

Enhanced Serial Communication Controller with 2/8 Channels ESCC2/ESCC8

SAB 82532 Version 3.2A and SAB 82538 Version 3.1

Addendum 02.98

This document supplements and clarifies the latest ESCC2 User's Manual 07.96 and the ESCC8 Delta Sheet 02.97.

This Addendum provides additional information concerning

- the clocking concept
- DMA operation and timings
- serial interface timings
- clock timings unique to the ESCC8

All statements applicable to the ESCC2 also apply to the ESCC1.

1 Clocking Concept Overview

1.1 Clock Domains

The internal structure of each ESCC channel consists of 3 clocking domains, transmit, receive, and core. These three function blocks are clocked with internal transmit frequency f_{TRM} , internal receive frequency f_{REC} and control logic frequency f_{CORE} , respectively (control logic means any parts which are not dedicated transmitter or receiver or register access). The internal FIFO interfaces are used to transfer data between the different clock domains. FIFO and register accesses via the asynchronous μ C-interface do not need any clock edges. f_{CORE} clocks data between the registers and the core state machines. f_{CORE} is also used to copy data between the shadow FIFOs and the serial side FIFOs as well as for interrupt indications.

Clock Domains

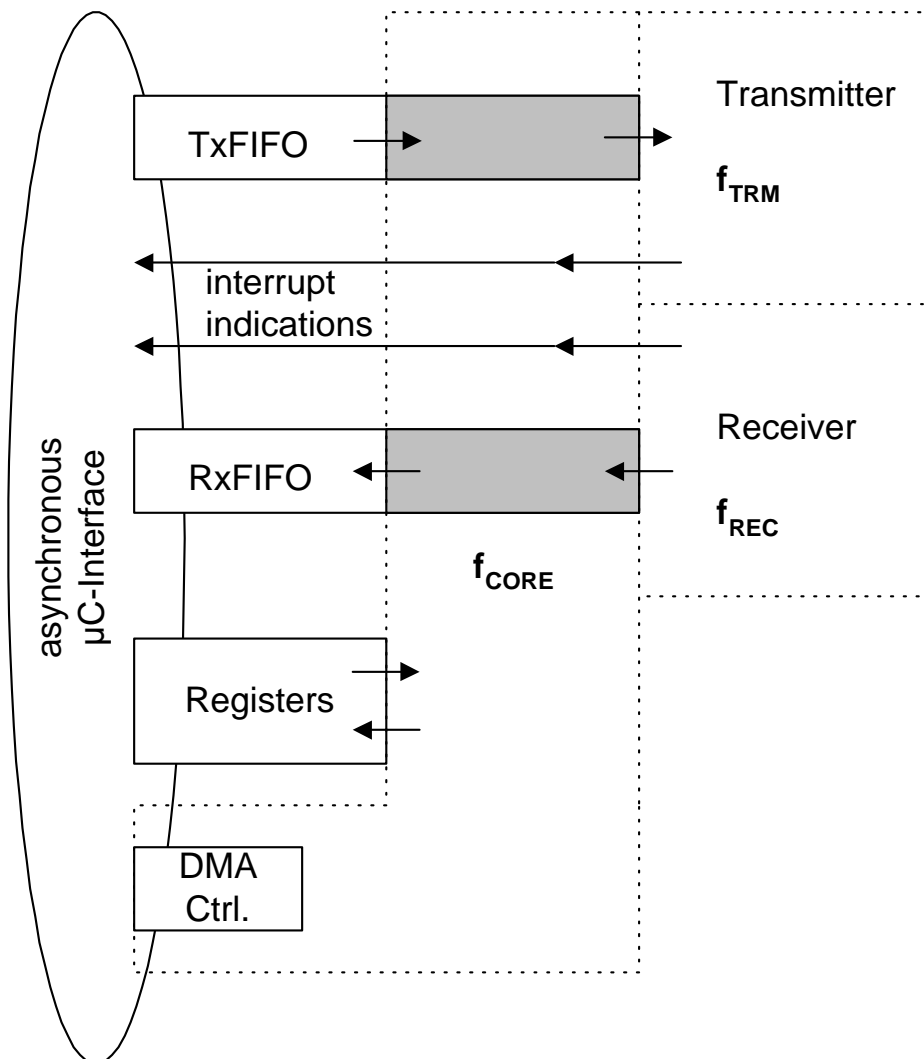


Figure 1 Clock Domains

These clocks are internal clocks only and need not be identical to external clock inputs e.g. f_{TRM} and TxCLK input pin.

