Data Sheet, DS 1, Sep. 2000

SEROCCO-D

2 Channel Serial Optimized Communication Controller with DMA

PEB 20542 Version 1.2
PEF 20542 Version 1.2

Datacom

Never stop thinking.

Edition 2000-09-14

Published by Infineon Technologies AG,
St.-Martin-Strasse 53,
D-81541 München, Germany

Attention please!

The information herein is given to describe certain components and shall not be considered as warranted characteristics.

Terms of delivery and rights to technical change reserved.

We hereby disclaim any and all warranties, including but not limited to warranties of non-infringement, regarding circuits, descriptions and charts stated herein.

Infineon Technologies is an approved CECC manufacturer.

Information

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office in Germany or our Infineon Technologies Representatives worldwide (see address list).

Warnings

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

Data Sheet, DS 1, Sep. 2000

SEROCCO-D

2 Channel Serial Optimized Communication Controller with DMA

PEB 20542 Version 1.2
PEF 20542 Version 1.2

Datacom

Never stop thinking.

Revision History:		**2000-09-14**	DS 1
Previous Version:		MISTRAL V1.1 Preliminary Data Sheet, 08.99, DS1	
Page (previous Version)	Page (current Version)	Subjects (major changes since last revision)	
34-36	36-38	Correction: signal ’OSR’ is multiplexed with signal ’CD’, signal ’OST’ is multiplexed with ’$\overline{\text{CTS}}$’ (was vice versa)	
85	87	corrected HDLC receive address recognition table	
218, 226	222, 230	Corrected location of TCD interrupt (async/bisync modes only) in registers ISR0 and IMR0 from bit 7 to bit 2.	
-	-	removed referneces to Intel Multiplexed Mode	
268	272	Chapter "Electrical Characteristics" updated with final characterization results.	

For questions on technology, delivery and prices please contact the Infineon Technologies Offices in Germany or the Infineon Technologies Companies and Representatives worldwide: see our webpage at http://www.infineon.com

Table of Contents **Page**

Table of Contents **Page**

Table of Contents **Page**

Table of Contents Page

List of Figures

List of Figures

List of Figures **Page**

List of Tables

List of Registers

Page

List of Registers Page

List of Registers **Page**

Preface

The 2 Channel Serial Optimized Communication Controller with DMA PEB 20542 (SEROCCO-D) is a Protocol Controller for a wide range of data communication and telecommunication applications. This document provides complete reference information on hardware and software related issues as well as on general operation.

Organization of this Document

This Data Sheet is divided into 9 chapters. It is organized as follows:

- **Chapter 1**, **Introduction**
  Gives a general description of the product, lists the key features, and presents some typical applications.

- **Chapter 2**, **Pin Descriptions**
  Lists pin locations with associated signals, categorizes signals according to function, and describes signals.

- **Chapter 3**, **Functional Overview**
  This chapter provides detailed descriptions of all SEROCCO-D internal functional blocks.

- **Chapter 4**, **Detailed Protocol Description**
  Gives a detailed description of all protocols supported by the serial communication controllers SCCs.

- **Chapter 5**, **Register Description**
  Gives a detailed description of all SEROCCO-D on chip registers.

- **Chapter 6**, **Programming**
  Provides programming help for SEROCCO-D initialization procedure and operation.

- **Chapter 7**, **Electrical Characteristics**
  Gives a detailed description of all electrical DC and AC characteristics and provides timing diagrams and values for all interfaces.

- **Chapter 8**, **Test Modes**
  Gives a detailed description of the JTAG boundary scan unit.

- **Chapter 9**, **Package Outlines**

Your Comments

We welcome your comments on this document. We are continuously trying improving our documentation. Please send your remarks and suggestions by e-mail to

sc.docu_comments@infineon.com

Please provide in the *subject* of your e-mail:
device name (SEROCCO-D), device number (PEB 20542, PEF 20542), device version (Version 1.2),

and in the *body* of your e-mail:
document type (Data Sheet), issue date (2000-09-14) and document revision number (DS 1).

1 Introduction

The SEROCCO-D is a DMA Integrated Serial Communication Controller with two independent serial channels[1)]. The serial channels are derived from updated protocol logic of the ESCC and DSCC4 device family providing a large set of protocol support and variety in serial interface configuration. This allows easy integration to different environments and applications.

A generic 8- or 16-bit demultiplexed master/slave interface provides fast device access with low bus utilization and easy software handshaking. The internal DMA controller is optimized for a minimum CPU intervention. Different control mechanisms allow easy software development well adapted to the needs of special applications (e.g. frame/ packet oriented and continuous transmission/reception).

Large on-chip FIFOs of 64 byte capacity per port and direction in combination with enhanced threshold control mechanisms allow decoupling of traffic requirements on host bus and serial interfaces with little exception probabilities such as data underruns or overflows.

Each of the two Serial Communication Controllers (SCC) contains an independent Baud Rate Generator, DPLL and programmable protocol processing (HDLC, PPP, ASYNC and BISYNC). Data rates of up to 16 Mbit/s (HDLC, PPP, bit transparent) and 2 Mbit/s (DPLL assisted modes) are supported. The channels can also handle a large set of layer-2 protocol functions (LAPD, SS7) reducing bus and host CPU load. Two channel specific timers are provided to support protocol functions.

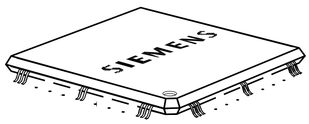
1) The serial channels are also called ’ports’ or ’cores’ depending on the context.

2 Channel Serial Optimized Communication Controller with DMA
SEROCCO-D

PEB 20542
PEF 20542

Version 1.2 **CMOS**

1.1 Features



P-TQFP-144-10

Serial communication controllers (SCCs)

- Two independent channels
- Full duplex data rates on each channel of up to 16 Mbit/s sync - 2 Mbit/s with DPLL
- 64 Bytes deep receive FIFO per SCC
- 64 Bytes deep transmit FIFO per SCC

Serial Interface

- On-chip clock generation or external clock sources
- On-chip DPLLs for clock recovery
- Baud rate generator
- Clock gating signals
- Clock gapping capability
- Programmable time-slot capability for connection to TDM interfaces (e.g. T1, E1)
- NRZ, NRZI, FM and Manchester data encoding

- Optional data flow control using modem control lines ($\overline{\text{RTS}}$, $\overline{\text{CTS}}$, CD)
- Support of bus configuration by collision detection and resolution

Bit Processor Functions

- HDLC/SDLC Protocol Modes
  - Automatic flag detection and transmission
  - Shared opening and closing flag
  - Generation of interframe-time fill ’1’s or flags
  - Detection of receive line status
  - Zero bit insertion and deletion

Type	Package
PEB 20542, PEF 20542	P-TQFP-144-10

  – CRC generation and checking (CRC-CCITT or CRC-32)
  – Transparent CRC option per channel and/or per frame
  – Programmable Preamble (8 bit) with selectable repetition rate
  – Error detection (abort, long frame, CRC error, short frames)
- Bit Synchronous PPP Mode
  – Bit oriented transmission of HDLC frame (flag, data, CRC, flag)
  – Zero bit insertion/deletion
  – 15 consecutive ’1’ bits abort sequence
- Octet Synchronous PPP Mode
  – Octet oriented transmission of HDLC frame (flag, data, CRC, flag)
  – Programmable character map of 32 hard-wired characters (00_H-$1F_H$)
  – Four programmable characters for additional mapping
  – Insertion/deletion of control-escape character ($7D_H$) for mapped characters
- Asynchronous PPP Mode
  – Character oriented transmission of HDLC frame (flag, data, CRC, flag)
  – Start/stop bit framing of single character
  – Programmable character map of 32 hard-wired characters (00_H-$1F_H$)
  – Four programmable characters for additional mapping
  – Insertion/deletion of control-escape character ($7D_H$) for mapped characters
- Asynchronous (ASYNC) Protocol Mode
  – Selectable character length (5 to 8 bits)
  – Even, odd, forced or no parity generation/checking
  – 1 or 2 stop bits
  – Break detection/generation
  – In-band flow control by XON/XOFF
  – Immediate character insertion
  – Termination character detection for end of block identification
  – Time out detection
  – Error detection (parity error, framing error)
- BISYNC Protocol Mode
  – Programmable 6/8 bit SYN pattern (MONOSYNC)
  – Programmable 12/16 bit SYN pattern (BISYNC)
  – Selectable character length (5 to 8 bits)
  – Even, odd, forced or no parity generation/checking
  – Generation of interframe-time fill ’1’s or SYN characters
  – CRC generation (CRC-16 or CRC-CCITT)
  – Transparent CRC option per channel and/or per frame
  – Programmable Preamble (8 bit) with selectable repetition rate
  – Termination character detection for end of block identification
  – Error detection (parity error, framing error)
- Extended Transparent Mode
  – Fully bit transparent (no framing, no bit manipulation)
  – Octet-aligned transmission and reception

- Protocol and Mode Independent
  – Data bit inversion
  – Data overflow and underrun detection
  – Timer

Protocol Support

- Address Recognition Modes
  – No address recognition (Address Mode 0)
  – 8-bit (high byte) address recognition (Address Mode 1)
  – 8-bit (low byte) or 16-bit (high and low byte) address recognition (Address Mode 2)
- HDLC Automode
  – 8-bit or 16-bit address generation/recognition
  – Support of LAPB/LAPD
  – Automatic handling of S- and I-frames
  – Automatic processing of control byte(s)
  – Modulo-8 or modulo-128 operation
  – Programmable time-out and retry conditions
  – SDLC Normal Response Mode (NRM) operation for slave
- Signaling System #7 (SS7) support
  – Detection of FISUs, MSUs and LSSUs
  – Unchanged Fill-In Signaling Units (FISUs) not forwarded
  – Automatic generation of FISUs in transmit direction (incl. sequence number)
  – Counting of errored signaling units

Integrated DMA Controller

- 4 independent DMA channels
- Optimized for minimum CPU intervention
- Efficient block-oriented data transfer
- Bus preemption
- Fragmented transmission/reception of data packets from/into multiple buffers
- Switched-Buffer mode for seamless update of buffer base address and size
- 24-bit adressable memory range
- Optional $\overline{\text{DTACK}}$/$\overline{\text{READY}}$ controlled cycles

Microprocessor Interface

- 8/16-bit bus interface
- De-multiplexed address/data bus
- Intel/Motorola style
- Asynchronous interface
- Maskable interrupts for each channel

General Purpose Port (GPP) Pins

General

- 3.3V power supply with 5V tolerant inputs
- Low power consumption
- Power safe features
- P-TQFP-144-10 Package (Thermal Resistance: R_{JA} = 39K/W)

1.2 Logic Symbol

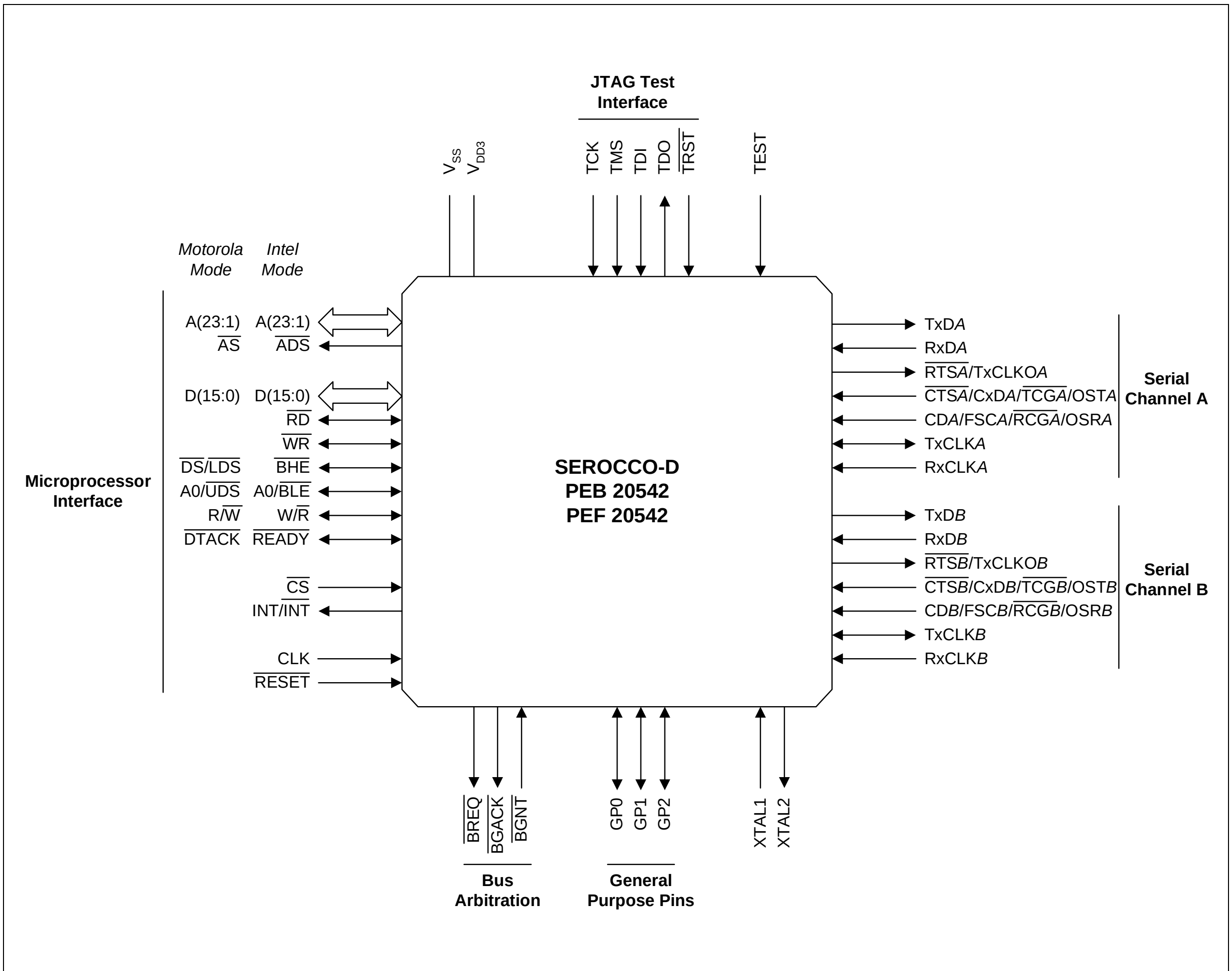


Figure 1 Logic Symbol

1.3 Typical Applications

SEROCCO-D devices can be used in LAN-WAN inter-networking applications such as Routers, Switches and Trunk cards and support the common V.35, ISDN BRI (S/T) and RFC1662 standards. Its new features provide powerful hardware and software interfaces to develop high performance systems.

1.3.1 System Integration Example

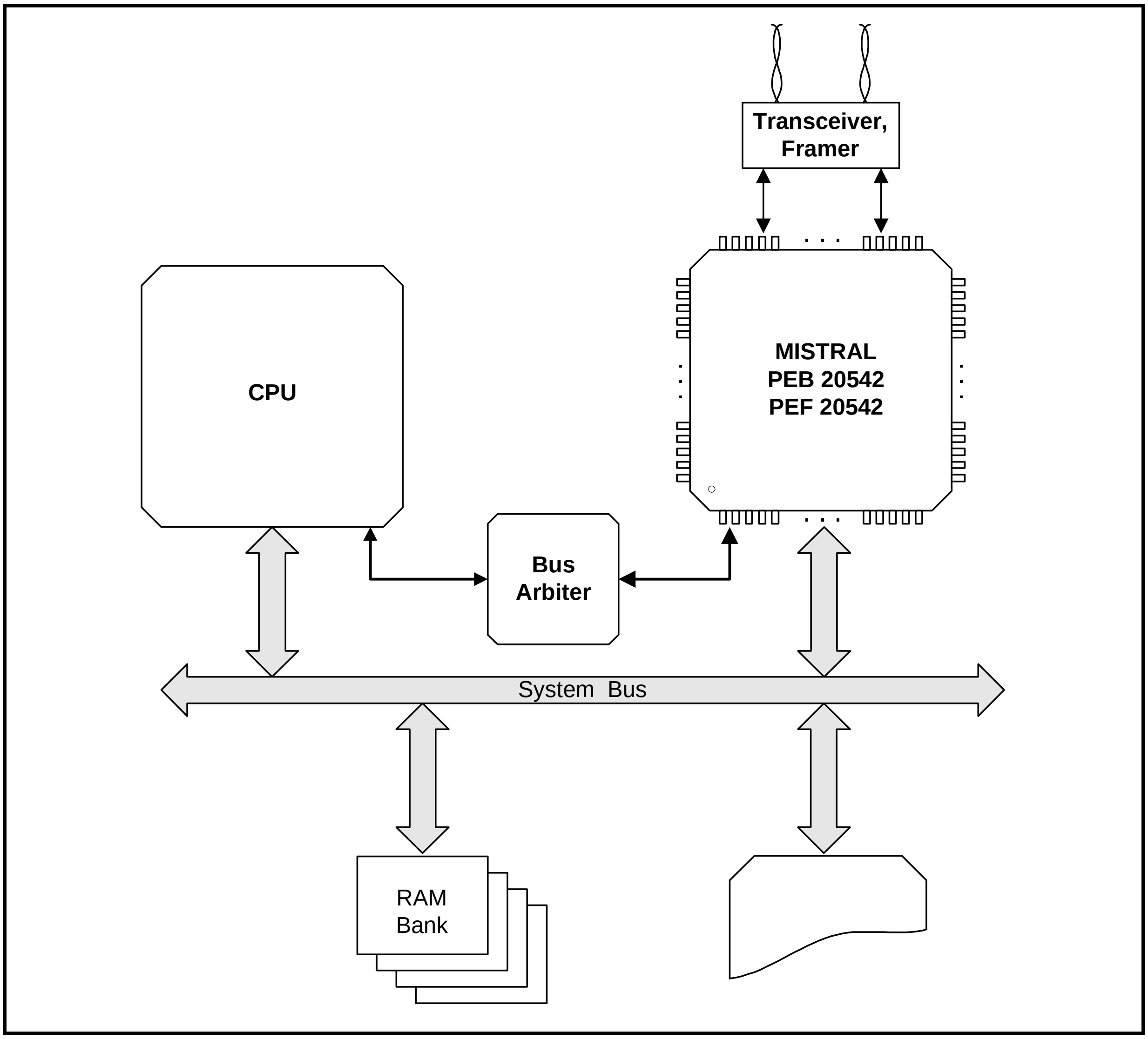


Figure 2 System Integration

1.3.2 Serial Configuration Examples

SEROCCO-D supports a variety of serial configurations at Layer-1 and Layer-2 level. The outstanding variety of clock modes supporting a large number of combinations of external and internal clock sources allows easy integration in application environments.

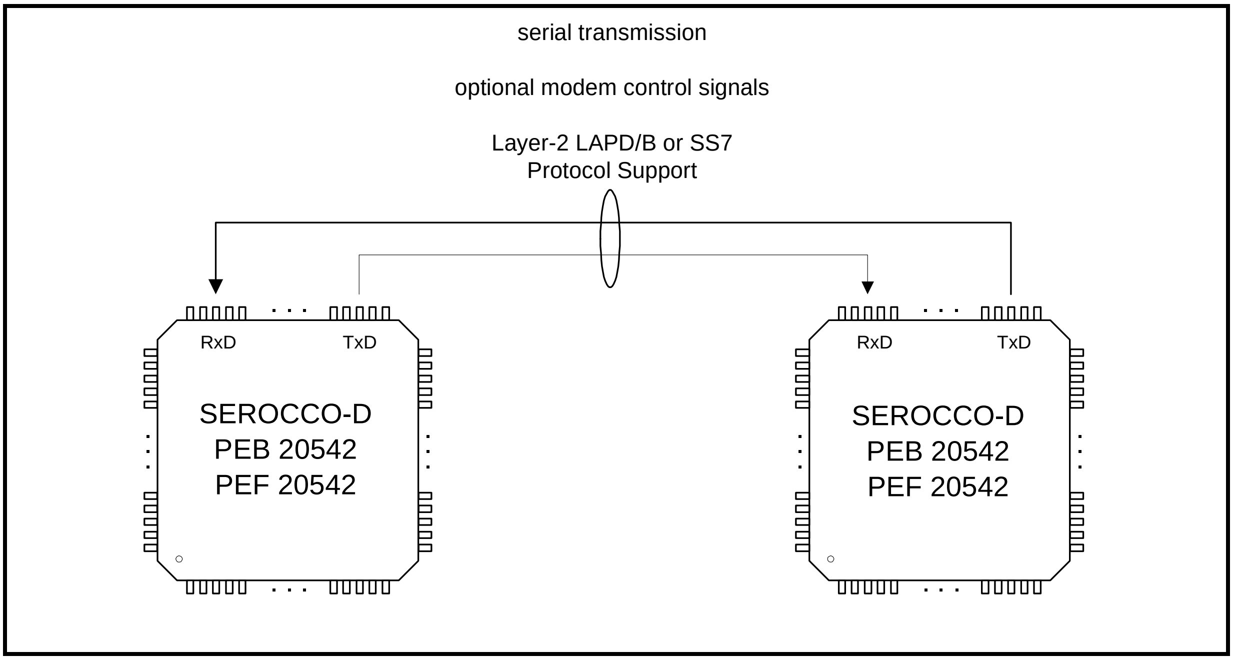


Figure 3 Point-to-Point Configuration

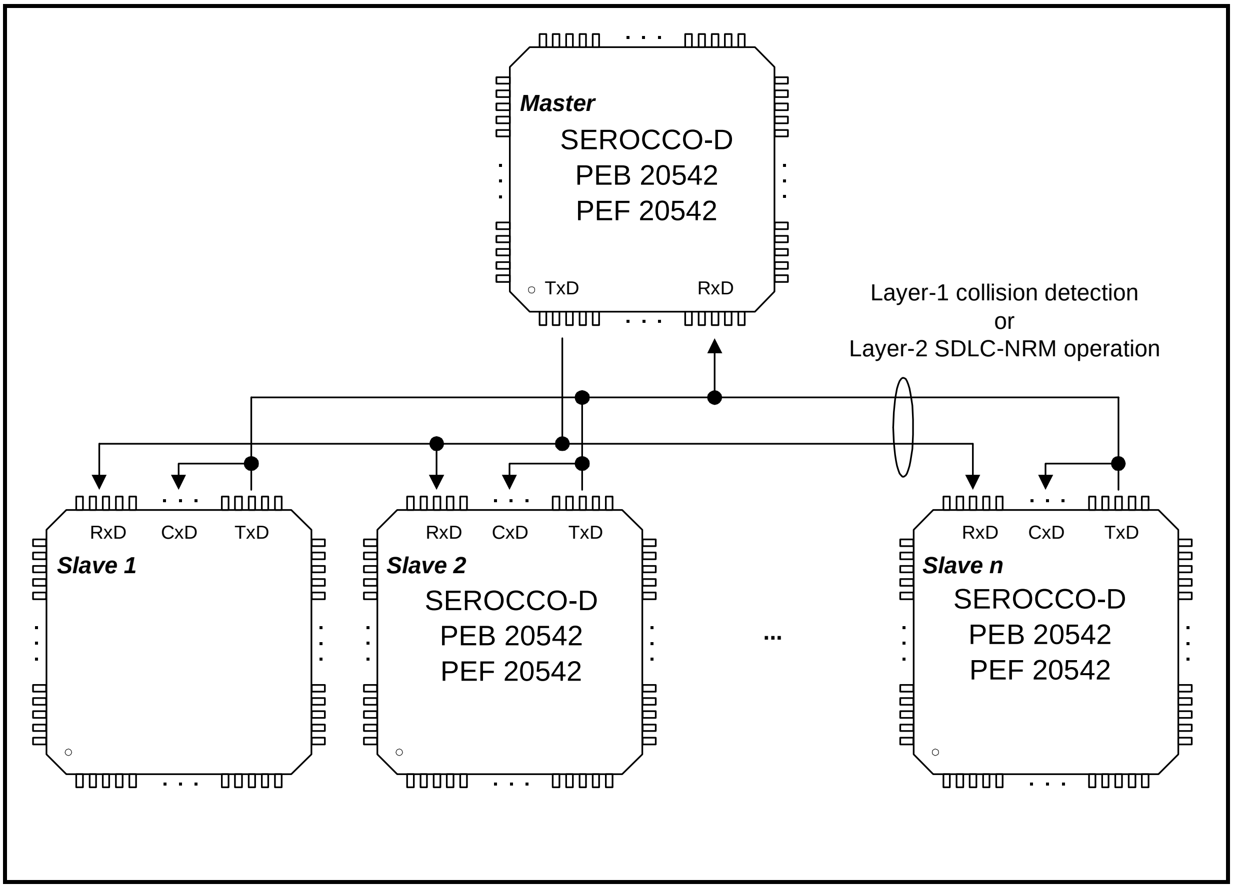


Figure 4 Point-to-Multipoint Bus Configuration

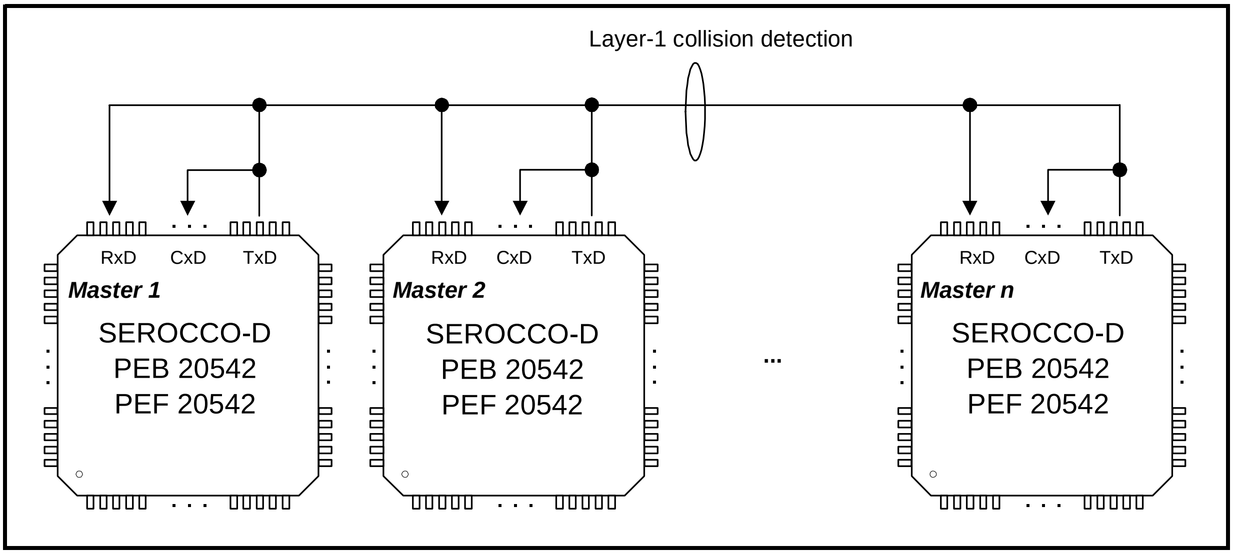


Figure 5 Multimaster Bus Configuration

1.4 Differences between SEROCCO-D and the ESCC Family

This chapter is useful for all being familiar with the ESCC family.

1.4.1 Enhancements to the ESCC Serial Core

The SEROCCO-D SCC cores contain the core logic of the ESCC as the heart of the device. Some enhancements are incorporated in the SCCs. These are:

- Integrated four-channel DMA controller
- Octet-, Bit Synchronous and Asynchronous PPP protocol support as in RFC-1662
- Signaling System #7 (SS7) support
- 4-kByte packet length byte counter
- Enhanced address filtering (16-bit maskable)
- Enhanced time slot assigner
- Support of high data rates (16 Mbit/s)

1.4.2 Simplifications to the ESCC Serial Core

The following features of the ESCC core have been removed:

- Extended transparent mode 0
  (this mode provided octet buffered data reception without usage of FIFOs; SEROCCO-D supports octet buffered reception via appropriate threshold configurations for the SCC receive FIFOs)
- Support of interrupt acknowledge cycles
- Multiplexed address/data bus in Infineon/Intel mode
- Master clock mode

2 Pin Descriptions

2.1 Pin Diagram P-TQFP-144-10

(top view)

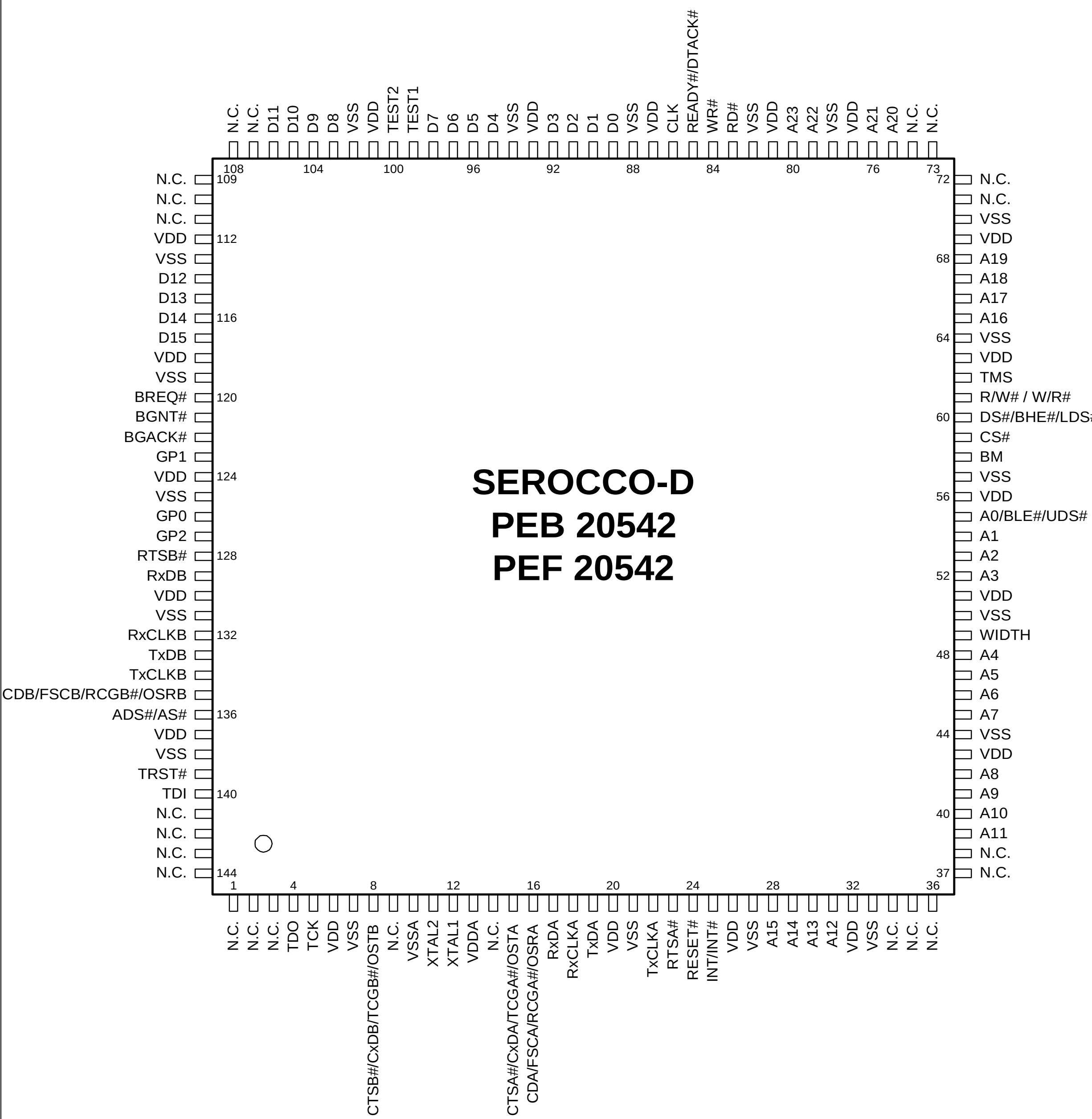


Figure 6 Pin Configuration P-TQFP-144-10 Package

2.2 Pin Definitions and Functions

Table 1 Microprocessor Bus Interface

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	117 116 115 114 106 105 104 103 98 97 96 95 92 91 90 89	D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0	I/O	**Data Bus** The data bus lines are bi-directional tri-state lines which interface with the system's data bus.

Table 1 Microprocessor Bus Interface

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	80	A23	O	**Address Bus** These pins connect to the system's address bus to select one of the internal registers for read or write. During operation of the internal DMA controller, these lines output the destination address when the bus is granted to the SEROCCO-D. *These lines are tri-state when unused.*
	79	A22	O	
	76	A21	O	
	75	A20	O	
	68	A19	O	
	67	A18	O	
	66	A17	O	
	65	A16	O	
	28	A15	O	
	29	A14	O	
	30	A13	O	
	31	A12	O	
	39	A11	O	
	40	A10	O	
	41	A9	O	
	42	A8	O	
	45	A7	I/O	
	46	A6	I/O	
	47	A5	I/O	
	48	A4	I/O	
	52	A3	I/O	
	53	A2	I/O	
	54	A1	I/O	
	55	A0	I/O	**Address Line A0 (8-bit modes)** In Motorola and in Intel 8-bit mode this signal represents the least significant address line.
		$\overline{\text{BLE}}$	I/O	**Byte Low Enable (16-bit Intel bus mode)** This signal indicates a data transfer on the lower byte of the data bus (D7..D0). Together with signal $\overline{\text{BHE}}$ the type of bus access is determined (byte or word access at even or odd address).
		$\overline{\text{UDS}}$	I/O	**Upper Data Strobe (16-bit Motorola bus mode)** This active low strobe signal serves to control read/write operations. Together with signal $\overline{\text{LDS}}$ the type of bus access is determined. *This line is tri-state when unused.*

Table 1 Microprocessor Bus Interface

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	58	BM	I	**Bus Mode** – BM = static '1' for operation in Motorola bus mode (de-multiplexed). – BM = static '0' for operation in Intel bus mode with de-multiplexed address and data buses. pins A(15:0)D(15:0)
	136	$\overline{\text{ADS}}$ $\overline{\text{AS}}$	O O	**Address Strobe (Intel Bus Mode)** **" (Motorola Bus Mode)** Indicates that the SEROCCO-D is driving a valid address and bus cycle definition on pins A(23:0), $\overline{\text{BHE}}$ (Intel mode) and R/$\overline{\text{W}}$. *This line is tri-state when unused.*
	60	$\overline{\text{DS}}$	I/O	**Data Strobe (8-bit Motorola bus mode only)** This active low strobe signal serves to control read/write operations.
		$\overline{\text{BHE}}$	I/O	**Bus High Enable (16-bit Intel bus mode only)** This signal indicates a data transfer on the upper byte of the data bus (D15..D8). In 8-bit Intel bus mode this signal has no function.
		$\overline{\text{LDS}}$	I/O	**Lower Data Strobe (16-bit Motorola bus mode)** This active low strobe signal serves to control read/write operations. Together with signal $\overline{\text{UDS}}$ the type of bus access is determined (byte or word access at even or odd address). *This line is tri-state when unused.* *In 8-bit Intel bus mode, a pull-up resistor to V_{DD3} is recommended on this pin.*
	83	$\overline{\text{RD}}$	I/O	**Read Strobe (Intel bus mode only)** This signal indicates a read operation. The current bus master is able to accept data on lines D(7:0) / D(15:0) during an active $\overline{\text{RD}}$ signal. *This line is tri-state when unused.* *In Motorola bus mode, a pull-up resistor to V_{DD3} is recommended on this pin.*

Table 1 Microprocessor Bus Interface

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	61	R/$\overline{\text{W}}$	I/O	**Read/Write Enable (Motorola bus mode)** This signal distinguishes between read and write operation. As an input it must be valid during data strobe ($\overline{\text{DS}}$).
		W/$\overline{\text{R}}$	O	**Write/Read Enable (Intel bus mode)** During master bus accesses from SEROCCO-D (DMA) this signal indicates the transaction direction (write/read). When SEROCCO-D is slave, this signal is not evaluated. *This line is tri-state when unused.* *If not used in Intel bus mode, a pull-up resistor to* V_{DD3} *is recommended on this pin.*
	59	$\overline{\text{CS}}$	I	**Chip Select** A low signal selects SEROCCO-D for read/write operations.
	84	$\overline{\text{WR}}$	I/O	**Write Strobe (Intel bus mode only)** This signal indicates a write operation. The current bus master presents valid data on lines D(7:0) / D(15:0) during an active $\overline{\text{WR}}$ signal. *This line is tri-state when unused.* *In Motorola bus mode, a pull-up resistor to* V_{DD3} *is recommended on this pin.*
	49	WIDTH	I	**Width Of Bus Interface** A low signal on this input selects the 8-bit bus interface mode. A high signal on this input selects the 16-bit bus interface mode. In this case word transfer to/from the internal registers is enabled. Byte transfers are implemented by using $\overline{\text{BLE}}$ and $\overline{\text{BHE}}$ (Intel bus mode) or $\overline{\text{LDS}}$ and $\overline{\text{UDS}}$ (Motorola bus mode)
	86	CLK	I	**Clock** The system clock for SEROCCO-D is provided through this pin.

Table 1 Microprocessor Bus Interface

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	25	INT/$\overline{\text{INT}}$	O o/d	**Interrupt Request** The INT/$\overline{\text{INT}}$ goes active when one or more of the bits in registers ISR0..ISR2 are set to ’1’. A read to these registers clears the interrupt. The INT/$\overline{\text{INT}}$ line is inactive when all interrupt status bits are reset. Interrupt sources can be unmasked in registers IMR0..IMR2 by setting the corresponding bits to ’0’.
	85	$\overline{\text{READY}}$ $\overline{\text{DTACK}}$	I/O I/O	**Ready (Intel bus mode)** **Data Transfer Acknowledge (Motorola mode)** During a slave access (register read/write) this signal (output) indicates, that the SEROCCO-D is ready for data transfer. The signal remains active until the data strobe ($\overline{\text{DS}}$ in Motorola bus mode, $\overline{\text{RD}}$/$\overline{\text{WR}}$ in Intel bus mode) and/or the chip select ($\overline{\text{CS}}$) go inactive. When the SEROCCO-D performs a DMA master access, the target may extend the read/write cycle using this signal (input) when enabled in register DCMDR. *This line is tri-state when unused.* *A pull-up resistor to V_{DD3} is recommended if this function is not used.*
	24	$\overline{\text{RESET}}$	I	**Reset** With this active low signal the on-chip registers and state machines are forced to reset state. During Reset all pins are in a high impedance state.

Table 2 Bus Arbitration

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	120	$\overline{\text{BREQ}}$	O o/d	**Bus Request** By asserting this signal to low, the SEROCCO-D requests master bus access from the external bus arbiter. As soon as the bus is granted to the SEROCCO-D and $\overline{\text{BGACK}}$ is asserted, this signal turns to inactive. If configured as open-drain output, bus preemption can be forced by the arbiter if it asserts this signal during the SEROCCO-D drives $\overline{\text{BGACK}}$ active (low). *A pull-up resistor < 1.5 kΩ to* V_{DD3} *must be connected to this pin if configured as open-drain output.*
	122	$\overline{\text{BGACK}}$	o/d	**Bus Grant Acknowledge** With this signal the SEROCCO-D indicates the period the bus is occupied for master read or write transfers of the internal DMA controller. *A pull-up resistor to* V_{DD3} *must be connected to this open-drain pin.*
	121	$\overline{\text{BGNT}}$	I	**Bus Grant** With this active-low input signal the bus arbiter grants the bus to SEROCCO-D. SEROCCO-D waits for $\overline{\text{BGACK}}$ to go high (indicating that the external bus master has released the bus) and takes over the bus by asserting $\overline{\text{BGACK}}$ low.

Table 3 Serial Port Pins

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	22	TxCLK A	I/O	**Transmit Clock Channel A** The function of this pin depends on the selected clock mode and the value of bit 'TOE' (CCR0L register, refer to **Table 8 "Clock Modes of the SCCs" on Page 49**). **If programmed as Input (CCR0L.TOE='0')**, either – the transmit clock for the channel (clock mode 0a, 2a, 4, 5b, 6a), or – a transmit strobe signal for the channel (clock mode 1) can be provided to this pin. **If programmed as Output (CCR0L.TOE='1')**, this pin supplies either – the transmit clock from the baud rate generator (clock mode 0b, 2b, 3b, 6b, 7b), or – the transmit clock from the DPLL circuit (clock mode 3a, 7a), or – an active-low control signal marking the programmed transmit time-slot in clock mode 5a.
	18	RxCLK A	I	**Receive Clock Channel A** The function of this pin depends on the selected clock mode (refer to **Table 8 "Clock Modes of the SCCs" on Page 49**). A signal provided on pin RxCLKA may supply – the receive clock (clock mode 0, 4, 5b), or – the receive and transmit clock (clock mode 1, 5a), or – the clock input for the baud rate generator (clock mode 2, 3).

Table 3 Serial Port Pins (cont'd)

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	16	CDA	I	**Carrier Detect Channel A** The function of this pin depends on the selected clock mode. It can supply – either a modem control or a general purpose input (clock modes 0, 2, 3, 6, 7). If auto-start is programmed, it functions as a receiver enable signal. – or a receive strobe signal (clock mode 1). Polarity of CDA can be set to 'active low' with bit ICD in register CCR1H. Additionally, an interrupt may be issued if a state transition occurs at the CDA pin (programmable feature).
		FSCA	I	**Frame Sync Clock Channel A (cm 5a)** When the SCC is in the time-slot oriented clock mode 5a, this pin functions as the Frame Synchronization Clock input.
		$\overline{\text{RCGA}}$	I	**Receive Clock Gating Channel A (cm 4)** In clock mode 4 this pin is used as Receive Clock Gating signal. *If no clock gating function is required, a pull-up resistor to* V_{DD3} *is recommended.*
		OSRA	I	**Octet Sync Receive Channel A (cm 5b)** (clock mode 5b) When the SCC is in the time-slot oriented clock mode with octet-alignment (clock mode 5b), received octets are aligned to this synchronization pulse input.

Table 3 Serial Port Pins (cont'd)

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	23	$\overline{\text{RTSA}}$	O	**Request to Send Channel A** The function of this pin depends on the settings of bits RTS, FRTS in register CCR1H . In bus configuration, $\overline{\text{RTS}}$ can be programmed to: – go low during the actual transmission of a frame shifted by one clock period, excluding collision bits. – go low during reception of a data frame. – stay always high ($\overline{\text{RTS}}$ disabled).
	15	$\overline{\text{CTSA}}$	I	**Clear to Send Channel A** A low on the $\overline{\text{CTSA}}$ input enables the transmitter. Additionally, an interrupt may be issued if a state transition occurs at the $\overline{\text{CTSA}}$ pin (programmable feature). *If no 'Clear To Send' function is required, a pull-down resistor to V_{SS} is recommended.*
		CxDA	I	**Collision Data Channel A** In a bus configuration, the external serial bus must be connected to the corresponding CxDA pin for collision detection. A collision is detected whenever a logical '1' is driven on the open drain TxDA output but a logical '0' is detected via CxDA input.
		$\overline{\text{TCGA}}$	I	**Transmit Clock Gating Channel A (cm 4)** In clock mode 4 these pins are used as Transmit Clock Gating signals. *If no clock gating function is required, a pull-up resistor to V_{DD3} is recommended.*
		OSTA	I	**Octet Sync Transmit Channel A (cm 5b)** When the SCC is in the time-slot oriented clock mode with octet-alignment (clock mode 5b), a synchronization pulse on this input pin aligns transmit octets.

Table 3 Serial Port Pins (cont'd)

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	19	TxDA	O o/d	**Transmit Data Channel A** Transmit data is shifted out via this pin. It can be configured as push/pull or open drain output characteristic via bit 'ODS' in register CCR1L.
	17	RxDA	I	**Receive Data Channel A** Serial data is received on this pin.
	134	TxCLK B	I/O	**Transmit Clock Channel B** (corresponding to channel A)
	132	RxCLK B	I	**Receive Clock Channel B** (corresponding to channel A)
	135	CDB FSCB $\overline{RCGB}$ OSRB	I I I I	**Carrier Detect Channel B** **Frame Sync Clock Channel B (cm 5a)** **Receive Clock Gating Channel B (cm 4)** **Octet Sync Receive Channel B (cm 5b)** (corresponding to channel A)
	128	$\overline{RTSB}$	O	**Request to Send Channel B** (corresponding to channel A)
	8	$\overline{CTSB}$ CxDB $\overline{TCGB}$ OSTB	I I I I	**Clear to Send Channel B** **Collision Data Channel B** **Transmit Clock Gating Channel B (cm 4)** **Octet Sync Transmit Channel B (cm 5b)** (corresponding to channel A)
	133	TxDB	O o/d	**Transmit Data Channel B** (corresponding to channel A)

Table 3 Serial Port Pins (cont'd)

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	129	RxDB	I	**Receive Data Channel B** (corresponding to channel A)
	12 11	XTAL1 XTAL2	I O	**Crystal Connection** If the internal oscillator is used for clock generation (clock modes 0b, 6, 7) the external crystal has to be connected to these pins. The internal oscillator should be powered up (GMODE:OSCPD = '0') and the signal shaper may be activated (GMODE:DSHP = '0'). Moreover, XTAL1 may be used as input for a common clock source to both SCCs, provided by an external clock generator (oscillator). In this case the oscillator unit may be powered down and it is recommended to bypass the shaper of the internal oscillator unit by setting bit 'DSHP' to '1'. *A pull-down resistor to* V_{SS} *is recommended for pin XTAL1 if not used.*

Table 4 General Purpose Pins

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	127 123 126	GP2 GP1 GP0	I/O	**General Purpose Pins** These pins serve as general purpose input/output pins. *A pull-up resistor to* V_{DD3} *is recommended if pin is not used.*

Table 5 Test Interface Pins

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	139	$\overline{\text{TRST}}$	I	**JTAG Reset Pin (internal pull-up)** For proper device operation, a reset for the boundary scan controller must be supplied to this active low pin. *If the boundary scan of the SEROCCO-D is not used, this pin can be connected to* V_{SS} *to keep it in reset state.*
	5	TCK	I	**JTAG Test Clock (internal pull-up)** *If the boundary scan of the SEROCCO-D is not used, this pin may remain unconnected.*
	140	TDI	I	**JTAG Test Data Input (internal pull-up)** *If the boundary scan of the SEROCCO-D is not used, this pin may remain unconnected.*
	4	TDO	O	**JTAG Test Data Output**
	62	TMS	I	**JTAG Test Mode Select (internal pull-up)** *If the boundary scan of the SEROCCO-D is not used, this pin may remain unconnected.*
	99	TEST1	I	**Test Input 1** When connected to V_{DD3} the SEROCCO-D works in a vendor specific test mode. *This pin must be connected to* V_{SS}.
	100	TEST2	I	**Test Input 2** When connected to V_{DD3} the SEROCCO-D works in a vendor specific test mode. *This pin must be connected to* V_{SS}.

Table 6 Power Pins

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	6, 20, 26, 32, 43, 51, 56, 63, 69, 77, 81, 87, 93, 101, 112, 118, 124, 130, 137	V_{DD3}	-	**Digital Supply Voltage** 3.3 V ± 0.3 V All pins must be connected to the same voltage potential.
	7, 21, 27, 33, 44, 50, 57, 64, 70, 78, 82, 88, 94, 102, 113, 119, 125, 131, 138	V_{SS}	-	**Digital Ground** (0 V) All pins must be connected to the same voltage potential.
	13	V_{DDA}	-	**Analog Supply Voltage** 3.3 V ± 0.3 V This pin supplies the on-chip oscillator of the SEROCCO-D. If no separate analog power supply is available, this pin can be directly connected to V_{DD3}.

Table 6 Power Pins (cont'd)

Pin No.		Symbol	In (I) Out (O)	Function
	P-TQFP-144-10			
	10	V_{SSA}	-	**Analog Ground** (0 V) This pin supplies the ground level to the on-chip oscillator of the SEROCCO-D. If no separate analog power supply is available, this pin can be directly connected to V_{SS}.
	1, 2, 3, 9, 14, 34, 35, 36, 37, 38, 71, 72, 73, 74, 107, 108, 109, 110, 111, 141, 142, 143, 144	N.C.	-	**Not Connected**

3 Functional Overview

The functional blocks of SEROCCO-D can be divided into two major domains:

– the microprocessor interface of SEROCCO-D provides access to on-chip registers and to the "user" portion of the receive and transmit FIFOs (RFIFO/XFIFO). Optionally these FIFOs can be accessed by the built-in 4-channel DMA controller.
– the Serial Communication Controller (SCC) is capable of processing bit-synchronous (HDLC/SDLC/bitsync PPP) and octet-synchronous (octet-sync PPP) as well as fully transparent data traffic.

Data exchange between the serial communication controller and the microprocessor interface is performed using FIFOs, decoupling these two domains.

3.1 Block Diagram

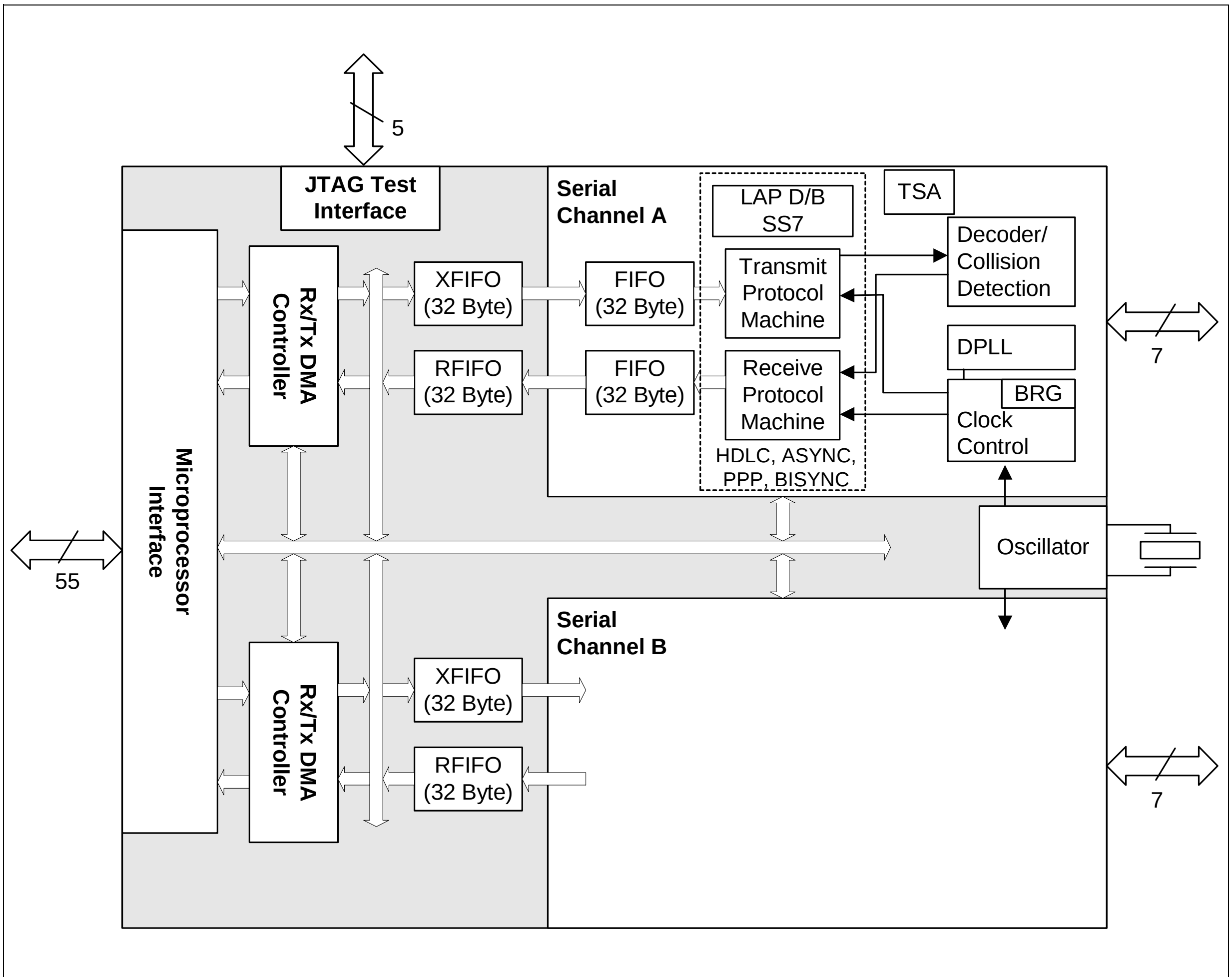


Figure 7 Block Diagram

3.2 Serial Communication Controller (SCC)

3.2.1 Protocol Modes Overview

The SCC is a multi-protocol communication controller. The core logic provides different protocol modes which are listed below:

- HDLC Modes
  - HDLC Transparent Operation (Address Mode 0)
  - HDLC Address Recognition (Address Mode 1, Address Mode 2 8/16-bit)
  - Full-Duplex LAPB/LAPD Operation (Automode 8/16-bit)
  - Half-Duplex SDLC-NRM Operation (Automode 8-bit)
  - Signaling System #7 (SS7) Operation

- Point-to-Point Protocol (PPP) Modes
  - Bit Synchronous PPP
  - Octet Synchronous PPP
  - Asynchronous PPP

- ASYNC Modes
  - Asynchronous Mode
  - Isochronous Mode

- BISYNC Modes
  - Bisynchronous Mode
  - Monosynchronous Mode

- Extended Transparent Mode

A detailed description of these protocol modes is given in **Chapter 4**, starting on **Page 86**.

3.2.2 SCC FIFOs

Each SCC provides its own transmit and receive FIFOs to handle internal arbitration and microcontroller latencies.

3.2.2.1 SCC Transmit FIFO

The SCC transmit FIFO is divided into two parts of 32 bytes each ('transmit pools'). The interface between the two parts provides synchronization between the microprocessor accesses and the protocol logic working with the serial transmit clock.

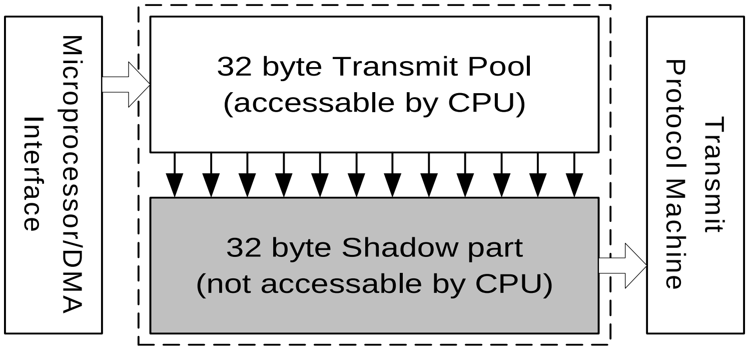


Figure 8 SCC Transmit FIFO

A 32 bytes FIFO part is accessable by the CPU/DMA controller; it accepts transmit data even if the SCC is in power-down condition (register CCR0H bit PU='0').

The only exception is a transmit data underrun (XDU) event. In case of an XDU event (e.g. after excessive bus latency), the FIFO will neither accept more data nor transfer another byte to the protocol logic. This XDU blocking mechanism prevents unexpected serial data. The blocking condition must be cleared by reading the interrupt status register ISR1 after the XDU interrupt was generated. Thus, the XDU interrupt indication should not be masked in register IMR1.

Transfer of data to the 32 byte shadow part only takes place if the SCC is in power-up condition and an appropriate transmit clock is provided depending on the selected clock mode.

Serial data transmission will start as soon as at least one byte is transferred into the shadow FIFO and transmission is enabled depending on the selected clock mode ($\overline{CTS}$ signal active, clock strobe signal active, timeslot valid or clock gapping signal inactive).

3.2.2.2 SCC Receive FIFO

The SCC receive FIFO is divided into two parts of 32 bytes each. The interface between the two parts provides synchronization between the microprocessor accesses and the protocol logic working with the serial receive clock.

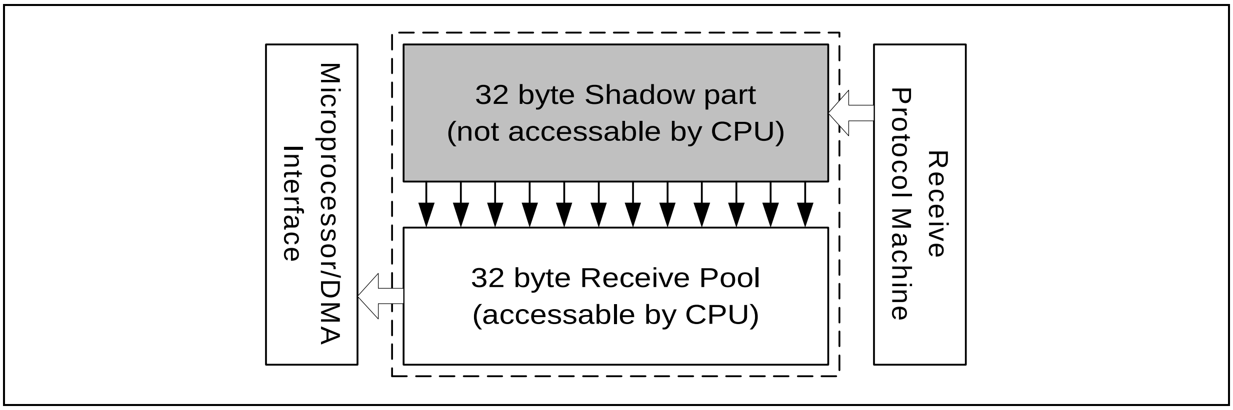


Figure 9 SCC Receive FIFO

New receive data is announced to the CPU with an interrupt latest when the FIFO fill level reaches a chosen threshold level (selected with bitfield 'RFTH(1..0)' in register **"CCR3H" on Page 171**). Default value for this threshold level is 32 bytes in HDLC/PPP modes and 1 byte in ASYNC or BISYNC mode.

If the SCC receive FIFO is completely filled, further incoming data is ignored and a receive data overflow condition ('RDO') is detected. As soon as the receive FIFO provides empty space, receive data is accepted again after a frame end or frame abort sequence. The automatically generated receive status byte (RSTA) will contain an 'RDO' indication in this case and the next incoming frame will be received in a normal way.

Therefore no further CPU intervention is necessary to recover the SCC from an 'RDO' condition.

A "frame" with 'RDO' status might be a mixture of a frame partly received before the 'RDO' event occured and the rest of this frame received after the receive FIFO again accepted data and the frame was still incoming. A quite arbitrary series of data or complete frames might get lost in case of an 'RDO' event. Every frame which is completely discarded because of an 'RDO' condition generates an 'RFO' interrupt.

The SCC receive FIFO can be cleared by command 'RRES' in register CMDRH. Note that clearing the receive FIFO during operation might delete a frame end / block end indication. A frame which was already partly transferred cannot be "closed" in this case. A new frame received after receiver reset command will be appended to this "open" frame.

3.2.2.3 SCC FIFO Access

Figure 10 and Figure 11 illustrate byte interpretation for Intel and Motorola 16-bit accesses to the transmit and receive FIFOs.

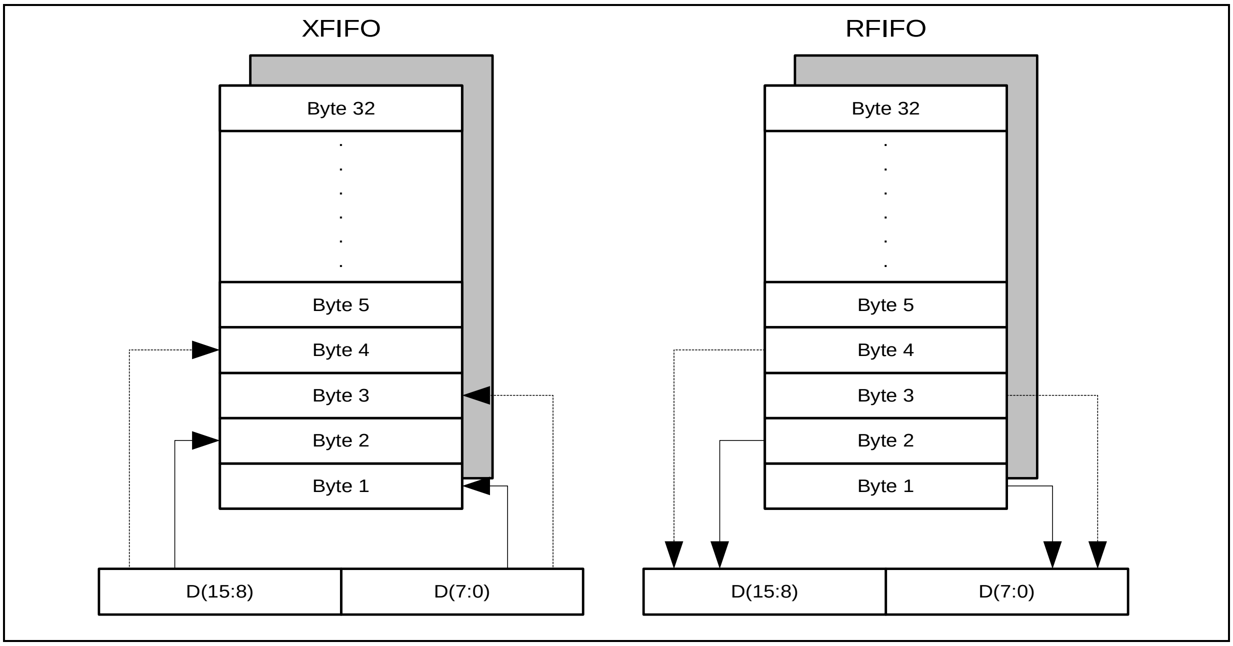


Figure 10 XFIFO/RFIFO Word Access (Intel Mode)

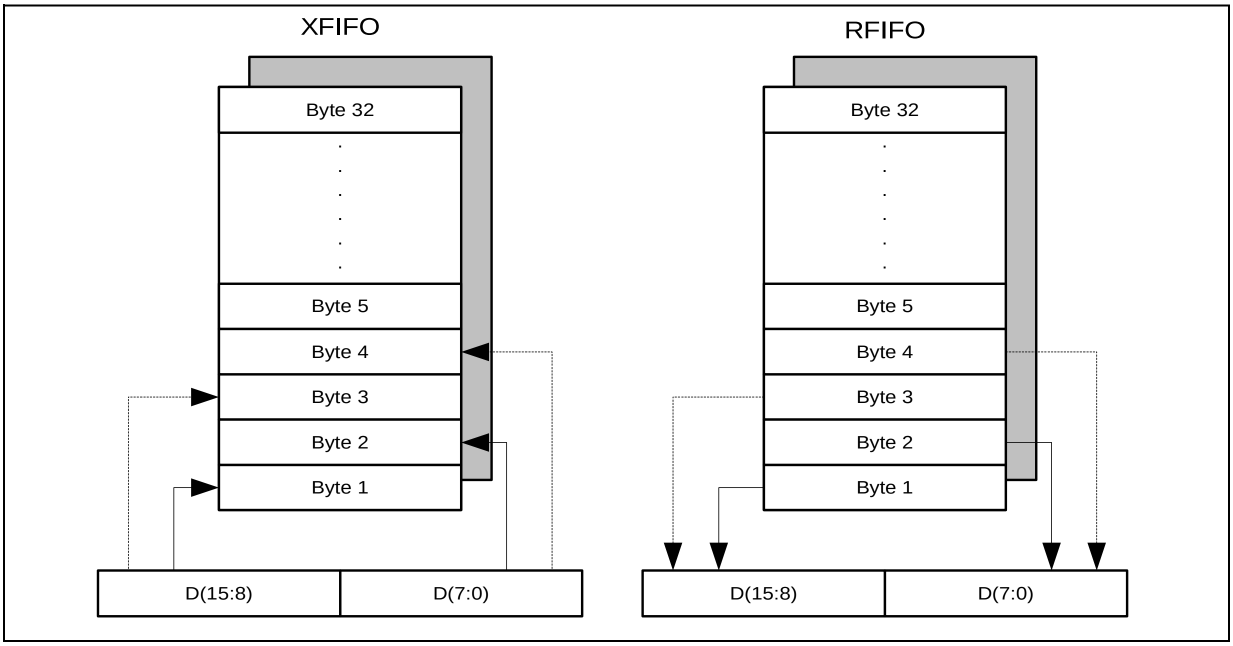


Figure 11 XFIFO/RFIFO Word Access (Motorola Mode)

3.2.3 Clocking System

The SEROCCO-D includes an internal Oscillator (OSC) as well as two independent Baud Rate Generators (BRG) and two Digital Phase Locked Loop (DPLL) circuits.

