FUJITSU SEMICONDUCTOR

MB91F465XA preliminary datasheet

MB91460 series

European MCU Design Centre (EMDC)
Fujitsu Microelectronics Europe GmbH
Pittlerstrasse 47
63225 Langen, Germany
Version 0.11, File: mb91f465xa_shortspec.doc

Revision History

Version	Date	Remark
0.01	2006-05-30	Initial draft
0.02	2006-05-31	Several updates (feature list, blockdiagram, IO map, etc)
0.03	2006-07-17	Corrected FlexRay interrupt and DMA RN assignment
0.10	2007-01-08	Several corrections and additions
0.11	2007-02-13	Corrected TOC, PLL2 register names, unified flash settings
0.11		Latest revision

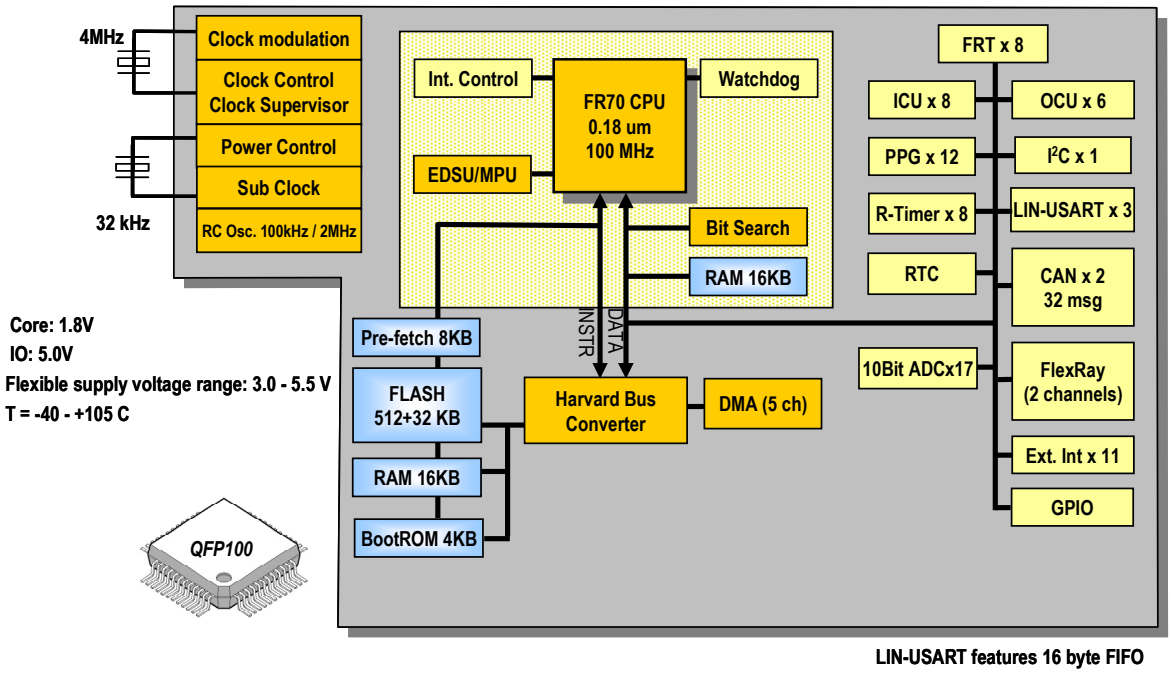
Table of contents

1 Overview

The MB91F465XA is the 100-pin flash MCU of the M91460 family (including FlexRay communication channels A/B).

The corresponding evaluation device is named MB91V460.

1.1 Block Diagram



2 Feature List

2.1 Overview Table

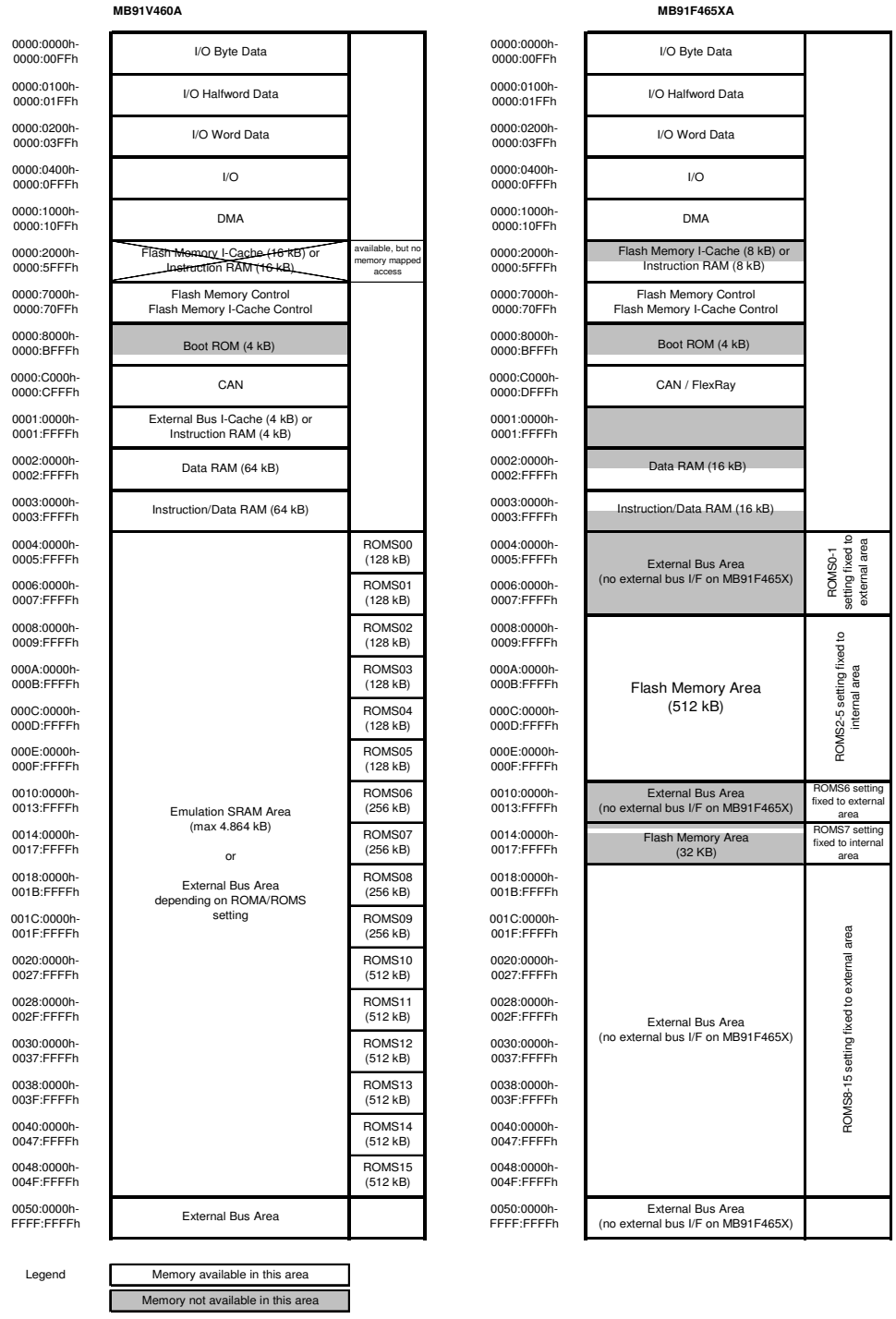
Feature	MB91V460	MB91F465XA	MB91F464AA
Max. Core frequency (CLKB)	80 MHz	100 MHz	80 MHz
Max. Resource frequency (CLKP)	40 MHz	50 MHz	40 MHz
Max. Ext-Bus frequency (CLKT)	40 MHz	-	-
Max. CAN frequency (CLKCAN)	20 MHz	50 MHz	40 MHz
Max. FlexRay frequency (SCLK)	-	80 MHz	-
Watchdog	yes	yes	yes
Bit Search	yes	yes	yes
Reset Input	yes	yes	yes
Clock Modulator	(yes)	yes	yes
DMA	5 ch	5 ch	5 ch
EDSU/MPU	32 BP (16 MPU ch)	16 BP (8 MPU ch)	8 BP (4 MPU ch)
Flash	external	512 KB + 32 KB	384 KB + 32 KB
Satellite Flash	external	-	-
Flash Protection	n.a.	yes	yes
Data RAM	64 KB	16 KB	8 KB
GP RAM	64 KB	16 KB	8 KB
Direct mapped cache	16 KB	8 KB	-
Boot-ROM	4 kB	4 KB	4 KB
RTC	1 ch	1 ch	1 ch
Free Running Timer	8 ch	8 ch	8 ch
ICU	8 ch	8 ch	8 ch
OCU	8 ch	6 ch	6 ch
Reload Timer	8 ch	8 ch	8 ch
PPG	16 ch	12 ch	10 ch

Feature	MB91V460	MB91F465XA	MB91F464AA
PFM	1 ch	-	-
Sound Generator	1 ch	-	-
UpDown Counter	4 ch	-	-
C_CAN	6 ch (128 msg.)	2 ch (32 msg.)	1 ch (32 msg.)
FlexRay	-	2 ch (A/B)	-
LIN-USART	16 ch (4 ch FIFO)	3 ch (3 ch FIFO)	5 ch (1 ch FIFO)
I^2C	4 ch	1 ch	1 ch
FR external bus	32-bit address / 32-bit data / 8 chip select	-	-
External Interrupts	16 ch	11 ch	10 ch
NMI	1 ch	-	-
SMC (quad option)	6 ch	-	-
LCD	1 ch 40x4	-	-
ADC (10-bit)	32 ch	17 ch	21 ch
Alarm Comparator	2 ch	-	-
General Purpose Port I/O (non-muxed)	14	-	-
Low voltage detection	yes	yes	yes
Clock Supervisor	yes	yes	yes
Clock Monitor Output	yes	(yes)[1]	-
Package	BGA-660	LQFP-100	LQFP-100

[1] Clock Monitor Output (MONCLK pin) is shared with pin P17_5 (PPG5)

2.2 Core Functionality

2.2.1 Memory Map



2.2.2 FR70 CPU Core

- 32-bit RISC, load/store architecture, pipeline 5 stages
- Maximum operating frequency: Core clock = 64 MHz to 100 MHz (device dependent)
  (Source oscillation= 4 MHz, multiplied by 16 to 25 (PLL clock multiplier method))
- General-purpose registers: 16 x 32 bits
- 16-bit fixed-length instruction (Base instruction)
- 32-bit linear address space: 4 GBytes
- Instructions suitable for embedded application
  - Transfer command between memories
  - Bit-processing instruction
  - Barrel-shift instructions
- Instructions supporting C-language
- Function's enter command /exit command
- Multi-load/store command of register contents
- Assembler statement is also easily available
  Register's interlock function
- Multiplier's embedded application/command level support
  - Signed 32-bit multiplication: 5 cycles
  - Signed 16-bit multiplication: 3 cycles
- Interrupt (PC/PS are saved): 6 cycles (16 priority level)
- Harvard architecture enables simultaneous execution of program access and data access
- Memory protection function
- Embedded debug support
- Commands compatible with FR family

2.2.3 Instruction Cache

- Direct mapped I-cache
  - 4 kByte integrated
  - Lock function enabling programs to be resident

2.2.4 Interrupt Controller

- A total of 11 external interrupt lines (8 dedicated interrupt pins, 3 interrupt pins shared with peripheral inputs for Wake Up from STOP mode (CAN RX and I^2C SDA)
- Interrupts from internal peripherals (128 interrupt vectors)
- Priority levels programmable for normal interrupt lines excluding the non-maskable one (16

levels)

- Capable of using the normal interrupt pins for Wake Up from STOP mode

2.2.5 External Bus Interface

- An External Bus Interface is not available on MB91F465XA.

2.2.6 DMA Controller

- Four transfer modes supported: single/block, burst, continuous transfer, and fly-by
- 5 channels
- 3 types of transfer sources (external pins/internal peripherals/and software)
- Up to 128 selectable internal transfer sources
- Addressing mode: Specifying up to 32-bit addresses (Increment/decrement/fixed)
- Transfer mode (Demand transfer/burst transfer/step transfer/block transfer)
- Transferred data size selectable from among 8, 16, and 32 bits

2.2.7 Internal Data RAM

- 16 kBytes integrated
- Zero wait state for read/write access of data

2.2.8 Internal GP RAM (program/data)

- 16 kBytes integrated
- Zero wait state for read access of program (code execution)
- One wait state for read/write access of data

2.3 Peripheral Function

- General-purpose port : All functional pins can be used as general-purpose ports, if the corresponding function is not needed.
  - N channel open drain port out of above: 2 (for I^2C)

- A/D converter : 17 channels (1 unit)
  - Series-parallel type
  - Resolution: 10 bits
  - Minimum conversion time: 3µs
  - Single conversion mode
  - Continuous conversion mode
  - Stop conversion mode
  - Activation by software or external trigger can be selected
  - Reload timer 7 and A/D Converter co-operate

- External interrupt input : 8 + 3 channels
  - Can be programmed to be edge sensitive or level sensitive
  - Interrupt mask and request pending bits per channel
  - 2 channels combined with CAN RX for wakeup
  - 1 channel combined with I^2C SDA for wakeup

- Bit search module (using REALOS)
  - Function to search the first bit position of “1”, “0”, “Changed” from MSB (most significant bit) within 1 word

- Reload timer : 16 bits x 8 channels
  - 16-bit reload counter
  - Includes clock prescaler ($f_{RES}/2^1$, $f_{RES}/2^3$, $f_{RES}/2^5$, $f_{RES}/2^6$, $f_{RES}/2^7$)

- Free-run timer : 16 bits x 8 channels
  - 16-bit free running counter, signals an interrupt when overflow or match with compare register
  - Includes prescaler ($f_{RES}/2^2$, $f_{RES}/2^4$, $f_{RES}/2^5$, $f_{RES}/2^6$)
  - Timer data register has R/W access

- PPG : 16 bit x 12 channels
  - 16 bit down counter, cycle and duty setting registers
  - Interrupt at triggering, cycle or duty match
  - PWM operation and one-shot operation
  - Internal prescaler allows $f_{RES}/2^0$, $f_{RES}/2^2$, $f_{RES}/2^4$, $f_{RES}/2^6$ as counter clock
  - Can be triggered by software, reload timer or external trigger event
  - Reload timer 0/1 available as trigger for PPG 0/1/2/3
  - Reload timer 2/3 available as trigger for PPG 4/5/6/7
  - Reload timer 4/5 available as trigger for PPG 8/9/10/11

- Input capture : 16 bits x 8 channels
  - Rising edge, falling edge or rising & falling edge sensitive
  - Free-run timer 0 and input capture 0/1 co-operate
  - Free-run timer 1 and input capture 2/3 co-operate
  - Free-run timer 4 and input capture 4/5 co-operate
  - Free-run timer 5 and input capture 6/7 co-operate

- Output compare : 16 bits x 6 channels
  - Signals an interrupt when a match with of 16-bit IO timer occurs
  - An output signal can be generated
  - Free-run timer 2 and output compare 0/1 co-operate
  - Free-run timer 3 and output compare 2/3 co-operate
  - Free-run timer 6 and output compare 4/5 co-operate

- LIN-USART (LIN=Local Interconnect Network) : 3 channels
  - Full-duplex double buffer system
  - With parity/without parity selectable
  - 1 or 2 stop bits selectable
  - 7 or 8 bits data length selectable
  - NRZ type transfer format
  - Asynchronous /synchronous communications selectable
  - Master-slave communication function (multiprocessor mode)
  - Dedicated baud rate prescaler is embedded in each channel
  - External clock is able to use as transfer clock
  - Parity error, frame error, and overrun error detecting functions

- SPI compatible
- LIN master and slave
- LIN-USART 4 and ICU 4 co-operate (for LIN sync field in slave mode)
- LIN USART 6 and ICU 6 co-operate (for LIN sync field in slave mode)
- LIN USART 7 and ICU 7 co-operate (for LIN sync field in slave mode)

- CAN : 2 channels
  - Supports CAN protocol version 2.0 part A and B
  - Bit rates up to 1 Mbit/s
  - 32 message objects
  - Each message object has its own identifier mask
  - Programmable FIFO mode (concatenation of message objects)
  - Maskable interrupt
  - Disabled Automatic Retransmission mode for Time Triggered CAN applications
  - Programmable loop-back mode for self-test operation

- FlexRay : 2 channels
  - Conformance with FlexRay protocol specification v2.1
  - Data rates of up to 10 Mbit/s on each channel
  - Up to 128 message buffers configurable
  - 8 kBytes of Message RAM for storage of e.g. 128 message buffers with max. 48 Bytes data section or up to 30 message buffers with 254 Bytes data section
  - Configuration of message buffers with different payload lengths possible
  - One configurable receive FIFO
  - Each message buffer can be configured as receive buffer, as transmit buffer or as part of the receive FIFO
  - Host access to message buffers via Input and Output Buffer
    - Input Buffer: Holds message to be transferred to the Message RAM
    - Output Buffer: Holds message read from the Message RAM
  - Filtering for slot counter, cycle counter, and channel
  - Maskable module interrupts
  - Network Management supported

- I^2C (400k fast mode) : 1 channel
  - Master or slave transmission
  - Arbitration function
  - Clock synchronization function

- Slave address and general call address detect function
- Transfer direction detect function
- Start condition repeat generation and detection function
- Bus error detect function
- Compatible to I2C standard and fast mode specification (operation up to 400 kHz, 10 bit addressing)
- Includes clock divider functionality
- SCL and SDA lines include optional noise filter. The noise filter allows the suppression of spikes in the range of 1 to 1.5 cycles of the Peripheral Clock.

- Stepper Motor Controller : 0 channels
  - There are no Stepper Motor Controllers on MB91F465XA.

- Timebase/watchdog timer (26 bits)
  - Adjustable watchdog timer interval (between 2^{20} and 2^{26} system clock cycles)

- Real-time clock (counts during stop mode)
  - RTC module can be clocked either from 32 kHz quartz, 4 MHz quartz or from the RC Oscillator
  - Facility to correct oscillation deviation (subclock calibration)
  - Read/write accessible second/minute/ hour registers
  - Can signal interrupts every halfsecond/second/ minute/hour/day
  - Internal clock divider and prescaler provide exact 1s clock based on a 4 MHz or a 32 kHz clock input
  - Prescaler value for 4 MHz is $1E847F_H$
  - Prescaler value for 32 kHz is $003FFF_H$

- Clock supervisor
  - Monitors external 32kHz and 4MHz for fails (e.g. crystal breaks)
  - Switches in case of fail to an available recovery clock (other oscillator, or RC oscillator)

- Clock modulator (reduction of EME)

- Subclock calibration
  - Calibration of the RTC timer in 32 kHz or RC oscillator operation, based on the more accurate 4 MHz quartz is possible

- Main oscillation stabilisation timer

- 23 bit counter for main oscillation stabilisation wait when running in sub clock mode
- Generates an interrupt when stabilisation time has elapsed

- Sub oscillation stabilisation timer
  - 15 bit counter for sub oscillation stabilisation wait when running in main clock mode
  - Generates an interrupt when stabilisation time has elapsed

2.4 Embedded Program/Data Memory (Flash)

2.4.1 Flash Features

- 512 + 32 kByte Flash
- Power: Single 3.0-5.5V supply
- Programmable wait states for read/write access;
- Read 16/32-bit width, write 16-bit width
- Flash security with security vector at 0x0014:8000 – 0x0014:800F[2]
- Operation modes:
- (1) 32-bit CPU mode:
  - CPU reads and executes programs in word (32-bit) length units.
  - Flash writing is not possible.
  - Actual Flash Memory access is performed in word (32-bit) length units.
- (2) 16-bit CPU mode:
  - CPU reads and writes in half-word (16-bit) length units.
  - Program execution from the Flash is not possible.
  - Actual Flash Memory access is performed in word (16-bit) length units.
- (3) Flash memory mode (external access to Flash memory enabled)
- Features (through combination of Flash memory macro and FR-CPU interface circuit):
  - Functions as CPU program/data storage memory.
  - Enables access to 32-bit bus width.
  - Enables read/write/erase by CPU (auto program algorithm*).
  - Functions equivalent to MBM29LV400TC stand-alone Flash-memory product.
  - Enables read/write/erase by parallel Flash programmer (auto program algorithm*).

    *: Auto program algorithm = Embedded Algorithm TM

[2] See MB91460 hardware manual for further details.

2.4.2 CPU Mode

2.4.2.1 Flash configuration in CPU mode

Flash memory map in CPU mode (MD[2:0] = 00x)

addr									
0014:FFFFh 0014:C000h	SA6 (8kB)				SA7 (8kB)				ROMS7
0014:BFFFh 0014:8000h	SA4 (8kB)				SA5 (8kB)				
0014:7FFFh 0014:4000h	SA2 (8kB)				SA3 (8kB)				
0014:3FFFh 0014:0000h	SA0 (8kB)				SA1 (8kB)				
0013:FFFFh 0012:0000h	SA22 (64kB)				SA23 (64kB)				ROMS6
0011:FFFFh 0010:0000h	SA20 (64kB)				SA21 (64kB)				
000F:FFFFh 000E:0000h	SA18 (64kB)				SA19 (64kB)				ROMS5
000D:FFFFh 000C:0000h	SA16 (64kB)				SA17 (64kB)				ROMS4
000B:FFFFh 000A:0000h	SA14 (64kB)				SA15 (64kB)				ROMS3
0009:FFFFh 0008:0000h	SA12 (64kB)				SA13 (64kB)				ROMS2
0007:FFFFh 0006:0000h	SA10 (64kB)				SA11 (64kB)				ROMS1
0005:FFFFh 0004:0000h	SA8 (64kB)				SA9 (64kB)				ROMS0
	addr+0	addr+1	addr+2	addr+3	addr+4	addr+5	addr+6	addr+7	
16bit read/write	dat[31:16]		dat[15:0]		dat[31:16]		dat[15:0]		
32bit read	dat[31:0]				dat[31:0]				

Legend	
	Memory available in this area
	Memory not available in this area

2.4.2.2 Flash access timing settings in CPU mode

The flash access timing is described in MB91460 Hardware Manual chapter 11. The following tables list all settings for a given Core Frequency for Flash read and write access.

Flash read timing settings for MB91F465XA

Core clock (CLKB)	ATD	ALEH	EQ	WEXH	WTC	Comment
to 24 MHz	0	0	0	-	1	
to 48 MHz	0	0	1	-	2	
to 100 MHz	1	1	3	-	4	

Flash write timing settings for MB91F465XA (synchronous write)

Core clock (CLKB)	ATD	ALEH	EQ	WEXH	WTC	Comment
to 16 MHz	0	-	-	0	3	
to 32 MHz	0	-	-	0	4	
to 48 MHz	0	-	-	0	5	
to 64 MHz	1	-	-	0	6	
to 96 MHz	1	-	-	0	7	
to 100 MHz	1	-	-	1	8	

2.4.2.3 Address mapping from CPU to parallel programming mode

8kB Sectors (SA4 – SA7)

SA4, SA6:

Condition: addr >= 14:8000h && addr <= 14:FFFFh && addr[2]==0 :

FA := addr - addr%00:4000h + (addr%00:4000h)/2 - (addr/2)%4 + addr%4 – 0D:0000h

SA5, SA7:

Condition: addr >= 14:8000h && addr <= 14:FFFFh && addr[2]==1:

FA := addr - addr%00:4000h + (addr%00:4000h)/2 + 00:2000h - (addr/2)%4 + addr%4 – 0D:0000h

64kB Sectors (SA12 – SA19)

SA12, SA14, SA16, SA18:

Condition: addr >= 08:0000h && addr <= 13:FFFFh && addr[2]==0:

FA := addr - addr%02:0000 + (addr%02:0000h)/2 - (addr/2)%4 + addr%4

SA13, SA15, SA17, SA19:

Condition: addr >= 08:0000h && addr <= 13:FFFFh && addr[2]==1:

FA := addr - addr%02:0000h + (addr%02:0000h)/2 + 01:0000h - (addr/2)%4 + addr%4

Remark: FA result is without 10:0000h offset for parallel flash programming[3].

[3] Set offset by keeping FA[20] = 1 as described in section 2.4.2.3.

2.4.3 Parallel flash programming mode

2.4.3.1 Flash configuration in parallel flash programming mode

Parallel Flash programming mode (MD[2:0] = 111)

FA[20:0]	
001F:FFFFh 001F:0000h	SA19 (64kB)
001E:FFFFh 001E:0000h	SA18 (64kB)
001D:FFFFh 001D:0000h	SA17 (64kB)
001C:FFFFh 001C:0000h	SA16 (64kB)
001B:FFFFh 001B:0000h	SA15 (64kB)
001A:FFFFh 001A:0000h	SA14 (64kB)
0019:FFFFh 0019:0000h	SA13 (64kB)
0018:FFFFh 0018:0000h	SA12 (64kB)
0017:FFFFh 0017:E000h	SA7 (8kB)
0017:DFFFh 0017:C000h	SA6 (8kB)
0017:BFFFh 0017:A000h	SA5 (8kB)
0017:9FFFh 0017:8000h	SA4 (8kB)

	FA[1:0]=00	FA[1:0]=10
16bit write mode	DQ[15:0]	DQ[15:0]

Remark: Always keep FA[0] = 0 and FA[20] = 1

2.4.3.2 Pin connections in parallel programming mode

Resetting after setting the MD[2:0] pins to [111] will halt CPU functioning. At this time, the Flash memory's interface circuit enables direct control of the Flash memory unit from external pins by directly linking some of the signals to GP-Ports. Please see table below for signal mapping.

In this mode, the Flash memory appears to the external pins as a stand-alone unit. This mode is generally set when writing/erasing using the parallel Flash programmer. In this mode, all operations of the Flash memory's Auto Algorithms are available.

Correspondence between MBM29LV400TC and Flash Memory Control Signals

MBM29LV400TC External pins	FR-CPU mode	MB91F465XA external pins			Comment
		Flash memory mode	Normal function	Pin number	
-	INITX	-	INITX	52	
RESET	-	FRSTX	P16_7	53	
-	-	MD2	MD_2	99	Set to '1'
-	-	MD1	MD_1	98	Set to '1'
-	-	MD0	MD_0	92	Set to '1'
RY/BY	FMCS:RDY bit	RY/BYX	P24_0	74	
BYTE	Internally fixed to 'H'	BYTEX	P24_2	78	
WE	Internal control signal + control via interface circuit	WEX	P28_3	29	
OE		OEX	P28_2	28	
CE		CEX	P28_1	27	
-		ATDIN	P22_1	73	Set to '0'
-		EQIN	P22_0	72	Set to '0'
-		TESTX	P24_3	79	Set to '1'
-		RDYI	P24_1	77	Set to '0'
A-1	Internal address bus	FA0	P19_2	47	Set to '0'
A0 to A7		FA1 to FA8	P16_0 to P16_3, P27_0 to P27_3	16 to 23	
A8 to A15		FA9 to FA16	P15_0 to P15_5, P18_0, P18_1	68 to 71, 10, 11, 57, 58	
A16 to A18		FA17 to FA19	P18_2, P18_4, P18_5	59 to 61	
A19		FA20	P18_6	62	Set to '1'
DQ0 to DQ7	Internal data bus	DQ0 to DQ7	P17_0 to P17_7	48, 49, 54 to 56, 65 to 67	
DQ8 to DQ15		DQ8 to DQ15	P23_0, P23_1, P31_0 to P31_2, P31_4 to P31_6	2 to 9	

2.4.4 Flash Security

2.4.4.1 Vector addresses

Two Flash Security Vectors (FSV1, FSV2) are located parallel to the Boot Security Vectors (BSV1, BSV2) controlling the protection functions of the flash security module:

FSV1: 0x14:8000 **BSV1: 0x14:8004**

FSV2: 0x14:8008 **BSV2: 0x14:800C**

2.4.4.2 Security Vector FSV1

The setting of the Flash Security Vector FSV1 is responsible for the read and write protection modes and the individual write protection of the 8 kByte sectors.

■ **FSV1 (bits 31 to 16)**

The setting of the Flash Security Vector FSV1 bits [31:16] is responsible for the read and write protection modes.

Explanation of the bits in the Flash Security Vector FSV1[31:16]

FSV1[31:19]	FSV1[18] Write Protection Level	FSV1[17] Write Protection	FSV1[16] Read Protection	Flash Security Mode
set all to '0'	set to '0'	set to '0'	set to '1'	Read Protection (all device modes, except INTVEC mode MD[2:0]="000")
set all to '0'	set to '0'	set to '1'	set to '0'	Write Protection (all device modes, without exception)
set all to '0'	set to '0'	set to '1'	set to '1'	Read Protection (all device modes, except INTVEC mode MD[2:0]="000") and Write Protection (all device modes)
set all to '0'	set to '1'	set to '0'	set to '1'	Read Protection (all device modes, except INTVEC mode MD[2:0]="000")
set all to '0'	set to '1'	set to '1'	set to '0'	Write Protection (all device modes, except INTVEC mode MD[2:0]="000")
set all to '0'	set to '1'	set to '1'	set to '1'	Read Protection (all device modes, except INTVEC mode MD[2:0]="000") and Write Protection (all device modes except INTVEC mode MD[2:0]="000")

■ **FSV1 (bits 15 to 0)**

The setting of the Flash Security Vector FSV1 bits [15:0] is responsible for the individual write protection of the 8 kByte sectors. It is only evaluated if write protection bit FSV1[17] is set.

Explanation of the bits in the Flash Security Vector FSV1[15:0]

FSV1 bit	Sector	Enable Write Protection	Disable Write Protection	Comment
FSV1[0]	-	set to '0'	set to '1'	not available
FSV1[1]	-	set to '0'	set to '1'	not available
FSV1[2]	-	set to '0'	set to '1'	not available
FSV1[3]	-	set to '0'	set to '1'	not available
FSV1[4]	SA4	set to '0'	-	**Write protection is mandatory!**
FSV1[5]	SA5	set to '0'	set to '1'	
FSV1[6]	SA6	set to '0'	set to '1'	
FSV1[7]	SA7	set to '0'	set to '1'	
FSV1[8]	-	set to '0'	set to '1'	not available
FSV1[9]	-	set to '0'	set to '1'	not available
FSV1[10]	-	set to '0'	set to '1'	not available
FSV1[11]	-	set to '0'	set to '1'	not available
FSV1[12]	-	set to '0'	set to '1'	not available
FSV1[13]	-	set to '0'	set to '1'	not available
FSV1[14]	-	set to '0'	set to '1'	not available
FSV1[15]	-	set to '0'	set to '1'	not available

Remark: It is mandatory to always set the sector where the Flash Security Vectors FSV1 and FSV2 are located to write protected (here sector SA4). Otherwise it is possible to overwrite the Security Vector to a setting where it is possible to either read out the flash content or manipulate data by writing.

2.4.4.3 Security Vector FSV2

The setting of the Flash Security Vector FSV2 bits [31:0] is responsible for the individual write protection of the 64 kByte sectors. It is only evaluated if write protection bit FSV1[17] is set.

Explanation of the bits in the Flash Security Vector FSV2[31:0]

FSV1 bit	Sector	Enable Write Protection	Disable Write Protection	Comment
FSV2[0]	-	set to '0'	set to '1'	not available
FSV2[1]	-	set to '0'	set to '1'	not available
FSV2[2]	-	set to '0'	set to '1'	not available
FSV2[3]	-	set to '0'	set to '1'	not available
FSV2[4]	SA12	set to '0'	set to '1'	
FSV2[5]	SA13	set to '0'	set to '1'	
FSV2[6]	SA14	set to '0'	set to '1'	
FSV2[7]	SA15	set to '0'	set to '1'	
FSV2[8]	SA16	set to '0'	set to '1'	
FSV2[9]	SA17	set to '0'	set to '1'	
FSV2[10]	SA18	set to '0'	set to '1'	
FSV2[11]	SA19	set to '0'	set to '1'	
FSV2[12-32]	-	-	-	not available

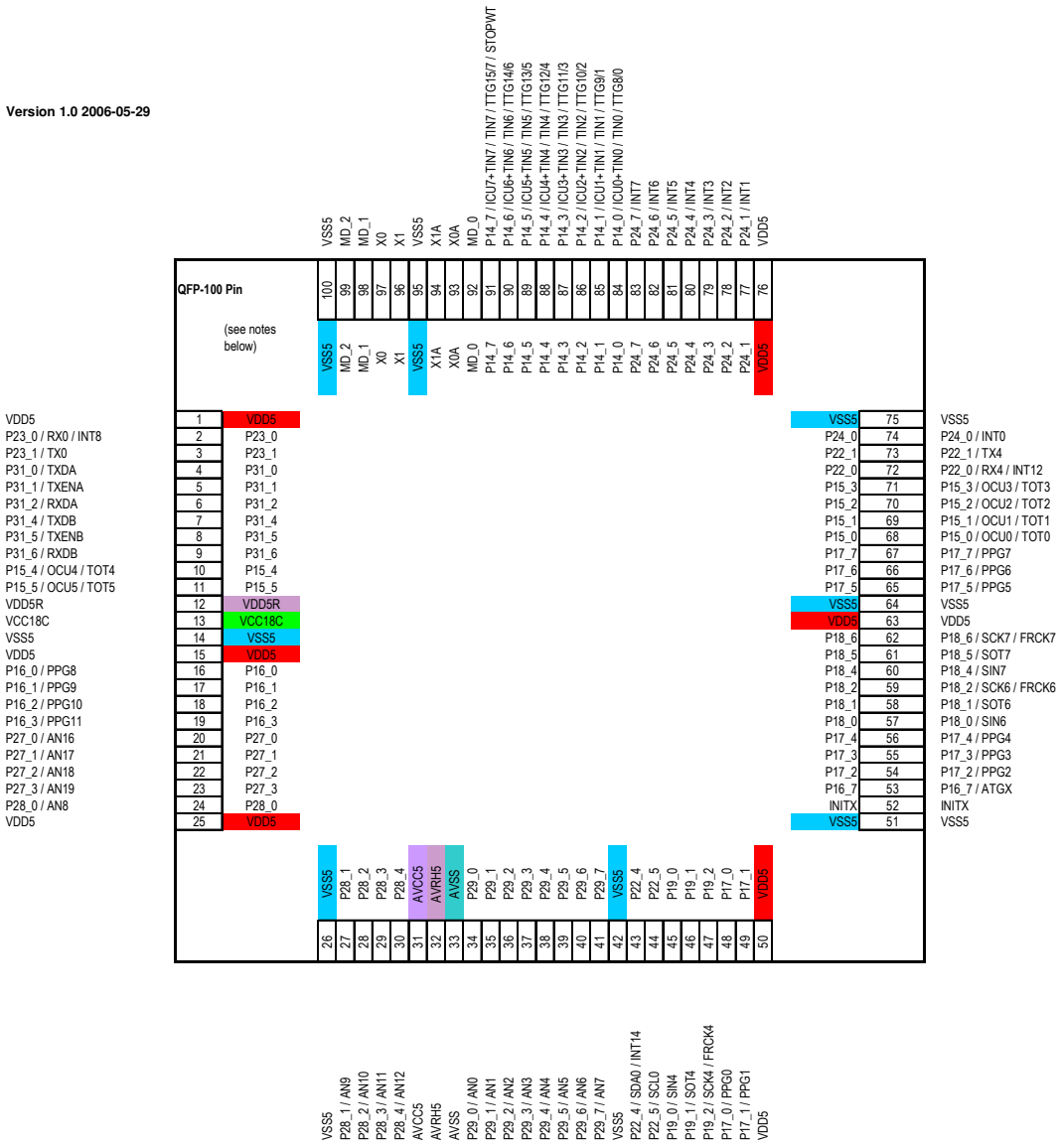
2.4.4.4 Register description for Flash Security

For a description of Flash Security registers please refer to Hardware Manual chapter 55.