The different sources of the internal clocks are described in chapter 1.2.

1.2 Internal Clocking Concept

To assure flexibility for various applications, the ESCCx provides many possibilities to derive or generate the internal clocks from external sources. The different configurations are called 'clock modes' and are described in the following table taken from the data sheet, modified with additional column 'CORE':

Table 1 Clock Modes of ESCC2

Channel Configuration		Clock Sources					Control Sources				
Clock Mode CCR1: CM2, CM1,CM0	CCR2: SSEL	CORE	BRG	DPLL	REC	TRM	CD	R- Strobe	X- Strobe	Frame-Sync	Output via TxCLK (if CCR2: TOE = '1')
0a	0	TRM CCR0.MCE='0'; CCR0.MCE='1' and CCR4.MCK4='0': OSC CCR0.MCE='1' and CCR4.MCK4='1': OSC/4	–	–	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		–	–	OSC	OSC	CD	--	--	–	OSC
5	X		–	–	RxCLK	RxCLK	–	--(TSAR)	--(TSAX)	CD	TS-Control
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	

The first two columns list all possible clock modes configured via bits CCR1.CMi and CCR2.SSEL.

For example, clock mode 6b is chosen by writing a '6' to bits CCR1.CMi and by setting bit CCR2.SSEL equal to 1. The following 5 columns (grouped as 'Clock Sources') specify the source of the internal clocks. Columns CORE, REC and TRM correspond to the domain clocks f_{CORE} , f_{REC} and f_{TRM} described in figure 1.

An example of how to read the table follows:

For clock mode 6b (row '6b') the TRM clock (column 'TRM') is derived from the baudrate generator 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.

And finally the source of f_{CORE} (column 'CORE') depends only on bits CCR0.MCE and CCR4.MCK4. In normal operation (Master Clock Enable MCE='0') the core logic is supplied from the internal transmit clock f_{TRM} . However in some applications use of f_{TRM} to stimulate f_{CORE} is inappropriate. Thus an alternative source for f_{CORE} can be chosen by setting bit CCR0.MCE equal to 1 and is called "master clock mode". In this case the clock f_{CORE} is derived from the oscillator output and thus independent from the internal transmit clock f_{TRM} . f_{CORE} may be limited to a value less than the oscillator frequency, in which case f_{OSC} can be divided by 4 to supply f_{CORE} .

The division is controlled by setting bit CCR4.MCK4 = '1'. This allows running the oscillator up to its maximum frequency while still meeting the limitations for f_{CORE} in master clock operation.

The clocking concept is illustrated in a block diagram manner in the following figure 2:
Clocking Concept