The transmit and receive clock can be generated either

- externally, and supplied directly via the RxCLK and/or TxCLK pins (called external clock modes)
- internally, by selecting
  - the internal oscillator (OSC) and/or the channel specific baud rate generator (BRG)
  - the internal DPLL, recovering the receive (and optionally transmit) clock from the receive data stream.

  (called internal clock modes)

There are a total of 14 different clocking modes programmable via bit field ’CM’ in register CCR0L, providing a wide variety of clock generation and clock pin functions, as shown in **Table 8.**

The transmit clock pins (TxCLK) may also be configured as output clock and control signals in certain clock modes if enabled via bit ’TOE’ in register CCR0L.

The clocking source for the DPLL’s is always the internal channel specific BRG; the scaling factor (divider) of the BRG can be programmed through BRRL and BRRH registers.

There are two channel specific internal operational clocks in the SCC:

One operational clock (= transmit clock) for the transmitter part and one operational clock (= receive clock) for the receiver part of the protocol logic.

Note: The internal timers always run using the internal transmit clock.

Table 7 Overview of Clock Modes

Clock			
Type	**Source**	**Generation**	**Clock Mode**
Receive Clock	RxCLK Pins	Externally	0, 1, 4, 5
	OSC, DPLL, BRG,	Internally	2, 3a, 6, 7a 3b, 7b
Transmit Clock	TxCLK Pins, RxCLK Pins	Externally	0a, 2a, 4, 6a 1,5
	OSC, DPLL, BRG/BCR, BRG	Internally	3a, 7a 2b, 6b 0b, 3b, 7b

The internal structure of each SCC channel consists of a transmit protocol machine clocked with the transmit frequency $\mathbf{f_{TRM}}$ and a receive protocol machine clocked with the receive frequency $\mathbf{f_{REC}}$.

The clocks $\mathbf{f_{TRM}}$ and $\mathbf{f_{REC}}$ are internal clocks only and need not be identical to external clock inputs e.g. $\mathbf{f_{TRM}}$ and TxCLK input pin.

The features of the different clock modes are summarized in **Table 8**.

Table 8 Clock Modes of the SCCs

Channel Configuration		Clock Sources				Control Sources					
Clock Mode CCR0L: CM(2..0)	CCR0L: SSEL	to BRG	to DPLL	to REC	to TRM	CD	R- Strobe	X- Strobe	Frame-Sync Tx	Rx	Output via TxCLK (if CCR0L: TOE = '1')
0a	0	–	–	RxCLK	TxCLK	CD	–	–	–	–	–
0b	1	OSC	–	RxCLK	BRG	CD	–	–	–	–	BRG
1	X	–	–	RxCLK	RxCLK	–	CD	TxCLK	–	–	–
2a	0	RxCLK	BRG	DPLL	TxCLK	CD	–	–	–	–	–
2b	1	RxCLK	BRG	DPLL	BRG/16	CD	–	–	–	–	BRG/16
3a	0	RxCLK	BRG	DPLL	DPLL	CD	–	–	–	–	DPLL
3b	1	RxCLK	–	BRG	BRG	CD	–	–	–	–	BRG
4	X	–	–	RxCLK	TxCLK	–	$\overline{\text{RCG}}$	$\overline{\text{TCG}}$	–	–	-
5a	0	–	–	RxCLK	RxCLK	–	(TSAR/ PCMRX)	(TSAX/ PCMTX)	FSC	FSC	TS-Control
5b	1	–	–	RxCLK	TxCLK	–	(TSAR/ PCMRX)	(TSAX/ PCMTX)	OST	OSR	–
6a	0	OSC	BRG	DPLL	TxCLK	CD	–	–	–	–	–
6b	1	OSC	BRG	DPLL	BRG/16	CD	–	–	–	–	BRG/16
7a	0	OSC	BRG	DPLL	DPLL	CD	–	–	–	–	DPLL
7b	1	OSC	–	BRG	BRG	CD	–	–	–	–	BRG

Note: If asynchronous operation is selected (asynchronous PPP, ASYNC mode), some clock mode frequencies can or must be divided by 16 as selected by the Bit Clock Rate bit CCR0L:BCR:

Clock Mode	$\mathbf{f_{REC}}$	$\mathbf{f_{TRM}}$
0a	f_{RxCLK}/BCR	f_{TxCLK}
0b	f_{RxCLK}/BCR	f_{BRG}
1	f_{RxCLK}/BCR	f_{RxCLK}/BCR
3b, 7b	f_{BRG}/BCR	f_{BRG}/BCR

When bit clock rate is '16' (bit BCR = '1'), oversampling (3 samples) in conjunction with majority decision is performed. BCR has no effect when using clock mode 2, 3a, 4, 5, 6, or 7a.

Note: If one of the clock modes 0b, 6 or 7 is selected, the internal oscillator (OSC) should be enabled by clearing bit GMODE:OSCPD. This allows connection of an external crystal to pins XTAL1-XTAL2. The output signal of the OSC can be used for one serial channel, or for both serial channels (independent baud rate generators and DPLLs). Moreover, XTAL1 alone can be used as input for an externally generated clock.

The first two columns of **Table 8** list all possible clock modes configured via bit field ’CM’ and bit ’SSEL’ in register CCR0L.

For example, clock mode 6b is choosen by writing a ’6’ to register CCR0L.CM(2:0) and by setting bit CCR0L.SSEL equal to ’1’. The following 4 columns (grouped as ’Clock Sources’) specify the source of the internal clocks. Columns REC and TRM correspond to the domain clock frequencies $\mathbf{f}_{\mathbf{REC}}$ and $\mathbf{f}_{\mathbf{TRM}}$.

The columns grouped as ’Control Sources’ cover additional clock mode dependent control signals like strobe signals (clock mode 1), clock gating signals (clock mode 4) or synchronization signals (clock mode 5). The last column describes the function of signal TxCLK which in some clock modes can be enabled as output signal monitoring the effective transmit clock or providing a time slot control signal (clock mode 5).

The following is an example of how to read **Table 8**:

For clock mode 6b (row ’6b’) the TRM clock (column ’TRM’) is supplied by the baudrate generator (BRG) output divided by 16 (source BRG/16). The BRG (column ’BRG’) is derived from the internal oscillator which is supplied by pin XTAL1 and XTAL2.

The REC clock (column ’REC’) is supplied by the internal DPLL which itself is supplied by the baud rate generator (column ’DPLL’) again.

Note: The REC clock is DPLL clock divided by 16.

If enabled by bit ’TOE’ in register CCR0L the resulting transmit clock can be monitored via pin TxCLK (last column, row ’6b’).

The clocking concept is illustrated in a block diagram manner in the following figure:

Additional control signals are not illustrated (please refer to the detailed clock mode descriptions below).

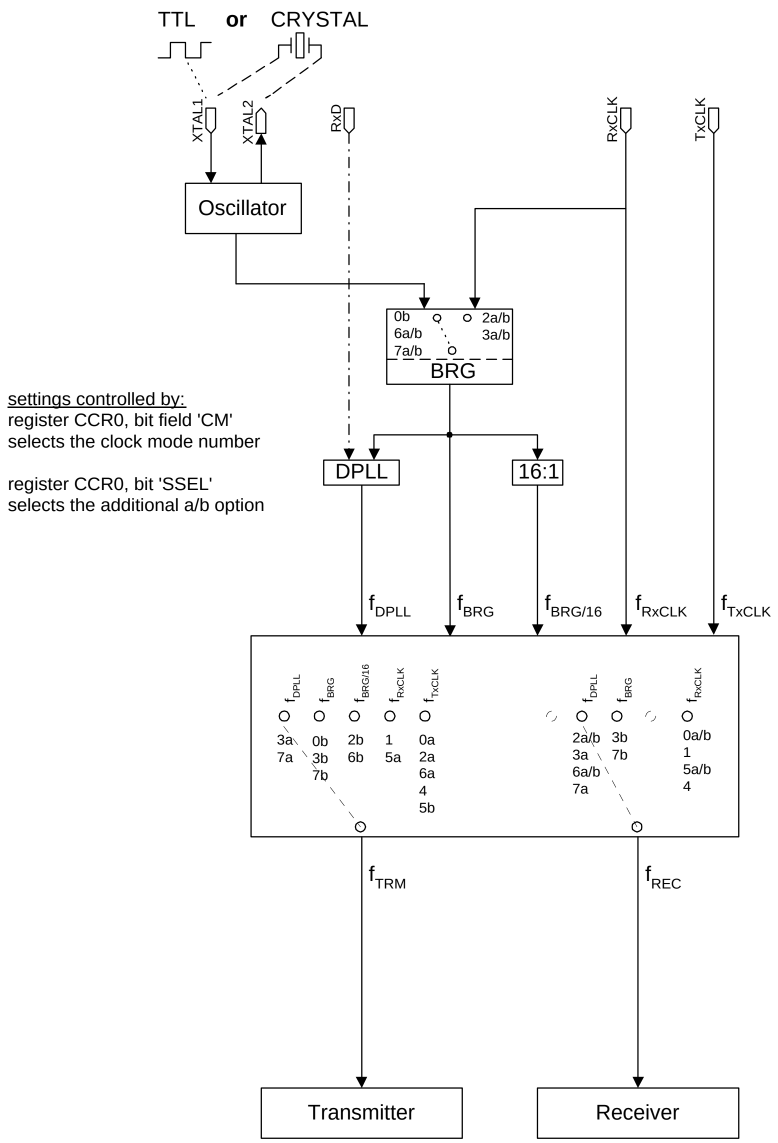


Figure 12 Clock Supply Overview

Clock Modes

3.2.3.1 Clock Mode 0 (0a/0b)

Separate, externally generated receive and transmit clocks are supplied to the SCC via their respective pins. The transmit clock may be directly supplied by pin TxCLK (clock mode 0a) or generated by the internal baud rate generator from the clock supplied at pin XTAL1 (clock mode 0b).

In clock mode 0b the resulting transmit clock can be driven out to pin TxCLK if enabled via bit 'TOE' in register CCR0L.

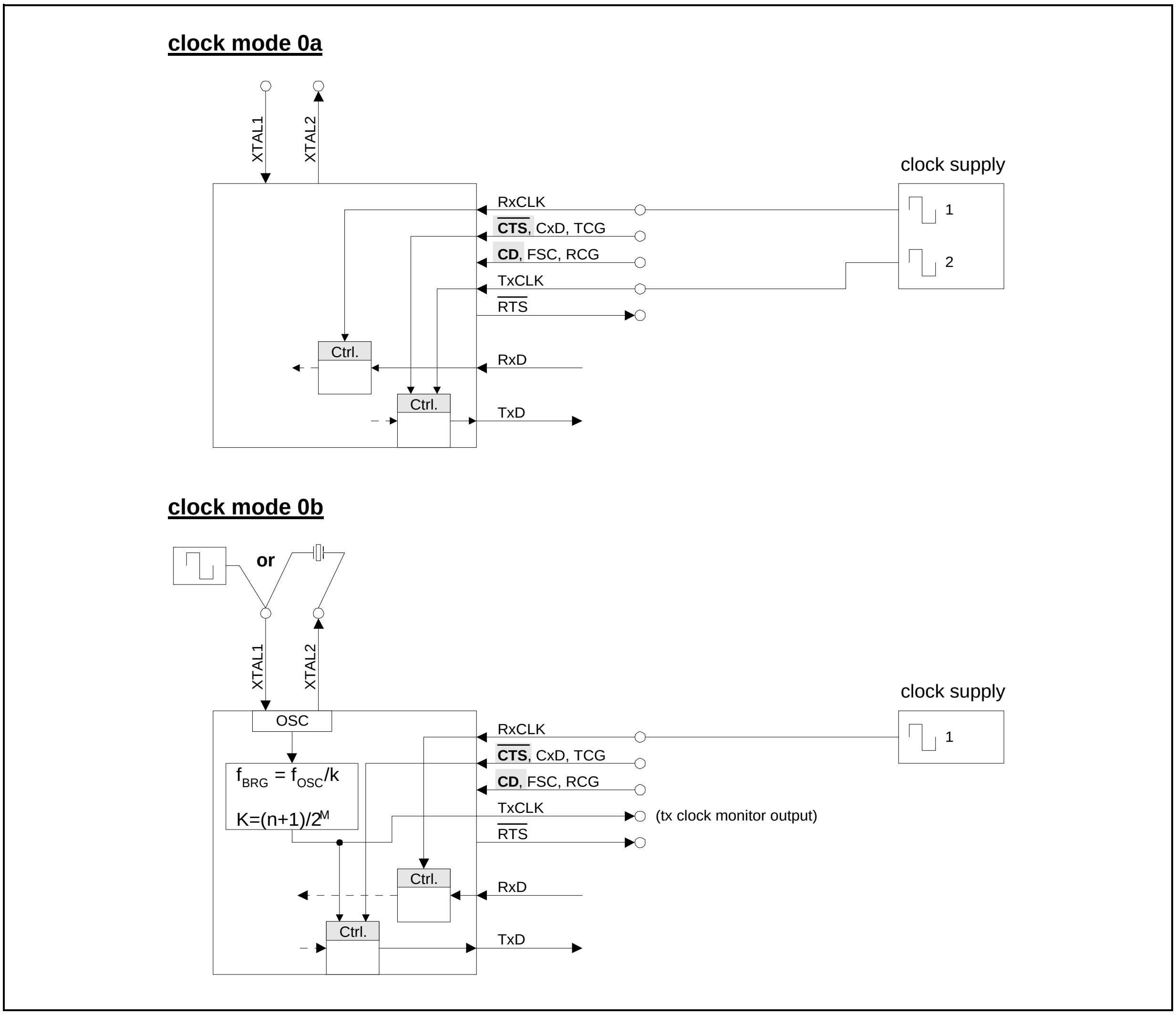


Figure 13 Clock Mode 0a/0b Configuration

3.2.3.2 Clock Mode 1

Externally generated RxCLK is supplied to both the receiver and transmitter. In addition, a receive strobe can be connected via CD and a transmit strobe via TxCLK pin. These strobe signals work on a per bit basis. This operating mode can be used in time division multiplex applications or for adjusting disparate transmit and receive data rates.

Note: In Extended Transparent Mode, the above mentioned strobe signals provide byte synchronization (byte alignment).
This means that the strobe signal needs to be detected once only to transmit or receive a complete byte.

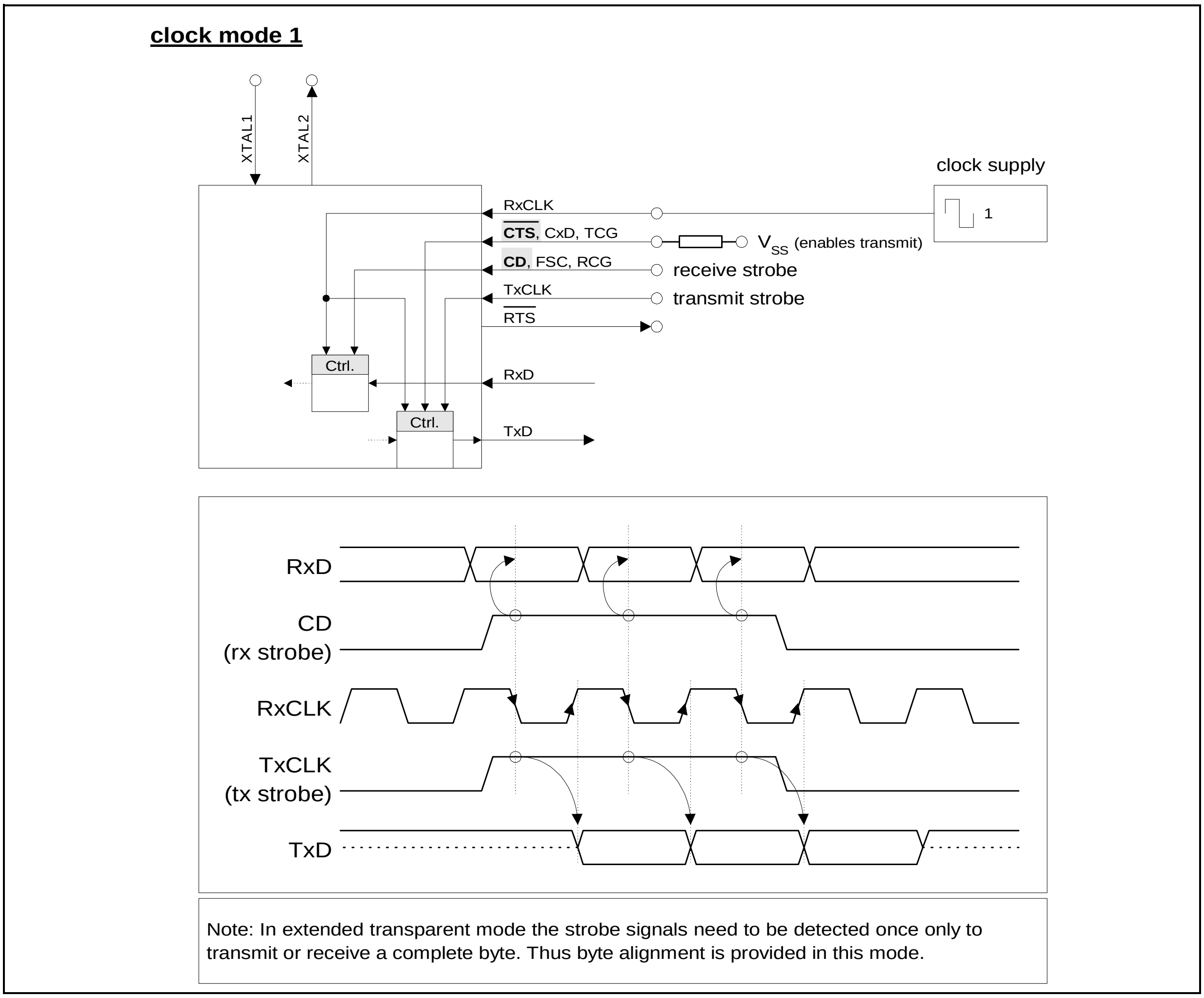


Figure 14 Clock Mode 1 Configuration

3.2.3.3 Clock Mode 2 (2a/2b)

The BRG is driven by an external clock (RxCLK pin) and delivers a reference clock for the DPLL which is 16 times of the resulting DPLL output frequency which in turn supplies the internal receive clock. Depending on the programming of register CCR0L bit 'SSEL', the transmit clock will be either an external input clock signal provided at pin TxCLK in clock mode 2a or the clock delivered by the BRG divided by 16 in clock mode 2b. In the latter case, the transmit clock can be driven out to pin TxCLK if enabled via bit 'TOE' in register CCR0L.

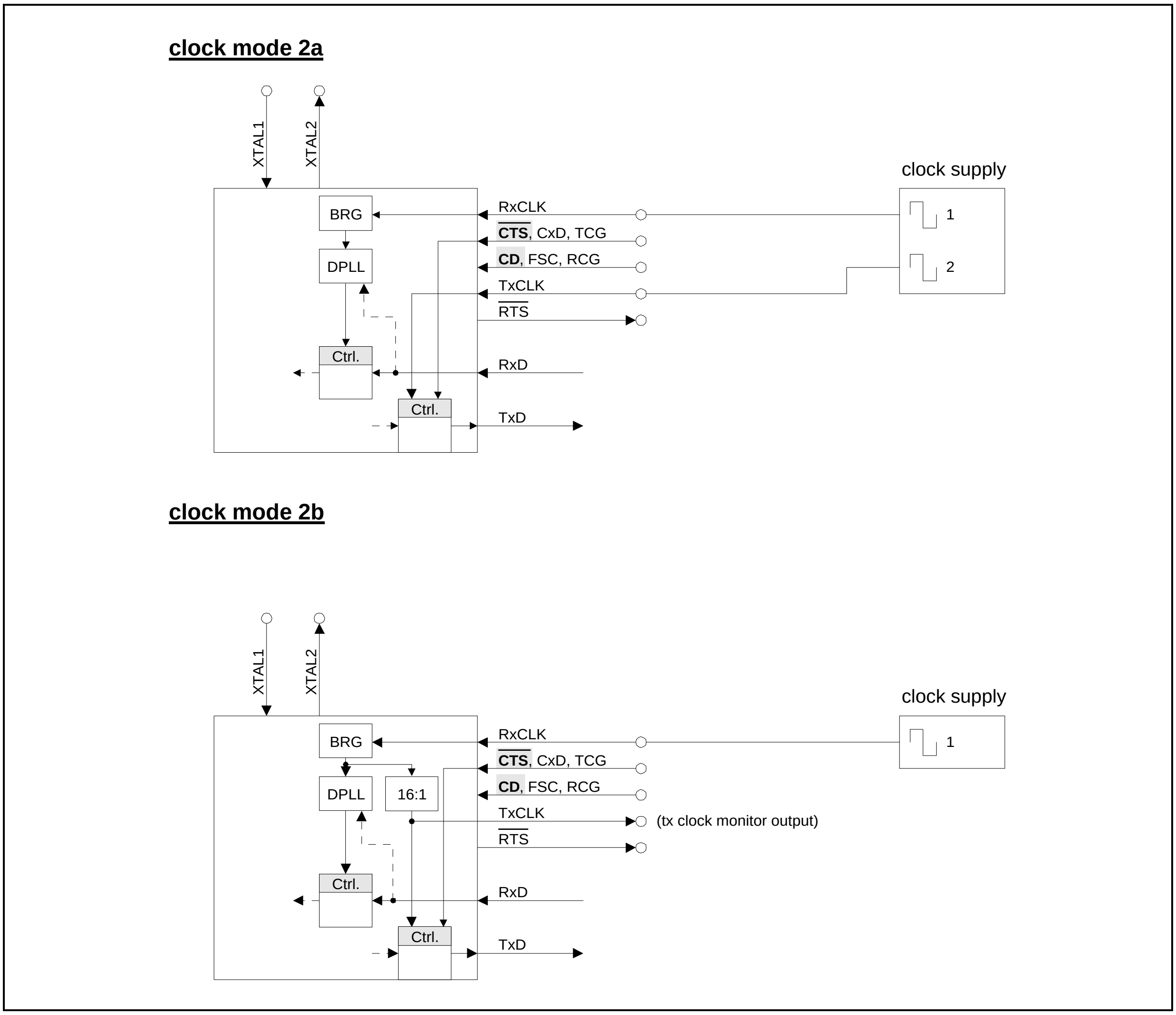


Figure 15 Clock Mode 2a/2b Configuration

3.2.3.4 Clock Mode 3 (3a/3b)

The BRG is fed with an externally generated clock via pin RxCLK. Depending on the value of bit 'SSEL' in register CCR0L the BRG delivers either a reference clock for the DPLL which is 16 times of the resulting DPLL output frequency (clock mode 3a) or delivers directly the receive and transmit clock (clock mode 3b). In the first case the DPLL output clock is used as receive and transmit clock.

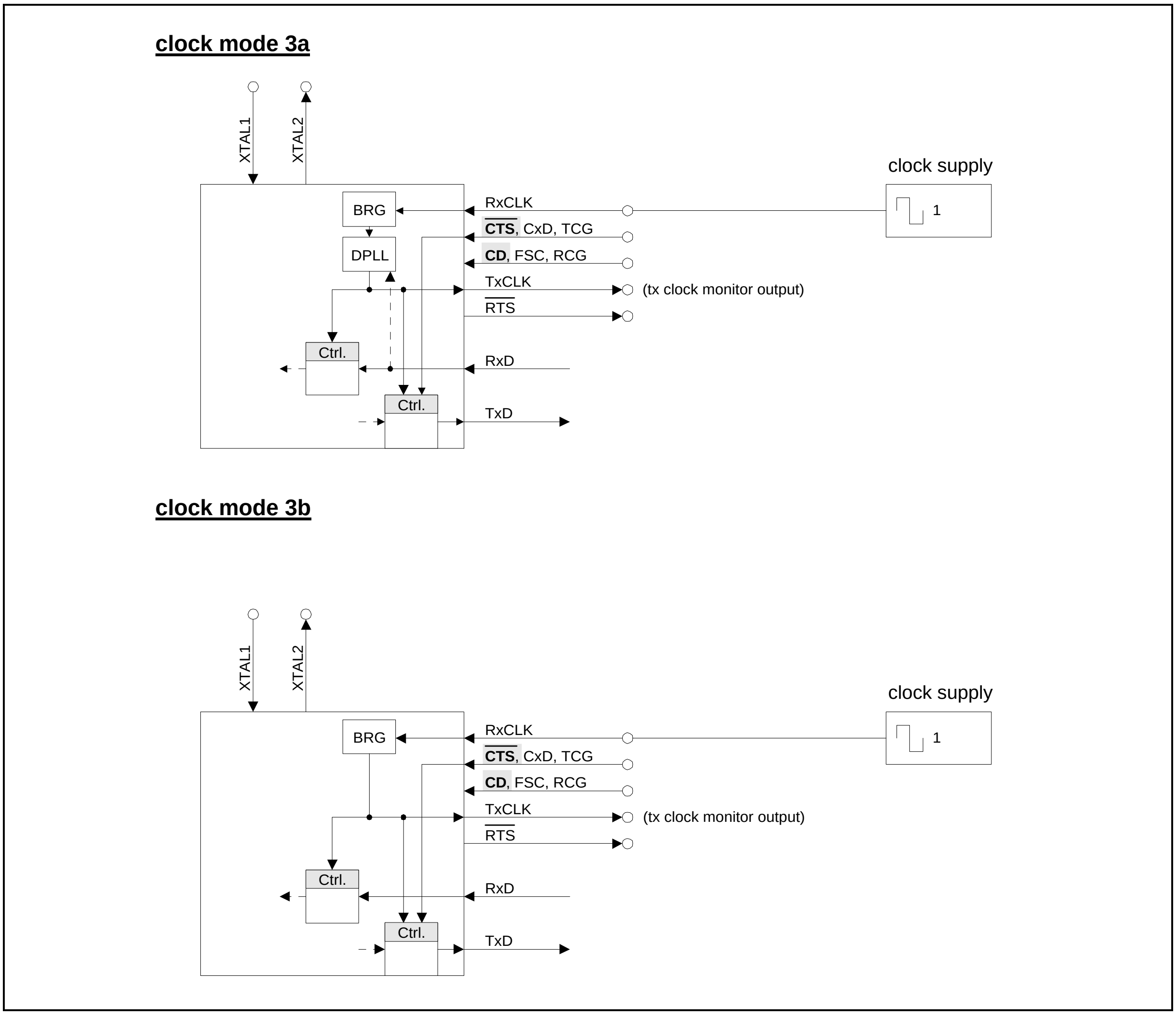


Figure 16 Clock Mode 3a/3b Configuration

3.2.3.5 Clock Mode 4

Separate, externally generated receive and transmit clocks are supplied via pins RxCLK and TxCLK. In addition separate receive and transmit clock gating signals are supplied via pins $\overline{\text{RCG}}$ and $\overline{\text{TCG}}$. These gating signals work on a per bit basis.

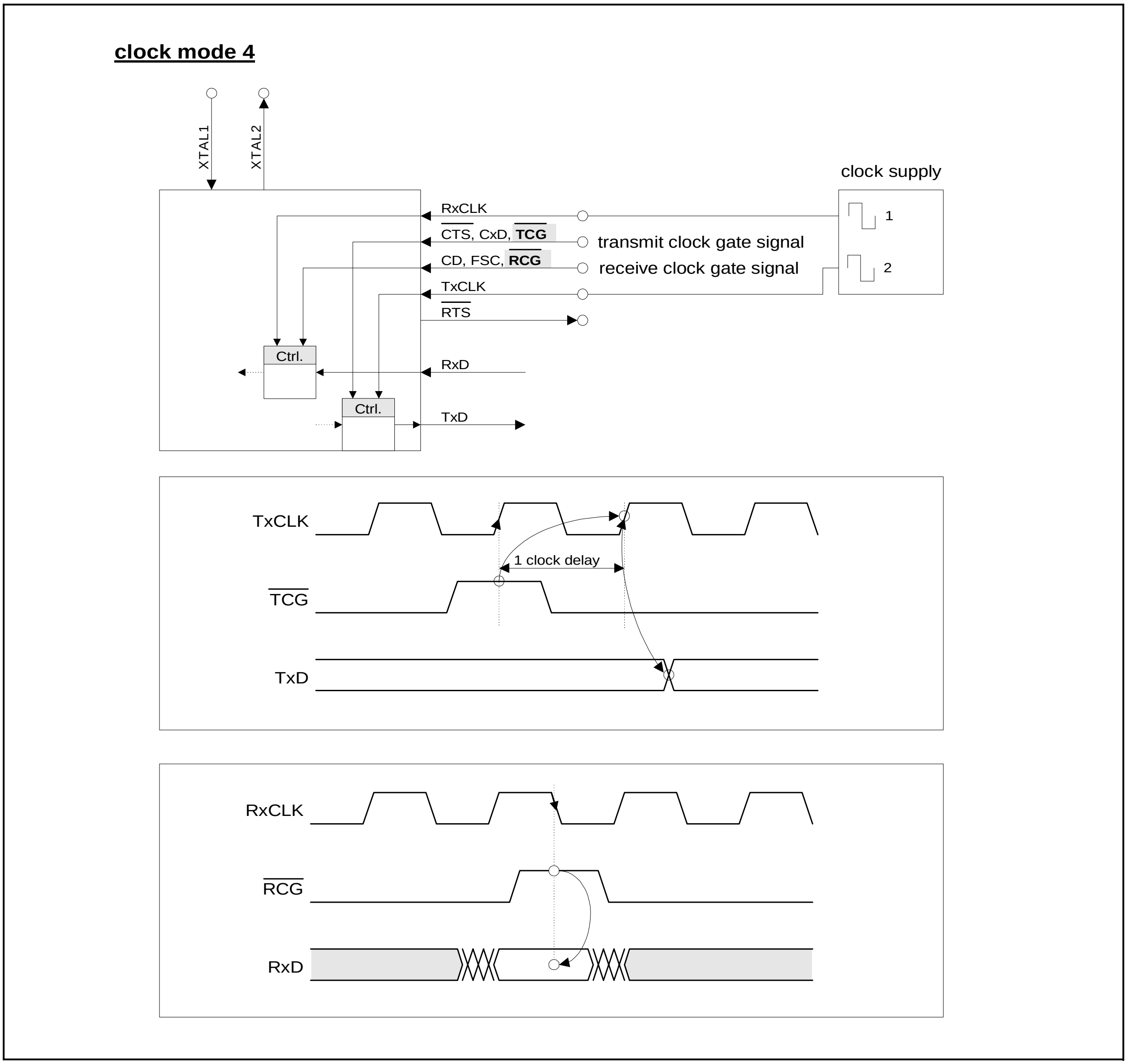


Figure 17 Clock Mode 4 Configuration

3.2.3.6 Clock Mode 5a (Time Slot Mode)

This operation mode has been designed for application in time-slot oriented PCM systems.

Note: For correct operation NRZ data coding/encoding should be used.

The receive and transmit clock are common for each channel and must be supplied externally via pin RxCLK. The SCC receives and transmits only during fixed time-slots. Either one time-slot

– of programmable width (1 … 512 bit, via TTSA and RTSA registers), and
– of programmable location with respect to the frame synchronization signal (via pin FSC)

or up to 32 time-slots

– of constant width (8 bits), and
– of programmable location with respect to the frame synchronization signal (via pin FSC)

can be selected.

The time-slot locations can be programmed independently for receive and transmit direction via TTSA/RTSA and PCMTX/PCMRX registers.

Depending on the value programmed via those registers, the receive/transmit time-slot starts with a delay of 1 (minimum delay) up to 1024 clock periods following the frame synchronization signal.

Figure 18 shows how to select a time-slot of programmable width and location and **Figure 19** shows how to select one or more time-slots of 8-bit width.

If bit 'TOE' in register CCR0L is set, the selected transmit time-slot(s) is(are) indicated at an output status signal via pin TxCLK, which is driven to 'low' during the active transmit window.

Bit 'TSCM' in register CCR1H determines whether the internal offset counters are continuously running even if no synchronization pulse is detected at FSC signal or stopping at their maximum value.

In the continuous case the repetition rate of offset counter operation is 1024 transmit or receive clocks respectively. An FSC pulse detected earlier resets the counters and starts operation again.

In the non-continuous case the time slot assigner offset counter is stopped after the counter reached its maximum value and is started again if an FSC pulse is detected.

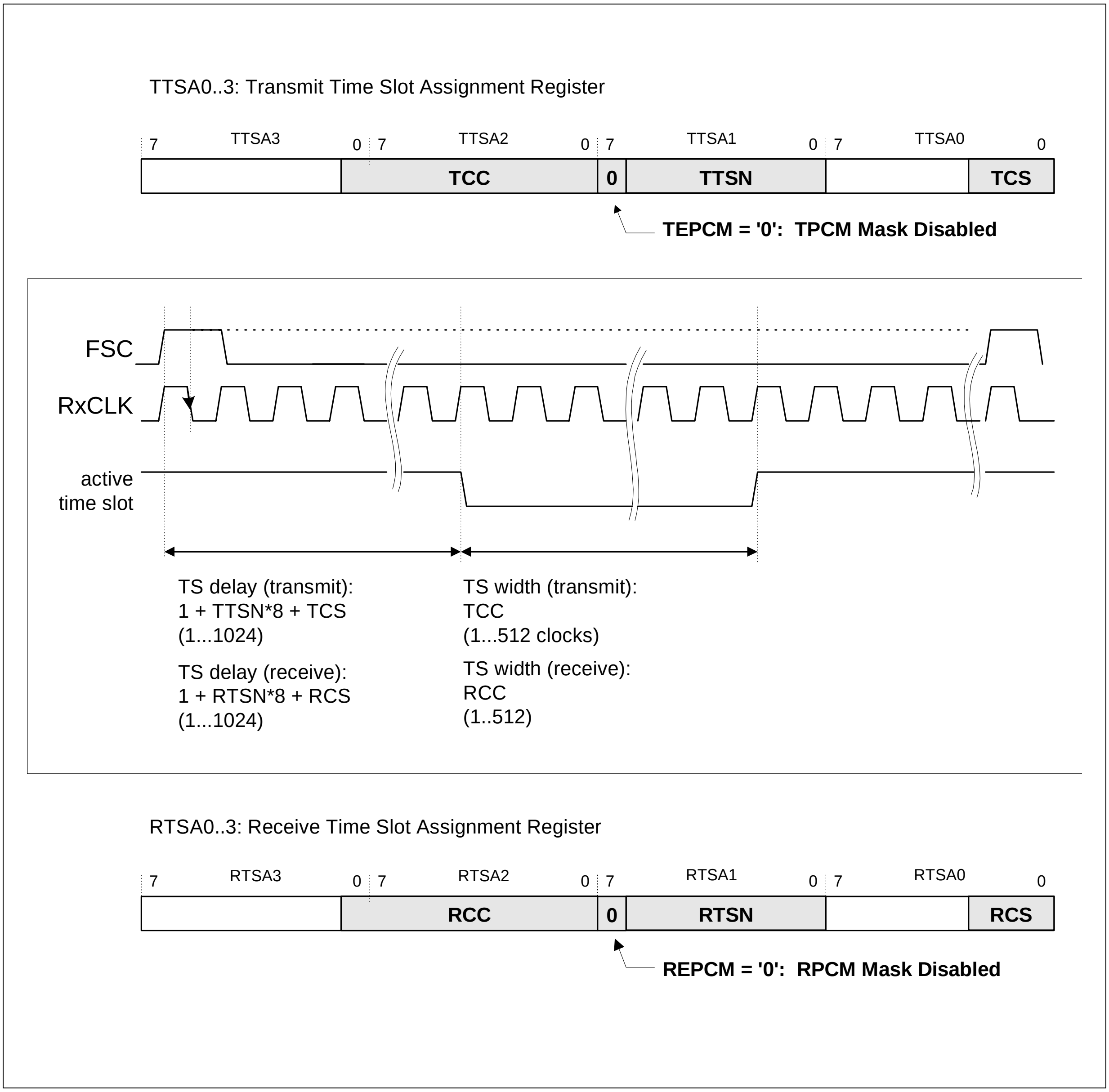


Figure 18 Selecting one time-slot of programmable delay and width

*Note: If time-slot 0 is to be selected, the DELAY has to be as long as the PCM frame itself to achieve synchronization (at least for the 2nd and subsequent PCM frames): DELAY = PCM frame length = 1 + xTSN*8 + xCS. xTSN and xCS have to be set appropriately.*

Example: Time-slot 0 in E1 (2.048 Mbit/s) system has to be selected.
*PCM frame length is 256 clocks. 256 = 1+ xTSN*8 + xCS. => xTSN = 31, xCS = 7.*

Note: In extended transparent mode the width xCC of the selected time-slot has to be $n \times 8$ bit because of character synchronization (byte alignment). In all other modes the width can be used to define windows down to a minimum length of one bit.

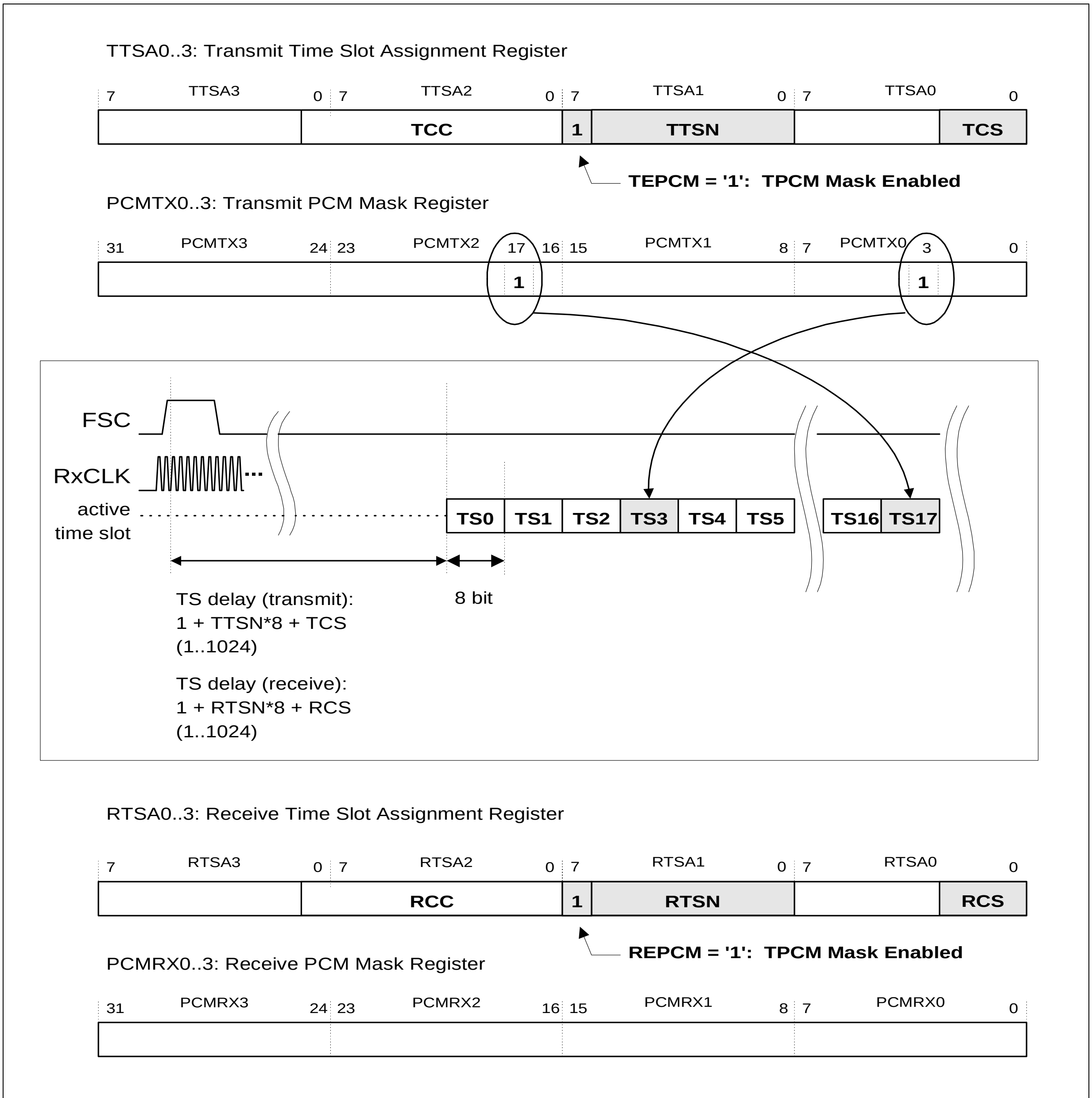


Figure 19 Selecting one or more time-slots of 8-bit width

The common transmit and receive clock is supplied at pin RxCLK and the common frame synchronisation signal at pin FSC. The "strobe signals" for active time slots are generated internally by the time slot assigner block (TSA) independent in transmit and receive direction.

When the transmit and receive PCM masks are enabled, bit fields ’TCC’ and ’RCC’ are ignored because of the constant 8-bit time slot width.

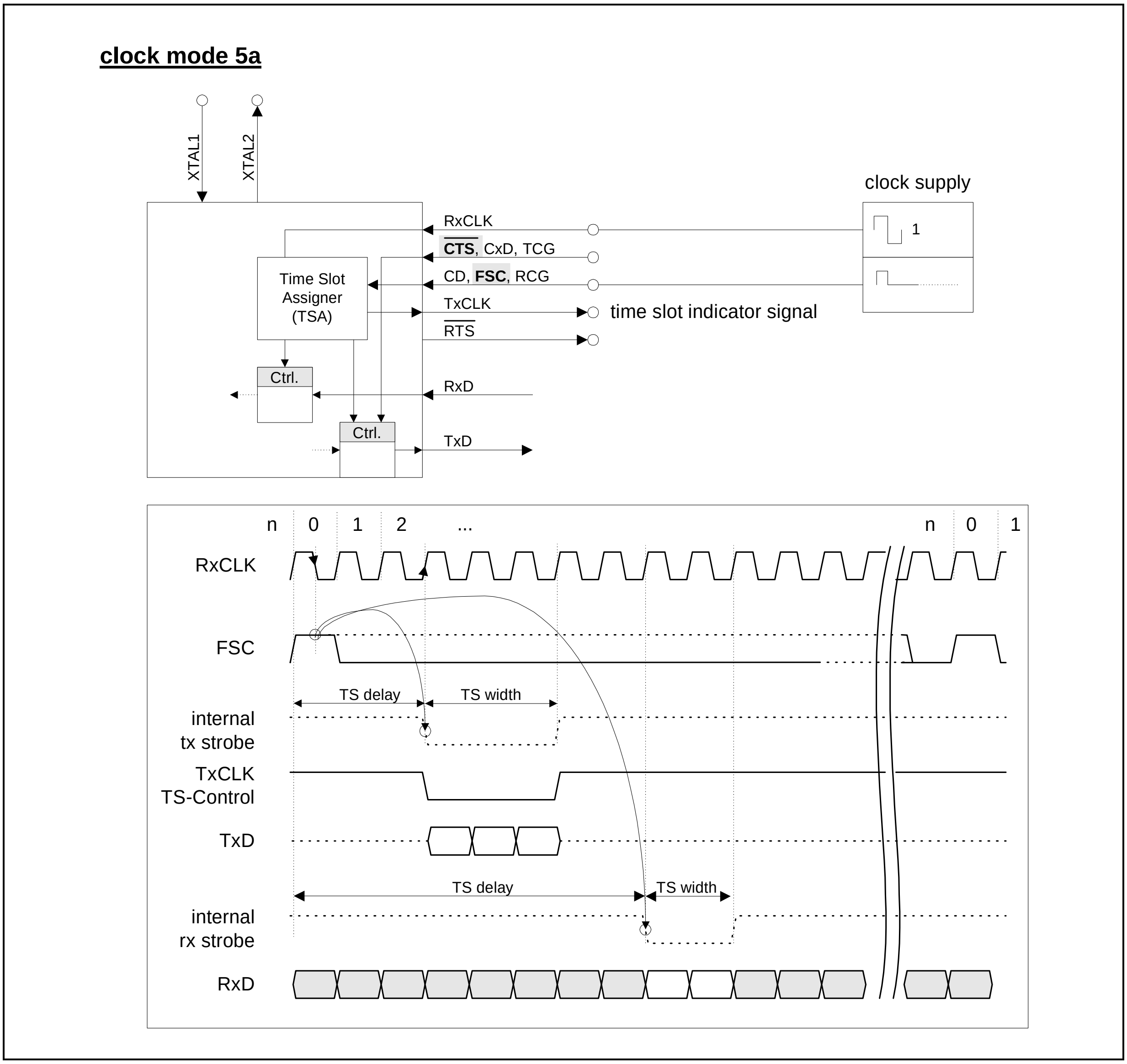


Figure 20 Clock Mode 5a Configuration

Note: The transmit time slot delay and width is programmable via bit fields 'TTSN', 'TCS' and 'TCC' in registers TTSA0..TTSA3.
The receive time slot delay and width is programmable via bit fields 'RTSN', 'RCS' and 'RCC' in registers RTSA0..RTSA3.

The following figures provide a more detailed description of the TSA internal counter operation and exceptional cases:

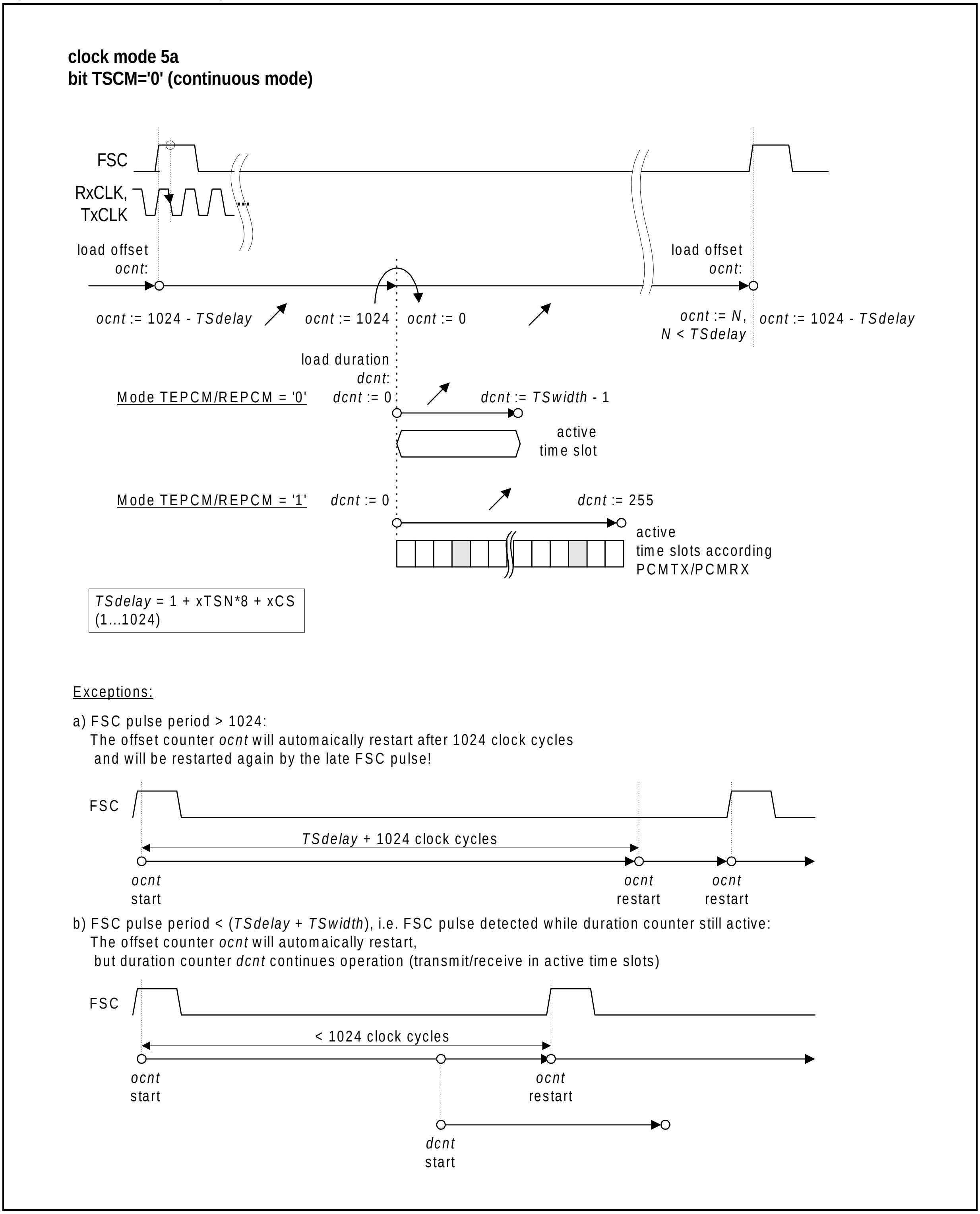


Figure 21 Clock Mode 5a "Continuous Mode"

Each frame sync pulse starts the internal offset counter with (1024 - TSdelay) whereas *TSdelay* is the configured value defining the start position. Whenever the offset counter reaches its maximum value 1024, it triggers the duration counter to start operation.

If continuous mode is selected (bit CCR1H.TSCM='0') the offset counter continues starting with value 0 until another frame sync puls is detected or again the maximum value 1024 is reached.

Once the duration counter is triggered it runs out independently from the offset counter, i.e. an active time slot period may overlap with the next frame beginning (frame sync event, refer to exception b) in **Figure 21**).

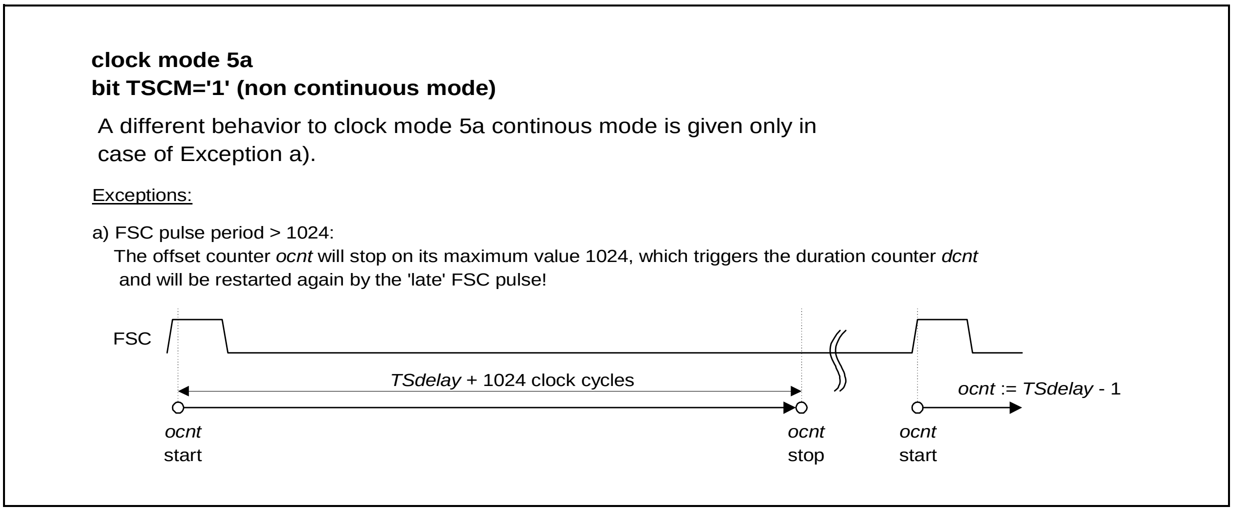


Figure 22 Clock Mode 5a "Non Continuous Mode"

If non-continuous mode is selected (bit CCR1H.TSCM='1') the offset counter is stopped on its maximum value 1024 until another frame sync puls is detected. This allows frame sync periods greater than 1024 clock cycles, but the accesible part is limited by the range of TSdelay value (1..1024) plus TSwidth (1..512) or plus 256 clock cycles if the PCM mask is selected.

3.2.3.7 Clock Mode 5b (Octet Sync Mode)

This operation mode has been designed for applications using Octet Synchronous PPP.

It is based on clock mode 5a, but only 8-bit (octet) wide time slot operation is supported, i.e. bits TTSA1.TEPCM and RTSA1.REPCM must be set to ’1’. Clock mode 5b provides octet alignment to time slots if Octet Synchronous PPP protocol mode or extended transparent mode is selected.

Note: For correct operation NRZ data coding/encoding should be used.

The receive and transmit clocks are separate and must be supplied at pins RxCLK and TxCLK. The SCC receives and transmits only during fixed octet wide time-slots of programmable location with respect to the octet synchronization signals (via pins OSR and OST)

The time-slot locations can be programmed independently for receive and transmit direction via registers TTSA0..TTSA3 / RTSA0..RTSA3 and PCMTX0..PCMTX3 / PCMRX0..PCMRX3.

Figure 23 shows how to select one or more octet wide time-slots.

Bit ’TSCM’ in register CCR1H determines whether the internal counters are continuously running even if no synchronization pulse is detected at OST/OSR signals or stopping at their maximum value.

In the continuous case the repetition rate of operation is 1024 transmit or receive clocks respectively. An OST/OSR pulse detected earlier resets the corresponding offset counter and starts operation again.

In the non-continuous case the transmit/receive time slot assigner offset counter is stopped after the counter reached its maximum value and is started again if an OST/OSR pulse is detected.

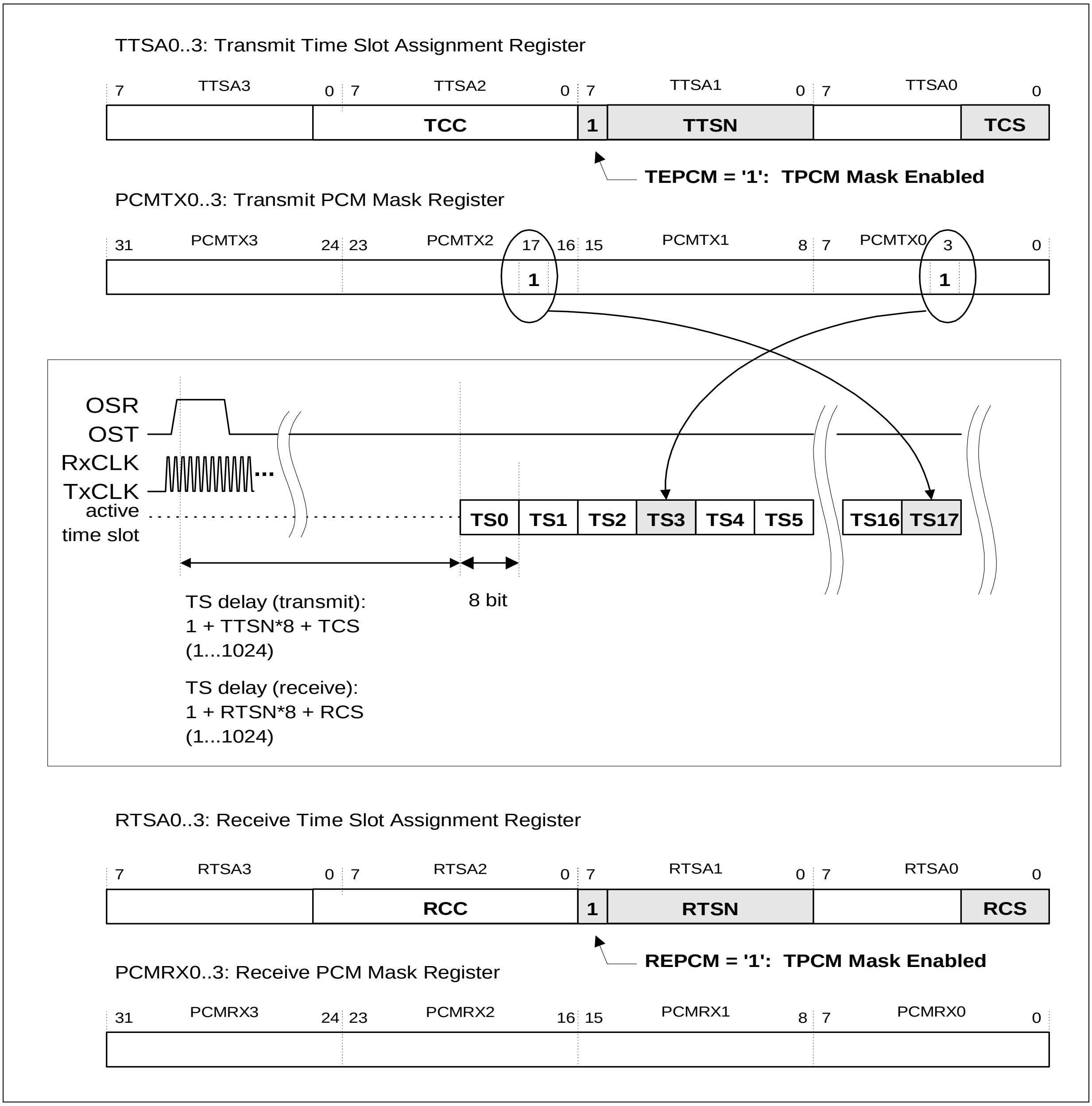


Figure 23 Selecting one or more octet wide time-slots

The transmit and receive clocks are supplied at pins RxCLK and TxCLK. The Octet synchronisation signals are supplied at pins OSR and OST. The "strobe signals" for active time slots are generated internally by the time slot assigner blocks (TSA) independent in transmit and receive direction.

Bit fields ’TCC’ and ’RCC’ are ignored because of the constant 8-bit time slot width.

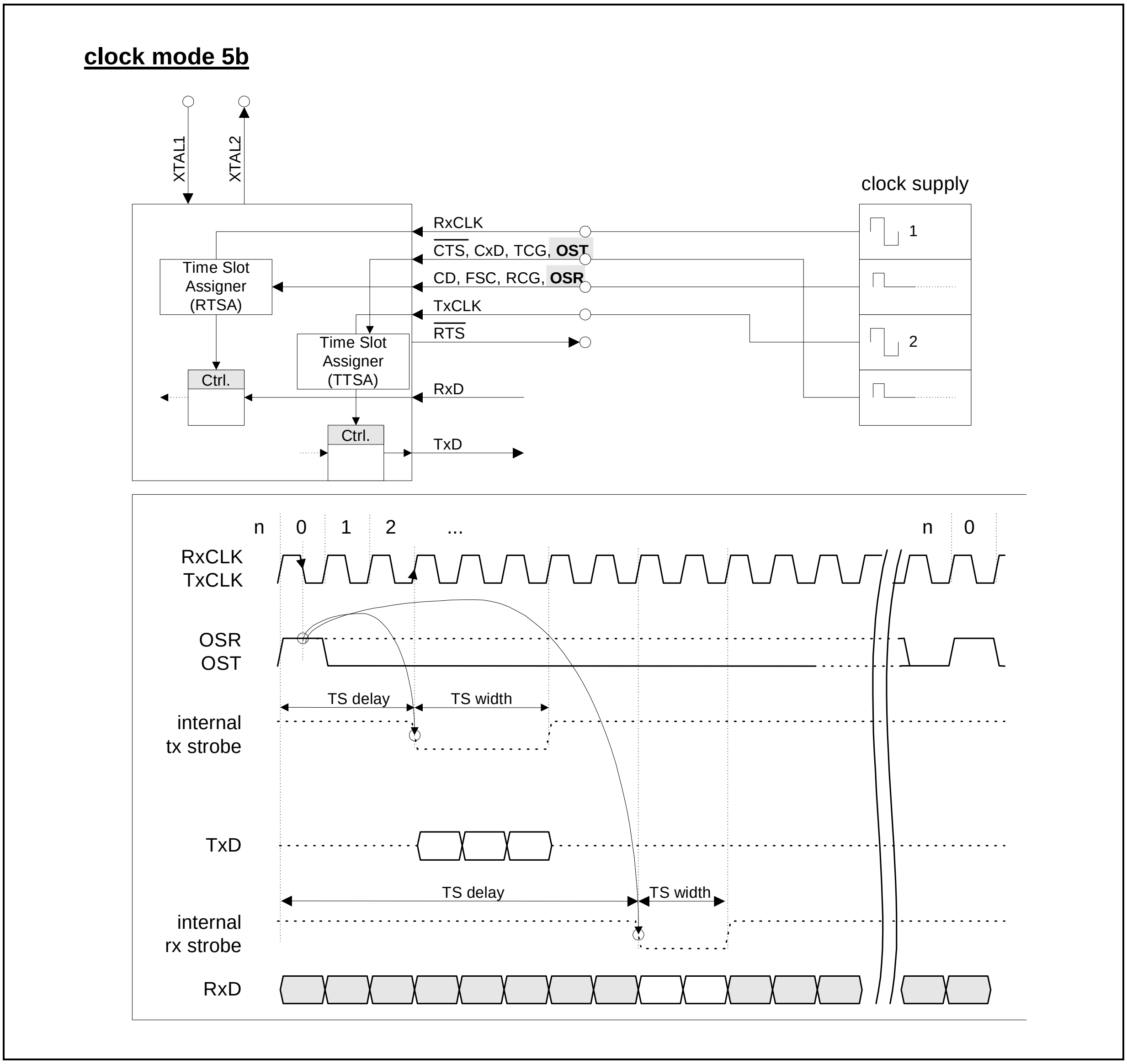


Figure 24 Clock Mode 5b Configuration

Note: The transmit time slot delay and width is programmable via bit fields 'TTSN', 'TCS' and 'TCC' in registers TTSA0..TTSA3.
The receive time slot delay and width is programmable via bit fields 'RTSN', 'RCS' and 'RCC' in registers RTSA0..RTSA3.

3.2.3.8 Clock Mode 6 (6a/6b)

This clock mode is identical to clock mode 2a/2b except that the clock source of the BRG is supplied at pin XTAL1.

The BRG is driven by the internal oscillator and delivers a reference clock for the DPLL which is 16 times the resulting DPLL output frequency which in turn supplies the internal receive clock. Depending on the programming of register CCR0L bit 'SSEL', the transmit clock will be either an external input clock signal provided at pin TxCLK in clock mode 6a or the clock delivered by the BRG divided by 16 in clock mode 6b. In the latter case, the transmit clock can be driven out to pin TxCLK if enabled via bit 'TOE' in register CCR0L.

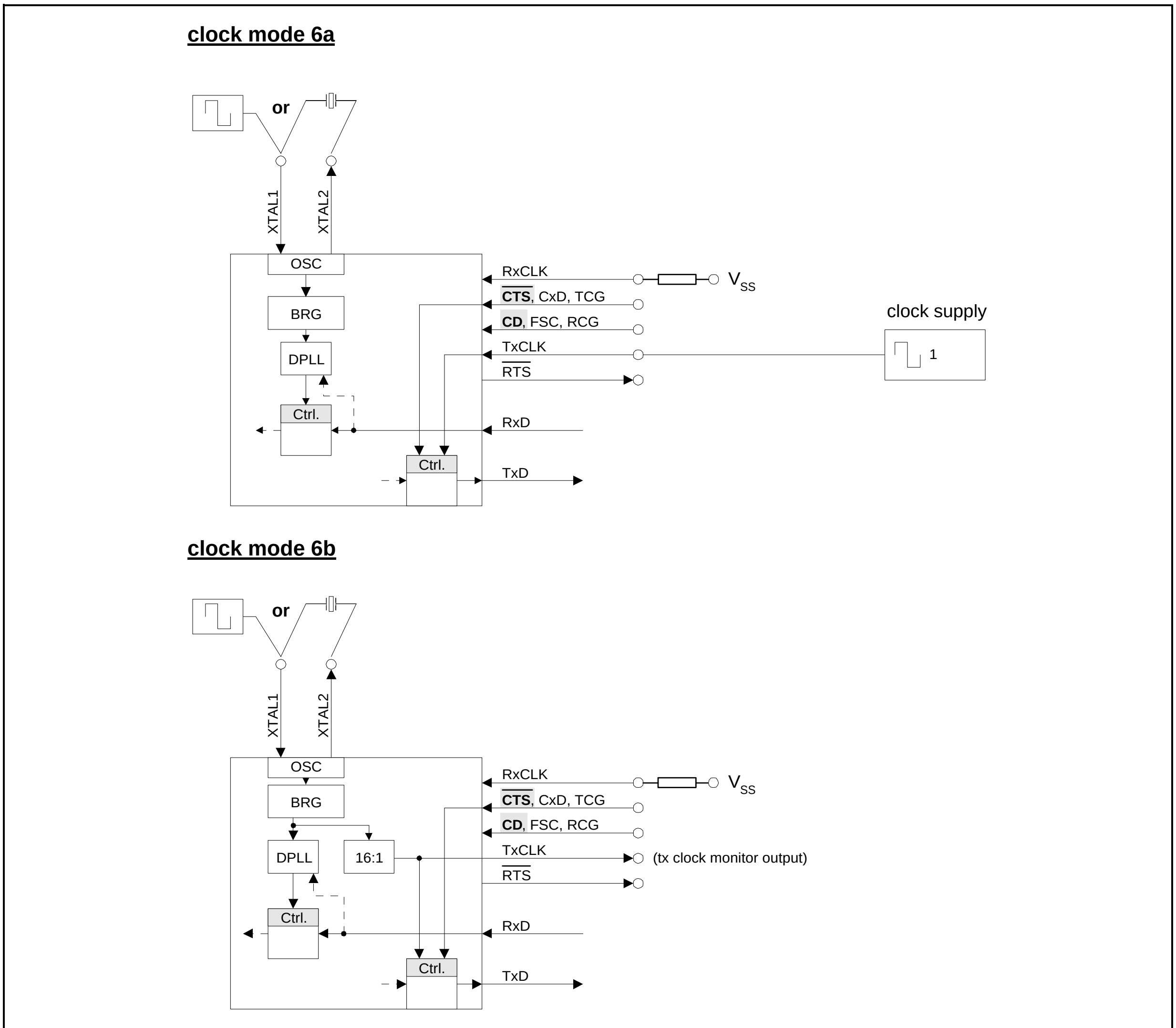


Figure 25 Clock Mode 6a/6b Configuration

3.2.3.9 Clock Mode 7 (7a/7b)

This clock mode is identical to clock mode 3a/3b except that the clock source of the BRG is supplied at pin XTAL1.

