.56 MHz carrier frequenzy
6	TVDD	PWR	**Transmitter Power Supply**: supplies the output stage of TX1 and TX2
7	TX2	O	**Transmitter 2**: delivers the modulated 13.56 MHz carrier frequenzy
8	TVSS	PWR	**Transmitter Ground**: supplies the output stage of TX1 and TX2
9	NCS	I	**Not Chip Select**: selects and activates the µ-Processor interface of the SL RC400
10[1]	NWR	I	**Not Write**: strobe to write data (applied on D0 to D7) into the SL RC400 register
	R/NW	I	**Read Not Write**: selects if a read or write cycle shall be performed.
	nWrite	I	**Not Write**: selects if a read or write cycle shall be performed
11[1]	NRD	I	**Not Read**: strobe to read data from the SL RC400 register (applied on D0 to D7)
	NDS	I	**Not Data Strobe**: strobe for the read and the write cycle
	nDStrb	I	**Not Data Strobe**: strobe for the read and the write cycle
12	DVSS	PWR	**Digital Ground**
13 … 20[1]	D0 to D7	I/O	**8 Bit Bi-directional Data Bus**
	AD0 to AD7	I/O	**8 Bit Bi-directional Address and Data Bus**
21[1]	ALE	I	**Address Latch Enable**: strobe signal to latch AD0 to AD5 into the internal address latch when HIGH.
	AS	I	**Address Strobe**: strobe signal to latch AD0 to AD5 into the internal address latch when HIGH.
	nAStrb	I	**Not Address Strobe**: strobe signal to latch AD0 to AD5 into the internal address latch when LOW.
22[1]	A0	I	**Address Line 1**: Bit 0 of register address
	nWait	O	**Not Wait:** signals with LOW that an access-cycle may started and with HIGH that it may be finished.
23	A1	I	**Address Line 1**: Bit 1 of register address
24	A2	I	**Address Line 2**: Bit 2 of register address
25	DVDD	PWR	**Digital Power Supply**
26	AVDD	PWR	**Analog Power Supply**

[1] These pins offer different functionality according to the selected µ-Processor interface type. For detailed information refer to chapter 4.

PIN Description (continued)

PIN	SYMBOL	TYPE	DESCRIPTION
27	AUX	O	**Auxiliary Output**: This pin delivers analog test signals. The signal delivered on this output may be selected by means of the *TestAnaOutSel Register*.
28	AVSS	PWR	**Analog Ground**
29	RX	I	**Receiver Input**: Input pin for the labels response, which is the load modulated 13.56 MHz carrier frequenzy, that is coupled out from the antenna circuit.
30	VMID	PWR	**Internal Reference Voltage**: This pin delivers the internal reference voltage. Note: It has to be supported by means of a 100 nF block capacitor.
31	RSTPD	I	**Reset and Power Down**: When HIGH, internal current sinks are switched off, the oscillator is inhibited, and the input pads are disconnected from the outside world. With a negative edge on this pin the internal reset phase starts.
32	OSCOUT	O	**Crystal Oscillator Output**: Output of the inverting amplifier of the oscillator.

Table 3-1: SL RC400 Pin Description

4 PARALLEL INTERFACE

4.1 Overview of Supported µ-Processor Interfaces

The SL RC400 supports direct interfacing of various µ-Processor. Alternatively the Enhanced Parallel Port (EPP) of personal computers can be connected directly.

The following table shows the parallel interface signals supported by the SL RC400:

Bus Control Signals	Bus	Separated Address and Data Bus	Multiplexed Address and Data Bus
Separated Read and Write Strobes	**control**	NRD, NWR, NCS	NRD, NWR, NCS, ALE
	address	A0, A1, A2	AD0, AD1, AD2, (AD3, AD4, AD5)
	data	D0 … D7	AD0 … AD7
Common Read and Write Strobe	**control**	R/NW, NDS, NCS	R/NW, NDS, NCS, AS
	address	A0, A1, A2	AD0, AD1, AD2, (AD3, AD4, AD5)
	data	D0 … D7	AD0 … AD7
Common Read and Write Strobe with Handshake (EPP)	**control**	-	nWrite, nDStrb, NCS, nAStrb, nWait
	address		AD0, AD1, AD2, (AD3, AD4, AD5)
	data		AD0 … AD7

Table 4-1: Supported µ-Processor Interface Signals

4.2 Automatic µ-Processor Interface Type Detection

After each Power-On or Hard Reset, the SL RC400 also resets its parallel µ-Processor interface mode and checks the current µ-Processor interface type.
The SL RC400 identifies the µ-Processor interface by means of the logic levels on the control pins after the Reset Phase. This is done by a combination of fixed pin connections (see below) and a dedicated initialisation routine (see 11.4).

4.3 Connection to Different µ-Processor Types

The connection to different µ-Processor types is shown in the following table:

SL RC400	Parallel Interface Type				
	Separated Read/Write Strobe		Common Read/Write Strobe		
	Dedicated Address Bus	**Multiplexed Address Bus**	**Dedicated Address Bus**	**Multiplexed Address Bus**	**Multiplexed Address Bus with Handshake**
ALE	HIGH	ALE	HIGH	AS	nAStrb
A2	A2	LOW	A2	LOW	HIGH
A1	A1	HIGH	A1	HIGH	HIGH
A0	A0	HIGH	A0	LOW	nWait
NRD	NRD	NRD	NDS	NDS	nDStrb
NWR	NWR	NWR	R/NW	R/NW	nWrite
NCS	NCS	NCS	NCS	NCS	LOW
D7 ... D0	D7 ... D0	AD7 ... AD0	D7 ... D0	AD7 ... AD0	AD7 ... AD0

Table 4-2: Connection Scheme for Detecting the Parallel Interface Type

4.3.1 SEPARATED READ/WRITE STROBE: INTEL TYPE COMPATIBLE

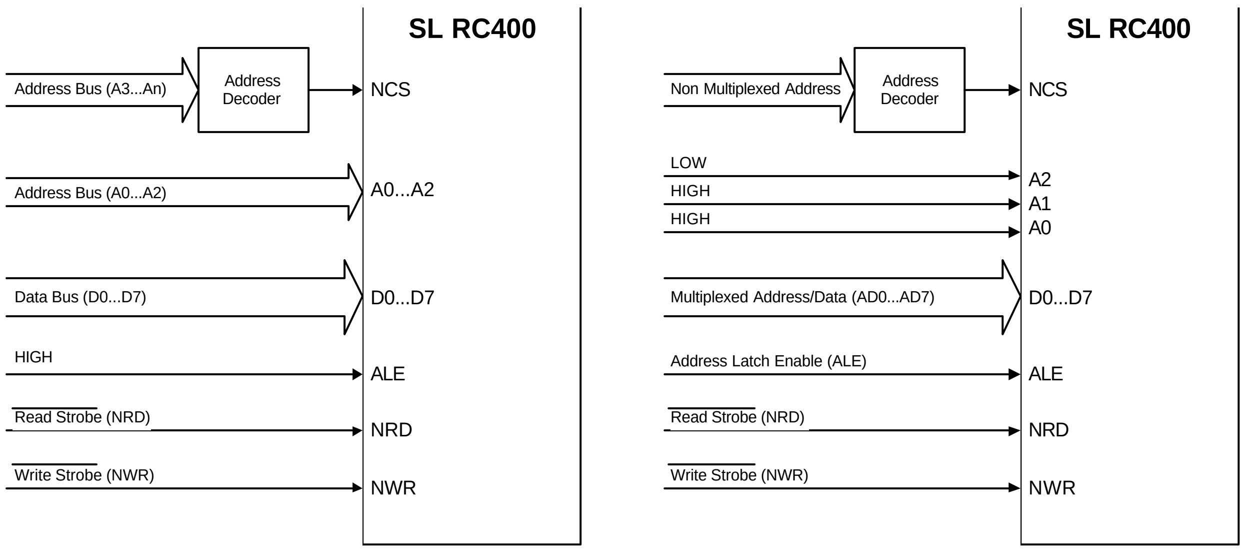


Figure 4-1: Connection to µ-Processors with Separated Read/Write Strobes

For timing specification refer to chapter 19.5.2.1.

4.3.2 COMMON READ/WRITE STROBE: MOTOROLA TYPE COMPATIBLE

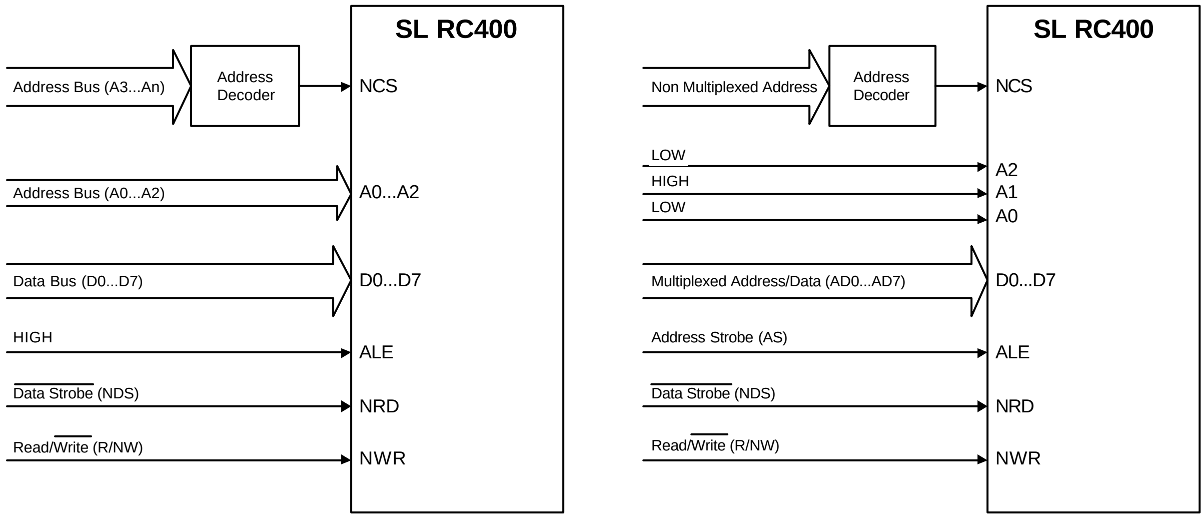


Figure 4-2: Connection to µ-Processors with Common Read/Write Strobes

For timing specification refer to chapter 19.5.2.2.

4.3.3 COMMON READ/WRITE STROBE AND HAND-SHAKE MECHANISM: EPP

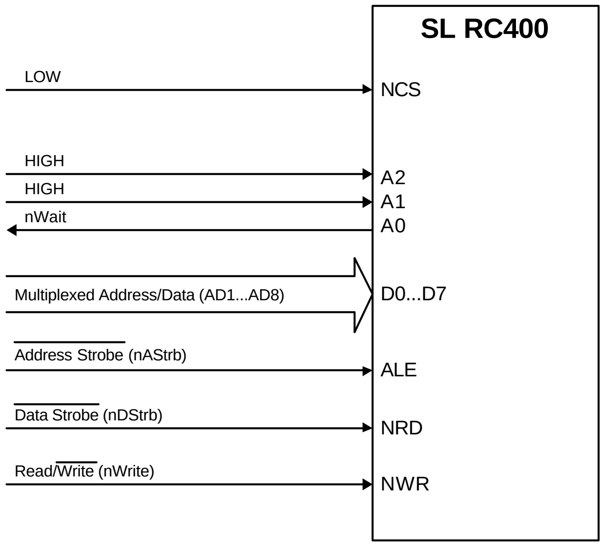


Figure 4-3: Connection to µ-Processors with Common Read/Write Strobes and Hand-Shake

For timing specification refer to chapter 19.5.2.3.

<u>Remarks for EPP:</u>
Although in the standard for the EPP no chip select signal is defined, the N_CS of the SL RC400 allows inhibiting the nDStrb signal. If not used, it shall be connected to DVSS.

After each Power-On or Hard Reset the nWait signal (delivered at pin A0) is high impedance. nWait will be defined at the first negative edge applied to nAStrb after the Reset Phase.

The SL RC400 does not support Read Address Cycle.

5 SL RC400 REGISTER SET

5.1 SL RC400 Registers Overview

Page	$Address_{hex}$	Register Name	Function
Page 0: Command and Status	0	Page	selects the register page
	1	Command	starts (and stops) the command execution
	2	FIFOData	in- and output of 64 byte FIFO buffer
	3	PrimaryStatus	status flags of the receiver and transmitter and of the FIFO buffer
	4	FIFOLength	number of bytes buffered in the FIFO
	5	SecondaryStatus	diverse status flags
	6	InterruptEn	control bits to enable and disable passing of interrupt requests
	7	InterruptRq	interrupt request flags
Page 1: Control and Status	8	Page	selects the register page
	9	Control	diverse control flags e.g.: timer, power saving
	A	ErrorFlag	error flags showing the error status of the last command executed
	B	Collpos	bit position of the first bit collision detected on the RF-interface
	C	TimerValue	actual value of the timer
	D	CRCResultLSB	LSB of the CRC-Coprocessor register
	E	CRCResultMSB	MSB of the CRC-Coprocessor register
	F	PreSet0F	these values shall not be changed
Page 2: Transmitter and Coder Control	10	Page	selects the register page
	11	TxControl	controls the logical behaviour of the antenna driver pins TX1 and TX2
	12	CwConductance	selects the conductance of the antenna driver pins TX1 and TX2
	13	ModConductance	selects the conductance of the antenna driver pins TX1 and TX2 during modulation
	14	CoderControl	Selects the bit coding mode and the framing during transmission
	15	ModWidth	selects the width of the modulation pulse
	16	ModWidthSOF	selects the width of the modulation pulse for SOF (I•CODE Fast-Mode)
	17	PreSet17	these values shall not be changed
Page 3: Receiver and Decoder Control	18	Page	selects the register page
	19	RxControl1	controls receiver behaviour
	1A	DecoderControl	controls decoder behaviour
	1B	BitPhase	selects the bit-phase between transmitter and receiver clock
	1C	RxThreshold	selects thresholds for the bit decoder
	1D	PreSet1D	these values shall not be changed
	1E	RxControl2	controls decoder behaviour and defines the input source for the receiver
	1F	ClockQControl	controls clock generation for the 90° phase shifted Q-channel clock

SL RC400 Register Set (continued)

Page	Address$_{hex}$	Register Name	Function
Page 4: RF-Timing and Channel Redundancy	20	Page	selects the register page
	21	RxWait	selects the time interval after transmission, before receiver starts
	22	ChannelRedundancy	selects the kind and mode of checking the data integrity on the RF-channel
	23	CRCPresetLSB	LSB of the pre-set value for the CRC register
	24	CRCPresetMSB	MSB of the pre-set value for the CRC register
	25	TimeSlotPeriod	selects the time between automatically mitted Frames see chapter 9.2.5
	26	SIGOUTSelect	selects internal signal applied to pin SIGOUT includes the MSB of value TimeSlotPeriod see register 0x25
	27	PreSet27	these values shall not be changed
Page 5: FIFO, Timer and IRQ-Pin Configuration	28	Page	Selects the register page
	29	FIFOLevel	defines level for FIFO over– and underflow warning
	2A	TimerClock	selects the divider for the timer clock
	2B	TimerControl	selects start and stop conditions for the timer
	2C	TimerReload	defines the pre-set value for the timer
	2D	IrqPinConfig	configures the output stage of pin IRq
	2E	PreSet2E	these values shall not be changed
	2F	PreSet2F	these values shall not be changed
Page 6: RFU	30	Page	selects the register page
	31	RFU	reserved for future use
	32	RFU	reserved for future use
	33	RFU	reserved for future use
	34	RFU	reserved for future use
	35	RFU	reserved for future use
	36	RFU	reserved for future use
	37	RFU	reserved for future use
Page 7: Test Control	38	Page	selects the register page
	39	RFU	reserved for future use
	3A	TestAnaSelect	selects analog test mode
	3B	PreSet3B	these values shall not be changed
	3C	PreSet3C	these values shall not be changed
	3D	TestDigiSelect	selects digital test mode
	3E	RFU	reserved for future use
	3F	RFU	reserved for future use

Table 5-1: SL RC400 Register Overview

5.1.1 REGISTER BIT BEHAVIOUR

Bits and flags for different registers behave differently, depending on their functions. In principle bits with same behaviour are grouped in common registers.

Abbreviation	Behaviour	Description
r/w	read and write	These bits can be written and read by the µ-Processor. Since they are used only for control means, there content is not influenced by internal state machines, e.g. the *TimerReload-Register* may be written and read by the µ-Processor. It will also be read by internal state machines, but never changed by them.
dy	dynamic	These bits can be written and read by the µ-Processor. Nevertheless, they may also be written automatically by internal state machines, e.g. the *Command-Register* changes its value automatically after the execution of the actual command.
r	read only	These registers hold flags, which value is determined by internal states only, e.g. the *ErrorFlag-Register* can not be written from external but shows internal states.
w	write only	These registers are used for control means only. They may be written by the µ-Processor but can not be read. Reading these registers returns an undefined value, e.g. the *TestAnaSelect-Register* is used to determine the signal on pin AUX, but it is not possible to read its content.

Table 5-2: Behaviour of Register Bits and its Designation

5.2 Register Description

5.2.1 PAGE 0: COMMAND AND STATUS

5.2.1.1 Page Register

Selects the register page.

Name: Page | Address: 0x00, 0x08, 0x10, 0x18, 0x20, 0x28, 0x30, 0x38 | Reset value: 10000000, 0x80

	7	6	5	4	3	2	1	0
	UsePage Select	0	0	0	0	PageSelect		
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7	UsePageSelect	If set to 1, the value of *PageSelect* is used as register address A5, A4, and A3. The LSBbits of the register address are defined by the address pins or the internal address latch, respectively. If set to 0, the whole content of the internal address latch defines the register address. The address pins are used as described in Table 4-2.
6-3	0000	Reserved for future use.
2-0	PageSelect	The value of *PageSelect* is used only if *UsePageSelect* is set to 1. In this case, it specifies the register page (which is A5, A4, and A3 of the register address).

5.2.1.2 *Command Register*

Starts and stops the command execution.

Name: Command Address: 0x01 Reset value:X0000000, 0xX0

	7	6	5	4	3	2	1	0
	IFDetect Busy	0	Command					
Access Rights	r	r	dy	dy	dy	dy	dy	dy

Description of the bits

Bit	Symbol	Function
7	IFDetectBusy	Shows the status of Interface Detection Logic: Set to 0 means ‘Interface Detection finished successfully’, Set to 1 signs ‘Interface Detection Ongoing’.
6	0	Reserved for future use.
5-0	Command	Activates a command according the Command Code. Reading this register shows, which command is actually executed. See chapter 16. SL RC400 Command Set.

5.2.1.3 *FIFOData Register*

In- and output of the 64 byte FIFO buffer

Name: FIFOData Address: 0x02 Reset value: XXXXXXXX, 0xXX

	7	6	5	4	3	2	1	0
	FIFOData							
Access Rights	dy	dy	dy	dy	dy	dy	dy	dy

Description of the bits

Bit	Symbol	Function
7-0	FIFOData	Data Input and Output Port for the internal 64 byte FIFO buffer. The FIFO buffer acts as parallel in/parallel out converter for all data stream in- and outputs.

5.2.1.4 PrimaryStatus Register

Status flags of the receiver, transmitter and the FIFO buffer.

Name: PrimaryStatus Address: 0x03 Reset value: 00000001, 0x01

	7	6	5	4	3	2	1	0
	0	ModemState			IRq	Err	HiAlert	LoAlert
Access Rights	r	r	r	r	r	r	r	r

Description of the bits

Bit	Symbol	Function
7	0	Reserved for future use.
6-4	ModemState	ModemState shows the state of the transmitter and receiver state machines. **State** / **Name of State** / **Description** 000 Idle: Neither the transmitter nor the receiver is in operation, since none of them is started or since none of them has got input data. 001 TxSOF: Transmitting the ‘Start Of Frame’ Pattern. 010 TxData: Transmitting data from the FIFO buffer (or redundancy check bits). 011 TxEOF: Transmitting the ‘End Of Frame’ Pattern. 100 GoToRx1: Mean-State passed, when receiver starts. GoToRx2: Mean-State passed, when receiver finishes. 101 PrepareRx: Waiting until the time period selected in the RxWait Register has expired. 110 AwaitingRx: Receiver activated; Awaiting an input signal at pin Rx. 111 Receiving: Receiving data.
3	IRq	This bit shows, if any interrupt source requests attention (with respect to the setting of the interrupt enable flags in the InterruptEn Register).
2	Err	This bit is set to 1, if any error flag in the ErrorFlag Register is set.
1	HiAlert	Is set to 1, when the number of bytes stored in the FIFO buffer fulfil the following equation: $HiAlert = (64 - FIFOLength) \leq WaterLevel$ Example: FIFOLength=60, WaterLevel=4 ⇒ HiAlert =1 FIFOLength=59, WaterLevel=4 ⇒ HiAlert =0
0	LoAlert	Is set to 1, when the number of bytes stored in the FIFO buffer fulfil the following equation: $LoAlert = FIFOLength \leq WaterLevel$ Example: FIFOLength=4, WaterLevel=4 ⇒ LoAlert =1 FIFOLength=5, WaterLevel=4 ⇒ LoAlert =0

5.2.1.5 FIFOLength Register

Number of bytes buffered in the FIFO.

Name: FIFOLength Address: 0x04 Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	FIFOLength						
Access Rights	r	r	r	r	r	r	r	r

Description of the bits

Bit	Symbol	Function
7	0	Reserved for future use.
6-0	FIFOLength	Indicates the number of bytes stored in the FIFO buffer. Writing to the *FIFOData* Register increments, reading decrements *FIFOLength.*

5.2.1.6 SecondaryStatus Register

Diverse Status flags.

Name: SecondaryStatus Address: 0x05 Reset value: 01100000, 0x60

	7	6	5	4	3	2	1	0
	TRunning	E2Ready	CRCReady	0	0	RxLastBits		
Access Rights	r	r	r	r	r	r	r	r

Description of the bits

Bit	Symbol	Function
7	TRunning	If set to 1, the SL RC400's timer unit is running, e.g. the counter will decrement the *Timer Value Register* with the next timer clock.
6	E2Ready	If set to 1, the SL RC 400 has finished programming the E^2PROM.
5	CRCReady	If set to 1, the SL RC400 has finished calculating the CRC.
4-3	00	Reserved for future use.
2-0	RxLastBits	Show the number of valid bits in the last received byte. If zero, the whole byte is valid.

5.2.1.7 InterrupEn Register

Control bits to enable and disable passing of interrupt requests.

Name: InterruptEn Address: 0x06 Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	SetIEn	0	TimerIEn	TxIEn	RxIEn	IdleIEn	HiAlertIEn	LoAlertIEn
Access Rights	w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7	SetIEn	Set to 1 *SetIEn* defines that the marked bits in the *InterruptEn Register* are set, Set to 0 clears the marked bits.
6	0	Reserved for future use.
5	TimerIEn	Allows the timer interrupt request (indicated by bit *TimerIRq*) to be propagated to pin IRQ. This bit can not be set or cleared directly but only by means of bit *SetIEn*.
4	TxIEn	Allows the transmitter interrupt request (indicated by bit *TxIRq*) to be propagated to pin IRQ. This bit can not be set or cleared directly but only by means of bit *SetIEn*.
3	RxIEn	Allows the receiver interrupt request (indicated by bit *RxIRq*) to be propagated to pin IRQ. This bit can not be set or cleared directly but only by means of bit *SetIEn*.
2	IdleIEn	Allows the idle interrupt request (indicated by bit *IdleIRq*) to be propagated to pin IRQ. This bit can not be set or cleared directly but only by means of bit *SetIEn*.
1	HiAlertIEn	Allows the high alert interrupt request (indicated by bit *HiAlertIRq*) to be propagated to pin IRQ. This bit can not be set or cleared directly but only by means of bit *SetIEn*.
0	LoAlertIEn	Allows the low alert interrupt request (indicated by bit *LoAlertIRq*) to be propagated to pin IRQ. This bit can not be set or cleared directly but only by means of bit *SetIEn*.

5.2.1.8 *InterruptRq Register*

Interrupt request flags.