3 Package and Pin Assignment

3.1 Package

A LQFP-100 package will be used for MB91F465XA. The package code is FPT-100P-M20 (100-pin plastic QFP, lead pitch: 0.50 mm, size 14 mm x 14 mm, Theta-ja = 30 K/W)



3.2 I/O Pins and Their Functions

Pin Number	Name	PFR=1	EPFR=1	Special	Pad Type	Description
96	X1	-	-	-	TO00_1	4 MHz Quartz Oscillator
97	X0	-	-	-	TO00_0	
94	X1A	-	-	-	TO01_1	32 kHz Quartz Oscillator
93	X0A	-	-	-	TO01_0	
91	P14_7	ICU7+TIN7	TIN7	TTG15/7 / STOPWT	TP04_0	General Purpose Port 14
90	P14_6	ICU6+TIN6	TIN6	TTG14/6	TP04_0	
89	P14_5	ICU5+TIN5	TIN5	TTG13/5	TP04_0	ICU: Input Capture Unit event input
88	P14_4	ICU4+TIN4	TIN4	TTG12/4	TP04_0	TIN: Reload Timer external trigger input
87	P14_3	ICU3+TIN3	TIN3	TTG11/3	TP04_0	TTG: PPG external trigger input
86	P14_2	ICU2+TIN2	TIN2	TTG10/2	TP04_0	STOPWT: FlexRay Stopwatch input
85	P14_1	ICU1+TIN1	TIN1	TTG9/1	TP04_0	
84	P14_0	ICU0+TIN0	TIN0	TTG8/0	TP04_0	
11	P15_5	OCU5	TOT5	-	TP04_0	General Purpose Port 15
10	P15_4	OCU4	TOT4	-	TP04_0	
71	P15_3	OCU3	TOT3	-	TP04_0	OCU: Output Compare waveform output
70	P15_2	OCU2	TOT2	-	TP04_0	TOT: Reload Timer output
69	P15_1	OCU1	TOT1	-	TP04_0	
68	P15_0	OCU0	TOT0	-	TP04_0	
53	P16_7	-	ATGX	-	TP04_0	General Purpose Port 16
19	P16_3	PPG11	-	-	TP04_0	
18	P16_2	PPG10	-	-	TP04_0	ATGX: ADC external trigger
17	P16_1	PPG9	-	-	TP04_0	PPG: PPG waveform output
16	P16_0	PPG8	-	-	TP04_0	
67	P17_7	PPG7	-	-	TP04_0	General Purpose Port 17
66	P17_6	PPG6	-	-	TP04_0	
65	P17_5	PPG5/ MONCLK	-	-	TP04_0	PPG: PPG waveform output
56	P17_4	PPG4	-	-	TP04_0	MONCLK: Clock Monitor output
55	P17_3	PPG3	-	-	TP04_0	
54	P17_2	PPG2	-	-	TP04_0	
49	P17_1	PPG1	-	-	TP04_0	
48	P17_0	PPG0	-	-	TP04_0	
62	P18_6	SCK7	FRCK7	-	TP04_0	General Purpose Port 18
61	P18_5	SOT7	-	-	TP04_0	FRCK: Free Run Timer external clock input
60	P18_4	SIN7	-	-	TP04_0	SOT: LIN-USART data output
59	P18_2	SCK6	FRCK6	-	TP04_0	SCK: LIN-USART serial clock in/out
58	P18_1	SOT6	-	-	TP04_0	SIN: LIN-USART data input
57	P18_0	SIN6	-	-	TP04_0	
47	P19_2	SCK4	FRCK4	-	TP04_0	General Purpose Port 19

46	P19_1	SOT4	-	-	TP04_0	SCK, SOT, SIN, FRCK see above (port 18)
45	P19_0	SIN4	-	-	TP04_0	
44	P22_5	SCL0	-	-	TP02_0	General Purpose Port 22
43	P22_4	SDA0	-	INT14	TP02_0	SCL, SDA: I2C Clock/Data in/out (open drain)
73	P22_1	TX4	-	-	TP04_0	TX, RX: CAN Transmit / Receive out/in
72	P22_0	RX4	-	INT12	TP04_0	INT: External Interrupt (I2C/CAN-Wakeup)
3	P23_1	TX0	-	-	TP04_0	General Purpose Port 23
2	P23_0	RX0	-	INT8	TP04_0	TX, RX, INT see above (port 22)
83	P24_7	INT7	-	--	TP04_0	General Purpose Port 24
82	P24_6	INT6	-	--	TP04_0	
81	P24_5	INT5	-	--	TP04_0	INT: External Interrupt input
80	P24_4	INT4	-	--	TP04_0	
79	P24_3	INT3	-	-	TP04_0	
78	P24_2	INT2	-	-	TP04_0	
77	P24_1	INT1	-	-	TP04_0	
74	P24_0	INT0	-	-	TP04_0	
23	P27_3	-	AN19	-	TP05_0	General Purpose Port 27
22	P27_2	-	AN18	-	TP05_0	
21	P27_1	-	AN17	-	TP05_0	AN: ADC Analog input
20	P27_0	-	AN16	-	TP05_0	
30	P28_4	AN12	-	-	TP03_0	General Purpose Port 28
29	P28_3	AN11	-	-	TP03_0	
28	P28_2	AN10	-	-	TP03_0	AN: ADC Analog input
27	P28_1	AN9	-	-	TP03_0	
24	P28_0	AN8	-	-	TP03_0	
41	P29_7	AN7	-	-	TP03_0	General Purpose Port 29
40	P29_6	AN6	-	-	TP03_0	
39	P29_5	AN5	-	-	TP03_0	AN: ADC Analog input
38	P29_4	AN4	-	-	TP03_0	
37	P29_3	AN3	-	-	TP03_0	
36	P29_2	AN2	-	-	TP03_0	
35	P29_1	AN1	-	-	TP03_0	
34	P29_0	AN0	-	-	TP03_0	
9	P31_6	-	RXDB	-	TP06_0	General Purpose Port 31
8	P31_5	-	TXENB	-	TP06_0	
7	P31_4	-	TXDB	-	TP06_0	RXD: FlexRay Receive inputs
6	P31_2	-	RXDA	-	TP06_0	TXD: FlexRay Transmit outputs
5	P31_1	-	TXENA	-	TP06_0	TXEN: FlexRay Transmit Enable outputs
4	P31_0	-	TXDA	-	TP06_0	
52	INITX	-	-	-	TC02_0	Initialization input (low active)
99	MD_2	-	-	-	TC01_0	
98	MD_1	-	-	-	TC01_0	Device Mode inputs
92	MD_0	-	-	-	TC01_0	

1	VDD5	-	-	-	TS02_0	Power Supply
15	VDD5	-	-	-	TS02_0	
25	VDD5	-	-	-	TS02_0	
50	VDD5	-	-	-	TS02_0	
63	VDD5	-	-	-	TS02_0	
76	VDD5	-	-	-	TS02_0	
14	VSS5	-	-	-	TS00_0	Ground Supply
26	VSS5	-	-	-	TS00_0	
42	VSS5	-	-	-	TS00_0	
51	VSS5	-	-	-	TS00_0	
64	VSS5	-	-	-	TS00_0	
75	VSS5	-	-	-	TS00_0	
95	VSS5	-	-	-	TS00_0	
100	VSS5	-	-	-	TS00_0	
33	AVSS	-	-	-	TA03_0	Analog Ground Supply
32	AVRH5	-	-	-	TA01_0	Analog Reference Supply
31	AVCC5	-	-	-	TA00_0	Analog Power Supply
12	VDD5R	-	-	-	TA00_0	Voltage Regulator Supply
13	VCC18C	-	-	-	TA10_0	Voltage Regulator Capacitance

3.3 I/O Pin Types

Pin Type	Pull Up/ Down	Input Type	STOP control	Output Driver	Comment
TP02_0	U/D control	CH / A / TTL / CH2	Stop	3 mA	I2C Pin (open drain if PFR=1)
TP03_0	U/D control	CH / A / TTL / CH2	Stop	2/5 mA	General Purpose I/O with 1 analog input line
TP04_0	U/D control	CH / A / TTL / CH2	Stop	2/5 mA	General Purpose I/O
TP05_0	U/D control	CH / A / TTL / CH2	Stop	2/5/30 mA	General Purpose I/O, 30mA SMC, 2 analog lines
TP05_1	U/D control	CH / A / TTL / CH2	Stop	2/5/30 mA	General Purpose I/O, 30mA SMC, 1 analog line
TC01_0	-	CH2	no	-	Mode Pin
TC02_0	Up	CH2	no	-	INITX
TC10_0	-	-	no	5 mA	Output port 5mA for MONCLK

Notes:

- The pull-up / pull-down resistors are typical 50 kOhm. The controlled pull-up/down's can be enabled by register setting.
- Input Types: CH CMOS Schmitt trigger
  CH2 CMOS Schmitt trigger
  A CMOS Automotive Schmitt trigger
  TTL TTL
  (for input high/low voltages, please see section Operating Conditions)
- Stop control: Switch to HIZ in STOP mode by register setting, and disable input lines in STOP if the port is not configured to be external interrupt input.
- Default output driver strength is 3 mA (I^2C pins) and 5 mA (all other pins).

4 Recommended Settings

4.1 PLL and Clockgear settings

Recommended PLL divider and clockgear settings

PLL Input (CK) [MHz]	Frequency Parameter		Clockgear Parameter		PLL Output (X) [MHz]	Core base Clock [MHz]
	DIVM	DIVN	DIVG	MULG		
4	2	25	16	24	200	100
4	2	24	16	24	192	96
4	2	23	16	24	184	92
4	2	22	16	24	176	88
4	2	21	16	20	168	84
4	2	20	16	20	160	80
4	2	19	16	20	152	76
4	2	18	16	20	144	72
4	2	17	16	16	136	68
4	2	16	16	16	128	64
4	2	15	16	16	120	60
4	2	14	16	16	112	56
4	2	13	16	12	104	52
4	2	12	16	12	96	48
4	2	11	16	12	88	44

4	4	10	16	24	160	40
4	4	9	16	24	144	36
4	4	8	16	24	128	32
4	4	7	16	24	112	28
4	6	6	16	24	144	24
4	8	5	16	28	160	20
4	10	4	16	32	160	16
4	12	3	16	32	144	12

4.2 Flash interface settings

Please refer to section 2.4.2.2 ‘Flash access timing settings in CPU mode’ for the recommended Flash interface settings.

4.3 Clock Modulator settings

The following table shows all possible settings for the Clock Modulator in a base clock frequency range from 32MHz up to 88MHz. The PLL and clockgear settings (see section 4.1) should be set according to base clock frequency in the table below.

Clock Modulator settings and frequency range

Modulation Degree	Random No	CMPR	Baseclk	Fmin	Fmax
(k)	(N)	[hex]	[MHz]	[MHz]	[MHz]
1	3	026F	88	79.5	98.5
1	3	026F	84	76.1	93.8
1	3	026F	80	72.6	89.1
1	5	02AE	80	68.7	95.8
2	3	046E	80	68.7	95.8
1	3	026F	76	69.1	84.5
1	5	02AE	76	65.3	90.8
1	7	02ED	76	62	98.1
2	3	046E	76	65.3	90.8
3	3	066D	76	62	98.1
1	3	026F	72	65.5	79.9
1	5	02AE	72	62	85.8
1	7	02ED	72	58.8	92.7
2	3	046E	72	62	85.8
3	3	066D	72	58.8	92.7
1	3	026F	68	62	75.3
1	5	02AE	68	58.7	80.9
1	7	02ED	68	55.7	87.3
1	9	032C	68	53	95
2	3	046E	68	58.7	80.9
2	5	04AC	68	53	95
3	3	066D	68	55.7	87.3
4	3	086C	68	53	95
1	3	026F	64	58.5	70.7
1	5	02AE	64	55.3	75.9
1	7	02ED	64	52.5	82
1	9	032C	64	49.9	89.1
1	11	036B	64	47.6	97.6
2	3	046E	64	55.3	75.9
2	5	04AC	64	49.9	89.1
3	3	066D	64	52.5	82
4	3	086C	64	49.9	89.1
5	3	0A6B	64	47.6	97.6
1	3	026F	60	54.9	66.1
1	5	02AE	60	51.9	71
1	7	02ED	60	49.3	76.7

Modulation Degree	Random No	CMPR	Baseclk	Fmin	Fmax
(k)	(N)	[hex]	[MHz]	[MHz]	[MHz]
1	9	032C	60	46.9	83.3
1	11	036B	60	44.7	91.3
2	3	046E	60	51.9	71
2	5	04AC	60	46.9	83.3
3	3	066D	60	49.3	76.7
4	3	086C	60	46.9	83.3
5	3	0A6B	60	44.7	91.3
1	3	026F	56	51.4	61.6
1	5	02AE	56	48.6	66.1
1	7	02ED	56	46.1	71.4
1	9	032C	56	43.8	77.6
1	11	036B	56	41.8	84.9
1	13	03AA	56	39.9	93.8
2	3	046E	56	48.6	66.1
2	5	04AC	56	43.8	77.6
2	7	04EA	56	39.9	93.8
3	3	066D	56	46.1	71.4
3	5	06AA	56	39.9	93.8
4	3	086C	56	43.8	77.6
5	3	0A6B	56	41.8	84.9
6	3	0C6A	56	39.9	93.8
1	3	026F	52	47.8	57
1	5	02AE	52	45.2	61.2
1	7	02ED	52	42.9	66.1
1	9	032C	52	40.8	71.8
1	11	036B	52	38.8	78.6
1	13	03AA	52	37.1	86.8
1	15	03E9	52	35.5	96.9
2	3	046E	52	45.2	61.2
2	5	04AC	52	40.8	71.8
2	7	04EA	52	37.1	86.8
3	3	066D	52	42.9	66.1
3	5	06AA	52	37.1	86.8
4	3	086C	52	40.8	71.8
5	3	0A6B	52	38.8	78.6
6	3	0C6A	52	37.1	86.8
7	3	0E69	52	35.5	96.9
1	3	026F	48	44.2	52.5
1	5	02AE	48	41.8	56.4
1	7	02ED	48	39.6	60.9
1	9	032C	48	37.7	66.1
1	11	036B	48	35.9	72.3
1	13	03AA	48	34.3	79.9
1	15	03E9	48	32.8	89.1
2	3	046E	48	41.8	56.4
2	5	04AC	48	37.7	66.1

Modulation Degree	Random No	CMPR	Baseclk	Fmin	Fmax
(k)	(N)	[hex]	[MHz]	[MHz]	[MHz]
2	7	04EA	48	34.3	79.9
3	3	066D	48	39.6	60.9
3	5	06AA	48	34.3	79.9
4	3	086C	48	37.7	66.1
5	3	0A6B	48	35.9	72.3
6	3	0C6A	48	34.3	79.9
7	3	0E69	48	32.8	89.1
1	3	026F	44	40.6	48.1
1	5	02AE	44	38.4	51.6
1	7	02ED	44	36.4	55.7
1	9	032C	44	34.6	60.4
1	11	036B	44	33	66.1
1	13	03AA	44	31.5	73
1	15	03E9	44	30.1	81.4
2	3	046E	44	38.4	51.6
2	5	04AC	44	34.6	60.4
2	7	04EA	44	31.5	73
2	9	0528	44	28.9	92.1
3	3	066D	44	36.4	55.7
3	5	06AA	44	31.5	73
4	3	086C	44	34.6	60.4
4	5	08A8	44	28.9	92.1
5	3	0A6B	44	33	66.1
6	3	0C6A	44	31.5	73
7	3	0E69	44	30.1	81.4
8	3	1068	44	28.9	92.1
1	3	026F	40	37	43.6
1	5	02AE	40	34.9	46.8
1	7	02ED	40	33.1	50.5
1	9	032C	40	31.5	54.8
1	11	036B	40	30	59.9
1	13	03AA	40	28.7	66.1
1	15	03E9	40	27.4	73.7
2	3	046E	40	34.9	46.8
2	5	04AC	40	31.5	54.8
2	7	04EA	40	28.7	66.1
2	9	0528	40	26.3	83.3
3	3	066D	40	33.1	50.5
3	5	06AA	40	28.7	66.1
3	7	06E7	40	25.3	95.8
4	3	086C	40	31.5	54.8
4	5	08A8	40	26.3	83.3
5	3	0A6B	40	30	59.9
6	3	0C6A	40	28.7	66.1
7	3	0E69	40	27.4	73.7
8	3	1068	40	26.3	83.3

Modulation Degree	Random No	CMPR	Baseclk	Fmin	Fmax
(k)	(N)	[hex]	[MHz]	[MHz]	[MHz]
9	3	1267	40	25.3	95.8
1	3	026F	36	33.3	39.2
1	5	02AE	36	31.5	42
1	7	02ED	36	29.9	45.3
1	9	032C	36	28.4	49.2
1	11	036B	36	27.1	53.8
1	13	03AA	36	25.8	59.3
1	15	03E9	36	24.7	66.1
2	3	046E	36	31.5	42
2	5	04AC	36	28.4	49.2
2	7	04EA	36	25.8	59.3
2	9	0528	36	23.7	74.7
3	3	066D	36	29.9	45.3
3	5	06AA	36	25.8	59.3
3	7	06E7	36	22.8	85.8
4	3	086C	36	28.4	49.2
4	5	08A8	36	23.7	74.7
5	3	0A6B	36	27.1	53.8
6	3	0C6A	36	25.8	59.3
7	3	0E69	36	24.7	66.1
8	3	1068	36	23.7	74.7
9	3	1267	36	22.8	85.8
1	3	026F	32	29.7	34.7
1	5	02AE	32	28	37.3
1	7	02ED	32	26.6	40.2
1	9	032C	32	25.3	43.6
1	11	036B	32	24.1	47.7
1	13	03AA	32	23	52.5
1	15	03E9	32	22	58.6
2	3	046E	32	28	37.3
2	5	04AC	32	25.3	43.6
2	7	04EA	32	23	52.5
2	9	0528	32	21.1	66.1
2	11	0566	32	19.5	89.1
3	3	066D	32	26.6	40.2
3	5	06AA	32	23	52.5
3	7	06E7	32	20.3	75.9
4	3	086C	32	25.3	43.6
4	5	08A8	32	21.1	66.1
5	3	0A6B	32	24.1	47.7
5	5	0AA6	32	19.5	89.1
6	3	0C6A	32	23	52.5
7	3	0E69	32	22	58.6
8	3	1068	32	21.1	66.1
9	3	1267	32	20.3	75.9
10	3	1466	32	19.5	89.1

4.4 FlexRay PLL and Clock settings

$0004DC_H$	PLL2DIVM [R/W] - - - - 0000	PLL2DIVN [R/W] - - 000000	PLL2DIVG [R/W] - - - - 0000	PLL2MULG [R/W] 00000000	**PLL2 Clock Control (FlexRay)**
$0004E0_H$	PLL2CTRL [R/W] - - - - 0000	res.	CLKR2 [R/W] - - - 00000	res.	

Recommended FlexRay PLL divider and clockgear settings

PLL Input (CK) [MHz]	Frequency Parameter		Clockgear Parameter		PLL2 Output (X) [MHz]	FlexRay SCLK Clock [MHz]
	DIVM2	DIVN2	DIVG2	MULG2		
4	2	20	0	0	160	80

Recommended FlexRay clock settings

```
- Specification of the CLKR2 registers is as follows:

// ============================================================================
//  REGISTER CLKR2 addr 0x04E2
// ============================================================================
//       15     14     13     12     11      10        9       8
//        7      6      5      4      3       2        1       0
//       +------+------+------+------+-------+--------+-------+-------+
// CLKR2 |      |      |      |      | CKDBL | PLL2EN | CLKS1 | CLKS0 | initial
//       +------+------+------+------+-------+--------+-------+-------+ 00000000
//                                    R/W     R/W      R/W     R/W
```

CLKS1, CLKS0: FlexRay SCLK Source Selection

CLKS[1:0] = 00 : SCLK is operated with CLKB (coreclock)
CLKS[1:0] = 01 : SCLK is operated with Main PLL (baseclock)
CLKS[1:0] = 10 : SCLK is operated with FlexRay PLL **<- recommended setting**
CLKS[1:0] = 11 : Testmode only, do not set !

PLL2EN: FlexRay PLL enable

PLL2EN = 0 : FlexRay PLL is disabled
PLL2EN = 1 : FlexRay PLL is enabled

CKDBL: FlexRay Clock disable (BCLK and SCLK)

CKDBL = 0 : FlexRay clocks are enabled
CKDBL = 1 : FlexRay clocks are disabled

5 Interrupt Vector Table

This section shows the allocation of interrupt and interrupt vector/interrupt register.

Interrupt	Interrupt number		Interrupt level[*1]		Interrupt vector[*2]		
	Decimal	Hexa-decimal	Setting Register	Register address	Offset	Default Vector address	RN
Reset	0	00	-	-	0x3FC	0x000FFFFC	
Mode vector	1	01	-	-	0x3F8	0x000FFFF8	
System reserved	2	02	-	-	0x3F4	0x000FFFF4	
System reserved	3	03	-	-	0x3F0	0x000FFFF0	
System reserved	4	04	-	-	0x3EC	0x000FFFEC	
CPU supervisor mode (INT #5 instruction) [*6]	5	05	-	-	0x3E8	0x000FFFE8	
Memory Protection exception [*6]	6	06	-	-	0x3E4	0x000FFFE4	
Co-processor fault trap [*5]	7	07	-	-	0x3E0	0x000FFFE0	
Co-processor error trap [*5]	8	08	-	-	0x3DC	0x000FFFDC	
INTE instruction [*5]	9	09	-	-	0x3D8	0x000FFFD8	
Instruction break exception [*5]	10	0A	-	-	0x3D4	0x000FFFD4	

Interrupt	Interrupt number		Interrupt level[*1]		Interrupt vector[*2]		
	Decimal	Hexa-decimal	Setting Register	Register address	Offset	Default Vector address	RN
Operand break trap [*5]	11	0B	-	-	0x3D0	0x000FFFD0	
Step trace trap [*5]	12	0C	-	-	0x3CC	0x000FFFCC	
NMI interrupt (tool) [*5]	13	0D	-	-	0x3C8	0x000FFFC8	
Undefined instruction exception	14	0E	-	-	0x3C4	0x000FFFC4	
NMI request	15	0F	F_H fixed		0x3C0	0x000FFFC0	
External Interrupt 0	16	10	ICR00	0x440	0x3BC	0x000FFFBC	0, 16
External Interrupt 1	17	11			0x3B8	0x000FFFB8	1, 17
External Interrupt 2	18	12	ICR01	0x441	0x3B4	0x000FFFB4	2, 18
External Interrupt 3	19	13			0x3B0	0x000FFFB0	3, 19
External Interrupt 4	20	14	ICR02	0x442	0x3AC	0x000FFFAC	20
External Interrupt 5	21	15			0x3A8	0x000FFFA8	21
External Interrupt 6	22	16	ICR03	0x443	0x3A4	0x000FFFA4	22
External Interrupt 7	23	17			0x3A0	0x000FFFA0	23
External Interrupt 8	24	18	ICR04	0x444	0x39C	0x000FFF9C	
reserved	25	19			0x398	0x000FFF98	
reserved	26	1A	ICR05	0x445	0x394	0x000FFF94	
reserved	27	1B			0x390	0x000FFF90	

Interrupt	Interrupt number		Interrupt level[*1]		Interrupt vector[*2]		
	Decimal	Hexa-decimal	Setting Register	Register address	Offset	Default Vector address	RN
External Interrupt 12	28	1C	ICR06	0x446	0x38C	0x000FFF8C	
reserved	29	1D			0x388	0x000FFF88	
External Interrupt 14	30	1E	ICR07	0x447	0x384	0x000FFF84	
reserved	31	1F			0x380	0x000FFF80	
Reload Timer 0	32	20	ICR08	0x448	0x37C	0x000FFF7C	4, 32
Reload Timer 1	33	21			0x378	0x000FFF78	5, 33
Reload Timer2	34	22	ICR09	0x449	0x374	0x000FFF74	34
Reload Timer 3	35	23			0x370	0x000FFF70	35
Reload Timer 4	36	24	ICR10	0x44A	0x36C	0x000FFF6C	36
Reload Timer 5	37	25			0x368	0x000FFF68	37
Reload Timer 6	38	26	ICR11	0x44B	0x364	0x000FFF64	38
Reload Timer 7	39	27			0x360	0x000FFF60	39
Free Run Timer 0	40	28	ICR12	0x44C	0x35C	0x000FFF5C	40
Free Run Timer 1	41	29			0x358	0x000FFF58	41
Free Run Timer 2	42	2A	ICR13	0x44D	0x354	0x000FFF54	42
Free Run Timer 3	43	2B			0x350	0x000FFF50	43
Free Run Timer 4	44	2C	ICR14	0x44E	0x34C	0x000FFF4C	44
Free Run Timer 5	45	2D			0x348	0x000FFF48	45

Interrupt	Interrupt number		Interrupt level[*1]		Interrupt vector[*2]		
	Decimal	Hexa-decimal	Setting Register	Register address	Offset	Default Vector address	RN
Free Run Timer 6	46	2E	ICR15	0x44F	0x344	0x000FFF44	46
Free Run Timer 7	47	2F			0x340	0x000FFF40	47
CAN 0	48	30	ICR16	0x450	0x33C	0x000FFF3C	
reserved	49	31			0x338	0x000FFF38	
reserved	50	32	ICR17	0x451	0x334	0x000FFF34	
reserved	51	33			0x330	0x000FFF30	
CAN 4	52	34	ICR18	0x452	0x32C	0x000FFF2C	
reserved	53	35			0x328	0x000FFF28	
reserved	54	36	ICR19	0x453	0x324	0x000FFF24	6, 48
reserved	55	37			0x320	0x000FFF20	7, 49
reserved	56	38	ICR20	0x454	0x31C	0x000FFF1C	8, 50
reserved	57	39			0x318	0x000FFF18	9, 51
reserved	58	3A	ICR21	0x455	0x314	0x000FFF14	52
reserved	59	3B			0x310	0x000FFF10	53
reserved	60	3C	ICR22	0x456	0x30C	0x000FFF0C	54
reserved	61	3D			0x308	0x000FFF08	55
System reserved	62	3E	ICR23 [*4]	0x457	0x304	0x000FFF04	
Delayed Interrupt	63	3F			0x300	0x000FFF00	

Interrupt	Interrupt number		Interrupt level[*1]		Interrupt vector[*2]		
	Decimal	Hexa-decimal	Setting Register	Register address	Offset	Default Vector address	RN
System reserved [*3]	64	40	(ICR24)	(0x458)	0x2FC	0x000FFEFC	
System reserved [*3]	65	41			0x2F8	0x000FFEF8	
USART (LIN FIFO) 4 RX	66	42	ICR25	0x459	0x2F4	0x000FFEF4	10, 56
USART (LIN FIFO) 4 TX	67	43			0x2F0	0x000FFEF0	11, 57
reserved	68	44	ICR26	0x45A	0x2EC	0x000FFEEC	12, 58
reserved	69	45			0x2E8	0x000FFEE8	13, 59
USART (LIN FIFO) 6 RX	70	46	ICR27	0x45B	0x2E4	0x000FFEE4	60
USART (LIN FIFO) 6 TX	71	47			0x2E0	0x000FFEE0	61
USART (LIN FIFO) 7 RX	72	48	ICR28	0x45C	0x2DC	0x000FFEDC	62
USART (LIN FIFO) 7 TX	73	49			0x2D8	0x000FFED8	63
I2C 0	74	4A	ICR29	0x45D	0x2D4	0x000FFED4	
reserved	75	4B			0x2D0	0x000FFED0	
reserved	76	4C	ICR30	0x45E	0x2CC	0x000FFECC	64
reserved	77	4D			0x2C8	0x000FFEC8	65
reserved	78	4E	ICR31	0x45F	0x2C4	0x000FFEC4	66
reserved	79	4F			0x2C0	0x000FFEC0	67
reserved	80	50	ICR32	0x460	0x2BC	0x000FFEBC	68

Interrupt	Interrupt number		Interrupt level[*1]		Interrupt vector[*2]		
	Decimal	Hexa-decimal	Setting Register	Register address	Offset	Default Vector address	RN
reserved	81	51			0x2B8	0x000FFEB8	69
reserved	82	52	ICR33	0x461	0x2B4	0x000FFEB4	70
reserved	83	53			0x2B0	0x000FFEB0	71
FlexRay 0	84	54	ICR34	0x462	0x2AC	0x000FFEAC	72 116 (IBF) 117 (OBF)
FlexRay Timer 0	85	55			0x2A8	0x000FFEA8	73
FlexRay 1	86	56	ICR35	0x463	0x2A4	0x000FFEA4	74 116 (IBF) 117 (OBF)
FlexRay Timer 1	87	57			0x2A0	0x000FFEA0	75
reserved	88	58	ICR36	0x464	0x29C	0x000FFE9C	76
reserved	89	59			0x298	0x000FFE98	77
reserved	90	5A	ICR37	0x465	0x294	0x000FFE94	78
reserved	91	5B			0x290	0x000FFE90	79
Input Capture 0	92	5C	ICR38	0x466	0x28C	0x000FFE8C	80
Input Capture 1	93	5D			0x288	0x000FFE88	81
Input Capture 2	94	5E	ICR39	0x467	0x284	0x000FFE84	82
Input Capture 3	95	5F			0x280	0x000FFE80	83
Input Capture 4	96	60	ICR40	0x468	0x27C	0x000FFE7C	84
Input Capture 5	97	61			0x278	0x000FFE78	85

Interrupt	Interrupt number		Interrupt level[*1]		Interrupt vector[*2]		
	Decimal	Hexa-decimal	Setting Register	Register address	Offset	Default Vector address	RN
Input Capture 6	98	62	ICR41	0x469	0x274	0x000FFE74	86
Input Capture 7	99	63			0x270	0x000FFE70	87
Output Compare 0	100	64	ICR42	0x46A	0x26C	0x000FFE6C	88
Output Compare 1	101	65			0x268	0x000FFE68	89
Output Compare 2	102	66	ICR43	0x46B	0x264	0x000FFE64	90
Output Compare 3	103	67			0x260	0x000FFE60	91
Output Compare 4	104	68	ICR44	0x46C	0x25C	0x000FFE5C	92
Output Compare 5	105	69			0x258	0x000FFE58	93
reserved	106	6A	ICR45	0x46D	0x254	0x000FFE54	94
reserved	107	6B			0x250	0x000FFE50	95
reserved	108	6C	ICR46	0x46E	0x24C	0x000FFE4C	
reserved	109	6D			0x248	0x000FFE48	
System reserved	110	6E	ICR47 [*4]	0x46F	0x244	0x000FFE44	
System reserved	111	6F			0x240	0x000FFE40	
Prog. Pulse Gen. 0	112	70	ICR48	0x470	0x23C	0x000FFE3C	15, 96
Prog. Pulse Gen. 1	113	71			0x238	0x000FFE38	97
Prog. Pulse Gen. 2	114	72	ICR49	0x471	0x234	0x000FFE34	98
Prog. Pulse Gen. 3	115	73			0x230	0x000FFE30	99

Interrupt	Interrupt number		Interrupt level[*1]		Interrupt vector[*2]		
	Decimal	Hexa-decimal	Setting Register	Register address	Offset	Default Vector address	RN
Prog. Pulse Gen. 4	116	74	ICR50	0x472	0x22C	0x000FFE2C	100
Prog. Pulse Gen. 5	117	75			0x228	0x000FFE28	101
Prog. Pulse Gen. 6	118	76	ICR51	0x473	0x224	0x000FFE24	102
Prog. Pulse Gen. 7	119	77			0x220	0x000FFE20	103
Prog. Pulse Gen. 8	120	78	ICR52	0x474	0x21C	0x000FFE1C	104
Prog. Pulse Gen. 9	121	79			0x218	0x000FFE18	105
Prog. Pulse Gen. 10	122	7A	ICR53	0x475	0x214	0x000FFE14	106
Prog. Pulse Gen. 11	123	7B			0x210	0x000FFE10	107
reserved	124	7C	ICR54	0x476	0x20C	0x000FFE0C	108
reserved	125	7D			0x208	0x000FFE08	109
reserved	126	7E	ICR55	0x477	0x204	0x000FFE04	110
reserved	127	7F			0x200	0x000FFE00	111
reserved	128	80	ICR56	0x478	0x1FC	0x000FFDFC	
reserved	129	81			0x1F8	0x000FFDF8	
reserved	130	82	ICR57	0x479	0x1F4	0x000FFDF4	
reserved	131	83			0x1F0	0x000FFDF0	

Interrupt	Interrupt number		Interrupt level[*1]		Interrupt vector[*2]		
	Decimal	Hexa-decimal	Setting Register	Register address	Offset	Default Vector address	RN
Real Time Clock	132	84	ICR58	0x47A	0x1EC	0x000FFDEC	
Calibration Unit	133	85			0x1E8	0x000FFDE8	
A/D Converter 0	134	86	ICR59	0x47B	0x1E4	0x000FFDE4	14, 112
reserved	135	87			0x1E0	0x000FFDE0	
reserved	136	88	ICR60	0x47C	0x1DC	0x000FFDDC	
reserved	137	89			0x1D8	0x000FFDD8	
Low Voltage Detection	138	8A	ICR61	0x47D	0x1D4	0x000FFDD4	
reserved	139	8B			0x1D0	0x000FFDD0	
Timebase Overflow	140	8C	ICR62	0x47E	0x1CC	0x000FFDCC	
PLL Clock Gear / PLL2 Clock Gear (FlexRay)	141	8D			0x1C8	0x000FFDC8	
DMA Controller	142	8E	ICR63	0x47F	0x1C4	0x000FFDC4	
Main/Sub OSC stability wait	143	8F			0x1C0	0x000FFDC0	
Security vector	144	90	-	-	0x1BC	0x000FFDBC	
Used by the INT instruction.	145 to 255	91 to FF	-	-	0x1B8 to 0x000	0x000FFDB8 to 0x000FFC00	

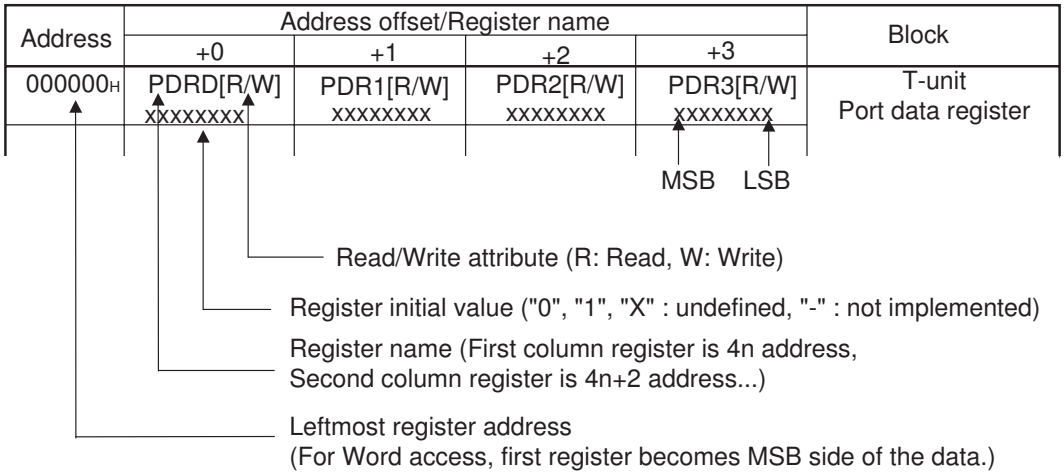
Notes:

[*1] The ICRs are located in the interrupt controller and set the interrupt level for each interrupt request. An ICR is provided for each interrupt request.
[*2] The vector address for each EIT (exception, interrupt or trap) is calculated by adding the listed offset to the table base register value (TBR). The TBR specifies the top of the EIT vector table. The addresses listed in the table are for the default TBR value (0x000FFC00). The TBR is initialized to this value by a reset. After execution of the internal boot ROM TBR is set to 0x000FFC00.
[*3] Used by REALOS
[*4] ICR23 and ICR47 can be exchanged by setting the REALOS compatibility bit (addr 0x0C03 : IOS[0])
[*5] System reserved
[*6] Memory Protection Unit (MPU) support

6 I/O Map

This section shows the association between memory space and each register of peripheral resources.

• Table convention:



Note: Bit value of register shows initial values as follows.

- "1": Initial value is "1".
- "0": Initial value is "0".
- "X": Initial value is indeterminate.
- "-": No physical register exists in the position.

Do not use other data access attributes to access data.

Table 6-1 I/O Map

Address	Register				Block
	+0	+1	+2	+3	
000000_H - 000008H	reserved				
$00000C_H$	res.	res.	PDR14 [R/W] XXXXXXXX	PDR15 [R/W] - - XXXXXX	R-bus Port Data Register
000010_H	PDR16 [R/W] X - - - XXXX	PDR17 [R/W] XXXXXXXX	PDR18 [R/W] - XXX - XXX	PDR19 [R/W] - - - - - XXX	
000014_H	res.	res.	PDR22 [R/W] - - XX - - XX	PDR23 [R/W] - - - - - - XX	
000018_H	PDR24 [R/W] XXXXXXXX	res.	res.	PDR27 [R/W] - - - - XXXX	
$00001C_H$	PDR28 [R/W] - - - XXXXX	PDR29 [R/W] XXXXXXXX	res.	PDR31 [R/W] - XXX - XXX	
000020_H - $00002C_H$	reserved				
000030_H	EIRR0 [R/W] XXXXXXXX	ENIR0 [R/W] 00000000	ELVR0 [R/W] 00000000 00000000		Ext. INT 0-7 NMI
000034_H	EIRR1 [R/W] XXXXXXXX	ENIR1 [R/W] 00000000	ELVR1 [R/W] 00000000 00000000		Ext. INT 8-15
000038_H	DICR [R/W] - - - - - - - 0	HRCL [R/W] 0 - - 11111	RBSYNC *1		DLYI/I-unit
$00003C_H$	reserved				
000040_H - $00005C_H$	reserved				

Address	Register				Block
	+0	+1	+2	+3	
000060_H	SCR04 [R/W,W] 00000000	SMR04 [R/W,W] 00000000	SSR04 [R/W,R] 00001000	RDR04/TDR04 [R/W] 00000000	USART (LIN) 4 with FIFO
000064_H	ESCR04 [R/W] 00000X00	ECCR04 [R/W,R,W] -00000XX	FSR04 [R] - - - 00000	FCR04 [R/W] 0001 - 000	
000068_H - $00006C_H$	reserved				
000070_H	SCR06 [R/W,W] 00000000	SMR06 [R/W,W] 00000000	SSR06 [R/W,R] 00001000	RDR06/TDR06 [R/W] 00000000	USART (LIN) 6 with FIFO
000074_H	ESCR06 [R/W] 00000X00	ECCR06 [R/W,R,W] -00000XX	FSR06 [R] - - - 00000	FCR06 [R/W] 0001 - 000	
000078_H	SCR07 [R/W,W] 00000000	SMR07 [R/W,W] 00000000	SSR07 [R/W,R] 00001000	RDR07/TDR07 [R/W] 00000000	USART (LIN) 7 with FIFO
$00007C_H$	ESCR07 [R/W] 00000X00	ECCR07 [R/W,R,W] -00000XX	FSR07 [R] - - - 00000	FCR07 [R/W] 0001 - 000	
000080_H - 000084_H	reserved				
000088_H	BGR104 [R/W] 00000000	BGR004 [R/W] 00000000	res.	res.	Baudrate Generator USART (LIN) 4, 6-7
$00008C_H$	BGR106 [R/W] 00000000	BGR006 [R/W] 00000000	BGR107 [R/W] 00000000	BGR007 [R/W] 00000000	
000090_H - $0000CC_H$	reserved				
$0000D0_H$	IBCR0 [R/W] 00000000	IBSR0 [R] 00000000	ITBAH0 [R/W] - - - - - - 00	ITBAL0 [R/W] 00000000	I2C 0
$0000D4_H$	ITMKH0 [R/W] 00 - - - - 11	ITMKL0 [R/W] 11111111	ISMK0 [R/W] 01111111	ISBA0 [R/W] - 0000000	
$0000D8_H$	res.	IDAR0 [R/W] 00000000	ICCR0 [R/W] - 0011111	res.	