The BRG is driven by the internal oscillator. Depending on the value of bit 'SSEL' in register CCR0L the BRG delivers either a reference clock for the DPLL which is 16 times the resulting DPLL output frequency (clock mode 7a) or delivers directly the receive and transmit clock (clock mode 7b). In clock mode 7a the DPLL output clocks receive and transmit data.

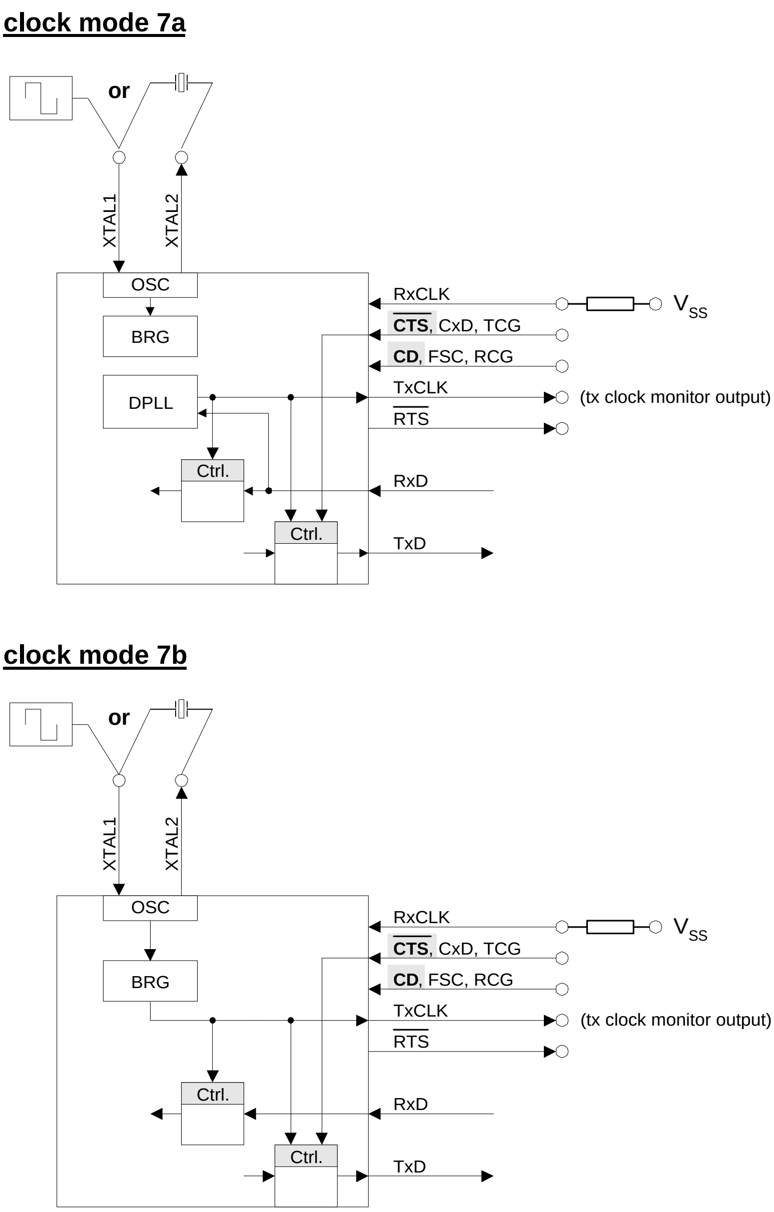


Figure 26 Clock Mode 7a/7b Configuration

3.2.4 Baud Rate Generator (BRG)

Each serial channel provides a baud rate generator (BRG) whose division factor is controlled by registers BRRL and BRRH. Whether the BRG is in the clocking path or not depends on the selected clock mode.

Table 9 BRRL/BRRH Register and Bit-Fields

Register	Bit-Fields			
Offset	Pos.	Name	Default	Description
BRRL $38_H/88_H$	5..0	BRN	0	Baud Rate Factor N range N = 0..63
BRRH $39_H/89_H$	11..8	BRM	0	Baud Rate Factor M, range M = 0..15

The clock division factor k is calculated by:

$$k = (N + 1) \times 2^M$$

$$f_{BRG} = f_{in} / k$$

3.2.5 Clock Recovery (DPLL)

The SCC offers the advantage of recovering the received clock from the received data by means of internal DPLL circuitry, thus eliminating the need to transfer additional clock information via a separate serial clock line. For this purpose, the DPLL is supplied with a 'reference clock' from the BRG which is 16 times the expected data clock rate (clock mode 2, 3a, 6, 7a). The transmit clock may be obtained by dividing the output of the BRG by a constant factor of 16 (clock mode 2b, 6b; bit 'SSEL' in register CCR0L set) or also directly from the DPLL (clock mode 3a, 7a).

The main task of the DPLL is to derive a receive clock and to adjust its phase to the incoming data stream in order to enable optimal bit sampling.

The mechanism for clock recovery depends on the selected data encoding (see **"Data Encoding" on Page 75**).

The following functions have been implemented to facilitate a fast and reliable synchronization:

Interference Rejection and Spike Filtering

Two or more edges in the same directional data stream within a time period of 16 reference clocks are considered to be interference and consequently no additional clock adjustment is performed.

Phase Adjustment (PA)

Referring to **Figure 27**, **Figure 28** and **Figure 29**, in the case where an edge appears in the data stream within the PA fields of the time window, the phase will be adjusted by 1/16 of the data.

Phase Shift (PS) (NRZ, NRZI only)

Referring to **Figure 27** in the case where an edge appears in the data stream within the PS field of the time window, a second sampling of the bit is forced and the phase is shifted by 180 degrees.

Note: Edges in all other parts of the time window will be ignored.

This operation facilitates a **fast** and reliable synchronization for most common applications. Above all, it implies a very fast synchronization because of the phase shift feature: one edge on the received data stream is enough for the DPLL to synchronize, thereby eliminating the need for synchronization patterns, sometimes called preambles. However, in case of **extremely** high jitter of the incoming data stream the reliability of the clock recovery cannot be guaranteed.

The SCC offers the option to disable the Phase Shift function for NRZ and NRZI encodings by setting bit ’PSD’ in register CCR0L to ’1’. In this case, the PA fields are extended as shown in **Figure 28**.

Now, the DPLL is more insensitive to high jitter amplitudes but needs **more time** to reach the optimal sampling position. To ensure correct data sampling, preambles should precede the data information.

Figure 27, Figure 28 and **Figure 29** explain the DPLL algorithms used for the different data encodings.

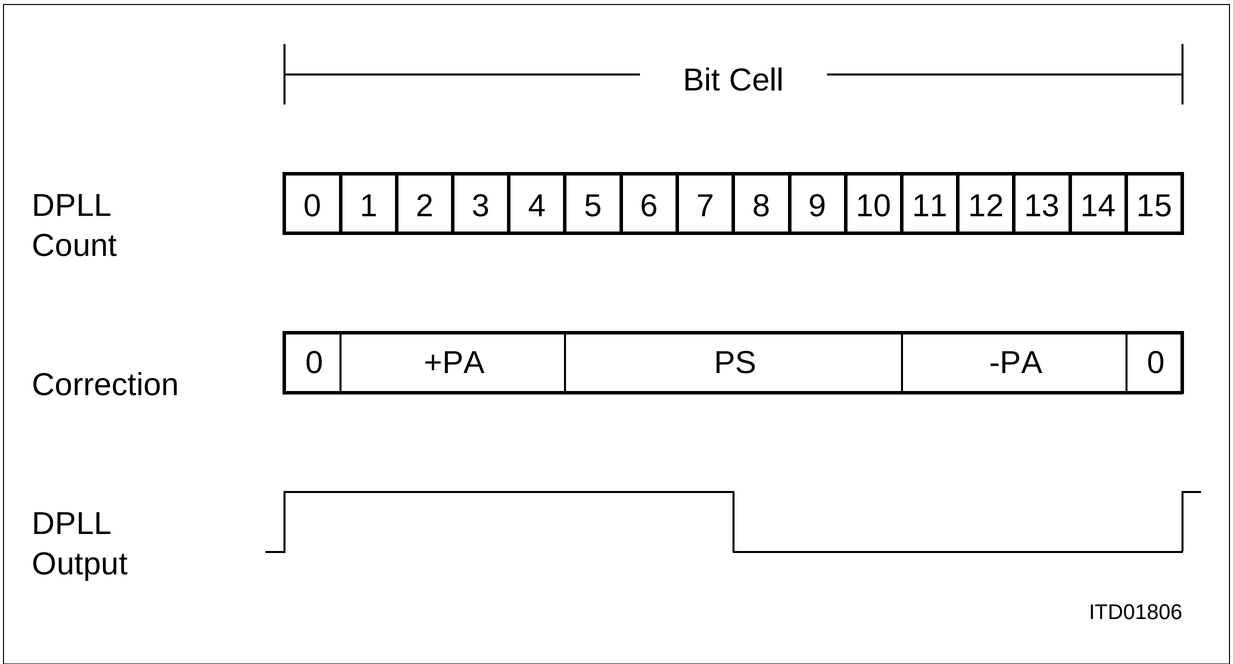


Figure 27 DPLL Algorithm (NRZ and NRZI Encoding, Phase Shift Enabled)

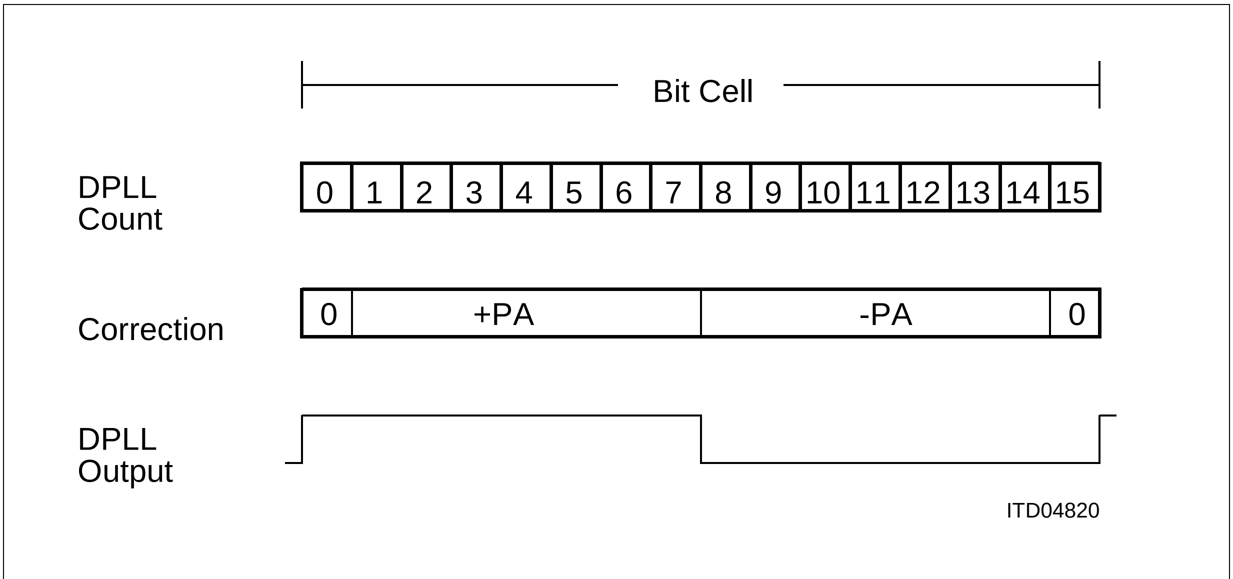


Figure 28 DPLL Algorithm (NRZ and NRZI Encoding, Phase Shift Disabled)

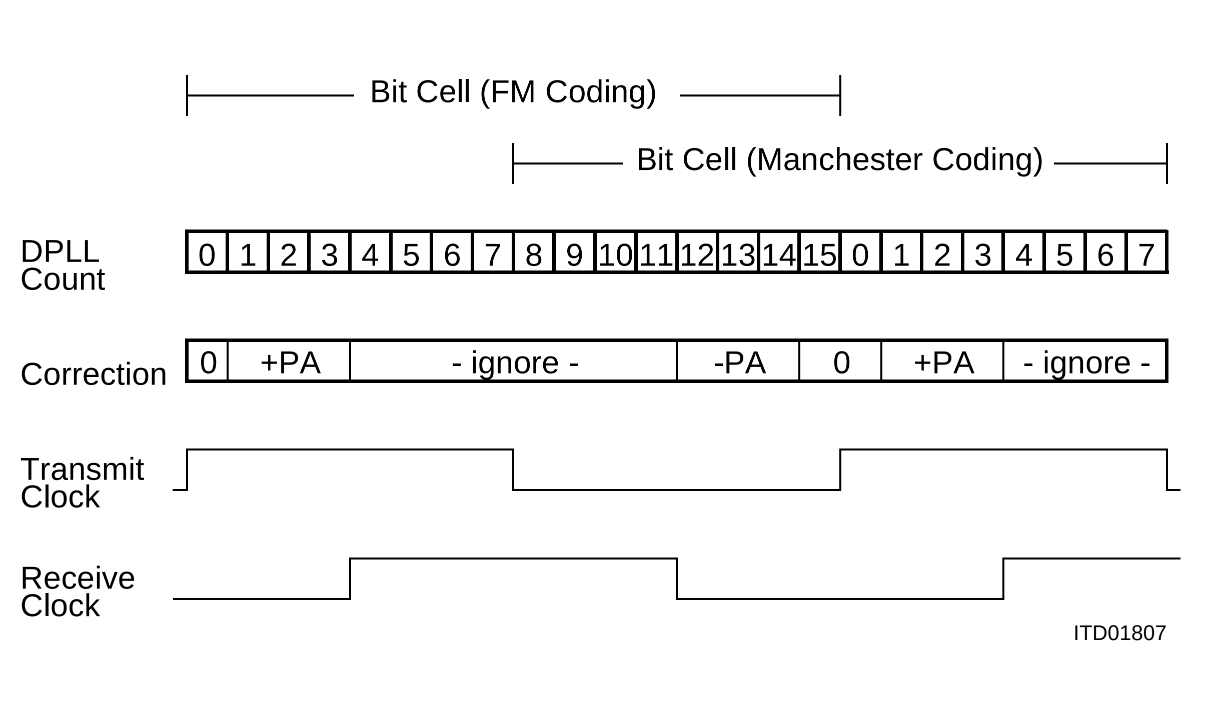


Figure 29 DPLL Algorithm for FM0, FM1 and Manchester Encoding

To supervise correct function when using bi-phase encoding, a status flag and a maskable interrupt inform about synchronous/asynchronous state of the DPLL.

3.2.6 SCC Timer Operation

Each SCC provides a general purpose timer e.g. to support protocol functions. In all operating modes the timer is clocked by the effective transmit clock. In clock mode 5 (time-slot oriented mode) the clock source for the timer can be optionally switched to the frame sync clock (input pin FSC) by setting bit 'SRC' in register TIMR3.

The timer is controlled by the CPU via access to registers CMDRL and TIMR0..TIMR3. The timer can be started any time by setting bit 'STI' in register CMDRL. After the timer has expired it generates a timer interrupt ('TIN').

With bit field 'CNT(2..0)' in register TIMR3 the number of automatic timer restarts can be programmed. If the maximum value '111' is entered, a timer interrupt is generated periodically, with the time period determined by bit field 'TVALUE' (registers TIMR0..TIMR3).

The timer can be stopped any time by setting bit 'TRES' in register CMDRL to '1'.

In HDLC Automode the timer is used internally for autonomous protocol functions (refer to the chapter **"Automode" on Page 87**). If this operating mode is selected, bit 'TMD' in register TIMR3 must be set to '1'.

3.2.7 SCC Serial Bus Configuration Mode

Beside the point-to-point configuration, the SCC effectively supports point-to-multipoint (pt-mpt, or bus) configurations by means of internal idle and collision detection/collision resolution methods.

In a pt-mpt configuration, comprising a central station (master) and several peripheral stations (slaves), or in a multimaster configuration, data transmission can be initiated by each station over a common transmit line (bus). In case more than one station attempts to transmit data simultaneously (collision), the bus has to be assigned to only one station. A collision-resolution procedure is implemented in the SCC. Bus assignment is based on a priority mechanism with rotating priorities. This allows each station a bus access within a predetermined maximum time delay (deterministic CSMA/CD), no matter how many transmitters are connected to the serial bus.

Prerequisites for bus operation are:

- NRZ encoding
- 'OR'ing of data from every transmitter on the bus (this can be realized as a wired-OR, using the TxD open drain capability)
- Feedback of bus information (CxD input).

The bus configuration is selected via bitfield SC(2:0) in register CCR0H.

Note: Central clock supply for each station is not necessary if both the receive and transmit clock is recovered by the DPLL (clock modes 3a, 7a). This minimizes the phase shift between the individual transmit clocks.

The bus configuration mode operates independently of the clock mode, e.g. also together with clock mode 1 (receive and transmit strobe operation).

3.2.8 Serial Bus Access Procedure

The idle state of the bus is identified by eight or more consecutive '1's. When a device starts transmission of a frame, the bus is recognized to be busy by the other devices at the moment the first 'zero' is transmitted (e.g. first 'zero' of the opening flag in HDLC mode).

After the frame has been transmitted, the bus becomes available again (idle).

Note: If the bus is occupied by other transmitters and/or there is no transmit request in the SCC, logical '1' will be continuously transmitted on TxD.

3.2.9 Serial Bus Collisions and Recovery

During the transmission, the data transmitted on TxD is compared with the data on CxD. In case of a mismatch ('1' sent and '0' detected, or vice versa) data transmission is immediately aborted, and idle (logical '1') is transmitted.

HDLC/SDLC: Transmission will be initiated again by the SCC as soon as possible if the first part of the frame is still present in the SCC transmit FIFO. If not, an XMR interrupt is generated.

Since a ‘zero’ (‘low’) on the bus prevails over a ‘1’ (high impedance) if a wired-OR connection is implemented, and since the address fields of the HDLC frames sent by different stations normally differ from one another, the fact that a collision has occurred will be detected prior to or at the latest within the address field. The frame of the transmitter with the highest temporary priority (determined by the address field) is not affected and is transmitted successfully. All other stations cease transmission immediately and return to bus monitoring state.

Note: If a wired-OR connection has been realized by an external pull-up resistor without decoupling, the data output (TxD) can be used as an open drain output and connected directly to the CxD input.
For correct identification as to which frame is aborted and thus has to be repeated after an XMR interrupt has occurred, the contents of SCC transmit FIFO have to be unique, i.e. SCC transmit FIFO should not contain data of more than one frame. For this purpose new data may be provided to the transmit FIFO only after ’ALLS’ interrupt status is detected.

3.2.10 Serial Bus Access Priority Scheme

To ensure that all competing stations are given a fair access to the transmission medium, a two-stage bus access priority scheme is supported by SEROCCO-D:

Once a station has successfully completed the transmission of a frame, it is given a lower level of priority. This priority mechanism is based on the requirement that a station may attempt transmitting only when a determined number of consecutive ‘1’s are detected on the bus.

Normally, a transmission can start when eight consecutive ‘1’s on the bus are detected (through pin CxD). When an HDLC frame has been successfully transmitted, the internal priority class is decreased. Thus, in order for the same station to be able to transmit another frame, ten consecutive ‘1’s on the bus must be detected. This guarantees that the transmission requests of other stations are satisfied before the same station is allowed a second bus access. When ten consecutive ‘1’s have been detected, transmission is allowed again and the priority class (of all stations) is increased (to eight ‘1’s).

Inside a priority class, the order of transmission (individual priority) is based on the HDLC address, as explained in the preceding paragraph. Thus, when a collision occurs, it is always the station transmitting the only ‘zero’ (i.e. all other stations transmit a ‘one’) in a bit position of the address field that wins, all other stations cease transmission immediately.

3.2.11 Serial Bus Configuration Timing Modes

If a bus configuration has been selected, the SCC provides two timing modes, differing in the time interval between sending data and evaluation of the transmitted data for collision detection.

- Timing mode 1 (CCR0H:SC(2:0) = ‘001’)
  Data is output with the rising edge of the transmit clock via the TxD pin, and evaluated 1/2 a clock period later at the CxD pin with the falling clock edge.
- Timing mode 2 (CCR0H:SC(2:0) = ‘011’)
  Data is output with the falling clock edge and evaluated with the next falling clock edge. Thus one complete clock period is available between data output and collision detection.

3.2.12 Functions Of Signal $\overline{\text{RTS}}$ in HDLC Mode

In clock modes 0 and 1, the $\overline{\text{RTS}}$ output can be programmed via register CCR1 (SOC bits) to be active when data (frame or character) is being transmitted. This signal is delayed by one clock period with respect to the data output TxD, and marks all data bits that could be transmitted without collision (see **Figure 30**). In this way a configuration may be implemented in which the bus access is resolved on a local basis (collision bus) and where the data are sent one clock period later on a separate transmission line.

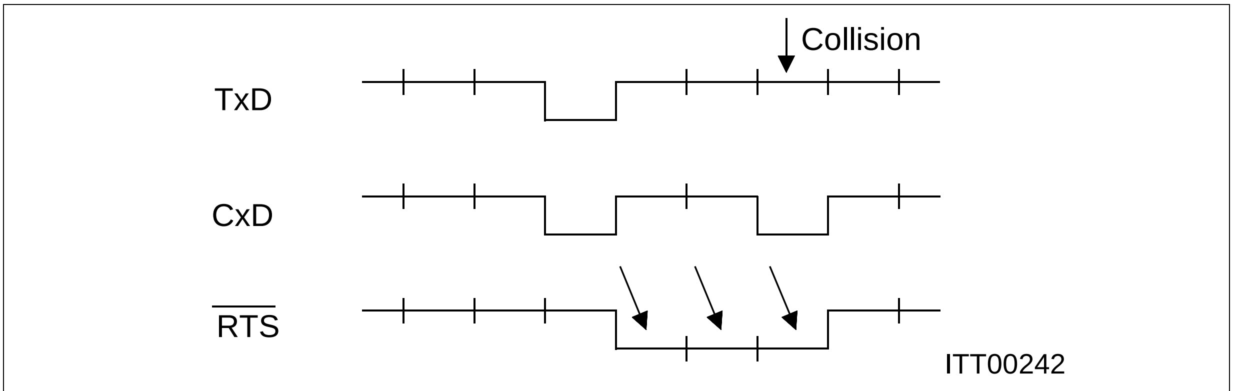


Figure 30 Request-to-Send in Bus Operation

*Note: For details on the functions of the $\overline{\text{RTS}}$ pin refer to **“Modem Control Signals (RTS, CTS, CD)” on Page 78**.*

3.2.13 Data Encoding

The SCC supports the following coding schemes for serial data:

– Non-Return-To-Zero (NRZ)
– Non-Return-To-Zero-Inverted (NRZI)
– FM0 (also known as Bi-Phase Space)
– FM1 (also known as Bi-Phase Mark)

– Manchester (also known as Bi-Phase)

The desired line coding scheme can be selected via bit field 'SC(2:0)' in register CCR0H.

3.2.13.1 NRZ and NRZI Encoding

NRZ: The signal level corresponds to the value of the data bit. By programming bit 'DIV' (CCR1L register), the SCC may invert the transmission and reception of data.

NRZI: A logical '0' is indicated by a transition and a logical '1' by no transition at the beginning of the bit cell.

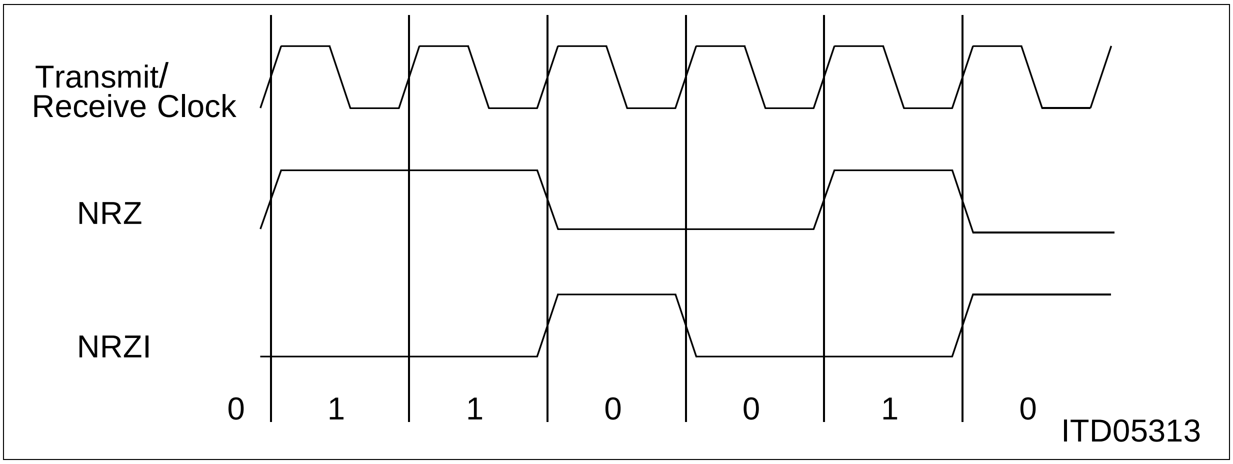


Figure 31 NRZ and NRZI Data Encoding

3.2.13.2 FM0 and FM1 Encoding

FM0: An edge occurs at the beginning of every bit cell. A logical '0' has an additional edge in the center of the bit cell, whereas a logical '1' has none. The transmit clock precedes the receive clock by 90°.

FM1: An edge occurs at the beginning of every bit cell. A logical '1' has an additional edge in the center of the bit cell, a logical '0' has none. The transmit clock precedes the receive clock by 90°.

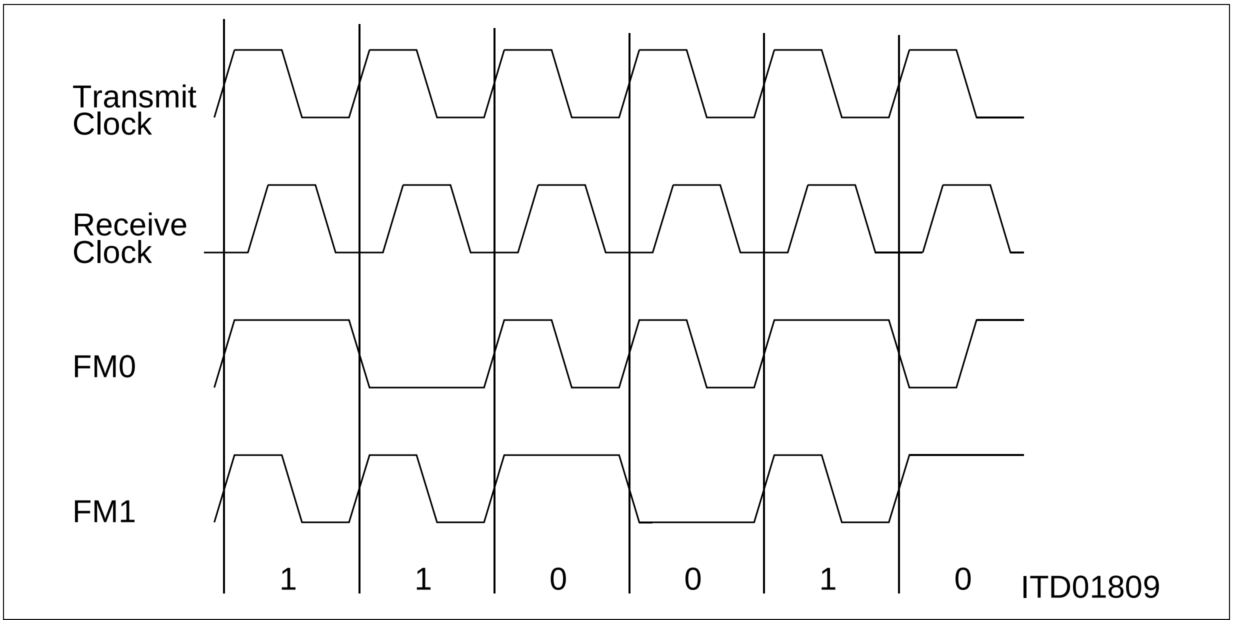


Figure 32 FM0 and FM1 Data Encoding

3.2.13.3 Manchester Encoding

Manchester: In the first half of the bit cell, the physical signal level corresponds to the logical value of the data bit. At the center of the bit cell this level is inverted. The transmit clock precedes the receive clock by 90°. The bit cell is shifted by 180° in comparison with FM coding.

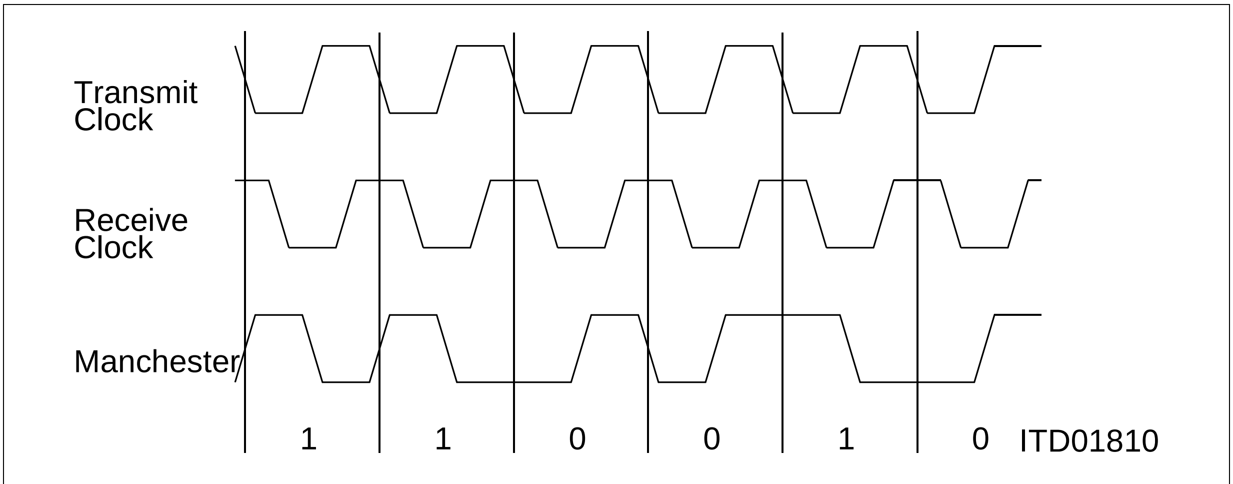


Figure 33 Manchester Data Encoding

3.2.14 Modem Control Signals ($\overline{\text{RTS}}$, $\overline{\text{CTS}}$, CD)

3.2.14.1 $\overline{\text{RTS}}$/$\overline{\text{CTS}}$ Handshaking

The SCC provides two pins ($\overline{\text{RTS}}$, $\overline{\text{CTS}}$) per serial channel supporting the standard request-to-send modem handshaking procedure for transmission control.

A transmit request will be indicated by outputting logical ‘0’ on the request-to-send output ($\overline{\text{RTS}}$). It is also possible to control the $\overline{\text{RTS}}$ output by software. After having received the permission to transmit ($\overline{\text{CTS}}$) the SCC starts data transmission.

In the case where permission to transmit is withdrawn in the course of transmission, the frame is aborted and IDLE is sent. After transmission is enabled again by re-activation of $\overline{\text{CTS}}$, and if the beginning of the frame is still available in the SCC, the frame will be re-transmitted (self-recovery). However, if the permission to transmit is withdrawn after the data available in the shadow part of the SCC transmit FIFO has been completely transmitted and the pool is released, the transmitter and the SCC transmit FIFO are reset, the $\overline{\text{RTS}}$ output is deactivated and an interrupt (XMR) is generated.

Note: For correct identification as to which frame is aborted and thus has to be repeated after an XMR interrupt has occurred, the contents of SCC transmit FIFO have to be unique, i.e. SCC transmit FIFO should not contain data of more than one frame, which could happen if transmission of a new frame is started by providing new data to the transmitter too early. For this purpose the ’All Sent’ interrupt (ISR1.ALLS) has to be waited for before providing new transmit data.

Note: In the case where permission to transmit is not required, the $\overline{\text{CTS}}$ input can be connected directly to V_{SS} and/or bit ’FCTS’ (register CCR1H) may be set to ’1’.

Additionally, any transition on the $\overline{\text{CTS}}$ input pin, sampled with the transmit clock, will generate an interrupt indicated via register ISR1, if this function is enabled by setting the ’CSC’ bit in register IMR1 to ’0’.

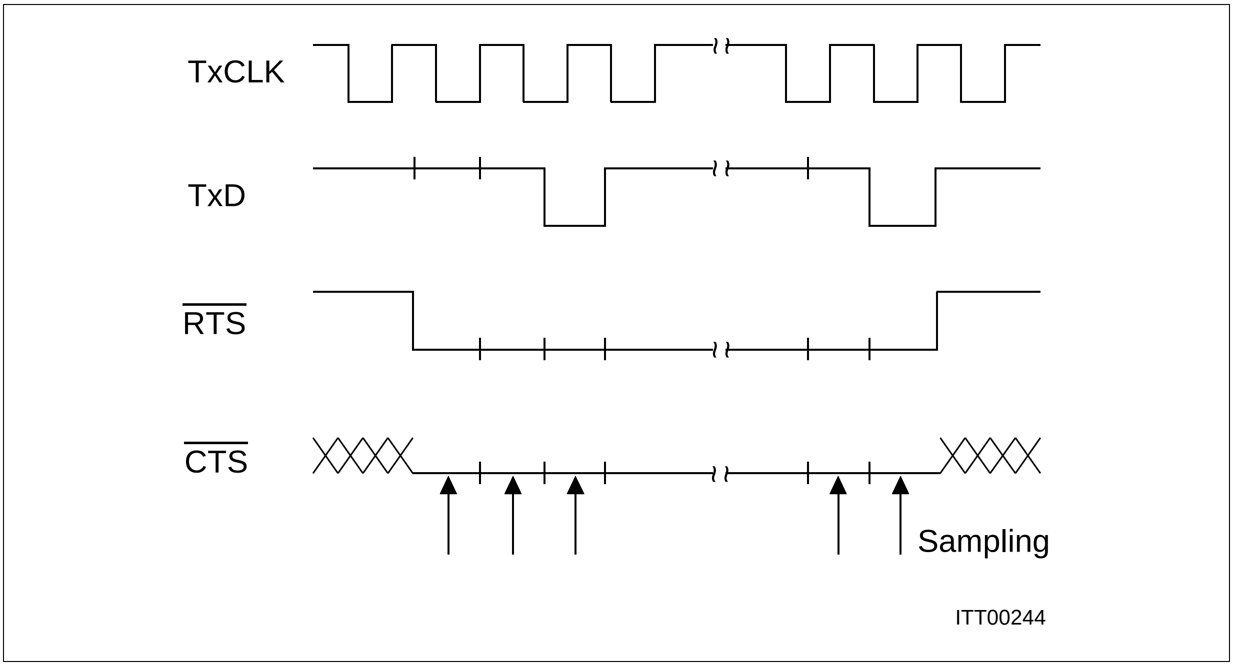


Figure 34 $\overline{RTS}$/$\overline{CTS}$ Handshaking

Beyond this standard $\overline{RTS}$ function, signifying a transmission request of a frame (Request To Send), in HDLC mode the $\overline{RTS}$ output may be programmed for a special function via SOC1, SOC0 bits in the CCR1L register. This is only available if the serial channel is operating in a bus configuration mode in clock mode 0 or 1.

- If SOC1, SOC0 bits are set to ‘11’, the $\overline{RTS}$ output is active (= low) during the reception of a frame.
- If SOC1, SOC0 bits are set to ‘10’, the $\overline{RTS}$ output function is disabled and the $\overline{RTS}$ pin remains always high.

3.2.14.2 Carrier Detect (CD) Receiver Control

Similar to the $\overline{RTS}$/$\overline{CTS}$ control for the transmitter, the SCC supports the carrier detect modem control function for the serial receiver if the Carrier Detect Auto Start (CAS) function is programmed by setting the ’CAS’ bit in register CCR1H. This function is always available in clock modes 0, 2, 3, 6, 7 via the CD pin. In clock mode 1 the CD function is not supported. See **Table 8** for an overview.

If the CAS function is selected, the receiver is enabled and data reception is started when the CD input is detected to be high. If CD input is set to ‘low’, reception of the current character (byte) is still completed.

3.2.15 Local Loop Test Mode

To provide fast and efficient testing, the SCC can be operated in a test mode by setting the ’TLP’ bit in register CCR2L. The on-chip serial data input and output signals (TxD, RxD) are connected, generating a local loopback. As a result, the user can perform a self-test of the SCC.

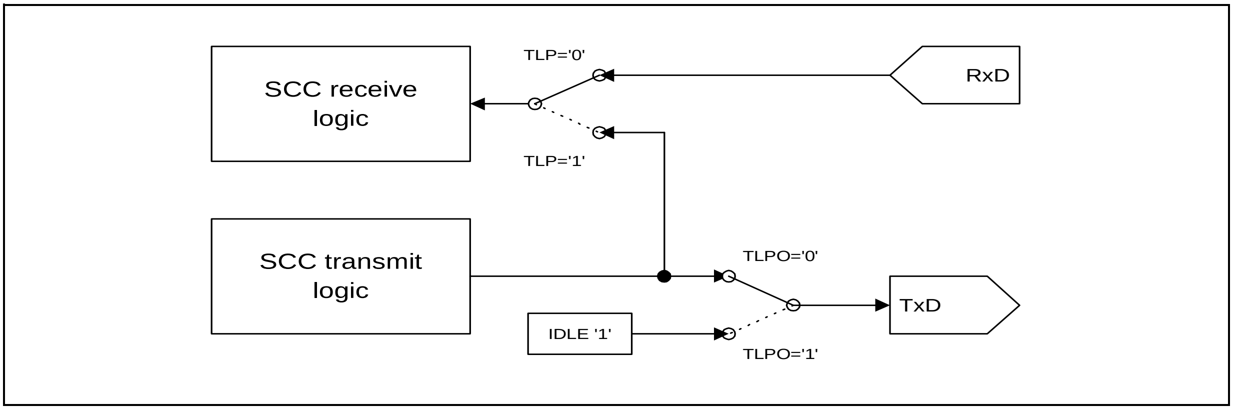


Figure 35 SCC Test Loop

Transmit data can be disconnected from pin TxD by setting bit TLPO in register CCR2L.

Note: A sufficient clock mode must be used for test loop operation such that receiver and transmitter operate with the same frequencies depending on the clock supply (e.g. clock mode 2b or 6b).

3.3 Microprocessor Interface

The communication between the CPU and SEROCCO-D is done via a set of directly accessible registers. The interface may be configured as Intel or Motorola type (refer to description of pin 'BM') with a selectable data bus width of 8 or 16 bit (refer to description of pin 'WIDTH').

The CPU transfers data to/from SEROCCO-D (via 64 byte deep FIFOs per direction and channel), sets the operating modes, controls function sequences, and gets status information by writing or reading control/status registers.

All accesses can be done as byte or word accesses if enabled. If 16-bit bus width is selected, access to the lower/upper part of the data bus is determined by signals $\overline{\text{BHE}}$/$\overline{\text{BLE}}$ as shown in **Table 10** (Intel mode) or by the upper and lower data strobe signals $\overline{\text{UDS}}$/$\overline{\text{LDS}}$ as shown in **Table 11** (Motorola mode).

Table 10 Data Bus Access 16-bit Intel Mode

$\overline{\text{BHE}}$	$\overline{\text{BLE}}$	Register Access	Data Pins Used
0	0	Word access (16 bit)	D(15:0)
0	1	Byte access (8 bit), odd address	D(15:8)

Table 10 Data Bus Access 16-bit Intel Mode

$\overline{BHE}$	$\overline{BLE}$	Register Access	Data Pins Used
1	0	Byte access (8 bit), even address	D(7:0)
1	1	no data transfer	-

Table 11 Data Bus Access 16-bit Motorola Mode

$\overline{UDS}$	$\overline{LDS}$	Register Access	Data Pins Used
0	0	Word access (16 bit)	D(15:0)
0	1	Byte access (8 bit), even address	D(15:8)
1	0	Byte access (8 bit), odd address	D(7:0)
1	1	no data transfer	-

Each of the two serial channels of SEROCCO-D is controlled via an identical, but completely independent register set (Channel A and B). Global functions that are common to or independent from the two serial channels are located in global registers.

3.4 Internal DMA Controller

3.4.1 Arbitration for Bus Control

Every time SEROCCO-D needs to access the bus in order to DMA transfer receive data from the RFIFO to host memory or transmit data from host memory to the XFIFO, it has to request the bus arbiter for the bus mastership. This is achieved by asserting the open-drain $\overline{BREQ}$ signal to low. When SEROCCO-D samples the bus grant ($\overline{BGNT}$) active, it acknowledges bus ownership by asserting the bus grant acknowledge ($\overline{BGACK}$) line to low.

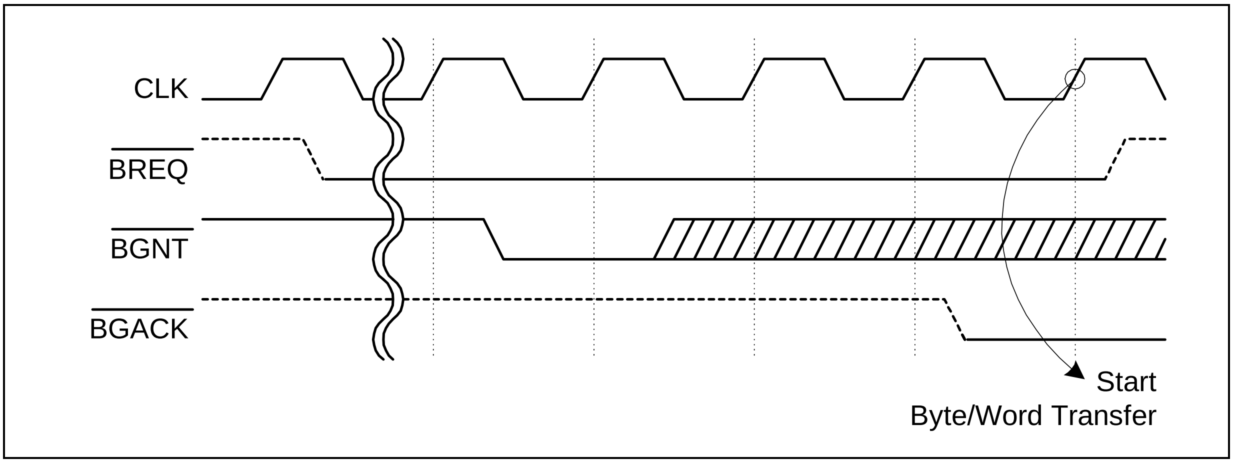


Figure 36 SEROCCO-D requests and gets the bus

The $\overline{BREQ}$ signal becomes inactive one clock later and DMA transfer cycles start.

SEROCCO-D holds the $\overline{BGACK}$ line low for the time it performs DMA read or write cycles on the bus.

3.4.2 Performing DMA Transfers

The maximum number of bus transfers in sequence is 16 (word transfers in 16-bit bus modes) or 32 (byte transfers in 8-bit bus modes). Each DMA initiated read and write cycle is performed in four clock cycles, see **Figure 37** with numbering of the cycle sections in Intel (T1/T2) and Motorola (S0..S7) fashion.

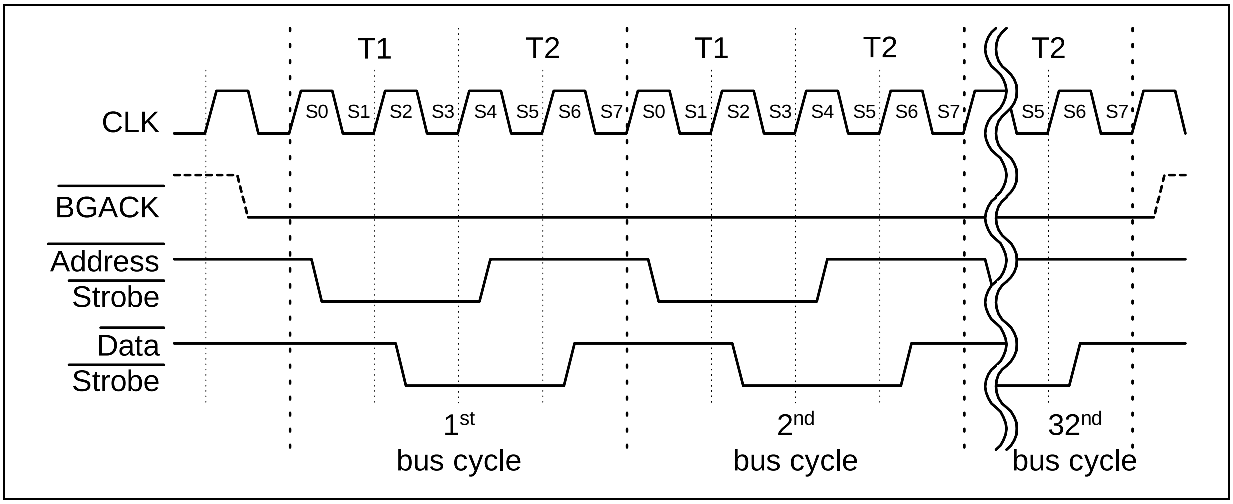


Figure 37 Un-interrupted Series of 32 DMA Bus Cycles

3.4.3 Bus Preemption

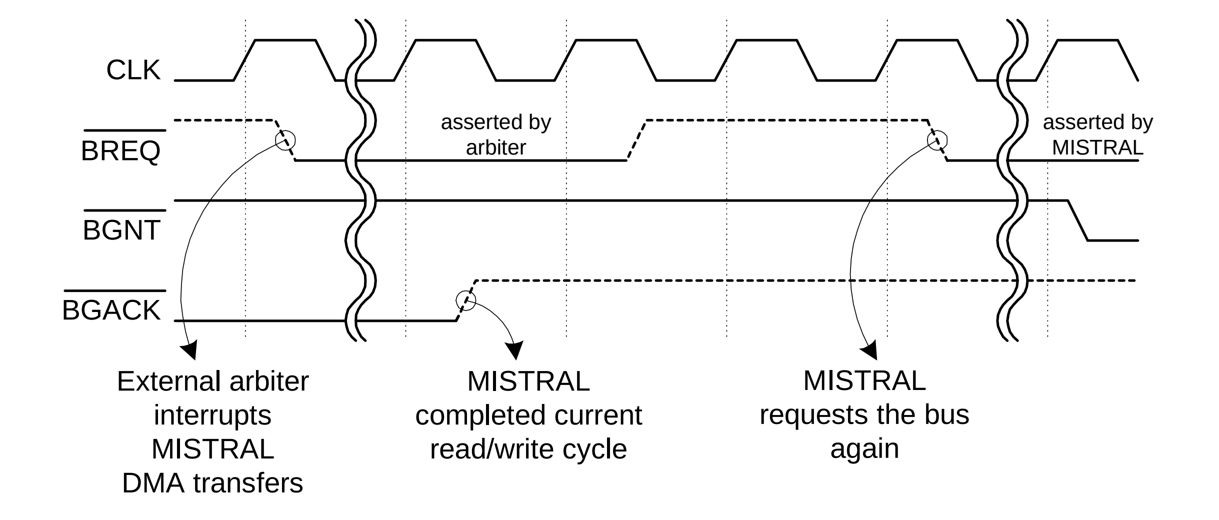


Figure 38 Bus Preemption and Re-gain of Bus Control

3.4.4 Ending DMA Transfers

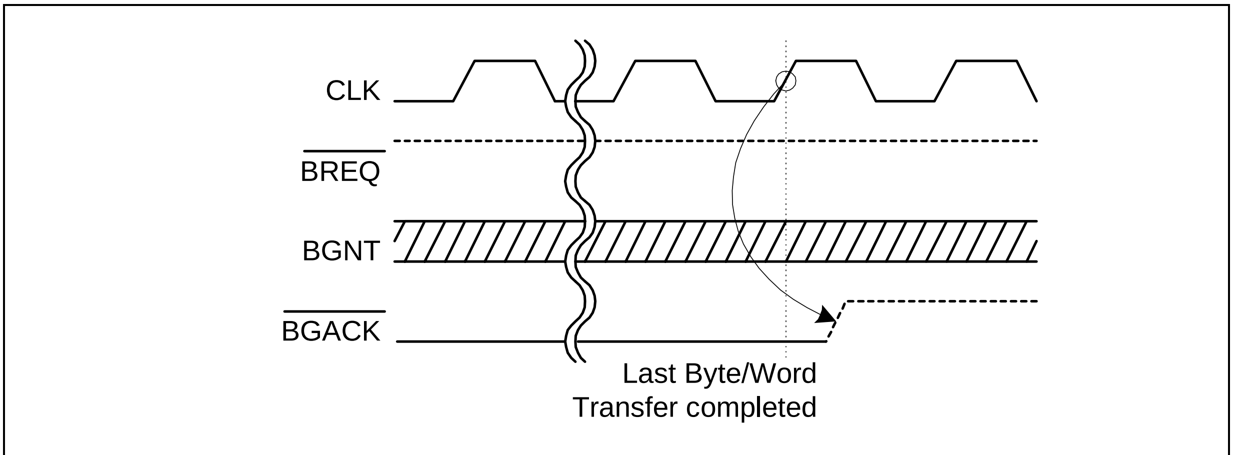


Figure 39 SEROCCO-D requests and gets the bus

3.5 Interrupt Architecture

For certain events in SEROCCO-D an interrupt can be generated, requesting the CPU to read status information from SEROCCO-D. The interrupt line INT/$\overline{INT}$ is asserted with the output characteristics programmed in bit field 'IPC(1..0)' in register **"GMODE" on Page 127** (open drain/push pull, active low/high).

Since only one interrupt request output is provided, the cause of an interrupt must be determined by the CPU by reading the interrupt status registers (GSTAR, ISR0, ISR1, ISR2, DISR, GPIS).

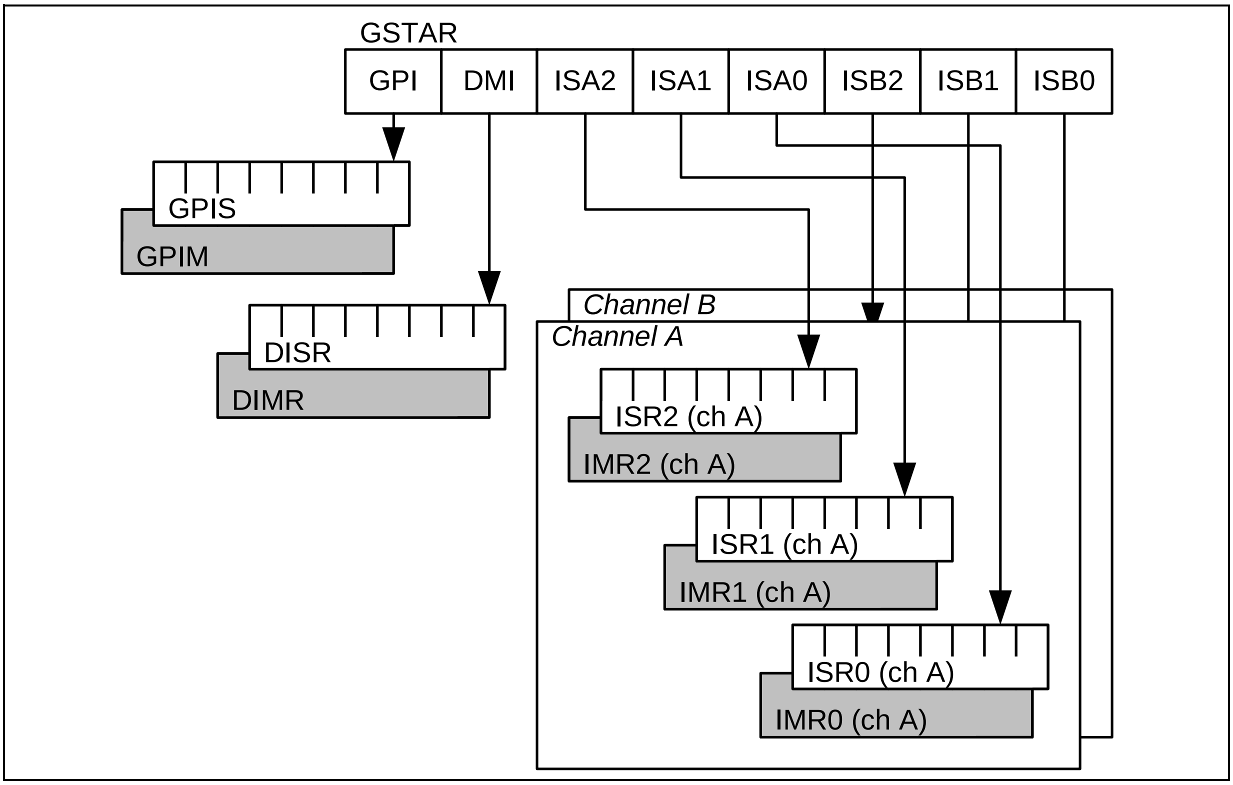


Figure 40 Interrupt Status Registers

Each interrupt indication of registers ISR0, ISR1, ISR2, DISR and GPIS can be selectively unmasked by resetting the corresponding bit in the corresponding mask registers IMR0, IMR1, IMR2, DIMR and GPIM. Use of these registers depends on the selected serial mode.

If bit 'VIS' in register CCR0L is set to '1', masked interrupt status bits are visible in the interrupt status registers ISR0..ISR2. Interrupts masked in registers IMR0..IMR2 will not generate an interrupt though. A read access to the interrupt status registers clears the bits.

A global interrupt mask bit (bit 'GIM' in register GMODE) suppresses interrupt generation at all. To enable the interrupt system after reset, this bit must be set to '0'.

The Global Interrupt Status Register (GSTAR) serves as pointer to pending channel related interrupts and general purpose port interrupts.

3.6 General Purpose Port Pins

3.6.1 GPP Functional Description

General purpose pins are provided on pins GP0...GP2.

Every pin is separately programmable via the General Purpose Port Direction register GPDIR to operate as an output (bit GPnDIR=’0’) or as an input (bit GPnDIR=’1’, reset value).

If defined as output, the state of the pin is directly controlled via the General Purpose Port Data register GPDAT. Read access to these registers delivers the current state of all GPP pins (input and output signals).

If defined as input, the state of the pin is monitored. The signal state of the corresponding GP pins is sampled with a rising edge of CLK and is readable via register GPDAT.

3.6.2 GPP Interrupt Indication

The GPP block generates interrupts for transitions on each input signal. All changes may be indicated via interrupt (optional). To enable interrupt generation, the corresponding interrupt mask bit in register GPIM must be reset to ’0’.

Bit GPI in the gloabl interrupt status register (GSTAR) is set to ’1’ if an interrupt was generated by any one or more of the the general purpose port pins. The GPP pin causing the interrupt can be located by reading the GPIS register.

4 Detailed Protocol Description

The following **Table 12** provides an overview of all supported protocol modes and . The desired protocol mode is selected via bit fields in the channel configuration registers CCR0L, CCR0H, CCR2L and CCR3L.

Table 12 Protocol Mode Overview

Register CCR0H - Bit Field SM(1:0) = ’00’ (HDLC/SDLC/PPP protocol engine)		Register CCR2L - Bit Field: MDS	ADM	PPPM	CCR3L ESS7
HDLC Automode (LAP D / LAP B / SDLC-NRM)	16 bit	’00’	’1’	’00’	’0’
	8 bit	’00’	’0’		
HDLC Address Mode 2	16 bit	’01’	’1’		
	8 bit	’01’	’0’		
HDLC Address Mode 1		’10’	’1’		
HDLC Address Mode 0		’10’	’0’		
Signaling System #7 (SS7) Operation		’10’	’0’	’00’	’1’
Bit Synchronous PPP Mode		’10’	’0’	’11’	’0’
Octet Synchronous PPP Mode				’01’	
Asynchronous PPP Mode				’10’	
Extended Transparent Mode[1]		’11’	’1’	’00’	’0’

Register CCR0H - Bit Field SM(1:0) = ’11’ (ASYNC protocol engine)	Register CCR0L - Bit Field: BCR
Asynchronous Mode	’1’
Isochronous Mode	’0’

Register CCR0H - Bit Field SM(1:0) = ’10’ (BISYNC protocol engine)	Register CCR0L - Bit Field: EBIM
Bisynchronous Mode	’1’
Monosynchronous Mode	’0’

[1] Extended transparent mode is a fully bit-transparent transmission/reception mode which is treated as sub-mode of the HDLC/SDLC/PPP block.

All modes are discussed in details in this chapter.

4.1 HDLC/SDLC Protocol Modes

The HDLC controller of each serial channel (SCC) can be programmed to operate in various modes, which are different in the treatment of the HDLC frame in receive direction. Thus, the receive data flow and the address recognition features can be performed in a very flexible way satisfying almost any application specific requirements.

There are 4 different HDLC operating modes which can be selected via register bits CCR2L:MDS[1:0] and CCR2L:ADM.

4.1.1 HDLC Submodes Overview

The following table provides an overview of the different address comparison mechanisms in HDLC operating modes:

Table 13 Address Comparison Overview

Mode	Address Field	Recognized Address Bytes for a Match:		
		High Address Byte		Low Address Byte
Address Mode 2 - Auto Mode	16 bit	FE_H / FC_H (1111 11 C/R 0_2)	*and*	RAL1
		FE_H / FC_H (1111 11 C/R 0_2)	*and*	RAL2
		RAH1	*and*	RAL1
		RAH1	*and*	RAL2
		RAH2	*and*	RAL1
		RAH2	*and*	RAL2
	8 bit	RAL1		*don't care*
		RAL2		*don't care*
Address Mode 1	8 bit	FE_H / FC_H (1111 11 C/R 0_2)		*don't care*
		RAH1		*don't care*
		RAH2		*don't care*
Address Mode 0	None	*don't care*		*don't care*

4.1.1.1 Automode

Characteristics: Window size 1, random message length, address recognition.

The SCC processes autonomously all numbered frames (S-, I-frames) of an HDLC protocol. The HDLC control field, I-field data of the frames and an additional status byte are temporarily stored in the SCC receive FIFO.

Depending on the selected address mode, the SCC can perform a 2-byte or 1-byte address recognition.

If a 2-byte address field is selected, the high address byte is compared with the fixed value FE_H or FC_H (group address) as well as with two individually programmable values in RAH1 and RAH2 registers. According to the ISDN LAPD protocol, bit 1 of the high byte address will be interpreted as COMMAND/RESPONSE bit (C/R), depending on the setting of the CRI bit in RAH1, and will be excluded from the address comparison.

Similarly, two comparison values can be programmed in special registers (RAL1, RAL2) for the low address byte. A valid address will be recognized in case the high and low byte of the address field correspond to one of the compare values. Thus, the SCC can be called (addressed) with 6 different address combinations, however, only the logical connection identified through the address combination RAH1/RAL1 will be processed in the auto-mode, all others in the non auto-mode. HDLC frames with address fields that do not match any of the address combinations, are ignored by the SCC.

In the case of a 1-byte address, only RAL1 and RAL2 will be used as comparison values. According to the X.25 LAPB protocol, the value in RAL1 will be interpreted as COMMAND and the value in RAL2 as RESPONSE.

The address bytes can be masked to allow selective broadcast frame recognition. For further information see **"Receive Address Handling" on Page 91**.

4.1.1.2 Address Mode 2

Characteristics: address recognition, arbitrary window size.

All frames with valid addresses (address recognition identical to auto-mode) are forwarded directly to the RFIFO.

The HDLC control field, I-field data and an additional status byte are temporarily stored in the SCC receive FIFO.

In address mode 2, all frames with a valid address are treated similarly.

The address bytes can be masked to allow selective broadcast frame recognition.

4.1.1.3 Address Mode 1

Characteristics: address recognition high byte.

Only the high byte of a 2-byte address field will be compared. The address byte is compared with the fixed value FE_H or FC_H (group address) as well as with two individually programmable values RAH1 and RAH2. The whole frame excluding the first address byte will be stored in the SCC receive FIFO.

The address bytes can be masked to allow selective broadcast frame recognition.

4.1.1.4 Address Mode 0

Characteristics: no address recognition

No address recognition is performed and each complete frame will be stored in the SCC receive FIFO.

4.1.2 HDLC Receive Data Processing

The following figures give an overview about the management of the received frames in the different HDLC operating modes. The graphics show the actual HDLC frame and how SEROCCO-D interprets the incoming octets. Below that it is shown which octets are stored in the RFIFO and will thus be transferred into memory.