Name: InterruptRq Address: 0x07 Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	SetIRq	0	TimerIRq	TxIRq	RxIRq	IdleIRq	HiAlertIRq	LoAlertIRq
Access Rights	w	r/w	dy	dy	dy	dy	dy	dy

Description of the bits

Bit	Symbol	Function
7	SetIRq	Set to 1 *SetIRq* defines that the marked bits in the *InterruptRq Register* are set. Set to 0 defines, that the marked bits in the *InterruptRq Register* are cleared.
6	0	Reserved for future use.
5	TimerIRq	Set to 1, when the timer decrements the *TimerValue Register* to zero.
4	TxIRq	Set to 1, when one of the following events occurs: *Transceive Command*: All data transmitted. *CalcCRC Command*: All data is processed. *WriteE2 Command*: All data is programmed.
3	RxIRq	This bit is set to 1, when the receiver terminates.
2	IdleIRq	This bit is set to 1, when a command terminates by itself e.g. when the *Command Register* changes its value from any command to the *Idle Command*. If an unknown command is started bit *IdleIRq* is set. Starting the *Idle Command* by the µ-Processor does not set bit *IdleIRq*.
1	HiAlertIRq	This bit is set to 1, when bit *HiAlert* is set. In opposite to *HiAlert*, *HiAlertIRq* stores this event and can only be reset by means of bit *SetIRq*.
0	LoAlertIRq	This bit is set to 1, when bit *LoAlert* is set. In opposite to *LoAlert*, *LoAlertIRq* stores this event and can only be reset by means of bit *SetIRq*.

5.2.2 PAGE 1: CONTROL AND STATUS

5.2.2.1 Page Register

Selects the register page. See 5.2.1.1 *Page* register.

5.2.2.2 Control Register

Diverse control flags, e.g. : timer, power saving

Name: Control Address: 0x09 Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	StandBy	PowerDown	0	TStopNow	TStartNow	FlushFIFO
Access Rights	r/w	r/w	dy	dy	dy	w	w	w

Description of the bits

Bit	Symbol	Function
7-6	00	Reserved for future use
5	StandBy	Setting this bit to 1 enters the Soft PowerDown Mode. This means, internal current consuming blocks switch off, the oscillator keeps running.
4	PowerDown	Setting this bit to 1 enters the Soft PowerDown Mode. This means, internal current consuming blocks switch off including the oscillator.
3	0	Reserved for future use
2	TStopNow	Setting this bit to 1 starts the timer immediately. Reading this bit will always return 0.
1	TStartNow	Setting this bit to 1 stops the timer immediately. Reading this bit will always return 0.
0	FlushFIFO	Setting this bit to 1clears the internal FIFO-buffer's read- and write-pointer (*FIFOLength* becomes 0) and the flag *FIFOOvfl* immediately. Reading this bit will always return 0.

5.2.2.3 *ErrorFlag Register*

Error flags showing the error status of the last executed command.

Name: ErrorFlag Address: 0x0A Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	AccessErr	FIFOOvfl	CRCErr	FramingErr	0	CollErr
Access Rights	r	r	r	r	r	r	r	r

Description of the bits

Bit	**Symbol**	**Function**
7-6	0	Reserved for future use.
5	AccessErr	This bit is set to 1, if the access rights to the E²PROM are violated. This bit is set to 0 starting an E²PROM related command.
4	FIFOOvfl	This bit is set to 1, if the µ-Processor or a SL RC400's internal state machine (e.g. receiver) tries to write data into the FIFO buffer although the FIFO buffer is already full.
3	CRCErr	This bit is set to 1, if RxCRCEn is set and the CRC fails. It is cleared to 0 automatically at receiver start phase during the state PrepareRx.
2	FramingErr	This bit is set to 1, if the SOF is incorrect. It is cleared automatically at receiver start (that is during the state PrepareRx).
1	0	RFU
0	CollErr	This bit is set to 1, if a bit-collision is detected. It is cleared automatically at receiver start (that is during the state PrepareRx).

5.2.2.4 *CollPos Register*

Bit position of the first bit collision detected on the RF- interface.

Name: CollPos Address: 0x0B Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	CollPos							
Access Rights	r	r	r	r	r	r	r	r

Description of the bits

Bit	Symbol	Function
7-0	CollPos	This register shows the bit position of the first detected collision in a received frame. Example: 0x00 indicates a bit collision in the start bit 0x01 indicates a bit collision in the 1^{st} bit 0x08 indicates a bit collision in the 8^{th} bit

5.2.2.5 *TimerValue Register*

Actual value of the timer

Name: TimerValue Address:0x0C Reset value: XXXXXXXX, 0xXX

	7	6	5	4	3	2	1	0
	TimerValue							
Access Rights	r	r	r	r	r	r	r	r

Description of the bits

Bit	Symbol	Function
7-0	TimerValue	This register shows the actual value of the timer counter.

5.2.2.6 CRCResultLSB Register

LSB of the CRC-Coprocessor register.

Name: CRCResultLSB Address: 0x0D Reset value: XXXXXXXX, 0xXX

	7	6	5	4	3	2	1	0
	CRCResultLSB							
Access Rights	r	r	r	r	r	r	r	r

Description of the bits

Bit	Symbol	Function
7-0	CRCResultLSB	This register shows the actual value of the least significant byte of the CRC register. It is valid only if bit CRCReady is set to 1.

5.2.2.7 CRCResultMSB Register

MSB of the CRC-Coprocessor register.

Name: CRCResultMSB Address: 0x0E Reset value: XXXXXXXX, 0xXX

	7	6	5	4	3	2	1	0
	CRCResultMSB							
Access Rights	r	r	r	r	r	r	r	r

Description of the bits

Bit	Symbol	Function
7-0	CRCResultMSB	This register shows the actual value of the most significant byte of the CRC register. It is valid only if bit CRCReady is set to 1. For 8-bit CRC calculation the registers value is undefined.

5.2.2.8 PreSet0F Register

Name: PreSet0F Address: 0x0F Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Note: These values shall not be changed !

5.2.3 PAGE 2: TRANSMITTER AND CONTROL

5.2.3.1 Page Register

Selects the register page. See 5.2.1.1 Page register.

5.2.3.2 TxControl Register

Controls the logical behaviour of the antenna pin TX1 and TX2

Name: TxControl Address: 0x11 Reset value: 01001000, 0x48

	7	6	5	4	3	2	1	0
	0	ModulatorSource		Force100 ASK	TX2Inv	TX2Cw	TX2RFEn	TX1RFEn
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7	0	This value shall not be changed
6-5	Modulator Source	Selects the source for the modulator input: 00: LOW 01: HIGH 10: Internal Coder 11: RFU
4	Force100ASK	Set to 1, forces a 100% ASK Modulation independent of the setting in the ModConductance Register.
3	TX2Inv	Set to 1, the output signal on pin TX2 will deliver an inverted 13.56 MHz carrier frequenzy.
2	TX2Cw	Set to 1, the output signal on pin TX2 will deliver continuously the un-modulated 13.56 MHz carrier frequenzy. Setting *TX2Cw* to 0 enables modulation of the 13.56 MHz carrier frequenzy.
1	TX2RFEn	Set to 1, the output signal on pin TX2 will deliver the 13.56 MHz carrier frequency modulated by the transmission data. If *TX2RFEn* is 0, TX2 drives a constant output level. See chapter 13.
0	TX1RFEn	Set to 1, the output signal on pin TX1 will deliver the 13.56 MHz carrier frequency modulated by the transmission data. If *TX1RFEn* is 0, TX1 drives a constant output level. See chapter 13.

5.2.3.3 CwConductance Register

Selects the conductance of the antenna driver pins TX1 and TX2.

Name: CwConductance Address: 0x12 Reset value: 00111111, 0x3F

	7	6	5	4	3	2	1	0
	0	0	GsCfgCW					
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-6	00	These values shall not be changed
5-0	GsCfgCW	The value of this register defines the conductance of the output driver. This may be used to regulate the output power and subsequently current consumption and operating distance.

For detailed information about GsCfgCW see 13.2.1

5.2.3.4 ModConductance Register

Name: ModConductance Address: 0x13 Reset value: 00000101, 0x05

	7	6	5	4	3	2	1	0
	0	0	GsCfgMod					
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-6	00	These values shall not be changed
5-0	GsCfgMod	The value of this register defines the conductance of the output driver for the time of modulation. This may be used to regulate the modulation index.

For detailed information about GsCfgMod see 13.3

5.2.3.5 CoderControl Register

Name: CoderControl Address:0x14 Reset value: 00101100, 0x2C

	7	6	5	4	3	2	1	0
	SendOne Pulse	0	CoderRate			TxCoding		
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7	SendOnePulse	Set to 1, forces to generate only one Mudulation (for ISO 15693 only). This is used to switch to the next TimeSlot if the Inventory command is used. This bit is not cleared automatically, it has to be re-set to 0 by the user.
6	0	These values shall not be changed
5-3	CoderRate	This register defines the clock rate for Coder Circuit 000: RFU 001: RFU 010: RFU 011: RFU 100: RFU 101: For I?CODE1 standard mode and ISO 15693 (~52.97kHz) 110: For I?CODE1 fast mode (~26.48kHz) 111: RFU
2-0	TxCoding	This register defines the bit coding Mode and Framing during Transmission 000: RFU 001: RFU 010: RFU 011: RFU 100: For I?CODE1 standard mode (1 out of 256 coding) 101: For I?CODE1 fast mode (RZ coding) 110: For ISO 15693 standard mode (1 out of 256 coding) 111: For ISO 15693 fast mode (1 out of 4 coding)

5.2.3.6 *ModWidth Register*

selects the width of the modulation pulse.

Name: ModWidth Address: 0x15 Reset value: 00111111, 0x3F

	7	6	5	4	3	2	1	0
	ModWidth							
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-0	ModWidth	This register defines the width of the modulation pulse according to T_{mod} = 2·(ModWidth +1) / fc (fc = Oscillator clock 13.56 MHz). Preset for I•CODE1 (Fast and Standard Mode) and ISO 15693 is 0x3F (Modulation width: 9.44µs).

5.2.3.7 *ModWidthSOF Register*

Name: ModWidthSOF Address: 0x16 Reset value: 00111111, 0x3F

	7	6	5	4	3	2	1	0
	ModWidthSOF							
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-0	ModWidthSOF	This register defines the width of the modulation pulse for SOF T_{mod} = 2·(ModWidth +1) / fc . Register setting: I•CODE1 Standard Mode: 0x3F (Modulation width SOF: 9.44µs). I•CODE1 Fast Mode: 0x73 (Modulation width SOF: 18.88µs). ISO 15693: 0x3F (Modulation width SOF: 9.44µs).

5.2.3.8 *PreSet17 Register*

Name: PreSet17 Address: 0x17 Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Note: These values shall not be changed !

5.2.4 PAGE 3: RECEIVER AND DECODER CONTROL

5.2.4.1 *Page Register*

Selects the register page. See 5.2.1.1 Page Register.

5.2.4.2 *RxControl1 Register*

controls receiver behaviour.

Name: RxControl1 Address: 0x19 Reset value: 10001011, 0x8B

	7	6	5	4	3	2	1	0
	1	0	0	0	1	0	Gain	
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-2	100010	These values shall not be changed
1-0	Gain	This register defines the receivers signal voltage gain factor: 00: 27 dB 01: 30 dB 10: 38 dB 11: 42 dB

5.2.4.3 DecoderControl Register

controls decoder behaviour.

Name: DecoderControl Address: 0x1A Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	Rx Multiple	ZeroAfter Coll	RxFraming		RxInvert	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7	0	These values shall not be changed
6	RxMultiple	If set to 0, the receiver is deactivated after receiving the Datastream. If set to 1, it is possible to receive more than one Frame.
5	ZeroAfterColl	If set to 1, any bits received after a bit-collision are masked to zero. This eases resolving the anti-collision procedure defined in the standard ISO 15693.
4-3	RxFraming	Selects the receiving frame type 00 for I•CODE1 01 RFU 10 ISO 15693 11 RFU
2	RxInvert	If set to 0, a modulation at the first half bit results a logic 1 (according I•CODE1) If set to 1, a modulation at the first half bit results a logic 0 (according ISO15693)
1-0	00	These values shall not be changed

5.2.4.4 BitPhase Register

selects the bit-phase between transmitter and receiver clock.

Name: BitPhase Address: 0x1B Reset value: 01010100, 0x54

	7	6	5	4	3	2	1	0
	BitPhase							
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-0	BitPase	Defines the phase relation between transmitter and receiver clock. Note: The correct value of this register is essential for proper operation.

5.2.4.5 *RxThreshold Register*

selects thresholds for the bit decoder.

Name: RxThreshold Address: 0x1C Reset value: 01101000, 0x68

	7	6	5	4	3	2	1	0
	MinLevel				CollLevel			
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-4	MinLevel	Defines the minimum signal strength at the decoder input that shall be accepted. If the signal strength is below this level, it is not evaluated.
3-0	CollLevel	Defines the minimum signal strength at the decoder input that has to be reached by the weaker half-bit of the Manchester-coded signal to generate a bit-collision relatively to the amplitude of the stronger half-bit.

5.2.4.6 *PreSet1D Register*

Name: PreSet1D Address: 0x1D Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Note: These values shall not be changed !

5.2.4.7 *RxControl2 Register*

controls decoder behaviour and defines the input source for the receiver.

Name:RxControl2 Address: 0x1E Reset value: 01000001, 0x41

	7	6	5	4	3	2	1	0
	RcvClkSelI	RxAutoPD	0	0	0	0	DecoderSource	
Access Rights	R/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	**Symbol**	**Function**
7	RcvClkSelI	If set to 1, the I-clock is used for the receiver clock. 0 indicates, the Q-clock is used. I-clock and Q-clock are 90° phase shifted to each other
6	RxAutoPD	If set to 1, the receiver circuit is automatically switched on before receiving and switched off afterwards. This may be used to reduce current consumption. If set to 0, the receiver is always activated.
5-2	0000	These values shall not be changed
1-0	DecoderSource	Selects the source for the decoder input: 00: Low 01: Internal Demodulator 10: RFU 11: RFU

5.2.4.8 *ClockQControl Register*

controls clock generation for the 90° phase shifted Q-channel clock.

Name: ClockQControl Address: 0x1F Reset value: 000XXXXX, 0xXX

	7	6	5	4	3	2	1	0
	ClkQ180Deg	ClkQCalib	0	ClkQDelay				
Access Rights	r	r/w	r/w	dy	dy	dy	dy	dy

Description of the bits

Bit	Symbol	Function
7	ClkQ180Deg	If the Q-clock is phase shifted more than 180° compared to the I-clock, this bit is set to 1, otherwise it is 0.
6	ClkQCalib	If this bit is 0, the Q-clock is calibrated automatically after the Reset Phase and after data reception from the label. If this bit is set to 1, no calibration is performed automatically.
5	0	This value shall not be changed
4-0	ClkQDelay	This register shows the number of delay elements actually used to generate a 90° phase shift of the I-clock to obtain the Q-clock. It can be written directly by the µ-Processor or by the automatic calibration cycle.

5.2.5 PAGE 4: RF-TIMING AND CHANNEL REDUNDANCY

5.2.5.1 Page Register

Selects the register page. See 5.2.1.1 Page register.

5.2.5.2 RxWait Register

Selects the time interval after transmission, before receiver starts.

Name: RxWait Address: 0x21 Reset value: 00001000, 0x08

	7	6	5	4	3	2	1	0
	RxWait							
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-0	RxWait	After data transmission, the activation of the receiver is delayed for *RxWait* bit-clocks (proportional to CoderRate). During this ‘frame guard time’ any signal at pin Rx is ignored.

5.2.5.3 ChannelRedundancy Register

Selects kind and mode of checking the data integrity on the RF-channel.

Name: ChannelRedundancy Address: 0x22 Reset value: 00001100, 0x0C

	7	6	5	4	3	2	1	0
	0	CRCMSB First	CRC 3309	CRC8	RxCRCEn	TxCRCEn	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	**Symbol**	**Function**
7	0	This value shall not be changed
6	CRCMSBFirst	If set to 1, CRC-calculation shifts the MSBit into the CRC-Coprocessor first. If set to 0, CRC-calculation starts with the LSBit. Note: For usage according ISO 15693 and I?CODE1 this bit has to be 0.
5	CRC 3309	If set to 1, CRC-calculation is done according ISO/IEC3309 as it is defined in ISO 15693. Note: For usage according to I•CODE1 this bit has to be 0.
4	CRC8	If set to 1, an 8-bit CRC is calculated. If set to 0, a 16-bit CRC is calculated.
3	RxCRCEn	If set to 1, the last byte(s) of a received frame is/are interpreted as CRC byte/s. If the CRC itself is correct the CRC byte(s) is/are not passed to the FIFO. In case of an error, the *CRCErr* flag is set. If set to 0, no CRC is expected.
2	TxCRCEn	If set to 1, a CRC is calculated over the transmitted data and the CRC byte(s) are appended to the data stream. If set to 0, no CRC is transmitted.
1-0	00	RFU

5.2.5.4 *CRCPresetLSB Register*

LSB of the preset value for the CRC register.

Name: CRCPresetLSB Address: 0x23 Reset value: 11111110, 0xFE

	7	6	5	4	3	2	1	0
	CRCPresetLSB							
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-0	CRCPresetLSB	*CRCPresetLSB* defines the starting value for CRC-calculation. This value is loaded into the CRC at the beginning of transmission, reception and the CalcCRC Command, if the CRC calculation is enabled. The Preset value is set for I?CODE1 To use the ISO 15693 functionality the *CRCPresetLSB* Register has to be set to 0xFF.

5.2.5.5 *CRCPresetMSB Register*

MSB of the preset value for the CRC register.

Name: CRCPresetMSB Address: 0x24 Reset value: 11111111, 0xFF

	7	6	5	4	3	2	1	0
	CRCPresetMSB							
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-0	CRCPresetMSB	*CRCPresetMSB* defines the starting value for CRC-calculation. This value is loaded into the CRC at the beginning of transmission, reception and the CalcCRC Command, if the CRC calculation is enabled. Note: The Preset value of *CRCPresetMSB* Register is the same for I?CODE1 and ISO 15693. Note: This register is not relevant, if CRC8 is 1.

5.2.5.6 *TimeSlotPeriod Register*

Name: TimeSlotPeriod Address: 0x25 Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	TimeSlotPeriod							
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-0	TimeSlotPeriod	*TimeSlotPeriod* defines the time between automatically mitted Frames. To send a Quit-Frame according to the I•CODE1 protocol, it is necessary to have a relation to the beginning of the Command-Frame. The TimeSlotPeriod will start at the End of the Command transmission. For detailed information see also chapter 9.2.5

5.2.5.7 *SIGOUTSelect Register*

Selects internal signal applied to pin SIGOUT.

Name: SIGOUTSelect Address: 0x26 Reset value:00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	TimeSlot Period MSB	0	SIGOUTSelect		
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-5	000	These values shall not be changed
4	TimeSlotPeriod MSB	MSB of value TimeSlotPeriod see register 0x25
3	0	These values shall not be changed
2-0	SIGOUTSelect	*SIGOUTSelect* defines which signal is routed to pin SIGOUT. 000 Constant Low 001 Constant High 010 Modulation Signal (envelope) from internal coder, actual used coded 011 Serial data stream 100 Output signal of the carrier frequency demodulator (label modulation signal) 101 Output signal of the subcarrier demodulator (Manchester coded label signal) 110 RFU 111 RFU

5.2.5.8 PreSet27 Register

Name: PreSet27 | Address: 0x27 | Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Note: These values shall not be changed !

5.2.6 PAGE 5: FIFO, TIMER AND IRQ- PIN CONFIGURATION

5.2.6.1 Page Register

Selects the register page. See 5.2.1.1 Page register.

5.2.6.2 FIFOLevel Register

Defines the level for FIFO under- and overflow warning.

Name: FIFOLevel | Address: 0x29 | Reset value:00111110, 0x3E

	7	6	5	4	3	2	1	0
	0	0	WaterLevel					
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-6	00	These values shall not be changed
5-0	WaterLevel	This register defines, the warning level of the SL RC400 for the µ-Processor for a FIFO-buffer over- or underflow: *HiAlert* is set to 1, if the remaining FIFO-buffer space is equal or less than *WaterLevel* bytes in the FIFO-buffer. *LoAlert* is set to 1, if equal or less than *WaterLevel* bytes are in the FIFO-buffer,.

5.2.6.3 *TimerClock Register*

Selects the devider for the timer clock.

Name: TimerClock Address: 0x2A Reset value: 00001011, 0x0B

	7	6	5	4	3	2	1	0
	0	0	TAutoRestart	TPreScaler				
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	**Symbol**	**Function**
7-6	00	These values shall not be changed
5	TAutoRestart	If set to 1, the timer automatically restart its count-down from *TReloadValue*, instead of counting down to zero. If set to 0 the timer decrements to zero and the bit *TimerIRq* is set to 1.
4-0	TPreScaler	Defines the timer clock f_{Timer}. *TPreScaler* can be adjusted from 0x00 up to 0x15. The following formula is used to calculate f_{Timer} : f_{Timer} = 13.56 MHz / $2^{TPreScaler}$.

5.2.6.4 TimerControl Register

Selects start and stop conditions for the timer.

Name: TimerControl Address: 0x2B Reset value: 00000010, 0x02

	7	6	5	4	3	2	1	0
	0	0	0	0	TStopRxEnd	TStopRxBegin	TStartTxEnd	TStartTxBegin
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-4	0000	These values shall not be changed
3	TStopRxEnd	If set to 1, the timer is stopped automatically when data reception ends. 0 indicates, that the timer is not influenced by this condition.
2	TStopRxBegin	If set to 1, the timer is stopped automatically, when the first valid bit is received. 0 indicates, that the timer is not influenced by this condition.
1	TStartTxEnd	If set to 1, the timer is started automatically when data transmission ends. If the timer is already running, it is restarted by loading *TReloadValue* into the timer. 0 indicates, that the timer is not influenced by this condition.
0	TStartTxBegin	If set to 1, the timer is started automatically when the first bit is transmitted. If the timer is already running, it is restarted by loading *TReloadValue* into the timer. 0 indicates, that the timer is not influenced by this condition.

5.2.6.5 TimerReload Register

Defines the preset value for the timer.

Name: TimerReload Address: 0x2C Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	TReloadValue							
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-0	TReloadValue	With a start event the timer loads with the *TreloadValue*. Changing this register affects the timer only with the next start event. If *TReloadValue* is set to 0, the timer cannot start.

5.2.6.6 *IRQPinConfig Register*

Configures the output stage for pin IRQ.

Name: IRQPinConfig Address: 0x2D Reset value: 00000010, 0x02

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	IRQInv	IRQPushPull
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Description of the bits

Bit	Symbol	Function
7-2	000000	These values shall not be changed
1	IRQInv	If set to 1, the signal on pin IRQ is inverted with respect to bit *IRq*. 0 indicates, that the signal on pin IRQ is equal to bit *IRQ*.
0	IRQPushPull	If set to 1, pin IRQ works as standard CMOS output pad. 0 indicates, that pin IRQ works as open drain output pad.

5.2.6.7 *PreSet2E*

Name: PreSet2E Address: 0x2E Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Note: These values shall not be changed !

5.2.6.8 *Preset2F*

Name: Preset2F Address: 0x2F Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Note: These values shall not be changed !

Page 6: RFU

5.2.6.9 Page Register

Selects the register page. See 5.2.1.1 Page register.

5.2.6.10 RFU Registers

Name: RFU

Address: 0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 037

Reset value:00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Note: These registers are reserved for future use.

5.2.7 PAGE 7: TEST CONTROL

5.2.7.1 Page Register

Selects the register page. See 5.2.1.1 Page register.

5.2.7.2 RFU Register

Name: RFU Address: 0x39 Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	w	w	w	w	w	w	w	w

Note: These registers are reserved for future use.

5.2.7.3 *TestAnaSelect Register*

Selects analog test signals.

Name: TestAnaSelect Address: 0x3A Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	TestAnaOutSelect			
Access Rights	w	w	w	w	w	w	w	w

Description of the bits

Bit	Symbol	Function	
7-4	0000	These values shall not be changed	
3-0	TestAnaOutSel	This register selects the internal analog signal that is routed to pin AUX. For detailed information see 18.3	
		Value	Signal Name
		0	V_{mid}
		1	$V_{bandgap}$
		2	$V_{RxFollI}$
		3	$V_{RxFollQ}$
		4	V_{RxAmpI}
		5	V_{RxAmpQ}
		6	V_{CorrNI}
		7	V_{CorrNQ}
		8	V_{CorrDI}
		9	V_{CorrDQ}
		A	V_{EvalL}
		B	V_{EvalR}
		C	V_{Temp}
		D	RFU
		E	RFU
		F	RFU

5.2.7.4 PreSet3B

Name: PreSet3B Address: 0x3B Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	w	w	w	w	w	w	w	w

Note: These values shall not be changed !