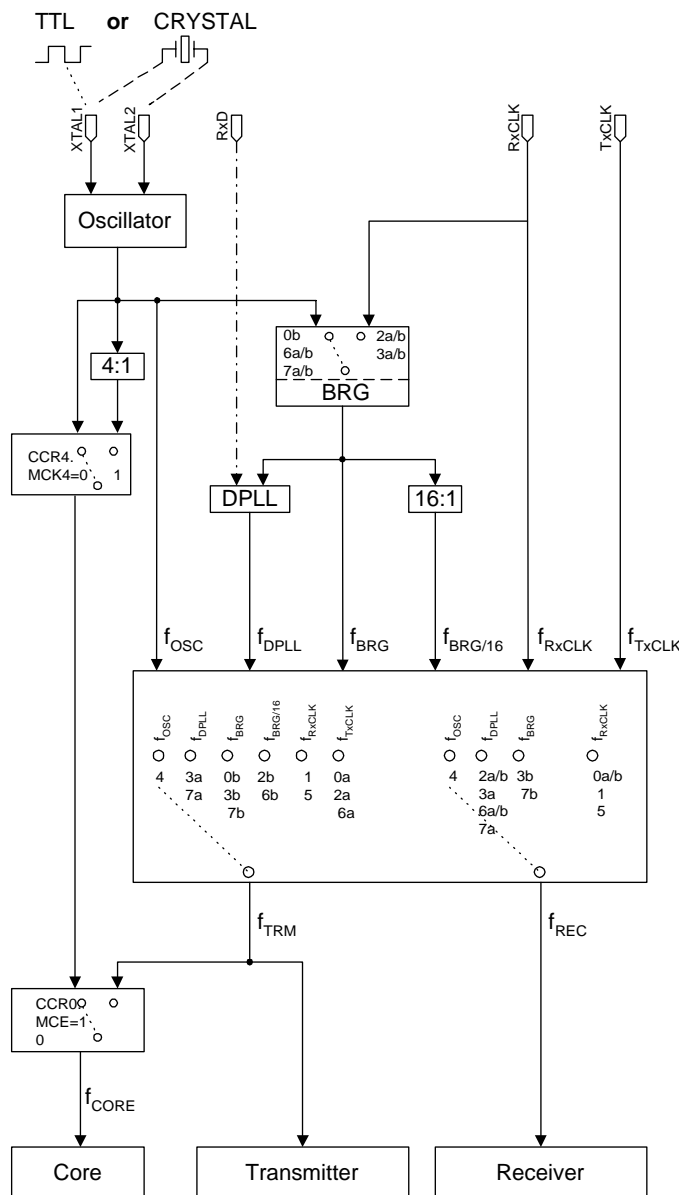


Figure 2 Clocking Concept

Figure 2 shows diagrammatically how the internal clocks f_{CORE} , f_{TRM} and f_{REC} are derived from external clock sources depending on the clock mode control bits.

This drawing corresponds to table 1. The serial channels are totally independent with the exception that they share one common XTAL1, XTAL2 oscillator clock supply.

Note: To reduce complexity, an ASYNC mode specific feature (Bit Clock Rating, BCR) is not included in figure 2. This option introduces an additional divider by 16 to f_{TRM} and f_{REC} depending on the chosen clock mode and is selected by bit 'BCR' in register CCR1 (ASYNC mode only).

The effect of setting bit BCR equal '1' is as follows:

|

Clock Mode	f_{REC}	f_{TRM}
0a	RxCLK/16	TxCLK
0b	RxCLK/16	BRG
1	RxCLK/16	RxCLK/16
3b, 7b	BRG/16	BRG/16
4	OSC/16	OSC/16

When BCR is chosen, oversampling (3 samples) in conjunction with majority decision is performed.

BCR has no effect when using clock mode 2, 3a, 6a, 6b, or 7a.

1.3 Master Clock Operation

There are only three cases where master clock operation is needed.

- a) Applications in which the transmitter runs with a very slow transmit frequency compared to the receiver clock speed. In this case, command execution, FIFO operation, interrupt indication, and operation of the internal state machines is slow because of f_{CORE} equal to f_{TRM} in non master clock mode. Slow operation of the state machines may adversely impact servicing the fast receiver.
- b) Some applications realize flow control or special TDM solutions by clock gapping. This is possible because the ESCCx is a static design thus the transmitter will stay stable in its current state while f_{TRM} is gapped and continue with the next clock edge. Nevertheless the core will be stopped too while f_{TRM} is gapped not performing any FIFO transactions or responding on register accesses like command bits in the CMDR register. Especially receive data cannot be transferred from the receiver into the RFIFO.
- c) In some applications the CPU handles the ESCCx in a polling manner e.g. waiting for "Command Execution" (CEC) bit after any command. In non master clock mode a slow f_{TRM} results in long command execution times slowing down CPU operation.

The master clock capability was introduced to meet these very special applications. However, use of master clock operation also introduces some additional limitations:

Frequency f_{CORE} is limited to a maximum of 2 or 10 MHz, depending on the speed version of the ESCCx, although the internal crystal oscillator is capable of much higher frequency operation. An internal divide by 4 circuit between the oscillator f_{OSC} and f_{CORE} , controlled by bit CCR4.MCK4, satisfies the limitations on f_{CORE} and yet allows maximum f_{osc} .

In master clock mode f_{CORE} is different from f_{TRM} . This difference causes additional frequency ratio limitations described in Note 2 and 3 under table 5 'Clock Modes of ESCC2' and notes under table 5 'Clock Modes of ESCC8' in the data sheets.

It is recommended not to use master clock beyond the three cases described above.

Note: If clock mode 5 (time slot operation) is selected, the master clock mode option cannot be used.