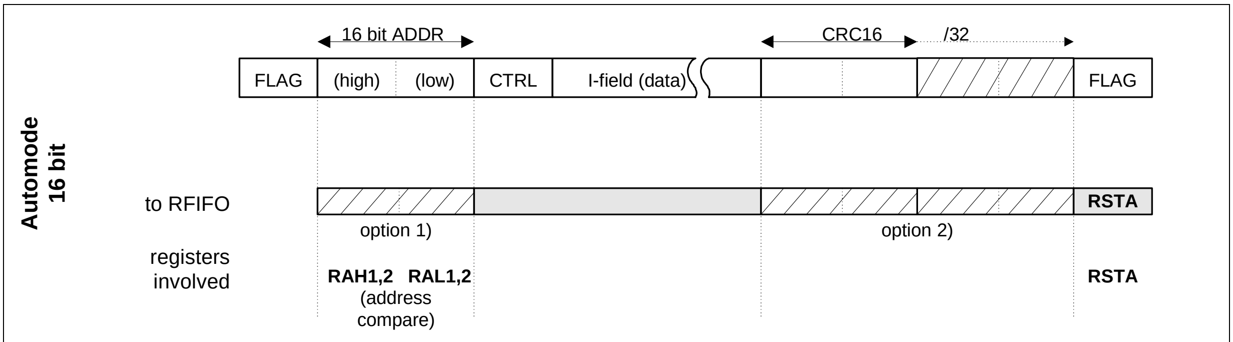


Figure 41 HDLC Receive Data Processing in 16 bit Automode

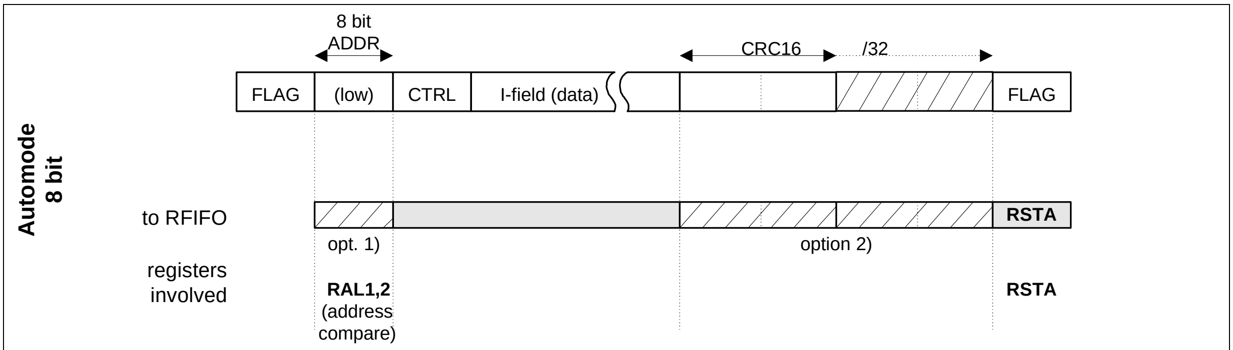


Figure 42 HDLC Receive Data Processing in 8 bit Automode

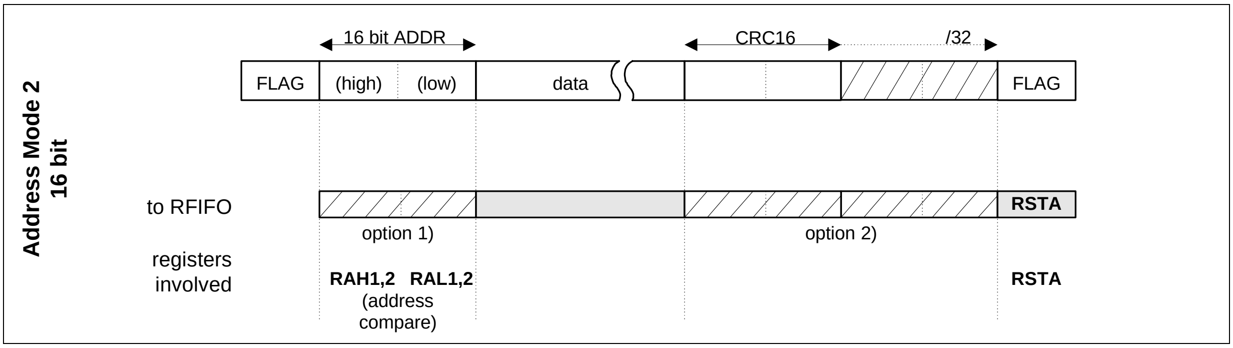


Figure 43 HDLC Receive Data Processing in Address Mode 2 (16 bit)

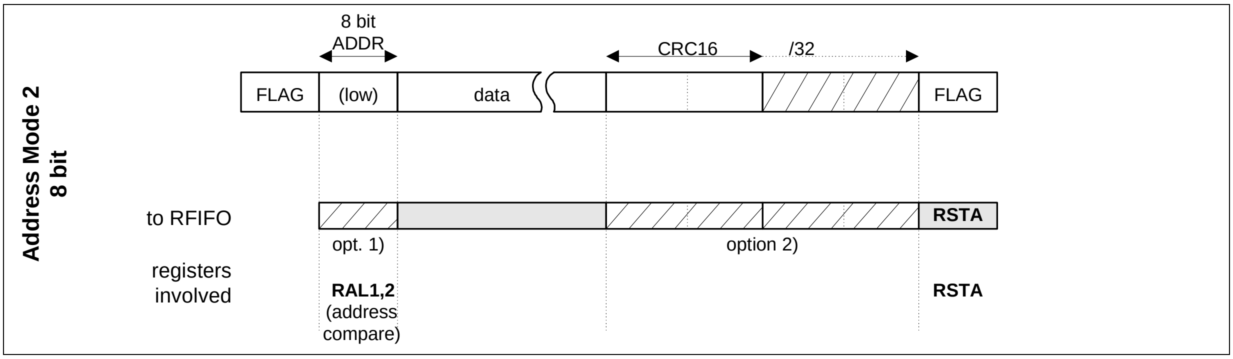


Figure 44 HDLC Receive Data Processing in Address Mode 2 (8 bit)

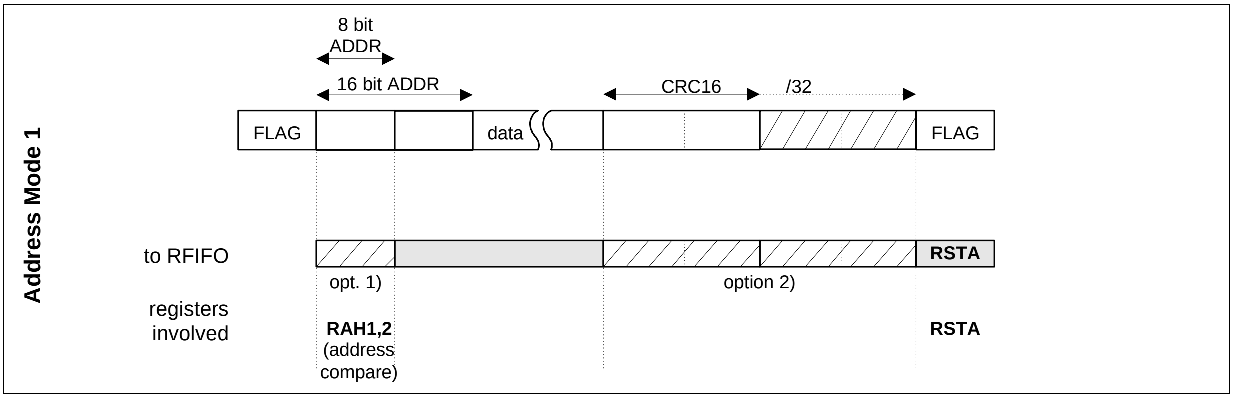


Figure 45 HDLC Receive Data Processing in Address Mode 1

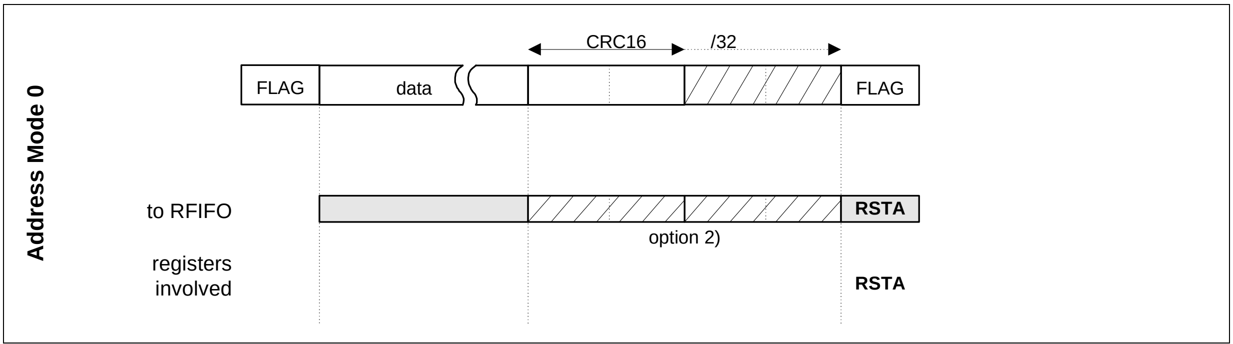


Figure 46 HDLC Receive Data Processing in Address Mode 0

option 1)

The address field (8 bit address, 16 bit address or the high byte of a 16 bit address) can optionally be forwarded to the RFIFO (bit 'RADD' in register CCR3H)

option 2)

The 16 bit or 32 bit CRC field can optionally be forwarded to the RFIFO (bit 'RCRC' in register CCR3H)

4.1.3 Receive Address Handling

The Receive Address Low/High Bytes (registers RAL1/RAH1 and RAL2/RAH2) can be masked on a per bit basis by setting the corresponding bits in the mask registers AMRAL1/AMRAH1 and AMRAL2/AMRAH2. This allows extended broadcast address recognition. Masked bit positions always match in comparison of the received frame address with the respective address fields in the Receive Address Low/High registers.

This feature is applicable to all HDLC protocol modes with address recognition (auto mode, address mode 2 and address mode 1). It is disabled if all bits of mask bit fields AMRAL1/AMRAH1 and AMRAL2/AMRAH2 are set to 'zero' (which is the reset value).

Detection of the fixed group address FE_H or FC_H, if applicable to the selected operating mode, remains unchanged.

As an option in the auto mode, address mode 2 and address mode 1, the 8/16 bit address field of received frames can be pushed to the receive data buffer (first one/two bytes of the frame). This function is especially useful in conjunction with the extended broadcast address recognition. It is enabled by setting control bit 'RADD' in register CCR3H.

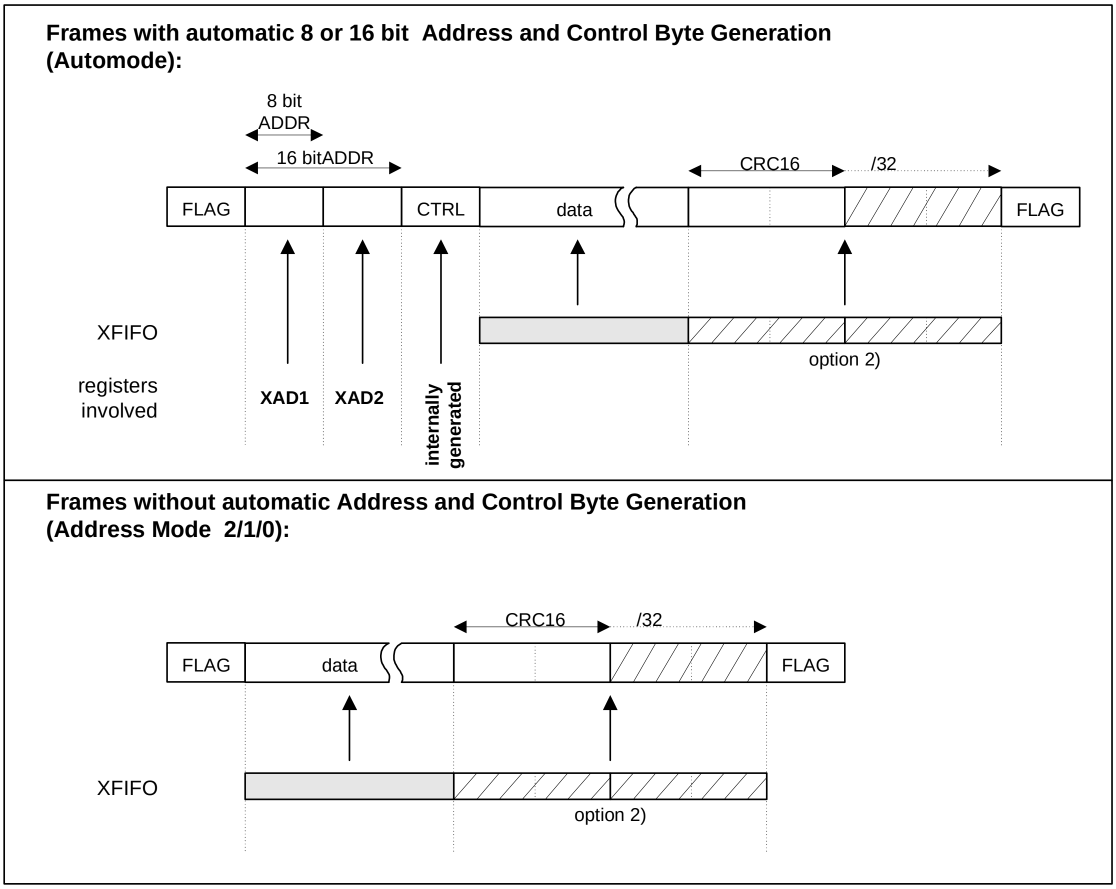
4.1.4 HDLC Transmit Data Processing

Two different types of frames can be transmitted:

– I-frames and

– transparent frames

as shown below.



option 2)
Generation of the 16 or 32 bit CRC field can optionally be disabled by setting bit 'XCRC' in register CCR2H, in which case the CRC must be calculated and written into the last 2 or 4 bytes of the transmit FIFO, to immediately proceed closing flag.

Figure 47 SCC Transmit Data Flow (HDLC Modes)

For transmission of I-frames (selected via transmit command ’XIF’ in register CMDRL), the address and control fields are generated autonomously by the SCC and the data in the corresponding transmit data buffer is entered into the information field of the frame. This is possible only if the SCC is operated in Automode.

For (address-) transparent frames, the address and the control fields have to be entered in the transmit data buffer by software. This is possible in all operating modes and used also in auto-mode for sending U-frames.

If bit ’XCRC’ in register CCR2H is set, the CRC checksum will not be generated internally. The checksum has to be provided via the transmit data buffer as the last two or four bytes by software. The transmitted frame will be closed automatically only with a (closing) flag.

Note: The SCC does not check whether the length of the frame, i.e. the number of bytes, to be transmitted makes sense according the HDLC protocol or not.

4.1.5 Shared Flags

If the ‘Shared Flag’ feature is enabled by setting bit ’SFLG’ in register CCR1L the closing flag of a previously transmitted frame simultaneously becomes the opening flag of the following frame if there is one already available in the SCC transmit FIFO.

In receive direction the SCC always expects and handles ’Shared Flags’. ’Shared Zeroes’ of consecutive flags are also supported.

4.1.6 One Bit Insertion

Similar to the zero bit insertion (bit stuffing) mechanism, as defined by the HDLC protocol, the SCC offers a feature of inserting/deleting a ’one’ after seven consecutive ‘zeros’ into the transmit/receive data stream, if the serial channel is operating in bus configuration mode. This method is useful if clock recovery is performed by DPLL.

Since only NRZ data encoding is supported in a bus configuration, there are possibly long sequences without edges in the receive data stream in case of successive ‘0’s received, and the DPLL may lose synchronization.

Enabling the one bit insertion feature by setting bit ’OIN’ in register CCR2H, it is guaranteed that at least after

– 5 consecutive ‘1’s a ‘0’ will appear (bit stuffing), and after
– 7 consecutive ‘0’s a ‘1’ will appear (one insertion)

and thus a correct function of the DPLL is ensured.

Note: As with the bit stuffing, the ‘one insertion’ is fully transparent to the user, but it is not in accordance with the HDLC protocol, i.e. it can only be applied in proprietary systems using circuits that also implement this function, such as the PEB 20532 and PEB 20525.

4.1.7 Preamble Transmission

If enabled via bit ’EPT’ in register CCR2H, a programmable 8-bit pattern is transmitted with a selectable number of repetitions after Interframe Timefill transmission is stopped and a new frame is ready to be sent out. The 8 bit preamble pattern can be programmed in register PREAMB and the repetition time in bit field ’PRE’ of register CCR2H.

Note: Zero Bit Insertion is disabled during preamble transmission.

4.1.8 CRC Generation and Checking

In HDLC/SDLC mode, error protection is done by CRC generation and checking.

In standard applications, CRC-CCITT algorithm is used. The Frame Check Sequence at the end of each frame consists of two bytes of CRC checksum.

If required, the CRC-CCITT algorithm can be replaced by the CRC-32 algorithm, enabled via bit ’C32’ in register CCR1L. In this case the Frame Check Sequence consists of four bytes.

Optionally the internal handling of received and transmitted CRC checksum can be influenced via control bits ’RCRC’, ’DRCRC’ in register CCR3H and ’XCRC’ in register CCR2H.

Receive direction:
If not disabled by setting bit ’DRCRC’ (register CCR3H), the received CRC checksum is always assumed to be in the 2 (CRC-CCITT) or 4 (CRC-32) last bytes of a frame, immediately preceding a closing flag. If bit ’RCRC’ is set, the received CRC checksum is treated as data and will be forwarded to the RFIFO, where it precedes the frame status byte. Nevertheless the received CRC checksum is additionally checked for correctness. If CRC checking is disabled with bit CCR3H:DRCRC, the limits for ‘Valid Frame’ check are modified accordingly (refer to description of the Receive Status Byte, RSTA:VFR).

Transmit direction:
If bit ’XCRC’ is set, the CRC checksum is not generated internally. The checksum has to be provided via the transmit data buffer by software. The transmitted frame will only be closed automatically with a (closing) flag.

Note: The SCC does not check whether the length of the frame, i.e. the number of bytes, to be transmitted makes sense or not according the HDLC protocol.

4.1.9 Receive Length Check Feature

The SCC offers the possibility to supervise the maximum length of received frames and to terminate data reception in the case that this length is exceeded.

This feature is controlled via the special Receive Length Check Registers RLCRL/ RLCRH.

The function is enabled by setting bit ’RCE’ (Receive Length Check Enable) and the maximum frame length to be checked is programmed via bit field ’RL’. The maximum receive length can be determined as a multiple of 32-byte blocks as follows:

$MAX\_LENGTH = (RL + 1) \times 32$,

where RL is the value written to bit field ’RL’. Thus, the maximum length of receive frames can be programmed between 32 and 65536 bytes.

All frames exceeding this length are treated as if they had been aborted by the remote station, i.e. the CPU is informed via

– an 'RME' interrupt generated by the SCC, and
– the receive abort indication 'RAB' in the Receive Status Byte (RSTA).

Additionally an optional 'FLEX' interrupt is generated prior to 'RME', indicating that the maximum receive frame length was exceeded.

Receive operation continues with the beginning of the next receive frame.

4.2 Point-to-Point Protocol (PPP) Modes

PPP (as described in RFC1662) can work over 3 modes: asynchronous HDLC, synchronous HDLC, and octet synchronous. The SEROCCO-D supports asynchronous HDLC PPP over ISDN or DDS circuits as well as bit and octet synchronous HDLC PPP for use over dial-up connections. The octet synchronous mode of PPP protocol (RFC 1662) supports PPP over SONET applications.

Both the asynchronous HDLC PPP mode, as well as the synchronous HDLC PPP modes are submodes of the HDLC mode. Either mode is selected by configuring SEROCCO-D for the standard HDLC mode. In addition the appropriate PPP mode is selected via bit field 'PPPM' in register CCR2L.

The SEROCCO-D provides logic to convert an HDLC frame to an ASYNC character stream with the specified mapping functions. Layer 3 PPP functions are normally implemented in software.

The PPP-support hardware allows software to perform segmentation and reassembly of PPP payloads, and allows SEROCCO-D to perform the asynchronous HDLC PPP **or** the synchronous HDLC PPP protocol conversions as required for the network interface.

4.2.1 Bit Synchronous PPP

The SEROCCO-D transmits a data block, inserts HDLC Header (Opening Flag), and appends the HDLC Trailer (CRC, Ending Flag). Zero-bit stuffing algorithm is also performed. No character mapping is performed. The bit-synchronous PPP mode differs from the HDLC mode (address mode 0) only in the abort sequence:

HDLC requires at least 7 consecutive '1' bits as abort sequence, whereas PPP requires at least 15 '1' bits.

For receive operation SEROCCO-D monitors the incoming data stream for the Opening Flag (7E Hex) to identify the beginning of a HDLC packet. Subsequent bytes are part of data and are processed as normal HDLC packet including checking of CRC.

4.2.2 Octet Synchronous PPP

The SEROCCO-D transmits a data block, inserts HDLC Header (Opening Flag), and appends the HDLC Trailer (CRC, Ending Flag). Beside this standard HDLC operation, zero-bit stuffing is not performed, but character mapping is performed.

For receive operation SEROCCO-D monitors the incoming data stream for the Opening Flag (7E Hex) to identify the beginning of a HDLC packet. Subsequent bytes are part of data and are processed as normal HDLC packet including checking of CRC. Received mapped characters are unmapped.

The abort sequence consists of the control escape character $7D_H$ followed by a flag character $7E_H$ (not stuffed). Between two frames, the interframe time fill character should be programmed to $7E_H$ by setting bit CCR2H:ITF to '1'.

Octet alignment is provided through the synchronization pulses in clock mode 5b.

4.2.3 Asynchronous PPP

For transmit operation, SEROCCO-D inserts the HDLC header (Opening Flag), and appends the HDLC trailer (CRC, Ending Flag), surrounding the transmit data read from the XFIFO. Each octet (including HDLC framing flags and idle flags) is converted into async character format (1 start, 8 data bits, 1 stop bit) and then transmitted using the asynchronous character formatter block. Character mapping like in Octet Synchronous PPP mode is performed.

In receive direction any async character is transferred into SEROCCO-D's ASYNC Character De-Formatting logic block, where it is translated back into the original information octet. Mapped characters are unmapped and the information octets are then transferred to the RFIFO (as in Octet Synchronous PPP mode).

4.2.4 Data Transparency in PPP Mode

When transporting bit-files (as opposed to text files), or compressed files, the characters could easily represent MODEM control characters (such as CTRL-Q, CTRL-S) which the MODEM would not pass through. SEROCCO-D maintains an Async Control Character Map (ACCM) for characters 00-1F Hex. Whenever there is a mapped character in the data stream, the transmitter precedes that character with a control-escape character of $7D_H$. After the control-escape, the character itself is transmitted with bit 5 inverted. character e.g. 13_H is mapped to $7D_H$, 33_H).

At the receive end, a $7D_H$ character is discarded and the following character is modified by inverting bit 5 (e.g. if $7D_H$, 33_H is received, the $7D_H$ is discarded and the 33_H is changed to 13_H the original character). This character is received into RFIFO and included in CRC calculation, even if it is not mapped.

The 32 lookup octet values (00_H-$1F_H$) are stored within the on-chip registers ACCM0..3.

In addition to the ACCM, 4 user programmable characters (especially outside the range 00-1F Hex) can also be mapped using the control-escape sequence described above. These characters are specified in registers UDAC0..3.

The receiver discards all characters which are received unmapped, but expected to be mapped because of ACCM0..3 and UDAC0..3 register contents. If this occurs within an

HDLC frame, the unexpected characters are discarded before forwarded to the receive CRC checking unit.

$7D_H$ (control-escape) and $7E_H$ (flag) octets in the data stream are mapped in general. The sequence of mapping control logic is:

1. $7D_H$ and $7E_H$ octets,
2. ACCM0..3,
3. UDAC0..3.

This mechanism is applied to asynchronous HDLC PPP mode as well as to octet synchronous HDLC PPP mode.

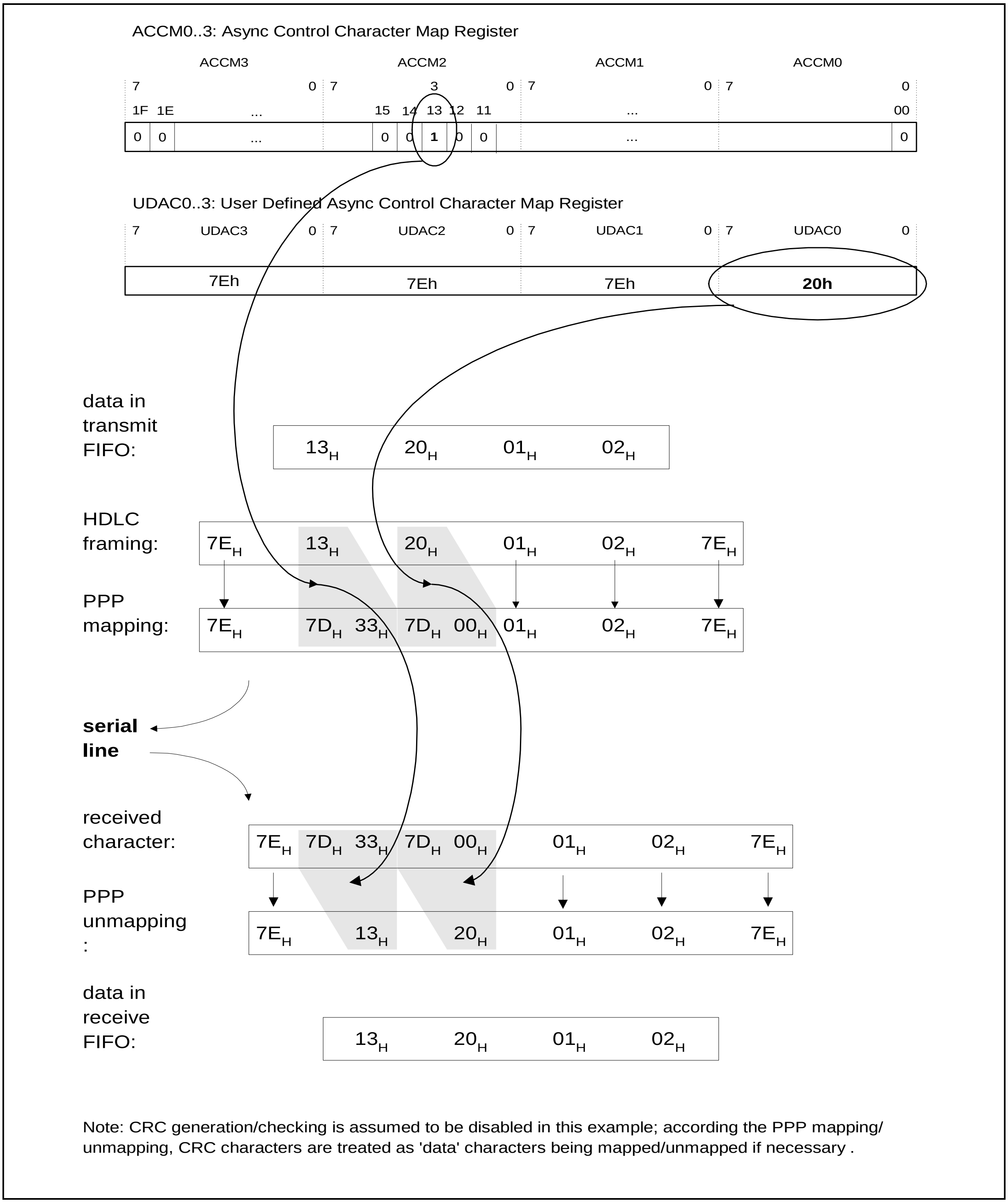


Figure 48 PPP Mapping/Unmapping Example

4.3 Extended Transparent Mode

Characteristics: fully transparent

When programmed in the extended transparent mode via the CCR2L register (bits MDS1, MDS0, ADM = ‘111’), the SCC performs fully transparent data transmission and reception without HDLC framing, i.e. without

- FLAG insertion and deletion
- CRC generation and checking
- bit stuffing.

This feature can be profitably used e.g. for:

- user specific protocol variations
- line state monitoring, or
- test purposes, in particular for monitoring or intentionally generating HDLC protocol rule violations (e.g. wrong CRC)

Character or octet boundary synchronization can be achieved by using clock mode 5 or clock mode 1 with an external receive strobe input to pin CD.

Note: Data is transmitted and received with the least significant bit (LSB) first.

4.4 Asynchronous (ASYNC) Protocol Mode

4.4.1 Character Framing

Character framing is achieved by start and stop bits. Each data character is preceded by one Start bit and terminated by one or two stop bits. The character length is selectable from 5 up to 8 bits. Optionally, a parity bit can be added which complements the number of ones to an even or odd quantity (even/odd parity). The parity bit can also be programmed to have a fixed value (Mark or Space). The character format configuration is performed via appropriate bit fields in registers CCR3L/CCR3H. **Figure 49** shows the asynchronous character format.

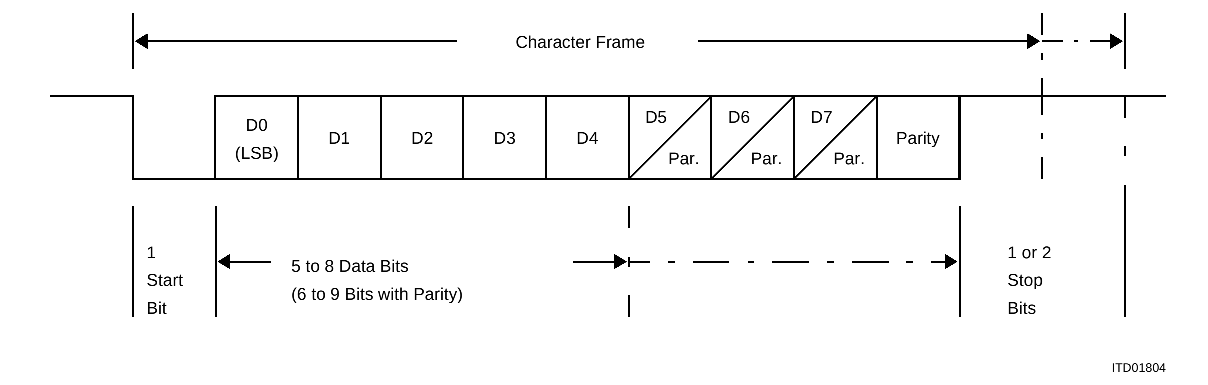


Figure 49 Asynchronous Character Frame

4.4.2 Data Reception

The SCC offers the flexibility to combine clock modes, data encoding and data sampling in many different ways. However, only definite combinations make sense and are recommended for correct operation:

4.4.2.1 Asynchronous Mode

Prerequisites:

- Bit clock rate 16 selected (register CCR0L, bit BCR = ‘1’)
- Clock mode 0, 1, 3b, 4, or 7b selected (register CCR0L, bit field ’CM’)
- NRZ data encoding selected (register CCR0H, bit field ’SC’)

The receiver which operates with a clock rate equal to 16 times the nominal (expected) data bit rate, synchronizes itself to each character by detecting and verifying the start bit. Since character length, parity and stop bit length is known, the ensuing valid bits are sampled. Oversampling (3 samples) around the nominal bit center in conjunction with majority decision is provided for every received bit (including start bit).

The synchronization lasts for one character, the next incoming character causes a new synchronization to be performed. As a result, the demand for high clock accuracy is reduced. Two communication stations using the asynchronous procedure are clocked independently, their clocks need not be in phase or locked to exactly the same frequency but, in fact, may differ from one another within a certain range.

4.4.2.2 Isochronous Mode

Prerequisites:

- Bit clock rate 1 selected (register CCR0L bit BCR = ‘0’)

- Clock mode 2, 3a, 6, or 7a (DPLL mode) has to be used in conjunction with FM0, FM1 or Manchester encoding (register CCR0L/CCR0H bit fields 'CM' and 'SC').

The isochronous mode uses the asynchronous character format. However, each data bit is only sampled once (no oversampling).

In clock modes 0 ,1 and 4, the input clock has to be externally phase locked to the data stream. This mode allows much higher transfer rates. Clock modes 3b and 7b are not recommended due to difficulties with bit synchronization when using the internal baud rate generator.

In clock modes 2, 3a, 6, and 7a, clock recovery is provided by the internal DPLL. Correct synchronization of the DPLL is achieved if there are enough edges within the data stream, which is generally ensured only if Bi-Phase encoding (FM0, FM1 or Manchester) is used.

4.4.2.3 Storage of Receive Data

If the receiver is enabled, received data is stored in the SCC receive FIFO (the LSB is received first). Moreover, the CD input may be used to control data reception. Character length, number of stop bits and the optional parity bit are checked. Storage of parity bits can be disabled. Errors are indicated via interrupts. Additionally, the character specific error status (framing and parity) can optionally be stored in the SCC receive FIFO.

Filling of the the SCC receive FIFO is controlled by

- a programmable threshold level (bit field 'RFTH' in register CCR3H),
- the selected data format (bit 'RFDF' in register CCR3H),
- the parity storage selection (bit 'DPS' in register CCR3H),
- detection of the programmable Termination Character (bit 'TCDE' in register CCR3L and bit field 'TC' in register TCR).

Additionally, the time-out event interrupt as an optional status information indicates that a certain time (refer to register TOLEN) has elapsed since the reception of the last character.

4.4.3 Data Transmission

The selection of asynchronous or isochronous operation has no further influence on the transmitter. The bit clock rate is solely a dividing factor for the selected clock source.

Transmission of the contents of the SCC transmit FIFO starts after the 'XF' command is issued (the LSB is sent out first). Further data is requested by an 'XPR' interrupt (or by DMA). The character frame for each character, consisting of start bit, the character itself with defined character length, optionally generated parity bit and stop bit(s) is assembled.

After finishing transmission (indicated by the 'ALLS' interrupt), IDLE sequence (logical '1') is transmitted on pin TxD.

Additionally, the $\overline{CTS}$ signal may be used to control data transmission.

4.4.4 Special Functions

4.4.4.1 Break Detection/Generation

Break generation:

On issuing the transmit break command (bit ’XBRK’ in register CCR3L), the TxD pin is immediately forced to physical ‘0’ level with the next following transmit clock edge, and released with the first transmit clock edge after this command is reset again by software.

Break detection:

The SCC recognizes the break condition upon receiving consecutive (physical) ‘0’s for the defined character length, the optional parity and the selected number of stop bits (‘zero’ character and framing error). The ‘zero’ character is not pushed to RFIFO. If enabled, the ’Break’ interrupt (BRK) is generated.

The break condition will be present until a ‘1’ is received which is indicated by the ‘Break Terminated’ interrupt (BRKT).

4.4.4.2 In-band Flow Control by XON/XOFF Characters

Programmable XON and XOFF characters:

The XON/XOFF registers contain the programmable values for XON and XOFF characters. The number of significant bits in a register is determined by the programmed character length via bit field ’CHL’ in register CCR3L.

Additionally, two programmable eight-bit values in registers MXON and MXOFF serve as masks for the characters XON and XOFF, respectively:

A ‘1’ in any mask bit position has the effect that no comparison is performed between the corresponding bits in the received characters (‘don’t cares’) and the XON/XOFF value. At RESET, the masks are zero’ed, i.e. all bit positions will be compared.

A received character is considered to be recognized as a valid XON or XOFF character

– if it is correctly framed (correct length),
– if its bits match the ones in the XON or XOFF registers over the programmed character length,
– if it has correct parity (if applicable).

Received XON and XOFF characters are stored in the SCC receive FIFO, as any other characters, when bit ’DXS’ is set to ’0’ in register CCR3L. Otherwise they are not stored in the receive FIFO.

In-Band Flow Control of Transmitted Characters:

Recognition of an XON or XOFF character causes always a corresponding maskable interrupt status to be generated.

Further action depends on the setting of control bit ’FLON’ (Flow Control On) in register CCR2H:

0: No further action is automatically taken by the SCC.

1: The reception of an XOFF character automatically turns off the transmitter after the currently transmitted character (if any) has been shifted out completely (entering XOFF state). The reception of an XON character automatically makes the transmitter resume transmitting (entering XON state).

After hardware RESET, bit CCR2H:FLON is ‘0’.

When bit CCR2H:FLON is programmed from ‘0’ to ‘1’, the transmitter is first in the ‘XON state’, until an XOFF character is received.
When bit CCR2H:FLON is programmed from ‘1’ to ‘0’, the transmitter always goes in the ‘XON state’, and transmission is only controlled by the user and by the $\overline{\text{CTS}}$ signal input.

The in-band flow control of the transmitter via received XON and XOFF characters can be combined with control via $\overline{\text{CTS}}$ pin, i.e. the effect of the $\overline{\text{CTS}}$ pin is independent of whether in-band control is used or not. The transmitter is enabled only if $\overline{\text{CTS}}$ is ‘low’ and XON state has been reached.

Transmitter Status Bit:

The status bit ‘Flow Control Status’ (bit ’FCS’ in register STARL) indicates the current state of the transmitter, as follows:

0: if the transmitter is in XON state,

1: if the transmitter is in XOFF state.

Note: The transmitter cannot be turned off by software without disrupting data possibly remaining in the transmit FIFO.

Flow Control for Received Data:

After writing a character value to register TICR (Transmit Immediate Character, ’TIC’) its character contents is inserted into the outgoing character stream

- immediately upon writing this register by the microprocessor if the transmitter is in IDLE state. If no further characters (transmit FIFO empty) are to be transmitted, i.e. the transmitter returns to IDLE state after transmission of the ’TIC’ and an ALLS (All Sent) interrupt will be generated.
- after the end of a character currently being transmitted if the transmitter is not in IDLE state. This does not affect the contents of the transmit FIFO. Transmission of characters from transmit FIFO is resumed after the ’TIC’ is send out.

Transmission via this register is possible even when the transmitter is in XOFF state (however, CTS must be 'low').

The 'TIC' value is an eight-bit value. The number of significant bits is determined by the programmed asynch character length via bit field 'CHL' in register CCR3L. Parity value (if programmed) and selected number of stop bits are automatically appended, equal to the characters provided via the transmit data buffer. The usage of 'TIC' is independent of in-band flow control mechanism, i.e. is not affected by bit 'FLON' in register CCR2H anyway.

To control multiple accesses to register TICR, an additional status bit STARL:TEC (TIC Executing) is provided which signals that the transmission command of currently programmed 'TIC' is accepted but not yet completely executed. Further access to register TICR is only allowed if bit STARL:TEC is '0' again.

4.4.4.3 Out-of-band Flow Control

Transmitter:

The transmitter output is enabled if CTS signal is 'LOW' AND the XON state is reached in case of in-band flow control is enabled. If the in-band flow control is disabled (CCR2H:FLON = '0'), the transmitter is only controlled by the CTS signal.

Nevertheless setting bit CCR1H**:FCTS = '1' allows the transmitter to send data independent of the condition of the CTS signal,** the in-band flow control (XON/XOFF) mechanism would still be operational if enabled via bit CCR2H:FLON = '1'.

Receiver:

For some applications it is desirable to provide means of out-of-band flow control to indicate to the far end transmitter that the local receiver's buffer is getting full.

This flow control can be used between two DTEs as shown in **Figure 50** and between a DTE and a DCE (MODEM) as shown in **Figure 51** that supports this kind of bi-directional flow control.

Setting bit CCR1H:FRTS = '1' and CCR1H:RTS = '0' invokes this out-of-band flow control for the receiver. When the shadow part of the receive FIFO has reached a set threshold of 28 bytes, the RTS signal is forced inactive (high). When the shadow part of the receive FIFO is empty, the RTS is re-asserted (low). Note that the data is immediately transferred from the shadow receive FIFO to the user accessible RFIFO (as long as there is space available). So when the shadow receive FIFO reaches the 28 bytes threshold, there is 4 more byte storage available before overflow can occur. This allows sufficient time for the far end transmitter to react to the change in the RTS signal and stop sending more data.

Figure 50 shows the connection between two SCC devices as DTEs. The RTS of DTE-A (SCC) feeds the CTS input of the second DTE-B (another SCC). For example while

DTE-A is receiving data and its receive FIFO threshold is reached, the $\overline{\text{RTS}}$ signal goes in-active 'HIGH' forcing the $\overline{\text{CTS}}$ of DTE-B to become in-active indicating that transmission has to stop after finishing the current character. Both DTE devices should also be using the $\overline{\text{CTS}}$ signal to flow control their transmitters. When the shadow receive FIFO in DTE-A is cleared its $\overline{\text{RTS}}$ goes active (low) and this signals the far end DTE-B to resume transmission. Data flow control from DTE-B to DTE-A works in the same way.

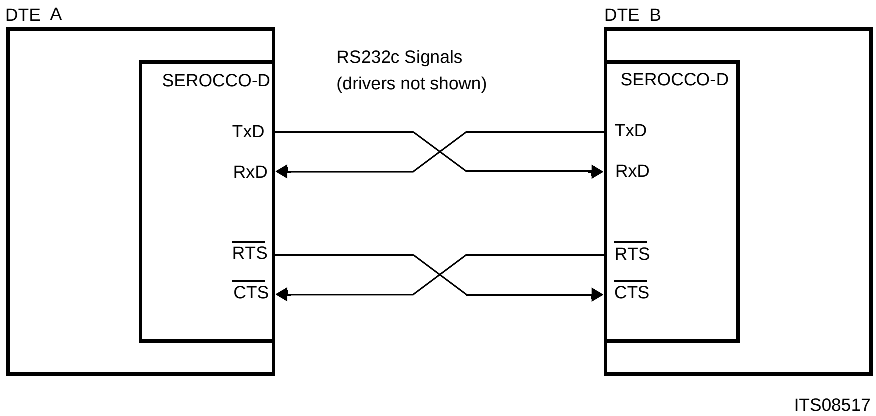


Figure 50 Out-of-Band DTE-DTE Bi-directional Flow Control

Figure 51 shows an SCC as a DTE connected to a DCE (MODEM equipment).

The $\overline{\text{RTS}}_A$ feeds the $\overline{\text{RTS}}_B$ input of the DCE (MODEM equipment) that supports bi-directional flow control. So when the DTE-A's receiver threshold is reached, the $\overline{\text{RTS}}_A$ signal goes inactive 'HIGH' which is sensed by the DCE and it stops transmitting. Similarly if the DCE's receiver threshold is reached, it deactivates the $\overline{\text{CTS}}_B$ ('HIGH') and causes the DTE to stop transmission. These types of DCEs have fairly deep buffers to ensure that it can continue to receive data from the line even though it is unable to pass the data to the DTE for short periods of time. Note that a SCC can also be used in the DCE equipment as shown. Exchange of signals (e.g. $\overline{\text{RTS}}$ to $\overline{\text{CTS}}$) is necessarily inside the DCE equipment.

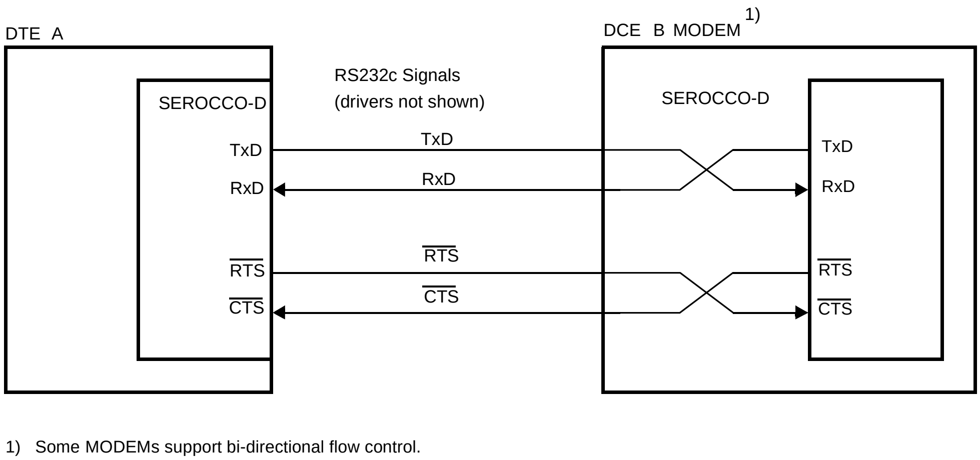


Figure 51 Out-of-Band DTE-DCE Bi-directional Flow Control

$\overline{\text{RTS}}$ and $\overline{\text{CTS}}$ are used to indicate when the local receiver's buffer is nearly full. This alerts the far end transmitter to stop transmission.

The combination of transmitter and receiver out-of-band control features mentioned above enables data to be exchanged between two devices without software intervention for flow control.

4.5 BISYNC Protocol Mode

4.5.1 Character Framing

Character oriented protocols achieve synchronization between transmitting and receiving station by means of special SYN characters. Two examples are the MONOSYNC and IBM's BISYNC procedures. BISYNC has two starting SYN characters while MONOSYNC uses only one SYN. **Figure 52** gives an example of the message format.

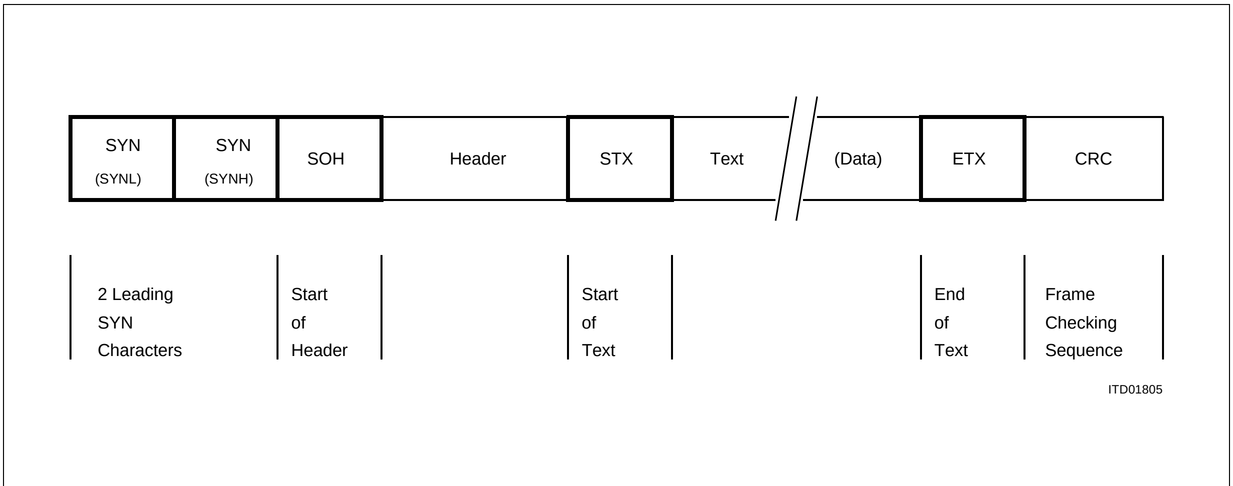


Figure 52 BISYNC Message Format

The SYN character, its length, the length of data characters and additional parity generation are programmable:

- 1 SYN character with 6 or 8 bit length (MONOSYNC), programmable via register SYNCL.
- 2 SYN characters with 6 or 8 bit length each (BISYNC), programmable via registers SYNCH/SYNCL.
- Data character length may vary from 5 to 8 bits (bit field ’CHL’ in register CCR3L).
- Parity information (even/odd parity, mark, space) may be appended to the character (bit ’PARE’ and bit field ’PAR’ in register CCR3H).

4.5.2 Data Reception

The receiver is generally activated by setting bit ’RAC’ in register CCR3L. Additionally, the CD signal may be used to control data reception depending on the selected clock mode. After issuing the HUNT command, the receiver monitors the incoming data stream for the presence of specified SYN character(s). However, data reception is still disabled. If synchronization is gained by detecting the SYN character(s), an SCD interrupt is generated and **all** following data is pushed to the receive FIFO, i.e. control sequences, data characters and optional CRC frame checking sequence (the LSB is received first). In normal operation, SYN characters are excluded from storage to receive FIFO. SYN character length can be specified independently of the selected data character length. If required, the character parity bit and/or parity status is stored together with each data byte in the receive FIFO.

As an option, the loading of SYN characters in receive FIFO may be enabled by setting the bit ’SLOAD’ in register CCR3L. Note that in this case SYN characters are treated as data. Consequently, for correct operation it must be guaranteed that SYN character length equals the character length + optional parity bit. This is the user's responsibility by appropriate software settings.

Filling of the receive FIFO is controlled by a programmable threshold level.

Reception is stopped if

1. the receiver is deactivated by resetting the bit CCR3L:RAC bit, or
2. the CD signal goes inactive (if Carrier Detect Auto Start is enabled in register CCR1H), or
3. the CMDRH:HUNT command is issued again, or
4. the Receiver Reset command (CMDRH:RRES) is issued, or
5. a programmed Termination Character has been found (optional).

On actions 1. and 2., reception remains disabled until the receiver is activated again. After this is done, and generally in cases 3. and 4., the receiver returns to the (non-synchronized) Hunt state. In case 5. a HUNT command has to be issued. Reception of data is internally disabled until synchronization is regained.

Note: Further checking of frame length, extraction of text or data information and verifying the Frame Checking Sequence (e.g. CRC) has to be done by the microprocessor.

4.5.3 Data Transmission

Transmission of data provided in the memory is started after the Transmit Frame ('XF') command is issued (the LSB is sent out first). Additionally, the $\overline{\text{CTS}}$ signal may be used to control data transmission. The message frame is assembled by appending all data characters to the specified SYN character(s) until Transmit Message End condition is detected ('XME' command in interrupt mode or, in DMA mode, when the number of characters specified in XBC1L/XBC1H have been transferred). Internally generated parity information may be added to each character (SYN, CRC and Preamble characters are excluded).

If enabled via CRC Append bit (bit 'CAPP' in register CCR2H), the internally calculated CRC checksum (16 bit) is added to the message frame. Selection between CRC-16 and CRC-CCITT algorithms is provided.

Note: - Internally generated SYN characters are always excluded from CRC calculation,
- CRC checksum (2 bytes) is sent without parity.

The internal CRC generator is automatically initialized before transmission of a new frame starts. The initialization value is selectable.

After finishing data transmission, interframe-time-fill (SYN characters or IDLE) is automatically sent.

A transmit data underrun condition in the XFIFO is indicated with an 'XDU' interrupt. Nevertheless, transmission continues inserting SYN characters into the data stream until new data is available in the transmit FIFO. Inserted SYN characters are not part of the frame and thus not used for CRC calculation.

4.5.4 Special Functions

4.5.4.1 Preamble Transmission

If enabled via register CCR2H, a programmable 8-bit pattern (register PREAMB) is transmitted with a selectable number of repetitions after interframe-time-fill transmission is stopped and a new frame is ready to be sent out.

Note: If the preamble pattern equals the SYN pattern, reception is triggered by the preamble.

4.6 Procedural Support (Layer-2 Functions)

When operating in the auto mode, the SCC offers a high degree of protocol support. In addition to address recognition, the SCC autonomously processes all (numbered) S- and I-frames (window size 1 only) with either normal or extended control field format (modulo-8 or modulo-128 sequence numbers – selectable via register CCR2H bit 'MCS').

The following functions will be performed:

– updating of transmit and receive counter
– evaluation of transmit and receive counter
– processing of S commands
– flow control with RR/RNR
– generation of responses
– recognition of protocol errors
– transmission of S commands, if acknowledgement is not received
– continuous status query of remote station after RNR has been received
– programmable timer/repeater functions.

In addition, all unnumbered frames are forwarded directly to the processor. The logical link can be initialized by software at any time (Reset HDLC Receiver by RRES command in register CMDRH).
Additional logical connections can be operated in parallel by software.

4.6.1 Full-Duplex LAPB/LAPD Operation

Initially (i.e. after RESET), the LAP controllers of the two serial channels are configured to function as a combined (primary/secondary) station, where they autonomously perform a subset of the balanced X.25 LAPB/ISDN LAPD protocol.

Reception of Frames:

The logical processing of received S-frames is performed by the SCC without interrupting the host. The host is merely informed by interrupt of status changes in the remote station (receiver ready / receiver not ready) and protocol errors (unacceptable N(R), or S-frame with I-field).

I-frames are also processed autonomously and checked for protocol errors. The I-frame will not be accepted in the case of sequence errors (no interrupt is forwarded to the host), but is immediately confirmed by an S-response. If the host sets the SCC into a 'receive not ready' status, an I-frame will not be accepted (no interrupt) and an RNR response is transmitted. U-frames are always stored in the RFIFO and forwarded directly to the host. The logical sequence and the reception of a frame in auto mode is illustrated in **Figure 53**.

Note: The state variables N(S), N(R) are evaluated within the window size 1, i.e. the SCC checks only the least significant bit of the receive and transmit counter regardless of the selected modulo count.

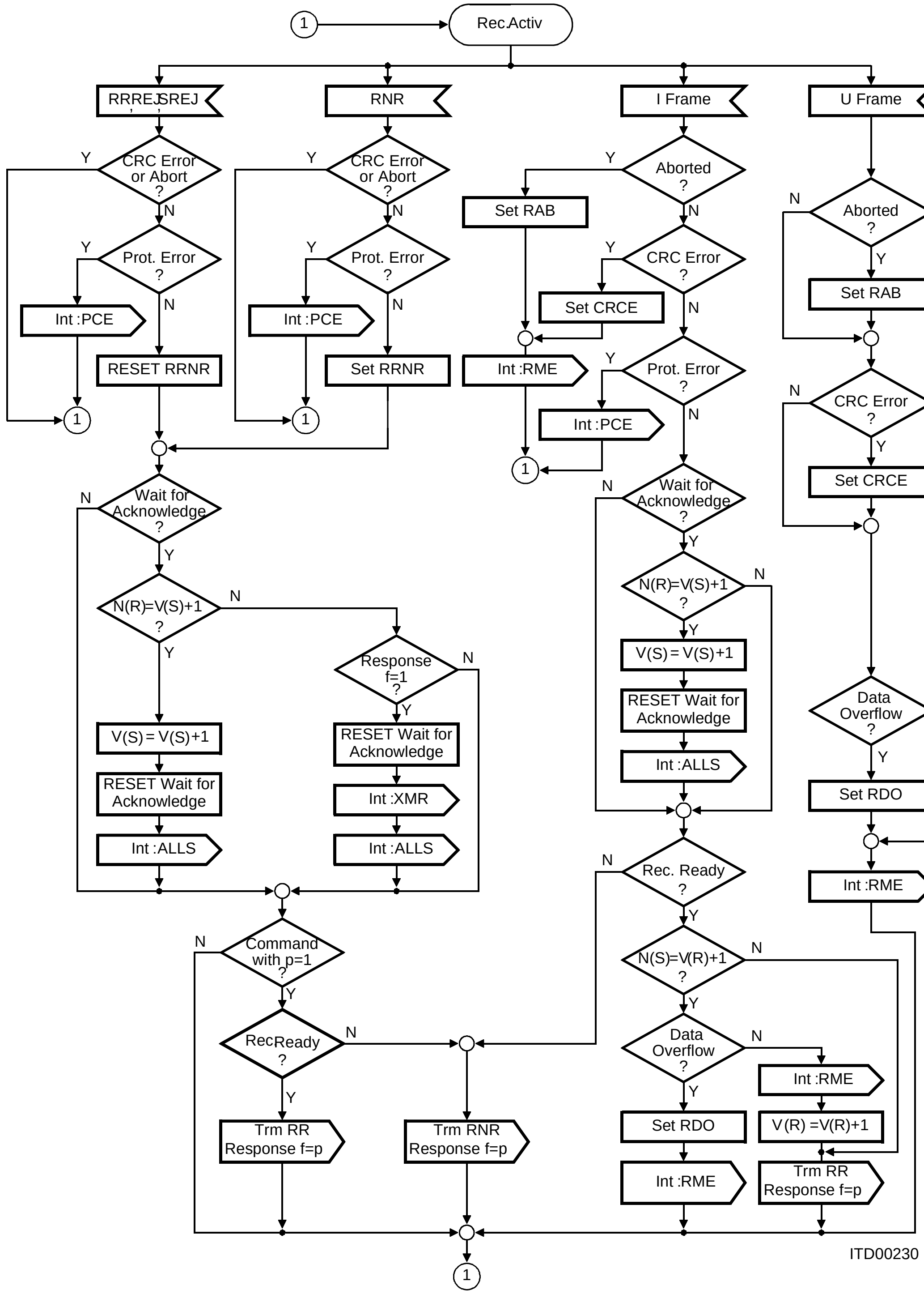


Figure 53 Processing of Received Frames in Auto Mode

Transmission of Frames:

The SCC autonomously transmits S commands and S responses in the auto mode. Either transparent or I-frames can be transmitted by the user. The software timer has to be operated in the internal timer mode to transmit I-frames. After the frame has been transmitted, the timer is self-started, the XFIFO is inhibited, and the SCC waits for the arrival of a positive acknowledgement. This acknowledgement can be provided by means of an S- or I-frame.

If no positive acknowledgement is received during time t_1, the SCC transmits an S-command (p = '1'), which must be answered by an S-response (f = '1'). If the S-response is not received, the process is performed n1 times (in HDLC known as N2, refer to register TIMR3).

Upon the arrival of an acknowledgement or after the completion of this poll procedure the XFIFO is enabled and an interrupt is generated. Interrupts may be triggered by the following:

- message has been positively acknowledged (ALLS interrupt)
- message must be repeated (XMR interrupt)
- response has not been received (TIN interrupt).

In automode, only when the ALLS interrupt has been issued data of a new frame may be provided to the XFIFO!

Upon arrival of an RNR frame, the software timer is started and the status of the remote station is polled periodically after expiration of t_1, until the status 'receive ready' has been detected. The user is informed via the appropriate interrupt. If no response is received after n1 times, a TIN interrupt, and t_1 clock periods thereafter an ALLS interrupt is generated and the process is terminated.

Note: The internal timer mode should only be used in the auto mode.

Transparent frames can be transmitted in all operating modes.

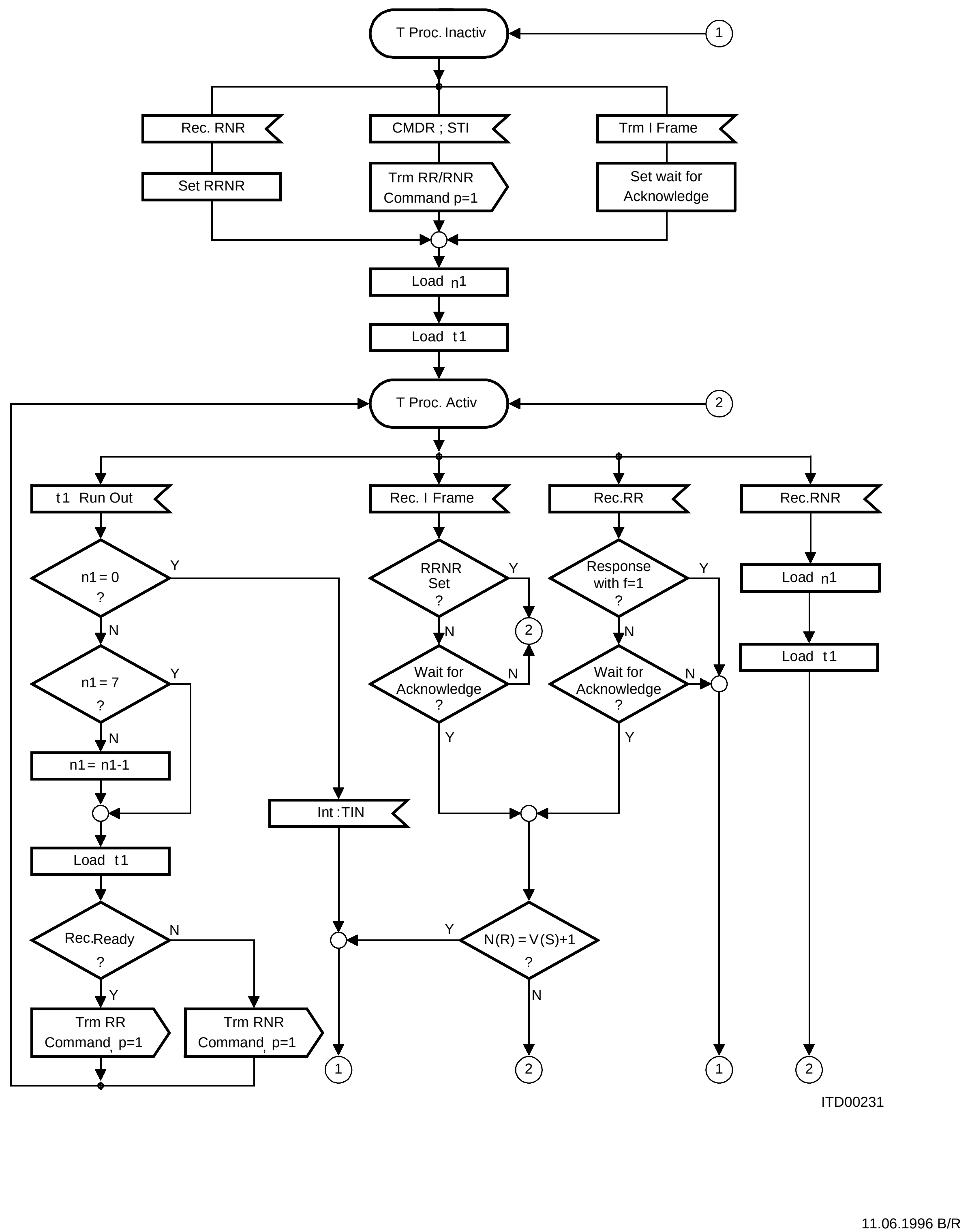


Figure 54 Timer Procedure/Poll Cycle

Examples

The interaction between SCC and the host during transmission and reception of I-frames is illustrated in the following two figures. The flow control with RR/RNR of I-frames during transmission/reception is illustrated in **Figure 55**. Both, the sequence of the poll cycle and protocol errors are shown in **Figure 56**.

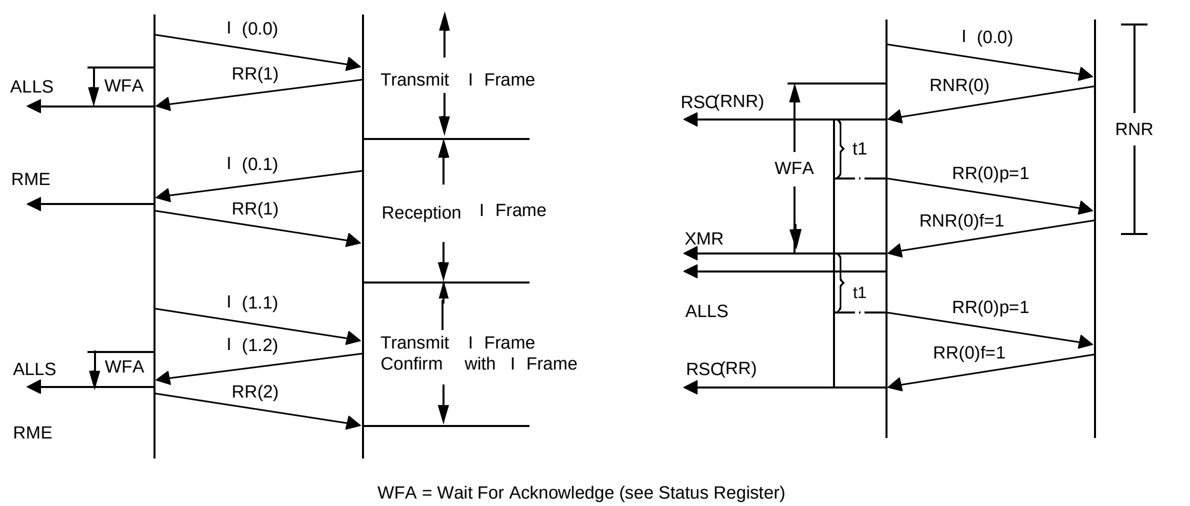


Figure 55 Transmission/Reception of I-Frames and Flow Control

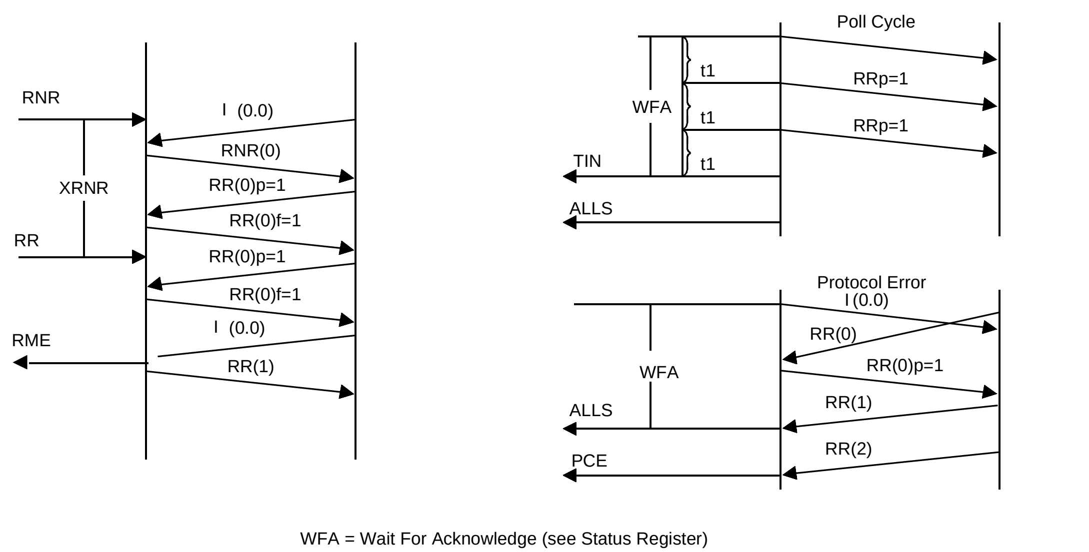


Figure 56 Flow Control: Reception of S-Commands and Protocol Errors

Protocol Error Handling:

Depending on the error type, erroneous frames are handled according to **Table 14**.

Table 14 Error Handling

Frame Type	Error Type	Generated Response	Generated Interrupt	Rec. Status
I	CRC error Aborted Unexpected N(S) Unexpected N(R)	– – S-frame –	RME RME – PCE	CRC error Abort – –
S	CRC error Aborted Unexpected N(R) With I-field	– – – –	– – PCE PCE	– – – –

Note: The station variables (V(S), V(R)) are not changed.

4.6.2 Half-Duplex SDLC-NRM Operation

The LAP controllers of the two serial channels can be configured to function in a half-duplex Normal Response Mode (NRM), where they operate as a slave (secondary) station, by setting the NRM bit in the CCR2L register of the corresponding channel.

In contrast to the full-duplex LAP B/LAP D operation, where the combined (primary + secondary) station transmits both commands and responses and may transmit data at any time, the NRM mode allows only responses to be transmitted **and** the secondary station may transmit only when instructed to do so by the master (primary) station. The SCC gets the permission to transmit from the primary station via an S-, or I-frame with the poll bit (p) **set**.