5.2.7.5 PreSet3C

Name: PreSet3C Address: 0x3C Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	w	w	w	w	w	w	w	w

Note: These values shall not be changed !

5.2.7.6 TestDigiSelect Register

Selects digital test mode.

Name: TestDigiSelect Address:0x3D Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	SignalTo SIGOUT	TestDigiSignalSel						
Access Rights	w	w	w	w	w	w	w	w

Description of the bits

Bit	Symbol	Function	
7	SignalToSIGOUT	Set to 1, overrules the setting in *SIGOUTSelect* and the digital test signal defined in *TestDigiSignalSel* is routed to pin SIGOUT instead. Set to 0, *SIGOUTSelect* defines the signal delivered at pin SIGOUT.	
6-0	TestDigiSignalSel	Selects the digital test signal to be routed to pin SIGOUT.	
		For detailed information refer to chapter 18.4	
		TestDigiSelect	Signal Name
		74_{hex}	s_data
		64_{hex}	s_valid
		54_{hex}	s_coll
		44_{hex}	s_clock
		35_{hex}	rd_sync
		25_{hex}	wr_sync
		16_{hex}	int_clock

5.2.7.7 RFU Registers

Name: RFU Address: 0x3E, 0x3F Reset value: 00000000, 0x00

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
Access Rights	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

Note: These registers are reserved for future use.

5.3 SL RC400 Register Flags Overview

Flag(s)	Register	Address Register, Bit Position
AccessErr	ErrorFlag	0x0A, bit 5
BitPhase	BitPhase	0x1B, bits 7:0
ClkQ180Deg	ClockQControl	0x1F, bit 7
ClkQCalib	ClockQControl	0x1F, bit 6
ClkQDelay	ClockQControl	0x1F, bits 4:0
CollErr	ErrorFlag	0x0A, bit 0
CollLevel	RxThreshold	0x1C, bits 3:0
CollPos	CollPos	0x0B, bits 7:0
Command	Command	0x01, bits 5:0
CRC3309	ChannelRedundancy	0x22, bit 5
CRC8	ChannelRedundancy	0x22, bit 4
CRCErr	ErrorFlag	0x0A, bit 3
CRCMSBFirst	ChannelRedundancy	0x22, bit 6
CRCPresetLSB	CRCPresetLSB	0x23, bits 7:0
CRCPresetMSB	CRCPresetMSB	0x24, bits 7:0
CRCReady	SecondaryStatus	0x05 , bit 5
CRCResultMSB	CRCResultMSB	0x0E, bits 7:0
CRCResultLSB	CRCResultLSB	0x0D, , bits 7:0
DecoderSource	RxControl2	0x1E, bits 1:0
E2Ready	SecondaryStatus	0x05, bit 6
Err	PrimaryStatus	0x03, bit 2
FIFOData	FIFOData	0x02, bits 7:0
FIFOLength	FIFOLength	0x04, bits 7:0
FIFOOvfl	ErrorFlag	0x0A, bit 4
FlushFIFO	Control	0x09, bit 0
FramingErr	ErrorFlag	0x0A, bit 2
Gain	RxControl1	0x19, bits 1:0
GsCfgCW	CWConductance	0x12, bits 5:0
GsCfgMod	ModConductance	0x13, bits 5:0
HiAlert	PrimaryStatus	0x03, bit 1
HiAlertIEn	InterruptEn	0x06, bit 1
HiAlertIRq	InterruptRq	0x07, bit 1

Flag(s)	Register	Address Register, Bit Position
IdleIEn	InterruptEn	0x06, bit 2
IdleIRq	InterruptRq	0x07, bit 2
IFDetectBusy	Command	0x01, bit 7
IRq	PrimaryStatus	0x03, bit 3
IRQInv	IRQPinConfig	0x2D, bit 1
IRQPushPull	IRQPinConfig	0x2D, bit 0
LoAlert	PrimaryStatus	0x03, bit 0
LoAlertIEn	InterruptEn	0x06, bit 0
LoAlertIRq	InterruptRq	0x07, bit 0
SIGOUTSelect	SIGOUTSelect	0x26, bits 2:0
MinLevel	RxThreshold	0x1C, bits 7:4
ModemState	PrimaryStatus	0x03 , bit 6:4
ModulatorSource	TxControl	0x11, bits 6:5
ModWidth	ModWidth	0x15, bits /:0
PageSelect	Page	0x00, 0x08, 0x10, 0x18, 0x20, 0x28, 0x30, 0x38, bits 2:0
PowerDown	Control	0x09, bit4
RcvClkSelI	RxControl2	0x1E, bit 7
RxAutoPD	RxControl2	0x1E, bit 6
RxCRCEn	ChannelRedundancy	0x22, bit 3
RxIEn	InterruptEn	0x06, bit 3
RxIRq	InterruptRq	0x07, bit 3
RxLastBits	SecondaryStatus	0x05, bits 2:0
RxWait	RxWait	0x21, bits 7:0
SetIEn	InterruptEn	0x06, bit 67
SetIRq	InterruptRq	0x07, bit 7
SignalToSIGOUT	TestDigiSelect	0x3D, bit 7
StandBy	Control	0x09, bit 5
TAutoRestart	TimerClock	0x2A, bit 5
TestAnaOutSel	TestAnaSelect	0x3A, bits 6:4
TestDigiSignalSel	TestDigiSelect	0x3D, bit 6:0
TimerIEn	InterruptEn	0x06, bit 5
TimerIRq	InterruptRq	0x07, bit 5
TimerValue	TimerValue	0x0C, bits 7:0

Flag(s)	**Register**	**Address** Register, Bit Position
TPreScaler	TimerClock	0x2A, bits 4:0
TReloadValue	TimerReload	0x2C, bits 7:0
TRunning	SecondaryStatus	0x05, bit 7
TStartTxBegin	TimerControl	0x2B, bit 0
TStartTxEnd	TimerControl	0x2B, bit 1
TStartNow	Control	0x09, bit 1
TStopRxBegin	TimerControl	0x2B, bit 2
TStopRxEnd	TimerControl	0x2B, bit 3
TStopNow	Control	0x09, bit 2
TX1RFEn	TxControl	0x11, bit 0
TX2Cw	TxControl	0x11, bit 3
TX2Inv	TxControl	0x11, bit 3
TX2RFEn	TxControl	0x11, bit 1
TxCRCEn	ChannelRedundancy	0x22, bit 2
TxIEn	InterruptEn	0x06, bit 4
TxIRq	InterruptRq	0x07, bit 4
TxLastBits	BitFraming	0x0F, bits 2:0
UsePageSelect	Page	0x00, 0x08, 0x10, 0x18, 0x20, 0x28, 0x30, 0x38, bit 7
WaterLevel	FIFOLevel	0x29, bits 5:0
ZeroAfterColl	DecoderControl	0x1A, bit 5

5.4 Modes of Register Addressing

There are three mechanisms to operate the SL RC400:

- Initiating functions and controlling data manipulation by executing *commands*
- Configuring electrical and functional behaviour via a set of *configuration bits*
- Monitoring the state of the SL RC400 by reading *status flags*

The commands, configurations bits and flags are accessed via the µ-Processor interface.

The SL RC400 can internally address 64 registers. This basically requires six address lines.

5.4.1 PAGING MECHANISM

The SL RC400 register set is segmented into 8 pages with 8 register each. The *Page-Register* can always be addressed, no matter which page is currently selected.

5.4.2 DEDICATED ADDRESS BUS

Using the SL RC400 with dedicated address bus, the µ-Processor defines three address lines via the address pins A0, A1, and A2. This allows addressing within a page. To switch between registers in different pages the paging mechanism needs then to be used.

The following table shows how the register address is assembled:

Register Bit: *UsePageSelect*	**Register-Address**					
1	*PageSelect2*	*PageSelect1*	*PageSelect0*	A2	A1	A0

Table 5-3: Dedicated Address Bus: Assembling the Register Address

5.4.3 MULTIPLEXED ADDRESS BUS

Using the SL RC400 with multiplexed address bus, the µ-Processor may define all six address lines at once. In this case either the paging mechanism or linear addressing may be used.

The following table shows how the register address is assembled:

Interface Bus Type	**Register Bit: *UsePageSelect***	**Register-Address**					
Multiplexed Address Bus (paging mode)	1	*PageSelect2*	*PageSelect1*	*PageSelect0*	AD2	AD1	AD0
Multiplexed Address Bus (linear addressing)	0	AD5	AD4	AD3	AD2	AD1	AD0

Table 5-4: Multiplexed Address Bus: Assembling the Register Address

6 MEMORY ORGANISATION OF THE E²PROM

6.1 Diagram of the E²PROM Memory Organisation

Block Number	Block Address	Byte Addresses	Access Rights	Memory Content	See Also
0	0	00 … 0F	r	Product Information Field	6.2
1	1	10 … 1F	r/w	Start Up Register Initialisation File	6.3.1
2	2	20 … 2F	r/w		
3	3	30 … 3F	r/w	Register Initialisation File For User data or second Initialisation	6.3.3
4	4	40 … 4F	r/w		
5	5	50 … 5F	r/w		
6	6	60 … 6F	r/w		
7	7	70 … 7F	r/w		

Table 6-1:Diagram of E²PROM Memory Organisation

Note: It is strictly recommended to use only the described E²PROM address area.

6.2 Product Information Field (Read Only)

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Meaning	Product Type Identification					RFU			Product Serial Number				Internal		RsMaxP	CRC

Table 6-2: Product Information Field

PRODUCT TYPE IDENTIFICATION

The SL RC400 is a member of a new family for highly integrated reader IC's. Each member of the product family has its unique Product Type Identification. The value of the Product Type Identification is shown in the table below:

	Product Type Identification				
Byte	**0**	**1**	**2**	**3**	**4**
Value	30_{hex}	33_{hex}	$F1_{hex}$	00_{hex}	XX_{hex}

Table 6-3: Product Type Identification Definition

PRODUCT SERIAL NUMBER

The SL RC400 holds a four byte serial number that is unique for each device.

INTERNAL:

These 2 bytes hold internal trimming parameters.

MAXIMUM SOURCE RESISTANCE FOR THE P-CHANNEL DRIVER TRANSISTOR OF PIN TX1 AND TX2

The source resistance of the p-channel driver transistors of pin TX1 and TX2 may be adjusted via the *GsConfCW Register* (see chapter 13.2.1). The mean value of the maximum adjustable source resistance of the pins TX1 and TX2 is stored as an integer value in Ohms in byte RsMaxP.

This value is denoted as maximum adjustable source resistance $Rs_{ref,max,n}$ and is measured with *GsConfCW Register* set to 01_{hex}. It is in the range between about 80 to 120 O.

CRC

The content of the product information field is secured via a CRC-byte, which is checked during start up.

6.3 Register Initialisation Files (Read/Write)

Register initialisation in the register address range from 10_{hex} to $2F_{hex}$ is done automatically during the Initialising Phase (see 11.3), using the Start Up Register Initialisation File.
Furthermore, the user may initialise the SL RC400 registers with values from the Register Initialisation File executing the *LoadConfig-Command* (see 16.6.1).

Notes:

- The *Page-Register* (addressed with 10_{hex}, 18_{hex}, 20_{hex}, 28_{hex}) is skipped and not initialised.
- Make sure that all *PreSet* registers are not changed.
- Make sure, that all register bits that are reserved for future use (RFU) are set to 0.

6.3.1 START UP REGISTER INITIALISATION FILE (READ/WRITE)

The content of the E²PROM memory block address 1 and 2 are used to initialise the SL RC400 registers 10_{hex} to $2F_{hex}$ during the Initialising Phase automatically. The default values written into the E²PROM during production are shown in chapter 6.3.2.

The assignment is the following:

E²PROM Byte Address	Register Address	Remark
10_{hex} (Block 1, Byte 0)	10_{hex}	Skipped
11_{hex}	11_{hex}	Copied
…	…	…
$2F_{hex}$ (Block 2, Byte 15)	$2F_{hex}$	Copied

Table 6-4: Byte Assignment for Register Initialisation at Start Up

6.3.2 SHIPMENT CONTENT OF START UP REGISTER INITIALISATION FILE

During production test, the Start Up Register Initialisation File is initialised with the values shown in the table below. With each power up these values are written into the SL RC400 register during the Initialising Phase.

E²PROM Byte Address	Reg. Address	Value	Description
10	10	00	*Page*: free for user
11	11	58	*TxControl:* Transmitter pins TX1 and TX2 switched off, bridge driver configuration, modulator driven from internal digital circuitry
12	12	3F	*CwConductance:* Source resistance of TX1 and TX2 to minimum.
13	13	05	*ModGsCfg: Source resistance of TX1 and TX2 at the time of Modulation, to determine the modulation index*
14	14	2C	*CoderControl:* Selects the bit coding mode and the framing during transmission
15	15	3F	*ModWidth*: Pulse width for “used code (1 out of 256, RZ or 1 out of 4)” pulse coding is set to standard configuration.
16	16	3F	*ModWidthSOF* Pulse width of SOF
17	17	00	*PreSet17*
18	18	00	*Page:* free for user
19	19	8B	*RxControl1:* Amplifier gain is maximum.
1A	1A	00	*DecoderControl:* A bit-collision always evaluates to HIGH in the data bit stream.
1B	1B	54	*BitPhase: BitPhase* is set to standard configuration.
1C	1C	68	*RxThreshold: MinLevel* and *CollLevel* are set to maximum.
1D	1D	00	*PreSet1D*
1E	1E	41	*RxControl2:* Use Q-clock for the receiver, ‘Automatic Receiver Off’ is switched on, decoder is driven from internal analog circuitry.
1F	1F	00	*ClockQControl:* Automatic Q-clock Calibration’ is switched on.
20	20	00	*Page:* free for user
21	21	08	*RxWait*: Frame Guard Time is set to six bit clocks.
22	22	0C	*ChannelRedundancy:* Channel Redundancy is set according to I?CODE1.
23	23	FE	*CRCPresetLSB:* CRC-Preset value is set according to I?CODE1.
24	24	FF	*CRCPresetMSB:* CRC-Preset value is set according to I?CODE1.
25	25	00	*PreSet25*
26	26	00	*SIGOUTSelect:* Pin SIGOUT is set to LOW.
27	27	00	*PreSet27*
28	28	00	*Page:* free for user
29	29	3E	*FIFOLevel: WaterLevel*: FIFO buffer warning level is set to standard configuration.
2A	2A	0B	*TimerClock: TPreScaler* is set to standard configuration, timer unit restart function is switched off.
2B	2B	02	*TimerControl:* Timer is started at the end of transmission, stopped at the beginning of reception.
2C	2C	00	*TimerReload: TReloadValue:* the timer unit preset value is set to standard configuration
2D	2D	02	*IRQPinConfig:* Pin IRQ is set to high impedance.
2E	2E	00	*PreSet2E*
2F	2F	00	*PreSet2F*

Table 6-5: Shipment Content of Start Up Configuration File

6.3.3 REGISTER INITIALISATION FILE (READ/WRITE)

The content of the E²PROM memory from block address 3 to 7 may be used to initialise the SL RC400 registers 10_{hex} to $2F_{hex}$ by execution of the *LoadConfig-Command* (see 16.6.1). It requires a two byte argument, that is used as the two byte long E²PROM starting byte address for the initialisation procedure.

The assignment is the following:

E²PROM Byte Address	Register Address	Remark
Starting Byte address for the E²PROM	10_{hex}	Skipped
Starting Byte address for the E²PROM +1	11_{hex}	Copied
…	…	…
Starting Byte address for the E²PROM + 31	$2F_{hex}$	Copied

Table 6-6: Byte Assignment for Register Initialisation at Start Up

The Register Initialisation File is big enough to hold the values for two initialisation sets and leaves one more block (16 bytes) for the user.

Note: The Register Initialisation File is read- and write-able for the user. Therefore, these bytes may also be used to store user specific data for other purposes.

7 FIFO BUFFER

7.1 Overview

An 8x64 bit FIFO buffer is implemented in the SL RC400 acting as a parallel-to-parallel converter. It buffers the input and output data stream between the µ-Processor and the internals of the SL RC400. Thus, it is possible to handle data streams with lengths of up to 64 bytes without taking timing constraints into account.

7.2 Accessing the FIFO Buffer

7.2.1 ACCESS RULES

The FIFO-buffer input and output data bus is connected to the *FIFOData Register*. Writing to this register stores one byte in the FIFO-buffer and increments the internal FIFO-buffer write-pointer. Reading from this register shows the FIFO-buffer content stored at the FIFO-buffer read-pointer and increments the FIFO-buffer read-pointer. The distance between the write- and read-pointer can be obtained by reading the *FIFOLength Register*.

When the µ-Processor starts a command, the SL RC400 may, while the command is in progress, access the FIFO-buffer according to that command. Physically only one FIFO-buffer is implemented, which can be used in input- and output direction. Therefore the µ-Processor has to take care, not to access the FIFO-buffer in an unintended way.

The following table gives an overview on FIFO access during command processing:

Active Command	µ-Processor is allowed to		Remark
	Write to FIFO	Read from FIFO	
StartUp	-	-	
Idle	-	-	
Transmit	✓	-	
Receive	-	✓	
Transceive	✓	✓	µ-Processor has to know the actual state of the command (transmitting or receiving)
WriteE2	✓	-	
ReadE2	✓	✓	The µ-Processor has to prepare the arguments, then only reading is allowed
LoadConfig	✓	-	
CalcCRC	✓	-	

Table 7-1: Allowed Access to the FIFO-Buffer

7.3 Controlling the FIFO-Buffer

Besides writing and reading the FIFO-buffer, the FIFO-buffer pointers may be reset by setting the bit *FlushFIFO*. The consequence is, that *FIFOLength* becomes zero, *FIFOOvfl* is cleared, the actually stored bytes are not accessible anymore and the FIFO-buffer can be filled with another 64 bytes again.

7.4 Status Information about the FIFO-Buffer

The µ-Processor may obtain the following data about the FIFO-buffers status:

- Number of bytes already stored in the FIFO-buffer: *FIFOLength*
- Warning, that the FIFO-buffer is quite full: *HiAlert*
- Warning, that the FIFO-buffer is quite empty: *LoAlert*
- Indication, that bytes were written to the FIFO-buffer although it was already full: *FIFOOvfl*
  FIFOOvfl can be cleared only by setting bit *FlushFIFO*.

The SL RC400 can generate an interrupt signal

- If *LoAlertIRq* is set to 1 it will activate Pin IRQ when *LoAlert* changes to 1.
- If *HiAlertIRq* is set to 1 it will activate Pin IRQ when *HiAlert* changes to 1.

The flag *HiAlert* is set to 1 if only *WaterLevel* bytes or less can be stored in the FIFO-buffer. It is generated by the following equation:

$$HiAlert = (64 - FIFOLength) \leq WaterLevel$$

The flag *LoAlert* is set to 1 if *WaterLevel* bytes or less are actually stored in the FIFO-buffer. It is generated by the following equation:

$$LoAlert = FIFOLength \leq WaterLevel$$

7.5 Register overview FIFO Buffer

The following table shows the related flags of the FIFO buffer in alphabetic order.

Flags	**Register**	**Address** Register, bit position
FIFOLength	FIFOLength	0x04, bits 6-0
FIFOOvfl	ErrorFlag	0x0A, bit 4
FlushFIFO	Control	0x09, bit 0
HiAlert	PrimaryStatus	0x03, bit 1
HiAlertIEn	InterruptIEn	0x06, bit 1
HiAlertIRq	InterruptIRq	0x07, bit 1
LoAlert	PrimaryStatus	0x03, bit 0
LoAlertIEn	InterruptIEn	0x06, bit 0
LoAlertIRq	InterruptIRq	0x07, bit 0
WaterLevel	FIFOLevel	0x29, bits 5-0

Table 7-2. Registers associated with the FIFO Buffer

8 INTERRUPT REQUEST SYSTEM

8.1 Overview

The SL RC400 indicates certain events by setting bit *IRq* in the *PrimaryStatus-Register* and, in addition, by activating pin IRQ. The signal on pin IRQ may be used to interrupt the µ-Processor using its interrupt handling capabilities. This allows the implementation of efficient µ-Processor software.

8.1.1 INTERRUPT SOURCES OVERVIEW

The following table shows the integrated interrupt flags, the related source and the condition for its setting.
The interrupt flag *TimerIRq* indicates an interrupt set by the timer unit. The setting is done when the timer decrements from 1 either down to zero (*TAutoRestart flag disabled*) or to the TPreLoad value if TAutoRestart is enabled.
The *TxIRq* bit indicates interrupts from different sources. If the transmitter is active and the state changes from sending data to transmitting the end of frame pattern, the transmitter unit sets automatically the interrupt bit. The CRC coprocessor sets *TxIRq* after having processed all data from the FIFO buffer. This is indicated by the flag *CRCReady* = 1. If the E^2Prom programming has finished the *TxIRq* bit is set, indicated by the bit *E2Ready* = 1.
The *RxIRq* flag indicates an interrupt when the end of the received data is detected.
The flag *IdleIRq* is set if a command finishes and the content of the command register changes to idle.
The flag *HiAlertIRq* is set to 1 if the *HiAlert* bit is set to one, that means the FIFO buffer has reached the level indicated by the bit *WaterLevel*, see chapter 7.4.
The flag *LoAlertIRq* is set to 1 if the *LoAlert* bit is set to one, that means the FIFO buffer has reached the level indicated by the bit *WaterLevel*, see chapter 7.4.

Interrupt Flag	Interrupt Source	Is set automatically, when
TimerIRq	Timer Unit	the timer counts from 1 to 0
TxIRq	Transmitter	a data stream, transmitted to the label, ends
	CRC-Coprocessor	all data from the FIFO buffer has been processed
RxIRq	Receiver	a data stream, received from the label, ends
IdleIRq	*Command Register*	a command execution finishes
HiAlertIRq	FIFO-buffer	the FIFO-buffer is getting full
LoAlertIRq	FIFO-buffer	the FIFO-buffer is getting empty

Table 8-1: Interrupt Sources

8.2 Implementation of Interrupt Request Handling

8.2.1 CONTROLLING INTERRUPTS AND THEIR STATUS

The SL RC400 informs the µ-Processor about the interrupt request source by setting the according bit in the *InterruptRq Register*. The relevance of each interrupt request bit as source for an interrupt may be masked with the interrupt enable bits of the *InterruptEn Register*.

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
InterruptEn	SetIEn	RFU	TimerIEn	TxIEn	RxIEn	IdleIEn	HiAlertIEn	LoAlertIEn
InterruptRq	SetIRq	RFU	TimerIRq	TxIRq	RxIRq	IdleIRq	HiAlertIRq	LoAlertIRq

Table 8-2: Interrupt Control Registers

If any interrupt request flag is set to 1 (showing that an interrupt request is pending) and the corresponding interrupt enable flag is set the status flag *IRq* in the *PrimaryStatus Register* is set to 1. Furthermore, different interrupt sources can be set active simultaneously. Therefore, all interrupt request bits are ‘OR’ed and connected to the flag *IRq* and forwarded to pin IRQ.

8.2.2 ACCESSING THE INTERRUPT REGISTERS

The interrupt request bits are set automatically by the internal state machines of the SL RC400. Additionally the µ-Processor has access in order to set or to clear them.
A special implementation of the *InterruptRq* and the *InterruptEn* Register allows to change the status of a single bit without influencing the other ones. If a specific interrupt register shall be set to one, the bit *SetIxx* has to be set to 1 and simultaneously the specific bit has to be set to 1 too. Vice versa, if a specific interrupt flag shall be cleared, a zero has to be written to the *SetIxx* and simultaneously the specific address of the interrupt register has to be set to 1. If a bit content shall not be changed during the setting or clearing phase a zero has to be written to the specific bit location.

Example: writing $3F_{hex}$ to the *InterruptRq Register* clears all bits as *SetIRq* in this case is set to 0 and all other bits are set to 1. Writing 81_{hex} sets bit *LoAlertIRq* to 1 and leaves all other bits untouched.

8.3 Configuration of Pin IRQ

The logic level of the status flag *IRq* is visible at pin IRQ. In addition, the signal on pin IRQ may be controlled by the following bits of the *IRQPinConfig Register*:

- *IRQInv*: if set to 0, the signal on pin IRQ is equal to the logic level of bit *IRq*.
  If set to 1, the signal on pin IRQ is inverted with respect to bit *IRq*.
- *IRQPushPull*: if set to 1, pin IRQ has standard CMOS output characteristics
  otherwise it is an open drain output and an external resistor is necessary to achieve a HIGH level at this pin.