Address	Register				Block
	+0	+1	+2	+3	
$0000DC_H$ - $0000FC_H$	reserved				
000100_H	GCN10 [R/W] 00110010 00010000		res.	GCN20 [R/W] - - - - 0000	PPG Control 0-3
000104_H	GCN11 [R/W] 00110010 00010000		res.	GCN21 [R/W] - - - - 0000	PPG Control 4-7
000108_H	GCN12 [R/W] 00110010 00010000		res.	GCN22 [R/W] - - - - 0000	PPG Control 8-11
000110_H	PTMR00 [R] 11111111 11111111		PCSR00 [W] XXXXXXXX XXXXXXXX		PPG 0
000114_H	PDUT00 [W] XXXXXXXX XXXXXXXX		PCNH00 [R/W] 0000000 -	PCNL00 [R/W] 000000 - 0	
000118_H	PTMR01 [R] 11111111 11111111		PCSR01 [W] XXXXXXXX XXXXXXXX		PPG 1
$00011C_H$	PDUT01 [W] XXXXXXXX XXXXXXXX		PCNH01 [R/W] 0000000 -	PCNL01 [R/W] 000000 - 0	
000120_H	PTMR02 [R] 11111111 11111111		PCSR02 [W] XXXXXXXX XXXXXXXX		PPG 2
000124_H	PDUT02 [W] XXXXXXXX XXXXXXXX		PCNH02 [R/W] 0000000 -	PCNL02 [R/W] 000000 - 0	
000128_H	PTMR03 [R] 11111111 11111111		PCSR03 [W] XXXXXXXX XXXXXXXX		PPG 3
$00012C_H$	PDUT03 [W] XXXXXXXX XXXXXXXX		PCNH03 [R/W] 0000000 -	PCNL03 [R/W] 000000 - 0	
000130_H	PTMR04 [R] 11111111 11111111		PCSR04 [W] XXXXXXXX XXXXXXXX		PPG 4
000134_H	PDUT04 [W] XXXXXXXX XXXXXXXX		PCNH04 [R/W] 0000000 -	PCNL04 [R/W] 000000 - 0	
000138_H	PTMR05 [R] 11111111 11111111		PCSR05 [W] XXXXXXXX XXXXXXXX		PPG 5

Address	Register				Block
	+0	+1	+2	+3	
00013C$_H$	PDUT05 [W] XXXXXXXX XXXXXXXX		PCNH05 [R/W] 0000000 -	PCNL05 [R/W] 000000 - 0	
000140$_H$	PTMR06 [R] 11111111 11111111		PCSR06 [W] XXXXXXXX XXXXXXXX		PPG 6
000144$_H$	PDUT06 [W] XXXXXXXX XXXXXXXX		PCNH06 [R/W] 0000000 -	PCNL06 [R/W] 000000 - 0	
000148$_H$	PTMR07 [R] 11111111 11111111		PCSR07 [W] XXXXXXXX XXXXXXXX		PPG 7
00014C$_H$	PDUT07 [W] XXXXXXXX XXXXXXXX		PCNH07 [R/W] 0000000 -	PCNL07 [R/W] 000000 - 0	
000150$_H$	PTMR08 [R] 11111111 11111111		PCSR08 [W] XXXXXXXX XXXXXXXX		PPG 8
000154$_H$	PDUT08 [W] XXXXXXXX XXXXXXXX		PCNH08 [R/W] 0000000 -	PCNL08 [R/W] 000000 - 0	
000158$_H$	PTMR09 [R] 11111111 11111111		PCSR09 [W] XXXXXXXX XXXXXXXX		PPG 9
00015C$_H$	PDUT09 [W] XXXXXXXX XXXXXXXX		PCNH09 [R/W] 0000000 -	PCNL09 [R/W] 000000 - 0	
000160$_H$	PTMR10 [R] 11111111 11111111		PCSR10 [W] XXXXXXXX XXXXXXXX		PPG 10
000164$_H$	PDUT10 [W] XXXXXXXX XXXXXXXX		PCNH10 [R/W] 0000000 -	PCNL10 [R/W] 000000 - 0	
000168$_H$	PTMR11 [R] 11111111 11111111		PCSR11 [W] XXXXXXXX XXXXXXXX		PPG 11
00016C$_H$	PDUT11 [W] XXXXXXXX XXXXXXXX		PCNH11 [R/W] 0000000 -	PCNL11 [R/W] 000000 - 0	
000170$_H$ - 00017C$_H$	reserved				

Address	Register				Block
	+0	+1	+2	+3	
000180_H	res.	ICS01 [R/W] 00000000	res.	ICS23 [R/W] 00000000	Input Capture 0-3
000184_H	IPCP0 [R] XXXXXXXX XXXXXXXX		IPCP1 [R] XXXXXXXX XXXXXXXX		
000188_H	IPCP2 [R] XXXXXXXX XXXXXXXX		IPCP3 [R] XXXXXXXX XXXXXXXX		
$00018C_H$	OCS01 [R/W] - - - 0 - - 00 0000 - - 00		OCS23 [R/W] - - - 0 - - 00 0000 - - 00		Output Compare 0-3
000190_H	OCCP0 [R/W] XXXXXXXX XXXXXXXX		OCCP1 [R/W] XXXXXXXX XXXXXXXX		
000194_H	OCCP2 [R/W] XXXXXXXX XXXXXXXX		OCCP3 [R/W] XXXXXXXX XXXXXXXX		
000198_H - $00019C_H$	reserved				
$0001A0_H$	ADERH [R/W] 00000000 00000000		ADERL [R/W] 00000000 00000000		A/D Converter
0001A4	ADCS1 [R/W] 00000000	ADCS0 [R/W] 00000000	ADCR1 [R] 000000XX	ADCR0 [R] XXXXXXXX	
$0001A8_H$	ADCT1 [R/W] 00010000	ADCT0 [R/W] 00101100	ADSCH [R/W] - - - 00000	ADECH [R/W] - - - 00000	
$0001AC_H$	reserved				
$0001B0_H$	TMRLR0 [W] XXXXXXXX XXXXXXXX		TMR0 [R] XXXXXXXX XXXXXXXX		Reload Timer 0 (PPG 0-1)
$0001B4_H$	res.		TMCSRH0 [R/W] - - - 00000	TMCSRL0 [R/W] 0 - 000000	
$0001B8_H$	TMRLR1 [W] XXXXXXXX XXXXXXXX		TMR1 [R] XXXXXXXX XXXXXXXX		Reload Timer 1 (PPG 2-3)
$0001BC_H$	res.		TMCSRH1 [R/W] - - - 00000	TMCSRL1 [R/W] 0 - 000000	

Address	Register				Block
	+0	+1	+2	+3	
$0001C0_H$	TMRLR2 [W] XXXXXXXX XXXXXXXX		TMR2 [R] XXXXXXXX XXXXXXXX		Reload Timer 2 (PPG 4-5)
$0001C4_H$	res.		TMCSRH2 [R/W] - - - 00000	TMCSRL2 [R/W] 0 - 000000	
$0001C8_H$	TMRLR3 [W] XXXXXXXX XXXXXXXX		TMR3 [R] XXXXXXXX XXXXXXXX		Reload Timer 3 (PPG 6-7)
$0001CC_H$	res.		TMCSRH3 [R/W] - - - 00000	TMCSRL3 [R/W] 0 - 000000	
$0001D0_H$	TMRLR4 [W] XXXXXXXX XXXXXXXX		TMR4 [R] XXXXXXXX XXXXXXXX		Reload Timer 4 (PPG 8-9)
$0001D4_H$	res.		TMCSRH4 [R/W] - - - 00000	TMCSRL4 [R/W] 0 - 000000	
$0001D8_H$	TMRLR5 [W] XXXXXXXX XXXXXXXX		TMR5 [R] XXXXXXXX XXXXXXXX		Reload Timer 5 (PPG 10-11)
$0001DC_H$	res.		TMCSRH5 [R/W] - - - 00000	TMCSRL5 [R/W] 0 - 000000	
$0001E0_H$	TMRLR6 [W] XXXXXXXX XXXXXXXX		TMR6 [R] XXXXXXXX XXXXXXXX		Reload Timer 6
$0001E4_H$	res.		TMCSRH6 [R/W] - - - 00000	TMCSRL6 [R/W] 0 - 000000	
$0001E8_H$	TMRLR7 [W] XXXXXXXX XXXXXXXX		TMR7 [R] XXXXXXXX XXXXXXXX		Reload Timer 7 (ADC)
$0001EC_H$	res.		TMCSRH7 [R/W] - - - 00000	TMCSRL7 [R/W] 0 - 000000	
$0001F0_H$	TCDT0 [R/W] XXXXXXXX XXXXXXXX		res.	TCCS0 [R/W] 00000000	Free Running Timer 0 (ICU 0-1)
$0001F4_H$	TCDT1 [R/W] XXXXXXXX XXXXXXXX		res.	TCCS1 [R/W] 00000000	Free Running Timer 1 (ICU 2-3)

Address	Register				Block
	+0	+1	+2	+3	
$0001F8_H$	TCDT2 [R/W] XXXXXXXX XXXXXXXX		res.	TCCS2 [R/W] 00000000	Free Running Timer 2 (OCU 0-1)
$0001FC_H$	TCDT3 [R/W] XXXXXXXX XXXXXXXX		res.	TCCS3 [R/W] 00000000	Free Running Timer 3 (OCU 2-3)
000200_H	DMACA0 [R/W] 00000000 0000XXXX XXXXXXXX XXXXXXXX				DMAC
000204_H	DMACB0 [R/W] 00000000 00000000 XXXXXXXX XXXXXXXX				
000208_H	DMACA1 [R/W] 00000000 0000XXXX XXXXXXXX XXXXXXXX				
$00020C_H$	DMACB1 [R/W] 00000000 00000000 XXXXXXXX XXXXXXXX				
000210_H	DMACA2 [R/W] 00000000 0000XXXX XXXXXXXX XXXXXXXX				
000214_H	DMACB2 [R/W] 00000000 00000000 XXXXXXXX XXXXXXXX				
000218_H	DMACA3 [R/W] 00000000 0000XXXX XXXXXXXX XXXXXXXX				
$00021C_H$	DMACB3 [R/W] 00000000 00000000 XXXXXXXX XXXXXXXX				
000220_H	DMACA4 [R/W] 00000000 0000XXXX XXXXXXXX XXXXXXXX				
000224_H	DMACB4 [R/W] 00000000 00000000 XXXXXXXX XXXXXXXX				
000228_H - $00023C_H$	reserved				
000240_H	DMACR [R/W] 00 - - 0000	reserved			

Address	Register				Block
	+0	+1	+2	+3	
000244_H - $0002CC_H$	reserved				
$0002D0_H$	res.	ICS045 [R/W] 00000000	res.	ICS67 [R/W] 00000000	Input Capture 4-7
$0002D4_H$	IPCP4 [R] XXXXXXXX XXXXXXXX		IPCP5 [R] XXXXXXXX XXXXXXXX		
$0002D8_H$	IPCP6 [R] XXXXXXXX XXXXXXXX		IPCP7 [R] XXXXXXXX XXXXXXXX		
$0002DC_H$	OCS45 [R/W] - - - 0 - - 00 0000 - - 00		reserved		Output Compare 4-5
$0002E0_H$	OCCP4 [R/W] XXXXXXXX XXXXXXXX		OCCP5 [R/W] XXXXXXXX XXXXXXXX		
$0002E4_H$ - $0002EC_H$	reserved				
$0002F0_H$	TCDT4 [R/W] XXXXXXXX XXXXXXXX		res.	TCCS4 [R/W] 00000000	Free Running Timer 4 (ICU 4-5)
$0002F4_H$	TCDT5 [R/W] XXXXXXXX XXXXXXXX		res.	TCCS5 [R/W] 00000000	Free Running Timer 5 (ICU 6-7)
$0002F8_H$	TCDT6 [R/W] XXXXXXXX XXXXXXXX		res.	TCCS6 [R/W] 00000000	Free Running Timer 6 (OCU 4-5)
$0002FC_H$	TCDT7 [R/W] XXXXXXXX XXXXXXXX		res.	TCCS7 [R/W] 00000000	Free Running Timer 7
000300_H - $00038C_H$	reserved				
000390_H	ROMS [R] 11111111 01000011		res.		ROM Select Register

Address	Register				Block
	+0	+1	+2	+3	
000394_H - $0003EC_H$	reserved				
$0003F0_H$	BSD0 [W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				Bit Search Module
$0003F4_H$	BSD1 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$0003F8_H$	BSDC [W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$0003FC_H$	BSRR [R] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000400_H - $00043C_H$	reserved				
000440_H	ICR00 [R/W] ---11111	ICR01 [R/W] ---11111	ICR02 [R/W] ---11111	ICR03 [R/W] ---11111	Interrupt Control Unit
000444_H	ICR04 [R/W] ---11111	ICR05 [R/W] ---11111	ICR06 [R/W] ---11111	ICR07 [R/W] ---11111	
000448_H	ICR08 [R/W] ---11111	ICR09 [R/W] ---11111	ICR10 [R/W] ---11111	ICR11 [R/W] ---11111	
$00044C_H$	ICR12 [R/W] ---11111	ICR13 [R/W] ---11111	ICR14 [R/W] ---11111	ICR15 [R/W] ---11111	
000450_H	ICR16 [R/W] ---11111	ICR17 [R/W] ---11111	ICR18 [R/W] ---11111	ICR19 [R/W] ---11111	
000454_H	ICR20 [R/W] ---11111	ICR21 [R/W] ---11111	ICR22 [R/W] ---11111	ICR23 [R/W] ---11111	
000458_H	ICR24 [R/W] ---11111	ICR25 [R/W] ---11111	ICR26 [R/W] ---11111	ICR27 [R/W] ---11111	
$00045C_H$	ICR28 [R/W] ---11111	ICR29 [R/W] ---11111	ICR30 [R/W] ---11111	ICR31 [R/W] ---11111	
000460_H	ICR32 [R/W] ---11111	ICR33 [R/W] ---11111	ICR34 [R/W] ---11111	ICR35 [R/W] ---11111	

Address	Register				Block
	+0	+1	+2	+3	
000464_H	ICR36 [R/W] ---11111	ICR37 [R/W] ---11111	ICR38 [R/W] ---11111	ICR39 [R/W] ---11111	
000468_H	ICR40 [R/W] ---11111	ICR41 [R/W] ---11111	ICR42 [R/W] ---11111	ICR43 [R/W] ---11111	
$00046C_H$	ICR44 [R/W] ---11111	ICR45 [R/W] ---11111	ICR46 [R/W] ---11111	ICR47 [R/W] ---11111	
000470_H	ICR48 [R/W] ---11111	ICR49 [R/W] ---11111	ICR50 [R/W] ---11111	ICR51 [R/W] ---11111	
000474_H	ICR52 [R/W] ---11111	ICR53 [R/W] ---11111	ICR54 [R/W] ---11111	ICR55 [R/W] ---11111	
000478_H	ICR56 [R/W] ---11111	ICR57 [R/W] ---11111	ICR58 [R/W] ---11111	ICR59 [R/W] ---11111	
$00047C_H$	ICR60 [R/W] ---11111	ICR61 [R/W] ---11111	ICR62 [R/W] ---11111	ICR63 [R/W] ---11111	
000480_H	RSRR [R/W] 10000000	STCR [R/W] 00110011	TBCR [R/W] X0000X00	CTBR [W] XXXXXXXX	Clock Control Unit
000484_H	CLKR [R/W] 00000000	WPR [W] XXXXXXXX	DIVR0 [R/W] 00000011	DIVR1 [R/W] 00000000	
000488_H	res.	res.	res.	res.	
$00048C_H$	PLLDIVM [R/W] - - - - 0000	PLLDIVN [R/W] - - 000000	PLLDIVG [R/W] - - - - 0000	PLLMULG [R/W] 00000000	PLL Clock Gear Unit
000490_H	PLLCTRL [R/W] - - - - 0000	res.	res.	res.	
000494_H	OSCC1 [R/W] - - - - - 010	OSCS1 [R/W] 00001111	OSCC2 [R/W] - - - - - 010	OSCS2 [R/W] 00001111	Main/Sub Oscillator Control
000498_H	PORTEN [R/W] - - - - - - 00	res.	res.	res.	Port Input Enable Control

Address	Register				Block
	+0	+1	+2	+3	
$0004A0_H$	res.	WTCER [R/W] - - - - - - 00	WTCR [R/W] 00000000 000 - 00 - 0		Real Time Clock (Watch Timer)
$0004A4_H$	res.	WTBR [R/W] - - - XXXXX XXXXXXXX XXXXXXXX			
$0004A8_H$	WTHR [R/W] - - - 00000	WTMR [R/W] - - 000000	WTSR [R/W] - - 000000	res.	
$0004AC_H$	CSVTR [R/W] - - - 00010	CSVCR [R/W] 00011100	CSCFG [R/W] 0X000000	res.	Clock-Supervisor / Selector
$0004B0_H$	CUCR [R/W] - - - - - - - - - - - - 0 - - 00		CUTD [R/W] 10000000 00000000		Calibration Unit of Sub Oscillation
$0004B4_H$	CUTR1 [R] - - - - - - - - 00000000		CUTR2 [R] 00000000 00000000		
$0004B8_H$	CMPR [R/W] - - 000010 11111101		res.	CMCR [R/W] - 001 - - 00	Clock Modulation
$0004BC_H$	CMT1 [R/W] 00000000 1 - - - 0000		CMT2 [R/W] - - 000000 - - 000000		
$0004C0_H$	CANPRE [R/W] 0 - - - 0000	CANCKD [R/W] - - - 0 - - - 0	res.	res.	CAN Clock Control
$0004C4_H$	LVSEL [R/W] 00000111	LVDET [R/W] 00000-00	HWWDE [R/W] - - - - - - 00	HWWD [R/W,W] 00011000	LV Detection / Hardware-Watchdog
$0004C8_H$	OSCRH [R/W] 000 - - 001	OSCRL [R/W] - - - - - 000	WPCRH [R/W] 00 - - - 000	WPCRL [R/W] - - - - - - 00	Main/Sub-Oscillation Stabilisation Timer
$0004CC_H$	OSCCR [R/W] - - - - - - 00	res.	REGSEL [R/W] - - 000110	REGCTR [R/W] - - - 0 - - 00	Main/Sub-Oscillation Standby Control / Main/Sub Regulator Control

Address	Register				Block
	+0	+1	+2	+3	
$0004D0_H$ - $0004D8_H$	reserved				
$0004DC_H$	PLL2DIVM [R/W] - - - - 0000	PLL2DIVN [R/W] - - 000000	PLL2DIVG [R/W] - - - - 0000	PLL2MULG [R/W] 00000000	**PLL2 Clock Control (FlexRay)**
$0004E0_H$	PLL2CTRL [R/W] - - - - 0000	res.	CLKR2 [R/W] - - - 00000	res.	
$0004E4_H$ - $000BFC_H$	reserved				
$000C00_H$	res.	res.	res.	IOS [R/W] - - - - - - - 0	I-Unit
$000C04_H$ - $000D08_H$	reserved				
$000D0C_H$	res.	res.	PDRD14 [R] XXXXXXXX	PDRD15 [R] - - XXXXXX	R-bus Port Data Direct Read Register
$000D10$	PDRD16 [R] X - - - XXXX	PDRD17 [R] XXXXXXXX	PDRD18 [R] - XXX - XXX	PDRD19 [R] - - - - - XXX	
$000D14_H$	res.	res.	PDRD22 [R] - - XX - - XX	PDRD23 [R] - - - - - - XX	
$000D18_H$	PDRD24 [R] XXXXXXXX	res.	res.	PDRD27 [R] - - - - XXXX	
$000D1C_H$	PDRD28 [R] - - - XXXXX	PDRD29 [R] XXXXXXXX	res.	PDRD31 [R] - XXX - XXX	
$000D20_H$ - $000D48_H$	reserved				
$000D4C_H$	res.	res.	DDR14 [R/W] 00000000	DDR15 [R/W] - - 000000	R-bus Port Direction Register
$000D50_H$	DDR16 [R/W] 0 - - - 0000	DDR17 [R/W] 00000000	DDR18 [R/W] - 000 - 000	DDR19 [R/W] - - - - - 000	

Address	Register				Block
	+0	+1	+2	+3	
$000D54_H$	res.	res.	DDR22 [R/W] - - 00 - - 00	DDR23 [R/W] - - - - - - 00	
$000D58_H$	DDR24 [R/W] 00000000	res.	res.	DDR27 [R/W] - - - - 0000	
$000D5C_H$	DDR28 [R/W] - - - 00000	DDR29 [R/W] 00000000	res.	DDR31 [R/W] - 000 - 000	
$000D60_H$ - $000D88_H$	reserved				
$000D8C_H$	res.	res.	PFR14 [R/W] 00000000	PFR15 [R/W] - - 000000	R-bus Port Function Register
$000D90_H$	PFR16 [R/W] 0 - - - 0000	PFR17 [R/W] 00000000	PFR18 [R/W] - 000 - 000	PFR19 [R/W] - - - - - 000	
$000D94_H$	res.	res.	PFR22 [R/W] - - 00 - - 00	PFR23 [R/W] - - - - - - 00	
$000D98_H$	PFR24 [R/W] 00000000	res.	res.	PFR27 [R/W] - - - - 0000	
$000D9C_H$	PFR28 [R/W] - - - 00000	PFR29 [R/W] 00000000	res.	PFR31 [R/W] - 000 - 000	
$000DA0_H$ - $000DC8_H$	reserved				
$000DCC_H$	res.	res.	EPFR14 [R/W] 00000000	EPFR15 [R/W] - - 000000	R-bus Port Extra Function Register
$000DD0_H$	EPFR16 [R/W] 0 - - - - - - -	EPFR17 [R/W] - - - - - - - -	EPFR18 [R/W] - 0 - - - 0 - -	EPFR19 [R/W] - - - - - 0 - -	
$000DD4_H$	res.	res.	EPFR22 [R/W] - - - - - - - -	EPFR23 [R/W] - - - - - - - -	
$000DD8_H$	EPFR24 [R/W] - - - - - - - -	res.	res.	EPFR27 [R/W] - - - - 0000	
$000DDC_H$	EPFR28 [R/W] - - - - - - - -	EPFR29 [R/W] - - - - - - - -	res.	EPFR31 [R/W] - 000 - 000	

Address	Register				Block
	+0	+1	+2	+3	
$000DE0_H$ - $000E08_H$	reserved				
$000E0C_H$	res.	res.	PODR14 [R/W] 00000000	PODR15 [R/W] - - 000000	R-bus Port Output Drive Select Register
$000E10_H$	PODR16 [R/W] 0 - - - 0000	PODR17 [R/W] 00000000	PODR18 [R/W] - 000 - 000	PODR19 [R/W] - - - - - 000	
$000E14_H$	res.	res.	PODR22 [R/W] - - 00 - - 00	PODR23 [R/W] - - - - - - 00	
$000E18_H$	PODR24 [R/W] 00000000	res.	res.	PODR27 [R/W] - - - - 0000	
$000E1C_H$	PODR28 [R/W] - - - 00000	PODR29 [R/W] 00000000	res.	PODR31 [R/W] - 000 - 000	
$000E20_H$ - $000E48_H$	reserved				
$000E4C_H$	res.	res.	PILR14 [R/W] 00000000	PILR15 [R/W] - - 000000	R-bus Port Input Level Select Register
$000E50_H$	PILR16 [R/W] 0 - - - 0000	PILR17 [R/W] 00000000	PILR18 [R/W] - 000 - 000	PILR19 [R/W] - - - - - 000	
$000E54_H$	res.	res.	PILR22 [R/W] - - 00 - - 00	PILR23 [R/W] - - - - - - 00	
$000E58_H$	PILR24 [R/W] 00000000	res.	res.	PILR27 [R/W] - - - - 0000	
$000E5C_H$	PILR28 [R/W] - - - 00000	PILR29 [R/W] 00000000	res.	PILR31 [R/W] - 000 - 000	
$000E60_H$ - $000E88_H$	reserved				
$000E8C_H$	res.	res.	EPILR14 [R/W] 00000000	EPILR15 [R/W] - - 000000	R-bus Port Extra Input Level Select Register

Address	Register				Block
	+0	+1	+2	+3	
$000E90_H$	EPILR16 [R/W] 0 - - - 0000	EPILR17 [R/W] 00000000	EPILR18 [R/W] - 000 - 000	EPILR19 [R/W] - - - - - 000	
$000E94_H$	res.	res.	EPILR22 [R/W] - - 00 - - 00	EPILR23 [R/W] - - - - - - 00	
$000E98_H$	EPILR24 [R/W] 00000000	res.	res.	EPILR27 [R/W] - - - - 0000	
$000E9C_H$	EPILR28 [R/W] - - - 00000	EPILR29 [R/W] 00000000	res.	EPILR31 [R/W] - 000 - 000	
$000EA0_H$ - $000EC8_H$	reserved				
$000ECC_H$	res.	res.	PPER14 [R/W] 00000000	PPER15 [R/W] - - 000000	R-bus Port Pull-Up/Down Enable Register
$000ED0_H$	PPER16 [R/W] 0 - - - 0000	PPER17 [R/W] 00000000	PPER18 [R/W] - 000 - 000	PPER19 [R/W] - - - - - 000	
$000ED4_H$	res.	res.	PPER22 [R/W] - - 00 - - 00	PPER23 [R/W] - - - - - - 00	
$000ED8_H$	PPER24 [R/W] 00000000	res.	res.	PPER27 [R/W] - - - - 0000	
$000EDC_H$	PPER28 [R/W] - - - 00000	PPER29 [R/W] 00000000	res.	PPER31 [R/W] - 000 - 000	
$000EE0_H$ - $000F08_H$	reserved				
$000F0C_H$	res.	res.	PPCR14 [R/W] 11111111	PPCR15 [R/W] - - 111111	R-bus Port Pull-Up/Down Control Register
$000F10_H$	PPCR16 [R/W] 1 - - - 1111	PPCR17 [R/W] 11111111	PPCR18 [R/W] - 111 - 111	PPCR19 [R/W] - - - - - 111	
$000F14_H$	res.	res.	PPCR22 [R/W] - - 11 - - 11	PPCR23 [R/W] - - - - - - 11	

Address	Register				Block
	+0	+1	+2	+3	
$000F18_H$	PPCR24 [R/W] 11111111	res.	res.	PPCR27 [R/W] - - - - 1111	
$000F1C_H$	PPCR28 [R/W] - - - 11111	PPCR29 [R/W] 11111111	res.	PPCR31 [R/W] - 111 – 111	
$000F20_H$ - $000F3C_H$	reserved				
001000_H	DMASA0 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				DMAC
001004_H	DMADA0 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
001008_H	DMASA1 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00100C_H$	DMADA1 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
001010_H	DMASA2 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
001014_H	DMADA2 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
001018_H	DMASA3 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00101C_H$	DMADA3 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
001020_H	DMASA4 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
001024_H	DMADA4 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
001028_H - $006FFC_H$	reserved				
007000_H	FMCS [R/W] 01101000	FMCR [R] - - - 00000	FCHCR [R/W] - - - - - - 00 10000011		Flash Memory/ Cache Control

Address	Register				Block
	+0	+1	+2	+3	
007004_H	FMWT [R/W] 11111111 11111111		FMWT2 [R] - 001 - - - -	FMPS [R/W] - - - - - 000	Register
007008_H	FMAC [R] 00000000 00000000 00000000 00000000				
$00700C_H$ - $007FFC_H$	reserved				
008000_H - $00BFFC_H$	MB91F465XA Boot-ROM size is 4kB : $00B000_H$ - $00BFFC_H$ (instruction access is 1 waitcycle, data access is 1 waitcycle)				Boot ROM 16 kB
$00C000_H$	CTRLR0 [R/W] 00000000 00000001		STATR0 [R/W] 00000000 00000000		CAN 0 Control Register
$00C004_H$	ERRCNT0 [R] 00000000 00000000		BTR0 [R/W] 00100011 00000001		
$00C008_H$	INTR0 [R] 00000000 00000000		TESTR0 [R/W] 00000000 X0000000		
$00C00C_H$	BRPE0 [R/W] 00000000 00000000		CBSYNC0 *2		
$00C010_H$	IF1CREQ0 [R/W] 00000000 00000001		IF1CMSK0 [R/W] 00000000 00000000		CAN 0 IF 1 Register
$00C014_H$	IF1MSK20 [R/W] 11111111 11111111		IF1MSK10 [R/W] 11111111 11111111		
$00C018_H$	IF1ARB20 [R/W] 00000000 00000000		IF1ARB10 [R/W] 00000000 00000000		
$00C01C_H$	IF1MCTR0 [R/W] 00000000 00000000		res.		
$00C020_H$	IF1DTA10 [R/W] 00000000 00000000		IF1DTA20 [R/W] 00000000 00000000		
$00C024_H$	IF1DTB10 [R/W] 00000000 00000000		IF1DTB20 [R/W] 00000000 00000000		
$00C028_H$ - $00C02C_H$	reserved				

Address	Register				Block
	+0	+1	+2	+3	
$00C030_H$	*IF1DTA20 [R/W]* *00000000 00000000*		*IF1DTA10 [R/W]* *00000000 00000000*		
$00C034_H$	*IF1DTB20 [R/W]* *00000000 00000000*		*IF1DTB10 [R/W]* *00000000 00000000*		
$00C038_H$ - $00C03C_H$	reserved				
$00C040_H$	IF2CREQ0 [R/W] 00000000 00000001		IF2CMSK0 [R/W] 00000000 00000000		CAN 0 IF 2 Register
$00C044_H$	IF2MSK20 [R/W] 11111111 11111111		IF2MSK10 [R/W] 11111111 11111111		
$00C048_H$	IF2ARB20 [R/W] 00000000 00000000		IF2ARB10 [R/W] 00000000 00000000		
$00C04C_H$	IF2MCTR0 [R/W] 00000000 00000000		res.		
$00C050_H$	IF2DTA10 [R/W] 00000000 00000000		IF2DTA20 [R/W] 00000000 00000000		
$00C054_H$	IF2DTB10 [R/W] 00000000 00000000		IF2DTB20 [R/W] 00000000 00000000		
$00C058_H$ - $00C05C_H$	reserved				
$00C060_H$	*IF2DTA20 [R/W]* *00000000 00000000*		*IF2DTA10 [R/W]* *00000000 00000000*		
$00C064_H$	*IF2DTB20 [R/W]* *00000000 00000000*		*IF2DTB10 [R/W]* *00000000 00000000*		
$00C068_H$ - $00C07C_H$	reserved				
$00C080_H$	TREQR20 [R] 00000000 00000000		TREQR10 [R] 00000000 00000000		CAN 0 Status Flags
$00C084_H$ - $00C08C_H$	reserved				

Address	Register				Block
	+0	+1	+2	+3	
$00C090_H$	NEWDT20 [R] 00000000 00000000		NEWDT10 [R] 00000000 00000000		
$00C094_H$ - $00C09C_H$	reserved				
$00C0A0_H$	INTPND20 [R] 00000000 00000000		INTPND10 [R] 00000000 00000000		
$00C0A4_H$ - $00C0AC_H$	reserved				
$00C0B0_H$	MSGVAL20 [R] 00000000 00000000		MSGVAL10 [R] 00000000 00000000		
$00C0B4_H$ - $00C3FC_H$	reserved				
$00C400_H$	CTRLR4 [R/W] 00000000 00000001		STATR4 [R/W] 00000000 00000000		CAN 4 Control Register
$00C404_H$	ERRCNT4 [R] 00000000 00000000		BTR4 [R/W] 00100011 00000001		
$00C408_H$	INTR4 [R] 00000000 00000000		TESTR4 [R/W] 00000000 X0000000		
$00C40C_H$	BRPE4 [R/W] 00000000 00000000		CBSYNC4 *2		
$00C410_H$	IF1CREQ4 [R/W] 00000000 00000001		IF1CMSK4 [R/W] 00000000 00000000		CAN 4 IF 1 Register
$00C414_H$	IF1MSK24 [R/W] 11111111 11111111		IF1MSK14 [R/W] 11111111 11111111		
$00C418_H$	IF1ARB24 [R/W] 00000000 00000000		IF1ARB14 [R/W] 00000000 00000000		
$00C41C_H$	IF1MCTR4 [R/W] 00000000 00000000		res.		
$00C420_H$	IF1DTA14 [R/W] 00000000 00000000		IF1DTA24 [R/W] 00000000 00000000		

Address	Register				Block
	+0	+1	+2	+3	
$00C424_H$	IF1DTB14 [R/W] 00000000 00000000		IF1DTB24 [R/W] 00000000 00000000		
$00C428_H$ - $00C42C_H$	reserved				
$00C430_H$	*IF1DTA24 [R/W]* *00000000 00000000*		*IF1DTA14 [R/W]* *00000000 00000000*		
$00C434_H$	*IF1DTB24 [R/W]* *00000000 00000000*		*IF1DTB14 [R/W]* *00000000 00000000*		
$00C438_H$ - $00C43C_H$	reserved				
$00C440_H$	IF2CREQ4 [R/W] 00000000 00000001		IF2CMSK4 [R/W] 00000000 00000000		CAN 4 IF 2 Register
$00C444_H$	IF2MSK24 [R/W] 11111111 11111111		IF2MSK14 [R/W] 11111111 11111111		
$00C448_H$	IF2ARB24 [R/W] 00000000 00000000		IF2ARB14 [R/W] 00000000 00000000		
$00C44C_H$	IF2MCTR4 [R/W] 00000000 00000000		res.		
$00C450_H$	IF2DTA14 [R/W] 00000000 00000000		IF2DTA24 [R/W] 00000000 00000000		
$00C454_H$	IF2DTB14 [R/W] 00000000 00000000		IF2DTB24 [R/W] 00000000 00000000		
$00C458_H$ - $00C45C_H$	reserved				
$00C460_H$	*IF2DTA24 [R/W]* *00000000 00000000*		*IF2DTA14 [R/W]* *00000000 00000000*		
$00C464_H$	*IF2DTB24 [R/W]* *00000000 00000000*		*IF2DTB14 [R/W]* *00000000 00000000*		

Address	Register				Block
	+0	+1	+2	+3	
$00C468_H$ - $00C47C_H$	reserved				
$00C480_H$	TREQR24 [R] 00000000 00000000		TREQR14 [R] 00000000 00000000		CAN 4 Status Flags
$00C484_H$ - $00C48C_H$	reserved				
$00C490_H$	NEWDT24 [R] 00000000 00000000		NEWDT14 [R] 00000000 00000000		
$00C494_H$ - $00C49C_H$	reserved				
$00C4A0_H$	INTPND24 [R] 00000000 00000000		INTPND14 [R] 00000000 00000000		
$00C4A4_H$ - $00C4AC_H$	reserved				
$00C4B0_H$	MSGVAL24 [R] 00000000 00000000		MSGVAL14 [R] 00000000 00000000		
$00C4B4_H$ - $00CFFC_H$	reserved				
$00D000_H$	CIF0 [R] 00000100 11111111 01011011 11111111				**FlexRay CIF**
$00D004_H$	CIF1 [R/W] 00000000 00000000 00000000 00000000				
$00D008_H$- $00D00C_H$	reserved (2)				
$00D010_H$	TEST1 [R/W] 00000000 00000000 00000011 00000000				**FlexRay GIF**
$00D014_H$	TEST2 [R/W] 00000000 00000000 00000000 00000000				
$00D018_H$	reserved (1)				
$00D01C_H$	LCK [R/W] 00000000 00000000 00000000 00000000				

Address	Register				Block
	+0	+1	+2	+3	
$00D020_H$	EIR [R/W] 00000000 00000000 00000000 00000000				FlexRay INT
$00D024_H$	SIR [R/W] 00000000 00000000 00000000 00000000				
$00D028_H$	EILS [R/W] 00000000 00000000 00000000 00000000				
$00D02C_H$	SILS [R/W] 00000011 00000011 11111111 11111111				
$00D030_H$	EIES [R/W] 00000000 00000000 00000000 00000000				
$00D034_H$	EIER [R/W] 00000000 00000000 00000000 00000000				
$00D038_H$	SIES [R/W] 00000000 00000000 00000000 00000000				
$00D03C_H$	SIER [R/W] 00000000 00000000 00000000 00000000				
$00D040_H$	ILE [R/W] 00000000 00000000 00000000 00000000				
$00D044_H$	T0C [R/W] 00000000 00000000 00000000 00000000				
$00D048_H$	T1C [R/W] 00000000 00000010 00000000 00000000				
$00D04C_H$	STPW1 [R/W] 00000000 00000000 00000000 00000000				
$00D050_H$	STPW2 [R/W] 00000000 00000000 00000000 00000000				
$00D050_H$- $00D07C_H$	reserved (11)				
$00D080_H$	SUCC1 [R/W] 00001100 01000000 00010000 00000000				FlexRay SUC
$00D084_H$	SUCC2 [R/W] 00000001 00000000 00000101 00000100				
$00D088_H$	SUCC3 [R/W] 00000000 00000000 00000000 00010001				

Address	Register				Block
	+0	+1	+2	+3	
$00D08C_H$	NEMC [R/W] 00000000 00000000 00000000 00000000				**FlexRay NEM**
$00D090_H$	PRTC1 [R/W] 00001000 01001100 00000110 00110011				**FlexRay PRT**
$00D094_H$	PRTC2 [R/W] 00001111 00101101 00001010 00001110				
$00D098_H$	MHDC [R/W] 00000000 00000000 00000000 00000000				**FlexRay MHD**
$00D09C_H$	reserved (1)				
$00D0A0_H$	GTUC1 [R/W] 00000000 00000000 00000010 10000000				**FlexRay GTU**
$00D0A4_H$	GTUC2 [R/W] 00000000 00000010 00000000 00001010				
$00D0A8_H$	GTUC3 [R/W] 00000010 00000010 00000000 00000000				
$00D0AC_H$	GTUC4 [R/W] 00000000 00001000 00000000 00000111				
$00D0B0_H$	GTUC5 [R/W] 00001110 00000000 00000000 00000000				
$00D0B4_H$	GTUC6 [R/W] 00000000 00000010 00000000 00000000				
$00D0B8_H$	GTUC7 [R/W] 00000000 00000010 00000000 00000100				
$00D0BC_H$	GTUC8 [R/W] 00000000 00000000 00000000 00000010				
$00D0C0_H$	GTUC9 [R/W] 00000000 00000000 0000001 00000001				
$00D0C4_H$	GTUC10 [R/W] 00000000 00000010 00000000 00000101				
$00D0C8_H$	GTUC11 [R/W] 00000000 00000000 00000000 00000000				
$00D0CC_H$ - $00D0FC_H$	reserved (11)				