2 Microprocessor Interface

2.1 Mixed Byte/Word Access

There are mistakes in table 1 and 2 of the **ESCC2 User's Manual 7/96**. The correct tables are as follows:

Table 2
Data Bus Access (16-Bit Intel Mode)

BHE	A0	Register Access	ESCC2 Data Pins Used
0	0	FIFO word access Register word access (even addresses)	D0 – D15
0	1	Register byte access (odd addresses)	D8 – D15
1	0	Register byte access (even addresses)	D0 – D7
1	1	No transfer performed	None

Table 3
Data Bus Access (16-Bit Motorola Mode)

BLE	A0	Register Access	ESCC2 Data Pins Used
0	0	FIFO word access Register word access (even addresses)	D0 – D15
0	1	Register byte access (odd addresses)	D0 – D7
1	0	Register byte access (even addresses)	D8 – D15
1	1	No transfer performed	None

The assignment of registers with even/odd addresses to the data lines in case of 16-bit register access depends on the selected microprocessor interface mode:

Siemens/Intel	(Adr. n + 1)	(Adr. n)
Motorola	(Adr. n)	(Adr. n + 1)
	↑	↑
	↓	↓

Data Lines	D15	D8	D7	D0
------------	-----	----	----	----

n: even address

2.2 DMA-Mode

The timing of the DMA control signals DRTn and DRRn is described in section 11.4.1. Depending on specific bus timing and FIFO access sequences unexpected behavior may occur and is described in this section.

Transmit FIFO DMA access (Intel bus mode):

As a common rule the ESCC will de-assert a transmit DMA request (DRTn) signal as soon as the last TFIFO access is expected as the ESCCx monitors the bus control signals.

(References to figures apply to the ESCC2 Data Sheet, ESCC8 Data Sheet references in brackets)

Figure 3 describes a continuous access to the transmit FIFO. The $(\overline{CS} + \overline{DACK})$ pulse width is equal to the \overline{WR} strobe. In this case the ESCC2 will expect last TFIFO access with the falling edge $(\overline{CS} + \overline{DACK})$. (This behavior is also described in figure 57(55) of the data sheet.)

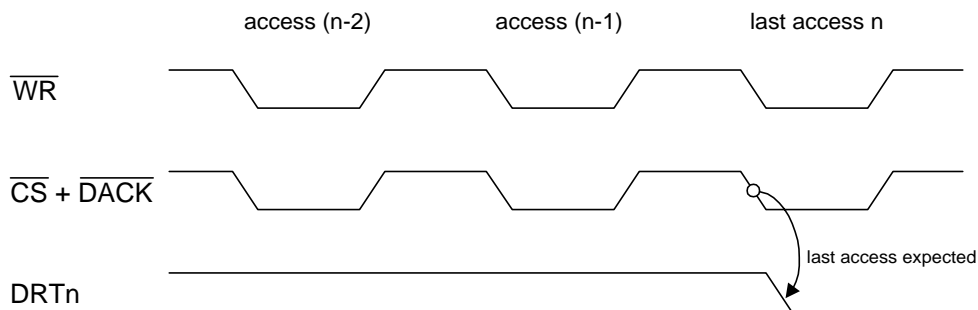


Figure 3

Figure 4 describes a continuous access to the transmit FIFO and the rising edge of ($\overline{CS} + \overline{DACK}$) pulse is behind the corresponding \overline{WR} strobe edge. In this case the ESCCx will expect last TFIFO access already with the rising edge of the next to the last \overline{WR} strobe because ($\overline{CS} + \overline{DACK}$) is still active. When ($\overline{CS} + \overline{DACK}$) is deasserted the ESCCx request the missing FIFO access by asserting DRT again. Now the last access is expected with the falling edge of ($\overline{CS} + \overline{DACK}$) as described in figure 3. This behavior often appears as a glitch on the DRT signal and may cause malfunction to the DMA controller device.

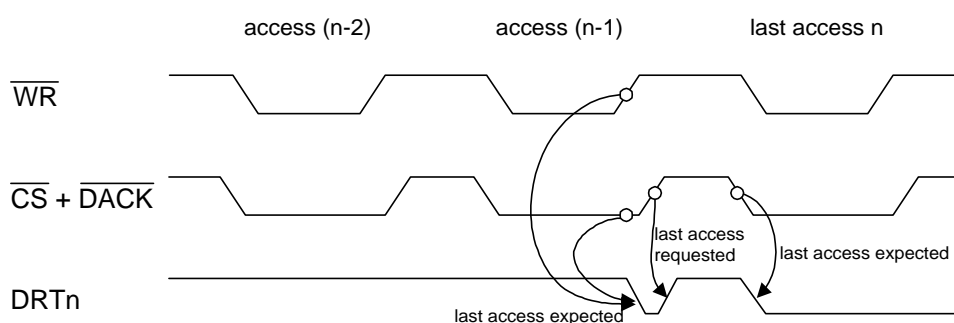


Figure 4

Figure 5 describes a continuous access to the transmit FIFO while ($\overline{CS} + \overline{DACK}$) remains active until end of the last data transfer. In this case the ESCCx will expect last FIFO access already with the rising edge of the next to the last \overline{WR} strobe and de-assert DRT. DRT will remain inactive if the last FIFO access is performed before ($\overline{CS} + \overline{DACK}$) becomes inactive again.