Note: During the Reset Phase (see 11.2) *IRQInv* is set 1 and *IRQPushPull* to 0. This results in a high impedance at pin IRQ.

8.4 Register Overview Interrupt Request System

The following table shows the related flags of the Interrupt Request System in alphabetic order.

Flags	**Register**	**Address** Register, bit position
HiAlertIEn	InterruptEn	0x06, bit 1
HiAlertIRq	InterruptRq	0x07, bit 1
IdleIEn	InterruptEn	0x06, bit 2
IdleIRq	InterruptRq	0x07, bit 2
IRq	PrimaryStatus	0x03, bit 3
IRQInv	IRQPinConfig	0x07, bit 1
IRQPushPull	IRQPinConfig	0x07, bit 0
LoAlertIEn	InterruptEn	0x06, bit 0
LoAlertIRq	InterruptRq	0x07, bit 0
RxIEn	InterruptEn	0x06, bit 3
RxIRq	InterruptRq	0x07, bit 3
SetIEn	InterruptEn	0x06, bit 7
SetIRq	InterruptRq	0x07, bit 7
TimerIEn	InterruptEn	0x06, bit 5
TimerIRq	InterruptRq	0x07, bit 5
TxIEn	InterruptEn	0x06, bit 4
TxIRq	InterruptRq	0x07, bit 4

Table 8-3 Registers associated with the Interrupt Request System

9 TIMER UNIT

9.1 Overview

A timer is implemented in the SL RC400. It derives its clock from the 13.56 MHz chip-clock. The µ-Processor may use this timer to manage timing relevant tasks.

The timer unit may be used in one of the following configurations:

- Timeout-Counter
- Watch-Dog Counter
- Stop Watch
- Programmable One-Shot
- Periodical Trigger

The timer unit can be used to measure the time interval between two events or to indicate that a specific event occurred after a specific time. The timer can be triggered by events which will be explained in the following, but the timer itself does not influence any internal event. A timeout during data receiving does not influence the receiving process automatically. Furthermore, several timer related flags are set and these flags can be used to generate an interrupt.

9.2 Implementation of the Timer Unit

9.2.1 BLOCK DIAGRAM

The following block diagram shows the timer module.

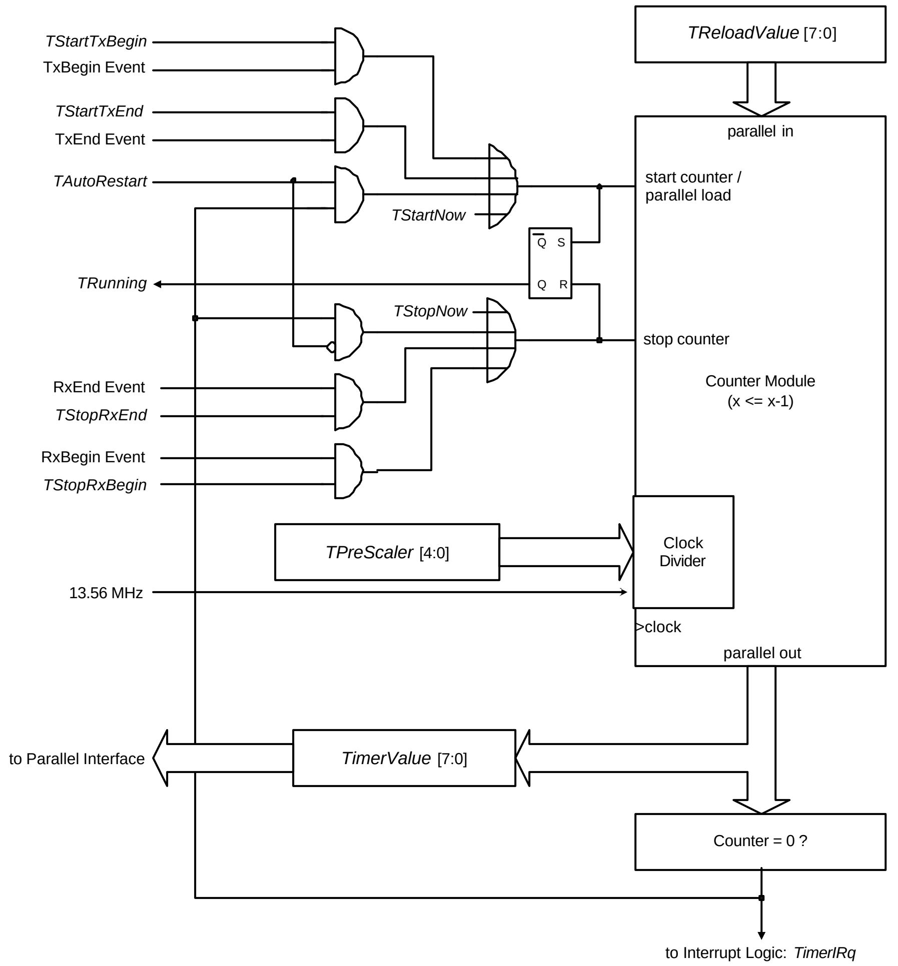


Figure 9-1: Timer Module Block Diagram

The timer unit is designed in a way, that several events in combination with enabling flags start or stop the counter. For example, setting the bit *TstartTxEnd* to 1 enables to control the receiving of data using the timer unit. In addition the first received bit is indicated by *TxEndEvent*. This combination starts the counter at the defined *TReloadValue*.
The timer stops either automatically if the counter value is equal to zero, or if a defined stop event happens (TautoRestart not enabled).

9.2.2 CONTROLLING THE TIMER UNIT

The main part of the timer unit is a down-counter. As long as the down-counter value is unequal zero, it decrements its value with each timer clock.
If *TAutoRestart* is enabled the timer does not decrement down to zero. Having reached the value 1 the timer reloads with the next clock with the *TimerReload value.*
The timer is started by loading a value from the *TimerReload Register* into the counter module. This may be triggered by one of the following events:

- Transmission of the first bit to the label (TxBegin Event) and bit *TStartTxBegin* is 1
- Transmission of the last bit to the label (TxEnd Event) and bit *TStartTxEnd* is 1
- The counter module decrements down to zero and bit *TAutoRestart* is 1
- Bit *TStartNow* is set to 1 (by the µ-Processor)

Note: Every start-event re-loads the timer from the *TimerReload Register*. Thus, the timer unit is re-triggered.

The timer can be configured to stop with one of the following events:

- Reception of the first valid bit from the label (RxBegin Event)and bit *TStopRxBegin* is set to 1
- Reception of the last bit from the label (RxEnd event) and bit *TStopRxEnd* is set to 1
- The counter module has decremented down to zero and bit *TAutoRestart* is set to 0
- Bit *TStopNow* is set to 1 (by the µ-Processor)

Loading a new value, e.g. zero, into the *TimerReload* Register does not immediately influence the counter, since the *TimerReload* Register affects the counter units content only with the next start-event. Thus, the *TimerReload* Register may be changed even if the timer unit is already counting. The consequence of changing the *TimerReload* Register will be visible after the next start-event.

If the counter is stopped by setting bit *TStopNow*, no *TimerIRq* is signalled.

9.2.3 TIMER UNIT CLOCK AND PERIOD

The clock of the timer unit is derived from the 13.56 MHz chip clock via a programmable divider. The clock selection is done with the *TPreScaler* Register, that defines the timer unit clock frequency according to the following formula:

$$T_{TimerClock} = \frac{1}{f_{TimerClock}} = \frac{2^{TPreScaler}}{13.56MHz}$$

The possible values for the *TPreScaler* Register range from 0 up to 21. This results in minimum time $T_{TimerClock}$ of about 74 ns up to about 150 ms.

The time period elapsed since the last start event is calculated with

$$T_{Timer} = \frac{TReLoadValue - TimerValue}{f_{TimerClock}}$$

This results in a minimum time T_{Timer} of about 74 ns up to about 40 s.

9.2.4 STATUS OF THE TIMER UNIT

The *TRunning* bit in the *SecondaryStatus* Register shows the timer's current status. Any configured start event starts the timer at the *TReloadValue* and changes the status flag *TRunning* to 1, any configured stop event stops the timer and sets the status flag *TRunning* back to 0. As long as status flag *TRunning* is set to 1, the *TimerValue Register* changes with the next timer unit clock.
The actual timer unit content can be read on-the-fly via the *TimerValue Register*.

9.2.5 TIMESLOTPERIOD

For sending of I•CODE1-Quit-Frames it is necessary to generat a exact chronological relation to the begin of the command frame.
Is *TimeSlotPeriod* > 0, with the end of command transmission the TimeSlotPeriod starst.
If there are Data in the FIFO after reaching the end of TimeSlotPeriod, these data were sent at that moment.
If the FIFO is empty nothing happens.
As long as the contend of TimeSlotPeriod is > 0 the counter for the TimeSlotPeriod will start automatically after reaching the end.
This allows a exact time relation to the end (as well as to the beginning) of the command frame for the generation and sending of the I•CODE1-Quit-Frames

Is TimeSlotPeriod > 0 the next Frame starts exact with the interval

TimeSlotPeriod/CoderRate

delayed after each previous Send Frame. CoderRate (see 5.2.3.5) defines the clock frequency of the coder.
If TimeSlotPeriod = 0, the send function will not be triggered automatically.

The contend of the register TimeSlotPeriod can be changed during the active mode. The modification take effect at the next restart of the TimeSlotPeriod.

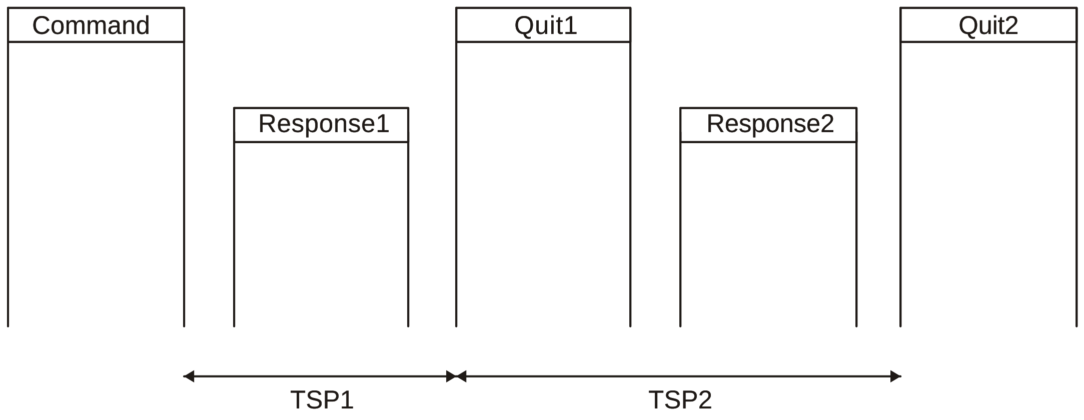
Example:

CoderRate = 0x05 (~52.97kHz)

For I•CODE1 standard mode the interval should be 8.458ms

->TimeSlotPeriod = CoderRate * interval = 52.97kHz * 8.458ms –1 = 447 (447 = 0x1BF)

Note: The MSB of the TimeSlotPeriod is in the SIGOUTSelect register see 5.2.5.7



	TimeSlotPeriod for TSP1	TimeSlotPeriod for TSP2
I•CODE1 Standard Mode	0xBF	0x1BF
I•CODE1 Fast Mode	0x5F	0x67

Note: The MSB of the TimeSlotPeriod is in the SIGOUTSelect register see 5.2.5.7

Note: It is strictly recommended that bit TxCRCEn is set to 0 (see 5.2.5.3) before the Quit-Frame is sent. If the TxCRCEn is not set to 0 a CRC value is calculated and sent with the Quit-Frame.
To calculate the Quit value a CRC8 algorithm has to be used.

9.3 Usage of the Timer Unit

9.3.1 TIME-OUT- AND WATCH-DOG-COUNTER

Having started the timer by setting *TReloadValue* the timer unit decrements the *TimerValue Register* beginning with a certain start event. If a certain stop event occurs e.g. a bit is received from the label, the timer unit stops (no interrupt is generated).
On the other hand, if no stop event occurs, e.g. the label does not answer in the expected time, the timer unit decrements down to zero and generates a timer interrupt request. This signals indicate the µ-Processor that the expected event has not occurred in the given time T_{Timer}.

9.3.2 STOP WATCH

The time T_{Timer} between a certain start- and stop event may be measured by the µ-Processor by means of the SL RC400 timer unit. Setting *TReloadValue* the timer starts to decrement. If the defined stop event occurs the timers stops. The time between start and stop can be calculated by

$$\Delta T = \left(T\mathrm{Re}load_{value} - Timer_{value}\right) * T_{Timer}$$

if the timer does not decrements down to zero.

9.3.3 PROGRAMMABLE ONE-SHOT TIMER

The µ-Processor starts the timer unit and waits for the timer interrupt. After the specified time T_{Timer} the interrupt will occur (*TautoRestart = 0)*.

9.3.4 PERIODICAL TRIGGER

If the µ-Processor sets bit *TautoRestart* and *TreloadValue* not equal 0, it will generate an interrupt request periodically after every T_{Timer}.

9.4 Register Overview Timer Unit

The following table shows the related flags of the Timer Unit in alphabetic order.

Flags	Register	Address
TAutoRestart	TimerClock	0x2A, bit 5
TimerValue	TimerValue	0x0C, bits 7-0
TimerReloadValue	TimerReload	0x2C, bits 7-0
TPreScaler	TimerClock	0x2A, bits 4-0
TRunning	SecondaryStatus	0x05, bit 7
TStartNow	Control	0x09, bit 1
TStartTxBegin	TimerControl	0x2B, bit 0
TStartTxEnd	TimerControl	0x2B, bit 1
TStopNow	Control	0x09, bit 2
TStopRxBegin	TimerControl	0x2B, bit 2
TStopRxEnd	TimerControl	0x2B, bit 3

Table 9-1 Registers associated with the Timer Unit

10 POWER REDUCTION MODES

10.1 Hard Power Down

A Hard Power Down is enabled with HIGH on pin RSTPD. This turns off all internal current sinks including the oscillator. All digital input buffers are separated from the input pads and defined internally (except pin RSTPD itself). The output pins are frozen at a certain value.

This is shown in the following table:

SYMBOL	PIN	TYPE	DESCRIPTION
OSCIN	1	I	Not separated from input, pulled to AVSS
IRQ	2	O	High impedance
RFU	3	I	Separated from Input
SIGOUT	4	O	LOW
TX1	5	O	HIGH
TX2	7	O	LOW
NWR	9	I	Separated from Input
NRD	10	I	Separated from Input
NCS	11	I	Separated from Input
D0 to D7	13 to 20	I/O	Separated from Input
ALE	21	I	Separated from Input
A0	22	I/O	Separated from Input
A1	23	I	Separated from Input
A2	24	I	Separated from Input
AUX	27	O	High impedance
RX	29	I	Not changed
VMID	30	A	Pulled to AVDD
RSTPD	31	I	Not changed
OSCOUT	32	O	HIGH

Table 10-1: Signal on Pins during Hard Power Down

10.2 Soft Power Down

This mode is immediately entered by setting bit *PowerDown* in the *Control-Register*. All internal current sinks are switched off (including the oscillator buffer).
Different from the Hard Power Down Mode, the digital input buffers are not separated from the input pads but keep their functionality. The digital output pins do not change their state.
After resetting bit *PowerDown* in the *Control-Register* it needs 512 clocks until the Soft Power Down mode is left. This is indicated by the *PowerDown* bit itself. Resetting it does not immediately clear it, but it is cleared automatically by the SL RC400 when the Soft Power Down Mode is left.

Note: If the internal oscillator is used, you have to take into account that it is supplied by AVDD and it will take a certain time t_{osc} until the oscillator is stable.

10.3 Stand By Mode

This mode is immediately entered by setting bit *StandBy* in the *Control-Register*. All internal current sinks are switched off (including the internal digital clock buffer but except the oscillator buffer).
Different from the Hard Power Down Mode, the digital input buffers are not separated from the input pads but keep their functionality. The digital output pins do not change their state.
Different from the Soft Power Down Mode, the oscillator does not need time to wake up.
After resetting bit *StandBy* in the *Control-Register* it needs 4 clocks on pin OSCIN until the Stand By Mode is left. This is indicated by the *StandBy* bit itself. Resetting it does not immediately clear it, but it is cleared automatically by the SL RC400 when the Stand By Mode is left.

10.4 Receiver Power Down

It is power saving to switch off the receiver circuit when it is not needed and switched it on again right before data is to be received from the label. This is done automatically by setting bit *RxAutoPD* to 1. If it is set to 0 the receiver is continuously switched on.

11 START UP PHASE

The phases executed during the start up are shown in the following figure:

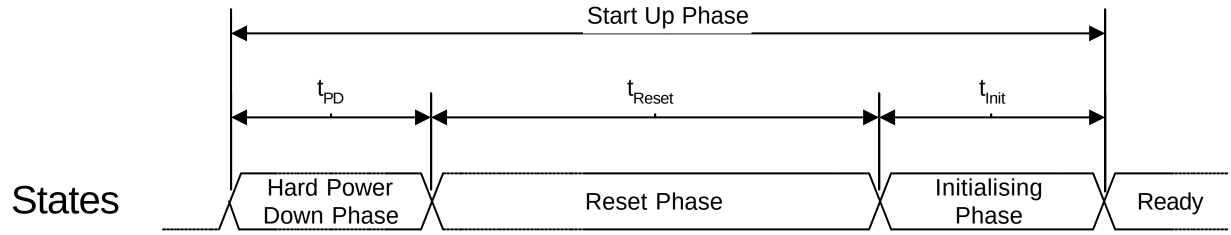


Figure 11-1: Start Up Procedure

11.1 Hard Power Down Phase

The Hard Power Down Phase is active during the following cases:

- Power On Reset caused by power up at pin DVDD
  (active while DVDD is below the digital reset threshold)
- Power On Reset caused by power up at pin AVDD
  (active while AVDD is below the analog reset threshold)
- A HIGH level on pin RSTPD
  (active while pin RSTPD is HIGH)

11.2 Reset Phase

The Reset Phase follows the Hard Power Down Phase automatically. It takes 512 clocks. During the Reset Phase, some of the register bits are preset by hardware. The respective reset values are given in the description of each register (see 5.2.).

Note: If the internal oscillator is used, you have to take into account that it is supplied by AVDD and that it will take a certain time t_{osc} until the oscillator is stable.

11.3 Initialising Phase

The Initialising Phase follows the Reset Phase automatically. It takes 128 clocks. During the Initialising Phase the content of the E^2PROM blocks 1 and 2 is copied into the registers 10_{hex} to $2F_{hex}$ (see 6.3.).

Note: At production test, the SL RC400 is initialised with default configuration values. This reduces the µ-Processors effort for configuring the device to a minimum.

11.4 Initialising the Parallel Interface-Type

For the different connections for the different µ-Processor interface types (see 4.3), a certain initialising sequence shall be applied to enable a proper µ-Processor interface type detection and to synchronise the µ-Processor's and the SL RC400's Start Up.

During the whole Start Up Phase, the *Command* value reads as $3F_{hex}$. At the end of the Initialising Phase the SL RC400 enters the *Idle Command* automatically. Consequently the *Command* value changes to 00_{hex}.

To ensure proper detection of the µ-Processor interface, the following sequence shall be executed:

- Read from the *Command-Register* until the six bit register value for *Command* is 00_{hex}.
  The internal initialisation phase is now completed and the SL RC400 is ready to be controlled.
- Write the value 80_{hex} to the *Page-Register* to initialise the µ-Processor interface.
- Read the *Command-Register*. If its value is 00_{hex} the µ-Processor interface initialisation was successful.

After interface initialisation, the linear addressing mode can be activated by writing 0x00 to the page register(s).

12 OSCILLATOR CIRCUITRY

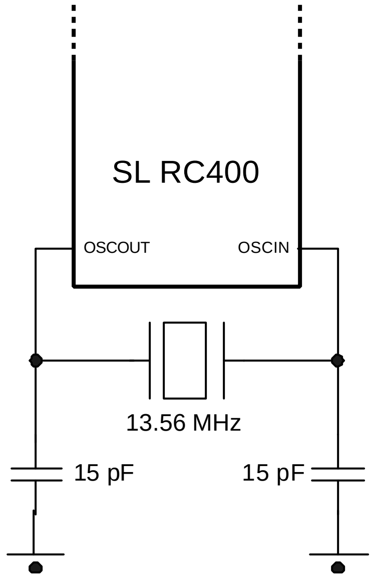


Figure 12-1: Quartz Connection

The clock applied to the SL RC400 acts as time basis for the coder and decoder of the synchronous system. Therefore stability of clock the frequency is an important factor for proper performance. To obtain highest performance, clock jitter has to be as small as possible. This is best achieved by using the internal oscillator buffer with the recommended circuitry. If an external clock source is used, the clock signal has to be applied to pin OSCIN. In this case special care for clock duty cycle and clock jitter is needed and the clock quality has to be verified. It needs to be in accordance with the specifications in chapter 19.5.3.

Remark: It is recommend not to use an external clock source.

13 TRANSMITTER PINS TX1 AND TX2

The signal delivered on TX1 and TX2 is the 13.56 MHz carrier frequency modulated by an envelope signal. It can be used to drive an antenna directly, using a few passive components for matching and filtering (see chapter 17). For that, the output circuitry is designed with an very low impedance source resistance. The signal of TX1 and TX2 can be controlled via the *TxControl Register*.

13.1 Configuration of TX1 and TX2

The configuration possibilities of TX1 are described in the table below:

Register Configuration in *TxControl* *TX1RFEn*	Envelope	Signal on TX1
0	X	LOW (GND)
1	0	13.56 MHz carrier frequenzy modulated
1	1	13.56 MHz carrier frequenzy

Table 13-1: Configurations of Pin TX1

The configuration possibilities of TX2 are described in the table below:

<table>
<tr><th colspan="3">Register Configuration in TxControl</th><th rowspan="2">Envelope</th><th rowspan="2">Signal on TX2</th></tr>
<tr><th>TX2RFEn</th><th>TX2CW</th><th>InvTX2</th></tr>
<tr><td>0</td><td>X</td><td>X</td><td>X</td><td>LOW (GND)</td></tr>
<tr><td rowspan="6">1</td><td rowspan="4">0</td><td rowspan="2">0</td><td>0</td><td>13.56 MHz carrier frequenzy modulated</td></tr>
<tr><td>1</td><td>13.56 MHz carrier frequenzy</td></tr>
<tr><td rowspan="2">1</td><td>0</td><td>13.56 MHz carrier frequenzy modulated,
180° phase shift relative to TX1</td></tr>
<tr><td>1</td><td>13.56 MHz carrier frequenzy,
180° phase shift relative to TX1</td></tr>
<tr><td rowspan="2">1</td><td>0</td><td>X</td><td>13.56 MHz carrier frequenzy</td></tr>
<tr><td>1</td><td>X</td><td>13.56 MHz carrier frequenzy,
180° phase shift relative to TX1</td></tr>
</table>

Table 13-2: Configurations of Pin TX2

13.2 Operating Distance versus Power Consumption

The user has the possibility to find a trade-off between maximum achievable operating distance and power consumption by using different antenna matching circuits as described in 17.3.1 and/or by varying the supply voltage at the antenna driver supply pin TVDD.

13.2.1 ANTENNA DRIVER OUTPUT SOURCE RESISTANCE

The output source conductance of TX1 and TX2 for driving a HIGH level may be adjusted via the value *GsCfgCW* in the *CwConductance Register* in the range from about 1 up to 100 Ohm. The values given are relative to the reference resistance Rs_{rel}, that is measured during production test and stored in the SL RC400 E²PROM. It can be obtained from the Product Information Field (see chapter 6.2).
The electrical specification can be found in chapter 19.4.3.