Address	Register				Block
	+0	+1	+2	+3	
$00D100_H$	CCSV [R] 00000000 00010000 01000000 00000000				**FlexRay SUC**
$00D104_H$	CCEV [R] 00000000 00000000 00000000 00000000				
$00D108_H$- $00D10C_H$	reserved (2)				
$00D110_H$	SCV [R] 00000000 00000000 00000000 00000000				**FlexRay GTU**
$00D114_H$	MTCCV [R] 00000000 00000000 00000000 00000000				
$00D118_H$	RCV [R] 00000000 00000000 00000000 00000000				
$00D11C_H$	OCV [R] 00000000 00000000 00000000 00000000				
$00D120_H$	SFS [R] 00000000 00000000 00000000 00000000				
$00D124_H$	SWNIT [R] 00000000 00000000 00000000 00000000				
$00D128_H$	ACS [R/W] 00000000 00000000 00000000 00000000				
$00D12C_H$	reserved (1)				
$00D130_H$- $00D168_H$	ESIDn[1-15] [R] 00000000 00000000 00000000 00000000				
$00D16C_H$	reserved (1)				
$00D170_H$ $00D1A8_H$	OSIDn[1-15] [R] 00000000 00000000 00000000 00000000				
$00D1AC_H$	reserved (1)				
$00D1B0_H$ $00D1B8_H$	NMVn[1-3] [R] 00000000 00000000 00000000 00000000				**FlexRay NEM**
$00D1BC_H$ - $00D2FC_H$	reserved (81)				
$00D300_H$	MRC [R/W] 00000001 10000000 00000000 00000000				**FlexRay**

Address	Register				Block
	+0	+1	+2	+3	
$00D304_H$	FRF [R/W] 00000001 10000000 00000000 00000000				MHD
$00D308_H$	FRFM [R/W] 00000000 00000000 00000000 00000000				
$00D30C_H$	FCL [R/W] 00000000 00000000 0000000 10000000				
$00D310_H$	MHDS [R/W] 00000000 00000000 0000000 10000000				
$00D314_H$	LDTS [R] 00000000 00000000 00000000 00000000				
$00D318_H$	FSR [R] 00000000 00000000 00000000 00000000				FlexRay MHD
$00D31C_H$	MHDF [R/W] 00000000 00000000 00000000 00000000				
$00D320_H$	TXRQ1 [R] 00000000 00000000 00000000 00000000				
$00D324_H$	TXRQ2 [R] 00000000 00000000 00000000 00000000				
$00D328_H$	TXRQ3 [R] 00000000 00000000 00000000 00000000				
$00D32C_H$	TXRQ4 [R] 00000000 00000000 00000000 00000000				
$00D330_H$	NDAT1 [R] 00000000 00000000 00000000 00000000				
$00D334_H$	NDAT2 [R] 00000000 00000000 00000000 00000000				
$00D338_H$	NDAT3 [R] 00000000 00000000 00000000 00000000				
$00D33C_H$	NDAT4 [R] 00000000 00000000 00000000 00000000				
$00D340_H$	MBSC1 [R] 00000000 00000000 00000000 00000000				
$00D344_H$	MBSC2 [R] 00000000 00000000 00000000 00000000				

Address	Register				Block
	+0	+1	+2	+3	
$00D348_H$	MBSC3 [R] 00000000 00000000 00000000 00000000				
$00D34C_H$	MBSC4 [R] 00000000 00000000 00000000 00000000				
$00D350_H$- $00D3EC_H$	reserved (40)				
$00D3F0_H$	CREL [R] 00010000 00000110 00000101 00011001				**FlexRay GIF**
$00D3F4_H$	ENDN [R] 10000111 01100101 0100011 00100001				
$00D3F8_H$- $00D3FC_H$	reserved (2)				
$00D400_H$- $00D4FC_H$	WRDSn[1-64] [R/W] 00000000 00000000 0000000 00000000				**FlexRay IBF**
$00D500_H$	WRHS1 [R/W] 00000000 00000000 00000000 00000000				
$00D504_H$	WRHS2 [R/W] 00000000 00000000 00000000 00000000				
$00D508_H$	WRHS3 [R/W] 00000000 00000000 00000000 00000000				
$00D50C_H$	reserved (1)				
$00D510_H$	IBCM [R/W] 00000000 00000000 00000000 00000000				
$00D514_H$	IBCR [R/W] 00000000 00000000 00000000 00000000				
$00D518_H$- $00D5FC_H$	reserved (58)				
$00D600_H$ $00D6FC_H$	RDDSn[1-64] [R] 00000000 00000000 00000000 00000000				**FlexRay OBF**
$00D700_H$	RDHS1 [R] 00000000 00000000 00000000 00000000				
$00D704_H$	RDHS2 [R] 00000000 00000000 00000000 00000000				
$00D708_H$	RDHS3 [R] 00000000 00000000 00000000 00000000				

Address	Register				Block
	+0	+1	+2	+3	
$00D70C_H$	MBS [R] 00000000 00000000 00000000 00000000				
$00D710_H$	OBCM [R/W] 00000000 00000000 00000000 00000000				
$00D714_H$	OBCR [R/W] 00000000 00000000 00000000 00000000				
$00D718_H$- $00D7FC_H$	reserved (58)				
$00D800_H$- $00EFFC_H$	reserved				
$00F000_H$	BCTRL [R/W] - - - - - - - - - - - - - - - - 11111100 00000000				EDSU / MPU
$00F004_H$	BSTAT [R/W] - - - - - - - - - - - - - 000 00000000 10 - - 0000				
$00F008_H$	BIAC [R] - - - - - - - - - - - - - - - - 00000000 00000000				
$00F00C_H$	BOAC [R] - - - - - - - - - - - - - - - - 00000000 00000000				
$00F010_H$	BIRQ [R/W] - - - - - - - - - - - - - - - - 00000000 00000000				
$00F014_H$ - $00F01C_H$	reserved				
$00F020_H$	BCR0 [R/W] - - - - - - - - 00000000 00000000 00000000				
$00F024_H$	BCR1 [R/W] - - - - - - - - 00000000 00000000 00000000				
$00F028_H$	BCR2 [R/W] - - - - - - - - 00000000 00000000 00000000				
$00F02C_H$	BCR3 [R/W] - - - - - - - - 00000000 00000000 00000000				
$00F030_H$ - $00F07C_H$	reserved				
$00F080_H$	BAD0 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				

Address	Register				Block
	+0	+1	+2	+3	
$00F084_H$	BAD1 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F088_H$	BAD2 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F08C_H$	BAD3 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F090_H$	BAD4 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F094_H$	BAD5 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F098_H$	BAD6 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F09C_H$	BAD7 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F0A0_H$	BAD8 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F0A4_H$	BAD9 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F0A8_H$	BAD10 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F0AC_H$	BAD11 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F0B0_H$	BAD12 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F0B4_H$	BAD13 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F0B8_H$	BAD14 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F0BC_H$	BAD15 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00F0C0_H$ - $01FFFC_H$	reserved				
020000_H - $02FFFC_H$	MB91F465XA D-RAM size is 16kB : $02C000_H$ - $02FFFC_H$ (data access is 0 waitcycles)				D-RAM 64 kB

Address	Register				Block
	+0	+1	+2	+3	
030000_H - $03FFFC_H$	MB91F465XA I-/D-RAM size is 16kB : 030000_H - $033FFC_H$ (instruction access is 0 waitcycles, data access is 1 waitcycle)				GP-RAM 64 kB
040000_H - $07FFFC_H$	reserved				
080000_H - $09FFFC_H$	ROMS02 area (128kB)				
$0A0000_H$ - $0BFFFC_H$	ROMS03 area (128kB)				
$0C0000_H$ - $0DFFFC_H$	ROMS04 area (128kB)				
$0E0000_H$ - $0FFFF4_H$	ROMS05 area (128kB)				
$0FFFF8_H$	FMV [R] 06 00 00 00_H				Fixed Reset/Mode Vector
$0FFFFC_H$	FRV [R] 00 00 BF $F8_H$				
100000_H - $147FFC_H$	reserved				
148000_H - $14FFFC_H$	ROMS07 area (32kB)				
150000_H - $4FFFFC_H$	reserved				
Write operations to address $0FFFF8_H$ and $0FFFFC_H$ are not possible. When reading these addresses, the values shown above will be read.					

7 Electrical Characteristics

7.1 Absolute Maximum Ratings

Parameter	Symbol	min.	max.	Unit	Condition
Digital supply voltage	VDD-VSS	-0.3	6.0	V	
Storage temperature	T_{ST}	-55	125	°C	
Power consumption	P_{TOT}		1000	mW	T_A = 25°C
Digital input voltage	V_{IDIG}	VSS-0.3 *	VDD+0.3	V	
Analog input voltage	V_{IA}	AVSS-0.3 *	AVCC+0.3	V	AVCC = AVRH
Analog supply voltage	AVCC-AVSS	-0.3	5.8	V	
Analog reference voltage	AVRH-AVSS	-0.3	5.8	V	
Static DC current into digital I/O	$I_{I/ODC}$	-2	2	mA	$\Sigma I_{I/ODC} < I_{SRUN}$
Relationship of the supply voltages	AVCC	VDD - 0.3	VDD + 0.3	V	At least one of the pins P27[3:0], P28[4:0] or P29[7:0] (AN*) is used as digital input or output
		VSS - 0.3	VDD + 0.3	V	All of the pins P27[3:0], P28[4:0] or P29[7:0] (AN*) follow the condition of V_{IA}

***** Making full use of the allowed static DC current into digital I/Os will lead to lower values here.

7.2 Operating Conditions

Parameter	**Symbol**	**min.**	**typ.**	**max.**	**Unit**	**Condition**
Operating temperature	T_{OP}	-40		105	°C	
Supply voltage - Digital supply	VDD-VSS	3.0		5.5	V	VSS = 0V Internal voltage regulator: VDD_{CORE} = 1.8V
- Analog supply	AVCC-AVSS	3.0		5.5	V	AVSS = 0V

Current consumption						
- Run mode	I_{sRUN}		125		mA	
- Sleep mode	I_{sSLEEP}		25		mA	
- RTC mode	I_{sRTC}		100 40 500 250		µA µA µA µA	f = 4MHz, T_A=25°C f ≤100kHz, T_A=25°C f = 4MHz, T_A=105°C f ≤100kHz, T_A=105°C
- Stop mode	I_{sSTOP}		30 200		µA µA	T_A = 25°C T_A = 105°C
Digital Inputs [1]						all voltage ranges valid at 4.5V ≤ VDD ≤ 5.5V
CMOS Schmitt-Trigger (CH)						
- High voltage range	V_{IH}	0.7*VDD		VDD	V	
- Low voltage range	V_{IL}	VSS		0.3*VDD	V	
CMOS Schmitt-Trigger (CH2)						
- High voltage range	V_{IH}	0.8*VDD		VDD	V	
- Low Voltage range	V_{IL}	VSS		0.2*VDD	V	
CMOS Automotive Schmitt-Trigger (A)						
- High voltage range	V_{IH}	0.8*VDD		VDD	V	
- Low voltage range	V_{IL}	VSS		0.5*VDD	V	
- hysteresis voltage		0.2		0.5	V	
TTL (TTL)						
- High voltage range	V_{IH}	2.0		VDD	V	
- Low voltage range	V_{IL}	VSS		0.8	V	
All inputs						
- Input capacitance	C_{IN}			tbd	pF	T_A = 25°C
- Input leakage current	I_{IL}	-1	50	1	µA	
- Pull up resistor	R_{up}		50		kΩ	
- Pull down resistor	R_{down}				kΩ	
Digital outputs						4.5V ≤ VDD ≤ 5.5V: I_{LOAD} = ±2mA / ±5mA
- Output "H" voltage	V_{OH}	VDD-0.5		VDD	V	
- Output "L" voltage	V_{OL}	VSS		VSS+0.4	V	3.0V ≤ VDD ≤ 4.5V: I_{LOAD} = ±1.6mA / ±3mA
Digital outputs I^2C Ports (open drain)						4.5V ≤ VDD ≤ 5.5V: I_{LOAD} = ±3mA
- Output "H" voltage	V_{OH}	-		-	V	3.0V ≤ VDD ≤ 4.5V:
- Output "L" voltage	V_{OL}	VSS		VSS+0.4	V	I_{LOAD} = ±2mA

ADC inputs [2]						
- Reference voltage input	AVRH AVRL	AVCC*0.75 AVSS		AVCC AVCC*0.25	V V	
- Input voltage range	V_{imax} V_{imin}	 AVRL		AVRH	V V	
- Input resistance	R_I			2.6 12.1	kΩ kΩ	4.5V ≤ AVCC ≤ 5.5V 3.0V ≤ AVCC ≤ 4.5V
- Input capacitance	C_I			8.5	pF	
- Impedance of external output driving the ADC input				100	kΩ	
- Input leakage current	I_{IL}	-1		1	µA	T_A = 25°C
Lock-up time PLL1 (4MHz->16…100MHz)				0.6	ms	
Lock-up time PLL2 (4MHz->80MHz)				0.6	ms	
ESD Protection (Human body model)	V_{surge}	2			kV	$R_{discharge}$ = 1.5kΩ $C_{discharge}$ = 100pF
RC Oscillator	$f_{RC100KHz}$ f_{RC2MHz}	50 1	100 2	200 4	kHz MHz	VDD_{CORE} >=1.65V

[1] Valid for bidirectional tristate I/O PAD cell

[2] The protection diodes at the analog inputs are connected to the digital supply voltage

7.3 Converter Characteristics

- **Table 11:** A/D Converter

Parameter	Symbol	Rating			Unit	Remark
		Minimum	Typical	Maximum		
Resolution				10	Bit	
Conversion error				+/- 3.0	LSB	Overall error
Non-linearity				+/-2.5	LSB	
Differential Non-linearity				+/-1.9	LSB	
Zero Reading voltage	V_{0T}	AVRL -1.5	AVRL+0.5	AVRL+2.5	LSB	t.b.d.
Full scale reading voltage	V_{FST}	AVRH-3.5	AVRH-1.5	AVRH+0.5	LSB	
Input current	IA @ AVCC		2.4	4.7	mA	
Reference voltage current	IR		0.65	1.0	mA	

Table 11: Sampling Time Calculation

Tsamp = (2.6 kOhm + Rext) * 8.5pF * 7; for 4.5V ≤ Avcc ≤ 5.5V
Tsamp = (12.1 kOhm + Rext) * 8.5pF * 7; for 3.0V ≤ Avcc ≤ 4.5V

8 E-Ray Programmer's Model

This chapter is the E-Ray Programmer's Model Revision 1.2.3. Used with permission by the Robert Bosch GmbH.

8.1 Register Map

The E-Ray module allocates an address space of 2 Kbytes (0xD000 to 0xD7FF). The registers are organized as 32-bit registers. 8/16-bit accesses are also supported. Host access to the Message RAM is done via the Input and Output Buffers. They buffer data to be transferred to and from the Message RAM under control of the Message Handler, avoiding conflicts between Host accesses and message reception / transmission. Addresses 0xD000 to 0xD00F are reserved for customer specific purposes. All functions related to these addresses are located in the Customer CPU Interface. The test registers located on address 0xD010 and 0xD014 are writeable only under the conditions described in (use when TMC[1:0]=01)
Special Registers.
The assignment of the message buffers is done according to the scheme shown in Assignment of message buffers below. The number N of available message buffers depends on the payload length of the configured message buffers. The maximum number of message buffers is 128. The maximum payload length supported is 254 bytes.

The message buffers are separated into three consecutive groups:

- Static Buffers - Transmit / receive buffers assigned to static segment
- Static + Dynamic Buffers - Transmit / receive buffers assigned to static or dynamic segment
- FIFO - Receive FIFO

The message buffer separation configuration can be changed only in DEFAULT_CONFIG or CONFIG state only by programming register MRC (see Message RAM Configuration (MRC)).
The first group starts with message buffer 0 and consists of static message buffers only. Message buffer 0 is dedicated to hold the startup / sync frame or the single slot frame, if the node transmits one, as configured by **SUCC1.TXST**, **SUCC1.TXSY**, and **SUCC1.TSM**. In addition, message buffer 1 may be used for sync frame transmission in case that sync frames or single-slot frames should have different payloads on the two channels. In this case bit **MRC.SPLM** has to be programmed to '1' and message buffers 0 and 1 have to be configured with the key slot ID and can be (re)configured in DEFAULT_CONFIG or CONFIG state only.
The second group consists of message buffers assigned to the static or to the dynamic segment. Message buffers belonging to this group may be reconfigured during run time from dynamic to static or vice versa depending on the state of **MRC.SEC[1:0]**.

The message buffers belonging to the third group are concatenated to a single receive FIFO.

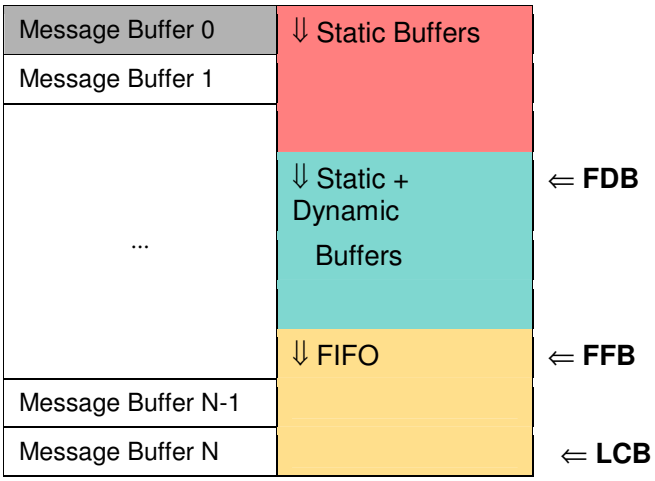


Table 11: Assignment of message buffers

Address	Symbol	Name	Page	Reset	Acc	Block
Customer Registers						
0xD000	see Customer CPU Interface Specification					CIF
0xD004						
0xD008						
0xD00C						
Special Registers						
0xD010	TEST1	Test Register 1	Test Register 1 (TEST1)	0000 0300	r/w	GIF
0xD014	TEST2	Test Register 2	Test Register 2 (TEST2)	0000 0000	r/w	
0xD018		*reserved (1)*		0000 0000	r	
0xD01C	LCK	Lock Register	Lock Register (LCK)	0000 0000	r/w	GIF
Interrupt Registers						
0xD020	EIR	Error Interrupt Register	Error Interrupt Register (EIR)	0000 0000	r/w	INT
0xD024	SIR	Status Interrupt Register	Status Interrupt Register (SIR)	0000 0000	r/w	
0xD028	EILS	Error Interrupt Line Select	Error Interrupt Line Select (EILS)	0000 0000	r/w	
0xD02C	SILS	Status Interrupt Line Select	Status Interrupt Line Select (SILS)	0303 FFFF	r/w	

0xD030	EIES	Error Interrupt Enable Set	Error Interrupt Enable Set / Reset (EIES, EIER)	0000 0000	r/w	
0xD034	EIER	Error Interrupt Enable Reset	Error Interrupt Enable Set / Reset (EIES, EIER)	0000 0000	r/w	
0xD038	SIES	Status Interrupt Enable Set	Status Interrupt Enable Set / Reset (SIES, SIER)	0000 0000	r/w	
0xD03C	SIER	Status Interrupt Enable Reset	Status Interrupt Enable Set / Reset (SIES, SIER)	0000 0000	r/w	
0xD040	ILE	Interrupt Line Enable	Interrupt Line Enable (ILE)	0000 0000	r/w	
0xD044	T0C	Timer 0 Configuration	Timer 0 Configuration (T0C)	0000 0000	r/w	
0xD048	T1C	Timer 1 Configuration	Timer 1 Configuration (T1C)	0002 0000	r/w	
0xD04C	STPW1	Stop Watch Register 1	Stop Watch Register 1	0000 0000	r/w	
0xD050	STPW2	Stop Watch Register 2	Stop Watch Register 2 (STPW2)	0000 0000	r/w	
0xD054 - 0xD07C		*reserved (11)*		0000 0000	r	
CC Control Registers						
0xD080	SUCC1	SUC Configuration Register 1	SUC Configuration Register 1 (SUCC1)	0C40 1080	r/w	SUC
0xD084	SUCC2	SUC Configuration Register 2	SUC Configuration Register 2 (SUCC2)	0100 0504	r/w	
0xD088	SUCC3	SUC Configuration Register 3	SUC Configuration Register 3 (SUCC3)	0000 0011	r/w	
0xD08C	NEMC	NEM Configuration Register	NEM Configuration Register (NEMC)	0000 0000	r/w	NEM
0xD090	PRTC1	PRT Configuration Register 1	PRT Configuration Register 1 (PRTC1)	084C 0633	r/w	PRT
0xD094	PRTC2	PRT Configuration Register 2	PRT Configuration Register 2 (PRTC2)	0F2D 0A0E	r/w	
0xD098	MHDC	MHD Configuration Register	MHD Configuration Register (MHDC)	0000 0000	r/w	MHD
0xD09C		*reserved (1)*		0000 0000	r	
0xD0A0	GTUC1	GTU Configuration Register 1	GTU Configuration Register 1 (GTUC1)	0000 0280	r/w	GTU
0xD0A4	GTUC2	GTU Configuration Register 2	GTU Configuration Register 2 (GTUC2)	0002 000A	r/w	
0xD0A8	GTUC3	GTU Configuration Register 3	GTU Configuration Register 3 (GTUC3)	0202 0000	r/w	

0xD0AC	GTUC4	GTU Configuration Register 4	GTU Configuration Register 4 (GTUC4)	0008 0007	r/w	
0xD0B0	GTUC5	GTU Configuration Register 5	GTU Configuration Register 5 (GTUC5)	0E00 0000	r/w	
0xD0B4	GTUC6	GTU Configuration Register 6	GTU Configuration Register 6 (GTUC6)	0002 0000	r/w	
0xD0B8	GTUC7	GTU Configuration Register 7	GTU Configuration Register 7 (GTUC7)	0002 0004	r/w	
0xD0BC	GTUC8	GTU Configuration Register 8	GTU Configuration Register 8 (GTUC8)	0000 0002	r/w	
0xD0C0	GTUC9	GTU Configuration Register 9	GTU Configuration Register 9 (GTUC9)	0000 0101	r/w	
0xD0C4	GTUC10	GTU Configuration Register 10	GTU Configuration Register 10 (GTUC10)	0002 0005	r/w	
0xD0C8	GTUC11	GTU Configuration Register 11	GTU Configuration Register 11 (GTUC11)	0000 0000	r/w	
0xD0CC - 0xD0FC		*reserved (13)*		0000 0000	r	
CC Status Registers						
0xD100	CCSV	CC Status Vector	CC Status Vector (CCSV)	0010 4000	r	SUC
0xD104	CCEV	CC Error Vector	CC Error Vector (CCEV)	0000 0000	r	
0xD108 - 0xD10C		*reserved (2)*		0000 0000	r	
0xD110	SCV	Slot Counter Value	Slot Counter Value (SCV)	0000 0000	r	GTU
0xD114	MTCCV	Macrotick and Cycle Counter Value	Macrotick and Cycle Counter Value (MTCCV)	0000 0000	r	
0xD118	RCV	Rate Correction Value	Rate Correction Value (RCV)	0000 0000	r	
0xD11C	OCV	Offset Correction Value	Offset Correction Value (OCV)	0000 0000	r	
0xD120	SFS	Sync Frame Status	Sync Frame Status (SFS)	0000 0000	r	
0xD124	SWNIT	Symbol Window and NIT Status	Symbol Window and NIT Status (SWNIT)	0000 0000	r	
0xD128	ACS	Aggregated Channel Status	Aggregated Channel Status (ACS)	0000 0000	r/w	
0xD12C		*reserved (1)*		0000 0000	r	
0xD130 - 0xD168	ESIDn	Even Sync ID [1...15]	Even Sync ID [1	0000 0000	r	
0xD16C		*reserved (1)*		0000 0000	r	
0xD170 - 0xD1A8	OSIDn	Odd Sync ID [1...15]	Odd Sync ID [1	0000 0000	r	

0xD1AC		*reserved (1)*		0000 0000	r	
0xD1B0 - 0xD1B8	NMVn	Network Management Vector [1...3]	Network Management Vector [1	0000 0000	r	NEM
0xD1BC - 0xD2FC		*reserved (81)*		0000 0000	r	
Message Buffer Control Registers						
0xD300	MRC	Message RAM Configuration	Message RAM Configuration (MRC)	0180 0000	r/w	MHD
0xD304	FRF	FIFO Rejection Filter	FIFO Rejection Filter (FRF)	0180 0000	r/w	
0xD308	FRFM	FIFO Rejection Filter Mask	FIFO Rejection Filter Mask (FRFM)	0000 0000	r/w	
0xD30C	FCL	FIFO Critical Level	FIFO Critical Level (FCL)	0000 0080	r/w	
Message Buffer Status Registers						
0xD310	MHDS	Message Handler Status	Message Handler Status (MHDS)	0000 0080	r/w	MHD
0xD314	LDTS	Last Dynamic Transmit Slot	Last Dynamic Transmit Slot (LDTS)	0000 0000	r	
0xD318	FSR	FIFO Status Register	FIFO Status Register (FSR)	0000 0000	r	
0xD31C	MHDF	Message Handler Constraints Flags	Message Handler Constraints Flags (MHDF)	0000 0000	r/w	
0xD320	TXRQ1	Transmission Request 1	Transmission Request 1/2/3/4 (TXRQ1/2/3/4)	0000 0000	r	
0xD324	TXRQ2	Transmission Request 2	Transmission Request 1/2/3/4 (TXRQ1/2/3/4)	0000 0000	r	
0xD328	TXRQ3	Transmission Request 3	Transmission Request 1/2/3/4 (TXRQ1/2/3/4)	0000 0000	r	
0xD32C	TXRQ4	Transmission Request 4	Transmission Request 1/2/3/4 (TXRQ1/2/3/4)	0000 0000	r	
0xD330	NDAT1	New Data 1	New Data 1/2/3/4 (NDAT1/2/3/4)	0000 0000	r	
0xD334	NDAT2	New Data 2	New Data 1/2/3/4 (NDAT1/2/3/4)	0000 0000	r	
0xD338	NDAT3	New Data 3	New Data 1/2/3/4 (NDAT1/2/3/4)	0000 0000	r	
0xD33C	NDAT4	New Data 4	New Data 1/2/3/4 (NDAT1/2/3/4)	0000 0000	r	
0xD340	MBSC1	Message Buffer Status Changed 1	Message Buffer Status Changed 1/2/3/4 (MBSC1/2/3/4)	0000 0000	r	
0xD344	MBSC2	Message Buffer Status Changed 2	Message Buffer Status Changed 1/2/3/4 (MBSC1/2/3/4)	0000 0000	r	

0xD348	MBSC3	Message Buffer Status Changed 3	Message Buffer Status Changed 1/2/3/4 (MBSC1/2/3/4)	0000 0000	r	
0xD34C	MBSC4	Message Buffer Status Changed 4	Message Buffer Status Changed 1/2/3/4 (MBSC1/2/3/4)	0000 0000	r	
0xD350 - 0xD3EC		*reserved (40)*		0000 0000	r	
Identification Registers						
0xD3F0	CREL	Core Release Register	Core Release Register (CREL)	[release info]	r	GIF
0xD3F4	ENDN	Endian Register	Endian Register (ENDN)	8765 4321	r	
0xD3F8 - 0xD3FC		*reserved (2)*		0000 0000	r	
Input Buffer						
0xD400 - 0xD4FC	WRDSn	Write Data Section [1...64]	Write Data Section [1	0000 0000	r/w	IBF
0xD500	WRHS1	Write Header Section 1	Write Header Section 1 (WRHS1)	0000 0000	r/w	
0xD504	WRHS2	Write Header Section 2	Write Header Section 2 (WRHS2)	0000 0000	r/w	
0xD508	WRHS3	Write Header Section 3	Write Header Section 3 (WRHS3)	0000 0000	r/w	
0xD50C		*reserved (1)*		0000 0000	r/w	
0xD510	IBCM	Input Buffer Command Mask	Input Buffer Command Mask (IBCM)	0000 0000	r/w	
0xD514	IBCR	Input Buffer Command Request	Input Buffer Command Request (IBCR)	0000 0000	r/w	
0xD518 - 0xD5FC		*reserved (58)*		0000 0000	r	

Output Buffer						
0xD600 - 0xD6FC	RDDSn	Read Data Section [1...64]	Read Data Section [1	0000 0000	r	OBF
0xD700	RDHS1	Read Header Section 1	Read Header Section 1 (RDHS1)	0000 0000	r	
0xD704	RDHS2	Read Header Section 2	Read Header Section 2 (RDHS2)	0000 0000	r	
0xD708	RDHS3	Read Header Section 3	Read Header Section 3 (RDHS3)	0000 0000	r	
0xD70C	MBS	Message Buffer Status	Message Buffer Status (MBS)	0000 0000	r	
0xD710	OBCM	Output Buffer Command Mask	Output Buffer Command Mask (OBCM)	0000 0000	r/w	
0xD714	OBCR	Output Buffer Command Request	Output Buffer Command Request (OBCR)	0000 0000	r/w	
0xD718 - 0xD7FC		*reserved (58)*		0000 0000	r	

Table 12: E-Ray register map

8.2 Customer Registers

The address space from 0xD000 to 0xD00F is reserved for customer-specific registers. These registers, if implemented, are located in the Customer CPU Interface block. A description can be found in the specific Customer CPU Interface specification document.

Specification of the CIF registers is as follows:

```
// CIF0 = 0xD000, CIF1 = 0xD004, CIF2 = 0xD008 und CIF3 = 0xD00C

// ************************************************************************
// ** Customer interface logic                                           **
// ************************************************************************

//       31                                                      0
//      +-------+-------+-------+-------+-------+-------+-------+-------+
// CIF0 |                            VERSION                            |
//      +-------+-------+-------+-------+-------+-------+-------+-------+
//       R

//       31      30      29      28      27      26      25      24
//        7       6       5       4       3       2       1       0
//      +-------+-------+-------+-------+-------+-------+-------+-------+
// CIF1 | DREQO | DLVLO | DMODO | DENBO | DREQI | DLVLI | DMODI | DENBI | initial
//      +-------+-------+-------+-------+-------+-------+-------+-------+ 00000000
//       R/W(RM1) R/W     R/W     R/W    R/W(RM1) R/W     R/W     R/W

//       23      22      21      20      19      18      17      16
//        7       6       5       4       3       2       1       0
//      +-------+-------+-------+-------+-------+-------+-------+-------+
// CIF1 |       |       |       | MASK4 | MASK3 | MASK2 | MASK1 | MASK0 | initial
//      +-------+-------+-------+-------+-------+-------+-------+-------+ 00000000
//        R0/W0   R0/W0   R0/W0   R/W     R/W     R/W     R/W     R/W

//       15      14      13      12      11      10       9       8
//        7       6       5       4       3       2       1       0
//      +-------+-------+-------+-------+-------+-------+-------+-------+
// CIF1 | RTEST |       |       | SWAP  | TREQ1 | TENB1 | TREQ0 | TENB0 | initial
//      +-------+-------+-------+-------+-------+-------+-------+-------+ 00000000
//       R0/W0   R0/W0   R0/W0   R/W    R/W(RM1) R/W    R/W(RM1) R/W

// R           Read only register
// R/W         Read/write register
// R/W(RM1)    Read/write register (outputs '1' on RMW)
// R0/W0       Read only register (always outputs '0', write '0' is recommended)

// All other CIF registers (CIF2 and CIF3) not mentioned here are R0/W0.



   parameter  VERSION           = 32'h04_FF_5B_FF ;

// 0x04 : Fujitsu
// 0xFF : see Boot-ROM Device-ID
// 0x5B : FR:91(0x5B), FX:96(0x60)
// 0xFF : see E-Ray-ID
```

8.2.1 CIF register functions

DENBI: DMA request enable on IBF
DENBO: DMA request enable on OBF

DENBx = 0 : DMA request is disabled
DENBx = 1 : DMA request is enabled

DMODI: DMA request mode on IBF
DMODO: DMA request mode on OBF

DMODx = 0 : DMA request mode is **eray_ibusy**/**eray_obusy** level mode
DMODx = 1 : DMA request mode is **eray_ibusy**/**eray_obusy** edge mode

DLVLI: DMA level/edge selector on IBF
DLVLO: DMA level/edge selector on OBF

with DMODx = 0
DLVLx = 0 : DMA request level is non-inverted **eray_ibusy**/**eray_obusy**
DLVLx = 1 : DMA request level is inverted **eray_ibusy**/**eray_obusy**

with DMODx = 1
DLVLx = 0 : DMA request is negative edge of **eray_ibusy**/**eray_obusy**
DLVLx = 1 : DMA request is positive edge of **eray_ibusy**/**eray_obusy**

DREQI: DMA request flag on IBF
DREQO: DMA request flag on OBF

with DMODx = 0
DREQx = Read only, displays the **eray_ibusy**/**eray_obusy** level
- displays the modified **eray_ibusy**/**eray_obusy** level if changed with DLVLx
- On a read-modify-write bit-operation '1' is read

with DMODx = 1
DREQx = 0 : DMA request is inactive
DREQx = 1 : DMA request is active
- Request is automatically cleared to '0' if a DMA transfer has started
- Possible to write '0' to clear the DMA request by the CPU
- On a read-modify-write bit-operation '1' is read

MASK0: DMA Channel 0 Interrupt Mask (for OBF configuration)

MASK1: DMA Channel 1 Interrupt Mask (for OBF configuration)
MASK2: DMA Channel 2 Interrupt Mask (for OBF configuration)
MASK3: DMA Channel 3 Interrupt Mask (for OBF configuration)
MASK4: DMA Channel 4 Interrupt Mask (for OBF configuration)

MASKx = 0 : DMA channel x interrupt is not masked
MASKx = 1 : DMA channel x interrupt is masked while **eray_obusy** = 1

TENB0: Timer 0 Interrupt Enable
TENB1: Timer 1 Interrupt Enable

TENBx = 0 : Direct **eray_tint0**/**eray_tint1** signal is used for interrupt generation
TENBx = 1 : Registered TREQx flag is used for interrupt generation

TREQ0: Timer 0 Interrupt Request
TREQ1: Timer 1 Interrupt Request

with TENBx = 0
TREQx = Read only, displays the **eray_tint0**/**eray_tint1** level
- On a read-modify-write bit-operation '1' is read

with TENBx = 1
TREQx = 0 : Timer interrupt request is inactive
TREQx = 1 : Timer interrupt request is active
- Possible to write '0' to clear the interrupt request by the CPU
- On a read-modify-write bit-operation '1' is read

SWAP: IBF/OBF data swap enable

SWAP = 0 : Read and write data on IBF/OBF is not swapped
SWAP = 1 : Read and write data on IBF/OBF is swapped

```
SWAP = 0                          | SWAP = 1
----------------------------------+-------------------------------
MD[ 7: 0] = DW(n),   byte(n-1)    | MD[ 7: 0] = DW(n+1), byte(n+2)
MD[15: 8] = DW(n),   byte(n)      | MD[15: 8] = DW(n+1), byte(n+1)
MD[23:16] = DW(n+1), byte(n+1)    | MD[23:16] = DW(n),   byte(n)
MD[31:24] = DW(n+1), byte(n+2)    | MD[31:24] = DW(n),   byte(n-1)
```

RTEST: RAM Test address range enable

RTEST = 0 : Normal operation address mapping
RTEST = 1 : RAM Test operation address mapping (use when TMC[1:0]=01)

8.3 Special Registers

8.3.1 Test Register 1 (TEST1)

The Test Register 1 holds the control bits to configure the test modes of the E-Ray module. Write access to these bits is only possible if bit **WRTEN** is set to '1'.
When the E-Ray IP is operated in one of its test modes that requires **WRTEN** to be set (RAM Test Mode, I/O Test Mode, Asynchronous Transmit Mode, and Loop Back Mode) only the selected test mode functionality is available.
The test functions are not available in addition to the normal operational mode functions, they change the functions of parts of the E-Ray module. Therefore normal operation as specified outside this chapter and as required by the FlexRay protocol specification and the FlexRay conformance test is not possible. Test mode functions may not be combined with each other or with FlexRay protocol functions.
The test mode features are intended for hardware testing or for FlexRay bus analyzer tools. They are not intended to be used in FlexRay applications.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEST1	R	CERB3	CERB2	CERB1	CERB0	CERA 3	CERA 2	CERA 1	CERA 0	0	0	TXEN B	TXEN A	TXB	TXA	RXB	RXA
0xD010	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	AOB	AOA	0	0	TMC1	TMC0	0	0	ELBE	WRTEN
	W																
Reset		0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0

WRTEN

Write Test Register Enable

Enables write access to the test registers. To set the bit from '0' to '1' the test mode key has to be written as defined in Section Lock Register (LCK). The unlock sequence is not required when **WRTEN** is kept at '1' while other bits of the register are changed. The bit can be reset to '0' at any time.