The NRM mode can be profitably used in a point-to-multipoint configuration with a fixed master-slave relationship, which guarantees the absence of collisions on the common transmit line. It is the responsibility of the master station to poll the slaves periodically and to handle error situations.

Prerequisite for NRM operation is:

– auto mode with 8-bit address field selected
  Register CCR2L bit fields ’MDS1’, ’MDS0’, ’ADM’ = ‘000’
– Register TIMR3 bit ’TMD’ = ‘0’
– same transmit and receive addresses, since only responses can be transmitted, i.e.
  Register XAD1 = XAD2 and register RAL1 = RAL2 (address of secondary).

Note: The broadcast address may be programmed in register RAL2 if broadcasting is required.
In this case registers RAL1 and RAL2 are not equal.

The primary station has to operate in transparent HDLC mode.

Reception of Frames:

The reception of frames functions similarly to the LAPB/LAPD operation (see **"Full-Duplex LAPB/LAPD Operation" on Page 109**).

Transmission of Frames:

The SCC does **not** transmit S-, or I-frames if not instructed to do so by the primary station via an S-, or I-frame with the poll bit set.

The SCC can be told to send an I-frame issuing the transmit command 'XIF' in register CMDRL. The transmission of the frame, however, will not be initiated by the SCC until reception of either an

- RR, or
- I-frame

with poll bit set (p = '1').

After the frame has been transmitted (with the final bit set), the host has to wait for an ALLS or XMR interrupt.

A secondary does not poll the primary for acknowledgements, thus timer supervision must be done by the primary station.

Upon the arrival of an acknowledgement the SCC transmit FIFO is enabled and an interrupt is forwarded to the host, either the

– message has been positively acknowledged (ALLS interrupt), or the
– message must be repeated (XMR interrupt).

Additionally, the on-chip timer can be used **under host control** to provide timer recovery of the secondary if no acknowledgements are received at all.

Note: A secondary will transmit transparent frames only if the permission to send is given by receiving an S-frame or I-frame with poll bit set (p = '1').

Examples:

A few examples of SCC/host interaction in the case of normal response mode (NRM) mode are shown in **Figure 57** and **Figure 58**.

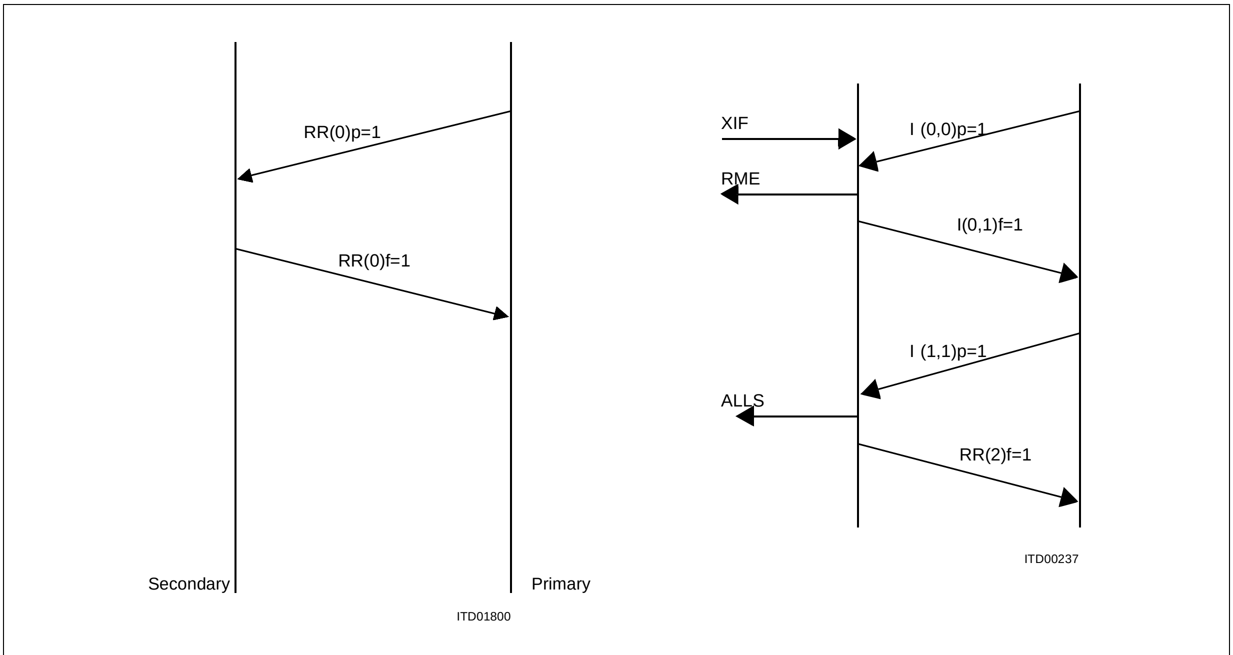


Figure 57 No Data to Send: Data Reception/Transmission

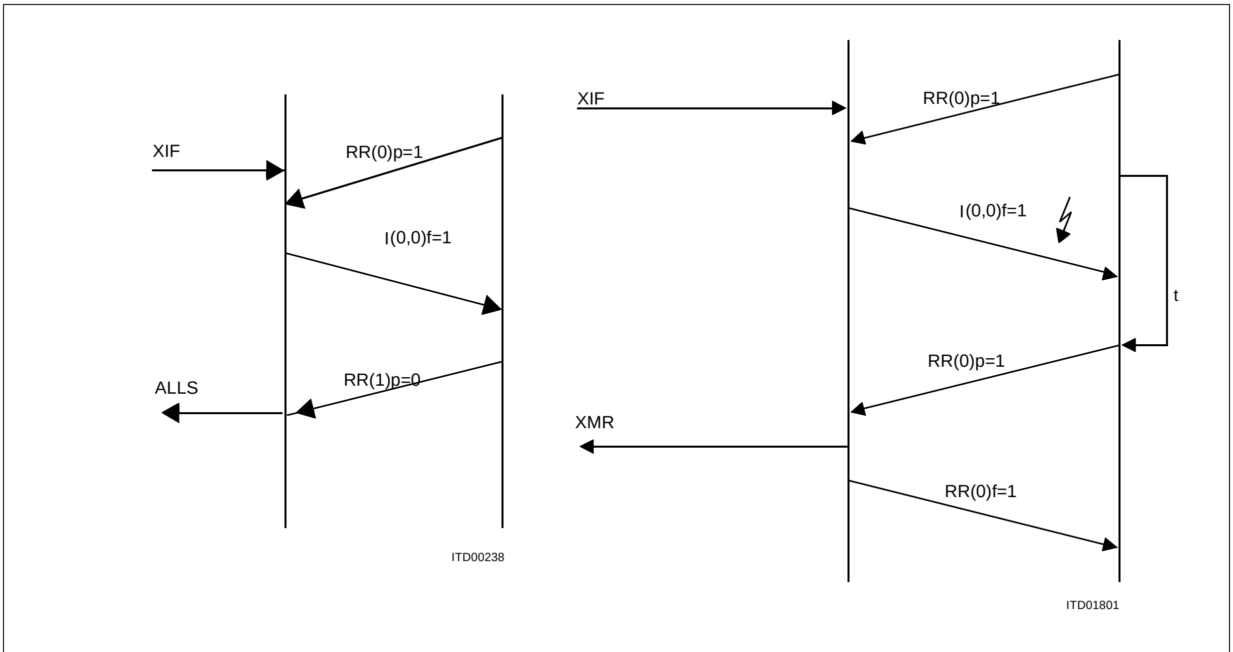


Figure 58 Data Transmission (without error), Data Transmission (with error)

4.6.3 Signaling System #7 (SS7) Operation

The SEROCCO-D supports the signaling system #7 (SS7) which is described in ITU-Q.703. SS7 support must be activated by setting bit ’ESS7’ in register CCR3L.

Receive

The SS7 protocol is supported by the following hardware features in receive direction:

- Recognition of Signaling Unit type
- Discard of repeatedly received FISUs and optionally of LSSUs if content is unchanged
- Check if the length of the received signaling unit is at least six octets (including the opening flag)
- Check if the signal information field of a received signaling unit consists of more than 272 octets (enabled with bit CCR3L.ELC). In this case, reception of the current signaling unit will be aborted.
- Counting and processing of errored signaling units

In order to reduce the microprocessor load, Fill In Signaling Units (FISUs) are processed automatically. By examining the length indicator of a received Signal Unit (SU) SEROCCO-D decides whether a FISU has been received. Consecutively received FISUs will be compared and not stored in the RFIFO, if the content is equal to the previous one. The same applies to Link Status Signaling Units (LSSUs), if enabled with bit CCR3L.CSF. The different types of Signaling Units as Message Signaling Unit (MSU), Link Status Signaling Unit (LSSU) and Fill-In Signaling Units (FISU) are indicated in the RSTA byte (bit field ’SU’), which is automatically added to the RFIFO with each received Signaling Unit. The complete Signaling Unit except start and end flags is stored in the receive FIFO. The functions of bits CCR3H.RCRC and CCR3H.RADD are also valid in SS7 mode, with bit ’RADD’ related to BSN (backward sequence number) and FSN (forward sequence number).

Errored signaling units are counted and processed according to ITU-T Q.703. The SU counter and errored-SU counter are reset by setting CMDRH.RSUC to ’1’. The error threshold can be selected to be 64 (default) or 32 by clearing/setting bit CCR3L.SUET. If the defined error limit is exceeded, an interrupt (ISR1.SUEX) is generated, if not masked by bit IMR1.SUEX.

Transmit

In transmit direction, following features are supported:

- single or repetitive transmission of signaling units
- automatic generation of Fill-In Signaling Units (FISU)

Each Signaling Unit (SU) written to the transmit FIFO (XFIFO) will be sent once or repeatedly including flags, CRC checksum and stuffed bits. After e.g. an MSU has been transmitted completely, SEROCCO-D optionally starts sending of Fill In Signaling Units (FISUs) containing the forward sequence number (FSN) and the backward sequence number (BSN) of the previously transmitted signaling unit. Setting bit CCR3L.AFX to ’1’ causes FISUs to be sent continuously if no Signaling Unit is to be transmitted from XFIFO. After a new signaling unit has been written to the XFIFO and a transmission has been initiated, the current FISU is completed and the new SU is sent. After this, transmission of FISUs continues. The internally generated FISUs contain FSN and BSN of the last transmitted signaling unit written to XFIFO.

Using CMDRL.XREP='1', the contents of XFIFO (1..32 bytes) can be sent continuously. This cyclic transmission can be stopped with the CMDRL.XRES command.

5 Register Description

5.1 Register Overview

The SEROCCO-D global registers are used to configure and control the Serial Communication Controllers (SCCs), General Purpose Pins (GPP) and DMA operation.

All registers are 8-bit organized registers, but grouped and optimized for 16 bit access. 16 bit access is supported to even addresses only.

Table 15 provides an overview about all on-chip registers:

Table 15 Register Overview

<table>
<tr><th colspan="2">Offset Ch</th><th colspan="2">Register</th><th rowspan="2">Res Val</th><th rowspan="2">Meaning</th><th rowspan="2">Page</th></tr>
<tr><th>A</th><th>B</th><th>read</th><th>write</th></tr>
<tr><td colspan="6">Global registers:</td><td></td></tr>
<tr><td colspan="2">00_H</td><td colspan="2">GCMDR</td><td>00_H</td><td>Global Command Register</td><td>126</td></tr>
<tr><td colspan="2">01_H</td><td colspan="2">GMODE</td><td>$0B_H$</td><td>Global Mode Register</td><td>127</td></tr>
<tr><td colspan="2">02_H</td><td colspan="2">DBSR</td><td>00_H</td><td>DMA Buffer Status Register</td><td>130</td></tr>
<tr><td colspan="2">03_H</td><td>GSTAR</td><td></td><td>00_H</td><td>Global Status Register</td><td>131</td></tr>
<tr><td colspan="2">04_H</td><td colspan="2">Reserved</td><td></td><td></td><td></td></tr>
<tr><td colspan="2">05_H</td><td colspan="2">GPDIR</td><td>FF_H</td><td>GPP Direction Register</td><td>133</td></tr>
<tr><td colspan="2">06_H</td><td colspan="2">Reserved</td><td></td><td></td><td></td></tr>
<tr><td colspan="2">07_H</td><td colspan="2">GPDAT</td><td>-</td><td>GPP Data Register</td><td>134</td></tr>
<tr><td colspan="2">08_H</td><td colspan="2">Reserved</td><td></td><td></td><td></td></tr>
<tr><td colspan="2">09_H</td><td colspan="2">GPIM</td><td>FF_H</td><td>GPP Interrupt Mask Register</td><td>135</td></tr>
<tr><td colspan="2">$0A_H$</td><td colspan="2">Reserved</td><td></td><td></td><td></td></tr>
<tr><td colspan="2">$0B_H$</td><td>GPIS</td><td></td><td>00_H</td><td>GPP Interrupt Status Register</td><td>136</td></tr>
<tr><td colspan="2">$0C_H$</td><td colspan="2">DCMDR</td><td>00_H</td><td>DMA Command Register</td><td>137</td></tr>
<tr><td colspan="2">$0D_H$</td><td colspan="2">DMODE</td><td>00_H</td><td>DMA Mode Register</td><td>139</td></tr>
<tr><td colspan="2">$0E_H$</td><td>DISR</td><td></td><td>00_H</td><td>DMA Interrupt Status Register</td><td>140</td></tr>
<tr><td colspan="2">$0F_H$</td><td colspan="2">DIMR</td><td>77_H</td><td>DMA Interrupt Mask Register</td><td>142</td></tr>
<tr><td colspan="6">Channel specific registers:</td><td></td></tr>
<tr><td>10_H</td><td>60_H</td><td rowspan="2">RFIFO</td><td rowspan="2">XFIFO</td><td>-</td><td>Receive/Transmit FIFO (Low Byte)</td><td>143</td></tr>
<tr><td>11_H</td><td>61_H</td><td>-</td><td>Receive/Transmit FIFO (High Byte)</td><td>143</td></tr>
</table>

Table 15 Register Overview (cont'd)

Offset Ch A	Offset Ch B	Register read	Register write	Res Val	Meaning	Page
12_H	62_H	STARL		00_H	Status Register (Low Byte)	145
13_H	63_H	STARH		10_H	Status Register (High Byte)	145
14_H	64_H	CMDRL		00_H	Command Register (Low Byte)	150
15_H	65_H	CMDRH		00_H	Command Register (High Byte)	150
16_H	66_H	CCR0L		00_H	Channel Configuration Register 0 (Low Byte)	155
17_H	67_H	CCR0H		00_H	Channel Configuration Register 0 (High Byte)	155
18_H	68_H	CCR1L		00_H	Channel Configuration Register 1 (Low Byte)	159
19_H	69_H	CCR1H		00_H	Channel Configuration Register 1 (High Byte)	159
$1A_H$	$6A_H$	CCR2L		00_H	Channel Configuration Register 2 (Low Byte)	164
$1B_H$	$6B_H$	CCR2H		00_H	Channel Configuration Register 2 (High Byte)	164
$1C_H$	$6C_H$	CCR3L		00_H	Channel Configuration Register 3 (Low Byte)	171
$1D_H$	$6D_H$	CCR3H		00_H	Channel Configuration Register 3 (High Byte)	171
$1E_H$	$6E_H$	PREAMB		00_H	Preamble Register	179
$1F_H$	$6F_H$	TOLEN		00_H	Time Out Length Register	180
20_H	70_H	ACCM0		00_H	PPP ASYNC Control Character Map 0	181
21_H	71_H	ACCM1		00_H	PPP ASYNC Control Character Map 1	181
22_H	72_H	ACCM2		00_H	PPP ASYNC Control Character Map2	182
23_H	73_H	ACCM3		00_H	PPP ASYNC Control Character Map 3	182
24_H	74_H	UDAC0		$7E_H$	User Defined PPP ASYNC Control Character Map 0	184
25_H	75_H	UDAC1		$7E_H$	User Defined PPP ASYNC Control Character Map 1	184
26_H	76_H	UDAC2		$7E_H$	User Defined PPP ASYNC Control Character Map 2	185

Table 15 Register Overview (cont'd)

Offset Ch A	Offset Ch B	Register read	Register write	Res Val	Meaning	Page
27_H	77_H	UDAC3		$7E_H$	User Defined PPP ASYNC Control Character Map 3	185
28_H	78_H	TTSA0		00_H	Transmit Time Slot Assignment Register 0	187
29_H	79_H	TTSA1		00_H	Transmit Time Slot Assignment Register 1	187
$2A_H$	$7A_H$	TTSA2		00_H	Transmit Time Slot Assignment Register 2	188
$2B_H$	$7B_H$	TTSA3		00_H	Transmit Time Slot Assignment Register 3	188
$2C_H$	$7C_H$	RTSA0		00_H	Receive Time Slot Assignment Register 0	190
$2D_H$	$7D_H$	RTSA1		00_H	Receive Time Slot Assignment Register 1	190
$2E_H$	$7E_H$	RTSA2		00_H	Receive Time Slot Assignment Register 2	191
$2F_H$	$7F_H$	RTSA3		00_H	Receive Time Slot Assignment Register 3	191
30_H	80_H	PCMTX0		00_H	PCM Mask Transmit Direction Register 0	193
31_H	81_H	PCMTX1		00_H	PCM Mask Transmit Direction Register 1	193
32_H	82_H	PCMTX2		00_H	PCM Mask Transmit Direction Register 2	194
33_H	83_H	PCMTX3		00_H	PCM Mask Transmit Direction Register 3	194
34_H	84_H	PCMRX0		00_H	PCM Mask Receive Direction Register 0	196
35_H	85_H	PCMRX1		00_H	PCM Mask Receive Direction Register 1	196
36_H	86_H	PCMRX2		00_H	PCM Mask Receive Direction Register 2	197
37_H	87_H	PCMRX3		00_H	PCM Mask Receive Direction Register 3	197
38_H	88_H	BRRL		00_H	Baud Rate Register (Low Byte)	199
39_H	89_H	BRRH		00_H	Baud Rate Register (High Byte)	199
$3A_H$	$8A_H$	TIMR0		00_H	Timer Register 0	201
$3B_H$	$8B_H$	TIMR1		00_H	Timer Register 1	201
$3C_H$	$8C_H$	TIMR2		00_H	Timer Register 2	202
$3D_H$	$8D_H$	TIMR3		00_H	Timer Register 3	202
$3E_H$	$8E_H$	XAD1		00_H	Transmit Address 1 Register	205
$3F_H$	$8F_H$	XAD2		00_H	Transmit Address 2 Register	205
40_H	90_H	RAL1		00_H	Receive Address 1 Low Register	207
41_H	91_H	RAH1		00_H	Receive Address 1 High Register	207
42_H	92_H	RAL2		00_H	Receive Address 2 Low Register	208
43_H	93_H	RAH2		00_H	Receive Address 2 High Register	208

Table 15 Register Overview (cont'd)

Offset Ch A	Offset Ch B	Register read	Register write	Res Val	Meaning	Page
44_H	94_H	AMRAL1		00_H	Mask Receive Address 1 Low Register	210
45_H	95_H	AMRAH1		00_H	Mask Receive Address 1 High Register	210
46_H	95_H	AMRAL2		00_H	Mask Receive Address 2 Low Register	211
47_H	96_H	AMRAH2		00_H	Mask Receive Address 2 High Register	211
48_H	98_H	RLCRL		00_H	Receive Length Check Register (Low Byte)	213
49_H	99_H	RLCRH		00_H	Receive Length Check Register (High Byte)	213
$4A_H$	$9A_H$	XON		00_H	XON In-Band Flow Control Character Register	215
$4B_H$	$9B_H$	XOFF		00_H	XOFF In-Band Flow Control Character Register	215
$4C_H$	$9C_H$	MXON		00_H	XON In-Band Flow Control Mask Register	217
$4D_H$	$9D_H$	MXOFF		00_H	XOFF In-Band Flow Control Mask Register	217
$4E_H$	$9E_H$	TCR		00_H	Termination Character Register	219
$4F_H$	$9F_H$	TICR		00_H	Transmit Immediate Character Register	220
50_H	$A0_H$	ISR0		00_H	Interrupt Status Register 0	222
51_H	$A1_H$	ISR1		00_H	Interrupt Status Register 1	222
52_H	$A2_H$	ISR2		00_H	Interrupt Status Register 2	223
53_H	$A3_H$	*Reserved*				
54_H	$A4_H$	IMR0		FF_H	Interrupt Mask Register 0	230
55_H	$A5_H$	IMR1		FF_H	Interrupt Mask Register 1	230
56_H	$A6_H$	IMR2		03_H	Interrupt Mask Register 2	231
57_H	$A7_H$	*Reserved*				
58_H	$A8_H$	RSTA		00_H	Receive Status Byte	233
59_H	$A9_H$	*Reserved*				
$5A_H$	AA_H	SYNCL		00_H	SYN Character Register (Low Byte)	237
$5B_H$	AB_H	SYNCH		00_H	SYN Character Register (High Byte)	237

Table 15 Register Overview (cont'd)

Offset Ch A	Offset Ch B	Register read	Register write	Res Val	Meaning	Page
$5C_H$ ... $5F_H$	AC_H ... AF_H	*Reserved*				
Channel specific DMA registers:						
$B0_H$	CA_H	TBADDR1L		00_H	Primary Transmit Base Address (Low Byte)	239
$B1_H$	CB_H	TBADDR1M		00_H	Primary Transmit Base Address (Mid Byte)	239
$B2_H$	CC_H	TBADDR1H		00_H	Primary Transmit Base Address (High Byte)	240
$B3_H$	CD_H	Reserved				
$B4_H$	CE_H	TBADDR2L		00_H	Secondary Transmit Base Address (Low Byte)	241
$B5_H$	CF_H	TBADDR2M		00_H	Secondary Transmit Base Address (Mid Byte)	241
$B6_H$	$D0_H$	TBADDR2H		00_H	Secondary Transmit Base Address (High Byte)	242
$B7_H$	$D1_H$	Reserved				
$B8_H$	$D2_H$	XBC1L		00_H	Primary Transmit Byte Count (Low Byte)	243
$B9_H$	$D3_H$	XBC1H		00_H	Primary Transmit Byte Count (High Byte)	243
BA_H	$D4_H$	XBC2L		00_H	Secondary Transmit Byte Count (Low Byte)	245
BB_H	$D5_H$	XBC2H		00_H	Secondary Transmit Byte Count (High Byte)	245
BC_H	$D6_H$	RBADDR1L		00_H	Primary Receive Base Address (Low Byte)	247
BD_H	$D7_H$	RBADDR1M		00_H	Primary Receive Base Address 1 (Mid Byte)	247
BE_H	$D8_H$	RBADDR1H		00_H	Primary Receive Base Address 1 (High Byte)	248
BF_H	$D9_H$	Reserved				

Table 15 Register Overview (cont'd)

Offset Ch A	Offset Ch B	Register read	Register write	Res Val	Meaning	Page
$C0_H$	DA_H	RBADDR2L		00_H	Secondary Receive Base Address (Low Byte)	249
$C1_H$	DB_H	RBADDR2M		00_H	Secondary Receive Base Address (Mid Byte)	249
$C2_H$	DC_H	RBADDR2H		00_H	Secondary Receive Base Address2 (High Byte)	250
$C3_H$	DD_H	Reserved				
$C4_H$	DE_H	RMBSL		00_H	Receive Maximum Buffer Size (Low Byte)	251
$C5_H$	DF_H	RMBSH		00_H	Receive Maximum Buffer Size (High Byte)	251
$C6_H$	$E0_H$	RBCL		00_H	Receive Byte Count (Low Byte)	253
$C7_H$	$E1_H$	RBCH		00_H	Receive Byte Count (High Byte)	253
$C8_H$	$E2_H$	*Reserved*				
$C9_H$	$E3_H$	*Reserved*				
Miscellaneous:						
$E4_H$ ... EB_H		*Reserved*				
EC_H		VER0		03_H	Version Register 0	255
ED_H		VER1		$E0_H$	Version Register 1	255
EE_H		VER2		05_H	Version Register 2	256
EF_H		VER3		20_H	Version Register 3	256

5.2 Detailed Register Description

5.2.1 Global Registers

Each register description is organized in three parts:

- a head with general information about reset value, access type (read/write), offset address and usual handling;
- a table containing the bit information (name of bit positions);
- a section containing the detailed description of each bit.

Register 1 **GCMDR**
Global Command Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

Offset Address: **00_H**

typical usage: written by CPU,
evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	Global Command Bits							
	0	0	0	0	0	0	0	SWR

SWR **Software Reset Command**

Self clearing command bit:

bit='0' No software reset command is issued.

bit='1' Causes SEROCCO-D to perform a complete reset identical to hardware reset.

Register 2 **GMODE**
Global Mode Register

CPU Accessibility: **read/write**

Reset Value: **$0B_H$**

Offset Address: **01_H**

typical usage: written by CPU
evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	DMA and Global Control							
	IDMA	0	IPC(1:0)		OSCPD	BRC	DSHP	GIM

IDMA **Enable Internal DMA**

This bit field controls the DMA operation mode:

IDMA='0' The internal DMA controller functions are disabled. SEROCCO-D is operated in standard register access controlled mode.

IDMA='1' The internal DMA controller is enabled. Single Buffer or Switched Buffer operation mode is selected with register DMODE.

IPC(1:0) **Interrupt Pin Characteristic**

These bits control the characteristic of interrupt output pin INT/$\overline{\text{INT}}$:

IPC(1:0)	Output Function:
'00'	Open Drain active low
'01'	Push/Pull active low
'10'	*Reserved*.
'11'	Push/Pull active high

OSCPD **Oscillator Power Down**

Setting this bit to ’0’ enables the internal oscillator. For power saving purposes (escpecially if clock modes are used which do not need the internal oscillator) this bit may remain set to ’1’.

OSCPD=’0’ The internal oscillator is active.

OSCPD=’1’ The internal oscillator is in power down mode.

Note: After reset this bit is set to ’1’, i.e. the oscillator is in power down mode!

BRC **Bus Request Pin Characteristic**

This bit controls the characteristic of output pin $\overline{\text{BREQ}}$:

BRC=’0’ Open Drain active low (reset value)

BRC=’1’ Push/Pull active low

Note: If bus preemption as shown in ***Chapter 3.4.3*** *is not needed, enabling the push/pull output characteristic makes a strong pull-up resistor for pin* $\overline{\text{BREQ}}$ *obsolete.*

DSHP **Disable Shaper**

This bit has to be set to ’0’ if the shaping function in the oscillator unit is desired. The shaper amplifies the oscillator signal and improves the slope of the clock edges.

DSHP=’0’ Shaper is enabled. Recommended setting if a crystal is connected to pins XTAL1/XTAL2.

DSHP=’1’ Shaper is disabled (bypassed). Recommended setting if
- a TTL level clock signal is supplied to pin XTAL1
- the oscillator unit is unused

Note: After reset this bit is set to ’1’, i.e. the shaper is disabled!

GIM **Global Interrupt Mask**

This bits disables all interrupt indications via pin INT/$\overline{\text{INT}}$. Internal operation (interrupt generation, interrupt status register update,...) is not affected.
If set, pin INT/$\overline{\text{INT}}$ immediately changes or remains in inactive state.

GIM='0' Global interrupt mask is cleared. Pin INT/$\overline{\text{INT}}$ is controlled by the internal interrupt control logic and activated as long as at least one unmasked interrupt indication is pending (not yet confirmed by read access to corresponding interrupt status register).

GIM='1' Global interrupt mask is set. Pin INT/$\overline{\text{INT}}$ remains inactive.

Note: After reset this bit is set to '1', i.e. all interrupts are disabled!

Register 3 **DBSR**
DMA Buffer Status Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

Offset Address: **02_H**

typical usage: written by SEROCCO-D evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	Internal DMAC Status Information							
	DTBB	0	DRBB	0	DTBA	0	DRBA	0

DTBB **DMA Transmit Buffer Channel B**

DRBB **DMA Receive Buffer Channel B**

DTBA **DMA Transmit Buffer Channel A**

DRBA **DMA Receive Buffer Channel A**

Only valid in internal DMA controller modes.
These bits indicate on which buffer the corresponding DMA channel is currently operating on.
These status bits are for debug purposes only.

bit = '0' base address 1 is active address

bit = '1' base address 2 is active address

Register 4 GSTAR
Global Status Register

CPU Accessibility: **read only**

Reset Value: **00_H**

Offset Address: **03_H**

typical usage: written by SEROCCO-D evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	Global Interrupt Status Information							
	GPI	DMI	ISA2	ISA1	ISA0	ISB2	ISB1	ISB0

GPI **General Purpose Port Indication** (-)

This bit indicates, that a GPP port interrupt indication is pending:

GPI='0' No general purpose port interrupt indication is pending.

GPI='1' General purpose port interrupt indication is pending. The source for this interrupt can be further determined by reading register GPIS (refer to page 5-136).

DMI **DMA Interrupt Indication** (-)

This bit indicates, that a DMA interrupt indication is pending:

DMI='0' No DMA interrupt indication is pending.

DMI='1' DMA interrupt indication is pending. The source for this interrupt (channel A/B, receive/transmit) can be further determined by reading register DISR (refer to page 5-140).

ISA2 **Channel A Interrupt Status Register 2**

ISA1 **Channel A Interrupt Status Register 1**

ISA0 **Channel A Interrupt Status Register 0**

ISB2 **Channel B Interrupt Status Register 2**

ISB1 **Channel B Interrupt Status Register 1**

ISB0 **Channel B Interrupt Status Register 0**

These bits indicate, that an interrupt indication is pending in the corresponding interrupt status register(s) ISR0/ISR1/ISR2 of the serial communication controller (SCC):

bit='0' No interrupt indication is pending.

bit='1' An interrupt indication is pending.

Register 5 **GPDIR**
GPP Direction Register

CPU Accessibility: **read/write**

Reset Value: **FF_H**

Offset Address: **05_H**

typical usage: written by CPU evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	GPP I/O Direction Control							
	1	1	1	1	1	GP2DIR	GP1DIR	GP0DIR

GPnDIR **GPP Pin n Direction Control** (-)

This bit selects between input and output function of the corresponding GPP pin:

bit = '0' output

bit = '1' input (reset value)

Register 6 **GPDAT**
GPP Data Register

CPU Accessibility: **read/write**

Reset Value: **-**

Offset Address: $\mathbf{07_H}$

typical usage: written by CPU(outputs) and SEROCCO-D(inputs), evaluated by SEROCCO-D(outputs) and CPU(inputs)

Bit	7	6	5	4	3	2	1	0
	GPP Data I/O							
	-	0	-	-	-	GP2DAT	GP1DAT	GP0DAT

GPnDAT **GPP Pin n Data I/O Value** (-)

This bit indicates the value of the corresponding GPP pin:

bit = ’0’ If direction is input: input level is ’low’;
if direction is output: output level is ’low’.

bit = ’1’ If direction is input: input level is ’high’;
if direction is output: output level is ’high’.

Register 7 **GPIM**
GPP Interrupt Mask Register

CPU Accessibility: **read/write**

Reset Value: **FF_H**

Offset Address: **09_H**

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	GPP Interrupt Mask Bits							
	1	1	1	1	1	GP2IM	GP1IM	GP0IM

GPnIM **GPP Pin n Interrupt Mask** (-)

This bit controls the interrupt mask of the corresponding GPP pin:

bit = ’0’ Interrupt generation is enabled. An interrupt is generated on any state transition of the corresponding port pin (inputs).

bit = ’1’ Interrupt generation is disabled (reset value).

Register 8 **GPIS**
GPP Interrupt Status Register

CPU Accessibility: **read only**
Reset Value: **00_H**
Offset Address: **$0B_H$**
typical usage: written by SEROCCO-D, read and evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	GPP Interrupt Status Bits							
	0	0	0	0	0	GP2I	GP1I	GP0I

GPnI **GPP Pin n Interrupt Indiction** (-)

This bit indicates if an interrupt event occured on the corresponding GPP pin:

bit = '0' No interrupt indication is pending at this pin (no state transition has occured).

bit = '1' An interrupt indication is pending (a state transition occured). The interrupt indication is cleared after read access.

Register 9 **DCMDR**
DMA Command Register

CPU Accessibility: **read/write**
Reset Value: **00_H**
Offset Address: **$0C_H$**
typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	DMA Controller Reset Command Bits							
	RDTB	DTACK TB	RDRB	DTACK RB	RDTA	DTACK TA	RDRA	DTACK RA

RDTB **Reset DMA Transmit Channel B**
RDRB **Reset DMA Receive Channel B**
RDTA **Reset DMA Transmit Channel A**
RDRA **Reset DMA Receive Channel A**

Self-clearing command bit.
These bits bring the corresponding DMA channel to the reset state:

bit='0' No reset is performed.

bit='1' Reset is performed.

DTACKTB **DMA Transfer Ack Transmit Channel B**

DTACKRB **DMA Transfer Ack Receive Channel B**

DTACKTA **DMA Transfer Ack Transmit Channel A**

DTACKRA **DMA Transfer Ack Receive Channel A**

Only valid in internal DMA controller modes.

bit = '0'	The data transfer acknowledge signal on pin $\overline{\text{DTACK}}$/$\overline{\text{READY}}$ pin is ignored for data transfer cycles initiated by the SEROCCO-D DMA controller.
bit = '1'	For data transfer cycles initiated by the internal DMA controller the handshake signal $\overline{\text{DTACK}}$/$\overline{\text{READY}}$ is evaluated by SEROCCO-D. This can be used to extend the duration of read or write cycles.

Register 10 **DMODE**
DMA Mode Register

CPU Accessibility: **read/write**

Reset Value: $\mathbf{00_H}$

Offset Address: $\mathbf{0D_H}$

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	DMA Controller Operation Mode							
	0	TMODEB	0	RMODEB	0	TMODEA	0	RMODEA

TMODEB **Transmit DMA Mode Channel B**

RMODEB **Receive DMA Mode Channel B**

TMODEA **Transmit DMA Mode Channel A**

RMODEA **Receive DMA Mode Channel A**

These bits select the operating mode of the corresponding DMA channel:

’0’ Single Buffer Mode (standard mode)
Used base address registers are TBADDR1L/M/H (transmit) and RBADDR1L/M/H (receive).

’1’ Switched Buffer Mode
Base address registers switch alternating from TBADDR1L/M/H and RBADDR1L/M/H to TBADDR2L/M/H and RBADDR2L/M/H.
After reset, transmit and receive buffers #1 are selected.

Register 11 **DISR**
DMA Interrupt Status Register

CPU Accessibility: **read only**
Reset Value: **00_H**
Offset Address: **$0E_H$**
typical usage: written by SEROCCO-D, evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	DMA Interrupt Status Register							
	0	RBFB	RDTEB	TDTEB	0	RBFA	RDTEA	TDTEA

Note: Interrupt indications are stored even if masked in register DIMR. Pending interrupts get presented to the system as soon as they get unmasked.

RBFB **Receive Buffer Full Channel B**

RBFA **Receive Buffer Full Channel A**

If a receive buffer size is defined in registers RMBSL/RMBSH and during reception the end of the receive buffer is reached this interrupt is generated indicating that the receive buffer is full. The corresponding DMA channel suspends the write transfers to memory until a new buffer is allocated and resumes the reception.

RDTEB **Receive DMA Transfer End Channel B**

RDTEA **Receive DMA Transfer End Channel A**

This bit set to ’1’ indicates that a DMA transfer of receive data is finished and the receive data is completely moved to the corresponding receive buffer in host memory.
If this completed DMA transfer filled up the receive buffer (i.e. the receive byte count RBC matches the buffer size RMBS), bit RBFA/B is also set.

TDTEB **Transmit DMA Transfer End Channel B**

TDTEA **Transmit DMA Transfer End Channel A**

This bit set to '1' indicates that a DMA transfer of transmit data is finished and the data is completely moved from the transmit buffer to the on-chip transmit FIFO.

Register 12 **DIMR**
DMA Interrupt Mask Register

CPU Accessibility: **read/write**

Reset Value: **77_H**

Offset Address: **$0F_H$**

typical usage:

Bit	7	6	5	4	3	2	1	0
	DMA Interrupt Mask Register							
	0	MRBFB	MRDTEB	MTDTEB	0	MRBFA	MRDTEA	MTDTEA

MRBFB **Mask Receive Buffer Full Interrupt Channel B**

MRBFA **Mask Receive Buffer Full Interrupt Channel A**

MRDTEB **Mask Receive DMA Transfer End Interrupt Channel B**

MRDTEA **Mask Receive DMA Transfer End Interrupt Channel A**

MTDTEB **Mask Transmit DMA Transfer End Interrupt Channel B**

MTDTEA **Mask Transmit DMA Transfer End Interrupt Channel A**

If a bit in this interrupt mask register is set to '1', the corresponding interrupt is not generated and not indicated in the corresponding bit position in the DISR register. After reset all interrupts are masked.

5.2.2 Channel Specific SCC Registers

Each register description is organized in three parts:

- a head with general information about reset value, access type (read/write), channel specific offset addresses and usual handling;
- a table containing the bit information (name of bit positions) distinguished for the three major protocol modes HDLC/PPP (H), ASYNC (A) and BISYNC (B);
- a section containing the detailed description of each bit; the corresponding modes, the bit is valid for, are marked again by a bracket term right beside the full bit name.

Register 13 **FIFOL**
Receive/Transmit FIFO (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **-**

	Channel A	Channel B
Offset Address:	**10**$_H$	**60**$_H$

typical usage: XFIFO: written by CPU, evaluated by SEROCCO-D
RFIFO: written by SEROCCO-D, evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	RFIFO/XFIFO Access Low Byte							
	FIFO(7:0)							

Register 14 **FIFOH**
Receive/Transmit FIFO (High Byte)

CPU Accessibility: **read/write**

Reset Value: **-**

	Channel A	Channel B
Offset Address:	**11**$_H$	**61**$_H$

typical usage: XFIFO: written by CPU, evaluated by SEROCCO-D
RFIFO: written by SEROCCO-D, evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	RFIFO/XFIFO Access High Byte							
	FIFO(15:8)							

Receive FIFO (RFIFO)

Reading data from the RFIFO can be done in 8-bit (byte) or 16-bit (word) accesses, depending on the selected microprocessor bus width using signal 'WIDTH'. In 16-bit bus mode only 16-bit accesses to RFIFO are allowed. Only for a frame with odd byte count the last access can be an 8-bit access.

The size of the accessible part of RFIFO is determined by programming the RFIFO threshold level in bit field CCR3H.RFTH(1:0). If the HDLC/PPP protocol machine is selected, the threshold can be adjusted to 32 (reset value), 16, 4 or 2 bytes. With the ASYNC and BISYNC protocol machines following threshold levels can be selected: 1 (reset value), 4, 16 or 32 bytes.

- Interrupt Controlled Data Transfer (GMODE.IDMA='0')

Up to 32 bytes/16 words of received data can be read from the RFIFO following an RPF or an RME interrupt (see ISR0 register). The address provided during an RFIFO read access is not incremental; it is always 10_H for channel A or 60_H for channel B.

RPF Interrupt: This interrupt indicates that the adjusted receive threshold level is reached. The message is not yet complete. A fix number of bytes, dependent from the threshold level, has to be read.

RME Interrupt: The message is completely received. The number of valid **bytes** is determined by reading the RBCL, RBCH registers.

The content of the RFIFO is released by issuing the "Receive Message Complete" command (CMDRH.RMC).

- DMA Controlled Data Transfer (GMODE.IDMA='1')

If DMA operation is enabled, the SEROCCO-D takes over control of the receive FIFO accesses. Refer to **"Internal DMA Controller" on Page 81** for details.

Transmit FIFO (XFIFO)

Writing data to the XFIFO can be done in 8-bit (byte) or 16-bit (word) accesses, depending on the selected microprocessor bus width using signal 'WIDTH'. In 16-bit bus mode only 16-bit accesses to XFIFO are allowed. Only for a frame with odd byte count the last access must be an 8-bit access.

- Interrupt Controlled Data Transfer (GMODE.IDMA='0')

Following an XPR (or an ALLS) interrupt, up to 32 bytes/16 words of new transmit data can be written into the XFIFO. Transmit data can be released for transmission with an XTF command. The address provided during an XFIFO write access is not incremental; it is always 10_H for channel A or 60_H for channel B.

- DMA Controlled Data Transfer (GMODE.IDMA='1')

If DMA operation is enabled, the SEROCCO-D takes over control of the transmit FIFO accesses. Refer to **"Internal DMA Controller" on Page 81** for details.

Register 15 **STARL**
Status Register (Low Byte)

CPU Accessibility: **read only**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**12_H**	**62_H**

typical usage: updated by SEROCCO-D
read and evaluated by CPU

Bit	7	6	5	4	3	2	1	0
Mode	Command Status				Transmitter Status			
H	XREPE	0	0	CEC	0	XDOV	XFW	CTS
A	XREPE	0	TEC	CEC	FCS	XDOV	XFW	CTS
B	XREPE	0	0	CEC	0	XDOV	XFW	CTS

Register 16 **STARH**
Status Register (High Byte)

CPU Accessibility: **read only**

Reset Value: **10_H**

	Channel A	Channel B
Offset Address:	**13_H**	**63_H**

typical usage: updated by SEROCCO-D
read and evaluated by CPU

Bit	7	6	5	4	3	2	1	0
Mode	Receiver Status					Automode Status		
H	0	0	CD	RLI	DPLA	WFA	XRNR	RRNR
A	0	RFNE	CD	0	DPLA	0	0	0
B	0	RFNE	CD	SYNC	DPLA	0	0	0

XREPE **Transmit Repetition Executing** (all modes)

XREPE='0' No transmit repetition command is in execution.

XREPE='1' A XREP command (register CMDRL) is currently in execution.

TEC **TIC Executing** (async mode)

TIC='0' No TIC (transmit immediate character) is currently in transmission. Access to register TICR is allowed to initiate a TIC transmission.

TIC='1' A TIC command (write access to register TICR) is accepted but not completely executed. No further write access to register TICR is allowed until 'TIC' bit is cleared by SEROCCO-D.

CEC **Command Executing** (all modes)

CEC='0' No command is currently in execution. The command registers CMDRL/CMDRH can be written by CPU.

CEC='1' A command (written previously to registers CMDRL/CMDRH) is currently in execution. No further command can be written to registers CMDRL/CMDRH by CPU.

Note: CEC will stay active if the SCC is in power-down mode or if no serial clock, needed for command execution, is available.

FCS **Flow Control Status** (async mode)

If (in-band) flow control mechanism is enabled via bit 'FLON' in register CCR2H this bit indicates the current state of transmitter:

FCS='0' Transmitter is ready (always after transmitter reset command or XON-character detected).

FCS='1' Transmitter is stopped (XOFF-character detected).

XDOV **Transmit FIFO Data Overflow** (all modes)

XDOV='0' Less than or equal to 32 bytes have been written to the XFIFO.

XDOV='1' More than 32 bytes have been written to the XFIFO. This bit is reset by:
- a transmitter reset command 'XRES'
- or when all bytes in the accessible half of the XFIFO have been moved into the inaccessible half.

XFW **Transmit FIFO Write Enable** (all modes)

XFW='0' The XFIFO is not able to accept further transmit data.

XFW='1' Transmit data can be written to the XFIFO.

CTS **$\overline{\text{CTS}}$ (Clear To Send) Input Signal State** (all modes)

CTS='0' $\overline{\text{CTS}}$ input signal is inactive (high level)

CTS='1' $\overline{\text{CTS}}$ input signal is active (low level)

Note: A transmit clock is necessary to detect the input level of $\overline{\text{CTS}}$. Optionally this input can be programmed to generate an interrupt on signal level changes.

RFNE **Receive FIFO Not Empty** (async/bisync modes)

This status bit is set if the SCC receive FIFO (RFIFO) holds at least one valid byte.

RFNE='0' The receive FIFO is empty.

RFNE='1' The receive FIFO is not empty.

CD **CD (Carrier Detect) Input Signal State** (all modes)

This status bit gives the signal state of CD input. This bit value is independent of the programmed polarity of the Carrier Detect function (bit 'ICD' in register CCR1H).

CD='0' CD input signal is low.

CD='1' CD input signal is high.

Note: Optionally this input can be programmed to generate an interrupt on signal level changes.

SYNC **Synchronization Status** (bisync mode)

This bit indicates whether the receiver is in synchronized state. After a 'HUNT' command 'SYNC' bit is cleared and the receiver starts searching for a SYNC character. When found the 'SYNC' status bit is set immediately, an SCD-interrupt is generated (if enabled) and receive data is forwarded to the receiver FIFO.

SYNC='0'	Synchronization is lost or not yet achieved. (after reset or after new 'HUNT' command has been issued and before SYNC character is found)
SYNC='1'	The receiver is in synchronized state.

RLI **Receive Line Inactive** (hdlc mode)

This bit indicates that neither flags as interframe time fill nor data are being received via the receive line.

RLI='0'	Receive line is active, no constant high level is detected.
RLI='1'	Receive line is inactive, i.e. more than 7 consecutive '1' are detected on the line.

Note: A receive clock must be provided in order to detect the receive line state.

DPLA **DPLL Asynchronous** (all modes)

This bit is only valid if the receive clock is recovered by the DPLL and FM0, FM1 or Manchester data encoding is selected. It is set when the DPLL has lost synchronization. In this case reception is disabled (receive abort condition) until synchronization has been regained. In addition transmission is interrupted in all cases where transmit clock is derived from the DPLL (clock mode 3a, 7a). Interruption of transmission is performed the same way as on deactivation of the $\overline{\text{CTS}}$ signal.

DPLA='0'	DPLL is synchronized.
DPLA='1'	DPLL is asynchronous (re-synchronization process is started automatically).

WFA **Wait For Acknowledgement** (hdlc mode)

This status bit is significant in Automode only. It indicates whether the Automode state machine expects an acknowledging I- or S-Frame for a previously sent I-Frame.

WFA='0'	No acknowledge I/S-Frame is expected.
WFA='1'	The Automode state machine is waiting for an achnowledging S- or I-Frame.

XRNR **Transmit RNR Status** (hdlc mode)

This status bit is significant in Automode only. It indicates the receiver status of the local station (SCC).

XRNR='0'	The receiver is ready and will automatically answer poll-frames with a S-Frame with 'receiver-ready' indication.
XRNR='1'	The receiver is NOT ready and will automatically answer poll-frames with a S-Frame with a 'receiver-not-ready' indication.

RRNR **Received RNR (Receiver Not Ready) Status** (hdlc mode)

This status bit is significant in Automode only. It indicates the receiver status of the remote station.

RRNR='0'	The remote station receiver is ready.
RRNR='1'	The remote receiver is NOT ready. (A 'receiver-not-ready' indication was received from the remote station)

Register 17 CMDRL
Command Register (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**14_H**	**64_H**

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Timer		Transmitter Commands					
H	STI	TRES	XIF	XRES	XF	XME	XREP	0
A	STI	TRES	TXON	XRES	XF	XME	XREP	TXOFF
B	STI	TRES	0	XRES	XF	XME	XREP	0

Register 18 CMDRH
Command Register (High Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**15_H**	**65_H**

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receiver Commands							
H	RMC	RNR	0	0	RSUC	0	0	RRES
A	RMC	0	0	0	0	0	RFRD	RRES
B	RMC	0	0	0	HUNT	0	RFRD	RRES

The command register contains self-clearing command bits. The command bits read a ’1’ until the corresponding command is executed completely.

For a write access to the register, the new value gets OR'ed with the current register contents.

The 'CEC' bit in register STARL/STARH is the OR-function over all command bits.

STI **Start Timer Command** (all modes)

Self-clearing command bit:
HDLC Automode:
In HDLC Automode the timer is used internally for the autonomous protocol support functions. The timer is started automatically by the SCC when an I-Frame is sent out and needs to be acknowledged.
If the 'STI' command is issued by software:

STI='1' An S-Frame with poll bit set is sent out and the internal timer is started expecting an acknowledge from the remote station via an I- or S-Frame.
The timer is stopped after receiving an acknowledge otherwise the timer expires generating a timer interrupt.

Note: In HDLC Automode, bit 'TMD' in register TIMR3 must be set to '1'

All protocol modes except HDLC Automode:
In these modes the timer is operating as a general purpose timer.

STI='1' This commands starts timer operation.
The timer can be stopped by setting bit 'TRES'.

Note: Bit 'TMD' in register TIMR3 must be cleared for proper operation

TRES **Timer Reset** (all modes)

Self-clearing command bit.
This bit deactivates timer operation:

TRES='0' Timer operation enabled.

TRES='1' Timer operation stopped.

XIF **Transmit I-Frame** (hdlc mode)

Self-clearing command bit.
This command bit is significant in HDLC Automode only.

XIF='1' Initiates the transmission of an I-frame in auto-mode.
Additional to the opening flag, the address and control fields of the frame are added by SEROCCO-D.

TXOFF **Transmit Off Command** (async mode)

Self-clearing command bit:
This command bit is significant if in-band flow-control is selected.

TXOFF='1' Forces the transmitter to enter its 'transmit off' state. This is equal to receiving an XOFF character.

TXON **Transmit On Command** (async mode)

Self-clearing command bit:
This command bit is significant if in-band flow-control is selected.

TXON='1' Forces the transmitter to enter its 'transmit on' state. This is equal to receiving an XON character.

XRES **Transmitter Reset Command** (all modes)

Self-clearing command bit:

XRES='1' The SCC transmit FIFO is cleared and the transmitter protocol engines are reset to their initial state.
A transmitter reset command is recommended after all changes in protocol mode configurations (e.g. switching between the protocol engines HDLC/ASYNC/BISYNC or sub-modes of HDLC).

XF **Transmit Frame** (all modes)

This self-clearing command bit is significant in interrupt driven operation only (GMODE.IDMA='0').

XF='1' After having written up to 32 bytes to the XFIFO, this command initiates transmission. In packet oriented protocols like HDLC/PPP the opening flag is automatically added by SEROCCO-D. If the end of the packet is part of the transmit data, bit 'XME' should be set in addition.

DMA Mode

After having written the length of the data block to be transmitted to registers XBC1L and XBC1H, this command initiates the data transfer from host memory to SEROCCO-D by DMA. Transmission on the serial side starts as soon as 32 bytes are transferred to the XFIFO or the transmit byte counter value is reached.

XME	**Transmit Message End** (hdlc/bisync modes)	
	Self-clearing command bit:	
	XME='1'	Indicates that the data block written last to the XFIFO contains the end of the packet. This bit should always be set in conjunction with a transmit command ('XF' or 'XIF').
XREP	**Transmission Repeat Command** (hdlc mode)	
	Self-clearing command bit:	
	XREP='1'	If bit 'XREP' is set together with bit 'XME' and 'XF', SEROCCO-D repeatedly transmits the contents of the XFIFO (1..32 bytes). The cyclic transmission can be stopped with the 'XRES' command.
RMC	**Receive Message Complete** (all modes)	
	Self-clearing command bit:	
	RMC='1'	With this bit the CPU indicates to SEROCCO-D that the current receive data has been fetched out of the RFIFO. Thus the corresponding space in the RFIFO can be released and re-used by SEROCCO-D for further incoming data.
RNR	**Receiver Not Ready Command** (hdlc mode)	
	NON self-clearing command bit: This command bit is significant in HDLC Automode only.	
	RNR='0'	Forces the receiver to enter its 'receiver-ready' state. The receiver acknowledges received poll or I-Frames with a 'receiver-ready' indication.
	RNR='1'	Forces the receiver to enter its 'receiver-not-ready' state. The receiver acknowledges received poll or I-Frames with a 'receiver-not-ready' indication.
RSUC	**Reset Signaling Unit Counter** (hdlc mode)	
	Self-clearing command bit: This command bit is significant if HDLC SS7 mode is selected.	
	RSUC='1'	The Signaling System #7 (SS7) unit counter is reset.

HUNT	**Enter Hunt State Command** (bisync mode)	
	Self-clearing command bit:	
	HUNT='1'	This command forces the receiver to enter its 'HUNT' state immediately. Thus synchronization is 'lost' and the receiver starts searching for new SYNC characters.
RFRD	**Receive FIFO Read Enable Command** (async/bisync modes)	
	Self-clearing command bit:	
	RFRD='1'	This command forces insertion of a 'block end' condition into the RFIFO before the receive FIFO threshold is exceeded or a block end condition (termination character detected or time-out) is fulfilled. The execution of this command is reported with a TCD interrupt.
RRES	**Receiver Reset Command** (all modes)	
	Self-clearing command bit:	
	RRES='1'	The SCC receive FIFO is cleared and the receiver protocol engines are reset to their initial state. The SCC receive FIFO accepts new receive data from the protocol engine immediately after receiver reset procedure. It is recommended to disable data reception before issuing a receiver reset command by setting bit CCR3L.RAC = '0' and enabling data reception afterwards. A 'receiver reset' command is recommended after all changes in protocol mode configurations (switching between the protocol engines HDLC/ASYNC/BISYNC or sub-modes of HDLC).

Register 19 **CCR0L**
Channel Configuration Register 0 (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **00$_H$**

	Channel A	Channel B
Offset Address:	**16$_H$**	**66$_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	misc.				Clock Mode Selection			
H	VIS	PSD	BCR	TOE	SSEL	CM(2:0)		
A	VIS	PSD	BCR	TOE	SSEL	CM(2:0)		
B	VIS	PSD	0	TOE	SSEL	CM(2:0)		

Register 20 **CCR0H**
Channel Configuration Register 0 (High Byte)

CPU Accessibility: **read/write**

Reset Value: **00$_H$**

	Channel A	Channel B
Offset Address:	**17$_H$**	**67$_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Power	Line Coding			Protocol Mode			
H	PU	SC(2:0)			0	0	SM(1:0)	
A	PU	SC(2:0)			0	0	SM(1:0)	
B	PU	SC(2:0)			0	0	SM(1:0)	

VIS **Masked Interrupts Visible** (all modes)

VIS='0' Masked interrupt status bits are not displayed in the interrupt status registers (ISR0..ISR2).

VIS='1' Masked interrupt status bits are visible and automatically cleared after interrupt status register (ISR0..ISR2) read access.

Note: Interrupts masked in registers IMR0..IMR2 will not generate an interrupt.

PSD **DPLL Phase Shift Disable** (all modes)

This option is only applicable in the case of NRZ or NRZI line encoding is selected.

PSD='0' Normal DPLL operation.

PSD='1' The phase shift function of the DPLL is disabled. The windows for phase adjustment are extended.

BCR **Bit Clock Rate** (async PPP, ASYNC modes)

This bit is only valid in asynchronous PPP and ASYNC protocol mode and only in clock modes not using the DPLL (0, 1, 3b, 7b). It is also invalid in clock mode 4.

BCR='0' Selects isochronous operation with bit clock rate 1. Data bits are sampled once.

BCR='1' Selects standard asynchronous operation with bit clock rate 16. Using 16 samples per bit, data bits are sampled 3 times around the nominal bit center. The resulting bit value is determined by majority decision of the 3 samples. For correct operation NRZ data encoding has to be selected.

TOE **Transmit Clock Out Enable** (all modes)

For clock modes 0b, 2b, 3a, 3b, 6b, 7a and 7b, the internal transmit clock can be monitored on pin TxCLK as an output signal. In clock mode 5, a time slot control signal marking the active transmit time slot is output on pin TxCLK.
Bit 'TOE' is invalid for all other clock modes.

TOE='0'	TxCLK pin is input.
TOE='1'	TxCLK pin is switched to output function if applicable for the selected clock mode.

SSEL **Clock Source Select** (all modes)

Distinguishes between the 'a' and 'b' option of clock modes 0, 2, 3, 5, 6 and 7.

SSEL='0'	Option 'a' is selected.
SSEL='1'	Option 'b' is selected.

CM(2:0) **Clock Mode** (all modes)

This bit field selects one of main clock modes 0..7. For a detailed description of the clock modes refer to **Chapter 3.2.3**

CM = '000'	clock mode 0
CM = '001'	clock mode 1
CM = '010'	clock mode 2
CM = '011'	clock mode 3
CM = '100'	clock mode 4
CM = '101'	clock mode 5 (time-slot oriented clocking modes)
CM = '110'	clock mode 6
CM = '111'	clock mode 7

PU **Power Up** (all modes)

PU='0' The SCC is in 'power-down' mode. The protocol engines are switched off (standby) and no operation is performed. This may be used to save power when SCC is not in use.

Note: The SCC transmit FIFO accepts transmit data even in 'power-down' mode.

PU='1' The SCC is in 'power-up' mode.

SC(2:0) **Serial Port Configuration** (all modes)

This bit field selects the line coding of the serial port.
Note, that special operation modes and settings may require or exclude operation in special line coding modes. Refer to the 'prerequisites' in the dedicated mode descriptions.

SC = '000'	NRZ data encoding
SC = '001'	Bus configuration, timing mode 1 (NRZ data encoding)
SC = '010'	NRZI data encoding
SC = '011'	Bus configuration, timing mode 2 (NRZ data encoding)
SC = '100'	FM0 data encoding
SC = '101'	FM1 data encoding
SC = '110'	Manchester data encoding
SC = '111'	Reserved

Note: If bus configuration mode is selected, only NRZ data encoding is supported.

SM(1:0) **Serial Port Mode** (all modes)

This bit field selects one of the three protocol engines.
Depending on the selected protocol engine the SCC related registers change or special bit positions within the registers change their meaning.

SM = '00'	HDLC/PPP protocol engine
SM = '01'	*Reserved* (do not use)
SM = '10'	BISYNC protocol engine
SM = '11'	ASYNC protocol engine

Register 21 **CCR1L**
Channel Configuration Register 1 (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**18_H**	**68_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	misc.							
H	CRL	C32	SOC(1:0)		SFLG	DIV	ODS	0
A	0	0	0	0	0	DIV	ODS	0
B	0	0	0	0	0	DIV	ODS	0

Register 22 **CCR1H**
Channel Configuration Register 1 (High Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**19_H**	**69_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	misc.							
H	0	ICD	0	RTS	FRTS	FCTS	CAS	TSCM
A	0	ICD	0	RTS	FRTS	FCTS	CAS	TSCM
B	0	ICD	0	RTS	FRTS	FCTS	CAS	TSCM

CRL **CRC Reset Value** (hdlc mode)

This bit defines the initial value of the internal transmit/receive CRC generators:

CRL='0' Initial value is $0xFFFF_H$ (16 bit CRC), $0xFFFFFFFF_H$ (32 bit CRC).
This is the default value for most HDLC/PPP applications.

CRL='1' Initial value is $0x0000_H$ (16 bit CRC), $0x00000000_H$ (32 bit CRC).

C32 **CRC 32 Select** (hdlc mode)

This bit enables 32-bit CRC operation for transmit and receive.

C32='0' 16-bit CRC-CCITT generation/checking.

C32='1' 32-bit CRC generation/checking.

Note: The internal 'valid frame' criteria is updated depending on the selected number of CRC-bytes.

SOC(1:0) **Serial Output Control** (hdlc mode)

This bit field selects the $\overline{RTS}$ signal output function.
(This bit field is only valid in bus configuration modes selected via bit field SC(2:0) in register CCR0H).

SOC = '0X' $\overline{RTS}$ ouput signal is active during transmission of a frame (active low).

SOC = '10' $\overline{RTS}$ ouput signal is always inactive (high).

SOC = '11' $\overline{RTS}$ ouput signal is active during reception of a frame (active low).

SFLG **Shared Flags Transmission** (hdlc mode)

This bit enables 'shared flag transmission' in HDLC protocol mode. If another transmit frame begin is stored in the SCC transmit FIFO, the closing flag of the preceding frame becomes the opening flag of the next frame (shared flags):

SFLG = '0' Shared flag transmission disabled.

SFLG = '1' Shared flag transmission enabled.

Note: The receiver always supports shared flags and shared zeros of consecutive flags.

DIV **Data Inversion** (all modes)

This bit is only valid if NRZ data encoding is selected via bit field SC(2:0) in register CCR0H.

DIV='0' No Data Inversion.

DIV='1' Data is transmitted/received inverted (on a per bit basis). In HDLC and HDLC Synchronous PPP modes the continuous '1' idle sequence is NOT inverted. Thus it is recommended to select the flag sequence for interframe time fill transmission (CCR2H:ITF = '1'), which is inverted.

ODS **Output Driver Select** (all modes)

The transmit data output pin TxD can be configured as push/pull or open drain output chracteristic.

ODS='0' TxD pin is open drain output.

ODS='1' TxD pin is push/pull output.

ICD **Invert Carrier Detect Pin Polarity** (all modes)

ICD='0' Carrier Detect (CD) input pin is active high.

ICD='1' Carrier Detect (CD) input pin is active low.

RTS **Request To Send Pin Control** (all modes)

The request to send pin $\overline{\text{RTS}}$ can be controlled by SEROCCO-D as an output autonomously or via setting/clearing bit 'RTS'.
This bit is not valid in clock mode 4.

RTS='0' Pin $\overline{\text{RTS}}$ (output) pin is controlled by SEROCCO-D autonomously.

HDLC Mode:

$\overline{\text{RTS}}$ is activated during transmission. In bus configuration mode the functionality depends on bit field 'SOC' setting.

Note: For autonomous $\overline{\text{RTS}}$ pin control a transmit clock is necessary.

ASYNC/BISYNC Mode:

The functionality depends on setting of bit 'FRTS'

RTS='1' Pin $\overline{\text{RTS}}$ can be controlled by software. The output level of this pin depends on bit 'FRTS'.

FRTS **Flow Control (using signal $\overline{\text{RTS}}$)** (all modes)

Bit 'FRTS' together with bit 'RTS' determine the function of signal $\overline{\text{RTS}}$:

RTS, FRTS

RTS,	FRTS	
0,	0	Pin $\overline{\text{RTS}}$ is controlled by SEROCCO-D autonomously. $\overline{\text{RTS}}$ is activated (low) as soon as transmit data is available within the SCC transmit FIFO.
0,	1	Pin $\overline{\text{RTS}}$ is controlled by SEROCCO-D autonomously supporting bi-directional data flow control. $\overline{\text{RTS}}$ is activated (low) if the shadow part of the SCC receive FIFO is empty and de-activated (high) when the SCC receive FIFO fill level reaches its receive FIFO threshold.
1,	0	Forces pin $\overline{\text{RTS}}$ to active state (low).
1,	1	Forces pin $\overline{\text{RTS}}$ to inactive state (high).

FCTS **Flow Control (using signal $\overline{\text{CTS}}$)** (all modes)

This bit controls the function of pin $\overline{\text{CTS}}$.

FCTS = '0' The transmitter is stopped if $\overline{\text{CTS}}$ input signal is inactive (high) and enabled if active (low).

Note: In the character oriented protocol modes (ASYNC, BISYNC, asynchronous PPP), the current byte is completely sent even if $\overline{\text{CTS}}$ becomes inactive during transmission.

FCTS = '1' The transmitter is enabled, disregarding $\overline{\text{CTS}}$ input signal.

CAS **Carrier Detect Auto Start** (all modes)

CAS = ’0’ The CD pin is used as general input.
In clock mode 1, 4 and 5, clock mode specific control signals must be provided at this pin (receive strobe, receive gating $\overline{RCG}$, frame sync clock $\overline{FSC}$).
A pull-up/down resistor is recommended if unused.

CAS = ’1’ The CD pin enables/disables the receiver for data reception. (Polarity of CD pin can be configured via bit ’ICD’.)

Note: (1) In clock mode 1, 4 and 5 this bit must be set to ’0’.
(2) A receive clock must be provided for the autonomous receiver control function of the CD input pin.
(3) In ASYNC mode the transmitter is additionally controlled by in-band flow control mechanism (if enabled).

TSCM **Time Slot Control Mode** (all modes)

This bit controls internal counter operation in time slot oriented clock mode 5:

TSCM=’0’ The internal counter keeps running, restarting with zero after being expired.

TSCM=’1’ The internal counter stops at its maximum value and restarts with the next frame sync pulse again.

Register 23 **CCR2L**
Channel Configuration Register 2 (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$1A_H$**	**$6A_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	misc.							
H	MDS(1:0)		ADM	NRM	PPPM(1:0)		TLPO	TLP
A	0	0	0	0	0	0	TLPO	TLP
B	0	0	0	0	SLEN	BISNC	TLPO	TLP

Register 24 **CCR2H**
Channel Configuration Register 2 (High Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$1B_H$**	**$6B_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	misc.							
H	MCS	EPT	NPRE(1:0)		ITF	0	OIN	XCRC
A	0	0	0	0	0	0	0	FLON
B	0	EPT	NPRE(1:0)		ITF	0	CAPP	CRCM

MDS(1:0) **Mode Select** (hdlc modes)

This bit field selects the HDLC protocol sub-mode including the 'extended transparent mode'.

MDS = '00' Automode.

MDS = '01' Address Mode 2.