13.2.1.1 Source Resistance Table

GsConfCW	$EXP_{GsConfCW}$	$MANT_{GsConfCW}$	Rs_{rel}	GsConfCW	$EXP_{GsConfCW}$	$MANT_{GsConfCW}$	Rs_{rel}
0	0	0	8	24	1	8	0,0652
16	1	0	8	25	1	9	0,0580
32	2	0	8	37	2	5	0,0541
48	3	0	8	26	1	A	0,0522
1	0	1	1,0000	27	1	B	0,0474
17	1	1	0,5217	51	3	3	0,0467
2	0	2	0,5000	38	2	6	0,0450
3	0	3	0,3333	28	1	C	0,0435
33	2	1	0,2703	29	1	D	0,0401
18	1	2	0,2609	39	2	7	0,0386
4	0	4	0,2500	30	1	E	0,0373
5	0	5	0,2000	52	3	4	0,0350
19	1	3	0,1739	31	1	F	0,0348
6	0	6	0,1667	40	2	8	0,0338
7	0	7	0,1429	41	2	9	0,0300
49	3	1	0,1402	53	3	5	0,0280
34	2	2	0,1351	42	2	A	0,0270
20	1	4	0,1304	43	2	B	0,0246
8	0	8	0,1250	54	3	6	0,0234
9	0	9	0,1111	44	2	C	0,0225
21	1	5	0,1043	45	2	D	0,0208
10	0	A	0,1000	55	3	7	0,0200
11	0	B	0,0909	46	2	E	0,0193
35	2	3	0,0901	47	2	F	0,0180
22	1	6	0,0870	56	3	8	0,0175
12	0	C	0,0833	57	3	9	0,0156
13	0	D	0,0769	58	3	A	0,0140
23	1	7	0,0745	59	3	B	0,0127
14	0	E	0,0714	60	3	C	0,0117
50	3	2	0,0701	61	3	D	0,0108
36	2	4	0,0676	62	3	E	0,0100
15	0	F	0,0667	63	3	F	0,0093

Table 13-3: Source Resistance of n-Channel Driver Transistor of TX1 and TX2 vs. GsConfCW

13.2.1.2 Formula for the Source Resistance

The relative resistance Rs_{rel} is about

$$Rs_{rel} = \frac{1}{MANT_{GsConfCW} \cdot \left(\frac{77}{40}\right)^{EXP_{GsConfCW}}}$$

13.2.1.3 Calculating the Effective Source Resistance

13.2.1.3.1 Wiring Resistance

Wiring and bonding adds a constant offset to the driver resistance, that is relevant if TX1 and TX2 are switched to low impedance.

$$Rs_{wire,TX1} \approx 500\, m\Omega$$

13.2.1.3.2 Effective Resistance

The source resistances of the driver transistors found in the Product Information Field (see 6.2) are measured at production test with *GsModCW* set to 01_{hex}. To get the driver resistance for a specific value set in *GsModCW* the following formula may be used:

$$Rs_x = \left(Rs_{ref,\max,n} - Rs_{wire,TX1}\right) \cdot Rs_{rel} + Rs_{wire,TX1}.$$

13.3 Changing the Modulation Index

The following table shows the modulation index, if a 50 ohm antenna is used and GsCfgCW is set to 0x3F. To change the modulation index the GsCfgMod register has to be changed similar as the GsCfgCW register.

GsCfgMod	rel. resistance	Mod. index
	Rrel(during modulation)	Rant=50Ω
0x00	Infite	
0x10	Infite	
0x20	Infite	
0x30	Infite	
0x01	1,000	43,45%
0x11	0,522	28,44%
0x02	0,500	27,57%
0x03	0,333	20,08%
0x21	0,270	16,83%
0x12	0,261	16,33%
0x04	0,250	15,73%
0x05	0,200	12,88%
0x13	0,174	11,32%
0x06	0,167	10,88%
0x07	0,143	9,38%
0x31	0,140	9,21%
0x22	0,135	8,89%
0x14	0,130	8,59%
0x08	0,125	8,23%
0x09	0,111	7,32%
0x15	0,104	6,86%
0x0A	0,100	6,57%
0x0B	0,091	5,95%
0x23	0,090	5,89%
0x16	0,087	5,68%
0x0C	0,083	5,43%
0x0D	0,077	4,98%
0x17	0,075	4,81%
0x0E	0,071	4,59%
0x32	0,070	4,50%
0x24	0,068	4,32%
0x0F	0,067	4,26%

GsCfgMod	rel. resistance	Mod. index
	Rrel(during modulation)	Rant=50Ω
0x18	0,065	4,15%
0x19	0,058	3,63%
0x25	0,054	3,35%
0x1A	0,052	3,22%
0x1B	0,047	2,87%
0x33	0,047	2,82%
0x26	0,045	2,69%
0x1C	0,043	2,58%
0x1D	0,040	2,33%
0x27	0,039	2,22%
0x1E	0,037	2,12%
0x34	0,035	1,95%
0x1F	0,035	1,93%
0x28	0,034	1,86%
0x29	0,030	1,58%
0x35	0,028	1,43%
0x2A	0,027	1,35%
0x2B	0,025	1,17%
0x36	0,023	1,08%
0x2C	0,023	1,01%
0x2D	0,021	0,88%
0x37	0,020	0,82%
0x2E	0,019	0,77%
0x2F	0,018	0,67%
0x38	0,018	0,63%
0x39	0,016	0,48%
0x3A	0,014	0,36%
0x3B	0,013	0,26%
0x3C	0,012	0,18%
0x3D	0,011	0,11%
0x3E	0,010	0,05%
0x3F	0,009	0,00%

Note: If the output source conductance (*GsCfgCW*) has been changed GsCfgMod must also be changed to get the same modulation index.

13.4 Pulse Width

The envelope carries the information of the data signal, that shall be transmitted to the label. This is done by coding the data signal according to the 1 out of 256, RZ or 1 out of 4 code. Furthermore, each pause of the coded signal again is coded as a pulse of certain length. The width of this pulse can be adjusted by means of the *ModWidth Register*. The pulse length is calculated by

$$T_{Pulse} = 2\frac{ModWidth + 1}{f_C}$$

where f_c = 13.56MHz.

14 RECEIVER CIRCUITRY

14.1 General

The SL RC400 employs an integrated quadrature-demodulation circuit which extracts the sub-carrier signal from the 13.56 MHz ASK-modulated signal applied to pin RX. The quadrature-demodulator uses two different clocks, Q- and I-clock, with a phase shift of 90° between them. Both resulting subcarrier signals are amplified, filtered and forwarded to the correlation circuitry. The correlation results are evaluated, digitised and passed to the digital circuitry.

For all processing units various adjustments can be made to obtain optimum performance.

14.2 Block Diagram

Figure 14-1 shows the block diagram of the receiver circuitry. The receiving process includes several steps. First the quadrature demodulation of the carrier signal of 13.56 MHz is done. To achieve an optimum in performance an automatic clock Q calibration is recommended (see 14.3.1). The demodulated signal is amplified by an adjustable amplifier. A correlation circuit calculates the degree of similarity between the expected and the received signal. The bit phase register allows to align the position of the correlation intervals with the bit grid of the received signal. In the evaluation and digitizer circuitry the valid bits are detected and the digital results are send to the FIFO register. Several tuning steps in this circuit are possible.

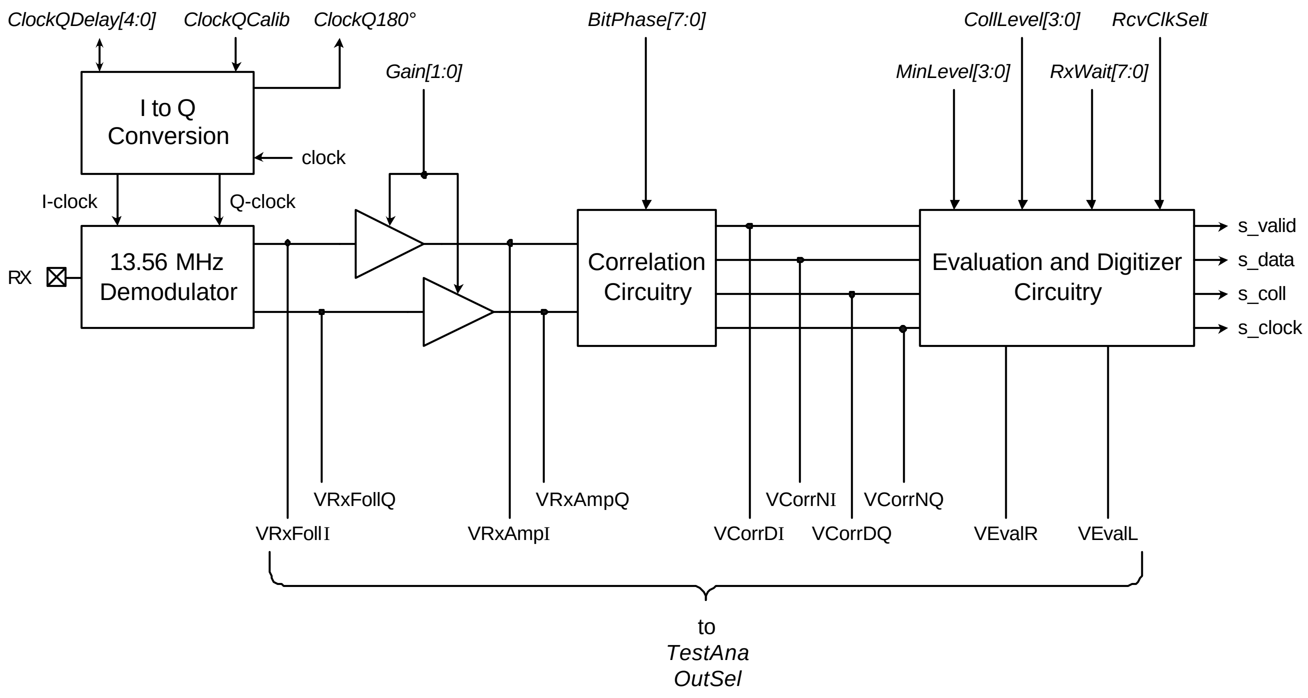


Figure 14-1: Block Diagram of Receiver Circuitry

The user may observe the signal on its way through the receiver as shown in the block diagram above. One signal at a time may be routed to pin AUX using the *TestAnaSelect-Register* as described in 18.3.

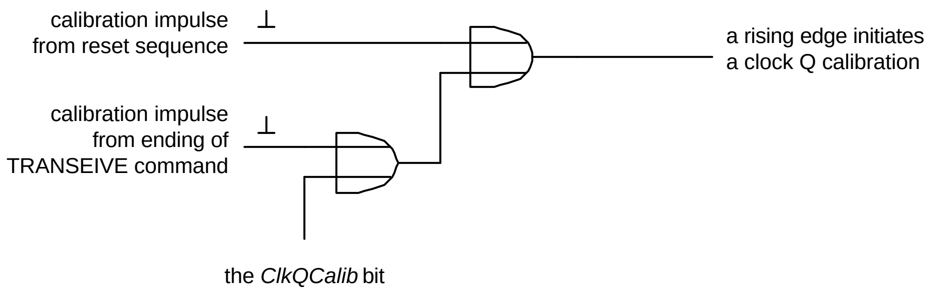
14.3 Putting the Receiver into Operation

In general, the default settings programmed into the Start Up Initialisation File are suitable to use the SL RC400 for data communication with I•CODE labels. However, in some environments specific user settings may achieve better performance.

14.3.1 AUTOMATIC CLOCK-Q CALIBRATION

The quadrature demodulation concept of the receiver generates a phase signal I-clock and a 90° shifted quadrature signal Q-clock. To achieve an optimum demodulator performance, the Q- and the I-clock have to have a difference in phase of 90°. After the reset phase of the SL RC400, a calibration procedure is done automatically. It is possible to have an automatic calibration done at the ending of each Transceive command. To do so, the *ClkQCalib* bit has to be configured to a value of 0.
Configuring this bit to a constant value of 1 disables all automatic calibrations except the one after the reset sequence.
It is also possible to initiate one automatic calibration by software. This is done with a 0 to 1 transition of bit *ClkQCalib.*

The details:



Note: The duration of the automatic clock Q calibration is at most 65 oscillator periods which is approx. 4,8µs.

The value of *ClkQDelay* is proportional to the phase shift between the Q- and the I-clock. The status flag *ClkQ180Deg* shows, that the phase shift between the Q- and the I-clock is greater than 180°.

Notes:

- The startup configuration file enables an automatically Q-clock calibration after the reset.
- While *ClkQCalib* is 1, no automatic calibration is done. Therefore leaving this bit 1 can be used to permanently disable the automatic calibration.
- It is possible to write data to *ClkQDelay* via the µ-Processor. The aim could be a disabling of the automatic calibration and to pre-set the delay by software. But notice, that configuring the delay value by software requires that bit *ClkQCalib* has already been set to 1 before and that a time interval of at least 4.8µs has elapsed since then. Each delay value must be written with the *ClkQCalib* bit set to 1. If *ClkQCalib* is 0 the configured delay value will be overwritten by the next interval automatic calibration.

14.3.2 AMPLIFIER

The demodulated signal has to be amplified with the variable amplifier to achieve the best performance. The gain of the amplifiers can be adjusted by means of the register bits *Gain[1:0]*. The following gain factors are selectable:

Register Setting	**Gain Factor (Simulation Results)**	**Gain Factor [dB] (Simulation Results)**
0	22	26.9
1	35	30.9
2	82	38.3
3	130	42.2

Table 14-1: Gain Factors for the Internal Amplifier

14.3.3 CORRELATION CIRCUITRY

The correlation circuitry calculates the degree of matching between the received and an expected signal. The output is a measure for the amplitude of the expected signal in the received signal. This is done for both, the Q- and the I-channel. The correlator delivers two outputs for each of the two input channels, resulting in four output signals in total.

For optimum performance, the correlation circuitry needs the phase information for the signal coming from the label. This information has to be defined by the µ-Processor by means of the register *BitPhase[7:0]*. This value defines the phase relation between the transmitter and receiver clock in multiples of $t_{BitPhase}$ = 1/13.56 MHz.

14.3.4 EVALUATION AND DIGITIZER CIRCUITRY

For each bit-half of the Manchester coded signal the correlation results are evaluated. The evaluation and digitizer circuit decides from the signal strengths of both bit-halves, whether the current bit is valid, and, if it is valid, the value of the bit itself or whether the current bit-interval contains a collision.

To do this in an optimum way, the user may select the following levels:

- *MinLevel*: Defines the minimum signal strength of the stronger bit-half's signal for being considered valid.
- *CollLevel*: Defines the minimum signal strength that has to be exceeded by the weaker half-bit of the Manchester-coded signal to generate a bit-collision. If the signal's strength is below this value, a 1 and 0 can be determined unequivocally.
  CollLevel defines the minimum signal strength relative to the amplitude of the stronger half-bit.

After transmission of data, the label is not allowed to send its response before a certain time period, called frame guard time in the standard ISO 15693 (similar to I·CODE1). The length of this time period after transmission shall be set in the *RxWait-Register*. The *RxWait-Register* defines when the receiver is switched on after data transmission to the label in multiples of one bit-duration.

If register bit *RcvClkSelI* is set to 1, the I-clock is used to clock the correlator and evaluation circuits. If set to 0, the Q-clock is used.

Note: It is recommended to use the Q-clock.

15 SERIAL SIGNAL SWITCH

15.1 General

Two main blocks are implemented in the SL RC400. A digital circuitry, comprising state machines, coder and decoder logic and so on and an analog circuitry with the modulator and antenna drivers, receiver and amplification circuitry. The interface between these two blocks can be configured in the way, that the interfacing signals may be routed to the pin SIGOUT.

15.2 Block Diagram

Figure 15-1 describes the serial signal switches. Three different switches are implemented in the serial signal switch in order to use the SL RC400 in different configurations.

The serial signal switch may also be used during the design In phase or for test purposes to check the transmitted and received data. Chapter 18.2, describes analog test signals as well as measurements at the serial signal switch.

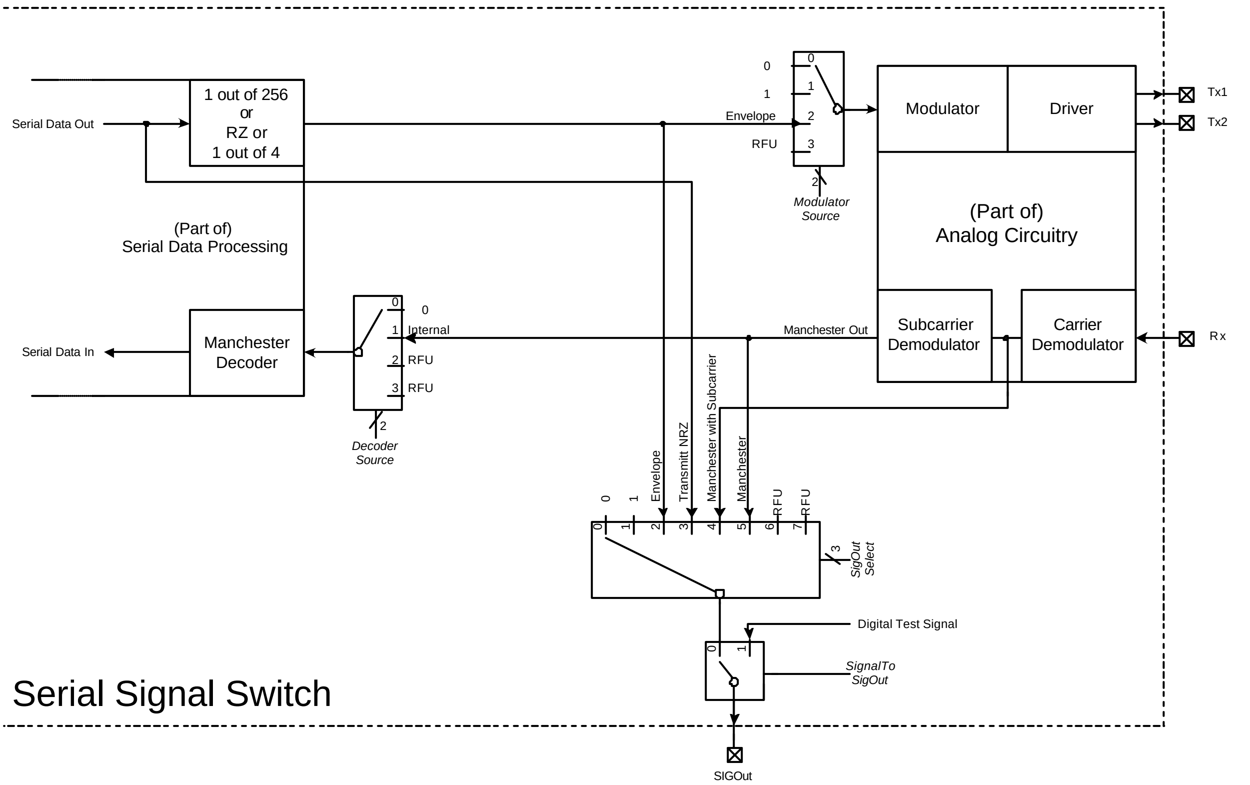


Figure 15-1: Serial Signal Switch

The following chapters describe the relevant registers used to configure and control the serial signal switch.

15.3 Registers Relevant for the Serial Signal Switch

The flags *DecoderSource* define the input signal for the internal Manchester decoder in the following way:

DecoderSource	**Input Signal for Decoder**
0	Constant 0
1	Output of the analog part. This is the default configuration.
2	RFU
3	RFU

Table 15-1: Values for DecoderSource

ModulatorSource defines the signal that modulates the transmitted 13.56 MHz carrier frequenzy. The modulated signal drives the pins TX1 and TX2.

ModulatorSource	**Input Signal for Modulator**
0	Constant 0 (carrier frequency off at pin TX1 and TX2).
1	Constant 1 (continuous carrier frequency delivered at pin TX1 and TX2).
2	Modulation signal (envelope) from the internal coder. This is the default configuration.
3	RFU

Table 15-2: Values for ModulatorSource

SIGOUTSelect defines the input signal for the internal Manchester decoder in the following way:

SIGOUTSelect	**Signal Routed to Pin SIGOUT**
0	Constant 0
1	Constant 1
2	Modulation signal (envelope) from the internal coder.
3	Serial data stream that is to be transmitted (same as for *SIGOUTSelect* = 2, but not coded by the “selected” pulse coder yet).
4	Output signal of the receiver circuit (label modulation signal regenerated and delayed)
5	Output signal of the sub-carrier demodulator (Manchester-coded label signal)
6	RFU
7	RFU

Table 15-3: Values for SIGOUTselect

<u>Note:</u> To use *SIGOUTSelect*, the value of test signal control bit *SignalToSIGOUT* has to be 0.

16 SL RC400 COMMAND SET

16.1 General Description

The SL RC400 behaviour is determined by an internal state machine that is capable to perform a certain set of commands. These commands are started by writing the according command-code to the *Command-Register*.

Arguments and/or data necessary to process a command are exchanged via the FIFO buffer.

16.2 General Behaviour

- Each command, that needs a data stream (or data byte stream) as input will immediately process the data it finds in the FIFO buffer.
- Each command, that needs a certain number of arguments will start processing only when it has received the correct number of arguments via the FIFO buffer.
- The FIFO buffer is not cleared automatically at command start. Therefore, it is also possible to write the command arguments and/or the data bytes into the FIFO buffer and start the command afterwards.
- Each command (except the *StartUp-Command*) may be interrupted by the µ-Processor by writing a new command code into the *Command-Register* e.g.: the *Idle-Command*.

16.3 SL RC400 Commands Overview

Command	Code	Action	Arguments and Data passed via FIFO	Returned Data via FIFO	see Chapter
StartUp	$3F_{hex}$	Runs the Reset- and Initialisation Phase. Note: This command can not be activated by software, but only by a Power-On or Hard Reset	-	-	16.3.2
Idle	00_{hex}	No action: cancels current command execution.	-	-	16.3.3
Transmit	$1A_{hex}$	Transmits data from the FIFO buffer to the label.	Data Stream	-	16.4.1
Receive	16_{hex}	Activates Receiver Circuitry. Note: Before the receiver actually starts, the state machine waits until the time configured in the register RcvWait has passed. Note: This command may be used for test purposes only, since there is no timing relation to the *Transmit-Command*.	-	Data Stream	16.4.2
Transceive	$1E_{hex}$	Transmits data from FIFO buffer to the label and activates automatically the receiver after transmission. Note: Before the receiver actually starts, the SL RC400 waits until the time configured in the register RcvWait has passed. Note: This command is the combination of Transmit and Receive	Data Stream	Data Stream	16.4.3
WriteE2	01_{hex}	Gets data from FIFO buffer and writes it to the internal E²PROM.	Start Address LSB Start Address MSB Data Byte Stream	-	16.5
ReadE2	03_{hex}	Reads data from the internal E²PROM and puts it into the FIFO buffer.	Start Address LSB Start Address MSB Number of Data Bytes	Data Bytes	16.5.2
LoadConfig	07_{hex}	Reads data from E²PROM and initialises the SL RC400 registers.	Start Address LSB Start Address MSB	-	16.6.1
CalcCRC	12_{hex}	Activates the CRC-Coprocessor. Note: The result of the CRC calculation can be read from the registers *CRCResultLSB* and *CRCResultMSB*	Data Byte-Stream	-	16.5

Table 16-1: SL RC400 Command Overview

16.3.1 BASIC STATES

16.3.2 STARTUP COMMAND $3F_{HEX}$

Command	Code$_{hex}$	Action	Arguments and Data	Returned Data
StartUp	3F	Runs the Reset- and Initialisation Phase Note: This command can not be activated by software, but only by a Power-On or Hard Reset	-	-

The *StartUp-Command* runs the Reset- and Initialisation Phase. It does not need or return any data. It can not be activated by the µ-Processor but is started automatically after one of the following events:

- Power On Reset caused by power up at Pin DVDD
- Power On Reset caused by power up at Pin AVDD
- Negative Edge at Pin RSTPD

The Reset-Phase defines certain register bits by an asynchronous reset. The Initialisation-Phase defines certain registers with values taken from the E²PROM.

When the *StartUp-Command* has finished, the *Idle-Command* is entered automatically.

Notes:

- The µ-Processor must not write to the SL RC400 as long as the SL RC400 is busy executing the *StartUp-Command*. To ensure this, the µ-Processor shall poll for the *Idle-Command* to determine the end of the Initialisation Phase (see also chapter 11.4).
- As long as the *StartUp-Command* is active, only reading from page 0 of the SL RC400 is possible.
- The *StartUp-Command* can not be interrupted by the µ-Processor.