1 = Write access to test registers enabled

0 = Write access to test registers disabled

ELBE

External Loop Back Enable

There are two possibilities to perform a loop back test. External loop back via physical layer or internal loop back for in-system self-test (default). In case of an internal loop back pins **eray_txen1,2_n** are in their inactive state, pins **eray_txd1,2** are set to HIGH, pins **eray_rxd1,2** are not evaluated. Bit **ELBE** is evaluated only when POC is in loop back mode and test multiplexer control is in non-multiplexing mode **TMC[1:0]** = "00".

1 = External loop back
0 = Internal loop back (default)

TMC[1:0]

Test Multiplexer Control

00, 11= Normal signal path (default)

01 = RAM Test Mode - Internal busses are multiplexed to make all RAM blocks of the E-Ray module directly accessible by the Host. This mode is intended to enable testing of the embedded RAM blocks during production testing.

10 = I/O Test Mode - Output pins **eray_txd1**, **eray_txd2**, **eray_txen1_n**, **eray_txen2_n**, are driven to the values defined by bits **TXA**, **TXB**, **TXENA**, **TXENB**. The values applied to the input pins **eray_rxd1**, **eray_rxd2** can be read from register bits **RXA**, **RXB**.

AOA

Activity on A

The channel idle condition is specified in the FlexRay protocol spec v2.1, chapter 3, BITSTRB process zChannelIdle.

1 = Activity detected, channel A not idle
0 = No activity detected, channel A idle

AOB

Activity on B

The channel idle condition is specified in the FlexRay protocol spec v2.1, chapter 3, BITSTRB process zChannelIdle.

1 = Activity detected, channel B not idle
0 = No activity detected, channel B idle

CERA[3:0]

Coding Error Report Channel A

Set when a coding error is detected on channel A. Reset to zero when register TEST1 is read or written. Once the **CERA[3:0]** is set it will remain unchanged until the Host accesses the TEST1 register.

0000 = No coding error detected
0001 = Header CRC error detected
0010 = Frame CRC error detected
0011 = Frame Start Sequence FSS too long
0100 = First bit of Byte Start Sequence BSS seen LOW
0101 = Second bit of Byte Start Sequence BSS seen HIGH
0110 = First bit of Frame End Sequence FES seen HIGH
0111 = Second bit of Frame End Sequence FES seen LOW
1000 = CAS / MTS symbol seen too short
1001 = CAS / MTS symbol seen too long
1010...1111 = reserved

CERB[3:0]

Coding Error Report Channel B

Set when a coding error is detected on channel B. Reset to zero when register TEST1 is read or written. Once the **CERB[3:0]** is set it will remain unchanged until the Host accesses the TEST1 register.

0000 = No coding error detected
0001 = Header CRC error detected
0010 = Frame CRC error detected
0011 = Frame Start Sequence FSS too long
0100 = First bit of Byte Start Sequence BSS seen LOW
0101 = Second bit of Byte Start Sequence BSS seen HIGH
0110 = First bit of Frame End Sequence FES seen HIGH
0111 = Second bit of Frame End Sequence FES seen LOW
1000 = CAS / MTS symbol seen too short
1001 = CAS / MTS symbol seen too long
1010...1111 = reserved

Note: Coding errors are also signalled when the CC is in MONITOR_MODE. The error codes regarding CAS / MTS symbols concern only the monitored bit pattern, irrelevant whether those bit patterns are seen in the symbol window or elsewhere.

The following TEST1 bits are used to test the interface to the physical layer (connectivity test) by driving / reading the respective pins.

RXA

Monitor Channel A Receive Pin

0 = **eray_rxd1** = '0'
1 = **eray_rxd1** = '1'

RXB

Monitor Channel B Receive Pin

0 = **eray_rxd2** = '0'
1 = **eray_rxd2** = '1'

TXA

Control of Channel A Transmit Pin

0 = **eray_txd1** pin drives a '0'
1 = **eray_txd1** pin drives a '1'

TXB

Control of Channel B Transmit Pin

0 = **eray_txd2** pin drives a '0'
1 = **eray_txd2** pin drives a '1'

TXENA

Control of Channel A Transmit Enable Pin

0 = **eray_txen1_n** pin drives a '0'
1 = **eray_txen1_n** pin drives a '1'

TXENB

Control of Channel B Transmit Enable Pin

0 = **eray_txen2_n** pin drives a '0'

1 = **eray_txen2_n** pin drives a '1'

8.3.1.1 Asynchronous Transmit Mode (ATM)

The asynchronous transmit mode is entered by writing **SUCC1.CMD[3:0]** = "1110" while the CC is in CONFIG state and bit **TEST1.WRTEN** is set to '1'. This write operation has to be directly preceded by two consecutive write accesses to the Configuration Lock Key (unlock sequence). When called in any other state or when bit **TEST1.WRTEN** is not set, **SUCC1.CMD[3:0]** will be reset to "0000" = command_not_accepted. Reading **CCSV.POCS[5:0]** will return "00 1110" while the E-Ray module is in ATM mode. Asynchronous transmit mode can be left by writing **SUCC1.CMD[3:0]** = "0001" (CHI command: CONFIG).
In ATM mode transmission of a FlexRay frame is triggered by writing the number of the respective message buffer to **IBCR.IBRH[6:0]** while **IBCM.STXR** is set to '1'. In this mode wakeup, startup, and clock synchronization are bypassed. The CHI command SEND_MTS results in the immediate transmission of an MTS symbol. The cycle counter value of frames send in ATM mode can be programmed via **MTCCV.CCV[5.0]** (writeable in ATM and loop back mode only).

8.3.1.2 Loop Back Mode

The loop back mode is entered by writing **SUCC1.CMD[3:0]** = "1111" while the CC is in CONFIG state and bit **TEST1.WRTEN** is set to '1'. This write operation has to be directly preceded by two consecutive write accesses to the Configuration Lock Key (unlock sequence). When called in any other state or when **TEST1.WRTEN** is not set, **SUCC1.CMD[3:0]** will be reset to "0000" = command_not_accepted. Reading **CCSV.POCS[5:0]** will return "00 1101" while the E-Ray module is in loop back mode. Loop back mode can be left by writing **SUCC1.CMD[3:0]** = "0001" (CHI command: CONFIG).
The loop back mode is intended to check the module's internal data paths. Normal, time triggered operation is not possible in loop back mode.
There are two possibilities to perform a loop back test. External loop back via physical layer (**TEST1.ELBE** = '1') or internal loop back for in-system self-test (**TEST1.ELBE** = '0'). In case of an internal loop back pins **eray_txen1,2_n** are in their inactive state, pins **eray_txd1,2** are set to HIGH, pins **eray_rxd1,2** are not evaluated.
When the CC is in loop back mode, a loop back test is started by the Host writing a message to the Input Buffer and requesting the transmission by writing to register IBCR. The Message Handler will transfer the message into the Message RAM and then into the Transient Buffer of the selected channel. The Channel Protocol Controller (PRT) will read (in 32-bit words) the message from the transmit part of the Transient Buffer and load it into its Rx / Tx shift register. The serial transmission is looped back into the shift register; its content is written into the receive part of the channels's Transient Buffer before the next word is loaded.

The PRT and the Message Handler will then treat this transmitted message like a received message, perform an acceptance filtering on frame ID and receive channel, and store the message into the Message RAM if it passed acceptance filtering. The loop back test ends with the Host requesting this received message from the Message RAM and then checking the contents of the Output Buffer.
Each FlexRay channel is tested separately. The E-Ray cannot receive messages from the FlexRay bus while it is in the loop back mode.
The cycle counter value of frames used in loop back mode can be programmed via **MTCCV.CCV[5.0]** (writeable in ATM and loop back mode only).
Note that in case of an odd payload the last two bytes of the looped-back payload will be shifted by 16 bits to the right inside the last 32-bit data word.

8.3.2 Test Register 2 (TEST2)

The Test Register 2 holds all bits required for the RAM test of the seven embedded RAM blocks of the E-Ray module. Write access to this register is only possible when **TEST1.WRTEN** is set to '1'.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEST2	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD014	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	RDPB	WRPB	0	0	0	0	0	0	0	SSEL2	SSEL1	SSEL0	0	RS2	RS1	RS0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

RS[2:0]

RAM Select

In RAM Test mode the RAM blocks selected by **RS[2:0]** are mapped to module address 0x400 to 7FF (1024 byte addresses).

000 = Input Buffer RAM 1 (IBF1)
001 = Input Buffer RAM 2 (IBF2)
010 = Output Buffer RAM 1 (OBF1)
011 = Output Buffer RAM 2 (OBF2)
100 = Transient Buffer RAM A (TBF1)
101 = Transient Buffer RAM B (TBF2)
110 = Message RAM (MBF)
111 = unused

SSEL[2:0]

Segment Select

To enable access to the complete Message RAM (8192 byte addresses) the Message RAM is segmented.

000 = access to RAM bytes 0000h to 03FFh enabled
001 = access to RAM bytes 0400h to 07FFh enabled
010 = access to RAM bytes 0800h to 0BFFh enabled
011 = access to RAM bytes 0C00h to 0FFFh enabled
100 = access to RAM bytes 1000h to 13FFh enabled
101 = access to RAM bytes 1400h to 17FFh enabled
110 = access to RAM bytes 1800h to 1BFFh enabled
111 = access to RAM bytes 1C00h to 1FFFh enabled

WRPB

Write Parity Bit

Value of parity bit to be written to bit 32 of the addressed RAM word.

RDPB

Read Parity Bit

Value of parity bit read from bit 32 of the addressed RAM word.

8.3.2.1 RAM Test Mode

In RAM test mode (**TEST1.TMC[1:0]** = "01"), one of the seven RAM blocks can be selected for direct RD/WR access by programming **TEST2.RS[2:0]**.
For external access the selected RAM block is mapped to address space 400h to 7FF (1024 byte addresses or 256 word addresses).
Because the length of the Message RAM exceeds the available address space, the Message RAM is segmented into segments of 1024 bytes. The segments can be selected by programming **TEST2.SSEL[2:0]**.

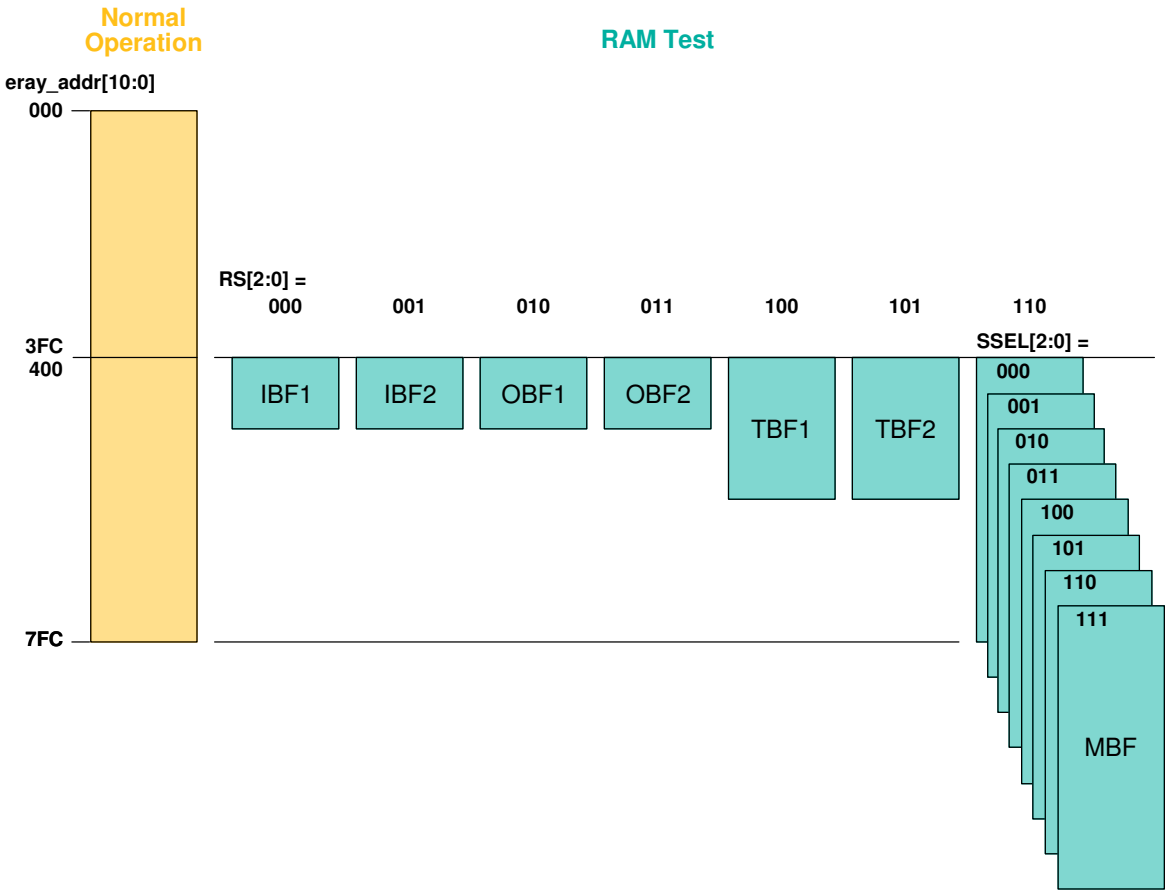


Figure 7: RAM test mode access to E-Ray RAM blocks

8.3.3 Lock Register (LCK)

The Lock Register is write-only. Reading the register will return 0x0000 0000.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LCK	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD01C	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	W	TMK7	TMK6	TMK5	TMK4	TMK3	TMK2	TMK1	TMK0	CLK7	CLK6	CLK5	CLK4	CLK3	CLK2	CLK1	CLK0
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CLK[7:0]

Configuration Lock Key

To leave CONFIG state by writing **SUCC1.CMD[3:0]** (commands READY, MONITOR_MODE, ATM, LOOP_BACK), the write operation has to be directly preceded by two write accesses to the Configuration Lock Key (unlock sequence). If the write sequence below is interrupted by other write accesses between the second write to the Configuration Lock Key and the write access to the SUCC1 register, the CC remains in CONFIG state and the sequence has to be repeated.

First write: **LCK.CLK[7:0]** = "1100 1110" (0xCE)
Second write: **LCK.CLK[7:0]** = "0011 0001" (0x31)
Third write: **SUCC1.CMD[3:0]**

TMK[7:0]

Test Mode Key

To write bit **TEST1.WRTEN**, the write operation has to be directly preceded by two consecutive write accesses to the Test Mode Key (unlock sequence). If the write sequence below is interrupted by other write accesses between the second write to the Test Mode Key and the write access to the TEST1 register, **TEST1.WRTEN** is not set to '1' and the sequence has to be repeated.

First write: **LCK.TMK[7:0]** = "0111 0101" (0x75)
Second write: **LCK.TMK[7:0]** = "1000 1010" (0x8A)
Third write: **TEST1.WRTEN** = '1'

Note: In case that the Host uses 8/16-bit accesses to write the listed bit fields, the programmer has to ensure that no "dummy accesses" e.g. to the remaining register bytes / words are inserted by the compiler.

8.4 Interrupt Registers

8.4.1 Error Interrupt Register (EIR)

The flags are set when the CC detects one of the listed error conditions. The flags remain set until the Host clears them. A flag is cleared by writing a '1' to the corresponding bit position. Writing a '0' has no effect on the flag. A hard reset will also clear the register.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EIR	R	0	0	0	0	0	TABB	LTVB	EDB	0	0	0	0	0	TABA	LTVA	EDA
0xD020	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	MHF	IOBA	IIBA	EFA	RFO	PERR	CCL	CCF	SFO	SFBM	CNA	PEMC
	W																

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

PEMC

POC Error Mode Changed

This flag is set whenever the error mode signalled by **CCEV.ERRM[1:0]** has changed.

1 = Error mode has changed

0 = Error mode has not changed

CNA

Command Not Accepted

The flag signals that the write access to the CHI command vector **SUCC1.CMD[3:0]** was not successful because the requested command was not valid in the actual POC state, or because the CHI command was locked (**CCL** = '1').

1 = CHI command not accepted

0 = CHI command accepted

SFBM

Sync Frames Below Minimum

This flag signals that the number of sync frames received during the last communication cycle was below the limit required by the FlexRay protocol. May be set during startup and therefore should be cleared by the Host after the CC entered NORMAL_ACTIVE state.

1 = Less than the required minimum of sync frames received

0 = Sync node: 1 or more sync frames received
Non-sync node: 2 or more sync frames received

SFO

Sync Frame Overflow

Set when the number of sync frames received during the last communication cycle exceeds the maximum number of sync frames as defined by **GTUC2.SNM[3:0]**.

1 = More sync frames received than configured by **GTUC2.SNM[3:0]**

0 = Number of received sync frames ≤ **GTUC2.SNM[3:0]**

CCF

Clock Correction Failure

This flag is set at the end of the cycle whenever one of the following errors occurred:

- Missing offset and / or rate correction
- Clock correction limit reached

The clock correction status is monitored in registers CCEV and SFS. A failure may occur during startup, therefore bit **CCF** should be cleared by the Host after the CC entered NORMAL_ACTIVE state.

1 = Clock correction failed

0 = No clock correction error

CCL

CHI Command Locked

The flag signals that the write access to the CHI command vector **SUCC1.CMD[3:0]** was not successful because the execution of the previous CHI command has not yet completed. In this case bit **CNA** is also set to ’1’.

1 = CHI command not accepted

0 = CHI command accepted

PERR

Parity Error

The flag signals a parity error to the Host. It is set whenever one of the flags **MHDS.PIBF**, **MHDS.POBF**, **MHDS.PMR**, **MHDS.PTBF1**, **MHDS.PTBF2** changes from ’0’ to ’1’.

1 = Parity error detected

0 = No parity error detected

RFO

Receive FIFO Overrun

The flag is set by the CC when a receive FIFO overrun is detected. When a receive FIFO overrun occurs, the oldest message is overwritten with the actual received message. The actual state of the FIFO is monitored in register FSR.

1 = A receive FIFO overrun has been detected

0 = No receive FIFO overrun detected

EFA

Empty FIFO Access

This flag is set by the CC when the Host requests the transfer of a message from the receive FIFO via Output Buffer while the receive FIFO is empty.

1 = Host access to empty FIFO occurred

0 = No Host access to empty FIFO occurred

IIBA

Illegal Input Buffer Access

This flag is set by the CC when the Host wants to modify a message buffer via Input Buffer while the CC is not in CONFIG or DEFAULT_CONFIG state and one of the following conditions applies:

1) The Host writes to the Input Buffer Command Request register to modify the

- Header section of message buffer 0, 1 if configured for transmission in key slot
- Header section of static message buffers with buffer number < **MRC.FDB[7:0]** while **MRC.SEC[1:0]** = "01"
- Header section of any static or dynamic message buffer while **MRC.SEC[1:0]** = "1x"
- Header and / or data section of any message buffer belonging to the receive FIFO

2) The Host writes to any register of the Input Buffer while **IBCR.IBSYH** is set to ’1’.

1 = Illegal Host access to Input Buffer occurred

0 = No illegal Host access to Input Buffer occurred

IOBA

Illegal Output buffer Access

This flag is set by the CC when the Host requests the transfer of a message buffer from the Message RAM to the Output Buffer while **OBCR.OBSYS** is set to ’1’.

1 = Illegal Host access to Output Buffer occurred

0 = No illegal Host access to Output Buffer occurred

MHF

Message Handler Constraints Flag

The flag signals a Message Handler constraints violation condition. It is set whenever one of the flags **MHDF.SNUA**, **MHDF.SNUB**, **MHDF.FNFA**, **MHDF.FNFB**, **MHDF.TBFA**, **MHDF.TBFB**, **MHDF.WAHP** changes from ’0’ to ’1’.

1 = Message Handler failure detected

0 = No Message Handler failure detected

Channel-specific error flags:

EDA

Error Detected on Channel A

This bit is set whenever one of the flags **ACS.SEDA**, **ACS.CEDA**, **ACS.CIA**, **ACS.SBVA** changes from ’0’ to ’1’.

1 = Error detected on channel A

0 = No error detected on channel A

LTVA

Latest Transmit Violation Channel A

The flag signals a latest transmit violation on channel A to the Host.

1 = Latest transmit violation detected on channel A

0 = No latest transmit violation detected on channel A

TABA

Transmission Across Boundary Channel A

The flag signals to the Host that a transmission across a slot boundary occurred for channel A.

1 = Transmission across slot boundary detected on channel A

0 = No transmission across slot boundary detected on channel A

EDB

Error Detected on Channel B

This bit is set whenever one of the flags **ACS.SEDB**, **ACS.CEDB**, **ACS.CIB**, **ACS.SBVB** changes from ’0’ to ’1’.

1 = Error detected on channel B

0 = No error detected on channel B

LTVB

Latest Transmit Violation Channel B

The flag signals a latest transmit violation on channel B to the Host.

1 = Latest transmit violation detected on channel B

0 = No latest transmit violation detected on channel B

TABB

Transmission Across Boundary Channel B

The flag signals to the Host that a transmission across a slot boundary occurred for channel B.

1 = Transmission across slot boundary detected on channel B

0 = No transmission across slot boundary detected on channel B

8.4.2 Status Interrupt Register (SIR)

The flags are set when the CC detects one of the listed events. The flags remain set until the Host clears them. A flag is cleared by writing a '1' to the corresponding bit position. Writing a '0' has no effect on the flag. A hard reset will also clear the register.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SIR	R	0	0	0	0	0	0	MTSB	WUPB	0	0	0	0	0	0	MTSA	WUPA
0xD024	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	SDS	MBSI	SUCS	SWE	TOBC	TIBC	TI1	TI0	NMVC	RFCL	RFNE	RXI	TXI	CYCS	CAS	WST
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

WST

Wakeup Status

This flag is set whenever the wakeup status vector **CCSV.WSV[2:0]** has changed.

1 = Wakeup status changed

0 = Wakeup status unchanged

CAS

Collision Avoidance Symbol

This flag is set by the CC during STARTUP state when a CAS or a potential CAS was received.

1 = Bit pattern matching the CAS symbol received

0 = No bit pattern matching the CAS symbol received

CYCS

Cycle Start Interrupt

This flag is set by the CC when a communication cycle starts.

1 = Communication cycle started

0 = No communication cycle started

TXI

Transmit Interrupt

This flag is set by the CC at the end of frame transmission if bit **MBI** in the respective message buffer is set to ’1’ (see Table 27).

1 = At least one frame was transmitted from a transmit buffer with **MBI** = ’1’

0 = No frame transmitted from a transmit buffer with **MBI** = ’1’

Receive Interrupt

This flag is set by the CC whenever the set condition of a message buffers ND flag is fulfilled (see New Data 1/2/3/4 (NDAT1/2/3/4)), and if bit **MBI** of that message buffer is set to ’1’ (see Table 27).

1 = At least one ND flag of a receive buffer with **MBI** = ’1’ has been set to ’1’

0 = No ND flag of a receive buffer with **MBI** = ’1’ has been set to ’1’

RFNE

Receive FIFO Not Empty

This flag is set by the CC when a received valid frame was stored into the empty receive FIFO. The actual state of the receive FIFO is monitored in register FSR.

1 = Receive FIFO is not empty

0 = Receive FIFO is empty

RFCL

Receive FIFO Critical Level

This flag is set when the receive FIFO fill level **FSR.RFFL[7:0]** is equal or greater than the critical level as configured by **FCL.CL[7:0]**.

1 = Receive FIFO critical level reached

0 = Receive FIFO below critical level

NMVC

Network Management Vector Changed

This interrupt flag signals a change in the Network Management Vector visible to the Host.

1 = Network management vector changed

0 = No change in the network management vector

TI0

Timer Interrupt 0

This flag is set whenever timer 0 matches the conditions configured in register T0C. A Timer Interrupt 0 is also signalled on pin **eray_tint0**.

1 = Timer Interrupt 0 occurred

0 = No Timer Interrupt 0

TI1

Timer Interrupt 1

This flag is set whenever timer 1 matches the conditions configured in register T1C. A Timer Interrupt 1 is also signalled on pin **eray_tint1**.

1 = Timer Interrupt 1 occurred

0 = No Timer Interrupt 1

TIBC

Transfer Input Buffer Completed

This flag is set whenever a transfer from Input Buffer to the Message RAM has completed and **IBCR.IBSYS** has been reset by the Message Handler.

1 = Transfer between Input Buffer and Message RAM completed

0 = No transfer completed

TOBC

Transfer Output Buffer Completed

This flag is set whenever a transfer from the Message RAM to the Output Buffer has completed and **OBCR.OBSYS** has been reset by the Message Handler.

1 = Transfer between Message RAM and Output Buffer completed

0 = No transfer completed

SWE

Stop Watch Event

If enabled by the respective control bits located in register STPW1, a rising or falling edge on pin **eray_stpwt**, an interrupt 0,1 event (rising edge on pin **eray_int0** or **eray_int1**) or a software trigger event will generate a stop watch event.

1 = Stop Watch Event occurred

0 = No Stop Watch Event

SUCS

Startup Completed Successfully

This flag is set whenever a startup completed successfully and the CC entered NORMAL_ACTIVE state.

1 = Startup completed successfully

0 = No startup completed successfully

MBSI

Message Buffer Status Interrupt

This flag is set by the CC when the message buffer status **MBS** has changed if bit **MBI** of that message buffer is set (see Table 26).

1 = Message buffer status of at least one message buffer with **MBI** = ’1’ has changed

0 = No message buffer status change of message buffer with **MBI** = ’1’

SDS

Start of Dynamic Segment

This flag is set by the CC when the dynamic segment starts.

1 = Dynamic segment started

0 = Dynamic segment not yet started

Channel-specific status flags:

WUPA

Wakeup Pattern Channel A

This flag is set by the CC when a wakeup pattern was received on channel A. Only set when the CC is in WAKEUP, READY, or STARTUP state, or when in Monitor mode.

1 = Wakeup pattern received on channel A
0 = No wakeup pattern received on channel A

MTSA

MTS Received on Channel A (vSS!ValidMTSA)

Media Access Test symbol received on channel A during the preceding symbol window. Updated by the CC for each channel at the end of the symbol window.

1 = MTS symbol received on channel A
0 = No MTS symbol received on channel A

WUPB

Wakeup Pattern Channel B

This flag is set by the CC when a wakeup pattern was received on channel B. Only set when the CC is in WAKEUP, READY, or STARTUP state, or when in Monitor mode.

1 = Wakeup pattern received on channel B
0 = No wakeup pattern received on channel B

MTSB

MTS Received on Channel B (vSS!ValidMTSB)

Media Access Test symbol received on channel B during the preceding symbol window. Updated by the CC for each channel at the end of the symbol window.

1 = MTS symbol received on channel B
0 = No MTS symbol received on channel B

8.4.3 Error Interrupt Line Select (EILS)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EILS	R	0	0	0	0	0	TABBL	LTVBL	EDBL	0	0	0	0	0	TABAL	LTVAL	EDAL
0xD028	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	0	0	0	MHFL	IOBAL	IIBAL	EFAL	RFOL	PERRL	CCLL	CCFL	SFOL	SFBML	CNAL	PEMCL
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The Error Interrupt Line Select register assigns an interrupt generated by a specific error interrupt flag from register EIR to one of the two module interrupt lines:

1 = Interrupt assigned to interrupt line **eray_int1**

0 = Interrupt assigned to interrupt line **eray_int0**

PEMCL

POC Error Mode Changed Interrupt Line

CNAL

Command Not Accepted Interrupt Line

SFBML

Sync Frames Below Minimum Interrupt Line

SFOL

Sync Frame Overflow Interrupt Line

CCFL

Clock Correction Failure Interrupt Line

CCLL

CHI Command Locked Interrupt Line

PERRL

Parity Error Interrupt Line

RFOL

Receive FIFO Overrun Interrupt Line

EFAL

Empty FIFO Access Interrupt Line

IIBAL

Illegal Input Buffer Access Interrupt Line

IOBAL

Illegal Output Buffer Access Interrupt Line

MHFL

Message Handler Constraints Flag Interrupt Line

EDAL

Error Detected on Channel A Interrupt Line

LTVAL

Latest Transmit Violation Channel A Interrupt Line

TABAL

Transmission Across Boundary Channel A Interrupt Line

EDBL

Error Detected on Channel B Interrupt Line

LTVBL

Latest Transmit Violation Channel B Interrupt Line

TABBL

Transmission Across Boundary Channel B Interrupt Line

8.4.4 Status Interrupt Line Select (SILS)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SILS	R	0	0	0	0	0	0	MTSBL	WUPBL	0	0	0	0	0	0	MTSAL	WUPAL
0xD02C	W																
Reset		0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W	SDSL	MBSIL	SUCSL	SWEL	TOBCL	TIBCL	TI1L	TI0L	NMVCL	RFCLL	RFNEL	RXIL	TXIL	CYCSL	CASL	WSTL
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

The Status Interrupt Line Select register assign an interrupt generated by a specific status interrupt flag from register SIR to one of the two module interrupt lines:

1 = Interrupt assigned to interrupt line **eray_int1**

0 = Interrupt assigned to interrupt line **eray_int0**

WSTL

Wakeup Status Interrupt Line

CASL

Collision Avoidance Symbol Interrupt Line

CYCSL

Cycle Start Interrupt Line

TXIL

Transmit Interrupt Line

RXIL

Receive Interrupt Line

RFNEL

Receive FIFO Not Empty Interrupt Line

RFCLL

Receive FIFO Critical Level Interrupt Line

NMVCL

Network Management Vector Changed Interrupt Line

TI0L

Timer Interrupt 0 Line

TI1L

Timer Interrupt 1 Line

TIBCL

Transfer Input Buffer Completed Interrupt Line

TOBCL

Transfer Output Buffer Completed Interrupt Line

SWEL

Stop Watch Event Interrupt Line

SUCSL

Startup Completed Successfully Interrupt Line

MBSIL

Message Buffer Status Interrupt Line

SDSL

Start of Dynamic Segment Interrupt Line

WUPAL

Wakeup Pattern Channel A Interrupt Line

MTSAL

Media Access Test Symbol Channel A Interrupt Line

WUPBL

Wakeup Pattern Channel B Interrupt Line

MTSBL

Media Access Test Symbol Channel B Interrupt Line

8.4.5 Error Interrupt Enable Set / Reset (EIES, EIER)

The settings in the Error Interrupt Enable register determine which status changes in the Error Interrupt Register will result in an interrupt.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EIES,R	R	0	0	0	0	0	TABBE	LTVBE	EDBE	0	0	0	0	0	TABAE	LTVAE	EDAE
S:0xD030 R:0xD034	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	MHFE	IOBAE	IIBAE	EFAE	RFOE	PERRE	CCLE	CCFE	SFOE	SFBME	CNAE	PEMCE
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The enable bits are set by writing to address 0xD030 and reset by writing to address 0xD034. Writing a ’1’ sets / resets the specific enable bit, writing a ’0’ has no effect. Reading from both addresses will result in the same value.

1 = Interrupt enabled

0 = Interrupt disabled

PEMCE

CNAE

Command Not Accepted Interrupt Enable

SFBME

Sync Frames Below Minimum Interrupt Enable

SFOE
Sync Frame Overflow Interrupt Enable

CCFE
Clock Correction Failure Interrupt Enable

CCLE
CHI Command Locked Interrupt Enable

PERRE
Parity Error Interrupt Enable

RFOE
Receive FIFO Overrun Interrupt Enable

EFAE
Empty FIFO Access Interrupt Enable

IIBAE
Illegal Input Buffer Access Interrupt Enable

IOBAE
Illegal Output Buffer Access Interrupt Enable

MHFE
Message Handler Constraints Flag Interrupt Enable

EDAE
Error Detected on Channel A Interrupt Enable

LTVAE
Latest Transmit Violation Channel A Interrupt Enable

TABAE
Transmission Across Boundary Channel A Interrupt Enable

EDBE
Error Detected on Channel B Interrupt Enable

LTVBE
Latest Transmit Violation Channel B Interrupt Enable

TABBE
Transmission Across Boundary Channel B Interrupt Enable

8.4.6 Status Interrupt Enable Set / Reset (SIES, SIER)

The settings in the Status Interrupt Enable register determine which status changes in the Status Interrupt Register will result in an interrupt.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SIES,R	R	0	0	0	0	0	0			0	0	0	0	0	0		
S:0xD038 R:0xD03C	W							MTSBE	WUPBE							MTSAE	WUPAE

		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit																	
	R	SDSE	MBSIE	SUCSE	SWEE	TOBCE	TIBCE	TI1E	TI0E	NMVCE	RFCLE	RFNEE	RXIE	TXIE	CYCSE	CASE	WSTE
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The enable bits are set by writing to address 0xD038 and reset by writing to address 0xD03C. Writing a ’1’ sets / resets the specific enable bit, writing a ’0’ has no effect. Reading from both addresses will result in the same value.

1 = Interrupt enabled
0 = Interrupt disabled

WSTE
Wakeup Status Interrupt Enable

CASE
Collision Avoidance Symbol Interrupt Enable

CYCSE
Cycle Start Interrupt Enable

TXIE
Transmit Interrupt Enable

RXIE
Receive Interrupt Enable

RFNEE
Receive FIFO Not Empty Interrupt Enable

RFCLE
Receive FIFO Critical Level Interrupt Enable

NMVCE
Network Management Vector Changed Interrupt Enable

TI0E
Timer Interrupt 0 Enable

TI1E
Timer Interrupt 1 Enable

TIBCE
Transfer Input Buffer Completed Interrupt Enable

TOBCE
Transfer Output Buffer Completed Interrupt Enable

SWEE
Stop Watch Event Interrupt Enable

SUCSE

Startup Completed Successfully Interrupt Enable

MBSIE

Message Buffer Status Interrupt Enable

SDSE

Start of Dynamic Segment Interrupt Enable

WUPAE

Wakeup Pattern Channel A Interrupt Enable

MTSAE

MTS Received on Channel A Interrupt Enable

WUPBE

Wakeup Pattern Channel B Interrupt Enable

MTSBE

MTS Received on Channel B Interrupt Enable

8.4.7 Interrupt Line Enable (ILE)

Each of the two interrupt lines to the Host (**eray_int0**, **eray_int1**) can be enabled / disabled separately by programming bit **EINT0** and **EINT1**.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ILE	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD040	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	EINT1	EINT0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EINT0

Enable Interrupt Line 0

1 = Interrupt line **eray_int0** enabled

0 = Interrupt line **eray_int0** disabled

EINT1

Enable Interrupt Line 1

1 = Interrupt line **eray_int1** enabled

0 = Interrupt line **eray_int1** disabled

8.4.8 Timer 0 Configuration (T0C)

Absolute timer. Specifies in terms of cycle count and macrotick the point in time when the timer 0 interrupt occurs. When the timer 0 interrupt is asserted, output signal **eray_tint0** is set to '1' for the duration of one macrotick and **SIR.TI0** is set to '1'.
Timer 0 can be activated as long as the POC is either in NORMAL_ACTIVE state or in NORMAL_PASSIVE state. Timer 0 is deactivated when leaving NORMAL_ACTIVE state or NORMAL_PASSIVE state except for transitions between the two states.
Before reconfiguration of the timer, the timer has to be halted first by writing bit **T0RC** to '0'.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
T0C	R	0	0	T0MO 13	T0MO 12	T0MO 11	T0MO 10	T0MO 9	T0MO 8	T0MO 7	T0MO 6	T0MO 5	T0MO 4	T0MO 3	T0MO 2	T0MO 1	T0MO 0
0xD044	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	T0CC6	T0CC5	T0CC4	T0CC3	T0CC2	T0CC1	T0CC0	0	0	0	0	0	0	T0MS	T0RC
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

T0RC

Timer 0 Run Control

1 = Timer 0 running

0 = Timer 0 halted

T0MS

Timer 0 Mode Select

1 = Continuous mode

0 = Single-shot mode

T0CC[6:0]

Timer 0 Cycle Code

The 7-bit timer 0 cycle code determines the cycle set used for generation of the timer 0 interrupt. For details about the configuration of the cycle code see Section 5.7.2 Cycle Counter Filtering.

T0MO[13:0]

Timer 0 Macrotick Offset

Configures the macrotick offset from the beginning of the cycle where the interrupt is to occur. The Timer 0 Interrupt occurs at this offset for each cycle of the cycle set.

Note: The configuration of timer 0 is compared against the macrotick counter value, there is no separate counter for timer 0. In case the CC leaves NORMAL_ACTIVE or NORMAL_PASSIVE state, or if timer 0 is halted by Host command, output signal

eray_tint0 is reset to ’0’ immediately.

8.4.9 Timer 1 Configuration (T1C)

Relative timer. After the specified number of macroticks has expired, the timer 1 interrupt is asserted, output signal **eray_tint1** is set to ’1’ for the duration of one macrotick and **SIR.TI1** is set to ’1’.
Timer 1 can be activated as long as the POC is either in NORMAL_ACTIVE state or in NORMAL_PASSIVE state. Timer 1 is deactivated when leaving NORMAL_ACTIVE state or NORMAL_PASSIVE state except for transitions between the two states.
Before reconfiguration of the timer, the timer has to be halted first by writing bit **T1RC** to ’0’.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
T1C	R	0	0	T1MC 13	T1MC 12	T1MC 11	T1MC 10	T1MC 9	T1MC 8	T1MC 7	T1MC 6	T1MC 5	T1MC 4	T1MC 3	T1MC 2	T1MC 1	T1MC 0
0xD048	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	T1MS	T1RC
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

T1RC

Timer 1 Run Control

1 = Timer 1 running
0 = Timer 1 halted

T1MS

Timer 1 Mode Select

1 = Continuous mode
0 = Single-shot mode

T1MC[13:0]

Timer 1 Macrotick Count

When the configured macrotick count is reached the timer 1 interrupt is generated. Valid values are: 2 to 16383 MT in continuous mode

1 to 16383 MT in single-shot mode

Note: In case the CC leaves NORMAL_ACTIVE or NORMAL_PASSIVE state, or if timer 1 is halted by Host command, output signal **eray_tint1** is reset to '0' immediately.