MDS = '10' Address Mode 0/1.
(Option '0' or '1' is selected via bit 'ADM'.)

MDS = '11' Extended transparent mode (bit transparent transmission/ reception).

Note: 'MDS(1:0)' must be set to '10' if any PPP mode is enabled via bit field 'PPPM' or if SS7 is enabled via bit 'ESS7' in register CCR3L.

ADM **Address Mode Select** (hdlc modes)

The meaning of this bit depends on the selected protocol sub-mode:

Automode, Address Mode 2:
Determines the address field length of an HDLC frame.

ADM = '0' 8-bit address field.

ADM = '1' 16-bit address field.

Address Mode 0/1:
Determines whether address mode 0 or 1 is selected.

ADM = '0' Address Mode 0 (no address recognition).

ADM = '1' Address Mode 1 (high byte address recognition).

Extended Transparent Mode:

ADM = '1' recommended setting

NRM **Normal Response Mode** (hdlc modes)

This bit is valid in HDLC Automode operation only and determines the function of the Automode LAP-Controller:

NRM = '0' Full-duplex LAP-B / LAP-D operation.

NRM = '1' Half-duplex normal response mode (NRM) operation.

PPPM(1:0) **PPP Mode Select** (hdlc modes)

This bit field enables and selects the HDLC PPP protocol modes:

PPPM = '00' No PPP protocol operation. The HDLC sub-mode is determined by bit field 'MDS'.

PPPM = '01' Octet synchronous PPP protocol operation.

PPPM = '10' Asynchronous PPP protocol operation.
Bit 'BCR' in register CCR0L must be set to ensure proper asynchronous reception.

PPPM = '11' Bit synchronous PPP protocol operation.

Note: 'Address Mode 0' must be selected by setting bit field 'MDS(1:0)' to '10' and bit 'ADM' to '0' if any PPP mode is enabled.

TLPO **Test Loop Out Function** (all modes)

This bit is only valid if test loop is enabled and controls whether test loop transmit data is driven on pin TxD:

TLPO = '0' Test loop transmit data is driven to TxD pin.

TLPO = '1' Test loop transmit data is NOT driven to TxD pin. TxD pin is idle '1'. Depending on the selected output characteristic the pin is high impedance (bit CCR1L.ODS ='0') or driving high (CCR1L.ODS ='1').

TLP **Test Loop** (all modes)

This bit controls the internal test loop between transmit and receive data signals. The test loop is closed at the far end of serial transmit and receive line just before the respective TxD and RxD pins:

TLP = '0' Test loop disabled.

TLP = '1' Test loop enabled.
The software is responsible to select a clock mode which allows correct reception of transmit data depending on the external clock supply. Transmit data is sent out via pin TxD if not disabled with bit 'TLPO'. The receive input pin RxD is internally disconnected during test loop operation.

SLEN **SYNC Character Length** (bisync mode)

This bit selects the SYNC character length in BISYNC/MONOSYNC operation mode:

SLEN = '0' 6 bit (MONOSYNC), 12 bit (BISYNC).

SLEN = '1' 8 bit (MONOSYNC), 16 bit (BISYNC).

BISNC **Select MONOSYNC/BISYNC Mode** (bisync mode)

This bit selects BISYNC or MONOSYNC operation mode:

BISNC = '0' MONOSYNC mode.

BISNC = '1' BISYNC mode.

MCS **Modulo Count Select** (hdlc modes)

This bit is valid in HDLC Automode operation only and determines the control field format:

MCS = '0' Basic operation, one byte control field (modulo 8 counter operation).

MCS = '1' Extended operation, two bytes control field (modulo 128 counter operation).

EPT **Enable Preamble Transmission** (hdlc/bisync mode)

This bit enables preamble transmission. The preamble is started after interframe time fill (ITF) transmission is stopped because a new frame is ready to be transmitted. The preamble pattern consists of 8 bits defined in register PREAMB, which is sent repetitively. The number of repetitions is determined by bit field 'PRE(1:0)':

EPT='0' Preamble transmission is disabled.

EPT='1' Preamble transmission is enabled.

Note: Preamble operation does NOT influence HDLC shared flag transmission if enabled.

NPRE(1:0) **Number of Preamble Repetitions** (hdlc/bisync mode)

This bit field determines the number of preambles transmitted:

NPRE = ’00’ 1 preamble.

NPRE = ’01’ 2 preambles.

NPRE = ’10’ 4 preambles.

NPRE = ’11’ 8 preambles.

ITF **Interframe Time Fill** (hdlc/bisync mode)

This bit selects the idle state of the transmit pin TxD:

ITF=’0’ Continuous logical ’1’ is sent during idle phase.

ITF=’1’ **HDLC Mode:**
Continuous flag sequences are sent (’01111110’ flag pattern).
BISYNC Mode:
Continuous SYN characters are output.

Note: It is recommended to clear bit ’ITF’ in bus configuration modes, i.e. continuous ’1’s are sent as idle sequence and data encoding is NRZ.

OIN **One Insertion** (hdlc mode)

In HDLC mode a one-insertion mechanism similar to the zero-insertion can be activated:

OIN=’0’ The ’1’ insertion mechanism is disabled.

OIN=’1’ In transmit direction a logical ’1’ is inserted to the serial data stream after 7 consecutive zeros.
In receive direction a ’1’ is deleted from the receive data stream after receiving 7 consecutive zeros.
This enables clock information to be recovered from the receive data stream by means of a DPLL, even in the case of NRZ data encoding, because a transition at bit cell boundary occurs at least every 7 bits.

XCRC **Transmit CRC Checking Mode** (hdlc mode)

XCRC='0' The transmit checksum (2 or 4 bytes) is generated and appended to the transmit data automatically.

XCRC='1' The transmit checksum is not generated automatically. The checksum is expected to be provided by software as the last 2 or 4 bytes in the transmit data buffer.

FLON **Flow Control Enable** (async mode)

In ASYNC mode, in-band flow control is supported:

FLON='0' No automatic in-band flow-control is performed. However recognition of a flow control character (XON/XOFF) causes always a maskable interrupt event.

FLON='1' Automatic in-band flow-control is performed.
Reception of a XOFF character (defined in register XNXF) turns off the transmitter after the currently transmitted character has been shifted out completely (XOFF state).
Reception of a XON character (defined in register XNXF) resumes the transmitter from XOFF into XON state ready to send available transmit data bytes.
The current flow control state is indicated via bit 'FCS' in register Star.
Any transmitter reset switches the flow-control logic to XON state.

CAPP **CRC Append** (bisync mode)

In BISYNC mode the CRC generator can be activated:

CAPP = '0' No CRC generation/checking is active in BISYNC mode.

CAPP = '1' The CRC generator is activated:

1. The CRC generator is initialized every time the transmission of a new 'frame' starts. The CRC initialization value can be selected via bit 'CRL' in register CCR2 (for BISYNC operation).
2. The CRC is automatically to the last transmitted data of a 'frame'.

CRCM **CRC Mode Select** (bisync mode)

In BISYNC mode the CRC generator can be configured for two different generator polynoms:

CRCM = ’0’ CRC-16:
The polynominal is $x^{16}+x^{15}+x^{2}+1$.

CRCM = ’1’ CRC-CCITT:
The polynominal is $x^{16}+x^{15}+x^{5}+1$.

Register 25 **CCR3L**
Channel Configuration Register 3 (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$1C_H$**	**$6C_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	misc.							
H	ELC	AFX	CSF	SUET	RAC	0	0	ESS7
A	TCDE	0	CHL(1:0)		RAC	DXS	XBRK	STOP
B	TCDE	SLOAD	CHL(1:0)		RAC	0	0	STOP

Register 26 **CCR3H**
Channel Configuration Register 3 (High Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$1D_H$**	**$6D_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	misc.							
H	0	DRCRC	RCRC	RADD	0	0	RFTH(1:0)	
A	PAR(1:0)		PARE	DPS	RFDF	0	RFTH(1:0)	
B	PAR(1:0)		PARE	DPS	RFDF	0	RFTH(1:0)	

ELC **Enable Length Check** (hdlc mode)

This bit is only valid in HDLC SS7 mode:
If the number of received octets exceeds 272 + 7 within one Signaling Unit, reception is aborted and bit RSTA.RAB is set.

ELC='0' Length Check disabled.

ELC='1' Length Check enabled.

TCDE **Termination Character Detection Enable** (async/bisync modes)

This bit is valid in ASYNC/BISYNC modes only and enables/disables the termination character detection mechanism:

TCDE = '0' No receive termination character detection is performed.

TCDE = '1' The termination character detection is enabled. The receive data stream is monitored for the occurence of a termination character (TC) programmed via register TCR. When this character is detected, a 'TCD' interrupt is generated to the CPU (unless masked).

Note: If the programmed character length (bit field 'CHL(1:0)') is less than 8 bits, the most significant unused bits in register TCR must be set to '0'. Otherwise no termination character will be detected.

AFX **Automatic FISU Transmission** (hdlc mode)

This bit is only valid in HDLC SS7 mode:
After the contents of the transmit FIFO (XFIFO) has been transmitted completely, FISUs are transmited automatically. These FISUs contain the FSN and BSN of the last transmitted Signaling Unit (provided in XFIFO).

AFX='0' Automatic FISU transmission disabled.

AFX='1' Automatic FISU transmission enabled.

SLOAD **Enable SYN Character Load** (bisync mode)

In BISYNC mode, SYN characters might be filtered out or stored to the SCC receive FIFO.

SLOAD='0' SYN characters are filtered out and not stored in the receive FIFO.

SLOAD='1' All received characters, including SYN characters, are stored in the receive FIFO.

CSF **Compare Status Field** (hdlc mode)

This bit is only valid in HDLC SS7 mode:
If the status fields of consecutive LSSUs are equal, only the first will be stored and every following is ignored

CSF='0' Compare is disabled, all received LSSUs are stored in the receive FIFO.

CSF='1' Compare is enabled, only the first one of consecutive equal LSSUs is stored in the receive FIFO.

SUET **Signalling Unit Counter Threshold** (hdlc mode)

This bit is only valid in HDLC SS7 mode:
Defines the number of signaling units received in error that will cause an error rate high indication (ISR1.SUEX).

SUET='0' threshold is 64 errored signaling units.

SUET='1' threshold is 32 errored signaling units.

CHL(1:0) **Character Length** (async/bisync modes)

This bit field selects the number of data bits within a character:

CHL = '00' 8-bit data.

CHL = '01' 7-bit data.

CHL = '10' 6-bit data.

CHL = '11' 5-bit data.

RAC **Receiver active** (all modes)

Switches the receiver between operational/inoperational states:

RAC='0' Receiver inactive, receive line is ignored.

RAC='1' Receiver active.

DXS **Disable Storage of XON/XOFF Characters** (async mode)

In ASYNC mode, XON/XOFF characters might be filtered out or stored to the SCC receive FIFO:

DXS='0' All received characters including XON/XOFF characters are stored in the receive FIFO.

DXS='1' XON/XOFF characters are filtered out and not stored in the receive FIFO.

XBRK **Transmit Break** (async mode)

XBRK='0' Normal transmit operation.

XBRK='1' Forces the TxD pin to 'low' level immediately (break condition), regardless of any character being currently transmitted. This command is executed immediately with the next rising edge of the transmit clock and further transmission is disabled. The currently sent character is lost.
Data stored in the SCC transmit FIFO will be sent as soon as the break condition is cleared (XBRK='0'). A transmit reset command (bit 'XRES' in register CMDRL) does NOT clear the break condition automatically.

ESS7 **Enable SS7 Mode** (hdlc mode)

This bit is only valid in HDLC mode only.

ESS7='0' Disable signaling system #7 (SS7) support.

ESS7='1' Enable signaling system #7 (SS7) support.

Note: If SS7 mode is enabled, 'Address Mode 0' must be selected by setting bit field CCR2L:MDS(1:0) to '10' and bit CCR2L:ADM to '0'.

STOP **Stop Bit number** (async mode)

This bit selects the number of stop bits per ASYNC character:

STOP='0' 1 stop bit per character.

STOP='1' 2 stop bits per character.

PAR(1:0) **Parity Format** (async/bisync modes)

This bit field selects the parity generation/checking mode:

PAR = '00' SPACE ('0'), a constant '0' is inserted as parity bit.

PAR = '01' Odd parity.

PAR = '10' Even parity.

PAR = '11' MARK ('1'), a constant '1' is inserted as parity bit.

The received parity bit is stored in the SCC receive FIFO depending on the selected character format:

- as leading bit immediately preceding the data bits if character length is 5, 6 or 7 bits and bit 'DPS' is cleared ('0').
- as LSB of the status byte belonging to the character if character length is 8 bits and the corresponding receive FIFO data format is selected (bit 'RFDF' = '1').

A parity error is indicated in the MSB of the status byte belonging to each character if enabled. In addition, a parity error interrupt can be generated.

DRCRC **Disable Receive CRC Checking** (hdlc mode)

DRCRC='0' The receiver expects a 16 or 32 bit CRC within a HDLC frame. CRC processing depends on the setting of bit 'RCRC'.
Frames shorter than expected are marked 'invalid' or are discarded (refer to RSTA description).

DRCRC='1' The receiver does not expect any CRC within a HDLC frame. The criteria for 'valid frame' indication is updated accordingly (refer to RSTA description).
Bit 'RCRC' is ignored.

RCRC **Receive CRC Checking Mode** (hdlc mode)

RCRC='0' The received checksum is evaluated, but NOT forwarded to the receive FIFO.

RCRC='1' The received checksum (2 or 4 bytes) is evaluated and forwarded to the receive FIFO as data.

PARE **Parity Enable** (async/bisync modes)

PARE='0' Parity generation/checking is disabled.

PARE='1' Parity generation/checking is enabled.

RADD **Receive Address Forward to RFIFO** (hdlc mode)

This bit is only valid
- if an HDLC sub-mode with address field support is selected (Automode, Address Mode 2, Address Mode 1)
- in SS7 mode

RADD='0' The received HDLC address field (either 8 or 16 bit, depending on bit 'ADM') is evaluated, but NOT forwarded to the receive FIFO.
In SS7 mode, the signaling unit fields 'FSN' and 'BSN' are NOT forwarded to the receive FIFO.

RADD='1' The received HDLC address field (either 8 or 16 bit, depending on bit 'ADM') is evaluated and forwarded to the receive FIFO.
In SS7 mode, the signaling unit fields 'FSN' and 'BSN' are forwarded to the receive FIFO.

DPS **Data Parity Storage** (async/bisync modes)

Only valid if parity generation/checking is enabled via bit 'PARE':

DPS='0' The parity bit is stored.

DPS='1' The parity bit is not stored in the data byte containing character data.
The parity bit is always stored in the status byte.

RFDF **Receive FIFO Data Format** (async/bisync mode)

In ASYNC mode, the character format is determined as follows:

RFDF='0'

Data Byte:

7		5	4 – 0
		P	Char5

7	6	5 – 0
	P	Char6

7	6 – 0
P	Char7

7 – 0
Char8

(no parity bit stored)

RFDF='1'

Data Byte (DB):

7		5	4 – 0
		P	Char5

7	6	5 – 0
	P	Char6

7	6 – 0
P	Char7

7 – 0
Char8

(no parity bit stored)

Status Byte (SB) (one for each data byte format):

7	6		0
PE	FE		P

P: Parity bit stored in data byte (can be disabled via bit 'DPS')
PE: Parity Error
FE: Frame Error
P: Parity bit stored in status byte

RFTH(1:0) **Receive FIFO Threshold** (all modes)

This bit field defines the level up to which the SCC receive FIFO is filled with valid data before an ’RPF’ interrupt is generated.
(In case of a ’frame end / block end’ condition the SEROCCO-D notifies the CPU immediately, disregarding this threshold.)
The meaning depends on the selected protocol engine:

HDLC Modes:

RFTH(1:0)	Threshold level in number of data bytes.
’00’	32 byte
’01’	16 byte
’10’	4 byte
’11’	2 byte

ASYNC/BISYNC Mode:

RFTH(1:0) Threshold level in number of data bytes (DB) and status bytes (SB) depending on bit ’RFDF’:

	RFDF = ’0’	RFDF = ’1’
’00’	1 DB	1 DB + 1 SB
’01’	4 DB	2 DB + 2 SB
’10’	16 DB	8 DB + 8 SB
’11’	32 DB	16 DB + 16 SB

Register 27 **PREAMB**
Preamble Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$1E_H$**	**$6E_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Preamble Pattern							
H	PRE(7:0)							
A	0	0	0	0	0	0	0	0
B	PRE(7:0)							

PRE(7:0) **Preamble** (hdlc/bisync modes)

This bit field determines the preamble pattern which is send out during preamble transmission.

Note: In HDLC-mode, zero-bit insertion is disabled during preamble transmission.

Register 28 **TOLEN**
Time Out Length Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$1F_H$**	**$6F_H$**

typical usage: written by CPU; read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Time Out Length							
H	0	0	0	0	0	0	0	0
A	TOIE	TOLEN(6:0)						
B	0	0	0	0	0	0	0	0

TOIE **Time Out Indication Enable** (async mode)

If this bit is set to ’1’ in ASYNC mode, any time out event will automatically generate an ’RFRD’ command thus inserting a ’block end’ indication into the RFIFO. This time-out condition is indicated with the ’TIME’ interrupt (if unmasked).

TOIE = ’0’ Automatic Time Out processing disabled.

TOIE = ’1’ Automatic Time Out processing enabled.

TOLEN(6:0) **Time Out Length** (async mode)

This bit field determines the time out period. If there is no receive line activity for the configured period of time, a time out indication is generated if enabled via bit ’TOIE’.
The period of time is programmable in multiples of character frame length <CFL> time equivalents including start, parity and stop bits (refer to **Figure 49**):

TOLEN T = ((TOLEN + 1) * 4) * <CFL>

Register 29 **ACCM0**
PPP ASYNC Control Character Map 0

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**20_H**	**70_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	ASYNC Character Control Map 07..00							
H	07	06	05	04	03	02	01	00
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 30 **ACCM1**
PPP ASYNC Control Character Map 1

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**21_H**	**71_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	ASYNC Character Control Map 0F..08							
H	0F	0E	0D	0C	0B	0A	09	08
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 31 **ACCM2**
PPP ASYNC Control Character Map2

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**22_H**	**72_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	ASYNC Character Control Map 17..10							
H	17	16	15	14	13	12	11	10
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 32 **ACCM3**
PPP ASYNC Control Character Map 3

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**23_H**	**73_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	ASYNC Character Control Map 1F..18							
H	1F	1E	1D	1C	1B	1A	19	18
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

ACCM **ASYNC Character Control Map** (hdlc modes)

This bit field is valid in HDLC asynchronous and octet-synchronous PPP mode only:

Each bit selects the corresponding character (indicated as hex value $1F_H..00_H$ in the register description table) as control character which has to be mapped into the transmit data stream.

Register 33 **UDAC0**
User Defined PPP ASYNC Control Character Map 0

CPU Accessibility: **read/write**

Reset Value: **$7E_H$**

	Channel A	Channel B
Offset Address:	**24_H**	**74_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	ASYNC Character 0							
H	AC0							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 34 **UDAC1**
User Defined PPP ASYNC Control Character Map 1

CPU Accessibility: **read/write**

Reset Value: **$7E_H$**

	Channel A	Channel B
Offset Address:	**25_H**	**75_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	ASYNC Character 1							
H	AC1							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 35 UDAC2
User Defined PPP ASYNC Control Character Map 2

CPU Accessibility: **read/write**

Reset Value: **$7E_H$**

	Channel A	Channel B
Offset Address:	**26_H**	**76_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	ASYNC Character 2							
H	AC2							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 36 UDAC3
User Defined PPP ASYNC Control Character Map 3

CPU Accessibility: **read/write**

Reset Value: **$7E_H$**

	Channel A	Channel B
Offset Address:	**27_H**	**77_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	ASYNC Character 3							
H	AC3							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

AC3..0 **User Defined ASYNC Character Control Map** (hdlc mode)

This bit field is valid in HDLC asynchronous and octet-synchronous PPP mode only:

These bit fields define user determined characters as control characters which have to be mapped into the transmit data stream.
In register ACCM only characters $00_H..1F_H$ can be selected as control characters. Register UDAC allows to specify any four characters in the range $00_H..FF_H$.
The default value is a $7E_H$ flag which must be always mapped. Thus no additional character is mapped if $7E_H$'s are programed to bit fields AC3...0 (reset value).
($7E_H$ is mapped automatically, even if not defined via a AC bit field.)

Register 37 TTSA0
Transmit Time Slot Assignment Register 0

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**28_H**	**78_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode						Tx Clock Shift		
H	0	0	0	0	0	TCS(2:0)		
A	0	0	0	0	0	TCS(2:0)		
B	0	0	0	0	0	TCS(2:0)		

Register 38 TTSA1
Transmit Time Slot Assignment Register 1

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**29_H**	**79_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Tx Time Slot Number							
H	TEPCM	TTSN(6:0)						
A	TEPCM	TTSN(6:0)						
B	TEPCM	TTSN(6:0)						

Register 39 TTSA2
Transmit Time Slot Assignment Register 2

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$2A_H$**	**$7A_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Transmit Channel Capacity							
H	TCC(7:0)							
A	TCC(7:0)							
B	TCC(7:0)							

Register 40 TTSA3
Transmit Time Slot Assignment Register 3

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$2B_H$**	**$7B_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Transmit Channel Capacity							
H	0	0	0	0	0	0	0	TCC8
A	0	0	0	0	0	0	0	TCC8
B	0	0	0	0	0	0	0	TCC8

The following register bit fields allow flexible assignment of bit- or octet-aligned transmit time-slots to the serial channel. For more detailed information refer to chapters **"Clock Mode 5a (Time Slot Mode)" on Page 57** and **"Clock Mode 5b (Octet Sync Mode)" on Page 64**.

TCS(2:0) **Transmit Clock Shift** (all modes)

This bit field determines the transmit clock shift.

TEPCM **Enable PCM Mask Transmit** (all modes)

This bit selects the additional Transmit PCM Mask (refer to register PCMTX0..PCMTX3):

TEPCM='0' Standard time-slot configuration.

TEPCM='1' The time-slot width is constant 8 bit, bit fields 'TTSN' and 'TCS' determine the offset of the PCM mask and 'TCC' is ignored. Each time-slot selected via register PCMTX0..PCMTX3 is an active transmit timeslot.

TTSN(6:0) **Transmit Time Slot Number** (all modes)

This bit field selects the start position of the timeslot in time-slot configuration mode (clock mode 5a/5b):

Offset = 1+TTSN*8 + TCS (1..1024 clocks)

TCC(8:0) **Transmit Channel Capacity** (all modes)

This bit field determines the transmit time-slot width in standard time-slot configuration (bit TEPCM='0'):
Number of bits = TCC + 1, (1..512 bits/time-slot)

Register 41 **RTSA0**
Receive Time Slot Assignment Register 0

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$2C_H$**	**$7C_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode						Rx Clock Shift		
H	0	0	0	0	0	RCS(2:0)		
A	0	0	0	0	0	RCS(2:0)		
B	0	0	0	0	0	RCS(2:0)		

Register 42 **RTSA1**
Receive Time Slot Assignment Register 1

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$2D_H$**	**$7D_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Rx Time Slot Number							
H	REPCM	RTSN(6:0)						
A	REPCM	RTSN(6:0)						
B	REPCM	RTSN(6:0)						

Register 43 **RTSA2**
Receive Time Slot Assignment Register 2

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$2E_H$**	**$7E_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Channel Capacity							
H	RCC(7:0)							
A	RCC(7:0)							
B	RCC(7:0)							

Register 44 **RTSA3**
Receive Time Slot Assignment Register 3

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$2F_H$**	**$7F_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Channel Capacity							
H	0	0	0	0	0	0	0	RCC8
A	0	0	0	0	0	0	0	RCC8
B	0	0	0	0	0	0	0	RCC8

The following register bit fields allow flexible assignment of bit- or octet-aligned receive time-slots to the serial channel. For more detailed information refer to chapters **"Clock Mode 5a (Time Slot Mode)" on Page 57** and **"Clock Mode 5b (Octet Sync Mode)" on Page 64**.

RCS(2:0) **Receive Clock Shift** (all modes)

This bit field determines the receive clock shift.

REPCM **Enable PCM Mask Receive** (all modes)

This bit selects the additional Receive PCM Mask (refer to register PCMRX0..PCMRX3):

REPCM='0' Standard time-slot configuration.

REPCM='1' The time-slot width is constant 8 bit, bit fields 'RTSN' and 'RCS' determine the offset of the PCM mask and 'RCC' is ignored. Each time-slot selected via register PCMRX0..PCMRX3 is an active receive timeslot.

RTSN(6:0) **Receive Time Slot Number** (all modes)

This bit field selects the start position of the timeslot in time-slot configuration mode (clock mode 5a/5b):

Offset = 1+RTSN*8 + RCS (1..1024 clocks)

RCC(8:0) **Receive Channel Capacity** (all modes)

This bit field determines the receive time-slot width in standard time-slot configuration (bit REPCM='0'):
Number of bits = RCC + 1, (1..512 bits/time-slot)

Register 45 **PCMTX0**
PCM Mask Transmit Direction Register 0

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**30_H**	**80_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	PCM Mask for Transmit Direction							
H	T07	T06	T05	T04	T03	T02	T01	T00
A	T07	T06	T05	T04	T03	T02	T01	T00
B	T07	T06	T05	T04	T03	T02	T01	T00

Register 46 **PCMTX1**
PCM Mask Transmit Direction Register 1

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**31_H**	**81_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	PCM Mask for Transmit Direction							
H	T15	T14	T13	T12	T11	T10	T09	T08
A	T15	T14	T13	T12	T11	T10	T09	T08
B	T15	T14	T13	T12	T11	T10	T09	T08

Register 47 **PCMTX2**
PCM Mask Transmit Direction Register 2

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**32_H**	**82_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	PCM Mask for Transmit Direction							
H	T23	T22	T21	T20	T19	T18	T17	T16
A	T23	T22	T21	T20	T19	T18	T17	T16
B	T23	T22	T21	T20	T19	T18	T17	T16

Register 48 **PCMTX3**
PCM Mask Transmit Direction Register 3

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**33_H**	**83_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	15	14	13	12	11	10	9	8
Mode	PCM Mask for Transmit Direction							
H	T31	T30	T29	T28	T27	T26	T25	T24
A	T31	T30	T29	T28	T27	T26	T25	T24
B	T31	T30	T29	T28	T27	T26	T25	T24

PCMTX **PCM Mask for Transmit Direction** (all mode)

This bit field is valid in clock mode 5 only and the PCM mask must be enabled via bit 'TEPCM' in register TTSA1.

Each bit selects one of 32 (8-bit) transmit time-slots. The offset of time-slot zero to the frame sync pulse can be programmed via register TTSA1 bit field 'TTSN'.

Register 49 **PCMRX0**
PCM Mask Receive Direction Register 0

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**34_H**	**84_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	PCM Mask for Receive Direction							
H	R07	R06	R05	R04	R03	R02	R01	R00
A	R07	R06	R05	R04	R03	R02	R01	R00
B	R07	R06	R05	R04	R03	R02	R01	R00

Register 50 **PCMRX1**
PCM Mask Receive Direction Register 1

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**35_H**	**85_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	15	14	13	12	11	10	9	8
Mode	PCM Mask for Receive Direction							
H	R15	R14	R13	R12	R11	R10	R09	R08
A	R15	R14	R13	R12	R11	R10	R09	R08
B	R15	R14	R13	R12	R11	R10	R09	R08

Register 51 **PCMRX2**
PCM Mask Receive Direction Register 2

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**36_H**	**86_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	PCM Mask for Receive Direction							
H	R23	R22	R21	R20	R19	R18	R17	R16
A	R23	R22	R21	R20	R19	R18	R17	R16
B	R23	R22	R21	R20	R19	R18	R17	R16

Register 52 **PCMRX3**
PCM Mask Receive Direction Register 3

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**37_H**	**87_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	15	14	13	12	11	10	9	8
Mode	PCM Mask for Receive Direction							
H	R31	R30	R29	R28	R27	R26	R25	R24
A	R31	R30	R29	R28	R27	R26	R25	R24
B	R31	R30	R29	R28	R27	R26	R25	R24

PCMRX **PCM Mask for Receive Direction** (all mode)

This bit field is valid in clock mode 5 only and the PCM mask must be enabled via bit 'REPCM' in register RTSA1.

Each bit selects one of 32 (8-bit) receive time-slots. The offset of time-slot zero to the frame sync pulse can be programmed via register RTSA1 bit field 'RTSN'.

Register 53 **BRRL**
Baud Rate Register (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**38_H**	**88_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Baud Rate Generator Factor N							
H	0	0	BRN(5:0)					
A	0	0	BRN(5:0)					
B	0	0	BRN(5:0)					

Register 54 **BRRH**
Baud Rate Register (High Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**39_H**	**89_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Baud Rate Generator Factor M							
H	0	0	0	0	BRM(3:0)			
A	0	0	0	0	BRM(3:0)			
B	0	0	0	0	BRM(3:0)			

BRM(3:0) **Baud Rate Factor 'M'** (all modes)

BRN(5:0) **Baud Rate Factor 'N'** (all modes)

These bit fields determine the division factor of the internal baud rate generator. The baud rate generator input clock and the usage of baud rate generator output depends on the selected clock mode.
The division factor k is calculated by:

$$k = (N + 1) \times 2^M$$

with M=0..15 and N=0..63.

$$f_{BRG} = f_{in} / k$$

Register 55 **TIMR0**
Timer Register 0

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$3A_H$**	**$8A_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Timer Value							
H	TVALUE(7:0)							
A	TVALUE(7:0)							
B	TVALUE(7:0)							

Register 56 **TIMR1**
Timer Register 1

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$3B_H$**	**$8B_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Timer Value							
H	TVALUE(15:0)							
A	TVALUE(15:0)							
B	TVALUE(15:0)							

Register 57 **TIMR2**
Timer Register 2

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$3C_H$**	**$8C_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Timer Value							
H	TVALUE(23:16)							
A	TVALUE(23:16)							
B	TVALUE(23:16)							

Register 58 **TIMR3**
Timer Register 3

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$3D_H$**	**$8D_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Timer Configuration							
H	SRC	0	0	TMD	0	CNT(2:0)		
A	SRC	0	0	0	0	CNT(2:0)		
B	SRC	0	0	0	0	CNT(2:0)		

SRC **Clock Source (valid in clock mode 5 only)** (all modes)

This bit selects the clock source of the internal timer:

SRC = ’0’ The timer is clocked by the effective transmit clock.

SRC = ’1’ The timer is clocked by the frame-sync synchronization signal supplied via the FSC pin in clock mode 5.

TMD **Timer Mode** (hdlc modes)

This bit must be set to ’1’ if HDLC Automode operation is selected. In all other protocol modes it must remain ’0’:

TMD=’0’ The timer is controlled by the CPU via access to registers CMDRL and TIMR0..TIMR3.
The timer can be started any time by setting bit ’STI’ in register CMDRL. After the timer has expired it generates a timer interrupt. The timer can be stopped any time by setting bit ’TRES’ in register CMDRL to ’1’.

TMD=’1’ The timer is used by the SEROCCO-D for protocol specific time-out and retry transactions in HDLC Automode.

CNT(2:0) **Counter** (all modes)

The meaning of this bit field depends on the selected protocol mode.
In HDLC Automode, with bit TMD=’1’:

- Retry Counter (in HDLC protocol known as ’N2’):
Bit field ’CNT’ indicates the number of S-Command frames (with poll bit set) which are transmitted autonomously by SEROCCO-D after every expiration of the time out period ’t’ (determined by ’TVALUE’), in case an I-Frame gets not acknowledged by the opposite station. The maximum value is 6 S-command frames. If ’CNT’ is set to ’7’, the number of S-commands is unlimited in case of no acknowledgement.

In all other modes, with bit TMD=’0’:

- Restart Counter :
Bit field ’CNT’ indicates the number of automatic restarts which are performed by SEROCCO-D after every expiration of the time-out period ’t’, in case the timer is not stopped by setting bit ’TRES’ in register CMDRL to ’1’. The maximum value is 6 restarts. If ’CNT’ is set to ’7’, a timer interrupt is generated periodically with time period ’t’ determined by bit field ’TVALUE’.

TVALUE (23:0)	**Timer Expiration Value**	(all modes)
	This bit field determines the timer expiration period 't':	

$$t = (TVALUE + 1) \cdot CP$$

('CP' is the clock period, depending on bit 'SRC'.)

Register 59 **XAD1**
Transmit Address 1 Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$3E_H$**	**$8E_H$**

typical usage: written by CPU; read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Transmit Address (high)							
H	XAD1 (high byte)						0	XAD1_0
	or XAD1 (COMMAND)							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 60 **XAD2**
Transmit Address 2 Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$3F_H$**	**$8F_H$**

typical usage: written by CPU; read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Transmit Address (low)							
H	XAD2 (low byte)							
	or XAD2 (RESPONSE)							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

XAD1 and XAD2 bit fields are valid in HDLC modes with automatic address field handling only (Automode, Address Mode 1, Address Mode 2). They can be programmed with one individual address byte which is inserted automatically into the address field (8 or 16 bit) of a HDLC transmit frame. The function depends on the selected protocol mode and address field size (bit 'ADM' in register CCR2L).

XAD1 **Transmit Address 1** (hdlc modes)

– 2-byte address field:
  Bit field XAD1 constitutes the high byte of the 2-byte address field. Bit 1 must be set to '0'. According to the ISDN LAP-D protocol, bit 1 is interpreted as the C/R (COMMAND/RESPONSE) bit. This bit is manipulated automatically by SEROCCO-D according to the setting of bit 'CRI' in register RAH1. The following is the C/R value (on bit 1), when:
  - transmitting COMMANDs: '1' (if 'CRI'='1') ; '0' (if 'CRI'='0')
  - transmitting RESPONSEs: '0' (if 'CRI'='1') ; '1' (if 'CRI'='0')
  (In ISDN LAP-D, the high byte is known as 'SAPI'.)
  In accordance with the HDLC protocol, bit 'XAD1_0' should be set to '0', to indicate that the address field contains (at least) one more byte.

– 1-byte address field:
  According to the X.25 LAP-B protocol, XAD1 is the address of a 'COMMAND' frame.

XAD2 **Transmit Address 2** (hdlc modes)

– 2-byte address field:
  Bit field XAD2 constitutes the low byte of the 2-byte address field.
  (In ISDN LAP-D, the low byte is known as 'TEI'.)

– 1-byte address field:
  According to the X.25 LAP-B protocol, XAD2 is the address of a 'RESPONSE' frame.

Register 61 **RAL1**
Receive Address 1 Low Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**40_H**	**90_H**

typical usage: written by CPU; read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Address 1 (low)							
H	RAL1							
	RAL1							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 62 **RAH1**
Receive Address 1 High Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**41_H**	**91_H**

typical usage: written by CPU; read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Address 1 (high)							
H	RAH1						CRI	RAH1_0
	or RAH1							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 63 **RAL2**
Receive Address 2 Low Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**42_H**	**92_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Address 2 (low)							
H	RAL2							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 64 **RAH2**
Receive Address 2 High Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**43_H**	**93_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Address 2 (high)							
H	RAH2							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

In operating modes that provide address recognition, the high/low byte of the received address is compared with the individually programmable values in register RAH2/RAL2/RAH1/RAL1.
This addresses can be masked on a per bit basis by setting the corresponding bits in registers AMRAL1/AMRAH1/AMRAL2/AMRAH2 to allow extended broadcast address recognition. This feature is applicable to all HDLC sub-modes with address recognition.

RAH1 **Receive Address 1 Byte High** (hdlc modes)

In HDLC Automode bit '1' is reserved for 'CRI' (Command Response Interpretation). In all other modes RAH1 is an 8 bit address.

CRI **Command/Response Interpretation**
The setting of this bit effects the meaning of the 'C/R' bit in the receive status byte (RSTA). This status bit 'C/R' should be interpreted after reception as follows:
'0' (if 'CRI'='1') ; '1' (if 'CRI'='0') : COMMAND received
'1' (if 'CRI'='1') ; '0' (if 'CRI'='0') : RESPONSE received

Note: If 1-byte address field is selected in HDLC Automode, RAH1 must be set to $0x00_H$.

RAL1 **Receive Address 1 Byte Low** (hdlc modes)

The general function and its meaning depends on the selected HDLC operating mode:

- **Automode / Address Mode 2 (16-bit address)**
  RAL1 can be programmed with the value of the first individual low address byte.
- **Automode / Address Mode 2 (8-bit address)**
  According to X.25 LAP-B protocol, the address in RAL1 is considered as the address of a 'COMMAND' frame.

RAH2 **Receive Address 2 Byte High** (hdlc modes)

RAL2 **Receive Address 2 Byte Low** (hdlc modes)

Value of the second individually programmable high/low address byte. If a 1-byte address field is selected, RAL2 is considered as the address of a 'RESPONSE' frame according to X.25 LAP-B protocol.

Register 65 **AMRAL1**
Mask Receive Address 1 Low Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**44_H**	**94_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Mask Address 1 (low)							
H	AMRAL1							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 66 **AMRAH1**
Mask Receive Address 1 High Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**45_H**	**95_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Mask Address 1 (high)							
H	AMRAH1							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 67 **AMRAL2**
Mask Receive Address 2 Low Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**46_H**	**96_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Mask Address 2 (low)							
H	AMRAL2							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 68 **AMRAH2**
Mask Receive Address 2 High Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**47_H**	**97_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Mask Address 2 (high)							
H	AMRAH2							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

AMRAH2	**Receive Mask Address 2 Byte High**	(hdlc modes)
AMRAL2	**Receive Mask Address 2 Byte Low**	(hdlc modes)
AMRAH1	**Receive Mask Address 1 Byte High**	(hdlc modes)
AMRAL1	**Receive Mask Address 1 Byte Low**	(hdlc modes)

Setting a bit in this registers to '1' masks the corresponding bit in registers RAH2/RAL2/RAH1/RAL1. A masked bit position always matches when comparing the received frame address with registers RAH2/RAL2/RAH1/RAL1, allowing extended broadcast mechanism.

bit = '0' The dedicated bit position is NOT masked. This bit position in the received address must match with the corresponding bit position in registers RAH2/RAL2/RAH1/RAL1 to accept the frame.

bit = '1' The dedicated bit position is masked. This bit position in the received address NEED NOT match with the corresponding bit position in registers RAH2/RAL2/RAH1/RAL1 to accept the frame.

Register 69 **RLCRL**
Receive Length Check Register (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**48_H**	**98_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Length Limit							
H	RL(7:0)							
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

Register 70 **RLCRH**
Receive Length Check Register (High Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**49_H**	**99_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Receive Length Check Control				Receive Length Limit			
H	RCE	0	0	0	0	RL(10:8)		
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0

RCE **Receive Length Check Enable** (hdlc modes)

This bit is valid in HDLC mode only and enables/disables the receive length check function:

RCE = ’0’ No receive length check on received HDLC frames is performed.

RCE = ’1’ The receive length check is enabled. All bytes of a HDLC frame which are transferred to the receive FIFO (depending on the selected protocol sub-mode and receive CRC handling) are counted and checked against the maximum length check limit which is programmed in bit field ’RL’.
A frame exceeding the maximum length is treated as if it were aborted on the receive line (’RME’ interrupt and bit ’RAB’ (receive abort) set in the RSTA byte).
In addition a ’FLEX’ interrupt is generated prior to ’RME’, if enabled.

Note: The Receive Status Byte (RSTA) is part of the frame length checking.

RL(10:0) **Receive Length Check Limit** (hdlc modes)

This bit-field defines the receive length check limit (32..65536 bytes) if checking is enabled via bit ’RCE’:

RL(10:0) The receive length limit is calculated by:

$$\text{Limit} = (RL + 1) \cdot 32$$

Register 71 **XON**
XON In-Band Flow Control Character Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$4A_H$**	**$9A_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	XON Character							
H	0	0	0	0	0	0	0	0
A	XON(7:0)							
B	0	0	0	0	0	0	0	0

Register 72 **XOFF**
XOFF In-Band Flow Control Character Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$4B_H$**	**$9B_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	XOFF Character							
H	0	0	0	0	0	0	0	0
A	XOFF(7:0)							
B	0	0	0	0	0	0	0	0

XON(7:0) **XON Character** (async mode)

This bit field specifies the XON character for in-band flow control in ASYNC protocol mode. The number of significant bits starting with the LSB depends on the character length (5..8 bits) selected via bit field ’CHL(1:0)’ in register CCR3L.
A received character is recognized as a valid XON-character, if

- the character was correctly framed (character length as programmed and correct parity if checking is enabled)
- each bit position of the received character which is not masked via register MXON matches with the corresponding bit in register XON.

Received characters recognized as XON character are stored in the receive FIFO as normal receive data unless disabled with bit CCR3L:DXS. An appropriate ’XON’ interrupt is generated (if enabled) and the transmitter is switched into ’XON’ state if in-band flow control is enabled via bit ’FLON’ in register CCR2H.

XOFF(7:0) **XOFF Character** (async mode)

This bit field specifies the XOFF character for in-band flow control in ASYNC protocol mode. The number of significant bits starting with the LSB depends on the character length (5..8 bits) selected via bit field ’CHL(1:0)’ in register CCR3L.
A received character is recognized as a valid XOFF-character, if

- the character was correctly framed (character length as programmed and correct parity if checking is enabled)
- each bit position of the received character which is not masked via register MXOFF matches with the corresponding bit in register XOFF.

Received characters recognized as XOFF character are stored in the receive FIFO as normal receive data unless disabled with bit CCR3L:DXS. An appropriate ’XOFF’ interrupt is generated (if enabled) and the transmitter is switched into ’XOFF’ state if in-band flow control is enabled via bit ’FLON’ in register CCR2H.

Register 73 **MXON**
XON In-Band Flow Control Mask Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$4C_H$**	**$9C_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	XON Character Mask							
H	0	0	0	0	0	0	0	0
A	MXON(7:0)							
B	0	0	0	0	0	0	0	0

Register 74 **MXOFF**
XOFF In-Band Flow Control Mask Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$4D_H$**	**$9D_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	XOFF Character Mask							
H	0	0	0	0	0	0	0	0
A	MXOFF(7:0)							
B	0	0	0	0	0	0	0	0

MXON(7:0) **XON Character Mask** (async mode)

Setting a bit in this bit field to ’1’ masks the corresponding bit in bit field ’XON(7:0)’ of register XON. A masked bit position always matches when comparing the received character with bit field ’XON(7:0)’.

bit = ’0’	The dedicated bit position is NOT masked. This bit position in the received character must match with the corresponding bit position in bit field ’XON’ to recognize the received character as an XON character.
bit = ’1’	The dedicated bit position is masked. This bit position in the received character NEED NOT match with the corresponding bit position in bit field ’XON’ to recognize the received character as an XON character.

MXOFF(7:0) **XOFF Character Mask** (async mode)

Setting a bit in this bit field to ’1’ masks the corresponding bit in bit field ’XOFF(7:0)’ of register XOFF. A masked bit position always matches when comparing the received character with bit field ’XOFF(7:0)’.

bit = ’0’	The dedicated bit position is NOT masked. This bit position in the received character must match with the corresponding bit position in bit field ’XOFF’ to recognize the received character as an XOFF character.
bit = ’1’	The dedicated bit position is masked. This bit position in the received character NEED NOT match with the corresponding bit position in bit field ’XOFF’ to recognize the received character as an XOFF character.

Register 75 **TCR**
Termination Character Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$4E_{H8}$**	**$9E_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Termination Character							
H	0	0	0	0	0	0	0	0
A	TC(7:0)							
B	TC(7:0)							

TC(7:0) **Termination Character** (async mode)

This bit-field defines the termination character which is monitored on the receive data stream if enabled via bit 'TCDE' in register CCR3L.

Register 76 **TICR**
Transmit Immediate Character Register

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$4F_H$**	**$9F_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	Transmit Immediate Character							
H	0	0	0	0	0	0	0	0
A	TIC(7:0)							
B	0	0	0	0	0	0	0	0

TIC **Transmit Immediate Character** (async mode)

On write access to this register, the ASYNC protocol engine will automatically insert the character defined by bit field ’TIC’ into the transmit data stream.
This happens

- immediately after write access to register TICR if the transmitter is in IDLE state (no other character is currently transmitted). The transmitter returns to IDLE state after transmission of the TIC.
- immediately after the character which is currently in transmission is completed. After transmission of the TIC, the transmitter continues with transmission of characters which are still stored in the transmit FIFO. Thus the TIC is inserted into the data stream between the characters provided via the transmit FIFO.

The TIC transmission is independent of in-band flow control. Thus the TIC is sent out even if the transmitter is in ’XOFF’ state. However the transmitter must be enabled via signal $\overline{\text{CTS}}$ (depending on bit ’FCTS’ in register CCR1H).

The number of significant bits (starting with the LSB) depends on the character length programmed in bit field ’CHL(1:0)’ in register CCR3L. All character framing related settings in registers CCR3L/CCR3H (start bit, parity generation, number of stop bits) also apply to the TIC character framing.

As long as the TIC character is not completely sent, status bit TIC Execution (’TEC’) in status register STARL is set to ’1’ by SEROCCO-D. No further write access to register TICR is allowed until ’TEC’ status indication is cleared by SEROCCO-D.

Register 77 **ISR0**
Interrupt Status Register 0

CPU Accessibility: **read only**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**50_H**	**$A0_H$**

typical usage: updated by SEROCCO-D
read and evaluated by CPU

Bit	7	6	5	4	3	2	1	0
Mode	ISR0							
H	RDO	RFO	PCE	RSC	RPF	RME	RFS	FLEX
A	0	RFO	FERR	PERR	RPF	TCD	TIME	0
B	0	RFO	SCD	PERR	RPF	TCD	0	0

Register 78 **ISR1**
Interrupt Status Register 1

CPU Accessibility: **read only**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**51_H**	**$A1_H$**

typical usage: updated by SEROCCO-D
read and evaluated by CPU

Bit	7	6	5	4	3	2	1	0
Mode	ISR1							
H	TIN	CSC	XMR	XPR	ALLS	XDU	SUEX	0
A	TIN	CSC	XOFF	XPR	ALLS	XON	BRK	BRKT
B	TIN	CSC	XMR	XPR	ALLS	XDU	0	0

Register 79 **ISR2**
Interrupt Status Register 2

CPU Accessibility: **read only**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**52_H**	**$A2_H$**

typical usage: updated by SEROCCO-D
read and evaluated by CPU

Bit	7	6	5	4	3	2	1	0
Mode	ISR2							
H	0	0	0	0	0	0	PLLA	CDSC
A	0	0	0	0	0	0	PLLA	CDSC
B	0	0	0	0	0	0	PLLA	CDSC

RDO **Receive Data Overflow Interrupt** (hdlc mode)

This bit is set to ’1’, if receive data of the current frame got lost because of a SCC receive FIFO full condition. However the rest of the frame is received and discarded as long as the receive FIFO remains full and is stored as soon as FIFO space is available again. The receive status byte (RSTA) of such a frame contains an ’RDO’ indication. In DMA operation the ’RDO’ indication is also set in the receive byte count register RBCH.

RFO **Receive FIFO Overflow Interrupt** (all modes)

HDLC Mode:

This bit is set to ’1’, if the SCC receive FIFO is full and a complete frame must be discarded. This interrupt can be used for statistical purposes, indicating that the host was not able to service the SCC receive FIFO quickly enough, e.g. due to high bus latency.

ASYNC/BISYNC Mode:

This bit is set to ’1’, if the SCC receive FIFO is full and another received character has been discarded. This interrupt can be used for statistical purposes and might indicate that the host was not able to service the SCC receive FIFO quickly enough, e.g. bus latencies are too high.

PCE **Protocol Error Interrupt** (hdlc mode)

This bit is valid in HDLC Automode only.

It is set to ’1’, if the receiver has detected a protocol error, i.e. one of the following events occured:

- an S- or I-frame was received with wrong N(R) counter value;
- an S-frame containing an Information field was received.

FERR **Framing Error Interrupt** (async mode)

This bit is set to ’1’, if a character framing error is detected, i.e. a ’0’ was sampled at a position where a stop bit ’1’ was expected due to the selected character format.

SCD **Sync Character Detected** (bisync mode)

Only valid in Hunt Mode.
This bit is set to ’1’ if a SYN character is found in the received data stream after the ’HUNT’ command has been issued in register CMDRH. The receiver now is in the synchronous state.

RSC **Receive Status Change Interrupt** (hdlc mode)

This bit is valid in HDLC Automode only.
It is set to ’1’, if a status change of the remote station receiver has been detected by receiving a S-frame with receiver ready (RR) or receiver not ready (RNR) indication. Because only a status change is indicated via this interrupt, the current status can be evaluated by reading bit ’RRNR’ in status register STARH.

PERR **Parity Error Interrupt** (async/bisync modes)

This bit is only valid if parity checking/generation is enabled via bit ’PARE’ in register CCR3H.
It is set to ’1’, if a character with wrong parity has been received. If enabled via bit CCR3H:RFDF, this error status is additionally stored in the receive status byte generated for each receive character.

RPF **Receive Pool Full Interrupt** (all modes)

This bit is set to ’1’ if the RFIFO threshold level, set with bit field ’RFTH(1:0)’ in register CCR3H, is reached. Default threshold level is 32 data bytes in HDLC/PPP modes, 1 data byte in ASYNC/BISYNC modes.

RME **Receive Message End Interrupt** (hdlc mode)

This bit set to '1' indicates that the reception of one message is completed, i.e. either

- one message which fits into RFIFO not exceeding the receive FIFO threshold, or
- the last part of a message, all in all exceeding the receive FIFO threshold

is stored in the RFIFO.

The complete message length can be determined by reading the RBCL/ RBCH registers. The number of bytes stored in RFIFO is given by the 5, 4, 2 or 1 least significant bits of register RBCL, depending on the selected RFIFO threshold (bit field 'RFTH(1:0)' in register CCR3H). Additional frame status information is available in the RSTA byte, stored in the RFIFO as the last byte of each frame.

Note: After the RFIFO contents have been read, an CMDRH:RMC command must be issued to free the RFIFO for new receive data.

TCD **Termination Character Detected Interrupt** (async/bisync mode)

This bit is set to '1', if a termination character (TCR) has been detected in the receive data stream or an 'RFRD' command, issued in the CMDRH register, has been completed. The SCC will insert a 'block end' indication to the RFIFO. The actual block length can be determined by reading register RBCL.

Note: After the RFIFO contents have been read, an CMDRH:RMC command must be issued to free the RFIFO for new receive data.

RFS **Receive Frame Start Interrupt** (hdlc mode)

This bit is set to '1', if the beginning of a valid frame is detected by the receiver. A valid frame start is detected either if a valid address field is recognized (in all operating modes with address recognition) or if a start flag is recognized (in all operating modes with no address recognition).

TIME **Time Out Interrupt** (async mode)

This bit is set to '1', if the time out limit is exceeded, i.e. no new character was received in a programmable period of time (refer to register TOLEN bit fields 'TOIE' and 'TOLEN' for more information).

FLEX **Frame Length Exceeded Interrupt** (hdlc mode)

This bit is set to ’1’, if the frame length check feature is enabled and the current received frame is aborted because the programmed frame length limit was exceeded (refer to registers RLCRL/RLCRH for detailed description).

TIN **Timer Interrupt** (all modes)

This bit is set to ’1’, if the internal timer was activated and has expired (refer also to description of timer registers TIMR0..TIMR3).

CSC **$\overline{\text{CTS}}$ Status Change** (all modes)

This bit is set to ’1’, if a transition occurs on signal $\overline{\text{CTS}}$. The current state of signal $\overline{\text{CTS}}$ is monitored by status bit ’CTS’ in status register STARL.

Note: A transmit clock must be provided to detect a transition of $\overline{\text{CTS}}$.

XMR **Transmit Message Repeat** (hdlc/bisync modes)

This bit is set to ’1’, if transmission of the last frame has to be repeated (by software), because

- the SCC has received a negative acknowledge to an I-frame (in HDLC Automode operation);
- a collision occured after at least 14.5 bytes of data have been completely sent out, i.e. automatic re-transmission cannot be performed by the SCC;
- $\overline{\text{CTS}}$ signal was deasserted after at least 14.5bytes of data have been completely sent out.

Note: For easy recovery from a collision event (in bus configuration only), the SCC transmit FIFO should not contain more than one complete frame. This can be achieved by using the ’ALLS’ interrupt to control the corresponding transmit channel forwarding a new frame on all sent (ALLS) event only.

XOFF **XOFF Character Detected Interrupt** (async mode)

ASYNC Mode:

This bit is set to ’1’, if the currently received character matched the XOFF character programmed in register XOFF and indicates, that the transmitter is switched to ’XOFF’ state if in-band flow control is enabled via bit ’FLON’ in register CCR2H.

XPR **Transmit Pool Ready Interrupt** (all modes)

This bit is set to ’1’, if a transmitter reset command was executed successfully (command bit ’XRES’ in register CMDRL) and whenever the XFIFO is able to accept new transmit data again.

An ’XPR’ interrupt is not generated, if no sufficient transmit clock is available (depending on the selected clock mode).

ALLS **ALL Sent Interrupt** (all modes)

HDLC Mode:

This bit is set to ’1’:

- if the last bit of the current HDLC frame is sent out via pin TxD and no further frame is stored in the SCC transmit FIFO, i.e. the transmit FIFO is empty (Address Mode 2/1/0);
- if an I-frame is sent out completely via pin TxD and either a valid acknowledge S-frame has been received or a time-out condition occured because no valid acknowledge S-frame has been received in time (Automode).

ASYNC/BISYNC Mode:

This bit is set to ’1’, if the last character is completely sent via pin TxD and no further data is stored in the SCC transmit FIFO, i.e. the transmit FIFO is empty.

XDU **Transmit Data Underrun Interrupt** (hdlc/bisync modes)

This bit is set to ’1’, if the current frame was terminated by the SCC with an abort sequence, because neither a ’frame end / block end’ indication was detected in the FIFO (to complete the current frame) nor more data is available in the SCC transmit FIFO.

Note: The transmitter is stopped if this condition occurs. The XDU condition MUST be cleared by reading register ISR1, thus bit ’XDU’ should not be masked via register IMR1.

XON **XON Character Detected Interrupt** (async mode)

This bit is set to ’1’, if the currently received character matched the XON character programmed in register XON and indicates, that the transmitter is switched to ’XON’ state if in-band flow control is enabled via bit ’FLON’ in register CCR2H.

SUEX **Signalling Unit Counter Exceeded Interrupt** (hdlc mode)

This bit is set to '1', if 256 correct or incorrect SU's have been received and the internal counter is reset to 0.

BRK **Break Interrupt** (async mode)

This bit is set to '1', if a break condition was detected on the receive line, i.e. a low level for a time equal to (character length + parity bit + stop bit(s)) bits depending on the selected ASYNC character format.

BRKT **Break Terminated Interrupt** (async mode)

This bit is set to '1', if a previously detected break condition on the receive line is terminated by a low to high transition.

PLLA **DPLL Asynchronous Interrupt** (all modes)

This bit is only valid, if the receive clock is derived from the internal DPLL and FM0, FM1 or Manchester data encoding is selected (depending on the selected clock mode and data encoding mode). It is set to '1' if the DPLL has lost synchronization. Reception is disabled until synchronization has been regained again. If the transmitter is supplied with a clock derived from the DPLL, transmission is also interrupted.

CDSC **Carrier Detect Status Change Interrupt** (all modes)

This bit is set to '1', if a state transition has been detected at signal CD. Because only a state transition is indicated via this interrupt, the current status can be evaluated by reading bit 'CD' in status register STARH.

Note: A receive clock must be provided to detect a transition of CD.

Register 80 **IMR0**
Interrupt Mask Register 0

CPU Accessibility: **read/write**

Reset Value: **FF_H**

	Channel A	Channel B
Offset Address:	**54_H**	**$A4_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	IMR0							
H	RDO	RFO	PCE	RSC	RPF	RME	RFS	FLEX
A	1	RFO	FERR	PERR	RPF	TCD	TIME	1
B	1	RFO	SCD	PERR	RPF	TCD	1	1

Register 81 **IMR1**
Interrupt Mask Register 1

CPU Accessibility: **read/write**

Reset Value: **FF_H**

	Channel A	Channel B
Offset Address:	**55_H**	**$A5_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	IMR1							
H	TIN	CSC	XMR	XPR	ALLS	XDU	SUEX	1
A	TIN	CSC	XOFF	XPR	ALLS	XON	BRK	BRKT
B	TIN	CSC	XMR	XPR	ALLS	XDU	1	1

Register 82 **IMR2**
Interrupt Mask Register 2

CPU Accessibility: **read/write**

Reset Value: **03_H**

	Channel A	Channel B
Offset Address:	**56_H**	**$A6_H$**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	IMR2							
H	0	0	0	0	0	0	PLLA	CDSC
A	0	0	0	0	0	0	PLLA	CDSC
B	0	0	0	0	0	0	PLLA	CDSC

(IM) **Interrupt Mask Bits**

Each SCC interrupt event can generate an interrupt signal indication via pin INT/$\overline{\text{INT}}$. Each bit position of registers IMR0..IMR2 is a mask for the corresponding interrupt event in the interrupt status registers ISR0..ISR2. Masked interrupt events never generate an interrupt indication via pin INT/$\overline{\text{INT}}$.

bit = '0' The corresponding interrupt event is NOT masked and will generate an interrupt indication via pin INT/$\overline{\text{INT}}$.

bit = '1' The corresponding interrupt event is masked and will NEITHER generate an interrupt vector NOR an interrupt indication via pin INT/$\overline{\text{INT}}$.

Moreover, masked interrupt events are:

- not displayed in the interrupt status registers ISR0..ISR2 if bit 'VIS' in register CCR0L is programmed to '0'.
- displayed in interrupt status registers ISR0..ISR2 if bit 'VIS' in register CCR0L is programmed to '1'.

Note: After RESET, all interrupt events are masked. Undefined bits must not be cleared to '0'.

For detailed interrupt event description refer to the corresponding bit position in registers ISR0..ISR2.

Register 83 **RSTA**
Receive Status Byte

CPU Accessibility: **read only**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**58_H**	**$A8_H$**

typical usage: written by SEROCCO-D to RFIFO;
read from RFIFO and evaluated by CPU

Bit	7	6	5	4	3	2	1	0
Mode	Receive Status Byte							
H	VFR	RDO	CRCOK	RAB	HA(1:0)/ SU(1:0)		C/R	LA
A	PE	FE	0	0	0	0	0	P
B	PE	0	0	0	0	0	0	P

The Receive Status Byte ’RSTA’ contains comprehensive status information about the last received frame (HDLC/PPP) or the last received ASYNC/BISYNC character.

The SCC attaches this status byte to the receive data and thus it should be read from the RFIFO.

In HDLC/PPP modes the RSTA value can optionally be read from this register address; in ASYNC and BISYNC modes a read to this register is not specified. In extended transparent mode this status field does not apply.

VFR **Valid Frame** (hdlc modes)

Determines whether a valid frame has been received.

VFR='0' The received frame is invalid.
An invalid frame is either a frame which is not an integer number of 8 bits (n * 8 bits) in length (e.g. 25 bits), or a frame which is too short, taking into account the operation mode selected via CCR2L (MDS1, MDS0, ADM) and the selected CRC algorithm (CCR1L:C32) as follows:

for CCR3H:DRCRC = '0' (CRC reception enabled):

- automode / address mode 2 (16-bit address)
  4 bytes (CRC-CCITT) or 6 (CRC-32)
- automode / address mode 2 (8-bit address)
  3 bytes (CRC-CCITT) or 5 (CRC-32)
- address mode 1:
  3 bytes (CRC-CCITT) or 5 (CRC-32)
- address mode 0:
  2 bytes (CRC-CCITT) or 4 (CRC-32)

for CCR3H:DRCRC = '1' (CRC reception disabled):

- automode / address mode 2 (16-bit address):
  2 bytes
- automode / address mode 2 (8-bit address):
  1 byte
- address mode 1:
  1 byte
- address mode 0:
  1 byte

Note: Shorter frames are not reported at all.

VFR='1' The received frame is valid.

RDO **Receive Data Overflow** (hdlc modes)

RDO='0' No receive data overflow has occurred.

RDO='1' A data overflow has occurred during reception of the frame. Additionally, an interrupt can be generated (refer to ISR0:RDO/IMR0:RDO).

CRCOK **CRC Compare/Check** (hdlc modes)

CRCOK='0' CRC check failed, received frame contains errors.

CRCOK='1' CRC check OK; the received frame does not contain CRC errors.

RAB **Receive Message Aborted** (hdlc modes)

RAB='0' No abort condition was detected during reception of the frame.

RAB='1' The received frame was aborted from the transmitting station. According to the HDLC protocol, this frame must be discarded by the receiver station.
This bit is also set to '1' if the maximum receive byte count (set in registers RLCRL/RLCRH) is reached.

HA(1:0) **High Byte Address Compare** (hdlc modes)

Significant only if an address mode with automatic address handling has been selected. In operating modes which provide high byte address recognition, SEROCCO-D compares the high byte of a 2-byte address with the contents of two individually programmable addresses (RAH1, RAH2) and the fixed values FE_H and FC_H (broadcast address).
Dependent on the result of this comparison, the following bit combinations are possible:

HA(1:0)='10' RAH1 has been recognized.

HA(1:0)='00' RAH2 has been recognized.

HA(1:0)='01' broadcast address has been recognized.

If RAH1 and RAH2 contain identical values, a match is indicated by HA(1:0)='10'.

SU(1:0) **SS7 Signaling Unit Type** (hdlc modes)

If Signaling System #7 support is activated (see CCR3L register, bit 'ESS7'), the bit functions are defined as follows:

SU(1:0)='00' not valid

SU(1:0)='01' Fill In Signaling Unit (FISU) detected

SU(1:0)='10' Link Status Signaling Unit (LSSU) detected

SU(1:0)='11' Message Signaling Unit (MSU) detected

C/R **Command/Response** (hdlc modes)

Significant only if 2-byte address mode has been selected.
Value of the C/R bit (bit 1 of high address byte) in the received frame.
The interpretation depends on the setting of the 'CRI' bit in the RAH1 register (See "RAH1" on page 207.).

LA **Low Byte Address Compare** (hdlc modes)

Significant in automode and address mode 2 only.
The low byte address of a 2-byte address field, or the single address byte of a 1-byte address field is compared with two addresses (RAL1, RAL2).

LA='0' RAL2 has been recognized.

LA='1' RAL1 has been recognized.

According to the X.25 LAPB protocol, RAL1 is interpreted as the address of a COMMAND frame and RAL2 is interpreted as the address of a RESPONSE frame.

P **Parity** (async/bisync mode)

'1' This bit carries the parity bit of the last received character.

FE **Framing Error** (async mode)

'1' A character framing error was detected, i.e. a '0' was sampled at a bit position where a stop bit '1' was expected due to the selected character format.

PE **Parity Error** (async/bisync mode)

'1' The calculated parity did not match the received parity bit. Optionally the interrupt PERR can be generated.

Register 84 **SYNCL**
SYN Character Register (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$5A_H$**	**AA_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	SYN Character Low							
H	0	0	0	0	0	0	0	0
A	0	0	0	0	0	0	0	0
B	SYNCL(7:0)							

Register 85 **SYNCH**
SYN Character Register (High Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$5B_H$**	**AB_H**

typical usage: written by CPU;
read and evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
Mode	SYN Character High							
H	0	0	0	0	0	0	0	0
A	0	0	0	0	0	0	0	0
B	SYNCH(7:0)							

SYNCH(7:0) Synchronization Character (high) (bisync mode)

SYNCL(7:0) Synchronization Character (low) (bisync mode)

This register is only valid in BISYNC protocol mode.
The synchronization (SYN) character format depends on the setting of bit 'BISNC' and 'SLEN' in register CCR2L:

- MONOSYNC Mode (CCR2L.BISNC = '0')
  The SYN character is defined by register 'SYNCL':
  a) SLEN = '0': the 6 bit SYN character is specified by bits (5..0)
  b) SLEN = '1': the 8 bit SYN character is specified by bits (7..0).
- BISYNC Mode (CCR2L.BISNC = '1')
  The SYN character is defined by registers 'SYNCL' and 'SYNCH':
  a) SLEN = '0': the 12 bit SYN character is specified by bits (5..0) of each register, i.e. SYN(11..0) = SYNCH(5:0), SYNCL(5:0)
  b) SLEN = '1': the 16 bit SYN character is specified by bits (7..0) of each register, i.e. SYN(15..0) = SYNCH(7:0), SYNCL(7:0).