16.3.3 IDLE COMMAND 00_{HEX}

Command	Code$_{hex}$	Action	Arguments and Data	Returned Data
Idle	00	No action: cancels current command execution	-	-

The *Idle-Command* switches the SL RC400 to its inactive state. In this Idle-state it waits for the next command. It does not need or return any data. The device automatically enters the Idle-state when a command finishes. In this case the SL RC400 simultaneously initiates an interrupt request by setting bit *IdleIRq*. Triggered by the µ-Processor, the *Idle-Command* may be used to stop execution of all other commands (except the *StartUp Command*). In that case no IdleIRq is generated.

Remark: Stopping a command with the *Idle Command* does not clear the FIFO buffer content.

16.4 Commands for Label Communication

The SL RC400 is a fully ISO 15693 and I·CODE1 compliant reader IC. The following chapter describe the command set for label communication in general.

16.4.1 TRANSMIT COMMAND $1A_{HEX}$

Command	$Code_{hex}$	Action	Arguments and Data	Returned Data
Transmit	1A	Transmits data from FIFO buffer to the label	Data Stream	-

The *Transmit-Command* takes data from the FIFO buffer and forwards it to the transmitter. It does not return any data. The *Transmit-Command* can only be started by the µ-Processor.

16.4.1.1 Working with the Transmit Command

To transmit data one of the following sequences may be used:

1. All data, that shall be transmitted to the label is written to the FIFO while the *Idle-Command* is active. After that, the command code for the *Transmit-Command* is written to the *Command-Register*.
   Note: This is possible for transmission of data with a length of up to 64 bytes.

2. The command code for the *Transmit-Command* is written to *Command-Register* first. Since no data is available in the FIFO, the command is only enabled but transmission is not triggered yet.
   Data transmission really starts with the first data byte written to the FIFO. To generate a continuous data stream on the RF-interface, the µ-Processor has to put the next data bytes to the FIFO in time.
   Note: This allows transmission of data of any length but requires that data is available in the FIFO in time.

3. A part of the data, that shall be transmit to the label is written to the FIFO while the *Idle-Command* is active. After that, the command code for the *Transmit-Command* is written to the *Command-Register*. While the *Transmit-Command* is active, the µ-Processor may feed further data to the FIFO, causing the transmitter to append it to the transmitted data stream.
   Note: This enables transmission of data of any length but requires that data is available in the FIFO in time.

When the transmitter requests the next data byte to keep the data stream on the RF-interface continuous but the FIFO buffer is empty, the *Transmit-Command* automatically terminates. This causes the internal state machine to change its state from Transmit to Idle.

If data transmission to the label is finished, the SL RC400 sets the flag *TxIRq* to signal it to the µ-Processor.

Remark: If the µ-Processor overwrites the transmit code in the *Command-Register* with the *Idle-Command* or any other command, transmission stops immediately with the next clock cycle. This may produce output signals that are not according to the standard ISO 15693 or the I•CODE1 protocol.

16.4.1.2 RF-Channel Redundancy and Framing

Each transmitted ISO 15693 frame consists of a SOF (start of frame) pattern, followed by the data stream and is closed by an EOF (end of frame) pattern. All I•CODE1 command frames consists of a START PULSE followed by the data stream. The I•CODE1 commands have a fix length and no EOF is needed. These different phases of the transmit sequence may be monitored by watching ModemState of *PrimaryStatus-Register* (see 16.4.4).

Depending on the setting of bit TxCRCEn in the *ChannelRedundancy-Register* a CRC is calculated and appended to the data stream. The CRC is calculated according the settings in the *ChannelRedundancy Register*.

16.4.1.3 Transmission of Frames with more than 64 Bytes

To generate frames with more than 64 bytes, the µ-Processor has to write data into the FIFO buffer while the *Transmit Command* is active. The state machine checks the FIFO status when it starts transmitting the last bit of the actual data stream (the check time is marked below with arrows).

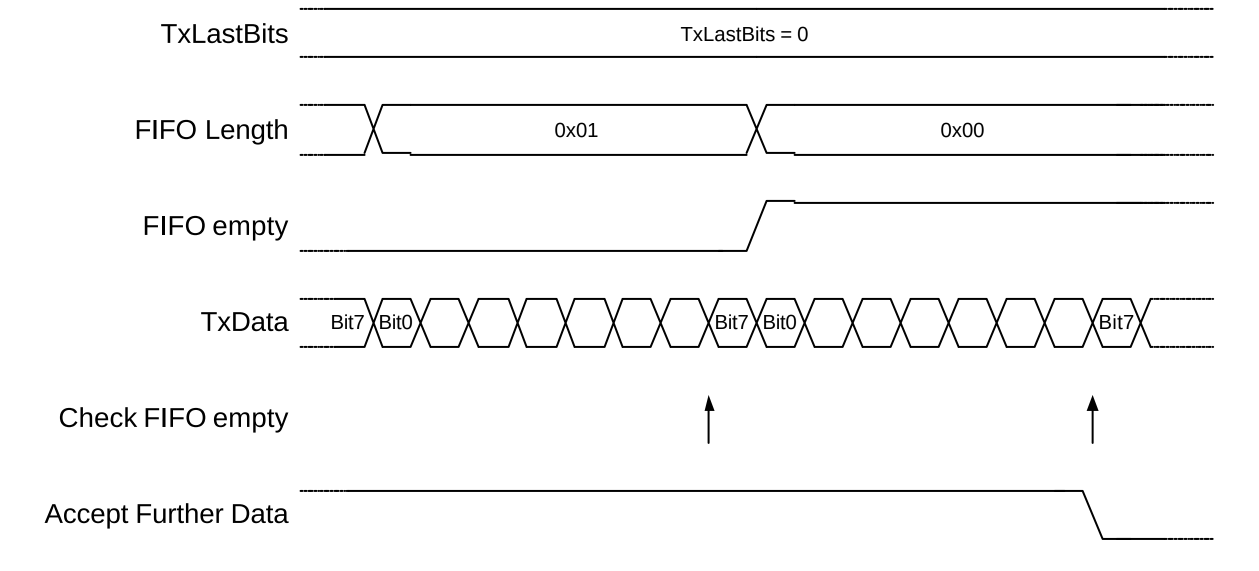


Figure 16-1: Timing for Transmitting Byte Oriented Frames

As long as the internal signal ‘Accept Further Data’ is 1 further data may be loaded into the FIFO. The SL RC400 appends this data to the data stream transmitted via the RF-interface.
If the internal signal ‘Accept Further Data’ is 0 the transmission will terminate. All data written into the FIFO buffer after ‘Accept Further Data’ went 0 will not be transmitted anymore, but remain in the FIFO buffer.

16.4.2 RECEIVE COMMAND 16_{HEX}

Command	$Code_{hex}$	Action	Arguments and Data	Returned Data
Receive	16	Activates Receiver Circuitry	-	Data Stream

The *Receive-Command* activates the receiver circuitry. All data received from the RF interface is returned via the FIFO buffer. The *Receive-Command* can be started either by the µ-Processor or automatically during execution of the *Transceive-Command*.

Note: This command may be used for test purposes only, since there is no timing relation to the *Transmit-Command*.

16.4.2.1 Working with the Receive Command

After starting the *Receive Command* the internal state machine decrements the value set in the *RxWait-Register* with every bit-clock. From 3 down to 1 the analog receiver circuitry is prepared and activated. When the counter reaches 0, the receiver starts monitoring the incoming signal at the RF-interface. If the signal strength reaches a level higher than the value set in the *MinLevel-Register* it finally starts decoding. The decoder stops, if no more signal can be detected on the receiver input pin Rx. The decoder indicates termination of operation by setting bit *RxIRq*.

The different phases of the receive sequence may be monitored by watching ModemState of the *PrimaryStatus-Register* (see 16.4.4).

Note: Since the counter values from 3 to 0 are necessary to initialise the analog receiver circuitry the minimum value for RxWait is 3.

16.4.2.2 RF-Channel Redundancy and Framing

For ISO 15693 the decoder expects a SOF pattern at the beginning of each data stream. If a SOF is detected, it activates the serial to parallel converter and gathers the incoming data bits. For I•CODE1 the decoder do not expects a SOF pattern at the beginning of each data stream. It activates the serial to parallel converter with the first received bit of the data. Every completed byte is forwarded to the FIFO. If an EOF pattern (ISO15693) is detected or the signal strength falls below *MinLevel* set in the *RxThreshold Register*, the receiver and the decoder stop, the *Idle-Command* is entered and an appropriate response for the µ-Processor is generated (interrupt request activated, status flags set).

If bit *RxCRCEn* in the *ChannelRedundancy Register* is set a CRC block is expected. The CRC block may be one byte or two bytes according to bit *CRC8* in the *ChannelRedundancy Register*.

Remark: The received CRC block is not forwarded to the FIFO buffer if it is correct. This is realised by shifting the incoming data bytes through an internal buffer of either one or two bytes (depending on the defined CRC). The CRC block remains in this internal buffer. As a consequence all data bytes are available in the FIFO buffer one or two bytes delayed.
If the CRC fails all received bytes are forwarded to the FIFO buffer (including the faulty CRC itself).

16.4.2.3 Collision Detection

If more than one label is within the RF-field during the label selection phase, they will respond simultaneously. The SL RC400 supports the algorithm defined in ISO 15693 as well as the I•CODE1 anti-collision algorithm to resolve data-collisions of label serial numbers by doing the so-called anti-collision procedure. The basis for this is the ability to detect bit-collisions.

Bit-collision detection is supported by the used bit-coding scheme, namely the Manchester-coding. If in the first and second half-bit of a bit a sub-carrier modulation is detected, instead of forwarding a 1 or a 0 a bit collision will be signalled. To distinguish a 1 or 0-bit from a bit-collision, the SL RC400 uses the setting of *CollLevel*. If the amplitude of the half-bit with smaller amplitude is larger than defined by *CollLevel*, the SL RC400 indicates a bit-collision.

If a bit-collision is detected, the error flag *CollErr* is set.

Independent from the detected collision the receiver continues receiving the incoming data stream. In case of a bit-collision, the decoder forwards 1 at the collision position.

Note: As an exception, if bit *ZeroAfterColl* is set, all bits received after the first bit-collision are forced to zero, regardless whether a bit-collision or an unequivocal state has been detected. This feature eases for the software to carry out the anti-collision procedure defined in ISO 15693.

When the first bit collision in a frame is detected, the bit position of this collision is stored in the *CollPos Register*.

The collision position follows the table below:

Collision in Bit	Value of CollPos
SOF	0
LSBit of LSByte	1
…	…
MSBit of LSByte	8
LSBit of second Byte	9
…	…
MSBit of second Byte	16
LSBit of third Byte	17
…	…

Table 16-2: Returned Values for Bit Collision Positions

If a collision is detected in the SOF a frame error is reported and no data is forwarded to the FIFO buffer. In this case the receiver continues to monitor the incoming signal and generates the correct notifications to the µ-Processor when the ending of the faulty input stream is detected. This helps the µ-Processor to determine the time when it is allowed next to send anything to the label.

16.4.2.4 Communication Errors

The following table shows which event causes the setting of error flags:

Cause	Bit, that is set
Received data did not start with a SOF pattern.	FramingErr
The CRC block is not equal the expected value.	CRCErr
The received data is shorter than the CRC block.	CRCErr
A collision is detected.	CollErr

Table 16-3: Communication Error Table

16.4.3 TRANSCEIVE COMMAND 1E$_{HEX}$

Command	Code$_{hex}$	Action	Arguments and Data	Returned Data
Transceive	1E	Transmits data from FIFO buffer to the label and then activates automatically the receiver	Data Stream	Data Stream

The *Transceive-Command* first executes the *Transmit-Command* (see 16.4.1) and then automatically starts the *Receive-Command* (see 16.4.2). All data that shall be transmitted is forwarded via the FIFO buffer and all data received is returned via the FIFO buffer. The *Transceive-Command* can be started only by the µ-Processor.

Note: To adjust the timing relation between transmitting and receiving, the *RxWait Register* is used to define the time delay from the last bit transmitted until the receiver is activated. Furthermore, the *BitPhase Register* determines the phase-shift between the transmitter and the receiver clock.

16.4.4 STATES OF THE LABEL COMMUNICATION

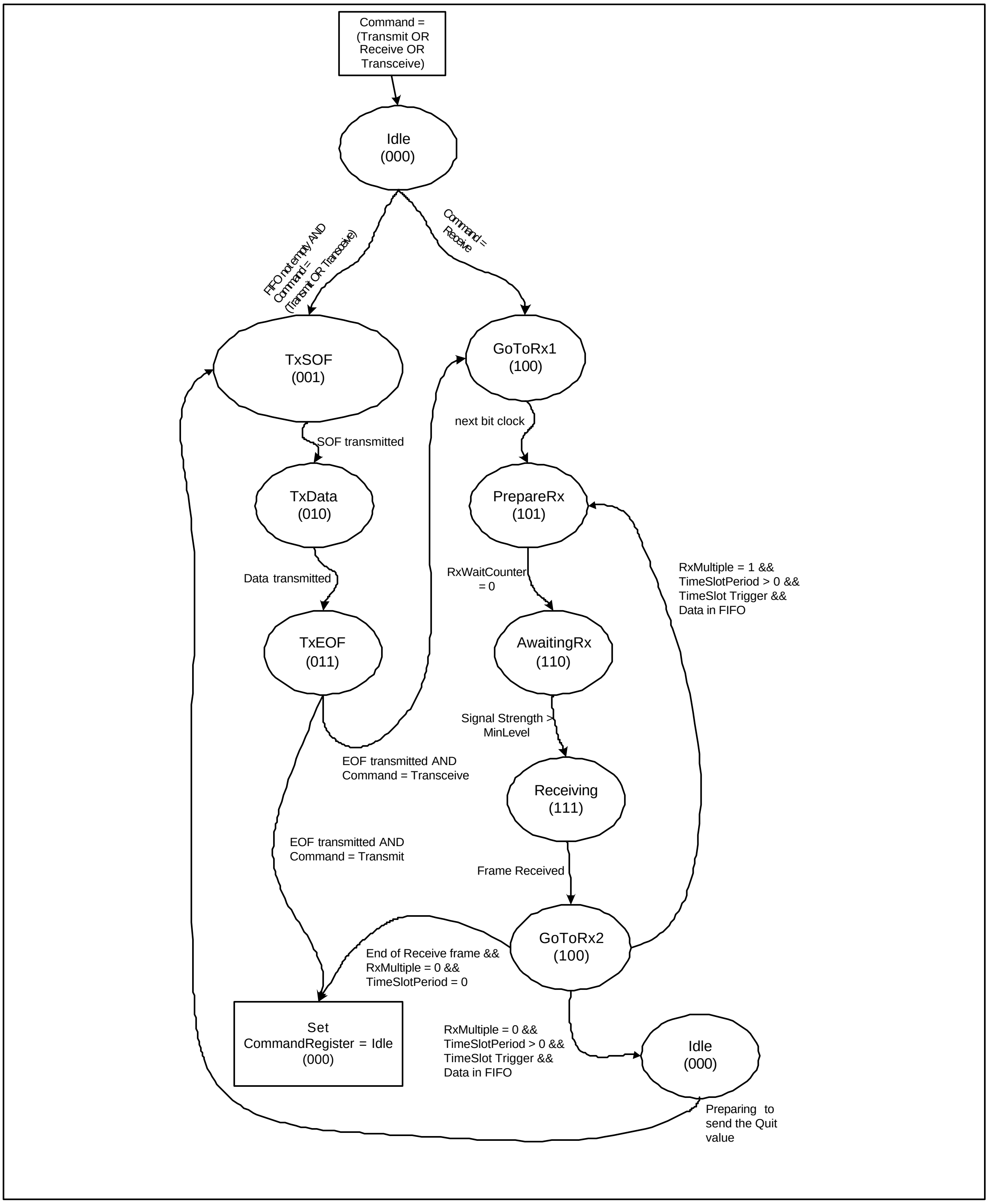
The actual state of the transmitter and receiver state machine can be fetched from *ModemState* in the *PrimaryStatus Register*.

The assignment of *ModemState* to the internal action is shown in the following table:

ModemState	Name of State	Description
000	Idle	Neither the transmitter nor the receiver is in operation, since none of them is started or the transmitter has not got input data
001	TxSOF	Transmitting the ‘Start Of Frame’ Pattern
010	TxData	Transmitting data from the FIFO buffer (or redundancy check bits)
011	TxEOF	Transmitting the ‘End Of Frame’ Pattern
100	GoToRx1	Intermediate state passed, when receiver starts
	GoToRx2	Intermediate state passed, when receiver finishes
101	PrepareRx	Waiting until the time period selected in the *RxWait Register* has expired
110	AwaitingRx	Receiver activated; Awaiting an input signal at pin Rx
111	Receiving	Receiving data

Table 16-4: Meaning of ModemState

16.4.5 STATE DIAGRAM FOR THE LABEL COMMUNICATION



Remark: I•CODE1 do not have a SOF and a EOF

16.5 Commands to Access the E²PROM

16.5.1 WRITEE2 COMMAND 01_{HEX}

16.5.1.1 Overview

Command	Code$_{hex}$	Action	Arguments and Data passed via FIFO	Returned Data via FIFO
WriteE2	01	Get data from FIFO buffer and write it to the E²PROM	Start Address LSB Start Address MSB Data Byte Stream	-

The *WriteE2-Command* interprets the first two bytes in the FIFO buffer as E²PROM starting byte-address. Any further bytes are interpreted as data bytes and are programmed into the E²PROM, starting from the given E²PROM starting byte-address. This command does not return any data.

The *WriteE2-Command* can only be started by the µ-Processor. It will not stop automatically but has to be stopped explicitly by the µ-Processor by issuing the *Idle-Command*.

16.5.1.2 Programming Process

One byte up to 16 byte can be programmed into the E^2PROM in one programming cycle. The time needed will be in any case about 5.8ms.

The state machine copies all data bytes prepared in the FIFO buffer to the E²PROM input buffer. The internal E²PROM input buffer is 16 byte long which is equal the block size of the E²PROM. A programming cycle is started either if the last position of the E²PROM input buffer is written or if the last byte of the FIFO buffer has been fetched.

As long as there are unprocessed bytes in the FIFO buffer or the E²PROM programming cycle still is in progress, the flag *E2Ready* is 0. If all data from the FIFO buffer are programmed into the E²PROM, the flag *E2Ready* is set to1. Together with the rising edge of *E2Ready* the interrupt request flag *TxIRq* indicates a 1. This may be used to generate an interrupt when programming of all data is finished.

After the *E2Ready* bit is set to 1, the *WriteE2-Command* may be stopped by the µ-Processor by issuing the *Idle-Command*.

Important: The *WriteE2-Command* must not be stopped by starting another command before the *E2Ready* flag is set to 1. Otherwise the content of the currently processed E²PROM block will not be defined or in worst case the SL RC400 functionality is in-reversibly reduced.

16.5.1.3 Timing Diagram

The following diagram shows programming of 5 bytes into the E²PROM:

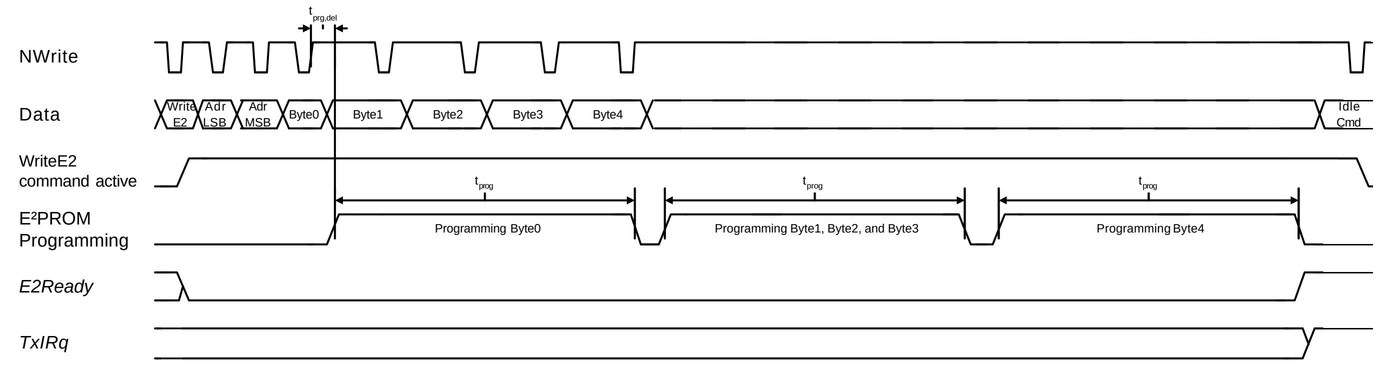


Figure 16-3: Timing Diagram for E²PROM programming

Explanation: It is assumed, that the SL RC400 finds and reads Byte 0 before the µ-Processor is able to write Byte 1 ($t_{prog,del}$ = 300 ns). This causes the SL RC400 to start the programming cycle, which needs about t_{prog} = 2.9 ms. In the meantime the µ-Processor stores Byte 1 to Byte 4 to the FIFO buffer. Assuming, that the E²PROM starting byte-address is e.g. $4C_{hex}$ then Byte 0 is stored exactly there. The SL RC400 copies the following data bytes into the E²PROM input buffer. Copying Byte 3, it detects, that this data byte has to be programmed at the E²PROM byte-address $4F_{hex}$. Since this is the end of the memory block, the SL RC400 automatically starts a programming cycle. In the next turn, Byte 4 will be programmed at the E²PROM byte-address 50_{hex}. Since this is the last data byte, the flags (*E2Ready* and *TxIRq*) that indicate the end of the E²PROM programming activity will be set.

Although all data has been programmed into the E2PROM, the SL RC400 stays in the *WriteE2-Command*. Writing further data to the FIFO would lead to further E²PROM programming, continuing at the E²PROM byte-address 51_{hex}. The command is stopped using the *Idle-Command*.

16.5.1.4 Error Flags for the WriteE2 Command

Programming is inhibited for the E²PROM blocks 0 (E²PROM's byte-address 00_{hex} to $0F_{hex}$). Programming to these addresses sets the flag *AccessErr*. No programming cycle is started (for the E²PROM memory organisation refer to chapter 6.). It is strictly recommended to use only the described E²PROM address area.

16.5.2 READE2 COMMAND 03_{HEX}

16.5.2.1 Overview

Command	$Code_{hex}$	Action	Arguments	Returned Data
ReadE2	03	Reads data from E²PROM and puts it to the FIFO buffer	Start Address LSB Start Address MSB Number of Data Bytes	Data Bytes

The *ReadE2-Command* interprets the first two bytes found in the FIFO buffer as E²PROM starting byte-address. The next byte specifies the number of data bytes that shall be returned.
When all three argument-bytes are available in the FIFO buffer, the specified number of data bytes is copied from the E²PROM into the FIFO buffer, starting from the given E²PROM starting byte-address.

The *ReadE2-Command* can be triggered only by the µ-Processor. It stops automatically when all data has been delivered.

Note: It is strictly recommended to use only the described E²PROM address area.

16.6 Diverse Commands

16.6.1 LOADCONFIG COMMAND 07_{HEX}

16.6.1.1 Overview

Command	$Code_{hex}$	Action	Arguments and Data	Returned Data
LoadConfig	07	Reads data from E²PROM and initialises the registers	Start Address LSB Start Address MSB	-

The *LoadConfig-Command* interprets the first two bytes found in the FIFO buffer as E²PROM starting byte-address. When the two argument-bytes are available in the FIFO buffer, 32 bytes from the E²PROM are copied into the SL RC400 control and configuration registers, starting at the given E²PROM starting byte-address. The *LoadConfig-Command* can only be started by the µ-Processor. It stops automatically when all relevant registers have been copied.

Note: It is strictly recommended to use only the described E²PROM address area.

16.6.1.2 Register Assignment

The 32 bytes of E²PROM content, beginning with the E²PROM starting byte-address, is written to the SL RC400 register 10_{hex} up to register $2F_{hex}$ (for the E²PROM memory organisation see 6).

Note: The procedure for the register assignment is the same as it is for the Start Up Initialisation (see 11.3). The difference is, that the E²PROM starting byte-address for the Start Up Initialisation is fixed to 10_{hex} (Block 1, Byte 0). With the *LoadConfig-Command* it can be chosen.

16.6.1.3 Relevant Error Flags for the LoadConfig-Command

Valid E²PROM starting byte-addresses are in the range from 10_{hex} up to 60_{hex}.