8.4.10 Stop Watch Register 1 (STPW1)

The stop watch is activated by a rising or falling edge on pin **eray_stpwt**, by an interrupt 0,1 event (rising edge on pin **eray_int0** or **eray_int1**) or by the Host by writing bit **SSWT** to '1'. With the macrotick counter increment following next to the stop watch activation the actual cycle counter and macrotick values are captured in register STPW1 while the slot counter values for channel A and B are captured in register STPW2.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STPW1	R	0	0	SMTV 13	SMTV 12	SMTV 11	SMTV 10	SMTV9	SMTV8	SMTV7	SMTV6	SMTV5	SMTV4	SMTV3	SMTV2	SMTV1	SMTV0
0xD04C	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	SCCV5	SCCV4	SCCV3	SCCV2	SCCV1	SCCV0	0	EINT1	EINT0	EETP	SSWT	EDGE	SWMS	ESWT
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESWT

Enable Stop Watch Trigger

If enabled an edge on input **eray_stpwt** or an interrupt 0,1 event (rising edge on pin **eray_int0** or **eray_int1**) activates the stop watch. In single-shot mode this bit is reset to '0' after the stop watch event occurred.

1 = Stop watch trigger enabled

0 = Stop watch trigger disabled

SWMS

Stop Watch Mode Select

1 = Continuous mode

0 = Single-shot mode

EDGE

Stop Watch Trigger Edge Select

1 = Rising edge

0 = Falling edge

SSWT

Software Stop Watch Trigger

When the Host writes this bit to ’1’ the stop watch is activated. After the actual cycle counter and macrotick value are stored in the Stop Watch register this bit is reset to ’0’. The bit is only writeable while **ESWT** = ’0’.

1 = Stop watch activated by software trigger

0 = Software trigger reset

EETP

Enable External Trigger Pin

Enables stop watch trigger event via pin **eray_stpwt** if **ESWT** = ’1’.

1 = Edge on pin **eray_stpwt** triggers stop watch

0 = Stop watch trigger via pin **eray_stpwt** disabled

EINT0

Enable Interrupt 0 Trigger

Enables stop watch trigger by interrupt 0 event if **ESWT** = ’1’.

1 = Interrupt 0 event triggers stop watch

0 = Stop watch trigger by interrupt 0 disabled

EINT1

Enable Interrupt 1 Trigger

Enables stop watch trigger by interrupt 1event if **ESWT** = ’1’.

1 = Interrupt 1 event triggers stop watch

0 = Stop watch trigger by interrupt 1 disabled

SCCV[5:0]

Stop Watch Captured Cycle Counter Value

State of the cycle counter when the stop watch event occurred. Valid values are 0 to 63.

SMTV[13:0]

Stop Watch Captured Macrotick Value

State of the macrotick counter when the stop watch event occurred. Valid values are 0 to 16000.

Note: Bits **ESWT** and **SSWT** cannot be set to ’1’ simultaneously. In this case the write access is ignored, and both bits keep their previous values. Either the external stop watch trigger or the software stop watch trigger may be used.

8.4.11 Stop Watch Register 2 (STPW2)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STPW2	R	0	0	0	0	0	SSCV B10	SSCV B9	SSCV B8	SSCV B7	SSCV B6	SSCV B5	SSCV B4	SSCV B3	SSCV B2	SSCV B1	SSCV B0
0xD050	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

R	0	0	0	0	0	SSCV A10	SSCV A9	SSCV A8	SSCV A7	SSCV A6	SSCV A5	SSCV A4	SSCV A3	SSCV A2	SSCV A1	SSCV A0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SSCVA[10:0]

Stop Watch Captured Slot Counter Value Channel A

State of the slot counter for channel A when the stop watch event occurred. Valid values are 0 to 2047.

SSCVB[10:0]

Stop Watch Captured Slot Counter Value Channel B

State of the slot counter for channel B when the stop watch event occurred. Valid values are 0 to 2047.

8.5 CC Control Registers

This section describes the registers provided by the CC to allow the Host to control the operation of the CC. The FlexRay protocol specification requires the Host to write application configuration data in CONFIG state only. Please consider that the configuration registers are not locked for writing in DEFAULT_CONFIG state.
The configuration data is reset when DEFAULT_CONFIG state is entered from hard reset. To change POC state from DEFAULT_CONFIG to CONFIG state the Host has to apply CHI command CONFIG. If the Host wants the CC to leave CONFIG state, the Host has to proceed as described in Section Lock Register (LCK).
All bits marked with an asterisk * can be updated in DEFAULT_CONFIG or CONFIG state only!

8.5.1 SUC Configuration Register 1 (SUCC1)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SUCC1	R	0	0	0	0	CCHB*	CCHA*	MTSB*	MTSA*	HCSE*	TSM*	WUCS*	PTA4*	PTA3*	PTA2*	PTA1*	PTA0*
0xD080	W																
Reset		0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	CSA4*	CSA3*	CSA2*	CSA1*	CSA0*	0	TXSY*	TXST*	PBSY	0	0	0	CMD3	CMD2	CMD1	CMD0
	W																
Reset		0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0

CMD[3:0]

CHI Command Vector

The Host may write any CHI command at any time, but certain commands are enabled only in certain POC states. If a command is not enabled, it will not be executed, the CHI command vector **CMD[3:0]** will be reset to "0000" = command_not_accepted, and flag **EIR.CNA** will be set to '1'. In case the previous CHI command has not yet completed, **EIR.CCL** is set to '1' together with **EIR.CNA**; the CHI command needs to be repeated. Except for HALT state, a POC state change command applied while the CC is already in the requested POC state will be ignored.

0000 = command_not_accepted
0001 = CONFIG
0010 = READY
0011 = WAKEUP
0100 = RUN
0101 = ALL_SLOTS
0110 = HALT
0111 = FREEZE
1000 = SEND_MTS
1001 = ALLOW_COLDSTART
1010 = RESET_STATUS_INDICATORS
1011 = MONITOR_MODE
1100 = CLEAR_RAMS
1101 = reserved
1110 = reserved
1111 = reserved

Reading **CMD[3:0]** shows whether the last CHI command was accepted. The actual POC state is monitored by **CCSV.POCS[5:0]**. The reserved CHI commands belong to the hardware test functions.

command_not_accepted
CMD[3:0] is reset to "0000" due to one of the following conditions:
• Illegal command applied by the Host
• Host applied command to leave CONFIG state without preceding config lock key
• Host applied new command while execution of the previous Host command has not completed
• Host writes **command_not_accepted**
When **CMD[3:0]** is reset to "0000", **EIR.CNA** is set, and - if enabled - an interrupt is generated. Commands which are not accepted are not executed.

CONFIG
Go to POC state CONFIG when called in POC states DEFAULT_CONFIG, READY, or in MONITOR_MODE. When called in HALT state the CC transits to POC state

DEFAULT_CONFIG. When called in any other state, **CMD[3:0]** will be reset to "0000" = command_not_accepted.

READY
Go to POC state READY when called in POC states CONFIG, NORMAL_ACTIVE, NORMAL_PASSIVE, STARTUP, or WAKEUP. When called in any other state, **CMD[3:0]** will be reset to "0000" = command_not_accepted.

WAKEUP
Go to POC state WAKEUP when called in POC state READY. When called in any other state, **CMD[3:0]** will be reset to "0000" = command_not_accepted.

RUN
Go to POC state STARTUP when called in POC state READY. When called in any other state, **CMD[3:0]** will be reset to "0000" = command_not_accepted.

ALL_SLOTS
Leave SINGLE slot mode after successful startup / integration at the next end of cycle when called in POC states NORMAL_ACTIVE or NORMAL_PASSIVE. When called in any other state, **CMD[3:0]** will be reset to "0000" = command_not_accepted.

HALT
Set halt request **CCSV.HRQ** and go to POC state HALT at the next end of cycle when called in POC states NORMAL_ACTIVE or NORMAL_PASSIVE. When called in any other state, **CMD[3:0]** will be reset to "0000" = command_not_accepted.

FREEZE
Set the freeze status indicator **CCSV.FSI** and go to POC state HALT immediately. Can be called from any state.

SEND_MTS
Send single MTS symbol during the next following symbol window on the channel configured by **MTSA**, **MTSB**, when called in POC state NORMAL_ACTIVE after CC entered ALL slot mode (**CCSV.SLM[1:0]** = "11"). When called in any other state, or when called while a previously requested MTS has not yet been transmitted, **CMD[3:0]** will be reset to "0000" = command_not_accepted.

ALLOW_COLDSTART
The command resets **CCSV.CSI** to enable the node to become leading coldstarter. When called in states DEFAULT_CONFIG, CONFIG, HALT, or MONITOR_MODE, **CMD[3:0]** will be reset to "0000" = command_not_accepted. To become leading coldstarter it is also required that both **TXST** and **TXSY** are set.

RESET_STATUS_INDICATORS
Reset status flags **CCSV.FSI**, **CCSV.HRQ**, **CCSV.CSNI**, **CCSV.CSAI**, **CCSV.WSV[2:0]**, and register CCEV. Flags internally evaluated in the actual POC state are not reset. May be called in any state.

MONITOR_MODE

Enter MONITOR_MODE when called in POC state CONFIG. In this mode the CC is able to receive FlexRay frames and wakeup pattern. It is also able to detect coding errors. The temporal integrity of received frames is not checked. This mode can be used for debugging purposes, e.g. in case that the startup of a FlexRay network fails. When called in any other state, **CMD[3:0]** will be reset to "0000" = command_not_accepted. For details see 5.5.4 MONITOR_MODE

CLEAR_RAMS

Sets **MHDS.CRAM** when called in DEFAULT_CONFIG or CONFIG state. When called in any other state, **CMD[3:0]** will be reset to "0000" = command_not_accepted. **MHDS.CRAM** is also set when the CC leaves hard reset. By setting **MHDS.CRAM** all internal RAM blocks are initialized to zero. During the initialization of the RAMs, **PBSY** will show POC busy. Access to the configuration and status registers is possible during execution of CHI command CLEAR_RAMS.

The initialization of the E-Ray internal RAM blocks requires 2048 **eray_bclk** cycles. There should be no Host access to IBF or OBF during initialization of the internal RAM blocks after hard reset or after assertion of CHI command CLEAR_RAMS. Before asserting CHI command CLEAR_RAMS the Host should make sure that no transfer between Message RAM and IBF / OBF or the Transient Buffer RAMs is ongoing. This command also resets the Message Buffer Status registers MHDS, LDTS, FSR, MHDF, TXRQ1/2/3/4, NDAT1/2/3/4, and MBSC1/2/3/4.

Note: All accepted commands with exception of CLEAR_RAMS and SEND_MTS will cause a change of register CCSV after at most 8 cycles of the slower of the two clocks **eray_bclk** and **eray_sclk**, counted from the falling edge of the CHI input signal **eray_select**, assumed that POC was not busy when the command was applied and that no POC state change was forced by bus activity in that time frame. Reading register CCSV will show data that is delayed by synchronization from **eray_sclk** to **eray_bclk** domain and by the Host-specific CPU interface.

PBSY

POC Busy

Signals that the POC is busy and cannot accept a command from the Host. **CMD[3:0]** is locked against write accesses. Set to ’1’ after hard reset during initialization of internal RAM blocks.

1 = POC is busy, **CMD[3:0]** locked

0 = POC not busy, **CMD[3:0]** writeable

TXST

Transmit Startup Frame in Key Slot (pKeySlotUsedForStartup)

Defines whether the key slot is used to transmit startup frames. The bit can be modified in DEFAULT_CONFIG or CONFIG state only.

1 = Key slot used to transmit startup frame, node is leading or following coldstarter

0 = No startup frame transmission in key slot, node is non-coldstarter

TXSY

Transmit Sync Frame in Key Slot (pKeySlotUsedForSync)

Defines whether the key slot is used to transmit sync frames. The bit can be modified in DEFAULT_CONFIG or CONFIG state only.

1 = Key slot used to transmit sync frame, node is sync node

0 = No sync frame transmission in key slot, node is neither sync nor coldstart node

Note: The protocol requires that both bits **TXST** and **TXSY** are set for coldstart nodes.

CSA[4:0]

Cold Start Attempts (gColdStartAttempts)

Configures the maximum number of attempts that a cold starting node is permitted to try to start up the network without receiving any valid response from another node. It can be modified in DEFAULT_CONFIG or CONFIG state only. Must be identical in all nodes of a cluster. Valid values are 2 to 31.

PTA[4:0]

Passive to Active (pAllowPassiveToActive)

Defines the number of consecutive even / odd cycle pairs that must have valid clock correction terms before the CC is allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state. If set to "00000" the CC is **not** allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state. It can be modified in DEFAULT_CONFIG or CONFIG state only. Valid values are 0 to 31 even / odd cycle pairs.

WUCS

Wakeup Channel Select (pWakeupChannel)

With this bit the Host selects the channel on which the CC sends the Wakeup pattern. The CC ignores any attempt to change the status of this bit when not in DEFAULT_CONFIG or CONFIG state.

1 = Send wakeup pattern on channel B

0 = Send wakeup pattern on channel A

TSM

Transmission Slot Mode (pSingleSlotEnabled)

Selects the initial transmission slot mode. In SINGLE slot mode the CC may only transmit in the preconfigured key slot. The key slot ID is configured in the header section of message buffer 0 respectively message buffers 0 and 1 depending on bit **MRC.SPLM**. In case **TSM** = '1', message buffer 0 respectively message buffers 0,1 can be (re)configured in DEFAULT_CONFIG or CONFIG state only. In ALL slot mode the CC may transmit in all slots. **TSM** is a configuration bit which can only be set / reset by the Host. The bit can be written in DEFAULT_CONFIG or CONFIG state only. The CC changes to ALL slot mode when the Host successfully applied the ALL_SLOTS command by writing **CMD[3:0]** = "0101" in POC states NORMAL_ACTIVE or NORMAL_PASSIVE. The actual slot mode is monitored by **CCSV.SLM[1:0]**.

1 = SINGLE Slot Mode (default after hard reset)

0 = ALL Slot Mode

HCSE

Halt due to Clock Sync Error (pAllowHaltDueToClock)

Controls the transition to HALT state due to a clock synchronization error. The bit can be modified in DEFAULT_CONFIG or CONFIG state only.

1 = CC will enter HALT state

0 = CC will enter / remain in NORMAL_PASSIVE

MTSA

Select Channel A for MTS Transmission

The bit selects channel A for MTS symbol transmission. The flag is reset by default and may be modified only in DEFAULT_CONFIG or CONFIG state.

1 = Channel A selected for MTS transmission

0 = Channel A disabled for MTS transmission

MTSB

Select Channel B for MTS Transmission

The bit selects channel B for MTS symbol transmission. The flag is reset by default and may be modified only in DEFAULT_CONFIG or CONFIG state.

1 = Channel B selected for MTS transmission

0 = Channel B disabled for MTS transmission

Note: If both bits **MTSA** and **MTSB** are set to '1' an MTS symbol will be transmitted on both channels when requested by writing **CMD[3:0]** = "1000".

CCHA

Connected to Channel A (pChannels)

Configures whether the node is connected to channel A.

1 = Node connected to channel A (default after hard reset)

0 = Not connected to channel A

CCHB

Connected to Channel B (pChannels)

Configures whether the node is connected to channel B.

1 = Node connected to channel B (default after hard reset)

0 = Not connected to channel B

Reference to CHI Host command summary from FlexRay protocol specificatio below references the CHI commands from the FlexRay Protocol Specification v2.1 (section 2.2.1.1, Table 2-2) to the E-Ray CHI command vector CMD[3:0].

CHI command	Where processed (POC States)	CHI Command Vector CMD[3:0]
ALL_SLOTS	POC:normal active, POC:normal passive	ALL_SLOTS

ALLOW_COLDSTART	All except POC:default config, POC:config, POC:halt	ALLOW_COLDSTART
CONFIG	POC:default config, POC:ready	CONFIG
CONFIG_COMPLETE	POC:config	Unlock sequence & READY
DEFAULT_CONFIG	POC:halt	CONFIG
FREEZE	All	FREEZE
HALT	POC:normal active, POC:normal passive	HALT
READY	All except POC:default config, POC:config, POC:ready, POC:halt	READY
RUN	POC:ready	RUN
WAKEUP	POC:ready	WAKEUP

Table 13: Reference to CHI Host command summary from FlexRay protocol specification

8.5.2 SUC Configuration Register 2 (SUCC2)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SUCC2	R	0	0	0	0	LTN3*	LTN2*	LTN1*	LTN0*	0	0	0	LT20*	LT19*	LT18*	LT17*	LT16*
0xD084	W																
Reset		0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	LT15*	LT14*	LT13*	LT12*	LT11*	LT10*	LT9*	LT8*	LT7*	LT6*	LT5*	LT4*	LT3*	LT2*	LT1*	LT0*
	W																
Reset		0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0

LT[20:0]

Listen Timeout (pdListenTimeout)

Configures wakeup / startup listen timeout in µT. The range for pdListenTimeout is 1284 to 1283846 µT.

LTN[3:0]

Listen Timeout Noise (gListenNoise - 1)

Configures the upper limit for startup and wakeup listen timeout in the presence of noise expressed as a multiple of pdListenTimeout. The range for gListenNoise is 2 to 16. **LTN[3:0]** must be configured identical in all nodes of a cluster.

Note: The wakeup / startup noise timeout is calculated as follows: pdListenTimeout • gListenNoise **= LT[20:0]** • (**LTN[3:0]** + 1)

8.5.3 SUC Configuration Register 3 (SUCC3)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SUCC3	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD088	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	0	0	WCF3 *	WCF2 *	WCF1 *	WCF0 *	WCP3 *	WCP2 *	WCP1 *	WCP0 *
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1

WCP[3:0]

Maximum Without Clock Correction Passive (gMaxWithoutClockCorrectionPassive)

Defines the number of consecutive even / odd cycle pairs with missing clock correction terms that will cause a transition from NORMAL_ACTIVE to NORMAL_PASSIVE state. Must be identical in all nodes of a cluster. Valid values are 1 to 15 cycle pairs.

WCF[3:0]

Maximum Without Clock Correction Fatal (gMaxWithoutClockCorrectionFatal)

Defines the number of consecutive even / odd cycle pairs with missing clock correction terms that will cause a transition from NORMAL_ACTIVE or NORMAL_PASSIVE to HALT state. Must be identical in all nodes of a cluster. Valid values are 1 to 15 cycle pairs.

8.5.4 NEM Configuration Register (NEMC)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NEMC	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD08C	W																

		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit																	
	R	0	0	0	0	0	0	0	0	0	0	0	0	NML3*	NML2*	NML1*	NML0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

NML[3:0]

Network Management Vector Length (gNetworkManagementVectorLength)

These bits configure the length of the NM vector. The configured length must be identical in all nodes of a cluster. Valid values are 0 to 12 bytes.

8.5.5 PRT Configuration Register 1 (PRTC1)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit																	
PRTC1	R	RWP5*	RWP4*	RWP3*	RWP2*	RWP1*	RWP0*	0	RXW8*	RXW7*	RXW6*	RXW5*	RXW4*	RXW3*	RXW2*	RXW1*	RXW0*
0xD090	W																
Reset		0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	0

		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit																	
	R	BRP1*	BRP0*	SPP1*	SPP0*	0	CASM6	CASM5*	CASM4*	CASM3*	CASM2*	CASM1*	CASM0*	TSST3*	TSST2*	TSST1*	TSST0*
	W																
Reset		0	0	0	0	0	1	1	0	0	0	1	1	0	0	1	1

TSST[3:0]

Transmission Start Sequence Transmitter (gdTSSTransmitter)

Configures the duration of the Transmission Start Sequence (TSS) in terms of bit times (1 bit time = 4 µT = 100ns @ 10Mbps). Must be identical in all nodes of a cluster. Valid values are 3 to 15 bit times.

CASM[6:0]

Collision Avoidance Symbol Max (gdCASRxLowMax)

Configures the upper limit of the acceptance window for a collision avoidance symbol (CAS). **CASM6** is fixed to '1'. Valid values are 67 to 99 bit times.

SPP[1:0]

Strobe Point Position

Defines the sample count value for strobing. The strobed bit value is set to the voted value when the sample count is incremented to the value configured by **SPP[1:0]**.

00, 11= Sample 5 (default)

01 = Sample 4

10 = Sample 6

Note: The current revision 2.1 of the FlexRay protocol requires that **SPP[1:0]** = "00". The alternate strobe point positions could be used to compensate for asymmetries in the physical layer.

BRP[1:0]

Baud Rate Prescaler (gdSampleClockPeriod, pSamplesPerMicrotick)

The Baud Rate Prescaler configures the baud rate on the FlexRay bus. The baud rates listed below are valid with a sample clock **eray_sclk** = 80 MHz. One bit time always consists of 8 samples independent of the configured baud rate.

00 = 10 MBit/s (default)
gdSampleClockPeriod = 12.5 ns = 1 • **eray_sclk**
pSamplesPerMicrotick = 2 (1 µT = 25 ns)

01 = 5 MBit/s
gdSampleClockPeriod = 25 ns = 2 • **eray_sclk**
pSamplesPerMicrotick = 1 (1 µT = 25 ns)

10, 11 = 2.5 MBit/s
gdSampleClockPeriod = 50 ns = 4 • **eray_sclk**
pSamplesPerMicrotick = 1 (1 µT = 50 ns)

RXW[8:0]

Wakeup Symbol Receive Window Length (gdWakeupSymbolRxWindow)

Configures the number of bit times used by the node to test the duration of the received wakeup pattern. Must be identical in all nodes of a cluster. Valid values are 76 to 301 bit times.

RWP[5:0]

Repetitions of Tx Wakeup Pattern (pWakeupPattern)

Configures the number of repetitions (sequences) of the Tx wakeup symbol. Valid values are 2 to 63.

8.5.6 PRT Configuration Register 2 (PRTC2)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRTC2	R	0	0	TXL5*	TXL4*	TXL3*	TXL2*	TXL1*	TXL0*	TXI7*	TXI6*	TXI5*	TXI4*	TXI3*	TXI2*	TXI1*	TXI0*
0xD094	W																
Reset		0	0	0	0	1	1	1	1	0	0	1	0	1	1	0	1

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	RXL5*	RXL4*	RXL3*	RXL2*	RXL1*	RXL0*	0	0	RXI5*	RXI4*	RXI3*	RXI2*	RXI1*	RXI0*
	W																

Reset	0	0	0	0	1	0	1	0	0	0	0	0	1	1	1	0

RXI[5:0]

Wakeup Symbol Receive Idle (gdWakeupSymbolRxIdle)

Configures the number of bit times used by the node to test the duration of the idle phase of the received wakeup symbol. Must be identical in all nodes of a cluster. Valid values are 14 to 59 bit times.

RXL[5:0]

Wakeup Symbol Receive Low (gdWakeupSymbolRxLow)

Configures the number of bit times used by the node to test the duration of the low phase of the received wakeup symbol. Must be identical in all nodes of a cluster. Valid values are 10 to 55 bit times.

TXI[7:0]

Wakeup Symbol Transmit Idle (gdWakeupSymbolTxIdle)

Configures the number of bit times used by the node to transmit the idle phase of the wakeup
symbol. Must be identical in all nodes of a cluster. Valid values are 45 to 180 bit times.

TXL[5:0]

Wakeup Symbol Transmit Low (gdWakeupSymbolTxLow)

Configures the number of bit times used by the node to transmit the low phase of the wakeup symbol. Must be identical in all nodes of a cluster. Valid values are 15 to 60 bit times.

8.5.7 MHD Configuration Register (MHDC)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MHDC	R	0	0	0	SLT12*	SLT11*	SLT10*	SLT9*	SLT8*	SLT7*	SLT6*	SLT5*	SLT4*	SLT3*	SLT2*	SLT1*	SLT0*
0xD098	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	0	0	0	SFDL6*	SFDL5*	SFDL4*	SFDL3*	SFDL2*	SFDL1*	SFDL0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SFDL[6:0]

Static Frame Data Length (gPayloadLengthStatic)

Configures the cluster-wide payload length for all frames sent in the static segment in double bytes. The payload length must be identical in all nodes of a cluster. Valid values are 0 to 127.

SLT[12:0]

Start of Latest Transmit (pLatestTx)

Configures the maximum minislot value allowed before inhibiting frame transmission in the dynamic segment of the cycle. There is no transmission in dynamic segment if **SLT[12:0]** is set to zero. Valid values are 0 to 7981 minislots.

8.5.8 GTU Configuration Register 1 (GTUC1)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC1	R	0	0	0	0	0	0	0	0	0	0	0	0	UT19*	UT18*	UT17*	UT16*
0xD0A0	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	UT15*	UT14*	UT13*	UT12*	UT11*	UT10*	UT9*	UT8*	UT7*	UT6*	UT5*	UT4*	UT3*	UT2*	UT1*	UT0*
	W																
Reset		0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0

UT[19:0]

Microtick per Cycle (pMicroPerCycle)

Configures the duration of the communication cycle in microticks. Valid values are 640 to 640000 µT.

8.5.9 GTU Configuration Register 2 (GTUC2)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC2	R	0	0	0	0	0	0	0	0	0	0	0	0	SNM3*	SNM2*	SNM1*	SNM0*
0xD0A4	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	MPC13*	MPC12*	MPC11*	MPC10*	MPC9*	MPC8*	MPC7*	MPC6*	MPC5*	MPC4*	MPC3*	MPC2*	MPC1*	MPC0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

MPC[13:0]

Macrotick Per Cycle (gMacroPerCycle)

Configures the duration of one communication cycle in macroticks. The cycle length must be identical in all nodes of a cluster. Valid values are 10 to 16000 MT.

SNM[3:0]

Sync Node Max (gSyncNodeMax)

Maximum number of frames within a cluster with sync frame indicator bit **SYN** set to '1'. Must be identical in all nodes of a cluster. Valid values are 2 to 15.

8.5.10 GTU Configuration Register 3 (GTUC3)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC3	R	0	MIOB6*	MIOB5*	MIOB4*	MIOB3*	MIOB2*	MIOB1*	MIOB0*	0	MIOA6*	MIOA5*	MIOA4*	MIOA3*	MIOA2*	MIOA1*	MIOA0*
0xD0A8	W																
Reset		0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	UIOB7*	UIOB6*	UIOB5*	UIOB4*	UIOB3*	UIOB2*	UIOB1*	UIOB0*	UIOA7*	UIOA6*	UIOA5*	UIOA4*	UIOA3*	UIOA2*	UIOA1*	UIOA0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

UIOA[7:0]

Microtick Initial Offset Channel A (pMicroInitialOffset[A])

Configures the number of microticks between the actual time reference point on channel A and the subsequent macrotick boundary of the secondary time reference point. The parameter depends on pDelayCompensation[A] and therefore has to be set for each channel independently. Valid values are 0 to 240 µT.

UIOB[7:0]

Microtick Initial Offset Channel B (pMicroInitialOffset[B])

Configures the number of microticks between the actual time reference point on channel B and the subsequent macrotick boundary of the secondary time reference point. The parameter depends on pDelayCompensation[B] and therefore has to be set for each channel independently. Valid values are 0 to 240 µT.

MIOA[6:0]

Macrotick Initial Offset Channel A (pMacroInitialOffset[A])

Configures the number of macroticks between the static slot boundary and the subsequent macrotick boundary of the secondary time reference point based on the nominal macrotick duration. Must be identical in all nodes of a cluster. Valid values are 2 to 72 MT.

MIOB[6:0]

Macrotick Initial Offset Channel B (pMacroInitialOffset[B])

Configures the number of macroticks between the static slot boundary and the subsequent macrotick boundary of the secondary time reference point based on the nominal macrotick duration. Must be identical in all nodes of a cluster. Valid values are 2 to 72 MT.

8.5.11 GTU Configuration Register 4 (GTUC4)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only. For details about configuration of **NIT[13:0]** and **OCS[13:0]** see Section 5.1.5 Configuration of NIT Start and Offset Correction Start.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC4	R	0	0	OCS13*	OCS12*	OCS11*	OCS10*	OCS9*	OCS8*	OCS7*	OCS6*	OCS5*	OCS4*	OCS3*	OCS2*	OCS1*	OCS0*
0xD0AC	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	NIT13*	NIT12*	NIT11*	NIT10*	NIT9*	NIT8*	NIT7*	NIT6*	NIT5*	NIT4*	NIT3*	NIT2*	NIT1*	NIT0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

NIT[13:0]

Network Idle Time Start (gMacroPerCycle - gdNIT - 1)

Configures the starting point of the Network Idle Time NIT at the end of the communication cycle expressed in terms of macroticks from the beginning of the cycle. The start of NIT is recognized if Macrotick = gMacroPerCycle - gdNIT -1 and the

increment pulse of Macrotick is set. Must be identical in all nodes of a cluster. Valid values are 7 to 15997 MT.

OCS[13:0]

Offset Correction Start (gOffsetCorrectionStart - 1)

Determines the start of the offset correction within the NIT phase, calculated from start of cycle. Must be identical in all nodes of a cluster. For cluster consisting of E-Ray implementations only, it is sufficient to program **OCS** = **NIT** + 1. Valid values are 8 to 15998 MT.

8.5.12 GTU Configuration Register 5 (GTUC5)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC5	R	DEC7*	DEC6*	DEC5*	DEC4*	DEC3*	DEC2*	DEC1*	DEC0*	0	0	0	CDD4*	CDD3*	CDD2*	CDD1*	CDD0*
0xD0B0	W																
Reset		0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	DCB7*	DCB6*	DCB5*	DCB4*	DCB3*	DCB2*	DCB1*	DCB0*	DCA7*	DCA6*	DCA5*	DCA4*	DCA3*	DCA2*	DCA1*	DCA0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DCA[7:0]

Delay Compensation Channel A (pDelayCompensation[A])

Used to compensate for reception delays on the indicated channel. This covers assumed propagation delay up to cPropagationDelayMax for microticks in the range of 0.0125 to

0.05µs. In practice, the minimum of the propagation delays of all sync nodes should be applied.

Valid values are 0 to 200 µT.

DCB[7:0]

Delay Compensation Channel B (pDelayCompensation[B])

Used to compensate for reception delays on the indicated channel. This covers assumed propagation delay up to cPropagationDelayMax for microticks in the range of 0.0125 to 0.05µs. In practice, the minimum of the propagation delays of all sync nodes should be applied.

Valid values are 0 to 200 µT.

CDD[4:0]

Cluster Drift Damping (pClusterDriftDamping)

Configures the cluster drift damping value used in clock synchronization to minimize accumulation of rounding errors. Valid values are 0 to 20 µT.

DEC[7:0]

Decoding Correction (pDecodingCorrection)

Configures the decoding correction value used to determine the primary time reference point. Valid values are 14 to 143 µT.

8.5.13 GTU Configuration Register 6 (GTUC6)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC6	R	0	0	0	0	0	MOD 10*	MOD9*	MOD8*	MOD7*	MOD6*	MOD5*	MOD4*	MOD3*	MOD2*	MOD1*	MOD0*
0xD0B4	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	ASR10*	ASR9*	ASR8*	ASR7*	ASR6*	ASR5*	ASR4*	ASR3*	ASR2*	ASR1*	ASR0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASR[10:0]

Accepted Startup Range (pdAcceptedStartupRange)

Number of microticks constituting the expanded range of measured deviation for startup frames during integration. Valid values are 0 to 1875 µT.

MOD[10:0]

Maximum Oscillator Drift (pdMaxDrift)

Maximum drift offset between two nodes that operate with unsynchronized clocks over one communication cycle in µT. Valid values are 2 to 1923 µT.

8.5.14 GTU Configuration Register 7 (GTUC7)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC7	R	0	0	0	0	0	0	NSS9*	NSS8*	NSS7*	NSS6*	NSS5*	NSS4*	NSS3*	NSS2*	NSS1*	NSS0*
0xD0B8	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	SSL9*	SSL8*	SSL7*	SSL6*	SSL5*	SSL4*	SSL3*	SSL2*	SSL1*	SSL0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

SSL[9:0]

Static Slot Length (gdStaticSlot)

Configures the duration of a static slot in macroticks. The static slot length must be identical in all nodes of a cluster. Valid values are 4 to 659 MT.

NSS[9:0]

Number of Static Slots (gNumberOfStaticSlots)

Configures the number of static slots in a cycle. At least 2 coldstart nodes must be configured to startup a FlexRay network. The number of static slots must be identical in all nodes of a cluster. Valid values are 2 to 1023.

8.5.15 GTU Configuration Register 8 (GTUC8)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC8	R	0	0	0	NMS 12*	NMS 11*	NMS 10*	NMS9 *	NMS8 *	NMS7 *	NMS6 *	NMS5 *	NMS4 *	NMS3 *	NMS2 *	NMS1 *	NMS0 *
0xD0BC	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	0	0	0	0	MSL5*	MSL4*	MSL3*	MSL2*	MSL1*	MSL0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

MSL[5:0]

Minislot Length (gdMinislot)

Configures the duration of a minislot in macroticks. The minislot length must be identical in all nodes of a cluster. Valid values are 2 to 63 MT.

NMS[12:0]

Number of Minislots (gNumberOfMinislots)

Configures the number of minislots within the dynamic segment of a cycle. The number of minislots must be identical in all nodes of a cluster. Valid values are 0 to 7986.

8.5.16 GTU Configuration Register 9 (GTUC9)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC9	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DSI1*	DSI0*
0xD0C0	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	MAPO 4*	MAPO 3*	MAPO 2*	MAPO 1*	MAPO 0*	0	0	APO5*	APO4*	APO3*	APO2*	APO1*	APO0*
	W																
Reset		0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

APO[5:0]

Action Point Offset (gdActionPointOffset)

Configures the action point offset in macroticks within static slots and symbol window. Must be identical in all nodes of a cluster. Valid values are 1 to 63 MT.

MAPO[4:0]

Minislot Action Point Offset (gdMinislotActionPointOffset)

Configures the action point offset in macroticks within the minislots of the dynamic segment. Must be identical in all nodes of a cluster. Valid values are 1 to 31 MT.

DSI[1:0]

Dynamic Slot Idle Phase (gdDynamicSlotIdlePhase)

The duration of the dynamic slot idle phase has to be greater or equal than the idle detection time. Must be identical in all nodes of a cluster. Valid values are 0 to 2 Minislot.

8.5.17 GTU Configuration Register 10 (GTUC10)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC10	R	0	0	0	0	0	MRC 10*	MRC9*	MRC8*	MRC7*	MRC6*	MRC5*	MRC4*	MRC3*	MRC2*	MRC1*	MRC0*
0xD0C4	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	MOC 13*	MOC 12*	MOC 11*	MOC 10*	MOC9*	MOC8*	MOC7*	MOC6*	MOC5*	MOC4*	MOC3*	MOC2*	MOC1*	MOC0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

MOC[13:0]

Maximum Offset Correction (pOffsetCorrectionOut)

Holds the maximum permitted offset correction value to be applied by the internal clock synchronization algorithm (absolute value). The CC checks only the internal offset correction value against the maximum offset correction value. Valid values are 5 to 15266 µT.

MRC[10:0]

Maximum Rate Correction (pRateCorrectionOut)

Holds the maximum permitted rate correction value to be applied by the internal clock synchronization algorithm. The CC checks only the internal rate correction value against the maximum rate correction value (absolute value). Valid values are 2 to 1923 µT.

8.5.18 GTU Configuration Register 11 (GTUC11)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GTUC11	R	0	0	0	0	0	ERC2*	ERC1*	ERC0*	0	0	0	0	0	EOC2*	EOC1*	EOC0*
0xD0C8	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	ERCC1	ERCC0	0	0	0	0	0	0	EOCC 1	EOCC 0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EOCC[1:0]

External Offset Correction Control (vExternOffsetControl)

By writing to **EOCC[1:0]** the external offset correction is enabled as specified below. Should be modified only outside NIT.

00, 01 = No external offset correction
10 = External offset correction value subtracted from calculated offset correction value
11 = External offset correction value added to calculated offset correction value

ERCC[1:0]

External Rate Correction Control (vExternRateControl)

By writing to **ERCC[1:0]** the external rate correction is enabled as specified below. Should be modified only outside NIT.

00, 01 = No external rate correction
10 = External rate correction value subtracted from calculated rate correction value
11 = External rate correction value added to calculated rate correction value

EOC[2:0]

External Offset Correction (pExternOffsetCorrection)

Holds the external offset correction value in microticks to be applied by the internal clock synchronization algorithm. The value is subtracted / added from / to the calculated offset correction value. The value is applied during NIT. May be modified in DEFAULT_CONFIG or CONFIG state only. Valid values are 0 to 7 µT.

ERC[2:0]

External Rate Correction (pExternRateCorrection)

Holds the external rate correction value in microticks to be applied by the internal clock synchronization algorithm. The value is subtracted / added from / to the calculated rate correction value. The value is applied during NIT. May be modified in DEFAULT_CONFIG or CONFIG state only. Valid values are 0 to 7 µT.

8.6 CC Status Registers

During 8/16-bit accesses to status variables coded with more than 8/16-bit, the variable might be updated by the CC between two accesses (non-atomic read accesses). All internal counters and the CC status flags are reset when the CC transits from CONFIG to READY state.