In transmit direction the SYN character is sent continuously if no data has to be transmitted and interframe timefill control is enabled by setting bit 'ITF' to '1' in register CCR2H.

In receive direction the receiver monitors the data stream for occurence of the specified SYN pattern if operating in 'HUNT' mode (bit 'HUNT' in register CMDRH).

5.2.3 Channel Specific DMA Registers

Each register description is organized in three parts:

- a head with general information about reset value, access type (read/write), channel specific offset address and usual handling;
- a table containing the bit information (name of bit positions);
- a section containing the detailed description of each bit.

Register 86 TBADDR1L
Primary Transmit Base Address (Low Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$B0_H$**	**CA_H**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	7	6	5	4	3	2	1	0
	TBADDR1(7:0)							

Register 87 TBADDR1M
Primary Transmit Base Address (Mid Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$B1_H$**	**CB_H**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	15	14	13	12	11	10	9	8
	TBADDR1(15:8)							

Register 88 **TBADDR1H**
Primary Transmit Base Address (High Byte)

CPU Accessibility: **read/write**
Reset Value: **00_H**
Offset Address: **$B2_H$** **CC_H**
typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	TBADDR1(23:16)							

TBADDR1 (23:0) **Primary Transmit Base Address**

Only valid in internal DMA controller modes.
This bit field determines the base address of the primary DMA transmit buffer (buffer 1).

- If single-buffer operation is selected, this base address is the only one used; the secondary base address TBADDR2(23:0) is "don't care" in this case.
- If switched-buffer operation is selected (refer to register DMODE), transmission takes place based on two transmit buffers that are sent alternating.

Note: If 16-bit bus operation is selected, the base address must be word aligned, i.e. bit TBADDR1(0) must be set to '0'.

Register 89 **TBADDR2L**
Secondary Transmit Base Address (Low Byte)

CPU Accessibility: **read/write**
Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$B4_H$**	**CE_H**

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	TBADDR2(7:0)							

Register 90 **TBADDR2M**
Secondary Transmit Base Address (Mid Byte)

CPU Accessibility: **read/write**
Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$B5_H$**	**CF_H**

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	15	14	13	12	11	10	9	8
	TBADDR2(15:8)							

Register 91 **TBADDR2H**
Secondary Transmit Base Address (High Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$B6_H$**	**$D0_H$**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	7	6	5	4	3	2	1	0
	TBADDR2(23:16)							

TBADDR2 (23:0) **Secondary Transmit Base Address**

Only valid in switched-buffer DMA controller mode.
This bit field determines the base address of the secondary DMA transmit buffer (buffer 2) in switched-buffer operation.

- If single-buffer operation is selected, this base address is "don't care". Only address TBADDR1(23:0) is used in this case.
- If switched-buffer operation is selected (refer to register DMODE), transmission takes place based on both transmit buffers (TBADDR1 and TBADDR2). Data from these transmit buffers is sent alternating.

Note: If 16-bit bus operation is selected, the base address must be word aligned, i.e. bit TBADDR2(0) must be set to '0'.

Register 92 **XBC1L**
Primary Transmit Byte Count (Low Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$B8_H$**	**$D2_H$**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	7	6	5	4	3	2	1	0
	XBC1(7:0)							

Register 93 **XBC1H**
Primary Transmit Byte Count (High Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$B9_H$**	**$D3_H$**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	7	6	5	4	3	2	1	0
	XME	XF	XIF	0	XBC1(11:8)			

XBC1 (11:0) **Primary Transmit Byte Count**

Only valid in internal DMA controller modes.
This bit field determines the size in number of bytes of the primary transmit buffer (with base address TBADDR1(23:0)).

- If single-buffer operation is selected, the primary buffer is the only one used; the secondary transmit byte count bit field XBC2(11:0) is "don't care" in this case.
- If switched-buffer operation is selected (refer to register DMODE), transmission takes place based on two transmit buffers (primary and secondary) that are sent alternating.

XME **Transmit Message End Command**

Only valid in internal DMA controller mode.
This bit is identical to 'XME' command bit (refer to register **"CMDRL" on Page 150**).

XF **Transmit Frame Command**

Only valid in internal DMA controller mode.
This bit is identical to 'XF' command bit (refer to register **"CMDRL" on Page 150**).

XIF **Transmit I-Frame Command**

Only valid in internal DMA controller mode.
This bit is identical to 'XIF' command bit (refer to register **"CMDRL" on Page 150**).

Register 94 **XBC2L**
Secondary Transmit Byte Count (Low Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**BA_H**	**$D4_H$**

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	XBC2(7:0)							

Register 95 **XBC2H**
Secondary Transmit Byte Count (High Byte)

CPU Accessibility: **read/write**

Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**BB_H**	**$D5_H$**

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	XME	XF	XIF	0	XBC2(11:8)			

XBC2 (11:0)	**Secondary Transmit Byte Count** *Only valid in switched-buffer DMA controller mode.* This bit field determines the size in number of bytes of the secondary transmit buffer (with base address TBADDR2(23:0)). • If single-buffer operation is selected, this bit field is "don’t care". The primary transmit buffer with transmit byte count XBC1(11:0) is the only one used in this case. • If switched-buffer operation is selected (refer to register DMODE), transmission takes place based on two transmit buffers (primary and secondary) that are sent alternating.
XME	**Transmit Message End Command** *Only valid in internal DMA controller modes.* This bit is identical to ’XME’ command bit (refer to register **“CMDRL” on Page 150**).
XF	**Transmit Frame Command** *Only valid in internal DMA controller modes.* This bit is identical to ’XF’ command bit (refer to register **“CMDRL” on Page 150**).
XIF	**Transmit I-Frame Command** *Only valid in internal DMA controller modes.* This bit is identical to ’XIF’ command bit (refer to register **“CMDRL” on Page 150**).

Register 96 **RBADDR1L**
Primary Receive Base Address (Low Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**BC_H**	**$D6_H$**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	7	6	5	4	3	2	1	0
	RBADDR1(7:0)							

Register 97 **RBADDR1M**
Primary Receive Base Address 1 (Mid Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**BD_H**	**$D7_H$**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	15	14	13	12	11	10	9	8
	RBADDR1(15:8)							

Register 98 **RBADDR1H**
Primary Receive Base Address 1 (High Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**BE_H**	**$D8_H$**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	7	6	5	4	3	2	1	0
	RBADDR1(23:16)							

RBADDR1 (23:0) **Primary Receive Base Address**

Only valid in internal DMA controller modes.
This bit field determines the base address of the primary DMA receive buffer (buffer 1).

- If single-buffer operation is selected, this base address is the only one used; the secondary base address RBADDR2(23:0) is "don't care" in this case.
- If switched-buffer operation is selected (refer to register DMODE), reception takes place based on two receive buffers that are filled alternating.

Note: If 16-bit bus operation is selected, the base address must be word aligned, i.e. bit RBADDR1(0) must be set to '0'.

Register 99 **RBADDR2L**
Secondary Receive Base Address (Low Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$C0_H$**	**DA_H**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	7	6	5	4	3	2	1	0
	RBADDR2(7:0)							

Register 100 **RBADDR2M**
Secondary Receive Base Address (Mid Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$C1_H$**	**DB_H**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	15	14	13	12	11	10	9	8
	RBADDR2(15:8)							

Register 101 **RBADDR2H**
Secondary Receive Base Address2 (High Byte)

CPU Accessibility:	**read/write**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$C2_H$**	**DC_H**
typical usage:	written by CPU, evaluated by SEROCCO-D	

Bit	7	6	5	4	3	2	1	0
	RBADDRA2(23:8)							

RBADDR2 (23:0) **Secondary Receive Base Address**

Only valid in switched-buffer DMA controller mode.
This bit field determines the base address of the secondary DMA receive buffer (buffer 2) in switched-buffer operation.

- If single-buffer operation is selected, this base address is "don't care". Only address RBADDR1(23:0) is used in this case.
- If switched-buffer operation is selected (refer to register DMODE), reception takes place based on both receive buffers (RBADDR1 and RBADDR2). Data is received into these buffers alternating.

Note: If 16-bit bus operation is selected, the base address must be word aligned, i.e. bit RBADDR2(0) must be set to '0'.

Register 102 **RMBSL**
Receive Maximum Buffer Size (Low Byte)

CPU Accessibility: **read/write**
Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$C4_H$**	**DE_H**

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	7	6	5	4	3	2	1	0
	Receive Maximum Buffer Size							
	RMBS(7:0)							

Register 103 **RMBSH**
Receive Maximum Buffer Size (High Byte)

CPU Accessibility: **read/write**
Reset Value: **00_H**

	Channel A	Channel B
Offset Address:	**$C5_H$**	**DF_H**

typical usage: written by CPU, evaluated by SEROCCO-D

Bit	15	14	13	12	11	10	9	8
	Receive Maximum Buffer Size							
	RE	0	0	0	RMBS(11:8)			

RE **Receive DMA Enable**

Only valid in internal DMA controller modes.
Self-clearing command bit:

RE=’0’	The DMA controller is not set up to forward receive data into a buffer in memory.
RE=’1’	If this bit is set to ’1’, the DMA controller is activated for transferring receive data into a buffer in memory. This buffer in memory has to be set up with a valid base address RBADDRi(23:0) and the maximum buffer size RMBS(11:0) in advance.

RMBS(11:0) **Receive Maximum Buffer Size**

Only valid in internal DMA controller modes.
This bit field determines the reserved size (0..4095 byte) for a receive buffer in memory. With the base address RBADDRi(23:0), the location of the receive buffer is defined.

Register 104 **RBCL**
Receive Byte Count (Low Byte)

CPU Accessibility:	**read only**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$C6_H$**	**$E0_H$**
typical usage:	written by SEROCCO-D, evaluated by CPU	

Bit	7	6	5	4	3	2	1	0
	RBC(7:0)							

Register 105 **RBCH**
Receive Byte Count (High Byte)

CPU Accessibility:	**read only**	
Reset Value:	**00_H**	
	Channel A	Channel B
Offset Address:	**$C7_H$**	**$E1_H$**
typical usage:	written by SEROCCO-D, evaluated by CPU	

Bit	7	6	5	4	3	2	1	0
	RBCO	0	0	0	RBC(11:8)			

RBC(11:0) **Receive Byte Count**

This bit field determines the receive byte count (1..4095) of the currently received frame/block.

RBCO **Receive Byte Counter Overflow**

Only valid in DMA controller mode.
This bit indicates an overflow of the receive byte conter RBC(11:0), i.e. the receive frame length exceeded 4095 bytes.

5.2.4 Miscellaneous Registers

Register 106 VER0
Version Register 0

CPU Accessibility: **read only**
Reset Value: **83_H**
Offset Address: **EC_H**
typical usage: evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	Manufacturer Code							Fix '1'
	VER(7:0)							

Register 107 VER1
Version Register 1

CPU Accessibility: **read only**
Reset Value: **$E0_H$**
Offset Address: **ED_H**
typical usage: evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	Device Code (bits 3 .. 0)				Manufacturer Code			
	VER(15:8)							

Register 108 **VER2**
Version Register 2

CPU Accessibility: **read only**
Reset Value: **05_H**
Offset Address: **EE_H**
typical usage: evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	Device Code (bits 11 .. 4)							
	VER(23:16)							

Register 109 **VER3**
Version Register 3

CPU Accessibility: **read only**
Reset Value: **20_H**
Offset Address: **EF_H**
typical usage: evaluated by CPU

Bit	7	6	5	4	3	2	1	0
	Version Number				Device Code (bits 15 .. 12)			
	VER(31:24)							

VER(31:0)	**Version Register**	
	Identical to 32 bit boundary scan ID string.	
	The 32 bit string consists of the bit fields:	
VER(31:28)	2_H	Version Number
VER(27:12)	$005E_H$	Device Code
VER(11:0)	083_H	Manufacturer Code (LSB fixed to ’1’)

6 Programming

6.1 Initialization

After Reset the CPU has to write a minimum set of registers and an optional set depending on the required features and operating modes.

First, the following initialization steps must be taken:

- Select serial protocol mode (refer to **Table 12 "Protocol Mode Overview" on Page 86**),
- Select encoding of the serial data (refer to **Chapter 3.2.13 "Data Encoding" on Page 75**),
- Program the output characteristics of
  - pin TxD (selected with bit 'ODS' in **"Channel Configuration Register 1 (Low Byte)" on Page 159**) and
  - interrupt pin INT/$\overline{\text{INT}}$ (selected with bit field 'IPC(1:0)' in **"Global Mode Register" on Page 127**),
- Choose a clock mode (refer to **Table 7 "Overview of Clock Modes" on Page 48**).
- Power-up the oscillator unit (with or without shaper) by re-setting bit GMODE:OSCPD to '0', if appropriate (GMODE:DSHP='0' enables the shaper).

The clock mode must be set before power-up (CCR0H.PU). The CPU may switch the SEROCCO-D between power-up and power-down mode. This has no influence upon the contents of the registers, i.e. the internal state remains stored. In power-down mode however, all internal clocks are disabled, no interrupts from the corresponding channel are forwarded to the CPU. This state can be used as a standby mode, when the channel is (temporarily) not used, thus substantially reducing power consumption.

The SEROCCO-D should usually be initialized in Power-Down mode.

The need for programming further registers depends on the selected features (serial mode, clock mode specific features, operating mode, address mode, user demands).

6.2 Interrupt Mode

6.2.1 Data Transmission (Interrupt Driven)

In transmit direction 2 $\times$ 32 byte FIFO buffers (transmit pools) are provided for each channel. After checking the XFIFO status by polling the Transmit FIFO Write Enable bit (bit 'XFW' in STARL register) or after a Transmit Pool Ready ('XPR') interrupt, up to 32 bytes may be entered by the CPU into the XFIFO.

HDLC/SDLC/PPP

The transmission of a packet can be started by issuing an 'XF' or 'XIF' command via the CMDRL register. If enabled, a specified number of preambles (refer to registers CCR2H and PREAMB) are sent out optionally before transmission of the current packet starts.

If the transmit command does not include an end of message indication (CMDRL.XME), SEROCCO-D will repeatedly request for the next data block by means of an 'XPR' interrupt as soon as no more than 32 bytes are stored in the XFIFO, i.e. a 32-byte pool is accessible to the CPU.

This process will be repeated until the CPU indicates the end of message per 'XME' command, after which packet transmission is finished correctly by appending the CRC and closing flag sequence. Consecutive packets may be transmitted as back-to-back packets and may even share a flag (enabled via CCR1L.SFLG), if service of XFIFO is quick enough.

In case no more data is available in the XFIFO prior to the arrival of the end-of-message indiction ('XME'), the transmission of the packet is terminated with an abort sequence and the CPU is notified per interrupt (ISR1.XDU, transmit data underrun). The packet may also be aborted per software at any time (CMDRL.XRES).

The data transmission sequence, from the CPU's point of view, is outlined in **Figure 59**.

ASYNC

The transmission of character(s) can be started by issuing a 'XF' command via the CMDRL register. SEROCCO-D will repeatedly request for the next data block by means of an 'XPR' interrupt as soon as no more than 32 bytes are stored in the XFIFO, i.e. a 32-byte pool is accessible to the CPU. Transmission may be aborted per software (CMDRL.XRES).

BISYNC

The transmission of a block can be started by issuing an 'XF' command via the CMDRL register. Further handling of data transmission with respect to preamble transmission and command 'XME' is similar to HDLC/SDLC mode. After 'XME' command has been issued, the block is finished by appending the internally generated CRC if enabled (refer to description of register CCR2H).

In case no more data is available in the XFIFO prior to the arrival of 'XME', the transmission of the block is terminated with IDLE and the CPU is notified per interrupt (ISR1.XDU). The block may also be aborted per software (CMDRL.XRES). The data transmission flow, from the CPU's point of view, is outlined in **Figure 59.**

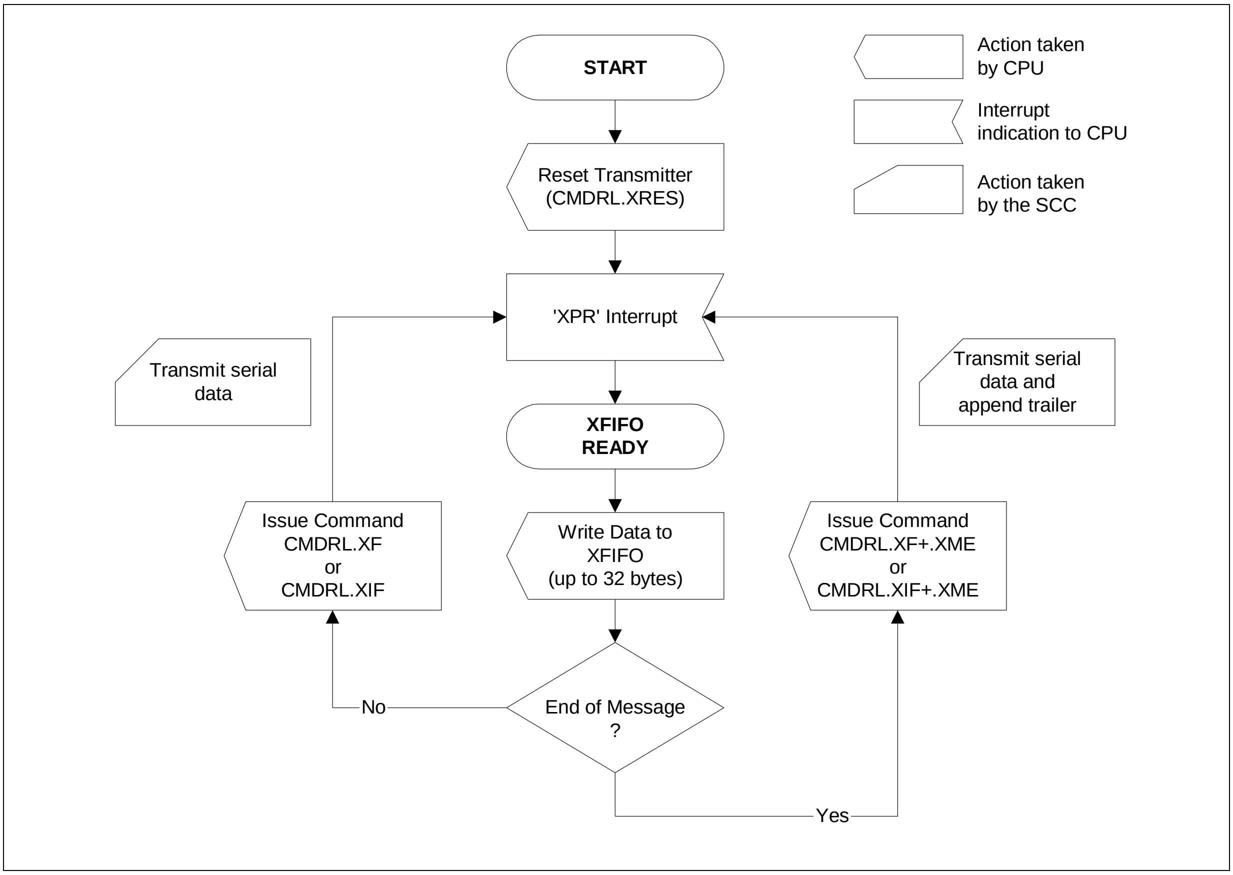


Figure 59 Interrupt Driven Data Transmission (Flow Diagram)

6.2.2 Data Reception (Interrupt Driven)

Also 2 × 32 byte FIFO buffers (receive pools) are provided for each channel in receive direction.

There are different interrupt indications concerned with the reception of data:

HDLC/SDLC/PPP

- ’RPF’ (Receive Pool Full) interrupt, indicating that a specified number of bytes (limited with the receive FIFO threshold in register CCR3H, bit field ’RFTH(1..0)’; default is 32 bytes) can be read from RFIFO and the received message is not yet complete.
- ’RME’ (Receive Message End) interrupt, indicating that the reception of one message is completed, i.e. either
  - one message which fits into RFIFO not exceeding the receive FIFO threshold, or
  - the last part of a message, all in all exceeding the receive FIFO threshold

  is stored in the RFIFO.

In addition to the message end (’RME’) interrupt the following information about the received packet is stored by SEROCCO-D in special registers and/or RFIFO:

Table 16 Status Information after RME interupt

Status Information	Location
Length of received message	registers RBCH, RBCL
CRC result (good/bad)	RSTA register (or last byte of received data)
Valid frame (yes/no)	RSTA register (or last byte of received data)
ABORT sequence recognized (yes/no)	RSTA register (or last byte of received data)
Data overflow (yes/no)	RSTA register (or last byte of received data)
Results from address comparison (with automatic address handling)	RSTA register (or last byte of received data)
Type of frame (COMMAND/RESPONSE) (with automatic address handling)	RSTA register (or last byte of received data)
Type of Signaling Unit (in SS7 mode)	RSTA register (or last byte of received data)

ASYNC, BISYNC

- 'RPF' (Receive Pool Full) interrupt, indicating that a specified number of bytes (refer to register CCR3H, bit field 'RFTH(1..0)') can be read from RFIFO.
- 'TCD' (Termination Character Detected) interrupt, indicating that reception has been terminated by reception of a specified character (refer to register TCR and bit CCR3L.TCDE).

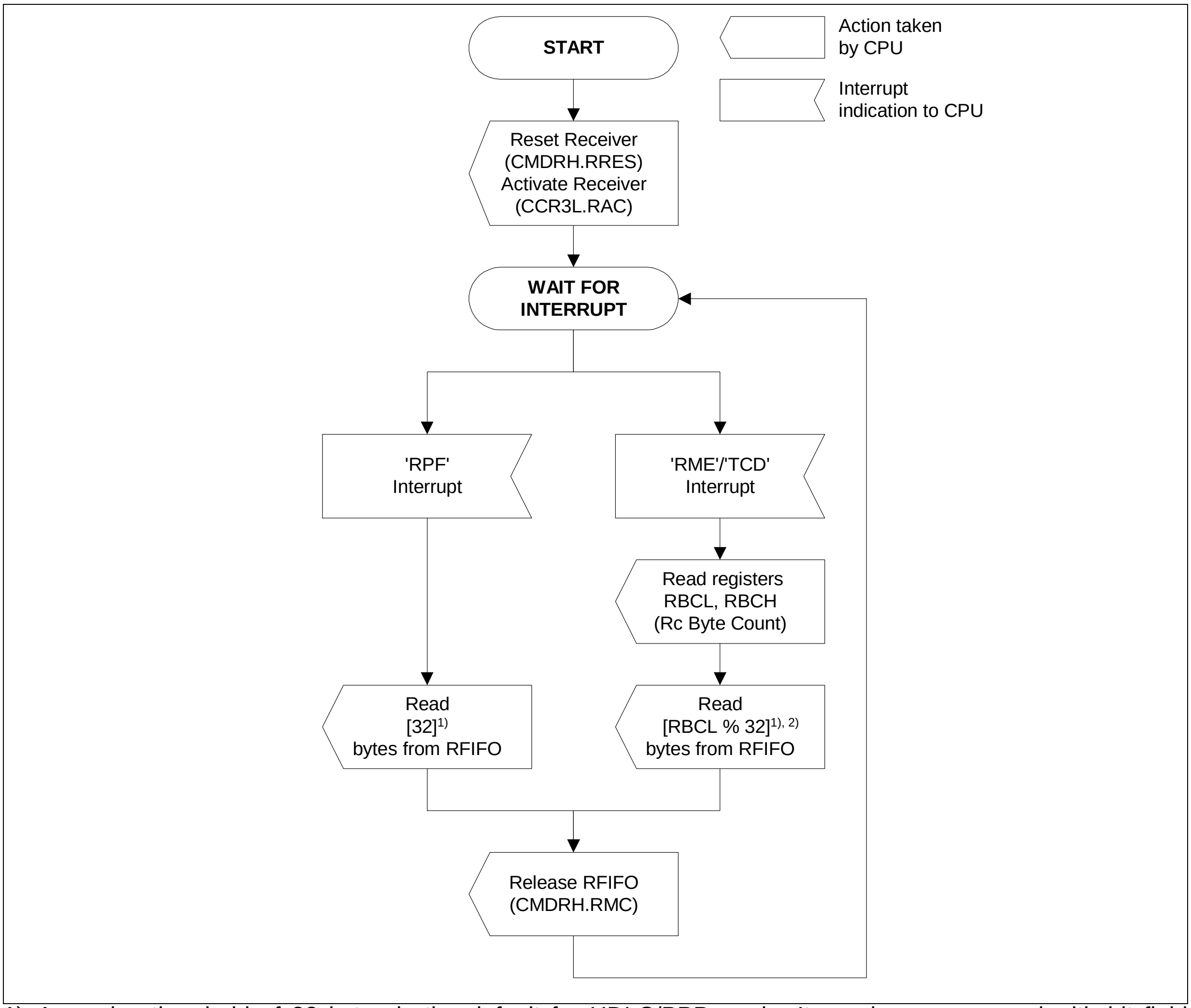
Additionally, the CPU can have access to contents of RFIFO without having received an interrupt (and thereby causing 'TCD' to occur) by issuing the RFIFO Read command (CMDRH.RFRD).

In addition to every received character the assigned status information Parity bit (0/1), Parity Error (yes/no), Framing Error (yes/no, ASYNC only!) is optionally stored in RFIFO.

With an end condition ('TCD' interrupt or after 'RFRD' command) the length of the last received data block is stored in register RBCL. The number of bytes to read from RFIFO is determined by the 1, 2, 4 or 5 least significant bits of register RBCL, depending on the selected RFIFO threshold (bit field 'RFTH(1..0)' in register CCR3H).

Note: (For all serial modes) After the received data has been read from the RFIFO, this must be explicitly acknowledged by the CPU issuing an 'RMC' (Receive Message Complete) command. The CPU has to handle the 'RPF' interrupt before the complete 2 x 32-byte FIFO is filled up with receive data which would cause a "Receive Data Overflow" condition.

The data reception sequence, from the CPU's point of view, is outlined in **Figure 60**.



1) A receive threshold of 32 bytes is the default for HDLC/PPP mode. It can be programmed with bit field RFTH(1:0) in register CCR3H.
2) The number of bytes stored in RFIFO can be determined by evaluating the lower bits in register RBCL (depending on the selected receive threshold RFTH(1:0)).

Figure 60 Interrupt Driven Data Reception (Flow Diagram)

6.3 Internal DMA Mode

The following table provides a definition of terms used in this chapter to describe the operation of the DMA controller.

Table 17 DMA Terminology

Packet	A "Packet" is a connected block of data bytes. This can be an HDLC/PPP frame as well as a number of ASYNC/ BISYNC characters up to a specific limit (received termination character, CMDRH:RFRD command). If a receive status byte (RSTA) is attached to data bytes, it is also considered as part of the packet.
Buffer	A "Buffer" is a limited space in memory that is reserved for DMA reception/transmission. Every time the DMA controller completes a buffer transfer, it notifies the CPU with an appropriate interrupt. A packet can go into one single buffer, or it can go fragmented into multiple buffers.
Block	A "Block" is the amount of data that is transfered from the memory to the XFIFO (transmit DMA transfer) or from the RFIFO to the memory. In HDLC/PPP modes the block size is 32 bytes by default. It can be lowered with the receive FIFO threshold in register CCR3H, bit field ’RFTH(1..0)’.
Bus Cycle	A "Bus Cycle" corresponds to a single byte/word transfer. Multiple bus cycles make up a block transfer.
DMA Transfer	A "DMA Transfer" is the movement of complete buffers and/or packets between the XFIFO/RFIFO and the memory.

6.3.1 Data Transmission (DMA Controlled)

Standard Transfer Mode:

Any packet transmission is prepared by writing the transmit buffer start address into registers TBADDR1L/M/H and the packet size in number of bytes to registers XBC1L/ XBC1H.

Now there are two possible scenarios:

- If the prepared transmit buffer in memory contains a complete packet, the start command for DMA transmission is issued by setting bits ’XF’ and ’XME’ in register XBC1H to ’1’. The DMA controller will request the external bus and then read transmit data beginning at address TBADDR1. The data is immediately transferred into the XFIFO. After the last byte has been transmitted, the protocol machine appends the trailer (e.g. CRC and Flag in HDLC), if applicable. The Transmit DMA Transfer End (TDTE) interrupt is generated (refer to **Figure 61**).

- If a transmit packet is distributed over more than one transmit buffer in memory, the 'XF' command (without setting the 'XME' bit)starts transmission of a buffer. A Transmit DMA Transfer End (TDTE) interrupt is generated whenever a block of <XBC1> bytes is completely transferred. For the last buffer, containing the end of the transmit packet, the 'XF' command is issued together with bit 'XME' set (refer to **Figure 62**).

After transmission is complete, the optional generation of the ALLS interrupt indicates that all transmit data has been sent on pin TxD.

Any 'XF' command resets the transmit DMA controller for new operation starting with TBADDR1 and XBC1 again.

Note: In HDLC Automode, the 'XF' command may be replaced by the 'XIF' command in the same register, when transmission of an I-frame is desired.

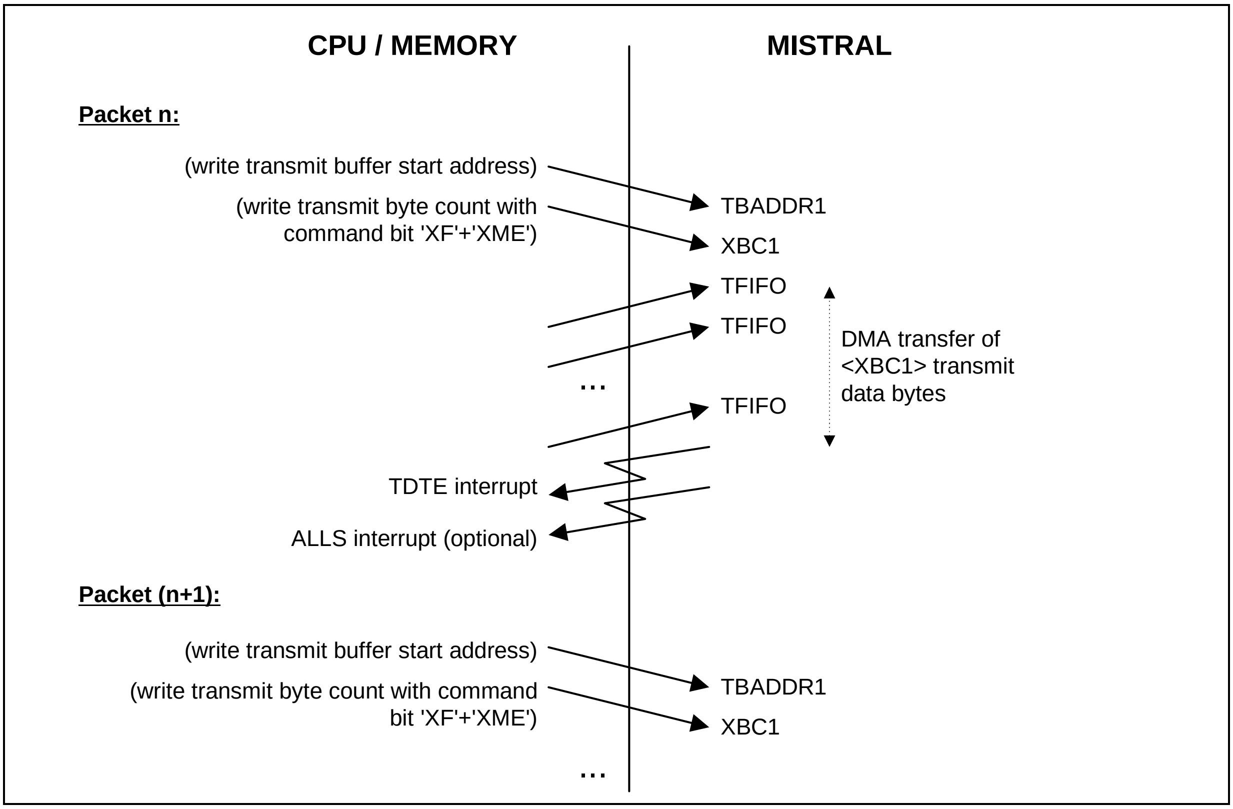


Figure 61 DMA Transmit (Single Buffer per Packet)

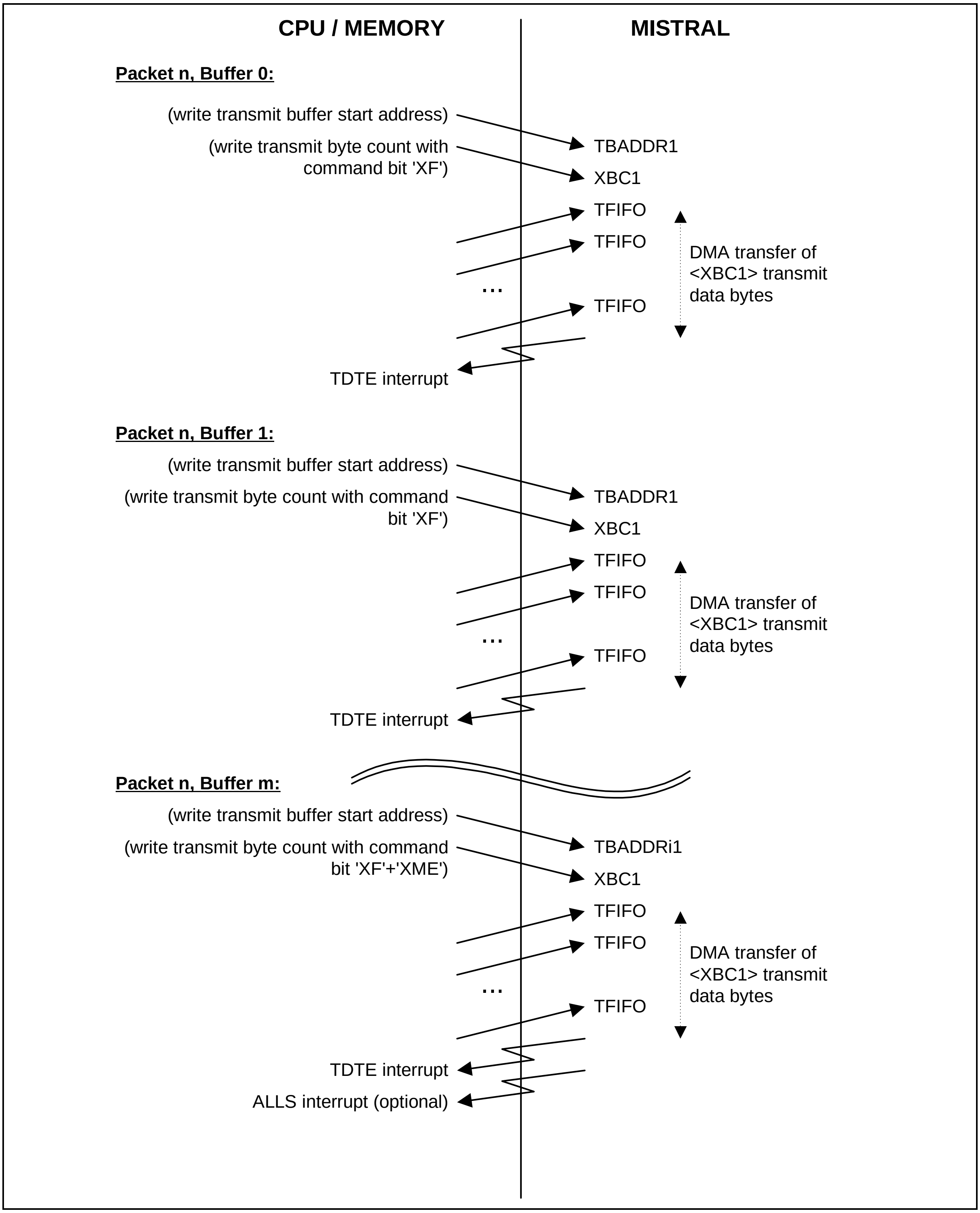


Figure 62 Fragmented DMA Transmission (Multiple Buffers per Packet)

The data transmission flow, from the CPU's point of view, is outlined in **Figure 67**.

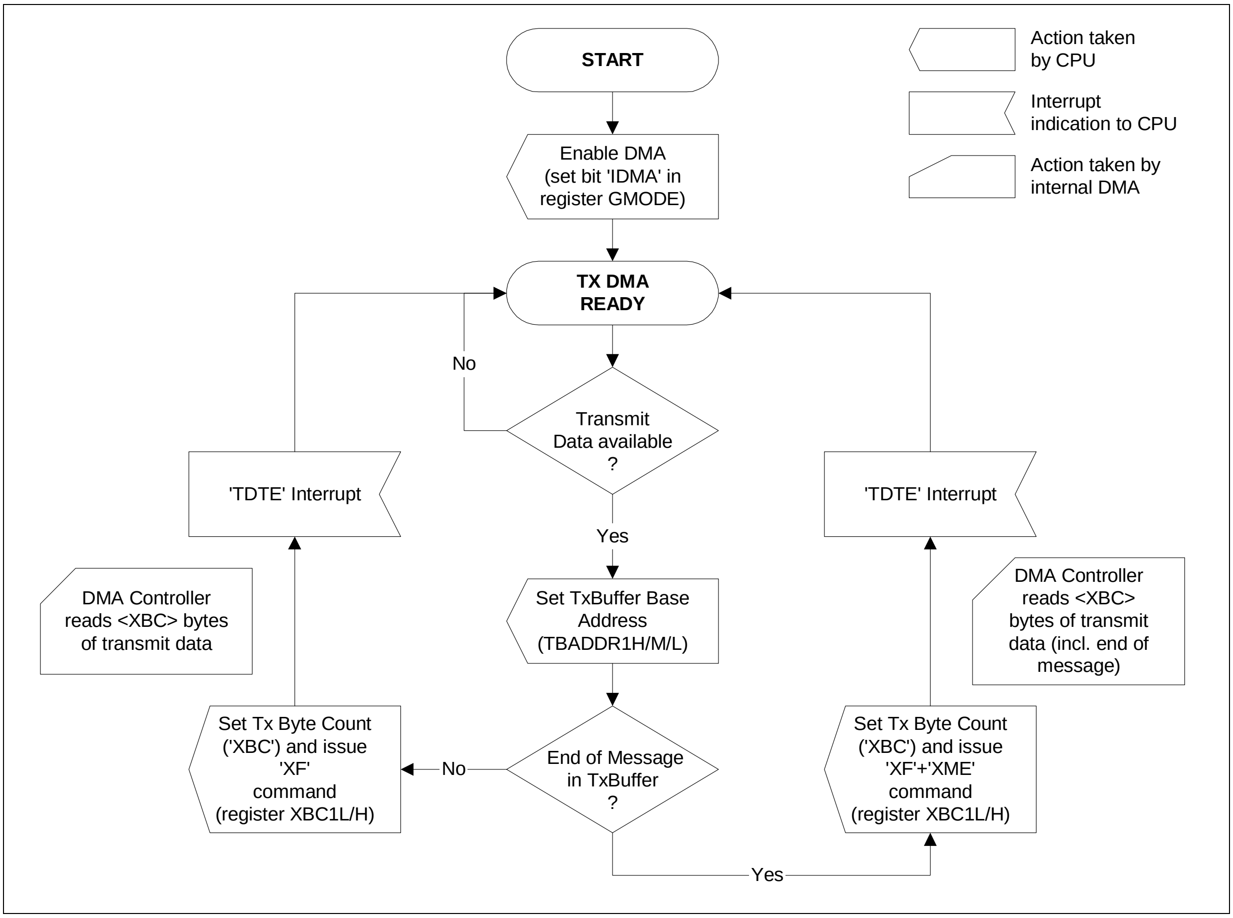


Figure 63 DMA Controlled Data Transmission (Flow Diagram)

6.3.2 Data Reception (DMA Controlled)

The receive DMA controller has to be prepared by writing an appropriate address to its RBADDR1L/M/H registers and the maximum buffer size to register RMBSL/RMBSH. If a new packet is received by the SCC, the DMA controller will request the external bus and then move receive data out of the RFIFO. The receive data is directly written on the external bus, beginning at address RBADDR1.

Now the DMA has to face two possible scenarios:

- If the maximum buffer size programmed in register RMBSL/RMBSH has been transferred, DMA transfer stops and a Receive Buffer Full (RBF) interrupt is generated. The CPU now updates the receive buffer base address in the appropriate registers RBADDR1L/M/H and restarts the DMA receiver by setting the 'RE' bit in register RMBSH. Optionally the maximum buffer size value can be updated with the same register write access.

- If the end of a received packet/block is contained in the current receive buffer, the DMA controller generates a Receive DMA Transfer End (RDTE) interrupt and stops operation. The CPU now reads the received byte count from registers RBCL/RBCH. The receive DMA controller will not continue operation until it is set up again with the ’RE’ command in register RMBSH. The software should update RBADDR1L/M/H registers if necessary before issuing the ’RE’ command.

If in packet oriented protocol modes (HDLC, PPP) the maximum receive buffer size RMBS is chosen to be larger than the expected receive packets, each buffer will contain the whole packet (see **Figure 64**). In this case a Receive Buffer Full (RBF) interrupt will never occur, simplifying the software. To ensure that no packets exceeding the maximum buffer size are forwarded from the SCC to the RFIFO, the receive packet length should be limited with registers RLCRL/RLCRH.

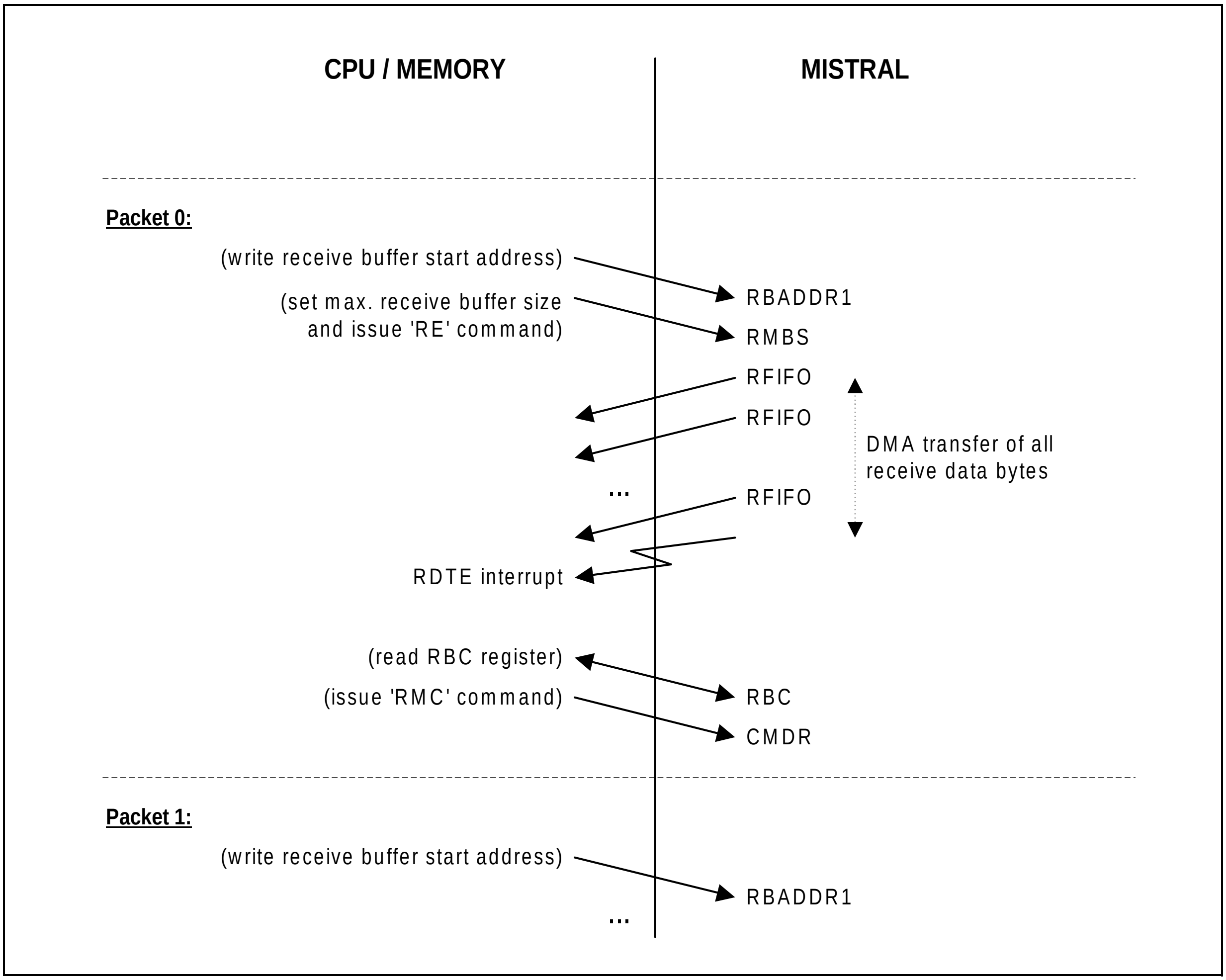


Figure 64 DMA Receive (Single Buffer per Packet)

Figure 65 shows an example for fragmented reception of a packet larger than the prepared receive buffers in memory. In this case the length of the received packet is 199 bytes, each of the buffers in host memory is 128 bytes deep:

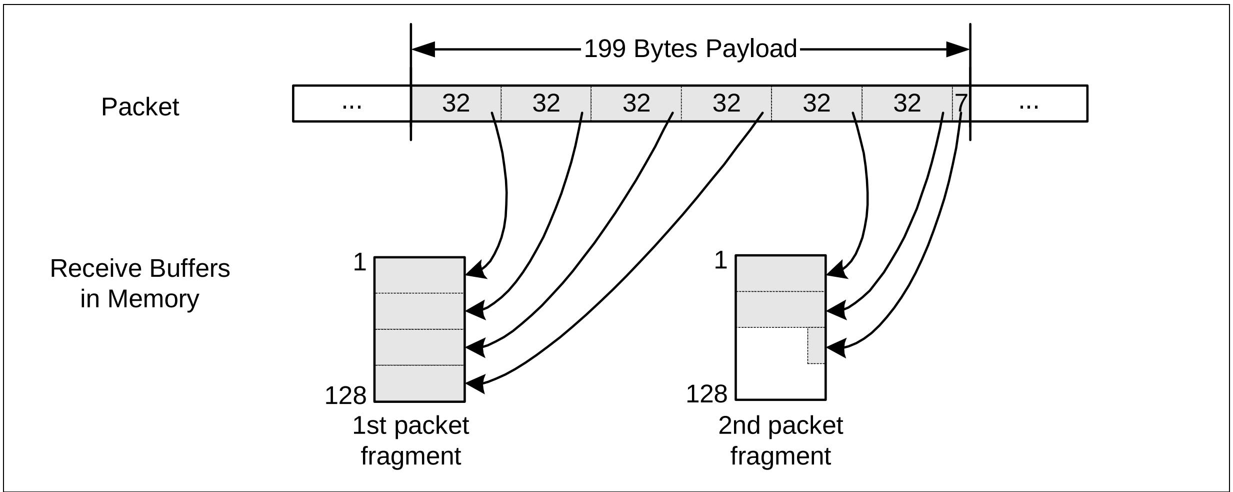


Figure 65 Fragmented Reception per DMA (Example)

After the DMA controller is initialized with the base address of receive buffer #1 and the maximum buffer size RMBS, simultaneously activated with the ’RE’ command, DMA transfer from the RFIFO to the receive buffer takes place in blocks of 32 bytes (unless changed with bit field ’RFTH’ in register CCR3H).

After four 32-byte-blocks have been transferred, the first receive buffer is filled up completely with receive data. The DMA controller indicates this by generating the RBF interrupt.

Now the CPU has to provide the base address of the second receive buffer to the DMA controller and issue the ’RE’ command again. This allows the DMA controller to continue data transfers into the second receive buffer. After another two 32-byte-blocks have been transferred, the remaining 7 bytes (including the RSTA byte) are written to the buffer, follwed by the generation of the RDTE interrupt. Now the DMA transfer is completed and software has to read the number of received bytes from the Receive Byte Count registers RBCL/RBCH.

The following figure (**Figure 66**) gives the sequence of actions from both, the internal DMA controller of SEROCCO-D and the CPU for this example (fragmented reception of 199 bytes into two receive buffers):

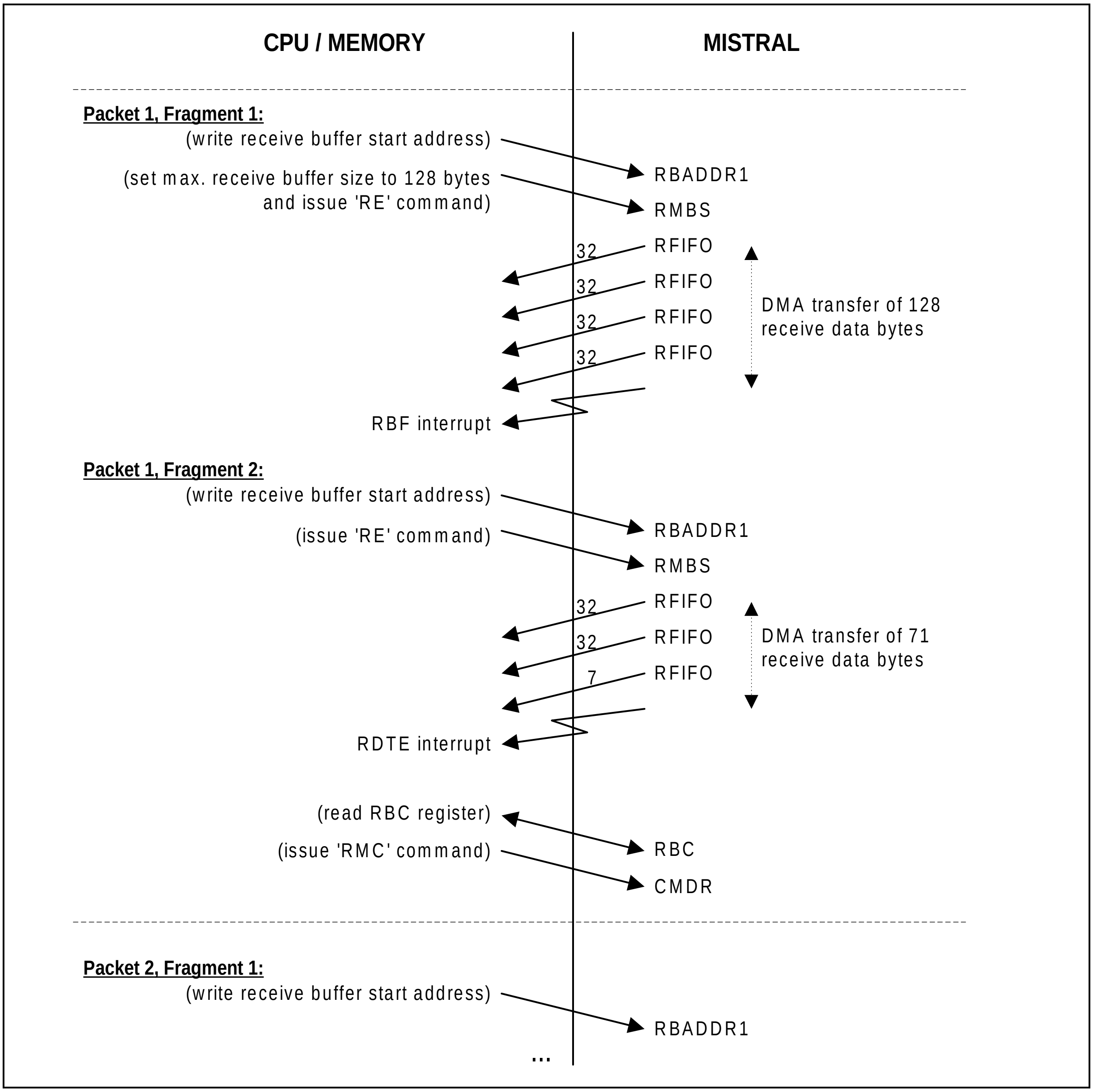


Figure 66 Fragmented Reception Sequence (Example)

Performance:

In single buffer operation, only 2 register accesses are required for transmission and 4 for reception for each buffer besides first initialization:

- update TBADDR1, update XBC1 (including command 'XF')
- update RBADDR1, issue 'RE' command (in RMBS register), read register RBC, issue 'RMC' command (in register CMDRH).

The data reception flow, from the CPU's point of view, is outlined in **Figure 67**.

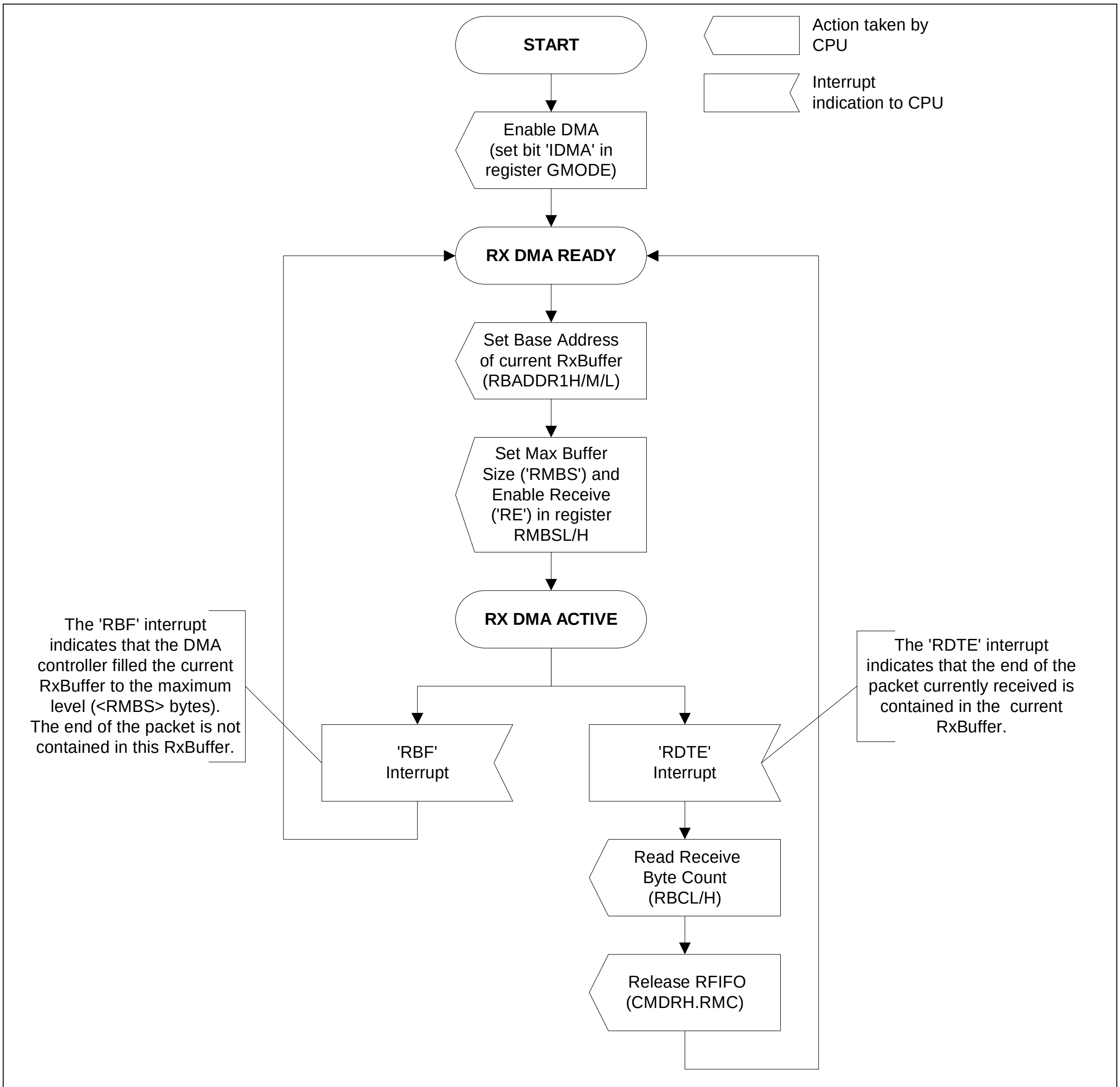


Figure 67 DMA Controlled Data Reception (Flow Diagram)

6.3.3 Buffer Switched Mode

In buffer switched mode, operation will be similar but the DMA controller will autonomously switch between buffer base addresses TBADDR1/RBADDR1 and TBADDR2/RBADDR2 after any buffer completion.

A reset command will force the DMAC to begin with base addresses TBADDR1/RBADDR1. Setting bit ’XF’ (and ’XME’) in the XBC1H register will start transmission. Setting bit ’RE’ in register RMBSH enables reception. The DMA controller automatically reloads the configuration values (base address TBADDRi/RBADDRi and byte count XBCi/RMBS) from the alternate register set. Transmission can be stopped by resetting the ’XF’ bit back to ’0’.

Status bits in register DBSR indicate which buffer is currently in operation (for debug purposes).

A TDTE interrupt indicates completion of a transmit buffer, an RDTE/RBF interrupt indicates completion of a receive buffer.

Thus DMA operation is completely autonomous with no additional register access during operation.

In case of HDLC/PPP packet oriented protocol mode, the buffer size is assumed to contain complete transmit/receive packets (single buffer operation).

Furthermore this mode supports continuous transmission (no HDLC/PPP framing) very effectively.

7 Electrical Characteristics

7.1 Absolute Maximum Ratings

Parameter		Symbol	Limit Values	Unit
Ambient temperature under bias	PEB PEF	T_A T_A	0 to 70 – 40 to 85	°C °C
Storage temperature		T_{stg}	– 65 to 125	°C
IC supply voltage		V_{DD3}	– 0.3 to 3.6	V
Voltage on any signal pin with respect to ground		V_S	– 0.4 to 5.5	V
ESD robustness[1)] HBM: 1.5 kΩ, 100 pF		$V_{ESD,HBM}$	2000	V

[1)] According to MIL-Std 883D, method 3015.7 and ESD Ass. Standard EOS/ESD-5.1-1993.

Note: ***Stresses above those listed here may cause permanent damage to the device. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.***

7.2 Operating Range

Parameter		Symbol	Limit Values		Unit	Test Condition
			min.	max.		
Ambient temperature	PEB PEF	T_A T_A	0 -40	70 85	°C °C	
Junction temperature		T_J	0	125	°C	
Supply voltage		V_{DD3}	3.0	3.6	V	
Ground		V_{SS}	0	0	V	

Note: In the operating range, the functions given in the circuit description are fulfilled.

7.3 DC Characteristics

Parameter		Symbol	Limit Values		Unit	Notes
			min.	max.		
Input low voltage		V_{IL}	– 0.4	0.8	V	
Input high voltage		V_{IH}	2.0 2.1	5.5 5.5	V V	V_{DD} = 3.3 V V_{DD} = 3.6 V
Output low voltage		V_{OL}		0.45	V	I_{OL} = 7 mA [1)] I_{OL} = 2 mA [2)]
Output high voltage		V_{OH}	2.4		V	I_{OH} = – 1.0 mA
Power supply current	operational (average)	I_{CC} (AV)		50	mA	V_{DD} = 3.3 V, T_A = 25 °C, CLK = 33 MHz, XTAL = 20 MHz, inputs at V_{SS}/V_{DD}, no output loads
	power down (no clocks)	I_{CC} (PD)		0.01	mA	V_{DD} = 3.3 V, T_A = 25 °C
Power dissipation		P		150	mW	V_{DD} = 3.3 V, T_A = 25 °C, CLK = 33 MHz, XTAL = 20 MHz, inputs at V_{SS}/V_{DD}, no output loads
Input leakage current		I_{LI}		1	µA	V_{DD} = 3.3 V, GND = 0 V; inputs at V_{SS}/V_{DD}, no output loads
Output leakage current		I_{LO}		1	µA	V_{DD} = 3.3 V, GND = 0 V; V_{OUT} = 0 V, V_{DDP} + 0.4

1) Apply to the next pins: TxDA, TxDB.

2) Apply to all the I/O and O pins that do not appear in the list in note [1)], except XTAL2.

The listed characteristics are ensured over the operating range of the integrated circuit. Typical characteristics specify mean values expected over the production spread. If not otherwise specified, typical characteristics apply at T_A = 25 °C and the given supply voltage.

7.4 AC Characteristics

Interface Pins

T_A = 0 to + 70 °C; V_{DD3} = 3.3 V ± 0.3 V

Inputs are driven to 2.4 V for a logical "1" and to 0.4 V for a logical "0". Timing measurements are made at 2.0 V for a logical "1" and at 0.8 V for a logical "0".

The AC testing input/output waveforms are shown below.