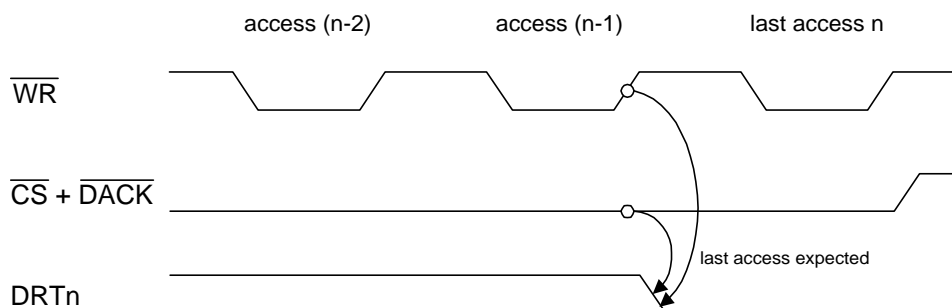


Figure 5

Figure 6 describes an access to the transmit FIFO which is interrupted by an access to another address e.g. a register. If this access is located before the last TFIFO access the ESCCx will expect the last TFIFO access with the falling edge of the \overline{WR} strobe and de-assert DRT. After evaluating the address which is not the expected transmit FIFO

address the ESCC will assert DRT again to request the missing last TFIFO access. DRT becomes inactive with the falling edge of the next \overline{WR} strobe again expecting the last TFIFO access. (This behavior is mentioned in Note 2 of figure 57(55) in the Data Sheet.)

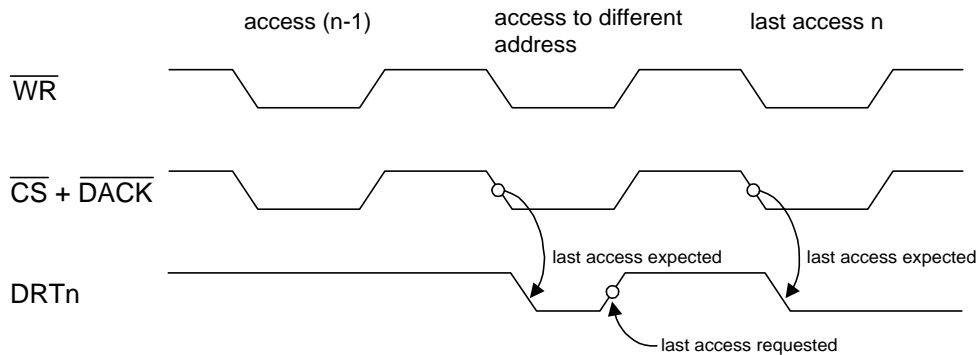


Figure 6

Receive FIFO DMA access (Intel bus mode):

The ESCCx will de-assert the receive FIFO DMA request ($DRRn$) during the last read access to the RFIFO (Note 2 of figure 56(54) in the Data Sheet). $DRRn$ is de-asserted after RFIFO address decoding triggered by the falling edge of the \overline{RD} strobe. Therefore $DRRn$ is independent of the $(\overline{CS} + \overline{DACK})$ timing and is not effected by register accesses in between. $DRRn$ remains active until the last RFIFO access is performed.

Motorola bus mode:

If the device is working in Motorola bus mode the data strobe signal \overline{DS} takes over functionality of the read/write strobes (\overline{WR} , \overline{RD}) in Intel bus mode. The behavior of the DMA control signals is the same respectively. Thus all figures above also apply to the Motorola bus mode if \overline{RD} and \overline{WR} signal names are replaced by data strobe \overline{DS} .

In this case the R/\overline{W} signal is not illustrated because it has no influence on the DMA signal timing (in Motorola bus mode). Its value is expected to be high during read cycles and low during write cycles as described in the "Motorola Bus Interface Mode" section in the data sheets.

3 Serial Interface

3.1 Clock Mode 1 Timing

ESCC2 timing notations (ESCC8 timing numbers in brackets)

Note: ERRATA:

Timing 87 (87), transmit data delay from strobe, the max. value is 7 transmit clock cycles.