8.6.1 CC Status Vector (CCSV)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCSV	R	0	0	PSL5	PSL4	PSL3	PSL2	PSL1	PSL0	RCA4	RCA3	RCA2	RCA1	RCA0	WSV2	WSV1	WSV0
0xD100	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	CSI	CSAI	CSNI	0	0	SLM1	SLM0	HRQ	FSI	POCS5	POCS4	POCS3	POCS2	POCS1	POCS0
	W																

Reset	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

POCS[5:0]

Protocol Operation Control Status

Indicates the actual state of operation of the CC Protocol Operation Control

00 0000 = DEFAULT_CONFIG state
00 0001 = READY state
00 0010 = NORMAL_ACTIVE state
00 0011 = NORMAL_PASSIVE state
00 0100 = HALT state
00 0101 = MONITOR_MODE state
00 0110...00 1110 = reserved
00 1111 = CONFIG state

Indicates the actual state of operation of the POC in the wakeup path

01 0000 = WAKEUP_STANDBY state
01 0001 = WAKEUP_LISTEN state
01 0010 = WAKEUP_SEND state
01 0011 = WAKEUP_DETECT state
01 0100...01 1111 = reserved

Indicates the actual state of operation of the POC in the startup path

10 0000 = STARTUP_PREPARE state
10 0001 = COLDSTART_LISTEN state
10 0010 = COLDSTART_COLLISION_RESOLUTION state
10 0011 = COLDSTART_CONSISTENCY_CHECK state
10 0100 = COLDSTART_GAP state
10 0101 = COLDSTART_JOIN State
10 0110 = INTEGRATION_COLDSTART_CHECK state
10 0111 = INTEGRATION_LISTEN state
10 1000 = INTEGRATION_CONSISTENCY_CHECK state
10 1001 = INITIALIZE_SCHEDULE state
10 1010 = ABORT_STARTUP state
10 1011...11 1111 = reserved

FSI

Freeze Status Indicator (vPOC!Freeze)

Indicates that the POC has entered the HALT state due to CHI command FREEZE or due to an error condition requiring an immediate POC halt. Reset by CHI command RESET_STATUS_INDICATORS or by transition from HALT to DEFAULT_CONFIG state.

HRQ

Halt Request (vPOC!CHIHaltRequest)

Indicates that a request from the Host has been received to halt the POC at the end of the communication cycle. Reset by CHI command RESET_STATUS_INDICATORS or by transition from HALT to DEFAULT_CONFIG state or when entering READY state.

SLM[1:0]

Slot Mode (vPOC!SlotMode)

Indicates the actual slot mode of the POC. Default is SINGLE. Changes to ALL, depending on **SUCC1.TSM**. In NORMAL_ACTIVE or NORMAL_PASSIVE state the CHI command ALL_SLOTS will change the slot mode from SINGLE over ALL_PENDING to ALL. When not in NORMAL_ACTIVE or NORMAL_PASSIVE state then reset by CHI command RESET_STATUS_INDICATORS to the value defined by **SUCC1.TSM**.

00 = SINGLE
01 = reserved
10 = ALL_PENDING
11 = ALL

CSNI

Coldstart Noise Indicator (vPOC!ColdstartNoise)

Indicates that the cold start procedure occurred under noisy conditions. Reset by CHI command RESET_STATUS_INDICATORS or by transition from HALT to DEFAULT_CONFIG state or from READY to STARTUP state.

CSAI

Coldstart Abort Indicator

Coldstart aborted. Reset by CHI command RESET_STATUS_INDICATORS or by transition from HALT to DEFAULT_CONFIG state or from READY to STARTUP state.

CSI

Cold Start Inhibit (vColdStartInhibit)

Indicates that the node is disabled from cold starting. The flag is set whenever the POC enters READY state. The flag has to be reset under control of the Host by CHI command ALLOW_COLDSTART (**SUCC1.CMD[3:0]** = "1001").

1 = Cold starting of node disabled
0 = Cold starting of node enabled

WSV[2:0]

Wakeup Status (vPOC!WakeupStatus)

Indicates the status of the current wakeup attempt. Reset by CHI command RESET_STATUS_INDICATORS or by transition from HALT to DEFAULT_CONFIG state or from READY to STARTUP state.

000 = UNDEFINED. No wakeup attempt since CONFIG state was left.

001 = RECEIVED_HEADER. Set when the CC finishes wakeup due to the reception of a frame header without coding violation on either channel in WAKEUP_LISTEN state.

010 = RECEIVED_WUP. Set when the CC finishes wakeup due to the reception of a valid wakeup pattern on the configured wakeup channel in WAKEUP_LISTEN state.

011 = COLLISION_HEADER. Set when the CC stops wakeup due to a detected collision during wakeup pattern transmission by receiving a valid header on either channel.

100 = COLLISION_WUP. Set when the CC stops wakeup due to a detected collision during wakeup pattern transmission by receiving a valid wakeup pattern on the configured wakeup channel.

101 = COLLISION_UNKNOWN. Set when the CC stops wakeup by leaving WAKEUP_DETECT state after expiration of the wakeup timer without receiving a valid wakeup pattern or a valid frame header.

110 = TRANSMITTED. Set when the CC has successfully completed the transmission of the wakeup pattern.

111 = reserved

RCA[4:0]

Remaining Coldstart Attempts (vRemainingColdstartAttempts)

Indicates the number of remaining coldstart attempts. The READY command resets this counter to the maximum number of coldstart attempts as configured by **SUCC1.CSA[4:0]**.

PSL[5:0]

POC Status Log

Status of POCS[5:0] immediately before entering HALT state. Set when entering HALT state. Set to HALT when FREEZE command is applied during HALT state. Reset to "00 0000" when leaving HALT state.

Note: CHI command RESET_STATUS_INDICATORS (**SUCC1.CMD[3:0]** = "1010") resets flags **FSI**, **HRQ**, **CSNI**, **CSAI**, the slot mode **SLM[1:0]**, and the wakeup status **WSV[2:0]**.

8.6.2 CC Error Vector (CCEV)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCEV	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD104	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	PTAC4	PTAC3	PTAC2	PTAC1	PTAC0	ERRM1	ERRM0	0	0	CCFC3	CCFC2	CCFC1	CCFC0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Reset by CHI command RESET_STATUS_INDICATORS or by transition from HALT to DEFAULT_CONFIG state or when entering READY state.

CCFC[3:0]

Clock Correction Failed Counter (vClockCorrectionFailed)

The Clock Correction Failed Counter is incremented by one at the end of any odd communication cycle where either the missing offset correction error or missing rate correction error are active. The Clock Correction Failed Counter is reset to ’0’ at the end of an odd communication cycle if neither the offset correction failed nor the rate correction failed errors are active. The Clock Correction Failed Counter stops at 15.

ERRM[1:0]

Error Mode (vPOC!ErrorMode)

Indicates the actual error mode of the POC.

00 = ACTIVE (green)

01 = PASSIVE (yellow)

10 = COMM_HALT (red)

11 = reserved

PTAC[4:0]

Passive to Active Count (vAllowPassiveToActive)

Indicates the number of consecutive even / odd cycle pairs that have passed with valid rate and offset correction terms, while the node is waiting to transit from NORMAL_PASSIVE state to NORMAL_ACTIVE state. The transition takes place when **PTAC[4:0]** equals **SUCC1.PTA[4:0] -1**.

8.6.3 Slot Counter Value (SCV)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SCV	R	0	0	0	0	0	SCCB10	SCCB9	SCCB8	SCCB7	SCCB6	SCCB5	SCCB4	SCCB3	SCCB2	SCCB1	SCCB0
0xD110	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	SCCA10	SCCA9	SCCA8	SCCA7	SCCA6	SCCA5	SCCA4	SCCA3	SCCA2	SCCA1	SCCA0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SCCA[10:0]

Slot Counter Channel A (vSlotCounter[A])

Current slot counter value on channel A. The value is incremented by the CC and reset at the start of a communication cycle. Valid values are 0 to 2047.

SCCB[10:0]

Slot Counter Channel B (vSlotCounter[B])

Current slot counter value on channel B. The value is incremented by the CC and reset at the start of a communication cycle. Valid values are 0 to 2047.

8.6.4 Macrotick and Cycle Counter Value (MTCCV)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MTCCV	R	0	0	0	0	0	0	0	0	0	0	CCV5	CCV4	CCV3	CCV2	CCV1	CCV0
0xD114	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	MTV13	MTV12	MTV11	MTV10	MTV9	MTV8	MTV7	MTV6	MTV5	MTV4	MTV3	MTV2	MTV1	MTV0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MTV[13:0]

Macrotick Value (vMacrotick)

Current macrotick value. The value is incremented by the CC and reset at the start of a communication cycle. Valid values are 0 to 16000.

CCV[5:0]

Cycle Counter Value (vCycleCounter)

Current cycle counter value. The value is incremented by the CC at the start of a communication cycle. Valid values are 0 to 63.

8.6.5 Rate Correction Value (RCV)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RCV	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD118	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	RCV11	RCV10	RCV9	RCV8	RCV7	RCV6	RCV5	RCV4	RCV3	RCV2	RCV1	RCV0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

RCV[11:0]

Rate Correction Value (vRateCorrection)

Rate correction value (two's complement). Calculated internal rate correction value **before** limitation. If the RCV value exceeds the limits defined by **GTUC10.MRC[10:0]**, flag **SFS.RCLR** is set to '1'.

8.6.6 Offset Correction Value (OCV)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OCV	R	0	0	0	0	0	0	0	0	0	0	0	0	0	OCV18	OCV17	OCV16
0xD11C	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	OCV15	OCV14	OCV13	OCV12	OCV11	OCV10	OCV9	OCV8	OCV7	OCV6	OCV5	OCV4	OCV3	OCV2	OCV1	OCV0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

OCV[18:0]

Offset Correction Value (vOffsetCorrection)

Offset correction value (two's complement). Calculated internal offset correction value **before** limitation. If the OCV value exceeds the limits defined by **GTUC10.MOC[13:0]**, flag **SFS.OCLR** is set to '1'.

Note: The external rate / offset correction value is added to the limited rate / offset correction value.

8.6.7 Sync Frame Status (SFS)

The maximum number of valid sync frames in a communication cycle is 15.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SFS	R	0	0	0	0	0	0	0	0	0	0	0	0	RCLR	MRCS	OCLR	MOCS
0xD120	W																

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	VSBO3	VSBO2	VSBO1	VSBO0	VSBE3	VSBE2	VSBE1	VSBE0	VSAO3	VSAO2	VSAO1	VSAO0	VSAE3	VSAE2	VSAE1	VSAE0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

VSAE[3:0]

Valid Sync Frames Channel A, **even** communication cycle (vSyncFramesEvenA)

Holds the number of valid sync frames received on channel A in the even communication cycle. If transmission of sync frames is enabled by **SUCC1.TXSY** the value is incremented by one. The value is updated during the NIT of each even communication cycle.

VSAO[3:0]

Valid Sync Frames Channel A, **odd** communication cycle (vSyncFramesOddA)

Holds the number of valid sync frames received on channel A in the odd communication cycle. If transmission of sync frames is enabled by **SUCC1.TXSY** the value is incremented by one. The value is updated during the NIT of each odd communication cycle.

VSBE[3:0]

Valid Sync Frames Channel B, **even** communication cycle (vSyncFramesEvenB)

Holds the number of valid sync frames received on channel B in the even communication cycle. If transmission of sync frames is enabled by **SUCC1.TXSY** the value is incremented by one. The value is updated during the NIT of each even communication cycle.

VSBO[3:0]

Valid Sync Frames Channel B, **odd** communication cycle (vSyncFramesOddB)

Holds the number of valid sync frames received on channel B in the odd communication cycle. If transmission of sync frames is enabled by **SUCC1.TXSY** the value is incremented by one. The value is updated during the NIT of each odd communication cycle.

Note: The bit fields above are only valid if the respective channel is assigned to the CC by **SUCC1.CCHA** or **SUCC1.CCHB**.

MOCS

Missing Offset Correction Signal

The Missing Offset Correction flag signals to the Host, that no offset correction calculation can be performed because no sync frames were received. The flag is updated by the CC at start of offset correction phase.

1 = Missing offset correction signal

0 = Offset correction signal valid

OCLR

Offset Correction Limit Reached

The Offset Correction Limit Reached flag signals to the Host, that the offset correction value has exceeded its limit as defined by **GTUC10.MOC[13:0]**. The flag is updated by the CC at start of offset correction phase.

1 = Offset correction limit reached

0 = Offset correction below limit

MRCS

Missing Rate Correction Signal

The Missing Rate Correction flag signals to the Host, that no rate correction calculation can be performed because no pairs of even / odd sync frames were received. The flag is updated by the CC at start of offset correction phase.

1 = Missing rate correction signal

0 = Rate correction signal valid

RCLR

Rate Correction Limit Reached

The Rate Correction Limit Reached flag signals to the Host, that the rate correction value has exceeded its limit as defined by **GTUC10.MRC[10:0]**. The flag is updated by the CC at start of offset correction phase.

1 = Rate correction limit reached

0 = Rate correction below limit

8.6.8 Symbol Window and NIT Status (SWNIT)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWNIT	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD124	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	SBNB	SENB	SBNA	SENA	MTSB	MTSA	TCSB	SBSB	SESB	TCSA	SBSA	SESA
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Symbol window related status information. Updated by the CC at the end of the symbol window for each channel. During startup the status data is not updated.

SESA

Syntax Error in Symbol Window Channel A (vSS!SyntaxErrorA)

1 = Syntax error during symbol window detected on channel A

0 = No syntax error detected

SBSA

Slot Boundary Violation in Symbol Window Channel A (vSS!BViolationA)

1 = Slot boundary violation during symbol window detected on channel A

0 = No slot boundary violation detected

TCSA

Transmission Conflict in Symbol Window Channel A (vSS!TxConflictA)

1 = Transmission conflict in symbol window detected on channel A

0 = No transmission conflict detected

SESB

Syntax Error in Symbol Window Channel B (vSS!SyntaxErrorB)

1 = Syntax error during symbol window detected on channel B

0 = No syntax error detected

SBSB

Slot Boundary Violation in Symbol Window Channel B (vSS!BViolationB)

1 = Slot boundary violation during symbol window detected on channel B

0 = No slot boundary violation detected

TCSB

Transmission Conflict in Symbol Window Channel B (vSS!TxConflictB)

1 = Transmission conflict in symbol window detected on channel B

0 = No transmission conflict detected

MTSA

MTS Received on Channel A (vSS!ValidMTSA)

Media Access Test symbol received on channel A during the preceding symbol window. Updated by the CC for each channel at the end of the symbol window. When this bit is set to '1', also interrupt flag **SIR.MTSA** is set to '1'.

1 = MTS symbol received on channel A

0 = No MTS symbol received on channel A

MTSB

MTS Received on Channel B (vSS!ValidMTSB)

Media Access Test symbol received on channel B during the preceding symbol window. Updated by the CC for each channel at the end of the symbol window. When this bit is set to '1', also interrupt flag **SIR.MTSB** is set to '1'.

1 = MTS symbol received on channel B

0 = No MTS symbol received on channel B

NIT related status information. Updated by the CC at the end of the NIT for each channel:

SENA

Syntax Error during NIT Channel A (vSS!SyntaxErrorA)

1 = Syntax error during NIT detected on channel A

0 = No syntax error detected

SBNA

Slot Boundary Violation during NIT Channel A (vSS!BViolationA)

1 = Slot boundary violation during NIT detected on channel A

0 = No slot boundary violation detected

SENB

Syntax Error during NIT Channel B (vSS!SyntaxErrorB)

1 = Syntax error during NIT detected on channel B

0 = No syntax error detected

SBNB

Slot Boundary Violation during NIT Channel B (vSS!BViolationB)

1 = Slot boundary violation during NIT detected on channel B

0 = No slot boundary violation detected

8.6.9 Aggregated Channel Status (ACS)

The aggregated channel status provides the Host with an accrued status of channel activity for all communication slots regardless of whether they are assigned for transmission or subscribed for reception. The aggregated channel status also includes status data from the symbol window and the network idle time. The status data is updated (set) after each slot and aggregated until it is reset by the Host. During startup the status data is not updated. A flag is cleared by writing a '1' to the corresponding bit position. Writing a '0' has no effect on the flag. A hard reset will also clear the register.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ACS	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD128	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	SBVB	CIB	CEDB	SEDB	VFRB	0	0	0	SBVA	CIA	CEDA	SEDA	VFRA
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

VFRA

Valid Frame Received on Channel A (vSS!ValidFrameA)

One or more valid frames were received on channel A in any static or dynamic slot during the observation period.

1 = Valid frame(s) received on channel A

0 = No valid frame received

SEDA

Syntax Error Detected on Channel A (vSS!SyntaxErrorA)

One or more syntax errors in static or dynamic slots, symbol window, and NIT were observed on channel A.

1 = Syntax error(s) observed on channel A

0 = No syntax error observed

CEDA

Content Error Detected on Channel A (vSS!ContentErrorA)

One or more frames with a content error were received on channel A in any static or dynamic slot during the observation period.

1 = Frame(s) with content error received on channel A

0 = No frame with content error received

CIA

Communication Indicator Channel A

One or more valid frames were received on channel A in slots that also contained any additional communication during the observation period, i.e. one or more slots received a valid frame AND had any combination of either syntax error OR content error OR slot boundary violation.

1 = Valid frame(s) received on channel A in slots containing any additional communication

0 = No valid frame(s) received in slots containing any additional communication

SBVA

Slot Boundary Violation on Channel A (vSS!BViolationA)

One or more slot boundary violations were observed on channel A at any time during the observation period (static or dynamic slots, symbol window, and NIT).

1 = Slot boundary violation(s) observed on channel A

0 = No slot boundary violation observed

VFRB

Valid Frame Received on Channel B (vSS!ValidFrameB)

One or more valid frames were received on channel B in any static or dynamic slot during the observation period. Reset under control of the Host.

1 = Valid frame(s) received on channel B

0 = No valid frame received

SEDB

Syntax Error Detected on Channel B (vSS!SyntaxErrorB)

One or more syntax errors in static or dynamic slots, symbol window, and NIT were observed on channel B.

1 = Syntax error(s) observed on channel B

0 = No syntax error observed

CEDB

Content Error Detected on Channel B (vSS!ContentErrorB)

One or more frames with a content error were received on channel B in any static or dynamic slot during the observation period.

1 = Frame(s) with content error received on channel B

0 = No frame with content error received

CIB

Communication Indicator Channel B

One or more valid frames were received on channel B in slots that also contained any additional communication during the observation period, i.e. one or more slots received a valid frame AND had any combination of either syntax error OR content error OR slot boundary violation.

1 = Valid frame(s) received on channel B in slots containing any additional communication

0 = No valid frame(s) received in slots containing any additional communication

SBVB

Slot Boundary Violation on Channel B (vSS!BViolationB)

One or more slot boundary violations were observed on channel B at any time during the observation period (static or dynamic slots, symbol window, and NIT).

1 = Slot boundary violation(s) observed on channel B

0 = No slot boundary violation observed

Note: The set condition of flags **CIA** and **CIB** is also fulfilled if there is only one single frame in the slot and the slot boundary at the end of the slot is reached during the frames channel idle recognition phase.

When one of the flags **SEDB**, **CEDB**, **CIB**, **SBVB** changes from '0' to '1', interrupt flag **EIR.EDB** is set to '1'.When one of the flags **SEDA**, **CEDA**, **CIA**, **SBVA** changes from '0' to '1', interrupt flag **EIR.EDA** is set to '1'.

8.6.10 Even Sync ID [1...15] (ESIDn)

Registers ESID1 to ESID15 hold the frame IDs of the sync frames received in **even** communication cycles, sorted in ascending order, with register ESID1 holding the lowest received sync frame ID. If the node itself transmits a sync frame in an even communication cycle, register ESID1 holds the respective sync frame ID as configured in message buffer 0 and flags **RXEA**, **RXEB** are set. The value is updated during the NIT of each even communication cycle.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ESIDn	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD130 - 0xD168	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RXEB	RXEA	0	0	0	0	EID9	EID8	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EID[9:0]

Even Sync ID (vsSyncIDListA,B even)

Sync frame ID even communication cycle.

RXEA

Received / Configured Even Sync ID on Channel A

Signals that a sync frame corresponding to the stored even sync ID was received on channel A or that the node is configured to be a sync node with key slot = **EID[9:0]** (ESID1 only).

1 = Sync frame received on channel A / node configured to transmit sync frames

0 = No sync frame received on channel A / node not configured to transmit sync frames

RXEB

Received / Configured Even Sync ID on Channel B

Signals that a sync frame corresponding to the stored even sync ID was received on channel B or that the node is configured to be a sync node with key slot = **EID[9:0]** (ESID1 only).

1 = Sync frame received on channel B / node configured to transmit sync frames

0 = No sync frame received on channel B / node not configured to transmit sync frames

8.6.11 Odd Sync ID [1...15] (OSIDn)

Registers OSID1 to OSID15 hold the frame IDs of the sync frames received in **odd** communication cycles, sorted in ascending order, with register OSID1 holding the lowest received sync frame ID. If the node itself transmits a sync frame in an odd communication cycle, register OSID1 holds the respective sync frame ID as configured in message buffer 0 and flags **RXOA**, **RXOB** are set. The value is updated during the NIT of each odd communication cycle.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OSIDn R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD170 - 0xD1A8 W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RXOB	RXOA	0	0	0	0	OID9	OID8	OID7	OID6	OID5	OID4	OID3	OID2	OID1	OID0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

OID[9:0]

Odd Sync ID (vsSyncIDListA,B odd)

Sync frame ID odd communication cycle.

RXOA

Received / Configured Odd Sync ID on Channel A

Signals that a sync frame corresponding to the stored odd sync ID was received on channel A or that the node is configured to be a sync node with key slot = **OID[9:0]** (OSID1 only).

1 = Sync frame received on channel A / node configured to transmit sync frames

0 = No sync frame received on channel A / node not configured to transmit sync frames

RXOB

Received / Configured Odd Sync ID on Channel B

Signals that a sync frame corresponding to the stored odd sync ID was received on channel B or that the node is configured to be a sync node with key slot = **OID[9:0]** (OSID1 only).

1 = Sync frame received on channel B / node configured to transmit sync frames

0 = No sync frame received on channel B / node not configured to transmit sync frames

8.6.12 Network Management Vector [1...3] (NMVn)

The three network management registers hold the accrued NM vector (configurable 0 to 12 bytes). The accrued NM vector is generated by the CC by bit-wise ORing each NM vector received (valid static frames with PPI = '1') on each channel (see 5.6 Network Management).
The CC updates the NM vector at the end of each communication cycle as long as the CC is either in NORMAL_ACTIVE or NORMAL_PASSIVE state.
NMVn-bytes exceeding the configured NM vector length are not valid.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NMVn	R	NM31	NM30	NM29	NM28	NM27	NM26	NM25	NM24	NM23	NM22	NM21	NM20	NM19	NM18	NM17	NM16
0xD1B0 - 0xD1B8	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	NM15	NM14	NM13	NM12	NM11	NM10	NM9	NM8	NM7	NM6	NM5	NM4	NM3	NM2	NM1	NM0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Assignment of data bytes to network management vector below shows the assignment of the received payload's data bytes to the network management vector.

Bit / Word	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NMç1	Data3								Data2								Data1								Data0							
NMç2	Data7								Data6								Data5								Data4							
NMç3	Data11								Data10								Data9								Data8							

Table 14: Assignment of data bytes to network management vector

8.7 Message Buffer Control Registers

8.7.1 Message RAM Configuration (MRC)

The Message RAM Configuration register defines the number of message buffers assigned to the static segment, dynamic segment, and FIFO. The register can be written during DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MRC	R	0	0	0	0	0	SPLM*	SEC1*	SEC0*	LCB7*	LCB6*	LCB5*	LCB4*	LCB3*	LCB2*	LCB1*	LCB0*
0xD300	W																
Reset		0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	FFB7*	FFB6*	FFB5*	FFB4*	FFB3*	FFB2*	FFB1*	FFB0*	FDB7*	FDB6*	FDB5*	FDB4*	FDB3*	FDB2*	FDB1*	FDB0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FDB[7:0]

First Dynamic Buffer

0 = No group of message buffers exclusively for the static segment configured

1...127 = Message buffers 0 to **FDB** - 1 reserved for static segment

≥128 = No dynamic message buffers configured

FFB[7:0]

First Buffer of FIFO

0 = All message buffers assigned to the FIFO

1...127 = Message buffers from FFB to LCB assigned to the FIFO

≥128 = No message buffer assigned to the FIFO

LCB[7:0]

Last Configured Buffer

0...127 = Number of message buffers is **LCB** + 1

≥128 = No message buffer configured

SEC[1:0]

Secure Buffers

Not evaluated when the CC is in DEFAULT_CONFIG or CONFIG state.

00 = Reconfiguration of message buffers enabled with numbers < **FFB** enabled
Exception: In nodes configured for sync frame transmission or for single slot mode operation message buffer 0 (and if **SPLM** = '1', also message buffer 1) is always locked

01 = Reconfiguration of message buffers with numbers < **FDB** and with numbers ≥ **FFB** locked
and transmission of message buffers for static segment with numbers ≥ **FDB** disabled

10 = Reconfiguration of all message buffers locked

11 = Reconfiguration of all message buffers locked
and transmission of message buffers for static segment with numbers ≥ **FDB** disabled

SPLM

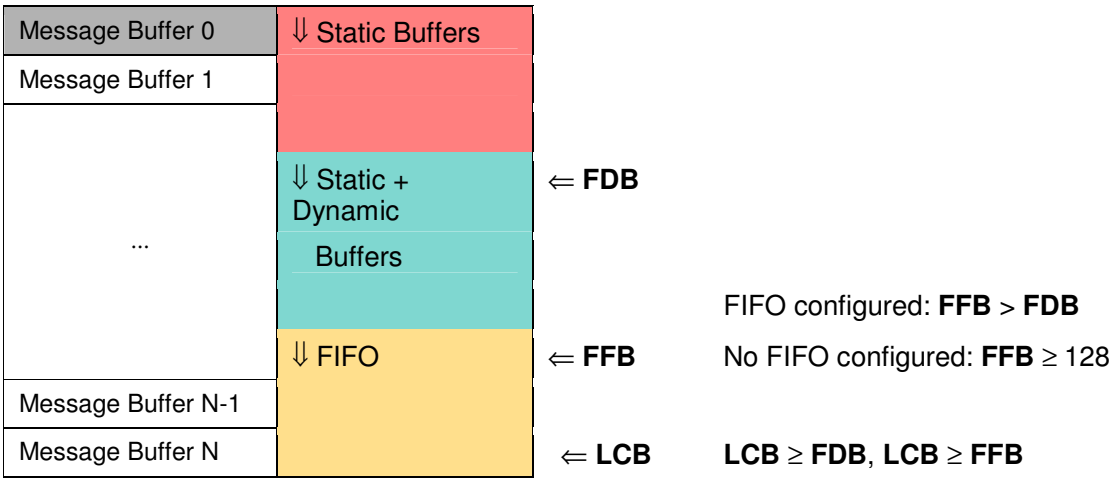
Sync Frame Payload Multiplex

This bit is only evaluated if the node is configured as sync node (**SUCC1.TXSY** = '1') or for single slot mode operation (**SUCC1.TSM** = '1'). When this bit is set to '1' message buffers 0 and 1 are dedicated for sync frame transmission with different payload data on channel A and B. When this bit is set to '0', sync frames are transmitted from message buffer 0 with the same payload data on both channels. Note that the channel filter configuration for message buffer 0 resp. message buffer 1 has to be chosen accordingly.

1 = Both message buffers 0 and 1 are locked against reconfiguration

0 = Only message buffer 0 locked against reconfiguration

Note: In case the node is configured as sync node (**SUCC1.TXSY** = '1') or for single slot mode operation (**SUCC1.TSM** = '1'), message buffer 0 resp. 1 is reserved for sync frames or single slot frames and have to be configured with the node-specific key slot ID. In case the node is neither configured as sync node nor for single slot operation message buffer 0 resp. 1 is treated like all other message buffers.



The programmer has to ensure that the configuration defined by **FDB[7:0]**, **FFB[7:0]**, and **LCB[7:0]** is valid. **The CC does not check for erroneous configurations!**

Note: The maximum number of header sections is 128. This means a maximum of 128 message buffers can be configured. The maximum length of a data section is 254 bytes. The length of the data section may be configured differently for each message buffer. For details see Section 5.12 Message RAM.
The payload length configured and the length of the data section need to be configured identical for all message buffers belonging to the FIFO via **WRHS2.PLC[6:0]** and **WRHS3.DP[10:0]**.
When the CC is not in DEFAULT_CONFIG or CONFIG state reconfiguration of message buffers belonging to the FIFO is locked.

8.7.2 FIFO Rejection Filter (FRF)

The FIFO Rejection Filter defines a user specified sequence of bits to which channel, frame ID, and cycle count of the incoming frames are compared. Together with the FIFO Rejection Filter Mask this register determines whether a message is rejected by the FIFO. The FRF register can be written during DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FRF	R	0	0	0	0	0	0	0	RNF*	RSS*	CYF6*	CYF5*	CYF4*	CYF3*	CYF2*	CYF1*	CYF0*
0xD304	W																
Reset		0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	FID10*	FID9*	FID8*	FID7*	FID6*	FID5*	FID4*	FID3*	FID2*	FID1*	FID0*	CH1*	CH0*
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CH[1:0]

Channel Filter

11 = no reception
10 = receive only on channel A
01 = receive only on channel B
00 = receive on both channels

Note: If reception on both channels is configured, also in static segment always both frames (from channel A and B) are stored in the FIFO, even if they are identical.

FID[10:0]

Frame ID Filter

Determines the frame ID to be rejected by the FIFO. With the additional configuration of register FRFM, the corresponding frame ID filter bits are ignored, which results in further rejected frame IDs. When **FRFM.MFID[10:0]** is zero, a frame ID filter value of zero means that **no** frame ID is rejected.

0...2047 = Frame ID filter values

CYF[6:0]

Cycle Counter Filter

The 7-bit cycle counter filter determines the cycle set to which frame ID and channel rejection filter are applied. In cycles **not** belonging to the cycle set specified by **CYF[6:0]**, all frames are rejected. For details about the configuration of the cycle counter filter see Section 5.7.2 Cycle Counter Filtering.

RSS

Reject in Static Segment

If this bit is set, the FIFO is used only for the dynamic segment.

1 = Reject messages in static segment

0 = FIFO also used for static segment

RNF

Reject Null Frames

If this bit is set, received null frames are not stored in the FIFO.

1 = Reject all null frames

0 = Null frames are stored in the FIFO

8.7.3 FIFO Rejection Filter Mask (FRFM)

The FIFO Rejection Filter Mask specifies which of the corresponding frame ID filter bits are relevant for rejection filtering. If a bit is set, it indicates that the corresponding bit in the FRF register will not be considered for rejection filtering. The FRFM register can be written during DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FRFM	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD308	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	MFID 10*	MFID 9*	MFID 8*	MFID 7*	MFID 6*	MFID 5*	MFID 4*	MFID 3*	MFID 2*	MFID 1*	MFID 0*	0	0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MFID[10:0]

Mask Frame ID Filter

1 = Ignore corresponding frame ID filter bit.

0 = Corresponding frame ID filter bit is used for rejection filtering

8.7.4 FIFO Critical Level (FCL)

The CC accepts modifications of the register in DEFAULT_CONFIG or CONFIG state only.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FCL	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD30C	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	0	0	CL7*	CL6*	CL5*	CL4*	CL3*	CL2*	CL1*	CL0*
	W																
Reset		0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

CL[7:0]

Critical Level

When the receive FIFO fill level **FSR.RFFL[7:0]** is equal or greater than the critical level configured by **CL[7:0]**, the receive FIFO critical level flag **FSR.RFCL** is set. If **CL[7:0]** is programmed to values > 128, bit **FSR.RFCL** is never set. When **FSR.RFCL** changes from ’0’ to ’1’ bit **SIR.RFCL** is set to ’1’, and if enabled, an interrupt is generated.

8.8 Message Buffer Status Registers

8.8.1 Message Handler Status (MHDS)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MHDS	R	0	MBU6	MBU5	MBU4	MBU3	MBU2	MBU1	MBU0	0	MBT6	MBT5	MBT4	MBT3	MBT2	MBT1	MBT0
0xD310	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	FMB6	FMB5	FMB4	FMB3	FMB2	FMB1	FMB0	CRAM	MFMB	FMBD	PTBF2	PTBF1	PMR	POBF	PIBF
	W																
Reset		0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

A flag is cleared by writing a ’1’ to the corresponding bit position. Writing a ’0’ has no effect on the flag. The register will also be cleared by hard reset or by CHI command CLEAR_RAMS.

PIBF

Parity Error Input Buffer RAM 1,2

1 = Parity error occurred when reading Input Buffer RAM 1,2

0 = No parity error

POBF

Parity Error Output Buffer RAM 1,2

1 = Parity error occurred when reading Output Buffer RAM 1,2

0 = No parity error

PMR

Parity Error Message RAM

1 = Parity error occurred when reading the Message RAM

0 = No parity error

PTBF1

Parity Error Transient Buffer RAM A

1 = Parity error occurred when reading Transient Buffer RAM A

0 = No parity error

PTBF2

Parity Error Transient Buffer RAM B

1 = Parity error occurred when reading Transient Buffer RAM B

0 = No parity error

Note: When one of the flags **PIBF**, **POBF**, **PMR**, **PTBF1**, **PTBF2** changes from ’0’ to ’1’ **EIR.PERR** is set to ’1’.

FMBD

Faulty Message Buffer Detected

1 = Message buffer referenced by **FMB[6:0]** holds faulty data due to a parity error

0 = No faulty message buffer

MFMB

Multiple Faulty Message Buffers detected

1 = Another faulty message buffer was detected while flag **FMBD** is set

0 = No additional faulty message buffer

CRAM

Clear all internal RAM’s

Signals that execution of the CHI command CLEAR_RAMS is ongoing (all bits of all internal RAM blocks are written to ’0’). The bit is set by hard reset or by CHI command CLEAR_RAMS.

1 = Execution of the CHI command CLEAR_RAMS ongoing

0 = No execution of the CHI command CLEAR_RAMS

FMB[6:0]

Faulty Message Buffer

Parity error occurred when reading from the message buffer or when transferring data from Input Buffer or Transient Buffer 1,2 to the message buffer referenced by **FMB[6:0]**. Value only valid when one of the flags **PIBF**, **PMR**, **PTBF1**, **PTBF2**, and flag **FMBD** is set. Updated only after the Host has reset flag **FMBD**.

MBT[6:0]

Message Buffer Transmitted

Number of last successfully transmitted message buffer. If the message buffer is configured for single-shot mode, the respective **TXR** flag in the TXRQ1/2/3/4 registers was reset.

MBU[6:0]

Message Buffer Updated

Number of message buffer that was updated last by the CC. For this message buffer the respective **ND** and / or **MBC** flag in the NDAT1/2/3/4 registers and the MBSC1/2/3/4 registers are also set.

Note: **MBT[6:0]** and **MBU[6:0]** are reset when the CC leaves CONFIG state or enters STARTUP state.

8.8.2 Last Dynamic Transmit Slot (LDTS)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LDTS	R	0	0	0	0	0	LDTB10	LDTB9	LDTB8	LDTB7	LDTB6	LDTB5	LDTB4	LDTB3	LDTB2	LDTB1	LDTB0
0xD314	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	LDTA10	LDTA9	LDTA8	LDTA7	LDTA6	LDTA5	LDTA4	LDTA3	LDTA2	LDTA1	LDTA0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The register is reset when the CC leaves CONFIG state or enters STARTUP state.

LDTA[10:0]

Last Dynamic Transmission Channel A

Value of vSlotCounter[A] at the time of the last frame transmission on channel A in the dynamic segment of this node. It is updated at the end of the dynamic segment and is reset to zero if no frame was transmitted during the dynamic segment.

LDTB[10:0]

Last Dynamic Transmission Channel B

Value of vSlotCounter[B] at the time of the last frame transmission on channel B in the dynamic segment of this node. It is updated at the end of the dynamic segment and is reset to zero if no frame was transmitted during the dynamic segment.

8.8.3 FIFO Status Register (FSR)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FSR	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD318	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	RFFL7	RFFL6	RFFL5	RFFL4	RFFL3	RFFL2	RFFL1	RFFL0	0	0	0	0	0	RFO	RFCL	RFNE
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The register is reset when the CC leaves CONFIG state or enters STARTUP state.

RFNE

Receive FIFO Not Empty

This flag is set by the CC when a received valid frame (data or null frame depending on rejection mask) was stored in the FIFO. In addition, interrupt flag **SIR.RFNE** is set. The bit is reset after the Host has read all message from the FIFO.

1 = Receive FIFO is not empty

0 = Receive FIFO is empty

RFCL

Receive FIFO Critical Level

This flag is set when the receive FIFO fill level **RFFL[7:0]** is equal or greater than the critical level as configured by **FCL.CL[7:0]**. The flag is cleared by the CC as soon as **RFFL[7:0]** drops below **FCL.CL[7:0]**. When **RFCL** changes from '0' to '1' bit **SIR.RFCL** is set to '1', and if enabled, an interrupt is generated.

1 = Receive FIFO critical level reached

0 = Receive FIFO below critical level

RFO

Receive FIFO Overrun

The flag is set by the CC when a receive FIFO overrun is detected. When a receive FIFO overrun occurs, the oldest message is overwritten with the actual received message. In addition, interrupt flag **EIR.RFO** is set.The flag is cleared by the next FIFO read access issued by the Host.

1 = A receive FIFO overrun has been detected

0 = No receive FIFO overrun detected

RFFL[7:0]

Receive FIFO Fill Level

Number of FIFO buffers filled with new data not yet read by the Host. Maximum value is 128.

8.8.4 Message Handler Constraints Flags (MHDF)

Some constraints exist for the Message Handler regarding eray_bclk frequency, Message RAM configuration, and FlexRay bus traffic (see Addendum to E-Ray FlexRay IP-Module Specification). To simplify software development, constraints violations are reported by setting flags in the MHDF.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MHDF	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD31C	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	0	WAHP	0	0	TBFB	TBFA	FNFB	FNFA	SNUB	SNUA
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

A flag is cleared by writing a ’1’ to the corresponding bit position. Writing a ’0’ has no effect on the flag. A hard reset will also clear the register. The register is reset when the CC leaves CONFIG state or enters STARTUP state.

SNUA

Status Not Updated Channel A

This flag is set by the CC when the Message Handler, due to overload condition, was not able to update a message buffer’s status MBS with respect to channel A.

1 = MBS for channel A not updated

0 = No overload condition occurred when updating MBS for channel A

SNUB

Status Not Updated Channel B

This flag is set by the CC when the Message Handler, due to overload condition, was not able to update a message buffer’s status MBS with respect to channel B.