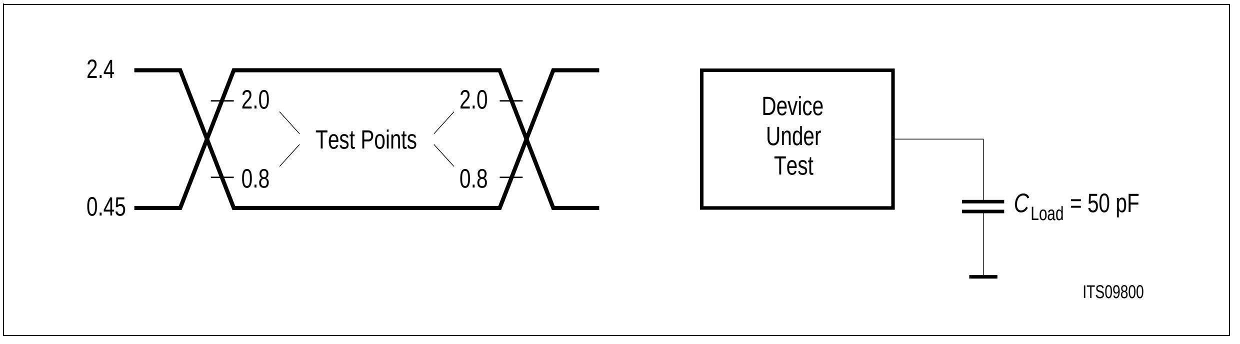


Figure 68 Input/Output Waveform for AC Tests

7.5 Capacitances

Interface Pins

Table 18 Capacitances
T_A = 25 °C; V_{DD3} = 3.3 V ± 0.3 V, V_{SS} = 0 V

Parameter	Symbol	Limit Values		Unit	Test Condition
		min.	max.		
Input capacitance	C_{IN}		5	pF	
Output capacitance	C_{OUT}		10	pF	
I/O-capacitance	C_{IO}		15	pF	

7.6 Thermal Package Characteristics

Table 19 Thermal Package Characteristics P-TQFP-144-10

Parameter		Symbol	Value	Unit
Thermal Package Resistance Junction to Ambient				
Airflow:	Ambient Temperature:			
without airflow	T_A=-40°C	$\theta_{JA(0,-40)}$	43.0	K/W
without airflow	T_A=+25°C	$\theta_{JA(0,25)}$	38.9	K/W
airflow 1 m/s (~200 lfpm)	T_A=+25°C	$\theta_{JA(1,25)}$	37.0	K/W
airflow 2 m/s (~400 lfpm)	T_A=+25°C	$\theta_{JA(2,25)}$	36.4	K/W
airflow 3 m/s (~600 lfpm)	T_A=+25°C	$\theta_{JA(3,25)}$	36.0	K/W

7.7 Timing Diagrams

7.7.1 Microprocessor Interface Timing

7.7.1.1 Microprocessor Interface Clock Timing

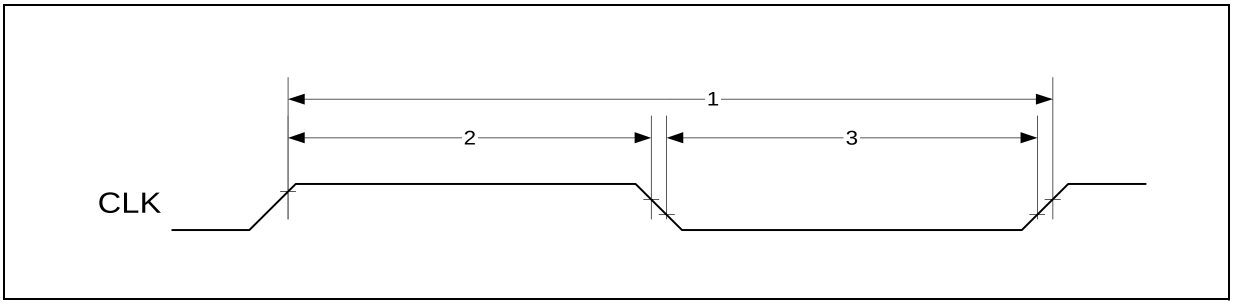


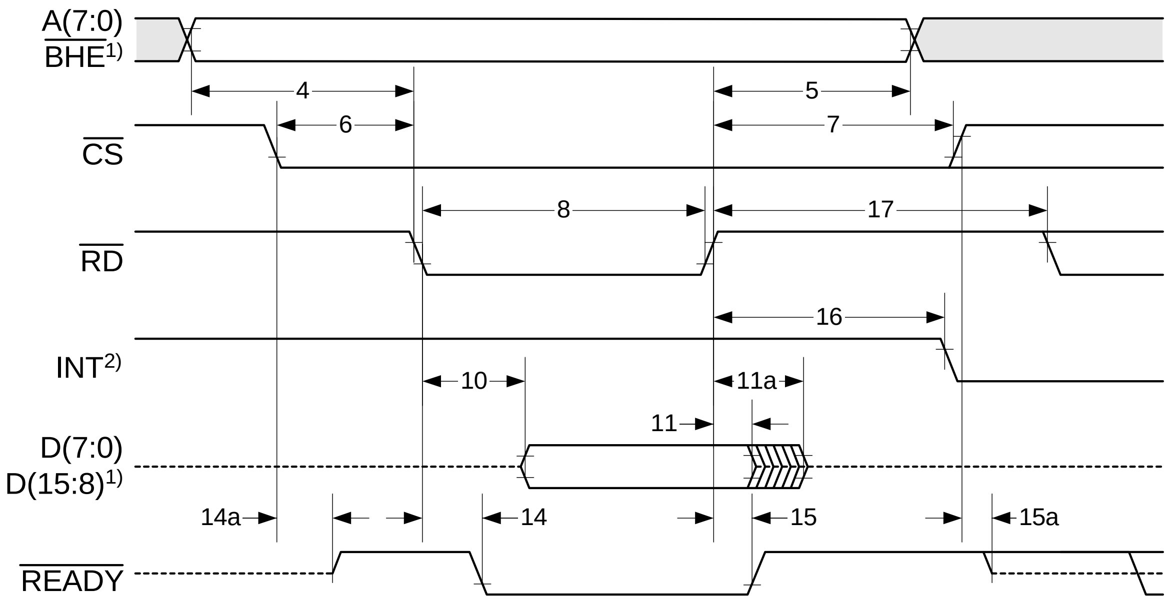
Figure 69 Microprocessor Interface Clock Timing

Table 20 Microprocessor Interface Clock Timing

No.	Parameter	Limit Values		Unit
		min.	max.	
1	CLK clock period	30	∞[1)]	ns
	CLK frequency	0	33	MHz
2	CLK high time	11	∞	ns
3	CLK low time	11	∞	ns

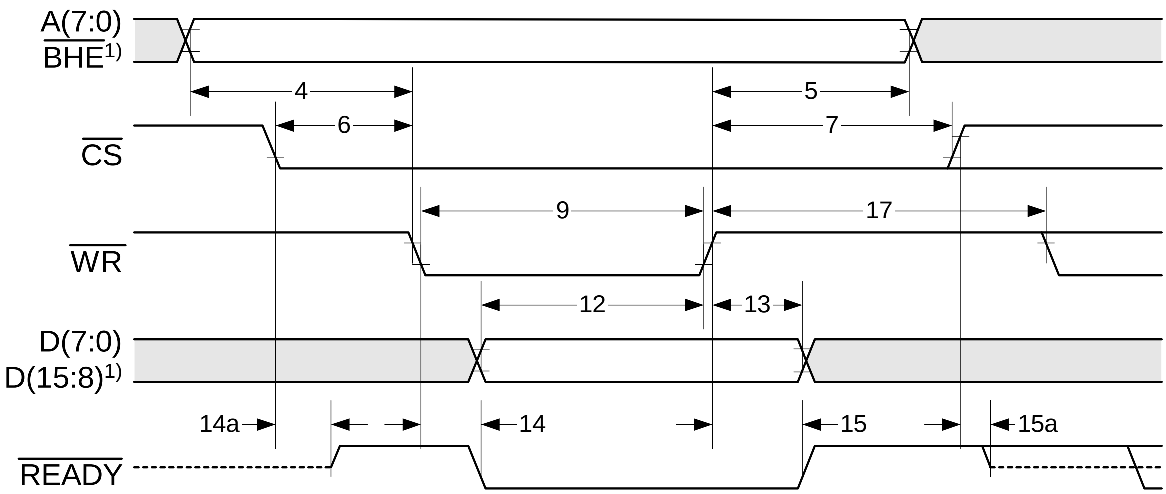
1) A clock supply is needed for read access to the on-chip interrupt status registers (ISR, DISR) and for general purpose port (GPP) operation.

7.7.1.2 Infineon/Intel Bus Interface Timing (Slave Access)



(1) Signals BHE and D(15:8) only available in 16-bit Infineon/Intel bus mode.
(2) Interrupt signal shown is push-pull, active high. Same timings apply to push-pull, active low interrupt signal. In case of open-drain output the timing depends on external components.

Figure 70 Infineon/Intel Read Cycle Timing (Slave Access)



(1) Signals BHE and D(15:8) only available in 16-bit Infineon/Intel bus mode.

Figure 71 Infineon/Intel Write Cycle Timing (Slave Access)

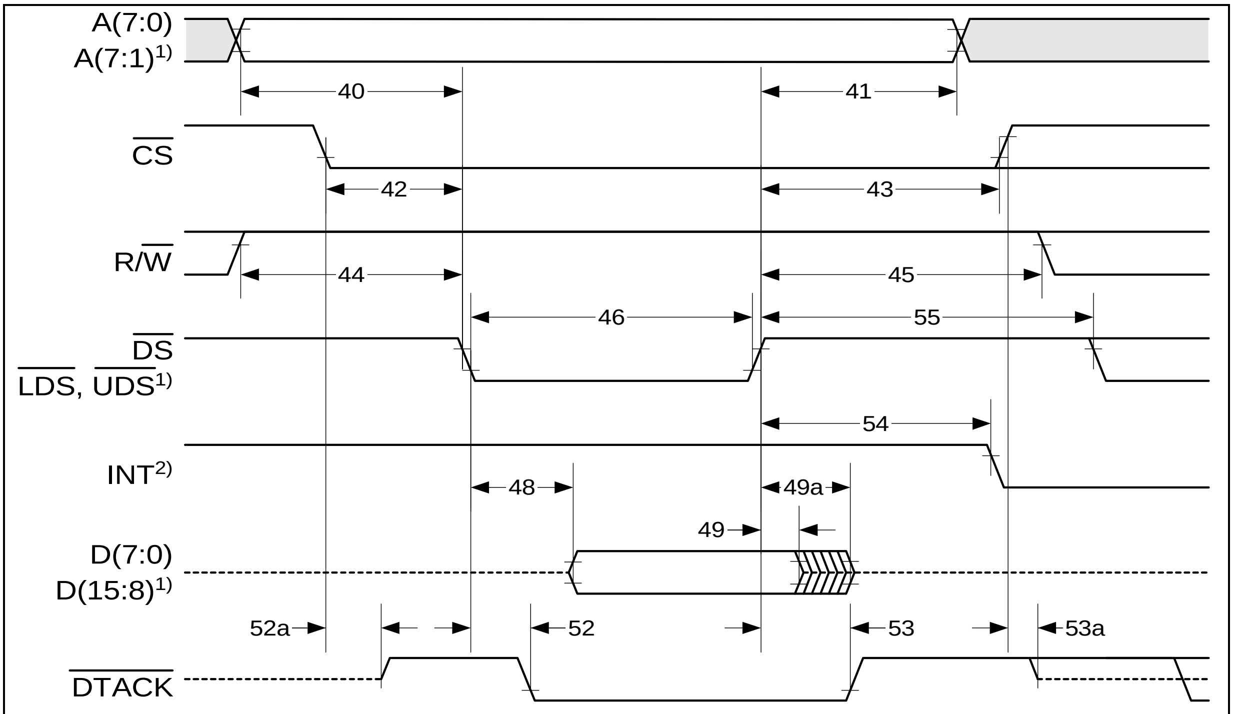
Table 21 Infineon/Intel Bus Interface Timing (Slave Access)

No.	Parameter	Limit Values		Unit
		min.	max.	
4	active address to active $\overline{\text{RD}}/\overline{\text{WR}}$ setup time	8		ns
5	inactive $\overline{\text{RD}}/\overline{\text{WR}}$ to inactive address hold time	0		ns
6	active $\overline{\text{CS}}$ to active $\overline{\text{RD}}/\overline{\text{WR}}$ setup time	2		ns
7	inactive $\overline{\text{RD}}/\overline{\text{WR}}$ to inactive $\overline{\text{CS}}$ hold time	0		ns
8	$\overline{\text{RD}}$ active pulse width	30[1)]		ns
9	$\overline{\text{WR}}$ active pulse width	30		ns
10	active $\overline{\text{RD}}$ to valid data delay		20	ns
11	inactive $\overline{\text{RD}}$ to invalid data hold time	5		ns
11a	inactive $\overline{\text{RD}}$ to data high impedance delay		25	ns
12	valid data to inactive $\overline{\text{WR}}$ setup time	6		ns
13	inactive $\overline{\text{WR}}$ to invalid data hold time	5		ns
14	active $\overline{\text{RD}}/\overline{\text{WR}}$ to active $\overline{\text{READY}}$ delay		20	ns
14a	active $\overline{\text{CS}}$ to driven $\overline{\text{READY}}$ delay		20	ns
15	inactive $\overline{\text{RD}}/\overline{\text{WR}}$ to inactive $\overline{\text{READY}}$ delay		15	ns
15a	inactive $\overline{\text{CS}}$ to $\overline{\text{READY}}$ high impedance delay		15	ns
16	inactive $\overline{\text{RD}}$ to inactive INT/$\overline{\text{INT}}$ delay		1	T_{CLK}[2)]
17	$\overline{\text{RD}}/\overline{\text{WR}}$ inactive pulse width	30		ns

1) At least one rising CLK edge must appear during read pulse active for interrupt status register (ISR, DISR) read.

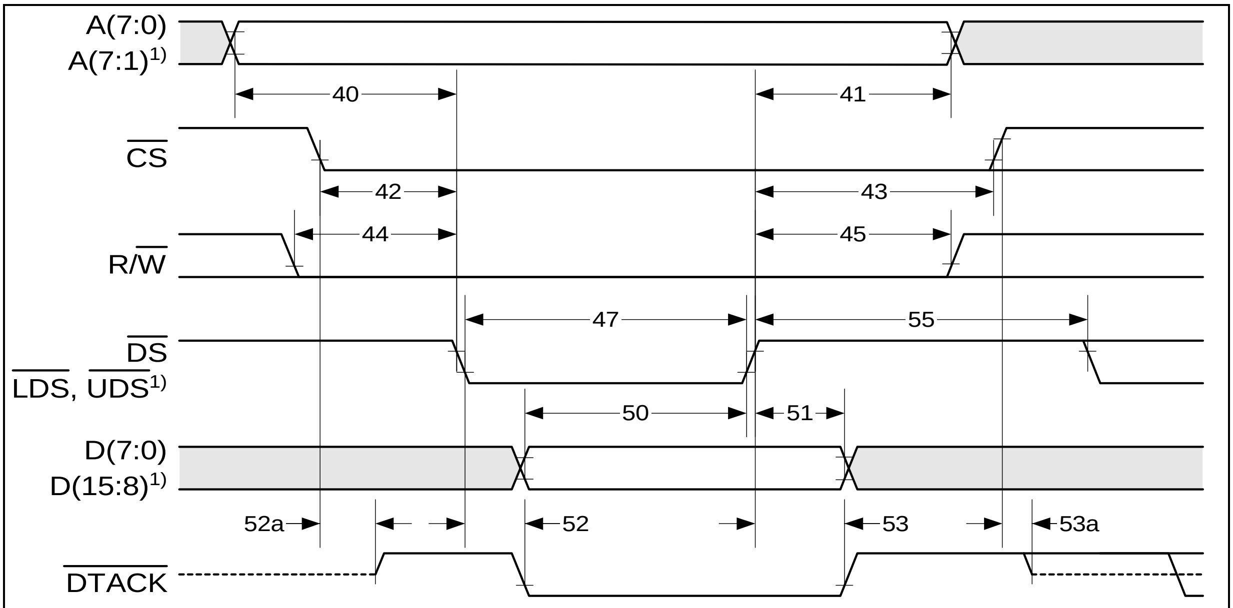
2) T_{CLK} is the system clock (CLK) period.

7.7.1.3 Motorola Bus Interface Timing (Slave Access)



(1) Signals LDS, UDS and D(15:8) only available in 16-bit Motorola bus mode
(2) Interrupt signal shown is push-pull, active high. Same timings apply to push-pull, active low interrupt signal. In case of open-drain output the timing depends on external components.

Figure 72 Motorola Read Cycle Timing (Slave Access)



(1) Signals LDS, UDS and D(15:8) only available in 16-bit Motorola bus mode

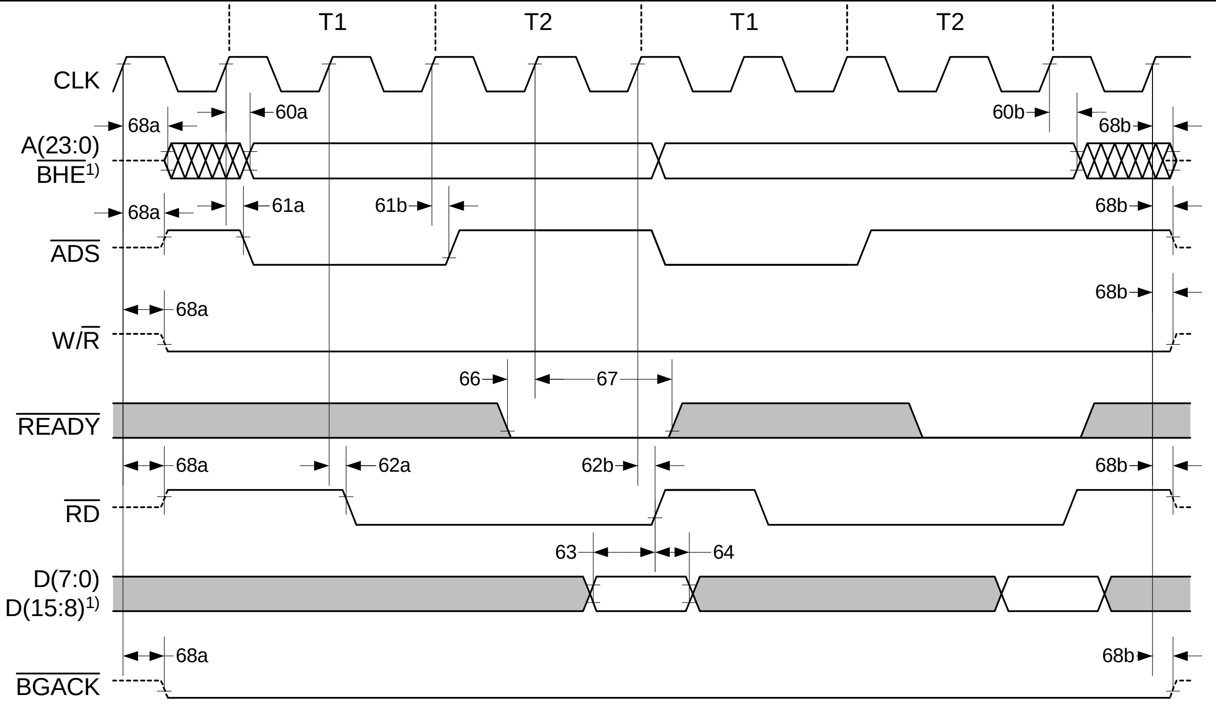
Figure 73 Motorola Write Cycle Timing (Slave Access)

Table 22 Motorola Bus Interface Timing (Slave Access)

No.	Parameter	Limit Values		Unit
		min.	max.	
40	active address to active $\overline{DS}$ setup time	0		ns
41	inactive $\overline{DS}$ to inactive address hold time	0		ns
42	active $\overline{CS}$ to active $\overline{DS}$ setup time	0		ns
43	inactive $\overline{DS}$ to inactive $\overline{CS}$ hold time	0		ns
44	active R/$\overline{W}$ to active $\overline{DS}$ setup time	0		ns
45	inactive $\overline{DS}$ to inactive R/$\overline{W}$ hold time	0		ns
46	$\overline{DS}$ active pulse width (read access)	30[1]		ns
47	$\overline{DS}$ active pulse width (write access)	30		ns
48	active $\overline{DS}$ (read) to valid data delay		20	ns
49	inactive $\overline{DS}$ (read) to invalid data hold time	5		ns
49a	inactive $\overline{DS}$ (read) to data high impedance delay		20	ns
50	valid data to inactive $\overline{DS}$ (write) setup time	10		ns
51	inactive $\overline{DS}$ (write) to invalid data hold time	10		ns
52	active $\overline{DS}$ to active $\overline{DTACK}$ delay		20	ns
52a	active $\overline{CS}$ to driving $\overline{DTACK}$ delay		20	ns
53	inactive $\overline{DS}$ to inactive $\overline{DTACK}$ delay		15	ns
53a	inactive $\overline{CS}$ to $\overline{DTACK}$ high impedance delay		15	ns
54	inactive $\overline{DS}$ (read) to inactive INT/$\overline{INT}$ delay		1	T_{CLK}
55	$\overline{DS}$ inactive pulse width	30		ns
56	active $\overline{DS}$ (read) to inactive DRR delay	22		ns
57	active $\overline{DS}$ (write) to inactive DRT delay	22		ns

[1] At least one rising CLK edge must appear during read data strobe active for interrupt status register (ISR, DISR) read.

7.7.1.4 Infineon/Intel Bus Interface Timing (Master Access)



(1) Signals BHE and D(15:8) only available in 16-bit Infineon/Intel bus mode

Figure 74 Infineon/Intel Read Cycle Timing (Master Access)

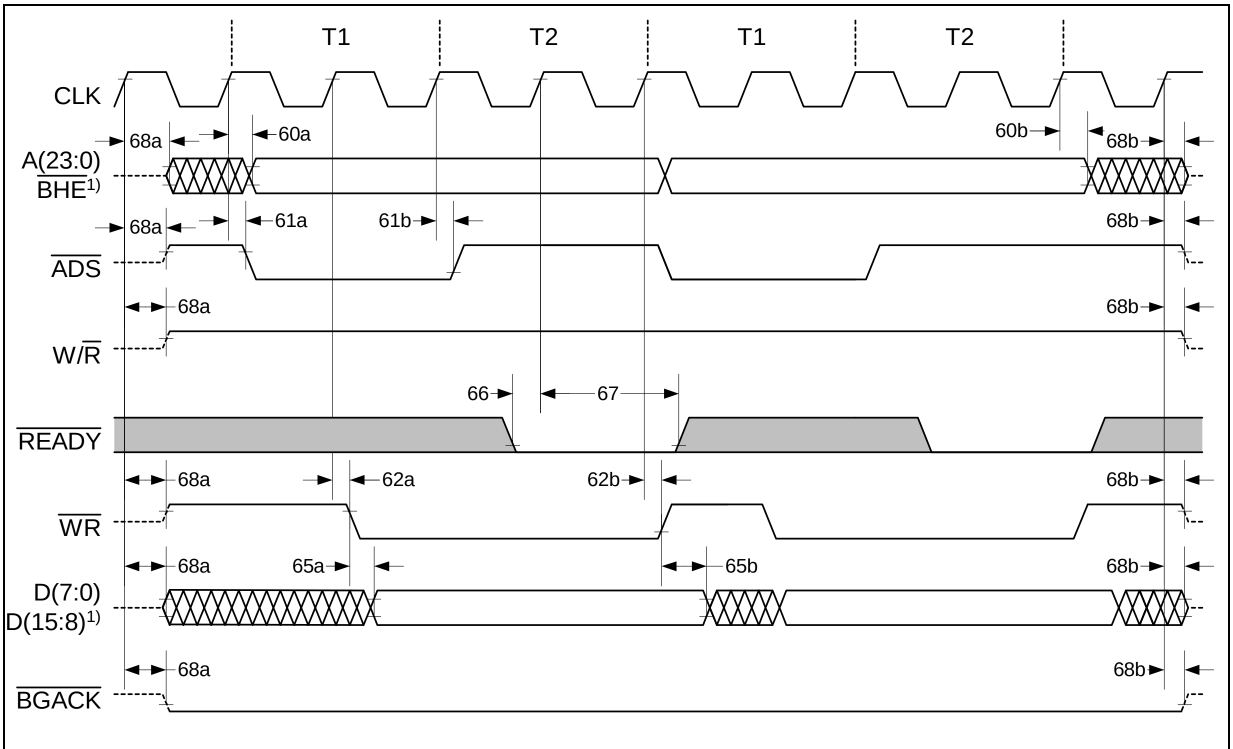


Figure 75 Infineon/Intel Write Cycle Timing (Master Access)

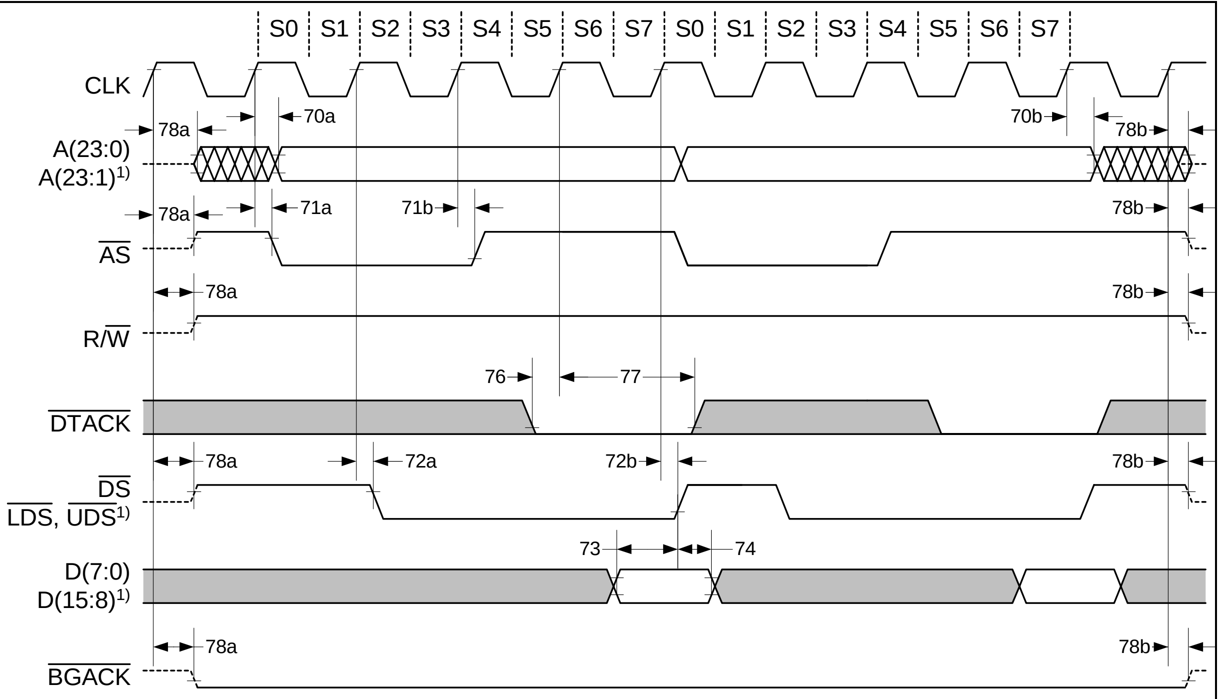
Table 23 Infineon/Intel Bus Interface Timing (Master Access)

No.	Parameter	Limit Values		Unit
		min.	max.	
60a	clock to valid address delay		22	ns
60b	clock to invalid address delay		22	ns
61a	clock to active ADS delay		15	ns
61b	clock to inactive ADS delay		15	ns
62a	clock to active RD / WR delay		20	ns
62b	clock to inactive RD / WR delay		20	ns
63	valid data to inactive RD setup time	5		ns
64	inactive RD to invalid data hold time	5		ns
65a	active WR to valid data delay		20	ns
65b	inactive WR to invalid data delay		20	ns
66	active READY to clock setup time	5		ns
67	clock to inactive READY hold time	5		ns

Table 23 Infineon/Intel Bus Interface Timing (Master Access) (cont'd)

No.	Parameter	Limit Values		Unit
		min.	max.	
68a	clock to driving bus delay		22	ns
68b	clock to bus high impedance delay		22	ns

7.7.1.5 Motorola Bus Interface Timing (Master Access)



(1) Signals LDS, UDS and D(15:8) only available in 16-bit Motorola bus mode

Figure 76 Motorola Read Cycle Timing (Master Access)

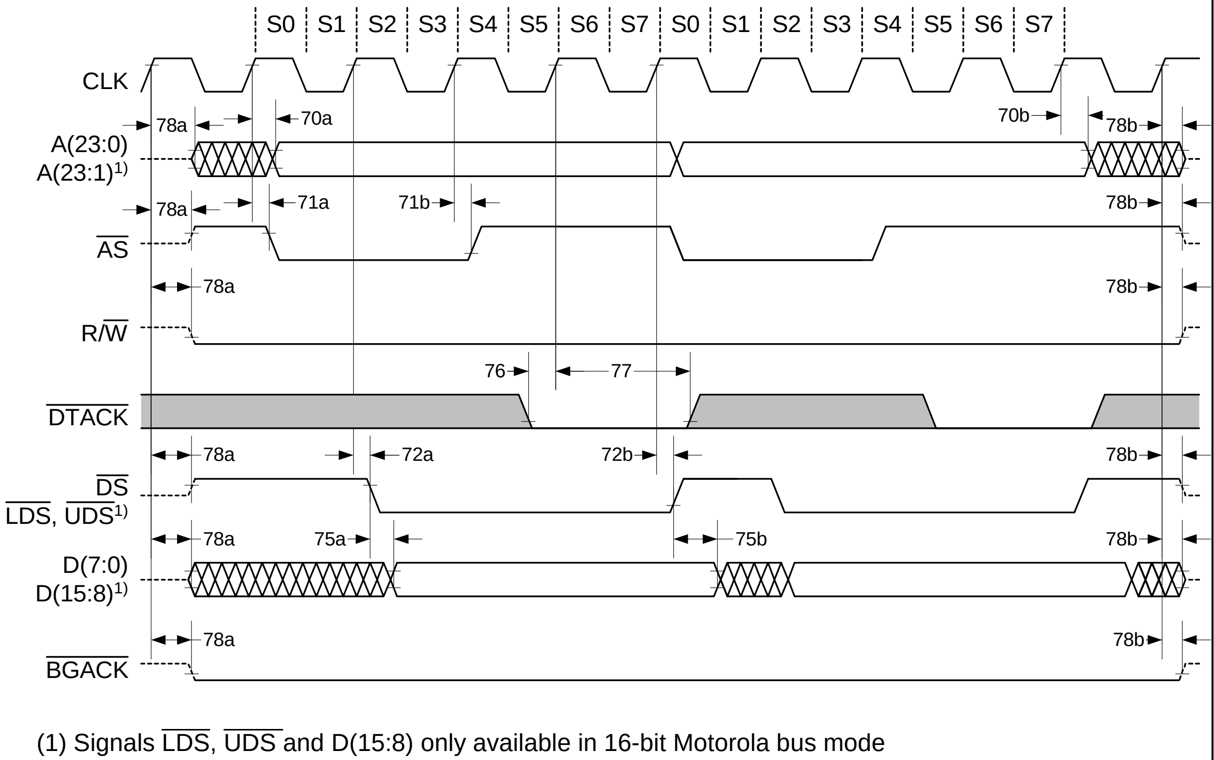


Figure 77 Motorola Write Cycle Timing (Master Access)

Table 24 Motorola Bus Interface Timing (Master Access)

No.	Parameter	Limit Values		Unit
		min.	max.	
70a	clock to valid address delay		22	ns
70b	clock to invalid address delay		22	ns
71a	clock to active AS delay		20	ns
71b	clock to inactive AS delay		20	ns
72a	clock to active DS / LDS/UDS delay		20	ns
72b	clock to inactive DS / LDS/UDS delay		20	ns
73	valid data to inactive DS (read) setup time	5		ns
74	inactive DS (read) to invalid data hold time	5		ns
75a	active DS (write) to valid data delay		20	ns
75b	inactive DS (write) to invalid data delay		20	ns
76	active DTACK to clock setup time	5		ns
77	clock to inactive DTACK hold time	5		ns

Table 24 Motorola Bus Interface Timing (Master Access) (cont'd)

No.	Parameter	Limit Values		Unit
		min.	max.	
78a	clock to driving bus delay		22	ns
78b	clock to bus high impedance delay		22	ns

7.7.1.6 Bus Arbitration Timing

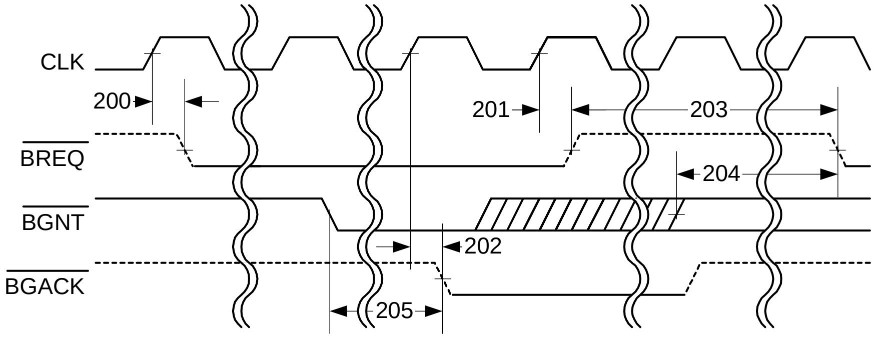


Figure 78 Bus Arbitration Timing

Table 25 Bus Arbitration Timing

No.	Parameter	Limit Values		Unit
		min.	max.	
200	clock to active $\overline{BREQ}$ delay		22	ns
201	clock to inactive $\overline{BREQ}$ delay		24	ns
202	clock to active $\overline{BGACK}$ delay		22	ns
203	$\overline{BREQ}$ inactive time	7		T_{CLK}
204	inactive $\overline{BGNT}$ to active $\overline{BREQ}$ delay	2		T_{CLK}
205	active $\overline{BGNT}$ to active $\overline{BGACK}$ delay		4	T_{CLK}

7.7.2 PCM Serial Interface Timing

7.7.2.1 Clock Input Timing

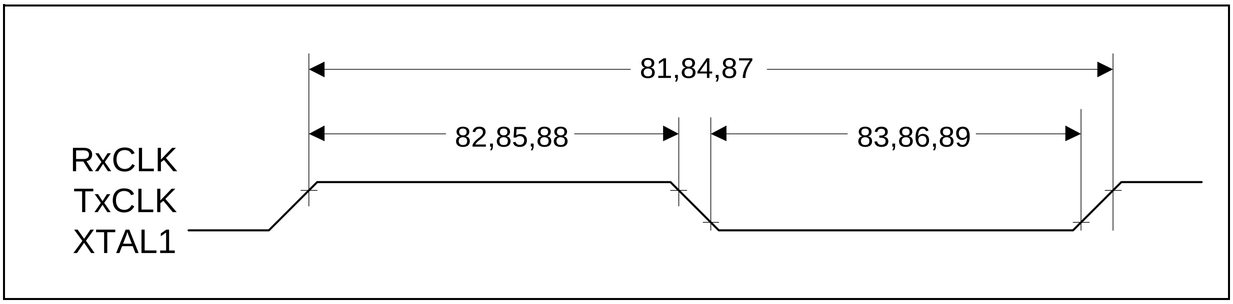
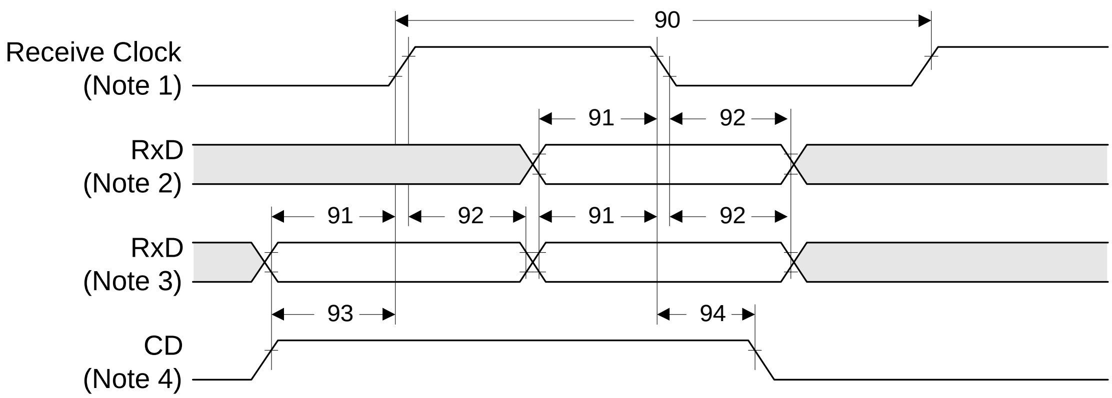


Figure 79 Clock Input Timing

Table 26 Clock Input Timing

No.	Parameter	Limit Values		Unit
		min.	max.	
81	RxCLK clock period	62	∞	ns
82	RxCLK high time	25	∞	ns
83	RxCLK low time	25	∞	ns
84	TxCLK clock period	62	∞	ns
85	TxCLK high time	25	∞	ns
86	TxCLK low time	25	∞	ns
87	XTAL1 clock period (internal oscillator used)	25	100	ns
	XTAL1 clock period (TTL clock signal supplied)	25	∞	ns
88	XTAL1 high time (internal oscillator used)	12	46	ns
	XTAL1 high time (TTL clock signal supplied)	12	∞	ns
89	XTAL1 low time (internal oscillator used)	12	46	ns
	XTAL1 low time (TTL clock signal supplied)	12	∞	ns

7.7.2.2 Receive Cycle Timing



(1) Whichever supplies the receive clock depending on the selected clock mode:
externally clocked via RxCLK or XTAL1 or
internally clocked via DPLL, BCR or BRG.
(No edge relation can be measured if the internal receive clock is derived from the external clock source by division stages (BRG, BCR) or DPLL)
(2) NRZ, NRZI and Manchester data encoding
(3) FM0 and FM1 data encoding
(4) If Carrier Detect auto start feature enabled (not for clock modes 1, 4 and 5)

Figure 80 Receive Cycle Timing

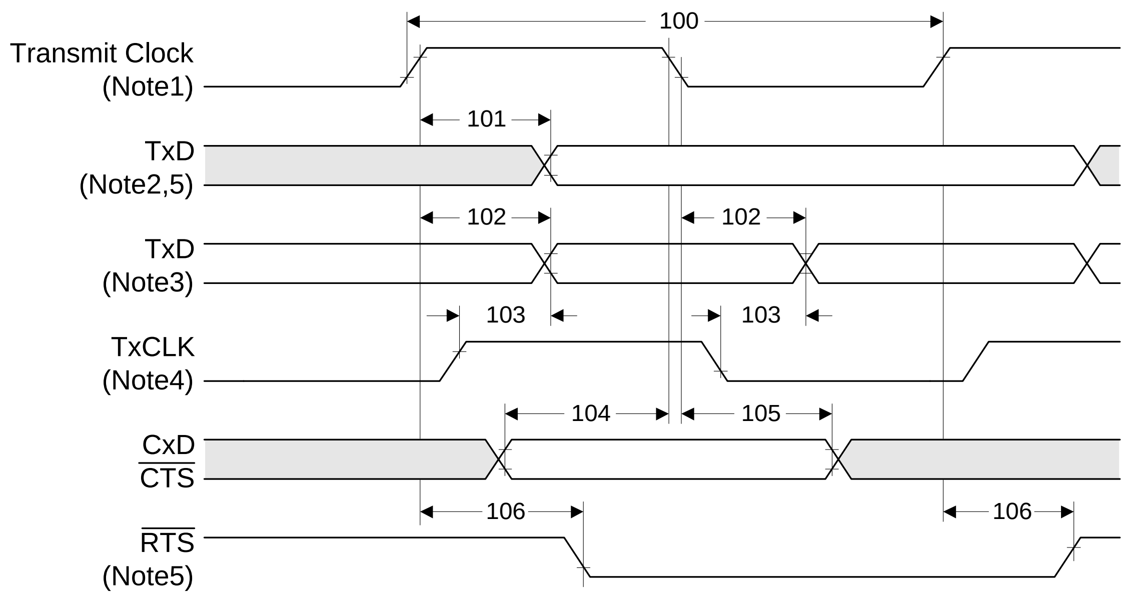
Table 27 Receive Cycle Timing

No.	Parameter		Limit Values		Unit
			min.	max.	
Receive data rates		externally clocked (HDLC)	0	16	Mbit/s
		internally clocked (DPLL modes)	0	2	Mbit/s
		internally clocked (non DPLL modes)	0	16	Mbit/s
90	Clock period	externally clocked	62	∞	ns
		internally clocked (DPLL modes)	480	∞	ns
		internally clocked (non DPLL modes)	62	∞	ns
91	RxD to RxCLK setup time		5		ns
92	RxD to RxCLK hold time		5		ns

Table 27 Receive Cycle Timing (cont'd)

No.	Parameter	Limit Values		Unit
		min.	max.	
93	CD to RxCLK rising edge setup time	5		ns
94	CD to RxCLK falling edge hold time	5		ns

7.7.2.3 Transmit Cycle Timing



(1) Whichever supplies the transmit clock depending on the selected clock mode:
externally clocked via TxCLK, RxCLK or XTAL1 or
internally clocked via DPLL, BCR or BRG.
(No edge relation can be measured if the internal transmit clock is derived from the external clock source by division stages (BRG, BCR) or DPLL)
(2) NRZ, NRZI and Manchester data encoding
(3) FM0 and FM1 data encoding
(4) If TxCLK output feature is enabled (only in some clock modes)
(5) The timing is valid for non bus configuration modes and bus configuration mode 1. In bus configuration mode 2, TxD and RTS are right shifted for 0.5 TxCLK periods i.e. driven by the falling TxCLK edge.

Figure 81 Transmit Cycle Timing

Table 28 Transmit Cycle Timing

No.	Parameter		Limit Values min.	Limit Values max.	Unit
Transmit data rates		externally clocked	0	16	Mbit/s
		internally clocked (DPLL modes)	0	2	Mbit/s
		internally clocked (non DPLL modes)	0	16	Mbit/s
100	Clock period	externally clocked	62	∞	ns
		internally clocked (DPLL modes)	480	∞	ns
		internally clocked (non DPLL modes)	62	∞	ns
101	TxD to TxCLK delay (NRZ, NRZI encoding)			25	ns
102	TxD to TxCLK delay (FM0, FM1, Manchester encoding)			25	ns
103	TxD to TxCLK(out) delay (output function enabled)		10	25	ns
104	CxD to TxCLK setup time		5		ns
	$\overline{\text{CTS}}$ to TxCLK setup time		5		ns
105	CxD to TxCLK hold time		5		ns
	$\overline{\text{CTS}}$ to TxCLK hold time		5		ns
106	$\overline{\text{RTS}}$ to TxCLK delay (not bus configuration mode)			20	ns
	$\overline{\text{RTS}}$ to TxCLK delay (bus configuration mode)			20	ns

7.7.2.4 Clock Mode 1 Strobe Timing

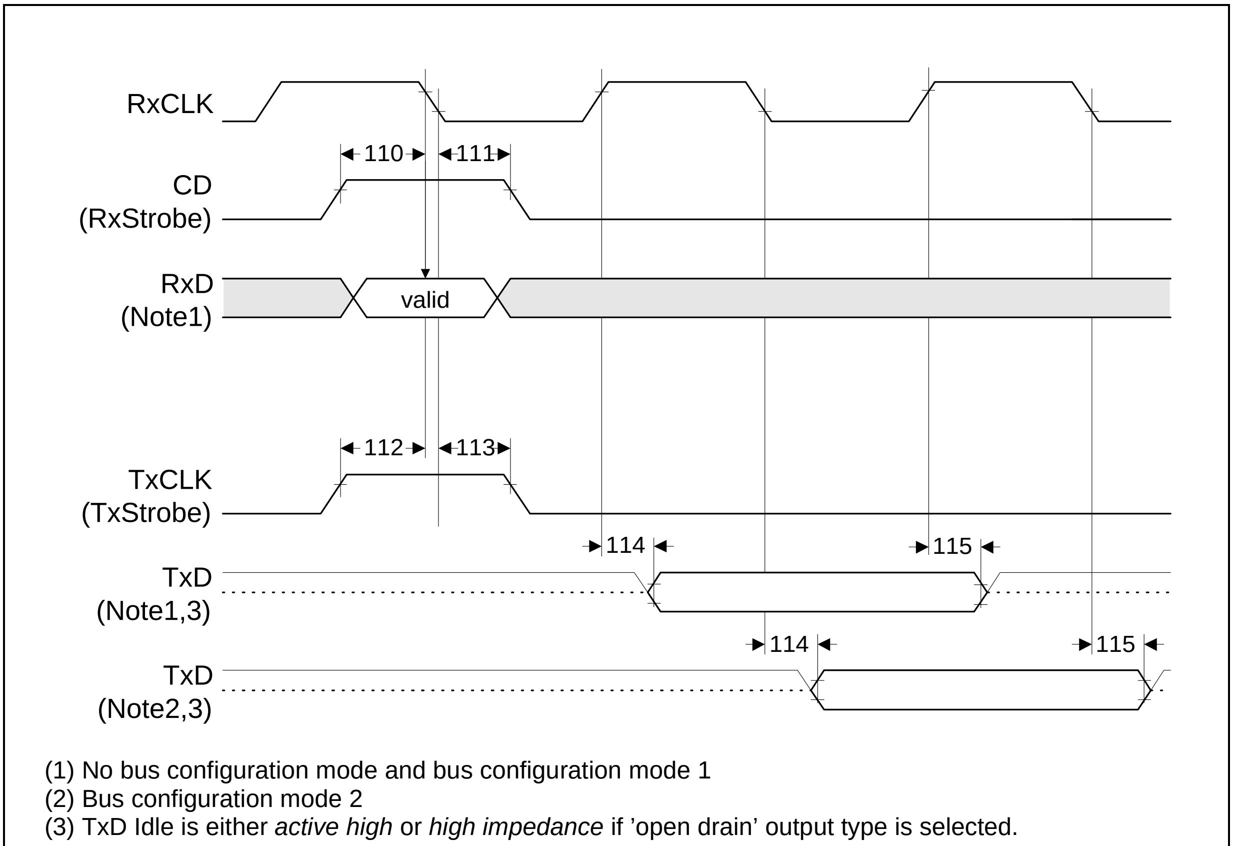


Figure 82 Clock Mode 1 Strobe Timing

Table 29 Clock Mode 1 Strobe Timing

No.	Parameter	Limit Values		Unit
		min.	max.	
110	Receive strobe to RxCLK setup	5		ns
111	Receive strobe to RxCLK hold	5		ns
112	Transmit strobe to RxCLK setup	5		ns
113	Transmit strobe to RxCLK hold	5		ns
114	TxD to RxCLK delay	10	25	ns
115	TxD to RxCLK high impedance delay	10	25	ns

7.7.2.5 Clock Mode 4 Gating Timing

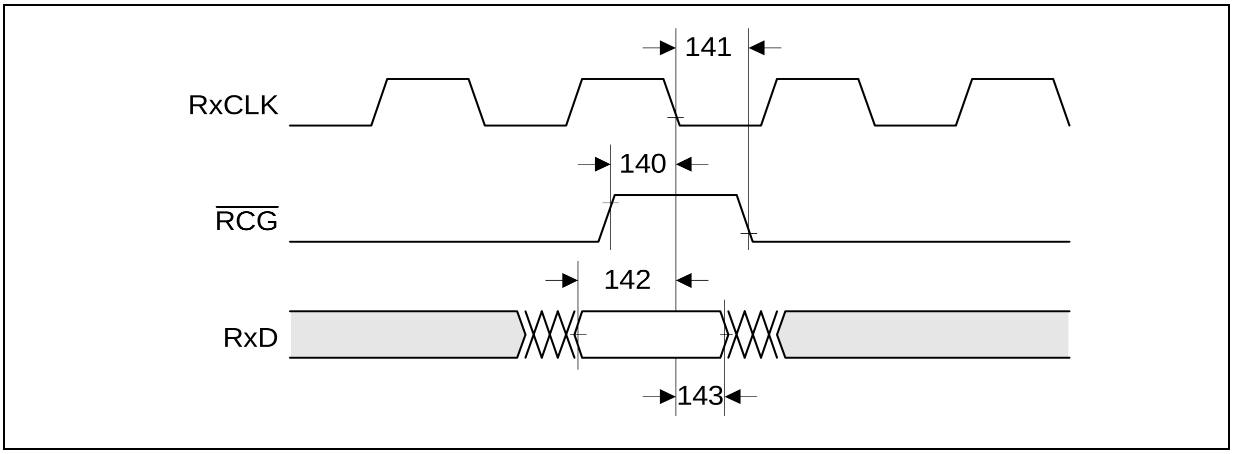


Figure 83 Clock Mode 4 Receive Gating Timing

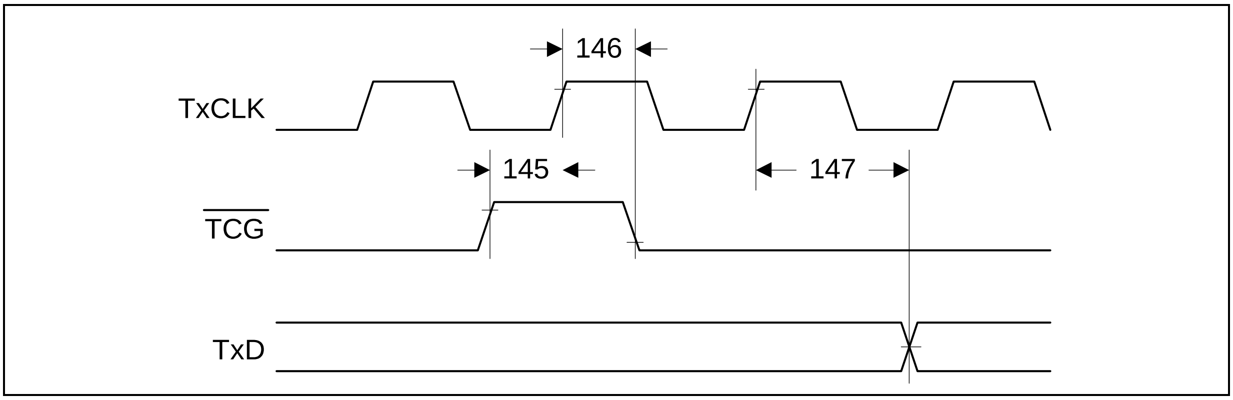


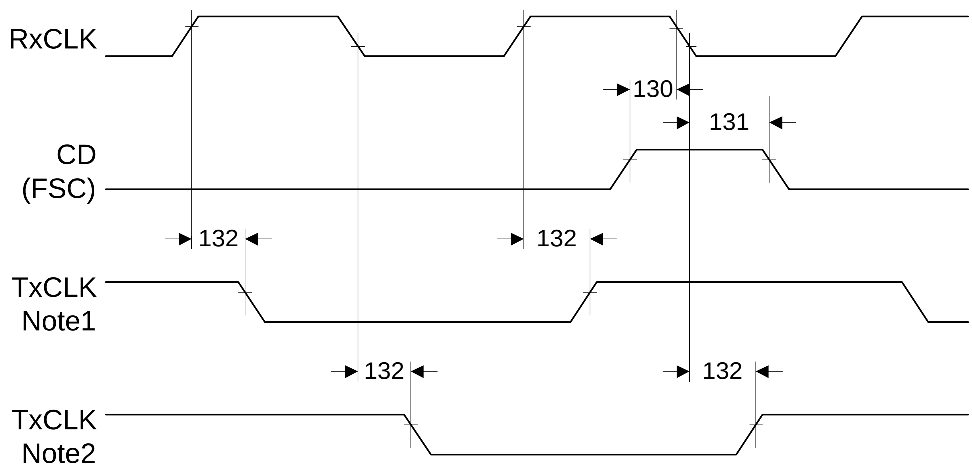
Figure 84 Clock Mode 4 Transmit Gating Timing

Table 30 Clock Mode 4 Gating Timing

No.	Parameter	Limit Values		Unit
		min.	max.	
140	$\overline{\text{RCG}}$ setup time	5		ns
141	$\overline{\text{RCG}}$ hold time	5		ns
142	RxD setup time	5		ns
143	RxD hold time	5		ns
145	$\overline{\text{TCG}}$ setup time	0		ns
146	$\overline{\text{TCG}}$ hold time	6		ns
147	TxCLK to TxD delay[1)]	10	25	ns

[1)] Note that the TxD output is delayed for one additional clock with respect to the gating signal $\overline{\text{TCG}}$!

7.7.2.6 Clock Mode 5 Frame Synchronisation Timing



(1) Normal operation and bus configuration mode 1
(2) Bus configuration mode 2

Figure 85 Clock Mode 5 Frame Synchronisation Timing

Table 31 Clock Mode 5 Frame Synchronisation Timing

No.	Parameter	Limit Values		Unit
		min.	max.	
130	Sync pulse to RxCLK setup time	10		ns
131	Sync pulse to RxCLK hold time	0		ns
132	TxCLKout to RxCLK delay (time slot monitor)	10	27	ns

7.7.3 Reset Timing

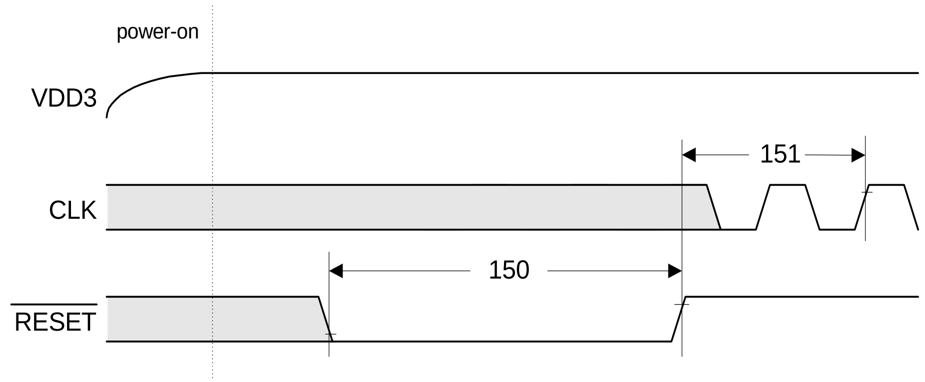


Figure 86 Reset Timing

Table 32 Reset Timing

No.	Parameter	Limit Values		Unit
		min.	max.	
150	$\overline{\text{RESET}}$ pulse width	500		ns
151	Number of CLK cycles after $\overline{\text{RESET}}$ inactive	2		CLK cycles

Note: $\overline{RESET}$ may be asserted and deasserted asynchronous to CLK at any time.

7.7.4 JTAG-Boundary Scan Timing

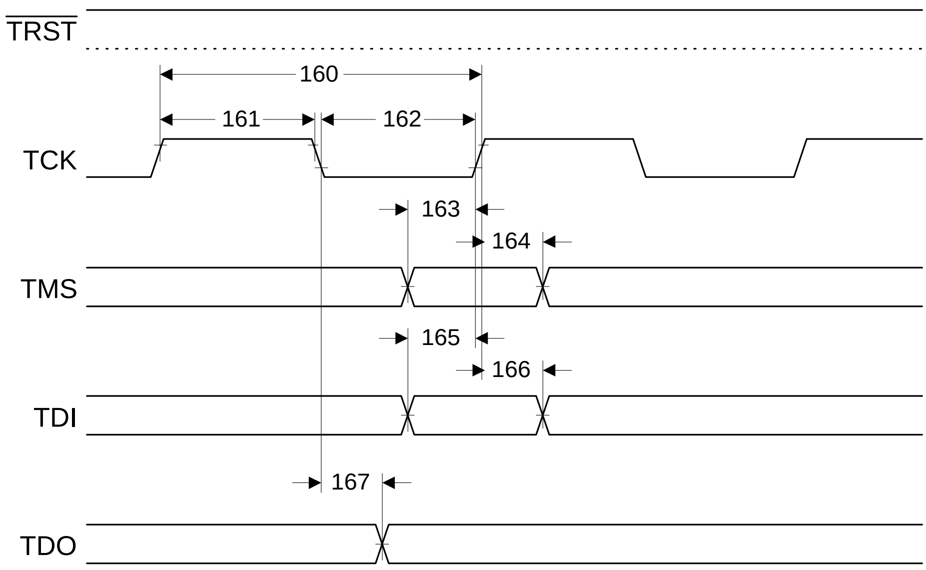


Figure 87 JTAG-Boundary Scan Timing

Table 33 JTAG-Boundary Scan Timing

No.	Parameter	Limit Values		Unit
		min.	max.	
160	TCK period	166	∞	ns
161	TCK high time	80		ns
162	TCK low time	80		ns
163	TMS setup time	30		ns
164	TMS hold time	10		ns
165	TDI setup time	30		ns
166	TDI hold time	20		ns
167	TDO valid delay		60	ns

8 Test Modes

8.1 JTAG Boundary Scan Interface

In the SEROCCO-D a Test Access Port (TAP) controller is implemented. The essential part of the TAP is a finite state machine (16 states) controlling the different operational modes of the boundary scan. Both, TAP controller and boundary scan, meet the requirements given by the JTAG standard: IEEE 1149.1. **Figure 88** gives an overview about the TAP controller.

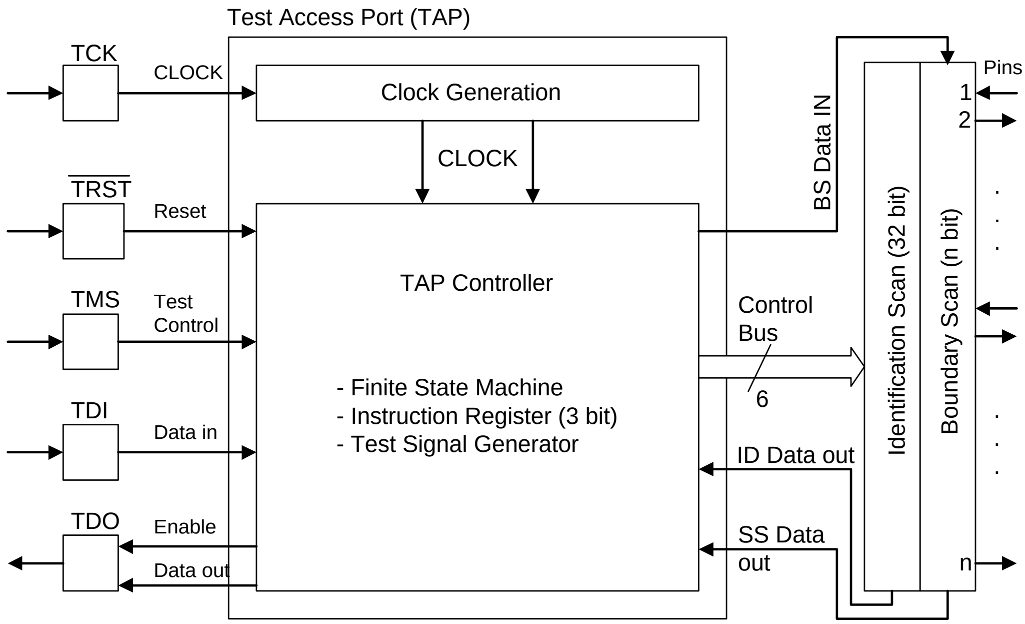


Figure 88 Block Diagram of Test Access Port and Boundary Scan Unit

If no boundary scan operation is planned $\overline{\text{TRST}}$ has to be connected with V_{SS}. TMS, TCK and TDI do not need to be connected since pull-up transistors ensure high input levels in this case. Nevertheless it would be a good practice to put these unused inputs to defined levels, using pull-up resistors.

Test handling (boundary scan operation) is performed via the pins TCK (Test Clock), TMS (Test Mode Select), TDI (Test Data Input) and TDO (Test Data Output) when the TAP controller is not in its reset state, i.e. $\overline{\text{TRST}}$ is connected to V_{DD} or it remains unconnected due to its internal pull-up. Test data at TDI are loaded with a 4-MHz clock signal connected to TCK. ‘1’ or ‘0’ on TMS causes a transition from one controller state to another; constant ’1’ on TMS leads to normal operation of the chip.

Table 34 Boundary Scan Sequence of SEROCCO-D

Seq. No.	Pin	I/O	Number of Boundary Scan Cells	Constant Value In, Out, Enable
	TDI –>			
1	$\overline{\text{CTSB}}$	I	1	**0**
2	$\overline{\text{CTSA}}$	I	1	**0**
3	CDA	I	1	**1**
4	RxDA	I	1	**0**
5	RxCLKA	I	1	**0**
6	TxDA		2	**00**
7	TxCLKA		3	**000**
8	$\overline{\text{RTSA}}$	O	1	**0**
9	$\overline{\text{RESET}}$	I	1	**0**
10	$\overline{\text{INT}}$	O	2	**01**
11	A15	I/O	3	**011**
12	A14	I/O	3	**110**
13	A13	I/O	3	**000**
14	A12	I/O	3	**010**
15	A11	I/O	3	**000**
16	A10	I/O	3	**001**
17	A9	I/O	3	**1**00
18	A8	I/O	3	000
19	A7	I/O	3	000
20	A6	I/O	3	000
21	A5	I/O	3	000
22	A4	I/O	3	000
23	A3	I/O	3	000
24	A2	I/O	3	000
25	A1	I/O	3	000
26	A0	I/O	3	000

Table 34 Boundary Scan Sequence of SEROCCO-D

Seq. No.	Pin	I/O	Number of Boundary Scan Cells	Constant Value In, Out, Enable
27	BM	I	1	0
28	$\overline{CS}$	I	1	0
29	$\overline{BHE}$	I/O	3	000
30	W/$\overline{R}$	I/O	3	000
31	A16	O	2	00
32	A17	O	2	00
33	A18	O	2	00
34	A19	O	2	00
35	A20	O	2	00
36	A21	O	2	00
37	A22	O	2	00
38	A23	O	2	00
39	$\overline{RD}$	I/O	3	000
40	$\overline{WR}$	I/O	3	000
41	$\overline{READY}$	I/O	3	000
42	CLK	I	1	0
43	D0	I/O	2	00
44	D1	I/O	2	00
45	D2	I/O	2	00
46	D3	I/O	2	00
47	D4	I/O	2	00
48	D5	I/O	2	00
49	D6	I/O	2	00
50	D7	I/O	3	000
51	D8	I/O	2	00
52	D9	I/O	2	00
53	D10	I/O	2	00
54	D11	I/O	2	00
55	D12	I/O	2	00
56	D13	I/O	2	00

Table 34 Boundary Scan Sequence of SEROCCO-D

Seq. No.	Pin	I/O	Number of Boundary Scan Cells	Constant Value In, Out, Enable
57	D14	I/O	2	00
58	D15	I/O	3	000
59	$\overline{\text{BREQ}}$	I/O	3	000
60	$\overline{\text{BGNT}}$	I	1	0
61	$\overline{\text{BGACK}}$	I/O	3	000
62	GP1	I/O	3	000
63	GP0	I/O	3	000
64	GP2	I/O	3	000
65	$\overline{\text{RTSB}}$	O	1	0
66	RxDB	I	1	0
67	RxCLKB	I	1	0
68	TxDB	O	2	00
69	TxCLKB	I/O	3	000
70	CDB	I	1	0
71	$\overline{\text{ADS}}$	O	2	00

–> *TDO*

An input pin (I) uses one boundary scan cell (data in), an output pin (O) uses two cells (data out, enable) and an I/O-pin (I/O) uses three cells (data in, data out, enable). Note that some functional output and input pins of SEROCCO-D are tested as I/O pins in boundary scan, hence using three cells. The boundary scan unit of SEROCCO-D contains a total of n = 158 scan cells.

The right column of **Table 34** gives the initialization values of the cells.

The desired test mode is selected by serially loading a 3-bit instruction code into the instruction register via TDI (LSB first); see **Table 35.**

Table 35 Boundary Scan Test Modes

Instruction (Bit 2 … 0)	Test Mode
000	EXTEST (external testing)
001	INTEST (internal testing)
010	SAMPLE/PRELOAD (snap-shot testing)
011	IDCODE (reading ID code)
111	BYPASS (bypass operation)
others	handled like BYPASS

EXTEST is used to examine the interconnection of the devices on the board. In this test mode at first all input pins **capture** the current level on the corresponding external interconnection line, whereas all output pins are held at constant values ('0' or '1', according to **Table 34**). Then the contents of the boundary scan is **shifted** to TDO. At the same time the next scan vector is loaded from TDI. Subsequently all output pins are **updated** according to the new boundary scan contents and all input pins again capture the current external level afterwards, and so on.

INTEST supports internal testing of the chip, i.e. the output pins **capture** the current level on the corresponding internal line whereas all input pins are held on constant values ('0' or '1', according to **Table 34**). The resulting boundary scan vector is **shifted** to TDO. The next test vector is serially loaded via TDI. Then all input pins are **updated** for the following test cycle.

Note: In capture IR-state the code '001' is automatically loaded into the instruction register, i.e. if INTEST is wanted the shift IR-state does not need to be passed.

SAMPLE/PRELOAD is a test mode which provides a snap-shot of pin levels during normal operation.

IDCODE: A 32-bit identification register is serially read out via TDO. It contains the version number (4 bits), the device code (16 bits) and the manufacturer code (11 bits). The LSB is fixed to '1'.

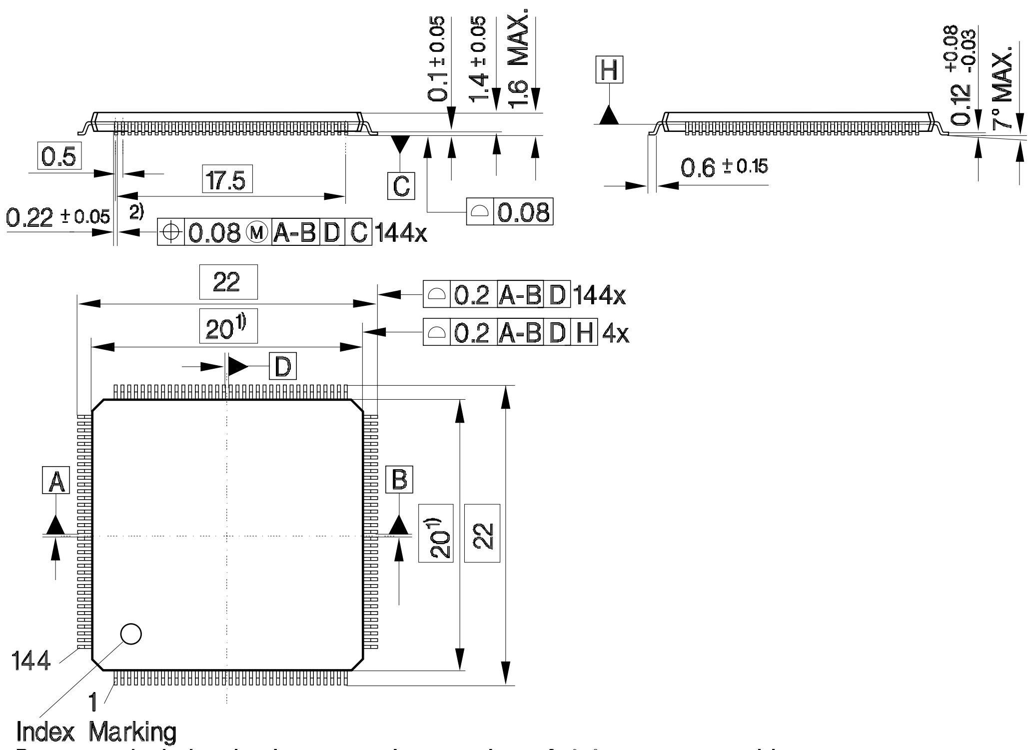
TDI ->	0010	0000 0000 0101 1110	0000 1000 001	1	-> TDO

Note: Since in test logic reset state the code '011' is automatically loaded into the instruction register, the ID code can easily be read out in shift DR state which is reached by TMS = 0, 1, 0, 0.

BYPASS: A bit entering TDI is shifted to TDO after one TCK clock cycle.

9 Package Outlines

P-TQFP-144-10
(Plastic Thin Quad Flat Package)



1) Does not include plastic or metal protrusion of 0.25 max. per side
2) Does not include dambar protrusion of 0.08 max. per side

GPP09243

Sorts of Packing
Package outlines for tubes, trays etc. are contained in our Data Book “Package Information”.

Dimensions in mm