16.6.2 CALCCRC COMMAND 12_{HEX}

16.6.2.1 Overview

Command	$Code_{hex}$	Action	Arguments and Data	Returned Data
CalcCRC	12	Activates the CRC-Coprocessor	Data Byte-Stream	-

The *CalcCRC-Command* takes all data from the FIFO buffer as input bytes for the CRC-Coprocessor. All data stored in the FIFO buffer before the command is started will be processed. This command does not return any data via the FIFO buffer, but the content of the CRC-register can be read back via the *CRCResultLSB-register* and the *CRCResultMSB-register*. The *CalcCRC-Command* can only be started by the µ-Processor. It does not stop automatically but has to be stopped explicitly by the µ-Processor with the *Idle-Command*. If the FIFO buffer is empty, the *CalcCRC-Command* waits for further input from the FIFO buffer.

Note: Do not use this command to calculate the Quit value for I?CODE1 tag’s because this would terminate the Transceive command.

16.6.2.2 CRC-Coprocessor Settings

For the CRC-Coprocessor the following parameters may be configured:

Parameter	Value	Bit	Register
CRC Register Length	8 Bit or 16 Bit CRC	CRC8	ChannelRedundancy
CRC Algorithm	1 = Algorithm according ISO 15693 or according ISO/IEC3309 0 = algorithm according to I•CODE1	CRC3309	ChannelRedundancy
Bit-Processing Direction	Shift the MSBit or LSBit first into the CRC-register	CRCMSBFirst	ChannelRedundancy
CRC Preset Value	Any	CRCPresetLSB, CRCPresetMSB	CRCPresetLSB, CRCPresetMSB

Table 16-5: CRC-Coprocessor Parameters

The CRC polynomial for the 8-bit CRC is fixed to $x^8 + x^4 + x^3 + x^2 + 1$.
The CRC polynomial for the 16-bit CRC is fixed to $x^{16} + x^{12} + x^5 + 1$.

16.6.2.3 Status Flags of the CRC-Coprocessor

The status flag *CRCReady* indicates, that the CRC-Coprocessor has finished processing of all data bytes found in the FIFO buffer. With the *CRCReady* flag setting to 1, an interrupt is requested with *TxIRq* being set. This supports interrupt driven usage of the CRC-Coprocessor.

When *CRCReady* and *TxIRq* are set to 1, respectively, the content of the *CRCResultLSB-* and *CRCResultMSB-register* and the flag *CRCErr* is valid.

The *CRCResultLSB-* and *CRCResultMSB-register* hold the content of the CRC register, the *CRCErr* flag indicates CRC validity for the processed data.

16.7 Error Handling during Command Execution

If any error is detected during command execution, this is shown by setting the status flag *Err* in the *PrimaryStatus Register*. For information about the cause of the error, the µ-Processor may evaluate the status flags in the *ErrorFlag Register*.

Error Flag of the *ErrorFlag Register*	Related to Command
AccessError	*WriteE2, ReadE2, LoadConfig*
FIFOOvl	*Not related to a command*
CRCErr	*Receive, Transceive, CalcCRC*
FramingErr	*Receive, Transceive*
CollErr	*Receive, Transceive*

Table 16-6: Error Flags Overview

17 TYPICAL APPLICATION

17.1 Circuit Diagram

The figure below shows a typical application, where the antenna is direct connected to the SL RC400:

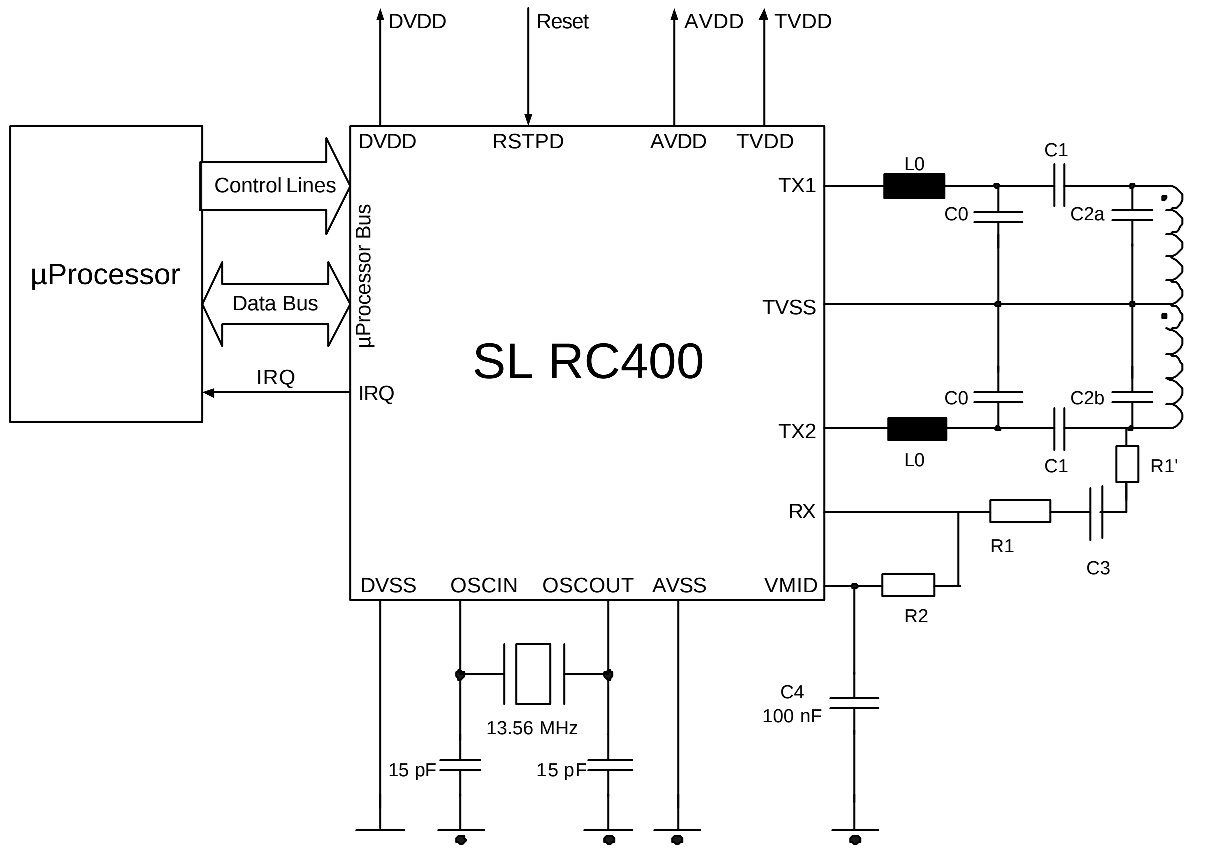


Figure 17-1: Circuit Diagram for Application Example: Direct Matched Antenna

The matching circuit consists of an EMC low pass filter, a receiving circuit, an antenna matching circuit and the antenna itself.

For more detailed information about designing and tuning an antenna please refer to chapter 17.3.1.

17.2 Circuit Description

17.2.1 EMC LOW PASS FILTER

The I•CODE system operates at a frequency of 13.56 MHz. This frequency is generated by a quartz oscillator to clock the SL RC400 and is also the basis for driving the antenna with the 13.56 MHz carrier frequenzy. This will not only cause emitted power at 13.56 MHz but will also emit power at higher harmonics. The international EMC regulations define the amplitude of the emitted power in a broad frequency range. Thus, an appropriate filtering of the output signal is necessary to fulfil these regulations.

A multi-layer board it is strongly recommended to implement a low pass filter as shown in the circuit above. The low pass filter consists of the components L0 and C0. The values are given in table below.

Note: To achieve best performance all components shall have at least the quality of the recommended ones.

17.2.2 RECEIVING CIRCUIT

The internal receiving part of the SL RC400 uses a concept that benefits from both side-bands of the sub-carrier load modulation of the label response. It is recommended to use the internally generated VMID potential as the input potential of pin RX. To provide a stable reference voltage a capacitance C4 to ground has to be connected to VMID. The receiving part of the reader needs a voltage divider connected between the RX and the VMID pin. Additionally, it is recommended to use a series capacitance between the antenna coil and the voltage divider. The circuit diagram above shows the recommended receiving circuit. The receiving circuit consists of the components R1, R2, C3 and C4. The values are given in the table below.

Components	Value	Remark
L0	1 µH ± 5%	Magnetic shielded e.g. TDK NL322522T-1R0J
C0	2 * 68 pF ± 2%	NP0 material, Value depending on the antenna inductance
R1’	3.9 kΩ ± 1%	
R1	560 Ω ± 1%	
R2	820Ω ± 1%	
C3	1 nF ± 2%	NP0 material

Table 17-1: Values for the EMC- Filter and Receiving Circuit

Note: It is recommended not to use X7R material for the capacitors.

17.3 Calculation of the Antenna Coil Inductance

The precise calculation of the antenna coils inductance is not practicable but the inductance can be **estimated** using the following formula. We recommend designing an antenna either with a circular or rectangular shape.

$$L_1\,[nH] = 2 \cdot l_1\,[cm] \cdot \left(\ln\left(\frac{l_1}{D_1} \right) - K \right) N_1^{1,8}$$

l_1............. Length of the conductor loop of one turn

D_1........... Diameter of the wire or width of the PCB conductor respectively

K .... Antenna Shape Factor (K = 1,07 for circular antennas and K = 1,47 for square antennas)

N_1........... Number of turns

ln............. Natural logarithm function

17.3.1 IMPEDANCE MATCHING FOR DIRECTLY MATCHED ANTENNAS

To design a matching circuit for a directly matched antenna we recommend to use the circuit shown in 17.1. The values for the capacitors C1 and C2a, C2b depend on the antenna's electrical properties and environmental influences.

The values for the capacitors shown in the table below are guidelines. In fact, they are used as starting values for the tuning procedure.

Antenna Coil Inductance [µH]	C1 [pF]	C2a [pF]	C2b [pF]
0.8	27	270	330
0.9	27	270	270
1.0	27	220	270
1.1	27	180 || 22	220
1.2	27	180	180 || 22
1.3	27	180	180
1.4	27	150	180
1.5	27	150	150
1.6	27	120 || 10	150
1.7	27	120	150
1.8	27	120	120

Table 17-2: Capacitance Values for the Matching Circuit

However, for optimum performance, the accurate values have to be found by the tuning, variation of the capacitance's C2x and C1.

The above table assumes a stray capacitance of 15 pF of the antenna coil. The capacitors C1 and C2s should have a NP0 dielectric with a tolerance of +/-2 %.

The actual values of the antenna inductance and capacitance depend on various parameters like:

- antenna construction (Type of PCB)
- thickness of conductor
- distance between the windings
- shielding layer
- metal or ferrite in the near environment

18 TEST SIGNALS

18.1 General

The SL RC400 allows different kind of signal measurements. These measurements can be used to check the internally generated and received signals using the possibilities of the serial signal switch as described in chapter 15.
Furthermore, with the SL RC400 the user may select internal analog signals to measure them at pin AUX and internal digital signals to observe them on pin SIGOUT by register selections. These measurements can be helpful during the design-in phase to optimise the receiver's behaviour or for test purpose.

18.2 Measurements Using the Serial Signal Switch

Using the serial signal switch at pin SIGOUT the user may observe data send to the label or data received from the label. The following tables give an overview of the different signals available.

SignalToSIGOUT	SIGOUTSelect	Signal routed to SIGOUT pin
0	0	LOW
0	1	HIGH
0	2	Envelope
0	3	Transmit NRZ
0	4	Manchester with Subcarrier
0	5	Manchester
0	6	RFU
0	7	RFU
1	X	Digital Test signal

Table 18-1 Signal routed to SIGOUT pin

18.3 Analog Test-Signals

The analog test signals may be routed to pin AUX by selecting them with the register bits *TestAnaOutSel*.

Value	Signal Name	Description
0	V_{mid}	Voltage at internal node Vmid
1	$V_{bandgap}$	Internal reference voltage generated by the band gap.
2	$V_{RxFollI}$	Output signal from the demodulator using the I-clock.
3	$V_{RxFollQ}$	Output signal from the demodulator using the Q-clock.
4	V_{RxAmpI}	I-channel subcarrier signal amplified and filtered.
5	V_{RxAmpQ}	Q-channel subcarrier signal amplified and filtered.
6	V_{CorrNI}	Output signal of N-channel correlator fed by the I-channel subcarrier signal.
7	V_{CorrNQ}	Output signal of N-channel correlator fed by the Q-channel subcarrier signal.
8	V_{CorrDI}	Output signal of D-channel correlator fed by the I-channel subcarrier signal.
9	V_{CorrDQ}	Output signal of D-channel correlator fed by the Q-channel subcarrier signal.
A	V_{EvalL}	Evaluation signal from the left half bit.
B	V_{EvalR}	Evaluation signal from the right half bit.
C	V_{Temp}	Temperature voltage derived from band gap.
D	RFU	Reserved for future use
E	RFU	Reserved for future use
F	RFU	Reserved for future use

Table 18-2: Analog Test Signal Selection

18.4 Digital Test-Signals

Digital test signals may be routed to pin SIGOUT by setting bit *SignalToSIGOUT* to 1. A digital test signal may be selected via the register bits *TestDigiSignalSel* in Register *TestDigiSelect*.

The signals selected by a certain *TestDigiSignalSel* setting is shown in the table below:

TestDigiSignalSel	Signal Name	Description
$F4_{hex}$	s_data	Data received from the label.
$E4_{hex}$	s_valid	Shows with 1, that the signals s_data and s_coll are valid.
$D4_{hex}$	s_coll	Shows with 1, that a collision has been detected in the current bit.
$C4_{hex}$	s_clock	Internal serial clock: during transmission, this is the coder-clock and during reception this is the receiver clock.
$B5_{hex}$	rd_sync	Internal synchronised read signal (derived from the parallel µ-Processor interface).
$A5_{hex}$	wr_sync	Internal synchronised write signal (derived from the parallel µ-Processor interface).
96_{hex}	int_clock	Internal 13.56 MHz clock.
00_{hex}	no test signal	output as defined by *SIGOUTSelect* are routed to pin SIGOUT.

Table 18-3: Digital Test Signal Selection

If no test signals are used, the value for the *TestDigiSel-Register* shall be 00_{hex}.

Note: All other values of *TestDigiSignalSel* are for production test purposes only.

18.5 Examples of Analog- and Digital Test Signals

Fig. 17 shows the answer of an I•CODE1 Label IC to a unselected read command using the Qclock receiving path.

RX –Reference is given to show the Manchester modulated signal at the RX pin. This signal is demodulated and amplified in the receiver circuitry VRXAmpQ shows the amplified side band signal having used the Q-Clock for demodulation. The signals VCorrDQ and VCorrNQ generated in the correlation circuitry are evaluated and digitised in the evaluation and digitizer circuitry. VEvalR and VEvalL show the evaluation signal of the right and left half bit. Finally, the digital test-signal S_data shows the received data which is send to the internal digital circuit and S_valid indicates that the received data stream is valid.

Receiving path Q-Clock

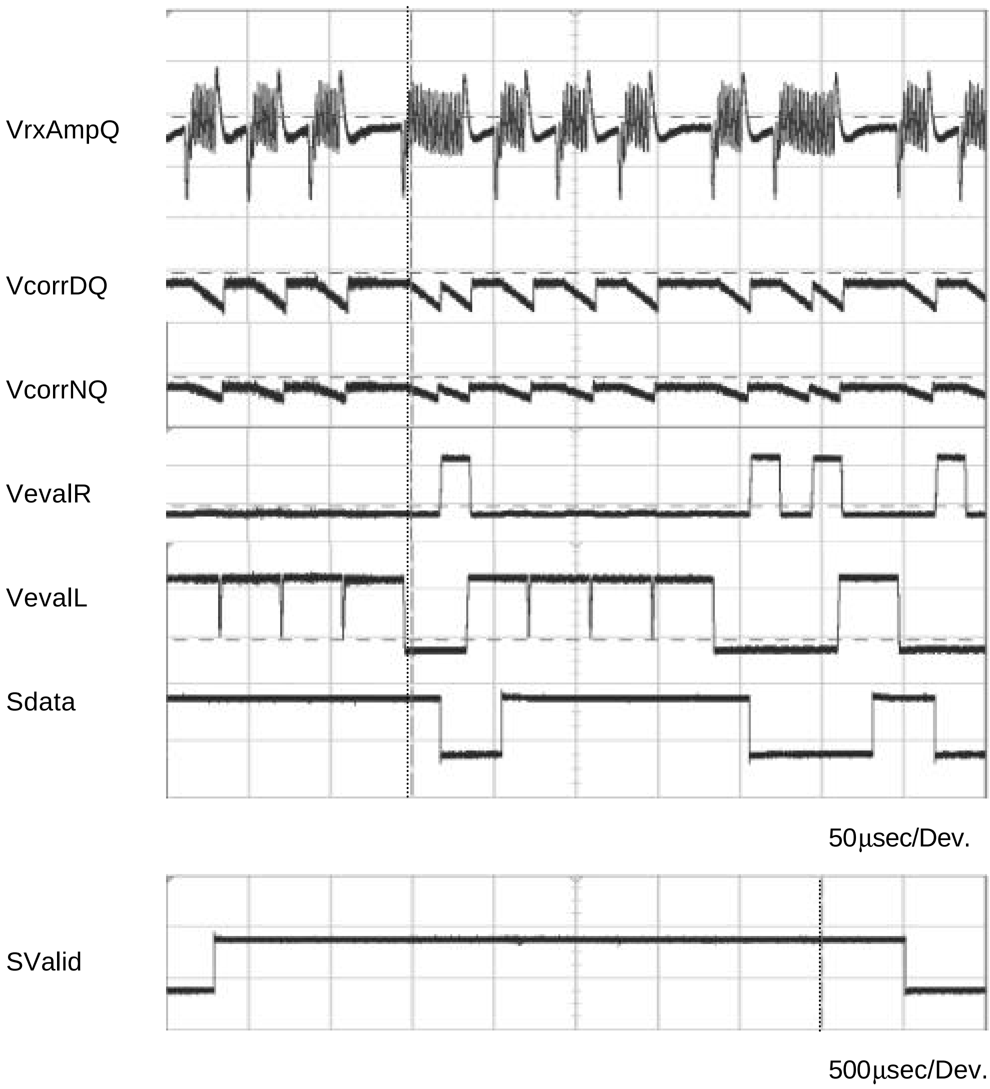


Figure 17. Receiving path Q-Clock

19 ELECTRICAL CHARACTERISTICS

19.1 Absolute Maximum Ratings

SYMBOL	PARAMETER	MIN	MAX	UNIT
$T_{amb,abs}$	Ambient or Storage Temperature Range	-40	+150	°C
DVDD AVDD TVDD	DC Supply Voltages	-0.5	6	V
$V_{in,abs}$	Absolute voltage on any digital pin to DVSS	-0.5	DVDD + 0.5	V
$V_{RX,abs}$	Absolute voltage on RX pin to AVSS	-0.5	AVDD + 0.5	V

Table 19-1: Absolute Maximum Ratings

19.2 Operating Condition Range

SYMBOL	PARAMETER	CONDITIONS	MIN	TYP	MAX	UNIT
T_{amb}	Ambient Temperature	-	-25	+25	+85	°C
DVDD	Digital Supply Voltage	DVSS = AVSS = TVSS = 0V	4.5	5.0	5.5	V
AVDD	Analog Supply Voltage		4.5	5.0	5.5	V
TVDD	Transmitter Supply Voltage		3.0	5.0	5.5	V

Table 19-2: Operating Condition Range

19.3 Current Consumption

SYMBOL	PARAMETER	CONDITIONS	MIN	TYP	MAX	UNIT
I_{DVDD}	Digital Supply Current	Idle Command		6		mA
		Stand By Mode		3		mA
		Soft Power Down Mode		800		µA
		Hard Power Down Mode		1		µA
I_{AVDD}	Analog Supply Current	Idle Command, Receiver On		29		mA
		Idle Command, Receiver Off		10		mA
		Stand By Mode		8		mA
		Soft Power Down Mode		1		µA
		Hard Power Down Mode		1		µA
I_{TVDD}	Transmitter Supply Current	Continuous Wave			150	mA
		TX1 and TX2 unconnected *TX1RFEn*, *TX2RFEn* = 1		4.5	9	mA
		TX1 and TX2 unconnected *TX1RFEn*, *TX2RFEn* = 0		65	130	µA

Table 19-3: Current Consumption

19.4 Pin Characteristics

19.4.1 INPUT PIN CHARACTERISTICS

Pins D0 to D7, A0 and A1 have TTL input characteristics and behave as defined in the following table.

SYMBOL	PARAMETER	CONDITIONS	MIN	MAX	UNIT
I_{Leak}	Input Leakage Current		-1.0	+1.0	µA
V_T	Threshold		0.8	2.0	V

Table 19-4: Standard Input Pin Characteristics

The digital input pins NCS, NWR, NRD, ALE and A2 have Schmitt-Trigger characteristics, and behave as defined in the following table.

SYMBOL	PARAMETER	CONDITIONS	MIN	MAX	UNIT
I_{Leak}	Input Leakage Current		-1.0	+1.0	µA
V_{T+}	Positive-Going Threshold		1.4	2.0	V
V_{T-}	Negative-Going Threshold		0.8	1.3	V

Table 19-5: Schmitt-Trigger Input Pin Characteristics

Pin RSTPD has Schmitt-Trigger CMOS characteristics. In addition, it is internally filtered with an RC-low-pass filter, which causes a relevant propagation delay for the reset signal:

SYMBOL	PARAMETER	CONDITIONS	MIN	MAX	UNIT
I_{Leak}	Input Leakage Current		-1.0	+1.0	µA
V_{T+}	Positive-Going Threshold		0.65 DVDD	0.75 DVDD	V
V_{T-}	Negative-Going Threshold		0.25 DVDD	0.4 DVDD	V
$t_{RSTPD,p}$	Propagation Delay			20	µs

Table 19-6: RSTPD Input Pin Characteristics

The analog input pin RX has the following input capacitance:

SYMBOL	PARAMETER	CONDITIONS	MIN	MAX	UNIT
C_{RX}	Input Capacitance			15	pF

Table 19-7: RX Input Capacitance

19.4.2 DIGITAL OUTPUT PIN CHARACTERISTICS

Pins D0 to D7, SIGOUT and IRQ have TTL output characteristics and behave as defined in the following table.

SYMBOL	PARAMETER	CONDITIONS	MIN	TYP	MAX	UNIT
V_{OH}	Output Voltage HIGH	DVDD = 5 V, I_{OH} = -1 mA	2.4	4.9		V
		DVDD = 5 V, I_{OH} = -10 mA	2.4	4.2		V
V_{OL}	Output Voltage LOW	DVDD = 5 V, I_{OL} = 1 mA		25	400	mV
		DVDD = 5 V, I_{OL} = 10 mA		250	400	mV
I_O	Output Current source or sink	DVDD = 5 V			10	mA

Table 19-8:Digital Output Pin Characteristics

Note: IRQ pin may also be configured as open collector. In that case the values for V_{OH} do not apply.

19.4.3 ANTENNA DRIVER OUTPUT PIN CHARACTERISTICS

The source conductance of the antenna driver pins TX1 and TX2 for driving the HIGH level can be configured via *GsCfgCW* in the *CwConductance Register*, while their source conductance for driving the LOW level is constant.