1 = MBS for channel B not updated

0 = No overload condition occurred when updating MBS for channel B

FNFA

Find Sequence Not Finished Channel A

This flag is set by the CC when the Message Handler, due to overload condition, was not able to finish a find sequence (scan of Message RAM for matching message buffer) with respect to channel A.

1 = Find sequence not finished for channel A

0 = No find sequence not finished for channel A

FNFB

Find Sequence Not Finished Channel B

This flag is set by the CC when the Message Handler, due to overload condition, was not able to finish a find sequence (scan of Message RAM for matching message buffer) with respect to channel B.

1 = Find sequence not finished for channel B

0 = No find sequence not finished for channel B

TBFA

Transient Buffer Access Failure A

This flag is set by the CC when a read or write access to TBF A requested by PRT A could not complete within the available time.

1 = TBF A access failure

0 = No TBF A access failure

TBFB

Transient Buffer Access Failure B

This flag is set by the CC when a read or write access to TBF B requested by PRT B could not complete within the available time.

1 = TBF B access failure

0 = No TBF B access failure

WAHP

Write Attempt to Header Partition

Outside DEFAULT_CONFIG and CONFIG state this flag is set by the CC when the message handler tries to write message data into the header partition of the Message RAM due to faulty configuration of a message buffer. The write attempt is not executed, to protect the header partition from unintended write accesses.

1 = Write attempt to header partition

0 = No write attempt to header partition

Note: When one of the flags **SNUA**, **SNUB**, **FNFA**, **FNFB**, **TBFA**, **TBFB**, **WAHP** changes from '0' to '1', interrupt flag **EIR.MHF** is set to '1'.

8.8.5 Transmission Request 1/2/3/4 (TXRQ1/2/3/4)

The four registers reflect the state of the **TXR** flags of all configured message buffers. The flags are evaluated for transmit buffers only. If the number of configured message buffers is less than 128, the remaining **TXR** flags have no meaning.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXRQ4	R	TXR127	TXR126	TXR125	TXR124	TXR123	TXR122	TXR121	TXR120	TXR119	TXR118	TXR117	TXR116	TXR115	TXR114	TXR113	TXR112
0xD32C	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	TXR111	TXR110	TXR109	TXR108	TXR107	TXR106	TXR105	TXR104	TXR103	TXR102	TXR101	TXR100	TXR99	TXR98	TXR97	TXR96

		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXRQ3	R	TXR95	TXR94	TXR93	TXR92	TXR91	TXR90	TXR89	TXR88	TXR87	TXR86	TXR85	TXR84	TXR83	TXR82	TXR81	TXR80
0xD328	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	TXR79	TXR78	TXR77	TXR76	TXR75	TXR74	TXR73	TXR72	TXR71	TXR70	TXR69	TXR68	TXR67	TXR66	TXR65	TXR64
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXRQ2	R	TXR63	TXR62	TXR61	TXR60	TXR59	TXR58	TXR57	TXR56	TXR55	TXR54	TXR53	TXR52	TXR51	TXR50	TXR49	TXR48
0xD324	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	TXR47	TXR46	TXR45	TXR44	TXR43	TXR42	TXR41	TXR40	TXR39	TXR38	TXR37	TXR36	TXR35	TXR34	TXR33	TXR32
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXRQ1	R	TXR31	TXR30	TXR29	TXR28	TXR27	TXR26	TXR25	TXR24	TXR23	TXR22	TXR21	TXR20	TXR19	TXR18	TXR17	TXR16
0xD320	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	TXR15	TXR14	TXR13	TXR12	TXR11	TXR10	TXR9	TXR8	TXR7	TXR6	TXR5	TXR4	TXR3	TXR2	TXR1	TXR0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TXR[127:0]

Transmission Request

If the flag is set, the respective message buffer is ready for transmission respectively transmission of this message buffer is in progress. In single-shot mode the flags are reset after transmission has completed.

8.8.6 New Data 1/2/3/4 (NDAT1/2/3/4)

The four registers reflect the state of the **ND** flags of all configured message buffers. **ND** flags belonging to transmit buffers have no meaning. If the number of configured message buffers is less than 128, the remaining **ND** flags have no meaning. The registers are reset when the CC leaves CONFIG state or enters STARTUP state.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NDAT4	R	ND127	ND126	ND125	ND124	ND123	ND122	ND121	ND120	ND119	ND118	ND117	ND116	ND115	ND114	ND113	ND112
0xD33C	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	ND111	ND110	ND109	ND108	ND107	ND106	ND105	ND104	ND103	ND102	ND101	ND100	ND99	ND98	ND97	ND96
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NDAT3	R	ND95	ND94	ND93	ND92	ND91	ND90	ND89	ND88	ND87	ND86	ND85	ND84	ND83	ND82	ND81	ND80
0xD338	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	ND79	ND78	ND77	ND76	ND75	ND74	ND73	ND72	ND71	ND70	ND69	ND68	ND67	ND66	ND65	ND64
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NDAT2	R	ND63	ND62	ND61	ND60	ND59	ND58	ND57	ND56	ND55	ND54	ND53	ND52	ND51	ND50	ND49	ND48
0xD334	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	ND47	ND46	ND45	ND44	ND43	ND42	ND41	ND40	ND39	ND38	ND37	ND36	ND35	ND34	ND33	ND32
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NDAT1	R	ND31	ND30	ND29	ND28	ND27	ND26	ND25	ND24	ND23	ND22	ND21	ND20	ND19	ND18	ND17	ND16
0xD330	W																

		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	ND15	ND14	ND13	ND12	ND11	ND10	ND9	ND8	ND7	ND6	ND5	ND4	ND3	ND2	ND1	ND0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ND[127:0]

New Data

The flags are set when a valid received data frame matches the message buffer's filter configuration, independent of the payload length received or the payload length configured for that message buffer. The flags are not set after reception of null frames except for message buffers belonging to the receive FIFO. An **ND** flag is reset when the header section of the corresponding message buffer is reconfigured or when the data section has been transferred to the Output Buffer.

8.8.7 Message Buffer Status Changed 1/2/3/4 (MBSC1/2/3/4)

The four registers reflect the state of the **MBC** flags of all configured message buffers. If the number of configured message buffers is less than 128, the remaining **MBC** flags have no meaning. The registers are reset when the CC leaves CONFIG state or enters STARTUP state.

		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MBSC4	R	MBC127	MBC126	MBC125	MBC124	MBC123	MBC122	MBC121	MBC120	MBC119	MBC118	MBC117	MBC116	MBC115	MBC114	MBC113	MBC112
0xD34C	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	MBC111	MBC110	MBC109	MBC108	MBC107	MBC106	MBC105	MBC104	MBC103	MBC102	MBC101	MBC100	MBC99	MBC98	MBC97	MBC96
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MBSC3	R	MBC95	MBC94	MBC93	MBC92	MBC91	MBC90	MBC89	MBC88	MBC87	MBC86	MBC85	MBC84	MBC83	MBC82	MBC81	MBC80
0xD348	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	MBC79	MBC78	MBC77	MBC76	MBC75	MBC74	MBC73	MBC72	MBC71	MBC70	MBC69	MBC68	MBC67	MBC66	MBC65	MBC64
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MBSC2	R	MBC63	MBC62	MBC61	MBC60	MBC59	MBC58	MBC57	MBC56	MBC55	MBC54	MBC53	MBC52	MBC51	MBC50	MBC49	MBC48
0xD344	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	MBC47	MBC46	MBC45	MBC44	MBC43	MBC42	MBC41	MBC40	MBC39	MBC38	MBC37	MBC36	MBC35	MBC34	MBC33	MBC32
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MBSC1	R	MBC31	MBC30	MBC29	MBC28	MBC27	MBC26	MBC25	MBC24	MBC23	MBC22	MBC21	MBC20	MBC19	MBC18	MBC17	MBC16
0xD340	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	MBC15	MBC14	MBC13	MBC12	MBC11	MBC10	MBC9	MBC8	MBC7	MBC6	MBC5	MBC4	MBC3	MBC2	MBC1	MBC0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MBC[127:0]

Message Buffer Status Changed

An **MBC** flag is set whenever the Message Handler changes one of the status flags **VFRA**, **VFRB**, **SEOA**, **SEOB**, **CEOA**, **CEOB**, **SVOA**, **SVOB**, **TCIA**, **TCIB**, **ESA**, **ESB**, **MLST**, **FTA**, **FTB** in the header section (see Message Buffer Status (MBS) and 5.12.1 Header Partition, header 4) of the respective message buffer. An **MBC** flag is reset when the header section of the corresponding message buffer is reconfigured or when it has been transferred to the Output Buffer.

8.9 Identification Registers

8.9.1 Core Release Register (CREL)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CREL	R	REL3	REL2	REL1	REL0	STEP7	STEP6	STEP5	STEP4	STEP3	STEP2	STEP1	STEP0	YEAR3	YEAR2	YEAR1	YEAR0
0xD3F0	W																
Reset		release info															
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	MON7	MON6	MON5	MON4	MON3	MON2	MON1	MON0	DAY7	DAY6	DAY5	DAY4	DAY3	DAY2	DAY1	DAY0

	W																
Reset		release info															

DAY[7:0]

Design Time Stamp, Day

Two digits, BCD-coded.

MON[7:0]

Design Time Stamp, Month

Two digits, BCD-coded.

YEAR[3:0]

Design Time Stamp, Year

One digit, BCD-coded.

STEP[7:0]

Step of Core Release

Two digits, BCD-coded.

REL[3:0]

Core Release

One digit, BCD-coded.

Coding for re below shows how releases are coded in register CREL.

Release	Step	Sub-Step	Name
0	7	0	Beta2
0	7	1	Beta2ct
1	0	0	Revision 1.0.0

Table 15: Coding for releases

8.9.2 Endian Register (ENDN)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ENDN	R	ETV31	ETV30	ETV29	ETV28	ETV27	ETV26	ETV25	ETV24	ETV23	ETV22	ETV21	ETV20	ETV19	ETV18	ETV17	ETV16
0xD3F4	W																
Reset		1	0	0	0	0	1	1	1	0	1	1	0	0	1	0	1
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	ETV15	ETV14	ETV13	ETV12	ETV11	ETV10	ETV9	ETV8	ETV7	ETV6	ETV5	ETV4	ETV3	ETV2	ETV1	ETV0
	W																
Reset		0	1	0	0	0	0	1	1	0	0	1	0	0	0	0	1

ETV[31:0]

Endianness Test Value

The endianness test value is 0x87654321.

8.10 Input Buffer

Double buffer structure consisting of Input Buffer Host and Input Buffer Shadow. While the Host can write to Input Buffer Host, the transfer to the Message RAM is done from Input Buffer Shadow. The Input Buffer holds the header and data sections to be transferred to the selected message buffer in the Message RAM. It is used to configure the message buffers in the Message RAM and to update the data sections of transmit buffers.
When updating the header section of a message buffer in the Message RAM from the Input Buffer, the Message Buffer Status as described in Section Message Buffer Status (MBS) is automatically reset to zero.
The header sections of message buffers belonging to the receive FIFO can only be (re)configured when the CC is in DEFAULT_CONFIG or CONFIG state. For those message buffers only the payload length configured and the data pointer need to be configured via **WRHS2.PLC[6.0]** and **WRHS3.DP[10:0]**. All information required for acceptance filtering is taken from the FIFO rejection filter and the FIFO rejection filter mask.
The data transfer between Input Buffer (IBF) and Message RAM is described in detail in Section 5.11.2.1 Data Transfer from Input Buffer to Message RAM.

8.10.1 Write Data Section [1...64] (WRDSn)

Holds the data words to be transferred to the data section of the addressed message buffer. The data words (DW_n) are written to the Message RAM in transmission order from DW_1 (byte0, byte1) to DW_{PL} (PL = number of data words as defined by the payload length configured **WRHS2.PLC[6:0]**).

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRDSn	R																
0xD400 - 0xD4FC	W	MD31	MD30	MD29	MD28	MD27	MD26	MD25	MD24	MD23	MD22	MD21	MD20	MD19	MD18	MD17	MD16
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R																
	W	MD15	MD14	MD13	MD12	MD11	MD10	MD9	MD8	MD7	MD6	MD5	MD4	MD3	MD2	MD1	MD0
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MD[31:0]

Message Data

MD[7:0] = DW_n, $byte_{n-1}$
MD[15:8] = DW_n, $byte_n$
MD[23:16] = DW_{n+1}, $byte_{n+1}$

MD[31:24] = DW_{n+1}, $byte_{n+2}$

Note: DW127 is located on WRDS64.MD[15:0]. In this case WRDS64.MD[31:16] is unused (no valid data). The Input Buffer RAMs are initialized to zero when leaving hard reset or by CHI command CLEAR_RAMS.

8.10.2 Write Header Section 1 (WRHS1)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRHS1	R	0	0	MBI	TXM	PPIT	CFG	CHB	CHA	0	CYC6	CYC5	CYC4	CYC3	CYC2	CYC1	CYC0
0xD500	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	0	0	0	0	FID10	FID9	FID8	FID7	FID6	FID5	FID4	FID3	FID2	FID1	FID0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FID[10:0]

Frame ID

Frame ID of the selected message buffer. The frame ID defines the slot number for transmission / reception of the respective message. **Message buffers with frame ID = '0' are considered as not valid.**

CYC[6:0]

Cycle Code

The 7-bit cycle code determines the cycle set used for cycle counter filtering. For details about the configuration of the cycle code see Section 5.7.2 Cycle Counter Filtering.

CHA, CHB

Channel Filter Control

The 2-bit channel filtering field associated with each buffer serves as a filter for receive buffers, and as a control field for transmit buffers.

CHA	CHB	**Transmit Buffer** transmit frame on	**Receive Buffer** store frame received from
1	1	both channels (static segment only)	channel A or B (store first semantically valid frame, static segment only)
1	0	channel A	channel A
0	1	channel B	channel B
0	0	no transmission	ignore frame

Note: If a message buffer is configured for the dynamic segment and both bits of the channel filtering field are set to '1', no frames are transmitted resp. received frames are ignored (same function as **CHA = CHB** = '0')

CFG

Message Buffer Direction Configuration Bit

This bit is used to configure the corresponding buffer as transmit buffer or as receive buffer. For message buffers belonging to the receive FIFO the bit is not evaluated.

1 = The corresponding buffer is configured as **Transmit Buffer**

0 = The corresponding buffer is configured as **Receive Buffer**

PPIT

Payload Preamble Indicator Transmit

This bit is used to control the state of the Payload Preamble Indicator in transmit frames. If the bit is set in a static message buffer, the respective message buffer holds network management information. If the bit is set in a dynamic message buffer the first two bytes of the payload segment may be used for message ID filtering by the receiver. Message ID filtering of received FlexRay frames is not supported by the E-Ray module, but can be done by the Host.

1 = Payload Preamble Indicator set

0 = Payload Preamble Indicator not set

TXM

Transmission Mode

This bit is used to select the transmission mode (see Section 5.8.3 Transmit Buffers).

1 = Single-shot mode

0 = Continuous mode

MBI

Message Buffer Interrupt

This bit enables the receive / transmit interrupt for the corresponding message buffer. After a dedicated receive buffer has been updated by the Message Handler, flag **SIR.RXI** and /or **SIR.MBSI** are set. After a transmission has completed flag **SIR.TXI** is set.

1 = The corresponding message buffer interrupt is enabled

0 = The corresponding message buffer interrupt is disabled

8.10.3 Write Header Section 2 (WRHS2)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRHS2	R	0	0	0	0	0	0	0	0	0	PLC6	PLC5	PLC4	PLC3	PLC2	PLC1	PLC0
0xD504	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	CRC10	CRC9	CRC8	CRC7	CRC6	CRC5	CRC4	CRC3	CRC2	CRC1	CRC0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CRC[10:0]

Header CRC (vRF!Header!HeaderCRC)

Receive Buffer: Configuration not required

Transmit Buffer: Header CRC calculated and configured by the Host

For calculation of the header CRC the payload length of the frame send on the bus has to be considered. In static segment the payload length of all frames is configured by **MHDC.SFDL[6:0]**.

PLC[6:0]

Payload Length Configured

Length of data section (number of 2-byte words) as configured by the Host. During static segment the static frame payload length as configured by **MHDC.SFDL[6:0]** defines the payload length for all static frames. If the payload length configured by **PLC[6:0]** is shorter than this value padding bytes are inserted to ensure that frames have proper physical length. The padding pattern is logical zero.

8.10.4 Write Header Section 3 (WRHS3)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRHS3	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD508	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	DP10	DP9	DP8	DP7	DP6	DP5	DP4	DP3	DP2	DP1	DP0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DP[10:0]

Data Pointer

Pointer to the first 32-bit word of the data section of the addressed message buffer in the Message RAM.

8.10.5 Input Buffer Command Mask (IBCM)

Configures how the message buffer in the Message RAM selected by register IBCR is updated. When IBF Host and IBF Shadow are swapped, also mask bits **LHSH**, **LDSH**, and **STXRH** are swapped with bits **LHSS**, **LDSS**, and **STXRS** to keep them attached to the respective Input Buffer transfer.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IBCM	R	0	0	0	0	0	0	0	0	0	0	0	0	0	STXRS	LDSS	LHSS
0xD510	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

R	0	0	0	0	0	0	0	0	0	0	0	0	0	STXRH	LDSH	LHSH
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

LHSH

Load Header Section Host

1 = Header section selected for transfer from Input Buffer to the Message RAM

0 = Header section is not updated

LDSH

Load Data Section Host

1 = Data section selected for transfer from Input Buffer to the Message RAM

0 = Data section is not updated

STXRH

Set Transmission Request Host

If this bit is set to '1', the **TXR** flag for the selected message buffer is set in the TXRQ1/2/3/4 registers to release the message buffer for transmission. In single-shot mode the flag is cleared by the CC after transmission has completed. **TXR** is evaluated for transmit buffers only.

1 = Set **TXR** flag, transmit buffer released for transmission

0 = Reset **TXR** flag

LHSS

Load Header Section Shadow

1 = Header section selected for transfer from Input Buffer to the Message RAM (transfer ongoing or finished)

0 = Header section is not updated

LDSS

Load Data Section Shadow

1 = Data section selected for transfer from Input Buffer to the Message RAM (transfer ongoing or finished)

0 = Data section is not updated

STXRS

Set Transmission Request Shadow

1 = Set **TXR** flag, transmit buffer released for transmission (operation ongoing or finished)

0 = Reset **TXR** flag

8.10.6 Input Buffer Command Request (IBCR)

When the Host writes the number of the target message buffer in the Message RAM to **IBRH[6:0]**, IBF Host and IBF Shadow are swapped. In addition the message buffer numbers stored under **IBRH[6:0]** and **IBRS[6:0]** are also swapped (see also Section 5.11.2.1 Data Transfer from Input Buffer to Message RAM).

With this write operation the **IBSYS** is set to ’1’. The Message Handler then starts to transfer the contents of IBF Shadow to the message buffer in the Message RAM selected by **IBRS[6:0]**.
While the Message Handler transfers the data from IBF Shadow to the target message buffer in the Message RAM, the Host may write the next message into the IBF Host. After the transfer between IBF Shadow and the Message RAM has completed, **IBSYS** is set back to ’0’ and the next transfer to the Message RAM may be started by the Host by writing the respective target message buffer number to **IBRH[6:0]**.
If a write access to **IBRH[6:0]** occurs while **IBSYS** is ’1’, **IBSYH** is set to ’1’. After completion of the ongoing data transfer from IBF Shadow to the Message RAM, IBF Host and IBF Shadow are swapped, **IBSYH** is reset to ’0’. **IBSYS** remains set to ’1’, and the next transfer to the Message RAM is started. In addition the message buffer numbers stored under **IBRH[6:0]** and **IBRS[6:0]** are also swapped.
Any write access to an Input Buffer register while both **IBSYS** and **IBSYH** are set will cause the error flag **EIR.IIBA** to be set. In this case the Input Buffer will not be changed.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IBCR	R	IBSYS	0	0	0	0	0	0	0	0	IBRS6	IBRS5	IBRS4	IBRS3	IBRS2	IBRS1	IBRS0
0xD514	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	IBSYH	0	0	0	0	0	0	0	0	IBRH6	IBRH5	IBRH4	IBRH3	IBRH2	IBRH1	IBRH0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IBRH[6:0]

Input Buffer Request Host

Selects the target message buffer in the Message RAM for data transfer from Input Buffer. Valid values are 0x00 to 0x7F (0...127).

IBSYH

Input Buffer Busy Host

Set to ’1’ by writing **IBRH[6:0]** while **IBSYS** is still ’1’. After the ongoing transfer between IBF Shadow and the Message RAM has completed, the **IBSYH** is set back to ’0’.

1 = Request while transfer between IBF Shadow and Message RAM in progress

0 = No request pending

IBRS[6:0]

Input Buffer Request Shadow

Number of the target message buffer actually updated / lately updated. Valid values are 0x00 to 0x7F (0...127).

IBSYS

Input Buffer Busy Shadow

Set to '1' after writing **IBRH[6:0]**. When the transfer between IBF Shadow and the Message RAM has completed, **IBSYS** is set back to '0'.

1 = Transfer between IBF Shadow and Message RAM in progress

0 = Transfer between IBF Shadow and Message RAM completed

8.11 Output Buffer

Double buffer structure consisting of Output Buffer Host and Output Buffer Shadow. Used to read out message buffers from the Message RAM. While the Host can read from Output Buffer Host, the Message Handler transfers the selected message buffer from Message RAM to Output Buffer Shadow. The data transfer between Message RAM and Output Buffer (OBF) is described in Section 5.11.2.2 Data Transfer from Message RAM to Output Buffer.

8.11.1 Read Data Section [1...64] (RDDSn)

Holds the data words read from the data section of the addressed message buffer. The data words (DW_n) are read from the Message RAM in reception order from DW_1 (byte0, byte1) to DW_{PL} (PL = number of data words as defined by the payload length configured **RDHS2.PLC[6:0]**).

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDDSn	R	MD31	MD30	MD29	MD28	MD27	MD26	MD25	MD24	MD23	MD22	MD21	MD20	MD19	MD18	MD17	MD16
0xD600 - 0xD6FC	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	MD15	MD14	MD13	MD12	MD11	MD10	MD9	MD8	MD7	MD6	MD5	MD4	MD3	MD2	MD1	MD0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MD[31:0]

Message Data

MD[7:0] = DW_n, $byte_{n-1}$

MD[15:8] = DW_n, $byte_n$

MD[23:16] = DW_{n+1}, $byte_{n+1}$

MD[31:24] = DW_{n+1}, $byte_{n+2}$

Note: DW127 is located on RDDS64.MD[15:0]. In this case RDDS64.MD[31:16] is unused (no valid data). The Output Buffer RAMs are initialized to zero when leaving hard reset or by CHI command CLEAR_RAMS.

8.11.2 Read Header Section 1 (RDHS1)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDHS1	R	0	0	MBI	TXM	PPIT	CFG	CHB	CHA	0	CYC6	CYC5	CYC4	CYC3	CYC2	CYC1	CYC0
0xD700	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	0	0	0	0	FID10	FID9	FID8	FID7	FID6	FID5	FID4	FID3	FID2	FID1	FID0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Values as configured by the Host via WRHS1:

FID[10:0]

Frame ID

CYC[6:0]

Cycle Code

CHA, CHB

Channel Filter Control

CFG

Message Buffer Direction Configuration Bit

PPIT

Payload Preamble Indicator Transmit

TXM

Transmission Mode

MBI

Message Buffer Interrupt

In case that the message buffer read from the Message RAM belongs to the receive FIFO, **FID[10:0]** holds the received frame ID, while **CYC[6:0]**, **CHA**, **CHB**, **CFG**, **PPIT**, **TXM**, and **MBI** are reset to '0'.

8.11.3 Read Header Section 2 (RDHS2)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDHS2	R	0	PLR6	PLR5	PLR4	PLR3	PLR2	PLR1	PLR0	0	PLC6	PLC5	PLC4	PLC3	PLC2	PLC1	PLC0
0xD704	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	CRC10	CRC9	CRC8	CRC7	CRC6	CRC5	CRC4	CRC3	CRC2	CRC1	CRC0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CRC[10:0]

Header CRC (vRF!Header!HeaderCRC)

Receive Buffer: Header CRC updated from received data frames

Transmit Buffer: Header CRC calculated and configured by the Host

PLC[6:0]

Payload Length Configured

Length of data section (number of 2-byte words) as configured by the Host.

PLR[6:0]

Payload Length Received (vRF!Header!Length)

Payload length value updated from received data frames (exception: if message buffer belongs to the receive FIFO **PLR[6:0]** is also updated from received null frames)

When a message is stored into a message buffer the following behaviour with respect to payload length received and payload length configured is implemented:

PLR[6:0] > **PLC[6:0]**: The payload data stored in the message buffer is truncated to the payload length configured if **PLC[6:0]** even or else truncated to **PLC[6:0]** + 1.

PLR[6:0] ≤ **PLC[6:0]**: The received payload data is stored into the message buffers data section. The remaining data bytes of the data section as configured by **PLC[6:0]** are filled with undefined data

PLR[6:0] = zero: The message buffer's data section is filled with undefined data

PLC[6:0] = zero: Message buffer has no data section configured. No data is stored into the message buffer's data section.

Note: The Message RAM is organized in 4-byte words. When received data is stored into a message buffer's data section, the number of 2-byte data words written into the message buffer is **PLC[6:0]** rounded to the next even value. **PLC[6:0]** should be configured identical for all message buffers belonging to the receive FIFO. Header 2 is updated from data frames only.

8.11.4 Read Header Section 3 (RDHS3)

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDHS3	R	0	0	RES	PPI	NFI	SYN	SFI	RCI	0	0	RCC5	RCC4	RCC3	RCC2	RCC1	RCC0
0xD708	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	0	0	0	0	DP10	DP9	DP8	DP7	DP6	DP5	DP4	DP3	DP2	DP1	DP0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DP[10:0]

Data Pointer

Pointer to the first 32-bit word of the data section of the addressed message buffer in the Message RAM.

RCC[5:0]

Receive Cycle Count (vRF!Header!CycleCount)

Cycle counter value updated from received data frame.

RCI

Received on Channel Indicator (vSS!Channel)

Indicates the channel from which the received data frame was taken to update the respective receive buffer.

1 = Frame received on channel A

0 = Frame received on channel B

SFI

Startup Frame Indicator (vRF!Header!SuFIndicator)

A startup frame is marked by the startup frame indicator.

1 = The received frame is a startup frame

0 = The received frame is not a startup frame

SYN

Sync Frame Indicator (vRF!Header!SyFIndicator)

A sync frame is marked by the sync frame indicator.

1 = The received frame is a sync frame

0 = The received frame is not a sync frame

NFI

Null Frame Indicator (vRF!Header!NFIndicator)

Is set to '1' after storage of the first received data frame.

1 = At least one data frame has been stored into the respective message buffer

0 = Up to now no data frame has been stored into the respective message buffer

PPI

Payload Preamble Indicator (vRF!Header!PPIndicator)

The payload preamble indicator defines whether a network management vector or message ID is contained within the payload segment of the received frame.

1 = Static segment: Network management vector in the first part of the payload
Dynamic segment: Message ID in the first part of the payload

0 = The payload segment of the received frame does not contain a network management vector nor a message ID

RES

Reserved Bit (vRF!Header!Reserved)

Reflects the state of the received reserved bit. The reserved bit is transmitted as '0'.

Note: Header 3 is updated from data frames only.

8.11.5 Message Buffer Status (MBS)

The message buffer status is updated by the CC with respect to the assigned channel(s) latest at the end of the slot following the slot assigned to the message buffer. The flags are updated only when the CC is in NORMAL_ACTIVE or NORMAL_PASSIVE state. If only one channel (A or B) is assigned to a message buffer, the channel-specific status flags of the other channel are written to zero. If both channels are assigned to a message buffer, the channel-specific status flags of both channels are updated. The message buffer status is updated only when the slot counter reached the configured frame ID and when the cycle counter filter matched. When the Host updates a message buffer via Input Buffer, all MBS flags are reset to zero independent of which IBCM bits are set or not. For details about receive / transmit filtering see Sections 5.7 Filtering and Masking, 5.8 Transmit Process, and 5.9 Receive Process. Whenever the Message Handler changes one of the flags **VFRA**, **VFRB**, **SEOA**, **SEOB**, **CEOA**, **CEOB**, **SVOA**, **SVOB**, **TCIA**, **TCIB**, **ESA**, **ESB**, **MLST**, **FTA**, **FTB** the respective message buffer's **MBC** flag in registers MBSC1/2/3/4 is set.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MBS	R	0	0	RESS	PPIS	NFIS	SYNS	SFIS	RCIS	0	0	CCS5	CCS4	CCS3	CCS2	CCS1	CCS0
0xD70C	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	FTB	FTA	0	MLST	ESB	ESA	TCIB	TCIA	SVOB	SVOA	CEOB	CEOA	SEOB	SEOA	VFRB	VFRA
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

VFRA

Valid Frame Received on Channel A (vSS!ValidFrameA)

A valid frame indication is set if a valid frame was received on channel A.

1 = Valid frame received on channel A

0 = No valid frame received on channel A

VFRB

Valid Frame Received on Channel B (vSS!ValidFrameB)

A valid frame indication is set if a valid frame was received on channel B.

1 = Valid frame received on channel B

0 = No valid frame received on channel B

SEOA

Syntax Error Observed on Channel A (vSS!SyntaxErrorA)

A syntax error was observed in the assigned slot on channel A.

1 = Syntax error observed on channel A

0 = No syntax error observed on channel A

SEOB

Syntax Error Observed on Channel B (vSS!SyntaxErrorB)

A syntax error was observed in the assigned slot on channel B.

1 = Syntax error observed on channel B

0 = No syntax error observed on channel B

CEOA

Content Error Observed on Channel A (vSS!ContentErrorA)

A content error was observed in the assigned slot on channel A.

1 = Content error observed on channel A

0 = No content error observed on channel A

CEOB

Content Error Observed on Channel B (vSS!ContentErrorB)

A content error was observed in the assigned slot on channel B.

1 = Content error observed on channel B

0 = No content error observed on channel B

SVOA

Slot Boundary Violation Observed on Channel A (vSS!BViolationA)

A slot boundary violation (channel active at the start or at the end of the assigned slot) was observed on channel A.

1 = Slot boundary violation observed on channel A

0 = No slot boundary violation observed on channel A

SVOB

Slot Boundary Violation Observed on Channel B (vSS!BViolationB)

A slot boundary violation (channel active at the start or at the end of the assigned slot) was observed on channel B.

1 = Slot boundary violation observed on channel B

0 = No slot boundary violation observed on channel B

TCIA

Transmission Conflict Indication Channel A (vSS!TxConflictA)

A transmission conflict indication is set if a transmission conflict has occurred on channel A.

1 = Transmission conflict occurred on channel A

0 = No transmission conflict occurred on channel A

TCIB

Transmission Conflict Indication Channel B (vSS!TxConflictB)

A transmission conflict indication is set if a transmission conflict has occurred on channel B.

1 = Transmission conflict occurred on channel B

0 = No transmission conflict occurred on channel B

ESA

Empty Slot Channel A

In an empty slot there is no activity detected on the bus. The condition is checked in static and dynamic slots.

1 = No bus activity detected in the assigned slot on channel A

0 = Bus activity detected in the assigned slot on channel A

ESB

Empty Slot Channel B

In an empty slot there is no activity detected on the bus. The condition is checked in static and dynamic slots.

1 = No bus activity detected in the assigned slot on channel B

0 = Bus activity detected in the assigned slot on channel B

MLST

Message Lost

The flag is set in case the Host did not read the message before the message buffer was updated from a received data frame. Not affected by reception of null frames except for message buffers belonging to the receive FIFO. The flag is reset by a Host write to the message buffer via IBF or when a new message is stored into the message buffer **after** the message buffers **ND** flag was reset by reading out the message buffer via OBF.

1 = Unprocessed message was overwritten

0 = No message lost

FTA

Frame Transmitted on Channel A

Indicates that this node has transmitted a data frame in the configured slot on channel A.

1 = Data frame transmitted on channel A

0 = No data frame transmitted on channel A

FTB

Frame Transmitted on Channel B

Indicates that this node has transmitted a data frame in the configured slot on channel B.

1 = Data frame transmitted on channel B

0 = No data frame transmitted on channel B

Note: The FlexRay protocol specification requires that **FTA**, and **FTB** can only be reset by the Host. Therefore the Cycle Count Status **CCS[5:0]** for these bits is only valid for

the cycle where the bits are set to ’1’.

CCS[5:0]

Cycle Count Status

Actual cycle count when status was updated.

The following status bits are updated from both valid data and null frames. If no valid frame was received, the previous value is maintained.

RCIS

Received on Channel Indicator Status (vSS!Channel)

Indicates the channel on which the frame was received.

1 = Frame received on channel A

0 = Frame received on channel B

SFIS

Startup Frame Indicator Status (vRF!Header!SuFIndicator)

A startup frame is marked by the startup frame indicator.

1 = The received frame is a startup frame

0 = No startup frame received

SYNS

Sync Frame Indicator Status (vRF!Header!SyFIndicator)

A sync frame is marked by the sync frame indicator.

1 = The received frame is a sync frame

0 = No sync frame received

NFIS

Null Frame Indicator Status (vRF!Header!NFIndicator)

If set to ’0’ the payload segment of the received frame contains no usable data.

1 = Received frame is **not** a null frame

0 = Received frame is a null frame

PPIS

Payload Preamble Indicator Status (vRF!Header!PPIndicator)

The payload preamble indicator defines whether a network management vector or message ID is contained within the payload segment of the received frame.

1 = Static segment: Network management vector at the beginning of the payload Dynamic segment: Message ID at the beginning of the payload

0 = The payload segment of the received frame does not contain a network management vector or a message ID

RESS

Reserved Bit Status (vRF!Header!Reserved)

Reflects the state of the received reserved bit. The reserved bit is transmitted as ’0’.

8.11.6 Output Buffer Command Mask (OBCM)

Configures how the Output Buffer is updated from the message buffer in the Message RAM selected by register OBCR. When OBF Host and OBF Shadow are swapped, also mask bits **RDSH** and **RHSH** are swapped with bits **RDSS** and **RHSS** to keep them attached to the respective Output Buffer transfer.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OBCM	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	RDSH	RHSH
0xD710	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	RDSS	RHSS
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

RHSS

Read Header Section Shadow

1 = Header section selected for transfer from Message RAM to Output Buffer

0 = Header section is not read

RDSS

Read Data Section Shadow

1 = Data section selected for transfer from Message RAM to Output Buffer

0 = Data section is not read

RHSH

Read Header Section Host

1 = Header section selected for transfer from Message RAM to Output Buffer

0 = Header section is not read

RDSH

Read Data Section Host

1 = Data section selected for transfer from Message RAM to Output Buffer

0 = Data section is not read

Note: After the transfer of the header section from the Message RAM to OBF Shadow has completed, the message buffer status changed flag **MBC** of the selected message buffer in the MBSC1/2/3/4 registers is cleared. After the transfer of the data section from the Message RAM to OBF Shadow has completed, the new data flag **ND** of the selected message buffer in the NDAT1/2/3/4 registers is cleared.

8.11.7 Output Buffer Command Request (OBCR)

The message buffer selected by **OBRS[6:0]** is transferred from the Message RAM to the Output Buffer as soon as the Host has set **REQ** to '1'. Bit **REQ** can only be set to '1' while **OBSYS** is '0' (see also Section 5.11.2.2 Data Transfer from Message RAM to Output Buffer).
After setting **REQ** to '1', **OBSYS** is automatically set to '1', and the transfer of the message buffer selected by **OBRS[6:0]** from the Message RAM to OBF Shadow is started. When the transfer between the Message RAM and OBF Shadow has completed, this is signalled by setting **OBSYS** back to '0'. By setting the **VIEW** bit to '1' while **OBSYS** is '0', OBF Host and OBF Shadow are swapped. Now the Host can read the transferred message buffer from OBF Host. In parallel the Message Handler may transfer the next message from the Message RAM to OBF Shadow if **VIEW** and **REQ** are set at the same time.
Any write access to an Output Buffer register while **OBSYS** is set will cause the error flag **EIR.IOBA** to be set. In this case the Output Buffer will not be changed.

Bit		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OBCR	R	0	0	0	0	0	0	0	0	0	OBRH6	OBRH5	OBRH4	OBRH3	OBRH2	OBRH1	OBRH0
0xD714	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R	OBSYS	0	0	0	0	0	REQ	VIEW	0	OBRS6	OBRS5	OBRS4	OBRS3	OBRS2	OBRS1	OBRS0
	W																
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

OBRS[6:0]

Output Buffer Request Shadow

Number of source message buffer to be transferred from the Message RAM to OBF Shadow. Valid values are 0x00 to 0x7F (0...127). If the number of the first message buffer of the receive FIFO is written to this register the Message Handler transfers the message buffer addressed by the GET Index (GIDX, see Section 5.10 FIFO Function) to OBF Shadow.

VIEW

View Shadow Buffer

Toggles between OBF Shadow and OBF Host. Only writeable while **OBSYS** = '0'.

1 = Swap OBF Shadow and OBF Host

0 = No action

REQ

Request Message RAM Transfer

Requests transfer of message buffer addressed by **OBRS[6:0]** from Message RAM to OBF Shadow. Only writeable while **OBSYS** = '0'.

1 = Transfer to OBF Shadow requested
0 = No request

OBSYS

Output Buffer Busy Shadow

Set to ’1’ after setting bit **REQ**. When the transfer between the Message RAM and OBF Shadow has completed, **OBSYS** is set back to ’0’.

1 = Transfer between Message RAM and OBF Shadow in progress
0 = No transfer in progress

OBRH[6:0]

Output Buffer Request Host

Number of message buffer currently accessible by the Host via RDHS[1...3], MBS, and RDDS[1...64]. By writing **VIEW** to ’1’ OBF Shadow and OBF Host are swapped and the transferred message buffer is accessible by the Host. Valid values are 0x00 to 0x7F (0...127).