Note: The transmit data delay from strobe (timing 87(87)) is synchronized by an internal state machine. Thus the delay depends on the current state of this state machine when the strobe signal becomes active and therefore is not predictable.

The transmit strobe of clock mode 1 is not suitable as a kind of clock gating signal or proprietary TDM application where a determined relationship between strobe signal and transmit data is required. It is recommended to use clock mode 5 in these applications.

The receive strobe behavior is as described in the User's Manual figure 69 (67) and suitable for bitwise receive data filtering.

4 ESCC8 only: Correct Clock Timing Limitations

4.1 Clock Input Timing

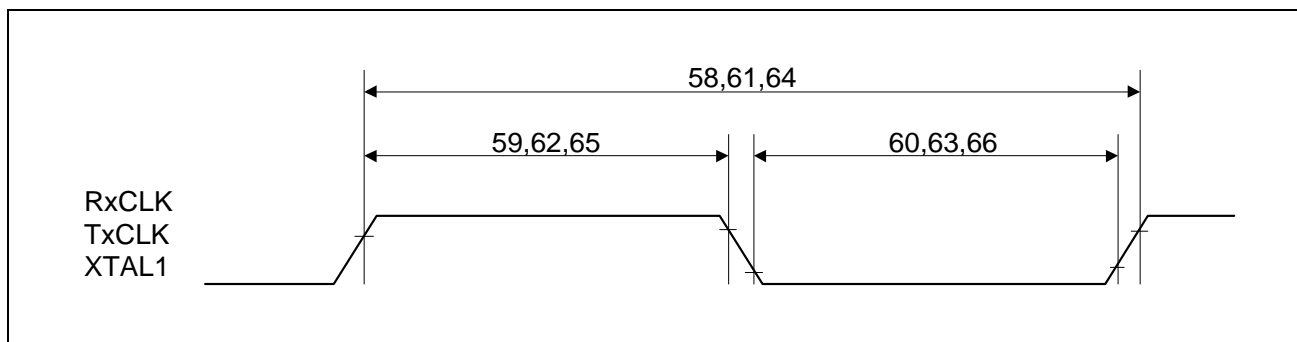


Figure 7

Table 4 Clock Timing

No.	Parameter	Symbol	Limit Values				Unit
			H		N-10		
			min.	max.	min.	max.	
58	RxCLK clock period	$t_{c(RxC)}$	480 ¹⁾ 52 ³⁾		100 ¹⁾ 52 ³⁾		
59	RxCLK high time	$t_{w(RxCH)}$	150 ¹⁾ 27 ³⁾		45 ¹⁾ 27 ³⁾		
60	RxCLK low time	$t_{w(RxCL)}$	150 ¹⁾ 13 ³⁾		45 ¹⁾ 13 ³⁾		
61	TxCLK clock period	$t_{c(TxC)}$	480		100		
62	TxCLK high time	$t_{w(TxCH)}$	150		45		
63	TxCLK low time	$t_{w(TxCL)}$	150		45		
64	XTAL1 clock period	$t_{c(XTAL1)}$	480 ²⁾ 75 ³⁾		75 ²⁾ 75 ³⁾		
65	XTAL1 high time	$t_{w(XTAL1H)}$	150 ²⁾ 37 ³⁾		37 ²⁾ 37 ³⁾		
66	XTAL1 low time	$t_{w(XTAL1L)}$	150 ²⁾ 37 ³⁾		37 ²⁾ 37 ³⁾		

1) Externally clocked: clock mode 0, 1 except ASYNC, BCR = 16.

2) Externally clocked: clock mode 4 except ASYNC, BCR = 16;
Master clock mode generally.

3) Internally clocked: HDLC, BISYNC: DPLL + baud rate generator used; ASYNC all other clocking modes.

Maximal Data Rates if the DPLL is used:

The maximum achievable bit rates in internal timing modes (where bit timing is extracted from the incoming data by the ESCC8 by means of the internal DPLL) are:

- a) 1.2 MBit/s when RxCLK is used as clock source for the baud rate generator (clock modes 2, 3); refer to timing No. 58.
- b) 0.8 MBit/s when XTAL1 (or XTAL1 and XTAL2) is used to supply the clock source for the baud rate generator (clock modes 6, 7); refer to timing No. 64.

Maximal oscillator operation:

Oscillator operation is limited to 13.3 MHz (refer to timing No. 64). Nevertheless the limitation of the internal core clock f_{CORE} is 10 MHz H/N-10 versions.

When simultaneous limits occur, the more restrictive limitation applies.