For the default configuration, the output characteristic is specified below:

SYMBOL	PARAMETER	CONDITIONS	MIN	TYP	MAX	UNIT
V_{OH}	Output Voltage HIGH	TVDD = 5.0 V, I_{OL} = 20 mA		4.97		V
		TVDD = 5.0 V, I_{OL} = 100 mA		4.85		V
V_{OL}	Output Voltage LOW	TVDD = 5.0 V, I_{OL} = 20 mA		30		mV
		TVDD = 5.0 V, I_{OL} = 100 mA		150		mV
I_{TX}	Transmitter Output Current	Continuous Wave			200	mA_{peak}

Table 19-9:Antenna Driver Output Pin Characteristics

19.5 AC Electrical Characteristics

19.5.1 AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' for time. The other characters indicate the name of a signal or the logic state of that signal (depending on position):

Designation:	Signal:	Designation:	Logic Level:
A	address	H	HIGH
D	data	L	LOW
W	NWR or nWait	Z	high impedance
R	NRD or R/NW or nWrite	X	any level or data
L	ALE or AS	V	any valid signal or data
C	NCS		
S	NDS or nDStrb and nAStrb		

<u>Example</u>: t_{AVLL} = time for address valid to ALE low

19.5.2 AC OPERATING SPECIFICATION

19.5.2.1 Bus Timing for Separated Read/Write Strobe

SYMBOL	PARAMETER	MIN	MAX	UNIT
t_{LHLL}	ALE pulse width	20		ns
t_{AVLL}	Multiplexed Address Bus valid to ALE low (Address Set Up Time)	15		ns
t_{LLAX}	Multiplexed Address Bus valid after ALE low (Address Hold Time)	8		ns
t_{LLWL}	ALE low to NWR, NRD low	15		ns
t_{CLWL}	NCS low to NRD, NWR low	0		ns
t_{WHCH}	NRD, NWR high to NCS high	0		ns
t_{RLDV}	NRD low to DATA valid		65	ns
t_{RHDZ}	NRD high to DATA high impedance		20	ns
t_{WLDV}	NWR low to DATA valid		35	ns
t_{WHDX}	DATA hold after NWR high (Data Hold Time)	8		ns
t_{WLWH}	NRD, NWR pulse width	65		ns
t_{AVWL}	Separated Address Bus valid to NRD, NWR low (Set Up Time)	30		ns
t_{WHAX}	Separated Address Bus valid after NWR high (Hold Time)	8		ns
t_{WHWL}	period between sequenced read / write accesses	150		ns

Table 19-10: Timing Specification for Separated Read/Write Strobe

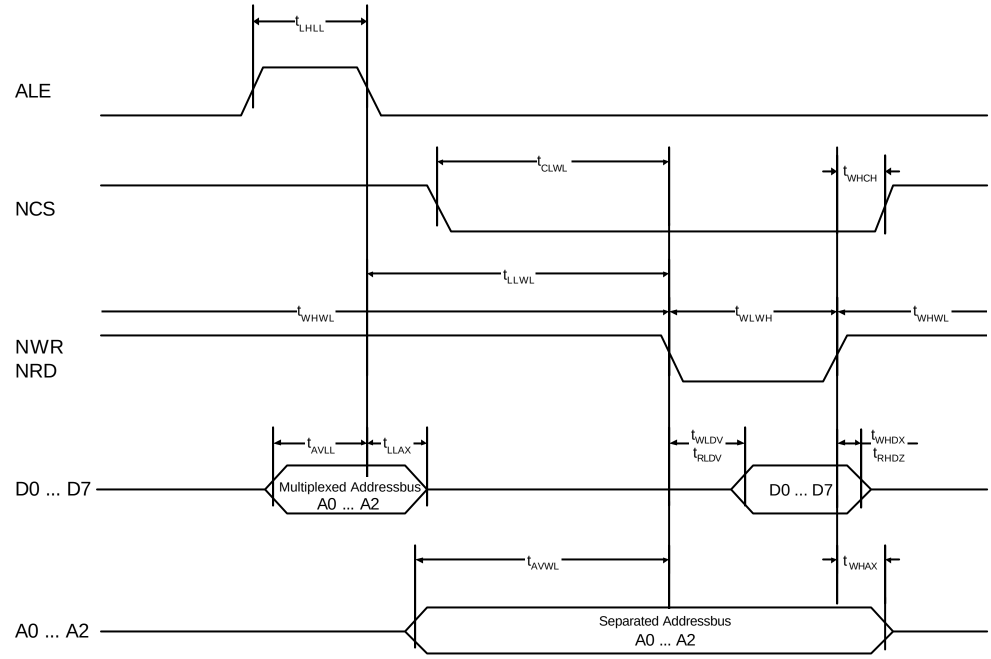


Figure 19-1: Timing Diagram for Separated Read/Write Strobe

Note: For separated address and data bus the signal ALE is not relevant and the multiplexed addresses on the data bus don't care.
For the multiplexed address and data bus the address lines A0 to A2 have to be connected as described in 4.3.

19.5.2.2 Bus Timing for Common Read/Write Strobe

SYMBOL	PARAMETER	MIN	MAX	UNIT
t_{LHLL}	AS pulse width	20		ns
t_{AVLL}	Multiplexed Address Bus valid to AS low (Address Set Up Time)	15		ns
t_{LLAX}	Multiplexed Address Bus valid after AS low (Address Hold Time)	8		ns
t_{LLSL}	AS low to NDS low	15		ns
t_{CLSL}	NCS low to NDS low	0		ns
t_{SHCH}	NDS high to NCS high	0		ns
$t_{SLDV,R}$	NDS low to DATA valid (for read cycle)		65	ns
t_{SHDZ}	NDS low to DATA high impedance (read cycle)		20	ns
$t_{SLDV,W}$	NDS low to DATA valid (for write cycle)		35	ns
t_{SHDX}	DATA hold after NDS high (write cycle, Hold Time)	8		ns
t_{SHRX}	R/NW hold after NDS high	8		ns
t_{SLSH}	NDS pulse width	65		ns
t_{AVSL}	Separated Address Bus valid to NDS low (Hold Time)	30		ns
t_{SHAX}	Separated Address Bus valid after NDS high (Set Up Time)	8		ns
t_{SHSL}	period between sequenced read/write accesses	150		ns
t_{RVSL}	R/NW valid to NDS low	8		ns

Table 19-11: Timing Specification for Common Read/Write Strobe

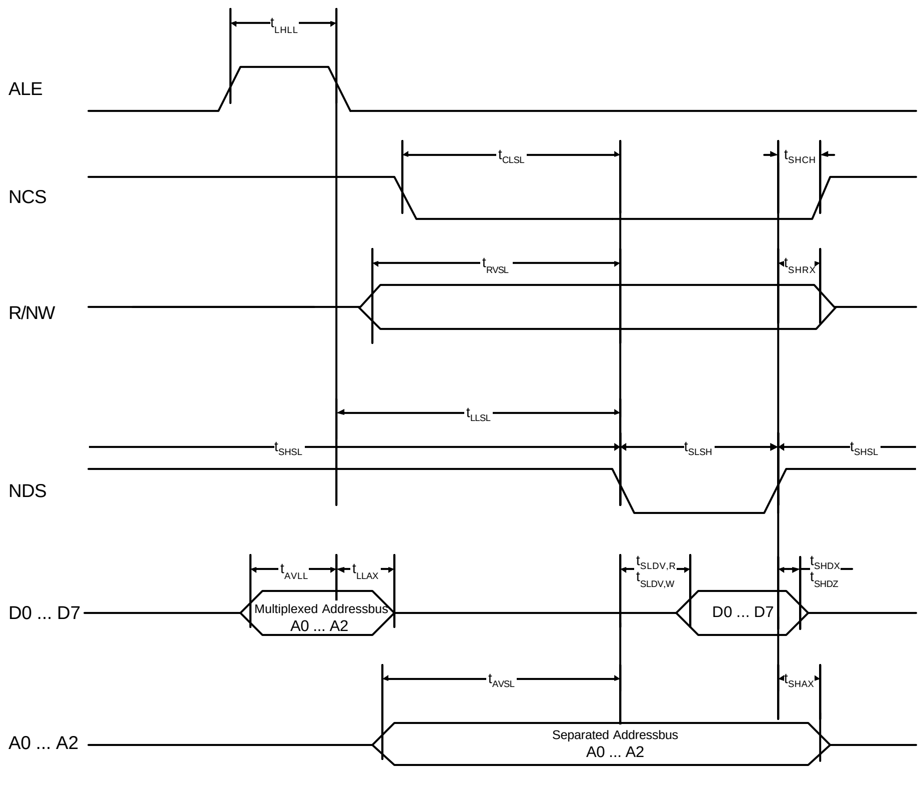


Figure 19-2: Timing Diagram for Common Read/Write Strobe

<u>Note:</u> For separated address and data bus the signal ALE is not relevant and the multiplexed addresses on the data bus don't care. For the multiplexed address and data bus the address lines A0 to A2 have to be connected as described in 4.3.

19.5.2.3 Bus Timing for EPP

SYMBOL	PARAMETER	MIN	MAX	UNIT
t_{LLLH}	nAStrb pulse width	20		ns
t_{AVLH}	Multiplexed Address Bus valid to nAStrb high (Set Up Time)	15		ns
t_{LHAX}	Multiplexed Address Bus valid after nAStrb high (Hold Time)	8		ns
t_{CLSL}	NCS low to nDStrb low	0		ns
t_{SHCH}	nDStrb high to NCS high	0		ns
$t_{SLDV,R}$	nDStrb low to DATA valid (read cycle)		65	ns
t_{SHDZ}	nDStrb low to DATA high impedance (read cycle)		20	ns
$t_{SLDV,W}$	nDStrb low to DATA valid (write cycle, Set up Time)		35	ns
t_{SHDX}	DATA hold after nDStrb high (write cycle, Hold Time)	8		ns
t_{SHRX}	nWrite hold after nDStrb high	8		ns
t_{SLSH}	nDStrb pulse width	65		ns
t_{RVSL}	nWrite valid to nDStrb low	8		ns
t_{SLWH}	nDStrb low to nWait high		75	ns
t_{SHWL}	nDStrb high to nWait low		75	ns

Table 19-12: Timing Specification for Common Read/Write Strobe

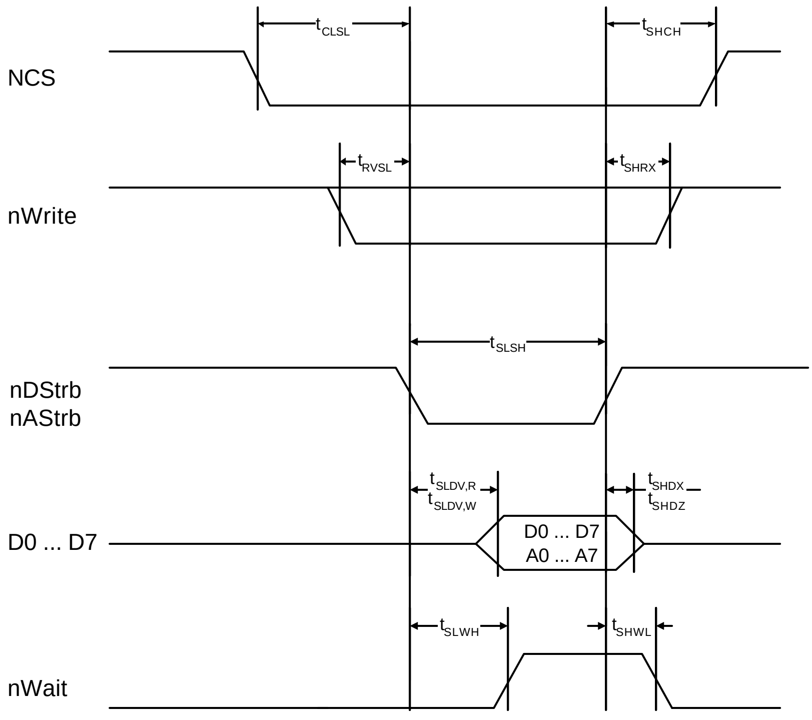


Figure 19-3: Timing Diagram for Common Read/Write Strobe

Remark: The figure does not distinguish between the Address Write Cycle and a Data Write Cycle. Take in account, that timings for the Address Write and Data Write Cycle different. For the EPP-Mode the address lines A0 to A2 have to be connected as described in 4.3.

19.5.3 CLOCK FREQUENCY

The clock input is pin 1, OSCIN.

PARAMETER	SYMBOL	MIN	TYP	MAX	UNIT
Clock Frequency (checked by the clock filter)	f_{OSCIN}		13.56		MHz
Duty Cycle of Clock Frequency	d_{FEC}	40	50	60	%
Jitter of Clock Edges	t_{jitter}			10	ps

The clock applied to the SL RC400 acts as time basis for the coder and decoder of the synchronous system. Therefore stability of clock frequency is an important factor for proper performance. To obtain highest performance, clock jitter shall be as small as possible. This is best achieved using the internal oscillator buffer with the recommended circuitry (see 12).

20 E²PROM CHARACTERISTICS

The E²PROM has a size of 8x16x8 = 1.024 bit.

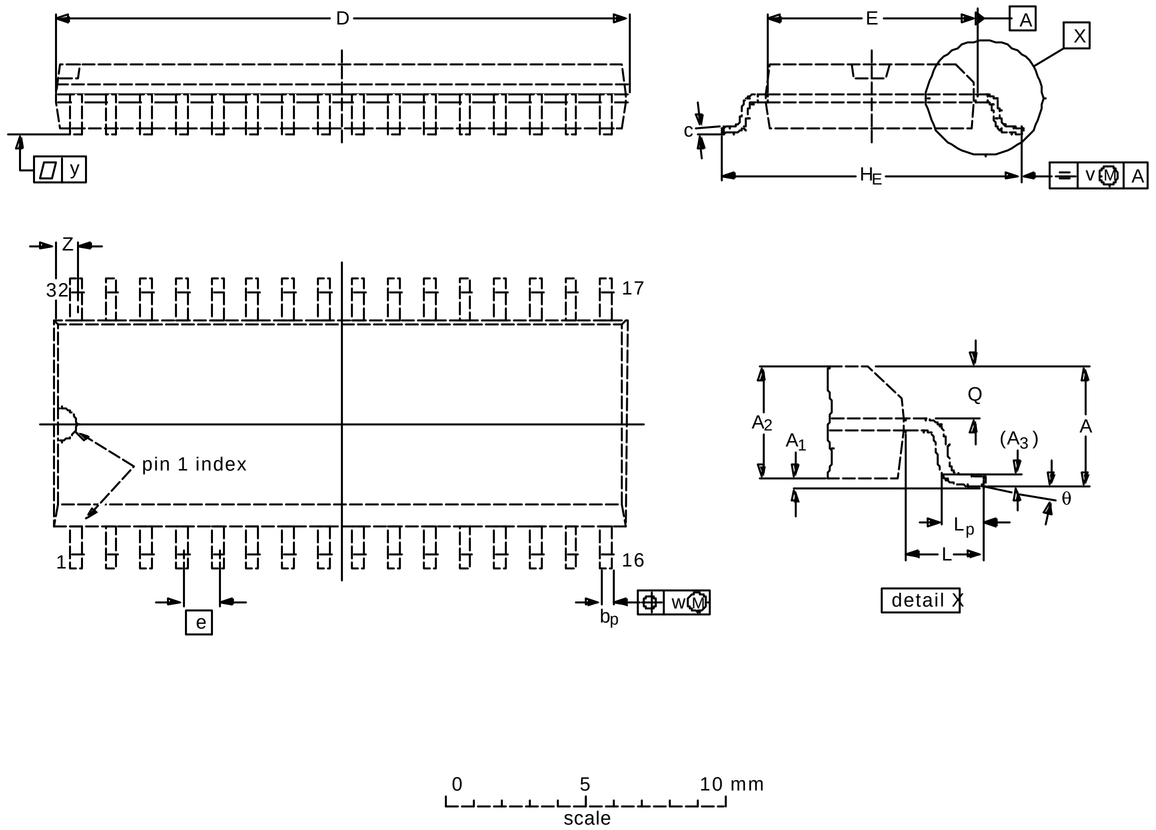
SYMBOL	PARAMETER	CONDITIONS	MIN	MAX	UNIT
$t_{EEEndurance}$	Data Endurance		100.000		erase/write cycles
$t_{EERetention}$	Data Retention	$T_{amb} \leq 55°C$	10		years
$t_{EEErase}$	Erase Time			2.9	ms
$t_{EEWrite}$	Write Time			2.9	ms

Table 20-1:E²PROM Characteristics

21 PACKAGE OUTLINES

21.1 SO32

SO32: plastic small outline package; 32 leads; body width 7.5 mm **SOT287-1**



DIMENSIONS (inch dimensions are derived from the original mm dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_E	L	L_p	Q	v	w	y	$Z^{(1)}$	θ
mm	2.65	0.3 0.1	2.45 2.25	0.25	0.49 0.36	0.27 0.18	20.7 20.3	7.6 7.4	1.27	10.65 10.00	1.4	1.1 0.4	1.2 1.0	0.25	0.25	0.1	0.95 0.55	8^o 0^o
inches	0.10	0.012 0.004	0.096 0.086	0.01	0.02 0.01	0.011 0.007	0.81 0.80	0.30 0.29	0.050	0.419 0.394	0.055	0.043 0.016	0.047 0.039	0.01	0.01	0.004	0.037 0.022	

Note

1. Plastic or metal protrusions of 0.15 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT287-1						~~95-01-25~~ 97-05-22

Figure 21-1: Outline and Dimension of SL RC400 in SO32

22 TERMS AND ABBREVIATIONS

Designation:	Description:
µ-Processor	Micro Processor
E^2PROM	Electrically Erasable Programmable Read Only Memory
EOF	End of Frame
FWT	Frame Waiting Time: maximum time delay between last bit transmitted by the reader and first bit received from the label's response.
I?CODE	A family of hard-wired logic contactless label ICs. The protocol of these labels is according to I•CODE1 and ISO 15693. On top they use a fixed set of commands.
POR	Power On Reset: triggers a reset, caused by a rising edge on a supply pin.
ROM	Read Only Memory
SOF	Start of Frame

23 DEFINITIONS

Data sheet status	
Objective specification	This data sheet contains target or goal specifications for product development.
Preliminary specification	This data sheet contains preliminary data; supplementary data may be published later.
Product specification	This data sheet contains final product specifications.
Limiting values	
Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics section of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.	
Application information	
Where application information is given, it is advisory and does not form part of the specification.	

24 LIFE SUPPORT APPLICATIONS

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips customers using or selling these products for use in such applications do so on their own risk and agree to fully indemnify Philips for any damages resulting from such improper use or sale.

25 REVISION HISTORY

REVISION	DATE	CPCN	PAGE	DESCRIPTION
1.0				First published version
2.0	14.11.01			Preliminary version

Table 25-1: Document Versions Up to Revision 1.0

Philips Semiconductors - a worldwide company

Argentina: see South America
Australia: 34 Waterloo Road, NORTHRYDE, NSW 2113,
Tel. +612 9805 4455, Fax. +612 9805 4466
Austria: Computerstraße 6, A-1101 WIEN, P.O.Box 213,
Tel. +431 60 101, Fax. +431 30 101 1210
Belarus: Hotel Minsk Business Centre, Bld. 3, r.1211, Volodarski Str. 6,
220050 MINSK, Tel. +375172 200 733, Fax. +375172 200 773
Belgium: see The Netherlands
Brazil: see South America
Bulgaria: Philips Bulgaria Ltd., Energoproject, 15th floor,
51 James Bourchier Blvd., 1407 SOFIA
Tel. +3592 689 211, Fax. +3592 689 102
Canada: Philips Semiconductors/Components,
Tel. +1800 234 7381
China/Hong Kong: 501 Hong Kong Industrial Technology Centre,
72 Tat Chee Avenue, Kowloon Tong, HONG KONG,
Tel. +85223 19 7888, Fax. +85223 19 7700
Colombia: see South America
Czech Republic: see Austria
Denmark: Prags Boulevard 80, PB 1919, DK-2300 COPENHAGEN S,
Tel. +4532 88 2636, Fax. +4531 57 1949
Finland: Sinikalliontie 3, FIN-02630 ESPOO,
Tel. +3589 61 5800, Fax. +3589 61 58rfuxx
France: 4 Rue du Port-aux-Vins, BP 317, 92156 SURESNES Cedex,
Tel. +331 40 99 6161, Fax. +331 40 99 6427
Germany: Hammerbrookstraße 69, D-20097 HAMBURG,
Tel. +4940 23 53 60, Fax. +4940 23 536 300
Greece: No. 15, 25th March Street, GR 17778 TAVROS/ATHENS,
Tel. +301 4894 339/239, Fax. +301 4814 240
Hungary: see Austria
India: Philips INDIA Ltd., Shivsagar Estate, A Block, Dr. Annie Besant Rd.
Worli, MUMBAI 400018, Tel. +9122 4938 541, Fax. +9122 4938 722
Indonesia: see Singapore
Ireland: Newstead, Clonskeagh, DUBLIN 14,
Tel. +3531 7640 000, Fax. +3531 7640 200
Israel: RAPAC Electronics, 7 Kehilat Saloniki St., TEL AVIV 61180,
Tel. +9723 645 0444, Fax. +9723 649 1007
Italy: Philips Semiconductors, Piazza IV Novembre 3,
20124 MILANO, Tel. +392 6752 2531, Fax. +392 6752 2557
Japan: Philips Bldg. 13-37, Kohnan 2-chome, Minato-ku, TOKYO 108,
Tel. +813 3740 5130,Fax. +813 3740 5077
Korea: Philips House, 260-199, Itaewon-dong, Yonsan-ku, SEOUL,
Tel. +822 709 1412, Fax. +822 709 1415
Malaysia: No. 76 Jalan Universiti, 46200 PETALING JAYA, Selangor,
Tel. +60 3750 5214, Fax. +603 757 4880
Mexico: 5900 Gateway East, Suite 200, EL PASO, Texas 79905,
Tel. +9 5800 234 7381
Middle East: see Italy
Netherlands: Postbus 90050, 5600 PB EINDHOVEN, Bldg. VB,
Tel. +3140 27 82785, Fax +3140 27 88399
New Zealand: 2 Wagener Place, C.P.O. Box 1041, AUCKLand,
Tel. +649 849 4160, Fax. +649 849 7811
Norway: Box 1, Manglerud 0612, OSLO,
Tel. +4722 74 8000, Fax. +4722 74 8341
Philippines: Philips Semiconductors Philippines Inc.,
106 Valero St. Salcedo Village, P.O.Box 2108 MCC, MAKATI,
Metro MANILA, Tel. +632 816 6380, Fax. +632 817 3474
Poland: Ul. Lukiska 10, PL 04-123 WARSZWA,
Tel. +4822 612 2831, Fax. +4822 612 2327
Portugal: see Spain
Romania: see Italy
Russia: Philips Russia, Ul. Usatcheva 35A, 119048 MOSCOW,
Tel. +7095 247 9145, Fax. +7095 247 9144
Singapore: Lorong 1, Toa Payoh, SINGAPORE 1231,
Tel. +65350 2538, Fax. +65251 6500
Slovakia: see Austria
Slovenia: see Italy
South Africa: S.A. Philips Pty Ltd., 195-215 Main Road Martindale,
2092 JOHANNESBURG, P.O.Box 7430 Johannesburg 2000,
Tel. +2711 470 5911, Fax. +2711 470 5494
South America: Rua do Rocio 220, 5th floor, Suite 51,
04552-903 Sao Paulo, SAO PAULO - SP, Brazil,
Tel. +5511 821 2333, Fax. +5511 829 1849
Spain: Balmes 22, 08007 BARCELONA,
Tel. +343 301 6312, Fax. +343 301 4107
Sweden: Kottbygatan 7, Akalla, S-16485 STOCKHOLM,
Tel. +468 632 2000, Fax. +468 632 2745
Switzerland: Allmendstraße 140, CH-8027 ZÜRICH,
Tel. +411 488 2686, Fax. +411 481 7730
Taiwan: Philips Taiwan Ltd., 2330F, 66,
Chung Hsiao West Road, Sec. 1, P.O.Box 22978,
TAIPEI 100, Tel. +8862 382 4443, Fax. +8862 382 4444
Thailand: Philips Electronics (Thailand) Ltd.,
209/2 Sanpavuth-Bangna Road Prakanong, BANGKOK 10260,
Tel. +662 745 4090, Fax. +662 398 0793
Turkey: Talapasa Cad. No. 5, 80640 GÜLTEPE/ISTANBUL,
Tel. +90212 279 2770, Fax. +90212 282 6707
Ukraine: Philips Ukraine, 4 Patrice Lumumba Str., Building B, Floor 7,
252042 KIEV, Tel. +38044 264 2776, Fax. +38044 268 0461
United Kingdom: Philips Semiconductors Ltd., 276 Bath Road, Hayes,
MIDDLESEX UM3 5BX, Tel. +44181 730 5000, Fax. +44181 754 8421
United States: 811 Argues Avenue, SUNNYVALE, CA94088-3409,
Tel. +1800 234 7381
Uruguay: see South America
Vietnam: see Singapore
Yugoslavia: Philips, Trg N. Pasica 5/v, 11000 BEOGRAD,
Tel. +38111 625 344, Fax. +38111 635 777

Published by:

Philips Semiconductors Gratkorn GmbH, Mikron-Weg 1, A-8101 Gratkorn, Austria Fax: +43 3124 299 - 270

For all other countries apply to: Philips Semiconductors, Marketing & Sales Communications, Internet: http://www.semiconductors.philips.com
Building BE-p, P.O.Box 218, 5600 MD EINDHOVEN, The Netherlands, Fax: +3140 27 24825

Philips Semiconductors

PHILIPS